机器学习（L03）：线性回归

授课教师：王贝伦 / 助教：张嘉琦，周嘉莹，黄旭

1 有监督学习 (Supervised Learning)

1.1 定义

定义：有监督学习

从给定的训练数据集中学习出一个函数（function）（模型参数），当新的数据到来时，可以根据这个函数预测结果。

1.2 理解

在监督学习中，每个样本（sample）都是由一组输入特征（feature）（通常为矢量）和一个期望的输出值（label）（也称为标签）组成。训练集中的标签是由人标注的。通过已有的训练样本（即已知数据及其对应的输出）去训练得到一个最优模型（这个模型属于某个函数的集合，**最优**表示某个评价准则下是最佳的），再利用这个模型将所有的输入映射为相应的输出，对输出进行简单的判断从而输出所需要结果，也就具有了对未知数据预测的能力。

监督学习的目标往往是让计算机去学习我们已经创建好的分类系统（模型）的参数。

监督学习算法是分析训练集数据，并生成一个模型或函数，可以用于预测新的样本。

1.3 特点

有**数据特征** (x) 以及**标签** (y)。数据特征可以是单个（如直线上点的 x 坐标），也可以是多个（如一个人的身高、体重、肺活量）（通常用矢量表示）；每个数据对应的标签可以是 1 个（如男或女、猫或狗、音乐发行的具体年份），也可以是多个（如某某首歌既属于摇滚音乐，也属于华语音乐）。

有监督学习的目标是学习到一个可以将数据映射到标签的函数或模型（参数）。

1.4 有监督学习的分类

1.4.1 回归问题 (Regression)

回归问题，就是根据数据集拟合一条直线 (平面）或曲线（曲面），使得损失函数 L 最小。回归问题的目标是：对每一个输入的样本 $\mathbf{x}$，都能得到一个预测值 y，使得 y 接近样本的实际值。在回归问题中 y 是实数，为连续值。

例 1. 根据楼房的房屋面积、卧室数量、地段等特征预测房价。

每个楼房代表一个样本 (sample)，输入值 (input) 为样本特征 (feature)，例如大小、所处地段，输出值 (output) 为房屋价格，是一个实数。

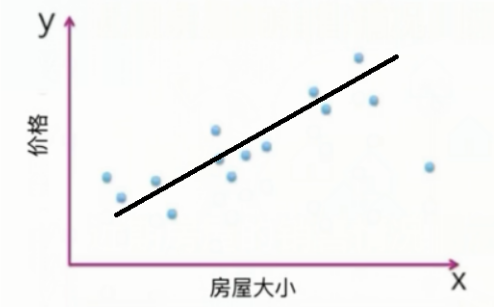

图 1：房屋大小与价格之间的关系

1.4.2 分类问题 (Classification)

分类问题，根据数据集的特征训练出一个或多个决策边界，使得具有相似特征的数据被划分到同一个集合内。

分类问题的目标是：根据样本的属性值，预测样本属于哪个或哪几个类别。分类问题与回归问题的不同点在于 y 值属于一个有限集合 r，可以看做类标号。

例 2. 使用合适的决策边界（直线）将坐标平面划分为两个区域，使得坐标平面上的点尽可能按照颜色分类到同一个区域。

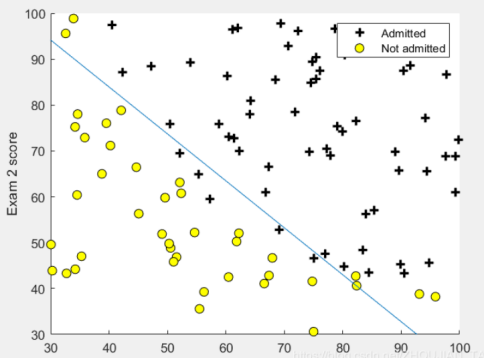

图 2：平面上不同颜色的点以及决策边界

每个点代表一个样本，输入值为点的坐标 (x_i, x_k)，输出值为点的颜色 $y \in \{0, 1\}$，其中 $y = 1$ 代表黑色，$y = 0$ 代表黄色。

当出现一个未知颜色的点，则可根据点在坐标系中的位置预测其颜色为黄色还是黑色。

1.4.3 实例

- 图片分类中输入一张图片，输出图片的分类。
- 目标检测中输入一张图片，输出包裹目标物体的边框。
- 自然语言处理中，输入一段文本，判断褒贬色彩。
- 估计肿瘤性质中，输入发现的肿瘤的特征，判断肿瘤是良性或恶性。
- 歌曲年代预测中，输入一首歌，判断歌曲发行的年代。

2 回归问题（Regression）

2.1 定义

回归，在统计学上指研究一组随机变量 $(y_1, y_2, \cdots, y_i)$ 和另一组 $(x_1, x_2, \cdots, x_k)$ 变量之间关系的统计分析方法，通常 $y_1, y_2, \cdots, y_i$ 是因变量，$x_1, x_2, \cdots, x_k$ 是自变量。

定义：回归

回归问题的学习等价于函数拟合：选择一条函数曲线使其很好地拟合已知数据且很好地预测未知数据。

在机器学习中，回归问题就是根据数据集拟合样本特征和标签之间的一条直线或曲线，使得损失函数 L 最小，从而具备预测能力。

2.2 特点

回归问题的输出是**连续型变量**。

2.3 回归问题分类

2.3.1 线性回归 (Linear Regression)

顾名思义，变量之间成线性关系，一般可通过作图直观观测出来。线性回归使用线性模型，形如 $y = \mathbf{w}^\top \mathbf{x}$。回归拟合函数为直线或平面。

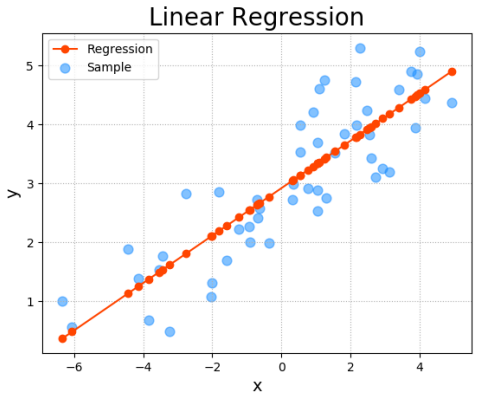

图 3：当样本特征空间 $\chi = \mathbb{R}$，回归函数为一条直线

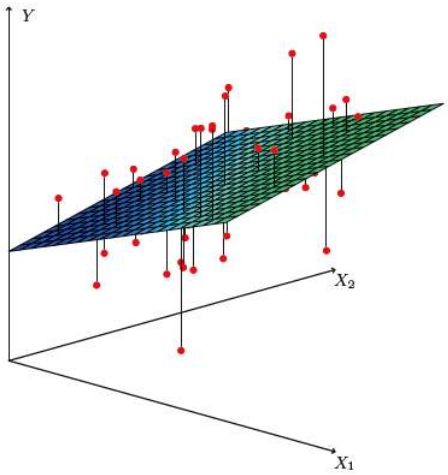

图 4：当样本特征空间 $\chi = \mathbb{R}^2$，回归函数为一个平面

2.3.2 非线性回归（Non-linear Regression）

很多场合线性模型无法很好的拟合目标数据曲线，这就需要引入非线性回归模式。非线性回归中变量之间存在非线性关系。回归拟合函数为曲线或曲面。

但是，非线性回归问题一般比较复杂，许多非线性回归算法将非线性回归转化为线性回归，再按照线性回归求解。

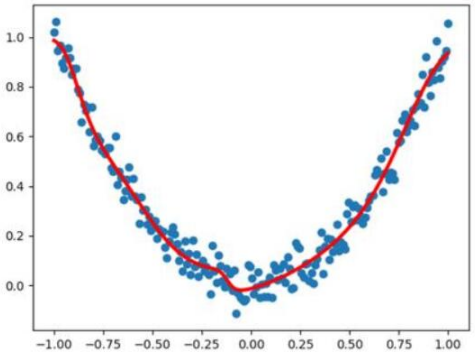

图 5：当样本特征空间 $\chi = \mathbb{R}$，回归函数为一条曲线

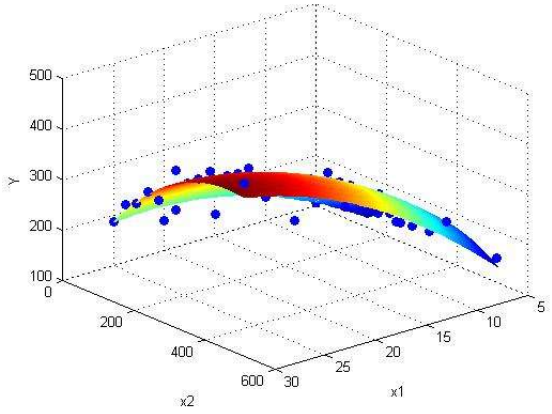

图 6：当样本特征空间 $\chi = \mathbb{R}^2$，回归函数为一个曲面

2.3.3 回归问题实例

- 根据某地区的气温、幼虫数量预测夏季蝗虫数量。
- 根据新冠肺炎各省确诊人数预测武汉实际感染人数。

3 线性回归（Linear Regression）

3.1 定义

定义：线性回归

线性回归是回归问题中的一种，线性回归假设样本标签与特征之间**线性相关**，即满足一个多元一次方程。通过构建损失函数，来求解损失函数最小时的参数 $\mathbf{w}$。

通常线性回归模型函数如下所示：

$$\hat{y} = \mathbf{w}^\top \mathbf{x}$$

其中 $\hat{y}$ 为预测值，自变量 $\mathbf{x}$ 和因变量 y 是已知的，而我们想实现的是预测一个不在数据集中的 $\mathbf{x}$，其对应的 y 是多少。因此，需要通过已知数据集，求解线性模型中参数 $\mathbf{w}$。

3.2 线性回归模型特点

- 建模速度快，不需要很复杂的计算，在数据量大的情况下依然运行速度很快。
- 可以根据系数给出每个变量的理解和解释。
- 对异常值很敏感。

  例如在图 7.b 的数据集上建立回归，因最右边噪点的存在，使回归模型在训练集上表现都很差。

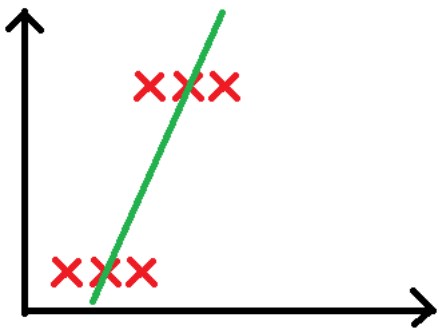

图 7.a 无噪点的数据集使用线性回归

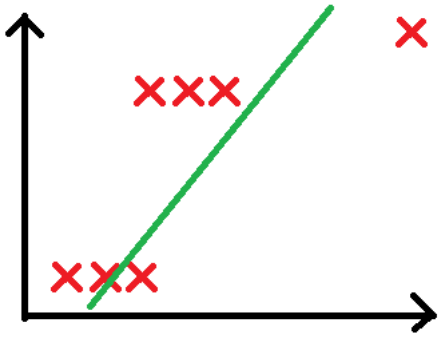

图 7.b 有噪点的数据集使用线性回归

3.3 数学表示

3.3.1 样本特征集

对单一数据样本，若样本特征数为 p，则每个样本的特征表示为 p 维列向量，每一行对应一个特征。

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}$$

对 n 个数据样本，若每个样本特征数为 p，样本特征集表示为 $n \times p$ 维矩阵，每一行包含一个不同的样本，每一列对应于不同的特征。

$$\mathbf{X} = \begin{bmatrix} -- & x_1^\top & -- \\ -- & x_2^\top & -- \\ \vdots & \vdots & \vdots \\ -- & x_n^\top & -- \end{bmatrix} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & \vdots & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{bmatrix}$$

3.3.2 样本标签集

对单一样本，标签为 y

对 n 个数据样本，标签集表示为 n 维列向量，每一行对应一个样本的标签。

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

3.3.3 线性回归函数

回归函数 $\hat{y} = f(\mathbf{x})$ 表示为 $\mathbf{x}$ 的线性函数，其中 $\hat{y}$ 为预测值，$\mathbf{w}$ 为参数集合集合，为 p 维列向量（对应 p 个样本特征）。

$$\begin{aligned} \hat{y} = f(\mathbf{x}) &= w_1 x_1 + w_2 x_2 + ... w_p x_p \\ &= \mathbf{x}^\top \mathbf{w} \\ &= \mathbf{w}^\top \mathbf{x} \end{aligned}$$

其中

$$\mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_p \end{bmatrix}$$

3.3.4 损失函数：和方差（The sum of squares due to error，以下简写为 SSE）

线性回归模型最终是可以得到一组预测值 $\hat{y}$ ，对比已有的真实值 y，可以将损失函数定义如下：

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

即预测值与真实值之间的平方距离和，称为和方差（The sum of squares due to error）。

现任务是求解最小化 $J(\mathbf{w})$ 时 $\mathbf{w}$ 的值，即核心目标优化式为：

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{2}\sum_{i=1}^{n}(f(\mathbf{x}_i) - y_i)^2$$

$1/2$ 的系数是为了便于求导，对求解结果无影响。

3.4 线性回归实例

3.4.1 牛肉价格预测

问题 牛肉价格受到牛的养殖数量、饲料价格、人工成本等多方面的影响。现在需要分析牛肉价格和以上变量之间的关系。

数学表示 假设牛肉价格与养殖数量为 x_1、饲料价格为 x_2、人工成本为 x_3，牛肉价格为 y，x_1, x_2, x_3 与 y 之间为线性关系，目标函数如下所示：

$$\begin{aligned}\hat{y} = f(x) &= w_1x_1 + w_2x_2 + w_3x_3 \\ &= \mathbf{x}^\top\mathbf{w}\end{aligned}$$

其中 x_1, x_2, x_3 为三个样本特征，w_1, w_2, w_3 为对应的系数。损失函数为：

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

Table 1: 线性回归总结

步骤	举例
明确问题	回归问题 $\mathbf{y} = \mathbf{X}\mathbf{w}$
数据表示	$\mathbf{X} \in \mathbb{R}^{n\times p}, \mathbf{y} \in \mathbb{R}^n$
损失函数	$(J(\mathbf{w}) = \frac{1}{2}(\mathbf{X}\mathbf{w} - \mathbf{y})^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$
优化方法	正规方程/梯度下降……
模型/参数	$\mathbf{w}$

4 线性回归解法——正规方程（Normal Equation）

为求解 $J(\mathbf{w})$ 的最小值，可以先求出 $J(\mathbf{w})$ 的极值，然后判断极值点是否为最小值点，若是，则该函数的极小值为其最小值。

由矩阵性质可展开损失函数：

$$\begin{aligned}J(\mathbf{w}) &= \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i^\top\mathbf{w} - \mathbf{y}_i)^2 \\ &= \frac{1}{2}(\mathbf{X}\mathbf{w} - \mathbf{y})^\top(X\mathbf{w} - \mathbf{y}) \\ &= \frac{1}{2}(\mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} - \mathbf{w}^\top\mathbf{X}^\top\mathbf{y} - \mathbf{y}^\top\mathbf{X}\mathbf{w} + \mathbf{y}^\top\mathbf{y})\end{aligned}$$

求 $J(\mathbf{w})$ 的一阶偏导（梯度）：

$$\begin{aligned}\nabla_{\mathbf{w}}J(\mathbf{w}) &= \frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} \\ &= \frac{1}{2}(2\mathbf{X}^\top\mathbf{X}\mathbf{w} - 2\mathbf{X}^\top\mathbf{y})\end{aligned}$$

求 $J(\mathbf{w})$ 的二阶偏导（黑塞矩阵）判断极值点情况：

$$\begin{aligned} H(J(\mathbf{w})) &= \frac{\partial \nabla_{\mathbf{w}} J(\mathbf{w})}{\partial \mathbf{w}} \\ &= \frac{\partial (\mathbf{X}^\top \mathbf{X}\mathbf{w})}{\partial \mathbf{w}} \\ &= \mathbf{X}^\top \mathbf{X} \end{aligned}$$

由 $\mathbf{X}^\top \mathbf{X} \succeq 0$ 知 $J(\mathbf{w})$ 为凸函数，则 $J(\mathbf{w})$ 的局部极小值是全局极小值。

当 $J(\mathbf{w})$ 取极小值时，$\nabla_{\mathbf{w}} J(\mathbf{w}) = 0$，则有：

$$\mathbf{X}^\top \mathbf{X}\mathbf{w} = 2\mathbf{X}^\top \mathbf{y}$$

$$\mathbf{w}^* = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

得到模型参数 $\mathbf{w}^*$。

4.1 黑塞矩阵

4.1.1 定义

定义：黑塞矩阵

黑塞矩阵（Hessian Matrix），是一个多元函数的二阶偏导数构成的方阵，描述了函数的局部曲率。

设 n 元实函数 $f(x_1, x_2, ..., x_n)$ 在点 M_0 的邻域内有二阶连续偏导，则

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 x_n} \\ \frac{\partial^2 f}{\partial x_2 x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n x_1} & \frac{\partial^2 f}{\partial x_n x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

4.1.2 损失函数、梯度与黑塞矩阵

梯度与黑塞矩阵为损失函数的一阶偏导和二阶偏导矩阵

Cost function	Gradient	Hessian
$J(\boldsymbol{\theta})$	$\boldsymbol{g} = \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$	$\boldsymbol{H}$
	$g_i = \frac{\partial}{\partial \theta_i} J(\boldsymbol{\theta})$	$H_{i,j} = \frac{\partial}{\partial \theta_j} g_i$

4.2 正规方程方程性质

$\mathbf{w}^* = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$，当 $\mathbf{X}$ 满秩时：

- 当样本数量 n 大于样本空间的维数 p，并且输入矩阵 $\mathbf{X}$ 满足列满秩，则 $\mathbf{X}^\top \mathbf{X}$ 可逆。
- 若 $\mathbf{X}^\top \mathbf{X}$ 满足可逆，则也满足正定，因此我们找到的临界点 (通过求解梯度为零) 是最小的。
- 如果 $\mathbf{X}$ 为非满秩矩阵，则 $\mathbf{X}$ 不可逆。使用正规方程解法则需要对 $\mathbf{X}$ 进行正则化。

4.3 时间复杂度分析

在计算 $\mathbf{w}^* = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$ 过程中，设 $\mathbf{X}$ 为 $n * p$ 的矩阵

- $\mathbf{X}^\top \mathbf{X}$：矩阵相乘时间复杂度 $O(p^2 n)$。
- $(\mathbf{X}^\top \mathbf{X})^{-1}$：矩阵转置时间复杂度 $O(p^3)$。

总时间复杂度 $O(p^2 n + p^3)$。

当 $n >> p$ 时，矩阵的乘法慢于矩阵的逆运算。

4.4 线性回归的概率论解释

假设线性回归方程为：

$$y^{(i)} = \mathbf{w}^\top \mathbf{x}^{(i)} + \varepsilon^{(i)}$$

ε 可以代表各种误差，比如测量误差，或者因为其他未知的样本特征引起的误差。假设这些误差符合独立高斯分布 $\varepsilon^{(i)} \sim \mathcal{N}(0, \sigma^2)$ 则有：

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

所以：

$$y^{(i)}|\mathbf{x}^{(i)}; \mathbf{w} \sim \mathcal{N}(\mathbf{w}^\top \mathbf{x}^{(i)}, \sigma^2)$$

$$p(y^{(i)}|\mathbf{x}^{(i)}; \mathbf{w}) = \frac{1}{\sqrt{2\pi}} exp(-\frac{(y^{(i)} - \mathbf{w}^\top \mathbf{x}^{(i)})^2}{2\sigma^2}).$$

因为 ε 是独立同分布的，所以每个样本的输出 y 也是独立同分布的。那么就可以用极大似然估计（MLE）来估计 $\mathbf{w}$。似然函数为：

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

似然函数的 ln 形式为：

$$\begin{aligned} \ell(\mathbf{w}) &= logL(\mathbf{w}) \\ &= mlog\frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{\sigma^2} \cdot \frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - \mathbf{w}^\top \mathbf{x}^{(i)})^2. \end{aligned}$$

可以看出，MLE 的最终结果就是要最小化

$$\frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - \mathbf{w}^\top \mathbf{x}^{(i)})^2$$

恰好为损失函数 $J(\mathbf{w})$

引用

[1] https://qiyanjun.github.io/2019f-UVA-CS6316-MachineLearning/