# 互评作业1: 数据探索性分析与数据预处理

## 1120212298 孙敏豪

## 数据运行时间可见随仓库提交的运行日志文件，下面是日志内容展示，使用的是logging库记录时间戳和执行内容

10g

```
2025-03-31 12:31:41,498 - INFO - None
2025-03-31 12:31:51,692 - INFO -                  id          age       income  credit_score
count 100000000.00 100000000.00 100000000.00  100000000.00
mean      50000.50        59.02    499354.66        575.11
std       28867.51        23.95    289019.13        159.00
min           1.00        18.00         0.00        300.00
25%       25000.75        38.00    249000.00        437.00
50%       50000.50        59.00    499000.00        575.00
75%       75000.25        80.00    750000.00        713.00
max      100000.00       100.00   1000000.00        850.00
2025-03-31 12:32:26,866 - INFO - 数据中不存在缺失值
2025-03-31 12:32:27,066 - INFO - 数据中不存在年龄非法值
2025-03-31 12:35:37,424 - WARNING - 数据中性别列存在异常值，异常值数量: 0
2025-03-31 12:35:37,424 - WARNING - 数据中存在重复行，重复行重复量: 99200000
2025-03-31 12:42:50,358 - INFO - 删除重复行后，数据框中已无重复行。
2025-03-31 12:42:58,673 - INFO - 开始执行 cluster_and_analyze 函数
2025-03-31 12:42:58,673 - INFO - 开始提取 purchase_history 中的 average_price 和 items 的长度
2025-03-31 12:43:03,260 - INFO - 提取完成
2025-03-31 12:43:03,263 - INFO - 选择的特征为: ['income', 'average_price', 'items_count']
2025-03-31 12:43:03,263 - INFO - 开始进行数据标准化
2025-03-31 12:43:03,285 - INFO - 数据标准化完成
2025-03-31 12:43:03,286 - INFO - 最优聚类数 k = 7
2025-03-31 12:43:03,286 - INFO - 开始使用最优聚类数 k = 7 进行 K-means 聚类
2025-03-31 12:43:03,854 - INFO - K-means 聚类完成
2025-03-31 12:43:03,854 - INFO - 开始分析每个聚类的特征
2025-03-31 12:43:03,880 - INFO - 聚类特征分析完成
2025-03-31 12:43:03,880 - INFO - 开始进行 PCA 降维以可视化聚类结果
2025-03-31 12:43:03,907 - INFO - PCA 降维完成
2025-03-31 12:43:03,908 - INFO - 开始绘制散点图可视化聚类结果
2025-03-31 12:43:11,944 - INFO - 将可视化结果保存到 /home/sunminhao/grade8_HOMEWORK/DataMining/Homew
2025-03-31 12:43:57,380 - INFO - 可视化结果保存完成
2025-03-31 12:43:57,380 - INFO - cluster_and_analyze 函数执行结束
2025-03-31 12:43:57,480 - INFO - 每个聚类的特征分析:
2025-03-31 12:43:57,480 - INFO -              income  average_price  items_count
cluster
0       764294.01         774.06         3.67
1       757003.71         246.23         3.48
2       251774.02         257.54         2.96
3       790086.81         485.85         8.42
```

```
4       303637.47         767.36         8.05
5       291742.06         244.93         8.02
6       260681.81         750.20         2.95
```

30g

```
2025-04-02 19:34:24,069 - INFO - None
2025-04-02 19:34:56,425 - INFO -                  id          age       income  credit_score
count 300000000.00 300000000.00 300000000.00  300000000.00
mean      49933.83        59.00    499701.97        575.01
std       28867.44        23.95    288981.11        159.01
min           1.00        18.00         0.00        300.00
25%       24934.00        38.00    249000.00        437.00
50%       49868.00        59.00    499000.00        575.00
75%       74934.00        80.00    750000.00        713.00
max      100000.00       100.00   1000000.00        850.00
2025-04-02 19:36:35,776 - INFO - 数据中不存在缺失值
2025-04-02 19:36:36,362 - INFO - 数据中不存在年龄非法值
2025-04-02 19:50:00,070 - WARNING - 数据中性别列存在异常值，异常值数量: 0
2025-04-02 19:50:00,071 - WARNING - 数据中存在重复行，重复行重复量: 297600000
2025-04-02 20:15:10,335 - INFO - 删除重复行后，数据框中已无重复行。
2025-04-02 20:15:33,292 - INFO - 开始执行 cluster_and_analyze 函数
2025-04-02 20:15:33,292 - INFO - 开始提取 purchase_history 中的 average_price 和 items 的长度
2025-04-02 20:15:46,712 - INFO - 提取完成
2025-04-02 20:15:46,719 - INFO - 选择的特征为: ['income', 'average_price', 'items_count']
2025-04-02 20:15:46,720 - INFO - 开始进行数据标准化
2025-04-02 20:15:46,781 - INFO - 数据标准化完成
2025-04-02 20:15:46,781 - INFO - 最优聚类数 k = 7
2025-04-02 20:15:46,781 - INFO - 开始使用最优聚类数 k = 7 进行 K-means 聚类
2025-04-02 20:15:47,956 - INFO - K-means 聚类完成
2025-04-02 20:15:47,956 - INFO - 开始分析每个聚类的特征
2025-04-02 20:15:48,023 - INFO - 聚类特征分析完成
2025-04-02 20:15:48,023 - INFO - 开始进行 PCA 降维以可视化聚类结果
2025-04-02 20:15:48,100 - INFO - PCA 降维完成
2025-04-02 20:15:48,100 - INFO - 开始绘制散点图可视化聚类结果
2025-04-02 20:16:06,133 - INFO - 将可视化结果保存到 /home/sunminhao/grade8_HOMEWORK/DataMining/Homew
2025-04-02 20:17:46,923 - INFO - 可视化结果保存完成
2025-04-02 20:17:46,923 - INFO - cluster_and_analyze 函数执行结束
2025-04-02 20:17:47,112 - INFO - 每个聚类的特征分析：
2025-04-02 20:17:47,112 - INFO -              income  average_price  items_count
cluster
0       759447.07         290.67         8.00
1       749966.35         257.96         2.98
2       756339.34         755.74         3.34
3       251822.95         269.87         2.93
```

```
4       550493.05         790.89         8.39
5       231234.18         321.39         8.07
6       225116.49         770.87         3.83
```

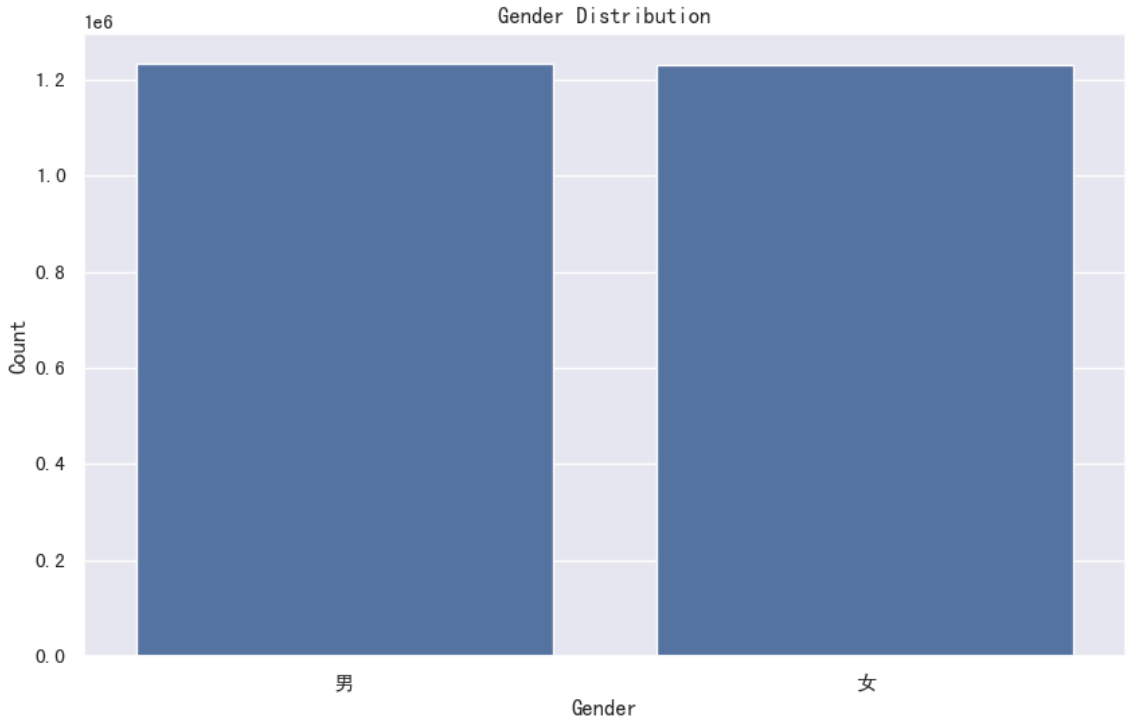
## 30G数据

# 探索性分析和可视化

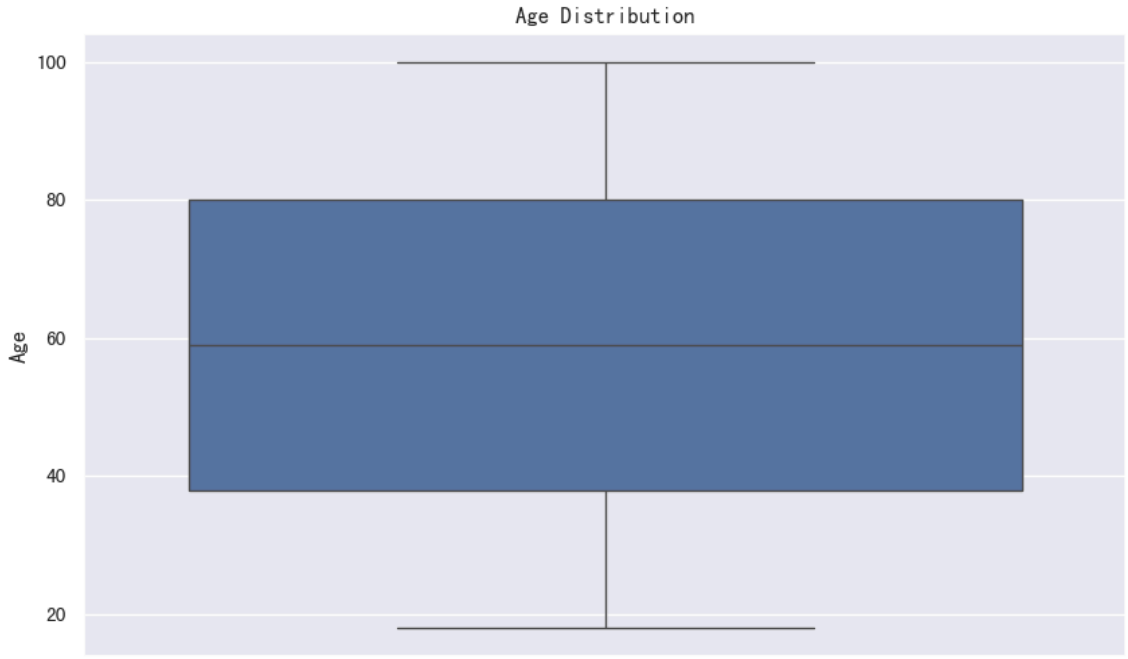
首先，进行了数据基本分析

```
2025-04-02 19:34:56,425 - INFO -                 id          age       income  credit_score
count 300000000.00 300000000.00 300000000.00  300000000.00
mean      49933.83        59.00    499701.97        575.01
std       28867.44        23.95    288981.11        159.01
min           1.00        18.00         0.00        300.00
25%       24934.00        38.00    249000.00        437.00
50%       49868.00        59.00    499000.00        575.00
75%       74934.00        80.00    750000.00        713.00
max      100000.00       100.00   1000000.00        850.00
```

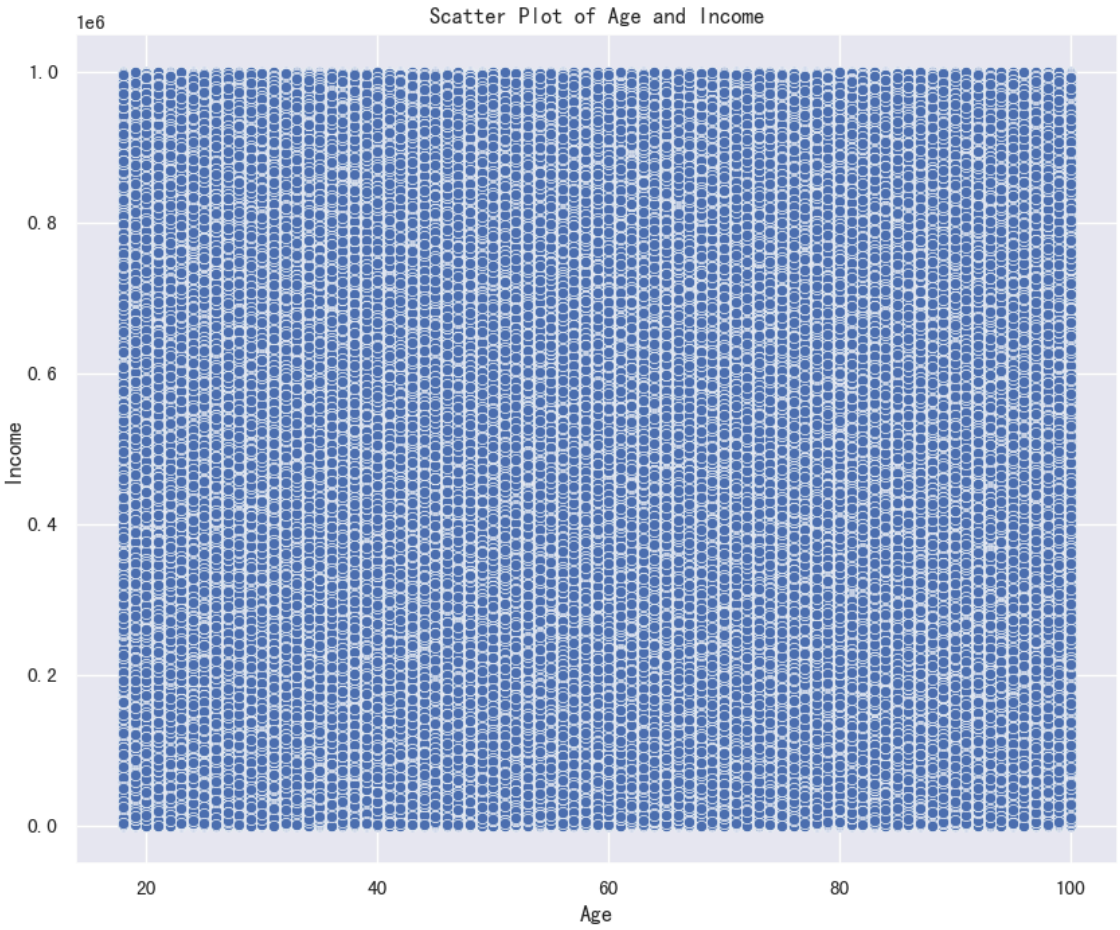
性别分布：通过柱状图展示不同性别的用户数量分布。



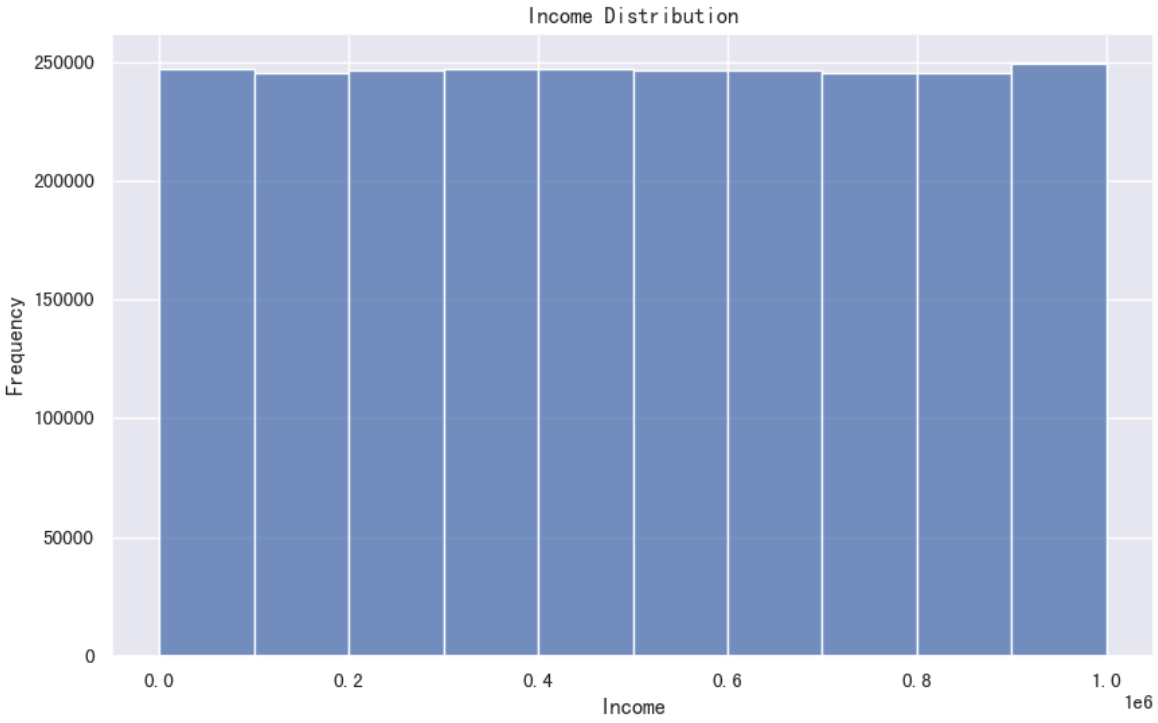

*年龄分布*：采用箱线图展示年龄分布情况，非常均匀。



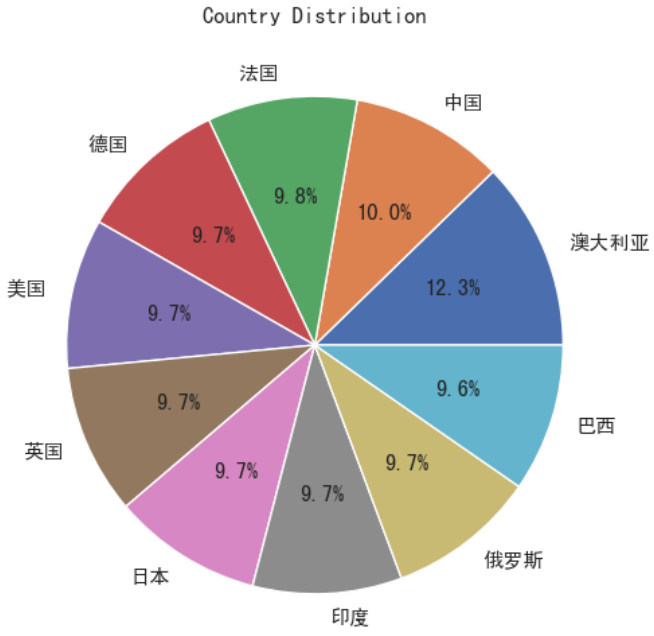

*年龄与收入关系*：使用散点图展示年龄和收入的关系。



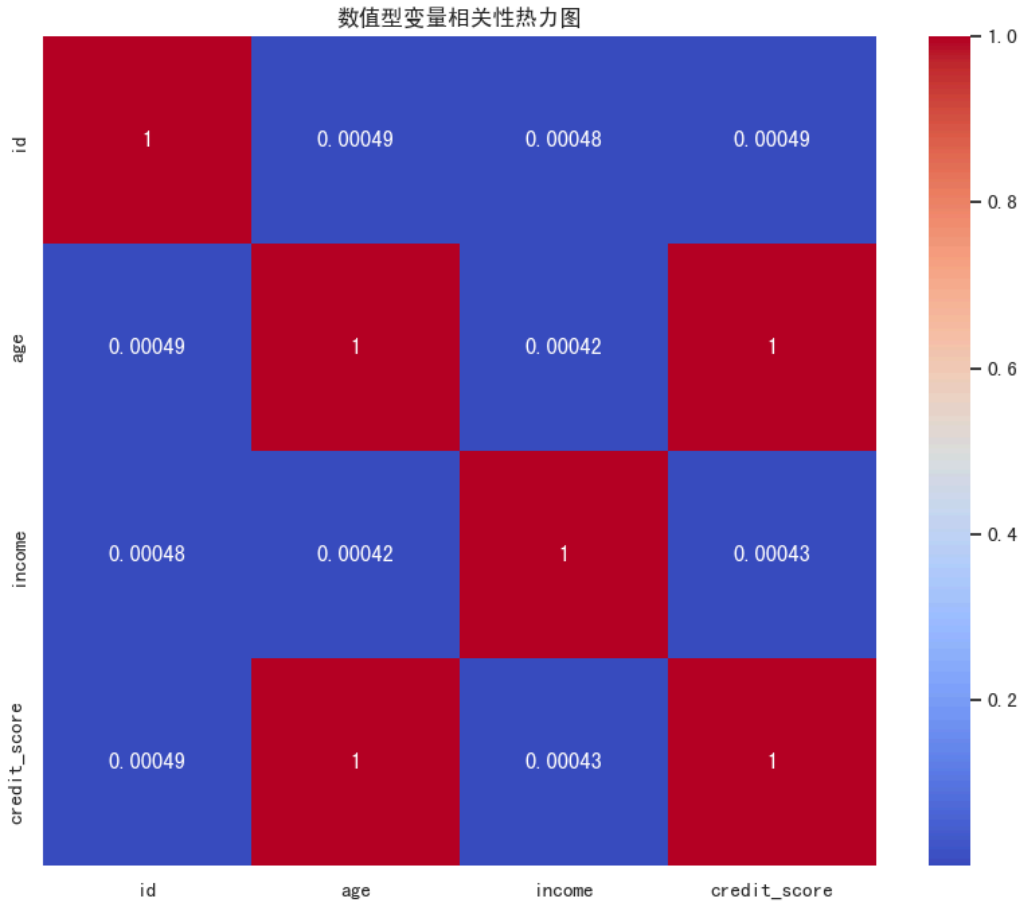

*收入分布*：利用直方图展示收入的分布情况。



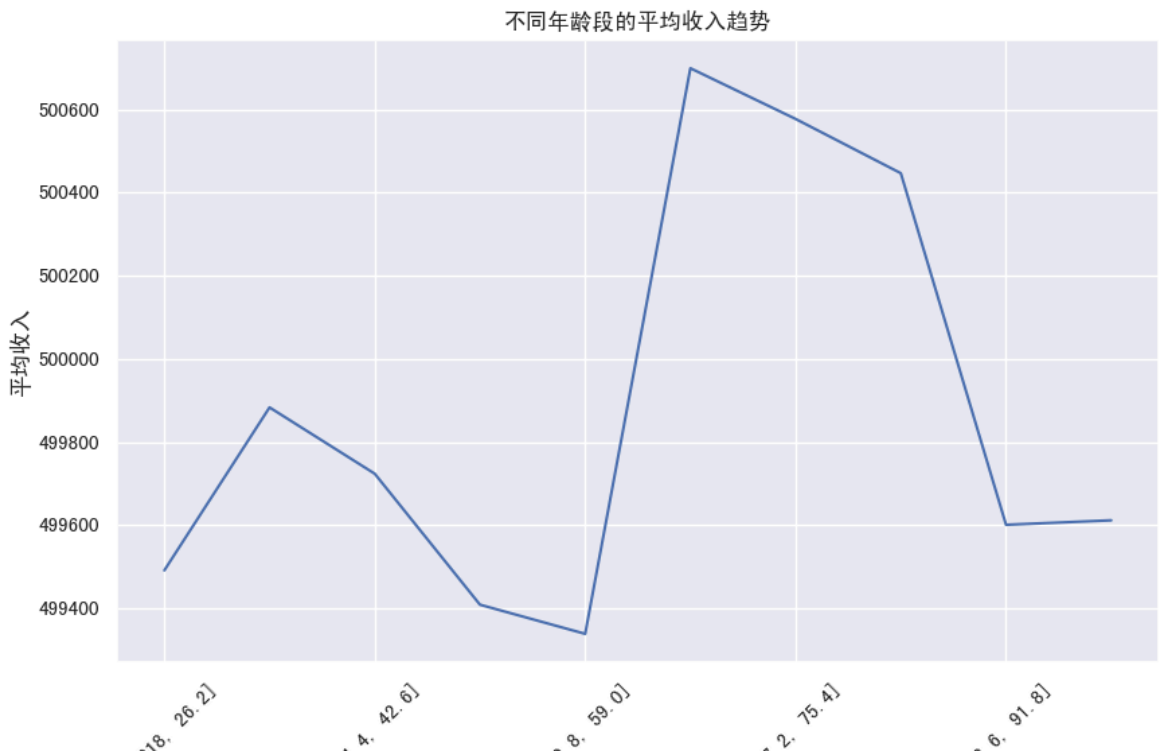

*国家分布*：以饼图呈现不同国家的用户比例。



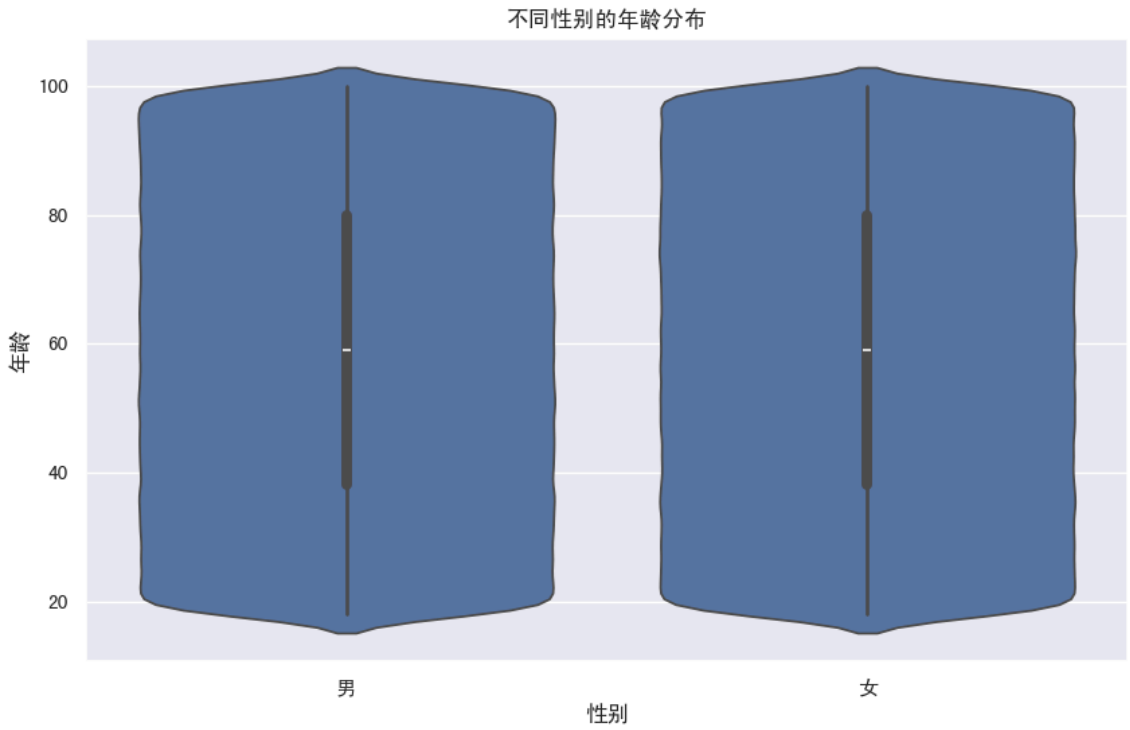

*数值型变量相关性*：显示出社会信用分和年龄的相关性。



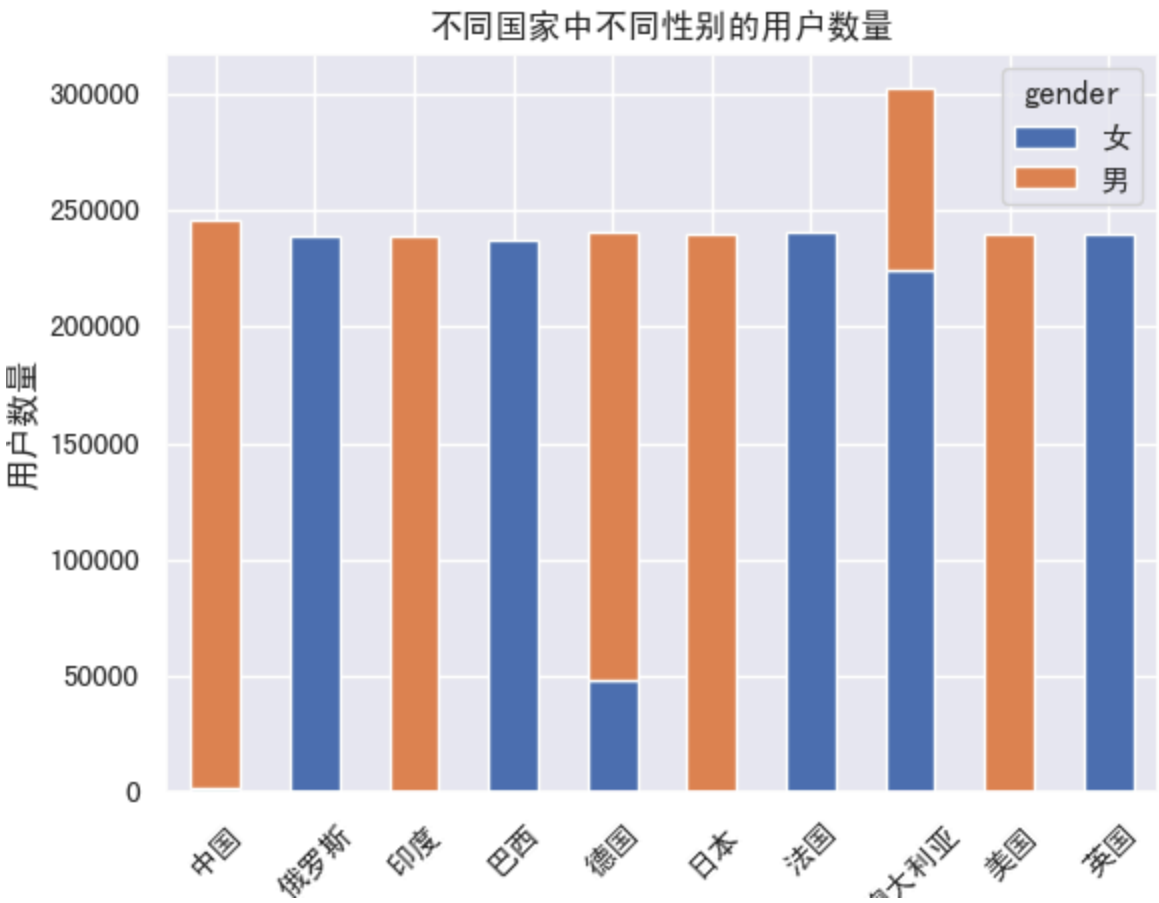

*年龄与平均收入趋势*：采用折线图展示不同年龄段的平均收入趋势。




*性别与年龄分布对比*：利用小提琴图对比不同性别的年龄分布




*国家与性别用户数量*：通过柱状堆积图展示不同国家中不同性别的用户数量。




# 数据预处理

*对数据质量进行评价，发现数据中存在的问题，提出和实现相应的预处理方法。*

*数据质量评价*:
通过evaluate_data_quality函数对数据质量进行评价。该函数检查了数据中的缺失值、异常值和重复数据。
*缺失值检查*:
使用data.isnull().sum()方法统计各列缺失值数量。若存在缺失值，记录相关信息到日志，并将问题添加到issues列表中。
*异常值检查*:
对于年龄列，检查是否存在小于 0 或大于 150 的值；对于性别列，检查是否存在不在['男', '女']范围内的值。若发现异常值，记录问题到issues列表。
*重复数据检查*:
使用data[data.duplicated()]查找重复行，若存在重复行，记录相关信息到issues列表。
*发现的数据问题*:

通过数据质量评价，发现数据中可能存在以下问题：
存在缺失值，可能影响数据分析的准确性和完整性。
性别列存在非法值，即不在预期的['男', '女']范围内。
存在重复行.
预处理方法及实现：针对发现的数据问题，我们提出并实现了以下预处理方法：
*缺失值处理*:
通过handle_missing_values函数，对于数值型列（如age、income、credit_score），使用均值填充缺失值；对于非数值型列，使用众数填充缺失值。代码如下：

```
python
numerical_cols = data.select_dtypes(include=['number']).columns
data[numerical_cols] = data[numerical_cols].fillna(data[numerical_cols].mean())
non_numerical_cols = data.select_dtypes(exclude=['number']).columns
for col in non_numerical_cols:
    mode_value = data[col].mode()[0]
    data[col] = data[col].fillna(mode_value)
```

*异常值处理*:
在handle_outliers函数中，对于年龄列，将小于 0 或大于 150 的异常值修正为合理年龄范围内的平均值；
对于性别列，将非法值随机替换为['男', '女']中的一个值。代码如下：

```
python
valid_age_data = data[(data['age'] >= 0) & (data['age'] <= 150)]['age']
mean_age = valid_age_data.mean()
data.loc[(data['age'] < 0) | (data['age'] > 150), 'age'] = mean_age
valid_genders = ['男', '女']
data.loc[~data['gender'].isin(valid_genders), 'gender'] = [random.choice(valid_genders) for _ in r
```

*重复值处理*:
handle_duplicated_rows函数使用data.drop_duplicates()方法删除数据中的重复行，并检查删除后是否还存在重复行。若不存在，记录相关信息到日志。代码如下：

```
python
data = data.drop_duplicates()
remaining_duplicated_mask = data.duplicated(keep=False)
remaining_duplicated_rows = data[remaining_duplicated_mask]
if remaining_duplicated_rows.empty:
    logging.info("删除重复行后，数据框中已无重复行。")
```

# 分析目标

*建立用户画像*

*K - means 用于用户画像*

K - means 聚类算法在用户画像构建中扮演着关键角色，通过对用户数据的深入挖掘和分析，能够将具有相似特征的用户划分到同一聚类中，从而清晰地展现不同用户群体的特征，为精准的用户画像提供有力支持。

1. 特征工程

   1.1 数据提取与转换

   在构建用户画像的过程中，需要从原始数据中提取有价值的特征。代码中从purchase_history字段提取average_price和items_count的操作是关键步骤。

```
python
def parse_purchase_history(x):
    parsed = json.loads(x.replace('""', '"'))
    return parsed['average_price'], len(parsed['items'])
data['average_price'], data['items_count'] = zip(*data['purchase_history'].apply(parse_purchase_hi
```

1.2 多维度特征选择

除了从purchase_history提取的特征外，结合income（收入）字段，构建了一个多维度的特征空间。选择这些特征的原因在于，收入水平往往决定了用户的消费能力，而平均购买价格和购买物品数量则直接反映了用户的消费行为。例如，高收入用户可能倾向于购买价格更高的商品，或者购买更多数量的商品。

```
python
features = data[['income', 'average_price', 'items_count']]
通过上述代码，从原始数据data中选取了income、average_price和items_count这三个特征，组成了用于 K - means
```

2. 数据标准化
   由于不同特征的量纲和取值范围可能存在较大差异，为了避免某些特征对聚类结果产生过大影响，需要对数据进行标准化处理。在本案例中，使用StandardScaler对特征数据进行标准化。

```
python
scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)
```

StandardScaler将数据按照特征进行标准化，使其符合标准正态分布，即均值为 0，标准差为 1。这样处理后，所有特征在聚类算法中的权重更加均衡，能够提高聚类结果的准确性和稳定性。

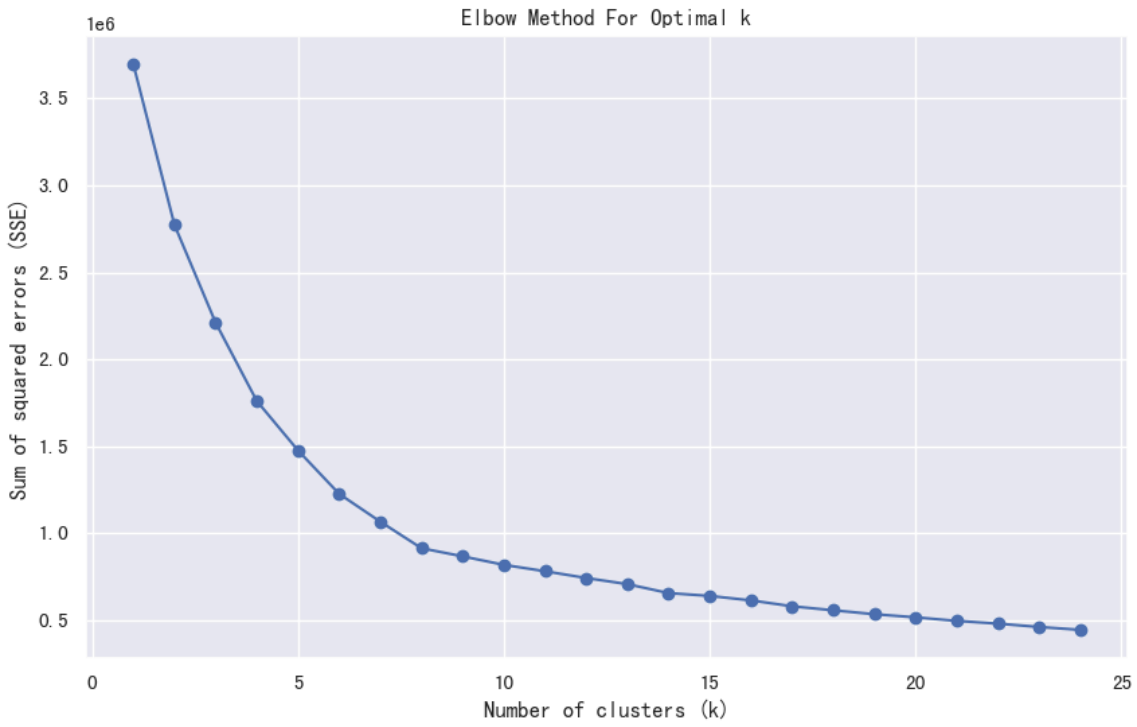
3. 确定最优聚类数

确定合适的聚类数对于 K - means 聚类的效果至关重要。这里采用肘部图的方法。

```
python
    for k in k_range:
        logging.info(f"开始尝试聚类数 k = {k}")
        kmeans = KMeans(n_clusters=k, random_state=42)
        kmeans.fit(features_scaled)
        sse.append(kmeans.inertia_)
        logging.info(f"完成聚类数 k = {k} 的尝试，当前 SSE: {kmeans.inertia_}")

    # 绘制 SSE 随聚类数变化的曲线
    plt.plot(k_range, sse, marker='o')
    plt.xlabel('Number of clusters (k)')
    plt.ylabel('Sum of squared errors (SSE)')
    plt.title('Elbow Method For Optimal k')
    elbow_file_path = os.path.join(save_path, 'Elbow_cluster_visualization.png')
    plt.savefig(elbow_file_path)
```

根据肘部图




采用7个类

4. 进行 K - means 聚类

在确定了最优聚类数best_k后，使用KMeans算法对标准化后的特征数据进行聚类。

```
python
kmeans = KMeans(n_clusters=best_k, random_state=42)
data['cluster'] = kmeans.fit_predict(features_scaled)
```

这里创建了一个KMeans模型，完成聚类。

5. 分析聚类特征

聚类完成后，对每个聚类的特征进行分析，以深入了解不同用户群体的特点。这里计算了每个聚类中income、average_price、items_count的均值。

```
python
cluster_analysis = data.groupby('cluster').agg({
    'income': 'mean',
    'average_price': 'mean',
    'items_count': 'mean',
})
```

6. 可视化聚类结果
   为了更直观地展示聚类结果，使用PCA将数据降至二维，并通过散点图进行可视化。

```
python
pca = PCA(n_components=2)
features_pca = pca.fit_transform(features_scaled)
data['pca_1'] = features_pca[:, 0]
data['pca_2'] = features_pca[:, 1]
sns.scatterplot(data=data, x='pca_1', y='pca_2', hue='cluster')
save_path = '/home/sunminhao/grade8_HOMEWORK/DataMining/Homework1/visualization'
if not os.path.exists(save_path):
    os.makedirs(save_path)
file_path = os.path.join(save_path, 'cluster_visualization.png')
plt.savefig(file_path)
plt.show()
```

7. 用户画像分析
   根据聚类结果图，可以看到，较好的区分了各个类

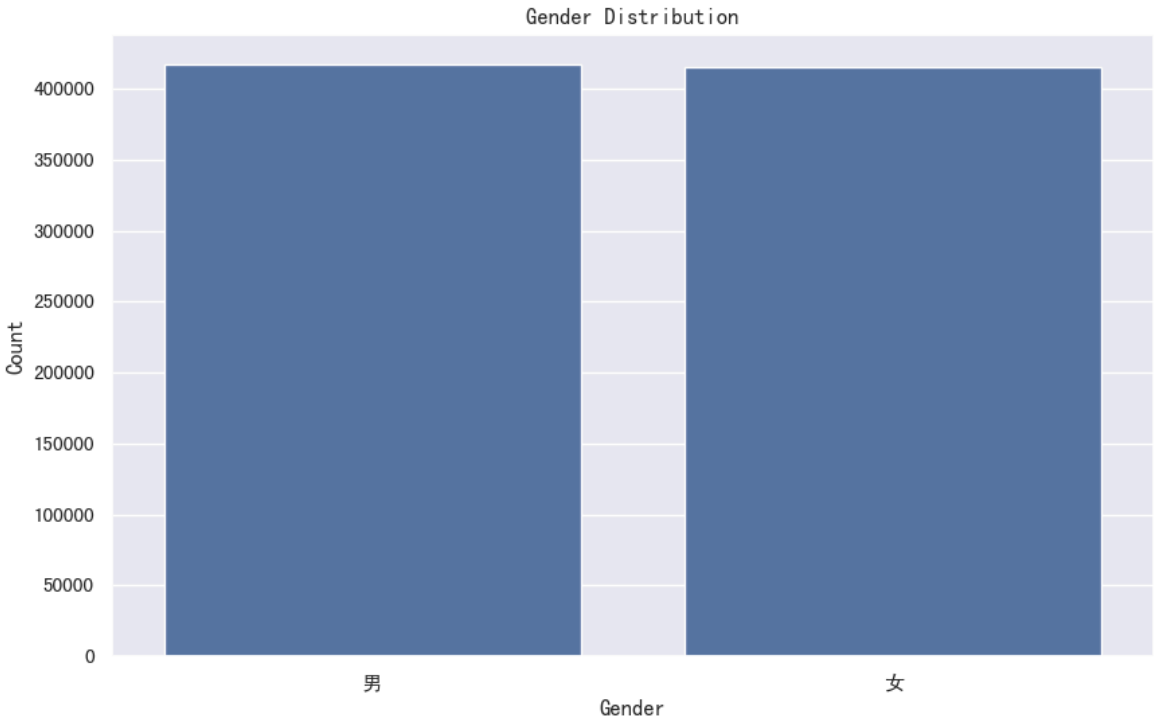

以及聚类的分析结果

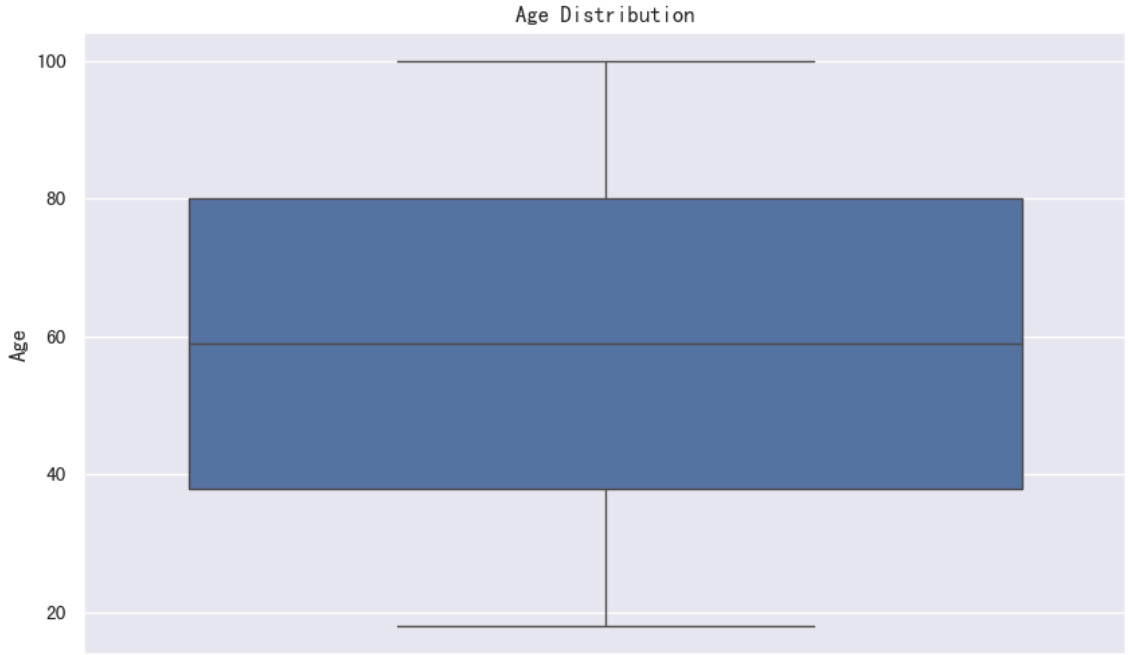
```
2025-03-31 12:43:57,480 - INFO -            income  average_price  items_count
cluster
0        764294.01         774.06         3.67
1        757003.71         246.23         3.48
2        251774.02         257.54         2.96
3        790086.81         485.85         8.42
4        303637.47         767.36         8.05
5        291742.06         244.93         8.02
6        260681.81         750.20         2.95
```

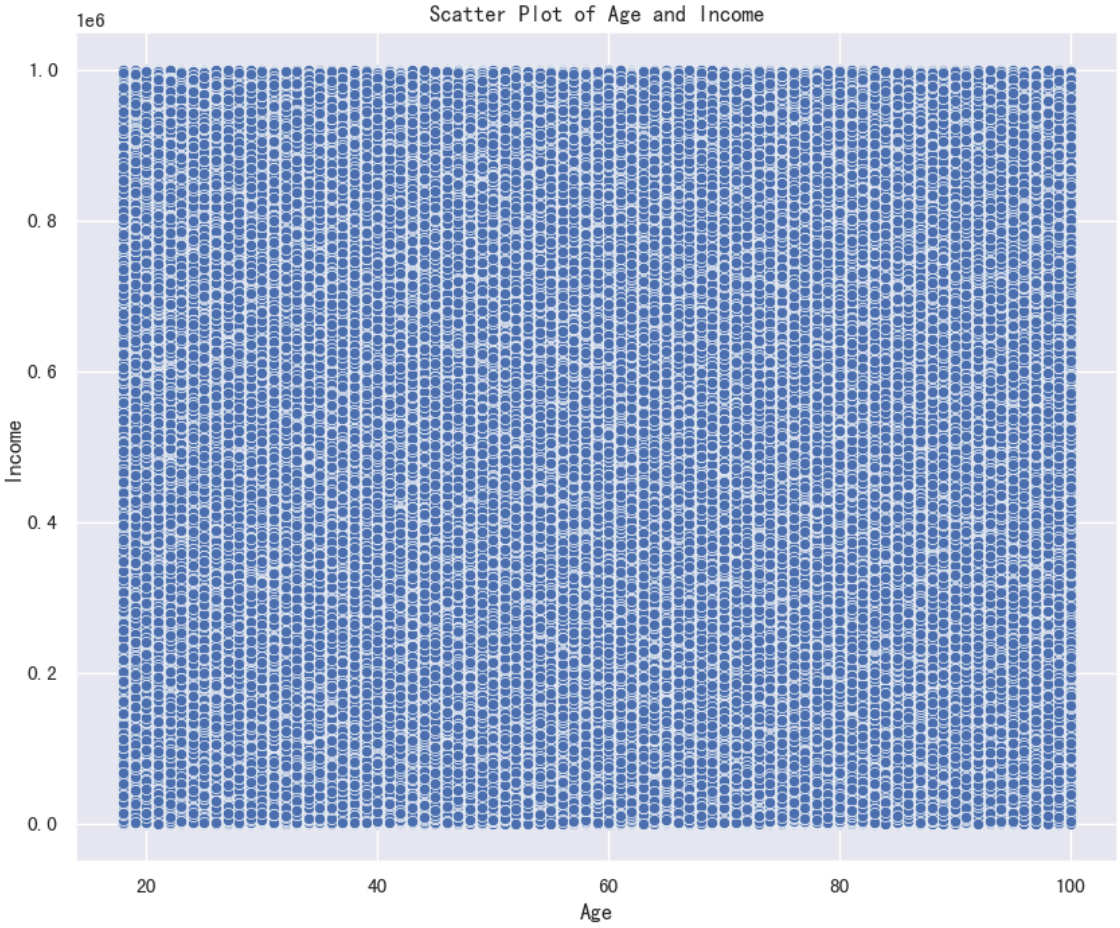
可以进行如下用户画像

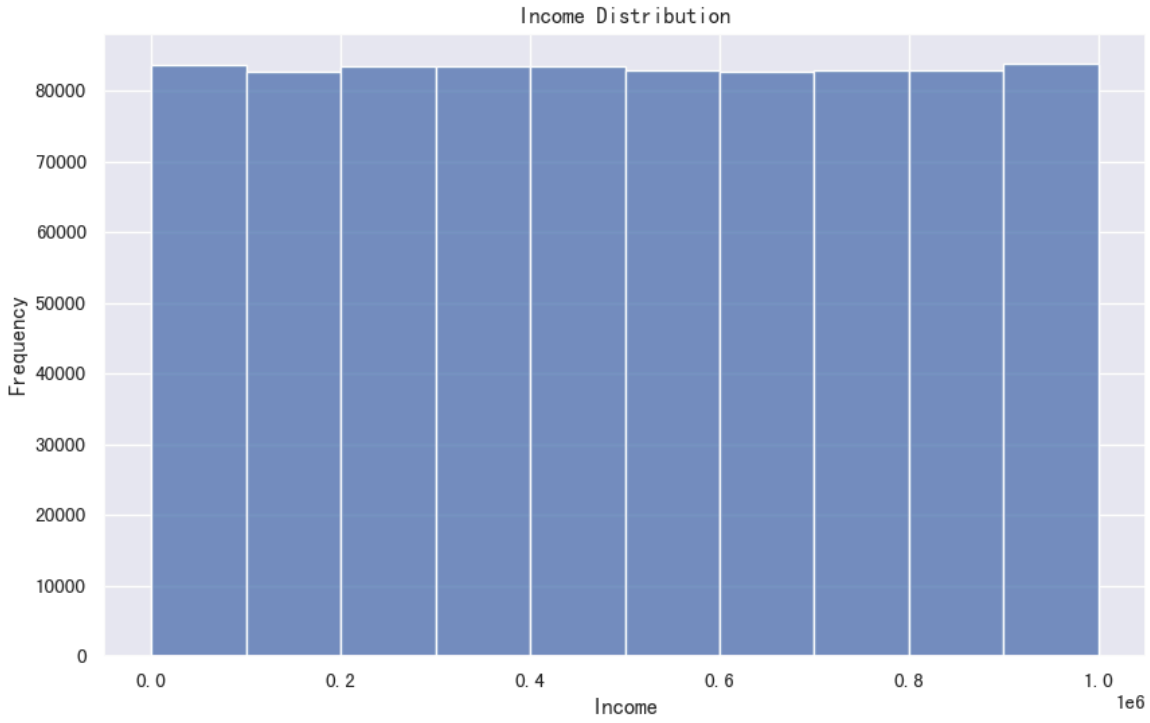
聚类 0
收入：较高，平均达到 764294.01 。
平均价格：774.06 ，相对较高，说明倾向购买中高价位商品。
订单商品数量：3.67 ，不算高。
用户画像：高收入群体，有较强消费能力，偏好中高价位商品，但每次订单购买商品数量不算多，可能注重品质、追求品牌
聚类 1
收入：757003.71 ，属于高收入。
平均价格：246.23 ，较低。
订单商品数量：3.48 ，较少。
用户画像：高收入但消费较为节俭，可能对价格敏感，倾向购买性价比高的商品，购物频次可能不高，注重实用性，不追求
聚类 2
收入：251774.02 ，收入较低。
平均价格：257.54 ，较低。
订单商品数量：2.96 ，较少。
用户画像：低收入群体，消费能力有限，注重商品价格，购买商品数量少，可能在购物时会反复比较，选择价格低廉且必需
聚类 3
收入：790086.81 ，高收入。
平均价格：485.85 ，中等水平。
订单商品数量：8.42 ，较多。
用户画像：高收入且购物较频繁，对价格敏感度低，可能喜欢批量采购，消费需求广泛，可能是家庭采购主力，注重商品实
聚类 4
收入：303637.47 ，中等收入。
平均价格：767.36 ，较高。
订单商品数量：8.05 ，较多。
用户画像：中等收入但愿意为喜欢的商品支付较高价格，可能对某些品类有偏好，购物频次较高，有一定消费追求，可能是
聚类 5
收入：291742.06 ，中等偏低收入。
平均价格：244.93 ，较低。
订单商品数量：8.02 ，较多。
用户画像：收入不高但购物频次高，倾向购买低价商品，可能善于寻找优惠、折扣，是价格敏感型消费者，注重性价比，通
聚类 6
收入：260681.81 ，低收入。
平均价格：750.20 ，较高。
订单商品数量：2.95 ，较少。
用户画像：低收入但偶尔会购买高价位商品，可能在某些特定品类上有较高消费追求，不追求购物数量，更在意购买到符合

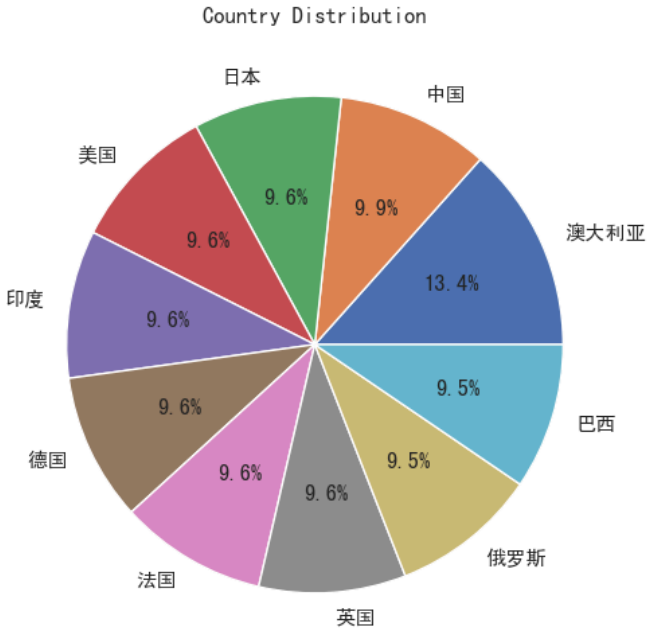
# 10G数据分析过程和上述大同小异 因此仅展示图片

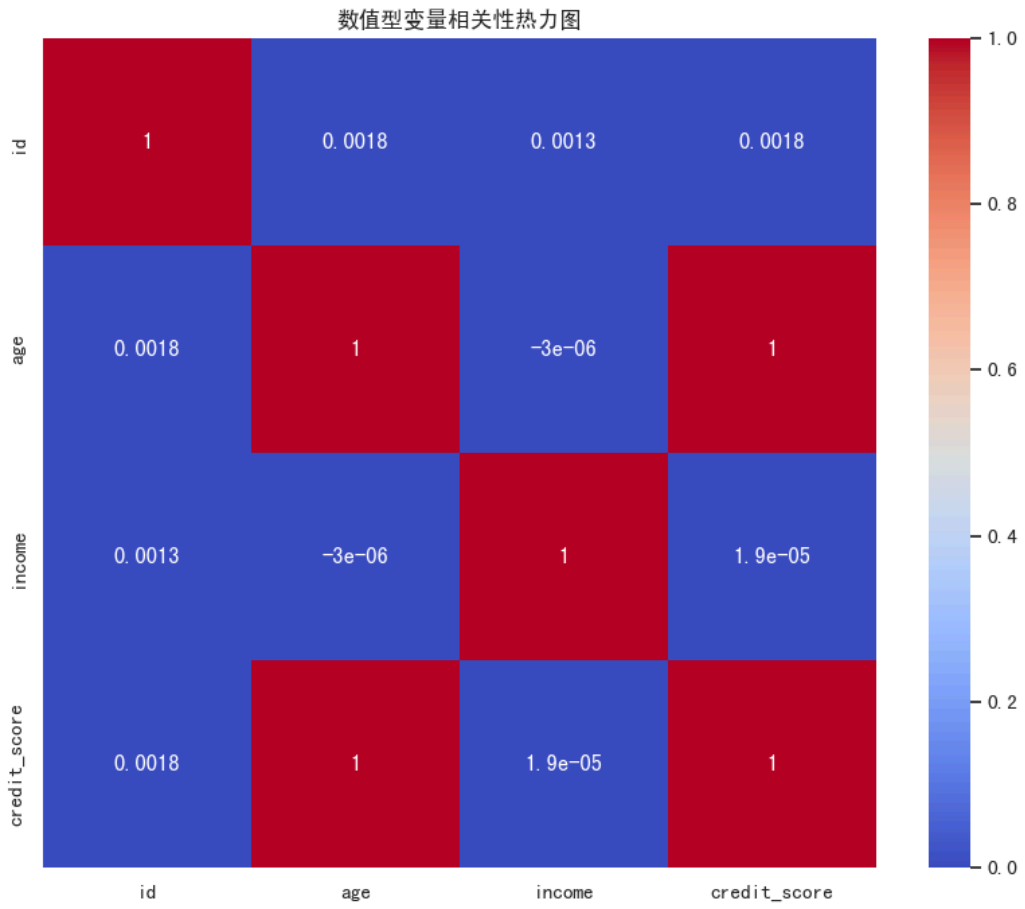


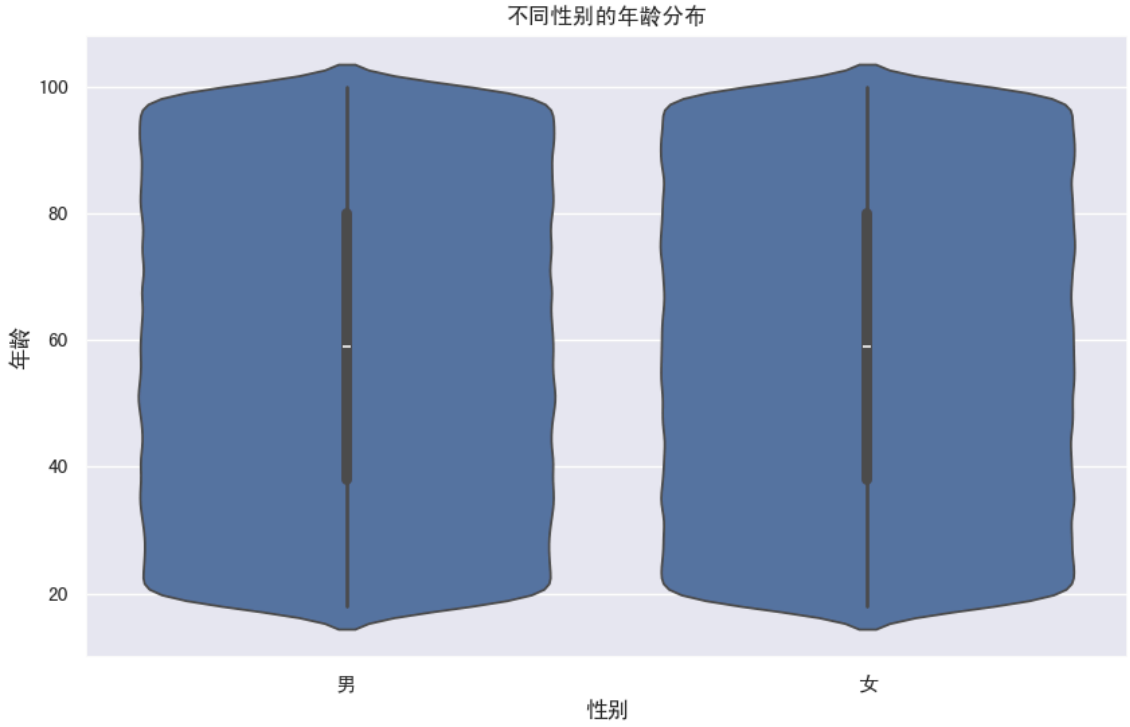

Scatter Plot of Age and Income
1e6
1.0
0.8
0.6
0.4
0.2
0.0
Income
20
40
60
80
100
Age

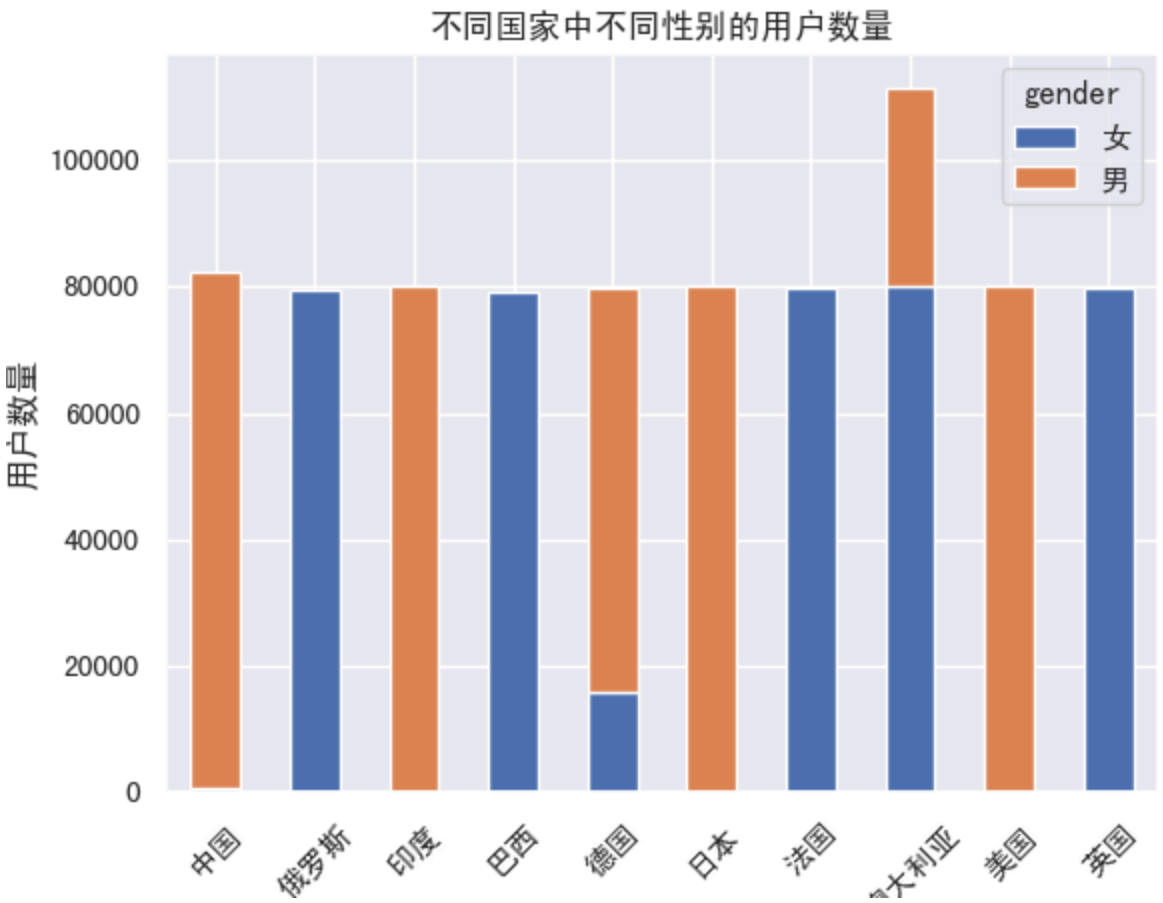

Income Distribution
Frequency
80000
70000
60000
50000
40000
30000
20000
10000
0
0.0
0.2
0.4
0.6
0.8
1.0
1e6
Income


Country Distribution
日本
中国
美国
澳大利亚
印度
巴西
德国
俄罗斯
法国
英国
9.6%
9.9%
9.6%
13.4%
9.6%
9.5%
9.6%
9.5%
9.6%
9.6%


数值型变量相关性热力图
id
age
income
credit_score
1
0.0018
0.0013
0.0018
0.0018
1
-3e-06
1
0.0013
-3e-06
1
1.9e-05
0.0018
1
1.9e-05
1
id
age
income
credit_score
1.0
0.8
0.6
0.4
0.2
0.0


不同年龄段的平均收入趋势
平均收入
501500
501000
500500
500000
499500
499000
498500
18, 26.2]
4, 42.6]
8, 59.0]
2, 75.4]
6, 91.8]


不同性别的年龄分布
年龄
100
80
60
40
20
男
女
性别


不同国家中不同性别的用户数量
gender
女
男
用户数量
100000
80000
60000
40000
20000
0
中国
俄罗斯
印度
巴西
德国
日本
法国
美国
英国