

lec05

Algorithm

我们可以将偏置 b 放进权重矩阵中，设置 $w_0 = b$ ，这样就能方便计算

```
// D是训练集, MaxIter是迭代次数
PerceptronTrain(D, MaxIter)
W = [1, 2, ..., d]
b = 0
for iter in [1, MaxIter] do
    for X, y in D do
        a = WX + b
        if y*a <= 0 then
            for i in [1, d] do
                W[i] = W[i] + y*X[i]
                b = b + y

return b, W

PerceptronTest(b, W, X)
a = WX + b
return sign(a)
```

简单来说，`PerceptronTrain` 函数就是

1. 初始化权重与偏置
2. 对于每轮迭代，计算激发分数 a
3. 如果 a 和数据标签 y 相乘后结果小于 0，我们就更新权重与偏置，更新量就是 y 。对于二元分类问题， y 的取值通常为 1 或 -1
4. `MaxIter` 是超参数
5. 对于每次迭代 `for iter in [1, MaxIter] do` 打乱训练集的顺序往往能产生好结果

对于 `PerceptronTest` 函数

$$\text{sign}(a) = \begin{cases} +1, & a > 0 \\ -1, & a < 0 \\ \text{根据情况}, & a = 0 \end{cases}$$

感知器的一些 feature

Online Algorithm: 一个接一个处理训练数据集中的对象（与需要访问整个数据集的批量学习相反，例如 k-NN）

Error Driven: 仅当使用当前参数（权重和偏置）对测试对象分类错误时才更新参数