

# 机器学习系列之 感知器模型

## 前言

---

感知器模型是线性分类模型的一个基本模型，是当今主流的神经元基本结构，掌握好感知器模型有利于我们的机器学习进一步学习，这里简单介绍下感知器模型并且用Python代码演示。

如有谬误，请联系指正。转载请注明出处。

联系方式：

**e-mail:** FesianXu@163.com

**QQ:** 973926198

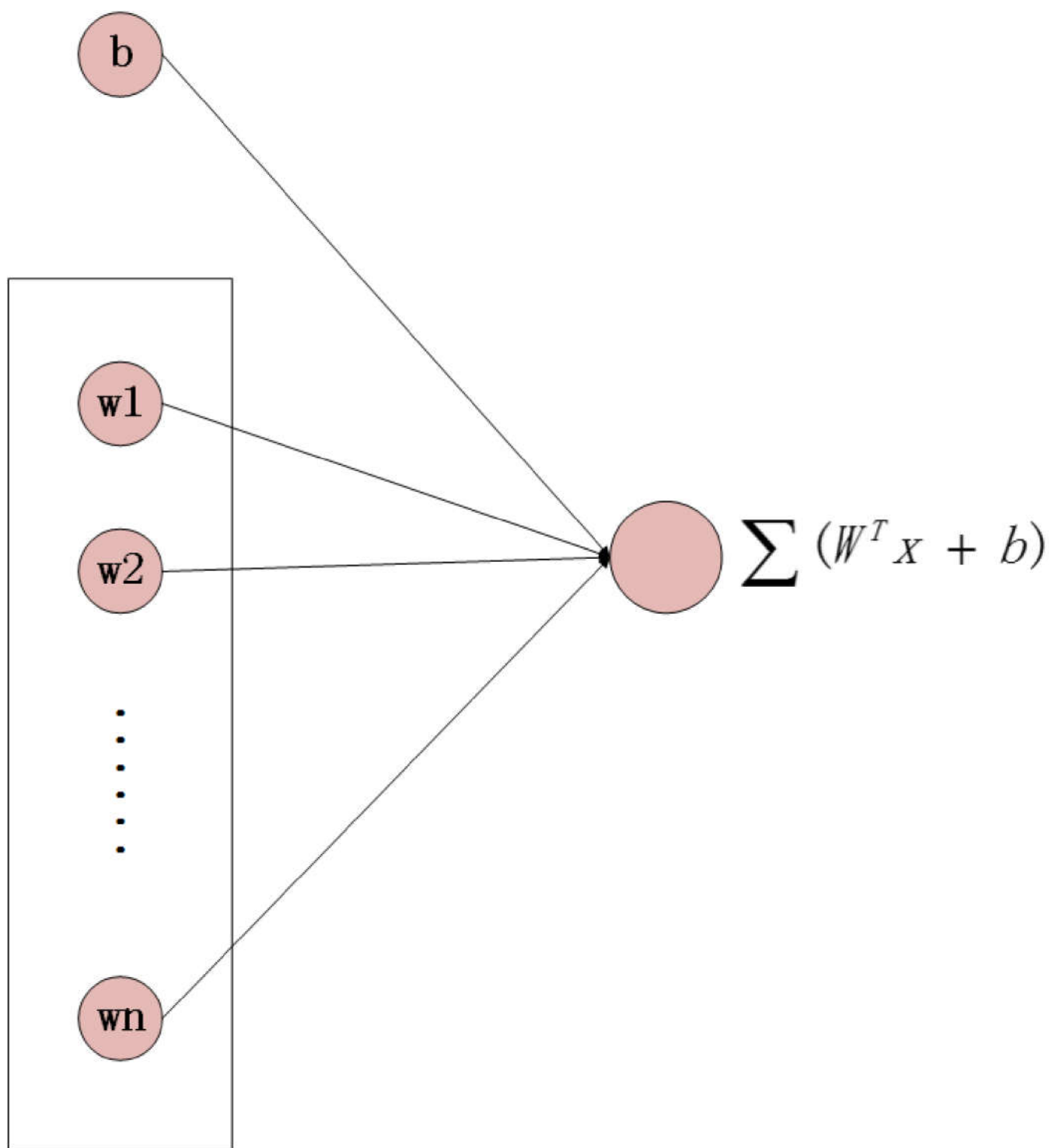
**github:** <https://github.com/FesianXu>

代码开源：[click](#)

---

## 线性分类模型与线性回归模型的区别

在[线性回归模型](#)中我们谈到了**线性回归模型**，而机器学习中很多任务是涉及到分类任务的，单纯的回归模型不能离散输出而只能连续输出，比如只能连续输出区间 $[0, 1]$ 的值，而不能离散输出0或者1，因此需要对线性回归模型进行一定的改造才能变为线性分类模型。下图为线性回归模型的图示，这个也称为一个神经元：



Linear Regression model

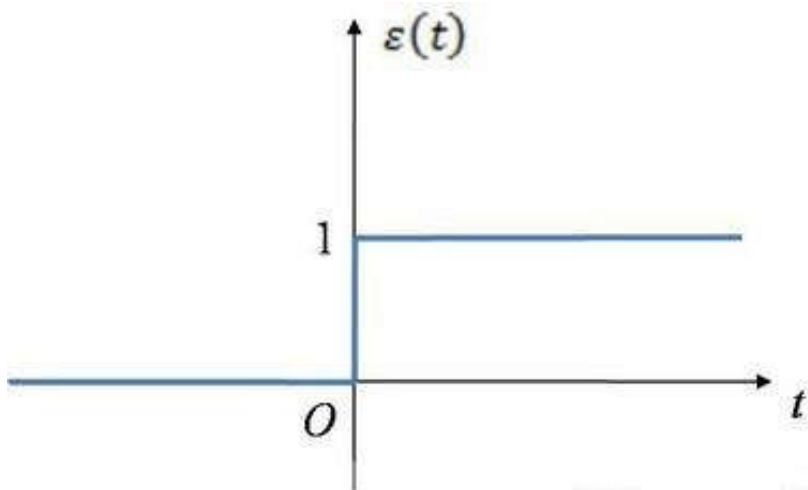
进行所谓的‘改造’指的是在神经元的输出端添加一个**激活函数**，使得输出从连续取值变为离散输出如0或者1。

## 感知器模型

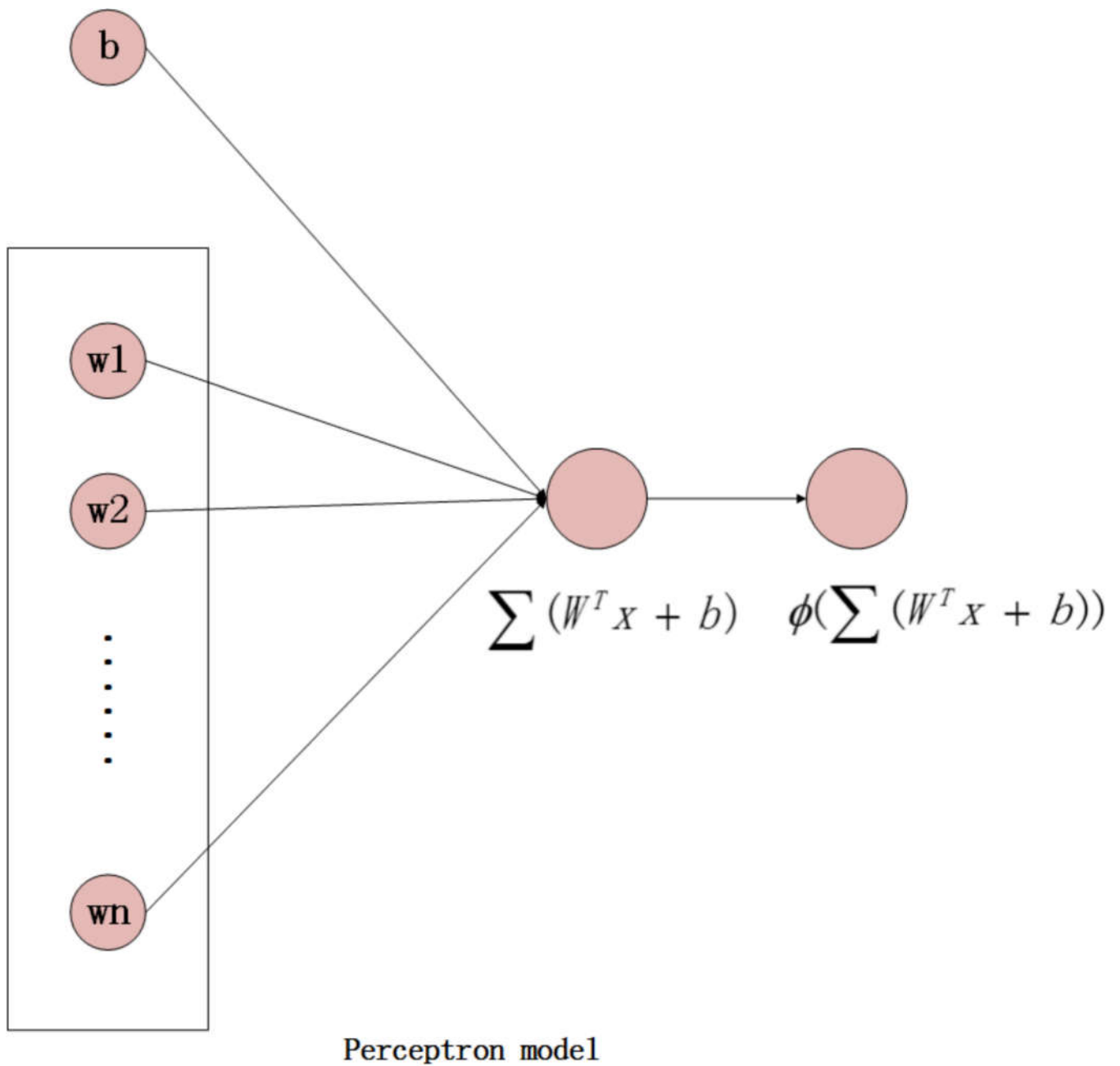
在经典的感知器模型(Perceptron)中，在输出端添加了一个**阶跃函数 ( Step Function )**，这样就将输出离散到了0或者1，表达式如：

$$\phi(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$

图像如：



经过改造后的模型示意图如：



这个模型类似于线性回归模型，参数共有 $W^T$ 和 $b$ ，因此在训练过程中需要学习到这两个参数。

## 训练策略

---

感知器采用的训练策略和线性回归，BP反向传播算法等是不同的，感知器采用了激活函数，而且这个激活函数 $\phi(x)$ 是不可导的，因此误差函数的梯度将会没有办法传播到输入层的权值中，因此**不能采用梯度反向传播的策略**去学习参数，我们在感知器中，采用的是**误差驱动更新**的策略，也就是**将误分类的样本用来更新参数，将正确分类的样本忽略**。

## 代码演示，基于Python和numpy

### 数据描述

---

### 代码

---