# Lab 3 Data Visualization

## 1. Dataset Introduction

This project is based on the dataset Google Play Store Apps, which contains varieties of apps available in Google Play Store, along with other information about them, such as the categories they belong to, the rating score and the number of reviews they have got and the content rating they belong to.

The reason why we analyze this dataset is that the dataset may imply certain characteristics in certain categories of apps, which is helpful in searching for potential in phone apps and making business decision.

## 2. Project introduction and demonstration

### 1) Project introduction

This project is based on a package for data visualization in python: dash. Along with other packages like plotly and pandas, it is convenient to draw different types of diagrams clearly with the dataset. In this project, I utilized four types of diagrams to display characteristics of the dataset: pie chart, bar chart, scatter chart and line chart.

The project is made up of three main components: the dataset, which is stored in googleplaystore.csv; util.py, which provides functions for the project to draw diagrams; dashboard.py, which reads and processes data and eventually display results.

In util.py, first we need to import graph_objs from plotly in order to draw and return diagrams. Then there are four functions, each of which is responsible for drawing a certain type of diagram. For example, the following code in pic 1 is called when we need to draw a line chart.

In dashboard.py, apart from dash, we also need to import numpy, which is used to process data, and import util to draw diagrams. After that, we process data and do preparations, and design the layout of the whole page. When it comes to display, all we need to do is to process the data in the right way and call functions in util, shown in pic 2.

Finally, run dashboard.py, and since it is based on flask, we can easily access the dashboard page on http://localhost:8050/.

```python
# 折线图
1个用法
def draw_line(x_values, x_title, y_values, y_title, title):
    return {
        'data': [{
            'x': x_values,
            'y': y_values,
            'mode': 'lines+markers',
            'type': 'Scatter',
            'name': 'Line',
            'line': dict(color='"rgb(153,204,102)"')
        }],
        'layout': go.Layout(
            xaxis={
                'title': x_title
            },
            yaxis={
                'title': y_title
            },
            title=title,
            margin={
                'l': 130,
                'r': 40,
                't': 50,
                'b': 50
            },
            height=500,
            hovermode='closest'
        )
    }
```

Pic 1. Code called when drawing a line chart

```python
@app.callback(
    dash.dependencies.Output('installs-rating', 'figure'),
    [
        dash.dependencies.Input('Category', 'value'),
        dash.dependencies.Input('Type', 'value'),
        dash.dependencies.Input('rating-radio', 'value')
    ]
)
def refresh_installs_rating(category, type_, radio):
    dff = df[(df['Category'] == category) & (df['Type'] == type_) & (df['Content Rating'] == radio)]

    installs = [0, 0, 0, 0, 0, 0]
    ratings = ['0~1', '1~2', '2~3', '3~4', '4~5', '5+']

    for index, row in dff.iterrows():
        installs[int(float(row['Rating']))] += int(row['Installs'][:-1].replace(",", ""))

    return draw_bar(ratings, 'Rating', installs, 'Installs', 'Installs-Rating Diagram (Bar)')
```

Pic 2. Example for calling functions in util: draw_bar

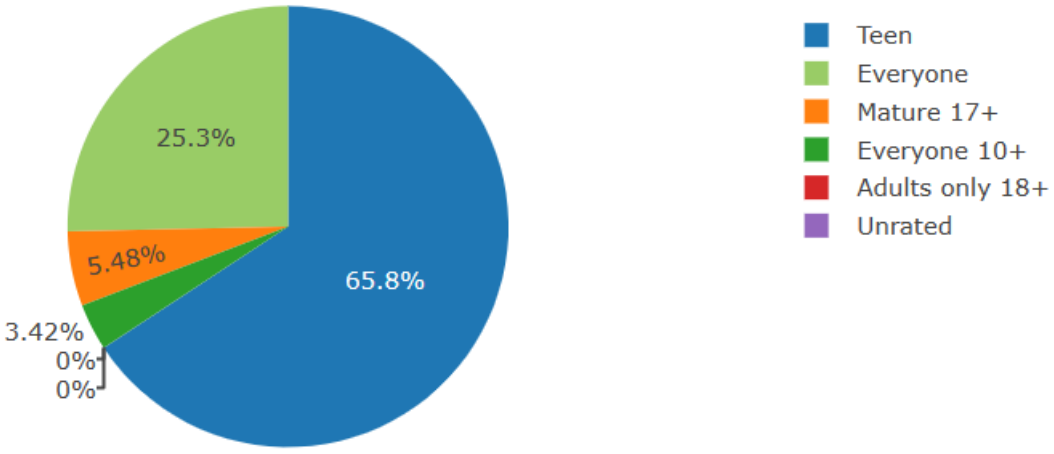
## 2) Layout and demonstration

The whole dashboard page is shown below:


Pic 3. Dashboard page

The filter of category and type, as well as the rating ratio serves as the classification bases. Their modification will lead to the alternation of the graphs.

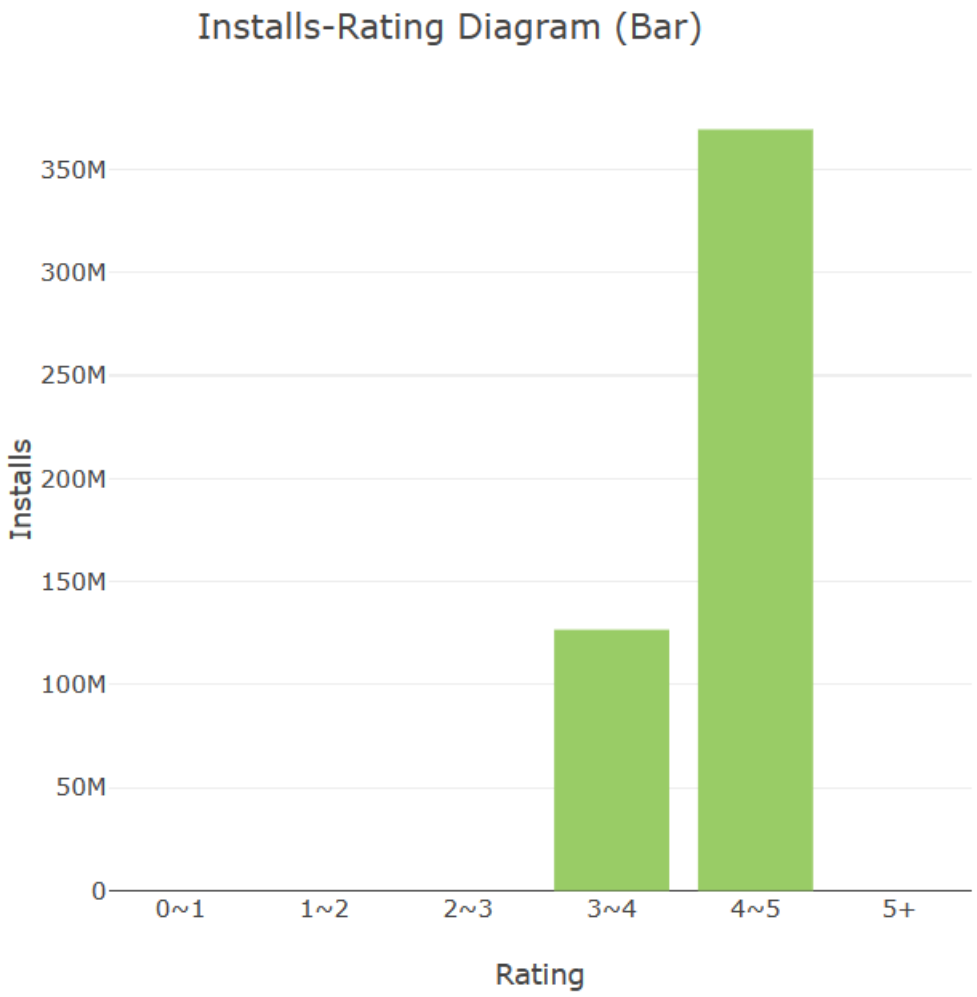
The pie chart describes the number of apps under each rating, with given category and type.




Pic 4. Pie chart

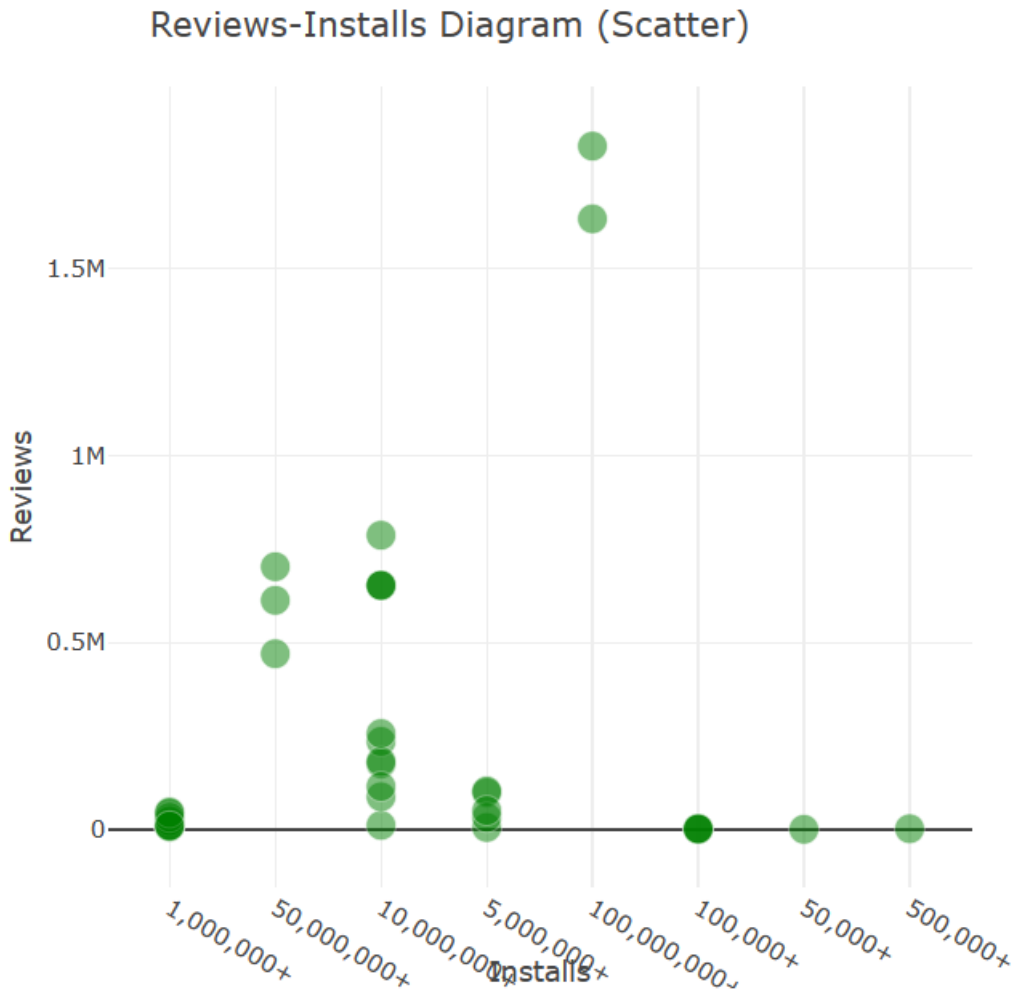
The first bar chart shows the relationships between installs and ratings with given category, type and rating, while the second shows the relationships between reviews

and ratings.



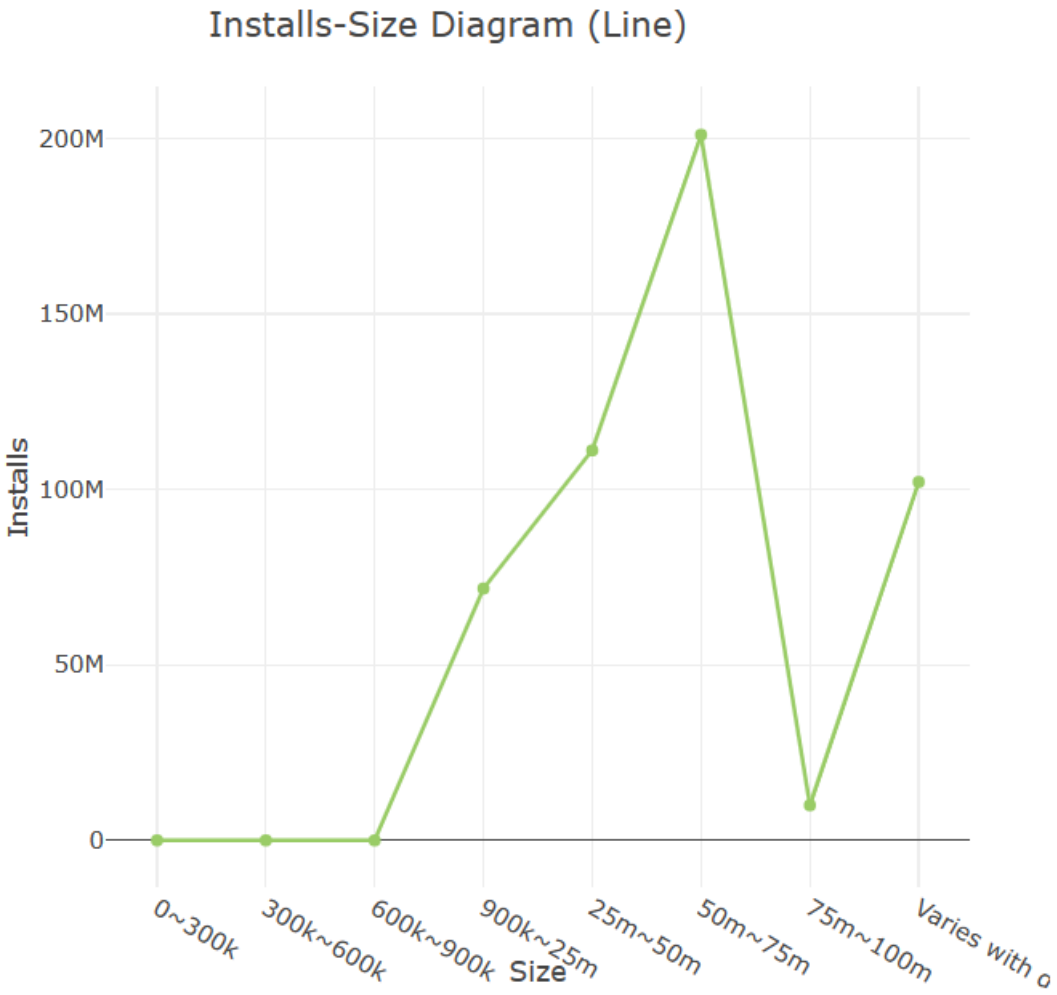

Pic 5. Bar chart

The scatter chart shows the relationships between reviews and installs with given category, type and rating.




Pic 6. Scatter chart

The line chart shows the relationships between installs and size with given category, type and rating.




Pic 7. Line chart