Introduction:

Linear regression is a simple method in machine learning, used to create a fit of a dataset of which it can produce continuous predictions. This is achieved by assuming that we can model the output values as a linear combination of continuous functions of the input values. For example, we can assume that the observed values in our dataset can be approximated by different polynomial degrees of the input values and find the linear coefficients $\beta$ which minimize the difference between our observed and predicted values. This general idea that we can create a model which approximates our observed data with a minimal difference between observed and predicted values is at the very heart of machine learning.

However, in order to create an optimal model, we require optimal parameters which cannot be known a priori. Thus, we require extensive testing and tuning of parameters to inform our selection. This presents a major obstacle, as we must for example find the optimal polynomial degree $N$ for which our model best approximates the observed values at the same time as we tune the hyperparameter $\lambda$ responsible for an adequate regularisation. In this paper we will find these parameters which lie at an intersection, where all parameters yield the minimal error for our model in unity.

Furthermore, we will compare different models of linear regression, namely Ordinary Least Squares (OLS), Ridge and Lasso regression, and employ resampling methods, namely bootstrap and cross-validation to ensure that we draw conclusive results as these help us reduce the randomness in our parameters based off how we split our data. <source?>

METHOD

BIG THEORY ABOUT OLS -> WHATS DIFFERENT ABOUT RIDGE AND FINNA LASSO

Method

Resampling methods:

Bootstrap

Bootstrap is a resampling method divided into simple steps. First, we reshuffle our dataset with replacement, which will provide us with a new dataset. Note that we will already have split our data into train and test data before we begin the bootstrap, so when we reshuffle, we will only reshuffle the training data. The reshuffled datasets act as our samples.

Secondly, we will perform our linear regression on this new dataset and store the predictions of the test data. Thirdly, we repeat for a given large number of iterations. Since we use all the test predictions to calculate the MSE, bias and variance, we will then according to the central limit theorem <reference?> obtain a normal distribution of these variables. The law of large numbers <reference?> then tells us that the mean sample values of the MSE, bias and variance approach the true value of MSE, bias and variance. This is dependent on our assumption that our data is independently and identically distributed.

These approximations of the true value of MSE, bias and variance will become useful when performing our bias-variance trade-off analysis.

Cross validation

Cross validation is another resampling technique, in which we reshuffle our data and then split them into k-folds, where k specifies the number of folds. One fold is used as our test data, and the rest is used as training data. We run our linear regression, then switch up which fold is used for testing until all folds have been used. As in bootstrapping, we will approach the true values of for example the mean squared error for our model.

Theoretically, because both two resampling techniques use the central limit theorem and the law of large numbers, they should return the same values for the MSE, bias and variance given the same dataset. Though resampling techniques can be computationally expensive, they make up for it by aiding in model evaluation by supplying us with variables obtained from a larger number of samples.

Bias-variance trade-off

A crucial part of model evaluation is ensuring that the model is neither underfitted nor overfitted, which we can determine by observing the relationship between the bias and the variance. Bias is defined as $(Bias[\tilde{y}])^2 = (y-E[\tilde{y}])^2$, the difference between the expected value of our prediction and the observed value. Models with high bias are usually underfitted, whilst models with low bias are the opposite. Variance on the other hand is defined as $var[\tilde{f}] = \frac{1}{n}\sum(\tilde{y\_i}-E[\tilde{y}])^2$, how far away predictions are from the mean prediction. Models with high variance are usually overfitted to the training data, whilst low variance models are the opposite.

A good rule of thumb for finding a good fit is looking at the intercept between bias and variance. This is because the cost function <PROOF task c>. Because the cost function is a sum of the bias and the variance, the lowest error will generally occur when both bias and variance are low. This also fits our understanding of the definitions of bias and variance, as we wish to obtain a model that is neither high in bias (underfitted) or high in variance (overfitted).