跟我学人工智能

刘森

2018 年 1 月 16 日

你好，世界hello, world

1 机器学习算法

1 不同的算法，本身没有好坏之分，有的只是，根据不同的场景选择合适的算法。

2 线性回归和Logistic回归，虽然听起来都叫作“回归”，但其实两者却是做不一样的事情：一个是做连续数据的预测，一个是做离散数据的预测；一个是真正做回归的，一个是做分类的，它们两个【用途】是完全不一样的。【如何推导出来？】线性回归是用高斯分布的方式推导出来，Logistic回归既然是做分类，就用Bnody分布，两点分布来推导出来。两者大的工具都是【最大似然估计】。在线性回归里面，要讨论一个东西：【最小二乘法的本质是什么】。或者说，为什么有最小二乘法呢？有没有最小三乘法呢？有没有最小四乘法呢？在【线性回归】和【Logistic回归】中强调两个工具：【梯度下降算法】和【极大似然估计】。

1.1 线性回归

高斯分布
极大似然估计MLE
最小二乘法的本质

1.1.1 什么是线性回归

线性回归$y = ax + b$

考虑多个变量情形，例如两个变量，$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$ ，可以写成如下形式：

$$h_\theta(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x$$

其中，θ展开后，呈现如下形式：

$$\begin{pmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{pmatrix}$$

其中，x展开后，呈现如下形式：

$$\begin{pmatrix} 1 \\ x_1 \\ x_2 \end{pmatrix}$$

上式中的1就表示x_0，而相应的θ_0表示截距，是比较难以直接解释的。

再把上面的式子拿过来，

$$h_\theta(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x$$

在$h_\theta(x)$中，x看起来是【自变量】，但事实上是【样本】，所以x是已知的，而θ是未知的，我们要通过某一种办法来求解出θ，这个就是【线性回归要解决的问题】。

第05课《回归》00:10:30

目前讲的问题是【what】，即什么是线性回归。过一会儿，会讲【how】，用什么样的工具去求，如何去求的问题。

1.1.2 使用极大似然估计解释最小二乘

第05课《回归》00:20:00

使用极大似然估计解释最小二乘

$$y^{(i)} = \theta^T xY(i) + \epsilon^{(i)}$$

the $\epsilon^{(i)}$ are distributed IID (independently and identically distributed) according to a Gaussian distribution (also called a Normal distribution) with mean $zero$ and some variance σ^2.

误差$\epsilon^{(i)}(1 \le i \le m)$是独立同分布的，服从均值为0，方差为某定值$\sigma^2$的【高斯分布】。原因：【中心极限定理】，可以查阅一下“中心极限定理的意义”。

似然函数第05课《回归》00:21:21

首先，两边是相等的：

$$y^{(i)} = \theta^T x^{(i)} + \epsilon^{(i)}$$

其中，$x^{(i)}$表示第i个【样本】，$\theta^T x^{(i)}$表示第i个样本的【预测值】，$y^{(i)}$表示第i个样本的【真实值】，而$\epsilon^{(i)}$表示第i个样本的误差。

根据【中心极限定理】，$\epsilon^{(i)}$应该是呈现一个高斯分布的形态。

$$p(\epsilon^{(i)}) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(\epsilon^{(i)})^2}{2\sigma^2})$$

另外，$\epsilon^{(i)} = y^{(i)} - \theta^T x^{(i)}$，此时将$\epsilon^{(i)}$代入上式：

$$p(y^i|x^i;\theta) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2})$$

如此一来，上式当中就没有误差ϵ了，因此只要指定了x和θ，就可以认为是一个y的分布。换句话讲，y其实服从的是【均值是$\theta^T x$，方差是某一个σ的高斯分布（正态分布）】。

那么，用什么可以估计这个θ呢？答：【最大似然估计】。

在上面的公式中，i只是表示第i个样本，假设一共有m个样本，那么，【m个样本的似然估计】就可以表示为：

$$L(\theta) = \prod_{i=1}^{m} p(y^i|x^i;\theta) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2})$$

如此一来，怎么求θ呢？直接对【似然函数】取对数，然后再想办法。

高斯的对数似然与最小二乘

$$
\begin{aligned}
l(\theta) &= \log L(\theta) \\
&= \log \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}) \\
&= \sum_{i=1}^{m} \log \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}) \\
&= m \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{\sigma^2} \cdot \frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - \theta^T x^{(i)})^2
\end{aligned} \tag{1}
$$

现在，其实是通过【最大似然估计】加上【高斯分布】来得到了【最小二乘法】目标函数。换句话说，这就是解释的“为什么会有最小二乘法”这个概念。

$$
J(\theta) = \frac{1}{2} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 \tag{2}
$$

1.1.3 θ的解析式的求解过程

θ的解析式的求解过程第05课《回归》00:29:52

1.2 逻辑回归：分类问题的首选算法

1.3 工具

梯度下降算法
极大似然估计

1.4 Softmax

2 Tensorflow

2.1 安装Tensorflow

conda install tensorflow
pip install tensorflow

3 数学知识

1.LATEX控制序列的概念（类似于函数）
控制序列可以是作为命令：以“\”开头，参数：必须参数和可选参数。