逻辑回归

peghoty

peghoty@163.com

目　录

§1 回归分析

§1.1 基本概念

回归分析 ([1]), 是研究一组变量 $\{y_1, y_2, \cdots, y_k\}$ 与另一组变量 $\{x_1, x_2, \cdots, x_n\}$ 相互依赖关系的一种统计分析方法. 通常称 $\{y_1, y_2, \cdots, y_k\}$ 为**因变量**, 称 $\{x_1, x_2, \cdots, x_n\}$ 为**自变量**. 当 $n > 1$ 时称为**多元回归分析**, 当 $k > 1$ 时称为**多重回归分析**.

考虑 $k = 1$ 的一般情形, 此时只有一个因变量 $y = y_1$, 它可以表示成 $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ 的函数 (这个函数也称为 **hypothesis 函数**), 假设函数的形式 $y = h(\mathbf{x})$ 是**已知**的, 但其中包含若干未知的参数*. 回归分析的目标则是: 根据一组**样本数据** $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, 建立数学模型来估计 $y = h(\mathbf{x})$ 中的未知参数. 当 h 为线性函数时, 我们称之为 (n 元) **线性回归**; 当 h 为非线性函数时, 称之为**非线性回归**. 本文主要讨论**线性回归**.

当然, 回归分析除了估计未知参数 (数据拟合) 外, 也包含以下其他内容:

1. 对建立的关系式的可信程度进行检验.
2. 在包含多个自变量的函数关系中, 对自变量作**显著性分析**, 判断哪个 (或哪些) 自变量对因变量的影响是显著的.
3. 对于样本数据外的其他数据, 利用所求关系式**预测**其函数值.

一般来说, 回归分析的大致流程如下: 通过规定因变量和自变量来确定变量之间的因果关系, 建立回归模型, 并根据实测数据 (样本数据) 来求解模型中的各个未知参数, 然后评价回归模型是否能够很好地拟合实测数据. 如果能够很好地拟合, 则可以根据所求关系式作进一步预测.

注意, 前面我们有介绍**相关性分析**, 那么相关性分析和回归分析有什么区别呢? 前者研究的是现象之间是否相关、相关的方向和密切程度, 一般不区别谁是自变量谁是因变量; 而后者则要分析现象之间相关的具体形式, 确定其因果关系, 并用数学模型来表现其具体关系.

§1.2 最小二乘法

估计参数最常用的方法是**最小二乘法**, 最小二乘法是一个古老的方法, 它于 1809 年由**高斯**提出并发表在其著作《天体运动论》中†. 本小节我们将分别通过一个一元线性模型和一个多元线性模型来介绍这个方法.

例 1.1 (**一元线性模型**) *已知样本数据 (x_i, y_i), $i = 1, 2, \cdots, m$, 将这些数据在坐标平面上标出来(这样的图也叫做**散点图**), 发现它们大致在一条直线上, 请确定参数 a, b, 使得函数 $y = h(x) = ax + b$ 可拟合所有的样本数据.*

*更一般的函数关系为 $y = h(\mathbf{x}) + \epsilon$, 其中 ϵ 为**随机误差**.

†法国科学家**勒让德** 于 1806 年独立发现 "最小二乘法", 但因不为世人所知而默默无闻. 勒让德曾与高斯为谁最早创立最小二乘法原理发生争执.

记 $\delta_i = y_i - h(x_i)$ 表示用函数 h 近似表示 x_i 和 y_i 的关系时所产生的**误差**, 利用 δ_i 可以定义**整体误差**

$$\delta := \delta(a,b) = \sum_{i=1}^{m} \delta_i^2, \tag{1.1}$$

为使得拟合效果最好, 求得的参数 a, b 应使得整体误差 δ 最小. 这种利用误差平方和最小来计算未知参数的方法即为**最小二乘法**, 这里的 "二乘" 表示平方的意思, 因为 δ_i 可正可负, 相加会出现抵消, 因此公式中采用 δ_i^2 的形式. 然而, 读者可能要问: 为什么不用 $|\delta_i|$ 的形式呢, 不是更简单吗? 那是因为 δ_i^2 的求导运算比 $|\delta_i|$ 要来得更方便.

注 1.1 关于这个问题, 苏冉旭在其博客 ([2]) 中还给出了另一个更深层次的分析, 这里摘录如下, 供读者参考.

"其实我更喜欢的一个解释是, 如果我们认为实际的点在偏离预测值时是按 '正态分布' 偏离的, 那么就应该最小化平方和, 而不是最小化绝对值的和或其它的和. 因为正态分布概率大概长这个样子:

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}, \tag{1.2}$$

注意其中的 $(\frac{x-\mu}{\sigma})^2$, 它是平方不是绝对值. 让误差平方最小就是让正态分布的 $P(x)$ 最大, 也就是让当前的误差解释起来最为合理, 直观上不难理解吧?"

注 1.2 整体误差是我们最小化的目标函数, 更一般地, 也将其称为**成本函数**或**代价函数**, 其表达式有时被写成

$$\delta = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{2} \delta_i^2, \tag{1.3}$$

但这些只是形式上的差别 (系数 $\frac{1}{2}$ 主要是为了求导方便, $\frac{1}{m}$ 做一次平均), 对后面的推导过程并没有影响.

那么, 为使得整体误差 δ 达到最小, 参数 a, b 该如何计算呢? 下面给出具体计算方法.

利用**多元函数的极值定理**, 若 $(\widetilde{a}, \widetilde{b})$ 为函数 $\delta(a,b)$ 的极小点, 则必有

$$\frac{\partial \delta}{\partial a}(\widetilde{a}, \widetilde{b}) = 0, \quad \frac{\partial \delta}{\partial b}(\widetilde{a}, \widetilde{b}) = 0, \tag{1.4}$$

计算偏导数, 得

$$\begin{cases} \frac{\partial \delta}{\partial a}(\widetilde{a}, \widetilde{b}) = \sum_{i=1}^{m} 2(y_i - \widetilde{a}x_i - \widetilde{b}) \cdot (-x_i) = 0 \\ \frac{\partial \delta}{\partial b}(\widetilde{a}, \widetilde{b}) = \sum_{i=1}^{m} 2(y_i - \widetilde{a}x_i - \widetilde{b}) \cdot (-1) = 0 \end{cases}$$

整理可得关于 $\widetilde{a}, \widetilde{b}$ 的二元一次方程组

$$\begin{cases} (\sum_{i=1}^{m} x_i^2) \cdot \widetilde{a} + (\sum_{i=1}^{m} x_i) \cdot \widetilde{b} &= \sum_{i=1}^{m} x_i y_i \\ (\sum_{i=1}^{m} x_i) \cdot \widetilde{a} + \quad m \cdot \widetilde{b} &= \sum_{i=1}^{m} y_i \end{cases} \tag{1.5}$$

利用消元法 (或直接用 Crammer 法则) 容易求得, 上述方程组的解为

$$\begin{cases} \widetilde{a} = \dfrac{m(\sum\limits_{i=1}^{m} x_i y_i) - (\sum\limits_{i=1}^{m} x_i)\cdot(\sum\limits_{i=1}^{m} y_i)}{m(\sum\limits_{i=1}^{m} x_i^2) - (\sum\limits_{i=1}^{m} x_i)^2} \\ \widetilde{b} = \dfrac{(\sum\limits_{i=1}^{m} x_i^2)\cdot(\sum\limits_{i=1}^{m} y_i) - (\sum\limits_{i=1}^{m} x_i)\cdot(\sum\limits_{i=1}^{m} x_i y_i)}{m(\sum\limits_{i=1}^{m} x_i^2) - (\sum\limits_{i=1}^{m} x_i)^2} \end{cases}$$

注意, $\widetilde{a}, \widetilde{b}$ 表达式中的分母 $m(\sum\limits_{i=1}^{m} x_i^2) - (\sum\limits_{i=1}^{m} x_i)^2$ (即线性代数方程组 (1.5) 的系数矩阵的行列式) 是否为零呢? 容易证明: 当且仅当

$$x_1 = x_2 = \cdots = x_m \tag{1.6}$$

时, 分母才等于零. 对于我们的例子, 条件 (1.6) 显然不成立.

至此, 我们已经算出了使得 (1.4) 成立的参数 $\widetilde{a}, \widetilde{b}$, 然而, (1.4) 只是 $(\widetilde{a}, \widetilde{b})$ 为函数 $\delta(a, b)$ 的极小值点的必要条件. 接下来, 还需进一步验证以下两个条件:

1. $\delta_{aa}(\widetilde{a}, \widetilde{b}) > 0$;

2. $\delta_{aa}(\widetilde{a}, \widetilde{b}) \cdot \delta_{bb}(\widetilde{a}, \widetilde{b}) - \delta_{ab}^2(\widetilde{a}, \widetilde{b}) > 0$,

其中 δ_{aa}, δ_{bb} 分别表示 δ 对 a, b 的二阶偏导数, δ_{ab} 表示混合偏导数.

上述两个条件的验证是 straightforward 的, 这里不再累述, 请读者自行推导.

最后, 还有一个问题: 上述方法得到 $(\widetilde{a}, \widetilde{b})$ 是 $\delta(a, b)$ 的**极小值点**, 而我们的目标是求 $\delta(a, b)$ 的**最小值点**. 那么, $(\widetilde{a}, \widetilde{b})$ 是否也是 $\delta(a, b)$ 的最小值点呢? 答案是肯定的, 因为函数 $\delta(a, b)$ 是一个**凸函数**, 而凸函数有很多好性质, 例如任何极值点都是最小值点.

注 1.3 *到这里, 读者可以继续思考之前提到的那个问题, 在整体误差 δ 表达式 (1.1) 中, 如果采用了 $|\delta_i|$ 而不是 δ_i^2, 我们能否用上述方法来求解最小值问题? 如果能, 推导过程变简单了还是变复杂了? 如果不能, 为什么?*

接下来, 我们再看一个多元线性模型的例子.

例 1.2 (**多元线性模型**) *已知样本数据 $(\mathbf{x}_i, y_i)$, $i = 1, 2, \cdots, m$, 其中 $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,n})^T$. 假设 $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ 与 y 的关系可用函数*

$$y = h(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n \tag{1.7}$$

来进行建模, 请确定参数 $\theta_0, \theta_1, \cdots, \theta_n$, 使得函数 (1.7) 可拟合所有的样本数据.

有了关于一元线性模型的经验, 对于多元线性模型的参数求解, 我们也可以依葫芦画瓢地进行. 首先, 定义误差

$$\delta_i = y_i - h(\mathbf{x}_i) \tag{1.8}$$

和整体误差

$$\delta = \sum_{i=1}^{m} \delta_i^2. \tag{1.9}$$

接着, 令

$$\frac{\partial \delta}{\partial \theta_0} = 0, \ \frac{\partial \delta}{\partial \theta_1} = 0, \ \cdots, \ \frac{\partial \delta}{\partial \theta_n} = 0 \tag{1.10}$$

通过联立 (1.10) 展开后得到的 $n+1$ 个式子, 可得到一个关于 $\theta_0, \theta_1, \cdots, \theta_n$ 的 $n+1$ 阶线性代数方程组. 求解这个方程组, 即可解得所需的参数. 但接下来, 我们想用**矩阵 - 向量**表示的方法来求解这个问题.

令矩阵

$$A = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ & & \cdots & & \\ 1 & x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{pmatrix}_{m\times(n+1)}, \tag{1.11}$$

向量

$$\theta = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \cdots \\ \theta_n \end{pmatrix}_{(n+1)\times 1}, \quad \beta = \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}_{m\times 1}, \tag{1.12}$$

并记

$$\varepsilon = \beta - A\theta, \tag{1.13}$$

则有

$$\delta = \varepsilon^T \varepsilon, \tag{1.14}$$

其中上标 T 表示转置.

将 (1.13) 代入 (1.14), 有

$$\begin{aligned} \delta &= (\beta - A\theta)^T(\beta - A\theta) \\ &= \beta^T\beta - \beta^T A\theta - \theta^T A^T\beta + \theta^T A^T A\theta \\ &= \beta^T\beta - 2\theta^T A^T\beta + \theta^T A^T A\theta \quad (\text{利用了 } \beta^T A\theta = \theta^T A^T\beta) \end{aligned}$$

记 $\frac{\partial \delta}{\partial \theta} = (\frac{\partial \delta}{\partial \theta_0}, \frac{\partial \delta}{\partial \theta_2}, \cdots, \frac{\partial \delta}{\partial \theta_n})^T$, 则利用上式容易验算

$$\frac{\partial \delta}{\partial \theta} = -2A^T\beta + 2A^T A\theta,$$

进一步, 由 δ 达到极小值的必要条件 $\frac{\partial \delta}{\partial \theta} = 0$, 可得

$$A^T A\theta = A^T \beta, \tag{1.15}$$

由该方程组解得的 θ 即为我们所需的参数. 可以证明线性代数方程组 (1.15) 一定有解, 但解不一定唯一, 当 $det(A^T A) = 0$ 时有多解.

当然, 由于 (1.10) 只是一个必要条件, 因此由 (1.15) 解得的 θ 是否确实为极小点还需进一步判断, 其中涉及到二阶偏导数对应的 Hesse 矩阵, 这里不再展开讨论.

注 1.4 *值得一提的是, 一元线性模型是多元线性模型的一个特例, 因此, 上述通过矩阵 - 向量表示化归到求解方程组 (1.15) 的方法同样可用于一元线性模型的参数求解.*

§2 逻辑回归

生活中我们经常会碰到**二分类问题**, 例如, 某封邮件是否为垃圾邮件, 某个人是否为潜在客户, 某次在线交易是否存在欺诈行为, 等等. 设 $\{(\mathbf{x}_i, y_i),\ i = 1, 2, \cdots, m\}$ 为一个二分类问题的样本数据, 则有 $y_i \in \{0, 1\}$, 当 $y_i = 1$ 时称相应的样本为**正例**, 当 $y_i = 0$ 时称相应的样本为**负例**.

利用上节介绍的 n 元线性回归方法, 我们可以求得 hypothesis 函数

$$h(\mathbf{x}) \triangleq h_\theta(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n, \tag{2.16}$$

那么这个函数是否可以用来进行二分类呢?

给定一个样本 $\mathbf{x}$, $h_\theta(\mathbf{x})$ 的值是任意的, 它不一定等于 0 或 1. 此时我们可以设定一个阀值 T, 并约定: 当 $h_\theta(\mathbf{x}) \geq T$ 时 $\mathbf{x}$ 为正例, 否则为负例.

然而, 由于 $h_\theta(\mathbf{x})$ 的值是任意的, 阀值 T 的合理选取就成了一件很困难的事情. 此时, 我们自然地想到了**归一化**: 如果能通过某个函数将 $\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$ 的值归一到区间 $[0, 1]$ 上, 那么, 阀值 T 取区间中点 0.5 不就合理了吗?

归一化函数有很多, 本文我们只介绍所谓的**逻辑函数**.

§2.1 逻辑函数

逻辑函数 (logistic function) 也叫 **sigmoid 函数**, 其定义为

$$f(z) = \frac{1}{1 + e^{-z}},\ z \in R, \tag{2.17}$$

图 1 给出了其图像.

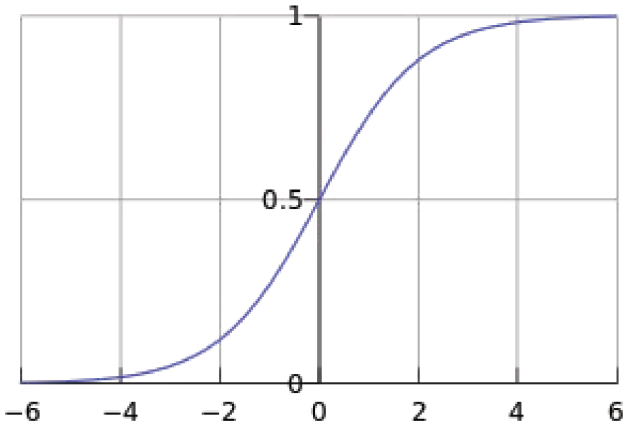

图 1 sigmoid 函数的图像

从图 1 可见: 逻辑函数是一个典型的 “S” 型函数, 它关于 $(0, 0.5)$ 具有对称性. 事实上, 逻辑函数还有很多好性质, 这里不做展开. 此外, 逻辑函数的导数满足

$$f^{'}(z) = f(z)(1 - f(z)), \tag{2.18}$$

这个公式将用于梯度下降法中梯度的推导.

利用逻辑函数, 我们可以将二分类问题的 hypothesis 函数改造成

$$h_\theta = f(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n), \tag{2.19}$$

这里, 为了符号上简化起见, 我们引入 x_0 将向量 $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ 扩展为 $(x_0, x_1, x_2, \cdots, x_n)^T$, 其中 $x_0 = 1$, 且在不引起混淆的情况下仍将其记为 $\mathbf{x}$. 于是, h_θ 可简写为

$$h_\theta(\mathbf{x}) = f(\theta^T \mathbf{x}) = \frac{1}{1 + e^{-\theta^T \mathbf{x}}}. \tag{2.20}$$

取阀值 $T = 0.5$, 则二分类的判别公式为

$$y(\mathbf{x}) \quad = \quad \begin{cases} 1, & h_\theta(\mathbf{x}) \geq 0.5; \\ 0, & h_\theta(\mathbf{x}) < 0.5. \end{cases} \tag{2.21}$$

“线性回归 + 逻辑函数” 就得到了逻辑回归, 因此, **逻辑回归可视为一个通过逻辑函数做了归一化的线性回归**.

回顾一下, 在线性回归中, 我们将**成本函数**定义为 (1.3), 即

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} cost(h_\theta(\mathbf{x}_i), \mathbf{y_i}), \tag{2.22}$$

其中

$$cost(h_\theta(\mathbf{x}), \mathbf{y}) = \frac{1}{2} \left(h_\theta(\mathbf{x}) - \mathbf{y}\right)^2. \tag{2.23}$$

上一节已说明由 (2.22) 定义的 $J(\theta)$ 是**凸函数**. 但对于逻辑回归而言, (2.19) 定义的 $h_\theta(\mathbf{x})$ 是个**非凸函数**, 将其代入 (2.22), 其对应的 $J(\theta)$ 也是个**非凸函数**. 因此, 需要其他形式的成本函数来保证逻辑回归的成本函数是凸函数. 这里, 我们选择**对数似然损失函数**作为逻辑回归的成本函数, 其定义为

$$cost(h_\theta(\mathbf{x}), \mathbf{y}) = \begin{cases} -log\left(h_\theta(\mathbf{x})\right), & y = 1; \\ -log\left(1 - h_\theta(\mathbf{x})\right), & y = 0. \end{cases} \tag{2.24}$$

下面直观地来解释 (2.24). 这里以 $y = 1$ 的情形为例 (图 2, [3]).

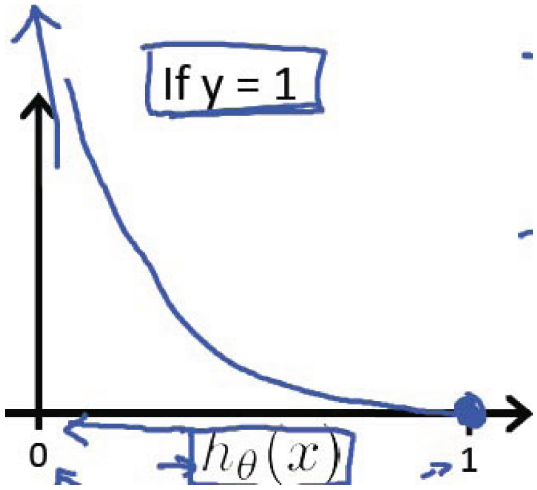

图 2 $y = 1$ 的情形

由图 2 可见: 当 $h_\theta(\mathbf{x})$ 越接近于 1 时, $cost(h_\theta(\mathbf{x}), 1)$ 越小 (特别地, 当 $h_\theta(\mathbf{x}) = 1$ 时有 $cost(h_\theta(\mathbf{x}), 1) = 0$); 当 $h_\theta(\mathbf{x})$ 越接近于 0 时, $cost(h_\theta(\mathbf{x}), 1)$ 越大 (趋于 $+\infty$). 因此, 当预测的值和真实的值相差越大时, $cost(h_\theta(\mathbf{x}), 1)$ 的值越大, 即对于这个学习算法给予一个很大的 $cost$ 作为惩罚. 类似也可分析 $y = 0$ 的情形 (图 3, [3]).

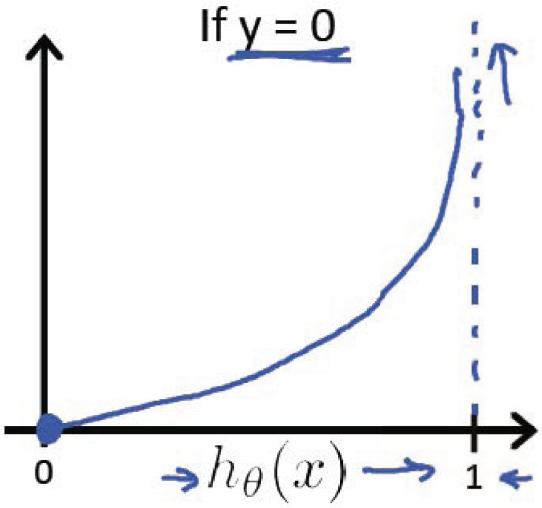

图 3 $y = 0$ 的情形

注意, 代价函数 (2.24) 是一个分段函数, 为下面推导其梯度公式时简单起见, 我们将其进行简化, 写成如下**整体表达式**

$$cost(h_\theta(\mathbf{x}), \mathbf{y}) = -y \cdot log\left(h_\theta(\mathbf{x})\right) - (1-y) \cdot log\left(1 - h_\theta(\mathbf{x})\right), \tag{2.25}$$

容易验证, (2.25) 与 (2.24) 是等价的.

将 (2.25) 代入 (2.22), 可得

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y_i \cdot log\left(h_\theta(\mathbf{x}_i)\right) + (1-y_i) \cdot log\left(1 - h_\theta(\mathbf{x}_i)\right)\right], \tag{2.26}$$

接下来, 我们需求解参数 θ^*, 使得

$$J(\theta^*) = \min_\theta J(\theta). \tag{2.27}$$

§2.2 梯度下降法

求解极小问题 (2.27) 常用的方法是**梯度下降法**, 其迭代公式为

$$\theta_j := \theta_j - \alpha\frac{\partial J(\theta)}{\partial \theta_j}, \quad j = 0, 1, \cdots, n, \tag{2.28}$$

其中 $\alpha > 0$ 为**学习率**, 偏导数

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(\mathbf{x}_i) - y_i)x_{i,j}. \tag{2.29}$$

下面给出偏导数公式 (2.29) 的详细推导过程.

首先, 利用 $J(\theta)$ 的定义式 (2.26), 有

$$\frac{\partial J(\theta)}{\partial \theta_j} = -\frac{1}{m}\sum_{i=1}^{m}\left[y_i \cdot \frac{\partial log\left(h_\theta(\mathbf{x}_i)\right)}{\partial \theta_j} + (1-y_i) \cdot \frac{\partial log\left(1 - h_\theta(\mathbf{x}_i)\right)}{\partial \theta_j}\right], \tag{2.30}$$

接下来分别计算 (2.30) 中的 $\frac{\partial log(h_\theta(\mathbf{x}_i))}{\partial \theta_j}$ 和 $\frac{\partial log(1-h_\theta(\mathbf{x}_i))}{\partial \theta_j}$.

注意到 $h_\theta(\mathbf{x}_i) = f(\theta^T\mathbf{x}_i)$, 以及 $f'(z) = f(z) * (1 - f(z))$ (即 (2.18) 式), 并利用求导链式法则, 可得

$$\begin{aligned}
& \frac{\partial log\left(h_\theta(\mathbf{x}_i)\right)}{\partial \theta_j} \\
= \ & \frac{\partial log\left(g(\theta^T\mathbf{x}_i)\right)}{\partial \theta_j} \\
= \ & \frac{1}{f(\theta^T\mathbf{x}_i)} \cdot \frac{\partial f(\theta^T\mathbf{x}_i)}{\partial \theta_j} \\
= \ & \frac{1}{f(\theta^T\mathbf{x}_i)} \cdot f(\theta^T\mathbf{x}_i)(1 - f(\theta^T\mathbf{x}_i)) \cdot \frac{\partial(\theta^T\mathbf{x}_i)}{\partial \theta_j} \\
= \ & (1 - f(\theta^T\mathbf{x}_i)) \cdot x_{i,j} \\
= \ & (1 - h_\theta(\mathbf{x}_i)) \cdot x_{i,j}
\end{aligned}$$

类似地, 有

$$\begin{aligned}
& \frac{\partial log\,(1-h_\theta(\mathbf{x}_i))}{\partial\theta_j} \\
= & \frac{\partial log\,\big(1-g(\theta^T\mathbf{x}_i)\big)}{\partial\theta_j} \\
= & -\frac{1}{1-f(\theta^T\mathbf{x}_i)}\cdot f(\theta^T\mathbf{x}_i)(1-f(\theta^T\mathbf{x}_i))\cdot\frac{\partial(\theta^T\mathbf{x}_i)}{\partial\theta_j} \\
= & -f(\theta^T\mathbf{x}_i)\cdot x_{i,j} \\
= & -h_\theta(\mathbf{x}_i)\cdot x_{i,j}
\end{aligned}$$

将它们代入 (2.30), 并化简即可得到 (2.29).

注 2.1 虽然上一节中求解线性回归模型的最小值问题时, 采用的是最小二乘法, 事实上, 我们也可以采用梯度下降法.

最后, 我们给出一个完整的**逻辑回归**的训练算法.

§2.3 训练算法

算法 2.1 (**Logistic Regression**)

Step 1 初始化 (Initialization)

(1) 给定参数 K (迭代次数) 和 α (学习率).

(2) 零初始化参数: $\theta := (0, 0, \cdots, 0)^T$.

Step 2 训练 (Training)

$FOR\ \ k = 1, 2, \cdots, K$ // 迭代 K 次

{

  $FOR\ \ i = 1, 2, \cdots, m$ // 对所有样本数据循环

  {

    $FOR\ \ j = 1, 2, \cdots, n$ // 对 θ 的所有分量作一次刷新

    {

      $\theta_j := \theta_j - \alpha\cdot\frac{1}{m}\sum_{l=1}^{m}(h_\theta(\mathbf{x}_l)-y_l)x_{l,j}$

    }

  }

}

训练得到了参数 θ 后, 便可以利用公式 (2.21) 来进行预测了.

注 2.2 算法 *2.1* 中, 对参数 θ 进行刷新时, 用到了所有样本数据的误差, 这种方法叫做**标准梯度下降**. 还有一种叫做**随机梯度下降 (stochastic gradient descent, SGD)** 或**增量梯度下降**, 根据某个单独样本数据的误差来对参数进行刷新, 此时算法 *2.1* 中的迭代公式可以改为

$$\theta_j := \theta_j - \alpha \cdot (h_\theta(\mathbf{x}_i) - y_i)x_{i,j}.$$

与随机梯度下降法相比, 标准梯度下降法由于使用了真实的梯度, 因此可使用更大的步长 *(*或学习率*)*.

参考文献

[1] http://baike.baidu.com/view/359236.htm#sub9073975

[2] http://hi.baidu.com/hehehehello/item/40025c33d7d9b7b9633aff87

[3] http://52opencourse.com/125/coursera 公开课笔记 - 斯坦福大学机器学习第六课 - 逻辑回归 -logistic-regression