Big Data Platforms

NETFLIX

Recommender Systems

Abhi Ghose, Akarsh Sahu, Markus Wehr

NETFLIX

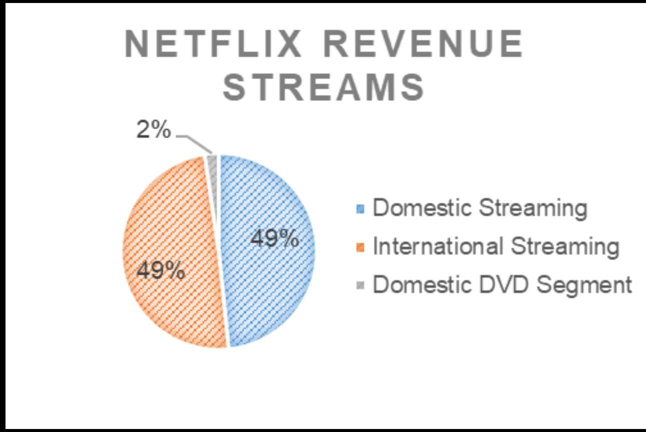
Outline

- Business Problem
- Methodology
- Data Sources
- Data Visualizations
- Models
- Evaluation
- Customer Profiles
- Recommendation Visualizations
- Challenges & Scope for Improvements

Business Problem

Netflix Revenue Streams:

- Membership fees ($ 7.6B domestic, $ 7.8B international, $ 0.36B DVD domestic)[1]
- Potential future streams: Ad-placement (e.g., Stranger Things season 3 alone had placements worth ~$ 15M)[2]
- Placements also help to reduce marketing expenses up to $ 1B per year[3] (e.g. KFC advertised Stranger Things, because their products appear in season 2)



The longer people watch:

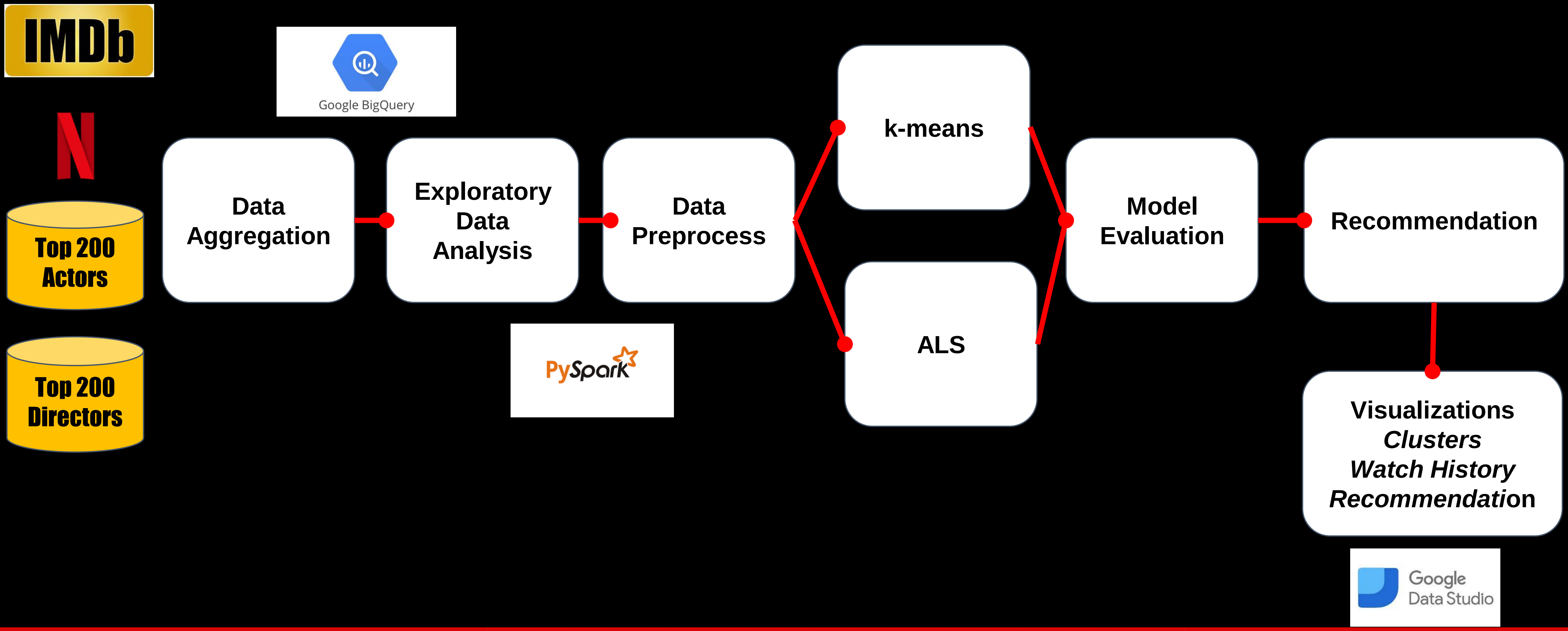
- The lower churn rate (increase revenue from membership fees)
- The more placements can be shown per person (potential revenue + reduced marketing expenses)

…RECOMMENDATION SYSTEM CRUCIAL TO NETFLIX' SUCCESS!

[1] Form 10K Q4 2018.
[2]https://www.fastcompany.com/90380266/more-product-placements-may-come-to-netflix-but-dont-call-them-ads
[3] https://www.businessofapps.com/data/netflix-statistics/

NETFLIX

Methodology



NETFLIX

Data Sources

Data Sources	Data Structure	Combined Size	Processed	# Files
IMDB Database	Individual .csv files for genres, actors, directors, ratings, etc.	6 GB	21 GB master dataset (6 distributed clusters on RCC Hadoop)	7
Netflix Database	4 combined .txt files with single row for each customer	4 GB		4
Top 200 Actors	Oscar winning popular actors .csv file	5 KB		1
Top 100 Directors	Oscar winning popular directors .csv file	5 KB		1

Data Preprocessing

- Load netflix and imdb datasets as tables on HIVE

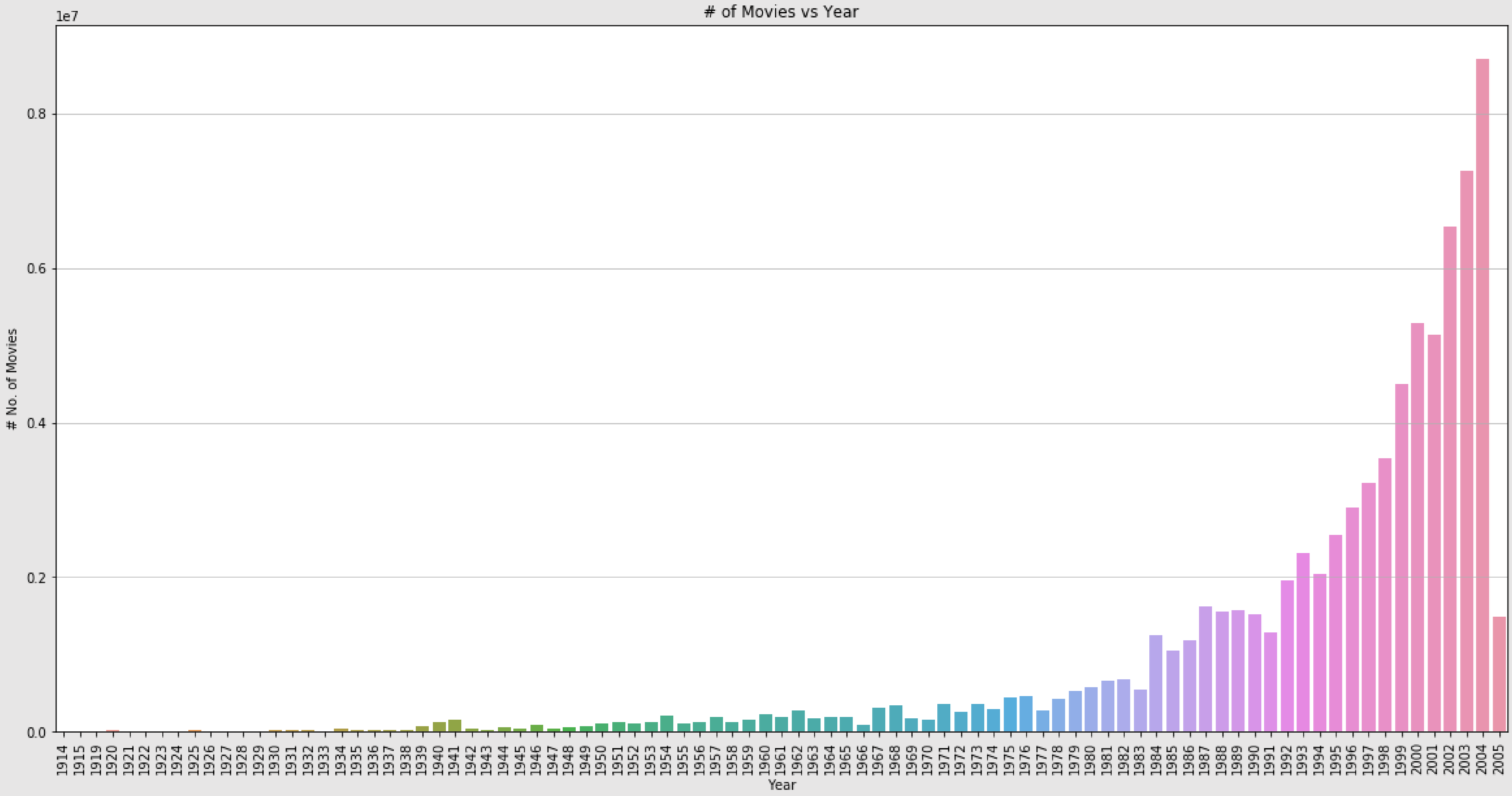
- Clean raw data, remove missing values, scale numeric variables, binning, exploratory data analysis

- Combine Netflix ratings dataset (source: Kaggle) and IMDB movie features (source: IMDB) to generate master dataset

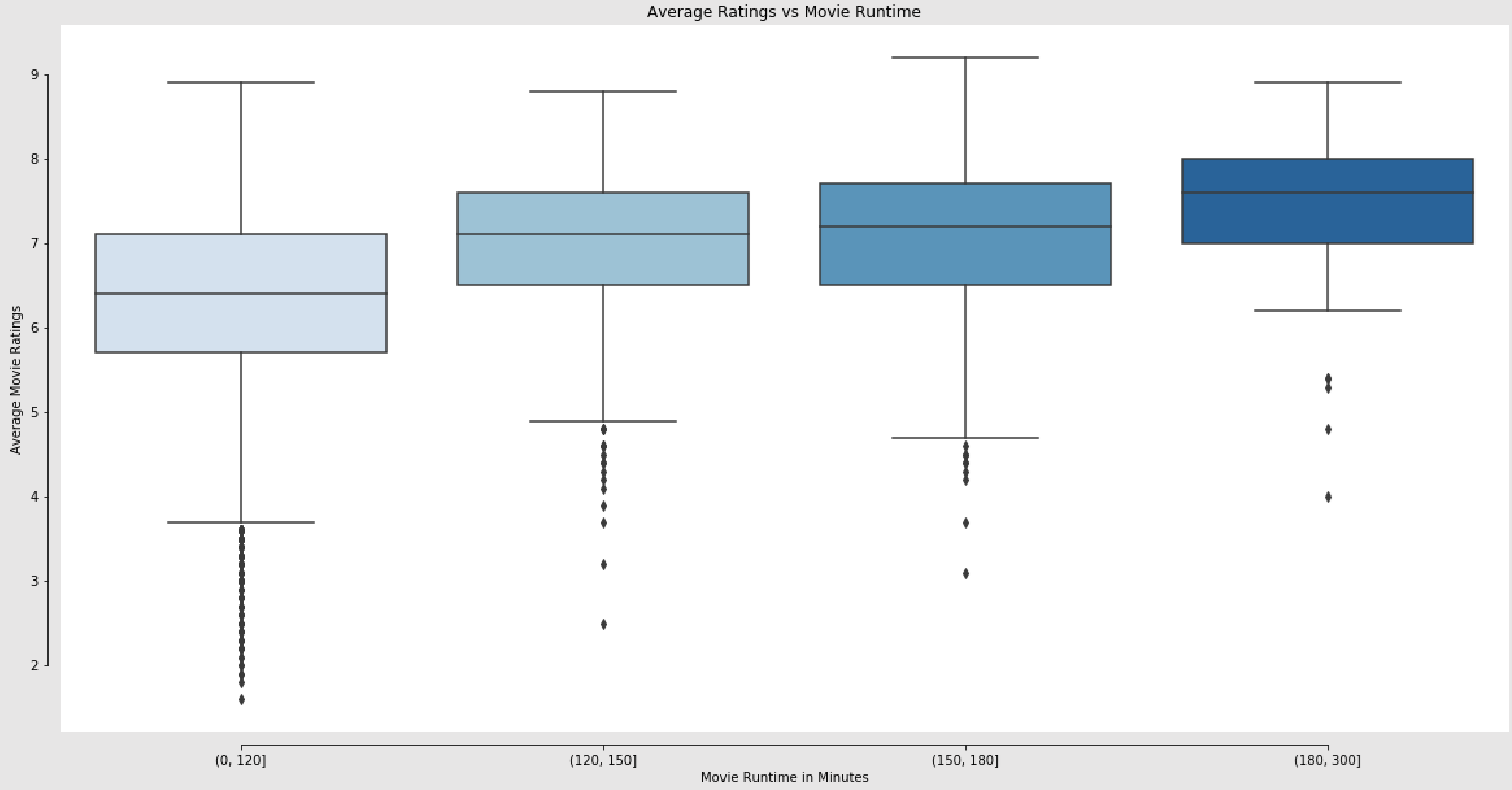
- Feature Engineering:
  - Split genres and combine 22 genres into 8 main genres
  - Customer level aggregation for cluster analysis
    - For eg, Average rating, Average runtime minutes, Average genre rating

- Create google cloud storage bucket to store processed(coalesced) data and model outputs, create tables in Google Cloud Bigquery for querying, visual analysis on Google Data Studio

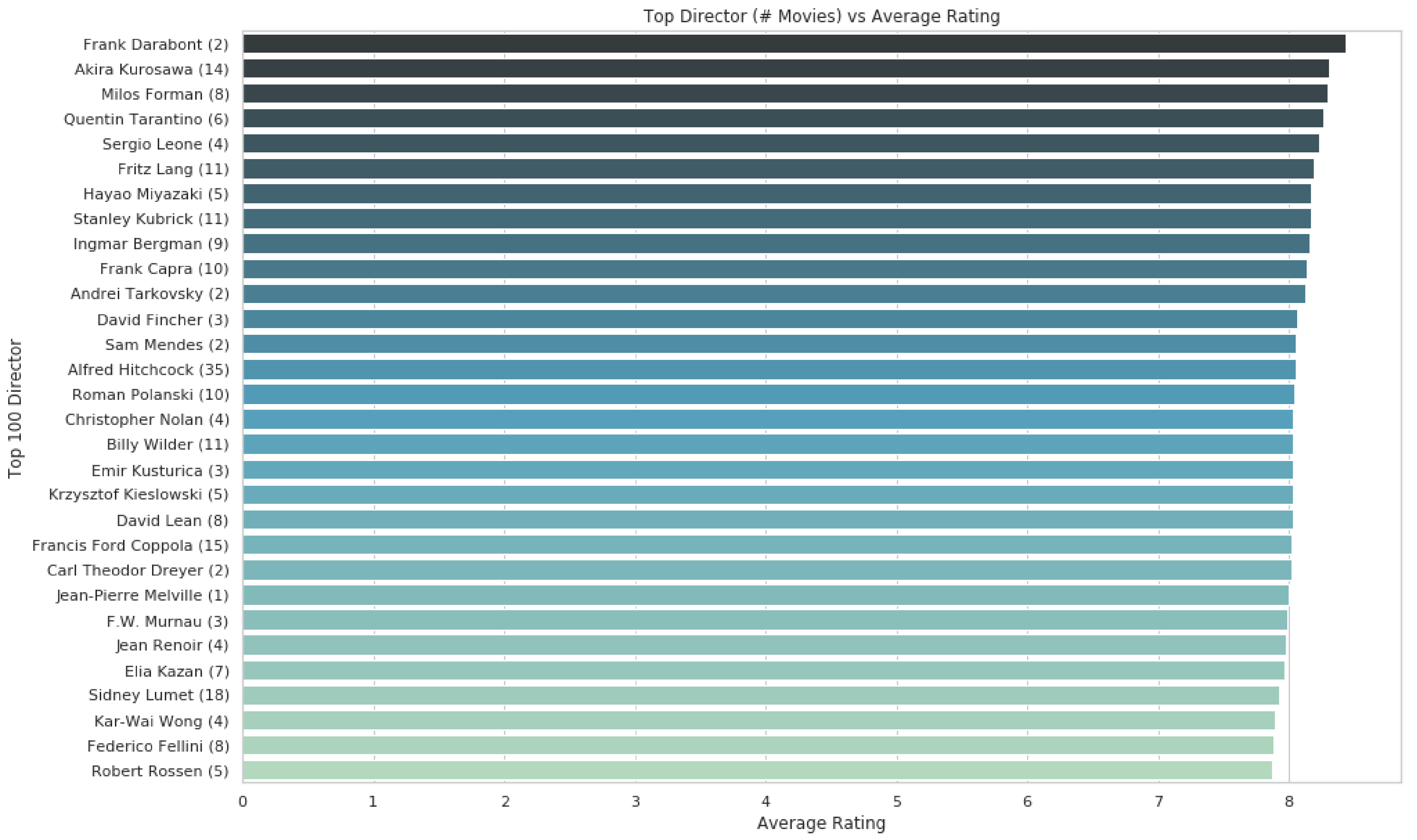
Movies vs Year



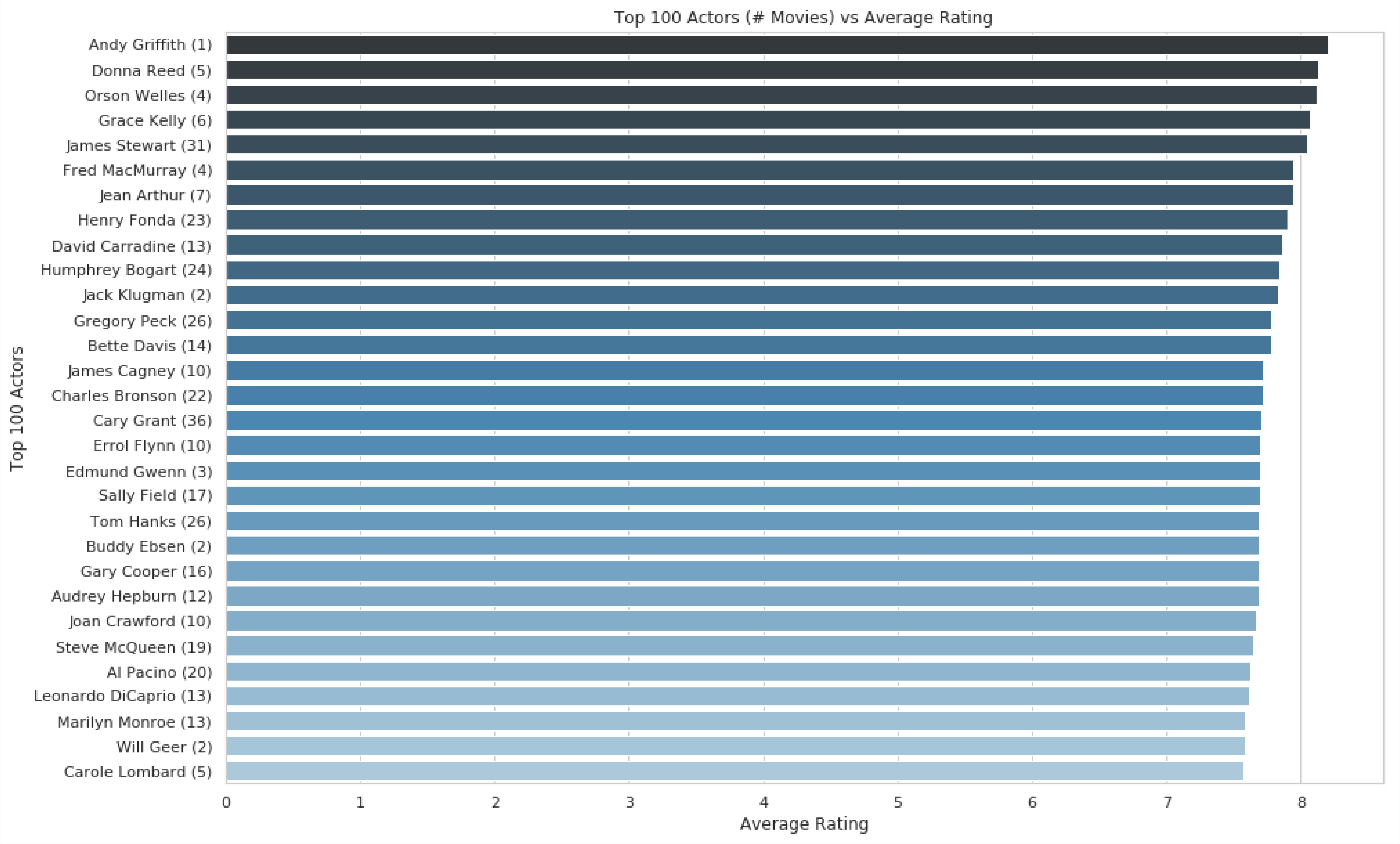
Movie Runtime (mins) vs IMDB Ratings



Top Director (#Movies) vs IMDB Ratings

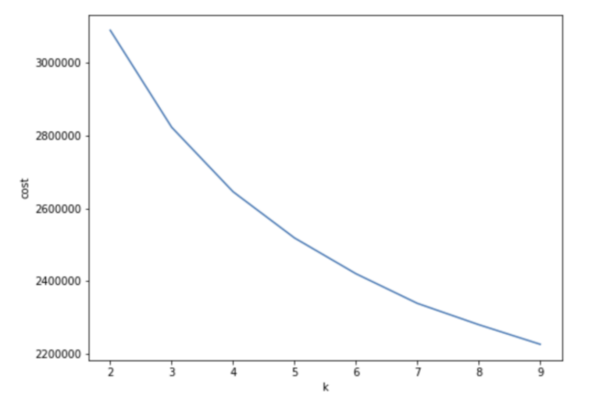


Top Actors(#Movies) vs IMDB Ratings

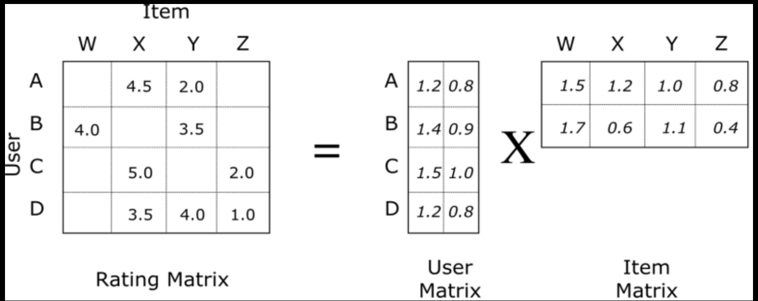


Modeling

KMEANS (k=6)



ALTERNATING LEAST SQUARES (ALS)



Matrix Factorization to reduce sparsity

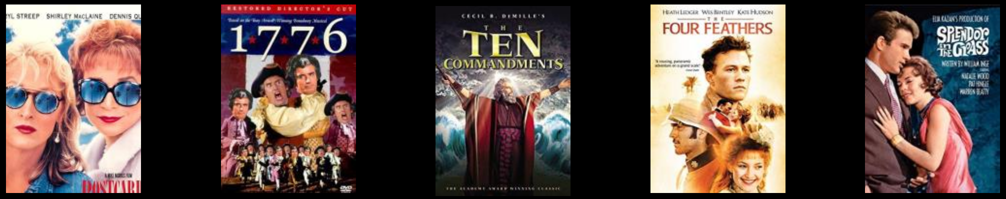
- Vectorizing columns used for clustering, since StandardScaler in pyspark can only handle one column
- After scaling, clustering customers using KMeans, where k based on optimal tradeoff between cost function and number of clusters
- Building individual recommendation engines per cluster using collaborative filtering with ALS
- Evaluate model quality using Root Mean Square Error for each cluster

Clusters	RMSE Score
Cluster 1	1.12
Cluster 2	0.86
Cluster 3	0.85
Cluster 4	0.80
Cluster 5	0.84
Cluster 6	0.89

NETFLIX

Top Movie Recommendations

CLUSTER ONE (historical/documentaries)

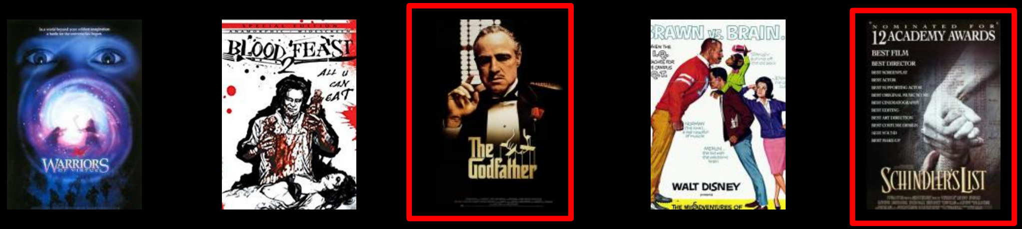


CLUSTER TWO (action & family)

CLUSTER THREE

CLUSTER FOUR

CLUSTER FIVE



CLUSTER SIX (mainstream)

NETFLIX

Harrison Ford and Marlon Brando are guarantees for success!

Top Actors

CLUSTER ONE

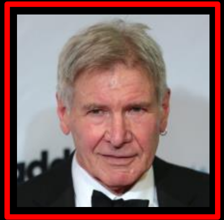
CLUSTER TWO

CLUSTER THREE (Only men?!)

CLUSTER FOUR

CLUSTER FIVE

CLUSTER SIX (the famous ones)

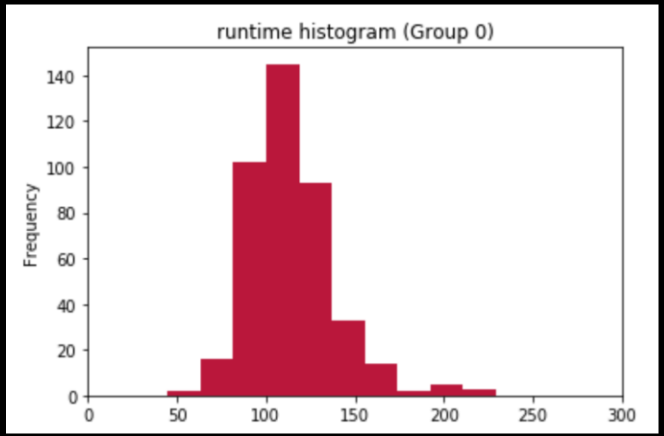
    

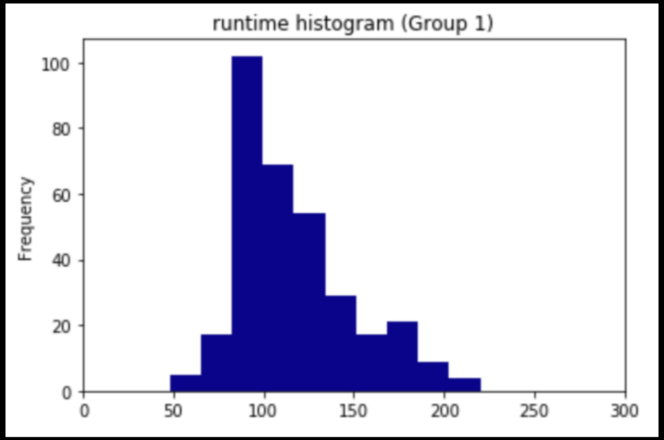
NETFLIX

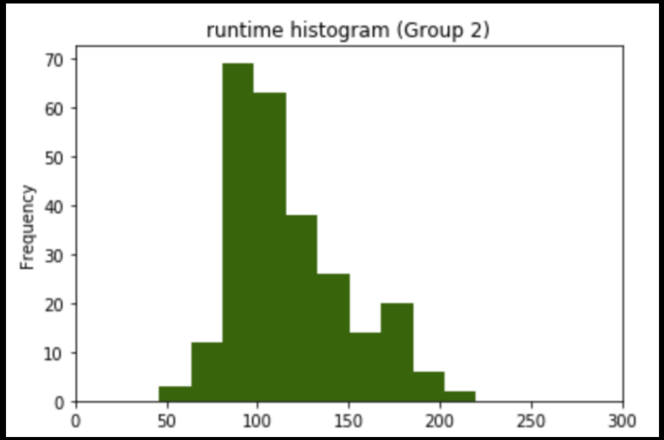
Prefer **shorter** movies

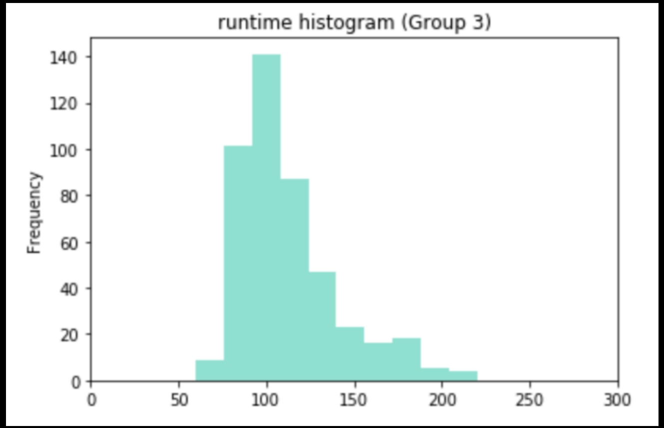
Runtime per Cluster

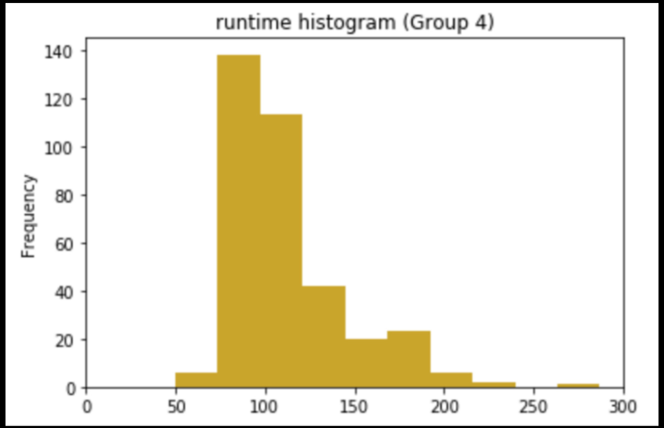
Prefer **longer** movies

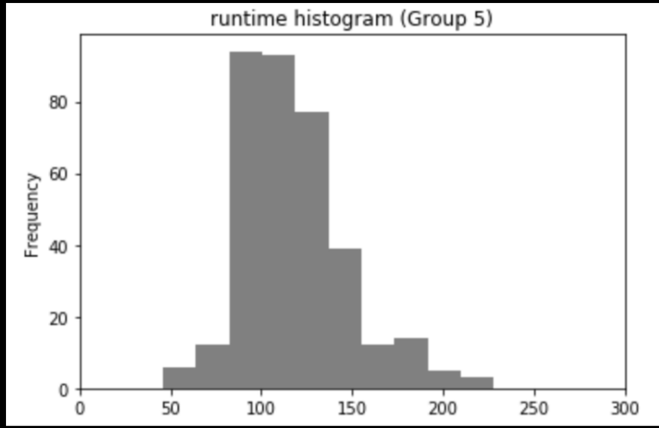






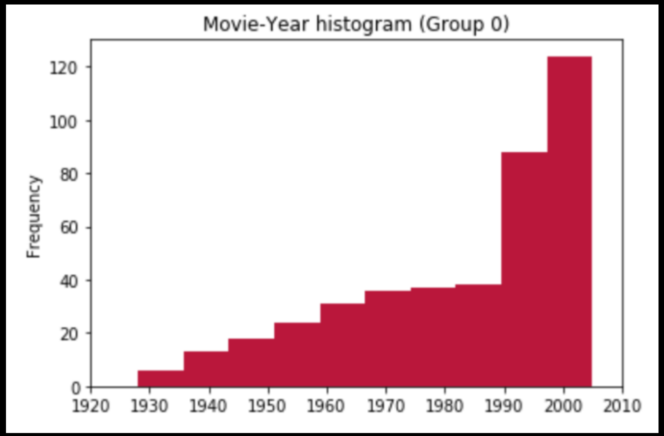


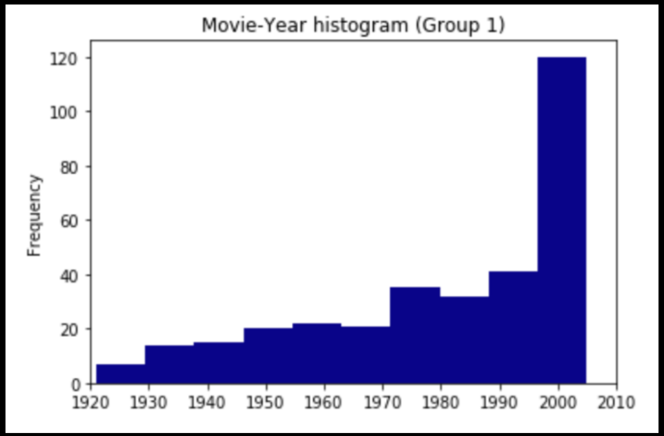


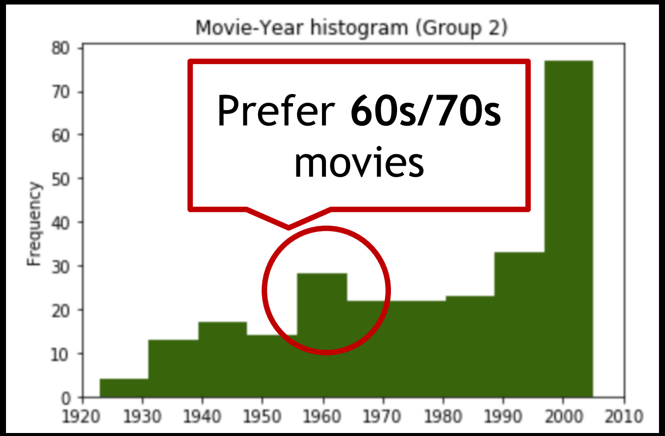


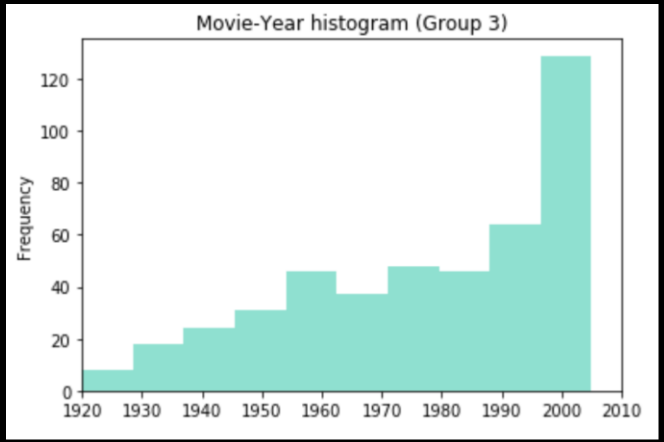
Prefer **new** movies

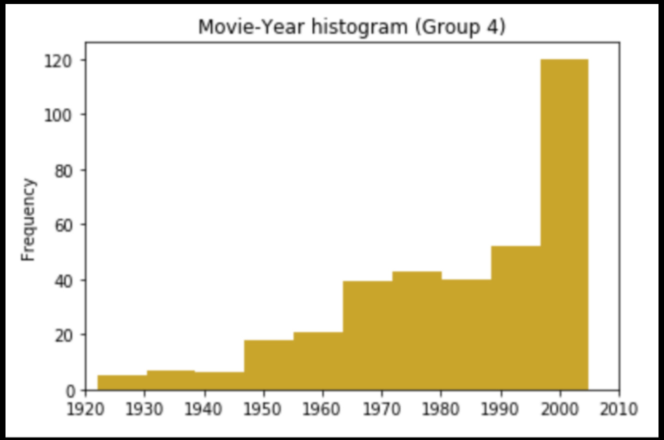
Movie Year Preference

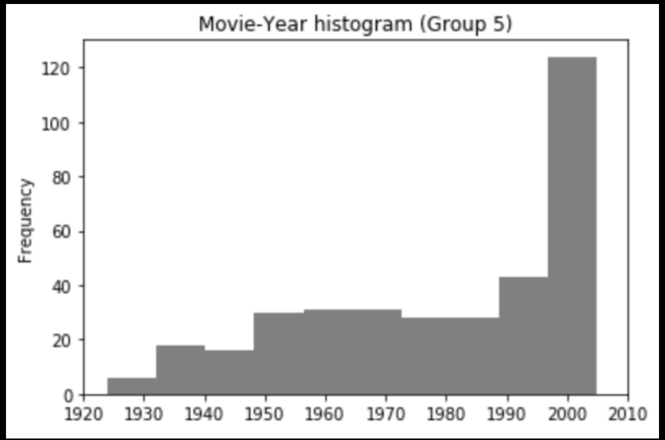












Customer Profiles

1. “THE SERIOUS ONE”

- Interested in historical movies, documentaries, adventure, drama, western, and noir
- Do not want to focus on a movie for too long (busy people!)
- Interested in recent productions

2. “THE FAMILY GUY”

- Like movies with a lot of action, romance and comedy
- Also musical fan
- Fun for the whole family

3. “THE PICKY ONE”

- Take their time to watch a movie
- Like to watch older movies as well (thoroughly selecting what to watch)
- Watches only highly rated movies

4. “THE EXPLORER”

- Like to watch sci-fi, anime, mystery, or horror movies
- Explores the unknown…

5. “THE INTELLECTUAL”

- People with this personality like to watch noir movies, old westerns, or documentaries
- They have a strong personality and express their thoughts (most likely to rate a movie)

6. “THE MAINSTREAMER”

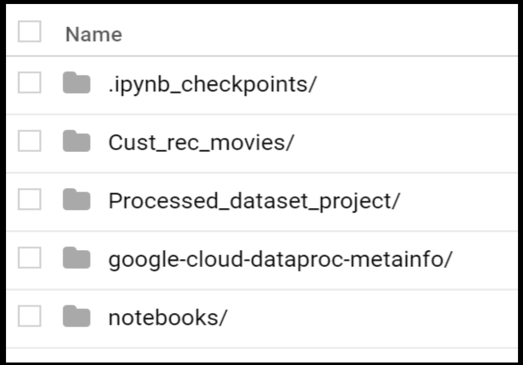
- Go with the most popular movies, actors, and directors
- Do not have a specific genre they are interested in

NETFLIX

Visualizing Recommendations on Google Cloud Services

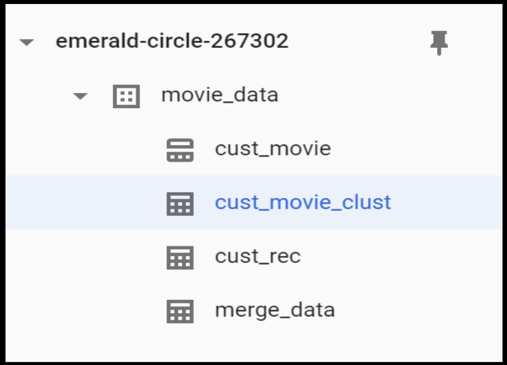
Google Cloud Storage

- Store pre processed, customer level and model prediction dataset as separate datafiles utilizing google sdk and gsutil



Google BigQuery

- Create tables in gcloud bigquery and pull data from gcloud storage for OLAP
- Total data stored : 107 gb



Google Data Studio

- Create 3 page report to analyze cluster groups, visualize customer watch history and attributes, and recommendations

[Click here to access the visualization tool]

NETFLIX

Challenges & Improvements

Challenges

- ❑ Limited availability of clustering methods (e.g. DBSCAN)
- ❑ Need to vectorize columns to perform column-wise processing

Scope for Improvements

- ❑ Build own DBSCAN model in pyspark
- ❑ Get more features (e.g. customer related)
- ❑ Perform hypothesis tests on different customer clusters and features to verify differences
- ❑ Perform graph filtering to sort recommended movies by no. of degrees

NETFLIX

Thank You !

Questions?

NETFLIX