

## Problem 1 - Cross-Validation

1. The value of  $k$  dictates the number of times we divide the data and by extension the number of cross validation folds/rounds. 1) A  $k$  value of 1 is the same as the validation set approach. 2) Whereas, a value of  $k = n-1$ , where  $n$  is the number of observations in the dataset, is equivalent to LOOCV. 3) A value of 5 or 10 for example is a common middle ground. In 1) we have high bias low variance, in 2) we have low bias high variance (but averaged) and in 3) we have ok bias and variance and it's less expensive to compute.
2. Leverage can be defined as how different/far sample  $i$  is, for a given independent variable, from the expected value of that variable. A high leverage sample may have advantage over the rest, in terms of regression, skewing the fit towards it, influencing/changing the prediction and increasing bias. Removing such a sample would result in a less biased model that better fits the data (i.e. less MSE), provided the other samples are similar in that sense.
3. We have the entire dataset, the responses, the dataset without a sample  $i$  (LOOCV) and the responses without the corresponding response to sample  $i$ :

$$X \in R^{n \times p}, y \in R^n, X_{-i} \in R^{(n-1) \times p}, y_{-i} \in R^{n-1}$$

Now we fit least squares once on the entire dataset:

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

Now what would be our fit had we fit least squares with LOOCV:

$$\hat{\beta}_{-i} = (X_{-i}^T X_{-i})^{-1} X_{-i}^T y_{-i}$$

Now we have the hat matrices for both cases:

$$H = X(X^T X)^{-1} X^T$$

$$H_{-i} = X_{-i}(X_{-i}^T X_{-i})^{-1} X_{-i}^T$$

Our predictions would now look like this:

$$\hat{y} = x_i^T \hat{\beta}$$

$$\hat{y}_{-i} = x_i^T \hat{\beta}_{-i}$$

Since the prediction in least squares is a projection of the response using the hat matrix. It is possible to use hat matrix from the fitted least squares over the

entire dataset to simulate predictions without a given sample by removing its influence when projecting.

$$\begin{aligned}\hat{y}_{-i} &= \sum_{i \neq j} H_{ij} y_j + H_{ii} \hat{y}_{-i} \\ \hat{y}_{-i} &= \sum_{j=1}^n H_{ij} y_j - H_{ii} y_i + H_{ii} \hat{y}_{-i} \\ \hat{y}_{-i} &= \hat{y}_i - H_{ii} y_i + H_{ii} \hat{y}_{-i}\end{aligned}$$

Now our prediction error:

$$\begin{aligned}y_i - \hat{y}_{-i} &= y_i - (\hat{y}_i - H_{ii} y_i + H_{ii} \hat{y}_{-i}) \\ y_i - \hat{y}_{-i} - H_{ii} y_i + H_{ii} \hat{y}_{-i} &= y_i - \hat{y}_i\end{aligned}$$

$$MSE_i = y_i - \hat{y}_{-i} = \frac{y_i - \hat{y}_i}{1 - H_{ii}}$$

Now going back to the original formula for calculating the average MSE in CV:

$$\begin{aligned}CV_{(n)} &= \frac{1}{n} \sum_i^n MSE_i \\ CV_{(n)} &= \frac{1}{n} \sum_i^n \frac{y_i - \hat{y}_i}{1 - H_{ii}}\end{aligned}$$

This proof was heavily inspired by [1].

## References:

1. Linear Regression LOOCV Trick by user: krisrs1128 on notesofstatisticswatcher, <https://notesofstatisticswatcher.wordpress.com/2012/12/18/linear-regression-loocv-trick/>