Foundations of Machine Learning - Lab 5 Report

Johnny Joyce (jvjj1u19@soton.ac.uk)

November 21, 2019

1 Building the model

Given our initial model, the following changes were implemented:

- Each feature of the input data was normalised to have a mean of 0 and standard deviation of 1, meaning each feature would be prioritised equally. For example, if a particular feature x has a standard deviation of 0.0001 (when all other features have standard deviation 1), then the x feature each data point would be very close to the mean of the feature, making the Euclidean distance between the two points close to 0. Therefore, feature x would contribute very little towards the basis functions, even if certain data points vary relatively wildly. However, if the data is normalised, this is no longer an issue.
- The width of the basis functions was taken to be the mean of the set $\{||x_i - y_i|| \mid (x_i, y_i) \in X\}$ where X is a sample of 50 distinct pairs of data points from the training data. In comparison to simply choosing the distance between two random points, this reduces the possibility that we sample two data points that are either very close to one another (leading to basis functions which do not encompass enough data) or are very far from one another (leading to basis functions that cannot accurately represent smaller changes in data).
- The centres of the basis functions were set to the centers of clusters produced by K-means clustering. Since K-means clustering minimises the sum of squared Euclidean distances from each coordinate to the center of its nearest cluster, this meant that the centres of the basis functions were as close as possible to as many coordinates as possible.
- The data was split into testing data and training data.

The results of this updated model can be seen in Figure 1.

Figure 1: The results of the model after having made all changes. Green coordinates indicate training data and red coordinates indicate testing data.

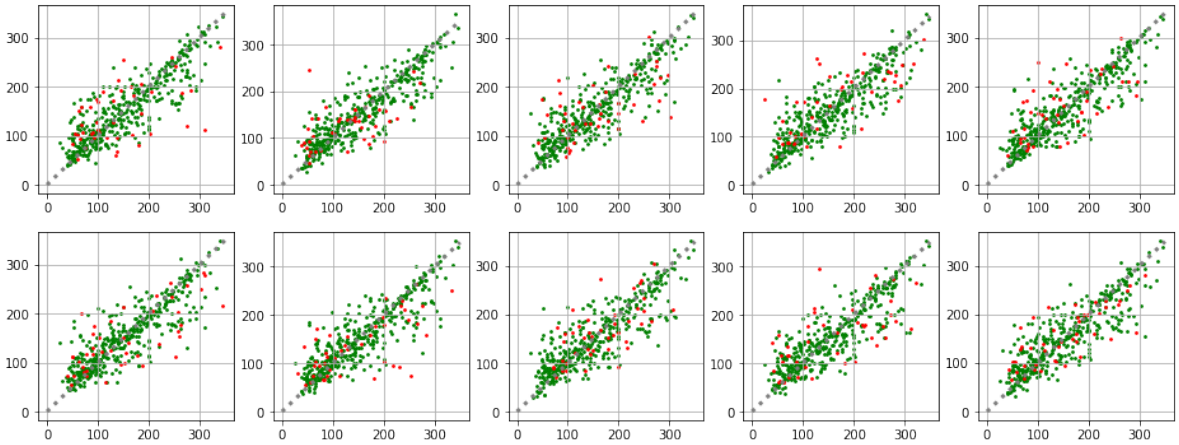


Figure 2: The results of ten-fold cross validation for each iteration. Green coordinates indicate training data and red coordinates indicate testing data.

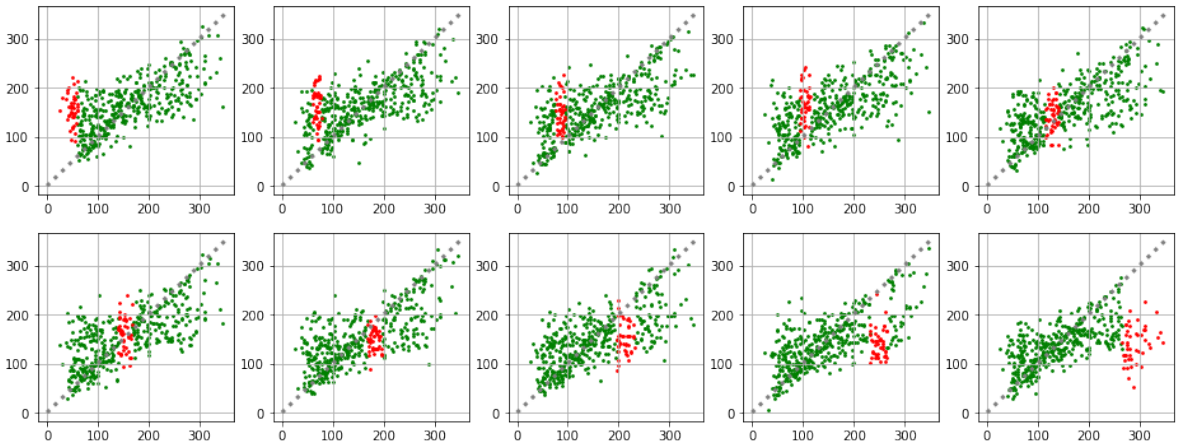


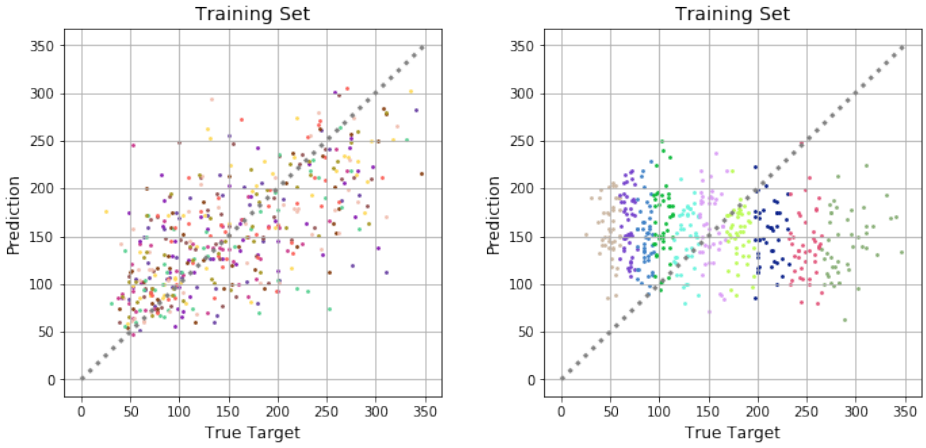
Figure 3: The results of ten-fold cross validation for each iteration after testing data has been sorted. Green coordinates indicate training data and red coordinates indicate testing data.

2 Ten-fold cross validation

We can now implement ten-fold cross validation on our model by setting the testing data as 10% of the total data, then testing the model before moving onto the next 10% as the new testing data, and so on. The results of this can be seen in Figure 2.

We can also sort the data before implementing ten-fold cross validation in order to visually verify that the data has been correctly separated. The results of doing so can be seen in Figure 3.

We now notice that the model's predictions on the training data when it is sorted appears to be less accurate than when it is unsorted. To check the validity of this observation, we can plot the testing data from each iteration on a single plot, as seen in Figure 4. We can see that the sorted predictions are significantly worse than the unsorted predictions, which is most likely due to the fact that the basis functions are centred around the centres of the clusters produced by K-means clustering. When the data is sorted, there are no clusters centred around the testing data, making the model unable to make accurate predictions.



(a) (b)

Figure 4: Results of the testing data used in ten-fold validation compiled into a single graph with (a) unsorted data (b) sorted data. A random colour was used for each iteration.