Report

数据集的选择：

在老师给出的三个数据集中，我选择了谷歌商店数据集。该数据集共10000+个样本，每个样本由有以下元素构成：

App	Category	Rating	Reviews	Size	Installs	Type	Price

Content Rating	Genres	Last Updated	Current Ver	Android Ver

数据集的处理：

原数据集中有缺失数据（即部分元素为NaN）的数据，通过python处理，使用dropna()函数去除含有NaN的行。

对App列进行排序后发现，原数据集中有很多重复样本，通过Excel的去除重复数据功能将重复元素删除。

对不同的列进行排序，去除了不合理的数据（如Rating列大于5的数据等）

最终处理后的数据集由原本的10000多个样本缩减为8000多个样本。

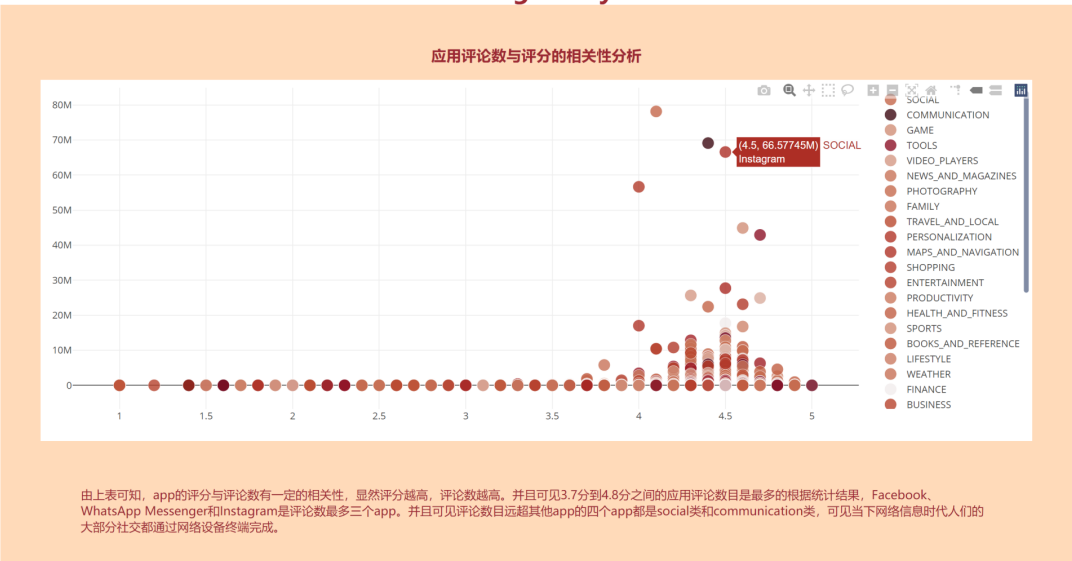
DashBoard

应用评论数与评分的相关性分析

图表效果

横坐标为应用评级（0-5），纵坐标为应用的评价数量。右边栏通过应用分类为散点图展示了不同的颜色。鼠标悬浮在某个散点上不仅可以显示该坐标的xy值，还设计显示了该散点对应的应用名称和对应类别。

Lab3: Google Play Store



图表分析

由上表可知，app的评分与评论数有一定的相关性，显然评分越高，评论数越高。并且可见3.7分到4.8分之间的应用评论数目是最多的根据统计结果，Facebook、WhatsApp Messenger和Instagram是评论数最多三个app。并且可见评论数目远超其他app的四个app都是social类和communication类，可见当下网络信息时代人们的大部分社交都通过网络设备终端完成。

代码

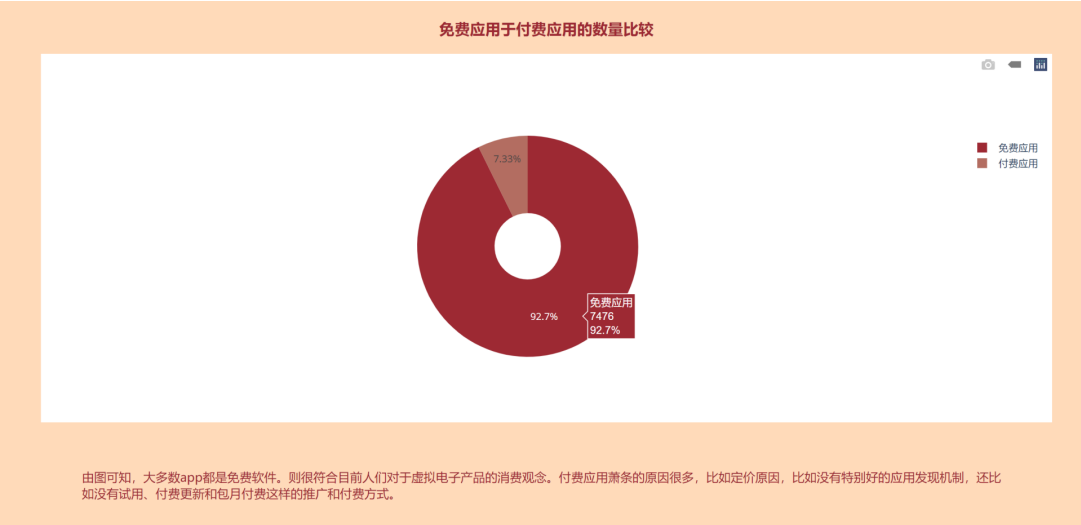
```
def get_first_pic():
    figure = dict(
        data=[
            go.Scatter(
                x=df[df['Category'] == i]['Rating'],
                y=df[df['Category'] == i]['Reviews'],
                text=df[df['Category'] == i]['App'],
                name=i,
                mode='markers',
                opacity=0.8,
                marker=dict(size=15, color=np.random.randn(400),
colorscale='amp', line=dict(width=0.5, color='white'))
            ) for i in df.Category.unique()],
        layout=go.Layout(
            margin={'l': 40, 'b': 40, 't': 10, 'r': 10},
            hovermode='closest'
        )
    )

    return figure
```

免费应用于付费应用的数量比较

图表效果

饼图，展示了免费应用和付费应用的占比。



图表分析

由图可知，大多数app都是免费软件。则很符合目前人们对于虚拟电子产品的消费观念。付费应用萧条的原因很多，比如定价原因，比如没有特别好的应用发现机制，还比如没有试用、付费更新和包月付费这样的推广和付费方式。

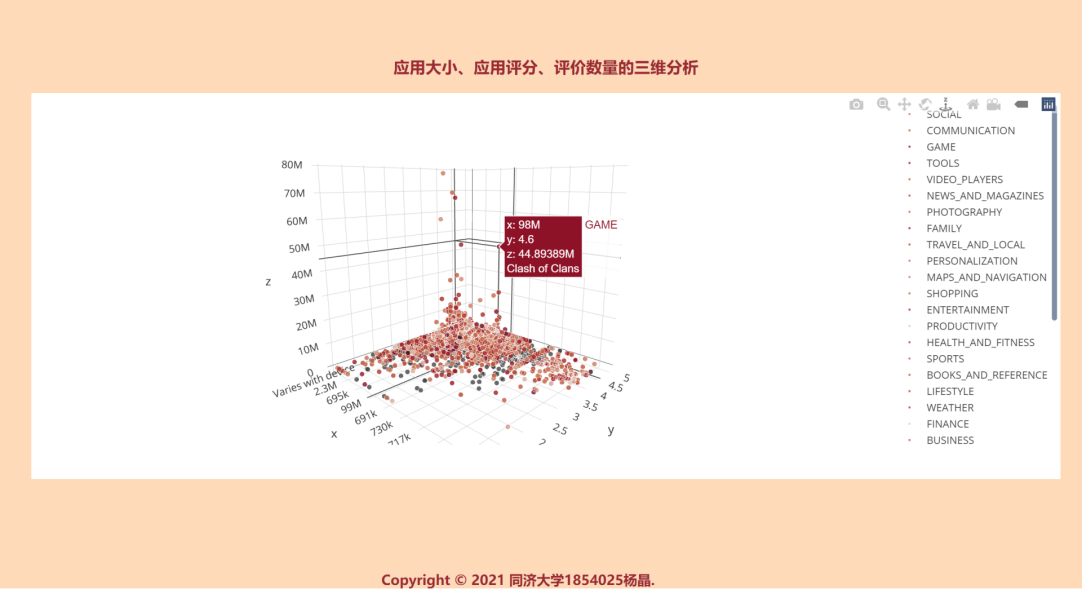
代码

```
def get_second_pic():
    free_count = len(df[df['Type'] == 'Free'])
    paid_count = len(df[df['Type'] == 'Paid'])
    labels = ['免费应用', '付费应用']
    values = [free_count, paid_count]
    colors = ['#9d2933', '#b36d61']
    trace = [go.Pie(labels=labels, values=values,
marker=dict(colors=colors), hole=.3)]
    fig = go.Figure(data=trace)
    return fig
```

应用大小、应用评分、评价数量的三维分析

图表效果

x坐标为应用大小，y坐标为应用的评级，z坐标为评价数量。右边栏通过应用分类为散点图展示了不同的颜色。鼠标悬浮在某个散点上不仅可以显示该坐标的xyz值，还设计显示了该散点对应的应用名称和对应类别。



图表分析

由上述三维图可见，大部分应用的评价数量集中在10M一下，评分集中在2.5以上，应用大小在20M+。同时我们可以发现，评分和评论数都很卓越的几个应用的大小都是“Varies with device”。

代码

```
def get_third_pic():
    figure = dict(
        data=[
            go.Scatter3d(
                x=df[df['Category'] == i]['Size'],
                y=df[df['Category'] == i]['Rating'],
                z=df[df['Category'] == i]['Reviews'],
                text=df[df['Category'] == i]['App'],
                name=i,
                mode='markers',
                opacity=0.8,
                marker=dict(size=3, color=np.random.randn(400), colorscale='amp', line=dict(width=0.5, color='white'))
            ) for i in df.Category.unique()],
        layout=go.Layout(
            margin={'l': 40, 'b': 40, 't': 10, 'r': 10},
            hovermode='closest'
        )
    )

    return figure
```