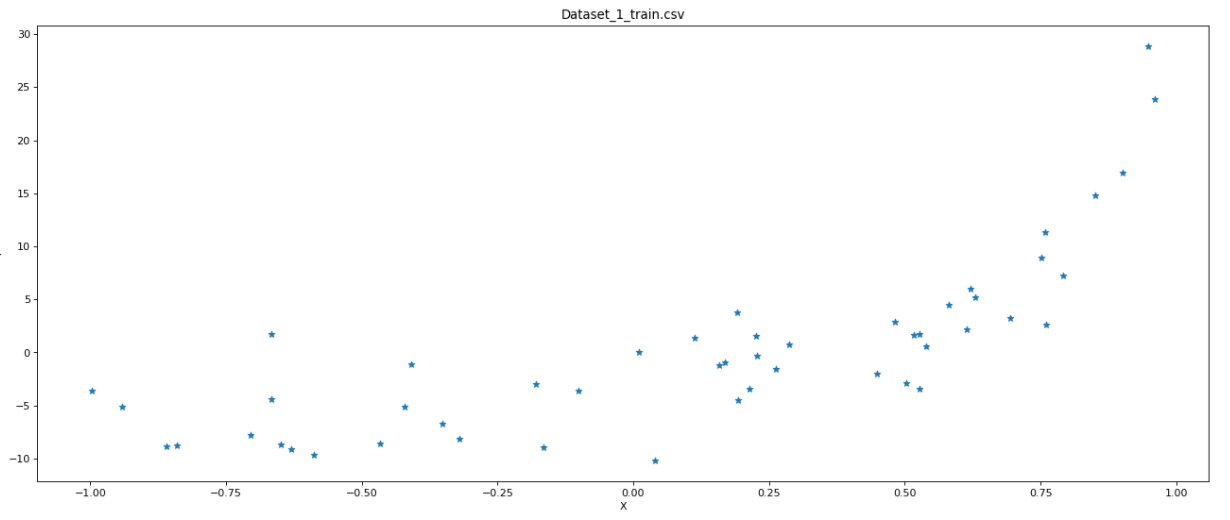
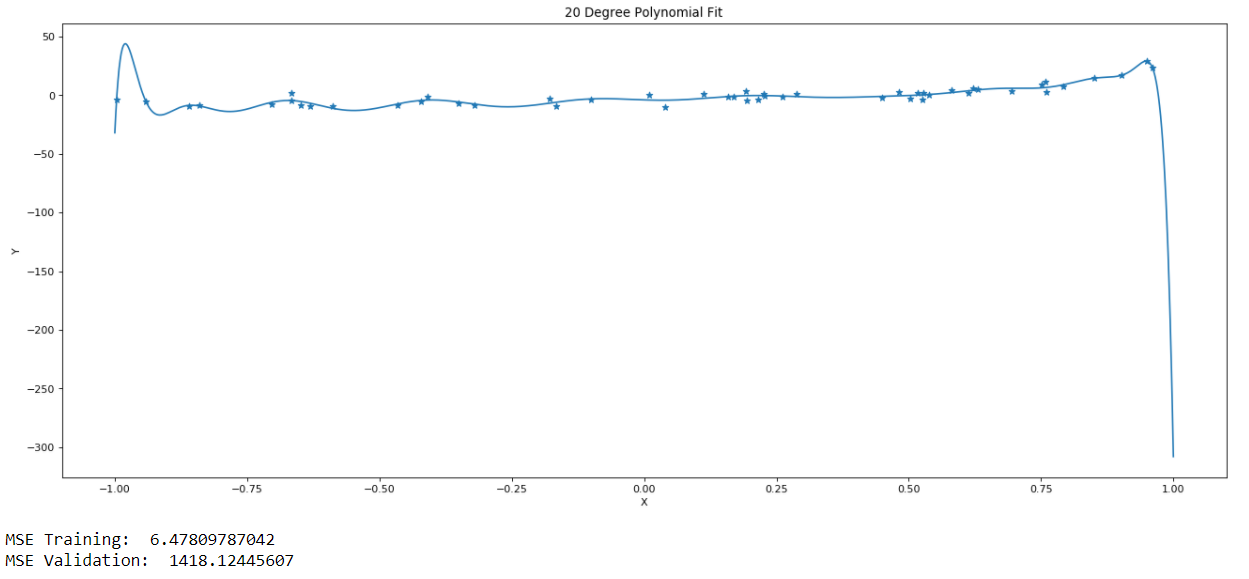
# Question 1

## 1.1

**Visualizing the Dataset**



**Visualizing the 20-Degree Polynomial Fit**

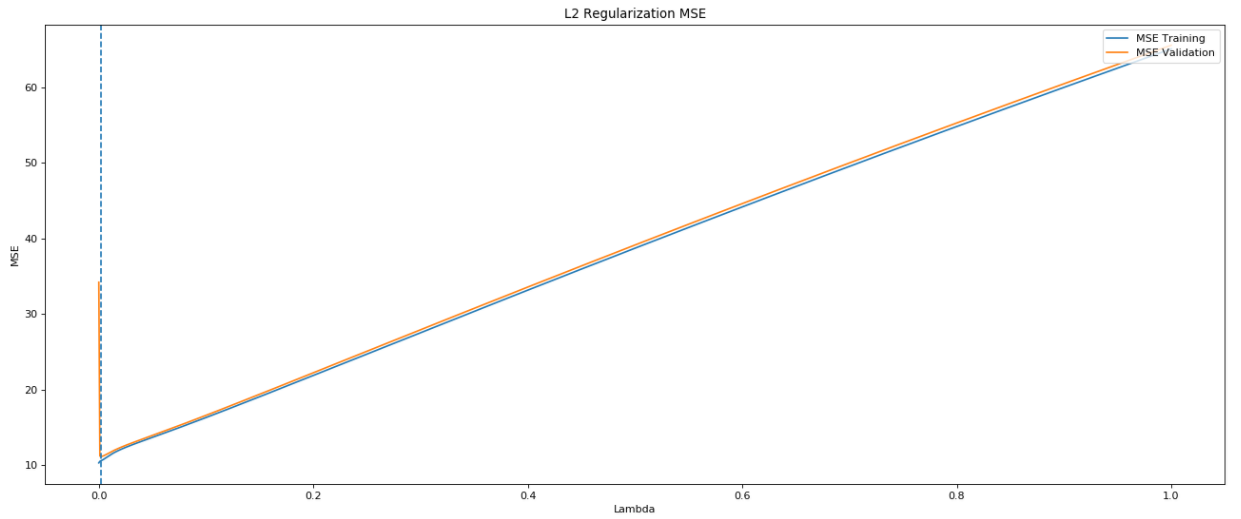
****

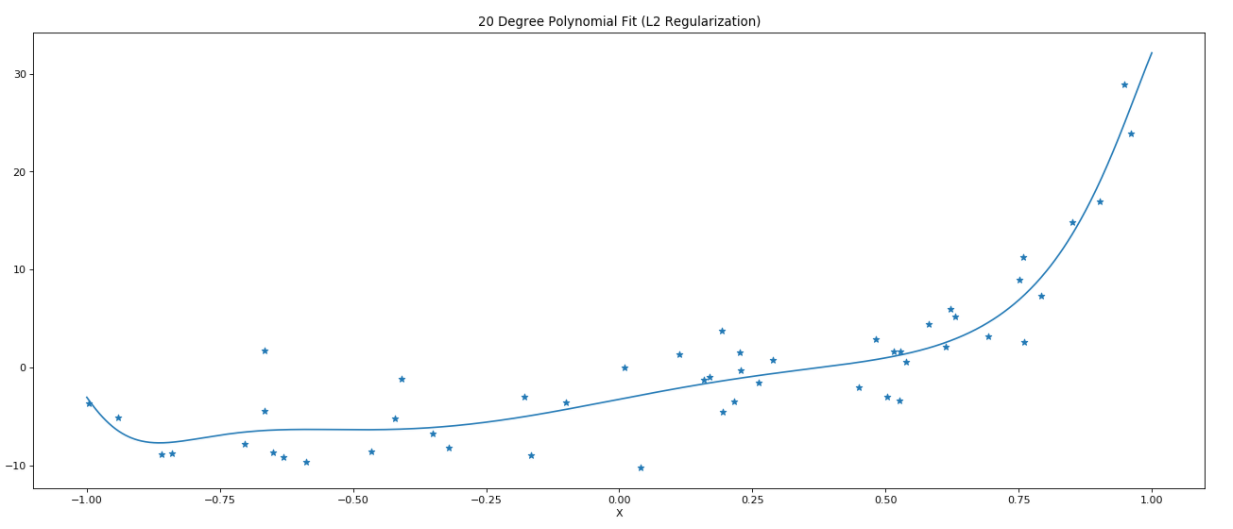
**MSE Training = 6.47**

**MSE Validation = 1418**

The quality of the fit looks fine if our target vector is between -0.8 and 0.8. However, there is clear overfitting. For any target Y greater than 0.8, the predicted value will be off significantly. We need to add regularization to the fit.

## 1.2





The best L2 regularization occurs at **Lambda = 0.002003**. At this point, the minimum MSE for training is **10.57**, and the minimum MSE for validation is **11.16**, and the minimum MSE for testing is **12.84.**

## 1.3

It looks like the fit with regularization is a 6th degree polynomial. I am inferring this from the number of highs and lows of the graph. The fit has 2 local maximum, and 3 global minimum. This would mean that in theory, this function would cross the X-axis 5 times, and therefore be a 6th degree polynomial.

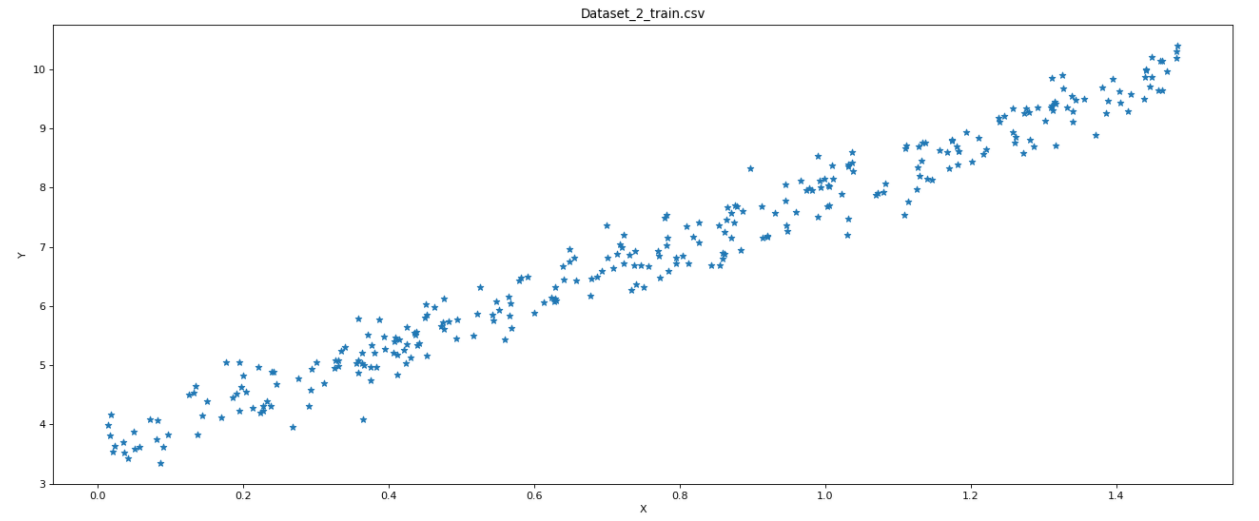
Shown below is an example of a 6th degree polynomial.



# Question 2

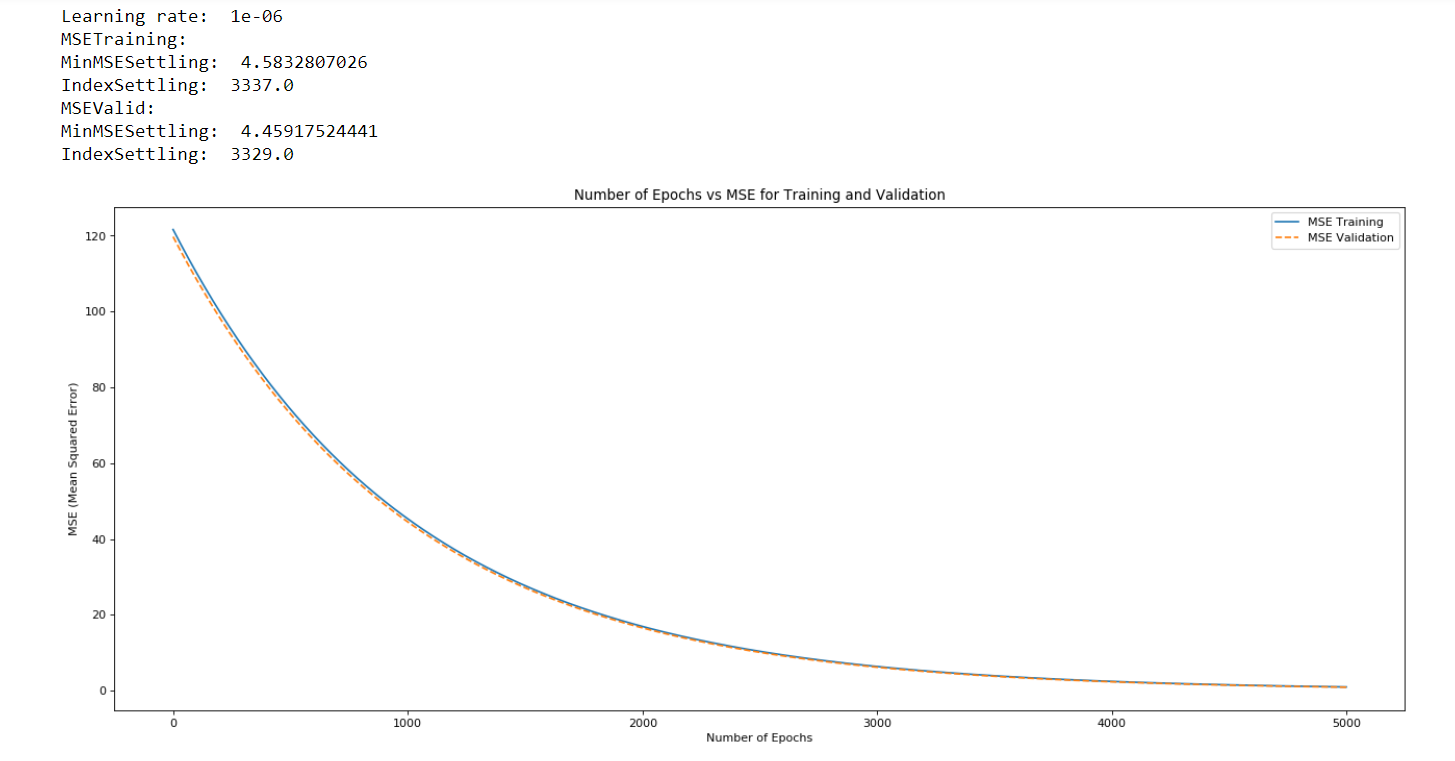
## 2.1

**Visualizing the Data**

****

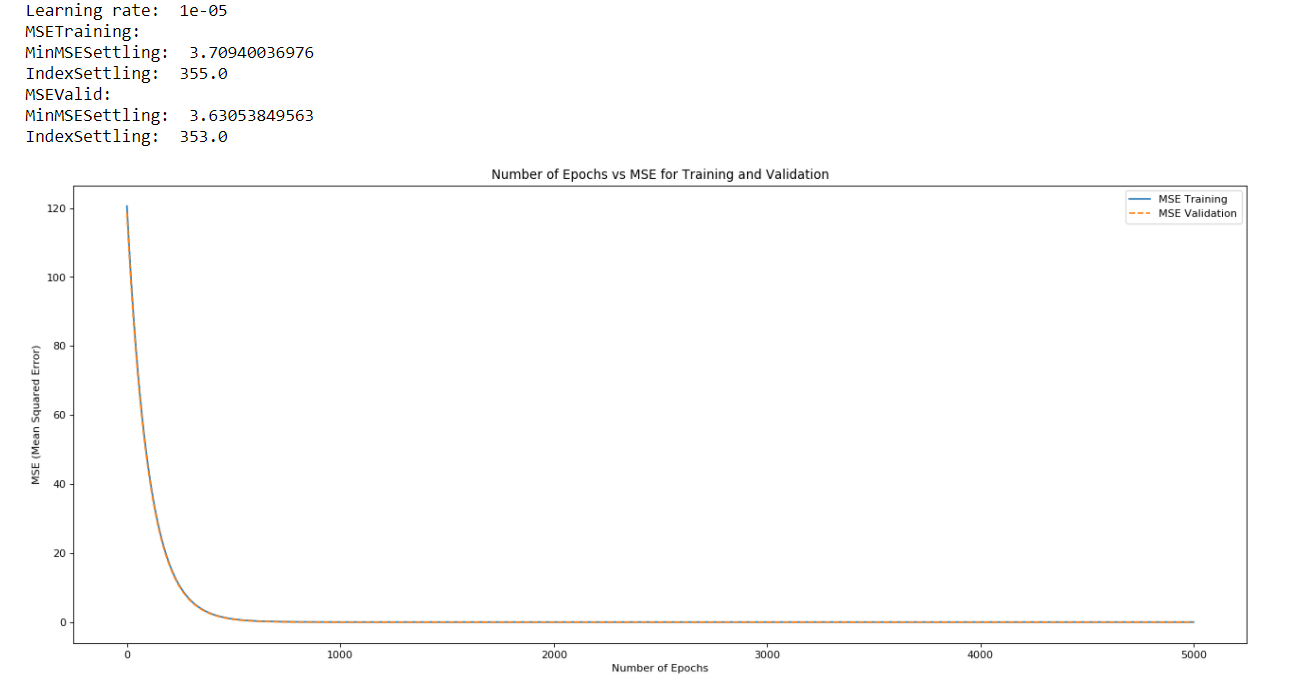
For this question, I was running 5k epochs. I declare that my MSE (therefore weights as well) have converged when the MSE has completed 5 half-lives. This corresponds to 97% of the final value.

**Learning rate = 1e-06**



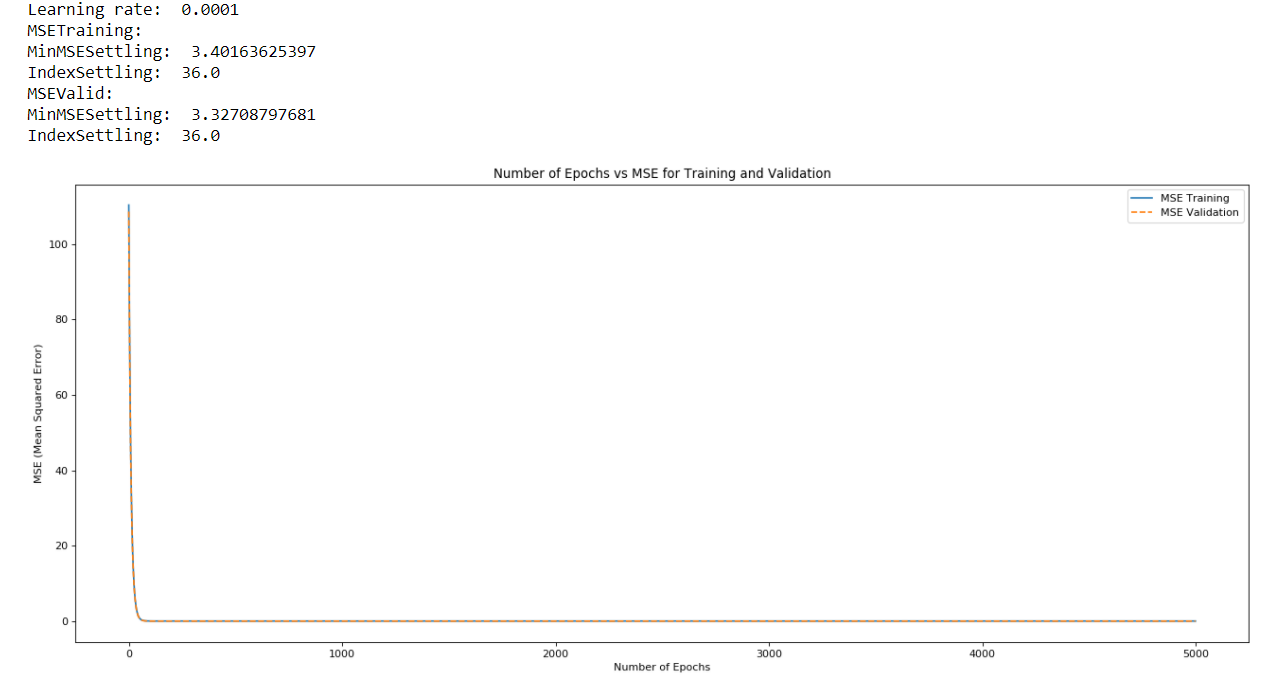
MSE converges at epoch number **3337**. The settling training MSE is **4.58**, and the settling validation MSE is **4.46**.

**Learning rate = 1e-05**



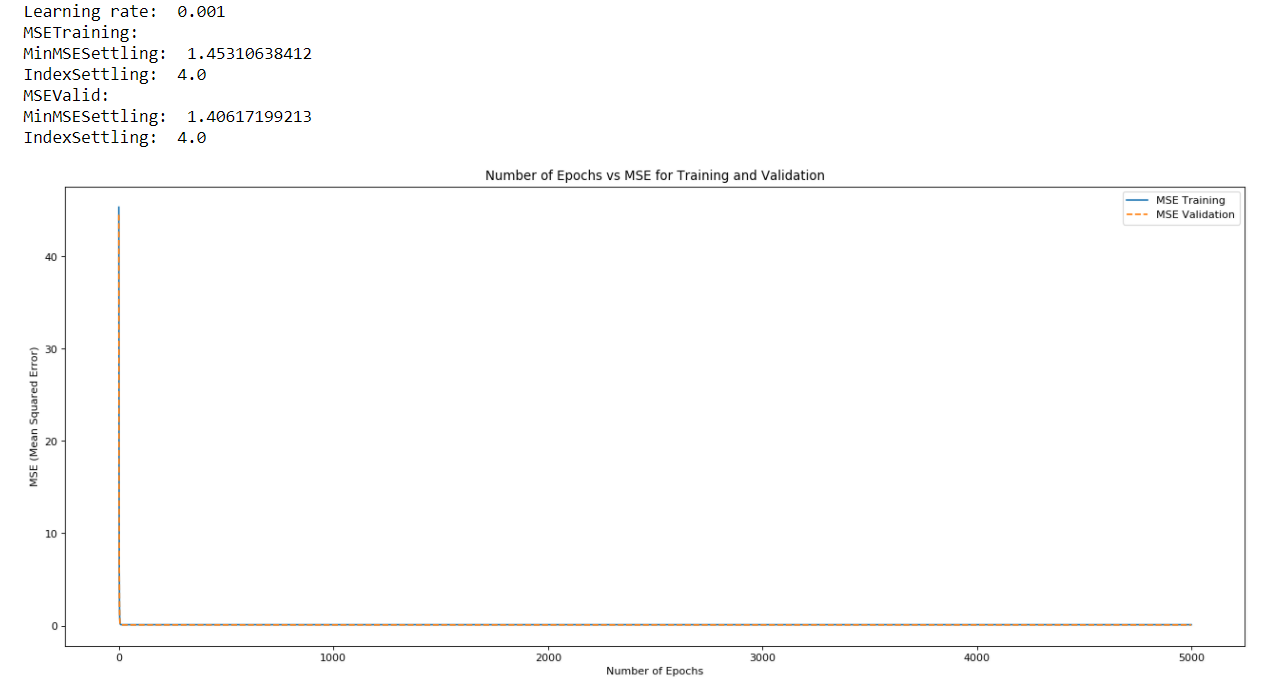
MSE converges at epoch number **335**. The settling training MSE is **3.71**, and the settling validation MSE is **3.63**.

**Learning rate = 1e-04**



MSE converges at epoch number **36**. The settling training MSE is **3.4**, and the settling validation MSE is **3.33**.

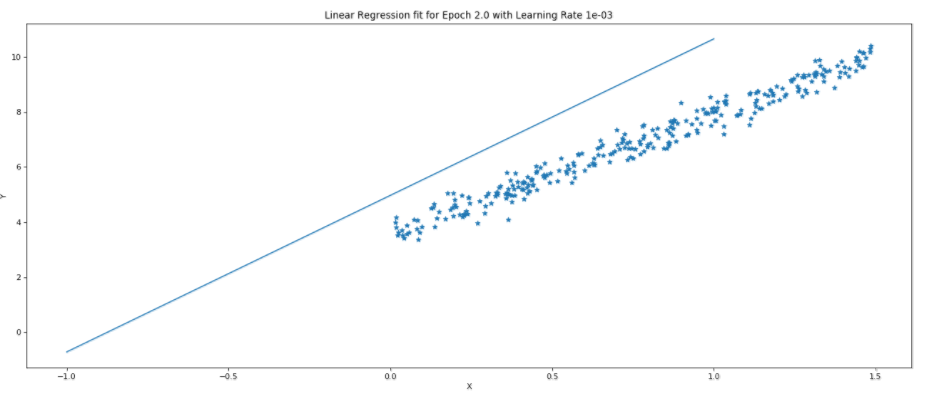
**Learning rate = 1e-03**



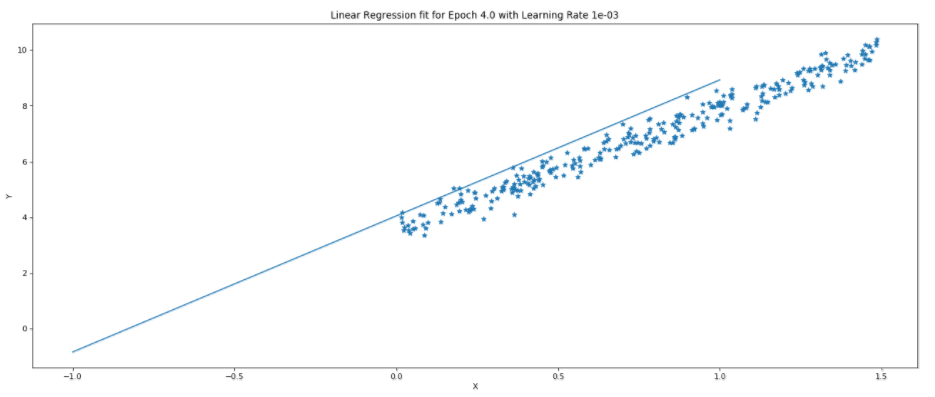
MSE converges at epoch number **4**. The settling training MSE is **1.45**, and the settling validation MSE is **1.406**.

It seems as if the best learning rate is 1e-03. Any higher than this and the function converges in less than 1 epoch, which is dangerous as it could diverge. Here are 5 epochs for how the fit evolves.

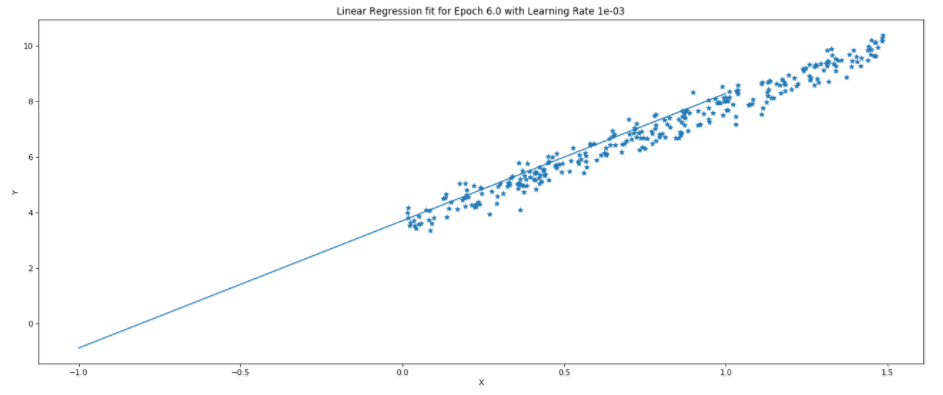
**Epoch 2**



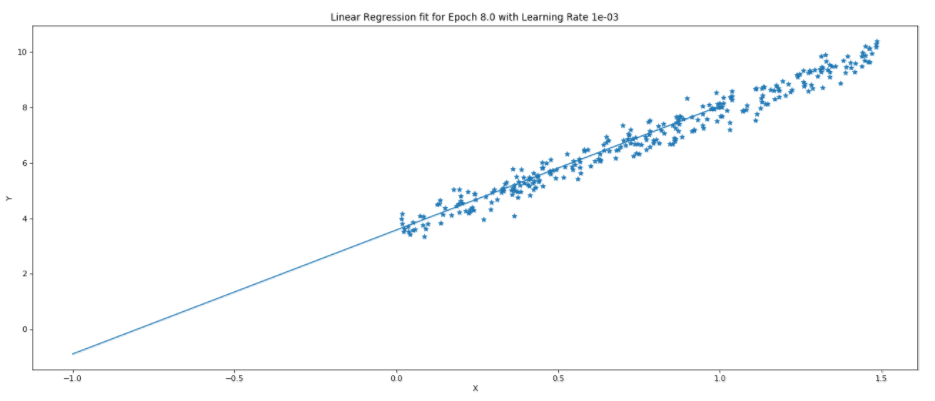
**Epoch 4**



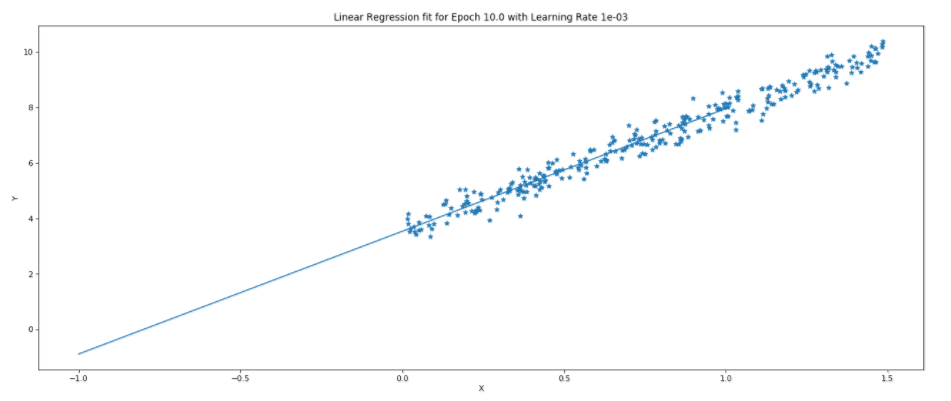
**Epoch 6**

****

**Epoch 8**

****

**Epoch 10**

****

As is clear from the visualizations above, the fit evolves quite rapidly for Learning rate = 1e-03. After 6 epochs we are pretty close to the final value, and after 8 we have practically found the best fit possible.