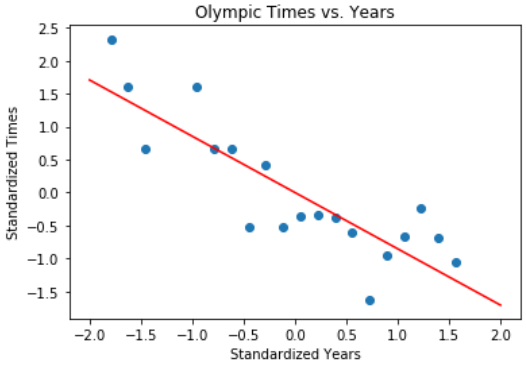
HOMEWORK 1B: LINEAR REGRESSION

Garrett Robins, with Lucas Mendicino and Jonah Danzinger

February 26, 2020

1 Implementing a Regression Solver from Scratch

b. Below is the figure that I plotted. When I compared my coefficients with the book's, I found that they were the same bar a linear transformation.

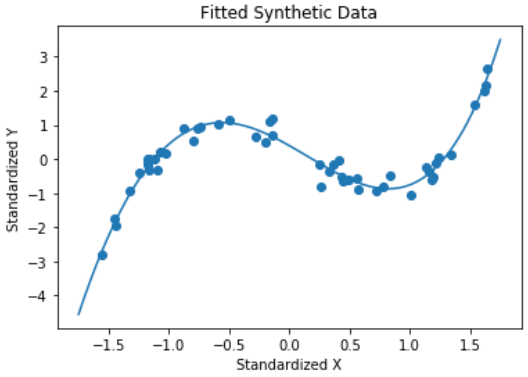


c. When I use the model to predict the 2012 and 2016 times, it does fairly well. It gives a prediction of 10.60 for 2012 and 10.54 for 2016. The actual times were 10.75 and 10.71 leading to a prediction error of .**0514**

...

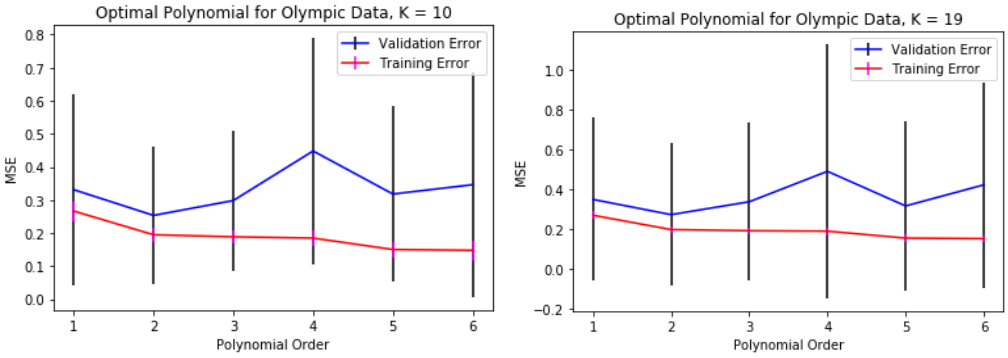
g. See below:

h. When we make lambda bigger on a cubic, the result is just making the prediction line flatter. That is, it more closely resembles a single-predictor line.



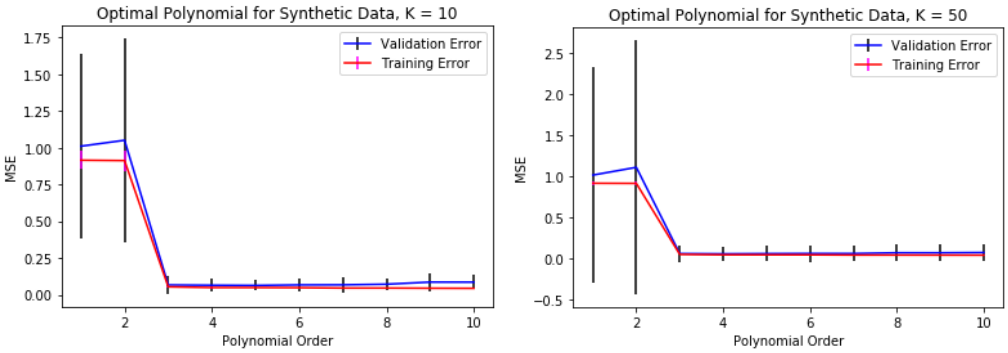
2 Cross-Validation With Regression

d. Below are both graphs for the Olympic dataset. With the 10-fold validation, we can see that the optimal choice for the Olympic data is a second order polynomial. Similarly, with the "leave-one-out" cross-validation, we see that the optimal choice of polynomial is a second-order one.



On the next page we can see both graphs for the synthetic data. Using the 10-fold cross-validation graph we can see that the optimal choice of polynomial is between 3rd and 5th order. Similarly, if we look at the "leave-one-out" cross-validation graph, we see that the optimal choice of polynomial is ambiguous. Likely the best model is between a 3rd and 7th degree polynomial.

e. When we use the 2012 and 2016 actual times as a true test set, we're testing whether our choice of second order polynomial is correct. When we test every order polynomial up to 10, we see that the generalization error for the 2012 data is minimized for a 6th order polynomial, which disagrees with what we supposed. However, when we do the same thing



for 2016 we get that a 2nd order polynomial is optimal, which supports our prior conclusion.

f. Using a grid search, it seems that the 10-fold cross-validation MSE is minimized for a first order polynomial with a ridge coefficient of 2. That is, the order is 2 and $\lambda = 2$. When we check whether this does better than the minimizing OLS orders, we find that this model performs better for both 2012 and 2016. (See code for proof.)

3 Product Rule of Matrix Differentiation

(a) Proof: Let $a = f(\mathbf{v})^T g(\mathbf{v})$.

We have that

$$a = \sum_{i=1}^{m} \sum_{j=1}^{m} (f(v_i) g(v_j))$$

Therefore by the chain rule of matrix differentiation:

$$\frac{\partial a}{\partial v_n} = \sum_{i=1}^{m} f(v_i) \frac{\partial g_n}{\partial v_n} + \sum_{i=1}^{m} g(v_i) \frac{\partial f_n}{\partial v_n}$$

$$\iff \frac{\partial a}{\partial v_n} = f(\mathbf{v})^T \frac{\partial g}{\partial \mathbf{v}} + g(\mathbf{v})^T \frac{\partial f}{\partial \mathbf{v}}$$

(b) Proof: Let $a = \mathbf{v}^T A \mathbf{v}$. We have that $g(\mathbf{v}) = A\mathbf{v}$ and $f(\mathbf{v}) = \mathbf{v}$. Notice that $\frac{dg}{d\mathbf{v}} = A$ and $\frac{df}{d\mathbf{v}} = I$.
So from the fact above:

$$\begin{aligned}
&= \frac{d}{d\mathbf{v}} \mathbf{v}^T A \mathbf{v} \\
&= f(\mathbf{v})^T \frac{dg}{d\mathbf{v}} + g(\mathbf{v})^T \frac{df}{d\mathbf{v}} \\
&= \mathbf{v}^T A + \mathbf{v}^T A I \\
&= 2\mathbf{v}^T A
\end{aligned}$$

4 Deriving Weighted Least Squares Regression Fit

- Find an expression for the estimated weight vector that minimizes the generalization of the OLS loss function with weights applied to each observation
- Proof:

$$\mathcal{L} = \frac{1}{N} (\mathbf{t} - \mathbf{X}\mathbf{w})^T A (\mathbf{t} - \mathbf{X}\mathbf{w})$$

$$\mathcal{L} = \frac{1}{N} (\mathbf{t}^T \mathbf{A}\mathbf{t} - \mathbf{t}^T \mathbf{A}\mathbf{X}\mathbf{w} - (\mathbf{X}\mathbf{w})^T \mathbf{A}\mathbf{t} + (\mathbf{X}\mathbf{w})^T \mathbf{A}\mathbf{X}\mathbf{w})$$

$$\mathcal{L} = \frac{1}{N} (\mathbf{t}^T \mathbf{A}\mathbf{t} - \mathbf{t}^T \mathbf{A}\mathbf{t}\mathbf{w} - \mathbf{w}^T \mathbf{X}^T \mathbf{A}\mathbf{t} + \mathbf{w}^T \mathbf{X}^T \mathbf{A}\mathbf{X}\mathbf{w})$$

$$\mathcal{L} = \frac{1}{N} (\mathbf{t}^T \mathbf{A}\mathbf{t} - 2\mathbf{t}^T \mathbf{A}\mathbf{X}\mathbf{w} + \mathbf{w}^T \mathbf{X}^T \mathbf{A}\mathbf{X}\mathbf{w})$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \frac{1}{N}(-2\mathbf{t}^T\mathbf{A}\mathbf{X} - 2\mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{X})$$

$$\nabla_{\mathcal{L}} = 2(\mathbf{X}^T\mathbf{A}\mathbf{X})\mathbf{w} - 2\mathbf{X}^T\mathbf{A}\mathbf{t} = 0$$

$$\iff \hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{t}$$