线性回归

笔记下载修改版本链接：https://drive.google.com/file/d/1ixZ3ugeJtFIdhy_F4yEM6hBAo33ZF4DN/view?usp=share_link

1. 什么是回归算法

- 回归算法是一种有监督算法
- 建立“解释”变量(自变量X)和观测值(因变量Y)之间的关系
- 从机器学习的角度来讲，用于构建一个算法模型(函数)来做属性(X)与标签(Y)之间的映射关系，在算法的学习过程中，试图寻找一个函数h: R^d->R，使得参数之间的关系拟合性最好。
- 回归算法中算法(函数)的最终结果是一个连续的数据值，输入值(属性值)是一个d维度的属性/数值向量

2. 线性回归

> 作用：连续值的预测
>
> 最优模型: 最优模型也就是所有样本(训练数据)离模型的直线或者平面距离最小
>
> 线性关系: 特征属性*X*和目标属性*Y*之间的关系是满足线性关系

$$
\begin{aligned}
h_\theta(x) &= \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n \\
&= \theta_0 1 + \theta_1 x_1 + \cdots + \theta_n x_n \\
&= \theta_0 x_0 + \theta_1 x_1 + \cdots + \theta_n x_n \\
&= \sum_{i=0}^{n} \theta_i x_i = \theta^T x
\end{aligned} \tag{20}
$$

- 目标属性h(x)，x代表特征值，x前面的代表参数，θ要求解的。求出后就可以确定h(x)
- θ(T)：(1, n), x: (n, 1)，等号右边是一个标量
- 机器学习中通常采用列向量为基本向量，所以需要要把θ转置为行向量

3. 最小二乘法

> 计算预测值和实际值的差值的平方然后求出这个值的最小值对应的参数, 就是我们要的模型

$$J(\theta)=\frac{1}{2}\sum_{i=1}^{m}\left(\varepsilon^{(i)}\right)^{2}=\frac{1}{2}\sum_{i=1}^{m}\left(h_{\theta}\left(x^{(i)}\right)-y^{(i)}\right)^{2} \tag{21}$$

- 差值有正有反，会互相抵消，用平方来避免。ε代表差值。其中1/2是为了后面方便求导，不会对所求产生影响
- 求出这个差值函数的最小值时的参数值得到模型
- 房价预测

> 二元二次函数， 凸函数， 求极小值得出结果。n元n次超平面

4. 最大似然估计

> 解释最大似然估计*(maximum likelihood estimation，MLE)*：估计参数的方式，投掷硬币，独立事件，同时发生的概率， 即每个事件发生概率相乘，就是联合概率，联合概率越大越好，关于参数p的似然函数，极大化，取对数，求最大值

4.1. 正态分布

- 理想误差$\varepsilon^{(i)}(1\leq i\leq n)$，独立同分布的，服从均值为0，方差为某$\theta^2$定值的高斯分布
- 随机现象可以看做众多因素的独立影响的综合反应，往往服从正态分布，误差出现的概率：

$$f(x)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^{2}}{2\sigma^{2}}\right) \tag{22}$$

- 原因：中心极限定理，解释了为什么服从正态分布

设随机变量X_1，X_2，……X_n，……独立同分布，并且具有有限的数学期望和方差：$E(X_i)=\mu$，$D(X_i)=\sigma^2 0(i=1,2\ldots)$，则对任意x，分布函数

$$F_n(x)=P\left\{\frac{\sum_{i=1}^{n}X_i-n\mu}{\sigma\sqrt{n}}\leq x\right\}$$

满足

$$\lim_{n\to\infty}F_n(x)=\lim_{n\to\infty}P\left\{\frac{\sum_{i=1}^{n}X_i-n\mu}{\sqrt{n}\sigma}\leq x\right\}=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x}e^{-\frac{t^2}{2}}dt=\varnothing(x)$$

该定理说明，当n很大时，随机变量$Y_n=\frac{\sum_{i=1}^{n}X_i-n\mu}{\sqrt{n}\sigma}$近似地服从标准正态分布N(0，1)。因此，当n很大时，$\sum_{i=1}^{n}X_i=\sqrt{n}\sigma Y_n+n\mu$近似地服从正态分布N(nμ，nσ^2)。该定理是中心极限定理最简单又最常用的一种形式，在实际工作中，只要n足够大，便可以把独立同分布的随机变量之和当作正态变量。这种方法在数理统计中用得很普遍，当处理大样本时，它显重要工具。 [2]

4.2. 似然函数

$$y^{(i)}=\theta^{T}x^{(i)}+\varepsilon^{(i)}\quad \text{第i组数据下的关系} \tag{23}$$

$$p\left(\varepsilon^{(i)}\right)=\frac{1}{\sigma\sqrt{2\pi}}e^{\left(-\frac{\left(\varepsilon^{(i)}\right)^{2}}{2\sigma^{2}}\right)}\quad \text{误差符合正态分布-概率越大越靠近均值-预测就越准确} \tag{24}$$

$$\frac{1}{\left(-\frac{\left(y^{(i)}-\theta^{T}x^{(i)}\right)^{2}}{2\sigma^{2}}\right)}$$

$$xp\left(y^{(i)} \mid x^{(i)};\theta\right)=\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(y^{(i)}-\theta^{T}x^{(i)})^{2}}{2\sigma^{2}}\right)$$ 结合（4）与（5）得到 (25)

> 此式表示，在θ和$x^{(i)}$下，$y^{(i)}$符合程度，即概率

$$\begin{aligned}L(\theta)&=\prod_{i=1}^{m}p\left(y^{(i)} \mid x^{(i)};\theta\right)\\&=\prod_{i=1}^{m}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{\left(y^{(i)}-\theta^{T}x^{(i)}\right)^{2}}{2\sigma^{2}}\right)\end{aligned}$$ θ下的似然估计-联合概率-似然函数 (26)

> $\ln L(\theta)$要极大化似然函数，值越大说明越符合模型，即可得到θ

4.2.1. 似然函数取对数

> 目的:（4）式需要处理才可方便计算，一般都是取对数，结果如下：

$$\begin{aligned}&\ell(\theta)=\ln L(\theta)\\&=\ln\textstyle\prod_{i=1}^{m}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{\left(y^{(i)}-\theta^{T}x^{(i)}\right)^{2}}{2\sigma^{2}}\right)\\&=\textstyle\sum_{i=1}^{m}\ln\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{\left(y^{(i)}-\theta^{T}x^{(i)}\right)^{2}}{2\sigma^{2}}\right)\\&=m\ln\frac{1}{\sigma\sqrt{2\pi}}-\frac{1}{\sigma^{2}}\bullet\frac{1}{2}\textstyle\sum_{i=1}^{m}\left(y^{(i)}-\theta^{T}x^{(i)}\right)^{2}\\&\operatorname{loss}\left(y_j,\hat{y}_j\right)=J(\theta)=\frac{1}{2}\textstyle\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)^{2}\end{aligned}$$ 和最小二乘法式（2）做对比--结果一致 (2

- 得到$J(\theta)$目标函数，极大化似然函数转化求$J(\theta)$的最小值， 如下

5. θ的求解过程

$$J(\theta)=\frac{1}{2}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)^{2}$$ 目标函数-待处理函数 (28)

$$\begin{aligned}&J(\theta)=\tfrac{1}{2}\textstyle\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)^{2}=\tfrac{1}{2}(X\theta-Y)^{T}(X\theta-Y)\to\min_\theta J(\theta)\\&\nabla_\theta J(\theta)=\nabla_\theta\left(\tfrac{1}{2}(X\theta-Y)^{T}(X\theta-Y)\right)=\nabla_\theta\left(\tfrac{1}{2}\left(\theta^{T}X^{T}-Y^{T}\right)(X\theta-Y)\right)\\&=\nabla_\theta\left(\tfrac{1}{2}\left(\theta^{T}X^{T}X\theta-\theta^{T}X^{T}Y-Y^{T}X\theta+Y^{T}Y\right)\right)\\&=\tfrac{1}{2}\left(2X^{T}X\theta-X^{T}Y-\left(Y^{T}X\right)^{T}\right)\\&=X^{T}X\theta-X^{T}Y\\&\theta=\left(X^{T}X\right)^{-1}X^{T}Y\end{aligned}$$ 求偏导-矩阵对向

> 对$J(\theta)$求偏导，就是求梯度，梯度意味着是对θ内每一个参数求导了，求导公式如下：
>
> $$\begin{aligned}\tfrac{\partial A\cdot x}{\partial x}&=A^{T}\quad\tfrac{\partial x\cdot A}{\partial x}=A\\\tfrac{\partial A\cdot x}{\partial x^{T}}&=A\quad\tfrac{\partial x\cdot A}{\partial x}=A^{T}\\\tfrac{\partial \mathbf{x^{T}x}}{\partial \mathbf{x}}&=2\mathbf{x}\\\tfrac{\partial \mathbf{x^{T}Ax}}{\partial \mathbf{x}}&=(\mathbf{A}+\mathbf{A^{T}})\mathbf{x}\end{aligned}$$ (30)

$\partial \mathbf{x}$ $= (\mathbf{A} + \mathbf{A}^T)\mathbf{x}$

> 推导：
>
> $$\frac{1}{2}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)^2=\frac{1}{2}(X\theta-Y)^T(X\theta-Y) \tag{31}$$
>
> *1.* 由*(4)*和*(9)*得*:*
>
> $$J(\theta)=\frac{1}{2}\sum_{i=1}^{m}\left(\varepsilon^{(i)}\right)^2 \tag{32}$$
>
> > 变换成矩阵形式：
> >
> > $$\begin{pmatrix}\varepsilon^{(1)}\\ \varepsilon^{(2)}\\ \cdots\\ \varepsilon^{(n)}\end{pmatrix}\begin{pmatrix}\varepsilon^{(1)} & \varepsilon^{(2)} & \cdots & \varepsilon^{(n)}\end{pmatrix} \tag{33}$$
>
> *2.* $$\frac{1}{2}(X\theta-Y)^T(X\theta-Y)\rightarrow(X\theta-Y)^T \tag{34}$$
>
> > X（*m*n*）表示*m*个数据，*n*个特征
>
> > Y（*m*1*）表示*m*个数据预测的结果
> >
> > θ （*n*1*）表示特征参数
> >
> > 因此$X\theta-Y$为（*m*1*），观察比较*(11)*可得等式

6. 最小二乘法θ参数最优解

> 作用：最小二乘法的使用要求矩阵是可逆的；为了防止不可逆或者过拟合的问题存在，可以增加额外数据影响，导致最终的矩阵是可逆的
>
> 作用原理？

- 表达式：

$$\theta=\left(X^TX+\lambda I\right)^{-1}X^Ty \tag{35}$$

7. 多项式扩展

> 产生：基于现有数据构造出新的数据

> 目的：解决欠拟合问题， 不易一定能解决；多一点特征向量，模型越复杂；模型越复杂，就可能使结果更准确；扩展越多， 过拟合， 模型复杂， 利用交叉验证来减少

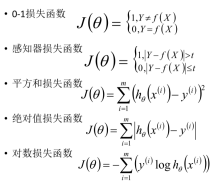
- $\begin{pmatrix} x^{(1)} & x^{(2)} & x^{(3)} \end{pmatrix} \rightarrow \begin{pmatrix} x^{(1)} & x^{(2)} & x^{(3)} & x^{(1)}x^{(2)} & x^{(2)}x^{(3)} & x^{(3)}x^{(1)} \end{pmatrix}$

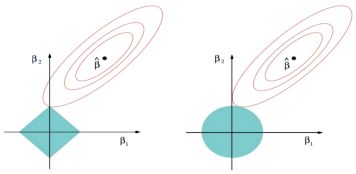
  > 扩展原则：三项以上的交互项不可出现元素重复

8. 线性回归的过拟合

> 预测结果由参数θ决定， 可能出现过大与过小的情况， 主要是因为太大了
>
> 过拟合：训练集上的效果好，在测试集评估效果不好--模型过于复杂，数据冗余特征多--无效特征多--数据量少，把背景学习进去了，影响有用的特征
>
> > 解决：增加数据集和去除冗余特征，加入惩罚项（正则项）
>
> 欠拟合：训练的评估指标不好--模型过于简单--数据量不够与特征不够好，特征不够好：不是有效的特征和特征数据不好--没有处理相关特征
>
> > 解决：从处理特征层面，增加模型复杂度，不是扩展的阶数越多越好，容易导致过拟合，多项式扩展

- 正则项(norm)/惩罚项

  > 目的：为了防止数据过拟合，也就是的θ值在样本空间中不能过大，可以在目标函数之上增加一个平方和损失
  >
  > > 损失函数：
  > >
  > > 
  > >
  > >
  > > 对数损失和*0-1*用于分类，其余用于回归问题
  >
  > 惩罚项：
  >
  > $$\lambda\sum_{j=1}^{n}\theta_j^2 \tag{36}$$
  >
  > 加入惩罚项表达式：
  >
  > $$J(\theta)=\frac{1}{2}\sum_{i=1}^{n}\left(h_\theta(x^{(i)})-y^{(i)}\right)^2+\lambda\sum_{i=1}^{n}\theta_j^2 \tag{37}$$
  >
  > 对于惩罚项的理解：
  >
  > > 为了防止过拟合问题，就转化成了，有约束条件的函数优化转化成无约束体条件的优化问题



注释：左：*Ridge(L2-norm)* 右：*LASSO(L1-norm)*

如右图所示为半径为*t*的圆，圆内是约束条件（惩罚项）下的特征值，圆越小代表约束越大；右上方是红圈表示初始目标函数的等高线，由抛物面投影到平面得到，越靠近圆心表示预测值越接近真实值；等值线越小越好，但惩罚项的作用也限定了范围，所以在像相切除为最优化处；相切除为最优化处，目标函数的梯度和约束条件梯度相反，大小不同，不同量纲的两个数，配平梯度长度，所以要加入λ，使得两者大小相同

- *Ridge(L2-norm)*和*LASSO(L1-norm)*比较

  > *LASSO*：*Least Absolute Shrinkage andSelection Operator*
  >
  > *Ridge(L2-norm)*(岭回归)：具有较高的求解速度；不可能导致有维度参数变为*0*的情况，那么也就不会产生稀疏解；数据的维度中是存在噪音和冗余的
  >
  > *LASSO(L1-norm)*：*Ridge*模型具有较高的准确性、鲁棒性以及稳定性*(*冗余特征已经被删除了*)*；稀疏的解可以找到有用的维度并且减少冗余，提高后续算法预测的准确性和鲁棒性

  - 如果既要考虑稳定性也考虑求解的速度，就使用*Elasitc Net*

    > $$I(\theta)=\frac{1}{2}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)^2+\lambda\left(p\sum_{j=1}^{n}|\theta_j|+(1-p)\sum_{j=1}^{n}\theta_j^2\right)$$
    >
    > $$\begin{cases}\lambda>0\\ p\in[0,1]\end{cases}$$

9. 模型效果判断

- MSE：误差平方和，越趋近于0表示模型越拟合训练数据。
- RMSE：MSE的平方根，作用同MSE
- R^2：取值范围(负无穷,1]，值越大表示模型越拟合训练数据；最优解是1；当模型预测为随机值的时候，有可能为负；若预测值恒为样本期望，R^2为0
- TSS：总平方和TSS(Total Sum of Squares)，表示样本之间的差异情况，是伪方差的m倍
- RSS：残差平方和RSS（Residual Sum of Squares），表示预测值和样本值之间的差异情况，是MSE的m倍

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

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

$$R^2 = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2} \qquad \bar{y} = \frac{1}{m}\sum_{i=1}^{m} y_i$$

10. 机器学习调参

> 算法模型*(*线性回归*)*来讲，我们需要获取θ、λ、p的值；θ的求解其实就是算法模型的求解，一般不需要开发人员参与*(*算法已经实现*)*，主要需要求解的是λ和p的值，这个过程就叫做调参*(*超参*)*
>
> 目的：找到一组超参数（均值）

- 交叉验证：将训练数据分为多份，其中一份进行数据验证并获取最优的超参：λ和p训练（返回训练）-验证8000（分5等分）-测试（最后）
- 多则交叉验证可以获得多组，更稳定，进行求均值，比如：十折交叉验证、五折交叉验证(scikit-learn中默认)等；

  > 多则交叉成本会增加