机器学习算法系列（9）：感知机

Introduction

感知机（perceptron）是二类分类的线性分类模型，输入为实例的特征向量，输出为实例的类别，取+1和-1二值。

感知机对应于输入空间（特征空间）中将实例划分为正负两类的分离超平面，导入基于误分类的损失函数，利用梯度下降对损失函数进行极小化，求得感知机模型，属于判别模型

感知机学习算法简单易于实现，分为原始形式和对偶形式。1957年由Rosenblatt提出，是神经网络和支持向量机的基础

本章框架如下：

- 感知机模型
- 感知机的学习策略（损失函数）
- 感知机学习算法（原始形式与对偶形式），并证明算法的收敛性

一、 感知机模型

1.1 感知机模型

感知机是一种线性分类器，属于判别模型。

假设我们的输入空间（特征空间）是$\chi \subseteq R^{n}$，输出空间是$\boldsymbol{y}=\{+1,-1\}$。输入$\boldsymbol{x} \in \boldsymbol{\chi}$表示实例的特征向量，对应于输入空间（特征空间）的点；输出$y \in \boldsymbol{y}$表示实例的类别。由输入空间到输出空间的函数

$$f(x)=\operatorname{sign}(\boldsymbol{w} \cdot x+\boldsymbol{b})$$

其中，$\boldsymbol{w} \in \boldsymbol{R}^{n}$为权值或权值向量，$\boldsymbol{b} \in \boldsymbol{R}^{n}$叫做偏置，sign是符号函数，即

$$\operatorname{sign}(\mathrm{x})=\begin{cases}+1, & \mathrm{x} \geqslant 0 \\ -1, & \mathrm{x}<0\end{cases}$$

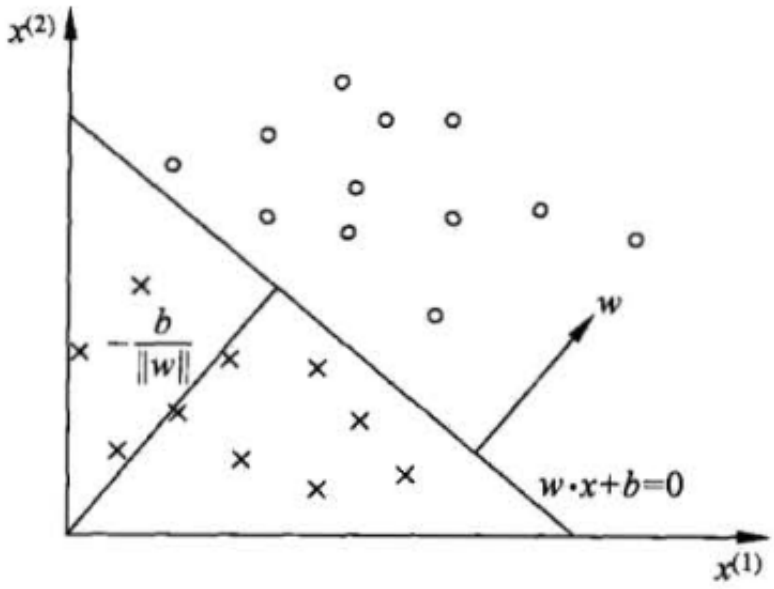
它的假设空间为：函数集合

$$\{f|f(x) = w \cdot x + b\}$$

感知机学习就是由训练数据集（实例的特征向量及类别）求得感知机模型，即求得模型参数w, b。
感知机预测即为通过学习得到的感知机模型，对新的输入实例给出其对应的输出类别。

1.2 感知机的几何解释

线性方程$w \cdot x + b = 0$对应于特征空间R^n中的一个超平面S，其中w是超平面的法向量，b是超平面的截距，超平面将特征空间划分为两个部分。两部分的特征向量被分为正、负两类，超平面S也称为分离超平面。



二、感知机学习策略

2.1 数据集的线性可分性

给定一个数据集

$$T = \left\{\left(x_1, y_1\right), \left(x_2, y_2\right), \cdots, \left(x_N, y_N\right)\right\}$$

若存在某个超平面S

$$w \cdot x + b = 0$$

能够将数据集的正实例点和负实例点完全正确的划分到超平面的两侧，即对所有的$y_i = +1$的实例，有$w \cdot x + b > 0$；对所有的$y_i = -1$的实例，有$w \cdot x + b < 0$。则数据集T为线性可分数据集；

否则称数据集线性不可分。

2.2 感知机学习策略

我们选择将误分类点到超平面S的总距离作为感知机模型的损失函数

由几何解释可以清楚地看到，任一点到超平面S的距离为:

$$\frac{1}{||w||}|w \cdot x_0 + b|$$

而我们对于误分类点的定义为:

$$-y_i\left(w \cdot x_0 + b\right) > 0$$

误分类点到超平面的距离:

$$-\frac{1}{||w||}y_i(w \cdot x + b)$$

则误分类点到超平面的总距离:

$$-\frac{1}{||w||}\sum_{x_i \in M} y_i\left(w \cdot x_i + b\right)$$

据上述我们定义损失函数为:

$$L(w, b) = -\sum_{x_i \in M} y_i\left(w \cdot x_i + b\right)$$

其中M为误分类点的集合，此即为感知机学习的经验风险函数。一个特定样本点的损失函数，在误分类时是参数w, b的线性函数，在正确分类时是0.因此，给定训练数据集T，损失函数$L(w, b)$是w, b的连续可导函数。感知机学习的策略就是在假设空间中选取使损失函数最小的模型参数，即感知机模型。

三、感知机学习算法

这样我们就把感知机的学习问题转化为求解损失函数的最优化问题，最优化的方法是随机梯度下降法。

3.1 感知机学习算法

首先我们确定要求解的最优化问题是：

$$\min_{w,b} L(w,b) = -\sum_{x_i \in M} y_i\left(w \cdot x_i + b\right)$$

通过随机梯度下降法来求解最优化问题。首先，任意选择一个超平面w_0, b_0，然后用梯度下降法不断地极小化目标函数，一次随机选取一个误分类点使其梯度下降，而不是一次使M中所有误分类点的梯度下降。

计算得到梯度为：

$$\nabla_w L(w,b) = -\sum_{x_i \in M} y_i x_i$$

$$\nabla_b L(w,b) = -\sum_{x_i \in M} y_i$$

对权值进行更新：

$$w \leftarrow w + \eta y_i x_i$$

$$b \leftarrow b + \eta y_i$$

其中η称为学习率，通过迭代可以期待损失函数不断减小，直到为0.

对于上述算法过程，我们可以有一个直观的解释：当一个实例点被误分类，则调整w, b的值，使分离超平面向该误分类点的一侧移动，以较少该误分类点与超平面的距离，直至超平面越过该误分类点使其被正确分类。当然感知机学习算法由于采用不同的初值或选取不同的误分类点，解可以不同。

3.2 对偶形式

对偶形式的基本想法是，将w和b表示为实例x_i和标记y_i的线性组合的形式，通过求解其系数而求得w和b，我们假设初始值w_0和b_0均为0。对误分类点(x_i, y_i)通过

$$w \leftarrow w + \eta y_i x_i$$

$$b \leftarrow b + \eta y_i$$

逐步修改w, b,设修改n次，则最后学习到的w, b可以分别表示为

$$w = \sum_{i=1}^{N} n_i \eta y_i x_i = \sum_{i=1}^{N} a_i y_i x_i$$

$$b = \sum_{i=1}^{N} a_i y_i$$

当$\eta = 1$时，表示第i个实例点由于误分而进行更新的次数。实例点更新次数越多，意味着它距离分离超平面越近，也就越难正确分类、换句话说，这样的实例对学习结果影响最大。

因为对偶形式的训练实例仅以内积的形式出现。为了方便，可预先将训练实例间的内积计算出来并以矩阵的形式存储，这个矩阵就是所谓的Gram矩阵。

$$G = \left[x_i \cdot x_j\right]_{N \times N}$$

与原始形式一样，感知机学习算法的对偶形式迭代是收敛的，存在多个解。

总结感知机学习算法的对偶形式如下：

输入：线性可分的数据集训练数据集$T = \left\{\left(x_1, y_1\right), \left(x_2, y_2\right), \cdots, \left(x_N, y_N\right)\right\}$，其中$x_i \in \chi = \boldsymbol{R^n}$ $y_i \in \boldsymbol{y} = \{-1, +1\}, i = 1, 2, \cdots, N$，学习率$\eta(0 < \eta \leq 1)$；

输出：a, b；感知机模型

$$f(x) = sign\left(\sum_{j=1}^{N} a_j y_j x_j \cdot x + b\right)$$

其中$a = \left(a_1, a_2, \cdots, a_N\right)^T$

- 1）$a \leftarrow 0, b \leftarrow 0$
- 2）在训练集中选取数据$\left(x_i, y_i\right)$
- 3）如果$y_i\left(\sum_{j=1}^{N} a_j y_j x_j \cdot x_i + b\right) \leq 0$

$$a_i \leftarrow a_i + \eta$$

$$b \leftarrow b + \eta y_i$$

- 4）转至(2)直到没有误分类数据

四、参考资料

李航《统计学习方法》