

**机器学习作业**

**题 目: 解析解法及梯度下降法求解线性回归问题**

**学 号: 919106840333**

**姓 名: 孙傲歆**

**指导教师: 潘金山**

**2021年11月**

**一、题目要求**

给出历史数据

Year 𝑥 = [2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008,

2009, 2010, 2011, 2012, 2013]

Price 𝑦 = [2.000, 2.500, 2.900, 3.147, 4.515, 4.903, 5.365, 5.704,

6.853, 7.971, 8.561, 10.000, 11.280, 12.900]

• 假设: 房价和年度呈线性关系

• 任务

– 分别通过1)闭式解 和 2)梯度下降算法建立线性回归模型，

得到𝑥和𝑦的关系；

– 用该模型预测2014年南京的平均房价。

**二、算法思路**

**1.解析解法求解线性回归模型**

此方法求解线性回归模型较为简单，只需要直接将相应的数值代入到闭式解之中，即可求得结果。通过matlab编程实现。

**2.梯度下降法求解线性回归模型**

此方法相比闭式解法较为复杂。经过尝试，发现X的值（即年份的值）如果不经过处理直接应用于梯度下降法之中，学习率不好控制，会出现无限次迭代，最终结果趋于无穷的情况。所以先将年份数据进行标准化，公式为：(X-X\_mean)/X\_std。预处理过后在运用递归下降法的公式进行迭代，初始值theta\_0=1、theta\_1=1、学习率a=0.1。迭代时设置两个参数loss和loss\_old来计算本次迭代和上次迭代计算结果的差值，若差值小于0.00001则认为收敛，迭代结束。

此外，我还通过改变学习率a的值，探寻了学习率与迭代次数之间的关系。

**三、运行结果**

**1.解析解法运行结果**

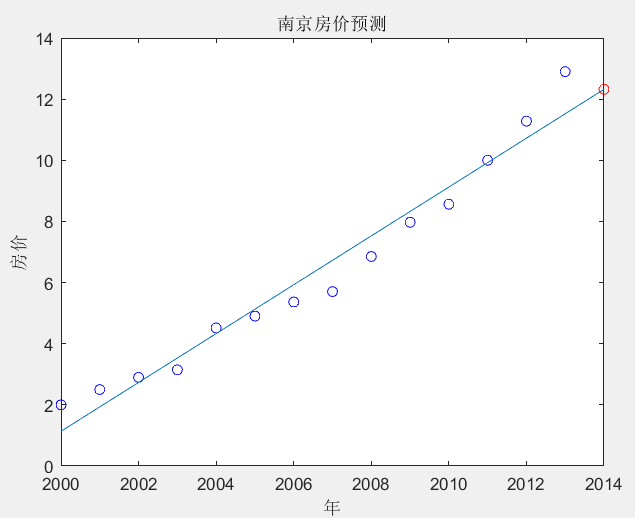


图1.1



图1.2

上图为代码运行结果，其中图1.1的蓝线是解析解法求得的回归方程。蓝色的点是原始数据值，红色的点是2014年预测房价值。图1.2预测结果，可以看到2014年房价预测值是12.321

**2.梯度下降法运行结果**

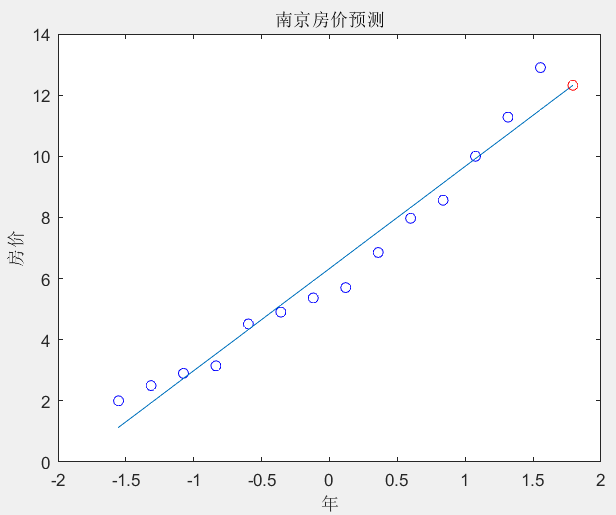


图2.1

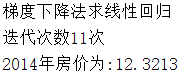


图2.2

上图为代码运行结果，其中图2.1的蓝线是解析解法求得的回归方程。蓝色的点是原始数据值，红色的点是2014年预测房价值。图2.2是预测及迭代结果，可以看到在学习率设置为0.1的情况下，仅通过11次迭代便达到了收敛，2014年房价预测值为12.3213和解析解法求得的结果十分相近。

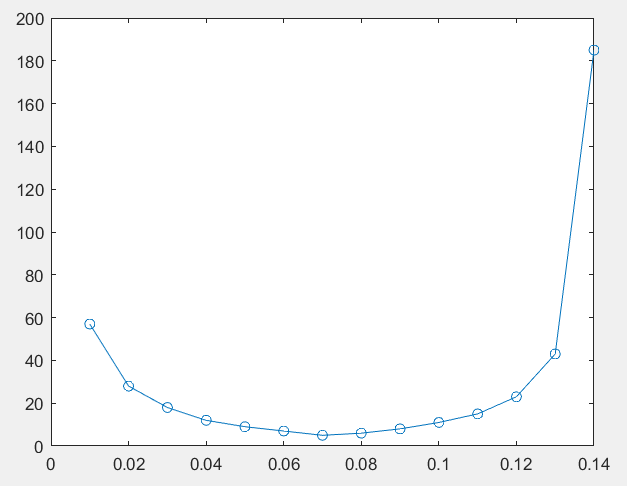


图2.3

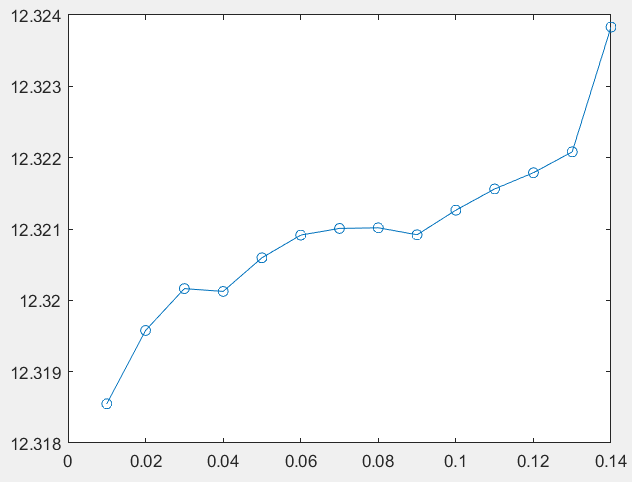


图2.4

上图为测试学习率和迭代次数之间关系的运行结果。我们这里去a=0.01、0.02、0.03……0.14。从图2.3可以看出，学习率与迭代次数是一个先减后增的关系，即学习率过低或过高都会导致迭代次数增加，甚至达到一定值时会出现无穷迭代，结果趋于无穷的情况。（经个别值的测试，a取0.15时结果已趋向于无穷；而a取1\*10^-7时，迭代次数已经开始激增，结果也开始偏离正常值）从图2.4可以看出，改变学习率，在正常范围内结果收敛后其预测结果相差不大，但整体是一个递增的趋势。

**四、可运行代码**

**1.解析解法代码**

clc

clear;

x=[2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008,2009, 2010, 2011, 2012, 2013];

X=[1,2000;1,2001;1,2002;1,2003;1,2004;1,2005;1,2006;1,2007;1,2008;

1,2009;1,2010;1,2011;1,2012;1,2013];

Y=[2.000;2.500;2.900;3.147;4.515;4.903;5.365;5.704;6.853;7.971;8.561;10.000;11.280;12.900];

theta=inv(X'\*X)\*X'\*Y;

Y\_2014=[1,2014]\*theta;

disp("解析解法求线性回归");

disp("2014年房价为:"+Y\_2014);

figure(1);

plot(x,Y','bo');

xlabel('年');

ylabel('房价');

title('南京房价预测');

hold on;

x=[2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008,2009, 2010, 2011, 2012, 2013 ,2014];

y=[X\*theta;Y\_2014];

plot(x,y,'-');

hold on;

plot(2014,Y\_2014,'ro');

**2.梯度下降法代码**

clc;

clear;

theta\_0=1;

theta\_1=1;

x=[2000;2001;2002;2003;2004;2005;2006;2007;2008;2009;2010;2011;2012;2013];

y=[2.000;2.500;2.900;3.147;4.515;4.903;5.365;5.704;6.853;7.971;8.561;10.000;11.280;12.900];

a=0.1;

mean=mean(x);

std=std(x);

for i=1:14

x(i)=(x(i)-mean)/std;

end

old\_loss=0;

loss=0;

for i=1:14

loss=loss+(x(i)\*theta\_1+theta\_0-y(i))\*(x(i)\*theta\_1+theta\_0-y(i));

end

sum=0;

while (abs(loss - old\_loss) > 0.00001)

dew=0;

for i=1:14

dew= dew+(x(i)\*theta\_1-y(i))\*x(i);

end

theta\_1=theta\_1-dew\*a;

dew=0;

for i=1:14

dew=dew+(x(i)\*theta\_1+theta\_0-y(i));

end

theta\_0=theta\_0-dew\*a;

old\_loss=loss;

loss=0;

for i=1:14

loss=loss+(x(i)\*theta\_1+theta\_0-y(i))\*(x(i)\*theta\_1+theta\_0-y(i));

end

sum=sum+1;

end

Y\_2014=((2014-mean)/std)\*theta\_1+theta\_0;

disp("梯度下降法求线性回归");

disp("迭代次数"+sum+"次");

disp("2014年房价为:"+Y\_2014);

figure(1);

plot(x,y','bo');

xlabel('年');

ylabel('房价');

title('南京房价预测');

hold on;

x2=[x;(2014-mean)/std];

y2=[x\*theta\_1+theta\_0;Y\_2014];

plot(x2,y2,'-');

hold on;

plot((2014-mean)/std,Y\_2014,'ro');