Python
机器学习实战

逻辑回归和最大似然

逻辑回归

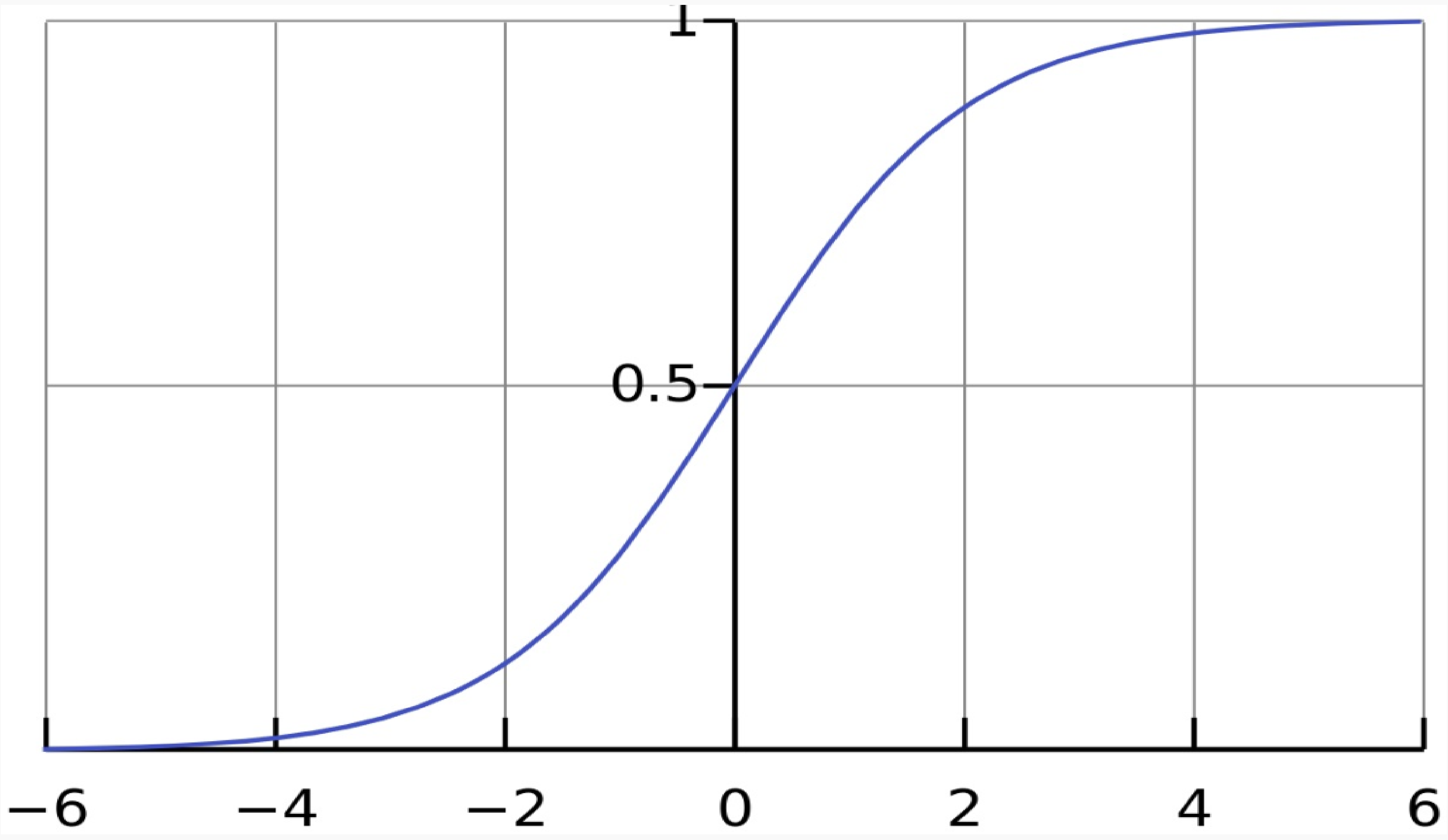
逻辑回归和线性回归

- 对数几率是关于x线性变化的

logit= $\theta^T x$

逻辑函数（Sigmod函数）

p= $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$



逻辑函数的导数

$$f(x)=\frac{1}{1+e^{-x}}$$

$$f'(x)= f(x)\cdot(1-f(x))$$

$$h_\theta(x) =\frac{1}{1+e^{-\theta^T x}}$$

$$\frac{\partial}{\partial\theta_j}h_\theta'(x) = f(x)\cdot(1-f(x))\,x_j$$

最大似然估计

逻辑回归

- 最大似然估计法（MLE）是在总体的分布类型已知的条件下所使用的一种参数估计方法.
- 它首先是由德国数学家高斯在1821年提出的 .然而，这个方法常归功于英国统计学家费歇 .
- 费歇在1922年重新发现了这一方法，并首先研究了这种方法的一些性质 .

逻辑回归

极大似然估计法是基于极大似然原理提出的。为了说明极大似然原理, 我们先看个例子。

某同学与一位猎人一起外出打猎。忽然，一只野兔从前方窜过，只听一声枪响，野兔应声倒下 .

若让你推测一下，

是谁击中的野兔,你会怎样想?

逻辑回归

只一枪便击中,一般情况下猎人击中的概率比同学击中的概率大。 故这一枪极大可能是猎人打的。

这一想法中就已经包含了最大似然原理的基本思想 .

最大似然原理

- 概率大的事件在一次观测中更容易发生
- 在一次观测中发生了的事件其概率应该最大

似然函数和极大似然

设总体X为离散型，其分布律为P{X=x}=p(x; θ)的形式已知，θ为待估参数，Θ是参数θ的可能取值范围。

设x_1 ,x_2 ,⋯, x_n为一组样本值，即事件{X_1= x_1,⋯, X_n = x_n}的联合概率密度函数为

$$L(\theta)=L(x_1,\cdots,x_n;\theta)=\prod_{i=1}^{n}p(x_i;\theta),\ \theta\in\Theta.$$

函数L(θ) 是关于θ的 函数，称为似然函数。

极大似然估计法就是在参数θ的可能取值范围内，选取函数L(θ)达到最大的参数值θ.

最大似然估计的步骤

1. 由总体分布写出样本的联合分布律或联合概率密度；即似然函数

$$L(\theta_1,\theta_2,\cdots,\theta_n)=\begin{cases}\prod_{i=1}^{n} p(x_i;\theta_1,\theta_2,\cdots,\theta_n)\\ \prod_{i=1}^{n} f(x_i;\theta_1,\theta_2,\cdots,\theta_n)\end{cases}$$

2. 取对数似然
3. 对对数似然函数求导数，得驻点（最大值点）
4. 用样本值代入最大值点的表达式，就得到参数的估计值

Example

有一个正反面不是很匀称的硬币，如果正面朝上记为H，方面朝上记为T，抛10次的结果如下： T,T,T,H,T,T,T,H,T,T

求这个硬币正面朝上的概率有多大?

显然这个概率是0.2。

如何用MLE思想求解?

用MLE的思想

- 抛硬币是二项分布，设正面朝上的概率是p，分布律为

$$P\{X=x\}=p^x(1-p)^{1-x},\ x=0,1$$

- 似然函数为

$$L(p)=\prod_{i=1}^{n} p^{x_i}(1-p)^{1-x_i}=p^{\sum_{i=1}^{n}x_i}(1-p)^{n-\sum_{i=1}^{n}x_i}$$

用MLE的思想（续）

- 对数似然

$$\ln L(p)=(\sum_{i=1}^{n}x_i)\ln p+(n-\sum_{i=1}^{n}x_i)\ln(1-p)$$

- 一阶导数为0

$$\frac{d}{dp}\ln L(p)=\frac{1}{p}\sum_{i=1}^{n}x_i-\frac{1}{1-p}(n-\sum_{i=1}^{n}x_i)$$

$$令\frac{d}{dp}\ln L(p)=0$$

$$\hat{p}=\frac{1}{n}\sum_{i=1}^{n}x_i=\bar{x}$$

Example2（连续型变量）

设X~N（μ, σ^2），μ, σ^2是未知参数，x_1、x_2、…、x_n是来自X的一组样本，求μ, σ^2的最大似然估计

- 概率密度函数

$$f(x;\mu,\sigma^2)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

- 联合概率密度函数

$$p(x_1,x_2,\cdots,x_n)=\prod_{i=1}^{n}\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2\sigma^2}(x_i-\mu)^2}$$

逻辑回归

- 似然函数

$$L(\mu,\sigma^2)=\prod_{i=1}^{n}\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{1}{2\sigma^2}(x_i-\mu)^2}$$

$$=\left(\frac{1}{\sqrt{2\pi\sigma}}\right)^n e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2}$$

- 对数似然

$$\ln L=-\frac{n}{2}\ln(2\pi)-\frac{n}{2}\ln(\sigma^2)-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2$$

逻辑回归

- 偏导数为0

$$\begin{cases} \dfrac{\partial \ln L}{\partial \mu} = 0 \\ \dfrac{\partial \ln L}{\partial \sigma^2} = 0 \end{cases}$$

$$即:\begin{cases} \dfrac{1}{\sigma^2}[\sum_{i=1}^{n} x_i - n\mu] = 0 \\ -\dfrac{\mathrm{n}}{2\sigma^2} + \dfrac{1}{(2\sigma^2)^2}\sum_{i=1}^{n}(x_i - \mu)^2 = 0 \end{cases}$$

- 求解

符合对正态分布的估计

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i = \bar{x}$$

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

逻辑回归

最大似然估计

对数似然函数值最大

似然函数

$$L(x_1,x_2,\cdots,x_n;\theta_1,\theta_2,\cdots\theta_k)=\prod_{i=1}^{n}f(x_i;\theta_1,\theta_2,\cdots\theta_k)$$

对数似然

$$\log L(\theta_1,\theta_2,\cdots\theta_k)=\sum_{i=1}^{n}\log f(x_i;\theta_1,\theta_2,\cdots\theta_k)$$

梯度上升

逻辑回归

令：
$$\begin{aligned} P(y=1 \mid x;\theta) &= h_\theta(x) \\ P(y=0 \mid x;\theta) &= 1-h_\theta(x) \end{aligned}$$

有：$p(y \mid x;\theta) = (h_\theta(x))^y \left(1-h_\theta(x)\right)^{1-y}$

目标函数（对数似然）

$$\begin{aligned} \ell(\theta) &= \log L(\theta) \\ &= \sum_{i=1}^{m} y^{(i)} \log h(x^{(i)}) + (1-y^{(i)}) \log(1-h(x^{(i)})) \end{aligned}$$

目标函数的梯度

$$\frac{\partial}{\partial \theta_j}\ell(\theta) = \left(y - h_\theta(x)\right) x_j$$

和线性回归的梯度对比

$$\left(h_\theta(x) - y\right) x_j$$

梯度上升法伪代码

Repeat until convergence {

$$\theta_j := \theta_j + \alpha \sum_{i=1}^{m} \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)}$$

}

与线性回归的梯度下降法伪代码完全一样，但是二者的梯度是相反的，所以一个叫梯度下降一个叫梯度上升。

实战