# class 6

## 1 提出和分析问题

建立一个数学模型（机器学习）根据房屋特征预测房价。要求预测的准确度要超过70%

## 2 声明假设

- 数据准确可靠；
- 可以通过数据推出普遍规律；
- 房价仅受到数据集所包含因素影响，其他因素忽略不计；
- 售价与因素的关系是线性的，

## 3 说明符号

- $Price$ ： 房屋价格
- $Condition$ :房屋的条件
- $Bedrooms$ :卧室数量
- $Bathrooms$ :厕所数量
- $View$ ： 景观分值
- $Waterfront$

## 4 建立模型

根据假设，各个因素与房屋价格呈正相关，故建立如下预测模型

$$Price = w_1Condition + w_2Bedrooms + w_3Bathrooms + w_4View + w_5Waterfront + w_0$$

$w_i$ 代表上述变量的系数以及常数项。

## 5 求解模型

通过最小二乘法（求解线性模型系数的方式）计算得到模型的系数如下：

```
系数值 [ 43682.97326379  21487.69378766 215043.546058   119753.33157832
 578086.36834578]
常数项 -168460.80577590293
```

## 6 分析和检验模型

## 7 反思和提升

## 作业

- 使用https://reformship.github.io/pages/44datasets.html 界面的 贷款审批(load_loan_approvals)数据集，建立模型，利用除贷款审批是否通过（最后一列）的其他变量预测某人是否能通过贷款审批
- 将建模过程按照七步建模法写在word文档中

在下节课上课前（10月30日之前）将作业发送到learningmm@163.com 作业主题为“宋校+姓名+第6次课作业”。

## 附录

代码

```
# 引入相应的库
import pandas as pd
import matplotlib.pyplot as plt
# 引入数据
data = pd.read_csv('load_kc_house_data.csv')
data.columns
features = ['condition','bedrooms','bathrooms','view','waterfront']
X = data[features] # 特征数据（自变量）
y = data['price'] # 预测数据（目标变量或者因变量）

# 引入线性回归模型
from sklearn.linear_model import LinearRegression
# 初始化模型
model = LinearRegression()
# 训练模型
model.fit(X,y)
# 获得结果
# 获得变量系数
print('系数值',model.coef_)
# 获得常数项值
print('常数项',model.intercept_)
```

散点图

```
data_new = data[data['price']<=1e6]
features3 = ['bedrooms', 'bathrooms', 'sqft_living',
       'sqft_lot', 'floors', 'waterfront', 'view', 'condition', 'grade',
       'sqft_above', 'sqft_basement', 'yr_built', 'yr_renovated',
             'sqft_living15', 'sqft_lot15']
for feature in features3:
    plt.scatter(data_new[feature],data_new['price'])
    plt.xlabel(feature)
    plt.show()
```

神经网络

```

data_new = data[data['price']<=1e6]
data_new2 = data_new.copy()
data_new2['grade2'] = data_new['grade']**2
features4 = ['bedrooms', 'bathrooms', 'sqft_living',
       'sqft_lot', 'floors', 'waterfront', 'view', 'condition', 'grade',
       'sqft_above', 'sqft_basement', 'yr_built', 'yr_renovated', 'zipcode',
       'lat', 'long', 'sqft_living15', 'sqft_lot15','grade2']

X = data_new2[features4] # 特征数据(自变量)
y = data_new2['price'] # 预测数据(目标变量或者因变量)

# 引入神经网络模型
from sklearn.neural_network import MLPRegressor
# 初始化模型
model = MLPRegressor()
# 训练模型
model.fit(X,y)
# 获得结果
# 获得变量系数
# 评估模型的准确度
score = model.score(X,y)
print('模型r2值',score)
```

```
data_new = data[data['price']<=1e6]
data_new2 = data_new.copy()
data_new2['grade2'] = data_new['grade']**2
features4 = ['bedrooms', 'bathrooms', 'sqft_living',
       'sqft_lot', 'floors', 'waterfront', 'view', 'condition', 'grade',
       'sqft_above', 'sqft_basement', 'yr_built', 'yr_renovated', 'zipcode',
       'lat', 'long', 'sqft_living15', 'sqft_lot15','grade2']

X = data_new2[features4] # 特征数据(自变量)
y = data_new2['price'] # 预测数据(目标变量或者因变量)

# 引入支持向量机
from sklearn.svm import SVR
# 初始化模型
model = SVR()
# 训练模型
model.fit(X,y)
# 获得结果
```