3F8 FULL TECHNICAL REPORT BAYESIAN LOGISTIC CLASSIFICATION

NAME: MIHAI VARSANDAN

CRSID: mv436
COLLEGE: GIRTON
DATE: 04/03/2019

I. INTRODUCTION

The aim of this project was to implement a Bayesian Classifier to the data and then compare it against Maximum Likelihood(ML) Logistic Classification studied in the lab and with the Maximum of a Posterior(MAP).

It was shown in the coursework the ML Classifier has a tendency to overfit in certain cases. This can be avoided by introducing a prior distribution on the weights which reflects our initial guess about how the weights would be distributed. Up to this point, enough information has been acquired to develop a MAP classifier because the MAP classifier is just the ML classier with accounting the prior distribution.

To improve the MAP classifier it could be possible to integrate over all possible solutions to create a predictive distribution. This process results in a Bayesian Classifier.

II. THEORY

During this report, **bold** values will be considered as column vectors while characters with both **bold** will be considered matrices.

A. Initial Model

Likelihood

The equation for the likelihood:

$$p(\boldsymbol{y} \mid \underline{\boldsymbol{X}}, \boldsymbol{w}) = \prod_{n=1}^{N} \sigma(\boldsymbol{w}^T \boldsymbol{x}_n)^{y_n} (1 - \sigma(\boldsymbol{w}^T \boldsymbol{x}_n))^{1-y_n} \quad (1).$$

Equation 1 was given in the lab handout. In this case 'N' is equal to: N=D(training data) +1 . The 'one' accounts for the bias term.

Prior

A prior distribution $p(\boldsymbol{w})$ is chosen to encapsulate any prior knowledge about the weights. The equation for the prior is given in the FTR handout. It is equal to:

$$p(\boldsymbol{w}) = \frac{1}{Z} \exp\left(\frac{1}{2\sigma_0^2} \boldsymbol{w}^T \boldsymbol{w}\right) \quad (2)$$

This implies that the weights distribution was chosen so that it the weights would have a '0' mean and 'σ_0^2'.

Posterior

Using Bayesian rule it can be found that:

$$p(\boldsymbol{w} \mid \underline{\boldsymbol{X}}, \boldsymbol{y}) = \frac{p(\boldsymbol{y} \mid \underline{\boldsymbol{X}}, \boldsymbol{w}) p(\boldsymbol{w})}{p(\underline{\boldsymbol{X}})} \propto p(\boldsymbol{y} \mid \underline{\boldsymbol{X}}, \boldsymbol{w})\ p(\boldsymbol{w}) \quad (3)$$

From now on it has been decided to work in the form of logarithms as it will simplify the maths and it will prevent computational overflow. Let '$\mathscr{L}^*(\boldsymbol{w})$' be the log of the posterior. It will be equal to:

$$\mathscr{L}^*(\boldsymbol{w}) = \underbrace{\mathscr{L}(\boldsymbol{w})}_{\text{log likelihood}} - \underbrace{\frac{1}{2\sigma_0^2}\boldsymbol{w}^T\boldsymbol{w}}_{\text{log of prior}} \quad (4)$$

Using Equation in 1 in logarithm mode:

$$\mathscr{L}^*(\boldsymbol{w}) = \sum_{n=1}^{N} y_n \log(\sigma(\boldsymbol{w}^T\boldsymbol{x}_n)) + (1-y_n)\log(1-\sigma(\boldsymbol{w}^T\boldsymbol{x}_n)) - \frac{1}{2\sigma_0^2}\boldsymbol{w}^T\boldsymbol{w} \quad (5)$$

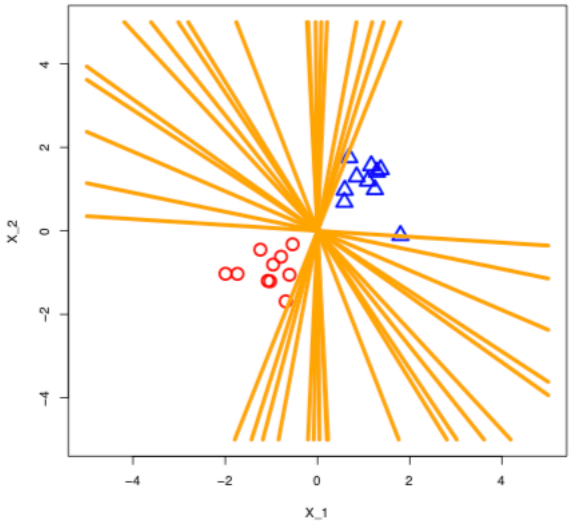

Figure 1: Multiple Weights Fitting the Dataset

The gradient of the log of the posterior is equal to:

$$\frac{\partial \mathscr{L}^*}{\partial \boldsymbol{w}} = \sum_{n=1}^{N} (y_n - \sigma(\boldsymbol{w}^T\boldsymbol{x}_n))\boldsymbol{x}_n - \frac{1}{\sigma_0^2}\boldsymbol{w} \quad (6)$$

Equation 5 and 6 are used to find the '$\boldsymbol{w}_{\text{MAP}}$' using the minimum optimisation software **scipy.optimize.fmin_l_bfgs_b**. After the '$\boldsymbol{w}_{\text{MAP}}$' is found the MAP classifier can be implemented.
As can be seen in Figure 1 multiple weights can fit the data very well. The problem is that in high dimensions(many basis functions) the data will be linearly inseparable. So, to overcome this problem a Bayesian Inference model is introduced

B. Laplace Approximation

To produce a Bayesian predictive distribution an integral over all possible weights is required. This implies that Bayesian looks at how likely the weights are then it averages the distribution for all weights possible. Let '$\underline{\boldsymbol{X}}^*$' be the test data and '$\boldsymbol{y}^*$' is predicted class. The equation is:

$$p(\boldsymbol{y}^* \mid \underline{\boldsymbol{X}}^*, \boldsymbol{y}) = \int p(\boldsymbol{y}^* \mid \underline{\boldsymbol{X}}^*, \boldsymbol{w})\; p(\boldsymbol{w} \mid \boldsymbol{y})\; dw \quad (7)$$

However, this equation cannot be computed as it is computationally infeasible to iterate over all possible weights. Therefore an approximation is made, called Laplace approximation. The posterior distribution '$p(\boldsymbol{w} \mid \boldsymbol{y})$' is approximated to a Gaussian distribution '$q(\boldsymbol{w})$'. In figure 2 , it is showed the process behind the Laplace Approximation. The probability in yellow is equal to the logistic function which corresponds to the likelihood multiplied by the posterior distribution leading to a very difficult computational integral. It is desired to approximate the yellow distribution to a Gaussian shown in the red. To achieve this Gaussian approximation it is required to the mode of the Gaussian equal to the mode of the posterior distribution. Therefore the Laplace Approximation is:

$p(\boldsymbol{w} \mid \boldsymbol{y}) \approx q(\boldsymbol{w})$ where $q(\boldsymbol{w}) \sim \mathcal{N}(\boldsymbol{w} \mid \boldsymbol{\mu}_a, \boldsymbol{\sigma}_a)$

and $\boldsymbol{\mu}_a = \boldsymbol{w}_{MAP}^T \underline{\boldsymbol{X}}^*, \quad \boldsymbol{\sigma}_a^2 = \underline{\boldsymbol{X}}^{*T} \underline{\boldsymbol{S}}_n \underline{\boldsymbol{X}}^* \quad (8)$

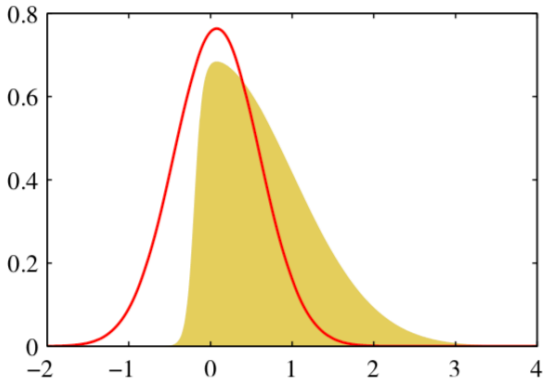

Figure 2: Laplace Approximation

As '$\boldsymbol{w}_{\text{MAP}}$' has been found, only the covariance matrix '$\underline{\boldsymbol{S}}_n$' is to be found. '$\underline{\boldsymbol{S}}_n$' is equal to the Hessian of the posterior distribution. In this lab the Hessian is approximated to be:

$$\underline{\boldsymbol{S}}_n^{-1} = \underline{\boldsymbol{S}}_0^{-1} + \sum_{n=1}^{N} \sigma(\boldsymbol{w}^T \boldsymbol{x}_n)(1 - \sigma(\boldsymbol{w}^T \boldsymbol{x}_n))\boldsymbol{x}_n \boldsymbol{x}_n^T = \underline{\boldsymbol{S}}_0^{-1} + \underline{\boldsymbol{X}}_{scaled}^T \underline{\boldsymbol{X}} \quad (9)$$

where $\underline{\boldsymbol{S}}_0 = \sigma_0^2 \underline{\boldsymbol{I}}$ and $\underline{\boldsymbol{X}}_{scaled}$ is just the dataset scaled by $\sigma(\boldsymbol{w}^T \boldsymbol{x}_n)(1 - \sigma(\boldsymbol{w}^T \boldsymbol{x}_n))$.
Therefore by inverting '$\underline{\boldsymbol{S}}_n^{-1}$','$\underline{\boldsymbol{S}}_n$' can be found. The complete derivation of '$\underline{\boldsymbol{S}}_n$','$\boldsymbol{\mu}_a$' and $\boldsymbol{\sigma}_a$ can be seen in Chapter 4.5.1 and 4.5.2 of Bishop's book.
It can be observed that the major problem with Laplace Approximation is the lack of accuracy when trying to model '$p(\boldsymbol{w} \mid \boldsymbol{y})$' as '$q(\boldsymbol{w})$'. However, for a large amount of data, the accuracy will increase leading to a better approximation and therefore to a better prediction according to Bayesian central limit theorem.

C. Model Evidence

In equation 3 it was explained that the posterior is proportional to the likelihood multiplied by the prior. To make it equal a normalisation constant must be introduced. From Bayesian Rule the model evidence to make posterior a valid distribution is:

$$p(\underline{\boldsymbol{X}}) = \int p(\underline{\boldsymbol{X}} \mid \boldsymbol{w}) p(\boldsymbol{w}) d\boldsymbol{w} \quad (10)$$

Let $p(\underline{\boldsymbol{X}}) = Z$ and $p(\underline{\boldsymbol{X}} \mid \boldsymbol{w})p(\boldsymbol{w}) = f(\boldsymbol{w})$. By Taylor expanding $f(\boldsymbol{w})$ and taking natural logarithms while at the same time ignore higher order terms the $f(\boldsymbol{w})$ will be equal to :

$$\ln(f(\boldsymbol{w}) \approx \ln(\boldsymbol{w}_{MAP}) - \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{MAP})^T \underline{\boldsymbol{A}} (\boldsymbol{w} - \boldsymbol{w}_{MAP}) \quad (11)$$

Where :

- At $\boldsymbol{w} = \boldsymbol{w}_{MAP}$, $\nabla f(\boldsymbol{w}) \mid_{\boldsymbol{w}=\boldsymbol{w}_{MAP}} = 0$ as $\boldsymbol{w}_{MAP}$ will be the mode of Gaussian.
- $\underline{\boldsymbol{A}} = \nabla \nabla - \ln f(\boldsymbol{w}) \mid_{\boldsymbol{w}=\boldsymbol{w}_{MAP}}$ will be the Hessian

Therefore by taking the exponential both sides it is found that:

$$f(\boldsymbol{w}) \approx f(\boldsymbol{w}_{MAP}) \exp(-\frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{MAP})^T \underline{\boldsymbol{A}} (\boldsymbol{w} - \boldsymbol{w}_{MAP})) \quad (12)$$

So therefore using Equation 10, Z can be found to be:

$$Z = \int f(\boldsymbol{w}) d\boldsymbol{w} = \int f(\boldsymbol{w}_{MAP}) \exp(-\frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{MAP})^T \underline{\boldsymbol{A}} (\boldsymbol{w} - \boldsymbol{w}_{MAP}))) d\boldsymbol{w}$$

$$Z = f(\boldsymbol{w}_{MAP}) \frac{(2\pi)^{M/2}}{|\underline{\boldsymbol{A}}|^{1/2}} \quad (13)$$

Therefore by reverting back to the initial probabilities the model evidence is found to be :

$$\ln(p(\underline{\boldsymbol{X}})) \approx \ln(p(\underline{\boldsymbol{X}} \mid \boldsymbol{w})) + \underbrace{\ln(p(\boldsymbol{w})) + \frac{M}{2} \ln(2\pi) - \frac{1}{2} \ln(|\underline{\boldsymbol{A}}|)}_{OccamFactor} \quad (14)$$

The last three terms are penalising the model evidence. It is further assumed that the prior Gaussian distribution is broad and the Hessian $\underline{\boldsymbol{A}}$ is full rank which allows Equation 14 to be further simplified to:

$$\ln(p(\underline{\boldsymbol{X}})) \approx \ln(p(\underline{\boldsymbol{X}} \mid \boldsymbol{w})) - \frac{1}{2} M \ln N \quad (15)$$

This simplification is very useful because computational complexity required to calculate the Hessian which is avoided using Equation 15.

However, to implement the Laplace Approximation, it is not necessary to calculate the normal constant the proportionality is used.

D. Predictive Distribution

Using the Laplace Approximation the new Bayesian expression is equal to:

$$p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \boldsymbol{y}) = \int p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \boldsymbol{w})\ q(\boldsymbol{w})\ dw = \int \sigma(a) \mathcal{N}(a \mid \mu_a, \sigma_a^2) \quad (16)$$

But even after the Laplace Approximation the Equation 16 is not possible to compute. A further approximation is introduced as a sigmoid function is equal to probit function with the x-axis rescaled. The result of Equation 16 would equal to a probit function due to properties of a Gaussian. Therefore the reverse process is repeated to transform it back to a sigmoid function:

$$\sigma(a) = \phi(\lambda a) \text{where } \lambda = \sqrt{\frac{\pi}{8}}$$

$$\therefore p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \boldsymbol{y}) = \int \phi(\lambda a) \mathcal{N}(a \mid \mu_a, \sigma_a^2) = \phi\left(\frac{\mu_a}{(\lambda^{-2} + \sigma_a^2)^{1/2}}\right) \approx \sigma(\kappa(\sigma_a^2)\mu_a) \quad (17)$$

$$\text{where } \kappa(\sigma_a^2) = (1 + \pi\sigma_a^2/8)^{-1/2}$$

It is therefore shown that the Bayesian Classifier is equal to:

$$p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \boldsymbol{y}) = \sigma(\kappa(\sigma_a^2)\mu_a) \quad (18)$$

Because of the fact the threshold $p(\boldsymbol{y}^* = 1)$ to be 0.5 the MAP classifier and the Bayesian classier will predict the same $\boldsymbol{y}^*$ for new $\boldsymbol{x}^*$. This is because as the threshold is 0.5, $\boldsymbol{\mu}_a$ will be 0 so it implies that $\boldsymbol{w}^T \underline{\boldsymbol{X}}^* = 0$.

II. PERFORMANCE

A. MAP vs Bayesian

One of the scopes of this coursework was to explain how the Bayesian approach differs from the MAP solution. In the previous section, the theoretical differences were discussed. Now, the results of both approaches will be analysed and compared.

I
n both cases the RBF basis function is used. Firstly, it was decided to fix the variance of the prior distribution $\sigma_0^2 = 1$ and the width of the RBF basis function, $l = 0.1$. A clear description of the MAP approach vs the Bayesian approach is given below.

MAP

- Using Equations 5 and 6 together with the optimisation software the $\boldsymbol{w}_{MAP}$ is found
- Using the same logistic function that was developed in the lab the new predictive distribution is shown in Figure 2 below.

Bayesian

- Using Equations 5 and 6 together with the optimisation software the $\boldsymbol{w}_{MAP}$ is found
- Find the Hessian $\underline{\boldsymbol{S}}_n$ using Equation 9
- Find the Laplace Approximation $q(\boldsymbol{w})$ to the posterior by finding $\boldsymbol{\mu}_a$ and $\boldsymbol{\sigma}_a$ using Equation 8
- Using Equation 17 a new logistic classifier is developed find the predictive distribution. The results are shown in Figure 3 below

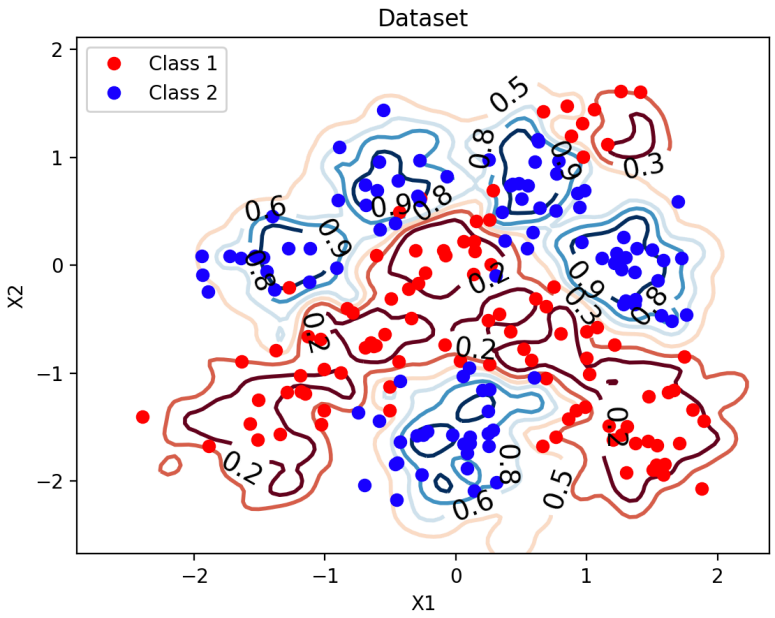

Figure 2: MAP Prediction

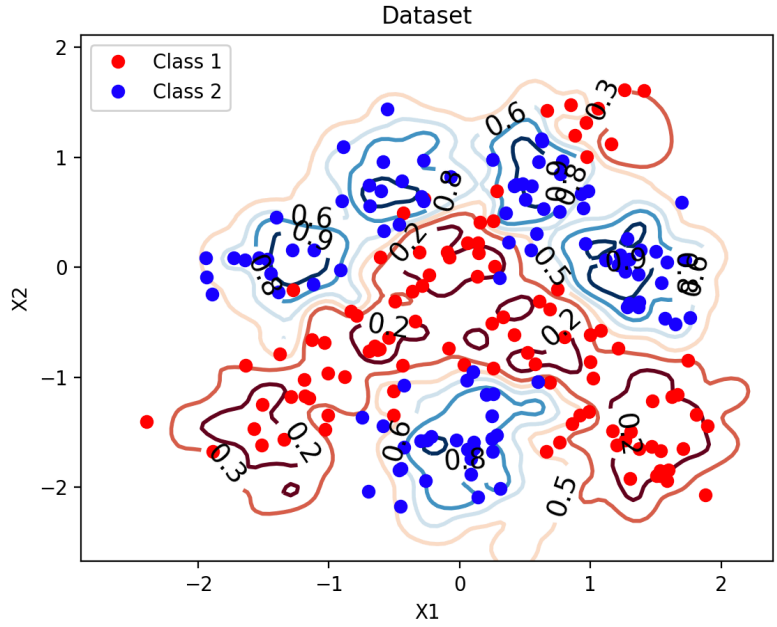

Figure 3: Bayesian Prediction

The confusion matrices for training data and test data of both the MAP prediction and Bayesian Prediction are seen below:

Table 1: Confusion Matrix: MAP for test data

		Predicted label	
		0	1
True Label	0	91.1%	8.9%
	1	14.14%	85.56%

Table 2: Confusion Matrix: Bayesian for test data

		Predicted label	
		0	1
True Label	0	91.1%	8.9%
	1	14.14%	85.56%

Table 3: Confusion Matrix: MAP for train data

		Predicted label	
		0	1
True Label	0	94.57%	5.43%
	1	5.37%	95.63%

Table 4: Confusion Matrix: Bayesian for train data

		Predicted label	
		0	1
True Label	0	94.57%	5.43%
	1	5.37%%	95.63%

As it can be observed from both the contour and the confusion matrices both the MAP and the Bayesian will produce the same prediction models. This is as expected conform to the theory in section 1.D, which states that because our threshold value to decide whether a point belongs to a class is 0.5.

To try to see the underlying differences between the two approaches it is necessary to look at the average log-likelihoods for both methods considering again both the test data and the train data.

Table 5:Average Log-Likelihood comparison

	Training Data	Test Data
MAP	−0.2166	−0.3243
BAYESIAN	−0.256	−0.3482

From Table 5 it can be observed that the MAP performs better than the Bayesian method as both the average log-likelihoods of test and train data are bigger. This is expected due to the fact the MAP classifier will only use the weights that maximise the posterior while the Bayesian classifier will find the average distribution taking into account all possible weights. Therefore lower weights will decrease the log-likelihood as seen in Table 5. The advantage of Gaussian is hidden just by looking at the results. However, for a high dimensional data, the MAP classifier tends to overfit the data while the Bayesian method is less prone to overfitting.

There is a final point that needs to be addressed which is the why do the contours of the MAP classifier and Bayesian classifier are the same. By looking at Equation 9, noting that the logistic function will take values between 0 and 1 and that $\boldsymbol{x}_n \boldsymbol{x}_n^T$ values are greater than 0, it observed that for a big datasets $\underline{\boldsymbol{S}}_n^{-1}$ grows more positive leading to $\underline{\boldsymbol{S}}_n$ to become smaller and therefore the variance of the Laplace Approximation, $\boldsymbol{\sigma}_a = 0$. Therefore Equation 18 becomes:

$$p(\boldsymbol{y}^* | \boldsymbol{x}^*, \boldsymbol{y}) = \sigma((\mu_a) = \overbrace{\sigma(\boldsymbol{w}_{MAP}^T \underline{\boldsymbol{X}})}^{\text{MAP Classifier}} \quad (18)$$

This explains why the contours in both plots are the same.

III. Grid Optimisation

In the last part of this report, a method of tuning the hyperparameters 'σ_0' and 'l'. It is suggested to look at Equations 14 and 15 in the 1st section which contains the expression for the model evidence and to find the hyperparameters that maximise this function. Indeed it is possible to compute the model evidence using Equation 14. It is noted from Equation 15 that maximising the model evidence means maximising the log-likelihood therefore, the hyperparameters would tend to the ML(maximum likelihood) solution and σ_0^2 would tend to infinity. It is tempted to look at both the training data and test data and do the ML solution for both, however, that way a model that would fit another test data would fail as now the model is not generalised. The ML optimisation for the RBF width and the Variance are in the figure seen below. It was decided that the logarithmic spaced grid should be used so that it would be easy to visualise what is happening at low values. Also, instead of 10x10 grid, a 50x50 grid was chosen so that the shape of contours would be better defined.

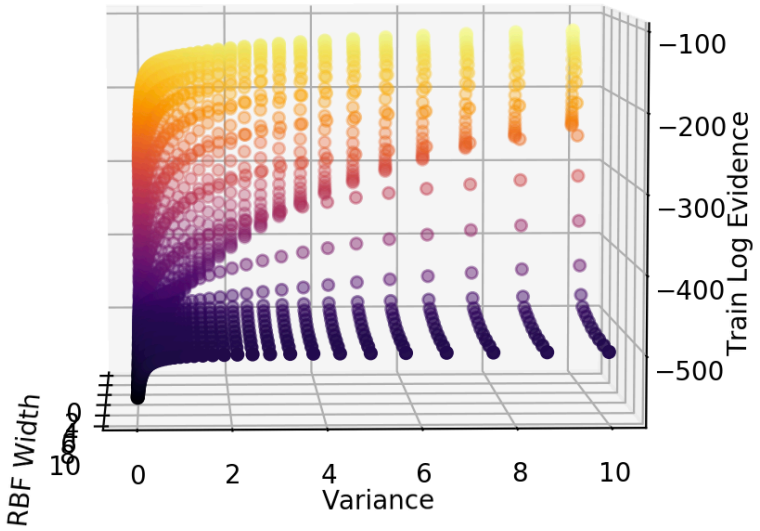

Figure 5: ML optimisation for the Variance on the training data

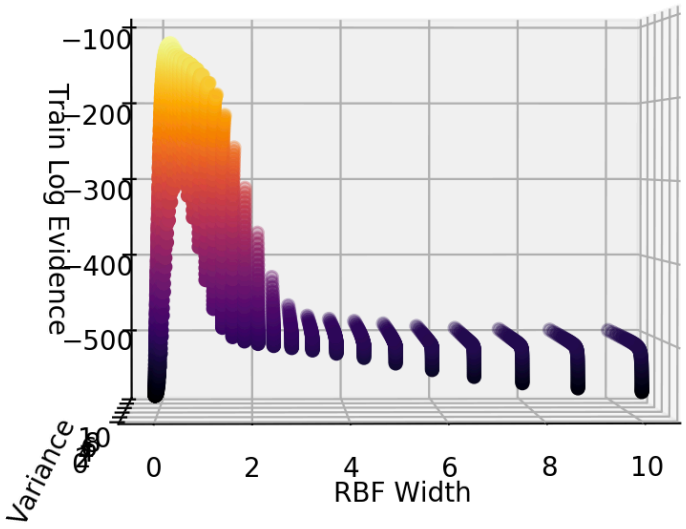

Figure 6: ML optimisation for the RBF width on the training data

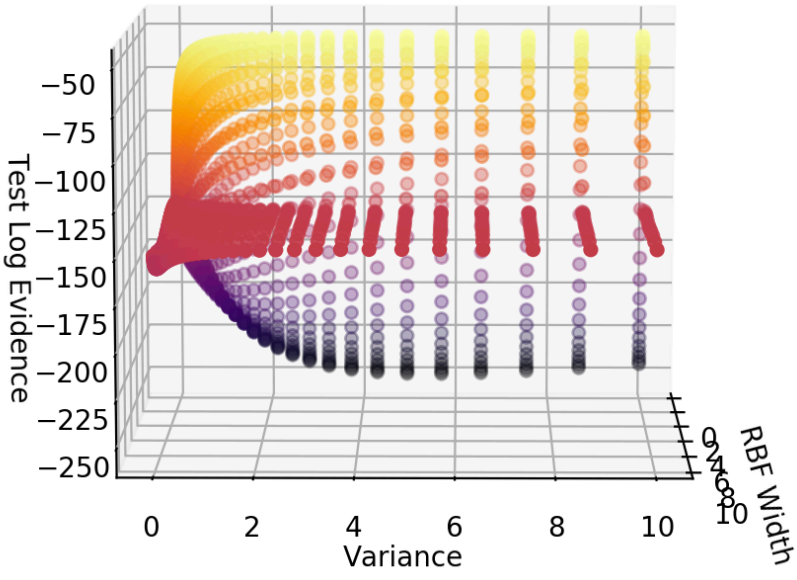

Figure 7: ML optimisation for the Variance on the test data

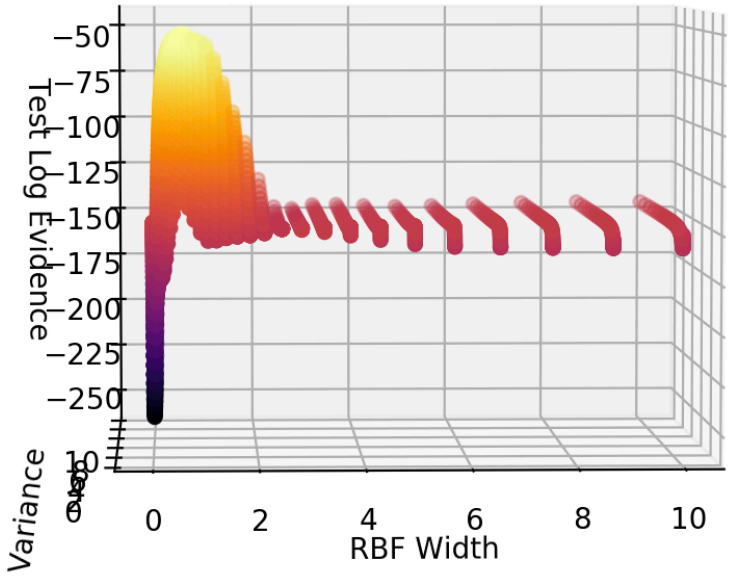

Figure 8: ML optimisation for the RBF width on the test data

By looking at Figure 5 the optimum width of the RBF for the train data was found to be 0.16768 and from Figure 6 the optimum variance for the training dataset is found to be 10.0 as expected. However if it is decided to look at the test data instead of the training data (Figure 7 and 8) the optimum RBF width is found to be 0.5179 while the variance is found to be 8.6851. Therefore it is clearly seen that by using the test data different results are obtained.

Now as the new parameters are chosen the new results containing the contour plots, confusion matrices and average log-likelihood are seen below.

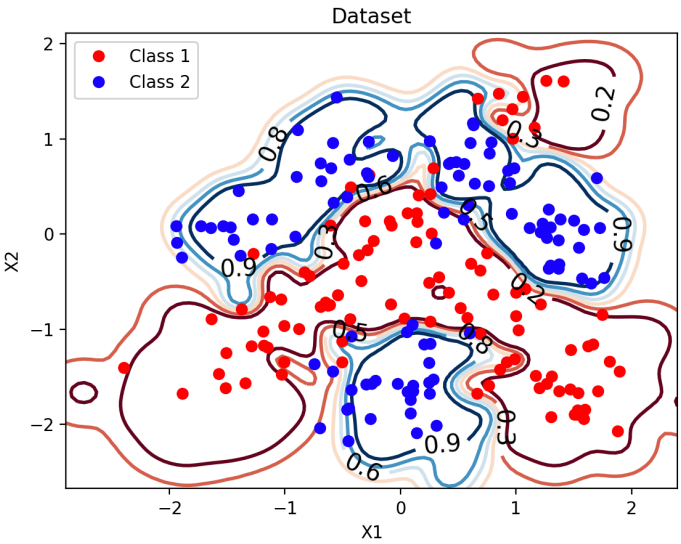

Figure 9: MAP and Bayesian contours

Table 6: Confusion Matrix: MAP and Bayesian for test data

		Predicted label	
		0	1
True Label	0	88.12%	11.88%
	1	8.08%	91.92%

Table 7: Confusion Matrix: MAP and Bayesian for training data

		Predicted label	
		0	1
True Label	0	94.57%	5.43%
	1	4.81%	95.19%

Table 8:Average Log-Likelihood comparison

	Training Data	Test Data
MAP	−0.12124	−0.25832
BAYESIAN	−0.21391	−0.29539

Comparing table 8 and table 5 it can be observed a large improvement for the MAP log-likelihood and modest improvement in the Bayesian log-likelihood. This was expected because now the hyper-parameters are designed to fit the training dataset. Comparing the confusion matrices for the test data (Table 2 and Table 6) a small improvement can be observed in the test classification precision. The same thing can be observed by looking at the confusion matrices for the training data (Table 7 and Table 4).

In order to check that the algorithm does not overfit, an early termination of the algorithm can be done before the minimum is reached. This method is better done using a gradient descent rather than using a 50x50 grid search.

IV. Conclusions

The sum up of the observations made in this full technical report and some final conclusions:

- Bayesian Classifier performs worse than the MAP classifier because it takes into account all the weights, therefore lower weights will reduce the log-likelihood
- In our case it was not possible to distinguish between the results of the contour plots(section II) and the confusion matrices(section II) for the test data for both MAP and Bayesian
- The advantage of the Bayesian is that because it takes into account all the weights it is less prone to overfitting therefore it results in much better generalised model.
- The downsize of using the Laplace approximation is that for the low datasets it will fail to approximate the posterior distribution. Therefore large datasets are required for the Bayesian central limit theorem to be applied.
- It was found that tuning the hyper-parameters by maximising the evidence is essentially optimising for the ML solution. It was observed the by this approach it lead to overfitting the parameters to the training dataset therefore the test data improved by a very small amount. A better method is explained at the end of section III.

Appendix:

```
def learning(X,y,var,func,grad):
    x=np.concatenate((np.ones((X.shape[0],1)),X),1)
    W_init=np.transpose(np.random.randn(x.shape[1])*0.01)
    w_map, val, d =optimize.fmin_l_bfgs_b(func,W_init,grad,(x,y,var))
    return w_map
```