Milestone1 Report

1. Preprocessing techniques: -
2. Train Test Split: convert original data set into 2 parts train (60%) and test (40%).
3. Remove all null values: - Df.dropna().

### Taking care of Categorical Features: - take care of categorical features by converting them to integers. There are 2 common ways to do so

### Label encoding: - used for converting artists name to integer values

* One Hot Encoding: - used in key category

1. Perform analysis on the dataset

We have dropped some features which is not important and not affecting on popularity(artists, name, release\_data)

1. Regression techniques: -
2. Linear regression : is an attractive model because the representation is so simple, The representation is a linear equation that combines a specific set of input values (x) the solution to which is the predicted output for that set of input values (y). As such, both the input values (x) and the output value are numeric, the linear equation assigns one scale factor to each input value or column, called a coefficient and represented by the capital Greek letter Beta (B). One additional coefficient is also added, giving the line an additional degree of freedom (e.g. moving up and down on a two-dimensional plot) and is often called the intercept or the bias coefficient, For example, in a simple regression problem (a single x and a single y), the form of the model would be:

y = B0 + B1\*x

1. Polynomial regression: - Polynomial Regression is a regression algorithm that models the relationship between a dependent(y) and independent variable(x) as nth degree polynomial. The Polynomial Regression equation is given below:

y= b0+b1x1+ b2x12+ b2x13+...... bnx1n

It is also called the special case of Multiple Linear Regression in ML. Because we add some polynomial terms to the Multiple Linear regression equation to convert it into Polynomial Regression.

1. Difference between two models:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Accuracy | Error | Training time |
| Linear regression | 0.7319121907395754 | 125.77831644934817 | 0.5s |
| Polynomial regression | 0.7555346028023568 | 10.70411462117986 | 0.2s |

1. screenshots of the resultant(s) regression line plots:-

A picture containing graphical user interface

Description automatically generated

1. Size of training set is 60% and test set 40%
2. Techniques used to improve the result: - SelectKBest to get most important features.
3. Conclusion

* Reading dataset
* To apply preprocessing first drop all nulls apply label and one hot encoding to convert all categorical features into integers
* Divide dataset into two part one for training set (60%) and one for test set (40%).
* Apply linear or polynomial regression to get accuracy and error in this steps :
* Fit all x\_trains and y\_trains
* Get prediction from all x\_tests
* Calculate score and mean square error