PA 1

Letao Qi lq4
Dan Ye dy13

Part 1

Problem 1

Making predictions on unseen data

Theta found by gradient descent: 24.039777 -0.438154
For lower status percentage = 5, we predict a median home value of 218490.093455
For lower status percentage = 50, we predict a median home value of 21320.998737

Problem 2

Making predictions on unseen data

Predicted median home price for the average census tract (using gradient descent):
$225285.711187

Normal equations

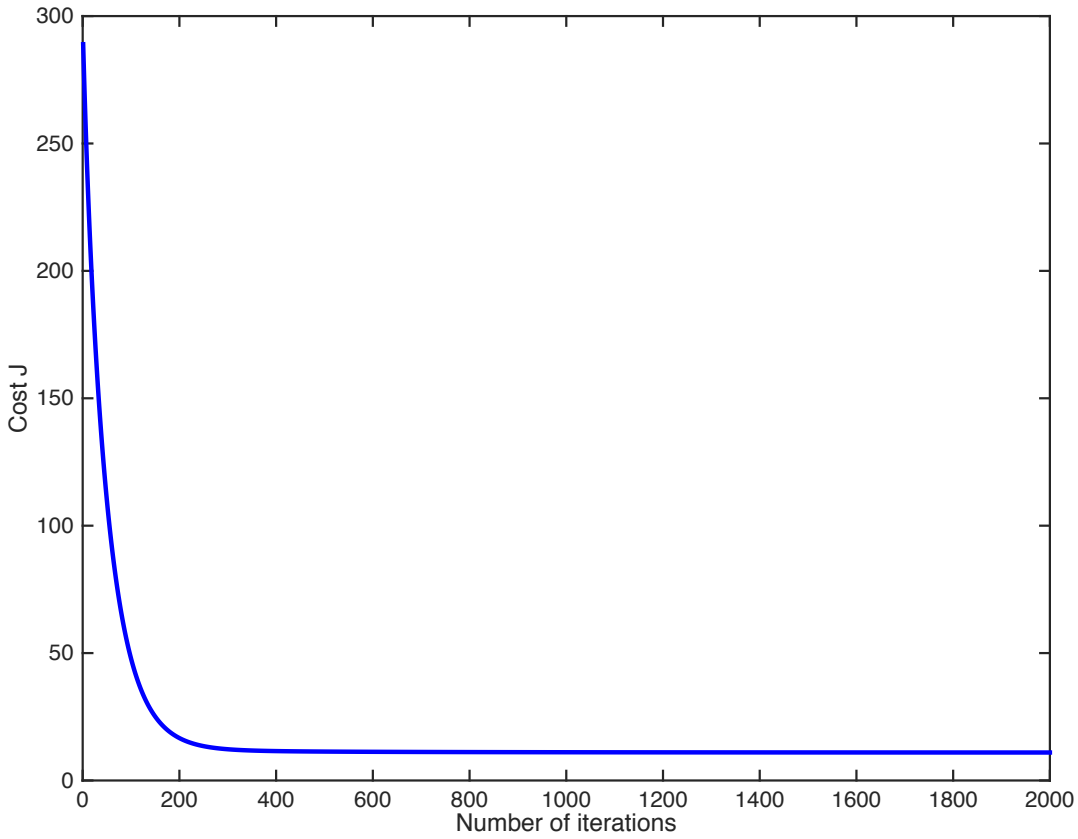
Predicted median home price for the average census tract (using normal equations):
$225328.063240

The result from normal equations equals the one being predicted approximately.

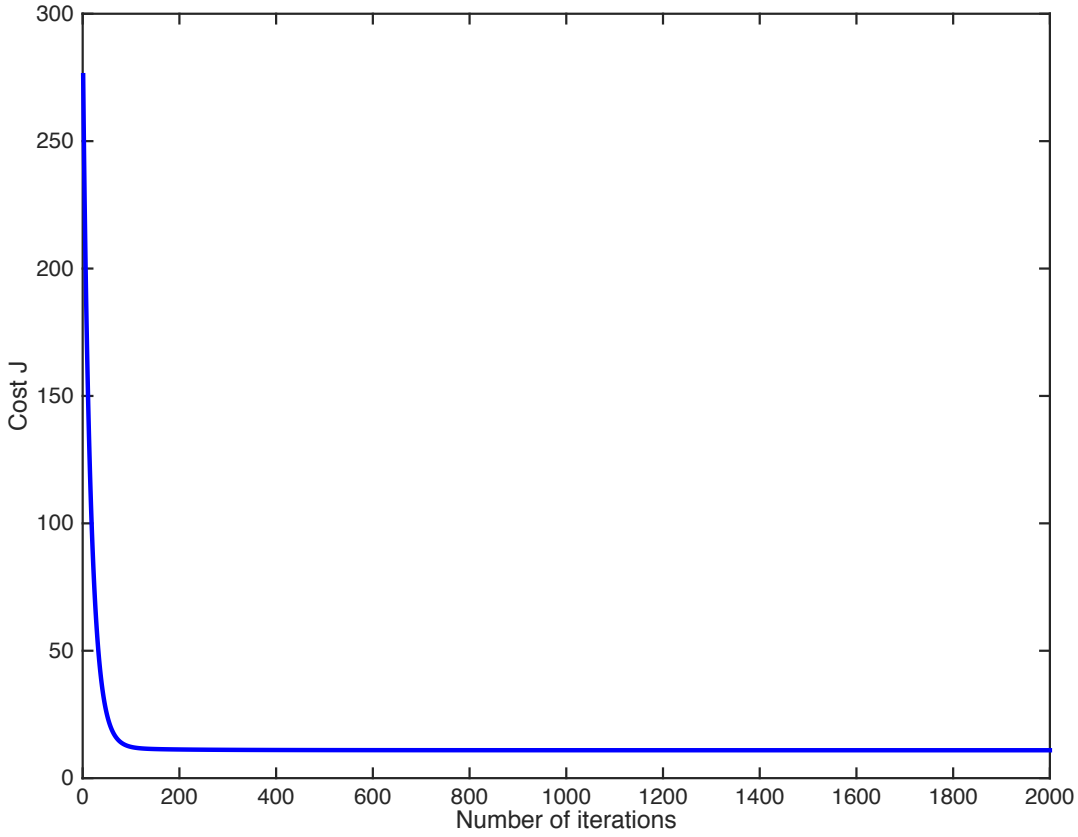
Exploring convergence of gradient descent

From the following figures, we can see that a better pair of values for alpha and number of iterations would be 30 and 50.

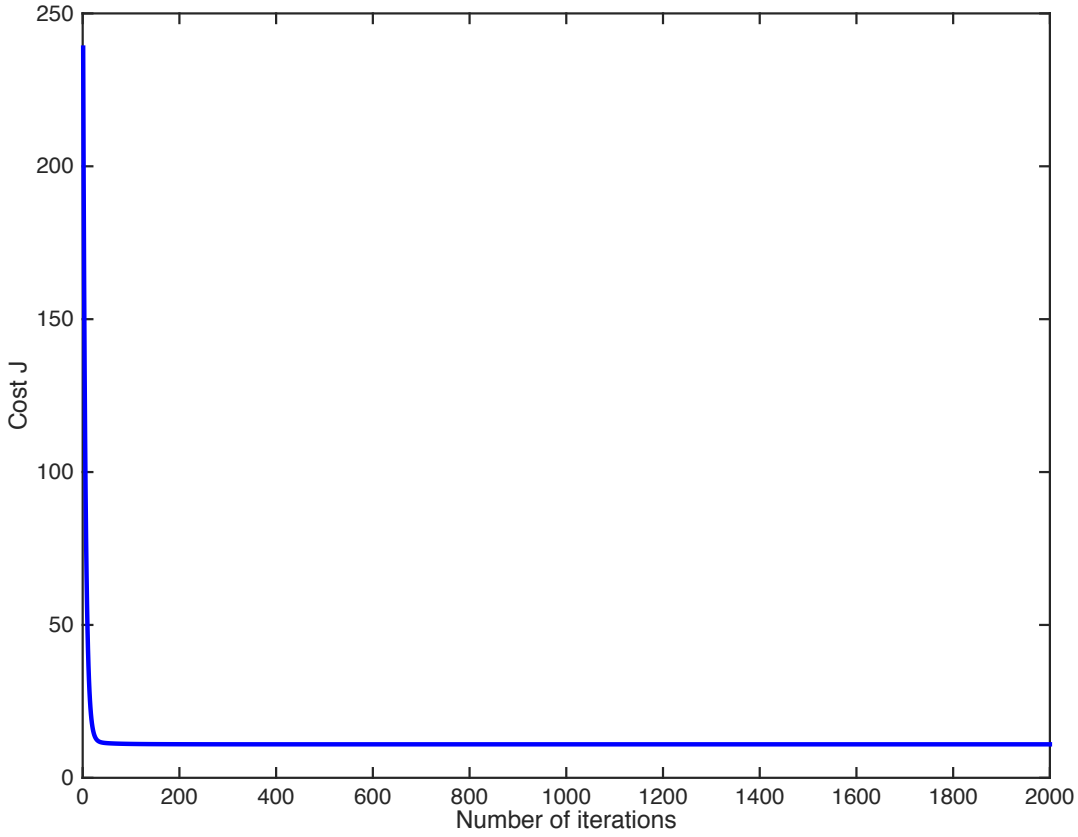
alpha = 0.01


300
250
200
150
100
50
0
Cost J
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Number of iterations

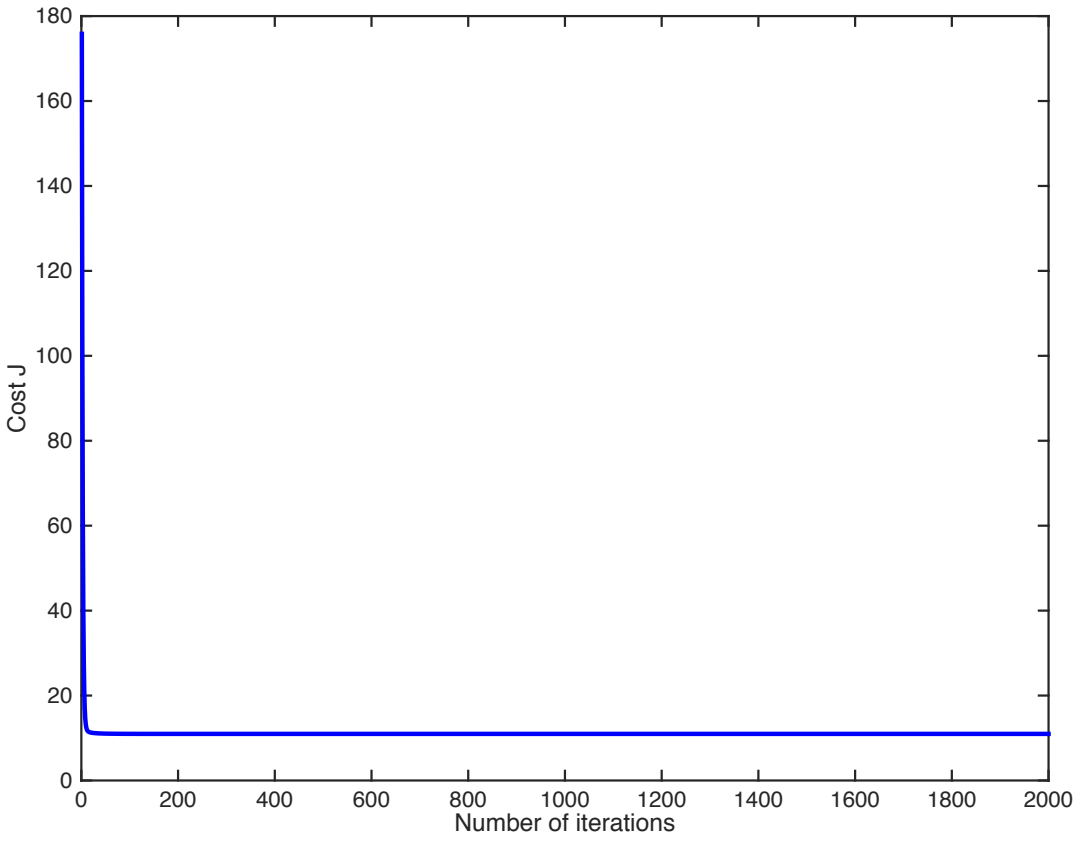
alpha = 0.03


300
250
200
150
100
50
0
Cost J
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Number of iterations

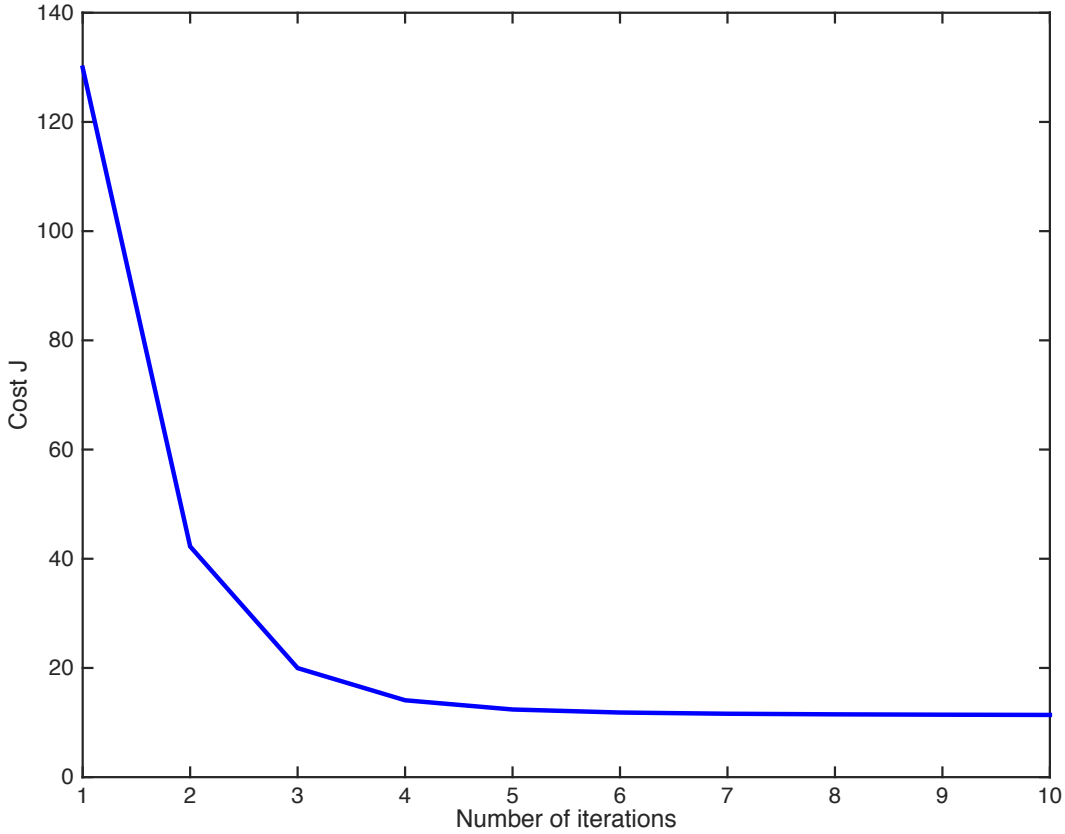
alpha = 0.1


250
200
150
100
50
0
Cost J
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Number of iterations

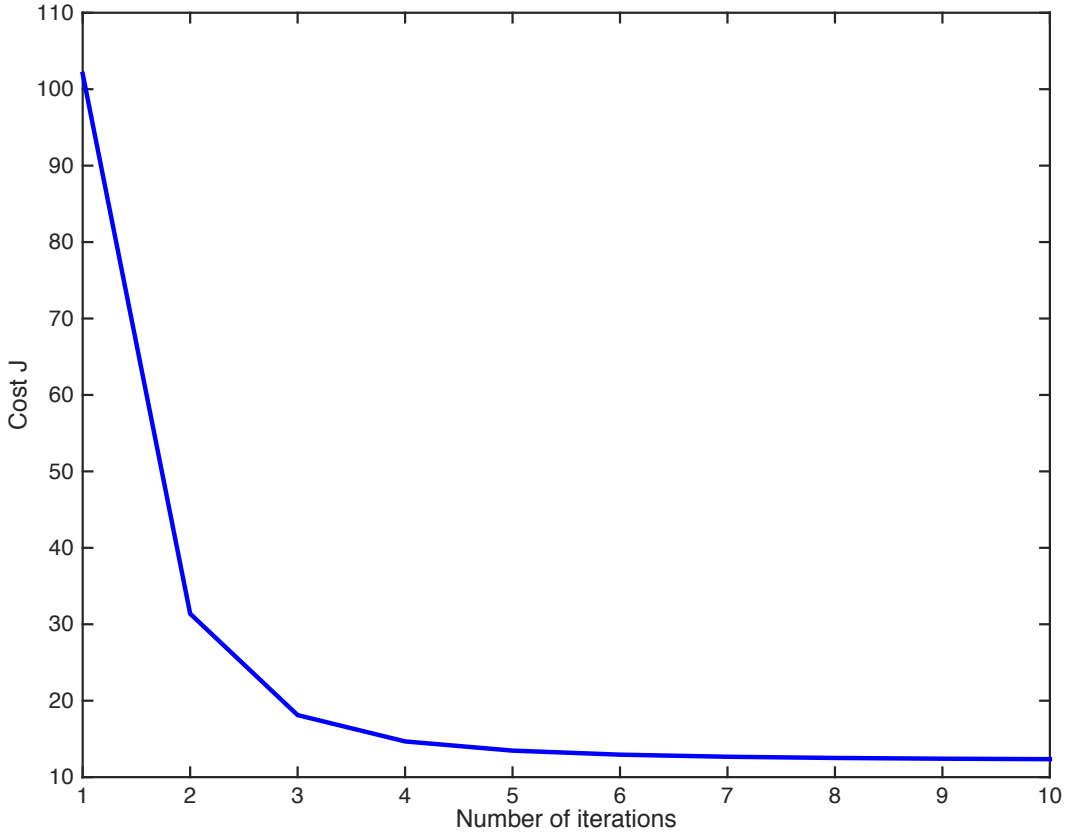
alpha = 0.3


180
160
140
120
100
80
60
40
20
0
Cost J
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Number of iterations

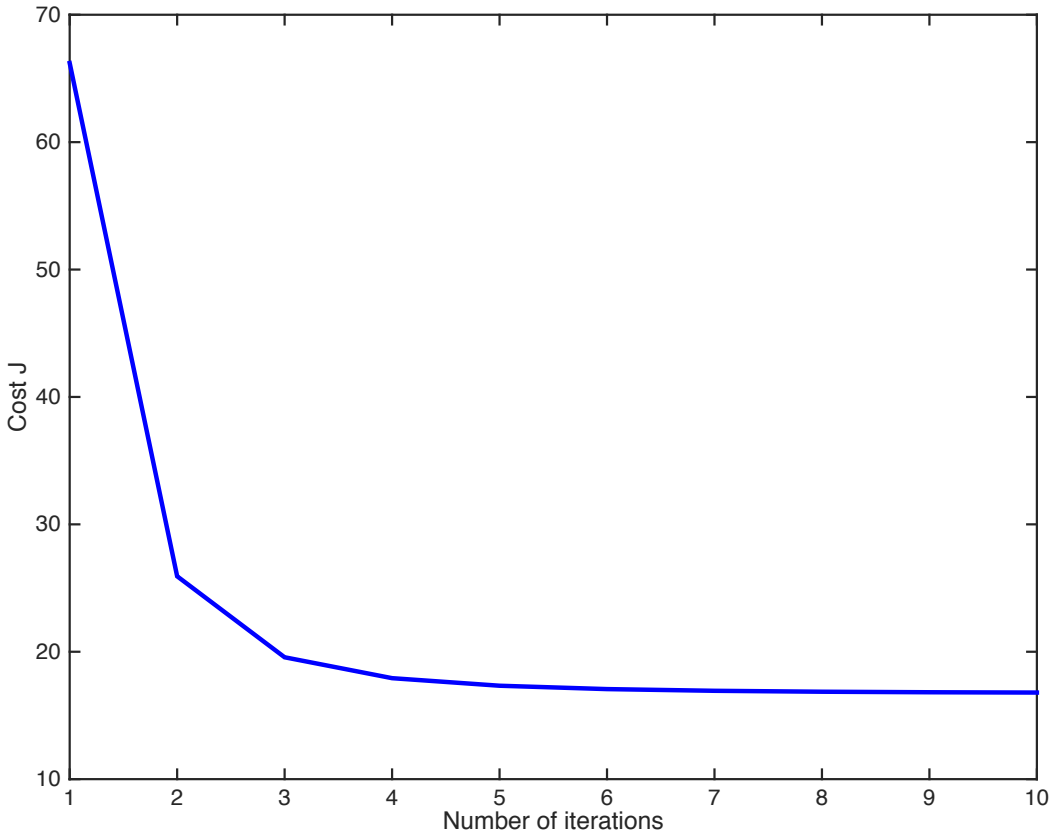
alpha = 1


140
120
100
80
60
40
20
0
Cost J
1
2
3
4
5
6
7
8
9
10
Number of iterations

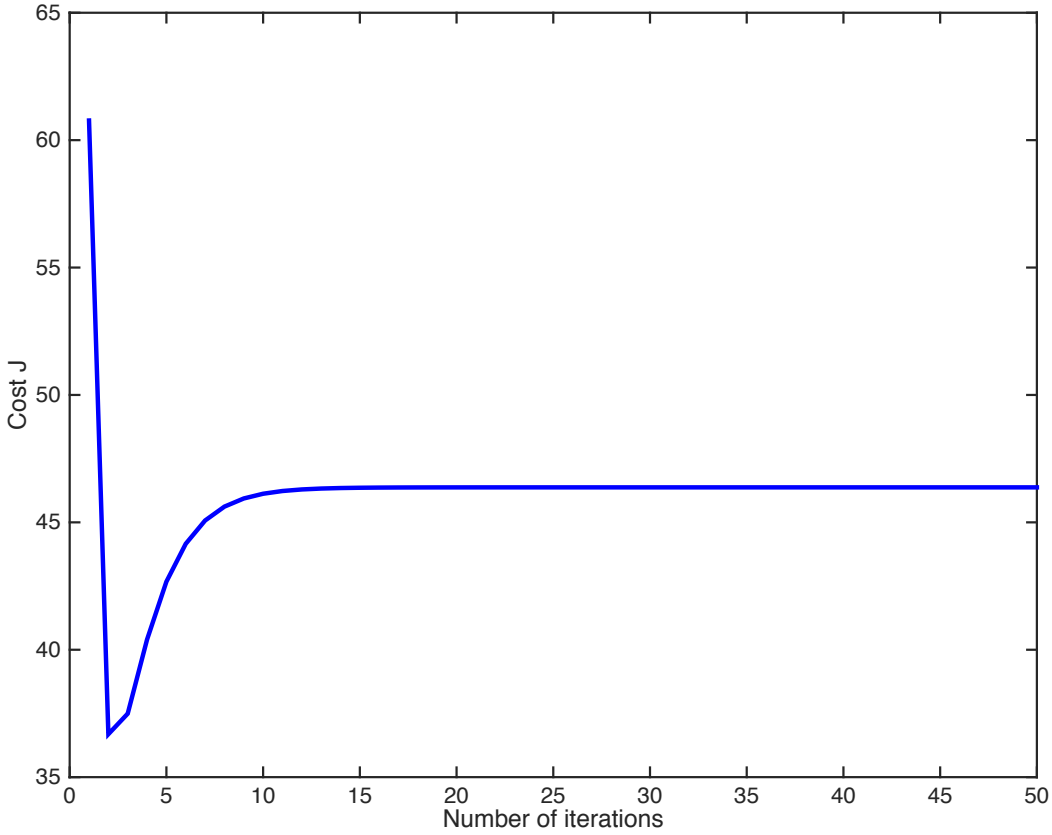
alpha = 3


110
100
90
80
70
60
50
40
30
20
10
Cost J
1
2
3
4
5
6
7
8
9
10
Number of iterations

alpha = 10



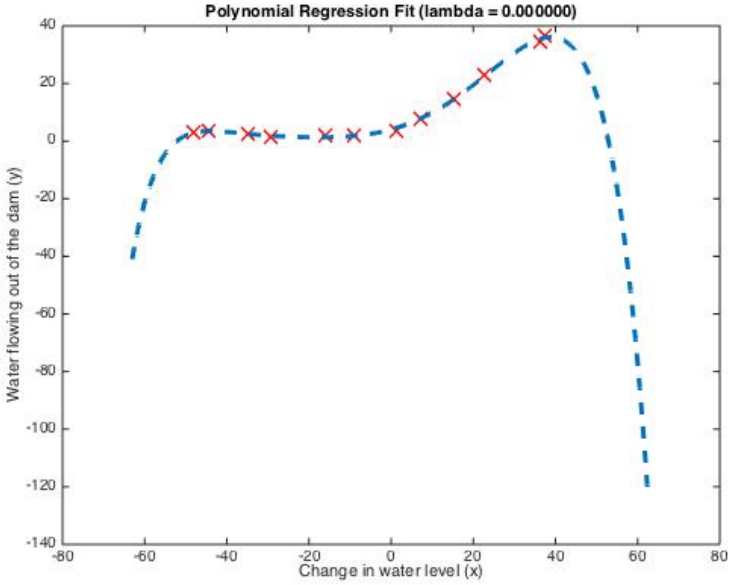
alpha = 30

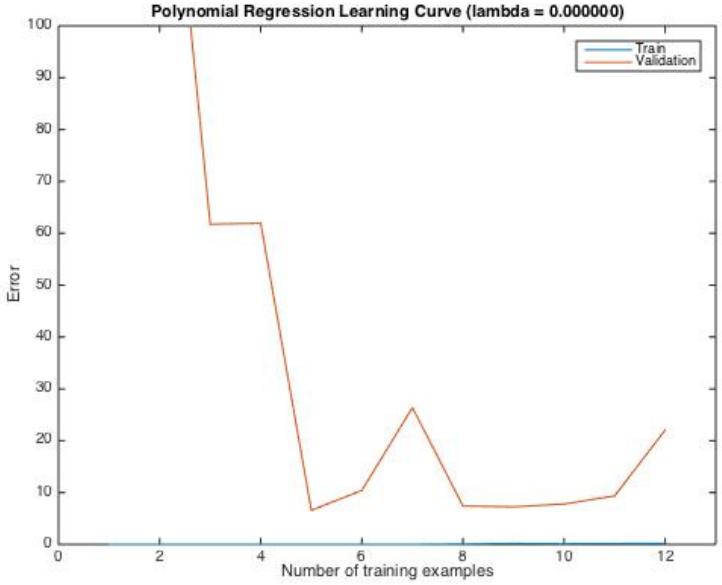

65
60
55
50
45
40
35
Cost J
0
5
10
15
20
25
30
35
40
45
50
Number of iterations

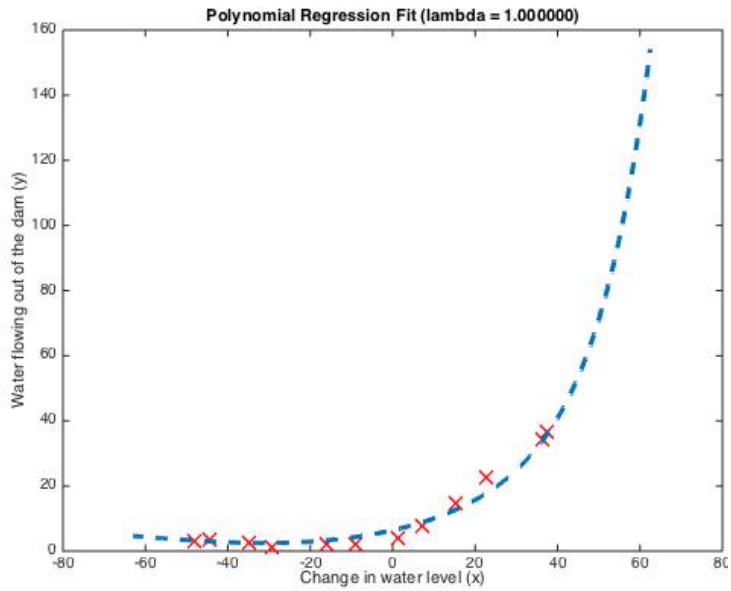
Part 2

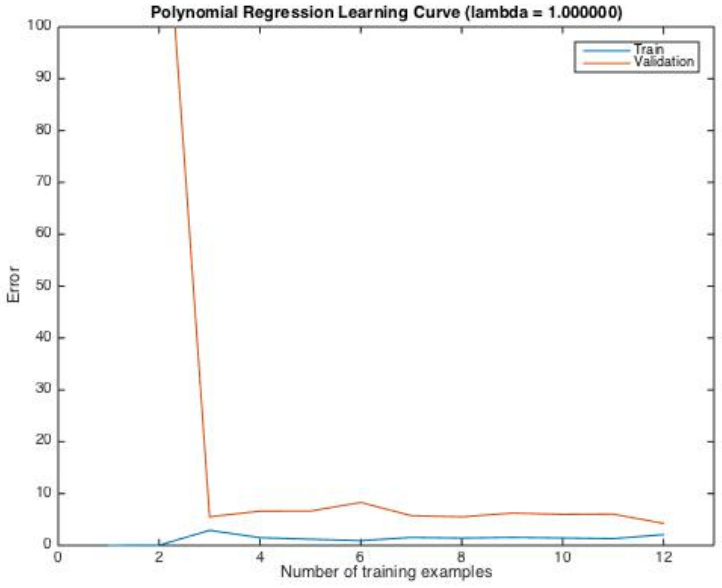
Problem 1

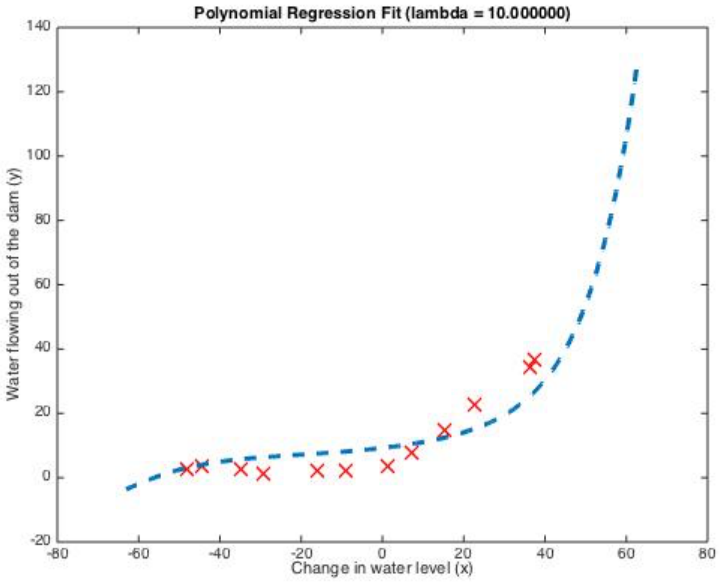
Adjusting the regularization parameter

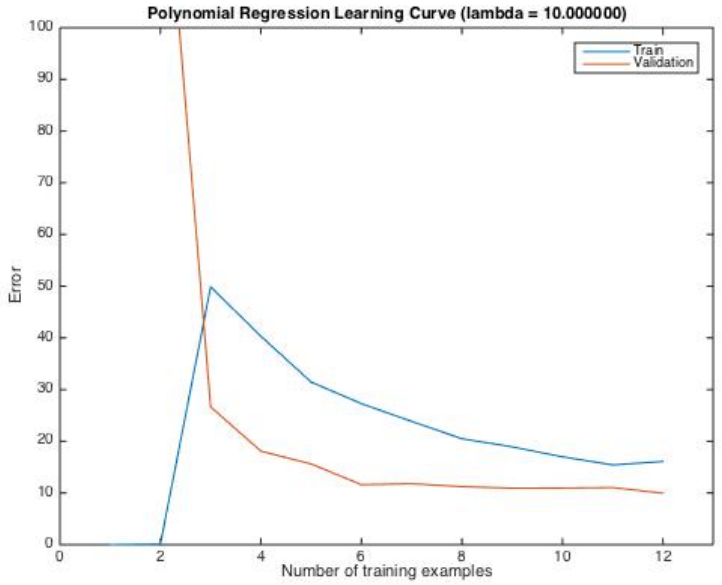


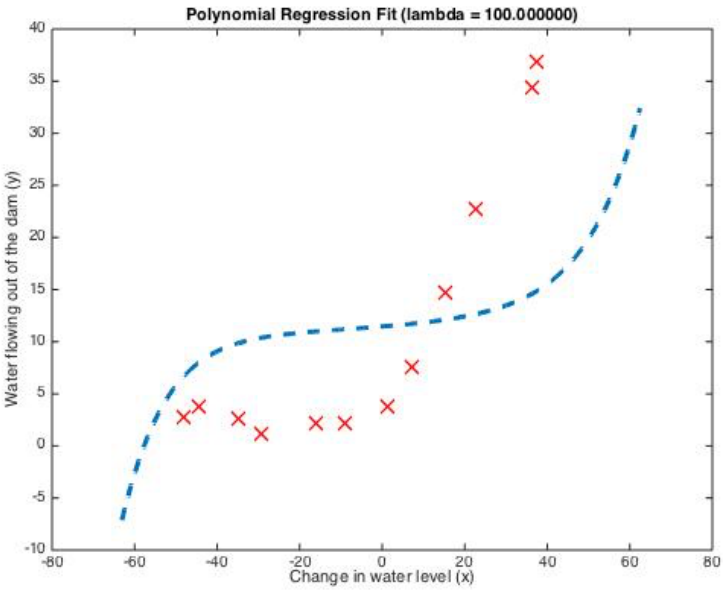


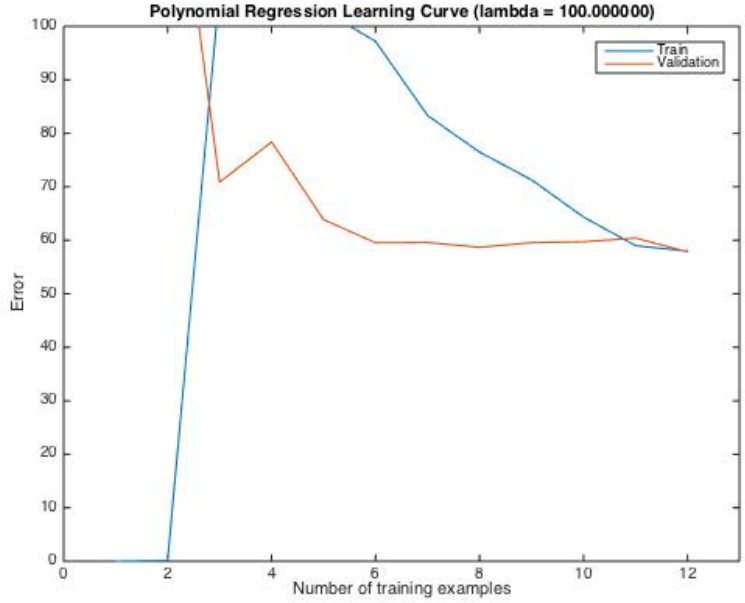





Polynomial Regression Fit (lambda = 10.000000)
Water flowing out of the dam (y)
Change in water level (x)

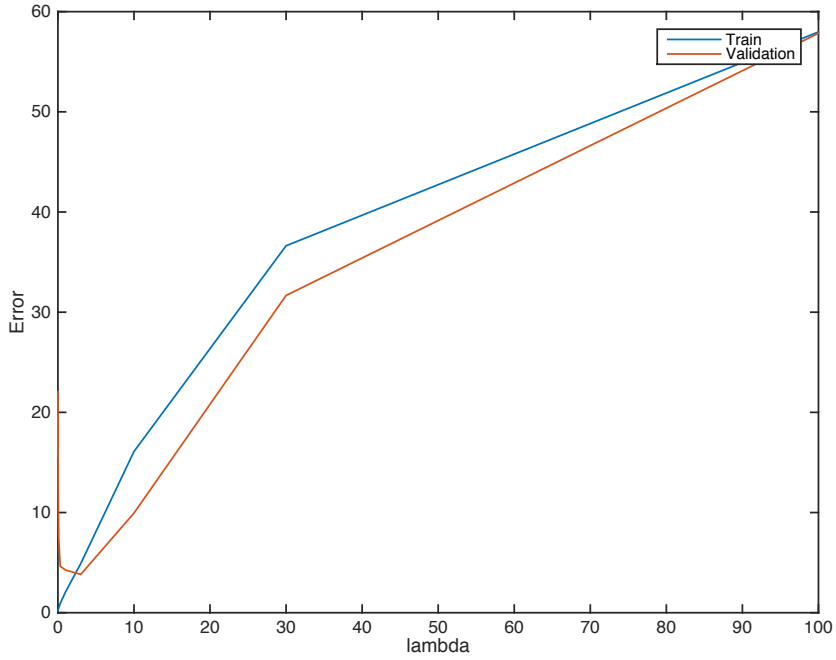

Polynomial Regression Learning Curve (lambda = 10.000000)
Train
Validation
Error
Number of training examples


Polynomial Regression Fit (lambda = 100.000000)
Water flowing out of the dam (y)
Change in water level (x)


Polynomial Regression Learning Curve (lambda = 100.000000)
Train
Validation
Error
Number of training examples

When the value of lambda approaches 0, the training results fit the training data very well. But the validation error is relatively high. After increasing lambda a little bit to 1, there is a good compromise between training and validation errors. But when lambda keeps increasing, the regularization parameter influences cost function so much that the training result cannot fit either training or validation data.

Selecting lambda using a validation set



We have the results as follow:

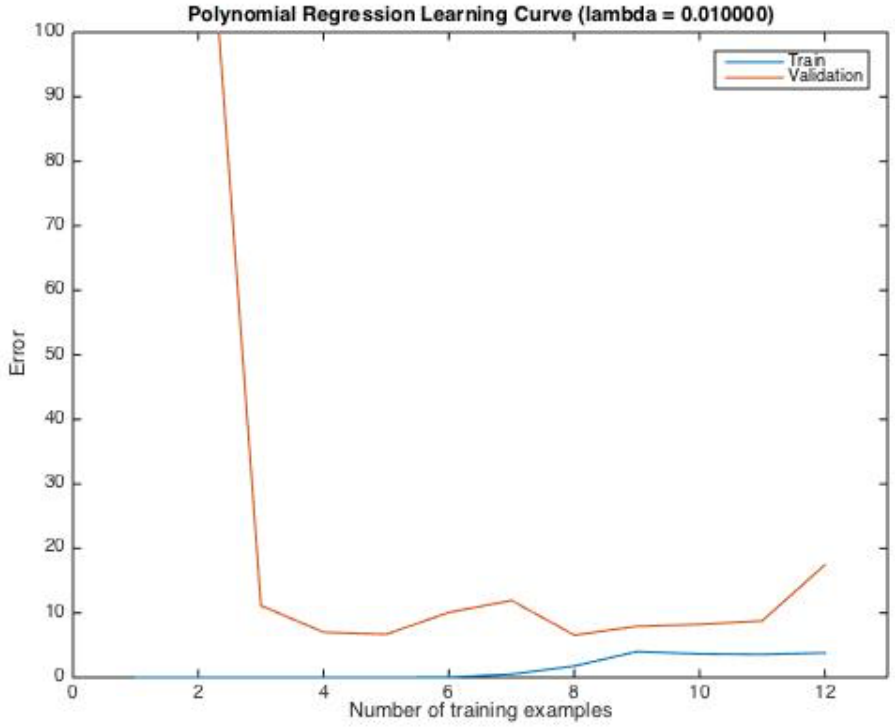
lambda	Train Error	Validation Error
0.000000	0.173616	22.066602
0.001000	0.135361	15.524064
0.003000	0.183285	19.815801
0.010000	0.222441	17.110508
0.030000	0.281850	12.831224
0.100000	0.459318	7.587013
0.300000	0.921760	4.636833
1.000000	2.076188	4.260625
3.000000	4.901351	3.822907
10.000000	16.092213	9.945508

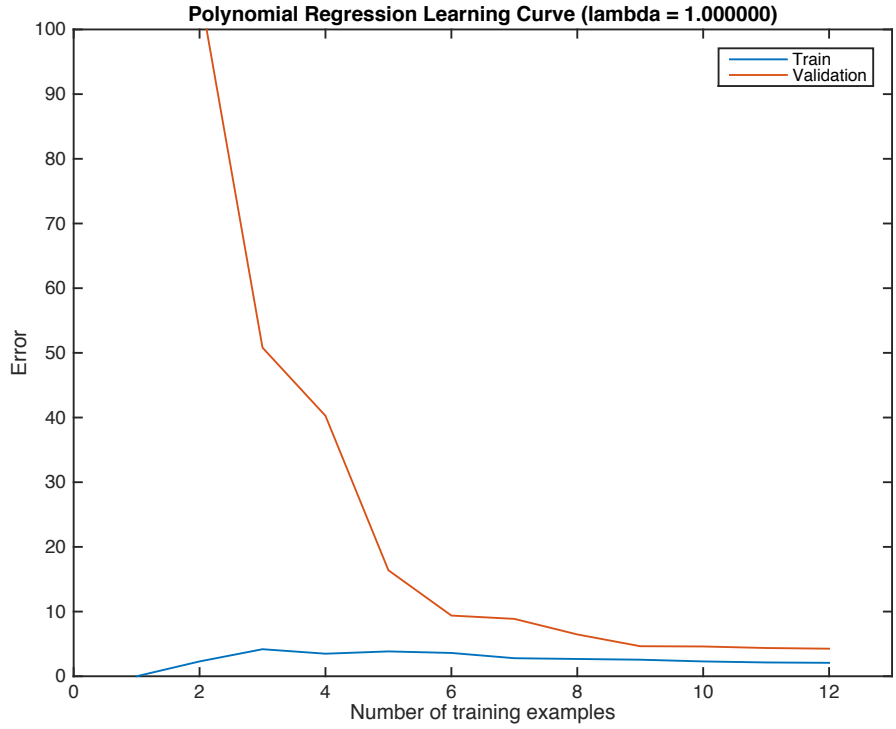
To compromise both train and validation error, we found that when lambda = 3 both errors remain in relatively low values. So we select lambda = 3.

Computing test set errors

test error:3.859888

Plotting learning curves with randomly selected examples



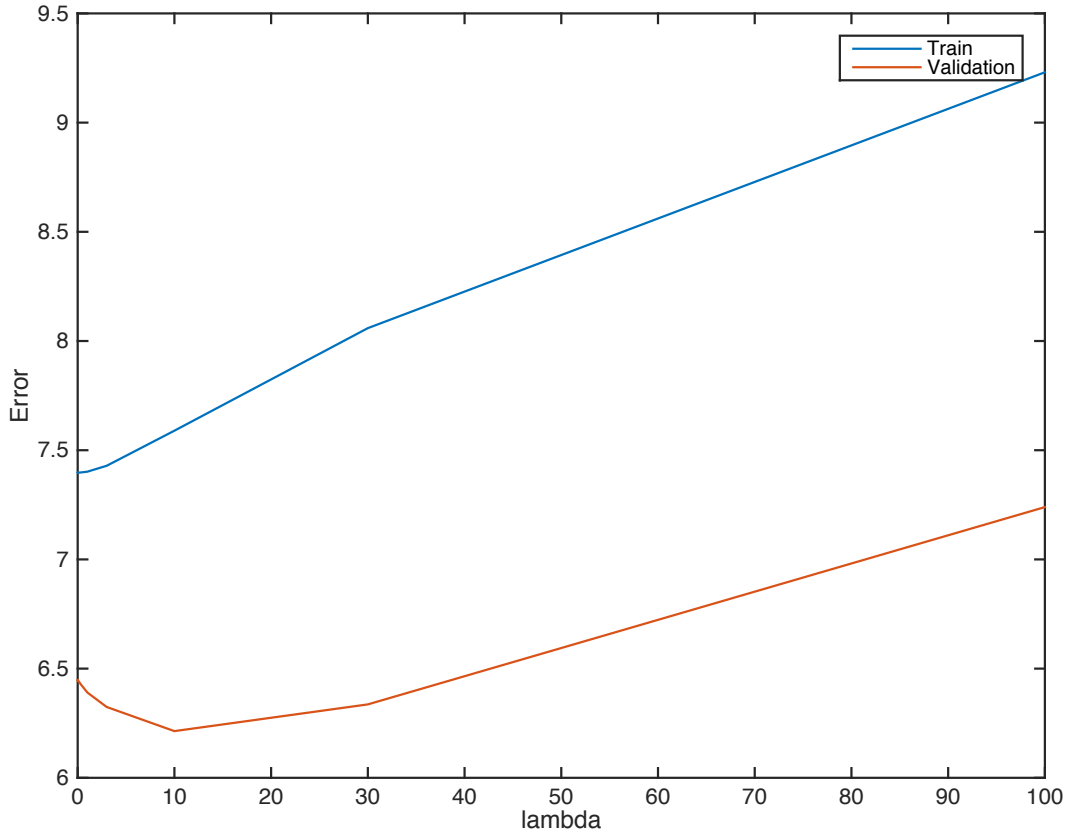


Problem 3

During the calculation, we found that the results may vary among different round of execution. So the following analyses and results are based on results from one round of execution.

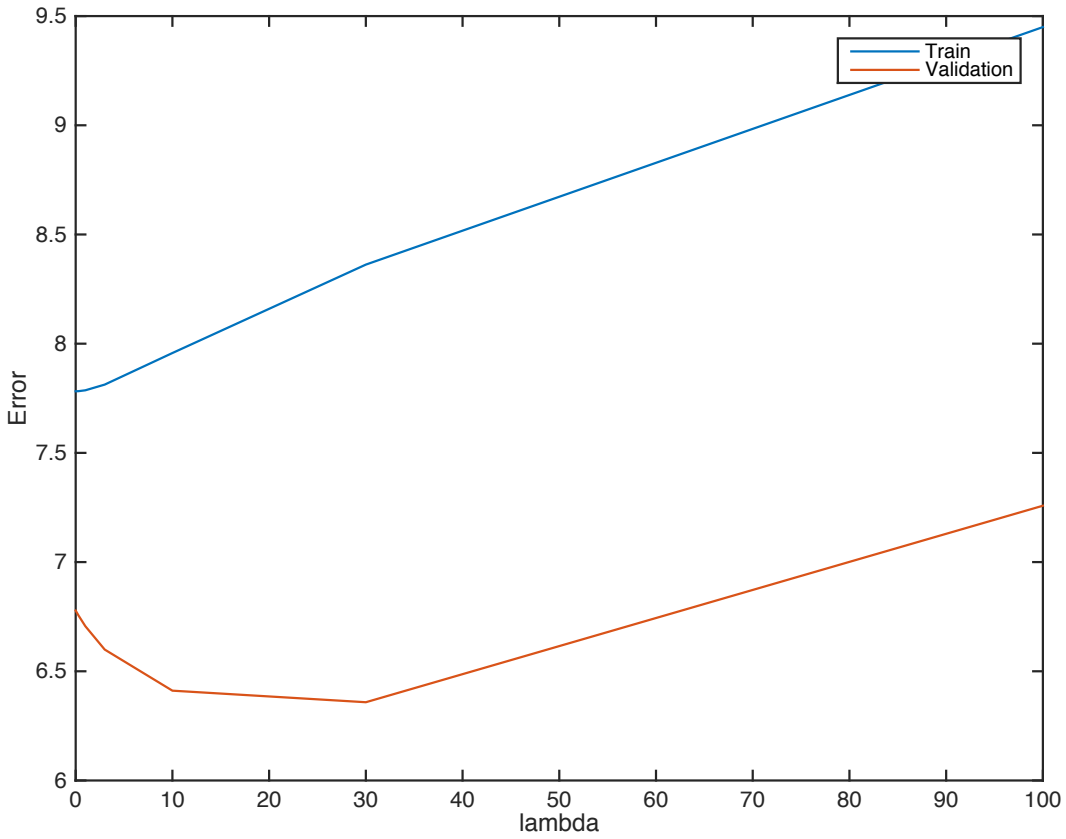
For the original data set, we have:

When lambda = 0, test error = 14.614076.



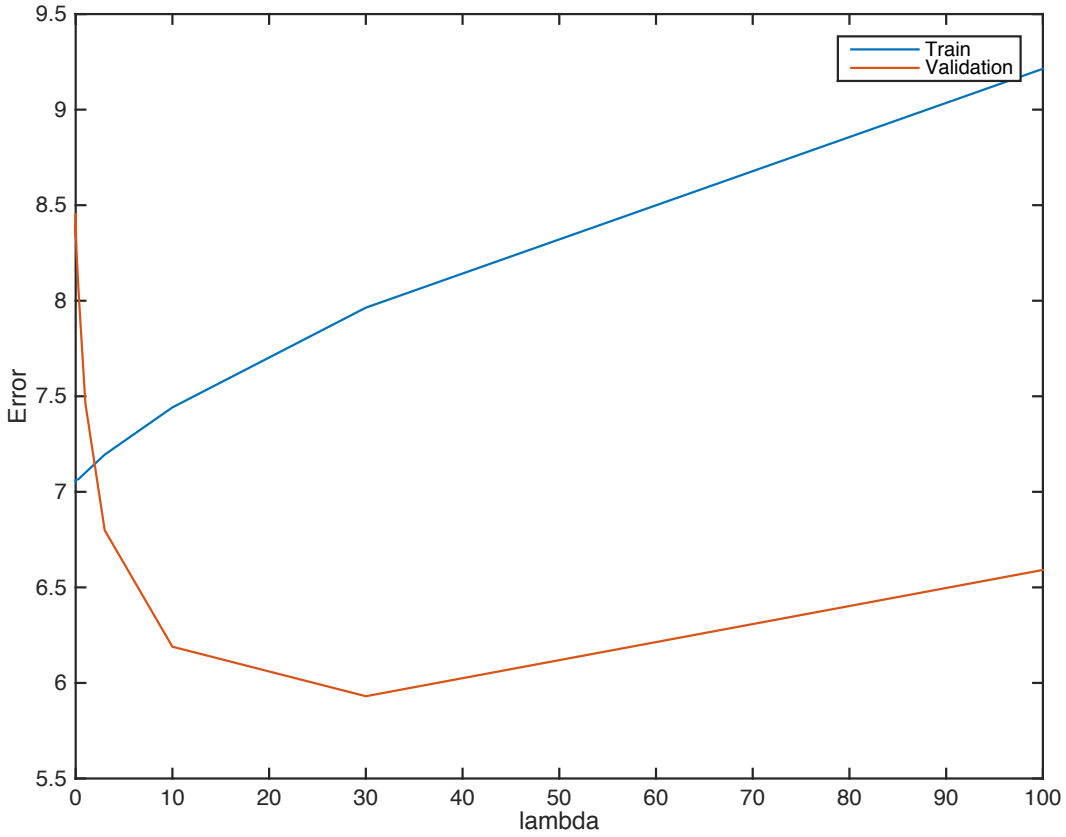
From the figure above we set the best value for lambda as 10, and the test error = 9.856734

For quadratic features, we have:



We set the best value for lambda as 30, and the test error = 6.617729

For cubic features, we have:



We set the best value for lambda as 30, and the test error = 6.490769

For this problem, we have 506 groups of data. If the we use 13 features to describe the relationship between features and housing price, it is hard to fit all the data. So the test error will be high. To fit such a large and complex data set, we may need more features. From the results of the program, we can find that by introducing lambda and technique of adding features, we can build a model to describe the data set better.

Also, the separation of data will influence the values of errors. In this problem setting too much data as training data will increase the training error and make the results unstable. So separating data carefully is very important for a good model.