UNIVERSITETET I BERGEN

Det matematisk-naturvitenskapelige fakultet

A Dimensionality Reducing Extension of Bayesian Relevance Learning

Master of Science

Financial Theory and Insurance Mathematics

Sandra Heimsæter

March 2, 2021

Acknowledgements

I would like to thank my supervisor Yushu Li, and my secondary supervisor Ingvild M. Helgøy for their guidance, scientific discussions and valuable feedback on my work. I want to thank them for having faith in my ability to do this research, and for introducing me to the topic. Thanks to both, I have grown and developed a lot.

Thank you to the mathematical institute at the University of Bergen for giving me this opportunity, and for being helpful and cooperative.

I would also like to thank Chang Li, one of the researchers behind *"Probabilistic Feature Selection and Classification Vector Machine"* (2019), for helping me understand his research and for sharing the code associated with their work.

To all my beloved friends and family, thank you for being patient, caring and supportive, and not least for bringing me a lot of fun and laughter. I especially want to thank my partner for being exceptionally patient and supportive through my worst days, for good help with the programming, and for proofreading my thesis.

Sandra Heimsæter,
March 2, 2021

Abstract

When modeling with big data and high dimensional data, the ability to extract the most important information from the data set and avoid overfitting is crucial. However, by using well developed sparse methods, we can construct models that are less likely to overfit as they use only the most informative part of the data. In this thesis, we are developing an algorithm which can simultaneously achieve sample and feature selection when facing big data in supervised learning. This parametric Bayesian regression learning method is based on a well known Bayesian sparse learning method: the Relevance Vector Machine (RVM). The deduction of the algorithm is inspired by, the probabilistic feature selection and classification vector machine (PFCVM), which is a simultaneous sample and feature selective extension of the RVM classification model. Our resulting method is called the dimensionality reducing relevance vector machine (DRVM), and it performs simultaneous feature and sample selection in the regression case. The proposed model is sparse in terms of choosing only the most important features and samples to explain the input data, as well as being accurate in predictions.

Keywords Big Data · Dimensionality Reduction · High Dimensional Data · Kernel basis function · Probabilistic Prediction · Sparse Bayesian Learning

Table of Contents

List of Algorithms i

List of Figures iii

List of Tables v

List of Symbols vii

Notations ix

Mathematical Formulas xi

1 Introduction 1

1.1 Background and Previous Research 1

1.2 Sparse Modeling 2

1.2.1 Sparse Sample Selective Framework 3

1.2.2 Bayesian Modeling 5

2 The Relevance Vector Machine 9

2.1 Sparse Sample Selective Framework 9

2.2 Calculating Posteriors 10

2.3 Optimization of the Parameters 12

2.3.1 Parameter Learning Using Maximum Likelihood and Cyclically Re-Estimation 13

2.3.2 Fast Type-II Maximum Likelihood Optimization 14

2.4 Making Predictions 16

2.5 The Relevance Vector Algorithm 17

2.5.1 Update Formulas for Effective Estimation 19

2.6 The Relevance Vector Classification Machine 22

2.6.1 Framework of RVM Classification 22
2.6.2 Laplace's Approximation 23
2.6.3 Iterative Reweighted Least Squares 27
2.6.4 Calculating Posteriors in RVM for Classification 28
2.6.5 Parameter Learning in RVM for Classification 29
2.6.6 The Predictive Distribution 30

3 Probabilistic Feature Selection and Classification Vector Machine 33
3.1 Probabilistic Classification Vector Machines 33
3.2 Sparse Sample and Feature Selective Framework 34
3.3 Calculating Posteriors 36
3.4 Parameter Learning 40
3.5 The Predictive Distribution and the Algorithm of PFCVM 41

4 Dimensionality Reducing Relevance Vector Machine for Regression 45
4.1 Sparse Sample and Feature Selective Framework 45
4.2 Calculating Posteriors 46
4.3 Parameter Learning in DRVM 48
4.4 Algorithm of the Sample and Feature Selective Relevance Vector Based Model 49
4.5 Making Predictions 49

5 Experimental Results 51
5.1 Examples on One Dimensional Synthetic Input Data 51
5.2 Comparisons on Benchmark Data Sets 53

6 Discussion 57

References 58

A Appendix 63
A.1 Possible Feature Selective Extension of Fast Bayesian Learning 63
A.1.1 Extension of the Fast Relevance Vector Machine 63
A.1.2 Extension of the Noise-Robust Fast Sparse Bayesian Learning Model 65
A.1.3 Challenges with Establishing the Algorithms 74
A.2 Code Snippets from the DRVM Learning 74
A.3 Explanation of the Boston Housing Features 77

List of Algorithms

1 Relevance Vector Machine (RVM) 17

2 Fast Relevance Vector Machine (FRVM) 18

3 Probabilistic Feature Selection and Classification Vector Machine (PFCVM) 42

4 Dimensionality Reducing Relevance Vector Machine (DRVM) 49

5 Noise-Robust Fast Sparse Bayesian Learning Model (BLS) 70

List of Figures

2.1 Step function approximation 24

3.1 Sigmoid approximation 37

5.1 Sinc without noise 52
5.2 Sinc without noise 53
5.3 Friedman #1 feature frequency and size 54
5.4 Feature frequency Diabetes 55
5.5 Feature frequency Boston 56

List of Tables

5.1 Comparison of the average number of relevance vectors (nRV), relevance features (nRF) and RMSE for the different data sets and methods. 53

List of Symbols

$\boldsymbol{\Phi}$	Kernel basis function matrix	(0.1)
$\boldsymbol{w}$	Vector of sample weights	
$\boldsymbol{\phi}(\boldsymbol{x}_i)$	i'th row of the kernel basis function matrix	
$\boldsymbol{\phi}_j(\boldsymbol{x})$	j'th column of the kernel basis function matrix	
$\boldsymbol{t}$	Vector of targets corresponding to the input vector	(0.2)
$\boldsymbol{x}$	Input vector	(0.3)
$\boldsymbol{\alpha}$	Vector of hyperparameters corresponding to $\boldsymbol{w}$	
$\boldsymbol{A}$	Diagonal matrix of hyperparameters corresponding to $\boldsymbol{w}$ in RVM	
$\boldsymbol{\Sigma}$	Covariance matrix of the posterior distributions	
$\boldsymbol{\mu}$	Mean vector of the posterior distributions	
$\boldsymbol{C}$	Covariance matrix of $p(\boldsymbol{t}\|\boldsymbol{\alpha}, \sigma^2)$ in RVM and FRVM	(2.12)
y_*	The prediction of the distributions	
σ_*^2	The uncertainty in the predictions	
$\boldsymbol{\vartheta}$	Vector of feature weights	(3.2)
$\boldsymbol{\beta}$	Vector of hyperparameters corresponding to $\boldsymbol{\vartheta}$	
$\boldsymbol{B}$	Diagonal matrix of hyperparameters corresponding to $\boldsymbol{\vartheta}$	

Notations

We are using N to denote the total number of observations in the data set, and P to denote the total number of different predictors, or variables, for each observation. Further, we are denoting vectors with bold lower case letters, and matrices with bold capitals. The bold matrix $\boldsymbol{\Phi}$ of kernel basis functions $K()$, with one additional row of ones corresponding to the weight w_0, is of dimension $N \times (N+1)$ and has the form

$$\boldsymbol{\Phi} = \begin{pmatrix} 1 & K(\boldsymbol{x}_1, \boldsymbol{x}_1) & K(\boldsymbol{x}_2, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_N, \boldsymbol{x}_1) \\ 1 & K(\boldsymbol{x}_1, \boldsymbol{x}_2) & K(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & K(\boldsymbol{x}_N, \boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & K(\boldsymbol{x}_1, \boldsymbol{x}_N) & K(\boldsymbol{x}_2, \boldsymbol{x}_N) & \cdots & K(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{pmatrix}. \tag{0.1}$$

The vector of unknown weight parameters $\boldsymbol{w}$ is given by

$$\boldsymbol{w} = (w_0, w_1, \cdots, w_N)^\top,$$

where each weight w_i corresponds to vector number i of kernel basis functions, that is the i'th row of the kernel basis function matrix $\boldsymbol{\Phi}$ from (0.1), that is

$$\boldsymbol{\phi}(\boldsymbol{x}_i) = \big(1, K(\boldsymbol{x}_1, \boldsymbol{x}_i), K(\boldsymbol{x}_2, \boldsymbol{x}_i), \cdots, K(\boldsymbol{x}_N, \boldsymbol{x}_i)\big).$$

The columns of the kernel matrix $\boldsymbol{\Phi}$ in (0.1) will further be denoted by $\boldsymbol{\phi}_j(\boldsymbol{x})$, and has the form:

$$\boldsymbol{\phi}_j(\boldsymbol{x}) = \big(K(\boldsymbol{x}_j, \boldsymbol{x}_1), K(\boldsymbol{x}_j, \boldsymbol{x}_2), \cdots, K(\boldsymbol{x}_j, \boldsymbol{x}_N)\big)^\top,$$

for j in $[1, N]$. The kernel function at position (i, j) is then

$$\Phi_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

for j in $[1, N]$. To avoid confusion around the indexing and the first column of the kernel basis function matrix $\boldsymbol{\Phi}$ we will use $j = 0$ to denote this first column of ones. Thus, the corresponding column and functions are:

$$\boldsymbol{\phi}_0(\boldsymbol{x}) = \big(1, \ldots, 1\big),$$

$$\Phi_{i,0} = 1.$$

We are going to use the bold capital $\boldsymbol{I}$ to indicate the identity matrix, that is

$$\boldsymbol{I} = \mathrm{diag}(1, 1, \cdots, 1),$$

and a bold $\mathbf{1} = (1, 1, \cdots, 1)$ to denote a vector of ones. Further the index $^\top$ will consistently be used to denote the transpose of a vector or a matrix. By a bold lower case $\boldsymbol{t}$ denoting the vector of observed response variables or targets:

$$\boldsymbol{t} = (t_1, t_2, \cdots, t_N)^\top, \tag{0.2}$$

and with

$$\boldsymbol{x}_i = (x_{i1}, x_{i2}, \cdots, x_{iP}) \tag{0.3}$$

being the input vector corresponding to the output t_i, the observed data are given by the data points

$$\big\{(\boldsymbol{x}_1, t_1), (\boldsymbol{x}_2, t_2), \cdots, (\boldsymbol{x}_N, t_N)\big\}. \tag{0.4}$$

When the index $_{MP}$ is used, it is referring to the most probable values of the given parameter.

Mathematical Formulas

This section covers mathematical formulas and results that will be used several times later in the thesis.

Woodbury matrix identity.

The inverse of a rank-k matrix can be simplified by rewriting it as (Higham, 2002, p. 258)

$$\big(\boldsymbol{A}+\boldsymbol{UCV}\big)^{-1}=\boldsymbol{A}^{-1}-\boldsymbol{A}^{-1}\boldsymbol{U}\big(\boldsymbol{C}^{-1}+\boldsymbol{VA}^{-1}\boldsymbol{U}\big)^{-1}\boldsymbol{VA}^{-1}, \tag{0.5}$$

for any matrices $\boldsymbol{A}$, $\boldsymbol{U}$, $\boldsymbol{C}$ and $\boldsymbol{V}$ of the right sizes. More specifically, A must be $n \times n$, U is $n \times k$, C is $k \times k$ and V is $k \times n$.

Determinant identity.

The determinant of a matrix equation of the given form can be rewritten using the identity (Magnus and Neudecker, 2019, p. 201)

$$|\boldsymbol{X}+\boldsymbol{AB}|=|\boldsymbol{X}||\boldsymbol{I}+\boldsymbol{BX}^{-1}\boldsymbol{A}|, \tag{0.6}$$

for any matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{X}$, and the identity matrix $\boldsymbol{I}$, of the right sizes.

Jacobi's formula.

The Jacobi's formula gives the derivative of a matrix determinant in terms of its adjugate and its trace, that is (Magnus and Neudecker, 2019, p. 201)

$$\begin{aligned}\frac{d}{dt}|\boldsymbol{A}(t)| &= trace\left[adjugate\big(\boldsymbol{A}(t)\big)\,\frac{d\boldsymbol{A}(t)}{dt}\right]\\ &= |\boldsymbol{A}(t)|\,trace\left[\boldsymbol{A}^{-1}(t)\,\frac{d\boldsymbol{A}(t)}{dt}\right].\end{aligned} \tag{0.7}$$

Inverse of 2×2 Block Matrices.

Let $\boldsymbol{R}$ be a 2×2 block matrix given by

$$\boldsymbol{R}=\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}\\ \boldsymbol{C} & \boldsymbol{D}\end{bmatrix},$$

where $\boldsymbol{A}$ is a $k \times m$ nonsingular matrix, $\boldsymbol{B}, \boldsymbol{C}$ and $\boldsymbol{D}$ are, respectively, $k \times n$, $l \times m$ and $l \times n$ matrices. In addition, the matrix $\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}$ must be invertible. In that case the inverse $\boldsymbol{R}^{-1}$ is given by (Lu and Shiou, 2002, p. 120)

$$\boldsymbol{R}^{-1} = \begin{bmatrix} \boldsymbol{A}^{-1} + \boldsymbol{A}^{-1}\boldsymbol{B}\big(\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}\big)^{-1}\boldsymbol{C}\boldsymbol{A}^{-1} & -\boldsymbol{A}^{-1}\boldsymbol{B}\big(\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}\big)^{-1} \\ -\big(\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}\big)^{-1}\boldsymbol{C}\boldsymbol{A}^{-1} & \big(\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}\big)^{-1} \end{bmatrix}. \tag{0.8}$$

The fundamental property of the Dirac delta function.

A Dirac delta function $\delta()$ has the fundamental property that (Oldham et al., 2010)

$$\int_{-\infty}^{\infty} f(x)\delta(x-a)dx = f(a). \tag{0.9}$$

1 | Introduction

1.1 Background and Previous Research

Today, companies and other institutions are collecting enormous amounts of data, and nothing suggests that this trend will slow down. Thus, the need to extract the most important information from vast amounts of data has never been greater. This thesis will concentrate on Sparse Bayesian supervised learning in analysis of big data. When talking about big data, we are in this thesis referring to two specific situations: data that contains high dimensional input variables, and data with large sample size. When facing so called big data, model constructing by standard methods using the entire data set can be time consuming and computationally expensive. In such situations we want to construct models that can extract the most informative part of the data, and at the same time achieve high predictive ability. Learning algorithms not using all the data in prediction, can be called sparse learning, and they can be sparse in terms of variable selection and in terms of sample size reduction. A well known example of sparse learning is the Support Vector Machine (SVM) (Platt et al., 1999), which aims to select the most important samples to affect the predictions. However, the SVM is a fully deterministic machine learning method, and it is limited to the use of kernel functions that follows the Mercer's condition (Smola et al., 1998). To overcome these limitations, Tipping (2001) suggested a sparse Bayesian, and hence probabilistic, approach to the SVM, called the Relevance Vector Machine (RVM). This method was using remarkably fewer basis function than the SVM method while it also had several advantages, including the ability to give probabilistic predictions, automatically estimate the nuisance parameters, and it was also able to use arbitrary basis functions (Tipping, 2001). Still, the method suffered from being slow in the learning procedure and Tipping et al. (2003) followed up with a faster optimization algorithm for the model, reffered to as the Fast Relevance Vector Machine (FRVM). These original RVM methods are sparse in terms of sample size and can be extended to achieve sparsity in high dimensional data. Our paper will develop a method which can achieve simultaneous sparsity in both sample and feature size. The resulting model is called the Dimensionality Reducing Relevance Vector Machine (DRVM) and is a feature selective extension of the original RVM in the regression case. The method is inspired by a similar simultaneous feature and sample selective extension of the classification case of RVM, developed by Jiang et al. (2019) which is called the Probabilistic Feature Selection and Classification Vector Machine (PFCVM). We will show that our method can more accurately compared to the original RVM when data are multidimensional, as it is more robust towards the noise variance than models using the entire data set.

We will in this chapter explain the sparse Bayesian framework in detail. Then, in Chapter 2 we will look into the RVM and FRVM model by Tipping (2001) and Tipping et al. (2003), before we are going to investigate the extension to the PFCVM model by Jiang et al. (2019) in Chapter 3. In Chapter 4 we will develop the dimensionality reducing extension in the regression case called DRVM. Further, we will do some simulational experiments on the performance of the proposed DRVM model in Chapter 5, to see if the model is choosing the parameters that for sure is affecting the model. Lastly, in Chapter 6, we will discuss our findings in the research and potential further research topics.

1.2 Sparse Modeling

When the sample size N in a dataset is too large, we can expect algorithms that are using all the data to be slow and computationally expensive. Sparse methods will often handle data with large sample size by choosing only the most important observations to affect in the prediction, instead of using the whole original data set, and hence make the processing less expensive. By using methods that are sparse in sample size, we can overcome this problem, or at least make the models run faster and be less expensive in the computations.

We can also use the term big data when data is high dimensional, meaning that the number of input variables P is large compared to the number of observations N. As postulated in the introductory part, modeling big data or high dimensional data with simple methods using all the data, has several possible limitations. First, if the data are sufficiently high dimensional, we can experience what is called the curse of dimensionality (Bellman and Dreyfus, 1957), that is when the number of variables increases the number of observations needed to avoid serious bias problem is increasing even more. Therefore, the number of observations in the data at hand is often not sufficiently large when the number of variables is large. In addition, if we are modeling with all variables, we can experience overfitting and a model that is too complex and captures the random noise in the data. To reduce or avoid these problems, we have to fit models that are performing variable selection or dimensionality reduction. Such models aim to choose only the most important features to affect the predicted output variable. Thus, using sparse methods, can result in more parsimonious models with better generalization capacities.

In the next section, we will look into the sparse framework of the RVM models that is sparse in terms of sample size reduction, while we will investigate the sparse framework for RVM based models being sparse both in terms of feature selection and sample size reduction in Section 3.2.

1.2.1 Sparse Sample Selective Framework

This thesis is an investigation within the framework of sparse supervised machine learning, that aims at capturing the systematic information in the training data $\{\boldsymbol{x}_i, t_i\}_{i=1}^N$ given by Equation (0.2):(0.4), with the purpose of making accurate predictions for future values t_*. This is frequently done by modeling the dependency between input vectors $\{\boldsymbol{x}_i\}_{i=1}^N$ and the corresponding outputs $\{t_i\}_{i=1}^N$, by defining a function $y(\boldsymbol{x}_i)$ given by M basis functions:

$$y(\boldsymbol{x}_i) = w_0 + \sum_{l=1}^{M} w_l \phi_l(\boldsymbol{x}_i) = \boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w}. \tag{1.1}$$

In Equation (1.1), $\boldsymbol{w}$ is the vector of unknown weight parameters to be estimated, and in general supervised learning the basis function $\boldsymbol{\phi}(\boldsymbol{x}_i)$ is a vector corresponding to the input vector $\boldsymbol{x}_i$, given by

$$\boldsymbol{\phi}(\boldsymbol{x}_i) = \big(1, \phi_1(\boldsymbol{x}_i), \phi_2(\boldsymbol{x}_i), \cdots, \phi_M(\boldsymbol{x}_i)\big).$$

However, in most cases of sparse learning, these basis functions $\boldsymbol{\phi}(\boldsymbol{x}_i)$ are given by the kernel or covariance functions $K(\boldsymbol{x}, \boldsymbol{x}_i)$, that measures similarity between $\boldsymbol{x}_i$ and the other input vectors $\boldsymbol{x}$. That is

$$\begin{aligned} \boldsymbol{\phi}(\boldsymbol{x}_i) &= \Big(1, K(\boldsymbol{x}, \boldsymbol{x}_i)\Big) \\ &= \Big(1, \big(K(\boldsymbol{x}_1, \boldsymbol{x}_i), K(\boldsymbol{x}_2, \boldsymbol{x}_i), \cdots, K(\boldsymbol{x}_N, \boldsymbol{x}_i)\big)\Big), \end{aligned} \tag{1.2}$$

where we can see that the number of elements in the basis function $\boldsymbol{\phi}(\boldsymbol{x}_i)$ must be $(N+1)$, ant that we need to have $M = N$, which often is the case in sparse learning. The most common kernel, and the one we will be using, is the Gaussian, also called a Radial Basis Function (RBF). For i and j in $[1, N]$ the RBF kernel function is given by

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\Big\{-\vartheta ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2\Big\}, \tag{1.3}$$

where ϑ is a non-negative free parameter (Vert et al., 2004, p. 63). By the model constructed above, the output is a linear combination of N, usually not linear basis functions, which makes the output linear in the parameters $\boldsymbol{w}$. This makes the model function in Equation (1.1) relatively simple to work with. All summed up, our models will make predictions based on:

$$\begin{aligned}\boldsymbol{y}(\boldsymbol{x};\boldsymbol{w}) &= \sum_{l=0}^{N} w_l \boldsymbol{\phi}_l(\boldsymbol{x}) \\ &= w_o + \sum_{l=1}^{N} w_i K(\boldsymbol{x}, \boldsymbol{x}_l) \\ &= \boldsymbol{\Phi w}. \end{aligned} \tag{1.4}$$

From Equation (1.4), with the preferred kernel function, the modeling problem generally is to estimate $\boldsymbol{w}$ as good as possible using the relevant known data. Thus, we can predict for new unseen target values t_*, while the new input values are not yet known.

When estimating the weight parameters $\boldsymbol{w}$ in Equation (1.1) we are assuming that the targets t_i can be expressed by the true model $y(\boldsymbol{x}_i)$ with an additional random noise ϵ_i, that is

$$t_i = y(\boldsymbol{x}_i) + \epsilon_i.$$

The ϵ_i's are Gaussian zero-mean with variance σ^2, such that

$$\boldsymbol{t}|\boldsymbol{x}, \boldsymbol{w}, \sigma^2 \sim \mathcal{N}\big(\boldsymbol{t}|\boldsymbol{\Phi w}, \sigma^2\big),$$

and hence the likelihood of the targets $\boldsymbol{t}$ is given by

$$p\big(\boldsymbol{t}|\boldsymbol{w}, \sigma^2\big) = \big(2\pi\sigma^2\big)^{-\frac{N}{2}} \exp\Big\{ -\frac{1}{2\sigma^2}||\boldsymbol{t} - \boldsymbol{\Phi w}||^2 \Big\}. \tag{1.5}$$

This presence of noise makes the key challenge of the modeling to avoid overfitting, while still capturing the systematic information in the data (Tipping, 2001). When fitting Equation (1.4) using methods that is sparse in sample size, some of the estimated weight values will be zero. In that way the model is not using all the N observations in the data but is rather choosing the most important ones when it comes to prediction. An efficient way to do this is by setting the weights that corresponds to the least influential basis functions to zero, which also is controlling the complexity in the model and makes overfitting less likely. If we model (1.4) using a method that performs variable selection, or dimensionality reduction, the fitting procedure will hopefully choose only the most explanatory features in the data. The method we are developing is sparse both in feature and sample size, and will probably be selective in terms of choosing only the most important observations as well as features to affect the model.

1.2.2 Bayesian Modeling

If we try to estimate the parameters $\boldsymbol{w}$ and make predictions using all observations, by (1.5), we can expect the model to be computationally expensive. If the data are high dimensional, the risk of overfitting is high. A common way to reduce or avoid these problems is to use a Bayesian framework, and place sparse priors on the weight parameters $\boldsymbol{w}$. Frequentist modeling handles uncertainty in the data in terms of noise and errors, but from a Bayesian point of view, we would in addition aim to capture the uncertainty in the models, and in the corresponding parameters. This is achieved by using prior intuitions and treating parameters like random variables with their own distributions. In that way we can learn more about the uncertainty in the predictions. All the methods considered in this thesis are based on such a Bayesian framework, which makes the models sparse and probabilistic. In the frequentist case we would have assumed a vector of true, unknown deterministic parameters $\boldsymbol{\Omega}$ to exist, and try to estimate them as good as possible based on certain criteria. Using a Bayesian approach, we would not make the assumption of a single true $\boldsymbol{\Omega}$, but rather try to find a distribution of the parameters (Tipping et al., 2003).

The likelihood of observing the current data is defined as the probability $p(\boldsymbol{t}|\boldsymbol{\Omega})$, where $\boldsymbol{\Omega}$ is the parameters we want to estimate. We will also specify a prior distribution for the parameters, which represents our thoughts or expectations about the data before anything is observed. It is denoted $p(\boldsymbol{\Omega})$. We can now use Bayes theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)},$$

where A and B are random variables, to find the posterior distribution over the parameters. This is given by

$$p(\boldsymbol{\Omega}|\boldsymbol{t}) = \frac{p(\boldsymbol{t}|\boldsymbol{\Omega})p(\boldsymbol{\Omega})}{p(\boldsymbol{t})}, \tag{1.6}$$

which can be interpreted as

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}},$$

and represents our beliefs about the data after collecting it. Using the framework established, we can now predict for new data points $\boldsymbol{t}_*$ using the predictive distribution

$$p(\boldsymbol{t}_*|\boldsymbol{t}) = \int p(\boldsymbol{t}_*|\boldsymbol{\Omega})p(\boldsymbol{\Omega}|\boldsymbol{t})\, d\boldsymbol{\Omega},$$

given by the law of total probability (Tipping et al., 2003). As we are integrating out the parameters $\boldsymbol{\Omega}$, this predictive distribution is determined purely by the observed data $\boldsymbol{t}$, and no further information is needed in the Bayesian framework. In addition, a Bayesian approach will estimate nuisance parameters, and is able to quantify uncertainty in the predictions. The most important advantage of Bayesian learning is, in our setting, the ability to extract a full posterior distribution instead of just returning a most probable point estimate as a fully deterministic approach.

Brief Consideration of Sparseness in the Priors

Using the Bayesian framework above, we are able to train models with a great amount of sparseness by using sparse priors as pre-assumption for the parameters. In this paper, we will use a zero mean Gaussian prior on each weight w_i given the hyperparameters α_i, that is

$$w_i|\alpha_i \sim \mathcal{N}(w_i|0, \alpha_i),$$

with a Gamma(a, b) hyperprior on α_i. Now, we are going to show that this kind of prior is sparse as it gives a marginal Student-t distribution (Tipping, 2001). With this hierarchical prior, and by integrating out the α_i's Tipping (2001) got:

$$\begin{aligned} p(w_i) &= \int p(w_i|\alpha_i)p(\alpha_i)\, d\alpha_i \\ &= \int \sqrt{\frac{\alpha_i}{2\pi}} e^{-\frac{1}{2}\alpha_i w_i^2} \frac{b^a}{\Gamma(a)} \alpha_i^{a-1} e^{-b\alpha_i}\, d\alpha_i. \end{aligned}$$

By multiplying this with

$$\frac{(b + \frac{w_i^2}{2})^{a+\frac{1}{2}}}{\Gamma(a+\frac{1}{2})}$$

and writing the terms not including α_i outside the integral, he got

$$= \frac{b^a \Gamma(a+\frac{1}{2})}{\sqrt{2\pi}\Gamma(a)(b+\frac{w_i^2}{2})^{a+\frac{1}{2}}} \int \frac{(b + \frac{w_i^2}{2})^{a+\frac{1}{2}}}{\Gamma(a+\frac{1}{2})} \alpha_i^{(a+\frac{1}{2})-1} e^{-\alpha_i(b+\frac{1}{2}w_i^2)}\, d\alpha_i.$$

In the equation above, the terms in the integral gives the Gamma$(a+\frac{1}{2}, b+\frac{w_i^2}{2})$ distribution, which integrates to one. Thus, he was left with

$$p(w_i) = \frac{b^a \Gamma(a+\frac{1}{2})}{\sqrt{2\pi}\Gamma(a)} \left(b + \frac{w_i^2}{2}\right)^{-\left(a+\frac{1}{2}\right)},$$

where $\Gamma(\cdot)$ is the gamma function. The equation above is the Student-t distribution, and the complete marginal distribution over the weights $\boldsymbol{w}$ will hence be a product of Student-t distributions. Using this Bayesian prior, the marginal distribution $p(w_i)$ over the weights will have a Student-t distribution, that is sparse compared to a Gaussian marginal distribution over w_i as it is strongly peaked at zero. Using uniform hyperpriors by fixing $a = b = 0$, as we will do later, one will get the improper prior $p(w_i) \propto 1/|w_i|$ (Tipping, 2001). This is approximately the student-t distribution with degrees of freedom close to zero, which is very sparse.

2 | The Relevance Vector Machine

The Relevance Vector Machine (RVM) which we will look into in this chapter, is utilizing a Bayesian learning framework to obtain probabilistic predictions that is sparse in terms of sample size reduction. As each sample weight w_i is related to one basis function $\boldsymbol{\phi}(\boldsymbol{x}_i)$, we will experience that some of the weights from Equation (1.4) will be infinitely peaked at zero, and hence pruned from the model together with their corresponding basis functions. The remaining non-zero weights are the relevance vectors (Tipping, 2001).

2.1 Sparse Sample Selective Framework

In the Relevance Vector Machine, Tipping (2001) used a Bayesian framework. By assigning a sparse prior on the weight parameters w_i, he achieved sparse solutions. That is, each weight w_i is assigned an individual zero-mean hierarchical Gaussian prior. He argued that this made a smooth prior, as preferred to reduce th complexity in the model. The hierarchical sparse prior on the weights $\boldsymbol{w}$ is thus the distribution

$$\boldsymbol{w}|\boldsymbol{\alpha} \sim \mathscr{N}\big(\boldsymbol{w}|0, \boldsymbol{A}^{-1}\big), \tag{2.1}$$

that is

$$p\big(\boldsymbol{w}|\boldsymbol{\alpha}\big) = (2\pi)^{-\frac{N+1}{2}} |\boldsymbol{A}|^{\frac{1}{2}} \exp\bigg\{ -\frac{1}{2}\boldsymbol{w}^\top \boldsymbol{A}\boldsymbol{w}\bigg\}. \tag{2.2}$$

The bold lower case $\boldsymbol{\alpha}$ and the bold capital $\boldsymbol{A}$ is respectively a $N+1$ vector and a $(N+1) \times (N+1)$ diagonal matrix of the hyperparameters α_i corresponding to each separate weight w_i, that is:

$$\begin{aligned}\boldsymbol{\alpha} &= (\alpha_0, \alpha_1, \dots, \alpha_N),\\ \boldsymbol{A} &= \operatorname{diag}(\alpha_0, \alpha_1, \dots, \alpha_N).\end{aligned}$$

In these equations, every element α_i is the inverse variance of the weight parameter w_i, and measures its precision, and therefore also the power of the prior in Equation (2.1). This individual assignment of Gaussian priors is a valuable detail of the RVM, as it gives the model its sparse qualities. The sparseness of this prior distribution was illustrated in Section 1.2.2. Tipping (2001) then defined Gamma distributed hyperpriors on each inverse variance α_i of the hierarchical prior (2.1), and on the noise variance σ^2, that is:

$$\begin{aligned} \alpha_i &\sim \text{Gamma}(\alpha_i|a, b), \\ \sigma^{-2} &\sim \text{Gamma}\big(\sigma^{-2}|c, d\big). \end{aligned}$$

To make the hyperparameters α_i and σ^2 uninformative he fixed all the hyper hyperparameters to be $a = b = c = d = 10^{-4}$, which made the Gamma distributed hyperpriors uniform in practice (Tipping, 2001).

2.2 Calculating Posteriors

From Equation (1.6), using the prior (2.1), and the likelihood of the targets in Equation (1.5), he got the posterior distribution over the unknown parameters

$$p\big(\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2|\boldsymbol{t}\big) = \frac{p\big(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2\big)p\big(\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2\big)}{p\big(\boldsymbol{t}\big)},$$

and a predictive distribution of the form

$$p(t_*|\boldsymbol{t}) = \int p\big(t_*|\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2\big)p\big(\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2|\boldsymbol{t}\big)\, d\boldsymbol{w}\, d\boldsymbol{\alpha}\, d\sigma^2. \tag{2.3}$$

In Equation (2.3) he had no problem calculating the likelihood $p\big(t_*|\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2\big)$. However, it is not possible to compute the posterior distribution in the second term analytically, as it is not possible to take the integral $p(\boldsymbol{t})$ in the denominator (Tipping, 2001). Tipping (2001) then proposed to decompose the posterior distribution as

$$p\big(\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2|\boldsymbol{t}\big) = p\big(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2\big)p\big(\boldsymbol{\alpha}, \sigma^2|\boldsymbol{t}\big). \tag{2.4}$$

In Equation (2.4) it is possible to calculate the posterior distribution over the weights $p\big(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2\big)$ by the following relation:

$$p\big(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2\big) = \frac{p\big(\boldsymbol{t}|\boldsymbol{w}, \sigma^2\big)p\big(\boldsymbol{w}|\boldsymbol{\alpha}\big)}{p\big(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2\big)}. \tag{2.5}$$

Tipping (2001) showed that Equation (2.5) is Gaussian with covariance matrix and mean vector given by

$$\boldsymbol{\Sigma} = \Big(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{A}\Big)^{-1}, \tag{2.6}$$

$$\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top\boldsymbol{t}. \tag{2.7}$$

That is, the posterior distribution over the weights is given by

$$\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha},\sigma^2 \sim \mathscr{N}\big(\boldsymbol{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}\big). \tag{2.8}$$

In footnote number 5 at page 216 Tipping (2001) explained that the derivation of this exact posterior distribution over the weights $\boldsymbol{w}$ can be done by first rewriting Equation (2.5) as

$$p\big(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha},\sigma^2\big)p\big(\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2\big) = p\big(\boldsymbol{t}|\boldsymbol{w},\sigma^2\big)p\big(\boldsymbol{w}|\boldsymbol{\alpha}\big). \tag{2.9}$$

By doing this and using the distribution in Equation (1.5) and (2.1), he was able to write the right hand side of Equation (2.9) as

$$\big(2\pi\sigma^2\big)^{-\frac{N}{2}}(2\pi)^{-\frac{N+1}{2}}|\boldsymbol{A}|^{\frac{1}{2}}\exp\Bigg\{-\frac{1}{2}\bigg[\sigma^{-2}||\boldsymbol{t}-\boldsymbol{\Phi}\boldsymbol{w}||^2+\boldsymbol{w}^\top\boldsymbol{A}\boldsymbol{w}\bigg]\Bigg\}. \tag{2.10}$$

From Equation (2.10) they expanded the exponential part to

$$\begin{aligned} &\exp\bigg\{-\frac{1}{2}\big(\boldsymbol{w}-\boldsymbol{\mu}\big)^\top\boldsymbol{\Sigma}^{-1}\big(\boldsymbol{w}-\boldsymbol{\mu}\big)\bigg\} \\ &\qquad\cdot\exp\bigg\{-\frac{1}{2}\boldsymbol{t}^\top\boldsymbol{C}^{-1}\boldsymbol{t}\bigg\}, \end{aligned} \tag{2.11}$$

where the covariance matrix corresponding to $\boldsymbol{t}$ is given by

$$\boldsymbol{C} = \Big(\sigma^2\boldsymbol{I}+\boldsymbol{\Phi}\boldsymbol{A}^{-1}\boldsymbol{\Phi}^\top\Big). \tag{2.12}$$

The terms $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ are the covariance matrix and the mean vector of the posterior distribution over the weights $\boldsymbol{w}$, given by Equation (2.6) and (2.7). This part of the deduction is not described in detail by Tipping (2001), but to deduce Equation (2.11), we have completed the square in the exponential of Equation (2.10) and used the Woodbury identity to get the covariance matrix $\boldsymbol{C}$ in the second exponential of Equation (2.11). By doing this, and using the relation

$$(\sigma^2)^{-\frac{N}{2}}|\boldsymbol{A}|^{\frac{1}{2}} = |\boldsymbol{\Sigma}|^{-\frac{1}{2}}|\boldsymbol{C}|^{-\frac{1}{2}},$$

we were able to split (2.10) into two distributions, one given by the random weight variable $\boldsymbol{w}$ and the other by the random target variable $\boldsymbol{t}$. By a similar deduction, Tipping (2001)gave the resulting posterior distribution over the weights $\boldsymbol{w}$ by the distribution

$$p\big(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha},\sigma^2\big) = (2\pi)^{-\frac{N+1}{2}}|\boldsymbol{\Sigma}|^{-\frac{1}{2}}\exp\bigg\{-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{w}-\boldsymbol{\mu})\bigg\}.$$

The remaining elements of Equation (2.10) and (2.11) constituted to the marginal likelihood of the targets $p\big(\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2\big)$:

$$p\big(\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2\big) = (2\pi)^{-\frac{N}{2}}|\boldsymbol{C}|^{-\frac{1}{2}}\exp\bigg\{-\frac{1}{2}\boldsymbol{t}^\top(\boldsymbol{C})^{-1}\boldsymbol{t}\bigg\}. \tag{2.13}$$

Thus, Tipping (2001) got that the posterior distribution over the weights $\boldsymbol{w}$ were given by Equation (2.8), and that the marginal likelihood over the targets $\boldsymbol{t}$ is

$$\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2 \sim \mathscr{N}\big(\boldsymbol{t}|0,\boldsymbol{C}\big).$$

2.3 Optimization of the Parameters

Although one primarily wants the complete model to be calculated analytically, this is not possible for the second part of Equation (2.4), and therefore Tipping (2001) was forced to do some approximations. He found the most probable mode estimates $\boldsymbol{\alpha}_{MP}$ and σ^2_{MP}, using maximum likelihood estimation and was then re-estimating cyclically until convergence, which we will look at in Section 2.3.1. However, as the optimization algorithm of the original RVM model has shown to suffer from being computationally slow in the maximization algorithm, Tipping et al. (2003) developed a faster optimization algorithm based on a type-II maximization to handle this limitation. This method is explained in Section 2.3.2.

No matter which method one is using, the estimates $\boldsymbol{\alpha}_{MP}$ and σ^2_{MP} computed will substitute for the hyperparameters $\boldsymbol{\alpha}$ and σ^2 in (2.6) and (2.7). Hence, the RVM modeling turns into a search for the posterior mode estimates of the hyperparameters by maximizing the posterior distribution $p\big(\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}\big)$. Tipping (2001) approximated this distribution by

$$\begin{aligned} p\big(\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}\big) &= \frac{p\big(\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2\big)p\big(\boldsymbol{\alpha}\big)p\big(\sigma^2\big)}{p\big(\boldsymbol{t}\big)} \\ &\propto p\big(\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2\big), \end{aligned}$$

as the denominator will be uninformative in terms of maximization with respect to $\boldsymbol{\alpha}$ and σ^2, and as the uninformative hyperpriors $p(\boldsymbol{\alpha})$ and $p(\sigma^2)$ can be ignored. This means that he was able to maximize $p\big(\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}\big)$ by maximizing the marginal likelihood of the targets $\boldsymbol{t}$ given by the distribution in Equation (2.13). By ignoring all terms not involving σ^2 and α_i, using the relation

$$|\boldsymbol{C}|^{-\frac{1}{2}} = |\boldsymbol{\Sigma}|^{\frac{1}{2}}|\boldsymbol{A}|^{\frac{1}{2}},$$

rewriting the exponential as

$$\exp\left\{-\frac{1}{2}\Big(\sigma^{-2}\boldsymbol{t}^\top\boldsymbol{t}-\boldsymbol{\mu}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\Big)\right\},$$

and taking the logarithm, this is maximizing the two following log-likelihood functions:

$$\mathcal{L}(\boldsymbol{\alpha}) = \frac{1}{2}\ln|\boldsymbol{\Sigma}| + \frac{1}{2}\ln|\boldsymbol{A}| + \frac{1}{2}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}, \tag{2.14}$$

$$\mathcal{L}(\sigma^2) = -\frac{N}{2}\ln\sigma^2 + \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\sigma^{-2}\boldsymbol{t}^\top\boldsymbol{t}-\boldsymbol{\mu}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}). \tag{2.15}$$

2.3.1 Parameter Learning Using Maximum Likelihood and Cyclical Re-Estimation

Tipping (2001) then differentiated the log likelihoods in Equation (2.14) and (2.15) with respect to α_i and σ^2. Using some matrix algebra, and equating to zero, he got the maximum iterative re-estimates:

$$\alpha_i^{\text{new}} = \frac{1-\alpha_i\Sigma_{ii}}{m_i^2}, \tag{2.16}$$

$$\big(\sigma^2\big)^{\text{new}} = \frac{||\boldsymbol{t}-\boldsymbol{\Phi}\boldsymbol{\mu}||^2 + \text{trace}\Big[\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top\boldsymbol{\Phi}\Big]}{N}. \tag{2.17}$$

To calculate these expressions, we are using Jacobi's formula from Equation (0.7). By doing some simplification on the trace term, adding and subtracting the expression $\sigma^2\boldsymbol{\Sigma}\boldsymbol{A}$, we get

$$\begin{aligned}
\text{trace}\big[\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top\boldsymbol{\Phi}\big] &= \text{trace}\Big[\sigma^2\boldsymbol{\Sigma}\big(\boldsymbol{\Phi}^\top\boldsymbol{\Phi}\sigma^{-2}+\boldsymbol{A}\big) - \sigma^2\boldsymbol{\Sigma}\boldsymbol{A}\Big] \\
&= \text{trace}\Big[\sigma^2\big(\boldsymbol{I}-\boldsymbol{A}\boldsymbol{\Sigma}\big)\Big] \\
&= \sigma^2\sum_i 1-\alpha_i\Sigma_{ii} \\
&= \sigma^2\sum_i \gamma_i,
\end{aligned}$$

where $\gamma_i \equiv 1-\alpha_i\Sigma_{ii}$ and $\boldsymbol{\Sigma}$ is from Equation (2.6). The term γ_i can be interpreted as a precision parameter, measuring how accurate the corresponding parameter w_i is determined (MacKay, 1992). If α_i is large, it means that the corresponding weight w_i will be close to zero and not well determined by the data, in that case γ_i is reflecting this by being close to zero. On the other hand, if the weight w_i is well determined by the data,

γ_i will be larger. Using this definition of γ_i, where Σ_{ii} is the i'th element on diagonal of the covariance matrix $\boldsymbol{\Sigma}$ in Equation (2.6) with the present values of $\boldsymbol{\alpha}$ and σ^2, Tipping (2001) simplified the re-estimates:

$$\alpha_i^{\text{new}} = \frac{\gamma_i}{\mu_i^2},$$

$$\left(\sigma^2\right)^{\text{new}} = \frac{||\boldsymbol{t} - \boldsymbol{\Phi}\boldsymbol{\mu}||^2}{N - \sum_i \gamma_i}.$$

In these equations μ_i is the i'th element of the mean vector $\boldsymbol{\mu}$ in Equation (2.7), which means that the estimates are dependent on the previous α_i, and hence that one cannot find any closed form solution for these expressions. Tipping (2001) got the numerically approximated values by re-estimating α_i^{new} and $\left(\sigma^2\right)^{\text{new}}$, and updating $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ cyclically until a reasonable convergence criteria was met.

During this re-estimation some of the α_i-estimates will go to infinity, which is resulting in both the corresponding mean and variance of the posterior distribution over the weights, given by Equation (2.6) and (2.7), being infinitely small. When this happens, the weight w_i will be infinitely peaked at zero, that is $w_i \approx 0$, and the associated basis function is pruned from the model. The remaining non-zero weights are called relevance vectors. This is the way the relevance vector machines by Tipping (2001) achieves sparsity.

2.3.2 Fast Type-II Maximum Likelihood Optimization

As the RVM by Tipping (2001) often is computationally slow in the marginal likelihood maximization, Tipping et al. (2003) developed a faster optimization method for the RVM model. Using this method, they only had to update one α_i at each iteration instead of the whole vector $\boldsymbol{\alpha}$, and they were able to do a incremental and cyclical addition, re-estimation and deletion of basis function. Today, this is the most common version of the RVM and the one that is mostly used. This is because it has all the advantages of the original RVM while at the same time being faster. Hence, this extension of the RVM is important, and we will give a detailed description of it in this section.

From the distribution in Equation (2.13) Tipping et al. (2003) took the logarithm and got the log marginal likelihood

$$\mathcal{L}(\boldsymbol{\alpha}) = -\frac{1}{2}\Big[Nln(2\pi) + ln|\boldsymbol{C}| + \boldsymbol{t}^\top \boldsymbol{C}^{-1}\boldsymbol{t}\Big], \tag{2.18}$$

where the term $\boldsymbol{C}$ is from Equation (2.12). They then decomposed $\boldsymbol{C}$ by separating the terms corresponding to α_i from the others, that is

$$\begin{aligned}\boldsymbol{C} &= \sigma^2\boldsymbol{I} + \sum_{m\neq i} \alpha_m^{-1}\boldsymbol{\phi}_m\boldsymbol{\phi}_m^\top + \alpha_i^{-1}\boldsymbol{\phi}_i\boldsymbol{\phi}_i^\top \\ &= \boldsymbol{C}_{-i} + \alpha_i^{-1}\boldsymbol{\phi}_i\boldsymbol{\phi}_i^\top,\end{aligned}$$

where $\boldsymbol{C}_{-i}$ is the matrix $\boldsymbol{C}$ with the elements corresponding to basis function number i eliminated. By doing this, Tipping et al. (2003) where able to find expressions for $\boldsymbol{C}^{-1}$ and $|\boldsymbol{C}|$ by using the Woodbury and the determinant identities from Equation (0.5) and 0.6, respectively. The resulting expressions are:

$$\boldsymbol{C}^{-1} = \boldsymbol{C}_{-i}^{-1} - \frac{\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i\boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}}{\alpha_i + \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i},$$

$$|\boldsymbol{C}| = |\boldsymbol{C}_{-i}||1 + \alpha_i^{-1}\boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i|.$$

With $\boldsymbol{C}^{-1}$ and $|\boldsymbol{C}|$ inserted into Equation (2.18), they rewrote the log marginal likelihood like

$$\begin{aligned}\mathcal{L}(\boldsymbol{\alpha}) =& -\frac{1}{2}\Bigg[N\ln(2\pi) + \ln|\boldsymbol{C}_{-i}| + \boldsymbol{t}^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{t} \\ & - \ln\alpha_i + \ln(\alpha_i + \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i) - \frac{(\boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{t})^2}{\alpha_i + \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i}\Bigg] \\ =&\mathcal{L}(\boldsymbol{\alpha}_{-i}) + \frac{1}{2}\left[\ln\alpha_i - \ln(\alpha_i + s_i) + \frac{q_i^2}{\alpha_i + s_i}\right] \\ =&\mathcal{L}(\boldsymbol{\alpha}_{-i}) + \ell(\alpha_i),\end{aligned}$$

where

$$s_i \equiv \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i \qquad \text{and} \qquad q_i \equiv \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{t}. \tag{2.19}$$

The log marginal likelihood was then decomposed into two terms, the log marginal likelihood with $\boldsymbol{\alpha}_i$ eliminated, $\mathcal{L}(\boldsymbol{\alpha}_{-i})$, and the function $\ell(\alpha_i)$, which is the only place the term α_i appears. This means that differentiating $\mathcal{L}(\boldsymbol{\alpha})$ with respect to α_i is the same as differentiating $\ell(\alpha_i)$, which obviously is less computationally expensive than working with the whole matrix as Tipping (2001) did in the slower algorithm. Doing this differentiation and equating to zero Tipping et al. (2003) got an explicit solution for the α_i estimate:

$$\alpha_i = \begin{cases} \frac{s_i^2}{q_i^2 - s_i} & \text{if } q_i^2 > s_i \\ & \\ \infty & \text{if } q_i^2 \leq s_i \end{cases}. \tag{2.20}$$

When $\alpha_i = \infty$ both the variance and the mean from Equation (2.6) and (2.7) goes to zero, and the corresponding weight w_i is infinitely peaked at zero. Thus, observation number i is pruned from the model. The important difference between the optimization algorithm of Tipping (2001) and this faster one by Tipping et al. (2003) is that the latter one can find explicit solutions to the maximization problem. To estimate σ^2 Tipping et al. (2003) still used the re-estimate from Equation (2.17).

Tipping et al. (2003) then suggested to update and keep the expressions

$$\begin{aligned} S_m &= \boldsymbol{\phi}_m^\top \boldsymbol{C}^{-1} \boldsymbol{\phi}_m \\ &= \sigma^{-2} \boldsymbol{\phi}_m^\top \boldsymbol{\phi}_m - (\sigma^{-2})^2 \boldsymbol{\phi}_m^\top \boldsymbol{\Phi} \boldsymbol{\Sigma} \boldsymbol{\Phi}^\top \boldsymbol{\phi}_m, \\ Q_m &= \boldsymbol{\phi}_m^\top \boldsymbol{C}^{-1} \boldsymbol{t} \\ &= \sigma^{-2} \boldsymbol{\phi}_m^\top \boldsymbol{t} - (\sigma^{-2})^2 \boldsymbol{\phi}_m^\top \boldsymbol{\Phi} \boldsymbol{\Sigma} \boldsymbol{\Phi}^\top \boldsymbol{t}, \end{aligned} \tag{2.21}$$

as it is easier to work with than s_i and q_i. To deduce the Equations in (2.21) they used the Woodbury identity in Equation (0.5). Using the expressions in Equation (2.21) it follows that

$$s_m = \frac{\alpha_m S_m}{\alpha_m - S_m} \qquad \text{and} \qquad q_m = \frac{\alpha_m Q_m}{\alpha_m - S_m},$$

where σ_{MP}^2 is updated sequentially together with α_i, using the expression in Equation (4.11).

2.4 Making Predictions

With the estimates defined as above it is now possible to predict for new targets t_* using the predictive distribution in Equation (2.3). With the posterior distribution over the weights $\boldsymbol{w}$ given by a Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$ and mean vector $\boldsymbol{\mu}$ from Equation (2.6) and (2.7), conditioning on the values $\boldsymbol{\alpha}_{MP}$ and σ_{MP}^2, the predictive distribution is given by

$$p(t_* | \boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \sigma_{MP}^2) = \int p(t_* | \boldsymbol{w}, \sigma_{MP}^2) p(\boldsymbol{w} | \boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \sigma_{MP}^2)\, d\boldsymbol{w},$$

In this Equation, both distributions are Gaussian such that it is easily shown that also the integral is Gaussian with

$$\mu_* = \boldsymbol{\mu}^\top \boldsymbol{\phi}(\boldsymbol{x}_*),$$

$$\sigma_*^2 = \sigma_{MP}^2 + \boldsymbol{\phi}(\boldsymbol{x}_*)^\top \boldsymbol{\Sigma} \boldsymbol{\phi}(\boldsymbol{x}_*).$$

This can be shown by completing the squares, integrating out the sample weights $\boldsymbol{w}$ and doing some calculus. Thus, by using the RVM method Tipping (2001) got probabilistic predictions based on

$$t_* | \boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \sigma_{MP}^2 \sim \mathscr{N}\big(t_* | \mu_*, \sigma_*^2\big). \tag{2.22}$$

Hence, in RVM the predicted value of t_* is given by the mean μ_* with the associated uncertainty σ_*^2. This predictive part of the method follows the same approach both for the original RVM and the faster version, just with the parameter estimated by different procedures, which will be described in further detail below.

2.5 The Relevance Vector Algorithm

The above sections shows that it is possible to estimate the parameters in two different ways, one being faster than the other. The algorithms of these different approaches on finding the estimates will be quite different from each other, with the main difference being if one considers the whole kernel basis function matrix or just one vector at a time. The resulting procedures are similar, but still very different from each other.

Algorithm 1 Relevance Vector Machine (RVM)

```
Initialize $\boldsymbol{\alpha}$ and $\sigma^2$ to some reasonable values
Compute $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$
while convergence criteria is not met do
    for all $\alpha_i$ in $\boldsymbol{\alpha}$ do
        if $\alpha_i > \alpha_{\text{Thresh}}$ then
            delete $\boldsymbol{\phi}_i$ and $\alpha_i$
        end if
    end for
    Update $\boldsymbol{\Sigma}$, $\boldsymbol{\mu}$, $\boldsymbol{\alpha}$ and $\sigma^2$
end while
```

The algorithm of the Relevance Vector Machine by Tipping (2001) is iterative and requires cyclically re-estimating $\boldsymbol{\alpha}$ and σ^2 until some convergence criteria on the total change in estimates is met. In addition, a threshold on the α_i-estimates is set, which indicates that when $\alpha_i > \alpha_{\text{Thresh}}$, the hyperparameter α_i is assumed to be infinitely large and hence w_i infinitely peaked at zero. The algorithm will be as in Algorithm 1 (Tipping

(2001), Fletcher (2010)), where a reasonable value of σ^2 could simply be the variance in the data or a scaling of the variance. Tipping et al. (2003) suggested to use $\mathrm{var}(t)/10$ as the initial value.

In the Fast Relevance Vector algorithm Tipping et al. (2003) started with an empty kernel basis function matrix, and was then cyclically adding the relevant kernel basis function vectors $\boldsymbol{\phi}_i$ to the model. By continuously evaluating random $\boldsymbol{\phi}_i$'s until some convergence criteria was met, they added, deleted, and re-estimated the α_i's and the corresponding kernel functions.

Algorithm 2 Fast Relevance Vector Machine (FRVM)

```
Initialize $\sigma^2$ to a reasonable value
Initialize $\alpha_i$ with a single basis vector $\boldsymbol{\phi}_i$, by Equation (2.20):

    $\alpha_i = \frac{||\boldsymbol{\phi}_i||^2}{||\boldsymbol{\phi}_i^\top \boldsymbol{t}||^2/||\boldsymbol{\phi}_i||^2 - \sigma^2}.$

All other $\alpha_m$ are notionally set to infinity
Compute $\boldsymbol{\Sigma}$, $\boldsymbol{\mu}$, $s_m$ and $q_m$
while convergence criteria is not met do
    Choose a basis vector $\boldsymbol{\phi}_i$
    Compute $q_i^2 - s_i$
    if $q_i^2 - s_i > 0$ and $\alpha_i < \infty$ then
        Re-estimate $\alpha_i$
    else if $q_i^2 - s_i > 0$ and $\alpha_i = \infty$ then
        Add $\boldsymbol{\phi}_i$ to the model
    else if $q_i^2 - s_i \leq 0$ and $\alpha_i < \infty$ then
        Delete $\boldsymbol{\phi}_i$ from the model (set $\alpha_i = \infty$)
    end if
    Update $\boldsymbol{\Sigma}$, $\boldsymbol{\mu}$, $s_m$, $q_m$ and $\sigma^2$
end while
```

The interpretation of the addition, deletion, and re-estimation procedure in Algorithm 2 is that $q_i^2 - s_i > 0$ indicates that α_i should be in the model. When $q_i^2 - s_i \leq 0$, the hyperparameter α_i should not be in the model. Together with $\alpha_i < \infty$ and $\alpha_i = \infty$ indicating if the given α_i is in the model or not, Tipping et al. (2003) are adding, deleting and re-estimating due to this combination. The initial value of α_i is chosen specifically like given in the algorithm, because when $\boldsymbol{C}_{-i}^{-1} = \sigma^2$ Equation (2.20) gives

$$\begin{aligned}\alpha_i &= \frac{(\sigma^{-2})^2||\boldsymbol{\phi}_i||^4}{(\sigma^{-2})^2||\boldsymbol{\phi}_i^\top \boldsymbol{t}||^2 - \sigma^{-2}||\boldsymbol{\phi}_i||^2} \\ &= \frac{||\boldsymbol{\phi}_i||^2}{||\boldsymbol{\phi}_i^\top \boldsymbol{t}||^2/||\boldsymbol{\phi}_i||^2 - \sigma^2}.\end{aligned}$$

2.5.1 Update Formulas for Effective Estimation

Tipping et al. (2003) gave expressions for effective calculations for the updated values in the addition, re-estimation, and deletion procedure. However, it is not clear in the paper how they deduced these expressions. In this section we are giving a deduction of the update formulas for the FRVM method. The updated quantities are denoted with a tilde, e.g. $\widetilde{\alpha}$ is the updated value of α. The indexes $_{\text{add}}$, $_{\text{re}}$ and $_{\text{del}}$ are used to denote addition, re-estimation and deletion, respectively. Further, they used the index i to denote a basis function where the hyperparameter α_i should be updated, and the index j to denote the index within the given basis that corresponds to i.

Adding a new basis function

Adding basis function number i means that the updated kernel basis function matrix and the new matrix of hyperparameters should respectively be of the form

$$\widetilde{\boldsymbol{\Phi}}_{\text{add}} = (\boldsymbol{\Phi}, \boldsymbol{\phi}_i) \qquad \text{and} \qquad \widetilde{\boldsymbol{A}}_{\text{add}} = \text{diag}(\boldsymbol{\alpha}, \alpha_i).$$

This means that the new covariance matrix will be

$$\begin{aligned}\widetilde{\boldsymbol{\Sigma}}_{\text{add}} &= (\sigma^{-2}\widetilde{\boldsymbol{\Phi}}_{\text{add}}^\top \widetilde{\boldsymbol{\Phi}}_{\text{add}} + \widetilde{\boldsymbol{A}}_{\text{add}})^{-1} \\ &= \begin{bmatrix} \boldsymbol{\Sigma}^{-1} & \sigma^{-2}\boldsymbol{\Phi}^\top \boldsymbol{\phi}_i \\ \sigma^{-2}\boldsymbol{\phi}_i^\top \boldsymbol{\Phi} & \alpha_i + \sigma^{-2}\boldsymbol{\phi}_i^\top \boldsymbol{\phi}_i \end{bmatrix}^{-1}.\end{aligned}$$

By using the inverse block matrix formula in Equation (0.8) to compute this invert, one get

$$\widetilde{\boldsymbol{\Sigma}}_{\text{add}} = \begin{bmatrix} \boldsymbol{\Sigma} + \sigma^{-4}G_{ii}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top \boldsymbol{\phi}_i \boldsymbol{\phi}_i^\top \boldsymbol{\Phi}\boldsymbol{\Sigma} & -\sigma^{-2}G_{ii}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top \boldsymbol{\phi}_i \\ -\sigma^{-2}G_{ii}(\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top \boldsymbol{\phi}_i)^\top & G_{ii} \end{bmatrix},$$

where $G_{ii} = (\alpha_i + S_i)^{-1}$. By inserting this and completing the calculations, the updated mean vector by Tipping et al. (2003) is

$$
\begin{aligned}
\widetilde{\boldsymbol{\mu}}_{\text{add}} &= \sigma^{-2}\widetilde{\boldsymbol{\Sigma}}_{\text{add}}\widetilde{\boldsymbol{\Phi}}^{\top}_{\text{add}}\boldsymbol{t} \\
&= \begin{bmatrix} \boldsymbol{\mu} - \sigma^{-2}\mu_i\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\top}\boldsymbol{\phi}_i \\ m_i \end{bmatrix},
\end{aligned}
$$

where $m_i = G_{ii}Q_i$. Further, the updated expressions for $\widetilde{S}_{m,\text{add}}$ and $\widetilde{Q}_{m,\text{add}}$ are given by

$$
\begin{aligned}
\widetilde{S}_{m,\text{add}} &= \boldsymbol{\phi}_m\boldsymbol{B}\boldsymbol{\phi}_m - \boldsymbol{\phi}_m^{\top}\boldsymbol{B}\widetilde{\boldsymbol{\Phi}}_{\text{add}}\widetilde{\boldsymbol{\Sigma}}_{\text{add}}\widetilde{\boldsymbol{\Phi}}^{\top}_{\text{add}}\boldsymbol{B}\boldsymbol{\phi}_m, \\
\widetilde{Q}_{m,\text{add}} &= \boldsymbol{\phi}_m\boldsymbol{B}\boldsymbol{t} - \boldsymbol{\phi}_m^{\top}\boldsymbol{B}\widetilde{\boldsymbol{\Phi}}_{\text{add}}\widetilde{\boldsymbol{\Sigma}}_{\text{add}}\widetilde{\boldsymbol{\Phi}}^{\top}_{\text{add}}\boldsymbol{B}\boldsymbol{t}.
\end{aligned}
$$

By rewriting $\widetilde{\boldsymbol{\Phi}}_{\text{add}}\widetilde{\boldsymbol{\Sigma}}_{\text{add}}\widetilde{\boldsymbol{\Phi}}^{\top}_{\text{add}}$ as

$$
\widetilde{\boldsymbol{\Phi}}_{\text{add}}\widetilde{\boldsymbol{\Sigma}}_{\text{add}}\widetilde{\boldsymbol{\Phi}}^{\top}_{\text{add}} = \boldsymbol{\Phi}\widetilde{\boldsymbol{\Sigma}}_{\text{add},1,1}\boldsymbol{\Phi}^{\top} + \boldsymbol{\phi}_i\widetilde{\boldsymbol{\Sigma}}_{\text{add},2,1}\boldsymbol{\Phi}^{\top} + \boldsymbol{\Phi}\widetilde{\boldsymbol{\Sigma}}_{\text{add},1,2} + \boldsymbol{\phi}_i\widetilde{\boldsymbol{\Sigma}}_{\text{add},2,2}\boldsymbol{\phi}_i^{\top}\boldsymbol{\phi}_i^{\top},
$$

one gets the estimates (Tipping et al., 2003):

$$
\begin{aligned}
\widetilde{S}_{m,\text{add}} &= S_m - G_{ii}(\sigma^{-2}\boldsymbol{\phi}_m^{\top}\boldsymbol{e}_i)^2, \\
\widetilde{Q}_{m,\text{add}} &= Q_m - m_i(\sigma^{-2}\boldsymbol{\phi}_m^{\top}\boldsymbol{e}_i).
\end{aligned}
$$

In the equations above $\boldsymbol{e}_i = \boldsymbol{\phi}_i - \sigma^{-2}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\top}\boldsymbol{\phi}_i$. Lastly, the change in marginal likelihood is straightforward calculated as (Tipping et al., 2003):

$$
\begin{aligned}
2\Delta\mathcal{L}_{\text{add}} &= 2\ell\left(\frac{S_i^2}{Q_i^2 - S_i}\right) \\
&= \frac{Q_i^2 - S_i}{S_i} + \ln\frac{S_i}{Q_i^2}.
\end{aligned}
$$

Re-estimating a basis function

When re-estimating α_i, the kernel basis function matrix is unchanged, but the matrix of hyperparameters $\boldsymbol{\alpha}$ will be

$$
\widetilde{\boldsymbol{A}}_{\text{re}} = \boldsymbol{A} + \boldsymbol{1}_j(\tilde{\alpha}_i - \alpha_i)\boldsymbol{1}_j^{\top},
$$

where $\boldsymbol{1}_j^{\top} = \big(0, \ldots, 1, \ldots, 0\big)$, with one at position j. Thus, using the Woodbury identity in Equation (0.5), the update formulas for re-estimation of the new covariance matrix is of the form:

$$\begin{aligned}
\widetilde{\boldsymbol{\Sigma}}_{\text{re}} &= \left(\sigma^{-2}\boldsymbol{\Phi}^{\top}\boldsymbol{\Phi} + \widetilde{\boldsymbol{A}}_{\text{re}}\right)^{-1} \\
&= \left(\boldsymbol{\Sigma}^{-1} + \mathbf{1}_j(\tilde{\alpha}_i - \alpha_i)\mathbf{1}_j^{\top}\right)^{-1} \\
&= \boldsymbol{\Sigma} - \kappa_j \boldsymbol{\Sigma}_j \boldsymbol{\Sigma}_j^{\top}. \qquad (2.23)
\end{aligned}$$

In Equation (2.23)

$$\kappa_j = \left((\tilde{\alpha}_i - \alpha_i)^{-1} + \Sigma_{jj}\right)^{-1},$$

and Σ_j is the j'th column of the covariance matrix $\boldsymbol{\Sigma}$. Using this expression, one gets the update formulas for the mean vector $\boldsymbol{\mu}$:

$$\begin{aligned}
\widetilde{\boldsymbol{\mu}}_{\text{re}} &= \sigma^{-2}\widetilde{\Sigma}_{re}\boldsymbol{\Phi}^{\top}\boldsymbol{t} \\
&= \boldsymbol{\mu} - \kappa_j \sigma^{-2}\boldsymbol{\Sigma}_j\boldsymbol{\Sigma}_j^{\top}\boldsymbol{\Phi}^{\top}\boldsymbol{t} \\
&= \boldsymbol{\mu} - \kappa_j \mu_j.
\end{aligned}$$

Lastly, using the update formula from Equation (2.23), the corresponding formulas for S_m, Q_m and the likelihood is given by (Tipping et al., 2003):

$$\begin{aligned}
\widetilde{S}_{m,\text{re}} &= \sigma^{-2}\boldsymbol{\phi}_m^{\top}\boldsymbol{\phi} - (\sigma^{-2})^2\boldsymbol{\phi}_m\boldsymbol{\Phi}\widetilde{\boldsymbol{\Sigma}}_{\text{re}}\boldsymbol{\Phi}^{\top}\boldsymbol{\phi}_m \\
&= S_m + \kappa_j(\sigma^{-2}\boldsymbol{\Sigma}_j^{\top}\boldsymbol{\Phi}^{\top}\boldsymbol{\phi}_m)^2, \\
\widetilde{Q}_{m,\text{re}} &= \sigma^{-2}\boldsymbol{\phi}_m^{\top}\boldsymbol{\phi} - (\sigma^{-2})^2\boldsymbol{\phi}_m\boldsymbol{\Phi}\widetilde{\boldsymbol{\Sigma}}_{\text{re}}\boldsymbol{\Phi}^{\top}\boldsymbol{t} \\
&= Q_m + \kappa_j\mu_j(\sigma^{-2}\boldsymbol{\Sigma}_j^{\top}\boldsymbol{\Phi}^{\top}\boldsymbol{\phi}_m), \\
2\nabla\boldsymbol{\mathcal{L}}_{\text{re}} &= 2\ell(\tilde{\alpha}_i^{-1} - \alpha_i^{-1})^{-1}) \\
&= \frac{Q_i^2}{S_i + (\tilde{\alpha}_i^{-1} - \alpha_i^{-1})^{-1}))} - \ln\left\{1 + \frac{S_i}{(\tilde{\alpha}_i^{-1} - \alpha_i^{-1})^{-1})}\right\}.
\end{aligned}$$

Deleting a basis function

When deleting a basis function, one must remove every element of the covariance matrix that corresponds to the given basis function and hyperparameter. Based on Tipping et al.

(2003) the updated covariance matrix is given by

$$\widetilde{\boldsymbol{\Sigma}}_{\text{del}} = \boldsymbol{\Sigma} - \frac{1}{\Sigma_{jj}} \boldsymbol{\Sigma}_j \boldsymbol{\Sigma}_j^\top .$$

Thus, the update formula for the mean vector $\boldsymbol{\mu}$ is straight forward given by

$$\begin{aligned} \widetilde{\boldsymbol{\mu}}_{\text{del}} &= \sigma^{-2} \widetilde{\boldsymbol{\Sigma}}_{\text{del}} \boldsymbol{\Phi}^\top \boldsymbol{t} \\ &= \boldsymbol{\mu} - \frac{\mu_j}{\Sigma_{jj}} \boldsymbol{\Sigma}_j . \end{aligned}$$

The formulas corresponding to the update of S_m, Q_m and $2\nabla\mathcal{L}$ are easily shown to be given by:

$$\begin{aligned} \widetilde{S}_{m,\text{del}} &= \sigma^{-2} \boldsymbol{\phi}_m^\top \boldsymbol{\phi} - (\sigma^{-2})^2 \boldsymbol{\phi}_m \boldsymbol{\Phi} \widetilde{\boldsymbol{\Sigma}}_{\text{del}} \boldsymbol{\Phi}^\top \boldsymbol{\phi}_m \\ &= S_m + \frac{1}{\Sigma_{jj}} (\sigma^{-2} \boldsymbol{\Sigma}_j^\top \boldsymbol{\Phi}^\top \boldsymbol{\phi}_m)^2 , \\ \widetilde{Q}_{m,\text{del}} &= \sigma^{-2} \boldsymbol{\phi}_m^\top \boldsymbol{\phi} - (\sigma^{-2})^2 \boldsymbol{\phi}_m \boldsymbol{\Phi} \widetilde{\boldsymbol{\Sigma}}_{\text{del}} \boldsymbol{\Phi}^\top \boldsymbol{t} \\ &= Q_m + \frac{\mu_j}{\Sigma_{jj}} (\sigma^{-2} \boldsymbol{\Sigma}_j^\top \boldsymbol{\Phi}^\top \boldsymbol{\phi}_m), \\ 2\nabla\mathcal{L}_{\text{del}} &= 2\ell(-\alpha_i) \\ &= \frac{Q_i^2}{S_i - \alpha_i} - \ln\Big(1 - \frac{S_i}{\alpha_i}\Big). \end{aligned}$$

2.6 The Relevance Vector Classification Machine

In this section, we will go into the RVM in the case of classification where the likelihood over the targets $\boldsymbol{t}$ is assumed to be Bernoulli distributed. We will look at the model for a two class random variable, but it works similar for multi class variables.

2.6.1 Framework of RVM Classification

When data are categorical, the RVM method for classification can be used. In that case the targets $\boldsymbol{t}$ are assumed to be Bernoulli distributed with

$$p(\boldsymbol{t}|\boldsymbol{w}) = \prod_{i=1}^{N} \sigma_i^{t_i} \{1 - \sigma_i\}^{1-t_i} \quad \text{where} \quad t_i \quad \epsilon \quad \{0,1\}, \tag{2.24}$$

and $\boldsymbol{\sigma} = \sigma(\boldsymbol{\Phi w}) = \sigma(\boldsymbol{y})$ and $\sigma_i = \sigma(\boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})$, with $\sigma(\cdot)$ being the logistic sigmoid link function (Tipping, 2001):

$$\sigma(x) = \frac{1}{1+e^{-x}}.$$

It should be noted that this distribution does not depend on the noise-variance σ^2, such that one does not have to work with the noise-variance when doing classification. With this prior distribution, the posterior distribution over the sample weights $\boldsymbol{w}$ can be written as:

$$\begin{aligned} p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}) &= \frac{p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})}{\int p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w}} \\ &= \frac{p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})}{p(\boldsymbol{t}|\boldsymbol{\alpha})}. \qquad (2.25) \end{aligned}$$

The logarithm of Equation (2.25) with respect to the sample weights $\boldsymbol{w}$ gives

$$\begin{aligned} \ln p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}) &= \ln p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha}) \qquad (2.26) \\ &= \sum_{i=1}^{N} t_i \ln\left[\sigma(\boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})\right] + (1-t_i)\ln\left[1-\sigma(\boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{w})\right] \qquad (2.27) \\ &\quad + \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{A w} + \text{const.}. \qquad (2.28) \end{aligned}$$

In this situation, as the likelihood over the targets $\boldsymbol{t}$ are not Gaussian, it is not possible to find analytical expressions for the posterior distribution over the sample weights $\boldsymbol{w}$, and one cannot integrate over these weights. Therefore, the theory of the Laplace approximation and the iterative reweighted least squares (IRLS) (Bishop, 2006) must be introduced, which both will be used to approach this Bayesian treatment in the classification case.

2.6.2 Laplace's Approximation

The Laplace approximation is about finding a Gaussian approximation to a probability distribution, which enable us to apply all the handy properties of the Gaussian distribution to more complex distributions. We will explain the approximation using a single variable, but it works in the same way for a multidimensional space of variables. We will go through the general Laplace approximation, and further deduce the Laplace approximation to a posterior distribution, as this is the version needed here.

Laplace's Approximation in General

The Laplace approximation will work for uni-modal functions that has most of its mass concentrated in a small area of its domain, that is functions $f(z)$ of the $\mathcal{L}^2$-class (Peng, 2018), meaning that

$$\int_a^b f(z)^2 dz < \infty.$$

One can imagine a function that looks something like the one in Figure 2.1, where the integral is approximated with a step function, that is

$$\int_a^b f(z)dz \approx f(z_0)\epsilon,$$

where the term ϵ is a small value.

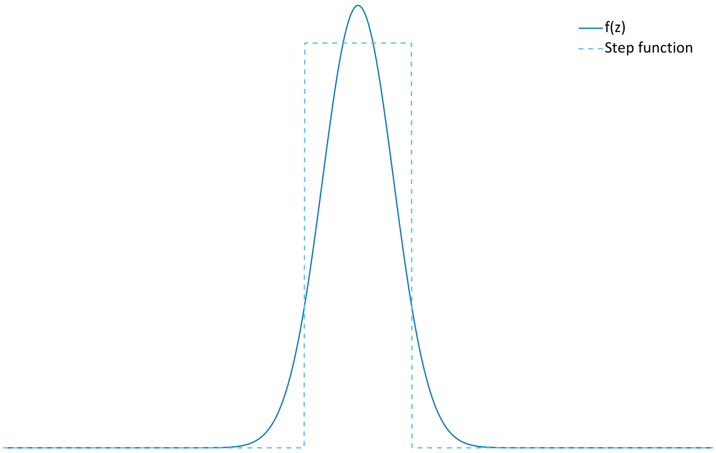

Figure 2.1: A function $f(x)$ in solid and an example of a step function approximation of the integral in stipulated.

This is the fundamental idea of the Laplace approximation, where a Gaussian distribution is used instead of a step function. Thus, it is possible to approximate the integral

$$\int_a^b f(z)dz,$$

which is the equivalent of finding

$$\int_a^b exp\{\ln f(z)\}dz = \int exp\{g(z)\}dz,$$

where $g(z) = \ln f(z)$. From here, a Taylor expansion of $g(z)$ around z_0 gives

$$\int_a^b f(z)dz \simeq \int exp\{g(z_0) - \frac{A}{2}(z - z_0)^2\}dz, \tag{2.29}$$

where

$$A = -\frac{d^2}{dz^2}g(z)|_{z=z_0}, \tag{2.30}$$

and the part corresponding to the first derivative of $g(z)$ is zero. In Equation (2.29) z_0 is the z value at the mode of the function to be approximated, that is z_0 satisfying:

$$\frac{df(z)}{dz}|_{z=z_0} = 0. \tag{2.31}$$

Further, Equation (2.29) can be simplified by using that $g(z_0)$ is a constant that can be taken outside the integral:

$$\int_a^b f(z)dz \simeq f(z_0) \int_a^b exp\{ - \frac{A}{2}(z - z_0)^2\}dz. \tag{2.32}$$

In Equation (2.32) one can recognize the part inside the integral to be proportional to a Gaussian distribution with mean z_0 and covariance A^{-1}. Thus, this can be rewritten as

$$\int_a^b f(z)dz \simeq f(z_0)\sqrt{\frac{2\pi}{A}} \int_a^b \mathscr{N}\big(z|z_0, A^{-1}\big)dz,$$

which when $a = -\infty$ and $b = \infty$ is simplified to

$$\int f(z)dz \simeq f(z_0)\sqrt{\frac{2\pi}{A}}.$$

For a multidimensional variable $\boldsymbol{z}$, this is equivalently given by:

$$\int f(\boldsymbol{z})dz \simeq f(z_0)\sqrt{2\pi}|\boldsymbol{A}|^{-\frac{1}{2}}. \tag{2.33}$$

Laplace's Approximation for Posterior Distribution

When using Laplace approximation to approximate a posterior distribution, one can assume that the not Gaussian distribution $p(z)$ is given by

$$p(z) = \frac{1}{C} f(z), \tag{2.34}$$

where

$$C = \int f(z)dz$$

is the unknown normalization constant (Bishop, 2006). Next, one can approximate a Gaussian distribution $q(z)$ that is centered in the mode of the distribution that is to be approximated. Therefore, the first thing to do, is to find the mode of the distribution. That is the point satisfying Equation (2.31). By again using a Taylor expansion of $g(z) = \ln f(z)$ around the mode z_0, Bishop (2006) got

$$g(z) = \ln f(z) \approx lnf(z_0) - \frac{1}{2}A(z - z_0)^2,$$

where A is given by Equation (2.30), and the term corresponding to the first derivative of $f(z)$ disappear, because of the relation in Equation (2.31). The exponential of this equation is then given by

$$f(z) \simeq f(z_0) \exp -\frac{A}{2}(z - z_0)^2. \tag{2.35}$$

That is the term corresponding to the approximation of $f(z)$ in Equation (2.34). The next step is to find the normalization constant C. For the normal distribution $X \sim \mathcal{N}\big(\mu, \gamma^{-1}\big)$ one has that

$$p(x) = \frac{\sqrt{\gamma}}{\sqrt{2\pi}} \exp -\frac{\gamma}{2}(x - \mu)^2,$$

and as the integral over a normal distribution gives 1:

$$\frac{\sqrt{\gamma}}{\sqrt{2\pi}} \int \exp -\frac{\gamma}{2}(x - \mu)^2 dx = 1,$$

$$\Downarrow$$

$$\int \exp -\frac{\gamma}{2}(x - \mu)^2 dx = \frac{\sqrt{2\pi}}{\sqrt{\gamma}}. \tag{2.36}$$

By the result in Equation (2.36) and the approximation of $f(z)$ in Equation (2.35), the normalization constant C of $f(z)$ is approximated by:

$$C = \int f(z) dz \tag{2.37}$$

$$\approx f(z_0) \int \exp -\frac{A}{2}(z - z_0)^2 dz \tag{2.38}$$

$$= f(z_0)\sqrt{\frac{2\pi}{A}}. \tag{2.39}$$

Altogether, using the approximations of $f(z)$ and C from Equation (2.35) and (2.12), this gives the approximation

$$q(z) = \frac{1}{C} f(z) \tag{2.40}$$

$$= \left(\frac{A}{\sqrt{2\pi}}\right)^{\frac{1}{2}} \exp\left\{ -\frac{A}{2}(z - z_0)^2 \right\}, \tag{2.41}$$

that is

$$q(z) \sim \mathcal{N}\big(z|z_0, A^{-1}\big), \tag{2.42}$$

where z_0 is the mode of $p(z)$ and A^{-1} is the inverse of the negative Hessian matrix. Thus, the Laplace approximation to a Gaussian distribution of $p(z)$ is given by Equation (2.42). For a multidimensional variable $\boldsymbol{z}$ with distribution $p(\boldsymbol{z})$, the approximation is given by an equivalent derivation (Bishop, 2006):

$$q(\boldsymbol{z}) \sim \mathscr{N}\big(\boldsymbol{z}|\boldsymbol{z_0}, \boldsymbol{A}^{-1}\big).$$

Thus, the Laplace approximation is approximating the not Gaussian distribution $p(\boldsymbol{z})$ by the Gaussian distribution $q(\boldsymbol{z})$. In fact, if the distribution $p(\boldsymbol{z})$ is Gaussian itself the Laplace approximation $q(\boldsymbol{z})$ is exact, as the Gaussian distribution $p(z) \sim \mathscr{N}\big(z|\mu, \sigma^2\big)$ have the properties:

$$z_0 = \mu \quad \text{and} \quad A = \sigma^{-2}.$$

2.6.3 Iterative Reweighted Least Squares

In some cases, it is not possible to find a closed-form solution to minimize the error. In such cases, as the error function is concave, one can use the Newton-Raphson iterative reweighted least squares (IRLS) (Bishop, 2006). The Newton-Raphson update formula to minimize an error function $E(\boldsymbol{w})$ is given by:

$$\boldsymbol{w}^{(new)} = \boldsymbol{w}^{(old)} - \boldsymbol{H}^{-1}\nabla E\big(\boldsymbol{w}^{(old)}\big),$$

where $\boldsymbol{H}$ is the Hessian matrix. When using Laplace's approximation on RVM for regression, the likelihood given by Equation (1.5) gives that the gradients of the posterior distribution in Equation (2.9) with respect to the sample weights are given by:

$$\nabla \mathcal{L}(\boldsymbol{w}) = \sigma^{-2}\boldsymbol{\Phi}^\top \boldsymbol{t} - \big(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{A}\big)\boldsymbol{w}, \tag{2.43}$$

$$\nabla\nabla \mathcal{L}(\boldsymbol{w}) = -\big(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{A}\big). \tag{2.44}$$

Thus, the mean vector is given by Equation (2.7) and the Hessian matrix is given by the negative inverse of Equation (2.6), which is as expected since the Laplace approximation always is exact for a Gaussian distribution. The Newton-Raphson update formula is then given by

$$\boldsymbol{w}^{(new)} = \boldsymbol{w}^{(old)} + \big(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{A}\big)^{-1}\Big\{\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{t} - \big(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{A}\big)\boldsymbol{w}^{old}\Big\} \tag{2.45}$$

$$= \sigma^{-2}\big(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{A}\big)^{-1}\boldsymbol{\Phi}^\top\boldsymbol{t}, \tag{2.46}$$

that is equal to Equation (2.7) and is hence exact. As the quadratic likelihood gives a constant Hessian matrix in terms of the sample weights $\boldsymbol{w}$ this is as expected (Bishop, 2006). However, when the likelihood function is not quadratic, as in the RVM for classification, the Laplace approximation based on the likelihood function in Equation (2.24) and (2.26) are not exact. The gradient and Hessian are then given by (Bishop, 2006):

$$\nabla\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^{N}\Big\{\frac{t_i}{\sigma_i}\sigma_i(1-\sigma_i)\boldsymbol{\phi}(\boldsymbol{x}_i) - \frac{1-t_i}{1-\sigma_i}\sigma_i(1-\sigma_i)\boldsymbol{\phi}(\boldsymbol{x}_i)\Big\}$$

$$= \boldsymbol{\Phi}^\top(\boldsymbol{t}-\boldsymbol{\sigma}) - \boldsymbol{A}\boldsymbol{w}, \tag{2.47}$$

$$\nabla\nabla\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^{N}\Big\{-\sigma_i(1-\sigma_i)\boldsymbol{\phi}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_i)^\top - \boldsymbol{A}\Big\}$$

$$= -\big(\boldsymbol{\Phi}^\top\boldsymbol{R}\boldsymbol{\Phi} + \boldsymbol{A}\big), \tag{2.48}$$

where

$$\boldsymbol{R} = \mathrm{diag}(\sigma_i(1-\sigma_i)). \tag{2.49}$$

To deduce the equations above, the following property is used:

$$\frac{\partial\sigma_i}{\partial w_i} = \sigma_i(1-\sigma_i)\boldsymbol{\phi}(\boldsymbol{x}_i).$$

In this case, one can see that the Hessian matrix is dependent on the sample weights $\boldsymbol{w}$, and is therefore not exact. Anyhow, the logistic sigmoid function will always be between zero and one: $0 < \sigma_i < 1$. As the Hessian matrix $\boldsymbol{H}$ is positive definite, and the error function is concave in terms of the sample weights $\boldsymbol{w}$, it will have a unique minimum (Bishop, 2006). Therefore, one can use the Newton-Raphson update formula given by:

$$\boldsymbol{w}^{new} = \boldsymbol{w}^{old} - (\boldsymbol{\Phi}^\top\boldsymbol{R}\boldsymbol{\Phi} - \boldsymbol{A})^{-1}(\boldsymbol{\Phi}^\top(\boldsymbol{\sigma}-\boldsymbol{t}) - \boldsymbol{A}\boldsymbol{w}).$$

As the update formula is dependent on the sample weights $\boldsymbol{w}$, one must re-estimate until a convergence criteria is met. This method is called iterative reweighted least squares (IRLS) (Rubin, 1983).

2.6.4 Calculating Posteriors in RVM for Classification

From Equation (2.25), one cannot take the integral as the distribution from Equation (2.24) is not Gaussian. Tipping (2001) is therefore using a Laplace approximation to a Gaussian distribution, explained in Section 2.6.3. Thus, one can approximate the posterior distribution in Equation (2.26) by the Laplace approximation

$$p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}) \simeq \mathscr{N}\big(\boldsymbol{w}|\boldsymbol{w}_{MP},\boldsymbol{\Sigma}\big) \tag{2.50}$$

with:

$$\boldsymbol{w}_{MP} = \boldsymbol{A}^{-1}\boldsymbol{\Phi}^{\top}(\boldsymbol{t} - \boldsymbol{\sigma}), \tag{2.51}$$

$$\boldsymbol{\Sigma} = (\boldsymbol{\Phi}^{\top}\boldsymbol{R}\boldsymbol{\Phi} + \boldsymbol{A})^{-1}. \tag{2.52}$$

The term $\boldsymbol{w}_{MP}$ is the solution when equating Equation (2.47) to zero, and $\boldsymbol{\Sigma}$ is the negative inverse of Equation (2.48). As the mean and covariance in Equation (2.51) is dependent on the sample weights $\boldsymbol{w}$, one must use the IRLS method to find the mean vector and covariance matrix at convergence by the Newton-Raphson update formula:

$$\boldsymbol{w}^{(new)} = \boldsymbol{w}^{(old)} + \boldsymbol{\Sigma}\nabla\mathcal{L}(\boldsymbol{w}^{(old)}). \tag{2.53}$$

2.6.5 Parameter Learning in RVM for Classification

The next problem Tipping (2001) had to face in the RVM classification case was that he were not able to integrate over the sample weights $\boldsymbol{w}$ to approach the marginal likelihood over the targets $\boldsymbol{t}$ given by

$$p(\boldsymbol{t}|\boldsymbol{\alpha}) = \int p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w}.$$

Using the result in Equation (2.33) and the fact that $p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha}) \propto p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha})$ in terms of $\boldsymbol{w}$, the integral above can be approximated by

$$p(\boldsymbol{t}|\boldsymbol{\alpha}) = \int p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w} \tag{2.54}$$

$$\simeq p(\boldsymbol{t}|\boldsymbol{w}_*)p(\boldsymbol{w}_*|\boldsymbol{\alpha})(2\pi)^{\frac{\mu}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}. \tag{2.55}$$

Thus, by inserting the distribution in Equation (2.2) and (2.24) with the converged value of $\boldsymbol{w}_{MP}$ one gets the update formula equivalent of Equation (4.10) in the regression case, that is

$$\alpha_i^{\text{new}} = \frac{\gamma_i}{\mu_{MP,i}^{*2}},$$

with $\gamma_i \equiv 1 - \alpha_i\Sigma_{ii}$, where $\boldsymbol{w}_{MP}$ and $\boldsymbol{\Sigma}$ are given by the converged values from Equation (2.51) and (2.52). The algorithm of RVM for classification is identical to the one for regression given by Algorithm 1 without having to deal with the noise-variance, and instead there is a little more work at step 3 and 12. At these steps, where the mean and covariance of the posterior distribution is calculated, one is using the IRLS method with the Newton-steps specified in Equation (2.53).

2.6.6 The Predictive Distribution

When predicting for categorical data, the predictive distribution is obtained using a different approach than for regression. This is not explained in Tipping (2001), so we will fill in the details from Bishop (2006). By marginalizing with respect to the posterior distribution $p(\boldsymbol{w}|\boldsymbol{t})$, the predictive distribution for class $\mathcal{C}_1$, given new input data $\boldsymbol{x}_*$ is

$$p(\mathcal{C}_1|\boldsymbol{\phi}(\boldsymbol{x}_*), \boldsymbol{t}_*) = \int p(\mathcal{C}_1|\boldsymbol{\phi}, \boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{t})d\boldsymbol{w} \simeq \int \sigma(\boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})q(\boldsymbol{w})d\boldsymbol{w}, \tag{2.56}$$

where $q(\boldsymbol{w})$ is the Laplace approximation from Equation (2.50). By defining $a = \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w}$, and using the fundamental property of the Dirac delta function in Equation (0.9), Bishop (2006) got

$$\sigma(\boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w}) = \int \sigma(a)\,\delta(a - \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})da,$$

where $\delta(\cdot)$ is the Dirac delta function (Dirac, 1958). Equation (2.56) can then be written as

$$p(\mathcal{C}_1|\boldsymbol{\phi}(\boldsymbol{x}_*), \boldsymbol{t}_*) \simeq \int\int \delta(a - \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})\,\sigma(a)\,da\,q(\boldsymbol{w})\,d\boldsymbol{w} \tag{2.57}$$

$$= \int \sigma(a)\,p(a)\,da, \tag{2.58}$$

where

$$p(a) = \int \delta(a - \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})\,q(\boldsymbol{w})\,d\boldsymbol{w}.$$

By looking closer at the distribution $p(a)$, one can use the moments to find a Laplace approximation. The first moment is given by

$$\mu_a = \mathbb{E}(a) = \int a\,dp(a) = \int a\,p(a)\,da = \int\int \delta(a - \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})\,q(\boldsymbol{w})\,d\boldsymbol{w}\,a\,da,$$

which by organizing with respect to a, this can be written as

$$\mu_a = \int\int \delta(a - \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})\,a\,da\,q(\boldsymbol{w})\,d\boldsymbol{w}.$$

By the fundamental property of the Dirac delta function (Dirac, 1958) in Equation (0.9) this is:

$$\mu_a = \int a\,q(\boldsymbol{w})\,d\boldsymbol{w} = \int \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w}\,q(\boldsymbol{w})\,d\boldsymbol{w}.$$

Writing $\boldsymbol{\phi}(\boldsymbol{x}_*)$ outside the integral and observing that the remaining expression is the definition of the mean value of $q(\boldsymbol{w})$, this is:

$$\mu_a = \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w}_{MP}.$$

In the same way, by writing $\boldsymbol{\phi}(\boldsymbol{x}_*)$ outside the integral and observing that the remaining expression is the definition of the variance, one get:

$$\begin{aligned}
\sigma_a^2 = \text{var}(a) &= \int p(a)\{a^2 - \mathbb{E}(a)^2\}da \\
&= \int\int \delta(a - \boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})\, da\, q(\boldsymbol{w})\, \{\boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})^2 - (\boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w}_{MP})^2\}\, d\boldsymbol{w} \\
&= \int q(\boldsymbol{w})\, \{(\boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w})^2 - (\boldsymbol{\phi}(\boldsymbol{x}_*)\boldsymbol{w}_{MP})^2\}\, d\boldsymbol{w} \\
&= \boldsymbol{\phi}(\boldsymbol{x}_*)^\top \boldsymbol{\Sigma}\boldsymbol{\phi}(\boldsymbol{x}_*),
\end{aligned}$$

where $\boldsymbol{w}_{MP}$ and $\boldsymbol{\Sigma}$ are given by Equation (2.51). Thus, using the Laplace approximation of $p(a)$, Equation (2.56) can be approximated by

$$p(\mathcal{C}_1|\boldsymbol{\phi}(\boldsymbol{x}_*, \boldsymbol{t}_*) = \int \sigma(a)p(a)da \simeq \int \sigma(a)\mathcal{N}(\mu_a, \sigma_a^2)da.$$

Now, using the close similarity between the sigmoid function $\sigma(a)$ and the probit function $\varphi(a)$ given by

$$\varphi(a) = \int_{-\infty}^{a} \mathcal{N}(\vartheta|0,1)d\vartheta,$$

Bishop (2006) is approximating $\sigma(a)$ by a horizontal scaling of the probit function, that is $\varphi(\lambda a)$. To obtain the best possible approximation, the value is chosen to be $\lambda^2 = \frac{\pi}{8}$ (LI, 2017). Hence, by using that

$$p(\mathcal{C}_1|\boldsymbol{\phi}(\boldsymbol{x}_*), \boldsymbol{t}_*) \simeq \int \varphi\left(\sqrt{\frac{\pi}{8}}a\right)\mathcal{N}(a|\mu_a, \sigma_a^2)da,$$

the predictive distribution for class $\mathcal{C}_1$ is given by

$$p(\mathcal{C}_1|\boldsymbol{\phi}(\boldsymbol{x}_*), \boldsymbol{t}_*) \simeq \varphi\left(\frac{\mu_a}{\sqrt{\frac{8}{\pi} + \sigma_a^2}}\right).$$

In practice, the estimate $\sigma(\boldsymbol{w}^\top \boldsymbol{\phi})$ is used for the mean value to make predictions. There are several examples of academic research where this estimate is used, like Tipping (2016) and the code associated with Jiang et al. (2019), and it seems to be a common approximation in Bayesian classification.

3 | Probabilistic Feature Selection and Classification Vector Machine

The RVM methods is sparse in sample size, but sometimes it is necessary to also have models that are sparse in terms of the number of features affecting the model. Jiang et al. (2019) developed such a method based on the probabilistic classification vector machine (PCVM) by Chen et al. (2009) that is similar to the RVM for classification. Due to the experiments by Jiang et al. (2019), their feature selective extension of the RVM for classification method is jointly selective in terms of both samples and features. In addition, their method seemed to be more accurate in the predictions than other similar methods. The next section will be a brief introduction to the PCVM method before the algorithm proposed by Jiang et al. (2019) is derived. However, the theory of Chen et al. (2009) and Jiang et al. (2019) is so far only derived for two class classification problems. Chapter 4 will give a suggested extension in the RVM regression case by a simliar approach.

3.1 Probabilistic Classification Vector Machines

The probabilistic classification vector machines (PCVM) by Chen et al. (2009) is a modification of the RVM for classification with the prior over the sample weights $\boldsymbol{w}$ changed to a left-truncated Gaussian distribution:

$$\begin{aligned} p(\boldsymbol{w}|\boldsymbol{\alpha}) &= \prod_{i=1}^{N} \mathcal{N}_t\big(w_i|0, \alpha_i^{-1}\big) \\ &= 2\prod_{i=1}^{N} \mathcal{N}\big(w_i|0, \alpha_i^{-1}\big) 1_{w_i \geq 0}(w_i). \end{aligned}$$

In the first line of the equation, $\mathcal{N}_t$ is denoting the left-truncated Gaussian distribution, and $1_{w_i \geq 0}(w_i)$ in the second line is an indicator function that is either 1 or 0. The weight w_0 is assigned a zero mean Gaussian distribution (Chen et al., 2009):

$$p(w_0|\alpha_0) = \mathcal{N}\big(w_0|0, \alpha_0^{-1}\big).$$

Altogether, the prior distribution is

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) = 2\mathscr{N}\big(w_0|0, \alpha_0^{-1}\big)\prod_{i=1}^{N}\mathscr{N}\big(w_i|0, \alpha_i^{-1}\big)1_{w_i\geq 0}(w_i), \tag{3.1}$$

which Chen et al. (2009) argued that made the final model more stable in prediction than the original RVM for classification, where a non-truncated zero mean Gaussian distribution is used. Except this modification of the prior, the sparse framework of the PCVM model is identical to the original RVM for classification.

3.2 Sparse Sample and Feature Selective Framework

In Section 1.2.1, we gave the theory behind the sparse framework with respect to the sample size. In this section, we will give a description of the sparse framework in the model developed by Jiang et al. (2019) which makes the models sparse both in terms of feature selective strength and in the ability to do sample size reduction. Based on the framework of the PCVM model, Jiang et al. (2019) extended the model to also be simultaneously feature selective. The model they proposed is named the probabilistic feature selection and classification vector machine (PFCVM). To achieve sparsity in terms of feature selection, the key principle for Jiang et al. (2019) was to define a new vector of feature weights, that is

$$\boldsymbol{\vartheta} = (\vartheta_1 \cdots \vartheta_p)^\top, \tag{3.2}$$

and a kernel basis function matrix $\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}$ that depends on the values of the feature weights $\boldsymbol{\vartheta}$. Their model was of the form

$$y = \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}. \tag{3.3}$$

Further, they are modifying the free parameter ϑ in the RBF kernel given by Equation (1.3) to be individual and possibly different for each feature weight ϑ_k. Thus, the basis function matrix from Equation (1.3) was modified, such that each element (i, j) is of the form:

$$\begin{aligned}\Phi_{\boldsymbol{\vartheta},ij} &= K(\boldsymbol{x}_i, \boldsymbol{x}_j)\\ &= \exp\Bigg\{-\sum_{k=1}^{P}\vartheta_k\big(\boldsymbol{x}_{ik} - \boldsymbol{x}_{jk}\big)^2\Bigg\}.\end{aligned} \tag{3.4}$$

The subscript $_{\boldsymbol{\vartheta}}$ is used several times, and it is always denoting that the feature weights $\boldsymbol{\vartheta}$ from Equation (3.2) is included in all the kernel functions that appears in the original expression, like in Equation (3.4). Regarding the sparseness with respect to the features, one can see from Equation (3.4) that if a feature weight ϑ_k is zero, the corresponding feature in element number k in all the input vectors given by Equation (0.3) does not contribute to the sum in the kernel function. Hence, using an appropriate sparse prior on the feature weights, one can avoid that an irrelevant feature ϑ_k will affect the predictions. The likelihood over the targets $\boldsymbol{t}$ in the PFCVM model is given as in the RVM for classification case in Equation (2.24), but with the feature weights $\boldsymbol{w}$ included in the kernel function matrix. For the prior distribution over the sample weights $\boldsymbol{w}$ and feature weights $\boldsymbol{\vartheta}$ Jiang et al. (2019) are using the left-truncated zero mean Gaussian distribution from the PCVM method, derived in Equation (3.1). The same approach is used for the prior distribution over the feature weights $\boldsymbol{\vartheta}$, and a left-truncated zero mean Gaussian distribution is assigned:

$$\begin{aligned} p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) &= \prod_{k=1}^{P} \mathcal{N}_t(\vartheta_k|0, \beta_k^{-1}) \\ &= 2\prod_{k=1}^{P} \mathcal{N}(\vartheta_k|0, \beta_k^{-1}) \cdot 1_{\vartheta_k>0}(\vartheta_k) \\ &= 2 \cdot (2\pi)^{-\frac{p}{2}}|\boldsymbol{B}|^{\frac{1}{2}} \exp\left\{ -\frac{1}{2}\boldsymbol{\vartheta}^\top \boldsymbol{B}\boldsymbol{\vartheta}\right\} \cdot 1_{\vartheta_k>0}(\vartheta_k), \end{aligned} \tag{3.5}$$

where each of the hyperparameters corresponding to

$$\boldsymbol{B} = \operatorname{diag}(\beta_1, \ldots, \beta_p)^\top$$

is gamma distributed. That is:

$$\beta_i \sim \operatorname{Gamma}(\beta_i|e, f). \tag{3.6}$$

By using this prior on the feature weights $\boldsymbol{\vartheta}$, they are forcing the parameters to be positive, as the free parameter in the RBF kernel function should not be negative (Krishnapuram et al., 2004). Jiang et al. (2019) is then making the hyperparamer $\boldsymbol{\beta}$ uninformative by fixing the hyper hyperparameters to be $e = f = 10^{-4}$. This is similar to what Tipping (2001) did in the original RVM case, and makes the hyperparameters behave like a uniform distribution. By modifying the kernel basis functions like shown in Equation (3.3) and using the sparse prior from Equation (3.5) on the feature weights $\boldsymbol{\vartheta}$, they were able to create a learning procedure only choosing the most informative features to affect the predictions. The Bayesian approach of Jiang et al. (2019) is similar to the one of Tipping (2001) used in the RVM case, but where the posterior distribution also includes the feature weights $\boldsymbol{\vartheta}$, and the hyperparameter $\boldsymbol{\beta}$ for these feature weights. The posterior distribution over the parameters is given by

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\boldsymbol{t}) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\alpha}, \boldsymbol{\beta})p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{p(\boldsymbol{t})},$$

and the predictive distribution is

$$p(t^*|\boldsymbol{t}) = \int p(t^*|\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\alpha}, \boldsymbol{\beta})p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\boldsymbol{t})\, d\boldsymbol{w}\, d\boldsymbol{\vartheta}\, d\boldsymbol{\alpha}\, d\boldsymbol{\beta}.$$

Further, the posterior distribution over all the unknown parameters in the decomposed form, corresponding to Equation (2.4) in RVM, is given by

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\boldsymbol{t}) = p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{t})p(\boldsymbol{\alpha}, \boldsymbol{\beta}|\boldsymbol{t}). \tag{3.7}$$

Jiang et al. (2019) is further writing the simultaneous posterior distribution over the sample weights $\boldsymbol{w}$ and the feature weights $\boldsymbol{\vartheta}$ in the first term of Equation (3.7) as

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta})p(\boldsymbol{w}|\boldsymbol{\alpha})p(\boldsymbol{\vartheta}|\boldsymbol{\beta})}{p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta})}. \tag{3.8}$$

With this framework established, Jiang et al. (2019) used a Laplace approximation to a Gaussian distribution for the simultaneous posterior distribution over the weights $\boldsymbol{w}$ and $\boldsymbol{\vartheta}$ given by Equation (3.8), as it is not possible to calculate this distribution analytically.

3.3 Calculating Posteriors

Based on a similar approach as Tipping (2001), Jiang et al. (2019) first calculated the simultaneous posterior distribution over both weights. As the likelihood over the targets $\boldsymbol{t}$ is Bernoulli distributed, they were not able to find an analytical solution. Thus, they used Laplace's approximation of the distribution in (3.8), as described in Section 2.6.2, with respect to each of the weight parameters $\boldsymbol{w}$ (Mohsenzadeh et al. (2013), Mohsenzadeh et al. (2016)). By first taking the logarithm of the joint posterior over the sample weights $\boldsymbol{w}$ and the feature weights $\boldsymbol{\vartheta}$, given by Equation (3.8), Jiang et al. (2019) got

$$\begin{aligned} \ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}) =& \ln p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}) + \ln p(\boldsymbol{w}|\boldsymbol{\alpha}) \\ &+ \ln p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) - \ln p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}), \end{aligned} \tag{3.9}$$

where $p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta})$, $p(\boldsymbol{w}|\boldsymbol{\alpha})$ and $p(\boldsymbol{\vartheta}|\boldsymbol{\beta})$ are given by Equation (2.24) with the feature weights $\boldsymbol{\vartheta}$ included, (3.1) and (3.5) respectively. Only considering the terms that involves $\boldsymbol{\vartheta}$, this is

$$\mathcal{L}(\boldsymbol{\vartheta}) = \sum_{i=1}^{N} \Big[t_i \ln \sigma_i + (1-t_i) \ln (1-\sigma_i) \Big] - \frac{1}{2} \boldsymbol{\vartheta}^\top \boldsymbol{B} \boldsymbol{\vartheta} + \sum_{k=1}^{P} \ln 1_{\vartheta_k \geq 0}(\vartheta_k) + \text{const.}. \tag{3.10}$$

It is not possible to take the derivative of the indicator function in Equation (2.24), and Jiang et al. (2019) used a parameterized sigmoid approximation for the indicator function, which they were able to differentiate. Figure 3.1 is illustrating how the sigmoid function $\sigma(cx)$ is a good approximation for the indicator function.

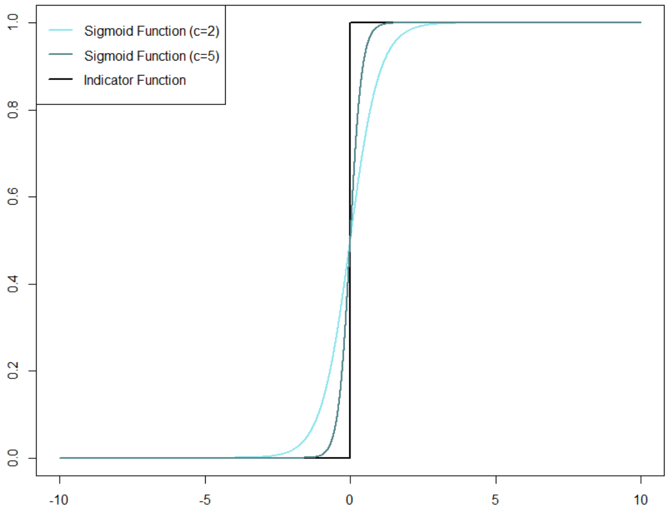

Figure 3.1: Comparison of the indicator function in black against the sigmoid approximation $\sigma(cx)$ in blue with different scales c.

By differentiating the log posterior in Equation (3.10) with respect to $\boldsymbol{\vartheta}$, they got

$$\frac{\partial \mathcal{L}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} = -\boldsymbol{B}\boldsymbol{\vartheta} + \boldsymbol{D}^\top (\boldsymbol{t} - \boldsymbol{\sigma}) + \boldsymbol{k}_{\boldsymbol{\vartheta}},$$

where

$$\boldsymbol{k}_{\boldsymbol{\vartheta}} = \big(\lambda(1-\sigma(\lambda\vartheta_1)), \ldots, \lambda(1-\sigma(\lambda\vartheta_P)) \big)^\top, \tag{3.11}$$

and $\boldsymbol{D} = \frac{\partial(\boldsymbol{\Phi}_{\boldsymbol{\vartheta}} \boldsymbol{w})}{\partial \boldsymbol{\vartheta}}$, with dimension $N \times P$. Using an RBF kernel function of the form (3.4), each element of $\boldsymbol{D}$ is given by

$$\begin{aligned} D_{i,k} &= \frac{\partial(\boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_i)\boldsymbol{w})}{\partial \vartheta_k} \\ &= -\sum_{j=1}^{N} w_j \boldsymbol{\phi}_{\vartheta}(\boldsymbol{x}_i, \boldsymbol{x}_j)(\boldsymbol{x}_{ik} - \boldsymbol{x}_{jk})^2, \end{aligned} \tag{3.12}$$

with $\boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_i)$ from Equation (1.2), where the feature weights $\boldsymbol{\vartheta}$ is included. By now equating (4.5) to zero, Jiang et al. (2019) got by the Laplace approximation a vector of mean values with respect to $\boldsymbol{\vartheta}$ given by

$$\boldsymbol{\vartheta}_{MP} = \boldsymbol{B}^{-1}\big(\boldsymbol{D}^\top(\boldsymbol{t} - \boldsymbol{\sigma}) + \boldsymbol{k}_{\boldsymbol{\vartheta}}\big). \tag{3.13}$$

By taking the second derivative of (3.10), that is finding the Hessian matrix, they got

$$\frac{\partial^2 \mathcal{L}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}^2} = -\boldsymbol{B} - \boldsymbol{D}^\top \boldsymbol{C}\boldsymbol{D} + \boldsymbol{E} - \boldsymbol{O}_{\boldsymbol{\vartheta}}, \tag{3.14}$$

where each element of the matrix $\boldsymbol{E} = \frac{\partial \boldsymbol{D}^\top}{\partial \boldsymbol{\vartheta}}(\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w})$ is

$$\begin{aligned} E_{i,k} &= \sum_{l=1}^{N} \frac{\partial D_{l,i}}{\partial \vartheta_k}\Big[t_l - \sigma\big(\boldsymbol{\phi}(\boldsymbol{x}_l)\boldsymbol{w}\big)\Big] \\ &= \sum_{l=1}^{N} \Big[t_l - \sigma\big(\boldsymbol{\phi}(\boldsymbol{x}_l)\boldsymbol{w}\big)\Big] \sum_{j=1}^{N} w_j \boldsymbol{\phi}_\vartheta(\boldsymbol{x}_l, \boldsymbol{x}_j)(\boldsymbol{x}_{li} - \boldsymbol{x}_{ji})^2(\boldsymbol{x}_{lk} - \boldsymbol{x}_{jk})^2). \end{aligned} \tag{3.15}$$

The term $\boldsymbol{O}_{\boldsymbol{\vartheta}}$ in Equation (3.14) is the second derivative of the sigmoid approximation given by

$$\boldsymbol{O}_{\boldsymbol{\vartheta}} = \operatorname{diag}\big(\lambda^2\sigma(\lambda\vartheta_1)(1 - \sigma(\lambda\vartheta_1)), \ldots, \lambda^2\sigma(\lambda\vartheta_P)(1 - \sigma(\lambda\vartheta_P))\big). \tag{3.16}$$

The term $\boldsymbol{C}$ is

$$\boldsymbol{C} = \operatorname{diag}\big((1 - y_1)y_1, \ldots, (1 - y_N)y_N\big).$$

The approximate covariance matrix with respect to $\boldsymbol{\vartheta}$ is the negative inverse of the Hessian matrix:

$$\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}} = \Big(\boldsymbol{B} + \boldsymbol{D}^\top\boldsymbol{C}\boldsymbol{D} - \boldsymbol{E} + \boldsymbol{O}_{\boldsymbol{\vartheta}}\Big)^{-1}. \tag{3.17}$$

To make later calculations easier they are simplifying the notation in both the mean vector and covariance matrix in (4.6) and (4.7) by

$$\boldsymbol{\vartheta}_{MP} = \boldsymbol{B}^{-1}\boldsymbol{\epsilon}_{\boldsymbol{\vartheta}} \qquad \text{and} \qquad \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}} = (\boldsymbol{B} + \boldsymbol{H}_{\boldsymbol{\vartheta}})^{-1}, \tag{3.18}$$

where $\boldsymbol{\epsilon}_{\boldsymbol{\vartheta}} = \big(\boldsymbol{D}^\top(\boldsymbol{t} - \boldsymbol{\sigma}) + \boldsymbol{k}_{\boldsymbol{\vartheta}}\big)$ and $\boldsymbol{H}_{\boldsymbol{\vartheta}} = \boldsymbol{D}^\top\boldsymbol{C}\boldsymbol{D} - \boldsymbol{E} + \boldsymbol{O}_{\boldsymbol{\vartheta}}$ are independent of $\boldsymbol{\vartheta}$. In the same way, by only considering the terms of Equation (3.9) that includes the sample weights $\boldsymbol{w}$, Jiang et al. (2019) got

$$\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^{N}\Big[t_i \ln y\sigma_i + (1 - t_i)\ln 1 - \sigma_i\Big] - \frac{1}{2}\boldsymbol{w}^\top\boldsymbol{A}\boldsymbol{w} + \sum_{i=1}^{N}\ln 1_{w_i \geq 0}(w_i) + \text{const.}.$$

By differentiating this once and twice with respect to $\boldsymbol{w}$, one gets

$$\frac{\partial \mathcal{L}(\boldsymbol{w})}{\partial \boldsymbol{w}} = -\boldsymbol{A}\boldsymbol{w} + \boldsymbol{\Phi_{\vartheta}}^{\top}(\boldsymbol{t} - \boldsymbol{\sigma}) + \boldsymbol{k}_w,$$

$$\frac{\partial^2 \mathcal{L}(\boldsymbol{w})}{\partial \boldsymbol{w}^2} = -\boldsymbol{A} - \boldsymbol{\Phi_{\vartheta}}^{\top}\boldsymbol{C}\boldsymbol{\Phi_{\vartheta}} - \boldsymbol{O}_w.$$

In these equations

$$\boldsymbol{k_w} = \big(0, \lambda(1 - \sigma(\lambda w_1)), \ldots, \lambda(1 - \sigma(\lambda w_N))\big)^{\top}, \tag{3.19}$$

and

$$\boldsymbol{O_w} = \operatorname{diag}\big(0, \lambda^2\sigma(\lambda w_1)(1 - \sigma(\lambda w_1)), \ldots, \lambda^2\sigma(\lambda w_N)(1 - \sigma(\lambda w_N))\big). \tag{3.20}$$

Hence, the mean and covariance of the Laplace approximation with respect to the sample weights $\boldsymbol{w}$ is

$$\boldsymbol{\Sigma_w} = \Big(\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^{\top}\boldsymbol{C}\boldsymbol{\Phi_{\vartheta}} + \boldsymbol{A} + \boldsymbol{O_w}\Big)^{-1}, \tag{3.21}$$

$$\boldsymbol{w}_{MP} = \boldsymbol{A}^{-1}\Big(\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^{\top}(\boldsymbol{t} - \boldsymbol{\sigma}) + \boldsymbol{k_w}\Big), \tag{3.22}$$

which can be simplified to

$$\boldsymbol{w}_{MP} = \boldsymbol{A}^{-1}\boldsymbol{\epsilon_w} \qquad \text{and} \qquad \boldsymbol{\Sigma_w} = (\boldsymbol{A} + \boldsymbol{H_w})^{-1}, \tag{3.23}$$

where $\boldsymbol{\epsilon_w} = \big(\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^{\top}(\boldsymbol{t} - \boldsymbol{\sigma}) + \boldsymbol{k_w}\big)$ and $\boldsymbol{H_w} = \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^{\top}\boldsymbol{C}\boldsymbol{\Phi_{\vartheta}} + \boldsymbol{O_w}$ are independent of $\boldsymbol{\vartheta}$. All together, the simultaneous posterior distribution over the sample and feature weights are given by the Laplace approximated distribution of the form (Mohsenzadeh et al. (2016), Jiang et al. (2019)):

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \approx \mathscr{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{\vartheta}}) \cdot \mathscr{N}(\boldsymbol{w}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{w}}). \tag{3.24}$$

As the simultaneous distribution in Equation (3.24) is not analytical they had to use the IRLS method described in Section 2.6.3, which gives:

$$\boldsymbol{w}^{new} = \boldsymbol{w}^{old} + \boldsymbol{\Sigma}^{\boldsymbol{w}}\nabla\mathcal{L}(\boldsymbol{w}^{old}),$$

$$\boldsymbol{\vartheta}^{new} = \boldsymbol{\vartheta}^{old} + \boldsymbol{\Sigma}^{\boldsymbol{\vartheta}}\nabla\mathcal{L}(\boldsymbol{\vartheta}^{old}).$$

3.4 Parameter Learning

As in the original RVM by Tipping et al. (2003) the problem of maximizing the posterior distribution over all parameters boils down to maximizing the marginal likelihood $p(\boldsymbol{\alpha}, \boldsymbol{\beta}|\boldsymbol{t})$ in the second expression of Equation (3.7). This term is not possible to calculate analytically, and one must approximate it by finding the most probable values of the parameters $\boldsymbol{\alpha}_{MP}$ and $\boldsymbol{\beta}_{MP}$. Thus, Jiang et al. (2019) approximated this likelihood by

$$\begin{aligned} p(\boldsymbol{\alpha}, \boldsymbol{\beta}|\boldsymbol{t}) &= \frac{p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta},)p(\boldsymbol{\alpha})p(\boldsymbol{\beta})}{p(\boldsymbol{t})} \\ &\propto p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}), \end{aligned}$$

as the denominator will be uninformative in terms of maximization with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and as $p(\boldsymbol{\alpha})$ and $p(\boldsymbol{\beta})$ are uniform in practice. By rewriting Equation (3.8), they got the posterior distribution over the hyperparamaters

$$p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta})p(\boldsymbol{w}|\boldsymbol{\alpha})p(\boldsymbol{\vartheta}|\boldsymbol{\beta})}{p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta})}.$$

Taking the logarithm, only considering the terms that involves $\boldsymbol{\alpha}$, gives the log posterior over the hyperparameters

$$\begin{aligned} \mathcal{L}(\boldsymbol{\alpha}) &= \ln p(\boldsymbol{w}|\boldsymbol{\alpha}) - \ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) \\ &= \frac{1}{2}\ln|\boldsymbol{A}| - \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{A}\boldsymbol{w} + \frac{1}{2}\ln|\boldsymbol{\Sigma}^{\boldsymbol{w}}| \\ &\quad + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{MP})^\top (\boldsymbol{\Sigma}^{\boldsymbol{w}})^{-1}(\boldsymbol{w} - \boldsymbol{w}_{MP}) + \text{const.}. \end{aligned}$$

In a similar manner, only considering the terms of Equation (4.9) that involves $\boldsymbol{\beta}$, Jiang et al. (2019) got

$$\begin{aligned} \mathcal{L}(\boldsymbol{\beta}) &= \ln p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) - \ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) \\ &= \frac{1}{2}\ln|\boldsymbol{B}| - \frac{1}{2}\boldsymbol{\vartheta}^\top \boldsymbol{B}\boldsymbol{\vartheta} + \frac{1}{2}\ln|\boldsymbol{\Sigma}^{\boldsymbol{\vartheta}}| \\ &\quad + \frac{1}{2}(\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_{MP})^\top (\boldsymbol{\Sigma}^{\boldsymbol{\vartheta}})^{-1}(\boldsymbol{\vartheta} - \boldsymbol{\vartheta}_{MP}) + \text{const.}. \end{aligned} \tag{3.25}$$

By using the simplifications from Equation (3.18) and (3.23) this is:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\alpha}) &= \frac{1}{2}\ln|\boldsymbol{A}| + \frac{1}{2}\ln|\boldsymbol{\Sigma_w}| - \frac{1}{2}\boldsymbol{w}^\top\big(\boldsymbol{A} - (\boldsymbol{A} + \boldsymbol{H_w})\big)\boldsymbol{w} \\ &\quad - \boldsymbol{\vartheta}^\top \boldsymbol{H_\vartheta}\boldsymbol{\vartheta}_{MP} + \frac{1}{2}\boldsymbol{\vartheta}_{MP}^\top \boldsymbol{H_\vartheta}\boldsymbol{\vartheta}_{MP} + \frac{1}{2}\boldsymbol{\vartheta}_{MP}^\top \boldsymbol{B}\boldsymbol{\vartheta}_{MP} \\ &= \frac{1}{2}\ln|\boldsymbol{A}| + \frac{1}{2}\ln|\boldsymbol{\Sigma_w}| + \frac{1}{2}\boldsymbol{\epsilon_w}^\top \boldsymbol{A}^{-1}\boldsymbol{\epsilon_w} + \frac{1}{2}(\boldsymbol{w}_{MP} - \boldsymbol{w}^\top)\boldsymbol{H_w}\boldsymbol{w}_{MP} \qquad (3.26)\end{aligned}$$

and

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}) &= \frac{1}{2}\ln|\boldsymbol{B}| + \frac{1}{2}\ln|\boldsymbol{\Sigma}^{\boldsymbol{\vartheta}}| - \frac{1}{2}\boldsymbol{\vartheta}^\top\big(\boldsymbol{B} - (\boldsymbol{B} + \boldsymbol{H_\vartheta})\big)\boldsymbol{\vartheta} \\ &\quad - \boldsymbol{\vartheta}^\top \boldsymbol{H_\vartheta}\boldsymbol{\vartheta}_{MP} + \frac{1}{2}\boldsymbol{\vartheta}_{MP}^\top \boldsymbol{H_\vartheta}\boldsymbol{\vartheta}_{MP} + \frac{1}{2}\boldsymbol{\vartheta}_{MP}^\top \boldsymbol{B}\boldsymbol{\vartheta}_{MP} \\ &= \frac{1}{2}\ln|\boldsymbol{B}| + \frac{1}{2}\ln|\boldsymbol{\Sigma_\vartheta}| + \frac{1}{2}\boldsymbol{\epsilon_\vartheta}^\top \boldsymbol{B}^{-1}\boldsymbol{\epsilon_\vartheta} + \frac{1}{2}(\boldsymbol{\vartheta}_{MP} - \boldsymbol{\vartheta}^\top)\boldsymbol{H_\vartheta}\boldsymbol{\vartheta}_{MP} \qquad (3.27)\end{aligned}$$

In the deduction in the appendix of Jiang et al. (2019), the last term in Equation (3.27) disappear, even though it is not clear why. In practice one often use a maximum *a posteriori* (MAP), that is the mode of the posterior distribution, to estimate for the mean $\boldsymbol{\vartheta}_{MP}$, and therefore one can use a heuristic argument about the last term behaving like a constant. Thus, Equation(3.26) and (3.27) can be approximated by

$$\mathcal{L}(\boldsymbol{\alpha}) = \frac{1}{2}\ln|\boldsymbol{A}| + \frac{1}{2}\ln|\boldsymbol{\Sigma_w}| + \frac{1}{2}\boldsymbol{\epsilon_w}^\top \boldsymbol{A}^{-1}\boldsymbol{\epsilon_w}$$

and

$$\mathcal{L}(\boldsymbol{\beta}) = \frac{1}{2}\ln|\boldsymbol{B}| + \frac{1}{2}\ln|\boldsymbol{\Sigma_\vartheta}| + \frac{1}{2}\boldsymbol{\epsilon_\vartheta}^\top \boldsymbol{B}^{-1}\boldsymbol{\epsilon_\vartheta}. \qquad (3.28)$$

By differentiating each of these equations and equating to zero they got

$$\alpha_i^{\text{new}} = \frac{\gamma_i^{\boldsymbol{w}}}{(w_{i,MP}^{\boldsymbol{w}})^2} \quad \text{and} \quad \beta_i^{\text{new}} = \frac{\gamma_i^{\boldsymbol{\vartheta}}}{(\vartheta_{i,MP}^{\boldsymbol{\vartheta}})^2}, \qquad (3.29)$$

where $\gamma_i^{\boldsymbol{w}} = 1 - \alpha_i \Sigma_{\boldsymbol{w},ii}$ and $\gamma_i^{\boldsymbol{\vartheta}} = 1 - \beta_i \Sigma_{\boldsymbol{\vartheta},ii}$.

3.5 The Predictive Distribution and the Algorithm of PFCVM

It is not clear in Jiang et al. (2019) how they are predicting for new input data. However, in theory the predictions should follow the same approach as described in Section 2.6.6. When predicting for categorical data, the distribution is obtained using a different approach than used in the regression case. This is not explained in Jiang et al. (2019), so we will apply the theory from Bishop (2006).

By marginalizing with respect to the posterior distribution $p(\boldsymbol{w}, \boldsymbol{\vartheta} | \boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ from Equation (3.24), the predictive distribution for class $\mathcal{C}_1$ is

$$\begin{aligned} p(\mathcal{C}_1 | \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*, \boldsymbol{t}_*) &= \int p(\mathcal{C}_1 | \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*), \boldsymbol{w}) p(\boldsymbol{w}, \boldsymbol{\vartheta} | \boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}) d\boldsymbol{w} d\boldsymbol{\vartheta} \\ &\simeq \int \sigma\big(\boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*)\boldsymbol{w}\big) \mathscr{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{\vartheta}}) \cdot \mathscr{N}(\boldsymbol{w}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{w}}) d\boldsymbol{w} d\boldsymbol{\vartheta} \\ &= \int \sigma\big(\boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*)\boldsymbol{w}\big) \mathscr{N}(\boldsymbol{w}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{w}}) d\boldsymbol{w}, \end{aligned}$$

where $\mathscr{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{\vartheta}})$ and $\mathscr{N}(\boldsymbol{w}_{MP}, \boldsymbol{\Sigma}^{\boldsymbol{w}})$ is the Laplace approximation given by Equation (2.50), (3.13), (3.17), (3.22) and (3.21). By the same reasoning as in Section 2.6.6, we get

$$\mu_a = \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*)\boldsymbol{w}_{MP} \qquad \text{and} \qquad \sigma_a^2 = \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*)^\top \boldsymbol{\Sigma}_{\boldsymbol{w}} \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*), \tag{3.30}$$

where $\boldsymbol{w}_{MP}$ and $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ is given by (3.21) and (3.22). Thus, the predictive distribution for class $\mathcal{C}_1$ is given by

$$p(\mathcal{C}_1 | \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*), \boldsymbol{t}_*) \simeq \varphi\left(\frac{\mu_a}{\sqrt{\frac{8}{\pi} + \sigma_a^2}}\right),$$

where μ_a and σ_a^2 is from Equation (3.30).

Algorithm 3 Probabilistic Feature Selection and Classification Vector Machine (PFCVM)

```
Initialize $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ to some reasonable values
Compute $\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\vartheta}_{\boldsymbol{MP}}$, $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ and $\boldsymbol{w}_{MP}$ using IRLS method
while convergence criteria are not met do
    for all $\alpha_i$ in $\boldsymbol{\alpha}$ do
        if $\alpha_i > \alpha_{\text{Thresh}}$ then
            delete $\boldsymbol{\phi}_i$ and $\alpha_i$
        end if
    end for
    for all $\beta_k$ in $\boldsymbol{\beta}$ do
        if $\beta_k > \beta_{\text{Thresh}}$ then
            delete feature number k and hence $\beta_i$
        end if
    end for
    Update $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{w}}$, $\boldsymbol{\vartheta}_{MP}$ and $\boldsymbol{w}_{MP}$ using IRLS method, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$
end while
```

This is the theoretical reasoning, but as described in Section 2.6.6 Jiang et al. (2019) is using the estimate $\sigma(\boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*)\boldsymbol{w}_{MP})$ to predict for new input data. Using the theory deduced above, the algorithm of the PFCVM method is given by Algorithm 3 (Jiang et al., 2019).

4 | Dimensionality Reducing Relevance Vector Machine for Regression

This chapter gives an extension of the RVM for regression by Tipping (2001), utilizing the approach of Jiang et al. (2019) that is outlined in Chapter 3. The proposed method is called the dimensionality reducing relevance vector machine (DRVM).

4.1 Sparse Sample and Feature Selective Framework

The sparse framework in the proposed DRVM model for feature selection in the RVM for regression framework is similar to the one for classification given in Section 3. The main difference is due to the likelihood of the targets $\boldsymbol{t}$ being Gaussian, given by Equation (2.2). In addition, we are using the original not-truncated zero mean Gaussian prior on the distribution over the sample weights $\boldsymbol{w}$, from Equation (1.5) with the independent feature weights $\boldsymbol{\vartheta}$ included in the kernel function matrix:

$$p(\boldsymbol{t}|\boldsymbol{w},\boldsymbol{\vartheta},\sigma^2) = (2\pi\sigma^2)^{-\frac{N}{2}} \exp\left\{ -\frac{1}{2\sigma^2}||\boldsymbol{t}-\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}||^2 \right\}. \tag{4.1}$$

The Bayesian approach in the dimensionality reducing RVM for regression must be similar to the one used by Jiang et al. (2019), but where the noise-variance is included as the likelihood of the targets $\boldsymbol{t}$ are Gaussian distributed. That is the posterior distribution over all hyperparameters is given by

$$p(\boldsymbol{w},\boldsymbol{\vartheta},\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2|\boldsymbol{t}) = \frac{p(\boldsymbol{t}|\boldsymbol{w},\boldsymbol{\vartheta},\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2)p(\boldsymbol{w},\boldsymbol{\vartheta},\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2)}{p(\boldsymbol{t})},$$

with the predictive distribution

$$p(\boldsymbol{t}_*|\boldsymbol{t}) = \int p(\boldsymbol{t}_*|\boldsymbol{w},\boldsymbol{\vartheta},\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2)p(\boldsymbol{w},\boldsymbol{\vartheta},\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2|\boldsymbol{t})\, d\boldsymbol{w}\, d\boldsymbol{\vartheta}\, d\boldsymbol{\alpha}\, d\boldsymbol{\beta}\, d\sigma^2.$$

The decomposed posterior distribution, is given by

$$p(\boldsymbol{w},\boldsymbol{\vartheta},\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2|\boldsymbol{t}) = p(\boldsymbol{w},\boldsymbol{\vartheta}|\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2,\boldsymbol{t})p(\boldsymbol{\alpha},\boldsymbol{\beta},\sigma^2|\boldsymbol{t}),$$

and the simultaneous posterior distribution over the sample weights $\boldsymbol{w}$ and the feature weights $\boldsymbol{\vartheta}$, is given by

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha})p(\boldsymbol{\vartheta}|\boldsymbol{\beta})}{p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)}. \tag{4.2}$$

4.2 Calculating Posteriors

From Equation (4.2), the log posterior distribution over the sample weights $\boldsymbol{w}$ and the feature weights $\boldsymbol{\vartheta}$ is given by

$$\begin{aligned} \ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) =& \ln p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2) + \ln p(\boldsymbol{w}|\boldsymbol{\alpha}) \\ &+ \ln p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) - \ln p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2), \end{aligned} \tag{4.3}$$

where $p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2)$, $p(\boldsymbol{w}|\boldsymbol{\alpha})$ and $p(\boldsymbol{\vartheta}|\boldsymbol{\beta})$ are given by Equation (4.1), (2.2) and (3.5) respectively. Only considering the terms that involves $\boldsymbol{\vartheta}$, this is

$$\begin{aligned} \mathcal{L}(\boldsymbol{\vartheta}) =& -\frac{1}{2}\Big[\sigma^{-2}||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}||^2 + \boldsymbol{\vartheta}^\top \boldsymbol{B}\boldsymbol{\vartheta}\Big] + \sum_{k=1}^{P} \ln 1_{\vartheta_k \geq 0}(\vartheta_k) + \text{const.} \\ =& \sigma^{-2}\boldsymbol{t}^\top \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w} - \frac{1}{2}\sigma^{-2}\boldsymbol{w}^\top \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^\top \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w} - \frac{1}{2}\boldsymbol{\vartheta}^\top \boldsymbol{B}\boldsymbol{\vartheta} + \sum_{k=1}^{P} \ln 1_{\vartheta_k \geq 0}(\vartheta_k) + \text{const.}, \end{aligned} \tag{4.4}$$

As the posterior distribution over the feature weights $\boldsymbol{\vartheta}$ depends on the indicator function, the posterior distribution does not have an analytical solution. Thus, we are using the Laplace approximation on each of the weight parameters, and the indicator function is approximated by a sigmoid function as used in the PFCVM method by Jiang et al. (2019). With respect to the feature weights $\boldsymbol{\vartheta}$, the first derivative is

$$\frac{\partial \mathcal{L}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} = -\boldsymbol{\vartheta}\boldsymbol{B} + \sigma^{-2}\boldsymbol{D}^\top(\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}) + \boldsymbol{k}_{\boldsymbol{\vartheta}}, \tag{4.5}$$

where $\boldsymbol{D}$ is given by Equation (3.12) and $\boldsymbol{k}_{\boldsymbol{\vartheta}}$ is given by Equation (3.11). By now equating (4.5) to zero, we get the mean vector:

$$\boldsymbol{\vartheta}_{MP} = \boldsymbol{B}^{-1}\big(\sigma^{-2}\boldsymbol{D}^\top(\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}) + \boldsymbol{k}_{\boldsymbol{\vartheta}}\big). \tag{4.6}$$

The Hessian matrix of (4.4) is given by

$$\frac{\partial^2 \mathcal{L}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}^2} = -\boldsymbol{B} - \sigma^{-2}(\boldsymbol{D}^\top \boldsymbol{D} - \boldsymbol{E}) - \boldsymbol{O}_{\boldsymbol{\vartheta}},$$

where each element of $\boldsymbol{E}$ is given by Equation (3.15) with $\boldsymbol{\phi}(\boldsymbol{x}_l)\boldsymbol{w}$ instead of the sigmoid function $\sigma\big(\boldsymbol{\phi}(\boldsymbol{x}_l)\boldsymbol{w}\big)$, and the term $\boldsymbol{O}_{\boldsymbol{\vartheta}}$ is from Equation (3.16). Thus, the covariance matrix with respect to $\boldsymbol{\vartheta}$ is given by

$$\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}} = \Big(\boldsymbol{B} + \sigma^{-2}\big[\boldsymbol{D}^\top\boldsymbol{D} - \boldsymbol{E}\big] + \boldsymbol{O}_{\boldsymbol{\vartheta}}\Big)^{-1}. \tag{4.7}$$

Similarly, as Jiang et al. (2019), we are simplifying the expressions in Equation (4.6) and (4.7). These simplifications are identical to Equation (3.18), but with:

$$\boldsymbol{\epsilon}_{\boldsymbol{\vartheta}} = \big(\sigma^{-2}\boldsymbol{D}^\top(\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}) + \boldsymbol{k}_{\boldsymbol{\vartheta}}\big),$$

$$\boldsymbol{H}_{\boldsymbol{\vartheta}} = \sigma^{-2}\big[\boldsymbol{D}^\top\boldsymbol{D} - \boldsymbol{E}\big] + \boldsymbol{O}_{\boldsymbol{\vartheta}}.$$

In the same way, by only considering the terms of Equation (4.3) that includes the sample weights $\boldsymbol{w}$, the likelihood is given by

$$\mathcal{L}(\boldsymbol{w}) = -\frac{1}{2}\Big[\sigma^{-2}||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}||^2 + \boldsymbol{w}^\top\boldsymbol{A}\boldsymbol{w}\Big] + \text{const.}.$$

Differentiating this once and twice with respect to the sample weights $\boldsymbol{w}$ gives:

$$\boldsymbol{\Sigma}_{\boldsymbol{w}} = (\sigma^{-2}\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^\top\boldsymbol{\Phi}_{\boldsymbol{\vartheta}} + \boldsymbol{A})^{-1},$$

$$\boldsymbol{\mu}_{\boldsymbol{w}} = \sigma^{-2}(\boldsymbol{\Sigma}_{\boldsymbol{w}})^{-1}\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^\top\boldsymbol{t}.$$

In these equations $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ and $\boldsymbol{\mu}_{\boldsymbol{w}}$ is the mean vector and covariance matrix from the original RVM, given by Equation (2.6) and (2.7), but where $\boldsymbol{\Phi}$ is substituted with $\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}$. Thus, the Laplace approximation with respect to $\boldsymbol{w}$ is exact, which is expected as both the likelihood of the targets $\boldsymbol{t}$ and the prior over the sample weights $\boldsymbol{w}$ is Gaussian.

The Laplace approximation of the posterior distribution in Equation (4.3) is thus given by

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) \approx \mathscr{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}) \cdot \mathscr{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}}), \tag{4.8}$$

where $\boldsymbol{\vartheta}_{MP}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}$ is given by Equation (4.6) and (4.7). In contrast to the PFCVM model, the last term in Equation (4.8) is exact. This, means that the mode with respect to the sample weights $\boldsymbol{w}$ can be found analytically, while we have to use the IRLS method to find the mode with respect to $\boldsymbol{\vartheta}$:

$$\boldsymbol{\vartheta}^{(new)} = \boldsymbol{\vartheta}^{(old)} + \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}\nabla\mathcal{L}(\boldsymbol{\vartheta}^{(old)}).$$

4.3 Parameter Learning in DRVM

In a similar manner as Tipping (2001) and Jiang et al. (2019) we can maximize the posterior distribution over all parameters by the approximation

$$p(\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2 | \boldsymbol{t}) = \frac{p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)p(\boldsymbol{\alpha})p(\boldsymbol{\beta})}{p(\boldsymbol{t})}$$

$$\propto p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2).$$

By rewriting Equation (4.2), we get the posterior distribution over the hyperparamaters:

$$p(\boldsymbol{t}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha})p(\boldsymbol{\vartheta}|\boldsymbol{\beta})}{p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)} \tag{4.9}$$

The distribution from Equation (4.8) with respect to the sample weights $\boldsymbol{w}$ is identical to the one in the original RVM by Tipping (2001) in Equation (2.8), just with the inclusion of the individual feature weights $\boldsymbol{\vartheta}$. Thus, the update-formula for α_i and σ^2 are given by Equation (2.16) and (2.17) with the inclusion of the feature weights, that is

$$\alpha_i^{\text{new}} = \frac{\gamma_{w,i}}{\mu_{w,i}^2}, \tag{4.10}$$

$$\left(\sigma^2\right)^{\text{new}} = \frac{||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{\mu}_{\boldsymbol{w}}||^2}{N - \sum_i \gamma_{w,i}}, \tag{4.11}$$

where $\gamma_{w,i} \equiv 1 - \alpha_i \Sigma_{w,ii}$. Further, the simultaneous posterior distribution from Equation (4.8) with respect to the feature weights $\boldsymbol{\vartheta}$ is identical to the one in the PFCVM method by Jiang et al. (2019), in Equation (3.24) with mean vector and covariance matrix given by the simplification in Equation (3.18). Thus, we get the same update formula for β_i as in the PFCVM, given by the last part of Equation (3.29).

4.4 Algorithm of the Sample and Feature Selective Relevance Vector Based Model

The algorithm of the DRVM model given by Algorithm 4 is similar to the PFCVM model, but were we have to use the Newton-Raphson update formula only when updating with respect to the feature weights $\boldsymbol{\vartheta}$, as the Laplace approximation is exact with respect to the sample weights $\boldsymbol{w}$.

Algorithm 4 Dimensionality Reducing Relevance Vector Machine (DRVM)

```
Initialize $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\sigma^2$ to some reasonable values
Compute $\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\vartheta}_{MP}$, $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ and $\boldsymbol{\mu}_{\boldsymbol{w}}$
while convergence criteria are not met do
    for all $\alpha_i$ in $\boldsymbol{\alpha}$ do
        if $\alpha_i > \alpha_{\text{Thresh}}$ then
            delete $\boldsymbol{\phi}_i$ and $\alpha_i$
        end if
    end for
    for all $\beta_k$ in $\boldsymbol{\beta}$ do
        if $\beta_k > \beta_{\text{Thresh}}$ then
            delete feature number k and hence $\beta_k$
        end if
    end for
    Update $\boldsymbol{\Sigma}_{\boldsymbol{w}}$, $\boldsymbol{\mu}_{\boldsymbol{w}}$, $\boldsymbol{\alpha}$, $\sigma^2$, $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\vartheta}_{MP}$ using IRLS method, $\boldsymbol{\beta}$ and $\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}$
end while
```

4.5 Making Predictions

By simultaneously iterating α_i, β_i and σ^2 until convergence to the most probable vectors of values $\boldsymbol{\beta}_{MP}$ and $\boldsymbol{\alpha}_{MP}$, and σ^2_{MP}, we can predict for new target t_*. The approximated predictive distribution is given by

$$\begin{aligned} p(t_*|\boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma^2_{MP}) &= \int p(t_*|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2_{MP}) p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma^2_{MP})\, d\boldsymbol{w}\, d\boldsymbol{\vartheta} \\ &\approx \int p(t_*|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2_{MP}) \mathcal{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}) \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}})\, d\boldsymbol{w}\, d\boldsymbol{\vartheta}, \end{aligned}$$

where we have used the relation in Equation (4.8). By integrating out the feature weights $\boldsymbol{\vartheta}$, we are left with (Jiang et al., 2019)

$$p(t_*|\boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma^2_{MP}) \approx \int p(t_*|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2_{MP}) \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}})\, d\boldsymbol{w}.$$

This equation is equivalent to the predictive distribution in the original RVM method given by Equation (2.22). This means that we make prediction for future target variables t_* based on the feature selective algorithm in the previous section, by using the predictive distribution from RVM, with the inclusion of the feature weights $\boldsymbol{\vartheta}$ in the kernel basis functions and with the other parameters estimated like described in this chapter. That is

$$t_*|\boldsymbol{t}, \boldsymbol{\alpha}_{MP}, \sigma^2_{MP} \sim \mathscr{N}(\mu_*, \sigma^2_*),$$

where

$$\begin{aligned}
\mu_* &= \boldsymbol{\mu_w}^\top \boldsymbol{\phi_\vartheta}(\boldsymbol{x}_*), \\
\sigma^2_* &= \sigma^2_{MP} + \boldsymbol{\phi_\vartheta}(\boldsymbol{x}_*)^\top \boldsymbol{\Sigma_w} \boldsymbol{\phi_\vartheta}(\boldsymbol{x}_*).
\end{aligned}$$

5 | Experimental Results

In this chapter, we will first do some illustrations of the suggested DRVM method compared to the RVM by Tipping (2001), on simple synthetic data. Further, we are showing results for the two methods on some benchmark data sets. The methods will be compared both in terms of feature and sample selective strength, and in their ability to make accurate predictions for future target values $\boldsymbol{t}$. The DRVM method is sensitive with respect to initial values, and with respect to the partitioning in test and training set. Therefore, we have been using cross-validation (CV) on five different partitions of the data sets to choose the initial values. This is not an easy task as the model is slow in the learning procedure. Further, we have been using the RVM method implemented in the R-package *kernlab* to train the RVM by Tipping (2001). In this algorithm the initial RBF kernel parameter ϑ is chosen by an estimation method that is dependent on the specific partitioning in test and training data set. Thus, in the RVM method the initial RBF kernel parameter is estimated individually for each different training data set, while for the DRVM the initial values are chosen on a more general level, using CV, and is equal for every partitioning in training set for the same data set. This is a significant difference between the RVM and DRVM training procedure in these experiments.

5.1 Examples on One Dimensional Synthetic Input Data

In the following sections, we are doing experiments on synthetic data to see how the model fits for a known system and output function. We will first inspect how well the model fits on a simple one dimensional case, with and without noise. To illustrate support vector regression, the sinc function is often used (Tipping, 2001), and we will make no exception. The sinc function is given by

$$\boldsymbol{t} = \frac{sin(\boldsymbol{x})}{\boldsymbol{x}} + \boldsymbol{\epsilon},$$

where $\boldsymbol{x}$ is the input vector, and $\boldsymbol{\epsilon}$ is the random noise vector. In both examples we are using a training data set of 100 samples with only one feature equally spaced on $[-10, 10]$. In the first example, the output $\boldsymbol{t}$ is a sinc function of the single column without noise, that is $\boldsymbol{\epsilon} = (0, \dots, 0)$.

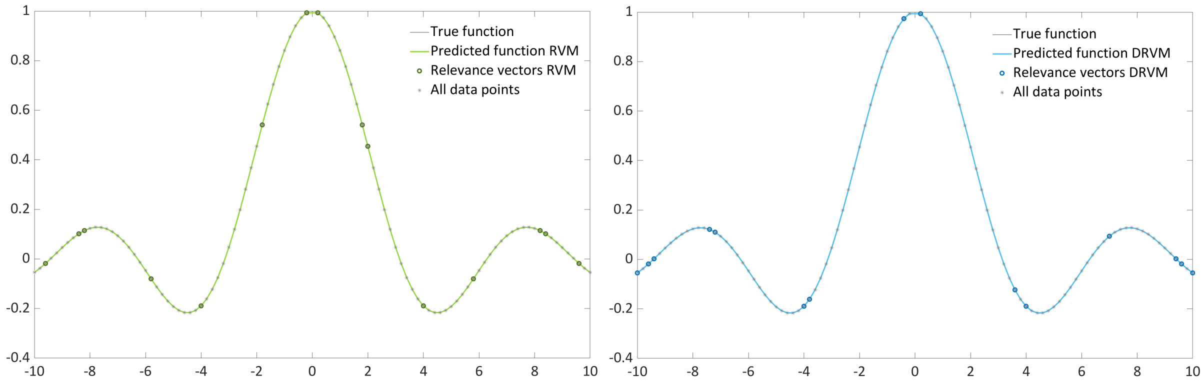

Figure 5.1: Data generated from the sinc function without noise modeled with RVM and DRVM. The predicted function from RVM is shown in green and the one from DRVM in blue. The true functions in gray are hidden behind the predicted functions as the models fits perfectly. The true data points are marked with stars and the relevance vectors with circles.

As we can see in Figure 5.1, both the RVM and DRVM model seems to fit the true function perfectly for this one dimensional noise free data set. The difference in test error between the two methods is negligible, and when it comes to the sparseness of the models, both requires 15 relevance vectors, and is equally sparse in this case. In the next example, we are adding random uniform noise in $[-0.1, 0.1]$ (Tipping, 2001), to see if the model is able to capture the form also when data are noisy. The error is calculated on 1000 samples of the true function without noise, and the average number of relevance vectors (nRV) and the root-mean-square error (RMSE) for both methods are given in Table 5.1. As expected, Figure 5.2 shows that when the targets $\boldsymbol{t}$ is built up by only one column, the DRVM model in blue works similar as the original RVM model in green. Again, the DRVM model is fitting the system in the output data very well and is slightly more accurate than the original RVM method with an average RMSE of 0.017 against 0.023 for the RVM. When it comes to the sparseness of the model, the RVM is on average choosing 10 relevance vectors while the DRVM model is on average choosing 9.

From these experiments on one dimensional input data, we can see that our proposed DRVM model is capturing the form of the output function equally as good as the RVM model. In addition, it seems to be equally as sparse, and the test error is in fact a tiny bit better. This slight improvement in test error can be due to the differences in how the initial value of the RBF kernel parameter ϑ is chosen. However, the strength of the DRVM model is not due to the sample selective aspect. It is the feature selective strength that makes the main difference between the RVM and the DRVM. To examine the feature selective strength of the DRVM model we need to use multidimensional data sets, which we will do in the next sections.

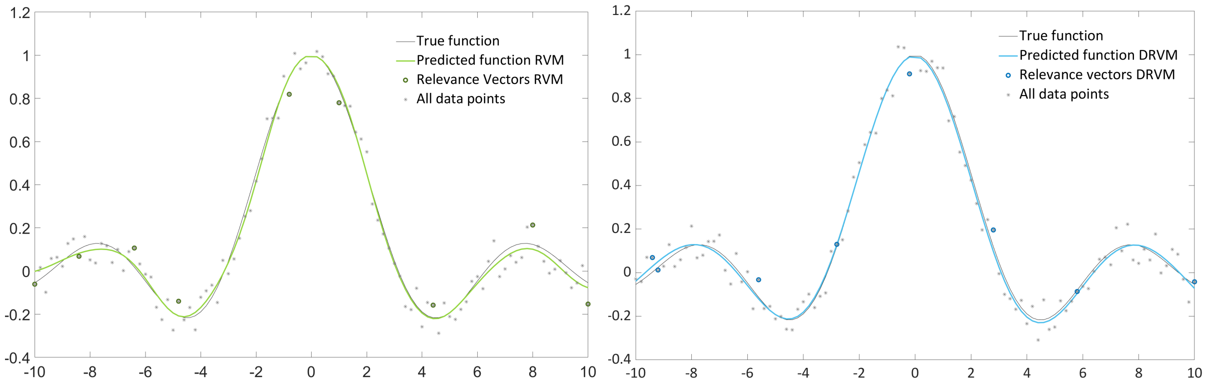

Figure 5.2: Data generated from the sinc function with uniform noise in $[-0.1, 0.1]$ modeled with RVM and DRVM. The predicted function from RVM is shown in green and the one from DRVM in blue, while true functions is shown in gray. The true data points are marked with stars and the relevance vectors with circles.

5.2 Comparisons on Benchmark Data Sets

In this section we are doing experiments on three different benchmark data sets, to get a more clear impression of how accurate and sparse our proposed models is when data are multidimensional. We are still using CV to choose the best initial values for the parameters. Table 5.1 below shows the average results in terms of root-mean-square error (RMSE), number of chosen relevance vectors (nRV) and number of chosen relevance features (nRF) over 100 repetitions of modeling in all of our experiments, where N is the number of observations and P the number of features in the training data set. It will be referred to this table several times during this section.

			nRV		nRF		RMSE	
Data set	**N**	**P**	**RVM**	**DRVM**	**RVM**	**DRVM**	**RVM**	**DRVM**
Sinc (Uniform noise)	100	1	10.1	9.2	1	1	0.023	0.017
Friedman # 1	240	10	28.4	18.0	10	7.0	1.60	1.14
Diabetes	221	10	20.1	3.3	10	7.3	61.24	55.54
Boston Housing	253	13	32.1	11.0	13	3.3	6.21	5.87

Table 5.1: Comparison of the average number of relevance vectors (nRV), relevance features (nRF) and RMSE for the different data sets and methods.

The Friedman # 1 Data

The *Friedman # 1* data was first constructed by Friedman (1991) and is generated from 10 random uniform input variables in [0, 1]. The outputs are given by the function (Gramacy, 2020):

$$y(\boldsymbol{x}) = 10\sin(\pi \boldsymbol{x}_1 \boldsymbol{x}_2) + 20\left(\boldsymbol{x}_3 - \frac{1}{2}\right)^2 + 10\boldsymbol{x}_4 + 5\boldsymbol{x}_5.$$

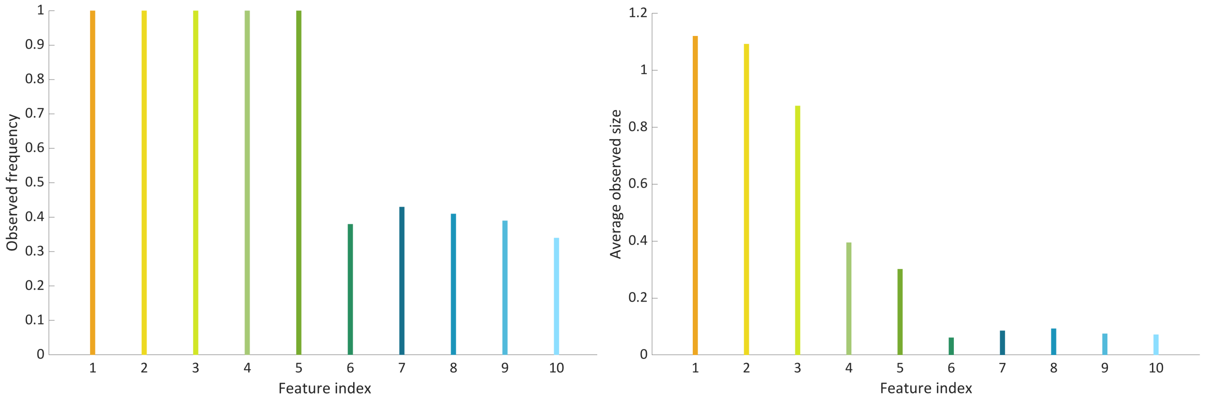

Figure 5.3: Bar plot of the frequency of chosen features by DRVM on the Friedman #1 data over 100 repetitions to the left. The right side shows the average observed size of the different feature weights over these 100 repetitions. The data set is constructed such that the latter five features are only noise, and we can see that in many of the repetitions they are pruned from the model the model.

This output is dependent on the five first features, while feature six to ten are only noise columns (Tipping (2001), Gramacy (2020)). The models in this experiment are trained on 240 randomly generated samples of the data, with Gaussian noise of one standard deviation added, and they are tested on 1000 randomly generated samples without noise. The results are averaged over 100 repetitions. As the last five features are only noise columns, we hope for the DRVM model to choose the first five features which are actually affecting the output, while ignoring the last five. In addition, as the RVM model is using all the features including the irrelevant ones, we expect the DRVM to predict more accurately for new data than the original RVM method. The left side of Figure 5.3 shows the frequency of chosen features over the 100 repetitions of modeling on the Friedman # 1 data. The left side of the figure shows the average size of the different features weights in the vector $\boldsymbol{\vartheta}$, where the weight is counted as zero if the corresponding feature is not chosen. As we can see from the figure, the DRVM model is choosing the five first features in 100% of the repetitions, while the latter five is chosen in approximately $35-40\%$ of the repetitions, which is what we hoped for the model to do. As shown in Table 5.1, on average the DRVM model chose seven features to affect the predictions and five of these must be those which are actually affecting the output, as all these are chosen in all of the repetitions as shown in the figure. This means that on this specific data set, the DRVM model is always choosing he relevant features along with on average two additional irrelevant features. Regarding the sparseness towards the samples, the DRVM model is for the Friedman # 1 data choosing on average 18 relevance vectors, while the RVM model is choosing 28, as shown in Table 5.1. In addition, the error measure on this data set is on average 1.14 for the DRVM model against 1.60 for the RVM model. Thus, it seems like the DRVM model is sparser than the RVM model both with respect to the samples and features, while also being more accurate.

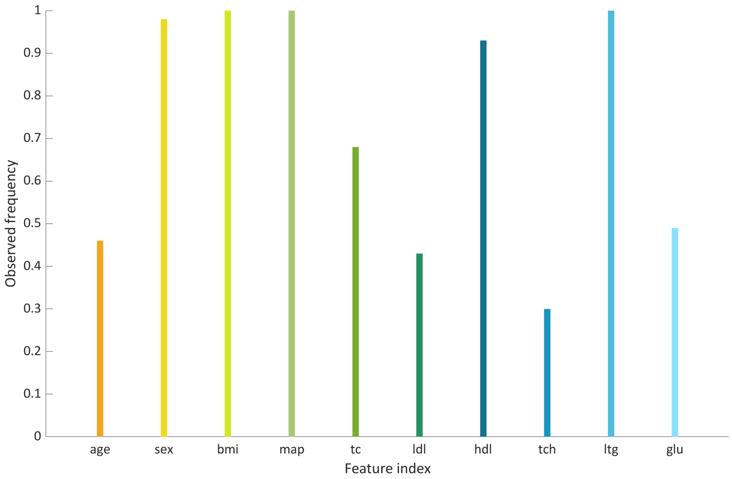

Figure 5.4: Bar plot of the frequency of chosen features by DRVM on the Diabetes data over 100 repetitions. Some of the features: sex, bmi, map, hdl and ltg are chosen in almost 100 % of the repetitions, while others: age, ldl and tch are chosen in less than 50 % of them.

The Diabetes Data

The *Diabetes* data is part of the *lars* library in *R*, and has 10 variables: age, sex, BMI, blood pressure and six different measurements of blood serum levels (Efron et al., 2004). The data is based on 442 diabetes patients, together with measures of disease progression one year after baseline. We want to estimate this progression. We are training on 50% of data, and hence testing on the remaining 50%. The results are averaged over 100 repetitions. In Figure 5.4, we can see the frequency of how often the different features are chosen, and the summed up results are given in Table 5.1. As the frequency plot shows, the model is consistently choosing the sex, bmi, map, hdl and ltg to be relevant features, while age, tc, ldl, tch and glu seems to not be that important in this model. We can see in the table that the DRVM model is on average choosing seven of the features to affect the model, and it has an error measure on the test data of 55.5 against 61.2 for the RVM. With the DRVM method choosing on average just above three vectors to contribute to the predictions compared to the RVM model choosing 20 relevance vectors, the DRVM is sparser both with respect to samples and features and at the same time being more accurate.

The Boston Housing Data

The last data set we are using, is the *Boston Housing* data, that was first used by Harrison Jr and Rubinfeld (1978). The data set contains 506 observations of the median house value from different areas of Boston Mass together with 13 features, which are described in Appendix A.3. The data set is divided into test and training set with 50 % in each. The average result over 100 repetitions are shown in Table 5.1.

We can see on Figure 5.5 that the DRVM model is consistently choosing the feature *nox*,

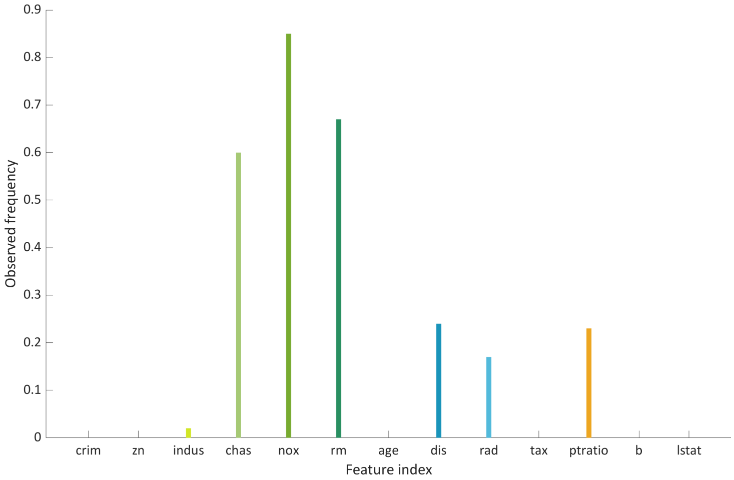

Figure 5.5: Bar plot of the frequency of chosen features by DRVM on the Boston Housing data over 100 repetitions. Some of the features: chas, nox and rm are chosen in over 60 % of the repetitions, while some of them are not chosen at all.

that is the nitric oxide concentration in the area, which seems credible as the previous research by Harrison Jr and Rubinfeld (1978) have found this feature to be a highly significant feature. We can also see that on this data set the model is consistently pruning six of the features in 100 % of the repetitions, which indicates that these features are not significantly informative. Further, the DRVM is again sparser than the RVM model by choosing 11 relevance vectors against 32. The error measure is slightly improved by using the dimensionality reducing extension in the DRVM method, and it is on average predicting using 3.3 out of 13 features.

Altogether, we can see that on all these four data sets Sinc, Friedman # 1, Diabetes and Boston Housing, the DRVM model is sparser both with respect to relevance vectors and relevance features, while at the same time predicting more accurately. This is what we aimed for with the model, as it is reasonable to think that the DRVM model will be similar to the RVM when all features are significant, while it should be both sparser and more accurate when data includes irrelevant noise features.

6 Discussion

The aim of this project was to develop a feature selective regression model which at the same time was sparse in sample size, to deal with data that is possibly both big and high dimensional. As always in machine learning, it is not possible to find a single best method that always works the best, regardless of differences in the input data. However, methods may have characteristic properties which makes them work better for some kinds of data. Due to our experiments, the DRVM model seems to be an improvement both in accuracy and interpretability compared to the original RVM. Although, this may not always be the case, our experiments is indicating that our method can make better and sparser predictions on multidimensional data. If we take a closer look at the Sinc data with uniform noise, we can see that the DRVM method on average is more accurate than the RVM method. However, as the data set only contains one feature, we were expecting that the methods would be approximately equally accurate. Most likely the slight increase in accuracy of fit is due to the differences in how the RBF kernel parameter ϑ is chosen, as described in the beginning of previous chapter. For one dimensional data it therefore seems reasonable to conclude that the RVM and DRVM method is almost identical to each other. When the number of features is larger, the DRVM seems to be at least as accurate as the original RVM model, which is expected as some of the features may not actually affect the output.

The key principle of our proposed model is modifying the kernel basis function by using indivdual kernel parameters ϑ_k, and we have not discovered many papers which is using this approach and we have not found anyone using it in the regression case of RVM. Thus, our work stands out as innovative, and our new approach may be used for other kernel based Bayesian learning methods in further research.

One of the limitations of the DRVM method is that it is slow in the learning procedure, and at the same time sensitive with respect to initial values and the partitioning into training and test data. Hence, training the model is often a cumbersome task.

It is also worth mentioning how the DRVM method stands with respect to interpretability and parsimonity. Kernel-based methods are not always easy to interpret as every element of the model matrix is a function of input data. However, by using sparse and feature selective methods, like DRVM we are reducing the complexity and are predicting using fewer features. This means that the resulting model by our algorithm is more parsimonious and maybe a bit easier to interpret.

In many research papers, including those referred to in this thesis, the mathematical deductions are not explained in detail. However, working with this research, we have deduced all the mathematical expressions that are used. This has been time consuming and not straight forward. Where it was not clear in the actual paper how the formulas were deduced, we had to search for fundamental mathematical formulas and properties. We also contacted some of the researchers to get a better understanding of the mathematics and how to implement the model in MATLAB. Implementing the method was challenging as well, as we had to first understand the implementation of both the original RVM and the PFCVM model, notice all their differences and then write the code associated with this method. Hence, there is work behind this thesis that is not shown in the paper. Still, it has been challenging, educational and very interesting to work with this thesis.

Further Research

As this method is slow in the learning procedure, we will suggest for further research to extend the FRVM method from Chapter 2.3.2 using a similar approach as in the PFCVM and DRVM methods. We have in fact done some research on this and started to develop a possible approach for both FRVM and for the Noise-Robust Fast Sparse Bayesian Learning (BLS) method by Helgøy and Li (2019). Considerations about these extensions for the faster methods are postulated in Appendix A.1. In addition, our method is only developed with respect to the RBF kernel basis function. Extending this method so that other kernel basis functions can be used could be interesting. Another research idea based on this work could be to extend other kernel based Bayesian learning methods to be feature selective, using the same approach and modification of kernel parameters.

References

Babacan, S. D., R. Molina, and A. K. Katsaggelos
2009. Bayesian compressive sensing using laplace priors. *IEEE Transactions on Image Processing*, 19(1):53–63.

Bellman, R. E. and S. E. Dreyfus
1957. On the formulation of dynamic-programming problems–i.

Bishop, C. M.
2006. *Pattern Recognition and Machine Learning.* springer.

Chen, H., P. Tino, and X. Yao
2009. Probabilistic classification vector machines. *IEEE Transactions on Neural Networks*, 20(6):901–914.

Choi, S. C. and R. Wette
1969. Maximum likelihood estimation of the parameters of the gamma distribution and their bias. *Technometrics*, 11(4):683–690.

Dirac, P.
1958. The δ function. *The Principles of Quantum Mechanics (4th ed.)*, P. 58.

Efron, B., T. Hastie, I. Johnstone, R. Tibshirani, et al.
2004. Least angle regression. *The Annals of statistics*, 32(2):407–499.

Fletcher, T.
2010. Relevance vector machines explained. *University College London: London, UK.*

Friedman, J. H.
1991. Multivariate adaptive regression splines. *The annals of statistics*, Pp. 1–67.

Gramacy, R.
2020. "friedman.1.data" from tgp v2.4-17.

Harrison Jr, D. and D. L. Rubinfeld
1978. Hedonic housing prices and the demand for clean air.

Helgøy, I. M. and Y. Li
2019. A noise-robust fast sparse bayesian learning model. *arXiv preprint arXiv:1908.07220*.

Higham, N. J.
2002. *Accuracy and Stability of Numerical Algorithms*, volume 80. Siam.

Jiang, B., C. Li, M. D. Rijke, X. Yao, and H. Chen
2019. Probabilistic feature selection and classification vector machine. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(2):21.

Krishnapuram, B., A. Harternink, L. Carin, and M. A. Figueiredo
2004. A bayesian approach to joint feature selection and classifier design. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1105–1111.

LI, X.
2017. Tricks of sigmoid function.

Lu, T.-T. and S.-H. Shiou
2002. Inverses of 2× 2 block matrices. *Computers & Mathematics with Applications*, 43(1-2):119–129.

MacKay, D. J.
1992. Bayesian interpolation. *Neural computation*, 4(3):415–447.

Magnus, J. R. and H. Neudecker
2019. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. John Wiley & Sons.

Mohsenzadeh, Y., H. Sheikhzadeh, and S. Nazari
2016. Incremental relevance sample-feature machine: A fast marginal likelihood maximization approach for joint feature selection and classification. *Pattern Recognition*, 60:835–848.

Mohsenzadeh, Y., H. Sheikhzadeh, A. M. Reza, N. Bathaee, and M. M. Kalayeh
2013. The relevance sample-feature machine: A sparse bayesian learning approach to joint feature-sample selection. *IEEE transactions on cybernetics*, 43(6):2241–2254.

Oldham, K. B., J. Myland, and J. Spanier
2010. *An Atlas of Functions: with Equator, the Atlas Function Calculator*. Springer Science & Business Media.

Park, T. and G. Casella
2008. The bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686.

Peng, R. D.
2018. Advanced statistical computing. *Work in progress.*

Platt, J. C., J. Shawe-Taylor, A. J. Smola, R. C. Williamson, et al.
1999. Estimating the support of a high-dimensional distribution. *Technical Report MSR-T R-99–87, Microsoft Research (MSR).*

Rubin, D. B.
1983. Iteratively reweighted least squares. *Encyclopedia of Statistical Sciences*, 4:272–275.

Smola, A. J., B. Schölkopf, and K.-R. Müller
1998. The connection between regularization operators and support vector kernels. *Neural networks*, 11(4):637–649.

Tipping, M.
1996. The boston housing dataset.

Tipping, M.
2016. Sparsebayes software.

Tipping, M. E.
2001. Sparse bayesian learning and the relevance vector machine. *Journal of machine learning research*, 1(Jun):211–244.

Tipping, M. E., A. C. Faul, et al.
2003. Fast marginal likelihood maximisation for sparse bayesian models. In *AISTATS*.

Vert, J.-P., K. Tsuda, and B. Schölkopf
2004. A primer on kernel methods. *Kernel methods in computational biology*, 47:35–70.

A | Appendix

The appendix will be a brief introduction to some other methods that I have been working with and some of the interesting things I have discovered during my research period. In addition, it will show part of the code I used to experiment with the DRVM method.

A.1 Possible Feature Selective Extension of Fast Bayesian Learning

As the DRVM method is slow in the learning procedure, I have started to look at the possibility to extend the DRVM method from Section 4 to be fast in a similar manner as Tipping et al. (2003) did, explained in Section 2.3.2. In addition, I have worked with a noise-robust sparse Bayesian learning method by Helgøy and Li (2019) and how that method can be extended to be simultaneously feature selective. I did not fully complete any of these two methods, but I will give some details of what I have done and the challenges I ran into.

A.1.1 Extension of the Fast Relevance Vector Machine

From the optimization equations in Section 4.3 and 3.4 it is possible to divide the expressions into one part including the α_i and β_i and one part not including the actual index, like Tipping et al. (2003) did for the FRVM method. As described in Section 4.3, the marginal likelihood over the hyperparameters $\boldsymbol{\alpha}$ corresponding to the sample weights $\boldsymbol{w}$ is identical to the corresponding equation for the original RVM with the inclusion of the feature weights $\boldsymbol{\vartheta}$ in the kernel basis functions. Thus, the fast optimization with respect to the sample weights $\boldsymbol{w}$ is equal to the one in FRVM, just with the feature weights $\boldsymbol{\vartheta}$ included. That is

$$\alpha_i = \begin{cases} \frac{s_{\boldsymbol{\vartheta},i}^2}{q_{\boldsymbol{\vartheta},i}^2 - s_{\boldsymbol{\vartheta},i}} & \text{if } q_{\boldsymbol{\vartheta},i}^2 > s_{\boldsymbol{\vartheta},i} \\ \infty & \text{if } q_{\boldsymbol{\vartheta},i}^2 \leq s_{\boldsymbol{\vartheta},i} \end{cases},$$

where $s_{\boldsymbol{\vartheta},i}$ and $q_{\boldsymbol{\vartheta},i}$ are given by Equation (2.19) where the individual feature weights are included in the kernel basis functions. By the exact same arguments, we can also use the same estimate for σ^2 as in Equation (4.11), again with $\boldsymbol{\vartheta}$ included in the kernel functions, that is

$$\sigma^2 = \frac{||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}} \boldsymbol{w}_{MP}||^2}{N - \sum_i \gamma_i^{\boldsymbol{w}}}.$$

Now looking for the estimate of β_i, we get by using the estimates from Equation (3.18), that the likelihood function of $\boldsymbol{\beta}$ from Equation (3.28) with the simplification from Equation (3.18) can be written as

$$\mathcal{L}(\boldsymbol{\beta}) = \frac{1}{2} \ln |\boldsymbol{B}| - \frac{1}{2} \ln |\boldsymbol{B} - \boldsymbol{H}_{\boldsymbol{\vartheta}}| + \frac{1}{2} \boldsymbol{\epsilon_{\vartheta}}^{\top} \boldsymbol{B}^{-1} \boldsymbol{\epsilon_{\vartheta}},$$

where $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}} = (\boldsymbol{B} - \boldsymbol{H}_{\boldsymbol{\vartheta}})^{-1}$. We are rewriting the inverse of the covariance matrix ${\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}^{-1}$ like

$$\begin{aligned}
{\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}^{-1} &= \boldsymbol{IBI} + \boldsymbol{H}_{\boldsymbol{\vartheta}} \\
&= \boldsymbol{H}_{\boldsymbol{\vartheta}} + \sum_{m \neq i} \beta_m \boldsymbol{1}_m \boldsymbol{1}_m^{\top} + \beta_i \boldsymbol{1}_i \boldsymbol{1}_i^{\top} \\
&= {\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}_{-i}^{-1} + \beta_i \boldsymbol{1}_i \boldsymbol{1}_i^{\top},
\end{aligned}$$

where $\boldsymbol{1}_m = \big(0, \dots, 1, \dots, 0\big)^{\top}$ with 1 at position m. By now using the determinant identity in Equation (0.6), we get

$$|{\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}^{-1}| = |\ln {\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}_{-i}^{-1}||\boldsymbol{I} + \beta_i \boldsymbol{1}_i^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}-i} \boldsymbol{1}_i|,$$

such that

$$\ln |{\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}^{-1}| = \ln |{\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}_{-i}^{-1}| + \ln |\boldsymbol{I} + \beta_i \boldsymbol{1}_i^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}-i} \boldsymbol{1}_i|.$$

Thus, using this rewrite, the log likelihood function above can be split into one term including and one term not including β_i, in the following manner:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\beta}) &= \frac{1}{2} \sum_{m \neq i} \Big\{ \ln \beta_m - \ln |\boldsymbol{\Sigma}_{\boldsymbol{\vartheta},i}^{-1}| + \frac{\epsilon_m^2}{\beta_m} \Big\} + \ln \beta_i - \ln (1 + \beta_i \boldsymbol{1}_i^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\vartheta},i} \boldsymbol{1}_i) + \frac{\epsilon_i^2}{\beta_i} \\
&= \mathcal{L}(\boldsymbol{\beta}_{-1}) + \ell(\beta_i).
\end{aligned}$$

In the deduction above $\epsilon_{\boldsymbol{\vartheta},m}$ is the m'th diagonal element of the vector $\boldsymbol{\epsilon_{\vartheta}}$ from Equation (3.18). Thus, differentiating $\mathcal{L}(\boldsymbol{\beta})$ is equivalent to differentiating $\ell(\beta_i)$, and we get

$$\frac{\partial \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_i} = \frac{\partial \ell(\beta_i)}{\partial \beta_i} = \frac{1}{2} \bigg(\frac{1}{\beta_i} - \frac{\boldsymbol{1}_i^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\vartheta},i} \boldsymbol{1}_i}{1 + \beta_i \boldsymbol{1}_i^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\vartheta},i} \boldsymbol{1}_i} - \frac{\epsilon_i^2}{\beta_i^2} \bigg),$$

which by equating to zero gives

$$\beta_i = \frac{\epsilon_i^2}{1 - \epsilon_i^2 \mathbf{1}_i^\top \mathbf{\Sigma}_{\boldsymbol{\vartheta},i} \mathbf{1}_i}.$$

As we need the estimate to be positive defined and the numerator is always positive, we need the denominator to also be positive. This is satisfied when

$$\epsilon_i^2 \mathbf{1}_i^\top \mathbf{\Sigma}_{\boldsymbol{\vartheta},i} \mathbf{1}_i < 1,$$

and the estimate for β_i is:

$$\beta_i = \begin{cases} \frac{\epsilon_i^2}{1-\epsilon_i^2 \mathbf{1}_i^\top \mathbf{\Sigma}_{\boldsymbol{\vartheta},i} \mathbf{1}_i} & \text{if } \epsilon_i^2 \mathbf{1}_i^\top \mathbf{\Sigma}_{\boldsymbol{\vartheta},i} \mathbf{1}_i < 1 \\ \infty & \text{if } \epsilon_i^2 \mathbf{1}_i^\top \mathbf{\Sigma}_{\boldsymbol{\vartheta},i} \mathbf{1}_i \geq 1 \end{cases}.$$

A.1.2 Extension of the Noise-Robust Fast Sparse Bayesian Learning Model

Based on the Fast Relevance Vector Machine (FRVM) by Tipping et al. (2003), and inspired by the Fast Laplace (FLAP) model by Babacan et al. (2009), Helgøy and Li (2019) developed a fast sparse Bayesian learning method which is also robust to the noise variance. They utilized the hierarchical prior from the Bayesian Lasso model by Park and Casella (2008) together with a fast type-II maximization algorithm as used by Tipping et al. (2003). The procedure led to a model that is both sparser, more flexible and at the same time stable when data is noisy. This model is referred to as the Noise-Robust Fast Sparse Bayesian Learning (BLS) method. If we are able to construct a model based on BLS that is simultaneously selective with respect to both samples and features, we may get a Bayesian learning model that is both fast, sparse, feature selective and robust to the noise variance. In this section, we will postulate some hypothesis and calculations about how this can be done.

The BLS Method

This section will be a short illustration of the BLS method, and further details about the development is to be find in Helgøy and Li (2019). In this method, a Laplacian prior conditional on the noise variance is used:

$$p(\boldsymbol{w}|\sigma^2) = \prod_{i=1}^{N} \frac{\sqrt{\lambda}}{2\sqrt{\sigma^2}} e^{-\frac{\sqrt{\lambda}|w_i|}{\sqrt{\sigma^2}}}. \tag{A.1}$$

This prior is complicated to work with, and a scale mixture of normals is used:

$$\frac{\sqrt{\lambda}}{2\sqrt{\sigma^2}} e^{-\frac{\sqrt{\lambda}|w_i|}{\sqrt{\sigma^2}}} = \int_0^\infty \frac{1}{\sqrt{2\pi\gamma_i\sigma^2}} e^{-\frac{w_i^2}{2\gamma_i\sigma^2}} \frac{\sqrt{\lambda}^2}{2} e^{-\frac{\sqrt{\lambda}^2\gamma_i}{2}} \, dw_i. \tag{A.2}$$

The parameters in (A.1) and (A.2) have the following hierarchical structure (Park and Casella, 2008):

$$\begin{aligned}
\boldsymbol{t}|\boldsymbol{w}, \sigma^2 &\sim \mathscr{N}\big(\boldsymbol{t}|\boldsymbol{\Phi}\boldsymbol{w}, \sigma^2\big), && \text{(A.3)}\\
\boldsymbol{w}|\boldsymbol{\gamma}, \sigma^2 &\sim \mathscr{N}\big(\boldsymbol{w}|0, \boldsymbol{\Lambda}\big), \quad \boldsymbol{\Lambda} = \operatorname{diag}\big(\gamma_0\sigma^2, \dots, \gamma_N\sigma^2\big), && \text{(A.4)}\\
\boldsymbol{\gamma}|\lambda &\sim \prod_{i=0}^{N} \operatorname{Exp}\Big(\gamma_i|\frac{\lambda}{2}\Big), \\
\lambda &\sim \operatorname{Gamma}\big(\lambda|a, b\big), \quad (a, b > 0) \\
\sigma^2 &\sim \operatorname{Gamma}\big(\sigma^2|c, d\big). \quad (c, d > 0)
\end{aligned}$$

By integrating out the hyperparameters in Equation (A.2), the prior distribution is reduced to the sparse prior in Equation (A.1). This hierarchical structure is even more sparse than the student-t distribution that is used in the RVM and illustrated in Section 1.2.2 (Helgøy and Li, 2019).

Helgøy and Li (2019) got the posterior distribution

$$p(\boldsymbol{w}, \boldsymbol{\gamma}, \lambda, \sigma^2|\boldsymbol{t}) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\gamma}, \lambda, \sigma^2)p(\boldsymbol{w}, \boldsymbol{\gamma}, \sigma^2, \lambda)}{p(\boldsymbol{t})},$$

and the predictive distribution

$$p(t_*|\boldsymbol{t}) = \int p(t_*|\boldsymbol{w}, \boldsymbol{\gamma}, \lambda, \sigma^2)p(\boldsymbol{w}, \boldsymbol{\gamma}, \lambda, \sigma^2|\boldsymbol{t}) \, d\boldsymbol{w} \, d\boldsymbol{\gamma} \, d\lambda \, d\sigma^2. \tag{A.5}$$

As it is not possible to find the posterior distribution $p(\boldsymbol{w}, \boldsymbol{\gamma}, \lambda, \sigma^2|\boldsymbol{t})$ in Equation (A.5) analytically, Helgøy and Li (2019) are using that

$$p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\gamma}, \lambda, \sigma^2) = \frac{p(\boldsymbol{w}, \boldsymbol{\gamma}, \lambda, \sigma^2|\boldsymbol{t})}{p(\boldsymbol{\gamma}, \lambda, \sigma^2|\boldsymbol{t})}, \tag{A.6}$$

and shows that the posterior distribution over the sample weights $\boldsymbol{w}$ is given by

$$\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\gamma}, \lambda, \sigma^2 \sim \mathscr{N}(\boldsymbol{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

where:

$$\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^\top\boldsymbol{t}, \tag{A.7}$$

$$\boldsymbol{\Sigma} = \left(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{\Lambda}^{-1}\right)^{-1}. \tag{A.8}$$

The marginal likelihood over targets $\boldsymbol{t}$ is given by (Helgøy and Li, 2019)

$$\boldsymbol{t}|\boldsymbol{\gamma}, \sigma^2, \lambda \sim \mathscr{N}(\boldsymbol{t}|0, \boldsymbol{C}),$$

where the covariance matrix $\boldsymbol{C}$ is:

$$\boldsymbol{C} = (\sigma^2\boldsymbol{I}_N + \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^\top). \tag{A.9}$$

Helgøy and Li (2019) approximated the joint posterior distribution over all the parameters by

$$\begin{aligned} p(\boldsymbol{\gamma}, \lambda, \sigma^2|\boldsymbol{t}) &= \frac{p(\boldsymbol{t}, \boldsymbol{\gamma}, \lambda, \sigma^2)}{p(\boldsymbol{t})} \\ &\propto p(\boldsymbol{t}, \boldsymbol{\gamma}, \lambda, \sigma^2), \end{aligned} \tag{A.10}$$

where (Helgøy and Li, 2019)

$$\begin{aligned} p(\boldsymbol{t}, \boldsymbol{\gamma}, \lambda, \sigma^2) &= \int p(\boldsymbol{t}|\boldsymbol{w}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}|\lambda)p(\lambda)p(\sigma^2)\, d\boldsymbol{w} \\ &= p(\boldsymbol{t}|\boldsymbol{\gamma}, \sigma^2, \lambda)p(\boldsymbol{\gamma}|\lambda)p(\lambda)p(\sigma^2) \\ &= (2\pi)^{-\frac{N}{2}}|\boldsymbol{C}|^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2}\boldsymbol{t}^\top\boldsymbol{C}^{-1}\boldsymbol{t}\right\} p(\boldsymbol{\gamma}|\lambda)p(\lambda)p(\sigma^2). \end{aligned} \tag{A.11}$$

Helgøy and Li (2019) took the logarithm of (A.11), which gave

$$\begin{aligned} \ln p(\boldsymbol{t}, \boldsymbol{\gamma}, \sigma^2, \lambda) = &-\frac{1}{2}\log|\boldsymbol{C}| - \frac{1}{2}\boldsymbol{t}^\top\boldsymbol{C}^{-1}\boldsymbol{t} + N\log\frac{\lambda}{2} - \frac{\lambda}{2}\sum_i \gamma_i \\ &+ a\log b - \log\Gamma(a) + (a-1)\log\lambda - b\lambda \\ &+ c\log d - \log\Gamma(c) + (c-1)\log\sigma^2 - d\sigma^2. \end{aligned} \tag{A.12}$$

Using a similar decomposition strategy as Tipping et al. (2003) they decomposed the covariance matrix $\boldsymbol{C}$ as

$$\begin{aligned}\boldsymbol{C} &= \sigma^2\boldsymbol{I} + \sum_{m\neq i}\sigma^2\gamma_m\boldsymbol{\phi}_m\boldsymbol{\phi}_m^\top + \sigma^2\gamma_i\boldsymbol{\phi}_i\boldsymbol{\phi}_i^\top \\ &= \boldsymbol{C}_{-1} + \sigma^2\gamma_i\boldsymbol{\phi}_i\boldsymbol{\phi}_i^\top,\end{aligned}$$

and calculated the expressions for $\boldsymbol{C}^{-1}$ and $|\boldsymbol{C}|$:

$$\boldsymbol{C}^{-1} = \boldsymbol{C}_{-i}^{-1} - \frac{\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i\boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}}{\gamma_i^{-1}\sigma^{-2} + \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i}, \tag{A.13}$$

$$|\boldsymbol{C}| = |\boldsymbol{C}_{-i}||1 + \sigma^2\gamma_i\boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i|. \tag{A.14}$$

Helgøy and Li (2019) then got the log-likelihood function of $\boldsymbol{\gamma}$ as

$$\mathcal{L}(\boldsymbol{\gamma}) = \mathcal{L}(\boldsymbol{\gamma}_{-i}) + \frac{1}{2}\left[\log\frac{1}{1+\sigma^2\gamma_i r_i} + \frac{\gamma_i\sigma^2\nu_i^2}{1+\sigma^2\gamma_i r_i} - \lambda\gamma_i\right], \tag{A.15}$$

where

$$r_i \equiv \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{\phi}_i \qquad \text{and} \qquad \nu_i \equiv \boldsymbol{\phi}_i^\top\boldsymbol{C}_{-i}^{-1}\boldsymbol{t}. \tag{A.16}$$

By these steps they split the log-likelihood of $\boldsymbol{\gamma}$ into one term including, and one term excluding γ_i, that is $\ell(\gamma_i)$ and $\mathcal{L}(\boldsymbol{\gamma}_{-i})$ respectively. They are then differentiating $\mathcal{L}(\boldsymbol{\gamma})$ with respect to γ_i, that is differentiating $\ell(\gamma_i)$, giving:

$$\frac{d\mathcal{L}(\boldsymbol{\gamma})}{d\gamma_i} = \frac{d\ell(\gamma_i)}{d\gamma_i} = -\frac{1}{2}\left[-\frac{r_i}{\sigma^{-2}+\gamma_i r_i} + \frac{\nu_i^2\sigma^{-2}}{(\sigma^{-2}+\gamma_i r_i)^2} - \lambda\right].$$

Equating this to zero and investigate the expression, they got the maximum likelihood estimate for γ_i:

$$\gamma_i = \begin{cases}\frac{-r_i(r_i+2\lambda\sigma^{-2})+r_i\sqrt{(r_i+2\lambda\sigma^{-2})^2-4\lambda\sigma^{-2}(r_i-\nu_i^2)}}{2\lambda r_i^2} & \text{if } \nu_i^2 - r_i > \lambda\sigma^{-2} \\ 0 & \text{otherwise}\end{cases}. \tag{A.17}$$

When some of the γ_i's are set to zero, the corresponding weights and input vectors are pruned. To optimize other hyperparameters λ, a and b, Helgøy and Li (2019) differentiated Equation (A.12) with respect to each of the parameters and equated to zero they got (Choi and Wette, 1969):

$$\lambda = \frac{2(N+a-1)}{\sum_i\gamma_i + 2b},$$

$$b = \frac{a}{\lambda} \qquad \text{and} \qquad \ln a = \ln \overline{\lambda} - \overline{\ln \lambda} + \psi(a).$$

As all these parameters are dependent on the others, Helgøy and Li (2019) simulated a small sample of λ using Gibbs sampler as described in Park and Casella (2008) to get the initial values for a and b. These estimates are again used to compute λ. In the same way as Tipping et al. (2003), Helgøy and Li (2019) suggested that instead of updating r_i and ν_i in Equation (A.16), it is easier to first calculate the expressions (Helgøy and Li, 2019):

$$\begin{aligned}
R_i &= \boldsymbol{\phi}_i^\top \boldsymbol{C}^{-1} \boldsymbol{\phi}_i \\
&= \sigma^{-2} \boldsymbol{\phi}_i^\top \boldsymbol{\phi}_i - \sigma^{-2} \boldsymbol{\phi}_i^\top \boldsymbol{\phi} \boldsymbol{\Sigma} \boldsymbol{\phi}^\top \boldsymbol{\phi}_i \sigma^{-2}, \\
N_i &= \boldsymbol{\phi}_i^\top \boldsymbol{C}^{-1} \boldsymbol{t}, \\
&= \sigma^{-2} \boldsymbol{\phi}_i^\top \boldsymbol{t} - \sigma^{-2} \boldsymbol{\phi}_i^\top \boldsymbol{\phi} \boldsymbol{\Sigma} \boldsymbol{\phi}^\top \boldsymbol{t} \sigma^{-2}.
\end{aligned}$$

The predictive distribution of the BLS model is given by

$$p(t_* | \boldsymbol{t}, \boldsymbol{\gamma}_{MP}, \sigma^2_{MP}) = \int p(t_* | \boldsymbol{w}, \boldsymbol{\gamma}_{MP}, \lambda_{MP} \sigma^2_{MP}) p(\boldsymbol{w} | \boldsymbol{t}, \boldsymbol{\gamma}_{MP}, \lambda_{MP}, \sigma^2_{MP}) \, d\boldsymbol{w}, \tag{A.18}$$

which is (Helgøy and Li, 2019)

$$t_* | \boldsymbol{t}, \boldsymbol{\gamma}_{MP}, \lambda_{MP}, \sigma^2_{MP} \sim \mathscr{N}(\mu_*, \sigma^2_*), \tag{A.19}$$

where

$$\mu_* = \boldsymbol{\mu}^\top \boldsymbol{\phi}(\boldsymbol{x}_*), \tag{A.20}$$

$$\sigma^2_* = \hat{\sigma}^2 + \boldsymbol{\phi}(\boldsymbol{x}_*)^\top \boldsymbol{\Sigma} \boldsymbol{\phi}(\boldsymbol{x}_*). \tag{A.21}$$

The algorithm of the BLS method by Helgøy and Li (2019) is given in Algorithm 5, and as in the RVM method by Tipping (2001), they fix σ^2 in step one to a scaling of the data variance. Further, with $\check{r}_i$ and $\check{\nu}_i$ being r_i and ν_i given by Equation (A.16) with σ^{-2} excluded, Helgøy and Li (2019) shows the following rewrite of the threshold criteria:

$$\begin{aligned}
r_i^2 - \nu_i &\leq \lambda \sigma^{-2}, \\
(\sigma^{-2} \check{r}_i)^2 - \sigma^{-2} \check{\nu}_i &\leq \lambda \sigma^{-2}, \\
\sigma^{-2} \check{r}_i^2 - \check{\nu}_i &\leq \lambda.
\end{aligned}$$

The relation above, shows that when σ^2 is increasing the more likely it is that γ_i will be

Algorithm 5 Noise-Robust Fast Sparse Bayesian Learning Model (BLS)

```
Fix $\sigma^2$ to a reasonable value
Initialize all $\gamma_i = 0$ and $\lambda = 0$
while convergence criteria are not met do
    Choose a $\gamma_i$
    if $\nu_i^2 - r_i > \lambda\sigma^{-2}$ and $\gamma_i = 0$ then
        Add $\gamma_i$ to the model
    else if $\nu_i^2 - r_i > \lambda\sigma^{-2}$ and $\gamma_i > 0$ then
        Re-estimate $\gamma_i$
    else if $\nu_i^2 - r_i < \lambda\sigma^{-2}$ and $\gamma_i < 0$ then
        Prune observation $i$ from the model (set $\gamma_i = 0$)
    end if
    Update $\mathbf{\Sigma}$, $\boldsymbol{\mu}$, $\nu_i$, $r_i$, $\lambda$, $a$ and $b$
end while
```

set to infinity and hence that the basis function is pruned. This illustrates the robustness in the model towards the noise variance, and thus how this model can reduce the risk of overfitting when data is noisy.

Simultaneous Feature and Sample Selective BLS

In this feature selective method we are using the same kind of sparse framework as in the DRVM model. That is defining feature weights, and kernel basis functions that includes these new weights, given by Equation (3.2), (3.3) and (3.4). Using this framework the posterior distribution over all unknown parameters is given by

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2 | \boldsymbol{t}) = \frac{p(\boldsymbol{t} | \boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2)}{p(\boldsymbol{t})},$$

with the predictive distribution:

$$p(t^* | \boldsymbol{t}) = \int p(t^* | \boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2 | \boldsymbol{t}) \, d\boldsymbol{w} \, d\boldsymbol{\vartheta} \, d\boldsymbol{\gamma} \, d\boldsymbol{\beta} \, d\lambda \, d\sigma^2. \tag{A.22}$$

We are then again decomposing in the same way as Tipping (2001), which gives

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2 | \boldsymbol{t}) = p(\boldsymbol{w}, \boldsymbol{\vartheta} | \boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) p(\boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2 | \boldsymbol{t}). \tag{A.23}$$

From here we can find the simultaneous posterior distribution over the feature weights $\boldsymbol{\vartheta}$ and sample weights $\boldsymbol{w}$ by

$$p(\boldsymbol{w}, \boldsymbol{\vartheta} | \boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) = \frac{p(\boldsymbol{t} | \boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2) p(\boldsymbol{w} | \boldsymbol{\gamma}, \sigma^2) p(\boldsymbol{\vartheta} | \boldsymbol{\beta})}{p(\boldsymbol{t} | \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2)}. \tag{A.24}$$

In Equation (A.24) the likelihood of the targets is again similar to the RVM case with kernel basis functions dependent on the feature weights $\boldsymbol{\vartheta}$, that is the Gaussian distribution given by Equation (4.1). Further, the distribution over the sample weights $\boldsymbol{w}$ is identical to the one in the original BLS model, given by Equation (A.4) and the distribution over the feature weights is identical to the dimensionality reducing RVM method, given by Equation (3.5) and (3.6), with the hyper hyperparameters fixed to be $e = f = 10^{-4}$.

To find the simultaneous posterior distribution over the weights we are using the same procedure as for the dimensionality reducing method based on the Relevance Vector Machine, that is a Laplacian approximation. The first step is to take the logarithm of Equation (A.24), giving

$$\begin{aligned}\ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) =& \ln p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2) + \ln p(\boldsymbol{w}|\boldsymbol{\gamma}, \sigma^2) \\ &+ \ln p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) - \ln p(\boldsymbol{t}|\boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2).\end{aligned} \tag{A.25}$$

Only considering the terms that is including the sample weights, we get the log posterior with respect to the sample weights $\boldsymbol{w}$ given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}) &= \ln p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2) + \ln p(\boldsymbol{w}|\boldsymbol{\gamma}, \sigma^2) \\ &= -\frac{1}{2}\sigma^{-2}||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}||^2 + \boldsymbol{w}^\top \boldsymbol{\Lambda}^{-1}\boldsymbol{w}.\end{aligned} \tag{A.26}$$

Equation (A.26) is the logarithm with respect to the sample weights $\boldsymbol{w}$ in the BLS method, given by the logarithm of Equation (A.6), just with the inclusion of individual feature weights $\boldsymbol{\vartheta}$ in the kernel basis functions. Thus, the maximization give the same result and we have that (A.24) with respect to $\boldsymbol{w}$ is approximately

$$\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}}),$$

where $\boldsymbol{\mu}_{\boldsymbol{w}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ is given by Equation (A.7) and (A.8) with $\boldsymbol{\vartheta}$ included in every kernel basis functions.

Considering only the terms of the likelihood function (A.25) that is including the feature weights $\boldsymbol{\vartheta}$, we get the log posterior with respect to $\boldsymbol{\vartheta}$ by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\vartheta}) &= \ln p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2) + \ln p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) \\ &= -\frac{1}{2}\sigma^{-2}||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}||^2 + \boldsymbol{\vartheta}^\top \boldsymbol{B}^{-1}\boldsymbol{\vartheta}.\end{aligned} \tag{A.27}$$

Equation (A.27) is identical to the likelihood function of $\boldsymbol{\vartheta}$ from Equation (4.4) of the DRVM method, and we get the same Laplace approximation with respect to $\boldsymbol{\vartheta}$. All together, using Laplace's approximation, we get that

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) \approx \mathscr{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}) \cdot \mathscr{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}}), \tag{A.28}$$

where $\boldsymbol{\vartheta}_{MP}$, $\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{w}}$ and $\boldsymbol{\mu}_{\boldsymbol{w}}$ are given by Equation (4.6), (4.7), (A.8) and (A.7) respectively, with the inclusion of $\boldsymbol{\vartheta}$ in the kernel basis functions.

From Equation (A.23) we are not able to find the second term analytically, and we are therefore approximating it using the simultaneous distribution over alle parameters, as Helgøy and Li (2019) did in the original BLS method. That is the approximation

$$\begin{aligned} p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2|\boldsymbol{t}) &= \frac{p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2)}{p(\boldsymbol{t})} \\ &\propto p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2), \end{aligned}$$

as we can ignore the distribution of the targets $\boldsymbol{t}$ as the MAP-estimates of the other hyperparameters will not depend on it. This simultaneous distribution can be decomposed into

$$p(\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) = \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\vartheta}, \sigma^2)p(\boldsymbol{\vartheta}|\boldsymbol{\beta})p(\boldsymbol{w}|\boldsymbol{\gamma}, \sigma^2)p(\boldsymbol{\gamma}|\lambda)p(\lambda)p(\sigma^2)}{p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2)}, \tag{A.29}$$

and by taking the logarithm with respect to $\boldsymbol{\gamma}$ we get

$$\begin{aligned} \mathcal{L}(\boldsymbol{\gamma}) &= \ln p(\boldsymbol{w}|\boldsymbol{\gamma}, \sigma^2) + \ln p(\boldsymbol{\gamma}|\lambda) - \ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\gamma}, \boldsymbol{\beta}, \lambda, \sigma^2) \qquad \text{(A.30)} \\ &= -\frac{1}{2}\ln|\boldsymbol{\Lambda}| + \frac{1}{2}\ln|\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}| - \frac{1}{2}\boldsymbol{w}^\top\boldsymbol{\Lambda}^{-1}\boldsymbol{w} \\ &\quad + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{\mu}_{\boldsymbol{\vartheta}})^\top{\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}^{-1}(\boldsymbol{w} - \boldsymbol{\mu}_{\boldsymbol{\vartheta}}) - \frac{\lambda}{2}\sum_i \gamma_i. \end{aligned}$$

Further we know the following relation from the deduction of the posterior distribution over the sample weights in the original BLS method:

$$|\boldsymbol{\Lambda}|^{-\frac{1}{2}}|\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}|^{\frac{1}{2}} = (\sigma^2)^{-\frac{N}{2}}|\boldsymbol{C}_{\boldsymbol{\vartheta}}|^{-\frac{1}{2}}, \tag{A.31}$$

with $\boldsymbol{C}_{\boldsymbol{\vartheta}}$ being the matrix given by Equation (A.9) in the original BLS with the kernel functions dependent on the feature weights $\boldsymbol{\vartheta}$. From the same equations we have that

$$\boldsymbol{w}^\top\boldsymbol{\Lambda}^{-1}\boldsymbol{w} - (\boldsymbol{w} - \boldsymbol{\mu}_{\boldsymbol{\vartheta}})^\top{\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}}^{-1}(\boldsymbol{w} - \boldsymbol{\mu}_{\boldsymbol{\vartheta}}) = \boldsymbol{t}^\top{\boldsymbol{C}_{\boldsymbol{\vartheta}}}^{-1}\boldsymbol{t} - \sigma^{-2}||\boldsymbol{t} - \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}\boldsymbol{w}||^2, \tag{A.32}$$

where the only term on the left hand side that includes the sample weights $\boldsymbol{w}$ is the first one. By inserting the relations given by Equation (A.31) and (A.32) into Equation (A.30), and only considering the terms that includes $\boldsymbol{\gamma}$, we get

$$\mathcal{L}(\boldsymbol{\gamma}) = -\frac{1}{2}\ln|\boldsymbol{C}_{\boldsymbol{\vartheta}}| - \frac{1}{2}\boldsymbol{t}^\top \boldsymbol{C}_{\boldsymbol{\vartheta}}{}^{-1}\boldsymbol{t} - \frac{\lambda}{2}\sum_i \gamma_i.$$

We recognize this equation as the log posterior distribution over the sample weights given in the original BLS method, by the first line of Equation (A.12) and further by the decomposed form in Equation (A.15). We just have to remember that the kernel basis functions is dependent on the feature weights $\boldsymbol{\vartheta}$. Hence the maximum value of γ_i is given by Equation (A.17) with the inclusion of $\boldsymbol{\vartheta}$ in the kernel functions, that is:

$$\gamma_i = \begin{cases} \frac{-r_{\boldsymbol{\vartheta},i}(r_{\boldsymbol{\vartheta},i}+2\lambda\sigma^{-2})+r_{\boldsymbol{\vartheta},i}\sqrt{(r_{\boldsymbol{\vartheta}i}+2\lambda\sigma^{-2})^2-4\lambda\sigma^{-2}(r_{\boldsymbol{\vartheta},i}-\nu_{\boldsymbol{\vartheta},i}^2)}}{2\lambda r_{\boldsymbol{\vartheta},i}^2+\lambda\sigma^{-2}} & \text{if } \nu_{\boldsymbol{\vartheta},i}^2 - r_{\boldsymbol{\vartheta},i} > \lambda\sigma^{-2} \\ 0 & \text{otherwise} \end{cases}.$$

To find the estimate of the hyperparameter corresponding to the feature weights, we have to take the logarithm of Equation (A.29) with respect to $\boldsymbol{\beta}$. This is:

$$\mathcal{L}(\boldsymbol{\beta}) = \ln p(\boldsymbol{\vartheta}|\boldsymbol{\beta}) - \ln p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2),$$

which considered only with respect to the feature weights $\boldsymbol{\vartheta}$ is the exact same expression as we got in the RVM dimensionality reducing method when investigating with respect to the sample weights $\boldsymbol{\vartheta}$, that is given by Equation (3.25). Hence the rest will follow the same argumentation, and we get the estimate for β_i by Equation (3.29).

The other hyperparameters will have the same estimates as in the original BLS model, just with the inclusion of the feature weights $\boldsymbol{\vartheta}$ in the kernel basis functions.

In a similar manner as for the predictive distribution in DRVM and FRVM, we get the predictive distribution:

$$p(t_*|\boldsymbol{t}) = \int p(t_*|\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}_{MP}, \boldsymbol{\beta}_{MP}, \lambda_{MP}, \hat{\sigma}^2)p(\boldsymbol{w}, \boldsymbol{\vartheta}|\boldsymbol{t}, \boldsymbol{\gamma}_{MP}, \boldsymbol{\beta}_{MP}, \lambda_{MP}, \hat{\sigma}^2)\, d\boldsymbol{w}\, d\boldsymbol{\vartheta}.$$

By using the relation from Equation (A.28) we get that this distribution can be approximated by the integral

$$p(t_*|\boldsymbol{t}) = \int p(t_*|\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}_{MP}, \boldsymbol{\beta}_{MP}, \lambda_{MP}, \hat{\sigma}^2)\mathcal{N}(\boldsymbol{\vartheta}_{MP}, \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}) \cdot \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}})\, d\boldsymbol{w}\, d\boldsymbol{\vartheta},$$

which by integrating out the feature weights gives

$$p(t_*|\boldsymbol{t}) = \int p(t_*|\boldsymbol{w}, \boldsymbol{\vartheta}, \boldsymbol{\gamma}_{MP}, \boldsymbol{\beta}_{MP}, \lambda_{MP}, \hat{\sigma}^2)\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}})\, d\boldsymbol{w}.$$

The equation above is the predictive distribution from the original BLS method given by Equation (A.18), just with the inclusion of the feature weights $\boldsymbol{\vartheta}$ in the kernel basis functions. Thus, we can predict for new target variables in the dimensionality reducing method using the predictive distribution from the earlier described BLS method. This is given by Equation (A.19), (A.20) and (A.21), just with the inclusion of the separate feature weights $\boldsymbol{\vartheta}$ in the kernel basis functions, and the parameters estimated using the approach in this chapter. That is

$$t_*|\boldsymbol{t}, \boldsymbol{\gamma}_{MP}, \sigma^2_{MP} \sim \mathscr{N}(\mu_*, \sigma^2_*),$$

with

$$\mu_* = \boldsymbol{\mu}_{\boldsymbol{\vartheta}}{}^\top \boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*),$$

$$\sigma^2_* = \hat{\sigma}^2 + \boldsymbol{\phi}_{\boldsymbol{\vartheta}}{}^\top(\boldsymbol{x}_*)\boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}\boldsymbol{\phi}_{\boldsymbol{\vartheta}}(\boldsymbol{x}_*).$$

A.1.3 Challenges with Establishing the Algorithms

After developing the theory behind the two methods, I had to stop working with them to prioritize other topics. The next challenge is to figure out how the algorithms of the two methods should be. It is not straight forward to do the updating simultaneously and iteratively with respect to both sample and feature weights when only considering one hyperparameter at a time.

A.2 Code Snippets from the DRVM Learning

The following section shows part of the MATLAB-code for fitting the DRVM model, which is highly inspired by Tipping (2016) and the code developed by the authors of Jiang et al. (2019). Starting out with the code for updating the covariance matrix and the mean vector with respect to the sample weights $\boldsymbol{w}$:

```
PHI2     = PHI'*PHI;
Hessian  = PHI2*invvar + A;
U        = chol(Hessian);
Ui       = inv(U);
SIGMA    = Ui*Ui';


w = invvar*SIGMA*PHI'*t;
```

Code for updating the hyperparameters $\boldsymbol{\alpha}$, and selecting the ones that are less than a given threshold:

```
diagSig      = sum(Ui.^2, 2);
```

```
gamma       = 1 - alpha(used).*diagSig;
alpha(used) = gamma ./ w(used).^2;
used        = find(alpha < MAXIMUM);
w_nz        = w(used);
alpha_nz    = alpha(used);
```

Calculating the residual and updating the σ^2 estimate:

```
y      = PHI*w;
e      = (t - y);
ED     = e'*e;
var    = ED/(ndata-sum(gamma));
invvar = var^(-1);
```

Code for checking the maximum change and stop updating $\boldsymbol{w}$ if satisfied:

```
if i > 5 && max(abs(w_nz - w_old(used))) < MINIMUM
 update_w = false;
 if ~update_t; break; end
end
w_old = w;
```

Calculating the mean vector with respect to the feature weights $\boldsymbol{\vartheta}$:

```
PHI_used = ker(trainX, trainX(used, :), theta);

y = PHI_used*w + b;
e = t - y;

sigmoid_theta = sigmoid(theta, Lambda);
sigmoid_theta(sigmoid_theta < realmin) = realmin;

data_term   = - 1/2*invvar*(e'*e);
regulariser = beta'*(theta.^2)/2;
Q_out       = data_term + sum(log(sigmoid_theta));
Q           = Q_out - regulariser;
```

Using Newton step to approximate the mean vector:

```
for j  = 1:its
    e  = t - y;
    D  = Dfast(w, dist(:, used, :), PHI_used, Mused);
    kB = Lambda*(1-sigmoid_theta);
    g  = -beta.*theta + invvar*D'*e + kB;
    % See if converged
    if j >= 2 && norm(g)/Mused < GRAD_STOP
        break
    end

    OB          = diag(Lambda*Lambda*(sigmoid_theta.* (1-
       sigmoid_theta)) + beta);
    D2          = sum(D'.*D', 2);
    Hessian     = diag(OB) + invvar*D2;
    Hessian     = Hessian.^(-1);
    delta_theta = g.*Hessian;
    delta       = 0.5;
    while delta > 2^-10
        theta_new = theta + delta*delta_theta;
        PHI_used  = ker(trainX, trainX(used,:), theta_new);
        y              = PHI_used*w + b;
        data_term_new = - 1/2*invvar*(e'*e);
        regulariser   = beta'*(theta_new.^2)/2;
        sigmoid_theta = sigmoid(theta_new, Lambda);
        sigmoid_theta(sigmoid_theta < realmin) = realmin;
        Q_new = data_term_new - regulariser + sum(log(
           sigmoid_theta));
        if Q_new > Q
            Q         = Q_new;
            Q_out     = Q + regulariser;
            theta     = theta_new;
            data_term = data_term_new;
            delta     = 0;
        else
            delta     = delta/2;
        end
    end

    if delta
        break;
    end

end
```

Updating the hyperparameters $\boldsymbol{\beta}$, selecting the ones that are smaller than a given threshold and checking the maximal change:

```
gamma = 1-beta.*Hessian;
beta  = gamma./theta.^2;
Ui = diag(sqrt(Hessian));
Q     = Q/Mused;
Q_out = Q_out/Mused;
theta_used = find(beta < MAXIMUM);
theta_nz   = theta(theta_used);
beta_nz    = beta(theta_used);

if i > 5 && max(abs(theta_nz - theta_old(theta_used))) <
   MINIMUM
 update_t = false;
 if ~update_w; break; end
end
theta_old = theta;
```

A.3 Explanation of the Boston Housing Features

This section gives a direct copy of Tipping (1996)s explanation of the features in the Boston Housing data set used in the experimental part. The data set was first published by Harrison Jr and Rubinfeld (1978), and includes the following features (Tipping, 1996):

crim	per capita crime rate by town
zn	proportion of residential land zoned for lots over 25000 sq.ft.
indus	proportion of non-retail business acres per town
chas	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
nox	nitric oxides concentration (parts per 10 million)
rm	average number of rooms per dwelling
age	proportion of owner-occupied units built prior to 1940
dis	weighted distances to five Boston employment centres
rad	index of accessibility to radial highways
tax	full-value property-tax rate per $10,000$
ptratio	pupil-teacher ratio by town
b	$1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
lstat	% lower status of the population
medv	Median value of owner-occupied homes in 1000’s