



实验三——感知机学习算法

Perceptron Learning Algorithm

PPT制作：陈昱夫 郑铠奇



上节课的一些问题

- 1. 什么是过拟合
- 2. 什么是归一化/标准化
- 3. 实验原理写的不是怎么实现的流程，而是对算法的分析和理解
- 4. 伪代码的规范
- 5. 思考题从网上粘贴复制



用来解决二元分类问题（+1 和-1）。

样本 $x = \{x_1, x_2, \dots, x_d\}$ 权重向量 $w = \{w_1, w_2, \dots, w_d\}$ 阈值 θ

If $\sum_{i=1}^d w_i x_i \geq \theta$, 预测为+1

If $\sum_{i=1}^d w_i x_i < \theta$, 预测为-1

用一个符号函数表示 $y = \text{sign}(\sum_{i=1}^d w_i x_i - \theta)$

int sign(int x) {return x>0? +1 : -1}



为简便计算

$$\begin{aligned} & \text{sign}(\sum_{i=1}^d w_i x_i - \theta) \\ &= \text{sign}(\sum_{i=1}^d w_i x_i + (-\theta)(+1)) \\ &= \text{sign}(\sum_{i=0}^d w_i x_i) \\ &= \text{sign}(w^T x) \end{aligned}$$

样本 $x = \{+1, x_1, x_2, \dots, x_d\}$ 权重向量 $w = \{w_0, w_1, w_2, \dots, w_d\}$



步骤 1: 给每一个样本前加常数项 1。

样本 1: $x_1 = \{+1, x_{11}, x_{12}, \dots, x_{1d}\}$ label= y_1

样本 2: $x_2 = \{+1, x_{21}, x_{22}, \dots, x_{2d}\}$ label= y_2

样本 3: $x_3 = \{+1, x_{31}, x_{32}, \dots, x_{3d}\}$ label= y_3

步骤 2: 初始化权重向量 $w_0=0$ 或者其他值。



步骤 3: 遍历所有样本,

每当找到一个预测错误的样本, 即 $\text{sign}(w_t^T x_n) \neq y_n$

就更新 $w_{t+1} \leftarrow w_t + y_n x_n$

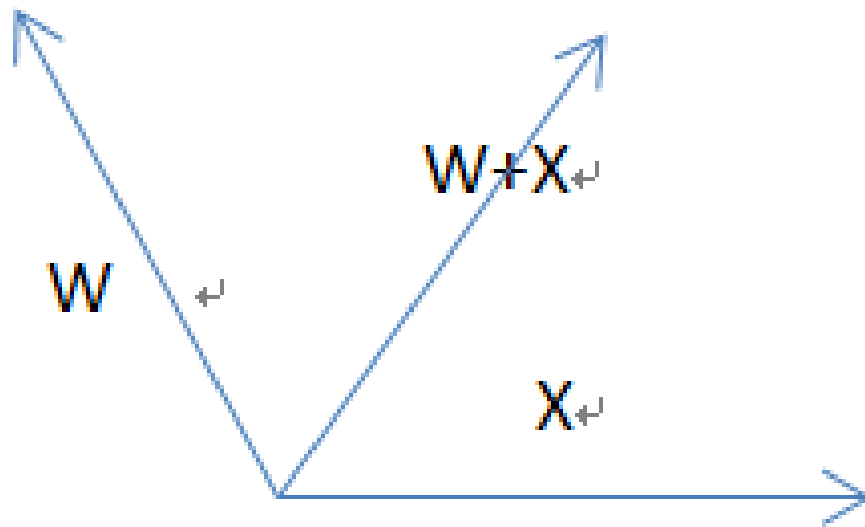
重复步骤 3 直至所有预测正确。

此时得到的 w 就是我们要求的值,

用此 w 来预测测试集的 label。

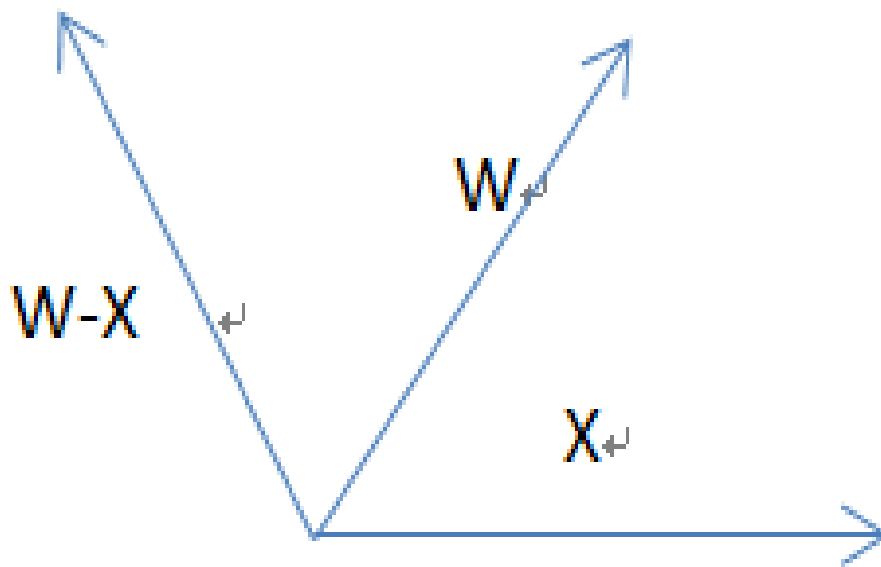


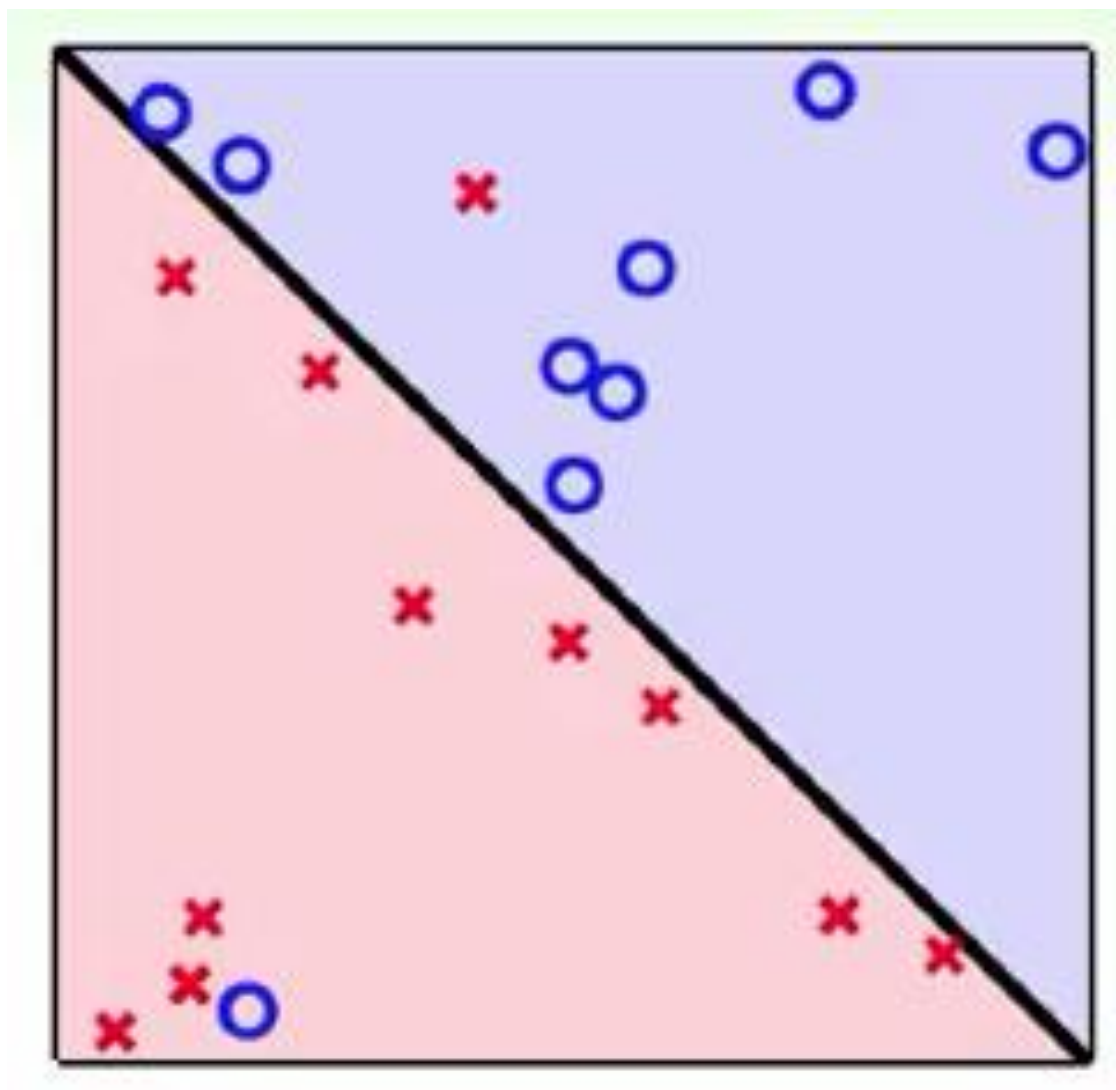
- $W_{t+1} \leftarrow W_t + [y_n \neq \text{sign}(W^T X)] * (y_n X_n)$
- 正样例被预测为负的情况下：





- $W_{t+1} \leftarrow W_t + [y_n \neq \text{sign}(W^T X)] * (y_n X_n)$
- 负样例被预测为正的情况下:







- **PLA**不适用非线性的问题，很多时候 w 无法满足全部点，这时候有两种方法：

- 1. 设置迭代次数，到一定程度就返回此时的 w , 不管它到底满不满足所有训练集。
- 2. 找一个 w ，使得在训练集里以此 w 来划分后，分类错误的样本最少。即相当于有一个口袋放着一个 w ，把算到的 w 跟口袋里的 w 比对，放入比较好的一个 w ，这种算法又被称为口袋（pocket）算法。

思考题：
有什么其他的手段可以解决数据集非线性可分的问题？



口袋算法

步骤 1: 给每一个样本前加常数项 1。

步骤 2: 初始化权重向量 $w_0=0$ 或者其他值,
以及一个全局最优向量 w 。

步骤 3: 遍历所有样本,

每当找到一个预测错误的样本, 即 $sign(w_t^T x_n) \neq y_n$

就更新 $w_{t+1} \leftarrow w_t + y_n x_n$

若 w_{t+1} 错误率小于 w , 则 w_{t+1} 赋给 w 。

重复步骤 3 直至达到指定迭代次数。

此时得到的 w 就是我们要求的值, 用此 w 来预测测试集的 label。



简单的例子

编号	特征1	特征2	标签
Train1	-4	-1	+1
Train2	0	3	-1
Test1	-2	3	?

步骤1：样本数据加常数项1

train1: $x_1 = \{1, -4, -1\}$

train2: $x_2 = \{1, 0, 3\}$

test1: $x_3 = \{1, -2, 3\}$



简单的例子

train1: $x_1 = \{1, -4, -1\}$ $y_1 = +1$

train2: $x_2 = \{1, 0, 3\}$ $y_2 = -1$

test1: $x_3 = \{1, -2, 3\}$ $y_3 = ?$

步骤2: 初始化向量 $\mathbf{w} = \{1, 1, 1\}$

步骤3: 计算 $\text{sign}(\mathbf{w}^T x_1) = -1 \neq y_1 \rightarrow$ train1 错误

更新 \mathbf{w} 得 $\mathbf{w} = \mathbf{w} + y_1 x_1 = \{2, -3, 0\}$

计算 $\text{sign}(\mathbf{w}^T x_2) = +1 \neq y_2 \rightarrow$ train2 错误

更新 \mathbf{w} 得 $\mathbf{w} = \mathbf{w} + y_2 x_2 = \{1, -3, -3\}$

计算得 $\text{sign}(\mathbf{w}^T x_1) = y_1$ 且 $\text{sign}(\mathbf{w}^T x_2) = y_2$

预测全正确, 停止迭代

预测: 计算 $\text{sign}(\mathbf{w}^T x_3) = -1$, 所以 y_3 预测为 -1



评测指标

- 本次实验共有四个指标：
 - Accuracy(准确率)
 - Precision(精确率)
 - Recall(召回率)
 - F1(F值)



评测指标

- 对于二元分类：
 - TP: 本来为+1, 预测为+1
 - FN: 本来为+1, 预测为-1
 - TN: 本来为-1, 预测为-1
 - FP: 本来为-1, 预测为+1

T: True F: False

N: negative P: positive



评测指标

- 四个指标：
 - TP: 本来为+1, 预测为+1
 - FN: 本来为+1, 预测为-1
 - TN: 本来为-1, 预测为-1
 - FP: 本来为-1, 预测为+1

思考题：请查询相关资料，解释为什么要用这四种评测指标，各自的意义是什么。

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{F1} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$



实验任务

1. 实现PLA原始算法和口袋算法
2. 采用4种指标评价并分析你的实验结果
3. 尝试优化，并对优化后的结果进行分析
4. 在报告中回答两个思考题
5. 验收使用的模型是：权重全部初始化为一的PLA原始算法。



提交要求

- 一份报告
- 两份代码：
 1. 原始PLA算法（若有优化，请交最优版本）
 2. 口袋PLA算法（若有优化，请交最优版本）
 3. 命名格式为PLA_initial_1535XXXX.xxx和PLA_pocket_1535XXXX.xxx
- 一份结果：
 1. 你认为最优的模型下，对test数据的预测结果。
 2. 命名格式为 学号_姓名拼音_PLA.csv



注意事项

1、作业提交地址

FTP地址: <ftp://39.108.233.34>

登录用户名与密码均为 student

2、命名方式

查询“实验课须知”，实验报告，所有代码文件以及结果文件都需要上交。

3、编程语言可用 C++, python, matlab, java等，**不能使用现成库（如 sklearn 等）**，否则扣分

4、提交截止时间

2017年10月25日23: 59: 59前