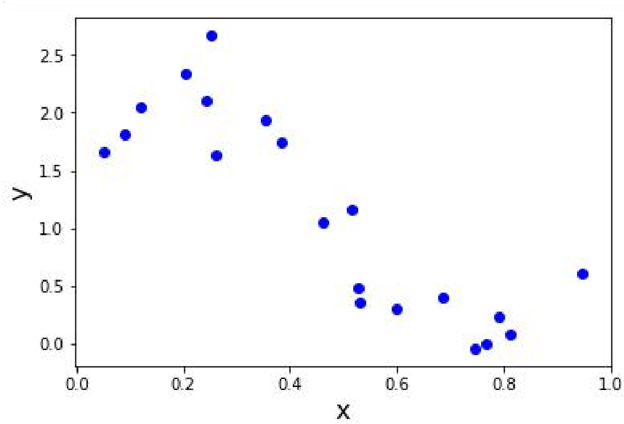
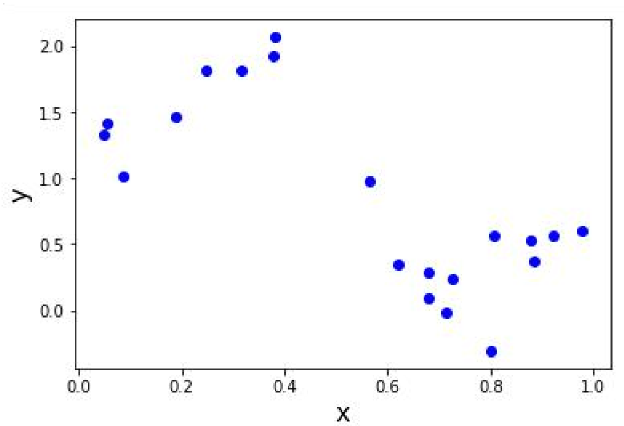
(a)

Training data:



Testing data:



From these two figures, we know that neither y for testing data nor y for training data can be predicted by a linear regression model. We need to train a polynomial regression model to get a better prediction.

(d)

η	W	# of iterations	cost
0.0001	[2.27044798, -2.46064834]	10000	4.086397
0.001	[2.44640678, -2.81635296]	7021	3.912576
0.01	[2.44640703, -2.81635347]	765	3.912576
0.0407	[-9.40470931e+18, -4.65229095e+18]	10000	2.7109e+39

When we choose η=0.01, after 765th iteration it converges to the final result, which is the quickest.

When we choose η=0.001, it will also converges after 7021 iterations, which is slower than when we use η=0.01.

When we use η=0.0001, since the step size is really small, the self coefficients decreases really slow and it needs much more iterations than 10000 to get the final result.

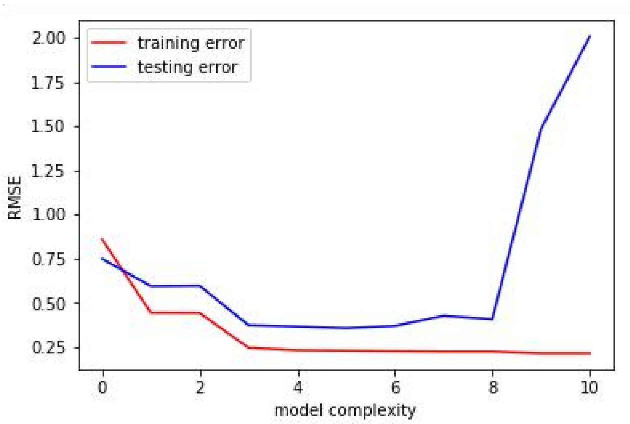
When we choose η=0.0407, since the step size is too big, self coefficients will overshooting the optimal value and will never converge.
From different step size we choose, we can conclude that choosing step size properly really matter a lot for gradient descent method.

(e)
When we use closed-form solution, W=[2.44640709, -2.81635359], and the cost is 3.9125764
The closed-form solution is faster since there is no iterations in this method.

(f)
When we use $\eta_k = \dfrac{1}{1+k}$ as the step size, W=[2.44640678, -2.81635296]. And it takes 1682 iterations to converge, which is slower than when we use η=0.01.

(h)
RMSE is better since it's normalized by the number of instances and it is the sample standard deviation. It is on the same scale with your data.

(i)



From the plot, 5th degree polynomial fits the data best since it has the lowest testing error and also a low training error. It's evident that when m=8,9,10 there is overfitting since it has low traning error but really high testing error.