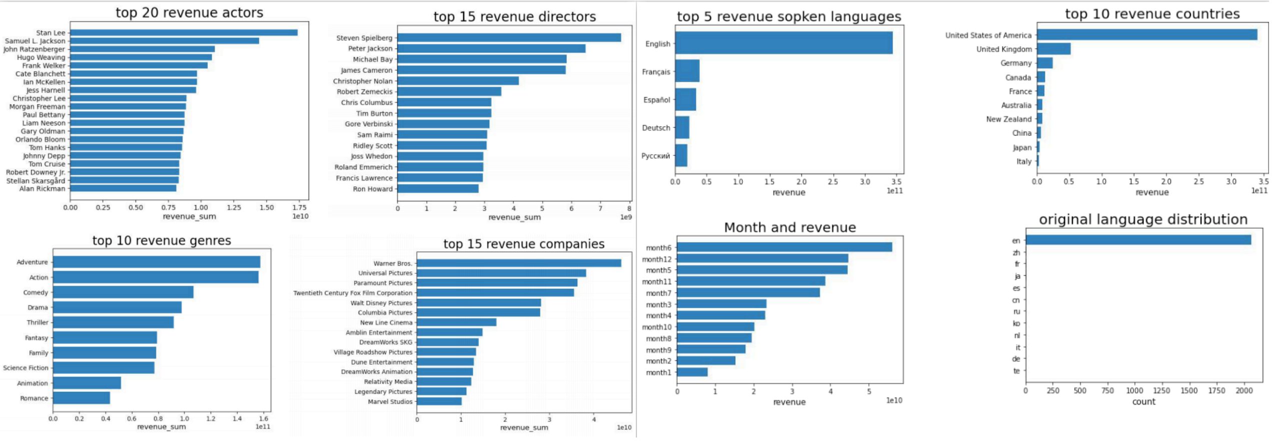
Part1

Step1: data visualization

Read the `trainning.csv` as dataframe, then process columns, then get the realationship betwen columns and revenue.(figure1)

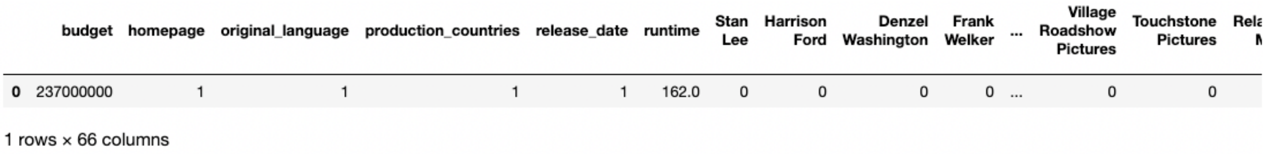


`Figure1`

Step2: analysis and process

From figure 1, features that most revelent to revenues can be found. The data process:

- Add top actors, directors, genres, companies to the df as coulmns. (eg. If the movie has such actor in cast then df[actor]=1 else 0).
- For `spoken_languages, original languages and production_countries`, label the first rank as 1 and the rest as 0. For `release_date`, label month in top5 rank as 1 and the rest as 0. For `homepage`, label 0 for no homepage else 1.
- Drop text data such as `overview, tagline, keywords, status...`, drop `revenue, rating`.
- Finally drop all nonnumerical columns to fit the model.(figure 2)



	budget	homepage	original_language	production_countries	release_date	runtime	Stan Lee	Harrison Ford	Denzel Washington	Frank Welker	...	Village Roadshow Pictures	Touchstone Pictures	Rela
0	237000000	1	1	1	1	162.0	0	0	0	0	...	0	0	

1 rows × 66 columns

`Figure2 processed df.head(1)`

Step3: Performance and evaluation

I found the Random forest can have a better performance than Linear regression, Logic regression and Decision tree. I set n_estimators=30, random_state=60 as final parameter value,

```
zid,MSE,correlation
z5239235,6077834850815273,0.52
```

Discussion

- The most annoying part is find the top revenue actors and directors, some actors occur twice or more in the `cast` column. Dict and List operations are used to deal with that problem.
- It takes me much time to change the parameters of RF model.
- MSE is very large, I think the reason is the values are large.
- I tried to use text data to improve performance. However, my processing does not improve the performance, I think it's mainly because my NLP skill is not good enough.

Part2

Step1: data visualization

Part 2 data process is very similar with task1, the only difference is change the `revenue` to `rating`.

Step2: analysis and process

Same with part1

Step3: Performance and evaluation

I found the Random forest Classifier can have a better performance than Linear Classifier, SVM Classifier and KNN Classifier.

I set n_estimators=30, random_state=17 as final parameter value.

```
average_precision,average_recall,accuracy
z5239235,0.71,0.67,0.75
```

Discussion

- Setting parameters for different model is the most difficult part.
- I tried to use text data to improve performance. However, my processing does not improve the performance, I think it's mainly because my NLP skill is not good enough.