Delft University of Technology

Master Thesis

Applied Mathematics

Bayesian Variable Selection in Probability of Default Models

Author:
K.M. Carmiggelt (4103793)

Supervisor TU Delft:
Dr. D. Kurowicka

Supervisor Deloitte:
B.P. Ritzema Msc.

3rd October 2019

Preface

I wrote this thesis as a graduation project for the degree of Master of Science in Applied Mathematics at the Delft University of Technology. I followed the Financial Engineering track. This specialisation has a focus on valuation of financial products. Besides that I took courses in statistics, machine learning and credit risk management.

I wrote the thesis during an internship at Deloitte at the Financial Risk Management team, which part of Risk Advisory Department. I investigated variable selection method for Probability of Default models. This is a topic in credit risk management, which is one area of expertise of the Financial Risk department. The department does not only limit itself to banking, but also gives advise other types of companies in the financial sector.

I would like to thank Berend Ritzema, my thesis supervisor at Deloitte, for his helpful weekly input and guidance during the process. I would also like to thank other member of the FRM and fellow interns for their help during my thesis and for giving me a better insight into the financial sector. I would like to thank Dorota Kurowicka, my supervisor at the TU Delft, for our helpful meetings and her quick feedback on my work. Finally, I would like to thank Jakob Söhl for being my second reader.

Summary

Banks are financial institutions that lend money from other parties and provide loans to individuals and organisation for a higher interest. Lending out money is associated with the risk that debtors are not able to fully or partially repay the loans. This is called credit risk.

Banks have to make an estimate of the credit risk in their portfolios and have to keep reserves for potential losses. The way this risk is to be determined, is decided by the government where the bank is established. In Europe, the United States, Russia, China among others, the legislation on credit risk is derived from Basel III. Basel III is an international framework to homogenise banking regulation across the world.

There are three important factors to determine credit risk In Basel III, namely Probability of Default, Loss Given Default and Exposure at Default. In this thesis I investigate Probability of Default (PD) modelling.

The size of the portfolio, for which the Probability of Default has to be estimated, can vary greatly. When the amount of defaults in a portfolio is low and the amount of explanatory variables is high, there is a risk of overfitting. Variable selection methods can be used to counteract overfitting and give understanding of the important predictors. I apply variable selection methods on on a logistic regression.

I look at three Frequentist variable selection methods, namely Forward Selection, Lasso and Relaxed Lasso. I compare these three methods with Predictive Projection combined with a Horseshoe prior, which is a Bayesian approach to variable selection.

Forward Selection starts with only the intercept in the model and adds variables one by one to the model. The variables are added in such a way that each step increase the estimated performance the most.

The Horseshoe prior and Lasso Regression are types of regularisation, where the estimates of the regression coefficients of the logistic regression get shrunk to zero. In Lasso regression, this is done by adding a L^1 penalty of the regression coefficients to the logistic regression. This causes weak signals to be pulled to zero. Lasso shrinks all regression coefficients to zero to some degree, even those with a strong signal.

Lasso can also be used to find an order of importance for the regression coefficients by varying the strength of the L^1 penalty. Regression coefficients are set to zero one-by-one as the penalty increases. Relaxed Lasso uses this rank and refits the variables without regularisation.

In Bayesian statics, regularisation is added via the prior. The Horseshoe prior can adjust to the average sparsity in the model and the Horseshoe prior either shrinks a signal aggressively to zero, or leaves the signal almost unchanged. The posterior of the model is never truly sparse. Predictive Projection can induce sparsity by setting the Monte Carlo samples of the posterior to zero for certain variables. This is done in such a way that the Kullback-Leibler divergence between the full posterior and the projected sparser posterior is minimised.

I investigate the behaviour of the variable selection methods. The main focus is on the predictive performance, the sparsity, the computation time and the reliability of the estimated performance for the selected models.

I apply the methods to various types of simulated data to compare the variable selection methods. The simulated data consist of data with independent predictors, collinear predictors and non-normal predictors, among others. The simulations studies show that Lasso and Predictive Projection lead to models with the highest performance overall and the predictive performance is more stable over different realisation of the data. For the same performance the Predictive Projection produces models with less variables. This makes Predictive Projection the most attractive method. I also employ the techniques to FreddieMac data, which is a data set on single-family mortgages. The results are similar to the simulated data and Predictive Projection with the Horseshoe prior is the most attractive variable selection method. Both the simulation studies and the FreddieMac application imply that the estimated performance of the Predictive Projection and Lasso are better than those of Forward Selection and Relaxed Lasso. However, the behaviour of the estimated performance remain unclear to a certain degree. More simulations per data type and more data types are needed for more insight into the estimated performance. Additional resources are needed to achieve this.

List of Symbols and Abbreviations

Latin Symbols

H	Hamiltonian
K	Kinetic Energy
n	Index of sample from data set
N	Size of data set
p	Probability mass distribution, probability density distribution, likelihood
P	Probability measure
s	Index of Monte Carlo sample
S	Size of set of Monte Carlo Samples
T	Markov Chain Transition Kernel
V	Potential Energy
x	Explanatory variables $x \in X$
X	Set of explanatory variable
$\tilde{X}$	Explanatory variable corresponding to unobserved $\tilde{y}$
y	Response variables
$\tilde{y}$	Unobserved value of response variable

Greek Symbols

α	acceptance rate (MCMC)
β_0	Intercept of Regression
β	Vector of Regression Coefficients
β_i	Single Regression Coefficient for $i \in \{1, ..., D\}$
θ	Parameters $\theta \in \Theta$, Probability of Default
Θ	Random Variable (Parameter)
κ	Shrinkage Weight
λ	Regularisation parameter, Local shrinkage parameter (Horseshoe prior)
ρ	Correlation coefficient
σ	Standard deviation
Σ	Covariance matrix
τ	Global shrinkage paramater (Horseshoe prior)
ζ	Momentum (HMC)
Ω	Parameters/Sample space

Abbreviations

EAD	Exposure at Default
elpd	expected log predictive density
ESS	Effective Sample Size
HMC	Hamiltonian Monte Carlo
KL	Kullback Leibler Divergence
LGD	Loss Given Default
LOO	Leave-One-Out Cross Validation
MAP	Maximum a Posteriori estimate
mlpd	mean log predictive density
MCMC	Markov Chain Monte Carlo
PD	Probability of Default
PSIS	Pareto Smoothed Importance Sampling

Contents

1 Introduction

Lending money to people or organisations is associated with certain risk. One of those risks is that the borrowed amount will not be repaid on time, not fully or not at all. A reason this might happen is that the debtor has a structural deficiency in his cash flow, for example a debtor may have lost its job. Even if the debtor has sufficient cash flow over an extended period of time, it might happen that the debtor is not able to pay for a couple of months. This risk associated with lending money to people is called credit risk and is the topic of this thesis.

There are a couple of reasons to determine the credit risk. The first one is to determine whether to provide a loan to a potential debtor and to determine which interest rate needs to be charged. The second reason is to determine the amount of defaults and losses that will be incurred during a upcoming period. A bank might get into bankruptcy itself, when too many defaults happen at the same time. Banks need to keep capital reserves to mitigate this risk. Furthermore, the correct identification of the loans that go into default next year is of interest to the banks, so that they can take appropriate measures to prevent excessive losses on loans.

1.1 Basel III

Banks need to maintain capital reserves to handle defaulting debtors. The way the capital reserve is determined is highly influenced by the laws of the country where a bank is located. In 1974 the Basel Committee on Banking Supervision (BCBS) was founded in order to homogenise legislation, related to banking activities, across its members states. The Basel Committee does not produce binding legislation, however the advice given by the committee often leads to legislation in the 48 member's jurisdiction. These members include the European Union, The United States, Japan, China and Russia (*Basel Committee membership*, 2013).

Currently the third Basel accord, Basel III, is being implemented. Under Basel III, three important factors to determine the necessary reserves are the Probability of Default (PD), the Exposure at Default (EAD) and the Loss Given Default (LGD). Basel III defines these factors as (Basel Committee on Banking Supervision, 2017) :

- Probability of Default: A bank should consider a client to be in default when the client is 90 days past-due on any of its credit obligations to the banking group, or/and the client is unlikely to pay its credit obligation. Furthermore, the Probability of Default should be calculated for a one-year basis, so the probability that the debtor goes into default in the upcoming year.

  Throughout this thesis the definition of PD only consists of a three month pay delinquency. This definition neglects clients that are not likely to repay their debts. Also, this definition disregards other credit products where the client may have defaulted. The reason for this choice is that the data is not available and therefore is not remains outside of the scope of this thesis.

- Exposure at Default: The outstanding amount of the loan at the time of default. This amount can be higher than the current outstanding amount. For example, when client with liquidity issues has a credit line, the client might withdraw extra money, which leads to a higher exposure at default. In case of a mortgage the Exposure at default can be higher than the current exposure due to cumulative interest that has not been paid during the 90 days prior to the default.

- Loss Given Default: The default on a loan does not immediately lead to the loss of the entire EAD. The loan can sometimes be restructured to such an extent that the client can continue paying a portion of the original loan. For a mortgage, the collateral can be sold off, leading to a lower loss then the Exposure at Default. The LGD is measured as a percentage of the Exposure at Default.

Under Basel III, there are three different approaches for banks to determine their mandatory capital reserves, namely:

- Standardised Approach.

- Internal Ratings-Based Approach.
  - Foundation Internal Rating Based approach (F-IRB).
  - Advanced Internal Rating Based approach (A-IRB).

Under the Standardised Approach, banks do not provide their own estimates of PD, LGD and EAD. Instead, banks use risk weights provided by the regulator, for example the European Central Bank. These risk weights are based on the credit rating and the type of the credit product. For example, a low risk sovereign AAA-bond has a risk weight of 0 % and a risky corporate B-bond has a risk weight of 150%. Multiplying the weight of the assets by the value of the assets results in the Risk Weighted Assets (RWA). For RWA the regulator demands that the bank maintains certain capital reserves.

Under the Foundation Internal Rating Based (F-IRB) approach, banks must provide their own Probability of Default estimates. The supervisor provides the Loss Given Default estimates and Exposure at Defaults estimates. The calculation of the PD gives banks a better insight into the risks that are present in their portfolios. In general, the required capital reserves under F-IRB are lower than the reserves under the Standardised Approach.

The last approach is the Advanced Internal-Ratings Based approach, where banks have the most freedom to calculate their risk, namely it has to calculate the PD, EAD and LGD. This approach gives the most insight into the risks of the loans, and in general has the lowest capital requirement of the three methods.

I only consider the Probability of Default. The Exposure at Default and the Loss Given Default are also important in calculating the risk. However these are not in the scope of this thesis.

1.2 Probability of Default models

A famous model Probability of Default is Merton's model (Merton, 1974). This model tries to predict the future value of a company with a geometric Brownian motion. The company goes into default when the value of the company falls below its debts. The Merton model is closely related to the Black-Scholes model for pricing financial options (Black & Scholes, 1973).

The geometric Brownian motion of the Merton model is not an accurate model for the real future value. The geometric Brownian motion assumes that the company has lognormal returns, however in reality the returns often have more heavy tails. Moody's KMV is a model of the American cooperation Moody's, that tries to give a better prediction of the future value of the company by using empirical data. Moody's KMV also relaxes some unrealistic assumptions of Merton's model. For example in Moody's KMV, companies can go into default multiple times a year throughout the year. While in Merton's model this can only happen once a year on a fixed predetermined date.

The previous two models are structural models, in which there is a clear theoretical explanation of the mechanisms behind the default, that is to say, the value of the company drops below the value of its debts. The problem with these models is that they are only applicable to publicly traded companies where the value of the company is known. These types of models are not really applicable to consumer credit, because it is hard to evaluate the value of the consumer. I model the Probability of Default of mortgages, which is a type of consumer credit.

In consumer credit non-structural models are more often employed. Examples of these models are time-series models, survival analysis models and the logistic regression. The choice of the model depends on which types of variables are used. For different types of variables, the model gives either a Through-the-cycle (TTC) estimate or Point-in-Time (PiT) estimate. The goal of a Through-the-Cycle estimates is to give a stable estimate over multiple years. Macroeconomic and other time dependent variables are avoided in these types of estimates. The goal of Point-in-Time estimates are to give a as well as possible estimate of the PD for any given time. These types of estimates utilise macroeconomic and other time dependent variables, when this increases the predictive power of the model.

In the Basel Framework, Point-in-Time estimates are not desirable as they lead to procyclicality. When a model is dependent on macroeconomic variables, the reserves in a booming economy will be low, because the Probability of Default and the Loss Given Default are low in those periods.

On the other hand, when the economy goes into a recession, the mandatory reserves will increase. This can lead to a situation where banks are less likely to lend money to others, which causes a credit crush that may aggravate the recession.

Through-the-Cycle estimates are meant to be stable estimates throughout a credit cycle, and will lead to less fluctuation in the capital demands of banks. This is the reason why in the Internal-Ratings Based approaches, TTC estimates are preferred to PiT estimates. Even if the models has less accurate predictions than the PiT estimates.

When modelling Point-in-Time PD, a natural choice would be a model which takes time effects into account, for example, a vector autoregressive model. In a TTC estimate, we do not want this time effect, therefore time series models are not considered. Instead, I use a logistic regression, this model is a parametric model that assumes a linear relation between the log-odds of the Probability of Default and explanatory variables X.

1.3 Variable Selection

Portfolios of debtors can vary greatly in size. The size can range from only a couple of observations up to millions of observations. An observation is a data input of a loan for every year that a loan is in the portfolio. The issues associated with the two extremes are different. For huge data set, there is an abundance of information, but due to the size of the data set it can take a long time to fit models and only simple models, like Frequentist logistic regression, are a real option. On the other extreme there are the low-default portfolios with only a couple of observed defaults. For low-default portfolios there is little information in the data set. In this situation using explanatory data is not feasible as it is even hard to estimate average default rate of the portfolio.

For portfolios that are slightly larger than a low-default portfolio, it is still hard to estimate the Probability of Default. However, it might be possible to use some explanatory variables. There are many potential features that might help predict the Probability of Default. logistic regression is likely to overfit when using all there features. Furthermore, models become less comprehensible when too many variables are used. This means that only certain variables should be included in the model. Where the preference is of course for the variables which have the highest explanatory power of the Probability of Default.

There are multiple approaches to select variables for the model. A method where all possible combination of explanatory variables are fitted and taking the model with the best estimated performance is not a good approach. If there are D explanatory variables, the amount of model that need to be fitted is 2^D. For twenty variables this already leads to fitting over a million models. Moreover, this approach has the tendency to pick models that have too many explanatory variables, overestimate the performance of the chosen model, while performing badly (Piironen & Vehtari, 2017).

Heuristics can greatly reduce the amount of submodels that need to be fitted and evaluated. Forward Selection is the first type of variable selection that I consider. This method starts with only the intercept and adds variables one at a time to the model.

It is also possible to use regularisation as a heuristic to find the most important explanatory variables. In Frequentist statistics, Lasso regression is candidate for variable selection. By increasing the regularisation parameter in Lasso regression, more and more regression coefficients corresponding to the explanatory variables are set to zero. This gives a ranking of the features, which can be used for feature selection. In Section 4 Forward Selection and Lasso variables selection are discussed. In section 5, I investigate a type of Bayesian variable selection, namely Predictive Projection with a Horseshoe prior. The Horseshoe prior is a type of Bayesian regularisation, which shrinks features with a weak signal towards zero, while leaving the strong signals almost unchanged. Lasso regression does not have this property and has the tendency to also partially shrinks strong signals towards zero. The posterior of the Horseshoe prior is not truly sparse. True sparsity is induced by Predictive Projection, which projects the posterior from the full model space to a model space with less variables. I apply the variable selection methods to simulated data (Section 6) to investigate the way they perform. In these simulation studies the data generating process is known, so the models can be compared to the real data generating process.

Lastly the variable selection methods are also applied on real life data, namely on mortgage loans in the FreddieMac data set. FreddieMac is a American government-sponsored company,

which is tasked with buying and selling mortgages on the secondary market with as goal to increase the liquidity, stability and affordability of houses in the United States. On their website, FreddieMac has a publicly available data set on single-family mortgages.

1.4 Research Design

In this thesis, I investigate modelling credit risk of portfolio that only contain a couple of dozen defaults. This means that the information in the data set is scarce, which makes inference hard. Nevertheless, portfolio with few observation can have represent large money value. Getting a better insight in these portfolios are important to mitigate risks.

Probability of Default is relevant for both Internal Rating Based approaches. I only consider Probability of Default modelling. Loss Given Default and Exposure at Default modelling can only be applied to the Advanced Internal Rating Based approach. Furthermore, Loss Given Default and Exposure at Default models only consider the defaults of the portfolio. The amount of data is even more limited, and therefore LGD and EAD are not discussed.

There are many potential features that may help in predicting defaults. Including too few variables gives the risk of not finding the important variables and suboptimal prediction. Including all variables can lead to overfitting and bad predictive performance as well. Variable selection methods can counteract overfitting and increase the performance of the model. I look at methods that finds the features necessary for predicting Probability of Default.

As a base model I use logistic regression, which is an industry standard. Although, the Probability of Default is dependent on the year of the observation, I still make the assumption that the observations are time independent to get a Through-The-Cycle estimate. I apply feature selection methods on the logistic regression to find the important variables. The new method I examine is a logistic regression with a Horseshoe prior. The fitted model contains all variables, where noisy variables are shrunk towards zero. Subsequently, Predictive Projection selects the variables which contributes the most to the prediction. This is a Bayesian approach to variable selection.

I compare the Horseshoe prior with Predictive Projection to three common methods. The three methods are Forward Selection, Lasso regression and relaxed Lasso.

Bayesian models automatically represent some correlation in the model fit. Forward Selection does not directly take this into account. Furthermore, the Horseshoe prior has better shrinkage characteristics than the Lasso regression. Therefore, I expect that the Horseshoe prior has better predictive performance than the other methods, especially for correlated data. This leads to the first research question.

- **Research Question:** How does Bayesian variable selection, with a Horseshoe prior and Predictive Projection, compare to Forward Selection, Lasso variable selection and relaxed Lasso variable selection in simulated PD data?
  - Independent Explanatory Data.
  - Collinear Explanatory Data with:
    - Masking effect.
    - Aligned effects.
    - Correlation with unimportant variables.
  - Non-normal Predictors.
  - Misspecified Models.

To answer this question, I apply the different methods to simulated data. The simulations offer a controlled environment, making them suitable for the comparison of the methods on different types of data.

Throughout the thesis I use 4,000 samples with 80 observed defaults and 20 explanatory variables in the examples. All the variable selection methods work reasonably in this setting.

In the simulations studies I consider 1,000 samples, with 20 defaults. For 1,000 samples the difference between the variable selection methods becomes more pronounced, because the variable

selection becomes harder. Even though 1,000 observations sounds like sufficient data, variable selection is hard due to the fact that the data is imbalanced. I run 30 simulations for the four methods and six data types, which This results in 720 runs.

The drawback of simulated data is that it might not be representative of real data. It is likely that in real data there are unforeseen relations. For this reason, I also test the variable selection methods on real life data.

- **Research Question:** How do the variable selection methods perform on real life data?

I look at the performance of the different methods on FreddieMac data. This data is publicly available and easily accessible. Consequently, it is often used for research. This data set contains 26.6 million loans, however I only take a small portion to recreate the hypothetical low-information feature-selection setting. Due to the abundance of data, I can make numerous training data sets with 1,000 observations, and a hold-out data set with 100,000 observations to test the predictive performance. Combined with the simulation, this gives seven different types of data. I run 30 iterations for different realisation of the data for the seven scenarios and four variable selection methods, Which results in 840 runs.

1.4.1 Evaluation Criteria

To compare the performance of the variable selection methods, I keep the following criteria in mind:

- **Predictive performance:** Methods with better performance are preferred as they give more insight into the credit risk of the portfolio.
- **Variability in the predictive performance:** Everything equal, methods that have similar predictive performance for different realisation of the data are preferable, because they induce less risk.
- **Number of variables in the model:** Models that need less variables to give the same predictive performance are preferred, because they give more insight into the important risk drivers of the portfolio.
- **Computation Time:** Everything equal, methods with lower computation time are preferred.

As a measure of predictive performance I use the expected log predictive density (elpd). Which is an estimate for the cross entropy of the model. I predict the elpd via psis-loo and K-fold cross validation.

- **Research Question:** Do PSIS-LOO and K-fold cross validation give good estimates for the real out-of-sample performance for the different variable selection methods?

To answer this question, I look at the difference of the real performance and the estimated performance in the simulations. I also consider the variability of the error of the estimated performance.

2 Logistic Regression

In this section I give a short recap on the basics of the logistic regression. Here I discuss the difference between Bayesian and Frequentist logistic regression. Furthermore, I discuss the problems associated with imbalanced data. In credit risk, data is often imbalanced as there are often more non-defaults than defaults in the portfolio. These topics are the building blocks for the variable selection methods.

2.1 Notation

Throughout this thesis, the response variable y is written as a vector with n observations. The notation y_i denotes a single observation in the vector y. The response variable y is a Boolean, with values default and no default. Sometimes I write $y \sim p(y)$, this denotes that the observed response variable is generated according to a certain data generating process. The models can depend on explanatory data X, which is a $D \times n$ matrix, where D is the number of explanatory variables. As in case of the response variable, the explanatory variables X_i refers to a single observation of the data with D variables. The goal is to make prediction whether a loan will default or not. This is an unobserved events and is denoted by $\tilde{y}$. Again the prediction can be made with use of explanatory data $\tilde{X}$. This explanatory data is observed, but the tilde means that it corresponds to the unobserved default event $\tilde{y}$.

θ refers to a realisation of the parameter Θ. This notation is used as a general notation, but when in specific cases the more common notation is used. For example, the symbol that is used for regression coefficients is β. A list of common symbols is shown in Table 1 and a full list of symbols and abbreviations is on page 5.

Symbol	Definition	Dimension
D	Dimension/Number of explanatory variables	
n	Number of observations	
S	Number of Monte Carlo Samples	
y	Observations of the response variables	$(1 \times n)$
$\tilde{y}, \tilde{\theta}$	Unobserved variables	
X	Matrix of explanatory variables	$(D \times n)$
X^d	Vector of one explanatory variable	$(1 \times n)$
X_i	Response variables of single observation	$(D \times 1)$
$\tilde{X}$	Observed explanatory data associated with $\tilde{y}$	
β	Row vector of regression coefficients	$(1 \times D)$
β_d	Single regression coefficient	
β_0	Intercept	(1×1)
$\beta^s, \theta^s, \tilde{y}^s$	Monte Carlo Sample of variable	
$\hat{\beta}, \hat{\theta}$	Estimates of the parameter	$(1 \times D)$
θ	Probability of Default	

Table 1: Symbols and definitions

The likelihood function is written as $p(y|\theta)$, which is a common notation in Bayesian statistics. This denotes the same as $\mathcal{L}(\theta; y)$, which is more commonly used in Frequentist statistics. Often the scaling parameter in the normal distribution is written as a variance, such that $N(\mu, \sigma^2)$. However, I prefer the standard deviation notation, such that it is $N(\mu, \sigma)$. Standard deviations scales linearly instead of quadratic, and linear relations are more intuitive than quadratic relations. The same is done for other distributions when applicable.

2.2 Logistic Regression

Modelling defaults is a classification problem, where the outcome is either 1 for a default, or 0 for no default. The probability of a default is denoted as $P(y = 1|\theta)$ and the probability that the loan will not go into default is $P(y = 0|\theta)$. For a Bernoulli distribution $P(y = 1|\theta) = \theta$, so the θ

is the Probability of Default and the probability of no default is $P(y = 0|\theta) = 1 - \theta$. There are many models that can be used for classification. The one that I am using is the logistic regression. The model is most conservative in the sense that it is a maximum entropy model for independent observations (see Appendix B for more information on entropy and Information Theory). The logistic regression has a highest entropy given the data, and therefore is more conservative than for example the probit model (McElreath, 2018). Ohlson (1980) applied logistic regression to estimate the Probability of Default of companies. Besides the theoretical justification, the logistic regression is also an industry standard. This makes the logistic regression useful for explaining the results of the model in the banking branch.

Logistic regression assumes a linear relation between the log-odds and explanatory variables X, which is a matrix with n observations and D variables. Regression coefficients β express the effect size of this relation. This is a vector with D variables. The intercept β_0 does not represent a relation between y and a variables, but serves as parameters that set the model to the right amount of average y.

$$\begin{aligned} \text{logit}(\theta) = \log\left(\frac{\theta}{1-\theta}\right) &= \beta_0 + \beta X \\ y &\sim \text{Bernoulli}(\theta) \end{aligned} \tag{1}$$

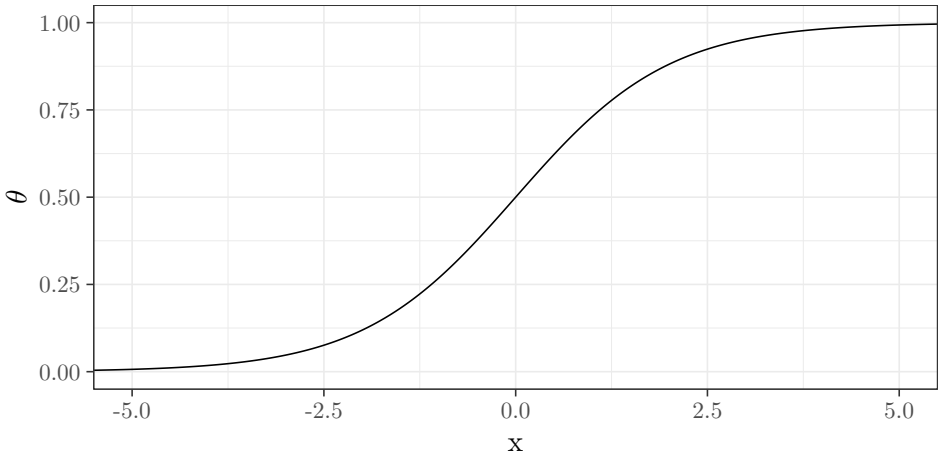

Figure 1: Logistic function with one explanatory variable x, where $\beta_0 = 0$ and $\beta = 1$, in Equation 1

Figure 1 shows a logistic function with one explanatory variable. The logistic function is bounded between zero and one. This is also the case for a probability. If $x = 0$ then $P(y_i = 1) = P(y_i = 0) = \frac{1}{2}$. As x get smaller the Probability of Defaults θ get lower, for example for $x = -5$, the probability is 0.67%.

When the response variables y are drawn conditionally independent given β_0 and β, then the likelihood function is:

$$\begin{aligned} p(y|\beta_0, \beta, X) &= \prod_{i=1}^{n} p(y_i|\beta_0, \beta, X_i) \\ &= \prod_{i:y_i=1} \frac{1}{1+\exp(-\beta_0 - \beta X_i)} \prod_{i:y_i=0} \frac{1}{1+\exp(\beta_0 + \beta X_i)} \end{aligned}$$

2.2.1 Frequentist Logistic Regression

In this thesis I deal with two statistical paradigms, namely Bayesian and Frequentist. The main focus is on Bayesian statistics, however due to the wide prevalence of Frequentist statistics this is also considered. A fundamental difference between Frequentist and Bayesian statistics is the definition of probability. In the Frequentist paradigm the assumption is that there exists a fixed value θ, which represents the a real data generating process. The goal is get a estimate $\hat{\theta}$ which get as close as possible to this real value θ. This estimate $\hat{\theta}$ is seen as a fixed value as well. As the name suggest Frequentist statistics is based around the frequency of events occurring. Where probability is defined as relative frequency of events occurring as the number of observations goes to infinity.

$$\theta = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} y_i}{n}$$

A common technique to get an estimate is the maximum likelihood estimate, which is defined as:

$$\hat{\theta}^{MLE} = \underset{\theta}{\operatorname{argmax}}\, p(y|\theta)$$

But in practise this problem is often solved by maximising the logarithm of the likelihood, which is numerically more attractive. Optimisation algorithm can find β_0 and β that maximise the values of the likelihood function. For the logistic regression the maximum likelihood is solved by:

$$\underset{\beta_0,\beta\in\mathbb{R}^{D+1}}{\operatorname{argmax}} \left\{ -\sum_{i:y_i=0} \log(1+\exp(-\beta_0-\beta X_i)) - \sum_{i:y_i=1} \log(1+\exp(\beta_0+\beta X_i)) \right\}$$

2.2.2 Bayesian Logistic Regression

In Bayesian statistics, probability is not seen as a asymptotic relative frequency, but as a representation of uncertainty about knowledge. In the Bayesian paradigm the consensus is that the only way to represent the uncertainty is to model it as a parameters as random variable Θ. From this philosophical difference, a different approach of inference originates. To get inference on a parameter θ Bayes' Theorem is used.

Theorem 1 (Bayes' Theorem)**.** *For a parametric model with data y and parameters θ.*

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

Where $p(\theta|y)$ is called the posterior, the posterior is a distribution on θ after seeing the data y. As in the case of Frequentist statistics, the likelihood function $p(y|\theta)$ is used. Unlike in Frequentist statistics the $p(\theta)$ defined. This is the prior and it represents the knowledge of the parameter θ before seeing the data. The interpretation of priors has been a subject of debate and this can be divided into two ideal typical views:

- Subjective view - Probability represent uncertainty about knowledge and personal beliefs. The prior can be used to implement prior knowledge and beliefs in the model. For example if we expect a parameter to have a influence, the prior can represent this belief by setting a prior with much mass around the prior value. (reference subjective)
- Objective view - The prior should have the least possible effect on the posterior inference and there is not place for subjectivity in science. A prior suggested in this view are the reference priors, which are least informative in the information-theoretic sense (Berger et al., 2009).

Throughout this thesis, I adapt a subjective view. The priors I use are weakly informative in the model as I do not have a priori information on which variables are important. The weakly informative priors still imply a believe on what is possible amount of non-zero parameters β corresponding to in the data set. These priors are also sometimes needed to guarantee numerical stability.

A common way to find results in the Bayesian framework is to use the proportional sign $\propto$, which means left hand side of the equation is proportional up to a constant to the right hand side. The denominator $p(y)$ is an normalising constant, therefore the proportional sign can be applied to the posterior. When this notation is used the right hand side of the following formula is called the unnormalised posterior.

$$p(\theta|y) \propto p(y|\theta)p(\theta)$$

In some cases an analytic result can be found for the posterior $p(\theta|y)$. A class of priors for which this is possible are the conjugate priors. A prior is conjugate when the posterior is of the same class of distributions as the prior. For example, for a binomial likelihood the beta distribution is a conjugate prior. The Beta(α, β) distribution is proportional to:

$$p(\theta|\alpha, \beta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$$

And the binomial likelihood can be written as proportional to:

$$p(y|\theta) \propto \theta^{k}(1-\theta)^{n-k}$$

Where k denotes the number of observation where $y = 1$ and n is the total amount of observations. The unnormalised posterior is:

$$p(\theta|y) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}\theta^{k}(1-\theta)^{n-k} = \theta^{\alpha+k-1}(1-\theta)^{\beta+n-k-1}$$

This unnormalised posterior corresponds to the proper posterior Beta$(\alpha + k, \beta + n - k)$. These conjugate priors are computationally convenient, however do not always exist or are not desirable.

The conjugate prior can be used for the estimation of Probability of Default if no explanatory variables are considered. However, explanatory variables could benefit the prediction. The logistic regression can do this and the equation to solve is:

$$p(\beta_0, \beta|y, X) \propto p(y|\beta_0, \beta, X)p(\beta_0, \beta)$$

In general priors are not conjugate and the priors applied in thesis are not conjugate either. The unnormalised posterior is often easy to calculate, however the normalising constant, which can be found by integration, is hard to calculate in high dimensional probability space. To find the posterior, I use a Markov Chain Monte Carlo method called Hamiltonian Monte Carlo. Markov Chain Monte Carlo methods are numerical methods that draw samples from the posterior with use of (pseudo-)random numbers. Hamiltonian Monte Carlo only needs the unnormalised posterior to draw samples from the posterior. These samples are called Monte Carlo samples and are denoted by θ^s. From the Monte Carlo samples, statistics can be calculated, such as the mean and variance of the posterior distribution. There is more information on Markov Chain Monte Carlo in Appendix A. I implement the Bayesian Logistic regression with R-package *rstanarm* and the Frequentist logistic regression with R-package *glm*. These two packages have the same syntax, but for *rstanarm* a prior has to be chosen.

Example 2.1 and Example 2.2 show two applications of Bayesian and Frequentist logistic regression on two types of data.

Example 2.1: Logistic Regression on Independent Explanatory data

For a model with data generating process of a logistic regression such that:

$$\text{logit}(\theta) = \beta_0 + \beta X$$

With the parameters β as in Table 2. Furthermore let:

$$y \sim \text{Bernoulli}(\theta)$$

The explanatory variables X are drawn from a multivariate normal (MVN) with correlation and a standard deviation of one:

$$X \sim MVN(0, \mathbb{I}_D)$$

Where $\mathbb{I}_D$ is the $D \times D$ identity matrix

	β_0	β_1	β_2	β_3	β_4	other β
Value	0	1	0.75	-1	-0.75	0

Table 2: Parameters of the data generating process

Draw $n = 4000$ observations, from this data generating process. This data generating process is balanced, that is to say that are as many defaults as non defaults. This can be seen in Figure 1. Because β_0 is zero and X is symmetric all the probabilities are centred around a probability of 50%.

For the Frequentist estimates I use the glm-packages in R . The estimates are show in Table 3. The estimates are close to the coefficients of the data generating process. The $*$-symbol indicates that the estimates are significantly different from zero ($p < 0.05$). The Frequentist method correctly identifies the first four important β to be non-zero. All other parameters are correctly not significantly different from zero.

	Frequentist		Bayesian	
	MLE	std. error	$\mathbb{E}[\beta_i \vert y]$	std. dev.
β_0	0.00	0.04	0.00	0.05
β_1	0.99*	0.05	1.00*	0.04
β_2	0.80*	0.04	0.81*	0.04
β_3	-0.91*	0.04	-0.92*	0.04
β_4	-0.70*	0.04	-0.70*	0.04
β_5	0.04	0.04	0.04	0.04
β_6	-0.01	0.04	-0.01	0.04
β_7	-0.05	0.04	-0.05	0.04
β_8	0.05	0.04	0.05	0.04
β_9	0.04	0.04	0.04	0.04
β_{10}	-0.02	0.04	-0.02	0.04

Table 3: Frequentist and Bayesian estimates and variability of the first 11 parameters. For Frequentist the estimate is the maximum likelihood (MLE) and the variability the standard error (std. error). For Bayesian, the estimates are the expected values of the posterior ($\mathbb{E}[\beta_i|y]$ and the standard deviation of the posterior (std. dev.)

The Bayesian model needs a prior. For now choose the flat prior.

$$p(\beta_i) \propto 1, \quad \text{for } i \in \{0, ..., D\}$$

The flat prior is not a probability density distribution, because it does not integrate to one. A prior that does not integrate to one it is called an improper prior. Still the posterior can be proper probability density in case that the likelihood function integrates to a constant, which is the case in this example.

For the Bayesian method the result is a posterior distribution. Table 3 shows the expected values of the posteriors of the regression coefficients $\mathbb{E}[\beta_i|y]$. The results of the Bayesian and Frequentist methods are very similar. The $*$-symbol means that zero is not part of the 95% credible interval. So the Bayesian method also correctly identify the non-zero parameters. The expected value is a point estimate of the posterior distribution. Figure 2 shows the Monte Carlo samples of β_1 and β_2. These Monte Carlo samples represent the joint distribution of β_1 and β_2.

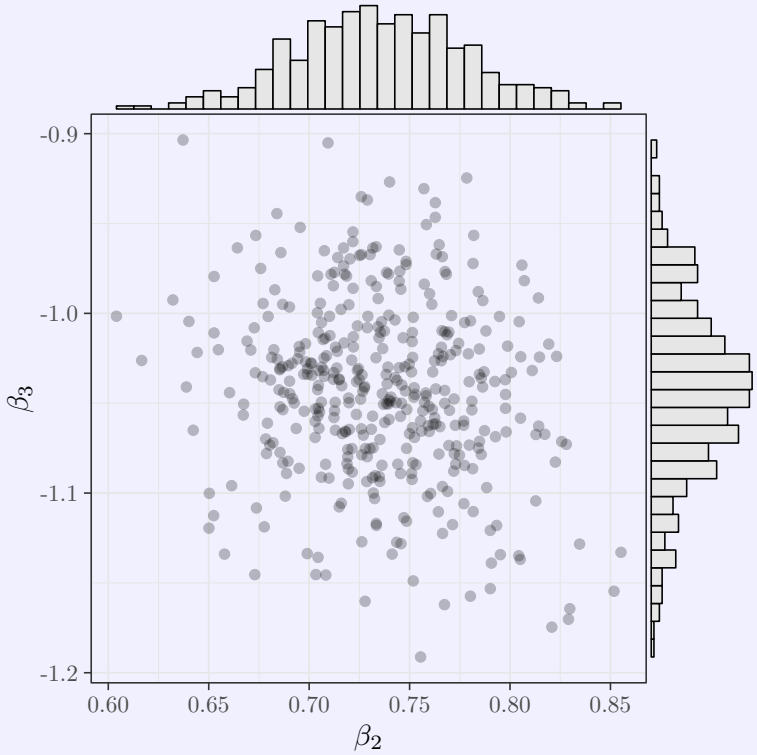

Figure 2: Dot plot of 400 Monte Carlo samples from independent data, the histograms represent the marginal posterior distributions of the β_1 and β_2.

Example 2.2: Logistic Regression on Collinear Explanatory Data

Take the same data generating process as in example 1, but now change the way the X variables are generated such that there is collinearity between some X variables. The explanatory variables X are drawn from a multivariate normal, with covariance matrix Σ.

$$X \sim MNV(\mathbf{0}, \Sigma)$$

With the following covariance matrix Σ:

$$\Sigma = \left[\begin{array}{ccccc|c} 1 & 0 & 0 & 0 & 0 & \\ 0 & 1 & 0.8 & 0 & 0 & \\ 0 & 0.8 & 1 & 0 & 0 & \mathbf{0}_{7\times 13} \\ 0 & 0 & 0 & 1 & -0.8 & \\ 0 & 0 & 0 & -0.80 & 1 & \\ \hline \multicolumn{5}{c|}{\mathbf{0}_{13\times 7}} & \mathbb{I}_{13} \end{array}\right] \tag{2}$$

Where $\mathbf{0}_{i\times j}$ is the zero matrix, with i rows and j columns. In this matrix the following relations are present:

- X^1 has a real effect and is uncorrelated with the remaining variables in X.
- X^2 and X^3 are correlated and both have a real effects
- X^4 has a real effect and is correlated with X^5, which has no real effect
- All other X are uncorrelated and have no influence on y.

The maximum likelihood and the expected value of the posterior distribution give similar results as the previous example.

	Frequentist		Bayesian	
	MLE	std. error	$\mathbb{E}_\theta[p(\theta\|y)]$	std. dev.
β_0	0.00	0.04	0.00	0.04
β_1	1.05*	0.05	1.06*	0.05
β_2	0.73*	0.6	0.73*	0.06
β_3	-1.00*	0.07	-1.00*	0.06
β_4	-0.76*	0.07	-0.77*	0.07
β_5	0.04	0.06	0.04	0.04
β_6	-0.05	0.04	-0.05	0.04
β_7	0.04	0.04	0.04	0.04
β_8	0.05	0.04	0.06	0.04
β_9	0.04	0.04	0.04	0.04
β_{10}	0.00	0.04	0.04	0.04

Table 4: Intercept and first ten regression coefficient

Figure 3 shows the joint distribution of β_2 and β_3. Even though the priors were independent, the posteriors are correlated. This can be seen in the sloping orientation of the Monte Carlo samples. The parameters that are not correlated have similar round shapes as depicted in figure 2. The correlation in the posteriors is a result of the collinearity in the data. The marginal distributions of correlated X variables are wider than for the uncorrelated X variables in Figure 2. However, the conditional distribution distribution, for example $p(\beta_2|\beta_3 = -1)$ is much narrower than the marginal distribution. So in case of collinearity, the marginal distribution can give a wrong impression of the uncertainty in the model. This phenomenon is also present in maximum likelihood estimate where the standard error of the estimates with correlated explanatory variables is about 50% larger, see table 4.

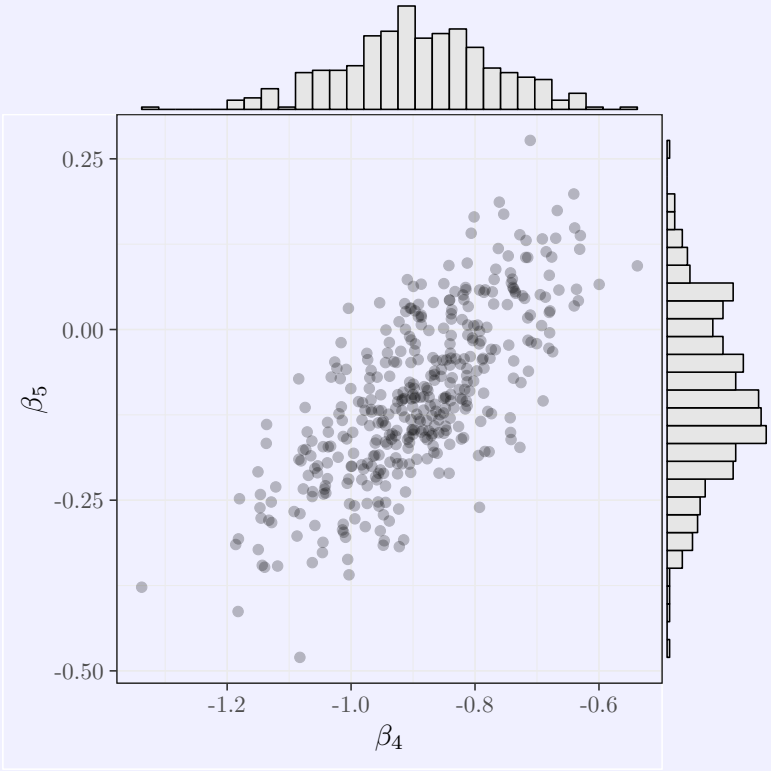

Figure 3: Dot plot of the Monte Carlo samples from correlated data. The Monte Carlo samples of the posteriors β_4 and β_5 are negatively correlated.

The Bayesian methods incorporate the correlation structure of the X variables, while the Frequentist methods do not do this directly.

2.3 Predictions

In Frequentist statistic prediction are made by using the point estimates $(\hat{\beta}_0, \hat{\beta})$ and the new explanatory data $\tilde{X}$. This is done by the logistic formula, and the result is a point prediction of Probability of Default $\tilde{\theta}_i$. Where the Probability of Default is given by:

$$\tilde{\theta}_i = \frac{\exp(\hat{\beta}_0 + \hat{\beta}\tilde{X}_i)}{1 + \exp(\hat{\beta}_0 + \hat{\beta}\tilde{X}_i)}$$

In Bayesian statistics the posteriors of the β_0 and β are not point estimates, but a distribution. From this distribution the posterior predictive distribution of the Probability of Default $p(\tilde{y}|\beta_0, \beta, X)$ can be calculated. Where $p(\beta_0, \beta|y, X)$ is the posterior after fitting the model.

$$p(\tilde{y}_i|y, X, \tilde{X}) = \int\int p(\tilde{y}|\tilde{X}, \beta_0, \beta)p(\beta_0, \beta|y, X)d\beta d\beta_0$$

For the logistic regression this gives:

$$p(\tilde{y}_i|y, X, \tilde{X}) = \int\int \left(\frac{\exp(\beta_0 + \beta\tilde{X}_i)}{1 + \exp(\beta_0 + \beta\tilde{X}_i)}\right) p(\beta_0, \beta|y, X)d\beta d\beta_0$$

The predictive probability is found by a Monte Carlo approximation:

$$p(\tilde{y}_i|y, X, \tilde{X}) \approx \frac{1}{S}\sum_{s=1}^{S} \frac{\exp(\beta_0^s + \beta^s\tilde{X}_i)}{1 + \exp(\beta_0^s + \beta^s\tilde{X}_i)}$$

In the Frequentist setting, the Probability of Default is a point prediction, whereas in the Bayesian setting the Probability of Default has a distribution. For this reason Bayesian methods take longer to compute, however from the Monte Carlo samples of the prediction $\tilde{y}^s$, uncertainty statistics, like standard deviation, can easily be calculated.

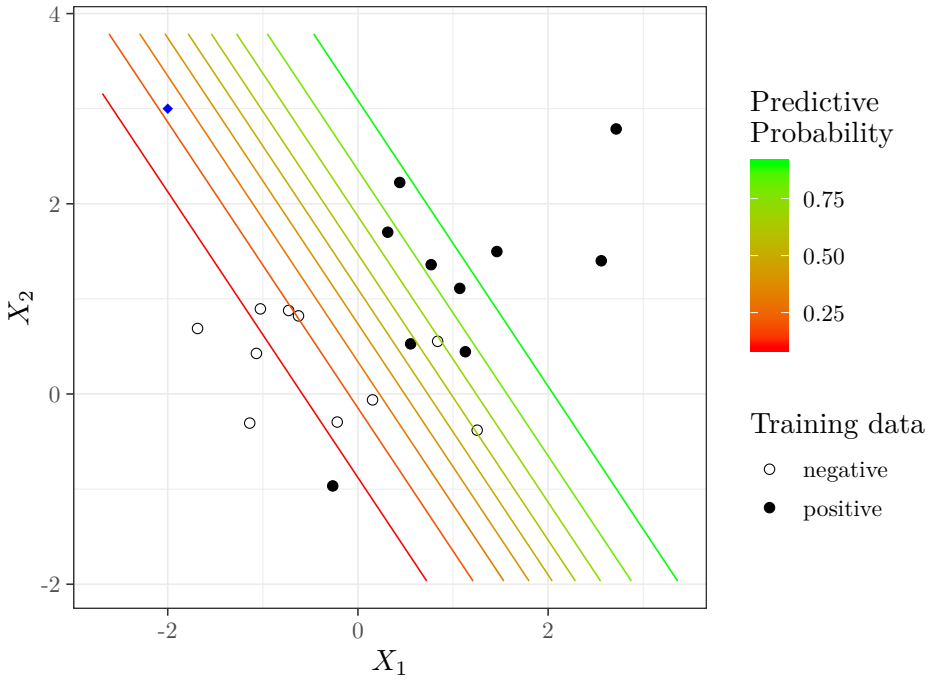

Figure 4: Predictive distribution of a Frequentist logistic regression

There is also a difference in the predictive values. To illustrate the difference between the Frequentist and Bayesian method, I use a schematic two dimensional example. Figure 4 shows an example of the predictive distribution of the Frequentist logistic regression. The model deems the lower left corner to be most likely to produce a negative outcomes (non defaults), and the upper right corner the most likely to give positive outcomes.

The predictions of the Frequentist logistic model are shown as straight lines for different levels of Probability of Default. The most left line represents a Probability of Default of 10% and the most right line gives a PD of 90%. The probability increases in steps of 10% per line. The blue diamond represents new explanatory data. For this new data point, the logistic regression makes a prediction for the Probability of Default. For the blue diamond this is approximately 22%.

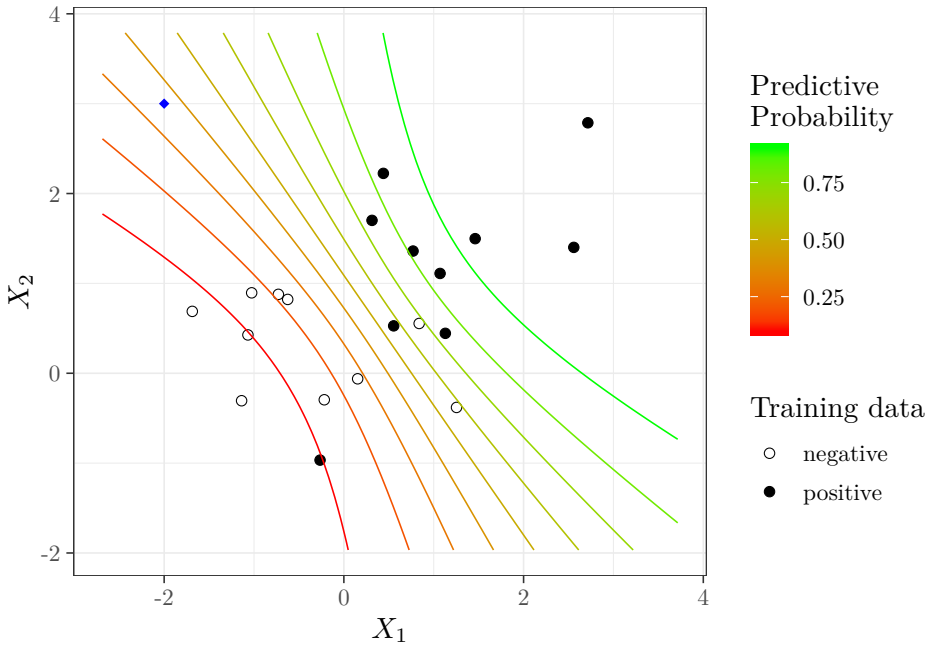

Figure 5: Posterior Predictive distribution of a Bayesian logistic regression

Figure 5 shows the Bayesian predictive probability for the same data as in figure 4. The uncertainty in the parameters β causes the Bayesian prediction to be less certain than the Frequentist predictions. The uncertainty of the parameters also cause the equal probability lines to be curved.

For every Monte Carlo sample of the posterior, a prediction can be made such that it has straight lines like in Figure 4. The exact slope changes per Monte Carlo sample as shown in Figure 6. The posterior predictive distribution is calculated by averaging the predicted value for all the Monte Carlo samples, which causes the curved lines in Figure 5.

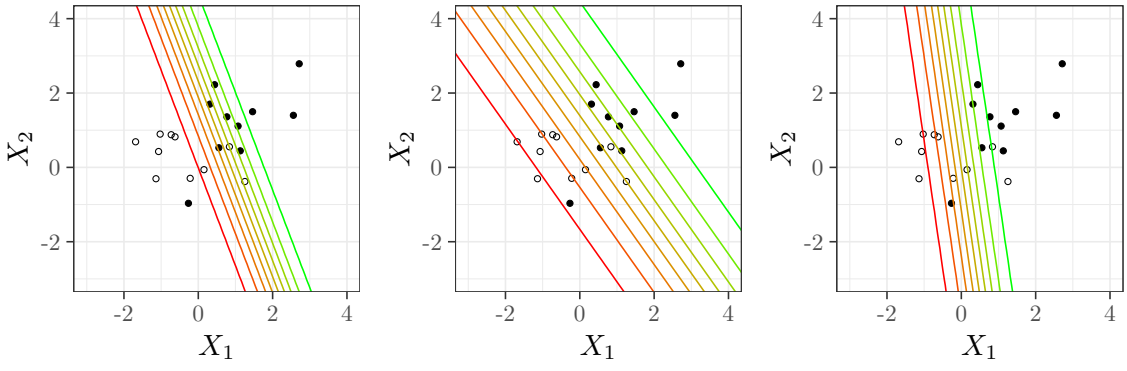

Figure 6: Posterior Predictive distribution of three different Monte Carlo samples of β

The Bayesian predictions have a tendency to be closer to 50% for new explanatory data, which is far away from the training data. For the blue diamond the predicted probability is 36%. This is about 50% higher than the Frequentist prediction.

2.4 Imbalanced Data

Data on defaults is often imbalanced. In a year most clients pay their amortisation and interest. Even though there can be thousands of observation, only in a fraction of these observation a default is observed. As the regression coefficients β explain the difference between clients that will and clients that will not go into default, the effective information on this relation is less then when the data would have equal occurrences of defaults and non-defaults.

The problem with the low amount of defaults in the data set is two fold. First of all, the prediction of the regression coefficients is more difficult than for balanced data. The regression coefficients represent a relation between explanatory variables X and response variables y. Especially when the amount of potential explanatory variables is high this can lead to overfitting as illustrated in Example 2.3.

Data Generating Process 1: Imbalanced Data with Independent Predictors

The data generating process is almost the same as in Example 1. The only difference is the intercept $\beta_0 = -5$, instead of $\beta_0 = 0$. This causes the data generating process to create less positive (defaults) outcomes. The different techniques throughout the thesis are applied, among others, to this data generating process.

Draw $n = 4,000$ observations form the following data generating process:

$$\text{logit}(\theta) = \beta_0 + \beta X$$

With the parameters β as in Table 5. Furthermore let:

$$y \sim \text{Bernoulli}(\theta)$$

The explanatory variables X are drawn i.i.d from a multivariate normal with a standard deviation $\sigma = 1$ and dimension $D = 20$:

$$X \sim MVN(0, \mathbb{I}_D)$$

	β_0	β_1	β_2	β_3	β_4	other β
Value	- 5	1	0.75	-1	-0.75	0

Table 5: Parameters of the data generating process

Example 2.3: Imbalanced Data with Independent Predictors

Using a Frequentist logistic regression the model with the balanced data has better predictions of the β than the imbalanced model. The L2-norm of the difference between the real coefficients β and the estimates $\hat{\beta}$ show the squared error of the estimates. And the L2-norm of the difference is:

$$||\beta - \hat{\beta}||_2 = \sqrt{\sum_{d=0}^{20} (\beta_d - \hat{\beta}_d)^2}$$

The L2-norm for the balanced data is 0.17 and for the imbalanced data this is 0.49. So the performance on balanced data is much better than that on the imbalanced data.

	MLE	std. error	$\mathbb{E}_\theta p(\theta\|y)$	std. dev.
β_0	-4.83*	0.20	-5.00*	0.20
β_1	0.83*	0.11	0.85*	0.11
β_2	0.65*	0.11	0.66*	0.11
β_3	-0.94*	0.11	-0.95*	0.11
β_4	-0.89*	0.11	-0.90*	0.11
β_5	-0.03	0.11	-0.03	0.11
β_6	0.10	0.11	0.10	0.11
β_7	0.22*	0.10	0.23*	0.10
β_8	-0.08	0.10	-0.08	0.10
β_9	-0.14	0.10	-0.13	0.11
β_{10}	0.04	0.10	0.04	0.10

Table 6: Estimate of the intercept and the first ten regression coefficients

Table 6 shows the estimates for the logistic regression. There is a clear increase in the standard error and the standard deviation compared to example 2.1. The regression coefficient β_7 is significantly ($p < 0.05$) different from zero, due to randomness.

Data Generating Process 2: Imbalanced Data with Collinear Predictor

Take the same data generating process as in example 2.3, but now change the way the X variables are generated such that there is collinearity between some X variables. The example data contains $D = 20$ variables, and $n = 4,000$ samples.

	β_0	β_1	β_2	β_3	β_4	other β
Value	- 5	1	0.75	-1	-0.75	0

Table 7: Parameters of the data generating process

The explanatory variables X are drawn from a multivariate normal, with covariance matrix Σ.

$$X \sim MNV(\mathbf{0}, \Sigma)$$

With the following covariance matrix Σ:

$$\Sigma = \left[\begin{array}{ccccc|c} 1 & 0 & 0 & 0 & 0 & \\ 0 & 1 & 0.8 & 0 & 0 & \\ 0 & 0.8 & 1 & 0 & 0 & \mathbf{0}_{7\times 13} \\ 0 & 0 & 0 & 1 & -0.8 & \\ 0 & 0 & 0 & -0.80 & 1 & \\ \hline \multicolumn{5}{c|}{\mathbf{0}_{13\times 7}} & \mathbb{I}_{13} \end{array}\right] \tag{3}$$

Example 2.4: Imbalanced data with Collinear Predictors

This data generating process is the go to data process for collinear data throughout the thesis. The L2-norm of the error of β is 0.57 for the imbalanced data and 0.17 for the balanced data. This error is more than 3 times as big for the imbalanced data.

	MLE	std. error	$\mathbb{E}[\beta\|y]$	std. dev.
β_0	-5.03*	0.22	-5.19*	0.22
β_1	0.86*	0.13	0.87*	0.13
β_2	0.87*	0.21	0.85*	0.22
β_3	-1.05*	0.22	-1.06*	0.22
β_4	-0.93*	0.21	-0.95*	0.21
β_5	-0.16	0.20	-0.18	0.21
β_6	0.03	0.12	0.03	0.13
β_7	0.00	0.13	0.00	0.13
β_8	0.11	0.13	0.11	0.12
β_9	-0.06	0.12	-0.06	0.12
β_{10}	-0.04	0.13	-0.04	0.13

Table 8: Estimates of the regression coefficients and their variability

The estimates and variability of the estimates are shown in table 8. The standard error and standard deviation of correlated parameters are almost 2 times as big as the uncorrelated parameters. Because 16 of the 20 variables do not have any predictive power these are preferably left out of the model. In this case, the 4 variables with a real relation are known. The second problem with the imbalanced data is that it is also more difficult to estimate the predictive performance of the model. These estimates of performance can have a high variance, which makes feature selection more difficult.

2.5 Final Remarks

Logistic regression is a model which assumes a log-odds relation between the response variable y and explanatory data X. The regression coefficients β express the strength of the relation.

In Frequentist statistics inference is done by finding point estimates of the regression coefficients β. Subsequently, these point estimates, in combination with new explanatory variable $\tilde{X}$, can be used to make predictions. For Frequentist statistics these are point predictions.

The posterior distribution is the result of inference in Bayesian statistics. Because the posterior often does not have an analytic solution, Monte Carlo methods are used to represent the posterior. This means that Bayesian methods are slower than Frequentist, but the posterior contains more information than the Frequentist point estimate. One example is that the posterior automatically contain correlation among its regression coefficients β, that results from collinearity in the explanatory variables X. For correlated explanatory variables this can be important, because the standard error in Frequentist statistics and the standard deviation of the posterior can give misleading high variability.

Imbalanced data makes it harder for the methods to find the estimates for the regression coefficient and the standard error and standard deviation of the posterior is higher than for balanced data. The regression coefficients represent the strength of the relation between the explanatory variables and the response variable. In imbalanced data there is less information on this relation, causing higher variance in the estimates.

3 Model Evaluation

To select variables, criteria are needed to define what a good model is. In this section I discuss model evaluation. This chapter is divided in two parts. The first part deals with the criteria on which to evaluate the model performance. The second part deals with calculating a value for the criteria. For Frequentist statistics this is cross validation. For Bayesian statistics this is PSIS-LOO, which is an approximation of leave-one-out cross validation.

3.1 Expected log predictive density

The goal of the model is to predict defaults in the upcoming year. I use the log loss of the predictive distribution as a measure of fit. When the model makes good predictions the expected log pointwise predictive density (elpd) for new data $(\tilde{y}, \tilde{X})$ has value near zero and a bad models has a big negative elpd. Elpd is defined as:

$$\text{elpd} = \sum_{n=1}^{N} \int p_t(\tilde{y}_i) \log p(\tilde{y}|\theta) d\tilde{y}$$

Where $p(\tilde{y}|\theta)$ refers to the posterior predictive distribution, and $p_t(\tilde{y})$ is the real data generating process. In the Frequentist case the θ is a point estimate, so the elpd becomes:

$$\text{elpd} = \sum_{n=1}^{N} p_t(\tilde{y}_i) \log p(\tilde{y}_i|\hat{\theta})$$

Elpd is founded in information theory, where it is related to the cross entropy (for Information Theory see Appendix B). For a single point the elpd is $\int p_t(\tilde{y}_i) \log p(\tilde{y}_i|\theta)$. This quantity contains two factors. The $\log p(\tilde{y}_i|\theta)$ factor is a the log loss of prediction. If the model perfectly predicts a event y_i happening, then the log loss is zero. On the other hand, when the likelihood goes to zero, the log loss goes to minus infinity. The log loss does not only consider whether the prediction is right or wrong, but also how sure the model is of the prediction. The second factor is $p_t(\tilde{y}_i)$, which is the Probability of Default under the data generating process. Even if the event is associated with a high log loss, if the probability of this event is very low, then it contributes little to the elpd.

The data generating process $p_t(\tilde{y})$ is unknown, and the goal of the model was to give a good approximation of this data generating process in the first place. This means that the elpd can not be calculated directly and needs to be approximated.

A quantity that is easy to calculate directly from the model is the log predictive density (lpd):

$$\text{lpd} = \sum_{i=1}^{N} \log p(y_i|y) = \sum_{i=1}^{N} \log \int p(y_i|\theta_{post}) p(\theta|y) d\theta$$

The problem with this quantity is that it uses the data set twice, first to fit the model and secondly to evaluate the performance. This means that the lpd overestimates the performance of the model and it has a bias towards the training data. Various techniques have be proposed to deal with this bias, some examples are Akaike Information Criterion (Akaike, 1998), in the Frequentist setting, and Watanabe-Akaike Information Criterion (Watanabe, 2010), in the Bayesian setting. These criterion have the following property:

$$\mathbb{E}_{\tilde{y}}[AIC] = \mathbb{E}_{\tilde{y}}[WAIC] = -2 \text{ elpd}$$

Instead of these information criteria, I use K-fold cross validation for Frequentist statistics (Section 3.2) and Pareto Smoothed Importance Sampling Leave-One-Out (PSIS-LOO) Cross Validation for the Bayesian models (Section 3.3). The advantage of these methods is that they have less assumption than AIC. For example, the that the prior has to be flat. PSIS-LOO also has some useful diagnostics, which the WAIC does not have. Besides elpd, I use the Mean Log Predictive Density (mlpd), which is the average of the elpd over n observations.

$$\text{mlpd} = \frac{\text{elpd}}{n}$$

When comparing the same data generating process, with different amount of data, elpd is hard to compare as this quantity scales linear with the amount of data. The mlpd does not have this problem. In this thesis, decision are based on the elpd and mlpd.

3.2 K-fold Cross Validation

Both the elpd and the precision-recall plot should be based on the out-of-sample fit to give a good representation of the performance of the model. A better approximation for the out-of-sample performance can be found by splitting up the data in K different sets. Then one of these sets is left out, and on the other $K-1$ sets the logistic regression is fitted. The set that has been left out did not influence the fit of the model. Therefore, the log predictive density (lpd) of the model on this set is actually an out-of-sample elpd. This process is repeated for all the K set and the sum of the lpd gives the $\text{elpd}_{\text{K-fold}}$.

$$\text{elpd}_{\text{K-fold}} = \sum_{k=1}^{K} \sum_{i \in I_k} \log p(y_i | y_{-I_k})$$

With I_k being the set of the indexes in the k-th fold. Such that $\{I_1, ..., I_K\} = \{\{1, ..., n_1\}, .., \{1, .., n_K\}\}$.

In this process the model needs to be fitted K times, which means that the computational time of K-fold cross validation takes about K times as long. The more folds there are the more accurate the model, but it also takes longer. I choose to have one fold for every 10 observation, because the speed of the Frequentist methods this is not a problem.

Example 3.1: K-fold

In this example I show that using K-fold cross validation is better approximation for the out-of-sample performance than using the log predictive density (lpd). For the model in Example 2.3 on page 24 the value of the lpd, K-fold elpd and the elpd on a data set with 100,000 observation are shown. Both the lpd and $\text{elpd}_{\text{k-fold}}$ overestimate the performance of the Frequentist model, considering that a value closer indicate a better performance. The error of the lpd is approximately four times as big as $\text{elpd}_{\text{k-fold}}$.

lpd	$\text{elpd}_{\text{k-fold}}$	$\text{elpd}_{\text{hold-out}}$
-281.83	-304.82	-312.59

Table 9: Estimated performance and hold-out performance

For the Frequentist model in Example 2 the lpd also overestimate the performance. In this case $\text{elpd}_{\text{k-fold}}$ underestimates the performance. The error of lpd is in this case two times as big compared to the error of $\text{elpd}_{\text{k-fold}}$ (Table 10).

lpd	$\text{elpd}_{\text{k-fold}}$	$\text{elpd}_{\text{hold-out}}$
-318.83	-340.06	-334.83

Table 10: Estimated performance and hold-out performance

The K-fold cross validation is not perfect, but the performance is much closer to the real performance of the model.

3.3 PSIS-LOO

K-fold cross validation can be a time consuming process in Bayesian statistic as every model has to be fitted K times, and as the Monte Carlo methods are slower than the optimisation algorithms used in Frequentist inference. Instead of this I use Pareto-Smoothed Importance Sampling Leave-One-Out Cross Validation (PSIS-LOO), this is approximation technique of the leave-one-out cross validation for Bayesian statistics, which is a fast methods with a number of diagnostics.

3.3.1 Leave-one-out cross validation

Taking the amount of folds K equal to the amount of data n, K-fold cross validation becomes leave-one-out cross validation (loo). By giving the response variable the following partition $y = \{y_i, y_{-i}\} = \{\{y_{-i}\}, \{y_1, ..., y_{i-1}, y_{i+1}, ..., y_n\}\}$, the leave-one-out estimate is:

$$\text{elpd}_{\text{loo}} = \sum_{i=1}^{n} \log p(y_i|y_{-i})$$

With:

$$p(y_i|y_{-i}) = \int p(y_i|\theta)p(\theta|y_{-i})d\theta$$

So this is the likelihood of a point y_i when the model is fitted on the data without point y_i.

3.3.2 Importance sampling

To get around refitting the model I use a technique called Importance sampling, which is a method to sample from one distribution while only having samples from another distribution. After fitting a single model on all data, the samples that are available are $p(y_i|y)$, however the samples of interest are $p(y_i|y_{-i})$. First of all, assume that the output variables y are conditionally independent given all Monte Carlo samples from the posterior θ^s (see appendix A for Monte Carlo methods), and using the partition as specified before the following result holds:

$$p(y|\theta^s) = p(y_i|\theta^s)p(y_{-i}|\theta^s)$$

And define the importance ratios as:

$$r_i^s = \frac{1}{p(y_i|\theta^s)} = \frac{p(y_{-i}|\theta^s)}{p(y|\theta^s)} \propto \frac{p(\theta|y_i)}{p(\theta|y)}$$

Using this ratio leave-one-out likelihood can be written as:

$$p(y_i|y_{-i}) \approx \frac{\sum_{s=1}^{S} r_i^s p(y_i|\theta^s)}{\sum_{s=1}^{S} r_i^s} \tag{4}$$

In this equation the ratios r_i^s with high values have a greater influence on the approximation, hence the name importance sampling. The previous equation can be simplified by plugging in the ratios.

$$p(y_i|y_{-i}) \approx \frac{1}{\frac{1}{S}\sum_{s=1}^{S} \frac{1}{p(y_i|\theta^s)}}$$

3.3.3 Pareto Smoothing

The raw importance ratios r_i^s can have a fat tail, resulting in high variability, especially in the domain with high values of the ratios. Combined with the fact that the high importance ratios have a relatively big impact on the final estimate, the estimate can have high variance as well. This in turn can cause the estimates to be an unreliable measure of the leave-one-out performance.

Because the importance ratios of the importance sampling can be highly variable, Vehtari et al. (2015) suggest smoothing the M largest importance ratios, by fitting a generalised Pareto

distribution and reweighing the largest M ratios with this distribution. Fitting a generalised Pareto distribution to the largest ratios is justified by the Pickard-Balkema-de Haan theorem.

Theorem 2 (Pickard-Balkema-de Haan theorem (Pickands III et al., 1975))**.** *For a sequence W_i which are identically and independently distributed and a threshold u, define the tail distribution as:*

$$f_u(w) := \frac{p(w)}{1 - P(w \leq u)}$$

The tail distribution converges to a Generalised Pareto Distribution as as the threshold u goes to infinity:

$$f_u(w) \to p_{gpd}(w|u, \sigma, k), \text{ as } u \to \infty$$

The probability density of the generalised Pareto distribution is:

$$p_{gpd}(w|u, \sigma, k) = \frac{1}{\sigma}\left(1 + k\left(\frac{w-u}{\sigma}\right)\right)^{-\frac{1}{k}-1}$$

The distribution has support on (u, ∞). In this case, set u equal to the smallest importance ratio of the M largest ratios with:

$$M = \min\left(\frac{S}{5}, 3\sqrt{S}\right)$$

Where S is the amount of Monte Carlo samples. This choice for M is based on numerical test (Vehtari et al., 2015). In figure 7 a generalised Pareto distribution is fitted to the M biggest importance ratios. k is a shape parameter and determines the shape of the tail, and σ is a scale parameter. Most of the importance ratios are near the threshold u, however there are importance ratios with values that 40 times higher than the threshold.

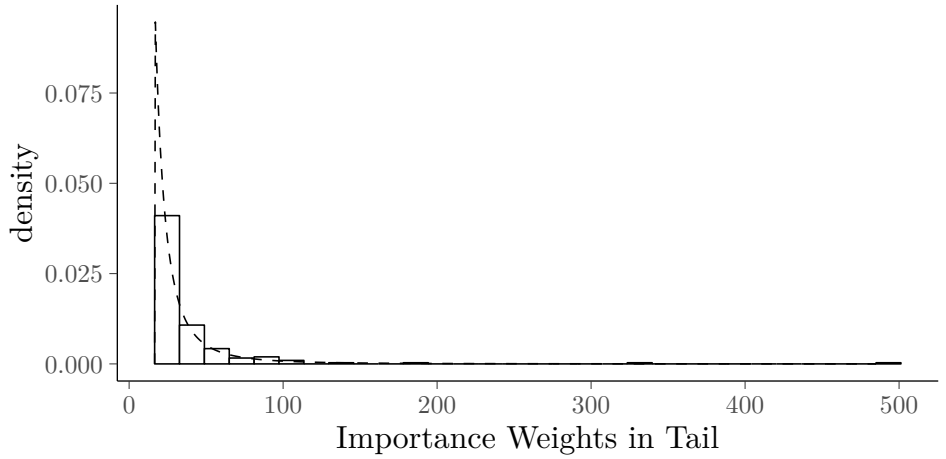

Figure 7: Fitted generalised Pareto distribution to the tail of the importance ratios as in figure 8, where $u = 16.57$, $k = 0.66$, and $\sigma = 9.81$

The fitted generalised Pareto distribution is used to smooth the M largest importance ratios to stabilise the estimate. The M largest ratios are replaced by the Pareto smoothed weight w_i^s, which are given by:

$$w_i^s = F_{\text{gpd}}^{-1}\left(\frac{z - 1/2}{M}\right)$$

F_{gpd}^{-1} is the inverse cumulative density function of the generalised Pareto distribution and $z = \{1, ..., M\}$, and M is the amount of ratios being used in the smoothing. The rest of the weights

are not smoothed and are used as raw ratio, so $w_i^s = r_i^s$. This gives the Pareto Smoothed version of equation 4.

$$p(y_i|y_{-i}) \approx \frac{\sum_s^S w_i^s p(y_i|\theta^s)}{\sum_s^S w_i^s}$$

And the Pareto Smoothed Importance Sampling Leave-one-out expected log posterior density (elpd_{psis}):

$$\text{elpd}_{\text{psis}} = \sum_{i=1}^{N} \log \left(\frac{\sum_{s=1}^S w_i^s p(y_i|\theta^s)}{\sum_{s=1}^S w_i^s} \right)$$

Besides approximating the elpd, $p(y_i|y_{-i})$ can be used as an input for measures like precision and recall. Which makes it possible to get an cross validation approximate for these measures.

The calculation of the elpd_{psis} only takes fraction of the time needed to fit the model itself. I often found that fitting a model with a Horseshoe prior could take in the order of tens of minutes. Using K-fold cross validation would take K times as long and using a real leave-one-out cross validation could take half a day. In my experience PSIS-LOO only takes a couple of minutes, making this a very useful tool.

Example 3.2: PSIS-LOO

For this example I introduce a new data generating process, which is has only one explanatory variable. This makes it possible to make a 2D plot and give insight into the mechanism of psis-loo.

$$\text{logit}(\theta) = -3 + 1.5x$$

with 20 data points $\{y_i, x_i\}$. After fitting the model, the likelihood $p(y_i|\theta_{post})$ can be calculated. By taking the inverse of the $p(y_i|\theta^s)$ we find the importance ratios. For a single y_i the ratios are plotted in figure 8.

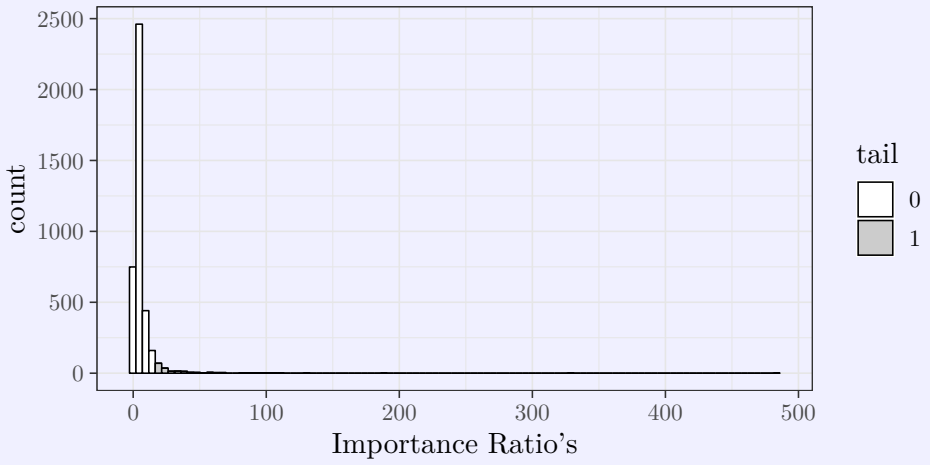

Figure 8: Importance ratios for y_i

The importance ratios in the tail are highly dependent on a particular realisation of the sample $p(y_i|\theta^s)$. This causes high variance in the tails of the distribution of the importance ratios.

In figure 9 the 20 y-values of given there corresponding x are plotted. The approximation of the importance sampling using raw importance ratio does not give a good approximation of the out-of-sample fit. However Pareto Smoothed Importance sampling gives a good approximation. This technique is discussed after the example. The log loss calculated via the raw importance sampling is 10% higher than the log loss calculated by PSIS-LOO as shown in Table 11.

lpd	Real elpd	PSIS elpd	Raw IS elpd
-1.05	-1.62	-1.68	-1.85

Table 11: Expected log predictive density for data point 19 in figure 9

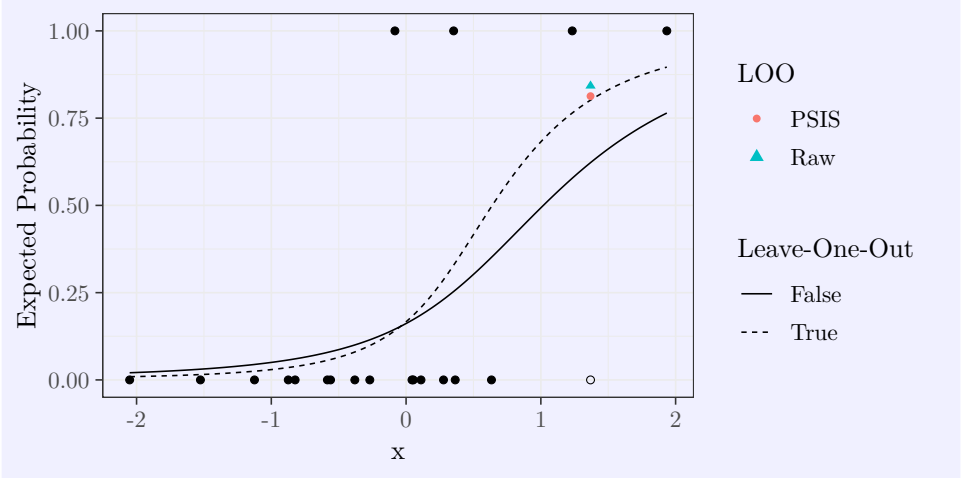

Figure 9: Logistic regression on 20 y-values given there corresponding x. The solid line corresponds to the expected probability of the posterior fitted on all 20 points. The dashed line corresponds to the model fitted on the data except the data point depicted as a hollow point. A good approximation of the out-of-sample performance for the hollow point should lie on the dashed line.

3.3.4 Diagnostics

When a model has a good fit to the data the PSIS-LOO likelihood $p(y_i|y_{-i})$ should have a value that is close to the likelihood of the entire model $p(y_i|\theta)$. When this is not the case a single data point $\{y_i, X_i\}$ has a big influence on the inference. This can be seen by rewriting the importance weights and using Bayes' formula:

$$r_i^s = \frac{1}{p(y_i|\theta^s)} = \frac{p(y_{-i}|\theta^s)}{p(y|\theta^s)} \propto \frac{p(\theta^s|y_{-i})}{p(\theta^s|y)}$$

In this case the raw importance ratios r_i^s can have very fat tails. When fitting the generalised Pareto distribution to the raw importance ratios a high tail shape parameter k will be found. The k-value in the generalised Pareto distribution corresponds to the fatness of the tail of the distribution. The amount of moments of the distribution that are defined, is always less then $\lfloor \frac{1}{k} \rfloor$, this meas that the variance is only defined when $k < 0.5$ and the mean of the distribution only exists when $k < 1$.

In case that the variance does not exist raw importance sampling does not converge to a solution, however the Pareto smoothed version still works for $k > 0.5$. Vehtari et al. (2015) find that if $k < 0.7$ that the PSIS-LOO is still reliable. In example 3.2 the Pareto distribution has a k-value of 0.56, so the raw importance sampling approximate gives a bad representation and the PSIS approximation is still good. A data point y_i has a k-value over 0.7 indicates that either the point is an outlier or the model is misspecified .

3.4 Final Remarks

In this thesis, I use a logarithmic loss to evaluate the performance of a model. The logarithmic loss is closely related to information theory, Where it is called the cross entropy of the model.

The goal is to find a quantity that predicts the logarithmic loss on unseen data. This quantity is the expected log predictive density (elpd). To compare data sets of different sizes I take the

mean of this value, which is the mean log predictive density (mlpd).

The logarithmic loss gives overly optimistic estimation of the performance when it is calculated on the training data.

K-fold cross validation gives a better estimate of the out-of-sample performance. The data is split up in K-fold. One fold is left out and the model is trained on the rest of the folds. Then the logarithmic performance is calculated on the left-out fold. This is repeated for all the folds, and summing the all the logarithmic losses give the elpd.

Bayesian model takes longer to fit, therefore, K-fold cross validation takes to long. Instead, I use Pareto smoothed importance sampling leave-one-out cross validation (psis-loo). This is an approximation of leave-one-out cross validation. The method takes less time than fitting a single Bayesian model. Furthermore, it is easy to detect if the method does not produce good approximations.

4 Frequentist Variable Selection

Out of the variable used in the logistic regression only a few might actually contribute to the performance of the model. The variables that do not add predictive power to the model are undesirable in the model. On the one hand, the extra variables make the model less interpretable. However, the more problematic fact is that the added variables can lead to overfitting of the model.

For a small dimensional problem a way to approach this problem would be to fit the model for all possible combinations of the variables and taking the model with the highest performance criterion. In the high dimensional case this would not be viable due to the exponential increase of possible model, namely the amount of fits would be 2^D. Where D is the dimension of the data. Furthermore, this approach has the problem that the performance criteria can have a high variance. This can lead to overfitting in the model selection. The regularisation based variable selection methods, do not have the same exponential growth with dimension, but has a polynomial growth depending on the method that is being used.

4.1 Forward Selection

The first type of variable selection considered is the Forward Selection. In the first step of this method a logistic regression is fitted with only an intercept β_0. In the second step of the algorithm, D different models with the intercept and one extra β_d are fitted.

$$\text{logit}(\theta) = \beta_0 + \beta_d X^d, \text{ for } d \in \{1, ..., D\}$$

Where X^d is the explanatory variable associated with parameter β_d. Then by using K-fold cross validation (Section 3.2) the parameter with the best predictive performance is chosen. Now in the subsequent steps one extra parameters per step is added, until all parameters are included in the model. After this is done the model with the highest predictive power is chosen. The choice of the submodel is based on the difference between the submodel with the highest elpd, the base model, and the smallest submodel for which the following relation holds.

$$P(\text{elpd}_{\text{base}} > \text{elpd}_{\text{submodel}}) < 0.84$$

This is the smallest submodel which is one standard deviation away from the model with the best predicted performance. This selection criterion is applied to all other variable selection methods.

Using this approach, instead of using all 2^D possible model, reduces the amount of steps needed to fit the model is $\sum_d^D (D - d) = \frac{D(D+1)}{2}$. In case of dimension $D = 20$, this heuristic reduces the complexity from 1,048,576 combinations to 210 combinations. Which decreases the time to fit the model by a factor of approximately 5,000. This method can also be used in Bayesian statistics, but I only consider the frequentist method, because fitting 210 model is in Bayesian statistics takes long time.

Example 4.1: Forward Selection

For data generating process 1, Forward Selection algorithm gives the output in Figure 10. In this case the method correctly identifies the amount of variables that model should have (namely four). The model performance is better on the out-of-sample data than the cross validation indicates. The grey line (corresponding to external set) has higher value than the black line (internal cross validation) . This has to do with the difference in amount of defaults that are in the data set. The default parameter y in the training data has an average of 2.25% defaults, while the external set has a average of 1.96%.

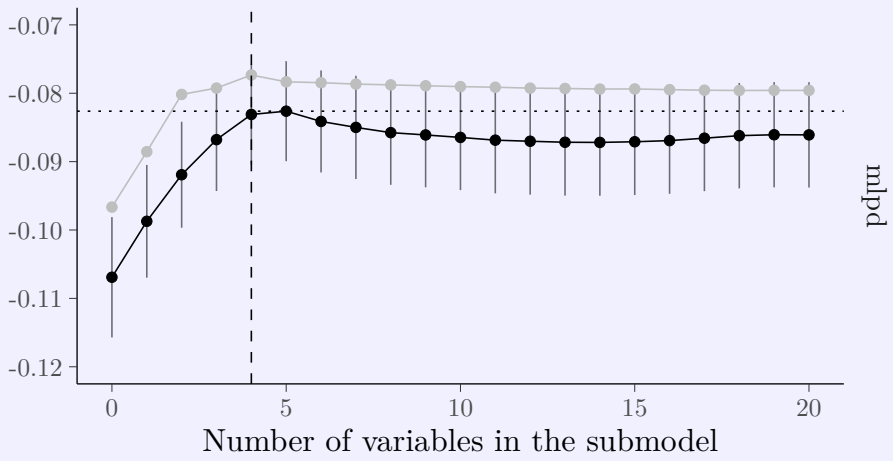

Figure 10: Variable selection when using Forward Selection. The black line is based on K-fold cross validation (in-sample) and the grey line is based on an external set. The dashed horizontal line is the suggested model size

4.2 Variable selection & Regularisation

When fitting a logistic regression with many variables and relatively low amounts of data, the regression has the tendency to start modelling noise, this phenomenon is called overfitting. This modelling of noise deteriorates the predictive performance of the regression.

When the regression models noise, the regression is too sensitive to the particular realisation of the data. If some data points would change a little, the inference could change a lot, which leads to unstable inference. Regularisation makes the model more resilient against overfittig, but still consider all the parameters in the model. Regularisation is done by restricting the freedom of the values that the parameters β can take. In Frequentist inference this is done by restricting a norm of the parameters β. In Bayesian statistics is done via the priors.

Furthermore, regularisation can be divided into two classes depending on the assumption on the regression coefficients β.

The first assumption is that all parameters β are important, thus all add predictive power to the model. In this case the parameter vector β is said to be dense. On the other hand the assumption can be made that the parameter vector contains parameters β_i which are equal to zero, in this case the parameter vector is sparse. Methods with this assumption have the tendency to set some parameters in β to zero or shrink some of them heavily to zero.

Techniques like Lasso are focused on sparse regression vectors β, where a part of the regression coefficients in the vector are zero.

Techniques like the ridge regression assume that all variables β have some influence on the predictive power of default, hence all $\beta \neq 0$. These techniques typically do not set parameters to zero.

The different types regularisation do not necessarily lead to variable selection, but can be used in variable selection methods. In particular the techniques which assume sparsity are useful for variable selection, as they produce a ranking of importance for the variables. The methods that do not induce sparsity, do not produce such a ranking.

4.2.1 Ridge Regression

In Frequentist regularisation, a penalty is given to the use of bigger parameters β. These techniques are only applied to the regression coefficients β and not to the intercept β_0. As the model with only β_0 is seen as the base model and we want to keep the intercept in the model after variable selection. One type of regularisation in the Frequentist framework is ridge regression, which is

this case is a logistic regression where a $L2$-norm penalty is added. The estimate for the ridge regression $\hat{\beta}_{ridge}$ are found by the following equation.

$$\underset{\beta_0,\beta\in\mathbb{R}^{D+1}}{\operatorname{argmin}} \{-\log p(y|\beta_0,\beta,X)\} \text{ subjected to } ||\beta||_2^2 \leq t \tag{5}$$

So ridge regression searches for the highest likelihood for values of β that are in a sphere with radius t. Figure 11 shows a schematic example for a two dimensional ridge regression. The maximum likelihood is represented by a dot, and the concentric ellipsoid represent the different levels of the likelihood function. The value of the ridge estimate $\hat{\beta}_{ridge}$ is represented by the plus symbol.

Equation 5 is often solved via the Lagrangian from of the ridge regression, which is shown in Equation 6. This form can has a derivative. This derivative is used in the algorithm coordinate descent, which finds the solution to β_0 and β.

$$\underset{\beta_0,\beta\in\mathbb{R}^{D+1}}{\operatorname{argmin}} \left\{-\log p(y|\beta_0,\beta,X) + \lambda||\beta||_2^2\right\} \tag{6}$$

The exact relation between λ and t depends on the realisation data and there is no nice way to represent this relation. There is not clear a priori way to pick λ, instead multiple different values of λ are used and the value with the highest elpd is chosen.

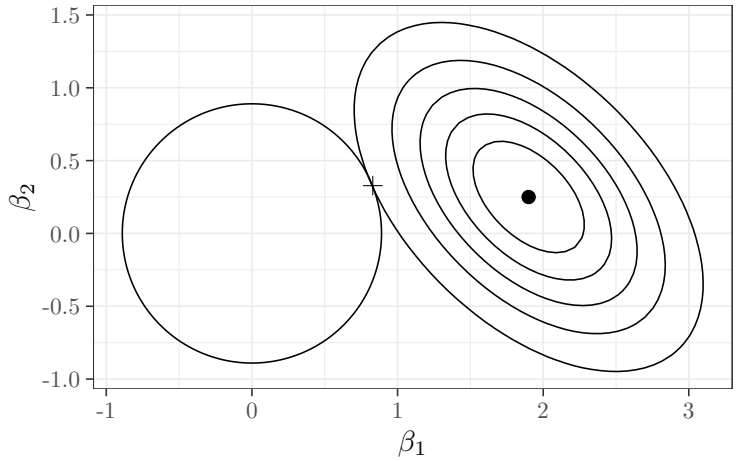

Figure 11: Schematic representation of ridge regression. The circle represent the constrain as in equation 5. The ellipsoids represent different levels of the likelihood. The dot is the maximum likelihood estimate and the plus symbol is the ridge estimate.

Ridge regression does not induce sparsity, because of the smooth geometries, and does not give a ranking of the importance of the different variables. Making it unsuitable for variables selection. However, it can still counteract overfitting, this can also be seen in Example 4.2.

Example 4.2: Ridge Regression

The data in this example is a little different than the normal data from data generating process 1. Here we take only $n = 1,000$ samples instead of 4,000 samples from the data generating process. The deviation from the standard is because the effects are more pronounced than in the standard example data from the data generating process.

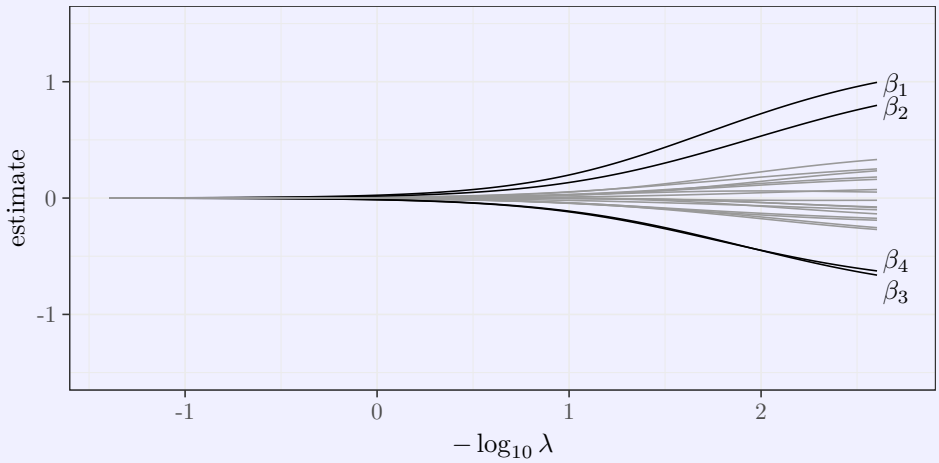

Figure 12: Ridge Regression for different levels of regularisation λ, the regularisation is weakest on the right hand side and strongest on the left hand side. The estimate of the get closer the zero as the regularisation gets stronger.

Figure 12 show the ridge estimate for various values of the parameter λ, the ridge regression does not induce sparsity and it shrinks all parameters towards zero, but never puts them on zero. There are two opposite effects in this regularisation, on the one hand the prediction get better because the noise, originating from the 16 zero-valued parameters, get shrunk towards zero. On the other hand, the prediction get worse, because the effect of the four important β also get shrunk. The regression does somewhat improve the model as can be seen in figure 13 at a regularisation coefficient $\lambda \approx 0.01$. This is point where the two opposite effects are equal.

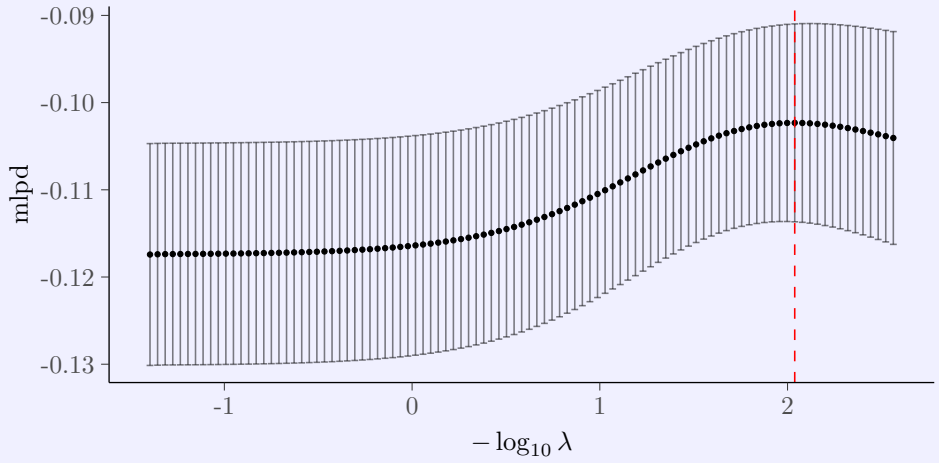

Figure 13: Performance of the Ridge Regression for different levels of regularisation λ.

Table 12 contains the first ten regression coefficients of the ridge regression. The other 10

variables are all similar to β_5 to β_{10}.

β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
0.85	0.42	-0.73	-0.39	0.02	0.13	-0.24	-0.16	-0.04	-0.20

Table 12: Estimates of the regression coefficients for the Ridge Regression with the highest mlpd

4.2.2 Lasso Regression

A frequentist method to induce sparsity is Least Absolute Shrinkage and Selection Operator (Lasso) regression. Lasso regression penalises the $L1$-norm of the regression coefficients β (Tibshirani, 1996), where the $L1$-norm is the sum of the absolute values of the parameters β:

$$||\beta||_1 = \sum_{i=1}^{D} |\beta_i|$$

And the Lasso regression is defined as:

$$\underset{\beta_0, \beta \in \mathbb{R}^{D+1}}{\operatorname{argmin}} \{-\log p(y|\beta_0, \beta, X)\} \text{ subjected to } ||\beta||_1 \leq t \tag{7}$$

The Lagrangian form of Lasso, used in computing the Lasso regression is equation 8.

$$\underset{\beta_0, \beta \in \mathbb{R}^{D+1}}{\operatorname{argmin}} \{-\log p(y|\beta_0, \beta, X) + \lambda||\beta||_1\} \tag{8}$$

Solving for λ in equation 8 and t in equation 7 have is the same. Equation 8 is the Lagragian form of Equation 7. The exact relation between λ and t depends on the realisation data.

In this equations there are to two terms, which have an opposite effect in the minimisation. The fist term of the equations $-\log p(y|\beta_0, \beta, X)$ is the minus loglikelihood of the logistic model. Minimising this term is equivalent to maximising the likelihood. In general the likelihood gets higher the more parameters get added to the model, as the model becomes more flexible and it can more easily fit the data points.

The second term $\lambda||\beta||_1$ has the opposite effect. This is the $L1$-norm of the parameters. For larger β, this term becomes bigger.

Figure 14 shows a schematic representation of a two dimensional Lasso regression. The constraints are represented by the diamond. Due to the perpendicular corners of the constrains, the equal-likelihood ellipsoids are likely to be maximised in the corners. The corners are zero for at least one regression coefficient β, thereby inducing sparsity in the regression vector β. The likelihood of the estimate of the Lasso regression and the likelihood of the Ridge estimate (Figure 11) are the same.

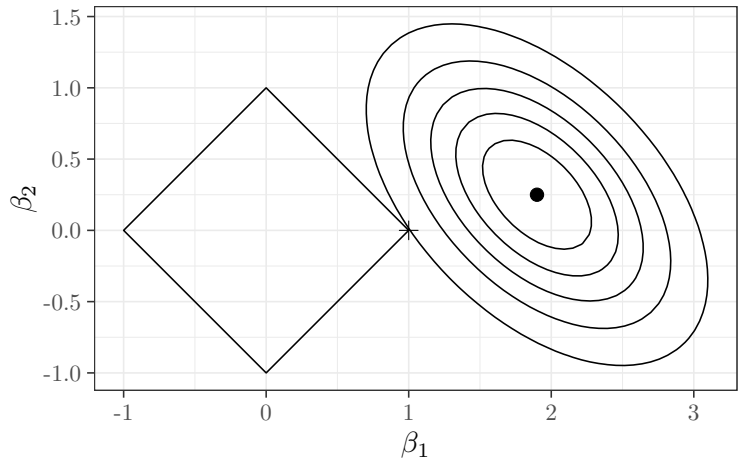

Figure 14: Schematic representation of Lasso regression. The diamond represent the constrain as in equation 7, where $t = 1$. The ellipsoids represent different levels of the likelihood. The dot is the maximum likelihood estimate and the plus symbol is the Lasso estimate.

Example 4.3: Lasso Regression on Independent Data

Fit a Lasso regression for the example data with 4,000 observations from the data generating process 1.

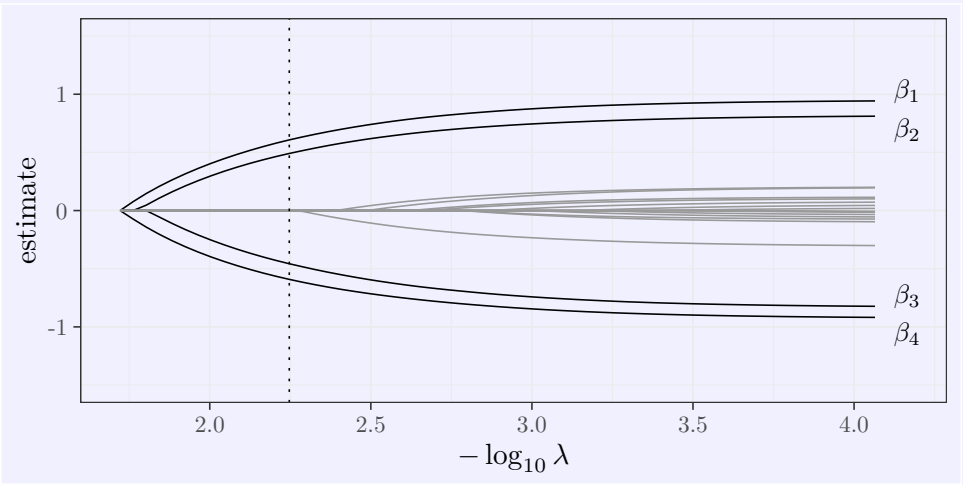

Figure 15: Lasso regression for different values of λ, the dotted line is the model where the 4 non zero β are the only parameters in the model

The estimates when using a Lasso regression are shown in Figure 15. As λ get bigger ($-\log_{10}\lambda$ gets smaller) the regularisation effect gets stronger. This corresponds to a smaller 20 dimensional diamond (Figure 14). The four parameters with a real effect are labelled, the other 16 parameters are not labelled. On the left side the regularisation is the strongest and on the right side the weakest. As the regularisation gets stronger more and more variables are set to zero. The parameters β that do not have a real effect are set to zero, which is desirable. However as the regularisation continues to get stronger even the real parameters are estimated to be zero.

The vertical line represent the least regularised estimate where only the four important parameters are included.

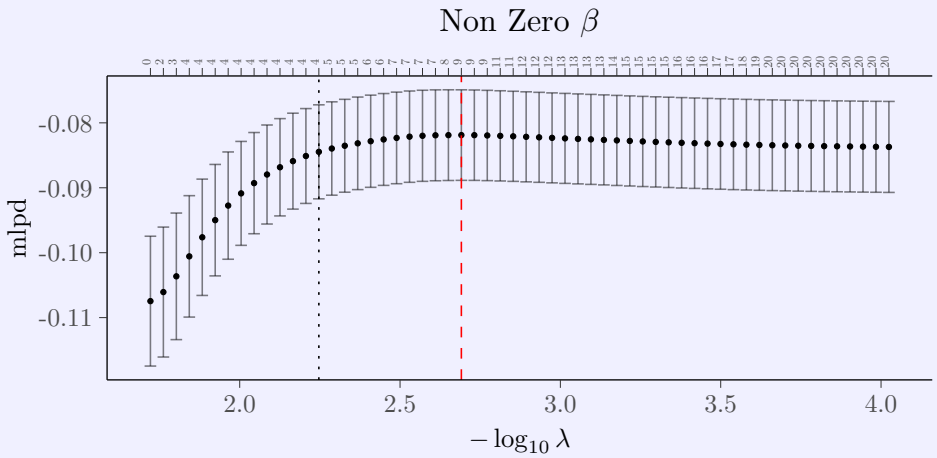

Figure 16: Cross Validation of the Lasso Regression with different values of λ. The top axis shows the amount of non-zero parameters β. The black dots are the estimates and the bars around them is the standard deviation of the estimate. The black dotted line is the model with the correct amount of β, the red dashed line is the model with the best predictive value

In Figure 16 the results of cross validation are shown for a Lasso regression with multiple values regularisation values of λ. On the left hand side the regularisation is the weakest and on the right hand side the strongest. When increasing the regularisation the performance of the model becomes better, until the regularisation shrinks the non-zero β to much and the performance deteriorates.

Table 13 shows the estimates of parameters of the model with the highest elpd and the model only containing the four important regression coefficients. When Lasso only includes the four parameters, the important regression coefficients are heavily biased towards zero. The model performs better when more variables are included. In the highest elpd model the important parameters are much closer to their true value. The unimportant included variables are relatively small, for example $\beta_7 = 0.08$ and $\beta_9 = 0.15$. This means that these parameters do not have a big influence on the Probability of Default.

	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
four important	0.61	0.49	-0.46	-0.59	0.00	0.00	0.00	0.00	0.00	0.00
highest elpd	0.78	0.66	-0.64	-0.75	0.00	0.00	0.08	0.00	-0.15	0.00

Table 13: Estimates of the regression coefficients

Example 4.4: Lasso with Collinearity in Data

Now the same is done with the collinear data as in data generating process 2.4 (p. 26) and the results are shown in Figure 17. The regression coefficient with correlated data have the tendency to go faster to zero than the coefficients without correlated data. For example parameter β_3, which has a correlation with β_2, goes quickly towards zero, until $\beta_2 = 0$, and then its descent goes slower again. The same happens for β_4, which is correlated with β_5. In this example the Lasso regression does not correctly identify the most important parameters. Parameter β_2 is the seventh parameter to be kept in the model, this meas that if all relevant parameters are included, that there are 3 noisy parameters in the model.

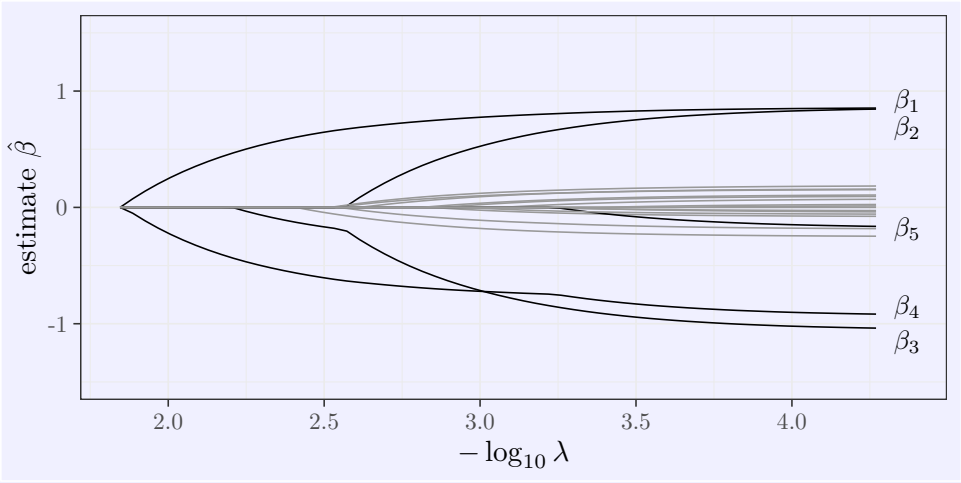

Figure 17: Lasso regression on data with collinearity, the four real contributing β are shown in black. The regression coefficient β_5, which correspond to the explanatory X^5 that is correlated with X^4, is also shown in black. All other β are shown in grey.

Figure 18 shows the cross validation performance estimate of the Lasso regression. Again the Lasso regression picks a model that is too large. The selected model is smaller than in the case of independent data, however the performance of the model is worse than that of the independent data.

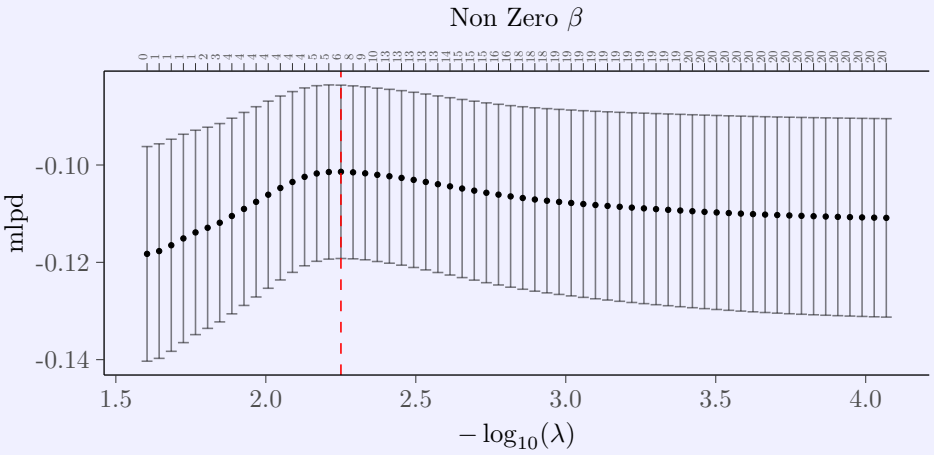

Figure 18: Caption

In this case the Lasso regression overestimates the amount of true variables in the model. It should be four nonzero parameters, but it picks six nonzero parameters.

4.2.3 Relaxed Lasso

The Lasso could be used directly, however it is also be possible to just use the Lasso regression as a way to find the most important variables. As the regularisation get stronger more and more regression coefficients are set to zero one-by-one. This gives a rank of the regression coefficients. The most important regression coefficient needs the most regularisation to be set to zero, and the least important regression coefficients is set to zero with the least amount of regularisation.

Relaxed Lasso is a procedure where the model is refit without regularisation. Where the first refit is done on the most important variable, the second model on the two most important

variables, and so forth. So after the Lasso regression the model is fitted D (amount of variables) times and from these D models the best model is chosen.

For the model with the real four parameters, the real contributing parameters β in Figure 15 are shrunken to zero. If the Lasso regression correctly identifies the rank of the regression coefficients, then refitting the model would lead to a better estimate of the true parameters β. In general this lead to a sparser model.

However, the Lasso regression does not always identify the the right order of the parameters as can be seen in Figure 17. In this case, noisy parameters β are added to the model without any regularisation. This can deteriorate the predictive performance of the model. So the performance of Relaxed Lasso is dependent on the discriminatory power of the Lasso regression.

Example 4.5: Relaxed Lasso

The Lasso Regression in example 4.3 gives the parameters β a ranking, which is shown in table 14.

Lasso Rank	1	2	3	4	5	6	7	8	9	10
$i : \beta_i$	1	4	2	3	9	7	15	10	5	6

Table 14: Importance rank given by Lasso regression

Picking the λ with the highest value and refitting the model without regularisation.

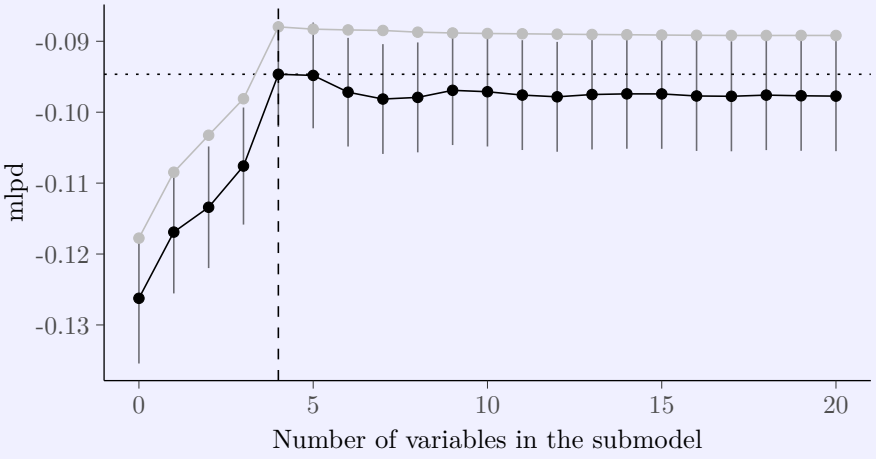

Figure 19: Relaxed Lasso variable selection

4.3 Final Remarks

Frequentist methods are quick and can solve the problem in a couple of seconds. Forward Selection

Ridge regression is not a good option for variable selection, but it is still applicable to counteract overfitting. Lasso and Relaxed Lasso are both the quickest methods for variable selection. Lasso shrinks also shrinks the important variables, when only the important variables are included in the model. Therefore, the model with the highest mlpd is has more variables in the model. Relaxed Lasso refits the models for the rank given by Lasso. This causes the important variables to be shrunk less. The risk of this approach is that if the Lasso rank is not correct that a noisy feature is included without any regularisation.

All the models are also easy to implement, because they are easily available in packages in most popular programming languages.

5 Bayesian Variable Selection

Just like in the Frequentist case, there are many ways to approach variable selection in Bayesian statistics.

The spike-and-slab prior (Mitchell & Beauchamp, 1988) is considered to be the gold standard in Bayesian variable selection. This is a mixture of a spike on zero, and a very wide slab, for example a Cauchy(0,1).

$$\gamma_i \sim \text{Bernoulli}(\eta)$$
$$p(\beta_i) = (1-\gamma_i)\delta_0(\beta_i) + \gamma_i \frac{1}{\pi(1+\beta_i^2)}$$

Where δ_0 is a dirac delta on zero and η indicates the a priori sparsity of the model. If $\gamma_i = 1$ then the prior on β is the slab and if $\gamma_i = 0$, then the prior on β is the spike. The posterior of the γ_i expresses the belief whether the variable should be included in the model. So when the γ_i is small it should not be included and when γ_i is big is should be included.

A main concern in Bayesian statistics is computing the posterior distribution. As discussed, Markov Chain Monte Carlo sampler are the common way to solve non-conjugate problems. Traditional solvers like Metropolis-Hastings and the Gibbs Sampler are too slow in high dimensional space and for correlated data. Algorithm that are the only real option for high dimension and correlated data use the gradient of the logarithm of the posterior. This is not defined for the spike-and-slab prior and are therefore not applicable to the spike-and-slab prior.

In the Bayesian setting it is also possible to get variable selection via regularisation, and very similar results can be found in Frequentist and Bayesian statistics. To continue building on this understanding, I consider the Horseshoe prior as a type of regularisation. This Horseshoe prior is seen as a continuous equivalent of the spike-and-slab prior, and it can be solved by Hamiltonian Monte Carlo.

5.1 Prior choice on Intercept

In Bayesian logistic regression the logistic function the same as in Frequentist statistics. However, in Bayesian statistics a prior should be defined before the model can be fitted. On the regression coefficients β special priors are applied such that it can be used for variable selection. We always want to keep the intercept in the model as it a measure of the average default probability. Therefore, these priors are not being used on the intercept, but still prior for the intercept needs to be chosen.

Sometimes the Normal distribution is suggested as a prior for the logistic regression. However, the normal distribution is prone to outliers (O'Hagan, 1979). This means that a single outlier can heavily influence the posterior distribution. This makes the model non-robust and this is generally true for light tailed prior distributions. To prevent the outlier sensitivity Gelman et al. (2008) proposed using a Cauchy distribution as a prior. The fat tails of the Cauchy distribution limit the effect of outliers on the inference of β_0.

The Cauchy distribution has very fat tails, the tails are so fat that the expected value of the Cauchy distribution is not defined. If a combination of explanatory variables X is fully predictive of defaults, the logistic regression is called separable. In this case the likelihood function becomes almost flat from a certain threshold till infinity. This causes the Monte Carlo to linger on in this area. This can be detrimental to the performance of MCMC (Ghosh et al., 2018). For the intercept β_0 the use of a student-t distribution with a degree of freedom between 3 and 7 is recommended (Gelman et al., 2013). This prior gives a more robust solution than the normal prior and is numerically more attractive than the Cauchy prior. As the flat prior $p(\beta) \propto 1$, is even more diffuse than the Cauchy distribution, it is even less numerical stable than the Cauchy distribution.

Normally the student-t is a good choice for the intercept β_0 and regression coefficients β. In this thesis this prior is only applied to the intercept β_0, and not to the regression coefficients β. The priors on β are used for variable selection. However, in general the tails of the priors on β have a similar effect on numerical stability of the Monte Carlo methods.

5.2 Bayesian Regularisation

In Frequentist regularisation an penalising term is added to the likelihood to penalise larger regression coefficient β. In Bayesian regularisation no extra terms are added, but regularisation is done via the prior. In case of the normal prior and the Laplace prior, the results are very similar to that of Ridge regression and Lasso regression respectively, and for the Maximum a posteriori point estimate the results are actually the same. Besides these prior, a more elaborate Horseshoe prior can be used, which has the property that it filters out weak signals, while leaving strong signal almost unchanged. The Normal prior and the Laplace prior are treated to give an intuition on Bayesian regularisation and to show how the different types of statistics can have similar results. The Horseshoe prior is the only Bayesian technique is applied in the simulation studies and to the FreddieMac data.

5.2.1 Normal Prior

In Bayesian statistics the posterior distribution is proportional the prior $p(\beta)$ times the likelihood function $p(y|\beta_0, \beta, X)$.

$$p(\beta_0, \beta|y, X) \propto p(y|\beta_0, \beta, X)p(\beta)p(\beta_0)$$

In Bayesian statistics the point which maximises the posterior distribution is the Maximum a-posteriori (MAP) estimate. When taking the flat prior $p(\beta_i) \propto 1$, the posterior is proportional to the likelihood function. For the flat prior the MAP estimate is the same as the maximum likelihood estimate in frequentist statistics. I only use the MAP estimate as an illustrative tool, and not for prediction as it loses information from the posterior distribution. For example the variance of the posterior is lost.

If a normal prior is set on the regression coefficients β,

$$\beta_i \sim \text{Normal}(0, \sigma)$$

and a mean of 0 with a small standard deviation σ is chosen, then the prior pulls the posterior away from the maximum likelihood estimate towards 0. This phenomenon is called shrinkage.

To show that Bayesian Regression with a normal prior gives similar results to the Ridge regression, let the model be as follows:

$$\begin{aligned} y &\sim \text{Bernoulli}(\theta) \\ \text{logit}(\theta) &= \beta_0 + \beta X \\ \beta_i &\overset{iid}{\sim} N(0, \sigma_0) \\ \beta_0 &\propto 1 \end{aligned}$$

This model has the following unnormalised posterior:

$$\begin{aligned} p(\beta|y, X) &\propto p(y|\beta_0, \beta, X)p(\beta)p(\beta_0) \\ &\propto p(y|\beta_0, \beta, X) \exp\left(-\frac{||\beta||_2^2}{2\sigma_0^2}\right) \end{aligned}$$

The maximum a-posteriori (MAP) estimate can be found by using the logarithm and then maximising that:

$$\begin{aligned} \hat{\beta}^{\text{MAP}} &= \underset{\beta_0, \beta \in \mathbb{R}^{D+1}}{\text{argmax}} \left\{ \log p(y|\beta_0, \beta, X) - \frac{||\beta||^2}{2\sigma_0^2} \right\} \\ &= \underset{\beta_0, \beta \in \mathbb{R}^{D+1}}{\text{argmin}} \left\{ -\log p(y|\beta_0, \beta, X) + \frac{1}{2\sigma_0^2}||\beta||^2 \right\} \end{aligned}$$

The MAP estimate is the same as in the ridge estimate in equation 6 on page 36, when $\lambda_{\text{Lasso}} = \frac{1}{2\sigma_0^2}$.

5.2.2 Laplace Prior

The Bayesian equivalent of the Lasso regression is using a Laplace prior on the regression coefficient.

$$\beta_i \sim \text{Laplace}(\mu, \lambda)$$

Where $\mu = 0$. The Laplace distribution has the following distribution:

$$p(\beta_i|\mu, \lambda) = \frac{1}{2\lambda} \exp\left(-\frac{|\beta_i - \mu|}{\lambda}\right)$$

Figure 20 depicts the probability density function of the Laplace distribution.

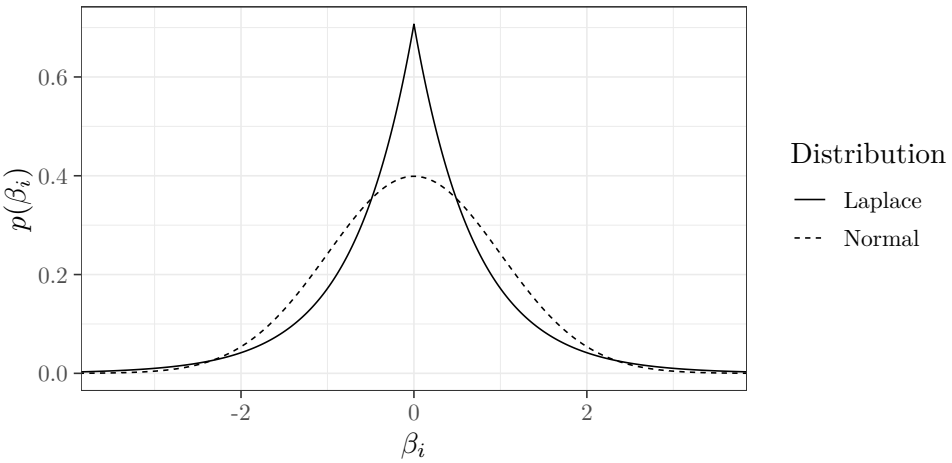

Figure 20: Laplace distribution with $\lambda = \sqrt{0.5}$ and $\mu = 0$ and Normal distribution with $\sigma = 1$ and $\mu = 0$. The distributions have the same mean and variance

The Laplace prior puts more weight around 0 compared to the normal distribution. Weak signals, that are β_i for which the maximum likelihood are close to 0, will get more mass on zero compared to the normal prior. Furthermore, the Laplace distribution has thicker tails than the normal distribution, which means that strong signals, β_i far from zero, are shrunk less than with the normal distribution.

To make the comparison between Lasso regression and Bayesian regression with a Laplace prior, define the model:

$$\begin{aligned} y &\sim \text{Bernoulli}(\theta) \\ logit(\theta) &= \beta_0 + \beta X \\ \beta_i &\overset{iid}{\sim} \text{Laplace}(0, \lambda) \\ \beta_0 &\propto 1 \end{aligned}$$

The MAP estimate for this model is:

$$\begin{aligned} \hat{\beta}^{\text{MAP}} &= \underset{\beta_0, \beta \in \mathbb{R}^D}{\text{argmax}} \left\{ p(y|\beta_0, \beta, X) \exp\left(-\frac{||\beta||_1}{\lambda}\right) \right\} \\ &= \underset{\beta_0, \beta \in \mathbb{R}^D}{\text{argmax}} \left\{ \log p(y|\beta_0, \beta, X) - \frac{||\beta||_1}{\lambda} \right\} \\ &= \underset{\beta_0, \beta \in \mathbb{R}^D}{\text{argmin}} \left\{ -\log p(y|\beta_0, \beta, X) + \frac{1}{\lambda}||\beta||_1 \right\} \end{aligned}$$

So the Lasso regression and the Bayesian regression with a Laplace prior are equivalent, only the λ of the Laplace prior is the inverse of the λ_{Lasso} of the Lasso regression (equation 8 on page 38).

5.2.3 Horseshoe Prior

The Laplace or a normal prior with give similar results to respectively the Lasso regression and ridge regression. These methods shrink parameters β to zero depending on a common shrinkage parameter λ for the Laplace prior and the normal prior. The shrinkage parameters of these prior are the same for all parameters β, so they have a global effect. This causes these methods to shrink all regression coefficients towards zero, even the strong ones.

Carvalho et al. (2010) introduces the Horseshoe prior, which has a local shrinkage prior λ, besides global shrinkage parameter τ . The prior is constructed as follows:

$$\begin{aligned}\beta_i &\sim \text{Normal}(0, \lambda_i \tau) \\ \lambda_i &\overset{iid}{\sim} \text{Cauchy}^+(0, 1) \\ \tau &\sim p(\tau_0)\end{aligned}$$

The global shrinkage parameter τ is the same for all β, if τ is small, then it will shrink all parameters towards 0, and as τ goes to infinity, all parameters are unregularised. The choice of τ_0 depends on the prior assumption on the sparsity of the parameter vector β. This is often a small value with $\tau < 1$, the precise choice of τ_0 is discussed later in this section.

Whereas the global shrinkage parameter τ shrinks all parameters towards zero, the local shrinkage parameter λ has the opposite effect for some β. There is only one τ for all β, but every β_i has its own local shrinkage parameter λ_i. Due to the fat tail of the half-Cauchy distribution, β with strong signals can escape the shrinkage τ. In the case of a strong signal the posterior of λ_i can have a very high value, which negates the shrinkage effect of the small value of τ.

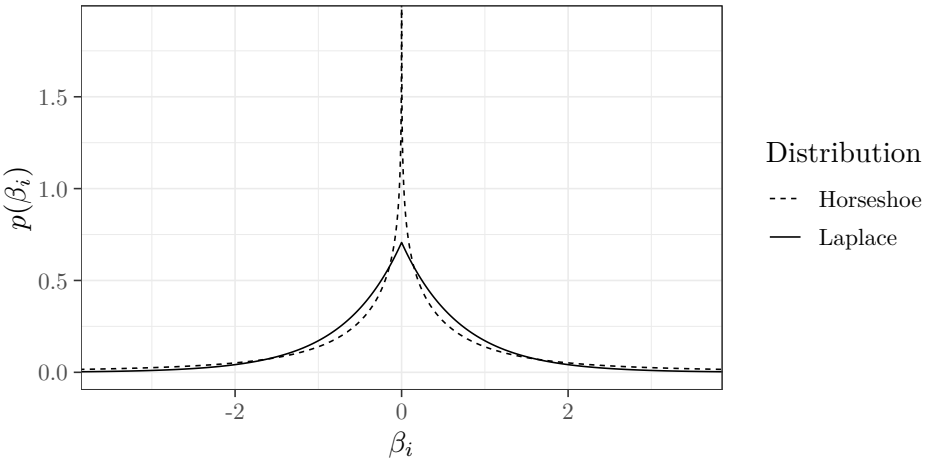

Figure 21: Laplace distribution and distribution of the Horseshoe prior

The density of Horseshoe prior does not have a closed form but behaves like $\log(1 + \frac{2}{\beta_i^2})$ (Carvalho et al., 2009) and is plotted in figure 21. The probability density function of the Horseshoe prior has a infinite peak at $\beta = 0$, and has fat Cauchy like tails. Due to infinite peak, the weak signals β are heavily shrunk towards zero. On the other hand, the fat tails do hardly influence the signal, when the signals are far from zero.

Normal scale-mixtures In the case of the logistic regression, many results do not have a closed-form, which makes the behaviour difficult to understand. To illustrate the behaviour of the Horseshoe prior assume simple model where $y \sim N(\beta, \sigma)$.

The different types of prior discussed before, can be expressed as normal scale-mixtures. Normal scale-mixtures are distribution where the scale parameter of the normal distribution has a distribution itself, such that:

$$\begin{aligned}\beta|\lambda,\tau &\sim N(0,\lambda\tau)\\ \lambda &\sim p(\lambda)\ \text{,or}\\ \lambda^2 &\sim p(\lambda^2)\\ \tau &\sim p(\tau)\end{aligned}$$

By integrating out the λ and τ the prior distribution on β is found:

$$p(\beta) = \int_0^\infty \int_0^\infty p(\beta|\lambda,\tau)p(\lambda)p(\tau)d\lambda d\tau$$

The Laplace prior is a normal scale-mixture where $\lambda^2 \sim Exp(2b^2)$. Integrating out λ^2 results in a $\beta_i \sim \text{Laplace}(0,b)$. The student-t distribution can be seen as a normal scale-mixture model where $\lambda^2 \sim \text{Gamma}(\frac{\nu}{2}, \frac{\nu s^2}{2})$. Integrating out λ^2 gives $\beta \sim \text{student-t}_\nu(0,s)$. Where ν are the degrees of freedom and s is a scale parameter. The normal prior is a normal scale-mixture where $\lambda = \sigma$. So this is for a fixed scale. The Horseshoe prior is a normal scale-mixture where $\lambda \sim \text{Cauchy}^+(0,1)$. The derivations of these statements are in Appendix C. The posterior of a normal scale-mixture model, given λ and τ, is:

$$p(\beta|y,\lambda,\tau) \propto p(y|\beta)p(\beta|\lambda,\tau,\sigma) = \exp\left(-\frac{(\beta-y)^2}{2\sigma^2}\right)\exp\left(-\frac{\beta^2}{2\lambda^2\tau^2}\right)$$

This the product of two Gaussian models is conjugate, so the posterior is:

$$\beta|\lambda,\tau,y \sim N\left(\frac{\tau^2\lambda^2}{1+\tau^2\lambda^2}y, \frac{\lambda^2\tau^2}{1+\tau^2\lambda^2}\sigma^2\right)$$

So the conditional expected value of β in posterior is.

$$\mathbb{E}[\beta|y,\kappa,\sigma] = \left(1 - \frac{1}{1+\lambda^2\tau^2}\right)y$$

For the normal scale-mixtures, except the Horseshoe prior, τ is always equal to one and may be omitted when not applicable.

Now define $\kappa_i = \frac{1}{1+\tau^2\lambda_i^2}$. This is the shrinkage weight associated with β_i. The shrinkage weight has support of $[0,1]$, when $\lambda^2\tau^2 \to \infty$ the shrinkage weight κ goes to zero and for $\lambda^2\tau^2 \to 0$, then the shrinkage weight κ goes to one. The expected value of the posterior can be expressed in terms of κ_i:

$$\mathbb{E}[\beta|y,\lambda,\sigma] = (1 - \mathbb{E}[\kappa_i|y,\sigma,\tau])\,y$$

And variance (Datta et al., 2013):

$$var(\beta|y,\lambda,\kappa) = \left(1 - \frac{1}{1+\lambda^2\tau^2}\right)\sigma^2 = (1-\kappa_i)\,\sigma^2$$

The shrinkage weight κ_i determines the amount of shrinkage towards zero. In case that κ_i is zero there will be no shrinkage. When κ_i is one there will be perfect shrinkage and the expected value of the posterior on β is equal to zero. The variance of the posterior also goes to zero as the shrinkage weight κ_i goes to one.

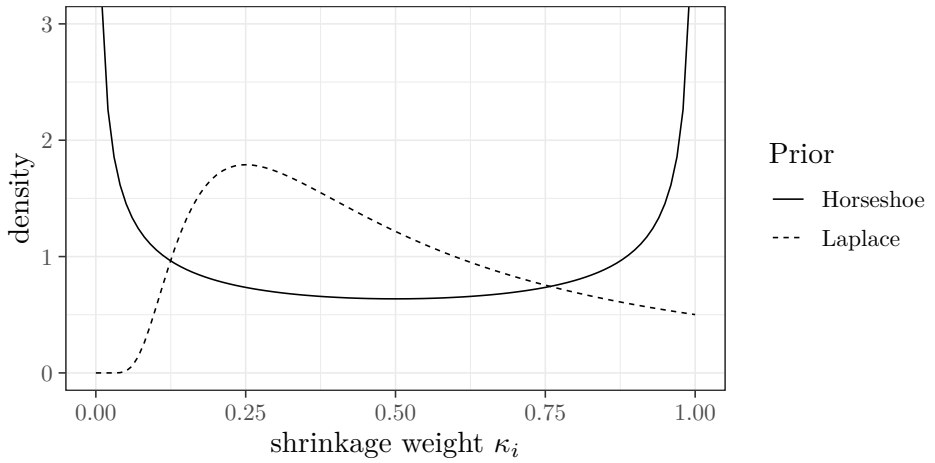

Figure 22: Shrinkage profile of Laplace and Horseshoe prior

By change of variable from $\lambda \sim \text{Cauchy}^+(0,1)$, the implicit prior on κ_i can be found. The unnormalised prior on κ_i for the Horseshoe prior is:

$$p(\kappa_i) \propto \kappa_i^{-1/2}(1-\kappa_i)^{-1/2}$$

and the proper prior is $\kappa_i \sim \text{Beta}(1/2, 1/2)$. This distribution is shaped like a Horseshoe as depicted in Figure 22, hence the name Horseshoe prior. The prior shrinkage weight puts a lot of mass on either total shrinkage or no shrinkage at all. So there is a prior preference to estimate a β_i as either near zero or near the unconstrained signal.

The same change of variable can be done by for the Laplace prior, with $\lambda^2 \sim Exp(2)$. This gives:

$$p(\kappa_i) \propto \kappa_i^{-2} e^{-\frac{1}{2\kappa_i}}$$

This is also depicted in figure 22. The prior shrinkage weight associated with a Laplace prior has little weight on no shrinkage and most of the weight in the middle. This causes the Laplace prior shrink even the strongest signals towards zero. The Lasso regression in figure 15 on page 39, also shows this behaviour.

For the normal prior the shrinkage weight is just a fixed value $\kappa = \frac{1}{1+\lambda}$. So it shrinks all β indiscriminately of signal strength. This is visible as well for the ridge regression in Figure 12 on page 37.

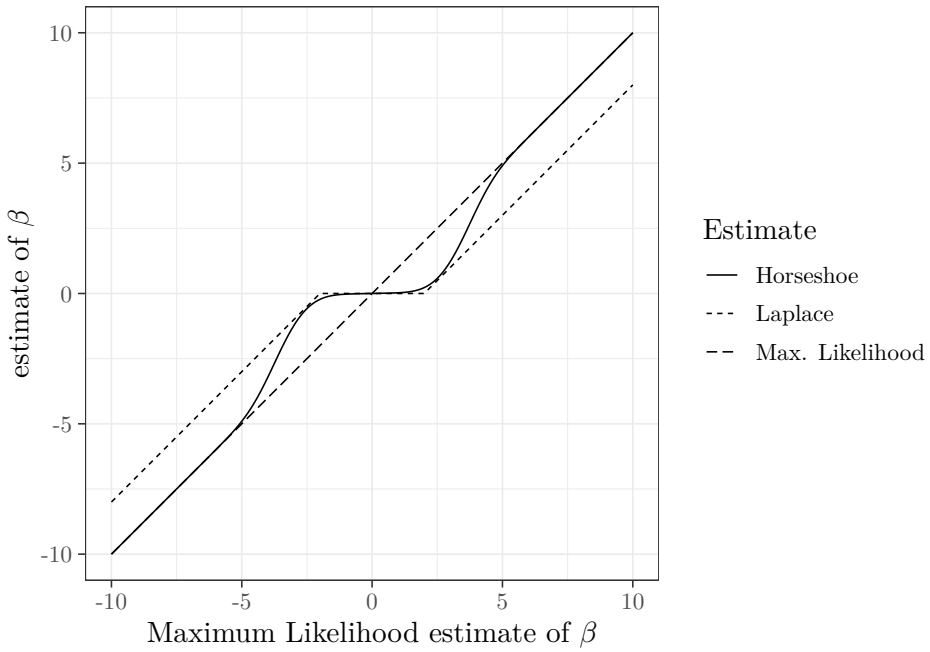

Figure 23: Estimate of the maximum likelihood, Horseshoe prior and the Laplace prior for given maximum likelihood estimate

Figure 23 shows the the expected value of posterior β for different priors. The Laplace prior gives a estimates closer to zero than the maximum likelihood, even for strong signals. The Horseshoe prior shrinks parameters β with a weak signal, but parameters β with a strong signal are almost unchanged (Carvalho et al., 2010).

Global shrinkage parameter Besides the local shrinkage parameters λ, the global shrinkage parameter τ has influence on the sparsity of the parameter vector β.

$$\kappa_i = \frac{1}{1 + \tau^2 \lambda_i^2}$$

When τ has a value that is much less than zero, then it shrinks all β towards zero. So the choice of τ depends on the sparsity assumption of the regression parameters β. If a priori the amount of non-zero parameters is low, then τ could be picked as a low number and vice verse for a high amount of non-zero parameters. The effect of two different τ is shown in figure 24. A smaller τ puts more weight on larger κ, hence more prior shrinkage weight. Furthermore, the implied prior on β has more weight around zero. In the right plot the smaller τ shrinks a bigger range of the maximum likelihood estimate Y towards zero.

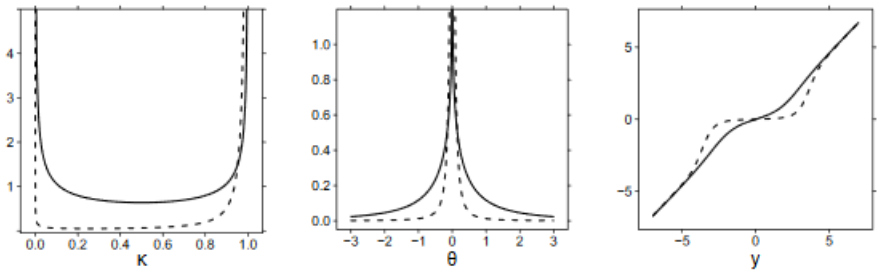

Figure 24: Effect of τ on prior distributions. The dotted is $\tau = 0.05$, and black line is $\tau = 1$ (Reprinted from Van Der Pas et al. (2014))

Now consider the model which use more than one data points (y, X), consider a linear regression instead of the signal model and define the effective amount of parameters m_{eff} of the in the parameter vector:

$$m_{eff} = \sum_{i}^{D} (1 - \kappa_i)$$

Much a priori shrinkage means that the κ_i are big so m_{eff} is small. For a linear regression the expected value of m_{eff} is (Piironen et al., 2017):

$$\mathbf{E}[m_{eff}|\tau, \sigma] = \frac{\tau\sigma^{-1}\sqrt{n}}{1 + \tau\sigma^{-1}\sqrt{n}} D$$

If the true number of effective parameters m_0 is known then the previous equation can set equal to m_0, which gives the following result:

$$\tau_0 = \frac{m_0}{D - m_0} \frac{\sigma}{\sqrt{n}}$$

In general the amount of non zero parameters m_0 is not know, and setting a fixed τ_0 may be to restrictive. To give the model more flexibility to choose the right amount of important parameters, a half Cauchy prior can be put on τ.

$$p(\tau) \sim C^{+}(0, \tau_0)$$

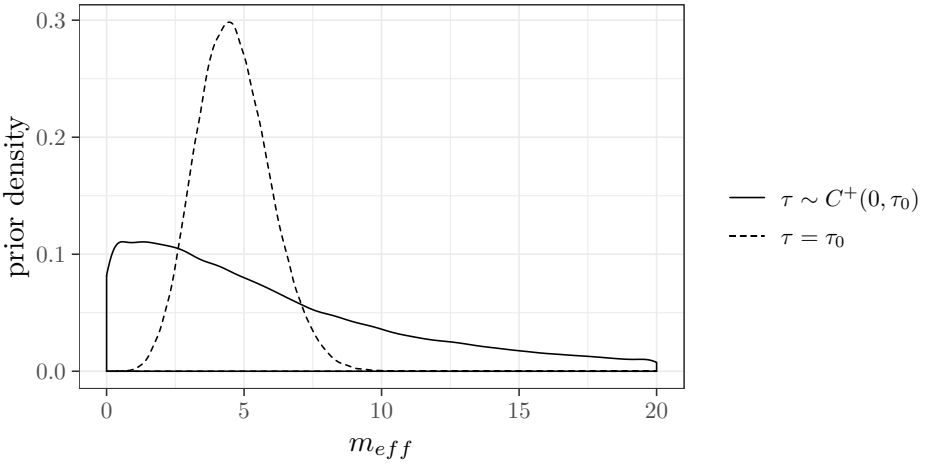

Figure 25: Difference between fixing τ and giving τ a half Cauchy prior.

The value of τ_0 is often a really small value and so the scale of the Cauchy distribution is small. Most of the prior mass of τ is located between $[0, 1]$. In figure 25 the effect of using the Cauchy prior is compared with using a fixed value for τ.The prior on m_{eff} for the fixed τ has most mass around m_0. Whereas the implied prior on m_{eff}, with a half Cauchy prior on τ has more evenly distributed weight in its support of m_{eff}.

In case that τ has a prior, it also has a posterior. Given the shrinkage κ , τ is conditionally independent on y. Also the κ_i's are conditionally independent given τ. τ is estimated via the average signal strength. So the Horseshoe prior adapts to the average strength of β from the data itself. This is different from the other regularisation techniques, which are not adaptive. In the case of Lasso regression many runs, with different regularisation coefficient λ, are done to find the right amount of regularisation.

The previous results are in case of linear regression, however the classification of defaults is done via a logistic regression. For a linear model the variance σ^2 is a constant for all data points, but for the logistic regression this is not the case. Still a value for σ is needed and as a plug-in value for the variance the pseudo variance of logistic regression is used, which is (Gelman et al., 2013):

$$\tilde{\sigma}^2 = \bar{y}^{-1}(1 - \bar{y})^{-1}$$

Where $\bar{y}$ is the sample mean of y.

Finnish Horseshoe The Horseshoe prior has Cauchy-like tails with $\mathcal{O}(\beta^{-2})$. As discussed as in section 5.1, this can cause computational problems when solving the model. For computational reasons it would be preferable to have thinner tails than the Cauchy distribution.

Piironen et al. (2017) introduced the Finnish Horseshoe, which is an adaptation of the Horseshoe prior and is constructed as follows:

$$\begin{aligned}
\beta_i &\sim \text{Normal}(0, \lambda_i \tau) \\
\tilde{\lambda}_i &\sim \text{Cauchy}^+(0, 1) \\
\tau &\sim \text{Cauchy}^+(0, \tau_0) \\
c &\sim \text{Inv-Gamma}(\nu/2, \nu s^2/2) \\
\lambda_i^2 &= \frac{c^2 \tilde{\lambda_i}^2}{c^2 + \tau^2 \tilde{\lambda}_i^2}
\end{aligned}$$

The Finnish Horseshoe prior has a student-$t_\nu(0, s^2)$ tails instead of a Cauchy tail (see Appendix C). Which makes it numerically more stable .

Where as the Horseshoe prior does not regularise parameters with strong effect the Finnish Horseshoe prior has some regularisation for strong signals. When $\tau^2 \tilde{\lambda}_i \ll c^2$, then $\lambda_i \approx \hat{\lambda}_i$, and the Finnish Horseshoe has the same value as the Horseshoe prior. When $\tau^2 \tilde{\lambda}_i^2 \gg c^2$, then $\lambda_i^2 \to \frac{c^2}{\tau^2}$.

So for strong signal the Finnish Horseshoe slightly regularises β. The shrinkage for strong signals is much weaker than the shrinkage for weak signals. Also the shrinkage of the Laplace prior is much stronger than the Horseshoe prior for strong signals.

The Horseshoe prior is a special case of the Finnish Horseshoe prior, where $\nu = 1$ and $s = 1$. This is the basic setting I use, unless error occur in fitting the model or when psis-loo gives errors.

Example 5.1: Horseshoe prior on Sparse Data

For data generating process 1, the logistic regression is fitted with a Finnish Horseshoe prior and the problem is solved using Hamiltonian Monte Carlo. The results of the first eight parameters are shown in Figure 26.

The a priori effective numbers of parameters $m_0 = 4$. $s = 1$ and $\nu = 1$, and $D = 20$ This gives:

$$\tau_0 = \frac{4}{20 - 4} \frac{\sqrt{50}}{\sqrt{4000}} \approx 0.03$$

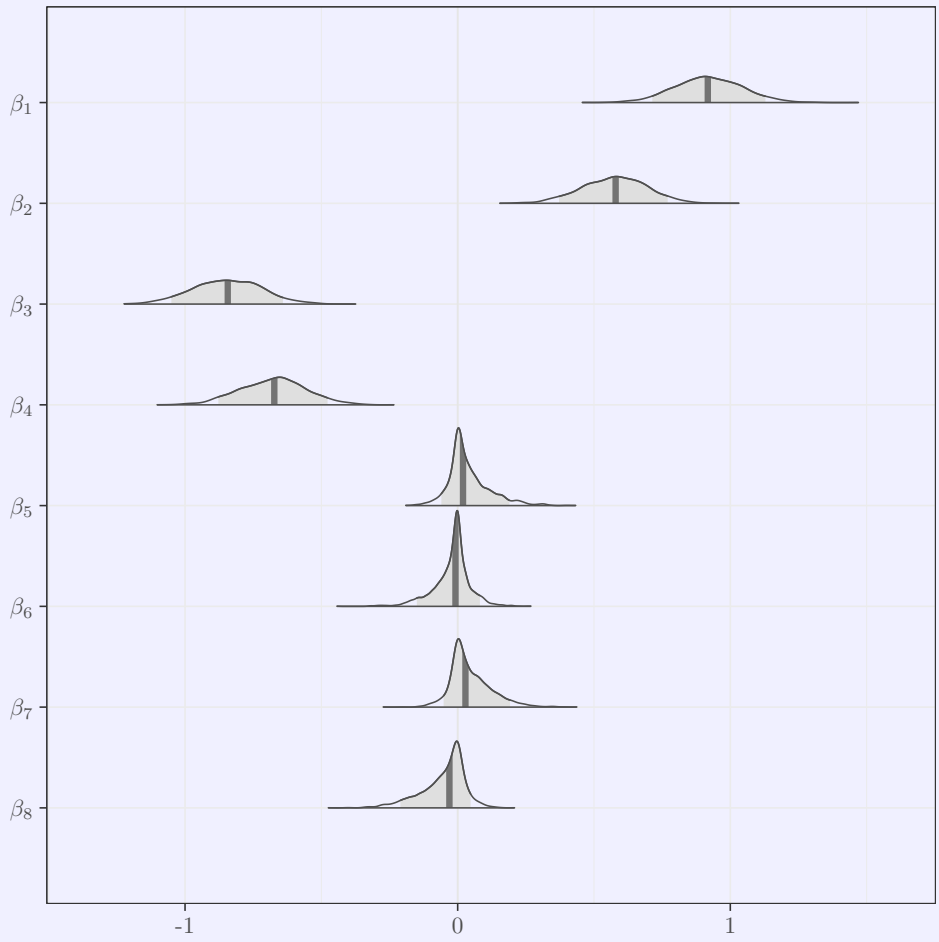

Figure 26: Marginal distribution of posterior associated with the first eight (out of twenty) variables.

The data generating process of the first four parameters have a real effect on the prediction of the response variable y. The posterior distribution do not include zero in their 95% credible interval. The other parameters have no influence in the data generating process. In this case the Finnish Horseshoe prior causes the posterior to have a peak around zero.

The time to find 1,000 Monte Carlo samples of the posterior takes a couple of minutes.

Example 5.2: Horseshoe with Collinearity

For data generating process 2.4, fit the Horseshoe prior. The posterior of uncorrelated data shows the same behaviour as in Figure 26. For the correlated contributing data X^2 and X^3 the joint distribution looks similar to the ridge as in Figure 3 on 21. Figure 27 shows the Monte Carlo samples of the β_4 and β_5, which corresponds to the correlated data, where X^4 really contributes to the prediction and X^5 is noise. The marginal posterior distribution of β_5 has a peak around zero. Just like in the case of the uncorrelated data in Figure 26. The posterior of β_4 does not have this peak. It is still visible that the two parameters are correlated. This correlation causes the marginal distribution of β_4 to be wider than the conditional distribution $p(\beta_4|\beta_5 = 0)$

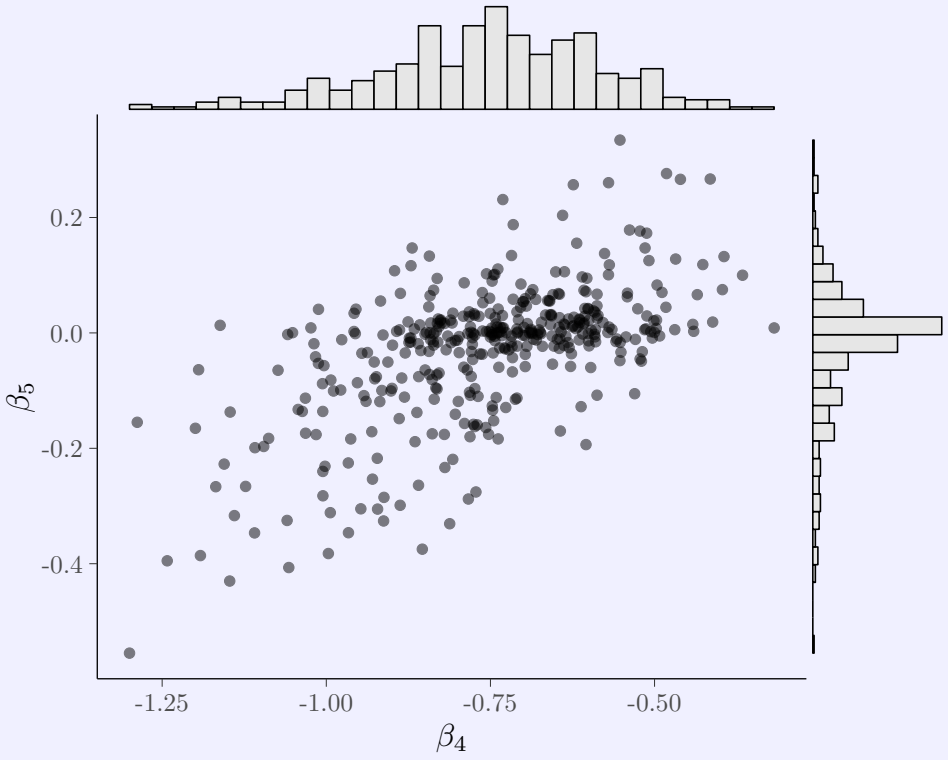

Figure 27: The effect of the Horseshoe on correlated parameters

Example 5.3: Dense Data

Now consider a model where all parameters β are non-zero. Such that the first ten $\beta = 1$ and the last ten $\beta = -1$. Like before, the explanatory variables X are drawn i.i.d. with standard deviation $\sigma = 1$. The Horseshoe prior is specified as in Example 5.1. Figure 28 shows the posteriors intervals of the dense data. Even though the initial guess of the amount of effective parameters m_0 is four, the Horseshoe prior adapts to the dense signals. The 95% credible interval all contain the true value.

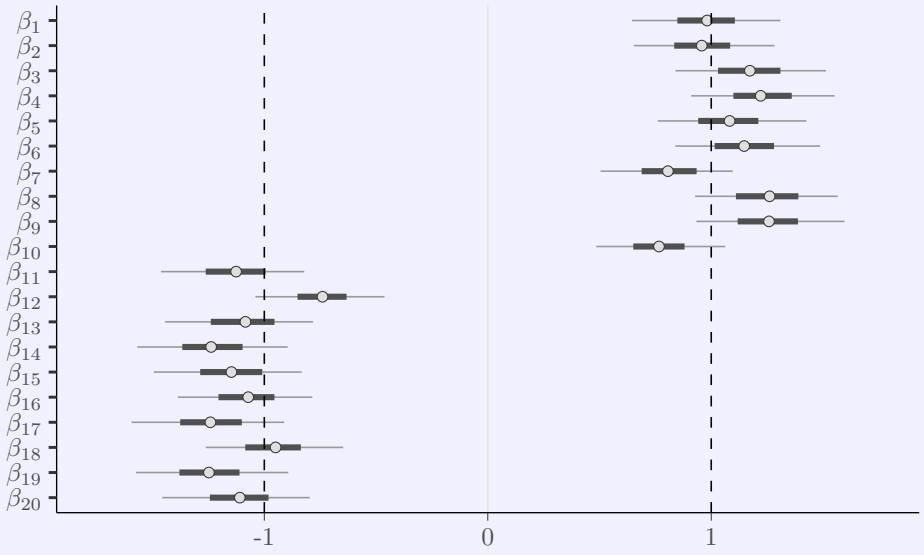

Figure 28: Posterior intervals of the logistic regression with Horseshoe prior and dense data.

Finding the posterior of the Horseshoe prior is computationally more difficult than the finding the posterior when using a Laplace prior or a Gaussian prior. This has to with the fact that the Horseshoe prior has a hierarchical structure, where all the parameters β depend on a the global shrinkage parameter τ. This hierarchical structure causes high gradients in the posterior where τ is small. Therefore, the steps size in the leapfrog integrator (Appendix $A.2.2$, p. 94) of Hamiltonian Monte Carlo needs to be smaller than for the other priors. The (Finnish) Horseshoe prior takes about 5 to 10 times longer than non-hierarchical priors. In our case this is not a real problem, because it takes only couple of minutes. On the other hand, the Horseshoe prior does not require as much tuning as for example the Laplace prior, which needs many runs with different regularisation parameters λ to do something comparable with Lasso Regression.

5.3 Predictive Projection

After fitting a Bayesian logistic regression, the posteriors of the regression coefficients β are never truly sparse. Even though a parameter has almost no influence on the prediction, there is never any posterior mass on zero. Features that do not have any predictive power to the models are preferably left out.

Dupuis & Robert (2003) suggest projecting from the full model, or reference model, to a subset of the parameter space that contains less features. The goal of the projection is to get a smaller model, with as similar predictive power as possible to the reference model. In this case the reference model is the posterior of logistic regression with Horseshoe prior with all the variables.

A measure for the difference predictive power between two distributions is the Kullback-Leibler (KL) divergence, which is grounded in Information Theory (see Appendix B). The Kullback-Leibler measures the information loss of describing one distribution by another distribution. Let $q(\theta^{\perp})$ be the reduced model, where a part of regression coefficients from the full reference model $p(\theta|y)$

are set to 0. The difference loss of information by using the reduced model is:

$$\mathrm{KL}(q(\theta^{\perp}), p(\theta|y)) = \mathbb{E}_{\tilde{y}}\left[p(\tilde{y}|X, y) - \log q(\tilde{y})\right] \tag{9}$$

And this gives a approach to find the projected parameters $\theta^{\perp}$:

$$\theta^{\perp} = \underset{\theta_* \in \Omega}{\operatorname{argmin}} \mathrm{KL}(p(\tilde{y}|\theta_*), p(\tilde{y}|\theta)) \tag{10}$$

The projection does not consider the data anymore, and the flow of information from the data set is stopped after fitting the model. The projection only uses information from the reference model. In other words, the goal is to make a smaller model with less dimensions, that is as similar as possible to the reference model.

In the reference model the parameters that contribute to the prediction are the regression coefficients β with the design matrix X. Let the parameters of the reduced model be $\theta^{\perp} = \{\gamma_0, ..., \gamma_r\}$, where $\dim(\gamma) < \dim(\beta)$. The explanatory variables matrix $Z \subset X$ is the subset of the full design matrix, associated with the projected parameters γ.

Dupuis & Robert (2003) show that, in the case of the logistic regression, solving equation 9 is equal to solving:

$$\underset{\gamma \in \mathbb{R}^{dim(\gamma)}}{\operatorname{argmin}} \sum_{i}^{n} \left\{ (\beta X_i - \gamma Z_i) - \log\left(\frac{1 + \beta X_i}{1 + \gamma Z_i}\right) \right\}$$

A similar simplification can be found for all generalised linear models. The projected γ are found by solving the following equation.

$$\sum_{i=1}^{n} \frac{\exp(\gamma^s Z_i)}{1 + \exp(\gamma^s Z_i)} Z_i = \sum_{i=1}^{n} \frac{\exp(\beta^s X_i)}{1 + \exp(\beta^s X_i)} Z_i$$

In the fitted model, β is not a single point but a collection of Monte Carlo samples such that $\beta = \{\beta^1, .., \beta^S\}$. For every value of Monte Carlo estimate β^s, a projection can be made to the reduced space in which case there are as many Monte Carlo samples in γ as in β. This is called a draw-by-draw projection. On the other hand, it is also possible to project the parameters β to a single point estimate γ. This single point process is quicker than the draw-by-draw procedure, but it loses information that is present in the full posterior. Piironen et al. (2018) suggest a clustered approach, which is a generalisation of the single point projection and the draw-by-draw projection, where for a cluster of β a single γ is calculated. Where draw-by-draw has an amount of clusters C equal to the number of Monte Carlo Samples S and the single point projection only has one cluster. In the clustered approach the Monte Carlo samples of β^s are split up in cluster and for each cluster a projected value of γ^c is calculated. Such that:

$$\gamma^c = \underset{\gamma \in \mathbb{R}^{dim(\gamma)}}{\operatorname{argmin}} \sum_{i}^{n} \left(\gamma Z_i + \log(1 + \gamma Z_i) + \frac{1}{S_c} \sum_{s}^{S_c} \beta^s X_i - \log(1 + \beta^s X_i) \right)$$

S_c is the size of the Monte Carlo cluster of β and γ^c is a single corresponding to the cluster.

For a model with 4,000 points the the single point projection takes about half an hour. The draw-by-draw approach would take days to complete. So the time difference is quite big. I choose 10 clusters for the predictive projection for model selection. For 10 clusters the predictive projection gives a better prediction than with one cluster and takes approximately four hours to finish. For the implementation I use the R-package *projpred*, This package is compatible with *rstanarm*, which is use for fitting of Bayesian general linear models.

The amount of submodels is 2^D and calculating the Kullback-Leibler divergence and the projected parameters γ for every possible submodel of the reference model takes too long. To get around this problem, the Forward Selection or the Lasso regression can be used on Equation 5.3. For less than 20 parameters *projpred* applies the Forward Selection and for larger it uses the Lasso ranking. Forward Selection is more accurate in picking the right features, but it is much slower for higher dimensions.

Example 5.4: Predictive Projection

Take the same logistic regression as before with the Horseshoe prior as in Example 5.1 on page 52. Projecting from the refence model with the Horseshoe prior, correctly identifies the important variables. In figure 5.4 the elpd is plotted against the number of variables the method picks. This is a typical profile of this method. The elpd increases till the amount of the selected variables and then is almost constant. The reason that the elpd remain constant is that the noisy parameters β are shrunk towards zero and do not influence the prediction of the reference model.

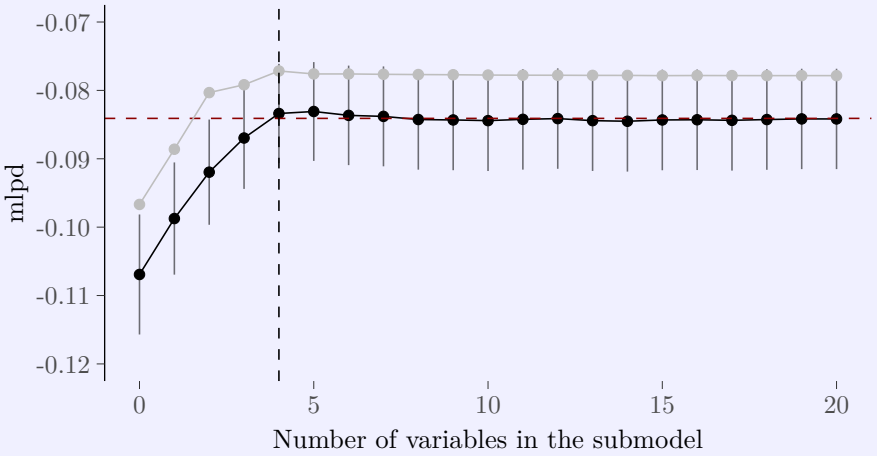

Figure 29: Projection of full model to sparse model, the vertical dashed line is the suggest model size

For the Forward Selection variant of the predictive projection and 4,000 samples, computation takes about three hours.

Example 5.5: Predictive Projection with Collinearity

Take the fit of the Horseshoe as in example 5.2 and use Forward Selection predictive projection. Again the method correctly identifies the 4 non-zero regression coefficients.

The marginal distribution of the posterior β_4 is wide due to the correlation with the posterior β_5. However the predictive projection algorithm finds that β_5 has no predictive contribution to the model and drops the variable out of the model. The Monte Carlo samples of β_4 are projected and the result is shown in Figure 5.5. The projected posterior is much narrower than the marginal posterior of the reference model. The figure also shows the posterior for the logistic regression which is refitted on the four important variables X. Even though it is not really visible, the projected posterior has a variance that is 12% bigger than the refitted model. This increase in variance is a result of the uncertainty of the feature selection.

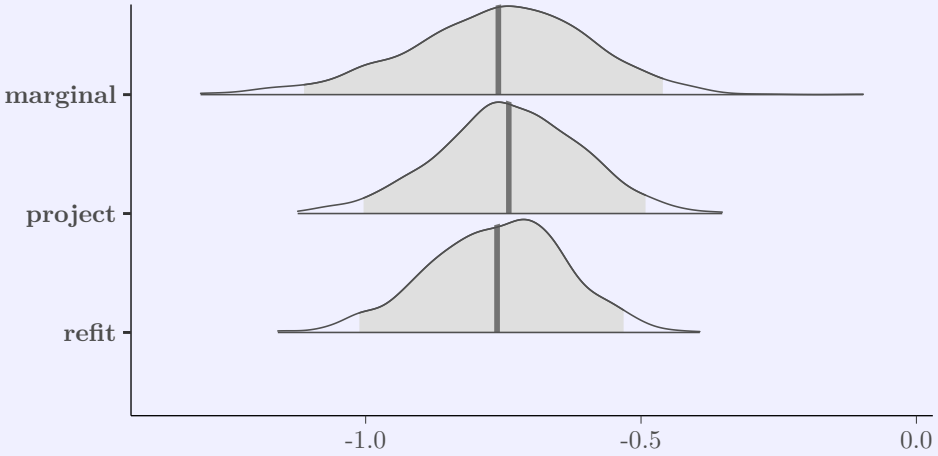

Figure 30: The marginal posterior, projected posterior and refitted posterior of β_4. The grey area is the 95% credible interval and the dark grey line is the expected value of the distribution.

5.4 Final Remarks

In Bayesian Statistics, prior provide regularisation. The normal prior and Laplace prior give similar results to the ridge regression and the Lasso regression respectively. The Horseshoe prior has a different shrinkage profile than Lasso regression. Where Lasso has the tendency to shrink all regression coefficients, the Horseshoe prior leaves certain signals unchanged while heavily shrinking others. If by chance an important variable has a maximum likelihood estimate near zero, than the Horseshoe prior aggressive shrinks the posterior of that parameters towards zero. Both the Bayesian and the frequentist variable selection methods have similar predictive power for 4,000 samples. The thing that stand out is that Lasso regression selects too many variables. In the next sections I consider a smaller amount of data, in this case the difference between the variable selection methods become more apparent.

6 Simulation studies

So far the variables selection have been applied to two data set with with 4,000 observation. For this amount of observation all methods worked reasonably. In this section the selection methods are applied on more simulated data. For these simulation I use 1,000 sample. For this amount overfitting is important, while this is not the case for 4,000 observations. In this section different types of simulated data are used to investigate the behaviour of the variable selection methods.

6.1 Independent Explanatory Data

Throughout the thesis I use data generating process 1 with independent explanatory variables (p. 24) to show the effect of different variable selection methods. The data is sparse, because only 4 out of the 20 parameters have a non-zero value. The explanatory variables X in this data are drawn from a multivariate normal distribution with a covariance matrix that is equal to the identity matrix. In this section the number of observations is 1,000 instead of 4,000 as on page 24.

Figure 31 shows the predictive performance for the amount of variables that are selected by the Predictive Projection with a Horseshoe prior. The method correctly identifies the amount of parameters. The performance of the model does not change much above a certain number of variables in the model. This has to do with the fact that in the reference model, unimportant regression coefficients β are shrunk to zero and do not influence prediction. This is typical behaviour for this method and means that selecting a model with too many variables does not induce much overfitting.

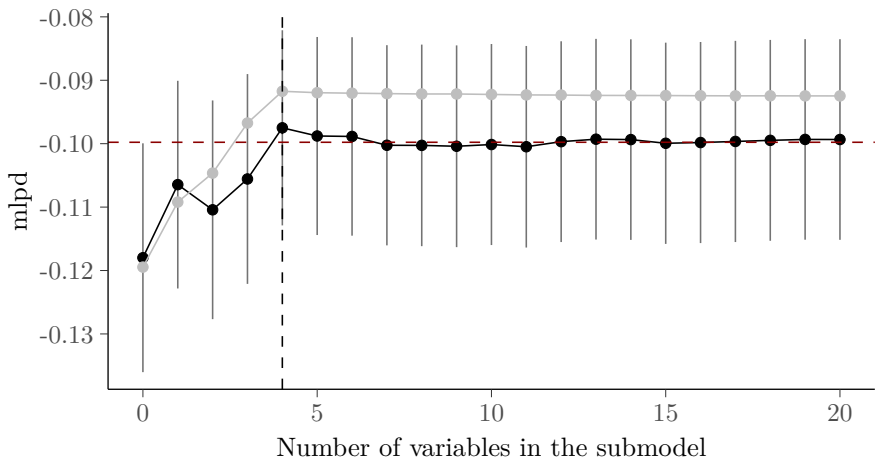

Figure 31: Horseshoe projection with 1000 data points. The black line is the psis-loo performance estimate and the grey line is the performance on the hold-out-set, the dotted vertical line is selected amount of variables.

In Figure 32 Lasso variable selection is shown. Lasso variable selection with 1,000 observations overestimates the amount of important variables in the model, just like Lasso with 4,000 observations op page **??**. Both the Horseshoe prior and Lasso regression the curve of the estimated performance and the real performance have a similar shape. The real performance is lower than the estimated performance. However, all predictions have a error in the same direction.

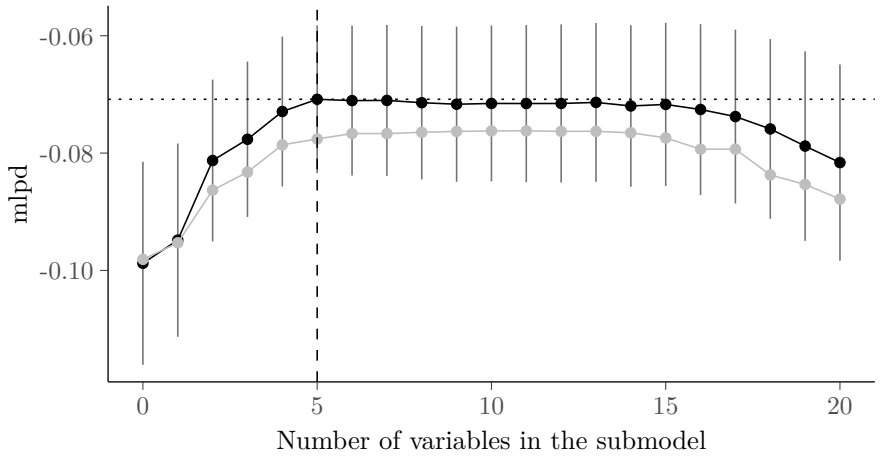

Figure 32: Lasso variable selection for 1000 data points. The black line is the K-fold estimate and the grey line is the performance on the hold-out-set, the dotted vertical line is selected amount of variables.

Relaxed Lasso (Figure 33) has a different pattern than Lasso regression. Compared to Lasso it has a relatively quick increase of the out-of-sample performance by adding more variables to the model. This is caused by the refitting of the parameters, such that they are not regularised anymore. When increasing the amount of parameters further, the performance of the model also decreases relatively quickly. This is again a result of removing the regularisation, which causes overfitting. The best out-of-sample performance occurs when the submodel has 4 parameters. However, Relaxed Lasso only picks three regression coefficients, causing worse performance than for Lasso and Horseshoe. Whereas there is a clear relation between the estimated and predictive performance of the Lasso and Horseshoe, this relation is less clear in the Relaxed Lasso.

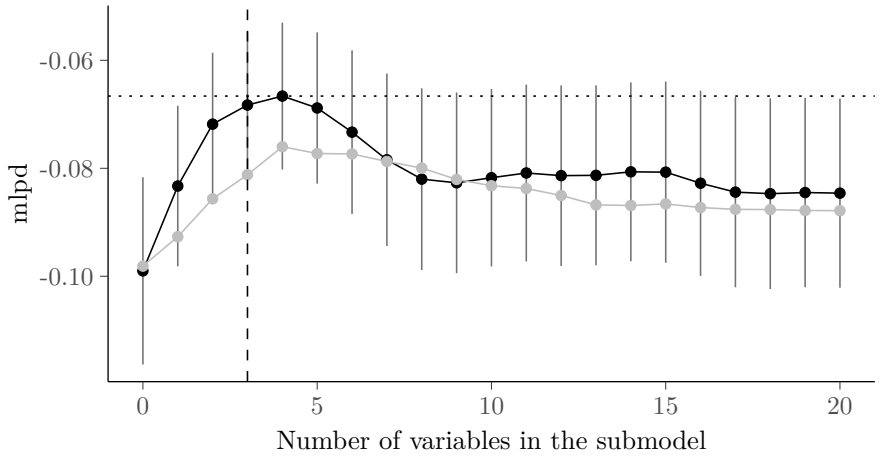

Figure 33: Relaxed Lasso variable selection for 1,000 data points. The black line is the K-fold estimate and the grey line is the performance on the hold-out-set, the dotted vertical line is selected amount of variables.

The same is true for Forward Selection in Figure 34. The out-of-sample performance increases quickly, but after reaching four variables the performance deteriorates. Forward Selection also picks the wrong number of variables. The relation between the predictive and estimated performance is also less clear than for the Horseshoe prior or Lasso regression.

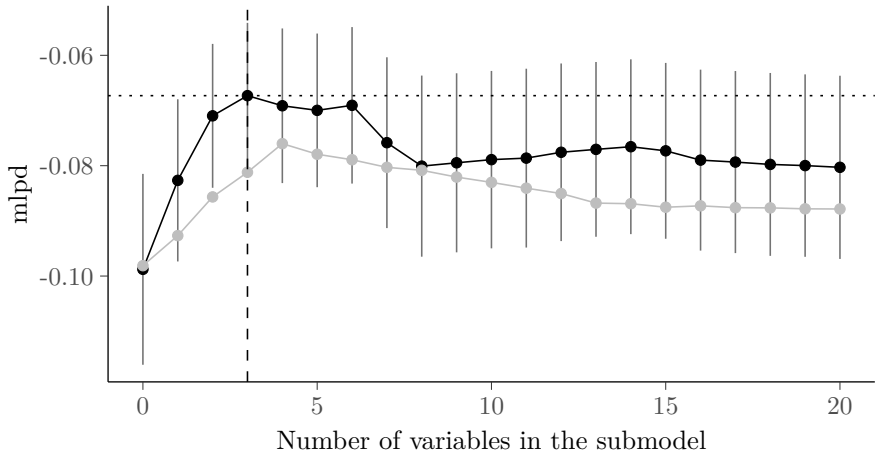

Figure 34: Forwards selection for 1,000 data points. The black line is the K-fold estimate and the grey line is the performance on the hold-out-set, the dotted vertical line is selected amount of variables.

The relation between the estimated and predictive performance is important for the variables selection, because, the variable selection is done on the relative performance. I decompose estimated mlpd in different parts to illustrate this:

$$\text{mlpd}_{\text{est}} = \text{mlpd}_{\text{real}} + \text{systematic error} + \text{idiosyncratic error}$$

The systematic error is the error that is the same for all submodels. The idiosyncratic error is the error that is unique to every submodel.

The selection criterion (SC) are based on the relative performance of the model. So:

$$SC(m_i) = \text{mlpd}_{\text{base}} - \text{mlpd}_i$$
$$SC(m_i) = \text{real effect} + \text{idiosyncratic error of base model} + \text{idiosyncratic error of i}$$
$$SC(m_i) = \text{real effect} + \text{variability of selection criterion}$$

$\text{mlpd}_{\text{base}}$ is the model with the highest estimated performance. This means that the systematic error is not relevant for the selecting features. The two thing that matter are the real effect of adding a parameter and the idiosyncratic errors. The selection criterion makes the right decision when the real effect is large compared to the variability of the selection criterion.

Figure 35 shows the hold-out mlpd for the variable selection methods. The performance of the best possible model is higher than both best Lasso and Horseshoe model. However, the selected model perform worse. Unlike the Horseshoe and Lasso selection procedure, Relaxed Lasso and Forward Selection do not have the same direction of bias for the predictions, which leads to a selection induced bias (Cawley & Talbot, 2010).

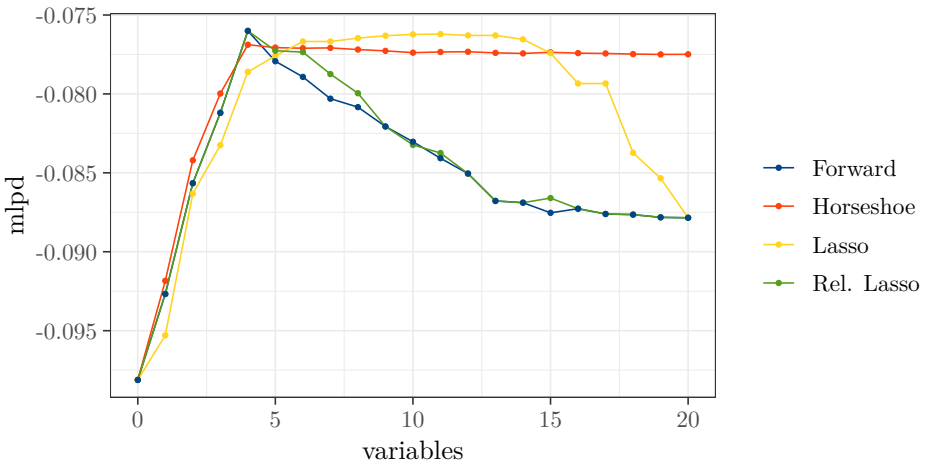

Figure 35: Hold-out performance of models of the variable selection methods on independent explanatory variables

6.1.1 Multiple runs

Due to randomness in the data generating process, one method might perform better than another method on one realisation of the data generating process and worse on another realisation. To deal with this randomness, I draw multiple training sets from data generating process 1 and I apply each variable selection method to all the training sets.

The amount of defaults in the generated data can also vary due to the randomness. However, I am interested in the case that there are 20 defaults in the data set. To guarantee that all data sets contain the same amount of defaults, I create a large data set from the data generating process. From the large data set, I use stratified sampling to get training sets with 20 defaults and 980 non-defaults. Besides the training set, I also create a test set containing 100,000 observation to get the hold-out performance.

Algorithm 1: Multi Run

```
Draw big data set with training data
Draw hold-out validation set (n = 100,000).
for i in 1:30 do
    train data = stratified sample from big data set
    Run variable selection methods.
    Check performance on the hold-out set
end
```

Predictive performance I calculate two reference values of the performance for the data generating process. Namely, an upper limit of performance and and a lower limit of performance. The lower limit is the performance of random guessing that a loan will default with a probability of 2% . This is equal to the entropy of a Bernoulli distribution with $P(y = 1) = 0.02$. The mlpd for this model is:

$$\text{mlpd}_{\text{random}} = -98.04 \cdot 10^{-3}$$

The other reference value is seen as the upper limit of prediction. This limit is found by making predictions with the true values of the four important variables on the hold-out set.

$$\text{mlpd}_{\text{datagen}} = -72.52 \cdot 10^{-3}$$

Table 15 shows the out-of-sample performance of the variable selection methods over 30 iterations. Predictive Projection and Lasso have the highest performance and Relaxed Lasso and Forward Selection have the worst performance. Lasso regression has the lowest variability of the performance over different realisation of the data, however, it selects the most variables. All methods perform

worse than the theoretical maximum value of the data generating process. This is caused by two thing, firstly, the methods sometimes include unimportant variables and exclude important variables. Furthermore, the estimation of the important regression coefficients is not perfect due to the lack of data.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) (×1000)	-77.28	-76.93	-77.70	-78.33
sd(mlpd) (×1000)	4.74	3.42	5.21	4.66
# var.	3.90	7.76	4.07	3.70

Table 15: Hold-out-performance (mlpd) and Average number of included variables (# var.) for selection methods on independent explanatory data. There are four important variables.

Table 16 shows the difference in performance between the best submodel and the selected submodel. The performance of the submodel with the highest performance on the hold-out set ismlpd$_{\text{best}}$ and the hold-out performance of the selected model is mlpd$_{\text{select}}$.

$$\Delta\text{mlpd} = \text{mlpd}_{\text{best}} - \text{mlpd}_{\text{select}}$$

Predictive Projection and Lasso regression have a lower difference between the best submodel and the selected model. For Relaxed Lasso and Forward Selection this difference is bigger. The performance of the best submodels of Forward Selection and Relaxed Lasso is on average better than the best submodels of the Horseshoe prior and Lasso regression. This shows that the former methods have more difficulty with finding the right submodel. Furthermore the first two method have a lower standard deviation of Δmlpd, so the selection mechanism is more stable.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd$_{\text{best}}$) (×1000)	-76.00	-75.78	-74.58	-75.07
mean(Δmlpd) (×1000)	1.28	1.15	3.11	3.26
sd(Δmlpd) (×1000)	1.74	1.71	4.14	3.11

Table 16: Difference between the hold-out performance of best submodel and hold-out perfomance of selected submodel

Estimated mlpd Besides that Relaxed Lasso and Forward Selection have difficulty with selecting the right submodel, there is another problem with these two methods. This has to do with the estimate of out-of-sample performance. K-fold cross validation and PSIS-LOO are unbiased estimates for the out-of-sample performance without variable selection. However, the selection procedures can cause these out-of-sample predictions to be biased. To show this, define the difference between the real mlpd of the selected submodel (mlpd$_{\text{real}}$) and the estimated mlpd of the selected submodel (mlpd$_{\text{est}}$) as,

$$\epsilon = \text{mlpd}_{\text{real}} - \text{mlpd}_{\text{est}}$$

Table 16 shows the difference between the estimated mlpd and the real mlpd. The average of the difference is small for Predictive Projection and for Lasso. For Relaxed Lasso and Forward Selection there is a clear difference between the estimate and the real value of mlpd. The prediction overestimates the performance of the model. Furthermore, the standard deviation is higher for Relaxed Lasso and Forward Selection. This makes the K-fold estimates less reliable for the last two methods.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) (×1000)	0.10	0.84	-2.85	-2.82
sd(ϵ) (×1000)	6.50	5.86	8.04	8.72

Table 17: Difference between estimated performance and real performance

The reason Relaxed Lasso and Forward Selection have a selection induced bias has to do with the variability of the selection criteria. The variable selection depends on the mlpd given by K-fold cross validation and psis-loo.

$$m_{select} = \underset{i:m_i}{\operatorname{argmin}} \left\{ m_i : P(\text{mlpd}_{\max} - \text{mlpd}_{m_i} < 0) > 0.36 \right\}$$

$$\text{mlpd}_{\max} = \max_{i \in D} \text{mlpd}_{i,\text{hold-out}}$$

$$\text{mlpd}_{\text{select}} = \text{mlpd}_{m_{\text{select}},\text{hold-out}}$$

Due to the maximisation operation the estimate is biased.

Selected Variables Table 18 displays which variables the variable selection methods pick. All methods are less likely to include regression coefficient β_2 and β_4 than β_1 and β_3. The effect of the second and fourth variables are weaker than those of the first and third variables, which makes it harder for the methods to detect the signal.

	Horseshoe	Lasso	Relaxed Lasso	Forward
β_1	0.87	0.97	0.93	0.87
β_2	0.53	0.80	0.57	0.60
β_3	0.93	0.97	0.90	0.93
β_4	0.70	0.90	0.73	0.73
other β	0.20	4.00	0.63	0.53

Table 18: Average Inclusion of parameters by the different methods.

Predictive Projection with a Horseshoe includes the least amount of unimportant variables and Lasso includes the most unimportant variables. Although Lasso includes the most variables, the performance of Lasso does not deteriorate much, because of regularisation. The amount of falsely included and excluded variables in Table 19 show that Relaxed Lasso and Forward Selection also include more unimportant variables. For these methods the unimportant variables are not regularised, which causes a worse performance.

	Horseshoe	Lasso	Relaxed Lasso	Forward
False Inclusion	0.20	4.00	0.63	0.53
False Exclusion	0.97	0.36	0.87	0.87

Table 19: False inclusion and false exclusion of variables.

Computation Time As expected, the Frequentist methods are fast compared to the Bayesian method. Both variations of the Lasso Regression are the quickest. The selection methods take a couple of second to find the result. Forward Selection takes a bit longer, with a couple between 30 and 60 seconds.

Finding the posterior of the model with the Horseshoe prior takes a couple of minutes. Predictive Projection takes the most time which runs for 30-40 minutes. This is approximately the same for all data generating processes.

6.2 Collinear Explanatory Variables

Correlation between different explanatory variables is a common occurrence in economic data. In this section I consider three different types of collinear data, namely correlation that causes a masking effect, correlation with aligned effects and correlation with unimportant variables.

6.2.1 Masking effect

The second example throughout the thesis is the a data where the explanatory has collinearity, and is drawn from data generating process 2.4 introduced op page 26. To be precise $\rho(X^2, X^3) = 0.8$, β_2 has a positive effect and β_3 has a negative effect. With an increase of X^2, variable X^3 is also likely to be increase. These effects almost cancel each other out on average. This means that both β_2 and β_3 are needed to make good predictions. If only one of the variables is included in the model, then the effects is not detected.

This cancellation can also be seen in the performance of the data generating process, which is lower than that of the independent explanatory variables.

$$\text{mlpd}_{\text{datagen}} = -81.13 \cdot 10^{-3}$$

The lower reference value remains the same, as random guessing remains the same.

$$\text{mlpd}_{\text{random}} = -98.02 \cdot 10^{-3}$$

Predictive performance The performance of the feature selection methods (Table 20) are also relatively worse than for the independent predictors. This can be contributed to the fact that two variables are needed to explain the weak effect. In this case Forward Selection performs best, but the differences between the selection procedures are small and can be contributed to randomness.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) (×1000)	-88.06	-87.78	-88.41	-87.42
sd(mlpd) (×1000)	3.45	3.97	4.51	3.97
# var.	1.90	4.63	2.07	2.40

Table 20: Hold-out-performance (mlpd) and Average number of included variables (# var.) different selection models on masked data. The true amount of variables is 4

The performance of the best submodel is again highest for Relaxed Lasso and Forward Selection, and they make a larger selection error than the Horseshoe and Lasso regression (Table 21).

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean($\text{mlpd}_{\text{best}}$) (×1000)	-85.10	-85.18	-84.39	-84.30
mean(Δmlpd) (×1000)	2.96	2.61	4.02	3.12
sd(Δmlpd) (×1000)	2.24	2.96	3.55	3.14

Table 21: Difference between best submodel and selected submodel

Estimated mlpd The estimated performance of Predictive Projection and Lasso is again more conservative than that of Relaxed Lasso and froward selection. The standard deviation of the error ϵ is also lower for the first two methods.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) (×1000)	0.64	1.12	-1.10	-0.86
sd(ϵ) (×1000)	5.64	5.35	7.66	6.91

Table 22: Difference between Estimated performance and real performance

Selected Variables The problem of the masking effect is visible in the selected variables (Table 23). Regression coefficients β_2 and β_3 are less likely be to picked than β_1 and β_2. From these two masked variables, β_3 has the stronger effect and is picked more on average than the unimportant variables, while β_2 is not. The data with masking effect acts like a data set with three important variables, where β_3 has a small effect.

	Horseshoe	Lasso	Relaxed Lasso	Forward
β_1	0.77	0.87	0.80	0.87
β_2	0.03	0.17	0.00	0.10
β_3	0.17	0.47	0.20	0.27
β_4	0.60	0.83	0.67	0.73
β_5	0.00	0.13	0.00	0.03
other β	0.33	2.17	0.40	0.40

Table 23: Average Inclusion of parameters by the different methods.

6.2.2 Aligned effects

The masking effect can only be identified if both variables are correctly diagnosed to have an effect. This makes the regression coefficients that are masked harder to find. On the other hand, when there is positive correlation between effects with the same sign, then one of the variables can explain part of the other variable's effect. Picking only one of the two correlated variables does have a minor cost of predictive power.

Data Generating Process 3: Aligned correlation

The data generating has 20 variables with regression coefficients as shown in Table 24.

	β_0	β_1	β_2	β_3	β_4	other β
Value	- 5.25	1	0.75	-1	-0.75	0

Table 24: Parameters of the data generating process

The explanatory variables X are drawn from a multivariate normal, with covariance matrix Σ.

$$X \sim MNV(\mathbf{0}, \Sigma)$$

With the following covariance matrix Σ:

$$\Sigma = \left[\begin{array}{cccc|c} 1 & 0.8 & 0 & 0 & \\ 0.8 & 1 & 0 & 0 & \\ 0 & 0 & 1 & 0.5 & \mathbf{0}_{4\times 16} \\ 0 & 0 & 0.5 & 1 & \\ \hline \multicolumn{4}{c|}{\mathbf{0}_{16\times 4}} & \mathbb{I}_{16} \end{array}\right] \tag{11}$$

The defaults are drawn from a Bernoulli distribution $y \sim \text{Bernoulli}(\theta)$ with:

$$logit(\theta) = \beta_0 + \beta X$$

The upper reference of the mlpd is:

$$\text{mlpd}_{\text{datagen}} = -62.49 \cdot 10^{-3}$$

When using the parameters from the data generating process the predictions are better than in the case of the masking effect and independent predictors. The lower limit remains the same, because the average amount of defaults remain the same.

$$\text{mlpd}_{\text{random}} = -98.02 \cdot 10^{-3}$$

Predictive performance The performance of Relaxed Lasso is highest for this data generating process. However there is only a small difference with the Lasso. In this case the Horseshoe with Predictive Projection performs slightly worse. Predictive Projection gives the most sparse model and the Lasso gives the least sparse model.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) (×1000)	-70.32	-69.65	-69.45	-71.18
sd(mlpd) (×1000)	1.92	2.13	2.48	2.45
# var.	2.66	6.51	3.09	3.09

Table 25: Hold-out-performance (mlpd) and Average number of included variables (# var.) different selection models on collinear data with aligned effects. The true amount of variables is 4

In the previous data types, Forward Selection had a best submodel with relatively good performance, but in this case it has the worst performance. Relax Lasso still has the best submodel with the highest performance.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean($\text{mlpd}_{\text{best}}$) (×1000)	-68.03	-68.52	-66.87	-69.26
mean(Δmlpd) (×1000)	2.28	1.15	2.53	1.93
sd(Δmlpd) (×1000)	1.77	1.30	2.30	2.82

Table 26: Difference between best submodel and selected submodel

Estimated mlpd For this simulation all the out-of-sample performances are overestimated by psis-loo and k-fold cross validation. However, Predictive Projection and Lasso are again more conservative than the mlpd_{est} belonging to the selected submodels by Relaxed Lasso and Forward Selection. The standard deviation of this difference is larger as well.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) (×1000)	-1.97	-1.44	-2.70	-3.65
sd(ϵ) (×1000)	6.35	5.30	7.02	7.96

Table 27: Difference between Estimated performance and real performance

Selected Variables The relative bad performance of Forward Selection is a result of the variables it chooses. Together with Predictive Projection it has highest false inclusion rate, however it also has a relatively high false exclusion rate compared to Relaxed Lasso and Predictive Projection. Lasso again has the highest false inclusion rate.

	Horseshoe	Lasso	Relaxed Lasso	Forward
β_1	0.56	0.97	0.80	0.59
β_2	0.62	0.91	0.68	0.59
β_3	0.85	0.97	0.91	0.88
β_4	0.47	0.93	0.53	0.59
other β	0.18	2.65	0.18	0.47

Table 28: Average Inclusion of parameters by the different methods.

6.2.3 Correlation with unimportant predictors

The masking effect and aligned effects occurs when two important variables are correlated. In this simulation I look at the effect of correlation between unimportant and important variables.

Data Generating Process 4: Correlation with unimportant variables

The data generating process has regression coefficient β as shown in Table 29.

	β_0	β_1	β_2	β_3	β_4	other β
Value	- 5.25	1	0.75	-1	-0.75	0

Table 29: Parameters of the data generating process

The explanatory variables X are drawn from a multivariate normal, with covariance matrix Σ.

$$X \sim MNV(\mathbf{0}, \Sigma)$$

The correlation among the variables is as follows:

- X^1 has a correlation of 0.8 with X^5, X^6, X^7, X^8, X^9.
- X^2 has a correlation of 0.8 with $X^{10}, X^{11}, X^{12}, X^{13}, X^{14}$.
- X^3 has a correlation of 0.8 with $X^{15}, X^{16}, X^{17}, X^{18}, X^{19}$.
- X^4 and X^{20} are uncorrelated with other variables in X.

The defaults y are drawn from a logistic model.

Two references, with the low point being random guessing whether a loan will default or not:

$$\text{mlpd}_{\text{random}} = -98.04 \cdot 10^{-3}$$

The other reference is the prediction on the hold out data with the data generating parameters:

$$\text{mlpd}_{\text{datagen}} = -72.52 \cdot 10^{-3}$$

The collinearity of the other data generating process changed the information that was available in the data set. For this data generating process the available information is the same as for the independent explanatory variables.

Predictive performance All the selection procedures have a worse performance than in the case of independent explanatory data. So the correlation makes it harder for all methods to find the important variables. Especially, the performance of Forward Selection and Relaxed Lasso is much worse. (Table 30). For the previous data types, this effect was less pronounced.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) ($\times$1000)	-80.92	-81.79	-87.20	-88.30
sd(mlpd) ($\times$1000)	3.87	4.16	6.19	6.05
# var.	4.20	7.90	3.60	2.93

Table 30: Hold-out-performance (mlpd) and Average number of included variables (# var.) different selection models for full correlation matrix. The true amount of variables is 4

Part of the worse performance, compared to the independent predictors, can be explained by the difference in the performance of the best submodels ($\text{mlpd}_{\text{best}}$). The performance of the best submodel is about $3 \cdot 10^{-3}$ to $5 \cdot 10^{-3}$ mlpd points worse than for the independent case. So far the

best submodels of Relaxed Lasso and Forward Selection have a better predictive performance than those of the Horseshoe and the Lasso, but in this case the best submodel of the Forward Selection performs worse than the other best submodels. Both Relaxed Lasso and Forward Selection have a high error in selecting the right submodels, which makes them perform poorly.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean($\text{mlpd}_{\text{best}}$) ($\times$1000)	-79.24	-79.61	-79.90	-80.62
mean(Δmlpd) ($\times$1000)	1.68	2.18	7.30	7.68
sd(Δmlpd) ($\times$1000)	2.45	3.65	6.24	7.06

Table 31: Difference between best submodel and selected submodel

Estimated performance In this type of data, the estimated performance of Forward Selection is again less conservative than that of Predictive Projection and Lasso. In table 32 there occurs something unusual. The estimated performance of Relaxed Lasso underestimate the real performance, while for all other procedures the estimated performance overestimates the real performance. In the previous data types the Relaxed Lasso always was less conservative than Lasso and Predictive Projection. The reason for this value is not clear to me. Because the bad predictive performance of Relaxed Lasso and a lack of time I did not investigate this further.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) ($\times$1000)	-2.28	-0.67	2.88	-4.44
sd(ϵ) ($\times$1000)	5.53	5.11	5.40	6.05

Table 32: Difference between Estimated performance and real performance

Selected Variables The cause of the poor performance of Relaxed Lasso and Forward Selection can also be explained by the selected variables. The correlation in the explanatory variables makes it harder for the methods to find the correct variables. Regression coefficient β_4, which is uncorrelated with other variables, is included more often than β_2, which is correlated with four unimportant variables. The Horseshoe prior and Lasso have a higher inclusion rate of the important variables. Predictive Projection includes the most important variables compared to the amount of variables that are correlated with the important variables (Table 33).

	Horseshoe	Lasso	Relaxed Lasso	Forward
β_1	0.86	0.83	0.66	0.59
corr. with β_1	0.31	1.45	0.45	0.38
β_2	0.57	0.52	0.28	0.21
corr. with β_2	0.59	1.65	0.62	0.45
β_3	0.79	0.90	0.62	0.45
corr. with β_3	0.34	1.48	0.55	0.38
β_4	0.69	0.86	0.41	0.41
β_{20}	0.03	0.24	0.03	0.07

Table 33: Average Inclusion of parameters by the different methods.

In this data generating process there are three types of variables, namely the important variables, the variables that are correlated with the important variables and unimportant variable β_{20}. It would be optimal to pick only the variables that are important. If the method picks a variable that is correlated with the important variable, instead of the important variable, this is suboptimal. However, it still gives a better prediction than not picking that variable.

	Horseshoe	Lasso	Relaxed Lasso	Forward
False Exclusion	0.99	0.89	2.03	2.44
False Inclusion				
- Corr.	1.24	4.59	1.62	1.21
- Uncorr.	0.03	0.24	0.03	0.07

Table 34: False inclusion and false exclusion of variables.

6.3 Misspecified Models

In all the previous example and simulation studies the assumption was made that the model was specified correctly. Both the model and the data generating process have log-odds of the Probability of Default θ that are a linear function of X. In simulated data the real relations are know, however, in real life data this is not the case. The assumption that the model is specified correctly might be too restrictive.

Example of model specification in credit data might be that a variable only has an effect is it reaches a certain threshold. So the correct relation would be to model the effect as a step function.

Other variables could have an effect that may not increase linearly, but it might be better to model a exponential relation, for example, the payment-to-income ratio (PTI). When this ratio is low, a slight increase of PTI, is unlikely to cause a lot more defaults. However, if the amount that has to be paid gets closer to the income of the household, a slight increase might cause a large change in the default probability.

Data Generating Process 5: Misspecified Model

The explanatory data is draw from a multivarate normal where the mean is zero and the covariance matrix is the identity matrix $\mathbb{I}_D$, with D=20.

$$X \sim MVN(0, \mathbb{I}_D)$$

There are two non-linear function, the first is the step function g.

$$g(X) = \mathbb{1}_{X \geq 0}$$

The second is the explonential function f.

$$f(X) = \exp(0.5X)$$

The regression coefficients are shown in table 35.

	β_0	β_1	β_2	β_3	β_4	other β
Value	- 5.25	-1	-0.75	-3	1	0

Table 35: Parameters of the data generating process

And functional relation of the data generating process is:

$$logit(\theta) = \beta_0 + \beta_1 X^1 + \beta_2 X^2 + \beta_3 f(X^3) + \beta_4 g(X^4)$$

For every data realisation 1,000 observation are drawn, where the defaults are drawn from a Bernoulli distribution.

$$y \sim \text{Bernoulli}(\theta)$$

When using the correct non-linear relations, as in the data generating process, to make predictions on the hold-out set, the performance of the model is:

$$\text{mlpd}_{\text{datagen}} = -70.18 \cdot 10^{-3}$$

the model performs worse than the data generating process, when the model is misspecified and only linear relations are assumed. I fit the a frequentist logistic regression with incorrect linear relations on 200,000 observation and the four important variables to get a reference value for the best linear model. The coefficients belonging to this regression are shown in Table 36.

Intercept β_0	β_1	β_2	β_3	β_4
-5.10	-0.97	-0.75	-0.82	0.74

Table 36: Regression coefficient of a linear model on data with 200,000 observations from data generating process 5

The out-of-sample predictive power for the best linear model is:

$$\text{mlpd}_{\text{best linear}} = -75.25 \cdot 10^{-3}$$

Like in all other cases the random model has a performance of:

$$\text{mlpd}_{\text{random}} = -98.03 \cdot 10^{-3}$$

Predictive performance Predictive Projection with a Horseshoe prior and Lasso perform better than Relaxed Lasso and Forward Selection under misspecification. Lasso again picks too many variables. The Horseshoe prior also picks slightly too many variables.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) ($\times$1000)	-82.49	-82.74	-84.78	-84.61
sd(mlpd) ($\times$1000)	4.02	3.04	6.26	7.55
# var.	4.20	6.60	3.70	3.70

Table 37: Hold-out-performance (mlpd) and average number of included variables (# var.) of selection models for a misspecified model.

The worse performance of Forward Selection and Relaxed Lasso is again caused by picking the wrong submodel as shown in Table 38

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean($\text{mlpd}_{\text{best}}$)($\times$1000)	-80.97	-81.11	-80.25	-80.59
mean(Δmlpd) ($\times$1000)	1.52	1.62	4.53	4.01
sd(Δmlpd) ($\times$1000)	2.52	2.30	4.58	5.28

Table 38: Difference between best submodel and selected submodel

Estimated mlpd On average the estimated performance of the selected models is higher than the real performance for all selection methods. For Predictive Projection and Lasso this overestimation is less severe.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) ($\times$1000)	-2.73	-1.32	-5.52	-5.82
sd(ϵ) ($\times$1000)	8.15	7.70	10.83	12.83

Table 39: Difference between Estimated performance and real performance

Selected Variables All the methods, except Lasso, pick approximately same amount of important variables. Predictive Projection with a Horseshoe prior picks more unimportant β than normally in this scenario.

	Horseshoe	Lasso	Relaxed Lasso	Forward
β_1	0.90	1.00	0.80	0.87
β_2	0.80	0.93	0.73	0.77
β_3	0.97	0.97	0.97	0.90
β_4	0.80	0.90	0.67	0.73
other β	0.73	2.80	0.53	0.43

Table 40: Average Inclusion of parameters by the different methods.

6.4 Non-normal Predictors

So far, all data generating process had normally distributed explanatory data. In real life data, variables are often not normally distributed. Data can be discrete, flags often indicate certain events or states in credit risk data. For example, what kind of house the debtor has.

Data can also be skewed, an example is the income distribution of a country. This also leads to skewed distributions in housing prices and loan sizes. Lastly variables can also have fatter tails than a normal distribution. In the data generating process 6, I consider these three different types of distributions.

Data Generating Process 6: Non-normal predictors

Data generating process with non-normal explanatory variables. All the variables are centred and scaled to a mean of zero have and a variance of one. The explanatory data is created as follows:

- X^2: Discrete variable, Bernoulli Variable with a probability of 50 percent on -1 and 1.
- X^3: Asymmetric, Centred Exponential distribution with a rate of 1.
- X^4: Fat tail, Normalised Student-t distribution with a degree of freedom of 4.

All other variables in X are drawn from a normal distribution with a standard deviation of one.

	β_0	β_1	β_2	β_3	β_4	other β
Value	- 5.25	1	0.75	-1	-0.75	0

Table 41: Parameters of the data generating process

Predictive performance For the last data simulated data type, the performance of the variable selection methods is similar to other simulated data. Horseshoe gives better performance with a sparse model, Lasso has gives model with good performance with too many variables. Relaxed Lasso is in third place and Forward Selection is the worst method.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) ($\times$1000)	-79.23	-78.51	-80.62	-81.33
sd(mlpd) ($\times$1000)	4.75	2.93	7.26	7.91
# var.	2.83	6.50	3.40	3.03

Table 42: Hold-out-performance (mlpd) and Average number of included variables (# var.) different selection models for non-normal explanatory data.

The performances of the best submodels is are now approximately equal and the selection error causes the poor performance of Forward Selection and Relaxed Lasso.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean($\text{mlpd}_{\text{best}}$)($\times$1000)	-76.67	-76.87	-76.51	-76.27
mean(Δmlpd) ($\times$1000)	2.56	1.63	4.11	5.05
sd(Δmlpd) ($\times$1000)	2.98	2.10	5.66	5.95

Table 43: Difference between best submodel and selected submodel

Estimated mlpd The estimated mlpd of the Horseshoe and Lasso regression are again more conservative and with have a lower standard deviation than those of the Relaxed Lasso and Forward Selection. In this simulation the estimated performance on average is closer to the real performance for Forward Selection and Relaxed Lasso.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) ($\times$1000)	2.29	3.35	-0.22	-0.58
sd(ϵ) ($\times$1000)	5.72	6.01	10.51	11.74

Table 44: Difference between Estimated performance and real performance

Selected Variables Regression coefficient β_3 that correspond to the skewed explanatory variable X^3, is picked less than β_1, even though they have the same effect size. The reason this difference is not clear.

	Horseshoe	Lasso	Relaxed Lasso	Forward
β_1	0.93	1.00	0.97	0.90
β_2	0.70	0.90	0.70	0.63
β_3	0.50	0.77	0.47	0.53
β_4	0.67	0.97	0.83	0.77
other β	0.03	2.87	0.43	0.20

Table 45: Average Inclusion of parameters by the different methods.

6.5 Final Remarks

Depending on the type of data, some variable selection method perform better than other. When the explanatory data is drawn from a independent multivariate normal, all method perform approximately the same. For harder data, for example correlated explanatory variables, the difference in the predictive performance becomes more apparent. In general, Forward Selection performs worse than the Horseshoe prior and the Lasso regression. Especially in the case where multiple variables have a strong correlation with each other. Relaxed Lasso sometimes performs similar to Lasso and Predictive Projection, but on other occasions performs badly.

Lasso and Predictive Projection give a relatively stable estimate over multiple realisations of the data. The variability of the predictive performance of these methods are often lower than those of for the other methods.

There is also a clear difference for the estimated performance between the variable selection method. The mlpd_{est} for the Horseshoe prior and Lasso are generally more conservative than the mlpd_{est} for Relaxed Lasso and Forward Selection. The standard deviation $\text{sd}(\text{mlpd}_{\text{est}} - \text{mlpd}_{\text{real}})$ was in general lower for Predictive Projection and Lasso.

The Horseshoe prior has the lowest false inclusion rate, combined with its other properties, makes it the most attractive variable selection method.

The Horseshoe is clearly the slowest method, it takes approximately 35 minutes to do one run. While the other methods all take under a minute.

7 FreddieMac Data

In this chapter I apply the feature selection methods to real life data. This data contains information on single-family mortgages in the United States. These mortgages finance the purchase of houses. The data comes from the website of FreddieMac, which is a government sponsored company which operates on the secondary mortgages market. The goal of the company is to provide, among others, liquidity and stability in the housing market.

The loans in the data set have fixed interest rate and the original loan term is between 25 and 35 years (Freddie Mac, 2019). The mortgages have a annuity amortisation scheme, which means that the payments of the debtor to the bank are equal for every month.

7.1 Variables in the Data Set

In the introduction I quickly presented the notion of Through-the-cycle and point-in-time estimates. The goal of the point-in-time estimates is to give an as good as possible estimate of the PD, and macro variables can be included to make a prediction. This approach leads to procyclicality, and therefore a Through-the-Cycle estimate is more preferable. This means that in this section no macro-economic variables are used. The data analysis is only done on the data available at the FreddieMac website.

The data is organised in two types of files. The first type relates to the state of the loan when it was taken out. An example of variables in these files are the original Unpaid Principal Balance (UPB), which is the original amount of the loan. Other information like the original maturity term and the fixed interest are also present.

In the other types of files, the FreddieMac data set also contains variables that are updated every month. An examples is the delinquency status, that is the time (in months) that the debtor does not meet its financial obligation to the creditor.

In the monthly data there are various variables that are related to costs of debtors that have defaulted. For example, legal cost, maintenance cost and cost associated with reorganising the loan structure. These can be used for the Loss Given Default, however these variable are not of interest for the modelling of Probability of Default. Therefore, I do not take these variables into consideration.

Besides the variables present in the original data set, I also combine certain variables to create new predictors. The goal of the model is to estimate the Probability of Default of the loans. However, the defaults are not present in the data set. I define the default to be a loan that has a delinquency status of 3 months or higher.

The first created explanatory variable is the Payment-to-Income ratio. This is ratio of the payment can easily be calculated as the sum of the paid amortisation and interest. Another variable I create is Prepayment. This happens when a debtor pays a higher amount of amortisation than the agreed amount. If the actual UPB is lower than a contractual UPB then a prepayment happened at time t. The contractual UPB is not present in the data set, but can be calculated with the annuity formula:

$$UPB_{t,\text{contratual}} = \frac{1-(1+r)^{t-T}}{1-(1+r)^{-T}} UPB_0$$

Where UPB_0 is the original UPB, t is the loan age, T is the original maturity and r is the interest rate.

An overview of the variables is shown in Table 46

Variable	Type	Description
Age ratio	Continuous	Age of the loan to its original maturity
UPB ratio	Continuous	Unpaid Principal Balance as a percentage of the original amount
FICO	Continuous	Credit Score provided by the Fair Isaac Cooperation (FICO).
Borrowers	Discrete	0 for a single borrower, 1 for multiple borrowers
Units	Discrete	0 for single-unit, and 1 for multi-unit. A single unit house is intended for one family, a multi-unit house is intended for multiple families.
cltv	Continuous	Cummalative amount of loan to original value of the collateral (house)
MI ratio	Continuous	Mortgage insurance as percentage of potential incurred loses
Log(Income)	Continuous	Logarithm of the income when loan was taken out.
Debt-to-Income	Continuous	Ratio of current UPB to original income.
current UPB	Continuous	Current Unpaid Principal Balance in $100.000
Purchase	Discrete	If the goal of the loan is to buy a house, then this variable is 1 and 0 otherwise.
Primary Residence	Discrete	Indicator whether the house is the primary residence of the debtor.
First Home	Discrete	Indicator whether the house is the first home of the debtor.
Prepayment	Discrete	Indicator whether the debtor paid more amortisation than agreed upon in the payment plan.
Payment-to-income	Continuous	Ratio of interest and amortisation to original income
Delinquency status	Discrete	Month of unpaid interest and/or amortisation
Super Conforming	Discrete	Indicator whether the loan is super conforming. Super confirming loans have higher permitted maximum loan limits designated for high cost areas.

Table 46: Explanatory variables for the FreddieMac data set (Freddie Mac, 2019)

7.2 Preprocessing & Sampling Procedure

The full FreddieMac data set contains 26.6 million observation. The data has monthly observation and spans 20 years. I only take a small portion of this data, because I am interested in the low information variable selection setting. Firstly, I reduce the size of the data set by sampling 300,000 loans and discarding the rest. Under Basel III a minimum of 5 years of data is required for PD modelling (Basel Committee on Banking Supervision, 2017).Therefore, I only keep the observations of these loans that occurred in 2010 to 2015 to meet this criterion.

Some variable have missing values. I drop observations that contain missing values. In a real life situation where data is scarce, this is not the ideal approach, because you lose information by dropping out observation that only miss few variables. Instead, a model of missingness can be made for the approximation of the missing data. I do not do this as it is not the goal of the thesis and the amount of missing variables is low (Table 47).

FICO	DTI	cltv
0.02%	0.54%	0.03%

Table 47: Percentage of missing variables in the data set. The other variables do not have missing values

The loans still have monthly observations, and the goal is to make a yearly prediction. For each loan I draw a random month. This turn the data into the right time format and the random month counteracts seasonal effects.

In this step I create the variables that are not present in the original data set as discussed in section 7.1.

The next step is to split up the data set into a test set and a raw training data set, such that they both contain 2% defaults. This is the same percentage as in the simulation studies. There are approximately 100.000 observations in both data set. The performance of the different methods is evaluated on the test set. From the raw training data I draw the small training data set with 1,000 observations.

Many variables are the same for all observations of the loan, because they represent the loan at its origin, for example the original UPB. This causes repetition in the data. Furthermore, the loans that did not default have on average more observations than the loan that did default.

The raw data set contain many loans with a maximum of 5 years of observation with 2% defaults and 98% non-defaults. From each loans I draw one observation and take 1,000 random loans, where I force the set to have 20 defaults and 980 non-defaults. This means the different observation can be from one of the 5 years, but there is only one observation per loan.

The regularisation depends on the scale of the explanatory variables X. Some parameters in the data set have small values, while others have values in the hundreds. Variables that have a large value would be more regularised than variables with smaller values. For an even regularisation over all variables, I normalise all variables, such that they all have a mean of zero and a standard deviation of one.

7.3 Single Run Example

In this section I take one realisation of the FreddieMac data and go through the selection method as an example. I refer to the realisation of the FreddieMac data as the training set. Figure 36 shows the marginal distributions of variables in the training set. Some variables, like Log(Income) and Debt-to-Income are somewhat similar to a normal distribution. However, most variables are clearly non-normal. There are skewed variables, such as the FICO score. Furthermore, there are many explanatory variables that are discrete. Some of these variables, like delinquency status, are imbalanced.

The Kendall τ correlations of the variables are shown in Figure 37. Most variables only have a moderate correlation, but the data set also contains some strong correlations. One example is the UPB% and the Age%, as the age of the loan increases, the value of the loan decreases due to payments by the debtor. This relation is not perfect as the debtor might do a prepayment. On the other hand, the debtor might take out a additional loan on the same mortgage and then the UPB increases.

The Kendall correlation between UPB and log(UPB) is equal to one, because the logarithm is a monotonic increasing function. So if UPB goes up, then the log(UPB) also increases. Even though, the Kendall τ correlation is one, the two variables are different.

The rest of the correlation coefficients make intuitive sense. The only unforeseen value is between cumulative loan to value (CLTV) and Mortgage Insurance as a percentage of potential losses. A possible explanation is that banks demand a mortgage insurance for people who have a high CLTV. When the size of the loan is small compared to the value of the collateral and the debtor default, the bank can minimise its loses by selling the collateral. The bank might not be able to cover its loses by selling the house if the CLTV is high.

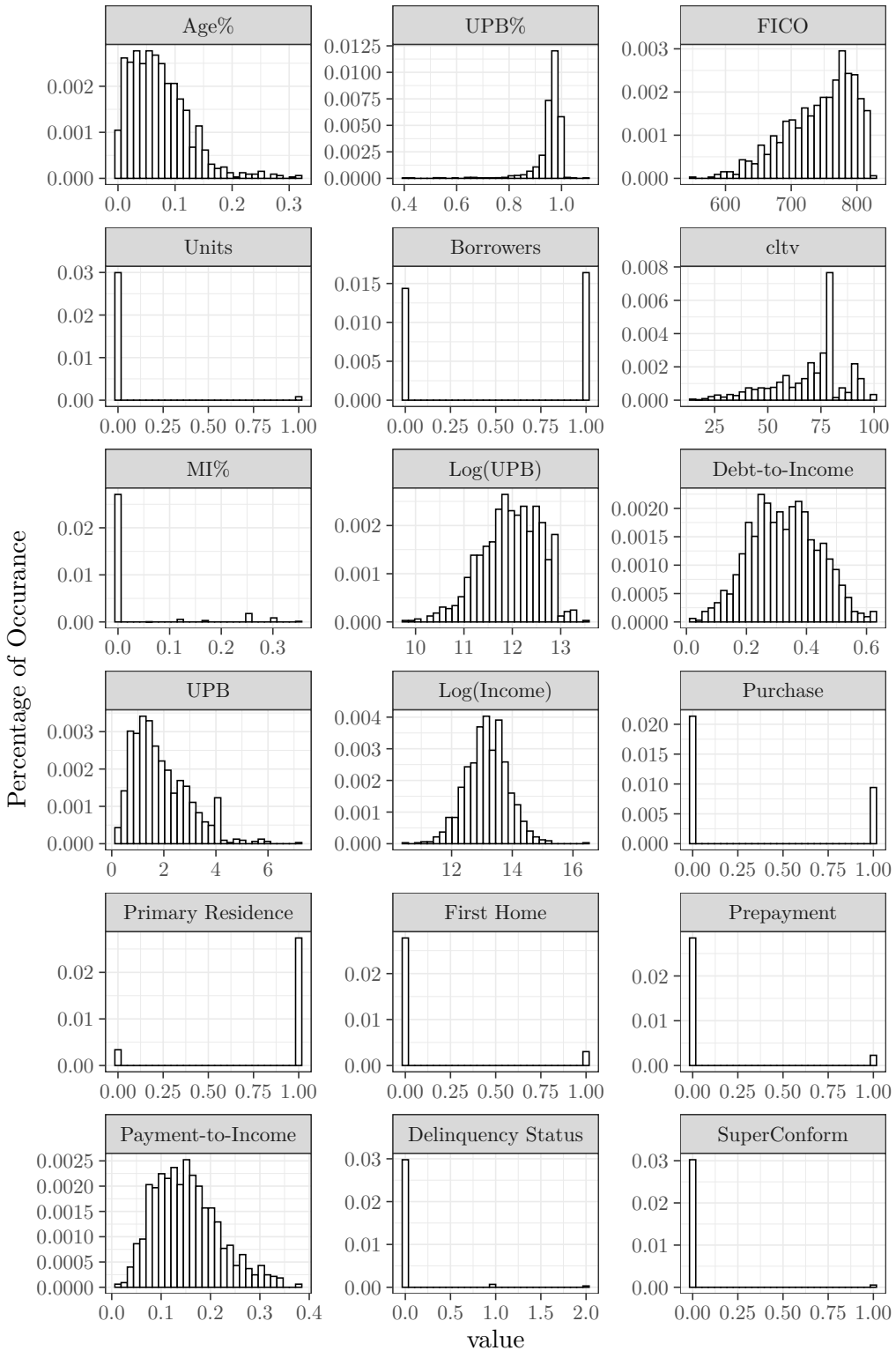

Figure 36: Marginal distributions of the explanatory variables in the FreddieMac data set.

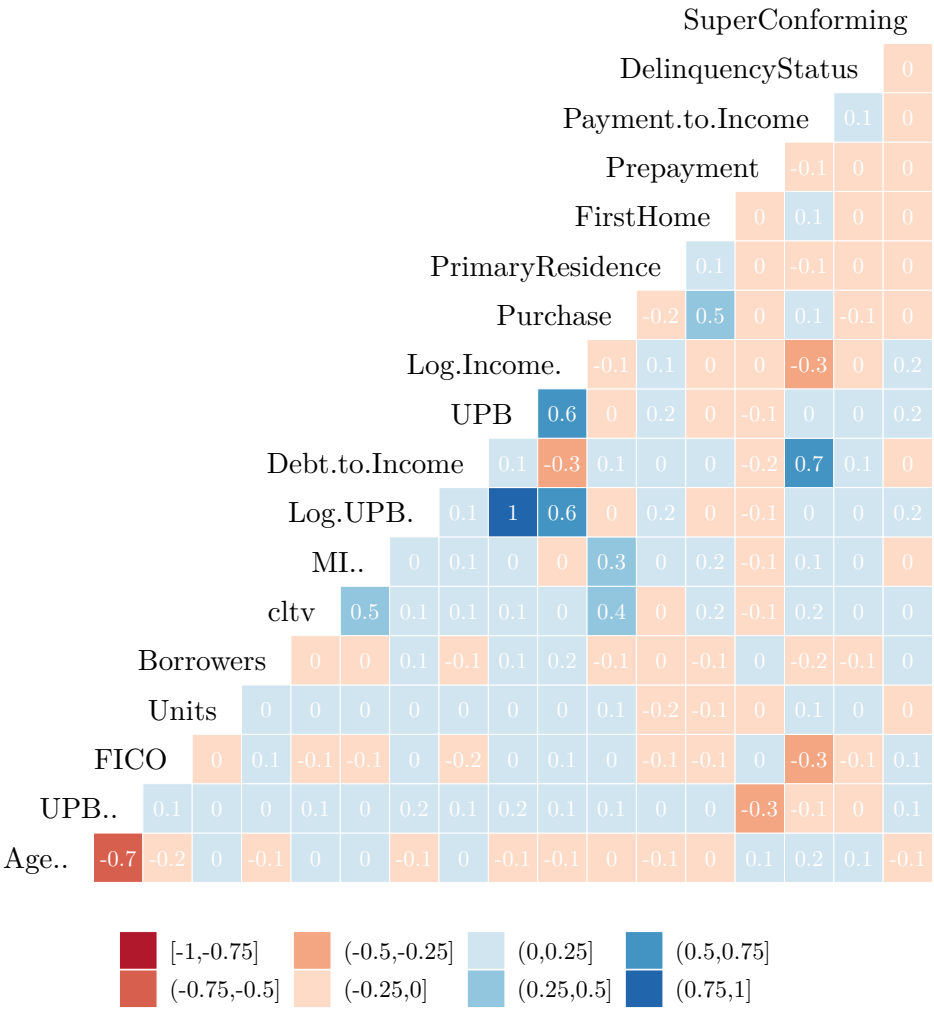

Figure 37: Kendall τ correlation of the explanatory variables in the training set.

The correlation matrix only shows the relation between two variables. The relations in the data can be more complex than the correlation might suggest. For example, the Prepayment is a value that can be deterministically calculated from the UPB%, Age%, Maturity time and interest rate. Figure 38 shows this interaction. For the prepayment variable this is easy to determine, because I constructed it myself. This might be hard to detect in other variables.

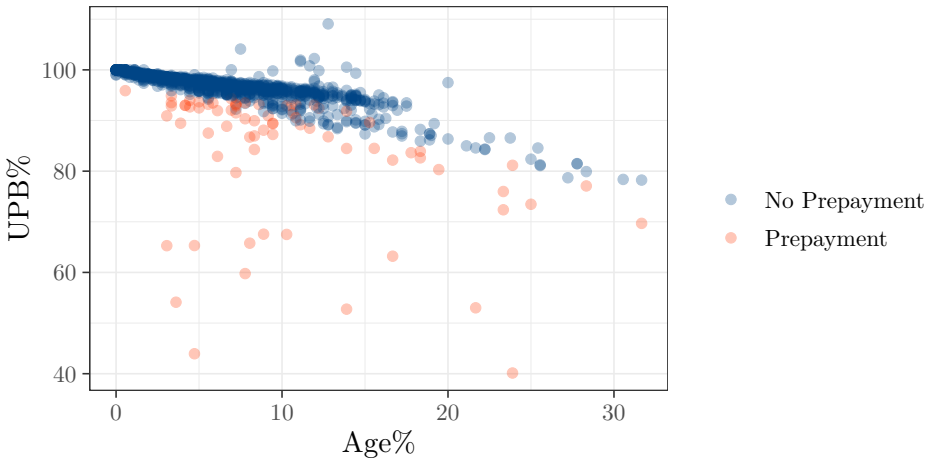

Figure 38: The relation between prepayment, UPB% and Age % is more complex than a pairwise relation.

7.3.1 Variable Selection

First I consider the Horseshoe prior with Predictive Projection. The marginals of the posteriors of the logistic regression with the Horseshoe prior is shown in Figure 39. The correlation in the data set causes the marginal posteriors to be wide an some variables have heavy tails. The posterior belong to the super confirming flag even has -4 in its credible interval. All the 95% credible intervals of the regression coefficients, except for the Delinquency status, include zero.

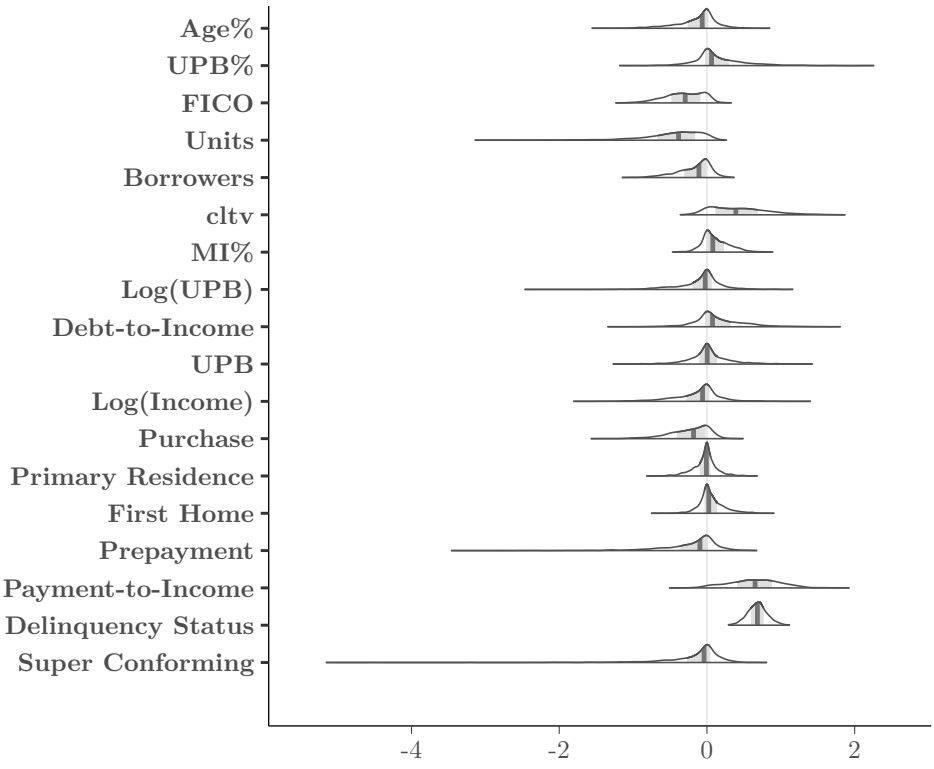

Figure 39: Posterior of logistic regression with a Horseshoe prior on FreddieMac data.

Predictive Projection determines that only Delinquency and the Payment-to-Income ratio are important variables. Where Delinquency is the most important variable, and PTI the second

most important variable. The projected posterior of the Payment-to-Income ratio, in Figure 40, is much narrower than the unprojected posterior. The posterior of the Payment-to-Income ratio a standard deviation that is about twice as big as the standard deviation of the Delinquency status, 0.21 instead of 0.11. The standard deviation of the intercept is 0.34.

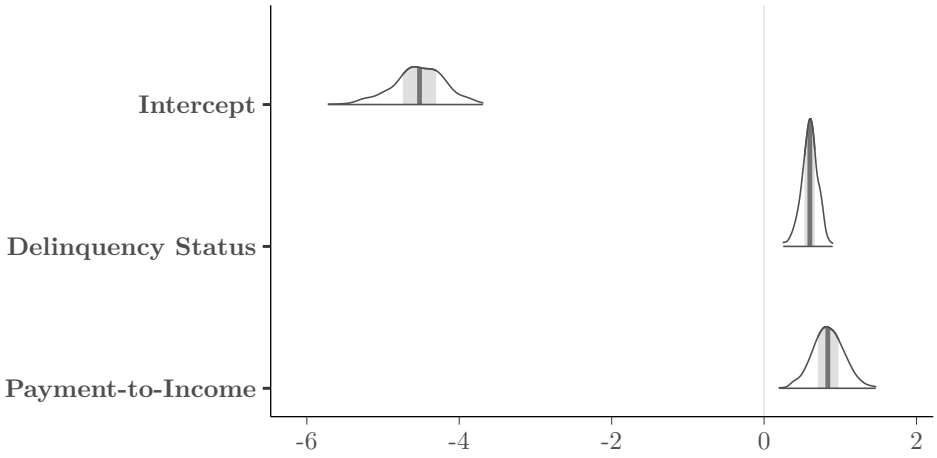

Figure 40: Projected Posterior of the two important variables according to Predictive Projection.

Figure 41 shows the Lasso regression on the training data. Lasso agrees with Predictive Projection that Delinquency status and Payment-to-Income ratio are the two most important variables. Just like in the simulation studies, Lasso Regression includes more variables in the model. Some variables are quite stable over different values of the regularisation parameters λ. While the regression coefficients of cltv and *Units* have a steeper descent.

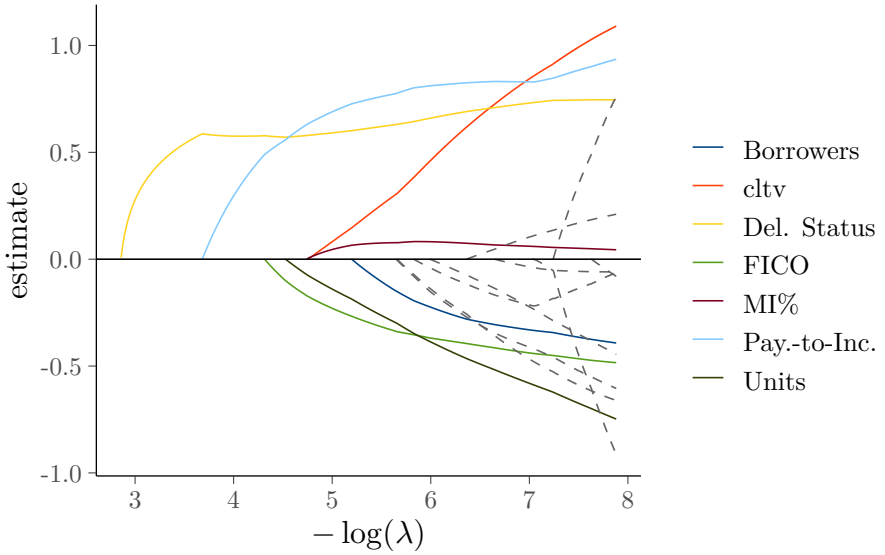

Figure 41: Lasso regression on the training set. The coloured lines are the included variables and the dotted black lines are the excluded variables.

I also apply the Relaxed Lasso and Forward Selection to the training data and test their performance on the hold-out set. The resulting models and their performances are shown in Figure 42. The lines have the typical pattern for the methods. Predictive Projection with the Horseshoe greatly increases in performance for the first two added parameters and remains relatively stable after adding more variables. Lasso also predicts better at first and slowly deteriorates with added

variables. The performance of Relaxed Lasso and Forward Selection increase quickly as well, but deteriorates the quickest.

Figure 42 shows the out-of-sample performance of the variable selection procedures.

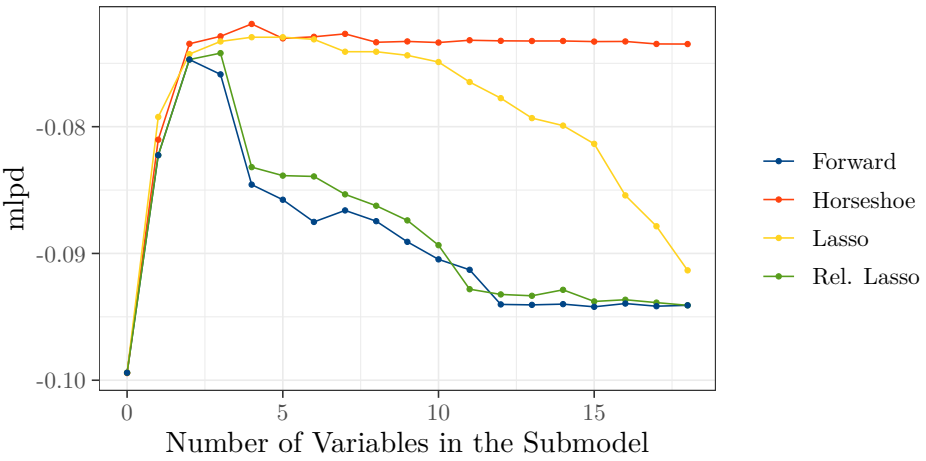

Figure 42: Predictive performance on hold-out-set for variable selection methods

Table 48 shows the performance and number of variables resulting from the feature selection. Horseshoe prior with Predictive Projection has the best predictive performance. It also gives the smallest model, together with Forward Selection.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mlpd (×1000)	-73.46	-74.08	-83.19	-74.27
#var.	2	7	4	2

Table 48: Predictive performance and number of variables for the variable selection methods.

The steep drop in performance for Relaxed Lasso and Forward Selection is a result that both models pick *Units* as the fourth important variable. This is an imbalanced variable and in the multiple *Units* only occurs when there is no default. The regression coefficients in the model found by the Relaxed Lasso method has a large value (Table 49). Even though there is a relation between defaults and Units, it is not likely that it is this large. When projecting the posterior (of Horseshoe) to have 5 variables, it also includes the Units variable. The expected value of that posterior $\mathbb{E}[\beta_{\text{Units}}|X] = -0.42$. Lasso, in Table 49, estimates the regression coefficient to be $\hat{\beta}^{\text{Lasso}}_{\text{Units}} = -0.30$. When adding this variable to model, projection prediction has a slight drop in performance and Lasso has a slight increase in the performance as shown in Figure 41.

The difference between Predictive Projection, Lasso, Relaxed Lasso and Forward Selection is that the first two methods are regularised, while the latter two are not. Thirteen loans are for more than one *Unit*, therefore, the likelihood of the corresponding regression coefficient is wide. This means that both Lasso and Predictive Projection heavily shrink the regression coefficient to zero.

	Intercept	Delq. sts	Pay.-to-Inc.	FICO	Units	cltv	MI%	Borrowers
Horseshoe	-4.53	0.60	0.84	-	-	-	-	-
Lasso	-4.83	0.63	0.77	-0.34	-0.30	0.30	0.08	-0.15
Rel. Lasso	-5.18	0.66	0.94	-0.46	-1.64	-	-	-
Forward	-4.78	0.63	1.00	-	-	-	-	-

Table 49: Estimate of the regression coefficients for the variable selection methods. The Horseshoe estimate is the expected value of the posterior.

7.4 Multirun

Like in the simulation studies, I use multiple training set to compare the variable selection methods. From the big data set I draw samples to get a training set with a size of 1,000 observations and 20 defaults. This is done 30 times. The average performance are shown in Table 50. The performance of Predictive Projection and Lasso regression perform the best and only differ a little. The performance of the Relaxed Lasso is performs worse, but is still better than Forward Selection.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(mlpd) ($\times$1000)	-73.98	-73.90	-76.13	-79.02
sd(mlpd) ($\times$1000)	2.71	2.07	5.73	6.37
# var.	3.17	5.00	2.43	2.47

Table 50: Hold-out-performance (mlpd) and Average number of included variables (# var.) different selection models on FreddieMac data

The best submodel of the Horseshoe prior and Lasso are better than the best submodels of Relaxed Lasso and Forward Selection. This behaviour also occurs in the data generating process 4, where the important variables have a strong correlation with other variables. Combined with the selection error the latter two methods perform badly. Especially the selection error in Forward Selection is big.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean($\text{mlpd}_{\text{best}}$)($\times$1000)	-72.76	-72.54	-73.66	-74.32
mean(Δmlpd) ($\times$1000)	1.22	1.35	2.46	4.70
sd(Δmlpd) ($\times$1000)	1.61	1.63	5.80	6.08

Table 51: Difference between best submodel and selected submodel

For all methods the estimated performance is overly optimistic about the real performance. As in the simulation studies, the mlpd of Horseshoe prior and Lasso are more conservative than Relaxed Lasso and Forward Selection. The errors have approximately the same standard deviation, except for the Relaxed Lasso, where the variability is higher.

	Horseshoe	Lasso	Relaxed Lasso	Forward
mean(ϵ) ($\times$1000)	-2.86	-2.51	-4.34	-7.50
sd(ϵ) ($\times$1000)	8.42	8.07	10.47	8.69

Table 52: Difference Estimated performance and real performance

7.4.1 Selected Variables

Table 53 shows the inclusion percentage of the feature selection methods. All method agree that the Delinquency status is an important parameters in the data set, as well as the original payment-to-income ratio. The payment-to-income ratio is a variable that has some strong correlation with other variables. This could explain the reason why the best submodels of Predictive Projection and Lasso are better than those of the Relaxed Lasso and Forward Selection.

The third most important variable is the FICO credit score. Lasso picks this variable 70% of the time and Predictive Projection 37% of the time. The other two methods also pick this variable to a lesser degree. Table 53 suggest that a lot of variables are related to the defaulting in some degree. However, the predictive power might not be strong enough for the variable selection methods, due to the low amount of data on defaults.

	Horseshoe	Lasso	Relaxed Lasso	Forward
Age %	0	0	0	3
UPB %	7	10	0	0
FICO	37	70	27	13
Units	7	40	3	3
Borrowers	13	37	3	3
cltv	20	27	0	10
MI %	7	23	3	3
Log curr. UPB	3	10	0	0
Debt-to-Income	7	3	0	3
UPB	0	7	0	0
Log Income	7	10	0	10
Purchase	7	10	0	3
Primary Residence	3	20	3	7
First Home	7	13	7	3
Prepayment	3	10	3	7
Payment-to-Income	90	100	93	87
Delinquency status	100	100	100	90
Super Conforming	0	10	0	0

Table 53: Inclusion percentage of parameters

7.5 Final Remarks

As in the simulation studies, Predictive Projection and Lasso have the best predictive performances, which are roughly equal. Predictive Projection needs less variables than Lasso for this performance, making it the preferred method. These two method also give a more stable performance over multiple training data sets.

FreddieMac data has elements from various simulated data. The explanatory variables are correlated, they have non-normal distributions and it is likely that the models are misspecified. This combination is probably the reason why Lasso and Predictive Projection preform better on the FreddieMac data set, than the two other methods.

8 Conclusion & Discussion

In this thesis I investigated variable selection methods in Probability of Default models, where there are few defaults in the data set, but there are relatively many potential predictors. In this setting, using all variables directly leads to overfitting. Variable selection are intended to counteract overfitting and give insight into the predictors of default risk. I compared four variable selection methods on a logistic regression, namely Predictive Projection on a Horseshoe priors, Lasso regression, Relaxed Lasso regression and Forward Selection.

8.1 Conclusion

The Horseshoe with Predictive Projection is a promising method for variable selection. Predictive Projection and Lasso give more stable and similar or better predictions on all data types compared to Relaxed Lasso and Forward Selection. Predictive Projection on a Horseshoe prior gives sparser model than Lasso, which makes the model more interpretable. Therefore, the Horseshoe prior with Predictive Projection is a more attractive method than Lasso, Relaxed Lasso and Forward Selection.

Besides the performance, Predictive Projection and Lasso have more conservative k-fold/psis estimated performance than Relaxed Lasso and Forward Selection. The standard deviation of the difference between the estimated and real performance is also smaller. Therefore, Predictive Projection and Lasso give a better insight into the real predictive performance of the model.

The main drawback of Predictive Projection is that it is slow compared to the other methods. For the 1,000 observations and 20 explanatory variables it takes approximately 35 minutes to run compared to seconds for Lasso. Nevertheless, this time is negligible compared to the amount of times it takes to implement an operational PD model. Furthermore, the time is of minor importance in comparison to the advantage of the methods.

Table 54 contains an overview of the relative advantages and disadvantages of the variable selection methods.

	Horseshoe	Lasso	Rel. Lasso	Forward
Performance				
Independent	+	+	+	-
Collinnearity				
Weak	+	+	+	-
Strong	+	+	-	-
Misspecified	+	+	-	-
Non-normal	+	+	-	-
FreddieMac	+	+	-	- -
Variability of Performance	+	-	-	+
Sparsity	+	-	+	+
Computation Time	- -	++	++	+

Table 54: Relative advantages and disadvantages of variable selection methods, + is better , - is worse.

This conclusion is based on the research questions, the simulations study and an analysis on FredieMac data.

8.1.1 Research Questions

How does Bayesian variable selection, with a Horseshoe prior and Predictive Projection, compare to Forward Selection, Lasso variable selection and Relaxed Lasso variable selection in simulated PD data?

- **Performance:** In some simulation the Horseshoe with Predictive Projection has the best results and in others Lasso regression has the best predictive performance. From these simulation the difference between these methods cannot be determined.

In some easy data types, where only low amounts of collinearity was present, Forward Selection and Relaxed Lasso had similar performances to Lasso and Predictive Projection. In the case of more collinearity, misspecification and non-normality, the performance of Forward Selection and Relaxed Lasso is worse than the performance of Horseshoe and Lasso.

The best submodels of Forward Selection and Relaxed Lasso are in general better than the best submodels of Lasso and Predictive Projection. The problem with Relaxed Lasso and Forward Selection is that they are more likely to pick a suboptimal submodel. Except for the data with a masking effect, Forward Selection and Relaxed Lasso have a larger predictive difference between the best submodel and the selected submodel (Δmlpd). This difference was about 50% to 100% bigger for most data types.

- **Variability of Performance:** Besides good performance, the variability of the performance is important.

  The performance of Forward Selection and Relaxed Lasso is more volatile than the performance of HS and Lasso. The standard deviation of the former methods is about 50% to 200% bigger in the simulations, depending on the type of data.

- **Sparsity:** Predictive Projection, Relaxed Lasso and Forward Selection consequently included less variables in the models than Lasso regression.

- **Computation Time:** Lasso and relaxed are clearly the quickest methods. The whole procedure takes under a minute. Forward Selection is also a quick procedure and takes a about a minute. The Horseshoe with Predictive Projection is by a great deal the slowest method. Finding the posterior of the full model takes a couple of minutes and Predictive Projection takes the most time with 30-40 minutes.

How do the different methods perform on real life data?

- **Performance** The FreddieMac data has elements of various simulations data. The explanatory data of the FreddieMac data set is non-normal and has correlation. Furthermore, it is very likely that there is some functional misspecification present in the model. Similar to the simulation data, HS and Lasso perform well compared to the other methods. Especially, Forward Selection performs badly.

- **Variability**: The variability of the predictive performance is lowest for Predictive Projection and Lasso.

- **Sparsity:** Relaxed Lasso and Forward Selection produces the most sparse models, followed by Predictive Projection. However, the sparsity of Relaxed Lasso and Forward Selection come at the cost of predictive power.

- **Computation time:** Is the same as in the simulation studies.

Do PSIS-LOO and K-fold cross validation give good estimates for the real out-of-sample performance for the different variable selection methods?
The estimated mlpd of the selected model was on average, on almost all data types, is higher for Predictive Projection and Lasso than for Relaxed Lasso and Forward Selection. This means that the former estimated performance mlpd_{est} is more conservative for predictive prediction and Lasso regression. Let the difference between the estimated performance and real performance ϵ, defined as:

$$\epsilon = \text{mlpd}_{hold-out} - \text{mlpd}_{est},$$

The standard deviation of ϵ was lower for these Predictive Projection and Lasso. Which means that the estimated mlpd_{est} is a better estimate of out-of-sample performance for the Horseshoe and Lasso than for Relaxed Lasso and Forward Selection. On basis of this research I cannot conclude whether psis-loo for Predictive Projection and K-fold Cross Validation for Lasso are good, nevertheless I can say that they are better than K-fold Cross Validation for Forward Selection and Relaxed Lasso.

8.1.2 Practical Implications

The choice of the method depends on the exact desire of the model. When sparsity is not a real concern and time is scarce, then I would advise using the Lasso regression. For all the data, Lasso gave a relative good prediction. The estimated performance gives a relatively good insight into the real performance. Lasso can also be used as a quick first check of the potential predictive power of the data set. In this scenario it would also be possible to use the Horseshoe prior without Predictive Projection. The effects of the different parameters is not clear in the case of Lasso regression, as it shrinks the strong parameters to zero and the effect of unprojected posterior is not clear as well.

If both performance and sparsity are important, then the Horseshoe prior with Predictive Projection is the only option of selection methods.

If Bayesian methods are not desired and sparsity is of the importance, then I would suggest Relaxed Lasso. This however comes at a cost. Judging by the results of this thesis, I would not advise to use Forward Selection for data containing few defaults.

8.2 Discussion & Recommendations

There are certain problems associated with the low information feature selection setting that need to be addressed and investigated, before the techniques can be applied to predict the probability of defaults. I also discuss other problems in credit risk with might be solved by other techniques.

8.2.1 Misspecification

Misspecification is a serious danger for all types of modelling, because it is never possible to guarantee that the specified relations are true. The simulation studies suggest that the performance of the Predictive Projection and the performance of Lasso is better than that of Forward Selection and Relaxed Lasso. I suggest to further investigate the effect of misspecification on these variable selection methods.

8.2.2 Simulation studies

Estimated Performance I could not answer the question whether psis-loo and K-fold Cross Validation where good estimates for the out-of-sample performance when a variable selection method was used, and I could only say that the estimates where better for Predictive Projection and Lasso. To answer this question, a reference of the estimated performance needs to be determined. A way this could be done is by fitting the model on 1,000 observation and calculate psis-loo/k-fold cross validation mlpd. Now also look at the performance of the on various small hold-out set with 1,000 observations and finally test the performance of the model on a big hold-out set (for example 200,000 observations). If the estimation error of psis-loo and K-fold Cross Validation are same as the error of the small hold-out sets, than the psis-loo/k-fold mlpd are good estimates of the out-of-sample performance. If the bias and variability of psis-loo/K-fold could be kept to a minimum, then the K-fold cross validation for Lasso, and psis-loo for the Horseshoe prior could be used for an estimate of the out-of-sample performance. If no hold-out set is needed that would mean that the scarce data would not have to be split up, which might lead to better predictive performance. From my simulation studies I am not sure if this is the case. So I recommend further research in this topic.

mlpd of data generating processes I picked the data generating processes such that they had similar regression coefficients. Correlation between different important variables can change the potential information in the data set. For the aligned effects data set, the theoretical maximum mlpd is $-62.49 \cdot 10^{-3}$, while for the data set with the masking effect the maximum mlpd is $81.13 \cdot 10^{-3}$. This is a big difference in information and makes it harder to compare the performance between the data generating processes.

Due to the lack of information in the masking effect the different methods almost never pick the variables that cause the masking effect. This is not a big problem, because the two variable contain almost no information. However, this also means that the data generating process acts

like a data generating process with three important variables, where one of them has a very small effect.

To give a better comparison of the different types of data, I suggest to let the data generating processes be similar in the maximum predictive performance (mlpd), instead of keeping the regression coefficients the same. So for the masked data, this would mean increasing the size of β_2 and β_3.

Amount of simulations In the simulation studies, I applied the methods to 30 realisations of the data generating process per type of data. Due to time constrains I had to make the choice between using more data types or using more realisations. I chose the former. In some measures of the simulation studies there is a high variability, which makes it hard to be definite about these results. I suggest rerunning the simulation studies with more realisation to get a better view of the exact impact of the different methods. I ran everything on a laptop, but running this on a computational server this would go much faster.

8.2.3 Clustering in Predictive Projection

I used 10 clusters for the Predictive Projection in this thesis. The reason being that this greatly decreases the time of the simulations. For the simulations this was really necessary, because time was an important constraint. However, running Predictive Projection on a single data set with more clusters would not be a problem. Doubling the amount of clusters would increase the time from 35 minutes to a little over an hour.

During the writing of this thesis I once saw that increasing the amount of clusters from 10 to 20 really improved the performance of the method, because the method was more likely to pick the right parameters. I did not further investigate this, so I suggest exploring this.

I expect that using more cluster would be beneficial for the performance, when the posterior becomes more complex due to correlation. In the conclusion I stated that all the computation times where the same for all data types. However, if more collinearity in the explanatory data demands more clusters, than the computation time is different for different data types.

8.2.4 Hierarchical Models

All the methods that have been used are highly dependent on the particular realisation of the data. Because of the low probability of default, there is a high variance compared to the expected value. This problem cannot be solved by using these techniques. One approach that might solve this problem is to use expert judgement to correct the estimate in the frequentist framework. It would also be possible to use expert judgement in Bayesian statistics, which can be implemented via the prior.

Another approach is to combine multiple portfolio that have similar characteristics and use hierarchical logistic regression. Every data set has its own regression coefficients, however these are combined by a hyperprior. This makes it possible for the data sets to share information via the hyperprior (Gelman et al., 2013). Depending on the similarity of regression coefficients corresponding to one explanatory variable, the model automatically chooses how much information to share among the data sets. This often leads to a better performance than using a single regression over all the data or one logistic regression for every data set.

8.2.5 FreddieMac data

Sampling Methods In the thesis I used a simple sampling method from 1,000 loans , such that the training data contains 20 defaults and 1,000 observation. This sampling method needs 1,000 loans to get 1,000 training points. By using a more elaborate sampling procedure, the amount of loans might be less to get 1,000 observation. Or stated otherwise, you could get more than 1,000 observation from 1,000 loans.

8.2.6 Computation

Monte Carlo methods are slow compared to most frequentist methods that are being used. Frequentist methods quickly output results. Bayesian methods take longer, however, there is currently a lot of development in Monte Carlo methods, which might decrease the time of fitting a model. In Stan, a couple of these developments are that research is done such that graphical processing units (GPU) can be used. Parallelising the calculation could decrease the time to fit a model.

Besides Hamiltonian Monte Carlo, there is also a development in Piecewise Deterministic Markov Processes. These methods are non-reversible and are potentially faster than there reversible counterpart (Bierkens et al., 2018).

8.2.7 Different Model Types

The projpred pacakge is only two years old, and the developers are currently working on a an implementation for survival models. Survival models are also a common model in credit risk management. If would recommend investigating the application of Predictive Projection on these types of models.

A Markov Chain Monte Carlo Methods

In Bayesian statistics the posterior of the model is found by using Bayes' Formula:

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

Let $\theta := \{\theta_1, ..., \theta_D\}$ be a D-dimensional vector of parameters and $y = \{y_i, ...y_N\}$ be a vector containing N data points. In some models it is possible to calculate the result of this equation analytically, however this is not true in general. In the logistic regression, for example, the resulting $p(\theta|y)$ does not have a closed form. The same is true for models with a Horseshoe prior (see section 5.2.3).

Still in most cases it is easy to calculate the values for the prior $p(\theta)$ and the likelihood $p(y|\theta)$. But the problem with the normalising constant $p(y)$ remains.

After finding the posterior we are interested in finding statistics to summarise the distributions on θ. There is a wide class of summary statistics which depend on an integral over the distribution of θ and a function $h : \theta \to \mathbb{R}$. One example of such a statistic is the mean of the parameters, this is the case when $h(\theta) = \theta$ and is calculated by the following integral.

$$\int h(\theta)p(\theta|y)d\theta = \frac{\int h(\theta)p(y|\theta)p(\theta)d\theta}{\int p(y|\theta)p(\theta)d\theta}$$

Other examples which can be calculated using such as the variance, which uses $h(\theta) = \theta^2$ among other things. Interval probabilities in $[a, b]$ can be be calculated by $h(\theta) = \mathbb{1}_{[a,b]}(\theta)$. Where $\mathbb{1}_{[a,b]}(\theta)$ is equal to one if $\theta \in [a, b]$ and zero elsewhere. This also includes a histogram of the posterior, which is a graphical representation of interval probabilities.

If it was possible to draw samples from the distribution, then we could use these samples to calculate, for example, a mean of the distribution by averaging the samples.

One method is to draw samples is to create a Markov Chain $\{\theta^0, .., \theta^S\}$ which explores the parameter space Ω. Every θ^s is a draw form the parameters space. A Markov Chain is a sequence of events, where the next event only depends on the previous event. The chain is created by using a transition kernel $T : \Omega \to \Omega$, which is a probability function such that $\theta^s = T(\theta^{s-1})$. To guarantee that the chain explores the entire posterior, the kernel needs to be both measure-preserving and ergodic.

Definition 1 (Measure-Preserving Transition Kernel)**.** *A transition kernel $T : \Omega \to \Omega$ defined on a probability space $(\Omega, \mathcal{F}, P)$ is measure preserving if for all $A \in \mathcal{F}$*

$$P(T^{-1}(A)) = P(A)$$

This condition is needed to guarantee that the samples are drawn from the right density.

Definition 2 (Ergodic Transition Kernel)**.** *A transition kernel $T : \Omega \to \Omega$ defined on a probability space $(\Omega, \mathcal{F}, P)$ is ergodic if for every $A \in \mathcal{F}$ with $T^{-1}(A) = A$, either $P(A) = 0$ or $P(A) = 1$*

The ergodic property of the transition kernel is a necessary property to guarantee that the chain fully explores the parameters space. If the transition kernel is not ergodic and $T^{-1}(A) = A$ for some $A \in \mathcal{F}$. This means that all values mapping to A are in A. If transition kernel T also maps to another region $B = \Omega/A \subset T(A)$, this means that once the chain is in B it never returns to A. The chain neglects A even though it has positive probability. If the chain is ergodic it does not get stuck on a subset of the parameter space.

Definition 3. *A function $h : \Omega \to \mathbb{R}$ is called Lebesgue integrable, denoted by $h \in L^1(\Omega, \mathcal{F}, P)$, if:*

$$\int h(\theta)p(\theta|y)d\theta < \infty$$

When a function $h(\theta)$ is not Lebesgue integrable,the Monte Carlo methods will not find the correct solution to the integration. This has to due with the fact that the samples drawn from the posterior are always finite. So the Monte Carlo solution will give a finite solution even though the real solution is infinite.

Theorem 3 (Birkhoff (1931) Ergodic Theorem)**.** *Let $T : \Omega \to \Omega$ be a ergodic measure-preserving transformation kernel and let $h \in L^1(\Omega, \mathcal{F}, P)$ be a lebesgue integrable function then:*

$$\lim_{S\to\infty} \frac{1}{S} \sum_{s=0}^{S-1} h(\theta^s) \to \int h(\theta) p(\theta|y) d\mu(\theta)$$

Theorem 3 combines the previous definition and show that they are a sufficient condition to find a integral via a Markov Chain..

A.1 Random Walk Metropolis Hastings

One of the earliest and most famous Markov Chain Monte Carlo methods is the Metropolis-Hastings algorithm and is named after the writer of two papers namely Metropolis et al. (1953) and Hastings (1970). The algorithm has the following steps.

The algorithm starts with a starting value θ, from this starting value a proposal θ' is drawn. The proposal value comes form a normal distribution around the original value:

$$\theta' \sim \text{Normal}(\theta^{s-1}, \sigma_0)$$

And let the $q(\theta', \theta^{s-1})$ denote the probability density function of the proposal distribution.

If every proposal value is used as a new input in the chain,the chain will be a random walk and therefore will not reproduce the distribution of interest. So a method is needed to "force" the chain to stay in the right region. This is done by accepting only certain proposals.

Whether the new proposal is accepted depends on the proportion of the probability density of the proposed value θ' and the starting value θ. If the probability of the proposed value is higher than the value of the the starting value, then the proposed value is accepted. The rationale of this process is that the Metropolis-Hastings Algorithm always accepts a proposed value that is has a higher density than the current value. So it has a preference to explore spaces with high densities. When a new value is proposed with lower density there is a probability $p(\theta'|y)/p(\theta^s|y)$ that the new value will be accepted. This means that the chain also can explore spaces with low probability density. The acceptance rate for the new point is:

$$\begin{aligned}
\alpha(\theta', \theta^s) &= \min\left(1, \frac{p(\theta'|y)}{p(\theta^s|y)}\right) \\
&= \min\left(1, \frac{p(y|\theta')p(\theta')}{p(y)} \Big/ \frac{p(y|\theta^s)p(\theta^s)}{p(y)}\right) \\
&= \min\left(1, \frac{p(y|\theta')p(\theta')}{p(y|\theta^s)p(\theta^s)}\right) \qquad (12)
\end{aligned}$$

So to calculate the acceptance rate for the normalised posteriors only the unnormalised posteriors are needed. This is how the Metropolis-Hasting algorithm gets around calculating the normalising constant $p(y)$. This combined with the proposal step gives algorithm 2.

Algorithm 2: Random Walk Metropolis Hastings

$\theta^0 \leftarrow$ random starting point
for $s \in \{1, ..., S\}$ **do**
  $\theta' \leftarrow N(\theta^{s-1}, \sigma_0)$
  $\alpha \leftarrow \min\left\{1, \frac{p(\theta')}{p(\theta^{s-1})}\right\}$
  $u \sim U[0, 1]$
  if $\alpha > u$ **then**
    $\theta^s \leftarrow \theta'$
  else
    $\theta^s \leftarrow \theta^{s-1}$
  end
end

Example A.1: Metropolis Hastings of logistic regression

Fit a logistic model with data consisting of a single value of $y = 1$ and one $x = 2$.

$$\log\left(\frac{\theta}{1-\theta}\right) = \beta_0 + \beta x$$

Where there is a standard normal prior on β_0 and β, such that the probability density for a single value:

$$p(\beta^*) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\beta^*}{2}\right)$$

The likelihood of the point y being 1 given β_0^*, β^* and x is:

$$p(y = 1|\beta_0^*, \beta^*, x) = \frac{1}{1 + \exp(-(\beta_0^* + \beta^* x))}$$

Combining the likelihood and the prior gives the unnormalised posterior:

$$p(\beta_0^*, \beta^*|y) \propto \frac{1}{1 + \exp(-(\beta_0^* + \beta^* x))} \exp\left(-\frac{\beta_0^*}{2}\right) \exp\left(-\frac{\beta^*}{2}\right)$$

To estimate the model draw a random starting proposal point β_0' and β' with a the mean being equal to the previous point β_0^{s-1} and β^{s-1}. And using the result from equation 12 and the likelihood function given before:

$$\alpha(\{\beta_0', \beta'\}, \{\beta_0^s, \beta^s\}) = \min\left(1, \frac{1 + \exp(-(\beta_0^s + \beta^s x)}{1 + \exp(-(\beta_0' + \beta' x)} \frac{\exp((\beta_0')^2/2)}{\exp((\beta_0^s)^2/2)} \frac{\exp((\beta')^2/2)}{\exp((\beta^s)^2/2)}\right)$$

Now draw a random sample from a uniform distribution and if the value of α is higher then accept the value as the new input in the Markov Chain.

The Metropolis Hastings algorithm is run for this problem and the first ten values of the Markov Chain are shown in figure 43 and in table A.1.

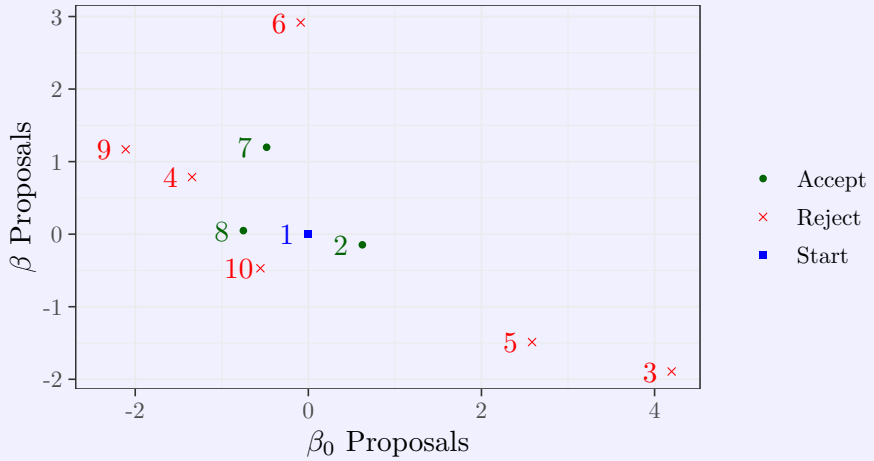

Figure 43: Ten iterations of the Metropolis Hastings algorithm, each dot and cross is a proposal which is either accepted or rejected.

	β_0^s	β^s	β_0'	β'	α	Accept/Reject
1	0.00	0.00	-	-	-	Start
2	0.62	-0.15	0.62	-0.15	0.95	Accept
3	0.62	-0.15	4.20	-1.89	0.00	Reject
4	0.62	-0.15	-1.34	0.79	0.35	Reject
5	0.62	-0.15	2.58	-1.49	0.01	Reject
6	0.62	-0.15	-0.09	2.92	0.03	Reject
7	-0.48	1.20	-0.48	1.20	0.80	Accept
8	-0.75	0.05	-0.75	0.05	0.68	Accept
9	-0.75	0.05	-2.11	1.17	0.12	Reject
10	-0.75	0.05	-0.55	-0.47	0.55	Reject

Table 55: Ten iterations of the Metropolis Hastings algorithm corresponding to figure 43

Figure 44 shows 1,000 samples from the posterior distribution. Due to the high auto-correlation, the Monte Carlo samples are clustered. Many proposed values are rejected, this means that a single point in the plot can represent multiple Monte Carlo samples.

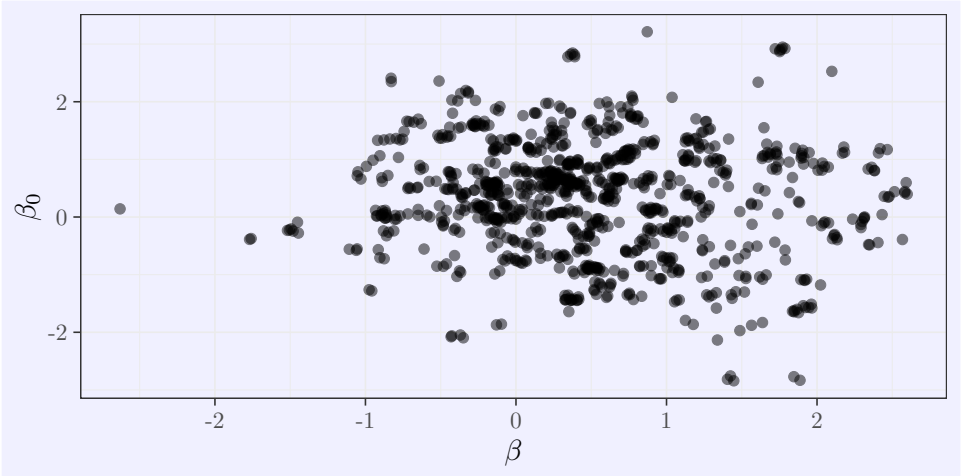

Figure 44: Thousand samples of the Metropolis Hastings algorithm

The algorithm has to be measure preserving to converge to the solution. A way to show that a Markov Chain is measure-preserving is by showing that the kernel satisfies detailed balance.

Definition 4 (Detailed Balance)**.** *A transition kernel T satisfies detailed balance if:*

$$T(\theta^s|\theta^{s-1})p(\theta^{s-1}) = T(\theta^{s-1}|\theta^s)p(\theta^s)$$

Because the proposal is normally distributed and therefore symmetric $q(\theta^s, \theta^{s-1}) = q(\theta^{s-1}, \theta^s)$. Now it can be shown that the Metropolis-Hastings kernel satisfies detailed balance.

$$\begin{aligned}
p(\theta^{s-1}|y)T(\theta^s|\theta^{s-1}) =& \frac{p(y|\theta^{s-1})p(\theta^{s-1})}{p(y)} \min\left(1, \frac{p(y|\theta^s)p(\theta^s)}{p(y|\theta^{s-1})p(\theta^{s-1})}\right) q(\theta^s|\theta^{s-1}) \\
=& \frac{1}{p(y)} \min\left(p(y|\theta^{s-1})p(\theta^{s-1}), p(y|\theta^s)p(\theta^s)\right) q(\theta^s|\theta^{s-1}) \\
=& \frac{p(y|\theta^s)p(\theta^s)}{p(y)} \min\left(1, \frac{p(y|\theta^s)p(\theta^s)}{p(y|\theta^{s-1})p(\theta^{s-1})}\right) q(\theta^{s-1}|\theta^s) \\
=& p(\theta^s|y)T(\theta^{s-1}|\theta^s)
\end{aligned}$$

Lemma 1. *A transition kernel that satisfies detailed balance is also measure preserving.*

To show that detailed balance implies that the transition kernel is measure preserving let $A \in F$ and let $B_i \in F$ be disjoint sets, such that $A = (\Omega / \cup_i B_i)$

$$\begin{aligned}
P(T^{-1}(A)) =& T(A|A)P(A) + \sum_i T(A|B_i)P(B_i) && \text{(all sets mapping to } A\text{)} \\
=& T(A|A)P(A) + \sum_i T(B_i|A)P(A) && \text{(Detailed balance)} \\
=& P(A)\left(T(A|A) + \sum_i T(B_i|A)\right) && (T(\Omega|A) = 1) \\
=& P(A)
\end{aligned}$$

Jarner & Hansen (2000) show that the posterior has to have at least an exponential heavy tail, the same a Laplace distribution, to guarantee the Metropolis-Hasting algorithm to be ergodic.

Ergodicity..korte toelichting over voorwaarden voor ergodiciteit

In the case that the Metropolis-Hastings algorithm is both ergodic and measure-preserving then Birkhoff's Ergodic Theory holds and the Monte Carlo solution converges to the real solution. However, the theory does not state how quick the solution converges. A metric for convergence speed is the Effective Sample Size (ESS) of a Monte Carlo method. The MCMC samples are by construction dependent on the previous sample, this often means that the samples are correlated with each other.

$$\text{ESS} = \frac{S}{1 + 2\sum_{l=1}^{\infty} \rho_l}$$

Where S is the number of samples and ρ is the l-lag autoregression coefficient of the Markov Chain. The Effective Sample Size is a measure to estimate how many independent samples from the distribution would give an equal error as the dependent Monte Carlo samples.

In high dimensions the Metropolis-Hastings algorithm has the problem that if the random step is chosen to big, then the proposed step often jumps out of the region of high probability. This causes that most of the proposed value are rejected and this means slow convergence to the distribution. On the other hand, when small random steps are used, most of the proposed values are accepted, however the convergence is still slow due to the fact that it slowly explores the space. In both cases the autogregressive coefficient ρ_l becomes large.

To show the problem of Metropolis Hastings in high dimension draw I draw samples from a multivariate normal with increasing dimension. Gelman et al. (1996) show that for this problem the efficient $\sigma_0 \approx 2.4d^{-1/2}$ and $ESS \approx \frac{0.3S}{d}$, the optimal acceptance rate is starts at 44% with $D = 1$ and the optimal acceptance rate goes to 23% as $D \to \infty$. In figure 45 the Effective Sample size divided by the total Monte Carlo samples is plotted against the dimensions of the multivariate normal. The Effective Sample Size deteriorates as the quickly as the dimensions increase.

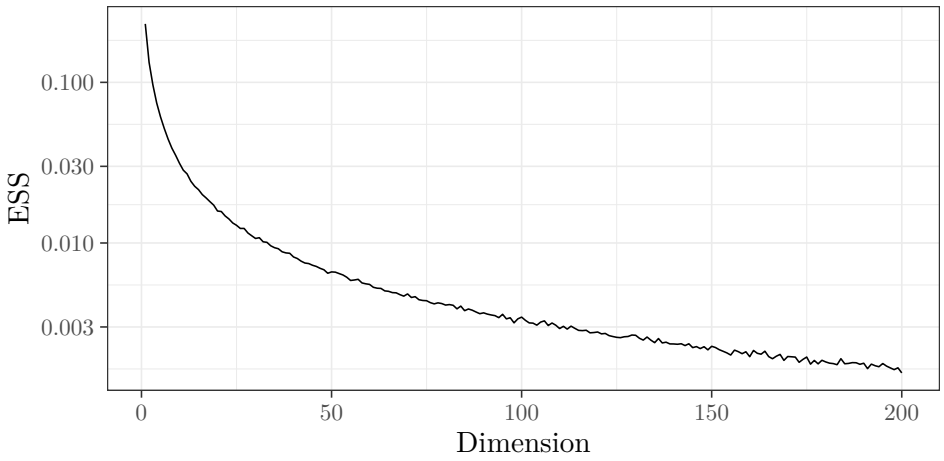

Figure 45: Metropolis Hastings simulation from a d-dimensional independent standard normal distribution.

A.2 Hamiltonian Monte Carlo

Because the Metropolis Hastings algorithm has slow convergence for high dimensional parameters spaces, another algorithm needs to be found to solve the high dimensional problems. Instead of using random steps to propose new values, the geometry of the distributions can be used to propose new values. One method that uses this information is a class of algorithms called Hamiltonian Monte Carlo algorithms.

The basic idea of Hamiltonian Monte Carlo is to double the parameters space from D-dimensional (θ) to $2D$-dimensional (θ, ζ). The doubled parameter space has certain properties, which makes it possible to make proposals with higher acceptances rates than the Metropolis Hastings algorithm. In Hamiltonian Monte Carlo, samples are drawn from the double parameter space. From this double parameter space the original parameter space easily can be recovered by marginalising out the auxiliary parameters ζ, such that:

$$\int p(\theta, \zeta) d\zeta = p(\theta) \int p(\zeta) d\zeta = p(\theta) \cdot 1 = p(\theta)$$

In practise this integration is not done, and the marginalisation is done by dropping out the ζ parameters. Which is equivalent to the integral.

A.2.1 Hamiltonian dynamics

Hamiltonian Monte Carlo does not use random steps to make new proposals, but makes use of the geometry of the unnormalised posterior. This is done via the Hamiltonian, which is defined as (Neal et al., 2011):

$$\begin{aligned} H(\theta, \zeta) &:= -\log(p(\theta, \zeta)|y) \\ &= -\log(p(\zeta|\theta)p(\theta|y)) \\ &= -\log(p(\zeta|\theta)) - \log(p(\theta|y)) \end{aligned}$$

Furthermore kinetic energy is defined as:

$$K(\theta, \zeta) = -\log p(\zeta|\theta)$$

And potential energy as:

$$V(\theta) = -\log p(\theta|y) = -\sum_i \log p(y_i|\theta) - \log p(\theta) + \log(p(y))$$

Because Hamiltonian originates from Hamiltonian dynamics, jargon from dynamics is used. That is why the auxiliary parameter ζ is called the momentum.

The Hamiltonian equations can be used to make generate a proposal value. This is done by making a path $\phi(\theta, \zeta)$ using the Hamilton's equations:

$$\frac{d\theta_i}{dt} = \frac{\partial H}{\partial \zeta_i} = \frac{\partial K}{\partial \zeta}$$
$$\frac{d\zeta_i}{dt} = -\frac{\partial H}{\partial \theta_i} = -\frac{\partial K}{\partial \theta_i} - \frac{\partial V}{\partial \theta_i}$$

This system has the property that it preserves the Hamiltonian analytically, so the Hamiltonian does not depend on time, this can easily be shown by:

$$\frac{dH}{dt} = \sum_i^D \frac{\partial H}{\partial \theta_i}\frac{d\theta_i}{dt} + \frac{\partial H}{\partial \zeta_i}\frac{d\zeta_i}{dt} = \sum_i^D \frac{\partial H}{\partial \theta_i}\frac{\partial H}{\partial \zeta_i} - \frac{\partial H}{\partial \theta_i}\frac{\partial H}{\partial \zeta_i} = 0$$

A way to look at this problem is that the path $\phi_t(\theta, \zeta)$ is a path with constant Hamiltonian. As the Hamiltonian is directly related to the probability in the doubled space, the analytic path follows the lines of equal probability. In the analytical case the new proposal is always accepted, however the the paths can only be calculated analytically in certain cases. In the general case a numerical integrator needs to be used.

A.2.2 Leapfrog integration

One class of integrators that is useful for this system are the symplectic integrators. These integrators have the property that they conserve the Hamiltonian much better than other methods, like Euler's method or higher order variants of this method. Where other classes quickly drift away from the original value of the Hamiltonian, these integrators stay close the Hamiltonian level, even after long times of integration.

The leapfrog integrator is one of these integrators and has the following procedure:

Algorithm 3: Single Leapfrog Integration

$(\theta_0, \zeta_0) \leftarrow (\theta^s, \zeta)$
$\zeta_{\frac{1}{2}} = \zeta_0 - \frac{\epsilon}{2}\frac{\partial V}{\partial \theta}(\theta_0)$
for $i \in \{1, ..., t\}$ **do**
  $\theta_i = \theta_{i-1} + \epsilon\zeta_{i+\frac{1}{2}}$
  $\zeta_{i+\frac{1}{2}} = \zeta_{i-\frac{1}{2}} - \epsilon\frac{\partial V}{\partial \theta}(\theta_{i-\frac{1}{2}})$
end
$\zeta_t = \zeta_{t-\frac{1}{2}} - \frac{\epsilon}{2}\frac{\partial V}{\partial \theta}(\theta_i)$
$(\theta', \zeta') \leftarrow (\theta_t, \zeta_t)$

The leapfrog integrator is time reversible (Leimkuhler et al., 1996). When starting from point (θ^s, ζ^s) and going to point (θ', ζ'). By changing the momentum parameters ζ' to $-\zeta'$, the path can exactly return to the start point (θ^s, ζ^s). This is not generally the case for numerical integrator, but it is an necessary condition to guarantee that the Hamiltonian Monte Carlo is measure preserving.

The last property of importance is that the leapfrog integrator, like other symplectic integrators, preserve volume (Channell & Scovel, 1990).

A.2.3 Accept Reject step

Due to the numerical approximation of the path, the Hamiltonian of the proposal values (θ', ζ') are not exactly the same as the Hamiltonian of the starting point. Just like in the Metropolis-Hastings algorithm a accept/reject step is needed to guarantee that the transition kernel is measure preserving.

$$\begin{aligned}\alpha(\theta', \zeta', \theta^s, \zeta) &= \min\left(1, \frac{p(\theta', \zeta'|y)}{p(\theta^s, \zeta|y)}\right) \\ &= \min\left(1, \frac{p(y|\theta')p(\theta')}{p(y)} \Big/ \frac{p(y|\theta^s)p(\theta^s)}{p(y)}\right) \\ &= \min\left(1, \frac{p(y|\theta')p(\theta')}{p(y|\theta^s)p(\theta^s)}\right) \\ &= \min\left(1, \exp(-H'(\theta', \zeta') + H'(\theta^s, \zeta))\right)\end{aligned}$$

Where H' is the unnormalised Hamiltonian, so without the constants that do not depend on θ and ζ.

A.2.4 Defining the Kinetic Energy

Euclidean-Gaussian Hamiltonian Monte Carlo defines the kinetic energy as:

$$p(\zeta|\theta) \sim N(0, M)$$

Such that:

$$K = \frac{1}{2}\zeta^T M^{-1}\zeta + \log|M| + \text{const.}$$

The kinetic energy determines the how big the jumps are of the momentum. Putting all previous steps together gives the following algorithm.

Algorithm 4: Hamilotian Monte Carlo

$\theta^0 \leftarrow$ random point
for $s \in \{1, ..., S\}$ **do**
  $\zeta \sim N(0, M)$
  $(\theta', \zeta') = \phi_t(\theta^{s-1}, \zeta)$
  $\alpha \leftarrow \min\left\{1, \exp\left(-H(\theta', \zeta') + H(\theta^{s-1}, \zeta)\right)\right\}$
  $u \sim U[0, 1]$
  if $\alpha > u$ **then**
    $\theta^s \leftarrow \theta'$
  else
    $\theta^s \leftarrow \theta^{s-1}$
  end
end

Example A.2: Hamiltonian Monte Carlo of logistic regression

Let the problem be the same as in example A.1. So define a logistic model with one data point with $y = 1$ and $x = 2$

$$\log\left(\frac{\theta}{1-\theta}\right) = \beta_0 + \beta x$$

Where there is a standard normal prior on β_0 and β, such that the probability density for a single value:

$$p(\beta^*) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\beta^*}{2}\right)$$

The potential energy in this problem is:

$$V(\beta_0, \beta, y) = \log(1 + e^{-\beta_0 - 2\beta}) + \frac{\beta_0^2}{2} + \frac{\beta^2}{2} + \text{const.}$$

When the momenta are drawn for a normal distribution $\zeta, \zeta_0 \sim N(0, m)$ then the kinetic energy is:

$$K(\zeta_0, \zeta) = \frac{\zeta_0^2}{2m_0} + \frac{\zeta^2}{2m} + \text{const.}$$

To calculate the leapfrog integration the derivative of the potential energy:

$$\begin{aligned}\frac{\partial V}{\partial \beta_0} =& \frac{-1}{1 + \exp(\beta_0 + 2\beta)} + \beta_0 \\ \frac{\partial V}{\partial \beta} =& \frac{-2}{1 + \exp(\beta_0 + 2\beta)} + \beta\end{aligned}$$

By arbitrarily setting the starting point to $(\beta_0, \beta) = (1, 1)$ and running the algorithm, the Markov Chain explores the probability space as depicted in figure 46.

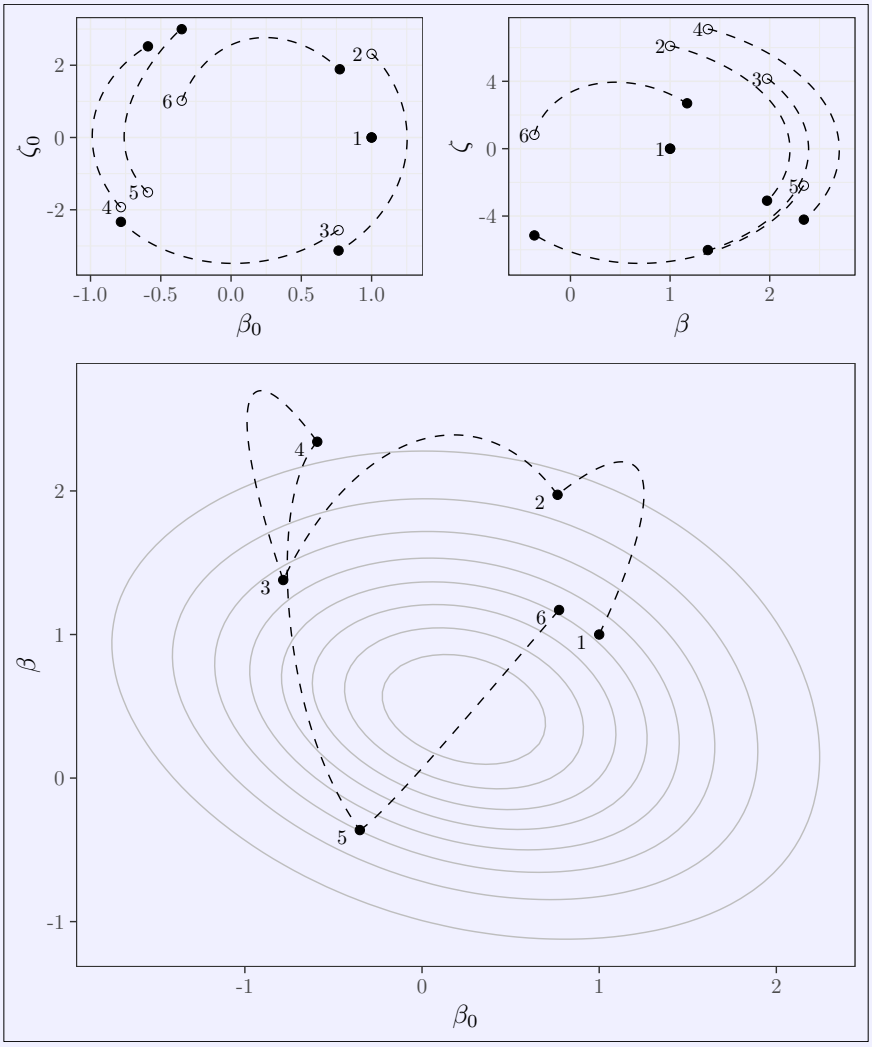

Figure 46: The first six points of the Hamiltonian Monte Carlo simulation with paths between samples.

The initial point is point 1, from this point two random momenta are drawn such that

momentum jumps to hollow point number two (in upper two plots). By using the leapfrog integrator, the a path is drawn for the original hollow point to the black proposal point. Because the Hamiltonian is nearly conserved all points are accepted in this case. Now new momenta are drawn and the points and and the chain jumps from the black 2 to the hollow 3 and the process is repeated. The lower plot is the plot of interest, which depicts the resulting paths and point for the β_0, β plane.

Figure 47 shows 1.000 draws from the posterior distribution. Compared to the Metropolis Hasting algorithm in figure 44, the Hamiltonian Monte Carlo method produces are more even distribution of points through the posterior distribution.

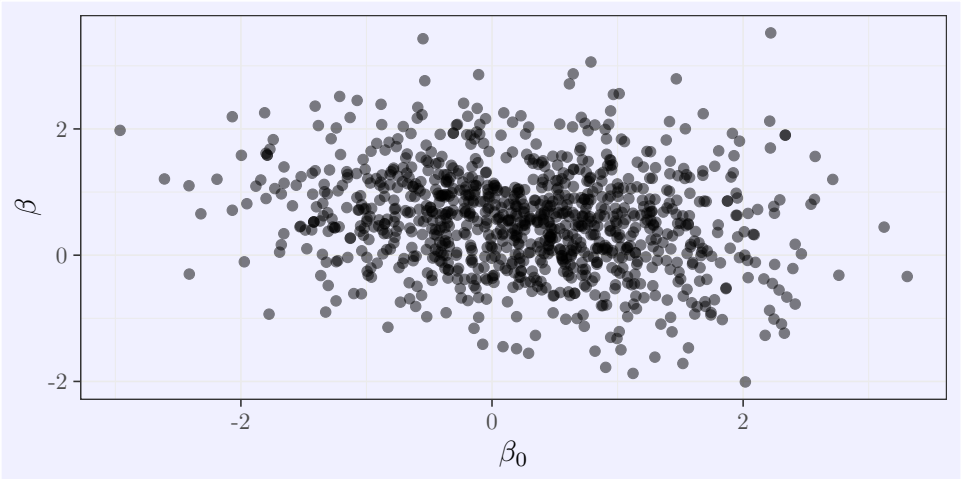

Figure 47: thousand samples from the distribution using Hamiltonian Monte Carlo

A.2.5 Ergodicity

Livingstone et al. (2016) prove that Hamiltonian Monte Carlo does not produce ergodic chains under certain tail geometric conditions of the posterior distribution.

1. $\lim_{||\theta||\to\infty} \frac{||\nabla_\theta V(\theta|y)||}{||\theta||} = \infty$
2. There is an $M < \infty$ such that $\nabla_\theta V(\theta|y) \leq M$ for all θ, and $\mathbb{E}[e^{t||\theta||}] = \infty$ for every $t > 0$

Where $||.||$ is the Euclidean norm.

When applying these conditions to the exponential distribution family $p(\theta) \propto \exp(-||\theta||^\gamma)$, the first conditions implies that the tails of the distribution should be heavier than the tails of a normal distribution ($\gamma = 2$).

In case that the first condition does not hold the gradient $\nabla V(\theta|y)$ becomes very large and it becomes hard for the numerical integrator to follow the analytic path. In this case the leapfrog integrator has the tendency to quickly diverge to high energy levels. This behaviour makes it easy to identify such divergent iteration. In the Stan, the package I use for implementing Hamiltonian Monte Carlo, these divergent iteration are automatically identified.

The second condition corresponds with the tails that are heavier than the tails of the Laplace($\gamma = 1$) distribution. In this case the gradient $\nabla V(\theta|y)$ has a low value, the path of the leapfrog is marginally influenced by the information in the gradient. This causes the path to turn into a random walk, which deteriorates the performance of HMC.

Because the shape of the posterior is unknown, these diagnostics are useful tools. In case that one of these diagnostics points out a problem, the result may not be reliable. Metropolis-Hastings does not have these kind of diagnostic and non-ergodicity may not be detected, leading to misleading results.

A.2.6 No-U-Turn sampler

Still it remains unclear what the integration time the leapfrog integrator should be. When only a few steps are taking the path might not have explored the space enough, which leads to highly correlated Monte Carlo samples. However when the integration time is too long, then the path might up where it began. Even if this does not happen, the calculation time is high, due to the calculation of the paths. Hoffman & Gelman (2014) suggest the No-U-Turn sampler (NUTS) to solve this problem. This method dynamically chooses the integration time of the leap frog integrator. The idea of the solver is that it adds leapfrog steps until the path makes a U-turn. When this happens the No-U-turn sampler picks one of the leapfrog points and uses this as the proposed value.

A.2.7 Stan

For Bayesian generalised linear models I use *rstanarm*. This package in turn opens the *Rstan*, the R version of Stan. *Rstan* is package that uses the NUTS (Carpenter et al., 2017) to sample from the posterior. The advantage of *Rstan* is that before sampling it optimises the kinetic energy and the leap-frog step size in its warm-up/burn-in period (Betancourt, 2017). Furthermore, it automatically calculates the derivatives needed in the NUTS sampler. This saves a lot of time and effort in fitting the model. It is possible to use Stan to define your own model, which gives a lot of flexibility. The code is compiled to C++ for fast calculations. I do not write my own models, but I use a generalised linear model function in *rstanarm*. This makes implementation easy and fast, for common statistical models.

A.2.8 Diagnostics

Hamiltonian Monte Carlo requires the posterior distribution to be differentiable, when the posterior is not differentiable Hamiltonian Monte Carlo cannot be used. Furthermore even if the posterior is differentiable, but the analytic paths have high curvature, the leapfrog integrator might not be able to follow the path. When this happens the leapfrog integrator has the tendency to quickly diverge to a high Hamiltonian value. Due to this extreme behaviour divergent iterations are easily identifiable. Stan has a build in detection for divergence, when divergent paths are present the results are unreliable and reparametrisation of the model might be needed.

B Information Theory

A wide class of measure to identify the performance of a model are rooted in information theory .

As a measure of information Shannon's entropy is used and is defined as (MacKay & Mac Kay, 2003):

$$\mathcal{H}(p) = -\int p(y)\log(p(y))dy$$

Entropy is a measure of the average information that is given by an outcome. So the more uncertain the system, the higher the value of the information that the outcomes gives and the higher the entropy.

Example B.1: Entropy

When taking a probability of default model where the default is modelled as a Bernoulli distribution, such that:

$$Y \sim \text{Bernoulli}(\theta)$$

If $\theta_1 = 0.5$ then the entropy of the distribution is:

$$\mathcal{H} = -\left(0.5\log_2(0.5) + 0.5\log_2(0.5)\right) = 1$$

And when $\theta_2 = 0.02$:

$$\mathcal{H} = -\left(0.02\log_2(0.02) + 0.98\log_2(0.98)\right) \approx 0.086$$

In the case of a 1 percent default probability there is much less uncertainty than in the of 50 percent default. In the case of a 1 percent default guessing that the model will not default is right most of the time. In the case of 50 percent default, this is right only half of the time.

Khinchin (2013) shows that entropy is the only measure of uncertainty that satisfies the following properties:

- The uniform distribution has the highest entropy.
- Independent events have additive uncertainty.
- Adding an outcome with zero probability has no effect on the uncertainty.
- Uncertainty is continuous in its arguments.

B.1 Cross Entropy

The cross entropy is the amount of information under the wrong assumption of the distribution. However this can also be seen as the log loss of the fitted model p_m model assumptions.

$$\mathcal{H}(p_t, p_m) = -\int p_t(y)\log(p_m(y))dy$$

The expected log predictive density (elpd) is equal to minus the cross entropy.

B.2 Kullback Leibler Divergence

The Kullback-Leibler divergence between the real data generating distribution p_t and the estimated distribution p_m is:

$$\begin{aligned}\mathrm{KL}(p_t, p_m) = &\int p_t(y) \log \frac{p_t(y)}{p_m(y)} dy \\ =&\mathcal{H}(p_t, p_m) - \mathcal{H}(p_t)\end{aligned}$$

The Kullback Leibler is the difference of the cross entropy and the entropy of the real data generating model. So this is the loss of information due to the fact that wrong model used instead of the real model.

In real life cases the data generating distribution p_t is unknown, therefore the Kullback Leibler divergence between the the model and the data generating process cannot be calculated. The next best thing is to use the measure the

$$\mathrm{KL}(p_t, p_{m1}) - \mathrm{KL}(p_t, p_{m2}) = \mathcal{H}(p_t, p_{m1}) - \mathcal{H}(p_t, p_{m2})$$

This equation holds because the entropy of the true model is constant over all fitted models.

So two models can be compared by using the cross entropy. Information criteria and cross validation techniques try to approximate the cross entropy of the fitted model.

C Normal Scale-Mixtures

As discussed in Section 5.2.3 various distributions can be written as a normal scale-mixture. Where the variance of a normal distribution is a random variable as well, such that:

$$\begin{aligned} \beta|\lambda &\sim N(0, \lambda) \\ \lambda &\sim p(\lambda) \end{aligned} \tag{13}$$

Where the prior is given by:

$$p(\beta) = \int p(\beta|\lambda)p(\lambda)d\lambda$$

The prior on λ gives an implicit prior on the shrinkage weight κ_i. Which is defined as:

$$\kappa_i = \frac{1}{1+\lambda^2}$$

C.1 Laplace distribution

The Laplace prior can be written as normal scale-mixture model, where the scale mixing is done with a exponential distribution:

$$\lambda^2 \sim Exp\left(\frac{1}{2b^2}\right)$$

Which has the following probability density distribution:

$$p(\lambda^2) = \frac{1}{2b}\exp\left(\frac{\lambda^2}{2b^2}\right)$$

The Laplace distribution on β is:

$$p(\beta) \propto \frac{1}{2b}\exp\left(-\frac{|\beta|}{b}\right)$$

C.1.1 Shrinkage profile Laplace

$$\begin{aligned} \lambda^2 &= \frac{1}{\kappa} - 1 \\ \frac{d\lambda^2}{d\kappa} &= -\frac{1}{\kappa^2} \end{aligned}$$

The implied shrinkage prior is a result of a change of measure from λ^2 to κ.

$$\begin{aligned} p(\kappa) &= p(\lambda^2)\frac{d\lambda^2}{d\kappa} \\ p(\kappa) &\propto \exp\left(-\frac{\lambda^2}{2b^2}\right)\frac{d\lambda}{d\kappa} \\ p(\kappa) &\propto \exp\left(-\frac{1}{2\kappa b^2} + \frac{1}{2\kappa b^2}\right)\frac{d\lambda}{d\kappa} \\ p(\kappa) &\propto \frac{1}{\kappa^2}\exp\left(-\frac{1}{2b\kappa}\right) \end{aligned}$$

C.2 Horseshoe prior

The Horseshoe prior is a normal scale-mixture with a half Cauchy distribution:

$$p(\lambda) \propto \frac{1}{1+\lambda^2}$$

The shrinkage parameters for the Horseshoe prior is defined as:

$$\kappa = \frac{1}{1+\lambda^2\tau^2}$$

Rewriting the expressions gives:

$$\frac{1}{1+\lambda^2} = \frac{\kappa\tau^2}{\kappa\tau^2+1-\kappa}$$

C.2.1 Shrinkage profile Horseshoe

The scale λ expressed in term of κ is:

$$\lambda = \sqrt{\frac{1}{\tau^2}\left(\frac{1}{\kappa}-1\right)} = \frac{1}{\tau}\sqrt{\frac{1}{\kappa}-1}$$

The derivative from λ to κ is:

$$\frac{d\lambda}{d\kappa} = \frac{1}{\tau}\sqrt{\frac{\kappa}{1-\kappa}}\cdot\frac{1}{\kappa^2} = \tau^{-1}\kappa^{-1.5}\left(1-\kappa\right)^{-0.5}$$

And the shrinkage prior of the Horseshoe prior is:

$$\begin{aligned}
p(\kappa|\tau) &= p(\lambda)\frac{d\lambda}{d\kappa} \\
p(\kappa|\tau) &\propto \frac{1}{1+\lambda^2}\frac{d\lambda}{d\kappa} \\
p(\kappa|\tau) &\propto \frac{\kappa\tau^2}{\kappa\tau^2+1-\kappa}\frac{d\lambda}{d\kappa} \\
p(\kappa|\tau) &\propto \frac{\kappa\tau^2}{\kappa\tau^2+1-\kappa}\cdot\tau^{-1}\kappa^{-1.5}\left(1-\kappa\right)^{0.5} \\
p(\kappa|\tau) &\propto \frac{\tau}{\kappa\tau^2+(1-\kappa)}\kappa^{-\frac{1}{2}}(1-\kappa)^{-\frac{1}{2}} \\
p(\kappa|\tau) &\propto \frac{\tau}{(\tau^2-1)\kappa+1}\kappa^{-\frac{1}{2}}(1-\kappa)^{-\frac{1}{2}}
\end{aligned}$$

For $\tau = 1$, the implied prior is a Beta($\frac{1}{2}, \frac{1}{2}$).

D Table of distributions

Distribution	Symbol	Probability density/mass function	Support
Normal	$N(\mu, \sigma)$	$p(x\|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{\sigma^2}\right)$	$x \in \mathbb{R}$
Cauchy	$C(x_0, \gamma)$	$p(x\|x_0,\gamma) = \frac{1}{\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]}$	$x \in \mathbb{R}$
Half Cauchy	$C^+(x_0, \gamma)$	$p(x\|x_0,\gamma) = \frac{2}{\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]}$	$x \in \mathbb{R}^+$
Generalised Pareto	$GPD(k, \sigma, u)$	$p(x\|k,\sigma,u) = \frac{1}{\sigma}\left(1 + k\left(\frac{x-u}{\sigma}\right)\right)^{-\frac{1}{k}-1}$	$x \in [u, \infty)$

References

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike* (pp. 199–213). Springer.

Basel Committee on Banking Supervision. (2017). *Basel iii: Finalising post-crisis reforms.* Bank for International Settlements Basel.

Basel committee membership. (2013, Jun). Retrieved from `https://www.bis.org/bcbs/membership.htm?m=3|14|573|71`

Berger, J. O., Bernardo, J. M., Sun, D. et al. (2009). The formal definition of reference priors. *The Annals of Statistics*, *37*(2), 905–938.

Betancourt, M. (2017). A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434* .

Bierkens, J., Bouchard-Côté, A., Doucet, A., Duncan, A. B., Fearnhead, P., Lienart, T., ... Vollmer, S. J. (2018). Piecewise deterministic markov processes for scalable monte carlo on restricted domains. *Statistics & Probability Letters*, *136*, 148–154.

Birkhoff, G. D. (1931). Proof of the ergodic theorem. *Proceedings of the National Academy of Sciences*, *17*(12), 656–660.

Black, F. & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of political economy*, *81*(3), 637–654.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., ... Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software, Articles*, *76*(1).

Carvalho, C. M., Polson, N. G. & Scott, J. G. (2009). Handling sparsity via the horseshoe. In *Artificial intelligence and statistics* (pp. 73–80).

Carvalho, C. M., Polson, N. G. & Scott, J. G. (2010). The horseshoe estimator for sparse signals. *Biometrika*, *97*(2), 465–480.

Cawley, G. C. & Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, *11*(Jul), 2079–2107.

Channell, P. J. & Scovel, C. (1990). Symplectic integration of hamiltonian systems. *Nonlinearity*, *3*(2), 231.

Datta, J., Ghosh, J. K. et al. (2013). Asymptotic properties of bayes risk for the horseshoe prior. *Bayesian Analysis*, *8*(1), 111–132.

Dupuis, J. A. & Robert, C. P. (2003). Variable selection in qualitative models via an entropic explanatory power. *Journal of Statistical Planning and Inference*, *111*(1-2), 77–94.

Freddie Mac. (2019). *Single family loan-level dataset general user guide.* Retrieved from `http://www.freddiemac.com/fmac-resources/research/pdf/user_guide.pdf`

Gelman, A., Jakulin, A., Pittau, M. G., Su, Y.-S. et al. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, *2*(4), 1360–1383.

Gelman, A., Roberts, G. O., Gilks, W. R. et al. (1996). Efficient metropolis jumping rules. *Bayesian statistics*, *5*(599-608), 42.

Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A. & Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall CRC.

Ghosh, J., Li, Y., Mitra, R. et al. (2018). On the use of cauchy prior distributions for bayesian logistic regression. *Bayesian Analysis*, *13*(2), 359–383.

Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications.

Hoffman, M. D. & Gelman, A. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, *15*(1), 1593–1623.

Jarner, S. F. & Hansen, E. (2000). Geometric ergodicity of metropolis algorithms. *Stochastic processes and their applications*, *85*(2), 341–361.

Khinchin, A. Y. (2013). *Mathematical foundations of information theory*. Courier Corporation.

Leimkuhler, B. J., Reich, S. & Skeel, R. D. (1996). Integration methods for molecular dynamics. In *Mathematical approaches to biomolecular structure and dynamics* (pp. 161–185). Springer.

Livingstone, S., Betancourt, M., Byrne, S. & Girolami, M. (2016). On the geometric ergodicity of hamiltonian monte carlo. *arXiv preprint arXiv:1601.08057*.

MacKay, D. J. & Mac Kay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

McElreath, R. (2018). *Statistical rethinking: A bayesian course with examples in r and stan*. Chapman and Hall/CRC.

Merton, R. C. (1974). On the pricing of corporate debt: The risk structure of interest rates. *The Journal of finance*, *29*(2), 449–470.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, *21*(6), 1087–1092.

Mitchell, T. J. & Beauchamp, J. J. (1988). Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, *83*(404), 1023–1032.

Neal, R. M. et al. (2011). Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, *2*(11), 2.

O'Hagan, A. (1979). On outlier rejection phenomena in bayes inference. *Journal of the Royal Statistical Society: Series B (Methodological)*, *41*(3), 358–367.

Ohlson, J. A. (1980). Financial ratios and the probabilistic prediction of bankruptcy. *Journal of accounting research*, 109–131.

Pickands III, J. et al. (1975). Statistical inference using extreme order statistics. *the Annals of Statistics*, *3*(1), 119–131.

Piironen, J., Paasiniemi, M. & Vehtari, A. (2018). Projective inference in high-dimensional problems: prediction and feature selection. *arXiv preprint arXiv:1810.02406*.

Piironen, J. & Vehtari, A. (2017). Comparison of bayesian predictive methods for model selection. *Statistics and Computing*, *27*(3), 711–735.

Piironen, J., Vehtari, A. et al. (2017). Sparsity information and regularization in the horseshoe and other shrinkage priors. *Electronic Journal of Statistics*, *11*(2), 5018–5051.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288.

Van Der Pas, S., Kleijn, B., Van Der Vaart, A. et al. (2014). The horseshoe estimator: Posterior concentration around nearly black vectors. *Electronic Journal of Statistics*, *8*(2), 2585–2618.

Vehtari, A., Gelman, A. & Gabry, J. (2015). Pareto smoothed importance sampling. *arXiv preprint arXiv:1507.02646*.

Watanabe, S. (2010). Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, *11*(Dec), 3571–3594.

List of Figures

List of Tables