一張含有 文字 的圖片

自動產生的描述一張含有 文字 的圖片

自動產生的描述

**Code**

1. import pandas as pd
2. import numpy as np
3. from sklearn.model\_selection import train\_test\_split
4. from sklearn.preprocessing import StandardScaler
5. def missingData(data):
6. print("DataFrame 中是否存在遺失值:",data.isnull().values.any(),"\n")
7. print("每個欄位的遺失值數量:\n",data.isnull().sum())
8. data.dropna(inplace=True)
9. return data
10. def dummyTransformation(data):
11. print("欄位類型：\n",data.dtypes,"\n")
12. # 將以下類別尺度的欄位做轉換
13. colEncodin =['Month', 'OperatingSystems', 'Browser', 'Region', 'TrafficType', 'VisitorType', 'Weekend', 'Revenue']
14. data = pd.get\_dummies(data, columns=colEncodin)
15. print("one-hot-encoding：\n",data)
16. return data
17. def dataCut(data):
18. # 目標欄位Revenue\_True，用於預測顧客是否購買
19. target =["Revenue\_True"]
20. feature = [col for col in data.columns if col != target]
21. # 將資料以7:3的方式切割
22. train\_data, test\_data, train\_target, test\_target = train\_test\_split(data[feature], data[target], test\_size=0.3, random\_state=42)
23. featureScaling(train\_data,test\_data)
24. def featureScaling(train\_data,test\_data):
25. # 創建 StandardScaler 物件
26. scaler = StandardScaler()
27. # 對 train\_data 進行特徵縮放
28. train\_data\_scaled = scaler.fit\_transform(train\_data)
29. print("train\_data 特徵縮放 \n",train\_data\_scaled)
30. # 對 test\_data 進行特徵縮放
31. test\_data\_scaled = scaler.transform(test\_data)
32. print("test\_data 特徵縮放 \n",test\_data\_scaled)
33. return [train\_data\_scaled,train\_data\_scaled]
34. if \_\_name\_\_== "\_\_main\_\_":
35. data = pd.read\_csv("online\_shoppers\_intention.csv")
36. data = missingData(data)
37. data = dummyTransformation(data)
38. dataPreprocessing = dataCut(data)