Homework1

付宇辉 无 05 2020010768

1. 证明：

首先记$\hat{x}=(x^T,1)^T,\hat{w}=(w^T,b)^T$, 如此$\hat{w}^T\hat{x}=w^Tx+b$

优化目的是要得到$f^\star(x)=sign(w^{\star T}x+b^\star)=sign(\hat{w}^{\star T}\hat{x}^\star)$, 即找到相应的$\hat{w}^\star$，使得准确分类。

由于数据线性可分，则遇到误分类点的次数有上界，即存在w_f使得所有数据点被准确分类。

使用随机梯度下降，会获得一系列$\widehat{w_k}$, 当k变大，$\widehat{w_k}$会越来越接近$\widehat{w_f}$

用两个向量的夹角去衡量两个向量的相似程度（向量的长度并不重要）。

$$\cos\theta=\frac{\widehat{w_k}\cdot\widehat{w_f}}{\|\widehat{w_k}\|\|\widehat{w_f}\|}$$

下面证明，有限次迭代内，θ会收敛

已知，存在γ使得$\min\limits_{\mathrm{i}} y_i(\widehat{w_f}^T\hat{x}_i)\geq\gamma$, 即存在比划分超平面更近的距离。

$$\begin{aligned}\widehat{w_f}\cdot\widehat{w_k}&=\widehat{w_f}(\widehat{w_{k-1}}+\eta y_i\hat{x}_i)=\widehat{w_f}\widehat{w_{k-1}}+\widehat{w_f}\eta y_i\hat{x}_i\geq\widehat{w_f}\widehat{w_{k-1}}+\eta\gamma\geq\widehat{w_f}\widehat{w_{k-2}}+2\eta\gamma\geq\cdots\\&\geq\widehat{w_f}\widehat{w_0}+k\eta\gamma\end{aligned}$$

令$R=\max\limits_{\mathrm{i}}\|x_i\|^2$

$$\begin{aligned}\|\widehat{w_k}\|^2&=\|\widehat{w_{k-1}}\|^2+2\eta y_i\widehat{w_{k-1}}\hat{x}_i+\eta^2\|\hat{x}_i\|^2\leq\|\widehat{w_{k-1}}\|^2+\eta^2\|\hat{x}_i\|^2\leq\|\widehat{w_{k-1}}\|^2+\eta^2R^2\\&\leq\|\widehat{w_{k-1}}\|^2+2\eta^2R^2\leq\cdots\leq k\eta^2R^2\end{aligned}$$

所以

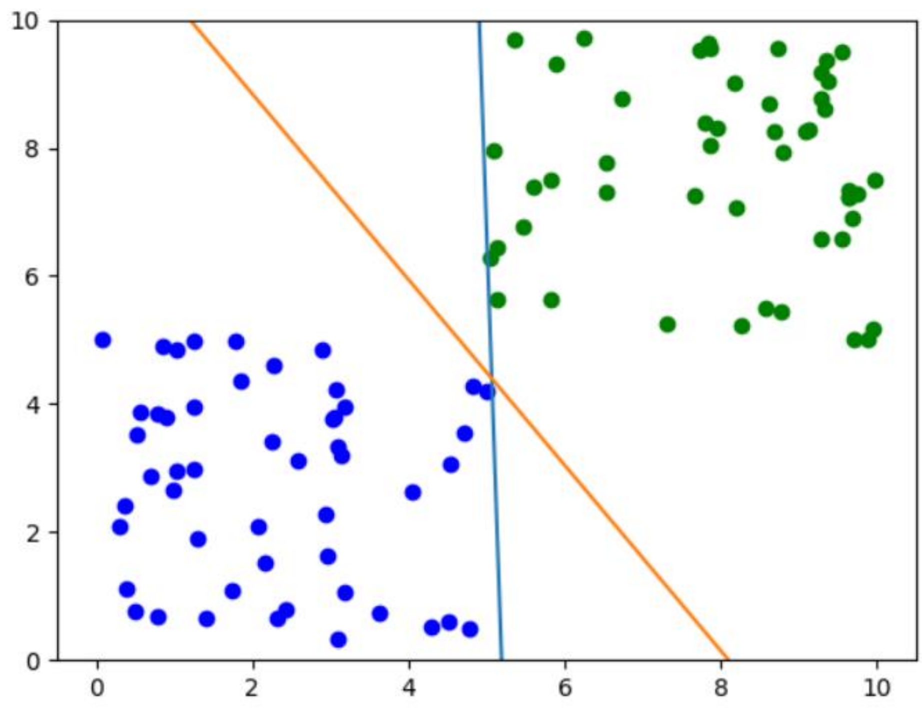
$$1\geq\cos\theta=\frac{\widehat{w_k}\cdot\widehat{w_f}}{\|\widehat{w_k}\|\|\widehat{w_f}\|}\geq\frac{\widehat{w_f}\widehat{w_0}+k\eta\gamma}{\|\widehat{w_f}\|\sqrt{k}\eta R}$$

得到

$$k\leq\frac{\left\|\widehat{w_f}\right\|^2R^2}{4\gamma^2}=\frac{R^2}{\left(4\min\limits_{\mathrm{i}}\frac{y_i\widehat{w_f}}{\|\widehat{w_f}\|}\hat{x}_i\right)^2}=\left(\frac{R}{\rho}\right)^2$$

即证明了有限次可得到收敛的w_k

2. 使用 numpy+matplotlib




```
0.29x_1+0.0083x_2+-1.5=0
The number of iterations is 29
0.26x_1+0.18x_2+-2.1=0
The number of iterations is 55
```

附源代码：

```
import numpy as np
import matplotlib.pyplot as plt

def update(X, y ,w, b, lr):
    cnt = 0
    flag = 1
    while flag:
        flag = 0
        for i in range(100):
            if y[i] * (w.dot(X[i]) + b) <= 0 :
                w += lr * y[i] * X[i]
                b += lr * y[i]
                flag = 1
                cnt += 1
    print(f"{w[0]:4.2}x_1+{w[1]:4.2}x_2+{b:4.2}=0")
```

```python
    print(f"The number of iterations is {cnt}")

    res_x = np.linspace(0, 10, 500)
    res_y = - (w[0] * res_x + b) / w[1]

    plt.plot(res_x,res_y)


if __name__ == '__main__':
    x_n = np.random.uniform(0, 5, [50, 2])
    x_p = np.random.uniform(5, 10, [50, 2])

    X = np.append(x_n, x_p).reshape((100,2))
    y = np.array([-1 if i < 50 else 1 for i in range(100)])

    plt.scatter(x_n[:,0], x_n[:,1], marker = 'o', color = 'blue')
    plt.scatter(x_p[:,0], x_p[:,1], marker = 'o', color = 'green')

    lr = 0.1
    w = np.zeros(2)
    b = 0
    update(X, y, w, b, lr)

    w = np.ones(2)
    b = 1
    update(X, y, w, b, lr)

    plt.ylim(0, 10)
    plt.show()
```