第六章

逻辑斯蒂回归与最大熵模型

袁春　清华大学深圳研究生院
李航　华为诺亚方舟实验室

目录

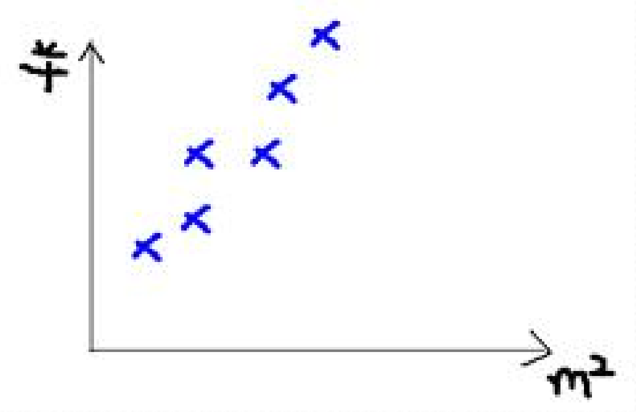
1. 逻辑斯蒂回归模型
2. 最大熵模型
3. 模型学习的最优化方法

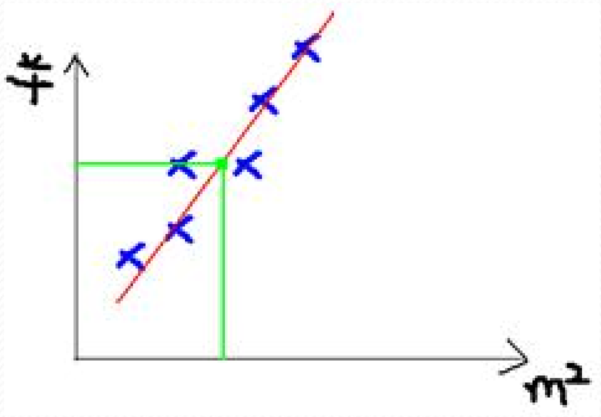
一、逻辑斯蒂回归

- 逻辑斯蒂分布
- 二项逻辑斯蒂回归
- 似然函数
- 模型参数估计
- 多项logistic回归

回归

面积 (m^2)	销售价钱（万元）
123	250
150	320
87	160
102	220
…	…





$$h(x) = h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

回归

- 回归：广义线性模型（generalized linear model）
- 分类：根据因变量的不同
  - 连续：多重线性回归
  - 二项分布：logistic回归
  - poisson分布：poisson回归
  - 负二项分布：负二项回归

逻辑斯蒂分布

- Logistic distribution
- 设X是连续随机变量，X服从Logistic distribution，
- 分布函数：

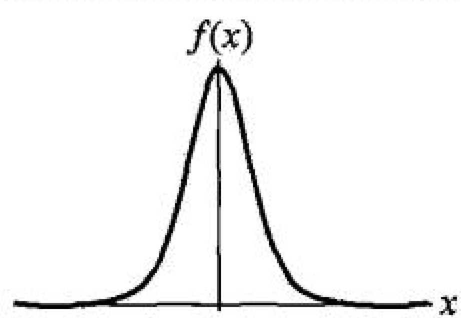
$$F(x)=P(X\leqslant x)=\frac{1}{1+e^{-(x-\mu)/\gamma}}$$

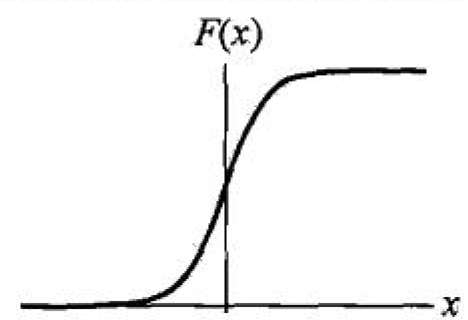
- 密度函数：

$$f(x)=F'(x)=\frac{e^{-(x-\mu)/\gamma}}{\gamma(1+e^{-(x-\mu)/\gamma})^2}$$

- μ 为位置参数，γ大于0为形状参数，(μ,1/2)中心对称

$$F(-x+\mu)-\frac{1}{2}=-F(x-\mu)+\frac{1}{2}$$

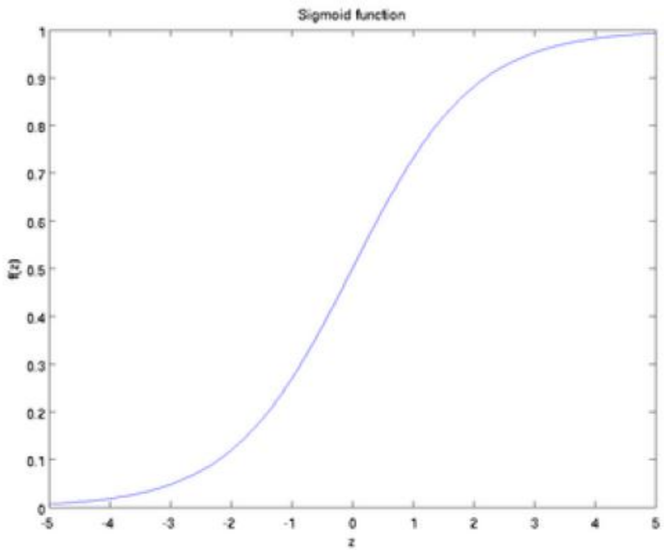




Sigmoid：

$$f(z) = \frac{1}{1 + \exp(-z)}.$$

$$f'(z) = f(z)(1 - f(z))$$

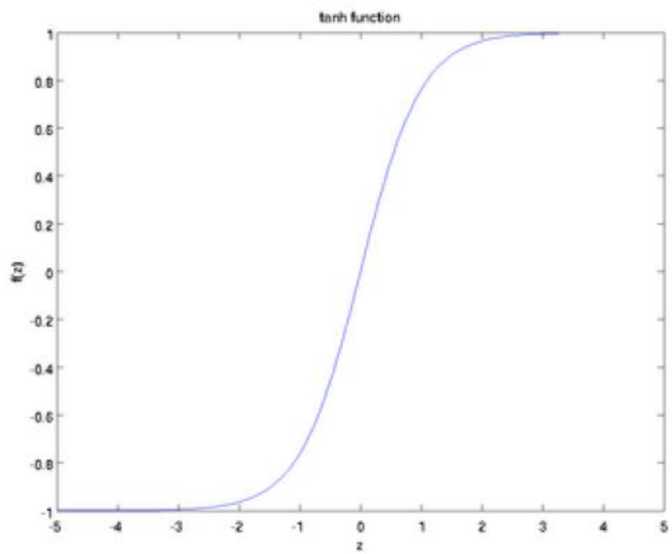


$[0, 1]$

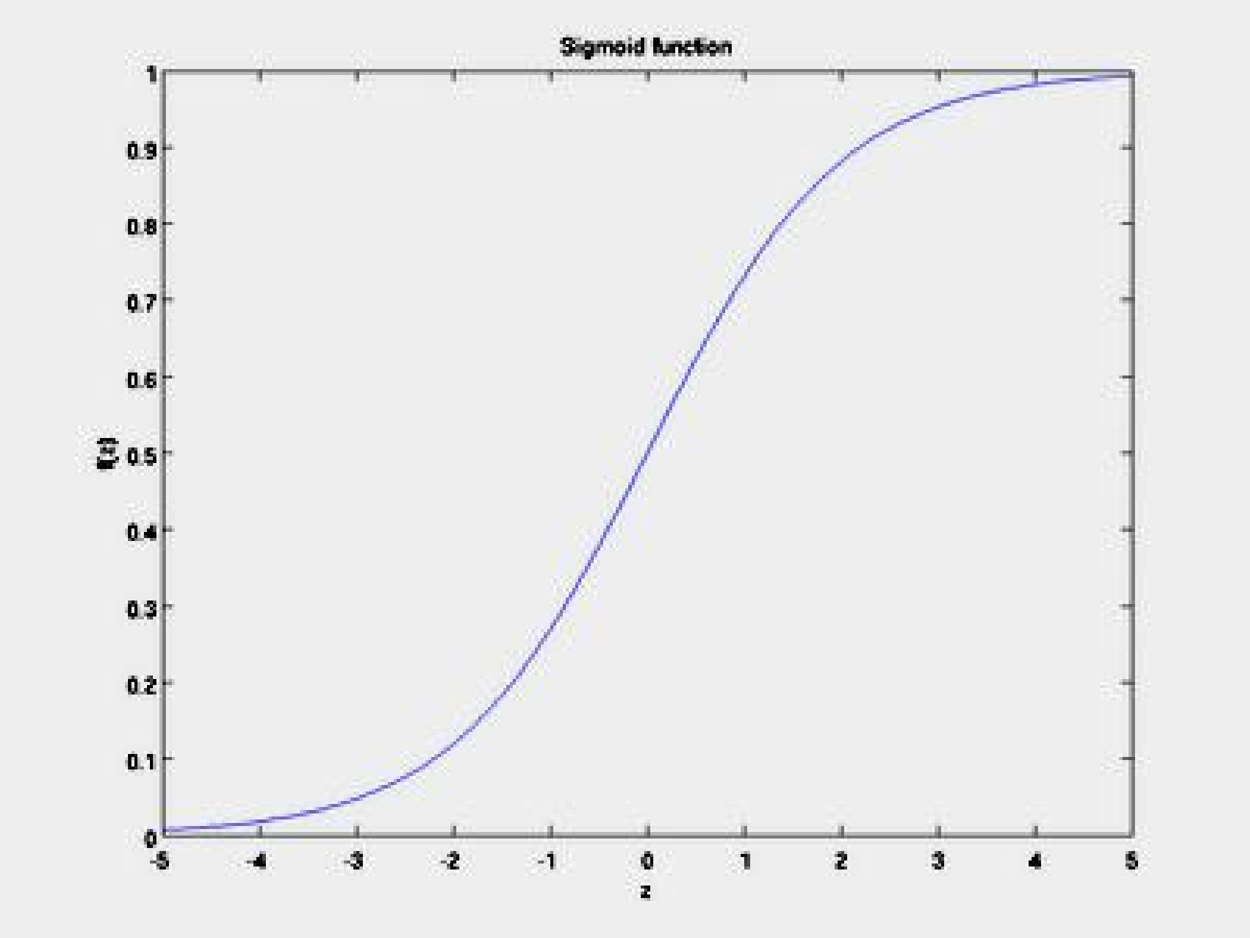
双曲正切函数（tanh）

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

$$f'(z) = 1 - (f(z))^2$$



$[-1, 1]$



Sigmoid function:

$$h_{W,b}(x) = f(W^T x) = f(\textstyle\sum_{i=1}^{3} W_i x_i + b), \quad f(z) = \frac{1}{1 + \exp(-z)}.$$

```
def sigmoid(inX):
    return 1.0/(1+exp(-inX))
```

二项逻辑斯蒂回归

- Binomial logistic regression model
  - 由条件概率P(Y|X)表示的分类模型
  - 形式化为logistic distribution
  - X取实数，Y取值1,0

$$P(Y=1|x)=\frac{\exp(w\cdot x+b)}{1+\exp(w\cdot x+b)}$$

$$P(Y=0|x)=\frac{1}{1+\exp(w\cdot x+b)}$$

$$P(Y=1|x)=\frac{\exp(w\cdot x)}{1+\exp(w\cdot x)}$$

$$P(Y=0|x)=\frac{1}{1+\exp(w\cdot x)}$$

$$w=(w^{(1)},w^{(2)},\cdots,w^{(n)},b)^{\mathrm{T}}$$

$$x=(x^{(1)},x^{(2)},\cdots,x^{(n)},1)^{\mathrm{T}}$$

二项逻辑斯蒂回归

- 事件的几率odds：事件发生与事件不发生的概率之比为

$$\frac{p}{1-p}$$

- 称为事件的发生比(the odds of experiencing an event),
- 对数几率：

$$\operatorname{logit}(p)=\log\frac{p}{1-p}$$

- 对逻辑斯蒂回归

$$\log\frac{P(Y=1\mid x)}{1-P(Y=1\mid x)}=w\cdot x$$

似然函数

- *logistic*分类器是由一组权值系数组成的，最关键的问题就是如何获取这组权值，通过极大似然函数估计获得，并且$Y \sim f(x;w)$

- 似然函数是统计模型中参数的函数。给定输出x时，关于参数θ的似然函数L(θ|x)（在数值上）等于给定参数θ后变量X的概率：$L(\theta|x)=P(X=x|\theta)$
- 似然函数的重要性不是它的取值，而是当参数变化时概率密度函数到底是变大还是变小。
- 极大似然函数：似然函数取得最大值表示相应的参数能够使得统计模型最为合理

似然函数

- 那么对于上述*m*个观测事件，设

$$P(Y=1|x)=\pi(x)\,,\quad P(Y=0|x)=1-\pi(x)$$

- 其联合概率密度函数，即似然函数为：

$$\prod_{i=1}^{N}[\pi(x_i)]^{y_i}[1-\pi(x_i)]^{1-y_i}$$

- 目标：求出使这一似然函数的值最大的参数估，$w_1,w_2,\ldots,w_n$，使得L(w)取得 最大值。
- 对L(w)取对数：

模型参数估计

- 对数似然函数

$$
\begin{aligned}
L(w) &= \sum_{i=1}^{N}[y_i \log \pi(x_i) + (1-y_i)\log(1-\pi(x_i))] \\
&= \sum_{i=1}^{N}\left[y_i \log \frac{\pi(x_i)}{1-\pi(x_i)} + \log(1-\pi(x_i))\right] \\
&= \sum_{i=1}^{N}[y_i(w \cdot x_i) - \log(1+\exp(w \cdot x_i))]
\end{aligned}
$$

- 对L(w)求极大值，得到w的估计值。
- 通常采用梯度下降法及拟牛顿法，学到的模型：

$$P(Y=1 \mid x) = \frac{\exp(\hat{w} \cdot x)}{1+\exp(\hat{w} \cdot x)} \qquad P(Y=0 \mid x) = \frac{1}{1+\exp(\hat{w} \cdot x)}$$

多项logistic回归

- 设Y的取值集合为

$$\{1,2,\cdots,K\}$$

- 多项logistic回归模型

$$P(Y=k\mid x)=\frac{\exp(w_k\cdot x)}{1+\sum_{k=1}^{K-1}\exp(w_k\cdot x)}, \quad k=1,2,\cdots,K-1$$

$$P(Y=K\mid x)=\frac{1}{1+\sum_{k=1}^{K-1}\exp(w_k\cdot x)}$$

二、最大熵模型

- 最大熵原理
- 最大熵模型的定义
- 最大熵模型的学习
- 极大似然估计

最大熵原理

- 最大熵模型(Maximum Entropy Model)由最大熵原理推导实现。
- 最大熵原理：
- 学习概率模型时，在所有可能的概率模型(分布)中，熵最大的模型是最好的模型，表述为在满足约束条件的模型集合中选取熵最大的模型。
- 假设离散随机变量X的概率分布是P(X)，
- 熵：

$$H(P) = -\sum_{x} P(x) \log P(x)$$

- 且：

$$0 \leqslant H(P) \leqslant \log |X|$$

- |X|是X的取值个数，X均匀分布时右边等号成立。

例子：

- 假设随机变量X有5个取值{A,B,C,D,E},估计各个值的概率。
- 解：满足　　P(A)+P(B)+P(C)+P(D)+P(E)=1
- 等概率估计：$P(A)=P(B)=P(C)=P(D)=P(E)=\frac{1}{5}$
- 加入一些先验：

$$P(A)+P(B)=\frac{3}{10}$$

$$P(A)+P(B)+P(C)+P(D)+P(E)=1$$

- 于是：

$$P(A)=P(B)=\frac{3}{20}$$

$$P(C)=P(D)=P(E)=\frac{7}{30}$$

例子：

- 假设随机变量X有5个取值{A,B,C,D,E},估计各个值的概率。
- 解：满足 P(A)+P(B)+P(C)+P(D)+P(E)=1
- 等概率估计：$P(A)=P(B)=P(C)=P(D)=P(E)=\frac{1}{5}$
- 加入一些先验：

$$P(A)+P(B)=\frac{3}{10}$$

$$P(A)+P(B)+P(C)+P(D)+P(E)=1$$

- 于是：

$$P(A)=P(B)=\frac{3}{20}$$

$$P(C)=P(D)=P(E)=\frac{7}{30}$$

再加入约束：

$$P(A)+P(C)=\frac{1}{2}$$

$$P(A)+P(B)=\frac{3}{10}$$

$$P(A)+P(B)+P(C)+P(D)+P(E)=1$$

最大熵原理

- X和Y分别是输入和输出的集合，这个模型表示的是对于给定的输入X，以条件概率P(Y|X)输出Y.
- 给定数据集： $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$
- 联合分布P(Y|X)的经验分布，边缘分布P(X)的经验分布：

$$\tilde{P}(X,Y) \Rightarrow \tilde{P}(X=x,Y=y)=\frac{\nu(X=x,Y=y)}{N}$$

$$\tilde{P}(X) \Rightarrow \tilde{P}(X=x)=\frac{\nu(X=x)}{N}$$

- 特征函数：

$$f(x,y)=\begin{cases}1, & x\text{与}y\text{满足某一事实}\\ 0, & \text{否则}\end{cases}$$

最大熵原理

- 特征函数f(x,y)关于经验分布$\tilde{P}(X,Y)$ 的期望值:

$$E_{\tilde{P}}(f)=\sum_{x,y}\tilde{P}(x,y)f(x,y)$$

- 特征函数f(x,y)关于模型P(Y|X)与经验分布$\tilde{P}(X)$ 的期望值:

$$E_{P}(f)=\sum_{x,y}\tilde{P}(x)P(y|x)f(x,y)$$

- 如果模型能够获取训练数据中的信息，那么就可以假设这两个期望值相等，即

$$E_P(f)=E_{\tilde{P}}(f) \Rightarrow \sum_{x,y}\tilde{P}(x)P(y|x)f(x,y)=\sum_{x,y}\tilde{P}(x,y)f(x,y)$$

- 假设有n个特征函数: $f_i(x,y)$，$i=1,2,\cdots,n$

最大熵模型的定义

- 定义：
- 假设满足所有约束条件的模型集合为：

$$\mathcal{C} \equiv \{P \in \mathcal{P} \mid E_P(f_i) = E_{\tilde{P}}(f_i),\quad i = 1, 2, \cdots, n\}$$

- 定义在条件概率分布P(Y|X)上的条件熵：

$$H(P) = -\sum_{x,y} \tilde{P}(x) P(y \mid x) \log P(y \mid x)$$

- 则模型集合C中条件熵H(P)最大的模型称为最大熵模型

最大熵模型的学习

- 最大熵模型的学习可以形式化为约束最优化问题。
- 对于给定的数据集以及特征函数：$f_i(x,y)$
- 最大熵模型的学习等价于约束最优化问题:

$$\max_{P\in C} \quad H(P) = -\sum_{x,y} \tilde{P}(x)P(y|x)\log P(y|x)$$

$$\text{s.t.} \quad E_P(f_i) = E_{\tilde{P}}(f_i), \quad i=1,2,\cdots,n$$

$$\sum_y P(y|x) = 1$$

$$\min_{P\in C} \quad -H(P) = \sum_{x,y} \tilde{P}(x)P(y|x)\log P(y|x)$$

$$\text{s.t.} \quad E_P(f_i) - E_{\tilde{P}}(f_i) = 0, \quad i=1,2,\cdots,n$$

$$\sum_y P(y|x) = 1$$

最大熵模型的学习

- 这里，将约束最优化的原始问题转换为无约束最优化的对偶问题,通过求解对偶问题求解原始问题:
- 引进拉格朗日乘子，定义拉格朗日函数:

$$
\begin{aligned}
L(P,w) &\equiv -H(P) + w_0\left(1-\sum_y P(y\mid x)\right) + \sum_{i=1}^{n} w_i\left(E_{\tilde{P}}(f_i) - E_P(f_i)\right) \\
&= \sum_{x,y}\tilde{P}(x)P(y\mid x)\log P(y\mid x) + w_0\left(1-\sum_y P(y\mid x)\right) \\
&\quad + \sum_{i=1}^{n} w_i\left(\sum_{x,y}\tilde{P}(x,y)f_i(x,y) - \sum_{x,y}\tilde{P}(x)P(y\mid x)f_i(x,y)\right)
\end{aligned}
$$

- 最优化原始问题 到 对偶问题:

$$
\min_{P\in C}\max_{w} L(P,w) \Rightarrow \max_{w}\min_{P\in C} L(P,w)
$$

最大熵模型的学习

- 最优化原始问题 到 对偶问题：

$$\min_{P\in C}\max_{w} L(P,w) \Rightarrow \max_{w}\min_{P\in C} L(P,w)$$

- L(P,w)是P的凸函数，解的等价性（证明部分在SVM部分介绍）
- 先求极小化问题： $\min_{P\in C} L(P,w)$ 是w的函数,

$$\Psi(w)=\min_{P\in C} L(P,w)=L(P_w,w)$$

$$P_w=\arg\min_{P\in C} L(P,w)=P_w(y\,|\,x)$$

最大熵模型的学习

- 求L(P,w)对P(y|x)的偏导数：

$$\frac{\partial L(P,w)}{\partial P(y|x)} = \sum_{x,y}\tilde{P}(x)\left(\log P(y|x)+1\right)-\sum_{y}w_0-\sum_{x,y}\left(\tilde{P}(x)\sum_{i=1}^{n}w_i f_i(x,y)\right)$$

$$= \sum_{x,y}\tilde{P}(x)\left(\log P(y|x)+1-w_0-\sum_{i=1}^{n}w_i f_i(x,y)\right)$$

- 得：

$$P(y|x)=\exp\left(\sum_{i=1}^{n}w_i f_i(x,y)+w_0-1\right)=\frac{\exp\left(\sum_{i=1}^{n}w_i f_i(x,y)\right)}{\exp(1-w_0)}$$

最大熵模型的学习

- 由：$$\sum_y P(y|x)=1$$
- 得：$$P_w(y|x)=\frac{1}{Z_w(x)}\exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right) \qquad (6.22)$$

规范化因子：$$Z_w(x)=\sum_y \exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right) \qquad (6.23)$$

模型 $P_w=P_w(y|x)$ 就是最大熵模型

求解对偶问题外部的极大化问题：

$$\max_w \Psi(w) \qquad w^*=\arg\max_w \Psi(w) \qquad P^*=P_{w^*}=P_{w^*}(y|x)$$

例子：

- 原例子中的最大熵模型：

$$\min \quad -H(P)=\sum_{i=1}^{5} P(y_i)\log P(y_i)$$

$$\text{s.t.} \quad P(y_1)+P(y_2)=\tilde{P}(y_1)+\tilde{P}(y_2)=\frac{3}{10}$$

$$\sum_{i=1}^{5} P(y_i)=\sum_{i=1}^{5} \tilde{P}(y_i)=1$$

$$L(P,w)=\sum_{i=1}^{5} P(y_i)\log P(y_i)+w_1\left(P(y_1)+P(y_2)-\frac{3}{10}\right)+w_0\left(\sum_{i=1}^{5} P(y_i)-1\right)$$

$$\max_{w}\min_{P} L(P,w)$$

例子：

$$\frac{\partial L(P,w)}{\partial P(y_1)}=1+\log P(y_1)+w_1+w_0$$

$$\frac{\partial L(P,w)}{\partial P(y_2)}=1+\log P(y_2)+w_1+w_0$$

$$\frac{\partial L(P,w)}{\partial P(y_3)}=1+\log P(y_3)+w_0$$

$$\frac{\partial L(P,w)}{\partial P(y_4)}=1+\log P(y_4)+w_0$$

$$\frac{\partial L(P,w)}{\partial P(y_5)}=1+\log P(y_5)+w_0$$

解得：

$$P(y_1)=P(y_2)=\mathrm{e}^{-w_1-w_0-1}$$

$$P(y_3)=P(y_4)=P(y_5)=\mathrm{e}^{-w_0-1}$$

例子：

$$\min_{P} L(P,w) = L(P_w,w) = -2e^{-w_1-w_0-1} - 3e^{-w_0-1} - \frac{3}{10}w_1 - w_0$$

- 得：

$$\max_{w}\ L(P_w,w) = -2e^{-w_1-w_0-1} - 3e^{-w_0-1} - \frac{3}{10}w_1 - w_0$$

- 对wi求偏导并令为o：

$$e^{-w_1-w_0-1} = \frac{3}{20}$$

$$e^{-w_0-1} = \frac{7}{30}$$

$$\Rightarrow$$

$$P(y_1) = P(y_2) = \frac{3}{20}$$

$$P(y_3) = P(y_4) = P(y_5) = \frac{7}{30}$$

极大似然估计

- 最大熵模型就是(6.22),(6.23)表示的条件概率分布，
- 证明：对偶函数的极大化等价于最大熵模型的极大似然估计.
- 已知训练数据的经验概率分布$\tilde{P}(X,Y)$，条件概率分布P(Y|X)的对数似然函数表示为：

$$L_{\tilde{P}}(P_w)=\log\prod_{x,y}P(y\mid x)^{\tilde{P}(x,y)}=\sum_{x,y}\tilde{P}(x,y)\log P(y\mid x)$$

$$=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}w_i f_i(x,y)-\sum_{x,y}\tilde{P}(x,y)\log Z_w(x)$$

$$=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}w_i f_i(x,y)-\sum_{x}\tilde{P}(x)\log Z_w(x)$$

极大似然估计

- 而：

$$
\begin{aligned}
\Psi(w) &= \sum_{x,y} \tilde{P}(x) P_w(y \mid x) \log P_w(y \mid x) \\
&\quad + \sum_{i=1}^{n} w_i \left(\sum_{x,y} \tilde{P}(x,y) f_i(x,y) - \sum_{x,y} \tilde{P}(x) P_w(y \mid x) f_i(x,y) \right) \\
&= \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} w_i f_i(x,y) + \sum_{x,y} \tilde{P}(x) P_w(y \mid x) \left(\log P_w(y \mid x) - \sum_{i=1}^{n} w_i f_i(x,y) \right) \\
&= \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} w_i f_i(x,y) - \sum_{x,y} \tilde{P}(x) P_w(y \mid x) \log Z_w(x) \\
&= \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} w_i f_i(x,y) - \sum_{x} \tilde{P}(x) \log Z_w(x)
\end{aligned}
$$

极大似然估计

- 最大熵模型与逻辑斯谛回归模型有类似的形式，它们又称为对数线性模型(log linear model). 模型学习就是在给定的训练数据条件下对模型进行极大似然估计或正则化的极大似然估计。

三、模型学习的最优化算法

- 最优化算法简介
- 梯度下降法
- 无约束最优化问题 - 牛顿法、拟牛顿法、DFP算法、BFGS算法

最优化算法简介

- 逻辑斯谛回归模型、最大熵模型学习归结为以似然函数为目标函数的最优化问题，通常通过迭代算法求解，它是光滑的凸函数，因此多种最优化的方法都适用。
- 常用的方法有：
  - 改进的迭代尺度法
  - 梯度下降法
  - 牛顿法
  - 拟牛顿法

梯度下降法

- 梯度下降法(gradient descent)
- 最速下降法(steepest descent)
- 梯度下降法是一种迭代算法.选取适当的初值$x^{(0)}$，不断迭代，更新x的值，进行目标函数的极小化，直到收敛。由于负梯度方向是使函数值下降最快的方向，在迭代的每一步，以负梯度方向更新x的值，从而达到减少函数值的目的.

梯度下降法

- 假设f(x)具有一阶连续偏导数的函数：

$$\min_{x\in\mathbb{R}^n} f(x)$$

- 一阶泰勒展开：$f(x)=f(x^{(k)})+g_k^{\mathrm{T}}(x-x^{(k)})$

- f(x)在$x^{(k)}$的梯度值：

$$g_k=g(x^{(k)})=\nabla f(x^{(k)})$$

$$x^{(k+1)}\leftarrow x^{(k)}+\lambda_k p_k$$

- 负梯度方向：

$$p_k=-\nabla f(x^{(k)})$$

$$f(x^{(k)}+\lambda_k p_k)=\min_{\lambda\geq 0} f(x^{(k)}+\lambda p_k)$$

无约束最优化问题

- 牛顿法（Newton method）
- 拟牛顿法（quasi Newton method）
- 有收敛速度快的优点.

- 牛顿法是迭代算法，每一步需要求解目标函数的海赛矩阵的逆矩阵，计算比较复杂。
- 拟牛顿法通过正定矩阵近似海赛矩阵的逆矩阵或海赛矩阵，简化了这一计算过程。

牛顿法

- 无约束最优化问题：

$$\min_{x\in\mathbf{R}^n} f(x)$$

- 假设f(x)具有二阶连续偏导数，若第k次迭代值为$x^{(k)}$，则可将f(x)在$x^{(k)}$附近进行二阶泰勒展开:

$$f(x)=f(x^{(k)})+g_k^{\mathrm{T}}(x-x^{(k)})+\frac{1}{2}(x-x^{(k)})^{\mathrm{T}}H(x^{(k)})(x-x^{(k)})$$

B$g_k=g(x^{(k)})=\nabla f(x^{(k)})$

- $H(x^{(k)})$ 是f(x)的梯度向量在$x^{(k)}$的值
- 是f(x) $x^{(k)}$的值

$$H(x)=\left[\frac{\partial^2 f}{\partial x_i \partial x_j}\right]_{n\times n}$$

牛顿法

- 函数f(x)有极值的必要条件是:在极值点处一阶导数为o, 即梯度向量为o.
- 特别是当$H(x^{(k)})$是正定矩阵时，函数f(x)的极值为极小值.
- 利用条件： $\nabla f(x)=0$
- 设迭代从$x^{(k)}$开始，求目标函数的极小点，

$$\nabla f(x^{(k+1)})=0$$

$$\nabla f(x)=g_k+H_k(x-x^{(k)})$$

$$H_k=H(x^{(k)}) \quad g_k+H_k(x^{(k+1)}-x^{(k)})=0$$

牛顿法

$$x^{(k+1)} = x^{(k)} - H_k^{-1} g_k \qquad (B.8)$$

$$x^{(k+1)} = x^{(k)} + p_k$$

$$H_k p_k = -g_k$$

牛顿法

- 算法步骤：

输入：目标函数 $f(x)$，梯度 $g(x)=\nabla f(x)$

海赛矩阵 $H(x)$，精度要求 ε

输出：$f(x)$ 的极小点 x^*.

（1）取初始点 $x^{(0)}$，置 $k=0$

（2）计算 $g_k=g(x^{(k)})$

（3）若 $\|g_k\|<\varepsilon$，则停止计算，得近似解 $x^*=x^{(k)}$

（4）计算 $H_k=H(x^{(k)})$，并求 p_k

$$H_k p_k=-g_k$$

求逆

（5）置 $x^{(k+1)}=x^{(k)}+p_k$

（6）置 $k=k+1$，转（2）.

拟牛顿法

- 考虑用一个n阶矩阵$G_k=G(x^{(k)})$来近似代替 $H_k^{-1}=H^{-1}(x^{(k)})$

$$\nabla f(x)=g_k+H_k(x-x^{(k)})$$

$$g_{k+1}-g_k=H_k(x^{(k+1)}-x^{(k)})$$

记 $y_k=g_{k+1}-g_k$， $\delta_k=x^{(k+1)}-x^{(k)}$

- 拟牛顿条件： $\boxed{y_k=H_k\delta_k}$ $\boxed{H_k^{-1}y_k=\delta_k}$
- 由：

$$x=x^{(k)}+\lambda p_k=x^{(k)}-\lambda H_k^{-1}g_k$$

$$f(x)=f(x^{(k)})+g_k^{\mathrm{T}}(x-x^{(k)})+\frac{1}{2}(x-x^{(k)})^{\mathrm{T}}H(x^{(k)})(x-x^{(k)})$$

$$f(x)=f(x^{(k)})-\lambda g_k^{\mathrm{T}}H_k^{-1}g_k$$

拟牛顿法

- 如果H_k是正定的, H_k^{-1}也是正定的，那么可以保证牛顿法搜索方向Pk是下降方向,因为搜索方向 $p_k=-\lambda g_k$
- 由B.8得： $x=x^{(k)}+\lambda p_k=x^{(k)}-\lambda H_k^{-1}g_k$
- 由B.2得： $f(x)=f(x^{(k)})-\lambda g_k^{\mathrm{T}}H_k^{-1}g_k$

因 H_k^{-1} 正定，故有 $g_k^{\mathrm{T}}H_k^{-1}g_k>0$

当 λ 为一个充分小的正数时，总有 $f(x)<f(x^{(k)})$

- 将G_k作为 H_k^{-1} 的近似 $G_{k+1}y_k=\delta_k$ ，拟牛顿条件

拟牛顿法

- 在每次迭代中可以选择更新矩阵

$$G_{k+1} = G_k + \Delta G_k$$

- Broyden类优化算法：
  - DFP(Davidon-Fletcher-Powell)算法(DFP algorithm)
  - BFGS(Broyden-Fletcher-Goldfarb-Shanno)算法(BFGS algorithm)
  - Broyden类算法(Broyden's algorithm)

BFGS(Broyden-Fletcher-Goldfarb-Shanno)算法

- 可以考虑用G_k逼近海赛矩阵的逆矩阵H^{-1}，也可以考虑用B_k逼近海赛矩阵H, 这时，相应的拟牛顿条件是：

$$B_{k+1}\delta_k = y_k$$

- 用同样的方法得到另一迭代公式.首先令

$$B_{k+1} = B_k + P_k + Q_k$$

$$B_{k+1}\delta_k = B_k\delta_k + P_k\delta_k + Q_k\delta_k$$

- 考虑使P_k和Q_k满足：

$$P_k\delta_k = y_k \qquad Q_k\delta_k = -B_k\delta_k$$

- B_{k+1}的迭代公式：
- $$B_{k+1} = B_k + \frac{y_k y_k^{\mathrm{T}}}{y_k^{\mathrm{T}}\delta_k} - \frac{B_k\delta_k\delta_k^{\mathrm{T}}B_k}{\delta_k^{\mathrm{T}}B_k\delta_k} \qquad \text{B.30}$$

改进的迭代尺度法

- 改进的迭代尺度法(improved iterative scaling,IIS)
- 由最大熵模型

$$P_w(y|x)=\frac{1}{Z_w(x)}\exp\left(\sum_{i=1}^{n}w_i f_i(x,y)\right) \qquad Z_w(x)=\sum_y \exp\left(\sum_{i=1}^{n}w_i f_i(x,y)\right)$$

- 对数似然函数

$$L(w)=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}w_i f_i(x,y)-\sum_x \tilde{P}(x)\log Z_w(x)$$

- 求对数似然函数的极大值 $\hat{w}$
- IIS思路：假设 $w=(w_1,w_2,\cdots,w_n)^{\mathrm{T}}$ 希望找到一个新的参数向量 $w+\delta=(w_1+\delta_1,w_2+\delta_2,\cdots,w_n+\delta_n)^{\mathrm{T}}$ 使得模型的对数似然函数值增大，如果有参数向量更新方法，那么就可以重复使用这一方法，直至找到对数似然函数的最大值。

改进的迭代尺度法

$$L(w+\delta)-L(w)=\sum_{x,y}\tilde{P}(x,y)\log P_{w+\delta}(y\mid x)-\sum_{x,y}\tilde{P}(x,y)\log P_{w}(y\mid x)$$

$$=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}\delta_i f_i(x,y)-\sum_{x}\tilde{P}(x)\log\frac{Z_{w+\delta}(x)}{Z_w(x)}$$

- 利用 $-\log\alpha \geqslant 1-\alpha,\quad \alpha>0$

$$L(w+\delta)-L(w)\geqslant\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}\delta_i f_i(x,y)+1-\sum_{x}\tilde{P}(x)\frac{Z_{w+\delta}(x)}{Z_w(x)}$$

$$=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}\delta_i f_i(x,y)+1-\sum_{x}\tilde{P}(x)\sum_{y}P_w(y\mid x)\exp\sum_{i=1}^{n}\delta_i f_i(x,y)$$

$$A(\delta\mid w)=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}\delta_i f_i(x,y)+1-\sum_{x}\tilde{P}(x)\sum_{y}P_w(y\mid x)\exp\sum_{i=1}^{n}\delta_i f_i(x,y)$$

改进的迭代尺度法

- 于是有 $L(w+\delta)-L(w)\geq A(\delta\mid w)$
- 如果能找到适当的δ使下界A(δ|w)提高，那么对数似然函数也会提高。
- δ是一个向量，含多个变量，一次只优化一个变量δ_i
- 引进一个量$f^{\#}(x,y)$, $f^{\#}(x,y)=\sum_i f_i(x,y)$
-
- $f_i(x,y)$是二值函数， $f^{\#}(x,y)$表示所有特征在(x,y)出现的次数。

$$A(\delta\mid w)=\sum_{x,y}\tilde{P}(x,y)\sum_{i=1}^{n}\delta_i f_i(x,y)+1-\sum_x\tilde{P}(x)\sum_y P_w(y\mid x)\exp\left(f^{\#}(x,y)\sum_{i=1}^{n}\frac{\delta_i f_i(x,y)}{f^{\#}(x,y)}\right)$$

改进的迭代尺度法

- 利用指数函数的凸性，以及

$$\frac{f_i(x,y)}{f^{\#}(x,y)} \geqslant 0 \text{ 且 } \sum_{i=1}^{n} \frac{f_i(x,y)}{f^{\#}(x,y)} = 1$$

- 根据Jensen不等式：

$$\exp\left(\sum_{i=1}^{n} \frac{f_i(x,y)}{f^{\#}(x,y)} \delta_i f^{\#}(x,y) \right) \leqslant \sum_{i=1}^{n} \frac{f_i(x,y)}{f^{\#}(x,y)} \exp(\delta_i f^{\#}(x,y))$$

$$A(\delta \mid w) \geqslant \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} \delta_i f_i(x,y) + 1 - \sum_{x} \tilde{P}(x) \sum_{y} P_w(y \mid x) \sum_{i=1}^{n} \left(\frac{f_i(x,y)}{f^{\#}(x,y)} \right) \exp(\delta_i f^{\#}(x,y))$$

$$B(\delta \mid w) = \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} \delta_i f_i(x,y) + 1 - \sum_{x} \tilde{P}(x) \sum_{y} P_w(y \mid x) \sum_{i=1}^{n} \left(\frac{f_i(x,y)}{f^{\#}(x,y)} \right) \exp(\delta_i f^{\#}(x,y))$$

改进的迭代尺度法

- 于是得到

$$L(w+\delta)-L(w)\geq B(\delta\mid w)$$

- $B(\delta\mid w)$ 是对数似然函数改变量的一个新的下界
- 对δ_i求偏导：

$$\frac{\partial B(\delta\mid w)}{\partial \delta_i}=\sum_{x,y}\tilde{P}(x,y)f_i(x,y)-\sum_{x}\tilde{P}(x)\sum_{y}P_w(y\mid x)f_i(x,y)\exp(\delta_i f^{\#}(x,y))$$

- 令偏导数为0，得到：

$$\sum_{x,y}\tilde{P}(x)P_w(y\mid x)f_i(x,y)\exp(\delta_i f^{\#}(x,y))=E_{\tilde{p}}(f_i)$$

- 依次对δ_i解方程。

改进的迭代尺度法

- 算法
- 输入：特征函数f₁,f₂…fn;经验分布 $\tilde{P}(X,Y)$,模型 $P_w(y|x)$
- 输出：最优参数w_i*；最优模型P_{w^*}

（1）对所有 $i\in\{1,2,\cdots,n\}$，取初值 $w_i=0$

（2）对每一 $i\in\{1,2,\cdots,n\}$：

（a）令 δ_i 是方程

$$\sum_{x,y}\tilde{P}(x)P(y|x)f_i(x,y)\exp(\delta_i f^{\#}(x,y))=E_{\tilde{p}}(f_i)$$

的解，这里 $f^{\#}(x,y)=\sum_{i=1}^{n}f_i(x,y)$

（b）更新 w_i 值：$w_i\leftarrow w_i+\delta_i$

关键

（3）如果不是所有 w_i 都收敛，重复步 (2)

改进的迭代尺度法

如果 $f^{\#}(x,y)$ 是常数 M

$$\delta_i = \frac{1}{M}\log\frac{E_{\tilde{P}}(f_i)}{E_P(f_i)}$$

如果 $f^{\#}(x,y)$ 不是常数　牛顿法

$$\sum_{x,y}\tilde{P}(x)P_w(y\mid x)f_i(x,y)\exp(\delta_i f^{\#}(x,y)) = E_{\tilde{P}}(f_i)$$

$$g(\delta_i) = 0$$

$$\delta_i^{(k+1)} = \delta_i^{(k)} - \frac{g(\delta_i^{(k)})}{g'(\delta_i^{(k)})}$$

拟牛顿法

- 最大熵模型：

$$P_w(y \mid x) = \frac{\exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right)}{\sum_y \exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right)}$$

- 目标函数：

$$\min_{w \in \mathbf{R}^n} \quad f(w) = \sum_x \tilde{P}(x) \log \sum_y \exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right) - \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} w_i f_i(x,y)$$

- 梯度：

$$g(w) = \left(\frac{\partial f(w)}{\partial w_1}, \frac{\partial f(w)}{\partial w_2}, \cdots, \frac{\partial f(w)}{\partial w_n}\right)^{\mathrm{T}}$$

$$\frac{\partial f(w)}{\partial w_i} = \sum_{x,y} \tilde{P}(x) P_w(y \mid x) f_i(x,y) - E_{\tilde{P}}(f_i), \quad i = 1, 2, \cdots, n$$

Q&A?