IMDb

Internet Movie Database

Yasin Edin, Ibrahim Gökce, Norbert Jendreizik

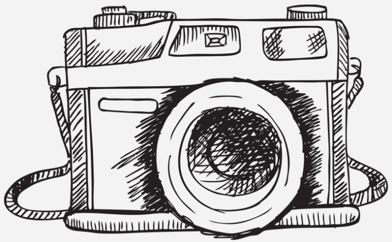
Content

Objectives

Work Packages:
- Databricks
  - Exploratory Data Analysis (EDA)
  - Feature Engineering
- Sagemaker
  - Model Selection
  - Tuning

Conclusion:
- Lessons Learned
- Future Outlook

2



Objectives

Objectives

1. **Explore** the data on a distributive computing cluster
2. **Select features** that may help you predict the ratings of movies
3. **Train** a model of your choice on your features to predict the rating of a movie
4. **Deploy** the model on AWS

Work Packages - Databricks

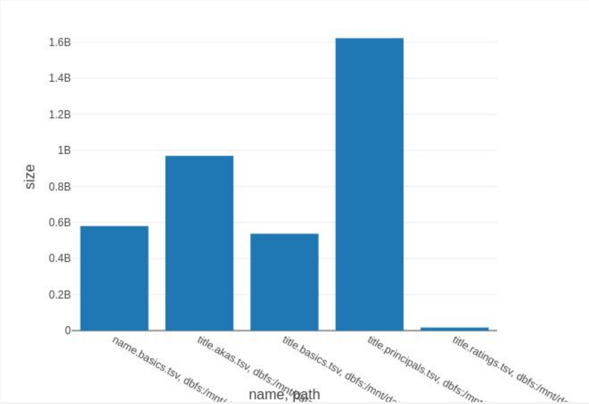
What we had to do

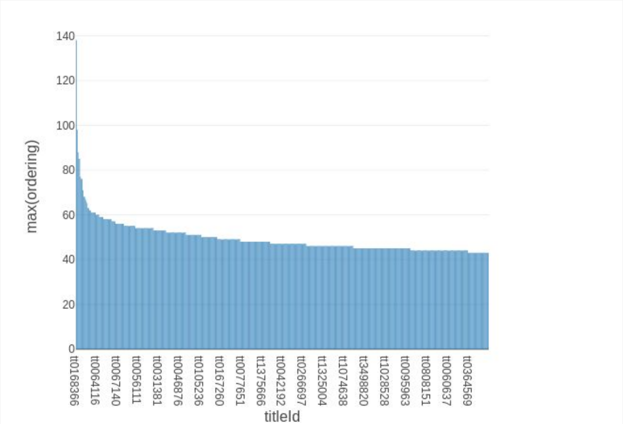
Preprocessing

- 5 tables, 5 dataframes:
  - df_akas,
  - df_basics,
  - df_principals,
  - df_ratings,
  - df_names*

- Cleaning files of e.g. NaN's, inconsistent values

- Encoding of e.g. region, genre etc.



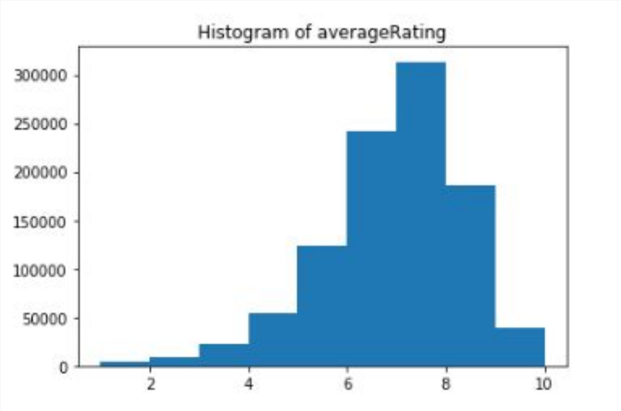


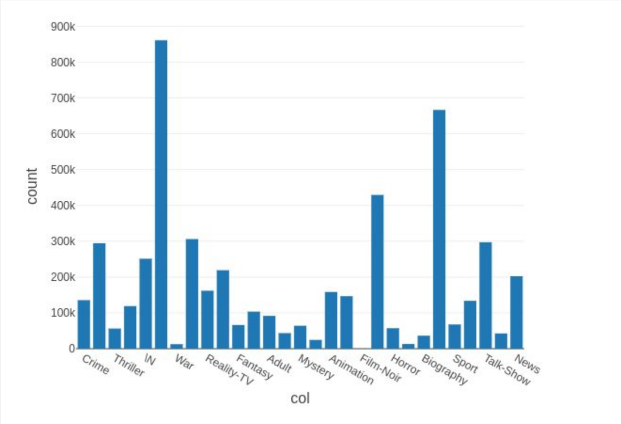
General overview

- df_principals has the the most entries
- There are more unique titles in df_principals than in df_ratings

Akas

- Contains alternative names of films. For e.g. Pokemon has one tt-number and diverse names
- Types and Attributes are more than 90% empty
- Most movies are from France, Germay, Spain, Italy, India,...





Ratings

- The mean rating is 6.89 (maybe people are a little bit kinder than expected), with a standard deviation of roughly 1.40.
  We have roughly 1 million entries in total.

- 75% of the movies have 74 or fewer total votes.

Basics

- Splitting the multiple columns into separate genres shows that the genre DRAMA has 866.7K and COMEDY 666.5K titles

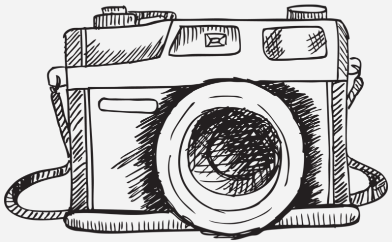
separate features by modularized tables

Modular features which are on a per table basis:

- df_ids = votes(df_ids)
- df_ids = principals(df_ids)
- df_ids = akas(df_ids)
- df_ids = basics(df_ids)

-----> We could have spent more time on the EDA, but we wanted to focus on the pipeline and wanted to sharpen our skills there.

2



Model Selection and Tuning

XGB performs best

	01	02	03
Model	**Linear Learner** Started with Linear Regression as a baseline model	**KNN** KNN as the idea of nearest neighbor came across as KNN brought the worst score, no further steps	**XGBoost** Trying to push weak learner features
Test RMSE	1.32	1.44	1.22
Hyper Parmeter Tuner	objective_type = 'Minimize' strategy='Random' 1.32		objective_type='Minimize' strategy='Bayesian' 1.20

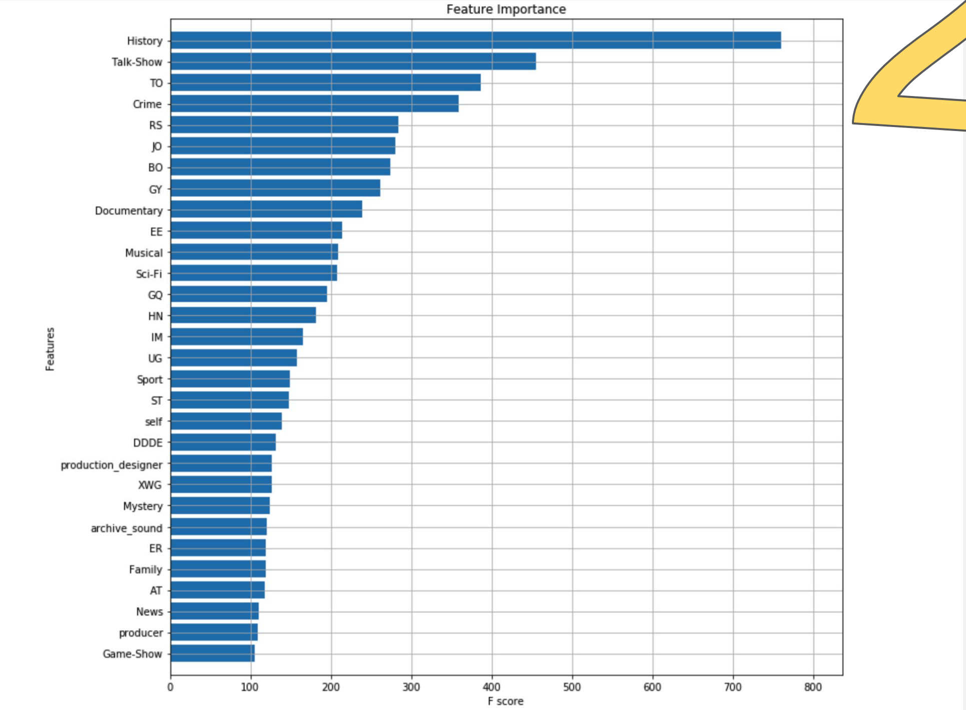
Model impacted by genre and country

First thoughts:

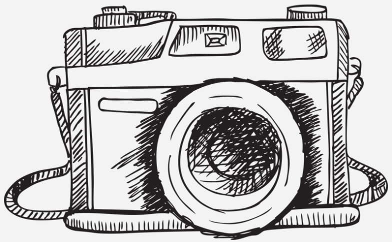
- History with biggest impact on features and it seems like that history movies have better ratings and smaller standard deviation compared to the overall count

- Other genres like Talk-Show and Crime are also important. Further Analysis required as to why.

- Countries: It could be that countries with a low number of movies are easier to split the dataset with.



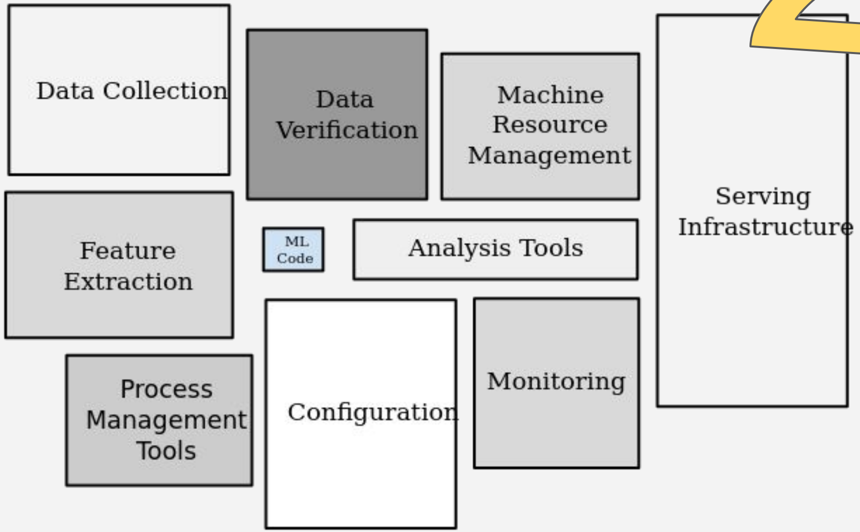
2



Conclusion

Lessons Learned

- Separate tasks (code cleaning, eda) can be conducted when waiting for a long query
- Always check the logic first (e.g. with a small sample) before conducting a query on the whole dataset
- Making different pieces of software talk to each other is complex



https://developers.google.com/machine-learning/crash-course/production-ml-systems

Future Outlook

- Continue EDA
- Analyze Feature importance
- Automated Feature Selection
- Incorporate MlFlow
- Make the endpoint publicly available
- Implement logic which takes care of input data
- Document the model
- ...

https://github.com/yaedin/imdb_project


Thank you!