# HW1: Regression, Cross-Validation, and Regularization

*Vedant Modi*

COMP135: Introduction to Machine Learning

Spring 2025, Tufts University

February 6, 2025

# Problem 1

## 1a Short answer

The weight coefficient values (to 2 decimal places) for each of the $F$ features of the degree = 1 model are as follows

```
 -10.43 : x0
 -18.23 : x1
  -1.15 : x2
   0.58 : x3
where
x0 = horsepower
x1 = weight
x2 = cylinders
x3 = displacement
```

## 1b Short answer

Intuitively, the heavier the engine, the *less* efficient the vehicle is bound to be. This is because the engine needs to expel more force to propel a larger weight. This would result in a negative correlation between weight and the mpg of the vehicle.

On the other hand, a higher engine displacement means that there is more volume of air being moved through the engine. So, one might think that the mpg is higher with a higher engine displacement. However, we also know that a larger engine would have a higher engine displacement. A larger engine necessitates a heavier engine, and thus a diminishing mpg for the vehicle.

In other words, the relationship could go both ways, a higher displacement on a small enough engine would cause a higher mpg. But, a higher displacement on an engine that is too large, would cause a lower mpg. For this reason, the magnitude of the weight is very low.

## 1c Short answer

The magnitudes of the weights in degree 4 are much higher than those in lower degrees (like degree 1, or degree 2). This is likely because of the idea we discussed in lecture, where the degree 4 polynomial will make large adjustments to try to hit all the training points, but perform worse in validation tests.

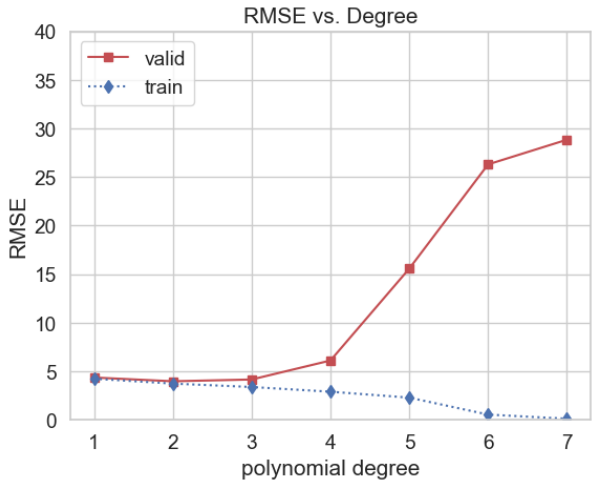


Figure 1: Chosen degree is 2

(i) I have chosen degree 2 since this minimizes the distance between RMSE of training and validation sets. At the same time, the validation error starts to increase at degrees higher than 2. This implies that for models with degree $> 2$, there might be overfitting. That is, the model may find difficulty in generalizing on "never-seen-before" data.

On the other hand, I selected a degree 2 polynomial over a degree 1 polynomial as minimized the effect of "underfitting" as well. That is, there are enough features that are being added such that both the training and validation data report lower errors.

(ii) Overfitting can be seen in higher degrees of the polynomial. (degrees 5, 6, 7)

We can tell that there is overfitting because the large differential between validation error and training error. We can tell this is overfitting because although training error is low (i.e. the model can predict the response for the training examples with great performance), we can see the validation error is very high. A very high validation error means that the model does not generalize well. This means that on new data, that the model is not trained on, we see worse performance (higher error).

## 1d Short answer

Training error has a lower rate of increase going from degree 6 to degree 7, as compared to degree 5 to degree 6. This can be thought of as the model "overfitting", which is seen at higher degrees of regression, as discussed in day03 and day04.

## 1e Short answer

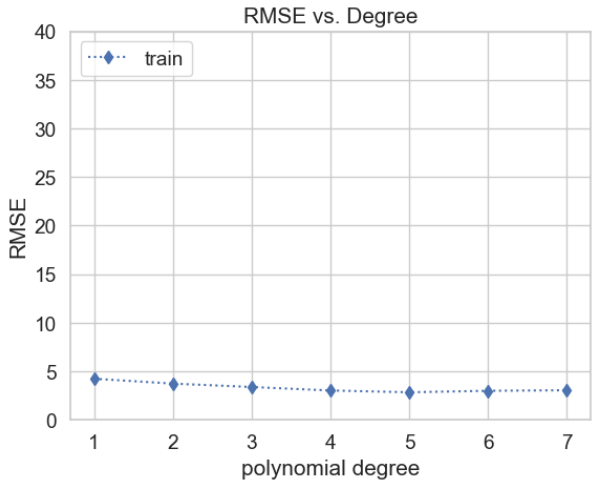


Figure 1.1: Lower training error compared to Figure 1

Train error increases slightly at higher degrees without the prescence of `MinMaxScaling`. This is because with `MinMaxScaling`, we produce weights of lesser magnitude, relative to without `MinMaxScaling`. We've seen before that with higher magnitude weights, we are prone to overfitting the model, as the coefficients will be very close to the training data, and perform very well on the training data. This leads to a lower error in the training data, which is what we see in Figure 1.1.

It's useful to scale raw input features to be within the range $[0, 1]$ as it brings the magnitudes of weights down. This way, we avoid overfitting (at higher degrees) as it occurred in the aforementioned example.

# Problem 2

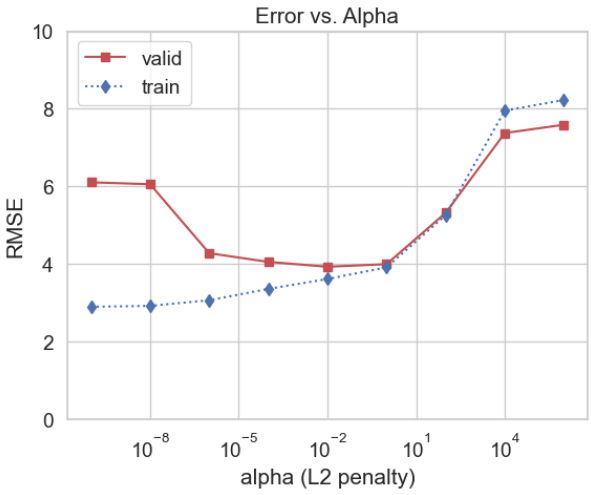


Figure 2: Error vs. Alpha

(i) As $\alpha$ increases, we start seeing a decrease in validation error. Then, at some minima, the validation error starts to increase again. On the other hand, training error is always increasing with higher $\alpha$.

(ii) We should choose the $\alpha$ that leads to the least validation error to perform best on new data. This is because the model has not been fitted with this data, so does not know the responses of these examples. So, we get a more true sense of how the model will perform out in the "real world". So, I would choose $\alpha = 1 \times 10^{-2}$.

## 2a Short answer

In the degree-4 model of Problem 1, the magnitudes of the weights are much larger than the chosen degree-4 model in this problem. This is likely because the L2 regression which we are now using has a term that penalizes large terms. So, larger magnitudes will now be lesser than those derived from a degree-4 regression without any penalization.

## 2b Short answer

To minimize the formula given, we would choose an $\alpha$ that is small as possible (i.e. $\alpha = 0$) since we want to minimize the whole expression, and any $\alpha > 0$ would increase the value of the expression.

For an $\alpha = 0$, we recover the unpenalized regression, which would lead to the same high

degree model that we saw overfitting in. With overfitting, we have a tendency to not generalize new data well, which would eliminate the purpose of penalties.

# Problem 3

| | name | hypers | testRMSE |
|---|---|---|---|
| 0 | predict mean of train ys | | 8.231074 |
| 1 | LR deg=best-on-val | deg: 2 | 3.991503 |
| 2 | ridgeLR deg=4 alph=best-on-val | alph: 1e-06 | 3.877668 |
| 3 | ridgeLR deg=best-on-CV alph=best-on-CV | deg: 7 alph: 0.1 | 3.816827 |

Table 3: Various RMSE's from regressors

(i) Problem 3's model sees better performance than Problem 2's model since we were able to select between the best degree and $\alpha$ in Problem 3. In Problem 2, we only selected the best $\alpha$, and fixed a degree, 4. Note that in Problem 3, we chose a higher degree, 7. Therefore, we likely saw some underfitting occur in Problem 2's model, and Problem 3 improved that by increasing model complexity. At the same time however, there might have been some decreased generalization. For this reason, we included an $\alpha$ parameter to decrease the magnitudes of the weights. This improves the generalization of the model in Problem 3, outperforming Problem 2's model.

We also have used cross validation when determining the model in Problem 3. This improves generalization as well, since we train using different sets of data, and reduce bias derived from using a specific set of training data. That is, all data gets a chance to be trained on, and we choose the set of data that has the best performance on the validation set. This way, we can observe more data to train the model, and improving generalization. For this reason, Problem 3's model also performs better than Problem 2.

(ii) We can't just select a degree and an $\alpha$ to minimize test set error, since we could potentially underfit the model, and actually end up hurting the generalization. We need to also verify that the training error is low enough, and not decreasing with an increased complexity.