

**实 验 报** **告**

**实验名称：** **基于Matlab的波士顿房价预测**



班级：智研211



姓名：王利猛



学号：2021540044



教师：徐文星



人工智能研究院

目录

[一、数据集介绍 1](#_Toc99471921)

[1.1数据集可视化： 1](#_Toc99471922)

[二、实验目标 2](#_Toc99471923)

[2.1 BP神经网络拟合 2](#_Toc99471924)

[2.2线性拟合 2](#_Toc99471925)

[2.2.1全特征线性拟合 2](#_Toc99471926)

[2.2.2部分特征线性拟合 3](#_Toc99471927)

[2.2.3变参数线性拟合 3](#_Toc99471928)

[2.3回归问题转为分类问题 3](#_Toc99471929)

[三、参数设置 3](#_Toc99471930)

[3.1BP神经网络拟合 3](#_Toc99471931)

[3.2线性拟合 3](#_Toc99471932)

[3.2.1全特征线性拟合 3](#_Toc99471933)

[3.2.2部分特征线性拟合 3](#_Toc99471934)

[3.2.3变参数线性拟合 4](#_Toc99471935)

[3.3回归问题转为分类问题 4](#_Toc99471936)

[四、结果分析 4](#_Toc99471937)

[4.1 BP神经网络预测结果 4](#_Toc99471938)

[4.2线性拟合结果 5](#_Toc99471939)

[4.2.1全特征线性拟合结果 6](#_Toc99471940)

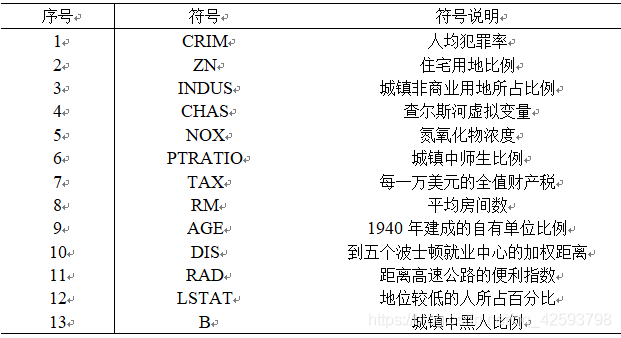
[4.2.2部分特征线性拟合结果 7](#_Toc99471941)

[4.2.3变参数线性拟合结果 7](#_Toc99471942)

[4.3回归问题转为分类问题结果 8](#_Toc99471943)

# 一、数据集介绍

波士顿住房数据是从卡内基梅隆大学维护的StatLib图书馆中获取的数据集。它有506个实例和13个变量，还有一个因变量房价。属性信息如下所示：

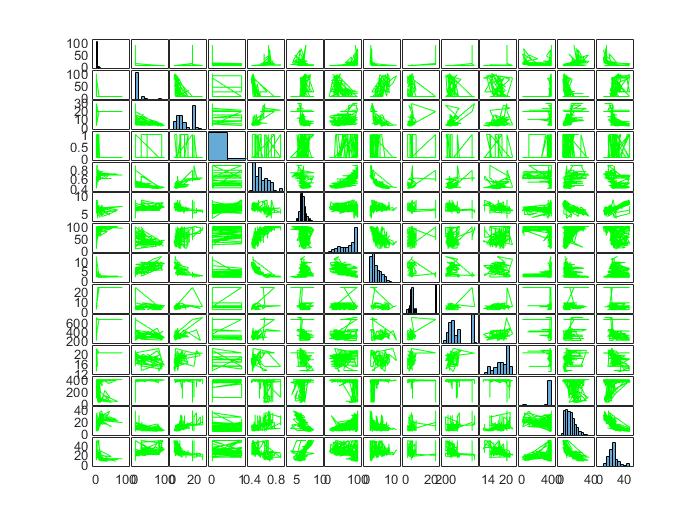


数据集获取地址：<http://archive.ics.uci.edu/ml/machine-learning-databases/housing>

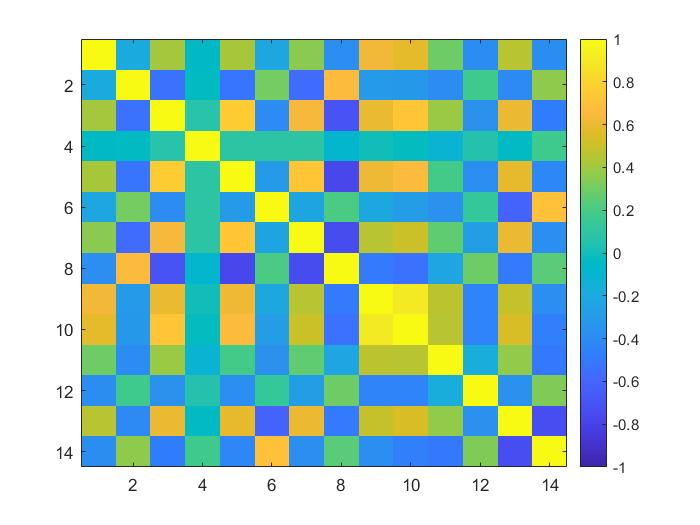
## 1.1数据集可视化：

Matlab代码里，使用plotmatrix函数生成散点图矩阵。使用corrcoef函数来计算14列数据两两之间的相关性。使用imagesc()函数来可视化相关系数。

下图表示根据数据集创建的的散点图矩阵。矩阵的第i行、第j列中的子图（绿色散点图）是数据集的第 i 列相对于数据集的第 j 列的散点图。对角线方向的子图是数据集的每一列的直方图。



下图表示13个变量和房价之间的相关系数（横纵轴数字代表变量和房价的序号，热力图代表相关度。如坐标（1,14）和（14,1）位置都是浅蓝色代表CRIM与房价的相关度等于房价与CRIM的相关度，都是-0.4左右）。



从以上两图中可以看出，房价和RM之间存在正相关，也就是随着房间号变大，房子会变得更贵。此外，房价和LSTA比率之间有负相关，地位较低的人定居在更便宜的街区。这很符合常理。

# 二、实验目标

## 2.1 BP神经网络拟合

按训测比随机抽取训练集测试集，调用Matlab的feedforwardnet函数训练BP神经网络，用sim函数对测试集数据进行预测。结果反归一化得到预测值。输出误差及其均值与方差，预测值与真实值对比图。本文采用两组参数对比实验，参数设置见3.1。

## 2.2线性拟合

数据集划分固定不变。随机选取训练集450条，测试集56条。设所有特征与因变量线性相关。改变以下条件：

### 2.2.1全特征线性拟合

用Matlab的fitlm函数对13个特征和因变量进行线性最小二乘拟合。用feval函数结合线性拟合多项式lm对56个测试集进行线性预测。

### 2.2.2部分特征线性拟合

用Matlab的fitlm函数对挑选出来的10个特征和因变量进行线性最小二乘拟合。用feval函数结合线性拟合多项式lmnew对56个测试集进行线性预测。

### 2.2.3变参数线性拟合

用Matlab的fitlm函数对13个特征和因变量进行线性最小二乘拟合。改变参数”linear”为” quadratic”.通过线性拟合多项式lmQ进行56个测试集预测。

## 2.3回归问题转为分类问题

将房价小于15的房屋分类为1(便宜)，15到30之间的为2(中等)，超过30的为3(昂贵)。将数据集因变量房价由连续值转化为离散值。使得该问题转化为分类问题。调用Matlab的fitcknn函数训练KNN分类器，再利用predict函数进行预测分类。

# 三、参数设置

## 3.1BP神经网络拟合

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 训测比 | 训练轮次 | 收敛误差 | 隐藏层神经元个数 |
| 第一次 | 7:3 | 5000 | 0.0000001% |  |
| 第二次 | 9:1 | 7000 | 0.00000001% |  |

## 3.2线性拟合

### 3.2.1全特征线性拟合

用Matlab的fitlm函数对13个特征和因变量进行线性最小二乘拟合。

### 3.2.2部分特征线性拟合

用Matlab的fitlm函数对挑选出来的10个特征（第1 2 4 5 6 8 10 11 12 13号特征）和因变量进行线性最小二乘拟合。

### 3.2.3变参数线性拟合

改变所得拟合模型包含的参数’linear’为’ quadratic’。

## 3.3回归问题转为分类问题

训练集和测试集输入数据不变（仍然为13个特征归一化结果）。训练集和测试集输出数据改为离散值1 2 3。

# 四、结果分析

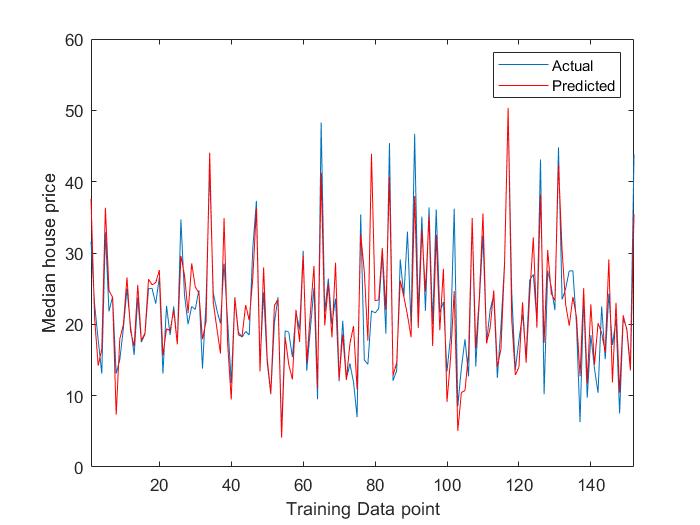
## 4.1 BP神经网络预测结果

第一次参数设置：

误差的均值方差如下：

Mean，Std = -0.1363 4.0725

房价（152组）真实值（蓝）与预测值（红）的图线对比

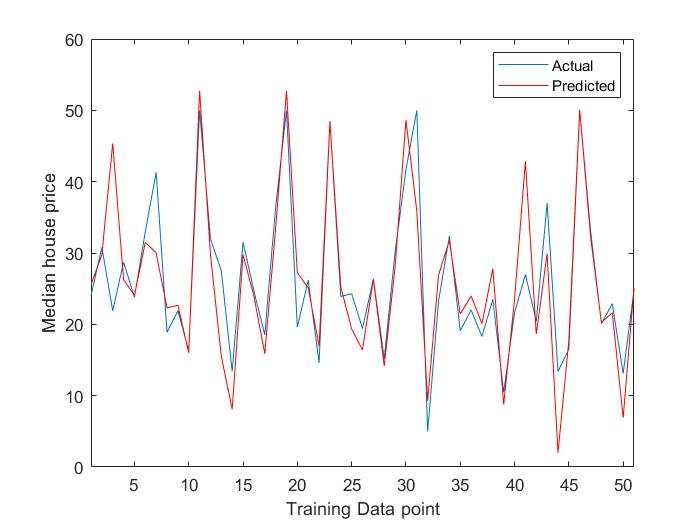


第二次参数设置：

误差的均值方差如下：

Mean，Std = 0.2080 6.0163

房价（152组）真实值（蓝）与预测值（红）的图线对比



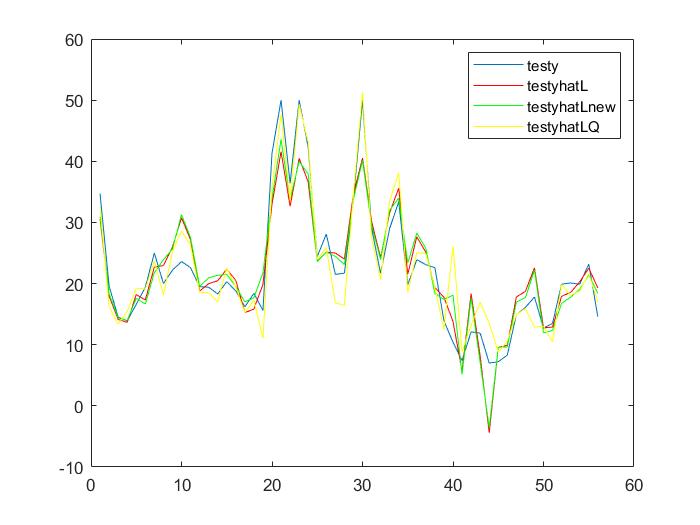
第二种结果方差大，误差大。尽管测试样本少但是预测结果并不好。很可能是因为出现了过拟合导致的。

## 4.2线性拟合结果

如图，蓝线是56个因变量房价testy的折线图；红线是3.2.1模型预测值testyhatL的折线图；绿线是3.2.2模型预测值testyhatLnew的折线图；黄线是3.2.3模型预测值testyhatLQ的折线图。

从图中我们可以直观地看出黄线最贴进蓝线，效果最好；红线最不贴近蓝线，效果最差。这是因为红线要有一个很强的假设就是房价和所有特征都线性相关。很明显第一第二张图已经说明了只是部分特征和房价相关。如RM和LSTAT。所以绿线，也就是剔除了几个相关性不好的特征后再进行线性拟合的结果，要比红线好一点。

当参数为'linear'时模型只包含每个预测变量的截距和线性项。而参数为'quadratic'模型包含每个预测变量的截距项、线性项和平方项，以及不同预测变量对组的所有乘积。所以3.2.3的模型相对3.2.1的模型而言包含更多信息。它的预测效比较好，也就是黄线比红线更贴近蓝线。



### 4.2.1全特征线性拟合结果

左图是用全特征线性拟合的模型lm，它对测试集输入的预测结果就是红线。右图是线性回归模型的各种参数。仅在此节说明，其余小节意义一致，不再赘述。

•Estimate - 模型中每个对应项的系数估计值。例如，常数项 (intercept) 的估计值为 47.977。

•SE - 系数的标准误差。

•tStat - 每个系数的 t 统计量，用于基于对应系数不为零的备择假设来检验对应系数为零的原假设（给出了模型中的其他预测变量）。

•pValue - 假设检验的 t 统计量的 p 值，该假设检验验证对应系数是否等于零。

•观测数值数目- 没有任何 NaN 值的行数。

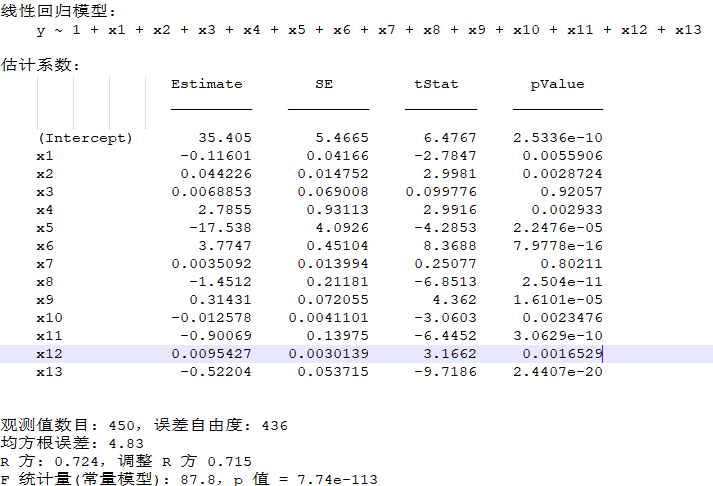
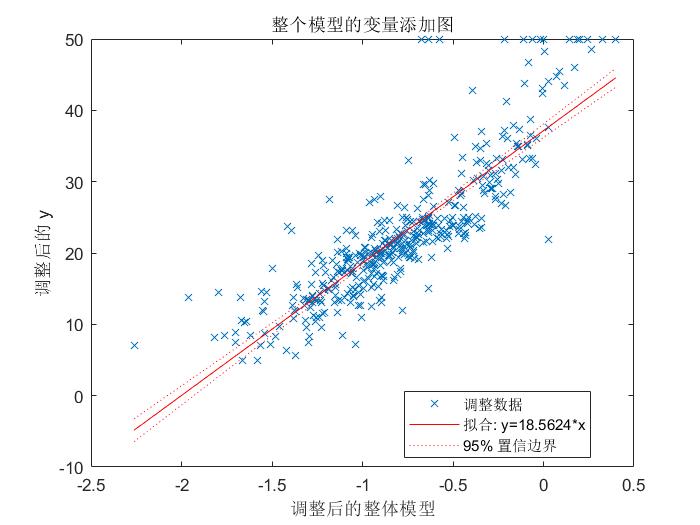
•误差自由度 - n - p，其中 n 是观测值数目，p 是模型中系数的数目，包括截距。

•均方根误差 - 均方误差的平方根，用于估计误差分布的标准差。

•R方和调整R方 - 分别为决定系数和调整决定系数。

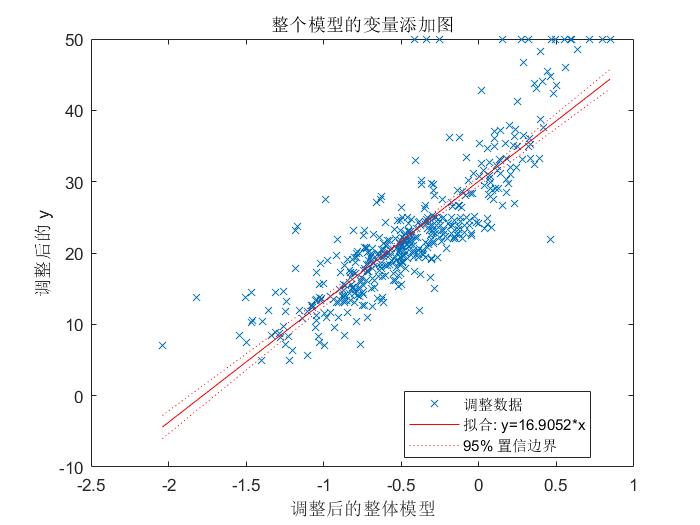
•F统计量.常量模型 - 对回归模型进行 F 检验的检验统计量，用于检验该模型的拟合是否显著优于仅包含常数项的退化模型。

•p值 - 对模型的 F 检验的 p 值。例如，p 值为 7.3816e-27，说明模型是显著的。



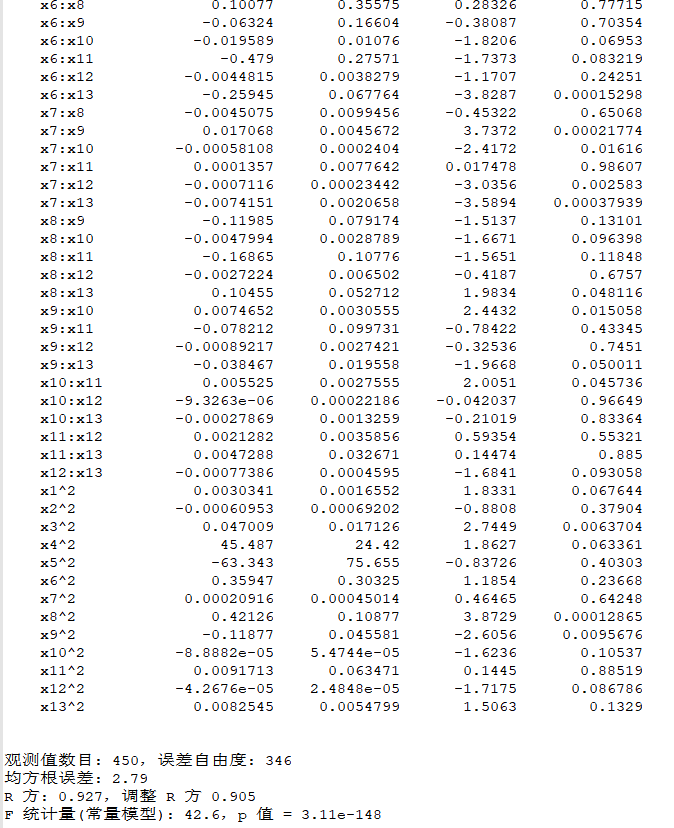
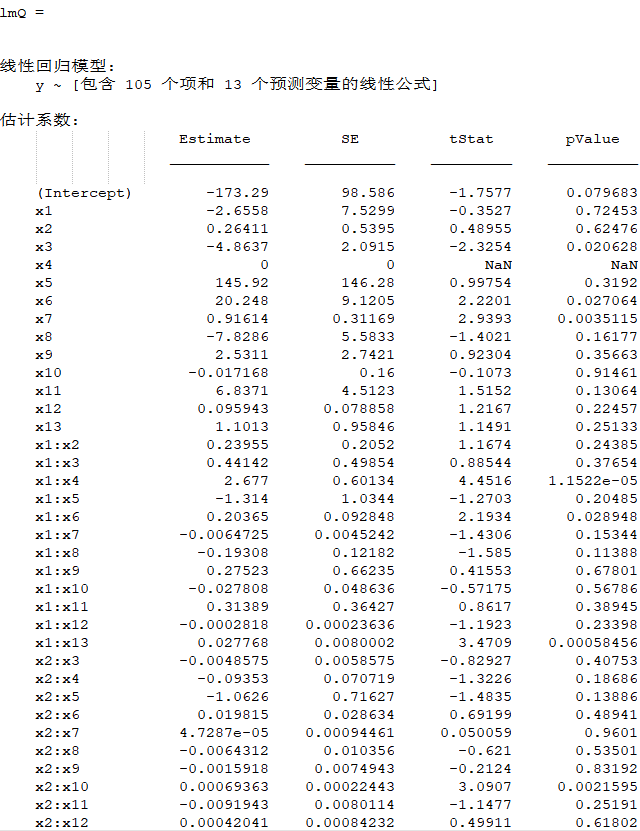
### 4.2.2部分特征线性拟合结果

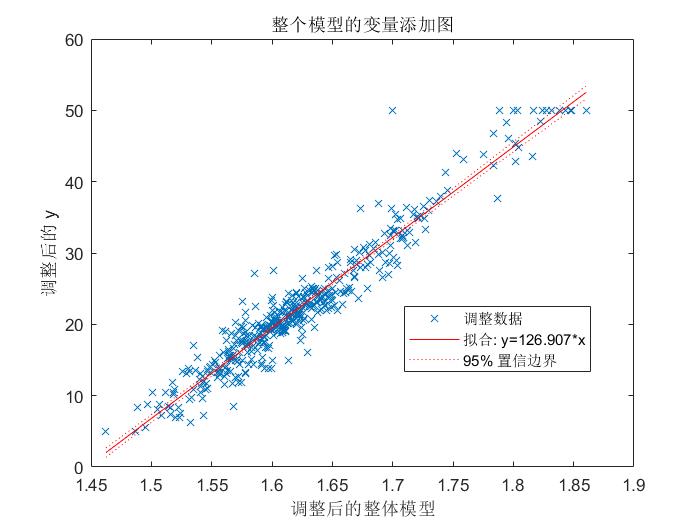
下图是用部分特征线性拟合的模型lmnew，它对测试集输入的预测结果就是绿线。



### 4.2.3变参数线性拟合结果

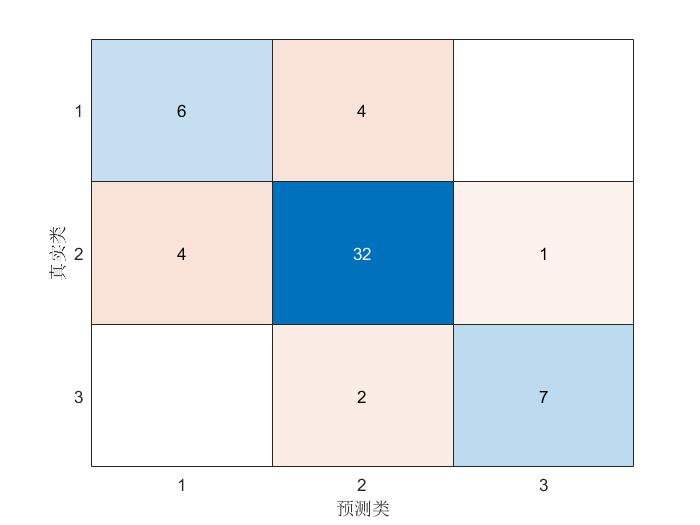
下图是用变参数线性拟合的模型lmQ，它对测试集输入的预测结果就是黄线。





## 4.3回归问题转为分类问题结果

将回归问题转变为分类问题，也就是数据集的房价由连续值改为离散值，记离散值测试集输出为testyC。根据KNN分类器很容易得到预测结果testyChat。用Matlab的confusionmat函数画出混淆矩阵如下。可以56个测试样例中误分类点共有11个。



在KNN分类器训练时，手动选择近邻参数值为5，实际上这个参数可以人为设定。至于为什么要选择5，我们可以将紧邻参数值设定为1-10，每设定一次训练一次KNN，同时分析它的损失（上折线loss）和交叉验证损失（下折线Kloss）。可见K=5时能使得loss和kloss相对地同时较小和较大。由此选择近邻参数为5。

