import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from sklearn.linear\_model import RidgeCV, Ridge  
from sklearn.metrics import mean\_squared\_error  
  
plt.rcParams['font.sans-serif']=['SimHei']  
plt.rcParams['axes.unicode\_minus']=False  
  
import warnings  
warnings.filterwarnings('ignore')  
  
#导入我们的数据  
df=pd.read\_excel('.\\第四问.xlsx')  
print(df)  
  
from sklearn import model\_selection #将数据划分训练组和测试组  
  
x = df[['美国进口指数','美国通用关税税率','日本进口指数','日本通用关税税率','德国进口指数','德国通用关税税率']]  
y = df['新能源汽车保有量']  
  
x\_train,x\_test,y\_train,y\_test=model\_selection.train\_test\_split(x,y,test\_size=0.2,random\_state=1234)  
Lambdas = np.logspace(-5, 2, 200)  
#这里求最优的λ值用的是交叉验证法  
ridge\_cv=RidgeCV(alphas=Lambdas,scoring='neg\_mean\_squared\_error',cv=6)  
ridge\_cv.fit(x\_train,y\_train)  
ridge\_best\_lambda=ridge\_cv.alpha\_  
print('最优lambda：',ridge\_best\_lambda)  
  
#用最优的λ值来搭建岭回归模型  
linear3=Ridge(alpha=ridge\_best\_lambda)  
linear3.fit(x\_train,y\_train)  
print('预测结果：',linear3.coef\_)  
pred2=linear3.predict(x\_test)  
MSE2=mean\_squared\_error(pred2,y\_test)  
print('MSE2:',MSE2)  
  
plt.scatter(pred2,y\_test)  
plt.show()  
  
from sklearn.linear\_model import LinearRegression  
linear1=LinearRegression()  
linear1.fit(x\_train,y\_train)  
  
print(linear1.coef\_)  
pred1=linear1.predict(x\_test)  
MSE1=mean\_squared\_error(pred1,y\_test)  
print(MSE1)  
  
  
plt.scatter(pred1,y\_test)  
plt.show()