Amin Fadaee

Machine Learning HW1

Python 3.5, Libraries: Numpy, Pandas, Matplotlib, Documentation: HTML, Bootstrap 4

Problem 1

- **Supervised Learning**: In supervised learning we work with *labeled data* that acts as a teacher and adds **supervision** to the system. Generally in the context of supervised learning we are trying to predict the label for a future unknown data.
- **Semi-Supervised Learning**: This paradigm deals with the amalgam of supervised and unsupervised learning. Generally there will be a small amount of labeled data with a large amount of unlabeled data.
- **Unsupervised Learning**: These models of learning incorporate *label-less* data and try to find some patterns in them, one of the prominent examples of these methods are clustering algorithms.
- **Reinforced Learning**: Model that starts with a basic incipient knowledge and is rewarded each time it acts correctly. In a sense these systems resemble a child that is learning the interactions with physical world.
- **Online Learning**: A class of learning algorithms that learn the model on a continuous flow of data

Problem 2

In order to detect the possible overfitting of the system, one simple way is to compare the training error with the test error. If the training error looks promising but the test error is high then we can assume that the model has overfit to the data, However, we should be wary of one potential rookie mistake! **We shouldn't rebuild our model and reselect our features based on test data!** This will lead to training the system on the test data and have no way of assessing the performance of our model on new data or even worse, this can lead to overfitting to the test data! As a good practice it is wise to divide the data to ***3*** partitions: Training, Validation and Test and use the validation data to validate the model or reselect the features.

Problem 3

RMSE is the **R**oot of MSE which by itself is the **M**ean **S**quared **E**rror. MSE is analogous to the variance or the error of our estimator (machine learning model) and RMSE is analogous to the standard deviation of the error. By error (E) we mean the residual error or the difference of the prediction with the real predicted value. Each error is then squared so that negative and positive errors don't cancel each other out and mean (M) is just an average over all of them.

In order to consider the outliers we should consider one simple thing: The MSE criterion squares the error and RMSE normalized this square by rooting it. Therefore the higher the error, the higher distance of a residul (distance between prediction and real value), the higher the effect of it on MSE. So incase of outliers RMSE is a better measure of error because MSE would jump up with every outlier.

Problem 4

The basic idea behind Gradient Descent with momentum is to compute an exponentially weighted average of your gradients, and then use that gradient to update your weights instead. This will almost always work better than the straightforward gradient descent algorithm without momentum. However we should be careful in choosing the value of momentum because in large or really small values of momentum there is a chance that we miss the local minimum.

Problem 5

θ_0 in θ represent the bias or the intercept in our model and it represents the mean of the response variable (dependant variable) when the explanatory variables (independent variables) are zero. Normally we don't regularize the intercept. Why? Remember that regularization is used in order to avoid overfitting by reducing the magnitude of the coefficients and the value of intercept does not contribute to overfitting and therefore it is not desirable to put penalty on its magnitude.

Problem 6

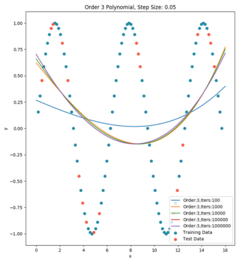
It will decrease the chances of overfitting. Due to having more data we can decrease our general error and our data represent the real world more and more as they increase, therefore having a good fit (or even an overfit) to the data will yield a better result.

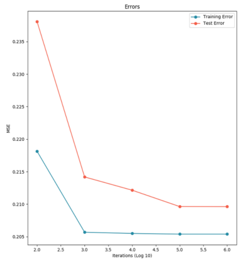
Problem 7

1. Using General Simple Models: This way the model is not to complex to overfit.
2. Use more data: As described in the previous problem
3. Use Cross Validation: Having a set of data to validate the results on would help avoiding overfitting and choosing a better model.
4. Use Regularization: Shrinking the coefficients will automatically lead to less overfitting.

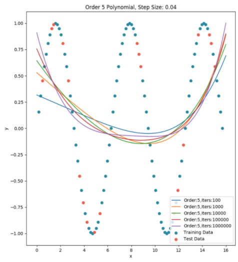
Programming Assignment

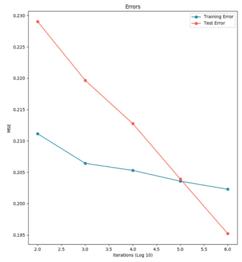
Part 1



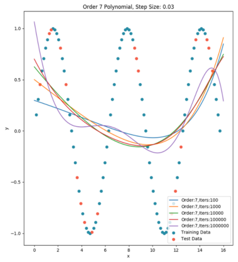


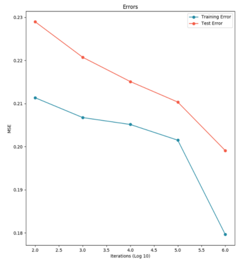

```
==========================
Order:          3
Alpha:          0.05
Iterations:     100
Training Error: 0.2181
Test Error:     0.2381
==========================
==========================
Order:          3
Alpha:          0.05
Iterations:     1000
Training Error: 0.2057
Test Error:     0.2142
==========================
==========================
Order:          3
Alpha:          0.05
Iterations:     10000
Training Error: 0.2055
Test Error:     0.2121
==========================
==========================
Order:          3
Alpha:          0.05
Iterations:     100000
Training Error: 0.2054
Test Error:     0.2096
==========================
==========================
Order:          3
Alpha:          0.05
Iterations:     1000000
Training Error: 0.2054
Test Error:     0.2096
==========================
```





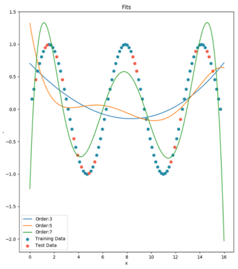

```
==========================
Order:          5
Alpha:          0.04
Iterations:     100
Training Error: 0.2111
Test Error:     0.229
==========================
==========================
Order:          5
Alpha:          0.04
Iterations:     1000
Training Error: 0.2064
Test Error:     0.2196
==========================
==========================
Order:          5
Alpha:          0.04
Iterations:     10000
Training Error: 0.2053
Test Error:     0.2128
==========================
==========================
Order:          5
Alpha:          0.04
Iterations:     100000
Training Error: 0.2036
Test Error:     0.2039
==========================
==========================
Order:          5
Alpha:          0.04
Iterations:     1000000
Training Error: 0.2023
Test Error:     0.1952
==========================
```

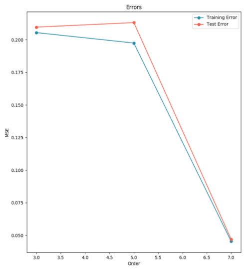





```
==========================
Order:          7
Alpha:          0.03
Iterations:     100
Training Error: 0.2113
Test Error:     0.229
==========================
==========================
Order:          7
Alpha:          0.03
Iterations:     1000
Training Error: 0.2067
Test Error:     0.2207
==========================
==========================
Order:          7
Alpha:          0.03
Iterations:     10000
Training Error: 0.2051
Test Error:     0.2151
==========================
==========================
Order:          7
Alpha:          0.03
Iterations:     100000
Training Error: 0.2015
Test Error:     0.2103
==========================
==========================
Order:          7
Alpha:          0.03
Iterations:     1000000
Training Error: 0.1796
Test Error:     0.1991
==========================
```

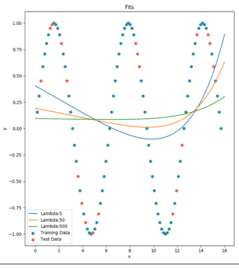
Part 2

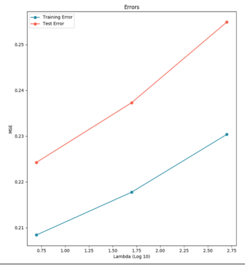





```
==========================
Order:          3
Training Error: 0.2054
Test Error:     0.2096
==========================
==========================
Order:          5
Training Error: 0.1974
Test Error:     0.2131
==========================
==========================
Order:          7
Training Error: 0.0453
Test Error:     0.0469
==========================
```

Part 3






```
==========================
Lambda:         5
Training Error: 0.2084
Test Error:     0.2243
==========================
==========================
Lambda:         50
Training Error: 0.2177
Test Error:     0.2373
==========================
==========================
Lambda:         500
Training Error: 0.2304
Test Error:     0.2549
==========================
```