Data Visualization (CS765)

Srujana (srujana@wisc.edu)

DC2 – Signs of Life

For this assignment, I have generated the visualization using pyplots in python. Till now I am able to get the base ready for generating the visualization like segregating the dataset into a NumPy array, dictionaries and in some cases as dataframes. The specialty of the visualization include dual axis so I will be able to compare two quantities with a common x axis i.e., dual y-axis. It can be extended to have dual x-axis and dual y-axis for final hand-in.

This visualization also has a feature of hover, when you hover over a graph it indicates the quantity. I am trying to address two simple tasks as a steppingstone for my design challenge.

1) Relationship between price and rating (price, rating, productid)

- Does the quality of the product increase with the increase in the price?

Findings: From the visualization it is clear that rating doesn't depend on the product price. Products with lesser price have good ratings as well as medium prized goods.

builder.py | main.py | subplots.py | figure.py | _axes.py | CDs_And_Vinyl_meta_5.csv

```
import pandas as pd
import numpy as np
import plotly.graph_objects as go
from collections import OrderedDict
from plotly.subplots import make_subplots

table_a = 'CDs_and_Vinyl_5.csv'
table_meta = 'CDs_And_Vinyl_meta_5.csv'

dfa = pd.read_csv(table_a)
df_meta = pd.read_csv(table_meta)
np_dfa = np.array(dfa)
np_df_meta = np.array(df_meta)
product_rating = {}

for x in np_df_meta:
    col = np_dfa[np_dfa[:, 0] == x[0], 2]
    product_rating.update({x[0]: sum(col)/len(col)})

d_sorted_by_value = dict(OrderedDict(sorted(product_rating.items(), key=lambda x: x[1])))
keyyy = list(d_sorted_by_value.keys())

d_price = {}
for k in d_sorted_by_value.keys():
    col = np_df_meta[np_df_meta[:, 0] == k, 2]
    d_price.update({k: sum(col)})
```

Figure 1: Python code to extract and store data from csv.

Figure 1 shows the Python code used to segregate data by extracting from csv.

```python
# create graph
fig = make_subplots(specs=[[{"secondary_y": True}]])


fig.add_trace(
    go.Scatter(x=list(d_sorted_by_value.keys()), y=list(d_sorted_by_value.values()), mode='lines', name='Average rating'
               ), secondary_y=False,)

fig.add_trace(
    go.Scatter(x=list(d_sorted_by_value.keys()), y=list(d_price.values()), name='Price'),
    secondary_y=True,)
# Set x-axis title
fig.update_xaxes(title_text="<b>Product ID's</b>")

# Set y-axes titles
fig.update_yaxes(title_text="<b>Average rating/product </b>", secondary_y=False)
fig.update_yaxes(title_text="<b>Price</b>", secondary_y=True)

# fig.savefig('dc2-1.png')
fig.show()
```

Figure 2: Python code used to generate visualization using pyplot.

Figure 2 shows python code which is used to generate graphs, here we are using secondary axis which makes this graph a dual axis graph. It includes two-line graphs which compares average rating of a product and price associated with each product.

Visualization:

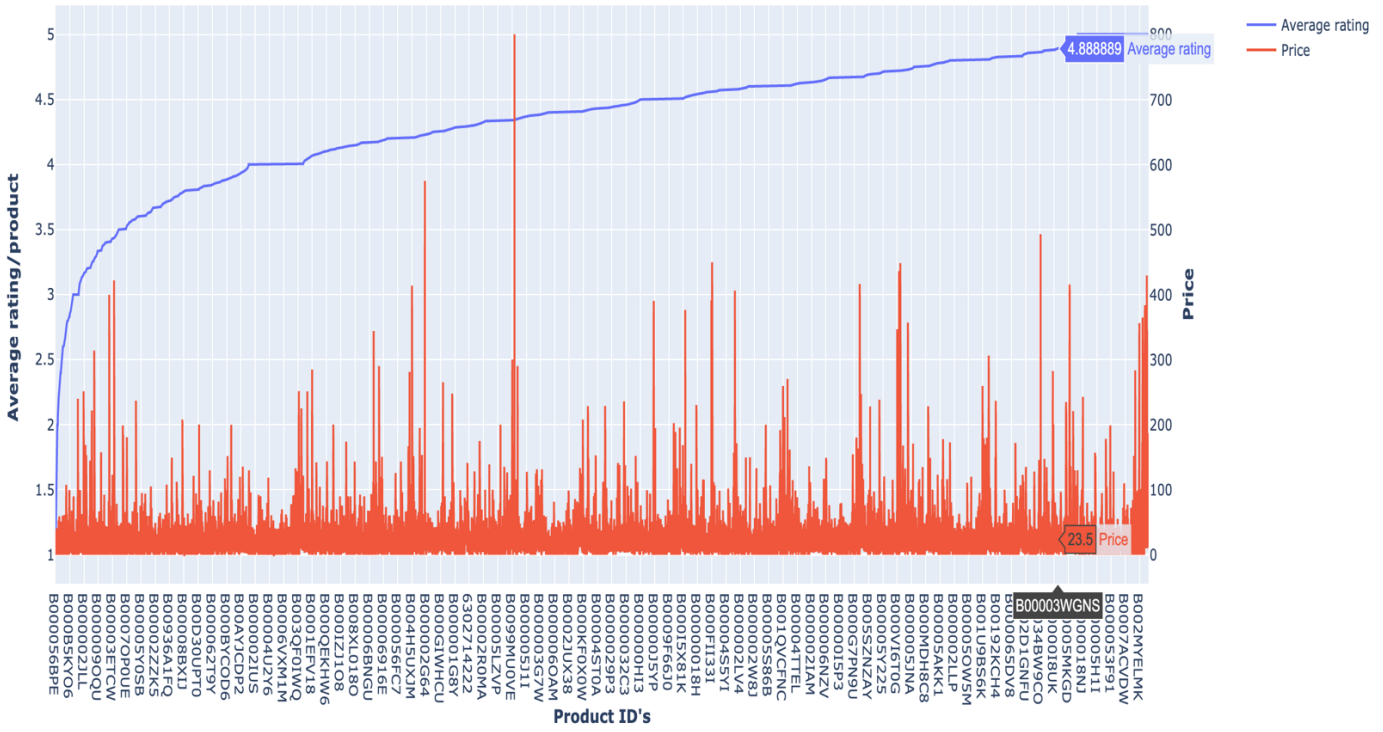

Figure 3: Visualization which compares price and rating of a product.

Figure 3 is a visualization which shows or compares price and average rating of a product. It is interactive since we can get to know the price and rating of a product with its product ID displayed.
With this visualization we can not just declare that product with lesser price has good rating or vice versa. But it gives an interesting story that there are few products whose price is extremely high versus lots of product with lower price.

Trying to carry out more and better visualization as part of final hand-in.

Link to repo: https://github.com/SrujanaN/CS765-DC2
It is a private repo, will be given access based on request.