Name: Nattapat Juthaprachakul, Student ID: 301350117

CMPT-726 Machine Learning: Assignment 1

1. Probabilistic Modeling:

1.1. What is the parameter that explains the behaviour of the die in this case (in analogy to the µ for the coin)?
ANS: Parameter µ here explains the probability of getting each side of six-sided dice.

1.2. What is the value of the parameter for a fair die (equal probability of rolling any number)?
ANS: Since there are six possible outcomes for a normal fair dice. The probability of landing each side of dice is equal to 0.1666667 (1/6).

1.3. What is the value of the parameter for a die that always rolls a 2?
ANS: The value of parameter for a dice that always rolls 2 is equal to 1 (100 percent) for parameter µ of getting 2 and 0 (0 percent) for parameter µ of getting other numbers except 2 (1,3,4,5,6).

1.4. Specify the domain of the parameter – which settings of the parameter are valid.
ANS: The probability of getting any 1-to-6 number must be positive and the sum of all of the parameter µ of getting each 1-to-6 number must be equal to 1 (100 percent).

2. Weighted Squared Error

ANS:

(2) Weight Squared Error.

$$E_D(w) = \frac{1}{2}\sum_{n=1}^{N} \alpha_n \{t_n - w^T\phi(x_n)\}^2 \quad \text{w/ } \alpha_n > 0$$

$$\frac{\partial E_D(w)}{\partial w} = \frac{1}{2}\sum_{n=1}^{N} \alpha_n (t_n - w^T\phi(x_n)]\, 2(-\phi(x_n)^T]$$

$$0 = -\left[\sum_{n=1}^{N} \alpha_n (t_n - w^T\phi(x_n))\phi(x_n)^T\right]$$

$$0 = -\left[\sum_{n=1}^{N} \alpha_n t_n \phi(x_n)^T - w^T\sum_{n=1}^{N} \alpha_n \phi(x_n)\phi(x_n)^T\right]$$

$$0 = -\sum_{n=1}^{N} \underbrace{[\sqrt{\alpha_n}\, t_n]}_{t'} \underbrace{[\sqrt{\alpha_n}\, \phi^T(x_n)]}_{\phi'^T(x_n)} + w^T\sum_{n=1}^{N} \underbrace{[\sqrt{\alpha_n}\, \phi(x_n)]}_{\phi'(x_n)} \underbrace{[\sqrt{\alpha_n}\, \phi^T(x_n)]}_{\phi'^T(x_n)}$$

$$w^T\sum_{n=1}^{N} \phi'(x_n)\phi'^T(x_n) = \sum_{n=1}^{N} t'\phi'^T(x_n)$$

$$[\Phi'\Phi'^T]^T w = \Phi'^T t'$$

$$[\Phi'^T\Phi'] w = \Phi'^T t'$$

$$w = [\Phi'^T\Phi']^{-1}\Phi'^T t'$$

3. Training vs. Test Error

3.1. Suppose we perform unregularized regression on a dataset. Is the validation error always higher than the training error? Explain

ANS: Yes, in theory the validation error is always higher than the training error because the goal of machine learning is to be able generalize and predict unknown new data from the model that we already trained/learnt from known data. However, there is some exception that makes validation error is lower than the training error such as many complex cases in training set and many easy cases in validation dataset.

3.2. Suppose we perform unregularized regression on a dataset. Is the training error with a degree 10 polynomial always lower than or equal to that using a degree 9 polynomial? Explain.

ANS: Yes, as the degree polynomial increases, the model becomes more complex which in turn results in better performance in the training dataset (lower training error because the more complex model becomes more flexible to fit or capture more data.)

3.3. Suppose we perform both regularized and unregularized regression on a dataset. Is the testing error with a degree 20 polynomial always lower using regularized regression compared to unregularized regression? Explain.

ANS: The model with a degree polynomial of 20 is very complex model. Under unregularized situation, the model will have very low training error but very high test error because the very complex model will overfit to training data and fail to generalize to new data (test data). On the other hand, the regularized model may have higher training error than the unregularized model but it will definitely have lower test error as the regularization will put penalty on over-complex model which is very good for our algorithm since the main goal of machine learning is to find the model that can generalize well with new data.

4. Basis Function Dependent Regularization

ANS:

(4) Basis function w/ Regularization

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}\Big(\underset{1\times 1}{t_n} - \underset{1\times d}{w^T}\,\underset{d\times 1}{\phi(x_n)}\Big)^2 + \underset{1}{\lambda}\,\underset{1\times d}{w^T}\,\underset{d\times 1}{w} \quad \rightarrow 1\times 1$$

$$\frac{\partial E(w)}{\partial w} = -\frac{2}{2}\sum_{n=1}^{N}\big(t_n - w^T\phi(x_n)\big]\phi^T(x_n) + \lambda w^T = 0$$

$$0 = -\left[\sum_{n=1}^{N} t_n\phi^T(x_n) - w^T\sum_{n=1}^{N}\phi(x_n)\phi^T(x_n)\right] + \lambda w^T$$

$$0 = -\sum_{n=1}^{N} t_n\phi^T(x_n) + w^T\sum_{n=1}^{N}\phi(x_n)\phi^T(x_n)\Big] + \lambda w^T$$

$$\sum_{n=1}^{N} t_n\phi^T(x_n) = w^T\left[\sum_{n=1}^{N}\phi(x_n)\phi^T(x_n) + \lambda I\right]$$

$$\Phi^T t = [\Phi^T\Phi + \lambda I]\,w$$

$$\underset{d\times 1}{w} = \underbrace{[\underset{d\times n}{\Phi^T}\,\underset{n\times d}{\Phi} + \underset{d\times d}{\lambda I}]}_{d\times d}\ \underbrace{\underset{d\times n}{\Phi^T}\,\underset{n\times 1}{t}}_{d\times 1}$$

5. Regression

5.1.) 5.1.1. Which country had the highest child mortality rate in 1990? What was the rate?

ANS: Niger

5.1.2. Which country had the highest child mortality rate in 2011? What was the rate?

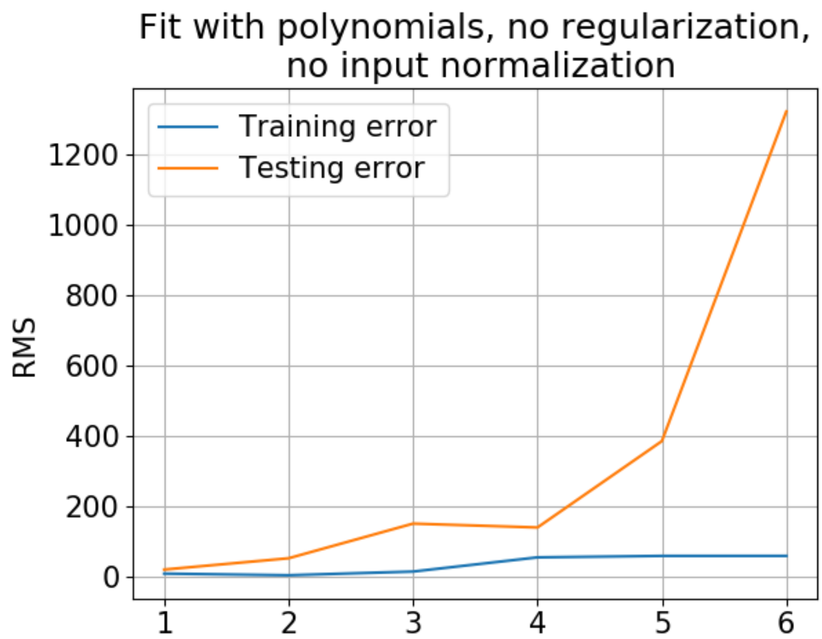
ANS: Sierra Leon

5.1.3. Some countries are missing some features (see original .xlsx/.csv spreadsheet). How is this handled in the function assignment1.load unicef data()?

ANS: Use mean computed along the column(feature) of that missing value.

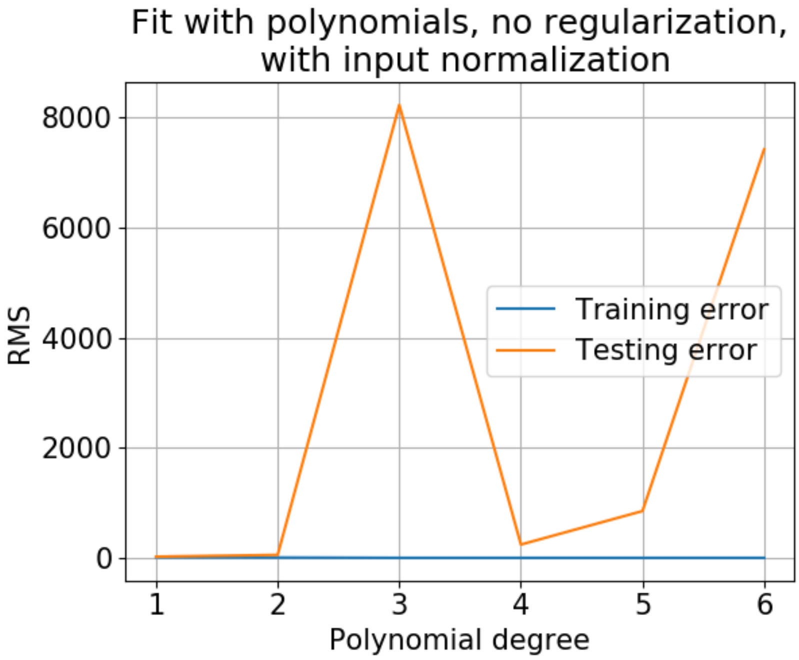
(Numpy.nanmean function)

5.2.) 5.2.1. **ANS:** The training error is below testing error in every degree polynomial. The testing error increases substantially after degree 4 and rises sharply after degree 5. Note that the RMS error is still in range of 0 to 1200.



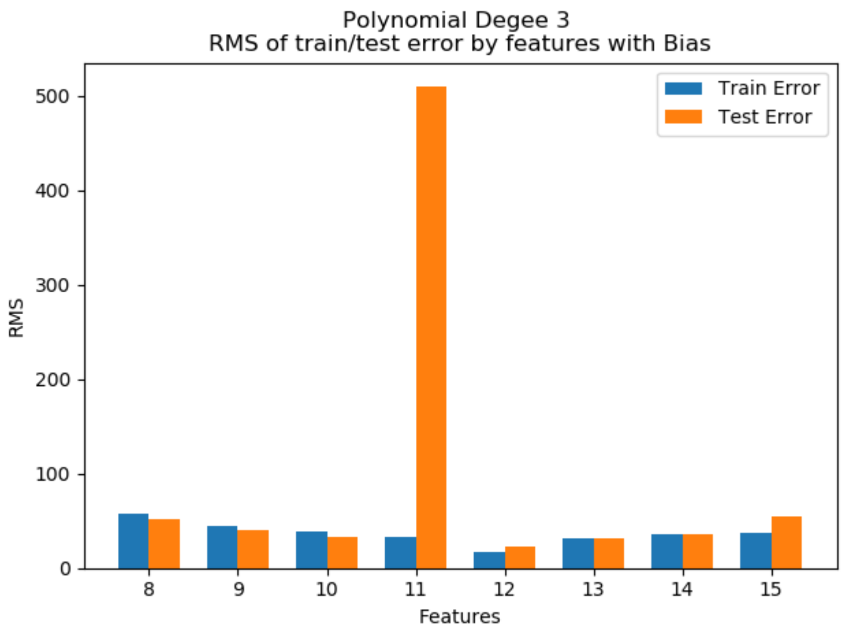
5.2.1) Figure1: Polynomial Regression without feature normalization.

In the Figure 2, we can see that as we normalize input feature the testing error is very different from Figure 1. The testing error is peaked at polynomial degree 3 and degree 6 at about 7000 to 8000.



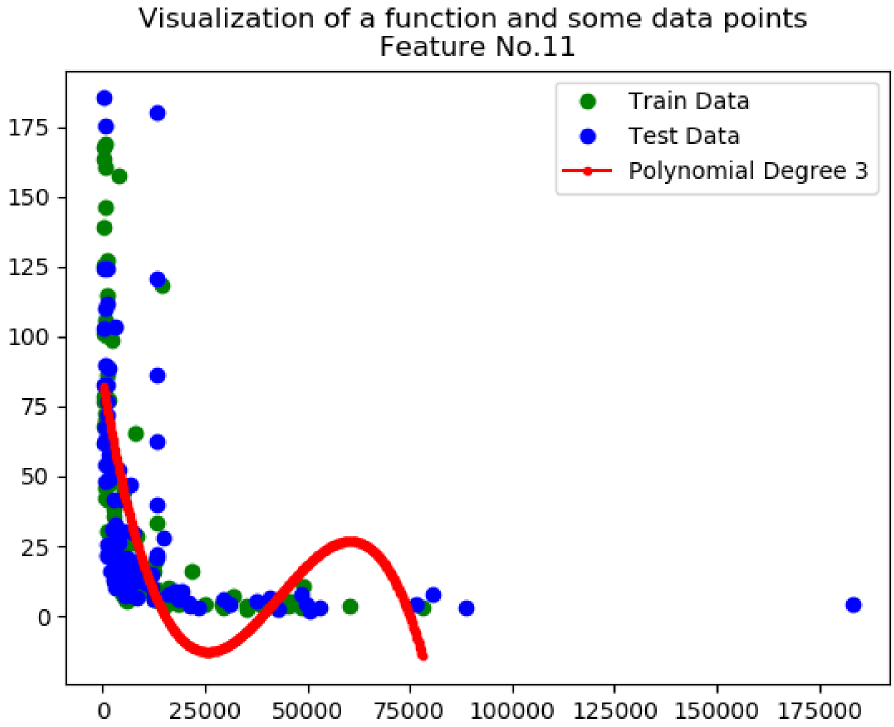
5.2.1) Figure2: Polynomial Regression with feature normalization.

5.2.2. **ANS:**

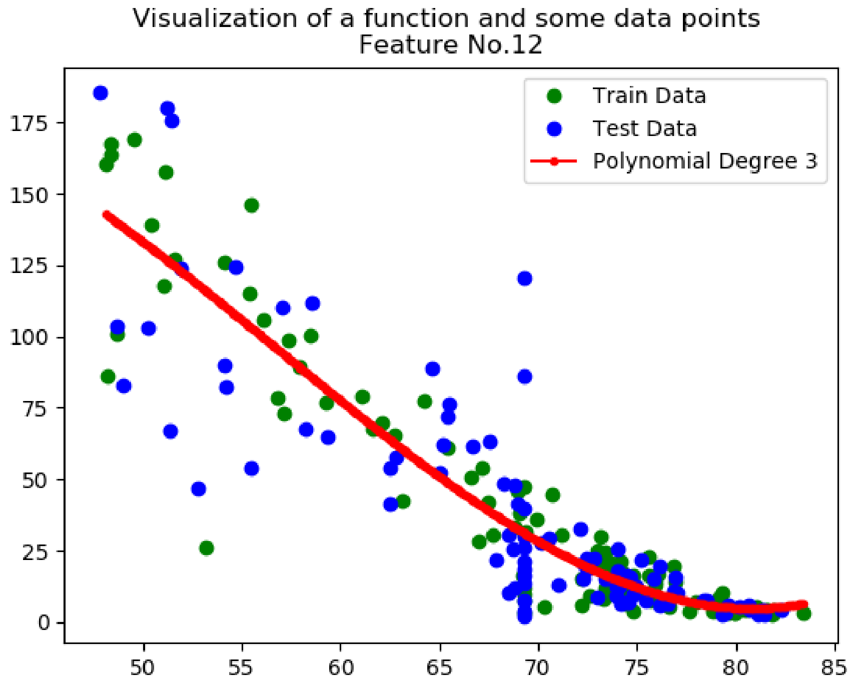


5.2.2) Figure1: Polynomial Regression Degree 3 with Bias

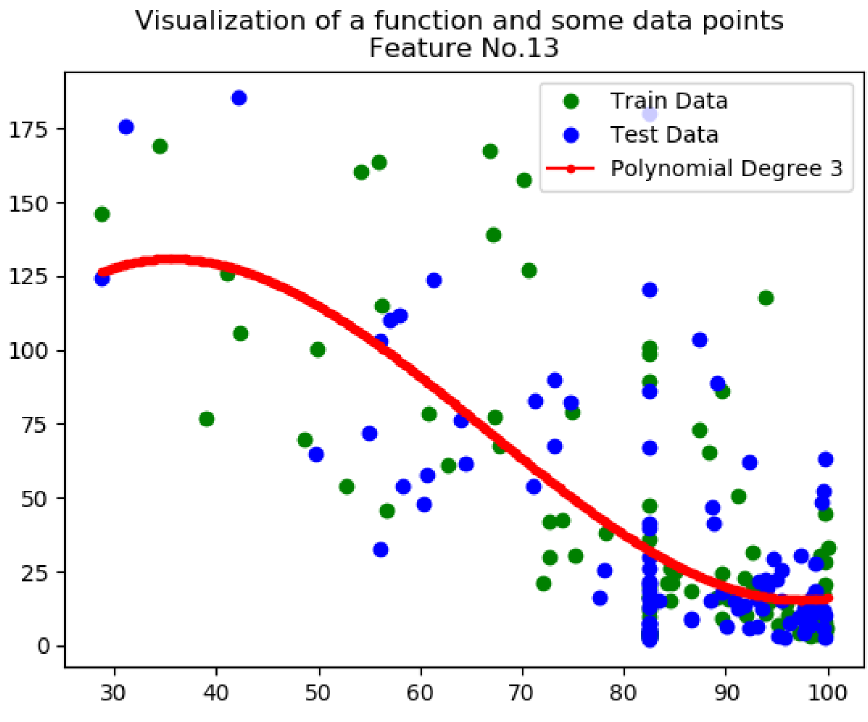
5.2.2) Figure2: Polynomial Regression Degree 3 without Bias



5.2.2) Figure3: Polynomial Regression Degree 3 with Bias on Feature 11 (GNI)

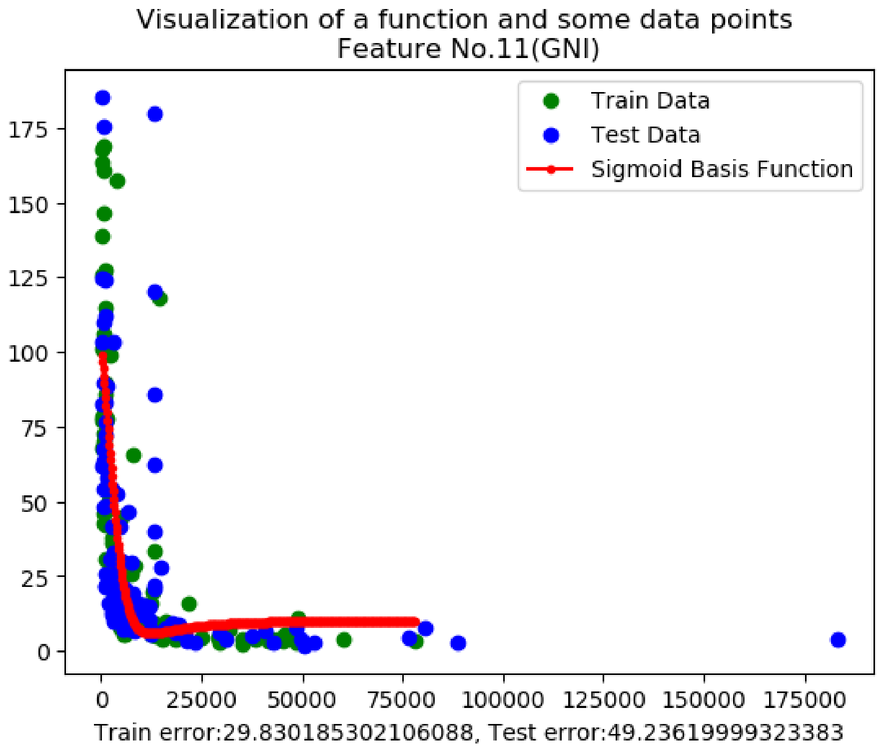


5.2.2) Figure4: Polynomial Regression Degree 3 with Bias on Feature 12 (Life Expectancy)



5.2.2) Figure5: Polynomial Regression Degree 3 with Bias on Feature 13 (Literacy)

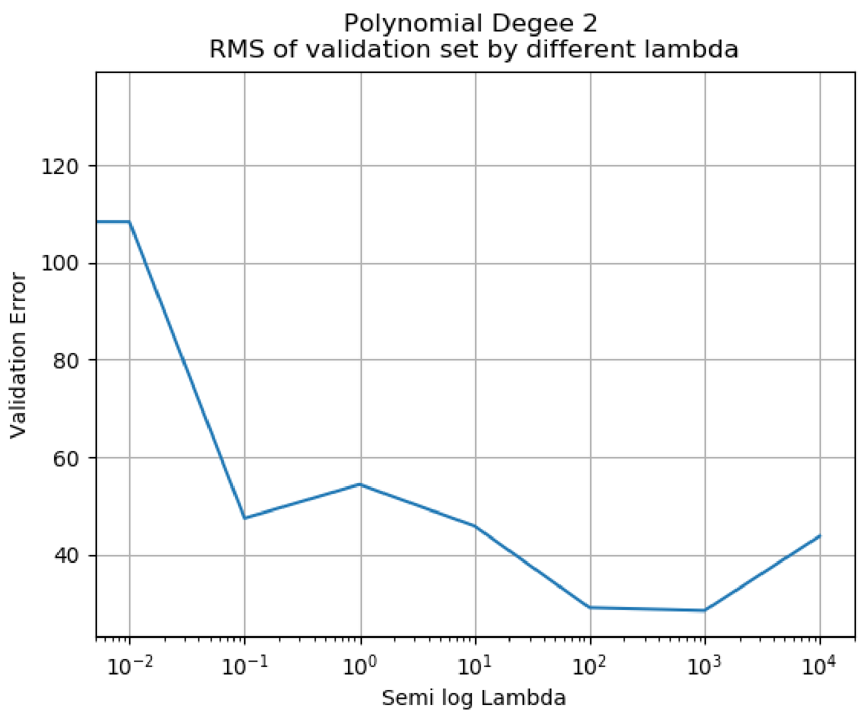
5.3.) **ANS:**



5.3) Figure1: Sigmoid Basis Function with Bias on Feature 11 (GNI)

Training error: 29.830 and Testing error: 49.236

5.4.) **ANS:**



5.4) Figure1: Regularized Polynomial Regression degree 2 with Bias

Lambda 100 and 1000 have the lowest validation error. Therefore, these are the lambda that I would choose.