**‍****实验目的**

Kaggle竞赛Spaceship Titanic预测，用svm、决策树、集成学习三种模型解决典型的二分类问题，通过处理13个特征预测是否被转移这个标签。 在本地运行后提交到Kaggle评测

# 数据处理

### 拆解用户姓名为长度

data['NameLength'] = data['Name'].str.len()-1

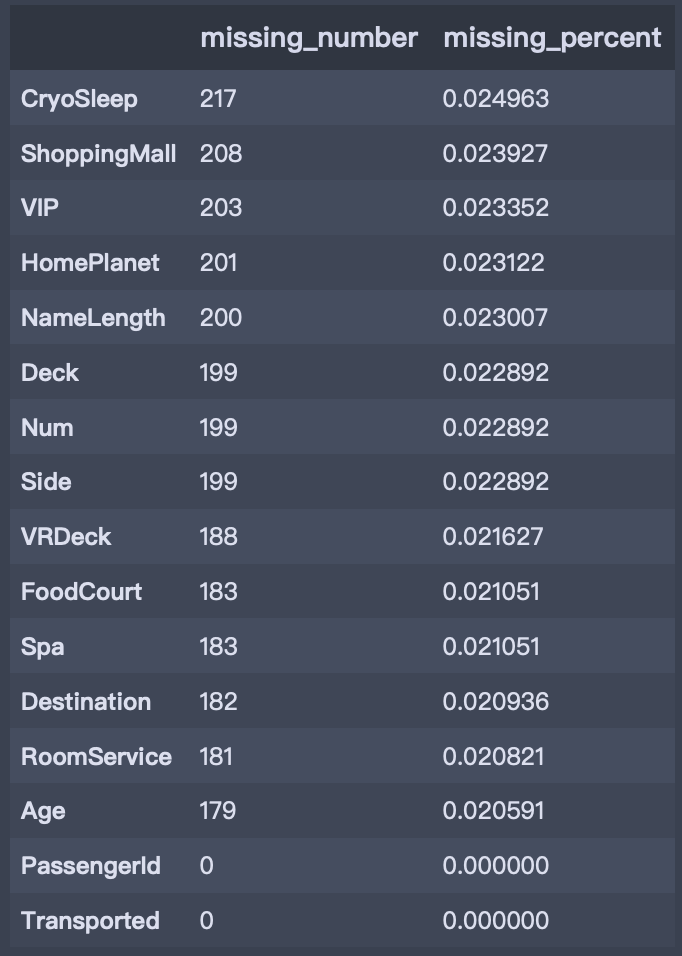
### 将cabin拆分为'Deck', 'Num','Side'三个特征

data[['Deck', 'Num','Side']] = data['Cabin'].str.split('/', expand=True)

### 判断乘客id是否相同, 是否有重复行

统计并且删除缺失值

Utils.missing(data)

​

# 训练集删除缺失值  
data=data.dropna(how='any')  
# 测试集用上一个数值填充  
test=test.fillna(method='ffill')  
data.describe()

### OrdinalEncoder编码

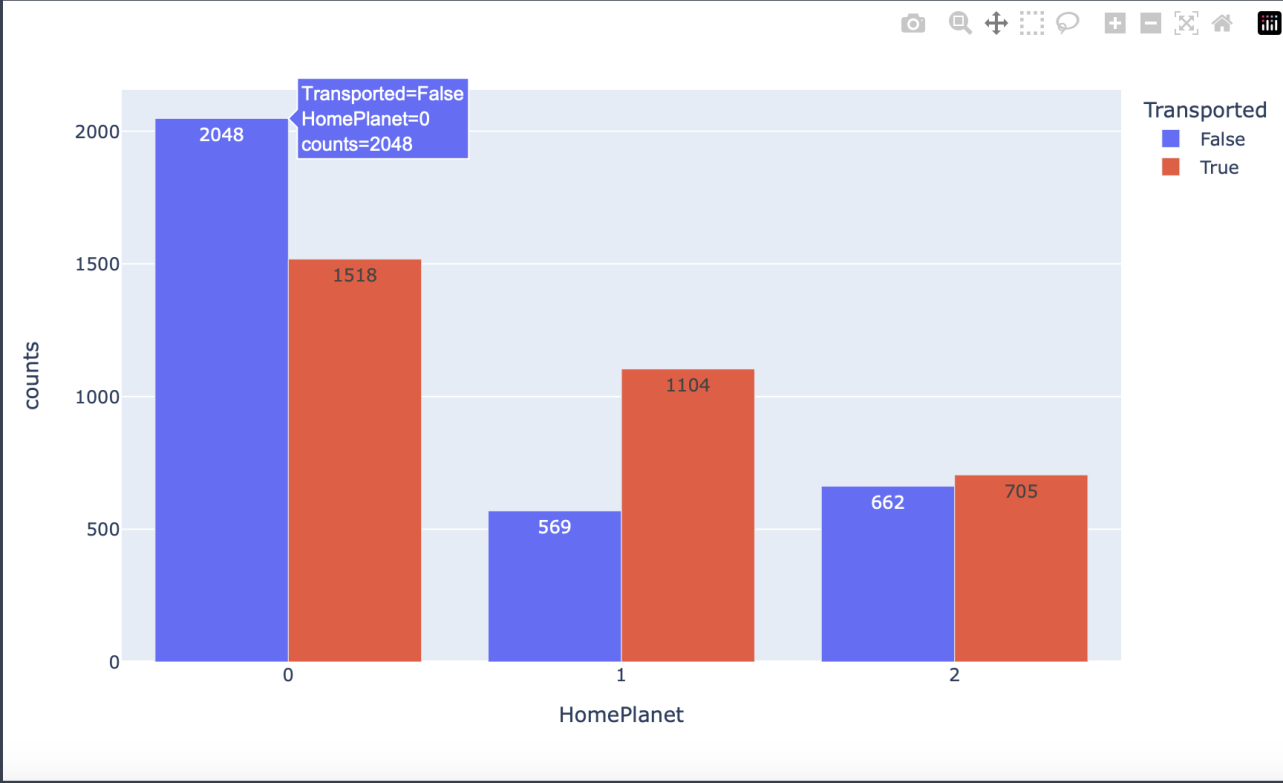
对'HomePlanet', 'Destination', 'Deck', 'Num', 'Side'五个字段进行OrdinalEncoder编码

cate = ['HomePlanet', 'Destination', 'Deck', 'Num', 'Side']#剩余的object特征  
data['CryoSleep'] = data['CryoSleep'].astype(bool)  
data['VIP'] = data['VIP'].astype(bool)  
from sklearn import preprocessing  
# 实例化转化器  
enc = preprocessing.OrdinalEncoder()  
enc.fit(data[cate])  
data[cate] = enc.transform(data[cate])

### 探究关系

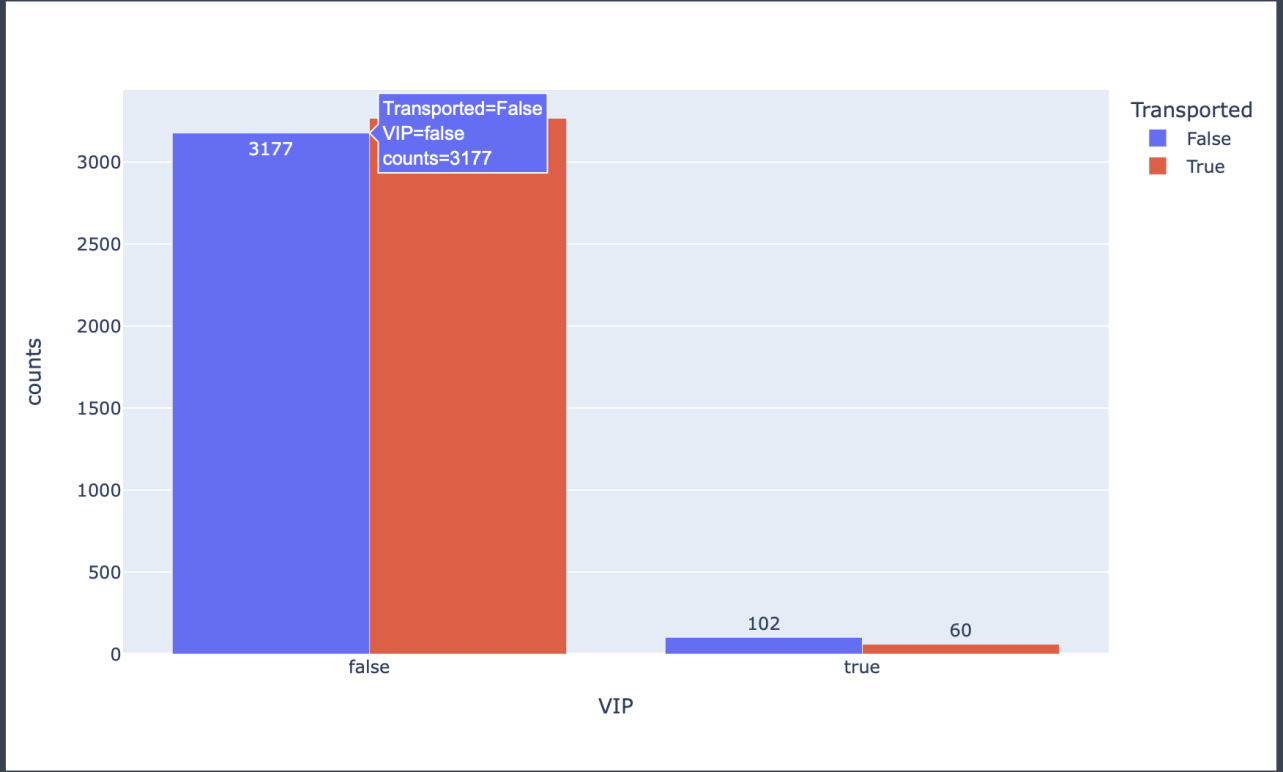
是否转移和出发星球的关系

# 统计出发星球和是否被转移的关系  
HomePlanet = data.groupby(['HomePlanet', 'Transported'])[['HomePlanet', 'Transported']].size().reset\_index(name = 'counts')  
HomePlanet  
# [['HomePlanet', 'Transported']].size().reset\_index(name = 'counts')  
fig = px.bar(HomePlanet, x = 'HomePlanet', y = 'counts',color = 'Transported', text\_auto = True, barmode = 'group')  
fig.show()



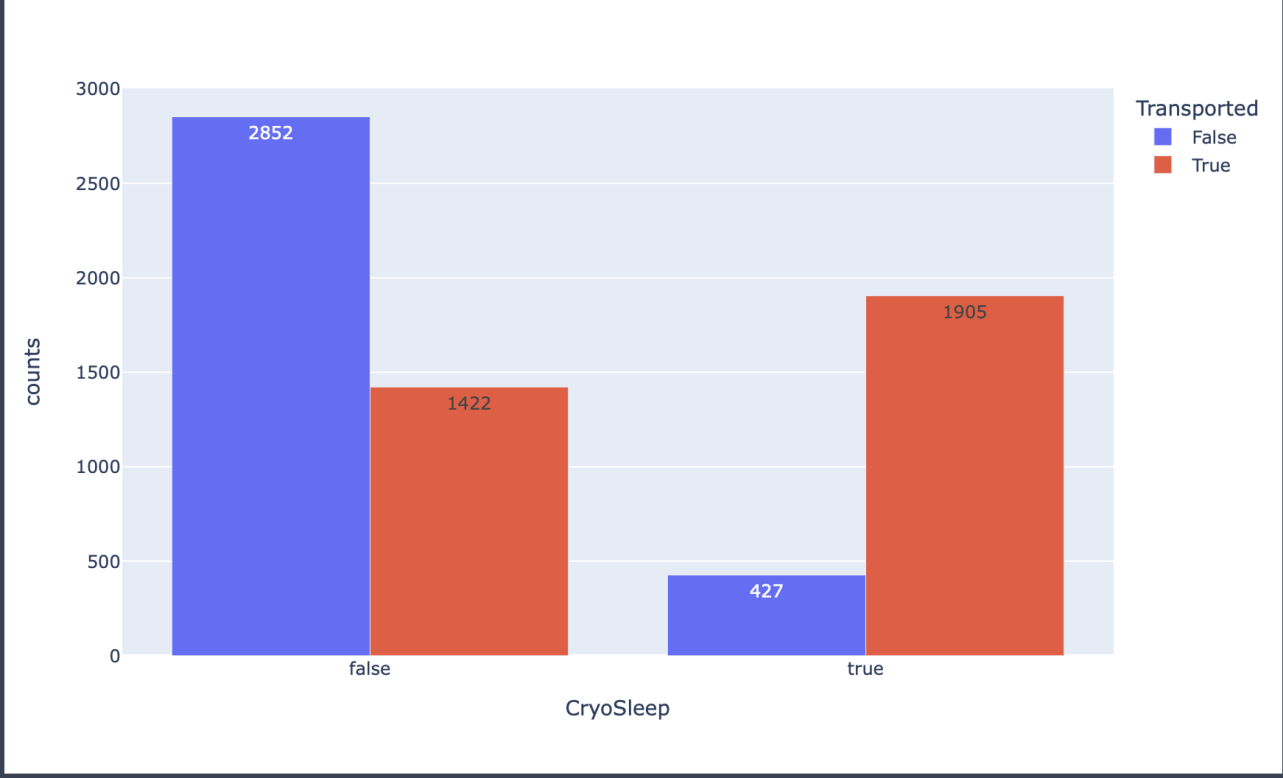
是否转移和VIP的关系

# 统计VIP和是否被转移的关系  
HomePlanet = data.groupby(['VIP', 'Transported'])[['VIP', 'Transported']].size().reset\_index(name = 'counts')  
HomePlanet  
# [['HomePlanet', 'Transported']].size().reset\_index(name = 'counts')  
fig = px.bar(HomePlanet, x = 'VIP', y = 'counts',color = 'Transported', text\_auto = True, barmode = 'group')  
fig.show()



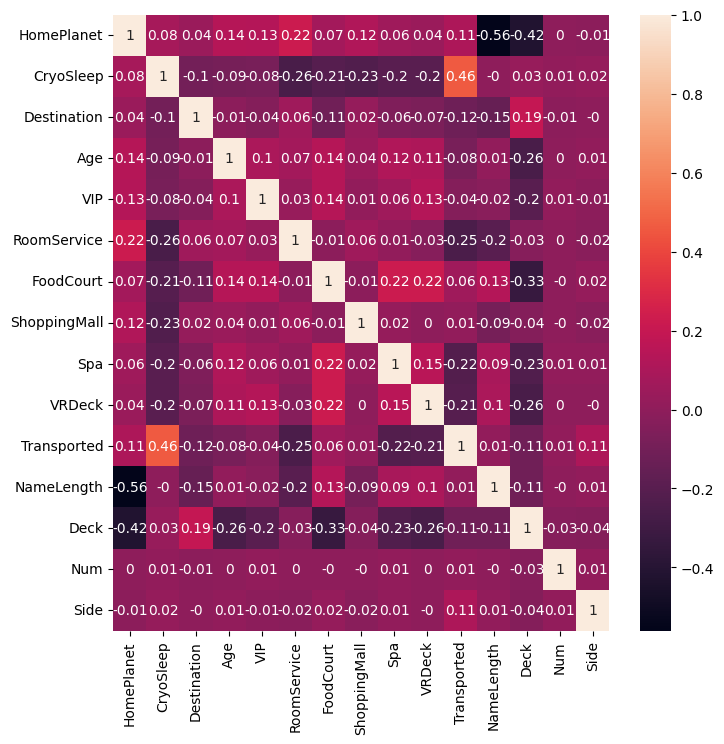
是否转移和是否休眠的关系

# 统计是否睡眠和是否被转移的关系  
HomePlanet = data.groupby(['CryoSleep', 'Transported'])[['CryoSleep', 'Transported']].size().reset\_index(name = 'counts')  
HomePlanet  
# [['HomePlanet', 'Transported']].size().reset\_index(name = 'counts')  
fig = px.bar(HomePlanet, x = 'CryoSleep', y = 'counts',color = 'Transported', text\_auto = True, barmode = 'group')  
fig.show()



### 相关系数

从图中可以看出是否休眠和是否转移相关性很大

​

## 模型设计和选择

模型选择支持向量机，决策树，adaboost三个模型

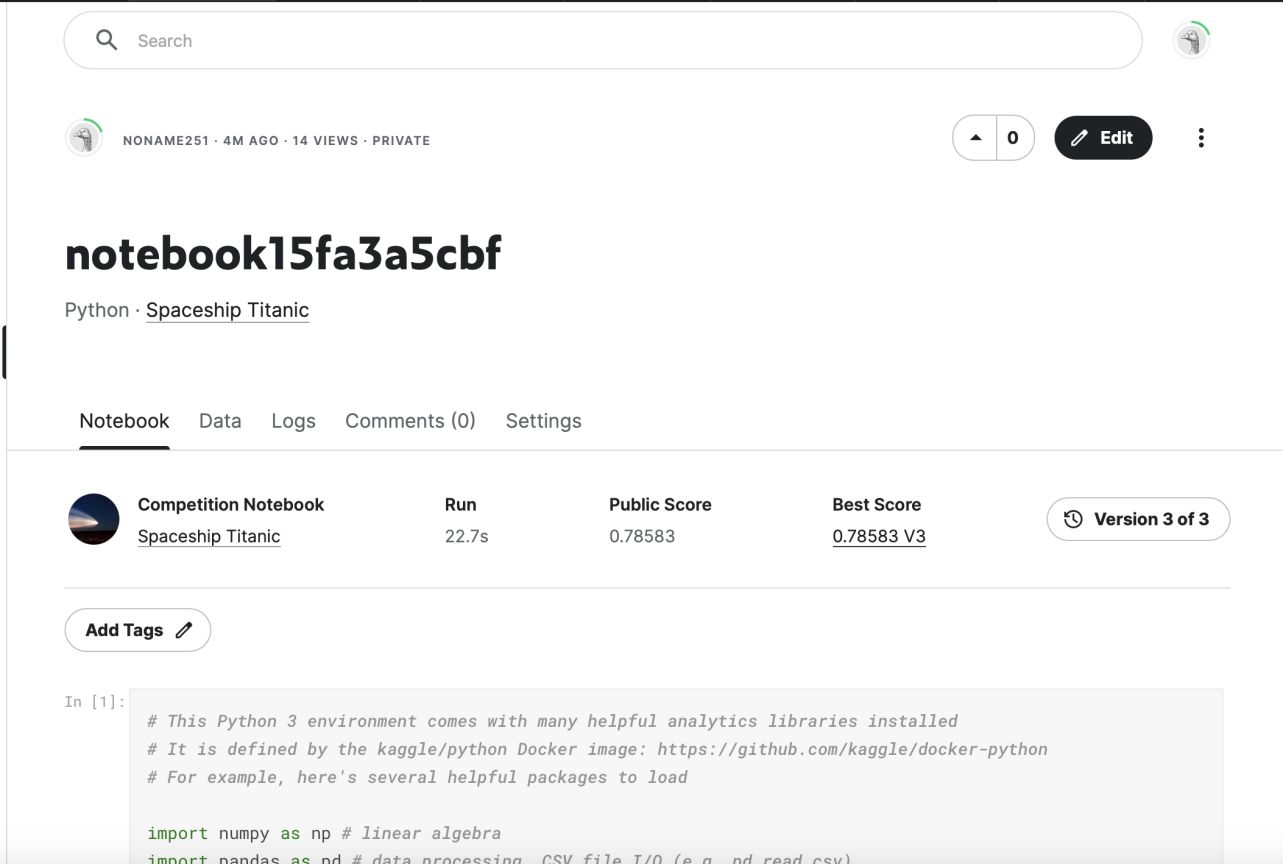
## 预测结果

训练集上

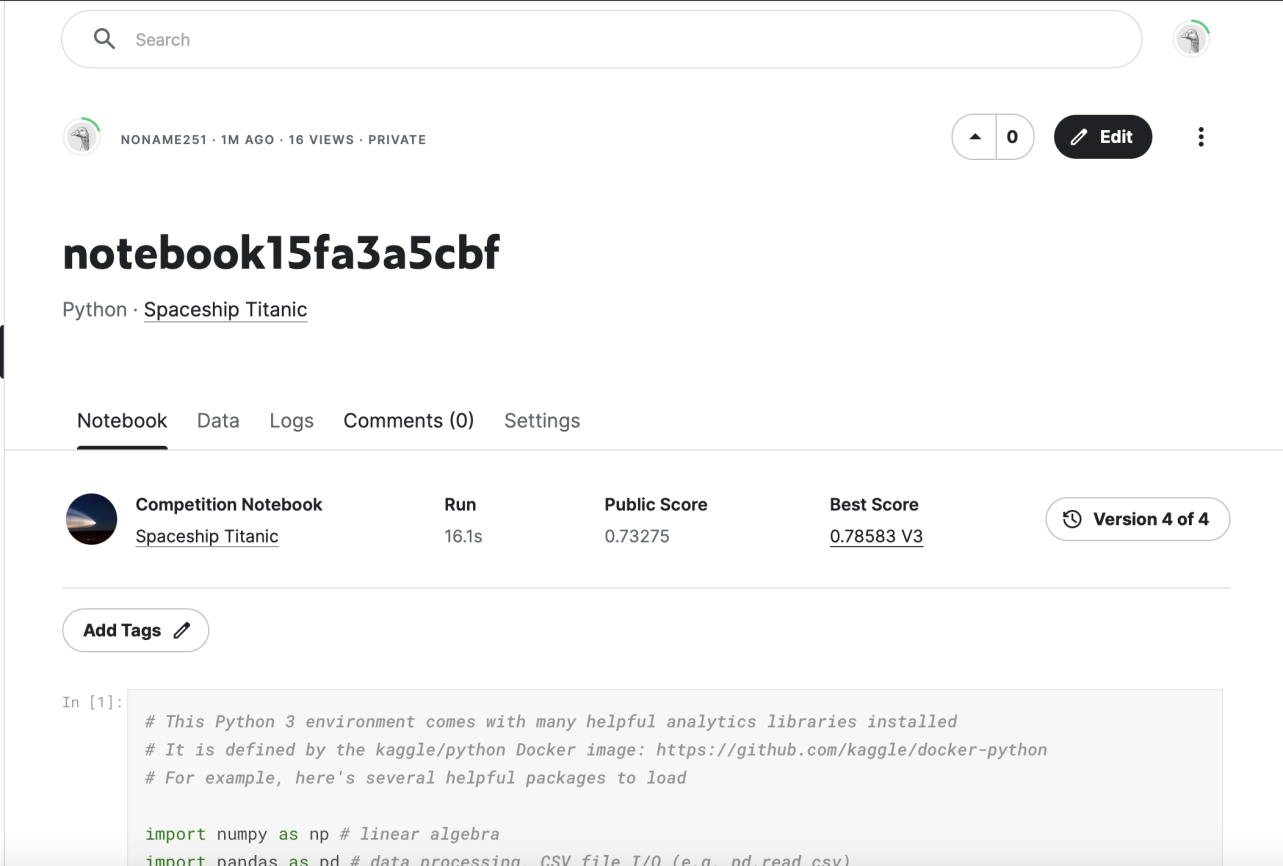
SVM 准确率：0.7905569007263923, 查准率：0.7467467467467468 查全率：0.8891537544696066

决策树 准确率：0.7639225181598063, 查准率：0.7644287396937574 查全率：0.7735399284862932

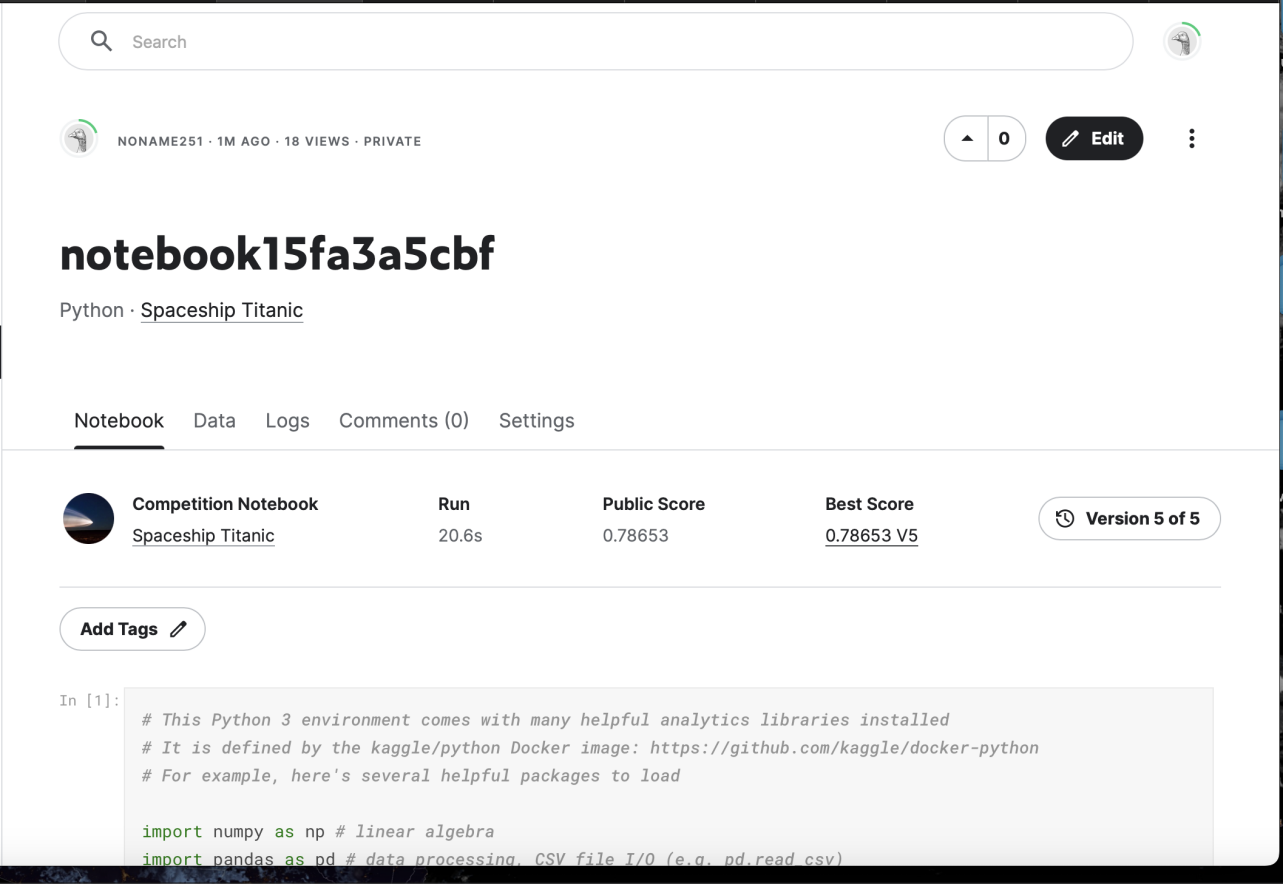
adaboost 准确率：0.7941888619854721, 查准率：0.7800224466891134 查全率：0.8283671036948749



决策树



adaboost

​

‍