**基于多元线性回归的北京市二手房价格预测**

张书宁 北京市第四中学

# 引言

# 1.数据选取和处理

## 1.1数据来源

本文的全部数据来源于链家房地产网站（lianjia.com）的二手房数据，通过爬虫和调用腾讯地图API的方式获取样本数据。选取10项显著体现二手房特征的指标和价格，具体见下表。

# 数据预处理

# ROOM 房间数

# FLOR 0-低楼层 1-中楼层 2-高楼层

# AREA 面积

# TYPE 0-塔楼 1-板塔结合 2-板楼

# EVAL 0-无电梯 1-有电梯

# PRIC 每平米价格

# DISC 0-朝阳 1-东城 2-房山 3-海淀 4-通州 5-西城

# OLD 建造时间

# SUBC 周围1000m地铁站个数

# SUBD 最近地铁站距离

# CEDT 距市中心距离

|  |  |  |
| --- | --- | --- |
| 指标名称 | 具体说明 | 取值范围 |
| ROOM | 房间数 | 正整数 |
| FLOR | 楼层高度 | 0，1，2 |
| AREA | 面积 | 正数 |
| TYPE | 建筑结构 | 0，1，2 |
| EVAL | 是否有电梯 | 0，1 |
| DISC | 行政区 | 0，1，2，3，4，5 |
| YEAR | 建造年代 | 四位正整数 |
| SUBC | 周围1000m地铁站数量 | 正整数 |
| SUBD | 最近地铁站距离 | 正数 |
| CEDT | 距市中心距离 | 正数 |

## 1.2数据处理

爬虫下载数据后，需要进行数据的清洗和整合处理。首先删除部分未使用的多余指标键值，并且将部分网站上的源数据处理成量化数据。具体代码如下：

*#处理键值数据*

*#删除多余项*

def process(*handle*, *district*) -> list:

    d = json.load(*handle*)

*for* kv *in* d:

*del* kv['INTRO']

*del* kv['ROUND']

        kv['AREA'] = kv['AREA'][:-1]

*if* "南" *in* kv['FACE']:

            kv['FACE'] = 1

*else*:

            kv['FACE'] = 0

        kv['DISC'] = *district*

*del* kv['FACE']

*del* kv['STRT']

*del* kv['FITM']

*del* kv['WARM']

        kv['OLD'] = kv['OLD'][0:kv['OLD'].find("年建")]

*return* d

并且将爬虫处理的JSON格式数据加载为DataFrame后转换成CSV格式。

# 2.建立多元回归线性方程和模型

## 2.1多元线性回归方程