Phase 6: Model Optimization

Team Report by Team OLAPPED CSCI 6401-01

Team Name :- OLAPPED

Course :- Data Mining

Instructor :- Prof. Shivanjali Khare

Course ID :- CSCI-6401-01

Session :- Fall 2023

Assignment :- Phase 6: Model Optimization

1. Team Name :- OLAPPED

Team Member Names :-

(1) Sean Vargas – svarg1@unh.newhaven.edu

(2) Kaylie N Neal – kneal5@unh.newhaven.edu

(3) Prashant Rana – prana4@unh.newhaven.edu

(4) Rajdeep Bhattacharya – rbhat6@unh.newhaven.edu

2. Selected Dataset :-

Description of the selected dataset that we want to work with –

The Data Set is named "trends" as it represents the Google Search Trends for a period of 20 years (2001 - 2020), pointing out the **[Top 5 Google Searches (Search Queries)]** by **[Categories]** with their **[Global Ranks]** and the ranks for the **[Top Countries]** by **[Year].**

It has 5 Attributes (represented by 5 columns) namely **location, year, category, rank and query**. And 26956 Data Points (represented by 26956 rows).

Research Question :-

The Research Question: - How can the patterns and trends in the **Google Search Engine Queries Dataset** be used to identify better rules for finding the most promising **Keywords (Search Queries)** and **Topics (Categories)** for showing more relevant **Search Results** in the future and targeting the relevant **SERPs (Search Engine Result Pages)** for **Ad Suggestions** by **timings (Seasonality)** and **locations (Geography)** for **Customer/Viewer Satisfaction** and higher **Ad Revenues**.

3. List of data mining techniques used

For this phase we have used 2 Data Modelling techniques, Linear Regression and Isotonic regression. We will be using more Data models in our optimization phase.

I. **Linear regression**

Linear regression is a data modelling technique that allows us to model data based on linear data attributes. In linear regression, the data are plotted using scatter plots and a linear line is drawn which best fits the data on the scatter plots. For our linear regression model, we try to form our linear model with expression y = b1x + b0, where b1 is the coefficient of x I.e. slope of our line and b0 is the y-intercept. We calculate the coefficients from our given data x and y which is known to us from our dataset I.e. x as year and y as Frequency of each category (Calculated by grouping by category).

Parameters: y-intercept (b0) is the slope intercept which forms the equation of the line, slope (b1) is the measure of the tangent of the angle made by the line with x-axis.

Hyperparameters: Lasso and Ridge regularization to prevent overfitting, optimization algorithms such as gradient descent or ordinary least squares, learning rate can be set if gradient descent is used.

II. **Isotonic regression**

Isotonic Regression is a type of regression analysis that focuses on modeling the relationship between a single independent variable and a dependent variable while maintaining a monotonic (non-decreasing or non-increasing) relationship between them. We have used isotonic regression from sci-kit learn library of python. For evaluation metric, we used mean squared error from scikit.metrics. Here, also we are using scatterplot to plot our data with x as year and y as Frequency of each category over period of time.

Parameters: isotonic segmentations represent the region where the function is constrained to be monotonic.

Hyperparameters: increasing or decreasing, we can specify whether we want increasing or decreasing monotonic regression; Y-min and Y-max, minimum and maximum values the fitted function should not go beyond; out of bound strategy, specifies how to handles predictions that go beyond specified constraints.

4. Hardware used:

Hardware specification being used for data mining project:

- CPU: Intel Core i7
- RAM: 16 GB
- Storage: 512 GB HDD

Software specifications being used:

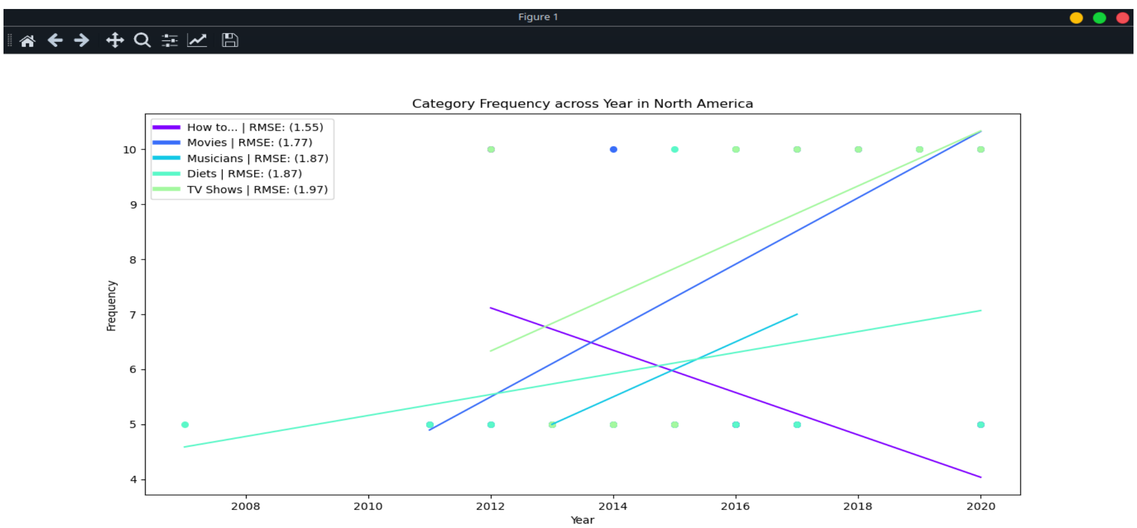
- IDE: Visual Studio Code
- Programming Language: Python 3.8
- Libraries used: Pandas, Numpy, Matplotlib, Sci-kit learn

5. Outcomes:

For our modelling, we plotted Linear and Isotonic regression for top 5 unique categories. For measuring our models, we used Root Mean Squared Error. While rmse between 0.2 and 0.9 is considered a good model. But due to deficiency of good datas in our dataset we got rmse along 1.8 to 2.9. Based on these values, we considered the lowest rmse value a good candidate for our models. Hence, we selected top 5 categories based on those metrics and plotted our top 5 categories over period of time. Along the line, we divided our dataset along different continents as most unique patterns can only be found in certain locations.

We performed different perspectives of visualization by changing certain metrics, like the minimum observation points required for plotting the model. For Linear regression, minimum observation points required was set to 4 while for Isotonic regression, it was set to 7.

Linear Regression:



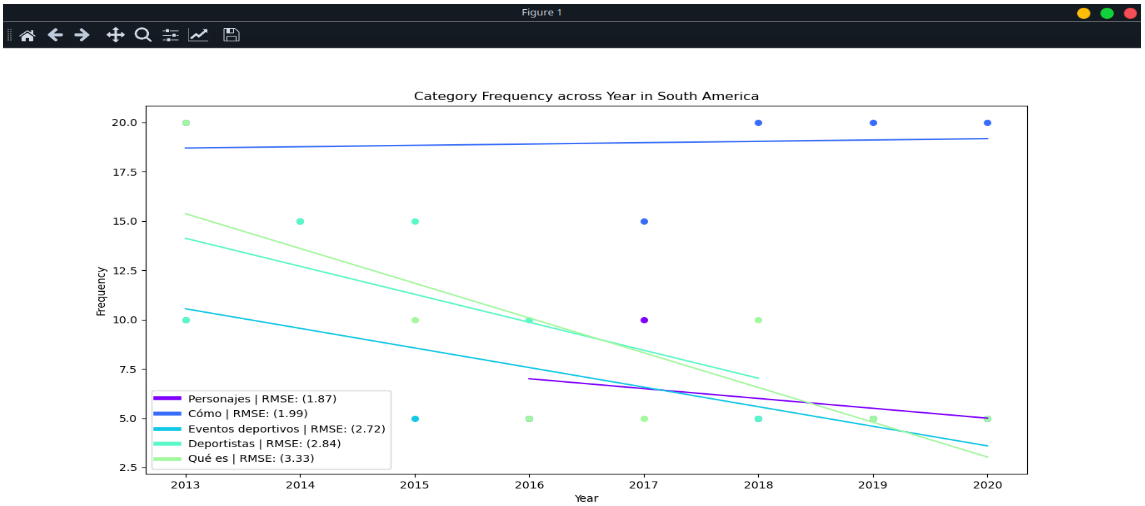


Figure 1
Category Frequency across Year in South America
Frequency
Year
Personajes | RMSE: (1.87)
Cómo | RMSE: (1.99)
Eventos deportivos | RMSE: (2.72)
Deportistas | RMSE: (2.84)
Qué es | RMSE: (3.33)

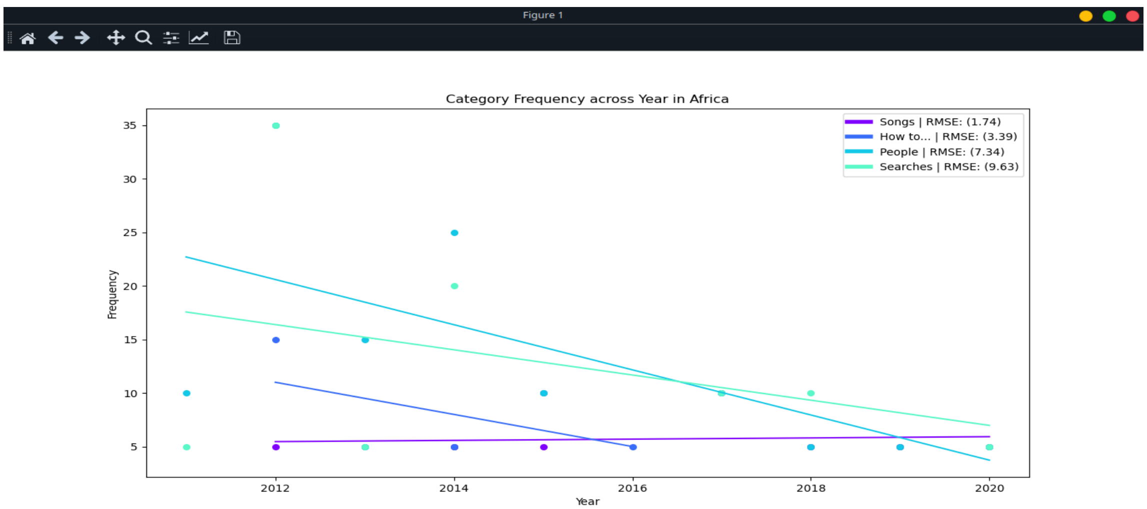


Figure 1
Category Frequency across Year in Africa
Frequency
Year
Songs | RMSE: (1.74)
How to... | RMSE: (3.39)
People | RMSE: (7.34)
Searches | RMSE: (9.63)

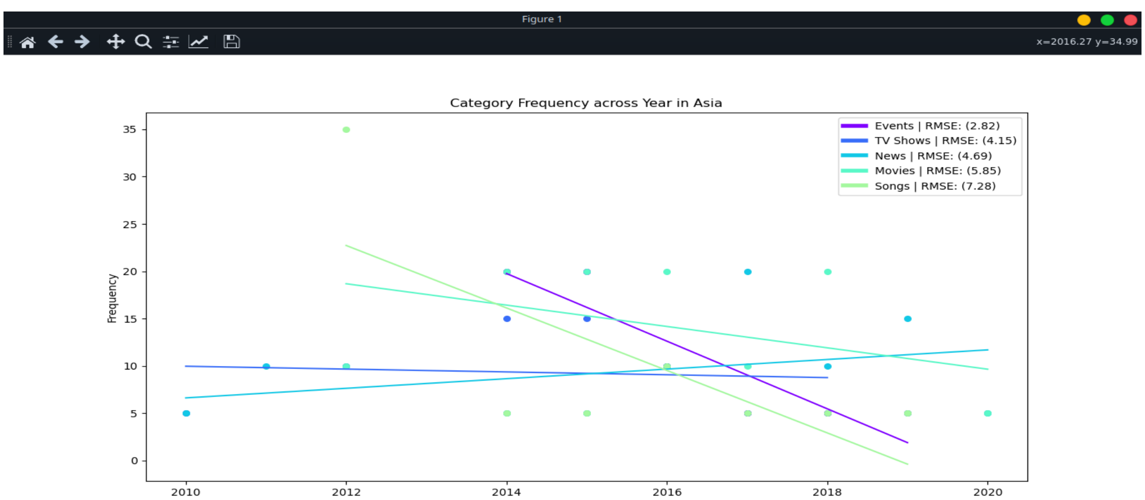
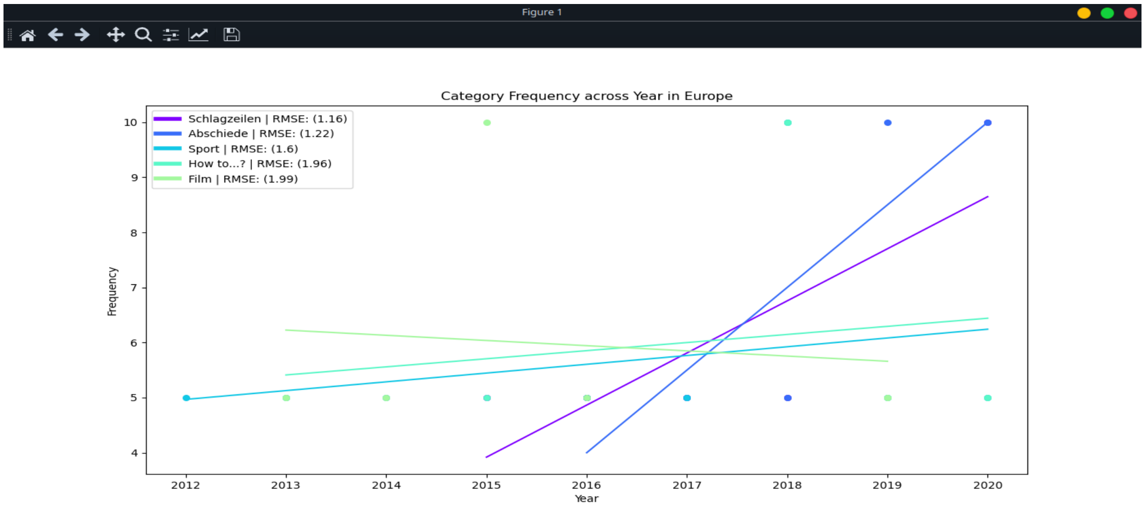
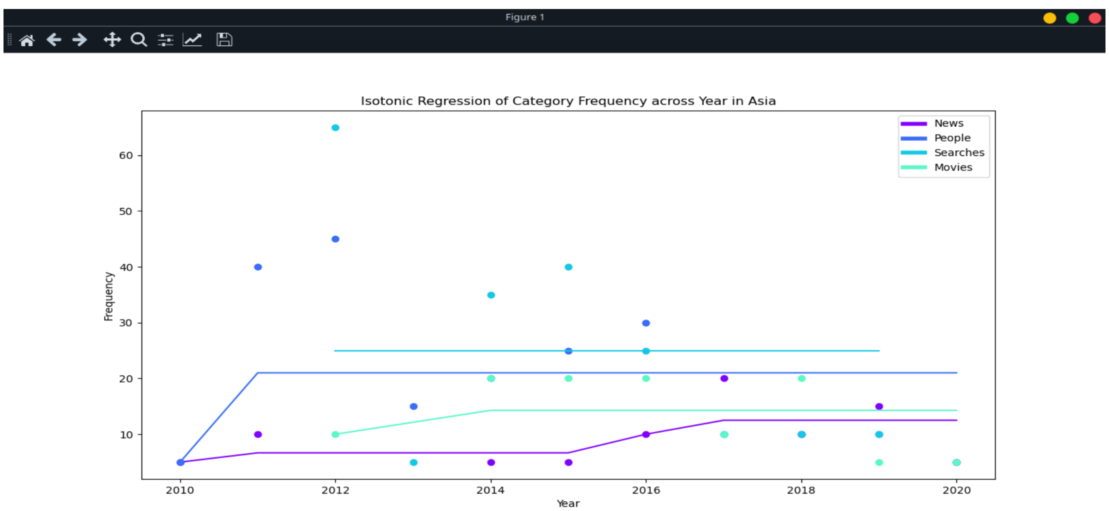


Figure 1
x=2016.27 y=34.99
Category Frequency across Year in Asia
Frequency
Year
Events | RMSE: (2.82)
TV Shows | RMSE: (4.15)
News | RMSE: (4.69)
Movies | RMSE: (5.85)
Songs | RMSE: (7.28)



Isotonic Regression:



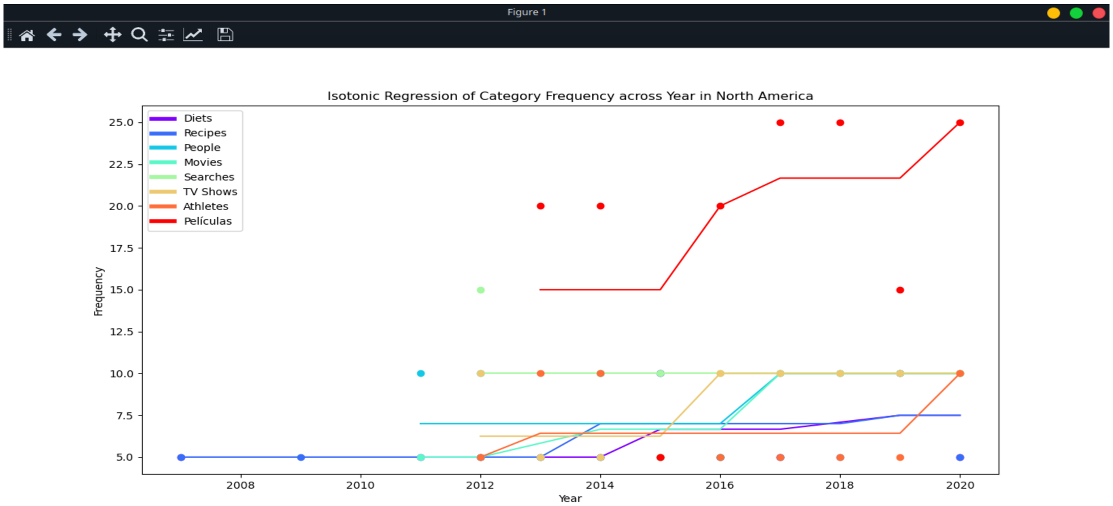


Figure 1
Isotonic Regression of Category Frequency across Year in North America
Diets
Recipes
People
Movies
Searches
TV Shows
Athletes
Películas
Frequency
Year
25.0
22.5
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2008
2010
2012
2014
2016
2018
2020

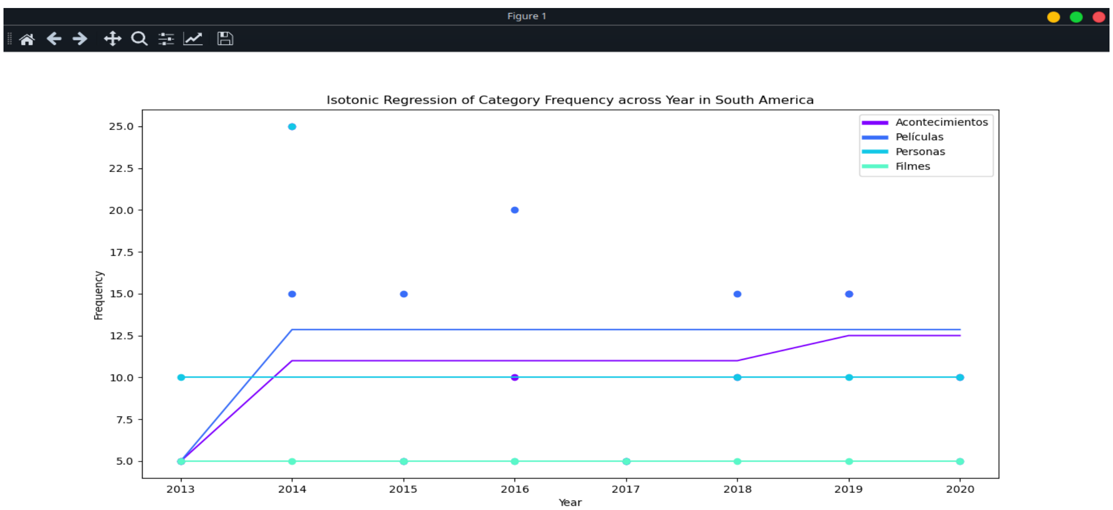


Figure 1
Isotonic Regression of Category Frequency across Year in South America
Acontecimientos
Películas
Personas
Filmes
Frequency
Year
25.0
22.5
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2013
2014
2015
2016
2017
2018
2019
2020

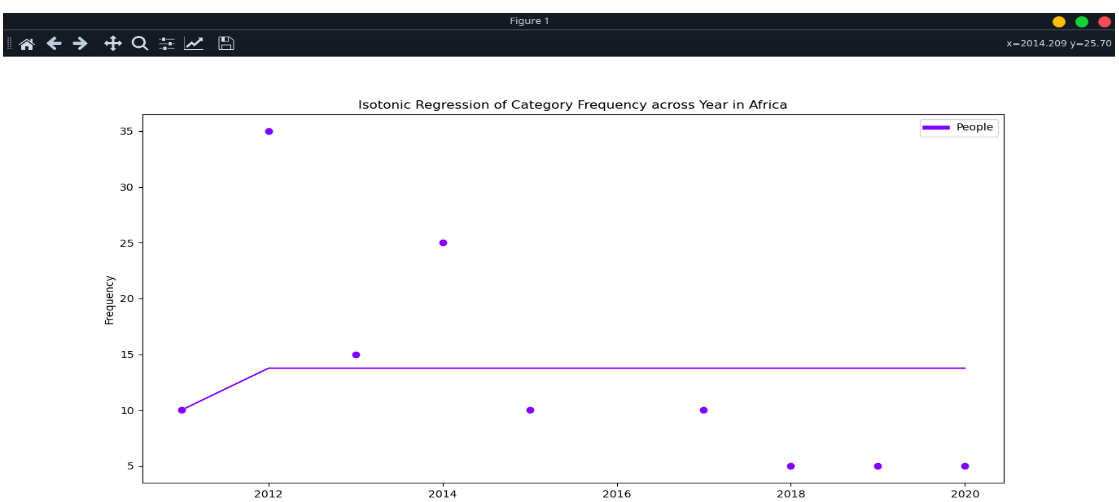
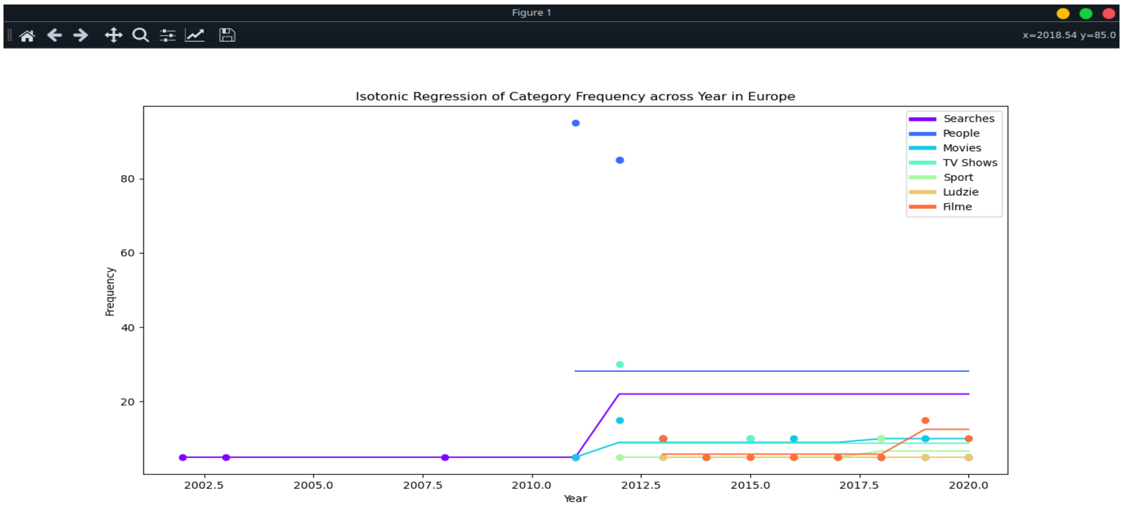


Figure 1
x=2014.209 y=25.70
Isotonic Regression of Category Frequency across Year in Africa
People
Frequency
Year
35
30
25
20
15
10
5
2012
2014
2016
2018
2020



Random Forest:

After analyzing the data with Linear Regression and Isotonic Regression, to dig deeper we tried our hands with Random Forest Algorithm based on the concept of bagging. The objective of this project was to find a relation between the greatest number of search categories and the year and continent they are searched on.

Random Forest is a widely-used machine learning algorithm developed by Leo Breiman and Adele Cutler, which combines the output of multiple decision trees to reach a single result. Its ease of use and flexibility have fueled its adoption, as it handles both classification and regression problems.

One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables, as in the case of regression, and categorical variables, as in the case of classification. It performs better for classification and regression tasks.

Important Hyperparameters in Random Forest:

Hyperparameters are used in random forests to either enhance the performance and predictive power of models or to make the model faster.

Parameters And Hyperparameters

Hyperparameters to Increase the Predictive Power:

n_estimators: Number of trees the algorithm builds before averaging the predictions.

max_features: Maximum number of features random forest considers splitting a node.

mini_sample_leaf: Determines the minimum number of leaves required to split an internal node.

criterion: How to split the node in each tree? (Entropy/Gini impurity/Log Loss)

max_leaf_nodes: Maximum leaf nodes in each tree

Hyperparameters to Increase the Speed:

n_jobs: it tells the engine how many processors it is allowed to use. If the value is 1, it can use only one processor, but if the value is -1, there is no limit.

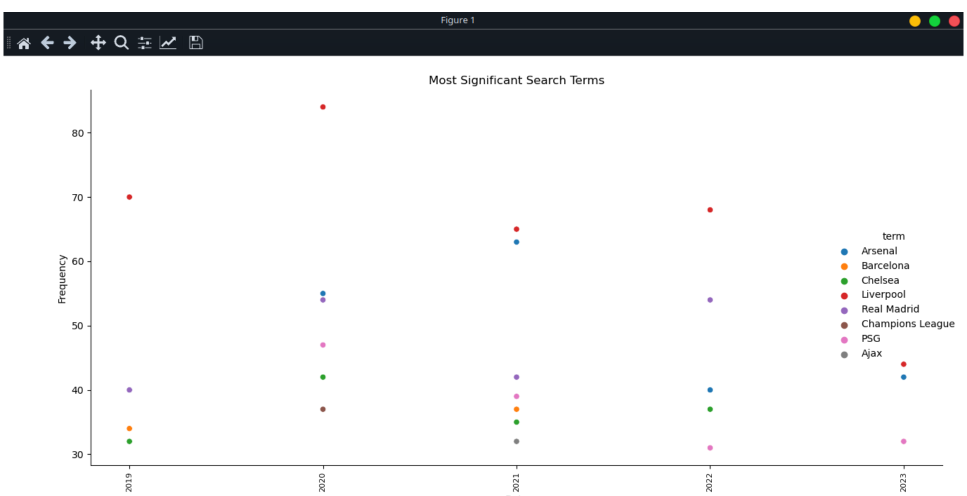
random_state: controls randomness of the sample. The model will always produce the same results if it has a definite value of random state and has been given the same hyperparameters and training data.

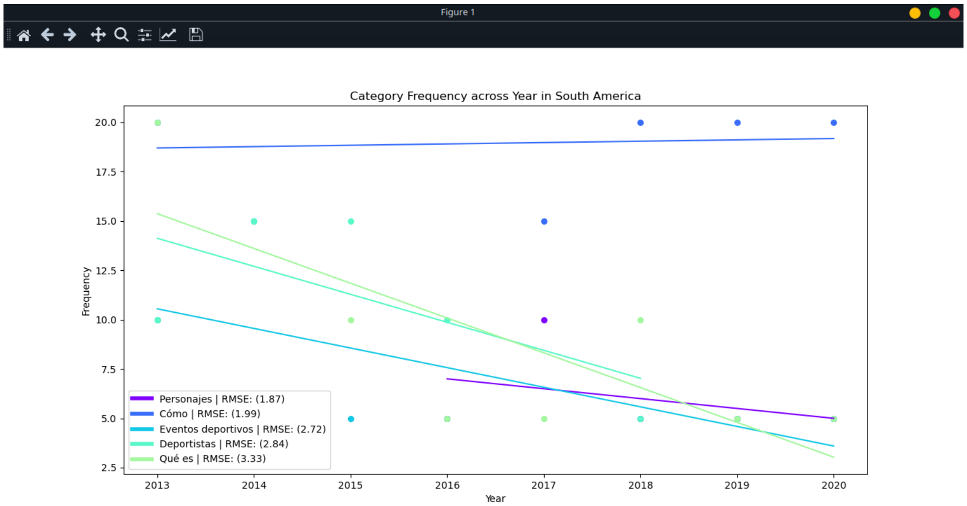
oob_score: OOB means out of the bag. It is a random forest cross-validation method. In this, one-third of the sample is not used to train the data; instead used to evaluate its performance. These samples are called out-of-bag samples.

6. Visualization Techniques used:

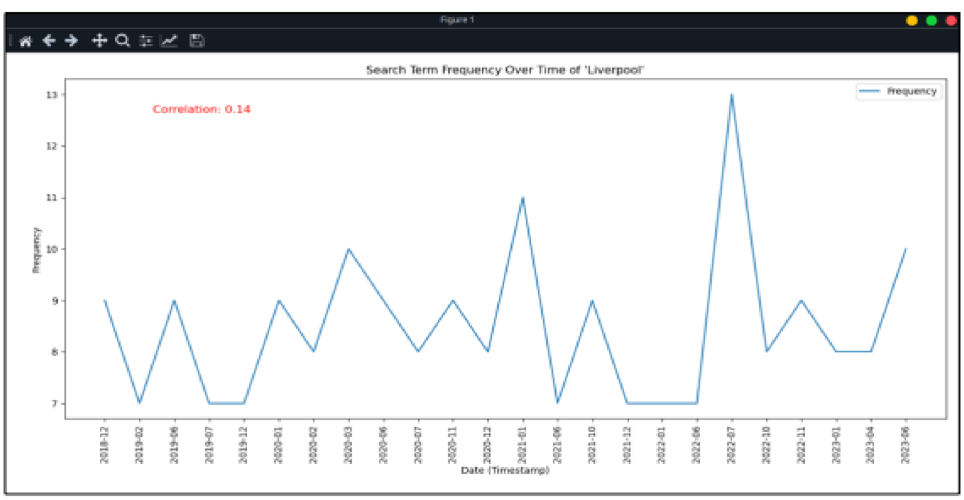
For our Data Mining project, we have used various visualization techniques which were used in our Data modelling and Data exploration phase and are listed as follows:

i. Scatter plot

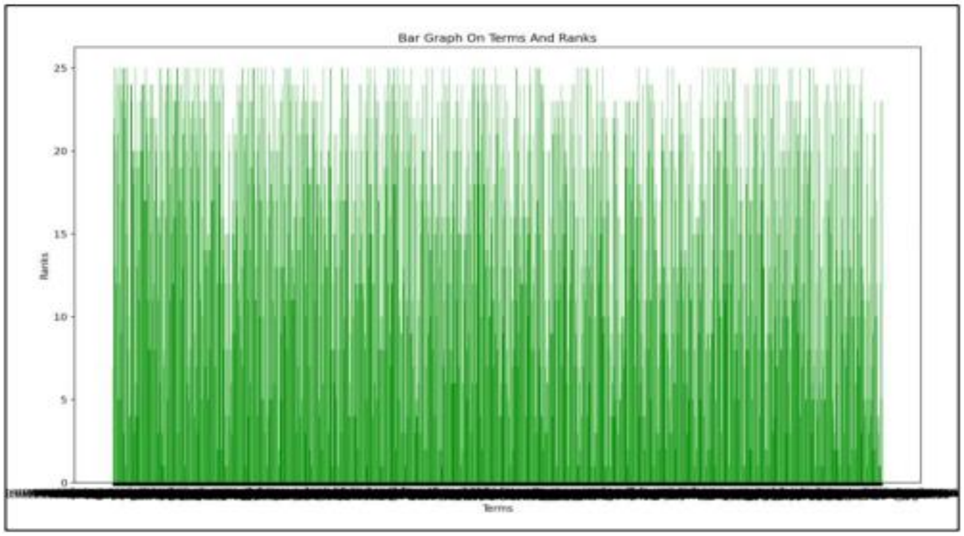


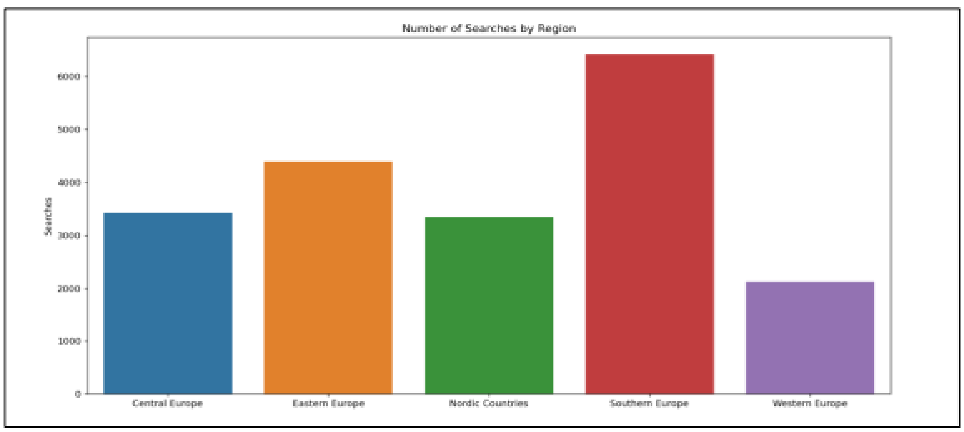


ii. Line graph



iii. Bar graph



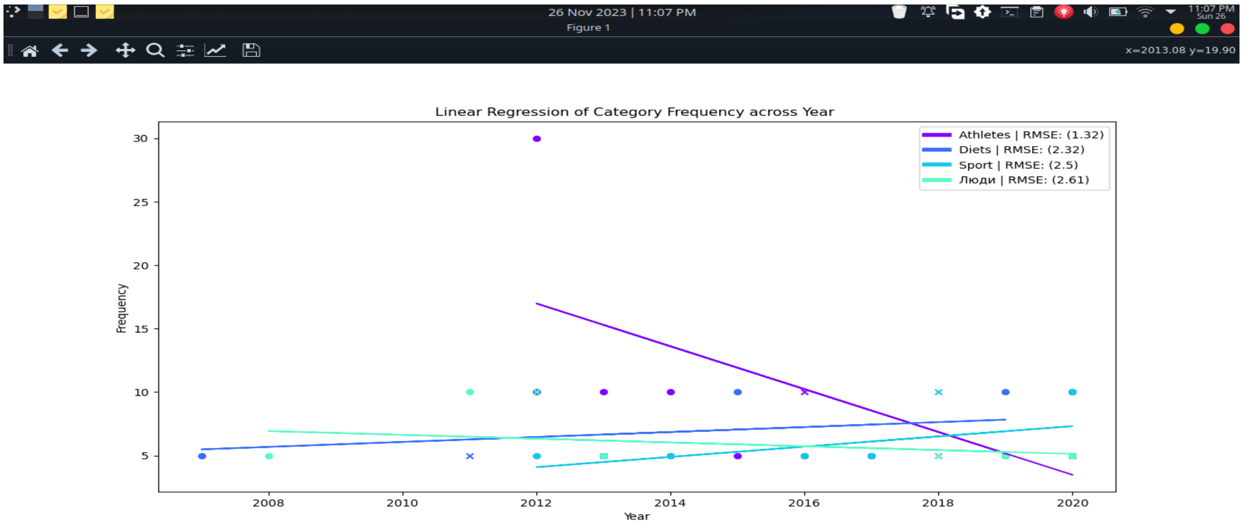


7. Optimization Conclusion:

I. Gradient Descent:

In this Gradient Descent implementation, the code iterates through unique categories in a DataFrame, performing linear regression using gradient descent to optimize the coefficients for each category. The process involves preprocessing data by filtering and grouping, followed by a loop for each category. Key components include setting hyperparameters such as the number of iterations and learning rate. Data is split into training and testing sets using scikit-learn's train_test_split, and accuracy is evaluated during the gradient descent process. Visualization includes plots of the best-fit line for each category and a verification plot showing the decrease in the cost function over iterations. Challenges include observed issues with accuracy improvement and displaying the verification plot, which need further investigation and debugging. Potential improvements involve careful debugging and code refactoring for modularity and clarity.

As observed, the model doesn't converges to optimal value so our first model is expected to be an optimal solution. As the data is already converged while modelling our linear regression, the model is optimum and with as many iterations no improvement is seen.



II. Grid Search Hyper Parameter Tuning Algorithm And The Grid Search Best Estimator

Grid Search passes all combinations of hyperparameters one by one into the model and check the result. Finally it gives us the set of hyperparemeters which gives the best result after passing in the model.

This python source code does the following:

1. Imports the necessary libraries

2. Loads the dataset and performs train_test_split

3. Applies GradientBoostingClassifier and evaluates the result

4. Hyperparameter tunes the GBR Classifier model using GridSearchCV

8. Github Repository Link:

https://github.com/svarg1-unh/Fall-2023-Data-Mining/tree/main/Phase6