Lab3 - Data Visualization

Name	ID
Yiteng Zhang(张艺腾)	1852137

Lab3 - Data Visualization

1. Data analysis
2. Design
    - 2.1 'Each Category's Installs' - Bar Graph
        - Data cleaning and data processing
        - Draw the table
    - 2.2 Scatter Graph
        - Installs-Rating
        - Reviews-Rating
    - 2.3 'Each Rate's Apps' Count ' - Bar Graph
        - Data processing
        - Draw the graph

1. Data analysis

Objectives

Among these different data sets, the most relevant to our profession is the data of Google app. In addition, I don't really understand the various attributes of the other data sets very well, so I finally chose Google's data set.

The relationship between the data I want to display is as follows:

- The relationship between the number of `Installs` and `Rating`
- The relationship between `Total Installs` and `Categories`
- The relationship between `Rating` and `Total App Count`
- The relationship between the number of `Reviews` and `Rating`

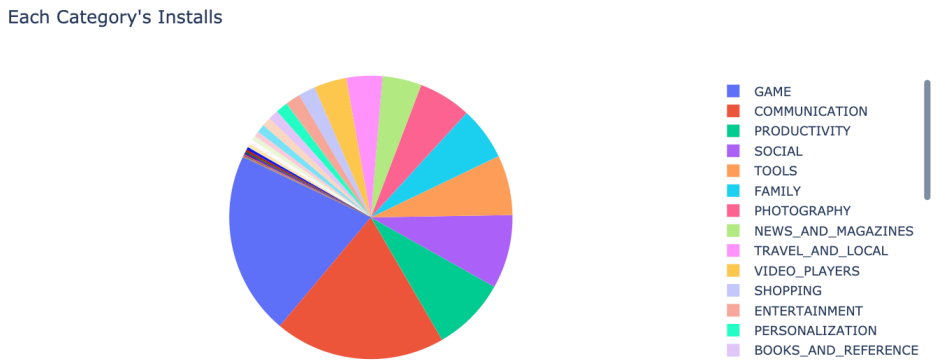
Characteristics

- The most import value for each app is the install amount.
- Review is proportional to Rating.
- Each category has a different proportion of the app market
- The number of apps and reviews are normally distributed.

2. Design

2.1 'Each Category's Installs' - Bar Graph

- by hovering on each sector, the scatter graphs on the right side can show the certain category's data



Data cleaning and data processing

```
# calculate each category's apps' install times
installs = []
for n in name:
    total = 0
    # print(n)
    dff = df[df['Category'] == n]['Installs']
    for d in dff:
        d = d[0:-1]
        d = d.replace(',', '')
        d = int(d)
        total += d
    # print(total)
    installs.append(total)
```

Draw the table

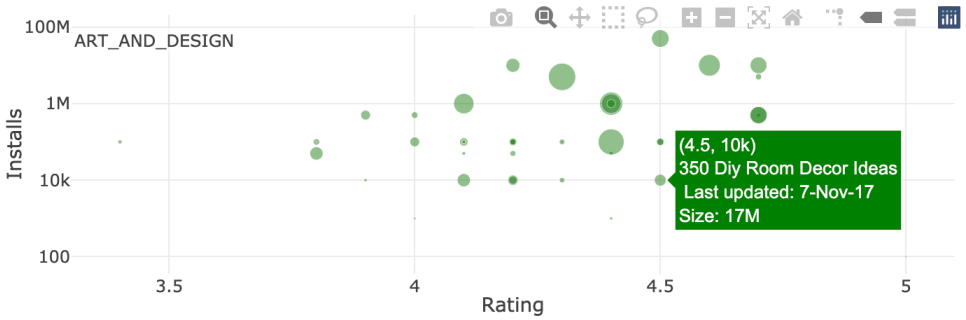
```
# draw the category-install Pie graph
categoryInstallPie = go.Figure(data=go.Pie(
    labels=name,
    values=installs,
    hoverinfo='label+value+percent',
    textinfo='none',
    rotation=220,
    customdata=name
),
    layout=go.Layout(
        title='Each Category\'s Installs'
    )
)
```

2.2 Scatter Graph

- You can change the display mode by select in the checkbox

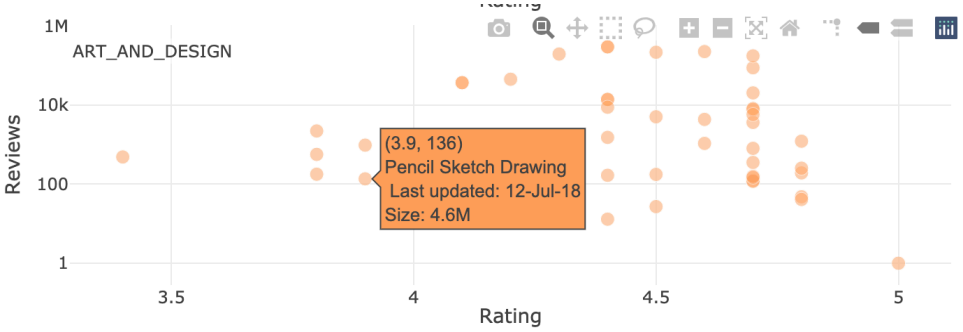
Installs-Rating

- By hovering on each point, it can show the detail information.
- The size of each point means the size of the app.



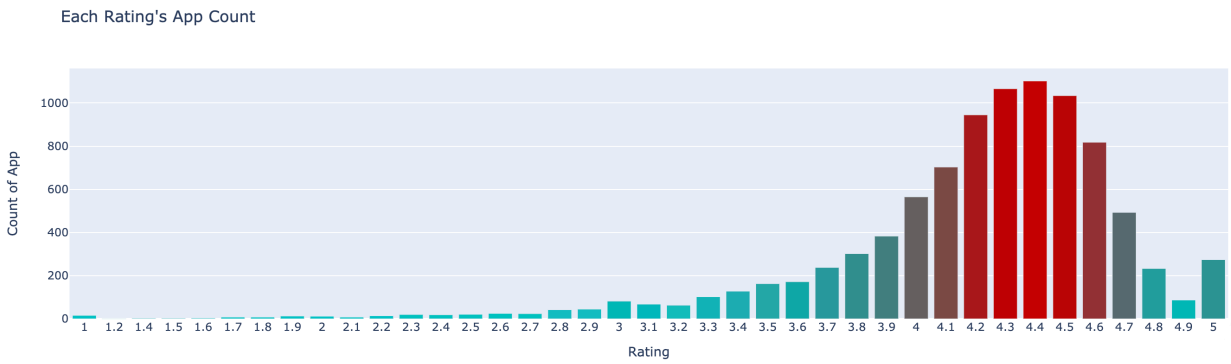
Reviews-Rating

- By hovering on each point, it can show the detail information.



2.3 'Each Rate's Apps' Count ' - Bar Graph

- By hovering on each point, it can show the detail information.



Data processing

```
# calculate the count of each rating's apps
rateCount = df.Rating.value_counts()
maxCount = max(rateCount)
rateCount = {'Rating': rateCount.index, 'Count': rateCount.values}
dfRateCount = pd.DataFrame(rateCount)
dfRateCount = dfRateCount.sort_values(by="Rating", ascending=True)
```

Draw the graph

```
def get_color(temp):
    return 'rgb({r}, {g}, {b})'.format(
        r=int(temp/maxCount*200),
        g=int((1-temp/maxCount)*200),
        b=int((1-temp/maxCount)*200)
    )

# draw the category-count Bar graph
barFig = go.Figure(
    data=go.Bar(
        x=dfRateCount.Rating,
        y=dfRateCount['Count'].astype(int),
        customdata=dfRateCount.Count,
        marker={
            'color': [get_color(count) for count in dfRateCount['Count'].astype(int)]
        }
    ),
    layout=go.Layout(
        yaxis={
            'title': 'Count of App',
        },
        xaxis={
            'title': 'Rating',
        },
        title='Each Rating\'s App Count',
    )
)
```