# Pattern Recognition Project
# Milestone 1

# **Movie Popularity Prediction**

## 1) Merge the Two Datasets

join by the columns: ‘movie id’ (dataset: movies-credit) and ‘id’ (dataset: movies-regression) for manipulation later by extraction top 20 cast members

```
bonus_table.rename(columns = {'movie_id' : 'id'}, inplace = True)
table = bonus_table.merge(table)
```

## 2) **Year, Month Columns**

We added two new columns ‘year’ & ‘month’ from the ‘release_date’ column and dropped the ‘release_date’ column from the dataset

```
0       2009
1       2015
2       2012
3       2012
4       2007
        ...
3028    2002
3029    2010
3030    1997
3031    2014
3032    2011
Name: year, Length: 3033, dtype: int64
0       10
1       10
2        7
3        7
4        1
        ..
3028     9
3029     9
3030     6
3031     1
3032    11
Name: month, Length: 3033, dtype: int64
```

## 3) **Split the Data:**

Using : `train_test_split()` on training size = 0.8, testing size = 0.2

trainTable → data frame contains xTrain and yTrain

trainTable.head(2)

```
         id                          title  \
491   72331  Abraham Lincoln: Vampire Hunter
1611   2176                 The Glass House

                                                  cast  \
491   [{"cast_id": 8, "character": "Abraham Lincoln"...
1611  [{"cast_id": 1, "character": "Ruby Baker", "cr...

                                                  crew    budget  \
491   [{"credit_id": "52fe4864c3a368484e0f66a9", "de...  69000000
1611  [{"credit_id": "58162f2b9251415abb00ea89", "de...  30000000

                                                genres homepage  \
491   [{"id": 28, "name": "Action"}, {"id": 14, "nam...      NaN
1611  [{"id": 18, "name": "Drama"}, {"id": 53, "name...      NaN

                                              keywords original_language  \
491   [{"id": 840, "name": "usa president"}, {"id": ...                en
1611  [{"id": 387, "name": "california"}, {"id": 115...                en

                       original_title  ...  \
491   Abraham Lincoln: Vampire Hunter  ...
1611                  The Glass House  ...

                                  production_countries    revenue runtime  \
491   [{"iso_3166_1": "US", "name": "United States o...  112265139    94.0
1611  [{"iso_3166_1": "US", "name": "United States o...   23619609   106.0
```

testTable → data frame contains xTest and yTest

testTable.head(2)

```
         id              title  \
123   10048            Stealth
35   105864  The Good Dinosaur

                                                  cast  \
123  [{"cast_id": 21, "character": "Lt. Ben Gannon"...
35   [{"cast_id": 20, "character": "Arlo (voice)", ...

                                                  crew     budget  \
123  [{"credit_id": "52fe43139251416c75002771", "de...  135000000
35   [{"credit_id": "52fe4a58c3a36847f81c8a07", "de...  175000000

                                                genres  \
123                         [{"id": 28, "name": "Action"}]
35   [{"id": 12, "name": "Adventure"}, {"id": 16, "...

                                       homepage  \
123                                         NaN
35   http://movies.disney.com/the-good-dinosaur

                                              keywords original_language  \
123  [{"id": 310, "name": "artificial intelligence"...                en
35   [{"id": 1720, "name": "tyrannosaurus rex"}, {"...                en

        original_title  ...  \
123            Stealth  ...
35   The Good Dinosaur  ...
```

4) **Preprocessing**

- Applied on training and testing data to have the same number of features.

**a. Null values detection**

- The picture below shows the total number of null values in each column.

```
id                          0
title                       0
cast                        0
crew                        0
budget                      0
genres                      0
homepage                 1908
keywords                    0
original_language           0
original_title              0
overview                    1
viewercount                 0
production_companies        0
production_countries        0
release_date                0
revenue                     0
runtime                     1
spoken_languages            0
status                      0
tagline                   383
vote_count                  0
vote_average                0
dtype: int64
```

**b. Fill with the mean values and convert values to numeric by Label Encoding:**

- The picture below shows the dataset after encoding

```
           id  title                                               cast  \
 374   10197   1223  [{"cast_id": 15, "character": "Lilli", "credit...
 2908  39269   1277  [{"cast_id": 2, "character": "Don", "credit_id...
 1158  85446   1641  [{"cast_id": 1010, "character": "Emily", "cred...

                                                   crew    budget  \
 374   [{"credit_id": "52fe43409251416c7500933d", "de...  80000000
 2908  [{"credit_id": "52fe47099251416c9106826f", "de...         0
 1158  [{"credit_id": "52fe49419251416c910a76d7", "de...  33000000

                                                 genres  homepage  \
 374   [{"id": 18, "name": "Drama"}, {"id": 10402, "n...       129
 2908                      [{"id": 18, "name": "Drama"}]       899
 1158  [{"id": 10402, "name": "Music"}, {"id": 18, "n...       899

                                               keywords  original_language  \
 374   [{"id": 10937, "name": "memory"}, {"id": 16573...                  4
 2908  [{"id": 4470, "name": "punk"}, {"id": 10183, "...                  4
 1158  [{"id": 186730, "name": "flash mob"}, {"id": 1...                  4

       original_title  ...                               production_countries  \
 374             1218  ...  [{"iso_3166_1": "IT", "name": "Italy"}, {"iso_...
 2908            1268  ...           [{"iso_3166_1": "CA", "name": "Canada"}]
 1158            1628  ...  [{"iso_3166_1": "US", "name": "United States o...

         revenue runtime                                   spoken_languages  \
 374    53825515   112.0  [{"iso_639_1": "en", "name": "English"}, {"iso...
 2908          0    94.0           [{"iso_639_1": "en", "name": "English"}]
```

**c. Final dataset**

- The picture below shows the dataset null values after label encoding and filling the null values with the mean values

```
budget                  0
genres                  0
homepage                0
id                      0
keywords                0
original_language       0
original_title          0
overview                0
viewercount             0
production_companies    0
production_countries    0
release_date            0
revenue                 0
runtime                 0
spoken_languages        0
status                  0
tagline                 0
title                   0
vote_count              0
vote_average            0
dtype: int64
```

### d. Top 20 Records

- For each column of List data type in the dataset, where for each List we determine the top 20 values for each key with the highest frequencies and create a new feature for those values.
- We iterate on the list by using json.
- **For example:**
    - Using ‘Crew’ Column from movies-credit dataset.
- Creating a new feature with the top 20 directors with the highest number of movies directed in the dataset.
- **For example:**
    - Using ‘Cast’ Column from movies-credit dataset.
- Creating a new feature with the top 20 cast with order 0 with the highest number of movies directed in the dataset
- Applying this on training and testing data to have the same number of features in (allmovies_df , allmovies_df_test).

| | id | title | cast | crew | budget | genres | homepage | keywords | original_language |
|---|---|---|---|---|---|---|---|---|---|
| **1041** | 259694 | 212 | DakotaJohnson | ChristianDitter | 38000000 | Comedy,Romance | 15 | new york,based on novel,one-night stand,single | 2 |
| **893** | 79 | 202 | JetLi | ZhangYimou | 31000000 | Drama,Adventure,Action,History | 221 | countryside,loss of lover,right and justice,pa... | 11 |

2 rows × 23 columns

e. **Data Normalization and dropping strings** (specially for modeling):
- Where the following columns were dropped as their data type is String:
- ['genres', 'keywords','spoken_languages', 'production_companies' , 'production_countries', 'cast' , 'crew']

- Data after normalization and dropping:

```
            id     title    budget  homepage  original_language  \
0     0.022800  0.504538  0.285714       0.0           0.210526
1     0.087835  0.526815  0.000000       0.0           0.210526
2     0.191134  0.676980  0.117857       0.0           0.210526
3     0.606469  0.135726  0.892857       0.0           0.210526
4     0.025815  0.600248  0.032143       0.0           0.210526
...        ...       ...       ...       ...                ...
2421  0.041902  0.818894  0.060714       0.0           0.210526
2422  0.027070  0.856023  0.114286       0.0           0.210526
2423  0.252659  0.601073  0.100000       0.0           0.210526
2424  0.024111  0.443894  0.232143       0.0           0.210526
2425  0.023142  0.981023  0.107143       0.0           0.210526

      original_title  overview  viewercount   revenue   runtime  ...  India  \
0           0.502268  0.279588     0.010687  0.019306  0.331361  ...    0.0
1           0.522887  0.376082     0.000775  0.000000  0.278107  ...    0.0
2           0.671340  0.415670     0.034006  0.050385  0.292899  ...    0.0
3           0.133196  0.437526     0.226560  0.413673  0.434911  ...    0.0
4           0.595876  0.936082     0.020888  0.006132  0.275148  ...    0.0
...              ...       ...          ...       ...       ...  ...    ...
2421        0.808247  0.607835     0.013622  0.019919  0.278107  ...    0.0
2422        0.844536  0.104330     0.018814  0.006654  0.316568  ...    0.0
2423        0.596701  0.188041     0.043090  0.035006  0.340237  ...    0.0
2424        0.441237  0.700206     0.003372  0.000000  0.310651  ...    0.0
2425        0.964124  0.774021     0.006007  0.000000  0.295858  ...    0.0
```

5) **Feature Selection:**
   - Correlation table between all features and ‘vote_average’ feature
   - Runtime, vote_count and viewer count are mostly correlated for vote_average.

```
vote_average          1.000000
runtime               0.413366
vote_count            0.372155
viewercount           0.303218
Drama                 0.281629
revenue               0.204972
Comedy                0.194653
History               0.132144
year                  0.129613
Horror                0.125754
War                   0.121860
United Kingdom        0.116108
Action                0.108980
Deutsch               0.087820
month                 0.086715
0.072311              العربية
Family                0.069376
TomHanks              0.068477
Crime                 0.064649
original_language     0.062824
Name: vote_average, dtype: float64
```

6) **Models**

- Linear Regression:
  - With mean square error (MSE) = 2.4365600694863505

```
The Mean Square Error (MSE) =  2.4365600694863505
```

- Lasso Regression:
  - With Mean Square Error (MSE) = 0.8247471287772892

```
The Mean Square Error (MSE) =  0.8247471287772892
```

- As shown from model results Lasso regression is better than Linear regression as it gives Mean Square Error with 0.8247471287772895.

# Milestone 2

# **Classification Models**

- `RandomForestClassifier`

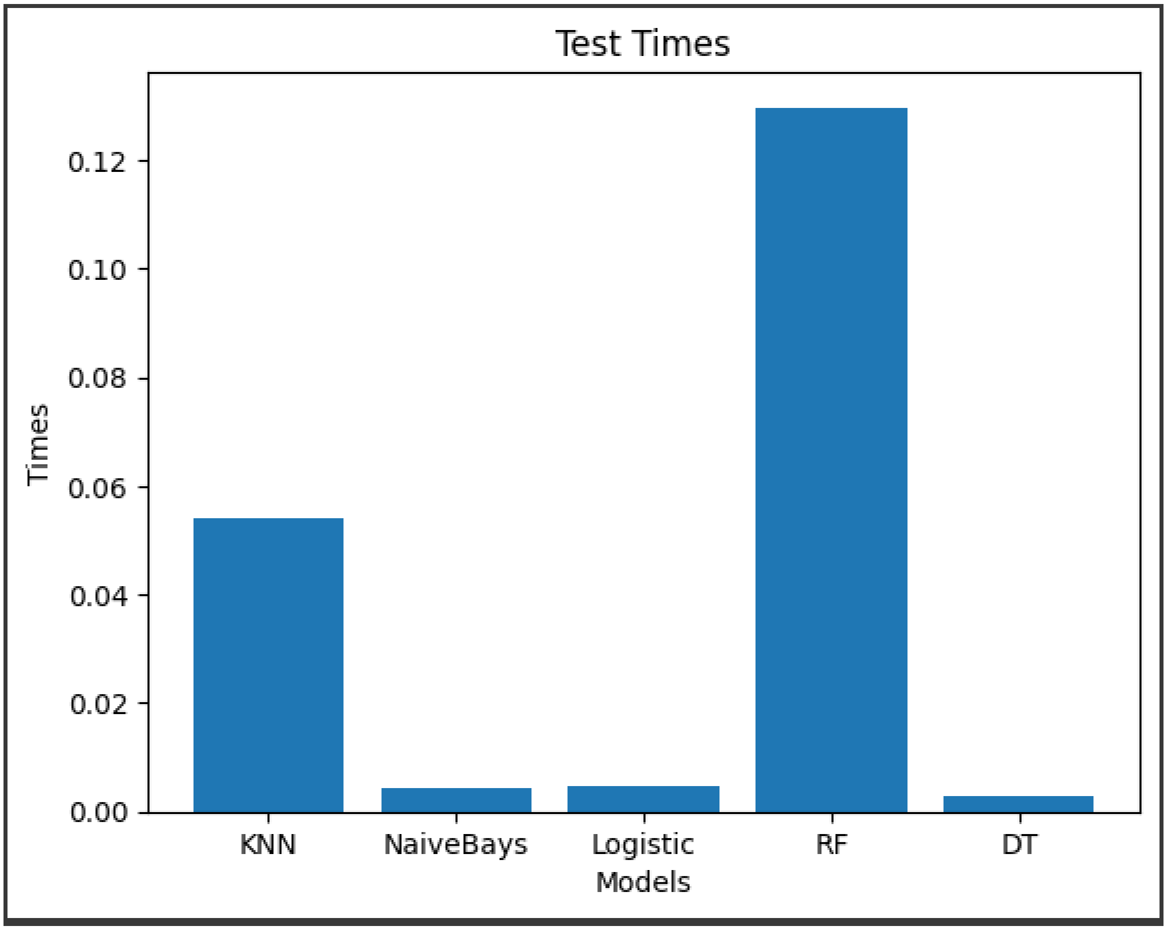
- **Accuracy Score:** `85.39 %`
- **Total Training Time:** `3.539412260055542`
- **Total Test Time:** `0.1166532039642334`

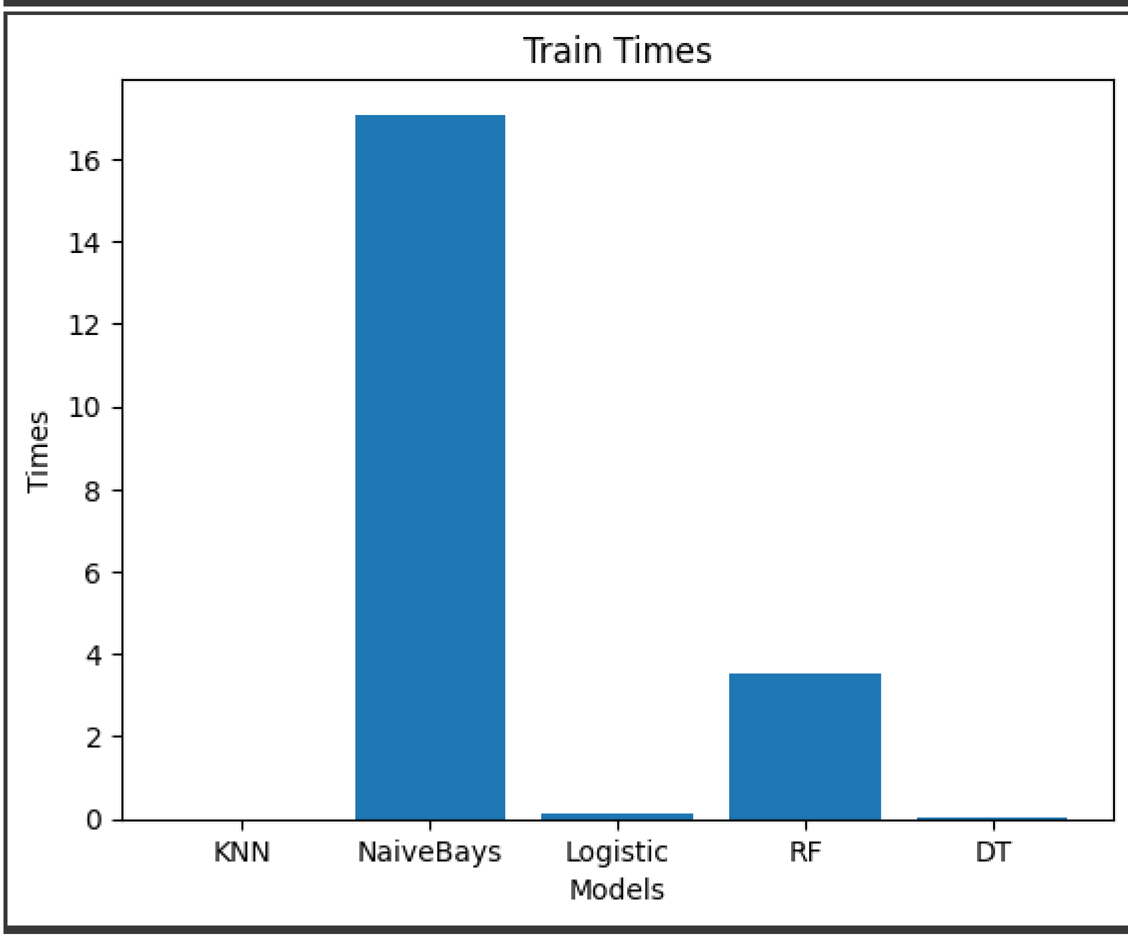
- `DecisionTreeClassifier`

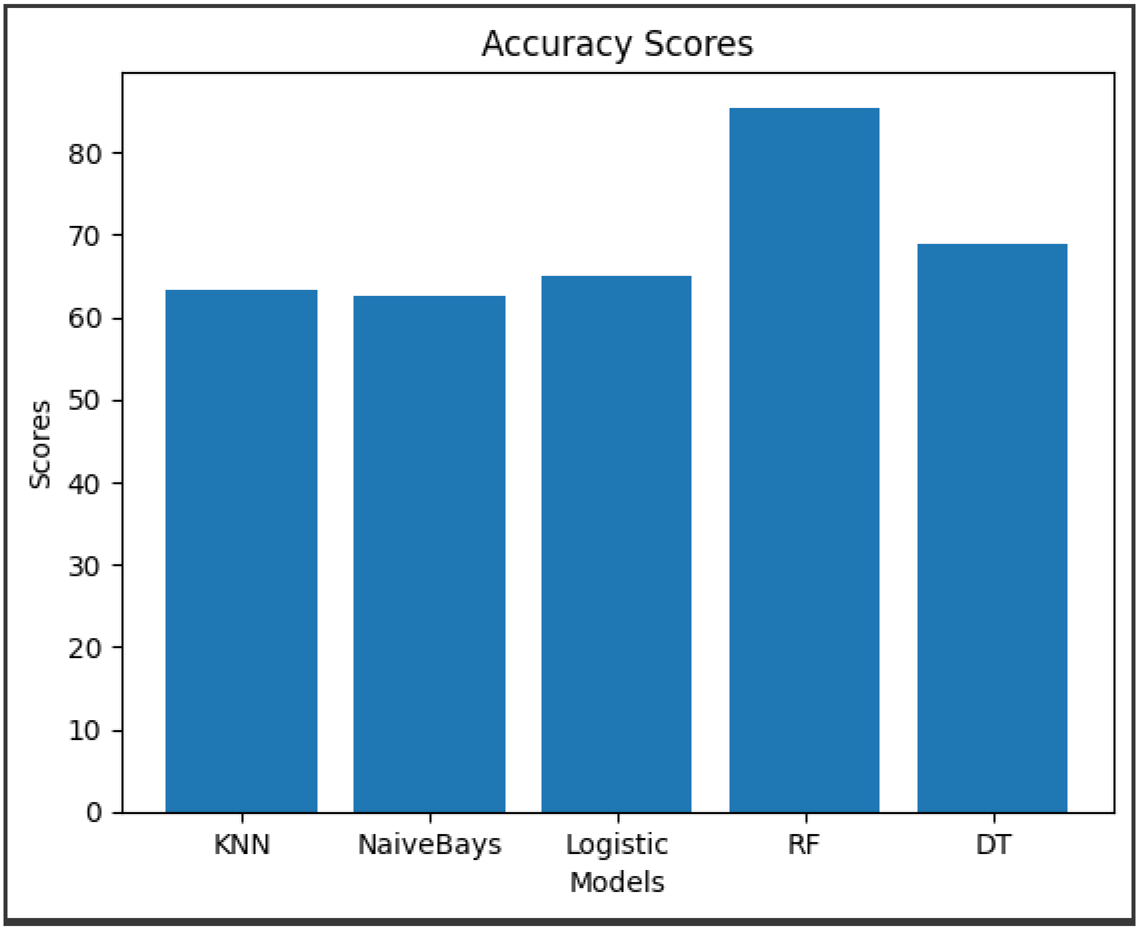
- **Accuracy Score:** `69.68698517298188 %`
- **Total Training Time:** `0.03876757621765137`
- **Total Test Time:** `0.0029397010803222656`

- `LogisticRegressionClassifier`

- **Accuracy Score:** `64.90939044481054 %`
- **Total Training Time:** `0.19150185585021973`
- **Total Test Time:** `0.010148048400878906`


Test Times
Times
0.12
0.10
0.08
0.06
0.04
0.02
0.00
KNN
NaiveBays
Logistic
RF
DT
Models


Train Times
Times
16
14
12
10
8
6
4
2
0
KNN
NaiveBays
Logistic
RF
DT
Models



Accuracy Scores
Scores
80
70
60
50
40
30
20
10
0
KNN
NaiveBays
Logistic
RF
DT
Models

# Feature Selection

- **In addition to feature selection done in ms1 Dropping “Status” Column**

All the values of this column are the same.

[NOT UNIQUE]

# HyperParameter Tuning

- **In DecisionTreeClassifier**

a. `max_leaf_nodes`

**Values:**

1. **[20]**
   *Accuracy Score:* `0.6803953871499177`
2. **[60]**
   *Accuracy Score:* `0.6935749588138386`
3. **[100]**
   *Accuracy Score:* `0.6507413509060955`

**Conclusion: Increasing the max_leaf_nodes param than 60 will lead to decreasing the accuracy score**

1. `max_depth`

**Values:**

4. **[4]**
   *Accuracy Score:* `0.6886326194398682`

5. **[10]**
   *Accuracy Score:* `0.6013179571663921`

6. **[20]**
   *Accuracy Score:* `0.5683690280065898`

**Conclusion: Increasing the max_depth param than 4 will lead to decreasing the accuracy score**

# Conclusion

In conclusion, the classification models applied on the movie regression dataset have revealed some interesting insights. It is evident that the inclusion of an increasing number of features has negatively impacted the performance of Naive Bayes and KNN models, resulting in lower accuracy. However, the Decision Tree model and Random Forest model have demonstrated their superiority in handling complex datasets, showcasing excellent accuracy rates. These models exhibit the ability to capture intricate relationships among features and make robust predictions, making them reliable choices for classification tasks in movie regression datasets. Their effectiveness can be attributed to their ability to handle high-dimensional data and capture non-linear interactions, leading to improved performance and accurate predictions.