Elements Of Statistical Learning Solutions

Solutions to select exercises

Krishan Bhalla

July 28, 2020

CONTENTS

1

INTRODUCTION

A selection of solutions to Elements of Statistical Learning by Hastie, Tibshirani, and Friedman. This will be continually updated as I work through the book.

I will not answer all exercises, but will endeavour to solve many.

This chapter largely exists to align latex chapter labels with those of the book.

2

CHAPTER 2 - OVERVIEW OF SUPERVISED LEARNING

2.1 Suppose each of K-classes has an associated target t_k which is a vector of all zeros except one in the kth position. Show that classifying to the largest element of $\hat{y}$ amounts to choosing the closest target $min_k \|t_k - \hat{y}\|$ if the elements of y sum to one.

Our norm here is the L^2 norm.

$$\begin{aligned} argmin_k \|t_k - \hat{y}\| &= argmin_k \|t_k - \hat{y}\|^2 \\ &= argmin_k \sum_i (\hat{y}_i - (t_k)_i)^2 \\ &= argmin_k \sum_i (\hat{y}_i - \delta_{i,k})^2 \\ &= argmin_k \left(1 - 2\hat{y}_k + \sum_i \hat{y}_i^2\right) \\ &= argmin_k \left(-2\hat{y}_k\right) \end{aligned}$$

Where the last line follows as it is the only term dependant on k. *argmin* is independent of scale, so the above states that the k corresponding to the minimum value of the norm is exactly the largest element of $\hat{y}$

2.2 Show how to compute the Bayes decision boundary for the example in Figure 2.5.

Setup: There are 10 means m_k from a bivariate guassian distribution $N((1,0)^T, I)$ and we label this class blue. We take 10 n_k from $N((0,1)^T, I)$ and label this class orange. For each class there were 100 observations, where each observation was generated by picking one of the m_k (n_k resp) each with equal probability, and taking a point randomly from a $N(m_k, I/5)$ distribution.

Let x be an observation. We equate posteriors for x being blue or yellow, and note that in our setup $\mathbb{P}(blue) = \mathbb{P}(orange)$ (priors are equal) to simplify to:

$$\sum_k exp(-5\|m_k - x\|^2) = \sum_k exp(-5\|n_k - x\|^2)$$

This defines a curve in the plane separating the two classes.

2.3 Derive equation 2.24.

Setup: Consider N data points uniformly distributed in a p-dimensional unit ball around the origin. Consider a nearest neighbour estimate at the origin. The median distance from the origin to the closest point is given by:

$$d(p_N) = \left(1 - \frac{1}{2}^{1/N}\right)^{1/p}$$

Let x be a point and let $y = \|x\|$. y has cdf equal to the ratio of a ball of radius y to a ball of radius 1, i.e. $F(y) = y^p$. The minimum over all x then has cdf $F_{ymin}(y) = 1 - (1 - F(y))^N$ (a general fact for order statistics). Thus $F_{ymin} = 1 - (1 - y^p)^N$.

The median distance for $ymin$ is when $F_{ymin}(y) = 1/2$. Solving for this yields the result.

2.4 Setup as in book. Projection in direction a.

Pick an orthonormal basis of $\mathbb{R}^p$ which includes the vector a, say $a_1, \ldots, a_p$ with $a_1 = a$. Then each $x_i = \sum_j X_{i,j} a_j$, and so $z_i = X_{i,1}$ where X is the matrix with rows x_i

The x_i have distribution $N(0, I_p)$, and under such a distribution each component of x_i has distribution $N(0, 1)$.

In particular this means that each $X_{i,j}$ has distribution $N(0, 1)$ and so the z_i do.

The squared distance from the origin is just z_i^2, with distribution χ_1^2, and this has mean 1.

2.6 Consider a regression problem with inputs x_i and outputs y_i, and a parameters model $f_\theta(x)$ to be fit by least squares. Show that if there are observations with tied or identical values of x then the fir can be obtained from a reduced weighted least squares problem.

The problem can of finding θ amounts to solving the following:

$$argmin_\theta \left(y - f_\theta(x)\right)^T \left(y - f_\theta(x)\right)$$

Denote by $z_1, \ldots, z_M$ the unique values of x in our training set, denote by n_j the number of occurrences of value z_j. Then let $t_j = \frac{1}{n_j} \sum_{i=1}^{n_j} y_i$ If we can get to the following, we're done (as it is in the form of weighted regression).

$$argmin_\theta \sum_j n_j \left(t_j - f_\theta(z_j)\right)^2$$

Expanding the initial expression we get (denoting by $y_{i,j}$ the ith value of y corresponding to input z_j):

$$\begin{aligned}
(y - f_\theta(x))^T (y - f_\theta(x)) &= \sum_i (y_i - f_\theta(x_i))^2 \\
&= \sum_i \left(y_i^2 + f_\theta(x_i)^2 - 2y_i f_\theta(x_i)\right) \\
&= \sum_i \left(y_i^2 + f_\theta(x_i)^2 - 2y_i f_\theta(x_i)\right) \\
&= \sum_{j=1}^{M} \sum_{i=1}^{n_j} y_{i,j}^2 - 2f_\theta(z_j) y_{i,j} + f_\theta(z_j)^2 \\
&= \sum_{j=1}^{M} \left(\sum_i y_{i,j}^2\right) - 2n_j f_\theta(z_j) t_j + n_j f_\theta(z_j)^2 \\
&= \sum_{j=1}^{M} n_j (t_j - f_\theta(z_j))^2 - \sum_{j=1}^{M} n_j t_j^2 + \sum_{j=1}^{M} \left(\sum_i y_{i,j}^2\right)
\end{aligned}$$

This last trick of adding 0 leaves us with an expression where only the first sum is dependant on θ Thus when taking $argmin_\theta$ we can ignore the last two terms.

This leaves us with the required equivalence.:

$$argmin_\theta (y - f_\theta(x))^T (y - f_\theta(x)) = argmin_\theta \sum_j n_j \left(t_j - f_\theta(z_j)\right)^2$$

3

CHAPTER 3 - LINEAR METHODS FOR REGRESSION

3.1 Show that the F-statistic for dropping a single coefficient of a model is equivalent to the square of the corresponding z score

Let X be our data, and let $v_{j,j}$ be the jth diagonal element of $V = (X^T X)^{-1}$. $z_j = \hat{\beta}_j / \hat{\sigma}\sqrt{v_{j,j}}$ is the z score.

The F statistic is

$$F = \frac{(RSS_0 - RSS_1)/(p1 - p0)}{RSS_1/(N - p_1 - 1)}$$

Where the regression models are have $p_1 + 1$ and $p_0 + 1$ degrees of freedom respectively. We also know that $\hat{\sigma}^2$ is equivalent to the denominator. In the case of dropping a single variable, this simplifies to:

$$F = \frac{RSS_0 - RSS_1}{\hat{\sigma}^2}$$

Where $\hat{\sigma}$ is derived from the bigger model.

Thus our question can be simplifed to showing that:

$$RSS_0 - RSS_1 = \hat{\beta}_n^2 / v_{j,j}$$

We know $\hat{\beta}$ $N(\beta, \sigma^2 V)$ and so $\hat{\beta}_j$ $N(\beta_j, \sigma^2 v_{j,j})$.

Under the null-hypothesis $\beta_j = 0$ and so: $\hat{\beta}_j = \sigma\sqrt{v_{j,j}}Z$ where $Z \sim N(0,1)$ Thus $\hat{\beta}_j^2 = \sigma^2 v_{j,j} Z^2 = \sigma^2 v_{j,j} Q$ where $Q \sim \chi_1^2$

Similarly RSS_0, RSS_1 have distribution $\sigma^2 \chi^2_{N_i}$ where N_i is the number of degrees of freedom. Thus $RSS_0 - RSS_1 \sim \sigma^2 \chi_1^2$

Hence $\hat{\beta}_j / v_{j,j}$ and $RSS_0 - RSS_1$ have the same distribution. They further test the same hypothesis and thus must be identical.

3.3 a. Prove the Gauss-Markov theorem: the least squares estimate of a parameter $a^T\beta$ has a variance no bigger than that of any other linear unbiased estimate of $a^T\beta$.
b. Secondly, show that if $\hat{V}$ is the variance-covariance matrix of the least squares estimate of β and $\tilde{V}$ is the variance covariance matrix of any other linear unbiased estimate, then $\hat{V} \leq \tilde{V}$, where $B \leq A$ if $A - B$ is positive semidefinite.

First note that part b implies part a. If β has dimension 1, then V is just the variance of beta, and $\leq$ is equivalent to the normal $\leq$ operator. Taking the inner product with a is just a linear operation. We thus only need to show b.

Suppose $\hat{\beta}$ is the OLS estimate of β and that $\tilde{\beta}$ is another linear unbiased estimate. The the variance-covariance matrices be $\hat{V}$ and $\tilde{V}$ resp. $\hat{\beta} = \left(X^TX\right)^{-1}X^Ty$ so $\hat{\beta} = Cy$ say, and write $\tilde{\beta} = (C+D)y$ for some non-zero mxn matrix D.

$$\begin{aligned}
\mathbb{E}\left(\tilde{\beta}\right) &= \mathbb{E}\left((C+D)\left(X\beta+\epsilon\right)\right) \\
&= \mathbb{E}\left(\left(\left(X^TX\right)^{-1}X^T + D\right)\left(X\beta+\epsilon\right)\right) \\
&= \mathbb{E}\left(\left(\beta + DX\beta\right)\right) + \zeta\mathbb{E}(\epsilon) \\
&= \mathbb{E}\left(\left(\beta + DX\beta\right)\right) \\
&= \beta + DX\beta \\
&= \beta
\end{aligned}$$

Where the last lines follow as $\tilde{\beta}$ is unbiased. In particular $DX\beta = 0$ and so $DX = 0$ as beta is an unobserved parameter to be estimated.

$$\begin{aligned}
\tilde{V} &= Var\left(\tilde{\beta}\right) \\
&= Var\left((C+D)\,y\right) \\
&= (C+D)\left(C+D\right)^T Var\left(y\right) \\
&= \sigma^2\left(C+D\right)\left(C+D\right)^T \\
&= \sigma^2\left(CC^T + CD^T + DC^T + DD^T\right) \\
&= \sigma^2\left(CC^T + CD^T + DC^T + DD^T\right) \\
&= \hat{V} + \sigma^2\left(+CD^T + DC^T + DD^T\right) \\
&= \hat{V} + \sigma^2\left(\left(X^TX\right)^{-1}X^TD^T + DX\left(X^TX\right)^{-1} + D^TD\right) \\
&= \hat{V} + \sigma^2 DD^T
\end{aligned}$$

Where we know that DD^T is positive semi-definite[1] and so are done.

1 $v^TDD^Tv = \left\|D^Tv\right\| \geq 0\ \forall v$

3.4 Show how the vector of least squares coefficients can be obtained from a single pass of the Gram–Schmidt procedure (Algorithm 3.1). Represent your solution in terms of the QR decomposition of X.

Let $X = QR$ be the QR decomposition of X. This can be attained via Gram-Schmidt. We assume that R has no zeros on the diagonal (i.e. the variables are linearly independent). Then

$$\begin{aligned}\hat{\beta} &= (X^T X)^{-1} X^T y \\ &= (R^T R)^{-1} R^T Q^T y \\ &= R^{-1} Q^T y\end{aligned}$$

We can invert R via backpropogation, and $Q^T y$ is easily calculable.

3.5 Show that the ridge regression problem (using α to denote the constant intercept vector)

$$\hat{\beta} = argmin_{\beta,\alpha} \left(y - \alpha - X\beta\right)^T \left(y - \alpha - X\beta\right) + \lambda \left\|\beta\right\|^2$$

is equivalent to the problem:

$$\hat{\beta^c} = argmin_{\beta^c,\alpha^c} \left(y - \alpha^c - \tilde{X}\beta^c\right)^T \left(y - \alpha^c - \tilde{X}\beta^c\right) + \lambda \left\|\beta^c\right\|^2$$

Where $\tilde{X} = X - \bar{X}$, and bar represents the the $Nxp)$ matrix where each value in the jth column is the mean x_j. Give the correspondence between β^c and the original β. Do the same for the lasso.[2]

This problem is easier with summation

$$\begin{aligned}&\left(y - \alpha - X\beta\right)^T \left(y - \alpha - X\beta\right) + \lambda \left\|\beta\right\|^2 \\ &= \left(y - \alpha - \bar{X}\beta - (X - \bar{X})\beta\right)^T \left(y - \alpha - \bar{X}\beta - (X - \bar{X})\beta\right) + \lambda \left\|\beta\right\|^2 \\ &= \left(y - (\alpha + \bar{X}\beta) - (X - \bar{X})\beta\right)^T \left(y - (\alpha + \bar{X}\beta) - (X - \bar{X})\beta\right) + \lambda \left\|\beta\right\|^2 \\ &= \left(y - \alpha^c - \tilde{X}\beta^c\right)^T \left(y - \alpha^c - \tilde{X}\beta^c\right) + \lambda \left\|\beta^c\right\|^2\end{aligned}$$

Where $\alpha_i^c = \alpha_i + \sum_{j=1}^{p} \bar{x}_j \beta_j$ in all coordinates, and $\beta^c = \beta$. This is an expression of our desired form. This problem is equivalent to demeaning the data and adjusting the intercept. One can do exactly the same for the lasso.

2 This is all a tad odd as regression with intercept is desired, but we only penalise the non-intercept terms.

3.6 Show that the ridge regression estimate is the mean (and mode) of the posterior distribution, under a Gaussian prior $\beta\ N(0, \tau I)$ and Gaussian sampling model $y\ N(X\beta, \sigma^2 I)$. Find the relationship between the ridge parameter λ and the variances τ and σ^2

This question really states that the pdf of the posterior is proportional to the pdfs of y given β and β. Hence:

$$f(\beta|y) \propto f(\beta)f(y|\beta) \tag{1}$$
$$log(f(\beta|y)) = C + log(f(\beta)) + log(f(y|\beta)) \tag{2}$$
$$= C + -\frac{1}{2\tau}\beta^T\beta + log\left(\frac{1}{\sqrt{2 * \pi * \tau}}\right) \tag{3}$$
$$- \frac{1}{2\sigma^2}\left(y - X\beta\right)^T\left(y - X\beta\right) + log\left(\frac{1}{\sqrt{2 * \pi * \sigma^2}}\right) \tag{4}$$
$$= C + -\frac{1}{2\tau}\beta^T\beta + -\frac{1}{2\sigma^2}\left(y - X\beta\right)^T\left(y - X\beta\right) \tag{5}$$

Where we absorb terms into the constant as needed. We have recovered that:

$$f(\beta|y) = C_1 e^{-\frac{1}{2\sigma^2}\left(\frac{\sigma^2}{\tau}\beta^T\beta + (y - X\beta)^T(y - X\beta)\right)} \tag{6}$$
$$= C_1 e^{-\frac{1}{2\sigma^2}\left(\beta^T\left(X^TX + \frac{\sigma^2}{\tau}I\right)\beta + y^Ty - y^TX\beta - \beta^TX^Ty\right)} \tag{7}$$
$$= C_1 e^{-\frac{1}{2\sigma^2}\left(\beta^T\Sigma\beta + y^Ty - y^TX\beta - \beta^TX^Ty\right)} \tag{8}$$
$$= C_1 e^{-\frac{1}{2\sigma^2}\left(\left(\Sigma\beta - X^Ty\right)^T\Sigma^{-1}\left(\Sigma\beta - X^Ty\right) + y^Ty - y^TX^TXy\right)} \tag{9}$$
$$= C_2 e^{-\frac{1}{2\sigma^2}\left(\Sigma\beta - X^Ty\right)^T\Sigma^{-1}\left(\Sigma\beta - X^Ty\right)} \tag{10}$$
$$\tag{11}$$

Where $\Sigma = \left(X^TX + \frac{\sigma^2}{\tau}I\right)$ is a $(p+1) * (p+1)$ matrix, and for (11) we absorbed into the constant any terms not dependant on β

Thus the posterior has a multivariate gaussian pdf (up to scaling), so the mean and mode of the distribution are identical, and can be found by maximising (5) over β, which due to the minus sign is sufficient.

This amounts to

$$argmin_\beta\left(\frac{1}{\tau}\beta^T\beta + \frac{1}{\sigma^2}\left(y - X\beta\right)^T\left(y - X\beta\right)\right)$$

And hence is equivalent to solving:

$$argmin_\beta\left(\frac{\sigma^2}{\tau}\beta^T\beta + \left(y - X\beta\right)^T\left(y - X\beta\right)\right)$$

Hence the ridge regression parameter is σ^2/τ

3.9 Forward stepwise regression: Suppose we have the QR decomposition for the N x q matrix X_1 in a multiple regression problem with response y, and suppose we have an additional $p - q$ predictors in the matrix X_2. Denote the residual by r. Describe an efficient procedure for establishing which additional variable will reduce the residual sum of squares the most.

Intuition - pick the column $\hat{v}$ such that v has the least angle with r. i.e.

$$\hat{v} = argmax_{v \in \{columns X_2\}} \frac{|r^T v|}{\|v\|}$$

With this in mind, let $u_j = x_j - \frac{|r^T x_j|}{\|x_j\|}$ where the x_j are the columns of X_2, and let $v_j = \frac{u_j}{\|u_j\|}$

$$\begin{aligned} RSS &= r^T r \\ r &= y - \hat{y} \\ &= y - R^{-1} Q^T y \\ let\ r_j &= r - \left(r^T u_j\right) u_j \\ and\ RSS_j &= RSS - 2\left(r^T u_j\right)^2 + \left(r^T u_j\right)^2 \\ then\ RSS_j &= RSS - \left(r^T u_j\right)^2 \end{aligned}$$

Where RSS_j is the residual sum of squares for our new model. This verifies our intuition, RSS_j is minimised when we pick the column of X_2 with least angle to r. I assume the efficiency in the question comes from the ease of inverting R compared to $X^T X$ (backpropogation will do), and that we can extend our QR decomposition to include the new variable easily via Gram-Schmidt. In particular most of what is needed for Gram-Schmidt is already computed when taking inner product with the residual.

3.10 Backward stepwise regression. Suppose we have the multiple regression fit of y on X along with the standard errors and Z-scores. We wish to establish which variable, when dropped, will increase the RSS the least. How would you do this?

From question 3.1 we know that the F-statistic for dropping a single coefficient of a model is equivalent to the square of the corresponding

Z score. Further, we know the F statistic in the case of dropping 1 variable is:

$$\frac{(RSS_0 - RSS_1)}{RSS_1/(N - p1 - 1)}$$

where N, p_1 and RSS_1 are constant. In particular, the change is RSS is proportional to the F-score with a constant that does not depend on choice of variable to be dropped, and so the change in RSS is proportional to the square of the z-score in the same manner. Thus the difference will be smallest (smallest increase in RSS) if the variable has minimal z-score in our model.

3.12 Show the ridge regression estimates can be obtained by OLS regression on an augmented data set. Add p rows to the centered matrix X, $\sqrt{\lambda}I_p$, and augment y with p zeros.

Let $X_2 = [X^T, \sqrt{\lambda}I_p]^T$ be our augmented matrix, and let $y_2 = [y^T, 0]^T$ be the augmented response. Under OLS, we have $\beta_2 = (X_2^T X_2)^{-1} X_2^T Y$.

$$\begin{aligned} X_2^T X_2 &= [X^T, \sqrt{\lambda}I_p][X^T, \sqrt{\lambda}I_p]^T \\ &= X^T X + \lambda I_p \\ X_2^T y_2 &= [X^T, \sqrt{\lambda}I_p][y^T, 0]^T \\ &= X^T y \end{aligned}$$

Thus $\beta = \left(X^T X + \lambda I_p\right) X^T y$ which is exactly the ridge regression beta for our non-augmented dataset.

3.13 Derive expression 3.62 and show that $\hat{\beta}^{pcr}(p) = \hat{\beta}^{ls}$

Given an SVD of X, $X = UDV^T$ say, with $V = [v_1, \ldots, v_p]$, the principal components z_m in equation 3.61 are defined as Xv_m for all m. The principal component regression is, for any $M <= p$:

$$\begin{aligned} \hat{y}^{pcr} &= \bar{y} + \sum_{m=1}^{M} \hat{\theta_m} z_m \\ &= \bar{y} + X \sum_{m=1}^{M} \hat{\theta_m} v_m \end{aligned}$$

So we can just set $hat\beta^{pcr}(M) = \sum_{m=1}^{M} \hat{\theta_m} v_m$ and we're done. Now it remains to show that $\hat{\beta}^{pcr}(p) = \hat{\beta}^{ls}$.

$$\begin{aligned}
\hat{\beta}^{pcr}(p) &= \sum_{m=1}^{p} \hat{\theta_m} v_m \\
&= V[\frac{z_1^T y}{z_1^T z_1}, \ldots, \frac{z_p^T y}{z_p^T z_p}]^T \\
&= V[\frac{z_1^T y}{d_1^2}, \ldots, \frac{z_p^T y}{d_p^2}^T \\
&= VD^{-2}[z_1^T y, \ldots, z_p^T y]^T \\
&= VD^{-2}[u_1^T d_1^T y, \ldots, u_p^T d_p^T y]^T \\
&= VD^{-2}D[u_1^T y, \ldots, u_p^T y]^T \\
&= VD^{-1}U^T y
\end{aligned}$$

Where we made use of the SVD. Now:

$$\begin{aligned}
\hat{\beta}^{ls} &= \left(X^T X\right)^{-1} X^T y \\
&= \left(VDU^T UDV^T\right)^{-1} UDV^T y \\
&= \left(VDDV^T\right)^{-1} VDU^T y \\
&= D^{-2}\left(VV^T\right)^{-1} VDU^T y \\
&= D^{-2}VDU^T y \\
&= VD^{-1}U^T y
\end{aligned}$$

Using the orthonomality of U and V.

3.14 Show that in the orthogonal case, partial least squares stops after m = 1 steps.

Assume X is such that each column has mean zero, unit variance, and the colums are orthogonal. let $z = \sum_i \left(x_i^T y\right) x_i$ and $\hat{\theta} = \frac{z^T y}{z^T z}$.

In this case

$$\begin{aligned}
z^T z &= \sum_i \sum_j \left(x_i^T y\right)\left(x_j^T y\right) x_i^T x_j \\
&= \sum_i \left(x_i^T y\right)^2 \left(x_i^T x_i\right) \\
&= \sum_i \left(x_i^T y\right)^2
\end{aligned}$$

Similarly $z^T y = \sum_i \left(x_i^T y\right)^2$. Now let $x_j^{(1)} = x_j - \frac{z^T x_j}{z^T z}$ Then let:

$$x_j^{(1)} = x_j - \frac{z^T x_j}{z^T z} z$$

In the next iteration, we have:

$$\begin{aligned}\langle x_j^{(1)}, y\rangle &= x_j^T y - \frac{\left(x_j^T y\right)\left(x_j^T x_j\right)}{\sum_i \left(x_i^T y\right)^2} z^T y \\ &= x_j^T y - \frac{x_j^T y}{\sum_i \left(x_i^T y\right)^2} \sum_i \left(x_i^T y\right)^2 \\ &= x_j^T y - x_j^T y \\ &= 0\end{aligned}$$

So algorithm 3.3 (page 81 in my edition) terminates after 1 step.

3.19 Show that $\left\|\hat{\beta}^{ridge}\right\|$ increases as the tuning parameter $\lambda \to 0$. Does the same property hold for the Lasso and PLS?

Throughout this question I use regression without intercept as the intercept is not included in the authors formulation of the penalty in the Lagrangian form.

a) Ridge

For data X with mean 0 and unit variance, we have

$$\hat{\beta}^{ridge} = (X^T X + \lambda I)^{-1} X^T y$$

Then:

$$\begin{aligned}\left\|\hat{\beta}^{ridge}\right\|^2 &= \left((X^T X + \lambda I)^{-1} X^T y\right)^T (X^T X + \lambda I)^{-1} X^T y \\ &= \left(V(D^T D + \lambda I)^{-1} D U^T y\right)^T V(D^T D + \lambda I)^{-1} D U^T y \\ &= y^T U D^2 (D^T D + \lambda I)^{-2} U^T y\end{aligned}$$

Using the SVD, the commutativity of D and V, and the orthonormality of V. $D^T D + \lambda I$ is diagonal, and so $D^2(D^T D + \lambda I)^{-2}$ is too with entries $\frac{d_j^2}{d_j^2+\lambda}$ on the diagonal. Let $z_j = (U^T y)_j$ Then we see:

$$\begin{aligned}\left\|\hat{\beta}^{ridge}\right\|^2 &= \sum_i \sum_j \frac{d_j^2 z_i^T z_j}{d_j^2 + \lambda} \\ &= \sum_j \frac{d_j^2 z_j^2}{d_j^2 + \lambda}\end{aligned}$$

All terms are non-negative, and increase with decreasing λ so $\hat{\beta}^{ridge}$ must too. Recall that we can view the ridge regression estimate as the solution to

$$\hat{\beta}^{ridge} = argmin_\beta (y - X\beta)^T (y - X\beta) + \lambda \|\beta\|_2^2$$

b) Lasso

Under a similar formulation, we have

$$\hat{\beta}^{lasso} = argmin_{\beta}(y - X\beta)^T(y - X\beta) + \lambda \|\beta\|_1$$

In a similar way one can see that if λ is sufficiently large, it dominates this expression (for fixed X and y), and so $\hat{\beta}^{lasso}$ will decrease in norm with increasing λ.

c) PLS

?

3.23. Please refer to the book. LAR

a)Correlation with residuals remains constant in absolute value.

$$\begin{aligned}
\frac{1}{N}X^T(y - u(\alpha)) &= \frac{1}{N}\left(X^T y - \alpha X^T X\hat{\beta}\right) \\
&= \frac{1}{N}\left(X^T y - \alpha X^T X\left(X^T X\right)^{-1} X^T y)\right) \\
&= \frac{1}{N}\left(X^T y - \alpha X^T y)\right) \\
&= \frac{1-\alpha}{N}X^T y \\
\text{Thus } |\frac{1}{N}X^T(y - u(\alpha))| &= (1-\alpha)[\lambda, \ldots, \lambda]^T
\end{aligned}$$

Which is exactly the required result in vector notation.

b)Explicit form of correlation

The question as stated ignores the need for an absolute value, so we shall assume that $\langle x_j, y - u(\alpha)\rangle \geq 0$ for every j, else replace x_j by $-x_j$ in our data.

Correlations are given by covariance divided by the product of the standard deviations. as everything (data and response) is assumed to be standardised, we have:

$$\begin{aligned}
\left(y - u(\alpha))^T(y - u(\alpha)\right) &= y^T y - 2y^T u(\alpha) + u(\alpha)^T u(\alpha) \\
&= y^T y - 2\alpha y^T X\hat{\beta} + \alpha^2\hat{\beta}^T X^T X\hat{\beta} \\
&= y^T y - 2\alpha y^T X\hat{\beta} + \alpha^2\hat{\beta}^T X^T X\left(X^T X\right)^{-1} X^T y \\
&= y^T y - 2\alpha y^T X\hat{\beta} + \alpha^2\hat{\beta}^T X^T y \\
&= y^T y + \alpha(\alpha - 2)y^T X\hat{\beta}
\end{aligned}$$

Setting $\alpha = 1$ we get $RSS = y^T y - y^T X\hat{\beta}$. Thus:

$$\Big(y - u(\alpha))^T (y - u(\alpha)\Big) = y^T y + \alpha(\alpha - 2)(y^T y - RSS)$$

And so:

$$\begin{aligned}\Big(y - u(\alpha))^T (y - u(\alpha)\Big) &= (1-\alpha)^2 y^T y + \alpha(2-\alpha)RSS \\ &= N(1-\alpha)^2 + \alpha(2-\alpha)RSS\end{aligned}$$

$$\begin{aligned}Corr\left(x_j, y - u(\alpha)\right) &= \frac{\langle x_j, y - u(\alpha)\rangle / N}{\sqrt{\langle x_j, x_j\rangle / N}\sqrt{\langle y - u(\alpha), y - u(\alpha)\rangle / N}} \\ &= \frac{\lambda(1-\alpha)}{1 \cdot \sqrt{(1-\alpha)^2 + \alpha(2-\alpha)RSS/N}} \\ &= \frac{(1-\alpha)}{\sqrt{(1-\alpha)^2 + \alpha(2-\alpha)RSS/N}}\lambda\end{aligned}$$

As required.

c) Show the LAR algorithm keeps correlations tied and monotonically decreasing.

Part a) showed that the LAR algorithm keeps correlations tied ad this is exactly the step taken for an active set of variables in the *kth* step. Part b) shows that correlations are monotonically decreasing with α as numerator falls faster than the numerator for α in $[0, 1]$[3]. In particular $\lambda(0) = \lambda$ and $\lambda(1) = 0$.

3.24 LAR directions

Let A_k be the active set of variables at the beginning of the kth step, and β_{A_k} be the coefficient vector for these variables at this step. There are $k-1$ non-zero values of β and one zero - the latest variable to enter the active set. The current residual is denoted by $r_k = y - X_{A_k}\beta_{A_k}$ where X_{A_k} is the subset of the data X consisting of only the variables in A_k. The direction for this step is

$$\delta_k = \left(X_{A_k}^T X_{A_k}\right)^{-1} X_{A_k}^T r_k$$

Let $u_k = X_{A_k}\delta_k$ be the LAR direction vector. Recall that the cosine of the angle between vectors u and v is $\frac{u^T v}{\|v\|\|u\|}$. The columns X_{A_k} have unit norm. Thus the angle between u_k and x_j is $\frac{x_j^T u_k}{\|u_k\|}$, and the denominator is constant given u_k. That is to say, the angle between u_k and the predictors in A_k is constant if and only if $x_j^T u_k$ is.

3 One can and should compute the gradient and check that it is negative on paper.

$$\begin{aligned} X_{A_k}^T u_k &= X_{A_k}^T X_{A_k} \delta_k \\ &= X_{A_k}^T X_{A_k} \left(X_{A_k}^T X_{A_k} \right)^{-1} X_{A_k} r_k \\ &= X_{A_k}^T r_k \end{aligned}$$

The last term the correlation of each predictor with the residual, and in the LAR algorithm (and from 3.23) we know that all predictors in A_k have the same correlation with the residual. Thus all the $x_j^T u_k$ are equal as required.

3.25 LAR Lookahead

We use the notation from 3.24. $\beta_{A_k}(\alpha) = \beta_{A_k} + \alpha\delta_k$

$$r_k(\alpha) = (I - \alpha\delta_k)\, r_k$$

For any variable x_i say, we have the correlation to $r_k(\alpha)$ defined as:

$$\begin{aligned} c_i(\alpha) &= x_i^T r_k(\alpha) \\ &= \left(x_i^T - \alpha x_i^T \delta_k \right) r_k \end{aligned}$$

For i in the active set and some j not in the active set, x_j can only be added if these $c_i(\alpha)$ and $c_j(\alpha)$ are equal in absolute value. There are finitely many j not in the active set, and for every i in the active set, the correlations are the same so we can fix i. There are two cases (depending on signs):

$$\alpha_j = \frac{(x_i - x_j)^T r_k}{(x_i - x_j)^T \delta_k r_k}$$
$$\alpha_j = \frac{(x_i + x_j)^T r_k}{(x_i + x_j)^T \delta_k r_k}$$

We can compute all such α_j, and then solve:

$$argmax_{j \notin A_k} \left| x_j^T r_k(\alpha_j) \right|$$

The result will be the next predictor to be included, with the corresponding α_j the value of α at which this predictor will be added.

3.29 Please see the book. Ridge regression with duplicate variables

Setup: The data X has identical columns x.

Let $X = [x, \ldots, x]$ say, for x some column vector, where X has dimension $n \times m$. The ridge regression beta is given by

$$\left(X^T X + \lambda I\right)^{-1} X^T y$$

Note $X^T y = [x^T y, \ldots, x^T y]^T$ $X^T X$ is a constant matrix with every value $x^T x$.

Let $M = X^T X/(x^T x)$ be the matrix of ones.

Looking for an inverse of the form $sM + \lambda^{-1} I$ yields

$$s = \frac{-x^T x}{\lambda(\lambda + m x^T x)}$$

Where one uses that $M^T M = mM$. We have

$$(sM + \lambda^{-1} I)[x^T y, \ldots, x^T y]^T = [c, \ldots, c]^T$$

Where

$$\begin{aligned} c &= \frac{-m \cdot x^T x \cdot x^T y}{\lambda(\lambda + m \cdot x^T x)} + \frac{x^T y}{\lambda} \\ &= \frac{(\lambda + m \cdot x^T x) x^T y - m \cdot x^T x \cdot x^T y}{\lambda(\lambda + m \cdot x^T x)} \\ &= \frac{\lambda x^T y}{\lambda(\lambda + m \cdot x^T x)} \\ &= \frac{x^T y}{\lambda + m \cdot x^T x} \end{aligned}$$

3.30 Consider the elastic net optimisation problem:

$$min_\beta \|y - X\beta\|_2^2 + \lambda \left[\alpha \|\beta\|_2^2 + (1 - \alpha) \|\beta\|_1\right]$$

Show how one can turn this into a lasso problem using an augmented version of X and y

Let $X_2 = [X, \sqrt{\alpha\lambda} I_p]^T$ where X is $n \times p$ and we assume X has been standardised while y has mean 0 (this is an easy augmentation to remove the constant in regression). Let $y_2 = [y, 0]$ where we have augmented y with p additional zeros. This is the setup in 3.12 except here we use $\alpha\lambda$ in place of λ

$$\begin{aligned} \|y_2 - X_2\beta\|_2^2 &= \left\| \left[y - X\beta, \sqrt{\alpha\lambda}\beta\right]^T \right\|_2^2 \\ &= \|y - X\beta\|_2^2 + \left\|\sqrt{\alpha\lambda}\beta\right\|_2^2 \\ &= \|y - X\beta\|_2^2 + \alpha\lambda \|\beta\|_2^2 \end{aligned}$$

With these data, the original problem becomes a lasso optimisation problem with parameter $\lambda(1 - \alpha)$:

$$min_\beta \|y_2 - X_2\beta\|_2^2 + \lambda(1 - \alpha) \|\beta\|_1$$

4

CHAPTER 4 - LINEAR METHODS FOR CLASSIFICATION

4.1 Show how to solve the generalised eigenvalue problem

$$max_a a^T B a$$
$$\textbf{subject to } a^T W a = 1$$

by transforming to a standard eigenvalue problem.

Via Lagrange multipliers we have $D(a^T B a) = \lambda D(a^T W a - 1)$ for some λ. This gives:

$$\begin{aligned} & D(a^T B a) = \lambda D(a^T W a - 1) \\ & \Rightarrow 2Ba = 2\lambda W a \\ & \Rightarrow Ba = \lambda W a \\ & \Rightarrow W^{-1} B a = \lambda a \end{aligned}$$

So a is an eigenvector of $W^{-1}B$ with eigenvalue λ. We would select the a corresponding to the largest eigenvalue[1], as we have $a^T B a = \lambda a^T W a = \lambda$.

4.2 Suppose we have features $x \in \mathbb{R}^p$, a two class response, with class sizes N_1, N_2, and the target coded as $-N/N_1, N/N_2$.

4.2a) Show that the LDA rule classifies to class 2 if

$$x^T \Sigma^{-1}(\hat{\mu}_2 - \hat{\mu}_1) > \frac{1}{2}(\hat{\mu}_2 + \hat{\mu}_1)\Sigma^{-1}(\hat{\mu}_2 - \hat{\mu}_1) - log(N_2/N_1)$$

and class 1 otherwise

We have

$$log\frac{\mathbb{P}(G=2|X=x)}{\mathbb{P}(G=1|X=x)} = \log\left(\frac{\pi_2}{\pi_1}\right) - \frac{1}{2}(\hat{\mu}_2 + \hat{\mu}_1)\Sigma^{-1}(\hat{\mu}_2 - \hat{\mu}_1) + x^T\Sigma^{-1}(\hat{\mu}_2 - \hat{\mu}_1)$$

1 We can always scale a such that our constraint is satisfied.

and we classify to class 2 if this value is at at least 0. Expanding gives:

$$\begin{aligned}
&\frac{\mathbb{P}(G=2|X=x)}{\mathbb{P}(G=1|X=x)} > 0 \\
&\iff \log\left(\frac{\pi_2}{\pi_1}\right) - \frac{1}{2}(\hat{\mu}_2+\hat{\mu}_1)\Sigma^{-1}(\hat{\mu}_2-\hat{\mu}_1) + x^T\Sigma^{-1}(\hat{\mu}_2-\hat{\mu}_1) > 0 \\
&\iff \log\left(\frac{N_2}{N_1}\right) - \frac{1}{2}(\hat{\mu}_2+\hat{\mu}_1)\Sigma^{-1}(\hat{\mu}_2-\hat{\mu}_1) + x^T\Sigma^{-1}(\hat{\mu}_2-\hat{\mu}_1) > 0 \\
&\iff x^T\Sigma^{-1}(\hat{\mu}_2-\hat{\mu}_1) > \frac{1}{2}(\hat{\mu}_2+\hat{\mu}_1)\Sigma^{-1}(\hat{\mu}_2-\hat{\mu}_1) - \log\left(\frac{N_2}{N_1}\right)
\end{aligned}$$

b) Consider minimisation of the least squares criterion $\|y - \beta_0 \cdot \mathbb{1} - X\beta\|_2^2$. *Show that the solution* $\hat{\beta}$ *satisfies*

$$\left[(N-2)\hat{\Sigma} + N\hat{\Sigma}_B\right]\beta = N(\hat{\mu}_2 - \hat{\mu}_1)$$

where $\hat{\Sigma}_B = \frac{N_1 N_2}{N^2}(\hat{\mu}_2-\hat{\mu}_1)(\hat{\mu}_2-\hat{\mu}_1)^T$

Write $\mathbb{1}_i$ for the indicator vector corresponding to class i, so that $\mathbb{1} = \mathbb{1}_1 + \mathbb{1}_2$. We will differentiate $(y - \beta_0 \cdot \mathbb{1} - X\beta)^T (y - \beta_0 \cdot \mathbb{1} - X\beta)$ w.r.t β and β_0 and solve. Setting the derivatives to zero gives:

$$\begin{aligned}
X^T(y - \hat{\beta}_0 \cdot \mathbb{1} - X\hat{\beta}) = 0 \\
\mathbb{1}^T(y - \hat{\beta}_0 \cdot \mathbb{1} - X\hat{\beta}) = 0
\end{aligned}$$

Let $\tilde{\mu} = (N_1\hat{\mu}_1 + N_2\hat{\mu}_2) = X^T\mathbb{1}$. Simplifying the above, we get.

$$X^T y - \frac{1}{N}X^T\mathbb{1}\mathbb{1}^T(y - X\hat{\beta}) = X^T X\hat{\beta} \tag{12}$$

$$\Rightarrow X^T y - \frac{1}{N}X^T\mathbb{1}\mathbb{1}^T y = X^T X\hat{\beta} - \frac{1}{N}X^T\mathbb{1}\mathbb{1}^T X\hat{\beta} \tag{13}$$

$$\Rightarrow X^T y - \frac{1}{N}\tilde{\mu}\mathbb{1}^T y = X^T X\hat{\beta} - \frac{1}{N}\tilde{\mu}\tilde{\mu}^T\hat{\beta} \tag{14}$$

$$\Rightarrow X^T y - \frac{1}{N}\tilde{\mu}(N_1 c_1 + N_2 c_2) = \left(X^T X - \frac{1}{N}\tilde{\mu}\tilde{\mu}^T\right)\hat{\beta} \tag{15}$$

Expanding $\tilde{\mu}\tilde{\mu}^T$ gives:

$$\begin{aligned}
\tilde{\mu}\tilde{\mu}^T &= (N_1\hat{\mu}_1 + N_2\hat{\mu}_2)(N_1\hat{\mu}_1 + N_2\hat{\mu}_2)^T \\
&= N_1^2\hat{\mu}_1\hat{\mu}_1^T + N_2^2\hat{\mu}_2\hat{\mu}_2^T + N_1N_2(\hat{\mu}_1\hat{\mu}_2^T + \hat{\mu}_2\hat{\mu}_1^T)
\end{aligned}$$

Now we want to establish the $\hat{\Sigma}$ terms

$$\begin{aligned}
\hat{\Sigma} &= \frac{1}{N-2}\left(\sum_{i=1}^{N_1}(x_i-\hat{\mu}_1)(x_i-\hat{\mu}_1)^T + \sum_{i=N_1 1}^{N}(x_i-\hat{\mu}_2)(x_i-\hat{\mu}_2)^T+\right)\\
&= \frac{1}{N-2}\left(\sum_{i=1}^{N_1}(x_i-\hat{\mu}_1 x_i^T - x_i\hat{\mu}_1^T + \hat{\mu}_1\hat{\mu}_1^T)\right)\\
&+ \frac{1}{N-2}\left(\sum_{i=N_1+1}^{N}(x_i-\hat{\mu}_2 x_i^T - x_i\hat{\mu}_2^T + \hat{\mu}_2\hat{\mu}_2^T)\right)\\
&= \frac{1}{N-2}\left(\sum_{i=1}^{N}(x_i x_i^T) - 2N_1\hat{\mu}_1\hat{\mu}_1^T + N_1\hat{\mu}_1\hat{\mu}_1^T - 2N_2\hat{\mu}_2\hat{\mu}_2^T + N_2\hat{\mu}_2\hat{\mu}_2^T\right)\\
&= \frac{1}{N-2}\left(X^TX - N_1\hat{\mu}_1\hat{\mu}_1^T - N_2\hat{\mu}_2\hat{\mu}_2^T\right)
\end{aligned}$$

This yields:

$$X^TX = (N-2)\hat{\Sigma} + N_1\hat{\mu}_1\hat{\mu}_1^T + N_2\hat{\mu}_2\hat{\mu}_2^T$$

Thus:

$$\begin{aligned}
(N-2)\hat{\Sigma} + N\hat{\Sigma}_B &= X^TX - N_1\hat{\mu}_1\hat{\mu}_1^T - N_2\hat{\mu}_2\hat{\mu}_2^T + \frac{N_1N_2}{N}(\hat{\mu}_2-\hat{\mu}_1)(\hat{\mu}_2-\hat{\mu}_1)^T\\
&= X^TX + \left(\frac{N_1N_2}{N} - N_1\right)\hat{\mu}_1\hat{\mu}_1^T + \left(\frac{N_1N_2}{N} - N_2\right)\hat{\mu}_2\hat{\mu}_2^T\\
&- \frac{N_1N_2}{N}(\hat{\mu}_1\hat{\mu}_2^T - \hat{\mu}_2\hat{\mu}_1^T)\\
&= X^TX - \frac{N_1^2}{N}\hat{\mu}_1\hat{\mu}_1^T - \frac{N_2^2}{N}\hat{\mu}_2\hat{\mu}_2^T - \frac{N_1N_2}{N}(\hat{\mu}_1\hat{\mu}_2^T - \hat{\mu}_2\hat{\mu}_1^T)\\
&= X^TX - \frac{1}{N}\tilde{\mu}\tilde{\mu}^T
\end{aligned}$$

This is one side of equation (15)

We also know that:

$$X^Ty = X^T(c_1\mathbb{1}_1 + c_2\mathbb{1}_2) = c_1N_1\mu_1 + c_2N_2\mu_2$$

And

$$\begin{aligned}
\frac{1}{N}\tilde{\mu}(N_1c_1 + N_2c_2) &= \frac{(c_1N_1^2 + c_2N_1N_2)}{N}\hat{\mu}_1 + \frac{(c_1N_1N_2 + c_2N_2^2)}{N}\hat{\mu}_2\\
&= \frac{(c_1N_1(N-N_2) + c_2N_1N_2)}{N}\hat{\mu}_1 + \frac{(c_1N_1N_2 + c_2N_2(N-N_1))}{N}\hat{\mu}_2\\
&= \frac{(c_1N_1(N-N_2) + c_2N_1N_2)}{N}\hat{\mu}_1 + \frac{(c_1N_1N_2 + c_2N_2(N-N_1))}{N}\hat{\mu}_2\\
&= c_1N_1\hat{\mu}_1 + c_2N_2\hat{\mu}_2 + \frac{N_1N_2}{N}\left((-c_1+c_2)\,\hat{\mu}_1 + (c_1-c_2)\right)\hat{\mu}_2)\\
&= c_1N_1\hat{\mu}_1 + c_2N_2\hat{\mu}_2 + \frac{N_1N_2}{N}(c_2-c_1)(\hat{\mu}_1-\hat{\mu}_2)
\end{aligned}$$

Then

$$X^Ty - \frac{1}{N}\tilde{\mu}(N_1c_1 + N_2c_2) = \frac{N_1N_2}{N}(c_2-c_1)(\hat{\mu}_1-\hat{\mu}_2)$$

Thus we have all components of our above equation (15)
Combining and simplifying gives

$$(N-2)\hat{\Sigma} + N\hat{\Sigma}_B = \frac{N_1 N_2}{N}(c_2 - c_1)(\hat{\mu}_1 - \hat{\mu}_2)$$

Substituting in our values for c_1 and c_2 gives $(N-2)\hat{\Sigma} + N\hat{\Sigma}_B = N(\hat{\mu}_1 - \hat{\mu}_2)$ as required.

c) Hence show that $\hat{\Sigma}_B\beta$ is in the direction $\hat{\mu}_2 - \hat{\mu}_1$ and thus

$$\hat{\beta} \propto \hat{\Sigma}^{-1}\hat{\mu}_2 - \hat{\mu}_1$$

For the direction:

$$\begin{aligned}
\hat{\Sigma}_B\beta &= \frac{N_1 N_2}{N^2}(\hat{\mu}_2 - \hat{\mu}_1)(\hat{\mu}_2 - \hat{\mu}_1)^T\beta \\
&= \frac{N_1 N_2}{N^2}(\hat{\mu}_2 - \hat{\mu}_1) \cdot \lambda \text{ where } \lambda \in \mathbb{R} \\
&= \lambda'(\hat{\mu}_2 - \hat{\mu}_1)
\end{aligned}$$

For proportionality:

$$\begin{aligned}
&\left[(N-2)\hat{\Sigma} + N\hat{\Sigma}_B\right]\hat{\beta} = N(\hat{\mu}_2 - \hat{\mu}_1) \\
&\Rightarrow (N-2)\hat{\Sigma}\hat{\beta} = N(\hat{\mu}_2 - \hat{\mu}_1 - \hat{\Sigma}_B\hat{\beta}) \\
&\Rightarrow (N-2)\hat{\Sigma}\hat{\beta} = N(\hat{\mu}_2 - \hat{\mu}_1 - \hat{\Sigma}_B\hat{\beta}) \\
&\Rightarrow (N-2)\hat{\Sigma}\hat{\beta} = N(\hat{\mu}_2 - \hat{\mu}_1 - \lambda'(\hat{\mu}_2 - \hat{\mu}_1)) \\
&\Rightarrow \hat{\Sigma}\hat{\beta} \propto \hat{\mu}_2 - \hat{\mu}_1 \\
&\Rightarrow \hat{\beta} \propto \hat{\Sigma}^{-1}(\hat{\mu}_2 - \hat{\mu}_1)
\end{aligned}$$

d) Show that c) holds for any (distinct) coding of the two classes.

This is simply the observation that $\frac{N_1 N_2}{N}(c_2 - c_1)$ is a scalar for any distinct coding.

e) Find the solution $\hat{\beta}_0$ up to the same scalar multiple as in c). Hence find the solution for the predicted value $\hat{f}(x) = \hat{\beta}_0 + x^T\hat{\beta}$. Consider the rule where we classify to class 2 if $\hat{f}(x) > 0$ and class 1 otherwise. Is this the same as the LDA rule? When?

$$\begin{aligned}
\hat{\beta}_0 &= \frac{1}{N}\mathbb{1}^T\left(y - X\hat{\beta}\right) \\
&= \frac{N_1 c_1 + N_2 c_2}{N} - \frac{\lambda}{N}\left(N_1\hat{\mu}_1 + N_2\hat{\mu}_2\right)^T\Sigma^{-1}\left(\hat{\mu}_2 - \hat{\mu}_1\right)
\end{aligned}$$

Where λ is the constant of proportionality from c). Using our encoding, we get

$$\hat{\beta}_0 = -\frac{\lambda}{N}\left(N_1\hat{\mu}_1 + N_2\hat{\mu}_2\right)^T \Sigma^{-1}\left(\hat{\mu}_2 - \hat{\mu}_1\right)$$

Then

$$\hat{f}(x) = \lambda\left(x - \frac{N_1}{N}\hat{\mu}_1 - \frac{N_2}{N}\hat{\mu}_2\right)^T \Sigma^{-1}\left(\hat{\mu}_2 - \hat{\mu}_1\right)$$

The classification is class 2 when:

$$x^T\Sigma^{-1}\left(\hat{\mu}_2 - \hat{\mu}_1\right) > \left(\frac{N_1}{N}\hat{\mu}_1 + \frac{N_2}{N}\hat{\mu}_2\right)^T \Sigma^{-1}\left(\hat{\mu}_2 - \hat{\mu}_1\right)$$

This is the same as the LDA classification when $N_1 = N_2$, but in general is different.[2]

4.3 Suppose we transform the original predictors X to $\hat{Y}$ via linear regression.

$$\hat{Y} = X\hat{B} = X(X^TX)^{-1}X^TY$$

where Y is the indicator response matrix. Similarly for any input $x \in \mathbb{R}^p$ we get $\hat{y} == \hat{B}^Tx \in \mathbb{R}^K$. Show that the LDA using $\hat{Y}$ is identical to the LDA in the original space.

The discriminant functions for LDA are:

$$\delta_k(x) = x^T\hat{\Sigma}_X^{-1}\mu_k - \frac{1}{2}\mu_k^T\hat{\Sigma}_X^{-1}\mu_k + log\pi_k$$

The discriminant rule using our transformed predictors is (denoting the mean by η):

$$\delta_k(\hat{y}) = \hat{y}^T\hat{\Sigma}_{\hat{Y}}^{-1}\eta_k - \frac{1}{2}\eta_k^T\hat{\Sigma}_{\hat{Y}}^{-1}\eta_k + log\pi_k$$

Where $\eta_k = \hat{B}^T\mu_k$.
A few things to note:

$$\begin{aligned}
\mu &= X^TYD^{-1} \\
\Sigma_X &= \frac{1}{N-K}\left(X^T - X^TYD^{-1}Y^T\right)\left(X^T - X^TYD^{-1}Y^T\right)^T \\
&= \frac{1}{N-K}X^T\left(I - YD^{-1}Y^T\right)^2X \\
&= \frac{1}{N-K}X^T\left(I - YD^{-1}Y^T\right)X
\end{aligned}$$

2 $N_1 = N_2$ gives $log(N_2/N_1) = 0$ and $N_1/N = N_2/N = 1/2$ so the classification rules become identical. To see the difference set $\mu_1 = 0$ say, and the result should be clear. The log term depends only on the relative number of samples in each class, whereas the expression above depends on the average of those samples.

Where D is the $K \times K$ diagonal matrix with $D_{i,i} = N_i$, the number of observations in class I, and the last line follows as $(I - YD^{-1}Y^T)^2 = (I - 2YD^{-1}Y^T + YD^{-1}Y^TYD^{-1}Y^T)$ and $Y^TY = D$

$$
\begin{aligned}
\hat{\Sigma}_{\hat{Y}} &= \frac{1}{N-K}\sum_{k=1}^{K}\sum_{g_i=k}(\hat{y}_i - \eta_i)(\hat{y}_i - \eta_i)^T \\
&= \frac{1}{N-K}\sum_{k=1}^{K}\sum_{g_i=k}\hat{B}^T(x_i - \mu_i)(x_i - \mu_i)^T\hat{B} \\
&= \hat{B}^T\left(\frac{1}{N-K}\sum_{k=1}^{K}\sum_{g_i=k}(x_i - \mu_i)(x_i - \mu_i)^T\right)\hat{B} \\
&= \hat{B}^T\hat{\Sigma}_X\hat{B} \\
&= \frac{1}{N-K}\hat{B}^TX^T\left(I - YD^{-1}Y^T\right)X\hat{B} \\
&= \frac{1}{N-K}\left(\hat{B}^TX^TX\hat{B} - \hat{B}^TX^TYD^{-1}Y^TX\hat{B}\right) \\
&= \frac{1}{N-K}\left(\hat{B}^TX^TY - \hat{B}^TX^TYD^{-1}Y^TX\hat{B}\right) \\
&= \frac{1}{N-K}\left(\Lambda - \Lambda D^{-1}\Lambda\right) \\
&= \frac{1}{N-K}\Lambda\left(I - D^{-1}\Lambda\right)
\end{aligned}
$$

Where $\Lambda = \hat{B}^TX^TY = Y^TX(X^TX)^{-1}X^TY$ is symmetric.

We will now consider each term of our discriminant function, vectorised to handle all cases at once.

$$
\begin{aligned}
\hat{Y}^T\hat{\Sigma}_{\hat{Y}}^{-1}\eta &= X^T\hat{B}\left(\hat{B}^T\hat{\Sigma}_X\hat{B}\right)^{-1}\hat{B}\mu \\
&= (N-K)X^T\hat{B}\left(\Lambda - \Lambda D^{-1}\Lambda\right)^{-1}\hat{B}^TX^TYD^{-1} \\
&= (N-K)X^T\hat{B}\left(\Lambda - \Lambda D^{-1}\Lambda\right)^{-1}\Lambda D^{-1} \\
&= (N-K)X^T\hat{B}\left(I - D^{-1}\Lambda\right)^{-1}D^{-1} \\
&= (N-K)X^T\Sigma_X^{-1}\Sigma_X\hat{B}\left(I - D^{-1}\Lambda\right)^{-1}D^{-1} \\
&= X^T\Sigma_X^{-1}\left(X^TX\hat{B} - X^TYD^{-1}Y^TX\hat{B}\right)\left(I - D^{-1}\Lambda\right)^{-1}D^{-1} \\
&= X^T\Sigma_X^{-1}\left(X^TY - X^TYD^{-1}\Lambda\right)\left(I - D^{-1}\Lambda\right)^{-1}D^{-1} \\
&= X^T\Sigma_X^{-1}X^TY\left(I - D^{-1}\Lambda\right)\left(I - D^{-1}\Lambda\right)^{-1}D^{-1} \\
&= X^T\Sigma_X^{-1}X^TYD^{-1} \\
&= X^T\Sigma_X^{-1}\mu
\end{aligned}
$$

So the first terms of our discriminant functions are equal. For the second term, again dealing with all classes at once:

$$\begin{aligned}
\eta_k^T \hat{\Sigma}_{\hat{Y}}^{-1} \eta &= (N-K)\, \mu_k^T \hat{B} \hat{\Sigma}_{\hat{Y}}^{-1} \hat{B}^T X^T Y D^{-1} \\
&= \mu_k^T \hat{B} \left(\hat{B}^T \hat{\Sigma}_X \hat{B} \right)^{-1} \hat{B}^T X^T Y D^{-1} \\
&= \mu_k^T \hat{\Sigma}_X^{-1} X^T Y D^{-1} \\
&= \mu_k^T \hat{\Sigma}_X^{-1} \mu
\end{aligned}$$

Where the penultimate line follows by the same algebra as used for the first term of our discriminant functions[3] The third term in each discriminant function is unchanged by our transformation as it depends only on the response. Thus all terms are equal and so the LDA on $\hat{Y}$ is equivalent to the LDA on X.

4.6 Convergence of the perceptron algorithm.

a) Please see textbook - existence of a β_sep such that $y_i \beta_{sep}^T z_i \geq 1$ for all i given separable data.

The data are separable so $\exists \beta$ s.t. $sign(\beta^T x_i^*) = y_i$. Thus we have $v_i := y_i \beta^T z_i > 0$ for every i. There are finitely many i, so we can set $\beta_{sep} = \beta / min_i v_i$ to get $y_i \beta_{sep}^T z_i \geq 1$ for all i.

b) Please see the textbook - convergence of the perceptron.

We update β via $\beta_{new} \leftarrow \beta_{old} + y_i z_i$

$$\begin{aligned}
\left\| \beta_{new} - \beta_{sep} \right\|^2 &= \left\| \beta_{old} + y_i z_i - \beta_{sep} \right\|^2 \\
&= \left(\beta_{old} + y_i z_i - \beta_{sep} \right)^T \left(\beta_{old} + y_i z_i - \beta_{sep} \right) \\
&= \left\| \beta_{old} - \beta_{sep} \right\|^2 - 2 y_i \beta_{sep}^T z_i + 2 y_i \beta_{old}^T z_i + y_i^2 z_i^T z_i \\
&= \left\| \beta_{old} - \beta_{sep} \right\|^2 + 2 y_i \left(\beta_{old} - \beta_{sep} \right)^T z_i + \| z \|_i^2 \\
&= \left\| \beta_{old} - \beta_{sep} \right\|^2 + 2 y_i \left(\beta_{old} - \beta_{sep} \right)^T z_i + 1 \\
&\leq \left\| \beta_{old} - \beta_{sep} \right\|^2 - 2 + 1 \\
&= \left(\beta_{old} - \beta_{sep} \right)^T \left(\beta_{old} - \beta_{sep} \right) - 1
\end{aligned}$$

Where the penultimate line follows as β_{old} misclassifies z_i - that is to say $y_i \beta_{old} z_i < 0$. Hence the term $y_i \left(\beta_{old} - \beta_{sep} \right)^T z_i$ is at most $-y_i \beta_{sep}^T z_i$, and we use part a).

3 Noting that when handling $\hat{Y}^T \hat{\Sigma}_{\hat{Y}}^{-1} \eta$ we could have dropped the leading X^T term throughout and dealt with $\hat{B} \hat{\Sigma}_{\hat{Y}}^{-1} \eta$ instead.

Hence the perceptron converges in at most $norm\beta_{old} - \beta_{sep}{}^2$ steps.

4.7 Consider the criterion

$$D^*(\beta) = -y^T X\beta$$

where X is prepended by a column of 1s. Consider maximising D^* subject to $\|\beta\| = 1$. Describe the criterion in words. Does it solve the optimal separating hyperplane problem?

This criterion seeks to maximise the sum of individual distances between the separating hyperplane and the response points. That is, it wants

$$\begin{aligned} &argmax_\beta\ y^T X\beta \\ &\|\beta\|_2 = 1 \end{aligned}$$

This in fact has an analytic solution such that $\beta \propto X^T y$ (one can replace the $\|\beta\|_2 = 1$ condition with $\|\beta\|_2^2 = 1$ to make life easier).

However the optimal separating hyperplane problem seeks to solve the following:

$$argmax_\beta\ min_i\ y_i x_i^T \beta$$

with the same norm constraint. It is a pointwise criterion, not a global one.

The difference is clear if you think of the two classes as having a large number of points in $\mathbb{R}^2$ at $(1,1)$ and $(-1,-1)$ respectively say, and one point from each class near $(0,1)$ and $(0,-1)$. The optimal separating hyperplane will be the line $x = 0$, whereas the D^* criterion will be close to the line $y = -x$.

Note that with some slight manipulation of the outlier points, it's entirely possible that D^* would misclassify some of the data even if it were separable.

5

CHAPTER 5 - BASIS EXPANSIONS AND REGULARISATION

5.1 Show that the truncated power basis functions represent a basis for a cubic spline with the two knots as indicated

We'll prove a more general result for p knots, $\xi_1, \ldots, \xi_p$ The basis described for these knots is:

$$\begin{aligned} h_1(X) &= 1 \\ h_2(X) &= X \\ h_3(X) &= X^2 \\ h_4(X) &= X^3 \\ h_{4+i}(X) &= (X - \xi_i)^3_+ \qquad \forall i \in \{1, \ldots, p\} \end{aligned}$$

We must show that given any cubic spline with knots at these places, it can be represented by a linear combination of the above and the representation is unique.

With $\xi_0 = -\infty$ and $\xi_{p+1} = \infty$, suppose that we have some piecewise cubic spline defined by:

$$f(X) = \sum_{i=0}^{p} f_i(X) \mathbb{1}_{[\xi_i, \xi_{i+1}]}$$

With $f_i(\xi_{i+1}) = f_{i+1}(\xi_{i+1})$, $0 \leq i \leq p$ and similar for first and second derivatives, each f_i being a cubic. Then let $g_i = f_{i+1} - f_i$, and we have $g_i(\xi_{i+1}) = 0$, $g_i'(\xi_{i+1}) = 0$, $g_i''(\xi_{i+1}) = 0$

g_i is a cubic, and so it is clear that g_i must take the form $a_i(X - \xi_{i+1})^3$ for some a_i. Adding in our indicator functions:

$$\begin{aligned} &f_i(X) \mathbb{1}_{[\xi_i, \xi_{i+1}]} + f_{i+1}(X) \mathbb{1}_{[\xi_{i+1}, \xi_{i+2}]} \\ &= f_i(X) \mathbb{1}_{[\xi_i, \xi_{i+2}]} + a_i(X - \xi_{i+1})^3 \mathbb{1}_{[\xi_{i+1}, \xi_{i+2}]} \end{aligned}$$

Applying this to each knot, we get:

$$f(X) = c_1 + c_2 X + c_3 X^2 + c_4 X^4 + \sum_{i=0}^{p} a_i (X - \xi_{i+1})^3 \mathbb{1}_{[\xi_i, \xi_{i+1}]}$$

For some constants c_i. Recasting this slightly gives

$$f(X) = \sum_{i=0}^{p+4} b_i h_i(X)$$

For some b_i as required.

For linear independence, note that all knots are distinct, and so if any of the h_i, $i > 4$ can be formed from a linear combination of the others, then left of ξ_i this combination must be zero. However this is an open set and a polynomial either has finitely many roots or is identically zero. Thus we have that the h_i form a basis as required.

5.4 Consider the truncated power series representation for cubic splines with K interior knots. Let

$$f(X) = \sum_{j=0}^{3} \beta_j X^j + \sum_{k=1}^{K} \theta_k (X - \xi_k)_+^3$$

Prove that the natural boundary conditions for the natural cubic splines (Section 5.2.1) imply the following linear constraints on the coefficients:

$$\begin{aligned} \beta_2 &= 0 \\ \beta_3 &= 0 \\ \sum_{k=1}^{K} \theta_k &= 0 \\ \sum_{k=1}^{K} \theta_k \xi_k &= 0 \end{aligned}$$

Hence derive the basis (5.4) and (5.5)

$f(X)$ is constrained to be linear for $X \leq \xi_1$ and $X \geq \xi_k$. On the left hand side, this implies that β_2 and β_3 are 0. On the right hand side, look at coefficients of the polynomial. The X^3 term has coefficient $\sum_{k=1}^{K} \theta_k$, and the X^2 term has coefficient $\sum_{k=1}^{K} \theta_k \xi_k$, which must then both be 0.

Thus we know that we can remove h_2 and h_3 in our basis from Ex 5.1.

Consider $N_{k+2}(X) = d_k(X) - d_{K-1}(X)$ in the book (5.4). Here the d_k are as in (5.5).

$$N_{k+2}(X) = \frac{(X-\xi_k)_+^3 - (X-\xi_K)_+^3}{\xi_k - \xi_K} - \frac{(X-\xi_{K-1})_+^3 - (X-\xi_K)_+^3}{\xi_{K-1} - \xi_K}$$

We want to write $\sum_{k=1}^{K} \theta_k (X-\xi_k)_+^3$ in terms of the N_{k+2} We will try with some coefficients a_k, and see if we can find a solution $\sum_{k=1}^{K} \theta_k (X - \xi_k)_+^3 = \sum_{k=1}^{K-2} a_k \theta_k N_{k+2}(X)$ Then:

$$\begin{aligned}
\sum_{k=1}^{K-2} a_k \theta_k N_{k+2}(X) &= \sum_{k=1}^{K-2} a_k \theta_k \left(d_k(X) - d_{K-1}(X)\right) \\
&= \sum_{k=1}^{K-2} a_k \theta_k \left(\frac{(X-\xi_k)_+^3 - (X-\xi_K)_+^3}{\xi_k - \xi_K} \right) \\
&- \left(\sum_{k=1}^{K-2} a_k \theta_k \right) \frac{(X-\xi_{K-1})_+^3 - (X-\xi_K)_+^3}{\xi_{K-1} - \xi_K}
\end{aligned}$$

To eliminate the denominator, let's try $a_k = \xi_k - \xi_K$

$$\begin{aligned}
\sum_{k=1}^{K-2} a_k \theta_k N_{k+2}(X) &= \sum_{k=1}^{K-2} \theta_k \left((X-\xi_k)_+^3 - (X-\xi_K)_+^3 \right) \\
&- \left(\sum_{k=1}^{K-2} (\xi_k - \xi_K) \theta_k \right) \frac{(X-\xi_{K-1})_+^3 - (X-\xi_K)_+^3}{\xi_{K-1} - \xi_K} \\
&= \sum_{k=1}^{K-2} \theta_k \left((X-\xi_k)_+^3 - (X-\xi_K)_+^3 \right) \\
&+ (\xi_{K-1}\theta_{K-1} - \xi_K \theta_{K-1}) \frac{(X-\xi_{K-1})_+^3 - (X-\xi_K)_+^3}{\xi_{K-1} - \xi_K} \\
&= \sum_{k=1}^{K-2} \theta_k (X-\xi_k)_+^3 + (\theta_{K-1} + \theta_K)(X-\xi_K)_+^3 \\
&+ (\xi_{K-1} - \xi_K)\theta_{K-1} \frac{(X-\xi_{K-1})_+^3 - (X-\xi_K)_+^3}{\xi_{K-1} - \xi_K} \\
&= \sum_{k=1}^{K-2} \theta_k (X-\xi_k)_+^3 + (\theta_{K-1} + \theta_K)(X-\xi_K)_+^3 \\
&+ \theta_{K-1}(X-\xi_{K-1})_+^3 - \theta_{K-1}(X-\xi_K)_+^3 \\
&= \sum_{k=1}^{K-2} \theta_k (X-\xi_k)_+^3 + \theta_K (X-\xi_K)_+^3 \\
&+ \theta_{K-1}(X-\xi_{K-1})_+^3 \\
&= \sum_{k=1}^{K} \theta_k (X-\xi_k)_+^3
\end{aligned}$$

Where use our θ conditions from above. Thus by setting $a_k = \xi_k - \xi_K$ we can in fact retrieve $f(X)$, and so we have K functions from which any element of the space can be generated. This space has dimension K and so (5.4) must define a basis as required.

5.7 Derivation of smoothing splines. Please see the book. g is a natural cubic spline interpolating N knots x_i $i \in \{1, \ldots, N\}$, with values $g(x_i) = z_i$. a and b are s.t. $a < x_1 < \ldots x_N < b$.

a) Let h is the difference between g and another differentiable function that interpolates the pairs (x_i, z_i).

$$\begin{aligned}
\int_a^b g''(x)h''(x)dx &= [g''(x)h'(x)]_a^b - \int_a^b g'''(x)h'(x)dx \\
&= [0 \cdot h'(b) - 0 \cdot h'(b)] - \int_{x_1}^{x_N} g'''(x)h'(x)dx \\
&= -\sum_{j=1}^{N-1} \int_{x_j}^{x_{j+1}} g'''(x)h'(x)dx \\
\int_{x_j}^{x_{j+1}} g'''(x)h'(x)dx &= [g'''(x)h(x)]_{x_j}^{x_{j+1}} - \int_{x_j}^{x_{j+1}} g''''(x)h(x)dx \\
&= [g'''(x)h(x)]_{x_j}^{x_{j+1}} \\
\Rightarrow \int_a^b g''(x)h''(x)dx &= -\sum_{j=1}^{N-1} [g'''(x)h(x)]_{x_j}^{x_{j+1}} \\
&= 0
\end{aligned}$$

The last line following as $h(x_i) = 0 \; \forall i$

b) Show that $\int_a^b \tilde{g}''(x)^2 dx \geq \int_a^b g''(x)^2 dx$ with equality only when $h = 0$ on $[a, b]$

$$\begin{aligned}
0 &= \int_a^b \tilde{g}''(x)^2 dx \\
&= \int_a^b \left(g''(x) + h''(x)\right)^2 dx \\
&= \int_a^b g''(x)^2 + 2 * g''(x)h''(x) + h''(x)^2 dx \\
&= \int_a^b g''(x)^2 dx + \int_a^b h''(x)^2 dx \\
&\geq \int_a^b g''(x)^2 dx
\end{aligned}$$

With the penultimate line following from a), and the final line being an equality only if h'' is identically o in $[a, b]$. That is to say h is linear in this interval, but at each x_i h is 0, and so given at least 2 x_i, the last line holds if and only if h is identically 0 in $[a, b]$

c) Consider the penalised least squares problem:

$$\min_f \left[\sum_{i=1}^{N} (y_i - f(x_i))^2 + \lambda \int_a^b f''(t)^2 dt \right]$$

Show that the minimiser must be a cubic spline with knots at each of the x_i

Let $\tilde{f}$ be the minimiser. let $z_i = \tilde{f}(x_i)$ for every i. Then we can find a natural cubic spline interpolating these points, g say. Thus $\sum_{i=1}^{N} \left(y_i - \tilde{f}(x_i)\right)^2 = \sum_{i=1}^{N} \left(y_i - g(x_i)\right)^2$

From b), we know that $\int_a^b \tilde{f}''(t)^2 dt \geq \int_a^b g''(t)^2 dt$, and so f can only be the minimiser if these values are in fact equal. Again from b), this only occurs if h is identically 0 in $[a, b]$, so $\tilde{f} = g$ in $[a, b]$, and the minimiser is a natural cubic spline.

5.9 Derive the Reinsch form $S_\lambda = (I + \lambda K)^{-1}$ for the smoothing spline.

We know $S_\lambda = N \left(N^T N + \lambda \Omega_N\right)^{-1} N^T$ We have a basis as in Ex 5.4, of the form $N_1(X) = 1$, $N_2(X) = X$, and $N_{k+2}(X) = d_k(X) - d_{K-1}(X)$ for $k <= N - 2$ where we have N knots. Let the knots be $x_1 < \cdots < x_n$ $d_j(x_i) = 0$ if $j \geq i$, and so our matrix N, which has $N_{i,j} = N_j(x_i)$ has all but the first two columns 0 above the diagonal [1]. The first column is all 1s and the second a column of x_i.

$$N = \begin{bmatrix} 1 & x_1 & 0 & \dots & 0 \\ 1 & x_2 & 0 & \dots & 0 \\ 1 & x_3 & N_3(x_3) & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_N & N_3(x_N) & \dots & N_N(x_N) \end{bmatrix}$$

We can subtract the first row form the rest to get:

$$N = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & x_2 - x_1 & 0 & \dots & 0 \\ 0 & x_3 - x_1 & N_3(x_3) & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & x_N - x_1 & N_3(x_N) & \dots & N_N(x_N) \end{bmatrix}$$

Now the determinant of N is the determinant of the submatrix consisting of N less the first row and column. This submatrix is lower triangular with a non-zero diagonal (as the x_i are distinct). Hence N is invertible.

1 $N_j(x_i) = 0$ for $i < j$ and $j >= 2$

Then:

$$\begin{aligned} S_\lambda &= N\left(N^T N + \lambda\Omega_N\right)^{-1} N^T \\ &= \left((N^T)^{-1}\left(N^T N + \lambda\Omega_N\right)N^{-1}\right)^{-1} \\ &= \left((N^T)^{-1}N^T N N^{-1} + \lambda(N^T)^{-1}\Omega_N N^{-1}\right)^{-1} \\ &= (I + \lambda K)^{-1} \end{aligned}$$

Where $K = (N^T)^{-1}\Omega_N N^{-1}$.

5.12 Characterise the solution to the following problem:

$$min_f RSS(f,\lambda) = \sum_{i=1}^{N} w_i \left(y_i - f(x_i)\right)^2 + \lambda \int f''(t)^2 dt$$

Where the $w_i \geq 0$ are the observation weights. Characterise the solution to the smoothing spline problem when the training data have ties in X

We know that the solution is a natural spline, and will reform the RSS as:

$$RSS(f,\theta) = (Y - N\theta)^T W (Y - N\theta) + \lambda\theta^T\Omega_N\theta$$

W being a diagonal matrix with $(W)_{i,i} = w_i$. Setting the first derivative to zero we get:

$$\begin{aligned} -2N^T W (Y - N\theta) + 2\lambda\Omega_N\theta &= 0 \\ \lambda\Omega_N\theta &= N^T W (Y - N\theta) \\ \left(N^T W N + \lambda\Omega_N\right)\theta &= N^T W Y \\ \theta &= \left(N^T W N + \lambda\Omega_N\right)^{-1} N^T W Y \end{aligned}$$

This is simply the solution to a weighted ridge regression.

Using the Reinsch form, we end up with $S_\lambda = (W + \lambda K)^{-1}$. In the smoothing spline problem with ties, given K values for x_i, of which L are unique, we can arrive at the weighted regression above on L parameters (knots), and N having dimension $L \times L$.

6

CHAPTER 6 - KERNEL SMOOTHING METHODS

6.1 Show that the Nadaraya-Watson kernel weighted average with fixed metric bandwidth λ and a Gaussian kernel is differentiable. What can be said for the Epanechnikov kernel? What can be said for the Epanechnikov kernel with adaptive nearest-neighbour bandwidth $\lambda(x_0)$

Nadaraya-Watson:

$$\hat{f}(x_0) = \frac{\sum_{i=1}^{N} K_\lambda(x_0, x_i) y_i}{\sum_{i=1}^{N} K_\lambda(x_0, x_i)}$$

Gaussian Kernel:

$$K_\lambda(x_0, x) = \frac{1}{\lambda\sqrt{2\pi}} \exp\left(-\frac{(x - x_0)^2}{2\lambda^2}\right)$$

Thus $K_\lambda(x_0, x)$ is strictly-positive and differentiable (in x_0 say) everywhere, so our denominator is too. Similarly the numerator is a linear combination of smooth functions so is smooth. Thus the Nadaraya-Watson kernel weighted average is differentiable everywhere[1] as a function of x_0.

The Epanechnikov kernel is defined as:

$$K_\lambda(x_0, x) = D\left(\frac{|x - x_0|}{\lambda}\right)$$
$$D(t) = \begin{cases} \frac{3}{4}(1 - t^2) & \text{if } |t| \leq 1 \\ 0 & \text{if otherwise} \end{cases}$$

Now note that differentiating piecewise gives:

$$\frac{\partial D(t)}{\partial t} = \begin{cases} -\frac{3}{2}t & \text{if } |t| \leq 1 \\ 0 & \text{if otherwise} \end{cases}$$

In particular, taking limits as $t \to 1$ from above and below gives that $D(t) = -\frac{3}{2}$ and $D(t) = 0$ respectively, and so D is not differentiable. For $t = \frac{|x - x_0|}{\lambda(x_0)}$,we have:

1 A ratio of smooth functions such that the denominator has no zeros is differentiable

$$\frac{\partial K_\lambda(x_0, x)}{\partial x_0} = \frac{\partial D}{\partial t}\frac{\partial t}{\partial x_0}$$

This will always have issues where $t = 1$, even if λ is differentiable in x_0.

6.2 Show that $\sum_{i=1}^N (x_i - x_0) l_i(x_0) = 0$ for local linear regression. Define $b_j(x_0) = \sum_{i=1}^N (x_i - x_0)^j l_i(x_0)$. Show that $b_0(x_0) = 1$ for local polynomial regression of any degree. Show that $b_j(x_0) = 0$ for $0 < j \leq k$ for local polynomial regression of degree k. What are the implications of this on the bias?

Let $l(x_0) = [l_1(x_0), \ldots, l_N(x_0)]^T$. Then using the notation of subsection 6.1.1, in particular (6.8), we have:

$$l(x_0)^T = b(x_0)^T \left(B^T W(x_0) B\right)^{-1} B^T W(x_0)$$

Where W is the diagonal matrix of kernel weights (i.e. $K_\lambda(x_0, x_i)$ and B being the $N \times (N+1)$ matrix with ith row $b(x_i)^T = [1, x, x^2, \ldots, x^N]$

Thus

$$\begin{aligned} l(x_0)^T B &= [\sum_{i=1}^N l_i(x_0), \sum_{i=1}^N l_i(x_0) x_i, \ldots, \sum_{i=1}^N l_i(x_0) x_i^N] \\ l(x_0)^T B &= b(x_0)^T \\ &= [1, x_0, \ldots, x_0^N] \end{aligned}$$

Comparing element-wise we see that $\sum_{i=1}^N l_i(x_0) = 1$ and $\sum_{i=1}^N l_i(x_0) x_i^j = x_0^j$. Thus $\sum_{i=1}^N l_i(x_0)(x_i - x_0) = 0$ and $b_0(x_0) = 1$. Lastly:

$$\begin{aligned} b_j(x_0) &= \sum_{i=1}^N (x_i - x_0)^j l_i(x_0) \\ &= \sum_{i=1}^N \sum_{k=0}^j \binom{j}{k} (-1)^{j-k} x_0^{j-k} x_i^k l_i(x_0) \\ &= \sum_{k=0}^j \binom{j}{k} (-1)^{j-k} x_0^{j-k} \sum_{i=1}^N x_i^k l_i(x_0) \\ &= \sum_{k=0}^j \binom{j}{k} (-1)^{j-k} x_0^{j-k} x_0^k \\ &= (x_0 - x_0)^j \\ &= 0 \end{aligned}$$

This suggests to bias is 0 to order k (i.e. if y is really a polynomial in x of degree at most k, the model is unbiased).

6.5 Show that fitting a locally constant multinomial logit model of the form (6.19) amounts to smoothing the binary response indicators for each class separately using a Nadaraya-Watson kernel smoother with kernel weights $K_\lambda(x_0, x_i)$

(6.19) The local log-likelihood for class j under a locally constant multinomial logit model is:

$$\sum_{i=1}^{N} K_\lambda(x_0, x_i) \left(\beta_j(x_0) - \log \left(1 + \sum_{j=1}^{J-1} \exp \beta_j(x_0) \right) \right)$$

In particular, writing $y_{i,j} = \delta_{y_i, g_j}$ for the function that is 1 if the ith observation is in class j and 0 otherwise[2], we can get:

$$l(\beta) = \sum_{i=1}^{N} K_\lambda(x_0, x_i) \left(\sum_{j=1}^{J-1} \beta_j(x_0) y_{i,j} - \log \left(1 + \sum_{j=1}^{J-1} \exp \beta_j(x_0) \right) \right)$$

We also have:

$$p_j = \mathbb{P}(G = j | X = x) = \frac{\exp(\beta_j)}{1 + \sum_{j=1}^{J-1} \exp(\beta_j}$$

Differentiating w.r.t β_j and setting the result to 0 we get:

$$\begin{aligned} \frac{\partial l(\beta)}{\partial \beta_j} &= \sum_{i=1}^{N} K_\lambda(x_0, x_i) \left(y_{i,j} - \frac{\exp(\beta_j)}{1 + \sum_{j=1}^{J-1} \exp(\beta_j} \right) \\ &= \sum_{i=1}^{N} K_\lambda(x_0, x_i) \left(y_{i,j} - p_j \right) \\ \Rightarrow p_j &= \frac{\sum_{i=1}^{N} K_\lambda(x_0, x_i) y_{i,j}}{\sum_{i=1}^{N} K_\lambda(x_0, x_i)} \end{aligned}$$

This condition for selecting the β_j is exactly the result of smoothing each class separately under a Nadaraya-Watson kernel smoother.

6.7 Derive an expression for the leave-one-out cross-validated RSS for local polynomial regression

$$\frac{1}{N} \sum_{i=1}^{N} \sum_{j \neq i} K_\lambda(x_j, x_i)^2 \left\| y - x_i^T \beta_j \right\|^2$$

2 This is an explicit form of the binary response indicator vector

6.8 Suppose that for continuous response Y and predictor X, we model the joint density of X, Y using a multivariate Gaussian kernel estimator. Note that the kernel in this case would be the product kernel $\phi_\lambda(X)\phi_\lambda(Y)$. Show that the conditional mean $\mathbb{E}[Y|X]$ derived from this estimate is a Nadaraya-Watson estimator. Extend this result to classification by providing a suitable kernel for the estimation of the joint distribution of a continuous X and discrete Y

$$\hat{f}_{X,Y}(x,y) = \frac{1}{N}\sum_{i=1}^{N}\phi_\lambda(x-x_i)\phi_Y(y-y_i)$$
$$\hat{f}_X(x) = \frac{1}{N}\sum_{i=1}^{N}\phi_\lambda(x-x_i)$$

Thus

$$\begin{aligned}
\mathbb{E}[Y|X] &= \int_{-\infty}^{\infty}\frac{\hat{f}_{X,Y}(x,y)y}{\hat{f}_X(x)}dy \\
&= \int_{-\infty}^{\infty}\frac{\sum_{i=1}^{N}\phi_\lambda(x-x_i)\phi_\lambda(y-y_i)y}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)}dy \\
&= \frac{\sum_{i=1}^{N}\phi_\lambda(x-x_i)\int_{-\infty}^{\infty}y\phi_\lambda(y-y_i)dy}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)} \\
&= \frac{\sum_{i=1}^{N}\phi_\lambda(x-x_i)\int_{-\infty}^{\infty}y\phi_\lambda(y-y_i)dy}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)} \\
\int_{-\infty}^{\infty}y\phi_\lambda(y-y_i)dy &= \int_{-\infty}^{\infty}(y-y_i)\phi_\lambda(y-y_i)dy + y_i\int_{-\infty}^{\infty}\phi_\lambda(y-y_i)dy \\
&= \mathbb{E}[Y] + y_i \\
&= y_i \\
\Rightarrow \mathbb{E}[Y|X] &= \frac{\sum_{i=1}^{N}\phi_\lambda(x-x_i)y_i}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)}
\end{aligned}$$

Thus the conditional expectation is a Nadaraya-Watson estimator.

For classification, let our kernel be $\phi_\lambda(X)\delta_g(Y)$ where:

$$\delta_g(y) = \begin{cases} \delta_g(y) = 1 & \text{if } y_i = g \\ 0 & \text{if otherwise} \end{cases}$$

That is $\hat{f}_{X,Y}(x,y) = \frac{1}{N}\sum_{i=1}^{N}\phi_\lambda(x-x_i)\delta_y(y_i)$. Then

$$\mathbb{P}(Y=y|X=x) = \frac{\sum_{y_i=y}\phi_\lambda(x-x_i)}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)}$$

This definition makes intuitive sense - in the case where we instead had X as a constant function say, it would be the sample estimate. Once again the expected value becomes a Nadaraya-Watson estimator[3]:

$$\begin{aligned}\mathbb{E}\left[Y|X\right] &= \frac{\sum_{j=1}^{M}\sum_{y_i=y_j}\phi_\lambda(x-x_i)y_i}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)} \\ &= \frac{\sum_{i=1}^{N}\phi_\lambda(x-x_i)y_i}{\sum_{i=1}^{N}\phi_\lambda(x-x_i)}\end{aligned}$$

3 Where we have j classes for the response, $y_1, \ldots, y_j$

7

CHAPTER 7 - MODEL ASSESSMENT AND SELECTION

7.1 Derive the estimate of in-sample error (7.24):

$$\mathbb{E}_y\left(Err_{in}\right) = \mathbb{E}_y\left(\overline{err}\right) + 2 \cdot \frac{d}{N}\sigma_\epsilon^2$$

We start from (7.22):

$$\mathbb{E}_y\left(Err_{in}\right) = \mathbb{E}_y\left(\overline{err}\right) + \frac{2}{N}\sum_{i=1}^{N} Cov(y_i, \hat{y}_i)$$

Now note that $\sum_{i=1}^{N} Cov(y_i, \hat{y}_i) = tr(Cov(y, \hat{y}))$, and all we wish to show is that this equals $d\sigma_\epsilon^2$ when fitting a linear model with d predictors.

$$\begin{aligned}
Cov(y, \hat{y}) &= Cov(y, X(X^TX)^{-1}X^Ty) \\
&= X(X^TX)^{-1}X^TCov(y, y) \\
&= X(X^TX)^{-1}X^T\sigma_\epsilon^2 \\
tr(Cov(y, \hat{y})) &= tr(X(X^TX)^{-1}X^T)\sigma_\epsilon^2 \\
&= tr(X^TX(X^TX)^{-1})\sigma_\epsilon^2 \\
&= tr(I_d)\sigma_\epsilon^2 \\
&= d\sigma_\epsilon^2
\end{aligned}$$

7.2 Please see the book.

a) Here $\mathbb{P}(Y = 1|x_0) = f(x_0)$

Let $p = \mathbb{P}(\hat{G}(x_0) = G(x_0))$.

$$\begin{aligned}
Err(x_0) &= \mathbb{P}(Y \neq \hat{G}(x_0)|X = x_0) \\
&= \mathbb{P}(Y \neq G(x_0)|X = x_0) \cdot p + \mathbb{P}(Y = G(x_0)|X = x_0) \cdot (1-p) \\
&= \mathbb{P}(Y \neq G(x_0)|X = x_0) \cdot p + \mathbb{P}(Y = G(x_0)|X = x_0) \cdot (1-p) \\
&+ \mathbb{P}(Y \neq G(x_0)|X = x_0) \cdot ((1-p) - (1-p)) \\
&= \mathbb{P}(Y \neq G(x_0)|X = x_0) \cdot 1 \\
&+ (\mathbb{P}(Y = G(x_0)|X = x_0) - \mathbb{P}(Y \neq G(x_0)|X = x_0)) \cdot (1-p) \\
&= Err_B(x_0) \\
&+ (\mathbb{P}(Y = G(x_0)|X = x_0) - \mathbb{P}(Y \neq G(x_0)|X = x_0)) \cdot (1-p) \\
&= Err_B(x_0) + \begin{cases} (2f(x_0) - 1)\,\mathbb{P}(\hat{G}(x_0) \neq G(x_0)) & \text{if } G(x_0) = 1 \\ (1 - 2f(x_0))\,\mathbb{P}(\hat{G}(x_0) \neq G(x_0)) & \text{if } G(x_0) = 0 \end{cases}
\end{aligned}$$

Now $G(x_0) = 1 \iff f(x_0) \geq 0.5 \iff (2 \cdot f(x_0) - 1) > 0$. Thus the last expression simplifies to:

$$Err_B(x_0) + |2 \cdot f(x_0) - 1| \cdot \mathbb{P}(\hat{G}(x_0) \neq G(x_0))$$

b)

Let $\sigma = \sqrt{Var(f(x_0))}$ and let $\mu = \mathbb{E}(\hat{f}(x_0))$. If$G(x_0) = 1$:

$$\begin{aligned}
\mathbb{P}(\hat{G}(x_0) \neq G(x_0)) &= \begin{cases} \mathbb{P}(\hat{f}(x_0) < 0.5) & \text{if } G(x_0) = 1 \\ \mathbb{P}(\hat{f}(x_0) > 0.5) & \text{if } G(x_0) = 0 \end{cases} \\
&\approx \begin{cases} \Phi\left(\frac{0.5-\mu}{\sigma}\right) & \text{if } G(x_0) = 1 \\ 1 - \Phi\left(\frac{0.5-\mu}{\sigma}\right) & \text{if } G(x_0) = 0 \end{cases} \\
&= \begin{cases} \Phi\left(\frac{0.5-\mu}{\sigma}\right) & \text{if } G(x_0) = 1 \\ \Phi\left(\frac{mu-0.5}{\sigma}\right) & \text{if } G(x_0) = 0 \end{cases} \\
&= \Phi\left(\frac{sign(0.5 - f(x_0))(\mu - 0.5)}{\sigma}\right)
\end{aligned}$$

As required.

7.3. Please see the book. Linear smoothing of y

a)

$\hat{f} = Sy$ is a linear smoothing of y. x_i is a column vector, and $X = [x_1, \ldots, x_N]^T$. $S = X(X^TX)^{-1}X^T$

$$\hat{f}^{-i}(x_i) = x_i^T(X^TX - x_ix_i^T)^{-1}\left(X^Ty - x_iy_i\right)$$

We use the following result:

$$(I + uv^T)^{-1} = I - \frac{uv^T}{1 + v^Tu}$$

This is verifiable by setting $A = (I + uv^T)$ and $BI - \frac{uv^T}{1+v^Tu}$ for some column vectord u, v with $v^Tu \neq -1$, then checking that $AB = BA = I$.

Now $X^TX - x_ix_i^T = X^TX(1 - (X^TX)^{-1}x_ix_i^T)$. Hence:

$$\begin{aligned}
\left(X^TX + x_ix_i^T\right)^{-1} &= (1 - (X^TX)^{-1}x_ix_i^T)^{-1}(X^TX)^{-1} \\
&= (I - (X^TX)^{-1}x_ix_i^T)^{-1}(X^TX)^{-1} \\
&= \left(I + \frac{(X^TX)^{-1}x_ix_i^T}{1 - x_i^T(X^TX)^{-1}x_i}\right)(X^TX)^{-1} \\
&= (X^TX)^{-1} + \frac{(X^TX)^{-1}x_ix_i^T(X^TX)^{-1}}{1 - x_i^T(X^TX)^{-1}x_i} \\
&= (X^TX)^{-1} + \frac{(X^TX)^{-1}x_ix_i^T(X^TX)^{-1}}{1 - S_{ii}} \\
\hat{f}^{-i}(x_i) &= x_i(X^TX)^{-1}X^Ty \\
&\quad + x_i^T\frac{(X^TX)^{-1}x_ix_i^T(X^TX)^{-1}}{1 - S_{ii}}X^Ty \\
&\quad - x_i^T(X^TX)^{-1}x_iy_i \\
&\quad - x_i^T\frac{(X^TX)^{-1}x_ix_i^T(X^TX)^{-1}}{1 - S_{ii}}x_iy_i \\
&= \hat{f}(x_i) + \frac{S_{ii}}{1 - S_{ii}}\hat{f}(x_i) \\
&\quad - S_{ii}y_i - \frac{S_{ii}^2}{1 - S_{ii}}y_i \\
&= \hat{f}(x_i) + \frac{S_{ii}}{1 - S_{ii}}\hat{f}(x_i) - \frac{S_{ii}}{1 - S_{ii}}y_i \\
&= \frac{1}{1 - S_{ii}}\hat{f}(x_i) - \frac{S_{ii}}{1 - S_{ii}}y_i
\end{aligned}$$

So:

$$\begin{aligned}
y_i - \hat{f}^{-i}(x_i) &= y_i - \left(\frac{1}{1 - S_{ii}}\hat{f}(x_i) - \frac{S_{ii}}{1 - S_{ii}}y\right) \\
&= -\frac{1}{1 - S_{ii}}\hat{f}(x_i) + y_i + \frac{S_{ii}}{1 - S_{ii}}y_i \\
&= -\frac{1}{1 - S_{ii}}\hat{f}(x_i) + \frac{1}{1 - S_{ii}}y_i \\
&= \frac{1}{1 - S_{ii}}\left(y_i - \hat{f}(x_i)\right)
\end{aligned}$$

With smoothing splines, we have $S = N^T(N^TN + \lambda\Omega)^{-1}N$, and a similar trick can be pulled using $N^TN + \lambda\Omega$ in place of X^TX and adjusting appropriately.

b)

Under OLS, S is symmetric and idempotent so[1] $0 \leq S_{ii} \leq 1$ for every i. Thus $\left|y_i - \hat{f}^{-i}(x_i)\right| \geq \left|y_i - \hat{f}(x_i)\right|$

c)

Unsolved.

7.4 Model Optimism. Please see the book, p258

We will use the following (substituting Y_i^0 as y_i to clear up notation). and writing $\hat{y}_i$ for $\hat{f}(x_i)$:

$$y_i - \hat{y}_i = (y_i - f(x_i)) + (f(x_i) - \mathbb{E}\hat{y}_i) + (\mathbb{E}\hat{y}_i - \hat{y}_i)$$

$$\begin{aligned}
(y_i - \hat{y}_i)^2 = (y_i - f(x_i))^2 & \quad (16)\\
+ (f(x_i) - \mathbb{E}(\hat{y}_i))^2 & \quad (17)\\
+ (\hat{y}_i - \mathbb{E}\hat{y}_i)^2 & \quad (18)\\
+ 2(\mathbb{E}\hat{y}_i - \hat{y}_i)(f(x_i) - \mathbb{E}(\hat{y}_i)) & \quad (19)\\
+ 2(\mathbb{E}\hat{y}_i - \hat{y}_i)(y_i - f(x_i)) & \quad (20)\\
+ 2(f(x_i) - \mathbb{E}(\hat{y}_i)))(y_i - f(x_i)) & \quad (21)
\end{aligned}$$

When we look at $\mathbb{E}\left(Err_{in} - \overline{err}\right)$, we will have many terms vanish - in particular (17), (18), and (19) will be the same for each side, and so will disappear in the difference.

Term (16) captures the error inherent in the model - y truly has the form $f(X) + \epsilon$, so under the expectation (16) will become the variance of ϵ, independent of whether we use y or Y^0. Hence this vanishes in the difference too. Term (21) has expectation 0, as the left part is unchanged under expectation, but $\mathbb{E}(y_i - f(x_i)) = \mathbb{E}\epsilon_i = 0$

Thus we are left with:

$$\mathbb{E}\left(Err_{in} - \overline{err}\right) = \frac{2}{N}\mathbb{E}\sum_i(\mathbb{E}\hat{y}_i - \hat{y}_i)\left[(\mathbb{E}_{Y^0}(Y_i^0 - f(x_i)) - (y_i - f(x_i))\right]$$

The first term on the in the square brackets vanishes as $\mathbb{E}_{Y^0}(Y_i^0) = f(x_i)$, and the second term is equal to $y_i - \mathbb{E}y_i$.

1 Consider that $S_{i,j} = (S^2)_{i,j}$ and explore the expanson when looking at the diagonal.

Hence:

$$\begin{aligned}\mathbb{E}\left(Err_{in} - \overline{err}\right) &= -\frac{2}{N}\sum_i (\mathbb{E}\hat{y}_i - \hat{y}_i)\mathbb{E}\left(y_i - \mathbb{E}y_i\right) \\ &= \frac{2}{N}\sum_i (\hat{y}_i - \mathbb{E}\hat{y}_i)\mathbb{E}\left(y_i - \mathbb{E}y_i\right) \\ &= \frac{2}{N}\sum_i Cov(\hat{y}_i, y_i)\end{aligned}$$

7.5 For a linear smoother $\hat{y} = Sy$ show that

$$\sum_i Cov(\hat{y}_i, y_i) = tr(S)\sigma_\epsilon^2$$

which justifies its use as the effective number of parameters.

$$\begin{aligned}\sum_i Cov(\hat{y}_i, y_i) &= tr(Cov(\hat{y}, y)) \\ &= tr(Cov(Sy, y)) \\ &= tr(SCov(y, y)) \\ &= tr(SVar(y)) \\ &= tr(S)\sigma_\epsilon^2\end{aligned}$$

Show that for the additive error model, the effective degrees-of-freedom for the k-nearest-neighbours regression fit is N/k

Under this model, S is a binary matrix with values $1/k$ or 0, such that each row sums to 1 and the diagonal entries are all $1/k$. This thus has trace N/k, which by 7.5 is our effective degrees-of-freedom.

7.7 Use the Taylor expansion of $1/(1-x)^2$ to expose the relationship between C_p and GCV. The main difference being the model used to estimate the noise variance σ_ϵ^2

$$\begin{aligned}
GCV &= \frac{1}{N}\sum_i \left[\frac{y_i - \hat{y}_i}{1 - tr(S)/N}\right]^2 \\
&= \frac{1}{N}\sum_i (y_i - \hat{y}_i)^2 \left(1 + 2\frac{tr(S)}{N}\right) \\
&= \frac{1}{N}\left(1 + 2\frac{d}{N}\right)\sum_i (y_i - \hat{y}_i)^2 \\
&= \frac{1}{N}\sum_i (y_i - \hat{y}_i)^2 + 2\frac{d}{N^2}\sum_i (y_i - \hat{y}_i)^2 \\
&= \overline{err} + 2\frac{d}{N^2}\sum_i (y_i - \hat{y}_i)^2 \\
&\approx \overline{err} + 2\frac{d}{N^2} N\sigma_\epsilon^2 \\
&= \overline{err} + 2\frac{d}{N}\sigma_\epsilon^2 \\
&= C_p
\end{aligned}$$

Show that the set of functions $\{I(\sin \alpha x > 0)\}$ can shatter the following points on the line:

$$\{z^i = 10^{-i} | i \in \{1, \ldots, l\}\}$$

Hence the VC dimension of this class is infinite.

Pick any i. Then the distance from z^i to z^{i-1} is greater than 10^{-i}. We just need a function that can separate these, and hence can find functions for any i. Then take $\alpha = 10^i\pi$. $\sin x$ crosses the axis at multiples of π, so $\sin \alpha x$ crosses the axis when x is an integer multiple of 10^{-i}. Hence our indicator function will cross the axis between z^l and z^{i-1} (multiple times infact!). Hence the functions can shatter arbitrarily long sequences of points on the line as required.

8

CHAPTER 8 - MODEL INFERENCE AND AVERAGING

Notes on the text. P 269, deriving the posterior distribution for β. This seems like a useful derivation

We assume that $\beta \sim N(0, \tau\Sigma)$, and have the model $y = H\beta + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$. From here we get:

$$y|\beta \sim N(H\beta, \sigma^2 I)$$

Then:

$$\begin{aligned}
f(\beta|y) &\propto f(\beta)f(y|\beta) \\
l(\beta|y) &= c + l(\beta) + l(y|\beta) \\
&= c - \frac{1}{2\tau}\beta^T\Sigma^{-1}\beta + \log\frac{1}{\sqrt{2\pi\tau\,|\Sigma|}} \\
&- \frac{1}{2\sigma^2}(y - H\beta)^T(y - H\beta) + \log\frac{1}{\sqrt{2\pi}\sigma} \\
&= c - \frac{1}{2\sigma^2}(y - H\beta)^T(y - H\beta) - \frac{1}{2\tau}\beta^T\Sigma^{-1}\beta \\
&= c - \frac{1}{2}\left(\frac{1}{\sigma^2}y^Ty + \frac{1}{\sigma^2}\beta^TH^TH\beta - \frac{2}{\sigma^2}\beta^TH^Ty\right) - \frac{1}{2\tau}\beta^T\Sigma^{-1}\beta \\
&= c - \frac{1}{2}\left(\beta^T\left(\frac{1}{\sigma^2}H^TH + \frac{1}{\tau}\Sigma^{-1}\right)\beta - \frac{2}{\sigma^2}\beta^TH^Ty\right)
\end{aligned}$$

Where we have absorbed terms not involving β into c at various points. Let $\Lambda = H^TH + \frac{\sigma^2}{\tau}\Sigma^{-1}$

$$\begin{aligned}
\frac{1}{\sigma^2}\left(\beta^T\Lambda\beta - 2y^TH\beta\right) &= \frac{1}{\sigma^2}\left(\beta^T\Lambda\beta - 2\beta^TH^Ty\right) \\
&= \frac{1}{\sigma^2}(\beta - \Lambda^{-1}H^Ty)^T\Lambda(\beta - \Lambda^{-1}H^Ty) + c
\end{aligned}$$

Again using a constant c to absorb non-β terms. Knowing that our result is to be a normal distribution, one can fill in the details for constants to recover that:

$$\begin{aligned}
\mathbb{E}\left[\beta|Z\right] &= \Lambda^{-1}H^Ty \\
Cov\left[\beta|Z\right] &= \Lambda^{-1}\sigma^2
\end{aligned}$$

This is very similar to an earlier exercise on posterior probabilities, except we now have a more general covariance matrix.

8.1. Let $r(y)$ and $q(y)$ be probability density functions. Jensen's inequality states that for a random variable X and a convex function $\phi(x)$, $\mathbb{E}[\phi(X)] \geq \phi(\mathbb{E}[X])$. Use Jensen's inequality to show that:

$$\mathbb{E}_q \log\left[\frac{r(Y)}{q(Y)}\right]$$

is maximised as a function of $y(y)$ when $r(y) = q(y)$. Hence show that $R(\theta, \theta) \geq R(\theta', \theta)$ as stated below equation (8.46)

log is concave, so

$$\begin{aligned}
\mathbb{E}_q \log\left[\frac{r(Y)}{q(Y)}\right] &\leq \log \mathbb{E}_q\left[\frac{r(Y)}{q(Y)}\right] \\
&= \log \int \frac{r(Y)}{q(Y)} dq \\
&= \log \int \frac{r(Y)}{q(Y)} q(y) dy \\
&= \log(1) \\
&= 0
\end{aligned}$$

Equality holds iff $r(y) = q(y)$. Thus:

$$\begin{aligned}
R(\theta', \theta) - R(\theta, \theta) &= \mathbb{E}_{T|Z,\theta}\left[\log \mathbb{P}\left(Z^m|Z, \theta'\right) - \log \mathbb{P}\left(Z^m|Z, \theta\right)\right] \\
&= \mathbb{E}_{T|Z,\theta}\left[\log \frac{\mathbb{P}\left(Z^m|Z, \theta'\right)}{\mathbb{P}\left(Z^m|Z, \theta\right)}\right] \\
&\leq 0
\end{aligned}$$

Where the last line follows from the previous part of the question, and equality holds iff $R(\theta, \theta) = R(\theta', \theta)$

Consider the maximisation of the log-likelihood (8.48), over distributions $\tilde{P}(Z^m) \geq 0$ and $\sum_{Z^m} \tilde{P}(Z^m) = 1$. Use Lagrange multipliers to show that the solution is the conditional distribution $\tilde{P}(Z^m) = \mathbb{P}(Z^m|Z, \theta')$ as in (8.49)

$$\begin{aligned}
F(\theta', \tilde{P}) &= \mathbb{E}_{\tilde{P}}\left[l_0\left(\theta'; T\right)\right] - \mathbb{E}_{\tilde{P}}\left[\log \tilde{P}(Z^m)\right] \\
&= \sum_{Z^m}\left[\log\left(\mathbb{P}(Z^m|Z, \theta')\mathbb{P}(Z|\theta')\right) - \log\left(\tilde{P}(Z^m)\right)\right]\tilde{P}(Z^m) \\
&= \sum_{Z^m} \log\left(\frac{\mathbb{P}(Z^m|Z, \theta')\mathbb{P}(Z|\theta')}{\tilde{P}(Z^m)}\right)\tilde{P}(Z^m) \\
&= \log\left(\mathbb{P}(Z|\theta')\right) + \sum_{Z^m} \log\left(\frac{\mathbb{P}(Z^m|Z, \theta')}{\tilde{P}(Z^m)}\right)\tilde{P}(Z^m)
\end{aligned}$$

Let x be any observation of $\tilde{P}(Z^m)$. We have

$$\begin{aligned} g(x,\lambda) &= \sum_{Z^m} \log\left(\frac{\mathbb{P}(Z^m|Z,\theta')}{\tilde{P}(Z^m)}\right)\tilde{P}(Z^m) - \lambda\left(\sum_{Z^m}\tilde{P}(Z^m) - 1\right) \\ \frac{\partial g(x,\lambda)}{\partial x} &= 0 + \frac{\partial \log\left(\frac{\mathbb{P}(Z^m|Z,\theta')}{x}\right)x}{\partial x} - \lambda \\ &= \log\frac{\mathbb{P}(Z^m|Z,\theta')}{x} - 1 - \lambda \\ &= 0 \\ \Rightarrow x &= \exp(-1-\lambda)\,\mathbb{P}(Z^m|Z,\theta') \end{aligned}$$

Now taking partial derivatives w.r.t λ gives $\sum_{Z^m} x = 1$ Hence:

$$\exp(-1-\lambda) = 1$$

Thus $x = \mathbb{P}(Z^m|Z,\theta')$, but also $x = \tilde{P}(Z^m)$ as it was an observation, recovering the result.

8.3 Justify the estimate (8.50)

$$\begin{aligned} \widehat{\mathbb{P}_{U_k}(u)} &= \int \mathbb{P}(u|u_l, l\neq k)d(\mathbb{P}(u_l)) \\ &= \int \mathbb{P}(u|u_l, l\neq k)\mathbb{P}(u_l)du_l \\ &\approx \sum_m^M \mathbb{P}(u|u_l, l\neq k)\frac{\sum_{l\neq k}\mathbb{1}_{U_l^{(t)}=u_l}}{M-m+1} \\ &= \frac{1}{M-m+1}\sum_m^M \mathbb{P}(u|U_l^{(t)}, l\neq k) \end{aligned}$$

8.4 Bagging reduces variance, please see the book

From (8.7), we have:

$$\hat{\mu}^*(x) \sim N(\hat{\mu}(x), h(x)^T(H^TH)^{-1}h(x)\hat{\sigma}^2)$$

Let $\hat{\Sigma}(x) = h(x)^T(H^TH)^{-1}h(x)\hat{\sigma}^2$
Then using the fact that different bootstraps are independent:

$$\begin{aligned} \hat{f}_{bag} &= \frac{1}{B}\sum_b \hat{f}_b^*(x) \\ &\sim N(\hat{\mu(x)}, \frac{B}{B^2}\hat{\Sigma}(x)) \\ &= N(\hat{f(x)}, \frac{1}{B}\hat{\Sigma}(x)) \\ &\xrightarrow{B\to\infty} \hat{f}(x) \end{aligned}$$

8.7 EM as a Majorization-Minorization algorithm. Please see the textbook.

$$\begin{aligned} l(\theta'; Z) &= Q(\theta', \theta) - R(\theta', \theta) \\ &\geq Q(\theta', \theta) - R(\theta, \theta) \\ &= Q(\theta', \theta) - \mathbb{E}_{T|Z,\theta} \left[\log \mathbb{P}\left(Z^m | Z, \theta\right) \right] \\ &= Q(\theta', \theta) - \mathbb{E}_{T|Z,\theta} \left[\log \frac{\mathbb{P}\left(T|\theta\right)}{\mathbb{P}\left(Z|\theta\right)} \right] \\ &= Q(\theta', \theta) + \mathbb{E}_{T|Z,\theta} \left[\log \mathbb{P}\left(Z|\theta\right) \right] - \mathbb{E}_{T|Z,\theta} \left[\log \mathbb{P}\left(T|\theta\right) \right] \\ &= Q(\theta', \theta) + \log \mathbb{P}\left(Z|\theta\right) - Q(\theta, \theta) \end{aligned}$$

In short:

$$l(\theta'; Z) \geq Q(\theta', \theta) + \log \mathbb{P}\left(Z|\theta\right) - Q(\theta, \theta) \tag{22}$$

With equality holding only for $\theta = \theta'$. Thus the RHS (22) of minorizes $l(\theta'; Z)$.

In step 3 of the EM algorithm (Algorithm 8.2), we maximise Q over θ'. This is equivalent to maximising (22) over θ'. Now in general if $g(x, y)$ minorizes $f(x)$, under the update $x^{s+1} = argmax_x g(x, x^s)$, we have that $f(x^{s+1}) \geq g(x^{s+1}, x^s) \geq g(x^s, x^s) = f(x)$, and so the update in step 3 of the EM algorithm uses (22) to minorize the log likelihood and force it to increase until $\theta = \theta'$, as this is where equality holds.

9

CHAPTER 9 - ADDITIVE MODELS, TREES, AND RELATED METHODS

9.2a Please see book

The ith row of (9.33) in the book gives:

$$f_i + \sum_{j \neq i} S_i f_j = S_j y$$

Hence:

$$f_i = S_i \left[y - sum_{j \neq i} f_j \right]$$

This is exactly the back-fitting step given that a smoothing spline fit to a mean-zero response has mean zero.

9.2b Please see book

We can take the eigendecomposition of the S_i, $S_i = U_i D_i U_i^T$, with D_i diagonal and the U_i orthonormal. Then:

$$\begin{aligned} f_1 &\leftarrow S_1 \left[y - f_2 \right] \\ f_2 &\leftarrow S_2 \left[y - f_1 \right] \end{aligned}$$

$$\begin{aligned} f_1 &\leftarrow U_1 D_1 U_1^T \left[y - f_2 \right] \\ f_2 &\leftarrow U_2 D_2 U_2^T \left[y - f_1 \right] \end{aligned}$$

$$\begin{aligned} U_1^T f_1 &\leftarrow D_1 U_1^T \left[y - f_2 \right] \\ U_2^T f_2 &\leftarrow D_2 U_2^T \left[y - f_1 \right] \end{aligned}$$

Let $g_i = U_i^T f_i$, and let $z_i = D_i U_i^T y_i$

$$\begin{aligned} g_1 &\leftarrow z_1 - D_1 g_2 \\ g_2 &\leftarrow z_2 - D_2 g_1 \end{aligned}$$

Hence:

$$\begin{aligned} g_1 &\leftarrow z_1 - D_1 g_2 \\ &= z_1 - D_1 z_2 + D g_1 \\ g_2 &\leftarrow z_1 - D_2 g_1 \\ &= z_2 - D_2 z_1 + D g_2 \end{aligned}$$

Here $D = D_1 D_2$ is diagonal with values in $[0, 1)$. Hence

$$g_1 = (I - D)^{-1}(z_1 - D_1 z_2)$$

and similar for g_2. Lastly:

$$\begin{aligned} f_1 &= U_1(I - D)^{-1}(D_1 U_1^T y_1 - D U_2^T y_2) \\ f_2 &= U_2(I - D)^{-1}(D_2 U_2^T y_2 - D U_1^T y_1) \end{aligned}$$

9.4 please see book

$$f_i \leftarrow S\left[y - f_j\right] = Sy - S^2 y + S^2 f_j$$

Hence:

$$\begin{aligned} (I - S^2) f_i &= (S - S^2) y \\ (I - S)(I + S) f_i &= (I - S) y \\ f_i &= (I + S)^{-1} S y \end{aligned}$$

Thus the residual is $r = y - f_1 - f_2$

$$\begin{aligned} r &= y - 2(I + S)^{-1} S y \\ &= (I - 2(I + S)^{-1} S) y \\ &= (I + S)^{-1}((I + S) - 2S) y \\ &= (I + S)^{-1}(I - S) y \end{aligned}$$

10

CHAPTER 10 - BOOSTING AND ADDITIVE TREES

NOTES ON ADABOOST.M1

Algorithm

1. Initialise $w = \left[\frac{1}{N}, \ldots, \frac{1}{N}\right]$
2. For $m = 1, \ldots M$
   a) Fit G_m with weights w
   b) Compute the error as $err_m = w^T \mathbb{I}(y \neq G_m(x))/w^T \mathbb{1}$
   c) Set $\alpha_m = log((1 - err_m)/err_m)$
   d) set $w \leftarrow w \exp\left[\alpha_m \mathbb{I}(y \neq G_m(x))\right]$
3. Output $G(x) = \sum \alpha_m G_m(x)$

Questions

- How to choose M? Fitting the aggregated model at each M until training error decreases by sufficiently little? What is the model-complexity trade off here. If G has K degrees of freedom say, does boosting give MK degrees of freedom?
- Can AdaBoost get "stuck" alternating between few poorly classified examples?
- What is the natural generalisation to multiple classes / to regression?

AdaBoost as a Forward Stagewise Additive Model

$L(y, f(x)) = \exp(-yf(x))$ We deal with binary classification again $y_i \in \{-1, 1\}$ The goal is to solve:

$$\begin{aligned}(\beta_m, G_m) &= argmin_{\beta,G} \sum_i \exp(-y_i(f_{m-1}(x_i) + \beta G(x_i))) \\ &= argmin_{\beta,G} \sum_i w_i^{(m)} \exp(-y_i \beta G(x_i)))\end{aligned}$$

Where $w_i^{(m)} = \exp(-y_i f_{m-1}(x_i)$ Fix $\beta \geq 0$, then we can rewrite expression to be minimised as (dropping the superscript m for convenience and using vectorised notation):

$$\begin{aligned}
&\exp(-\beta) \sum_{y_i = G(x_i)} w_i^{(m)} + \exp(\beta) \sum_{y_i \neq G(x_i)} w_i^{(m)} \\
&= (\exp(\beta) - \exp(-\beta)) \sum_i w_i^{(m)} \mathbb{I}\left(y_i \neq G(x_i)\right) + \exp(\beta) \sum_i w_i^{(m)}
\end{aligned}$$

Differentiating w.r.t β and setting the result to 0 gives:

$$\begin{aligned}
0 &= \frac{\partial}{\partial \beta} \left(\exp(\beta) - \exp(-\beta)\right) w^T \mathbb{I}(y \neq G(x)) + \exp(-\beta) w^T \mathbb{1} \\
&= \left(\exp(\beta) + \exp(-\beta)\right) w^T \mathbb{I}(y \neq G(x)) + \exp(-\beta) w^T \mathbb{1} \\
\Rightarrow \exp(-\beta) w^T \mathbb{1} &= \left(\exp(\beta) + \exp(-\beta)\right) \frac{w^T \mathbb{I}(y \neq G(x))}{w^T \mathbb{1}} \\
\Rightarrow \exp(-\beta) w^T \mathbb{1} &= \left(\exp(\beta) + \exp(-\beta)\right) err_m \\
\Rightarrow 1 &= \left(\exp(2\beta) + 1\right) err_m \\
\Rightarrow \beta &= \frac{1}{2} \log \frac{err_m - 1}{err_m}
\end{aligned}$$

Then $f_m(x) = f_{m-1}(x) + \beta_m G_m(x)$ So

$$w_i^{(m+1)} = w_i^{(m)} \exp\left(-\beta_m y_i G_m(x_i)\right)$$

Lastly $-y^T G_m(x) = 2\mathbb{I}\left(y \neq G_m(x)\right) - 1$ gives us:

$$w_i^{(m+1)} = w_i^{(m)} \exp\left(-\alpha_m \mathbb{I}\left(y \neq G_m(x)\right)\right) \exp\left(-\beta_m\right)$$

Adjusting w_i as needed, this is equivalent to AdaBoost.

EXCERCISES

10.1 Derive expression (10.12) for the update parameter in AdaBoost

We solved this in our notes above, by differentiating w.r.t. β.

10.2 Prove result (10.16), that is, the minimiser of the population of the AdaBoost criterion, is one-half of the log odds

In other words, find $f^*(x)$ such that:

$$f^*(x) = argmin_{f(x)} \mathbb{E}_Y e^{-Yf(x)}$$

$$
\begin{aligned}
0 &= \frac{\partial}{\partial f}\mathbb{E}_Y\left[e^{-Yf(x)}\right] \\
&= \mathbb{E}_Y\left[-Ye^{-Yf^*(x)}\right] \\
&= \sum_{y\in\{-1,1\}} -ye^{-yf^*(x)}P(Y=y|x) \\
&= -e^{-f^*(x)}P(Y=1|x) + e^{f^*(x)}P(Y=-1|x) \\
&= -P(Y=1|x) + e^{2f^*(x)}P(Y=-1|x) \\
\Rightarrow e^{2f^*(x)} &= \frac{P(Y=1|x)}{P(Y=-1|x)} \\
\Rightarrow f^*(x) &= \frac{1}{2}\log\frac{P(Y=1|x)}{P(Y=-1|x)}
\end{aligned}
$$

We're then done so long as this is truly a minimiser. To see this note that:

$$
\begin{aligned}
\frac{\partial^2}{\partial f^2}\mathbb{E}_Y\left[e^{-Yf(x)}\right] &= \frac{\partial}{\partial f}\mathbb{E}_Y\left[-Ye^{-Yf(x)}\right] \\
&= \mathbb{E}_Y\left[Y^2e^{-Yf(x)}\right] \\
&> 0
\end{aligned}
$$

As $Y \neq 0$.

10.0.1.

10.3 Show that the marginal average (10.47) recovers additive and multiplicative functions (10.50) and (10.51), while the conditional expectation (10.49) does not.

Marginal Average:

$$f_S(X_S) = \mathbb{E}_{X_C}\left[f(X_S, X_C)\right]$$

Conditional Expectation:

$$\tilde{f}_S(X_S) = \mathbb{E}\left[f(X_S, X_C)|X_S\right]$$

a) Additive f

$$f(X) = h_1(X_S) + h_2(X_C)$$

Then:

$$
\begin{aligned}
f_S(X_S) &= h_1(X_S) + \mathbb{E}_{X_C}\left[h_2(X_C)\right] \\
\tilde{f}_S(X_S) &= h_1(X_S) + \mathbb{E}\left[h_2(X_C)|X_S\right]
\end{aligned}
$$

b) Multipicative f

$$f(X) = h_1(X_S)h_2(X_C)$$

Then:

$$f_S(X_S) = h_1(X_S)\mathbb{E}_{X_C}\left[h_2(X_C)\right]$$
$$\tilde{f}_S(X_S) = h_1(X_S)\mathbb{E}\left[h_2(X_C)|X_S\right]$$

Note that $\mathbb{E}\left[h_2(X_C)|X_S\right]$ is a function of X_S, while $\mathbb{E}_{X_C}\left[h_2(X_C)\right]$ is a constant.

In particular f_S recovers the additive and multiplicative functions up to constants, while $\tilde{f}$ only does so if X_C and X_S are independent. Else it recovers a sum (product) of functions of X_S

10.5 Multiclass Exponential Loss. For a K-class classification problem consider the coding $Y = (Y_1, \ldots, Y_K)^T$ with

$$Y_K = \begin{cases} 1 & \textbf{if } G = \mathbf{G}_k \\ -\frac{1}{K-1} & \textbf{if } G \neq \mathbf{G}_k \end{cases}$$

Let $f = (f_1, \ldots, f_K)^T$ with $f^T\mathbb{1} = 0$, and define

$$L(Y, f) = \exp\left(-\frac{1}{K}Y^T f\right)$$

a) Using Lagrange multipliers, derive the population minimiser f^* of $L(Y, f)$, and relate these to the class probabilities.

Let k be such that $G = G_k$ for fixed Y. Then using that $\frac{1}{K} = \frac{1}{K-1} - \frac{1}{K(K-1)}$

$$L(Y, f) = \exp\left(\frac{-1}{K((K-1)}\sum_i f_i - \frac{1}{K-1}f_k\right) = \exp\left(-\frac{1}{K-1}f_k\right)$$

$$h(f, \lambda) = \mathbb{E}\left[L(Y, f)\right] + \lambda f^T\mathbb{1}$$

$$\frac{\partial h}{\partial f_k}|_Y = -\mathbb{P}(G = \mathbf{G}_k|x)\frac{1}{K-1}\exp\left(-\frac{1}{K-1}f_k\right) + \lambda$$
$$\frac{\partial h}{\partial \lambda} = \lambda f^T\mathbb{1}$$

Setting these equal to 0, and using that $\frac{1}{K} = \frac{1}{K-1} - \frac{1}{K(K-1)}$:

$$\lambda = \mathbb{P}(G = \mathbf{G}_k|x)\frac{1}{K-1}\exp\left(-\frac{1}{K-1}f_k(x)\right)$$
$$f_k(x) = -(K-1)\log\frac{(K-1)\lambda}{\mathbb{P}(G = \mathbf{G}_k|x)}$$

We know $\sum_i f_i = 0$, so:

$$\begin{aligned}0 &= -(K-1)\sum_i \log\frac{(K-1)\lambda}{\mathbb{P}(G = \mathbf{G}_i|x)}\\ &= -K(K-1)\log((K-1)\lambda)\\ &\quad + (K-1)\sum_i \log\mathbb{P}(G = \mathbf{G}_i|x)\\ \Rightarrow \log((K-1)\lambda) &= \frac{1}{K}\sum_i \log\mathbb{P}(G = \mathbf{G}_i|x)\\ \Rightarrow \lambda &= \frac{1}{K-1}e^{\frac{1}{K}\sum_i \log\mathbb{P}(G=\mathbf{G}_i|x)}\\ \Rightarrow f_k^*(x) &= \frac{K-1}{K}\sum_i \log\frac{\mathbb{P}(G = \mathbf{G}_k|x)}{\mathbb{P}(G = \mathbf{G}_i|x)}\end{aligned}$$

Note that this is a constant times the sum of the log-ratios of class probabilities - i.e. a function of the relative probability of this class vs all others individually.

An expression for the probabilities would be

$$\mathbb{P}(G = \mathbf{G}_k|x) = \frac{\exp\left(\frac{f_k^*(x)}{K-1}\right)}{\sum_i \exp\left(\frac{f_i^*(x)}{K-1}\right)}$$

b) Show that a multiclass boosting using this loss function leads to a reweighting algorithm similar to AdaBoost, as in Section 10.4

We start with the multiclass exponential loss above, and want to solve:

$$(\beta_m, G_m) = argmin_{\beta,G}\sum_i L(y_i, f_{m-1}(x_i) + \beta G(x_i))$$

Explicitly, this gives:

$$(\beta_m, G_m) = argmin_{\beta,G}\sum_i w_i^{(m)}\exp\left(-\frac{\beta}{K}y_i^T G(x_i)\right)$$

Where G is a classifier coded as Y was above[1], and

$$w_i^{(m)} = \exp\left(-y_i^T f_{m-1}(x_i)\right)$$

1 $G(x_i)_k = \begin{cases} 1 & \text{if } G = \mathbf{G_k} \\ -1/(K-1) & \text{if } G \neq \mathbf{G_k} \end{cases}$

We denote by $\Lambda^{(m)}(\beta, G)$ the term $\sum_i w_i^{(m)} \exp\left(-\frac{\beta}{K} y_i^T G(x_i)\right)$.

Then just as for two classes, we can separate into those with $y_i = G(x_i)$ and those without.

$$\begin{aligned}
\Lambda^{(m)}(\beta, G) &= \sum_i w_i^{(m)} \exp\left(-\frac{\beta}{K} y_i^T G(x_i)\right) \\
&= \sum_{y_i = G(x_i)} w_i^{(m)} \exp\left(-\frac{\beta}{K}\left[1 + \frac{K-1}{(K-1)^2}\right]\right) \\
&+ \sum_{y_i \neq G(x_i)} w_i^{(m)} \exp\left(-\frac{\beta}{K}\left[-\frac{2}{K-1} + \frac{K-2}{(K-1)^2}\right]\right) \\
&= \exp\left(-\frac{1}{K-1}\beta\right) \sum_{y_i = G(x_i)} w_i^{(m)} + \exp\left(\frac{1}{(K-1)^2}\beta\right) \sum_{y_i \neq G(x_i)} w_i^{(m)} \\
&= \exp\left(-\frac{1}{K-1}\beta\right) \sum_i w_i^{(m)} \\
&+ \left[\exp\left(\frac{1}{(K-1)^2}\beta\right) - \exp\left(-\frac{1}{K-1}\beta\right)\right] \sum_i w_i^{(m)} \mathbb{I}_{y_i \neq G(x_i)}
\end{aligned}$$

The only term involving G is the last, hence as for the 2 class case, we have

$$G_m = argmin_G \sum_i w_i^{(m)} \mathbb{I}_{y_i \neq G(x_i)}$$

Differentiating w.r.t β gives and setting it to 0 gives:

$$\begin{aligned}
0 &= \frac{\partial}{\partial \beta} \Lambda^{(m)}(\beta, G) \\
&= -\frac{1}{K-1} \exp\left(-\frac{1}{K-1}\beta\right) \sum_{y_i = G(x_i)} w_i^{(m)} \\
&+ \frac{1}{(K-1)^2} \exp\left(\frac{1}{(K-1)^2}\beta\right) \sum_{y_i \neq G(x_i)} w_i^{(m)} \\
\Rightarrow \exp\left[\frac{K}{(K-1)^2}\beta\right] &= (K-1) \frac{\sum_{y_i = G(x_i)} w_i^{(m)}}{\sum_{y_i \neq G(x_i)} w_i^{(m)}}
\end{aligned}$$

Hence

$$\begin{aligned}
\beta_m &= \frac{(K-1)^2}{K} \log\left[(K-1) \frac{\sum_{y_i = G(x_i)} w_i^{(m)}}{\sum_{y_i \neq G(x_i)} w_i^{(m)}}\right] \\
&= \frac{(K-1)^2}{K}\left[\log\left(\frac{1 - err_m}{err_m}\right) + \log(K-1)\right]
\end{aligned}$$

Note that this agrees with the regular AdaBoost criterion in the special case when $K = 2$.