Optimal design for generalized linear models with a multinomial response

Gwenda Thompson

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.
I, Gwenda Piera Thompson, declare that this thesis, submitted in fulfillment 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. This document has not been submitted for qualifications at any other academic institution.

Gwenda Thompson

May 2010
Acknowledgment

This thesis would not have been possible without the support of my family, friends and colleagues.

I wish to express my sincere gratitude to Associate Professor Ken Russell for providing this opportunity and being a patient and supportive supervisor. A special thank you to Dr Damian Collins for all his help and friendship. Thanks to Professors David Steel and David Griffiths for help and advice and to the staff of the School of Mathematics and Applied Statistics.

Funding for my scholarship has been provided by the Australian Research Council and grants were awarded by the Statistical Society of Australia, the Centre for Statistical and Survey Methodology, Design and Analysis of Experiments Conference and Industry and Investment NSW.

I wish to thank my parents, family and most importantly, my husband, Mark.
Abstract

The objective of optimal experimental design is to determine the values of the predictor variables at which to take observations in order to maximize the quality of data obtained by the experiment. When the response variable is, at least approximately, independent and normally distributed with a constant variance, an optimal design can be obtained by accessing the prolific body of established classical design techniques. When these assumptions are not satisfied, the design solution is not easily obtainable.

The difficulty in designing an experiment for a response with a nonlinear model is that the optimal design depends upon the values of the unknown parameters of the model, but the model parameters are estimated from the results obtained after running the experiment. This has been a major factor contributing to the complexity, and hindering the development, of design methodology for nonlinear models. As a consequence, the majority of results, including those in this thesis, are obtained utilizing numerical techniques.

We examine two design methods for a multinomial response, utilizing the generalized linear model framework. One technique, D-optimality, is based on minimizing the volume of the confidence ellipsoid of the parameter estimates and the other, IMSE-optimality, is based on minimizing the mean squared error of the estimated probabilities of occurrence of the possible responses. These methods are developed using point estimates of the parameters of the linear
predictor. When estimates are unavailable, we use partial knowledge about the parameters to make designs more robust to uncertainty in the parameter values. We also develop designs which allow for some uncertainty in the model by examining different types of predictors.

The experimental design space is shown to play an important role, especially in IMSE-optimality, and a quantitative method is presented to determine the design space based on the limits of the underlying multinomial probabilities.

Although the two design methods examined are based on different criteria, our simulations strongly suggest that they are asymptotically equivalent. The D-optimal design algorithm is shown to be faster to implement, compared to the IMSE-optimal design algorithm, and is free to determine the number of support points. The IMSE-optimal approach requires more computation as it requires taking expected values of estimated probabilities and specifying the number of observations at each support point.

These design techniques are applied to a practical example, a dose response experiment, and we develop design methods allowing for some uncertainty in the form of the predictor and the values of the parameter estimates.
## Contents

1 Introduction ................................................. 1  
1.1 Experimental Design ...................................... 1  
1.2 Linear Models ........................................... 3  
1.3 Generalized Linear Models ............................... 4  
  1.3.1 Random Component ................................. 6  
  1.3.2 Systematic Component .............................. 8  
  1.3.3 Link Function ...................................... 9  
1.4 Poisson and Binomial Distributions ..................... 10  
1.5 Multinomial Distribution ............................... 12  
1.6 Multinomial GLM Components ........................... 14  
1.7 Discrete Choice Models ............................... 17  
1.8 Research Outline ................................... 18  

2 Estimation ................................................. 21  
2.1 Maximum Likelihood Estimation ...................... 22  
2.2 Likelihood Function .................................. 23  
2.3 Score Function ....................................... 24  
2.4 MLE for Multinomial Distribution .................. 25  
  2.4.1 Implicit Solutions to MLE ...................... 27  
2.5 Fisher’s Information Matrix ......................... 29
# 5 IMSE-Optimal Design

5.1 Mean Squared Error .................................................. 125
5.2 Maximum Likelihood Estimators ..................................... 126
   5.2.1 Approach for Infinite Estimates ................................. 128
5.3 Maximum Penalized Likelihood Estimators ......................... 129
   5.3.1 Exact Convergence of Estimators ............................ 131
5.4 Minimization Criteria ............................................... 132
   5.4.1 Design Space .................................................. 132
   5.4.2 Mean Squared Error .......................................... 136
   5.4.3 Integrated Mean Squared Error ................................. 138
5.5 Methods to Find Optimal Design ................................... 139
   5.5.1 Grid Search .................................................. 140
   5.5.2 Minimax ..................................................... 143
   5.5.3 Optimization Routine ......................................... 145
   5.5.4 Comparing Methods ............................................. 147
5.6 Varying the Design Space .......................................... 151
5.7 Mean Square Error Components .................................... 156
5.8 Design Location Change ............................................. 160
5.9 MSE and IMSE-Optimal Design ...................................... 163
5.10 Three Support Points ............................................... 166
5.11 Unequally Weighted Support Points ................................ 170
5.12 IMSE- vs D-Optimal Design ....................................... 171
5.13 Summary ............................................................. 174

# 6 Immunization Example .................................................. 177

6.1 Introduction ......................................................... 177
6.2 Original Experiment ................................................ 178
6.3 Parameter Estimates ................................................ 181
   6.3.1 Numerical Convergence of Estimators ....................... 184
6.4 D-Optimal Design ................................................... 186
   6.4.1 Design Space ................................................ 186
### List of Figures

4.1 Contour plot (for \(x_1 < x_2\)) of values of the determinant of the information matrix for \(\mathbf{\beta} = (0, 1, 0, 1)^T\). .......................... 65

4.2 Shaded area showing where constrOptim diverges for \(\mathbf{\beta} = (0, 1, 0, 1)^T\). 68

4.3 Box plot of first and second support points \((x_1, x_2)\), top and bottom charts respectively, obtained from 10,000 randomly generated \(\mathbf{\beta} \sim\) Uniform. .......................... 73

4.4 Efficiency of range designs compared to D-optimal designs for random generated \(\mathbf{\beta} = (b_1, b_2, b_3, b_4)^T\) where \(b_1 \sim U(-2, 2)\), \(b_2 \sim U(-1, 3)\), \(b_3 \sim U(-2, 2)\) and \(b_4 \sim U(0, 4)\). .......................... 75

4.5 Distribution of optimal designs for randomly generated \(\mathbf{\beta} \sim\) Uniform. The left plot shows distribution of the top 10% of designs, ranked in terms of their D-efficiency, and the right plot shows the next top 10% of designs. .......................... 76

4.6 Efficiency of range design compared to D-optimal designs for 10,000 randomly generated \(\mathbf{\beta}\) from \(N(0, 0.5^2)\) in the top left graph, \(N(0, 1^2)\) in the top right graph, \(N(0, 2^2)\) in the bottom left graph and \(N(0, 4^2)\) in the bottom right graph. .......................... 79

4.7 Absolute difference between support points of range and mid-point designs for uniform distributions. .......................... 81

4.8 Absolute difference between support points of range and mid-point designs for normal distributions. .......................... 83

4.9 Derivative function of the D-optimal design for \(\mathbf{\beta} = (0, 1, 0, 1)^T\) with two support points \(\xi = (-1.3, 3)\). .......................... 85

4.10 Derivative function confirming the optimality of the D-optimal design for \(\mathbf{\beta} = (0, 1, 0, 1)^T\) over \(\chi = [-3, 3]\). .......................... 88

4.11 Derivative function confirming the optimality of the D-optimal design for \(\mathbf{\beta} = (0, 1, 0, 1)^T\) over \(\chi = [-5, 5]\). .......................... 89

4.12 Derivative function confirming the optimality of the D-optimal design for \(\mathbf{\beta} = (0, 1, 0, -1)^T\) over \(\chi = [-3, 3]\). .......................... 90

4.13 Derivative function confirming the optimality of the D-optimal design for \(\mathbf{\beta} = (0, 1, 0, -1)^T\) over \(\chi = [-5, 5]\). .......................... 91
4.14 Derivative function showing optimality of D-optimal design for
β = (0, 1, 1, 0, 1, 1)T with five support points. . . . . . . . . . . .

93

4.15 Standardized variance fixing z covariate at z = −4 in the top
left graph, z = −1.944 in the top right graph and z = 3 in the
bottom graph. . . . . . . . . . . . . . . . . . . . . . . . . . . .

94

4.16 Derivative function for the four point design in Table 4.8 for β =
(0, 1, 1, 0, 1, 1)T that occurred most often (66%) in the simulations. 96
4.17 Standardized variance fixing z covariate at z = −3.77 in the top
left graph, z = 2.634 in the top right graph and z = 3 in the
bottom graph. . . . . . . . . . . . . . . . . . . . . . . . . . . . .

97

4.18 Derivative function showing optimality of D-optimal design for
β = (0, 1, 1, 1, 0, 1, 1, 1)T with ten support points. . . . . . . . .

99

4.19 Standardized variance fixing z covariate at z = −4 in the top
left graph, z = −1.552 in the top right graph, z = −1.099 in
the middle left graph, z = −0.892 in the middle right graph,
z = −0.583 in the bottom left graph and z = 3 in the bottom
right graph. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100
4.20 Derivative function showing optimality of D-optimal design for
β = (1, 1, 1, 1, 1, 1)T with five support points over χ = [−3, 3]. . 102
4.21 Derivative function showing optimality of D-optimal design for
β = (1, 1, −1, 1, 1, −1)T with five support points over χ = [−3, 3]. 103
4.22 Derivative function showing optimality of D-optimal design over
χ = [−5, 5]. Left graph is for β = (1, 1, 1, 1, 1, 1)T and right
graph is for β = (1, 1, −1, 1, 1, −1)T . . . . . . . . . . . . . . . . . 104
4.23 Derivative function confirming the optimality of the D-optimal
designs, the left graph is for β = (1, 2, 3, 4)T and the right graph
is for β = (1, 1, 1, 6)T . . . . . . . . . . . . . . . . . . . . . . . . . 106
4.24 Derivative function confirming the optimality of the D-optimal
design for β = (1, 1, 1, 1, 1, 1)T . . . . . . . . . . . . . . . . . . . . 108
4.25 Derivative function confirming the optimality of the D-optimal
design for β = (2, 1, 1, 1, 1, 1, 1, 1)T . . . . . . . . . . . . . . . . . 110
4.26 Standardized variance for D-optimal designs for equivalent models η2 = η3 = 1 + x, showing model (A) from Table 4.12. . . . . 113
4.27 Standardized variance for D-optimal designs for equivalent models η2 = η3 = 1 + x, showing model (B) from Table 4.12. . . . . 114
4.28 Standardized variance for D-optimal designs for equivalent models η2 = η3 = 1 + x, showing model (C) from Table 4.12. . . . . 115
4.29 Standardized variance for D-optimal designs for equivalent models η2 = η3 = 1 + x + z, showing model (D) from Table 4.14. . . 117
x


4.30 Standardized variance for D-optimal designs for equivalent models \( \eta_2 = \eta_3 = 1 + x + z \), showing model (E) from Table 4.14. 

5.1 True probabilities for \( \beta = (0, 1, 0, -1)^T \), over a range of \( x \) values. 

5.2 Comparing MSE graphs over the design range \([-10, 10]\) for \( n = 5 \) at the support points \((-3, 4)\) for \( \beta = (0, 1, 0, 1)^T \) in the left graph (with \( \tilde{\pi}_2 = \tilde{\pi}_3 \)) and for \( \beta = (0, 1, 0, -1)^T \) in the right graph. 

5.3 Comparing the MSE for \( \beta = (0, 1, 0, 1)^T \) over the design range \([-10, 10]\) for support points \((-3, 4)\) in the left graph and for support points \((-3, 4)\) in the right graph. In both graphs, \( \tilde{\pi}_2 = \tilde{\pi}_3 \). 

5.4 Contour plot of weighted average integrated MSE for \( \beta = (0, 1, 0, 1)^T \), shows optimal design at \((-5.12, 4.86)\) but a secondary local minimum emerges at \( \xi = (-1.46, 4.12) \). 

5.5 Contour plot of weighted average integrated MSE for \( \beta = (0, 1, 0, -1)^T \), shows optimal design at \((-2.80, 2.80)\) with \( n = 5 \) and 70 intervals over the design space \([-7, 7]\). 

5.6 Contour plot of design values for \( \beta = (0, 1, 0, 1)^T \) and \( n = 25 \). The optimum design is at the global minimum \( \xi^* = (-5.12, 4.86) \) but a secondary local minimum emerges at \( \xi = (-1.46, 4.12) \). 

5.7 Comparison of optimal designs for \( \beta = (0, 1, 0, 1)^T \) over the design space \([-7.6, 6.3]\) as the number of observations at each support point increases. The left graph is for the smaller support point and the right graph is for the larger support point. 

5.8 Comparison of optimal IMSE for \( \beta = (0, 1, 0, 1)^T \) over the design space \([-7.6, 6.3]\) as the number of observations at each support point increases. 

5.9 Comparison of optimal designs for \( \beta = (0, 1, 0, -1)^T \) over the design space \([-7, 7]\) as the number of observations at each support point increases. The left graph is for the smaller support point and the right graph is for the larger support point. 

5.10 Comparison of optimal IMSE for \( \beta = (0, 1, 0, -1)^T \) over the design space \([-7, 7]\) as the number of observations at each support point increases. 

5.11 Comparison of the additional IMSE component for \( \beta = (0, 1, 0, 1)^T \) and \( n = 10 \) when the design space is widened. 

5.12 Comparison of the additional IMSE component for \( \beta = (0, 1, 0, -1)^T \) when the design space is widened. 

5.13 Comparison of optimal designs over two design spaces as the number of observations at each support point increases. The left graph is for \( \beta = (0, 1, 0, 1)^T \) and the right graph is for \( \beta = (0, 1, 0, -1)^T \).
5.14 Comparison average MSE of IMSE-optimal and non-optimal designs for $\beta = (0, 1, 0, 1)^T$ and $n = 10$. The left graph shows the IMSE-optimal designs $\xi^* = (-6.31, 7.41)$ over $\chi = [-5.3, 4]$ and $\xi^* = (-4.01, 3.89)$ over $\chi = [-19.2, 17.8]$. The right graph shows the same designs but with the design spaces reversed. 

5.15 Variance and squared bias components of MSE for IMSE-optimal design $(-4.3, 4.38)$ over the design space $[-7.6, 6.3]$ for $\beta = (0, 1, 0, 1)$ and $n = 10$. The left graph shows MSE $(\tilde{\pi}_1)$ components and the right graph shows MSE $(\tilde{\pi}_2)$ components. 

5.16 Variance and squared bias components of MSE for optimal IMSE design $(-2.43, 2.43)$ over the design space $[-7, 7]$ for $\beta = (0, 1, 0, -1)^T$ and $n = 10$. The left graph shows MSE $(\tilde{\pi}_1)$ components and the right graph shows MSE $(\tilde{\pi}_2)$ components. 

5.17 Variance and squared bias components of MSE at optimal design for $\beta = (0, 1, 0, 1)^T$ over the design space $\chi = [-7.6, 6.3]$. 

5.18 Variance and squared bias components of MSE at optimal design for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$. 

5.19 Contour plot of optimal design for $\beta = (0, 1, 0, 1)^T$ and $n = 42$, which shows the optimum design at the global minimum $\xi^* = (-1.25, 4.31)$ and a secondary minimum at $\xi = (-5.61, 5.31)$. 

5.20 Change in the location of the optimal design for $\beta = (0, 1, 0, 1)^T$ over the design space $[-7.6, 6.3]$ as the number of observations at each support point increases. 

5.21 Variance and squared bias components of MSE at optimal design for $\beta = (0, 1, 0, 1)^T$ over the design space $\chi = [-7.6, 6.3]$ where the optimal design trend reverses as $n$ increases. 

5.22 IMSE-optimal design for $\beta = (1, 2, 3, 4)^T$ over the design space $\chi = [-2.9, 1.4]$ for $n = 2$, showing MSE of the individual probability for each category. 

5.23 IMSE-optimal design for $\beta = (1, 2, 3, 4)^T$ over the design space $\chi = [-2.9, 1.4]$ for $n = 2$, showing the average MSE. 

5.24 IMSE-optimal design for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$ for $n = 10$ and corresponding MSE curves. The left graph shows the MSE of individual probabilities and the right graph shows the average MSE. 

5.25 Design logic for IMSE-optimal design for three support points. 

5.26 IMSE-optimal designs for $\beta = (0, 1, 0, 1)^T$ over $\chi = [-7.6, 6.3]$ with different numbers of observations at each support point for $N = 6$. The left graph shows the design points and the right graph shows corresponding WIMSE values for each design.
5.27 Change in the location of the optimal design as the design space increases. The left graph is for $\boldsymbol{\beta} = (0, 1, 0, -1)^T$ and the right graph is for $\boldsymbol{\beta} = (0, 1, 0, 1)^T$.

6.1 Comparison of the MLE and MPLE estimators for each parameter value of $\hat{\boldsymbol{\beta}}$.

6.2 Predicted probability curves for two estimators of $\boldsymbol{\beta}$, over a range of $x$ values. The left graph uses the MLE estimate $\hat{\boldsymbol{\beta}}$ and the right graph uses the MPLE estimate $\tilde{\boldsymbol{\beta}}$.

6.3 Derivative function of the D-optimal design, for three design spaces. The left graph uses the MLE estimator of $\boldsymbol{\beta}$ and the right graph uses the MPLE estimator of $\boldsymbol{\beta}$.

6.4 MLE parameter $\hat{\boldsymbol{\beta}}$, for each modified $Y_{ij}$ scenario.

6.5 Probability curves using $\hat{\boldsymbol{\beta}}$ from original and ten $Y_{ij}$ scenarios. The top left graph is $\pi_1$, the top right graph is $\pi_2$ and the bottom graph is $\pi_3$.

6.6 D-optimal designs for the original experiment and each modified response scenario. The size of a support point is proportional to its weight. The top left graph is for $\chi_1 = [-5.7, -2.3]$, the top right graph is for $\chi_2 = [-5.99, -1.61]$ and the bottom graph is for $\chi_3 = [-6.21, -0.69]$.

6.7 Derivative function of the D-optimal designs, using $\hat{\boldsymbol{\beta}}$, for original $Y_{ij}$ (bold curve) and ten $Y_{ij}$ scenarios. The top left graph is for $\chi_1 = [-5.7, -2.3]$, the top right graph is for $\chi_2 = [-5.99, -1.61]$ and the bottom graph is for $\chi_3 = [-6.21, -0.69]$.

6.8 MLE parameter $\hat{\boldsymbol{\beta}}$, for randomly generated $\hat{\boldsymbol{\beta}}$ from a normal distribution.

6.9 Probability curves corresponding to the original and ten simulated $\hat{\boldsymbol{\beta}}$ over $\chi_1 = [-5.7, -2.3]$. The top left graph is for $\pi_1$, the top right graph is for $\pi_2$ and the bottom graph is for $\pi_3$.

6.10 D-optimal designs for the original experiment and each simulated $\hat{\boldsymbol{\beta}}$. The size of a support point is proportional to its weight. The top left graph is for $\chi_1 = [-5.7, -2.3]$, the top right graph is for $\chi_2 = [-5.99, -1.61]$ and the bottom graph is for $\chi_3 = [-6.21, -0.69]$.

6.11 Derivative function of the D-optimal designs, using $\hat{\boldsymbol{\beta}}$, for original $Y_{ij}$ (bold curve) and ten simulated $\hat{\boldsymbol{\beta}}$. The top left graph is for $\chi_1 = [-5.7, -2.3]$, the top right graph is for $\chi_2 = [-5.99, -1.61]$ and the bottom graph is for $\chi_3 = [-6.21, -0.69]$.

6.12 IMSE-optimal designs with different weights and WIMSE for $\hat{\boldsymbol{\beta}}$. The left graph is for the design space $\chi_1 = [-5.7, -2.3]$ and the right graph is for $\chi_3 = [-6.21, -0.69]$. 

xiii
6.13 IMSE-optimal designs with different weight and WIMSE for $\tilde{\beta}$.
The left graph is for the design space $\chi = [-5.7, -2.3]$ and the right graph is for $\chi = [-6.21, -0.69]$.

6.14 Derivative function of the D-optimal design for the quadratic predictor, and three design spaces. The left graph is for the MLE estimator, $\hat{\beta}$, and the right graph is for the MPLE estimator, $\tilde{\beta}$. 213
List of Tables

1.1 Common link functions for GLMs .............................................. 9

4.1 Starting values for constrOptim and resulting design solution, rounded to one decimal place. ................................. 68

4.2 D-optimal design for randomly generated $\beta$ from the uniform distribution where $b_1 \sim U (-2, 2)$, $b_2 \sim U (-1, 3)$, $b_3 \sim U (-2, 2)$ and $b_4 \sim U (0, 4)$. ................................................. 72

4.3 Relative efficiency values for randomly generated $\beta$. ................................................................. 75

4.4 Range designs and quartiles for the efficiencies of designs when $\beta$ is randomly generated from uniform distributions. ........................................... 77

4.5 Range designs and quartiles for the efficiencies of designs when $\beta$ is randomly generated from normal distributions. .................................................. 78

4.6 Optimal designs corresponding to the midpoints of the uniform distributions, given in Table 4.4, with quartiles of the efficiencies of these designs relative to the D-optimal designs obtained. .................. 80

4.7 Optimal designs corresponding to the midpoints of the normal distributions, given in Table 4.5, with quartiles of the efficiencies of these designs relative to the D-optimal designs obtained. .................. 82

4.8 Maximum values of the determinant of the information matrix for $\beta = (0, 1, 1, 0, 1, 1)^T$ across various simulations, each of size 100,000, using different seed values. ................................. 95

4.9 Various D-optimal designs for different $\beta$ parameter settings where the linear predictor has one covariate. ......................... 106

4.10 Various D-optimal designs for different $\beta$ parameter settings where the linear predictor has two covariate terms. ......................... 107

4.11 Various D-optimal designs for different $\beta$ parameter settings when the linear predictor has two covariates and an interaction term. ................................. 109

4.12 Models which reduce to one covariate linear predictor. ......................... 111

4.13 D-optimal designs for the models in Table 4.12. ................................. 112

4.14 Models which reduce to a two covariate linear predictor. ......................... 115

4.15 D-Optimal designs for the models listed in Table 4.14. Both models have design space $\chi = [-4, 3] \times [-4, 3]$. ................................. 116
5.1 Probability of infinite MLE for response variable combinations when \( n = 5 \) ................................................................. 127
5.2 Probability of infinite MLE ................................................................. 127
5.3 Design spaces for various \( \beta \) at four different precision levels \( \delta \) ...... 135
5.4 Integrated mean squared error of each estimated probability for \( \beta = (0, 1, 0, 1)^T \) over the design space \([-7.6, 6.3]\] with \( n = 5 \), calculated for two different sets of support points. .......... 139
5.5 A design grid for \( \beta = (0, 1, 0, 1)^T \) and \( n = 5 \) with WIMSE values calculated at selected grid locations over the design space \([-7.6, 6.3]\]. ......................................................... 141
5.6 Example of how minimax values are calculated. ................................. 145
5.7 Comparing optimal designs found using three methods for \( \beta = (0, 1, 0, 1)^T \) ................................................................. 148
5.8 Improvements in processing speed for constrOptim compared to grid based methods, using 70 intervals. ................................. 151
5.9 Relationship between design space and resulting optimal design for \( \beta = (0, 1, 0, 1)^T \) and \( n = 10 \) ................................................................. 152
5.10 Relationship between design space and resulting optimal design for \( \beta = (0, 1, 0, -1)^T \) and \( n = 10 \) ................................................................. 153
5.11 Comparison of variance and bias components of MSE for \( \beta = (0, 1, 0, 1)^T \) over the design space \( \chi = [-7.6, 6.3]\) ......................................................... 158
5.12 Comparison of variance and bias components of MSE for \( \beta = (0, 1, 0, -1)^T \) over the design space \( \chi = [-7, 7]\) ......................................................... 159
5.13 Total number of iterations, in expected value calculations, for a two and three point design for various values of \( n \) ................................................................. 170
5.14 Comparing IMSE-optimal and D-optimal designs with two equally weighted support points. ................................................................. 173

6.1 Original experimental dose, design points \( x_i = \ln(\text{dose}) \) and sample size \( n_i \) at \( i^{th} \) treatment level ................................................................. 180
6.2 Original experiment responses at the \( i^{th} \) treatment level ......................... 181
6.3 Comparing output values of two IMSL subroutines: DUMINF and DBCONF ................................................................. 183
6.4 Original experiment responses at each \( i^{th} \) treatment level from Table 6.2, multiplied by a factor of three ................................................................. 185
6.5 Treatment group sizes, and totals, when original sample sizes are multiplied by various factors ................................................................. 185
6.6 Original experiment treatments and additional doses. 187

6.7 D-optimal designs using the MLE, \( \hat{\beta} \), for three different design spaces. 189

6.8 D-optimal designs using the MPLE, \( \tilde{\beta} \), for three different design spaces. 189

6.9 Original and ten scenarios of altered treatment responses, \( Y_{ij} \), all maintaining treatment replications from original experiment. 194

6.10 \( \hat{\beta} \) obtained from varying experiment outcomes as listed in Table 6.9 195

6.11 Standard deviation of \( \hat{\beta} \) for each design space. 201

6.12 Relative efficiency of original experiment to D-optimal design, using \( \hat{\beta} \) and \( \tilde{\beta} \) parameter estimates, over three design spaces. 206

6.13 Selected combinations of weights for an IMSE-optimal design with two support points and a total of \( N = n_1 + n_2 = 66 \) observations. 208

6.14 IMSE-optimal two point designs for selected combinations observations at each design point, using \( \chi_1 = [-5.7, -2.3] \) and \( \hat{\beta} \). 208

6.15 D-optimal designs for quadratic predictor with ML estimator, \( \hat{\alpha} \), and three different design spaces. 212

6.16 D-optimal designs for quadratic predictor with MPLE estimator, \( \tilde{\alpha} \), and three different design spaces. 212
Chapter 1

Introduction

1.1 Experimental Design

Many textbooks deal with the design and analysis of experiments; for example Dean & Voss (1999) and Montgomery (1991). In each of these books, you will read that designing an experiment involves determining the values of the explanatory variables. You will also discover that the way in which data from an experiment will be analyzed needs to be taken into account when designing an experiment. Each of these texts explores an array of designs, including completely randomized designs, block designs, latin squares, factorial designs, hierarchical designs and response surface designs. These designs are usually referred to as classical designs.

The models behind these designs all have one thing in common: the assumption that the random errors associated with the response variable are independent and normally distributed with a constant variance. A problem arises when the data, produced by the experiment, violates these assumptions. An example is an experiment where the covariate is the dose of an anti-depression drug and the response is categorical and describes the subject’s mood (such as happy,
sad, angry or anxious). In this case, the model does not satisfy the assumptions for a classical linear design.

We intend to explore the design of experiments when the response is categorical, in particular from the multinomial distribution. We model the probability of an observation falling in a specified multinomial category as a function of one or more covariates \( x \) and parameters \( \beta \). The design objective is to determine the optimal values of \( x \), that is, where to take observations in order to maximize the quality of data obtained by the experiment.

An optimal design with \( s \) support points will be denoted by

\[
\xi^* = \begin{pmatrix} x^T \\ \omega^T \end{pmatrix}
\]

where \( x^T = (x_1, x_2, \ldots, x_s) \) and \( x_i \) represents the value of the covariate at the \( i^{th} \) support point. Also \( \omega^T = (\omega_1, \omega_2, \ldots, \omega_s) \), and \( \omega_i \) represents the weight at the \( i^{th} \) point with \( 0 \leq \omega_i \leq 1 \) and \( \sum \omega_i = 1 \) for all \( i \). So, given the total number of observations, the weights determine the proportion of observations taken at each design point. When the \( s \) support points in the design are equally weighted, the optimal design is denoted by \( \xi^* = (x^T) \), which implies a weight of \( 1/s \) at each design point.

The support points in a design can be thought of as the specification of the levels of the treatments that provide predictions of the response. For example, in a clinical trial for a particular drug, the support points are the doses of the drug.
1.2 Linear Models

Many statistical techniques are based on linear models, such as the analysis of variance and regression. There are many references available on this subject, in particular Draper & Smith (1998) is considered a classic text. These methods are appropriate when the errors are independent and normally distributed with a constant variance. Part of the appeal of linear models are the mathematical properties they possess. For example, the least squares estimator of the parameters in a linear regression model is easily obtainable, which is not necessarily the case when the model is non linear.

The linear model is commonly written as

\[ Y = X\beta + \epsilon \]

where \( Y \) is a vector of \( n \) observations \( Y = (y_1, y_2, \ldots, y_n)^T \) and \( X \) is the \((n \times p)\) design matrix. Each row of \( X \) refers to a different observation, and each column refers to a different covariate. Associated with the \( i^{th} \) covariate is, \( \beta_i \), the \( i^{th} \) element of a vector of \( p \) parameters \( \beta = (\beta_1, \beta_2, \ldots, \beta_p)^T \). The vector \( \epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T \) contains \( n \) errors which are independent and identically distributed random variables with \( \epsilon_i \sim N(0, \sigma^2) \) for \( i = 1, 2, \ldots, n \).

Dobson (2002, page 45) gives some reasons why the normal distribution is commonly used to model continuous data that may not necessarily be normal. One reason is the Central Limit Theorem which says that, even if the data are not normal, the average of a random sample will be approximately normally distributed in a reasonable sized sample. The other reason, alluded to above, is the prevalence of statistical theory developed for normally distributed data. Thus in many cases, transforming the data to achieve (approximate) normality will enable the application of established methods of analysis.
A risk associated with linear models is their misuse, in particular their application when the underlying assumptions are not satisfied. In the context of experimental design, Woods et al. (2006) says “The most common type of designed experiment is where the observations are described adequately by a linear model. Experiments are sometimes performed based on such designs when this assumption is not justified.”

In designs for linear models, the assumptions of normal errors with constant variance comes into the criteria for optimum design through the form of the covariance matrix of $Y$ which is

$$\text{Cov}(Y) = \sigma^2 I_n,$$

where $I_n$ is the $n \times n$ identity matrix. Finding expressions for estimators, such as least squares or maximum likelihood estimators, is fairly straightforward since $\text{Cov}(Y)$ is not a function of the $\beta$ parameters. This has facilitated the extensive amount of literature in the design of experiments for linear models.

In the preface to his book, Silvey (1980, page viii) refers to the linear design theory and says it “... is now firmly established and there are few outstanding problems connected with it. The same cannot be said of nonlinear theory.” Even though this comment was made almost 30 years ago, it is still relevant today, even in the context of generalized linear models, a subclass of nonlinear models.

### 1.3 Generalized Linear Models

Generalized linear models (GLMs) provide the framework that encompasses various statistical models, including models for continuous response variables,
such as ordinary regression and ANOVA, and models for categorical response variables, such as logistic regression. Linear models are a special case of GLMs. GLMs allow methods similar to those developed for linear models to be applied to non-normal data, for example a general algorithm was developed for maximum likelihood estimation in GLM models; see Dobson (2002, page 62).

McCullagh & Nelder (1989, page 12) give a detailed account of how GLMs developed over time. The first example they describe is Fisher (1922), a dilution assay experiment with a Poisson response. Fisher used what is now known as a complementary log log transformation and applied maximum likelihood estimation to this problem. Other early examples, which have non-normal responses in common, are also detailed in McCullagh & Nelder (1989), and the different transformations used to achieve linearity are described by the authors.

The GLM theory that unifies these different models and allows them to be studied as a single class is attributable to Nelder & Wedderburn (1972). Underlying this theory is the assumption that the response has a probability distribution belonging to the exponential family. The mean of the response is a function of the unknown parameters and also a function of the explanatory variables through a function of the linear predictor. This link function is usually nonlinear. The variance of the response is a function of the mean. An established reference book on GLMs is McCullagh & Nelder (1989) and more recently Dobson (2002) and Myers, Montgomery & Vining (2002).

In design for GLMs, the asymptotic covariance matrix of $Y$ is a function of the unknown $\beta$ parameters and $X$ covariates

$$
\text{Cov}(Y) = f(X, \beta).
$$

(1.2)

Thus finding expressions for estimators is not straightforward since $\text{Cov}(Y)$ depends on parameters, $\beta$, which are unknown. This has hindered theoretical
work in the design of experiments for nonlinear models.

To describe how GLMs fit together, consider the three components of a GLM: the random, the systematic and the link function.

### 1.3.1 Random Component

The random component of a GLM defines the probability distribution of the response. We specify that \( n \) independent observations \( Y_1, Y_2, \ldots, Y_n \) on the response \( Y \) have a probability distribution that depends on \( \theta \). The probability density function belongs to the exponential family, which is written in two common forms.

One form of the exponential family is given by Dobson (2002, page 44)

\[
f(y; \theta) = \exp \left[ a(y) b(\theta) + c(\theta) + d(y) \right], \tag{1.3}
\]

where \( a(\cdot), b(\cdot), c(\cdot) \) and \( d(\cdot) \) are known functions. If \( a(y) = y \), the distribution is in canonical form and \( b(\theta) \) is called the natural parameter of the distribution. If the distribution has other parameters, in addition to \( \theta \), they are treated as nuisance parameters, whose values are assumed known.

The other common form of the exponential family is given by McCullagh & Nelder (1989, page 28)

\[
f(y; \theta, \phi) = \exp \left[ \frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right] \tag{1.4}
\]

where \( \theta \) depends on the values of the explanatory variables, \( \phi \) is the dispersion parameter, and the functions \( a(\cdot), b(\cdot) \) and \( c(\cdot, \cdot) \) describe the specific distribution. If \( \phi \) is known, this is an exponential family model with canoni-
cal parameter $\theta$. This is the model referred to in this thesis unless specifically stated otherwise.

Standard theory, McCullagh & Nelder (1989, page 29), gives the expected value $E(Y) = b'(\theta)$ and variance $\text{Var}(Y) = b''(\theta) a(\phi)$. Probability distributions of the response $Y$ in GLMs are usually parametrized in terms of the mean $\mu$ and the dispersion parameter $\phi$ rather than the natural parameter $\theta$.

Many well known distributions belong to the exponential family. For example, some common univariate distributions include: the normal, binomial, Poisson, gamma and inverse Gaussian.

For example, consider the normal distribution with probability distribution given by

$$f(y; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ \frac{1}{2} \left( \frac{y - \mu}{\sigma} \right)^2 \right\}$$

which can be written in exponential form (1.4) as

$$f(y; \mu, \sigma) = \exp \left\{ \frac{y\mu}{\sigma^2} - \frac{\mu^2}{2\sigma^2} - \frac{1}{2} \left[ \frac{y^2}{\sigma^2} + \ln (2\pi\sigma^2) \right] \right\},$$

so that $\theta = \mu$, $\phi = \sigma^2$ and $a(\phi) = \sigma^2 = \phi$, $b(\theta) = \mu^2/2 = \theta^2/2$ and $c(y_i, \phi) = -\frac{1}{2} \left[ y^2/\sigma^2 + \ln (2\pi\sigma^2) \right]$.

The exponential family of distributions also include multivariate distributions, see Agresti (2002, page 310). If $Y$ is a multivariate response vector, the random component of a GLM specifies that $n$ independent observations on the response vector $Y = (Y_1, Y_2, \ldots, Y_k)^T$ have a probability density function which belongs to the exponential family if it can be written as

$$f(y; \theta, \phi) = \exp \left[ \frac{y^T \theta - b(\theta)}{a(\phi)} + c(y, \phi) \right]$$

(1.5)
where \( a(\cdot), b(\cdot) \) and \( c(\cdot, \cdot) \) are known functions, \( \theta \) is a canonical parameter and \( \phi \) is a dispersion parameter.

### 1.3.2 Systematic Component

The systematic component of a GLM specifies the explanatory variables. We define \( \eta \) as a linear combination of \( p \) explanatory variables \( x_j, j = 1, 2, \ldots, p \). The predictor is expressed as a linear combination of unknown parameters \( \beta_j \).

For the \( i^{th} \) observation

\[
\eta_i = \sum_{j=1}^{p} x_{ij} \beta_j, \quad i = 1, 2, \ldots, n
\]

where \( x_{ij} \) is the value of the \( j^{th} \) explanatory variable at the \( i^{th} \) observation.

In matrix notation, we write

\[
\eta = X\beta
\]

where \( \eta \) is a vector of \( n \) linear predictors \( \eta = (\eta_1, \eta_2, \ldots, \eta_n)^T \), \( \beta \) is a vector of \( p \) parameters \( \beta = (\beta_1, \beta_2, \ldots, \beta_p)^T \) and \( X \) is the \((n \times p)\) model matrix written as

\[
X = \left[
\begin{array}{c}
x_1^T \\
\vdots \\
x_n^T
\end{array}
\right] = \left[
\begin{array}{c}
x_{11} \cdots x_{1p} \\
\vdots \\
x_{n1} \cdots x_{np}
\end{array}
\right]
\]

Each row of the model matrix \( X \) refers to a different observation, and each column refers to a different covariate.
1.3.3 Link Function

The link function $\eta$ is so called because it relates, or links, the systematic and random components of a GLM. It specifies how the expected value of the response relates to the explanatory variables. Let $\mu_i = \operatorname{E}(Y_i)$, then

$$
\eta_i = g(\mu_i) \quad \text{and} \\
g(\mu_i) = \sum_{j=1}^{p} x_{ij}\beta_j, \quad i = 1, 2, \ldots, n.
$$

In matrix notation

$$
\eta = g(\mu) = X\beta.
$$

The link function is usually chosen based on the form of the distribution of the response. The choice of a link function is similar to the choice of a transformation of the response, except that the link function is a transformation of the mean, not of the individual observations. Table 1.1 lists some common link functions.

<table>
<thead>
<tr>
<th>Link Function</th>
<th>$g(\mu)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity</td>
<td>$\mu$</td>
</tr>
<tr>
<td>Logarithmic</td>
<td>$\log(\mu)$</td>
</tr>
<tr>
<td>Logit</td>
<td>$\log[\mu/(1-\mu)]$</td>
</tr>
<tr>
<td>Probit</td>
<td>$\Phi^{-1}(\mu)$ where $\Phi(\cdot)$ is the normal cumulative distribution function</td>
</tr>
<tr>
<td>Complementary log-log</td>
<td>$\log[-\log(1-\mu)]$</td>
</tr>
<tr>
<td>Power</td>
<td>$\begin{cases} \mu^\lambda &amp; \lambda \neq 0 \ \log \mu &amp; \lambda = 0 \end{cases}$</td>
</tr>
</tbody>
</table>

Table 1.1: Common link functions for GLMs.

The canonical link occurs when $\theta = \eta$, where $\theta$ is the canonical parameter defined in (1.4). For example, when the response is normal, the canonical link
is the identity function, which does not restrict the values \( \eta \) and \( \mu \) are allowed to take. When the response is Poisson, the canonical link is the logarithmic function, which is chosen to ensure \( \mu > 0 \). This is shown in Section 1.4. When the response is binomial, the canonical link is the logit function, which ensures \( 0 < \mu < 1 \). McCullagh & Nelder (1989, page 32) state that the canonical link is favored as it results in mathematically appealing properties of the model.

The logit is the most popular link to use for a GLM with a multinomial response. Two other widely used links are the probit and complementary log-log, as given in Table 1.1.

1.4 Poisson and Binomial Distributions

The Poisson and binomial are closely related distributions because they both model count data and the binomial is the joint distribution of two independent Poisson random variables, conditional on their sum being fixed. Both distributions have a strong relationship between their respective mean and variance. Even after transformations, normal linear models cannot be used as these distributions violate the assumption of constant variance.

The Poisson distribution is used for the analysis of count data. It is determined by one parameter, its mean, \( \lambda \), and takes values 0, 1, 2, \ldots without an upper limit. McCullagh & Nelder (1989, page 15) made the comment that “The Poisson distribution is the nominal distribution for counted data in much the same way that the normal distribution is the bench-mark for continuous data.”

The probability function for a random variable \( Y \) with a Poisson distribution is given by \( f(y) = e^{-\lambda} \lambda^y / y! \), \( y = 0, 1, \ldots \), and can be written in exponential form.
as

\[ f(y) = \exp \{ y \ln \lambda - \lambda - \ln y! \} . \]

Equating this to the general form of the exponential family, (1.4), we can see that \( \theta = \ln \lambda \), which is the logarithmic link and the canonical link for the Poisson, \( a(\phi) = 1 \), \( b(\theta) = \lambda = e^\theta \) and \( c(y; \phi) = -\ln y! \). The expected value of a Poisson random variable is equal to its variance, \( E(Y) = \lambda = \text{Var}(Y) \), hence there is a non-constant variance.

The binomial distribution is also used when the response is count data. It can arise in two ways. The most common is as the sum of independent Bernoulli trials, where each trial can only have two possible outcomes, say 0 (failure) or 1 (success). The probability is the same for each trial and is defined as \( P(X = 1) = \pi \) or \( P(X = 0) = 1 - \pi \). Then, the number of successes, \( Y \), from the \( n \) outcomes has a binomial distribution with probability function \( f(y) = \binom{n}{y} \pi^y (1 - \pi)^{n-y} \), \( y = 0, 1, \ldots, n \). The expected value of a binomial random variable is \( E(Y) = n\pi \) and its variance is \( \text{Var}(Y) = n\pi(1 - \pi) \), again a non-constant variance.

The binomial can also be obtained as the joint distribution of two independent Poisson random variables, conditional upon their sum \( n \). If \( Y_1, Y_2 \) are independent Poisson random variables with means \( \lambda_1, \lambda_2 \) then it follows that \( Y_1 + Y_2 \) also has a Poisson distribution with mean \( \lambda_1 + \lambda_2 \), and

\[ f(y_1, y_2 \mid y_1 + y_2 = n) = \frac{e^{-\lambda_1} \lambda_1^{y_1}}{y_1!} \times \frac{e^{-\lambda_2} \lambda_2^{y_2}}{y_2!} / \frac{e^{-(\lambda_1 + \lambda_2)} (\lambda_1 + \lambda_2)^{y_1+y_2}}{n!} \]

\[ = \binom{n}{y_1} \pi^{y_1} (1 - \pi)^{n-y_1} \]

which is the binomial distribution, since \( y_1 + y_2 = n \) and \( \pi = \lambda_1 / (\lambda_1 + \lambda_2) \).
The binomial distribution in exponential form is written as

\[ f(y) = \exp \left\{ y \ln \left( \frac{\pi}{1 - \pi} \right) + n \ln (1 - \pi) + \ln \binom{n}{y} \right\}, \]

so that equating to (1.4) gives \( \theta = \ln \left( \frac{\pi}{1 - \pi} \right) \), which is the logit link and canonical link for the binomial, \( a(\phi) = 1 \), \( b(\theta) = -n \ln (1 - \pi) = n \ln (1 + e^{\theta}) \) and \( c(y, \phi) = \ln \binom{n}{y} \).

### 1.5 Multinomial Distribution

The multinomial distribution arises when the response variable has an outcome restricted to one of a fixed set of possible categories. This type of response is called polytomous or multicategory. The categories can be classified as either nominal or ordinal. The difference is that ordinal variables have a natural order, such as headache rating (extreme, severe, moderate, none), whereas nominal categories do not, such as political parties (Labor, Liberal, National, Green).

An example of an application of the multinomial distribution is tossing a die, which has six possible outcomes. The binomial distribution is a special form of the multinomial distribution when there are only two outcomes.

The random variables \((Y_1, Y_2, \ldots, Y_k)\) jointly have a multinomial distribution \((Y_1, Y_2, \ldots, Y_k) \sim \text{Multinomial}(n; \pi_1, \pi_2, \ldots, \pi_k)\) if, in a series of \(n\) independent trials, each trial results in exactly one of a fixed number, \(k\), of possible outcomes with probabilities \(\pi_1, \pi_2, \ldots, \pi_k\), where \(\pi_i > 0\) for all \(i\).

The probability function for the multinomial distribution with \(k\) categories is

\[ f(y_1, \ldots, y_k; \pi_1, \ldots, \pi_k) = \frac{n!}{y_1! y_2! \ldots y_k!} \pi_1^{y_1} \pi_2^{y_2} \cdots \pi_k^{y_k}. \]
The probabilities across all categories sum to one
\[ \sum_{j=1}^{k} \pi_j = 1, \quad (1.7) \]
and the total number of observations in all categories sum to \( n \)
\[ \sum_{j=1}^{k} y_j = n. \quad (1.8) \]

A vector of responses \( \mathbf{Y} = (Y_1, Y_1, \ldots, Y_k) \) is distributed as
\( \mathbf{Y} \sim \text{Multinomial}(n; \pi_1, \ldots, \pi_k) \) and has mean \( \mathbf{E}(Y_j) = n\pi_j \). The covariance matrix of \( \mathbf{Y} \) has main diagonal entries \( \text{Var}(Y_j) = n\pi_j(1 - \pi_j) \) and off-diagonal entries \( \text{Cov}(Y_i, Y_j) = -n\pi_i\pi_j \). The covariances are all negative because an increase in one response value means another must decrease, for fixed \( n \).

The random variable \( \mathbf{Y} = (Y_1, Y_2, Y_3) \) has a three-category multinomial distribution, also called trinomial, \( \mathbf{Y} \sim \text{Multinomial}(n; \pi) \), with probability function
\[
f(y; \pi) = \frac{n!}{y_1!y_2!y_3!} \pi_1^{y_1} \pi_2^{y_2} \pi_3^{y_3}
\quad (1.9)
\]
where the probabilities sum to one, \( \pi_1 + \pi_2 + \pi_3 = 1 \), and the numbers of observations in each category sum to \( n \), \( y_1 + y_2 + y_3 = n \).

The multinomial can also be defined as the joint distribution of \( k \) independent Poisson random variables, conditional upon their sum. This is demonstrated in many texts, including Dobson (2002, page 136) and Aitkin et al. (1989, page 231).

The multinomial distribution, although related to the binomial and Poisson distributions, differs in an important aspect. The multinomial is multivariate in the response and the observations are no longer independent from each other. If the random variable \( \mathbf{Y} = (Y_1, Y_2, \ldots, Y_k) \) has a \( k \)-category multinomial distribution,
then $\sum_{i=1}^{k} y_i = n$. If we know $y_1, y_2, \ldots, y_{k-1}$ then $y_k = n - y_1 - y_2 - \ldots - y_{k-1}$ and the observations are therefore correlated.

### 1.6 Multinomial GLM Components

Consider a random variable $\mathbf{Y}$ with $k$ response categories. We make $n_i$ independent observations of $\mathbf{y}_i$ at the $i^{th}$ support point, $i = 1, 2, \ldots, s$, which result in $y_{ij}$ outcomes in each of the $j = 1, 2, \ldots, k$ categories. A support point refers to the treatments in the experiment and is also called a design point.

We can write the vector of outcomes in each category as $\mathbf{y}_i = (y_{i1}, y_{i2}, \ldots, y_{ik})$ where $\sum_{j=1}^{k} y_{ij} = n_i$. The $y_{ij}$ values are independent for different $i$, but not for the same $i$. Let $\mathbf{\pi}_i = (\pi_{i1}, \pi_{i2}, \ldots, \pi_{ik})$ denote the vector of probabilities for each category where $\sum_{j=1}^{k} \pi_{ij} = 1$. The random variable $\mathbf{Y}$ has a multinomial distribution, which is a member of the exponential family.

The random component of the GLM, is the multinomial distribution. The function of the response is given by

$$f(\mathbf{y}_i, \mathbf{\pi}_i) = \exp \left\{ n_i \ln \pi_{i1} + \sum_{j=2}^{k} \left( y_{ij} \ln \left( \frac{\pi_{ij}}{\pi_{i1}} \right) \right) + \ln \left( \frac{n_i^{!}}{\prod_{i=1}^{k} y_{ik}^{!}} \right) \right\}$$

where the first category has been chosen as the reference category. This denotes the category to which the other response categories are compared.

The systematic component of a GLM can be written as

$$\eta = \mathbf{X}\boldsymbol{\beta},$$

where $\mathbf{X}$ is the design matrix. Its elements include the values of the covariates at each of the support points. The elements of $\boldsymbol{\beta}$ are the unknown parameters.
The dimensions of $\eta$, $\beta$ and $X$ depend on the number of categories in the multinomial distribution and the model parametrization. An example of the linear predictor for the multinomial distribution with three categories is given below.

The link function, which relates the random and systematic components, is given by

$$
\eta_{ij} = \ln \frac{\pi_{ij}}{\pi_{i1}} \quad \text{for } j = 2, 3, \ldots, k
$$

at the $i^{th}$ support point ($i = 1, 2, \ldots, s$). If we make $\pi_{ij}$ the subject, then the probabilities expressed in terms of the linear predictor are

$$
\pi_{ij} = \frac{e^{\eta_{ij}}}{1 + \sum_{j=2}^{k} e^{\eta_{ij}}} \quad \text{for } j = 2, 3, \ldots, k,
$$

where $\pi_{i1} = 1 - \sum_{j=2}^{k} \pi_{ik}$. To emphasize that the probability $\pi$ is a function of $x$, we can also write $\pi (x)$.

The reason the logit link uses a baseline category relates to a probability being associated with each multinomial category. If we have $k$ categories, we only need to know $k - 1$ probabilities as the probabilities sum to one. It follows that, for a $k$ category multinomial, we only require $k - 1$ linear predictor equations.

In the case of a multinomial distribution with $k = 3$ response categories, the random GLM component is the probability function written in exponential form

$$
f (y_i, \pi_i) = \exp \left( y_{i2} \ln \frac{\pi_{i2}}{\pi_{i1}} + y_{i3} \ln \frac{\pi_{i3}}{\pi_{i1}} + n_i \ln \pi_{i1} + \ln \frac{n_i!}{y_{i1}!y_{i2}!y_{i3}!} \right).
$$
The linear predictor of the explanatory variables, with one covariate, $x$, is

\[
\begin{align*}
\eta_{i2} &= \beta_1 + \beta_2 x_i \\
\eta_{i3} &= \beta_3 + \beta_4 x_i
\end{align*}
\]  

(1.10)

where the probabilities are a function of the parameters of the model and covariates, given by

\[
\begin{align*}
\pi_{i2} &= \frac{\exp(\beta_1 + \beta_2 x_i)}{1 + \exp(\beta_1 + \beta_2 x_i) + \exp(\beta_3 + \beta_4 x_i)} \\
\pi_{i3} &= \frac{\exp(\beta_3 + \beta_4 x_i)}{1 + \exp(\beta_1 + \beta_2 x_i) + \exp(\beta_3 + \beta_4 x_i)},
\end{align*}
\]  

(1.11)

and hence

\[
\begin{align*}
\pi_{i2} &= \frac{e^{\eta_{i2}}}{1 + e^{\eta_{i2}} + e^{\eta_{i3}}} \\
\pi_{i3} &= \frac{e^{\eta_{i3}}}{1 + e^{\eta_{i2}} + e^{\eta_{i3}}},
\end{align*}
\]

where $\pi_{i1} = 1 - \pi_{i2} - \pi_{i3}$.

In vector form, $\eta = X\beta$, with two support points then $\eta = (\eta_{12}, \eta_{13}, \eta_{22}, \eta_{23})^T$ and $\beta = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ and

\[
X = \begin{bmatrix}
1 & x_1 & 0 & 0 \\
0 & 0 & 1 & x_1 \\
1 & x_2 & 0 & 0 \\
0 & 0 & 1 & x_2
\end{bmatrix}.
\]

The link function connects the systematic and random components. We use the logit link, and need to designate a category as baseline. This denotes the category to which each other response category is referred.
The link function is given by

\[ \eta_{i2} = \ln \frac{\pi_{i2}}{\pi_{i1}} \]
\[ \eta_{i3} = \ln \frac{\pi_{i3}}{\pi_{i1}} \]  

(1.12)

for \( i = 1, 2, \ldots, s \). If we make \( \pi_{ij} \) the subject then the probabilities expressed in terms of the linear predictor are

\[ \pi_{i2} = \frac{e^{\eta_{i2}}}{1 + e^{\eta_{i2}} + e^{\eta_{i3}}} \]
\[ \pi_{i3} = \frac{e^{\eta_{i3}}}{1 + e^{\eta_{i2}} + e^{\eta_{i3}}} \]  

(1.13)

and \( \pi_{i1} = 1 - \pi_{i2} - \pi_{i3} \).

We shall be dealing with multinomial response categories and the baseline logit link. However, when the multinomial response categories have a natural order, it is preferable to use a model that utilizes this information, such as cumulative logits or probits and proportional odds models. There are also models for ordinal categories that do not use cumulative probabilities, such as the adjacent-categories logit model and the continuation-ratio logit model. See Agresti (2002, page 267) for more details on these models.

### 1.7 Discrete Choice Models

An important application of multinomial response models, that is not considered in this thesis but should be mentioned, is called discrete choice modeling, which is used to model the behaviour of consumer choice.

Discrete choice models relate the choice made by a subject to the characteristics of the alternatives available and to the attributes of the subject. For example,
the choice of which mode of transport to take to work (bus, car, bicycle, walk, train or other) is related to the characteristics of the alternatives (transport cost, trip duration, parking cost and availability) and the person’s attributes (age, income, distance from work). Discrete choice models estimate the probability that a subject chooses a particular alternative.

Discrete choice models can take many forms including binary and multinomial logit and probit models, models that assume correlation in the factors, nested models, generalized extreme value models and models that take orderings into account.

McFadden (1974) introduced the conditional logit model and has been recognized (joint Nobel Prize 2000) for his contribution to the development of theoretical work in this area. See Agresti (2002, page 298) for more details.

The main difference between the multinomial logit model and discrete choice models, discussed by Allison (1991, page 161), is that choice sets can include the choice options as well as options describing the association between the subject and the option. For example, in our transportation example, the cost of bus transport is related to where the person lives and works. Another difference is that the choice sets can vary among subjects. For example, not all subject will have train as an available mode of transport.

Even though the models are not the same, the multinomial baseline category logit model is a special case of the conditional logit model.

1.8 Research Outline

Chapter 2 establishes some of the theoretical groundwork that will be assumed in other chapters. We discuss maximum likelihood estimation for the multinomial distribution and demonstrate some cases where it is not possible to obtain
explicit solutions. Estimation becomes more difficult for distributions like the multinomial with correlated responses, which results in increased difficulty in design and estimation compared to distributions with univariate responses.

Chapter 3 explores the relevant literature and evaluates some common methods proposed for designing experiments for GLMs.

Chapter 4 develops locally optimal designs based on minimizing the determinant of the covariance matrix of the parameter estimates for a specified model and utilizes the General Equivalence theorem, which is fundamental to D-optimal designs. We compare various optimization methods and find designs for two equally weighted support points. We also develop design methods which allow for some uncertainty in the parameters and develop estimates based on their location and spread.

The constraints on the number of support points, and their weights, are relaxed and designs are developed for various types of linear predictors. We try to determine some design rules for particular cases but find that the assumed values of the parameters are the most dominant factor in determining the optimal design characteristics.

Chapter 5 develops an alternative optimal design criterion, which is based on minimizing the integrated mean squared error of the estimated probabilities, termed IMSE-optimal. The design space in the experiment is shown to play a crucial role in determining the optimal design and a method is proposed to calculate the design space when no constraints are imposed on the experimental treatments. IMSE-optimal designs are based on different criteria to D-optimality and we show how there is less flexibility and more effort required in implementing these designs.

Chapter 6 applies the techniques developed in previous chapters to a dose response experiment. The results from this experiment are used to calculate
parameter estimates and corresponding designs. We then compare these to the
designs that would have been obtained for similar parameter estimates, or if the
responses in the original experiment were slightly modified.
Chapter 2

Estimation

Maximum likelihood is a standard method for estimating the parameters of a generalized linear model. The likelihood function is the probability of observing the data as a function of the unknown parameters. There are two steps to obtain the maximum likelihood estimators (MLEs); first construct the likelihood and secondly maximize it. We seek values of the unknown parameters that make the likelihood as large as possible.

The likelihood is derived for the special case of a multinomial response with three categories. When there are two design points, we derive explicit mathematical expressions for the ML estimators. However, for more design points, explicit solutions cannot be found and, typically, numerical methods are required.

Fisher’s information matrix tells us how much information about the parameters is contained in the data. The inverse of the expected information matrix is the asymptotic covariance matrix of the model parameters. It is important in the field of experimental design because many design optimization techniques are based on maximizing aspects of the information matrix. The derivative of the log-likelihood is also called the score statistic and its variance is the Fisher information matrix.
Finally, we explain that estimation becomes more difficult for distributions like the multinomial with correlated responses. This results in increased difficulty in design and estimation compared to distributions with univariate responses.

2.1 Maximum Likelihood Estimation

Maximum likelihood estimation, originally developed by R. A. Fisher, is a standard approach to parameter estimation and inference. The basic principle behind ML is to choose the parameter values as estimators which would maximize the probability, or likelihood function.

The theory of maximum likelihood underlies many inference methods in statistics, such as, the chi-square test, and forms a component of Bayesian methods. ML is widely used due to its many optimal properties in estimation.

MLEs have good properties in large samples, Allison (1991, page 16). They are asymptotically efficient and consistent and their sampling distributions will be approximately normal in large samples, under fairly general conditions. A consequence of asymptotic consistency, means the estimator converges to the true value of the parameter value as the sample size gets larger, and the MLE is approximately unbiased in large samples. Also, MLEs are sufficient, defined by Fisher (1922) as "... when no other statistic which can be calculated from the same sample provides any additional information as to the value of the parameter".

The desirable properties of MLEs have only been proved for large samples. For small samples, the bias of MLEs can be considerable. We address this in Section 5.3 by referring to Firth (1993) who developed a penalized estimator to reduce the bias of the MLE.
2.2 Likelihood Function

Consider the random variable $Y$ which has a multinomial distribution with three categories, $Y \sim \text{Multinomial} (n; \pi)$, and probability density function given by (1.9) and repeated here for convenience,

$$f(y; \pi) = \frac{n!}{y_1!y_2!y_3!} \pi_1^{y_1} \pi_2^{y_2} \pi_3^{y_3},$$

subject to the constraints given by (1.7) and (1.8), where the probabilities are a function of the parameters of the model and covariates, given by (1.11).

The likelihood function is the probability of the observed data, expressed as a function of the unknown parameters, given by

$$L(\pi | y) = \prod_{i=1}^{s} f(y_i; \pi_i)$$

$$= \prod_{i=1}^{s} \frac{n_i!}{y_{i1}!y_{i2}!y_{i3}!} \pi_{i1}^{y_{i1}} \pi_{i2}^{y_{i2}} \pi_{i3}^{y_{i3}}$$

where $s$ is the number of support points.

It is generally easier to work with the logarithm of the likelihood because exponents become coefficients and products are converted to sums. The log-likelihood function is given by

$$\ell(\pi | y) = \sum_{i=1}^{s} \left\{ y_{i1}\ln \pi_{i1} + y_{i2}\ln \pi_{i2} + y_{i3}\ln \pi_{i3} + \ln (n_i!) - \ln (y_{i1}!y_{i2}!y_{i3}!) \right\}$$

(2.1)

where the probabilities are functions of the parameters, $\beta$, and the observations, $x_i$, given by (1.11).
2.3 Score Function

To find the MLE of $\beta$, a vector of $p$ parameters, we need to maximize the log-likelihood function, (2.1). To do this, we find the partial derivative, with respect to $\beta$, which gives the score function, in our case, $p = 4$.

$$ U(\beta) = \frac{\partial \ell}{\partial \beta} = \left( \frac{\partial \ell}{\partial \beta_1}, \frac{\partial \ell}{\partial \beta_2}, \ldots, \frac{\partial \ell}{\partial \beta_p} \right)^T. \tag{2.2} $$

Consider the first component of (2.2) for the three category multinomial distribution. Differentiating the log likelihood (2.1) with respect to $\beta_1$ gives

$$ U(\beta_1) = \sum_{i=1}^{s} \left( \frac{y_{i1} \partial \pi_{i1}}{\pi_{i1} \partial \beta_1} + \frac{y_{i2} \partial \pi_{i2}}{\pi_{i2} \partial \beta_1} + \frac{y_{i3} \partial \pi_{i3}}{\pi_{i3} \partial \beta_1} \right) $$

$$ = \sum_{i=1}^{s} \left( \frac{y_{i1}}{\pi_{i1}} (-\pi_{i1} \pi_{i2}) + \frac{y_{i2}}{\pi_{i2}} (\pi_{i2} - \pi_{i2}^2) + \frac{y_{i3}}{\pi_{i3}} (-\pi_{i2} \pi_{i3}) \right) $$

$$ = \sum_{i=1}^{s} (y_{i2} - n_i \pi_{i2}). $$

The remaining derivatives, with respect to each $\beta_i$, can be found in a similar manner, to give the scores

$$ U(\beta_1) = \sum_{i=1}^{s} (y_{i2} - n_i \pi_{i2}) $$

$$ U(\beta_2) = \sum_{i=1}^{s} x_i (y_{i2} - n_i \pi_{i2}) $$

$$ U(\beta_3) = \sum_{i=1}^{s} (y_{i3} - n_i \pi_{i3}) $$

$$ U(\beta_4) = \sum_{i=1}^{s} x_i (y_{i3} - n_i \pi_{i3}). \tag{2.3} $$

Note that the score is a function of $\beta$ as the probabilities are given by (1.11).
The ML estimators $\hat{\pi}$ and $\hat{\beta}$ are obtained by maximizing (2.1), which is equivalent to solving $U(\beta) = 0$, given as

\[
\sum_{i=1}^{s} (y_{i2} - n_i \hat{\pi}_{i2}) = 0 \tag{2.4}
\]
\[
\sum_{i=1}^{s} x_i (y_{i2} - n_i \hat{\pi}_{i2}) = 0
\]
\[
\sum_{i=1}^{s} (y_{i3} - n_i \hat{\pi}_{i3}) = 0
\]
\[
\sum_{i=1}^{s} x_i (y_{i3} - n_i \hat{\pi}_{i3}) = 0.
\]

### 2.4 MLE for Multinomial Distribution

To find the MLE $\hat{\beta}$ we first find the MLE $\hat{\pi}$. In our example of a multinomial response with three categories, consider a design with two support points. Using (2.4) for $s = 2$ gives us four equations and four values of $y_{ij}$, where

\[
y_{12} - n_1 \hat{\pi}_{12} + y_{22} - n_2 \hat{\pi}_{22} = 0
\]
\[
x_1 y_{12} - n_1 \hat{\pi}_{12} + x_2 y_{22} - n_2 \hat{\pi}_{22} = 0
\]
\[
y_{13} - n_1 \hat{\pi}_{13} + y_{23} - n_2 \hat{\pi}_{23} = 0
\]
\[
x_1 y_{13} - n_1 \hat{\pi}_{13} + x_2 y_{23} - n_2 \hat{\pi}_{23} = 0.
\]

We can solve these equations to find explicit solutions for $\hat{\pi}$, given as

\[
\hat{\pi}_{i2} = \frac{y_{i2}}{n_i}
\]
\[
\hat{\pi}_{i3} = \frac{y_{i3}}{n_i}, \quad i = 1, 2. \tag{2.5}
\]
From (1.13) we can write the MLE for $\pi_{ij}$ in terms of the linear predictor $\hat{\eta}_{ij}$

\[
\hat{\pi}_{i2} = \frac{e^{\hat{\eta}_{i2}}}{1 + e^{\hat{\eta}_{i2}} + e^{\hat{\eta}_{i3}}}
\]
\[
\hat{\pi}_{i3} = \frac{e^{\hat{\eta}_{i3}}}{1 + e^{\hat{\eta}_{i2}} + e^{\hat{\eta}_{i3}}}. \tag{2.6}
\]

Equating (2.5) and (2.6) we can write the MLE of $\eta_{ij}$ as

\[
\hat{\eta}_{i2} = \ln \frac{y_{i2}}{y_{i1}}
\]
\[
\hat{\eta}_{i3} = \ln \frac{y_{i3}}{y_{i1}}. \tag{2.7}
\]

From (1.10) we have

\[
\hat{\eta}_{i2} = \hat{\beta}_1 + \hat{\beta}_2 x_i
\]
\[
\hat{\eta}_{i3} = \hat{\beta}_3 + \hat{\beta}_4 x_i.
\]

For the case of two support points, the explicit solutions to the MLE equations give $\hat{\beta}$ as

\[
\hat{\beta}_1 = \frac{x_1 \hat{\eta}_{22} - x_2 \hat{\eta}_{12}}{x_1 - x_2}
\]
\[
\hat{\beta}_2 = \frac{\hat{\eta}_{12} - \hat{\eta}_{22}}{x_1 - x_2}
\]
\[
\hat{\beta}_3 = \frac{x_1 \hat{\eta}_{23} - x_2 \hat{\eta}_{13}}{x_1 - x_2}
\]
\[
\hat{\beta}_4 = \frac{\hat{\eta}_{13} - \hat{\eta}_{23}}{x_1 - x_2}. \tag{2.8}
\]

where $x_1 \neq x_2$. 

26
2.4.1 Implicit Solutions to MLE

As was shown in the previous section, there are special cases where explicit expressions can be found for ML estimators. However, in most cases, numerical methods are needed. These methods are typically iterative and based on a Newton-Raphson or Fisher scoring algorithm, Khuri et al. (2006).

Consider the multinomial response with three categories again, but now with a three support point design. The ML equations from (2.4) with $s = 3$ are

\[
\begin{align*}
y_{12} - n_1 \hat{\pi}_{12} + y_{22} - n_2 \hat{\pi}_{22} + y_{32} - n_3 \hat{\pi}_{32} &= 0 \\
x_1 y_{12} - n_1 \hat{x}_{12} + x_2 y_{22} - n_2 \hat{x}_{22} + x_3 y_{32} - n_3 \hat{x}_{32} &= 0 \\
y_{13} - n_1 \hat{\pi}_{13} + y_{23} - n_2 \hat{\pi}_{23} + y_{33} - n_3 \hat{\pi}_{33} &= 0 \\
x_1 y_{13} - n_1 \hat{x}_{13} + x_2 y_{23} - n_2 \hat{x}_{23} + x_3 y_{33} - n_3 \hat{x}_{33} &= 0
\end{align*}
\]

which, in this case, gives four equations and six values of $y_{ij}$.

We have a mismatch between number of equations and values of $y_{ij}$ because the four $\beta$ parameters yield four ML equations, when the log-likelihood is differentiated with respect to each parameter. However, two predictor equations, (1.10), with three support points yield six values of $y_{ij}$. This mismatch in equations and $y_{ij}$ values prevents explicit solutions for $\hat{\beta}$ from being obtained.

The relationship between the number of categories in the multinomial response, and hence the number of predictor equations, and the number of parameters in the predictor equations determine whether explicit solutions can be found for the MLE.

Say we have a multinomial response with four categories, this means there will
be three predictor equations,

\[
\begin{align*}
\eta_2 &= \beta_1 + \beta_2 x_i \\
\eta_3 &= \beta_3 + \beta_4 x_i \\
\eta_4 &= \beta_5 + \beta_6 x_i 
\end{align*}
\]

with a total number of parameters equal to six. Note there could be more or less than six parameters, depending upon the parametrization of the model.

To find the MLEs explicitly, we must have exactly \(6/3 = 2\) support points as this would give six equations and six unknowns. The number of design points required to achieve explicit MLE solutions is found by dividing the total number of parameters, six, by the total number of predictor equations, three.

Now suppose there are a total of seven parameters; this will give seven ML equations. For explicit solutions of the MLE, we must have \(7/3\) support points, but this is not an integer value. If we have two support points, this gives six values of \(y_{ij}, y_{12}, y_{13}, y_{14}, y_{22}, y_{23}, y_{24}\), but we still have seven equations. If we have three support points, this gives nine values of \(y_{ij}\), with the addition of \(y_{32}, y_{33}, y_{34}\). In this case, with seven parameters and three predictor equations, it is never possible to find an explicit solutions to the MLEs.

Generalizing this relationship, if we have a multinomial response with \(k\) categories, there will be \(k - 1\) predictor equations. The number of predictor equations is always one less than the number of multinomial categories, as the mult-category logit link expresses each category in terms of a baseline category. If the predictor equations contain \(r\) parameters then, in order for an explicit MLE solutions to exist, we must have \(r/(k - 1)\) support points, and this quantity must be an integer value.
2.5 Fisher’s Information Matrix

Fisher’s information matrix is a way of quantifying the amount of information about the unknown parameters. It is important in the field of experimental design because many design optimization techniques are based on maximizing aspects of the information matrix. Some of these are discussed in Section 3.2.

The covariance matrix of the score (2.2) is known as the Fisher information matrix. The information matrix is the negative expected value of the matrix of partial derivatives of the log likelihood function.

The \((i, j)^{th}\) element of Fisher’s information matrix, \(M\), is given by

\[
M(\xi, \beta)_{ij} = -E \left[ \frac{\partial^2 \ell}{\partial \beta_i \beta_j} \right]. \tag{2.9}
\]

which is equivalent to the \((i, j)^{th}\) element of the inverse of the asymptotic covariance matrix of the parameter estimates, Hald (1998, page 278). The information matrix is specified \(M = M(\xi, \beta)\) as it is evaluated for a particular design \(\xi\) and parameter values \(\beta\).

In the case of a multinomial response with three categories and a linear predictor with one covariate, taking the second partial derivative of the log-likelihood given by (2.1), gives the element in position \((1, 1)\) of \(M\) as

\[
\frac{\partial^2 \ell}{\partial \beta^2} = \frac{\partial}{\partial \beta_1} \frac{\partial \ell}{\partial \beta_1} = -\sum_{i=1}^{s} n_i \frac{\partial \pi_{i2}}{\partial \beta_1} = -\sum_{i=1}^{s} n_i \pi_{i2} (1 - \pi_{i2}),
\]

where \(s\) is the number of support points.
The element in position $(1, 3)$, and $(3, 1)$ due to the symmetry of $M$, is

$$\frac{\partial^2 \ell}{\partial \beta_1 \partial \beta_3} = \frac{\partial}{\partial \beta_1} \left( \frac{\partial \ell}{\partial \beta_3} \right) = \frac{\partial}{\partial \beta_1} s \sum_{i=1}^{s} (y_i \pi_i - n_i \pi_i^3)$$

$$= -\sum_{i=1}^{s} n_i \frac{\partial \pi_i}{\partial \beta_1} = s \sum_{i=1}^{s} n_i \pi_i \pi_i^2.$$

Taking the negative expected value of each matrix element, we can express the information matrix as a symmetric $(4 \times 4)$ matrix with upper triangle elements

$$M = \sum_{i=1}^{s} n_i \begin{bmatrix} \pi_i \pi_i^2 (1 - \pi_i) & \pi_i \pi_i^2 (1 - \pi_i) x_i & -\pi_i \pi_i^3 & -\pi_i \pi_i^3 x_i \\ \pi_i \pi_i^2 (1 - \pi_i) x_i & \pi_i \pi_i^2 \pi_i^3 x_i & -\pi_i \pi_i^2 \pi_i^3 x_i^2 & \pi_i \pi_i^3 (1 - \pi_i) x_i \\ \pi_i \pi_i^2 (1 - \pi_i) x_i & \pi_i \pi_i^2 \pi_i^3 x_i & \pi_i \pi_i^3 (1 - \pi_i) x_i & \pi_i \pi_i^3 (1 - \pi_i) x_i^2 \end{bmatrix}.$$  

(2.10)

The weight at the $i^{th}$ support point is the number of observations at that point divided by the total number of observations, $\omega_i = n_i / N$ where $N = \sum_{i=1}^{s} n_i$. We refer to weights rather than number of observations because it enables designs with different number of observations to be compared. For example, consider a design with two support points, $x_1^*$ and $x_2^*$. A design with, say, 9 observations at each point $n_1 = n_2 = 9$ will always be better than a design at the same support points but with fewer observations, say, $n_1 = n_2 = 3$. Looking at weights, in this case $\omega_1 = \omega_2 = 0.5$, eliminates the need to have to compare these two equivalent designs.
Since $N$ is a constant, we can express the information matrix as

$$M = N \sum_{i=1}^{s} \omega_i \begin{bmatrix} \pi_{i2} (1 - \pi_{i2}) & \pi_{i2} (1 - \pi_{i2}) x_i & -\pi_{i2} \pi_{i3} & -\pi_{i2} \pi_{i3} x_i \\ \pi_{i2} (1 - \pi_{i2}) x_i^2 & -\pi_{i2} \pi_{i3} x_i & -\pi_{i2} \pi_{i3} x_i^2 \\ \pi_{i3} (1 - \pi_{i3}) & \pi_{i3} (1 - \pi_{i3}) x_i & \pi_{i3} (1 - \pi_{i3}) x_i^2 \end{bmatrix}.$$  

The dimension of the information matrix is equal to the number of parameters in the model, which is dependent on the type of response and the number of covariate terms.

The likelihood function also determines the asymptotic covariance matrix of the maximum likelihood estimator, $\hat{\beta}$, Agresti (2002, page 137). The asymptotic covariance matrix of the parameter estimates is given as

$$\text{Cov} \left( \hat{\beta} \right) = \left( X^T W X \right)^{-1} = M^{-1}$$  

where the model matrix $X$ is

$$X = \begin{bmatrix} 1 & x_1 & 0 & 0 \\ 0 & 0 & 1 & x_1 \\ 1 & x_2 & 0 & 0 \\ 0 & 0 & 1 & x_2 \end{bmatrix}$$

and $W$ is the matrix of weights with elements
\[
W = \begin{bmatrix}
  n_1 \pi_{12} (1 - \pi_{12}) & -n_1 \pi_{12} \pi_{13} & 0 & 0 \\
  -n_1 \pi_{12} \pi_{13} & n_1 \pi_{13} (1 - \pi_{13}) & 0 & 0 \\
  0 & 0 & n_2 \pi_{22} (1 - \pi_{22}) & -n_2 \pi_{22} \pi_{213} \\
  0 & 0 & -n_2 \pi_{22} \pi_{23} & n_2 \pi_{23} (1 - \pi_{23}) 
\end{bmatrix}
\]

which can also be written as

\[
W = \begin{bmatrix}
  \text{Var}(Y_{12}) & \text{Cov}(Y_{12}, Y_{13}) & 0 & 0 \\
  \text{Cov}(Y_{12}, Y_{13}) & \text{Var}(Y_{13}) & 0 & 0 \\
  0 & 0 & \text{Var}(Y_{22}) & \text{Cov}(Y_{22}, Y_{23}) \\
  0 & 0 & \text{Cov}(Y_{22}, Y_{23}) & \text{Var}(Y_{23}) 
\end{bmatrix}.
\]

This can be denoted by the diagonal matrix

\[
W = \begin{bmatrix}
  Z_1 & 0 \\
  0 & Z_2 
\end{bmatrix}
\]

where

\[
Z_i = \begin{bmatrix}
  \text{Var}(Y_{i2}) & \text{Cov}(Y_{i2}, Y_{i3}) \\
  \text{Cov}(Y_{i2}, Y_{i3}) & \text{Var}(Y_{i3}) 
\end{bmatrix}, \quad i = 1, 2.
\]

As mentioned in Section 1.5, the multinomial distribution is multivariate in the response variable and this makes it different from the other distributions in the exponential family which have a univariate response. In a multinomial distribution with \( k \) categories, the value of the response for the \( k^{th} \) category can be determined from the values of the other \( k - 1 \) response variables, so that the observations are no longer independent.
A direct consequence of the correlated observations is seen in the form of $W$, which is block diagonal. This makes the algebra more difficult compared to the case of a univariate response, like the binomial, where $W$ is diagonal. See Agresti (2002, page 137) who derives diagonal matrices $W$ for a binomial and a Poisson response.
Chapter 3

Optimal Design Techniques for GLM

Optimal experimental design theory is a flexible approach used to design experiments. It is flexible because it can be used for any statistical model with any number of explanatory variables, both qualitative and quantitative, over any experimental design region and with any number of observations. When designing experiments using optimal design theory, the form of the model, whether it is linear or not, has important implications.

When the underlying model is linear, the optimal design process is not hindered by the model parameters being unknown because they do not enter the optimality criteria. As mentioned in Section 1.2, the covariance matrix of $Y$ is not a function of the model parameters. As such, most of the work in experimental design has been based on models with a continuous response variable where the error term is assumed to be normally distributed with a constant variance. When the model is linear, there are many classical design techniques to choose from, such as factorial designs or response surface methodology, and the optimal design can be determined explicitly, Khuri et al. (2006). For this
reason, in many situations, experiments are designed based on techniques for linear models even when this assumption is not valid.

When the conditions for a linear model are not satisfied, such as in experiments with a binary (for example, defective/non defective) or count (for example, number of defectives) response, GLMs are appropriate. However, in this case, the optimal design process becomes complicated because it depends on the values of the model parameters. That is, the parameters values must be known to design an optimal experiment to estimate the parameters. As mentioned in Section 1.3, for GLMs, the covariance matrix of $Y$ depends on the values of the model parameters. This conundrum has hindered the development of theory associated with experimental design for GLMs.

Performing an experiment allows us to find out information about the model. In particular, we can determine predictions of the responses and estimates of the parameters from the fitted model. The quality of the design is measured by the variances of the parameter estimates and predictions. The precision that can be achieved in estimating the unknown parameters of a model depends on the design: an optimal design will yield the smallest variances.

As the optimal design solution for a GLM depends on the unknown parameter values, the method used to find the optimal design must also incorporate an approach to deal with the unknown parameters. Some of these methods will be reviewed in this chapter. See Atkinson & Haines (1996) and Khuri et al. (2006) for detailed reviews.

### 3.1 Design Space

The design space is denoted by $\chi$ and represents the range of support point values of interest in the experiment. Ideally, the design space should be deter-
mined within the context of the experiment. For example, in Chapter 6 the design space is chosen as the range of acceptable doses of an immunization agent, in line with the original experiment.

In Chapter 4, arbitrary design spaces are used for the purpose of illustration. In Chapter 5, we introduce a quantifiable method for determining the design space and also illustrate how modifying the design space can result in different optimal designs when the minimization criteria is the mean squared error of the predicted probability.

3.2 Optimality Criteria

One of the earliest examples of optimal experimental designs is given by Smith (1918), who calculated optimal designs for one-factor polynomials. Her design criterion, which would later be called G-optimality, is described by the author as “... the curve of standard deviation with the lowest possible maximum value ...”. Some time passed before Wald (1943) compared designs using the determinant of the information matrix, which would lead to what is now called D-optimality. Further work in optimal experimental designs followed, including Plackett & Burman (1946), Elfving (1952, 1955, 1959) and de la Garza (1954). Chernoff (1953) published results on locally D-optimal designs for nonlinear models, Guest (1958) and Hoel (1958) expanded Smith’s results.

The alphabetical classification for this design criteria was introduced by Kiefer (1959). Atkinson et al. (2007, page 148) state that this paper and the publication of the equivalence theorem preceded much research in this field “... making use of results from the theory of convex optimization.” Other papers followed, with influential work published by Wynn (1970) and Fedorov (1972). For a more detailed history of optimal design see Silvey (1980), Pukelsheim (1993),
The alphabet optimality criteria, termed by Kiefer (1959), are functions of the information matrix. There are many different types of alphabet optimality criteria. See Goos (2002, page 16), Atkinson et al. (2007, page 135) and Silvey (1980) for detailed descriptions. The most commonly accepted criteria include A-, c-, D-, E- and G-optimality.

A-optimality minimizes the total or average asymptotic variance of the parameter estimates, and is equivalent to minimizing the trace of the inverse of the information matrix.

The c-optimality criterion minimizes a linear combination of the parameters with a minimum variance. Zocchi & Atkinson (1999) recommend using a c-optimal design if the purpose of the experiment is to estimate some function of the parameters. In Atkinson et al. (1995), the authors used a c-optimal design to estimate the dose required for a specified probability of death.

D-optimality, the most popular criterion, is the topic of research presented in Chapter 4. It minimizes the asymptotic variance of the parameter estimates, which is equivalent to maximizing the determinant of the information matrix. This minimizes the volume of a confidence ellipsoid for the parameters, giving estimates with high precision. D-optimality is recommended by Zocchi & Atkinson (1999) if the purpose of the experiment is the precise estimation of the model parameters.

E-optimality minimizes the maximum of the variances of linear combinations of the parameter estimates, which is equivalent to minimizing the maximum eigenvalue of the inverse of the information matrix. G-optimality minimizes the maximum variance of the predicted values.

D-optimality is intuitively appealing and, by far, the most preferred criterion.
Goos (2002, page 37) gives three other reasons why this is the case.

Firstly, it is a consistent method. As stated by Atkinson et al. (2007, page 153) “... D-optimum designs also perform well according to other criteria”. Donev & Atkinson (1988) gave an example of a D-optimal design that performs well under other alphabet criteria. Atkinson et al. (2007) showed a D-optimal design was also optimal under G-optimality criteria. Goos (2002) states that the opposite is often not the case and gave an example of an A-optimal design which performs poorly under the D-optimal design criterion.

Secondly, D-optimality is invariant to a linear transformation of the design matrix $X$, as the D-criterion value of the transformed design matrix is proportional to that of the original matrix. Goos (2002) demonstrates this by letting $Z = XA$ where $A$ is a $(p \times p)$ matrix that does not depend on the design. The D-criterion value becomes $|Z^TZ| = |(XA)^T(XA)| = |A|^2 |X^TX|$. 

Thirdly, D-optimal designs are facilitated by the availability of update formula for calculating the information matrix, its determinant and inverse after a design change. When constructing designs, points are added and deleted. Rather than re-calculate the information matrix again, the update formula can be used. They simplify the calculations of these matrices and determinants as they use the fact that the information matrix can be written as a sum of outer products. See Goos (2002, page 40) for the theory and a simple example of its application.

\[ \text{3.3 Locally Optimal Designs} \]

The most simple and widely used method to design an experiment is to use a best guess for the unknown parameters. These designs are called locally optimal designs, which were introduced by Chernoff (1953).
The information matrix for a GLM is a function of the unknown parameters. Hence, an optimal design based on an information matrix criterion will also be a function of the unknown parameters.

Locally optimal designs are intuitively appealing and relatively easy to construct, but they are optimal only for specific parameter values. The main drawback of this method is that good estimates of the unknown parameters are usually not available before the experiment has been performed. If the initial parameter estimate is poor, then the resulting design will be far from optimal. Furthermore, once the optimal design is found, it is applicable only for the parameter estimates used.

Dror & Steinberg (2006, page 520) state that “... two experiments having the same model but different coefficient values will typically require different designs.” In Chapter 4 we demonstrate the different optimal designs for the same model but with different parameter values for a multinomial response.

Despite these facts, there are still good reasons why locally optimal designs are important and used in practice. Some of these reasons are given below; see Ford et al. (1992) and Sitter & Fainaru (1997) for more detail.

In the situation when parameter estimates are available via previous experiments, pilot studies or consulting an expert in the field, the locally optimal designs are an available alternative.

Locally optimal design methods can be used in conjunction with other design methods. Ozol-Godfrey et al. (2008) used fractional design space plots to evaluate designs for GLMs which require initial estimates to be provided. The clustering procedure of Dror & Steinberg (2006) requires the ability to find locally D-optimal designs. Chipman & Welch (1996) used parameter estimates from an initial factorial design in the locally optimal design of a subsequent experi-
ment in a minimax approach to make the design more robust to the unknown parameters.

Ford et al. (1992) suggested that, for a multiple batch sequential design, one batch is used to determine the parameter estimates, which are then used to find the locally optimal design for the next batch of observations. Similarly Abdelbasit & Plackett (1983) derived a locally optimal design which was incorporated into a two stage sequential procedure.

Perevozskaya et al. (2003) developed locally optimal D-optimal designs for the proportional odds model for ordinal multinomial data with four categories. They also explored designs for the simultaneous estimation of multiple quantiles and compare designs in restricted and unrestricted design spaces.

The midpoint design, detailed in Section 4.5, uses a locally optimal design technique where the parameter estimates are given by the midpoint of the model parameter ranges.

Locally optimal designs provide a reference or benchmark for comparing the performance of other designs. Woods et al. (2006) assessed the performance of their design technique, under model uncertainty, relative to locally optimal designs across a range of model assumptions. They found that when the ranges of the parameter values are large, their designs performed better than a locally optimal design. Sitter (1992) compared his minimax approach with locally optimal designs. He stated that his method is marginally superior to locally optimal designs with correct parameter estimates but, when the initial parameter estimates are poor, the locally optimal design under performs to a much greater degree. Chaloner & Larntz (1989) compared the results of Bayesian designs with locally optimal designs and highlighted cases where the locally optimal design underperformed the Bayesian approach.
The IMSE-optimal designs in Chapter 5 are a locally optimal method. The
cparameter estimates are obtained by a best guess and the optimization criterion
is based on the mean squared error of the estimated probabilities.

In practice, the sensitivity of locally optimum designs to the choice of a param-
eter is an issue, as a poor guess for the true parameters will lead to an inefficient
design.

3.4 Bayesian Approach

Locally optimum designs are based on point estimates for the unknown param-
eters. In the Bayesian design approach, point estimates are replaced by a prior
distribution. Thus prior uncertainty about the parameters can be incorporated
into the model and, for this reason, the Bayesian approach to experimental
design is considered robust to poor parameter estimates.

The Bayesian optimal design is found by maximizing the expected utility over
a prior distribution for some function of the information matrix. The utility
function is chosen based on the goals of the experiment. The choice of util-
ity is important because a design that is optimal under one function may not
necessarily be so under another.

There are a variety of design criteria that can be used, depending upon the
purpose of the experiment. In Chaloner & Larntz (1989) where estimating the
parameters is the main objective, a popular utility function is based on Shannon
information. This translates to maximizing the average utility over the prior
distribution of the log of the determinant of the information matrix, which
is equivalent to D-optimality for locally optimal designs. Chaloner & Larntz
(1989) also looked at other Bayesian criteria, analogous to A-optimal design
for locally optimal designs, where they minimized the approximate expected
posterior variance of the quantities of interest. That is, they minimized the weighted trace of the inverse of the information matrix averaged over the prior distribution.

A drawback of Bayesian design is that, when integrating over the prior distribution, the integrals tend to be complicated or intractable and approximations must be used. Another downside is the subjective decision making required in Bayesian design techniques, such as the specification of the prior and utility functions, see Firth & Hinde (1997). In the concluding remarks in Chaloner & Verdinelli (1995), the authors state that “... very little guidance is available on how to collect and quantify such information”.

The strength of the Bayesian approach is its lack of dependence on point estimates of the model parameters. Chaloner & Verdinelli (1995, page 291) gave an example that showed that the more closely the prior distribution reflects the parameter estimates, the better the resulting Bayesian optimal design. They compared the performance of two Bayesian designs with an evenly spaced and equally weighted design over the range \((2.5, 5.0)\), referred to as the original design. One Bayesian design used the responses obtained from the original experiment to construct a Beta prior distribution on the parameters. The other Bayesian design used a uniform prior distribution for the parameters, reflecting the increased uncertainty before the original experiment was performed.

The authors compared these designs using the \(\phi_2\)-criterion, the Bayesian equivalent of A-optimal design. They showed that, compared to the equally spaced and weighted six point design, both Bayesian designs significantly reduced the variability of the estimates, in particular, the Bayesian design with the Beta prior performed the best. However, they also noted that when the original design was modified, by removing the two extreme design points, the resulting four point design showed a great improvement in efficiency, which was very close to
the Bayesian designs.

Chaloner & Larntz (1989) found a Bayesian design by using uniform priors and the $\phi_1$-criterion, the Bayesian equivalent of D-optimality. The method required that the number of design points be specified. They ran the optimization for two support points, then repeated the procedure for three support points, and so on, until their optimization criteria was maximized for the minimum number of support points. They demonstrated how the number of support points in the optimal design increases as the prior becomes more dispersed. They also found that the computing time also increased with the growing uncertainty in the prior distribution. These findings are consistent with many other papers, for example Atkinson et al. (2007).

Fan & Chaloner (2004) derived what they termed limiting optimal designs, which are a closed form expression for a design that is approximately optimal in the limit. They developed these designs using a continuation-ratio model for a three-category multinomial response. Agresti (2002, page 289) defined the continuation-ratio model as a logit based model for ordinal responses that is not based on cumulative probabilities.

Fan & Chaloner (2006) further developed this method and derived D-optimal designs for the continuation-ratio model with fixed slopes and an increasing intercept component. The authors also developed Bayesian designs for this model and for the case where the slopes are almost equal, which they defined as within 10% of each other.

Firth & Hinde (1997) found that Bayesian designs usually result in more support points than parameters. They suggested a new criterion, similar to Bayesian. They specified prior distributions for the parameters and simplified the utility function to develop a new design. They found many similarities between their method and the Bayesian approach. However, when they chose a highly infor-
mative prior they found that their design was very different from the Bayesian design.

Thall & Russell (1998) developed a Bayesian approach to find the optimal dose for an experimental treatment, using the proportional odds model, also referred to as a cumulative logits model, for a three-category ordered multinomial response. They used Bayesian methods to generate decision rules, incorporating information obtained during each trial, and used simulations under each dose response scenario to determine design parameters.

Zocchi & Atkinson (1999) applied Bayesian techniques to design efficient experiments to estimate the parameters of a model in a study of the influence of gamma radiation on the emergence of house flies. They used a three-category multinomial response and a hierarchical model with one covariate and five parameters. The authors developed a Bayesian design with the prior distribution derived from a previous experiment.

Since the Bayesian design would require the computation of a five-dimensional integral (as the model has five parameters) Zocchi & Atkinson (1999) approximated the prior distribution with a normal distribution, transforming the parameters if necessary, and used Monte Carlo methods to sample from this normal distribution. The authors noted that the assumption of normality can be avoided by using a Markov Chain Monte Carlo method to generate the parameter values.

Zocchi & Atkinson (1999) showed that the number of support points in the design increases with the uncertainty in the prior distribution. Also, the relative flatness of the derivative function indicates that there are other designs that are nearly optimal. The authors compared the Bayesian to locally optimal designs and found that they were very similar, although the Bayesian designs tended to have more design points.
The range design developed in Section 4.4 resembles a Bayesian approach in as much as the parameters are described by a distribution, rather than by point estimates. Comprehensive reviews of Bayesian optimal designs are given in Chaloner & Verdinelli (1995), Atkinson et al. (2007) and Khuri et al. (2006).

3.5 Minimax Methods

The minimax approach is intended to make the design more robust to the unknown parameters. It is a variant of locally optimal design techniques, minimax criterion requires specification of a region that contains the parameter values. This approach, suggested by Welch (1983), is constructed by choosing the “best” design amongst various locally D-optimal designs, corresponding to a collection of reasonable parameter values. The minimax design may not necessarily be the best design; it is a robust technique designed to protect against the worse possible performance.

Chipman & Welch (1996) specified a region which contains each parameter and, using data from an initial experiment, composed a set of feasible parameter vectors. For each individual vector, the D-optimal design was found from all the possible designs and relative efficiencies were calculated (see Section 4.2). The minimum efficiency was found for all the vectors initially specified and the design with the largest minimum efficiency was selected as optimal.

Sitter (1992) considered a restricted form of minimax design. Rather than using relative efficiencies, as in Chipman & Welch (1996), he generalized the criterion to minimizing the maximum of any optimality function under consideration, for example, the asymptotic confidence region of the parameters. Although the approach is intuitive, Sitter (1992) described it as “... mathematically intractable and numerically difficult ...” to implement. Hence he restricted the designs to
the set of \( kk \)-designs and a rectangular design region. (A \( kk \)-design allocates an equal number of observations to each of \( k \)-points symmetrically placed about the ED50 value).

Sitter (1992) found that his restricted minimax designs performed well when good estimates of the parameters were available. When only poor estimates were available, his method achieved results superior to those of locally optimal design.

King & Wong (2000) were motivated by minimax designs because the “... ranges are usually easier to specify in practice than providing a single best guess for each of the parameters”. They extended the work of Sitter (1992) by not restricting the class of available designs. Their method requires an initial starting design and continues iteratively by adding a single point, satisfying certain criteria, to the current design. See King & Wong (2000, page 1267) for the details of their search algorithm.

King & Wong (2000) applied their method to a fisheries example in Sitter (1992, page 1152), who found a restricted minimax D-optimal symmetric design of seven equally weighted support points, over the range (31.72, 48.28), with a D-efficiency of 0.96 (see Section 4.2). The King & Wong (2000) algorithm produced a minimax D-optimal design with nine support points and with a higher D-efficiency. They stated that the Sitter (1992) design is quite efficient, but the addition of two design points may be beneficial, especially if there is any uncertainty about the model assumptions. The authors demonstrated that the more iterations that are run on their algorithm, the higher the D-efficiencies obtained. In their own words “... when the algorithm stops, the generated design will have a minimax D-efficiency of at least \( \exp(-0.0005) = 0.9995 \)”.

Like Sitter (1992), King & Wong (2000) found minimax designs which lessen the detrimental effects of poor initial parameter estimates. They used a prior
estimate of the range of each parameter value because in practice, they argued, it is more realistic to be able to specify a range rather than a point estimate.

The restricted minimax designs of Sitter (1992) share a common feature with those of Chaloner & Larntz (1989) and King & Wong (2000). When initial parameter estimates are highly uncertain, the design will have more points with a greater spread.

The minimax approach discussed in these papers has been applied to the logistic distribution but can be applied to other nonlinear models. In Chapter 5, a minimax approach is used in IMSE-optimal design. It is different from the method discussed in the above papers because point estimates are used for the parameter values, rather than parameter ranges. The minimax approach is more similar, in concept, to the range design technique in Section 4.4, where ranges for the parameter values are specified and a number of possible parameter vectors are chosen from the ranges.

### 3.6 Clustering

This method, proposed by Dror & Steinberg (2006) is based on clustering a set of locally optimal designs. The motivation was to find a design method that performs well in spite of the uncertainty in the parameters, the linear predictor equation and the link function.

The method is based on finding D-optimal designs for a sequence of parameter vectors, which are combined into a single matrix. A clustering procedure on the matrix produces the optimal design. The authors gave a detailed algorithm summary on page 526. A simplistic example to illustrate the technique considers various four point D-optimal designs. The clustering procedure can be visualized...
as plotting all the design points on a chart and finding the robust design as the midpoints of each of the four clusters of support points.

The clustering design method does not constrain the support points to be equally spaced, but it does require them to be equally weighted. Dror & Steinberg (2006) suggested that, if desired, the weights can be adjusted using a method described as sequential quadratic programming.

When Dror & Steinberg (2006) compared clustering to Bayesian design methods by Chaloner & Larntz (1989), they found some cases where clustering outperformed Bayesian, but they concluded that Bayesian designs will usually outperform clustering designs. However, the benefit of using clustering designs is that they are much easier to construct and compute.

### 3.7 Compromise Designs

This method, proposed by Woods et al. (2006), seeks to produce designs that are as robust as possible to uncertainty in the values of the parameters, the functional form of the linear predictor, and the choice of the link function. The authors presented two applications and demonstrated the benefits over using a standard factorial design and a composite design.

The compromise design has a given number of support points and maximizes the average log determinant of the normalized information matrix. The uncertainty in the model is represented through sets of possible link functions, linear predictors and model parameters. To reduce computing time when evaluating integrals, rather than averaging over the entire model space, the authors averaged over a partial set, representative of the model space.

Woods et al. (2006) found that, when the ranges of the parameter values are not large, a reasonable choice was to use the range midpoints to calculate the
locally optimal design. They also recommended using preliminary runs to obtain initial estimates of the parameters, as it improves the information gained from successive experiments.

In other cases, a compromise design found from a coverage set usually outperformed a factorial design. A coverage set is defined as equally spaced points over the design range. For compromise designs across linear predictors and link functions, the authors found that compromise designs were more robust to the choice of model than locally optimal designs. In the examples considered, when determining the performance of the designs, the form of the linear predictor was more important than the type of link used.

When Dror & Steinberg (2006) compared methodologies, they found their clustering designs to be comparable to the compromise designs of Woods et al. (2006) when measured in terms of median efficiency. However, on the basis of minimum efficiency, the clustering designs were judged more robust as they had the smallest possible fraction of low efficiencies. As Dror & Steinberg (2006) stated, “... clustering creates a more robust design by decreasing the portion of the uncertainty space that, if discovered to be the true setup, would make the design seriously inefficient”.

Two robust design methods are considered in Chapter 4. We seek designs which perform well over a region of uncertainty in the parameter values. A range design which assumes a particular distribution for the parameter values, is considered in Section 4.4. A midpoint design method is introduced in Section 4.5 and compared to the range method.
3.8 Quantile Dispersion Graphs

Quantile dispersion graphs (QDGs) are a graphical technique used for comparing and assessing the quality of designs. They were first introduced by Khuri (1997) in the context of estimating variance components and further work followed in Lee & Khuri (1999, 2000). The technique was applied to nonlinear models in Khuri & Lee (1998).

Robinson & Khuri (2003) introduced the QDG technique for GLM designs. They applied it to a number of binary response examples. Khuri & Mukhopadhyay (2006) used QDGs for a Poisson response.

The basis of the QDG technique is the minimization of the mean squared error of prediction (MSEP) over the design region. This criterion is preferred because it includes estimation bias as opposed to the information matrix based measures, like D-optimal design, that just take variance into account.

Khuri & Mukhopadhyay (2006, page 210) provided a detailed account of the algorithm. In essence, it involves calculating various quantiles of the MSEP over a subset of parameter values. The minimum and maximum values are plotted against quantile values that range from 0 to 1, and conclusions can be drawn by analysis of these curves.

The QDGs provide information on the quality of the model’s prediction capability, with the better performing design having more stable and smaller quantile values. Also, it is desirable to have the minimum and maximum curves close together as this indicates the design is fairly robust and not overly dependent on the parameter values.

In their example, Khuri & Mukhopadhyay (2006) showed that the performance of the design can be assessed throughout the design region. They evaluated the
degree of sensitivity of the design to changes in parameter values, changes in the
link function (comparing log and square root links) and different distributions
(including negative binomial and gamma).

3.9 Sequential Approach

The sequential approach involves obtaining an initial estimate of the parameters
to determine the locally optimal design from which estimates of the response
can be obtained. The estimated responses are then used to update the values
of the parameter estimates. A new locally optimal design is generated and the
process is repeated until the method converges as per some pre-selected optim-
ality criteria. Sequential designs are usually categorized as fully sequential,
or batch sequential if the design is modified after each batch of observations.

Sequential designs are ideally suited to situations where experimental runs may
be expensive but response times are relatively quick. As Wu (1985, page 974)
says, “... the saving of a few runs by an efficient design outweighs the extra
effort required in designing a sequential experiment.”

The performance of a sequential method depends on the choice of the initial
values. This is usually a subjective decision, unless prior information is available.
Initial design values are commonly chosen as equally spaced values over the
design interval and include the end points.

The main advantage of a sequential design is provides continuous improvement
throughout the experiment. Dror & Steinberg (2008) give additional advan-
tages. A drawback is that it may become expensive to implement in practice;
see Anbar (1984).

Abdelbasit & Plackett (1983) used a sequential approach where, at each stage,
the parameters are estimated by maximum likelihood and the optimal design is found using D-optimality.

Wu (1985) developed a sequential design to estimate the percentiles of a quantal response curve. His designs performed well in certain cases, and he recommended further study.

Sequential procedures can also be used in the context of Bayesian design. Atkinson & Haines (1996) recommended using sequential procedures if the design is sensitive to prior assumptions. Dror & Steinberg (2008) initially used Bayesian design techniques to estimate model parameters to obtain a good initial design. They were concerned with small samples rather than asymptotic results.

\section{3.10 Summary}

The main obstacle in designs for experiments involving a GLM response is due to the dependence of the covariance matrix of the response on the model parameters. Hence, we have the paradoxical situation that we need to know the values of the unknown parameters in order to estimate the unknown parameters. This has increased the complexity and hindered the development of theory associated with experimental design for GLMs.

The simple and most widely used experimental design technique for GLMs is the use of locally optimal designs, which require a point estimate for the unknown parameters. Although these are valid design techniques, especially for benchmarking, the obvious drawback is that parameter values are seldom available before the experiment has been conducted. Furthermore, knowing the optimal design for one parameter set does not help in determining the optimal design for another parameter set.
Bayesian design techniques are a popular alternative to locally optimal designs due to their lack of dependence on point estimates. Instead, prior distributions for the unknown parameters are required. They can be specific distributions, if prior information is available, or uniform distributions to represent uncertainty. However, the Bayesian method, which has popular appeal, requires complicated integration and can be computationally demanding.

Minimax techniques do not rely on point estimates, nor on a prior distribution, but require a region containing each parameter to be specified. This method creates a number of designs across the parameter space, and the optimal minimax design is found by minimizing the maximum optimality criterion of the experiment. The minimax design is not intended to produce the best overall design; it is designed to prevent the worst possible scenario.

Minimax and Bayesian methods are robust to poor parameter estimates. Other robust techniques used to determine the optimal design for a GLM include clustering, compromise designs and quantile dispersion graphs. Sequential methods, although dependent on the choice of initial parameter estimates, can also be used in conjunction with other design techniques, such as Minimax and Bayesian.

Design techniques examined in the following chapters, include locally optimal and two robust designs. Two types of optimality criteria are also considered, namely minimization of the covariance of the parameter estimates and minimization of the mean square error of the estimated probability.
Chapter 4

D-Optimal Design

D-optimality was introduced in Section 3.2 as the most popular of the alphabetical optimality criteria.

The D-optimality criterion seeks designs that maximize the determinant of the information matrix. This is equivalent to minimizing the determinant of the covariance matrix of the parameter estimates. In other words, we are minimizing the volume of a confidence ellipsoid of the parameters, giving estimates with high precision. If convenient, natural logarithms can be used, and we either maximize the log determinant or minimize the negative log determinant of the information matrix.

D-optimal designs are model specific designs that can be used in cases when traditional or classical design methods are not appropriate. Examples of classical designs are incomplete block designs and fractional factorial designs. Some cases where classical design methods cannot be used are when the data cannot be modeled with a traditional linear model or where certain combinations of factor levels may be expensive or infeasible.

In this chapter, the theory behind D-optimal design will be discussed, including
the importance of the General Equivalence Theorem, which is fundamental to D-
optimal design. A number of locally D-optimal designs are developed, and two
methods based on maximizing the determinant of the information matrix are
used. A new design, called a range design, is proposed. It allows for uncertainty
in the parameter estimates over a defined range. Its performance is compared
to that of a midpoint design and of locally D-optimal designs.

The initial model under consideration is that of the multinomial response with
three categories and a linear predictor with one covariate. We start by finding
optimal designs for an equally weighted two point design and then find optimal
designs for more support points. The optimality of the designs is verified using
the General Equivalence Theorem and is illustrated in two and three dimen-
sions. Subsequently, designs when the linear predictor has two covariates and
an interaction term are also considered.

Locally optimal design methods are the simplest and most widely used. The
main drawback, however, is that we cannot explicitly state a general form for
a D-optimal design for a particular model. The locations of the design points
depend upon the values of the parameters. Furthermore, the exact number of
support points cannot be generalized, as it also depends upon the values of the
parameters.

4.1 General Equivalence Theorem

The General Equivalence Theorem (GET) is fundamental to the theory of opti-
mal design. It was originally developed by Kiefer & Wolfowitz (1960) for linear
models. The theorem does not depend on the linearity of the model, however
it can be more generally applied, in particular to GLMs. The GET is used
to construct and check designs, and applies to a wide variety of the alphabet
optimality criteria, many of which are based on the information matrix \( M(\xi) \), which is a function of the design \( \xi \). Atkinson et al. (2007, page 122) state that “The General Equivalence Theorem can be viewed as a consequence of the result that the derivatives are zero at a minimum of a smooth function over an unconstrained region.”

Let \( \Psi \{M(\xi)\} \) represent a general measure of imprecision for the design \( \xi \). In the context of D-optimal design, for example, this measure is the determinant of the inverse of the information matrix

\[
\Psi \{M(\xi)\} = \ln \left| M^{-1}(\xi) \right| = -\ln |M(\xi)| ,
\]

which is minimized.

Atkinson et al. (2007, page 122) showed that the function we are minimizing, \( \Psi \), depends on \( \xi \) through the information matrix \( M(\xi) \). Using the notation adopted by the authors, let \( \bar{\xi} \) be a one point design, denoted by \( x \), and let the design \( \xi' \) be given by \( \xi' = (1 - \alpha) \xi + \alpha \bar{\xi} \). Then the information matrix of \( \xi' \) can be written as

\[
M(\xi') = (1 - \alpha) M(\xi) + \alpha M(\bar{\xi}) ,
\]

and the derivative of \( \Psi \) in the direction of \( \bar{\xi} \) is

\[
\phi(x, \xi) = \lim_{\alpha \to 0} \frac{1}{\alpha} \left[ \Psi \{ (1 - \alpha) M(\xi) + \alpha M(\bar{\xi}) \} - \Psi \{M(\xi)\} \right] = \lim_{\alpha \to 0} \frac{1}{\alpha} \left[ \Psi \{ M(\xi') \} - \Psi \{M(\xi)\} \right]. \tag{4.1}
\]

The GET states that the following conditions are equivalent on \( \xi^* \), the optimal design:

1. The design \( \xi^* \) minimizes \( \Psi \{M(\xi)\} \);
2. The design $\xi^*$ maximizes the minimum of the standardized variance, or derivative function $\phi(x, \xi^*)$, given by (4.1), over the design space $\chi$;

3. The maximum value of $\phi(x, \xi^*)$ over $\chi$, is equal to $p$, the number of parameters in the model, and the maximum occurs at the support points of the optimal design $\xi^*$.

What this means is that the derivative function achieves a maximum value, equal to the number of parameters in the model, over the design space, and that this maximum occurs at the support points of the D-optimal design.

4.1.1 Carathéodory’s Theorem

Carathéodory’s Theorem is an integral result of complex analysis which relates convexity and dimension. Silvey (1980, page 72) adapts Carathéodory’s theorem to the context of optimal design for linear models and states that a continuous D-optimum design, based on a finite number of support points, denoted by $s$, has at most $p(p + 1)/2$ support points, where $p$ is the number of parameters.

Usually optimal designs contain fewer points. If a D-optimal design has $p$ support points, there is a probability $1/p$ at each point, that is, the D-optimal design will have equal weighting of observations at each support point (Silvey, 1980, page 42). Thus the bounds for a D-optimum design for a linear model are given by $p \leq s \leq p(p + 1)/2$.

Atkinson et al. (2007, page 152) noted that D-optimal designs are not necessarily unique, as there can exist different optimum designs, each with the same information matrix. As such, an optimal design can have $s$ support points where $s$ exceeds the upper limit.
The bound from Carathéodory’s theorem applies to a univariate response. However, when the response is multivariate, the D-optimal design can have fewer design points than number of parameters. This is because a multivariate variable produces more than one response per observation. For example, a multinomial response with $k$ categories will yield at least $k - 1$ responses per observation, which translates to the requirement for fewer design points.

### 4.1.2 GET for Multivariate Response

The General Equivalence Theorem of Kiefer & Wolfowitz (1960) is extended to multivariate logistic models in Atkinson & Haines (1996) and Zocchi & Atkinson (1999), where the expressions in this section have been sourced. The linear predictor has the form

$$\eta = X\beta,$$

where $\eta$ is a $k \times 1$ vector of linear predictors, $X$ denotes the $k \times (p + 1)$ design matrix and $\beta$ is a $(p + 1)$ dimensional vector of unknown parameters. The value $k$ represents the number of multinomial categories. The multivariate logistic transform is given by

$$\eta = C^T \log (L\pi) \quad (4.2)$$

where $L$ is a matrix of marginal indicators and $C$ is a matrix of contrasts.

The derivative matrix of $\eta$ with respect to $\pi$ ($\partial \eta / \partial \pi$) is given by

$$\frac{\partial \eta}{\partial \pi} = C^T D^{-1} L, \quad (4.3)$$

where $D = \text{diag}(L\pi)$. 

59
The derivative matrix of $\boldsymbol{\pi}$ with respect to $\boldsymbol{\beta}$ ($\partial \boldsymbol{\pi} / \partial \boldsymbol{\beta}$) is expressed as a function of $\mathbf{x}$, the point in the design space at which the prediction is made, and given by

$$
G (\mathbf{x}) = \frac{\partial \boldsymbol{\pi}}{\partial \boldsymbol{\beta}} = \left( \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\pi}} \right)^{-1} \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\beta}} = (\mathbf{C}^T \mathbf{D}^{-1} \mathbf{L})^{-1} \mathbf{X}.
$$

Depending upon the particular model, both derivative matrices given in (4.3) and (4.4) can also be obtained by directly evaluating the partial derivative.

Using these matrices, we can express the information matrix as

$$
\mathbf{M} (\xi, \beta) = \sum_{i=1}^{s} \left\{ \omega_i \mathbf{G}^T (\mathbf{x}) \text{diag} [\pi_{-1} (\mathbf{x}), \ldots, \pi_k^{-1} (\mathbf{x})] \mathbf{G} (\mathbf{x}) \right\},
$$

where $(\omega_1, \omega_2, \ldots, \omega_s)$ are the weights applied to the support points and $\sum_{i=1}^{s} \omega_i = 1$.

Now, consider the covariance matrix $\mathbf{V} (\mathbf{x})$ of $(Y_1, Y_2, \ldots, Y_k)$, given by

$$
\mathbf{V} (\mathbf{x}) = \text{diag} [\pi_1 (\mathbf{x}), \ldots, \pi_k (\mathbf{x})] - \boldsymbol{\pi} (\mathbf{x}) \boldsymbol{\pi} (\mathbf{x})^T
$$

where $\boldsymbol{\pi} (\mathbf{x}) = \{ \pi_1 (\mathbf{x}), \pi_2 (\mathbf{x}), \ldots, \pi_k (\mathbf{x}) \}$.

The standardized variance of the prediction at $\mathbf{x}$, also referred to as the derivative function, is given by

$$
\Phi (\mathbf{x}, \xi, \beta) = \text{trace} \left\{ \mathbf{v} (\mathbf{x}) g (\mathbf{x}) m^{-1} (\xi, \beta) g^T (\mathbf{x}) \right\}
$$

where $g (\mathbf{x}), v (\mathbf{x})$ and $m (\xi, \beta)$ are the respective matrices $G (\mathbf{x})$ in (4.4), $V (\mathbf{x})$ in (4.6), and $M (\xi, \beta)$ in (4.5), after deleting the first (or last) row and column.
(We delete the row and column corresponding to which category was specified as baseline in the model.)

This result was obtained from Zocchi & Atkinson (1999) and will be used later to demonstrate that a design is D-optimal.

Consider the three-category multinomial response model with a linear predictor with one covariate. The linear predictor equations, stated in (1.10) and (1.12), are

\[ \eta_2 = \beta_1 + \beta_2 x = \ln \left( \frac{\pi_2}{\pi_1} \right) \] and
\[ \eta_3 = \beta_3 + \beta_4 x = \ln \left( \frac{\pi_3}{\pi_1} \right) \] and \( \eta_1 = 0 \).

From (4.2) we can write the linear predictor as

\[
\begin{bmatrix}
\eta_1 \\
\eta_2 \\
\eta_3 \\
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & -1 & 1 & 0 \\
0 & -1 & 0 & 1 \\
\end{bmatrix}
\begin{bmatrix}
\pi_1 \\
\pi_2 \\
\pi_3 \\
\end{bmatrix}
\]

The derivative matrix of \( \eta \) with respect to \( \pi \) is given by

\[
\frac{\partial \eta}{\partial \pi} =
\begin{bmatrix}
1 & 1 & 1 \\
-\frac{1}{\pi_1} & \frac{1}{\pi_2} & 0 \\
-\frac{1}{\pi_1} & 0 & \frac{1}{\pi_3} \\
\end{bmatrix},
\]

and \( G(x) \), the derivative matrix of \( \pi \) with respect to \( \beta \), calculated using Maple-soft (2005), is

\[
G(x) =
\begin{bmatrix}
\pi_1 & -\pi_1 \pi_2 & -\pi_1 \pi_2 x & -\pi_1 \pi_3 & -\pi_1 \pi_3 x \\
\pi_2 & \pi_2 (1 - \pi_2) & \pi_2 (1 - \pi_2) x & -\pi_2 \pi_3 & -\pi_2 \pi_3 x \\
\pi_3 & -\pi_2 \pi_3 & -\pi_2 \pi_3 x & \pi_3 (1 - \pi_3) & \pi_3 (1 - \pi_3) x \\
\end{bmatrix}.
\]
The variance covariance matrix is

\[
V (x) = \begin{bmatrix}
\pi_1 (1 - \pi_1) & -\pi_1 \pi_2 & -\pi_1 \pi_3 \\
\pi_1 \pi_2 & \pi_2 (1 - \pi_2) & -\pi_2 \pi_3 \\
\pi_1 \pi_3 & -\pi_2 \pi_3 & \pi_3 (1 - \pi_3)
\end{bmatrix},
\]

and the information matrix is calculated using (4.5).

Expressions for \( g(x) \), \( v(x) \) and \( m(\xi, \beta) \) are obtained by removing the first row and column of \( G(x) \), \( V(x) \) and \( M(\xi, \beta) \) respectively, since we have designated the first category as baseline.

These results are used in Sections 4.6 and 6.4.3 to establish the D-optimality of a design.

### 4.2 D-Efficiency

The D-efficiency, or relative efficiency, as discussed in Jones & Goos (2007), is used to compare the quality of two designs \( \xi_1 \) and \( \xi_2 \) with information matrices \( M(\xi_1) \) and \( M(\xi_2) \) respectively. The D-efficiency of \( \xi_1 \) with respect to \( \xi_2 \), or ratio of two information matrices, is given as:

\[
D_{\text{eff}} = \left\{ \frac{|M(\xi_1)|}{|M(\xi_2)|} \right\}^{\frac{1}{p}}
\]

(4.8)

where \( p = \dim(\beta) \) is the number of parameters in the model. We take the \( p^{th} \) root so that the D-efficiency measure is a dimensionless ratio and will be comparable to the variance.

For example, if the design \( \xi_1 \) has 50% efficiency with respect to \( \xi^* \), that means that \( \xi^* \) requires 50% of the resources that \( \xi_1 \) requires, or \( \xi_1 \) requires twice as many resources as \( \xi^* \).
4.3 Methods to Find Optimal Design

To find the D-optimal design, we require a systematic approach to search a set of values and find the support points which maximize the determinant of the information matrix.

We will begin by considering the special case of a design with two equally weighted support points, that is, $\omega_1 = \omega_2 = 0.5$. As the weights are equal, we omit them and will denote the optimal design by $\xi^* = (x_1, x_2)$.

As we have two equally-weighted support points, the determinant of the information matrix, (2.11), simplifies to

$$\left| M(\xi, \beta) \right| = 0.5^4 (x_1 - x_2)^4 \pi_{11}\pi_{12}\pi_{13}\pi_{21}\pi_{22}\pi_{23}. \quad (4.9)$$

There are numerous ways to find maximums. Two methods will be discussed: a grid search and an optimization routine.

4.3.1 Machine Specifications

Computer processing was performed on a number of computers. Firstly: a DELL Intel Core duo @ 2GHz, 1 GB RAM laptop. Some processing was also carried out on an off-site computer, called Barossa. Barossa is a Linux Beowulf cluster with 155 dual 3GHz Pentium 4 nodes (152 compute nodes), supplied by Dell. It is located at the Australian Centre for Advanced Computing and Communication (AC3) within the Australian Technology Park in Redfern, Sydney. Also, a machine located at the University of Wollongong, called Runner, was used. This is a SUN 280R with a pair of 1.2GHz ultrasparc 3 CPUs with 4 GB RAM.
4.3.2 Grid Search

This approach places a grid over the design space and then searches the determinant values in the grid for the maximum. The steps are described as follows.

1. Determine the estimate of $\beta$.

2. Establish a design space $\chi$.

3. Select the number of intervals within the design region. This will determine the row $x_r$ and column $x_c$ labels that constitute the design space, referred to as the design grid.

4. For each unique support point pair $(x_r, x_c)$ in the design grid, calculate the determinant of the information matrix, as given in (4.9). Use (1.10) to calculate $\eta_{ij}$ and (1.13) to calculate $\pi_{ij}$. Since the support points are interchangeable, the values in the design grid are symmetric, hence only the upper-triangular values will be calculated, that is with $x_r < x_c$.

5. Search the design grid for the largest value of the determinant and note its row and column coordinates $(x_r^*, x_c^*)$.

6. The optimal design is the value of $\xi^* = (x_r, x_c)$ that maximizes $|M(\xi, \beta)|$ over the design space, for a particular $\beta$.

The following is an example of finding the D-optimal design using the grid search method.

1. Choose $\beta = (0, 1, 0, 1)^T$.

2. Let the design space be $\chi = [-10, 10] \times [-10, 10]$, which is denoted as $\chi = [-10, 10]$ in the examples that follow.

3. Use an interval of length 0.1 over the design space.
4. Calculate the determinant of the information matrix for each support point pair, $|M(\xi, \beta)|$, corresponding to the upper-triangle elements of the symmetric design grid.

5. The maximum value is $0.2487 \times 10^{-2}$ which is located at position $(-1.3, 3)$ on the design grid.

6. The optimal design on this grid is $\xi^* = (-1.3, 3)$.

Once the optimal design on this grid is located, we can refine the grid in the appropriate region and repeat the calculations to give more accuracy.

The optimal design can be visualized by plotting all the upper-triangle elements of the design grid. Figure 4.1 shows the contour plot for values of $|M(\xi^*, \beta)|$, with the optimal design $\xi^* = (-1.3, 3)$ located within the innermost contour.

![Contour plot](image)

**Figure 4.1:** Contour plot (for $x_1 < x_2$) of values of the determinant of the information matrix for $\beta = (0, 1, 0, 1)^T$.  

65
4.3.3 Optimization Routine

The D-optimal design is found by maximizing the determinant of the information matrix, which is equivalent to minimizing the negative value of the determinant of the information matrix. The optimization routine in R (R Development Core Team, 2009) used to minimize this objective function is called constrOptim. This routine uses a logarithmic barrier to enforce the constraints and minimizes the objective function over a defined region.

ConstrOptim requires several parameters to be initialized. These include the “feasible region”, which is the region where constrOptim will search for a minimum; “starting value”, which must lie within the feasible region; and “method”, which is set to “Nelder-Mead” when the gradient function is not provided.

The feasibility region constraints in constrOptim are defined by $a.x - b \geq 0$. For example, consider the feasibility region for a two point design $(x_1, x_2)$ in the design space $[-10, 10]$ where $x_1 \leq x_2$. These constraints can be described by the inequalities $-10 \leq x_1 \leq x_2 \leq 10$, which in matrix form are

\[
\begin{pmatrix}
1 & 0 \\
-1 & 1 \\
0 & -1
\end{pmatrix}
\begin{pmatrix}
x_1 \\
x_2
\end{pmatrix}
- \begin{pmatrix}
-10 \\
0 \\
-10
\end{pmatrix} \geq \begin{pmatrix}
0 \\
0 \\
0
\end{pmatrix}.
\]

In R syntax, $a = \text{rbind}(c(1,0), c(-1,1), c(0,-1))$ and $b = c(-10,0,-10)$. If the design has three points and $-10 \leq x_1 \leq x_2 \leq x_3 \leq 10$ then the matrix form of $a.x - b \geq 0$ is

\[
\begin{pmatrix}
1 & 0 & 0 \\
-1 & 1 & 0 \\
0 & -1 & 1 \\
0 & 0 & -1
\end{pmatrix}
\begin{pmatrix}
x_1 \\
x_2 \\
x_3
\end{pmatrix}
- \begin{pmatrix}
-10 \\
0 \\
0 \\
-10
\end{pmatrix} \geq \begin{pmatrix}
0 \\
0 \\
0 \\
0
\end{pmatrix}.
\]
The optimization routine methodology, using constrOptim, has the following steps:

1. Determine the estimate of $\beta$.

2. Establish a design space $\chi$ and obtain expressions for the feasibility constraints.

3. Choose a valid starting value $(x_1, x_2)$ that must lie in the design space.

4. Run constrOptim to minimize the determinant of the negative information matrix $-|M(\xi, \beta)|$.

5. ConstrOptim will return the optimal design and the minimum value of $-|M(\xi, \beta)|$.

For the design space $[-10, 10]$, the D-optimal design for $\beta = (1, 0, 1, 0)^T$ using constrOptim is $\xi^* = (-1.2813, 3.0251)$, which corresponds to a value of $|M(\xi^*, \beta)| = 0.2488 \times 10^{-2}$ and agrees with the result obtained using grid search, within the limitations of that search.

Using constrOptim to find the D-optimal design was faster and more accurate than using a grid search method, however with an exception. The convergence of constrOptim is dependent on the starting value. If the starting value is not sufficiently close to the optimal design, constrOptim diverges and will incorrectly return the starting value as the optimal design solution.

Table 4.1 shows four different starting values used to calculate the D-optimal design for $\beta = (0, 1, 0, 1)^T$ using constrOptim. The true D-optimal design, to one decimal place, $\xi^* = (-1.3, 3)$ was not achieved when the starting value was not close to the optimal design. This occurred for the starting values $(-10, -6)$ and $(-9.5, 5.5)$. 
Table 4.1: Starting values for constrOptim and resulting design solution, rounded to one decimal place.

<table>
<thead>
<tr>
<th>Starting Value</th>
<th>Optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-10, 2)</td>
<td>(-10, -6)</td>
</tr>
<tr>
<td>(-5, -2)</td>
<td>(-9.5, 5.5)</td>
</tr>
<tr>
<td>(-10, -6)</td>
<td>(-10, -6)</td>
</tr>
<tr>
<td>(-9.5, 5.5)</td>
<td>(-9.5, 5.5)</td>
</tr>
</tbody>
</table>

The divergence of the constrOptim algorithm is illustrated in the upper triangle segment of Figure 4.2. When the starting value is located in the shaded area, constrOptim diverges and incorrectly returns a value equal to the starting value. However, when the starting value is located anywhere in the unshaded area, constrOptim will converge to the true D-optimal design of $\xi^* = (-1.3, 3)$.

![Figure 4.2](image_url)

Figure 4.2: Shaded area showing where constrOptim diverges for $\beta = (0,1,0,1)^T$.

When a starting value is chosen in the shaded area of Figure 4.2, the value of the determinant is negligible and appears not to vary in any direction and so the optimization routine is not able to move from its starting point. Choosing a starting point for constrOptim to converge is difficult because the boundary of the shaded area in Figure 4.2 is not a uniform distance from the optimal design.
Also, different parameter values will yield differently shaped shaded areas. As such, this distance cannot be easily stated in general terms.

Given that there is no reliable method to determine the best starting value for constrOptim, a number of different starting values should be randomly generated.

4.4 Range Design

A locally optimal design is optimal for the parameter values used in the model. A robust design, on the other hand, aims to be optimal over a range of parameter values. There are also other types of robust designs, such as in Woods et al. (2006), where the optimal design allows for uncertainty in the choice of link function and the functional form of the linear predictor.

The robust design considered here allows for uncertainty in the parameters of the linear predictor. We define a range of parameter values and their distribution. If we have no information about the parameters, we can assume a flat, or uniform, distribution. Alternatively, we can assume other distributions, for example, a bell-shaped, or normal, curve which gives more probability to the mean and less to the tails.

We next consider a multinomial response with three categories, using the logistic link and a linear predictor with one covariate. The number of support points is fixed at two and the points are equally weighted. We specify the range for each $\beta$ parameter and take a random sample of $\beta$ vectors. For each individual $\beta$ we calculate the determinant of the information matrix $|M(\xi, \beta)|$, across the parameter space $\chi$. The range D-optimal design is found by maximizing the sum of these determinants over all the values of $\beta$. 
Some $\beta$ have very small determinants, while other $\beta$ have very large determinants. If we sum determinant values, the large determinant values dominate and produce range designs that do not reflect the full set of parameter values. Instead, we take natural logarithms before summing each determinant value to smooth the determinant values. Taking logs before summing the determinant values is equivalent to taking the log of the product of determinants.

To illustrate this calculation, we randomly generated 10,000 $\beta = (b_1, b_2, b_3, b_4)^T$ parameters from uniform distributions where $b_1 \sim U(-2, 2)$, $b_2 \sim U(-1, 3)$, $b_3 \sim U(-2, 2)$ and $b_4 \sim U(0, 4)$, to find the range design over the range $\chi = [-30, 30]$. Summing the matrices of determinant values, and finding the maximum value, gave the range design as $\xi_{sum}^R = (-21, 25.4)$. On the other hand, taking logs before summing (that is, taking the log of the product) gave a range design of $\xi_{logs}^R = (-1, 0.8)$.

These designs are very different, so we looked at the individual D-optimal designs for each of the 10,000 randomly generated parameter vectors. Taking the average of the first support point and the average of the second support point gave a design of $\xi_{avg}^* = (-1.4, 1.2)$, which is consistent with the range design found by taking logs, $\xi_{logs}^R$. When logs are not used, as in $\xi_{sum}^R$, the resulting optimal design is clearly being affected by large determinant values.

When we take logs, a small value $\epsilon$ is added to each value of $|M(\xi, \beta)|$ to avoid undefined values when $|M(\xi, \beta)|$ is very small. The value of $\epsilon$ must be very small, say the order $e^{-100}$, or else it will bias the value of the range design. For example, if $\epsilon = e^{-10}$ the range design changes to $\xi^R = (-1.5, -1.4)$, whereas if $\epsilon = e^{-100}$ or $\epsilon = e^{-500}$ then the range design is $\xi^R = (-1, 0.8)$.

The following steps describe the D-optimal range design methodology.

1. Establish a design space $\chi$. 

70
2. Determine a distribution that describes the model parameters, \( \beta \). Choose the number of simulations and randomly generate \( \beta \) from this distribution.

(a) When \( \beta \) are uniformly distributed, simulate from the standard uniform, \( U(0, 1) \), and convert to non-standard, \( U(a, b) \), using \( U_{ab} = (b - a) \times U_{01} + a \) where \( U_{01} \sim U(0, 1) \) and \( U_{ab} \sim U(a, b) \).

(b) When \( \beta \) are normally distributed, simulate from the standard normal, \( N(0, 1) \), and convert to non-standard \( N(\mu, \sigma^2) \), using \( X = Z\sigma + \mu \) where \( Z \sim N(0, 1) \) and \( X \sim N(\mu, \sigma^2) \).

3. For each randomly selected \( \beta \), perform the following steps:

(a) use the grid search method discussed in Section 4.3.2 to calculate the grid of determinant values \( |M(\xi, \beta)| \) that correspond to a range of designs across the design space. If we have two support points, then the support point pairs correspond to the row and column matrix elements, and the matrix will be symmetric;

(b) add a small value, \( \epsilon \), to each matrix element \( |M(\xi, \beta)| + \epsilon \); and

(c) take the natural logarithm of each matrix element \( \ln(|M(\xi, \beta)| + \epsilon) \).

4. Each matrix of \( |M(\xi, \beta)| \) values corresponds to a randomly generated \( \beta \).

Sum all the matrices to give one matrix containing values of \( \sum \ln|M(\xi, \beta)| \).

5. Search the matrix for the largest element and note its row and column location, which corresponds to the first, \( x_1 \), and second support point, \( x_2 \), respectively.

6. The range design is the value of \( \xi^R = (x_1, x_2) \) that maximizes \( \sum \ln|M(\xi, \beta)| \) over the design space.

Consider the following example to illustrate this technique. Using the design space \( \chi = [-60, 60] \), 10,000 \( \beta \) were randomly generated from the uniform dist-
tribution, with $b_1 \sim U(-2, 2)$, $b_2 \sim U(-1, 3)$, $b_3 \sim U(-2, 2)$ and $b_4 \sim U(0, 4)$. For each of the 10,000 $\beta$ vectors, a matrix of determinants, corresponding to the support points in the design grid, was calculated and a small value of $\epsilon = e^{-100}$ was added to each element before taking logarithms. The corresponding row-column elements, across all 10,000 matrices, were added to give one matrix containing values of $\sum \ln (|M(\xi, \beta)| + \epsilon)$. The range design, which maximized this matrix, was calculated as $\xi^R = (-1.04, 0.78)$.

The numbers in Table 4.2 show some of these randomly generated $\beta$ and their individual D-optimal designs.

<table>
<thead>
<tr>
<th>Run #</th>
<th>Random Generated $\beta$</th>
<th>D-optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>1883</td>
<td>(0.25, 1.96, 1.37, 0.73)</td>
<td>(-1.68, 1.14)</td>
</tr>
<tr>
<td>5557</td>
<td>(-0.7, 2.88, 0.29, 0.52)</td>
<td>(-0.8, 0.82)</td>
</tr>
<tr>
<td>9112</td>
<td>(1.2, 0.03, -0.01, 0.03)</td>
<td>(-50, 82.3)</td>
</tr>
</tbody>
</table>

Table 4.2: D-optimal design for randomly generated $\beta$ from the uniform distribution where $b_1 \sim U(-2, 2)$, $b_2 \sim U(-1, 3)$, $b_3 \sim U(-2, 2)$ and $b_4 \sim U(0, 4)$.

Some of the D-optimal designs, such as run #1883 and #5557 in Table 4.2, are similar to the range design, which indicates the range design is performing well. For other random $\beta$, such as run #9112, the range design is not robust to changing values of $\beta$ in terms of D-efficiency.

In the sample of optimal designs for the randomly generated $\beta$ from uniform distributions, there are many optimal designs for which the range design is a poor approximation. Consider the box plots in Figure 4.3. The top plot shows the distribution of all the first support points, $x_1$, and the bottom plot shows the distribution of all the second support points, $x_2$, for all the D-optimal designs corresponding to 10,000 randomly generated $\beta$. 

72
Figure 4.3: Box plot of first and second support points \((x_1, x_2)\), top and bottom charts respectively, obtained from 10,000 randomly generated \(\beta \sim \text{Uniform}\).

Figure 4.3 shows box plots of the first and second support points separately. If a support point has \(x_1\), say, as an outlier, it does not imply that \(x_2\) is also an outlier. In this simulation of size 10,000, only 221 support points had both \(x_1\) and \(x_2\) as outlying values. When the corresponding category probabilities at these support points are examined, there is no discernible difference compared to the probabilities at the support points which had neither \(x_1\) nor \(x_2\) as outlying values. As such, we cannot make any conjecture about the probabilities at the outlying support points.

Both box plots in Figure 4.3 show outlying values, defined as being outside the interval \((Q_1 - 1.5 \times \text{IQR}, Q_3 + 1.5 \times \text{IQR})\), where the interquartile range is \(\text{IQR} = Q_3 - Q_1\). Thus, for the first support point, \(x_1\), outliers are defined to fall
outside the range \((-3.01, 0.81)\) and for the second support point, \(x_2\), outliers fall outside the range \((-1.24, 3.17)\). By this measure, 7.3\% of \(x_1\) values and 6.9\% of \(x_2\) values are considered outliers. Depending on one’s perspective, these percentages may or may not be considered small. What is of note is how far out some of these design points are. For example, some outlying design points are more than 30 standard deviations away from the mean.

These outlying designs are cases when the range design does not perform well. For example, if we assume \(\beta\) is uniformly distributed, as \(b_1 \sim U (-2, 2)\), \(b_2 \sim U (-1, 3)\), \(b_3 \sim U (-2, 2)\) and \(b_4 \sim U (0, 4)\), then we can calculate the range design as \(\xi_R = (-1.04, 0.78)\). However, if the true value of \(\beta\) is actually \((1.2, 0.03, -0.01, 0.03)\), which is possible under this uniform distribution, this leads to a very different locally optimal design \(\xi^* = (-50, 82.3)\). In this case, the range design is not robust to changing values on the \(\beta\) values in terms of D-efficiency.

To quantify the performance of these designs, we calculate the D-efficiency of the range design, \(\xi_R\), relative to the optimal design, \(\xi^*\), for a particular \(\beta\). Using (4.8), we calculate the determinant of the information matrix \(|M(\xi^*, \beta)|\) at the support points of the optimal design, \(\xi^*\), for a particular \(\beta\). The determinant of the range design matrix \(|M(\xi_R, \beta)|\) is calculated at the support points of the range design \(\xi_R\), at the same value of \(\beta\). The D-efficiency is then

\[
D_{\text{eff}} = \left\{ \frac{|M(\xi_R)|}{|M(\xi^*, \beta)|} \right\}^{\frac{1}{2}}.
\]

Table 4.3 uses the same three \(\beta\) from Table 4.2 and shows the maximum value of the information matrix determinant for both the optimal and range designs.

The first row of Table 4.3 (run 1883) shows \(\beta = (0.25, 1.96, 1.37, 0.73)^T\). This has an optimal design of \(\xi^* = (-1.68, 1.14)\) and a maximum determinant value of \(\max |M(\xi^*, \beta)| = 0.206 \times 10^{-3}\). The range design is \(\xi_R = (-1.04, 0.78)\) and
for this $\beta$, the corresponding value is $\max |M(\xi^*,\beta)| = 0.1196 \times 10^{-3}$. Since the determinant of the information matrices for the optimal design and the range design are very similar, the relative efficiency is quite high at $D_{\text{eff}} = 87.3\%$.

The histogram in Figure 4.4 shows the distribution of efficiencies of D-optimal designs for the random uniform $\beta$ and indicates that the range design performs well over this large number of values of $\beta$. That is, the value of the determinant of the information matrix of the range design, $\xi^R = (-1.04, 0.78)$ is very close to the maximum values of the determinants of the information matrices of the designs for a large proportion of the randomly generated $\beta$. In fact, almost one
half of the 10,000 designs achieved a D-efficiency over 80% and more than one quarter achieved a D-efficiency over 90%.

This is further highlighted by the scatter plots in Figure 4.5.

Figure 4.5: Distribution of optimal designs for randomly generated $\beta \sim$ Uniform. The left plot shows distribution of the top 10% of designs, ranked in terms of their D-efficiency, and the right plot shows the next top 10% of designs.

The scatter plot on the left of Figure 4.5 shows the top 10% of optimal range designs, ranked in terms of the efficiency of the range design with respect to the D-optimal design. The top 10% of designs are clustered around the range design, which reinforces the behavior displayed in Figure 4.4. The next top 10% of optimal designs, shown in the scatter plot on the right in Figure 4.5, are further away from the range design.

Table 4.4 summarizes additional simulations of size 10,000 from various uniform distributions. The range design is obtained by maximizing the product of the determinants of the information matrices corresponding to the $\beta$ vectors over the range of support points. The “D$_{eff}$ Quartiles” are the first, second (or median) and third quartiles of the D-efficiencies of all the optimal designs.
\[
\begin{array}{cccccc}
 b_1, b_2, b_3, b_4 & \sim & \text{Uniform} (a, b) \\
 b_1 & b_2 & b_3 & b_4 & \text{Range Design} & D_{\text{eff}}, \text{Quartile} \\
\begin{array}{cccc}
 (-1, 1) & (-2, 0) & (-1, 1) & (0, 2) & (-1.06, 1.04) & 75\% \ 87\% \ 94\% \\
 (-2, 0) & (-2, 0) & (-2, 0) & (0, 2) & (-1.2, 1.19) & 78\% \ 89\% \ 95\% \\
 (-2, 2) & (-1, 3) & (-2, 2) & (0, 4) & (-1.04, 0.78) & 66\% \ 80\% \ 91\% \\
 (0, 4) & (-3, 1) & (-1, 3) & (-1, 3) & (-0.6, 1.18) & 42\% \ 61\% \ 80\% \\
 (-4, 4) & (-2, 6) & (-1, 7) & (0, 8) & (-1.07, 0) & 35\% \ 57\% \ 77\% \\
 (0, 8) & (-2, 6) & (-2, 6) & (-5, 3) & (-1.02, 0.22) & 18\% \ 38\% \ 65\% \\
\end{array}
\end{array}
\]

Table 4.4: Range designs and quartiles for the efficiencies of designs when \(\beta\) is randomly generated from uniform distributions.

The uniform\((a, b)\) distributions in the first two rows of Table 4.4 have a domain of length two, \(b - a = 2\), the middle two rows have \(b - a = 4\) and the last two rows have \(b - a = 8\).

When the \(\beta\) are simulated from uniform distributions with domains of two units, the range designs achieve median D-efficiencies ranging from 87\% to 89\%, which are much larger than those associated with uniform distributions with domains of eight units, with median D-efficiencies ranging from 38\% to 57\%. Further simulations support the assumption that, as the range, or dispersion, of the parameter distributions increase, the D-efficiency or performance of the range design decreases.

To compare results, a similar study was designed where 10,000 \(\beta\) vectors were simulated from normal \(N(\mu, \sigma^2)\) distributions. For example, the first row of Table 4.5 shows the first component of \(\beta\) has a normal distribution with zero mean and standard deviation equal to 0.5. As with the uniform distribution, we have chosen normal distributions with increasing dispersions. The first two rows of Table 4.5 all have have standard deviations equal to \(\sigma = 0.5\), the middle two rows have \(\sigma = 1\) and the last two rows have have \(\sigma = 2\).
When the $\boldsymbol{\beta}$ are simulated from normal distributions with $\sigma = 0.5$, the range design achieves median D-efficiencies ranging from 87% to 95%. These are much higher than when the $\boldsymbol{\beta}$ are simulated from normal distributions with $\sigma = 2$, which have median D-efficiencies ranging from 51% to 55%. These simulations also indicate that as the variability of the parameter distributions increase, the D-efficiency or performance of the range design decreases.

This is also evident in the graphs in Figure 4.6. The top left shows the distribution of D-efficiency measures when 10,000 $\boldsymbol{\beta}$ parameter vectors are generated from the $N(0, 0.5^2)$ distribution. An increasing density near the larger efficiencies indicates a greater distribution of range designs performing well.

<table>
<thead>
<tr>
<th>$b_1$</th>
<th>$b_2$</th>
<th>$b_3$</th>
<th>$b_4$</th>
<th>Range Design</th>
<th>$D_{\text{eff}}$</th>
<th>Quartile %</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(0, 0.5^2)$</td>
<td>$(2, 0.5^2)$</td>
<td>$(-1, 0.5^2)$</td>
<td>$(-0.5, 0.5^2)$</td>
<td>$(-1.45, 0.4)$</td>
<td>90%</td>
<td>95%</td>
</tr>
<tr>
<td>$(1, 0.5^2)$</td>
<td>$(-1.5, 0.5^2)$</td>
<td>$(-2, 0.5^2)$</td>
<td>$(-1, 0.5^2)$</td>
<td>$(-0.88, 1.61)$</td>
<td>75%</td>
<td>87%</td>
</tr>
<tr>
<td>$(0, 1^2)$</td>
<td>$(-2, 1^2)$</td>
<td>$(-1, 1^2)$</td>
<td>$(0.5, 1^2)$</td>
<td>$(-0.46, 1.22)$</td>
<td>67%</td>
<td>82%</td>
</tr>
<tr>
<td>$(-0.5, 1^2)$</td>
<td>$(1, 1^2)$</td>
<td>$(0.5, 1^2)$</td>
<td>$(-1, 1^2)$</td>
<td>$(-0.64, 1.29)$</td>
<td>56%</td>
<td>75%</td>
</tr>
<tr>
<td>$(0.5, 2^2)$</td>
<td>$(2.5, 2^2)$</td>
<td>$(2.5, 2^2)$</td>
<td>$(-0.5, 2^2)$</td>
<td>$(-0.5, 0.77)$</td>
<td>28%</td>
<td>51%</td>
</tr>
<tr>
<td>$(-1, 2^2)$</td>
<td>$(-1.5, 2^2)$</td>
<td>$(1, 2^2)$</td>
<td>$(1, 2^2)$</td>
<td>$(-0.97, 0.43)$</td>
<td>31%</td>
<td>55%</td>
</tr>
</tbody>
</table>

Table 4.5: Range designs and quartiles for the efficiencies of designs when $\boldsymbol{\beta}$ is randomly generated from normal distributions.
As the spread of the distribution increases, the efficiency of the range design worsens. Notice the bottom right graph in Figure 4.6. When 10,000 $\beta$ vectors are simulated from $N(0, 0.5^2)$, the lower efficiency values have a higher density. Similar behaviour is displayed when $\beta$ is generated from the uniform distributions with increasing dispersions. As the variability in the underlying distribution of $\beta$ increases, the performance of the range design, as measured by its D-efficiency, decreases.
4.5 Midpoint Design

The range design was shown to be a valid alternative to D-optimal designs when the dispersion of the underlying distribution of the $\beta$ parameters is not large. How do these designs compare to a simple midpoint design. The midpoint design is a D-optimal design using, as the parameter estimates, the midpoint values of the parameter distributions used in the range design.

For example, consider the first row of Table 4.4 which gives the distribution for $\beta$ as uniform with a range of two units where $b_1 \sim U(-1, 1)$, $b_2 \sim U(-2, 0)$, $b_3 \sim U(-1, 1)$ and $b_4 \sim U(0, 2)$. When we sampled 10,000 $\beta$ vectors, the equally weighted, two-point, range design was found to be $\xi^R = (-1.06, 1.04)$.

To find the corresponding midpoint design, we take the midpoints of each parameter distribution to obtain $\beta = (0, -1, 0, 1)^T$, which gives an equally weighted two-point D-optimal design of $\xi^M = (-1.09, 1.09)$. This is listed in Table 4.6 along with the quartiles of the D-efficiencies, which are found by comparison to the D-optimal designs for each of the 10,000 $\beta$ vectors.

<table>
<thead>
<tr>
<th>Uniform Range</th>
<th>Midpoint of Uniform $\beta$</th>
<th>Midpoint Design $\xi^M$</th>
<th>$D_{\text{eff}}$ Quartiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>(0, -1, 0, 1)</td>
<td>(-1.09, 1.09)</td>
<td>75% 86% 94%</td>
</tr>
<tr>
<td>2</td>
<td>(-1, -1, -1, 1)</td>
<td>(-1.3, 1.3)</td>
<td>76% 87% 95%</td>
</tr>
<tr>
<td>4</td>
<td>(0, 1, 0, 2)</td>
<td>(-1.09, 1.09)</td>
<td>61% 77% 89%</td>
</tr>
<tr>
<td>4</td>
<td>(2, -1, 1, 1)</td>
<td>(-0.46, 1.46)</td>
<td>39% 60% 80%</td>
</tr>
<tr>
<td>8</td>
<td>(0, 2, 3, 4)</td>
<td>(-1.23, -0.27)</td>
<td>29% 55% 77%</td>
</tr>
<tr>
<td>8</td>
<td>(4, 2, 2, -1)</td>
<td>(-1.45, -0.19)</td>
<td>14% 37% 65%</td>
</tr>
</tbody>
</table>

Table 4.6: Optimal designs corresponding to the midpoints of the uniform distributions, given in Table 4.4, with quartiles of the efficiencies of these designs relative to the D-optimal designs obtained.

When the $\beta$ are simulated from uniform distributions with domains of two units, the midpoint designs achieve median D-efficiencies ranging from 86% to
87%. These are much larger than those of uniform distributions with domains of eight units, which give median D-efficiencies ranging from 37% to 55%. Further simulations support the assumption that, as the range, or dispersion, of the parameter distributions increase, the D-efficiency or performance of the midpoint design decreases.

The quartile values of the efficiencies of the midpoint designs, in Table 4.6, are comparable to those of the range designs, in Table 4.4, and both show the efficiency of the designs decreasing as the dispersion of the uniform distribution increases.

In Figure 4.7 we compare these designs by displaying the absolute difference between the first and second support points of the D-optimal designs using the range and midpoint design methods, for uniform distributions.

![Figure 4.7: Absolute difference between support points of range and midpoint designs for uniform distributions.](image)

What is apparent from Figure 4.7 is that when the range of the uniform distri-
distri-
butions is two units, the midpoint and range designs are very similar. As the range of the uniform distributions increases, the difference between the range and midpoint designs also increases.

We next examine the behaviour when 10,000 $\beta$ vectors are generated from normal distributions. Table 4.7 shows the midpoint $\beta$ vectors obtained corresponding to the normal distributions displayed in Table 4.5.

<table>
<thead>
<tr>
<th>Normal Std Dev</th>
<th>Midpoint of $\beta \sim$ Normal</th>
<th>Midpoint Design</th>
<th>$D_{\text{eff}}$ Quartiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>(0, 2, −1, −0.5)</td>
<td>(−1.61, 0.36)</td>
<td>89% 95% 98%</td>
</tr>
<tr>
<td>0.5</td>
<td>(1, −1.5, −2, −1)</td>
<td>(−0.91, 1.62)</td>
<td>75% 87% 95%</td>
</tr>
<tr>
<td>1</td>
<td>(0, −2, −1, 0.5)</td>
<td>(−0.36, 1.61)</td>
<td>63% 79% 91%</td>
</tr>
<tr>
<td>1</td>
<td>(−0.5, 1, 0.5, −1)</td>
<td>(−0.59, 1.59)</td>
<td>54% 73% 88%</td>
</tr>
<tr>
<td>2</td>
<td>(0.5, 2.5, 2.5, −0.5)</td>
<td>(−0.32, 1.02)</td>
<td>26% 48% 73%</td>
</tr>
<tr>
<td>2</td>
<td>(−1, −1.5, 1, 1)</td>
<td>(−1.55, 0.16)</td>
<td>25% 50% 74%</td>
</tr>
</tbody>
</table>

Table 4.7: Optimal designs corresponding to the midpoints of the normal distributions, given in Table 4.5, with quartiles of the efficiencies of these designs relative to the $D$-optimal designs obtained.

When we compare the quartiles of the $D$-efficiencies of the midpoint designs in Table 4.7, with those of the range designs given in Table 4.5, we see behaviour similar to that identified for the uniform distribution. Namely, as the dispersion of the normal distributions increases, the efficiencies of both the range and midpoint designs decrease.

This is further illustrated in Table 4.8, which shows the absolute difference between the first and second support points of the $D$-optimal designs using the range and midpoint parameter estimates, with underlying normal distributions.
Again, the same conclusion holds as for the uniform distribution. When the dispersion of the normal distributions increases, the difference between the range and midpoint designs also increases.

We have seen that, when simulating from uniform and normal distributions and the dispersions are small, the midpoint and range designs are comparable. Given this, we would recommend using the midpoint design method when the spread of the underlying distribution is small. This is because the midpoint design method is much quicker to implement than the range design method.

This result obtained for the multinomial distribution is consistent with Woods et al. (2006), who developed various types of compromise designs. For a compromise design that allowed for uncertainty in the parameter values of the model, the authors used a binomial response with a logistic link function. They found that, when the parameter values are not large, a “... reasonable choice of de-
sign ... is a locally optimal design found for the midpoints of the ranges of the parameter values."

The range design method is more intensive as it requires simulating $\beta$ values from each range and calculating an optimal design by summing the information matrix of each $\beta$. The midpoint design is much quicker to implement. It does not require any simulation and can be found by maximizing the information matrix for the midpoint $\beta$. For this reason, when time or resources are limited, the midpoint design is preferred over the range design.

Both methods performed better for smaller parameter ranges compared to larger ones. In the case of the midpoint design, this could be because a midpoint provides a good approximation of a range due to the proximity of these values. When the ranges of the $\beta$ values are large, it is preferable to use the range design method as the simulation process ensures a cross section of parameter estimates are incorporated into the design process. It can be argued that this should give a better representation of the range of values compared to a midpoint.

4.6 Establishing Optimality with GET

4.6.1 Linear Predictor with One Covariate

We consider the earlier designs with two support points and use the GET to check the optimality of these designs. The response is the three-category multinomial and the linear predictor has one covariate, given by (1.10) and (1.12). The parameter estimate is $\beta = (0, 1, 0, 1)^T$ which has an optimal, equally weighted, two point design at $\xi = (-1.3, 3)$. The derivative function, given by (4.7), is evaluated at the prediction points, $x$, across the design space, and is plotted in Figure 4.9.
This graph has unexpected features. The derivative function should be a maximum at the optimum support points and the maximum value should equal the number of parameters, $p = 4$. This is not the case in Figure 4.9 and indicates that this is not an optimal design.

To find the D-optimal design, the algorithm must be modified to remove the restriction of two support points and equal weighting.

To find the D-optimal design we maximize the determinant of the information matrix, given in (2.11), by using FORTRAN 77 and various IMSL subroutines (Visual Numerics, Inc., 1970-2007). In particular, the subroutine DLCONF, which minimizes a general objective function subject to linear equality/inequality constraints, finds the minimum of the negative determinant of the information matrix.

The input parameters used in this implementation of DLCONF are defined as
follows.

- **FCN** - function to be minimized, in this case a subroutine which evaluates the determinant of the information matrix.

- **NVAR** - the number of variables, in this case 20. (10 variables represent the support points and 10 variables represent the corresponding weights).

- **NCON** - total number of linear constraints, in this case one.

- **NEQ** - number of equality constraints, in this case one, which represents the sum of the weights equal to one.

- **A** - matrix of dimension \((\text{NCON} \times \text{NVAR})\) which contains the equality constraints in the first \(\text{NEQ}\) rows, followed by the linear inequality constraints. In this case, as we have no inequality constraints, the matrix \(A\) has dimension \((1 \times 20)\) with values

\[
A(1,j) = \begin{cases} 
0 & j = 1, \ldots, 10 \\
1 & j = 11, \ldots, 20 
\end{cases}
\]

where the 0’s represent the equality constraints on \(x_i\) and the 1’s represent the equality constraints on \(w_i\). (The weights are constrained to sum to one.)

- **B** - vector of length \(\text{NCON} = 1\), containing the value one, which is the right hand side of the linear constraint, \(w_1 + w_2 + \ldots + w_{10} = 1\).

- **XLB, XUB** - lower and upper bounds of variables, which are the minimum and maximum values of the design space.

The subroutine DLFDDS is used to find the determinant of the information matrix, using Cholesky factorization, which is achieved using another subroutine DLFDTD.
The initial starting values for the design points and weights were randomly simulated from the $U(0,1)$ distribution. The initial starting values were converted to the design space $[a, b]$ using the linear transformation $U(a, b) = (b - a) \times U(0,1) + a$. Each initial weight was adjusted by dividing it by the sum of all weights to ensure the weights summed to unity.

The initial design values are randomly simulated as DLCONF does not always converge to the optimal design. This is discussed in more detail in Section 4.6.2.

An upper value for the number of support points was chosen as $s = 10$ and the design space was set to $\chi = [-3, 3]$ for $\beta = (0, 1, 0, 1)^T$. A total of 100,000 initial starting values for the design points and weights were randomly generated from the $U(0,1)$ distribution and converted to the $[-3, 3]$ design space. The optimal design, which gave the maximum value of the determinant of the information matrix $|M(\xi^*, \beta)| = 0.2775 \times 10^{-2}$, was found to be a three point design at $x_1 = -1.687$, $x_2 = 0.548$ and $x_3 = 3$ with corresponding weights $\omega_1 = 0.418$, $\omega_2 = 0.2168$ and $\omega_3 = 0.3652$ and $\omega_4 = \ldots = \omega_{10} = 0$. The optimal design is given as

$$\xi^* = \begin{pmatrix} -1.687 & 0.548 & 3 \\ 0.418 & 0.2168 & 0.3652 \end{pmatrix}.$$ 

The other seven support points had zero weight.

When this optimal design with three support points is used to calculate the derivative function in (4.7), the resulting graph is displayed in Figure 4.10.
Figure 4.10: Derivative function confirming the optimality of the D-optimal design for $\beta = (0, 1, 0, 1)^T$ over $\chi = [-3, 3]$.

Figure 4.10 shows that the derivative function achieves a maximum of $p = 4$ over the design space, and this maximum occurs at the support points of the D-optimal design. The increasing derivative function at the support point on the upper boundary at $x_3$, means that, if the design space is increased, this support point will move towards the new boundary. For example, if the design space is broadened to $\chi = [-5, 5]$ then the optimal design becomes

$$
\xi^* = \begin{pmatrix}
-1.867 & 0.758 & 5 \\
0.3562 & 0.3799 & 0.2639
\end{pmatrix}
$$

which maximizes the determinant of the information matrix at $|M| = 0.4934 \times 10^{-2}$ and has standardized derivative function in Figure 4.11.
Figure 4.11: Derivative function confirming the optimality of the D-optimal design for $\beta = (0, 1, 0, 1)^T$ over $\chi = [-5, 5]$.

As one support point lies on the boundary, the result is somewhat artificial because when we increase the boundary we improve our result, that is, the determinant of the information matrix increases.

Figure 4.12 shows that the optimal design for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-3, 3]$ is

$$\xi^* = \begin{pmatrix} -1.386 & 0 & 1.386 \\ 0.3756 & 0.2488 & 0.3756 \end{pmatrix}$$

which maximizes the determinant of the information matrix at $|M| = 0.2252 \times 10^{-3}$. Note that the design is symmetric in the support points and weights and no support points lie on the boundary.
Figure 4.12: Derivative function confirming the optimality of the D-optimal design for \( \beta = (0, 1, 0, -1)^T \) over \( \chi = [-3, 3] \).

The derivative function in Figure 4.12 is also symmetric and decreasing at the optimal support points extremes. In this case, if we increase the design space to \( \chi = [-5, 5] \), the optimal design does not change and nor does the value of the maximum determinant of the information matrix, \( |M| = 0.2252 \times 10^{-3} \). The standardized variance function is given in Figure 4.13.
We demonstrated the different behaviour of the optimal designs for different parameter estimates when the design space was increased. When the parameter estimate was given by $\beta = (0, 1, 0, -1)^T$, a wider design space did not change the D-optimal design. However, when the parameter estimate was given by $\beta = (0, 1, 0, 1)^T$, a wider design space resulted in a different D-optimal design. The design point on the original boundary, with an increasing derivative function, moved to the new boundary when the design space was widened.

This characteristic, of the design points on the boundary increasing with the design space, has been noted by several authors, including Russell et al. (2009) and Atkinson et al. (2007). It further highlights some of the difficulties involved in trying to determine a rule for the D-optimal design for a particular model. In this case, a multinomial response and a linear predictor with one covariate can yield two very different types of D-optimal designs for given parameter estimates.
In practical terms, an optimal design calculated for a particular set of parameters, cannot be used to construct an optimal design for the same model but with different parameter values. This underlines the difficulty in determining the design rules for GLM models, because the designs are not robust to the values of the parameters.

4.6.2 Linear Predictor with Two Covariates

Previously, all models considered for the linear predictor only had one covariate. We now look at optimal designs when the linear predictor has two covariates. Our response remains the three-category multinomial, but the linear predictor equations have two covariates, denoted by \( x \) and \( z \):

\[
\begin{align*}
\eta_2 &= \beta_1 + \beta_2 x_i + \beta_3 z_i \\
\eta_3 &= \beta_4 + \beta_5 x_i + \beta_6 z_i.
\end{align*}
\]  

(4.10)

As we now have six \( \beta \) parameters, the information matrix is square and dimension six.

The D-optimal design is found using a FORTRAN program similar to that described in Section 4.6.1. The parameter estimate is \( \beta = (0, 1, 1, 0, 1, 1)^T \) and the algorithm will allow for up to \( s = 10 \) support points in a design space of \( \chi = [-4, 3] \times [-4, 3] \). A total of 100,000 initial starting values for the design points and weights were randomly generated from the \( U(0, 1) \) distribution, and the design points were converted to the \( [-4, 3] \times [-4, 3] \) design space.

The D-optimal design which maximized the determinant of the information
matrix, \(|M(\xi^*, \beta)| = 0.8653 \times 10^{-2}\), was found to be

\[
\xi^* = \begin{pmatrix}
-4 & -1.944 & 3 & 3 & 3 \\
3 & 3 & -4 & -1.944 & 3 \\
0.2615 & 0.1566 & 0.2615 & 0.1566 & 0.1638
\end{pmatrix}.
\]

To show that this design is optimal, we use the GET as described in Section 4.1. The design \(\xi^*\) is optimal if the maximum of the standardized variance, or derivative function, equals \(p\), the number of parameters in the model, and this occurs at each of the design points \(\xi^*\). The derivative function is calculated using (4.7) but since there are two covariate terms, we have a three dimensional contour plot, as shown in Figure 4.14.

![Figure 4.14: Derivative function showing optimality of D-optimal design for \(\beta = (0, 1, 1, 0, 1, 1)^T\) with five support points.](image)

The five dots in Figure 4.14 are the design points \(\xi^*\) which maximize the standardized variance, in this case at \(p = 6\). This can also be demonstrated in two dimensions by fixing the values of the \(z\) covariate at the optimal design points.
Figure 4.15: Standardized variance fixing $z$ covariate at $z = -4$ in the top left graph, $z = -1.944$ in the top right graph and $z = 3$ in the bottom graph.

The three graphs in Figure 4.15 show that when the values of the $z$ covariate are fixed at the optimal values, $z = -4, -1.944, 3$, the resulting standardized variance, calculated over the range of the $x$ covariate, equals the number of parameters in the model, $p = 6$, and the maximum occurs at each of the support points.

We can also plot the derivative function in two dimensions by fixing the values of the $x$ covariate at the optimal values, $x = -4, -1.944, 3$. In this case, as the unique values of the $x$ and $z$ support points are the same, the standardized variance charts obtained by fixing the $x$ covariate are the same as those in Figure 4.15, obtained by fixing the $z$ covariate.

Of the 100,000 randomly simulated initial designs, DLCONF did not always
converge to the D-optimal design. In fact, various simulations of size 100,000, each with different seed values, produced four different designs, see Table 4.8.

| $|M(\xi, \beta)|$ | Occurrence |
|------------------|------------|
| $0.8653 \times 10^{-2}$ | 25% |
| $0.7539 \times 10^{-2}$ | 6% |
| $0.7428 \times 10^{-2}$ | 66% |
| $0.3342 \times 10^{-2}$ | 3% |

Table 4.8: Maximum values of the determinant of the information matrix for $\beta = (0, 1, 1, 0, 1, 1)^T$ across various simulations, each of size 100,000, using different seed values.

The surprising results in Table 4.8 show that in only 25% of the simulations the randomly generated starting values converged to the D-optimal design, with a maximum value of $|M(\xi^*, \beta)| = 0.8653 \times 10^{-2}$.

Consider the design, $\xi_3$, which has a maximum value of $|M(\xi_3, \beta)| = 0.7428 \times 10^{-2}$, given in the third row of Table 4.8:

$$\xi_3 = \begin{pmatrix} -4 & -1.944 & 3 & 3 \\ 2.631 & 3 & -3.77 & 3 \\ 0.2405 & 0.2575 & 0.3332 & 0.1688 \end{pmatrix}.$$  

This design occurs most often (66% of the time) but it does not maximize the determinant of the information matrix and hence is not D-optimal. This can be illustrated using the GET in Figure 4.16.
Figure 4.16: Derivative function for the four point design in Table 4.8 for $\beta = (0, 1, 1, 0, 1, 1)^T$ that occurred most often (66%) in the simulations.

A comparison of the derivative functions in Figures 4.14 and 4.16 shows that these two designs are similar except that $\xi_3$ does not have a design point near the location $(x, z) = (3, -1.944)$. Since $\xi_3$ is not the optimal design, we expect that the derivative function will not reach maximums at the contour value of six. This is more apparent when we plot the derivative function in two dimensions, fixing the values of the $z$ covariate at those of the design points $z = -3.77, 2.631, 3$. 
The three graphs in Figure 4.17 show that, when the design is not D-optimal, the standardized variance, calculated over the range of the $x$ covariate, does not achieve a maximum of $p = 6$ and nor does the maximum occur at the support points.

The implications of a non-optimal design means that the resulting data obtained from the experiment may not be of the best quality, for example, not have the lowest prediction variance. When an observation is made at a non-optimal design point, more observations are needed to obtain the same quality of information compared to when an observation is made at an optimal design point.
4.6.3 Linear Predictor With Two Covariates and Interaction

Now consider adding an interaction term to a model with a linear predictor with two covariates $x$ and $z$:

$$
\eta_{i2} = \beta_1 + \beta_2 x_i + \beta_3 z_i + \beta_4 x_i z_i \quad (4.11)
$$

$$
\eta_{i3} = \beta_5 + \beta_6 x_i + \beta_7 z_i + \beta_8 x_i z_i.
$$

As we now have eight $\beta$ parameters, the information matrix is a square matrix of dimension eight.

The D-optimal design is found using a FORTRAN program similar to that described in Section 4.6.1. The parameter estimate is $\beta = (0, 1, 1, 1, 0, 1, 1, 1)^T$ and the algorithm will allow for up to $s = 12$ support points in a design space of $\chi = [-4, 3] \times [-4, 3]$. A total of 100,000 initial starting values for the design points and weights were randomly generated from the $U(0, 1)$ distribution and converted to the $[-4, 3] \times [-4, 3]$ design space.

The D-optimal design, which maximized the determinant of the information matrix at $|M(\xi^*, \beta)| = 0.2531 \times 10^{-1}$, was found to be the following ten point design

$$
\xi^* = \begin{pmatrix}
-4 & -4 & -4 & -1.552 & -1.099 \\
-4 & -1.552 & -0.892 & -4 & 3 \\
0.0754 & 0.0912 & 0.0608 & 0.0912 & 0.1028 \\
-0.892 & -0.583 & 3 & 3 & 3 \\
-4 & 3 & -1.099 & -0.583 & 3 \\
0.0608 & 0.1501 & 0.1028 & 0.1501 & 0.1148
\end{pmatrix}
$$

This design $\xi^*$ is optimal because the maximum of the standardized variance,
or derivative function, is \( p = 8 \) and this maximum value occurs at each of the
design points. See Figure 4.18.

![Figure 4.18: Derivative function showing optimality of D-optimal design for
\( \beta = (0, 1, 1, 1, 0, 1, 1, 1)^T \) with ten support points.](image)

This is demonstrated in two dimensions in Figure 4.19.
Figure 4.19: Standardized variance fixing $z$ covariate at $z = -4$ in the top left graph, $z = -1.552$ in the top right graph, $z = -1.099$ in the middle left graph, $z = -0.892$ in the middle right graph, $z = -0.583$ in the bottom left graph and $z = 3$ in the bottom right graph.

The six graphs in Figure 4.19 show that, when the values of the $z$ covariate are fixed at the optimal support points, $z = -4, -1.552, -1.099, -0.892, -0.583, 3$, the maximum of the derivative function, over the range of the $x$ covariate, is
\[ p = 8 \] and this occurs at each of the support points.

If we fix the values of the \( x \) covariate support points at each of \( x = -4, -1.552, -1.099, -0.892, -0.583, 3 \), which, in this case, are the same distinct values as the \( z \) covariate, we also obtain standardized variance functions, confirming that the design is D-optimal.

### 4.6.4 Quadratic Linear Predictor

Now consider adding a quadratic term to both linear predictor equations:

\[
\begin{align*}
\eta_{i2} &= \beta_1 + \beta_2 x_i + \beta_3 x_i^2 \quad (4.12) \\
\eta_{i3} &= \beta_4 + \beta_5 x_i + \beta_6 x_i^2.
\end{align*}
\]

We have six \( \beta \) parameters, so the information matrix is a square matrix of dimension six.

The D-optimal design is found using a FORTRAN program similar to that described in Section 4.6.1. The parameter estimate is \( \beta = (1, 1, 1, 1, 1)^T \) and the algorithm will allow for up to \( s = 10 \) support points in a design space of \( \chi = [-3, 3] \). A total of 100,000 initial starting values for the design points and weights were randomly generated from the \( U(0, 1) \) distribution and converted to the \([-3, 3]\) design space.

The D-optimal design, for which \( |M(\xi^*, \beta)| = 0.434 \times 10^{-4} \), was found to be the following five point design

\[
\xi^* = \begin{pmatrix} -3 & -1.819 & -0.498 & 0.778 & 3 \\
0.1372 & 0.2214 & 0.228 & 0.255 & 0.1584 \end{pmatrix}.
\]
Maintaining the design space as $\chi = [-3,3]$ but for different parameter values, $\boldsymbol{\beta} = (1,1,-1,1,-1)^T$, gave an equally weighted three-point D-optimal design,

$$
\xi^* = \begin{pmatrix}
-1.163 & 0.5 & 2.163 \\
1/3 & 1/3 & 1/3
\end{pmatrix}.
$$

with $|M(\xi^*, \beta)| = 0.6164 \times 10^{-4}$ as the maximum value of the determinant of the information matrix.
Figure 4.21: Derivative function showing optimality of D-optimal design for $\beta = (1, 1, -1, 1, 1, -1)^T$ with five support points over $\chi = [-3, 3]$.

Figure 4.21 confirms the optimality of this design.

Note the differences between these two designs, Figures 4.20 and 4.21. The former design has five support points and the latter design has three equally weighted support points. Also, the former design has support points on the boundaries of the design space, with increasing derivative functions, whereas the latter design does not. If we were to increase the width of the design space, we would expect the former design to change but not the latter.
In fact, Figure 4.22 shows the standardized variance functions for the D-optimal designs for both of these sets of parameter values. As expected, the D-optimal design for $\mathbf{\beta} = (1, 1, 1, 1, 1)^T$ (left graph) has changed over the larger design space, with the support points moving to the new boundary values. The D-optimal design for $\mathbf{\beta} = (1, 1, -1, 1, 1)^T$ (right graph) is the same over both design spaces.

These simple examples highlight the difficulty in determining the optimal design for a specific linear predictor and link function. A change in parameter values can result in very different designs, both in the number of support points and the behaviour at the boundaries of the design space.

### 4.7 Number of Support Points

The main drawback of locally D-optimal designs is the dependence on the unknown parameters. Furthermore, we cannot even generalize the number of support points in a D-optimal design for a particular model because the number
of support points can vary depending on the values of the parameters. We now explore this behaviour.

Similar behaviour was noted by Atkinson & Haines (1996, page 467) in the univariate response case. Using a logistic regression model with two explanatory variables and three parameters, they concluded that the number and location of the design points in a locally D-optimal design are “... highly dependent upon the best guess adopted for the parameter values.”

Carathéodory’s theorem, in Section 4.1.1, gives the bounds for the number of support points in a D-optimum design for a linear model as \( p \leq s \leq p (p + 1) / 2 \). If we translate this to a three-category multinomial response with four parameters \((\beta_1, \beta_2, \beta_3, \beta_4)\), the bounds on the number of support points would be \( 4 \leq s \leq 10 \).

However, the lower bound, four, does not hold because we can find D-optimal designs with fewer support points. This was also noted in Fan & Chaloner (2004) for a three-category multinomial response model with three parameters. They found that, for a particular parameter set, “... a 2-point design leads to an information matrix of full rank as the response is multinomial.”

Table 4.9 shows some D-optimal designs for a three-category multinomial response with one covariate linear predictor and a four parameter model over the design space \( \chi = [-3, 3] \).
Table 4.9: Various D-optimal designs for different $\beta$ parameter settings where the linear predictor has one covariate.

For all the $\beta$ parameters examined, the number of support points in each D-optimal design ranged between two and three. The optimality of these designs was verified using the GET as in Figure 4.23.

![Graph 1](image1.png)  ![Graph 2](image2.png)

Figure 4.23: Derivative function confirming the optimality of the D-optimal designs, the left graph is for $\beta = (1, 2, 3, 4)^T$ and the right graph is for $\beta = (1, 1, 1, 6)^T$.

The derivative function for the left graph is for $\beta = (1, 2, 3, 4)^T$ and the right graph is for $\beta = (1, 1, 1, 6)^T$. 

<table>
<thead>
<tr>
<th>$\beta^T$</th>
<th>D-Optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(0, 1, 0, 1)$</td>
<td>$x_i$ -1.687</td>
</tr>
<tr>
<td></td>
<td>$\omega_i$ 0.418</td>
</tr>
<tr>
<td>$(0, 1, 0, -1)$</td>
<td>$x_i$ -1.386</td>
</tr>
<tr>
<td></td>
<td>$\omega_i$ 0.376</td>
</tr>
<tr>
<td>$(1, 2, 3, 4)$</td>
<td>$x_i$ -1.264</td>
</tr>
<tr>
<td></td>
<td>$\omega_i$ 0.5</td>
</tr>
<tr>
<td>$(1, 1, 1, 6)$</td>
<td>$x_i$ -2.904</td>
</tr>
<tr>
<td></td>
<td>$\omega_i$ 0.246</td>
</tr>
<tr>
<td>$(0.1, 1, 6, 2)$</td>
<td>$x_i$ -3</td>
</tr>
<tr>
<td></td>
<td>$\omega_i$ 0.5</td>
</tr>
</tbody>
</table>
When we have two covariates in the linear predictor, (4.10), Table 4.10 shows some D-optimal designs over the design space $\chi = [-4, 3] \times [-4, 3]$.

<table>
<thead>
<tr>
<th>$\beta^T$</th>
<th>$x_i$</th>
<th>$z_i$</th>
<th>$\omega_i$</th>
<th>$x_i$</th>
<th>$z_i$</th>
<th>$\omega_i$</th>
<th>$x_i$</th>
<th>$z_i$</th>
<th>$\omega_i$</th>
<th>$x_i$</th>
<th>$z_i$</th>
<th>$\omega_i$</th>
<th>$x_i$</th>
<th>$z_i$</th>
<th>$\omega_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(0, 1, 1, 0, 1, 1)$</td>
<td>-4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1.944</td>
<td>3</td>
<td>0.2615</td>
<td>0.1566</td>
<td>0.2615</td>
<td>0.1638</td>
<td>3</td>
<td>1.944</td>
<td>3</td>
<td>3</td>
<td>1.944</td>
<td>3</td>
<td>3</td>
<td>1.944</td>
<td>3</td>
</tr>
<tr>
<td>$(1, 1, 1, 1, 1, 1)$</td>
<td>-4</td>
<td>3</td>
<td>0.86</td>
<td>3</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>-3.209</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>-3.209</td>
<td>3</td>
<td>-3.209</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>-3.209</td>
<td>3</td>
<td>0.1032</td>
<td>0.2631</td>
<td>0.0501</td>
<td>0.1032</td>
<td>3</td>
<td>-3.209</td>
<td>3</td>
<td>3</td>
<td>-3.209</td>
<td>3</td>
<td>3</td>
<td>-3.209</td>
<td>3</td>
</tr>
<tr>
<td>$(4, 1, 1, 3, 1, 1)$</td>
<td>-4</td>
<td>-1.106</td>
<td>0.1205</td>
<td>0.2994</td>
<td>0.1205</td>
<td>0.1602</td>
<td>1.699</td>
<td>3</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>3</td>
<td>0.1205</td>
<td>0.2994</td>
<td>0.1602</td>
</tr>
<tr>
<td></td>
<td>-1.106</td>
<td>1.699</td>
<td>0.1205</td>
<td>0.2994</td>
<td>0.1205</td>
<td>0.1602</td>
<td>-4</td>
<td>-1.106</td>
<td>1.699</td>
<td>3</td>
<td>1.699</td>
<td>1.699</td>
<td>-4</td>
<td>-1.106</td>
<td>1.699</td>
</tr>
<tr>
<td>$(0, 1, -1, 0, -1, 1)$</td>
<td>-4</td>
<td>-4</td>
<td>0.2552</td>
<td>0.1224</td>
<td>0.1224</td>
<td>0.2552</td>
<td>-2.537</td>
<td>3</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>3</td>
<td>0.2552</td>
<td>0.1224</td>
<td>0.2552</td>
</tr>
<tr>
<td></td>
<td>-2.537</td>
<td>3</td>
<td>0.1224</td>
<td>0.1224</td>
<td>0.1224</td>
<td>0.2552</td>
<td>1.537</td>
<td>3</td>
<td>3</td>
<td>-4</td>
<td>3</td>
<td>3</td>
<td>0.1224</td>
<td>0.1224</td>
<td>0.2552</td>
</tr>
<tr>
<td>$(1, 2, 4, 1, 1, 1)$</td>
<td>-4</td>
<td>-4</td>
<td>0.1993</td>
<td>0.1554</td>
<td>0.3035</td>
<td>0.1648</td>
<td>2.213</td>
<td>3</td>
<td>3</td>
<td>2.294</td>
<td>3</td>
<td>3</td>
<td>0.1993</td>
<td>0.1554</td>
<td>0.1648</td>
</tr>
<tr>
<td></td>
<td>2.213</td>
<td>3</td>
<td>0.1554</td>
<td>0.3035</td>
<td>0.1648</td>
<td>0.177</td>
<td>2.294</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2.294</td>
<td>3</td>
<td>0.1554</td>
<td>0.3035</td>
<td>0.177</td>
</tr>
</tbody>
</table>

Table 4.10: Various D-optimal designs for different $\beta$ parameter settings where the linear predictor has two covariate terms.

Over all the $\beta$ vectors examined, the number of support points in each D-optimal design ranged between five to seven. For a linear model with $p = 6$ parameters, Carathéodory’s theorem gives the bounds on the support points as $6 \leq s \leq 21$. However, these designs do not satisfy these bounds as the model is not linear.

Figure 4.24 shows the standardized variance of the seven point design for $\beta = (1, 1, 1, 1, 1, 1)^T$ in Table 4.10.
Figure 4.24: Derivative function confirming the optimality of the D-optimal design for $\beta = (1, 1, 1, 1, 1)^T$.

When include an interaction term, (4.11), Table 4.11 shows some optimal designs over the design space $\chi = [-4, 3] \times [-4, 3]$. 

108
Table 4.11: Various D-optimal designs for different $\beta$ parameter settings when the linear predictor has two covariates and an interaction term.

<table>
<thead>
<tr>
<th>$\beta^T$</th>
<th>D-Optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(0, 1, 1, 1)$</td>
<td>$x_1$ -4 -4 -4 -1.552 -1.099 -0.892 -0.583 3 3 3</td>
</tr>
<tr>
<td></td>
<td>$z_1$ -4 -1.552 -0.892 -4 3 -4 3 -1.099 -0.583 3</td>
</tr>
<tr>
<td></td>
<td>$\omega_1$ .0754 .0912 .0608 .0912 .1028 .0608 .1501 .1028 .1501 .1148</td>
</tr>
<tr>
<td>$(0, -1, -1, 1)$</td>
<td>$x_1$ -4 -1.347 -0.877 -0.459 0.54 0.851 1.1312 3</td>
</tr>
<tr>
<td></td>
<td>$z_1$ 0.849 1.314 3 3 -0.461 0.533 -4 -1.346 -0.84</td>
</tr>
<tr>
<td></td>
<td>$\omega_1$ .1215 .0841 .1104 .0233 .1816 .1458 .1216 .0822 .1295</td>
</tr>
<tr>
<td>$(0, 1, -1, 1)$</td>
<td>$x_1$ -4 -4 -1.474 -0.837 -0.167 3 3</td>
</tr>
<tr>
<td></td>
<td>$z_1$ -1.104 -0.625 -4 3 0.679 -0.869 0.689</td>
</tr>
<tr>
<td></td>
<td>$\omega_1$ .1792 .1228 .1221 .1232 .1046 .2238 .1243</td>
</tr>
<tr>
<td>$(1, 1, 1, 0.01)$</td>
<td>$x_1$ -4 -1.484 -0.089 3 3</td>
</tr>
<tr>
<td></td>
<td>$z_1$ 3 -1.484 -0.089 -4 3</td>
</tr>
<tr>
<td></td>
<td>$\omega_1$ .2488 .1983 .1791 .2488 .1249</td>
</tr>
<tr>
<td>$(2, 1, 1, 1)$</td>
<td>$x_1$ -4 -4 -4 -1.51 -0.988 -0.979 -0.352 3 3 3</td>
</tr>
<tr>
<td></td>
<td>$z_1$ -4 -0.988 -0.352 3 -4 3 -4 -1.51 -0.979 3</td>
</tr>
<tr>
<td></td>
<td>$\omega_1$ .0878 .0935 .0594 .0984 .0935 .1475 .0594 .0984 .1475 .1146</td>
</tr>
<tr>
<td>$(1, 1, 1, 2)$</td>
<td>$x_1$ -4 -0.983 -0.496 -0.397 1.009 3 3</td>
</tr>
<tr>
<td></td>
<td>$z_1$ -0.45 3 3 -4 -0.756 -1.881 1.236</td>
</tr>
<tr>
<td></td>
<td>$\omega_1$ .1258 .1196 .1773 .2308 .1009 .1218 .1239</td>
</tr>
</tbody>
</table>

Depending on the values of the parameters, the locally D-optimal designs range between five and ten support points.

Figure 4.25 shows the standardized variance of the ten point design for $\beta = (2, 1, 1, 1, 1, 1)^T$ in Table 4.11.
Figure 4.25: Derivative function confirming the optimality of the D-optimal design for $\beta = (2, 1, 1, 1, 1, 1, 1, 1)^T$.

These examples highlight the difficulties in determining the behaviour of D-optimal designs for GLMs. For the models discussed, we showed how the locations of the D-optimal design points can vary depending upon the values of the parameters. Furthermore, even for a particular linear predictor, the exact number of support points cannot be explicitly determined as it also varies depending upon the values of the parameters.

D-optimal designs with fewer support points than the lower bound of $p$, the number of model parameters, can be found when the response is multivariate. In the univariate case, we have a single response per observation. When the response is multinomial, we have multiple responses per observation. For example, in the case of a $k$-category multinomial, we have $k - 1$ responses per observation because the $k^{th}$ response can always be determined from the preceding $k - 1$. So one observation will provide more than one response, which may reduce the number of support points required.
4.7.1 Equivalent Models

We have looked at the different types of designs for various forms of the linear predictor: with one covariate, two covariates, and with an interaction term. We have seen that, in general, the more terms in the model, the greater the number of support points in the D-optimal design. This is not always the case, however, as it depends on the values of the parameters.

In this section we look at how the D-optimal design changes when we have “equivalent” linear predictors. By equivalent we mean models that reduce to the same terms for particular parameter values. For example, the two models \( \eta_2 = a_1 + b_1 x \) and \( \eta_3 = a_2 + b_2 x + c_2 z \) are termed equivalent when the parameter values satisfy \( a_1 = a_2, b_1 = b_2 \) and \( c_2 = 0 \).

Table 4.12 shows three different linear predictors \( \eta = X \beta \) and specified values of \( \beta \) that give models that reduce the number of terms in these linear predictors. The models, denoted by (A), (B), and (C) in Table 4.12, are all different, but for the given parameter values, \( \beta \), all reduce the linear predictor to one covariate \( \eta_2 = \eta_3 = 1 + x \).

<table>
<thead>
<tr>
<th>Model</th>
<th>Linear Predictor</th>
<th>( \beta^T )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A) One covariate</td>
<td>( \eta_2 = \beta_1 + \beta_2 x ) ( \eta_3 = \beta_3 + \beta_4 x )</td>
<td>(1, 1, 1, 1)</td>
</tr>
<tr>
<td>(B) Two covariates</td>
<td>( \eta_2 = \beta_1 + \beta_2 x + \beta_3 z ) ( \eta_3 = \beta_4 + \beta_5 x + \beta_6 z )</td>
<td>(1, 1, 0, 1, 0)</td>
</tr>
<tr>
<td>(C) Two covariates plus interaction</td>
<td>( \eta_2 = \beta_1 + \beta_2 x + \beta_3 z + \beta_4 x z ) ( \eta_3 = \beta_5 + \beta_6 x + \beta_7 z + \beta_8 x z )</td>
<td>(1, 1, 0, 0, 1, 0, 0)</td>
</tr>
</tbody>
</table>

Table 4.12: Models which reduce to one covariate linear predictor.

Note that even though these models reduce to the same form, they are not the same. Model (A) has one covariate, \( x \), whereas model (B) has two covariates,
and \( z \). Even though the coefficient of \( z \) is zero, model (B) still allows for this additional covariate.

For these models, consider the locally D-optimal designs, listed in Table 4.13. The design space for the one covariate model is \( \chi = [-4, 3] \) and for the two covariate models is \( \chi = [-4, 3] \times [-4, 3] \).

<table>
<thead>
<tr>
<th>Model</th>
<th>D-Optimal Design for Reduced model (one covariate)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A)</td>
<td>( x_i ) -2.807 -0.322 3 ( \omega_i ) 0.3769 0.3298 0.2933</td>
</tr>
<tr>
<td>(B)</td>
<td>( x_i ) -2.445 -2.445 -0.395 -0.395 3 3 ( z_i ) -4 3 -4 3 -4 3 ( \omega_i ) 0.1868 0.1868 0.1791 0.1791 0.1341 0.1341</td>
</tr>
<tr>
<td>(C)</td>
<td>( x_i ) -2.807 -2.807 -0.322 -0.322 3 3 ( z_i ) -4 3 -4 3 -4 3 ( \omega_i ) 0.1884 0.1884 0.1649 0.1649 0.1467 0.1467</td>
</tr>
</tbody>
</table>

Table 4.13: D-optimal designs for the models in Table 4.12.

From Table 4.13 it is apparent there is a high degree of similarity between these designs. In fact, the design for model (C) contains all the support points in the design for model (A), plus an additional three support points. The six design points in (C) are the three design points in (A), each repeated twice. The weights of the support points in (C) are half the weight of the corresponding support point in (A).

The D-optimal design for model (A) has three design points, Figure 4.26 shows the standardized variance for this model.
Figure 4.26: Standardized variance for D-optimal designs for equivalent models $\eta_2 = \eta_3 = 1 + x$, showing model (A) from Table 4.12.

When a second covariate term is added to the linear predictor, we have model (B), whose standardized variance is shown in Figure 4.27.
Model (C) has two covariates and an interaction term and Figure 4.28 shows the standardized variance for this model.
Figure 4.28: Standardized variance for D-optimal designs for equivalent models $\eta_2 = \eta_3 = 1 + x$, showing model (C) from Table 4.12.

All three designs have very similar $x$-coordinates located in the vicinity of $x = -3, -0, 3$, with models (B) and (C) incorporating three extra design points at the boundaries of the design space. The designs for these three equivalent models are very similar, as the parameter values result in these models reducing to the same model. Note that even though the models all reduce to $\eta_2 = \eta_3 = 1 + x$, the D-optimal designs are not identical, as each underlying model is different.

Consider the resulting linear predictors when we choose $\beta$ so that both models reduce to a linear predictor with two covariates.

<table>
<thead>
<tr>
<th>Model</th>
<th>Linear Predictor</th>
<th>$\beta^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D) Two covariates</td>
<td>$\eta_2 = \beta_1 + \beta_2x + \beta_3z$</td>
<td>(1, 1, 1, 1, 1, 1)</td>
</tr>
<tr>
<td></td>
<td>$\eta_3 = \beta_4 + \beta_5x + \beta_6z$</td>
<td></td>
</tr>
<tr>
<td>(E) Two covariates</td>
<td>$\eta_2 = \beta_1 + \beta_2x + \beta_3z + \beta_4xz$</td>
<td>(1, 1, 1, 0, 1, 1, 1, 0)</td>
</tr>
<tr>
<td>plus interaction</td>
<td>$\eta_3 = \beta_5 + \beta_6x + \beta_7z + \beta_8xz$</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.14: Models which reduce to a two covariate linear predictor.
Table 4.14 shows two linear predictors and specified values of $\beta$ which, when substituted into the models, reduce to $\eta_2 = \eta_3 = 1 + x + z$. The models are not identical: model (E) includes an interaction term.

Consider the D-optimal designs for these models, listed in Table 4.15.

<table>
<thead>
<tr>
<th>Model</th>
<th>D-Optimal Design for Reduced model (two covariates)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D)</td>
<td>$x_i$  $z_i$  $\omega_i$</td>
</tr>
<tr>
<td></td>
<td>-4 -4 -3.209 0.86 3 3 3</td>
</tr>
<tr>
<td></td>
<td>0.86 3 3 -4 -4 -3.209 3</td>
</tr>
<tr>
<td></td>
<td>0.1032 0.2631 0.0501 0.1032 0.2631 0.0501 0.1672</td>
</tr>
<tr>
<td>(E)</td>
<td>$x_i$  $z_i$  $\omega_i$</td>
</tr>
<tr>
<td></td>
<td>-4 -1.482 -0.089 3 3</td>
</tr>
<tr>
<td></td>
<td>3 -1.482 -0.089 -4 3</td>
</tr>
<tr>
<td></td>
<td>0.2488 0.1983 0.179 0.2488 0.125</td>
</tr>
</tbody>
</table>

Table 4.15: D-Optimal designs for the models listed in Table 4.14. Both models have design space $\chi = [-4, 3] \times [-4, 3]$.

Note that the model with more terms, (E), has fewer D-optimal design points. Also, there are similarities between designs, although not to the same extent as in the reduced one covariate models case of Table 4.13. Figure 4.29 shows the standardized variance plot for model (D) and Figure 4.30 shows the standardized variance of model (E).
Figure 4.29: Standardized variance for D-optimal designs for equivalent models $\eta_2 = \eta_3 = 1 + x + z$, showing model (D) from Table 4.14.

Figure 4.30: Standardized variance for D-optimal designs for equivalent models $\eta_2 = \eta_3 = 1 + x + z$, showing model (E) from Table 4.14.
The derivative functions of these two designs are different. Model (D), with more design points, has all support points located on the boundaries of the design space, whereas model (E) has two design points located towards the centre of the design space.

These examples clearly illustrate that, even though parameter values were chosen to reduce the different models to equivalent versions, the resulting D-optimal designs are not identical.

4.8 Relationship Between Parameters and Support Points

We can examine if there is a relationship between the parameter values $\beta = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ and the location of the optimal design, when there are two equally weighted support points.

Consider, (1.10), the three-category multinomial response and a linear predictor with one covariate, which is a four parameter model. When the parameter is $\beta = (0, 1, 0, 1)^T$ the D-optimal design is $\xi^* = (-1.28, 3.02)$. If $\beta = (0, 2, 0, 2)^T$, the D-optimal design is $\xi^* = (-0.64, 1.51)$ and if $\beta = (0, 0.25, 0, 0.25)^T$, the D-optimal design is $\xi^* = (-5.12, 12.09)$. These numbers arise from the following generalized this result.

If a set of parameter values, $\beta_f = (c_1, m_1, c_2, m_2)^T$, has D-optimal design $\xi_f^* = (a, b)$, where $a < b$, then a set of rescaled parameter values, $\beta_{II} = (c_1, km_1, c_2, km_2)^T$, has D-optimal design $\xi_{II}^* = (a/k, b/k)$ for $k \neq 0$.

This result is apparent when we take the linear predictor equations given by (1.10), and substitute $\beta_f$ to give $\eta_{12} = c_1 + m_1x_i$ and $\eta_{13} = c_2 + m_2x_i$. At the support point $\xi_f^* = (a, b)$ we have $\eta_{12} = c_1 + m_1a$, $\eta_{22} = c_1 + m_1b$, $\eta_{13} = c_2 + m_2a$
and \( \eta_{23} = c_2 + m_2 b \). If we substitute \( \beta_{II} \) into the linear predictor equations, we get \( \eta_{22} = c_1 + km_1 x_i \) and \( \eta_{33} = c_2 + km_2 x_i \). At the support point \( \xi_{II}^* = (a/k, b/k) \), we have \( \eta_{12} = c_1 + m_1 a \), \( \eta_{22} = c_1 + m_1 b \), \( \eta_{13} = c_2 + m_2 a \) and \( \eta_{23} = c_2 + m_2 b \). Thus the two sets of linear predictor equations are identical. This intuitive result is due to rescaling the parameters; the linear predictor equations have not changed.

### 4.9 Summary

D-optimal designs are model specific designs. They minimize the determinant of the covariance matrix of the parameter estimates or, equivalently, maximize the determinant of the information matrix. D-optimal designs are widely used because they are intuitively appealing.

We illustrated the technique of finding D-optimal designs in the simple case of an equally weighted design with two support points. The response variable was the multinomial with three categories and the linear predictor had one covariate and two parameters for each multinomial category, excluding the baseline category. Recall that, in the link function, the multinomial categories are expressed in terms of the baseline category.

Two methods to maximize the determinant of the information matrix were compared: a recursive optimization routine and a grid search algorithm. Both of these methods were implemented in R and FORTRAN and each method possesses strengths and weaknesses.

The grid search algorithm is computationally intensive and is much slower. It is very reliable because it will always converge to the optimal value in the defined search grid. The recursive optimization routine is much faster than the grid search as it evaluates the objective function only at selected points. However,
it is dependent on the starting value used, and some starting values cause the algorithm to diverge. This was found to occur when the starting value used was not located near the optimal design. This occurs when the values of the determinant are negligible and appear not to vary in any direction and so the optimization routine is not able to move from its starting point.

In an attempt to find a D-optimal design which is not dependent on point estimates of the parameter, a range design method, based on a range of parameter estimates, was developed. The number of support points is fixed at two, and are equally weighted. We specify the range for each parameter and take a random sample of vectors. For each individual vector we calculate the determinant of the information matrix. The optimal range design is found by maximizing the sum of the logarithms of these determinants.

The method was implemented by simulating parameter vectors from the uniform and normal distributions. We showed that the range design provided a fairly good approximation to the D-optimal design. However, as the dispersion of the underlying distribution increased, the performance of the range design decreased.

The range design method was bench-marked against the midpoint technique. The midpoint technique finds the locally D-optimal design where the parameter estimates are obtained by the midpoint of the parameter ranges. The midpoint design was a good approximation for the range design, when the range, or spread, of the underlying distribution was not too large. In this case, it is preferable to use the midpoint design as it is much quicker to implement.

Up to this point, the selected designs were constrained to have two equally weighted design points. However, depending upon the parameter values, a two point equally weighted design may not be D-optimal. The general equivalence theorem, which is fundamental to the theory of optimal design, was used to
demonstrate the optimality of the designs. This occurs when the derivative function is maximized at \( p \), the number of parameters in the model, and the maximum occurs at each support point.

The bounds for the number of support points in a D-optimum design for a linear model, as per Carathéodory’s theorem, are given by \( p \leq s \leq p(p + 1)/2 \). We showed that this does not hold when the response is multivariate. Furthermore, for a GLM with a multivariate response, no theory has yet been developed to determine the bounds on the number of support points for a particular linear predictor. This again comes back to the fact that the design solution is a function of the unknown parameters of the model. In the multinomial response case, we examined various model parametrization. We showed that a specific model had a D-optimal design with five support points but, when the parameter values were changed, the same model had a D-optimal design with ten support points.

When designs for the same response but with different linear predictors were compared, it was not intuitive to find that designs for a linear predictor with more terms did not always require more support points than designs for linear predictors with fewer terms.

Some designs for equivalent models were compared. (Equivalent models have different linear predictors that reduce to the same linear predictor when the parameter values are taken into account.) The equivalent models did not produce the same D-optimal designs. However, the designs did share a number of similar features: the location of the design points, similar points on the boundary of the design space, and the behaviour of the derivative function at these points.

As mentioned throughout this chapter, the main drawback of D-optimal designs is that they are locally optimal and thus dependent on the values of the parameter estimates. It is difficult to determine a standard design template for
a particular multinomial response model because the D-optimal design can vary considerably based on the values of the model parameters. The optimal designs can vary in the number of support points, their locations and their behaviour with respect to the points on the boundaries of the design space.
Chapter 5

IMSE-Optimal Design

The IMSE-optimal approach to the design of experiments is based on minimizing the integrated mean squared error (IMSE) of the estimated parameters over the design space. In this case, the parameters are the probabilities of the multinomial distribution. The mean squared error (MSE) is a statistical measure that quantifies the difference between the estimator and its true value.

Other design methods in the literature are based on minimizing the integrated MSE of the estimated probability, but these are not as popular as those based on the information matrix. The Quantile Dispersion method, discussed in Section 3.8, is one method based on the minimization of the mean square error of prediction over the design region.

The choice of design space is shown to be an important factor in an IMSE-optimal design. The design space is the region or range of covariate values considered when searching for the optimal design. We show that, when we integrate the MSE function over the design space, different optimal designs can be obtained depending upon the shape of the MSE curve over the regions being considered.
Ideally, the design space is determined by the constraints imposed on the treatments by the experimenter. However, we demonstrate a new technique, based on the limits of the underlying probabilities, to quantify the design space if no other information is available.

The estimated probability in the MSE calculation is based on maximum likelihood methods, but we incorporate a penalty function developed by Firth (1993). The resulting penalized estimator, originally developed to reduce the bias avoids the possibility of obtaining infinite estimates.

The MSE incorporates variance and bias components associated with the estimated response. In general, as the number of observations at each design point increases, the MSE declines and hence the IMSE-optimal design improves. The IMSE then is reduced and the optimal design points move closer together. We demonstrate that, for some parameter sets, as the number of observations increases, the relationship between variance and bias changes, and there is a trend reversal in the location of the IMSE-optimal design.

It is important to understand the differences between IMSE-optimal and D-optimal algorithms. We show that, for identical parameter sets, these two algorithms may not produce the same optimal design. The number of support points, and the number of observations at each support point, also play different roles in these two methods. In the IMSE-optimal design algorithm, both of these values must be fixed in advance; in D-optimal design, the number of support points and the weights are determined by the algorithm.

Most analysis is based on the scenario of two equally weighted support points, which also facilitates comparison of methods. We extend the methodology to unequally weighted design points and also to the case when a design has three support points.
5.1 Mean Squared Error

The mean squared error (MSE) quantifies the amount by which an estimator differs from the true value of the quantity being estimated. In our case, we shall measure the error of the maximum likelihood estimator (MLE) $\hat{\pi}$, relative to the true probability $\pi$. The optimal design, per the MSE criteria, is defined as the optimal support points for which the MSE of a function of the ML estimators is minimized.

To calculate the MSE, we first calculate the MLE at a particular pair of support points $(x_1, x_2)$. We start with an initial estimate of $\beta$ and use (2.8) to calculate $\hat{\beta}$ where $\hat{\eta}$ is calculated using (2.7). Next we calculate the estimated probability in the $j^{th}$ category, $\hat{\pi}_j$, at a particular $x$ using (2.6). Finally we calculate the MSE for the MLE estimate of the probability in the $j^{th}$ category to give

$$\text{MSE} (\hat{\pi}_j) = E [(\hat{\pi}_j - \pi_j)^2] = E (\hat{\pi}_j^2) - 2\pi_j E (\hat{\pi}_j) + \pi_j^2$$ (5.1)

where the true probability, $\pi_j$, is calculated using (1.13) and the expected value of the $m^{th}$ power of the estimated probability is calculated as

$$E (\hat{\pi}_j^m) = \sum_{y_{12}=0}^{n_1} \sum_{y_{13}=0}^{n_1-y_{12}} \sum_{y_{22}=0}^{n_2} \sum_{y_{23}=0}^{n_2-y_{22}} \hat{\pi}_j^m f (y_1; \pi_{1j}) f (y_2; \pi_{2j})$$ (5.2)

where $f (\cdot)$ is the multinomial probability at the $i^{th}$ support point and is based on (1.9).
5.2 Maximum Likelihood Estimators

The estimated linear predictor $\hat{\eta}_{ij} = \ln \left( \frac{y_{ij}}{y_i} \right)$ for $j = 2, 3$, given by (2.7), is used to estimate the parameters $\hat{\beta}$, which are used in the MLE $\hat{\pi}_j$. We calculate $E(\hat{\pi}_j)$ and use it to evaluate $\text{MSE}(\hat{\pi}_j)$. Note that, if any $Y_{ij} = 0$, then $\hat{\eta}_{ij}$ will be undefined because $E(\hat{\pi}_j)$ is calculated by summing over all the possible combinations of $Y_{ij}$. Consequently, $\hat{\beta}$, $\hat{\pi}_{ij}$ and the $\text{MSE}(\hat{\pi}_j)$ will also be undefined.

The probability of getting undefined maximum likelihood estimates is equivalent to the probability of getting a value of zero in any category. To calculate this probability for the three-category multinomial distribution, given in (1.9), we need to know the sample size $n$, an estimate of the unknown parameters $\beta$ and the support points $\xi = (x_1, x_2)$.

The number of observations across all categories is $n$, where $y_1 + y_2 + y_3 = n$. Consider all the different possible combinations of values that can occur in each category. This is calculated as

\[
\text{No. of Combinations} = \sum_{y_2=0}^{n} \sum_{y_3=0}^{n-y_2} 1
= \sum_{y_3=0}^{n} 1 + \sum_{y_3=0}^{n-1} 1 + \sum_{y_3=0}^{n-2} 1 + \ldots + \sum_{y_3=0}^{0} 1
= (n + 1) + (n) + (n - 1) + \ldots + (1)
= (n + 1) (n + 2) / 2.
\]

(5.3)

For example, when $n = 5$, we have 21 combinations. The data in Table 5.1 shows some of these combinations of responses $(y_1, y_2, y_3)$ and the corresponding probabilities of getting a zero value in a particular category. This example has $\beta = (0, 1, 0, 1)$ at $x = -3.9$. Note that, if there are no zeros in any category, then the probability of any $Y_j = 0$ is zero.
Table 5.1: Probability of infinite MLE for response variable combinations when $n = 5$.

For example, to calculate the probability given in the second row of Table 5.1, first use (1.10) to calculate $\eta_2 = 0 + 1 \times -3.9 = -3.9$ and $\eta_3 = -3.9$. Then use equation (1.13) to calculate $\pi_2 = e^{-3.9} / (1 + e^{-3.9} + e^{-3.9}) = 0.0195$ and $\pi_3 = 0.0195$, then $\pi_1 = 1 - \pi_2 - \pi_3 = 0.9610$. Finally calculate the multinomial probability using (1.9) to give

$$P(Y_1 = 4 \cap Y_2 = 0 \cap Y_3 = 1) = \frac{5!}{4!0!1!} \times 0.9610^4 \times 0.195^0 \times 0.0195^1 = 0.0830.$$

Table 5.2 lists some probabilities of obtaining infinite estimates for two vectors $\beta$ for $n = 5$ and $n = 30$. It shows that the probability of getting a zero in any response category $Y_j$ varies considerably between zero and one, based on the parameter estimates, the location of the design points and the value of $n$.

Table 5.2: Probability of infinite MLE.

Given that we shall have infinite MLE when any $Y_j = 0$, how can we modify the MLE so the probability of any response category is not undefined.
5.2.1 Approach for Infinite Estimates

Various authors have addressed the case of infinite estimators. In particular, the penalized approach of Firth (1993) will be discussed in Section 5.3.

The MLE for the three-category multinomial distribution are undefined in the following cases. Recall from (2.7), we have \( \hat{\eta}_j = \ln \left( \frac{Y_j}{Y_1} \right) \) for \( j = 2, 3 \) then the following scenarios are possible.

- If \( Y_1 = 0 \) then \( \hat{\eta}_j = \ln \left( \frac{Y_j}{0} \right) \to \infty \)
- If \( (Y_2 = 0 \text{ or } Y_3 = 0) \) then \( \hat{\eta}_j = \ln \left( \frac{0}{Y_j} \right) \to -\infty \)
- If \( Y_1 = 0 \text{ and } (Y_2 = 0 \text{ or } Y_3 = 0) \) then \( \hat{\eta}_j = \ln \left( \frac{0}{0} \right) \) undetermined

Our approach involves substituting a large number \( \pm B \) when the linear predictor approaches infinity \( \hat{\eta} \to \pm \infty \). The following algorithm is used.

- If
  
  \( Y_1 = 0 \) then let \( \hat{\eta}_2 = \hat{\eta}_3 = B \)

- Else if
  
  \( Y_2 = 0 \) then let \( \hat{\eta}_2 = -B \) else \( \hat{\eta}_2 = \ln \left( \frac{Y_2}{Y_1} \right) \)
  
  \( Y_3 = 0 \) then let \( \hat{\eta}_3 = -B \) else \( \hat{\eta}_3 = \ln \left( \frac{Y_3}{Y_1} \right) \)

\( B \) must be chosen so that \( e^B \) is finite.

Preliminary results for this method were not encouraging. For some parameter values, for example \( \beta = (0, 1, 0, 1) \), this algorithm was unstable with different values of \( B \) giving an appreciably different optimal design. Other parameter vectors, for example \( \beta = (0, 1, 0, -1) \) led to marginally more stable designs.
When $B \leq 35$, the optimal design converged to $(-0.987, 0.987)$, with non-convergence of the optimal design for $B > 35$.

These results did not warrant pursuing this method further.

### 5.3 Maximum Penalized Likelihood Estimators

The penalized log-likelihood method was developed by Firth (1993) to obtain estimators with reduced bias. Firth developed a correction, or penalty, to the equation which produces the maximum likelihood estimate (MLE), rather than directly correcting the MLE. A benefit of Firth’s method is that the “corrected” MLE cannot be infinite, unlike the original MLE.

The penalized log-likelihood is given by

$$\tilde{\ell}(\pi_{ij}; y_{ij}) = \ell(\pi_{ij}; y_{ij}) + \frac{1}{2} \ln |M|,$$

where $\ell$ is the log-likelihood, $|M|$ is the determinant of the information matrix and $\frac{1}{2} \ln |M|$ is the penalty function. For the case of two support points, $x_1$ and $x_2$, the log-likelihood is given by (2.1) as

$$\ell = \sum_{i=1}^{2} \{y_{i1} \ln \pi_{i1} + y_{i2} \ln \pi_{i2} + y_{i3} \ln \pi_{i3}$$

$$+ \ln (n_i!) - \ln (y_{i1}!y_{i2}!y_{i3}!)\}$$

and the natural logarithm of the determinant of Fisher’s information matrix, using (2.11), is

$$\log |M| = \sum_{i=1}^{2} \{\ln \pi_{i1} + \ln \pi_{i2} + \ln \pi_{i3} + 2 \ln n_i \} + 4 \ln (x_2 - x_1)$$

where $x_2 > x_1$. We can evaluate (5.4), to give the penalized log-likelihood for a
three-category multinomial response as

\[
\hat{\ell} = \sum_{i=1}^{2} \left\{ (y_{i1} + 0.5) \ln \pi_{i1} + (y_{i2} + 0.5) \ln \pi_{i2} + (y_{i3} + 0.5) \ln \pi_{i3} 
+ \ln (n_i) + \ln (n_i!) - \ln (y_{i1}! y_{i2}! y_{i3}!) \right\} + 2 \ln (x_2 - x_1).
\]

To find the maximum penalized likelihood estimators (MPLE), we use the process given in Section 2.3. We first differentiate the penalized log-likelihood function with respect to each \( \beta_i \)

\[
\frac{\partial \hat{\ell}}{\partial \beta_1} = 2 \sum_{i=1}^{2} \left\{ (y_{i2} + 0.5) - (n_i + 1.5) \pi_{i2} \right\}
\]

\[
\frac{\partial \hat{\ell}}{\partial \beta_2} = 2 \sum_{i=1}^{2} \left\{ x_i (y_{i2} + 0.5) - x_i (n_i + 1.5) \pi_{i2} \right\}
\]

\[
\frac{\partial \hat{\ell}}{\partial \beta_3} = 2 \sum_{i=1}^{2} \left\{ (y_{i3} + 0.5) - (n_i + 1.5) \pi_{i3} \right\}
\]

\[
\frac{\partial \hat{\ell}}{\partial \beta_4} = 2 \sum_{i=1}^{2} \left\{ x_i (y_{i3} + 0.5) - x_i (n_i + 1.5) \pi_{i3} \right\}
\]

using the fact that \( y_{i1} + y_{i2} + y_{i3} = n_i \). For the case of two support points, we equate the above four derivatives to zero to give

\[
\hat{\pi}_{i2} = \frac{y_{i2} + 0.5}{n_i + 1.5}
\]

\[
\hat{\pi}_{i3} = \frac{y_{i3} + 0.5}{n_i + 1.5}
\]  

(5.6)

for \( i = 1, 2 \). From (1.13) we can write the MPLE \( \hat{\pi}_{ij} \) in terms of the linear predictor \( \hat{\eta}_{ij} \) as

\[
\hat{\pi}_{i2} = \frac{e^{\hat{\eta}_{i2}}}{1 + e^{\hat{\eta}_{i2}} + e^{\hat{\eta}_{i3}}}
\]

\[
\hat{\pi}_{i3} = \frac{e^{\hat{\eta}_{i3}}}{1 + e^{\hat{\eta}_{i2}} + e^{\hat{\eta}_{i3}}}
\]  

(5.7)
Equating (5.6) and (5.7) we obtain

\[
\tilde{\eta}_2 = \ln \frac{y_i + 0.5}{y_i + 0.5} \\
\tilde{\eta}_3 = \ln \frac{y_i + 0.5}{y_i + 0.5}
\]  

From (1.10) we have

\[
\tilde{\eta}_2 = \tilde{\beta}_1 + \tilde{\beta}_2 x_i \\
\tilde{\eta}_3 = \tilde{\beta}_3 + \tilde{\beta}_4 x_i
\]

which, in the case of two support points, we can solve explicitly to give the MPLE of \( \beta \) as

\[
\tilde{\beta}_1 = \frac{x_1 \tilde{\eta}_{22} - x_2 \tilde{\eta}_{12}}{x_1 - x_2} \\
\tilde{\beta}_2 = \frac{\tilde{\eta}_{12} - \tilde{\eta}_{22}}{x_1 - x_2} \\
\tilde{\beta}_3 = \frac{x_1 \tilde{\eta}_{23} - x_2 \tilde{\eta}_{13}}{x_1 - x_2} \\
\tilde{\beta}_4 = \frac{\tilde{\eta}_{13} - \tilde{\eta}_{23}}{x_1 - x_2}
\]

where \( x_2 > x_1 \). This is often called an “exact” method as we can explicitly solve the four MLE equations (5.5) for the four beta parameters.

### 5.3.1 Exact Convergence of Estimators

Even though the MLE and MPLE estimators for \( \beta \) are not equal, they are asymptotically equal. Consider the expressions for these estimators in the case of two support points. The MLE of \( \beta \), found by maximizing the log-likelihood function, is given by (2.8) where \( \hat{\eta}_{i2} = \ln (y_{i2}/y_{i1}) \) and \( \hat{\eta}_{i3} = \ln (y_{i3}/y_{i1}) \) for \( i = 1, 2 \).
The MPLE of $\beta$, found by maximizing the penalized log-likelihood function, is given by (5.10), except for $i = 1, 2 \tilde{\eta}_2 = \ln \left[ \frac{(y_{i2} + 0.5)}{(y_{i1} + 0.5)} \right]$ and $\tilde{\eta}_3 = \ln \left[ \frac{(y_{i3} + 0.5)}{(y_{i1} + 0.5)} \right]$. As the number of design points increases

$$
\lim_{n_i \to \infty} \tilde{\eta}_2 = \lim_{n_i \to \infty} \ln \left( \frac{y_{i2}/n_i + 0.5/n_i}{y_{i1}/n_i + 0.5/n_i} \right) 
= \ln \left( \frac{y_{i2}}{y_{i1}} \right) 
= \hat{\eta}_2.
$$

Then $0.5/n_i \to 0$ as $n_i \to \infty$. Similarly as $n_i \to \infty$, then $\tilde{\eta}_3 \to \hat{\eta}_3$.

Thus, in the case of two support points, the MLE and MPLE of $\beta$ are asymptotically equivalent.

When we have more than two design points, explicit expressions for the MLE and MPLE estimators cannot be obtained. Solutions can only be obtained numerically. In Section 6.3.1, we shall demonstrate convergence numerically when there are more than two design points.

### 5.4 Minimization Criteria

#### 5.4.1 Design Space

The design space or region is the set of covariate values considered when searching for an optimal design. If the underlying experiment puts constraints on the covariates, such information should be used to determine the design space. For example, Zocchi & Atkinson (1999) discussed efficient design in the context of an experiment which exposed flies to a gamma radiation source. As the original experiment used doses of radiation in the range of 80 to 200 Gray units, the authors used the design range $[0, 200]$ in their analysis.
When estimating probabilities at particular values of $x$, we are interested in the set of $x$ values where the probabilities may vary. We have no interest in predicting a probability that is equal to zero or one. Nor should we continue trying to estimate a probability, at a particular value of $x$, after the probability has reached a limit. Thus, the design space, $\chi$, should only include the values of $x$ where the probabilities are varying and not equal to zero or one.

When we seek an optimal design, subject to the condition of minimizing the MSE of the estimated probability $\tilde{\pi}$, we incorporate knowledge of the behaviour of the true value of $\pi$, based on the true value of $\beta$. Using (1.13) we can write the probabilities for the second and third categories as $\pi_j = e^{\eta_j} / (1 + e^{\eta_2} + e^{\eta_3})$ and for the first category as $\pi_1 = 1 - \pi_2 - \pi_3$, where $\eta$ is a function of $x$ as given by (1.10).

This method is independent of the method used to find the optimal design and can thus be used with D- or IMSE-optimal design.

The design space, $\chi$, is determined as the set of $x$-values where the probabilities do not reach their limits, as measured with a selected precision. The lower bound of the design space $\chi$ is the value of $x$ where the absolute value of the difference between each probability and its limit is less than a selected precision, denoted by $\delta$. Similarly, we can also determine the upper bound of $\chi$. The following steps describe the method to find the design region.

First find the upper bound of the design space $\chi$.

1. Choose the precision $\delta$ required when measuring the difference between a probability and its limit.

2. Determine the value of $\beta$ and calculate the probabilities $\pi_j (x), j = 1, 2, 3$ as a function of $x$. 

133
3. As $x \to \infty$, find the limits to which each probability $\{\pi_1, \pi_2, \pi_3\}$ converges, and denote these limits by $\{t_1, t_2, t_3\}$.

4. Start at a particular value of $x$, which is near the value where the probabilities are nearing their limits:
   
   (a) calculate the absolute difference between each $\pi_j$ and its limit $t_j$,
   
   \[ d_j = |\pi_j - t_j|, \; j = 1, 2, 3; \text{ and} \]
   
   (b) find the maximum value of these differences, $\max \{d_1, d_2, d_3\}$,

5. If $\max \{d_1, d_2, d_3\} > \delta$ then increase the value of $x$ and repeat step 4.

6. Otherwise, set the value of $x$ as the upper bound of the design space.

A similar approach is used to find the lower bound of the design space.

To illustrate this procedure, let $\beta = (0, 1, 0, -1)^T$ and plot the three probabilities as functions of $x$, as shown in Figure 5.1.

![Figure 5.1: True probabilities for $\beta = (0, 1, 0, -1)^T$, over a range of $x$ values.](image-url)
For the lower bound, Figure 5.1 shows that as $x \to -\infty$, the probabilities $\pi_j$ for $j = 1, 2, 3$ converge to $\{0, 0, 1\}$. For the upper bound, as $x \to +\infty$ the probabilities converge to $\{0, 1, 0\}$.

Using a precision of $\delta = 10^{-3}$, calculate the probabilities at $x = -6.5$, given as $\pi_j = \{0.0015, 0, 0.9985\}$. The difference between the limits and probabilities as $x \to -\infty$ is $d_j = \{0.0015, 0, 0.0015\}$. Since max $d_j > 0.001$, decrease $x$ and recalculate the differences until the maximum difference is less than the precision $10^{-3}$. When $x = -7$, the differences are $d_j = \{0.0009, 0, 0.0009\}$ which satisfies max $d_j < 0.001$. Thus the lower bound of the design space is $x = -7$. Repeating this process for the upper bound gives the design space for $\beta = (0, 1, 0, -1)^T$ as $\chi = [-7, 7]$, for a precision of $\delta = 0.001$.

The degree of precision used determines the design space for a particular parameter set. Some examples appear in Table 5.3.

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$\delta = 10^{-2}$</th>
<th>$\delta = 10^{-3}$</th>
<th>$\delta = 10^{-5}$</th>
<th>$\delta = 10^{-8}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(0, 1, 0, 1)$</td>
<td>$[-5.3, 4]$</td>
<td>$[-7.6, 6.3]$</td>
<td>$[-12.3, 10.9]$</td>
<td>$[-19.2, 17.8]$</td>
</tr>
<tr>
<td>$(0, 1, 0, -1)$</td>
<td>$[-4.7, 4.7]$</td>
<td>$[-7, 7]$</td>
<td>$[-11.6, 11.6]$</td>
<td>$[-18.5, 18.5]$</td>
</tr>
<tr>
<td>$(1, 1, 1, 1)$</td>
<td>$[-6.3, 3]$</td>
<td>$[-8.6, 5.3]$</td>
<td>$[-13.3, 9.9]$</td>
<td>$[-19.5, 16.8]$</td>
</tr>
<tr>
<td>$(0.1, 1, 6, 2)$</td>
<td>$[-5.6, -0.4]$</td>
<td>$[-7.3, 1.3]$</td>
<td>$[-11.7, 5.7]$</td>
<td>$[-18.6, 12.6]$</td>
</tr>
<tr>
<td>$(0.1, 1, 6, -2)$</td>
<td>$[0.2, 4.6]$</td>
<td>$[-0.7, 6.9]$</td>
<td>$[-2.8, 11.5]$</td>
<td>$[-6.3, 18.4]$</td>
</tr>
<tr>
<td>$(1, 2, 3, 4)$</td>
<td>$[-2.9, 1.4]$</td>
<td>$[-4, 2.5]$</td>
<td>$[-6.3, 4.8]$</td>
<td>$[-9.8, 8.3]$</td>
</tr>
</tbody>
</table>

Table 5.3: Design spaces for various $\beta$ at four different precision levels $\delta$.

When the precision level is changed, the design space can vary considerably. The impact of a change in design space to the optimal design will be discussed in Section 5.6.

It is possible to use a certain precision level $\delta$ and information the experimenter
has about the design region to determine the design space. One suggested
approach is to use various $\delta$ values to calculate a sample of design spaces. The
design space that best agrees with the experimenter’s boundaries can be used
in the experiment.

5.4.2 Mean Squared Error

The mean squared error (MSE) measures how much an estimator differs from
the true value. In this case, we estimate the vector of probabilities, of the cate-
gories of the multinomial distribution, using the maximum penalized likelihood
estimator, MPLE, $\hat{\pi}$.

The examples that follow are based on the three-category multinomial distri-
bution modeled using a linear predictor with one covariate, given by (1.10).
The optimal design is denoted by $\xi^* = (x_1, x_2)$ with both support points being
equally weighted.

To calculate the MSE, first calculate the MPLE $\tilde{\beta}$ at a particular pair of support
points $(x_1, x_2)$. We first choose an initial estimate of $\beta$ and use (5.10) to calcu-
late the parameter estimates and use (5.8) to calculate estimates of the linear
predictor. Next we calculate the estimated probabilities, $\hat{\pi}_j$, at a particular $x$
using (5.7). Finally we calculate MSE for the MPLE estimate of the probability
in the $j^{th}$ category as

$$
\text{MSE}(\hat{\pi}_j) = E(\hat{\pi}_j^2) - 2\pi_j E(\hat{\pi}_j) + \pi_j^2 \quad (5.11)
$$

where the true probabilities, $\pi_j$, are calculated using (1.13) and the expected
value of the $m^{th}$ power of the estimated probability is calculated as

$$
E(\hat{\pi}_j^m) = \sum_{y_{12} = 0}^{n_1} \sum_{y_{13} = 0}^{n_1 - y_{12}} \sum_{y_{22} = 0}^{n_2} \sum_{y_{23} = 0}^{n_2 - y_{22}} \hat{\pi}_j^m f(y_1; \pi_{1j}) f(y_2; \pi_{2j}) \quad (5.12)
$$
where \( f(\cdot) \) is based on the multinomial probability (1.9).

The graphs in Figure 5.2 show the MSE of each estimated probability for two different \( \beta \) vectors, but calculated for the same support points \((-3, 4)\), across a range of \( x \) values. The graph on the left shows the MSE for \( \beta = (0, 1, 0, 1)^T \); note that there is an overlap of the two curves, since \( \tilde{\pi}_2 = \tilde{\pi}_3 \). The graph on the right shows the MSE calculated for \( \beta = (0, 1, 0, -1)^T \). Even though both graphs are based on the same support points, the MSE functions have very different shapes because they are based on different \( \beta \).

Figure 5.3: Comparing the MSE for \( \beta = (0, 1, 0, 1)^T \) over the design range \([-10, 10]\) for support points \((-6, 4)\) in the left graph and for support points \((-3, 4)\) in the right graph. In both graphs, \( \tilde{\pi}_2 = \tilde{\pi}_3 \).
The graphs in Figure 5.3 compare the MSE of each estimated probability for the same \( \beta = (0, 1, 0, 1)^T \), but calculated at two different pairs of support points. The graph on the left shows the MSE for the support points \((-6, 4)\), and the graph on the right shows the MSE for the support points \((-3, 4)\). Even though both graphs are based on different pairs of support points, the MSE functions have similar shapes because they are for the same parameter values \( \beta \).

A comparison of the curves in Figure 5.3 shows that the area contained under the MSE curves for the support points \((-3, 4)\) appears to be smaller than the area contained under the MSE curves for the support points \((-6, 4)\). This indicates the design \((-3, 4)\) is preferable to \((-6, 4)\) because it has a smaller MSE, for each \( \tilde{\pi}_1, \tilde{\pi}_2, \) and \( \tilde{\pi}_3 \).

### 5.4.3 Integrated Mean Squared Error

Continuing along this line of thought, we find the optimal design over the entire design space by calculating the area under each MSE graph. This is done by integrating the MSE curve over the design space to obtain the integrated mean square error (IMSE). Thus

\[
\text{IMSE}(\tilde{\pi}_j) = \int_{\chi} \text{MSE}(\tilde{\pi}_j) \, dx.
\] (5.13)

The design for which the IMSE is minimum is the optimal design under the IMSE criterion. This IMSE-optimal design is denoted by \( \xi^* \). It is not possible to calculate the integral of MSE analytically, so Simpson’s rule is used as an approximation. The number of intervals used is determined from the length of the integration range and we choose the integration steps to be approximately 0.2 units in length.

Table 5.4 shows the integrated MSE values for the two MSE charts in Figure
5.3. The numbers indicate that the support points \((-3, 4)\) have lower IMSE values compared to the support points \((-6, 4)\).

<table>
<thead>
<tr>
<th>Support Points</th>
<th>(\tilde{\pi}_1)</th>
<th>(\tilde{\pi}_2)</th>
<th>(\tilde{\pi}_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((-6, 4))</td>
<td>0.2276</td>
<td>0.1617</td>
<td>0.1617</td>
</tr>
<tr>
<td>((-3, 4))</td>
<td>0.1095</td>
<td>0.1466</td>
<td>0.1466</td>
</tr>
</tbody>
</table>

Table 5.4: Integrated mean squared error of each estimated probability for \(\beta = (0, 1, 0, 1)^T\) over the design space \([-7.6, 6.3]\] with \(n = 5\), calculated for two different sets of support points.

We have an IMSE value for the estimated probability for each category, corresponding to a particular design. We need to summarize these values so we can compare the IMSE values across a range of support points. Consider taking a weighted average of the IMSE, denoted by WIMSE, given by

\[
WIMSE = \sum_{j=1}^{k} W_j \text{IMSE} (\tilde{\pi}_j)
\]

where \(k\) is the number of response categories, \(W_j\) is the weight for the \(j^{th}\) category and \(0 \leq W_j \leq 1\), with \(\sum_{j=1}^{k} W_j = 1\). If information about a particular category is available, it should be incorporated into the weight for that category. Otherwise, as in the examples that follow in Table 5.4, we suggest equal weights per category.

The design points \((-6, 4)\) have WIMSE = 0.1837 and the design points \((-3, 4)\) have WIMSE = 0.1342. Thus, the support points at \((-3, 4)\).

5.5 Methods to Find Optimal Design

Now that we have a WIMSE value for each design, we need a systematic approach to search a range of support points and find the design with the lowest
WIMSE, which is referred to as the IMSE-optimal design, $\xi^*$. There are numerous methods to find the minimum. We shall look at: grid search, minimax and an optimization routine.

These methods were first implemented using R and later, to address performance issues, were also programmed in FORTRAN 77. First we discuss the implementation using R and later, in Section 5.10, we describe the FORTRAN program.

### 5.5.1 Grid Search

The grid search method, as discussed in Section 4.3.2 for D-optimal design, is adapted here for IMSE-optimal design.

1. Determine an estimate of the model parameters $\beta$ and of the number of observations at each support point, $n$.
2. Establish the design space $\chi$, as described in Section 5.4.1.
3. Select the number of intervals within the region. This will determine the row and column labels that constitute the design space, referred to as the design grid.
4. For each unique support point pair $(x_r, x_c)$ in the design grid, calculate the WIMSE using (5.11), (5.12), (5.13) and (5.14). As the support points are interchangeable, the values in the design grid are symmetric; hence only the upper-triangle values will be calculated.
5. Search the design grid for the smallest WIMSE value and note its row and column coordinates $(x_r, x_c)$. This is the IMSE-optimal design $\xi^*$.  

140
The grid search method finds the minimum WIMSE on the design grid. This will not necessarily be equal to the absolute minimum over the design space. However, this can be addressed by searching over a finer grid, increasing the probability that the absolute minimum lies closer to a grid point location.

For example:

1. Choose $\beta = (0, 1, 0, 1)^T$ and $n = 5$.

2. The design space is $\chi = [-7.6, 6.3]$.

Use 70 intervals within the design space. Part of the design grid is shown in Table 5.5. Note the symmetry in the grid.

<table>
<thead>
<tr>
<th></th>
<th>-7.6</th>
<th>-7.40</th>
<th>...</th>
<th>4.12</th>
<th>4.31</th>
<th>4.51</th>
<th>...</th>
<th>6.30</th>
</tr>
</thead>
<tbody>
<tr>
<td>-7.6</td>
<td>n/a</td>
<td>2.1079</td>
<td></td>
<td>0.4495</td>
<td>0.4370</td>
<td>0.4254</td>
<td>...</td>
<td>0.3596</td>
</tr>
<tr>
<td>-7.40</td>
<td>n/a</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-4.03</td>
<td></td>
<td>0.2023</td>
<td>0.2016</td>
<td>0.2020</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-3.83</td>
<td></td>
<td>0.2013</td>
<td>0.2012</td>
<td>0.2023</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-3.63</td>
<td></td>
<td>0.2020</td>
<td>0.2026</td>
<td>0.2043</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>n/a</td>
</tr>
</tbody>
</table>

Table 5.5: A design grid for $\beta = (0, 1, 0, 1)^T$ and $n = 5$ with WIMSE values calculated at selected grid locations over the design space $[-7.6, 6.3]$.

3. Calculate WIMSE for each (upper-triangle) support point pair. Selected values are displayed in Figure 5.5.

4. The minimum WIMSE value is 0.2012, located at position $(-3.83, 4.31)$ of the design grid. This point is the IMSE-optimal design.

The may not be the absolute minimum. To obtain more precision in the location of the minimum, we can perform another search in the region of the selected location, using a finer grid.
When we increase the number of intervals in the grid to 200, we find $WIMSE = 0.2011$ at $\xi^* = (-3.85, 4.22)$.

The computational price for refining the grid is approximately an eight-fold increase in processing time when the number of intervals in a range are doubled. For example, when this range was divided into 50, 100 and 200 intervals, the processing time was 30, 230 and 1804 seconds respectively.

![Contour plot](image)

Figure 5.4: Contour plot of weighted average integrated MSE for $\beta = (0, 1, 0, 1)^T$, shows optimal design at $(-3.80, 4.30)$ with $n = 5$ and 70 intervals over the design space $[-7.6, 6.3]$. The location of the optimal design is shown in the contour plot of the values of WIMSE in Figure 5.4. The optimum design is located within the innermost contour.

A second example, for $\beta = (0, 1, 0, -1)^T$, a grid with 70 intervals was placed over the design space $\chi = [-7, 7]$. The optimal design, $\xi^* = (-2.80, 2.80)$ with $WIMSE = 0.1601$, is shown in the contour plot displayed in Figure 5.5.
Figure 5.5: Contour plot of weighted average integrated MSE for $\beta = (0, 1, 0, -1)^T$, shows optimal design at $(-2.80, 2.80)$ with $n = 5$ and 70 intervals over the design space $[-7, 7]$.

When the number of intervals in the grid is increased to 200 units, then $\xi^* = (-2.73, 2.73)$ with WIMSE = 0.1599.

### 5.5.2 Minimax

The minimax method is a decision rule to find the minimum value of a set of possible maximums. In this application, we calculate individual matrices of IMSE values, where each matrix represents a multinomial category. Thus for the $j^{th}$ response category we have a matrix of IMSE ($\pi_j$) values. As in the grid search method, the rows and columns of these matrices represent the possible support points across the design space. We combine the matrices by finding the maximum value in each corresponding (row, column) position. The minimum value of this combination matrix is found. Their location is the IMSE-optimal design.
In the grid search method, we find a weighted average across all the (multinomial category) matrices. This allows us to apply different weights to each category. In the minimax method, each category is equally weighted. The minimax approach is as follows.

1. Determine estimates of the model parameters in $\beta$ and of the numbers of observations to be made at each support point, $n$.

2. Establish the design space $\chi$, as described in Section 5.4.1.

3. Select the number of intervals within the region. This will determine the design grid.

4. For the first category, calculate a matrix of IMSE values across all support points ($x_1, x_2$) in the design grid using (5.11), (5.12) and (5.13). As we specified $x_r < x_c$, this provides a value of IMSE in each upper-triangle element of the design grid. Repeat for each of the remaining $(k - 1)$ categories.

5. Find the maximum element in the corresponding (row, column) position across all the $k$ matrices and produce a new matrix of the same dimension containing the maximums.

6. Search this matrix of maximums and for the minimum value and note its row and column coordinates ($x_r^*, x_c^*$).

7. The IMSE-optimal design is the value of $\xi^* = (x_r^*, x_c^*)$ that minimizes the matrix of IMSE maximums over the design space $\chi$, for a particular $\beta$.

A simple example of the minimax algorithm is given in Table 5.6 for a design space $\chi = [-2, 2]$ and distance between support points of one unit. The top row of matrices contain IMSE values for the $j^{th}$ response category $j = 1, 2, \ldots k$. 
<table>
<thead>
<tr>
<th>IMSE for $j = 1$</th>
<th>IMSE for $j = 2$</th>
<th>IMSE $j = k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2$ 1 2 7</td>
<td>$-2$ 0 3 0</td>
<td>$-2$ 1 2 6</td>
</tr>
<tr>
<td>$-1$ 2 4 3</td>
<td>3 5 8 ...</td>
<td>2 6 7</td>
</tr>
<tr>
<td>0 4 9</td>
<td>2 3</td>
<td>9 5</td>
</tr>
<tr>
<td>1 8</td>
<td></td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Max IMSE values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-2$ 1 3 7</td>
</tr>
<tr>
<td>$-1$ 3 6 8</td>
</tr>
<tr>
<td>0 9 9</td>
</tr>
<tr>
<td>1 8</td>
</tr>
<tr>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6: Example of how minimax values are calculated.

The matrix on the bottom row contains the maximum of each $(row, column)$ element of these $k$ matrices. The minimum value in this matrix of maximums is at location $(-2, 0)$, which is this the optimal design $\xi^*$.

In the case of $\beta = (0, 1, 0, 1)^T$, $n = 5$, $\chi = [-7.6, 6.3]$ with 70 intervals, the IMSE-optimal design using minimax was $\xi^* = (-3.83, 4.31)$ with IMSE = 0.2016. When we increase the intervals to 200, then the design changes slightly to $\xi^* = (-3.99, 4.28)$ with IMSE = 0.2014.

For $\beta = (0, 1, 0, -1)^T$, $n = 5$, $\chi = [-7, 7]$ with 70 intervals, the IMSE-optimal design using minimax is $\xi^* = (-2.6, 2.6)$ with IMSE = 0.1640. When we increase the intervals to 200, then $\xi^* = (-2.66, 2.66)$ with IMSE = 0.1601.

### 5.5.3 Optimization Routine

The constrOptim function in R, introduced in Section 4.3.3, is adapted here for use in IMSE-optimal designs. In this case we want to minimize the WIMSE using constrOptim, as in the following steps.
1. Determine an estimate of the model parameters, $\beta$ and the number of observations at each support point, $n$.

2. Establish a design space $\chi$, as described in Section 5.4.1, and obtain expressions for the feasibility constraints.

3. Choose a valid starting value $(x_1, x_2)$ that lies within the feasible region.

4. Run constrOptim to minimize the WIMSE function.

5. ConstrOptim will return the IMSE-optimal design as "$\text{par}$" and the minimum WIMSE as "$\text{value}$".

Using constrOptim, the optimal design for $\beta = (0, 1, 0, 1)^T$, when $n = 5$, over the design space $\chi = [-7.6, 6.3]$ is $\xi^* = (-3.86, 4.23)$ with WIMSE = 0.2011. The optimal design for $\beta = (0, 1, 0, -1)^T$, $n = 5$ over the design space $\chi = [-7, 7]$ is $\xi^* = (-2.72, 2.72)$ with WIMSE = 0.1599. These results are consistent with those obtained using the grid search and minimax algorithms.

The “black-box” approach of constrOptim can be a disadvantage if the algorithm does not converge to the global minimum. In particular, constrOptim is sensitive to the starting value used. If the starting value is close to a local minimum, constrOptim will converge to the local minimum and not the global minimum.

When finding the optimal design using constrOptim for $\beta = (0, 1, 0, 1)^T$ over the region $\chi = [-7.6, 6.3]$, when $n = 25$ the algorithm returns different results depending upon the starting value used. When a starting value of $(-5, 5)$ is used, constrOptim returns the optimal design $\xi^* = (-5.12, 4.86)$ located at the global minimum WIMSE = 0.0602. This is located in the left trough in Figure 5.6. When a starting value of $(-2, 2)$ is used, constrOptim returns a design at $\xi = (-1.46, 4.12)$, which corresponds to a minimum WIMSE = 0.0729. This is a local minimum, located in the right trough in Figure 5.6.
Figure 5.6: Contour plot of design values for $\beta = (0, 1, 0, 1)^T$ and $n = 25$. The optimum design is at the global minimum $\xi^* = (-5.12, 4.86)$ but a secondary local minimum emerges at $\xi = (-1.46, 4.12)$.

A possible method to find a reliable starting value for constrOptim is to use the optimal design calculated by the grid search algorithm for a small value of $n$. Alternatively, the starting values can be randomly generated and the algorithm run a number of times.

### 5.5.4 Comparing Methods

The grid search method, minimax method and constrOptim optimization routine have been used to find the optimal design by minimizing the WIMSE over the design space. We now compare these methods.

The grid search and minimax algorithms are similar in that they are both based on searching matrices of IMSE values. As expected, they give consistent results.
Table 5.7 shows comparable optimal designs using all three methods for $\beta = (0, 1, 0, 1)^T$ over the design space $\chi = [-7.6, 6.3]$ for three selected sample sizes.

<table>
<thead>
<tr>
<th></th>
<th>ConstrOptim</th>
<th>Grid Search</th>
<th>Minimax</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WIMSE</td>
<td>Optimal Design</td>
<td>WIMSE</td>
</tr>
<tr>
<td>5</td>
<td>0.2011</td>
<td>$(-3.86, 4.23)$</td>
<td>0.2012</td>
</tr>
<tr>
<td>20</td>
<td>0.0701</td>
<td>$(-4.90, 4.71)$</td>
<td>0.0702</td>
</tr>
<tr>
<td>40</td>
<td>0.0452</td>
<td>$(-5.61, 5.25)$</td>
<td>0.0452</td>
</tr>
</tbody>
</table>

Table 5.7: Comparing optimal designs found using three methods for $\beta = (0, 1, 0, 1)^T$.

Figure 5.7 shows the optimal designs achieved by each method for $\beta = (0, 1, 0, 1)^T$.

All three methods give consistent results with grid search and constrOptim being more closely aligned.

Figure 5.7: Comparison of optimal designs for $\beta = (0, 1, 0, 1)^T$ over the design space $[-7.6, 6.3]$ as the number of observations at each support point increases. The left graph is for the smaller support point and the right graph is for the larger support point.

Figure 5.8 shows the IMSE corresponding to the optimal designs achieved by each method for $\beta = (0, 1, 0, 1)^T$. The IMSE is consistent over all three methods and converges towards zero as the number of observations increases.
Figure 5.8: Comparison of optimal IMSE for $\beta = (0, 1, 0, 1)^T$ over the design space $[-7.6, 6.3]$ as the number of observations at each support point increases.

Similar optimal design and minimum IMSE charts are shown for $\beta = (0, 1, 0, -1)^T$ in Figures 5.9 and 5.10 respectively. Again, all three methods are consistent in finding the optimal design. Moreover both the optimal design and IMSE converge as the sample size increases.

Figure 5.9: Comparison of optimal designs for $\beta = (0, 1, 0, -1)^T$ over the design space $[-7, 7]$ as the number of observations at each support point increases. The left graph is for the smaller support point and the right graph is for the larger support point.
Figure 5.10: Comparison of optimal IMSE for $\beta = (0, 1, 0, -1)^T$ over the design space $[-7, 7]$ as the number of observations at each support point increases.

The systematic nature of the grid search and minimax algorithms ensures that convergence is to the global minimum on the grid, not to a local minimum. As discrete points represent the design space, the minimum returned by a grid based search is not necessarily equal to the global minimum, but is rather the grid point location closest to the global minimum. This can be improved by searching over a more refined grid.

As mentioned previously, the reliability of constrOptim depends on the starting value used. If a reliable starting value is used, constrOptim is more accurate because it is free to search for a minimum over the design space not restricted by the fixed location of grid points. Also, the performance improvement makes constrOptim more appealing. Table 5.8 shows constrOptim is up to 12 times faster than the grid based methods, using 70 intervals. Note that as grid search and minimax were implemented in the same R program, they have the same processing times. Also, due to processing times increasing as the sample size
increases, simulations for large $n$ can only be run using constrOptim.

<table>
<thead>
<tr>
<th>Sample $n$</th>
<th>ConstrOptim Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 5$</td>
<td>12 times</td>
</tr>
<tr>
<td>$n = 20$</td>
<td>10.5 times</td>
</tr>
<tr>
<td>$n = 40$</td>
<td>11.1 times</td>
</tr>
<tr>
<td>$n = 50$</td>
<td>11.8 times</td>
</tr>
</tbody>
</table>

Table 5.8: Improvements in processing speed for constrOptim compared to grid based methods, using 70 intervals.

When $n = 40$, the processing time for the grid search method was approximately 3.9 days.

Although they are computationally intensive, the grid search and minimax algorithms also produce the WIMSE values used to create contour plots, as seen in Figures 5.4 and 5.5.

5.6 Varying the Design Space

We recall from Section 5.4.1 that the design space is determined as the range of $x$-values before the probabilities have reached their limits, as measured within some selected precision criteria. For example, the lower bound of the design space is the value of $x$ where the absolute value of the difference between each probability and its limit is less than a selected precision.

As the IMSE-optimal design requires each MSE ($\tilde{\pi}_j$) curve to be integrated over the design space, modifying the area of integration changes the IMSE ($\tilde{\pi}_j$) values and potentially results in a different optimal design. Thus the choice of design space is crucial in determining the optimal design under IMSE criteria.

Table 5.9 shows how varying the level of precision, $\delta$, gives different design spaces $\chi$. For example, for $\beta = (0, 1, 0, 1)^T$ and $n = 10$, if the precision is
changed from $10^{-2}$ to $10^{-8}$, the design spaces widen and the optimal design points move further apart.

<table>
<thead>
<tr>
<th>Precision</th>
<th>Design Space</th>
<th>WIMSE</th>
<th>Optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-2}$</td>
<td>$[-5.3, 4]$</td>
<td>0.0771</td>
<td>$(-4.01, 3.89)$</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>$[-7.6, 6.3]$</td>
<td>0.1172</td>
<td>$(-4.30, 4.38)$</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>$[-12.3, 10.9]$</td>
<td>0.2241</td>
<td>$(-5.09, 5.56)$</td>
</tr>
<tr>
<td>$10^{-8}$</td>
<td>$[-19.2, 17.8]$</td>
<td>0.4313</td>
<td>$(-6.31, 7.41)$</td>
</tr>
</tbody>
</table>

Table 5.9: Relationship between design space and resulting optimal design for $\beta = (0, 1, 0, 1)^T$ and $n = 10$.

The MSE chart in Figure 5.11 shows that the larger integration range adds an additional IMSE component which, over the range $[3.4, 10.9]$, is large compared to the IMSE component over the smaller design space $\chi = [-5, 3.4]$. This additional IMSE component has contributed to a larger IMSE, which results in a different optimal design over the larger design space $\chi = [-12.3, 10.9]$.

Figure 5.11: Comparison of the additional IMSE component for $\beta = (0, 1, 0, 1)^T$ and $n = 10$ when the design space is widened.

The same precision values were also used for a different $\beta = (0, 1, 0, -1)^T$, with
the results shown in Table 5.10.

<table>
<thead>
<tr>
<th>Precision</th>
<th>Design Space</th>
<th>WIMSE</th>
<th>Optimal Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-2}$</td>
<td>$[-4.7, 4.7]$</td>
<td>0.0715</td>
<td>$(-2.29, 2.29)$</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>$[-7, 7]$</td>
<td>0.0819</td>
<td>$(-2.43, 2.43)$</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>$[-11.6, 11.6]$</td>
<td>0.0896</td>
<td>$(-2.60, 2.60)$</td>
</tr>
<tr>
<td>$10^{-8}$</td>
<td>$[-18.5, 18.5]$</td>
<td>0.0938</td>
<td>$(-2.75, 2.75)$</td>
</tr>
</tbody>
</table>

Table 5.10: Relationship between design space and resulting optimal design for $\beta = (0, 1, 0, -1)^T$ and $n = 10$

The design space for $\delta = 10^{-2}$ is $\chi = [-4.7, 4.7]$, which gives an optimal design of $\xi^* = (-2.29, 2.29)$. If we change the precision to $\delta = 10^{-5}$, the width of the design space more than doubles to $\chi = [-11.6, 11.6]$, but the resulting optimal design is only marginally different at $\xi^* = (-2.6, 2.6)$. Figure 5.12 shows why this is so.

![Figure 5.12: Comparison of the additional IMSE component for $\beta = (0, 1, 0, -1)^T$ when the design space is widened.](image)

Figure 5.12 shows that the additional integration range, below -4.7 and above 4.7 does not add much additional IMSE component. As such, the optimal
designs over these two different design spaces are similar. This contrasts with
the result for $\boldsymbol{\beta} = (0, 1, 0, 1)^T$, shown in Figure 5.11.

We see that the shape of the MSE curve over the integration range, and hence
the relative size of the IMSE components, determines how much the optimal
design will change across the different design spaces. This method of deter-
mining the design space is a function of the probabilities and hence of the $\boldsymbol{\beta}$
parameters. For a particular parameter set, the level of precision selected is key
to establishing the design space.

The plots in Figure 5.13 show optimal designs for two design spaces, for increas-
ing values of $n$, the number of observations at each support point. Both graphs
compare optimal designs obtained for two different precision values. The graph
on the left, for $\boldsymbol{\beta} = (0, 1, 0, 1)^T$, shows optimal designs over two design spaces
$\chi = [-7.6, 6.3]$ and $\chi = [-19.2, 17.8]$, corresponding to precisions of $\delta = 10^{-3}$
and $\delta = 10^{-8}$ respectively. The graph on the right, for $\boldsymbol{\beta} = (0, 1, 0, -1)^T$, is also
over two design spaces $\chi = [-7, 7]$ and $\chi = [-18.5, 18.5]$, which correspond to
the same precisions, $\delta = 10^{-3}$ and $\delta = 10^{-8}$ respectively.

In both cases, the support points optimized over the smaller design spaces (open
dots in Figure 5.13), are located closer together than for the designs over the
larger design spaces (black dots in Figure 5.13).
Figure 5.13: Comparison of optimal designs over two design spaces as the number of observations at each support point increases. The left graph is for $\beta = (0, 1, 0, 1)^T$ and the right graph is for $\beta = (0, 1, 0, -1)^T$.

It is interesting to compare the optimal designs for two design spaces when $\beta = (0, 1, 0, 1)^T$ and $n = 10$. When $\delta = 10^{-2}$, the design space is $\chi_{10^{-2}} = [-5.3, 4]$ and the optimal design is $\xi_{10^{-2}} = (-4.01, 3.89)$. When $\delta = 10^{-8}$, the design space is $\chi_{10^{-8}} = [-19.2, 17.8]$ and the optimal design is $\xi_{10^{-8}} = (-6.31, 7.41)$.

The left diagram in Figure 5.14 shows the area under the average MSE curves for these two optimal designs over their corresponding design spaces. Now consider swapping the designs and spaces, as shown in the chart on the right in Figure 5.14. The graph denoted by a continuous line is the average MSE of the optimal design for the larger design space $\xi_{10^{-2}}$ plotted over the smaller design space $\chi_{10^{-2}}$. The graph denoted by a dashed line is the average MSE of the optimal design for the smaller design space $\xi_{10^{-8}}$ plotted over the larger design space $\chi_{10^{-8}}$. 
These plots highlight the difference in performance of the optimal designs over their corresponding design spaces, compared to an increased error when the optimal designs are evaluated over non-optimal design spaces.

5.7 Mean Square Error Components

The MSE can be broken down into variance and squared bias components:

\[
\text{MSE} (\tilde{\pi}) = E \left[ (\tilde{\pi} - \pi)^2 \right] = E \left( \tilde{\pi}^2 \right) - 2\pi E (\tilde{\pi}) + \pi^2 = E \left( \tilde{\pi}^2 \right) - [E (\tilde{\pi})]^2 + [E (\tilde{\pi}) - \pi]^2 = \text{Var} (\tilde{\pi}) + [\text{Bias} (\tilde{\pi})]^2.
\]

To emphasize that the probability \(\pi\) is a function of \(x\), we can also write \(\pi (x)\).

We first examine the behaviour of the variance and squared bias components of MSE for a particular value of \(n\). For \(n = 10\), the graph on the left of Figure 5.15
shows the variance and squared bias components of MSE $\tilde{\pi}_1(x)$. Towards the centre of the design space $\chi = [-7.6, 6.3]$, the squared bias is at its minimum and the variance is near its maximum. Both quantities tend to zero as $|x|$ increases. The graph on the right shows a similar behaviour for the squared bias component of MSE $\tilde{\pi}_2(x)$, but not for the variance. As $x$ increases, the variance increases.

Figure 5.15: Variance and squared bias components of MSE for IMSE-optimal design $(-4.3, 4.38)$ over the design space $[-7.6, 6.3]$ for $\beta = (0, 1, 0, 1)$ and $n = 10$. The left graph shows MSE ($\tilde{\pi}_1$) components and the right graph shows MSE ($\tilde{\pi}_2$) components.

Compare the same graphs for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$ in Figure 5.16. As $|x|$ increases, both the squared bias and variance components of MSE $\tilde{\pi}_1(x)$ and MSE $\tilde{\pi}_2(x)$ decrease towards zero.
Figure 5.16: Variance and squared bias components of MSE for optimal IMSE design \((-2.43, 2.43)\) over the design space \([-7, 7]\) for \(\mathbf{\beta} = (0, 1, 0, -1)^T\) and \(n = 10\). The left graph shows MSE (\(\tilde{\pi}_1\)) components and the right graph shows MSE (\(\tilde{\pi}_2\)) components.

Now consider the behaviour of the variance and bias components when \(n\), the number of observations at each support point, increases. Table 5.11 gives these MSE components for \(\mathbf{\beta} = (0, 1, 0, 1)^T\) over the design space \([-7.6, 6.3]\) for increasing values of \(n\).

| No. of Observations | Weighted Average Integrated MSE | Variance | Bias^2 | |Bias|
|--------------------|---------------------------------|----------|--------|-------------|
| 5                  | 0.2011                          | 0.1201   | 0.0810 | 0.2683      |
| 10                 | 0.1172                          | 0.0772   | 0.0400 | 0.1886      |
| 40                 | 0.0452                          | 0.0300   | 0.0152 | 0.1163      |
| 50                 | 0.0363                          | 0.0324   | 0.0039 | 0.0590      |

Table 5.11: Comparison of variance and bias components of MSE for \(\mathbf{\beta} = (0, 1, 0, 1)^T\) over the design space \(\chi = [-7.6, 6.3]\).

As expected, the mean squared error decreases as the sample size increases, as displayed in Figure 5.17. It can also be shown that, over the range \(n \in [5, 40]\), the ratio of squared bias to variance remains fairly constant, between 60 – 66%. The behavior of the MSE components when \(n > 40\) has changed. This warrants special discussion in Section 5.8.
Compare these results to the values in Table 5.12 for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$ for various values of $n$.

| Sample size | Weighted Average Integrated Size | MSE | Variance | $\text{Bias}^2$ | $|\text{Bias}|$ |
|-------------|-----------------------------------|-----|----------|----------------|-----------|
| 5           | 0.1599                            | 0.0581 | 0.1017  | 0.3170        |           |
| 10          | 0.0819                            | 0.0434 | 0.0385  | 0.1962        |           |
| 40          | 0.0257                            | 0.0246 | 0.0012  | 0.0340        |           |
| 50          | 0.0204                            | 0.0197 | 0.0007  | 0.0271        |           |

Table 5.12: Comparison of variance and bias components of MSE for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$.

As expected, the mean squared error decreases as the sample size increases, as illustrated in Figure 5.18.
However, while the MSE is decreasing, the squared bias component is decreasing at a faster rate than the variance component. When $n = 5$, the variance represents about 36% of the MSE but, as the number of observations increases to $n = 50$, the variance is 97% of the MSE.

### 5.8 Design Location Change

As the number of observations at each support point increases from $n = 5$ to 40, Figure 5.7 shows that the optimal design for $\beta = (0, 1, 0, -1)^T$ converges towards $\xi^* = (-5.61, 5.31)$. An interesting development was illustrated in Figure 5.6, which is the contour plot for $n = 25$. The optimal design, where IMSE = 0.0602, is an absolute minimum and is located at $\xi^* = (-5.12, 4.86)$. Another contour at IMSE = 0.08 shows the emergence of a secondary minimum at $\xi = (-1.46, 4.12)$. 

![MSE Breakdown](image)

Figure 5.18: Variance and squared bias components of MSE at optimal design for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$. 

![Graph of MSE Breakdown](image)
When \( n = 42 \), the design located at the secondary minimum at \( n = 25 \) becomes the global minimum. The optimal design for \( n = 42 \) is now located at \( \xi^* = (-1.25, 4.31) \) with IMSE = 0.0433. This is illustrated by the contour to the right in Figure 5.19. What was previously the global optimum at \( n = 25 \) is now a secondary minimum for \( n = 42 \), located at \( \xi = (-5.61, 5.31) \) with IMSE = 0.0440.

![Contour plot of optimal design](image)

Figure 5.19: Contour plot of optimal design for \( \beta = (0, 1, 0, 1)^T \) and \( n = 42 \), which shows the optimum design at the global minimum \( \xi^* = (-1.25, 4.31) \) and a secondary minimum at \( \xi = (-5.61, 5.31) \).

This optimal design trend change is illustrated across a range of values of \( n \) in Figure 5.20. The location of the optimal design changes to \( \xi^* = (-1.25, 4.31) \) with IMSE = 0.0433 when \( n = 42 \) using the grid search method. The minimax method does not register the change in the optimal design until \( n = 46 \) when the optimal design shifts to \( \xi^* = (-1.205, 4.91) \). The constrOptim algorithm, given an appropriate starting value, agrees with the grid search method.
Figure 5.20: Change in the location of the optimal design for $\beta = (0, 1, 0, 1)^T$ over the design space $[-7.6, 6.3]$ as the number of observations at each support point increases.

This change in optimal design location was discussed in Russell et al. (2009) for a binomial model with a logit link. The authors noted a trend reversal in the optimal design as the sample size increases and concluded that it is caused because the IMSE-optimal design has almost zero integrated squared bias.

In the example for $\beta = (0, 1, 0, 1)^T$, when $5 \leq n \leq 40$, the proportion of MSE of the squared bias ranges from 35-40% and the optimal design is located at approximately $(-6, 5)$. When $n \geq 42$, the optimal design trend shifts to $(-1, 4)$ and the squared bias component reduces to 10% of MSE. These results are consistent with the binomial model in Russell et al. (2009).
This behaviour is also evident in Figure 5.21, which shows a reduction in the squared bias component of MSE where the optimal design trend shifts.

### 5.9 MSE and IMSE-Optimal Design

It is interesting to see if there is a relationship between the IMSE-optimal design points and the MSE. Recall the IMSE-optimal design is found where the integrated MSE is minimized. So, intuitively, we might expect that the MSE is minimized at or near the location of the optimal design points.

We consider a three-category multinomial response and a linear predictor with one covariate. We want to find the IMSE-optimal two point design, and we consider designs with equally weighted support points.
For $\beta = (1, 2, 3, 4)^T$, $n = 2$ and $\chi = [-2.9, 1.4]$, the IMSE-optimal design is $\xi^* = (-1.859, 0.359)$ with WIMSE = 0.1287. Figure 5.22 shows the MSE curves for this design, plotted over the design space.

Figure 5.22 does not show the MSE being minimized near the IMSE-optimal design points. In fact, if we average the MSE curves, as shown in Figure 5.23, the MSE is actually maximized near the location of the IMSE-optimal design points.

When the same graph was generated for different values of $n$ across different design spaces, the same pattern emerges; the MSE is maximized near the IMSE-optimal design points.
Figure 5.23: IMSE-optimal design for $\beta = (1, 2, 3, 4)^T$ over the design space $\chi = [-2.9, 1.4]$ for $n = 2$, showing the average MSE.

However, this is not the case for other parameter sets. For example, consider $\beta = (0, 1, 0, -1)^T$, $n = 10$ and $\chi = [-7, 7]$. The IMSE-optimal design is $\xi^* = (-2.429, 2.429)$ with WIMSE = 0.0819. Figure 5.24 shows the MSE curves for this design, and the average MSE, plotted over the design space.

Figure 5.24: IMSE-optimal design for $\beta = (0, 1, 0, -1)^T$ over the design space $\chi = [-7, 7]$ for $n = 10$ and corresponding MSE curves. The left graph shows the MSE of individual probabilities and the right graph shows the average MSE.
In this case, we do not see the same behaviour noted in Figures 5.22 and 5.23. In this example, there is no evidence of a relationship between the optimal design and MSE. Further simulations over different parameter sets, design spaces and number of observations also failed to reveal a relationship between these quantities.

These results are consistent with those of Russell et al. (2009). They examined a binomial response with linear predictor $\eta = \beta_0 + \beta_1 x$ and parameter values $\beta_0 = 0$ and $\beta_1 = 1$, which had an asymptotic, equally weighted, D-optimal design at $(-1.5434, 1.5434)$. The authors showed that for sample sizes $n_1 = n_2 = 4, 5, 10$ and 20, the MSE was not minimized at the support points. For very small or large values of $x$, the authors showed that $\pi$ approached 0 or 1, respectively.

### 5.10 Three Support Points

To extend IMSE-optimal design to three support points, we follow the methodology described for two support points. The IMSE-optimal design is found by maximizing the penalized log-likelihood, (5.4), which in the case of three support points is given by

$$\tilde{\ell} = \sum_{i=1}^{3} \{y_{i1} \ln \pi_{i1} + y_{i2} \ln \pi_{i2} + y_{i3} \ln \pi_{i3} + \ln (n_i!) \} $$

$$- \ln (y_{i1}!y_{i2}!y_{i3}!) \} + \frac{1}{2} \ln |M|$$

where $M$ is the information matrix with dimension $(4 \times 4)$, given by (2.11).

The maximum penalized likelihood estimator (MPLE) for three support points is found by differentiating the penalized log-likelihood with respect to each $\beta$
value, using (2.4) to give
\[
\frac{\partial \tilde{\ell}}{\partial \beta_1} = \sum_{i=1}^{3} (y_{i2} - n_i \pi_{i2}) + \frac{1}{2} \frac{\partial |M|}{\partial \beta_1} / |M|,
\]
\[
\frac{\partial \tilde{\ell}}{\partial \beta_2} = \sum_{i=1}^{3} (y_{i2} - n_i \pi_{i2}) x_i + \frac{1}{2} \frac{\partial |M|}{\partial \beta_2} / |M|,
\]
\[
\frac{\partial \tilde{\ell}}{\partial \beta_3} = \sum_{i=1}^{3} (y_{i3} - n_i \pi_{i3}) + \frac{1}{2} \frac{\partial |M|}{\partial \beta_3} / |M|,
\]
\[
\frac{\partial \tilde{\ell}}{\partial \beta_4} = \sum_{i=1}^{3} (y_{i3} - n_i \pi_{i3}) x_i + \frac{1}{2} \frac{\partial |M|}{\partial \beta_4} / |M|.
\]

Using Maplesoft (2005), expressions were obtained for (5.15) and (5.16) but are not provided here due to the large number of terms. To find the MLE of \( \beta \) we maximize the penalized log-likelihood (5.15). To solve the four derivative equations in (5.16) equal to zero we must use an iterative process because no closed form of the solution exists. Once we have obtained \( \hat{\beta} \), we use this to calculate the estimated probabilities, given by (5.7). We can then calculate the MSE of the estimated probabilities, given by (5.11), as
\[
MSE(\hat{\pi}_j) = E(\hat{\pi}_j^2) - 2\pi_j E(\hat{\pi}_j) + \pi_j^2
\]
where
\[
E(\hat{\pi}_j^m) = \sum_{y_{12}=0}^{n_1} \sum_{y_{13}=0}^{n_1} \sum_{y_{22}=0}^{n_2} \sum_{y_{23}=0}^{n_2} \sum_{y_{32}=0}^{n_3} \sum_{y_{33}=0}^{n_3} \hat{\pi}_j^m f_j
\]
and \( f_j = f(y_1; \pi_{1j}) f(y_2; \pi_{2j}) f(y_3; \pi_{3j}) \) is a product of the multinomial probabilities, based on (1.9).

The IMSE, (5.13), is calculated by integrating the MSE over the range of covariate values, and the IMSE-optimal design is found by minimizing the weighted average IMSE (WIMSE) of the estimated probabilities for each multinomial category, given by (5.14).

The flow chart in Figure 5.25 illustrates the processing logic for an IMSE-optimal design with three support points. Two FORTRAN 77 IMSL subroutines were
used for function optimization. The subroutine DUMINF, which maximizes
a function using a quasi-Newton method and a finite-difference gradient, was
used to minimize the penalized log-likelihood. The subroutine DLCONF, which
minimizes a function subject to linear equality/inequality constraints, was used
to minimize the WIMSE.

To ensure convergence of both DUMINF and DLCONF, the starting values for
these subroutines should be randomly generated and the search repeated several
times.

Using this program, the equally weighted IMSE-optimal three point design over
\( \chi = [-3, 3] \) with \( N = 6 \) observations is found as \( \xi^* = (-3, -2.673, 2.443) \) with WIMSE = 0.1274.

This method can be extended to more than three support points but the computations intensify with the number of support points. The MSE calculations for each estimated multinomial probability requires taking expected values, which involve summations over all possible values of responses in each multinomial category at each support point.

When we have three multinomial categories, the number of possible combinations of responses in each category is given by \((n + 1)(n + 2)/2\), where \(n\) is the number of observations at each support point, see (5.3).

When we have two support points, we need to consider each possible set of \((Y_{11}, Y_{12}, Y_{13})\) values at the 1\(^{st}\) support point with each possible set of \((Y_{21}, Y_{22}, Y_{23})\) values at the 2\(^{nd}\) support point. This can be thought of as a double loop, one for each support point, and would involve a total of \(((n + 1)(n + 2)/2)^2\) iterations, as in (5.12).

When we have three support points, we need to consider each possible set of responses at the 1\(^{st}\), 2\(^{nd}\) and 3\(^{rd}\) support points. This can be thought of as a triple loop, which would involve a total of \(((n + 1)(n + 2)/2)^3\) iterations, as in (5.17).

Consequently a design for a three-category multinomial response, with \(s\) support points, would require expected value calculations involving \(((n + 1)(n + 2)/2)^s\) iterations for each design considered by the algorithm. Table 5.13 shows the total number of iterations for a two and three point design for various values of \(n\).
<table>
<thead>
<tr>
<th>$n$</th>
<th>$(n+1)(n+2)/2$</th>
<th>Total No. of Iterations</th>
<th>Factor Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>6</td>
<td>72</td>
<td>648</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>882</td>
<td>27,783</td>
</tr>
<tr>
<td>10</td>
<td>66</td>
<td>8,712</td>
<td>862,488</td>
</tr>
<tr>
<td>20</td>
<td>231</td>
<td>106,722</td>
<td>36,979,173</td>
</tr>
<tr>
<td>50</td>
<td>1,326</td>
<td>3,516,552</td>
<td>6,994,421,928</td>
</tr>
</tbody>
</table>

Table 5.13: Total number of iterations, in expected value calculations, for a two and three point design for various values of $n$.

For example, when $n = 10$, the expected value calculations would require 66 iterations for each two point design and 8,712 iterations for each three point design considered, which is a 99-fold increase. As $n$ increases, the number of iterations for a three point design increases dramatically.

### 5.11 Unequally Weighted Support Points

The IMSE-optimal designs analyzed so far have been based upon two equally weighted support points. The algorithm also caters for unequally weighted support points and requires the total number of observations to be specified. The algorithm must be run for each different weight combination and finds the IMSE-optimal design with the minimum WIMSE.

For example, see Figure 5.26. This shows all possible designs, and respective WIMSE values, for the parameter set $\beta = (0, 1, 0, 1)^T$ over $\chi = [-7.6, 6.3]$ when the total number of observations is $N = 6$. The design in the left graph in Figure 5.26 labeled “5,1” is the design at $(-4.437, 3.397)$ with weights $(5/6, 1/6)$. It has WIMSE = 0.362, as shown on the right graph.
Figure 5.26: IMSE-optimal designs for $\beta = (0, 1, 0, 1)^T$ over $\chi = [-7.6, 6.3]$ with different numbers of observations at each support point for $N = 6$. The left graph shows the design points and the right graph shows corresponding WIMSE values for each design.

The equally weighted design at $(-3.722, 4.247)$ is IMSE-optimal with the lowest WIMSE = 0.303.

As numbers of observations and support points increase, the IMSE-optimal algorithm becomes more computationally intensive as there are more design and observation weight combinations to consider.

### 5.12 IMSE- vs D-Optimal Design

The IMSE- and D-optimal design methods are based on different criteria. IMSE-optimality minimizes the mean squared error of the estimated probabilities whereas D-optimality minimizes the determinant of the covariance matrix of the parameter estimates.

When we compare the two-point equally weighted designs obtained using these techniques, key differences emerge. The most apparent is the role played by $n$, the number of observations at each support point.
As the value of $n$ increases, the IMSE-optimal design improves because the WIMSE reduces. For most parameter values, the support points tend to move closer together, as shown in the left graph of Figure 5.27.

![Graph showing change in location of optimal design](image)

Figure 5.27: Change in the location of the optimal design as the design space increases. The left graph is for $\beta = (0, 1, 0, -1)^T$ and the right graph is for $\beta = (0, 1, 0, 1)^T$.

However in other cases, as the value of $n$ increases, for example $\beta = (0, 1, 0, 1)^T$, there is a noticeable location shift in the IMSE-optimal design, as shown in the right portion of Figure 5.27.

In D-optimal design, $n$ is a constant multiplier in the information matrix, (2.10), and does not change the location of the maximum. Hence the D-optimal design points do not change for different sample sizes.

Table 5.14 compares designs achieved under IMSE- and D-optimality for two parameter sets. There are various IMSE-optimal designs, based on the value of $n$, but only one D-optimal design.
The results in Table 5.14 raise the possibility that the IMSE-optimal design converges to the D-optimal design as $n \to \infty$.

The IMSE-optimal approach uses the estimate of $\pi$, based on the MPLE $\beta$, (5.10), and the optimal design is located where the WIMSE is minimized. When the linear predictor has one covariate and the design has two support points, this method is called “exact” as we can explicitly solve the four MLE equations (5.5). However, in most cases, numerical methods must be used.

The D-optimal approach does not use the maximum likelihood estimate of $\beta$. The optimal support points are located where the determinant of the covariance matrix of the parameter estimates is minimized. Since the covariance matrix is the inverse of the information matrix $V_{\text{asym}} = M^{-1}$, we find the D-optimal design by maximizing $M$. This method is called “asymptotic” because it is based on the covariance matrix of the asymptotic distribution of $\hat{\beta}$.

The D-optimal design algorithm is faster to implement. The IMSE-optimal approach requires the calculation of MPLE estimates and the taking expected values of the estimated probabilities.

### Table 5.14: Comparing IMSE-optimal and D-optimal designs with two equally weighted support points.

<table>
<thead>
<tr>
<th>Sample</th>
<th>IMSE-Optimal</th>
<th>D-Optimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 5$</td>
<td>$\beta = (0, 1, 0, 1)^T$</td>
<td>$(-3.86, 4.23)$</td>
</tr>
<tr>
<td>$n = 25$</td>
<td>$(-5.12, 4.86)$</td>
<td>$(-2.05, 2.05)$</td>
</tr>
<tr>
<td>$n = 50$</td>
<td>$(-1.21, 4.37)$</td>
<td>$(-1.35, 1.35)$</td>
</tr>
<tr>
<td></td>
<td>$(-1.28, 3.02)$</td>
<td>$(-1.09, 1.09)$</td>
</tr>
</tbody>
</table>
5.13 Summary

The IMSE-optimal approach is based on minimizing the integrated mean squared error of the estimated probability over the design space. The particular application examined was based on modelling a three-category multinomial response using a linear predictor with one covariate. To determine the optimal design, we evaluated the MSE curve, for each estimated probability, for a particular design. The IMSE was then found by integrating each MSE curve over the range of covariate values. The integral cannot be resolved analytically so numerical methods were used. The WIMSE was obtained by averaging across the multinominal categories and the IMSE-optimal design was found by minimizing the WIMSE.

The role of the design space was demonstrated by showing how different designs were obtained for the same parameter values but for different design spaces. Where possible, the treatment constraints in the underlying experiment were used to establish the design space. Otherwise, the design space was determined based on the limits of the underlying probabilities.

The design space plays a crucial role because of the shape of the MSE curves. If a larger design space adds a relatively small IMSE component, compared to a smaller design space, the optimal design for the two design spaces will be comparable. However, when the larger design space adds a relatively large additional IMSE component, there can be a marked difference between the optimal designs. The general pattern is for the design points to move further apart as the design space widens.

Three different methods were used to obtain the optimal design: a grid search, a minimax search and an optimization algorithm. These achieved comparable results. Similar conclusions were drawn from the D-optimal analysis Chapter 4.
The grid search and minimax algorithms reliably converge to the optimal value in the defined search grid, but are computationally intensive. The recursive optimization routines are faster but, because some starting values can cause divergence, reliable results can be obtained only when the starting values are randomly generated and the algorithm is run several times.

Interesting behaviour was noted in the MSE variance and squared bias components. As the number of observations $n$ at each support point increased, there was a change in the ratio of these components. For certain parameter sets, when the proportion of the squared bias component remained fairly steady, the optimal design also remained fairly steady. However, for other parameter sets, when the proportion of squared bias changed and approached zero, there was a noticeable shift in the location of the optimal design.

The IMSE- and D-optimal designs are based on different minimization criteria and behave differently. The IMSE-optimal design depends on a given number of support points and the number of observations at each point. The D-optimal algorithm determines the number of support points, their locations and weights. In simulation exercises, the optimal designs found under both methods indicated that IMSE- and D-optimality may asymptotically convergence.
Chapter 6

Immunization Example

6.1 Introduction

Speybroeck et al. (2008) evaluate two immunization trials for the parasite Theileria parva which causes East Coast Fever (ECF), a cattle disease in East and Central Africa. The method used to control ECF is “infection and treatment”. The animal is exposed to a particular dose of the live parasite and then treated with an oxytetracycline formulation to allow the animal to develop an immune response.

The objective of the experiment is to find the optimal dose of the immunization agent to ensure the animal’s survival. The experimental treatments are the doses of the immunization agent. The response measures the probability that the animal has a particular type of reaction when exposed to the parasite, and is modeled by a three-category multinomial distribution.

We use the observed responses from the experiment to calculate parameter estimates with the aim of constructing locally D- and IMSE-optimal designs. The selection of an appropriate design range is shown to be important in finding
an optimal design. We adopt the linear predictor suggested by the authors and compare the resulting designs to those obtained when the predictor has an additional quadratic term.

Using the results from the original experiment, we can derive estimates of the model parameters and then find locally optimal designs. As the locally optimal design depends on the parameter estimates, we investigate whether the design based on parameter estimates from the original experiment changes significantly when small changes are made to the parameter values. We explore two scenarios. In one scenario we make slight changes to the experimental outcomes and recalculate the parameter values. In another scenario, we recalculate the parameter estimates by taking their standard deviations into account.

The D-efficiency, or relative efficiency, is used to compare the quality of two designs. We shall show that the D-optimal designs clearly outperform the original experimental design.

Using the data from the original experiment, we compare the IMSE-optimal and D-optimal designs. There is no expectation that these designs will be the same because they are based on different criteria: IMSE-optimality minimizes the mean squared error of the estimated probabilities and D-optimality minimizes the covariances of the parameter estimates for a specified model.

The ML and penalized ML parameter estimates are calculated from the original experimental data and the resulting IMSE-optimal designs are evaluated and compared to D-optimal designs.

6.2 Original Experiment

Speybroeck et al. (2008) looked at two trials in which cattle were immunized against the Theileria parva parasite that causes ECF. Their objective was to find
the optimal immunization dose that is both safe for the animal and maximizes the chance of a successful immunization.

One trial used a stabilate composed of three different strains of the Theileria parva parasite and the other trial used a single Theileria parva strain. This latter trial is called the Katete trial and will be the subject of analysis in this chapter.

The cattle are inoculated with 1 ml of the stabilate, composed of the parasite, in one of the following doses diluted serially on a logarithmic scale: 1/300, 1/100, 1/50, 1/20 and 1/10. The cattle are also administered a long-acting oxytetracycline treatment, called Terramycin (Pfizer Inc.), at 20 mg/kg body weight. This restrains the infection and allows the animal to develop a protective immune response.

Five weeks after the inoculation, the animals are administered a lethal 1 ml dose of the neat stabilate and their immunity response is measured. In previous trials, the responses were classified into two categories, successful or unsuccessful immunization, and the binomial logistic model was used. However, given that unsuccessfully immunized animals can be broken down into two distinct groups, the data is better analyzed using a model which incorporates multinomial probabilities with the following response categories.

1. Dysimmunization - death or severe reaction during the immunization process.

2. Successful immunization.

3. Immunization failure - death or severe reaction when animal is reinfected after being inoculated.

The model used by the authors is the baseline category logit where the outcomes
are classified into nominal groups and compared separately to the response category. The first category, “dysimmunization”, has been specified as the baseline category. The multi-category logit model is given by

\[ \eta_2 = \ln \left( \frac{\pi_2}{\pi_1} \right) = \beta_1 + \beta_2 x \]

\[ \eta_3 = \ln \left( \frac{\pi_3}{\pi_1} \right) = \beta_3 + \beta_4 x \] (6.1)

where \( \eta_2 \) and \( \eta_3 \) are the linear predictors and \( x = \ln(dose) \) represents the treatment level. The probabilities of dysimmunization, successful immunization and immunization failure are denoted by \( \pi_1, \pi_2 \) and \( \pi_3 \) respectively.

The experiment has five treatments and a total sample of \( N = 66 \) Friesian steers. Table 6.1 shows, for each treatment, the dose, the logarithm of dose and the sample size.

<table>
<thead>
<tr>
<th>Treatments</th>
<th>Dose</th>
<th>ln(dose)</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/300</td>
<td>-5.7</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>1/100</td>
<td>-4.61</td>
<td>28</td>
</tr>
<tr>
<td>3</td>
<td>1/50</td>
<td>-3.91</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>1/20</td>
<td>-3</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>1/10</td>
<td>-2.3</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 6.1: Original experimental dose, design points \( x_i = \ln(dose) \) and sample size \( n_i \) at \( i^{th} \) treatment level.

The results are given in Table 6.2 where \( Y_{ij} \) is defined as the number of observations in the \( j^{th} \) category \((j = 1, 2, 3)\) in the \( i^{th} \) treatment group \((i = 1, 2, 3, 4, 5)\).

Note that 52 observations, or 79%, fall into the second category, and only one observation falls into the third category.
### 6.3 Parameter Estimates

Locally optimal design techniques depend upon the choice of a particular value of the unknown parameters. We can use the data from the original experiment to estimate the parameters of the linear predictor (6.1). The required input values are: the treatments (log doses), $x_i$, the number of observations, $n_i$, and the observed values $Y_{ij}$ for the $i^{th}$ $i = 1, 2, \ldots, 5$ treatment and the $j^{th}$ category, $j = 1, 2, 3$.

To find the maximum likelihood estimator (MLE) of $\beta$, denoted by $\hat{\beta}$, we maximize the log-likelihood, (2.1), given by

$$
\ell = \sum_{i=1}^{5} \{y_{i1}\ln\pi_{i1} + y_{i2}\ln\pi_{i2} + y_{i3}\ln\pi_{i3} + \ln (n_i!) - \ln (y_{i1}!y_{i2}!y_{i3}!)}.
$$

To find the maximum penalized likelihood estimator (MPLE) of $\beta$, denoted by $\tilde{\beta}$, we maximize the penalized log-likelihood, (5.4), which is $\tilde{\ell} = \ell + \frac{1}{2}\ln |M|$, where $|M|$ is the determinant of the $(4 \times 4)$ information matrix, (2.11).

Each probability, $\pi_{i1}, \pi_{i2}$ and $\pi_{i3}$, is calculated using the linear predictor (1.13), $\pi_{i2} = e^{\eta_2}/(1 + e^{\eta_2} + e^{\eta_3})$ and $\pi_{i3} = e^{\eta_3}/(1 + e^{\eta_2} + e^{\eta_3})$. When possible Maplesoft (2005), rather than numerical methods, is used to find an explicit expression for the information matrix. Otherwise, since we are summing over

<table>
<thead>
<tr>
<th>Treatments</th>
<th>Response $Y_{i1}$</th>
<th>Response $Y_{i2}$</th>
<th>Response $Y_{i3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>23</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 6.2: Original experiment responses at the $i^{th}$ treatment level.
five treatment points, the terms are too numerous for an explicit expression for
the determinant of the information matrix.

The initial FORTRAN 77 program developed to calculate $\hat{\beta}$ and $\tilde{\beta}$ used the Vi-
sual Numerics, Inc. (1970-2007) subroutine called DUMINF, which minimizes
a function of $N$ variables using a quasi-Newton method and finite difference
gradient. The subroutine DUMING was not used as it requires the gradient
functions, which involve significant computation. These subroutines were com-
pared in test programs and no significant difference in accuracy or performance
was found between the “F” (no gradient) and the “G” (gradient) versions.

The input parameters used in the implementation of DUMINF are

- FCN - function to be minimized, in this case a subroutine which evaluates
  the log-likelihood or penalized log-likelihood.
- NP - number of parameters, four in this case.
- BGUESS - the initial guess of $\hat{\beta}$ or $\tilde{\beta}$.

Default values were used for the remaining parameters. After running numerous
simulations for different data sets, a problem was discovered with DUMINF. It
became apparent that the problem was caused by the values of BGUESS
generated by DUMINF as it attempted to converge. DUMINF recursively calls
itself, each time updating BGUESS, until the optimality condition is satisfied.
Numerical errors resulted when DUMINF used values of BGUESS that were too
large. In particular, the probability values (1.13) cannot be evaluated when the
input values are too large because the exponential term cannot be calculated.

To address this, the subroutine DUMINF was replaced by DBCONF, which
uses the same input parameters, as DUMINF, in addition to specified upper
and lower bounds of BGUESS.
Table 6.3 gives an example of output from both subroutines used in two different programs. The “Iteration” column indicates the iteration number, “U” indicates the output is from DUMINF and “B” indicates if the output is from DBCONF. The subroutines have the first eleven lines of output in common but they diverge on the next iteration.

Iteration “12 U” shows the values of BGUESS generated by DUMINF, which are quite different from the previous values. This causes calculation errors as the values become too large and prevent DUMINF from converging. In iteration “12 B” since each element of BGUESS is bounded within $[-8, 8]$, DBCONF is able to eventually converge to the optimal value denoted by “Final B”.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>BGUESS$_1$</th>
<th>BGUESS$_2$</th>
<th>BGUESS$_3$</th>
<th>BGUESS$_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 U B</td>
<td>-6.472500</td>
<td>-0.568570</td>
<td>-7.599860</td>
<td>0.402420</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 U B</td>
<td>-6.472500</td>
<td>-0.568570</td>
<td>-7.599860</td>
<td>0.402420</td>
</tr>
<tr>
<td>12 U</td>
<td>20.228510</td>
<td>-125.9190</td>
<td>-5.606934</td>
<td>-7.884284</td>
</tr>
<tr>
<td>12 B</td>
<td>-4.889525</td>
<td>-8.000000</td>
<td>-7.481709</td>
<td>-0.088859</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final B</td>
<td>-4.749622</td>
<td>-1.037129</td>
<td>-5.236616</td>
<td>-0.560766</td>
</tr>
</tbody>
</table>

Table 6.3: Comparing output values of two IMSL subroutines: DUMINF and DBCONF.

This example shows the processing differences between DUMINF and DBCONF. After identifying this issue with DUMINF, DBCONF is used for future processing and information on DUMINF is provided for historical purposes.

Once the problem with the values used for BGUESS was solved, it was then evident that DBCONF was dependent on the starting value used for BGUESS. For example, when BGUESS = $(-7.07, 1.54, 7.25, -1.10)$ the optimal solution was given by $\hat{\beta} = (-1.74, -0.8, 6.19, 8.0)$ but when BGUESS = $(-3.99, -1.20, 0.02, 0.89)$ the optimal solution was given by a different $\hat{\beta} = (-1.77, -0.81, -7.23, -1.15)$. 183
Another program was developed which randomly generated the BGUESS values from the U \((a, b)\) distribution, where \((a, b)\) are the bounds for each element of BGUESS. Once this program has been run a number of times, the optimal solution is found by searching for \(\hat{\beta}\) with the maximum log-likelihood or minimum negative log-likelihood.

Using this program and data from the original experiment, the MLE is given as

\[
\hat{\beta} = (-1.769, -0.811, -7.225, -1.1475)^T
\]

which gives the linear predictors as

\[
\hat{\eta}_2 = -1.769 - 0.811x
\]
\[
\hat{\eta}_3 = -7.225 - 1.1475x.
\]

Similarly, the MPLE is given by

\[
\tilde{\beta} = (-1.6438, -0.7643, -4.9601, -0.7734)^T
\]

and the linear predictors are

\[
\tilde{\eta}_2 = -1.6438 - 0.7643x
\]
\[
\tilde{\eta}_3 = -4.9601 - 0.7734x.
\]

Both estimators have negative slopes and negative intercept terms.

### 6.3.1 Numerical Convergence of Estimators

In Section 5.3.1 we showed that the MLE and MPLE estimators for \(\beta\) are asymptotically equal when there are two support points.
When there are more than two support points, explicit expressions for the MLE and MPLE estimators cannot be obtained. Solutions must then be obtained numerically. Also, we can only show asymptotic equivalence of the MLE and MPLE empirically.

First we calculate the MLE and MPLE, using the values of $Y_{ij}$, the number of observations in each treatment, for the original experiment. We then increase each $Y_{ij}$ by a common factor, so the same proportions in the three categories are maintained. For example, if we apply a factor of three to Table 6.2, the new observations are as in Table 6.4.

<table>
<thead>
<tr>
<th>Treatments $i$</th>
<th>Response $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>$3Y_{i1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>30</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>69</td>
<td>3</td>
<td>84</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>27</td>
<td>0</td>
<td>36</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>15</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>15</td>
<td>0</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 6.4: Original experiment responses at each $i^{th}$ treatment level from Table 6.2, multiplied by a factor of three.

The number of observations at each treatment are given by $n_i = \sum_{j=1}^{3} Y_{ij}$ and the total number of observations is $N = \sum_{i=1}^{5} n_i$. Table 6.5 shows the values of $n_i$ and $N$ for various multiplicative factors.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Treatment Group Sizes</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$n_1$</td>
<td>$n_2$</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>28</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>56</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.5: Treatment group sizes, and totals, when original sample sizes are multiplied by various factors.

For each multiplicative factor, we calculate a new set of $Y_{ij}$ values and then
calculate new values of MLE and MPLE. The MLE and MPLE estimators of 
\( \beta = (\beta_1, \beta_2, \beta_3, \beta_4)^T \) are plotted in Figure 6.1.

![Graphs showing comparison of MLE and MPLE for each parameter value of \( \tilde{\beta} \).](image)

Figure 6.1: Comparison of the MLE and MPLE estimators for each parameter value of \( \tilde{\beta} \).

All four diagrams in Figure 6.1 suggest that, as the sample size increases, the MLE and MPLE estimators of \( \beta \) are asymptotically equivalent.

### 6.4 D-Optimal Design

#### 6.4.1 Design Space

In Section 5.4.1, we discussed a method to determine the design space. This method was based on the variation of the probabilities over the design space.
This method is recommended when no other information is available. In the immunization case, however, the design space should be based on the allowable treatment levels. As the covariate is a proportion converted to a ln(dose) measure, positive values in the design space make no sense. The original experiment used the design space \( \chi_1 = [-5.7, -2.3] \) which is equivalent to a dose range of \((1/300, 1/10)\).

For analysis, we shall use some additional larger design spaces. Table 6.6 shows some doses. From this data, we choose two additional design spaces as: \( \chi_2 = [-5.99, -1.61] \) and \( \chi_3 = [-6.21, -0.69] \).

<table>
<thead>
<tr>
<th>Experimental Treatments</th>
<th>1/500</th>
<th>1/400</th>
<th>1/300</th>
<th>1/100</th>
<th>1/50</th>
<th>1/20</th>
<th>1/10</th>
<th>1/5</th>
<th>1/2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ln (dose)</td>
<td>-6.21</td>
<td>-5.99</td>
<td>-5.7</td>
<td>-4.61</td>
<td>-3.91</td>
<td>-3.0</td>
<td>-2.3</td>
<td>-1.61</td>
<td>-0.69</td>
</tr>
</tbody>
</table>

Table 6.6: Original experiment treatments and additional doses.

It is interesting to see the resulting design space if we had used the probability limit method. Figure 6.2 shows predicted probabilities for both the MLE and MPLE estimators of \( \beta \), graphed over the range of \( x \) values.

As \( x \to -6 \), the probabilities \((\pi_1, \pi_2, \pi_3) \to (0, 1, 0)\), which implies the design space should be \([-6, 0]\), which is consistent with the selected design spaces.
However, as \( x \) gets smaller, it is evident that the probabilities are not close to converging to their limits. In fact, the probabilities using \( \hat{\beta} \) don’t reach their lower limits (\( \pi_1, \pi_2, \pi_3 \) \( \to (0, 0, 1) \)) until \( x < -30 \), which is equivalent to a dose of \( 10^{-13} \), effectively a zero dose. For the \( \tilde{\beta} \) chart, the probabilities reach their limits at an even smaller value of \( x \).

In this case, both the method we proposed to determine the design space and that used in the original experiment are consistent.

### 6.4.2 Locally Optimal Designs

Once a design space has been established, D-optimal designs can be found using FORTRAN and the IMSL library (Visual Numerics, Inc., 1970-2007) subroutine, DLCONF. The initial design value, XGUESS, is randomly generated from the U \((a, b)\) distribution, where \((a, b)\) are the bounds of the design space. The algorithm is configured to allow up to ten design points with corresponding weights constrained to sum to one. The optimization region is confined to the design space.

We ran this program for a number of different randomly generated design values to obtain a D-optimal design for each design space. Table 6.7 shows the D-optimal designs, \( \xi^* \), for the MLE, \( \hat{\beta} \), which is defined as

\[
\xi^* \left( \hat{\beta} \right) = \begin{bmatrix}
  x_1 & x_2 & \ldots & x_n \\
  w_1 & w_2 & \ldots & w_n
\end{bmatrix}
\]

where \( x_k \) is the \( k^{th} \) design point and \( w_k \) is the corresponding weight.
Table 6.7: D-optimal designs using the MLE, $\hat{\beta}$, for three different design spaces.

There are some cases when an experiment might not be conducted at the optimal design locations. Consider the optimal design over the design space $[-6.21, -0.69]$ in Table 6.7. The experimenter may not want to administer a log dose of $1/3$, which corresponds to the support point at $-1.012$, because it may be too concentrated and injure the animal. Alternatively, the experimenter may not see the value of administering a log dose of $1/500$, which corresponds to the support point at $-6.21$, because it may be too diluted to have any affect on the disease.

Table 6.8 shows the D-optimal designs, $\xi^* (\tilde{\beta})$, using the MPLE, $\tilde{\beta}$.

Table 6.8: D-optimal designs using the MPLE, $\tilde{\beta}$, for three different design spaces.

When the design space is $\chi_1 = [-5.7, -2.3]$, as in the original experiment, both the MLE and MPLE estimators give the same D-optimal design, two equally weighted design points at the boundaries. The original experiment had

$$\xi_{orig} = \begin{bmatrix} -5.7 & -4.61 & -3.91 & -3 & -2.3 \\ 0.1515 & 0.4242 & 0.1818 & 0.0909 & 0.1515 \end{bmatrix}$$

which implies that the three middle treatment doses are redundant, if the model is correct.
When the design space is increased to $\chi_2 = [-5.99, -1.61]$, the MLE and MPLE designs differ. The MLE design now has three design points, one on each boundary, with a similar weight, and a point near the centre, with a smaller weight. The MPLE design for $\chi_2$ is still two equally weighted points on the boundaries.

When $\chi_3 = [-6.21, -0.69]$, both estimators of $\beta$ give three-point designs which share a common design point at the lower bound, -6.21. Neither design has a design point at the upper bound, -0.69. However, the MPLE design could be approximated by two design points, as the weight at the centre point is minimal.

As the design space increases the support points for both estimates of $\beta$ move further apart and, in some cases, a third point is added.

### 6.4.3 Standardized Variance

The standardized variance, or derivative function, confirms the D-optimality of each design. Figure 6.3 shows the standardized variance curves for the D-optimal design across the three design spaces. In the left graph we have used MLE $\hat{\beta} = (-1.7690, -0.8110, -7.225, -1.1475)^T$. In the right graph we have used MPLE $\tilde{\beta} = (-1.6438, -0.7643, -4.9601, -0.7734)^T$.

![Derivative function of the D-optimal design, for three design spaces. The left graph uses the MLE estimator of $\beta$ and the right graph uses the MPLE estimator of $\beta$.](image-url)
In the designs with three support points, the relative flatness of the standardized variance curve between the central and upper support points indicates that other near optimum designs points can be found in this region. This is true for both MLE and MPLE estimates of $\beta$. We can demonstrate this by considering the MLE estimate of $\beta$.

When $\hat{\beta} = (-1.7690, -0.8110, -7.2250, -1.1475)^T$ and the design space is $\chi^3 = [-6.21, 0.69]$, the D-optimal design is

$$\xi^*_3(\hat{\beta}) = \begin{bmatrix} -6.21 & -3.361 & -1.012 \\ 0.395 & 0.2822 & 0.3228 \end{bmatrix},$$

where the determinant of the information matrix is maximized at $|M| = 2.26 \times 10^{-5}$. If we alter the value of the middle design point ($-3.361$) by $\pm 0.1$, the determinant of the information matrix decreases by $0.14\%$. Similarly, if we alter the value of the upper design point ($-1.012$) by $\pm 0.1$, the determinant of the information matrix decreases by $0.19\%$. However, if we alter the value of the lower design point ($-6.21$) by $\pm 0.1$, the determinant of the information matrix decreases by a much larger value, $7.0\%$.

Thus, when the optimality measure is the determinant of the information matrix, slightly shifting the design point in the region where the standardized variance curve is flattest reduces the optimality measure by a relatively small amount. When we shift the design point located where the standardized variance curve is much steeper, a relatively greater reduction in the optimality measure occurs.
6.5 Variability of Probability and Design

We have shown how to calculate parameter estimates from the results of an experiment. Given a design space, we can then obtain a D-optimal design, and estimated probabilities, using these parameter estimates. We next discuss the changes in the D-optimal design, and estimated probabilities, when the parameter estimates are slightly varied.

We shall use two methods to vary the parameter estimates. The first method, described as an indirect method, involves making slight changes to the experimental responses and then recalculating the parameter estimates.

Recall the parameter estimates we calculated, using MLE and MPLE, based on data from the original experiment. Consider the possibility that the experiment yielded slightly different results. For example, the original responses for the dose at $x_1 = -5.7$ were $(Y_{11}, Y_{12}, Y_{13}) = (0, 10, 0)$ but suppose we had observed $(Y_{11}, Y_{12}, Y_{13}) = (1, 9, 0)$ instead. Or, for example, rather than observing $(Y_{11}, Y_{12}, Y_{13}) = (4, 23, 1)$ at the dose $x_2 = -4.61$, say the responses were $(Y_{11}, Y_{12}, Y_{13}) = (5, 22, 1)$. In both cases, the total number of responses across the three categories has not changed and the majority of observations remain in the second category.

The second method used to vary the parameter estimates is referred to as a direct method. In this case, we randomly generated new parameter estimates based on the mean and variance of the original estimates.

In this case we had the benefit of retrospectively viewing a study and determining the optimal design for an experiment that was already run. When this is not possible, parameter values or ranges could be determined by referring to the experimenter who has subject matter knowledge. In addition to this, available literature could also be consulted on similar studies.
If the necessary information is not available, then parameter values could be classified into categories, such as positive or negative slope or gradients, and random values from these scenarios could be simulated. A design could be constructed and then model parameters estimated. A sequential design approach could also be incorporated, which would enable information from the experiment to be used to recalculate updated parameter estimates.

It is also feasible to use probabilities, for each response category, rather than actual regression model parameter values. A subject matter expert could provide possible values or ranges of probabilities for each multinomial category, at a particular value of \( x \). Given this information, the values or ranges of the parameters could then be derived using the method described in Section 6.3.

### 6.5.1 Indirectly Varying the Parameter Estimates

We first look at how the D-optimal design changes when the original experimental responses differ only slightly.

The observed responses, \( Y_{ij} \), across five treatments, \( i \), and three categories, \( j \), are modified by adding or subtracting one or two units, but maintaining the original treatment totals \( n_i = Y_{i1} + Y_{i2} + Y_{i3} \). Table 6.9 shows the original experiment responses and ten different scenarios where random \( Y_{ij} \) values have been modified.
Table 6.9: Original and ten scenarios of altered treatment responses, $Y_{ij}$, all maintaining treatment replications from original experiment.

For each scenario, using the original treatment values, $x_i$, and totals, $n_i$, we can calculate new parameter estimates, using ML estimates, see Table 6.10. For the estimation algorithm to converge, we set $[-10, 10]$ as the bounds in the DUMINF subroutine for calculating $\hat{\beta}$. 

<table>
<thead>
<tr>
<th>Original</th>
<th>$Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>$n_i$</th>
<th>1</th>
<th>$Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>23</td>
<td>1</td>
<td>28</td>
<td>2</td>
<td>5</td>
<td>23</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>9</td>
<td>0</td>
<td>12</td>
<td>3</td>
<td>2</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>10</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>3 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>4 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 1 9 0</td>
<td>1 1 9 0</td>
<td>1 1 8 1</td>
<td>2 6 21 1</td>
<td>2 5 22 1</td>
<td>2 5 23 0</td>
<td>3 3 9 0</td>
<td>3 3 9 0</td>
<td>3 3 8 1</td>
</tr>
<tr>
<td>2 2 4 0</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>3 3 9 0</td>
<td>3 3 9 0</td>
<td>3 3 8 1</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
</tr>
<tr>
<td>3 5 4 1</td>
<td>5 5 4 1</td>
<td>5 5 4 1</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>5 5 5 0</td>
<td>5 5 5 0</td>
<td>5 4 5 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>6 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>7 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 10 0</td>
<td>1 1 8 1</td>
<td>1 1 8 1</td>
<td>2 6 22 0</td>
<td>2 4 23 1</td>
<td>2 5 23 0</td>
<td>3 3 8 1</td>
<td>3 3 9 0</td>
<td>3 3 8 0</td>
</tr>
<tr>
<td>2 3 8 1</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>3 3 9 0</td>
<td>3 3 9 0</td>
<td>3 3 8 1</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
</tr>
<tr>
<td>3 5 4 1</td>
<td>5 5 5 0</td>
<td>5 5 5 0</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>4 2 4 0</td>
<td>5 5 5 0</td>
<td>5 5 5 0</td>
<td>5 4 5 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>9 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
<th>10 $Y_{i1}$</th>
<th>$Y_{i2}$</th>
<th>$Y_{i3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 9 1</td>
<td>1 2 8 0</td>
<td>1 1 9 0</td>
<td>2 6 22 0</td>
<td>2 5 23 0</td>
<td>2 5 22 1</td>
<td>3 3 8 1</td>
<td>3 4 8 0</td>
<td>3 4 8 0</td>
</tr>
<tr>
<td>2 3 9 0</td>
<td>4 1 4 1</td>
<td>4 1 4 1</td>
<td>3 3 8 1</td>
<td>3 4 8 0</td>
<td>3 4 8 0</td>
<td>4 1 4 1</td>
<td>4 1 4 1</td>
<td>4 1 4 1</td>
</tr>
<tr>
<td>3 6 4 0</td>
<td>5 6 4 0</td>
<td>5 6 4 0</td>
<td>4 1 4 1</td>
<td>4 1 4 1</td>
<td>4 1 4 1</td>
<td>5 5 5 0</td>
<td>5 5 5 0</td>
<td>5 5 5 0</td>
</tr>
</tbody>
</table>
Table 6.10: $\hat{\beta}$ obtained from varying experiment outcomes as listed in Table 6.9

Figure 6.4 displays the $\hat{\beta}$ parameter values of Table 6.10.

Figure 6.4 shows that the spread of the parameter values in the linear predictor equation for the third category ($\hat{\beta}_3, \hat{\beta}_4$) is much larger than for the parameters in the linear predictor for the second category ($\hat{\beta}_1, \hat{\beta}_2$). Both categories are
measured against the first category, which is the baseline.

If we look at the original $Y_{ij}$ values in Table 6.9, we can see that the observed counts in the third category are very small, the majority being equal to zero. From a total of 66 observations in the original experiment, about 79% of the responses fall in the second category and only 1% in the third. The very low proportion in the third category means that any change to these values will have a greater effect than changes to the other categories. This results in a greater variability of the parameters for the third category.

Figure 6.4 shows the different values of $\hat{\beta}$ obtained when the original experiment responses are varied by small amounts. We can see the relationship between the relative proportions in the multinomial categories and the resulting spread of the parameter estimates. However, the parameter estimates are used to calculate the multinomial probabilities in each category. Rearranging (2.9) gives

$$
\begin{align*}
\pi_2 &= \frac{e^{\eta_2}}{1 + e^{\eta_2} + e^{\eta_3}} \\
\pi_3 &= \frac{e^{\eta_3}}{1 + e^{\eta_2} + e^{\eta_3}} \\
\pi_1 &= 1 - \pi_2 - \pi_3.
\end{align*}
$$

(6.2)

We examine the estimated probabilities over a range of $x$-values to compare the parameter estimates derived from the original responses to the responses modified under the proposed scenarios in Table 6.9. Figure 6.5 shows the probabilities for $\pi_1$, $\pi_2$ and $\pi_3$, in separate charts, calculated using (6.2) over the largest design space $\chi_3 = [-6.21, -0.69]$. 

196
The probabilities, across the ten scenarios, are fairly consistent with the probability curve for the original \( \hat{\beta} \). The probabilities from this small data set, for the \( Y_{ij} \) scenarios examined, do not appear to be overly sensitive to small changes to the observed values. We believe this will be true more generally.

Once the \( \hat{\beta} \) parameter estimates for the modified \( Y_{ij} \) scenarios have been obtained, the locally D-optimal designs can be found. In Figure 6.6, the D-optimal design for the original experiment is labeled “Original” and the designs “1” to “10” the modified response scenarios. The size of each solid dot, which represents a design point, is adjusted by its weight. So, all the dots in an equally weighted design would be the same size.

Consider the top left chart in Figure 6.6, for the design space \( \chi_1 = [-5.7, -2.3] \).
The $Y_{ij}$ scenario labeled “6” in Table 6.9 has a D-optimal design

$$
\xi_{S1-6} = \begin{bmatrix}
-5.7 & -4.24 & -2.3 \\
0.4236 & 0.2524 & 0.324
\end{bmatrix}
$$

Figure 6.6: D-optimal designs for the original experiment and each modified response scenario. The size of a support point is proportional to its weight. The top left graph is for $\chi_1 = [-5.7, -2.3]$, the top right graph is for $\chi_2 = [-5.99, -1.61]$ and the bottom graph is for $\chi_3 = [-6.21, -0.69]$.

The majority of designs for $\chi_1 = [-5.7, -2.3]$ have two equally weighted support points on, or very close to, the boundaries. Incorporating some uncertainty into the experimental results can also result in the need for a third support point. In Figure 6.6, these designs are “1” and “4”. Even though the weight of the additional support points is small, 0.0121 and 0.0072 respectively, it seems prudent to add an additional design point when uncertainty exists. Design “5”
stands out because a design point has moved from the boundary to the middle of the design space.

When the design space is wider, as in $\chi_2$ and $\chi_3$, there is less consistency in the D-optimal designs. The majority of designs now have three points, but the middle point does not always have a smaller weight. We expect the design to have two points, as the predictor equations are linear, however we note that, as the width of the design space, or uncertainty, increases, so does the number of design points. One feature all these designs have in common is a support point located on the upper boundary. This is true whether the designs have two or three support points.

It is not possible to say whether the D-optimal design should have two or three design points, as this depends on the values of the parameters and design space. When the responses from the original experiment are slightly varied, and the design space is $\chi_1$, the resulting D-optimal designs are very similar. This is not the case with the larger design spaces as there is increased variation in the D-optimal design.

The graphs in Figure 6.7 show the standardized variance functions corresponding to the original $\hat{\beta}$, which is the bold line, and for the $\hat{\beta}$ corresponding to the ten modified response scenarios.
Figure 6.7: Derivative function of the D-optimal designs, using $\hat{\beta}$, for original $Y_{ij}$ (bold curve) and ten $Y_{ij}$ scenarios. The top left graph is for $\chi_1 = [-5.7, -2.3]$, the top right graph is for $\chi_2 = [-5.99, -1.61]$ and the bottom graph is for $\chi_3 = [-6.21, -0.69]$.

When a design has three points, the flatter derivative function between the smaller support points indicates that other near-optimal design points can be located in this region.

Overall, when we consider the smallest design space, $\chi_1$, slight changes in observed values do not seem to radically alter the D-optimal design. When the width of the design space increases, it is no longer possible to draw this conclusion.
6.5.2 Directly Varying the Parameter Estimates

Another method to assess how the D-optimal design changes when the original experiment responses differ slightly is to directly vary the values of the original parameter estimates. We now examine how the D-optimal design changes when the parameter estimates are simulated from a normal distribution parametrized by the mean and variance of the original estimates.

The standard deviations are found by taking the square roots of the main diagonal elements of the covariance matrix. The covariance matrix of $\hat{\beta}$, which is the inverse of the information matrix, is given by (2.12).

New values of $\hat{\beta}$ are simulated from a normal distribution, with the mean vector $\hat{\beta}$ and standard deviation vector SD$\hat{\beta}$. Table 6.11 shows the standard deviation values of $\hat{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ for each design space.

<table>
<thead>
<tr>
<th>Design Space</th>
<th>SD$\hat{\beta}_1$</th>
<th>SD$\hat{\beta}_2$</th>
<th>SD$\hat{\beta}_3$</th>
<th>SD$\hat{\beta}_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[-5.7, -2.3]$</td>
<td>0.7872</td>
<td>0.2507</td>
<td>4.3122</td>
<td>0.8341</td>
</tr>
<tr>
<td>$[-5.99, -1.61]$</td>
<td>0.6299</td>
<td>0.2165</td>
<td>4.2241</td>
<td>0.7818</td>
</tr>
<tr>
<td>$[-6.21, -0.69]$</td>
<td>0.6121</td>
<td>0.2024</td>
<td>4.1314</td>
<td>0.7444</td>
</tr>
</tbody>
</table>

Table 6.11: Standard deviation of $\hat{\beta}$ for each design space.

From Table 6.11, it is apparent that the standard deviations of the parameters in the linear predictor equation for the third category, $(\beta_3, \beta_4)$, are much larger than for the parameters in the linear predictor for the second category, $(\beta_1, \beta_2)$. As discussed in Section 6.5.1, this is caused by considerable differences between the proportion of observations within each category.

Figure 6.8 shows the $\hat{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ parameter values obtained when simulating from $N\left(\hat{\beta}, \text{diag} \left(\text{Var}\hat{\beta}_1, \ldots, \text{Var}\hat{\beta}_4\right)\right)$. Ten simulations were run in each scenario.
As expected, we have a larger variation in the simulated values of $\hat{\beta}_3$ and $\hat{\beta}_4$.

We examine the probabilities calculated using the simulated $\hat{\beta}$ over a range of $x$-values over the widest design space $\chi_3 = [-6.21, -0.69]$. Figure 6.9 shows the probabilities for $\pi_1$, $\pi_2$ and $\pi_3$ calculated using (6.2). These probabilities are fairly consistent, over the range of $x$-values, but not to the same degree as the probability curves in Figure 6.5. The $\hat{\beta}$ for simulations number “8” and “9” give quite different values of $\pi_2$ and $\pi_3$ probabilities, over smaller values of $x$, compared to the other simulations.
Figure 6.9: Probability curves corresponding to the original and ten simulated $\hat{\beta}$ over $\chi_1 = [-5.7, -2.3]$. The top left graph is for $\pi_1$, the top right graph is for $\pi_2$ and the bottom graph is for $\pi_3$.

We cannot make firm conclusions with only ten simulations. From those data, when the values of $\hat{\beta}$ are varied by taking their standard deviations into account, the majority of probabilities, over a range of $x$-values, seem to be fairly consistent. The curves generally tend in the same direction, but there are some simulated $\hat{\beta}$ that produce probabilities quite different from the rest. These probabilities do not exhibit the same consistency as the curves in Figure 6.5, which were obtained by varying the experimental responses.

Once all the $\hat{\beta}$ have been simulated, the locally D-optimal designs can be determined. Figure 6.10 shows the D-optimal design for the original experiment, labeled “Original”, and the designs labeled “1” to “10” represent the D-optimal design for each simulated $\hat{\beta}$. 

203
Figure 6.10: D-optimal designs for the original experiment and each simulated \( \hat{\beta} \). The size of a support point is proportional to its weight. The top left graph is for \( \chi_1 = [-5.7, -2.3] \), the top right graph is for \( \chi_2 = [-5.99, -1.61] \) and the bottom graph is for \( \chi_3 = [-6.21, -0.69] \).

When we incorporate some uncertainty into the parameter values, most simulations show that a third support point is added to the design. This is intuitively appealing. One feature that all the designs share is a support point on the upper boundary. This is true whether the designs have two or three support points. This is somewhat less pronounced in the widest design space.
The standardized variances of Figure 6.11 confirm the optimality of the designs. No other consistent behavior is noted in these charts.

6.6 Relative Efficiency

The D-optimal design maximizes the determinant of the information matrix so by definition, it will be most D-efficient. The relative efficiency of a design, in this case the original experiment described in Section 6.2, compared to the
D-optimal design and given by

$$RE = \left\{ \frac{|M(\xi_{orig})|^\frac{1}{p}}{|M(\xi^*)|} \right\}$$

where $p$ is the number of parameters, $|M|$ is the determinant of the information matrix, evaluated at either $\hat{\beta}$ or $\tilde{\beta}$, $\xi_{orig}$ is the original design and $\xi^*$ is the D-optimal design.

We compare the relative efficiency of the original experiment to the D-optimal designs. Table 6.7 gives the D-optimal designs for $\hat{\beta}$ and Table 6.8 gives the D-optimal designs for $\tilde{\beta}$, across the three design spaces.

<table>
<thead>
<tr>
<th>Design Space</th>
<th>$\hat{\beta}$</th>
<th>$\tilde{\beta}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[-5.7, -2.3]$</td>
<td>72.6%</td>
<td>69.6%</td>
</tr>
<tr>
<td>$[-5.99, -1.61]$</td>
<td>66.2%</td>
<td>61.2%</td>
</tr>
<tr>
<td>$[-6.21, -0.69]$</td>
<td>63.2%</td>
<td>59.3%</td>
</tr>
</tbody>
</table>

Table 6.12: Relative efficiency of original experiment to D-optimal design, using $\hat{\beta}$ and $\tilde{\beta}$ parameter estimates, over three design spaces.

Table 6.12 shows that the original experiment is 72.6% as efficient as the D-optimal design, using $\hat{\beta}$, on $\chi_1 = [-5.7, -2.3]$. Similarly, if we use $\tilde{\beta}$, then the original experiment has an efficiency of 69.6%.

As the design space widens, the efficiency of the original experiment declines. The original experimental design does not change, but the D-optimal design is recalculated with each new design space.

When observations are made at a non-optimal design point, more observations are needed to obtain the same quality of information compared to when observations are made at an optimal design point. For example, the relative efficiency of the original experiment to the D-optimal design using MLE parameters is 72.6%. This means that if the original experiment used 100 observations, only
73 observations would be required in the D-optimal design to achieve estimates of equivalent quality.

### 6.7 IMSE-Optimal Design

The optimization algorithm for D-optimal designs does not require the number of support points to be specified. Rather, the algorithm allows for a sufficiently large number of support points, $x_i$, and corresponding weights, $w_i$, and assigns a zero weight to points not included in the design. The number of observations at the $i^{th}$ design point can then be determined by multiplying its weight by the total number of observations, $n_i = N \times w_i$.

This is in contrast to the optimization algorithm for IMSE-optimal designs, where the number of design points at each observation is fixed in advance. That is, we search for the IMSE-optimal design given a particular number of design points and the number of observations at each point. In this example we will concentrate on the IMSE-optimal design with two support points.

When we allow different weights in the IMSE algorithm, we must consider all the possible weight combinations for a given total number of observations. For example, Table 6.13 shows some possible weight combinations of number of observations for two support points, when the total number of observations is 66, as used in the original experiment.
### Table 6.13: Selected combinations of weights for an IMSE-optimal design with two support points and a total of $N = n_1 + n_2 = 66$ observations.

The optimal designs, using $\hat{\beta}$ over $\chi_1$, are then calculated for the selected weight combinations, and are listed in Table 6.14.

### Table 6.14: IMSE-optimal two point designs for selected combinations observations at each design point, using $\chi_1 = [-5.7, -2.3]$ and $\hat{\beta}$.

The data in Table 6.14 are easier to visualize in the left chart in Figure 6.12.
Figure 6.12: IMSE-optimal designs with different weights and WIMSE for $\hat{\beta}$. The left graph is for the design space $\chi_1 = [-5.7, -2.3]$ and the right graph is for $\chi_3 = [-6.21, -0.69]$.

The graphs in Figure 6.12 show pairs of design points $(x_1, x_2)$, with their scale on the left-vertical axis. The sizes of the points are weight adjusted. The WIMSE values are joined by solid lines, with their scale on the right-vertical axis.

In the left graph of Figure 6.12, for $\chi_1 = [-5.7, -2.3]$, the minimum WIMSE value is 0.009037, corresponding to an IMSE-optimal design of

$$\xi_1^* (\hat{\beta}) = \begin{bmatrix} -4.878 & -2.3 \\ 58\% & 42\% \end{bmatrix}.$$  

In the right graph of Figure 6.12, for $\chi_3 = [-6.21, -0.69]$, the minimum WIMSE value is 0.016681, corresponding to an IMSE-optimal design of

$$\xi_3^* (\hat{\beta}) = \begin{bmatrix} -3.861 & -0.69 \\ 59\% & 41\% \end{bmatrix}.$$  

Both IMSE-optimal designs for $\hat{\beta}$ have more weight at the smaller design point. Recall the corresponding D-optimal designs in Table 6.7. For $\chi_1$, the D-optimal design had two equally weighted support points on the boundaries. For $\chi_3$, the D-optimal design had three support points.
Both of the IMSE-optimal designs have a support point on the upper boundary. In the case of $\chi_3$, the other point is not very close to the lower boundary. We usually expect that, as the design space increases in width, the support points move closer to the boundaries, as was the case in the D-optimal designs for the same parameter values and design spaces.

Figure 6.13 shows the designs and corresponding WIMSE values when the parameter estimate is $\tilde{\beta}$. Note that these two graphs are similar to those in Figure 6.12.

![Image of Figure 6.13: IMSE-optimal designs with different weight and WIMSE for $\tilde{\beta}$. The left graph is for the design space $\chi = [-5.7, -2.3]$ and the right graph is for $\chi = [-6.21, -0.69]$.](image)

In the left graph of Figure 6.13, for $\chi_1 = [-5.7, -2.3]$, the minimum WIMSE value is 0.009448, corresponding to an IMSE-optimal design of

$$\xi_1^* (\tilde{\beta}) = \begin{bmatrix} -5.418 & -2.3 \\ 52\% & 48\% \end{bmatrix}.$$

In the right graph of Figure 6.13, for $\chi_3 = [-6.21, -0.69]$, the minimum WIMSE value is 0.01814, corresponding to an IMSE-optimal design of

$$\xi_3^* (\tilde{\beta}) = \begin{bmatrix} -4.174 & -0.69 \\ 60\% & 40\% \end{bmatrix}.$$
The IMSE-optimal designs for $\tilde{\beta}$ exhibit features similar to those for $\hat{\beta}$; the larger support point has a smaller weight and is located on the upper boundary. In the case of the wider design spaces, the smaller support point is not as close to the boundary as we would have expected. This suggests that a three point design may be more appropriate for the larger design space.

6.8 Quadratic Predictor

The assumption so far, made from prior experience with this type of data, has been that the predictor equations are linear.

When $\hat{\beta}$ and $\tilde{\beta}$ were used as parameter estimates, and when the design space was the same as in the original experiment, D-optimal designs had two equally weighted support points on the boundaries. As the design space widened, the designs included a third support point.

In some cases, it may be sensible to assume a quadratic model. Consider the following predictor equations with the addition of a quadratic term:

$$
\eta_2 = \ln \left( \frac{\pi_2}{\pi_1} \right) = \beta_1 + \beta_2 x + \beta_3 x^2
$$

$$
\eta_3 = \ln \left( \frac{\pi_3}{\pi_1} \right) = \beta_4 + \beta_5 x + \beta_6 x^2,
$$

where $\eta_2$ and $\eta_3$ are the linear predictors, $x = \ln (\text{dose})$ represents the treatment level and $\pi_1$, $\pi_2$ and $\pi_3$ are the probabilities of dysimmunization, successful immunization and immunization failure respectively. The model now has six parameters and the information matrix has dimension $(6 \times 6)$.

We use the technique described in Section 6.3 and the data from the original experiment: the treatments, $x_i$, the number of observations, $n_i$, and the observed
values $Y_{ij}$ for $i^{th}$ ($i = 1, 2, \ldots, 5$) treatment in the $j^{th}$ category, ($j = 1, 2, 3$); the
MLE of $(\beta_1, \beta_2, \ldots, \beta_6)$ is calculated as

$$\hat{\beta} = (1.8739, 1.3587, 0.2983, -2.0, 1.459, 0.3188)^T.$$ 

The D-optimal designs for the three design spaces are shown in Table 6.15.

<table>
<thead>
<tr>
<th>Design Space</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>$w_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[-5.7, -2.3]$</td>
<td>-5.7</td>
<td>-3.922</td>
<td>-2.3</td>
<td>0.3333</td>
<td>0.3334</td>
<td>0.3333</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$[-5.99, -1.61]$</td>
<td>-5.99</td>
<td>-4.808</td>
<td>-3.599</td>
<td>-1.61</td>
<td>0.2945</td>
<td>0.081</td>
<td>0.2928</td>
<td>0.3317</td>
</tr>
<tr>
<td>$[-6.21, -0.69]$</td>
<td>-6.21</td>
<td>-4.749</td>
<td>-3.005</td>
<td>-0.69</td>
<td>0.2239</td>
<td>0.1892</td>
<td>0.2567</td>
<td>0.3302</td>
</tr>
</tbody>
</table>

Table 6.15: D-optimal designs for quadratic predictor with ML estimator, $\hat{\beta}$, and three different design spaces.

The D-optimal design for $\chi_1$ is an equally weighted, approximately equally spaced, three point design with two support points on the boundaries. The designs for $\chi_2$ and $\chi_3$ also have points on both boundaries but have four design points. Again we see that, as the design space increases in width, there is the tendency for the number of support points to increase.

The MPLE estimate is given by

$$\tilde{\beta} = (-0.4572, -0.0905, 0.0862, -1.8274, 0.8098, 0.1973)^T$$

and the D-optimal designs for the three design spaces are shown in Table 6.16.

<table>
<thead>
<tr>
<th>Design Space</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>$w_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[-5.7, -2.3]$</td>
<td>-5.7</td>
<td>-3.898</td>
<td>-2.3</td>
<td>0.3333</td>
<td>0.3334</td>
<td>0.3333</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$[-5.99, -1.61]$</td>
<td>-5.99</td>
<td>-3.653</td>
<td>-1.61</td>
<td>0.3333</td>
<td>0.3334</td>
<td>0.3333</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$[-6.21, -0.69]$</td>
<td>-6.21</td>
<td>-3.256</td>
<td>-0.69</td>
<td>0.3333</td>
<td>0.3334</td>
<td>0.3333</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.16: D-optimal designs for quadratic predictor with MPLE estimator, $\tilde{\beta}$, and three different design spaces.

Regardless of the size of the design space, all designs are equally weighted, approximately equally spaced, three-point designs with a point on each boundary.
Figure 6.14 shows the standardized variance plots. The left graph is for designs using $\hat{\beta}$ and the right graph is for designs using $\tilde{\beta}$.

In the standardized variance graphs for $\hat{\beta}$, the relative flatness of the functions near the middle design points indicates that other almost optimum designs can be found by varying the values of the design points in this region. This is not the case for $\tilde{\beta}$. The standardized variance graphs do not exhibit any flatness.

### 6.9 Summary

Speybroeck et al. (2008) analyzed the results of the Katete immunization experiment, which investigated the inoculation of cattle against the parasite that causes ECF. The response variable is multinomial with three categories, modelled by a multi category logit model with linear predictor. Two types of parameter estimates were calculated via maximum likelihood and maximum penalized likelihood methods, using data from the original experiment. The estimators were then used to determine locally optimal designs using D-optimal and IMSE-optimal techniques.
The original experiment in Speybroeck et al. (2008) consisted of five unequally weighted support points, which represented the dose in various dilutions.

The D-optimal design, over the same design space as the original experiment, was found to be two equally weighted support points on each boundary. As the design space was widened, the D-optimal support points moved further apart and sometimes a third point was added.

For two support points, the IMSE-optimal design did not behave similarly. As the design space widened, one support point always remained on the upper boundary and the other moved away from the lower boundary. The D-optimal algorithm always assigned equal weights, whereas the IMSE-optimal algorithm put slightly more weight on the lower support point. This indicates the IMSE-optimal designs place less importance on the lower dose measures.

If the original experiment was run multiple times, it is most unlikely that we would get the same responses. So we investigated how the estimated probabilities and D-optimal design would react to changes in the parameter values. First, an indirect method was considered, where slight changes were made to the experimental responses and new parameter values were calculated, as described in Section 6.5.1. Second, a direct method was considered, where new parameter estimates were simulated from a normal distribution based on the mean and variance of the original parameter estimates, as described in Section 6.5.2.

We were therefore able to incorporate some uncertainty into the design process by allowing for slightly different experimental results and also varying the parameter estimates. Most of these generated designs had two equally weighted support points on the boundaries. However, in many cases, there was also a tendency to include a third (centrally located) support point or a point shifted from the lower boundary to the centre of the design space. The estimated probability of observations falling in three categories, calculated for all these cases,
did not change dramatically.

We also found that, when the standardized variance graph for a D-optimal design was relatively flat, the support points in the flat region of the design space can be slightly varied without a major loss in optimality. This is important because it provides experimenters with some flexibility in design.

We examined design outcomes when some uncertainty was incorporated into the form of the predictor equations. When we added a quadratic term, the number of support points in the D-optimal design increased. Whether the MLE or MPLE estimators were used, the D-optimal designs were very similar for the smallest design space considered. For the larger design spaces, the MPLE designs had three equally weighted support points but the MLE designs included a fourth support point. All designs consistently have support points located on each end of the interval.

When we allow for uncertainty in the parameter values and in the linear predictor used in the model, it seems that an equally weighted design with three or four support points provides the best compromise for an optimal design. The two point design is not recommended as it is very specific to the results in the experiment examined. It makes more sense to allow for some uncertainty and incorporate more design points. The design we obtained is similar to the original experiment, except our design points are not necessarily evenly spaced.
Chapter 7

Conclusion

This thesis has investigated the difficulties in designing experiments for non-linear responses. We examined the case of a multinomial response, under the GLM framework, and demonstrated that numerical techniques must be used to obtain results because designs for GLMs do not lend themselves to analytical results.

The main problem is designing a GLM experiment is that the optimal solution is a function of the parameters of the model, but the very quantities we are trying to estimate. This complication has hindered the development of theory in this area.

The most common methods for designing a GLM experiment are called locally optimal techniques because the design is local to the particular values of the parameter estimates. We explored two design techniques, D- and IMSE-optimality, which are based on different optimization criteria and quite different in their implementation.

Various methods were used to optimize the specific design criteria. The grid search algorithm proved to be very reliable but with a significant computational burden. The recursive optimization routines, in R and FORTRAN, were
extremely fast but failed to converge for some initial values of the covariates. This problem was successfully addressed by randomizing the initial starting values and running the optimization routine until convergence was achieved. By far, the optimization routines in FORTRAN were the most efficient means of obtaining optimal designs.

We developed a robust method to find a two-point equally weighted D-optimal design, called a range design. Unlike locally optimal designs, this method does not require point estimates of the parameters. The method allows for uncertainty in their values, by describing each parameter estimate by its mean and spread, or range.

When the parameter spreads were small, range designs achieved high D-efficiencies, but as the spreads increased, their performance deteriorated. These designs were compared to midpoint designs, where the parameter estimates are the midpoints of the ranges. When the parameter spreads were not too large, the midpoint designs achieved D-efficiencies comparable to the range designs. It was then preferable to use midpoint designs because they required significantly less processing.

The range design was developed for a three-category multinomial response and a linear predictor. A natural extension to this research would analyze the performance of different types of predictors, such as those containing quadratic terms and interactions.

Also, these robust designs assumed that each parameter in the predictor equations came from an interval with the same width. It would be interesting to see the effect of removing this assumption. For example, how the designs perform when we have narrower slope coefficients compared to designs with narrower intercept terms.
We ran more simulations to find locally D-optimal designs for linear predictors with two covariates, interactions and quadratic terms. We expected to find that more covariates in the predictor equations would translate to more support points in the optimal design. While this was often the case, we found the opposite to be true for some parameter estimates. The values of the parameters seemed to have more influence in determining the number of support points than the form of the predictor.

The IMSE-optimal design technique minimizes the integrated mean squared error of the estimated probabilities. A key difference between IMSE- and D-optimal design is the role played by the number of observations in the experiment, or sample size.

The D-optimal design is not dependent on the fixed sample size as this quantity enters as a constant in the optimization function. The D-optimal designs determine the number of support points, their locations and the weight of observations at each support point. Obviously, the more observations one takes, the better the information obtained from the experiment. However, the D-optimal design locations are the same whether we have five or fifty observations.

The IMSE-optimal algorithm behaves differently. We must specify the number of support points, and the number of observations at each support point, in advance and the algorithm then determines the locations of these points. So, the IMSE-optimal design for six observations at each support point will not necessarily be the same if we have ten observations at each support point.

Without the restriction of equally-weighted support points, finding the IMSE-optimal design becomes more complicated. The IMSE-optimal designs, for each possible observation weighting combination, must be found and then compared to find the overall optimal design.
As the number of observations increases, the IMSE-optimal design optimization criterion improves and, in the case of an equally weighted two-point design, the support points move closer together. However, for some parameter values, there is a shift in the location of the support points of the IMSE-optimal design as the number of observations increases. This was shown to be related to a reduction in the squared bias component of the mean squared error.

Although the two design methods are based on different criteria, our simulations strongly suggested that the D-optimal and IMSE-optimal designs were asymptotically equivalent.

IMSE-optimal methods require the calculation of estimated probabilities and their expected values. We used penalized maximum likelihood estimates and showed that, in most cases, explicit expressions cannot be found and numerical methods must be used. The calculations of expected values proved to be a computational burden due to all of the probability combinations that must be considered.

Further work in the multinomial distribution would extend to the case of four or more categories. Some initial work indicates that the algebra is relatively straightforward but the numerical computations would be extensive. To develop IMSE-optimal algorithms for \( k \)-category multinomial distributions, investigation should be directed towards identifying methods to approximate the expected value of the estimated probabilities.

The design space plays an important role in the design methods. IMSE-optimality requires integration over the range of covariate values and is particularly influenced by the choice of design space. If there are any restrictions on the treatment values in the experiment, such information should be used to determine the design space. In the absence of such information, a quantitative method was presented to determine the design space, based on the limits of the underlying
probabilities.

These design techniques were also applied to a dose response experiment. We used data from the original experiment to calculate parameter values and then developed optimal designs. The original experiment had five, fairly equally spaced but unequally weighted, design points. Our simulations found that many optimal designs consisted of two points, one on each boundary of the design space, usually equally or near equally weighted. When we performed further analysis, and allowed for uncertainty in the model parameters and the form of the predictor, the number of design points increased to three or four. This makes sense because more design points may provide more information when there is uncertainty in the model.

Advances in GLM theory for analysis has lead to the proliferation of statistical software packages that facilitate the application of these methods. Unfortunately, the same cannot be said of the theoretical results in design of experiments for GLM models. Analytical results in optimal design have proved difficult to obtain and this thesis has relied heavily on simulations to gain insight into the behavior of designs for GLMs for the multinomial model.
Bibliography


URL http://www.R-project.org


Smith, K. (1918). On the standard deviations of adjusted and interpolated values of an observed polynomial function and its constants and the guid-
ance they give toward a proper choice of the distribution of the observations.


Appendix A

A.1 FORTRAN Programs

dopt4x.for - D-optimal design for three-category multinomial response and linear predictor with one covariate. Input five initial guesses for each $x$ covariate and five initial guesses for each weight. Allows for different starting values for each support point. Uses IMSL optimization subroutine DLCONF to optimize the design. Uses DLFDDS to find determinant of the information matrix and uses DLFTDS for Cholesky factorization.

dopt4xr.for - Finds the D-optimal design for for three-category multinomial response and linear predictor with two covariates. Input five design spaces for each $x$ covariate and five ranges for each weight. Initial guesses are then randomly generated from standard uniform or normal distributions and converted to the actual design space. Weights are constrained to sum to one. Searches for D-optimal design by maximizing the determinant of the information matrix. Uses IMSL optimization subroutine DLCONF to optimize the points and weights. Uses DLFDDS to find determinant of the information matrix and uses DLFTDS for Cholesky factorization.
dopt6x.for - As for dopt4xr.for, except linear predictor has two covariates.

   Input ten initial guesses for each $x$ covariate and weight.

dopt6xr.for - As for dopt4xr.for, except linear predictor has two covariates.

dopt8xr.for - As for dopt6xr.for, except linear predictor has two covariates and an interaction term.

dopt8xr_l.for - As for dopt8xr.for, except allows for 15 point design.

dopt_hier.for - As for dopt4xr.for, except equations adapted for hierarchical model. Expression for determinant of the information matrix, evaluated using Maple and formatted using WinEdit macros.

dopt_hierD.for - As for dopt_hierD.for, except uses DLFDDS/ DLFTDS to find the determinant of the information matrix by numerical methods.

dopt_imzsq.for - As for dopt4xr.for, except linear predictor has an additional quadratic term.

dopt_rg.for - As for dopt4xr.for, except initial parameter values are randomly generated from standard normal distribution and converted to $N(\mu, \sigma)$ using $\mu = \hat{\beta}$ or $\tilde{\beta}$ (calculated using mpleb_imz.for) and $\sigma = SE(\hat{\beta})$ (calculated using cov_imz.r).

imse2sp.for - IMSE-optimal equally weighted two-point design for three-category multinomial response and linear predictor with one covariate. Uses DLCONF to minimize WIMSE, calculated by subroutine called waimse. Since two support points give explicit solutions for MPLE of $\beta$, MPLE values are calculated in this subroutine.

imse2spN.for - As for imse2sp.for except allows for different number of observations at each support point.
imse2spG.for - As for imse2sp.for except uses grid search to find minimum WIMSE, rather than DLCONF.

imse3sp.for - IMSE-optimal equally weighted three-point design for three-category multinomial response and linear predictor with one covariate. Uses DLCONF to minimize WIMSE, calculated by subroutine WAIMSE. Requires MPLE of $\beta$ but since, in this case, we do not have explicit solutions for MPLE, these values are calculated using DUMINF to maximize subroutine called penlogli. DUMINF does not require a gradient function.

imse3spG.for - As for imse3sp.for except uses grid search to find minimum WIMSE rather than DLCONF.

mplebeta.for - Finds MPLE of $\beta$ for a particular $Y_{ij}$ combination for three-category multinomial response and linear predictor with one covariate. Design has three equally weighted support points. Since we do not have explicit solutions for MPLE, values are calculated using DUMING to maximize a subroutine called penlogli, given the gradient function, which is calculated by a subroutine called derplogli.

mplebloop.for - As for mplebeta.for except finds MPLE of $\beta$ for each combination of $Y_{ij}$ ($Y_{ij}$ values calculated using y.for).

mplebetaD.for - As for mplebeta.for except uses DLFDDS to find determinant of the information matrix and uses DLFTDS for Cholesky factorization.

mpleb_imzF.for - As for mplebetaD.for except uses DUMINF (rather than DUMING) which does not require gradient function.

mpleb_imz.for - As for mpleb_imzF.for except uses DBCONF (rather than DUMINF) which sets upper and lower bounds on BGUESS.
**mpleb_imzR.for** - As for mpleb_imzF.for except uses DBCONF (rather than DUMINF) which sets upper and lower bounds on BGUESS. Randomly generates starting values for DBCONF.

**mpleb_imzsq.for** - As for mpleb_imz.for except linear predictor has an additional quadratic term.

**mpleb_hier.for** - As for mplebetaF.for, except equations are for hierarchical model. Determinant evaluated using Maple.

**reffef.for** - Generate $\beta$ from the same uniform distributions as robust.for, with the same seed value. Input the robust design, found in robust.for, and will calculate the D-optimal design for each $\beta$ and the relative efficiency compared to the robust design.

**robust.for** - D-optimal design for three-category multinomial response and one covariate predictor with two support points. Parameters are randomly generated from standard uniform and then converted to appropriate design space. Uses grid search method to build a matrix of information matrix determinant values, where the rows correspond to the first support point and the columns correspond to the second support point. Adds the log of each matrix together and maximizes to find the robust design.

**y.for** - Generates all possible combinations of $Y_{ij}$ for a particular sample size value. $Y_{ij}$ is a multinomial response with three categories and three support points.

### A.2 R Programs

**cov_imz.r** - Covariance matrix of MLE of $\beta$. 
**dopt_cons.r** - D-optimal design using constrOptim routine to minimize negative determinant of the information matrix.

**dopt_grid.r** - D-optimal design using grid search to minimize negative determinant of the information matrix.

**dopt_rob_norm.r** - Randomly generates parameter values from normal distributions, calculates D-optimal design for each individual parameter vector and the robust design across all parameter vectors.

**dopt_rob_unif.r** - As for dopt_rob_norm.r, except simulates from uniform distributions.

**doptd1c2sp.r** - D-optimal design using grid search method for three-category response and one covariate linear predictor, with two support points.

**doptg1c5sp.r** - D-optimal design using grid search method for three-category response and one covariate linear predictor, allows for five support points.

**imse_cons.r** - IMSE-optimal two-point design for three-category multinomial response and linear predictor with one covariate using constrOptim optimization routine. Allows for different number of observations at each support point. Uses vectorized calculations for improved performance.

**imse_cons_lp.r** - IMSE-optimal two-point design using constrOptim. Must have same number of observations at each support point. Uses looping which is much slower than version with vector calculations.

**imseN_cons_lp.r** - As for imse_cons_lp.r except can have different number of observations at each support point.

**imse_grid.r** - IMSE-optimal two-point design for three-category multinomial response and linear predictor with one covariate using grid search over
a range of values. Allows for different number of observations at each support point. Uses vectorized calculations for improved performance.

imse_grid_lp.r - IMSE-optimal two-point design using grid search. Must have same number of observations at each support point. Uses looping

imseN_grid_lp.r - As for imse_grid_lp.r except can have different number of observations at each support point.

loglik_optim.r - Uses Optim subroutine with method = Nelder-Mead (as gradient is not specified) to maximize the penalized log-likelihood and solve for MPLE of $\beta$.

loglik_gradient.r - Uses Optim subroutine with method = BFGS (as gradient functions are specified) to maximize the penalized log-likelihood and solve for MPLE of $\beta$.

mse.r - MSE of penalized estimators.

mse_compare.r - Average MSE for penalized estimators over a range of $x$ values for two pairs of support points, each over their respective design ranges.

plot_beta.r - Plots probabilities for original $\beta$ (in bold) and all simulated $\beta$.

plot_hier.r - Plots probabilities for hierarchical model.

plot_imseimz.r - Plots IMSE-optimal design points, design point size adjusted by weight on same chart as WIMSE, with scale on second y-axis.

plot_mlemple.r - Plots MLE and MPLE for increasing values of sample size.

plot_mse2_3.r - Plots MSE for IMSE-optimal designs with two and three support points.

plot_mse_bd.r - Plots MSE broken down by variance and bias components.
**plot_optdesign.r** - Plots WIMSE design for increasing values of observations at each support point. Data sourced from OptDesign.xls and then relevant fields are copied to *.txt files read by this program.

**plot_pi.r** - Plots probabilities for three-category response and one covariate linear predictor.

**plot_piiimz.r** - Reads in parameter values and plots three probability charts.

**plot_re.r** - Uses relative efficiencies (calculated in releff.for) and plots relative efficiency frequencies in a barplot, top 10% designs in a scatterplot and boxplots of each support point.

**plot_x1w1.r** - Plots support points whose size is adjusted by weight.

**re_imz.r** - Relative efficiency of various D-optimal designs against the original design in immunization experiment.

**sv1cov.r** - Standardized variance curve (using the general equivalence theorem) for three-category response and linear predictor with one covariate.

**sv2cov.r** - Standardized variance contour plot for linear predictor with two covariates.

**sv2covint.r** - Standardized variance contour plot for linear predictor with two covariates and interaction term.

**sv2covfix_x.r** - Standardized variance curve for linear predictor with two covariates, fixes the x-values.

**sv2covfix_z.r** - As for sv2covfix_x.r, except fixes the z-values.

**sv2covintfix_x.r** - Standardized variance curve for linear predictor with two covariates and interaction term, fixes x-values.

**sv2covintfix_z.r** - As for sv2covintfix_x.r, except fixes the z-values.
sv_imz.r - Standardized variance for immunization example, shows all simulated parameter vectors (obtained from either modifying responses or standard deviation of parameter estimates). Original design parameters in bold.

sv_imz3g.r - Plots three standardized variance graphs, for three different design spaces, on one graph.

sv_imzsq.r - As for sv_imz.r except model includes a quadratic term.

sv_imzsq3.r - As for sv_imz3g.r except model includes a quadratic term.

sv_hier.r - Standardized variance for hierarchical model.

### A.3 Excel Programs

**design_space.xls** - Determines the design space for a particular $\beta$, showing the individual probability limits as $x \to \pm \infty$. Probabilities are calculated in pi.r and imported into this spreadsheet.

**determinant.xls** - Determinant values for different linear predictors.

**imse_imz.xls** - IMSE-optimal values for immunization example.

**prob_undef.xls** - Calculates the probability that any $Y_{ij} = 0$ for specified value of $\beta$ and sample size.

**mse_hat.xls** - Calculates the MSE for a particular $\beta$ and sample size for a design with two support points at a specified value of $x$. Uses ML estimators of $\beta$.

**mse_star.xls** - Calculates the MSE for a particular $\beta$ and sample size for a design with two support points at a specified value of $x$. Uses penalized ML estimators of $\beta$. 
sens_Yij.xls - Parameter values and designs when varying experimental responses.

sens_beta.xls - Parameter values and designs when varying the parameter estimates by their standard deviations.

x1x2_boxplots.xls - Statistical summaries on randomly generated robust designs.

A.4 Maple Programs

det_2sp.mw - Determinant of information matrix for three-category multinomial response, one covariate linear predictor and two support points.

det_3sp.mw - As for det_2sp.mw, except for three support points.

det_5sp.mw - As for det_3sp.mw, except for five support points.

det_hier7.mw - Determinant of information matrix for multinomial hierarchical model with seven support points.