16 JANUARY 2022

CSE-424
BIG DATA ANALYSIS TERM PROJECT REPORT

RECOMMENDATION SYSTEM WITH SPARK

Team Members:

AHMET NASUHCAN ÜNLÜ - 171805062
OZAN İRFAN BAYAR - 171805041
İLKER MAVİLİ - 181805084

WORK SHARING POLICY

- Researching for appropriate dataset for recommendation systems. (Ahmet)
- Transform of the data by handling errors in the csv file with using regular expressions library & splitting each record into fields. (Ahmet)
- Visualization of the important fields in the dataset with using Matplotlib library. (Ozan)
- Implementation of the ALS algorithm & building of the recommendation engine with using collaborative filtering. (İlker)
- Computing the scores and getting comparisons of the RMSE values. (All Team Members)

SYSTEM CONFIGURATION

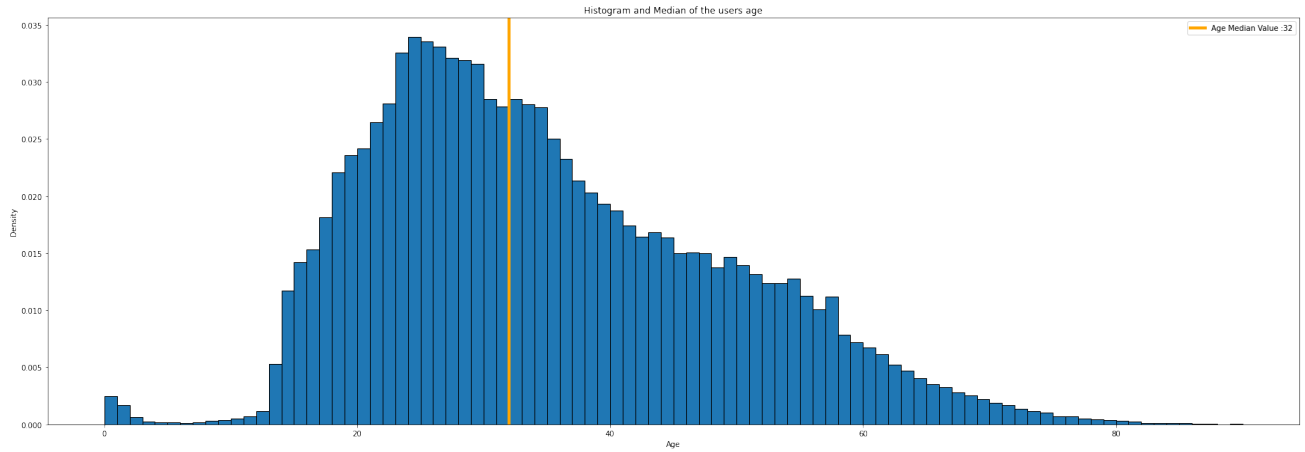

```python
import socket
import platform
hostname = socket.gethostname()
ip = socket.gethostbyname(hostname)
uname = platform.uname()
print("="*40, "System Information", "="*40)
print(f"Hostname: {hostname}")
print(f"Ip address: {ip}")
print(f"System: {uname.system}")
print(f"Release: {uname.release}")
print(f"Version: {uname.version}")
print(f"Machine: {uname.machine}")
print(f"Processor: {uname.processor}")
```

```
======================================== System Information ========================================
Hostname: Ahmets-MacBook-Pro.local
Ip address: 127.0.0.1
System: Darwin
Release: 21.2.0
Version: Darwin Kernel Version 21.2.0: Sun Nov 28 20:28:41 PST 2021; root:xnu-8019.61.5~1/RELEASE_ARM64_T6000
Machine: arm64
Processor: arm
```

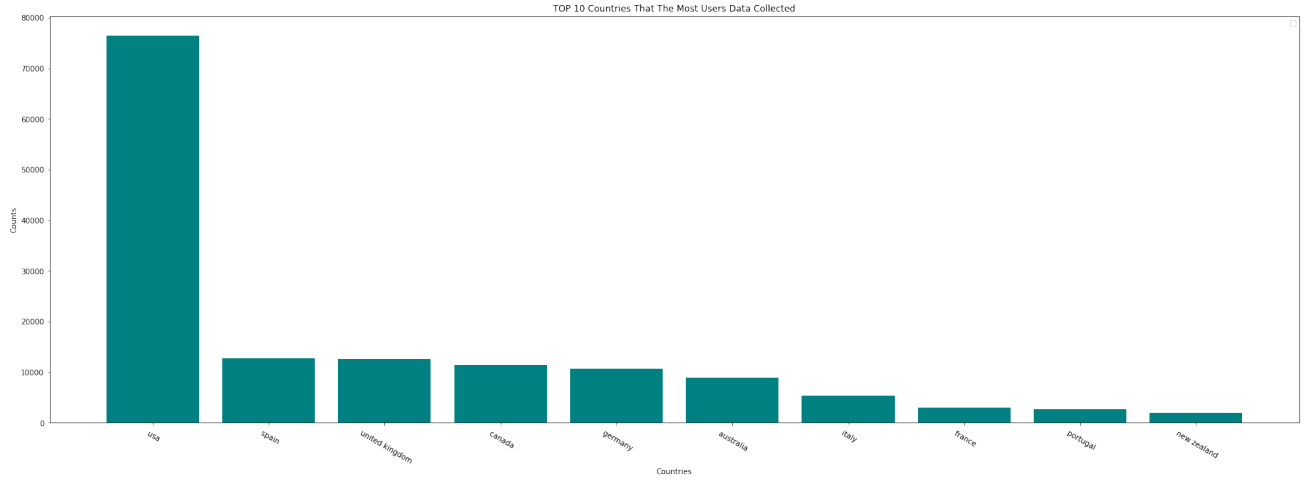
Outputs

```
age_list = users_age.map(lambda item: handle_age(item))
age_list_filtered = age_list.filter(lambda item: item <= 90 and item >=0 )
age_counts = collections.OrderedDict(sorted(age_list_filtered.countByValue().items()))
age_list_filtered.stats()
```

```
(count: 167666, mean: 34.54562045972328, stdev: 13.769988967606256, max: 90.0, min: 0.0)
```




```
TOP 10 Countries That The Most Data Collected:
[(' usa', 76495),
 (' spain', 12691),
 (' united kingdom', 12541),
 (' canada', 11385),
 (' germany', 10642),
 (' australia', 8896),
 (' italy', 5377),
 (' france', 3037),
 (' portugal', 2721),
 (' new zealand', 2026)]
```



Statistics of Ratings

```
(count: 397245, mean: 7.601852760890598, stdev: 1.8412729800421206, max: 10.0, min: 1.0)
```

```
       User Rated
[('Titan', 10.0),
 ('The Hippopotamus Pool (Amelia Peabody Mysteries (Paperback))', 10.0),
 ('Crystal Singer', 9.0),
 ("Winter's Tale", 9.0),
 ('Moreta: Dragonlady of Pern', 8.0),
 ("Nerilka's Story (Dragonriders of Pern (Paperback))", 8.0),
 ('Sphere', 8.0),
 ('The Chronicles of Pern: 1st Fall (The Dragonriders of Pern)', 8.0),
 ('Rising Sun', 8.0),
 ('Postmortem', 8.0)]
```

10 Book Recommend Ratings For ALS Train 10_10_01 Dataset

```
       ALS Recommend
[("Thumper's Little Sisters Fun-To-Read Library Vol.2", 26.67714207574988),
 ('Wizard of Oz Postcards in Full Color (Card Books)', 24.934340801690563),
 ('The Shining', 24.896158185579978),
 ('If Only It Were True', 24.16467141962802),
 ('Le Combat ordinaire, tome 1', 23.70749122836607),
 ('Sense and Sensibility (Wordsworth Classics)', 23.5952725600483),
 ("I can do it myself: Featuring Jim Henson's Sesame Street muppets",
  23.583219861286658),
 ("Chicken Soup for the Couple's Soul (Chicken Soup for the Soul)",
  23.23916077858971),
 ('The Golden Compass (His Dark Materials, Book 1)', 23.114180381551783),
 ("The Magical Household: Spells &amp, Rituals for the Home (Llewellyn's Practical Magick Series)",
  22.837874005071697)]
```

MSE and RMSE Values for Trained Models

```
print("Mean Squared Error Scores For Each Model:", )
c = 0
for i in modelsArray:
    print(f'Model {arr[c]} = {i[-2]}')
    c+=1
```

```
Mean Squared Error Scores For Each Model:
Model 10_10_01 = 53.34119117544696
Model 10_10_1 = 18.636155991534356
Model 10_50_01 = 22.122748588086736
Model 10_50_1 = 11.336808136605864
Model 10_200_01 = 14.227357999044946
Model 10_200_1 = 10.90573102377254
Model 50_10_01 = 36.457727810835074
Model 50_10_1 = 16.1902301464026
Model 50_50_01 = 17.618660766068086
Model 50_50_1 = 10.431964460630414
Model 50_200_01 = 12.06616164322049
Model 50_200_1 = 10.133502863434764
Model 200_10_01 = 32.22401184629114
Model 200_10_1 = 14.843206633579006
Model 200_50_01 = 16.68182390647972
Model 200_50_1 = 10.227754833152614
Model 200_200_01 = 11.894178300434724
Model 200_200_1 = 9.969889097231045
```

```
print("Root Mean Squared Error Scores For Each Model:", )
c = 0
for i in modelsArray:
    print(f'Model {arr[c]} = {i[-1]}')
    c+=1
```

```
Root Mean Squared Error Scores For Each Model:
Model 10_10_01 = 7.303505403259928
Model 10_10_1 = 4.316961430396889
Model 10_50_01 = 4.703482602081859
Model 10_50_1 = 3.3670176917571824
Model 10_200_01 = 3.7719170191091087
Model 10_200_1 = 3.302382628311344
Model 50_10_01 = 6.038023502010826
Model 50_10_1 = 4.023708506639442
Model 50_50_01 = 4.197458846262592
Model 50_50_1 = 3.2298551764174217
Model 50_200_01 = 3.4736380990570233
Model 50_200_1 = 3.183316331035099
Model 200_10_01 = 5.676619755302546
Model 200_10_1 = 3.8526882346718643
Model 200_50_01 = 4.084338857940135
Model 200_50_1 = 3.198086120346451
Model 200_200_01 = 3.4487937457080156
Model 200_200_1 = 3.1575131190908845
```

```
print('\tALS Recommend')
top_k_recs=model10_10_01.recommendProducts(33933,10)
sc.parallelize(top_k_recs).map(lambda rating: (Book_ISBN_Name_Map[rating.product], rating.rating)).collect()
```

```
        ALS Recommend
[("Thumper's Little Sisters Fun-To-Read Library Vol.2", 26.67714207574988),
 ('Wizard of Oz Postcards in Full Color (Card Books)', 24.934340801690563),
 ('The Shining', 24.896158185579978),
 ('If Only It Were True', 24.16467141962802),
 ('Le Combat ordinaire, tome 1', 23.70749122836607),
 ('Sense and Sensibility (Wordsworth Classics)', 23.5952725600483),
 ("I can do it myself: Featuring Jim Henson's Sesame Street muppets",
  23.583219861286658),
 ("Chicken Soup for the Couple's Soul (Chicken Soup for the Soul)",
  23.23916077858971),
 ('The Golden Compass (His Dark Materials, Book 1)', 23.1141803815517 83),
 ("The Magical Household: Spells &amp, Rituals for the Home (Llewellyn's Practical Magick Series)",
  22.837874005071697)]
```

```
print('\tALS Recommend that has Minimum RMSE Value')
top_k_recs=model50_50_01.recommendProducts(33933,10)
sc.parallelize(top_k_recs).map(lambda rating: (Book_ISBN_Name_Map[rating.product], rating.rating)).collect()
```

```
        ALS Recommend that has Minimum RMSE Value
[('Titan', 9.976675931151313),
 ('Dune (Remembering Tomorrow)', 9.897097353117973),
 ('The Return of the King (The Lord of the Rings, Part 3)', 9.67146481041655),
 ("Ender's Game (Ender Wiggins Saga (Paperback))", 9.439850386796705),
 ('Wizard and Glass (The Dark Tower, Book 4)', 9.417753279695077),
 ('The Two Towers (The Lord of the Rings, Part 2)', 9.37787168983112),
 ('The Hobbit : The Enchanting Prelude to The Lord of the Rings',
  9.312813812981306),
 ('Harry Potter and the Prisoner of Azkaban (Book 3)', 9.270850246985592),
 ('Maus a Survivors Tale: My Father Bleeds History', 9.223470860655919),
 ("My Sister's Keeper : A Novel (Picoult, Jodi)", 9.218495528553476)]
```

```
print('\tALS Recommend that has Minimum RMSE Value')
top_k_recs=model200_50_01.recommendProducts(33933,10)
sc.parallelize(top_k_recs).map(lambda rating: (Book_ISBN_Name_Map[rating.product], rating.rating)).collect()
```

```
        ALS Recommend that has Minimum RMSE Value

[('Titan', 9.974005912787725),
 ("Ender's Game (Ender Wiggins Saga (Paperback))", 9.338520619650806),
 ('Harry Potter and the Prisoner of Azkaban (Book 3)', 9.290752007946011),
 ('Wolves of the Calla (The Dark Tower, Book 5)', 9.140919077312903),
 ('The Return of the King (The Lord of the Rings, Part 3)', 9.077819904287406),
 ('The Calvin and Hobbes Tenth Anniversary Book', 9.02260784983266),
 ('The Fellowship of the Ring (The Lord of the Rings, Part 1)',
  9.0092766966513),
 ('The Two Towers (The Lord of the Rings, Part 2)', 8.998889770165846),
 ("Winter's Tale", 8.989993069941438),
 ('Harry Potter and the Prisoner of Azkaban (Book 3)', 8.986221159461156)]
```

```
print('\tALS Recommend that has Minimum RMSE Value')
top_k_recs=model200_200_01.recommendProducts(33933,10)
sc.parallelize(top_k_recs).map(lambda rating: (Book_ISBN_Name_Map[rating.product], rating.rating)).collect()
```

```
        ALS Recommend that has Minimum RMSE Value
[('Titan', 9.957538686988931),
 ("Ender's Game (Ender Wiggins Saga (Paperback))", 9.56989341135812),
 ('Harry Potter and the Goblet of Fire (Book 4)', 9.532632588052405),
 ('Harry Potter and the Prisoner of Azkaban (Book 3)', 9.424042762758077),
 ('84 Charing Cross Road', 9.410963004890537),
 ('Mangrove Squeeze', 9.323091876977323),
 ("\\Surely You're Joking, Mr. Feynman!\\: Adventures of a Curious Character",
  9.290815654546417),
 ('Harry Potter and the Chamber of Secrets Postcard Book', 9.27820770050233),
 ('Mostly True: Collected Stories &amp, Drawings', 9.124582233436957),
```

We have decided that the best model for efficient recommendation is configuration with rank = 50, iteration = 50 and lambda value = 0.01. The main reason of our choice is observing the optimum root mean square error and execution time with this parameters.