单变量线性回归——梯度下降法

alihuang

目录

1 Python code 1

2 R code 5

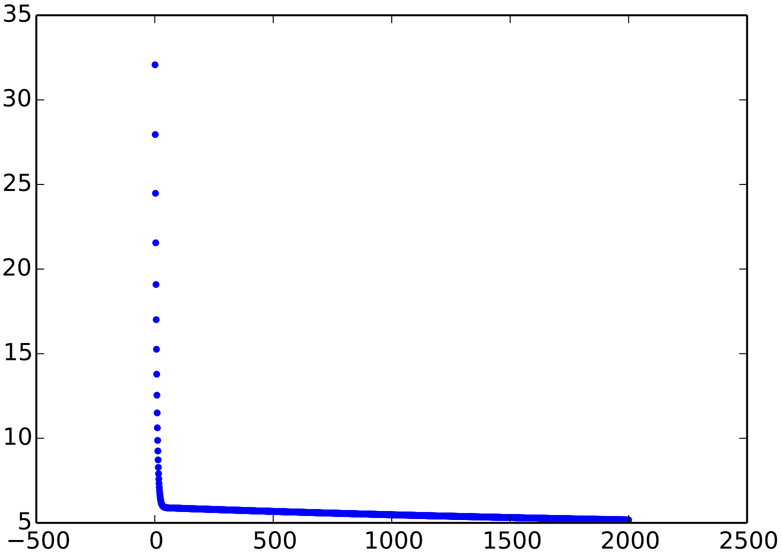
1 Python code

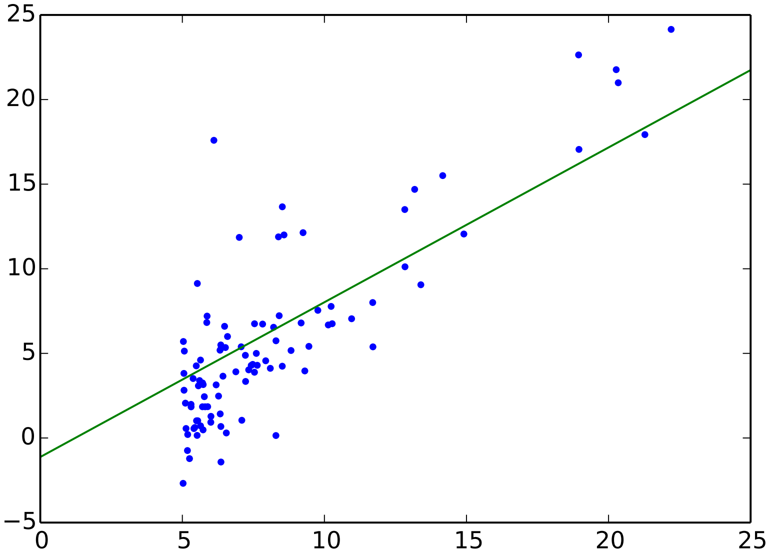
```
#encoding: utf-8  不加会报错
import numpy as np
from matplotlib import pyplot as plt
def y_hat(theta,x_trans):
    return np.dot(x_trans,theta)
def J_cost(m,theta,x_trans,y):
    s=(y_hat(theta,x_trans)-y)*(y_hat(theta,x_trans)-y)
    #ans=(1/(2*m))*sum(s)    ####两行在r调用python时不同，但在pycharm时允许结果相同
    ans=sum(s)/(2*m)     ###
    return ans[0]
def derivative(theta, m , x_trans , y,alpha):
    for j in range(0,len(theta)):
        x_j=np.array(x_trans[:, j])
        x_j=x_j.reshape(m,1)
        theta[j]=theta[j]-alpha/m*sum(np.multiply((y_hat(theta,x_trans)-y),x_j))
    return theta
def main():
```

```
data = np.loadtxt('ex1data1.txt', delimiter=',')
iterations = 2000
alpha = 0.001
m = len(data)
x = np.empty(m)
y = np.empty(m)
theta = np.zeros((2, 1))
for j in range(0, len(data)):
    x[j] = data[j][0]
    y[j] = data[j][1]
# 添加x0=1的常数项
ones = np.empty(m)
for i in range(0, m):
    ones[i] = 1
# 构建行为样本，列为特征的举证
x_trans = []
for i in range(0, m):
    x_trans.append([ones[i], x[i]])
x_trans = np.array(x_trans)
y = y.reshape(m, 1)  # y改为m行一列矩阵
n=0
j_cost_results=[]
n_results=[]
while n<iterations :
    j_cost_results.append(J_cost(m, theta, x_trans, y))
    n=n+1
    n_results.append(n)
    temp=derivative(theta, m, x_trans, y, alpha)
    theta=temp
plt.plot(n_results,j_cost_results,".")
plt.xlim(-500,2500)
plt.show()
plt.close()
```

```
    print(theta)
    print(J_cost(m,theta,x_trans,y))
    plt.plot(x,y,".")
    x = np.linspace(0,25)
    y = theta[0]+theta[1]*x
    plt.plot(x,y)
    plt.show()
    #x,y,theta均不能直接导入到r中
    x=x.tolist()
    y=y.tolist()
    theta=theta.tolist()
    c=1
    np.savetxt('theta.csv', theta , delimiter=',')
    np.savetxt('n_results.csv', n_results , delimiter=',')
    np.savetxt('j_cost_results.csv', j_cost_results , delimiter=',')
main()
```

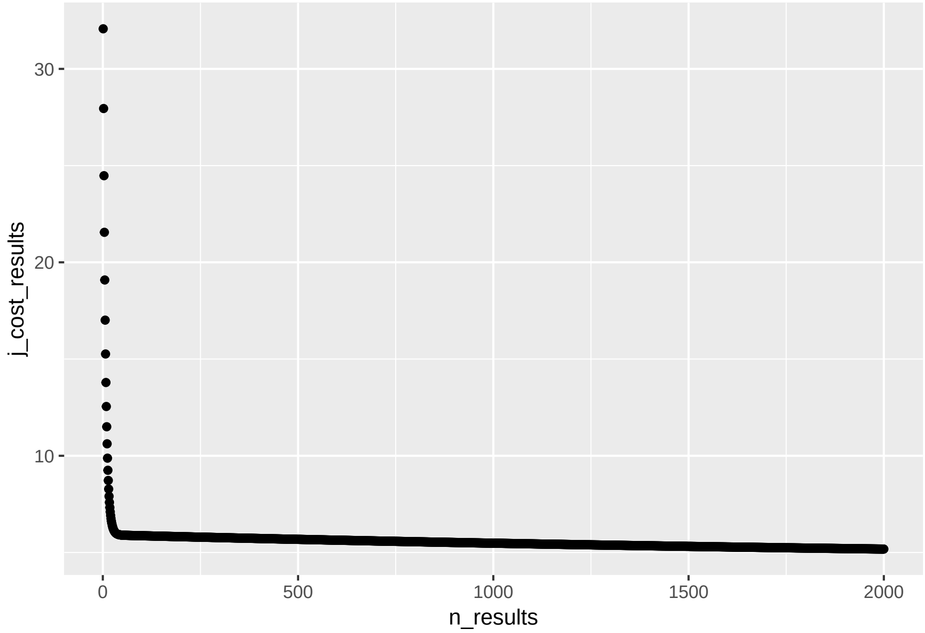
```
## [[-1.12445174]
##  [ 0.9146733 ]]
## 5.17617583609
```


35
30
25
20
15
10
5
−500
0
500
1000
1500
2000
2500

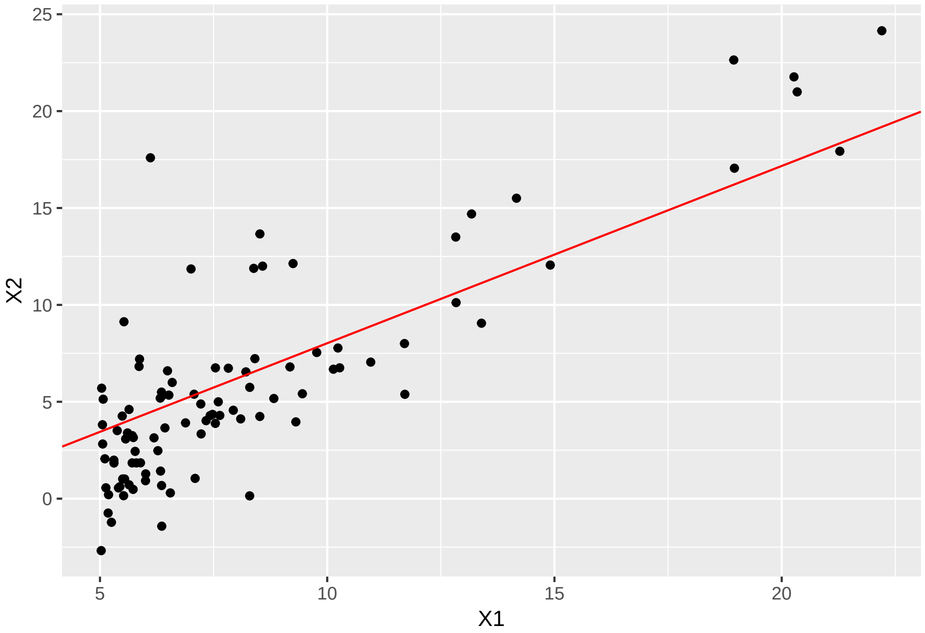

25
20
15
10
5
0
−5
0
5
10
15
20
25

2 R code

```
a <- read_csv('theta.csv',col_names = FALSE,skip=0)
b <- read_csv('ex1data1.txt',col_names = FALSE,skip=0)
a <- as.numeric(unlist(a))#直接as.numeric会显示无法强制变为double种类

n_results  <- read_csv('n_results.csv',col_names = FALSE,skip=0)
n_results <- as.numeric(unlist(n_results))

j_cost_results <- read_csv('j_cost_results.csv',col_names = FALSE,skip=0)
j_cost_results <- as.numeric(unlist(j_cost_results))

ggplot(mapping = aes(x=n_results,y=j_cost_results))+
  geom_point()
```



```
ggplot(data=b,mapping = aes(x=X1,y=X2))+
  geom_point()+
  geom_abline(aes(intercept=a[1],slope=a[2]),color="red")
```


X2
X1
25
20
15
10
5
0
5
10
15
20