Bachelor’s Programme in Science and Technology

Bayesian Regression Techniques for High-Dimensional Financial Time Series Data Structures

Otto Vintola

Aalto University

BACHELOR’S THESIS

Aalto University

BACHELOR'S THESIS 2024

Bayesian Regression Techniques for High-Dimensional Financial Time Series Data Structures

Otto Vintola

Thesis submitted in partial fulfillment of the requirements for
the degree of Bachelor of Science in Technology.
Otaniemi, 13 Dec 2024

Supervisor: Maarit Korpi-Lagg
Advisor: Postdoctoral Researcher Ersin Yılmaz

Aalto University
School of Science
Bachelor's Programme in Science and Technology

Author
Otto Vintola

Title
Bayesian Regression Techniques for High-Dimensional Financial Data Structures

School School of Science

Degree programme Bachelor's Programme in Science and Technology

Major Data Science **Code** SCI3095

Supervisor Maarit Korpi-Lagg

Advisor Postdoctoral Researcher Ersin Yılmaz

Level Bachelor's thesis **Date** 13.12.2024 **Pages** 35+4 **Language** English

Abstract

As the world adpots a culture of data-driven decision-making, the number of high-dimensional datasets increases. However, high dimensionality might bring issues such as ill-posedness, conflated models, and overfitting, thus necessitating shrinkage methods to reduce the dimensionality by selecting or penalizing the utilized features. Analogously, this thesis aims to find the most impactful stocks in a high-dimensional portfolio. Previous research regarding this topic has explored portfolios, high-dimensionality, sparsity, and Bayesian methods. Nevertheless, the literature concentrating on Bayesian shrinkage for high-dimensional portfolio data, is limited. Hence, this thesis aims to uncover sparsity present in one of the most popular portfolios in the world, the S&P500. Moreover, the training dataset consists of daily observations regarding the S&P500 and its constituent stocks across the years [2018, 2022], while the validation set is for the year 2023 alone. As the chosen methodology two common shrinkage priors, horseshoe and spike-and-slab, are placed on the Bayesian regression model. Conducting the trials reveals, that spike-and-slab provides superior predictive power over horseshoe. Spike-and-slab requires 174 unique stock ticers, corresponding to 180 regressors, for adequate predictive power measured by $\bar{R}^2$. The number of tickers could be researched through the selections made by credible intervals, however, they provide a lower bound for the number non-zero regressors required. Consequently, the contribution of this thesis is the uncovered sparsity in the S&P500 with Bayesian methods, along with a suggestion for the shrunk variable selection method by comparing the adjusted coefficient of determination and credible intervals.

Aalto-yliopisto

Tiivistelmä

Aalto-yliopisto, PL 11000, 00076 Aalto www.aalto.fi

Tekijä Otto Vintola	
Työn nimi Bayesialaiset Regressio Tekniikat Korkean Ulottuvuuden Rahoituksen Aikasarja Dataan	
Korkeakoulu Perustieteiden korkeakoulu	
Koulutusohjelma Bachelor’s Programme in Science and Technology yr	
Pääaine Datatiede	**Koodi** SCI3095
Vastuuopettaja Maarit Korpi-Lagg	
Ohjaaja Tutkijatohtori Ersin Yılmaz	
Työn laji Kandidaatintyö **Päiväys** 13.12.2024 **Sivuja** 35+4	**Kieli** englanti

Tiivistelmä

Maailman omaksuvan dataan perustuvan päätöksenteon kulttuurin, korkeaulotteisten tietoaineistojen määrä kasvaa. Korkea ulottuvuus voi kuitenkin tuoda mukanaan hankaluuksia, kuten huonosti asetettuja ongelmia, paisuneita malleja ja ylisovittamista, mitkä tekevät kutistusmenetelmistä välttämättömiä ulottuvuuden pienentämiseksi valitsemalla tai rankaisemalla käytettyjä selittäviä muuttujia. Vastaavasti tämä tutkielma pyrkii löytämään merkittävimmät osakkeet korkeaulotteisessa portfoliossa. Aiempi tutkimus on tarkastellut portfolioita, korkeaulotteisuutta, harvuisuutta ja Bayesilaisia menetelmiä. Kuitenkin kirjallisuus, joka keskittyy Bayesilaiseen kutistamiseen korkeaulotteisessa portfolioissa, on rajallista. Siksi tämä tutkielma pyrkii paljastamaan harvuisuutta yhdessä maailman tunnetuimmista portfolioista, S&P500:ssa. Tutkimusaineisto koostuu päivittäisistä havainnoista S&P500:sta ja sen osakkeista vuosilta 2018–2022, kun taas validointiaineisto kattaa vuoden 2023. Valittuna menetelmänä käytetään kahta yleistä kutistusjakaumaa, hevosenkenkä ja piikki ja laattaa, Bayesilaisessa regressiomallissa. Kokeiden suorittaminen paljastaa, että piikki ja laatta tarjoaa paremman ennustetarkkuuden kuin hevosenkenkä. Piikki ja laattaa vaatii ainakin 174 yksittäistä osaketunnusta, jotka vastaavat 180 selittävää muuttujaa, riittävään ennustetarkkuuteen, jota mitataan säädetyllä selitysasteella. Osaketunnusten määrää voitaisiin tutkia uskottavuusväleillä tehtyjen valintojen perusteella, mutta ne antavat alarajan vaadittavien selittävien muuttujien määrälle. Tämän tutkielman kontribuutio on siis näin ollen Bayesilaisilla menetelmillä paljastettu harvuisuus S&P500:ssa sekä ehdotus kutistettuun muuttujavalintaan vertailemalla säädettyä selitysastetta ja uskottavuusvälejä.

Notation

Symbol	**Meaning**
p	Variable representing a regressor or feature
y	Response variable
β	Coefficient in regression model
$\hat{\beta}$	Estimated parameter
$\bar{\beta}$	Mean of $p(\beta)$
σ^2	Variance of the error term
$\mathcal{D}$	The training dataset
k	Number of important regressors
n	Number of observations in the dataset
w_i	Weights of the stocks in a portfolio
$\mathbf{X}$	Regressor matrix
$\mathbf{y}$	Response matrix
$\mathcal{N}$	Normal distribution
$C^+(0,1)$	Half-Cauhcy distribution
$\mathbf{\Lambda}$	Prior covariance of parameters
$\mathbf{\Sigma}$	Posterior covariance of parameters
A^T	Transpose of matrix A
I	Identity Matrix
$A \circ B$	Hadamard product between matrix A and B
π	Prior inclusion probability
c^2	Slab scale
ϵ^2	Spike scale
λ_j	Local shrinkage coefficients
τ	Global shrinkage coefficient

Contents

1. Introduction

Is it probable that probability brings certainty?

— Blaise Pascal

Already now, there are clear trends pointing to the future world being built upon data. In this day and age, collecting an enormous amount of data and pondering the purpose afterwards is commonplace. The cost of gathering data has plunged, which has led to datasets with a large number of regressor[1] variables p [1].

Big data is discussed extensively in the media and industries today. However, the specification of it is usually ambiguous. Data can be considered big with respect to the number of observations, or the number of regressors—sometimes referred to as long and wide data. Big data can be defined as data with a high number of regressors compared to observations. This is supported by the fact that the term dimension can be thought of as the number of axes required to define a single data point in the specified space.

To highlight the issue with high-dimensional data, let us consider choosing the regressors for a regression model from a high-dimensional dataset. While it may seem reasonable to use all of the available ones, if the number of regressors is larger than the number of observations, the problem becomes ill-posed, and the inverse of the regressors Gram matrix [2] does not exist[2] Consequently, calculating the covariance of the parameter estimates becomes impossible, making it inconceivable to quantify the uncertainty associated with them.

Another motivation for dimensionality reduction is that computing power is likely to be regarded as a commodity, much like oil or gold. There-

[1] In literature concerning statistics and machine learning the term regressor is sometimes called a feature, however, due to the context of Bayesian regression, the term regressor is mainly used, however, feature appears occasionally—they can be regarded as synonyms for this thesis.

[2] The proof for this is presented later.

fore, any attempt at reducing the model size and hyperparameter search space makes training cost-effective, and accessible, allowing for faster experimentation and iteration in development. This is especially relevant with the advent of deep learning where model training can take weeks or even months. Furthermore, reducing dimensionality might foster more open-source collaboration from developers who lack access to large-scale compute clusters.

Last but not least, the risk of overfitting the model increases when the number of regressors p exceeds the number of observations n [3]. Optimizing the loss function over a dataset $\mathcal{D}$ with a model that is too complex, i.e., possessing too many regressor variables β_i, leads to the optimization procedure fitting the model to the training dataset with the loss of generalization performance.

These struggles related to dimensionality were coined the "curse of dimensionality" by Richard Bellman [4] while discussing the hardships of optimizing by enumeration on product spaces. At present, Bellman's maxim is used to describe any problem arising from either too few or many dimensions. Due to the curse of dimensionality, reducing the data to a few variables that capture most of its variance is immensely helpful. With this in mind, the objective is not to thoroughly search the entire space but to apply shrinkage and probabilistic methods to obtain the optimal model.

Shrinkage methods are widely utilized for reducing model complexity by determining the most important regressors and attempting to de-emphasize the less important ones. In genomics, for example, they could possibly discover the relevant gene expression for cancer classification. Traditional shrinkage methods, such as the elastic net [5] or ℓ_1 and ℓ_2 regularization [6], are widely used today for their simplicity and effectiveness. The principle behind these methods is to penalize the utilization of weights in the model to boost generalization performance.

Bayesian shrinkage methods approach shrinkage from a probabilistic point-of-view by applying a sparsifying prior to the model parameters, constraining their size. Treating the model parameters as samples from a distribution allows for the inclusion of prior information about the regressors and enables the quantification of uncertainty in the parameter estimates [7].

By applying the outlined methods, constructing models that are more explainable and interpretable becomes possible while also being easier to construct, scale, and train. Ultimately, utilizing these techniques can enhance the robustness of decision-making and improve predictive accuracy in complex problem settings.

Therefore, the goal of this thesis is to identify and discover the possible sparsity with Bayesian shrinkage and probabilistic approaches on high-dimensional financial time series data by conducting a state-of-the-art literature review. Moreover, a series of tests are performed, and the effectiveness of the methods is evaluated on a contrived example portfolio.

This thesis is structured as follows, in Section 2, the nature of high-dimensional data and its relation to this thesis. Afterwards, Section 3 presents the statistical modeling techniques applied to the data and introduces linear regression, generalized linear models and Bayesian regression. Section 4 discusses the methods to shrink the space of regressors as well as other variable selection methodologies. Section 5 describes the trials which test the hypothesis. The results are collected and interpreted in Section 6, with Section 7 closing out the thesis.

2. Data

In God we trust; all others bring data.

— W. Edwards Deming

This section considers the definition of high-dimensionality and its impact on genomics and finance. Additionally, a brief description about portfolio optimization and the dataset is included as context for the trials in Section 5.

2.1 High Dimensionality

As discussed in the introduction, high-dimensional data refers to information with a large number of regressor variables p compared to the number of observations n—typically, a dataset $\mathcal{D}$ is collected to represent a subset of this information. More formally, the dataset[1] can be expressed as

$$\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\},$$

where n is the number of observations (or samples) and p is the number of dimensions (or regressors) for each observation, such that each x_i is a p-dimensional vector

$$x_i \in \mathbb{R}^p \quad \text{for } i = 1, 2, \ldots, n.$$

Furthermore, p does not have to be large for the dataset to be high-dimensional. As long as the ratio p/n is large, the dataset can be regarded as high-dimensional. This can happen in two ways, either p is large or n is small. An extreme form of this situation is when both of them occur

[1] In many situations the formal definition of dimensionality is not required, but in this specific field there are many misconceptions about the term high-dimensional.

simultaneously. In this case, $p \gg n$, which would mean the ratio between the number of regressors and observations is extremely large.

There are common characteristics that occur when analyzing and utilizing high-dimensional data. One of them is overfitting. When the dataset is high-dimensional, there is insufficient data to train a model that can adequately describe the relationship between the regressors and the response variable. In this situation, the model likely learns to represent the noise in the dataset, instead of modeling the underlying distribution from which the data is sampled. Nevertheless, Section 4 expands on the motivation for dimensionality reduction by discussing variable selection, explainability, and model training.

2.2 Common Problems in High-Dimensional Data

High-dimensional data is ubiquitous in several application fields, where each domain brings unique characteristics and challenges. In natural language processing, for example, each unique word can represent a feature, meaning the number of features p is equal to the size of the vocabulary in the combined documents. Or in computer vision, each pixel of a flattened image could represent a feature for the model. Similarly, other domains exhibit high-dimensionality in the data, but it is especially common in genomics and finance.

Usually, genetic data is concerned with assessing thousands of genes simultaneously—each genetic variation is included as a feature. The human genome, for example, has approximately 20 000 genes, and millions of potential single nucleotide polymorphisms (SNPs) [8] that need to be analyzed. Furthermore, these datasets have a modest number of samples due to the limited availability of genetic donors or the practical issue of collecting and sequencing the data. Additionally, the collection, analysis and storage of genetic data might also be subject to ethical and legal battles. Processing sensitive genetic information requires wading through myriads of laws like the GDPR, which restrain the sharing and storing of data. Consequentially, it is common for genetic datasets to be high-dimensional.

Likewise, in finance, high-dimensional datasets are might arise. Not only is the number of assets in portfolios growing, but also the number of ways to measure the performance with indicators such as volatility, returns or price-to-earnings [9]. Moreover, it is common to incorporate non-numeric data, such as news articles or financial transactions, further increasing the dimensionality and risk that the true signal is obscured by noise. Additionally, in some situations there might be limited data, for example, when the instrument has not been listed on the stock exchange for a long time. In these situations, methods to sparsify the feature space are required to make meaningful insights.

2.3 Contexts for the Experiments

The trials in this text will be concerned with portfolio optimization, which is the practice of constructing a collection of financial instruments—in our case stocks—into a portfolio, and then analysing the risk-reward ratio to identify which instruments are worth keeping and which ones could be replaced. A common portfolio optimization technique was described by Markowitz in 1952 [10]. In his article, Markowitz framed the problem mathematically as a quadratic optimization problem aimed to find the optimal portfolio weights by maximizing the portfolio's expected return while minimizing the risk. So, the optimization objective is

$$\max_{\mathbf{w}}(\mathbf{w}^T\boldsymbol{\mu}), \tag{2.1}$$

where $\mathbf{w} = (w_1, w_2, w_3, \ldots, w_i)^T$ is the weight vector and $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_i)^T$ is the expected return of the instruments in the portfolio. Furthermore, there are three additional constraints to this procedure.

1. All of the weights have to be non-negative, so $w_i \geq 0, \forall i$. This requirement essentially blocks the possibility to place negative bets on an instrument, also known as short selling.

2. The weights have to sum up to 1, $\sum_{i=1}^{n} w_i = 1$. Concretely, this requirement states that the whole investment has to be fully allocated.

3. The last constraint is risk minimization. Mathematically it means $\mathbf{w^T}\Sigma\mathbf{w} \leq \sigma^2$, where σ^2 is the maximum variance or equivalently risk that the investor is willing to tolerate, and $\Sigma \in \mathbb{R}^{n\times n}$ is the covariance matrix of asset returns.

This same procedure serves as the foundation for our trials as well. The objective is to find the optimal weights for an arbitrary portfolio of stocks, while being subject to similar constraints. Roughly, the trials have a similar outlook and goal, however, the methods used to arrive to the optimal portfolio weights consist of Bayesian regression techniques instead of quadratic optimization. Additionally, the procedure will not be guided by expected returns, but by other engineered financial metrics discussed in Section 5.

2.4 Source, Retrieval and Preprocessing

The portfolio that is optimized consists of the same tickers that have been in the Standard & Poor’s 500 index (S&P500), which is a popular benchmark to assess the performance of a portfolio, for the [2018, 2022] period. The goal of this construction is to gather enough features p for the dataset to be high-dimensional, while keeping the number of observations n constrained, but still high enough that overfitting, even after sparsifying the feature space, is not a concern.

It is important to note here, that the dataset is not chosen to fit the methods in this paper. Rather, the opposite is done, by starting from the question, can Bayesian methods be utilized to find the optimal set of assets and corresponding weights, and then conduct trials to examine this hypothesis.

The data is fetched through the Yahoo Finance API, which returns the stock prices and other metrics used to calculate the features for the trials—further feature engineering details are explained in Section 4. Some cleaning is done to the data, for example, NaN values are replaced with linear interpolation and basic wrangling tasks are required to format the data. At the end, a typical time series dataset is constructed where $n = 1260$ and $p = 3018$. Only some regressors are included, and some of them are replaced in the trials. Table 2.1 shows the stripped version of the data.

Furthermore, a test dataset is collected to evaluate the performance of the models subsequent to the trials. It contains 250 observations continuing from the last date of the training dataset. Otherwise, there are not further differences between the two datasets.

Date	**Adj Close A**	**Adj Close AAPL**	...	**Volume ZBH**	**Volume ZBRA**	**Volume ZTS**
2018-01-02	64.298866	40.568928	...	1818259.0	310600.0	2135600.0
2018-01-03	65.934891	40.561863	...	1368664.0	253000.0	2328200.0
2018-01-04	65.440254	40.750267	...	1105396.0	435200.0	2534000.0
2018-01-05	66.486549	41.214237	...	1095302.0	301800.0	2166100.0
⋮	⋮	⋮	⋮	⋮	⋮	⋮
2022-12-27	147.500687	128.818420	...	755600.0	252500.0	957900.0
2022-12-28	146.060684	124.865593	...	750100.0	241200.0	1443900.0
2022-12-29	149.019592	128.402344	...	686600.0	274900.0	1298900.0
2022-12-30	147.819443	128.719345	...	785200.0	228200.0	1249500.0

Table 2.1. Stock Indicators as time series as the result of data wrangling.

3. Models

The map is not the territory.

— Alfred Korzybski

The following is a brief description of statistical modeling, linear regression, generalized linear models, and Bayesian regression.

3.1 Statistical Models

In the universe, there are many phenomena that cannot be explained, as of writing this thesis, by science in a deterministic way. Complex systems and dependencies, such as those in biology, economics, or social sciences are characterized by randomness. If the behavior of an individual could be modeled deterministically, then it could imply the absence of free will. However, the individual could always act opposite to what the model is instructing, leading to some peculiar conclusions.

So, instead of relying on deteministic models of the world, situations where the outcomes are perturbated by random variability—noise—are especially suited for statistical models. They are often constructed to account for the inherent variance exhibited by real-world phenomena. By adjusting for the noise, statistical models are able to provide insights into the real factors[1] affecting the causality.

The term statistical model is rather of an umbrella term. Generally, they are applied when there is a dataset $\mathcal{D}$ that contains the perceived constituents x_i of the outcome y_i, but there is an unmarked factor ϵ_i that is not quantized in x_i. The dependencies between x_i and y_i can be linear, but also more complex as will be described in the subsection regarding

[1] This is a general term for anything that accounts for the variance in phenomena, in the context of this thesis, it is synonymous with regressor.

Generalized Linear Regression. However, to begin with the simplest form of a statistical model—linear regression [11].

3.2 Linear Regression

Linear regression is one of the most fundamental statistical models for capturing a linear relationship between regressors x_i and the response variable y_i. Widely used in finance, genomics, and physics due to its simplicity, ease-of-use and interpretability. Additionally, it forms the basis for the more complex models used in the trials. At its core, linear regression is a weighted sum of the regressors calibrated by the parameters of the model. Mathematically expressed as:

$$y_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \ldots + x_p\beta_p + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2), \quad i = 1, \ldots, n, \tag{3.1}$$

where y_i is the response variable, x_{ip} represents the regressor, β_0 is the intercept term, while β_j for $j = 1, 2 \ldots p$ are the parameters responsible for adjusting the regressors, and ϵ_i is the independent random variability, the noise discussed earlier, that is present in the dataset $\mathcal{D}$ but is not captured by the model.

Often, 3.1 is presented in vectorized form, which makes the representation more compact and the subsequent derivations and proofs simpler. Let $\mathbf{y} = (y_1, y_2, \ldots y_i)^T \in \mathbb{R}^{n\times 1}$ represent the vector form of the response variables, and let $\mathbf{X} = (\mathbf{1}, \mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_i)^T \in \mathbb{R}^{n\times(p+1)}$ contain all of the observations of the regressors, and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots \epsilon_i)^T$ be the noise vector, then 3.1 can be indicated more compactly as:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{3.2}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{(p+1)\times 1}$ is the vector of model parameters $(1, \beta_0, \beta_1, \ldots \beta_p)^T$. The all-ones vector is concatenated as the first column in $\mathbf{X}$, and an extra 1 is added to the beginning of $\boldsymbol{\beta}$ so that it is not necessary to carry around β_0 in all of the calculations.

The goal of linear regression is to find the best estimates for the parameters $\beta_0, \beta_1 \ldots \beta_p$, which is usually performed by calculating the ordinary least-squares (OLS) estimate. Crucially to this thesis, the assumption for OLS is that $p < n$, which is an issue in high-dimensional data, due to the ill-posedness of the modeling problem—the proof for this can be found Section 4.

In simple terms, OLS is an optimization problem for finding the parameter estimates $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p)$ that minimize the sum of squared noise. The noise can be calculated as the difference between the real response variable values $\mathbf{y}$ and the predicted one $\hat{\mathbf{y}} = X\hat{\boldsymbol{\beta}}$. The formulation for the minimization can be expressed as:

$$\min_{\hat{\boldsymbol{\beta}}}(\mathbf{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}})^T(\mathbf{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}) = \min_{\hat{\boldsymbol{\beta}}}(\mathbf{y}^T\mathbf{y} - 2\hat{\boldsymbol{\beta}}^T\mathbf{X}^T\mathbf{y} + \hat{\boldsymbol{\beta}}^T\mathbf{X}^{\mathbf{T}}\mathbf{X}\hat{\boldsymbol{\beta}}).$$

To which the solution can be found by taking the derivative with respect to the parameter estimates $\hat{\boldsymbol{\beta}}$, and setting it equal to zero:

$$\frac{\partial}{\partial\hat{\boldsymbol{\beta}}}\left(\mathbf{y}^T\mathbf{y} - 2\hat{\boldsymbol{\beta}}^T\mathbf{X}^T\mathbf{y} + \hat{\boldsymbol{\beta}}^T\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}}\right) = -2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} = 0.$$

This yields the equation:

$$\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y}.$$

And then finally solving for $\hat{\boldsymbol{\beta}}$ gives the answer:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.$$

An important detail to note here, is that the matrix $(\mathbf{X}^T\mathbf{X})$ has to be invertible, in order for the OLS procedure to be possible, however, this is generally uncommon for high-dimensional data.

While linear regression is extremely useful, many of the dependencies in the real world, between regressors and response variables, are not linear. Furthermore, the response variable is assumed to be sampled from a normal distribution, which may not apply invariably. Due to these inabilities a more robust model has been conceived.

3.3 Generalized Linear Regression

As per the name, generalized linear models [12] provide a robust alternative to 3.2. The key differences are that the relationship between the regressors and response variable does not have to be linear, and the response variable no longer has to be sampled from a normal distribution, instead it is assumed to originate from the exponential family of distributions, such as the normal, binomial, or Poisson distribution.

The non-linear relationship between variables is expressed through a linear predictor that is passed through a link function that is usually denoted with $g(\cdot)$. Expressing the linear predictor as:

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p.$$

Then denoting $\mu = E(Y)$ where Y is the response variable as a random variable. Then the relationship can be expressed:

$$g(\mu) = \eta,$$

where the choice of the link function depends on the assumed distribution of Y. Some examples are the logit link $g(\eta) = \log(\frac{\mu}{1-\mu})$ for the binomial distribution, or the log-link $\eta = \log(\mu)$ for the Poisson distribution.

3.4 Bayesian Regression

GLMs are great and robust tools, but they do not estimate how uncertain the model parameters are in a probabilistic way. Expressing the parameters β_i of the model as random variables opens the possibility to quantify the uncertainty and incorporate prior information into the parameters. These advantages are what make Bayesian regression particularly valuable for high-dimensional financial data.

The setup is similar as in 3.1, there is a linear model for predicting the response variable $\mathbf{y}$ with parameters $\boldsymbol{\beta}$ and regressors $\mathbf{X}$ with independent noise $\boldsymbol{\epsilon}$:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma I), \boldsymbol{\beta} \sim \mathcal{N}(0, 1). \tag{3.3}$$

However, as this model is constructed from the Bayesian point-of-view, the parameters are assumed to follow some prior distribution [13]. For instance, a Gaussian prior with mean 0 and Λ covariance, can be assumed by:

$$\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Lambda}), \tag{3.4}$$

where $\boldsymbol{\Lambda}$ can be conceived as encoding the prior beliefs for the parameters. By virtue of Bayes' theorem, the posterior distribution $p(\boldsymbol{\beta}|\mathbf{y}, \mathbf{X})$ for the model parameters can be constructed as:

$$p(\boldsymbol{\beta}|\mathbf{y}, \mathbf{X}) \propto p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{X})p(\boldsymbol{\beta}). \tag{3.5}$$

Now it is possible to construct a confidence interval for parameter estimates, which can give meaningful insights into how uncertain the model is. In-order to calculate the posterior distribution by

$$p(\boldsymbol{\beta}|\mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{X})p(\boldsymbol{\beta})}{p(\mathbf{y}|\mathbf{X})}, \tag{3.6}$$

the whole parameter space needs to be integrated over with

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta})p(\boldsymbol{\beta})d\boldsymbol{\beta}. \tag{3.7}$$

As a result, the posterior distribution remains Gaussian [14]

$$p(\boldsymbol{\beta}|\mathbf{y}, \mathbf{X}) = \mathcal{N}(\boldsymbol{\beta}|\bar{\boldsymbol{\beta}}, \boldsymbol{\Sigma}), \tag{3.8}$$

where the covariance and mean can be calculated as

$$\boldsymbol{\Sigma} = (\boldsymbol{\Lambda}^{-1} + \frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X})^{-1} \tag{3.9}$$

$$\bar{\boldsymbol{\beta}} = \frac{1}{\sigma^2}(\boldsymbol{\Lambda}^{-1} + \frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{3.10}$$

So, from linear regression through GLMs and now to Bayesian regression, the original setup of 3.2 has conceptually remained the same, but the important details have been altered according to the Bayesian point-of-view. Importantly, the notion of Bayesian regression is used as the foundational modeling tool for the trials. Nevertheless, there are still some issues to be addressed.

Bayesian models tend to be computationally expensive—especially when the number of regressors becomes large. If the number of parameters grows with the number of regressors, then the integration space in 3.7 becomes substantial quickly. Consequently, numerical integration methods might be needed, if the integral does not have a closed-form solution.

To obtain parameter estimates sampling from the posterior distribution is often done with Markov Chain Monte Carlo (MCMC) methods. However, they might require a vast amount of computational resources to converge on high-dimensional regressor spaces [15]—or they might not converge at all. These challenges give rise to the need for methods to shrink the number of regressors, while maintaining the meaningful signal present in the data.

4. Shrinkage and Bayesian methods

It is useless to do with more what can be done with less.

— William of Ockham

The following part of this paper moves on to describe in greater detail the reason shrinkage methods are utilized, and two common Bayesian ones.

4.1 Motivation

Conceptually, if it is possible to shrink the complexity without sacrificing the performance, then it would be wise to do so. This hinges on the assumption of sparsity, which states that the meaningful signal is captured by a few meaningful factors, not by the combination of all of them. For high-dimensional modeling problems, sparsity is the optimal scenario when using shrinkage methods. The goal is to find the few meaningful variables that are responsible for modeling most of the variance in the dataset.

Typically, this procedure can reduce computational complexity by reducing the dimensionality of model. However, in some situations shrinkage is a necessity to continue the modeling. Here are some example scenarios where shrinkage techniques are necessary or beneficial.

4.1.1 Ill-posedness

In his seminal 1902 paper [16], the French mathematician Jacques Hadamard defined a problem to be well-posed if a unique solution that depends continuously on the input data exists. As one might guess, a problem is ill-posed if it does not satisfy the previous condition. The ill-posedness of a regression problem appears in different ways, often emerging as instability in the parameter estimates or sensitivity to noise in the input data.

Moreover, if $p > n$, the matrix $\mathbf{X}^T\mathbf{X}$ does not have an inverse. To quantify the uncertainty associated with the parameter estimates, the posterior distribution needs to be defined. However, when the inverse of $\mathbf{X}^T\mathbf{X}$ does not exist, the mean and covariance described in 3.9 might not be well defined, if no shrinkage prior covariance $\mathbf{\Lambda}$ is defined.

Proof for the inexistence of $\mathbf{X}^T\mathbf{X}$ when $p > n$:
From the definition, the rank of a matrix A is the maximum number of linearly independent columns or rows in A. From the definition of rank, it follows that if $\mathbf{X} \in \mathbb{R}^{n \times p}$, then rank($\mathbf{X}$) $\leq min(n, p)$. Moreover, if $p > n$, then rank($\mathbf{X}$) $\leq n$.

The rank of $\mathbf{X}^T\mathbf{X}$ is equal to the rank of $\mathbf{X}$, so it follows that:

$$rank(\mathbf{X}^T\mathbf{X}) \leq n < p.$$

This means that $\mathbf{X}^T\mathbf{X}$ is rank-deficient columnwise, since $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{p \times p}$ needs to have p linearly independent columns to be considered full rank. A matrix is invertible if and only if it is non-singular. But, the matrix $\mathbf{X}^T\mathbf{X}$ is non-singular, since it is not full rank. Consequently, the inverse $(\mathbf{X}^T\mathbf{X})^{-1}$ does not exist.

The ill-posedness of the Gram matrix [2] $\mathbf{X}^T\mathbf{X}$ can similarly occur when the data exhibits a high degree of correlation amongst regressors, a situation that is categorized as multicollinearity. In the presence of multicollinearity, the correlated columns are linearly dependent, which results in the matrix inversion problem once more.

Moreover, multicollinearity can inflate the variances in the parameter estimates, which makes them highly dependent on the small changes in the training data. As a consequence, the predictions from the model become unstable and the parameter estimates less interpretable. A natural solution for multicollinearity involves constraining or penalizing the excessive use of parameter estimates.

4.1.2 Variable selection

Performing the standard forward-backward selection algorithm to find the smallest subset of variables, becomes very slow for high-dimensional datasets [17]. For low-dimensional datasets the procedure is still practical, but extremely slow for datasets with thousands of regressors. Thus, backward-forward selection is not a reasonable option for variable selection when handling high-dimensional data and consequently a more sophisticated method is required.

A further concern in variable selection is the multiple comparisons problem [17], which arises when performing many statistical tests for the parameter estimates at once. The intuition is that conducting m statistical tests with confidence level $1 - \alpha$ means the expected number of incorrect inferences is $m\alpha$. Consequently, the probability of making one or more erroneous inferences is $1 - (1 - \alpha)^m$. As an illustration, let us choose $\alpha = 0.05$ and the number of variable combinations $m = 50$. Then the probability of making at least one erroneous inference is $1 - (1 - 0.05)^{50} \approx 0.92$—which is alarmingly high.

4.1.3 Overfitting

Training with high-dimensional datasets often produces models that are too complex. This scenario, known as overfitting, [3] occurs when the model has learned to represent the training data—along with the noise—closely. The objective of training is, however, to increase the generalization performance by learning the underlying distribution from which the dataset is gathered from, instead of the specific dataset in use. Ensuring that the model is reliable and robust beyond the seen observations can be achieved by selecting only the most important regressors.

4.1.4 Explainability

Let us consider a linear regression model where the number of parameters is extremely large, and each parameter represents some real world quantity. Explaining why the output variable y_i receives the value for a given input x_i becomes increasingly difficult when including more-and-more factors into the model. This can be, for instance, because each parameter dilutes the effect of the existing ones, or the possible appearance of interaction terms between the regressors.

For example, consider modeling house prices predicted by square footage, number of neighbors, distance to schools, and the number of bedrooms. So far, explaining the prediction is straightforward, however, adding factors about, i.e., age of the home and the energy efficiency may create unintuitive predictions. An older home with a high energy efficiency might be valued higher than a new home with the same energy efficiency, even if generally newer homes are valued higher.

4.2 Shrinkage and Variable Selection with Sparsifying Priors

The aforementioned reasons are plenty to motivate the usage of sparsifying methods. The objective is to reduce the disadvantages high-dimensionality brings by effectively reducing the number of regressors through penalizing or otherwise constraining model parameters. By introducing penalties that shrink the insignificant coefficients toward zero or to exactly zero, these methods help to simplify the model, reduce the risk of overfitting, ease variable selection, and reduce multicollinearity.

The frequentist approach to shrinkage is to apply a penalty to the parameters when performing OLS. Two common penalties are the ℓ_1 norm and ℓ_2 norm. These methods are, respectively, called least absolute shrinkage and selection operator (LASSO) [18], and ridge regression [18]. The objective in LASSO is to minimize the following expression:

$$\min_{\beta} \left(\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right),$$

where $\lambda \geq 0$ is the regularization parameter that controls the shrinkage and $\lambda \sum_{j=1}^{p} |\beta_j|$, is the ℓ_1 penalty constructed with λ. As for ridge regression, the minimization objective is:

$$\min_{\beta} \left(\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right).$$

The equations look quite similar—in fact the only difference is the exponent for β_j in the rightmost sum. However, the results that LASSO and ridge regression produce are slightly different. While LASSO forces some coefficients to be exactly zero, ridge regression only shrinks them closer to zero. Consequently, LASSO produces sparse models while ridge regression retains all predictors but reduces their impact by shrinkage.

Sparsifying priors, however, handle shrinkage from a Bayesian point-of-view by applying a prior distribution on the model parameters β_i that encourage sparsity. Consider the standard Gaussian linear regression model presented in 3.1. Let $p(\beta)$ denote the sparsifying prior, then the optimization problem, similar to LASSO and ridge, can be expressed in terms of 3.5 and the prior by:

$$\hat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} \left(\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) + \log p(\boldsymbol{\beta}) \right),$$

where $\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta})$ is the log-likelihood, and $\log p(\boldsymbol{\beta})$ is the log-prior term. The intention is to maximize the likelihood that assesses model fit while being under the constraint of the prior distribution.

The type of shrinkage depends on the choice of prior distributions. Analogously to frequentist shrinkage methods, with certain prior distributions, the insignificant coefficients can be either forced exactly to zero or all of the parameters can be shrunk toward it. The choice depends on the type of prior distribution [7]. Often, priors are defined by placing a hyperprior on the prior covariance $\boldsymbol{\Lambda}$ in 3.4. Table 4.1 presented in [14, Tab. 2.1] shows some common shrinkage methods and their corresponding hyperpriors.

Name	**Prior**	**Hyperprior**
Gaussian	$\mathrm{N}(0, \tau^2\lambda_j^2)$	$\lambda_j = 1$
Student-t_ν	"	$\lambda_j^2 \sim$ Inv-Gamma $\left(a = b = \frac{\nu}{2}\right)$
Laplace	"	$\lambda_j^2 \sim$ Exp$(b = 2)$
Horseshoe	"	$\lambda_j \sim \mathrm{C}^+(0, 1)$
Regularized horseshoe	$\mathrm{N}(0, \tau^2\xi_j^2)$	$\xi_j^2 = \frac{c^2\lambda_j^2}{c^2+\tau^2\lambda_j^2}, \quad \lambda_j \sim \mathrm{C}^+(0, 1)$
Spike-and-slab	$\mathrm{N}(0, c^2\lambda_j^2)$	$\lambda_j \sim$ Ber(π)

Table 4.1. Priors and Hyperpriors

For the trials, horseshoe is chosen due to its ability to identify sprase signals in the data and adapt to noise and possible outliers in the data. Spike-and-slab is utilized because it provides an interpretable result of the predictors due to its binary nature of a regressor being either useful or not useful. The combination of these two prior distributions provides a diversified possibilty of discoveries in the trials, exploring both the binary nature of the regressors and the scattered sparsity perhaps present.

4.3 Horseshoe

Similarly to the frequentist ridge regression, horseshoe prior [7], reduces all coefficients toward zero. The horseshoe prior is formulated as a continuous mixture of Gaussian distributions:

$$\beta_j \mid \lambda_j, \tau \sim \mathcal{N}(0, \lambda_j^2\tau^2), \tag{4.1}$$

$$\lambda_j \sim C^+(0, 1), \tag{4.2}$$

where $C^+(0, 1)$ is the half-Cauchy distribution, λ_j and τ are respectively the local and global shrinkage factors. The intuition behind the method is that τ attempts to shrink all coefficients toward zero, while λ_j allows a few to escape it. As for the posterior distribution in 3.5, the mean can be written as

$$\bar{\beta} = (1 - \kappa_j)\beta_j,$$

where the shrinkage factor κ_j is calculated as

$$\kappa_j = \frac{1}{1 + n^2\sigma^{-2}\lambda_j^2\tau^2}, \tag{4.3}$$

and σ is the variance of the noise.

Compared to the Student's t-distribution and Gaussian, Figure 4.1 shows that the distribution of the horseshoe, with $\tau \in [0.5, 0.8, 1]$, is sharper around zero. Additionally, the density of the shrinkage factor is plotted, which shows it favors values around zero and one.

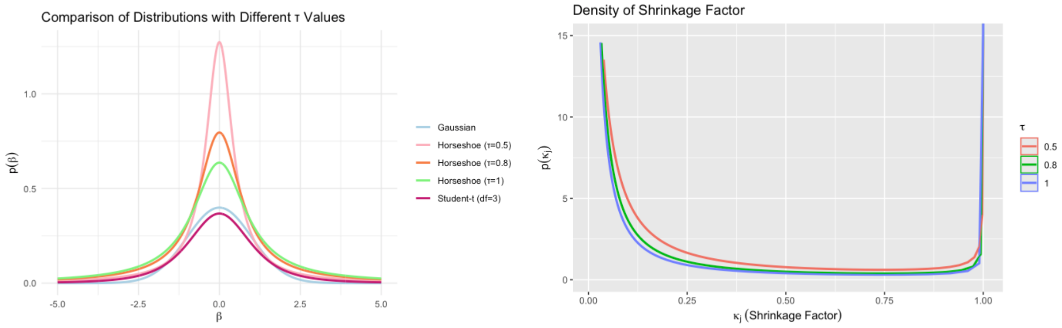


Figure 4.1. Horseshoe prior and density of shrinkage factor

4.4 Spike-and-Slab

Another popular prior choice is a mixture of two Gaussians, often called the spike-and-slab prior [7]. It can be expressed mathematically as:

$$\beta_j \mid \lambda_j, c, \epsilon \sim \lambda_j \cdot \mathcal{N}\left(0, c^2\right) + (1 - \lambda_j) \cdot \mathcal{N}\left(0, \epsilon^2\right), \tag{4.4}$$

$$\lambda_j \mid \pi \sim \text{Bernoulli}(\pi), \quad j = 1, \ldots, p, \tag{4.5}$$

where ϵ is much smaller than c and $\lambda_j \in \{0, 1\}$ is considered the indicator variable because it decides whether the coefficient β_j is non-zero or not. In fact, the *spike* comes from $\lambda_j = 0$, which samples values from a distribution that is *narrow*, so has considerable density at zero. Conversely, the *slab* comes from the *flat* distribution, which indicates sampling from a wide range of values.

The formula for the shrinkage factor 4.3 stays the same as in the horseshoe prior, however, the density distribution plotted against $\mathcal{N}(\mu = 0, \sigma = 0.5)$ and

Student’s t-distribution is quite different.

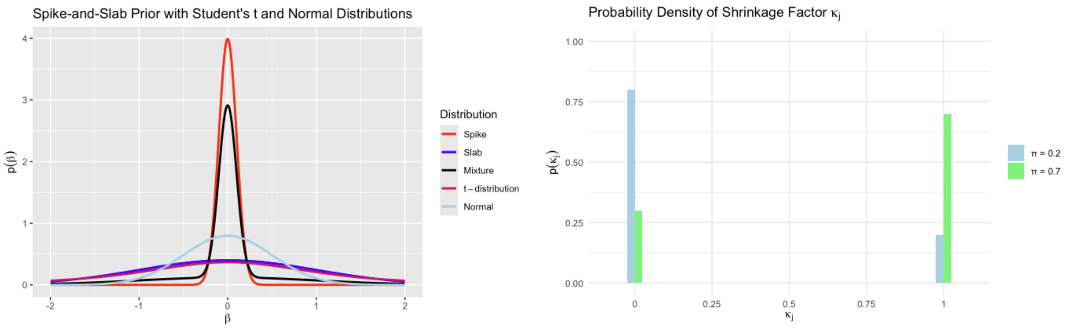


Figure 4.2. Spike-and-slab prior and density of shrinkage factor

Figure 4.2 highlights the conceptual functionality of spike-and-slab. The graph on the left shows the spike-and-slab distributions coming together to form the mixture distribution. The graph on the right visualizes the probability density function for two different prior inclusion probabilities $\pi = 0.2$ and $\pi = 0.7$.

5. Trials

The best time to plan an experiment is after you've done it

— Sir Ronald A. Fisher

The section below describes the experiments done to assess the performance of Bayesian shrinkage methods on financial data. The actual evaluation is discussed in Section 6.

5.1 Experimental Hypothesis

As a recap, the goal of the experiments is portfolio optimization [10]. So, finding the optimal stocks to include in the portfolio by analyzing the returns. In Section 2, constraints one, two, and three were presented for portfolio optimization, however, only the first two are followed as a guideline to the experiments, since risk minimization is not relevant in the scope of this thesis.

Nevertheless, the goal is to eliminate stocks that do not contribute to the performance of the portfolio. For these experiments the performance is characterized as daily returns (DR). The optimal weights $\mathbf{w}^*$ are determined by finding the few regressors that explain most of the variance of the data, and assigning weights to them depending on their contribution.

So, a small modification to the original equation 2.1 presented now as:

$$\mathbf{w}^* = \underset{w}{\operatorname{argmax}} \operatorname{Var}(\mathbf{X}^T \mathbf{w}), \tag{5.1}$$

where $\mathbf{X}$ is the feature vector. The precise procedure of assigning $\mathbf{w}^*$ is described later, when the results of the shrinkage are presented and analyzed.

5.2 Feature Engineering

The data has columns for adjusted closing price (AC), closing price (C), high (H), low (L), open (O), and volume (V). Additionally, the exponential moving average (EMA) is calculated as:

$$EMA_t = \alpha \cdot AC_t + (1 - \alpha) \cdot EMA_{t-1},$$

where EMA_t is EMA at time t, AC_t is the adjusted closing price at t, and $\alpha = \frac{2}{N+1}$ where N is the number of periods—20 is chosen. Additionally, the daily returns DR_t are calculated as features. After engineering the features they are all zero mean and unit variance normalized. Mathematically expressed as:

$$\text{Normalized Feature} = \frac{x - \mu}{\sigma}$$

As the response variable, the DR of the entire portfolio is calculated with:

$$DR_t = \frac{AC_t - AC_{t-1}}{AC_{t-1}}. \tag{5.2}$$

Since DR_t is a time series variable, it might exhibit autocorrelation, which means the noise terms ϵ_i are not independent, which breaks the assumption made in 3.3 regarding $\boldsymbol{\epsilon}$. However, examining the autocorrelation presented in 5.1, shows that there is no statistically significant evidence for autocorrelation. Additionally, neither the Ljung-Box [19] nor Durbin-Watson [20] tests indicate any statistically significant autocorrelation in DR_t. Thus, the concern about autocorrelated response variables can be set aside.

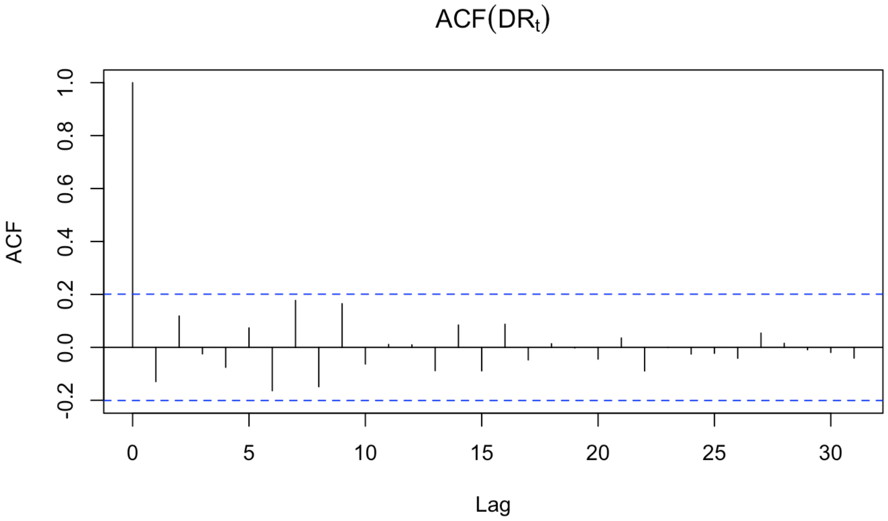


Figure 5.1. Autocorrelation Plot

No further normalization or feature engineering is done. As the end result, a dataset with $n = 1259$—same as before, and $p = 4026$ is constructed, where 4025 are the features to be used and the one excluded is the response variable DR_t.

5.3 Model training

The feature engineering was mainly done with `Pandas` [21], however, for model training the language is switched to `R`. For the experiments a model using the horseshoe and spike-and-slab is trained using the `RStan` package [22]. The code for model training can be found in the GitHub[1] repository. Figure 5.2 highlights the complete procedure

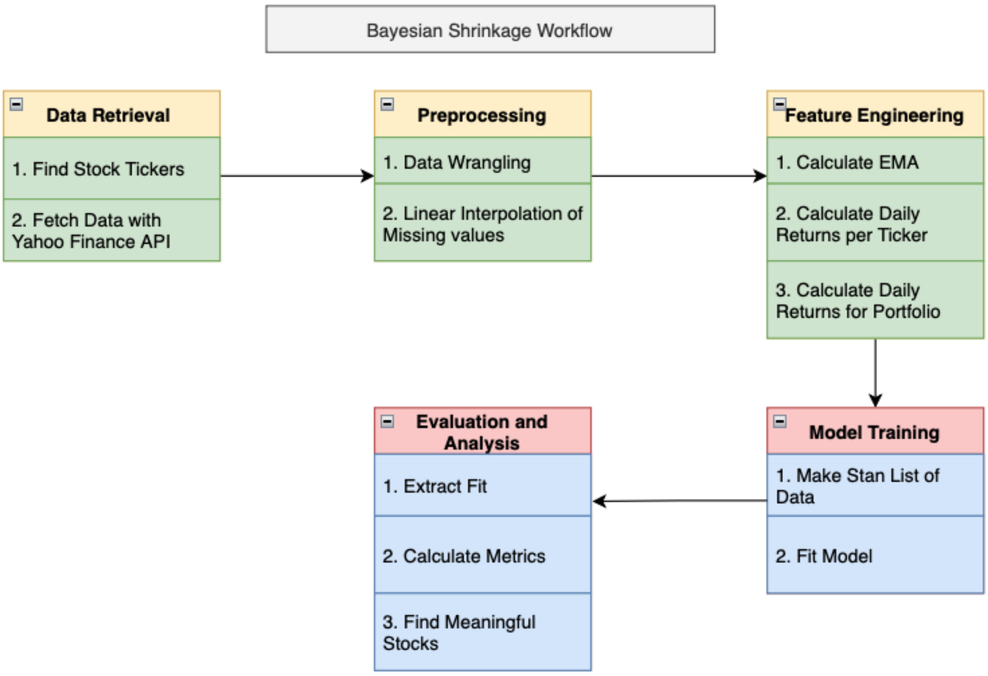

Bayesian Shrinkage Workflow
Data Retrieval
1. Find Stock Tickers
2. Fetch Data with Yahoo Finance API
Preprocessing
1. Data Wrangling
2. Linear Interpolation of Missing values
Feature Engineering
1. Calculate EMA
2. Calculate Daily Returns per Ticker
3. Calculate Daily Returns for Portfolio
Model Training
1. Make Stan List of Data
2. Fit Model
Evaluation and Analysis
1. Extract Fit
2. Calculate Metrics
3. Find Meaningful Stocks

Figure 5.2. The total workflow from gathering data to the analyzed portfolio. The segments shown in yellow and green were executed with Python, while the parts in red and blue were performed in R.

Ahead of traninning, however, the prior distributions need to be defined. For horseshoe the same formulation as presented in 4.1 is utilized—with looser shrinkage to encourage convergence—to obtain prior parameters,

$$\boldsymbol{\beta}_{prior} = \boldsymbol{\beta} \circ (\boldsymbol{\lambda} \cdot \tau), \quad \boldsymbol{\beta} \sim \mathcal{N}(0,1) \text{ and } \boldsymbol{\lambda}, \tau \sim C^{+}(0,5), \tag{5.3}$$

where $\circ$ is the Hadamard product. Similarly, for spike-and-slab the formu-

[1] https://github.com/OttoVintola/Bayes

lation presented in 4.4 is followed, albeit with a slight variation.

$$\boldsymbol{\beta}_{prior} = \boldsymbol{\beta} \circ \boldsymbol{\lambda} \cdot \mathcal{N}(0, c^2) + \boldsymbol{\beta} \circ (1 - \boldsymbol{\lambda}) \cdot \mathcal{N}(0, \epsilon^2), \quad \boldsymbol{\beta} \sim \mathcal{N}(0, 1), \tag{5.4}$$

where $c^2 = 1$, $\epsilon^2 = 0.1$, and λ_j follows an uninformative continuous Beta distribution Beta$(0, 1)$—a continuous distribution is chosen due to the faster gradient-based sampling methods in `RStan`.

The data is organized into a list, which contains the number of observations, number of regressors, the regressors, and the response variables. This is done because `RStan` accepts data in this format for model training.

The model training procedure uses MCMC sampling technique to approximate the parameter estimates $\hat{\beta}_j$. Specifically, the No-U-Turn (NUTS) sampler [23] is utilized due to its computational efficiency. The sampling is done with four parallel chains for 2000 iterations, where 500 of them are designated for the warm up. For spike-and-slab, thinning is set equal to one, since no autocorrelation between samples is assumed.

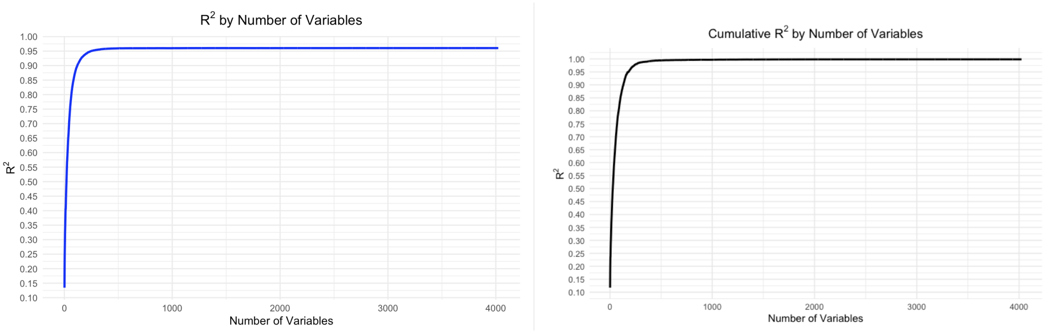
For horseshoe, however, thinning is set to two to help with convergence. Additionally, an adapt delta of 0.99 is employed to ensure robust exploration of the posterior distribution and mitigate issues with divergent transitions. Moreover, the maximum tree depth is set to 15 to allow the sampler sufficient flexibility for exploring the parameter space with the horseshoe prior.

5.4 Results

The goal is to find the few regressors that are responsible for most of the variance and explainability in the data. Therefore, the performance is evaluated using multiple metrics and graphs. The overall performance of the models is not the focus, but the relative predictive power of a few shrunk features is much more relevant for these experiments.

Currently, the parameter estimates are posterior distributions like in 3.5. So, in order to calculate performance metrics, a numeric value that best represents them needs to be calculated. For this the mean is chosen. Formally the posterior mean for shrunk parameter estimates is defined as

$$\hat{\boldsymbol{\beta}} = \int \boldsymbol{\beta} \cdot p(\boldsymbol{\beta} | \mathbf{y}, \mathbf{X}) \, d\boldsymbol{\beta}. \tag{5.5}$$



(a) Horseshoe model $\bar{R}^2$ score.

(b) Spike-and-slab model $\bar{R}^2$ score.

Figure 5.3. Comparison of $\bar{R}^2$ scores for the spike-and-slab and horseshoe models, plotted against an increasing number of variables.

5.4.1 Measuring Goodness of Fit

In regression analysis, measuring the goodness of fit is often performed with the coefficient of determination R^2 [24, p. 33]. The definition is

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}. \tag{5.6}$$

The measure can be described as the amount of total variation around the mean $\bar{y}$ explained by the predicted values $\hat{y}_i$. Consequently, the higher the value is, the better the model is at explaining the variance.

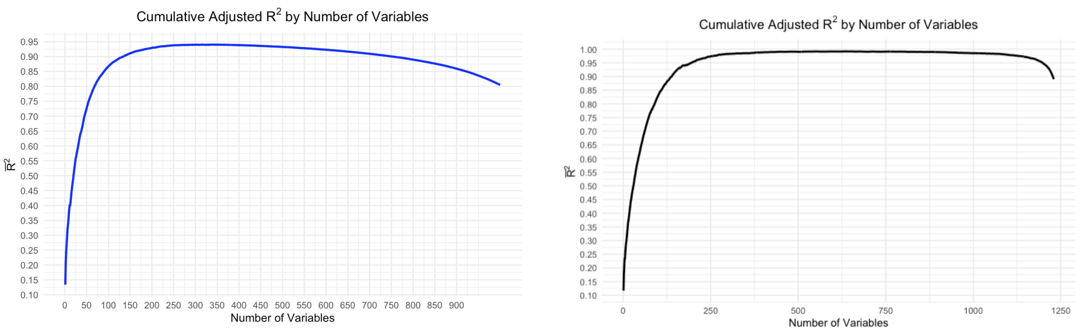
Using all of the possible shrunk variables given by the horseshoe model $R^2 \approx 0.96$, while for the spike-and-slab $R^2 \approx 0.99$. However, R^2 tends to increase as the number of regressors included in the model grows. Consequently, chasing a high R^2 would thus result in the inclusion of progressively more explanatory variables into the model. This effect can be seen in Figure 5.3 where horseshoe is colored blue and spike-and-slab as black—this theme is used throughout the section.

Typically, an adjustment is applied to counteract the increasing R^2. Adjusted R^2 is denoted as $\bar{R}^2$ [25, p. 208-211] and it penalizes adding explanatory regressors, which formulates as

$$\bar{R}^2 = 1 - (1 - R^2)\frac{n-1}{n-p-1}. \tag{5.7}$$

Plotting the cumulative $\bar{R}^2$ as the number of regressors increases reveals a more insightful picture. Figure 5.4 shows that there is a point where the inclusion of more variables decreases the $\bar{R}^2$ score.

Moreover, figure 5.4 establishes the effectiveness of spike-and-slab in finding the few important regressors. Accordingly, to maintain $\bar{R}^2 \geq 0.95$ spike-and-slab requires $p \approx 180$, while horseshoe never achieves similar



(a) Horseshoe model $\bar{R}^2$ score. **(b)** Spike-and-slab model $\bar{R}^2$ score.

Figure 5.4. Spike-and-slab and horseshoe models' $\bar{R}^2$ score plotted against an increasing number of variables.

performance. This contrast suggests that spike-and-slab is possibly more effective in identifying the significant predictors for this dataset.

5.4.2 Sparsity

Additionally, from 5.4 the number of apparently redundant regressors becomes clear. While both attempt to negate the effects of irrelevant features, horseshoe typically shrinks many coefficients towards zero but few to exactly zero, while spike-and-slab aims to eliminate the redundant ones completely. When examining Figure 5.5, it seems that horseshoe has used stricter shrinkage which results in more regressors at zero.

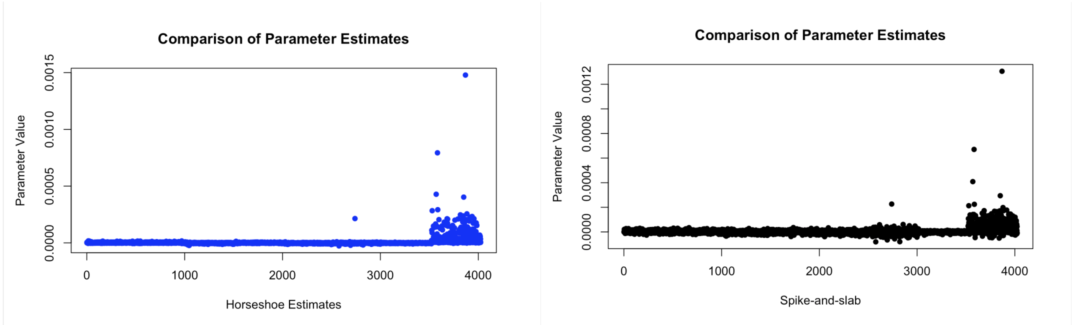


Figure 5.5. The means of the parameter values.

Additionally, examining the meaningful regressors can be performed by investigating the credible interval of model parameters' posterior distributions. Calculating the 95% credible interval $[q_l, q_u]$ and subsequently checking whether $q_l > 0$ or $q_u < 0$ allows to quantify the number of parameters that are statistically non-zero. In otherwords, the credible intervals reveal how many regressors are deemed important by the models. The procedure yields 51 non-zero regressors for horseshoe and 124 for spike-and-slab.

Visualizing only the important features reveals the important features. Figure 5.6 recreates Figure 5.5 with only the significant regressor parameters chosen via credible interval. A similar pattern persists, where horseshoe does not include parameters while spike-and-slab aims to simply select a few important ones.

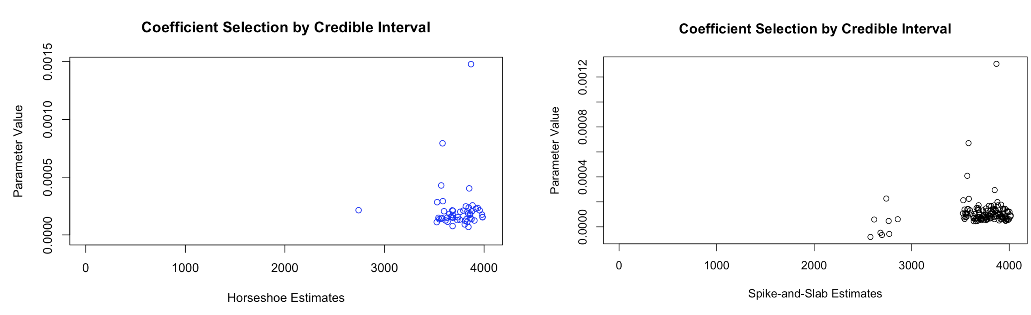
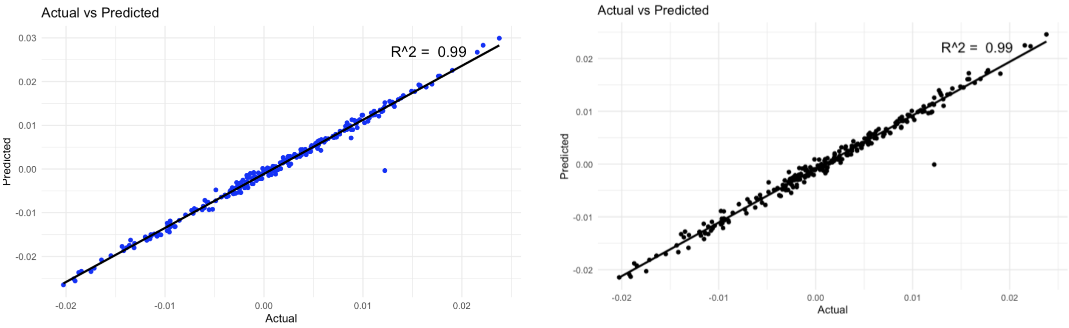


Figure 5.6. The means of the important parameter values.

Currently, there are two ways to interpret the results of the trials. One, is to use $\bar{R}^2$ as the regressor selection criteria, or the other one, is to use the regressors chosen by credible interval. However, analysing the difference between the regressors selected by them on this dataset, reveals that they select the same ones except the credible intervals are *stricter*. Consequently, the regressors selected by $\bar{R}^2$ are chosen due to the holistic predictive performance they give.

5.4.3 Predictive Performance

So far, all of the findings could be the result of overfitting. To address this concern, the performance of the models—with 180 and 450 regressors for spike-and-slab and horseshoe respectively—is evaluated on the test set discussed at the end of Section 2.4. In short, the test dataset continues where the training set ended, so the data is from 2022 to 2023. Figure 5.7 show the predicted values with horseshoe and spike-and-slab.



(a) Horseshoe model results.

(b) Spike-and-slab model results.

Figure 5.7. Comparison of predictions from the spike-and-slab and horseshoe models.

Additionally, the mean squared error (MSE) is measured on the test set.

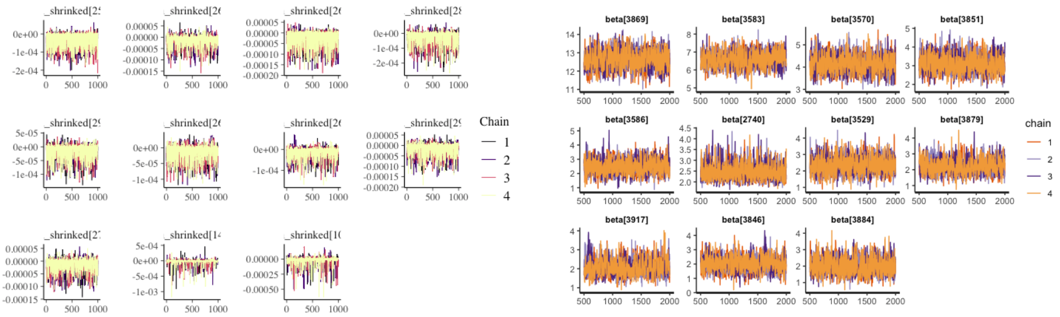
The behaviour of MSE is intuitive, which is apparent from the formula

$$\text{MSE} = \frac{1}{n}\sum_{i}^{n}(y_i - \hat{y}_i)^2.$$

MSE is chosen to measure the performance due to its intuitive formulation and excpected behavior—lower the better. On the test set the horseshoe model has MSE $\approx 6.2 \cdot 10^{-6}$, while spike-and-slab has MSE $\approx 2.1 \cdot 10^{-6}$.

Aside from overfitting, another concern in Bayeisan methods, are unrepresentative posterior distributions. In MCMC methods, ensuring the chains have mixed is crucial in order to verify the reliability of the posterior distributions. It can be done by examining the trace plots of the parameters by iteration. Figure 5.8 indicates that the chains have mixed and that the parameter space is searched efficiently.

From experimenting with the data and models, the horseshoe prior demonstrates good convergence for significant coefficients. However, non-essential ones show slower convergence or remain at zero. Overall, this observed behavior aligns with the shrinkage properties of the prior.



(a) Horseshoe traceplots.

(b) Spike-and-slab traceplots.

Figure 5.8. Figure that shows the traceplots of horseshoe and spike-and-slab respectively. For reference, chains that overlap and range between different values frequently is desirable.

Table 5.1 collects the information gathered in the trials. So far, the evidence regarding predictive power seems to be on the side of spike-and-slab, however, the credible intervals of horseshoe produce some interesting results. Consequently, the analysis of stock tickers in the data focuses on the regressors chosen by spike-and-slab.

5.5 The Connection to Stocks

Finding the important stocks is analogous to finding the important regressors. Figure 5.9 shows the most significant regressors, and it reveals a

Table 5.1. This table collects the results to compare the two priors used. MSE is measured on the test dataset of 250 observations. The column $\log(\sigma)$ describes taking $\log_{10}$ of the standard deviation. The reason for taking the base-10 logarithm is to scale the numbers for the sake of comparison. The Zero column is the number of zero variables as decided by the credible intervals. $\log(\sigma)$ describes taking $\log$ of the standard deviation of the model parameters $\hat{\beta}$, while the k/n column is the ratio of number of shrunk parameters compared to the number of observations. The columns k by CI and $\bar{R}^2$ describe the number of parameters chosen by each respective statistic.

Prior	**k by CI**	**k by $\bar{\mathrm{R}}^2$**	**Zero**	$\log(\mathbf{MSE})$	$\log(\boldsymbol{\sigma})$	$\mathbf{k/n}$
Horseshoe	51	250	3973	-5.2	0.62	0.04
Spike-and-slab	124	180	3900	-5.7	0.55	0.10

peculiar difference in the methods.

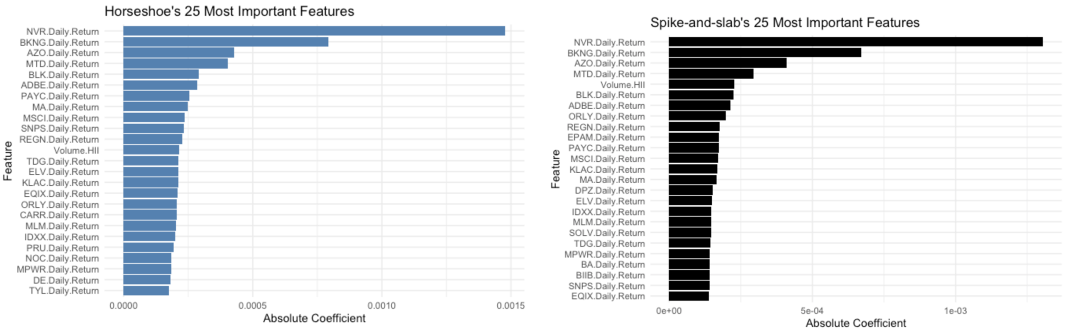


Figure 5.9. The 25 most important features according to horseshoe and spike-and-slab, respectively.

The response variable $\mathbf{y}$ was chosen to be the daily returns, described in Section 5.2, which spike-and-slab seems to deem relevant for all regressors. Similarly, horseshoe is also chooses the daily returns as the most important ones. Furthermore, the two models seem to agree, for the most part, on the most important regressors.

Currently, the regressors are with respect to the engineered or gathered features not the tickers. Prior to feature engineering and fetching the data, the thesis started with 503 tickers from the S&P500, and proposed the hypothesis that there is sparsity in the S&P500 portfolio. Consequently, the experiments were devised to find that sparsity and give weights $\mathbf{w}^*$ to the tickers.

The procedure to find $\mathbf{w}^*$ entails selecting the features that explain most of the variance in the data and their proportion, and extracting the unique tickers from which they are derived from. For $\bar{R}^2$ the top 180 regressors are chosen since it seems to be the threshold for $\bar{R}^2 \geq 0.95$, and 124 tickers are chosen based off credible interval.

Afterwards, the ticker regarding those regressors is extracted with a simple string filtering function. As a result, there are 174 unique tickers in those 180 regressors. Corresponding to those tickers there are coefficients indicating their magnitude. These coefficients are normalized in accordance to the requirements in Section 2.3 regarding portfolio optimization—weights have to sum to one and they have to be non-negative. As a result, the weights $\mathbf{w}^*$ are obtained and can be found in the appendix.

To recall, an advantage of Bayesian models is that the uncertainty can be quantified. Figure 5.10 depicts the parameter point estimates and corresponding 95% credible intervals for the parameters which were chosen to represent the weights $\mathbf{w}^*$. Each credible interval permits statistical interpretations regarding the possible inconsistencies of the estimates. Moreover, the figure does not depict any alarming credible intervals which would be considerably wider than others.

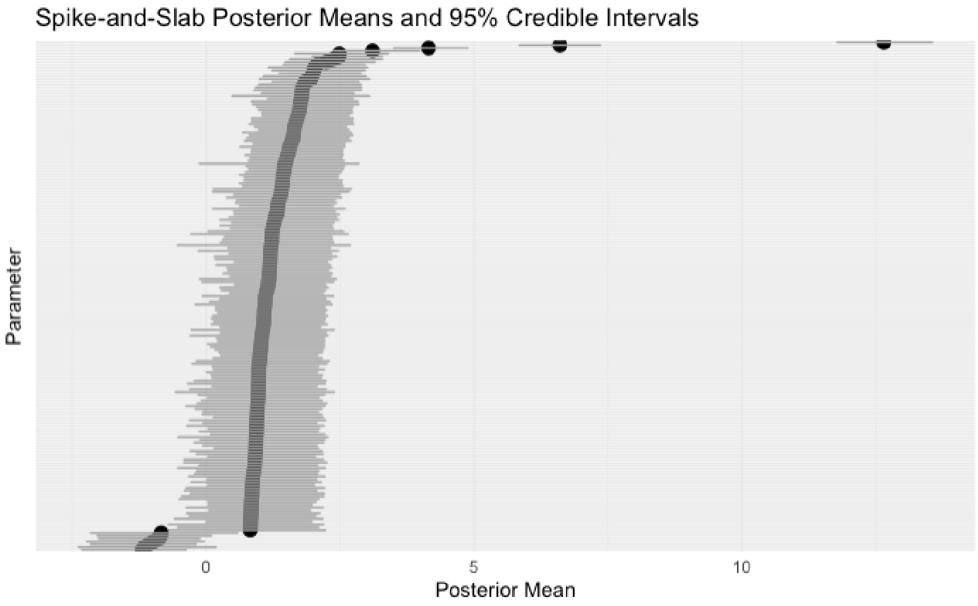


Figure 5.10. Point estimates for parameters β_i and 95% credible intervals.

6. Discussion

To consult the statistician after an experiment is finished is often merely to ask him to conduct a post mortem examination. He can perhaps say what the experiment died of.
— Sir Ronald A. Fisher

The following is a discussion regarding the findings, limits and future of the research. The results are summarized and compared to other related research, subsequently a likely misconception about the thesis is discussed, and the possible directions it could be extended to later for some interesting findings.

6.1 Findings

The notion of sparsity in portfolios has been studied previously [26–29] with non-Bayesian approaches. There have also been Bayesian studies regarding sparsity and portfolios such as [30–32], with more involved priors and separate portfolios than this thesis. However, [30] did utilize the Bayesian LASSO—with other ones as well—to optimize portfolios through sparsity. Moreover, the sparsity present in the S&P500 varies greatly over the years, which is seen in [33, Figure 6]. However, none of the previous studies have been conducted in a high-dimensional setting, which makes this thesis unique.

In [34], spike-and-slab was utilized to provide robust estimates with small errors about time-varying alphas and betas in an economic setting. So, examining the predictive power across a financial portfolio suggests the same. Training with the spike-and-slab prior distribution, offers scientific evidence for the capability of spike-and-slab to uncover sparsity with a small error rate.

Additionally, the evidence for the superiority of spike-and-slab over horseshoe as the methodology for revealing the inherent sparsity in the data is reinforced by the larger $\bar{R}^2$ score in Figure 5.4, the number of selected variables presented in 5.1, and the predictive performance shown in Figure 5.7. Conceptually, spike-and-slab is an intuitive solution. The notion of sparsity in this given context suggests that there exists a few meaningful tickers responsible for explaining most of the variance in the daily returns of the unweighted S&P500.

The amount of sparsity, however, is best highlighted by returning to the indicator of high dimensionality, namely the ratio of parameters over observations. From Figure 5.4 the optimal number of shrunk parameters while adhering to Occam's razor is atleast $k \approx 180$, although filtering regressors with credible intervals suggests 124 are suitable, but 180 are chosen to ensure predictive power. Since, the number of observations remain the same, the ratio $k/n \approx 0.14$ does not indicate high dimensionality anymore. Thus, the goal of dimensionality reduction by shrinkage seems to work.

Moreover, the goal of finding weights—introduced in Section 2—is concluded by normalizing the estimated coefficients $\hat{\beta}_i$ to be in the range of $[0, 1]$ such that $\sum_{i=1}^{k} \hat{\beta}_i = 1$ and $\hat{\beta}_i \geq 0$ for all $i = 1, 2, \ldots k$, where $k \leq p$ denotes the number of shrunk parameters. This vector of normalized estimated parameters is thus denoted $\mathbf{w}^*$ and attributed the concept of value normalized portfolio weights.

6.2 Financial Interpretation and Limit of Methods

The goal of this thesis is not predicting the value of the S&P500, because the regressors are not suited for that. Instead, the goal is to reveal the statistical sparsity present in the portfolio. Coincidentally, the predictive performance is measured on the test set which appears as predicting the price, but it is not.

As described in the feature engineering section of the thesis, other regressors such as daily returns and closing prices are utilized. Therefore, the trials indicate no evidence for a realistic model for predicting the future price as time series. It is not possible to use the values which are realized on the same day as the predicted value.

If the process would be designed with predicting the future price in mind, then all of the regressors should be masked out during inference—it would be possible to make an exception for open price, since the value is decided before the daily return of the portfolio.

Additionally, from a financial perspective, finding the most important tickers in the S&P500 and subsequently claiming them to be the most vital, could be erroneous. The actual S&P500's holdings are weighted by the market capitalization (MCAP), which is the number of shares multiplied by the price of one, of the company. So, an intuitive perspective is that the companies with the highest weight are the most influential with respect to the value.

However, comparing the MCAP of the 174 companies MCAP_{174} and all of the 503 companies MCAP_{503} in the S&P500, reveals that the weights are not deemed relevant by spike-and-slab. Since, $\text{MCAP}_{180}/\text{MCAP}_{503} \approx 0.367$, which is approximately the portion $180/500 \approx 0.36$. Thus, it can be concluded that the weighting of the true S&P500 did not affect the weights $\mathbf{w}^*$ in the aforementioned way.

6.3 Future Plans

As alluded by the previous section, a highly possible continuation of the trials would pertain to only using time and opening price to test the model. With this scheme, actual forecasting of the price of the S&P500 could be studied with a sparse regressor space. Literature in this area is limited, although forecasting the price of stocks has been studied with numerous methodologies [35–37].

Additionally, amplifying the shrinkage by decreasing ϵ^2 and increasing c^2 or alternatively setting a stricter prior for the local shrinkage parameter λ_j or a smaller value for the global shrinkage parameter τ, could be an interesting research direction. An analysis with stricter shrinkage would increase the explainability of the model discussed in Section 4.1.4.

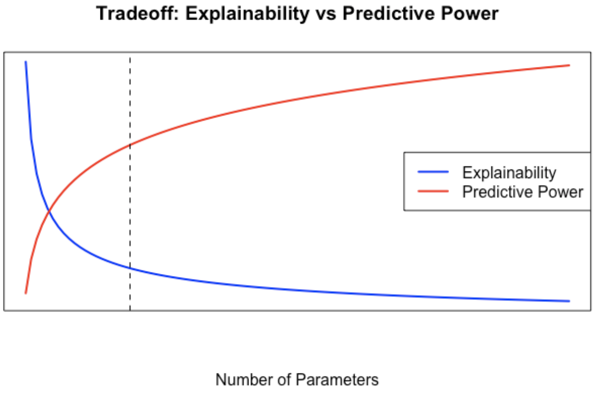


Figure 6.1. Trade-off between predictive power and model explainability. This figure does not represent true values which are calculated empirically. The picture is purely for reference and illustration. The true functions which are plotted are $1\sqrt(x)$ and $\log(x)$

With this, however, comes a trade-off in the predictive power of the model displayed in Figure 6.1. Increasing the number of parameters, decreases the model explainability. Currently, this thesis is situated at the dashed line on this graph. A possible future research direction would be to examine what happens at the intersection of the graph, and interpret the financial meaning of the findings.

7. Conclusion

If all the statisticians in the world were laid head to toe, they wouldn't be able to reach a conclusion

— Anonymous

The research question posed at the beginning of the thesis regarded the sparsity in the unweighted S&P500. The methodologies used to study were Bayesian regression with horseshoe and spike-and-slab priors for the parameters. From conducting trials on the gathered dataset, spike-and-slab proved to converge better and exhibited superior predictive power.

The main contribution of this thesis is examining the the performance of horseshoe and spike-and-slab in a financial setting, namely the unweighted S&P500. Consequently, the suggested weights $\mathbf{w}^*$ chosen by spike-and-slab for the 174 tickers are provided in the appendix.

Additionally, a side contribution of this thesis is to suggest a two-sided comparison method for evaluating the performance of shrinkage priors. Examining the $\bar{R}^2$ plots and deciding the cutoff point—for this thesis it was $\bar{R}^2 \geq 0.95$, or deciding based off the number of non-zero parameters given by the credible intervals $[q_l, q_u]$. In an ideal world, using a combination of both methods with common sense and domain knowledge is expected to provide the best results.

On top of everything, the procedure of portfolio optimization, in the scope of this thesis, hopefully raises thoughts about the possibility to find the most valuable stocks in the entire market by gathering a larger sample. That is, finding the stocks responsible for most of the variation in the entire market.

References

[1] D. Donoho, "High-dimensional data analysis: The curses and blessings of dimensionality," *AMS Math Challenges Lecture, Stanford University Department of Statistics*, pp. 1–32, 01 2000.

[2] V. Sreeram and P. Agathoklis, "On the properties of gram matrix," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 41, no. 3, pp. 234–237, 1994.

[3] X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, no. 2, 2019.

[4] R. Bellman, *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[5] H. Zou and T. Hastie, "Regularization and Variable Selection Via the Elastic Net," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 67, pp. 301–320, 03 2005.

[6] A. Y. Ng, "Feature selection, l1 vs. l2 regularization, and rotational invariance," in *Proceedings of the Twenty-First International Conference on Machine Learning*, ICML '04, (New York, NY, USA), p. 78, Association for Computing Machinery, 2004.

[7] J. Piironen, M. Paasiniemi, and A. Vehtari, "Projective inference in high-dimensional problems: Prediction and feature selection," *Electronic Journal of Statistics*, vol. 14, Jan. 2020.

[8] J. Petit, V. De Berardinis, E. Pelletier, and F. Artiguenave, "Finishing the euchromatic sequence of the human genome," *Nature*, vol. 431, no. 7011, pp. 931–945, 2004.

[9] E. Ulasan and A. Ö. Önder, "Large portfolio optimisation approaches," *Journal of Asset Management*, vol. 24, no. 6, pp. 485–497, 2023.

[10] H. Markowitz, "Portfolio selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[11] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine Series 1*, vol. 2, pp. 559–572, 1901.

[12] J. A. Nelder and R. W. M. Wedderburn, "Generalized linear models," *Journal of the Royal Statistical Society: Series A (General)*, vol. 135, no. 3, pp. 370–384, 1972.

[13] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian Data Analysis*. Boca Raton, FL: Chapman and Hall/CRC, 3rd ed., 2013. Corrected version, October 13, 2022.

[14] J. Piironen, *Bayesian Predictive Inference and Feature Selection for High-Dimensional Data*. Doctoral thesis, Aalto University, School of Science, Department of Computer Science, May 2019. Electronic archive copy available via Aalto Thesis Database.

[15] C. P. Robert, "Bayesian computational methods," 2010. arXiv:2010.1002.2702v1.

[16] J. Hadamard, "Sur les problèmes aux dérivés partielles et leur signification physique," *Princeton University Bulletin*, vol. 13, pp. 49–52, 1902.

[17] G. Borboudakis and I. Tsamardinos, "Forward-backward selection with early dropping," *Journal of Machine Learning Research*, vol. 20, pp. 1–39, 2019.

[18] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[19] G. M. Ljung and G. E. P. Box, "On a measure of a lack of fit in time series models," *Biometrika*, vol. 65, no. 2, pp. 297–303, 1978.

[20] J. Durbin and G. S. Watson, "Testing for serial correlation in least squares regression. i," *Biometrika*, vol. 37, pp. 409–428, 1950.

[21] Pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020.

[22] Stan Development Team, "RStan: the R interface to Stan," 2024. R package version 2.32.6.

[23] M. D. Homan and A. Gelman, "The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo," *J. Mach. Learn. Res.*, vol. 15, p. 1593–1623, Jan. 2014.

[24] N. R. Draper and H. Smith, *Applied Regression Analysis*. Wiley, 3rd ed., 1998.

[25] H. C. Carver, "Methods of correlation analysis by mordecai," *Journal of the American Statistical Association*, vol. 26, no. 175, pp. 350–353, 1931.

[26] J. Brodie, I. Daubechies, C. De Mol, D. Giannone, and I. Loris, "Sparse and stable markowitz portfolios," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 30, pp. 12267–12272, 2009.

[27] S. Arvanitis, O. Scaillet, and N. Topaloglou, "Sparse spanning portfolios and under-diversification with second-order stochastic dominance," 2024.

[28] J. Brodie, I. Daubechies, C. De Mol, D. Giannone, and I. Loris, "Sparse and stable markowitz portfolios," Working Paper Series 936, European Central Bank, September 2008. This paper can be downloaded without charge from `http://www.ecb.europa.eu` or the Social Science Research Network at `http://ssrn.com/abstract_id=1258442`.

[29] D. Bertsimas and R. Cory-Wright, "A scalable algorithm for sparse portfolio selection," *Sloan School of Management, Massachusetts Institute of Technology*, 2020. ORCID: 0000-0002-1985-1003 (Dimitris Bertsimas), 0000-0002-4485-0619 (Ryan Cory-Wright), Email: dbertsim@mit.edu, ryancw@mit.edu.

[30] C. Frey and W. Pohlmeier, "Bayesian shrinkage of portfolio weights," *SSRN Electronic Journal*, 2016. Available at SSRN: `https://ssrn.com/abstract=2730475` or `http://dx.doi.org/10.2139/ssrn.2730475`.

[31] Y. O. Taras Bodnar and N. Parolya, "Optimal shrinkage-based portfolio selection in high dimensions," *Journal of Business & Economic Statistics*, vol. 41, no. 1, pp. 140–156, 2023.

[32] D. Avramov and G. Zhou, "Bayesian portfolio analysis," *Working Paper*, 2009. First draft: March 2009; Current version: December 2009.

[33] T. Griveau-Billion and B. Calderhead, "Efficient structure learning with automatic sparsity selection for causal graph processes," *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 06 2019. Presented at NeurIPS 2019.

[34] M. Balcilar, R. Demirer, and F. V. Bekun, "Flexible time-varying betas in a novel mixture innovation factor model with latent threshold," *Mathematics*, vol. 9, no. 8, 2021. Cited by: 3; All Open Access, Gold Open Access.

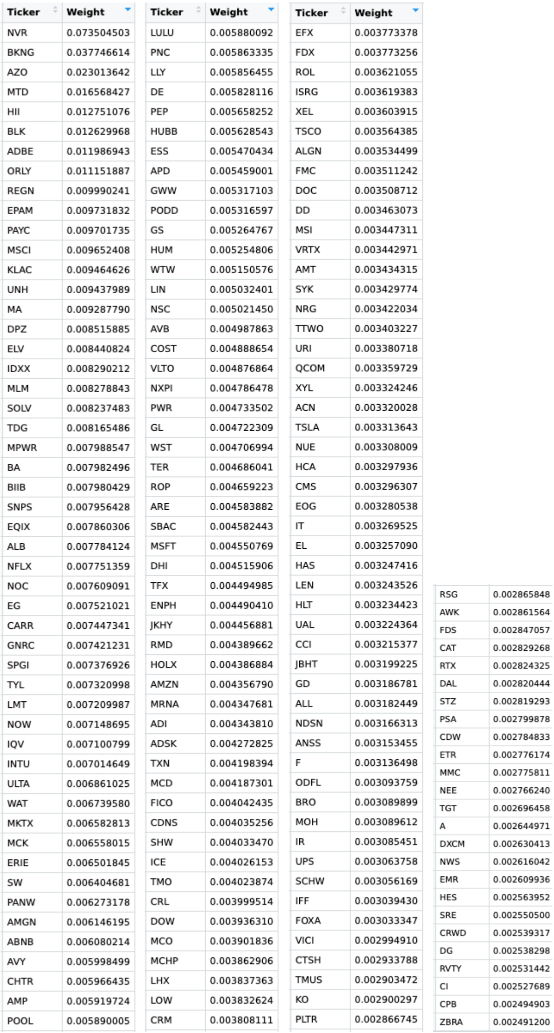
[35] A. Chatterjee, H. Bhowmick, and J. Sen, "Stock price prediction using time series, econometric, machine learning, and deep learning models," 2024.

[36] M. Vijh, D. Chandola, V. A. Tikkiwal, and A. Kumar, "Stock closing price prediction using machine learning techniques," *Procedia Computer Science*, vol. 167, pp. 599–606, 2020. International Conference on Computational Intelligence and Data Science.

[37] X. Li, Y. Li, H. Yang, L. Yang, and X.-Y. Liu, "Dp-lstm: Differential privacy-inspired lstm for stock prediction using financial news," December 2019. Published on arXiv, 20 Dec 2019.

A. Appendix

A.1 Weights



Ticker	Weight
NVR	0.073504503
BKNG	0.037746614
AZO	0.023013642
MTD	0.016568427
HII	0.012751076
BLK	0.012629968
ADBE	0.011986943
ORLY	0.011151887
REGN	0.009990241
EPAM	0.009731832
PAYC	0.009701735
MSCI	0.009652408
KLAC	0.009464626
UNH	0.009437989
MA	0.009287790
DPZ	0.008515885
ELV	0.008440824
IDXX	0.008290212
MLM	0.008278843
SOLV	0.008237483
TDG	0.008165486
MPWR	0.007988547
BA	0.007982496
BIIB	0.007980429
SNPS	0.007956428
EQIX	0.007860306
ALB	0.007784124
NFLX	0.007751359
NOC	0.007609091
EG	0.007521021
CARR	0.007447341
GNRC	0.007421231
SPGI	0.007376926
TYL	0.007320998
LMT	0.007209987
NOW	0.007148695
IQV	0.007100799
INTU	0.007014649
ULTA	0.006861025
WAT	0.006739580
MKTX	0.006582813
MCK	0.006558015
ERIE	0.006501845
SW	0.006404681
PANW	0.006273178
AMGN	0.006146195
ABNB	0.006080214
AVY	0.005998499
CHTR	0.005966435
AMP	0.005919724
POOL	0.005890005
LULU	0.005880092
PNC	0.005863335
LLY	0.005856455
DE	0.005828116
PEP	0.005658252
HUBB	0.005628543
ESS	0.005470434
APD	0.005459001
GWW	0.005317103
PODD	0.005316597
GS	0.005264767
HUM	0.005254806
WTW	0.005150576
LIN	0.005032401
NSC	0.005021450
AVB	0.004987863
COST	0.004888654
VLTO	0.004876864
NXPI	0.004786478
PWR	0.004733502
GL	0.004722309
WST	0.004706994
TER	0.004686041
ROP	0.004659223
ARE	0.004583882
SBAC	0.004582443
MSFT	0.004550769
DHI	0.004515906
TFX	0.004494985
ENPH	0.004490410
JKHY	0.004456881
RMD	0.004389662
HOLX	0.004386884
AMZN	0.004356790
MRNA	0.004347681
ADI	0.004343810
ADSK	0.004272825
TXN	0.004198394
MCD	0.004187301
FICO	0.004042435
CDNS	0.004035256
SHW	0.004033470
ICE	0.004026153
TMO	0.004023874
CRL	0.003999514
DOW	0.003936310
MCO	0.003901836
MCHP	0.003862906
LHX	0.003837363
LOW	0.003832624
CRM	0.003808111
EFX	0.003773378
FDX	0.003773256
ROL	0.003621055
ISRG	0.003619383
XEL	0.003603915
TSCO	0.003564385
ALGN	0.003534499
FMC	0.003511242
DOC	0.003508712
DD	0.003463073
MSI	0.003447311
VRTX	0.003442971
AMT	0.003434315
SYK	0.003429774
NRG	0.003422034
TTWO	0.003403227
URI	0.003380718
QCOM	0.003359729
XYL	0.003324246
ACN	0.003320028
TSLA	0.003313643
NUE	0.003308009
HCA	0.003297936
CMS	0.003296307
EOG	0.003280538
IT	0.003269525
EL	0.003257090
HAS	0.003247416
LEN	0.003243526
HLT	0.003234423
UAL	0.003224364
CCI	0.003215377
JBHT	0.003199225
GD	0.003186781
ALL	0.003182449
NDSN	0.003166313
ANSS	0.003153455
F	0.003136498
ODFL	0.003093759
BRO	0.003089899
MOH	0.003089612
IR	0.003085451
UPS	0.003063758
SCHW	0.003056169
IFF	0.003039430
FOXA	0.003033347
VICI	0.002994910
CTSH	0.002933788
TMUS	0.002903472
KO	0.002900297
PLTR	0.002866745
RSG	0.002865848
AWK	0.002861564
FDS	0.002847057
CAT	0.002829268
RTX	0.002824325
DAL	0.002820444
STZ	0.002819293
PSA	0.002799878
CDW	0.002784833
ETR	0.002776174
MMC	0.002775811
NEE	0.002766240
TGT	0.002696458
A	0.002644971
DXCM	0.002630413
NWS	0.002616042
EMR	0.002609936
HES	0.002563952
SRE	0.002550500
CRWD	0.002539317
DG	0.002538298
RVTY	0.002531442
CI	0.002527689
CPB	0.002494903
ZBRA	0.002491200

Figure 1.1. Weights by tickers.