Ain Shams University
Faculty of Computer & Information Sciences

جامعة عين شمس
كلية الحاسبات و المعلومات

Predicting Song Popularity

Final Report

Team ID: CS 13

Member name	Department	ID	Section
1. Basem Maher Ramadan	CS	2016170124	2A
2. Mohamed Samir Asaad	CS	2017170338	5A
3. Poula Atef Nashid	CS	2017170108	2A
4. Tawfek Hesham Tawfek	CS	2017170116	2A
5. Eman Ossama Mohamed	CS	2017170099	2A

Milestone 1

Regression Techniques

Preprocessing Techniques:

We use preprocessing techniques to transform the whole dataset with different data types to numerical data types only. The dataset contains columns with strings and date format which in turns should be encoded to a numerical value. So, we used both Label Encoding Technique and One Hot Encoding Technique.
Columns Label Encoded: artists, id, name & release date.
Column One Hot Encoded: key.

Steps for Label Encoding:

1- Import built in library for “LabelEncoder”.
2- Pass the parameters of the function which are data & specified columns.
3- Loop on the number of columns and for each column do the following steps.
4- Apply Label Encoder the use Fit-Transform function.

Steps for One Hot Encoding:

1- Import built in library for “One-Hot-Encoder”.
2- Pass the data to the function.
3- Convert data frame into numpy array.
4- Use Fit-Transform of one hot encoder class function.
5- Convert numpy array to data frame.

Results:

All dataset columns are mapped to numerical values. For None values we dropped all rows containing None value so the dataset had been reduced from 120,998 to around 118,632 rows.

Analysis:

The idea of feature analysis is to find out relations between features. So, we used correlation and plot it to decide which features we will work on in each model

Top 40% Correlation training features with the popularity.

Regression Techniques:

There are multiple regression techniques.
We used 5 different techniques which are:

1- Multiple Linear Regression:

a. Features: all features except “key” attribute of index 10.
b. Apply preprocessing to encode data.
c. Dataset size: 32% for testing and the rest 68% for training.
d. Validation size: 20% for testing and the rest 80% for training.
e. Mean Square Error for Validation: 119.47249331173556.
f. Mean Square Error for Test: 119.54597178916691.
g. Validation accuracy: 74.3%.
h. Test accuracy: 74.4%.
i. Time: 1.6 Sec.

2- **Polynomial Regression:**
   a. Features: all features except "key" attribute of index 10.
   b. Apply preprocessing to encode data.
   c. Dataset size: 32% for testing and the rest 68% for training.
   d. Validation size: 20% for testing and the rest 80% for training.
   e. Mean Square Error for Validation: 111.95045020284348.
   f. Mean Square Error for Test: 113.00241845521165.
   g. Validation accuracy: 75.9%.
   h. Test accuracy: 75.8%.
   i. Time: 1.8 Sec.

3- **Elastic Net Regression:**
   a. Features: all features except "key" of index 10.
   b. Apply preprocessing to encode data.
   c. Dataset size: 32% for testing and the rest 68% for training.
   d. Validation size: 20% for testing and the rest 80% for training.
   e. Hyper-Parameters: Alpha=0.00001, L1_ratio=1.
   f. Mean Square Error for Validation: 119.47244939076442.
   g. Mean Square Error for Test: 119.54178001316187.
   h. Validation accuracy: 74.3%.
   i. Test accuracy: 74.4%.
   j. Time: 1.5 Sec.

4- **Partial Least Squares Regression "PLS":**
   a. Features: all features except "key" of index 10.
   b. Apply preprocessing to encode data.
   c. Dataset size: 32% for testing and the rest 68% for training.
   d. Validation size: 20% for testing and the rest 80% for training.
   e. Hyper-Parameters: Number of components=6.
   f. Mean Square Error for Validation: 119.52643985712865.
   g. Mean Square Error for Test: 119.54812984038136.
   h. Validation accuracy: 74.2%.
   i. Test accuracy: 74.4%.
   j. Time: 1.1 Sec.

5- **Principal Components Regression "PCR":**
   a. Features: all features except "key" of index 10.
   b. Apply preprocessing to encode data.
   c. Dataset size: 32% for testing and the rest 68% for training.

d. Validation size: 20% for testing and the rest 80% for training.
e. Hyper-Parameters: Number of components=17.
f. Mean Square Error for Validation: 119.47243299769475.
g. Mean Square Error for Test: 119.54177590972576.
h. Validation accuracy: 74.3%.
i. Test accuracy: 74.4%.
j. Time: 1.1 Sec.

Multiple Linear Regression:

Index	*Feature Name*	*MSE*
0	Valence	450.2030756856612
1	Year	125.95715949328357
2	Acousticness	297.30067383136236
3	Artists	450.4506665068183
4	Danceability	429.83275436157226
5	Duration in ms	449.1745566457978
6	Energy	351.03442475536974
7	Explicit	413.0822673293138
8	ID	447.4860750172257
9	Instrumentalness	403.70159817818256
10	Key	450.3655981036182
11	Liveness	449.2292866719719
12	Loudness	364.17957583942416
13	Mode	449.8357253105536
14	Name	450.3113232366554
15	Tempo	443.1727891382329
16	Release date	434.730567705974
17	Speechiness	450.7136762545264

Features according to MSE ordered ascendingly: 1, 2, 6, 12, 9, 7, 4, 16, 15, 8, 5, 11, 13, 0, 14, 10, 3.
We tested each feature alone then according to the MSE results we started to gather features together one by one as the MSE was each time decreases until, we reached the smallest MSE using all features except the “key” feature.

Polynomial Regression:

Index	*Feature Name*	*MSE*
0	Valence	113.6448715699877
1	Year	209.1553978367558
2	Acousticness	115.068428802210714
3	Artists	113.17575021017677
4	Danceability	114.0114979654246
5	Duration in ms	113.17657406195963
6	Energy	113.32974810057648
7	Explicit	113.87504059785473
8	ID	113.5362218993563
9	Instrumentalness	114.9503516035143
10	Key	113.00241845521165
11	Liveness	113.16714073025796
12	Loudness	113.61382169262168
13	Mode	113.50049577234051
14	Name	113.0580363352783
15	Tempo	113.15938828224768
16	Release date	113.7872410922448
17	Speechiness	114.01027829165571

This table represents the MSE for all features used except one column. For example, the first row represents the MSE for all features except “valence” of index 0. We find that while dropping column 10 “key” the MSE was at his smallest value so, we decided to use all features and drop column 10.

Conclusion:

We have different regression techniques each model differs according to the problem and data used. But, in general polynomial regression gives better accuracy and less MSE. but we should be careful of higher polynomial degree to avoid overfitting case. Dataset size also plays an important role to avoid underfitting case to make sure that the model had been trained on a sufficient amount of data.

Polynomial regression gives higher accuracy in case of choosing suitable degree.

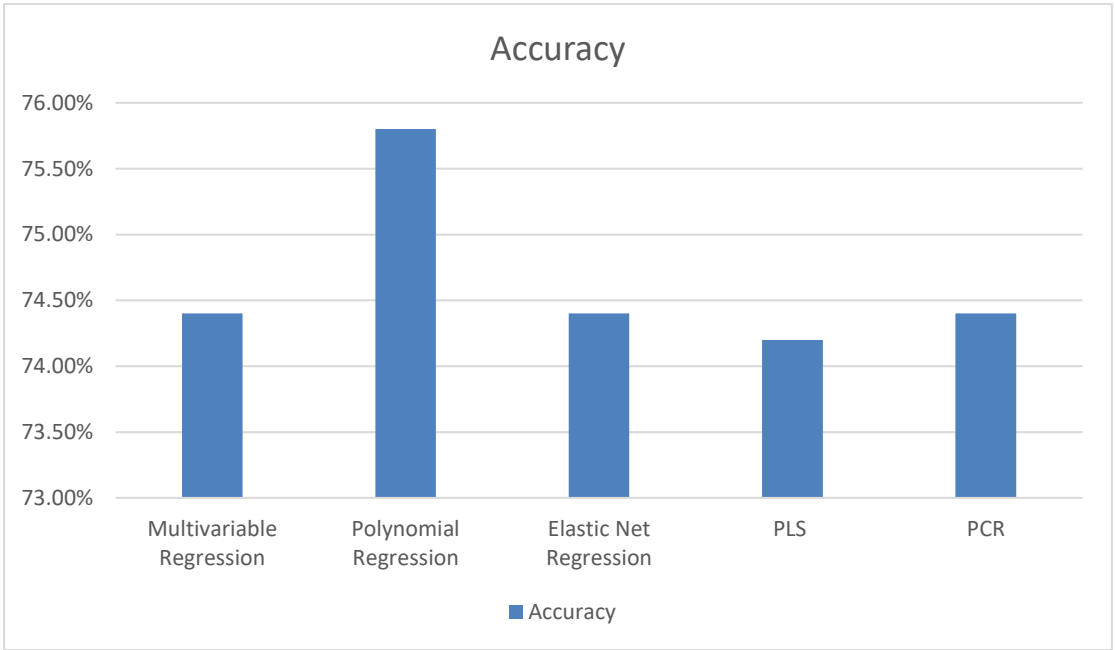
Multiple linear regression and Elastic Net regression both gives almost same MSE but both models are not the same. Elastic Net has two hyper parameters which affects the value of MSE. While, in Multiple regression is highly affected by the number of features used.

PLS as a performance is better than PCR both gives nearly the same accuracy but PLS uses a smaller number of components than PCR.

Outputs after saving the regression models:

```
main
C:\Users\pop\anaconda3\envs\ml\python.exe "D:/FCIS_ASU     2021/4 Year 2020_2021/Semester 1 CS/Machine Learning/Project/Milestone 1/Code/main.py"
Mean Square Error of Multivariable Regression : 122.80194717959183
R2Score of Multivariable Regression : 0.7368090573891282
Accuracy Score of Multivariable Regression : 73.68091%
Testing Time : 0.03303122520446777 Sec

Mean Square Error of Polynomial Regression : 116.19195831173197
R2Score of Polynomial Regression : 0.7509756829250832
Accuracy Score of Polynomial Regression : 75.09757%
Testing Time : 0.05773591995239258 Sec

Mean Square Error of Elastic Net Regression : 122.80194677114271
R2Score of Elastic Net Regression : 0.7368090582645224
Accuracy Score of Elastic Net Regression : 73.68091%
Testing Time : 0.023936748504638672 Sec

Mean Square Error of PLS Regression : 122.8614884112694
R2Score of PLS Regression : 0.7366814477440913
Accuracy Score of PLS Regression : 73.66814%
Testing Time : 0.03631997108459473 Sec

Mean Square Error of PCR Regression : 122.80194717959102
R2Score of PCR Regression : 0.7368090573891299
Accuracy Score of PCR Regression : 73.68091%
Testing Time : 0.08146381378173828 Sec
```

Accuracy graph for regression models:



Ain Shams University
Faculty of Computer & Information Sciences

جامعة عين شمس
كلية الحاسبات و المعلومات

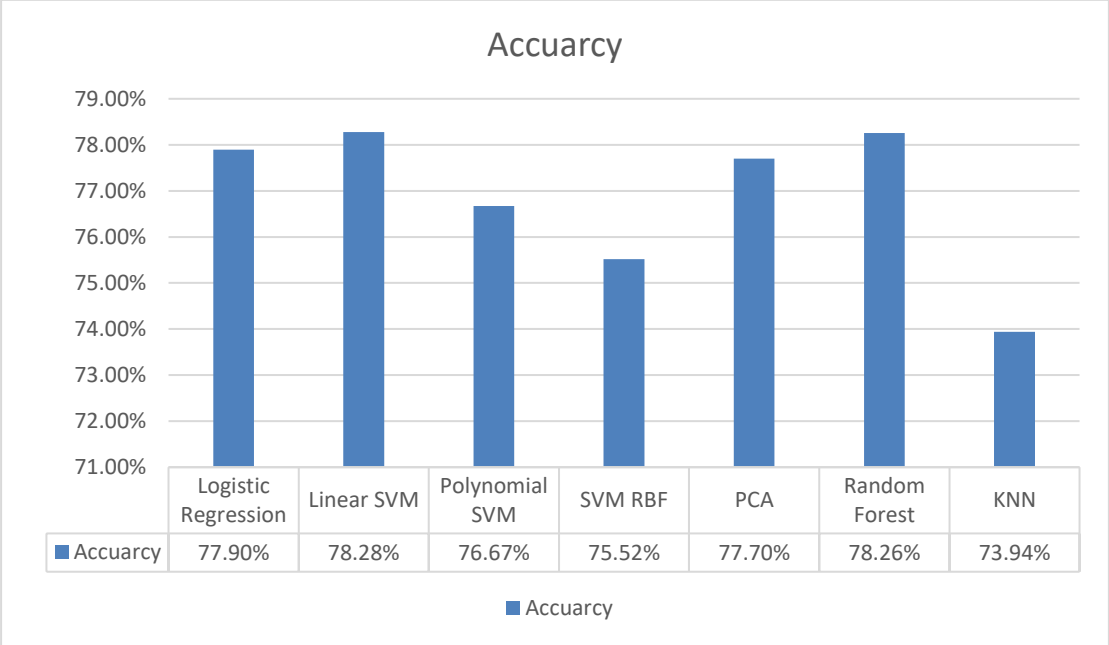
Milestone 2

Classification Techniques

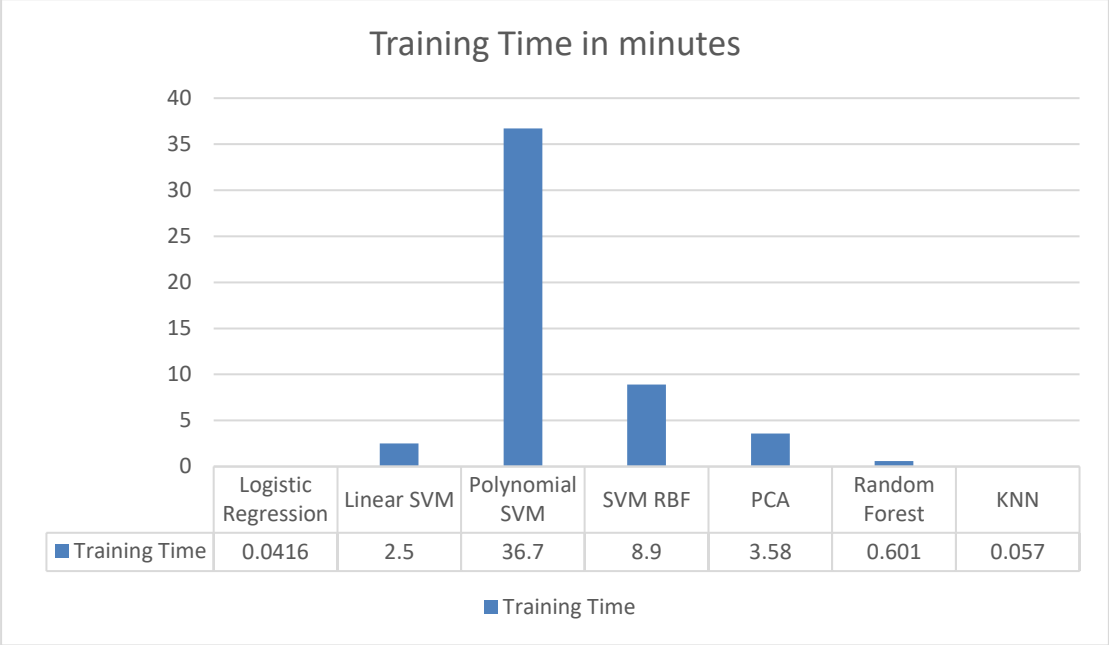
1- Logistic Regression.
2- Linear SVM.
3- Polynomial SVM.
4- Kernel SVM “RBF”.
5- Principal Component Analysis “PCA”.

Summary:

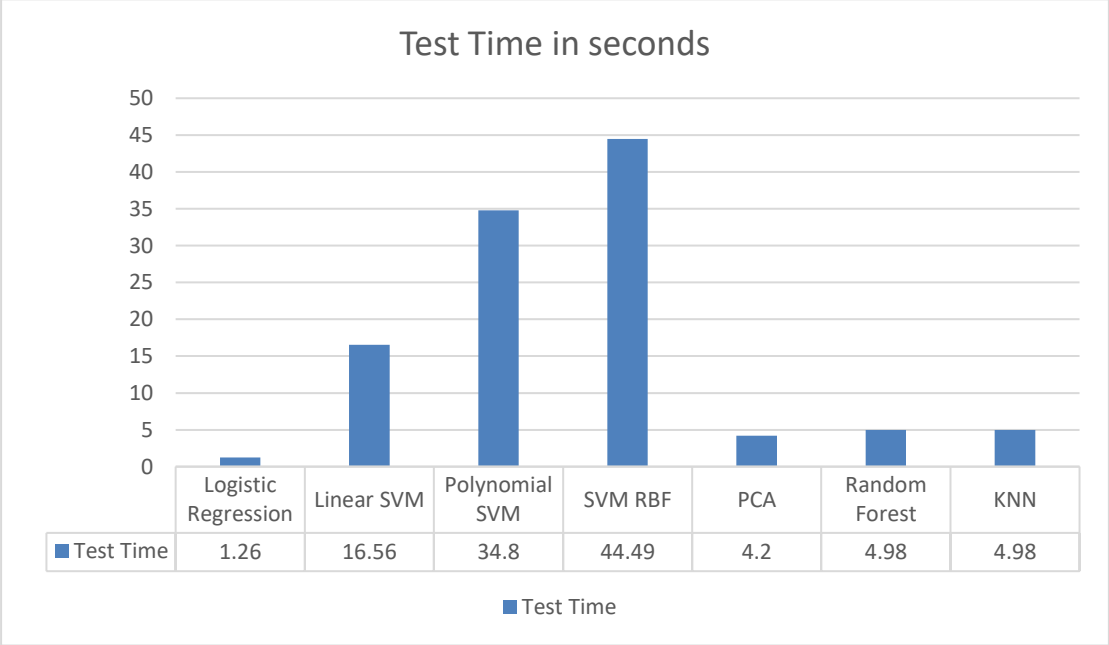
Accuracy graph:



Total Training Time Graph:



Total Testing Time Graph:



Feature Selection:

1- **Logistics Regression:**

Columns dropped: acouticness, artists, id, duration_ms, explicit, instrumentalness, liveness, mode, release_date.

Feature dropped	*Accuracy*
Valence	77.73187%
Year	68.84114%
Acousticness	77.68326%
Artists	77.70756%
Danceability	77.71729%
Duration_ms	77.73673%
Energy	77.79506%
Explicit	77.72215%
ID	77.71729%
Instrumentalness	77.77076%
Key	77.72215%
Liveness	77.78048%
Loudness	77.62007%
Mode	77.62493%
Name	77.71729%
Tempo	77.75617%
Release_date	77.78534%
Speechiness	77.69784%

Here, we trained the model every time with whole dataset except one feature is dropped and recognized the accuracy change to decide which features are going to be dropped or not. We decided to drop features that gave high accuracy so we arranged the features dropped descending order and started to accumulate until there was no huge difference in the MSE & accuracy.

2- **Linear Kernel:**

Columns dropped: explicit, acouticness, artists, instrumentalness, release_date, liveness, key.

Dataset used with only one feature:

Feature dropped	*Accuracy*
Without drop	77.85825%
Valence	77.89228%
Year	-30.41102%
Acousticness	6.55307%
Artists	6.08021%
Danceability	5.80313%
Duration_ms	5.86543%
Energy	6.48901%
Explicit	6.00149%
ID	5.79511%
Instrumentalness	5.81665%
Key	6.34328%
Liveness	6.09290%
Loudness	6.15364%
Mode	5.91348%
Name	5.76012%
Tempo	6.00574%
Release_date	6.21183%
Speechiness	5.80365%

3- **Polynomial Kernel:**

Columns dropped: mode, explicit, valence, acousticness, duration_ms, artists, speechiness, instrumentalness, key.

4- **Kernel RBF:**

Columns dropped: mode, explicit, valence, acouticness, duration_ms, artists, speechiness, instrumentalness, key.

5- **PCA:**

Columns dropped: No columns are dropped. It works with the number of components features are ordered according to the importance in the dataset.

6- **Random Forest:**
Columns dropped: acouticness, artists, id, duration_ms, explicit, instrumentalness, release_date, mode, liveness.

7- **KNN:**
Columns dropped: acouticness, artists, id, duration_ms, explicit, instrumentalness, release_date, mode, liveness.

Hyperparameter Tuning:

Models with hyperparameters:

1- **Polynomial SVM:**
Degree 1: same as linear SVM
Degree 2: the MSE decreases.
Degree 3: the MSE decreases.
Degree 4: the MSE decreases.
Degree 5: the MSE reached the maximum decay.
Degree 6: the MSE increased "Overfitting case".

2- **Kernel SVM Gaussian "RBF":**
Gamma = 0.1, C = 0.01, Accuracy = 78.15556%.
Gamma = 0.1, C = 0.10, Accuracy = 78.26475%.
Gamma = 0.1, C = 1.00, Accuracy = 78.35906%.
Gamma = 0.1, C = 2.00, Accuracy = 78.29454%. "test error is too small" so we neglect these results.
Gamma = 0.001, C = 0.01, Accuracy = 75.52%.

3- **Principal Component Analysis "PCA":**

1- Linear SVC:
Components = 05, Accuracy = 68.5%.
Components = 07, Accuracy = 70.9%.
Components = 10, Accuracy = 71.3%.

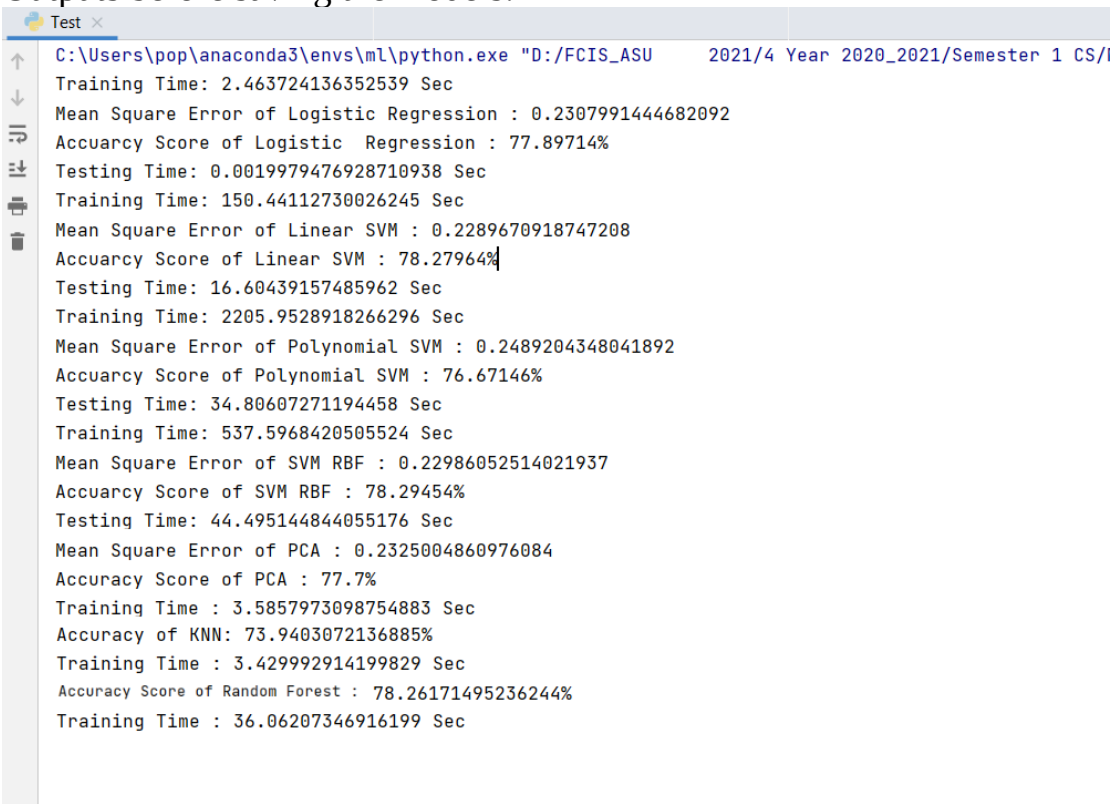
2- Polynomial Kernel:
Degree = 3, # Components = 16, Accuracy = 75.2%.
Degree = 4, # Components = 16, Accuracy = 75.0%
Degree = 5, # Components = 17, Accuracy = 74.8%.
Degree = 5, # Components = 15, Accuracy = 74.7%.
Degree = 5, # Components = 12, Accuracy = 73.9%
Degree = 6, # Components = 16, Accuracy = 74.4%.
Degree = 7, # Components = 15, Accuracy = 73.9%.

3- Logistic Regression:
Components = 24, Accuracy = 75.9%.
Components = 28, Accuracy = 77.7%.

Conclusion:

We have different classifiers each model differs according to the problem and data used. But, in general Random Forest gives better accuracy and less MSE. It took less than 1 minute to be trained which is acceptable comparing with the time taken by polynomial kernel.

Outputs before saving the models:



Outputs after saving the models:

```
C:\Users\pop\anaconda3\envs\ml\python.exe "D:/FCIS_ASU     2021/4 Year 2020_2021/Semester 1 CS/I
Mean Square Error of Logistic Regression : 0.23264857717846762
Accuracy Score of Logistic Regression : 77.77637%
Testing Time: 0.020127534866333008 Sec

Mean Square Error of Linear SVM : 0.2313749890626975
Accuracy Score of Linear SVM : 78.01165%
Testing Time: 90.95659828186035 Sec

Mean Square Error of Polynomial SVM : 0.24043593657336743
Accuracy Score of Polynomial SVM : 77.27472%
Testing Time: 150.14755773544312 Sec

Mean Square Error of SVM RBF : 0.25303570907747497
Accuracy Score of SVM RBF : 75.54225%
Testing Time: 357.0294461250305 Sec

Accuracy Score of Random Forest : 95.02523%
Testing Time : 4.991863012313843 Sec

Accuracy Score of KNN : 95.02523%
Testing Time : 5.0070481300354 Sec
```