Basics of Statistical Learning

Dylan Dijk

1 Decision Making an Introduction

1.1 Introduction

The general set-up for supervised learning is that we have random variables $\mathbf{X}$ and Y and we want to be able to estimate a function f that predicts y from $\mathbf{x}$. Where y and $\mathbf{x}$ are realisations from these random variables, in this report we will have $\mathbf{x} \in \mathbb{R}^d$.

Within supervised learning we have two different types of problems: **regression** and **classification**. Regression problems are those where the outcome variable Y can take any value in the real numbers, classification problems are those when the outcome variable takes values in a finite discrete set.

We choose our prediction function f by finding the function that minimises $\mathbb{E}[L(f(\mathbf{X}), Y)]$, where L is a loss function. This expectation is over the joint distribution of X and Y, and is often referred to as the risk of a prediction function f. The choice of loss function can effect our choice of f, for example when using the squared error loss the risk is minimised when we select $f(\mathbf{x}) = \mathbb{E}[Y \mid \mathbf{X}]$(A.1.1). Alternatively, if we use the absolute loss function then the optimal prediction function is $f(\mathbf{x}) = \text{median}(p(Y \mid \mathbf{X}))$

In practice, we do not know the joint distribution that is used to calculate the expectation, but we have a dataset $D := \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ which are realisations from the unknown joint distribution. Instead of calculating the risk we can use an empirical estimate of the risk given by $\frac{1}{n}\sum L(y_i, f(\mathbf{x}_i))$.

1.2 Least Squares (LS) Regression

LS regression is the estimate of the prediction function that minimises the empirical risk with respect to the squared error loss. We minimise the sum of squared errors over observations in $D_0 \subseteq D$ which we refer to as the training set.

The **LS problem** is therefore defined as:

$$\underset{f}{\text{argmin}} \sum_{i \in D_0} (y_i - f(\mathbf{x}_i))^2 \tag{1}$$

The **linear LS** problem is a special case of the LS problem, in linear LS the prediction function is a linear function of both the predictors and the weights. This problem can be formalised as:

$$\mathbf{W}_{LS} : = \underset{\mathbf{w}}{\text{argmin}} \sum_{i \in D_0} (y_i - f(\mathbf{x}_i, \mathbf{w}))^2 \tag{2}$$

$$f(\mathbf{x}; \mathbf{w}) := \mathbf{w}_1^T \mathbf{x} + w_0 \tag{3}$$

In this case we have a closed form solution for $\mathbf{W}_{LS}$(A.2.1).

The choice of the squared error loss function can be motivated by taking a probabilistic approach, by making an assumption on the conditional distribution $Y \mid \mathbf{X}$. More precisely, if we assume $(\mathbf{x}_i, y_i)$ for $i = 1, \ldots, n$ are i.i.d samples with $y_i|\mathbf{x}_i; \sigma \sim N(f(\mathbf{x}_i; \mathbf{w}), \sigma^2)$, then the MLE of $\mathbf{w}$ is equivalent to solving (2) (A.2.2).

The models given by (3) are restricted to only fitting straight lines. However, we can extend this class of models to look at linear combinations of non-linear transformations of the input variables. We do this by transforming the input $\mathbf{x}$ using a feature transform $\phi : \mathbb{R}^d \to \mathbb{R}^b$, and using $f(\phi(\mathbf{x}); \mathbf{w}) := \mathbf{w}_1^T \phi(\mathbf{x}) + w_0$. These models are still called linear models as they are linear in terms of $\mathbf{w}$, and therefore still has the closed form solution given by (19) with $\mathbf{X}$ replaced by $\phi(\mathbf{X})$, but now $f(\phi(\mathbf{x}); \mathbf{w})$ can be a non-linear function of $\mathbf{x}$.

If $\phi(\mathbf{X})$ is symmetric and invertible then the least squares closed form solution reduces to $[\phi(\mathbf{X})]^{-1} y^T$ (A.2.3).

1.3 Over-fitting

To measure the performance of a prediction function at estimating the outcome variable, we need to use data that the model has not been trained on. To carry this out, we can split our dataset into a partition $D = D_0 \bigcup D_1$. D_0 is the set that we train f over and is referred to as the training set.

We can calculate the empirical risk using either D_0 or D_1. The empirical risk of f calculated over the same set it is trained on is called the training error, and the empirical risk over D_1 for f trained on D_0 is called the testing error. The difference between these two estimates is an estimate for the generalization error of f.

If we look at look at a polynomial feature transform $\phi(\mathbf{x}) := [h(x^{(1)}), \ldots, h(x^{(d)})]$ with $h : \mathbb{R} \to \mathbb{R}^b$ $h(t) := [t^1, \ldots t^b]$, then $\phi(\mathbf{x}) \in \mathbb{R}^{db}$ which implies that $\mathbf{w}_1 \in \mathbb{R}^{db}$. If we increase b this causes the training error to reduce, because we are testing the prediction function on the same data we trained it on. However, the training error will start to increase as we increase b.

This problem of the generalization error becoming large as the model becomes too flexible is known as overfitting.

1.4 Regularization

One strategy to avoid over-fitting is to add a penalty term to the objective function which penalises the vector of coefficients $\mathbf{w}$ becoming too large. One choice of measure for the size of $\mathbf{w}$ is the squared euclidean norm, and can be formulated in the following way:

$$\mathbf{w}_{\text{LS-R}} := \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 \ + \ \lambda \mathbf{w}^T \mathbf{w} \tag{4}$$

The solution for this problem is given by $\mathbf{w}_{\text{LS-R}} = (\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I}_{p+1})^{-1} \mathbf{X}\mathbf{Y}^T$ (A.2.4)

We can tune for λ by selecting a λ that corresponds to a low testing error of the model. This can be done using a single validation set D_1, or we can partition the dataset into more blocks. The method of training a dataset on a part of a dataset then calculating the testing error using the rest of the dataset is called cross-validation.

1.5 Bayesian Approach

So far we have taken the Frequentist approach and have been finding point estimates for the parameters in our models. In the Bayesian approach we now assume some of the parameters are random variables, and we assign them a distribution, we call the distributions on the parameters the priors. We can then use Bayes' rule (6) with the priors and the data generating model (likelihood) to update the probability distribution of the parameters after observing the dataset.

Bayes theorem for two events from a probability space is given by:

$$\mathbb{P}(A \mid B) = \frac{\mathbb{P}(B \mid A)\mathbb{P}(A)}{\mathbb{P}(B)} \tag{5}$$

We also have Bayes theorem for the density functions for random variables X and Y

$$p_{X|Y=y}(x) = \frac{p_{Y|X=x}(y)p_X(x)}{p_Y(y)} \tag{6}$$

1.5.1 Bayesian Linear Regression

In Bayesian linear regression we have the model $y_i = f(\mathbf{x}_i; \mathbf{w}) + \epsilon$, and we assume ϵ is normally distributed with mean zero and variance σ^2. Moving towards a Bayesian approach, we now assume $\mathbf{w}$ is a realisation from a random variable $\mathbf{W}$, we call the probability distribution of this random variable the prior on $\mathbf{W}$. Here we assume the prior to be normal with mean zero and variance $\sigma^2_{\mathbf{W}}$.

If we assume the input data $\underline{\mathbf{x}} := \{\mathbf{x}_i, i = 1, \ldots, n\}$ are fixed, then using the Bayes rule we have that:

$$p(\mathbf{w} \mid \underline{y}; \underline{\mathbf{x}}) \propto p(\underline{y} \mid \mathbf{w}; \underline{\mathbf{x}})p(\mathbf{w}) \tag{7}$$

We have also used here the notation $\underline{y} := \{y_i, i = 1, \ldots, n\}$

An alternative assumption to get equation (7) would be for $\mathbf{X}$ to be independent of $\mathbf{w}$.

Once we have the posterior $p(\mathbf{w}|D)$ we can take our estimate of $\mathbf{w}$ to be the one that maximises the posterior distribution (MAP estimate), this is still a point estimate of $\mathbf{w}$. Now if we take the likelihood function and the prior to be normal, the MAP estimate of $\mathbf{w}$ is the same estimate as that from ridge regression (A.3.1).

We have included a prior distribution for $\mathbf{w}$, but once we have the posterior we have just calculated a point estimate. We can instead calculate the predictive distribution, which is given by:

$$p(\hat{y} \mid \mathbf{x}, D) = \int p(\hat{y} \mid \mathbf{x}, \mathbf{w}) p(\mathbf{w} \mid D) d\mathbf{w} \tag{8}$$

The solution for this integral can be found in the Appendix (A.3.2).

1.6 Binary Classification

Binary classification problems are a special case of classification problems, where the output variable has just two classes, we will denote these as C_1 and C_2. Our prediction function f will now just assign the input variables to one of these two classes, and therefore the rule will divide the input space into two regions R_1 and R_2.

The classification problem can be split into an **inference** stage and then a **decision** stage. There are different approaches to solving classification problems, in terms of how we carry out the inference and then utilise this to come up with a decision rule.

Different Inference Strategies:

- **Generative approach**: Estimate the class-conditional probabilities $p(\mathbf{x} \mid C_k)$ for each class, and the prior class probabilities $p(C_k)$. Then use Bayes rule to find the posterior class probabilities $p(C_k \mid \mathbf{x})$. Once you have the posterior probabilities, you can then use decision theory to determine a prediction function f.
- **Discriminative approach**: Directly estimate the posterior class probabilities.

Decision Theory

Once we have performed inference we can then use this to decide on a decision rule. Lets first look at using the zero-one loss function, defined as $L(f(\mathbf{X}), Y) := \mathbb{1}_{\{f(\mathbf{X}) \neq Y\}}$.

If we have a decision rule f then the risk with this loss function (probability of misclassification) is minimised if we have the decision rule $f(\mathbf{x}) = \underset{k \in \{-\mathbf{1}, +\mathbf{1}\}}{\operatorname{argmax}} \mathbb{P}(Y = k \mid \mathbf{X} = \mathbf{x})$ (A.4.1). Calculating the risk requires us to know the joint distribution, in this case to get the optimal prediction function we need the posterior class probabilities.

Minimising the empirical risk with respect to this loss function is equivalent to minimising the number of misclassifications. But, we may want to do more than just minimize the number of misclassifications. For example, if the consequences of a making a false negative outweighs that of a false positive, we might want to introduce a higher penalty to making such a prediction. Therefore we can use a loss function that assigns a higher cost to such misclassifications.

A Appendix

A.1 Minimising Risk

A.1.1 Optimal prediction function under squared loss

We want to show that: $\underset{f}{\operatorname{argmin}}\, \mathbb{E}[(f(X) - Y)^2] = \mathbb{E}[Y \mid X]$

$$\begin{aligned}
\mathbb{E}[(f(X) - Y)^2] &= \mathbb{E}[(f(X) - f^*(X) + f^*(X) - Y)^2] && (9)\\
&= \mathbb{E}[(f^*(X) - Y)^2] + \mathbb{E}[(f(X) - f^*(X))^2] + 2\mathbb{E}[(Y - f^*(X))(f^*(X) - f(X))] && (10)\\
&\geq \mathbb{E}[(f^*(X) - Y)^2] + 2\mathbb{E}[(Y - f^*(X))(f^*(X) - f(X))] && (11)\\
&= \mathbb{E}[(f^*(X) - Y)^2] + 2\mathbb{E}_X\mathbb{E}_{Y|X}[(Y - f^*(X))(f^*(X) - f(X))] && (12)\\
&= \mathbb{E}[(f^*(X) - Y)^2] + 2\mathbb{E}_X[(\mathbb{E}[Y|X] - f^*(X))(f^*(X) - f(X))] && (13)
\end{aligned}$$

Now if we let $f^*(X) = \mathbb{E}[Y|X]$ then we have that:

$$\mathbb{E}[(f(X) - Y)^2] \geq \mathbb{E}[(f^*(X) - Y)^2] \tag{14}$$

Therefore this risk of any other function is larger.

A.2 Least Squares Regression

A.2.1 Closed form solution to least squares regression

The solution of $\mathbf{W}_{LS} := \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i \in D_0} (y_i - f(\mathbf{x}_i, \mathbf{w}))^2$ is given by $(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{y}^T$.
Where $\mathbf{X} \in \mathbb{R}^{(p+1)\times n}$ is the design matrix with columns given by the input vectors $\mathbf{x}_i$, with an additional row of 1 which allows us to avoid including w_0 in the calculations.

$$\begin{aligned}
W_{LS} &= \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i \in D_0} (y_i - \mathbf{x}_i^T\mathbf{w})^2 && (15)\\
&= \underset{\mathbf{w}}{\operatorname{argmin}} (\mathbf{Y}^T - \mathbf{X}^T\mathbf{w})^T(\mathbf{Y}^T - \mathbf{X}^T\mathbf{w}) && (16)\\
&= \underset{\mathbf{w}}{\operatorname{argmin}} - 2\mathbf{w}^T\mathbf{X}\mathbf{Y}^T + \mathbf{w}^T\mathbf{X}\mathbf{X}^T\mathbf{w} && (17)
\end{aligned}$$

From the second to third line we just removed terms that did not depend on $\mathbf{w}$.
Now taking the derivative with respect to w, and setting to zero to find the stationary point we get:

$$0 = -2\mathbf{X}\mathbf{Y}^T + 2\mathbf{X}\mathbf{X}^T\mathbf{w}_{\text{LS}} \tag{18}$$

$$\mathbf{w}_{\text{LS}} = (\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{Y}^T \tag{19}$$

If $(\mathbf{X}\mathbf{X}^T)$ is not invertible then $\mathbf{w}_{\text{LS}}$ is not defined.

This is the case, for example, when $n < (p+1)$ because then $\exists\ \mathbf{v} \in \mathbb{R}^{p+1}$ such that we can write the following. The reason being is that $\mathbf{X}^T$ is a matrix with $p+1$ columns of size n. The dimension of the column-space is therefore n, and by the fact that if you have more vectors than the size of the dimension $(p+1 > n)$ of the vector space then they are linearly dependent (Corollary 6.24 of MA106 notes). That is there exists a linear combination of vectors that equals zero.

$$\mathbf{X}^T\mathbf{v} = \mathbf{0} \tag{20}$$

$$\implies \mathbf{X}\mathbf{X}^T\mathbf{v} = \mathbf{0} \tag{21}$$

Therefore $\ker(\mathbf{X}\mathbf{X}^T) > 0$ and hence by the rank-nullity theorem the matrix $\mathbf{X}\mathbf{X}^T$ is not full rank and so is not invertible.

If we add a $\lambda\mathbf{I}_{p+1}$ term to $\mathbf{X}\mathbf{X}^T$ it then becomes invertible.

A.2.2 MLE with Gaussian assumption on conditional distribution

The MLE for $\mathbf{w}$, given with the probabilistic approach, is equivalent to the solution found using the method of least squares.

In (2) we just try and find the vector that minimises the squared difference between the prediction and the outcome. But we can take a probabilistic approach, and model the uncertainty of the outcome variable using a probability distribution, then try and find $\mathbf{w}$ that maximises the likelihood of this model being correct.

Assume $\{(\mathbf{x}_i, y_i) \mid i = 1, \dots n\}$ are repeated independent samples from random variables $\mathbf{X}$ and Y respectively. With $Y \mid \mathbf{X} \sim N(f(\mathbf{x}; \mathbf{w}), \sigma^2)$.

We then can write that:

$$\mathbb{P}(y_1, \dots y_n \mid \mathbf{x}_1, \dots \mathbf{x}_n; \mathbf{w}, \sigma) = \frac{\prod_{i=1}^n \mathbb{P}(\mathbf{x}_i, y_i; \mathbf{w}, \sigma)}{\prod_{i=1}^n \mathbb{P}(\mathbf{x}_i; \mathbf{w}, \sigma)} \tag{22}$$

$$= \prod_{i=1}^n \mathbb{P}(y_i \mid \mathbf{x}_i; \mathbf{w}, \sigma) \tag{23}$$

The MLE of $\mathbf{w}$ is then the value of $\mathbf{w}$ that maximise the likelihood of our dataset under this model therefore:

$$\mathbf{w}_{\text{MLE}} := \underset{\mathbf{w}}{\operatorname{argmax}} \log\Big(\prod_{i=1}^{n} \mathbb{P}(y_i \mid \mathbf{x}_i; \mathbf{w}, \sigma)\Big) \tag{24}$$

$$= \underset{\mathbf{w}}{\operatorname{argmax}} \sum_{i=1}^{n} \log\Big(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{(y_i - f(\mathbf{x}_i;\mathbf{w}))^2}{2\sigma^2}\Big)\Big) \tag{25}$$

$$= \underset{\mathbf{w}}{\operatorname{argmax}} \sum_{i=1}^{n} \Big(-\frac{(y_i - f(\mathbf{x}_i;\mathbf{w}))^2}{2\sigma^2}\Big) \tag{26}$$

$$= \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i;\mathbf{w}))^2 \tag{27}$$

Therefore (24) is equivalent to solving (1) when we assume f takes a linear form.

If the noise in the data is not normally distributed then it will not make sense to assume a normal distribution. For example if the true data generating model as noise coming from a skewed distribution.

A.2.3 Symmetric feature transform

If $\phi(\mathbf{X})$ is symmetric and invertible then $\mathbf{w}_{\text{LS}} = [\phi(\mathbf{X})]^{-1}y^T$

$$\mathbf{w}_{\text{LS}} = [\phi(\mathbf{X})\phi(\mathbf{X})^T]^{-1}\phi(\mathbf{X})y^T \tag{28}$$

$$= \phi(\mathbf{X})^{-1}\phi(\mathbf{X})^{-1}\phi(\mathbf{X})y^T \tag{29}$$

$$= \phi(\mathbf{X})^{-1}y^T \tag{30}$$

A.2.4 Ridge regression solution

$$\mathbf{w}_{\text{LS-R}} := \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i;\mathbf{w}))^2 \; + \; \lambda \mathbf{w}^T\mathbf{w} \tag{31}$$

$$= \underset{\mathbf{w}}{\operatorname{argmin}} - 2\mathbf{w}^T\mathbf{X}\mathbf{Y}^T + \mathbf{w}^T\mathbf{X}\mathbf{X}^T\mathbf{w} \; + \; \lambda\mathbf{w}^T\mathbf{w} \tag{32}$$

Now taking the derivative with respect to w, and setting to zero to find the stationary point we get:

$$0 = -2\mathbf{X}\mathbf{Y}^T + 2\mathbf{X}\mathbf{X}^T\mathbf{w}_{\text{LS-R}} + 2\lambda\mathbf{w}_{\text{LS-R}} \tag{33}$$

$$\mathbf{X}\mathbf{Y}^T = (\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I}_{p+1})\mathbf{w}_{\text{LS-R}} \tag{34}$$

$$\mathbf{w}_{\text{LS-R}} = (\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I}_{p+1})^{-1}\mathbf{X}\mathbf{Y}^T \tag{35}$$

A.3 Bayesian Approach

A.3.1 MAP estimate in Bayesian linear regression with normal prior and likelihood

If we have that $p(y_i \mid \mathbf{w}; \mathbf{x}_i) = N(f(\mathbf{x}_i; \mathbf{w}), \sigma^2)$ and $p(\mathbf{w}) = N(0, \mathbf{I}\sigma_{\mathbf{w}}^2)$. Then the MAP estimate for $\mathbf{w}$ is equal to $\mathbf{w}_{\text{LS-R}}$.

$$\begin{aligned}
\mathbf{w}_{\text{MAP}} :&= \underset{\mathbf{w}}{\text{argmax}}\, p(\mathbf{w} \mid D) && (36)\\
&= \underset{\mathbf{w}}{\text{argmax}}\, p(\{y_1, \ldots, y_n\} \mid \mathbf{w}; \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) p(\mathbf{w}) && (37)\\
&= \underset{\mathbf{w}}{\text{argmax}} \prod_{i=1}^{n} N(f(\mathbf{x}_i; \mathbf{w}), \sigma^2)\, N(0, \sigma_{\mathbf{w}}^2) && (38)\\
&= \underset{\mathbf{w}}{\text{argmax}} \sum_{i=1}^{n} \Big(- \frac{(y_i - f(\mathbf{x}_i; \mathbf{w}))^2}{2\sigma^2} \Big) - \frac{\mathbf{w}^T \mathbf{w}}{2\sigma_{\mathbf{w}}^2} && (39)\\
&= \underset{\mathbf{w}}{\text{argmin}} \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 + \frac{\sigma^2}{\sigma_{\mathbf{w}}^2} \mathbf{w}^T \mathbf{w} && (40)
\end{aligned}$$

In line (37) we have used that $p(\mathbf{w} \mid \{y_1, \ldots, y_n\}; \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) \propto p(\{y_1, \ldots, y_n\} \mid \mathbf{w}; \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) p(\mathbf{w})$ which we get from Bayes rule.

A.3.2 Predictive distribution in Bayesian Linear Regression

Assuming $p(\mathbf{w}) = N(0, \mathbf{I}\sigma_{\mathbf{w}}^2)$ and $p(\underline{\mathbf{y}} \mid \mathbf{w}; \underline{\mathbf{x}}) = N(\phi(\mathbf{X})^T \mathbf{w} \mathbf{I} \sigma^2)$, where $\mathbf{X}$ is the matrix with columns given by the vectors in $\underline{\mathbf{x}}$. Then with new observation $\mathbf{x}^*$, we want to calculate the predictive distribution $p(\hat{y} \mid \mathbf{x}^*, D)$.

In (A.3.1) we just looked at $p(\mathbf{w} \mid \underline{\mathbf{y}}; \underline{\mathbf{x}})$ up to a proportionality constant, but we can calculate the full probability density. In the Pattern Recognition and Machine Learning book by Bishop, equation (2.116) on p.93 gives the full form for the posterior distribution $p(\mathbf{w} \mid \underline{\mathbf{y}}; \underline{\mathbf{x}})$.

Using this equation and substituting the values we get:

$$p(\mathbf{w} \mid \underline{\mathbf{y}}; \underline{\mathbf{x}}) = N\Big(\Big(\frac{1}{\sigma^2}\phi(\mathbf{X})\phi(\mathbf{X})^T + \frac{1}{\sigma_{\mathbf{w}}^2}\mathbf{I}\Big)^{-1} \phi(\mathbf{X})\underline{\mathbf{y}}\frac{1}{\sigma^2},\ \Big(\frac{1}{\sigma^2}\phi(\mathbf{X})\phi(\mathbf{X})^T + \frac{1}{\sigma_{\mathbf{w}}^2}\mathbf{I}\Big)^{-1}\Big) \tag{41}$$

Now to get $p(\hat{y} \mid \underline{\mathbf{y}}; \mathbf{x}^*, \underline{\mathbf{x}})$ we need to calculate the following integral:

$$\int p(\hat{y} \mid \mathbf{w}; \mathbf{x}^*) p(\mathbf{w} \mid \underline{\mathbf{y}}; \underline{\mathbf{x}})\, d\mathbf{w} \tag{42}$$

Equation (41) gives the density for $p(\mathbf{w} \mid \underline{\mathbf{y}}; \underline{\mathbf{x}})$. And we assume that $p(\hat{y} \mid \mathbf{w}; \mathbf{x}^*) = N(\phi(\mathbf{x}^*)^T \mathbf{w}, \sigma^2)$, recall that in Bayesian linear regression we are assuming that $y = f(\mathbf{x}; \mathbf{w}) + \epsilon$.

Now to calculate the integral (42) we refer again to the book by Bishop. In particular, equation (2.115) gives the **marginal distribution** when we have Gaussian marginal and conditional distributions.

After substituting the values we get:

$$
\begin{aligned}
p(\hat{y} \mid \underline{\mathbf{y}}; \mathbf{x}^*, \underline{\mathbf{x}}) &= N\Big(\phi(\mathbf{x}^*)^T \Big(\frac{1}{\sigma^2}\phi(\mathbf{X})\phi(\mathbf{X})^T + \frac{1}{\sigma^2_{\mathbf{w}}}\mathbf{I}\Big)^{-1} \phi(\mathbf{X})\underline{\mathbf{y}}\frac{1}{\sigma^2}, \\
&\quad \sigma^2 + \phi(\mathbf{x}^*)^T \Big(\frac{1}{\sigma^2}\phi(\mathbf{X})\phi(\mathbf{X})^T + \frac{1}{\sigma^2_{\mathbf{w}}}\mathbf{I}\Big)^{-1} \phi(\mathbf{x}^*)\Big) \qquad (43) \\
&= N\Big(f(\mathbf{x}^*; \mathbf{w}_{\text{LS-R}}), \sigma^2 + \phi(\mathbf{x}^*)^T \sigma^2 \Big(\phi(\mathbf{X})\phi(\mathbf{X})^T + \frac{\sigma^2}{\sigma^2_{\mathbf{w}}}\mathbf{I}\Big)^{-1} \phi(\mathbf{x}^*)\Big) \qquad (44)
\end{aligned}
$$

In line (44) $f(\mathbf{x}^*; \mathbf{w}_{\text{LS-R}}) := \phi(\mathbf{x}^*)^T \mathbf{w}_{\text{LS-R}}$, where $\mathbf{w}_{\text{LS-R}}$ is the same as (35) but with $\mathbf{X}$ replaced by $\phi(\mathbf{X})$.

A.4 Binary Classification

A.4.1 Bayes Classifier

The binary decision rule $f^*(\mathbf{x}) = \underset{k \in \{-\mathbf{1}, +\mathbf{1}\}}{\text{argmax}} \ \mathbb{P}(Y = k \mid \mathbf{X} = \mathbf{x})$ minimises the risk with respect to the zero-one loss function.

$$
\mathbb{E}_{X,Y}[L(f(X), Y)] = \mathbb{E}_X \mathbb{E}_{Y|X}[\mathbb{1}_{\{f(\mathbf{X}) \neq Y\}}] \qquad (45)
$$

Now if we minimise $\mathbb{E}_{Y|X}[\mathbb{1}_{\{f(\mathbf{X}) \neq Y\}}]$ for each $X = \mathbf{x}$, then this will minimise the risk.

$$
\begin{aligned}
\mathbb{E}_{Y|X}[\mathbb{1}_{\{f(\mathbf{X}) \neq Y\}}] &= \sum_{k=1}^{2} \mathbb{1}_{\{f(\mathbf{x}) \neq k\}} \mathbb{P}(Y = k \mid \mathbf{X} = \mathbf{x}) \qquad (46) \\
&= \sum_{k=1}^{2} (1 - \mathbb{1}_{\{f(\mathbf{x}) = k\}}) \mathbb{P}(Y = k \mid \mathbf{X} = \mathbf{x}) \qquad (47) \\
&= 1 - \sum_{k=1}^{2} \mathbb{1}_{\{f(\mathbf{x}) = k\}} \mathbb{P}(Y = k \mid \mathbf{X} = \mathbf{x}) \qquad (48)
\end{aligned}
$$

Therefore to minimize this expression, we want $f(\mathbf{x})$ to take the value k that has the largest posterior class probability.

$$
\underset{f(\mathbf{x})}{\text{argmin}} \ \mathbb{E}_{Y|X}[\mathbb{1}_{\{f(\mathbf{X}) \neq Y\}}] \ = \underset{k \in \{-\mathbf{1}, +\mathbf{1}\}}{\text{argmax}} \ \mathbb{P}(Y = k \mid \mathbf{X} = \mathbf{x}) \qquad (49)
$$

In the lecture slides we talked about a choosing a decision boundary given by the level set $g(\mathbf{x}) = 0$, where if $g(\mathbf{x}) \geq 0$ then we predict the class $+1$ and otherwise -1.

If we write:

$$f(\mathbf{x}) = \begin{cases} +1, & g(\mathbf{x}) \geq 0 \\ -1, & g(\mathbf{x}) \leq 0 \end{cases}$$

With $g(\mathbf{x}) := \mathbb{P}(Y = 1 \mid \mathbf{X} = \mathbf{x}) - \mathbb{P}(Y = -1 \mid \mathbf{X} = \mathbf{x})$, then this formulation is equivalent to (49).