线性回归算法

陈 鑫

2023 年 10 月 17 日

目录

目录

记号：数据集用矩阵 X 表示，每一行是一个样本，用 $\boldsymbol{x}^{(i)}$ 表示，每一列是一个特征（属性）feature，用 $\boldsymbol{x}_j^{(i)}$ 表示第 i 个样本的第 j 个特征。最后一列称为标签，用向量 $\boldsymbol{y}$ 表示，第 i 个样本的标签记为 $y^{(i)}$。假设有直线方程 $y = a\boldsymbol{x} + b$

$$J(a,b) = \sum_{i=1}^{m}(y^{(i)} - ax^{(i)} - b)^2$$

令 $\frac{\partial J(a,b)}{\partial \boldsymbol{a}} = 0$，$\frac{\partial J(a,b)}{\partial b} = 0$ 得到

$$\frac{\partial J(a,b)}{\partial a} = 0 \Rightarrow \sum_{i=1}^{m} 2(y^{(i)} - \boldsymbol{a}\boldsymbol{x}^{(i)} - b)(-\boldsymbol{x}^{(i)}) = 0 \tag{1}$$

$$\frac{\partial J(a,b)}{\partial b} = \sum_{i=1}^{m} 2(y^{(i)} - ax^{(i)} - b)(-1) = 0 \tag{2}$$

化简（2），

$$\sum_{i=1}^{m}(y^{(i)} - ax^{(i)} - b) = 0 \Rightarrow \sum_{i=1}^{m} y^{(i)} - a\sum_{i=1}^{m} x^{(i)} - \sum_{i=1}^{m} b = 0$$

$$\sum_{i=1}^{m} y^{(i)} - a\sum_{i=1}^{m} x^{(i)} - mb = 0 \Rightarrow mb = \sum_{i=1}^{m} y^{(i)} - a\sum_{i=1}^{m} x^{(i)}$$

得到参数 b 为

$$b = \bar{y} - a\bar{x}, \bar{x} = \frac{\sum_{i=1}^{m} x^{(i)}}{m}, \bar{y} = \frac{\sum_{i=1}^{m} y^{(i)}}{m}$$

将 b 代入 (1),

$$\sum_{i=1}^{m}(y^{(i)} - ax^{(i)} - \bar{y} + a\bar{x})x^{(i)} = 0 \Rightarrow$$

$$\sum_{i=1}^{m}(x^{(i)}y^{(i)} - a(x^{(i)})^2 - x^{(i)}\bar{y} + a\bar{x}x^{(i)}) = 0 \Rightarrow$$

$$\sum_{i=1}^{m}(x^{(i)}y^{(i)} - x^{(i)}\bar{y}) - \sum_{i=1}^{m}(a(x^{(i)})^2 - a\bar{x}x^{(i)}) \Rightarrow$$

$$\sum_{i=1}^{m}(x^{(i)}y^{(i)} - x^{(i)}\bar{y}) - a\sum_{i=1}^{m}((x^{(i)})^2 - \bar{x}x^{(i)}) = 0$$

求得 a

$$a = \frac{\sum_{i=1}^{m}(x^{(i)}y^{(i)} - x^{(i)}\bar{y})}{\sum_{i=1}^{m}((x^{(i)})^2 - \bar{x}x^{(i)})}$$

因为：

$$\sum_{i=1}^{m} x^{(i)}\bar{y} = \bar{y}\sum_{i=1}^{m} x^{(i)} = m\bar{y}\bar{x} = \bar{x}\sum_{i=1}^{m} y^{(i)} = \sum_{i=1}^{m} \bar{x}y^{(i)} = \sum_{i=1}^{m} \bar{x}\bar{y}$$

注意到 $\sum_{i=1}^{m} \bar{x}y^{(i)} = \sum_{i=1}^{m} \bar{x}\bar{y}$ 和 $\sum_{i=1}^{m} (\bar{x}x^{(i)} - \bar{x}^2) = 0$，所以

$$a = \frac{\sum_{i=1}^{m} (x^{(i)}y^{(i)} - x^{(i)}\bar{y} - \bar{x}y^{(i)} + \bar{x}\bar{y})}{\sum_{i=1}^{m} ((x^{(i)})^2 - \bar{x}x^{(i)} - \bar{x}x^{(i)} + \bar{x}^2)} = \frac{\sum_{i=1}^{m} (x^{(i)} - \bar{x})(y^{(i)} - \bar{y})}{\sum_{i=1}^{m} (x^{(i)} - \bar{x})^2}$$

例子 1：书中图 3-1 简单线性回归

例子 2：正规方程（最小二乘法）算法

机器学习中将以下 $\mathbb{R}^n \to \mathbb{R}$ 的函数 $h_{\boldsymbol{w},b}(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$ 称为一个线性模型。其中，$\boldsymbol{w}, \boldsymbol{x} \in \mathbb{R}^n$ 均为 n 维向量，$b \in \mathbb{R}$ 为偏置项。

一般情况下，线性回归算法实际上是一个经验损失最小化算法。其模型假设为线性模型，损失函数为平方损失函数。依照平方损失函数的定义，关于数据 $(\boldsymbol{x}, y)$，模型 h 的平方损失为 $(h(\boldsymbol{x}) - y)^2$。

Algorithm 1: 线性回归算法

样本空间 $X \subseteq \mathbb{R}^n$，$y \in \mathbb{R}$

Input: m 个训练数据 $S = \{(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \cdots, (\boldsymbol{x}^{(m)}, y^{(m)})\}, \boldsymbol{x} \in \mathbb{R}^n, y \in \mathbb{R}$

Output: 线性模型 $h_{\boldsymbol{w}^*,b^*}(\boldsymbol{x}) = w^T\boldsymbol{x} + b$，使得 $\boldsymbol{w}^*, b^*$ 为如下优化问题的最优解:

$$\min_{\boldsymbol{w}\in\mathbb{R}^n, b\in\mathbb{R}} \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + b - y^{(i)})^2$$

为了简化算法描述中的记号，将线性回归中的 n 维向量 $\boldsymbol{x}$ 的首位之前增补一个常数 1，使其成为一个 $n+1$ 维向量 $\tilde{\boldsymbol{x}}$，即 $\tilde{\boldsymbol{x}}=(1,\boldsymbol{x})$。令 $\tilde{\boldsymbol{w}}=(b,\boldsymbol{w})$，即将截距 b 用 w_0 表示，得到 $n+1$ 维的 $\tilde{\boldsymbol{w}}$。后面仍将用 n 维的 $\boldsymbol{x}$ 和 $\boldsymbol{w}$ 表示增广后的 $n+1$ 维向量，得到简化记号后的线性回归算法。

线性回归算法（简化记号）描述如下：

Algorithm 2: 线性回归算法

样本空间 $X\subseteq\mathbb{R}^n$，每个样本 $\boldsymbol{x}\in X$ 首位是 1，$y\in\mathbb{R}$

Input: m 个训练数据 $S=\{(\boldsymbol{x}^{(1)},y^{(1)}),(\boldsymbol{x}^{(2)},y^{(2)}),\cdots,(\boldsymbol{x}^{(m)},y^{(m)})\}$

Output: 线性模型 $h(\boldsymbol{x})=\boldsymbol{w}^T\boldsymbol{x}$，使得 $\boldsymbol{w}^*$ 为如下优化问题的最优解:

$$\min_{\boldsymbol{w}\in\mathbb{R}^n}\frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)}-y^{(i)})^2$$

将训练集 $S = \{(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \cdots, (\boldsymbol{x}^{(m)}, y^{(m)})\}$ 的特征和标签都使用矩阵描述，即：

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}^{(1)T} \\ \boldsymbol{x}^{(2)T} \\ \vdots \\ \boldsymbol{x}^{(m)T} \end{pmatrix}, \boldsymbol{y} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix}$$

其中，$\boldsymbol{X}$ 是一个 $m \times n$ 矩阵，$\boldsymbol{y}$ 是一个 $m \times 1$ 的列向量。$\boldsymbol{X}$ 称为特征矩阵，$\boldsymbol{y}$ 称为标签向量，则线性回归算法的目标函数等价于

$$\min_{\boldsymbol{w} \in \mathbb{R}^n} F(\boldsymbol{w}) = ||\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}||^2$$

当 $\boldsymbol{X}^T\boldsymbol{X}$ 可逆时，上式有唯一最优解

$$\boldsymbol{w}^* = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

先复习一下函数对矩阵变量的导数计算方法

数量函数对矩阵变量的导数

设 $f(\boldsymbol{X})$ 是以矩阵 $\boldsymbol{X}=(x_{ij})_{m\times n}$ 为自变量的 mn 元函数，且 $\frac{\partial f}{\partial x_{ij}}(i=1,2,\cdots,m;j=1,2,\cdots,n)$ 都存在，规定 f 对矩阵变量 $\boldsymbol{X}$ 的导数 $\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{X}}$ 为

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{X}}=\left(\frac{\partial f}{\partial x_{ij}}\right)_{m\times n}=\begin{pmatrix}\frac{\partial f}{\partial x_{11}} & \cdots & \frac{\partial f}{\partial x_{1n}}\\ \vdots & & \vdots\\ \frac{\partial f}{\partial x_{m1}} & \cdots & \frac{\partial f}{\partial x_{mn}}\end{pmatrix}$$

特别地，以 $\boldsymbol{x}=(\xi_1,\xi_2,\cdots,\xi_n)^T$ 为自变量的函数 $f(\boldsymbol{x})$ 的导数

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}=\left(\frac{\partial f}{\partial \xi_1},\frac{\partial f}{\partial \xi_2},\cdots,\frac{\partial f}{\partial \xi_n}\right)$$

称为数量函数对向量变量的导数，也就是数学分析中函数 f 的梯度向量，即为 $\mathrm{grad}f$ 或 ∇f

设 $\boldsymbol{a}=(a_1,a_2,\cdots,a_n)^T$ 是给定的向量，$\boldsymbol{x}=(\xi_1,\xi_2,\cdots,\xi_n)^T$ 是向量变量，且

$$f(\boldsymbol{x})=\boldsymbol{a}^T\boldsymbol{x}=\boldsymbol{x}^T\boldsymbol{a}$$

则 $\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}=\boldsymbol{a}$。

因为

$$f(\boldsymbol{x})=\sum_{k=1}^{n}a_k\xi_k$$

而

$$\frac{\partial f}{\partial \xi_j}=a_j\ \ (j=1,2,\cdots,n)$$

所以，

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}=\left(\frac{\partial f}{\partial \xi_1},\frac{\partial f}{\partial \xi_2},\cdots,\frac{\partial f}{\partial \xi_n}\right)^T=(a_1,a_2,\cdots,a_n)^T=\boldsymbol{a} \tag{3}$$

设 $\boldsymbol{A}=(a_{ij})_{n\times n}$ 是给定的矩阵，$\boldsymbol{x}=(\xi_1,\xi_2,\cdots,\xi_n)^T$ 是向量变量，且 $f(\boldsymbol{x})=\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}$，则 $\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}=(\boldsymbol{A}^T+\boldsymbol{A})\boldsymbol{x}$，特别地，当 $\boldsymbol{A}$ 是对称矩阵时，有 $\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}=2\boldsymbol{A}\boldsymbol{x}$。

因为

$$f(\boldsymbol{x})=\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}=\sum_{s=1}^{n}\sum_{k=1}^{n}\xi_s a_{sk}\xi_k=\sum_{s=1}^{n}\xi_s(\sum_{k=1}^{n}a_{sk}\xi_k),$$

而

$$\begin{aligned}\frac{\partial f}{\partial \xi_j}&=\xi_1 a_{1j}+\cdots+\xi_{j-1}a_{j-1,j}+(\sum_{k=1}^{n}a_{jk}\xi_k+\xi_j a_{jj})+\xi_{j+1}a_{j+1,j}+\cdots+\xi_n a_{nj}\\&=\sum_{s=1}^{n}a_{sj}\xi_s+\sum_{k=1}^{n}a_{jk}\xi_k\end{aligned}$$

所以，

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}=\begin{pmatrix}\frac{\partial f}{\partial \xi_1}\\ \vdots \\ \frac{\partial f}{\partial \xi_n}\end{pmatrix}=\begin{pmatrix}\sum_{s=1}^{n}a_{s1}\xi_s+\sum_{k=1}^{n}a_{1k}\xi_k\\ \vdots \\ \sum_{s=1}^{n}a_{sn}\xi_s+\sum_{k=1}^{n}a_{nk}\xi_k\end{pmatrix}=\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{x}+\boldsymbol{A}\boldsymbol{x}=(\boldsymbol{A}^T+\boldsymbol{A})\boldsymbol{x}\tag{4}$$

前面的线性回归算法的目标函数

$$\min_{w \in \mathbb{R}^n} F(w) = ||Xw - y||^2$$

$$\begin{aligned} F(w) = ||Xw - y||^2 &= (Xw - y)^T(Xw - y) = ((Xw)^T - y^T)(Xw - y) \\ &= (Xw)^T Xw - (Xw)^T y - y^T Xw + y^T y \\ &= w^T X^T Xw - w^T X^T y - y^T Xw + y^T y \end{aligned}$$

由(3),$\nabla_w w^T X^T y = X^T y$，$\nabla_w y^T Xw = X^T y$。由(4)，因为 $X^T X$ 是对称矩阵，$\nabla_w w^T X^T Xw = 2X^T Xw$。$\nabla_w y^T y = 0$。因此

$$\nabla_w F(w) = 2X^T Xw - 2X^T y$$

令 $\nabla_w F(w) = 0$，则有

$$\begin{aligned} 2X^T Xw - 2X^T y &= 0 \\ X^T Xw &= X^T y \end{aligned}$$

如果 $X^T X$ 可逆，方程两边左乘 $(X^T X)^{-1}$，得到

$$w = (X^T X)^{-1} X^T y$$

直接推导：因为 $F(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} - y^{(i)})^2$，为了求得上述目标函数 $F(\boldsymbol{w})$ 的最小值，将 F 对权重 $\boldsymbol{w}$ 求导并令其为 0，对于某一个 w_j，$F'(w_j) = \frac{2}{m}\sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} - y^{(i)}) \cdot x_j^{(i)}$，所以 $\frac{\partial F(\boldsymbol{w})}{\partial \boldsymbol{w}}$ 为：

$$\frac{\partial F}{\partial \boldsymbol{w}} = \begin{pmatrix} \frac{\partial F}{\partial w_0} \\ \frac{\partial F}{\partial w_1} \\ \vdots \\ \frac{\partial F}{\partial w_n} \end{pmatrix} = \frac{2}{m} \cdot \begin{pmatrix} \sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} - y^{(i)}) \cdot 1 \\ \sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} - y^{(i)}) \cdot x_1^{(i)} \\ \vdots \\ \sum_{i=1}^{m}(\boldsymbol{w}^T\boldsymbol{x}^{(i)} - y^{(i)}) \cdot x_n^{(i)} \end{pmatrix} =$$

$$\frac{2}{m} \cdot \left\{ \left(\boldsymbol{w}^T\boldsymbol{x}^{(1)} - y^{(1)}, \boldsymbol{w}^T\boldsymbol{x}^{(2)} - y^{(2)}, \cdots, \boldsymbol{w}^T\boldsymbol{x}^{(m)} - y^{(m)}\right) \cdot \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_n^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_1^{(m)} & x_2^{(m)} & \cdots & x_n^{(m)} \end{pmatrix} \right\}^T =$$

$$\frac{2}{m} \cdot \{(X\boldsymbol{w} - y)^T \cdot X\}^T = \frac{2}{m} \cdot X^T \cdot (X\boldsymbol{w} - y) = 0$$

有 $X^T(Xw - y) = 0$，因此，如果 X^TX 可逆，有最优解 $\boldsymbol{w} = (X^TX)^{-1}X^Ty$。

案例

均方误差式线性回归问题的目标函数，为了能够直观度量模型效果，需要为均方误差设定标尺。设 $\bar{y}=\frac{1}{m}\sum_{i=1}^{m}y^{(i)}$ 为训练数据标签平均值。以训练数据中标签的平均值作为任意样本的标签预测，因为它只利用了训练数据的标签而忽略了特征，其预测能力式有限的。因此可将平均值模型 h_{avg} 的均方误差作为其他模型拟合效果的度量参考标尺。

（定义）设 $\bar{y}=\frac{1}{m}\sum_{i=1}^{m}y^{(i)}$ 为训练数据标签平均值，则 $R^2=1-\frac{\sum_{i=1}^{m}(h(x^{(i)})-y^{(i)})^2}{\sum_{i=1}^{m}(\bar{y}-y^{(i)})^2}$ 称为模型 h 的决定系数。

如果一个模型的均方误差接近 h_{avg} 均方误差，说明这个模型的拟合效果就差。这时 $R^2\approx 0$

使用正规方程实现的线性回归类见书图 3.4 代码

案例实战 3.1：线性回归预测加州房价。

案例实战 3.2：线性回归预测糖尿病。

目录

有些实际问题中，标签与特征的关系并非线性的，而是呈多项式的关系。在这种情形下，标签与特征之间的关系就称为多项式关系。由于任一函数都可以用多项式逼近（泰勒展开），因此多项式回归有着广泛应用。

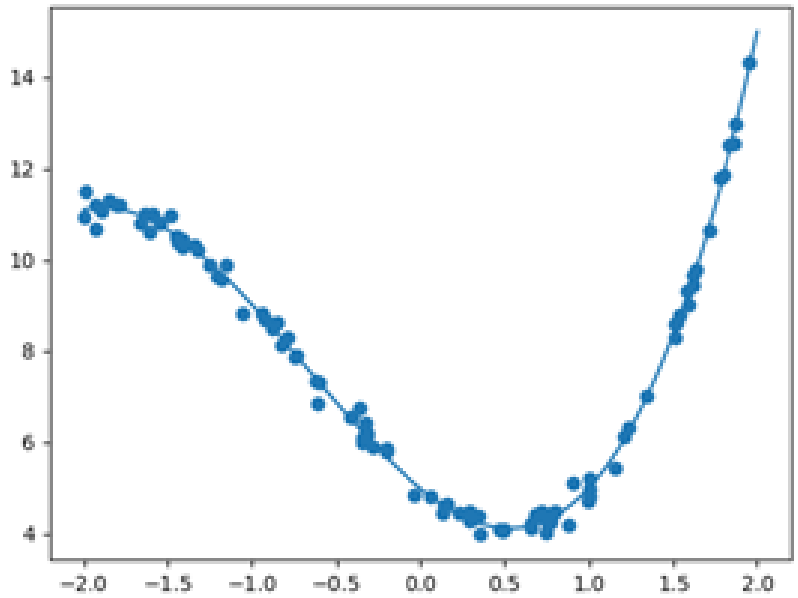


图 1: 多项式拟合模型结果

案例：书图 3-8 多项式模型拟合

目录

机器学习中普遍认为，参数 $\boldsymbol{w}$ 的范数越小，模型就越简单。由奥卡姆剃刀法则，如果两个模型在训练数据上的经验损失接近，则应该选择参数范数较小的哪一个。正则化是一种用于回归的技术，可以降低模型的复杂性并缩小独立特征的系数。该技术将复杂的模型转换为更简单的模型，从而避免了过拟合的风险并缩小了系数，从而降低了计算成本。常用的正则化方法有 L_1 和 L_2 正则化方法，它们分别用的是 L_1 范数和 L_2 范数。

（定义）线性回归的 L_1 正则化目标函数 $\min_{\boldsymbol{w}\in\mathbb{R}^n} F(\boldsymbol{w}) = \frac{1}{m}||\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}||^2 + \lambda|\boldsymbol{w}|$ 称为 LASSO（Least absolute shrinkage and selection operator）回归，其中 $\lambda(\lambda > 0)$ 为正则化系数。

（定义）线性回归的 L_2 正则化目标函数 $\min_{\boldsymbol{w}\in\mathbb{R}^n} F(\boldsymbol{w}) = \frac{1}{m}||\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}||^2 + \lambda||\boldsymbol{w}||^2$ 称为岭回归，其中 $\lambda(\lambda > 0)$ 为正则化系数。

正则化系数 λ 控制正则化的强度，随着 λ 的增大，模型变得越来越简单，算法的输出可能会导致拟合不足，即在训练数据上的经验损失过大，所以 λ 应该选择合适的大小。

对于 L_2 正则化，由 $\min_{\boldsymbol{w}\in\mathbb{R}^n} F(\boldsymbol{w}) = \frac{1}{m}||\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}||^2 + \lambda||\boldsymbol{w}||^2$，有：

$$\nabla_{\boldsymbol{w}} F(\boldsymbol{w}) = 2X^T Xw + 2\lambda I\boldsymbol{w} - 2X^T y$$

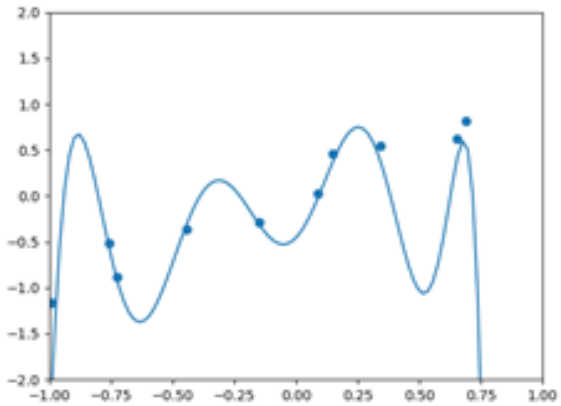
令 $\nabla_{\boldsymbol{w}} F(\boldsymbol{w}) = 0$，得到：

$$(X^T X + \lambda I)\boldsymbol{w} = X^T y$$
$$\boldsymbol{w} = (X^T X + \lambda I)^{-1} X^T y$$

利用该结论，可编写出 ridge_regression.py

正则化拟合多项式

案例实战 3.3 对比有无 L_2 正则化的模型拟合多项式。



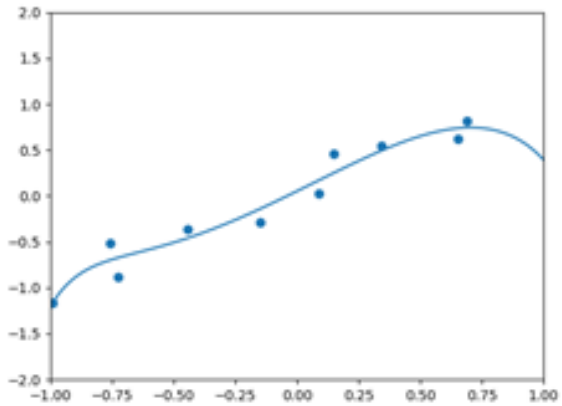


图 2: 图 3-13 $\lambda = 0$ （左）和 $\lambda = 0.01$（右）的 L_2 正则化多项式回归结果

目录

Sklearn 的线性回归

sklearn 库中 sklearn.linear_model 提供了多种支持线性回归分析的类。其中最基本的一个是普通最小二乘线性回归。class sklearn.linear_model.LinearRegression(*, fit_intercept=True, normalize=False, copy_X=True, n_jobs=None)。LinearRegression 使用系数 $w = (w_1, \cdots, w_p)$ 拟合线性模型，以最小化数据集中实际目标值与通过线性逼近预测的目标之间的残差平方和。

案例 3.4 仅使用 diabetes 数据集的第一个特征，使用 Sklearn 提供的方法实现线性回归。