LN9. 线性回归

李钦宾

先进智能计算与系统团队

邮箱：*qinbin@hust.edu.cn*

2025 年 3 月

Table of contents

目录

线性回归的形式化定义

形式化定义:

在统计学中，线性回归是一种对标量响应 (或因变量) 与一个或多个解释变量 (或自变量) 之间关系建模的线性方法。
只有一个解释变量 (自变量) 的情况称为简单线性回归。
对于多于一个的解释变量，这个过程称为多重线性回归（multiple linear regression）。
这个术语与多元线性回归（multivariate linear regression）不同，多元线性回归预测的是多个相关因变量，而不是单个标量变量。

- 数据假设: $y_i \in \mathbb{R}$
- 模型假设: $y_i = \mathbf{w}^\top \mathbf{x}_i + \epsilon_i$ 其中 $\epsilon_i \sim N(0, \sigma^2)$

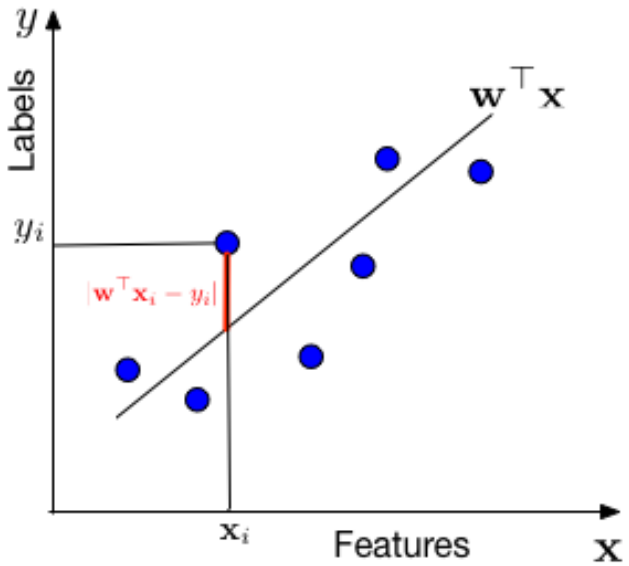
  $\Rightarrow y_i|\mathbf{x}_i \sim N(\mathbf{w}^\top \mathbf{x}_i, \sigma^2) \Rightarrow P(y_i|\mathbf{x}_i, \mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\mathbf{x}_i^\top \mathbf{w} - y_i)^2}{2\sigma^2}}$

线性回归的形式化定义

形式化定义：

换句话说，我们假设数据是从原点的“线” $\mathbf{w}^\top \mathbf{x}$ 获得的 (人们总是可以通过额外的维度添加偏差/偏移量，类似于感知机)。
对于每个具有 $\mathbf{x}_i$ 特征的数据点，标签 y 是从均值为 $\mathbf{w}^\top \mathbf{x}_i$ 和方差为 σ^2 的高斯分布中获得的。

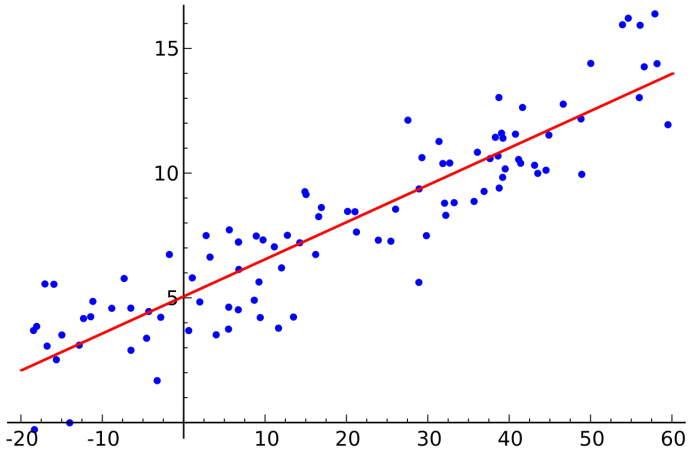
我们的任务是从数据中估计斜率 $\mathbf{w}$。



目录

简单和多重线性回归

单一标量自变量 x 和单一标量因变量 y 的最简单情况被称为简单线性回归。
扩展到多个和/或向量值预测变量（用大写 X 表示）被称为多重线性回归，也称为多变量线性回归。
几乎所有现实世界的回归模型都涉及多个预测因子，线性回归的基本描述通常是用多重回归模型来描述的。



广义线性模型

广义线性模型 (Generalized linear models，GLMs) 是一种为有界或离散的因变量建模的框架。例如:

- 当对变化幅度较大的正值变量 (例如价格或人口) 建模时，这些正值量能被偏态分布 (skewed distribution, 频数分布的高峰位于一侧，尾部向另一侧延伸的分布) 描述，如对数正态分布或泊松分布 (尽管 GLMs 不用于对数正态数据，但是因变量利用对数函数进行了简单的转换);
- 对分类（可枚举）数据建模时，例如在选举中对给定候选人的选择 (对于二元选择使用伯努利分布/二项分布可更好地描述，对于多元选择使用分类分布/多项分布)，其中这些数量固定的选择不能被有意义地排序;
- 当对有序数据建模时，例如从 0 到 5 的评分，不同的结果可以排序，但数量本身可能没有任何绝对意义 (例如评分 4 可能不是评分 2 在客观意义上的 “两倍好”，而只是表明它比 2 或 3 好，但不如 5)。

层次线性模型

层次线性模型 (或多层回归) 将数据组织成一个回归的层次结构（Hierarchy），例如 A 在 B 上回归，B 在 C 上回归。

它通常用于变量具有自然的层次结构，例如在教育统计中，学生嵌套在教室中，教室嵌套在学校中，学校嵌套在一些学区中。

因变量可以是学生成绩的衡量标准，如考试分数，不同的协变量将在教室、学校和学区级别收集。

目录

用 MLE 估计

$$
\begin{aligned}
\mathbf{w} &= \underset{\mathbf{w}}{\operatorname{argmax}} P(Y|X, \mathbf{w}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \prod_{i=1}^{n} P(y_i|\mathbf{x}_i, \mathbf{w}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \sum_{i=1}^{n} \log\left[P(y_i|\mathbf{x}_i, \mathbf{w})\right] \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \sum_{i=1}^{n} \left[\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \log\left(e^{-\frac{(\mathbf{x}_i^\top \mathbf{w} - y_i)^2}{2\sigma^2}}\right)\right]
\end{aligned}
$$

用 MLE 估计

$$= \operatorname*{argmax}_{\mathbf{w}} -\frac{1}{2\sigma^2} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$$

$$= \operatorname*{argmin}_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$$

- 我们最小化损失函数, $\ell(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$。这个特殊的损失函数也被称为均方损失或最小二乘损失 (Ordinary Least Squares)。最小二乘损失可以用梯度下降法、牛顿法或闭式解进行优化。

  闭式解形式 $\mathbf{w} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{y}$，其中 $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ 且 $\mathbf{y} = [y_1, \ldots, y_n]^\top$.

用 MLE 估计

$$\ell(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i^\top \mathbf{w} - y_i)^2$$

$$= (X^T w - y)^T (X^T w - y)$$

$$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n], \mathbf{y} = [y_1, \ldots, y_n]^\top$$

$$\Rightarrow \hat{\mathbf{w}} = (\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{y} \in \mathbb{R}^{d\times 1}, \quad \mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}, \mathbf{y} = [y_1, \ldots, y_n]^T \in \mathbb{R}^{n\times 1}.$$

可利用公式:

$$\frac{\partial Ax}{\partial x} = A^T$$

$$\frac{\partial x^T A}{\partial x} = A$$

$$\frac{\partial A^T x B}{\partial x} = AB^T$$

$$\frac{\partial A^T x^T B}{\partial x} = BA^T$$

目录

用 MAP 估计

附加模型假设: $P(\mathbf{w}) = \frac{1}{\sqrt{2\pi\tau^2}} e^{-\frac{\mathbf{w}^\top \mathbf{w}}{2\tau^2}}$

$$
\begin{aligned}
\mathbf{w} &= \underset{\mathbf{w}}{\operatorname{argmax}}\, P(\mathbf{w}|y_1, \mathbf{x}_1, \ldots, y_n, \mathbf{x}_n) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}}\, \frac{P(y_1, \mathbf{x}_1, \ldots, y_n, \mathbf{x}_n|\mathbf{w})P(\mathbf{w})}{P(y_1, \mathbf{x}_1, \ldots, y_n, \mathbf{x}_n)} \\
&= \underset{\mathbf{w}}{\operatorname{argmax}}\, P(y_1, \mathbf{x}_1, \ldots, y_n, \mathbf{x}_n|\mathbf{w})P(\mathbf{w}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \left[\prod_{i=1}^{n} P(y_i, \mathbf{x}_i|\mathbf{w})\right] P(\mathbf{w}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \left[\prod_{i=1}^{n} P(y_i|\mathbf{x}_i, \mathbf{w})P(\mathbf{x}_i|\mathbf{w})\right] P(\mathbf{w}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \left[\prod_{i=1}^{n} P(y_i|\mathbf{x}_i, \mathbf{w})P(\mathbf{x}_i)\right] P(\mathbf{w})
\end{aligned}
$$

$$= \underset{\mathbf{w}}{\operatorname{argmax}} \left[\prod_{i=1}^{n} P(y_i|\mathbf{x}_i, \mathbf{w}) \right] P(\mathbf{w})$$

$$= \underset{\mathbf{w}}{\operatorname{argmax}} \sum_{i=1}^{n} \log P(y_i|\mathbf{x}_i, \mathbf{w}) + \log P(\mathbf{w})$$

$$= \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{2\sigma^2} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2 + \frac{1}{2\tau^2} \mathbf{w}^\top \mathbf{w}$$

$$= \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{n}$$

$$\sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2 + \lambda ||\mathbf{w}||_2^2 \qquad \lambda = \frac{\sigma^2}{n\tau^2}$$

- **这个目标被称为岭回归。它有一个闭式解:** $\mathbf{w} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{X}\mathbf{y}$, **其中** $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ **且** $\mathbf{y} = [y_1, \ldots, y_n]^\top$.

金融

资本资产定价模型使用线性回归和 beta 的概念来分析和量化投资的系统风险。
这直接来自线性回归模型的 beta 系数，该模型将投资回报与所有风险资产的回报联系起来。

经济学

线性回归是经济学中主要的实证工具。可被用来预测消费支出、固定投资支出、库存投资、购买一个国家的出口产品、进口支出、持有流动资产的需求、劳动力需求和劳动力供给。

环境科学

线性回归在环境科学中有着广泛的应用。在加拿大，环境影响监测方案使用对鱼类和底栖生物调查的统计分析来衡量纸浆厂或金属矿山废水对水生生态系统的影响。

机器学习

线性回归在机器学习等人工智能领域发挥着重要作用。线性回归算法因其相对简单和广为人知的特性而成为监督机器学习的基本算法之一。

总结

- **最小二乘法**:
  $\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$
  均方损失
  没有正则项
  闭式解形式: $\mathbf{w} = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{y}$

- **岭回归**:
  $\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2 + \lambda ||\mathbf{w}||_2^2$
  均方损失
  ℓ_2-regularization
  闭式解形式: $\mathbf{w} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{X}\mathbf{y}$

简单的线性回归实现

Simple linear regression implementation code click here
https://github.com/IEC-lab/MachineLearning2019/blob/master/Linear%20Regression/linear_regression.py

房价预测

housing price prediction code click here
https://github.com/IEC-lab/MachineLearning2019/blob/master/Linear Regression/houseprice.py

The End