2010

Wavelet-based resampling techniques

Richard D. Kenderdine

University of Wollongong

Recommended Citation
NOTE

This online version of the thesis may have different page formatting and pagination from the paper copy held in the University of Wollongong Library.

UNIVERSITY OF WOLLONGONG

COPYRIGHT WARNING

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site. You are reminded of the following:

Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.
Wavelet-based Resampling Techniques

Richard D Kenderdine
BSc, SurvCert, GradDipAppSc(Industrial Maths), MAppStat

A thesis submitted in fulfilment of the requirements for the award of
the degree

Doctor of Philosophy

School of Mathematics and Applied Statistics

University of Wollongong

2010
CERTIFICATION

I, Richard D. Kenderdine, declare that this thesis, submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Mathematics and Applied Statistics, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Richard D Kenderdine
12 August 2010
Revised July 2011
Abstract

Resampling methods were first developed in the late 1970s with the aim of obtaining estimates of the error in a statistic calculated from a sample. A necessary restriction was that the observations in the sample had to be independent and various procedures have since been proposed to adapt the methods to dependent observations.

Wavelet transforms that permit data to be analysed from a frequency or time viewpoint simultaneously have also been developed over a similar period. In simple terms, a wavelet transform produces coefficients that are differences between adjacent averages over increasing scales and it has been noted that these differences are less correlated than the original data. The potential exists, therefore, for the wavelet transform to be applied to dependent data and the resulting coefficients, if deemed to be independent, can be resampled. Applying the inverse transform to the resampled coefficients may produce a surrogate set of data that shares similar characteristics of the original sample.

The aims of the thesis can be expressed thus:

1. to apply wavelet transforms, or variations thereof, to various types of dependent data and examine the resulting coefficients for independence;

2. if the coefficients are independent, apply the inverse transform and investigate whether the resulting surrogate data could have come from the population that produced the original sample

The types of dependent data considered are time series, restricted to AR(1) and MA(1) processes, one-dimensional point processes and two-dimensional point processes, both unmarked and marked.

The tone of the thesis is investigative, examining various methods and modifications that may be suitable, rather than seeking to justify one particular method.

The Discrete Wavelet Packet Transform, combined with Wavelet Lifting, was used to decompose time series and one-dimensional point processes into components whose elements were judged to be independent. These elements were resampled and the Inverse Transform subsequently yielded surrogate series. The surrogates of AR(1) and MA(1) time series appeared to be similar to the data only when the parameter was positive. The diagnostics for surrogates of homogeneous, non-homogeneous and clustered one-dimensional point processes indicated that the surrogates resembled the data.

The wavelet lifting philosophy was used as a basis for developing an algorithm for two-dimensional
point processes. Briefly, at each stage of the decomposition, the difference between the position co-ordinates of a point and the mean position of the point’s connected vertices is calculated. The resulting vector of differences is checked for independence and, if satisfied, is resampled. Recomposition yields the surrogate point patterns. Variations of the method were developed for clustered processes as well as qualitatively and quantitatively marked processes. In most instances the method or variations thereof were able to produce surrogates that could have come from the same population as the data.

*The wavelet transform thus provides a basis for developing algorithms that can produce surrogates of some types of time series and point processes.*
Acknowledgments

I acknowledge the supervision and assistance of Dr Pamela J Davy, School of Mathematics and Applied Statistics, University of Wollongong, in the preparation of this thesis.

The algorithms developed for this thesis were programmed using the computer algebra system Mathematica, Version 4.1 [42] and various add-on packages.

Extensive use was made of the R package spatstat [31], [4] for the diagnostics for two-dimensional surrogates.
# Contents

1 Introduction ........................................... 4
   1.1 Aims of the thesis ................................. 4
   1.2 Chapter summary .................................. 5

2 Literature Review .................................... 6
   2.1 Wavelets ........................................... 6
   2.2 Bootstrapping Time Series ....................... 6
   2.3 Bootstrapping 1D Point Patterns ............... 10
   2.4 Bootstrapping 2D Spatial Point Patterns ...... 11

3 Wavelet Theory ....................................... 16
   3.1 What are wavelets? ............................... 16
   3.2 The Wavelet Transform ........................... 17
      3.2.1 The Discrete Wavelet Transform (DWT) .... 18
      3.2.2 The Maximum Overlap DWT (MODWT) ....... 21
      3.2.3 Boundary Effects ............................. 21
      3.2.4 Which wavelet? ............................... 22
      3.2.5 An example ................................... 23
   3.3 Appropriately resampling a wavelet transform 23
      3.3.1 The Discrete Wavelet Packet Transform (DWPT) and Wavestrapping .. 25
      3.3.2 Wavestrapping the Manaus river data ....... 26
   3.4 The Lifting Scheme - Second Generation Wavelets 33
      3.4.1 Resampling the Manaus river data after Lifting .......... 34
   3.5 An issue to consider .............................. 37
   3.6 Conclusion ....................................... 38

4 Wavelet Resampling of Time Series ................. 42
   4.1 Introduction ..................................... 42
   4.2 Decorrelation properties ....................... 42
4.2.1 AR(1) processes .............................................. 43
4.2.2 MA(1) processes ............................................. 46
4.2.3 Conclusion ..................................................... 47
4.3 Parameter and variance estimates ................................... 47
4.3.1 Interim conclusion ............................................. 50
4.4 A closer look at the surrogates .................................... 54
4.4.1 AR(1) model with $\phi = 0.5$ .................................. 54
4.4.2 AR(1) model with $\phi = -0.5$ .............................. 57
4.4.3 MA(1) model with $\theta = 0.5$ ......................... 60
4.4.4 MA(1) model with $\theta = -0.5$ .......................... 62
4.5 A closer look at the white noise test ............................... 65
4.5.1 AR(1) model with $\phi = -0.5$ ......................... 66
4.5.2 MA(1) model with $\theta = -0.5$ .......................... 68
4.5.3 Results using the Box-Pierce test statistic ..................... 69
4.6 Final conclusion .................................................. 70

5 Wavelet Resampling of 1D Point Patterns ............................. 72
5.1 One-dimensional point processes ................................ 72
5.1.1 Simulated Data ................................................. 73
5.2 Resampling an homogeneous pattern ................................ 74
5.3 Resampling a non-homogeneous pattern ............................ 79
5.4 Resampling a clustered pattern ................................... 83
5.5 Testing the surrogates for homogeneity ............................. 87
5.5.1 Homogeneous patterns ........................................ 87
5.5.2 Non-homogeneous patterns ................................... 88
5.5.3 Clustered patterns ............................................. 92
5.6 Conclusion ........................................................ 92

6 Wavelet Resampling of 2-dimensional Point Patterns ................. 94
6.1 Introduction ..................................................... 94
6.2 Spatial Point Patterns ........................................... 94
6.3 Lifting point patterns ............................................ 95
6.3.1 Preparing the data ............................................ 95
6.3.2 The decomposition algorithm ............................... 96
6.3.3 The first level of the decomposition ......................... 99
6.3.4 The subsequent levels of the decomposition ................ 99
6.3.5 Assumptions behind the resampling procedure .............. 100
6.3.6 Resampling and recomposition .............................. 100
10 2-D Poisson Data with Qualitative Marks

10.1 Introduction ................................................................. 179
10.2 Simulated Poisson point pattern ....................................... 179
10.3 Method 1: Superposition .................................................. 182
10.4 Method 2: Randomly reallocated marks ............................... 189
10.5 Method 3: The two differences method ............................... 193
10.6 A simple example of the two differences method ..................... 194
10.7 Two differences method applied to the Poisson pattern ............. 197
10.8 Analysis of the three methods ............................................ 201
10.9 Conclusion ................................................................. 202

11 Example of Data with Qualitative Marks

11.1 Introduction ................................................................. 203
11.2 Analysing the surrogates .................................................. 205
11.3 The superposition method ................................................ 205
11.4 The two differences method .............................................. 215
11.5 Comparison of the two methods ........................................ 216

12 2-D Pattern with Quantitative Marks

12.1 Introduction ................................................................. 223
12.2 The algorithm ............................................................... 223
12.3 Data .................................................................. 225
12.3.1 Assumptions .............................................................. 225
12.4 Diagnostics ................................................................. 226
12.4.1 Summary characteristics ............................................ 226
12.4.2 Second-order characteristics ..................................... 227
12.5 Data analysis ............................................................... 227
12.5.1 Boese data .............................................................. 227
12.5.2 Spruce data .............................................................. 235
12.5.3 Longleaf data ........................................................... 243
12.6 Modifying the algorithm - resampling the location ................. 249
12.7 Conclusion ................................................................. 255

13 Alternative Resampling Methods for 2-D

13.1 Introduction ................................................................. 256
13.1.1 Tiling methods .......................................................... 256
13.1.2 Marked point method ................................................ 257
13.2 Analysis of block resampling ............................................ 258
13.2.1 Non-homogeneous Poisson point pattern ................. 259
13.2.2 Matern Cluster point pattern .......................... 263
13.3 Analysis of the marked point method ......................... 265
13.3.1 Non-homogeneous Poisson pattern - marks as the number of points within radii ................................. 267
13.3.2 Non-homogeneous Poisson pattern - marks as the isotropic correction weights 267
13.3.3 Non-homogeneous Poisson pattern - marks as the number of points within annuli ........................................ 268
13.3.4 Matern Cluster pattern - marks as the number of points within radii ............................................. 269
13.3.5 Matern Cluster pattern - marks as the isotropic correction weights ................................. 270
13.4 Conclusion ......................................................... 271

14 Conclusion ........................................................ 280
List of Figures

3.1 The Gaussian wavelet .................................................... 16
3.2 The Haar and Mexican hat wavelets ................................. 17
3.3 Manaus river: data and DWT using the LA(8) wavelet .......... 24
3.4 Manaus river: ACFs for the detail coefficients using DWT with LA(8) ......................................................... 25
3.5 Manaus river: data and MODWT using the LA(8) wavelet ......................................................... 26
3.6 Manaus river: ACFs for the detail coefficients using MODWT ......................................................... 27
3.7 DWT level diagram ......................................................... 28
3.8 DWPT level diagram ......................................................... 29
3.9 Manaus river: ACFs for the detail coefficients in Levels 1 and 2 using DWPT with LA(8) ......................................................... 29
3.10 Manaus river: ACFs for the detail coefficients in $W(3,0) - W(3,5)$ using DWPT with LA(8) ......................................................... 30
3.11 Manaus river: ACFs for the detail coefficients in $W(3,6) - W(4,3)$ using DWPT with LA(8) ......................................................... 30
3.12 Manaus river: data and surrogates using DWPT with LA(8) ......................................................... 31
3.13 Manaus river: data and mean of surrogates and extreme surrogate values using DWPT with LA(8) ......................................................... 31
3.14 Manaus river: data and surrogates using DWPT with Haar ......................................................... 32
3.15 Manaus river: data and mean of surrogates and extreme surrogate values using DWPT with Haar ......................................................... 33
3.16 Calculation of the wavelet coefficients in the lifting procedure ......................................................... 35
3.17 Manaus river: data and typical surrogates using lifting with original white-noise test and extrapolation for the final predicted value ......................................................... 36
3.18 Calculation of the detail coefficients in the lifting procedure with the lazy wavelet at the end ......................................................... 37
3.19 Manaus river: data and typical surrogates using lifting with altered white-noise test and extrapolation for the final predicted value ......................................................... 38
3.20 Manaus river: ACFs for the data and coefficients in level 1 using lifting ......................................................... 39
LIST OF FIGURES
3.21 Manaus river: ACFs for the data and coefficients in level 2 using lifting . . . . . .

xi
40

3.22 Manaus river: ACFs for the data and coefficients in crystals 3.4 - 3.7 decomposition
using lifting . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

40

3.23 Manaus river: data and typical surrogate using lifting with altered white-noise test
and using the lazy wavelet for the final predicted value . . . . . . . . . . . . . . . .

41

3.24 Manaus river using lifting with altered white-noise test and using the lazy wavelet
for the final predicted value: data and mean of surrogates and extreme surrogate
values . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

41

4.1

ACFs for level 1 coefficients after lifting an AR(1) process . . . . . . . . . . . . . .

45

4.2

ACF for Level 1 detail coefficients after lifting an MA(1) process . . . . . . . . . .

46

4.3

ACF for Level 1 smooth coefficients after lifting an MA(1) process . . . . . . . . .

47

4.4

Basis for AR(1) model with φ = 0.9 . . . . . . . . . . . . . . . . . . . . . . . . . .

53

4.5

Basis for AR(1) model with φ = −0.9 . . . . . . . . . . . . . . . . . . . . . . . . .

53

4.6

Basis for MA(1) model with θ = 0.9 . . . . . . . . . . . . . . . . . . . . . . . . . .

53

4.7

Basis for MA(1) model with θ = −0.9 . . . . . . . . . . . . . . . . . . . . . . . . .

53

4.8

AR(1) with φ = 0.5 : data and one surrogate series . . . . . . . . . . . . . . . . .

54

4.9

AR(1) with φ = 0.5 : data and mean of 1000 surrogates . . . . . . . . . . . . . . .

55

4.10 AR(1) with φ = 0.5 : a selection of surrogates . . . . . . . . . . . . . . . . . . . .

55

4.11 AR(1) with φ = 0.5 : ACF for data and three surrogates . . . . . . . . . . . . . .

56

4.12 AR(1) with φ = 0.5 : spectrum for data and three surrogates . . . . . . . . . . . .

56

4.13 AR(1) with φ = −0.5 : data and one surrogate series . . . . . . . . . . . . . . . .

57

4.14 AR(1) with φ = −0.5 : data and mean of 1000 surrogates . . . . . . . . . . . . . .

58

4.15 AR(1) with φ = −0.5 : a selection of surrogates . . . . . . . . . . . . . . . . . . .

58

4.16 AR(1) with φ = −0.5 : ACF for data and three surrogates . . . . . . . . . . . . .

59

4.17 AR(1) with φ = −0.5 : spectrum for data and three surrogates . . . . . . . . . . .

59

4.18 MA(1) with θ = 0.5 : data and one surrogate series . . . . . . . . . . . . . . . . .

60

4.19 MA(1) with θ = 0.5 : data and mean of 1000 surrogates . . . . . . . . . . . . . . .

60

4.20 MA(1) with θ = 0.5 : a selection of surrogates . . . . . . . . . . . . . . . . . . . .

61

4.21 MA(1) with θ = 0.5 : ACF for data and three surrogates . . . . . . . . . . . . . .

61

4.22 MA(1) with θ = 0.5 : spectrum for data and three surrogates . . . . . . . . . . . .

62

4.23 MA(1) with θ = −0.5 : data and one surrogate series . . . . . . . . . . . . . . . .

63

4.24 MA(1) with θ = −0.5 : data and mean of 1000 surrogates . . . . . . . . . . . . . .

63

4.25 MA(1) with θ = −0.5 : a selection of surrogates . . . . . . . . . . . . . . . . . . .

64

4.26 MA(1) with θ = −0.5 : ACF for data and three surrogates . . . . . . . . . . . . .

64

4.27 MA(1) with θ = −0.5 : spectrum for data and three surrogates . . . . . . . . . . .

65

4.28 AR(1) with φ = −0.5 : autocorrelation coefficients for packets at Levels 1 and 2 .

67


<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.29</td>
<td>AR(1) with $\phi = -0.5$ : autocorrelation coefficients for packets at Level 3</td>
<td>67</td>
</tr>
<tr>
<td>4.30</td>
<td>MA(1) with $\theta = -0.5$ : autocorrelation coefficients for packets at Levels 1 and 2</td>
<td>68</td>
</tr>
<tr>
<td>4.31</td>
<td>MA(1) with $\theta = -0.5$ : autocorrelation coefficients for packets at Level 3</td>
<td>69</td>
</tr>
<tr>
<td>5.1</td>
<td>Homogeneous process with $\lambda = 4$: Cumulative number of events by time, number of events within consecutive time periods and number of events by event number for one surrogate</td>
<td>75</td>
</tr>
<tr>
<td>5.2</td>
<td>Number of events by radius with $\lambda = 4$</td>
<td>77</td>
</tr>
<tr>
<td>5.3</td>
<td>Histograms showing events within periods 4 to 11 with $\lambda = 4$</td>
<td>78</td>
</tr>
<tr>
<td>5.4</td>
<td>Homogeneous process with $\lambda(t) = t$: Cumulative number of events by time, number of events within consecutive time periods and number of events by event number for one surrogate</td>
<td>79</td>
</tr>
<tr>
<td>5.5</td>
<td>Number of events by radius under non-homogeneous process with $\lambda(t) = t$</td>
<td>81</td>
</tr>
<tr>
<td>5.6</td>
<td>Histograms showing events within periods 4 to 11 with $\lambda(t) = t$</td>
<td>82</td>
</tr>
<tr>
<td>5.7</td>
<td>Clustered process: Cumulative number of events by time, number of events within consecutive time periods and number of events by event number for one surrogate</td>
<td>84</td>
</tr>
<tr>
<td>5.8</td>
<td>Number of events by radius under the clustered process</td>
<td>85</td>
</tr>
<tr>
<td>5.9</td>
<td>Histograms of events within periods 4 to 11 under the clustered process</td>
<td>86</td>
</tr>
<tr>
<td>5.10</td>
<td>Scatterplot showing value of the test statistic, assuming homogeneity, versus intensity for the simulated homogeneous series</td>
<td>89</td>
</tr>
<tr>
<td>5.11</td>
<td>Boxplots of the test statistics for the homogeneous surrogates</td>
<td>89</td>
</tr>
<tr>
<td>5.12</td>
<td>Scatterplot showing value of the test statistic, assuming homogeneity, versus $\frac{d}{e}$ ratio for the non-homogeneous patterns, and $</td>
<td>\lambda_2 - \lambda_1</td>
</tr>
<tr>
<td>5.13</td>
<td>Boxplots of the test statistics for the non-homogeneous surrogates</td>
<td>91</td>
</tr>
<tr>
<td>5.14</td>
<td>Boxplots of the test statistics for the clustered surrogates</td>
<td>93</td>
</tr>
<tr>
<td>6.1</td>
<td>Sample data: point locations and bounded Voronoi diagram</td>
<td>96</td>
</tr>
<tr>
<td>6.2</td>
<td>Voronoi diagram and triangulation with boundary points included</td>
<td>97</td>
</tr>
<tr>
<td>6.3</td>
<td>Updated data after deleting the first point</td>
<td>99</td>
</tr>
<tr>
<td>6.4</td>
<td>Levels 2 to 11 of the decomposition</td>
<td>101</td>
</tr>
<tr>
<td>6.5</td>
<td>Differences between deleted points and mean of their neighbours</td>
<td>101</td>
</tr>
<tr>
<td>6.6</td>
<td>Stages 11 to 2 of the recomposition</td>
<td>103</td>
</tr>
<tr>
<td>6.7</td>
<td>Triangulation of original data and a surrogate</td>
<td>103</td>
</tr>
<tr>
<td>6.8</td>
<td>Plot of simulated Poisson(50) data</td>
<td>107</td>
</tr>
<tr>
<td>6.9</td>
<td>Simulated Poisson data and output for quadrat test</td>
<td>108</td>
</tr>
<tr>
<td>6.10</td>
<td>KS test using $f(x, y) = x + y$ and $f(x, y) = xy$</td>
<td>108</td>
</tr>
<tr>
<td>6.12</td>
<td>Estimates of $G$ and $K$ functions</td>
<td>109</td>
</tr>
</tbody>
</table>
6.13 Bounded Voronoi diagram for simulated Poisson(50) point pattern
6.14 Delaunay triangulation after removing 35 points and all points
6.15 Differences in the $x$ and $y$ coordinates and ACFs
6.16 Plot of simulated Poisson(50) point pattern and a typical surrogate
6.17 Histograms of number of counts in specified cells for Poisson(50) surrogates
6.18 Estimated $F$ and $G$ functions for Poisson surrogates
6.19 Estimated $J$ and $K$ functions for Poisson surrogates
6.20 Simulated Thomas (10, 0.1, 5) data and output for quadrat test
6.21 KS test using $f(x,y) = x + y$ and $f(x,y) = xy$
6.22 Plots of $F$, $G$, $J$ and $K$ functions for simulated Thomas pattern
6.23 Estimated $G$ and $K$ functions for simulated Thomas pattern
6.24 Bounded Voronoi diagram for simulated Thomas (10, 0.1, 5) data
6.25 Delaunay triangulation for simulated Thomas data after removing 25 points
6.26 Delaunay triangulation for simulated Thomas data after removing all points
6.27 Differences in the $x$ and $y$ coordinates and ACFs
6.28 Plot of simulated Thomas pattern and a typical surrogate
6.29 Histograms of number of counts in specified cells for Thomas surrogates
6.30 Estimated $F$, $G$, $G$ and $K$ functions for Thomas surrogates
6.31 Histograms of parameter estimates: $\kappa$, $\sigma$ and $\mu$ for Thomas surrogates
6.32 Theoretical Thomas $K$ function with two values of $(\kappa, \sigma)$
6.33 Thomas $K(r) - \pi r^2$ by $\kappa$ for three radii and values of $\sigma$ and $\kappa$
6.34 Plot of simulated Strauss data with intensity = 50, interaction parameter = 0.5
and interaction radius = 0.05
6.35 Simulated Strauss point pattern: data and output for quadrat test
6.36 KS test for Strauss data using $f(x,y) = x + y$ and $f(x,y) = xy$
6.37 Plots of $F$, $G$, $J$ and $K$ functions for simulated Strauss data
6.38 Bounded Voronoi diagram and Delaunay triangulation after removing 25 points
6.39 Delaunay triangulation for simulated Strauss data after removing all points
6.40 Differences in the $x$ and $y$ coordinates and ACFs
6.41 Simulated Strauss pattern and a typical surrogate
6.42 Histograms of number of counts in specified cells for Strauss surrogates
6.43 Strauss surrogates: estimated $F$ and $G$ functions
6.44 Strauss surrogates: estimated $J$ and $K$ functions
6.45 Simulated Strauss(50, 0.8, 0.05) pattern: data and one surrogate
6.46 Strauss(50, 0.8, 0.05) surrogates: estimated $F$ and $G$ functions
6.47 Strauss(50, 0.8, 0.05) surrogates: estimated $J$ and $K$ functions
6.48 Simulated Strauss(50, 0.2, 0.05) pattern: data and one surrogate
<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.49</td>
<td>Strauss(50, 0.2, 0.05) surrogates: estimated F and G functions</td>
</tr>
<tr>
<td>6.50</td>
<td>Strauss(50, 0.2, 0.05) surrogates: estimated J and K functions</td>
</tr>
<tr>
<td>7.1</td>
<td>Non-homogeneous Poisson data: $\lambda(x) = 50 + 100x$</td>
</tr>
<tr>
<td>7.2</td>
<td>Differences in the x and y coordinates and ACFs</td>
</tr>
<tr>
<td>7.3</td>
<td>Non-homogeneous Poisson: quadrant counts for data and estimated intensity</td>
</tr>
<tr>
<td>7.4</td>
<td>Histograms of quadrant counts for 1000 surrogates</td>
</tr>
<tr>
<td>7.5</td>
<td>Point locations with overlaid data and estimated intensity for some surrogates</td>
</tr>
<tr>
<td>7.6</td>
<td>Histograms of fitted coefficients for modelling $\lambda(x) = 50 + 100x$</td>
</tr>
<tr>
<td>7.7</td>
<td>Non-homogeneous K and Pair Correlation Functions estimated from data</td>
</tr>
<tr>
<td>7.8</td>
<td>Pair Correlation Function estimated from $K_{inhom}$ and Pair Correlation Function simulation envelope</td>
</tr>
<tr>
<td>7.9</td>
<td>$K_{inhom}(r)$ simulation envelope</td>
</tr>
<tr>
<td>7.10</td>
<td>Non-homogeneous Poisson data: $\lambda(x, y) = 200x + 200y$</td>
</tr>
<tr>
<td>7.11</td>
<td>Differences in the x and y coordinates and ACFs for 2D intensity function</td>
</tr>
<tr>
<td>7.12</td>
<td>Quadrat counts for data and estimated intensity</td>
</tr>
<tr>
<td>7.13</td>
<td>Histograms of quadrant counts for 1000 surrogates</td>
</tr>
<tr>
<td>7.14</td>
<td>Point locations with overlaid data and estimated intensity for typical surrogates</td>
</tr>
<tr>
<td>7.15</td>
<td>Histograms of fitted coefficients for modelling intensity $\lambda(x, y) = 200x + 200y$</td>
</tr>
<tr>
<td>7.16</td>
<td>Non-homogeneous K and Pair Correlation Functions estimated from data</td>
</tr>
<tr>
<td>7.17</td>
<td>Pair Correlation Function estimated from the derivative of $K_{inhom}(r)$ and spatstat’s simulation envelope</td>
</tr>
<tr>
<td>7.18</td>
<td>$K_{inhom}(r)$ simulation envelope</td>
</tr>
<tr>
<td>8.1</td>
<td>Matern cluster pattern: data and four surrogates</td>
</tr>
<tr>
<td>8.2</td>
<td>Differences in the x and y coordinates and ACFs for Matern cluster example</td>
</tr>
<tr>
<td>8.3</td>
<td>Matern cluster surrogates: estimated F and G functions</td>
</tr>
<tr>
<td>8.4</td>
<td>Matern cluster surrogates: estimated J and K functions</td>
</tr>
<tr>
<td>8.5</td>
<td>Matern cluster surrogates: estimates of the K function</td>
</tr>
<tr>
<td>8.6</td>
<td>Matern cluster surrogates: estimated mean Pair Correlation Function</td>
</tr>
<tr>
<td>8.7</td>
<td>Matern cluster surrogates: histograms of estimators of $\kappa$ and radius</td>
</tr>
<tr>
<td>8.8</td>
<td>Data and one surrogate using parameters $\kappa = 15$, $R = 0.1$ and $\mu = 10$ (Run 1)</td>
</tr>
<tr>
<td>8.9</td>
<td>Data and one surrogate using parameters $\kappa = 15$, $R = 0.1$ and $\mu = 15$ (Run 2)</td>
</tr>
<tr>
<td>8.10</td>
<td>Data and one surrogate using parameters $\kappa = 10$, $R = 0.1$ and $\mu = 10$ (Run 3)</td>
</tr>
<tr>
<td>8.11</td>
<td>Data and one surrogate using parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)</td>
</tr>
<tr>
<td>8.12</td>
<td>Data and one surrogate using parameters $\kappa = 20$, $R = 0.05$ and $\mu = 10$ (Run 5)</td>
</tr>
<tr>
<td>8.13</td>
<td>Data and one surrogate using parameters $\kappa = 10$, $R = 0.1$ and $\mu = 20$ (Run 6)</td>
</tr>
<tr>
<td>8.14</td>
<td>Vector of differences in x and y coordinates for Runs 1 - 3</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

8.15 Vector of differences in \(x\) and \(y\) coordinates for Runs 4 - 6 . . . . . . . . . 161
8.16 Surrogates from Run 4 using differences unadjusted and adjusted for trend . . . . 161
8.17 Estimated \(G\) function for Run 4 without and after adjusting for trend in differences 162
8.18 Estimated \(K\) function for Run 4 without and after adjusting for trend in differences 162
8.19 Two difference vectors: data overlaid with one surrogate and estimated \(K\) function 164
8.20 Run 4 points separated into clusters . . . . . . . . . . . . . . . . . . . . . . . . . . . 165
8.21 Run 4 cluster centres and four surrogates of the cluster centres . . . . . . . . . . . 165
8.22 Run 4 surrogates using two-stage resampling . . . . . . . . . . . . . . . . . . . . . . 166
8.23 Run 4 using two-stage resampling: estimates of \(F\) and \(G\) functions . . . . . . . . 166
8.24 Run 4 using two-stage resampling: estimates of \(J\) and \(K\) functions . . . . . . . . 167
8.25 Run 4 using two-stage resampling: estimates of the Pair Correlation Function . . . 167
8.26 Run 4 using two-stage resampling: envelope of the \(K\) function . . . . . . . . . . . 168
8.27 Run 4 using two-stage resampling: differences between cluster centres and coordi-
nates of the points . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 169
8.28 Run 4 using two-stage resampling: surrogates and \(K\) function estimators . . . . . 169
8.29 Run 4 using two-stage resampling: estimators of the Pair Correlation Function . . . 169
8.30 Data from Section 3 separated into clusters . . . . . . . . . . . . . . . . . . . . . . . 170
8.31 Cluster centres . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 171
8.32 Differences in point coordinates and cluster centre . . . . . . . . . . . . . . . . . . 171
8.33 Four surrogates using two-stage resampling . . . . . . . . . . . . . . . . . . . . . . . 171
8.34 Estimates of the \(K\) and Pair Correlation Functions . . . . . . . . . . . . . . . . . 172

9.1 Data overlaid with cluster centres and points separated into clusters . . . . . . . . 174
9.2 Differences between coordinates of points and the relevant cluster centre . . . . . . 174
9.3 Four surrogates . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 175
9.4 Quadrat count histograms for the surrogates . . . . . . . . . . . . . . . . . . . . . . . 176
9.5 Histograms for coefficients of model fitted to 100 surrogates . . . . . . . . . . . . 178
9.6 Estimates of the \(K(r)\) and the Pair Correlation Functions . . . . . . . . . . . . . 178

10.1 Marked Poisson data with two types with the same intensity . . . . . . . . . . . . 180
10.2 Differences in the \(x\) and \(y\) coordinates and ACFs for Type 1 using superposition . . 182
10.3 Differences in the \(x\) and \(y\) coordinates and ACFs for Type 1 using superposition . . 183
10.4 Four surrogates using the superposition method . . . . . . . . . . . . . . . . . . . . 183
10.5 Diagnostics from surrogates using superposition: \(G_{12}\) and \(J_{12}\) . . . . . . . . 184
10.6 Diagnostics from surrogates using superposition: \(K_{12}\) and \(L_{12}\) . . . . . . . . 185
10.7 Diagnostics from surrogates using superposition: \(G_{21}\) and \(J_{21}\) . . . . . . . . 185
10.8 Diagnostics from surrogates using superposition: \(K_{21}\) and \(L_{21}\) . . . . . . . . 186
10.9 Diagnostics from surrogates using superposition: \(G_{1*}\) and \(K_{1*}\) . . . . . . . . 186
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10</td>
<td>Diagnostics from surrogates using superposition: $G_{2*}$ and $K_{2*}$</td>
<td>187</td>
</tr>
<tr>
<td>10.11</td>
<td>Diagnostics from surrogates using superposition: $L_{1*}$ and $L_{2*}$</td>
<td>187</td>
</tr>
<tr>
<td>10.12</td>
<td>Mark Correlation Function using superposition</td>
<td>188</td>
</tr>
<tr>
<td>10.13</td>
<td>Differences in the $x$ and $y$ coordinates and ACFs using random reallocation</td>
<td>190</td>
</tr>
<tr>
<td>10.14</td>
<td>Four surrogates using the random mark reallocation method</td>
<td>190</td>
</tr>
<tr>
<td>10.15</td>
<td>Diagnostics from surrogates using random mark reallocation: $G_{12}$ and $K_{12}$</td>
<td>191</td>
</tr>
<tr>
<td>10.16</td>
<td>Diagnostics from surrogates using random mark reallocation: $G_{21}$ and $K_{21}$</td>
<td>191</td>
</tr>
<tr>
<td>10.17</td>
<td>Diagnostics from surrogates using random mark reallocation: $G_{1*}$ and $K_{1*}$</td>
<td>192</td>
</tr>
<tr>
<td>10.18</td>
<td>Diagnostics from surrogates using random mark reallocation: $G_{2*}$ and $K_{2*}$</td>
<td>192</td>
</tr>
<tr>
<td>10.19</td>
<td>Mark Correlation Function using random mark reallocation</td>
<td>193</td>
</tr>
<tr>
<td>10.20</td>
<td>Subset of a marked Poisson process and one surrogate using two differences</td>
<td>196</td>
</tr>
<tr>
<td>10.21</td>
<td>Triangulation of the data in Figure 10.20(a) ignoring marks and data with inserted boundary points</td>
<td>196</td>
</tr>
<tr>
<td>10.22</td>
<td>Differences in the $x$ and $y$ coordinates and ACFs for points being lifted using two differences</td>
<td>197</td>
</tr>
<tr>
<td>10.23</td>
<td>Differences in the $x$ and $y$ coordinates and ACFs for points not being lifted using two differences</td>
<td>197</td>
</tr>
<tr>
<td>10.24</td>
<td>Four surrogates using the two differences method</td>
<td>198</td>
</tr>
<tr>
<td>10.25</td>
<td>Diagnostics for surrogates using with two differences: $G_{12}$ and $K_{12}$</td>
<td>198</td>
</tr>
<tr>
<td>10.26</td>
<td>Diagnostics for surrogates using with two differences: $G_{21}$ and $K_{21}$</td>
<td>199</td>
</tr>
<tr>
<td>10.27</td>
<td>Diagnostics for surrogates using with two differences: $G_{1*}$ and $K_{1*}$</td>
<td>199</td>
</tr>
<tr>
<td>10.28</td>
<td>Diagnostics for surrogates using with two differences: $G_{2*}$ and $K_{2*}$</td>
<td>200</td>
</tr>
<tr>
<td>10.29</td>
<td>Mark Correlation Function using two differences</td>
<td>200</td>
</tr>
<tr>
<td>11.1</td>
<td>Lansing Woods data</td>
<td>204</td>
</tr>
<tr>
<td>11.2</td>
<td>Lansing Woods data: Location of black oak and miscellaneous trees</td>
<td>204</td>
</tr>
<tr>
<td>11.3</td>
<td>Differences in the $x$ and $y$ coordinates and ACFs for Type 1 using superposition</td>
<td>205</td>
</tr>
<tr>
<td>11.4</td>
<td>Differences in the $x$ and $y$ coordinates and ACFs for Type 1 using superposition</td>
<td>206</td>
</tr>
<tr>
<td>11.5</td>
<td>Four of the surrogates using superposition</td>
<td>206</td>
</tr>
<tr>
<td>11.6</td>
<td>Diagnostics from surrogates using superposition: $G_{12}(r)$ and $K_{12}(r)$</td>
<td>207</td>
</tr>
<tr>
<td>11.7</td>
<td>Diagnostics from surrogates using superposition: $G_{21}(r)$ and $K_{21}(r)$</td>
<td>208</td>
</tr>
<tr>
<td>11.8</td>
<td>Diagnostics from surrogates using superposition: $L_{12}(r)$ and $L_{21}(r)$</td>
<td>208</td>
</tr>
<tr>
<td>11.9</td>
<td>Diagnostics from surrogates using superposition: $G_{1*}(r), K_{1*}(r), G_{2*}(r)$ and $K_{2*}(r)$</td>
<td>209</td>
</tr>
<tr>
<td>11.10</td>
<td>Diagnostics from surrogates using superposition: $L_{1*}(r)$ and $L_{2*}(r)$</td>
<td>210</td>
</tr>
<tr>
<td>11.11</td>
<td>Diagnostics from surrogates using superposition: Mark Correlation and Partial Pair Correlation Functions</td>
<td>212</td>
</tr>
<tr>
<td>11.12</td>
<td>Diagnostics from surrogates using superposition: $\delta_{12}(r)$ and $\delta_{21}(r)$</td>
<td>213</td>
</tr>
</tbody>
</table>
11.13 Mark Connection Function $p_{11}(r)$ using superposition ........................................ 213
11.14 Mark Connection Function $p_{12}(r)$ using superposition ........................................ 214
11.15 Mark Connection Function $p_{22}(r)$ using superposition ........................................ 214
11.16 Differences in the $x$ and $y$ coordinates and ACFs for points being lifted using two
   differences ...................................................................................................................... 215
11.17 Differences in the $x$ and $y$ coordinates and ACFs for points being lifted using two
   differences ...................................................................................................................... 215
11.18 Four of the surrogates using the two differences method ................................................ 216
11.19 Diagnostics from surrogates using two differences: $G_{12}(r)$ and $K_{12}(r)$ ................... 217
11.20 Diagnostics from surrogates using two differences: $G_{21}(r)$ and $K_{21}(r)$ ................... 218
11.21 Diagnostics from surrogates using two differences: $L_{12}(r)$ and $L_{21}(r)$ .................... 218
11.22 Diagnostics from surrogates using two differences: $G_{1\bullet}(r)$ and $K_{1\bullet}(r)$ .............. 218
11.23 Diagnostics from surrogates using two differences: $G_{2\bullet}(r)$ and $K_{2\bullet}(r)$ .............. 219
11.24 Diagnostics from surrogates using two differences: $L_{1\bullet}(r)$ and $L_{2\bullet}(r)$ .............. 219
11.25 Diagnostics from surrogates using two differences: Mark Correlation and Pair Cor-
   relation Functions ........................................................................................................... 220
11.26 Diagnostics from surrogates using two differences: $\delta_{12}(r)$ and $\delta_{21}(r)$ ................. 221
11.27 Diagnostics from surrogates using two differences: the mark connection functions
   $p_{11}(r), p_{12}(r)$ and $p_{22}(r)$ ...................................................................................... 221
11.28 Comparison of the estimates of the delta functions: $\delta_{12}(r)$ and $\delta_{21}(r)$ ................. 222
11.29 Comparison of the estimates of the MCFs: $p_{11}(r), p_{12}(r)$ and $p_{22}(r)$ ...................... 222
12.1  Boese data with mark information ................................................................................... 228
12.2  Differences in the marks and ACF for Boese data .......................................................... 228
12.3  Two surrogates ............................................................................................................... 229
12.4  Two more surrogates .................................................................................................... 230
12.5  Comparison of the markstat function for the data and the mean function for the
   surrogates ........................................................................................................................... 232
12.6  DBH differentiation (2) index for the data and location of specific points .................... 232
12.7  DBH differentiation (2) index for the surrogates .............................................................. 233
12.8  Mark Correlation Functions .......................................................................................... 234
12.9  $K_{multi}$ plots as given by spatstat .................................................................................. 234
12.10 Weighted prediction: Mark Correlation Function ......................................................... 235
12.11 Weighted prediction: $K_{multi}$ plots as given by spatstat .............................................. 236
12.12 Weighted prediction: difference between the markstat function for the data and the
   mean function for the surrogates ....................................................................................... 236
12.13 Spruce data with mark information ................................................................................ 238
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.14</td>
<td>Differences in the marks and ACF for Spruce data</td>
<td>239</td>
</tr>
<tr>
<td>12.15</td>
<td>Spruce data: two surrogates</td>
<td>239</td>
</tr>
<tr>
<td>12.16</td>
<td>Spruce data: two more surrogates</td>
<td>240</td>
</tr>
<tr>
<td>12.17</td>
<td>Spruce data: difference between the <code>markstat</code> function for the data and the mean function for the surrogates</td>
<td>240</td>
</tr>
<tr>
<td>12.18</td>
<td>Spruce data: DBH differentiation (2) index for the data and location</td>
<td>241</td>
</tr>
<tr>
<td>12.19</td>
<td>Spruce data: DBH differentiation (2) index for the surrogates</td>
<td>241</td>
</tr>
<tr>
<td>12.20</td>
<td>Spruce data: Mark Correlation Functions</td>
<td>242</td>
</tr>
<tr>
<td>12.21</td>
<td>Spruce data: Kmulti plots as given by <code>spatstat</code></td>
<td>243</td>
</tr>
<tr>
<td>12.22</td>
<td>Longleaf data: location of trees</td>
<td>244</td>
</tr>
<tr>
<td>12.23</td>
<td>Longleaf data with mark information</td>
<td>244</td>
</tr>
<tr>
<td>12.24</td>
<td>Differences in the marks and ACF for Longleaf data</td>
<td>245</td>
</tr>
<tr>
<td>12.25</td>
<td>Longleaf data: two surrogates</td>
<td>245</td>
</tr>
<tr>
<td>12.26</td>
<td>Longleaf data: two more surrogates</td>
<td>246</td>
</tr>
<tr>
<td>12.27</td>
<td>Longleaf data: difference between the <code>markstat</code> function for the data and the mean function for the surrogates</td>
<td>248</td>
</tr>
<tr>
<td>12.28</td>
<td>Longleaf data: DBH differentiation (2) index for the data and location</td>
<td>248</td>
</tr>
<tr>
<td>12.29</td>
<td>Longleaf data: DBH differentiation (2) index for the surrogates</td>
<td>249</td>
</tr>
<tr>
<td>12.30</td>
<td>Longleaf data: Mark Correlation Functions</td>
<td>250</td>
</tr>
<tr>
<td>12.31</td>
<td>Longleaf data: Kmulti plots as given by <code>spatstat</code></td>
<td>250</td>
</tr>
<tr>
<td>12.32</td>
<td>Spruce data: autocorrelations for the differences in co-ordinates and marks</td>
<td>252</td>
</tr>
<tr>
<td>12.33</td>
<td>Spruce data with resampled location: Mark Correlation Functions</td>
<td>252</td>
</tr>
<tr>
<td>12.34</td>
<td>Spruce data with resampled location: Kmulti plots as given by <code>spatstat</code></td>
<td>253</td>
</tr>
<tr>
<td>12.35</td>
<td>Longleaf data: autocorrelations for the differences in co-ordinates and marks</td>
<td>253</td>
</tr>
<tr>
<td>12.36</td>
<td>Longleaf data with resampled location: Mark Correlation Function</td>
<td>254</td>
</tr>
<tr>
<td>12.37</td>
<td>Longleaf data with resampled location: Kmulti plots as given by <code>spatstat</code></td>
<td>254</td>
</tr>
<tr>
<td>13.1</td>
<td>Non-homogeneous Poisson data: $\lambda(x) = 200x + 200y$</td>
<td>259</td>
</tr>
<tr>
<td>13.2</td>
<td>Data with 25 randomly placed tiles and a surrogate</td>
<td>260</td>
</tr>
<tr>
<td>13.3</td>
<td>Distribution of quadrat counts for surrogates using block resampling (1)</td>
<td>261</td>
</tr>
<tr>
<td>13.4</td>
<td>Distribution of quadrat counts for surrogates using block resampling (2)</td>
<td>261</td>
</tr>
<tr>
<td>13.5</td>
<td>Block resampling: estimates of $K(r)$ and the Pair Correlation Function</td>
<td>262</td>
</tr>
<tr>
<td>13.6</td>
<td>Wavelet resampling: estimates of $K(r)$ and the PCF using 25 tiles</td>
<td>262</td>
</tr>
<tr>
<td>13.7</td>
<td>Matern Cluster data</td>
<td>263</td>
</tr>
<tr>
<td>13.8</td>
<td>Matern Cluster pattern: data with 25 randomly placed tiles and a surrogate</td>
<td>264</td>
</tr>
<tr>
<td>13.9</td>
<td>Matern Cluster pattern: estimates of $K(r)$ and the PCF using 25 tiles</td>
<td>264</td>
</tr>
<tr>
<td>13.10</td>
<td>Matern Cluster pattern: data with 100 randomly placed tiles and a surrogate</td>
<td>265</td>
</tr>
<tr>
<td>Figure</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>13.11</td>
<td>Matern Cluster pattern: estimates of $K(r)$ and the PCF using 100 tiles</td>
<td>266</td>
</tr>
<tr>
<td>13.12</td>
<td>Matern Cluster pattern: estimates of $K(r)$ and the Pair Correlation Function using two-stage resampling</td>
<td>266</td>
</tr>
<tr>
<td>13.13</td>
<td>Non-homogeneous Poisson pattern: estimates of $K(r)$ using the marked point method where the marks are the number of points with radius $r$ from each point</td>
<td>268</td>
</tr>
<tr>
<td>13.14</td>
<td>Non-homogeneous Poisson pattern: estimates of $K(r)$ using the marked point method where the marks are the isotropic correction weights</td>
<td>269</td>
</tr>
<tr>
<td>13.15</td>
<td>Non-homogeneous Poisson pattern: estimates of $K(r)$ using the marked point method where marks are the number of points within approximate distance $r$</td>
<td>270</td>
</tr>
<tr>
<td>13.16</td>
<td>Matern Cluster pattern: estimates of $K(r)$ using the marked point method where the marks are the number of points with radius $r$ from each point</td>
<td>270</td>
</tr>
<tr>
<td>13.17</td>
<td>Matern Cluster pattern: estimates of $K(r)$ using the marked point method where the marks are the isotropic correction weights</td>
<td>271</td>
</tr>
<tr>
<td>13.18</td>
<td>Non-homogeneous Poisson pattern: comparison of marks (number of points within the indicated radius) for points in the data and wavelet lifted surrogates (1)</td>
<td>272</td>
</tr>
<tr>
<td>13.19</td>
<td>Non-homogeneous Poisson pattern: comparison of marks (number of points within the indicated radius) for points in the data and wavelet lifted surrogates (2)</td>
<td>273</td>
</tr>
<tr>
<td>13.20</td>
<td>Non-homogeneous Poisson pattern: comparison of marks (weights used in the isotropic correction) for points in the data and wavelet lifted surrogates (1)</td>
<td>274</td>
</tr>
<tr>
<td>13.21</td>
<td>Non-homogeneous Poisson pattern: comparison of marks (weights used in the isotropic correction) for points in the data and wavelet lifted surrogates (2)</td>
<td>275</td>
</tr>
<tr>
<td>13.22</td>
<td>Matern Cluster pattern: comparison of marks (number of points within the indicated radius) for points in the data and wavelet lifted surrogates (1)</td>
<td>276</td>
</tr>
<tr>
<td>13.23</td>
<td>Matern Cluster pattern: comparison of marks (number of points within the indicated radius) for points in the data and wavelet lifted surrogates (2)</td>
<td>277</td>
</tr>
<tr>
<td>13.24</td>
<td>Matern Cluster pattern: comparison of marks (weights used in the isotropic correction) for points in the data and wavelet lifted surrogates (1)</td>
<td>278</td>
</tr>
<tr>
<td>13.25</td>
<td>Matern Cluster pattern: comparison of marks (weights used in the isotropic correction) for points in the data and wavelet lifted surrogates (2)</td>
<td>279</td>
</tr>
</tbody>
</table>
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Manaus river data: comparison of Portmanteau test statistics for coefficients in crystal (2, 0) using Lifting</td>
<td>35</td>
</tr>
<tr>
<td>3.2</td>
<td>Manaus river data: basis crystals selected after wavelet decomposition and three different white noise tests</td>
<td>36</td>
</tr>
<tr>
<td>3.3</td>
<td>Manaus river data: parameter estimates using an AR(3) model</td>
<td>37</td>
</tr>
<tr>
<td>4.1</td>
<td>Theoretical autocovariance and autocorrelation functions for Level 1 smooth coefficients after lifting an AR(1) process</td>
<td>44</td>
</tr>
<tr>
<td>4.2</td>
<td>Theoretical autocorrelations for Level 1 detail coefficients after lifting AR(1)</td>
<td>45</td>
</tr>
<tr>
<td>4.3</td>
<td>Theoretical autocorrelations for Level 1 smooth coefficients after lifting AR(1)</td>
<td>45</td>
</tr>
<tr>
<td>4.4</td>
<td>Theoretical autocovariance functions for Level 1 detail coefficients after lifting an MA(1) process</td>
<td>46</td>
</tr>
<tr>
<td>4.5</td>
<td>Theoretical autocovariance functions for Level 1 smooth coefficients after lifting an MA(1) process</td>
<td>47</td>
</tr>
<tr>
<td>4.6</td>
<td>Estimates of parameters and variance for AR(1) simulations with $\phi &gt; 0$</td>
<td>48</td>
</tr>
<tr>
<td>4.7</td>
<td>Estimates of parameters and variance for AR(1) simulations with $\phi &lt; 0$</td>
<td>49</td>
</tr>
<tr>
<td>4.8</td>
<td>Estimates of parameters and variance for MA(1) simulations with $\theta &gt; 0$</td>
<td>50</td>
</tr>
<tr>
<td>4.9</td>
<td>Estimates of parameters and variance for MA(1) simulations with $\theta &lt; 0$</td>
<td>50</td>
</tr>
<tr>
<td>4.10</td>
<td>Basis for AR(1) simulations with $\phi &gt; 0$</td>
<td>51</td>
</tr>
<tr>
<td>4.11</td>
<td>Basis for AR(1) simulations with $\phi &lt; 0$</td>
<td>52</td>
</tr>
<tr>
<td>4.12</td>
<td>Basis for MA(1) simulations with $\theta &gt; 0$</td>
<td>52</td>
</tr>
<tr>
<td>4.13</td>
<td>Basis for MA(1) simulations with $\theta &lt; 0$</td>
<td>52</td>
</tr>
<tr>
<td>4.14</td>
<td>Simulations with parameter $-0.9$ and $n = 2048$ compared with $n = 256$</td>
<td>52</td>
</tr>
<tr>
<td>4.15</td>
<td>White noise test statistics: AR(1) with $\phi = -0.5$</td>
<td>66</td>
</tr>
<tr>
<td>4.16</td>
<td>White noise test statistics: MA(1) with $\theta = -0.5$</td>
<td>68</td>
</tr>
<tr>
<td>4.17</td>
<td>Basis for AR(1) simulations with $\phi &lt; 0$ using the Box-Pierce test statistic</td>
<td>70</td>
</tr>
<tr>
<td>4.18</td>
<td>Basis for MA(1) simulations with $\theta &lt; 0$ using the Box-Pierce test statistic</td>
<td>70</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>4.19</td>
<td>AR(1) simulations with $\phi &lt; 0$: estimates of parameters and variance using the Box-Pierce test statistic.</td>
<td></td>
</tr>
<tr>
<td>4.20</td>
<td>MA(1) simulations with $\theta &lt; 0$: estimates of parameters and variance using the Box-Pierce test statistic.</td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>Homogeneous process: time of occurrence of data events.</td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>Homogeneous process: number of events by time period for data and surrogates.</td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>Non-homogeneous process: time of occurrence of data events.</td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>Non-homogeneous process: number of events by time period for data and surrogates.</td>
<td></td>
</tr>
<tr>
<td>5.5</td>
<td>Clustered process: time of occurrence of data events.</td>
<td></td>
</tr>
<tr>
<td>5.6</td>
<td>Clustered process: number of events by time period for data and surrogates.</td>
<td></td>
</tr>
<tr>
<td>5.7</td>
<td>Mean $\chi^2$ test statistic values for surrogates of simulated homogeneous patterns with random intensities.</td>
<td></td>
</tr>
<tr>
<td>5.8</td>
<td>Details of runs for the homogeneous patterns where at least one of the surrogates had a test statistic larger than the critical value.</td>
<td></td>
</tr>
<tr>
<td>5.9</td>
<td>Mean $\chi^2$ test statistic values for surrogates of simulated non-homogeneous patterns with random intensity functions.</td>
<td></td>
</tr>
<tr>
<td>5.10</td>
<td>Details of runs for the non-homogeneous patterns where at least one of the surrogates had a test statistic smaller than the critical value.</td>
<td></td>
</tr>
<tr>
<td>5.11</td>
<td>Mean $\chi^2$ test statistic values for surrogates of simulated clustered series with random intensities.</td>
<td></td>
</tr>
<tr>
<td>5.12</td>
<td>Details of runs for the clustered series where at least one of the surrogates had a test statistic smaller than the critical value.</td>
<td></td>
</tr>
<tr>
<td>6.1</td>
<td>Sample data and calculated boundary point coordinates.</td>
<td></td>
</tr>
<tr>
<td>6.2</td>
<td>Updated point coordinates.</td>
<td></td>
</tr>
<tr>
<td>6.3</td>
<td>Differences between deleted points and mean position of nearest neighbours, the mean position of neighbours to a deleted point and the uplift values.</td>
<td></td>
</tr>
<tr>
<td>6.4</td>
<td>Point coordinates after one stage and two stages of resampling and recomposition.</td>
<td></td>
</tr>
<tr>
<td>6.5</td>
<td>$p$-values for Kolmogorov-Smirnov tests on surrogate Poisson(50) point patterns.</td>
<td></td>
</tr>
<tr>
<td>6.6</td>
<td>$p$-values for Kolmogorov-Smirnov tests on surrogate Thomas point patterns.</td>
<td></td>
</tr>
<tr>
<td>6.7</td>
<td>Parameter estimates for surrogate Thomas point patterns.</td>
<td></td>
</tr>
<tr>
<td>6.8</td>
<td>$p$-values for Kolmogorov-Smirnov tests on surrogate Strauss point patterns.</td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>Quadrat counts for data and mean quadrat counts for surrogates with intensity $\lambda(x) = 50 + 100x$.</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>Summary statistics for estimates of coefficients from surrogates modelling intensity $\lambda(x) = 50 + 100x$.</td>
<td></td>
</tr>
</tbody>
</table>
LIST OF TABLES

7.3 Quadrat counts for data and mean quadrat counts for 1000 surrogates with intensity
\[\lambda(x, y) = 200x + 200y\] .................................................. 146

7.4 Summary statistics for estimates of coefficients from surrogates modelling intensity
\[\lambda(x, y) = 200x + 200y\] .................................................. 146

8.1 Simulation parameters and their estimators from data ............................ 163

8.2 Simulation parameters and their estimators from surrogates .................. 163

9.1 Quadrat counts for data and mean quadrat counts for surrogates ............. 174

9.2 Summary statistics for estimates of coefficients of model fitted to surrogates ... 177

12.1 Summary statistics for Boese data marks ........................................ 228

12.2 Nearest neighbour indices ....................................................... 231

12.3 Summary statistics for the \textit{markstat} function ............................. 231

12.4 Boese data: points selected for calculation of DBH differentiation (2) .... 231

12.5 Summary statistics for Spruce data marks ....................................... 237

12.6 Spruce data: Nearest neighbour indices ........................................ 238

12.7 Spruce data: Summary statistics for the \textit{markstat} function ............... 238

12.8 Spruce data: points selected for calculation of DBH differentiation (2) .... 238

12.9 Summary statistics for Longleaf data marks ................................... 244

12.10 Longleaf data: Nearest neighbour indices ...................................... 246

12.11 Longleaf data: Summary statistics for the \textit{markstat} function ............ 246

12.12 Longleaf data points selected for calculation of DBH differentiation (2) . 247
Chapter 1

Introduction

1.1 Aims of the thesis

The motivation for the thesis is to ascertain whether the wavelet transform, or variations thereof, can be used as a non-parametric method for resampling some types of dependent data, thus generating surrogates that maintain the same signal as the data but with different noise components. The types of dependent data considered are time series and point patterns.

The thesis contains three areas of original work, of which the first two are minor:

1. using Wavelet Lifting to resample time series and analysing the effectiveness of the method for different parameter values

2. applying the method in (1) to 1-dimensional point patterns

3. developing a method based on the Wavelet Lifting philosophy to resample 2-dimensional spatial point patterns.

In each case the surrogates are compared for similarity to the data, using a variety of methods. By ‘similarity’ is meant that the surrogates share characteristics with the data and could have come from the same model that produced the data.

Time series surrogates are compared by the distribution of parameter values obtained from fitting standard models. Point patterns are compared either by calculation of various indices or graphically by plotting standard functions. The majority of the graphs show 95% envelopes that are derived from the surrogates.
1.2 Chapter summary

Chapter 2 outlines published methods of bootstrapping time series and 1- and 2-dimensional point patterns.

Chapter 3 provides background material on wavelet transforms and illustrates a method for bootstrapping the Discrete Packet Wavelet Transform. Also described are the simpler second generation wavelets, known as the Lifting Scheme, as developed by Sweldens [35] and Sweldens and Schröder [36].

In Chapter 4 I then use the Lifting Scheme to transform AR(1) and MA(1) time series. The decorrelation properties of Lifting are examined and the surrogates resulting from resampling the transformed data and inverting are analysed. Having found that the method for resampling time series is effective in some situations I then apply it, in Chapter 5, to some types of 1-dimensional point patterns, specifically homogeneous, non-homogeneous and clustered.

In Chapter 6 I then develop a method for resampling 2-dimensional spatial point patterns using an adaption of the wavelet Lifting philosophy and Delaunay Triangulation. The method is fully described before being applied to some examples. Various previously published indices and summary functions are used to compare the surrogates with the data.

The method is then applied in Chapter 7 to examples of non-homogeneous 2-D spatial point patterns while clustered patterns are considered in Chapter 8. It is shown that the method does not work with such patterns and I then use a 2-stage resampling method which produces better results. This method is then applied in Chapter 9 to previously examined non-homogeneous patterns.

Qualitatively marked spatial point patterns are then considered in Chapter 10 with three variations of the method developed in Chapter 6 to account for the marks. I then apply two of these methods in Chapter 11 to well-known data.

I then give a variation of the method in Chapter 12 for quantitatively marked spatial point patterns and illustrate its use with three forestry examples.

Finally, in Chapter 13 I compare the method with two published methods for bootstrapping spatial point patterns.
Chapter 2

Literature Review

2.1 Wavelets

There are many books available that develop wavelet theory and illustrate applications. Both Addison [1] and Vidakovic [40] provide good general introductions. Percival and Walden [26] apply wavelets specifically to Time Series and hence are more useful to this research.

Two introductions to the Lifting Scheme, otherwise known as Second Generation Wavelets, are provided by Sweldens [35] and Sweldens and Schröder [36].

Fernandez et al [12] describe the Lifting Scheme and explain the software package written in C for fast calculation of 2D biorthogonal wavelet transforms using the Lifting Scheme.

2.2 Bootstrapping Time Series

Bootstrapping involves sampling observations with replacement to produce a surrogate set of observations. The process is repeated a large number of times and a chosen statistic is then calculated for all the surrogates, thus creating an estimate of the distribution of the statistic.

Any correlation present in the observations will be destroyed in the sampling process and hence modifications are required for dependent data. One such modification is the block bootstrap where the observations are grouped into blocks of consecutive observations and the blocks are subsequently sampled.

The paper by Bühlmann [7] is a useful summary of various methods for resampling time series, principally block and sieve methods.

The block method is the most general and is simple to implement. However it does not produce stationary series and exhibits artifacts where blocks join. These artifacts therefore limit the usefulness of double bootstrapping. The method is very sensitive to the length of the block and the author lists three things that influence optimal block length: the data-generating process, the
2.2. BOOTSTRAPPING TIME SERIES

Sieve bootstraps resample from a reasonable time series model and seem to be less sensitive to the selection of a model than the block bootstrap to the block length. The author specifies two types of sieve bootstraps: the AR-sieve used with linear time series and the VLMC-sieve (Variable Length Markov Chain) used for categorical processes. Both are expected to adapt to the degree of dependence, with their accuracy improving as the degree of dependence decreases.

Vogel and Shallcross [43] apply the moving-blocks bootstrap to streamflows. The authors concluded that the method `can provide a very simple and attractive alternative to parametric time series models.’ They state that there is uncertainty in selecting relevant model parameters because of the short length of the data in addition to the uncertainty of the correct model to apply. In contrast, the only uncertainty in the moving-blocks method is the selection of the block length.

However the authors also recognize the limitations of the method. The bootstrap `will never generate an observation either larger or smaller than the maximum or minimum historical observation.’ In particular, considering the streamflow data studied, the bootstrap would never produce `a worse drought than that experienced historically.’

The overview by MacKinnon [21] of general bootstrap methods states that the block length in the block bootstrap is critical. It must increase as the number of observations, \( n \), increases and is often proportional to \( \sqrt{n} \). If the block length is too short then correlation is destroyed and if too long then the bootstrap samples are not random enough.

Politis and Romano [28] introduce the stationary bootstrap, a resampling method for stationary, weakly dependent time series that is similar to block resampling. In contrast to the latter, the stationary bootstrap generates pseudo time series that are stationary. It does this by resampling blocks of random size with the length of each block having a geometric distribution with parameter \( p \). Other block resampling procedures use blocks of fixed length \( b \).

It has been previously stated that the critical decision with block resampling is choosing the block length; with the stationary bootstrap the critical decision is the choice of \( p \).

No examples of the procedure’s application are given in the paper.

Politis [27] summarises the methods that have been used to bootstrap time series including the various versions of the block bootstrap.

The author finds that, in the case of dependent stationary data, under some regularity conditions, the appropriately standardized/Studentized block bootstrap (in all its variations, including the stationary and circular bootstrap) also possesses a `higher-order accuracy´ property when estimating the distribution of the sample mean.

He concluded that practically all time series problems fall under the `difficult´ category, and thus the block resampling/subsampling methods become invaluable.

The matched-block bootstrap was introduced by Carlstein et al [8] who showed that, compared to the simple block bootstrap, it produces standard error estimators that have similar variability.
and less bias. The method adopts a block joining rule which is more favourable to blocks that are similar. For example, if the series has positive autocorrelation, the end of one block with high observations is likely to be associated with a following block that has high readings at the beginning. Blocks are prepared as before, but are placed into bootstrap samples using a Markov chain, using transition probabilities which favor close matching.

Hesterberg [14] extended the method by examining other block joining rules, considering five procedures:

1. block bootstrapping without matching
2. matched-block based on the last observation in the block
3. matched-block based on predictions for the next observation following each block
4. matched-block based on predictions for the average of the next 100 observations following each block, with predictions based on observations in a block
5. matched-block based on predictions for the average of the next 100 observations following each block, with predictions based on observations preceding and within a block

The author concluded ‘It appears from these results that matched-block bootstrapping is feasible for obtaining standard errors for the sample mean, for series with short-range dependence and moderate long-range dependence, but that there is still significant bias.’

Davison and Hinkley [10] examined various methods for resampling dependent data. Considering time series, they specify model-based resampling, where residuals from a fitted model are resampled, block resampling and periodogram resampling.

As previously stated, the major limitation of block resampling is the generation of series that are less dependent than the original data. The authors mention three methods for overcoming this limitation:

1. the series is ‘pre-whitened’ by fitting a model and then a series of innovations is generated by block resampling of residuals from the fitted model. The innovation series is then ‘post-blackened’ by applying the estimated model to the resampled innovations.
2. blocks of blocks
3. stationary bootstrap

Application of the various methods to one real data set indicate that model-based, blocks of blocks and post-blackening procedures yield the best results while simple blocks and the stationary bootstrap are less successful.

Breakspear [5] examined whether the decorrelation property of the wavelet transform can be utilized to resample dependent data. A multiresolution decomposition produces detail coefficients
at various levels. These coefficients were then resampled and a surrogate signal was subsequently reconstructed using the inverse transform. Three types of resampling were used:

1. free permutation of wavelet coefficients within each scale
2. cyclic rotation of coefficients within each scale
3. block resampling of wavelet coefficients within each scale.

Application of these methods to three different time series showed that the best method for preservation of characteristics of the data in the surrogate series was block resampling of wavelet coefficients within each scale.

Percival et al [25] demonstrate that by applying the bootstrap in the wavelet domain (the wavelet decomposition) the distribution of the sample autocorrelation at lag 1 can be approximated for long memory processes. However the method does not work for short memory processes which they attribute to the inability for the Discrete Wavelet Transform (DWT) to adequately decorrelate such processes. They propose a modification of the DWT, called the Discrete Wavelet Packet Transform (DWPT), to provide a suitable decorrelation procedure. The adequacy of the method is demonstrated with both short and long memory processes.

Note that the authors use the term ‘wavestrapping’ to refer to bootstrapping with the DWPT only. DWT-based bootstrapping is a self-explanatory term used otherwise.

Tang et al [38] acknowledge that several papers have shown that wavelet-based resampling (which they term ‘wavestrapping’) approximately preserves the correlation within the data. They state that preservation of the correlation structure ‘does not guarantee enough variation for regression parameter estimators’. Their simulations show ‘that these wavestraps yield undercoverage of parameters for a simple linear regression for time series data of the type that arise in functional MRI experiments’ and they conclude that ‘the wavestrap method is not completely valid in obtaining resamples related to linear regression analysis and should be used with caution for hypothesis testing’.

The authors show that the deficiency with wavestrapping arises from too few coefficients being available to resample at higher levels of the decomposition, producing surrogate series with excessive repetition of the structure. In response they propose the ‘parametric wavestrap’. They observe that the Daubechies wavelet bases used in the decomposition are orthonormal and hence theoretically the detail coefficients at all levels and the scaling coefficients ‘follow independent normal distributions if the original data are standard normal white noise’. They propose to replace resampling the observed detail coefficients with generating such coefficients from a N(0,1) distribution.

The paper then went on to demonstrate that for a simple linear regression model typically used in fMRI experiments, ‘the wavestrap produces undercoverage of the confidence intervals of the
The authors acknowledge however that the parametric bootstrap may not apply when the distribution of the wavelet coefficients is unknown. Therefore the method has limited application.

Yi et al [45] use a two-step wavestrapping method to simulate non-stationary time series, specifically acceleration data from mobile computing users. The method consisted of `parallel wavestrapping´ and energy/trend adjustments.

`Parallel wavestrapping´ refers to a simultaneous stationary bootstrap among different scale levels with the aim of preserving both within- and between-level correlations. The energy/trend adjustments attempt to produce surrogate time series with the same trend and scaling properties as the original data.

Instead of a full DWT, only the first step is carried out to produce equal length detail and scaling coefficient vectors. Both are resampled in parallel and the inverse transform yields a surrogate series. A full DWT is applied to the surrogate and the final scaling coefficients are replaced by the corresponding coefficients from a full DWT applied to the original data.

The detail coefficients then have an energy adjustment applied before the inverse transform yields the final surrogate series.

The authors found that the resulting surrogates showed closer resemblance to the data than surrogates produced from bootstrapping the DWT.

2.3 Bootstrapping 1D Point Patterns

Sarma et al [33] `summarize bootstrap algorithms that are applicable to point process models of neural spike train data and focus on two algorithms based on the time-rescaling theorem for point process models that make no approximations or assumptions on the data.´

Such point process models are characterized by the spike rate function, or conditional intensity function, which gives the probability that a neuron will fire at time $t$ given the neuron’s firing history and relevant external stimuli.

The Time-Rescaling Theorem states `that any point process with an integrable conditional intensity function may be transformed into a Poisson process with unit rate.’ (Brown et al [6]).

The authors used two bootstrap methods, one parametric and the other semi-parametric, that are applicable to any point process model and simulated each bootstrap spike train sample using time-rescaling. Each method applied the time-rescaling theorem to represent the original spike-time samples as a set of uniform i.i.d. samples, and then implemented one of two parametric bootstrap techniques to the transformed data. The generated bootstrap samples were transformed back into spike arrival time samples, and then bootstrap replications of the estimate of conditional intensity function and/or statistics were computed.
2.4. Bootstrapping 2D Spatial Point Patterns

The semi-parametric results differ from the parametric results because the semi-parametric bootstrap algorithm first samples from the experimental CDF, \( \hat{F}(u) \), whereas the parametric bootstrap method first samples from a true uniform CDF \( F(u) \).

The authors showed that when analytic approximations to computing uncertainties in point process model statistics were available and accurate, the parametric bootstrap method gives comparable results.

Cowling et al [9] develop bootstrap methods for constructing confidence regions when estimating the intensity function of a non-homogeneous Poisson process.

A kernel estimator of the varying intensity, \( \hat{\lambda}(x) \), is obtained from the data. Then let \( \hat{\Lambda}(x) = \int_0^x \hat{\lambda}(y)dy \). Given a sample of size \( N \), the three methods of bootstrapping adopted by the authors were:

1. resample \( N^* \) times as points from a Poisson process with intensity \( \hat{\lambda} \), where \( N^* \) has a Poisson distribution with intensity \( \hat{\Lambda}(1) \)

2. resample \( N^* \) times randomly with replacement from the data, where \( N^* \) is distributed as in (1)

3. resample \( N \) times randomly with replacement from the data

The authors recommended the last method as the best.

2.4 Bootstrapping 2D Spatial Point Patterns

Scalon and Silva [34] investigated the performance of the bootstrap method for estimating the sampling distribution of intensity when sampling used quadrat counts. A simulated point pattern was subdivided into 16 quadrats of which 8 were randomly chosen 250 times to provide estimates of intensity. Both CSR (Complete Spatial Randomness) and clustered processes were simulated.

The conclusion was that the bootstrap method ‘was dubious to estimate sampling distribution for spatial intensity when the sampling is based on a sample of quadrats.’

Tang et al [37] proposed a new parametric spatial bootstrap. The new method combines spatial modelling and the parametric bootstrap to produce valid resamples of spatially correlated normal data, specifically brain images.

The method is applicable to geostatistical data and shown to be produce better results than block resampling. It is not suitable for point processes as it fits a model to the assumed random field.

The paper by Lahiri and Zhu [18] is also applicable to geostatistical data. The authors investigate bootstrapping a class of spatial regression models when the sampled sites are irregularly
spaced and generated by a stochastic design. They show that existing block bootstrap methods for
grid spatial data do not work for irregularly spaced data when the stochastic design is nonuniform.

Two versions of the block bootstrap are introduced. The data-site-shifted block bootstrap
(DSSBB) method is an extension of the standard spatial block bootstrap for regularly spaced grid
data to irregularly spaced data. In this method the lower left corner of the blocks are placed at
sampling sites. However the method is not suitable when the spatial sampling density is nonuni-
form.

The second method overcomes the limitation of the DSSBB by introducing a regular grid over
the sampling sites and placing the lower left corner of the blocks at a grid position. This method
is called the grid-based block bootstrap method or the GBBB method.

spatial statistics, they state ‘most research in this area discusses extensions of the block resampling
idea.’ The authors note the limitation of pasting independently resampled subrectangles together,
creating edge effects, and that ‘no natural analogy with the matched-block or sieve approaches
has yet been identified.’

Most of the work to that date assumed stationarity in the point locations and, if present, re-
sponses.

Nordman and Lahiri [22] are concerned with the block bootstrap variance estimator applied
to spatial data on a regular grid. They develop precise formulae for the optimal block sizes that
minimize the mean squared error of the bootstrap variance estimator and describe practical meth-
ods for estimating these spatial block sizes.

Most of the material presented by Tscheschel and Stoyan [39] is repeated in Illian et al [15].

Their approach uses summary characteristics of a stationary empirical pattern to generate a
simulated pattern with very similar summary characteristics. The algorithm uses numerical (eg
intensity) and functional summary characteristics (eg $K$-function).

The first step of the algorithm is to generate an arbitrary pattern having no relationship to
the pattern to be simulated except for an equivalent number of points. In each subsequent step
of the simulation one point is randomly chosen from the current pattern and replaced by a newly
generated point, values of the summary characteristics then being calculated for the new pattern.
If these values are closer to the empirical values than those from the current pattern then the newly
generated point replaces the chosen point. If not, the original point is retained and the process
repeated.

The authors are aware of the method’s limitations - different point patterns can have the same
intensity and summary functions. They give an example of two Neyman-Scott processes with the
same intensity and $L$-functions (in one pattern all clusters consist of 4 points, while in the other
there are usually one-point clusters and, with small probability, 10-point clusters) where the re-
construction algorithm generates a pattern somewhere between.
Illian et al [15] (p408) state ‘It is very important to carefully choose the summary characteristics such that the variability among the generated patterns is as small as possible.’

Pommerening and Stoyan [29] use data from three large, fully enumerated and mapped woodlands that were resampled using circular sample plots as subwindows and the original forest structures were then recreated using data from the sample trees only. Tree location, diameter at breast height and species was available, enabling various Nearest Neighbour Summary Statistics (NNSS) to be calculated.

Using information from the subwindows and the summary statistics, point patterns in the whole window were generated numerically with the aim that these are similar to the true point pattern. The position of the points in the subwindows are fixed during the whole reconstruction process while outside the subwindows points are located in such a way that the whole pattern has NNSS as close as possible to those estimated for the circular subwindows.

The initial point configuration for the reconstruction consists of the circular subwindows plus points placed completely randomly outside the subwindows such that the total number of points equals the point density estimated from the subwindows times the window area.

A contrast measure is used to quantify the difference between the NNSS calculated from the subwindows and the reconstructed point pattern in the whole window.

The authors concluded that, for the three types of woodland considered, the method was successful in yielding reconstructed point patterns that had similar second-order characteristics to the data.

Loh and Stein [19] introduce the marked point bootstrap method which is a variant of the block of blocks method to find confidence intervals for the estimator of the reduced second moment function, \( K(r) \). Briefly, to each point \( x \) in the observed window \( A \) is assigned a mark that is the sum of all the weights for points \( y \) in \( A \) within \( r \) of \( x \). The weight equals the reciprocal of the fraction of the area of the shell of radius \( |y - x| \) centered at \( x \) that is contained in \( A \).

The points together with the marks are resampled.

The contribution of a point at \( y \) to the mark at \( x \) includes information about the boundary of the observation window, through the weights used to account for edge effects.

The authors applied the method to three simulated point processes (Poisson, cluster field and soft core) and compared the results with three other methods - splitting, tiling (blocks), and subsets. The splitting method divides \( A \) into \( N \) congruent subregions and calculates \( N \) separate estimates of \( K(r) \). The subsets method also uses \( N \) square subregions but obtains only one estimate of \( K(r) \) by first summing the contribution of each pair of points within each subregion and then summing over subregions.

They found that

(1) the tiling method was the weakest and most computationally intensive method
(2) the marked point method appeared to be the best method for Poisson and soft core processes
(3) no method was more advantageous for the Matern cluster field. They state ‘there may not be any procedure that really works well’

Finally, the authors concluded ‘More studies should be done to investigate if the superiority of the marked point method indeed holds for a wider range of processes.’

Loh [20] analysed the marked point bootstrap introduced by Loh and Stein [19] as an alternative to the block bootstrap. In this method it is not the position of the points that is resampled but the marks associated with the points. The author claims three advantages:

(1) compared with the block bootstrap, the marked point bootstrap method produces confidence intervals with better empirical coverage
(2) as the marks are used to compute the bootstrap estimate the resampled points do not have to be arranged to form a new point pattern
(3) once the marks are calculated the only extra computation required is the selection of the points and keeping track of the number of times each point is resampled

Concerning the marked point bootstrap, the author concluded: ‘Our study here suggests that non-parametric bootstrap can yield valid estimates of errors under a wide range of point patterns. The simulation study performed here shows that bootstrap confidence intervals do attain coverage close to the nominal level, even for the clustered point patterns where Poisson errors are known to be inaccurate, when sample sizes are large. More specifically, bootstrap performance improves with increasing number density, and also with increasing observation region size relative to the correlation length.’

Jansen and Oonincx [16] describe the Lifting Scheme in detail, including lifting in two dimensions. In this case the authors prescribe Delaunay Triangulations for a multiscale analysis, removing one point at a time. Each point in the triangulation has a Voronoi cell and removal of a point means redistributing that cell’s area amongst the neighbouring cells. In the prediction step the coefficients are specified as the proportions of the Voronoi cell of the removed point that is taken by the neighbours.

Wu and Amaratunga [44] introduce the Wavelet Triangulated Irregular Network (WTIN) which applies second-generation wavelet theory to geographical height field data with the aim of producing a terrain surface. Given the location and height at a number of observed points, the method uses quaternary triangulation to introduce points at a finer scale. Repetition of the process ultimately yields the required surface.

The paper is useful because of its use of second generation wavelets and triangulation.

Pradhan et al [30] use second-generation wavelets to compress spatial data.

An interpolating wavelet filter first splits a triangle into several sub-triangles and an elevation
step subsequently modifies the point values. The data is then compressed at the desired locations by using second-generation wavelets.

The method utilizes Delaunay triangulation and Voronoi tessellations.
3.1 What are wavelets?

Wavelets are `small´ waveforms that are localized in time or space, contrasting with `large´ waveforms like the sine curve that oscillates with the same amplitude indefinitely.

For example, the first derivative of $e^{-\frac{t^2}{2}}$ (Gaussian curve) is called the Gaussian wave and is shown in Figure 3.1. It is localized as it decays rapidly towards zero, although it is equal to zero only at $t = 0$.

![Figure 3.1: The Gaussian wavelet $f(t) = -te^{-\frac{t^2}{2}}$](image)

If we restrict ourselves to real valued functions, then a function $\psi(t)$ can be a wavelet if it satisfies the following conditions:

1. $\int_{-\infty}^{\infty} \psi(t)dt = 0$
2. $\int_{-\infty}^{\infty} \psi^2(t)dt = 1$
3. If $\hat{\psi}(f)$ is the Fourier transform of $\psi(t)$ then

   $$C_\psi \equiv \int_0^{\infty} \frac{|\hat{\psi}(f)|}{f} df < \infty$$

This is called the admissibility condition, with $C_\psi$ being the admissibility constant.
3.2. THE WAVELET TRANSFORM

The Gaussian wave satisfies the first and third conditions but not the second; it is normalized by multiplying by $\sqrt{\frac{2}{\pi}}$.

Two other examples of simple wavelets are:

- the Haar wavelet (Figure 3.2(a)) which is defined in a number of ways, one definition being

  \[
  \psi^{(H)}(t) = \begin{cases} 
  1 & 0 \leq t < \frac{1}{2} \\
  -1 & \frac{1}{2} \leq t < 1 \\
  0 & x < 0 \text{ or } x > 1
  \end{cases}
  \]

  (3.1)

- the normalized Mexican hat wavelet (Figure 3.2(b)),

  \[
  \psi^{(Mh)}(t) = \frac{2}{\sqrt{3\sqrt{\pi}}} (1 - t^2)e^{-\frac{t^2}{2}}
  \]

  (3.2)

| Figure 3.2: The (a) Haar and (b) normalized Mexican hat wavelets |

3.2 The Wavelet Transform

The form $\psi(t)$ chosen for the wavelet is often called the mother wavelet from which we can derive an infinite number of daughter wavelets by translation and dilation. We can translate the original wavelet $b$ units to the right by using $\psi(t - b)$ and dilate by using $\psi(\frac{t-b}{a})$ where $a$ is known as the scale. We use the notation $\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a})$ where $\frac{1}{\sqrt{a}}$ is a weighting used to ensure that wavelets at every scale have the same energy.

The continuous wavelet transform (CWT) of a continuous signal $x(t)$ is given by

\[
W(a,b) = \int_{-\infty}^{\infty} x(t)\psi_{a,b}^*(t)dt \quad a > 0; \quad -\infty < b < \infty
\]

(3.3)
where \(^*\) indicates the complex conjugate of the wavelet.

The original signal can be recovered from the CWT using the inverse wavelet transform

\[
x(t) = \frac{1}{C_{\psi}} a^2 \int_{-\infty}^{\infty} \int_{0}^{\infty} W(a, b) \psi_{a,b} \, db \, da \tag{3.4}
\]

### 3.2.1 The Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) can be considered as a subsampling of the CWT or it can be developed from first principles as a stand-alone entity.

In simple terms the DWT splits the signal into a detail series and an approximation series each half the length of the signal. The approximation series is further split into another detail series and approximation series each one quarter the length of the signal. The process keeps repeating, the detail being extracted at each scale from the approximation at the previous scale, until the final detail and approximation series each have one component.

We have from above \(\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right)\) where \(a\) and \(b\) are the dilation and translation parameters respectively. If we restrict \(a\) and \(b\) to discrete values and use a logarithmic discretization of the \(a\) scale linked to the size of the steps between the \(b\) locations then the wavelet has the form

\[
\psi_{j,k}(t) = \frac{1}{\sqrt{a_0^j}} \psi\left(\frac{t-kb_0a_0^j}{a_0^j}\right) \tag{3.5}
\]

Dyadic grid scaling has \(a_0 = 2\) and \(b_0 = 1\) and hence

\[
\psi_{j,k}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t-k2^j}{2^j}\right) \tag{3.6}
\]

The discrete wavelet transform of the signal \(x(t)\) is then given by

\[
W_{j,k} = \int_{-\infty}^{\infty} x(t) \psi_{j,k} dt \tag{3.7}
\]

and the signal can be recovered, using the inverse discrete wavelet transform, from

\[
x(t) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} W_{j,k} \psi_{j,k}(t) \tag{3.8}
\]

The signal detail at scale \(j\) is given by

\[
d_j(t) = \sum_{k=-\infty}^{\infty} W_{j,k} \psi_{j,k}(t) \tag{3.9}
\]

Dyadic orthonormal discrete wavelets are associated with a scaling function, \(\phi(t)\), that relates
3.2. THE WAVELET TRANSFORM

to smoothing the signal. It can also be translated and dilated, having the form

\[ \phi_{j,k}(t) = \frac{1}{\sqrt{2^j}} \phi\left( \frac{t - k2^j}{2^j} \right) \] (3.10)

where \( \phi_{0,0}(t) = \phi(t) \). Such scaling functions have the property

\[ \int_{-\infty}^{\infty} \phi(t)dt = 1 \] (3.11)

For the Haar wavelet \( \phi(t) = 1 \) for \( 0 \leq t < 1 \) and 0 elsewhere.

The approximation coefficients of the signal are given by

\[ V_{j,k} = \int_{-\infty}^{\infty} x(t) \phi_{j,k}(t)dt \] (3.12)

and a continuous smooth approximation of the signal at scale \( j \) is given by

\[ x_j(t) = \sum_{k=\infty}^{\infty} V_{j,k} \phi_{j,k}(t) \] (3.13)

and hence we have

\[ x(t) = x_J(t) + \sum_{j=-\infty}^{\infty} d_j(t) \] (3.14)

Thus the signal is decomposed into a number of detail levels and a final smooth level. This is called a multiresolution analysis. It can be seen that the role of the wavelet function is to extract the details, or small changes, from the signal while the role of the scaling function is to smooth out the signal.

Instead of using integrals to calculate \( W_{j,k} \) and \( V_{j,k} \) we can make use of the scaling and wavelet equations

\[ \phi(t) = \sum_i c_i \phi(2t - i) \] (3.15)

and

\[ \psi(t) = \sum_i (-1)^i c_{N_i-1-i} \phi(2t - i) \] (3.16)

where \( c_i \) are scaling coefficients and \( N_i \) is the number of scaling coefficients. There are two restrictions placed on the scaling coefficients:

\[ \sum_i c_i = 2 \] (3.17)

and

\[ \sum_i c_i c_{i+2i'} = 2 \text{ if } i' = 0 \text{ and } 0 \text{ otherwise} \] (3.18)
Then we can write the following recursions

\[ \phi_{j+1,k}(t) = \frac{1}{\sqrt{2}} \sum_i c_i \phi_{j,2k+i}(t) \]  

(3.19)

and

\[ \psi_{j+1,k}(t) = \frac{1}{\sqrt{2}} \sum_i b_i \phi_{j,2k+i}(t) \]  

(3.20)

where \( b_i = (-1)^i c_{N_i-1-i} \). Note that both the scaling and wavelet functions at scale \( j + 1 \) are defined in terms of the scaling function at scale \( j \).

Using these equations we can find the approximation and wavelet coefficients at scale \( j + 1 \)

\[ V_{j+1,k} = \frac{1}{\sqrt{2}} \sum_i c_{i-2k} V_{j,i} \]  

(3.21)

and

\[ W_{j+1,k} = \frac{1}{\sqrt{2}} \sum_i b_{i-2k} V_{j,i} \]  

(3.22)

The vectors containing \( \frac{1}{\sqrt{2}} c_i \) are called low pass filters as they let through the low frequency components of the signal while \( \frac{1}{\sqrt{2}} b_i \) are high pass filters as they let through the high frequency components of the signal.

To start the recursion we require \( V_{0,k} \) which is defined by

\[ V_{0,k} = \int_{-\infty}^{\infty} x(t) \phi(t-k) dt \]  

(3.23)

but in most instances we simply set \( V_{0,k} = x_k \) ie we just use the raw data. This is not theoretically correct but since we don’t know the function \( x(t) \), only the discrete sampled values \( x_k \), and \( V_{0,k} \) is a weighted average of \( x(t) \) in the vicinity of \( k \), then it is appropriate to use the data.

To complete a full decomposition we require that the length of the signal, \( N \), to be a power of 2 ie \( N = 2^J \). The detail at scale \( j \) has \( \frac{N}{2^j} \) components and the final approximation has 1 component, thus the total number of components is \( \sum_{j=1}^{N} \frac{N}{2^j} + 1 = N \). This is a restrictive requirement as most time series don’t have a length that is a power of 2 (we could leave out some data or pad out with applicable values if the length is close to a power of 2). An alternative is to use a partial DWT where the requirement is that the signal length must be a multiple of \( 2^J \) in order to decompose to level \( J_0 \).

Another disadvantage of the DWT is that it is not translation invariant ie if we calculate the wavelet transform using another starting point in the series then the multiresolution analysis will be different. The wavelet coefficients are basically the differences between weighted averages and the intervals over which these averages are calculated are predetermined by the DWT. This characteristic means that we cannot line up features in the signal with the detail plots at each
3.2. THE WAVELET TRANSFORM

scale. There is another form of the DWT that overcomes these disadvantages.

3.2.2 The Maximum Overlap DWT (MODWT)

This version of the DWT is also known as ‘translation invariant DWT’ or ‘non-decimated DWT’ amongst others. Whereas the DWT described above was a non-redundant orthogonal transform, yielding just enough information to reconstruct the signal, the MODWT is a highly redundant nonorthogonal transform. Instead of translating through the signal in steps of powers of 2, each step in each scale is 1 unit, thus yielding the maximum overlap when calculating the coefficients. Each detail and approximation series are thus the same length as the original signal.

The MODWT is defined for signals of any length, not restricted to lengths that are powers of 2 or integral multiples of powers of 2.

The MODWT is also translation invariant thus making it easier to line up features of the signal with the detail and approximation plots. This is because the wavelet coefficients are differences in averages where the averages are calculated over all possible intervals at the given scale. For example suppose we had a signal of length 8 and we used the Haar wavelet. The last detail series with the DWT would consist of one component, being the difference between the average of the first 4 values and the last 4 values. In contrast, the last detail series for the MODWT would consist of 8 components, the first being the difference between the average of the first 4 values and the last 4 values; the second being the difference between the average of the values at times 2 - 5 and the average of the values at times 1, 6 - 8; the third component being the difference between the average of the values at times 3 - 6 and the average of the values at times 1, 2, 7 and 8 etc.

Another difference between the DWT and MODWT is that the wavelet and scaling filters for the MODWT are the corresponding values for the DWT filters divided by $\sqrt{2}$.

3.2.3 Boundary Effects

The wavelet coefficients and hence the detail and smooth plots are calculated using a circularity assumption, that is, the signal is assumed to be periodic and if the calculations need to go beyond the boundary then, for example, $x_1$ is assumed to be an accurate estimate of $x_{N+1}$. If the signal is periodic and the length of the signal equals the period then the circularity assumption does not create any problems. Usually, however, the circularity assumption is not valid and the ends of the detail and smooth plots are not true representations of the signal. We need to be able to determine how much of the detail and smooth plots are influenced by the periodic boundary conditions.

Suppose we have a wavelet filter of length $L$, ie the filter has $L$ components. For level $j$ the following two definitions apply:

$$L_j = (2^j - 1)(L - 1) + 1 \quad (3.24)$$
\[ L'_j = \left\lceil (L - 2)(1 - \frac{1}{2^j}) \right\rceil \] (3.25)

where \( \lceil x \rceil \) means the least integer greater than or equal to \( x \).

For the DWT detail and smooth plots at level \( j \), the first \( 2^j L'_j \) elements and the last \( L_j - 2^j \) elements are influenced by the boundary conditions. This applies to all smooth plots except the final plot at level \( J \) where \( N = 2^J \) as the plot is a constant value proportional to the signal mean and therefore not influenced by the circularity assumption.

The number of coefficients influenced by the boundary conditions increases as \( L \) and \( j \) increase. Note that the Haar wavelet produces detail and smooth plots that are not influenced by the boundary conditions since \( L = 2 \), giving \( L'_j = 0 \) and \( L_j = 2^j \).

For the MODWT detail and smooth plots at level \( j \), the first and last \( L_j - 1 \) coefficients are influenced by the boundary conditions. This means that the boundary conditions exert a greater influence on the MODWT than the DWT.

### 3.2.4 Which wavelet?

The choice of which wavelet to use for the decomposition depends upon a number of factors. One of those factors is the balancing of two conflicting considerations based on the filter length. Filters that have length less than 8 usually introduce artifacts into the detail and smooth plots (artifacts are shapes that appear in the plots purely because of the shape of the wavelet used in the decomposition and have nothing to do with the signal). This is most obvious when the Haar wavelet is used for a DWT.

On the other hand, filters with a large number of components have a larger number of coefficients that are influenced by the boundary conditions and produce plots that less able to relate to the signal.

For the MODWT the influence of artifacts is much reduced and consequently the choice of wavelet is not as clearcut.

In all situations the most appropriate wavelet depends upon the particular application.

In order for a filter \( (h_i : i = 0, 1, 2, ...L - 1) \) of length \( L \) to be a wavelet filter it must satisfy the following conditions:

\[
\sum_{i=0}^{L-1} h_i = 0 \quad \sum_{i=0}^{L-1} h_i^2 = 1 \quad \sum_{i=0}^{L-1} h_i h_{i+2n} = \sum_{n=-\infty}^{\infty} h_i h_{i+2n} = 0 \] (3.26)

for all nonzero integers \( n \).

We also want the filter to produce a transform that can be interpreted in terms of differences of weighted averages of parts of the signal and in order for this to occur further conditions need to be imposed. Daubechies developed a family of wavelets that had compact support (ie a limited number of filter coefficients) and had a finite number of vanishing moments which results in the
wavelet being able to represent polynomial aspects of the signal. These wavelets are denoted DN [1, 26] where N represents the number of nonzero scaling coefficients, or DAUBM [40] where M represents the number of vanishing moments. Note that D6 is the same as DAUB3 and Addison [1] gives coefficients that are \( \sqrt{2} \) times those in Percival and Walden [26] or Vidakovic [40]. The Haar wavelet is the first in the family of Daubechies wavelets. All these wavelets are asymmetric and Daubechies developed a family of more symmetric wavelets that are termed symmlets or the least asymmetric (LA) family of scaling filters.

### 3.2.5 An example

To illustrate the concepts introduced so far we can look at a time series recording the monthly averages of the daily stages (heights) of the Rio Negro at Manaus, 18km upstream from the confluence of the Rio Negro with the Amazon. The data cover 90 years from January 1903 until December 1992, a total of 1080 values. The first 10 years (120 observations) of these data have been analysed by Davison and Hinkley [10]. As wavelet decomposition requires the number of observations to be a power of 2, the first 128 values were used for comparison purposes.

The observations exhibit a seasonal component which was removed by dividing by the monthly averages over the full 90 year period. The average of the first 128 deseasonalised values was then subtracted to yield a zero-mean series.

The data and the DWT to level 4 are shown in Figure 3.3. The LA(8) wavelet was used in the decomposition. This wavelet is a good compromise between filter length and smoothness. The detail coefficients are shown in \( W_1 \) to \( W_4 \) with the level 4 smooth coefficients in \( V_4 \).

The corresponding MODWT is shown in Figure 3.5, again using the LA(8) wavelet. The number of detail coefficients at each level equals the number of observations, rather than reducing by a factor of 2 from level to another as in the DWT.

Figures 3.4 and 3.6 show the autocorrelation functions for the detail coefficients from the DWT and MODWT respectively. In Figure 3.4 it is apparent that the coefficients are correlated to level 3 at least. Note that there are 16 coefficients at this level and only 8 in level 4. The coefficients in the MODWT decomposition are correlated at every level which could be expected because of the unit step in the transform.

### 3.3 Appropriately resampling a wavelet transform

We can conclude that the MODWT is not suitable for decorrelating dependent data because the autocorrelation is still present in the wavelet coefficients at every level.

The DWT may be suitable for resampling if the detail coefficients exhibit little correlation. Breakspear et al [5] proposed various methods for resampling the wavelet coefficients obtained from a number of time series sources without regarding any correlation within each level of the
decomposition. The three methods were:

1. free permutation of wavelet coefficients within each scale
2. cyclic rotation of coefficients within each scale
3. block resampling of wavelet coefficients within each scale

It was found that the third method achieved the best results with data for which correlation remained in the wavelet coefficients. This comes as no surprise as block resampling is one method adopted by some authors (eg Davison and Hinkley [10]) to resample correlated data. Since the wavelet coefficients are also correlated then block resampling of these coefficients would be expected to yield reasonable results.

However a number of authors have pointed out the pitfalls of resampling the DWT coefficients, giving examples of time series where resampling a DWT is inappropriate. For example, Percival et al [25] conclude that the DWT is appropriate for long memory processes but not suitable for certain short-memory processes, particularly MA(1). Tang [38] has demonstrated ‘that for a simple linear regression model typical of those used in functional MRI experiments the wavestrap produces undercoverage of the confidence intervals of the slope coefficient, which is of primary scientific interest. This result comes from underestimation of the sampling variation of the slope coefficient in the fitted model. Thus, coverage on regression coefficients from wavestrap resamples may not be accurate. This defect extends to more complicated data such as fractional Brownian motion processes as well’.

Percival [25] investigated a generalisation of the DWT, termed the Discrete Wavelet Packet Transform (DWPT), and concluded that it was suitable for resampling short-memory processes.
3.3. APPROPRIATELY RESAMPLING A WAVELET TRANSFORM

The Discrete Wavelet Packet Transform (DWPT) and Wavestrapping

The Discrete Wavelet Packet Transform [26, 40, 1] generalises the approach taken in the DWT. Instead of decomposing one level into a subsequent detail vector and smooth vector and only continuing to decompose the latter, the DWPT continues to decompose both detail and smooth vectors. The resulting vectors, $w_{j,n}$, $n = (0,1,...,2^n - 1)$, are called crystals. The wavelet packet table can be used to select an orthonormal basis consisting of non-overlapping crystals that span the data with the crystals being selected by some cost function.

Figure 3.7 shows the DWT level diagram with the data in $W_{0,0}$ decomposed into smooth and detail vectors from level 1 to level 3. Note that only the smooth vectors, $W_{j,0}$, are decomposed into smooth and detail vectors at the next level.

In contrast, Figure 3.8 shows the full packet table for the DWPT where both smooth and detail vectors are further decomposed.

The wavelet packet table is useful for resampling purposes because crystals that have elements that appear to be independent can be chosen to be part of a basis. These crystals can be resampled and the wavelet transform then inverted to yield a surrogate series. The method was developed by Percival et al [25, 24] and given the name ‘wavestrapping’. It consists of the following steps:
Let $X$ be a time series of length $2^J$. Calculate a wavelet packet table to level $J_0 = J - 2$. Let $j = n = 0$ and $W_{0,0} = X$.

2. For $j < J_0$ test $W_{j,n}$ for white noise. If the null hypothesis of white noise is rejected decompose $W_{j,n}$ into $W_{j+1,2n}$ and $W_{j+1,2n+1}$, otherwise keep $W_{j,n}$ as part of the basis. Stop when $j = J_0$.

3. The basis consists of all the $W_{j,n}$ that weren’t rejected by the white noise test. The elements of these vectors are resampled.

4. Apply the inverse transform to the resampled vectors to produce a wavestrapped time series.

3.3.2 Wavestrapping the Manaus river data

The R package *waveslim* was used to obtain the DWPT for the Manaus river data and Figures 3.9 to 3.11 show the autocorrelation functions for the coefficients in the crystals from level 1 to level 4. As before, the LA(8) wavelet is used for the decomposition.

Figures 3.9 to 3.11 show the autocorrelation functions (ACF) for crystals $(1,0)$ to $(4,3)$. As there were 128 observations there are only 8 coefficients in each level 4 crystal.

The presence or absence of autocorrelation is estimated from the use of the Portmanteau test. In this case the crystals $(2,0)$, $(3,2)$, $(3,4)$, $(3,5)$, $(3,7)$, $(4,6)$, $(4,12)$, $(4,13)$, $(5,14)$ and $(5,15)$ form the basis. The coefficients in all these crystals except $(2,0)$ are resampled and the Inverse DWPT yields surrogate series.

Figure 3.12 shows some typical surrogates plotted against the data while Figure 3.13(a) shows
3.3. Appropriately Resampling a Wavelet Transform

\[ \text{ACF} - W1 \]
\[ \text{ACF} - W2 \]
\[ \text{ACF} - W3 \]
\[ \text{ACF} - W4 \]

Figure 3.6: Manaus river: autocorrelation functions for the detail coefficients in levels 1 to 4 of the MODWT decomposition

The mean of all 1000 surrogates against the data. The six surrogates shown appear as though they could be surrogates of the data and their extreme values are similar to those of the data. The mean of the surrogates exhibits a similar pattern to that of the data.

The largest and smallest values in each surrogate were plotted against the corresponding values in the data, as shown in Figure 3.13(b). This checks for abnormally extreme values obtained when a large resampled coefficient in a crystal at one level combines with another large coefficient at another level. While most of the surrogates have extremes greater than the data, these don’t appear to be abnormal.

Davison and Hinkley [10] fitted an AR(3) model to their data consisting of 120 observations and so the same type of model will be fit to the data used here (the same 120 observations plus eight more) and each surrogate. The fitted parameter values for the data were (0.059, 0.321, 0.202) while the corresponding mean values for the surrogates were (0.116, 0.195, 0.255) with standard deviations (0.056, 0.065, 0.051). Hence the surrogates appear to be slightly more correlated at lag 1 and less correlated at lag 2, compared with the data.

The procedure was repeated using the Haar wavelet for comparison purposes. This wavelet has only two components and would therefore be expected to result in less smooth outcomes than achieved with the LA(8) wavelet.

The basis obtained with the Haar wavelet consists of the crystals (2, 1), (3, 0), (3, 1), (3, 7) and (4, 8) to (4, 13). Thus correlation is not removed until level 3 with the Haar wavelet, compared with level 2 with the LA(8) wavelet. Whereas the latter wavelet had two crystals at level 5 in the
Figure 3.7: DWT level diagram from ‘Wavelet Methods for Time Series Analysis’ by Percival and Walden. The data is $W_{0,0}$ while the smooth and detail vectors at level $j$ are $W_{j,0}$ and $W_{j,1}$ respectively.

Crystal (3, 0) is not resampled but the remainder are. Figure 3.14 shows some typical surrogates against the data. There is a tendency for this wavelet to produce surrogate values that are more extreme than the surrounding data values and these surrogates look less than the data than those surrogates produced using the LA(8) wavelet. This is supported by Figure 3.15(a) which plots the mean of the surrogates against the data. There is less variability in this mean than that from the LA(8) wavelet.

Figure 3.15(b) shows the extreme values in the surrogates and this plot is similar to that obtained with the LA(8) wavelet with the exception of a few larger values, confirming the interpretation of the surrogates plotted in Figure 3.14.

The means of the parameter estimates for the fitted AR(3) model were $(0.306, 0.052, 0.210)$ with standard deviations $(0.075, 0.087, 0.045)$. While $\hat{\rho}_3$ is similar to that from the data $(0.202)$, the estimates of $\rho_1$ and $\rho_2$ are the reverse of the corresponding data estimates $(0.059, 0.321)$.

We can conclude from this brief analysis that resampling of wavelet packets looks promising for producing surrogate time series. Further, the simple Haar wavelet does not appear to be as good as the LA(8) wavelet in maintaining characteristics of the data in the surrogates.
3.3. APPROPRIATELY RESAMPLING A WAVELET TRANSFORM

3.3.1. Introduction

W_0 = X

W_1.0

W_1.1

W_2.0

W_2.1

W_2.2

W_2.3

W_3.0

W_3.1

W_3.2

W_3.3

W_3.4

W_3.5

W_3.6

W_3.7

Figure 3.8: DWPT level diagram from `Wavelet Methods for Time Series Analysis` by Percival and Walden. The detail vectors at each level are also decomposed, contrasting with the DWT in Figure 3.7.

Figure 3.9: Manaus river: autocorrelation functions for the detail coefficients in levels 1 and 2 of the DWPT decomposition using LA(8) wavelet.
Figure 3.10: Manaus river: autocorrelation functions for the detail coefficients in level 3, crystals 0 - 5, of the DWPT decomposition using LA(8) wavelet.

Figure 3.11: Manaus river: autocorrelation functions for the detail coefficients in crystals 3.6 - 4.3 of the DWPT decomposition using LA(8) wavelet.
3.3. Appropriately Resampling a Wavelet Transform

Figure 3.12: Manaus river: data and surrogates (in solid line) using DWPT with the LA(8) wavelet

Figure 3.13: Manaus river using DWPT with the LA(8) wavelet: (a) data and mean of 1000 surrogates and (b) maximum and minimum values of 1000 surrogates. The extreme values in the data are shown as solid lines.
Figure 3.14: Manaus river: data and typical surrogates (in solid line) using DWPT with the Haar wavelet
3.4 The Lifting Scheme - Second Generation Wavelets

The Lifting Scheme was developed by Sweldens [35, 36, 12] and leads to Second Generation Wavelets which are not translates and dilates of a single function. The scheme has three steps:

- split the data into two subsets termed even and odd. If the \( n \) data values are denoted \((y_0, y_1, y_2, \ldots, y_{n-2}, y_{n-1})\) then the \textit{evens} are \((y_0, y_2, \ldots, y_{n-2})\) and the \textit{odds} are \((y_1, y_3, \ldots, y_{n-1})\).
- the \textit{evens} are used to predict the \textit{odds} and the detail or wavelet coefficients are obtained from the difference between the \textit{odds} and their predicted values.
- the \textit{evens} are updated from the wavelet coefficients to maintain the mean.

The updated \textit{evens} are then further decomposed as per the DWT. Alternatively, both \textit{evens} and \textit{odds} can be further decomposed as per the DWPT.

In its simplest form the predicted values are obtained from linear interpolation between the \textit{evens} and if the observations are equally spaced then the predicted value at \textit{odd} position is the simple average of the adjacent \textit{evens}.

To use Sweldens’ [35] terminology, let \( \lambda_{0,k} = y_k \) and then the \textit{evens} are given by subsampling every second observation

\[
\lambda_{1,k} = \lambda_{0,2k}
\]  \hspace{1cm} (3.27)

The wavelet coefficient \((\psi_{1,k})\) is then the difference between the sample value and its predicted value at an \textit{odd} location

\[
\psi_{1,k} = \lambda_{0,2k+1} - \frac{1}{2}(\lambda_{1,k} + \lambda_{1,k+1})
\]  \hspace{1cm} (3.28)
Then to ensure that \( \sum_k \lambda_{1,k} = \frac{1}{2} \sum_k \lambda_{0,k} \) we lift the subsampled coefficients

\[
\lambda_{1,k} = \lambda_{1,k} + \frac{1}{4} (\psi_{1,k-1} + \psi_{1,k})
\]  

(3.29)

Figure 3.16 shows the calculation of the wavelet coefficients using linear interpolation. The \textit{evens} are joined by straight lines and the predicted values at the \textit{odd} locations obtained by interpolation; the wavelet coefficients are then calculated from \textit{actual} \textendash \textit{predicted} values.

The wavelet and subsampled vectors are themselves separated into \textit{evens} and \textit{odds} and the process repeated to yield four vectors. The procedure continues until there is only one coefficient in each crystal.

The inverse just reverses the steps, changing additions to subtractions and vice versa.

Slightly different procedures are required at the start and end of the series. The first observation, \( y_0 \), is an \textit{even} and hence the first and third observations are used to predict the first \textit{odd} value as normal. However there is only a wavelet coefficient to the right of \( y_0 \) and none to the left. Hence \( y_0 \) can only be lifted by \( \frac{1}{4} d_0 \).

The last observation is an \textit{odd} if there are an even number of observations. Hence there is only an \textit{even} to the left and none to the right. In this case the last two \textit{evens} are extrapolated to predict the last \textit{odd} value. More is said about this later.

The Lifting Scheme has three advantages over first generation wavelets:

1. the number of observations, \( n \), does not have to be a power or a multiple of a power of 2.

   The usual case requires \( n \) to be even but the method is also suitable when \( n \) is odd.

2. observations do not have to be evenly spaced. Predicted values are obtained by interpolating between adjacent values and there is no reason why these values cannot be irregularly spaced.

3. there is no need to adopt a boundary assumption as all calculations remain within the length of the given data.

The Lifting Scheme can therefore be applied to more types of data and hence is potentially more useful than the first generation wavelet transform.

### 3.4.1 Resampling the Manaus river data after Lifting

The Lifting Scheme with linear interpolation was used to decompose the Manaus river data into wavelet packets. As before, each packet was tested for white-noise and resampling occurred in those packets deemed to have independent coefficients. The inverse transform then yielded a surrogate series.

The packets in the basis are \((2, 1), (3, 0), (4, 2), (4, 3), (4, 8), (4, 13), (5, 18-25), (5, 28-31)\). Compared to the LA(8) and Haar wavelets we see that linear Lifting retained autocorrelation in the coefficients to a later level.
Figure 3.16: Diagram showing the calculation of the wavelet coefficients in the lifting procedure. The predicted values at the odd locations are obtained from linear interpolation between the evens and the wavelet coefficients are the differences between the actual and predicted values.

Table 3.1: Manaus river data: comparison of Portmanteau test statistics for coefficients in crystal (2, 0) using Lifting. There are $n = 32$ coefficients in $W(2, 0)$.

<table>
<thead>
<tr>
<th>Lags in test</th>
<th>Test statistic</th>
<th>$\chi^2$</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Min}(10, \frac{n}{2})$</td>
<td>16.616</td>
<td>16.919</td>
<td>9</td>
</tr>
<tr>
<td>$\text{Max}(3, \frac{n}{2})$</td>
<td>25.111</td>
<td>24.996</td>
<td>15</td>
</tr>
</tbody>
</table>

Figure 3.17 shows typical surrogates from this procedure. These surrogates do not appear to have the same characteristics as the data so the procedure was examined more closely. There are two areas where improvements can be made:

- The Portmanteau test classified the smooth packet at level 2, denoted by $W(2, 0)$, as having correlated coefficients and was therefore further decomposed. The ACF plot seems to indicate that correlation is not present. The test statistic is 25.111 using 16 lags, compared with the $\chi^2$ critical value of 24.996 with $df = 15$ at the 95\% level. Hence the test statistic is borderline.

To examine the effect of bringing $W(2, 0)$ into the basis the number of lags included into the test was reduced to 10. Table 3.1 shows that the test statistic is now 16.616, compared to the $\chi^2$ critical value of 16.919 with $df = 9$. Figure 3.19 shows typical surrogates when $W(2, 0)$ is included in the basis and these appear more reasonable. Table 3.2 summarises the basis crystals for all the decomposition methods.

- The last of the odds is predicted by extrapolating the final two evens and this can create instability in the inverse transform. An alternative is to use the `lazy wavelet' at the end so that the final detail coefficient is the last even value less the last odd value. Figure 3.18 shows the method.

The analysis continued after these two alterations were made to the procedure. Figures 3.20 to 3.22 show the ACF plots for packets in levels 1 and 2 and crystals 3.4 to 3.7.
CHAPTER 3. WAVELET THEORY

Figure 3.17: Manaus river: data and typical surrogates using lifting with original white-noise test and extrapolation for the final predicted value

Table 3.2: Manaus river data: basis crystals selected after wavelet decomposition and testing for white noise using three different methods - DWPT with the Haar and LA(8) wavelets compared with Lifting.

<table>
<thead>
<tr>
<th>Method</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
<th>Level 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>DWPT with Haar wavelet</td>
<td>1, 0, 7</td>
<td>8 - 13</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>DWPT with LA(8) wavelet</td>
<td>0, 2, 4, 5, 7</td>
<td>14, 15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lifting - original Portmanteau test</td>
<td>1, 0, 2, 3, 8, 13</td>
<td>18 - 25, 28 - 31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lifting - amended Portmanteau test</td>
<td>0, 1, -</td>
<td>8, 13</td>
<td>18 - 25, 28 - 31</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.23 shows some typical surrogates and while there are some values that exceed the extremes of the data, overall the plots look promising and worthy of further analysis. The mean of the 1000 surrogates is shown against the data in Figure 3.24(a) and this plot is very similar to that obtained using the LA(8) wavelet.

The extreme values for all the surrogates are shown in Figure 3.24(b). Comparing this plot with Figures 3.13(b) and 3.15(b) we see that the extreme values obtained with Lifting generally exceed those obtained with the LA(8) and Haar wavelets.

Again estimates of the AR(3) parameter values were obtained for all 1000 surrogates. The means of the parameter estimates for the fitted AR(3) model were (0.092, 0.190, 0.195) with standard deviations (0.067, 0.071, 0.065). These values are shown in Table 3.3 together with corresponding results from resampling with the LA(8) and Haar wavelets. The values of $\hat{\rho}_2$ are virtually identical for Lifting and the LA(8) wavelet while Lifting yields estimates of $\rho_1$ and $\rho_3$ that are closer to the data values.
3.5. AN ISSUE TO CONSIDER

Figure 3.18: Diagram showing the calculation of the detail coefficients in the lifting procedure with the lazy wavelet at the end. The predicted value at the last location is the actual value at the penultimate location

Table 3.3: Manaus river data: parameter estimates using an AR(3) model. The estimates for the data are shown together with the mean and standard deviation of the estimates from 1000 surrogates using three different methods - DWPT with the Haar and LA(8) wavelets compared with Lifting.

<table>
<thead>
<tr>
<th>Series</th>
<th>$\hat{\rho}_1$</th>
<th>$\hat{\rho}_2$</th>
<th>$\hat{\rho}_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>0.059</td>
<td>0.321</td>
<td>0.202</td>
</tr>
<tr>
<td>DWPT with Haar wavelet</td>
<td>0.306 (0.075)</td>
<td>0.052 (0.087)</td>
<td>0.210 (0.045)</td>
</tr>
<tr>
<td>DWPT with LA(8) wavelet</td>
<td>0.116 (0.059)</td>
<td>0.195 (0.065)</td>
<td>0.255 (0.051)</td>
</tr>
<tr>
<td>Lifting</td>
<td>0.092 (0.067)</td>
<td>0.190 (0.071)</td>
<td>0.195 (0.065)</td>
</tr>
</tbody>
</table>

3.5 An issue to consider

It was noted that the Portmanteau test statistic for the $W(2, 0)$ coefficients using Lifting was slightly higher than the critical value and hence classified the coefficients as being correlated. It was also noted that surrogates that appeared to have characteristics similar to the data were obtained when wavelet packet crystals were resampled at a higher level, i.e., at an earlier stage of the decomposition. Alternative test statistics are examined in a later section; these alternatives may have classified the $W(2, 0)$ coefficients as white-noise in this case.

Consideration could also be given to combining a white-noise test statistic with a penalty function that imposes a penalty for decomposing to another level. This would act in the same way as AIC which balances the decrease in variance obtained by adding another parameter to a model with the increase in model complexity.

Each increase in the level of wavelet packet decomposition results in halving the number of coefficients in a crystal. This reduces the variation in the resamples. In the Manaus river example there were 8 coefficients in each crystal in level 4 and 4 coefficients in each level 5 crystal. Tang [38] raised this issue as a potential problem, stating that the method ‘can result in resampled series with too much of the structure repeated’. Their solution was to introduce the parametric
bootstrap where ‘instead of resampling the observed detail coefficients, the detail coefficients can be generated from $N(0, 1)$’. This was justified by the statement ‘Since the Daubechies wavelet bases are orthonormal, the detail coefficients at all levels and the scaling coefficients theoretically follow independent normal distributions if the original data are standard normal white noise.’

This question needs to be considered: Is it better to accept some correlation and resample 8 coefficients or resample 4 coefficients in a crystal that is classified as white-noise? Wavestrapping with the DWT ignored correlation within the wavelet coefficients while resampling with the DWPT only occurred when the wavelet coefficients were deemed independent, regardless of how many coefficients there were to resample.

The wavelet packet table enables many different orthonormal transforms to be selected of which one can be considered optimal. The best basis algorithm [26, 40] was developed to find the optimal transform and is based on minimising an additive cost function. It may be worthwhile to consider a similar approach here rather than relying on a simple autocorrelation test.

3.6 Conclusion

The example using the Manaus river data has illustrated the wavestrapping method for resampling dependent data. The similarity of the resulting surrogate series to the data depends upon
3.6. CONCLUSION

The wavelet chosen for the decomposition with the more complex LA(8) wavelet achieving better results than the simple Haar wavelet. It appears that resampling the DWPT is superior to resampling the DWT because it aims to remove all autocorrelation before resampling occurs. However a potential problem can occur when there are only a few coefficients in a crystal available to be resampled.

The Lifting Method, combined with the DWPT, gave comparable results to those obtained with the LA(8) wavelet. Since it is applicable to more data types than the use of first generation wavelets it is worthwhile investigating its performance with other time series, specifically AR(1) and MA(1) models.
Figure 3.21: Manaus river: autocorrelation functions for the coefficients in level 2 of the decomposition using lifting

Figure 3.22: Manaus river: autocorrelation functions for the coefficients in crystals 3.4 - 3.7 of the decomposition using lifting
3.6. CONCLUSION

Figure 3.23: Manaus river: data and typical surrogate using lifting with altered white-noise test and using the lazy wavelet for the final predicted value

Figure 3.24: Manaus river using lifting with altered white-noise test and using the lazy wavelet for the final predicted value: (a) data and mean of 1000 surrogates and (b) maximum and minimum values of 1000 surrogates. The extreme values in the data are shown as solid lines
Chapter 4

Wavelet Resampling of Time Series

4.1 Introduction

The previous chapter concluded by adopting the DWPT combined with wavelet lifting as the method for producing decorrelated data able to be resampled; inverting the transform then yielded surrogate series. This chapter examines the application of the method to one-parameter autoregressive and moving average time series.

Section 2 examines the decorrelation properties of the lifting scheme when applied to such series. Section 3 looks at the parameter estimates of the surrogate series after resampling AR(1) and MA(1) processes with parameters $\phi$ or $\theta$ given as $(\pm 0.1, \pm 0.2, ... , \pm 0.9)$. The effectiveness of the resampling algorithm was then examined in greater depth in Section 4 for AR(1) models with $\phi = \pm 0.5$ and MA(1) models with $\theta = \pm 0.5$. Alternative white noise tests are examined in Section 5.

4.2 Decorrelation properties

This section examines the decorrelation properties of the lifting scheme. The autocorrelation function is derived for Level 1 smooth and detail coefficients obtained from applying the wavelet lifting algorithm to AR(1) and MA(1) models. Comparisons are made between the autocorrelation values for various parameter choices.

Recall that the lifting scheme consists of three steps:

1. split the data into even and odd indexed components;
2. predict the odd values by interpolating the even values and calculate the detail coefficients as the difference between actual and predicted values;

3. update the even values by a multiple of the adjacent detail coefficients, thus maintaining the mean. The resulting values are called smooth coefficients.

Linear interpolation is used to find the predicted values and thus the detail coefficients at Level 1 are calculated from the difference between the odd indexed sample values and the average of the adjacent even indexed sample values. Let \( x'_{2t+1} \) be the detail coefficient at time \( 2t+1 \) while \( x_{2t+1} \) is the sample value at time \( 2t+1 \). Then

\[
x'_{2t+1} = x_{2t+1} - \frac{1}{2}(x_{2t} + x_{2t+2}) \tag{4.1}
\]

Subsequent detail coefficients are derived from the relevant values multiples of two time periods later. For example:

\[
x'_{2t+3} = x_{2t+3} - \frac{1}{2}(x_{2t+2} + x_{2t+4}) \tag{4.2}
\]

The even indexed values of the data are then lifted by one-quarter of the sum of the immediately adjacent detail coefficients. Thus the even indexed value at time \( 2t \) becomes

\[
x'_{2t} = x_{2t} + \frac{1}{4}(x'_{2t-1} + x'_{2t+1}) \tag{4.3}
\]

In terms of the original data this becomes

\[
x'_{2t} = -\frac{1}{8} x_{2t-2} + \frac{1}{4} x_{2t-1} + \frac{3}{4} x_{2t} + \frac{1}{4} x_{2t+1} - \frac{1}{8} x_{2t+2} \tag{4.4}
\]

### 4.2.1 AR(1) processes

**a) Detail coefficients at Level 1**

The autocovariance function for an AR(1) process at lag \( h \) with parameter \( \phi \) and variance \( \sigma^2 \) is

\[
\gamma_h = \frac{\sigma^2 \phi^h}{1 - \phi^2} \tag{4.5}
\]

To determine the autocovariance between detail coefficients we use the formula for calculating the covariance between two linear combinations of random variables. Given \( L_1 \) is a linear combination of random variables \( Y_i \) and \( L_2 \) is a linear combination of random variables \( X_i \), that is

\[
L_1 = \sum_{i=1}^{n} a_i Y_i \quad \text{and} \quad L_2 = \sum_{j=1}^{m} b_j X_j
\]

the covariance between \( L_1 \) and \( L_2 \) is

\[
Cov(L_1, L_2) = \sum_{i=1}^{n} \sum_{j=1}^{m} a_ib_j Cov(Y_i, X_j) \tag{4.6}
\]
Table 4.1: Theoretical autocovariance and autocorrelation functions for Level 1 smooth coefficients after lifting an AR(1) process

<table>
<thead>
<tr>
<th>Lag</th>
<th>Autocovariance</th>
<th>Autocorrelation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(-\sigma^2(\phi^3 - 5\phi^2 - 3\phi + 23))</td>
<td>(1)</td>
</tr>
<tr>
<td>1</td>
<td>(-\sigma^2(\phi^2 - 4\phi + 1)(\phi^3 - \phi^2 - 8\phi - 8))</td>
<td>(\frac{(\phi^2 - 4\phi + 1)(\phi^3 - \phi^2 - 8\phi - 8)}{2(\phi^3 - 5\phi^2 - 3\phi + 23)})</td>
</tr>
<tr>
<td>2</td>
<td>(-\sigma^2(\phi^2 - 4\phi + 1)(\phi + 1)^3)</td>
<td>(\frac{(\phi^2 - 4\phi + 1)^2(\phi + 1)^3}{2(\phi^3 - 5\phi^2 - 3\phi + 23)})</td>
</tr>
</tbody>
</table>

Hence the autocovariance at lag 0, using \(\text{Cov}(X'_t, X'_t)\), is

\[
\gamma_0 = -\frac{\sigma^2(\phi - 3)}{2(\phi + 1)} \quad (4.7)
\]

For lags \(\geq 1\) the autocovariance takes the form:

\[
\gamma_h = -\frac{\sigma^2(\phi - 1)^3\phi^{2(h-1)}}{4(\phi + 1)} \quad (4.8)
\]

Hence the autocorrelation at lag \(h \geq 1\) is given by:

\[
\rho_h = \frac{(\phi - 1)^3\phi^{2(h-1)}}{2(\phi - 3)} \quad (4.9)
\]

These autocorrelations are always positive for \(-1 < \phi < 1\). Note that the ratio of successive autocorrelations for lags \(\geq 2\) equals \(\phi^2\); i.e., \(\frac{\rho_h}{\rho_{h-1}} = \phi^2\) for \(h \geq 2\).

b) Smooth coefficients at Level 1

The autocovariance and autocorrelation functions at lags \(0 - 2\) are shown in Table 4.1. For lags \(\geq 3\) the autocovariance at lag \(h\) is \(\phi^2\) times that at lag \((h - 1)\). The autocorrelations have a zero at \(\phi = (2 - \sqrt{3})\) and are non-negative for lags \(\geq 2\). Plots of the autocorrelation functions against \(\phi\) for lags 1 and 2 are shown in Figure 4.1.

Example

The formulae produce totally different results according to the magnitude and sign of \(\phi\). To illustrate, the theoretical autocorrelations for Level 1 detail coefficients when \(\phi = \pm 0.2\) and \(\pm 0.8\) are shown in Table 4.2.

These values indicate that the coefficients appear to follow an autoregressive model when \(\phi\) is highly negative (slowly decaying autocorrelations) and an MA(1) model when \(\phi\) is less negative (one significant autocorrelation). The coefficients appear to be uncorrelated when \(\phi\) is positive.

It is apparent, therefore, that more steps of the lifting algorithm will be required to yield inde-
4.2. DECORRELATION PROPERTIES

Figure 4.1: Autocorrelation functions at lags 1 and 2 for detail and smooth coefficients at Level 1 after lifting an AR(1) process

pendent coefficients when $\phi$ is highly negative.

The value of these theoretical autocorrelations for Level 1 smooth coefficients when $\phi = \pm 0.2$ and $\pm 0.8$ are shown in Table 4.3. It is apparent that these coefficients behave like an MA(1) process when $\phi$ is negative and an autoregressive process for high positive values of $\phi$.

Table 4.2: Theoretical autocorrelations for Level 1 detail coefficients after lifting an AR(1) process with parameter $\pm 0.2, \pm 0.8$

<table>
<thead>
<tr>
<th>Lag</th>
<th>$\phi = -0.8$</th>
<th>$\phi = -0.2$</th>
<th>$\phi = 0.2$</th>
<th>$\phi = 0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>1</td>
<td>0.7674</td>
<td>0.2700</td>
<td>0.0914</td>
<td>0.0018</td>
</tr>
<tr>
<td>2</td>
<td>0.4911</td>
<td>0.0108</td>
<td>0.0037</td>
<td>0.0012</td>
</tr>
<tr>
<td>3</td>
<td>0.3143</td>
<td>0.0004</td>
<td>0.0001</td>
<td>0.0007</td>
</tr>
<tr>
<td>4</td>
<td>0.2012</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0005</td>
</tr>
<tr>
<td>5</td>
<td>0.1287</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0003</td>
</tr>
</tbody>
</table>

Table 4.3: Theoretical autocorrelations for Level 1 smooth coefficients after lifting an AR(1) process with parameter $\pm 0.2, \pm 0.8$

<table>
<thead>
<tr>
<th>Lag</th>
<th>$\phi = -0.8$</th>
<th>$\phi = -0.2$</th>
<th>$\phi = 0.2$</th>
<th>$\phi = 0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>1</td>
<td>-0.3071</td>
<td>-0.2536</td>
<td>-0.0520</td>
<td>0.6326</td>
</tr>
<tr>
<td>2</td>
<td>0.0043</td>
<td>0.0108</td>
<td>0.0037</td>
<td>0.3962</td>
</tr>
<tr>
<td>3</td>
<td>0.0028</td>
<td>0.0015</td>
<td>0.0001</td>
<td>0.2536</td>
</tr>
<tr>
<td>4</td>
<td>0.0018</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.1623</td>
</tr>
<tr>
<td>5</td>
<td>0.0011</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.1039</td>
</tr>
</tbody>
</table>
4.2.2 MA(1) processes

a) Detail coefficients at Level 1

The autocovariance function for an MA(1) process is non-zero only at lags 0 and 1. Given the parameter $\theta$ and variance $\sigma^2$ the autocovariances at these lags are $\sigma^2(1 + \theta^2)$ and $\sigma^2\theta$ respectively.

Similarly, the autocovariances for the detail coefficients at Level 1 only exist for lags 0 and 1 as given in Table 4.4. The autocorrelation at lag 1 is therefore $\frac{\theta^2 - 4\theta + 1}{2(3\theta^2 - 4\theta + 3)}$ which has a zero at $\theta = (2 - \sqrt{3})$. A plot of the theoretical autocorrelation function is shown in Figure 4.2. Of note is the gradual reduction in autocorrelation as $\theta$ increases from $-1$ to 0, followed by a more rapid decline for parameter values up to around 0.6. Specifically, $\rho_1$ decreases from 0.30 when $\theta = -0.9$ to 0.26 when $\theta = -0.3$ and from $-0.21$ to $-0.49$ as $\theta$ increases from 0.5 to 0.9.

Table 4.4: Theoretical autocovariance functions for Level 1 detail coefficients after lifting an MA(1) process

<table>
<thead>
<tr>
<th>Lag</th>
<th>Autocovariance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\sigma^2((\theta - 1)^2 + \frac{1}{2}(\theta^2 + 1))$</td>
</tr>
<tr>
<td>1</td>
<td>$\sigma^2\left(\frac{1}{2}(1 + \theta^2) - \theta\right)$</td>
</tr>
</tbody>
</table>

Figure 4.2: Autocorrelation function at lag 1 for detail coefficients at Level 1 after lifting an MA(1) process

b) Smooth coefficients at Level 1

The autocovariance function for the smooth coefficients at level 1 arising from an MA(1) process is non-zero only up to and including lag 2. Table 4.5 shows these autocovariances and resulting autocorrelations. Figure 4.3 plots the theoretical autocorrelations for lags 1 and 2. Note that the autocorrelation at lag 1 is always negative while that at lag 2 has a zero at $\theta = (2 - \sqrt{3})$. Of note are the observations that autocorrelations at lag 1 are only significant for $\theta \leq -0.2$ and the flatness of the lag 2 autocorrelation function for positive values of $\theta$, notwithstanding that the autocorrelation is low for all values of $\theta$. 
4.3. PARAMETER AND VARIANCE ESTIMATES

Figure 4.3: Autocorrelation functions for smooth coefficients at Level 1 after lifting an MA(1) process: (a) lag 1 and (b) lag 2

Table 4.5: Theoretical autocovariance functions for Level 1 smooth coefficients after lifting an MA(1) process

<table>
<thead>
<tr>
<th>Lag</th>
<th>Autocovariance</th>
<th>Autocorrelation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\sigma^2/32(23\theta^2 + 20\theta + 23)$</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>$-\sigma^2/8(\theta - 1)^2$</td>
<td>$-4(\theta - 1)^2/(23\theta^2 + 20\theta + 23)$</td>
</tr>
<tr>
<td>2</td>
<td>$\sigma^2/64(\theta^2 - 4\theta + 1)$</td>
<td>$(\theta^2 - 4\theta + 1)/(2(23\theta^2 + 20\theta + 23))$</td>
</tr>
</tbody>
</table>

4.2.3 Conclusion

This section has shown that the lifting scheme shares the decorrelation property of wavelets, although this depends upon the process yielding the data (autoregressive or moving average) and the parameter values of that process. For an AR(1) process, lifting produces the least correlated detail coefficients when the parameter $\phi$ is positive while the smooth coefficients are uncorrelated for parameter values close to 0.

The reverse applies for MA(1) processes. The correlation of the detail coefficients is generally low, with the lowest values occurring when the parameter $\theta$ is close to 0 while the smooth coefficients are least correlated when $\theta$ is larger than 0.5.

4.3 Parameter and variance estimates

Mathematica’s Time Series package [13] was used to provide simulated series for AR(1) and MA(1) models with the values $(\pm 0.1, \pm 0.2, ..., \pm 0.9)$ for the respective parameter, $\phi$ or $\theta$.

Each simulated series consisted of the last 256 of 5000 generated values with white noise dis-
CHAPTER 4. WAVELET RESAMPLING OF TIME SERIES

Table 4.6: Estimates of parameters and variance for AR(1) simulations with $\phi > 0$. Note that $\hat{\phi}_D$ and $\hat{\sigma}^2_D$ refer to estimates from the data.

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>$\hat{\phi}$</th>
<th>$\sigma_{\hat{\phi}}$</th>
<th>$\hat{\phi}_D$</th>
<th>$\hat{\sigma}^2$</th>
<th>$\sigma_{\hat{\sigma}^2}$</th>
<th>$\hat{\sigma}^2_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.198</td>
<td>0.0487</td>
<td>0.189</td>
<td>5191.3</td>
<td>472.0</td>
<td>5210.7</td>
</tr>
<tr>
<td>0.2</td>
<td>0.287</td>
<td>0.0451</td>
<td>0.288</td>
<td>5172.7</td>
<td>438.6</td>
<td>5210.0</td>
</tr>
<tr>
<td>0.3</td>
<td>0.379</td>
<td>0.0403</td>
<td>0.386</td>
<td>5175.6</td>
<td>411.9</td>
<td>5209.7</td>
</tr>
<tr>
<td>0.4</td>
<td>0.474</td>
<td>0.0348</td>
<td>0.483</td>
<td>5190.9</td>
<td>390.9</td>
<td>5210.8</td>
</tr>
<tr>
<td>0.5</td>
<td>0.581</td>
<td>0.0322</td>
<td>0.579</td>
<td>5254.8</td>
<td>442.3</td>
<td>5215.5</td>
</tr>
<tr>
<td>0.6</td>
<td>0.664</td>
<td>0.0268</td>
<td>0.672</td>
<td>5231.6</td>
<td>430.3</td>
<td>5227.4</td>
</tr>
<tr>
<td>0.7</td>
<td>0.748</td>
<td>0.0206</td>
<td>0.760</td>
<td>5290.5</td>
<td>417.2</td>
<td>5253.4</td>
</tr>
<tr>
<td>0.8</td>
<td>0.829</td>
<td>0.0143</td>
<td>0.841</td>
<td>5387.2</td>
<td>404.0</td>
<td>5304.6</td>
</tr>
<tr>
<td>0.9</td>
<td>0.907</td>
<td>0.0083</td>
<td>0.915</td>
<td>5502.9</td>
<td>393.6</td>
<td>5387.1</td>
</tr>
</tbody>
</table>

tributed as $N(0, 5000)$. One thousand surrogates were obtained for each parameter value and estimates of the parameters and the variance were found for each surrogate. The Levinson-Durbin algorithm was used to obtain the estimates for the AR(1) models while the Innovation algorithm was used for the MA(1) models. The distributions of these estimates were summarised by their means and standard deviations, as shown in Tables 4.6 to 4.9.

Table 4.6 shows the parameter and variance estimates for AR(1) models with the positive values of $\phi$ given in column 1. The mean of the estimates of $\phi$ from the surrogates is shown in column 2, with the standard deviation of the estimates in column 3. The estimate of $\phi$ from the data, denoted by $\hat{\phi}_D$, is given in column 4. The mean of the variance estimates from the surrogates is shown in column 5, with the standard deviation of the estimates in column 6 and finally the estimate of the variance from the data in column 7 ($\hat{\sigma}^2_D$).

It is apparent from Table 4.6 that:

(i) the means of the parameter estimates from the surrogates are very similar to the estimates from the data and converge toward the nominal values as $\phi$ increases;

(ii) the standard deviations of the parameter estimates from the surrogates show a continual decrease from 0.05 ($\phi = 0.1$) to 0.008 ($\phi = 0.9$);

(iii) the mean of the variance estimates from the surrogates are very similar to the estimate from the data and diverge from the nominal value as $\phi$ increases;

(iv) the standard deviations of the variance estimates from the surrogates are consistent across the range of parameter values.

Table 4.7 shows the results for AR(1) series with negative parameters and it is observed that:

(i) the means of the parameter estimates from the surrogates are quite different to the estimates from the data and the nominal values for $\phi$ in $[-0.5, -0.1]$. There is better agreement for $\phi$ in $[-0.9, -0.6]$;
4.3. PARAMETER AND VARIANCE ESTIMATES

(ii) the standard deviations of the parameter estimates from the surrogates lie between 0.075 and 0.1 except for the value of 0.05 when $\phi = -0.9$;

(iii) the mean of the variance estimates from the surrogates are consistently larger than the estimate from the data and diverge from the nominal value as $\phi$ decreases. The mean variance is 24% higher than the nominal value when $\phi = -0.1$ and 69% higher when $\phi = -0.9$;

(iv) the standard deviations of the variance estimates from the surrogates increase as $\phi$ decreases from $-0.1$ to $-0.9$.

Table 4.7: Estimates of parameters and variance for AR(1) simulations with $\phi < 0$. Note that $\hat{\phi}_D$ and $\hat{\sigma}^2_D$ refer to estimates from the data.

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>$\hat{\phi}$</th>
<th>$\hat{\sigma}_2$</th>
<th>$\hat{\phi}_D$</th>
<th>$\hat{\sigma}^2$</th>
<th>$\hat{\sigma}^2_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>0.160</td>
<td>0.0960</td>
<td>-0.012</td>
<td>6189.6</td>
<td>676.5</td>
</tr>
<tr>
<td>-0.2</td>
<td>0.116</td>
<td>0.1006</td>
<td>-0.114</td>
<td>6614.7</td>
<td>818.4</td>
</tr>
<tr>
<td>-0.3</td>
<td>-0.073</td>
<td>0.0902</td>
<td>-0.220</td>
<td>6986.2</td>
<td>852.8</td>
</tr>
<tr>
<td>-0.4</td>
<td>-0.180</td>
<td>0.0874</td>
<td>-0.329</td>
<td>7191.8</td>
<td>905.2</td>
</tr>
<tr>
<td>-0.5</td>
<td>-0.298</td>
<td>0.0929</td>
<td>-0.441</td>
<td>7404.1</td>
<td>935.3</td>
</tr>
<tr>
<td>-0.6</td>
<td>-0.432</td>
<td>0.0743</td>
<td>-0.555</td>
<td>7228.6</td>
<td>802.9</td>
</tr>
<tr>
<td>-0.7</td>
<td>-0.562</td>
<td>0.0963</td>
<td>-0.670</td>
<td>7542.2</td>
<td>1075.5</td>
</tr>
<tr>
<td>-0.8</td>
<td>-0.691</td>
<td>0.0763</td>
<td>-0.779</td>
<td>8177.7</td>
<td>1033.7</td>
</tr>
<tr>
<td>-0.9</td>
<td>-0.838</td>
<td>0.0514</td>
<td>-0.881</td>
<td>8447.0</td>
<td>1032.4</td>
</tr>
</tbody>
</table>

Now looking at the results for MA(1) models with positive parameters, shown in Table 4.8, we see:

(i) the means of the parameter estimates from the surrogates are similar to the estimates from the data for $\theta = (0.1, 0.2, 0.3)$ but fall below thereafter. They agree with the nominal values up to $\theta = 0.6$;

(ii) the standard deviations of the parameter estimates from the surrogates lie between 0.039 and 0.058 and do not show any pattern;

(iii) the mean of the variance estimates from the surrogates are similar to the estimate from the data up to $\theta = 0.4$ and are larger thereafter, being 38% greater when $\theta = 0.9$;

(iv) the standard deviations of the variance estimates from the surrogates consistently lie around 400 except when $\theta = 0.1$ (469.8) and $\theta = 0.9$ (553.3).

The results for MA(1) models with negative parameters are shown in Table 4.9 and it appears that wavelet resampling has not produced surrogates that could have come from the data. In particular, it is noted that:

(i) the means of the parameter estimates from the surrogates are very different to the estimates from the data and the nominal value for all values of $\theta$; the mean estimates are positive when the nominal values of $\theta$ are $-0.3$ and below;
Table 4.8: Estimates of parameters and variance for MA(1) simulations with \( \theta > 0 \). Note that \( \hat{\theta}_D \) and \( \hat{\sigma}^2_D \) refer to estimates from the data.

<table>
<thead>
<tr>
<th>( \theta )</th>
<th>( \hat{\theta} )</th>
<th>( \sigma_\theta )</th>
<th>( \hat{\theta}_D )</th>
<th>( \hat{\sigma}^2 )</th>
<th>( \sigma_\sigma^2 )</th>
<th>( \hat{\sigma}^2_D )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.206</td>
<td>0.0580</td>
<td>0.182</td>
<td>5174.2</td>
<td>469.8</td>
<td>5215.4</td>
</tr>
<tr>
<td>0.2</td>
<td>0.277</td>
<td>0.0536</td>
<td>0.276</td>
<td>5159.5</td>
<td>426.6</td>
<td>5219.4</td>
</tr>
<tr>
<td>0.3</td>
<td>0.341</td>
<td>0.0496</td>
<td>0.367</td>
<td>5199.4</td>
<td>388.6</td>
<td>5225.1</td>
</tr>
<tr>
<td>0.4</td>
<td>0.396</td>
<td>0.0464</td>
<td>0.456</td>
<td>5303.0</td>
<td>358.3</td>
<td>5232.1</td>
</tr>
<tr>
<td>0.5</td>
<td>0.496</td>
<td>0.0515</td>
<td>0.545</td>
<td>5560.7</td>
<td>334.7</td>
<td>5239.4</td>
</tr>
<tr>
<td>0.6</td>
<td>0.566</td>
<td>0.0478</td>
<td>0.633</td>
<td>5736.4</td>
<td>376.0</td>
<td>5245.8</td>
</tr>
<tr>
<td>0.7</td>
<td>0.616</td>
<td>0.0461</td>
<td>0.721</td>
<td>6690.6</td>
<td>386.9</td>
<td>5252.2</td>
</tr>
<tr>
<td>0.8</td>
<td>0.644</td>
<td>0.0393</td>
<td>0.814</td>
<td>6034.0</td>
<td>397.5</td>
<td>5250.3</td>
</tr>
<tr>
<td>0.9</td>
<td>0.686</td>
<td>0.0515</td>
<td>0.906</td>
<td>7259.2</td>
<td>553.3</td>
<td>5252.4</td>
</tr>
</tbody>
</table>

Table 4.9: Estimates of parameters and variance for MA(1) simulations with \( \theta < 0 \). Note that \( \hat{\theta}_D \) and \( \hat{\sigma}^2_D \) refer to estimates from the data.

<table>
<thead>
<tr>
<th>( \theta )</th>
<th>( \hat{\theta} )</th>
<th>( \sigma_\theta )</th>
<th>( \hat{\theta}_D )</th>
<th>( \hat{\sigma}^2 )</th>
<th>( \sigma_\sigma^2 )</th>
<th>( \hat{\sigma}^2_D )</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>0.147</td>
<td>0.0815</td>
<td>-0.011</td>
<td>6205.9</td>
<td>694.6</td>
<td>5211.5</td>
</tr>
<tr>
<td>-0.2</td>
<td>0.104</td>
<td>0.0868</td>
<td>-0.111</td>
<td>6568.9</td>
<td>852.2</td>
<td>5210.6</td>
</tr>
<tr>
<td>-0.3</td>
<td>0.033</td>
<td>0.0905</td>
<td>-0.211</td>
<td>6782.5</td>
<td>894.9</td>
<td>5209.7</td>
</tr>
<tr>
<td>-0.4</td>
<td>-0.038</td>
<td>0.0939</td>
<td>-0.313</td>
<td>7051.6</td>
<td>947.1</td>
<td>5208.5</td>
</tr>
<tr>
<td>-0.5</td>
<td>-0.120</td>
<td>0.0821</td>
<td>-0.414</td>
<td>7189.4</td>
<td>919.0</td>
<td>5207.3</td>
</tr>
<tr>
<td>-0.6</td>
<td>-0.229</td>
<td>0.0764</td>
<td>-0.517</td>
<td>7090.6</td>
<td>755.2</td>
<td>5206.5</td>
</tr>
<tr>
<td>-0.7</td>
<td>-0.277</td>
<td>0.0733</td>
<td>-0.624</td>
<td>7524.3</td>
<td>840.5</td>
<td>5206.2</td>
</tr>
<tr>
<td>-0.8</td>
<td>-0.319</td>
<td>0.0767</td>
<td>-0.739</td>
<td>8077.8</td>
<td>901.7</td>
<td>5206.5</td>
</tr>
<tr>
<td>-0.9</td>
<td>-0.334</td>
<td>0.0684</td>
<td>-0.854</td>
<td>8527.6</td>
<td>775.5</td>
<td>5226.0</td>
</tr>
</tbody>
</table>

(ii) the standard deviations of the parameter estimates from the surrogates increase to a maximum of 0.094 when \( \theta = -0.4 \) and decline to 0.068 when \( \theta = -0.9 \);

(iii) the mean of the variance estimates from the surrogates increase from 6205.9 (\( \theta = -0.1 \)) to 8527.6 (\( \theta = -0.9 \));

(iv) the standard deviations of the variance estimates from the surrogates lie between 755 and 947.

4.3.1 Interim conclusion

We can conclude that the wavelet-based resampling method appears to have been successful in replicating AR(1) time series with positive parameters. The method has worked reasonably well with MA(1) models with positive parameters but disappointing results have been achieved with both AR(1) and MA(1) models with negative parameters, particularly the latter.

In an attempt to try to explain the different outcomes, a comparison was made of the bases used in the resampling process for the various parameter values. Recall that the basis in the resampling method consists of the wavelet packets in which the coefficients are considered to be uncorrelated.

Table 4.10 shows the bases for AR(1) models with \( \phi > 0 \). Both packets at the first level of de-
composition are considered to have uncorrelated coefficients when $\phi \leq 0.4$. Thereafter the second packet at Level 1 and the first two packets at Level 2 are judged to have uncorrelated coefficients. Thus AR(1) models with $\phi > 0$ decompose rapidly within the first two levels.

In contrast, Table 4.11 shows that AR(1) models with $\phi < 0$ take 5 or 6 levels to fully decompose. That is, correlation in the coefficients is not fully removed until the fifth or sixth level.

Similar outcomes are obtained for MA(1) models. Table 4.12 shows the results when $\theta > 0$ and we see that decomposition takes place fully at Level 1 when $\theta \leq 0.4$. The coefficients in the first packet in Level 1 and the last two packets in Level 2 are considered uncorrelated when $\theta = (0.5, 0.6, 0.7)$. However correlation is not fully removed until Level 6 when $\theta = 0.8$ or 0.9.

Finally, Table 4.13 shows that five or six levels are required to fully decompose MA(1) models with $\theta < 0$.

Figures 4.4 to 4.7 illustrate the bases for AR(1) and MA(1) models with parameters $\pm 0.9$.

Because the number of coefficients in a packet halves from one level to the next it is apparent that packets at the higher levels will have relatively few coefficients. For example, in the simulations here, the number of data values was 256, resulting in in only four coefficients in each Level 6 packet. If these coefficients differ widely in value then the resampled packet may be influential in the surrogate series.

Some indication of the impact of the sample size can be obtained with using a simulated series with substantially more values. The results for AR(1) and MA(1) models with parameter $-0.9$ and $n = 2048$ observations are shown in Table 4.14. The increase in sample size has improved the results for the AR(1) model but not the MA(1) model.

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>Level 1</th>
<th>Level 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0,1</td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>0,1</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0,1</td>
<td></td>
</tr>
<tr>
<td>0.4</td>
<td>0,1</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>0,1</td>
</tr>
<tr>
<td>0.6</td>
<td>1</td>
<td>0,1</td>
</tr>
<tr>
<td>0.7</td>
<td>1</td>
<td>0,1</td>
</tr>
<tr>
<td>0.8</td>
<td>1</td>
<td>0,1</td>
</tr>
<tr>
<td>0.9</td>
<td>1</td>
<td>0,1</td>
</tr>
</tbody>
</table>

Table 4.10: Basis for AR(1) simulations with $\phi > 0$
### Table 4.11: Basis for AR(1) simulations with $\phi < 0$

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
<th>Level 5</th>
<th>Level 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>1</td>
<td>-</td>
<td>3</td>
<td>0,3,4,5</td>
<td>2,3,4,5</td>
<td></td>
</tr>
<tr>
<td>-0.2</td>
<td>1</td>
<td>-</td>
<td>3</td>
<td>0,3,4,5</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.3</td>
<td>-</td>
<td>3</td>
<td>3,4,5</td>
<td>0,3,4,5</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.4</td>
<td>-</td>
<td>3</td>
<td>3,4,5</td>
<td>0,3,4,5</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.5</td>
<td>-</td>
<td>3</td>
<td>2,3,4,5</td>
<td>0,3</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.6</td>
<td>-</td>
<td>3</td>
<td>2,3,4,5</td>
<td>0,1</td>
<td>4,5</td>
<td></td>
</tr>
<tr>
<td>-0.7</td>
<td>-</td>
<td>3</td>
<td>2,3,5</td>
<td>0,1,3,8,9</td>
<td>4,5</td>
<td></td>
</tr>
<tr>
<td>-0.8</td>
<td>-</td>
<td>1,3</td>
<td>-</td>
<td>0,1,3,8,9</td>
<td>4,5</td>
<td></td>
</tr>
<tr>
<td>-0.9</td>
<td>-</td>
<td>1,3</td>
<td>5</td>
<td>0,1,2,3,8,9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 4.12: Basis for MA(1) simulations with $\theta > 0$

<table>
<thead>
<tr>
<th>$\theta$</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
<th>Level 5</th>
<th>Level 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0,1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>0,1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0,1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.4</td>
<td>0,1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0</td>
<td>2,3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td>0</td>
<td>2,3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0</td>
<td>2,3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>11</td>
<td>21</td>
<td>40,41</td>
</tr>
<tr>
<td>0.9</td>
<td>0</td>
<td>3</td>
<td>-</td>
<td>8,9,11</td>
<td>21</td>
<td>40,41</td>
</tr>
</tbody>
</table>

### Table 4.13: Basis for MA(1) simulations with $\theta < 0$

<table>
<thead>
<tr>
<th>$\theta$</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
<th>Level 5</th>
<th>Level 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>1</td>
<td>-</td>
<td>3</td>
<td>0,3,4,5</td>
<td>2,3,4,5</td>
<td></td>
</tr>
<tr>
<td>-0.2</td>
<td>1</td>
<td>-</td>
<td>3</td>
<td>0,3,4,5</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.3</td>
<td>1</td>
<td>-</td>
<td>3</td>
<td>0,3,4,5</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.4</td>
<td>1</td>
<td>-</td>
<td>3</td>
<td>0,3,4,5</td>
<td>2,4,5</td>
<td>6,7</td>
</tr>
<tr>
<td>-0.5</td>
<td>1</td>
<td>-</td>
<td>2,3</td>
<td>0,1,3</td>
<td>4,5</td>
<td></td>
</tr>
<tr>
<td>-0.6</td>
<td>-</td>
<td>2,3</td>
<td>2,3</td>
<td>0,1,3</td>
<td>4,5</td>
<td></td>
</tr>
<tr>
<td>-0.7</td>
<td>-</td>
<td>2,3</td>
<td>2,3</td>
<td>0,1,2,3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.8</td>
<td>-</td>
<td>2,3</td>
<td>2,3</td>
<td>0,1,2,3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.9</td>
<td>-</td>
<td>2,3</td>
<td>1,2,3</td>
<td>0,1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 4.14: Simulations with parameter $-0.9$ and $n = 2048$ compared with $n = 256$

<table>
<thead>
<tr>
<th>Variable</th>
<th>$n$</th>
<th>AR(1)</th>
<th>AR(1)</th>
<th>MA(1)</th>
<th>MA(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter estimate</td>
<td>256</td>
<td>-0.838</td>
<td>-0.859</td>
<td>-0.334</td>
<td>-0.393</td>
</tr>
<tr>
<td>Std. dev. of estimate</td>
<td>2048</td>
<td>0.0514</td>
<td>0.0149</td>
<td>0.0684</td>
<td>0.0287</td>
</tr>
<tr>
<td>Variance estimate</td>
<td>8447.0</td>
<td>936.6</td>
<td>8527.6</td>
<td>8949.7</td>
<td></td>
</tr>
<tr>
<td>Std. dev. of estimate</td>
<td>1032.4</td>
<td>316.9</td>
<td>775.5</td>
<td>345.6</td>
<td></td>
</tr>
</tbody>
</table>
4.3. PARAMETER AND VARIANCE ESTIMATES

Figure 4.4: Basis for AR(1) model with $\phi = 0.9$

Figure 4.5: Basis for AR(1) model with $\phi = -0.9$

Figure 4.6: Basis for MA(1) model with $\theta = 0.9$

Figure 4.7: Basis for MA(1) model with $\theta = -0.9$
4.4 A closer look at the surrogates

The results for the AR(1) models with $\phi = \pm 0.5$ and MA(1) models with $\theta = \pm 0.5$ are now examined in more detail. The following plots are shown for each model:

- Line plot of the data and one surrogate, split into two plots for clarity
- Line plot of the data and mean of the surrogates, also split into two plots
- Line plots for a selection of eight surrogates
- Autocorrelation function (ACF) for the data and three surrogates
- Power spectrum for the data and three surrogates

4.4.1 AR(1) model with $\phi = 0.5$

Figure 4.8 shows the data and one surrogate. There is nothing to indicate that this surrogate did not come from the same model that produced the data. The mean of 1000 surrogates is shown against the data in Figure 4.9 and it is apparent that the mean closely follows the data.

Line plots for a selection of eight surrogates are shown in Figure 4.10 and these are similar to the data.

Figure 4.11 shows the ACF for the data and three surrogates. These plots show that the surrogates share the correlation structure of the data. The spectra for the same surrogates are shown in Figure 4.12 and are consistent with the spectrum from the data.

All of the plots indicate that the surrogates could be replicates from the model that produced the data, confirming the conclusion from the comparison of the estimates of the parameter and variance for AR(1) models with $\phi > 0$.

Figure 4.8: AR(1) with $\phi = 0.5$ : data and one surrogate series (a) observations 1 - 128 and (b) observations 129 - 256
4.4. A CLOSER LOOK AT THE SURROGATES

Figure 4.9: AR(1) with $\phi = 0.5$: data and mean of 1000 surrogates (a) observations 1 - 128 and (b) observations 129 - 256

Figure 4.10: AR(1) with $\phi = 0.5$: a selection of surrogates
Figure 4.11: AR(1) with $\phi = 0.5$: autocorrelation function (ACF) for data and three surrogates.

Figure 4.12: AR(1) with $\phi = 0.5$: spectrum for data and three surrogates.
4.4. A CLOSER LOOK AT THE SURROGATES

4.4.2 AR(1) model with $\phi = -0.5$

Figure 4.13 shows the data and one surrogate. This surrogate exhibits more variation than the data, indicating that it may not be a replicate from the same model that produced the data. The mean of 1000 surrogates is shown against the data in Figure 4.14. It appears that the mean generally follows the data with aberrations around index values 5, 120, 155 and 210.

Line plots for a selection of eight surrogates are shown in Figure 4.15 and do not appear to be similar to the data.

Figure 4.16 shows the ACF for the data and three surrogates. These plots show that the surrogates exhibit some variation in correlation structure compared to the data. For example, the magnitude of the autocorrelation coefficient at lag 1 for the surrogates is less than that for the data.

In Figure 4.17 the power spectrum for the data shows that values of $\hat{f}(w)$ are consistently low for $w < 1.5$ with large values occurring when $w > 2.5$. In contrast, the spectra for the three surrogates have moderately high values of $\hat{f}(w)$ for $w < 2.5$.

All of the plots indicate that the surrogates exhibit variations from what could be expected of replicates from the model that produced the data, confirming the conclusion from the comparison of the estimates of the parameter and variance for AR(1) models with $\phi < 0$. 
Figure 4.14: AR(1) with $\phi = -0.5$ : data and mean of 1000 surrogates (a) observations 1 - 128 and (b) observations 129 - 256

Figure 4.15: AR(1) with $\phi = -0.5$ : a selection of surrogates
4.4. A CLOSER LOOK AT THE SURROGATES

Figure 4.16: AR(1) with $\phi = -0.5$: autocorrelation function (ACF) for data and three surrogates

Figure 4.17: AR(1) with $\phi = -0.5$: spectrum for data and three surrogates
4.4.3 MA(1) model with $\theta = 0.5$

Figure 4.18 shows the data and one surrogate. This surrogate is similar to the data, indicating that it may be a replicate from the same model that produced the data. The mean of 1000 surrogates is shown against the data in Figure 4.19 where it is apparent that the mean closely follows the data.

Line plots for a selection of eight surrogates are shown in Figure 4.20 and appear to be similar to the data.

Figure 4.21 shows the ACF for the data and three surrogates. These plots show that the surrogates have correlation structures that are very similar to that of the data.

In Figure 4.22 the power spectra for the data and the surrogates are similar.

All of the plots indicate that the surrogates exhibit similar behaviour to what could be expected of replicates from the model that produced the data, confirming the conclusion from the comparison of the estimates of the parameter and variance for MA(1) models with $\theta > 0$.

![Figure 4.18: MA(1) with $\theta = 0.5$ : data and one surrogate series (a) observations 1 - 128 and (b) observations 129 - 256](image1)

![Figure 4.19: MA(1) with $\theta = 0.5$ : data and mean of 1000 surrogates (a) observations 1 - 128 and (b) observations 129 - 256](image2)
4.4. A CLOSER LOOK AT THE SURROGATES

Figure 4.20: MA(1) with $\theta = 0.5$: a selection of surrogates

Figure 4.21: MA(1) with $\theta = 0.5$: autocorrelation function (ACF) for data and three surrogates
4.4.4 MA(1) model with $\theta = -0.5$

Figure 4.23 shows the data and one surrogate. This surrogate exhibits more extreme values than the data, indicating that it may not be a replicate from the same model that produced the data. The mean of 1000 surrogates is shown against the data in Figure 4.24 where it is appears that the mean approximates a moving average of the data.

Line plots for a selection of eight surrogates are shown in Figure 4.25 and appear to be quite different to each other and the data.

Figure 4.26 shows the ACF for the data and three surrogates. These plots show that the surrogates have correlation structures that are different to that of the data. While surrogates #1 and #500 have a significant autocorrelation coefficient at lag 1, it is weaker than that calculated for the data. Surrogate #100 has no significant autocorrelation coefficient at any lag. Of these three surrogates, only surrogate #500 appears similar to the data.

In Figure 4.27 the power spectra for the data and the surrogates confirm the observations from the ACF. Surrogate #500 has a spectrum closest to that of the data while surrogate #100 has high values of $\hat{f}(w)$ for all values of $w$. Note the differing vertical scales.

All of the plots indicate that the surrogates exhibit different behaviour to what could be expected of replicates from the model that produced the data, confirming the conclusion from the comparison of the estimates of the parameter and variance for MA(1) models with $\theta < 0$. 
4.4. A CLOSER LOOK AT THE SURROGATES

Figure 4.23: MA(1) with $\theta = -0.5$: data and one surrogate series (a) observations 1 - 128 and (b) observations 129 - 256

Figure 4.24: MA(1) with $\theta = -0.5$: data and mean of 1000 surrogates (a) observations 1 - 128 and (b) observations 129 - 256
Figure 4.25: MA(1) with $\theta = -0.5$ : a selection of surrogates

Figure 4.26: MA(1) with $\theta = -0.5$ : autocorrelation function (ACF) for data and three surrogates
4.5 A closer look at the white noise test

The test for white noise is used to classify whether the coefficients in the wavelet packets are correlated. The standard portmanteau test available in Mathematica’s Time Series package [13] is the Ljung-Box formulation:

\[ Q_{LB} = n(n + 2) \sum_{i=1}^{m} \frac{\hat{\rho}^2(i)}{n-i} \]  \hspace{1cm} (4.10)

where \( \hat{\rho}^2(i) \) is the sample autocorrelation at lag \( i \), \( n \) is the sample size and \( m \) is the maximum number of lags included. This test statistic is a version of the original Box-Pierce formulation:

\[ Q_{BP} = n \sum_{i=1}^{m} \hat{\rho}^2(i) \]  \hspace{1cm} (4.11)

Both test statistics are compared with \( \chi^2_{m-p-q} \) at significance level \( 1 - \alpha \), where \( p \) and \( q \) are the number of AR and MA parameters in the models. Here \( (p, q) = (1, 0) \) and \( (0, 1) \) for the AR and MA models respectively.

Arranz [2] states that the Box-Pierce statistic departs from the \( \chi^2 \) distribution for finite samples and the Ljung-Box statistic is a better approximation in such cases. He also states that a criticism of \( Q_{LB} \) is that ‘its variance could be substantially larger than its asymptotic distribution’.

Figure 4.27: MA(1) with \( \theta = -0.5 \) : spectrum for data and three surrogates
A further modification of $Q_{BP}$ is the Li-McLeod statistic:

$$Q_{LM} = \frac{m(m+1)}{2n} + n \sum_{i=1}^{m} \hat{\rho}^2(i)$$  \hspace{1cm} \text{(4.12)}$$

which moves the distribution of $Q_{BP}$ towards its asymptotic distribution ($\chi^2_{m-p-q}$) without its variance being inflated.

### 4.5.1 AR(1) model with $\phi = -0.5$

The simulated data were decomposed to Level 6 above before it was determined that all autocorrelation was removed. This analysis is restricted to the first three levels.

Table 4.15 shows the values of $Q_{LB}$, $Q_{BP}$ and $Q_{LM}$ for the coefficients at Levels 1 and 2 and the first 6 packets of Level 3, together with the critical $\chi^2$ value at $\alpha = 0.05$. The number of lags included in the calculation equals half the number of coefficients in the packet. In a number of cases $Q_{LM}$ is approximately mid-way between $Q_{BP}$ and $Q_{LB}$. The corresponding autocorrelation plots are shown in Figures 4.28 and 4.29.

Judging purely by these Figures, we could conclude that the coefficients in packets W(1, 0), W(1, 1) and W(2, 1) appear to be correlated while the coefficients in packets W(2, 2), W(2, 3) and W(3, 2) to W(3, 5) are not correlated. The coefficients in the remaining packets - W(2, 0), W(3, 0) and W(3, 1) - may or may not be correlated.

Now referring to Table 4.15, we see that all the test statistics lead to a conclusion that packets W(1, 0), W(1, 1) and W(2, 1) have correlated coefficients while packets W(2, 3) and W(3, 2) to W(3, 5) have uncorrelated coefficients.

The coefficients in W(2, 0) are uncorrelated according to Box-Pierce (resulting in W(2, 0) being in the basis and not decomposed further) but correlated by the other two tests which also classify W(3, 0) and W(3, 1) as retaining correlation.

<table>
<thead>
<tr>
<th>Level</th>
<th>Packet</th>
<th>Ljung-Box</th>
<th>Box-Pierce</th>
<th>Li-McLeod</th>
<th>Critical $\chi^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>105.4</td>
<td>86.0</td>
<td>102.7</td>
<td>82.5</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>199.5</td>
<td>157.8</td>
<td>174.5</td>
<td>82.5</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>58.5</td>
<td>42.0</td>
<td>50.8</td>
<td>45.0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>65.7</td>
<td>46.9</td>
<td>55.7</td>
<td>45.0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>51.9</td>
<td>38.9</td>
<td>47.6</td>
<td>45.0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>24.4</td>
<td>16.8</td>
<td>25.6</td>
<td>45.0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>51.1</td>
<td>34.9</td>
<td>39.7</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>34.6</td>
<td>23.5</td>
<td>28.3</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>22.9</td>
<td>14.4</td>
<td>19.2</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>11.6</td>
<td>8.4</td>
<td>13.1</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>24.1</td>
<td>16.7</td>
<td>21.5</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>11.8</td>
<td>8.5</td>
<td>13.3</td>
<td>25.0</td>
</tr>
</tbody>
</table>
4.5. A CLOSER LOOK AT THE WHITE NOISE TEST

Figure 4.28: AR(1) with $\phi = -0.5$ : autocorrelation coefficients for packets at Levels 1 and 2

Figure 4.29: AR(1) with $\phi = -0.5$ : autocorrelation coefficients for packets at Level 3
4.5.2 MA(1) model with $\theta = -0.5$

The simulated data was decomposed to Level 5 above before it was determined that all autocorrelation was removed. This analysis is again restricted to the first three levels.

Table 4.16 shows the values of $Q_{LB}$, $Q_{BP}$ and $Q_{LM}$ for the coefficients at Levels 1, the first two packets of Level 2 and the first 4 packets of Level 3, together with the critical $\chi^2$ value at $\alpha = 0.05$. The number of lags included in these calculation equals half the number of coefficients in the packet. At Level 1 this meant that 128 lags were used. Figure 4.30 shows that $W(1, 1)$ exhibits relatively large correlations around lags 10 and 20 which appear to unduly influence the test statistic. Only the first seven lags were included in the test statistic for packets at Level 1 in the final version of the lifting procedure, resulting in packet (1, 1) being classified as having uncorrelated coefficients.

Again $Q_{LM}$ is often approximately mid-way between $Q_{BP}$ and $Q_{LB}$. The corresponding autocorrelation plots are shown in Figures 4.30 and 4.31.

Table 4.16: White noise test statistics: MA(1) with $\theta = -0.5$

<table>
<thead>
<tr>
<th>Level</th>
<th>Packet</th>
<th>Ljung-Box</th>
<th>Box-Pierce</th>
<th>Li-McLeod</th>
<th>Critical $\chi^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>166.4</td>
<td>140.9</td>
<td>157.6</td>
<td>82.5</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>115.0</td>
<td>87.6</td>
<td>104.3</td>
<td>82.5</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>67.9</td>
<td>49.4</td>
<td>58.2</td>
<td>45.0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>74.9</td>
<td>53.9</td>
<td>62.6</td>
<td>45.0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>57.2</td>
<td>39.2</td>
<td>44.0</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>33.2</td>
<td>22.4</td>
<td>27.2</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>23.9</td>
<td>15.2</td>
<td>20.0</td>
<td>25.0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>11.2</td>
<td>8.4</td>
<td>13.2</td>
<td>25.0</td>
</tr>
</tbody>
</table>
4.5. A CLOSER LOOK AT THE WHITE NOISE TEST

4.5.3 Results using the Box-Pierce test statistic

The resampling procedure was re-run using the Box-Pierce test statistic to determine the presence of autocorrelation.

Tables 4.17 and 4.18 show the packets in the basis for the decomposition of the AR(1) and MA(1) models with negative parameters. A comparison with Tables 4.11 and 4.13 shows that decomposition occurs faster with Box-Pierce test statistic. By Level 3, according to the test statistic, all of the autocorrelation has been removed from the AR(1) models and most of the autocorrelation removed from the MA(1) models.

Tables 4.19 and 4.20 show the corresponding estimates of the parameters and variances. Comparing Table 4.19 with Table 4.7 we see that, for AR(1) models, use of the Box-Pierce test statistic has resulted in:

(i) better values for $\hat{\phi}$ when $\phi > -0.6$,

(ii) more consistent values of $\sigma^2_{\hat{\phi}}$ throughout,

(iii) lower values of $\hat{\sigma}^2$ throughout but particularly when $\phi > -0.6$,

(iv) lower and more consistent values of $\sigma^2_{\hat{\sigma}^2}$ throughout.

Hence it appears that, for AR(1) models, the use of wavelet packets at an earlier level of decomposition has resulted in better estimators of the parameter and variance.

However, comparing Table 4.20 with Table 4.8, it appears that mixed results occur for MA(1) models. While there is some improvement for models with $\theta > -0.6$ there is a deterioration for models with $\theta < -0.6$. In the latter case the estimates of $\hat{\theta}$ and $\hat{\sigma}^2$ remain poor.
Table 4.17: Basis for AR(1) simulations with $\phi < 0$ using the Box-Pierce test statistic

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>1</td>
<td>0</td>
<td>2,3</td>
</tr>
<tr>
<td>-0.2</td>
<td>1</td>
<td>0</td>
<td>2,3</td>
</tr>
<tr>
<td>-0.3</td>
<td>-</td>
<td>0.2,3</td>
<td>2,3</td>
</tr>
<tr>
<td>-0.4</td>
<td>-</td>
<td>0.2,3</td>
<td>2,3</td>
</tr>
<tr>
<td>-0.5</td>
<td>-</td>
<td>0.2,3</td>
<td>2,3</td>
</tr>
<tr>
<td>-0.6</td>
<td>-</td>
<td>0,1,2,3</td>
<td>-</td>
</tr>
<tr>
<td>-0.7</td>
<td>-</td>
<td>0,1,3</td>
<td>4,5</td>
</tr>
<tr>
<td>-0.8</td>
<td>-</td>
<td>0,1,3</td>
<td>4,5</td>
</tr>
<tr>
<td>-0.9</td>
<td>-</td>
<td>0,1,3</td>
<td>4,5</td>
</tr>
</tbody>
</table>

Table 4.18: Basis for MA(1) simulations with $\theta < 0$ using the Box-Pierce test statistic

<table>
<thead>
<tr>
<th>$\theta$</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>0</td>
<td>0</td>
<td>2,3</td>
<td>-</td>
</tr>
<tr>
<td>-0.2</td>
<td>0</td>
<td>0</td>
<td>2,3</td>
<td>-</td>
</tr>
<tr>
<td>-0.3</td>
<td>0</td>
<td>0</td>
<td>2,3</td>
<td>-</td>
</tr>
<tr>
<td>-0.4</td>
<td>0</td>
<td>-</td>
<td>1,2,3</td>
<td>0,1</td>
</tr>
<tr>
<td>-0.5</td>
<td>0</td>
<td>-</td>
<td>1,2,3</td>
<td>0,1</td>
</tr>
<tr>
<td>-0.6</td>
<td>-</td>
<td>2,3</td>
<td>1,2,3</td>
<td>0,1</td>
</tr>
<tr>
<td>-0.7</td>
<td>-</td>
<td>1,2,3</td>
<td>1</td>
<td>0,1</td>
</tr>
<tr>
<td>-0.8</td>
<td>-</td>
<td>1,2,3</td>
<td>1</td>
<td>0,1</td>
</tr>
<tr>
<td>-0.9</td>
<td>-</td>
<td>1,2,3</td>
<td>1</td>
<td>0,1</td>
</tr>
</tbody>
</table>

4.6 Final conclusion

The closer examination of series derived from models with parameters $\pm 0.5$ has confirmed the earlier conclusion that the wavelet resampling method, using linear lifting, is effective in producing surrogate series only when the parameter is positive. Line plots, autocorrelation functions and power spectra have shown that the surrogates share the main characteristics of the data. However, when the parameter is negative, the surrogates do not share the characteristics of the data, particularly in the case of MA(1) models. As has been shown, better parameter estimates are obtained when the wavelet packets are deemed to be uncorrelated at an earlier stage of decomposition. When correlation is carried through to later stages of decomposition there are fewer

Table 4.19: AR(1) simulations with $\phi < 0$: estimates of parameters and variance using the Box-Pierce test statistic to determine the presence of autocorrelation in the wavelet packets

<table>
<thead>
<tr>
<th>$\phi$</th>
<th>$\hat{\phi}$</th>
<th>$\hat{\sigma}_\phi$</th>
<th>$\hat{\phi}_D$</th>
<th>$\hat{\sigma}^2$</th>
<th>$\hat{\sigma}_{\phi^2}$</th>
<th>$\hat{\sigma}_{\phi^2}^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>0.071</td>
<td>0.0545</td>
<td>-0.012</td>
<td>5541.0</td>
<td>572.7</td>
<td>5213.0</td>
</tr>
<tr>
<td>-0.2</td>
<td>-0.005</td>
<td>0.0543</td>
<td>-0.114</td>
<td>5735.1</td>
<td>613.2</td>
<td>5215.3</td>
</tr>
<tr>
<td>-0.3</td>
<td>-0.142</td>
<td>0.0621</td>
<td>-0.220</td>
<td>5576.4</td>
<td>491.5</td>
<td>5219.1</td>
</tr>
<tr>
<td>-0.4</td>
<td>-0.241</td>
<td>0.0634</td>
<td>-0.329</td>
<td>5691.4</td>
<td>508.5</td>
<td>5224.8</td>
</tr>
<tr>
<td>-0.5</td>
<td>-0.344</td>
<td>0.0631</td>
<td>-0.441</td>
<td>5849.0</td>
<td>531.5</td>
<td>5232.0</td>
</tr>
<tr>
<td>-0.6</td>
<td>-0.423</td>
<td>0.0656</td>
<td>-0.555</td>
<td>6416.3</td>
<td>601.0</td>
<td>5239.7</td>
</tr>
<tr>
<td>-0.7</td>
<td>-0.594</td>
<td>0.0580</td>
<td>-0.670</td>
<td>6779.2</td>
<td>664.4</td>
<td>5246.1</td>
</tr>
<tr>
<td>-0.8</td>
<td>-0.702</td>
<td>0.0483</td>
<td>-0.779</td>
<td>6976.8</td>
<td>678.9</td>
<td>5249.5</td>
</tr>
<tr>
<td>-0.9</td>
<td>-0.812</td>
<td>0.0383</td>
<td>-0.881</td>
<td>7412.4</td>
<td>705.6</td>
<td>5258.6</td>
</tr>
</tbody>
</table>
4.6. FINAL CONCLUSION

Table 4.20: MA(1) simulations with $\theta < 0$: estimates of parameters and variance using the Box-Pierce test statistic to determine the presence of autocorrelation in the wavelet packets

<table>
<thead>
<tr>
<th>$\theta$</th>
<th>$\hat{\theta}$</th>
<th>$\sigma_A$</th>
<th>$\hat{\theta}_D$</th>
<th>$\hat{\sigma}^2$</th>
<th>$\hat{\sigma}_D^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>0.042</td>
<td>0.0535</td>
<td>-0.011</td>
<td>5437.5</td>
<td>462.2</td>
</tr>
<tr>
<td>-0.2</td>
<td>-0.037</td>
<td>0.0527</td>
<td>-0.111</td>
<td>5511.1</td>
<td>484.0</td>
</tr>
<tr>
<td>-0.3</td>
<td>-0.115</td>
<td>0.0522</td>
<td>-0.211</td>
<td>5625.4</td>
<td>509.8</td>
</tr>
<tr>
<td>-0.4</td>
<td>-0.101</td>
<td>0.0654</td>
<td>-0.313</td>
<td>6334.5</td>
<td>674.3</td>
</tr>
<tr>
<td>-0.5</td>
<td>-0.160</td>
<td>0.0664</td>
<td>-0.414</td>
<td>6621.0</td>
<td>711.4</td>
</tr>
<tr>
<td>-0.6</td>
<td>-0.214</td>
<td>0.0683</td>
<td>-0.517</td>
<td>6984.3</td>
<td>756.7</td>
</tr>
<tr>
<td>-0.7</td>
<td>-0.209</td>
<td>0.0715</td>
<td>-0.624</td>
<td>7958.7</td>
<td>820.1</td>
</tr>
<tr>
<td>-0.8</td>
<td>-0.243</td>
<td>0.0752</td>
<td>-0.739</td>
<td>8600.2</td>
<td>893.8</td>
</tr>
<tr>
<td>-0.9</td>
<td>-0.268</td>
<td>0.0788</td>
<td>-0.854</td>
<td>9369.1</td>
<td>978.7</td>
</tr>
</tbody>
</table>

coefficients to resample. Because of this, and the necessity of inverting the transform through more stages, the surrogates are less like the data than when correlation is removed earlier. Tables 4.10 - 4.13 show that decomposition is carried to much higher levels when the parameter is negative.

Some improvement in the results for surrogates from models with negative parameters was achieved by resampling from wavelet packets at an earlier stage of decomposition. This improvement occurred for AR(1) models and MA(1) models with $\theta > -0.6$ while a deterioration was observed for MA(1) models with $\theta < -0.6$. 
Chapter 5

Wavelet Resampling of 1D Point Patterns

Point processes concern events that occur in a continuous domain. An example in one dimensional space is earthquake occurrence times at a specified location.

5.1 One-dimensional point processes

A counting process is a one-dimensional point process that counts the number of events that have occurred up to time $t$. Data for such processes usually provide the time of an event, as measured from some specified origin, from which the number of events in a given time interval can be ascertained.

A counting process has:

1. *independent increments* if the number of events that occur in separate time intervals are independent, and

2. *stationary increments* if the distribution of the number of events in an interval of time is determined solely by the length of the interval.

The Poisson process, an example of a counting process that is frequently used to model event-time data, has the following properties:

1. the number of events at time $t=0$ is 0;

2. independent and stationary increments;

3. the number of events in an interval of length $t$ has a Poisson distribution with mean $\lambda t$ where $\lambda$ is the rate or intensity of the process.
5.1. ONE-DIMENSIONAL POINT PROCESSES

An alternative definition of the Poisson process comes from considering the times between events. If the interarrival times are independent, identically distributed exponential random variables with mean $\frac{1}{\lambda}$ then the number of events up to time $t$ is a Poisson process with rate $\lambda$.

If the stationary increment assumption is relaxed then it is possible to consider Poisson processes where the rate is no longer constant but is a function of time, $\lambda(t)$. Such a process is termed a nonhomogeneous Poisson process and allows modelling of events that occur with increasing or decreasing frequency. Processes that have clusters of events can be modelled by superimposing two or more homogeneous Poisson processes.

Data from a one-dimensional point pattern can be considered as a type of time series where the observations are a non-decreasing set of times rather than values of a random variable at regular time intervals. Therefore the lifting scheme can be employed to eliminate any autocorrelation in order to produce surrogate series that are possible realisations from the unknown model underlying the data.

Surrogates are produced in the following manner: perform the wavelet lifting transform, resample the crystals deemed to be independent and then invert the transform. With 1D-point processes an extra step of sorting the inverted series needs to occur to ensure a non-decreasing set of event times. Finally, all times are assumed to occur on a $(0, 1)$ interval and any resampled times that fell outside were brought back inside using the reflection boundary condition.

Three types of 1-D point processes will be considered:

1. homogeneous Poisson ie where $\lambda$ is constant;
2. non-homogeneous Poisson where $\lambda$ is a simple function of $t$;
3. clustered observations where the constant intensity over one section is lower than the constant intensity over the remainder.

Note that regular point processes, where there is inhibition between points, were not studied.

5.1.1 Simulated Data

(i) Homogeneous and non-homogeneous point processes

Data for these processes were simulated using the function pp.sim in the R package PtProcess [31]. Part of the description of the method employed in this function, provided in the package documentation, is:

'These functions use the thinning method (Ogata, 1981) [23] to simulate a point process with specified conditional intensity function. The method involves calculating an upper bound for the intensity function, simulating a value for the time to the next event using a rate equal to this upper bound, and then calculating the intensity at this simulated point. The ratio of this rate with the upper bound is compared with a uniform random number to randomly determine
whether the simulated time is retained or not.

The input used to simulate an homogeneous process was:

\[
d = \text{pp.sim(data=NULL, params=c(4,0,0), simple.cif, TT=c(0,50), output=TRUE, seed=5, magn.sim=1, stopping.condition=NULL, max.rate=4)}
\]

while the corresponding input for a non-homogeneous process was:

\[
d = \text{pp.sim(data=NULL, params=c(0,1,1), simple.cif, TT=c(0,20), output=TRUE, seed=5, magn.sim=1, stopping.condition=NULL, max.rate=21)}
\]

These simulations use the simple conditional intensity function option \textit{simple.cif} in PtProcess with the intensity given by \( \lambda(t) = a + bt^g \) and the vector of parameters, denoted \textit{params} in the function arguments, \((a, b, g)\). Although it is called a conditional intensity function it is actually not conditional on the history of the process.

The intensity function was \( \lambda(t) = 4 \) for the homogeneous case and \( \lambda(t) = t \) for the non-homogeneous case.

The variable TT defines a vector of length 2 denoting the required time interval of the simulation ie 50 periods for the homogeneous process and 20 periods for the non-homogeneous process.

(ii) Clustered data

Clustered data were simulated by generating random numbers from exponential distributions with different parameters. These random numbers represented the interarrival times and were accumulated to produce event times.

Specifically, fifty random numbers were generated from an exponential distribution with parameter 1, another fifty from an exponential distribution with parameter 3 and a final fifty again from an exponential distribution with parameter 1.

5.2 Resampling an homogeneous pattern

The simulation of an homogeneous process with intensity \( \lambda = 4 \) produced 146 events in 50 time periods ie an average of 2.92 events per period. The time interval was re-scaled to a \((0, 1)\) interval by dividing each event time by the ceiling of the last event time. Figure 5.1(a) shows the plot of the cumulative number of events by time for both the data and the mean of 1000 surrogates. The constant intensity of the simulation is shown by the approximate linearity of the plot. Clearly the
mean of the 1000 surrogates closely follows the data.

A number of plots can be generated to assess how well the surrogate series replicate the data:

![Graphs showing cumulative number of events by time, number of events within consecutive time periods, and number of events by event number for indicated radius for one surrogate.](image)

Figure 5.1: Homogeneous process with intensity $\lambda = 4$: (a) Cumulative number of events by time (b) Number of events within consecutive time periods of length 0.05 (c) Number of events by event number for indicated radius for one surrogate

(i) bin counts showing the number of events within consecutive time periods. Figure 5.1(b) shows the data bin counts for time periods of length 0.05 together with the mean bin count for the surrogates. The time interval of 0.05 corresponds to $0.05 \times 50 = 2.5$ time periods in the original simulation and therefore the average number of events per 0.05 time period is $2.5 \times 2.92 = 7.30$. After time 0.2 the mean surrogate bin count is relatively constant; the mean of the mean bin count for time periods 4 - 15 (inclusive) is 6.95. The plot illustrates
Table 5.1: Homogeneous process: time of occurrence of data events

<table>
<thead>
<tr>
<th>Event No.</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
<th>110</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>0.20</td>
<td>0.28</td>
<td>0.36</td>
<td>0.41</td>
<td>0.52</td>
<td>0.59</td>
<td>0.63</td>
<td>0.72</td>
</tr>
</tbody>
</table>

Table 5.2: Homogeneous process: number of events by time period for data and surrogates. The subscript $s$ refers to surrogates.

<table>
<thead>
<tr>
<th>Period</th>
<th>Actual</th>
<th>$\bar{x}_s$</th>
<th>$\sigma_s$</th>
<th>Period</th>
<th>Actual</th>
<th>$\bar{x}_s$</th>
<th>$\sigma_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>10</td>
<td>8.386</td>
<td>2.730</td>
<td>10</td>
<td>5</td>
<td>6.658</td>
<td>1.985</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>6.782</td>
<td>1.936</td>
<td>11</td>
<td>6</td>
<td>6.942</td>
<td>1.948</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>6.834</td>
<td>2.007</td>
<td>12</td>
<td>8</td>
<td>6.661</td>
<td>1.921</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>7.009</td>
<td>2.148</td>
<td>13</td>
<td>12</td>
<td>6.960</td>
<td>2.091</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>6.786</td>
<td>2.172</td>
<td>14</td>
<td>5</td>
<td>6.582</td>
<td>1.933</td>
</tr>
<tr>
<td>9</td>
<td>6</td>
<td>7.276</td>
<td>2.113</td>
<td>15</td>
<td>5</td>
<td>6.477</td>
<td>1.871</td>
</tr>
</tbody>
</table>

the variability in the bin counts for the data with two values outside the ± 2 std dev limits. The wide separation of these limits indicates that there is corresponding variability in the surrogate bin counts.

(ii) the number of events within fixed radius for the middle 80% of events. Figure 5.1(c) shows the number of events for radii 0.05, 0.10, 0.15 and 0.20 for the 15th to the 132nd event (inclusive) of the first surrogate together with the corresponding data plot. The surrogate series exhibits generally similar patterns for all shown radii.

(iii) the number of events within increasing radii centred on specific events. For example, Figure 5.2 shows such plots for eight events. The time of occurrences for the events shown in the plots are given in table 5.1. The plot for data values is shown against the mean of all surrogates which is linear for events 60 to 100. Note that the vertical scales are constant. Generally the data values lie within the 2 standard deviation limits.

(iv) histograms of the number of events in each period show the variability between surrogates. Figure 5.3 shows the histograms for periods 4 - 11 inclusive (these represent time intervals 0.15 - 0.20 to 0.50 - 0.55 respectively). Table 5.2 shows the data bin count and mean bin count for the surrogates together with the standard deviation of the surrogate bin counts.

All these plots show that the surrogates replicate the constant intensity adopted for the simulation.
Figure 5.2: Number of events by radius for indicated events under homogeneous process with $\lambda = 4$
Figure 5.3: Histograms showing events within periods 4 to 11 under homogeneous process with \( \lambda = 4 \)
5.3 Resampling a non-homogeneous pattern

The simulation of a non-homogeneous process with $\lambda(t) = t$ produced 204 events in 20 time periods. Again these event times were re-scaled to the (0, 1) interval. Figure 5.4(a) shows the plot of the cumulative number of events by time for both the data and the mean of 1000 surrogates. The increase in the intensity is shown by the increasing steepness of the plot. Clearly the mean of the 1000 surrogates closely follows the data.

Figure 5.4: Non-homogeneous process with intensity function $\lambda(t) = t$: (a) Cumulative number of events by time (b) Number of events within consecutive time periods of length 0.05 (c) Number of events by event number for indicated radius for one surrogate
Table 5.3: Non-homogeneous process: time of occurrence of data events.

<table>
<thead>
<tr>
<th>Event No.</th>
<th>20</th>
<th>30</th>
<th>70</th>
<th>100</th>
<th>120</th>
<th>130</th>
<th>140</th>
<th>150</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>0.32</td>
<td>0.39</td>
<td>0.59</td>
<td>0.72</td>
<td>0.78</td>
<td>0.81</td>
<td>0.84</td>
<td>0.86</td>
</tr>
</tbody>
</table>

Table 5.4: Non-homogeneous process: number of events by time period for data and surrogates.

<table>
<thead>
<tr>
<th>Period</th>
<th>Actual</th>
<th>$\bar{x}_s$</th>
<th>$\sigma_s$</th>
<th>Period</th>
<th>Actual</th>
<th>$\bar{x}_s$</th>
<th>$\sigma_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>5</td>
<td>4.863</td>
<td>0.787</td>
<td>10</td>
<td>9</td>
<td>11.43</td>
<td>2.209</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>5.15</td>
<td>0.850</td>
<td>11</td>
<td>13</td>
<td>11.821</td>
<td>2.350</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>5.06</td>
<td>0.771</td>
<td>12</td>
<td>12</td>
<td>11.521</td>
<td>2.359</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>5.137</td>
<td>0.875</td>
<td>13</td>
<td>12</td>
<td>11.383</td>
<td>2.106</td>
</tr>
<tr>
<td>8</td>
<td>7</td>
<td>5.359</td>
<td>0.928</td>
<td>14</td>
<td>11</td>
<td>12.995</td>
<td>2.984</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>7.852</td>
<td>1.414</td>
<td>15</td>
<td>19</td>
<td>15.534</td>
<td>3.398</td>
</tr>
</tbody>
</table>

The same plots used for the homogeneous pattern can help to assess how well the surrogate series replicate the data:

(i) Figure 5.4(b) shows the data bin counts for the number of events in consecutive time periods of length 0.05 together with the mean bin count for the surrogates.

(ii) the number of events within fixed radius for the middle 80% of events. Figure 5.4(c) shows the number of events for radii 0.01, 0.02, 0.05 and 0.10 for the 21st to the 184th event (inclusive) of the first surrogate together with the corresponding data plot. Note that the vertical scales vary with the radius. As expected, the surrogate series show some differences to the data when the radii are small but similar patterns with larger radii.

(iii) Figure 5.5 shows the number of events within increasing radii centred on eight specific events. The time of occurrences for the events shown in the plots are given in Table 5.3. It is clear that the surrogates replicate the data.

(iv) Figure 5.6 shows the histograms of the number of events in each period for periods 4 - 11 inclusive (these represent time intervals 0.15 - 0.20 to 0.50 - 0.55 respectively). Table 5.4 shows the data bin count and mean bin count for the surrogates together with the standard deviation of the surrogate bin counts.

All these plots show that the surrogates replicate the increasing intensity exhibited in the data.
Figure 5.5: Number of events by radius for indicated events under non-homogeneous process with $\lambda(t) = t$
Figure 5.6: Histograms showing events within periods 4 to 11 with $\lambda(t) = t$
5.4. Resampling a clustered pattern

The simulation of a clustered process produced 150 events. Figure 5.7(a) shows the plot of the cumulative number of events by time for both the data and the mean of 1000 surrogates. The sudden change in intensity is shown by the sharp increase in steepness of the plot. Clearly the mean of the 1000 surrogates closely follows the data.

Using the same plots as previously we can assess how well the surrogate series replicate the data:

(i) Figure 5.7(b) shows the data bin counts for time periods of length 0.05 together with the mean bin count for the surrogates. The clustering of events in the middle of the time interval is obvious.

(ii) the number of events within fixed radius for the middle 80% of events. Figure 5.7(c) shows the number of events for radii 0.05, 0.10, 0.15 and 0.20 for the 15th to the 135th event (inclusive) of the first surrogate together with the corresponding data plot. The surrogate series are very similar to the data for all radii shown.

(iii) Figure 5.8 shows the number of events within increasing radii centred on eight events. The time of occurrences for the events shown in the plots are given in table 5.5. Again it is clear that the surrogates replicate the data.

(iv) Figure 5.9 shows the histograms of the number of events for periods 4 - 11 inclusive (these represent time intervals 0.15 - 0.20 to 0.50 - 0.55 respectively). Table 5.6 shows the data bin count and mean bin count for the surrogates together with the standard deviation of the surrogate bin counts.

All these plots show that the surrogates replicate the change in intensity exhibited in the data.

<table>
<thead>
<tr>
<th>Period</th>
<th>Actual</th>
<th>$\bar{x}_s$</th>
<th>$\sigma_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>3</td>
<td>4.397</td>
<td>1.205</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>3.504</td>
<td>0.903</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>3.481</td>
<td>0.899</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>3.161</td>
<td>0.811</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>3.272</td>
<td>1.026</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>5.624</td>
<td>1.343</td>
</tr>
</tbody>
</table>

Table 5.5: Clustered process: time of occurrence of data events

<table>
<thead>
<tr>
<th>Event No.</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
<th>110</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>0.47</td>
<td>0.55</td>
<td>0.57</td>
<td>0.60</td>
<td>0.64</td>
<td>0.67</td>
<td>0.69</td>
<td>0.72</td>
</tr>
</tbody>
</table>

Table 5.6: Clustered process: number of events by time period for data and surrogates.
Chapter 5. Wavelet Resampling of 1D Point Patterns

Figure 5.7: Clustered process: (a) Cumulative number of events by time (b) Number of events within consecutive time periods of length 0.05 (c) Number of events by event number for indicated radius for one surrogate
Figure 5.8: Number of events by radius for indicated events under the clustered process specified in Sec 1.1.1(ii).
Figure 5.9: Histograms showing events within periods 4 to 11 under the clustered process specified in Sec 1.1.1(ii).
5.5 Testing the surrogates for homogeneity

The above analysis has shown that resampling after wavelet lifting is capable of generating 1D-point patterns that are possible realisations from the same source as the data. The analysis can now be taken a stage further to test the surrogates for homogeneity. The null hypothesis is that the data, and therefore the surrogates also, are homogeneous ie the time between any two events is independent of the time between any two other events.

A logical test statistic would be based on a comparison of the number of events within a suitable number of time intervals. There needs to be a balance between a reasonable number of intervals and the number of counts in each interval.

There were 150 events in each of the cases previously considered so a suitable number of intervals would be eight, with the time intervals \((0−0.125, 0.125−0.25, ..., 0.875−1)\). The expected number of events in each time interval, assuming homogeneity, is therefore \(\frac{150}{8} = 18.75\). The actual counts can be determined for each of the surrogate series, using the same time intervals, and the usual \(\chi^2\) test statistic used to test the homogeneity assumption.

5.5.1 Homogeneous patterns

Twenty simulated homogeneous patterns with randomly generated intensity from 1 to 10 events per period were used to test the procedure. Again PtProcess was utilised, with the following input:

\[
d=\text{pp.sim(data=NULL, params=c(intensity[j],0,0), simple.cif, TT=c(0,170), output=TRUE, seed=j, magn.sim=1,stopping.condition=NULL, max.rate=intensity[j])}
\]

where intensity=c(6,3,8,7,1,2,10,6,5,8,9,1,10,3,2,4,6,2,1,2).

The first 150 event times were taken for each series which were then lifted, resampled and inverted. The procedure outlined above was then used to find the expected number of counts within the given time intervals. The mean of 1000 \(\chi^2\) test statistics is shown in Table 5.7 for each run. Counts were determined for eight time intervals and therefore the critical value of the test statistic \((\alpha = 0.05)\) is \(\chi^2_{7,0.05} = 14.0671\). Only runs 15 and 20 (both with an intensity of 2) have a mean test statistic larger than this. Therefore the null hypothesis of homogeneity would be rejected for 2 of the 20 simulated series . Figure 5.10 shows the value of the test statistic plotted against the intensity.

Also shown in Table 5.7 are the \(\chi^2\) test statistics for the data. While generally there is good agreement between these values and the corresponding mean test statistics for the surrogates, there are a couple of discrepancies, for example runs 7, 9, 12, 15, 18 and 20. In particular, run 15 has a \(\chi^2\) test statistic for the data greater than the critical value (15.653 vs 14.067).

The fact that the mean of the test statistics is lower than the critical value for a particular
CHAPTER 5. WAVELET RESAMPLING OF 1D POINT PATTERNS

Table 5.7: Mean $\chi^2$ test statistic values for surrogates of simulated homogeneous series with random intensities (critical value = 14.0671). Also shown is the $\chi^2$ test statistic for the data ($\chi^2$ data).

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda$</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>5.224</td>
<td>2.427</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>4.373</td>
<td>4.027</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>3.030</td>
<td>4.453</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>6.289</td>
<td>5.307</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>5.550</td>
<td>8.613</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>7.473</td>
<td>7.44</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>8.255</td>
<td>3.813</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>4.985</td>
<td>7.12</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>10.483</td>
<td>3.707</td>
</tr>
<tr>
<td>10</td>
<td>8</td>
<td>3.049</td>
<td>3.493</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda$</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>9</td>
<td>7.371</td>
<td>7.867</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>5.716</td>
<td>10.427</td>
</tr>
<tr>
<td>13</td>
<td>10</td>
<td>6.249</td>
<td>8.72</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>4.839</td>
<td>8.507</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>20.111</td>
<td>15.653</td>
</tr>
<tr>
<td>16</td>
<td>4</td>
<td>5.495</td>
<td>6.48</td>
</tr>
<tr>
<td>17</td>
<td>6</td>
<td>8.576</td>
<td>10.96</td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>12.529</td>
<td>7.44</td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>5.044</td>
<td>5.627</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>16.490</td>
<td>10.107</td>
</tr>
</tbody>
</table>

Table 5.8: Details of runs for the homogeneous patterns where at least one of the surrogates had a test statistic larger than the critical value (14.0671). Shown are the run number, intensity, the percentage of series with a test statistic less and greater than critical value and the $p$-value of the test statistic from the data.

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda$</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>7</td>
<td>0.993</td>
<td>0.623</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>0.989</td>
<td>0.385</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>0.924</td>
<td>0.801</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>0.998</td>
<td>0.416</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>0.803</td>
<td>0.813</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda$</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>10</td>
<td>0.998</td>
<td>0.002</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>0.625</td>
<td>0.375</td>
</tr>
<tr>
<td>17</td>
<td>6</td>
<td>0.952</td>
<td>0.048</td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>0.635</td>
<td>0.375</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>0.340</td>
<td>0.660</td>
</tr>
</tbody>
</table>

Run of course does not ensure that the test statistics for each of the individual surrogate series are all lower. An analysis of the runs showed that half had at least one surrogate with test statistic greater than the critical value. The results are given in Table 5.8 where it can be seen that Runs 9 ($\lambda = 5$), 15, 18 and 20 ($\lambda = 2$ in each case) had a reasonable proportion of surrogates with a test statistic larger than the critical value, with run 15 being an extreme case. Also shown in Table 5.8 are the $p$-values for the test statistic from the data and run 15 only would be rejected as homogeneous on this basis.

Figure 5.11 shows boxplots of the test statistics for the surrogates from all 20 runs together with the data test statistic, clearly illustrating the similarity of the surrogates to the data.

5.5.2 Non-homogeneous patterns

The intensity function used in the PtProcess package is $\lambda(t) = a + bt^g$ and therefore various combinations of $a, b$ and $g$ can be used to test the effectiveness of the proposed hypothesis testing procedure. Letting $g = 1$ restricts the intensity function to be linear; the remaining parameters were randomly chosen subject to the arbitrarily chosen restrictions $0 \leq a \leq 5$ and $0 \leq b \leq 2$. Again 150 events were simulated for 20 different intensity functions and the critical $\chi^2$ value was 14.0671. Table 5.9 shows the parameter values and the means of the test statistics together with the test statistics for the data. The null hypothesis of homogeneity would be rejected in 17 out of
5.5. TESTING THE SURROGATES FOR HOMOGENEITY

Figure 5.10: Scatterplot showing value of the test statistic, assuming homogeneity, versus intensity for the simulated homogeneous series. The critical $\chi^2$ value is 14.0671.

Figure 5.11: Boxplots of the test statistics for the homogeneous surrogates with the test statistic from the data shown as a filled circle (a) runs 1 - 10 and (b) runs 11 - 20. The horizontal line is the critical value.
Table 5.9: Mean $\chi^2$ test statistic values for surrogates of simulated non-homogeneous patterns with random intensity functions (critical value = 14.0671). Also shown is the $\chi^2$ test statistic for the data ($\chi^2$ data).

<table>
<thead>
<tr>
<th>Run</th>
<th>a</th>
<th>b</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
<th>Run</th>
<th>a</th>
<th>b</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>1.0</td>
<td>29.327</td>
<td>29.627</td>
<td>11</td>
<td>1</td>
<td>0.4</td>
<td>28.699</td>
<td>30.267</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2.0</td>
<td>38.531</td>
<td>41.04</td>
<td>12</td>
<td>3</td>
<td>0.8</td>
<td>25.595</td>
<td>20.667</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1.1</td>
<td>34.360</td>
<td>37.52</td>
<td>13</td>
<td>4</td>
<td>0.5</td>
<td>19.377</td>
<td>18.747</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>1.9</td>
<td>25.754</td>
<td>20.88</td>
<td>14</td>
<td>4</td>
<td>1.2</td>
<td>40.912</td>
<td>43.387</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1.2</td>
<td>34.158</td>
<td>33.573</td>
<td>15</td>
<td>0</td>
<td>0.6</td>
<td>30.564</td>
<td>31.44</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>0.2</td>
<td>19.884</td>
<td>17.787</td>
<td>16</td>
<td>3</td>
<td>1.2</td>
<td>32.331</td>
<td>40.293</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>0.4</td>
<td>13.114</td>
<td>12.987</td>
<td>17</td>
<td>3</td>
<td>0.3</td>
<td>8.050</td>
<td>7.973</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>0.2</td>
<td>7.464</td>
<td>6.373</td>
<td>18</td>
<td>5</td>
<td>0.9</td>
<td>21.175</td>
<td>21.627</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>1.2</td>
<td>34.935</td>
<td>38.373</td>
<td>19</td>
<td>3</td>
<td>1.9</td>
<td>32.252</td>
<td>31.227</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>1.2</td>
<td>29.126</td>
<td>29.2</td>
<td>20</td>
<td>2</td>
<td>1.5</td>
<td>38.392</td>
<td>47.013</td>
</tr>
</tbody>
</table>

The 20 surrogates, the exceptions being runs 7, 8 and 17 with mean test statistics of 13.114, 7.464 and 8.050 respectively. In each case the test statistic from the data is less than the critical value, thus implying homogeneity.

In run 7 the intensity function is $\lambda(t) = 5 + 0.4t$, in run 8 it is $\lambda(t) = 4 + 0.2t$ while in run 17 it is $\lambda(t) = 3 + 0.3t$; in all cases the ratio $\frac{a}{b}$ is high (12.5, 20 and 10 respectively) so it may be the case that the constant part of the intensity functions dominates the variable part.

Table 5.10 shows, for these runs, the number of surrogates with test statistics less than and greater than the critical value (14.0671) for assuming homogeneity. Run 13 ($\frac{a}{b}$ ratio= 8) also appears in this Table because 6 of the 1000 surrogates had a test statistic less than the critical value. Also shown is the $p$-value for the test statistic from the data.

Figure 5.12(a) shows the value of the test statistic plotted against the $\frac{a}{b}$ ratio, indicating that the null hypothesis is most likely to be rejected when the constant term is a less dominant part of the intensity function.

Figure 5.13 shows boxplots of the test statistics for the surrogates from all 20 runs together with the data test statistic. The similarity between surrogates and data is stronger here than for the homogeneous patterns.

Table 5.10: Details of runs for the non-homogeneous patterns where at least one of the surrogates had a test statistic smaller than the critical value (14.0671). Shown are the run number, $a$ and $b$ values, the percentage of series with a test statistic less and greater than critical value and the $p$-value of the test statistic from the data.

<table>
<thead>
<tr>
<th>Run</th>
<th>a</th>
<th>b</th>
<th>$&lt;$</th>
<th>$&gt;$</th>
<th>$p$-value</th>
<th>ts</th>
<th>Run</th>
<th>a</th>
<th>b</th>
<th>$&lt;$</th>
<th>$&gt;$</th>
<th>$p$-value</th>
<th>ts</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>5</td>
<td>0.4</td>
<td>0.683</td>
<td>0.317</td>
<td>0.072</td>
<td></td>
<td>13</td>
<td>4</td>
<td>0.5</td>
<td>0.006</td>
<td>0.994</td>
<td>0.009</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>0.2</td>
<td>0.982</td>
<td>0.018</td>
<td>0.497</td>
<td></td>
<td>17</td>
<td>3</td>
<td>0.3</td>
<td>0.983</td>
<td>0.017</td>
<td>0.335</td>
<td></td>
</tr>
</tbody>
</table>
Figure 5.12: Scatterplot showing value of the test statistic, assuming homogeneity, versus (a) $a/b$ ratio for the non-homogeneous patterns, and (b) $|\lambda_2 - \lambda_1|$ for the clustered patterns. The dotted line shows the critical $\chi^2$ value.

Figure 5.13: Boxplots of the test statistics for the non-homogeneous surrogates with the test statistic form the data shown as a filled circle (a) runs 1 - 10 and (b) runs 11 - 20. The horizontal line is the critical value.
5.5.3 Clustered patterns

The effectiveness of the hypothesis testing procedure was also tested with examples of clustered patterns. In part (d) above a simulated cluster pattern with 150 events was obtained from combining three sets of 50 random exponentially distributed numbers, with the first and last set having the same mean and the middle set a higher mean. Again 20 runs with different parameters were used to generate data, each series also consisting of three sets of 50 random exponentially distributed numbers. As previously, the first and last set had the same mean, being randomly chosen from the integers 1 to 5 inclusive, while the middle set had a mean randomly chosen from the integers 1 to 10.

Table 5.11 shows the means ($\lambda_1$ for the first and last set and $\lambda_2$ for the middle set) and the means of the test statistics together with the test statistics for the data. The critical $\chi^2$ value was again 14.0671 and runs 5, 6, 8, 11 and 13 had mean test statistic values less than this ie the null hypothesis of homogeneity would not be rejected in these cases. With the exception of run 8, the test statistics for the data in these runs also are less than the critical value.

In runs 5, 6 and 11 the value of $\lambda_2$ is only one more than $\lambda_1$ so there is not a large clustering effect. The value of $\lambda_2 - \lambda_1$ is 4 for run 8 and 3 for run 13, indicating that the clustering effect should have been more pronounced. Figure 5.12(b) shows the value of the test statistic against the difference in the means.

Table 5.12 gives details of runs where at least one surrogate series had a test statistic less than the critical value. Besides those already mentioned, runs 9, 12 and 16 have a significant number of surrogates with test statistics less than the critical value. Of note is run 12 where $\lambda_2 - \lambda_1 = 6$ yet 90 surrogates have test statistics less then the critical value, and therefore would be classified as homogeneous. In contrast, run 2 also has $\lambda_2 - \lambda_1 = 6$ with all surrogates having test statistics greater than the critical value. The values of the test statistic for the data are 77.413 and 16.613 for runs 2 and 12 respectively so the surrogates do appear to be replicating the data.

Figure 5.14 shows boxplots of the test statistics for the surrogates from all 20 runs together with the data test statistic.

5.6 Conclusion

Previously the graphical approach was used to check the agreement between surrogates and data. In this section various types of processes with different parameter values were tested for homogeneity. Generally the mean of the test statistics from the surrogates agreed with the test statistic from the data and where the surrogates misclassified the data the same result would have occurred by applying the test to the data. The results give further support to the conclusion that the surrogates replicate the data.
Table 5.11: Mean $\chi^2$ test statistic values for surrogates of simulated clustered series with random intensities (critical value = 14.0671). Also shown is the $\chi^2$ test statistic for the data ($\chi^2$ data).

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda_1$</th>
<th>$\lambda_2$</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>18.674</td>
<td>19.813</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>10</td>
<td>69.244</td>
<td>77.413</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>8</td>
<td>27.268</td>
<td>30.907</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>58.174</td>
<td>42.43</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>11.100</td>
<td>12.453</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>6</td>
<td>10.387</td>
<td>9.253</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>10</td>
<td>68.318</td>
<td>97.36</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>9</td>
<td>12.041</td>
<td>16.187</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>3</td>
<td>18.368</td>
<td>24.4</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>4</td>
<td>19.399</td>
<td>17.04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda_1$</th>
<th>$\lambda_2$</th>
<th>$\chi^2$ test stat</th>
<th>$\chi^2$ data</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>3</td>
<td>4</td>
<td>4.394</td>
<td>3.6</td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>10</td>
<td>17.248</td>
<td>16.613</td>
</tr>
<tr>
<td>13</td>
<td>4</td>
<td>7</td>
<td>11.784</td>
<td>12.133</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>9</td>
<td>49.844</td>
<td>52.24</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>3</td>
<td>24.812</td>
<td>29.413</td>
</tr>
<tr>
<td>16</td>
<td>3</td>
<td>6</td>
<td>14.810</td>
<td>14.373</td>
</tr>
<tr>
<td>17</td>
<td>3</td>
<td>8</td>
<td>27.966</td>
<td>25.787</td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>9</td>
<td>74.515</td>
<td>80.933</td>
</tr>
<tr>
<td>19</td>
<td>2</td>
<td>7</td>
<td>22.466</td>
<td>22.16</td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>9</td>
<td>21.059</td>
<td>25.04</td>
</tr>
</tbody>
</table>

Table 5.12: Details of runs for the clustered series where at least one of the surrogates had a test statistic smaller than the critical value (14.0671). Shown are the run number, $\lambda_1$ and $\lambda_2$ values, the percentage of series with a test statistic less and greater than critical value and the $p$-value of the test statistic from the data.

<table>
<thead>
<tr>
<th>Run</th>
<th>$\lambda_1$</th>
<th>$\lambda_2$</th>
<th>&lt;</th>
<th>&gt;</th>
<th>p-value</th>
<th>ts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0.024</td>
<td>0.976</td>
<td>0.006</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>0.961</td>
<td>0.039</td>
<td>0.087</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>6</td>
<td>0.870</td>
<td>0.130</td>
<td>0.235</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>9</td>
<td>0.758</td>
<td>0.242</td>
<td>0.024</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>3</td>
<td>0.246</td>
<td>0.754</td>
<td>0.001</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>4</td>
<td>0.005</td>
<td>0.995</td>
<td>0.017</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>4</td>
<td>1.000</td>
<td>0.0</td>
<td>0.175</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>10</td>
<td>0.090</td>
<td>0.910</td>
<td>0.020</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>4</td>
<td>7</td>
<td>0.906</td>
<td>0.094</td>
<td>0.096</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>3</td>
<td>6</td>
<td>0.380</td>
<td>0.620</td>
<td>0.045</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>9</td>
<td>0.004</td>
<td>0.996</td>
<td>0.001</td>
<td></td>
</tr>
</tbody>
</table>

Figure 5.14: Boxplots of the test statistics for the clustered surrogates with the test statistic from the data shown as a filled circle (a) runs 1 - 10 and (b) runs 11 - 20. The horizontal line is the critical value.
Chapter 6

Wavelet Resampling of
2-dimensional Point Patterns

6.1 Introduction

Two-dimensional spatial data are categorised into three types known as:

1. Geostatistical where values of a pre-defined variable are recorded at chosen point locations in a domain;
2. Areal where observations are recorded for specific contiguous areas in a domain;
3. Point Patterns where the locations of points in a domain are recorded.

6.2 Spatial Point Patterns

Spatial Point Patterns are therefore the simplest type of 2-dimensional data where the item of interest is the location itself rather than any attribute at that location. The only information available are the $x$ and $y$ coordinates of each point.

A variation on this simple data type is to include the value of a covariate (either discrete or continuous) and such processes are termed Marked Point Patterns. For example, the positions of trees may be located in a field and the covariate might be the type of tree or trunk diameter.

The remainder of this work looks at wavelet resampling of point patterns and this chapter focuses on basic point patterns. These patterns, are characterised by:

- Intensity, which measures the number of points per unit area. Intensity may be homogeneous, where the expected number of points per unit area is constant over the domain, or non-homogeneous where the expected number of points varies from one section of the domain to another.
• Interaction, which measures the relationship between points. There may be no interaction at all, in which case the points are said to be independent, or various types of interaction present, for example, the points may be regularly spaced, clustered or inhibited from being too close together.

The basic point process is the uniform Poisson process, also known as Complete Spatial Randomness, which has the properties:

(a) the expected number of points lying in area $A$ is $\lambda A$ where $\lambda$ is the uniform intensity;

(b) the expected number of points lying in two disjoint areas are independent.

6.3 Lifting point patterns

The first stage in applying the lifting algorithm to 1-dimensional data is to split the data into odd and even subsets as defined by the observation number. With 2-dimensional data there is no corresponding concept of order so the odd/even split cannot be applied. Hence we are faced with a dilemma and have to find another way of applying the lifting concept.

The purpose of splitting the data into odd and even subsets was to use the evens to predict the odds and then find the difference between the actual and predicted odd values. The odd values were then deleted and the even values lifted to maintain the mean. The process was repeated with the remaining evens.

For 2-dimensional data a suitable approach would be to work with one point at a time. The selected point can be considered as a single odd point and the even points to be the surrounding points, defined in an appropriate way. The mean location of these surrounding points can be used to predict the location of the selected point and thus the difference between the predicted and actual location obtained as per the 1-dimensional case. Then the selected point can be deleted, the surrounding points lifted and the process repeated.

An appropriate method for obtaining surrounding points is to use Delaunay triangulation which is based on a Voronoi diagram. This diagram constructs polygons around points in a domain $D$ such that the polygons contain parts of $D$ that are closest to their associated point. The Delaunay triangulation joins points having neighbouring polygons that share a common side.

The Mathematica Add-on package DiscreteMath' ComputationalGeometry' [41] was used to obtain the Voronoi diagrams and associated information.

6.3.1 Preparing the data

Suppose we have the 2-dimensional location of a set of points. If there is no information regarding the window in which the points are enclosed then we have to define an appropriate window which may be a simple rectangle or an elaborate polygon. At present the algorithm given below only
caters for a rectangular window. The boundaries defining the window should not be too close or too far from the convex hull of the data. The R spatstat [4] package has a function called ripras that calculates the Ripley-Rasson estimate of the spatial domain from which the data are assumed to have originated. Obviously it would be appropriate to round the estimates to 1 or 2 decimal places. These coordinates are the window vertices and are used as input, together with the data, to the BoundedDiagram function in ComputationalGeometry that calculates the bounded Voronoi diagram. Included in the output from this function is a list of the vertices for each cell in the Voronoi diagram, some of which lie on the boundary but are not vertices. All the boundary points are then prepended to the data as input to the decomposition algorithm. The rationale for this is to treat all the original data consistently while constraining the surrogates to lie within a fixed domain.

Table 6.1 lists the coordinates of a sample set of data points as shown in Figure 6.1(a). The window vertices were chosen as the intersection of the lines \( x=0.2, \ x=0.7, \ y=0.1 \) and \( y=0.6. \) (The limits estimated by ripras were \( 0.24018 < x < 0.68571 \) and \( 0.1365 < y < 0.61436 \).) Table 6.1 also lists the coordinates of all the points that lie on the boundary, using the output from the BoundedDiagram function. Figure 6.1(b) shows the bounded Voronoi diagram while Figure 6.2(a) shows the Voronoi diagram when the boundary points are included with the original data. The corresponding triangulation is shown in Figure 6.2(b).

![Figure 6.1: Sample data (a) point locations and (b) bounded Voronoi diagram including points created on the boundary](image)

### 6.3.2 The decomposition algorithm

The steps in the algorithm are:

1. Extract the interior points by taking the complement of the Delaunay Triangulation with the convex hull (ie the introduced boundary points).

2. Set \( updatedata = data \) and counter \( j = 1. \)
Table 6.1: Sample data and calculated boundary point coordinates

<table>
<thead>
<tr>
<th>Data Point</th>
<th>x</th>
<th>y</th>
<th>Boundary Point</th>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.584591</td>
<td>0.558546</td>
<td>1</td>
<td>0.7</td>
<td>0.1</td>
</tr>
<tr>
<td>2</td>
<td>0.614659</td>
<td>0.532854</td>
<td>2</td>
<td>0.7</td>
<td>0.269514</td>
</tr>
<tr>
<td>3</td>
<td>0.583759</td>
<td>0.531099</td>
<td>3</td>
<td>0.7</td>
<td>0.349341</td>
</tr>
<tr>
<td>4</td>
<td>0.565701</td>
<td>0.296909</td>
<td>4</td>
<td>0.7</td>
<td>0.480788</td>
</tr>
<tr>
<td>5</td>
<td>0.625104</td>
<td>0.372021</td>
<td>5</td>
<td>0.7</td>
<td>0.6</td>
</tr>
<tr>
<td>6</td>
<td>0.605419</td>
<td>0.184828</td>
<td>6</td>
<td>0.646022</td>
<td>0.6</td>
</tr>
<tr>
<td>7</td>
<td>0.504882</td>
<td>0.566031</td>
<td>7</td>
<td>0.548278</td>
<td>0.6</td>
</tr>
<tr>
<td>8</td>
<td>0.640649</td>
<td>0.400344</td>
<td>8</td>
<td>0.326666</td>
<td>0.6</td>
</tr>
<tr>
<td>9</td>
<td>0.285243</td>
<td>0.423367</td>
<td>9</td>
<td>0.2</td>
<td>0.6</td>
</tr>
<tr>
<td>10</td>
<td>0.497919</td>
<td>0.454503</td>
<td>10</td>
<td>0.2</td>
<td>0.130119</td>
</tr>
<tr>
<td>11</td>
<td>0.473176</td>
<td>0.266632</td>
<td>11</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>12</td>
<td>0.285123</td>
<td>0.454503</td>
<td>12</td>
<td>0.461523</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Figure 6.2: Sample data (a) Voronoi diagram with boundary points included with the data and (b) Triangulation of original data and introduced boundary points
3. Calculate the Bounded Diagram using the specified boundary and updatedata. This provides the bounded Voronoi diagram and the output consists of two lists (a) the coordinates of all the vertices, and (b) a vertex adjacency list.

4. The Tile Areas of the bounded diagram are calculated next and sorted in ascending order. The point to be deleted is chosen as the interior point that has the minimum tile area. This follows the algorithm in the 1-dimensional case where values are compared with averages over increasing intervals as the decomposition progresses.

5. The Delaunay Triangulation is used to obtain a list of all the points, interior and boundary, together with a list of the points to which they are triangulated. The deleted point and its neighbours are taken from this list.

6. The information obtained in (4) is used to extract the coordinates of the deleted point and its neighbours. The mean position of the neighbours is calculated and then the difference vector, \( \mathbf{d}_j \), between this and the position of the deleted point is obtained.

7. Uplift the coordinates of the connected vertices that are interior points by the difference in the mean coordinates, \( \mathbf{d}^m \), when the coordinates of the deleted point are included/excluded. This difference in the means is just the difference between actual and predicted coordinates, as calculated in Step 6, divided by (the number of connected vertices + 1). This result is derived below, using \( \mathbf{x} \) as the position vector:

\[
\mathbf{d}^m = \frac{1}{n + 1} \sum_{i=1}^{n+1} \mathbf{x}_i - \frac{1}{n} \sum_{i=2}^{n+1} \mathbf{x}_i
\]

\[
= \frac{x_1}{n + 1} + \frac{1}{n + 1} \sum_{i=2}^{n+1} x_i - \frac{1}{n} \sum_{i=2}^{n+1} x_i
\]

\[
= \frac{x_1}{n + 1} + \frac{-1}{n(n + 1)} \sum_{i=2}^{n+1} x_i
\]

\[
= \frac{x_1}{n + 1} + \frac{-1}{n(n + 1)} n(x_1 - \mathbf{d})
\]

\[
= \frac{\mathbf{d}}{n + 1}
\]  

(6.1)

Note that the positions of boundary points are unaltered.

8. The point information in updatedata is updated by removing the deleted point and altering the neighbours that have their coordinates lifted.

9. Remove the deleted point from the list of interior points. Check to see if any of the points have moved beyond the border; if so reflect the point in the border so that it again lies in the interior. Increment \( j \) by 1.

10. Repeat steps (2) to (9) until there are no more interior points.
6.3. LIFTING POINT PATTERNS

6.3.3 The first level of the decomposition

The first point to be removed is number 13 \((0.584591, 0.558546)\) as this is the interior point with the minimum tile area. The points joined to 13 are 6, 7, 14, 15 and 19 of which the first two are on the boundary. The mean position of these five points is \((0.57952, 0.565997)\) and therefore the vector difference \((d_{1})\) between the location of point 13 and the mean of the connected points is \((0.584591 - 0.57952 = 0.005071, 0.558546 - 0.565997 = -0.007451)\). The remaining interior points are lifted by the vector difference in the means of the points including and excluding point 13. The mean position of points 6, 7, 13, 14, 15 and 19 is \((0.580365, 0.564755)\) and the difference between the means is \((0.580365 - 0.57952 = 0.000845, 0.564755 - 0.565997 = -0.001242)\). This difference in the means can also be calculated from \(d_{1}/6\) (there are five connected vertices and one deleted point giving the denominator 6) and then added to the coordinates of points 14, 15 and 19 as shown in Table 6.2. The updated data, shown in Figure 6.3, is then used as input to level 2 of the decomposition.

![Figure 6.3: Updated data after deleting the first point](image)

Table 6.2: Updated point coordinates

<table>
<thead>
<tr>
<th>Data Point</th>
<th>New number</th>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>13</td>
<td>0.615504</td>
<td>0.531612</td>
</tr>
<tr>
<td>15</td>
<td>14</td>
<td>0.584604</td>
<td>0.529857</td>
</tr>
<tr>
<td>19</td>
<td>18</td>
<td>0.505728</td>
<td>0.564789</td>
</tr>
</tbody>
</table>

6.3.4 The subsequent levels of the decomposition

Figure 6.4 shows the progressively updated data for levels 2 to 11 when there are no more interior points. Table 6.3 gives the differences between the location of the deleted points and the average location of their neighbours at each level. These differences are shown in Figures 6.5(a) and 6.5(b) for the \(x\) and \(y\) coordinates respectively.

A fundamental difference between wavelet lifting in one dimension and the approach taken here in two dimensions is the number of values at each level of the decomposition. If \(n\) is the number of observations in a 1-dimensional series then there are \(n/2^{j}\) values at level \(j = 1, 2, \ldots\) of the decomposition. However in 2 dimensions there is always only one point at each level. This has
repercussions when resampling occurs: with 1-dimensional data, resampling only occurs within each level where it is presumed there is consistency of scale. There is no choice but to resample across all levels with 2-dimensional data and therefore there is a possibility of mixing scales.

6.3.5 Assumptions behind the resampling procedure

In the 1-dimensional case only coefficients in wavelet packet crystals that passed a test for independence were resampled. In the 2-dimensional case each level has one coefficient, corresponding to the deletion of each point. Therefore we cannot resample within levels; resampling can only occur across all levels ie among all difference vectors. The assumption required is that the difference vectors are independent and this is tested by plotting the autocorrelation functions for the difference between the $x$ and $y$ co-ordinates of the deleted point and the mean position of the connected vertices. Since the difference vector is resampled, rather than the $x$ and $y$ co-ordinate differences separately, it is necessary to also plot the cross correlation functions.

No comparison was made, in the 1-dimensional case, of the relative magnitude of the coefficients within a wavelet packet crystal. Therefore no assumptions are made regarding the magnitude of the components of the difference vectors in the 2-dimensional case. The differences are plotted for illustrative purposes only and it may be seen in this and subsequent chapters that generally the magnitude remains fairly consistent except for the last few values, corresponding to the deletion of the last few points in the window. Since the points are deleted in order of the size of the cell, it may be thought that there could be a relationship between cell size and the difference between the location of the deleted point and the mean location of the connected vertices. However this does not appear to be the case and an argument might be made that there is no justification for such a relationship.

6.3.6 Resampling and recomposition

After all the interior points have been deleted we are left with the boundary points in their original positions and lists of:

1. the differences between the deleted points and mean position of their neighbours (the list of differences);
2. the index numbers of the point deleted at each stage;
3. the index numbers of the connected vertices at each stage.

The differences, mean positions and the uplift values for the sample data are given in Table 6.3.

Only the differences are resampled. The steps in the recomposition, including resampling, are:

1. Set $\text{revdata}$ to the final $\text{updatedata}$ from the decomposition.
6.3. LIFTING POINT PATTERNS

Figure 6.4: Levels 2 to 11 of the decomposition (read from left to right then down)

Figure 6.5: Differences between deleted points and mean of their neighbours: (a) $x$ coordinate and (b) $y$ coordinate
Table 6.3: Differences between deleted points and mean position of nearest neighbours \((d_x, d_y)\), the mean position of neighbours to a deleted point \((m_x, m_y)\) and the uplift values, being differences between mean positions of neighbours including and excluding deleted point \((u_x, u_y)\)

<table>
<thead>
<tr>
<th>Stage</th>
<th>(d_x)</th>
<th>(d_y)</th>
<th>(m_x)</th>
<th>(m_y)</th>
<th>(u_x)</th>
<th>(u_y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.005071</td>
<td>-0.007451</td>
<td>0.57952</td>
<td>0.565997</td>
<td>0.000845</td>
<td>-0.001242</td>
</tr>
<tr>
<td>2</td>
<td>-0.038751</td>
<td>0.009414</td>
<td>0.654255</td>
<td>0.522198</td>
<td>-0.006458</td>
<td>0.001569</td>
</tr>
<tr>
<td>3</td>
<td>0.005542</td>
<td>0.017585</td>
<td>0.619562</td>
<td>0.354436</td>
<td>0.000924</td>
<td>0.002931</td>
</tr>
<tr>
<td>4</td>
<td>-0.010852</td>
<td>0.013450</td>
<td>0.588997</td>
<td>0.517976</td>
<td>-0.001550</td>
<td>0.001921</td>
</tr>
<tr>
<td>5</td>
<td>0.046765</td>
<td>0.013592</td>
<td>0.457412</td>
<td>0.553119</td>
<td>0.011691</td>
<td>0.003398</td>
</tr>
<tr>
<td>6</td>
<td>0.014661</td>
<td>0.008584</td>
<td>0.618902</td>
<td>0.398181</td>
<td>0.002932</td>
<td>0.001717</td>
</tr>
<tr>
<td>7</td>
<td>-0.028545</td>
<td>-0.005400</td>
<td>0.598102</td>
<td>0.306957</td>
<td>-0.004757</td>
<td>-0.000900</td>
</tr>
<tr>
<td>8</td>
<td>0.018176</td>
<td>0.000117</td>
<td>0.582485</td>
<td>0.183811</td>
<td>0.003635</td>
<td>0.000023</td>
</tr>
<tr>
<td>9</td>
<td>-0.018308</td>
<td>-0.010608</td>
<td>0.525466</td>
<td>0.474179</td>
<td>-0.002288</td>
<td>-0.001326</td>
</tr>
<tr>
<td>10</td>
<td>-0.065987</td>
<td>-0.016869</td>
<td>0.348942</td>
<td>0.438911</td>
<td>-0.010998</td>
<td>-0.002811</td>
</tr>
<tr>
<td>11</td>
<td>-0.045285</td>
<td>-0.097245</td>
<td>0.504052</td>
<td>0.358863</td>
<td>-0.004528</td>
<td>-0.009724</td>
</tr>
</tbody>
</table>

2. Using a counter \(k\) from \(j - 1\) to 1 in steps of \(-1\), randomly select a difference vector, \(d_r\), and retrieve the index numbers of the \(n\) connected vertices at the relevant stage. Reverse the uplift step by subtracting \(\frac{d_r}{n+1}\) from the locations of the connected vertices that are interior points.

3. Find the mean position of the connected vertices and add \(d_r\) to give the position of the point to be added. Assign this point the index of the point removed at the relevant stage of the forward transform.

4. Any points that fall outside the domain are reflected back inside, using the appropriate boundary.

5. The above procedure is repeated until all deleted points are replaced ie the resampled point pattern, consisting of the interior points only, has the same number of points as the original data.

The coordinates of all points existing after one stage of resampling and reconstruction are shown in columns 2 and 3 of Table 6.4. Note that the first 12 points lie on the boundary and therefore have the same coordinates. Point 13 comes from line 11 of Table 6.3 and a randomly selected difference vector from Table 6.3, \((-0.010852, 0.01345)\) in this case. That is,

\[(0.504052, 0.358863) + (-0.010852, 0.01345) = (0.4932, 0.372313)\]

The coordinates of all points existing after two stages of resampling and reconstruction are shown in the final two columns of Table 6.4. Again the first 12 points have unaltered coordinates.

The new point, Point 14, is connected to Points 7, 8, 9, 10 and 13. The randomly selected difference vector is \((0.005071, -0.007451)\) and \(\frac{1}{8}\) of this difference is subtracted from Point 13 as
Figure 6.6: Stages 11 to 2 of the recomposition

Figure 6.7: Triangulation of (a) original data and (b) a surrogate
Table 6.4: Point coordinates after (i) one stage (columns 2 and 3) and (ii) two stages of resampling and recomposition (columns 4 and 5). Note that Point 14 does not exist in the first stage of recomposition.

<table>
<thead>
<tr>
<th>Point No.</th>
<th>x</th>
<th>y</th>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.7</td>
<td>0.1</td>
<td>0.7</td>
<td>0.1</td>
</tr>
<tr>
<td>2</td>
<td>0.7</td>
<td>0.269514</td>
<td>0.7</td>
<td>0.269514</td>
</tr>
<tr>
<td>3</td>
<td>0.7</td>
<td>0.349341</td>
<td>0.7</td>
<td>0.349341</td>
</tr>
<tr>
<td>4</td>
<td>0.7</td>
<td>0.480788</td>
<td>0.7</td>
<td>0.480788</td>
</tr>
<tr>
<td>5</td>
<td>0.7</td>
<td>0.6</td>
<td>0.7</td>
<td>0.6</td>
</tr>
<tr>
<td>6</td>
<td>0.646022</td>
<td>0.6</td>
<td>0.646022</td>
<td>0.6</td>
</tr>
<tr>
<td>7</td>
<td>0.548278</td>
<td>0.6</td>
<td>0.548278</td>
<td>0.6</td>
</tr>
<tr>
<td>8</td>
<td>0.326666</td>
<td>0.6</td>
<td>0.326666</td>
<td>0.6</td>
</tr>
<tr>
<td>9</td>
<td>0.2</td>
<td>0.6</td>
<td>0.2</td>
<td>0.6</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>0.130119</td>
<td>0.2</td>
<td>0.130119</td>
</tr>
<tr>
<td>11</td>
<td>0.2</td>
<td>0.1</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>12</td>
<td>0.461523</td>
<td>0.1</td>
<td>0.461523</td>
<td>0.1</td>
</tr>
<tr>
<td>13</td>
<td>0.4932</td>
<td>0.372313</td>
<td>0.492355</td>
<td>0.373555</td>
</tr>
<tr>
<td>14</td>
<td>—</td>
<td>—</td>
<td>0.358531</td>
<td>0.453284</td>
</tr>
</tbody>
</table>

it is the only connected point in the interior. Thus the coordinates of Point 13 become

$$(0.4932, 0.372313) - \frac{1}{6}(0.005071, -0.007451) = (0.492355, 0.373555)$$

The mean position of Points 7, 8, 9, 10 and 13 is then (0.353459, 0.460735) and the difference vector is added to obtain the position of Point 14

$$(0.353459, 0.460735) + (0.005071, -0.007451) = (0.358531, 0.453284)$$

Note that in Step 2 of the reconstruction shown in Figure 6.6 Mathematica has, confusingly, renumbered Point 13 as Point 14 and indexed the new point as Point 13. In contrast, Table 6.4 maintains the index number at the location.

Repeating the procedure until all interior points have been replaced yields the surrogate point pattern shown in Figure 6.7.

### 6.4 Diagnostics

Diagnostics can be used to determine how well the data fit an assumed model or, in this case, how well the surrogate point patterns agree with the original. There are two types of diagnostics that we can consider:

1. those that look at the position of the points in relation to the window, or tests for Complete Spatial Randomness (CSR);

2. those that look at the relationship of the points relative to each other, known as distance
The simplest diagnostic of the first type is the *quadrat test* that divides the window into equally sized quadrats and counts the number of points lying in each. A chi-square test is then used to test the similarity of two point patterns, under the assumption that the intensity is uniform over the domain. *spatstat* has a neat presentation of the chi-square test: the quadrats are overlaid on the data and the actual and expected counts together with the Pearson residual are shown for each quadrat.

A more powerful test for CSR is the Kolmogorov-Smirnov test that compares the actual and expected values of some predefined function $F(x, y)$ that is evaluated at each data point location. *spatstat* implements the test and plots the actual and expected CDFs and also provides a $p$-value.

There are three distance methods that are commonly used for homogeneous point patterns:

1. **empty space distances** which measure the distance from a fixed empty location in the window to the nearest data point;
2. **nearest neighbour distances** which measure the distances from each data point to their nearest neighbours;
3. **pairwise distances** which measure the distances between each distinct pair of points in the pattern.

Probability functions are defined for each of these distance methods.

1. The empty space function, $F$, is the cumulative distance function of the empty space distances, assuming that the point pattern is stationary. That is, $F(r) = P(|u - x_i| \leq r)$ for some empty point $u$, data point $x_i$ (both in the window) and radius $r$. The empty space distances are calculated for the data using a finely spaced grid and an edge correction is included so that the empirical distribution function, $\hat{F}(r)$, is an unbiased estimate of $F(r)$, under the assumption that the point process is homogeneous. The empty space function for a homogeneous Poisson process of intensity $\lambda$ is

   $$F_{Po}(r) = 1 - e^{-\lambda \pi r^2}$$

   (6.2)

   and is a useful standard for comparison. An indication of a clustered point pattern occurs when $\hat{F}(r) < F_{Po}(r)$ while the reverse indicates a more regular, or inhibited, pattern. *spatstat* calculates and plots the theoretical $F_{Po}(r)$ as well as border corrected and Kaplan-Meier estimates, denoted by theo, rs and km respectively.

2. The nearest-neighbour distance function, $G(r)$, is defined similarly to $F(r)$ except now $u$ is a data point and $x_i$ is another data point. Again an unbiased estimator for $G(r)$ can be
obtained and is denoted by $\hat{G}(r)$.

The nearest-neighbour distance function for a homogeneous Poisson process of intensity $\lambda$ has the same definition as $F_{P\lambda}(r)$, that is,

$$G_{P\lambda}(r) = 1 - e^{-\lambda\pi r^2}$$  \hspace{1cm} (6.3)

However this time the interpretation of a comparison of $\hat{G}(r)$ with $G_{P\lambda}(r)$ is reversed: a clustered pattern is indicated when $\hat{G}(r) > G_{P\lambda}(r)$ and a regular pattern otherwise.

`spatstat` calculates and plots the theoretical $G_{P\lambda}(r)$ as well as border corrected and Kaplan-Meier estimates, denoted by `theo`, `rs` and `km` respectively.

3. Ripley's $K$ function is used as a summary of pairwise distances. $K(r)$ is defined for stationary processes so that $\lambda K(r)$ equals the expected number of other points within radius $r$ of an arbitrarily chosen point. Again an edge correction is applied to the estimator $\hat{K}(r)$. For a Poisson process $K_{P\lambda}(r) = \pi r^2$, regardless of the intensity.

A clustered pattern would be indicated whenever $\hat{K}(r) > \pi r^2$ and a regular pattern whenever $\hat{K}(r) < \pi r^2$.

`spatstat` calculates and plots the theoretical $K_{P\lambda}(r)$ as well as border corrected, translation corrected and Ripley isotropic corrected estimates, denoted by `theo`, `border`, `trans` and `iso` respectively.

4. The $J$ function is defined in terms of $F$ and $G$ functions:

$$J(r) = \frac{1 - G(r)}{1 - F(r)}$$  \hspace{1cm} (6.4)

For Poisson processes, since $F(r) = G(r)$, we have $J(r) = 1$ which is a useful comparison.

`spatstat` calculates and plots the theoretical $J_{P\lambda}(r)$ as well as border corrected, Kaplan-Meier and uncorrected estimates, denoted by `theo`, `rs`, `km` and `un` respectively. These estimates just use the corresponding figures from the estimated $F$ and $G$ functions.

Details of these edge correction methods can be found in Illian et al. [15].

It is important to remember that the four distance functions outlined above apply only to homogeneous processes. These functions may suggest that the process is clustered or regular when in fact it is non-homogeneous. These functions can be modified to handle non-homogeneity (Baddeley et al. (2000) [3]).

### 6.5 Example A: Simulated Poisson point pattern

The `spatstat` command `rpoispp(50)` was used to generate an homogeneous Poisson point pattern with intensity 50 on the unit square. Figure 6.8 shows the resulting 55 points.
6.5. EXAMPLE A: SIMULATED POISSON POINT PATTERN

Figure 6.8: Plot of simulated Poisson(50) data

6.5.1 Tests for Complete Spatial Randomness

Figure 6.9 shows the spatstat output for quadrat.test after specifying 4 quadrats in both horizontal and vertical directions. Under the assumption of homogeneity each quadrat is assumed to contain \( \frac{55}{16} = 3.4375 \) points. This figure is shown together with the actual count and the Pearson residual (obtained by dividing the raw residuals by the square root of the fitted conditional intensity). The Pearson residuals are standardised, in the sense that if the model (true and fitted) is Poisson, then the sum of the Pearson residuals in a spatial region B has variance equal to the area of B (refer spatstat documentation). The value of the test statistic is 15.10909 with a \( p \)-value of 0.4436, thus indicating that the pattern is homogeneous.

Two Kolmogorov-Smirnov tests were performed, one using the test function \( f(x, y) = x + y \) and the other using \( f(x, y) = xy \). The plots of the actual and expected values of these functions for the data are shown in Figures 6.10(a) and 6.10(b) respectively. They indicate that the actual function values closely match the expected values and the \( p \)-values of 0.18 and 0.47 do not lead to rejection of the hypothesis that the point pattern is random.

Tests using distance methods

Figure 6.11 shows the standard spatstat presentation of the four distance methods in one plot. Although rather difficult to see, they do show that the estimates of the \( F \) function closely follow the theoretical curve while the estimates of the remaining three methods agree with the theoretical curves up to radius 0.06. Thereafter there is some divergence with the \( J \) function estimates being the most divergent.

The \( G \) and \( K \) functions are shown separately in Figures 6.12(a) and 6.12(b) respectively. Note that Figure 6.12(b) shows information for larger radii than Figure 6.11.
There is nothing in these four plots to indicate that the simulated data do not come from a random process.

Figure 6.9: Simulated Poisson data and output for quadrat test

Figure 6.10: KS test using (a) $f(x, y) = x + y$ and (b) $f(x, y) = xy$
Four summary functions for Poisson50.

![F function](image1)

![G function](image2)

![J function](image3)

![K function](image4)

Figure 6.11: Plots of \(F\), \(G\), \(J\) and \(K\) functions, using `spatstat`, for Poisson(50) simulated data with various types of border correction shown on the vertical axes. The colours used in order are black, red, green and blue.

Figure 6.12: Poisson(50) simulated data: (a) \(G\) function and (b) \(K\) function.
6.5.2 Resampling the simulated Poisson point pattern

The simulated point pattern was then lifted, resampled 1000 times and inverted using the algorithm of Section 6.3.

Figure 6.13 shows the bounded Voronoi diagram for the data after specifying the window vertices as those of the unit square. These vertices, plus the points generated on the boundary, are the fixed points used in the lifting algorithm. Figure 6.14(a) shows the Delaunay triangulation after 35 points have been removed while 6.14(b) shows the triangulation after all points have been removed.

The differences between the location of the deleted points and the mean location of their neighbours at each level are shown in Figure 6.15(a) for the x and y coordinates separately. There is no evidence of any trend in either the x or y differences. Figure 6.15(b) shows the corresponding autocorrelations for the x and y differences and we can conclude that there is no autocorrelation present. Therefore it is safe to resample these differences.

The resampling procedure is judged to be a success if the surrogate point patterns could be possible realisations of the same process that produced the data. To test this we use the diagnostics.
6.5. EXAMPLE A: SIMULATED POISSON POINT PATTERN

Figure 6.15: Poisson(50) point pattern: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations outlined in Section 6.4; that is, each surrogate point pattern should exhibit similar overall location to the data and the relationship between points should be similar. Note that each surrogate pattern has exactly the same number of points as the data. Figure 6.16 shows the data and one of the surrogates for comparison.

Figures 6.17(a) and 6.17(b) show histograms of the number of points within each of the 16 quadrats used in the chi-square test of randomness, for the 1000 surrogates, together with the corresponding data value. Note that the quadrats are referenced commencing at the bottom left corner and moving vertically so that cell (2, 4) is in the second column and top row, referring to the square with $x$ limits (0.25, 0.50) and $y$ limits (0.75, 1). With the exception of the first two quadrats, all of the histograms show the surrogate count values straddle the data counts. There appears to have been some movement between the first two quadrats, since the surrogate counts are above the data in the first quadrat and below in the second.

The quadrat.test was conducted for each surrogate and the mean test statistic was 12.14 with a standard deviation of 3.23. The critical value at $\alpha = 0.05$ and df=15 (16 quadrats) is 25.00 and
Figure 6.17: Histograms of number of counts in specified cells for resampled Poisson(50) point pattern (a) cells (1, 1) to (2, 4) and (b) cells (3, 1) to (4, 4)

Table 6.5: \( p \)-values for Kolmogorov-Smirnov tests on surrogate Poisson(50) point patterns

<table>
<thead>
<tr>
<th>Statistic</th>
<th>( f(x, y) = x + y )</th>
<th>( f(x, y) = xy )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.0745</td>
<td>0.0967</td>
</tr>
<tr>
<td>First quartile</td>
<td>0.2717</td>
<td>0.3870</td>
</tr>
<tr>
<td>Median</td>
<td>0.3643</td>
<td>0.5122</td>
</tr>
<tr>
<td>Mean</td>
<td>0.3643</td>
<td>0.5145</td>
</tr>
<tr>
<td>Third quartile</td>
<td>0.4571</td>
<td>0.6361</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.7651</td>
<td>0.9594</td>
</tr>
</tbody>
</table>
hence on this basis none of the surrogates would have rejected the randomness hypothesis.

The Kolmogorov-Smirnov test was used again with functions $f(x, y) = x + y$ and $f(x, y) = xy$. The summary of the 1000 $p$-values is given in Table 6.5 from which we can again conclude that none of the surrogates would have rejected the null hypothesis of randomness.

Each of the four distance methods was obtained for the first 100 surrogates from which the mean curves and the upper and lower limits, using the 95% envelope, were calculated.

Figure 6.18(a) shows the mean of the estimated $F$ function plots together with the upper and lower limits, the theoretical function and the estimated data $F$ function. The estimated $F$ functions for the surrogates are above the theoretical function for radii greater than 0.03 whereas the data follows the theoretical up to radius 0.7. As described in Section 6.4, an empty space function greater than the Poisson standard indicates a more regular process so we can conclude that the surrogates exhibit slightly more regularity than the data.

Figure 6.18(b) shows that the theoretical $G$ function and the Kaplan-Meier estimator of the data $G$ function lie entirely within the envelope calculated from the surrogates’ $G$ functions. However the theoretical function is always greater than the mean curve, again indicating a more regular pattern. This conclusion can countered by noting that the mean curve closely follows the data Kaplan-Meier estimator and hence if the latter agreed more with the theoretical it could be conjectured that the mean curve would do likewise.

Figure 6.19(a) shows the corresponding curves for the $J$ function and appear to be too variable to be of much use.

Figure 6.19(b) shows the results for the $K$ function, using the isotropic corrected estimator with the data. As with the $G$ function, the theoretical curve and data estimator lie within the envelope created by the surrogates’ $K$ function estimators, with the theoretical generally lying close to the upper limit. Again the mean curve approximates the data estimator.

To summarise the results of an analysis of the distance methods:

1. The $J$ function exhibits considerable variation among the surrogates.
2. The $F$ function indicates that the surrogates could be slightly more regular than the data.
3. The $G$ and $K$ functions show that the surrogates could be replicates from the process that produced the data.
CHAPTER 6. WAVELET RESAMPLING OF 2-DIMENSIONAL POINT PATTERNS

Figure 6.18: Estimated (a) $F$ and (b) $G$ functions for Poisson surrogates

Figure 6.19: Estimated (a) $J$ and (b) $K$ functions for Poisson surrogates
6.6 Example B: Simulated clustered point pattern

A Poisson cluster pattern is obtained by generating parent points from a Poisson process and then centering daughter points around each parent. A Neyman-Scott process is a type of Poisson cluster process where each daughter point is independently located around the parent. Furthermore, a Thomas process is a special type of Neyman-Scott process with the following characteristics:

(a) the parent points are replaced by the daughter points;

(b) the number of daughter points in a cluster is Poisson distributed;

(c) the daughter points have isotropic Gaussian displacements around the parent point. [4]

A simulated Thomas pattern was obtained using the \texttt{rThomas} command in \texttt{spatstat} with the following values specified for the variables:

- intensity of the Poisson process of cluster centres = 10

- standard deviation of the displacement of a point from its cluster centre = 0.1

- expected number of points per cluster = 5

- the window is the unit square.

This simulation yielded 50 points as shown in Figure 6.20(a). Note that there are two points just inside the boundary (one has an \(x\) coordinate of 0.993645 while the other has a \(y\) coordinate of 0.998348). Also there are no points in a relatively large section of the bottom left corner.

6.6.1 Tests for Complete Spatial Randomness

Figure 6.20(b) shows the \texttt{spatstat} output for \texttt{quadrat.test} after specifying 4 quadrats in both horizontal and vertical directions. Under the assumption of homogeneity each quadrat is assumed to contain \(\frac{50}{16} = 3.125\) points. The value of the test statistic is 33.84 with a \(p\)-value of 0.0036, thus indicating that the pattern is non-homogeneous.

Two Kolmogorov-Smirnov tests were performed, one using the test function \(f(x, y) = x + y\) and the other using \(f(x, y) = xy\). The plots of the actual and expected values of these functions for the data are shown in Figures 6.21(a) and 6.21(b) respectively. They indicate that the actual function values do not match the expected values. The \(p\)-value of 0.054 for \(f(x, y) = x + y\) is marginal using \(\alpha = 0.05\) and the \(p\)-value of 0.029 for \(f(x, y) = xy\) leads to rejection of the hypothesis that the point pattern is random.
CHAPTER 6. WAVELET RESAMPLING OF 2-DIMENSIONAL POINT PATTERNS

Tests using distance methods

Figure 6.22 shows the standard spatstat presentation of the four distance methods in one plot. Although rather difficult to see, they show that the estimates of the \( F \) function closely follow the theoretical curve while the estimates of the remaining three methods differ substantially from the theoretical curves.

The \( G \) and \( K \) functions are shown separately in Figures 6.23(a) and 6.23(b) respectively. Again the \( K \) function is plotted for larger radii than in Figure 6.22.

We would have to conclude from these four plots that the simulated data do not come from a random process.

Figure 6.21: KS test using (a) \( f(x, y) = x + y \) and (b) \( f(x, y) = xy \)
Four summary functions for Thomas data.

Figure 6.22: Plots of $F$, $G$, $J$ and $K$ functions for simulated Thomas pattern

Figure 6.23: Estimated (a) $G$ and (b) $K$ functions for simulated Thomas pattern
6.6.2 Resampling the simulated Thomas point pattern

The bounded Voronoi diagram for the data is shown in Figure 6.24. Figure 6.25 shows the Delaunay triangulation after 25 points have been removed while Figure 6.26 shows the triangulation after the removal of all 50 points.

The differences between the location of the deleted points and the mean location of their neighbours at each level are shown in Figure 6.27(a) for the $x$ and $y$ coordinates separately. There is no evidence of any trend in either the $x$ or $y$ differences. However there a couple of differences
in both coordinates that are relatively larger than the remainder. Figure 6.27(b) shows the autocorrelations for the $x$ and $y$ differences and we can conclude that autocorrelation is not present. Therefore it is safe to resample these differences.

Figures 6.28(a) and 6.28(b) show the data and one of the surrogates for comparison.

Figure 6.27: Thomas pattern: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations

Figure 6.28: Plot of (a) simulated Thomas pattern and (b) a typical surrogate

Figures 6.29(a) and 6.29(b) show histograms of the number of points within each of the 16 quadrats used in the chi-square test of randomness, for the 1000 surrogates, together with the corresponding data value. All of the histograms show the surrogate count values straddle the data counts. Sometimes the modal count for the surrogates is the same as the data count while other times there is a difference of at most 2. Therefore it appears that the surrogates have maintained the general position of the points in relation to the window.
Table 6.6: p-values for Kolmogorov-Smirnov tests on surrogate Thomas point patterns.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$f(x, y) = x + y$</th>
<th>$f(x, y) = xy$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.0010</td>
<td>0.0001</td>
</tr>
<tr>
<td>First quartile</td>
<td>0.0097</td>
<td>0.0045</td>
</tr>
<tr>
<td>Median</td>
<td>0.0183</td>
<td>0.0107</td>
</tr>
<tr>
<td>Mean</td>
<td>0.0276</td>
<td>0.0193</td>
</tr>
<tr>
<td>Third quartile</td>
<td>0.0343</td>
<td>0.0240</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.3277</td>
<td>0.2249</td>
</tr>
</tbody>
</table>

The quadrat.test was conducted for each surrogate and the mean test statistic was 36.19 with a standard deviation of 5.88. The minimum was 19.12 and the lower quartile was 32.56. The critical value at $\alpha = 0.05$ and df=15 (16 quadrats) is 25.00 and hence on this basis at least 75% of the surrogates would have rejected the randomness hypothesis.

The Kolgogorov-Smirnov test was used again with functions $f(x, y) = x + y$ and $f(x, y) = xy$. The summary of the 1000 p-values is given in Table 6.6. With both functions we see that over 75% of the p-values are less than 0.05. Therefore we can conclude that a majority of the surrogates reject the hypothesis of Complete Spatial Randomness.

Each of the four distance methods were obtained for the first 100 surrogates from which the mean curves and the 95% envelope were calculated.

Figure 6.30(a) shows the mean of the estimated $F$ function plots together with the upper and lower limits, the theoretical Poisson function and the estimated data $F$ function. The estimated $F$ functions for the surrogates are below the theoretical function for radii greater than 0.05. The mean $F$ function is above the data $F$ function up to radius 0.08 and below thereafter. As described in Section 6.4, an empty space function less than the Poisson standard indicates a clustered process so we can conclude that the surrogates exhibit clustering.

Figure 6.30(b) shows that the theoretical Poisson $G$ function and the Kaplan-Meier estimator of the data $G$ function lie entirely within the envelope calculated from the surrogates’ $G$ functions. However the theoretical function is less than the mean curve up to radius 0.10, also indicating a clustered pattern. Again the mean curve lies either side of the data Kaplan-Meier estimator with the cross-over again occurring at about radius 0.08.

Figure 6.30(c) shows the corresponding curves for the $J$ function with the mean curve and data estimator quickly dropping away from the Poisson theoretical value of 1.

Figure 6.30(d) shows the results for the $K$ function, using the isotropic corrected estimator for the data. The mean of the surrogates’ $K$ function estimator quickly rises above the theoretical Poisson function as the radius increases, thus indicating that the surrogates are clustered. The mean curve approximates the data estimator.

To summarise the results of an analysis of the distance methods:

1. The functions tells us the data and surrogates could come from the same model.
Figure 6.29: Histograms of number of counts in specified cells for resampled Thomas point pattern (a) cells (1, 1) to (2, 4) and (b) cells (3, 1) to (4, 4)
Figure 6.30: Estimated (a) $F$ function, (b) $G$ function, (c) $J$ function and (d) $K$ function for Thomas surrogates
2. The $F$, $G$ and $K$ functions indicate that the surrogates are clustered.

### 6.6.3 Thomas process parameter estimation

The diagnostics show that the surrogates appear to be clustered in a similar way to the data point pattern. We can take this analysis a stage further as spatstat has a function, `thomas.estK`, that estimates the parameters of the Thomas process $K$ function. If the intensity of the parent points is $\kappa$, the number of daughter points around each parent is a Poisson random variable with mean $\mu$ and the position of the daughter points are independent and isotropically Gaussian distributed with standard deviation $\sigma$, then the theoretical $K$-function for the Thomas process is

$$K(r) = \pi r^2 + \frac{1}{\kappa} \left( 1 - e^{-\frac{r^2}{4\sigma^2}} \right)$$  \hspace{1cm} (6.5)

spatstat uses the Method of Minimum Contrast to estimate $\kappa$ and $\sigma^2$. Since the overall theoretical intensity, $\lambda$, satisfies the relationship $\lambda = \kappa \times \mu$ and can be estimated from the data we can then find an estimate for $\mu$.

`thomas.estK` requires starting values for $\kappa$ and $\sigma^2$. The data values of 10 and 0.01 were used, respectively, to estimate the parameters for the first 100 resampled point patterns. These estimated parameters vary widely, as shown in the histograms in Figure 6.31 and by the table of means and standard deviations in Table 6.7. `thomas.estK` estimated the parameters for the data to be $\kappa=9.4$, $\sigma=0.12$ and $\mu=5.32$, which were close to the specified values of (10, 0.1, 5). However the corresponding estimates for the first two surrogates were (6.86, 0.12, 7.28) and (0.40, 0.70, 125.43). Plots of the theoretical $K$ function for these estimates of $\kappa$ and $\sigma$ are shown in Figure 6.32. It is obvious that there is little difference in the $K$ function for vastly different parameter values.

To analyse this function further, Figure 6.33 shows $K(r) - \pi r^2$ ie the part of the Thomas $K$-function additional to the Poisson function.

The top three plots of Figure 6.33 use $\kappa$ as the independent variable, from 0 to 5, with three values of $\sigma$ (0.1, 0.2, 0.5) for three different radii (0.05, 0.1, 0.15). Note the different vertical scale for each subplot. For fixed radius and $\kappa > 2$ it is apparent that there is little difference in $K(r) - \pi r^2$ when $\sigma$ increases from 0.1 to 0.5 or for any change in $\kappa$. Therefore the additional component to the Poisson $K$ function is relatively invariant to changes in either $\sigma$ or $\kappa$ for a fixed radius and $\kappa > 2$.

The lower three plots of Figure 6.33 uses $\sigma$ as the independent variable, from 0 to 0.5, with three values of $\kappa$ (1, 2, 5) for three different radii (0.05, 0.1, 0.15). Again note the different vertical scales and these scales are larger than the plots with $\kappa$ as the variable, highlighting the variations between function values for small values of $\sigma$ as $\kappa$ increases.
Table 6.7: Parameter estimates for surrogate Thomas point patterns

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$\kappa$</th>
<th>$\sigma$</th>
<th>$\mu$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>5.19</td>
<td>0.49</td>
<td>107.13</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>5.17</td>
<td>0.50</td>
<td>157.00</td>
</tr>
</tbody>
</table>

Figure 6.31: Histograms of parameter estimates: (left) $\kappa$, (centre) $\sigma$ and (right) $\mu$ for Thomas surrogates

Figure 6.32: Theoretical Thomas $K$ function with $(\kappa, \sigma) = (9.40, 0.12)$ (solid line) and $(0.33, 0.80)$ (dashed line)
6.7 Example C: Simulated Strauss point pattern

The Strauss process is an example of a pair-wise interaction process that is defined in terms of parameters that reflect inhibition.

If \( X \) is the point pattern consisting of the points \( \{x_1, ..., x_n\} \) then the probability density is given in spatstat by

\[
    f(X) = \alpha \beta^n(X) \gamma^s(X)
\]

(6.6)

where \( \alpha \) is a normalising constant, \( \beta \) is the intensity parameter, \( \gamma \) is the interaction parameter \( (0 \leq \gamma \leq 1) \), \( n(X) \) is the number of points in the pattern and \( s(X) \) is the number of distinct unordered pairs of points that are closer than \( R \) units apart where \( R \) is the interaction radius.

If \( \gamma = 1 \) then the process is Poisson with intensity \( \beta \); if \( \gamma = 0 \) then the process is termed 'hard core' where no points lie closer to each other than the specified interaction radius.

The spatstat command \texttt{rStrauss(50, 0.5, 0.05, owin())} was used to generate a Strauss point pattern on the unit square with intensity 50, interaction parameter 0.5 and interaction radius 0.05.

Figure 6.34 shows the resulting 51 points.
CHAPTER 6. WAVELET RESAMPLING OF 2-DIMENSIONAL POINT PATTERNS

6.7.1 Tests for Complete Spatial Randomness

Figure 6.35 shows the spatstat output for quadrat.test after specifying 4 quadrats in both horizontal and vertical directions. Under the assumption of homogeneity each quadrat is assumed to contain \( \frac{51}{16} = 3.1875 \) points. The value of the test statistic is 8.294 with a \( p \)-value of 0.9115, thus indicating that the pattern is homogeneous.

Two Kolmogorov-Smirnov tests were performed, one using the test function \( f(x, y) = x + y \) and the other using \( f(x, y) = xy \). The plots of the actual and expected values of these functions for the data are shown in Figures 6.36(a) and 6.36(b) respectively. They indicate that the actual function values match the expected values. The \( p \)-values of 0.1625 for \( f(x, y) = x + y \) and 0.4847 for \( f(x, y) = xy \) lead to non-rejection of the hypothesis that the point pattern is random.

Tests using distance methods

Figure 6.37 shows the four distance methods. Each method shows that the function estimators follow the Poisson theoretical values up to radius 0.04 and diverge thereafter. The estimates for both the \( G \) and \( K \) functions lie underneath the theoretical values for radii greater than 0.04, indicating a more regular process that Poisson. We would have to conclude from these four plots, therefore, that the simulated data do not come from a random process.

Thus we come to different conclusions using the tests for CSR and distance methods.
6.7. EXAMPLE C: SIMULATED STRAUSS POINT PATTERN

Figure 6.35: Simulated Strauss point pattern: data and output for quadrat test

Figure 6.36: KS test for Strauss data using (a) \( f(x, y) = x + y \) and (b) \( f(x, y) = xy \)
Six summary functions for Strauss data.

Four summary functions for Strauss data.

**Figure 6.37:** Plots of \(F\), \(G\), \(J\) and \(K\) functions for simulated Strauss data

### 6.7.2 Resampling the simulated Strauss point pattern

The bounded Voronoi diagram for the data is shown in Figure 6.38(a). Figure 6.38(b) shows the Delaunay triangulation after 25 points have been removed while Figure 6.39 shows the triangulation after the removal of all 51 points.

The differences between the location of the deleted points and the mean location of their

neighbours at each level are shown in Figure 6.40(a) for the \(x\) and \(y\) coordinates separately. There is no evidence of any trend in either the \(x\) or \(y\) differences. However there a couple of differences in both coordinates that are relatively larger than the remainder. Figure 6.40(b) also shows the autocorrelations for the \(x\) and \(y\) differences and we can conclude that autocorrelation is not present. Therefore it is safe to resample these differences. Figures 6.41(a) and 6.41(b) show the data and one of the surrogates for comparison.

Figures 6.42(a) and 6.42(b) show histograms of the number of points within each of the 16 quadrats used in the chi-square test of randomness, for the 1000 surrogates, together with the
6.7. EXAMPLE C: SIMULATED STRAUSS POINT PATTERN

Figure 6.39: Delaunay triangulation for simulated Strauss data after removing all points

![Delaunay triangulation](image)

Figure 6.40: Strauss data: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations

(a) ![Graph](image)

(b) ![Graph](image)

corresponding data value. All of the histograms show the surrogate count values straddle the data counts. Sometimes the modal count for the surrogates is the same as the data count while other times there is a difference of 1. Therefore it appears that the surrogates have maintained the general position of the points in relation to the window.

The `quadrat.test` was conducted for 100 surrogates and the mean test statistic was 15.72 with a standard deviation of 2.14. The maximum was 22.10. The critical value at $\alpha = 0.05$ and df=15 (16 quadrats) is 25.00 and hence on this basis all of the surrogates would not have rejected the randomness hypothesis.

The Kolmogorov-Smirnov test was used again with functions $f(x, y) = x + y$ and $f(x, y) = xy$. The summary of the 1000 $p$-values is given in Table 6.8. At least 75% of the $p$-values for both functions are greater than 0.05. Therefore we can conclude that a majority of the surrogates do not reject the hypothesis of Complete Spatial Randomness.

Each of the four distance methods was obtained for the first 100 surrogates from which the mean curves and the upper and lower limits, using the 95% envelope, were calculated.
Table 6.8: $p$-values for Kolmogorov-Smirnov tests on surrogate Strauss point patterns

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$f(x, y) = x + y$</th>
<th>$f(x, y) = xy$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.0105</td>
<td>0.0311</td>
</tr>
<tr>
<td>First quartile</td>
<td>0.1199</td>
<td>0.2381</td>
</tr>
<tr>
<td>Median</td>
<td>0.1823</td>
<td>0.3489</td>
</tr>
<tr>
<td>Mean</td>
<td>0.2058</td>
<td>0.3657</td>
</tr>
<tr>
<td>Third quartile</td>
<td>0.2664</td>
<td>0.4571</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.8033</td>
<td>0.9153</td>
</tr>
</tbody>
</table>

Figure 6.43(a) shows the mean of the estimated $F$ function plots together with the upper and lower limits, the theoretical function and the estimated data $F$ function. The estimated $F$ functions for the surrogates are close to the theoretical Poisson function. The data $F$ function is above the upper limit for radii between approximately 0.08 and 0.12.

Figure 6.43(b) shows that the mean surrogates’ $G$ function closely follows the theoretical Poisson $G$ function but the Kaplan-Meier estimator of the data $G$ function, after following the theoretical curve up to radius 0.04, generally follows the lower limit.

Figure 6.44(a) shows the corresponding curves for the $J$ function with the mean curve and data estimator quickly climbing away from the theoretical value of 1.

Figure 6.44(b) shows the results for the $K$ function, using the isotropic corrected estimator with the data. The results are much the same as for the $G$ function.

To summarise the results of an analysis of the distance methods we note that all four functions indicate that the surrogates are similar to Poisson random point patterns and do not replicate the data. So in this case the resampling after lifting algorithm has produced point patterns that cannot be assumed to be possible replicates from the same model as the data. Further investigation can be conducted to analyse the effect of changing the interaction parameter and interaction radius.

Figure 6.41: Simulated Strauss pattern: (a) data and (b) a typical surrogate
Figure 6.42: Histograms of number of counts in specified cells for resampled Strauss point patterns
(a) cells (1, 1) to (2, 4) and (b) cells (3, 1) to (4, 4)
Figure 6.43: Strauss surrogates: estimated (a) $F$ function and (b) $G$ function

Figure 6.44: Strauss surrogates: estimated (a) $J$ function and (b) $K$ function
6.8 Example D: Strauss point patterns with other interaction parameters

We saw in Section 6.7 that the surrogate series didn’t have the same properties as the data simulated from a Strauss(50, 0.5, 0.05) point pattern. In this section the analysis is repeated with the interaction parameter, $\gamma$, taking the values 0.8 and 0.2. It was stated in Section 6.7 that if $\gamma = 1$ then the process is Poisson and if $\gamma = 0$ the process is hard core. Hence when $\gamma = 0.8$ the process should be similar to Poisson and when $\gamma = 0.2$ there should be few points closer together than the interaction radius.

Figures 6.45(a) and 6.45(b) show simulated data from a Strauss(50, 0.8, 0.05) process and one of the 100 surrogates respectively. Figures 6.46(a) - 6.47(b) show the corresponding plots of the $F$, $G$, $J$ and $K$ functions for the surrogates. We see that the mean $F$ and $G$ curves approximate the data estimator with the theoretical Poisson curves marginal in both cases. The estimated $K$ function from the data generally follows the lower limit of the surrogates’ envelope with the theoretical curve lower still for radii greater than approximately 0.06. So it appears that when $\gamma = 0.8$ the surrogates seem to replicate the data and exhibit non-Poisson characteristics.

Figures 6.48(a) and 6.48(b) show simulated data from a Strauss(50, 0.2, 0.05) process and one of the 100 surrogates respectively. Figures 6.49(a) - 6.50(b) show the corresponding plots of the $F$, $G$, $J$ and $K$ functions for the surrogates. All the $F$ curves lie close together and therefore imply similarity between data, surrogates and theoretical. However the curves for the $G$ function are distinctly different with the surrogates implying similarity to Poisson but the data clearly exhibiting the interaction radius of 0.05. The same behaviour is shown by the $K$ function.

We can conclude that when the interaction parameter reduces toward 0 the less able is the algorithm to produce surrogates that appear to replicate the data.
Figure 6.46: Strauss(50, 0.8, 0.05) surrogates: estimated (a) $F$ function and (b) $G$ function

Figure 6.47: Strauss(50, 0.8, 0.05) surrogates: estimated (a) $J$ function and (b) $K$ function

Figure 6.48: Simulated Strauss(50, 0.2, 0.05) pattern: (a) data and (b) one surrogate
6.8. EXAMPLE D: STRAUSS POINT PATTERNS WITH OTHER INTERACTION PARAMETERS

Figure 6.49: Strauss(50, 0.2, 0.05) surrogates: estimated (a) $F$ function and (b) $G$ function

Figure 6.50: Strauss(50, 0.2, 0.05) surrogates: estimated (a) $J$ function and (b) $K$ function
Chapter 7

2-D Non-homogeneous Poisson data

7.1 Introduction

This chapter presents the results when the 2D wavelet lifting and resampling algorithm, as described in Chapter 6, is applied to non-homogeneous Poisson data where the intensity is some function of the location.

Here the intensity is limited to simple linear functions, one example having the $x$ coordinate as the only variable while a second example has both $x$ and $y$ coordinates as variables.

The non-homogeneous $K$ function and the Pair Correlation Function (PCF) were used to determine how well the procedure was able to produce point patterns that could have come from the same population as the data.

7.2 Intensity as a linear function of the $x$-coordinate

`spatstat` was used to generate data with intensity function $\lambda(x) = 50 + 100x$ with the commands:

```r
lamfun < - function(x,y) { 50 + 100 * x }
Y < - rpoispp(lamfun, 150, owin())
```

The resulting pattern, consisting of 111 points, is shown in Figure 7.1. The chi-squared test for Complete Spatial Randomness (CSR) using quadrat tests had a test statistic of 32.42 with 15 df, resulting in a $p$-value of 0.0056. Hence we can reject the null hypothesis that the point pattern is homogeneous.
7.2. INTENSITY AS A LINEAR FUNCTION OF THE X-COORDINATE

7.2.1 First-order analysis of surrogates

The lifting algorithm calculates differences between the co-ordinates of the points deleted at each stage and those of the mean position of the connected vertices. These differences are resampled and there exists the possibility of mixing of scales, which would occur if the differences showed a trend. Figure ?? shows the differences in the x and y co-ordinates and we can see that, except for the last few values, there is no trend in either the x or y differences.

There is slight autocorrelation at lag 2 in the differences between the x coordinates, as shown in Figure 7.2(b). Figure 7.3(a) gives the quadrat counts for the data while Figure 7.4 shows the distribution of the counts in each of 16 quadrats for 1000 surrogates. The mean quadrat counts for the surrogates are compared with the data quadrat counts in Table 7.1 and there are similar values for the quadrats in the first two rows. It is apparent that there is some movement between the quadrats in the final two rows.

Note that while the layout of Table 7.1 corresponds with that of Figure 7.4, it is the transpose of the quadrats shown in Figure 7.3(a).

The chi-squared test for CSR was performed for each surrogate. The minimum value of the test statistic was 19.45 with a lower quartile of 25.79 and a mean of 29.43, compared to a critical value of 24.9958 ($\alpha = 0.05$ with 15df). Hence the null hypothesis of CSR would be rejected for at least 75% of the surrogates.

Figure 7.5 shows the point pattern of the first four surrogates with the data superimposed. Also shown are plots of estimated intensity as derived from fitted models (see below).

The surrogates appear to be similar to the data. We can test this assumption by fitting models to the data and surrogates and comparing coefficients.

The spatstat command `ppm` was used to fit a simple model of the intensity to the data. The model was a first-degree polynomial in x and y such that the intensity is given by:

$$\lambda(x, y) = e^{\theta_0 + \theta_1 x + \theta_2 y}$$

(7.1)

Note that a vertical component is included in the model that wasn’t given in the intensity function that produced the data. It is not obvious from the data that the point locations are independent of y and therefore it is appropriate to include a vertical component in the initial model.

The relevant spatstat commands to fit the model and plot the estimated intensity, as shown in Figure 7.5, were:

```r
fit <- ppm(Y, polynom(x,y,1), Poisson())
lambda <- predict(fit,type='trend')
plot(lambda)
```

The coefficient estimates were ($\hat{\theta}_0 = 4.088, \hat{\theta}_1 = 1.090$ and $\hat{\theta}_2 = 0.055$). Figure 7.3(b) shows
Figure 7.1: Non-homogeneous Poisson data: $\lambda(x) = 50 + 100x$

Figure 7.2: Linear intensity function: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations

Table 7.1: Quadrat counts for data and mean quadrat counts (in brackets) for 1000 surrogates with intensity $\lambda(x) = 50 + 100x$

<table>
<thead>
<tr>
<th>$x \backslash y$</th>
<th>(0, 0.25)</th>
<th>(0.25, 0.5)</th>
<th>(0.5, 0.75)</th>
<th>(0.75, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0.25)</td>
<td>5 (5.6)</td>
<td>5 (5.8)</td>
<td>7 (4.5)</td>
<td>4 (2.9)</td>
</tr>
<tr>
<td>(0.25, 0.5)</td>
<td>2 (2.7)</td>
<td>2 (4.9)</td>
<td>4 (4.5)</td>
<td>7 (7.5)</td>
</tr>
<tr>
<td>(0.5, 0.75)</td>
<td>5 (8.3)</td>
<td>7 (4.5)</td>
<td>10 (5.4)</td>
<td>4 (2.9)</td>
</tr>
<tr>
<td>(0.75, 1)</td>
<td>15 (18.4)</td>
<td>5 (7.8)</td>
<td>16 (11.8)</td>
<td>9 (10.0)</td>
</tr>
</tbody>
</table>
7.2. INTENSITY AS A LINEAR FUNCTION OF THE X-COORDINATE

Figure 7.3: Non-homogeneous Poisson process: (a) quadrat counts for data and (b) estimated intensity using $\hat{\lambda}(x, y) = e^{4.088 + 1.090x + 0.055y}$

Table 7.2: Summary statistics for estimates of coefficients of model fitted to 100 surrogates for modelling $\lambda(x) = 50 + 100x$

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$\hat{\theta}_0$</th>
<th>$\hat{\theta}_1$</th>
<th>$\hat{\theta}_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>4.021</td>
<td>0.984</td>
<td>-0.668</td>
</tr>
<tr>
<td>First quartile</td>
<td>4.158</td>
<td>1.160</td>
<td>-0.459</td>
</tr>
<tr>
<td>Median</td>
<td>4.219</td>
<td>1.236</td>
<td>-0.389</td>
</tr>
<tr>
<td>Mean</td>
<td>4.215</td>
<td>1.236</td>
<td>-0.387</td>
</tr>
<tr>
<td>Third quartile</td>
<td>4.275</td>
<td>1.309</td>
<td>-0.319</td>
</tr>
<tr>
<td>Maximum</td>
<td>4.448</td>
<td>1.477</td>
<td>-0.057</td>
</tr>
</tbody>
</table>

the plot of the estimated intensity using this model.

Corresponding estimates were obtained for 100 surrogates. Figure 7.6 shows the resulting histograms from which we can see that the estimates fall in a relatively narrow range. Summary values of these estimates are given in Table 7.2.

The coefficient estimates for a model fitted to the data that did not include a vertical component were $(\hat{\theta}_0 = 4.088, \hat{\theta}_1 = 1.090)$. The corresponding mean estimates for the surrogates were similar $(\bar{\theta}_0 = 4.215, \bar{\theta}_1 = 1.236)$.

Taking the coefficients from the model fit to the data results in an estimated intensity function $\lambda(x) = e^{4.215+1.236x} = 67.69 \times 3.44^x$ compared to the specified function $\lambda(x) = 50 + 100x$.

Therefore it appears that the models fit to the surrogates are very similar to that fitted to the data and we can conclude, from this aspect, that the surrogates could be possible replicates from the population that produced the data.
Figure 7.4: Histograms of quadrat counts for 1000 surrogates: (a) quadrats 1 to 8 and (b) quadrats 9 to 16
7.2. INTENSITY AS A LINEAR FUNCTION OF THE X-COORDINATE

Figure 7.5: Plot of point locations (open circles) with superimposed data (filled circles) and estimated intensity for: (a) surrogates 1 and 2 (b) surrogates 3 and 4

Figure 7.6: Histograms of fitted coefficients for the model with 2-dimensional intensity function $\hat{\lambda}(x, y) = e^{(\theta_0 + \theta_1 x + \theta_2 y)}$ to estimate $\lambda(x) = 50 + 100x$
CHAPTER 7. 2-D NON-HOMOGENEOUS POISSON DATA

7.2.2 Second-order summary functions

To continue the analysis of the surrogates we can consider the non-homogeneous version of the \( K \) function, defined by

\[
\hat{K}_{\text{inhom}}(r) = \sum_i \sum_j \frac{1\{d_{ij} \leq r\} e(x_i, x_j, r)}{\lambda(x_i)\lambda(x_j)}
\]  

(7.2)

where \( d_{ij} \) is the distance between points \( x_i \) and \( x_j \), and \( e(x_i, x_j, r) \) is an edge correction factor. If \( \lambda(x) \) is not known an estimator, \( \hat{\lambda}(x) \), is obtained by predicting the intensity at location \( x \) from the fitted model. The isotropic edge correction is used in the calculations and defined as

\[
e(x_i, x_j, r) = \frac{1}{\text{area}(W)} g(x_i, x_j)
\]  

(7.3)

where \( g(x_i, x_j) \) is the fraction of the circumference of the circle with centre \( x_i \) and radius \( \|x_i - x_j\| \) which lies inside the window.

The spatstat commands are

\[
\text{lambda <- predict(fit, locations=Y, type=`trend´)}
\]

\[
\text{Ki <- Kinhom(Y, lambda, correction=`isotropic´)}
\]

where ‘fit’ refers to the fitted intensity function and ‘Y’ is the matrix of point locations.

The theoretical value of \( \hat{K}_{\text{inhom}}(r) \) for a non-homogeneous Poisson point pattern is \( \pi r^2 \).

Figure 7.7(a) shows the theoretical \( K_{\text{inhom}}(r) \) function for the data under consideration as well as an estimate from the data and the mean of the estimates from the surrogates plus the 95% envelope. It is clear that the two estimates are close to the theoretical function and that the 95% envelope is narrow.

It is also useful to consider the Pair Correlation Function (PCF) defined by

\[
g(r) = \frac{K'(r)}{2\pi r}
\]  

(7.4)

where \( K'(r) \) is the derivative of the relevant \( K \)-function. The theoretical value of \( g(r) \) for a stationary Poisson point process is 1. For non-homogeneous Poisson data the theoretical value is also 1 when \( g(r) \) is based on the derivative of the inhomogeneous \( K \) function, \( K_{\text{inhom}}(r) \).

spatstat calculates the PCF in a number of different ways according to the input parameters. The command \( \text{pcf(X)} \) will calculate the PCF based on the point pattern which uses a kernel smoothing method, while

\[
\text{Ki <- Kinhom(Y, lambda, correction=`isotropic´)}
\]

\[
\text{pcK <- -pcf(Ki)}
\]
7.3. INTENSITY FUNCTION IN X AND Y

Figure 7.7(b) shows the PCFs calculated on the point pattern and hence under the assumption of stationarity. Except for the drop off near radius = 0.24, the PCF from the data lies outside the 95% envelope of the PCFs obtained from the surrogates for radii in the interval (0.05, 0.10) and greater than 0.20. The theoretical value of 1 is generally outside the envelope in these intervals also. The data PCF has two distinct humps at radii 0.03 and 0.15; for other radii the PCF is close to 1. Values greater than 1 indicate clustering so the nonhomogeneity of the data is indicating that clusters are present. The mean of the PCFs from the surrogates is greater than 1 for radii less than 0.15 and shows a continual decline as the radius increases.

Alternatively, Figure 7.8(a) shows the PCFs based on the derivative of the inhomogeneous K function. The PCF from the data is more variable than in Figure 7.7(b) but appears to have high points near radius 0.03 and in (0.10, 0.15) as before. The mean PCF from the surrogates is close to the theoretical value of 1 throughout and the 95% envelope is more symmetrical than in Figure 7.7(b).

spatstat has the ability to compute simulation envelopes of summary functions, including the PCF and \( K_{inhom}(r) \). The former assumes Complete Spatial Randomness and therefore is not appropriate here. However it is instructive to compare the results with the 95% envelope shown previously. Figure 7.8(b) shows the simulation envelope for the PCF, relatively symmetrical about the theoretical value of 1, with the PCF from the data contained therein.

Figure 7.9 shows the simulation envelope for \( K_{inhom}(r) \), together with the estimated function from the data and the theoretical curve. A comparison of this plot with Figure 7.7(a) reveals that the envelope from the surrogates is narrower than from spatstat’s simulation.

Consideration of both \( K_{inhom}(r) \) and \( g(r) \) functions indicates that, from a second-order viewpoint, the surrogates could be realisations from the same population that produced the data.

7.3 Intensity function in x and y

We now repeat the analysis using an intensity function that depends upon both \( x \) and \( y \), specifically \( \lambda(x, y) = 200x + 200y \). The data, shown in Figure 7.10, consists of 193 points and the CSR chi-squared test statistic is 59.4352, giving a \( p \)-value of 3.153e-07 (15 df). Hence we can reject the hypothesis of homogeneity.

Figure 7.11(a) shows the differences in the \( x \) and \( y \) co-ordinates between the deleted points and the mean of their connected vertices. We can see that, except for the last few values, there is no trend in either the \( x \) or \( y \) differences. There is no autocorrelation in the differences between the \( x \) or \( y \) coordinates, as shown in Figure 7.11(b).

Figure 7.12(a) gives the quadrat counts for the data while Figure 7.13 shows the distribution
Figure 7.7: Non-homogeneous Poisson process (a) non-homogeneous $K$ function and (b) Pair Correlation Function estimated from data

Figure 7.8: Non-homogeneous Poisson process (a) Pair Correlation Function estimated from $K_{inhom}$ and (b) Pair Correlation Function simulation envelope

Figure 7.9: $K_{inhom}(r)$ simulation envelope
7.3. INTENSITY FUNCTION IN X AND Y

Figure 7.10: Non-homogeneous Poisson data: $\lambda(x, y) = 200x + 200y$

Figure 7.11: 2D intensity function: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations

Figure 7.12: Non-homogeneous Poisson process: (a) quadrat counts for data and (b) estimated intensity using $\hat{\lambda}(x, y) = e^{4.028 + 1.163x + 1.095y}$
Table 7.3: Quadrat counts for data and mean quadrat counts for 1000 surrogates (in brackets) with $\lambda(x, y) = 200x + 200y$

<table>
<thead>
<tr>
<th>$x \backslash y$</th>
<th>(0, 0.25)</th>
<th>(0.25, 0.5)</th>
<th>(0.5, 0.75)</th>
<th>(0.75, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0.25)</td>
<td>2 (3.2)</td>
<td>5 (5.3)</td>
<td>10 (11.0)</td>
<td>7 (8.5)</td>
</tr>
<tr>
<td>(0.25, 0.5)</td>
<td>6 (11.6)</td>
<td>11 (11.7)</td>
<td>18 (13.1)</td>
<td>12 (10.5)</td>
</tr>
<tr>
<td>(0.5, 0.75)</td>
<td>10 (11.0)</td>
<td>6 (5.4)</td>
<td>14 (7.3)</td>
<td>22 (17.7)</td>
</tr>
<tr>
<td>(0.75, 1)</td>
<td>11 (15.6)</td>
<td>20 (21.2)</td>
<td>11 (11.2)</td>
<td>28 (28.5)</td>
</tr>
</tbody>
</table>

Table 7.4: Summary statistics for estimates of coefficients of model fitted to 100 surrogates for modelling $\lambda(x, y) = 200x + 200y$

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$\hat{\theta}_0$</th>
<th>$\hat{\theta}_1$</th>
<th>$\hat{\theta}_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>4.135</td>
<td>0.985</td>
<td>0.386</td>
</tr>
<tr>
<td>First quartile</td>
<td>4.262</td>
<td>1.114</td>
<td>0.554</td>
</tr>
<tr>
<td>Median</td>
<td>4.307</td>
<td>1.161</td>
<td>0.600</td>
</tr>
<tr>
<td>Mean</td>
<td>4.305</td>
<td>1.168</td>
<td>0.603</td>
</tr>
<tr>
<td>Third quartile</td>
<td>4.344</td>
<td>1.220</td>
<td>0.643</td>
</tr>
<tr>
<td>Maximum</td>
<td>4.435</td>
<td>1.353</td>
<td>0.962</td>
</tr>
</tbody>
</table>

of the counts in each of 16 quadrats for 1000 surrogates. Table 7.3 shows quadrat counts for the data and the mean of the counts from all 1000 surrogates. In most quadrats the mean count is close to the data while a few of them show discrepancies.

The chi-squared test for CSR was performed for each surrogate. The minimum value of the test statistic was 40.53 (mean 53.80) compared to a critical value of 24.9958 ($\alpha = 0.05$ with 15 df). Hence the null hypothesis of CSR would be rejected for all the surrogates. Again the Poisson model was fitted to the data, producing estimated coefficients ($\hat{\theta}_0 = 4.028$, $\hat{\theta}_1 = 1.163$ and $\hat{\theta}_2 = 1.095$). Figure 7.12(b) shows the plot of the estimated intensity using this model.

Figure 7.14 shows the point pattern of first four surrogates with the data superimposed together with plots of estimated intensity as derived from fitted models. A comparison of the surrogates’ point patterns and intensity contour plots with those of the data shows that the surrogates are similar to the data.

Corresponding estimates were obtained for 100 surrogates. Figure 7.15 shows the resulting histograms from which we can see that the estimates again fall in a relatively narrow range. Summary values of these estimates are given in Table 7.4. Of the estimate obtained from the data, only $\hat{\theta}_1$ falls within the range of the estimates obtained from the surrogates.

Figure 7.16(a) again shows the theoretical $K_{inhom}(r)$ function for the data as well as an estimate from the data and the mean of the estimates from the surrogates plus the 95% envelope. As before, the two estimates are close to the theoretical function and the 95% envelope is narrow.

Figures 7.16(b) and 7.17(a) show the PCFs based on the data point pattern and the derivative of $K_{inhom}(r)$ respectively. The following points can be observed in Figure 7.16(b):

- the PCF for the data exhibits a wavy pattern, with $g(r) > 1$ for radii less than approximately 0.24, indicating clustering
Figure 7.13: Histograms of quadrat counts for 1000 surrogates: (a) quadrats 1 to 8 and (b) quadrats 9 to 16.
CHAPTER 7. 2-D NON-HOMOGENEOUS POISSON DATA

Figure 7.14: Plot of point locations (open) with superimposed data (filled) and estimated intensity for: (a) surrogates 1 and 2 (b) surrogates 3 and 4

Figure 7.15: Histograms of fitted coefficients for modelling intensity $\lambda(x, y) = 200x + 200y$ using $\hat{\lambda}(x, y) = e^{(\theta_0 + \theta_1 x + \theta_2 y)}$
7.4. CONCLUSION

- the mean PCF from the surrogates does not appear to be a smoothed version of the data PCF as it continuously declines, passing through 1 at radius approximately 0.14
- the data PCF is continuously contained within the 95% envelope only for radii less than approximately 0.07

In contrast, the mean of the PCFs from the surrogates in Figure 7.17(a) is close to 1 for all radii and the 95% envelope is relatively symmetrical about 1. The PCF from the data is more variable than in Figure 7.16(b) and appears to be slightly negative overall.

Figure 7.17(b) shows spatstat’s simulation envelope assuming CSR. As before, the envelope is relatively symmetrical about the theoretical value of 1 and the data PCF falls outside the envelope for many radii. Figure 7.18 shows the $K_{inhom}(r)$ envelope, again wider than the 95% envelope from the surrogates. Note that $\hat{K}_{inhom}(r)$ from the data closely follows the lower level of the envelope.

![Figure 7.16: Non-homogeneous Poisson process: (a) non-homogeneous $K$ function and (b) Pair Correlation Function estimated from data](image)

7.4 Conclusion

This chapter has analysed the surrogates after lifting two non-homogeneous Poisson point patterns. The effectiveness of the resampling procedure has been tested by examining the quadrat counts, the parameters of the fitted intensity model, the non-homogeneous $K$ function and the pair correlation function. All the tests for the first pattern, with a linear intensity function, indicate that the surrogates could have come from the same populations that produced the data. However the same conclusion cannot be made regarding the second pattern with intensity $\lambda(x, y) = 200x + 200y$ and therefore the algorithm does not appear to be suitable for non-homogeneous patterns. This is examined further in Chapter 9.
Figure 7.17: Non-homogeneous Poisson process: Pair Correlation Function (a) estimated from the derivative of $K_{inhom}(r)$ and (b) spatstat’s simulation envelope assuming CSR

Figure 7.18: $K_{inhom}(r)$ simulation envelope
Chapter 8

2-D Matern cluster process example

8.1 Introduction

The paper by Loh and Stein [19] simulated a Matern cluster process to test their method of bootstrapping point processes. This chapter uses the wavelet lifting approach to resampling a Matern cluster pattern with the same parameters.

8.2 The Matern cluster process

This point process is a version of the Neymann-Scott process. A Poisson process of intensity $\kappa$ generates parent points that are then replaced by a Poisson distributed number of daughter points with mean $\mu$, uniformly and independently distributed on a disc of fixed radius $R$ centered on the parent.

The theoretical $K$ function of the Matern cluster process is

$$K(r) = \pi r^2 + \frac{1}{\kappa} h\left(\frac{r}{2R}\right)$$  (8.1)

where

$$h(z) = 2 + \frac{1}{\pi} [8z^2 - 4\arccos(z) - 2\arcsin(z) + 4z\sqrt{(1 - z^2)^3} - 6z\sqrt{1 - z^2}]$$  (8.2)

for $z \leq 1$, and $h(z) = 1$ for $z > 1$. The theoretical intensity of the Matern cluster process is $\lambda = \kappa\mu$. 149
CHAPTER 8. 2-D MATERN CLUSTER PROCESS EXAMPLE

Figure 8.1: Matern cluster pattern: (a) data and (b) four surrogates

Figure 8.2: Matern cluster: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations
8.3 Simulated data

The parameters used in the simulation were $\kappa = 25$, $\mu = 10$ and $R = 0.1$, giving 250 as the expected number of points. The resulting pattern consisting of 198 points, generated using `spatstat` [4], is shown in Figure 8.1(a). There are four circular or rectangular clusters separated by regions containing no points.

The point pattern was lifted, using the algorithm described in Chapter 6, and the differences resampled before inverting to produce a surrogate pattern. This procedure was repeated 100 times. Four of the surrogate point patterns are shown in Figure 8.1(b) and exhibit similarity to the data but with some events now located in previously empty space.

The vectors of differences in the $x$ and $y$ coordinates that are resampled are shown in Figure 8.2(a). In both cases the magnitudes of approximately the first 100 differences are similar but exhibit greater diversity thereafter. No attempt has been made when resampling to adjust for the variation in magnitude.

Figure 8.2(b) shows the ACFs for the differences and it appears that there is slight autocorrelation at lag 3 for the difference in $x$ co-ordinates and at lag 2 for the $y$ co-ordinates.

Figures 8.3(a) to 8.4(b) show the estimated $F$, $G$, $J$ and $K$ functions together with the theoretical Poisson functions for stationary data. Also shown in Figure 8.4(b) is the theoretical $K$ function for the Matern cluster process using the simulation parameters $\kappa = 25$ and $R = 0.1$.

Looking at Figure 8.3(a) first we see that the mean of the surrogates’ estimated $F$ function is close to the Kaplan-Meier estimator obtained from the data for radii less than 0.04 but lies above thereafter. The estimated curves from both the data and surrogates lie well below the theoretical Poisson function for stationary data, thus giving clear evidence of clustering.

Figure 8.3(b) shows the relevant graphs for the $G$ function and we see that the estimator from
the data lies within the 95% envelope from the surrogates up to $r = 0.03$. Thereafter it lies close to the upper level of the envelope. Note that the $F$ function is plotted up to $r = 0.10$ which is the specified radius in the simulation, whereas the $G$ function is only plotted to $r = 0.04$. These maximum radii are chosen by spatstat but can be overridden if desired.

Figure 8.4(a) shows the graphs for the $J$ function and we can see that the surrogates follow the data up to $r = 0.02$; thereafter the $J$ function for the data decreases to 0 at $r = 0.08$ whereas the mean of the surrogates’ $J$ function stabilizes around 0.3.

Figure 8.4(b) shows the graphs for the $K$ function. Note that these graphs are plotted for radii up to 0.25, giving estimated $K$ function values of a little over 0.2. The mean of the surrogates’ $K$ functions is virtually identical to that from the data up to $r = 0.1$; it then moves lower until the upper level of the 95% envelope lies below the data $K$ function for radii (0.15, 0.20). Thereafter they appear to be very similar. Also shown is the theoretical Matern $K$ function (8.1) using $\kappa = 25$ and $R = 0.1$ as specified in the simulation. This function lies beneath both the data and surrogates’ $K$ functions for radii less than 0.2 and above thereafter.

The function `matclust.estK` in spatstat estimates the parameters $\kappa$ and $R$ by the method of Minimum Contrast. Figures 8.5(a) and 8.5(b) repeat Figure 8.4(b) but using the estimates of $\kappa$ and $R$ from the data and surrogates respectively to calculate the theoretical Matern $K$ function. The estimates from the data were ($\hat{\kappa} = 18.39, \hat{R} = 0.096$) while the mean of those from the surrogates were ($\tilde{\kappa} = 21.66, \tilde{R} = 0.088$), compared to ($\kappa = 25, R = 0.1$) specified in the simulation. The histograms for $\kappa$ and $R$ from the surrogates are shown in Figures 8.7(a) and 8.7(b).

Figure 8.5(a) shows that the theoretical $K$ function using the parameter estimates from the data is virtually identical to the mean from the surrogates for radii less than 0.15. The same function using the parameter estimates from the surrogates is similar but with slightly less agreement for radii between 0.10 and 0.15, as shown in 8.5(b).
The Pair Correlation Function (PCF) is also useful for examining how well the surrogates have replicated the data. Figure 8.6(a) shows the PCF for the data as well as the mean of the PCFs from the surrogates, the corresponding 95% envelope and the theoretical constant value of 1. Generally the mean of the PCFs from the surrogates is similar to that from the data but the latter is above the 95% envelope for radii between 0.08 and 0.16. Figure 8.6(b) shows the plots and PCFs for two surrogates.

![Figure 8.5: Matern cluster surrogates: estimated \( K \) function using parameter estimates obtained from (a) the data and (b) the surrogates](image1)

![Figure 8.6: Matern cluster surrogates: (a) estimated mean Pair Correlation Function and (b) location plot and estimated Pair Correlation Function for two surrogates](image2)
Figure 8.7: Matern cluster surrogates: histograms of (a) $\kappa$ estimates and (b) radius estimates

Figure 8.8: (a) Data and (b) one of the surrogates using parameters $\kappa = 15$, $R = 0.1$ and $\mu = 10$ (Run 1)
8.4 Further simulations

Six more Matern cluster point patterns with different parameter values were simulated to test both the resampling method and the effectiveness of estimating the parameters using \texttt{matclust.estK}. Table 8.1 lists the specified parameter values together with the number of points generated \((n)\) and the parameter values estimated from the data \((\hat{\kappa}, \hat{R} \text{ and } \hat{\mu})\) while Table 8.2 lists the mean parameter values estimated from 100 surrogates \((\tilde{\kappa}, \tilde{R} \text{ and } \tilde{\mu})\). The standard deviation of the estimates of \(\tilde{\kappa} \text{ and } \tilde{R}\) are also shown in brackets. Note that \(\kappa \times \mu = \lambda\) and \(\lambda = n\) here because the domain is the unit square in each case.

The estimates of \(\kappa\) obtained from the data are similar to the mean values obtained from the surrogates in the first two runs. In the remaining four runs the estimates are lower from the surrogates, with the greatest percentage differences occurring in Runs 3 and 5.

Except in Run 2 the estimates of the fixed radius are lower from the data than from the surrogates.

Figures 8.8 to 8.13 show the data and one of the surrogates for each of the six runs. In each case resampling has reduced the clustering effect, shown by events located in areas of empty space in the simulated data. This is most evident in Runs 4 and 5 where the fixed radius is 0.05, resulting in tightly clustered points.

The resulting estimates of \(\mu\) are similar for Runs 1, 2 and 4 whereas in Runs 3, 5 and 6 the estimates from the surrogates are much higher.

Point patterns that are strongly clustered, that is have high point density within clusters, result in lifting coefficients (differences) that increase in magnitude as points are removed. Figures 8.14 and 8.15 show these differences in \(x\) and \(y\) coordinates for the first three and last three runs respectively. The general pattern observed in all the plots is that the differences are relatively stable as a certain number of points are removed followed by differences that increase in magnitude.

A simple ratio adjustment can be used on the resampled differences that takes account of this trend. The randomly selected differences are multiplied by the ratio of the trend evaluated at the index of the point being replaced to the trend evaluated at the index of the randomly selected point. This procedure was trialled on Run 4, giving \(\tilde{\kappa} = 30.64 \ (4.93)\), \(\tilde{R} = 0.061 \ (0.011)\) and \(\tilde{\mu} = 4.11\). Although the estimate of \(\kappa\) has increased slightly, the estimate of \(R\) has decreased from 0.072 towards the data estimate of 0.04 and the parameter value of 0.05. Figure 8.16 shows one surrogate before and after adjusting for the trend in the differences.

Figures 8.17 and 8.18 show the estimated \(G\) and \(K\) functions without and with the adjustment for trend. It is apparent that the trend adjustment produced little change in the surrogate patterns.
Figure 8.9: (a) Data and (b) one of the surrogates using parameters $\kappa = 15$, $R = 0.1$ and $\mu = 15$ (Run 2)

Figure 8.10: (a) Data and (b) one of the surrogates using parameters $\kappa = 10$, $R = 0.1$ and $\mu = 10$ (Run 3)
Figure 8.11: (a) Data and (b) one of the surrogates using parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)

Figure 8.12: (a) Data and (b) one of the surrogates using parameters $\kappa = 20$, $R = 0.05$ and $\mu = 10$ (Run 5)
Figure 8.13: (a) Data and (b) one of the surrogates using parameters $\kappa = 10$, $R = 0.1$ and $\mu = 20$ (Run 6)

Figure 8.14: Vector of differences in $x$ [(a),(c),(e)] and $y$ coordinates [(b),(d),(f)] for Runs 1 - 3
8.4. FURTHER SIMULATIONS

Figure 8.15: Vector of differences in $x$ [(a),(c),(e)] and $y$ coordinates [(b),(d),(f)] for Runs 4 - 6

Figure 8.16: Surrogates using differences (a) unadjusted and (b) adjusted for trend. Data parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)
Figure 8.17: Estimated $G$ function (a) without and (b) after adjusting for trend in differences using parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)

Figure 8.18: Estimated $K$ function (a) without and (b) after adjusting for trend in differences using parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)
8.5 Two difference vector sets

An alternative procedure that was examined involved separating the difference vector into two sets using a cutpoint chosen from the plot of the differences. For example, Figure 8.15 (a) and (b) shows that the first 60 differences (denoted as set 1) are smaller than the remaining 66 differences (denoted as set 2). Hence in the reconstruction phase the differences for the first 66 re-inserted points would be randomly chosen from set 2 only while the last 60 differences would be chosen from set 1 only.

Figure 8.19(a) shows the data and one surrogate using this procedure while Figure 8.19(b) shows estimates of the $K$ function from the data and surrogates. It is clear that the method does not yield any improvement compared to randomly selecting the difference vector from all possible difference vectors.

8.6 Two-stage resampling for highly clustered patterns

An alternative method for highly clustered patterns is to use a two-stage resampling procedure, first resampling clusters and then resampling within each cluster. The rationale for this comes from noting that the connected vertices of a point could come from within the same cluster or from other clusters. If the characteristics of at least one cluster are different to the others then it may be more appropriate to relate each point to its cluster centre rather than connected vertices.

The method uses the R package cluster to separate the points into clusters, with the number of clusters, $k$, specified by the user. I used the pam function in cluster that stated:

The pam-algorithm is based on the search for $k$ representative objects or medoids among the observations of the dataset. These observations should represent the structure of the data. After
finding a set of k medoids, k clusters are constructed by assigning each observation to the nearest medoid. The goal is to find k representative objects which minimize the sum of the dissimilarities of the observations to their closest representative object.

Figure 8.20 shows the output, with the points separated into fourteen clusters.

The centres of these clusters are calculated from simple averages of the locations of the cluster members. Figure 8.21(a) shows each cluster with its centre point. The difference between the location of each point and its relevant cluster centre is then obtained. The first stage of the resampling involves lifting and resampling the cluster centres. Figure 8.21(b) shows four typical surrogates.

In the second stage, random entries are selected for each point from the list of differences and added to the coordinates of the appropriate resampled cluster centre. The difference vectors were combined from all clusters and hence a randomly chosen vector is more likely to have originated from a cluster other than the one to which the point belonged.

In all previous applications of the lifting method to spatial data the difference vectors were resampled i.e. the difference vector had an x and y component. This method was used here and compared with an alternative: the differences between the x coordinates were resampled separately from the differences in the y coordinates. For example the difference between the x coordinates of a point in cluster 2 and its centre could be combined with the difference between the y coordinates of a point in cluster 7 and its centre to produce a resampled point in cluster 5.

In practice the plots of the summary functions were similar for both methods.

Figure 8.22(a) shows the data in simulation Run 4 overlaid with the locations of points from one surrogate while 8.22(b) shows four typical surrogates. These plots indicate that the method is better able to maintain the clusters than the original one-stage resampling.
8.6. TWO-STAGE RESAMPLING FOR HIGHLY CLUSTERED PATTERNS

Figure 8.20: Output from \textit{cluster} package showing points separated into clusters. (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4))

Figure 8.21: Matern cluster surrogates (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)): (a) cluster centres and (b) four surrogates of the cluster centres
From the surrogates, the mean estimate of $\kappa$ was 29.98 (specified value was 20 and the estimate from the data was 33.49) while the estimate of $R$ was 0.038 (0.05 specified and 0.040 from the data).

Figures 8.23(a), 8.23(b), 8.24(a) and 8.24(b) show the usual plots for the $F$, $G$, $J$ and $K$ functions. The estimates from the surrogates show better agreement with the data than was evident in Figures 8.17 and 8.18. Figure 8.25(a) shows the Pair Correlation Function and there is generally good agreement between the surrogates and the data, although the estimate from the data is marginally outside the 95% envelope in a couple of instances.

Figure 8.25(b) shows the estimated $K$ functions from the data, the mean from the surrogates and two theoretical plots, one using the specified parameter values and the other using the
8.6. TWO-STAGE RESAMPLING FOR HIGHLY CLUSTERED PATTERNS

Figure 8.24: Matern cluster surrogates using two-stage resampling (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)): (a) estimated $J$ function and (b) estimated $K$ function

estimated parameter values from the data. Although sharing the same general shape, there is a large gap between the theoretical curves due to the difference between the parameter estimates. The estimated $K$ function from the data has more curvature than either of the theoretical but less curvature than the mean of the surrogate estimates. Figure 8.26 shows the results obtained from using the envelope function in spatstat to estimate the range of typical $K$ functions for data originating from a Matern point process with parameters $\kappa = 20$ and $R = 0.05$.

Figure 8.25: Matern cluster surrogates using two-stage resampling (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)): (a) estimated Pair Correlation Function and (b) theoretical and estimated $K$ functions
8.6.1 Resampling differences within each cluster

The differences between the coordinates of the points and their respective cluster centres were then plotted in an attempt to explain why the estimated $K$ function from the surrogates showed more curvature than the corresponding estimate from the data. Figures 8.27(a) and 8.27(b) show the differences in the $x$ and $y$ coordinates by cluster. It appears that the differences are not homogeneous and therefore worthwhile to amend the algorithm so that the randomly selected differences are only selected from the appropriate cluster rather than from all the clusters combined. For example, cluster 1 has seven points and to place the resampled points around the resampled centre, seven randomly selected differences are chosen from the differences between the coordinates of the original seven points and their cluster centre.

A disadvantage of the method is the increased probability of obtaining coincident points when there are few points in a cluster.

Figure 8.28(a) shows four of the surrogates and they appear to be similar to the data but also to exhibit sufficient variety. The estimated $K$ function is shown in Figure 8.28(b) and we see that the mean of the estimates from the surrogates now closely follows that from the data. The same plot for the data and surrogates is shown in Figure 8.29(a) without the theoretical curves but with the 95% envelope.

The Pair Correlation Function plots are shown in Figure 8.29(b) and we see that the estimate from the data lies within the 95% envelope of the surrogate estimates except for radii less than 0.02 approximately. There is better agreement between the data and surrogates here than shown in Figure 8.25(a).
8.6. TWO-STAGE RESAMPLING FOR HIGHLY CLUSTERED PATTERNS

Figure 8.27: Matern cluster (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)): differences between cluster centres and (a) x-coordinates and (b) y-coordinates of the points

Figure 8.28: Matern cluster surrogates using two-stage resampling after resampling the differences in each cluster (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)): (a) four surrogates and (b) theoretical and estimated $K$ functions

Figure 8.29: Matern cluster surrogates using two-stage resampling after resampling the differences in each cluster (Parameters $\kappa = 20$, $R = 0.05$ and $\mu = 5$ (Run 4)): (a) estimated $K$ functions and (b) Pair Correlation Function
8.7 Two-stage resampling of the original data

The two-stage resampling procedure appears to be successful in reproducing highly clustered point patterns. It is of interest to use this approach with the less clustered data of Section 3 and compare results by examining the estimated $K$ functions and PCFs.

Figure 8.30 shows the points separated by the cluster package into ten clusters (the number of clusters was selected arbitrarily) and Figure 8.31 shows the points with their respective cluster centres. These centres were resampled and new point patterns were created by adding resampled differences as before.

Figures 8.32(a) and 8.32(b) show the differences between the $x$ and $y$ coordinates of the points and their cluster centres respectively. These differences are more similar across the clusters than was the case for simulation Run 4. However the differences were still resampled by cluster, rather than resampling the differences of all clusters combined.

Four surrogates are shown in Figure 8.33 and are more similar to the data than the original surrogates in Figure 8.1(b). In particular there is more open space than previously.

The estimated $K$ function is shown in Figure 8.34(a) and now the mean of the estimates from the surrogates closely follows that from the data. This is a much better result than shown in Figure 8.4(b) where close agreement only occurred for radii less than 0.1.

The Pair Correlation Function plot in Figure 8.34(b) also shows a better result than given in Figure 8.6(a). Previously the PCF estimated from the data lay outside the 95% envelope for radii between (0.08, 0.16) approximately. Now the estimate from the data lies within the envelope except for radii less than 0.02 and at radius 0.15.
8.7. TWO-STAGE RESAMPLING OF THE ORIGINAL DATA

Figure 8.31: Cluster centres (Parameters $\kappa = 25$, $R = 0.1$ and $\mu = 10$)

Figure 8.32: Matern cluster (Parameters $\kappa = 25$, $R = 0.1$ and $\mu = 10$): Differences in point coordinates and cluster centre (a) $x$ coordinate and (b) $y$ coordinate

Figure 8.33: Four surrogates using two-stage resampling after resampling the differences in each cluster. (Parameters $\kappa = 25$, $R = 0.1$ and $\mu = 10$). Compare with Figure 8.1(b)
CHAPTER 8. 2-D MATERN CLUSTER PROCESS EXAMPLE

Figure 8.34: Matern cluster surrogates using two-stage resampling after resampling the differences in each cluster. (Parameters $\kappa = 25$, $R = 0.1$ and $\mu = 10$): estimates of (a) $K$ function (compare with Figure 8.4(b)) and (b) Pair Correlation Function (compare with Figure 8.6(a))

8.8 Conclusion

The resampling algorithm adopted for stationary and non-homogeneous spatial point patterns does not produce surrogates of clustered data. However, adopting a two-stage resampling plan by first grouping points into clusters, resampling the cluster centres and then resampling within each cluster, does produce suitable surrogates. The best results occurred when the differences available to be resampled in the second stage were restricted to the differences within the cluster rather than from all clusters. It was noted that a disadvantage of this method is that there are more likely to be coincident points.
Chapter 9

Revisiting Non-Homogeneous 2-D Point Patterns

9.1 Introduction

The conclusion reached in Chapter 7 was that the algorithm developed for 2-dimensional point patterns in Chapter 6 did not produce surrogates of non-homogeneous data. A modification of the algorithm, catering for clustered data, was introduced in the previous chapter and it is worthwhile investigating whether this approach could also be applied in the non-homogeneous case.

9.2 Intensity function in terms of \(x\) and \(y\)

We re-visit the second non-homogeneous pattern, with intensity function \(\lambda(x, y) = 200x + 200y\), shown in Figure 9.1(a) together with ten cluster centres. The number of clusters is arbitrary, dependent upon a number of considerations, one of which being the desire to keep the intensity uniform within a cluster.

The clusters are shown in Figure 9.1(b), part of the output from R’s cluster package [31].

Figure 9.2 shows the differences between the coordinates of the points and their relevant cluster centres, separately in terms of the \(x\)- and \(y\)-coordinates. It is apparent that the differences are relatively consistent across all clusters.

The surrogates are produced by the 2-stage resampling procedure, first resampling the cluster centres and then resampling within each cluster. Four of the surrogates are shown in Figure 9.3, with the data overlaid, and associated contour plot of the estimated intensity. These surrogates appear to be similar to the data.

Table 9.1 gives the quadrat counts for the data and the mean counts for the surrogates, using the same 16 quadrats as previously. Comparing Tables 9.1 and 7.3, it is apparent that adoption of
CHAPTER 9. REVISITING NON-HOMOGENEOUS 2-D POINT PATTERNS

Figure 9.1: Non-homogeneous Poisson data with intensity \( \lambda(x, y) = 200x + 200y \): (a) data and cluster centres and (b) output from the R Cluster package.

Figure 9.2: Differences between (a) \( x \) and (b) \( y \) coordinates of points and the relevant cluster centre.

The 2-stage approach has produced surrogates with quadrat counts closer to the data than occurred with the original algorithm.

As before, we can compare the coefficients of the Poisson model that was fitted to the data (\( \hat{\theta}_0 = 4.028 \), \( \hat{\theta}_1 = 1.163 \) and \( \hat{\theta}_2 = 1.095 \)) with corresponding estimates from 100 surrogates. Figure 9.5 shows the resulting histograms from which we can see that the estimates from the data again fall nicely in the middle. Summary values of the estimates from the surrogates are given in Table 9.2. There is thus better agreement between the models fitted to the surrogates and the data than occurred with the original algorithm.

Figure 9.6(a) shows the estimates of the non-homogeneous \( K \) function, using the isotropic

<table>
<thead>
<tr>
<th>( x \backslash y )</th>
<th>(0, 0.25)</th>
<th>(0.25, 0.5)</th>
<th>(0.5, 0.75)</th>
<th>(0.75, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0.25)</td>
<td>2 (3.1)</td>
<td>5 (4.2)</td>
<td>10 (10.0)</td>
<td>7 (7.0)</td>
</tr>
<tr>
<td>(0.25, 0.5)</td>
<td>6 (6.2)</td>
<td>11 (9.7)</td>
<td>18 (17.1)</td>
<td>12 (14.0)</td>
</tr>
<tr>
<td>(0.5, 0.75)</td>
<td>10 (8.9)</td>
<td>6 (9.0)</td>
<td>14 (14.2)</td>
<td>22 (19.8)</td>
</tr>
<tr>
<td>(0.75, 1)</td>
<td>11 (11.0)</td>
<td>20 (18.9)</td>
<td>11 (11.8)</td>
<td>28 (28.1)</td>
</tr>
</tbody>
</table>

Table 9.1: Quadrat counts for data and mean quadrat counts for 1000 surrogates (in brackets) with \( \lambda(x, y) = 200x + 200y \)
Figure 9.3: Four surrogates (with original data overlaid) and their contour plots of intensity
Figure 9.4: Histograms of number of points in each quadrat for the surrogates, with data value shown: (a) quadrats 1 - 8 and (b) quadrats 9 - 16
Table 9.2: Summary statistics for estimates of coefficients of model fitted to 100 surrogates for modelling $\lambda(x, y) = 200x + 200y$. Estimates from the data are ($\hat{\theta}_0 = 4.028$, $\hat{\theta}_1 = 1.163$ and $\hat{\theta}_2 = 1.095$)

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$\hat{\theta}_0$</th>
<th>$\hat{\theta}_1$</th>
<th>$\hat{\theta}_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>3.863</td>
<td>0.922</td>
<td>0.949</td>
</tr>
<tr>
<td>First quartile</td>
<td>3.979</td>
<td>1.097</td>
<td>1.057</td>
</tr>
<tr>
<td>Median</td>
<td>4.030</td>
<td>1.159</td>
<td>1.106</td>
</tr>
<tr>
<td>Mean</td>
<td>4.025</td>
<td>1.154</td>
<td>1.109</td>
</tr>
<tr>
<td>Third quartile</td>
<td>4.065</td>
<td>1.205</td>
<td>1.162</td>
</tr>
<tr>
<td>Maximum</td>
<td>4.168</td>
<td>1.346</td>
<td>1.297</td>
</tr>
</tbody>
</table>

data, the estimate from the data lies just below the 95% envelope for radii less than 0.17. The conclusion to be drawn here is that the surrogates are more clustered than the data. A comparison with Figure 7.16(a) shows that, based on the $K$ function, the 2-stage approach has produced surrogates that are less like the data than those from the original algorithm.

Figure 9.6(b) shows the estimates of the Pair Correlation Function and in this case the mean of the estimates from the surrogates is a better fit to the estimate from the data, particularly for radii greater than 0.05, than occurred using the previous algorithm.

9.3 Conclusion

Use of the 2-stage approach has produced mixed results. Consideration of quadrat counts, coefficients of fitted models and the Pair Correlation Function would lead to the conclusion that the 2-stage approach has produced surrogates more similar to the data than those obtained from the original algorithm. However, consideration of the estimates of the $K$ function would lead to the opposite conclusion.

On balance it would appear that the 2-stage approach is at least worthy of trial when trying to obtain surrogates of 2-dimensional non-homogeneous point patterns.
Figure 9.5: Histograms for coefficients of model fitted to 100 surrogates

Figure 9.6: Estimates of (a) $K(r)$ and (b) Pair Correlation Function
Chapter 10

2-D Poisson Data with Qualitative Marks

10.1 Introduction

This Chapter discusses three methods for lifting and resampling marked point patterns. In previous work, points were distinguished only by their location; here marks are attached to each point that provide further information about the points. In the literature the marks can be continuous, in which case each point could have a unique mark, or discrete where there are only a limited number of distinct values. In this Chapter the points have either of two possible marks, termed Type 1 and Type 2.

10.2 Simulated Poisson point pattern

The spatstat package was used to generate a Poisson point pattern consisting of two types each with intensity $\lambda = 50$. The pattern contains 59 points of Type 1 and 58 of Type 2 and is shown in Figure 10.1.

Summary functions corresponding to $F$, $G$, $J$ and $K$ functions for single type point patterns can also be used for multitype point patterns. We consider the following four:

- $G_{ij}(r)$ : the cumulative distribution function of the distance from a point of type $i$ to the nearest point of type $j$. The theoretical value of $G_{ij}(r)$ for a multitype Poisson point process is

$$G_{ij}(r) = 1 - e^{-\lambda_j \pi r^2} \quad (10.1)$$

where $\lambda_j$ is the intensity of type $j$ points. Dependence between points of type $i$ and $j$ is suggested when deviations occur between empirical and theoretical $G_{ij}(r)$ curves. If point
pattern $A$ has a $G_{ij}(r)$ curve above the corresponding curve for point pattern $B$, then, in
general, the points of type $i$ in $A$ have shorter distance to type $j$ points than in $B$.

- $K_{ij}(r) : \frac{1}{\lambda_j}$ times the expected number of points of type $j$ within a radius $r$ of a typical
point of type $i$. The theoretical value of $K_{ij}(r)$ when type $i$ points are independent of type
$j$ points is

$$K_{ij}(r) = \pi r^2$$  \hspace{1cm} (10.2)

- $G_{i\bullet}(r) :$ the cumulative distribution function of the distance from a point of type $i$ to the
nearest other point of any type. The theoretical value of $G_{i\bullet}(r)$ for a multitype Poisson point
process is

$$G_{ij}(r) = 1 - e^{-\lambda \pi r^2}$$  \hspace{1cm} (10.3)

where $\lambda$ is the total intensity of points of all types. Dependence between points of type $i$
and the other points is suggested when deviations occur between empirical and theoretical
$G_{i\bullet}(r)$ curves.

- $K_{i\bullet}(r) : \frac{1}{\lambda_i}$ times the expected number of points of any type within a radius $r$ of a typical
point of type $i$ (where $\lambda$ is the total intensity ie $\sum_j \lambda_j$). The theoretical value of $K_{i\bullet}(r)$ when
type $i$ points are independent of type $j$ points is

$$K_{i\bullet}(r) = \pi r^2$$  \hspace{1cm} (10.4)
10.2. SIMULATED POISSON POINT PATTERN

Also considered are

- $J_{ij}(r)$, defined as
  \begin{equation}
  J_{ij}(r) = \frac{1 - G_{ij}(r)}{1 - F_j(r)}
  \end{equation}
  \tag{10.5}
  
  where $F_j(r)$ is the distribution function of the distance from a fixed point in space to the nearest point of type $j$ in the pattern.

- $L_{ij}(r)$, defined as
  \begin{equation}
  L_{ij}(r) = \sqrt{\frac{K_{ij}(r)}{r}}
  \end{equation}
  \tag{10.6}
  
  The theoretical value of $L_{ij}(r)$ for a Poisson process is $L_{ij}(r) = r$. The variance of the estimator of $K_{ij}(r)$ is stabilized by the use of the square root transformation and hence the $L$ function is more appropriate than the $K$ function in hypothesis tests.

- The Mark Correlation Function is a measure of the dependence between the marks of two points at a distance of $r$. It is defined in Illian et al. [15] as
  \begin{equation}
  m(r) = \frac{E_{oo}(m(0)m(r))}{\mu^2}
  \end{equation}
  \tag{10.7}
  
  The numerator is the mean of the product of the marks of a pair of points in the pattern with one point at 0 at the other a distance $r$ from 0, conditional on the presence of points at the two locations.

  The function is defined informally in spatstat as
  \begin{equation}
  \rho_f(r) = \frac{E[f(M1,M2)]}{E[f(M,M')]}
  \end{equation}
  \tag{10.8}
  
  where $E[]$ denotes expectation and $M1, M2$ are the actual marks of two points $r$ apart. $M$ and $M'$ are independent realisations of the marginal distribution of marks.

  Any function that returns a nonnegative real value can be chosen for the function $f$, but for discrete marks the choice is usually
  \begin{equation}
  f(m1,m2) = (m1 = m2)
  \end{equation}
  \tag{10.9}

  A lack of correlation is suggested when $\rho_f(r) = 1$. 

CHAPTER 10. 2-D POISSON DATA WITH QUALITATIVE MARKS

Figure 10.2: Marked Poisson pattern: (a) differences between the x and y coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations for Type 1 points using superposition

10.3 Method 1: Superposition

In this first method the data are split by mark and the two point patterns are then separately lifted and resampled before recombining into a single pattern. The overall pattern can be considered as the superposition of two patterns.

Figures 10.2(a) and 10.3(a) show the differences between the coordinates of the deleted points and the mean location of the connected vertices, separately for each point type. The corresponding ACF plots in Figures 10.2(b) and 10.3(b) indicate some autocorrelation at lag 1 in the y coordinate differences for Type 1 points and at lag 3 in the y coordinate differences for Type 2 points.

Four of the 1000 surrogate point patterns generated are shown in Figure 10.4. Only the first 100 surrogates were used for further analysis because of the time required for processing the diagnostics.

Figure 10.5(a) shows the estimates of $G_{12}(r)$ from the data and surrogates against the theoretical values. The mean of $G_{12}(r)$ from the surrogates closely follows the theoretical values while the estimate from the data lies within the 95% envelope, and is similar to the theoretical curve up to $r = 0.04$, except near $r = 0.08$.

The plot of the $J_{12}(r)$ estimators in Figure 10.5(b) is not very informative and this function will not be considered further.

Figure 10.6(a) shows the estimators of $K_{12}(r)$ while the transformed values are shown in Figure 10.6(b) as $L_{12}(r)$. Considering the latter, the mean of the estimators from the surrogates is similar to the estimator from the data while the theoretical values generally follow the lower extremity of the 95% envelope.

Figure 10.7(a) shows estimators of $G_{21}(r)$ and here the mean of the estimators from the surrogates is similar to both the estimator from the data and the theoretical curve.
Figure 10.3: Marked Poisson pattern: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations for Type 2 points using superposition.

Figure 10.4: Four surrogates using the superposition method.
Figure 10.5: Marked Poisson pattern: diagnostics from surrogates using superposition (Type 1 to Type 2) (a) $G_{12}$ and (b) $J_{12}$

The estimators of $L_{21}(r)$, shown in Figure 10.8(b), are very similar to those of $L_{12}(r)$ in Figure 10.6(b).

Figure 10.9(a) shows estimators of $G_{1•}$ where the mean of the estimators closely follows the theoretical values while the estimator from the data generally lies above, within the 95% envelope except at $r \approx 0.45$.

There is close agreement for the estimators of $K_{1•}(r)$ from the surrogates and the data up to $r = 0.15$, as shown in Figure 10.9(b). Thereafter the estimator from the data follows the theoretical values, with the mean estimator from the surrogates lying above.

The estimator of $G_{2•}$ from the data, shown in Figure 10.10(a), lies close to the mean estimator from the surrogates and the theoretical values up to $r = 0.035$ and above thereafter, but still within the 95% envelope for radii less than 0.06. Thereafter the estimator from the data appears to be above the 95% envelope.

Figure 10.10(b) shows that the estimators of $K_{2•}$ are similar to the estimators of $K_{1•}$ in Figure 10.9(b); the main difference is that the mean of the estimators from the surrogates lies just below the estimator from the data for radii less than 0.15 and above thereafter.

Figure 10.12 shows the Mark Correlation Function. Clearly the estimators from the surrogates and the data lie close to the theoretical value of 1.

All these plots indicate that, for small to intermediate distances, the surrogates could come from the same model as the data. Not all the plots are required but are included for completeness.
10.3. METHOD 1: SUPERPOSITION

Figure 10.6: Marked Poisson pattern: diagnostics from surrogates using superposition (Type 1 to Type 2) (a) $K_{12}$ and (b) $L_{12}$

Figure 10.7: Marked Poisson pattern: diagnostics from surrogates using superposition (Type 2 to Type 1) (a) $G_{21}$ and (b) $J_{21}$
Figure 10.8: Marked Poisson pattern: diagnostics from surrogates using superposition (Type 2 to Type 1) (a) $K_{21}$ and (b) $L_{21}$

Figure 10.9: Marked Poisson pattern: diagnostics from surrogates using superposition (Type 1 to any) (a) $G_{i•}$ and (b) $K_{1•}$
10.3. METHOD 1: SUPERPOSITION

Figure 10.10: Marked Poisson pattern: diagnostics from surrogates using superposition (Type 2 to any) (a) $G_2\ast$ and (b) $K_2\ast$.

Figure 10.11: Marked Poisson pattern: diagnostics from surrogates using superposition (a) $L_1\ast$ and (b) $L_2\ast$. 
Figure 10.12: Mark Correlation Function for the simulated discrete marked Poisson point pattern: theoretical value of 1, estimate from the data and mean estimate from the surrogates using superposition. Note that the translate edge correction has been used.
10.4 Method 2: Randomly reallocated marks

In this method the mark is ignored thus reducing the data to an unmarked point process. Lifting and resampling are carried out as previously to produce resampled point patterns. The vector of marks from the data is randomly permuted for each resample and the marks are assigned in order to the point locations. That is, the first entry of the permuted mark vector is assigned to the first point location of the resampled point matrix and so on. The resulting point patterns would be expected to be random if follow a Poisson pattern.

Figure 10.13(a) shows the coordinate differences and Figure 10.13(b) shows the corresponding ACFs and there appears to be some autocorrelation in the \( x \) coordinate differences at lag 1.

Figure 10.14 shows four surrogates resulting from this method.

Figures 10.15(a) and 10.15(b) show the \( G_{12}(r) \) and \( K_{12}(r) \) functions for Type 1 points to Type 2 points. Figures 10.16(a) and 10.16(b) show similar graphs for Type 2 points to Type 1 points \((G_{21}(r) \text{ and } K_{21}(r))\). The mean of the estimators of the \( G \) functions from the surrogates are similar to the theoretical Poisson functions whereas the corresponding estimators of the \( K \) functions are above the theoretical functions for radii less than 0.2. The functions derived from the data are similar to the mean of the estimators from the surrogates with the exception of \( G_{12} \) for radii in the interval \((0.05, 0.09)\) where the data values are less, although still contained within the 95\% envelope.

Figures 10.17(a) and 10.17(b) show the \( G_{\star 1}(r) \text{ and } K_{\star 1}(r) \) functions for Type 1 points to points of any other type (either type in this case). Figures 10.18(a) and 10.18(b) show similar graphs for Type 2 points to points of any other type \((G_{\star 2}(r) \text{ and } K_{\star 2}(r))\). The estimator of \( G_{\star 1}(r) \) from the data lies above the mean estimator from the surrogates for all radii and lies above the 95\% envelope for radii in \((0.04, 0.05)\). The mean of the estimates of \( K_{\star 1}(r) \) from the surrogates lies above both the theoretical function and estimator from the data for radii in \((0.05, 0.20)\). Considering \( G_{\star 2}(r) \), there is good agreement between the theoretical function and the estimators from the data and surrogates for radii less than 0.03 but the estimator from the data lies above thereafter. The estimators of \( K_{\star 2}(r) \) from the data and surrogates are similar for radii less than 0.15 but the former follows the lower edge of the 95\% envelope thereafter.

The Mark Correlation Function plot in Figure 10.19 shows that the surrogates have a mean correlation function that is virtually 1 for all radii.

While the multitype function plots indicate that the method may be suitable for testing whether a marked point pattern is completely random, the dot function plots inject an element of uncertainty.
Figure 10.13: Marked Poisson pattern: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations using random reallocation

Figure 10.14: Four surrogates using the random mark reallocation method
10.4. METHOD 2: RANDOMLY REALLOCATED MARKS

Figure 10.15: Marked Poisson pattern: diagnostics from surrogates using random mark reallocation
(a) $G_{12}$ and (b) $K_{12}$

Figure 10.16: Marked Poisson pattern: diagnostics from surrogates using random mark reallocation
(a) $G_{21}$ and (b) $K_{21}$
Figure 10.17: Marked Poisson pattern: diagnostics from surrogates using random mark reallocation
(a) $G_{1•}$ and (b) $K_{1•}$

Figure 10.18: Marked Poisson pattern: diagnostics from surrogates using random mark reallocation
(a) $G_{2•}$ and (b) $K_{2•}$
10.5 Method 3: The two differences method

The first two methods ignored any possible relationship between points of different type and the method outlined below is an attempt to make allowance for such relationships. Briefly, the point being deleted at each step is related to its connected vertices in Delaunay triangulations of points of the same type and different type. This enables two differences to be calculated: the differences between the point’s location and the means of the locations of the connected vertices of both types. Only those non-boundary connected vertices of the same type are lifted while the coordinates of points of different type remain unaltered. In the reverse operation, points are reinserted at a location that is the average of the sum of the mean locations of connected vertices of both types and randomly selected differences of either the same or different type.

The full procedure where the mark can take one of two possible values is:

1. Specify a boundary and obtain the Voronoi diagram for all the points ignoring the mark.

2. Extract the coordinates of the vertices created on the boundary of the Voronoi diagram. Attach alternate marks to these points as the boundary is traversed (an alternative would be to randomly attach marks to the boundary points). Combine these boundary points with the data.

3. Obtain the Voronoi diagram for the expanded data set and find the interior point that has the minimum tile area. This point is the one to be removed and denote it by \( P' \). Determine
its location and mark.

4. Split the data by mark and assign the name \textit{liftdata} to the subset that has the same mark as \( P' \). Assign the name \textit{nonliftdata} to the other subset and insert the coordinates of \( P' \). Both subsets now include an entry for \( P' \).

5. Obtain a Delaunay triangulation for both subsets and therefore obtain two sets of connected vertices for \( P' \), the first consisting of points of the same type and the other points of the alternate type.

6. Calculate the mean locations of each of these sets of connected vertices and then the differences between the location of \( P' \) and these means. Hence there are two sets of difference vectors, one termed the lifting set which will be used to calculate the lifting vector in the next step and the other termed the non-lifting set.

7. Lift the coordinates of the non-boundary connected vertices of the same type as \( P' \). The lifting vector is \( \frac{d_m}{n+1} \) where \( d_m \) is the difference between the coordinates of \( P' \) and the mean location of the connected vertices of the same type, of which there are \( n \).

8. Join both \textit{liftdata} and \textit{nonliftdata} after removing \( P' \). Ensure the lifting step didn’t place any point outside the boundary. If so reflect the point back inside.

9. The procedure is repeated from Step 3 until all interior points have been removed.

10. The reverse procedure commences by randomly selecting a number in the range (1, length of the lifting set). This is used to choose a difference vector from the lifting set, denoted by \( d_{\text{random},1} \), thus giving the lifting vector \( \frac{d_{\text{random},1}}{n+1} \) which is subtracted from the appropriate interior points.

11. The mean positions of the connected vertices of the same and different types are then calculated. Denote these mean positions as \( m_{i,1} \) and \( m_{i,2} \) respectively. The inserted point is then located at

\[
p(m_{i,1} + d_{\text{random},1}) + (1 - p)(m_{i,2} + d_{\text{random},2})
\]

where \( p \) is the weighting given to points of the same type (usually 0.5) and \( d_{\text{random},2} \) is the randomly selected difference vector from the non-lifting set, using the same index \textit{random} as for the lifting set.

12. Finally all interior points are checked to ensure they remain inside the boundary.

### 10.6 A simple example of the two differences method

To show how the two differences method works, consider the point pattern shown in Figure 10.20(a) consisting of 18 points, eleven of which are notionally Type 1 and seven Type 2. These points are
10.6. A SIMPLE EXAMPLE OF THE TWO DIFFERENCES METHOD

a subset from a simulated Poisson process with \( \lambda = 50 \) for both types of mark.

Figure 10.21(b) shows the points with their mark and also the inserted boundary points with their assumed marks. Figure 10.21(a) shows the triangulation of all points disregarding marks. Point 26 is the one with the smallest Voronoi cell and therefore it is the one to be removed in the first step. It is a Type 1 with coordinates (0.7945, 0.9122) and it is joined to vertices (3, 5, 7, 18, 19 and 25) in a triangulation of Type 1 points and to vertices (2, 4, 6, 29 and 32) in a triangulation of the union of point 26 and Type 2 points. These connected vertices have mean locations of (0.7959, 0.8555) and (0.8071, 0.7914) respectively.

The differences between the location of point 26 and the means are therefore (-0.0014, 0.0567) and (-0.0126, 0.1208).

The points to be lifted are the non-boundary points of the same mark as the point to be removed. The vertices connected to point 26 are (3, 5, 7, 18, 19 and 25), ie \( n = 6 \), of which (18, 19 and 25) are to be lifted. The lifting vector is \((-\frac{0.0014}{6+1}, \frac{0.0567}{6+1}) = (-0.0002, 0.0081)\) and therefore the coordinates of points 18, 19 and 25 are increased by these amounts.

The procedure continues, some steps removing Type 1 points, and therefore lifting connected Type 1 points and other steps removing and lifting Type 2 points until no interior points remain. The last point removed is 32 (Type 2, (0.5837, 0.5210)) and the means of the connected vertices are (Type 1, (0.6335, 0.5262)) and (Type 2, (0.4349, 0.4987)).

The reverse procedure begins by reinserting point 32 as a Type 2 point at location (0.5407, 0.5787). These coordinates are the average of the sum of the mean location and a random difference for the two types. In this case the randomly chosen difference was from the fifth step of the removal procedure: (0.0726, 0.0117) for Type 1 and (-0.0596, 0.1209) for Type 2. We thus have:

\[
\frac{1}{2}(0.6335 + 0.0726 + 0.4349 - 0.0596) = 0.5407
\]

for the \( x \) coordinate and

\[
\frac{1}{2}(0.5262 + 0.0117 + 0.4987 + 0.1209) = 0.5787
\]

for the \( y \) coordinate.

Note that no points have the lifting step reversed as they are all boundary points.

The last point inserted is 26 and the random number selected is 3. The third members of the lifting and non-lifting sets are (-0.05965, 0.3430) and (0.01633, 0.03688) respectively. In the penultimate step the coordinates of the non-boundary connected vertices of the same type are (0.82839, 0.91195), (0.69500, 0.73511) and (0.84077, 0.74083). The lifting step is reversed by subtracting \((-\frac{0.05965}{6+1}, \frac{0.343}{6+1})\) from each of these, then combining these points with the boundary connected vertices of the same type (1.05, 0.75399), (0.83172, 1.05) and (0.42699, 1.05) to
give a mean position \((0.78307, 0.87120)\). The mean location of the vertices of the other type is \((0.77510, 0.78099)\). Hence the final location of point 26 is

\[
\frac{1}{2}(0.78307 - 0.05965 + 0.7751 + 0.01633) = 0.75743
\]

for the \(x\) coordinate and

\[
\frac{1}{2}(0.87120 + 0.03430 + 0.78099 + 0.03688) = 0.86168
\]

for the \(y\) coordinate.

As all points have been reinserted we have one surrogate as shown in Figure 10.20(b). No conclusion can be drawn from the observation that the Type 2 points have moved closer together; this is just one surrogate out of 1000 and the small number of Type 2 points in the data results in larger differences than would be expected with a larger number of points of the same type.

\[\text{Figure 10.20: (a) Subset of a simulated bivariate marked Poisson process with } \lambda = 50 \text{ for both types and (b) one surrogate of the test data using the two differences method}\]

\[\text{Figure 10.21: (a) Triangulation of the data in Figure 10.20(a) ignoring marks and (b) data and inserted boundary points with marks shown}\]
10.7 Two differences method applied to the Poisson pattern

The two differences method is now applied to the data from Section 10.2. Figures 10.22(a), 10.22(b), 10.23(a) and 10.23(b) show the coordinate differences and associated ACFs for the points being lifted and not lifted respectively. These differences arise from the differences between the location of the deleted point and mean location of connected vertices of points of the same or different type. Neither coordinates show any autocorrelation in either set of differences.

Four of the surrogates are shown in Figure 10.24 and diagnostic plots for all the surrogates are shown in Figures 10.25 to 10.29.

Figure 10.22: Marked Poisson pattern: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices of the same type and (b) the associated autocorrelations using two differences

Figure 10.23: Marked Poisson pattern: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices of different type and (b) the associated autocorrelations using two differences
Figure 10.24: Four surrogates using the two differences method

Figure 10.25: Diagnostics for surrogates after resampling test data from Section 10.2 with two differences: (a) $G_{12}$ and (b) $K_{12}$
10.7. TWO DIFFERENCES METHOD APPLIED TO THE POISSON PATTERN

Figure 10.26: Diagnostics for surrogates after resampling test data from Section 10.2 with two differences: (a) $G_{21}$ and (b) $K_{21}$

Figure 10.27: Diagnostics for surrogates after resampling test data from Section 10.2 with two differences: (a) $G_{1*}$ and (b) $K_{1*}$
**CHAPTER 10. 2-D POISSON DATA WITH QUALITATIVE MARKS**

Figure 10.28: Diagnostics for surrogates after resampling test data from Section 10.2 with two differences: (a) $G_{2\bullet}$ and (b) $K_{2\bullet}$.

Figure 10.29: Mark Correlation Function for the simulated discrete marked Poisson point pattern using two differences for resampling: theoretical value of 1, estimate from the data and mean estimate from the surrogates.
10.8 Analysis of the three methods

The following observations can be made after comparing plots using the three methods:

- $G_{12}(r)$ - Figures 10.5(a), 10.15(a) and 10.25(a). The random mark reallocation method produces surrogates whose mean $G_{12}$ function is closest to the theoretical Poisson. However, this method also produces the widest 95% envelope, resulting in the empirical function lying within. The two differences method produces the narrowest 95% envelope but the mean surrogate function deviates from the theoretical and the empirical function lies below the envelope for radii larger than 0.5.

- $K_{12}(r)$ - Figures 10.6(a), 10.15(b) and 10.25(b). The random reallocation and two differences methods produce similar agreement between the surrogates and the data for radii less than 0.14. Thereafter the former method is better. None of the methods show a good fit between the surrogates and the theoretical. The superposition method results in the mean of the surrogate $K$ values lying below the data values for radii less than 0.15 and above thereof.

- $G_{21}(r)$ - Figures 10.7(a), 10.16(a) and 10.26(a). The superposition method produces the best agreement between data and surrogates while the two differences method yields the mean surrogate generally above both the data and theoretical. Conversely, the random reallocation method produces surrogates whose mean $G_{21}$ curve is below the data values. This method also shows best agreement between surrogates and theoretical for radii less than 0.06 while the superposition method is best thereafter.

- $K_{21}(r)$ - Figures 10.8(a), 10.16(b) and 10.26(b). Both the random reallocation and two differences method produce surrogates whose mean curve best agrees with the data for radii less than 0.15. Thereafter the random reallocation method produces the best agreement. All three methods exhibit poor agreement between surrogates and theoretical except for radii larger than 0.2 with the random reallocation method.

- $G_{1\bullet}(r)$ - Figures 10.9(a), 10.17(a) and 10.27(a). The two differences method produces the best agreement between surrogates and data while the random reallocation method results in poor agreement overall. The superposition method shows good agreement between surrogates and theoretical.

- $K_{1\bullet}(r)$ - Figures 10.9(b), 10.17(b) and 10.27(b). For radii less than 0.15 the best agreement between surrogates and data is produced by the superposition method and the worst by the random reallocation method, although all methods are similar. Thereafter the random reallocation method shows a better fit to the data and theoretical values.
• $G_2^*(r)$ - Figures 10.10(a), 10.18(a) and 10.28(a). While the superposition and random reallocation methods show good agreement between surrogates, data and theoretical for radii less than 0.03, overall the two differences method produces the best agreement between surrogates and data with the superposition method next best. The superposition method yields best overall agreement between surrogates and theoretical.

• $K_2^*(r)$ - Figures 10.10(b), 10.18(b) and 10.28(b). The two differences and random reallocation methods produce the similar agreement between surrogates and data for radii less than 0.15 with the former being slightly better. Thereafter all methods yield similar results.

10.9 Conclusion

A comparison of the diagnostic plots from the three suggested resampling methods indicates that there is not a single method that produces the best results with every diagnostic. The two differences method produced the poorest results for $G_{12}(r)$ but the best results for the other $G$ functions and good agreement between surrogates and data for all $K$ functions for radii less than 0.15. The random reallocation method produced poor results for $G_{1*}(r)$ and $G_{2*}(r)$ for radii greater than 0.03 but yielded best agreement between surrogates and data at higher radii for all the $K$ functions.

However the random reallocation method would not be suitable for producing surrogates for a non-Poisson point pattern. The next chapter applies the superposition and two differences methods to such a pattern.
Chapter 11

Example of Data with Qualitative Marks

11.1 Introduction

The previous chapter examined variations of the wavelet resampling method that could be adopted for 2-dimensional spatial data with qualitative marks. While that chapter analysed a simulated Poisson point pattern, this chapter uses an interesting real dataset available in spatstat.

The data come from an investigation of a 924 ft x 924 ft (19.6 acre) plot in Lansing Woods, Clinton County, Michigan USA by D.J. Gerrard. The data give the locations of 2251 trees and their botanical classification (into hickories, maples, red oaks, white oaks, black oaks and miscellaneous trees). The original plot size (924 x 924 feet) has been rescaled to the unit square. [4]

Figure 11.1 shows the locations of the tree types separately.

Although the dataset contains information on six tree types, this analysis is restricted to black oaks and miscellaneous trees because of the interesting pattern shown in Figure 11.2. The miscellaneous trees appear to be in complementary locations to the black oaks - areas in the bottom left and in an L-shape from top centre to centre right devoid of black oaks are populated by miscellaneous trees. The black oaks exist in clusters in the top left and right of the window and widely spaced elsewhere. Here we are not interested in the reasons why black oaks exhibit the observed location pattern; the bivariate subset, being real and clearly non-Poisson, is useful for comparing two methods of wavelet lifting/resampling described in Chapter 10 for generating realisations that replicate the data.
CHAPTER 11. EXAMPLE OF DATA WITH QUALITATIVE MARKS

Figure 11.1: Lansing Woods data: Location of tree types shown separately

Figure 11.2: Lansing Woods data: Location of black oak (open circles) and miscellaneous trees (filled circles)
11.2 Analysing the surrogates

The following plots were generated as part of the analysis:

- The nearest neighbour distance functions ($G_{12}, G_{21}, G_{1*}$ and $G_{2*}$)
- The $K$ functions ($K_{12}, K_{21}, K_{1*}$ and $K_{2*}$)
- The corresponding $L$ functions ($L_{12}, L_{21}, L_{1*}$ and $L_{2*}$)
- The mark correlation function $\rho_f(r)$ with $f(m_1, m_2) = 1(m_1 = m_2)$
- The partial pair correlation functions ($g_{11}, g_{12}, g_{21}$ and $g_{22}$) with $g(r) = \frac{K(r)}{2\pi r}$ using the appropriate $K$ function.

The purposes of the analysis are:

1. to see whether the surrogates could be possible realisations from the same process that generated the data, and
2. to compare the results obtained from the superposition and 2-differences methods of lifting.

11.3 The superposition method

Figures 11.3(a) and 11.4(a) show the coordinate differences for both point types and Figures 11.3(b) and 11.4(b) the associated ACFs. The first half of the $x$ coordinate differences for type 1 points are smaller than the second half and show slight autocorrelation at lags 1 and 3 while the $x$ coordinate differences for type 2 points show slight autocorrelation at lags 2 and 3.

Figure 11.5 shows four of the 1000 surrogates generated. They show that the two types of
Figure 11.4: Lansing subset: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices and (b) the associated autocorrelations for Type 2 points using superposition.

tree are present in the same general locations as they were in the data. We can therefore anticipate that the surrogates will have similar first and second order characteristics as the data.

The cumulative distribution function $G_{ij}(r)$ provides an estimate of the distance from a point

Figure 11.5: Four of the 1000 surrogates using superposition

of type $i$ to the nearest point of type $j$. Figure 11.6(a) shows the results for $G_{12}$ from which we can conclude:

1. The theoretical Poisson curve is substantially above the data and all the surrogates, indicating that the distances from type 1 points to the nearest type 2 points are greater than would be expected in a Poisson pattern. This is obvious from the scatterplots.

2. The mean of the $G_{12}$ curves for the surrogates is a different shape to the corresponding curve
for the data, increasing at a greater rate before flattening out, indicating that in general the surrogates have shorter distances from type 1 points to type 2 points than occurs in the data.

3. The curve from the data lies below the 95% envelope for the surrogates up to radius 0.8 and lies within the envelope up to approximately radius 0.18.

We can conclude that the resampling method has reduced the distance between type 1 points and their closest type 2 points.

The multitype $K_{ij}(r)$ function estimates the number of points of type $j$ within distance $r$ of a typical point of type $i$. Figure 11.6(b) shows the relevant curves estimating the number of type 2 points within distance $r$ of a typical type 1 point. The data and surrogates are again clearly non-Poisson. The $K_{12}(r)$ curve from the mean of the surrogates is above that from the data for $r > 0.15$ but below thereafter. The estimate from the data is below the 95% envelope for $r < 0.10$, confirming the observation from $G_{12}(r)$ that the surrogates had type 1 points with closer nearest neighbour type 2 points than the data.

The $L$ function is preferred over the $K$ function because it is more stable at higher values of $r$ and because it plots as a straight line rather than a parabola. However in this case the $L_{12}(r)$ curves shown in Figure 11.8(a) offer no further insight than obtained from Figure 11.6(b).

Figure 11.7(a) shows the nearest neighbour distances from type 2 points to type 1 points, $G_{21}(r)$. In comparing this plot against that for $G_{12}(r)$ note that the horizontal scales are different. Here the estimate from the surrogates is well above the estimate from the data for radii less than 0.1, with the latter below the 95% envelope. Figures 11.7(b) and 11.8(b) show the plots for $K_{21}(r)$ (the expected number of type 1 points within radius $r$ of a typical type 2 point) and $L_{21}(r)$ respectively. They are very similar to $K_{12}(r)$ and $L_{12}(r)$ and therefore the same comments apply.
CHAPTER 11. EXAMPLE OF DATA WITH QUALITATIVE MARKS

Figure 11.7: Subset of Lansing data: diagnostics from surrogates using superposition (Type 2 to Type 1) (a) $G_{21}(r)$ and (b) $K_{21}(r)$

Figure 11.8: Subset of Lansing data: diagnostics from surrogates using superposition (a) $L_{12}(r)$ and (b) $L_{21}(r)$
11.3. THE SUPERPOSITION METHOD

The next six figures show the plots of the dot functions where observations are taken from points of type $i$ to points of any type. Figure 11.9(a) shows the curves for $G_{1\bullet}(r)$ for $0 \leq r \leq 0.05$. The theoretical Poisson curve lies just below the 95% envelope for radii less than approximately 0.038 and inside thereafter. Except for $r < 0.01$, the curve obtained from the data lies inside the 95% envelope from the surrogates.

Figures 11.9(b) and 11.10(a) show the curves for $K_{1\bullet}(r)$ and $L_{1\bullet}(r)$ respectively. We see that the estimate from the data lies below the 95% envelope from the surrogates for radii greater than 0.05.

The mean of the estimates of $G_{2\bullet}$ from the surrogates, shown in Figure 11.9(c), follows the estimate from the data. Figures 11.9(d) and 11.10(b) show the curves for $K_{2\bullet}(r)$ and $L_{2\bullet}(r)$ respectively. As with $K_{1\bullet}(r)$ and $L_{1\bullet}(r)$, the estimate from the data lies below the 95% envelope from the surrogates for radii greater than 0.05 but then comes inside the envelope for radii greater than 0.20.
CHAPTER 11. EXAMPLE OF DATA WITH QUALITATIVE MARKS

Figure 11.10: Subset of Lansing data: diagnostics from surrogates using superposition (a) $L_1(r)$ and (b) $L_2(r)$

With the exception of $G_1$ and $G_2$, the second-order functions show that the surrogates were not replicating some characteristics of the data. We can also examine other functions to clarify the situation.

The first of these is the Mark Correlation Function (MCF) which was defined in Chapter 10. Figure 11.11(a) plots the relevant graphs of the Mark Correlation Function. We see that both the graphs from the data and the mean of the surrogates approach the value of 1 at a radius greater than 0.25, (a value of 1 indicates that the marks are independent and identically distributed).

The Mark Correlation Function from the data is steeper than the mean estimate from the surrogates and generally lies outside the 95% envelope. This indicates that more points of different mark lie closer to a typical point in the surrogates than in the data, confirming the conclusion from the $G$ and $K$ multi functions.

Figure 11.11(b) shows all four Partial Pair Correlation Functions. These functions are defined in terms of the derivative of the corresponding $K$ function:

$$g_{ij}(r) = \frac{K_{ij}^{'}(r)}{2\pi r}$$

and have the properties

$$g_{ij}(r) \geq 0$$

and

$$\lim_{x \to \infty} g_{ij}(r) = 1$$

Clustering is indicated when $g_{ij}(r) > 1$ and inhibition when $g_{ij}(r) < 1$.

Both $g_{11}(r)$ and $g_{22}(r)$ show similar properties, having values between 2 and 4 for radii less than 0.05 and decay exponentially to 1 by radius 0.20. The mean of the surrogates’ function values
11.3. THE SUPERPOSITION METHOD

follows the data function value.

The cross-pair correlation functions, \( g_{12}(r) \) and \( g_{21}(r) \), also show similar properties - theoretically they should be the same. The mean of the surrogates commences at about 0.4 for both \( g_{12}(r) \) and \( g_{21}(r) \) and increase steadily towards 1, with the former reaching 0.75 approximately at radius 0.25 whereas the latter is approximately 0.6 at that radius. The function values derived from the data generally remain within the 95% envelope for radii between 0.05 and 0.15.

There are two other functions that are useful to analyse correlation of qualitative marks:

1. The delta function defined in Equation 5.3.14 of [15]

\[
\delta_{ij}(r) = \lambda_j g_{ij}(r) - \lambda_i g_{ii}(r)
\]

(11.4)

for \( r \geq 0 \). If there are more points of type \( j \) around a point of type \( i \) than points of type \( i \) at a distance \( r \) then \( \delta_{ij}(r) > 0 \). Figures 11.12(a) and 11.12(b) show estimates of \( \delta_{12}(r) \) and \( \delta_{21}(r) \) respectively, from the data and the surrogates. The graphs show similar characteristics increasing from a value around -400 for \( \delta_{12}(r) \) and -300 for \( \delta_{21}(r) \) at \( r = 0.01 \) to 0 at some \( r > 0.25 \). Because the values are all negative we can conclude that there are more points of the same type around a given point than points of the other type. The mean of the surrogates generally falls below the estimates from the data. In both cases the estimate from the data falls outside the 95% envelope for a number of radii.

2. The mark connection functions \( p_{ij}(r) \) [15] calculated using

\[
p_{ij}(r) = p_i p_j \frac{g_{ij}(r)}{g(r)}
\]

(11.5)

Given that two points are at distance \( r \) in the point process \( N \), \( p_{ij}(r) \) gives the probability that they have marks \( i \) and \( j \). We can therefore calculate \( p_{11}(r) \), \( p_{12}(r) \), and \( p_{22}(r) \) and these are shown in Figures 11.13(a), 11.14(a) and 11.15(a) respectively. The limits as \( r \to \infty \) of \( p_{ii}(r) \) and \( p_{ij}(r) \) are \( p_i^2 \) and \( 2p_i p_j \) respectively, where \( p_i = \frac{\lambda_i}{\lambda_i + \lambda_j} \).

In this example there are 135 points of type 1 and 105 points of type 2. Therefore we have

\[
\lambda_1 = \frac{135}{240} = 0.5625 \quad \text{and} \quad \lambda_2 = \frac{105}{240} = 0.4375,
\]

giving the limits of \( p_{11}(r) \), \( p_{12}(r) \) and \( p_{22}(r) \) as 0.316, 0.492 and 0.191 respectively. It appears that \( p_{11}(r) \) is approaching 0.4 and \( p_{22}(r) \) appears to approach 0.25 while \( p_{12}(r) \) is still increasing.

In all three graphs there is variability in the estimates from the data, the means from the surrogates generally are not a good fit and the 95% envelopes do not always contain the data estimate. This is particularly noticeable in the graph of \( p_{12}(r) \) for \( 0.15 < r < 0.20 \). However the plots of the estimated functions using the data have 513 coordinate pairs (as output from spatstat’s pcf function) whereas the means and envelopes from the surrogates use 26 coordinate pairs (evaluated at radii between 0 and 0.25 with 0.01 intervals). Therefore the
Figure 11.11: Diagnostics from surrogates using superposition: (a) Mark Correlation Function and (b) Partial Pair Correlation Functions
11.3. THE SUPERPOSITION METHOD

latter would not be expected to show sharp changes in value. Figures 11.13(b), 11.14(b) and 11.15(b) show the mark connection functions for the last of the 1000 surrogates and these show similar variability to the functions derived from the data.

Figure 11.12: Subset of Lansing data: diagnostics from surrogates using superposition: (a) $\delta_{12}(r)$ and (b) $\delta_{21}(r)$

Figure 11.13: Mark Connection Function $p_{11}(r)$ for the Lansing subset using superposition: mean estimate from the surrogates with 95% envelope and (a) estimate from the data or (b) estimate from one surrogate
Figure 11.14: Mark Connection Function $p_{12}(r)$ for the Lansing subset using superposition: mean estimate from the surrogates with 95% envelope and (a) estimate from the data or (b) estimate from one surrogate.

Figure 11.15: Mark Connection Function $p_{22}(r)$ for the Lansing subset using superposition: mean estimate from the surrogates with 95% envelope and (a) estimate from the data or (b) estimate from one surrogate.
11.4 The two differences method

The above procedure was repeated using the 2-differences method of lifting. Figures 11.16(a) and 11.17(a) show the differences between the location of the point being deleted and mean location of connected vertices of points of the same and different types respectively. The ACF plots in Figures 11.16(b) and 11.17(b) show smaller autocorrelations than those arising from the superposition method.

Figure 11.18 shows four surrogates and they appear to be similar to the data.

Figure 11.16: Lansing subset: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices of the same type and (b) the associated autocorrelations using two differences

Figure 11.17: Lansing subset: (a) differences between the $x$ and $y$ coordinates of the deleted point and the mean position of connected vertices of different type and (b) the associated autocorrelations using two differences
11.5 Comparison of the two methods

The following comparisons can be made of the diagnostics from the superposition and two differences methods:

- $G_{12}(r)$ - Figures 11.6(a) and 11.19(a). The mean curve from the surrogates using the superposition method flattens out at radius approximately 0.07, instead of increasing at a relatively constant rate. The two differences method produces an estimate that looks similar to that from the data but shifted upwards. Mostly the data estimate lies below the 95% envelope.

- $K_{12}(r)$ - Figures 11.6(b) and 11.19(b). The two differences method produces a narrower 95% envelope than the superposition method. Both methods produce similar estimates for radii less than 0.15 but the two difference method is better thereafter.

- $G_{21}(r)$ - Figures 11.7(a) and 11.20(a). Similar poor estimates are produced by both methods.

- $K_{21}(r)$ - Figures 11.7(b) and 11.20(b). The two differences method results in a better fit between the surrogates and the data, particularly for radii greater than 0.15.

- $G_{1\bullet}(r)$ - Figures 11.9(a) and 11.22(a). Both methods produce similar estimates from the surrogates with the two differences method yielding slightly better agreement with the data.

- $K_{1\bullet}(r)$ - Figures 11.9(b) and 11.22(b). Here the two differences method results in good agreement between the surrogates and the data for radii less than approximately 0.18. The
95% envelope from the superposition method is consistently above the estimate from the data for all but the lowest radii.

- \( G_{2\bullet}(r) \) - Figures 11.9(c) and 11.23(a). Both methods produce similar estimates from the surrogates with good agreement with the data.

- \( K_{2\bullet}(r) \) - Figures 11.9(d) and 11.23(b). The two differences method results in good agreement between the surrogates and data for radii less than 0.1; thereafter the estimate from the data follows the lower boundary of the 95% envelope from the surrogates. In contrast, the estimate from the data lies below the superposition method envelope for radii between 0.05 and 0.20.

- Mark Correlation Function - Figures 11.11(a) and 11.25(a). There is poor agreement between the surrogates and the data using the superposition method. Mostly the estimate from the data falls outside the 95% envelope from the surrogates. In contrast, for radii greater than 0.07, the estimate from the data lies within the envelope using the two differences method.

To make an easier comparison of the delta and mark connection functions Figures 11.28(a) to 11.29(c) show the mean of the estimates from the surrogates for \( \delta_{12}(r) \), \( \delta_{21}(r) \), \( p_{11}(r) \), \( p_{12}(r) \) and \( p_{22}(r) \) under both methods, together with the estimate from the data. Clearly, in all cases, the estimates using the two differences method show better agreement with the estimate from the data than those from the superposition method.

Figure 11.19: Diagnostics from surrogates using two differences: (a) \( G_{12}(r) \) and (b) \( K_{12}(r) \)
Figure 11.20: Diagnostics from surrogates using two differences: (a) $G_{21}(r)$ and (b) $K_{21}(r)$

Figure 11.21: Diagnostics from surrogates using two differences: (a) $L_{12}(r)$ and (b) $L_{21}(r)$

Figure 11.22: Diagnostics from surrogates using two differences: (a) $G_{1•}(r)$ and (b) $K_{1•}(r)$
11.5. COMPARISON OF THE TWO METHODS

Figure 11.23: Diagnostics from surrogates using two differences: (a) $G_{2*}(r)$ and (b) $K_{2*}(r)$

Figure 11.24: Diagnostics from surrogates using two differences: (a) $L_{1*}(r)$ and (b) $L_{2*}(r)$
Figure 11.25: Diagnostics from surrogates using two differences: (a) Mark Correlation Function and (b) Pair Correlation Functions
Figure 11.26: Diagnostics from surrogates using two differences: (a) $\delta_{12}(r)$ and (b) $\delta_{21}(r)$

Figure 11.27: Diagnostics from surrogates using two differences: the mark connection functions (a) $p_{11}(r)$, (b) $p_{12}(r)$ and (c) $p_{22}(r)$
CHAPTER 11. EXAMPLE OF DATA WITH QUALITATIVE MARKS

Figure 11.28: Comparison of the estimates of the delta functions using the superposition and two differences methods: (a) $\delta_{12}(r)$ and (b) $\delta_{21}(r)$

Figure 11.29: Comparison of the estimates of the MCFs using the superposition and two differences methods: (a) $p_{11}(r)$, (b) $p_{12}(r)$ and (c) $p_{22}(r)$
Chapter 12

2-D Pattern with Quantitative Marks

12.1 Introduction

This chapter presents the results when the 2-dimensional wavelet lifting and resampling algorithm is applied to data with quantitative marks.

The forestry industry is a good source of data for continuously marked spatial point patterns and this chapter examines three such patterns where the marks are tree diameters at breast height (DBH).

The algorithm is described in Section 2 and a description of the data is provided in Section 3. Section 4 details the various diagnostics adopted and Section 5 includes all the analysis. Section 6 looks at a modified algorithm and a conclusion is presented in Section 7.

12.2 The algorithm

The algorithm is a variation of that used to resample unmarked point patterns. There are two possible ways of developing the algorithm: the locations of the points can remain fixed with the marks resampled or both the locations and the marks are resampled. The former alternative is initially adopted in this chapter with the latter included for comparison afterwards.

The steps in the forward lifting phase consist of:

1. Specify the coordinates of the vertices of an appropriate window containing the data and then obtain the bounded Voronoi diagram. Extract the coordinates of the convex hull of this diagram and use the smoothed point pattern (from spatstat) to interpolate mark values of all points on the boundary. Add these boundary points to the data.
2. Using a loop to process all the interior points, commencing at \( j = 1 \), first calculate the areas of all the cells in the Voronoi diagram. The point to be removed at each step is that which has the smallest cell area.

3. Extract the coordinates of all points connected to the selected point in the Delaunay triangulation. Store the index numbers of these vertices for future use. Predict the value of the mark by averaging the marks of the connected vertices. The difference is then the mark of the selected point less the predicted mark.

Suppose the point to be deleted is connected to \( n_j \) vertices. Let \( m_1 \) be the mark of the deleted point and \( m_i \) the mark of vertex \( i \) with the vertices being numbered \( (2, 3, ..., n_j + 1) \). Then the difference, \( d \), is calculated from:

\[
d = m_1 - \frac{1}{n_j} \sum_{i=2}^{n_j+1} m_i
\]

4. Uplift the marks of the connected vertices by the difference in the mean marks, \( d^m \), when the mark of the selected point is included/excluded. It transpires that the difference in means is just the difference between actual and predicted mark, as calculated in Step 3, divided by (number of connected vertices + 1). The derivation of this result is:

\[
d^m = \frac{1}{n+1} \sum_{i=1}^{n+1} m_i - \frac{1}{n} \sum_{i=2}^{n+1} m_i = \frac{m_1}{n+1} + \frac{1}{n+1} \sum_{i=2}^{n+1} m_i - \frac{1}{n} \sum_{i=2}^{n+1} m_i = \frac{m_1}{n+1} + \frac{-1}{n(n+1)} \sum_{i=2}^{n+1} m_i = \frac{m_1}{n+1} + \frac{-1}{n(n+1)} n(1-n) = \frac{d}{n+1}
\]

The marks of all connected vertices are uplifted, regardless of whether the point is an interior or boundary point. This differs from the algorithm for unmarked point patterns where the location of boundary points was not uplifted.

5. The point is then deleted from the data and the loop continues until no interior points remain, when the loop counter, \( j \), takes the value \( N \), the number of interior points.

The next phase includes both resampling and inverting:

1. Loop until all interior points have been replaced, i.e., the loop counter \( j \) commences at \( N \) and reduces to 1.

2. A difference value is randomly chosen from the list of \( d \) values; denote this by \( d^r \).
3. Retrieve the index numbers of the vertices that were connected to the point removed at the relevant stage of the forward stage.

4. Reverse the uplift step by subtracting \( \frac{d^r}{(n_j + 1)} \) from the marks of the connected vertices.

5. Calculate the mean mark of the connected vertices. Denote this by \( m^r_j \).

6. Place an interior point at the same location from which it was removed in the forward step with the mark calculated from \( m^r_j + d^r \).

This completes the resampling, yielding one surrogate point pattern, and the procedure is repeated 1000 times.

12.3 Data

Three point patterns have been analysed to test the algorithm:

1. Boese data. This 50m by 50m point pattern shows a stand of mixed oak and beech trees in the Boeselagerestate, located in the Sauerland region of West Germany. The data were supplied by Dr Arne Pommerening of Bangor University. There were 21 oak and 151 beech included in the data and the analysis was restricted to the beech trees only. The location and diameter at breast height (DBH) was available for each tree.

2. Spruce data. This pattern of Norwegian spruce trees in a 56m by 38m natural forest stand in Saxonia, Germany is available in spatstat. Again the location and DBH values were available for each tree.

3. Longleaf data. This pattern is also available in spatstat. The location and diameters of 584 Longleaf pine (Pinus palustris) trees in a 200m by 200m region in southern Georgia (USA) are available. Both immature and mature trees are included and other researchers have restricted the analysis to mature trees.

12.3.1 Assumptions

The only assumption required when resampling the marks is that the differences between the mark of the deleted points and the mean mark of the connected vertices are independent. This assumption is checked by examining the autocorrelation plot.

When the location and mark are resampled the assumption is that the difference vectors, consisting of differences in \( x \) coordinates, \( y \) coordinates and marks, are independent.
CHAPTER 12. 2-D PATTERN WITH QUANTITATIVE MARKS

12.4 Diagnostics

The diagnostics used to assess the adequacy of the resampling algorithm are classified as either summary or second-order characteristics.

12.4.1 Summary characteristics

Three point-related indices that can be used for quantitative marks are the nearest-neighbour mark product index, the nearest-neighbour variogram index [15] and the DBH Differentiation (2) index [29] while spatstat provides a useful summary function. A brief description of these indices follows.

- The nearest-neighbour mark product index, defined by

\[ n_{mm} = \frac{E_o(m(0)m(z_1(0)))}{\mu^2} \]  

(12.3)

where \( E_o \) is the expectation with respect to the Palm distribution (the index ‘o’ indicates shifting of the pattern towards the origin ‘o’ - refer to pp 177/8 of Illian et al [15]), \( m(0) \) is the mark at the origin, \( m(z_1(0)) \) is the mark of the nearest neighbour of the point at the origin and \( \mu \) is the mean mark. If \( n_{mm} > 1 \) then the typical point and its nearest neighbour mutually stimulate each other; mutual inhibition is implied when \( n_{mm} < 1 \) and the marks are independent when \( n_{mm} = 1 \).

- The Nearest-neighbour Variogram Index, defined by

\[ n_\gamma = \frac{1}{2} \frac{E_o(m(0) - m(z_1(0)))^2}{\sigma^2_\mu} \]  

(12.4)

where \( \sigma^2_\mu \) is the mark variance. This index evaluates the variability of marks of nearest-neighbour points and independence is again implied when \( n_\gamma = 1 \).

- The DBH differentiation (2) index was developed in analysing tree patterns but is applicable for any quantitatively marked pattern. The index is defined by:

\[ T_i = 1 - \frac{1}{n} \sum_{j=1}^{n} \frac{\min(DBH_i, DBH_j)}{\max(DBH_i, DBH_j)} \]  

(12.5)

This index compares the mark at point \( i \) with the marks of its \( n \) nearest neighbours. If all the marks are identical then the index is 0; a value of 1 indicates large differences between marks.

- The markstat function in spatstat allows evaluation of any function of specified neighbouring points for all points in the pattern. In this paper the function was used to determine the
mean of the four nearest-neighbours.

### 12.4.2 Second-order characteristics

Two second-order characteristics that are used in the analysis are:

- The Mark Correlation Function and its generalisations. The classical mark correlation function \( m(r) \) was defined in Eqn 9.7. This correlation function can be generalised by replacing the product of the marks by a test function, dependent upon the marks, that is appropriate to the type of marks under consideration. The standard test function for quantitative marks is \( m_1 m_2 \) and this is used in the following analysis. An alternative is \( \frac{1}{2} (m_1 - m_2)^2 \), similar to that used in the variogram index. Both test functions are easily implemented in `spatstat`.

- The \( K \) function. `Spatstat` implements a version of the \( K \) function, \( K_{multi} \), that counts the expected number of points of subset \( J \) within a given distance from a typical point in subset \( I \). In the following analysis the subsets are defined by the median or upper and lower quartiles. The theoretical \( K \) function here is \( \pi r^2 \).

### 12.5 Data analysis

#### 12.5.1 Boese data

A summary of the mark data is given in Table 12.1. We see that the minimum mark is 7.0 cm, the middle 50% of marks lie between 12.0 cm and 19.5 cm and the largest mark is 40.0 cm.

Figure 12.1 shows the plot of the data, with the radius of the circles indicating the mark. Also shown are the smoothed plot as generated by `spatstat`, the mark histogram and the mark correlation function based on the product of the marks as the test function. A correlation value of 1 indicates that the marks are independent; the plot shows that correlation is less than 1 for radii less than 3 and greater thereafter but the values are always in the range (0.90, 1.05). For this test function, correlation values less than 1 indicate inhibition while values greater than 1 indicate stimulation.

Figure 12.2 shows the differences, for each deleted point, between the actual marks and the predicted marks, using the average of the marks at the connected vertices in the Delaunay triangulation. There is no trend, neither is autocorrelation present, as indicated in Figure 12.2. Therefore it is appropriate to resample these differences as we can assume they are independent.

Figures 12.3 and 12.4 show the same plots for four surrogates as were given in Figure 12.1 for the data. The smoothed mark plots and the histograms show similarities but the mark correlation functions are quite variable; however the latter are generally close to 1 except for small radii.
Table 12.1: Summary statistics for Boese data marks

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>7.00</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>12.00</td>
</tr>
<tr>
<td>Median</td>
<td>15.00</td>
</tr>
<tr>
<td>Mean</td>
<td>16.74</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>19.50</td>
</tr>
<tr>
<td>Maximum</td>
<td>40.00</td>
</tr>
</tbody>
</table>

Figure 12.1: Boese data: (top left) tree locations with circle diameters scaled by mark ie trunk diameter (top right) plot of smoothed marks (bottom left) histogram of marks (bottom right) mark correlation function based on the product of the marks

Figure 12.2: Boese data: differences between the marks of the deleted point and the mean position of connected vertices and the associated autocorrelations
Figure 12.3: Boese data: The same graphs as in Figure 12.1 for two surrogates after lifting, resampling the differences and inverting
Figure 12.4: Boese data: Two more surrogates after lifting, resampling and inverting.
Table 12.2: Boese data: Nearest neighbour indices. The mark product index for the data is 0.996 and the variogram index is 0.789

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Mean mark product index</th>
<th>Mean variogram index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.90</td>
<td>0.47</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>0.99</td>
<td>0.78</td>
</tr>
<tr>
<td>Median</td>
<td>1.01</td>
<td>0.85</td>
</tr>
<tr>
<td>Mean</td>
<td>1.01</td>
<td>0.86</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>1.03</td>
<td>0.94</td>
</tr>
<tr>
<td>Maximum</td>
<td>1.11</td>
<td>1.27</td>
</tr>
</tbody>
</table>

Table 12.3: Boese data: Summary statistics for the markstat function defined as the mean mark of 4 nearest neighbours

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Data</th>
<th>Mean of surrogate values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>10.50</td>
<td>13.15</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>14.00</td>
<td>15.28</td>
</tr>
<tr>
<td>Median</td>
<td>16.00</td>
<td>16.11</td>
</tr>
<tr>
<td>Mean</td>
<td>16.73</td>
<td>16.28</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>19.12</td>
<td>17.26</td>
</tr>
<tr>
<td>Maximum</td>
<td>27.75</td>
<td>19.84</td>
</tr>
</tbody>
</table>

Boese data - indices

Table 12.2 gives the 5-number summary of the mean mark product indices and the mean variogram indices for the surrogates. The respective indices for the data were 0.996 and 0.789, both values being approximately equal to the lower quartiles of the surrogate values. The middle 50% of the mean mark product indices for the surrogates lie in the range (0.99, 1.03) so there is not much deviation from 1.

Further analysis was performed using spatstat’s markstat function. The mean of the marks of the four nearest neighbours was calculated for each point in the data and each surrogate and the 5-number summary is given in Table 12.3. We see that the median and mean are 16.11 and 16.28 for the surrogates compared to 16.00 and 16.73 for the data. It should be remembered that the surrogate values are means of 1000 surrogates and hence will have less variability than a single point pattern. Figure 12.5 shows the difference for each point between the mean of the markstat function for the surrogates and the markstat function for the data. There is no obvious pattern in these differences.

Table 12.4: Boese data: points selected for calculation of DBH differentiation (2)

<table>
<thead>
<tr>
<th>Point number</th>
<th>x</th>
<th>y</th>
<th>Data DBH(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>4.95</td>
<td>10.49</td>
<td>0.16</td>
</tr>
<tr>
<td>25</td>
<td>12.27</td>
<td>36.16</td>
<td>0.26</td>
</tr>
<tr>
<td>58</td>
<td>37.83</td>
<td>37.09</td>
<td>0.29</td>
</tr>
<tr>
<td>83</td>
<td>21.85</td>
<td>25.49</td>
<td>0.13</td>
</tr>
<tr>
<td>95</td>
<td>27.61</td>
<td>7.72</td>
<td>0.27</td>
</tr>
<tr>
<td>140</td>
<td>42.31</td>
<td>11.1</td>
<td>0.37</td>
</tr>
</tbody>
</table>
Figure 12.5: Boese data: Residual plot by point number of the difference between the \textit{markstat} function for the data and the mean \textit{markstat} function for 1000 surrogates. The \textit{markstat} function is here defined as the mean of the marks of the 4 nearest neighbours.

The DBH differentiation (2) index was also calculated for each point in the data and all the surrogates. The distribution of this index for the data is shown in Figure 12.6(a); the majority of points have an index value in the range (0.2, 0.4). To make a comparison with the surrogates, a number of points were selected from the pattern and the distribution of the index from all 1000 surrogates was obtained. The selected points are shown in Figure 12.6(b) while their coordinates and index value are given in Table 12.4. Figure 12.7 shows the distributions of the 1000 index values from the surrogates for each of these selected points. They all appear to be similar to each other and the data’s index distribution. The data’s index at the relevant point is always contained within the distributions from the surrogates.

Figure 12.6: Boese data: (a) DBH differentiation (2) index for the data and (b) location of specific points for calculation of DBH differentiation (2) as shown in Figure 12.12
12.5. DATA ANALYSIS

Figure 12.7: Boese data: DBH differentiation (2) index for 1000 surrogates at specific points shown in Figure 12.6(b). The data values, given in Table 12.4, are shown as a circle on the horizontal axis

Boese data - second-order characteristics

Figure 12.8(a) shows the mark correlation functions, \( m(r) \), using the product of the marks \( (m_1m_2) \) as the test function. The theoretical value is 1 if the marks are independent. For the data, \( m(r) \) is less than 1 for \( r < 3 \) and greater than 1 otherwise, although the deviations from unity are not great. The smoothed mean of the mark correlation functions for each of the 1000 surrogates is close to 1 for all radii and the 95% envelope from the surrogates contains both the theoretical value and \( m(r) \) for the data for all radii. Note that the loess option has been used for smoothing and the translation option used for edge correction.

Figure 12.8(b) shows the mark correlation functions using \( \frac{1}{2}(m_1 - m_2)^2 \) as the test function. Here \( m(r) \) for the data oscillates around unity while the mean of the surrogates’ correlation functions is always less than 1 but not excessively so. Again the 95% envelope contains both the theoretical value and the data’s correlation function for all radii.

Both mark correlation function shown in Figures 12.8(a) and 12.8(b) indicate that the data and the surrogates have marks that are independent.

Figures 12.9(a) and 12.9(b) show the plots for the \( K_{multi} \) function as defined in \( \text{spatstat} \). The isotropic edge correction has been used in all estimates. In the first plot \( K_{multi} \) compares points with mark values less than the median (15; refer Table 12.1) with points with marks greater than the median. We see that the estimated \( K \) function for the data and the mean of the estimated functions for the surrogates are both very similar to the theoretical function.
CHAPTER 12. 2-D PATTERN WITH QUANTITATIVE MARKS

In the latter plot the comparison is between points with marks in the lower 25% with those with marks in the upper 25%. The mean of the surrogates’ $K_{multi}$ function is close to that of the data for radii less than 7 and falls below thereafter, where the data function is close to the theoretical curve. The 95% envelope from the surrogates contains the data $K_{multi}$ function for all radii while the theoretical curve only falls just outside for radii less than 2.

Figure 12.9: Boese data: $K_{multi}$ plots as given by spatstat. (a) group 1 consists of points whose mark $< 15$ while group 2 consists of points whose mark $> 15$, where 15 is the median and (b) group 1 consisting of points whose mark is less than the lower quartile and group 2 consisting of points whose mark exceeds the upper quartile.
Boese data - prediction using weighted means

The algorithm described in Section 2 calculated the predicted mark by a simple average of the marks at the connected vertices of the chosen point. This prediction can alternatively be calculated by a weighted average, with the weights being the normalised reciprocals of the distances of the connected vertices to the chosen point. For example, if the point chosen to be deleted has $k$ connected vertices and the Euclidean distance from the chosen point to connected vertex $i$ is $s_i$, then the weight given to point $i$ is $\frac{1}{\sum_{j=1}^{k} s_j}$.

The uplift procedure is suitably altered to take account of the weights as is the inverting step after resampling. The mark correlation and $K_{multi}$ functions together with the residuals from the markstat function were plotted after running the amended algorithm and are shown in Figures 12.10(a), 12.10(b), 12.11(a), 12.11(b) and 12.12. These plots show only slight differences to the corresponding plots shown previously so for the Boese data no advantage has been gained from using a more complicated prediction procedure. However this can be explained by the similarity of the marks of the nearest neighbours rather than a deficiency in the method.

Figure 12.10: Boese data with weighted prediction: Mark Correlation Function, using translate edge correction, based on (a) the product of the marks $(m_1m_2)$ and (b) half the square of the difference in the marks ie $\frac{1}{2}(m_1 - m_2)^2$

### 12.5.2 Spruce data

Summary statistics for Spruce data are shown in Table 12.5 and in contrast to Boese data the marks are consistently small, all values being in the range (0.16, 0.37) with a mean of 0.2504. Figure 12.13 shows the location of the trees, the smoothed mark plot, the mark histogram (which shows modes at approximately 0.25 and 0.33) and the mark correlation function which lies close to 1 for all radii.

Figure 12.14 shows the differences between actual and predicted marks; again no trend or
Figure 12.11: Boese data with weighted prediction: $K_{\text{multi}}$ plots as given by \textit{spatstat}. (a) group 1 consists of points whose mark < 15 while group 2 consists of points whose mark > 15, where 15 is the median and (b) group 1 consisting of points whose mark is less than the lower quartile and group 2 consisting of points whose mark exceeds the upper quartile.

Figure 12.12: Boese data with weighted prediction: Residual plot by point number of the difference between the $\text{markstat}$ function for the data and the mean $\text{markstat}$ function for 1000 surrogates. The $\text{markstat}$ function is here defined as the mean of the marks of the 4 nearest neighbours.
Table 12.5: Summary statistics for Spruce data marks

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.1600</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>0.2200</td>
</tr>
<tr>
<td>Median</td>
<td>0.2450</td>
</tr>
<tr>
<td>Mean</td>
<td>0.2504</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>0.2700</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.3700</td>
</tr>
</tbody>
</table>

pattern is present and the autocorrelation plot in Figure 12.14 confirms that we can consider the differences to be independent and thus able to be resampled.

Plots for four surrogates are shown in Figures 12.15 and 12.16 and are similar to the corresponding data plots.

**Spruce data - indices**

The nearest neighbour indices are shown in Table 12.6. The mean mark product for the data was 0.99 and the corresponding values for the surrogates vary between 0.97 and 1.04 with a mean of 1.00. The mean variogram index for the data was 1.118 which lies toward the upper and of the surrogate values (0.52, 1.37). The lower values for the surrogates indicate that the marks of a point and its nearest neighbour are generally more similar in the surrogates than in the data.

Table 12.7 shows the summary of the markstat function, again being the mean of the marks of the four nearest neighbours. The values for the data lie in (0.2025, 0.3150) with a mean of 0.2501. The mean of all 1000 surrogates at each point lie in (0.2334, 0.2908) with a mean of 0.2545. Again there is a reminder that a single pattern will have more extreme values than an average of 1000 values. The differences between the markstat function for the data and the mean of the function values from the surrogates for each point are shown in Figure 12.17. All the differences are small because the mark values are similar. There is no obvious pattern in the differences.

Figure 12.18(a) shows the distribution of the DBH differentiation (2) index values for the data. The majority of values are in the range (0.1, 0.2). The locations of four selected points are shown in Figure 12.18(b), with the coordinates and DBH (2) values given in Table 12.8. The distribution of the DBH(2) index values from the surrogates for these points are shown in Figure 12.19. All these distributions are similar to each other and the that from the data and the data index values lie within the distributions.

**Spruce data - second-order characteristics**

Figure 12.20(a) shows the mark correlation functions, \( m(r) \), using the product of the marks \( (m_1 m_2) \) as the test function. The theoretical value is 1 if the marks are independent. For the data, \( m(r) \) is close to 1 for all radii. The smoothed mean of the mark correlation functions for each of the
Table 12.6: Spruce data: Nearest neighbour indices. The mark product index for the data is 0.990 and the variogram index is 1.118

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Mean mark product index</th>
<th>Mean variogram index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.97</td>
<td>0.52</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>1.00</td>
<td>0.78</td>
</tr>
<tr>
<td>Median</td>
<td>1.00</td>
<td>0.86</td>
</tr>
<tr>
<td>Mean</td>
<td>1.00</td>
<td>0.87</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>1.01</td>
<td>0.95</td>
</tr>
<tr>
<td>Maximum</td>
<td>1.04</td>
<td>1.37</td>
</tr>
</tbody>
</table>

Table 12.7: Spruce data: Summary statistics for the markstat function defined as the mean mark of 4 nearest neighbours

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Data</th>
<th>Mean of surrogates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.2025</td>
<td>0.2334</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>0.2350</td>
<td>0.2472</td>
</tr>
<tr>
<td>Median</td>
<td>0.2475</td>
<td>0.2525</td>
</tr>
<tr>
<td>Mean</td>
<td>0.2501</td>
<td>0.2545</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>0.2625</td>
<td>0.2585</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.3150</td>
<td>0.2908</td>
</tr>
</tbody>
</table>

Table 12.8: Spruce data: points selected for calculation of DBH differentiation (2)

<table>
<thead>
<tr>
<th>Point number</th>
<th>x</th>
<th>y</th>
<th>Data DBH(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>10.6</td>
<td>7.7</td>
<td>0.30</td>
</tr>
<tr>
<td>25</td>
<td>13.1</td>
<td>25.9</td>
<td>0.17</td>
</tr>
<tr>
<td>85</td>
<td>36.2</td>
<td>7.3</td>
<td>0.14</td>
</tr>
<tr>
<td>105</td>
<td>44</td>
<td>25.3</td>
<td>0.14</td>
</tr>
</tbody>
</table>

Figure 12.13: Spruce data: (top left) tree locations with circle diameters scaled by mark ie trunk diameter (top right) plot of smoothed marks (bottom left) histogram of marks (bottom right) mark correlation function based on the product of the marks
Figure 12.14: Spruce data: differences between the marks of the deleted point and the mean position of connected vertices and the associated autocorrelations

Figure 12.15: Spruce data: The same graphs as in Figure 12.13 for two surrogates after lifting, resampling the differences and inverting
Figure 12.16: Spruce data: Two more surrogates after lifting, resampling and inverting

Figure 12.17: Spruce data: Residual plot by point number of the difference between the \textit{markstat} function for the data and the mean \textit{markstat} function for 1000 surrogates. The \textit{markstat} function is here defined as the mean of the marks of the 4 nearest neighbours.
Figure 12.18: Spruce data: (a) DBH differentiation (2) index for the data and (b) location of specific points for calculation of DBH differentiation (2) as shown in Figure 12.19.

Figure 12.19: Spruce data: DBH differentiation (2) index for 1000 surrogates at specific points shown in Figure 12.18(b). The data values, given in Table 12.8, are shown as a circle on the horizontal axis.
1000 surrogates is almost equal to 1 for all radii and the 95% envelope from the surrogates contains the theoretical value for all radii and \( m(r) \) for the data except for radii around 3. Note that the loess option has been used for smoothing and the translation option used for edge correction.

Figure 12.20(b) shows the mark correlation functions using \( \frac{1}{2}(m_1 - m_2)^2 \) as the test function. Here \( m(r) \) for the data is greater than 1 for radii less than 3 and less than 1 thereafter. The mean of the surrogates’ correlation functions is always less than 1 and converges toward 1 as the radius increases. The 95% envelope contains both the theoretical value and the data’s correlation function for all radii.

Both mark correlation functions shown in Figures 12.20(a) and 12.20(b) indicate that the data and the surrogates have marks that are independent.

Figures 12.21(a) and 12.21(b) show the plots for the \( K_{\text{multi}} \) function. The isotropic edge correction has been used in all estimates. In the first plot \( K_{\text{multi}} \) compares points with mark values less than the median (0.245) with points with marks greater than the median. We see that the mean of the estimated \( K \) functions from the surrogates lies below that from the data. The theoretical function lies above the 95% envelope.

In the latter plot the comparison is between points with marks in the lower 25% with those with marks in the upper 25%. The mean of the surrogates’ \( K_{\text{multi}} \) function is close to that of the data for all radii. The theoretical curve is again generally above the 95% envelope from the surrogates.

![Figure 12.20](image-url)
12.5. DATA ANALYSIS

Figure 12.21: Spruce data: Kmulti plots as given by spatstat. (a) group 1 consists of points whose mark \(< 0.245\) while group 2 consists of points whose mark \(> 0.245\), where 0.245 is the median and (b) group 1 consisting of points whose mark is less than the lower quartile and group 2 consisting of points whose mark exceeds the upper quartile.

12.5.3 Longleaf data

The number of trees in the pattern (584) was considered to be greater than required for the testing process so a subset of 105 trees was obtained. Figure 12.22 shows the location of the trees included in the analysis while Figure 12.23 shows the smoothed marks, mark histogram and mark correlation function. In contrast to the Boese data, where the marks were in the range (7, 40) with the majority in (10, 20), the Longleaf marks lie in (2, 68) and are more widely spread, as shown in Table 12.9. In particular there are a large number of trees with marks less than 10 so there is a mixture of immature and mature trees. Figure 12.22 shows that the majority of trees (mostly mature) are located in the north-west quadrant while the south-east quadrant is sparsely populated. The majority of immature trees are in the north-east quadrant.

These data are therefore more variable than in the previous two cases and more interesting from the viewpoint of testing the algorithm.

Figure 12.24 shows the differences between actual and predicted marks; no trend or pattern is present although there is some variation in the amplitude of the differences. The autocorrelation plot in Figure 12.24 confirms that we can consider the differences to be independent and thus able to be resampled.

Plots for four surrogates are shown in Figures 12.25 and 12.26. The histograms show that the surrogates’ marks are less uniform although the mark correlation functions are similar to those of the data.
Table 12.9: Summary statistics for Longleaf data marks

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>2.00</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>9.20</td>
</tr>
<tr>
<td>Median</td>
<td>22.80</td>
</tr>
<tr>
<td>Mean</td>
<td>26.68</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>42.30</td>
</tr>
<tr>
<td>Maximum</td>
<td>68.00</td>
</tr>
</tbody>
</table>

Figure 12.22: Longleaf data: location of trees

Figure 12.23: Longleaf data: (top left) tree locations with circle diameters scaled by mark ie trunk diameter (top right) plot of smoothed marks (bottom left) histogram of marks (bottom right) mark correlation function based on the product of the marks
Figure 12.24: Longleaf data: differences between the marks of the deleted point and the mean position of connected vertices and the associated autocorrelations

Figure 12.25: Longleaf data: The same graphs as in Figure 12.23 for two surrogates after lifting, resampling the differences and inverting
Figure 12.26: Longleaf data: Two more surrogates after lifting, resampling and inverting

Table 12.10: Longleaf data: Nearest neighbour indices. The mark product index for the data is 1.188 and the variogram index is 0.435

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Mean mark product index</th>
<th>Mean variogram index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.91</td>
<td>0.42</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>1.04</td>
<td>0.70</td>
</tr>
<tr>
<td>Median</td>
<td>1.07</td>
<td>0.78</td>
</tr>
<tr>
<td>Mean</td>
<td>1.07</td>
<td>0.78</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>1.10</td>
<td>0.86</td>
</tr>
<tr>
<td>Maximum</td>
<td>1.27</td>
<td>1.22</td>
</tr>
</tbody>
</table>

Table 12.11: Longleaf data: Summary statistics for the markstat function defined as the mean mark of 4 nearest neighbours

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Data Mean of resamples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>2.73</td>
</tr>
<tr>
<td>Lower quartile</td>
<td>12.05</td>
</tr>
<tr>
<td>Median</td>
<td>25.07</td>
</tr>
<tr>
<td>Mean</td>
<td>25.15</td>
</tr>
<tr>
<td>Upper quartile</td>
<td>36.12</td>
</tr>
<tr>
<td>Maximum</td>
<td>52.17</td>
</tr>
</tbody>
</table>
Table 12.12: Longleaf data points selected for calculation of DBH differentiation (2)

<table>
<thead>
<tr>
<th>Point number</th>
<th>x</th>
<th>y</th>
<th>Data DBH(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>156.1</td>
<td>38.7</td>
<td>0.52</td>
</tr>
<tr>
<td>10</td>
<td>48.9</td>
<td>67.8</td>
<td>0.22</td>
</tr>
<tr>
<td>52</td>
<td>151.9</td>
<td>127.5</td>
<td>0.46</td>
</tr>
<tr>
<td>60</td>
<td>66.5</td>
<td>150</td>
<td>0.80</td>
</tr>
</tbody>
</table>

Longleaf data - indices

The nearest neighbour indices are shown in Table 12.10. The mean mark product for the data was 1.188 and the corresponding values for the surrogates vary between 0.91 and 1.27 with a mean of 1.07. The mean variogram index for the data was 0.435 which lies close to the minimum of the surrogate values (0.42, 1.22). The higher values for the surrogates indicate that the marks of a point and its nearest neighbour are generally more varied in the surrogates than in the data.

Table 12.11 shows the summary of the markstat function, again being the mean of the marks of the four nearest neighbours. The values for the data lie in (2.73, 52.17) with a mean of 25.15. The mean of all 1000 surrogates at each point lie in (15.88, 47.95) with a mean of 30.32. The differences between the markstat function for the data and the mean of the function values from the surrogates for each point are shown in Figure 12.27. There is more variation in these differences that in the previous two examples and the differences are generally negative for the first 60 points and positive thereafter.

Figure 12.28(a) shows the distribution of the DBH differentiation (2) index values for the data. The majority of values lie fairly uniformly in the range (0.1, 0.7). The locations of four selected points are shown in Figure 12.28(b), with the coordinates and DBH (2) values given in Table 12.12. The distribution of the DBH(2) index values from the surrogates for these points are shown in Figure 12.29. These distributions show some variation corresponding to the appropriate data index value.

Longleaf data - second-order characteristics

Figure 12.30(a) shows the mark correlation functions, \( m(r) \), using the product of the marks \( m_1m_2 \) as the test function. The theoretical value is 1 if the marks are independent. For the data, \( m(r) \) increases from a value of 0.4 to just under 1 when the radius is 20 before declining to around 0.8 thereafter. The smoothed mean of the mark correlation functions from each of the 1000 surrogates has a similar shape but at a higher level. The 95% envelope from the surrogates contains the theoretical value for all radii and \( m(r) \) for the data except for radii less than 6. Again, the loess option has been used for smoothing and the translation option used for edge correction.

Figure 12.30(b) shows the mark correlation functions using \( \frac{1}{2}(m_1 - m_2)^2 \) as the test function. Here \( m(r) \) from the data is below the 95% envelope from the surrogates for radii less than 20. The
Figure 12.27: Longleaf data: Residual plot by point number of the difference between the \textit{markstat} function for the data and the mean \textit{markstat} function for 1000 surrogates. The \textit{markstat} function is here defined as the mean of the marks of the 4 nearest neighbours.

Figure 12.28: Longleaf data: (a) DBH differentiation (2) index for the data and (b) location of specific points for calculation of DBH differentiation (2) as shown in Figure 12.29.
mean of the surrogates’ correlation functions increases from about 0.7 to about 0.9 as the radius increases.

Both mark correlation function shown in Figures 12.30(a) and 12.30(b) indicate that the data and the surrogates have marks that are independent but there appear to be some differences between the data and the surrogates.

Figures 12.31(a) and 12.31(b) show the plots for the \( K_{multi} \) function. The isotropic edge correction has been used in all estimates. In the first plot \( K_{multi} \) compares points with mark values less than the median (22.8) with points with marks greater than the median. We see that the mean of the estimated \( K \) functions from the surrogates lies just above that from the data which almost coincides with the theoretical function, completely within the 95% envelope.

In the latter plot the comparison is between points with marks in the lower 25% with those with marks in the upper 25%. The mean of the surrogates’ \( K_{multi} \) function lies between that of the data and the theoretical curve. The 95% envelope from the surrogates is very wide for all radii. The problem here appears to be caused by comparing immature and mature trees which occupy different locations in the pattern.

12.6 Modifying the algorithm - resampling the location

The location of the points was fixed in the above analysis with only the mark being resampled. As a comparison the analysis was re-done with the locations also being resampled. This required
CHAPTER 12. 2-D PATTERN WITH QUANTITATIVE MARKS

Figure 12.30: Longleaf data: Mark Correlation Function, using the translate edge correction, based on (a) the product of the marks \( m_1 m_2 \) and (b) half the square of the difference in the marks ie \( \frac{1}{2}(m_1 - m_2)^2 \).

Figure 12.31: Longleaf data: Kmulti plots as given by spatstat. (a) group 1 consists of points whose mark < 22.8 while group 2 consists of points whose mark > 22.8, where 22.8 is the median and (b) group 1 consisting of points whose mark is less than the lower quartile and group 2 consisting of points whose mark exceeds the upper quartile.
12.6. MODIFYING THE ALGORITHM - RESAMPLING THE LOCATION

minor changes to the algorithm, the main alteration being the necessity to keep the locations inside the original window.

Both the Spruce and Longleaf data were re-analysed and the autocorrelation functions, including cross-correlations, shown in Figures 12.32 and 12.35, indicate that the difference vectors are independent.

Mark Correlation and $K_{multi}$ functions are shown in Figures 12.33(a) to 12.37(b). These graphs need to be compared with their corresponding graphs when the locations were fixed.

Taking the Spruce data first, the following observations can be made:

- The mark correlation functions, using the product of the marks, when the locations are resampled (Figure 12.33(a)) are similar to those shown in Figure 12.20(a) with the main difference being that the function estimated from the data now lies completely within the 95% envelope from the surrogates.
  The correlation functions using the alternative test function (Figures 12.33(b) and 12.20(b)) show little difference.

- The $K_{multi}$ functions show better agreement between the data, the mean of the surrogates and the theoretical function when the locations are resampled. The mean of the surrogates closely follows the data function when the cut point for the subgroups is the median (Figures 12.34(a) and 12.21(a)) and the theoretical function is now just inside the upper envelope.
  The main difference when the cut points are the lower and upper quartiles is the wider envelope, resulting in the theoretical curve now lying inside the envelope. (Figures 12.34(b) and 12.21(b)).

Now looking at the Longleaf data:

- When the locations are resampled the mean mark correlation function from the surrogates, using the product of the marks, is much flatter than before (Figures 12.36(a) and 12.30(a)) and the estimated function from the data is further outside the envelope.
  The correlation functions using the alternative test function (Figures 12.36(b) and 12.30(b)) show little difference.

- For the $K_{multi}$ functions with resampled locations, the mean of the surrogates more closely follows the data function, and the theoretical curve, when the cut point for the subgroups is the median (Figures 12.37(a) and 12.31(a)).
  There is little difference when the cut points are the lower and upper quartiles.
Figure 12.32: Spruce data: autocorrelations for the differences in co-ordinates and marks when locations and marks are resampled

Figure 12.33: Spruce data with resampled location: Mark Correlation Function, using the translate edge correction, based on (a) the product of the marks \(m_1 m_2\) and (b) half the square of the difference in the marks \(\frac{1}{2}(m_1 - m_2)^2\)
12.6. MODIFYING THE ALGORITHM - RESAMPLING THE LOCATION

Figure 12.34: Spruce data with resampled location: Kmulti plots as given by spatstat. (a) group 1 consists of points whose mark < 0.245 while group 2 consists of points whose mark > 0.245, where 0.245 is the median and (b) group 1 consisting of points whose mark is less than the lower quartile and group 2 consisting of points whose mark exceeds the upper quartile.

Figure 12.35: Longleaf data: autocorrelations for the differences in co-ordinates and marks when locations and marks are resampled.
Figure 12.36: Longleaf data with resampled location: Mark Correlation Function, using the translate edge correction, based on (a) the product of the marks \((m_1 m_2)\) and (b) half the square of the difference in the marks \(\frac{1}{2}(m_1 - m_2)^2\).

Figure 12.37: Longleaf data with resampled location: Kmulti plots as given by spatstat. (a) group 1 consists of points whose mark < 22.8 while group 2 consists of points whose mark > 22.8, where 22.8 is the median and (b) group 1 consisting of points whose mark is less than the lower quartile and group 2 consisting of points whose mark exceeds the upper quartile.
12.7 Conclusion

This chapter has shown how the resampling algorithm applies to 2-dimensional point processes with quantitative marks. The various diagnostics presented have indicated that the resampled patterns share important characteristics with the original patterns. This has been readily apparent with both Boese and Spruce data where the original patterns appeared to be homogeneous. In the case of the Longleaf data, where there was a mixture of immature and mature trees, the surrogates showed greater variability which highlighted the lack of homogeneity.

Resampling the locations as well as the marks appears to slightly improve the relationship between the Kmulti functions for the surrogates and the data. However there is no corresponding improvement for the Mark Correlation Function. Further investigation is required in order to ascertain whether it is sufficient to resample only marks.
Chapter 13

Alternative Methods for Resampling 2-D Point Processes

13.1 Introduction

This chapter examines alternative methods for resampling 2-dimensional spatial point patterns and compares summary functions with those obtained from surrogates using the wavelet lifting method.

13.1.1 Tiling methods

One of the first methods suggested for resampling spatial point processes was an adaption of the block resampling method used in time series. In the spatial context this involves either partitioning the observation window into congruent blocks or tiles (the fixed tile method) or selecting randomly located congruent blocks such that the total area of the blocks equals the area of the original observation window (the moving tile method). Toroidal wrapping is used in the latter method to ensure all points in the window have an equal chance of being selected in a block. [10]

Some work has been done in trying to determine the appropriate block size. There is a trade off between having a few large blocks which preserve dependence between points and having many blocks to better estimate variability of the summary function/statistic. In their paper, [22] Nordman et al.

"develop precise formulae for the optimal block sizes that minimize the mean squared error of the bootstrap variance estimator."

Alternatively, Loh [20] states

"A rule-of-thumb is to divide each dimension of the observation region into at least three parts, i.e. 9 blocks in 2D...This would ensure enough variability between bootstrap
samples. Of course, for correlation functions, the maximum value of the separation
distance $r$ at which these functions are estimated would influence the decision on block
size.

There are two ways of proceeding once the tiles have been selected. The subsampling method
resamples the tiles and calculates the summary function for each tile, assuming that the individual
tiles are representative of the observation window.

The block bootstrap method resamples the tiles and randomly joins them together to form an
area congruent to the original observation window. This new point pattern is assumed to share
similar characteristics to the data and the summary function is calculated.

Inherent problems with the subsampling method are the greater influence of the boundary of
each tile and calculation of the summary function for tiles where there may only be a few points.

The problem with the block resampling method is the creation of artificial patterns when areas
which may have originally been far apart are joined together.

13.1.2 Marked point method

The marked point method has been developed by Loh and Stein [19, 20]. The first step is to
assign a mark to each point which is a function of the distance to surrounding points. Randomly
located blocks are then selected and the marks for all points within the blocks are summed. This
is repeated a large number of times and the results used to estimate $K(r)$, the reduced second
moment function.

Some advantages claimed for this method are:

1. since the blocks are not joined together, artificial patterns are only created from toroidal
wrapping,

2. there is a computational advantage as the marks are calculated once only for each value of $r$
and based on the whole window, and

3. information about points outside the blocks is lost in both the subsampling and block re-
sampling methods. In contrast, the marked point method retains some information about
surrounding points through the marks.

In their papers Loh and Stein use two methods for calculating the marks:

1. In [19] the marks are the weights used for the isotropic correction. Suppose we are calculating
the contribution to a summary function from a point $x$ and we have a point $y$ at a distance
$h$ from $x$. Then the weight at $x$ for $y$ is reciprocal of the proportion of the circle centred at
$x$ with radius $h$ that is contained within the observation window $A$.

The marks are therefore functions of radii. For a given radius $r$ the mark at a specific point
is the sum of the weights \( w_A(x, y) \) of all points within \( r \). Equation (7) of \([19]\) gives

\[
m_x = \sum_{y : y \neq x} 1 \{ |y - x| \leq r \} w_A(x, y) \tag{13.1}
\]

The points with their marks are resampled with a chosen resampling scheme, for example randomly chosen blocks with toroidal wrapping. Suppose we have block \( A_i, \{i = 1, ..., N\} \) with points \( x_{ij}, \{j = 1, ..., n_i\} \) each with a mark \( m_{ij} \) given by

\[
m_{ij} = \sum_{y : y \neq x_{ij}} 1 \{ y \in A : |y - x_{ij}| \leq r \} w_A(x_{ij}, y) \tag{13.2}
\]

If \( a \) is the area of the window \( A \) and \( \sum n_i (\sum n_i - 1) / a \) is used to estimate \( \lambda^2 a \) then an estimate of \( K \) is given by

\[
\tilde{K}(r) = a \frac{\sum_{n_i} \sum_{i=1}^{n_i} m_{ij}}{\sum n_i (\sum n_i - 1)} \tag{13.3}
\]

2. In \([20]\) the marks are the number of points within annuli with inner and outer radii \((r - dr)\) and \((r + dr)\). Thus the mark associated with point \( x \) is

\[
\sum_{y \in D : y \neq x} 1 \{ |x - y| \in (r - dr, r + dr) \}
\]

being the number of points that are approximately \( r \) units from \( x \).

The procedure adopted in \([20]\) is

(a) calculate the mark \( m_i \) for each of the \( N \) points in the data
(b) estimate the \( K \) function for the data using \( \hat{K} = \frac{1}{N} \sum_{i=1}^{N} m_i \)
(c) resample the points by randomly placing blocks over the observation region and recording how many times point \( i \) is resampled, \( n_i^* \). Then \( N^* = \sum_{i=1}^{N} n_i^* \)
(d) the bootstrap estimate of \( K \) is \( \hat{K}^* = \frac{1}{N^*} \sum_{i=1}^{N} (n_i^* \times m_i) \)
(e) steps (c) and (d) are repeated a large number of times to obtain a confidence interval for \( \hat{K}^* \)

Instead of calculating the number of additional points that are approximately \( r \) units from \( x \), we can define the marks by the number of points within \( r \) units from \( x \).

### 13.2 Analysis of block resampling

This section compares summary functions obtained from block resampling with those obtained using wavelet lifting. The moving tile method was used to select the tiles.
13.2.1 **Non-homogeneous Poisson point pattern**

The first point pattern analysed was the non-homogeneous Poisson pattern used previously and defined by the intensity function \( \lambda(x, y) = 200x + 200y \) with 193 points. The pattern is shown in Figure 13.1 where we note the gradual increase in intensity toward the north-east corner. The observation window was surrounded by eight copies of the data and the following procedure was repeated 100 times:

1. twenty five square blocks of side 0.2 were randomly located in the square with corners \((-0.2, -0.2), (1.2, -0.2), (1.2, 1.2), (-0.2, 1.2)\). Figure 13.2(a) shows one selection.

2. these blocks were randomly joined together to create a point pattern located in the original unit square window. The number of points in the pattern will vary between resamples. Figure 13.2(b) shows one surrogate produced from joining 25 tiles.

3. the inhomogeneous \( K \) function and the Pair Correlation Function are calculated for each surrogate.

One of the surrogates, shown in Figure 13.2(b), has points more spread out on the unit square than the original pattern where points are more concentrated in the north-east corner.

To see whether the other surrogates also show patterns that are more uniform than the data we can compare the distribution of quadrat counts for the surrogates with the actual quadrat counts for the data. The region of observation was partitioned into 16 quadrats of side length 0.25. Figures 13.3 and 13.4 show the distributions of quadrat counts for the surrogates with the data values given above each histogram for comparison. It is apparent that the counts for the surrogates are highly variable with a range exceeding 20 in all quadrats. In some cases the quadrat count for the data lies near the middle of the distribution from the surrogates while in other cases it is toward the extremeties. This is particularly noticeable in quadrat \((4, 4)\) which is the most densely
Figure 13.2: Non-homogeneous Poisson pattern: (a) data (in area delineated by dashed lines) and 25 randomly placed tiles (the points located outside the observation window allow toroidal wrapping) and (b) a surrogate obtained from randomly joining the tiles similar to those in (a) populated quadrat in the data with 28 points. The mean count for this quadrat in the surrogates is 12 with the upper quartile being 15 and the maximum value 28. We can conclude from an examination of the quadrat counts that the tiling method has not produced point patterns that are similar to the data. The analysis can be extended by looking at the estimates of $K(r)$ and the Pair Correlation Function, $g(r)$.

Figure 13.5(a) shows the estimates of $K(r)$ from the data, $\hat{K}(r)$, as well as the mean of the surrogates’ $K$ function, $\bar{K}(r)$. The 95% envelope and the theoretical function are also shown. Both $\hat{K}(r)$ and $\bar{K}(r)$ are similar with the greatest difference occurring for radii between 0.10 and 0.15 where $\hat{K}(r)$ lies close to the lower extreme of the 95% envelope.

While the Pair Correlation Function (PCF) from the data is similar to that from the surrogates for radii less than about 0.07, as shown in Figure 13.5(b), there are large differences for radii greater than 0.08. While the mean of the PCFs from the surrogates is approximately equal to the theoretical value of 1 for a homogeneous Poisson process at those radii, the PCF from the data lies above the 95% envelope on a number of occasions.

By itself, the $K$ function plot would appear to indicate that the point patterns in the surrogates are similar to the data. However the PCF plot implies that the surrogates differ from the data and the quadrat count analysis supports this conclusion. Therefore the tiling method does not appear to be suitable for resampling non-homogeneous Poisson point processes.

For comparison, the estimates of $K(r)$ and the PCF using wavelet lifting to generate the surrogates are shown in Figures 13.6(a) and 13.6(b). While there is little difference between the estimates of $K(r)$, comparison of the PCFs indicates that wavelet lifting is better able to reproduce characteristics of the non-homogeneous data than block resampling.
13.2. ANALYSIS OF BLOCK RESAMPLING

Figure 13.3: Non-homogeneous Poisson pattern: distribution of quadrat counts for surrogates using block resampling. Half of the quadrats are shown with the data value.

Figure 13.4: Non-homogeneous Poisson pattern: distribution of surrogate quadrat counts for the remainder of the window, using block resampling.
Figure 13.5: Block resampling (tiling) a non-homogeneous Poisson pattern: estimates of (a) $K(r)$ and (b) the Pair Correlation Function

Figure 13.6: Wavelet resampling a non-homogeneous Poisson pattern: estimates of (a) $K(r)$ and (b) the Pair Correlation Function
13.2.2 Matern Cluster point pattern

A Matern Cluster point pattern is the second pattern analysed. In this type of pattern the centres of clusters are distributed as a Poisson pattern and then clusters are distributed around the centres. The parameters therefore consist of the intensity of the clusters (set at 25), the radius of the clusters (0.1) and the mean number of points per cluster (10). These parameters were used previously and generated a pattern with 198 points, shown in Figure 13.7.

As with the non-homogeneous pattern, 25 tiles of side 0.2 were randomly selected and rearranged to create a pattern in the unit square. One of the 100 tile selections is shown in Figure 13.8(a) with the a typical surrogate shown in Figure 13.8(b). Figure 13.9(a) shows the estimates of $K(r)$ from the data together with the mean and the 95% envelope from the surrogates. The theoretical function $K(r) = \pi r^2$ is also shown for comparison. It is evident that the estimates of $K(r)$ from the surrogates are different to that from the data except for radii less than approximately 0.05. Figure 13.9(b) shows the estimated Pair Correlation Function (PCF) from the data together with the mean and the 95% envelope from the surrogates. The PCF from the data lies above the upper extreme of the 95% envelope for radii between approximately 0.05 and 0.15.

The analysis was repeated using 100 blocks of side 0.1. Figure 13.10(a) shows one of the selections with the corresponding pattern formed by randomly joining the tiles together shown in Figure 13.10(b). Figure 13.11(a) shows the estimates of $K(r)$ and the surrogates appear to be less like the data than those obtained from using 25 tiles. The estimates of $K(r)$ from the surrogates are similar to that expected from a Poisson pattern. This is also noticeable with the PCF plot in Figure 13.11(b). Here the PCF from the data lies outside the upper extreme of the 95% envelope.
Figure 13.8: Matern Cluster pattern: (a) data (in area delineated by dashed lines) and 25 randomly placed tiles and (b) a surrogate obtained from randomly joining the tiles similar to those in (a)

Figure 13.9: Matern Cluster pattern: estimates of (a) $K(r)$ and (b) the Pair Correlation Function using 25 tiles
13.3 Analysis of the marked point method

As described earlier, this method calculates marks for each point in the data and uses these marks to estimate $K(r)$. Therefore the comparison of the method with wavelet lifting has two aspects:

1. comparison of the distribution of marks calculated for the data with the distribution of marks obtained from the surrogates generated by wavelet lifting, and

2. comparison of the estimates of $K(r)$

Two methods of calculating the marks are examined - the number of points within given radii and the weights used for isotropic correction, as previously described. These methods are used for both the non-homogeneous and Matern cluster data.
Figure 13.11: Matern Cluster pattern: estimates of (a) $K(r)$ and (b) the Pair Correlation Function using 100 tiles

Figure 13.12: Matern Cluster pattern: estimates of (a) $K(r)$ and (b) the Pair Correlation Function using two-stage resampling after resampling the differences in each cluster. Compare with Figures 13.9(a) and 13.9(b). (Parameters $\kappa = 25$, $R = 0.1$ and $\mu = 10$)
13.3.1 Non-homogeneous Poisson pattern - marks as the number of points within radii

Firstly we look at the non-homogeneous Poisson pattern with the marks calculated by the number of points within specified radii.

The marks are calculated for the data and for 1000 surrogates. The distribution of marks for radii \((0.01, 0.02, \ldots, 0.14)\) can be examined by the use of boxplots. Figures 13.18 and 13.19 show the boxplots for the data and surrogates for these radii. It is apparent that the distributions are similar in each case, although the boxplots from the surrogates have increasing numbers of outliers as the radius increases. Overall, however, it appears that the surrogates share the characteristics of the data.

The marked point method is then used to estimate \(K(r)\) from the data. A total of 999 blocks was selected and the 95% confidence interval found from

\[
[2 \hat{K} - 2 \hat{K}^{*}_{25}, 2 \hat{K} - 2 \hat{K}^{*}_{975}] 
\]

where \( \hat{K} \) is the estimate from the data and \( \hat{K}^{*}_{i} \) is the \( i^{th} \) ordered bootstrapped value.

The mean mark was calculated for each radius for each of 1000 wavelet lifted surrogates and \(K(r)\) was estimated by calculating the mean of these mean marks. It is apparent from Figure 13.13(a) that the mean of the estimates of \(K(r)\) from the marked point bootstraps agrees closely with that obtained from the data. Also the mean of the estimates of \(K(r)\) from each of the surrogates generated from ordinary wavelet lifting is very similar. However, Figure 13.13(b) shows that when the surrogates are generated from the two-stage wavelet lifting method the estimate of \(K(r)\) is consistently higher than from the marked point method.

Note that the vertical scale in Figure 13.13(a) is totally different to that in Figure 13.6(a). Presumably I have omitted a scale factor when implementing the marked point method where the marks consist of the number of additional points within radii, rather than points within annuli. However the purpose of obtaining estimates of \(K(r)\) was to compare results from the data, resamples from the data and surrogates using wavelet lifting and since the same method was applied in all three situations the discrepancy is not critical.

13.3.2 Non-homogeneous Poisson pattern - marks as the isotropic correction weights

Now we look at the non-homogeneous Poisson pattern with the marks given as the weights used for isotropic correction.

Again the distribution of marks for radii \((0.01, 0.02, \ldots, 0.14)\) can be examined by the use of boxplots. Figures 13.20 and 13.21 show the boxplots for the data and surrogates for these radii.
Figure 13.13: Non-homogeneous Poisson pattern: estimates of $K(r)$ using the marked point method (Bootstrap mean) where the marks are the number of points with radius $r$ from each point. Also shown is the mean of the estimates from each of the surrogate patterns using the wavelet lifting method. The surrogates used in (a) came from ordinary wavelet lifting while those used in (b) came from the two-stage cluster approach.

As before, it is apparent that the distributions are similar in each case, although the boxplots from the surrogates have increasing numbers of outliers as the radius increases. Overall, however, it appears that the surrogates share the characteristics of the data.

The estimates of $K(r)$ are shown in Figures 13.14(a) and 13.14(b). Comparing the results from ordinary lifting, shown in 13.14(a), we see all three estimates are very similar for radii less than 0.06 whereas, for higher radii, the estimate from the wavelet lifted surrogates is closer to the estimate from the data than the bootstrap estimates from the marked point method.

Figure 13.14(b) shows again that the estimate of $K(r)$ using the two-stage approach is consistently higher than from the marked point method.

13.3.3 Non-homogeneous Poisson pattern - marks as the number of points within annuli

Since Loh [20] uses the number of points separated by distance approximately $r$ as the mark definition it is useful to see what this produces for the non-homogeneous data.

The actual mark definition adopted was the number of points within distances (0, 0.01) to (0.13, 0.14) with respective radii (0.005, 0.015, ..., 0.135). These marks were assigned to the relevant points and the mean mark for each radius calculated. Then 999 blocks were resampled with the number of points and the sum of the marks recorded for each block. The estimate of $K(r)$ was obtained by dividing the total sum of the marks for all resamples by the total number of points.

Then the number of points within the above distances was calculated for each of 100 surrogates...
13.3. ANALYSIS OF THE MARKED POINT METHOD

Figure 13.14: Non-homogeneous Poisson pattern: estimates of $K(r)$ using the marked point method (Bootstrap mean) where the marks are the isotropic correction weights. Also shown is the mean of the estimates from each of the surrogate patterns using the wavelet lifting method. The surrogates used in (a) came from ordinary wavelet lifting while those used in (b) came from the two-stage cluster approach.

and the mean mark obtained for each radius. The results are shown in Figure 13.15(a). We see that the mean of the bootstraps closely follows the data (perhaps too closely) whereas the mean value from the ordinary wavelet lifted surrogates is smoother. The two methods follow the data for radii less than 0.08.

The estimate of $K(r)$ using the two-stage approach, shown in Figure 13.15(b) is again higher than from the marked point method. This estimate is a better fit at the larger radii, but poorer at the smaller radii, compared to the estimate using ordinary wavelet lifting.

Although Loh calls this an estimate of $K(r)$, the vertical scale is again vastly different to that obtained from other methods. Loh [20] states that the summation of the number of points within distance approximately $r$ is denoted $DD(r)$ and used in the estimation of the two-point correlation function and therefore not associated with the usual definition of $K(r)$.

13.3.4 Matern Cluster pattern - marks as the number of points within radii

Again the distribution of marks for the data and surrogates from wavelet lifting are summarized by boxplots. These show, in Figures 13.22 and 13.23, that apart from some differences in skewness the distributions are very similar. Outliers only occur at the lower radii.

Figure 13.16 shows the estimates of $K(r)$ and we see that the mean of the estimates from the marked point bootstraps agrees closely with the estimate from the data for radii below 0.08 and falls below thereafter. The estimate from the wavelet surrogates is consistently above the estimate from the data but they converge at higher radii.
Figure 13.15: Non-homogeneous Poisson pattern: estimates of $K(r)$ using the marked point method (Bootstrap mean) where the marks are the number of points within approximate distance $r$. Also shown is the mean of the estimates from each of the surrogate patterns using the wavelet lifting method. The surrogates used in (a) came from ordinary wavelet lifting while those used in (b) came from the two-stage cluster approach.

Comparison with Figure 13.12(a) shows that better results are obtained with spatstat’s Kest function. Note again the difference in vertical scales in Figures 13.16 and 13.12(a).

Figure 13.16: Matern Cluster pattern: estimates of $K(r)$ using the marked point method where the marks are the number of points with radius $r$ from each point. Also shown is the mean of the estimates from each of the surrogates using the wavelet lifting method.

13.3.5 Matern Cluster pattern - marks as the isotropic correction weights

The boxplots for the distribution of marks, shown in Figures 13.24 and 13.25, are very similar to those where the marks were based on the number of points within radii. A notable difference, however, is the number of outliers with the wavelet lifted surrogates.
13.4. CONCLUSION

The estimate of $K(r)$ using the marked point bootstrap is virtually identical with the estimate from the data, as shown in Figure 13.17. The estimate from the wavelet surrogates is above the other estimates for radii less than 0.1 but closer than that obtained with the previous mark definition.

![Figure 13.17: Matern Cluster pattern: estimates of $K(r)$ using the marked point method where the marks are the isotropic correction weights. Also shown is the mean of the estimates from each of the surrogates using the wavelet lifting method.](image)

13.4 Conclusion

This chapter has compared spatial pattern summary functions obtained from surrogates generated by the wavelet lifting method with those obtained by alternative methods.

The block resampling method joined randomly selected blocks of data to produce a resample in the same observation region as the data. The resulting estimates of $K(r)$ and Partial Correlation Function were not similar to estimates from the data for either of the sample non-homogeneous Poisson or Matern Cluster point patterns. Therefore the block resampling method does not appear to be a realistic method for resampling spatial point patterns.

The marked point pattern appeared to provide estimates of $K(r)$ consistent with the data. However it doesn’t yield point patterns that could be realisations of the same process that produced the data. This may be potentially less useful than the wavelet lifting method that does produce such realisations.
Figure 13.18: Non-homogeneous Poisson pattern: comparison of marks for points in the data and wavelet lifted surrogates (using the two-stage cluster approach). Marks are the number of points within the indicated radius.
Figure 13.19: Non-homogeneous Poisson pattern: comparison of marks for points in the data and wavelet lifted surrogates (using the two-stage cluster approach). Marks are the number of points within the indicated radius.
Figure 13.20: Non-homogeneous Poisson pattern: comparison of marks for points in the data and wavelet lifted surrogates (using the two-stage cluster approach). Marks are the weights used in the isotropic correction.
Figure 13.21: Non-homogeneous Poisson pattern: comparison of marks for points in the data and wavelet lifted surrogates (using the two-stage cluster approach). Marks are the weights used in the isotropic correction.
Figure 13.22: Matern Cluster pattern: comparison of marks for points in the data and wavelet lifted surrogates. Marks are defined as the number of points within the indicated radius.
13.4. CONCLUSION

Figure 13.23: Matern Cluster pattern: comparison of marks for points in the data and wavelet lifted surrogates. Marks are defined as the number of points within the indicated radius.
Figure 13.24: Matern Cluster pattern: comparison of marks for points in the data and wavelet lifted surrogates. Marks are the weights used in the isotropic correction.
Figure 13.25: Matern Cluster pattern: comparison of marks for points in the data and wavelet lifted surrogates. Marks are the weights used in the isotropic correction.
Chapter 14

Conclusion

The aims of the thesis as stated in Chapter 1 were to apply wavelet transforms, or variations thereof, to different types of dependent data, examine the resulting coefficients for independence, resample if independent, invert the transform and thus produce surrogate data that were then tested for similarity with the original data.

The following points summarise the results:

1. The wavelet packet transform, using the lifting technique, was used for AR(1) and MA(1) time series. It was found that the surrogates were similar only when the data came from a model with a positive parameter. Especially poor results were obtained with MA(1) models with negative parameter.

2. The same method was used with one-dimensional point processes and the surrogates appeared to be similar to the data in each of the cases considered - homogeneous, non-homogeneous and clustered.

3. A simple method for producing surrogates of two-dimensional data was developed based on the wavelet lifting philosophy. Initial testing of the method gave encouraging results.

4. Application of the method to a non-homogeneous pattern with linear intensity function showed that the surrogates were similar to the data but poor results were obtained with a bivariate intensity function.

5. A variation of the method was used successfully with clustered patterns. In these cases the centre of each cluster is obtained and surrogates of the cluster centres are generated. The deviations of each point around its cluster centre are resampled, thus producing surrogates of clustered data. This procedure was termed the two-stage resampling method.

6. Application of the two-stage method to the previously considered non-homogeneous pattern
with bivariate intensity function gave mixed results but may be useful in individual circumstances.

7. Marked point patterns provide more information about points than just position. Initially, qualitatively marked points were investigated with three possible methods for producing the surrogates. The first, termed superposition, separated the data by mark, used the previously developed basic method for producing surrogates of 2-D data and then combined the surrogates. The second method randomly reallocated marks amongst points whose location had been resampled. The final method used two differences - the difference between the location of a point and the mean locations of connected points of the same type or different type. None of the methods was best in terms of all the diagnostics but the superposition and two-differences methods appeared to be most promising.

8. These two methods were applied to a real-life example with trees of two species where the two-differences method proved superior to the superposition method.

9. Alternative methods were also investigated with quantitatively marked points. Firstly, the basic method was applied only to the marks while the alternative resampled both the location and the mark. Three real-life examples from forestry were analysed and encouraging results were obtained with two examples where the marks were relatively consistent. The third example contained a mixture of immature and mature trees and the diagnostics indicated that the surrogates were not similar to the data. Some improvement was obtained by resampling the location in addition to the mark but further work is required to ascertain whether resampling only marks is adequate in the majority of cases.

10. Finally, the developed methods were compared with two alternatives - block resampling and the marked point method. The former proved to be inadequate for producing surrogates but the latter proved to give similar or better results compared to the methods developed in this thesis. However, sometimes the marked point method appeared to give results that may have followed the data too closely and it doesn’t produce surrogate patterns, only estimators of a second-order summary function.
Bibliography


