Predict Matching in a Speed Dating Experiment

Khuyen Tran
May 2, 2020

1 Summary

In this project, I use logistic regression to classify the MNIST database of handwritten digits. The source code for this project could be found here.

2 Introduction

The MNIST database is a large database of handwritten digits that is common used for training various image processing systems. My goal is to classify which digit the image represents.

Logistic regression is a statistical model that is used for binary classification. I will use this model to predict whether an image represents a certain digit.

3 Development

I have a training set of observations $S = (x_i, y_i)$ with $x_i \in \mathbb{R}^n$ and $y_i \in 0, 1$. I want to estimate the parameters β that maximizes the function,

$$h(\beta) = -\sum_{i=1}^{n} y_i log\pi_i + (1 - y_i)log(1 - \pi_i)$$
$$\pi_i(\beta) = \frac{1}{1 + e^(- x_i^T \beta)}$$

using the set S. Gradient of the loss function above is

$$\begin{aligned}
\nabla h(\pi) &= -\sum_{i=1}^{n} (\frac{y_i}{\pi_i} - \frac{1 - y_i}{1 - \pi_i}) \nabla \pi_i \\
&= -\sum_{i=1}^{n} \frac{y_i - \pi_i}{\pi_i(1 - \pi_i)} \nabla \pi_i \\
\nabla \pi_i(\beta) &= (1 + \exp(-x_i^T \beta))^{-2} \exp(-x_i^T \beta) \\
&= \pi_i(\pi_i - 1)x_i \\
\nabla h(\beta) &= -\sum_{i=1}^{n} \frac{y_i - \pi_i}{\pi_i(1 - \pi_i)} \pi_i(\pi_i - 1)x_i \\
&= \sum_{i=1}^{n} (y_i - \pi_i)x_i
\end{aligned}$$

I use the test set of observations $\mathcal{T} = (x_i, y_i)$ such that $x_i \in \mathbb{R}^n$ and $y_i \in 0, 1$ to compute the classification error. The error is computed by the equation

$$error = \frac{1}{|\mathcal{T}|} \sum_{(x_i, y_i) \in \mathcal{T}} |1_{\pi_i(\beta) > 0.5}(x_i) - y_i|$$

where $|\mathcal{T}|$ represents the number of elements of the set $\mathcal{T}$.

4 Experiments

I use gradient descent algorithm to update the value of β based on the gradient computed above.The stop conditions are:

- The number of iteration exceeds max iteration
- The distance between the current and new cost functions are less than or equal to the tolerance
- The distance between the current and new β are less than or equal to the tolerance

While the stop conditions are not met, β and the cost function will be kept updated with gradient. Python code for gradient descent:

```
def grad_descent(B_0, X, y, tol_B, tol_f, max_iter, alpha):

    k = 0

    n = X.shape[0]
    ones = np.ones((n, 1))
    X = np.concatenate((X,ones), axis=1)
    dis_f = float('inf')
    dis_B = float('inf')
    B = B_0

    pi_ = pi(B, X)


    while k < max_iter and dis_f >= tol_f and dis_B >= tol_B:

        d = grad(B, pi_, X, y)

        B_new = B - alpha*d

        pi_new = pi(B_new, X)

        f_x = func(B, pi_, X, y)
        f_new = func(B_new, pi_new, X, y)

        B_old = B

        B = B_new

        pi_ = pi(B, X)

        k += 1

        dis_B = np.linalg.norm(B - B_old)/max(1, np.linalg.norm(B_old))

        dis_f = np.abs(f_new - f_x)/max(1,np.abs(f_x))

    return B
```

To make this dataset a data for binary classification, the label is equal to 1 if it is a 1 digit and equal to 0 otherwise.

5 Conclusions

The target of this project is to perform binary classification with logistic regression using MNIST handwritten digits dataset. The algorithm is able to predict the image with the accuracy of 0.8865. Overall, combination of binary cross entrophy loss and gradient descent does a good job in binary classification. Further change in parameters such as step side or stopping threshold could be used to improve the accuracy of the classification.

6 Bibliography

THE MNIST DATABASE. (n.d.). Retrieved from http://yann.lecun.com/exdb/mnist/