11/10/2020

DATA EXPLORATION RESULTS

Team 6:
Oguzhan Akan
Darius Hooks
Jaymish Raju Patel
Jason Sabal
Bibinur Zhursinbek

California State University, Northridge
COMP 541 – Data Mining – F2020
Assignment 3

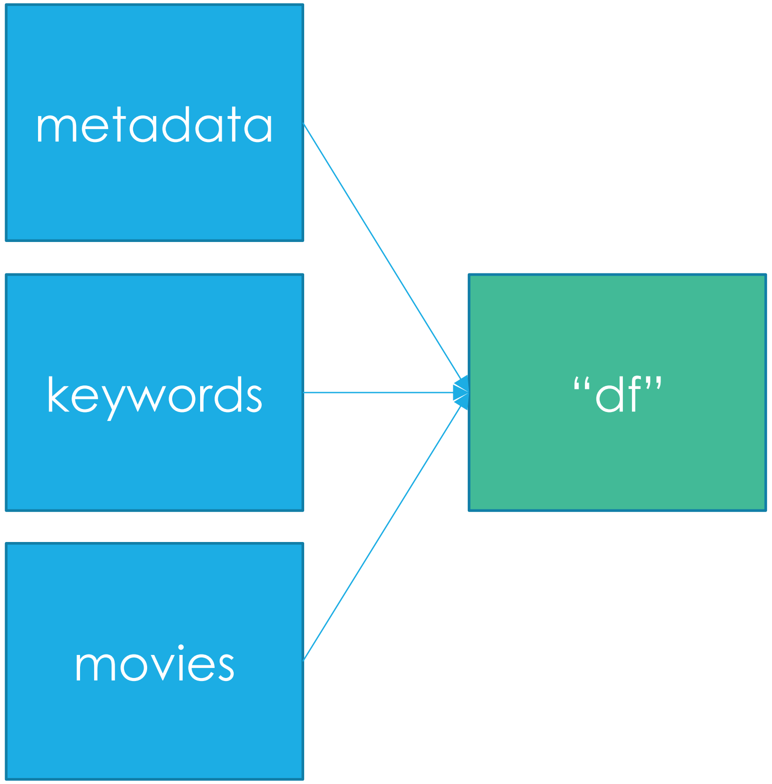
Contents

- 1. Preparing the Data
  - 1.1. Importing Libraries
  - 1.2. Combining the Source Tables
  - 1.3. Transforming Features to their Correct Types
- 2. Data Exploration
  - 2.1. Exploring Continuous Features
    - 2.1.1. Feature Overview
    - 2.1.2. Correlation Matrix
  - 2.2. Exploring Categorical Features
    - 2.2.1. Class Distributions & Imbalance
    - 2.2.2. Categorical Feature Dependency Using Chi-Square Test
      - 2.2.2.1. Chi-square Test for Each Feature
      - 2.2.2.2. Chi-square Test - Summary
- 3. Frequent Pattern Analysis
  - 3.1. Genres (Genre List)
  - 3.2. Spoken Languages
  - 3.3. Keywords
  - 3.4. Overview
- 4. Results
  - 4.1. Data Exploration Results - Continuous Features
  - 4.2. Data Exploration Results - Categorical Features
  - 4.3. Frequent Pattern Analysis Results

1. Preparing the Data

1.1. & 1.2. Combining the Source Tables

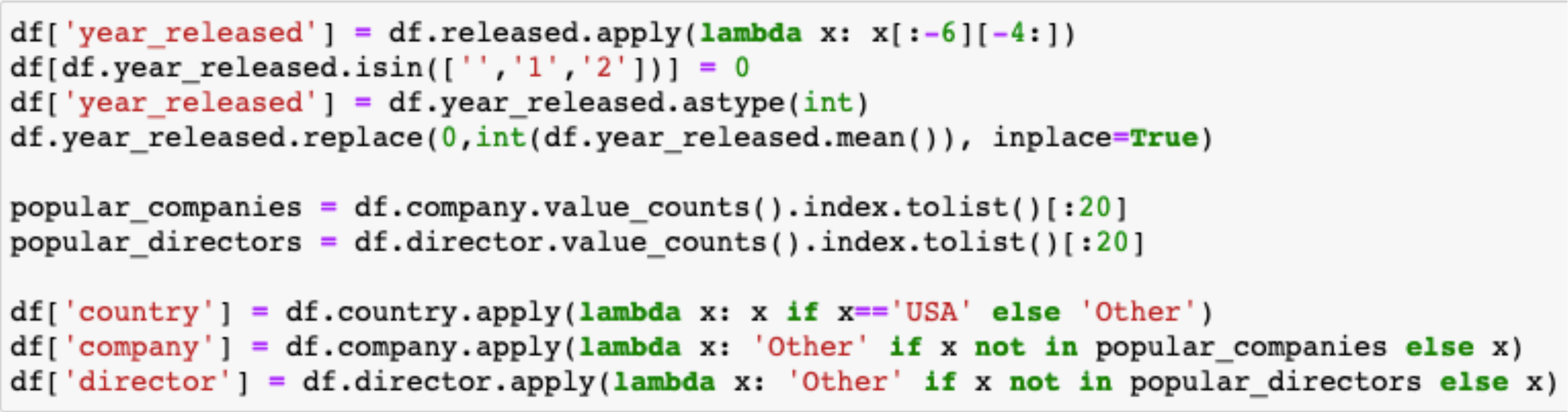
- > Reading raw data:
- metadata; records: 45466, fields: 11
- keywords; records: 46419, fields: 2
- movies; records: 4638, fields: 18
- > Removing duplicate rows from each table:
- > After the removal, we end up with the following number of rows for each table: metadata; records: 39943, fields: 11
- keywords; records: 44447, fields: 2
- movies; records: 4570, fields: 18
- > Combining movies+metadata+keywords tables using movie titles
- **df = movies + metadata + keywords --> records: 3524, fields: 30**
- ***The last table, "df" is the combined data from three sources and thus will be used in throughout the project. We will still manipulate the data in terms of preprocessing and feature engineering, but we do not expect any further reduction in number of rows. Still, it can happen due to corrupt data issues***



1. Preparing the Data

1.3. Transforming Features to their Correct Types

- By observing the data, we see that following columns need not any initial transformation:
- genre, company, budget, country, director, overview, tagline
- And the following features had to change due to:
- *released*: this is a date column with dd.mm.yyyy format. For the sake of simplicity, we are only interested in the year the movies are released. Therefore we apply a transformation to extract year information from this column and write it to *year_released* column.
- We also limited *company* and *director* columns to have at most 20 unique values, and *country* to have 2 unique values, to simplify data analysis processes.



```
df['year_released'] = df.released.apply(lambda x: x[:-6][-4:])
df[df.year_released.isin(['','1','2'])] = 0
df['year_released'] = df.year_released.astype(int)
df.year_released.replace(0,int(df.year_released.mean()), inplace=True)

popular_companies = df.company.value_counts().index.tolist()[:20]
popular_directors = df.director.value_counts().index.tolist()[:20]

df['country'] = df.country.apply(lambda x: x if x=='USA' else 'Other')
df['company'] = df.company.apply(lambda x: 'Other' if x not in popular_companies else x)
df['director'] = df.director.apply(lambda x: 'Other' if x not in popular_directors else x)
```

2. Data Exploration

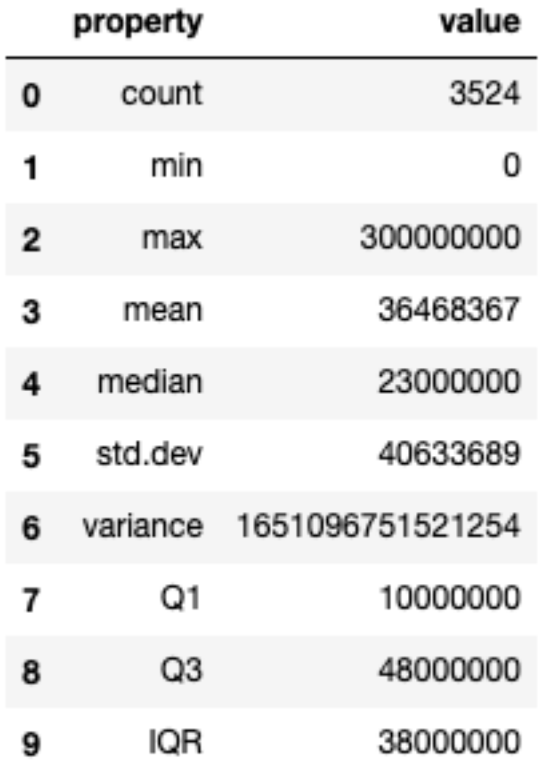
2.1. Exploring Continuous Features

- Our data has the following columns that we can use to conduct the statistical description:
  - Budget
  - Gross
  - Profitability ratio

In the following slides, we generate an initial outlook to these continuous features.

2. Data Exploration

2.1. Exploring Continuous Features - Budget



	property	value
0	count	3524
1	min	0
2	max	300000000
3	mean	36468367
4	median	23000000
5	std.dev	40633689
6	variance	1651096751521254
7	Q1	10000000
8	Q3	48000000
9	IQR	38000000

The table on the left represents the data summary based on statistical description of the feature **budget**.

There are measures of central tendency (mean, median), dispersion (quartiles, variance, standard deviation, interquartile range), and some other statistical descriptions, like minimum, maximum and count.

The count() function returns the number of cells for each row or column. The budget column has 3524 non-empty cells.

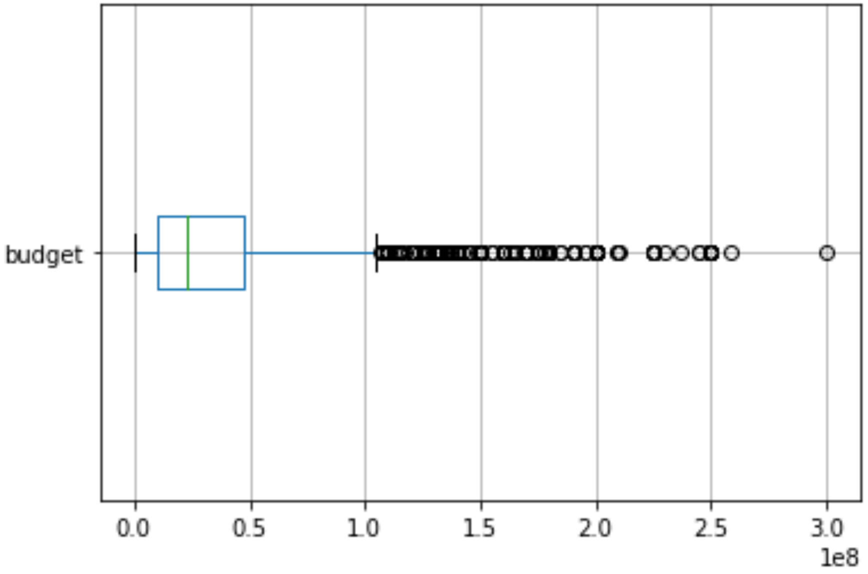
The minimum value of the budget is 0, while the maximum - 300,000,000.

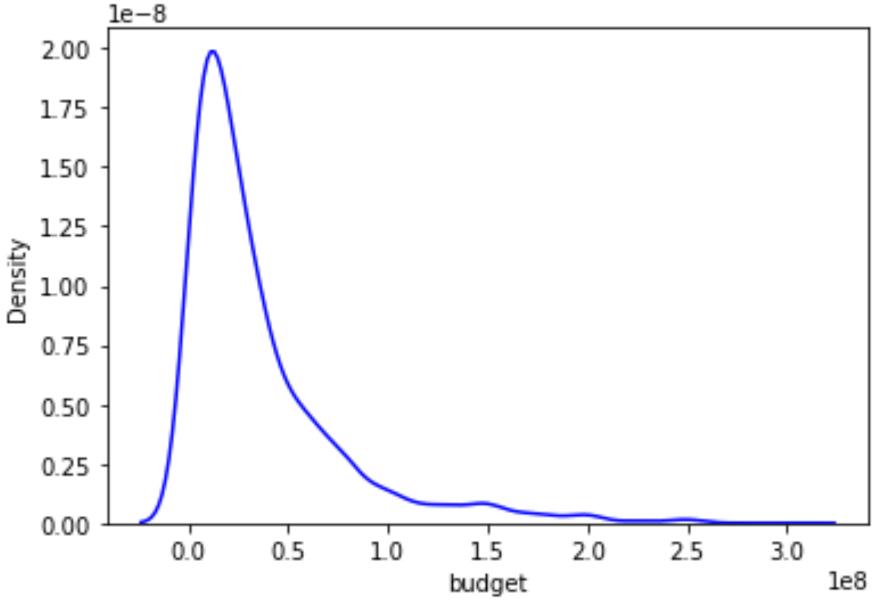
2. Data Exploration

2.1. Exploring Continuous Features - Budget

The figure on the left shows the boxplot of the feature **budget**. The graph represents five number summary: "minimum, first quartile, median, third quartile, and "maximum". The ends of the box are the quartiles, Q1 and Q3, while the box length is interquartile range (IQR). While from the graph, we can get an approximate value of Q1 and Q3, it can be seen that the exact value of Q1 is 10,000,000 and Q3 is 48,000,000 from the table in previous slide. The median is marked as the line within the box, which is equal to 23,000,000. Two lines (called whiskers) outside the box extend to the smallest (Minimum) and largest (Maximum) observations. The circles are the outliers.

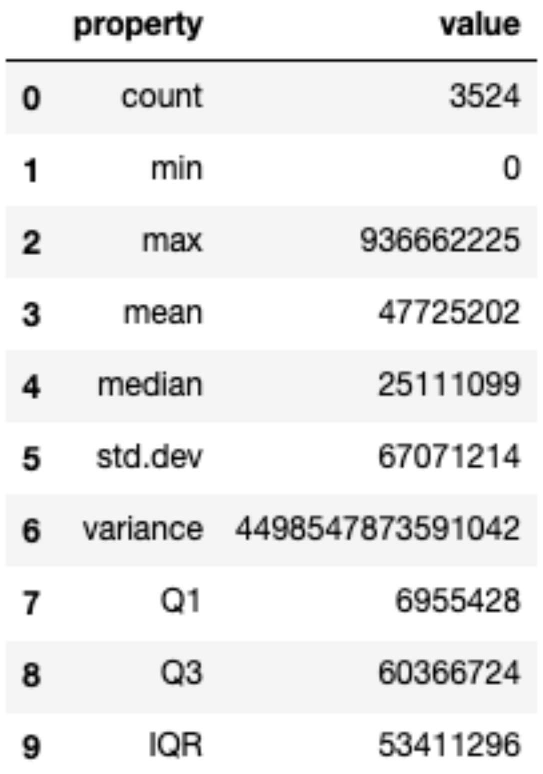
The figure on the right demonstrates that the budget's distribution is asymmetrical, because the tail is skewed to the right. As the tail is longer towards the right-hand side, the value of skewness is positive. The value of the skewness is 2.253961, which means that the distribution is highly skewed.





2. Data Exploration

2.1. Exploring Continuous Features - Gross



	property	value
0	count	3524
1	min	0
2	max	936662225
3	mean	47725202
4	median	25111099
5	std.dev	67071214
6	variance	4498547873591042
7	Q1	6955428
8	Q3	60366724
9	IQR	53411296

The table on the left represents the data summary based on statistical description of the feature **gross**.

There are measures of central tendency (mean, median), dispersion (quartiles, variance, standard deviation, interquartile range), and some other statistical descriptions, like minimum, maximum and count.

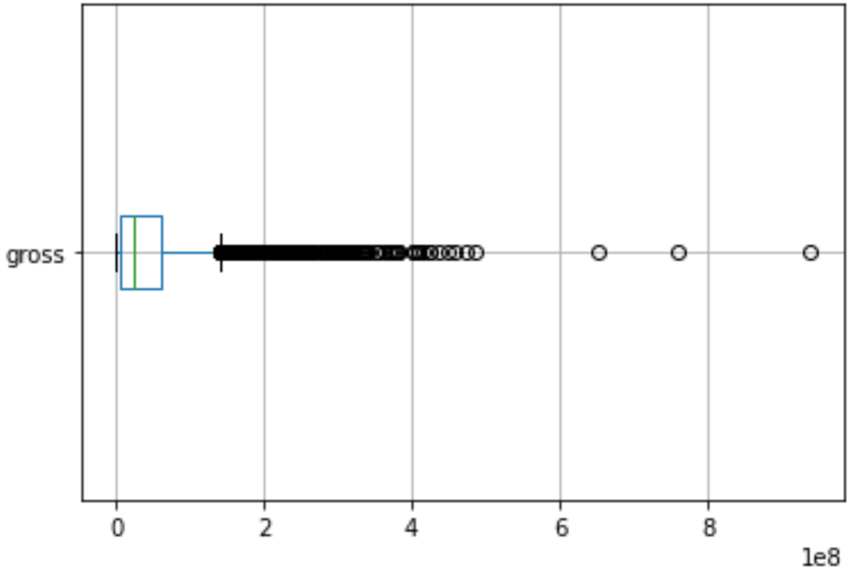
The gross column has 3524 non-empty cells. The minimum value of the budget is 0, while the maximum - 93,662,225.

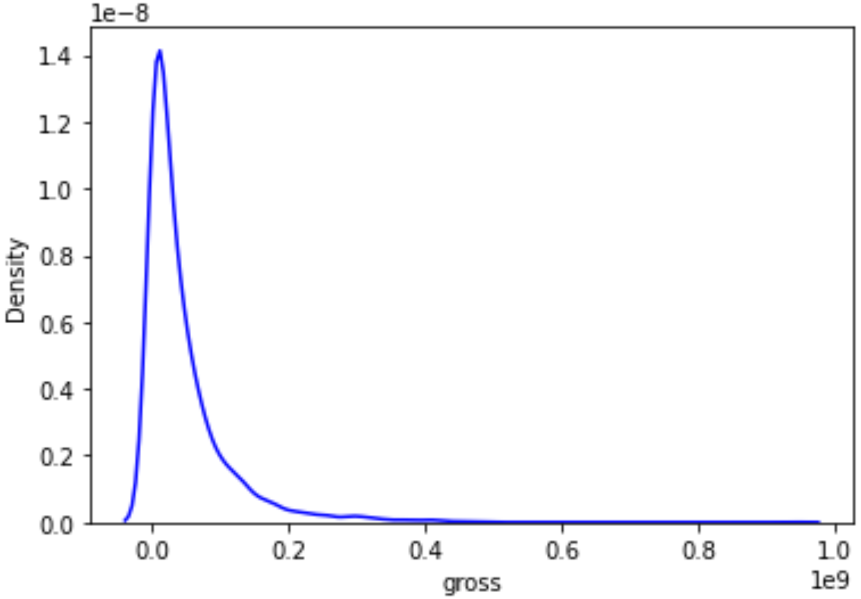
2. Data Exploration

2.1. Exploring Continuous Features - Gross

The figure on the left shows the boxplot of the feature **gross**. The graph represents five number summary: “minimum, first quartile, median, third quartile, and "maximum”. The ends of the box are the quartiles, Q1 and Q3, while the box length is interquartile range (IQR). While from the graph, we can get an approximate value of Q1 and Q3, it can be seen that the exact value of Q1 is 6,955,428 and Q3 is 60,366,724 from Figure 4. The median is marked as the line within the box, which is equal to 25,111,099. Two lines (called whiskers) outside the box extend to the smallest (Minimum) and largest (Maximum) observations. The circles are the outliers.

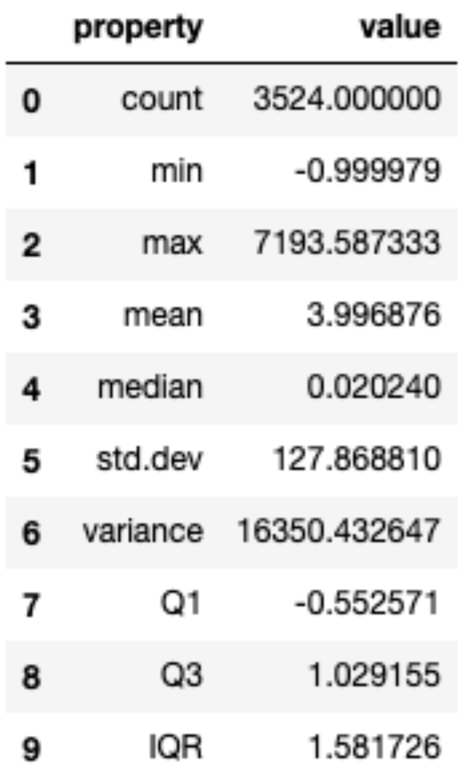
The figure on the right demonstrates that the grosst’s distribution is asymmetrical, because the tail is skewed to the right. As the tail is longer towards the right-hand side, the value of skewness is positive. The value of the skewness is **3.49795**, which means that the distribution is highly skewed.





2. Data Exploration

2.1. Exploring Continuous Features - Profitability



	property	value
0	count	3524.000000
1	min	-0.999979
2	max	7193.587333
3	mean	3.996876
4	median	0.020240
5	std.dev	127.868810
6	variance	16350.432647
7	Q1	-0.552571
8	Q3	1.029155
9	IQR	1.581726

The table on the left represents the data summary based on statistical description of the feature **profitability ratio**.

There are measures of central tendency (mean, median), dispersion (quartiles, variance, standard deviation, interquartile range), and some other statistical descriptions, like minimum, maximum and count.

The profitability ratio column has 3524 non-empty cells.

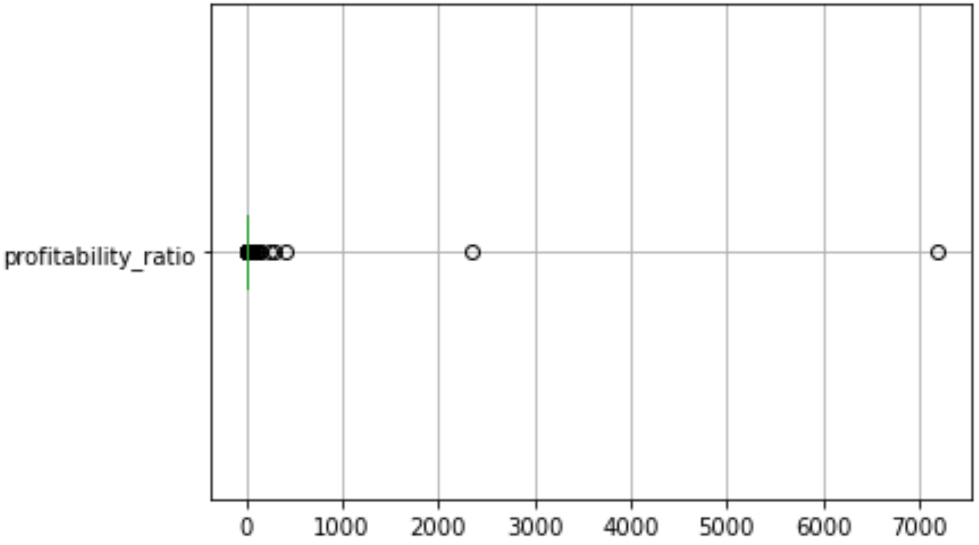
The minimum value of the budget is - 0.999979, while the maximum - 7,193.587333.

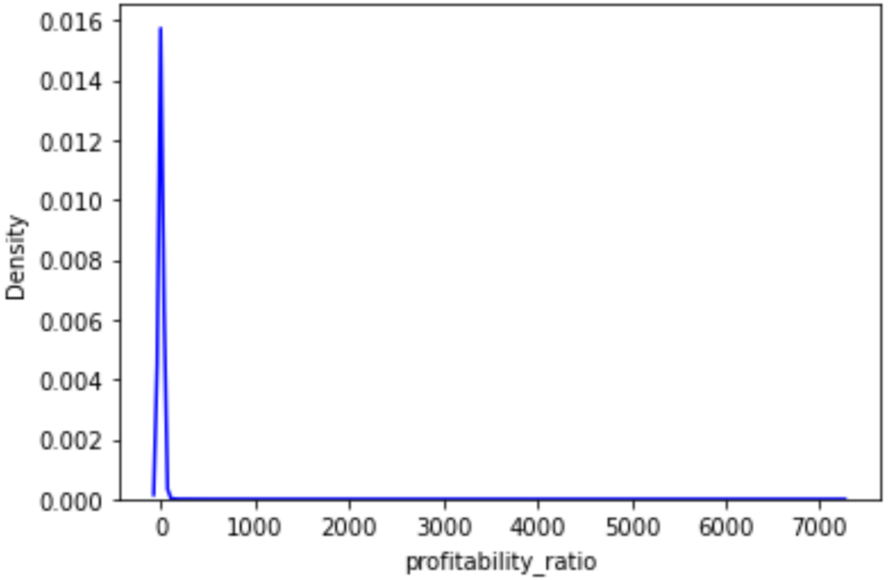
2. Data Exploration

2.1. Exploring Continuous Features - Profitability

The figure on the left shows the boxplot of the feature **profitability ratio**. The graph represents five number summary: “minimum, first quartile, median, third quartile, and "maximum”. The ends of the box are the quartiles, Q1 and Q3, while the box length is interquartile range (IQR). While from the graph, we can get an approximate value of Q1 and Q3, it can be seen that the exact value of Q1 is -0.552571 and Q3 is 1.029155 from Figure 7. The median is marked as the line within the box, which is equal to 0.02024. Two lines (called whiskers) outside the box extend to the smallest (Minimum) and largest (Maximum) observations. The circles are the outliers.

The figure on the right demonstrates that the profitability ratio’s distribution is asymmetrical, because the tail is skewed to the right. As the tail is longer towards the right-hand side, the value of skewness is positive. The value of the skewness is 52.233907, which means that the distribution is highly skewed.

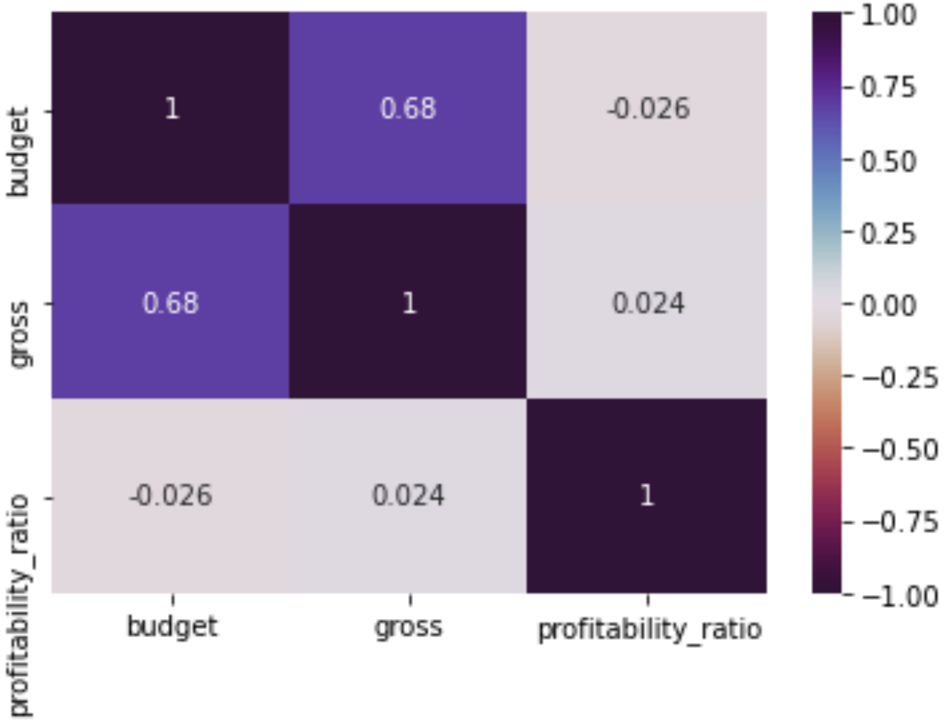




2. Data Exploration

2.1. Exploring Continuous Features - Correlations

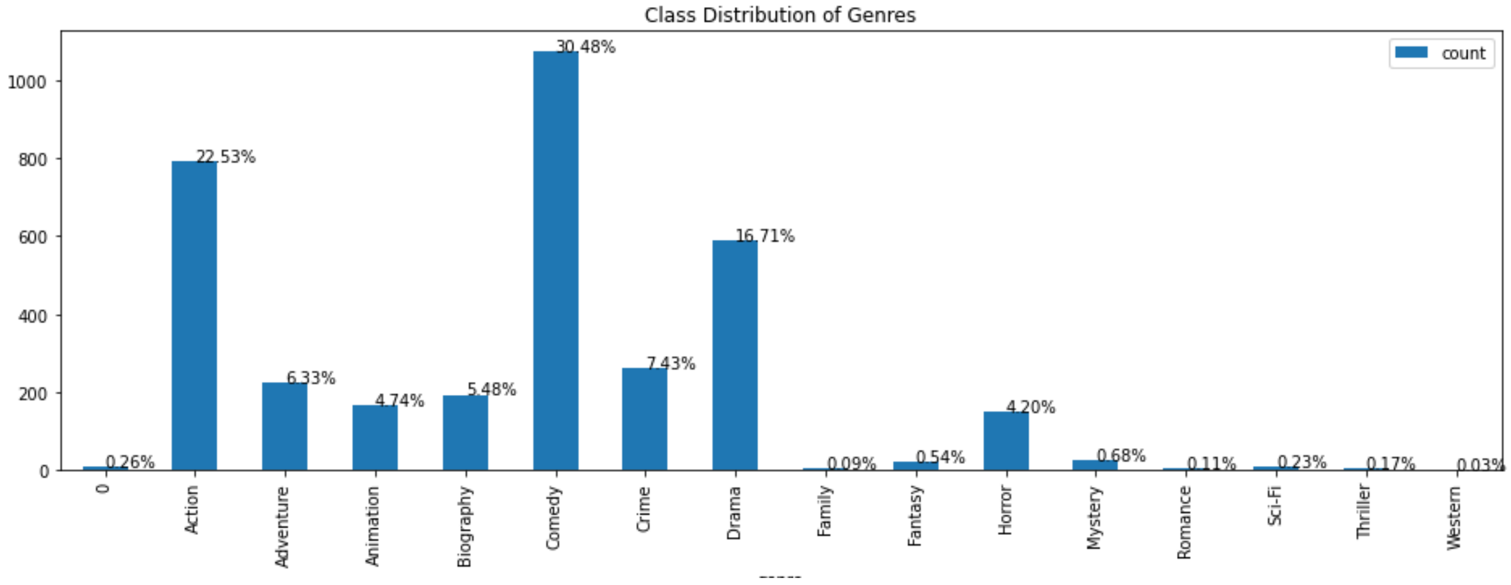
- The figure on the right shows the correlation matrix of our continuous features.
- As can be noticed, gross and budget are positively correlated with 68%.
- Profitability ratio's correlation with budget and gross are very small so that they are negligible.



2. Data Exploration

2.1. Exploring Categorical Features - Genre

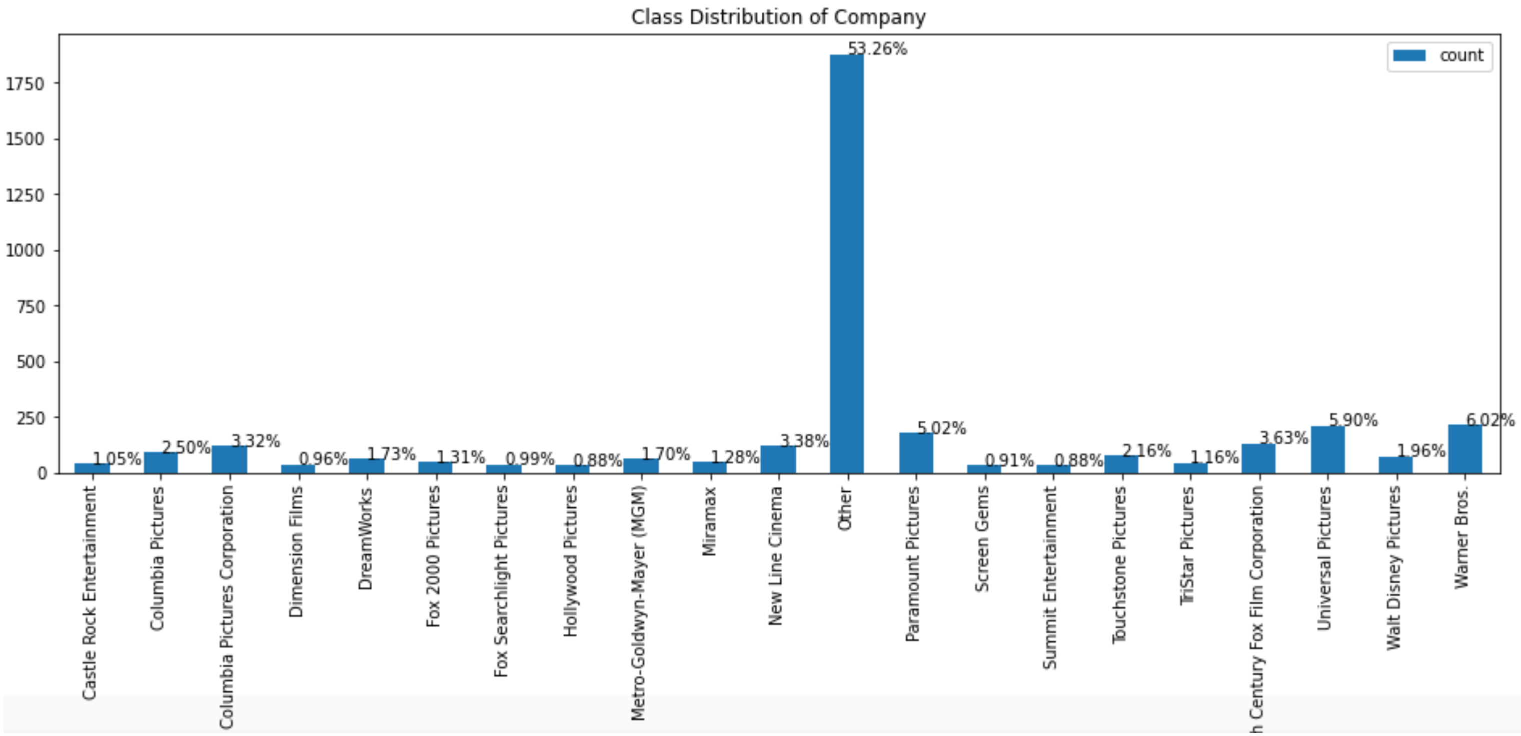
- The bar chart below shows the distribution of **genre** among our dataset, and subsequently we share the imbalances of each genre.



- Action is Slightly Imbalanced
- Adventure is Severely Imbalanced
- Animation is Severely Imbalanced
- Biography is Severely Imbalanced
- Comedy is balanced
- Crime is Severely Imbalanced
- Drama is Slightly Imbalanced
- Family is Severely Imbalanced
- Fantasy is Severely Imbalanced
- Horror is Severely Imbalanced
- Mystery is Severely Imbalanced
- Romance is Severely Imbalanced
- Sci-Fi is Severely Imbalanced
- Thriller is Severely Imbalanced
- Western is Severely Imbalanced

2. Data Exploration
2.1. Exploring Categorical Features - Company

- The bar chart below shows the distribution of **company** among our dataset, and subsequently we share the imbalances of each company.

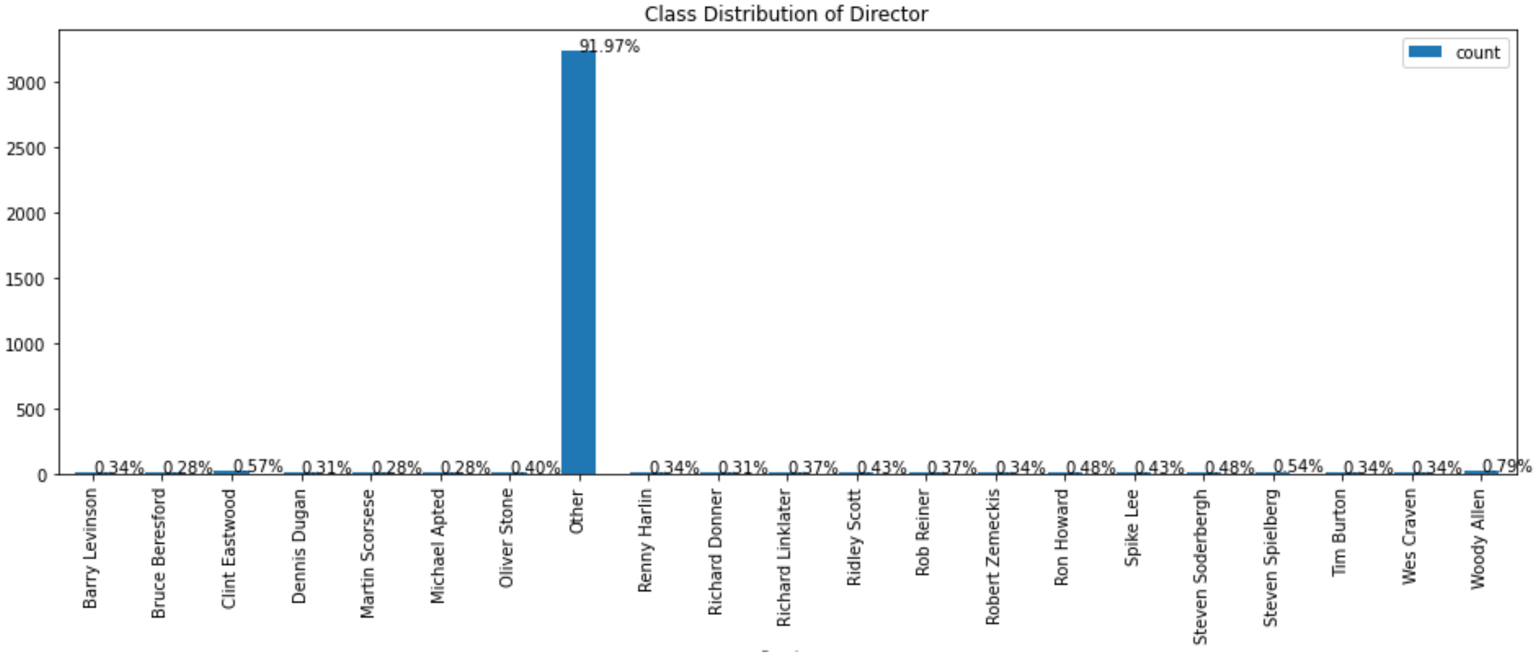


- Castle Rock Entertainment is Severely Imbalance
- Columbia Pictures is Severely Imbalance
- Columbia Pictures Corporation is Severely Imbalance
- Dimension Films is Severely Imbalance
- DreamWorks is Severely Imbalance
- Fox 2000 Pictures is Severely Imbalance
- Fox Searchlight Pictures is Severely Imbalance
- Hollywood Pictures is Severely Imbalance
- Metro-Goldwyn-Mayer (MGM) is Severely Imbalance
- Miramax is Severely Imbalance
- New Line Cinema is Severely Imbalance
- Other is balanced
- …

2. Data Exploration

2.1. Exploring Categorical Features - Director

- The bar chart below shows the distribution of **director** among our dataset, and subsequently we share the imbalances of each director.

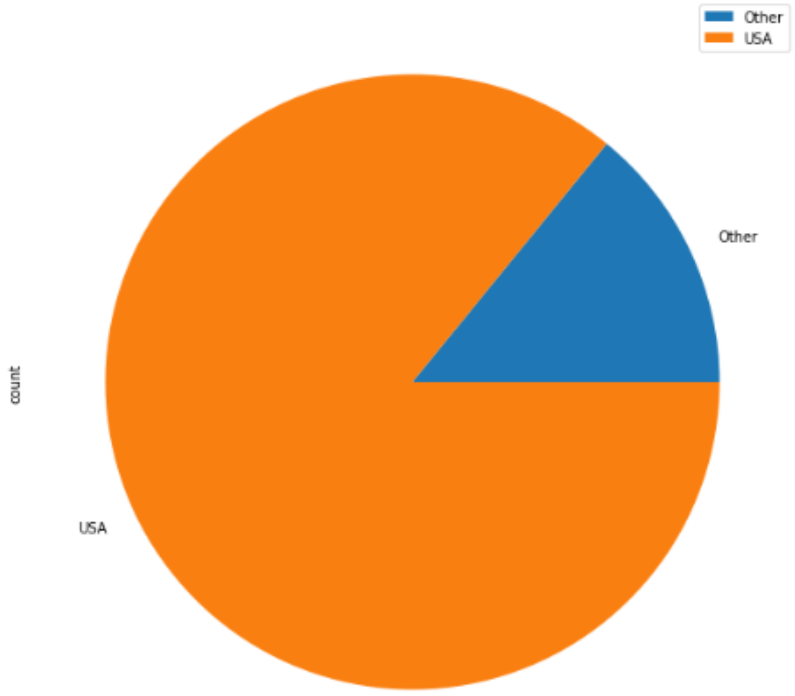


- Barry Levinson is Severely Imbalance
- Bruce Beresford is Severely Imbalance
- Clint Eastwood is Severely Imbalance
- Dennis Dugan is Severely Imbalance
- Martin Scorsese is Severely Imbalance
- Michael Apted is Severely Imbalance
- Oliver Stone is Severely Imbalance
- Other is balanced
- Renny Harlin is Severely Imbalance
- Richard Donner is Severely Imbalance
- …

2. Data Exploration

2.1. Exploring Categorical Features - Country

- The bar chart below shows the distribution of **country** among our dataset, and subsequently we share the imbalances.

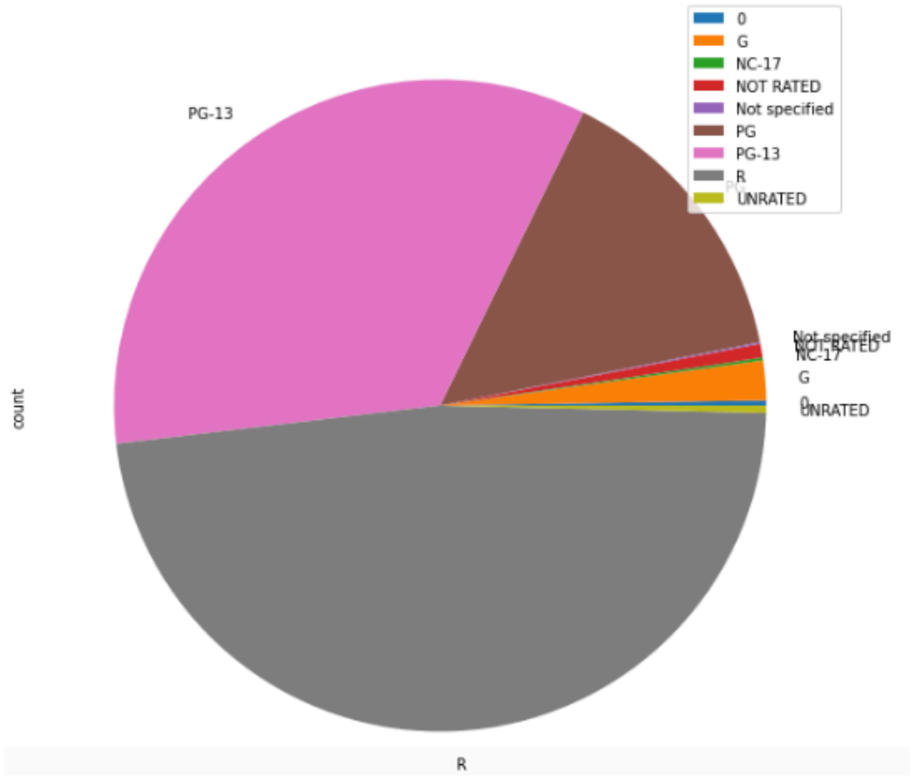


- Other is Slightly Imbalance
- USA is balanced

2. Data Exploration

2.1. Exploring Categorical Features - Rating

- The bar chart below shows the distribution of **rating** among our dataset, and subsequently we share the imbalances.

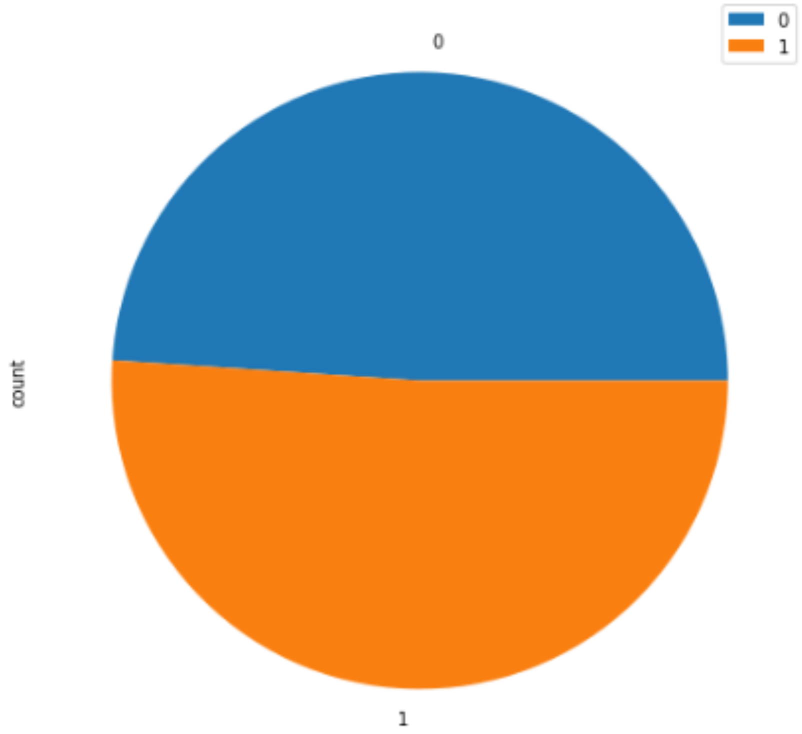


- G is Severely Imbalance
- NC-17 is Severely Imbalance
- NOT RATED is Severely Imbalance
- PG is Slightly Imbalance
- PG-13 is balanced
- R is balanced
- UNRATED is Severely Imbalance

2. Data Exploration

2.1. Exploring Categorical Features – Isprofit

- The bar chart below shows the distribution of **isprofit** among our dataset, and subsequently we share the imbalances.



- 0.0 (False) is balanced
- 1.0 (True) is balanced

2. Data Exploration

2.2. Categorical Feature Dependency Using Chi-Square Test

Our data has the following columns that we can use to conduct the chi-square test on:

- Genre
- Company
- Country
- Director
- Rating
- Released
- Isprofit

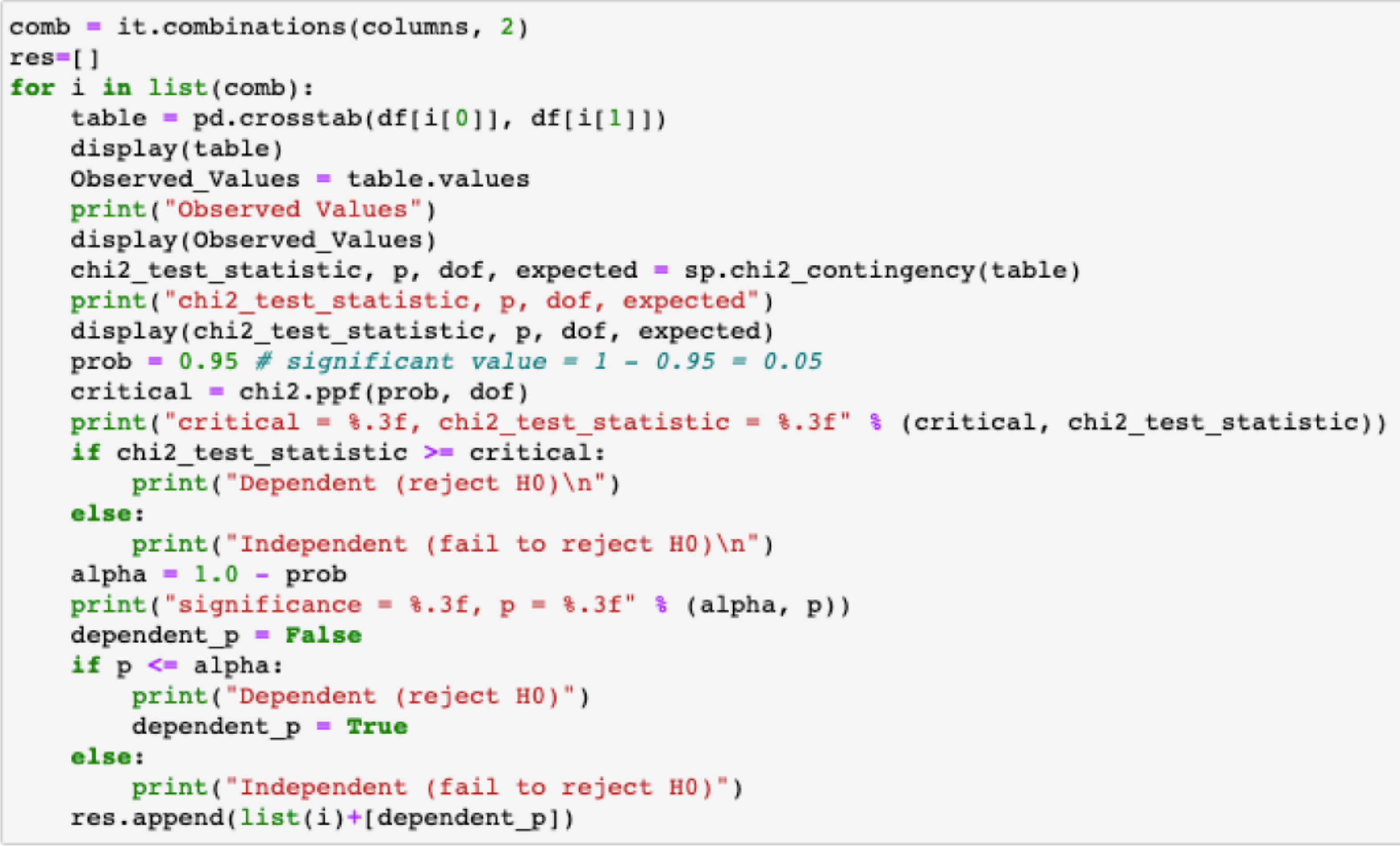
> We want to find out whether two categorical attributes are independent or dependent by using the chi-square test. We do this by getting all combinations of length two from the list above and conducting the chi-square test on each of the categorical pairs.

> After writing some general formulas, we will conduct the test on all 21 combinations.

2. Data Exploration

2.2. Categorical Feature Dependency Using Chi-Square Test

Using the code on the right, we conduct chi-square test for each pair. Please refer to the summary at the end of this section, or the HTML report for detailed results of the chi-square tests. The summary table contains the information of dependency while the HTML report also contains the test statistics and p-values.

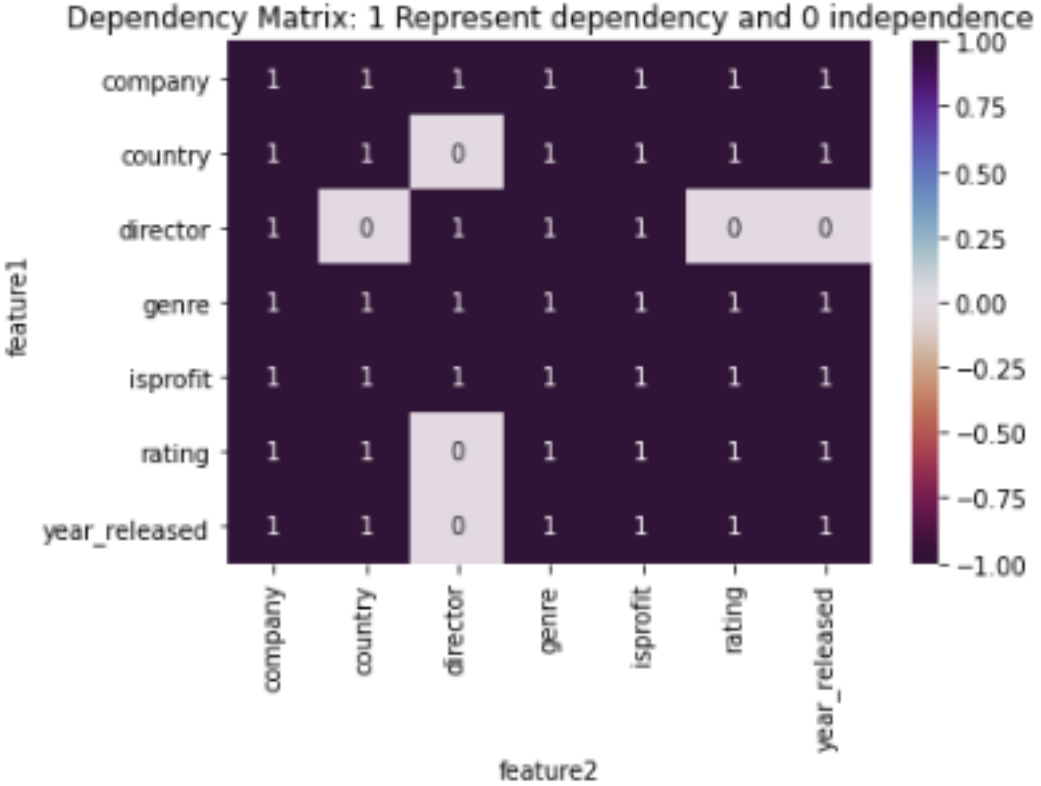


```
comb = it.combinations(columns, 2)
res=[]
for i in list(comb):
    table = pd.crosstab(df[i[0]], df[i[1]])
    display(table)
    Observed_Values = table.values
    print("Observed Values")
    display(Observed_Values)
    chi2_test_statistic, p, dof, expected = sp.chi2_contingency(table)
    print("chi2_test_statistic, p, dof, expected")
    display(chi2_test_statistic, p, dof, expected)
    prob = 0.95 # significant value = 1 - 0.95 = 0.05
    critical = chi2.ppf(prob, dof)
    print("critical = %.3f, chi2_test_statistic = %.3f" % (critical, chi2_test_statistic))
    if chi2_test_statistic >= critical:
        print("Dependent (reject H0)\n")
    else:
        print("Independent (fail to reject H0)\n")
    alpha = 1.0 - prob
    print("significance = %.3f, p = %.3f" % (alpha, p))
    dependent_p = False
    if p <= alpha:
        print("Dependent (reject H0)")
        dependent_p = True
    else:
        print("Independent (fail to reject H0)")
    res.append(list(i)+[dependent_p])
```

2. Data Exploration

2.2. Categorical Feature Dependency Using Chi-Square Test

- In results of chi-square test (see figure on the right), we observe there are quite enough couple of features that are dependent and independent.
- For example, *company* and *country* are correlated as we initially expected, because most of the movies in our data is made in the US.



Dependency Matrix: 1 Represent dependency and 0 independence

feature1 \ feature2	company	country	director	genre	isprofit	rating	year_released
company	1	1	1	1	1	1	1
country	1	1	0	1	1	1	1
director	1	0	1	1	1	0	0
genre	1	1	1	1	1	1	1
isprofit	1	1	1	1	1	1	1
rating	1	1	0	1	1	1	1
year_released	1	1	0	1	1	1	1

3. Frequent Pattern Analysis

Our data has the following columns that we can perform frequent pattern analysis on:

- Genres_edited
- Spoken_languages_edited
- Keywords_edited
- Overview

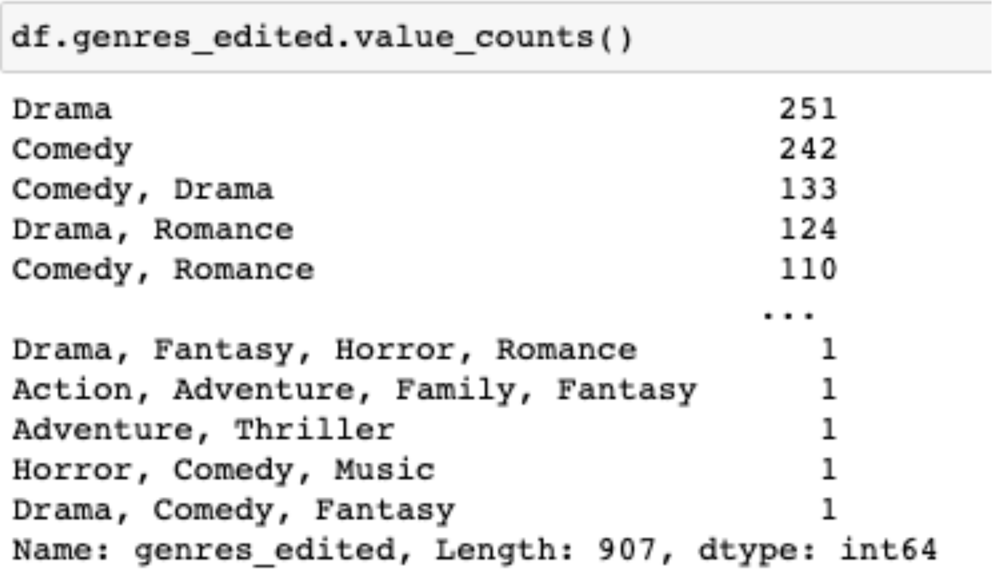
> We want to find out whether there are recurring relationships in the data. For example, which genres are frequently used together to produce movies? Or which keyword combinations are frequent? After performing the analyses, we can decide, for example, which genres to produce in order to make the most profit.

> After writing some general functions, we will perform the analyses one by one.

3. Frequent Pattern Analysis

3.1. Genres

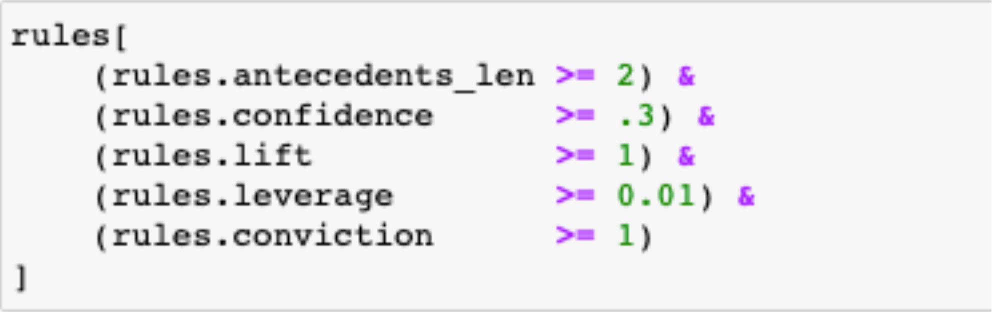
First, we show how does our *genres* feature look like in the figure below.



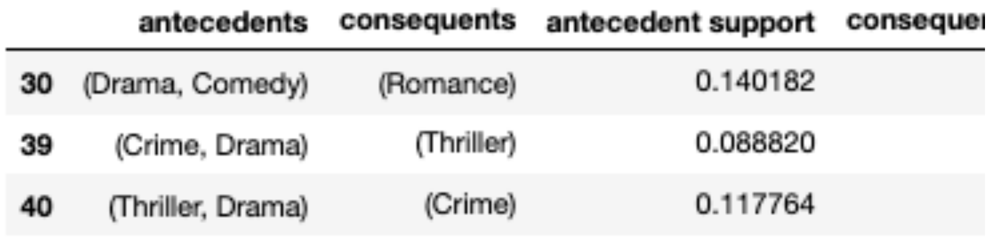
```
df.genres_edited.value_counts()
```

```
Drama                                     251
Comedy                                    242
Comedy, Drama                             133
Drama, Romance                            124
Comedy, Romance                           110
                                         ...
Drama, Fantasy, Horror, Romance             1
Action, Adventure, Family, Fantasy          1
Adventure, Thriller                         1
Horror, Comedy, Music                       1
Drama, Comedy, Fantasy                      1
Name: genres_edited, Length: 907, dtype: int64
```

After applying apriori algorithm and calculating the association metrics, we generate the following rules:



```
rules[
    (rules.antecedents_len >= 2) &
    (rules.confidence      >= .3) &
    (rules.lift            >= 1) &
    (rules.leverage        >= 0.01) &
    (rules.conviction      >= 1)
]
```

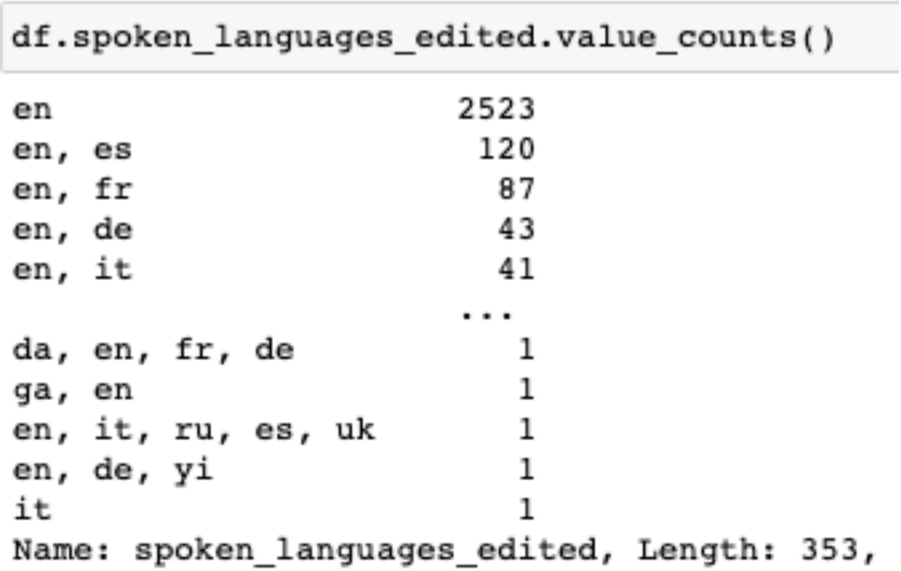


	antecedents	consequents	antecedent support	consequen
30	(Drama, Comedy)	(Romance)	0.140182	
39	(Crime, Drama)	(Thriller)	0.088820	
40	(Thriller, Drama)	(Crime)	0.117764	

3. Frequent Pattern Analysis

3.2. Spoken Languages

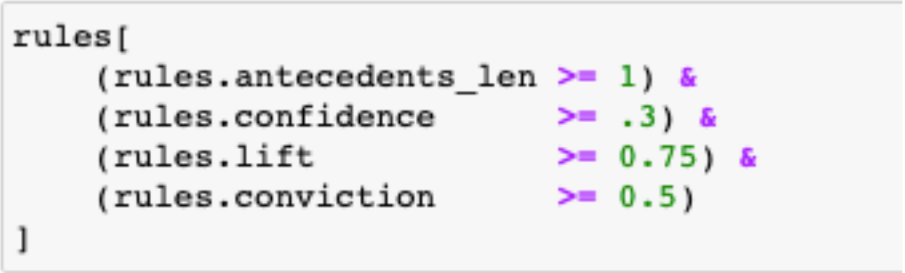
First, we show how does our *spoken languages* feature look like in the figure below.



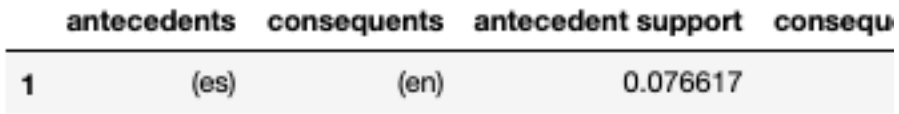
```
df.spoken_languages_edited.value_counts()
```

```
en                      2523
en, es                   120
en, fr                    87
en, de                    43
en, it                    41
                        ...
da, en, fr, de             1
ga, en                     1
en, it, ru, es, uk         1
en, de, yi                 1
it                         1
Name: spoken_languages_edited, Length: 353,
```

After applying apriori algorithm and calculating the association metrics, we generate the following rule. Although the rule is not very strong, it represent a somewhat frequent pattern:



```
rules[
    (rules.antecedents_len >= 1) &
    (rules.confidence      >= .3) &
    (rules.lift            >= 0.75) &
    (rules.conviction      >= 0.5)
]
```

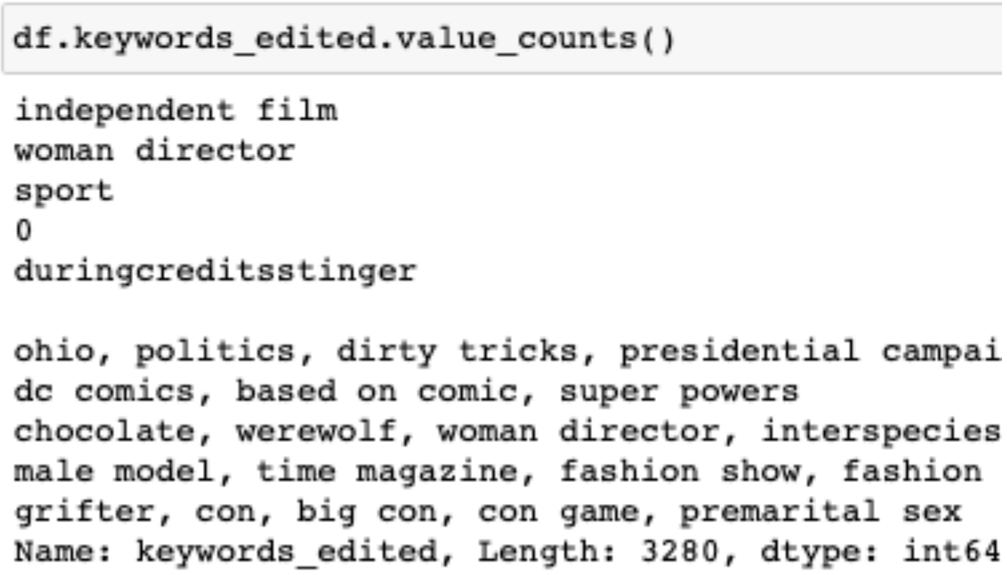


	antecedents	consequents	antecedent support	consequ
1	(es)	(en)	0.076617	

3. Frequent Pattern Analysis

3.3. Keywords

First, we show how does our *keywords* feature look like in the figure below.



```
df.keywords_edited.value_counts()
```

```
independent film
woman director
sport
0
duringcreditsstinger

ohio, politics, dirty tricks, presidential campai
dc comics, based on comic, super powers
chocolate, werewolf, woman director, interspecies
male model, time magazine, fashion show, fashion
grifter, con, big con, con game, premarital sex
Name: keywords_edited, Length: 3280, dtype: int64
```

For the keywords, we cannot create a strong rule, due to:

the data is not clean enough

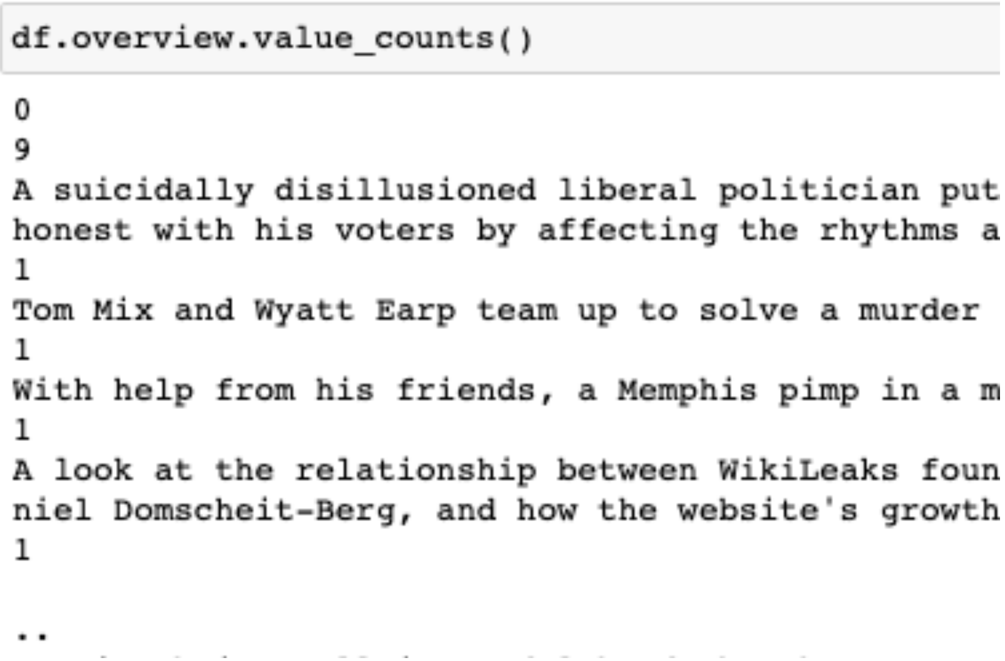
and/or

the keywords specified by different individuals so we cannot find a good association rule.

3. Frequent Pattern Analysis

3.4. Overview

First, we show how does our *overview* feature look like in the figure below.

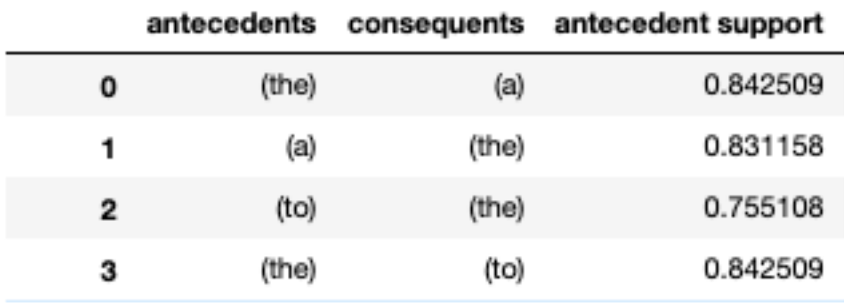


```
df.overview.value_counts()
```

```
0
9
A suicidally disillusioned liberal politician put
honest with his voters by affecting the rhythms a
1
Tom Mix and Wyatt Earp team up to solve a murder
1
With help from his friends, a Memphis pimp in a m
1
A look at the relationship between WikiLeaks foun
niel Domscheit-Berg, and how the website's growth
1

..
```

After applying apriori algorithm and calculating the association metrics, we generate the following rule. the *overview* feature consists of plain English sentences, therefore, although we find so many (~70 thousand) strong association rules, they are meaningless. All the strong rules in *overview* are the prefixes and conjunctions, like "a", "an", "the", "and", etc.



	antecedents	consequents	antecedent support
0	(the)	(a)	0.842509
1	(a)	(the)	0.831158
2	(to)	(the)	0.755108
3	(the)	(to)	0.842509

4. Results

4.1. Data Exploration Results - Continuous Features

- As we analyze three continuous features in Section 2.1., namely *budget*, *gross (revenue)*, and *profitability_ratio*, the data seems clean enough to build our model on. Even though all the three features are positively skewed, we do not observe so many outliers that will disturb training the model.
- Our initial hypothesis was that our three continuous features were meant to be correlated, because intiutively, the budget and gross and profitability of a movie sound correlated. That is, however, not the case, at least for profitability ratio and the others.
- Nonetheless, we can conclude that budget and gross are correlated. Although this information does not help us predicting the profitability ratio, we understand how independent the profitability ratio is and take it into account when we build our Machine Learning models.

4. Results

4.2. Data Exploration Results - Categorical Features

- After analyzing our categorical features in Section 2.2., we see somewhat imbalance in our 6 features. Although we have to take the imbalance problem while modeling, the one feature, *isprofit*, turns out to be very balanced. Since the *isprofit* feature is our **target feature**, we will choose a metric that objectively evaluates model performance in our classification problem.
- After calculating the imbalances, we performed chi-square tests for each pair possible out of our 7 categorical features. In results of chi-square test (shared in Section 2.2.2.2.), we observe there are quite enough couple of features that are dependent and independent. For example, *company* and *country* are correlated as we initially expected, because most of the movies in our data is made in the US.

4. Results

4.3. Frequent Pattern Analysis Results

- Our data included a set of features that we could perform frequent pattern analysis on, which we used for building association rules in Section 3.
- Since our dataset is not a kind of transactional data, these rules are not meant to be used for modeling. However, we still wanted to conduct this analysis in order to turn the learnings from our lectures into practice.
- For feature *genres* (Section 3.1.), we built the following association rules, after setting thresholds for confidence, leverage, conviction, and lift:
- (Family) => (Comedy)
- (Romance) => (Drama)
- (Mystery) => (Thriller)
- (Crime, Drama) => (Thriller)
- When we think about those rules, they actually makes sense! For example, most of the family movies have comedy inside as well.