Pattern Recognition Assignment 1

2031566 - Han Yang
hanyang_sh@tongji.edu.cn

October 16, 2020

1 Dataset

I choose the dataset Horse Colic [1] from UCI Machine Learning Repository. This dataset contain 300 training instances and 68 test instances. It has 28 attributes, such as rectal temperature and pulse and respiratory rate, with 30% of the values are missing. As for this assignment, I use rectal temperature and pulse. Rectal temperature is linear and in degrees celsius, may be reduced when the animal is in late shock, normal temp is 37.8. Pulse is the heart rate in beats per minute, 30 -40 is normal for adults, animals with painful lesions or suffering from circulatory shock may have an elevated heart rate. With those attributes, We can train a clssifier to predict whether the horse will survive.

2 Preprocess

As there exist 30% missing value, there are many ways to deal with it, for this assignment i just drop the incorrect instances. In the origin data, the outcome attribute hava three types(1 2 3)mean lived died and euthanized. For the binary logistic regression, we can only classify two classes, so i classify euthanasia and death into one category denoted by 1, and live denoted by 0.

3 Modules

3.1 Preprocess

As description above, we need to preprocess the original data, I use numpy and pandas to process the data and save processed data to csv format.

3.2 Logistic Regression Algorithm

The main part of logistic regression is the logistic function (sigmoid function) and probability function and cost function, I augment X to $X = (X; 1)$ and w dimension to feature number plus 1:

$$Sigmoid\ Function : g(z) = \frac{1}{1 + e^{-}z} \qquad 1.1$$

$$Hypothesis\ Function : h(X) = g(X\omega) \qquad 1.2$$

$$Cost\ Function : J(\omega) = -\frac{1}{m}\sum (Y \log h(X) + (1 - Y)\log(1 - h(X))) \qquad 1.3$$

Instead of using Newton's Methos showed in the slide, I use Gradient Descent to optimize this minmiun problem. Use the following formulas to get derivative of ω:

$$\frac{\partial J(\omega)}{\partial \omega} = \frac{1}{n} X^T (h(X) - y) \qquad 1.4$$

[1] http://archive.ics.uci.edu/ml/datasets/Horse+Colic

3.3 Plot

After 4000 iterations, we have get the augmented $\omega = [-0.07633064, 0.03203682, 0.00125518]^T$, and i use matplotlib to plot the decision boundary.

4 Result

This dataset is not well,i use the hyperparameter at $learning\_rate = 1e-5, num\_iterations = 4000$ and I get a result at $\omega = [-0.07633064, 0.03203682, 0.00125518]^T, train\_accurancy = 0.6837606837606838, test\_accuran$ 0.728813559322034. And I plot the decision boundary in Figure 1 and cost curve in Figure 2.

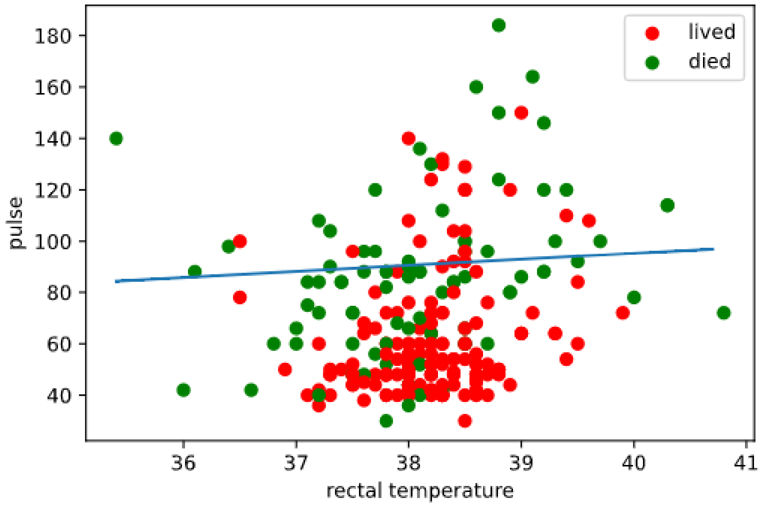

Figure 1: Decision boundary

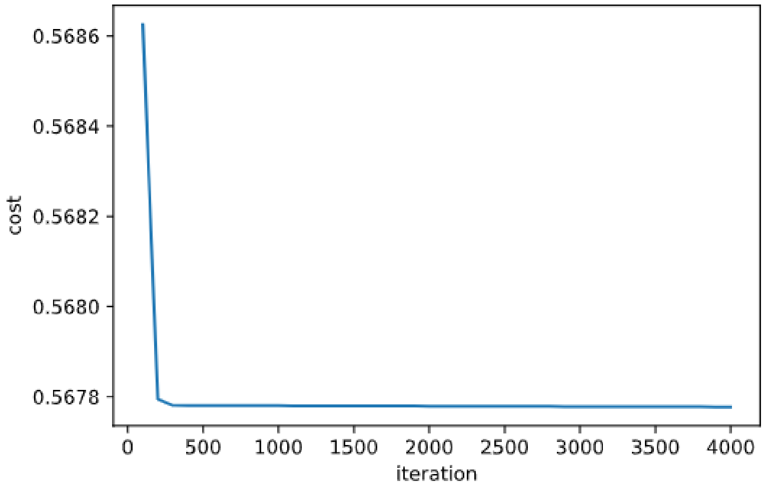

Figure 2: Cost curve

The horse dataset is linearly inseparable, to get a better result, I use the Iris dataset[2] to do this assignment again. I use the hyperparameter at $learning\_rate = 1e-4, num\_iterations = 4000$ and I get a result: $W = [0.95465512, -1.64464035, -0.17745044]^T, train\_accurancy = 0.9875, test\_accurancy =$ 1.0.And I plot the decision boundary in Figure 3 and cost curve in Figure 4.

[2] http://archive.ics.uci.edu/ml/datasets/Iris

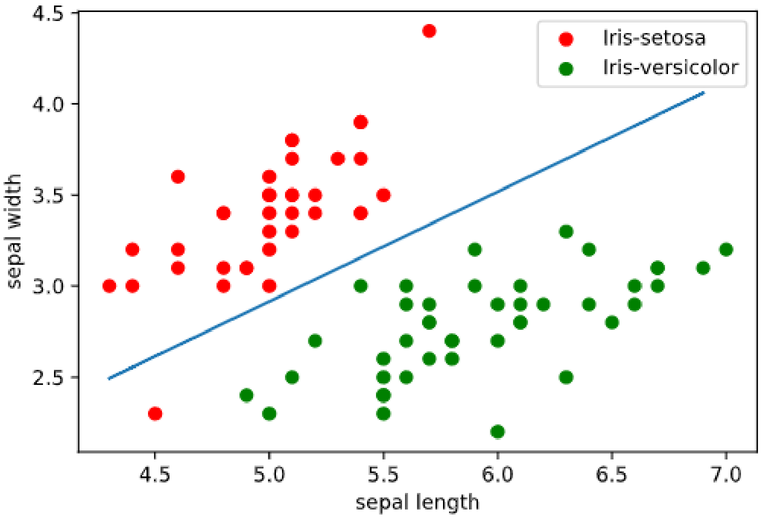

Figure 3: Decision boundary

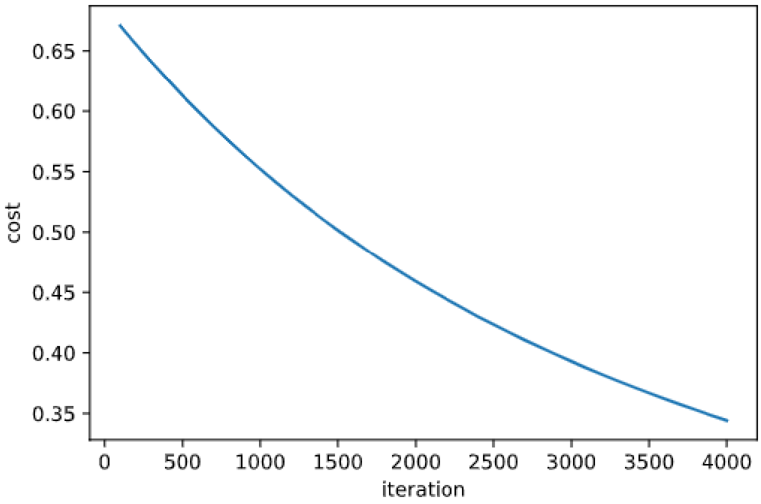

Figure 4: Cost curve