

Report for Week Three: Polynomial Fitting and Model Identification using Linear Regression

F221287

Week starting 19/02/24

1 Introduction

Provided with a dataset y which corresponds to x with the polynomial function

- $y_i = f_{poly}(x_i) + \eta_i$

where η is a noise variable. Using different losses and regressions, I intend to find the model which comes closest to fitting the data.

2 Implementation

I first started with splitting the dataset into Test and Train.

```
%This splits the data into Train and Test
ind = randperm(1000);
w3XTestOne = x(ind(1:300),:);
w3XTrainOne = x(ind(301:end),:);
w3YTestOne = y(ind(1:300),:);
w3YTrainOne = y(ind(301:end),:);
```

Then with a set of different loss functions

```

function yL0 = lZeroNorm(A,x,y)
    yHatL0 = polyval(A,x);
    yL0 = max(abs(yHatL0 - y));
end

function yL1 = lOneNorm(A,x,y)
    yHatL1 = polyval(A,x);
    yL1 = mean(abs(yHatL1 - y));
end

function yL2 = lTwoNorm(A,x,y)
    yHatL2 = polyval(A,x);
    yL2 = mean(abs(yHatL2 - y).^2);
end

```

I created a new set of coefficients

```

coefL0 = fminsearch(@(A) lZeroNorm(A,w3XTestOne,w3YTestOne),rand(D,1));
coefL1 = fminsearch(@(A) lOneNorm(A,w3XTestOne,w3YTestOne),rand(D,1));
coefL2 = fminsearch(@(A) lTwoNorm(A,w3XTestOne,w3YTestOne),rand(D,1));
%coefLasso = fminsearch(@(A) lasso(w3XTestOne,w3YTestOne),rand(D,1));
%Ridge works very well. Lasso not as much.
coefRidge = fminsearch(@(A) ridge(w3YTestOne,w3XTestOne,k),rand(D,1));

```

And used them to create predictions of Y

```

yHatL0TestOne = polyval(coefL0, w3XTestOne);
yHatL1TestOne = polyval(coefL1, w3XTestOne);
yHatL2TestOne = polyval(coefL2, w3XTestOne);
yHatRidgeTestOne = polyval(coefRidge, w3XTestOne);

yHatL0TrainOne = polyval(coefL0, w3XTrainOne);
yHatL1TrainOne = polyval(coefL1, w3XTrainOne);
yHatL2TrainOne = polyval(coefL2, w3XTrainOne);
yHatRidgeTrainOne = polyval(coefRidge, w3XTrainOne);

```

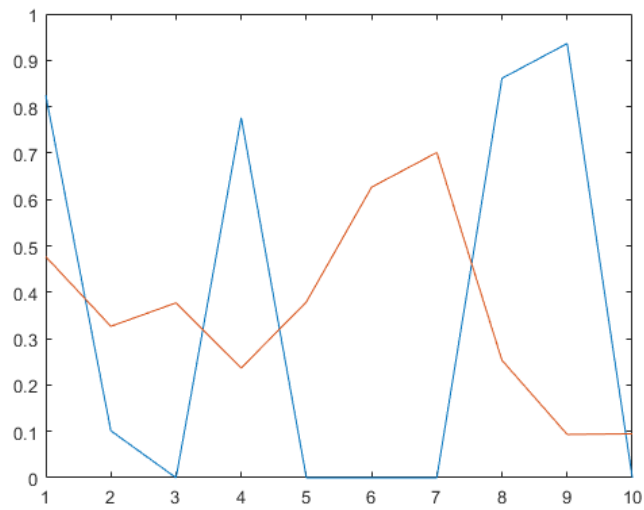
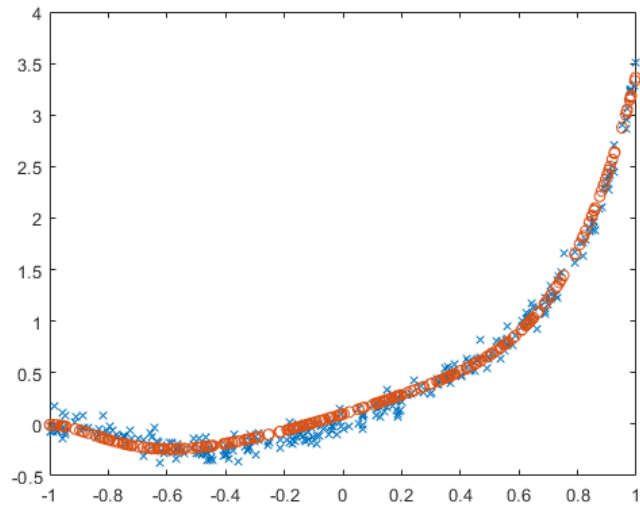
This was the easiest to implement of all the challenges so far, due to the code being this straightforward.

3 Results

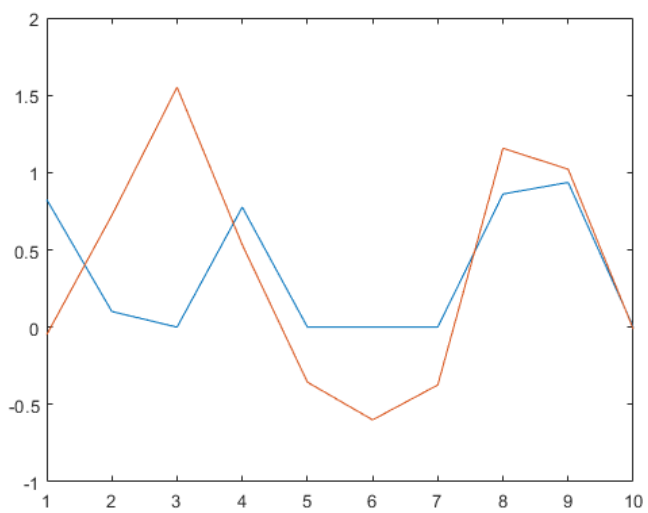
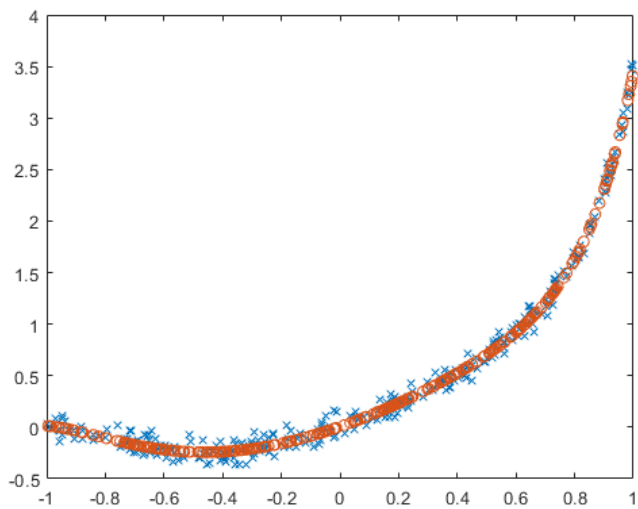
For each dataset, I compared the resulting graphs for each loss function against each other, determining which seemed to fit the dataset best, and for the first

two, where the original coefficients were provided, I compared the loss coefficients against the original to determine even further a better match.

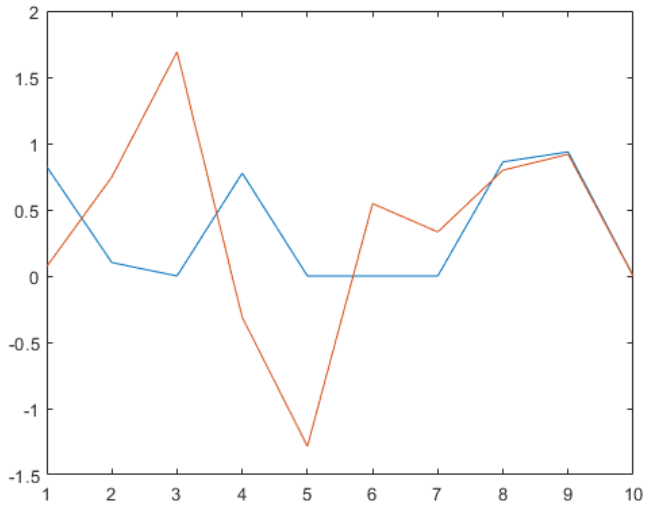
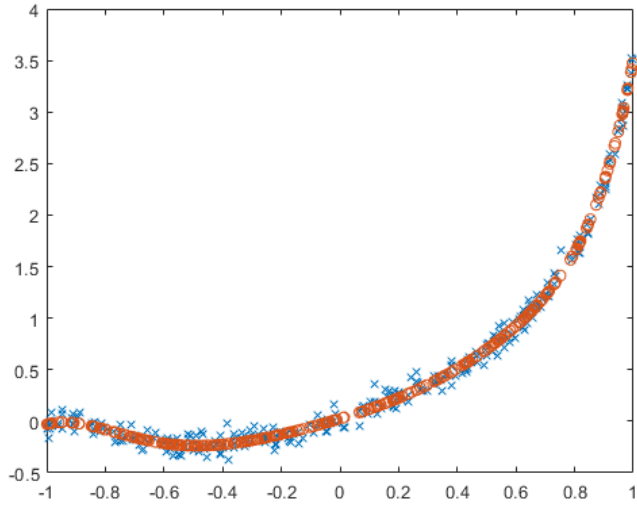
L0



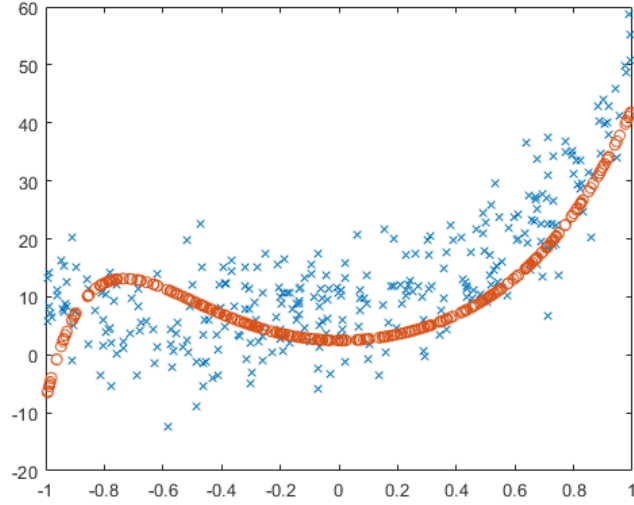
L1



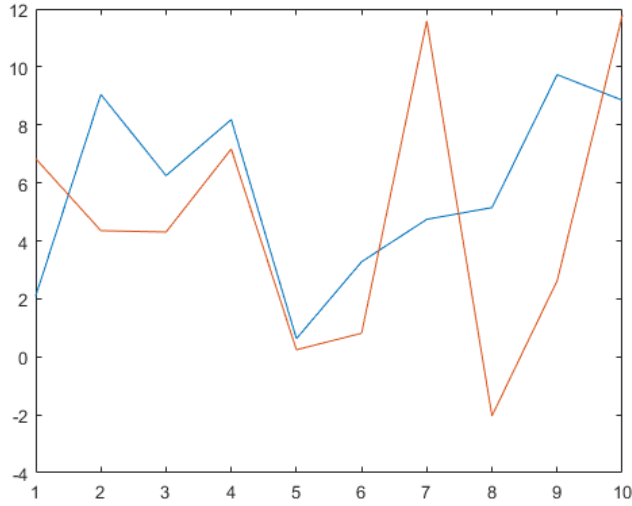
L2



Based on the graphs, one would seem to prefer the LTwo Norm for the polynomial fit, however, LOne's graph isn't far off, and seeing the coefficients, they match a lot closer than LTwo's, following a similar pattern but delayed or too early in some spots.

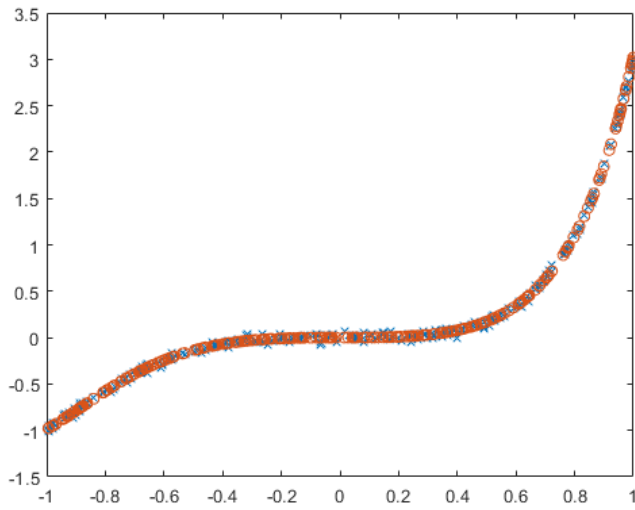
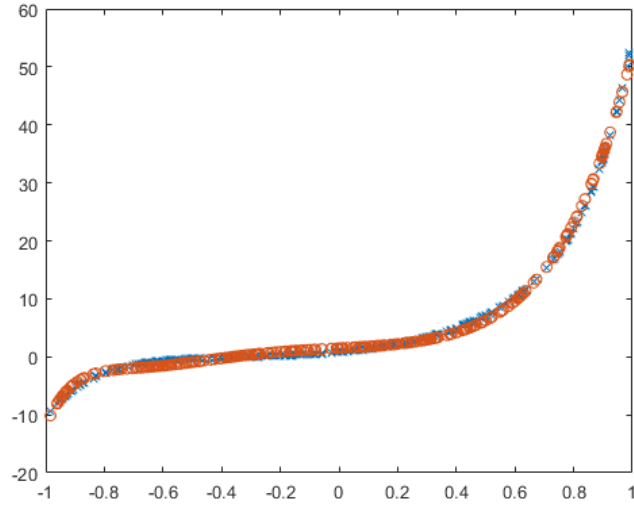


With the second dataset, the heavy noise can at first make it harder to determine which loss may match from the graphs. So, for these we rely heavier on the comparison of coefficients.



In this case, the coefficients of LZero seem to be the closest fit¹. These correspond to the graph above.

¹All of the graphs are found in the appendix.



Looking further onto sets three and four, the LTwo Norm again seems to have the best fit.

It would seem, when it comes to noisier datasets, such as dataset two, performing the LZero Norm returns the best predictions. Otherwise, LTwo is very much the go to.

4 Difficulties

Lasso for some reason was proving difficult to implement despite working similar in other programs so I went on without it.

Partway through implementation, I realised the MatLab on my computer at home didn't have the right toolbox for Lasso or Ridge regression, and downloading said toolbox proved near impossible with the mountain of errors whenever I tried. After a night of deliberation, I decided to delete and redownload MatLab, which stole a lot of time I could've spent working on this.

Other than these issues, this task was one that I found myself working through very quickly very well.

5 Appendix

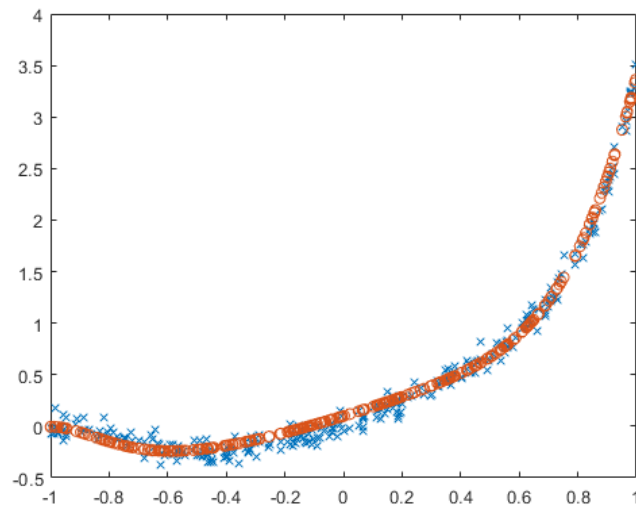


Figure 1: Dataset One LZero Norm

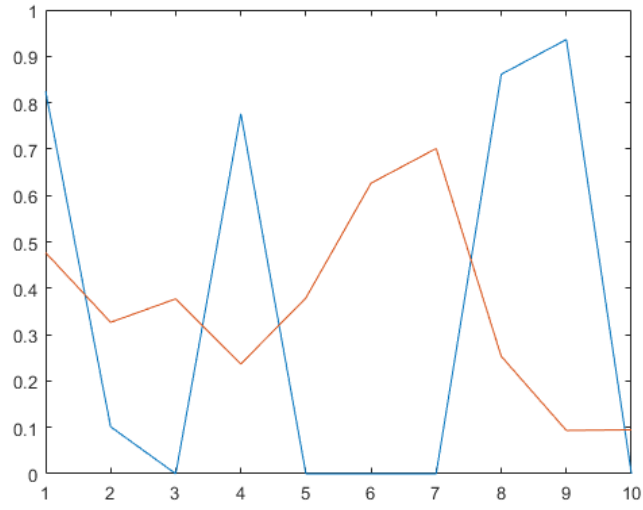


Figure 2: Dataset One LZZero Coefficients

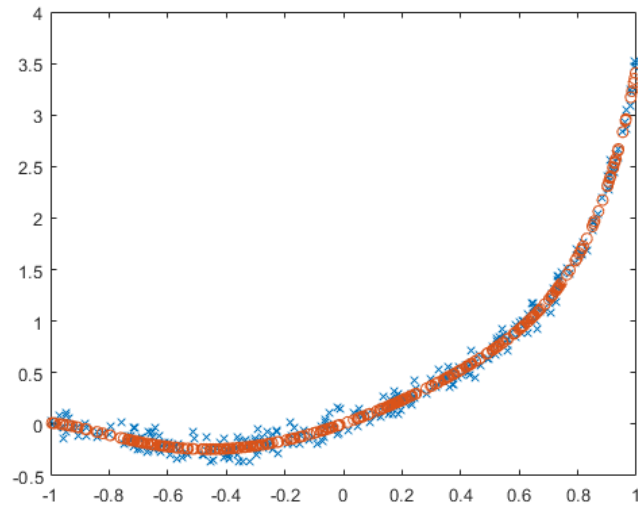


Figure 3: Dataset One LOne Norm

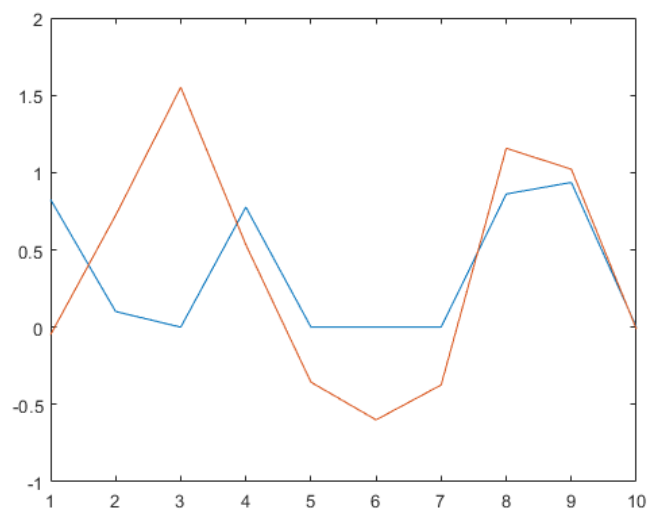


Figure 4: Dataset One LOne Coefficients

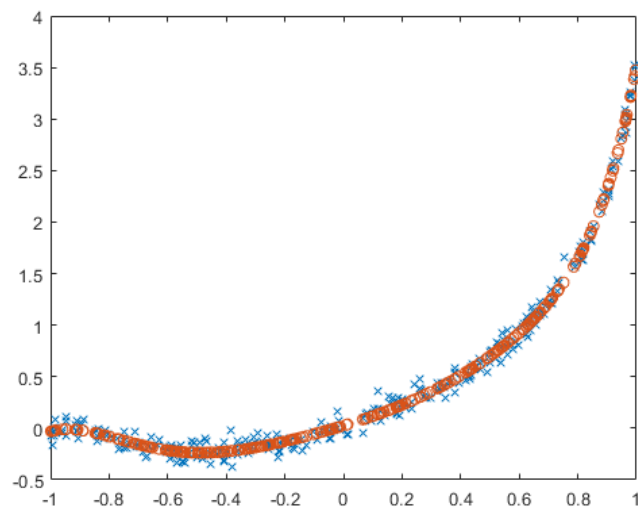


Figure 5: Dataset One LTwo Norm

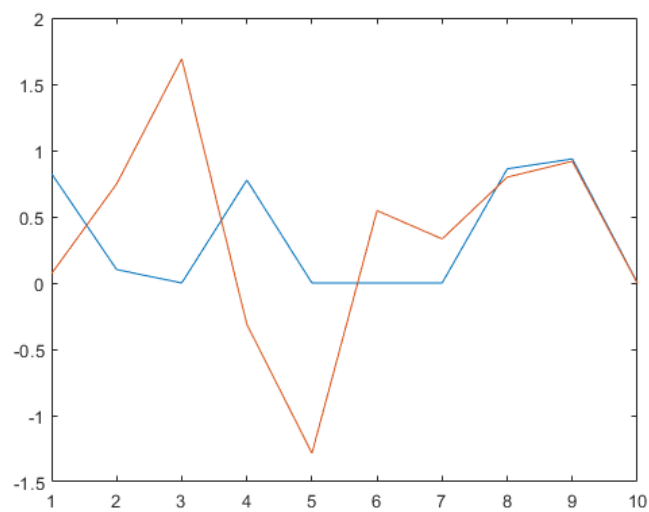


Figure 6: Dataset One LTwo Coefficients

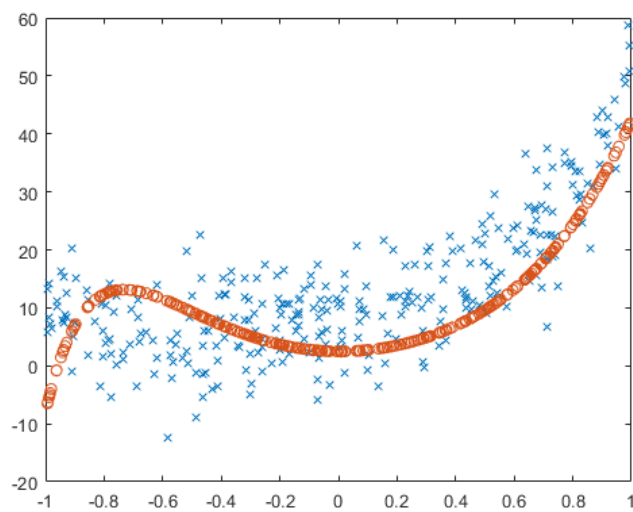


Figure 7: Dataset Two LZZero Norm

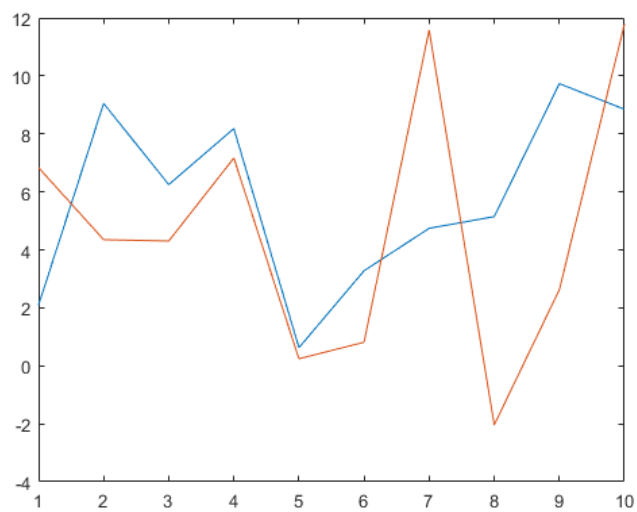


Figure 8: Dataset Two LZero Coefficients

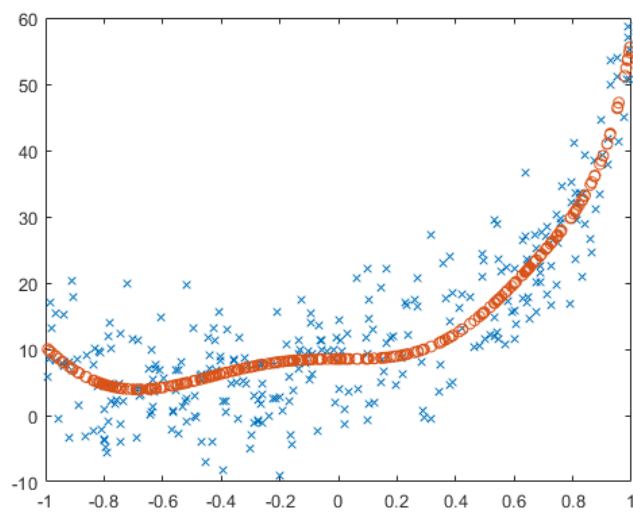


Figure 9: Dataset Two LOne Norm

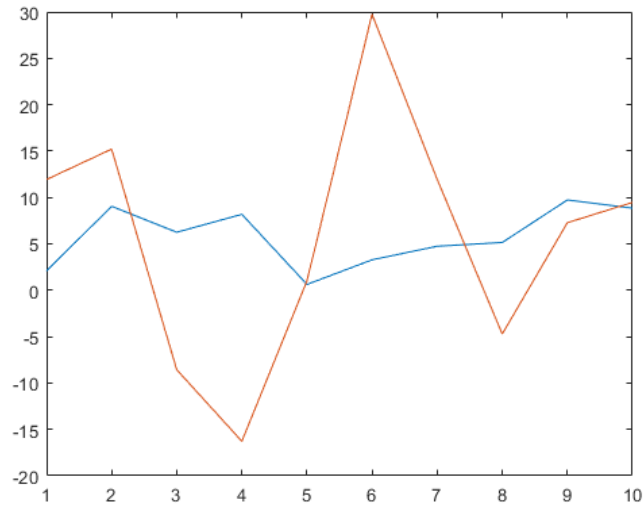


Figure 10: Dataset Two LOne Coefficients

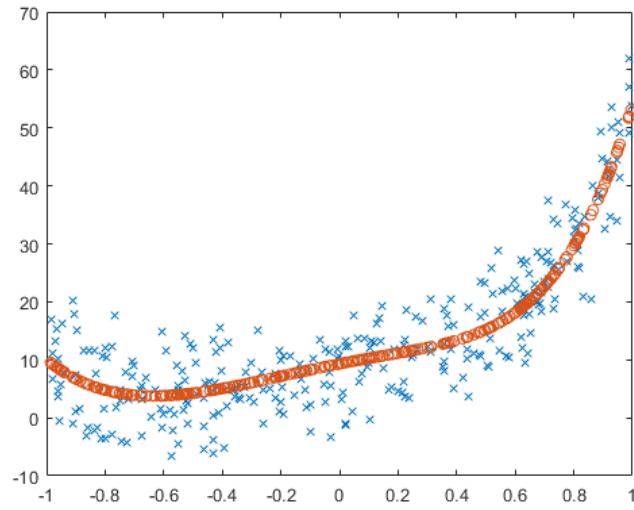


Figure 11: Dataset Two LTwo Norm

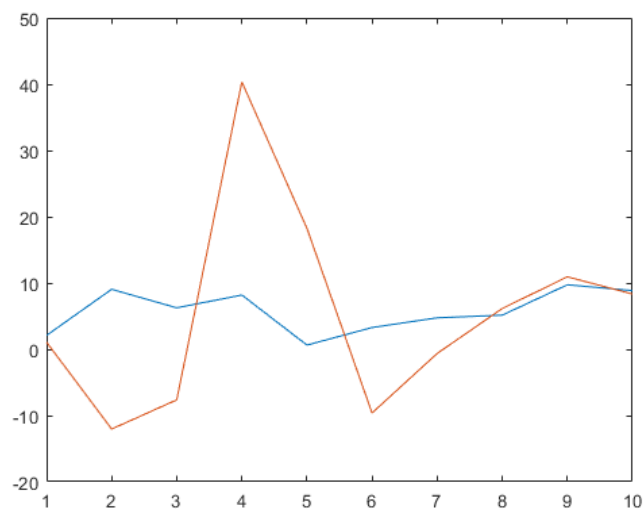


Figure 12: Dataset Two LTwo Coefficients

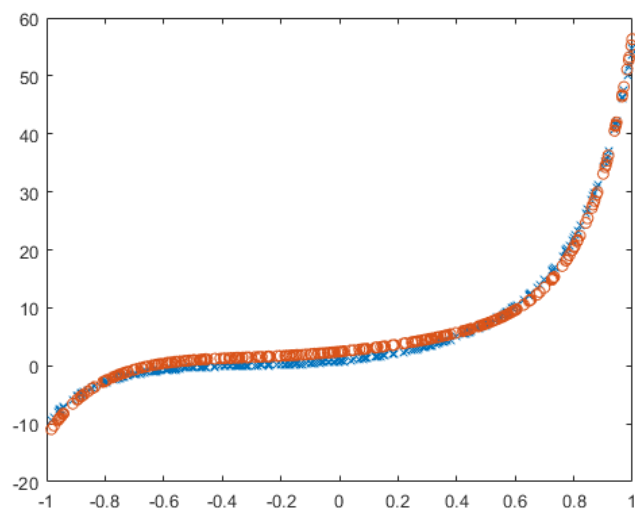


Figure 13: Dataset Three LZero Norm

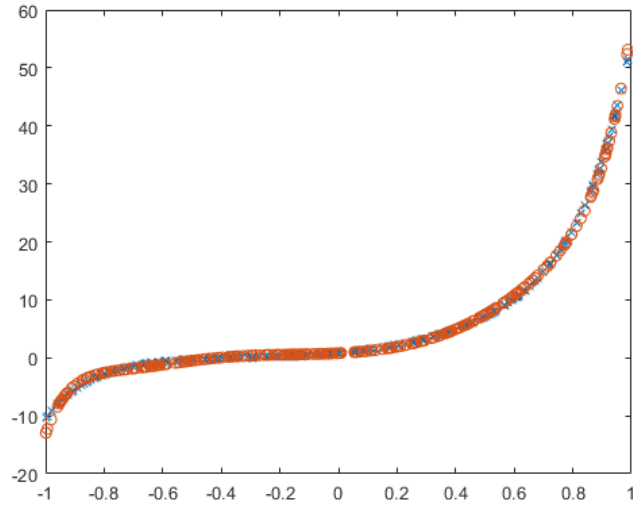


Figure 14: Dataset Three LOne Norm

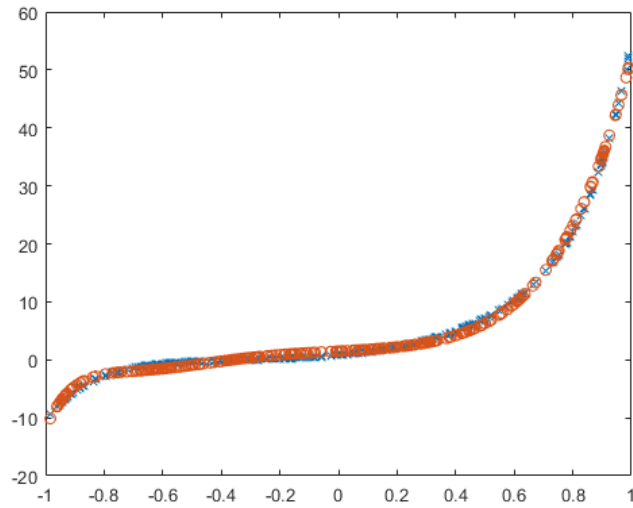


Figure 15: Dataset Three LTwo Norm

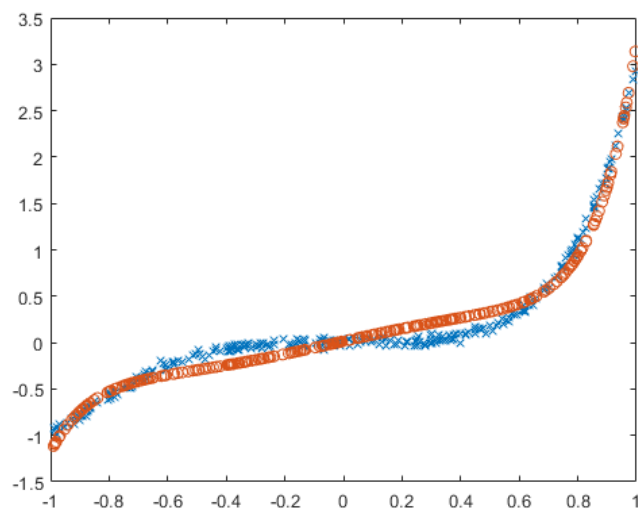


Figure 16: Dataset Four LZero Norm

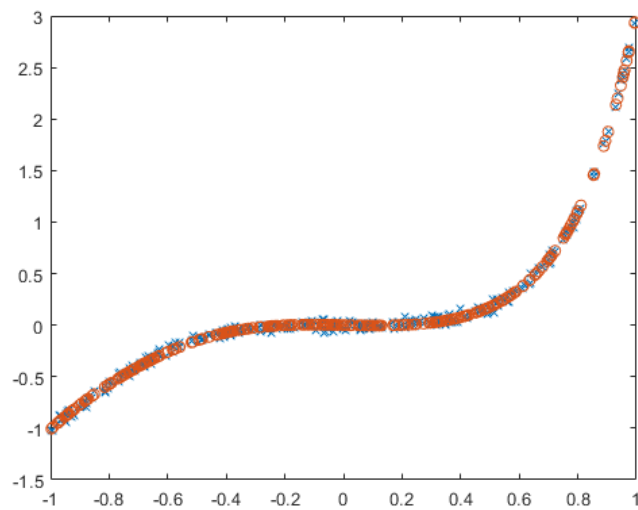


Figure 17: Dataset Four LOne Norm

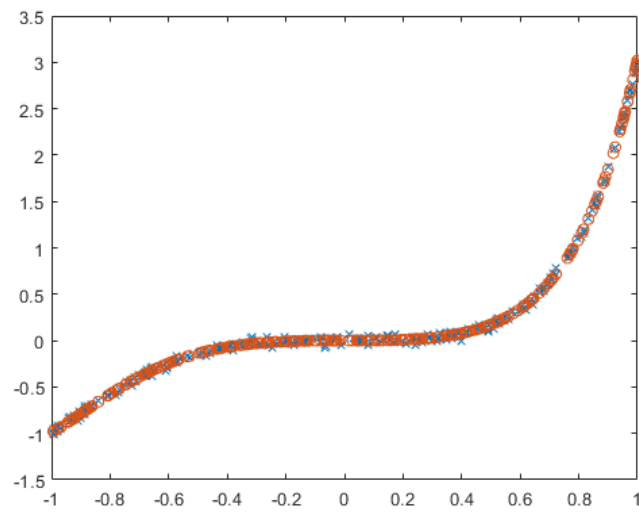


Figure 18: Dataset Four LTwo Norm