**数据科学导论第二章实验报告**

|  |  |  |  |
| --- | --- | --- | --- |
| **姓 名:** | 那铭心 | **学 号:** | 921127940133 |
| **组 号:** | 15 | | |
| **组 员:** | **顾翔 921127940122 那铭心921127940133 沈贝宁 92112790137 季诚921127940125** | | |
| **学 院:** | 网络空间安全学院 | | |

**南京理工大学基础前沿交叉中心**

**2022年 9月 22 日**

**目 录**

[案例一 森林火灾面积预测 - 1 -](#_Toc30024)

[【1】案例背景 - 1 -](#_Toc22247)

[【2】方法陈述 - 1 -](#_Toc25442)

[【3】实验代码 - 1 -](#_Toc7896)

[（1） 步骤一、数据处理 - 1 -](#_Toc10996)

[（2） 步骤二、使用Lasso进行特征选择与拟合 - 2 -](#_Toc26522)

[（3） 步骤三、结果展示 - 2 -](#_Toc31079)

[【4】结果分析 - 4 -](#_Toc27826)

[案例二 公共自行车租借预测 - 4 -](#_Toc22762)

[【1】 案例背景 - 4 -](#_Toc16743)

[【2】 方法陈述 - 4 -](#_Toc16945)

[【3】 实验代码（以day. csv为例） - 5 -](#_Toc3069)

[（1） 步骤一: 读入数据 - 5 -](#_Toc6475)

[（2） 步骤二: 对非数值型数据进行onehot编码, 取训练集 - 5 -](#_Toc955)

[（3）步骤三: 进行Lasso回归, 评价回归结果 - 6 -](#_Toc22028)

[4）步骤四: 利用得到的Lasso模型对test数据进行预测, 比较预测值和真实值 - 6 -](#_Toc4136)

[【4】 结果分析 - 7 -](#_Toc23770)

[案例三 根据音轨特征预测音乐起源地 - 8 -](#_Toc6946)

[【1】案例背景 - 8 -](#_Toc29412)

[【2】方法陈述 - 8 -](#_Toc20746)

[【3】实验代码 - 8 -](#_Toc24330)

[（1） 步骤一 :加载需要的包，并读入数据 - 8 -](#_Toc13053)

[（2） 步骤2：对数据进行标准化，选取训练集 - 8 -](#_Toc18751)

[（3） 步骤3：建立线性回归模型，进行Lasso回归，并得出决定系数与RMSE - 9 -](#_Toc27740)

[【4】结果分析 - 10 -](#_Toc15562)

案例四 医疗费用预测...........................................................................................- 10 -

[【1】案例背景 - 10 -](#_Toc7140)

[【2】方法陈述 - 11 -](#_Toc21592)

[【3】实验代码 - 11 -](#_Toc5351)

[【4】结果分析 - 13 -](#_Toc5091)

# 案例一 森林火灾面积预测

## 【1】案例背景

本题由 **顾翔 921127940122**负责

森林作为大自然的调节器，拥有涵养水源，净化空气等多种功能，作为第一产业的组成部分，具有重要的经济价值，然而，由于多种原因，森林火灾频发，给国民经济造成了重大损失，因此，我们需要做好森林火灾的预防工作，防止火灾发生。本题使用的数据集共有517个样本，每个样本包含12个特征，目标特征为火灾面积，由于部分特征之间存在相关性，预测时应进行特征选择。另外，数据集还存在相当一部分火灾面积为0的样本，需考虑如何对目标特征进行相应变化，请使用上述数据集构建一个回归模型，预测森林火灾的火灾面积，请使用合适的评估指标评估回归模型的效果

## 【2】方法陈述

首先进行数据处理，转化文字数据为onehot向量，对数据进行标准化。接着，确定合适的回归模型，这里选择lasso，对结果进行分析，对产生的问题进行改进。

运行环境，python3安装有pandas、scikit-learn、numpy库，运行软件：Anaconda、jupyter notebook

## 【3】实验代码

### 步骤一、数据处理

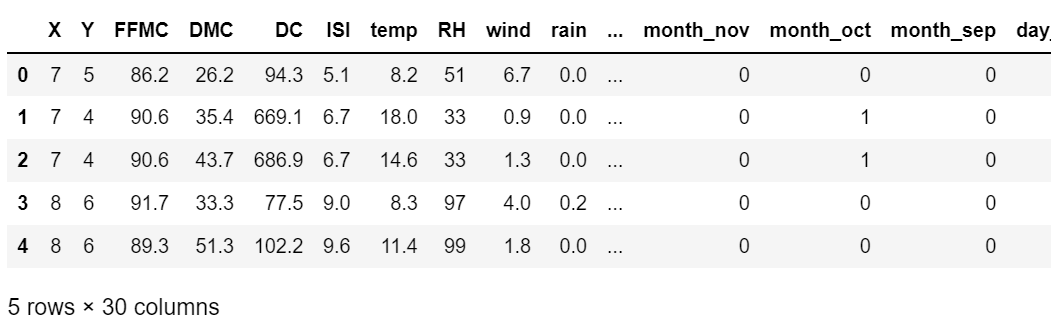
1. 将数据中的年月转化成onehot向量

import pandas as pd

df=pd.read\_csv("forestfires.csv")

df=pd.get\_dummies(df)

df.head()



1. 将数据集随机抽取数据，分成80%训练集和20%测试集，使用z-score进行标准化。

from sklearn.model\_selection import train\_test\_split

from sklearn import preprocessing

train\_x,test\_x=train\_test\_split(df,train\_size=0.8,random\_state=6)

train\_y=train\_x['area']

del train\_x['area']

train\_x=train\_x.values

train\_y=preprocessing.scale(train\_y.values.reshape(-1, 1))

test\_y=test\_x['area']

del test\_x['area']

test\_x=test\_x.values

test\_y=preprocessing.scale(test\_y.values.reshape(-1, 1))

### 步骤二、使用Lasso进行特征选择与拟合

这里选用了来自scikit-learn（<https://scikit-learn.org/stable/modules/classes.html>）的LassoCV库，可以通过交叉验证选择最佳模型

from sklearn.linear\_model import LassoCV

lasso =LassoCV()

lasso.fit(train\_x, train\_y)

pred\_y=lasso.predict(test\_x)

## (3) 步骤三、结果展示

from sklearn.metrics import mean\_squared\_error

**print**('lasso系数',lasso.alpha\_)

del df['area']

coef=pd.Series(lasso.coef\_,index=df.columns)

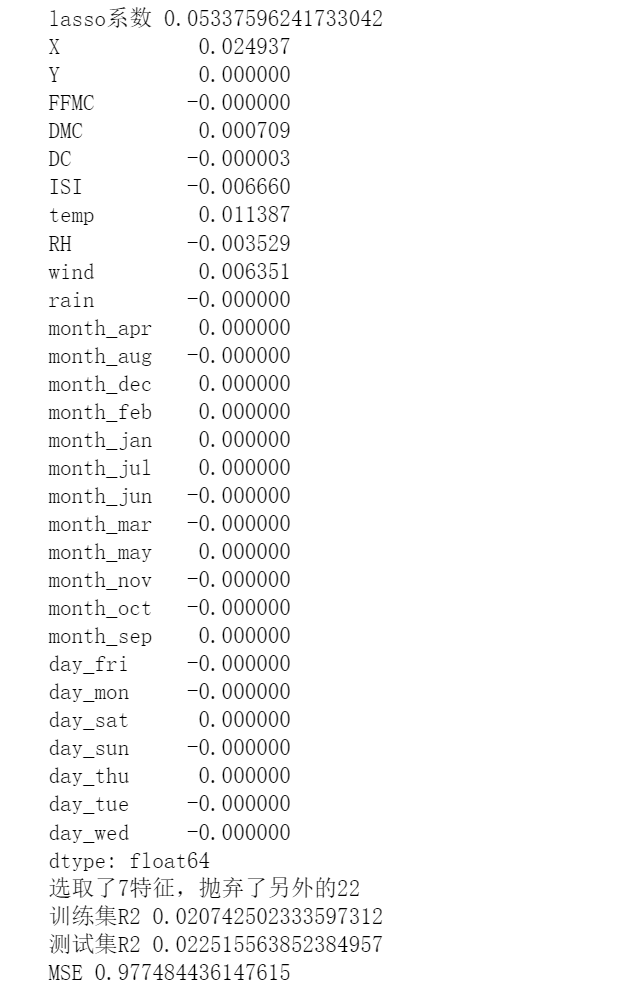
**print**(coef)

**print**("选取了"+str(**sum**(coef!=0))+"特征，抛弃了另外的"+str(**sum**(coef == 0)))

**print**('训练集R2',lasso.score(train\_x, train\_y))

**print**('测试集R2',lasso.score(test\_x, test\_y))

**print**('MSE',mean\_squared\_error(test\_y,pred\_y))



## 【4】结果分析

从结果上来看，在测试集上的MSE较低，但是不管是训练集还是测试集上的R2均较小，说明模型有较为严重的欠拟合，这与lasso系数较小，模型趋于线性回归的特征吻合。

特征选择上，lasso选取了7个相关特征，抛弃了其余的22个，给进一步优化提供了基础，之后可以利用筛选出来的特征结合其他模型进一步拟合优化结果。

# 案例二 公共自行车租借预测

1. **案例背景**

自行车分享系统较传统的自行车租借存在巨大的优势. 它避免了注册、租借和归还等繁琐的过程, 用户可以在线上完成所有相关操作. 同时, 自行车的行驶时长和路线等信息也可以完整记录下来. 本案例将通过一份公共自行车租借数据集, 根据城市当天的天气、温度、湿度和风速等环境信息对自行车的租借数量进行预测. **该部分由沈贝宁完成.**

1. **方法陈述**

首先将day. csv的数据读入python, 将前632组数据作为训练集, 后100组数据作为测试集. 观察数据可以发现, 属性season, yr, month, holiday, weekday, workingday, weathersit是非数值型变量, 因此采用onehot编码将其转化为多个数值型变量的组合. 删除掉例如instant的无关属性, 以cnt作为预测变量进行回归分析. 回归方法采用Lasso, 并对回归结果进行RMSE指标评价. 最后利用得到的Lasso回归模型对剩余的100组数据进行预测, 观察预测值和真实值之间的差异, 从而得出模型的适用性.

对于另外一个表hour. csv, 采用同样的方法处理。

1. **实验代码（以day. csv为例）**
2. **步骤一: 读入数据**



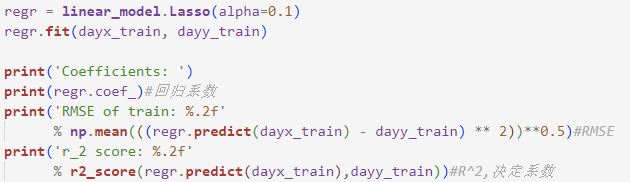
读入数据后, 删除instant, dteday, casual, registered这些无关变量.

1. **步骤二: 对非数值型数据进行onehot编码, 取前632行数据为训练集**



dayy变量存储预测目标, dayx存储输入特征.

**（****3）步骤三: 进行Lasso回归, 并根据RMSE和决定系数评价回归结果**



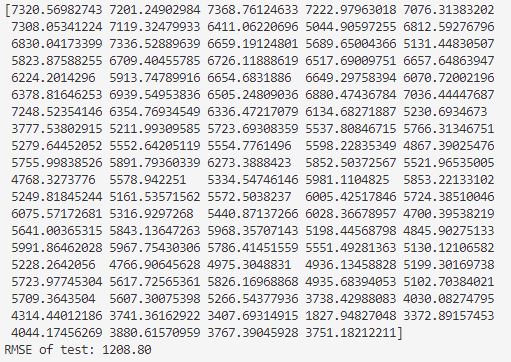


运行结果显示, R2的值为0. 85, 比较接近1, 因此回归结果可信度高.

**4）步骤四: 利用得到的Lasso模型对test数据进行预测, 比较预测值和真实值**



1. **结果分析**



比较预测结果和所得到的RMSE发现, 预测结果与真实值差距并不大, 置信度较高. 模型虽然有一定的适应能力, 但是并不完美. 可以考虑加入新的属性, 提高模型复杂度, 提高决定系数R2. 另外, 因为数据集中的样本个数远大于特征数量, 因此矩阵应该是可逆的. 所以采用OLS回归可能会获得更好的回归效果. 下图展示了采用OLS后的RMSE和R2:



可以发现其中决定系数没有变化, 但是均方根误差减少了0. 03. 相比于Lasso回归, OLS有微小的提升.

# 案例三 根据音轨特征预测音乐起源地

## 【1】案例背景

## 音轨就是音序器软件中的一条一条的平行“轨道”。每条音轨分别定义了该条音轨的属性，如音轨的音色，音乐库，通道数和音量等

**该部分有季诚完成**

## 【2】方法陈述

首先用read.csv()函数将default\_features\_1059\_tracks.csv的数据读入, 将前859组数据作为训练集，后200组数据作为测试集.以音乐起源地经度（即第70列的数据）作为预测变量进行回归分析.回归方法采用Lasso, 并对回归结果进行RMSE指标评价，并测得训练集的决定系数R^2.最后利用得到的Lasso回归模型对剩余的200组数据进行预测，观察预测值和真实值之间的差异，从而得出模型的适用性.

## 【3】实验代码

### 步骤一 :加载需要的包，并读入数据

data <- read.csv("default\_features\_1059\_tracks.csv",encoding = "utf-8",header=F)

library(Matrix)

library(glmnet)

y<-as.matrix(data[,70])

x<-as.matrix(data[,c(1:68)])

### 步骤2：对数据进行标准化，选取前859行为训练集

colMeans(x)

apply(x,2,sd)

x = scale(x,center = T,scale = T)

colMeans(x)

apply(x,2,sd)

train\_y<-y[1:859,1]

train\_x<-x[1:859,c(1:68)]

### 步骤3：建立线性回归模型，**进行Lasso回归，并得出决定系数与RMSE**

fit=glmnet(train\_x ,train\_y, family="gaussian", intercept = F, alpha=1)

plot(fit,xvar = "lambda", label = F)

mod\_cv <- cv.glmnet(x=train\_x, y=train\_y, family="gaussian", intercept = F, alpha=1)#交叉验证

plot(mod\_cv)

best\_lambda <- mod\_cv$lambda.min #选mod\_cv$lambda.min为最佳lambda

best\_lambda #最佳的lambda

best\_model <- glmnet(train\_x ,train\_y, alpha = 1, lambda = best\_lambda)#最佳的模型

coef(best\_model)#模型的自变量系数

y\_predicted <- predict(best\_model, s = best\_lambda, newx = train\_x)

sst <- sum((train\_y - mean(train\_y))^2)

sse <- sum((y\_predicted - train\_y)^2)

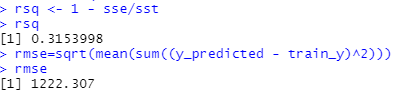
#find R-Squared

rsq <- 1 - sse/sst

rsq

rmse=sqrt(mean(sum((y\_predicted - train\_y)^2)))

rmse



1. **步骤四：利用得到的Lasso模型对test数据进行预测，比较预测值和真实值**

test\_x<-x[(860:1059),c(1:68)]

test\_y<-y[(860:1059),1]

test\_y\_predicted <- predict(best\_model, s = best\_lambda, newx = test\_x)

test\_sst <- sum((test\_y - mean(test\_y))^2)

test\_sse <- sum((test\_y\_predicted - test\_y)^2)

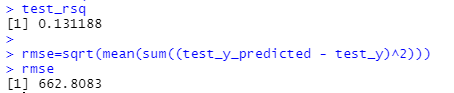
#find R-Squared

test\_rsq <- 1 - test\_sse/test\_sst

test\_rsq

rmse=sqrt(mean(sum((test\_y\_predicted - test\_y)^2)))

rmse



## 【4】结果分析

比较预测结果和所得到的RMSE发现，在训练集和测试集上预测结果与真实值都存在较大差距，决定系数R^2较小,目前的模型的回归效果较差，仍需改进。

**案例四 医疗费用预测**

## 【1】案例背景

保险公司通常需要募集比花费在受益者的医疗服务上更多的年度保费。因此，精确预测医疗费用对保险公司具有重要价值。**该部分由那铭心完成**

本体提供的数据集是从美国人口普查局的人口统计资料整理得出。数据集共有1338个样本，包含7个特征。

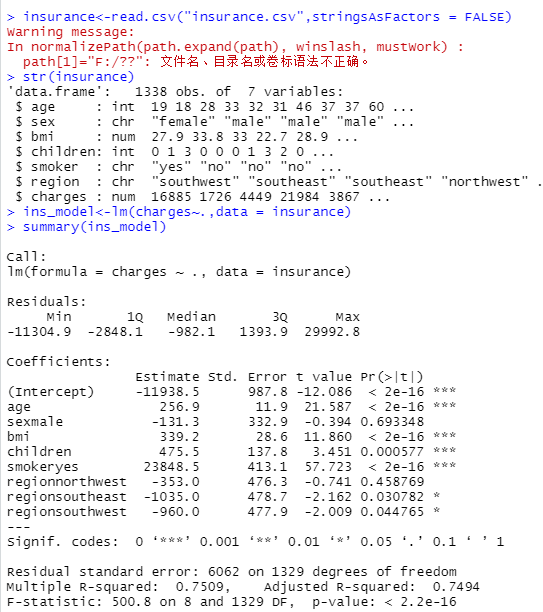
实验目的为以“charges”作为目标特征构建回归模型，预测收益者的平均医疗费用。

## 【2】方法陈述

确定非数值型变量和数值型变量，以所有数值型变量作为预测变量和“charges”进行多元线性回归拟合。最后测试适用性。

## 【3】实验代码

1. **步骤一 探索与准备数据**

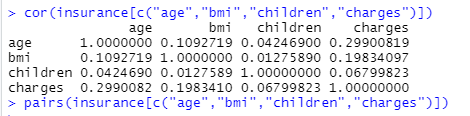


read.csv()读入数据,stringsAsFactors = T将名义变量转换成因子变量;

str()查看数据结构,确认各个变量的读取是否有异常;

summary()查看charges的分布情况;

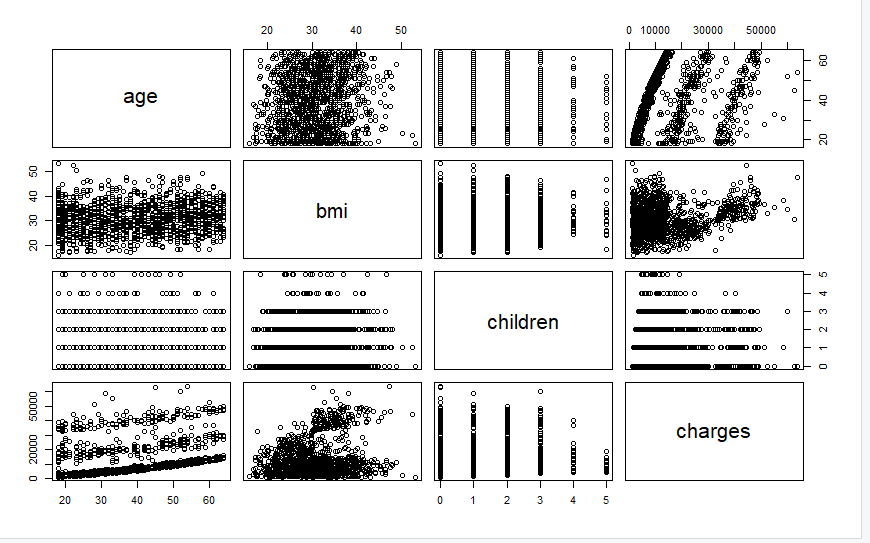
1. **探索与可视化特征之间关系**



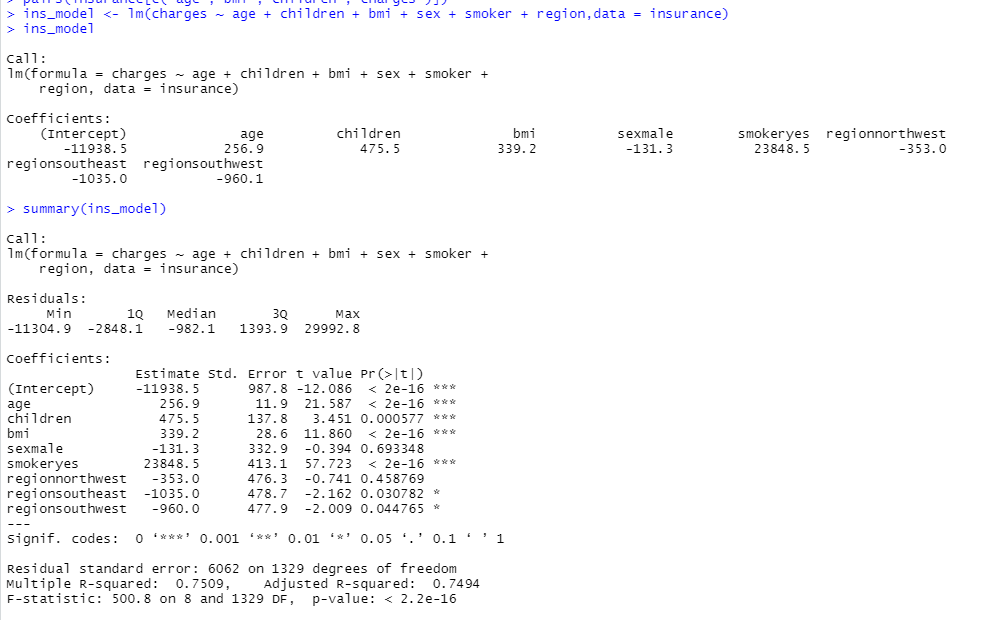
cor(insurance[c("age","bmi","children","charges")])创建相关系数矩阵

pairs(insurance[c("age","bmi","children","charges")])绘制散点图矩阵

散点图如下



**（3）初步拟合线性回归模型并测试**



ins\_model <- lm(charges ~ age + children + bmi + sex + smoker + region,data = insurance)

ins\_model 拟合线性回归模型

summary(ins\_model)查看分布情况以评估模型性能  
**【4】结果分析**

r方为0.75，拟合模型是正常合格的。但模型仍可以进行改进。