Product Recommender System EDA

Data-Source

Amazon electronics product reviews data – JSON formatted: from below dataset
https://nijianmo.github.io/amazon/index.html

Sample review:

```
{"overall": 5.0,
 "verified": true,
 "reviewTime": "06 2, 2014",
 "reviewerID": "AS97Z2TIG6DZA",
 "asin": "B0002MQGK4",
 "unixReviewTime": 1401667200}
```

- `reviewerID` - ID of the reviewer, e.g. A2SUAM1J3GNN3B
- `asin` - ID of the product, e.g. 0000013714
- `overall` - rating of the product
- `unixReviewTime` - time of the review (unix time)
- `reviewTime` - time of the review (raw)
- `verified` – if customer actually purchased the product reviewed

Sample metadata:

```
{ "asin": "0000031852”, "price": 3.17, "also_buy":
["B00JHONN1S", "B002BZX8Z6", "B00D2K1M3O", "0000031909",
"B00613WDTQ", "B00D0WDS9A", "B00D0GCI8S", "0000031895”],
"brand": "Coxlures", "categories": [["Sports & Outdoors",
"Other Sports", "Dance"]] }
```

- `asin` - ID of the product, e.g. 0000031852
- `price` - price in US dollars (at time of crawl)
- `related` - related products (also bought)
- `brand` - brand name
- `categories` - list of categories the product belongs to

Final data frame info:

We merged both product reviews and meta data , we also considered the verified reviews only , and renamed columns :

```
#    Column          Dtype
---  ------           -----
 0   Rating           float64
 1   verified         bool
 2   reviewTime       object
 3   reviewerID       object
 4   asin             object
 5   unixReviewTime   int64
 6   category         object
 7   also_buy         object
 8   brand            object
 9   price            object
```

Statistics and visualizations:

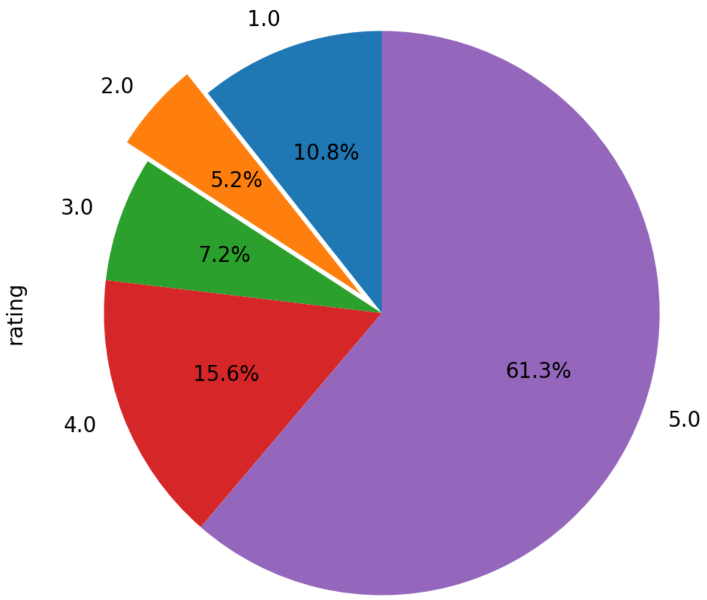
Statistics:

```
# Total reviews
total = len(product_reviews)
print("Number of reviews: ", total)
print()

# How many unique reviewers?
print("Number of unique reviewers: ", len(product_reviews.reviewer_id.unique()))
reviewer_prop = float(len(product_reviews.reviewer_id.unique()) / total)
print("Prop of unique reviewers: ", round(reviewer_prop, 3))
print()

# How many unique products?
print("Number of unique products: ", len(product_reviews.product_id.unique()))
product_prop = float(len(product_reviews.product_id.unique()) / total)
print("Prop of unique products: ", round(product_prop, 3))
print()

# Average star score
print("Average rating score: ", round(product_reviews.rating.mean(), 3))

# Review number per unique customer ?
print('\nReview per customer: {}'.format(
    (len(product_reviews) / len(product_reviews['reviewer_id'].unique()))))

# Review number per unique product ?
print(
    '\nReview per product: {}'.format((len(product_reviews) /
len(product_reviews['product_id'].unique()))))
```

```
Number of reviews:  19225906
Number of unique reviewers:  8897920
Number of unique products:  715774

Average rating score:  4.114
Review per customer: 2.16
Review per product: 26.86
```

Ratings distribution:

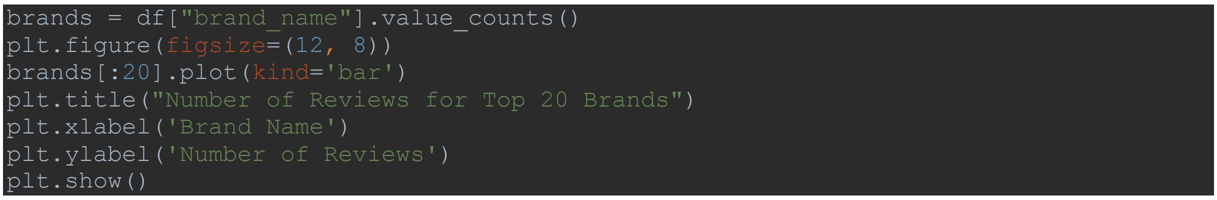
```
plt.figure(figsize=(10, 6))
df.groupby('rating').rating.count()
df.groupby('rating').rating.count().plot(kind='pie', autopct='%1.1f%%',
startangle=90, explode=(0, 0.1, 0, 0, 0), )
plt.show()
```



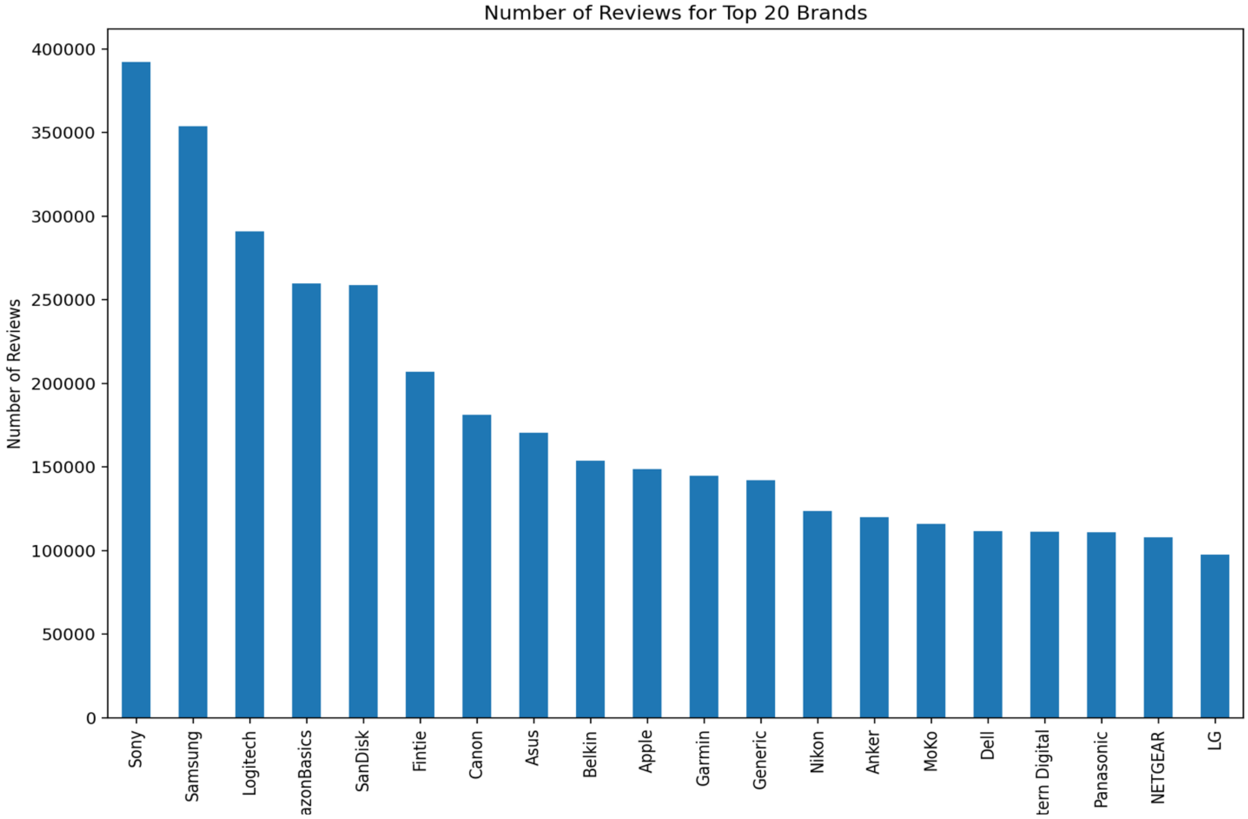
Observations:

- The majority of reviews are in (4,5) : 77% of reviews are highly positive

Top 20 brands reviewed:

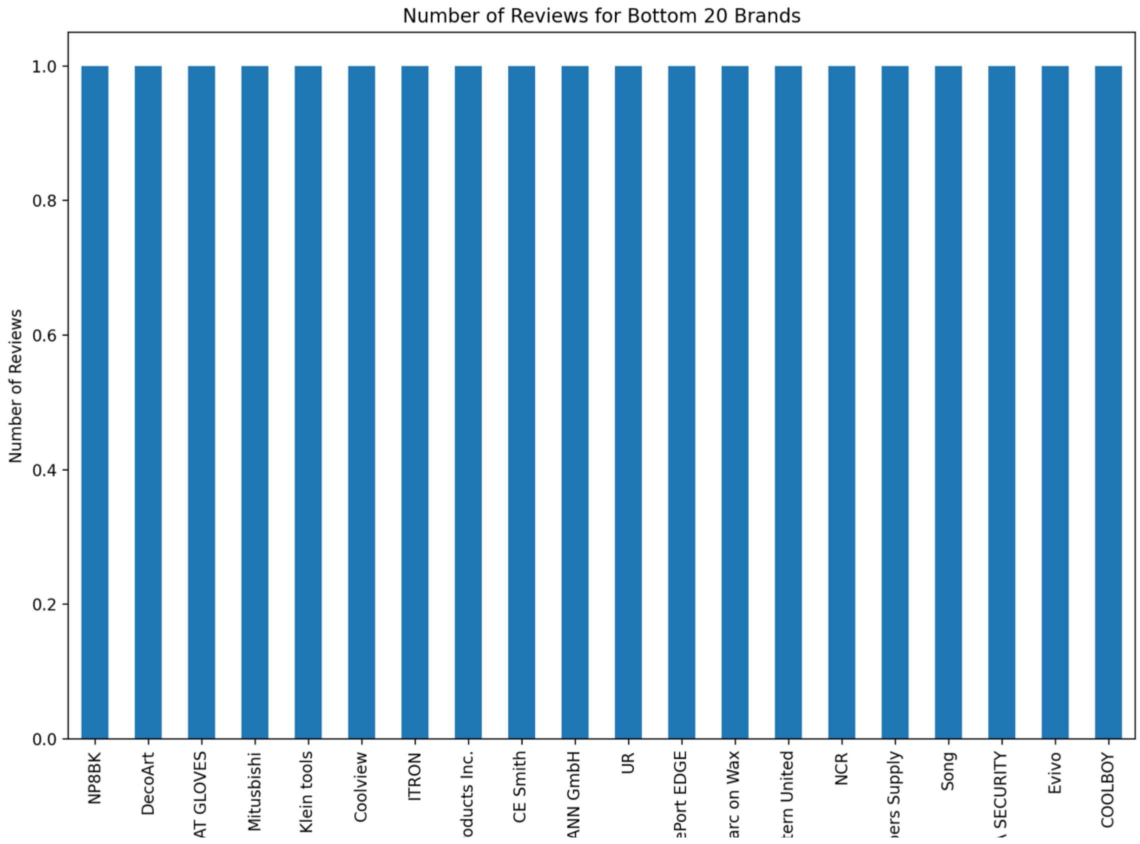


```
brands = df["brand_name"].value_counts()
plt.figure(figsize=(12, 8))
brands[:20].plot(kind='bar')
plt.title("Number of Reviews for Top 20 Brands")
plt.xlabel('Brand Name')
plt.ylabel('Number of Reviews')
plt.show()
```



Number of reviews or bottom 20 brands:

```
brands = df["brand_name"].value_counts()
# brands.count()
plt.figure(figsize=(12, 8))
brands[-20:].plot(kind='bar')
plt.title("Number of Reviews for Bottom 20 Brands")
plt.xlabel('Brand Name')
plt.ylabel('Number of Reviews')
plt.show()
```

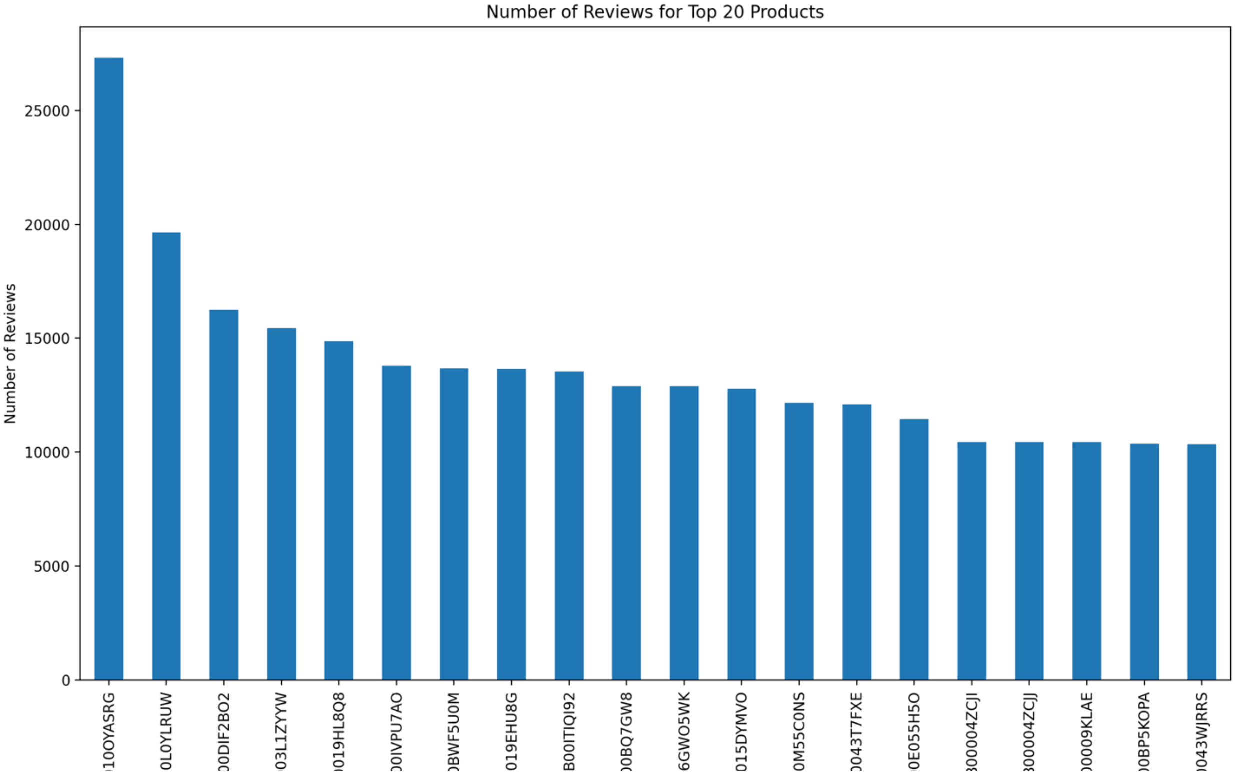


Observations:

- The above brands had one and only review – lowest review count

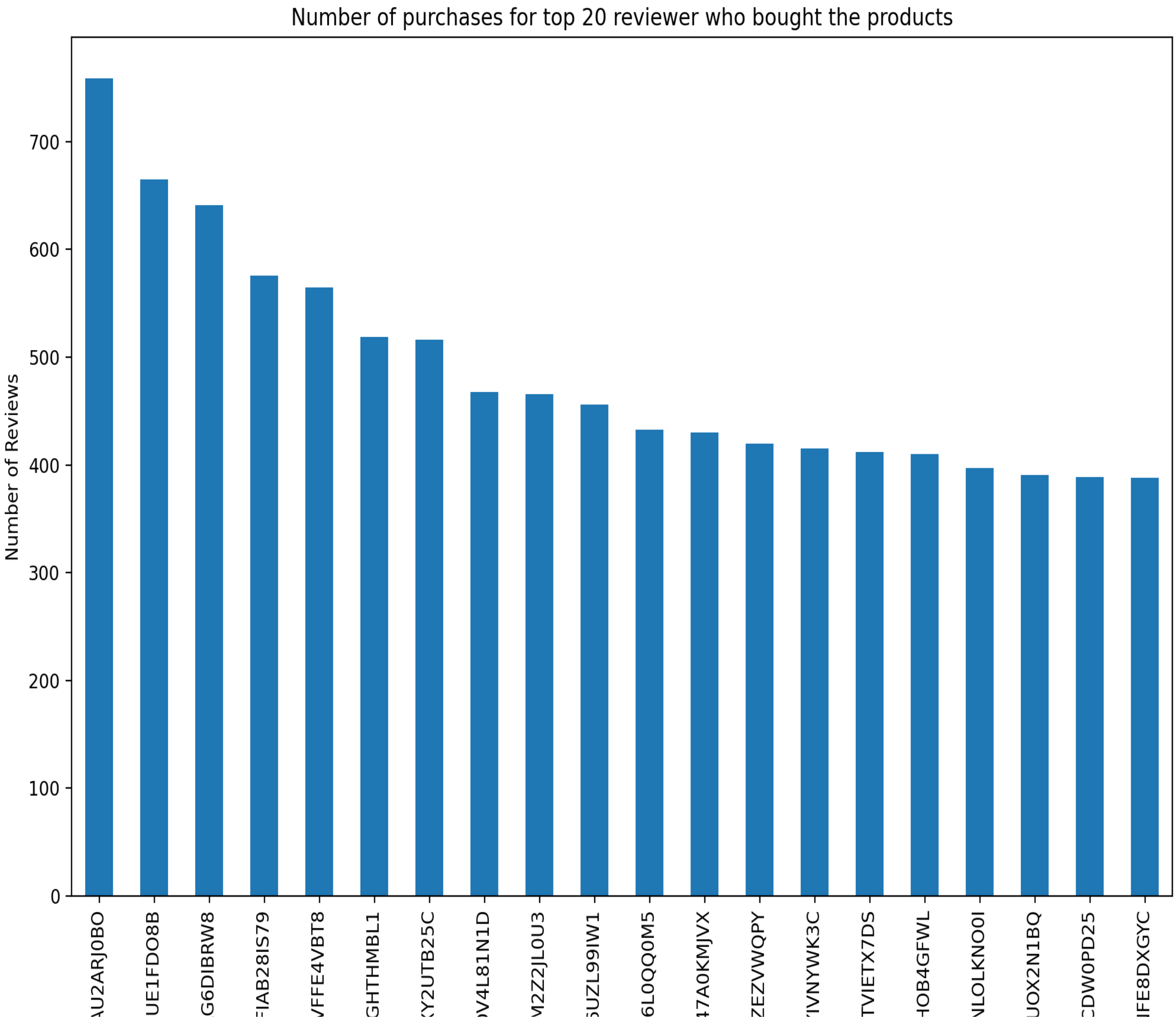
Top most 20 products reviewed:

```
products = df["product_id"].value_counts()
plt.figure(figsize=(12, 8))
products[:20].plot(kind='bar')
plt.title("Number of Reviews for Top 20 Products")
plt.xlabel('Product Name')
plt.ylabel('Number of Reviews')
plt.show()
```



Number of Reviews (puchases) for Top 20 reviewer who bought the product:

```
products = df["reviewer_id"].value_counts()
plt.figure(figsize=(12, 8))
products[:20].plot(kind='bar')
plt.title("Number of Reviews for Top 20 reviewer who bought the product")
plt.xlabel('Reviewer ID')
plt.ylabel('Number of Reviews')
plt.show()
```


Number of purchases for top 20 reviewer who bought the products
Number of Reviews
AU2ARJ0BO
UE1FDO8B
G6DIBRW8
FIAB28IS79
VFFE4VBT8
GHTHMBL1
KY2UTB25C
DV4L81N1D
M2Z2JL0U3
5UZL99IW1
6L0QQ0M5
47A0KMJVX
ZEZVWQPY
YVNYWK3C
TVIETX7DS
HOB4GFWL
NLOLKNO0I
UOX2N1BQ
CDW0PD25
IFE8DXGYC