线性回归（拟合）

概念：**均方误差（mean-square error、MSE）、最小二乘法（Least Square Method， LSM）、多元线性回归/多变量线性回归、正则化（regularization）、对数线性回归（log-linear regression）、广义线性模型（generalized linear model）、联系函数（link function）**

在只有一个属性的数据集中，在单变量线性回归试图学得：

$$f(x_i) = w \cdot x_i + b, \text{使得} f(x_i) \simeq y_i$$

可以使用**均方误差（mean-square error、MSE）**来衡量模型的好坏，因此让均方误差最小化来解 w,b：

$$\begin{aligned}(w^*, b^*) &= arg\min_{w,b} \sum_{i=1}^{m} (f(x_i) - y_i)^2 \\ &= arg\min_{w,b} \sum_{i=1}^{m} (y_i - w \cdot x_i - b)^2\end{aligned} \tag{A1}$$

均方误差有很好的几何意义，她对应了常用的欧式距离。**基于均方误差最小化来进行模型求解的方法称为最小二乘法（Least Square Method， LSM）**。在线性回归中，最小二乘法就是试图找到一条直线，使得所有的样本到直线上的欧氏距离之和最小。

求解 w 和 b 使得 $E_{w,b} = \sum_{i=1}^{m} (y_i - w \cdot x_i - b)^2$ 最小化的过程，称为**线性回归模型的最小二乘参数估计**。将 $E_{w,b}$ 对 w 和 b 求导，然后让导数为零就可得到 w 和 b 的最优**闭式解**：

$$\begin{aligned}\frac{\partial E_{(w,b)}}{\partial w} &= 2\sum_{i=1}^{m} (y_i - w \cdot x_i - b) \cdot (-x_i) \\ &= 2\left(w\sum_{i=1}^{m} x_i^2 - \sum_{i=1}^{m}(y_i - b)x_i\right)\end{aligned} \tag{1.1}$$

$$
\begin{aligned}
\frac{\partial E_{(w,b)}}{\partial b} &= 2\sum_{i=1}^{m}(y_i - w\cdot x_i - b)\cdot(-1) \\
&= 2\left(mb - \sum_{i=1}^{m}(y_i - w\cdot x_i)\right)
\end{aligned}
\tag{1.2}
$$

令 1.1 和 1.2 式为零就可以求得 w 和 b：

$$
w = \frac{\sum_{i=1}^{m} y_i(x_i - \overline{x})}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m} x_i^2\right)^2} \qquad \text{其中 } \overline{x} = \frac{1}{m}\sum_{i=1}^{m} x_i
$$

$$
b = \frac{1}{m}\sum_{i=1}^{m}(y_i - wx_i)
$$

对于有多个属性的数据集，如数据集 D 含有 d 个属性，学到的模型为：

$$
f(\boldsymbol{x_i}) = \mathbf{w^T}\boldsymbol{x_i} + b, \text{使得} f(x_i) \simeq y_i \tag{A2}
$$

称为**多元线性回归**，或**多变量线性回归**。

同样可以使用最小二乘法来对 w 和 b 进行估计。为了方便讨论可以将 w 和 b 合成一个变量$\hat{\boldsymbol{w}} = (\boldsymbol{w}, b)$，那么数据集 D 就可以表示为一个 m×(d+1)的矩阵 X，每一行代表一个样本，每一列代表一个属性，最后一行恒为1，对应于 b。形式如下：

$$
\mathbf{X} = \begin{pmatrix}
x_{11} & x_{12} & \cdots & x_{1d} & 1 \\
x_{21} & x_{12} & \cdots & x_{1d} & 1 \\
\vdots & & & & 1 \\
x_{m1} & x_{m2} & \cdots & x_{md} & 1
\end{pmatrix} = \begin{pmatrix}
\boldsymbol{x_1}^T & 1 \\
\boldsymbol{x_2}^T & 1 \\
\vdots & \vdots \\
\boldsymbol{x_m}^T & 1
\end{pmatrix}
$$

如果将标签写为 $\boldsymbol{y} = (y_1; y_2; \cdots; y_m)$，那么就有与 A1 类似的式子：

$$
\hat{w}^* = arg\min_{\hat{w}} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})^T(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}}) \text{ 或 } \min_{\hat{w}} ||\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}||_2^{\,2}
$$

对上式对$\hat{w}$ 求导，并零导数为零，求得最优解：

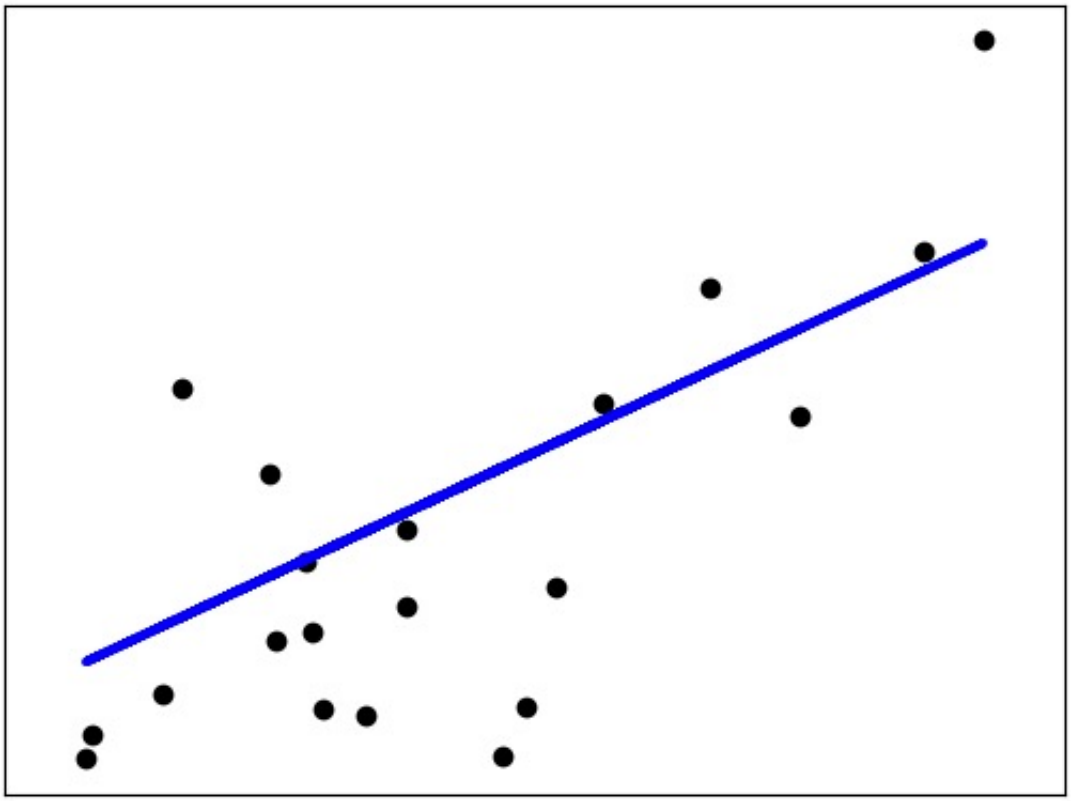
$$
\begin{aligned}
\frac{\partial E_{\hat{w}}}{\partial \hat{w}} &= 2\mathbf{X^T}(\mathbf{X}\hat{w} - \boldsymbol{y}) = 0 \\
&\Rightarrow \qquad \mathbf{X^T}\mathbf{X}\hat{w} = \mathbf{X^T}\boldsymbol{y} \\
&\Rightarrow \qquad \hat{w} = (\mathbf{X^T}\mathbf{X})^{-1}\mathbf{X^T}\boldsymbol{y}
\end{aligned}
$$

如果$\mathbf{X^T}\mathbf{X}$ 是满秩矩阵或者正定矩阵，w 的解就为上面的值，令 $\hat{\boldsymbol{x_i}} = (\boldsymbol{x_i})$ 那么此时的线性回归模型为：

$$
f(\hat{\boldsymbol{x_i}}) = \hat{\boldsymbol{x_i}}^T(\mathbf{X^T}\mathbf{X})^{-1}\mathbf{X^T}\boldsymbol{y}
$$

但在显示任务中矩阵$\mathbf{X}^{\mathrm{T}}\mathbf{X}$往往不是满秩的，即经常会遇到属性的个数远远大于样本的个数，此时 w 就不止有一个解，选择哪一个作为解，由算法的偏好来决定。常见的做好是引入**正则化（regularization）**

如下图，未引入正则化的线性回归：[sklearn - Linear Regression Example]



线性模型简单但却有丰富的变化，对于样本$(\boldsymbol{x}, y), y \in \mathbb{R}$，当希望线性模型 A2 预测值逼近真是标记 y 时，就得到了线性模型，可以将其简写为：

$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

我们也可以让其逼近 y 的衍生物，如标记是在指数尺度上的变化，那么输出的标记的对数作为线性模型逼近的目标即：

$$\ln y = \boldsymbol{w}^T \boldsymbol{x} + b \tag{1.4}$$

这就是**对数线性回归（log-linear regression）**，他其实是在让 $e^{\boldsymbol{w}^T \boldsymbol{x}+b}$ 来接近 y。上式虽然在形式上式线性回归，但实际上已经是在**求取输入空间到输出空间的非线性映射函数**

考虑到单调可以函数g(.)，令：

$$y = g^{-1}\left(\boldsymbol{w}^T \boldsymbol{x} + b\right) \tag{A3}$$

这样得到的模型为**广义线性模型（generalized linear model）**，g(.)称为**联系函数（link function）**（将线性模型回归的预测值与真实值联系起来），显然对数线性归回是广义线性模型 g(.) = ln(.)的特例。

小知识：

- 机器学习小组知识点1：均方误差(MSE)

误差的几种分类

1. SSE(和方差、误差平方和)：The sum of squares due to error
2. MSE(均方差、方差)：Mean squared error
3. RMSE(均方根、标准差)：Root mean squared error

方差是在概率论和统计方差衡量随机变量或一组数据的离散程度的度量方式，方差越大，离散度越大，以下是几种计算方法：

- 平均数

$$M = \frac{x_1 + x_2 + \cdots + x_n}{n}$$

- 方差

$$s^2 = \frac{(x_1 - M)^2 + (x_1 - M)^2 + \cdots + (x_n - M)^2}{n}$$

或

$$D(x) = E(x^2) - (E(x))^2$$

- 标准差

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

- 样本方差

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2$$

- SSE

$$SSE = \sum_{i=1}^{m} w_i(y_i - \hat{y}_i)^2$$

- MSE

$$MSE = \frac{SSE}{n} = \frac{1}{n}\sum_{i=1}^{m} w_i(y_i - \hat{y}_i)^2$$

- RMSE 又叫拟合误差

$$RMSE = \sqrt{MSE} = \sqrt{\frac{SSE}{n}} = \sqrt{\frac{1}{n}\sum_{i=1}^{m} w_i(y_i - \hat{y}_i)^2}$$

对数几率回归（分类）

相关概念：**单位阶跃函数（unit-step function）、替代函数（surrogate function）、对数几率函数（logistics function）、几率（odds）、对数几率（log odds 也做 logit）、对数几率模型（logistics regression或者 logit regression）、后验概率估计、极大似然法（maximum likelihood Method）、对数似然（log-likelihood）、梯度下降算法（gradient descent Method）、牛顿法（newton method）**

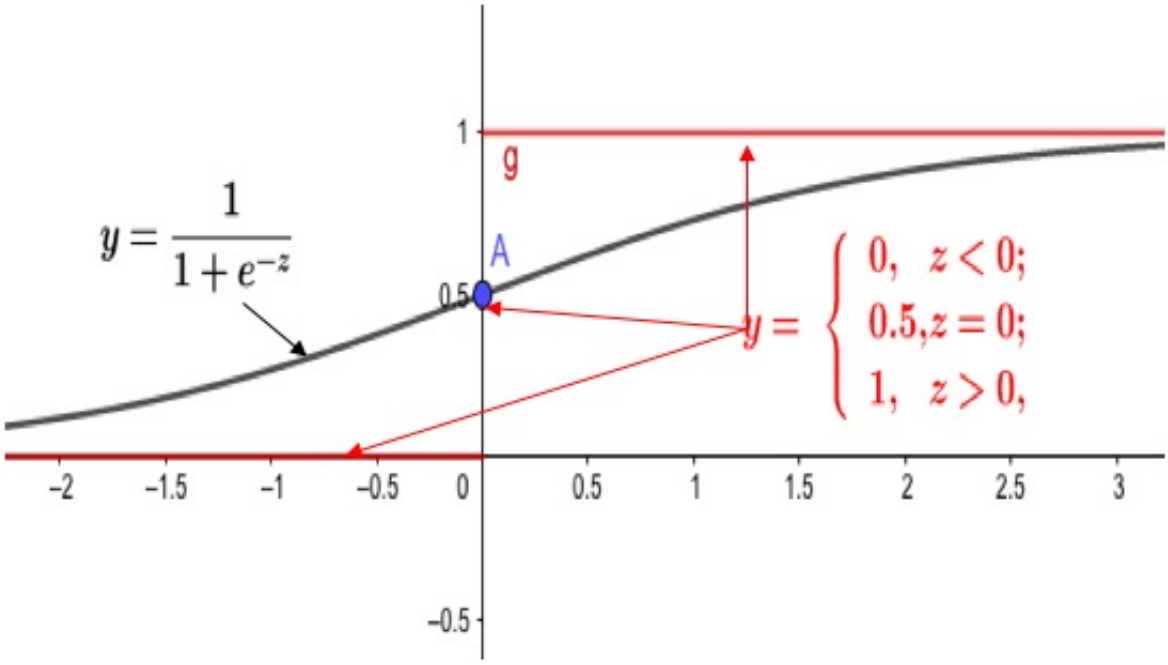
【参考】

- csdn - 极大似然估计详解

二分类任务中，输出的标记为$y \in \{0, 1\}$，线性回归模型产生的实值，$z = \boldsymbol{w}^T \boldsymbol{x} + b$，我们就需要将z值转为0/1，这时候就需要用到**单位阶跃函数（unit-step function）**：

$$y = \begin{cases} 0, & z < 0; \\ 0.5, & z = 0; \\ 1, & z > 0, \end{cases}$$

即预测值大于零是正例，小于零是负例，临界值任意判断，如下图：



单位阶跃函数是不连续的，因此不能作为联系函数，我们需要找到一个与单位阶跃函数近似的**替代函数（surrogate function）**，并且希望他是单调可微的，**对数几率函数（logistics function）**正是这样的一个替代函数：

$$y = \frac{1}{1 + e^{-z}} \tag{A4}$$

对数几率函数是一种"Sigmoid 的函数"，他将 z 值**转换**为一个接近 0 或者1 的值，且在输出值 z=0 的位置变化陡峭，将其作为$g^{-1}(\cdot)$带入 A3可得：

$$y = \frac{1}{1 + e^{-(\boldsymbol{w}^T \boldsymbol{x} + b)}} \tag{A5}$$

将 A5 写成与 (1.4) 类似的形式就有：

$$\ln \frac{y}{1-y} = \boldsymbol{w}^T \boldsymbol{x} + b \tag{A6}$$

y 可以当做样本 x 是正例的可能性，那么 1-y 就是负例的可能性，两者的比值成为**几率（odds）**：

$$\frac{y}{1-y}$$

反应了 x 作为正例的可能性，对几率去对数得到的就是**对数几率（log odds 也做 logit）**：

$$\ln \frac{y}{1-y}$$

由上可以看出，A5实际是在用线性回归模型的预测结果去接近真实标记的对数几率，因此该模型称为**对数几率模型（logistics regression或者 logit regression）**，虽然名字里有回归但其实是分类。

优点：

- 直接对分类的可能性进行建模，无需实现假设数据的分布，这样就避免了假设分布不准确带来的问题
- 她不仅预测出“类别”，还可以得到近似概率预测，这样需要利用概率辅助决策的任务很有用
- 对率函数是任意阶可导凸函数，有很好的数学性质，可直接用现成的算法包求解。

确定 w 和 b，将式 A5 中的 y 视为类**后验概率估计**p(y=1 | x)，那么式 A6 可以重新写为：

$$\ln \frac{p(y=1|x)}{p(y=0|x)} = \boldsymbol{w}^T \boldsymbol{x} + b \tag{1.4}$$

那么就有：

$$\begin{aligned} p(y=1|x) &= \frac{e^{\boldsymbol{w}^T \boldsymbol{x} + b}}{1 + e^{\boldsymbol{w}^T \boldsymbol{x} + b}} \\ p(y=0|x) &= \frac{1}{1 + e^{\boldsymbol{w}^T \boldsymbol{x} + b}} \end{aligned} \tag{1.5}$$

通过**极大似然法（maximum likelihood Method）**来估计 w 和 b。给定数据集$\{(x_i, y_i)\}_{i=1}^m$对率回归模型最大化**对数似然（log-likelihood）**：

$$\ell(\boldsymbol{w}, b) = \sum_{i=1}^{m} \ln p(y_i \mid \boldsymbol{x_i}; \boldsymbol{w}, b) \tag{A7}$$

即令每个样本属于这个标记的概率越大越好。令$\beta = (\boldsymbol{w}; b)$，$\hat{\boldsymbol{x}} = (\boldsymbol{x}; 1)$，那么

$$\boldsymbol{w}^T \boldsymbol{x} + b = \boldsymbol{\beta}^T \hat{\boldsymbol{x}}$$

再令：

$$p_1(\hat{\boldsymbol{x}}, \boldsymbol{\beta}) = p(y = 1 \mid \hat{\boldsymbol{x}}, \boldsymbol{\beta})$$
$$p_0(\hat{\boldsymbol{x}}, \boldsymbol{\beta}) = p(y = 0 \mid \hat{\boldsymbol{x}}, \boldsymbol{\beta}) = 1 - p_1(\hat{\boldsymbol{x}}, \boldsymbol{\beta})$$

那么重写 A7 中的似然项有：

$$p(y_i \mid \boldsymbol{x_i}; \boldsymbol{w}, b) = y_i p_1(\hat{\boldsymbol{x}}, \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\boldsymbol{x}}, \boldsymbol{\beta}) \tag{A8}$$

将 A8 带入 A7，根据 1.5 式，A8 有如下两种情况

- 当 $y_i = 1$ 时，

$$p(y_i \mid \boldsymbol{x_i}; \boldsymbol{w}, b) = p_1(\hat{\boldsymbol{x}}, \boldsymbol{\beta}) = \frac{e^{\boldsymbol{w}^T \boldsymbol{x} + b}}{1 + e^{\boldsymbol{w}^T \boldsymbol{x} + b}}$$

因此：

$$\ln p(y_i \mid \boldsymbol{x_i}; \boldsymbol{w}, b) = \boldsymbol{\beta}^T \hat{\boldsymbol{x}} - \ln\left(1 + e^{\beta^T \hat{x}_i}\right) \tag{a}$$

- 当 $y_i = 0$ 时，

$$p(y_i \mid \boldsymbol{x_i}; \boldsymbol{w}, b) = p_0(\hat{\boldsymbol{x}}, \boldsymbol{\beta}) = \frac{1}{1 + e^{\boldsymbol{w}^T \boldsymbol{x} + b}}$$

因此：

$$\ln p(y_i \mid \boldsymbol{x_i}; \boldsymbol{w}, b) = -\ln\left(1 + e^{\beta^T \hat{x}_i}\right) \tag{b}$$

结合(a)(b)，最大化 A7 ，相当于最小化：

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{m} \left(-y_i \boldsymbol{\beta}^T \hat{\boldsymbol{x}}_i + \ln\left(1 + e^{\beta^T \hat{x}_i}\right)\right) \tag{A9}$$

β是高阶连续可导函数，可用**梯度下降算法（gradient descent Method）**，或者**牛顿法（newton method）**来求解。

多重共线性

【参考】

- csdn - 机器学习线性回归：谈谈多重共线性问题及相关算法
- 简书 - 讲讲共线性问题
- 知乎 - 多元线性回归及多重共线性处理
- 博客园 - 多重共线性的解决方法之——岭回归与LASSO
- 百度文库 - 第七章 多重共线性（计量经济学-浙江大学 韩菁）

最小二乘法（OLS）在做回归时，一致地看待每一个样本点，是典型的无偏估计，会得到一个使得残差最小的权重参数。然而，在面对一堆数据集存在多重共线性时，最小二乘法（OLS）就变得对样本点的误差极为敏感，最终回归后的权重参数方差变大。这就是需要解决的共线性回归问题，一般思想是放弃无偏估计，损失一定精度，对数据做有偏估计，常用的方法就是将要提到的 岭回归和 LASSO。

多重共线性（Multicollinearity）是指线性回归模型中的自变量之间由于存在高度相关关系，而使模型的权重参数估计失真或难以估计准确的一种特性，多重是指一个自变量可能与多个其他自变量之间存在相关关系。
例如一件商品的销售数量可能与当地的人均收入和当地人口数这两个其他因素存在相关关系。在研究社会、经济问题时，因为问题本身的复杂性，设计的因素很多。在建立回归模型时，往往由于研究者认识水平的局限性，很难在众多因素中找到一组互不相关，又对因变量 y 产生主要影响的变量，不可避免地出现所选自变量出现多重相关关系的情形。

用数学语言表述就是，在多元线性回归中，我们求得的 w：

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

如果存在较强的共线性，即 X 中各列向量之间存在较强的相关性，会导致 $|\boldsymbol{X^T X}| \approx 0$, 从而引起 $(\boldsymbol{X^T X})^{-1}$ 对角线上的值很大。并且不一样的样本也会导致参数估计值变化非常大。即参数 $\hat{w}$ 估计量的方差也增大，对参数的估计会不准确。

为了解决这个问题，可以使用下面介绍的 岭回归和 LASSO 算。

岭回归(Ridge Regression)

【参考】

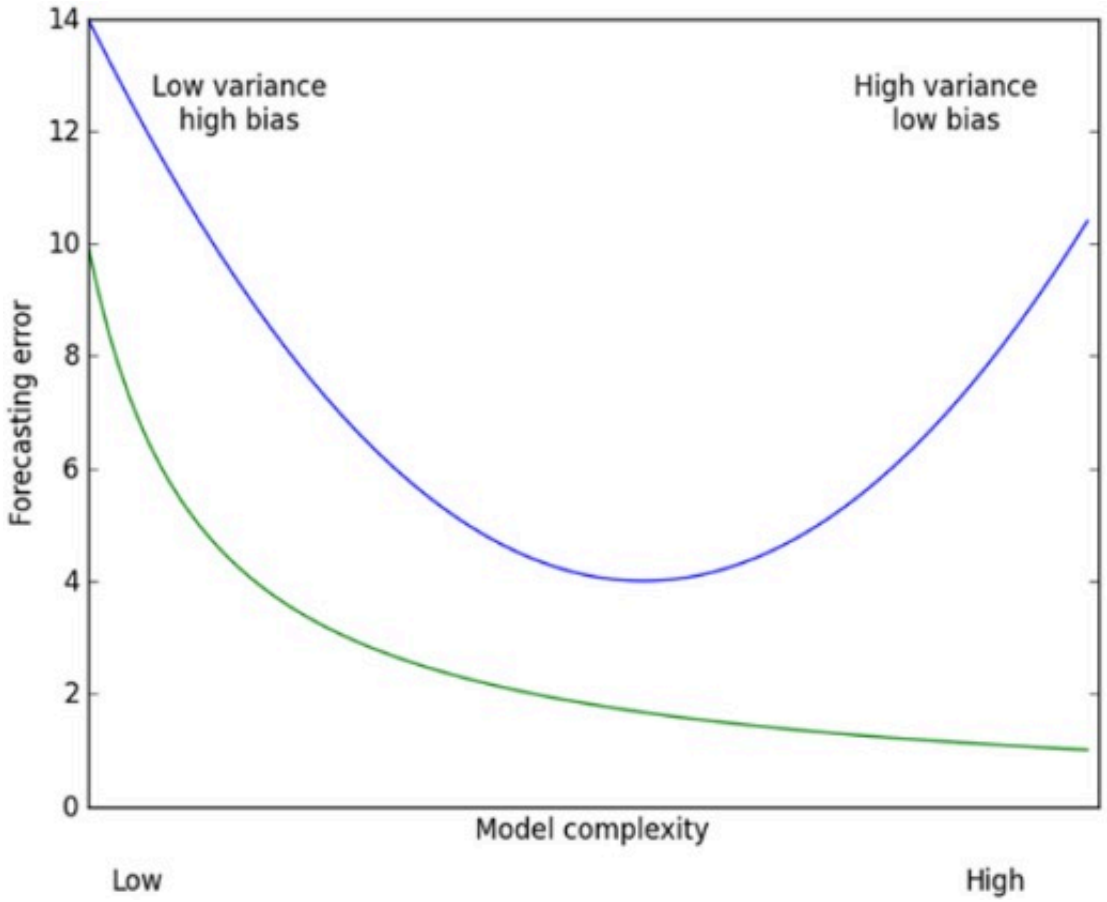
- csdn - 简单易学的机器学习算法——岭回归(Ridge Regression)
- 简书 - 岭回归

线性回归的问题

在处理复杂的数据的回归问题时，普通的线性回归会遇到一些问题，主要表现在：

- 预测精度：这里要处理好这样一对为题，即样本的数量 n 和特征的数量 p
  - n >> p 时，最小二乘回归会有较小的方差
  - n ≈ p 时，容易产生过拟合
  - n < p 时，最小二乘回归得不到有意义的结果
- 模型的解释能力：如果模型中的特征之间有相互关系，这样会增加模型的复杂程度，并且对整个模型的解释能力并没有提高，这时，我们就要进行特征选择。

以上的这些问题，主要就是表现在模型的方差和偏差问题上，这样的关系可以通过下图说明：



方差指的是模型之间的差异，而偏差指的是模型预测值和数据之间的差异。我们需要找到方差和偏差的折中。

岭回归

【参考】

- 简书 - 多元线性回归模型的特征压缩：岭回归和Lasso回归
- sklearn - ridge-regression
- 知乎- 机器学习算法实践-岭回归和LASSO
- csdn - 【机器学习】L1正则化与L2正则化详解及解决过拟合的方法

岭回归是在线性回归损失函数的基础之上添加正则项（L2范数）得到，对参数进行压缩惩罚（m 样本数，n 特征数量）：

$$f(\omega) = \sum_{i=1}^{m}(y_i - x_i^T\omega)^2 + \alpha\sum_{i=1}^{n}\omega_i^2 \tag{B1}$$

或者

$$\min_{\omega} {||\boldsymbol{X\omega} - \boldsymbol{y}||_2}^2 + \alpha||\boldsymbol{\omega}||_2^2 \tag{B2}$$

对 (B2) 式求导可有：

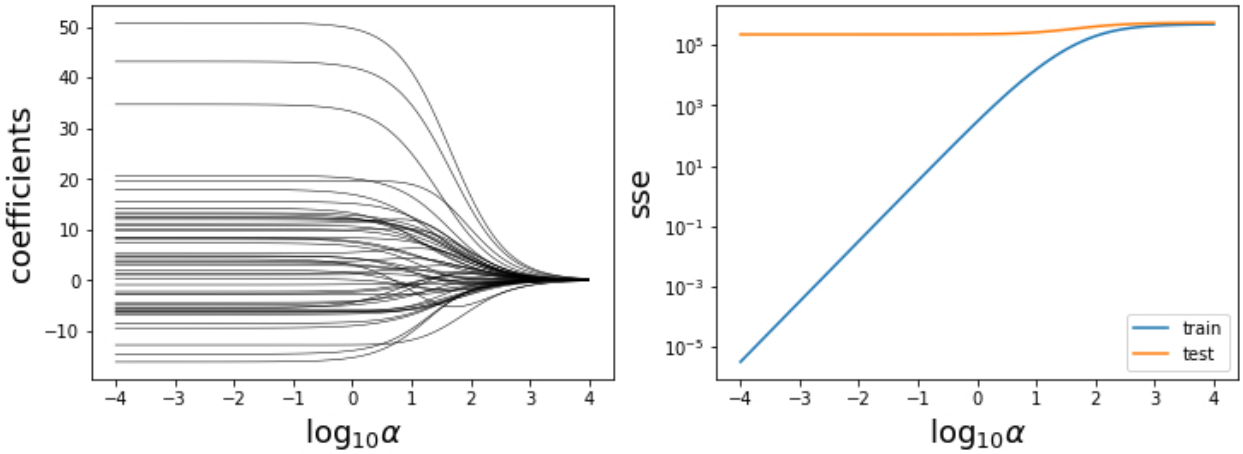
$$\boldsymbol{2X^T(y - X\omega)} - 2\alpha\boldsymbol{\omega} \tag{B3}$$

令其为 0 ，可以求得ω:

$$\hat{\omega} = (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X} + \alpha \boldsymbol{I})^{-1}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{y} \tag{B4}$$

岭回归使用了单位矩阵乘以常数 α，我们观察其中的单位矩阵 I,可以看到值 1 贯穿整个对角线，其余元素全是0. 形象的，在0构成的平面上有一条 1 组成的“岭”，这就是岭回归中“岭”的由来。岭回归又叫脊回归。

以下展示了不同的 α 对于参数 w 的影响：



α 是一个非负的调节参数，可以看到：当 $\alpha = 0$时，此时它与基础线性回归的损失函数一致，没有起到任何惩罚作用；当$\alpha \to \infty$时，它的惩罚项也就是无穷大，而为了使代价函数最小，只能压缩系数 $\boldsymbol{w}$ 趋近于0。

L2的优越性并不主要体现在让参数变小上，关键是在于让所有的**参数比较均衡**。也就是说所有的特征的表达能力都差不多，均等的对待每一个特征。这样就不至于让模型对某个特征特别敏感，也就是说在测试集上运行的时候，即使某个特征上有噪声异常突出，但对于整体模型的输出而言，并不会被这个噪声带偏特别多。

但是因为 α 不可能为无穷大，二次项求偏导时总会保留变量本身，所以事实上它也不可能真正地将某个特征压缩为 0。尽管**系数较小可以有效减小方差**，但依然留着一大长串特征会使模型不便于解释。这是岭回归的缺点，而 `LASSO` 回归可以解决这个问题。

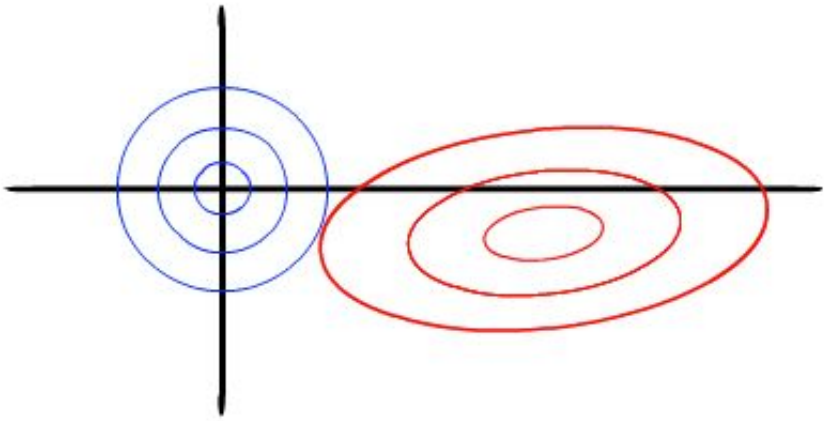
岭回归的几何意义

（B2）式在数学上可以证明与下面的式子是等价的：

$$\begin{aligned} f(\omega) &= \sum_{i=1}^{m}(y_i - x_i^T\omega)^2 \\ s.t.\ &\sum_{i=1}^{n}\omega_i^2 \le t \end{aligned} \tag{B5}$$

其中 t 是阈值。

以两个变量为例, 残差平方和可以表示为 w_1, w_2 的一个二次函数，是一个在三维空间中的抛物面，可以用等值线来表示。而限制条件 $w_1^2 + w_2^2 < t$， 相当于在二维平面的一个圆。这个时候等值线与圆相切的点便是在约束条件下的最优点，如下图所示:



LASSO 回归

岭回归限定了所有回归系数的平方和不大于 t , 在使用普通最小二乘法回归的时候当两个变量具有相关性的时候，可能会使得其中一个系数是个很大正数，另一个系数是很大的负数。通过岭回归的 $\sum_{i=1}^{n} w_i^2 \leq t$ 的限制，可以避免这个问题。

`LASSO`(Least Absolute Shrinkage and Selectionator Operator 最小绝对值收敛和选择算子、套索算法) 回归的正项则就把二次项改成了一次绝对值（L1范数），具体为：

$$f(\omega) = \sum_{i=1}^{m}(y_i - x_i^T\omega)^2 + \alpha\sum_{i=1}^{n}|\omega_i| \tag{C1}$$

或者

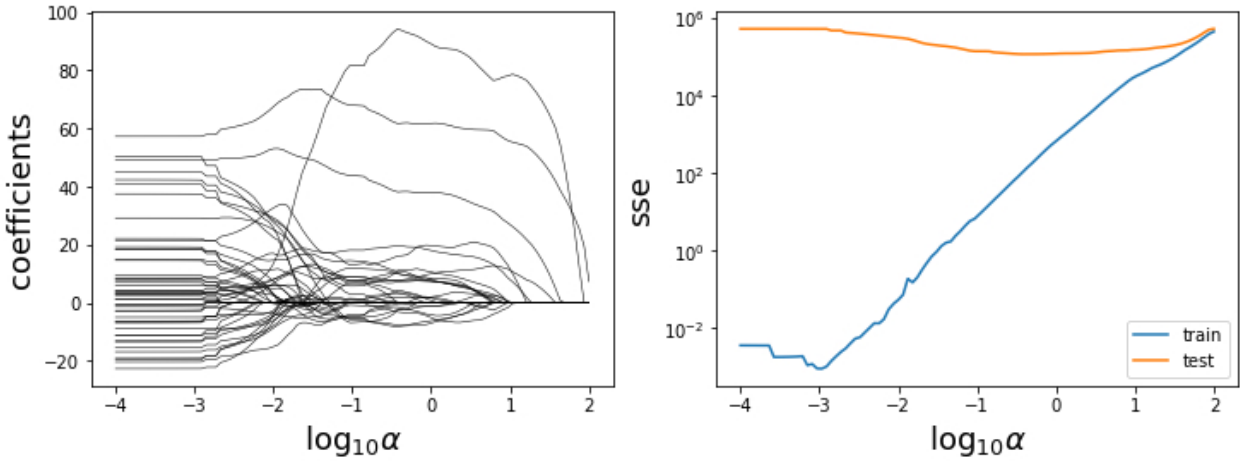
$$\min_{\omega} {||\boldsymbol{X}\boldsymbol{\omega} - \boldsymbol{y}||_2}^2 + \alpha||\boldsymbol{\omega}||_1 \tag{C2}$$

一次项求导可以抹去变量本身，因此lasso回归的系数可以为0。这样可以起来真正的特征筛选效果。无论对于岭回归还是 `LASSO` 回归，本质都是通过调节 α 来实现模型偏差 vs 方差的平衡调整。

`LASSO` 对于数据的要求是极其低的，所以应用程度较广；除此之外，`LASSO` 还能够对特征进行筛选和对模型的复杂程度进行降低:

- 这里的特征筛选是指不把所有的特征都放入模型中进行拟合，而是有选择的把特征放入模型从而得到更好的性能参数。
- 复杂度调整是指通过一系列参数控制模型的复杂度（值不为 0 的参数的个数），得到的解更稀疏，从而避免过度拟合(Overfitting)，也使得模型更具有解释性。

对于线性模型来说，复杂度与模型的变量数有直接关系，变量数越多，模型复杂度就越高。 更多的变量在拟合时往往可以给出一个看似更好的模型，但是同时也面临过度拟合的危险。



可以看到 α 很小时，大部分系数不为零，随着 α 值的增加系数为零项越来越多。同时在损失图中，随着 α 的增加 测试 损失先是下降让后增加，因此不是 α 越大越好，我们需要找到最合适的 α 值。

几何意义

【参考】

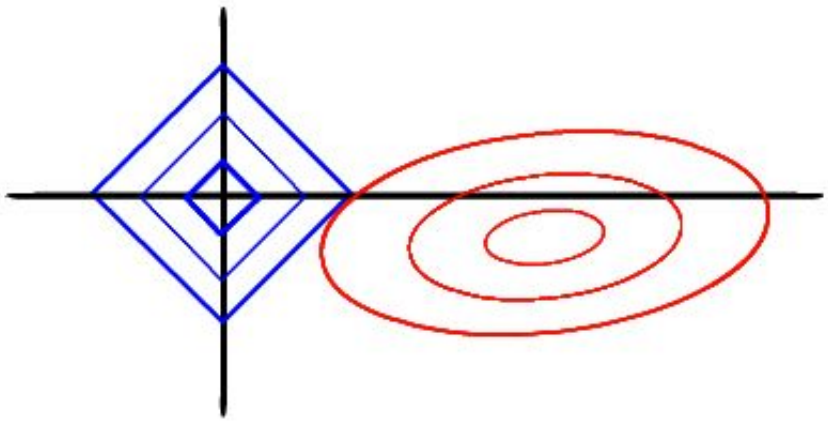
- sklearn - Elastic Net

（C2）式在数学上可以证明与下面的式子是等价的：

$$
\begin{aligned}
&f(\omega) = \sum_{i=1}^{m}(y_i - x_i^T\omega)^2 \\
&s.t. \sum_{i=1}^{n}|\omega_i| \leq t
\end{aligned}
\tag{C3}
$$

其中 t 是阈值。

同样以两个变量为例，标准线性回归的损失函数还是可以用二维平面的等值线表示，而约束条件则与岭回归的圆不同，LASSO的约束条件可以用方形表示，如下图:

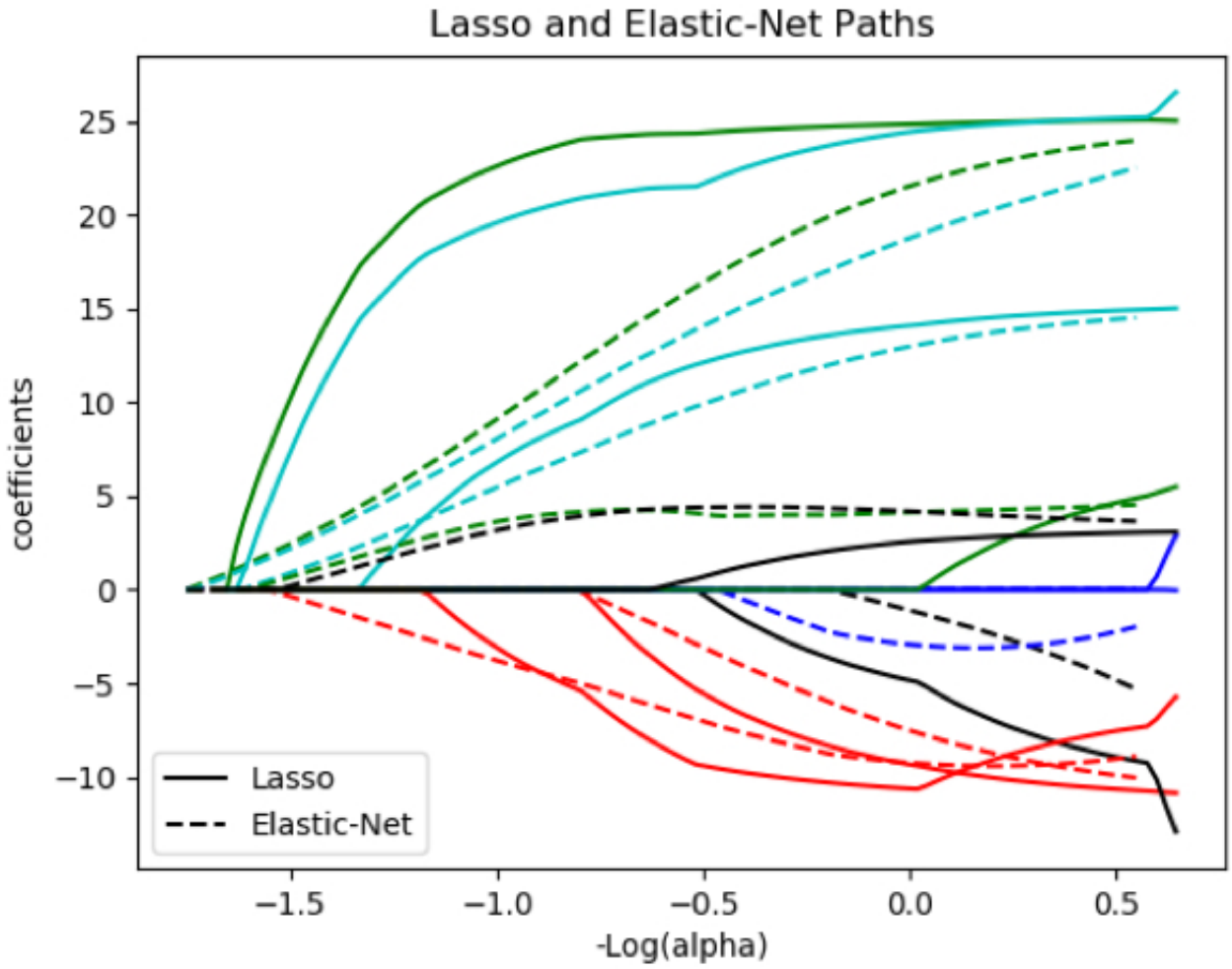


相比圆，方形的顶点更容易与抛物面相交，顶点就意味着对应的很多系数为0，而岭回归中的圆上的任意一点都很容易与抛物面相交很难得到正好等于0的系数。这也就意味着，`LASSO` 起到了很好的筛选变量的作用。

但是，如果存在一组高度相关的变量时，即多个特征与另一个特征相关，Lasso倾向于随机的从中选择一个变量，而忽视其他所有的变量，这样可能会导致结果的不稳定性。这时可以尝试使用 `elastic net penalty`。`ElasticNet` 是结合了 L1 与 L2，即保留了 L1 的稀疏解特性，也维护了 L2 正则化特性。`ElasticNet` 的最小化目标函数为：

$$\min_{w} \frac{1}{2n_{samples}} ||Xw - y||_2^2 + \alpha\rho||w||_1 + \frac{\alpha(1-\rho)}{2}||w||_2^2 \qquad (C6)$$

`LASSO` 与 `ElasticNet` 对比：



线性判别（ LDA）

即 Linear Discriminant Analysis