Cara Evangeline Chandra Mohan
190539421

Data Mining Lab Assignment - 1

1. Which polynomial has the lowest train MSE? Which one has the lowest test MSE?

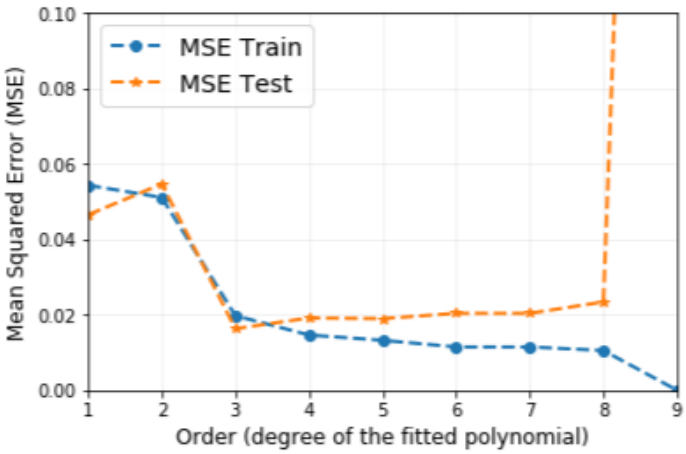

Figure - 1

Degree 9 polynomial has lowest train MSE

$-431.4\ x^9 + 17.67\ x^8 + 792.7\ x^7 - 28.15\ x^6 - 478.4\ x^5 + 9.386\ x^4 + 105.3\ x^3 + 1.091\ x^2 - 5.306\ x + 0.7414$

Degree 3 polynomial has lowest test MSE

$1.326\ x^3 - 0.09076\ x^2 - 0.1449\ x + 1.004$

2. What trend do you observe when you analyse the dependence of train and test MSE on the polynomial order? First describe the observed trends, and then explain them.

As the degree of polynomial increases, the MSE of test dataset is substantially increasing than MSE of train dataset from polynomial of degree 4. At degree 1 and 3 the MSE of test dataset is lesser than MSE of train dataset.

As the polynomial degree increases, the algorithm memorizes the value of train dataset and tries to make a perfect fit with train data which we call it as overfitting and when the test dataset is applied to the obtained algorithm (ie.polynomial equation) the possibility of getting a value closer to true value is very less and hence leads to more MSE of test dataset. For degree 2 and 3 the MSE of test dataset is closer to the MSE of train dataset since the algorithm does not memorize the value of training dataset

Cara Evangeline Chandra Mohan
190539421

3. Identify the models that are suffering from under-fitting and the ones suffering from over-fitting. Justify your choice based on your observations and use the theory that we have learnt to explain it.

Degree 1 Polynomial suffers from underfitting since it does not fit the model leading to high MSE of train and test data. Degree 2 suffers from the same.

From Degree 4 polynomial the algorithm tries to fit the data eventually leading to overfitting in case of degree 8 and degree 9 polynomial.

4.Which model would you pick as the best one amongst these 9 models? What are the parameter(s) and hyper-parameter(s) of your chosen model?

Degree 3 polynomial is the best model amongst the 9 models obtained since it does not overfit or underfit the data and the MSE of test data is the lowest and comparatively same to the MSE of train dataset.
Degree 3 Polynomial: $1.326\ x^3 - 0.09076\ x^2 - 0.1449\ x + 1.004$
Parameters: $w_3 = 1.326$, $w_2 = -0.09076$, $w_1 = -0.1449$, $w_0 = 1.004$
Hyperparameter: Degree = 3

5. Focus on order=9. Describe AND explain the trend of each of the metrics below with respect to increasing values of λ (that is, first describe what the effect of increasing λ from zero upward is it on the parameter in question and then explain briefly and clearly the reasons behind it)

(a) TRAIN MSE
(b) $\vec{W}^T.\vec{W}$
(c) TEST MSE

(a) As the value of λ is increasing, so does the value of MSE of train dataset because the training model not only considers mean square error but also a regularization factor which avoids overfitting of training dataset.

(b) As λ increases, the square of weight decreases so as to decrease Error of the model as can be seen in the below given formula (ie. λ is inversely proportional to $\vec{w}^T.\vec{w}$)

$$E_{MSER} = \frac{1}{N}\sum_{i=1}^{N} e_i^2 + \lambda w^T w$$

Cara Evangeline Chandra Mohan
190539421

(c) As λ increases from 0 to 0.01 the MSE of test data decreases and from 0.1 to 10 λ value the MSE of test data increases. For an initial rise of λ value the model avoids overfitting of training data and hence gives lesser MSE of test data, later as λ increases regularization factor increases leading to lesser concentration of MSE in MSER and hence increasing MSE of test data

i.) λ = 0 , w'*w = 9633512041 Regularization term = 0
ii.) λ = 0.001, w'*w = 6.6963 Regularization term = 0.00669
iii.) λ = 0.01, w'*w = 2.6850 Regularization term = 0.0268
iv.) λ = 0.1, w'*w = 1.4047 Regularization term = 0 .14047
v.) λ = 1, w'*w = 1.2026 Regularization term = 1.2026
vi.) λ = 10, w'*w = 0.9865 Regularization term = 9.865

6. Now suppose that instead of 10 training instances, we had access to 100 train instances. Run the following script and inspect the change in the test error. Describe and explain the effect of having more training data on the test error (test MSE) and over-fitting.

	Sample size = 10				Sample size = 100			
	1	**2**	**3**	**9**	**1**	**2**	**3**	**9**
λ = 0	0.0298	0.1513	0.0114	5988874.21	0.0324	0.0335	0.0091	0.0090
λ = 0.001	0.0298	0.1508	0.0115	0.0813	0.0324	0.0335	0.0091	0.0091
λ = 0.01	0.0299	0.1469	0.0135	0.0160	0.0324	0.0335	0.0091	0.0092
λ = 0.1	0.0311	0.1194	0.0326	0.0319	0.0323	0.0334	0.0096	0.0099
λ = 1	0.0467	0.0776	0.0425	0.0473	0.0317	0.0326	0.0130	0.0119
λ = 10	0.1190	0.1249	0.1076	0.1011	0.0316	0.0319	0.0204	0.0152

The above tabular form shows the comparison of values of MSE of test dataset for sample size 10 and 100. As shown for sample size 100, the test error is smaller in most cases for different values of λ since the model can't overfit the training data.

7. Which is the correct way to complete the script and calculate the variable SSE? Explain what each of the three options to calculate SSE would do and justify your choice. What is the resulting train and test errors?

SSE = np.sum(np.power(yvalset - np.matmul(phival, w_map), 2))
Adding the above line to the code will calculate SSE (Sum of Squared error) and finds the best λ over the validation set. For λ = 0.1 the model proves to be the best with least MSE of validation dataset.

SSE = np.sum(np.power(ytrainset - np.matmul(phitrain, w_map), 2)) will calculate the sum of

squared error of train dataset
SSE = np.sum(np.power(yTest - np.matmul(phitest, w_map), 2)) will calculate the sum of squared error of test dataset

Calculating the SSE for validation set is preferred since the test dataset is not given to the analyst to test the data and the input dataset is divided into two parts and used as train dataset and validation dataset. The model is obtained without overfitting of data. So the parameters obtained by lowering the MSE of validation dataset would prove to be the best to apply on test dataset.

Resulting MSE of test and train dataset:

```
Lambda: 0.10000, CV SSE: 0.13786. * New best
Train MSE =   0.0112
Test MSE =   0.0099
Validation MSE =   0.0055
```

8. Compare the training, validation and the test errors in both the linear model and the M5P model. First, explain the differences between the numerical values of each error separately for the linear model and for the M5P model. Then, bearing in mind that the M5P model is more complex than the linear model, explain why the numerical values of the errors seem to behave differently for the linear model and for the M5P model.

Linear Regression Model:

Train Root Mean Square error = 7357.4334
Validation Root Mean Square error = 8044.4336
Test Root Mean Square error = 8062.8269

M5P Classifier:

Train Root Mean Square error = 5776.4447
Validation Root Mean Square error = 7161.2058
Test Root Mean Square error = 7416.4842

The corresponding Root Mean Square errors of Linear regression model is higher than M5P classifier because Linear model tries to fit the data with order 1 equation which is not efficient resulting in huge RMS errors.

Cara Evangeline Chandra Mohan
190539421

1.) We use training dataset to compute the model and from the model obtained we apply the equation again to the train dataset and compute the train error.
2.) In 2- cross folding we use 50% of dataset as train dataset and the other 50% as validation dataset and vice-versa and compute the validation error by computing the average of the errors.
3.) 50% is test dataset and 50% is train dataset and the resulting test error is obtained from the learnt model.

In M5P classifier the tree is constructed based on the entropy of the attributes which gives a more complex model whereas in linear regression a single dependent variable is dependent on other attributes and the model is simple. Hence M5P model is better than linear regression and has lesser RMS error comparatively.