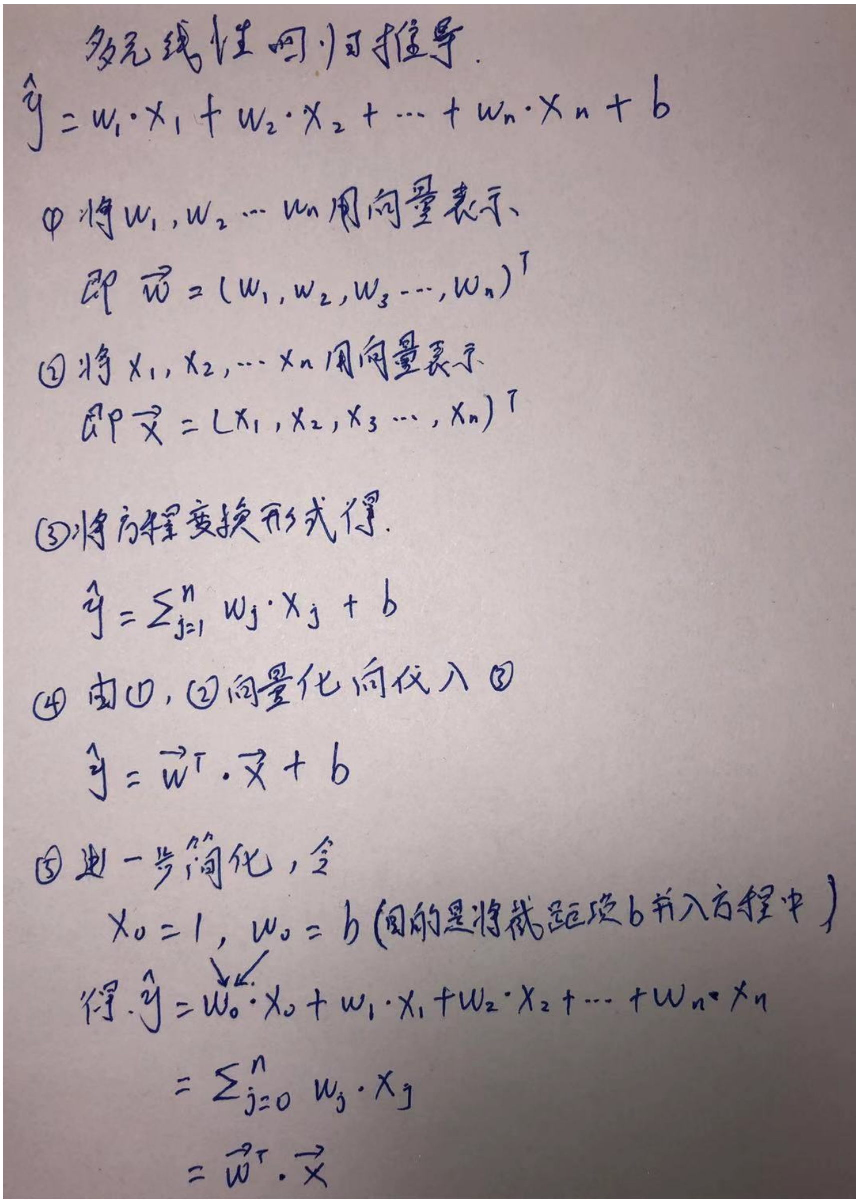
线性回归

1. 多远线性回归的向量表示法

1.1 多远线性回归的向量表示推导



多元线性回归推导.

$\hat{y} = w_1 \cdot x_1 + w_2 \cdot x_2 + \cdots + w_n \cdot x_n + b$

① 将 $w_1, w_2 \cdots w_n$ 用向量表示、

即 $\vec{w} = (w_1, w_2, w_3 \cdots, w_n)^T$

② 将 $x_1, x_2, \cdots x_n$ 用向量表示

即 $\vec{x} = (x_1, x_2, x_3 \cdots, x_n)^T$

③ 将方程变换形式得.

$\hat{y} = \sum_{j=1}^{n} w_j \cdot x_j + b$

④ 由①,②向量化向代入③

$\hat{y} = \vec{w}^T \cdot \vec{x} + b$

⑤ 进一步简化，令

$x_0 = 1, \ w_0 = b$ (目的是将截距项b并入方程中)

得 $\hat{y} = w_0 \cdot x_0 + w_1 \cdot x_1 + w_2 \cdot x_2 + \cdots + w_n \cdot x_n$

$= \sum_{j=0}^{n} w_j \cdot x_j$

$= \vec{w}^T \cdot \vec{x}$

2. 损失函数的由来和求解

2.1 损失函数的由来

极大似然推导.

表示预测值

背景：线性回归模型为 $\hat{y}=\theta X$ → 可参考线性回归模型推导.

误差总和为 $|\theta^T X - y_i|$，误差是满足高斯分布的.

① 则由误差和高斯分布公式可得.

似然函数.　已知量　确定值　自变量.

$$L(\theta)=\prod_{i=1}^{m} P(y^{(i)}|x^{(i)};\theta)=\prod_{i=1}^{m}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)$$

② 对似然函数取对数. 累加更易计算

$$\log L(\theta)=\log\prod_{i=1}^{m}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)$$

$$\Downarrow$$

$$=\sum_{i=1}^{m}\log\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)$$

$$\Downarrow$$

$$=m\log\frac{1}{\sqrt{2\pi\sigma^2}}-\frac{1}{\sigma^2}\cdot\frac{1}{2}\sum_{i=1}^{m}(y^{(i)}-\theta^T x^{(i)})^2$$

预测值 $\hat{y}$

$$\mathrm{loss}(y_j,\hat{y}_j)=J(\theta)=\frac{1}{2}\sum_{i=1}^{m}\left(h_\theta(x^{(i)})-y^{(i)}\right)^2$$

上述阐述了如何由高斯分布推导出损失函数.

损失函数的求解

损失函数求导.

矩阵转置与原矩阵相乘一定可以相乘.

$$J(\theta)=\frac{1}{2}\sum_{i=1}^{m}(y-y_i)^2=(X\theta-y_i)^2=(X\theta-y_i)^T(X\theta-y_i)$$

$$J(\theta)=X^T\theta^T X\theta-X^T\theta^T y_i-y_i^T X\theta+y_i^T y_i$$

$$\frac{\partial J(\theta)}{\partial\theta}=\frac{\partial X^T\cdot\theta^T X\theta}{\partial\theta}-\frac{\partial X^T y_i\theta^T}{\partial\theta}-\frac{y_i^T X\theta}{\partial\theta}+0$$

$$\frac{\partial J\theta}{\partial\theta}=(X^T\cdot X+XX^T)\theta-X^Ty_i-y_iX^T$$

$$=2X^TX\theta-2X^Ty_i$$

$$=2(X^TX\theta-X^Ty_i)$$

令 $\frac{\partial J\theta}{\partial\theta}=0$，即

$$2(X^TX\theta-X^Ty_i)=0$$

$$\theta=\frac{X^Ty_i}{X^TX}$$

3. 回归模型的评估

3.1　MSE(mean squared error)：平均平方误差

原理：所有样本数据误差（真实值与预测值之差）的平方和，然后取均值

公式如下

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

3.2 RMSE(root squareed error)：平均平方误差的平方根

原理：在 MSE 的基础上开根号

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

3.3 MAE(mean absolute error)：平均绝对值误差，为所有样本数据误差的绝对值和

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y^{(i)} - \hat{y}^{(i)}|$$

3.4 R^2：决定系数，用来表示模型拟合的好坏，值越高拟合越好，在训练集中，R^2 的取值范围为[0,1]，在测试集(未知数据)中，R^2 的取值范围是[-∞，1]，

3.5 各个评估标准的使用方法

```python
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score

print("均方误差(MSE)：", mean_squared_error(y_test, y_hat))
print("根均方误差(RMSE)：", np.sqrt(mean_squared_error(y_test, y_hat)))
print("平均绝对值误差(MAE)：", mean_absolute_error(y_test, y_hat))
print("训练集R^2：", r2_score(y_train, lr.predict(X_train)))
print("测试集R^2：", r2_score(y_test, y_hat))
# socre其实求解的就是r^2的值。但是注意，r2_score方法与score方法传递参数的内容是不同的。
print("训练集R^2：", lr.score(X_train, y_train))
print("测试集R^2：", lr.score(X_test, y_test))
```