CS5487 Program Assignment 1

——Regression

Ruyu Yan

闫如玉

55427256

Part Ⅰ polynomial function

1. Implement 5 regression algorithms for the K-th order polynomial given by



Where is the feature which transformed from the input vector variable. Then all of the calculation can be turn to about matrix and vectors. The aim of regression is to find the better parameter  vector in order to forecast the trend, and get the regression result of y (output of input x). Feature transformation is in PA1\_1b.m.The 5 kinds of regression algorithms is explained as following:

1. LS (Least Square)

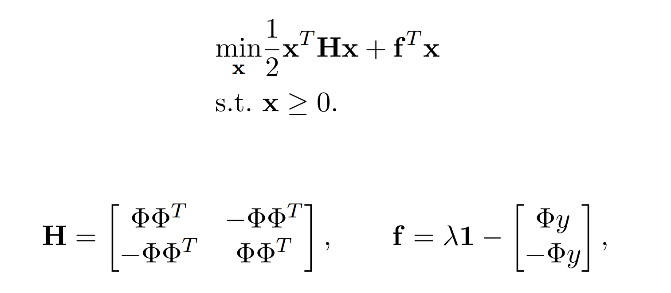
We can calculate  directly, and the function can be find in LS.m

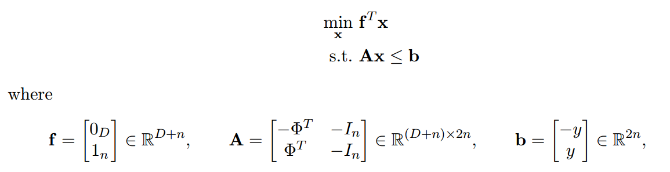
1. RLS (Regularized Square)

Calculate , and the function is RLS.m

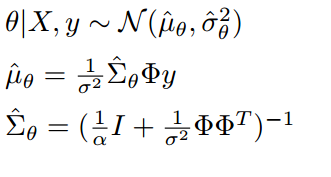
1. LASSO (L1-regularized LS)

From Prob. 3.12, the original regularization terms on the weights based on L1-norm can be rewritten in the standard form of a quadratic program(LASSO.m).

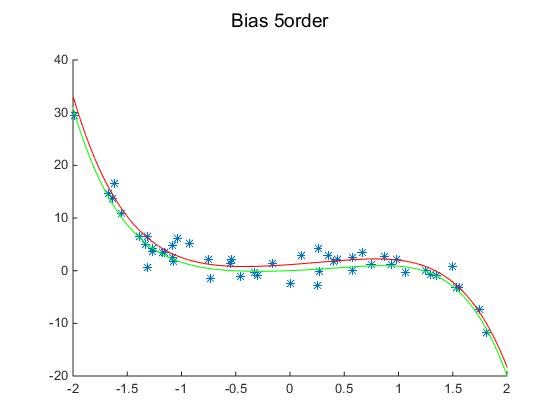
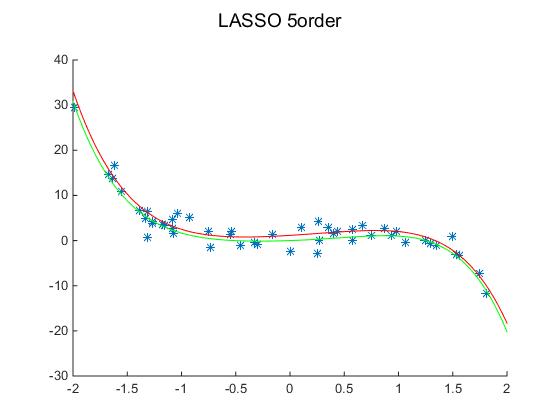
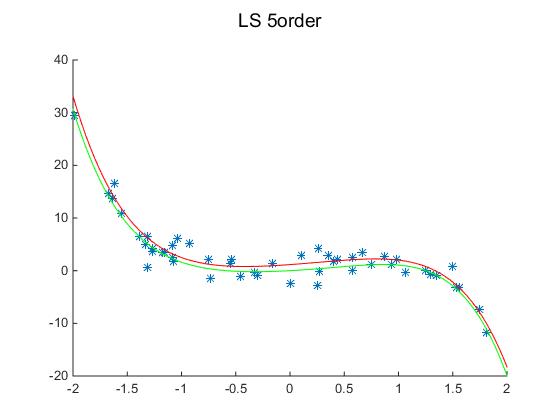
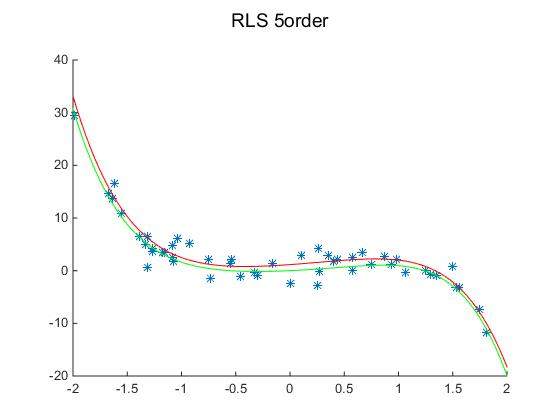
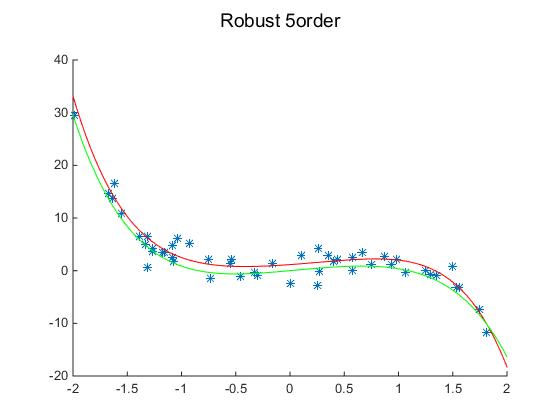
1. RR (Robust Regression)

From Prob. 2.10, the original regularization terms of L1-norm can be turned into a standard linear program(LASSO.m).

1. BR (Bayesian Regression)

From the regression algorithms, the predictive of parameter is a distribution, and the mean value is what we need as . Besides ,the variance  can be seen as the variation range of predictive function. They can calculate by(Bias.m) :

1. For each regression method, use the sample data (sampx, sampy) to estimate the parameters of a 5th order polynomial function, and the result is showed in the following picture(the true is the red line and predictive function is green line).

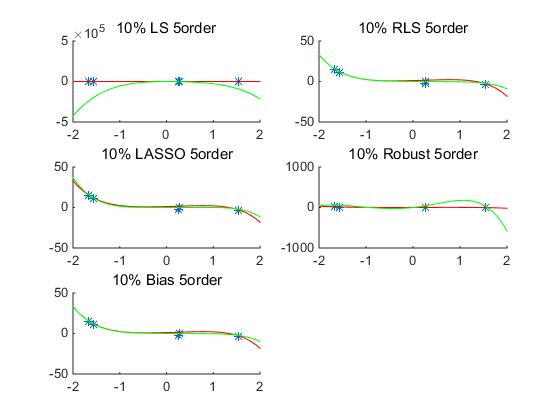
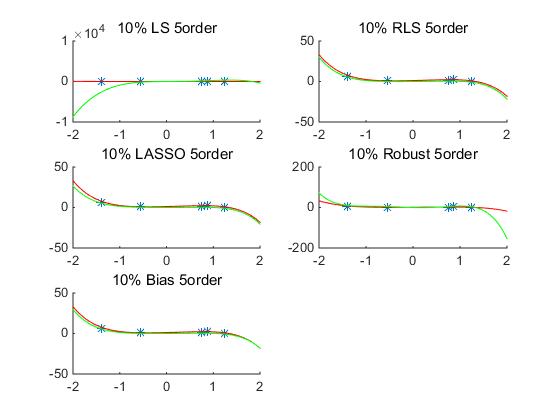


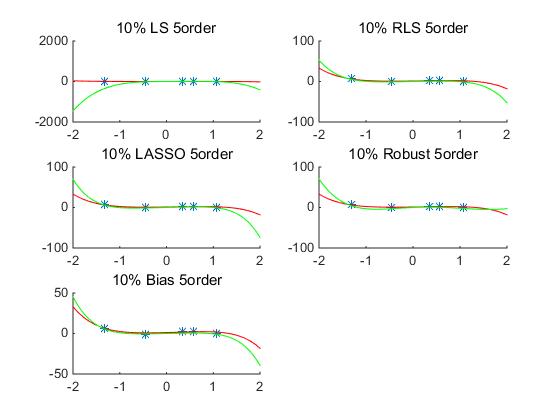
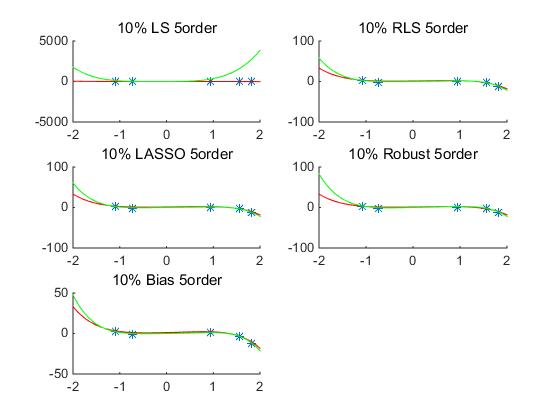
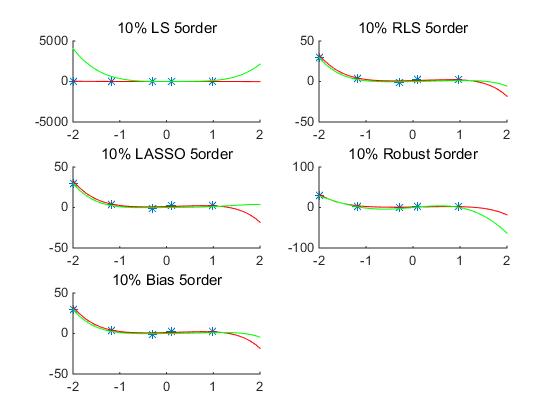
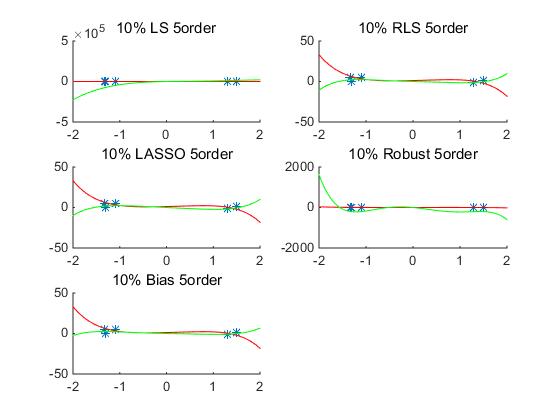
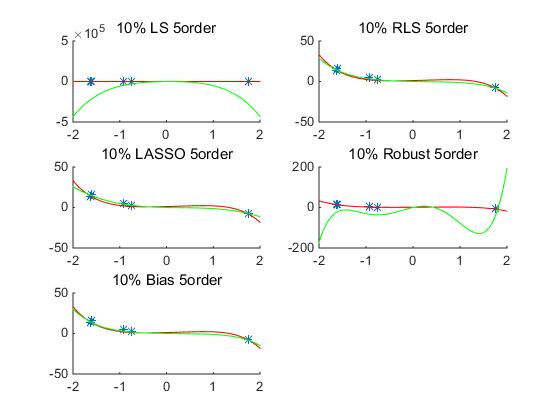
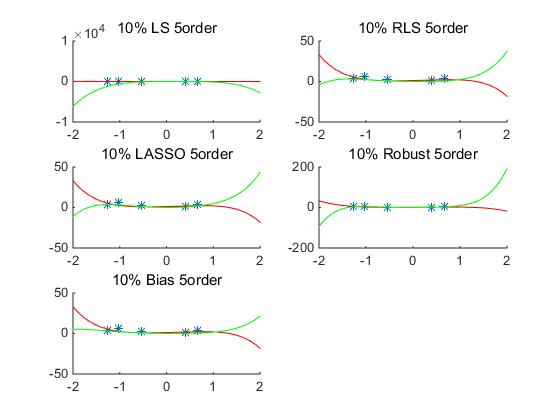
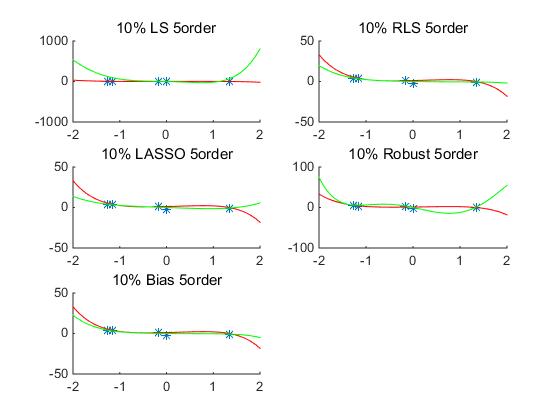
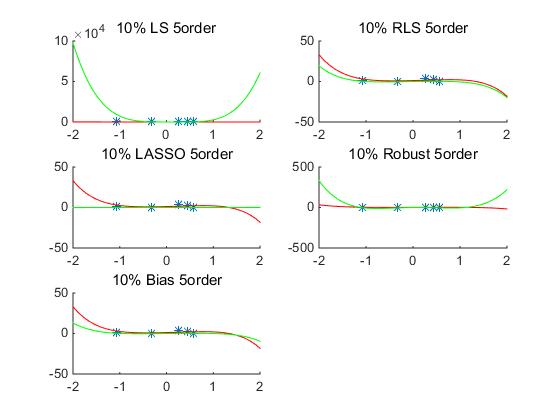
The mean-square errors between the learned function and the true function outputs of each algorithm are:

ms\_LS=1.4291; ms\_RLS=1.5135;ms\_LASSO=1.4782;ms\_Robust=2.6608;ms\_Bias=1.5796

From above, the mean-square errors of LS ,RLS, LASSO, and Bias are close, that means the four algorithm can get the same result in this training set.

1. Repeat (b), but reduce the amount of training data available, by selecting a subset of the samples .
2. 10% :I selected the data in average which means that from the first data, with spacing is ten, to the end of the last one of data. Then there are ten picture by different average data set.





Choose an set in Arbitrary (The last one), which mean-square is

mse\_LS = 1.797703885604259e+09

mse\_RLS = 30.657908739312518

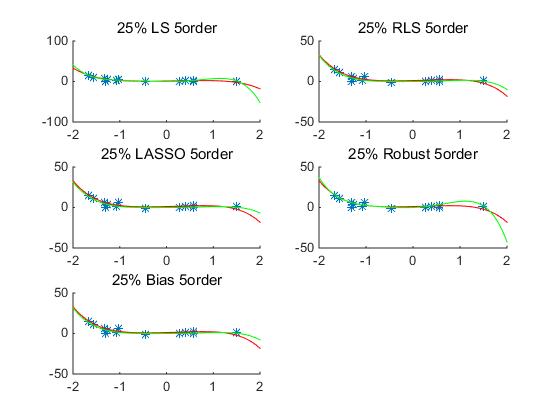
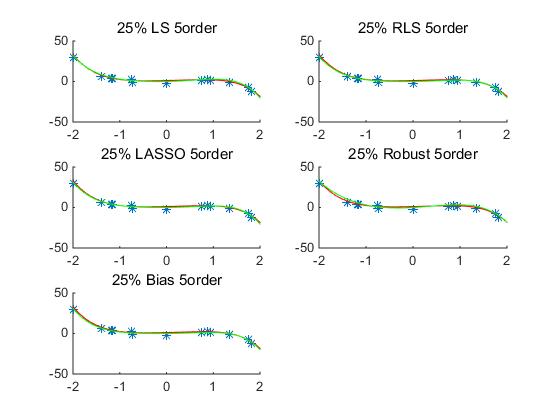
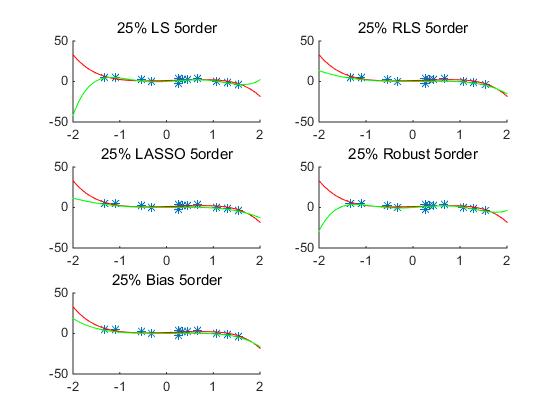
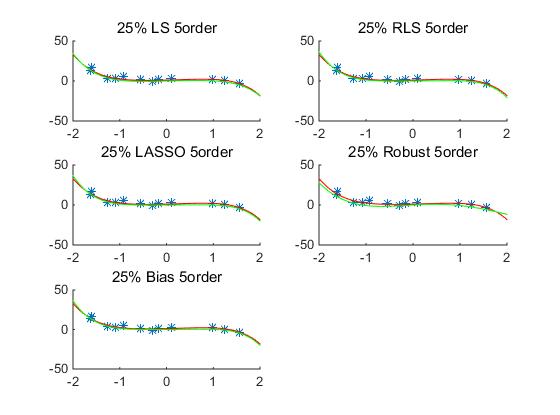
mse\_LASSO = 39.069799496542416

mse\_RR = 1.605140020001430e+04

mse\_BR = 30.657908739312518

From the above picture, RLS and Bias tend to be less error, oppositly, LS is no robust enough and Robust will over-fitting sometimes.

1. 25% in average (same select principle as 10%)



The Choose an set in Arbitrary (The last one), which mean-square is

mse\_LS = 70.153277000795870

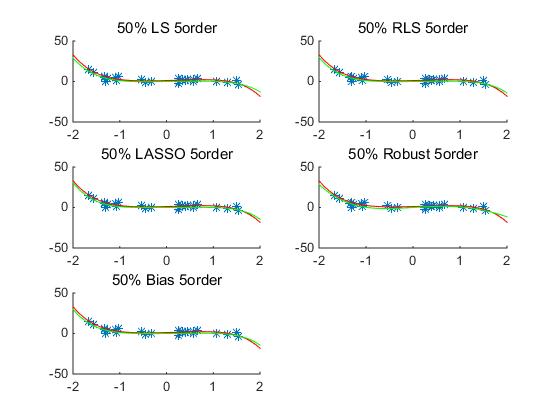
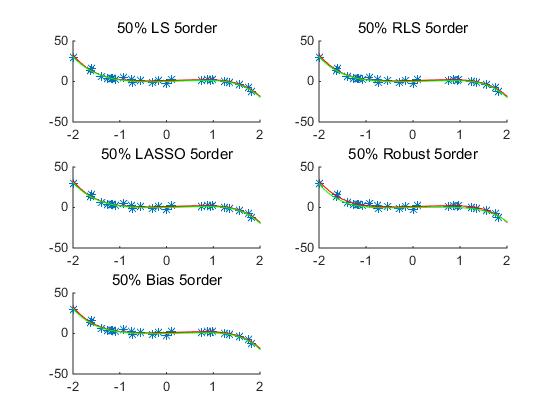
mse\_RLS = 4.647013016673829

mse\_LASSO = 5.777384507341600

mse\_RR = 47.150214353632016

mse\_BR = 4.647013016674033

1. 50% in average(same select principle as 10%)



The Choose an set in Arbitrary (The last one), which mean-square is

mse\_LS = 1.478958888497502

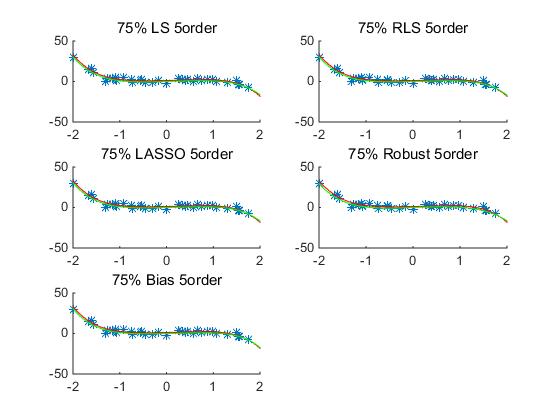
mse\_RLS = 1.102741313794592

mse\_LASSO = 1.276431503084854

mse\_RR = 1.856753697118480

mse\_BR = 1.102741313794367

1. 75% :I select the data from the 3th to



The Choose an set in Arbitrary (The last one), which mean-square is:

mse\_LS = 0.704311761067983

mse\_RLS = 0.560956677060883

mse\_LASSO = 0.567560841195825

mse\_RR = 0.727294660016577

mse\_BR = 0.560956677060892

When the size reach 75% of the data sets, the mean-squared errors already very close to the optimal error . Still RR > LS > LASSO > RLS > BR.

And we can find when data size is above 50%. There are very few difference beyond several regressions result. And when the data is small enough , LS and RLS have more uncertain trend based on data sets.

1. Add some outliers output values:

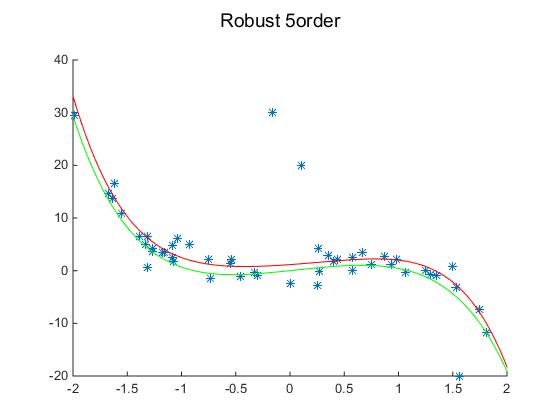
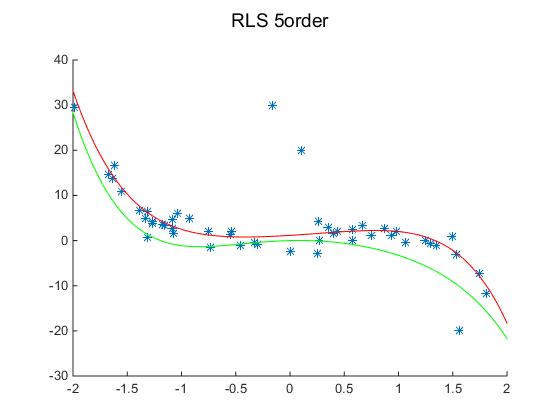
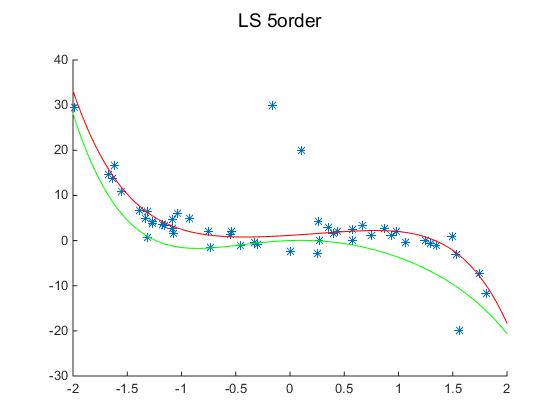
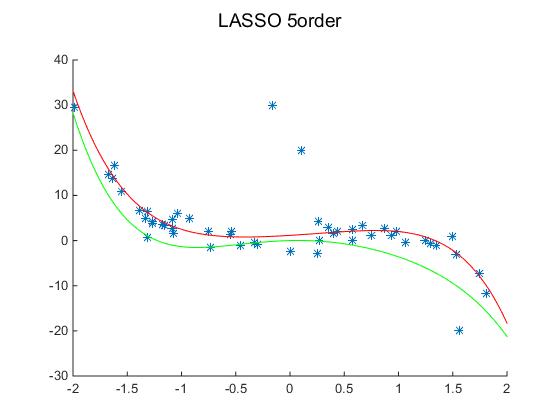
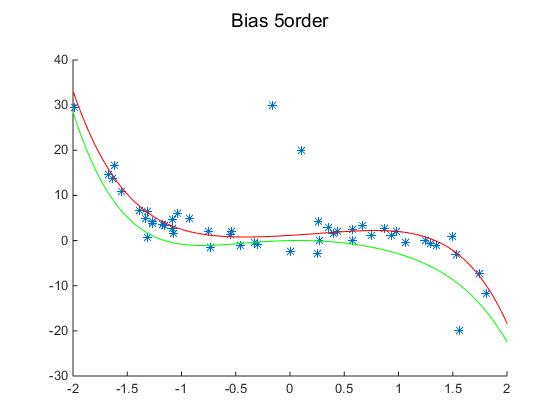
sampy\_out = sampy;

sampy\_out(12) = -20;

sampy\_out(24) = 5;

sampy\_out(36) = 30;

sampy\_out(48) = 20;



mse\_LS = 2.510650586343948

mse\_RLS = 2.015654141649858

mse\_LASSO = 2.243471687777492

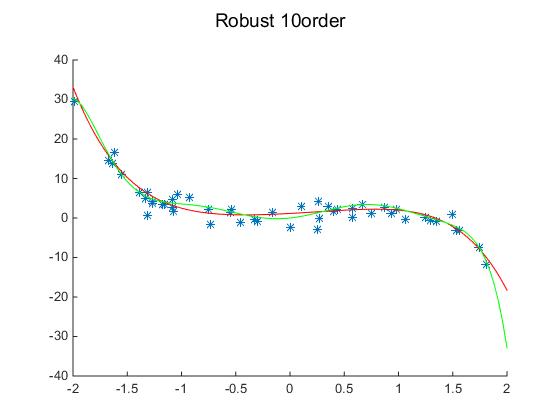
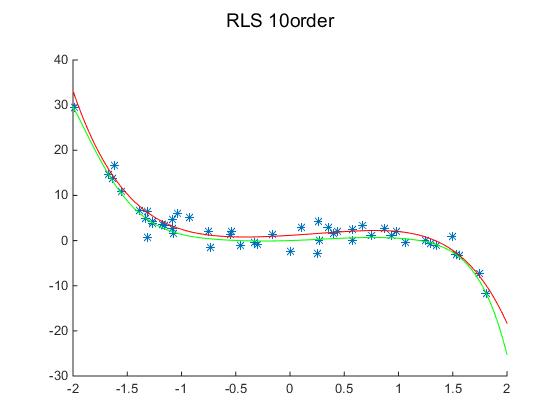
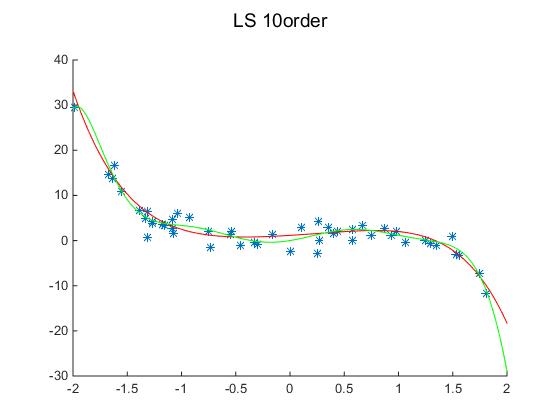
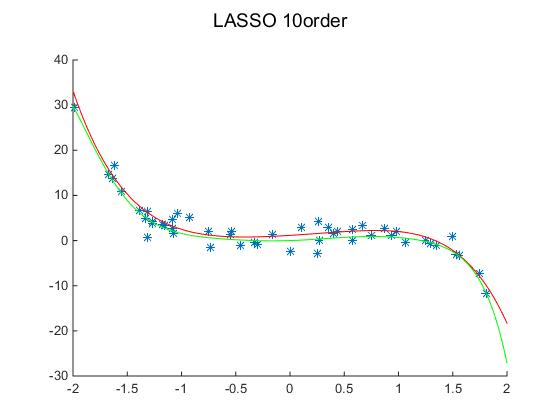
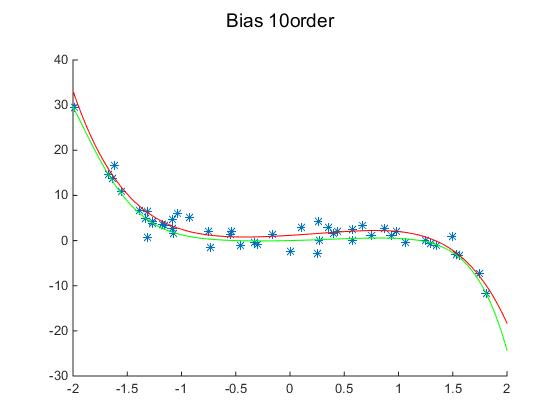
mse\_RR = 0.588367365640390

mse\_BR = 2.015654141649540

We can find that The Robust is much more robust than all other 4 methods. However, the difference between the four other methods is not so big, where LASSO is the least robust to the outliers.

Reason: I think the Robust didn’t use the squared difference between predicted value and true value, while others are all based on the squared value.

1. Repeat (b) but estimate a higher-order polynomial (e.g., 10th order).



mse\_LS = 7.983120896948182

mse\_RLS = 6.54809618840197

mse\_LASSO = 6.141728517181902

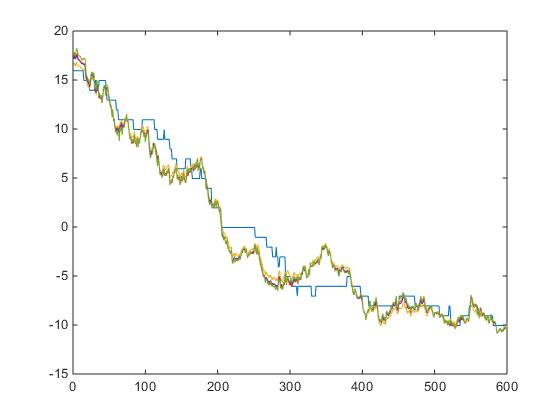
mse\_RR = 1.289856786131285

mse\_BR = 6.548096165217130

From picture we can find, the method of LS,LASSO and Robust tend to be overfitted.

Part Ⅱ A real world regression problem – counting people

1. Let’s ﬁrst look at using the features directly, i.e., set φ(x) = x. In following picture, I combine the five lines on one table in order to catch the difference between them.



mae\_LS = 1.358443521146659

mae\_RLS = 1.279469208496071

mae\_LASSO = 1.324740921426673

mae\_RR = 1.365140970365960

mae\_BR = 1.282432855732872

mse\_LS = 3.102838014135047

mse\_RLS = 2.650371501037698

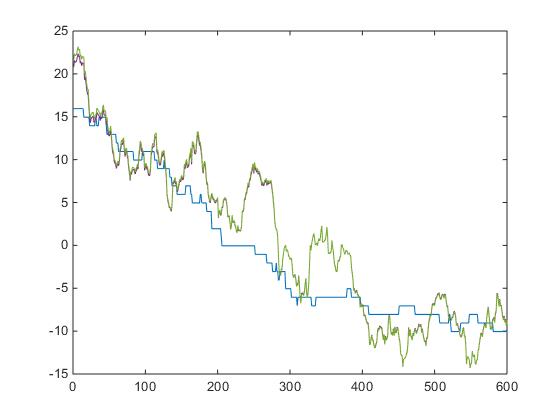
mse\_LASSO = 2.928052624145086

mse\_RR = 3.122047966381001

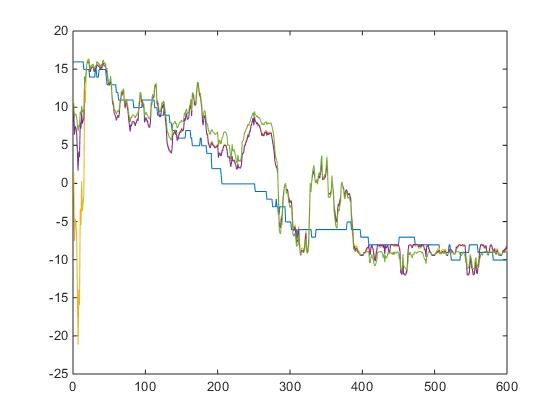
mse\_BR = 2.618733922241927

The result orders from good to bad is: RLS,Bias,LASSO,LS,Robust.

1. Now try some other feature transformations. For example, you can create a simple 2nd order: we can find the picture tend to be



If we change to do the 10 order, there will be:



Whatever we choose the number of order, the results will be over fitted, maybe the polynomial non-linear feature may not better than linear feature in true life.