NAME: DUONG DUY CHIEN

Regularized Linear Regression and Bias vs. Variance

I. Theory:

1. Multi-class Classification

Hypothesis:

$$h_\theta(x) = g(\theta^T x) = g(\theta_0 x_0 + \theta_1 x_1 + ... + \theta_n x_n)$$

$$=\mathrm{g(z)}=\frac{1}{1+e^{-z}}$$

Linear regression cost function:

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)} - y^{(i)})^2 + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

Gradient descent algorithm:

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} \qquad \text{for j =0}$$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} \quad + \frac{\lambda}{m}\theta_j \ \text{ for j } \geq 1$$

with m= number of samples

j= 1....n : number of feartures

The way to implement by vectorization is quite similar to the homework 1 so that I won't mention it again in here.

II. Results:

1. Linear regression and learning curve

- Calculate cost and gradient to fit linear regression with dataset (as shown in Figure 1)

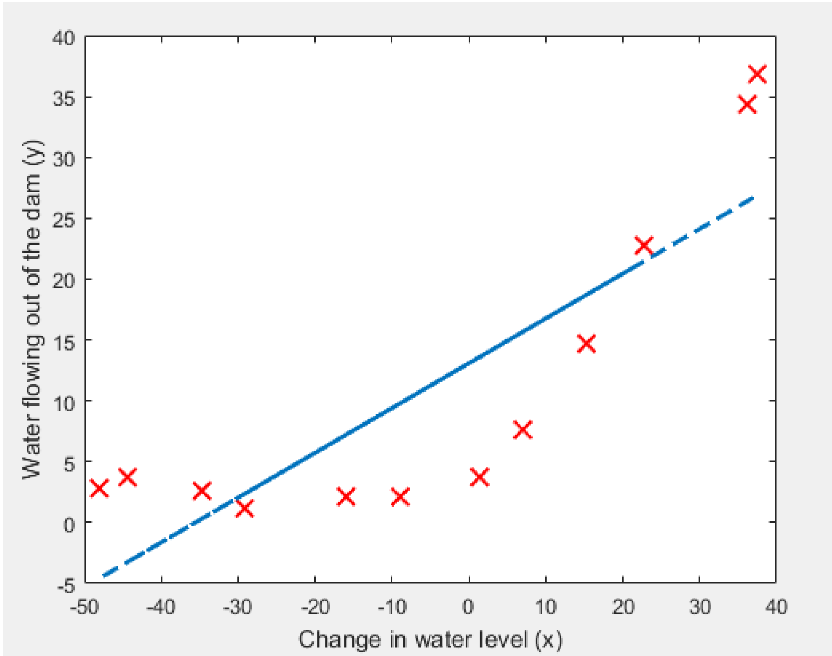

Figure 1 Visualize data and linear fit

- I use a loop from 1 to n (number of full training examples) to draw linear regression learning curve. The learning curve represented the cost function in Training and Cross Validation dataset base on the number of training examples. The result was illustrated in Fig.2. Note: we always calculate the cost for whole examples in cross validation set.

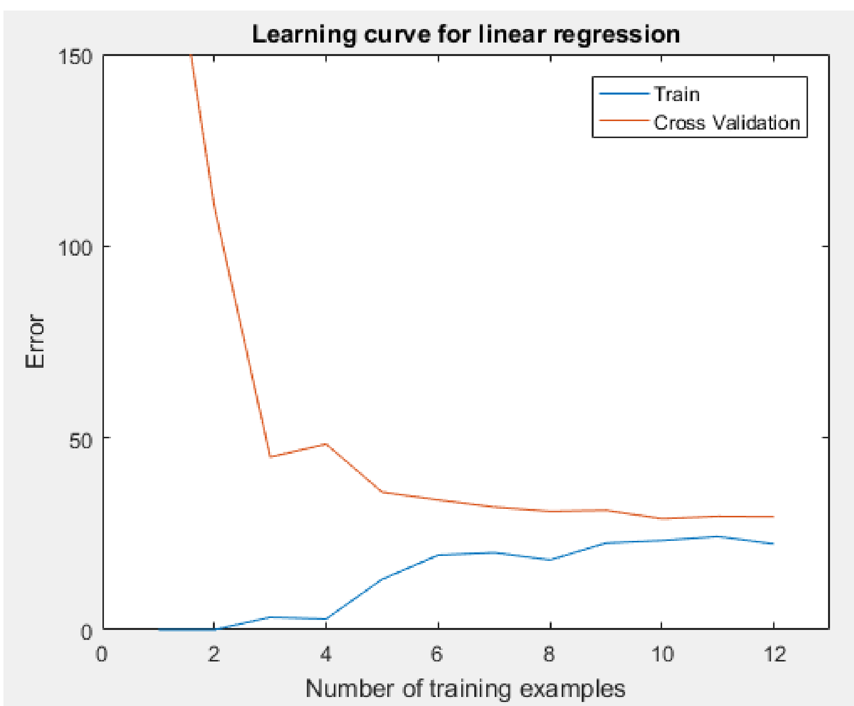

Figure 2 Linear regression learning curve

2. Polynomial regression

This part helps me to understand how to choose the best regularization term.

1. we will address this overfitting problem by adding more features. In Figure 3, we can see that the linear fit is overfitting the dataset. Figure 4 shown the learning curve, we can see the big gap between cost function of training and cross validation set which addressed that our current model is overfitting (cost function in trainingset is too small while in cross validatin dataset is too large).

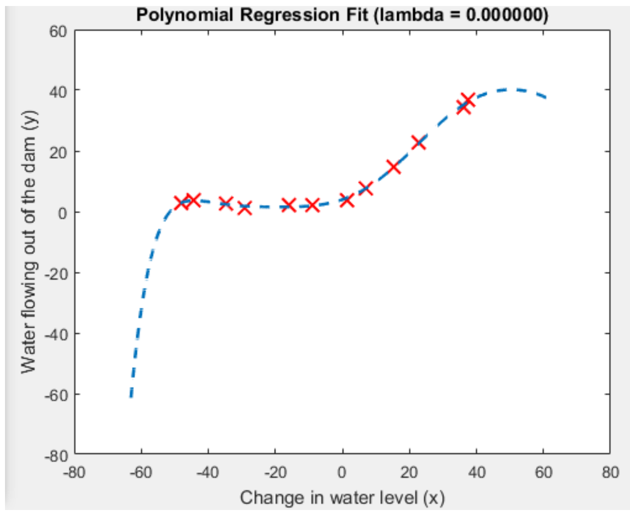

Figure 3 Linear fit with polynomial regression

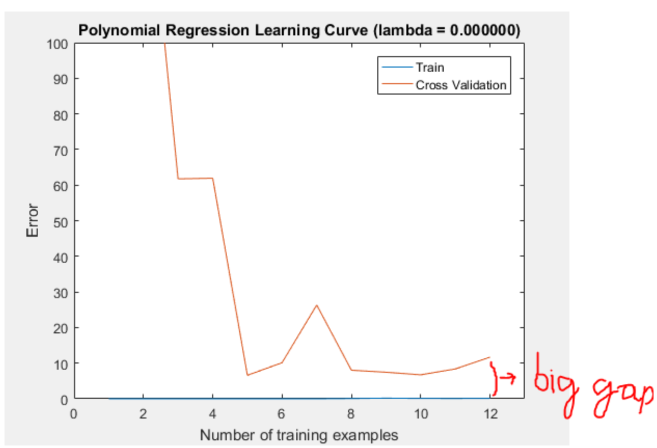

Figure 4 Polynomial learning curve, lambda=0

2. We can see the effect of regularization term by adjusting lambda (as shown in Figure 5, 6, 7, 8)

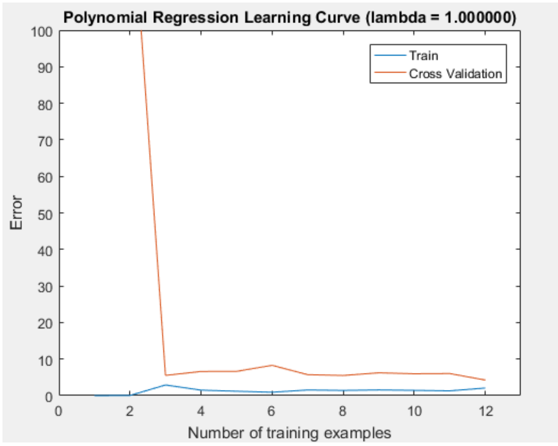

Figure 5 Polynomial learning curve, lambda=1

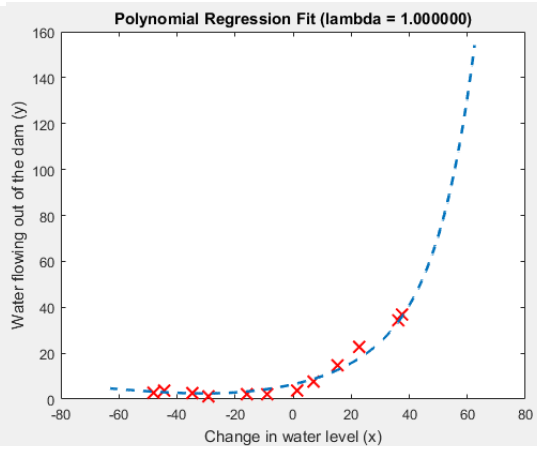

Figure 6 Polynomial fit, lambda=1

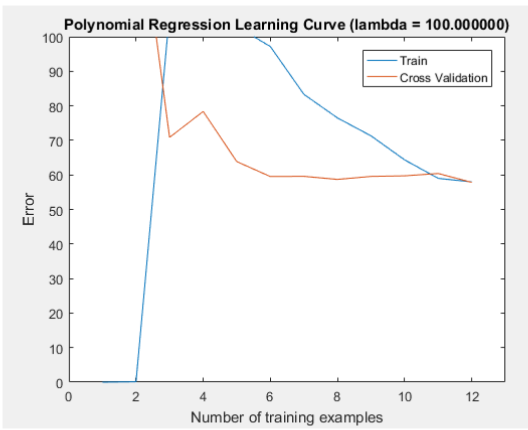

Figure 5 Polynomial learning curve, lambda=100

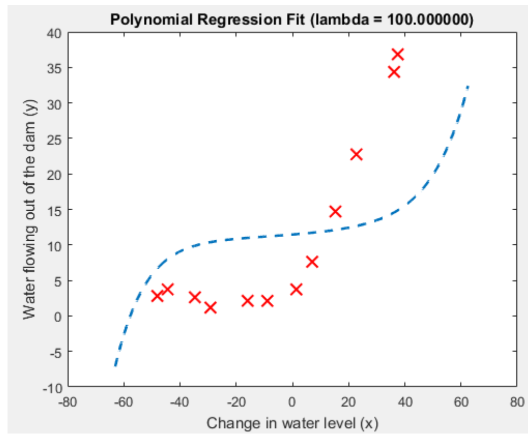

Figure 6 Polynomial fit, lambda=100

3. In order to choose the best lambda, we will draw the learning curve when we adjust the value of lambda. The best suitable lambda is the value that let the cost function in Cross Validation set smallest. As we can see in Figure 9, the best lambda we should choose is 3 where the cost function of Cross Validation set is smallest.

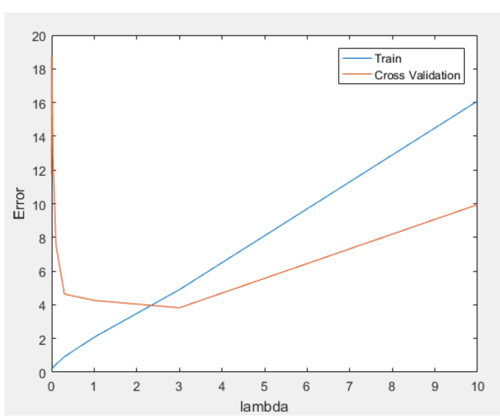

Figure 7 Selecting lambda using a cross validation test

4. After we chosen the best lambda=3, We can evaluate our result on test set and the result obtained : test error = 3.85988

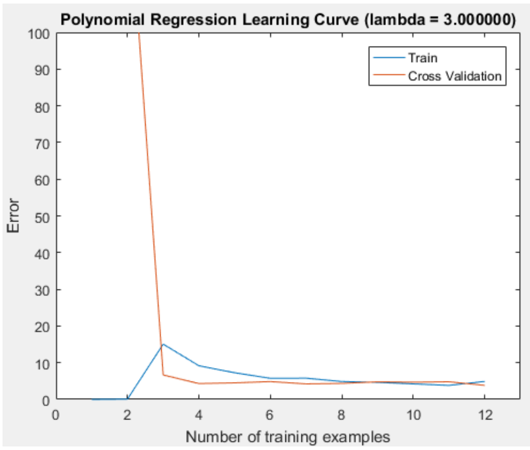

Figure 10 Polynomial learning curve, lambda=3

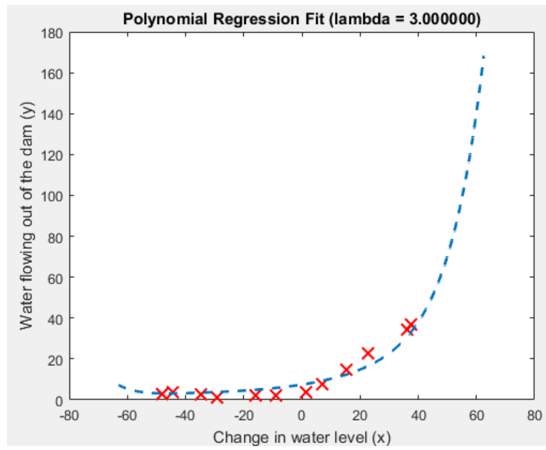

Figure 81 Polynomial fit, lambda=3

```
test error 3.859888
Polynomial Regression (lambda = 3.000000)
```