

lec08

Regularisation (正则化)

正则化是通过约束模型来减少模型过度拟合的过程 (降低参数的复杂性/数量)。对于使用权重向量的分类器，可以通过最小化权重向量的**范数 (norm)** (长度) 来完成正则化

常见的正则化算法：

- L2 regularisation (ridge regression 岭回归 or Tikhonov regularisation 吉洪诺夫正则化)
- L1 regularisation (Lasso regression)
- L1 + L2 regularisation (mixed regularisation)

她 PPT 🙌 写的一坨，我来解释

正则化的形式

正则化的作用是减少权重向量的幅度；对于较大的权重，正则化会惩罚权重的大值。

- **L1 正则化** (Lasso 正则化) 添加的惩罚项是权重的绝对值之和，即 $\lambda \sum |w_i|$
这种形式倾向于生成稀疏权重矩阵，即很多权重会变为零，从而在模型中进行特征选择（没有这个特征，直接就是0）
- **L2 正则化** (岭回归或 Tikhonov 正则化) 添加的惩罚项是权重的平方和，即 $\lambda \sum w_i^2$
这种形式倾向于将权重均匀地减小，而不会将它们减到零，这有助于处理因特征之间的相关性而引起的问题（如多重共线性）

惩罚机制

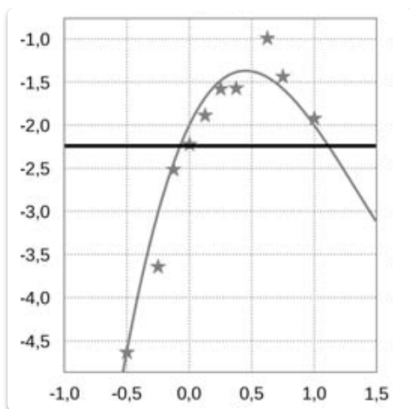
总成本函数包含了原始的损失函数和正则化项（下面有介绍），增加正则化项实际上是对大权重的惩罚。这意味着如果权重值增大，模型的总损失也会增大（可以从损失函数的计算公式中发现，权重过大的项对损失函数的结果有着非常大的影响），因此在训练过程中算法会倾向于选择较小的权重值，以最小化总成本。

训练模型时，权重的更新不仅受到数据误差（由损失函数衡量）的影响，还受到正则化项的影响。对于 L2 正则化，每次权重更新都会减小一定比例的当前权重值，从而有效控制权重不会变得过大

通过惩罚大的权重值，正则化帮助模型避免对训练数据中的噪声或非代表性特征过度敏感。较小的权重减少了模型的复杂度，使模型更容易泛化到新数据上

回到第一个标题

举例来说，对于函数 $f(x) = x^3 - 4x^2 + 3x - 2$ ，我们在其上面采样若干点，并且在 y 方向上加入一些噪声，结果如下图



我们的目的是在不知道原函数的情况下，根据这些加入了噪声的点，进行多项式拟合：

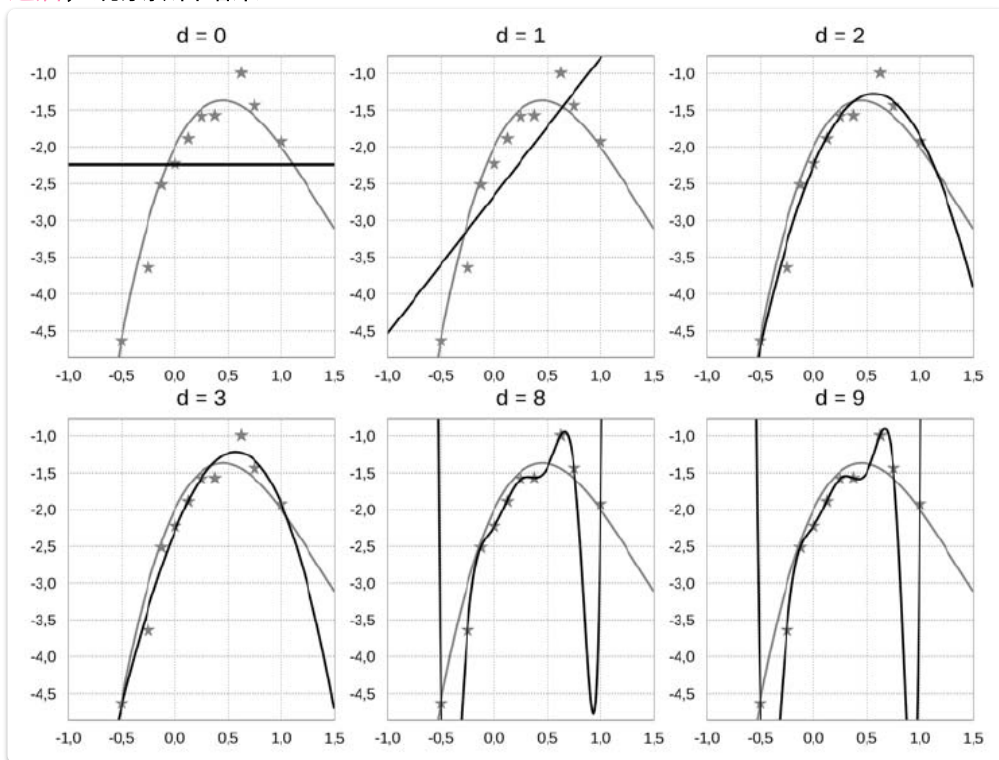
$$\hat{y}(x, W) = w_0 + \sum_{j=1}^d w_j x^j = (1, x, x^2, \dots, x^d) \cdot W$$

使得下面的损失函数最小化：

$$L(\mathcal{D}, W) = \sum_{i=1}^n (\hat{y}(x_i, W) - y_i)^2$$

其中， $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ 是原函数中加入噪声的采样点， d 是多项式的最高次幂

之后，观察拟合结果：



但是：

$$f_0(x) = -2.2393$$

$$f_1(x) = -2.6617 + 1.8775x$$

$$f_2(x) = -2.2528 + 3.4604x - 3.0603x^2$$

$$f_3(x) = -2.2937 + 3.5898x - 2.6538x^2 - 0.5639x^3$$

$$f_8(x) = -2.2324 + 2.2326x + 6.2543x^2 + 15.5996x^4 - 239.9751x^4 + 322.8516x^5 \\ + 621.0952x^6 - 1478.6505x^7 + 750.9032x^8$$

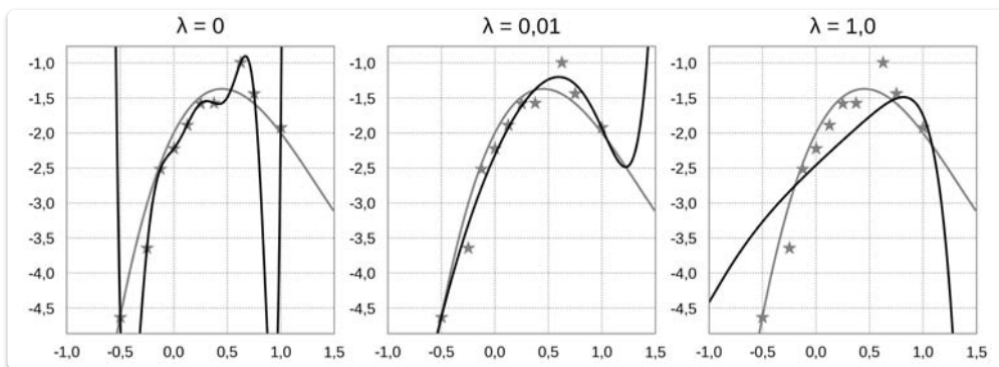
$$f_9(x) = -2.225 + 2.014x + 4.882x^2 + 31.13x^3 - 230.31x^4 + 103.72x^5 + 869.22x^6 \\ - 966.67x^7 - 319.31x^8 + 505.64x^9$$

参数增长的幅度过大，我们需要进行**约束**，增加 $\lambda \|W\|^2$ 项：

$$L(\mathcal{D}, W) = \sum_{i=1}^n (\hat{y}(x_i, W) - y_i)^2 + \lambda \|W\|^2$$

对于 $d = 9$ ，有

$$\begin{aligned} f_{\lambda=0}(x) &= -2.22 + 2.01x + 4.88x^2 + 31.13x^3 - 230.31x^4 + 103.72x^5 + 869.22x^6 \\ &\quad - 966.67x^7 - 319.31x^8 + 505.64x^9 \\ f_{\lambda=0.01}(x) &= -2.32 + 3.40x - 2.33x^2 + 0.05x^3 - 0.51x^4 - 0.29x^5 - 0.22x^6 \\ &\quad - 0.06x^7 + 0.09x^8 + 0.24x^9 \\ f_{\lambda=1}(x) &= -2.46 + 1.45x - 0.19x^2 + 0.22x^3 - 0.13x^4 - 0.05x^5 - 0.14x^6 \\ &\quad - 0.13x^7 - 0.16x^8 - 0.16x^9 \end{aligned}$$



可以发现，多项式系数的绝对值显著减小，同时拟合程度也有提高

这正是因为正则化使得每次梯度更新不会得着大权重硬凑，转而去修改小权重，最后得到更好的效果

L2 regularisation

我们对 W 施加 L2 正则化，要最小化的整体目标可以写成如下形式：

$$J(\mathcal{D}, W) = L(\mathcal{D}, W) + \lambda \|W\|_2^2 = L(\mathcal{D}, W) + \lambda \sum_{i=1}^d w_i^2$$

其中， λ 被称为**正则化系数**，通常通过**交叉验证**来设定

整体目标的梯度就变成了损失梯度与权重向量 W 的和：

$$\nabla_W J(\mathcal{D}, W) = \nabla_W L(\mathcal{D}, W) + 2\lambda W$$

而 SGD（随机梯度下降）更新规则会通过一个负的学习率 μ 乘以损失梯度，因此，L2 正则化的更新规则将包含一个 $-2\mu\lambda W$ 的项，如下例所示

$$\begin{aligned} W &\leftarrow W - \mu(y_i \cdot X_i + 2\lambda W) \\ &= W + \mu \cdot y_i \cdot X_i - 2\mu\lambda W \\ &= W + y_i \cdot X_i - 2\lambda W \\ &= (1 - 2\lambda) \cdot W + y_i \cdot W \end{aligned}$$

其中，对于第三行的变换， $\mu = 1$ ； y_i 指样本标签，这里应该是 -1 或 1 ； $W \leftarrow W + \mu y_i X_i$ 是基本感知机更新规则

设置 λ

将数据集分为**训练集**和**验证集**，在对数尺度上尝试不同的 λ ，分别训练后验证

K-Nearest Neighbours

训练： 储存整个训练集

分类： 对于一个输入 X' ，找到模型中 k 个最邻近的对象，这 k 个对象中占主导的标签就是 X' 的预测结果

衡量相似性/距离

矩阵

$X = (x_1, \dots, x_d)$, $Y = (y_1, \dots, y_d)$ 是 \mathbb{R}^n 中的向量

范数

- L1 范数是向量中各个元素绝对值的总和
- L2 范数是向量元素平方和的平方根
- 无穷范数表示向量中的最大元素的绝对值
- L0 范数是向量中非 0 元素的个数

Cosine：

$$\text{CosSim}(X, Y) = \frac{X^T Y}{\|X\| \|Y\|} = \cos(\theta)$$

其中, $\|X\| = \sqrt{\sum_{i=1}^d x_i^2}$, θ 是 X 和 Y 的夹角

Euclidean：

$$\text{EucDist}(X, Y) = \sqrt{\sum_{i=1}^d (x_i - y_i)^2} = \sqrt{(X - Y)^T (X - Y)}$$

Manhattan：

$$\text{ManDist}(X, Y) = \|X - Y\|_1 = \sum_{i=1}^d |x_i - y_i|$$

集合

Jaccard similarity coefficient：

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Overlap Coefficient：

$$\text{overlap}(X, Y) = \frac{|X \cap Y|}{\min(|X|, |Y|)}$$

Hamming distance：

$$d_H(A, B) = |A \triangle B| = (A \setminus B) \cup (B \setminus A)$$