CS6316: HW2

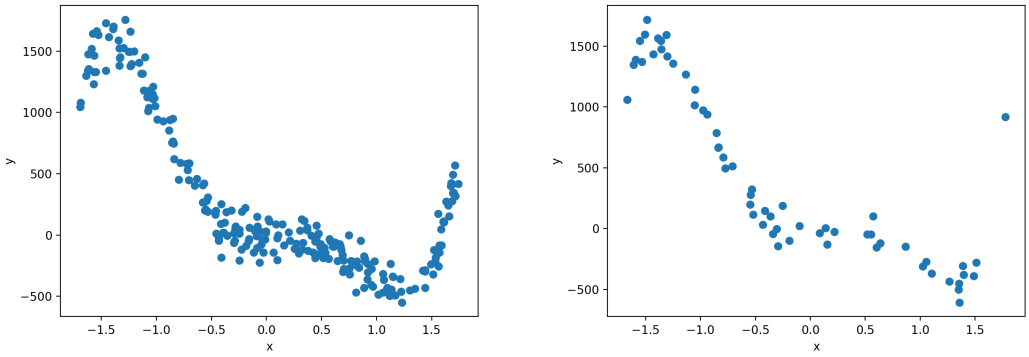
Aobo Yang (ay6gv)

October 1, 2019

1. Polynomial Regression

   1.1 Data Generation

   The generated data distribution is shown in Figure 1.



(a) Training (b) Testing

Figure 1: Gradient Descent

   1.2 Polynomial Regression Model Fitting

   The training and validation losses of different polynomial degrees is shown in Figure 2. The best hyperparamter degree d is 7.

   The best Θ learned is [−29.60722754, −13.93265366, 366.49596123, −1100.45557026, 23.79149025, 432.79168423, −20.91996714, −25.45551777], which sequentially act as the coefficients of order 0 to 7. The testing MSE loss is 8770.13. The curve is shown in Figure 3. As required, the best Θ is gotten in two steps. First, the training data is futher split into training and validation to find the best polynomial order. Second, merge validation back to training and use the whole with normal equation to get this Θ. Comparing with the numbers in data generation, our leaned Θ is actually reasonable. Although the absolute value scale is quite different, it is because the data has been normalized in data generation. Considering the sign and relative value scale, the learned Θ matches the data generation. For example, the values for order 6 and 7 are negligible, which do not exist in the original data generation.

   Keeping the degree as 7, the learned curve and corresponding epoch losses are shown in Firgure 4. The same plots for degree 2 is shown in Firgure 5. The model curve of degree 7 is much more complex

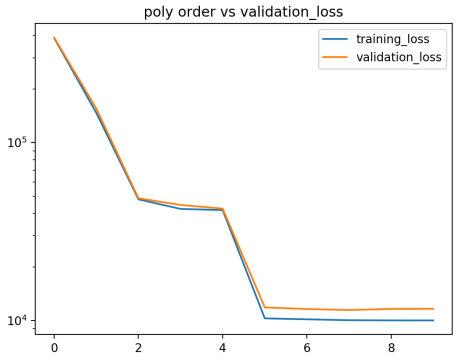
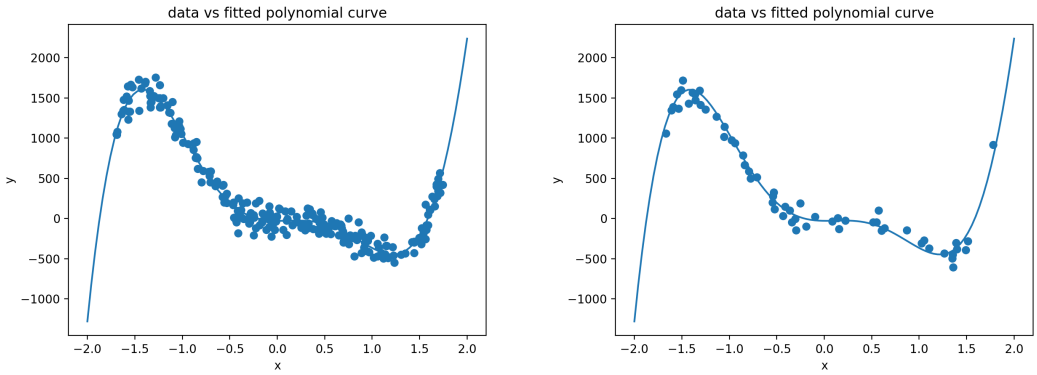


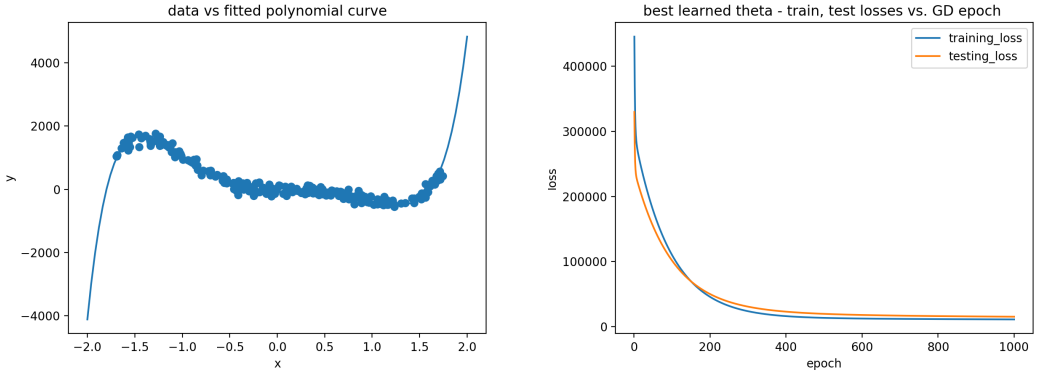
Figure 2: Loss of Polynomial Orders



(a) Training Data (b) Testing Data

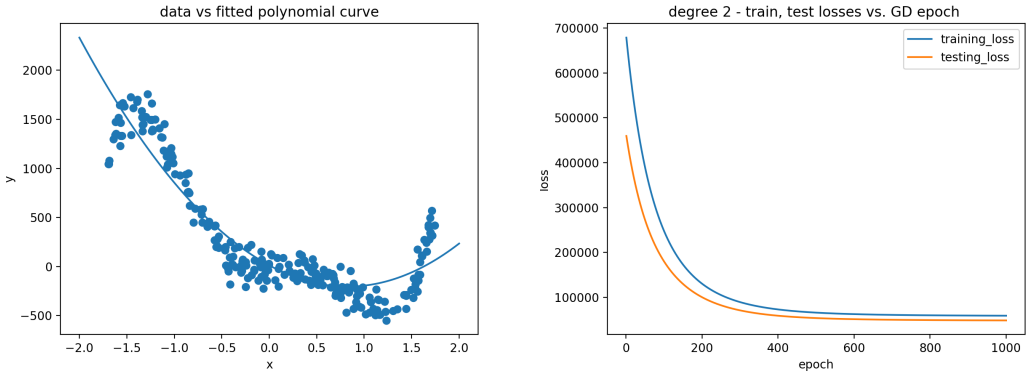
Figure 3: Learned Curve

and the end loss is obviously lower then degree 2. The training loss is lower than validation loss in degree 7, but in degree 2, the training loss is even higher. It means the model is too simple and cannot fit the data.



(a) Training Data (b) Training & Testing Loss

Figure 4: Gradient Descent with Degree 7



(a) Training Data (b) Training & Testing Loss

Figure 5: Gradient Descent with Degree 2

The plot of different example size is shown in Figure 6

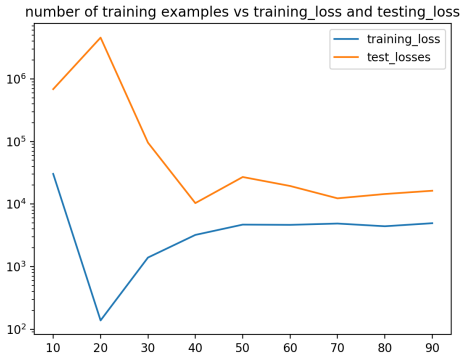


Figure 6: Loss per Example

2. Ridge Regression

2.1

2.1.1

The loss function of ridge regression equals to the sum of the linear regression loss and the coefficients' L2 norm

$$\begin{aligned} J(\beta) &= (y - X\beta)^T(y - X\beta) + \lambda\beta^T\beta \\ &= \beta^T X^T X\beta - \beta^T X^T y - y^T X\beta + y^T y + \lambda\beta^T\beta \\ &= \beta^T X^T X\beta - 2\beta^T X^T y + y^T y + \beta^T(\lambda I)\beta \end{aligned}$$

Get the derivative of the loss and find β make it to 0

$$\nabla_\beta J(\beta) = \frac{\partial J(\beta)}{\partial \beta} = 2X^T X\beta - 2X^T y + 2(\lambda I)\beta$$
$$\nabla_\beta J(\beta) = 0$$

$$
\begin{aligned}
2X^TX\beta - 2X^Ty + 2(\lambda I)\beta &= 0 \\
X^TX\beta + \lambda I\beta &= X^Ty \\
\beta &= (X^TX + \lambda I)^{-1}X^Ty
\end{aligned}
$$

2.1.2

No, it cannot be solve with linear regression. Use normal equation

$$
\begin{aligned}
\Theta &= (X^TX)^{-1}X^Ty \\
X^TX &= \begin{bmatrix} 35 & 70 \\ 70 & 140 \end{bmatrix} \\
|X^TX| &= 35 \times 140 - 70 \times 70 = 0
\end{aligned}
$$

Because the determinant of X^TX is 0, it cannot be inverted.

2.1.3

Lasso regression, because it tends to create sparse weights.

2.2

The training and validation loss with respect to lambda are plotted in the Figure 7. The training loss keep increasing while the lambda increasing. It makes sense because the regularization is used to generalize the model. Larger λ menas less overfit to the training data, so the loss increases. The validation loss is in a bowl shape. With descent amount of regularization, the model is generalized well to take care of the validation data. Overly large λ inhibits the model to learn anything, so the validation loss increases as well.

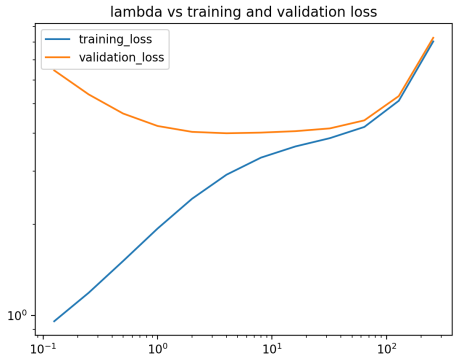


Figure 7: Loss per Lambda

The best λ leaned is 4. Its corresponding loss is 4.6368 and the norm of β is 6.6401. When the λ is 0, the loss is 11.0313 and the norm is 30.2697. When the λ is 512, the loss is 12.1264 and the norm is 4.6513. The larger the *lambda* the smaller the norm of the learned β. Overly large or small norm leads to large loss.

The learned β is shown in Figure 8. Comparing with the underlying distribution in the data generation, the values make sense. The second coefficient learned is indeed around 5. The other following coefficients for noises are relatively small.

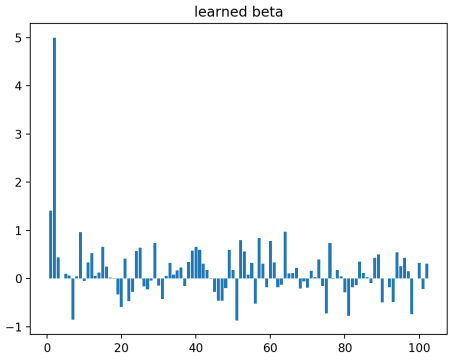


Figure 8: Beta

The gradient descent is also implemented for the ridge regression. The first 5 values of such learned beta is [1.28730024, 5.00181111, 0.38442863, 0.04989323, 0.10607296], which closely matches the values learned in normal equation [1.40916206, 5.00157608, 0.44139893, 0.00755006, 0.09740131]. Moreover, the loss and the norm of the gradient descent β are 4.6289 and 6.2333 which is almost the same with the ones of nomral equation reported above.

3. Sample Questions

3.1

It is not a good set of basis functions, because they do not overlap to cover all points. For example, when $x = 4$, no matter what β is, y can only be 0.

3.2

The three points are $[1, 1]$, $[2, 4]$, $[3, 9]$.

Leave $[1, 1]$ out, we have

$$4 = 2\beta_1 + 4\beta_2,\ 9 = 3\beta_1 + 9\beta_2$$
$$\beta_1 = 0,\ \beta_2 = 1$$

The validation loss is 0

Leave $[2, 4]$ out, we have

$$1 = \beta_1 + \beta_2,\ 9 = 3\beta_1 + 9\beta_2$$
$$\beta_1 = 0,\ \beta_2 = 1$$

The validation loss is 0

Leave $[3, 9]$ out, we have

$$1 = \beta_1 + \beta_2,\ 4 = 2\beta_1 + 4\beta_2$$
$$\beta_1 = 0,\ \beta_2 = 1$$

The validation loss is 0

So overall, the mean squared LOOCV error is 0