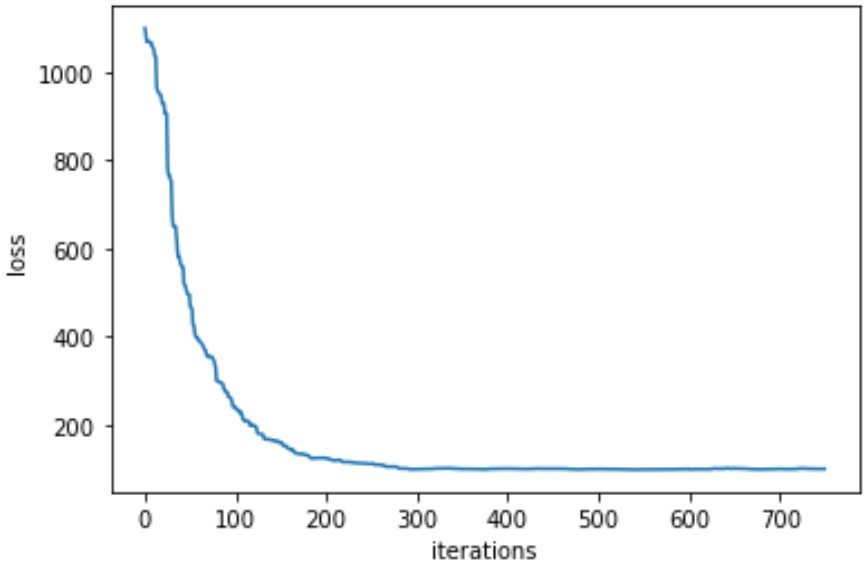
Machine Learning HW1

109550159 李驊恩

Coding Part:

Linear regression model

1. **Learning curve of training**



2. **Mean Square Error of my prediction and ground truth: 108.6755003**

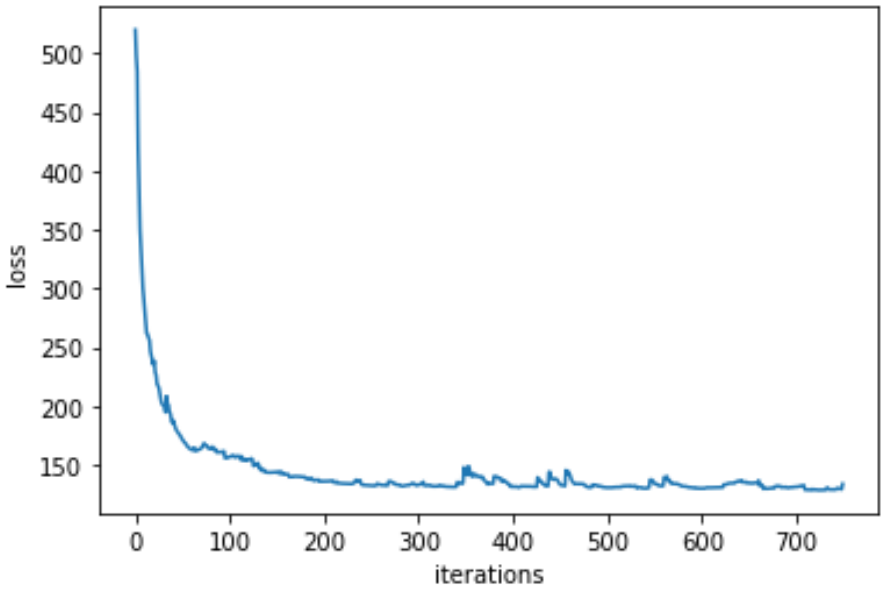
```
In [32]: runfile('C:/Users/user/Desktop/ML_HW1/ML_HW1-1.py', wdir='C:/
Users/user/Desktop/ML_HW1')
Mean Square Error of my prediction and ground truth: [108.6755003]
Weight: [51.56906919]
Intercept: [0.00090216]
```

3. **Weights and intercepts of my linear model: 51.56906919, 0.00090216**

```
In [32]: runfile('C:/Users/user/Desktop/ML_HW1/ML_HW1-1.py', wdir='C:/
Users/user/Desktop/ML_HW1')
Mean Square Error of my prediction and ground truth: [108.6755003]
Weight: [51.56906919]
Intercept: [0.00090216]
```

Logistic regression model

1. **learning curve of the training**



2. **Cross Entropy Error of my prediction and ground truth: 49.0675976**

```
In [22]: runfile('C:/Users/user/Desktop/ML_HW1/ML_HW1-2.py', wdir='C:/
Users/user/Desktop/ML_HW1')
Cross Entropy Error of my prediction and ground truth: [49.06759763]
Weight: [4.06759289]
Intercept: [1.68480571]
```

3. **weights and intercepts of my linear model: 4.06759289, 1.68480571**

```
In [22]: runfile('C:/Users/user/Desktop/ML_HW1/ML_HW1-2.py', wdir='C:/
Users/user/Desktop/ML_HW1')
Cross Entropy Error of my prediction and ground truth: [49.06759763]
Weight: [4.06759289]
Intercept: [1.68480571]
```

Question Part

1. **What's the difference between Gradient Descent, Mini-Batch Gradient Descent, and Stochastic Gradient Descent?**

What the difference between the three methods above is the number of data that are used to update the parameters in a single epoch.

In Gradient Descent, all the training data are taken at once to calculate the mean gradient to update the parameters.

In Mini-Batch Gradient Descent, we pick a mini-batch, which is a dataset that consists of few data from the actual dataset and calculate its mean gradient to update the parameters.

In Stochastic Gradient Descent, only one example of the whole dataset will be considered at a time to calculate its gradient used for updating parameters.

2. **Will different values of learning rate affect the convergence of optimization? Please explain in detail.**

Different values of learning rate do affect the convergence of optimization. Learning rate mainly affects the learning process of the model.

If the value of the learning rate is large, the model will learn rapidly. However, if the value is too large, the gradient descent may overshoot the minimum or keep oscillating. That is, the model fails to converge.

If the value of the learning rate is small, the model will learn slowly. It makes the learning curve become smoother but it slow down the process of converging.

3.

$$\sigma(a) = \frac{1}{1+e^{-a}} \Rightarrow \sigma(-a) = \frac{1}{1+e^{a}} = 1 - \frac{e^a}{e^a+1} = 1 - \frac{1}{1+\frac{1}{e^a}} = 1 - \frac{1}{1+e^{-a}}$$

$$= 1 - \sigma(a) \Rightarrow \sigma(-a) = 1 - \sigma(a) \quad \#$$

Let $\sigma(y) = \frac{1}{1+e^{-y}} = k \xrightarrow{\text{Inverse}} \frac{1}{1+e^{-k}} = y$

$$\frac{1}{1+e^{-k}} = y \Rightarrow 1 = y + ye^{-k} \Rightarrow \frac{1-y}{y} = e^{-k} \Rightarrow \ln\left(\frac{1-y}{y}\right) = -k \Rightarrow k = \ln\left(\frac{y}{1-y}\right)$$

$$\Rightarrow \sigma^{-1}(y) = \ln\left(\frac{y}{1-y}\right) \quad \#$$

4.

$$a_k = w_k^T \phi \Rightarrow a_{nj} = w_j^T \phi_n \Rightarrow \nabla_{w_j} a_{nj} = \phi_n$$

$$\frac{\partial y_k}{\partial a_j} = y_k(I_{kj} - y_j), \quad I_{kj} = \begin{cases} 1, & j = k \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial E}{\partial a_{nj}} = \frac{\partial E}{\partial y_{n1}} \cdot \frac{\partial y_{n1}}{\partial a_{nj}} + \frac{\partial E}{\partial y_{n2}} \cdot \frac{y_{n2}}{\partial a_{nj}} + \cdots + \frac{\partial E}{\partial y_{nK}} \cdot \frac{y_{nK}}{\partial a_{nj}} = \sum_{k=1}^{K} \frac{\partial E}{\partial y_{nk}} \frac{\partial y_{nk}}{\partial a_{nj}}$$

$$E(w_1, \cdots w_k) = -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \ln y_{nk} \Rightarrow \frac{\partial E}{\partial y_{nk}} = -\frac{t_{nk}}{y_{nk}}$$

$$\Rightarrow \sum_{k=1}^{K} \frac{\partial E}{\partial y_{nk}} \frac{\partial y_{nk}}{\partial a_{nj}} = \sum_{k=1}^{K} \left(-\frac{t_{nk}}{y_{nk}}\right) y_k(I_{kj} - y_j) = -\sum_{k=1}^{K} t_{nk}(I_{kj} - y_{nj}) = -t_{nj} + \sum_{k=1}^{K} t_{nk} y_{nj}$$

Since $\sum_{k=1}^{K} t_{nk} = 1$, $-t_{nj} + \sum_{k=1}^{K} t_{nk} y_{nj} = y_{nj} - t_{nj}$

$$\Rightarrow \text{By chain rule, } \nabla_{w_j} E(w_1, \cdots w_k) = \sum_{n=1}^{N} \frac{\partial E}{\partial a_{nj}} \nabla_{w_j} a_{nj} = \sum_{n=1}^{N} (y_{nj} - t_{nj}) \phi_n \quad \#$$