Government of Kerala

COMPENDIUM OF PROJECT REPORTS

#AIFORALL

Capacity Building in Artificial Intelligence & Data Analytics

Conducted by

DIGITAL UNIVERSITY KERALA

From 11th Oct 2023 to 21st Dec 2023

DEPARTMENT OF ECONOMICS & STATISTICS
THIRUVANANTHAPURAM

Government of Kerala

COMPENDIUM OF PROJECT REPORTS

#AIFORALL

Capacity Building in

Artificial Intelligence & Data Analytics

Conducted by

Kerala University of Digital Sciences Innovation and Technology

From 11th Oct 2023 to 21st Dec 2023

In collaboration with

Department of Electronics & Information Technology Kerala

&

Kerala State Information Technology Mission

February 2024

DIRECTORATE OF ECONOMICS & STATISTICS

VIKAS BHAVAN, THIRUVANANTHAPURAM

Government of Kerala

Dr. RATHAN U. KELKAR IAS
SECRETARY TO GOVERNMENT

Electronics & Information Technology&
Environment Department
Government Secretariat
Thiruvananthapuram, Govt. of Kerala

Phone - Office : 0471-2518444
0471-2336602
E-mail : secy.itd@kerala.gov.in
secy.envt@kerala.gov.in

Date :17-02-2024

MESSAGE

Government funded programme of Capacity Building in Artificial Intelligence and Data Analytics is aimed at creating a set of employees in government departments to manage data for informed decision making with the support of advanced IT and data science technology and also to integrate Machine Learning and AI in government IT projects. It is learned that a group of 11 statistical staff from Department of Economics & Statistics has participated this programme conducted by Digital University Kerala in 2023. Last year also, 19 employees from the department have successfully undergone this course. This course will be extremely helpful to statistical staff for expanding their IT knowledge and data processing capability and thus to bring more robust decision supportive analytics, in their organisations.

At this juncture, I would like to congratulate DUK Vice Chancellor Dr. Saji Gopinath and his team for organising this programme in a wonderful way. And, I would like to wish all the trainees in the Department for expanding their official career and serving the Government more better way.

I hope this compendium of reports will be a valuable reference for researchers and those who engaged in the field of Data Science and Official Statistics.

DR. RATHAN U KELKAR IAS

Prof (Dr.) Saji Gopinath
Vice Chancellor

Kerala University of Digital Sciences
Innovation & Technology
Technocity Campus, Mangalapuram
Thiruvananthapuram- 695317
Date: 09-02-2024

MESSAGE

I am extremely happy to note that the Capacity Development Program in Artificial Intelligence and Data Analytics designed and delivered by Digital University for officials of Department of Economics and Statistics (DES) has successfully completed its third edition. This unique program is designed to realise the Kerala's commitment to become a truly digital State where advanced technologies are used by Government for improving citizen services, by imparting high end technology skills to Government employees in a structured manner. It is heartening to note that the program supported financially by Department of Electronics and IT, Government of Kerala has immensely benefitted officers of DES with proven expertise in Statistics to hone their skills further, evidence of which are already visible in the excellent projects being initiated by them as part of the program. I am sure that DUK capacity building training has enhanced their capabilities to build their own services and systems using cutting edge tools like AI & data analytics to address specific challenges faced by the Department in a comprehensive manner.

Prof. Saji Gopinath

Prof. Elizabeth Sherly
Distinguished Professor
sherly@duk.ac.in

Kerala University of Digital Sciences
Innovation & Technology
Technocity Campus, Mangalapuram
Thiruvananthapuram- 695317
Date :09-02-2024

ACKNOWLEDGEMENT

We are indeed happy to present the third batch training #AIFORALL compendium, compiled by Economics and Statistics Department (DES) of Government of Kerala. The programme successfully completed with 45 days intensive training aiming of Creating Future talents in accordance with the changing needs of Government. Upskilling Government employees for current and future technologies by leveraging the potential of Artificial Intelligence to position themselves as leaders is the objective of the training programme. The capacity building imparted to DES was a great success and the officials made a remarkable contribution by implementing projects in critical areas of health, agriculture using their own department data. The projects include Utilization and control of Electricity Consumption, Suicide its cause and reduction, Time series analysis of Price of Tomato, farm prices, Paddy yield, Milk Production etc using different machine learning and deep learning models.

Congratulations to all the participants who have successfully completed the course and projects and we are grateful to the Director Mr. Sreekumar B, Deputy Director Mr. D S Shibukumar & Deputy Director Mr. Abhilash K for constant interaction and cooperation during the training. The programme would not have been a success without the dedication and sincere efforts of trainers from Virtual Resource Centre for Language Technology (VRCLC) of DUK and External trainers Mr. Prasad K Nair, Project Head, GBS Technologies Trivandrum,, Mr. Somasekharan, Professor (Rtd) in Statistics, University College and Dr. Satheesh Kumar, Professor, Future Studies of University of Kerala.

I am grateful to Dr. Saji Gopinath, Vice Chancellor and Dr. Mujeeb A, Registrar of Digital University of Kerala for their instinct support and encouragement throughout the training. We are indebted IT Department, Government of Kerala for their vision to enhance the capabilities of Government employees to meet the futuristic technologies for their services and the financial support to conduct the capacity building Programme.

Elizabeth Sherly

Government of Kerala

SREEKUMAR B
DIRECTOR

Department of Economics & Statistics
Vikas Bhavan P.O, Thiruvananthapuram- 695033
Phone - Office: 0471- 2305318, Fax: 0471- 2305317
Phone - Res: 0471- 230090
Email: ecostatdir@gmail.com
Website: www.ecostat.kerala.gov.in
Date: 17-02-2024

PREFACE

Information and Communication Technology sector is growing very fast and is very crucial for managing data. As the Department of Economics & Statistics is a data producing organisation, the application of ICT and data science have much importance in timely dissemination of the data products with adequate inference and interpretation.

This compendium is a consolidation of project reports prepared and presented by the staff of the Department of Economics & Statistics who have undergone the Capacity building Programme in Artificial Intelligence and Data Analytics which was conducted by Digital University of Kerala for a period of 45 days from 11th October to 21st December 2023. This is the 2nd batch of AI training which was exclusively conducted for statistical pesonnel. In this batch, time a team of 11 officers of different cadre has succesfully completed the AI course.

This data science related short term course was very helpful to the Department of Economics & Statistics for transforming its conventional practice of analysis to the advanced analysing practices by blending ICT capabilities with the domain knowledge of existing statistical personnel. I would like to express my gratitude to Dr. Saji Gopinath, Vice Chancellor DUK, Dr. Elizabeth Sherly, Distinguished Professor and Course Co-ordinator DUK, Sri. Surag M, Scientific Associate & Assistant Coordinator of the AI batch, DUK for their sincere co-operation to consider staff, exclusively for DES in the batch and this according to their convenience.

I congratulate all participants who have successfully completed the course and submitted their project reports. I also appreciate the participants for their extra effort to attend the course in addition to their normal duties. I extend my sincere thanks to Additional Director (General), Sri. Santhoshkumar P. D., Deputy Director Sri. D. S. Shibukumar, and Deputy Director, Sri. Abhilash K, for their earnest efforts in co-ordinating the programme and make it a big success.

I hope that this report would be a future reference to all statistical personnel engaged in the data analytics excercises and for transforming the State Statistical System (SSS) with the use of advanced data analytics tools.

Sreekumar B

Trainees of the Programme

1	**Smt. Yamuna A.R, Deputy Director, Vital Statistics Division, Directorate**
2	**Smt. Deepa S.A, Deputy Director, Directorate of Agriculture & Farmer's Welfare**
3	**Sri. Abdul Gafoor, Assistant Director (IIP), Directorate**
4	**Sri. Sajin Gopi, Assistant Director (State Income), Directorate**
5	**Sri. Sreekumar G, Research Officer, NSS Division, Directorate**
6	**Sri. Saju K, Research Assistant, EARAS Division, Directorate**
7	**Sri. Shibu B.T, Research Assistant, PPC Division, Directorate**
8	**Sri. Brijesh C.J, Statistical Assistant Grade I, PPC Division, Directorate**
9	**Sri. Adarsh R.S, Statistical Assistant Grade I, Directorate of Animal Husbandry**
10	**Kum. Lekshmi S, Statistical Assistant Grade II, EARAS Division, Directorate**
11	**Kum. Reshmi S, Statistical Assistant Grade II, EARAS Division, Directorate**

BEHIND THE PROGRAMME

Coordinators

Department of Economics & Statistics

1. Sri. Santhoshkumar P.D., Additional Director (General)
2. Sri. D.S. Shibukumar, Deputy Director, Computer Division
3. Sri. Abhilash K, Deputy Director, Evaluation Division

Digital University Kerala

1. Dr. Elizabeth Sherly, Distinguished Professor, DUK
2. Sri. Surag M, Scientific Associate (Training & Development), DUK

Faculty Members

Digital University Kerala

1. Dr. Elizabeth Sherly, Distinguished Professor, DUK
2. Dr. Malu G, Research Officer, DUK
3. Sri. Surag M, Scientific Associate (Training & Development), DUK
4. Smt. Nayana Uday, Research Scholar, DUK
5. Smt. Leena G Pillai, Scientist, DUK
6. Smt. Judy K George, Research Scholar, DUK
7. Smt. Sabitha Rani B S, Research Scholar, DUK
8. Smt. Kavya Manohar, Computational Linguist, DUK
9. Smt. Raji Gopinath, Research Scholar, DUK

Other Organizations

1. Sri. Prasad K Nair, Project Manager, GBS Technologies Thiruvananthapuram
2. Dr. Satheeshkumar Krishnan Nair, Professor, Department of Futures Studies, University of Kerala
3. Sri. Somasekharan Pillai, Former HoD, Department of Statistics, University College, Thiruvananthapuram

INTRODUCTION

The Department of Economics and Statistics is the Nodal Agency in Kerala for collection, compilation, analysis, objective interpretation and dissemination of data on all socio-economic sectors of Kerala economy. The State has a robust statistical system in India to organize and conduct census and sample surveys for the planning purpose of State and Central Governments. The department is operationalizing many schemes as per the guidelines and methodological support of Central Statistical Office and National Sample Survey Office under National Statistical Office of the Ministry of Statistics and Programme Implementation (MOSPI), Government of India.

The department is publishing on an average 30 reports of various categories in every year and the digital copies of the reports are available in the official website www.ecostat.kerala.gov.in. The department has started digital publication from 2004. The website now contains around 1100 publication of various category and the digital versions of them are available for free downloads. These publications includes those published from 1955-56 but digitized recently. In addition, the department library contains many more reports of statistical organizations in other States, Central and other State department publications and academic books and references. Conventional methods of statistical analysis are usually performed in the department for preparation of reports.

Data science is the study of data to extract meaningful insights for business. It is a multidisciplinary approach that combines principles and practices from the fields of mathematics, statistics, artificial intelligence, and computer engineering to analyze large amounts of data. Data Science combines Mathematics and Statistics, specialized programming, advanced analytics, Artificial Intelligence (AI) and Machine Learning with specific subject matter expertise to uncover actionable insights hidden in an organization's data. These insights can be used to guide decision making and strategic planning. Data scientists rely on popular programming languages to conduct exploratory data analysis and statistical regression. These open source tools support pre-built statistical modeling, machine learning and graphics capabilities. Python and R are the most popular programming languages in Data Science. Data Science provides latest techniques by blending the ICT technology to analyse the data more efficiently and to provide a best result for effective decision making in a quick and easier manner.

This Compendium consists of Project Reports prepared and submitted by a group of statistical personnel of the Department of Economics and Statistics as part of the Capacity Building Programme on Artificial Intelligence and Data Analytics which was conducted by Kerala University of Digital Sciences Innovation and Technology, which is popularly known as Digital University Kerala (DUK) during the year 2023-24. This programme of AI & DA is the third batch of the DUK with the funding of State IT department and Kerala State IT Mission. For the Department of Economics Statistics, this AI course was the second batch and the course was exclusively conducted for statistical people. The programme was conducted in 45 working days, started from 11 October 2023 and ended on 21 December 2023. For the convenience of the department and offices where the trainees are working, considering the long duration of the course, the programme was scheduled from 9.00 AM to 1.00 PM. The faculty support, lunch and light refreshment

charges were borne by DUK. The to and fro transport facility to the participants has been arranged by DES with the vehicle support of State Academy on Statistical Administration (SASA), Kerala. The 45 days course was successfully completed on 21 December 2023 with project presentation by each participant. After the completion of the programme, project presentations were arranged before the directorate staff during the months of January and February 2024 for famialiaring the data analysis done by the trainees and to give motivation to the rest of the staff.

Evolution of the programme

DUK has called for nominations form DES for joining the programme of third batch AI & DA conducted by DUK with the budgetary support of Govt of Kerala. DES has sought willingness from the directorate staff. Willingness was also sought from some staff working in line departments. Since the programme, as informed by DUK, was to be conducted on a part time manner with FN at DUK and AN at concerned office for normal office duties, priority was given to offices in Thiruvananthapuram headquarters. Priority was also given to statistical personnel having MSc degree in Statistics or Mathematics as the data analysts must know some statistical theories and techniques to understand the outcome. From the officers who have expressed willingness to participate, 14 officials were selected and informed to DUK and requested DUK to consider this as an exclusive batch for DES staff. DUK has agreed with the suggestion of DES and then proceeded to implement. During last year in 2022-23, a total of 19 statistical personnel in different cadre have successfully completed the AI & DA course conducted by DUK.

Digital University Kerala

DUK is a premier institution of excellence in science, technology and management. It actively promotes higher education through its IT facilitated education programs and services across Kerala and beyond. The institution is well-known for its research in Artificial Intelligence, Computational Linguistics and Remote Sensing among others. The institute is a pioneer in conceptualizing and implementing some of India's well recognized IT initiatives in education, agriculture and e-Governance.

Programme syllabus- an overview

Introduction to Python, Python Data Types, control statements in Python, Python functions, modules and packages, Python string, list and dictionary manipulation, Python file operation, Python Data Science- NumPy, MatPlotLib, SciPy, Introduction to machine learning, Regression, clustering, Attribute Selection, classification, SVM, Neural Networks, Text processing. Introduction to Pandas, SeaBorn, Exploratory Data Analysis (EDA) for data, Data cleaning (Null Value Analysis and removal, Dimensionality Reduction, Mean, Median, Mode, Quartile, Correlation analysis, Outlier Detection and removal), Data Pre-processing, Examples of Predictions using Various Models. Introduction to R and RStudio and data analysis in R.

Project Assignment and Evaluation

Two weeks project assignment was given to all trainees during the programme. Trainees have selected their own subjects and datasets for analysis and project preparation. Majority of the trainees used DES datasets for data analysis and project preparation.

Datasets used for project preparation

The course participants have selected their own datasets according to their area of interest. Majority of the trainees have considered department datasets for analysis. The details of datasets used for analysis in this project report are mentioned below.

Electricity consumption data of the household survey conducted in Cherukunnu Grama Panchayat ion Kannur district, milk production data of Integrated Sample Survey of Animal Husbandry Department, Farm price of coconut collected and published by DES on monthly basis, data of Sample Check study conducted for field level correction in methodology under EARAS survey, production statistics of tapioca in Kerala under EARAS survey, data collected for preparation of Consumer Price Index for Rural/ Urban and Combined, Cost of Cultivation survey data, data on suicides in Kerala available under State Crime Records Bureau, Monthly Percapita Consumption Expenditure under National Sample Survey and Civil Registration data of deaths in Kerala for the past years are the datasets used by the participants for project preparation.

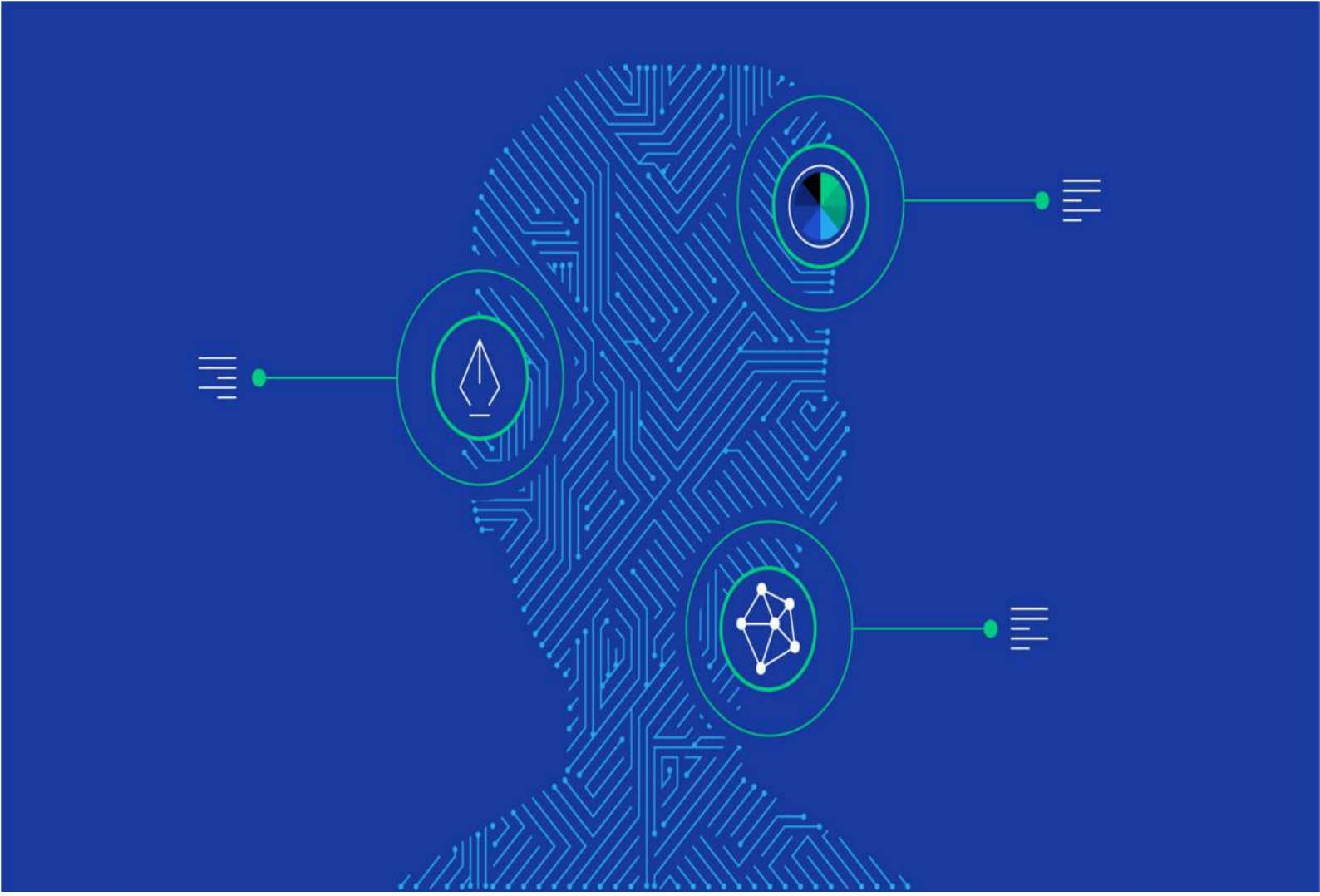
Methods used for analysis

Machine Learning Models- Long Short Term Memory (LSTM) models, Auto Regressive Integrated Moving Average (ARIMA), Seasonal ARIMA (SARIMA), Artificial Neural Networks (ANN), Support Vector Machines (SVM), Random Forest etc. are the methods used by the trainees for analyzing and forecasting data.

Future perspective

This course on AI & DA was extremely useful for the staff in an organization like Department of Economics & Statistics which is dealing with official statistics. The department already has taken steps to set up a Data Analytics Unit in the Directorate. The Data Analytics Unit in the directorate is presently having a team of five officers in different cadre, all have obtained AI training during the last batch of AI & DA conducted by Digital University Kerala. Definitely this kind of initiates will lead to improve the data analytics practices and research activities . Data Analytics Wing is essential to think like a data scientist and analyze the data and converting the department functions into the style of research-oriented activities. The department has a lot of datasets in official statistics and the scope for exploratory data analysis with the support of computer science and technology and preparing beautiful reports will be an essential need for effective planning and informed decision making. DES is one of the prime departments as a data provider for planning purpose. With the support of Data Science, the department can transform itself through the advanced level of capacity building programmes.

INDEX OF PROJECT REPORTS



Analyzing Features of Residential Electricity Consumption through AI Approaches

Submitted by

Sri. Abdul Gafoor,

Assistant Director

Executive Summary

This report presents a comprehensive analysis of residential electricity consumption using advanced AI approaches. The research uses advanced methods to figure out which factors have a significant impact on electricity usage. By leveraging artificial intelligence, we aim to uncover patterns, correlations, and insights that traditional methods may overlook.

Residential electricity consumption is a multifaceted phenomenon influenced by numerous factors, and this report delves into a comprehensive analysis utilizing advanced artificial intelligence (AI) approaches. The aim is to unravel intricate patterns, correlations, and insights that traditional methods often fail to capture. By leveraging the power of AI, this study seeks to enhance our understanding of the dynamic interplay between various features and their impact on residential electricity consumption.

From our investigation, we derive the linear equation: $Y=74.033\ X_1 +108.131\ X_2 + 349.248\ X_3 + 0.145\ X_4 + 165.839\ X_5 + 111.702\ X_6 + 5.438\ X_7 + 17.403\ X_8 + 146.187\ X_9 +133.309\ X_{10} + 119.654\ X_{11} - 141.982\ X_{12}+C$ *Where* X_1 = *Number of Fan,* X_2 = *Family size* , X_3 = *AC(1/0)** , X_4 = *Building Area* , X_5 = *Religion 2 (Islam)(1/0) ** , X_6 = *Income from Pravasi(1/0) **, X_7 = *Number of LED,* X_8 = *Number of CFL,* X_9 = *Number of Laptops,* X_{10} = *Number of Tablet,* X_{11} = *Income from Business(1/0) **, X_{12} = *Floor type (Cement,Tiles)(1/0) * ,C is a constant and here C=0,* *AC, Religion2 (Islam), Income from Pravasi, Income from Business, and Floor type (Cement, Tiles) are binary variables. This means they take the value of 1 if the corresponding facility is available, and 0 if it is not available.

This equation represents the relationship between these factors and residential electricity consumption. Each coefficient (e.g., 74.03, 108.13) signifies the impact of the corresponding variable on the predicted electricity consumption. The binary variables (1/0) indicate the presence (1) or absence (0) of certain features, such as AC, Religion 2 (Islam), Income from Pravasi, and Floor type (Cement, Tiles).

The findings from this analysis serve as a valuable basis for informed decision-making within the energy sector. It's crucial to recognize that the dataset originates from census data, specifically focusing on Cherukunnu Gramapanchayath in Kannur. However, it may not encompass all aspects of the entire population, offering a glimpse into the features relevant to this specific area. The R-squared value of 49.19% in the regression model indicates that approximately 49.19% of the variability in electricity bills (ElecBill) can be explained by the included independent variables. While this measure gauges the model's fitness, it's important to note that a portion of the variability remains unaccounted for.

1. Introduction:

Residential electricity consumption represents a critical facet of modern energy dynamics, and understanding the underlying factors influencing consumption patterns is paramount for effective energy management. In this report, we embark on a comprehensive exploration of residential electricity consumption, leveraging advanced Artificial Intelligence (AI) approaches to discern nuanced relationships and uncover insights that transcend traditional analytical methods.

The significance of this analysis lies in the recognition of the evolving landscape of energy consumption, shaped by diverse socio-economic factors, technological advancements, and changing lifestyles. As we transition towards more sustainable and efficient energy practices, a deep understanding of the features impacting electricity consumption becomes imperative. Unveiling the Synergy of Floor Area, Family Dynamics, and Beyond Provide a concise overview of how AI is applied to comprehend the impact of specific Features on residential electricity consumption

In conducting this analysis, a robust dataset involving a socio-economic survey conducted in November 2021 at Cherukunnu Gramapanchayath in Kannur District. A dataset encompassing information from 3800 households was meticulously curate, covering both household and individual details. This comprehensive dataset forms the basis for the AI-powered investigation outlined in this report. The survey involves gathering the electricity bills for each residential house over a two-month period.

To ensure the reliability and accuracy of the analysis, rigorous data preprocessing techniques were applied. This involved addressing missing values, encoding categorical variables, and creating a clean and standardized dataset suitable for advanced AI modeling.

The heart of the analysis lies in the exploration of influential features on residential electricity consumption. Leveraging machine learning algorithms, regression models were employed to quantify relationships and predict consumption patterns. Techniques such as Ordinary Least Squares (OLS), linear regression and Ridge regression were utilized to provide a nuanced understanding of how various features contribute to electricity usage.

Recognizing the potential impact of outliers on model performance, the analysis includes outlier removal techniques, specifically employing box plot analysis and Z threshold.

In assessing the multicollinearity among features, Variance Inflation Factor (VIF) analysis was conducted, offering insights into potential redundancies and correlations that might affect the stability of the regression models.

Principal Component Analysis (PCA) was incorporated to reduce dimensionality and identify patterns in the dataset. Cumulative explained variance ratios were scrutinized to determine the optimal number of principal components contributing to electricity consumption patterns.

The results and interpretations obtained from these analyses provide a nuanced understanding of the features influencing residential electricity consumption. Notably, the Linear Regression model emerged as a powerful tool, offering comprehensive insights into the relationships within the dataset.

In closing, this report stands as a testament to the integration of AI approaches in unraveling the intricacies of residential electricity consumption, providing actionable insights for stakeholders in the energy sector and beyond.

2. Objective of the Analysis:

1. **Identify Influential Features:** Uncover and quantify the features within households that significantly impact electricity consumption.
2. **Leverage AI Techniques:** Harness the power of AI, including machine learning algorithms, to analyze complex relationships and patterns within the dataset.
3. **Inform Decision-Making:** Provide actionable insights for stakeholders, policymakers, and energy professionals to make informed decisions for optimizing residential electricity consumption.
4. **Contribute to Sustainable Practices:** Facilitate the transition towards sustainable energy practices by understanding the dynamics of electricity usage in residential settings.

3. Data Collection:

The study utilized a robust dataset that include a socio-economic survey conducted in November 2021 at Cherukunnu Gramapanchayath in Kannur District. Information from 3800 households was gathered through census methodology. The survey specifically concentrated on gathering details related to households and amenities. Additionally, individual information was also collected. focusing on both household and individual details. This dataset provides a rich foundation for AI-driven analysis. This dataset includes a wealth of information, ranging from household demographics to specific details about amenities and appliances. The comprehensive nature of the survey ensures a holistic understanding of the factors contributing to electricity bill.

a. Dataset

The provided dataset includes 37 parameters and consists of 3785 entries after the removal of missing values.

	A1_ID	Ward	Team	SrlN	ClosStat	HsNo	Catag	Relegn	famMemb	RationType	...	NonStar	LEDN	CFLN	FilamentN	FanN	ToiletN	Desktop	Lapto
0	A110A009	10	A	9	1	317	1	1	1	2	...	0	0	3	0	0	1	0	
1	A101C085	1	C	85	1	315	3	3	1	5	...	0	4	0	0	0	4	0	
2	A103A032	3	A	32	1	90	3	3	1	1	...	0	2	0	0	0	0	0	
3	A103A042	3	A	42	1	92	3	1	1	5	...	0	2	0	0	0	0	0	
4	A106C067	6	C	67	1	44	3	1	1	2	...	0	2	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
3780	A113C024	13	C	24	1	283a	3	2	4	5	...	0	15	0	0	11	7	1	
3781	A107C082	7	C	82	1	273	3	2	7	3	...	0	5	5	1	5	5	0	
3782	A103B046	3	B	46	1	204	3	1	5	4	...	2	8	0	0	5	3	0	
3783	A107C126	7	C	126	1	444	3	1	5	3	...	0	3	5	0	9	5	0	
3784	A108B109	8	B	109	1	155/A	3	2	7	3	...	0	9	0	0	3	7	0	

3785 rows × 37 columns

b. All Parameters:

]'A1_ID', 'Ward', 'Team', 'SrlN', 'ClosStat', 'HsNo', 'Catag', 'Relegn','famMemb', 'RationType', 'IncmSrce', 'HousType', 'ResidYeras','HouseAge', 'RoadType', 'NewsPaper', 'Ownership', 'FloorTp', 'WallTp','RoofTp', 'SqrFeet', 'ElecBill', 'GasYN', 'GasCyl', 'ElecYN', 'SolarYN','ACYN', 'NonStar', 'LEDN', 'CFLN', 'FilamentN', 'FanN', 'ToiletN', 'Desktop', 'Laptop', 'TabN', 'TVYN']

c. Numerical Parameters:

['Ward', 'SrlN', 'HsNo', 'famMemb', 'ResidYeras', 'HouseAge', 'NewsPaper', 'SqrFeet', 'ElecBill', 'GasCyl', 'NonStar', 'LEDN', 'CFLN', 'FilamentN', 'FanN', 'ToiletN', 'Desktop', 'Laptop', 'TabN',]

d. Categorical Parameters:

['Team', 'ClosStat', 'Catag', 'Relegn', 'RationType', 'IncmSrce', 'HousType', 'RoadType', 'Ownership', 'FloorTp', 'WallTp', 'RoofTp', 'GasYN', 'ElecYN', 'SolarYN', 'ACYN', 'TVYN']

4. Data Preprocessing:

Data preprocessing is a crucial phase in our analysis, acting as the foundation for accurate and meaningful insights. This process involves a series of steps to clean, transform, and organize the raw data, ensuring that it is suitable for advanced AI modeling. The overarching goal is to address potential issues, such as missing values, scale disparities, and categorical data representations, that could impact the quality and reliability of the subsequent analyses. Prior to further analysis, the following 10 parameters were intentionally removed to mitigate their influence on electricity consumption:

- ['A1_ID', 'Ward', 'Team', 'SrlN', 'ClosStat', 'HsNo', 'RoadType', 'GasCyl', 'ElecYN', 'ToiletN']

a. Handling Missing Values:

The absence of data can create biases and diminish the efficiency of AI models. In this dataset, 12 households lacked electrification, and these records were excluded. Subsequently, robust imputation methods were utilized to handle missing values in the dataset, specifically for variables like Electricity Bill ('ElecBill') and Building Area ('SqrFeet'). Techniques such as mean imputation were applied, taking into consideration the characteristics of the missing data.

b. Encoding Categorical Variables:

Many machine learning algorithms require numerical input, necessitating the transformation of categorical variables. Techniques such as one-hot encoding or label encoding were employed to convert categorical data into a format suitable for model training. This ensures that the algorithms can effectively interpret and utilize these features. The categorical parameters yet to undergo one-hot encoding are as follows:

- ['Catag', 'Relegn', 'RationType', 'IncmSrce', 'HousType', 'Ownership', 'FloorTp', 'WallTp', 'RoofTp', 'GasYN', 'SolarYN', 'ACYN', 'TVYN']

Following the conversion of these categorical parameters into Boolean parameters using one-hot encoding, this process involves generating dummy columns.
(As shown in the provided code)

```
import pandas as pd
columns_to_encode = ['Catag', 'Relegn','RationType', 'IncmSrce','HousType','Ownership',
                     'FloorTp', 'WallTp','RoofTp','GasYN', 'SolarYN', 'ACYN','TVYN']
    # Apply one-hot encoding to each specified column
df= pd.get_dummies(df, columns=columns_to_encode, prefix=columns_to_encode)
```

Then the data set includes 68 parameters and consists of 3785 entries. Dataset given below:

	famMemb	ResidYeras	HouseAge	NewsPaper	SqrFeet	ElecBill	NonStar	LEDN	CFLN	FilamentN	...	RoofTp_5	RoofTp_6	GasYN_1	GasYN_2
0	1	54	45	1	80.00	100	0	0	3	0	...	False	False	True	False
1	1	4	0	1	86.00	100	0	4	0	0	...	True	False	False	True
2	1	25	100	0	86.08	100	0	2	0	0	...	False	False	False	True
3	1	30	10	0	86.08	100	0	2	0	0	...	False	False	True	False
4	1	70	50	0	344.00	100	0	2	0	0	...	False	False	True	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3780	4	24	5	1	3187.00	7000	0	15	0	0	...	True	False	True	False
3781	7	71	40	2	4217.00	7000	0	5	5	1	...	True	False	True	False
3782	5	46	13	1	2421.00	8000	2	8	0	0	...	True	False	True	False
3783	5	54	6	0	3895.00	8000	0	3	5	0	...	True	False	True	False
3784	7	50	4	0	5713.66	8000	0	9	0	0	...	True	False	True	False

3785 rows × 68 columns

Subsequently, transforming them into binary parameters

SqrFeet	ElecBill	NonStar	LEDN	CFLN	FilamentN	...	RoofTp_5	RoofTp_6	GasYN_1	GasYN_2	SolarYN_1	SolarYN_2	ACYN_1	ACYN_2	TVYN_1	TVYN_2
80	100	0	0	3	0	...	0	0	1	0	0	1	0	1	1	0
86	100	0	4	0	0	...	1	0	0	1	0	1	0	1	1	0
86	100	0	2	0	0	...	0	0	0	1	0	1	0	1	0	1
86	100	0	2	0	0	...	0	0	1	0	0	1	0	1	0	1
344	100	0	2	0	0	...	0	0	1	0	0	1	0	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3187	7000	0	15	0	0	...	1	0	1	0	0	1	1	0	1	0
4217	7000	0	5	5	1	...	1	0	1	0	0	1	1	0	1	0
2421	8000	2	8	0	0	...	1	0	1	0	0	1	0	1	1	0
3895	8000	0	3	5	0	...	1	0	1	0	0	1	1	0	1	0
5713	8000	0	9	0	0	...	1	0	1	0	0	1	0	1	1	0

The resulting 68 parameters are as follows:

- ['famMemb', 'ResidYeras', 'HouseAge', 'NewsPaper', 'SqrFeet', 'ElecBill', 'NonStar', 'LEDN', 'CFLN', 'FilamentN', 'FanN', 'Desktop', 'Laptop','TabN', 'Catag_1', 'Catag_3', 'Relegn_1', 'Relegn_2', 'Relegn_3','RationType_1', 'RationType_2', 'RationType_3', 'RationType_4', 'RationType_5', 'IncmSrce_1', 'IncmSrce_2', 'IncmSrce_3', 'IncmSrce_4','IncmSrce_5', 'IncmSrce_6', 'IncmSrce_7', 'IncmSrce_8', 'IncmSrce_9', 'IncmSrce_10','IncmSrce_11','IncmSrce_12','IncmSrce_13','IncmSrce_14','HousType_1','HousType_2','HousType_3','Ownership_1','Ownership_2','Ownership_3','FloorTp_2','FloorTp_3','FloorTp_4','FloorTp_5', 'FloorTp_6', 'FloorTp_7', 'WallTp_1', 'WallTp_2', 'WallTp_3', 'WallTp_4', 'WallTp_5', 'RoofTp_2', 'RoofTp_3', 'RoofTp_4', 'RoofTp_5', 'RoofTp_6', 'GasYN_1', 'GasYN_2', 'SolarYN_1', 'SolarYN_2', 'ACYN_1', 'ACYN_2', 'TVYN_1', 'TVYN_2']

c. Perfect co-linearity

Each categorical variable is expanded into multiple variables based on the number of codes or groups it contains, and collectively, these variables exhibit mutual co linearity. Subsequently, each categorical variable's last parameter was removed due to mutual co linearity.

- ['Catag_1','Relegn_3','RationType_5','IncmSrce_14','HousType_3','Ownership_3','FloorTp_7','WallTp_5','RoofTp_6','GasYN_2','SolarYN_2','ACYN_2','TVYN_2']

After removing these parameters, there are now 55 parameters, and we examined the summary of the data frame, as displayed below:

```
summary_statistics = df.describe()
summary_statistics
```

	famMemb	ResidYeras	HouseAge	NewsPaper	SqrFeet	ElecBill	NonStar	LEDN	CFLN	FilamentN	...	WallTp_3
count	3785.000000	3785.000000	3785.000000	3785.000000	3785.000000	3785.000000	3785.000000	3785.000000	3785.000000	3785.000000	...	3785.000000
mean	4.268956	49.217173	21.985205	0.540819	1210.526816	1072.014795	0.318890	7.160634	0.984941	0.222193	...	0.004756
std	2.237616	21.950137	18.230011	0.569657	708.688049	902.705232	0.764241	3.479111	1.795000	0.765974	...	0.068806
min	1.000000	0.000000	0.000000	0.000000	80.000000	100.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
25%	3.000000	37.000000	9.000000	0.000000	690.000000	500.000000	0.000000	5.000000	0.000000	0.000000	...	0.000000
50%	4.000000	53.000000	18.000000	1.000000	1016.000000	785.000000	0.000000	7.000000	0.000000	0.000000	...	0.000000
75%	5.000000	65.000000	30.000000	1.000000	1538.000000	1390.000000	0.000000	10.000000	1.000000	0.000000	...	0.000000
max	18.000000	105.000000	105.000000	3.000000	6456.000000	8000.000000	9.000000	25.000000	20.000000	8.000000	...	1.000000

8 rows × 55 columns

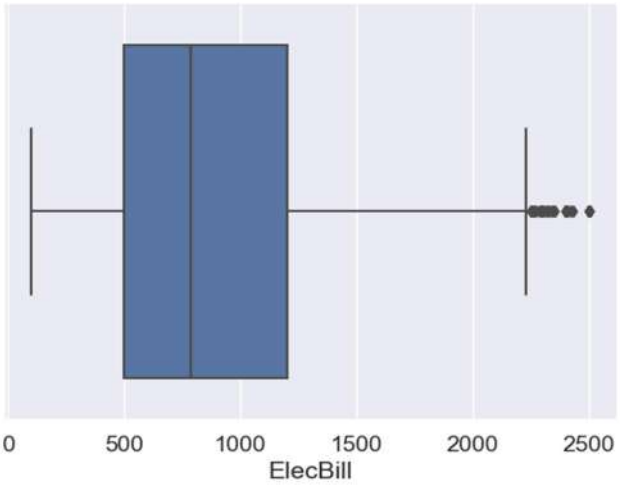
d. Outlier Removal:

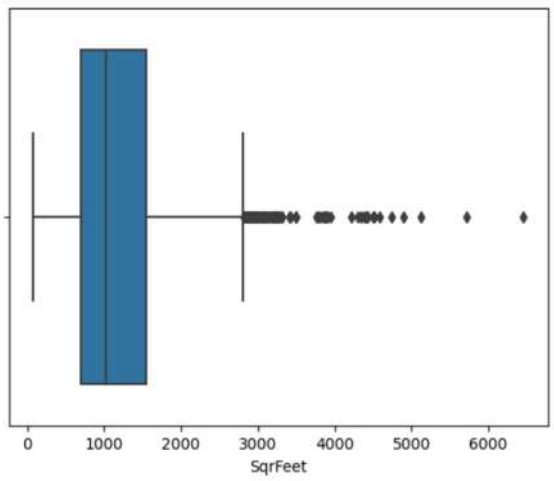
Outliers, which are data points that exhibit substantial deviation from the variables such as Electricity Bill ('ElecBill') and Building Area ('Sqr.Feet') in the dataset, have the potential to distort the training of models and affect predictions.

i. Box plot

Box plots are particularly useful for comparing the distribution of different groups or identifying potential outliers. They offer a visual summary that goes beyond simple measures of central tendency, providing a clearer understanding of the data's variability and skewness. Additionally, box plots are effective in conveying the symmetry or asymmetry of a dataset, making them a valuable tool in exploratory data analysis and statistical reporting.

We generate a box plot to visually represent the distribution of electricity bills and square footage.





ii. Z-threshold for outlier removal

The Z-threshold for outlier removal is a statistical criterion used to identify and eliminate outliers from a dataset. It involves calculating the Z-score for each data point, which represents how many standard deviations a particular value is from the mean of the dataset. A Z-threshold is then set and any data point with a Z-score beyond this threshold is considered an outlier and removed from the dataset. Adjusting the Z-threshold allows for flexibility in the sensitivity of outlier detection, with a higher threshold being more permissive and a lower threshold being more stringent. Upon conducting a Python-based analysis to determine the optimal Z-threshold value, we identified the optimal threshold as 4.5. Employing this optimal Z-threshold value allows us to effectively remove outliers and subsequently evaluate the summary of the data frame.

```
df1.describe()
```

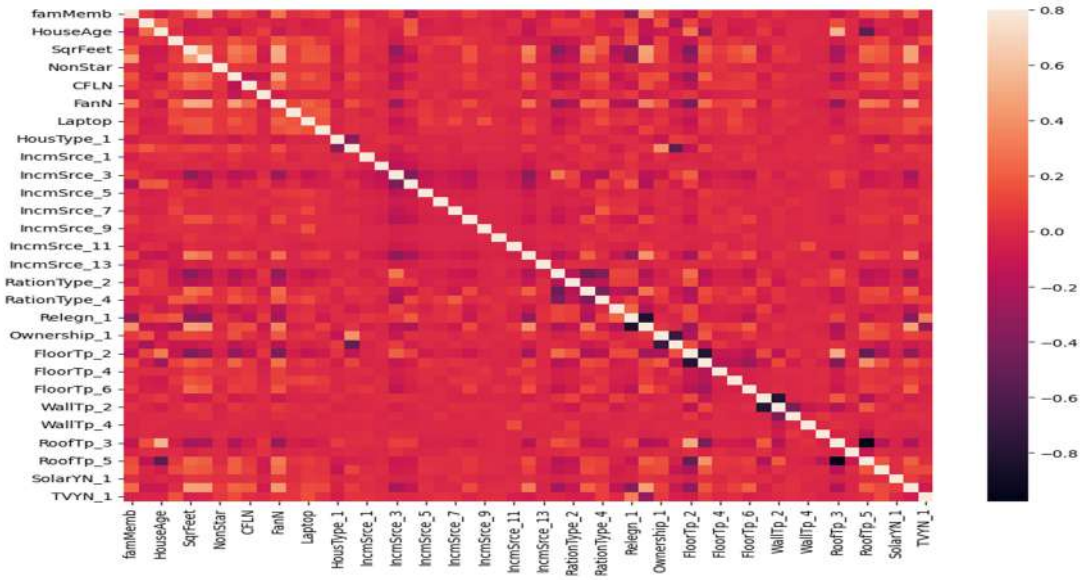
	famMemb	ResidYeras	HouseAge	NewsPaper	SqrFeet	ElecBill	NonStar	LEDN	CFLN	FilamentN	...	WallTp_3	
count	3756.000000	3756.000000	3756.000000	3756.000000	3756.000000	3756.000000	3756.000000	3756.000000	3756.000000	3756.000000	...	3756.000000	3
mean	4.258786	49.207401	21.969382	0.540469	1195.349840	1039.669862	0.319755	7.143770	0.982694	0.222311	...	0.004526	
std	2.236631	21.958340	18.252216	0.568795	675.504874	804.429368	0.765253	3.468188	1.790779	0.764626	...	0.067133	
min	1.000000	0.000000	0.000000	0.000000	80.000000	100.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	
25%	3.000000	37.000000	9.000000	0.000000	688.000000	500.000000	0.000000	5.000000	0.000000	0.000000	...	0.000000	
50%	4.000000	53.000000	18.000000	1.000000	1004.000000	780.000000	0.000000	7.000000	0.000000	0.000000	...	0.000000	
75%	5.000000	65.000000	30.000000	1.000000	1519.000000	1350.000000	0.000000	10.000000	1.000000	0.000000	...	0.000000	
max	18.000000	105.000000	105.000000	3.000000	4390.000000	5100.000000	9.000000	25.000000	20.000000	8.000000	...	1.000000	

8 rows × 55 columns

5. Relationships among parameters

i. Heat map

It's a valuable exploratory data analysis (EDA) tool. It provides an intuitive and comprehensive way to understand relationships within the data, enabling to make informed decisions during feature selection, model building, and other analytical tasks.

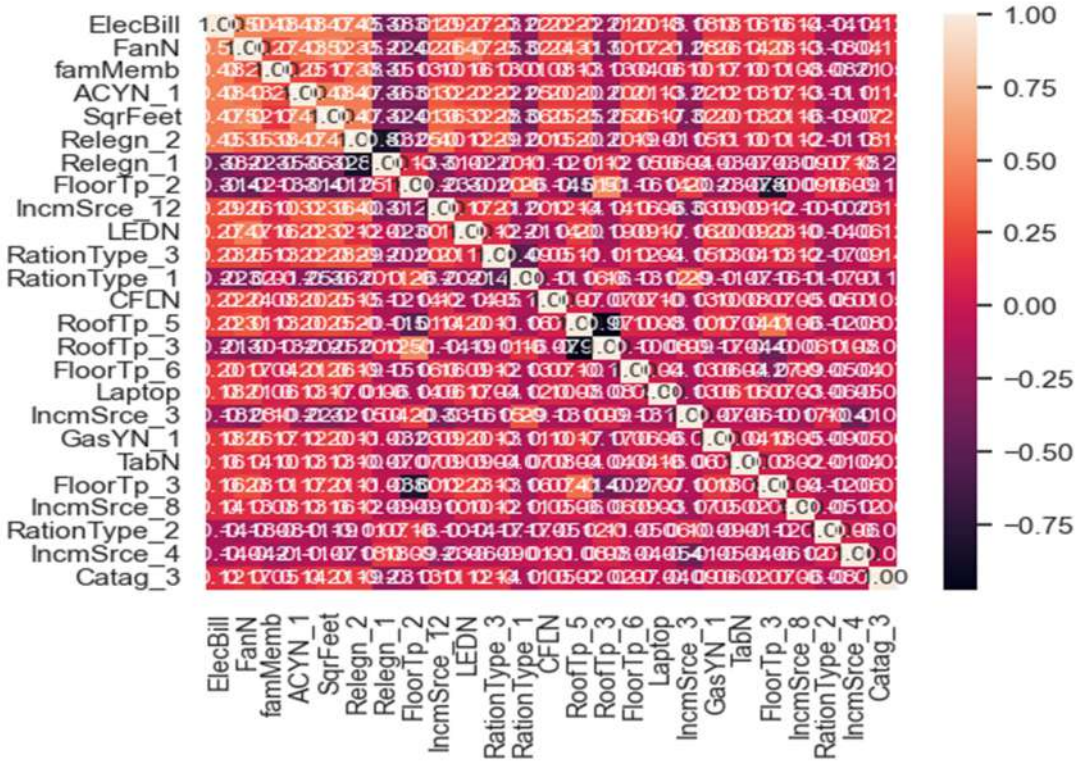


ii. Correlation Matrix

A correlation matrix is a powerful tool for understanding the interdependencies between variables, guiding decision-making in statistical analysis, and providing valuable insights for various analytical tasks. A correlation matrix is a square matrix that displays the correlation coefficients between multiple variables. Each cell in the matrix represents the correlation between two variables, ranging from -1 (perfect negative correlation) to 1 (perfect positive correlation).Display the Correlation Matrix of the Top 25 Most Correlated Features is using the code and output given below:

	ElecBill	FanN	famMemb	ACYN_1	SqrFeet	Relegn_2	\
ElecBill	1.000000	0.503350	0.481329	0.476789	0.465600	0.454999	
FanN	0.503350	1.000000	0.272387	0.427090	0.517174	0.346321	
famMemb	0.481329	0.272387	1.000000	0.245450	0.172665	0.379833	
ACYN_1	0.476789	0.427090	0.245450	1.000000	0.479152	0.467521	
SqrFeet	0.465600	0.517174	0.172665	0.479152	1.000000	0.471027	
Relegn_2	0.454999	0.346321	0.379833	0.467521	0.471027	1.000000	
Relegn_1	-0.381166	-0.218691	-0.353275	-0.363179	-0.324529	-0.832141	
FloorTp_2	-0.305009	-0.423830	-0.134845	-0.307347	-0.405229	-0.250491	
IncmSrce_12	0.294292	0.260609	0.099022	0.316339	0.356545	0.404081	
LEDN	0.266939	0.466017	0.155805	0.221558	0.324101	0.122063	
RationType_3	0.232031	0.245907	0.131948	0.219606	0.277066	0.291064	
RationType_1	-0.221852	-0.319636	0.005330	-0.245664	-0.364737	-0.197929	
CFLN	0.216257	0.242416	0.084230	0.203325	0.246244	0.154210	
RoofTp_5	0.215002	0.309256	0.128089	0.202539	0.249062	0.204282	
RoofTp_3	-0.213497	-0.300856	-0.125090	-0.200346	-0.246044	-0.202253	
FloorTp_6	0.195765	0.170421	0.044624	0.207729	0.256615	0.194494	
Laptop	0.183538	0.213030	0.062819	0.134676	0.173497	-0.014901	
IncmSrce_3	-0.180668	-0.278208	0.100881	-0.220590	-0.316860	-0.150414	
GasYN_1	0.178651	0.258686	0.168466	0.122259	0.195688	0.112865	
TabN	0.159666	0.144490	0.096134	0.131982	0.130757	0.096021	
FloorTp_3	0.157197	0.276371	0.112207	0.165782	0.205500	0.111916	
IncmSrce_8	0.143393	0.130396	0.084792	0.133480	0.156378	0.122222	
RationType_2	-0.136934	-0.182645	-0.075663	-0.113249	-0.192438	-0.109360	
IncmSrce_4	-0.136302	-0.036902	-0.206822	-0.112829	-0.072469	-0.179011	
Catag_3	0.122509	0.168990	0.051942	0.141116	0.212659	0.191452	

Heat Map of the Top 25 Most Correlated Features



Correlation with Electricity Bill

i. Positive Correlations:

There is a positive correlation between "Electricity Bill" and variables such as "FanN" (Number of Fan)- (0.5033), "SqrFeet" (Building area)- (0.4656), "ACYN_1" (Availability of AC) - (0.4767), "famMemb" (Family size)- (0.4813) and “Relegn_2” (Muslim)-(0.4549). This suggests that as these variables increase, there is a tendency for the "Electricity Bill" to increase as well.

ii. Moderate Correlations:

Moderate positive correlations are observed with variables like “IncmSrce_12” (Income from Pravasi) -(0.2942) , “LED No” - (0.2669) ,”RationType_3” (Blue Card)- (0.2320) ,”RoofTp_5 “(Concrete)- (0.2150) ,”CFL No” (0.2162), and “FloorTp_6” (Marble)- (0.1957). These relationships are not as strong as some of the other positive correlations.

iii. Negative Correlation:

'Relegn_1' (-0.3811), 'FloorTp_2' (-0.3050), and 'RationType_1' (-0.2218) exhibit a negative correlation with 'ElecBill.' This suggests that as the values of these variables decrease, the 'ElecBill' tends to increase.

Understanding these correlations is crucial for identifying influential factors that contribute to changes in electricity consumption. These correlations do not imply causation, and further analysis or experiments may be required to establish causal relationships.

6. Multi-collinearity

Multi-collinearity refers to a situation in multiple regression analysis where two or more independent variables in a model are highly correlated, making it difficult to isolate the individual effect of each variable on the dependent variable. This can lead to unstable and unreliable estimates of the coefficients and their standard errors. One common way to assess multi-collinearity is through the Variance Inflation Factor (VIF).

i. Variance Inflation Factor (VIF):

Variance Inflation Factor (VIF) analysis was conducted to assess multicollinearity among features. This provided insights into potential redundancies and correlations that might affect the stability of the regression models.

VIF measures how much the variance of the estimated regression coefficients increases when your predictors are correlated. A high VIF indicates that the associated independent variable is highly correlated with the other variables in the model, and this can create problems in the regression analysis.

The formula for VIF for a particular independent variable is:

$$VIFi = \frac{1}{(1 - \mathrm{Ri}^2)}$$

Where:

$VIFi$ is the Variance Inflation Factor for the i^{th} variable. Ri^2 is the R^2 value obtained by regressing the i^{th} predictor against all the other predictors.

A VIF of 1 indicates no multi-collinearity. VIF values greater than 5 or 10 are often considered high and may warrant further investigation.

	Feature	VIF
0	ElecBill	-1.685663
1	FanN	8.258871
2	famMemb	7.096159
3	ACYN_1	2.053070
4	SqrFeet	-1.754788
5	Relegn_2	6.047824
6	Relegn_1	9.108084
7	FloorTp_2	8.106958
8	IncmSrce_12	2.028002
9	LEDN	8.027431
10	RationType_3	2.297661
11	RationType_1	2.632863
12	CFLN	1.615967
13	RoofTp_5	42.140319
14	RoofTp_3	10.213247
15	FloorTp_6	2.050758
16	Laptop	1.291278
17	IncmSrce_3	3.092331
18	GasYN_1	18.331906
19	TabN	1.097360
20	FloorTp_3	12.162323
21	IncmSrce_8	1.290691
22	RationType_2	1.331836
23	IncmSrce_4	2.128107
24	Catag_3	12.595432

VIF values help assess the level of multicollinearity in a multiple regression model.VIF value above 10 is often considered high, indicating potential multicollinearity issues. Also a Variance Inflation Factor (VIF) negative value is unusual and technically not possible.

VIF values are expected to be non-negative. To fix problems like negative VIF values, try improving the regression model by adding a constant term. After we have added the constant term, use the following codes to recalculate the VIF for each variable. This will give you a more dependable measure of multicollinearity.

	Feature	VIF
0	const	202.159031
1	ElecBill	1.953175
2	FanN	2.039973
3	famMemb	1.536408
4	ACYN_1	1.607290
5	SqrFeet	2.014606
6	Relegn_2	4.345881
7	Relegn_1	3.610430
8	FloorTp_2	6.093210
9	IncmSrce_12	1.722827
10	LEDN	1.535326
11	RationType_3	1.544543
12	RationType_1	1.808347
13	CFLN	1.242818
14	RoofTp_5	20.036459
15	RoofTp_3	19.982347
16	FloorTp_6	2.009763
17	Laptop	1.145985
18	IncmSrce_3	2.001920
19	GasYN_1	1.132753
20	TabN	1.057398
21	FloorTp_3	5.577280
22	IncmSrce_8	1.229550
23	RationType_2	1.274721
24	IncmSrce_4	1.684059
25	Catag_3	1.115988

‘RoofTp_5’ and ’RoofTp_3’ have VIF values above 10, indicating potential issues with multicollinearity and ‘FloorTp_2’, ‘FloorTp_3’ and ‘FloorTp_6’ ‘have moderate VIF values, suggesting a moderate level of multicollinearity. To tackle this, we can combine ‘RoofTp_5’ and ‘RoofTp_3’ as ’Roof’ and ‘FloorTp_2’, ‘FloorTp_3’ and ’FloorTp_6’ as ’Floor’ helping to address the multicollinearity concerns. Following that, we proceed to recalculate the VIF, and the results are presented below.

	Feature	VIF
0	const	197.542952
1	ElecBill	1.944740
2	FanN	1.989799
3	famMemb	1.530967
4	ACYN_1	1.598317
5	SqrFeet	1.969738
6	Relegn_2	4.314165
7	Relegn_1	3.587489
8	IncmSrce_12	1.719467
9	LEDN	1.524587
10	RationType_3	1.541808
11	RationType_1	1.794165
12	CFLN	1.240248
13	Laptop	1.145843
14	IncmSrce_3	1.998759
15	GasYN_1	1.117944
16	TabN	1.056794
17	IncmSrce_8	1.226888
18	RationType_2	1.265520
19	IncmSrce_4	1.678244
20	Catag_3	1.109707
21	Roof	1.014050
22	Floor	1.039169

These results show that, despite the changes made, 'Relegn_1' and 'Relegn_2' still have VIF values above 3. Additionally, 'Relegn_1' shows a negative correlation with the electricity bill in the correlation matrix. To address multicollinearity, we decide to remove the variable 'Relegn_1' from the model. Finally, the dataset includes 22 parameters, including the 'const' term. We are now proceeding with fitting the model.

7. Data Splitting:

The dataset was split into training and testing sets to facilitate model training and evaluation. This separation ensures that the model is trained on one subset of the data and evaluated on an independent subset, providing a more accurate representation of its performance on unseen data.

8. Model Fitting

Model fitting refers to the process of training a statistical model based on the given data. In the context of linear regression, Ordinary Least Squares (OLS) is a common method used for model fitting.

i. Regression Modeling:

Regression models were employed to quantify and understand the relationships between influential features and electricity consumption. Techniques such as Ordinary Least Squares (OLS), Linear regression and Ridge regression were utilized to model and predict consumption patterns.

a. Ordinary Least Squares (OLS)

In linear regression, the OLS method is employed to find the line (or hyper plane in multiple dimensions) that minimizes the sum of the squared vertical distances (residuals) between the observed and predicted values. OLS model fitting aims to find the best-fitting linear relationship between variables by minimizing the sum of squared differences. It is a foundational method in linear regression analysis, providing estimates for the coefficients that define the regression equation

The Output is

OLS Regression Results

Dep. Variable:	ElecBill	R-squared:	0.483
Model:	OLS	Adj. R-squared:	0.481
Method:	Least Squares	F-statistic:	174.7
Date:	Fri, 19 Jan 2024	Prob (F-statistic):	0.00
Time:	15:02:36	Log-Likelihood:	-29217.
No. Observations:	3756	AIC:	5.848e+04
Df Residuals:	3735	BIC:	5.861e+04
Df Model:	20		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	159.7229	129.915	1.229	0.219	-94.989	414.435
FanN	74.2687	6.699	11.086	0.000	61.134	87.403
famMemb	113.9236	4.882	23.337	0.000	104.352	123.495
ACYN_1	310.8698	28.540	10.892	0.000	254.913	366.826
SqrFeet	0.1409	0.019	7.231	0.000	0.103	0.179
Relegn_2	155.5513	27.828	5.590	0.000	100.991	210.111
IncmSrce_12	124.8631	34.144	3.657	0.000	57.920	191.806
LEDN	4.3693	3.356	1.302	0.193	-2.210	10.949
RationType_3	20.8449	24.974	0.835	0.404	-28.119	69.808
RationType_1	-35.6538	27.023	-1.319	0.187	-88.636	17.328
CFLN	21.7147	5.873	3.697	0.000	10.200	33.230
Laptop	122.4955	27.027	4.532	0.000	69.506	175.485
IncmSrce_3	-45.8192	27.672	-1.656	0.098	-100.073	8.435
GasYN_1	-4.9875	42.690	-0.117	0.907	-88.685	78.710
TabN	110.1502	41.564	2.650	0.008	28.659	191.641
IncmSrce_8	116.0313	48.385	2.398	0.017	21.169	210.894
RationType_2	-44.2820	45.806	-0.967	0.334	-134.090	45.526
IncmSrce_4	-17.8461	29.535	-0.604	0.546	-75.752	40.059
Catag_3	-26.8552	35.809	-0.750	0.453	-97.061	43.351
Roof	-86.9685	108.684	-0.800	0.424	-300.054	126.117
Floor	-101.6163	44.637	-2.277	0.023	-189.131	-14.101

Omnibus:	1407.739	Durbin-Watson:	0.702
Prob(Omnibus):	0.000	Jarque-Bera (JB):	8443.935
Skew:	1.667	Prob(JB):	0.00
Kurtosis:	9.545	Cond. No.	2.36e+04

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified
[2] The condition number is large, 2.36e+04. This might indicate that there are
strong multicollinearity or other numerical problems.

Interpretation

1. R-squared:

The R-squared value is 0.483, indicating that the model explains 48.3% of the variability in the dependent variable 'ElecBill.'

2. Coefficients:

The coefficients represent the estimated impact of each predictor variable on 'ElecBill.' For example, 'FanN' has a coefficient of 74.2687, implying that for each unit increase in 'FanN,' 'ElecBill' is expected to increase by 74.2687 units, holding other variables constant.

3. Confidence Intervals:

The [0.025, 0.975] columns provide the 95% confidence interval for each coefficient. It gives a range within which we can be reasonably confident that the true coefficient lies.

4. P-values

The P-values associated with each coefficient indicate whether the corresponding predictor variable is statistically significant. A P-value less than 0.05 is often considered significant.

9. Feature Analysis:

The application of AI approaches, including machine learning algorithms, facilitated the identification of features influencing residential electricity consumption. Notable features, such as the number of fans, square feet area of the building, and the presence of air conditioning, demonstrated significant correlations with electricity bills.

i. Feature Engineering:

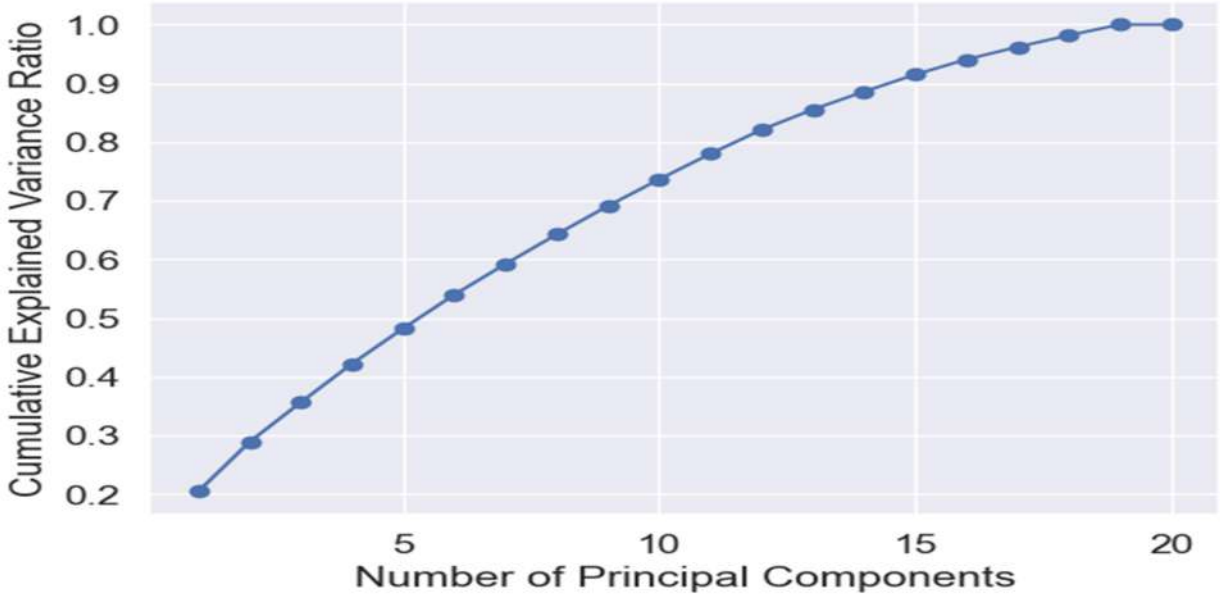
Feature engineering involves creating new features or transforming existing ones to enhance the model's ability to capture patterns. This step was undertaken judiciously,

exploring interactions and combinations of features that could provide additional insights into electricity consumption patterns.

ii. Principal Component Analysis (PCA):

PCA was employed to reduce dimensionality and identify patterns in the dataset. Cumulative explained variance ratios were analyzed to determine the optimal number of principal components. PCA is a powerful tool for reducing the complexity of high-dimensional data, aiding in visualization, and uncovering the underlying structure.PCA is employed to reduce the number of features (dimensions) in a dataset while retaining as much of the original variability as possible. PCA is used for various purposes, including noise reduction, feature extraction, and visualization. We use Principal Component Analysis and find cumulative Explained Variance Ratio

```
Cumulative Explained Variance Ratio:
[0.20411253 0.28842337 0.35599553 0.42180204 0.48196977 0.53953161
 0.59194587 0.64234266 0.69075162 0.73603143 0.77968592 0.8206637
 0.85514653 0.88569008 0.91476028 0.94057094 0.96178304 0.98161683
 1.         1.        ]
```



Based on the given conditions and considering both the p-value and the cumulative explained variance ratio, we decided to remove 9 parameters. The final set of parameters is then determined after this removal.

```
['const', 'FanN', 'famMemb', 'ACYN_1', 'SqrFeet', 'Relegn_2',
'IncmSrce_12', 'LEDN', 'CFLN', 'Laptop', 'TabN', 'IncmSrce_8', 'Floor']
```

10. Model Fitting and Comparison Using OLS, Linear Regression, and Ridge Regression

We proceed to fit the model once more, employing the OLS Regression Model, Linear Regression Model, and Ridge Regression Model. This is done to compare and evaluate the results obtained from each of these regression techniques. Table displaying Features and Coefficients for Ridge Regression, Linear Regression, and OLS

	Feature	Ridge Regression coef	Linear Regression coef	OLS Regression coef
0	const	0.000000	0.000000	42.3534
1	FanN	144.725451	74.032550	76.4887
2	famMemb	241.085857	108.131377	112.5755
3	ACYN_1	144.290072	349.247713	309.4161
4	SqrFeet	97.858021	0.145342	0.1424
5	Relegn_2	76.480320	165.838866	151.8887
6	IncmSrce_12	40.415066	111.702119	146.8790
7	LEDN	19.374532	5.437784	4.8299
8	CFLN	32.195888	17.402578	21.7621
9	Laptop	55.026305	146.186701	129.8658
10	TabN	32.489975	133.308522	109.1885
11	IncmSrce_8	25.987689	119.653945	137.2972
12	Floor	-43.131735	-141.981832	-149.2425
Mean Squared Error		316071.949474	316054.822257	--
Mean Absolute Error		368.990153	369.014567	--
R-squared:		0.491793	0.491820	0.484

Interpretation

The analysis revealed specific features that play a crucial role in influencing residential electricity consumption. Linear regression model provided a comprehensive understanding of the complex relationships between these features and energy usage.

	Feature	Linear Regression coef
0	const	0.000000
1	FanN	74.032550
2	famMemb	108.131377
3	ACYN_1	349.247713
4	SqrFeet	0.145342
5	Relegn_2	165.838866
6	IncmSrce_12	111.702119
7	LEDN	5.437784
8	CFLN	17.402578
9	Laptop	146.186701
10	TabN	133.308522
11	IncmSrce_8	119.653945
12	Floor	-141.981832
Mean Squared Error		316054.822257
Mean Absolute Error		369.014567
R-squared:		0.491820

i. Key Features Impacting Residential Electricity Consumption in the Dataset

1. **Number of Fan**
2. **Family size**
3. **AC (Yes or No)**
4. **Building Area**
5. **Religion 2 (Islam) (Yes or No)**
6. **Income from Pravasi (Yes or No)**
7. **Number of LED**
8. **Number of CFL**
9. **Number of Laptops**
10. **Number of Tablet**
11. **Income from Business (Yes or No)**

12. Floor type (Cement,Tiles) (Yes or No)

Based on the findings, we can calculate residential electricity consumption using the formula

$$Y=74.033\,X_1 +108.131\;\; X_2 + 349.248\,X_3 + 0.145\;\; X_4 + 165.839\,X_5 + 111.702\,X_6 + 5.438\,X_7 + 17.403\,X_8 + 146.187\,X_9 + 133.309\,X_{10} + 119.654\,X_{11} - 141.982\,X_{12} + C$$

Where X_1 = ***Number of Fan*** X_2 = ***Family size***

X_3 = ***AC(1/0)**** X_4 = ***Building Area***

X_5 = ***Religion 2 (Islam)(1/0) **** X_6 = ***Income from Pravasi(1/0) ****

X_7 = ***Number of LED*** X_8 = ***Number of CFL***

X_9 = ***Number of Laptops*** X_{10} = ***Number of Tablet***

X_{11} = ***Income from Business(1/0) **** X_{12} = ***Floor type (Cement,Tiles)(1/0)****

C is a constant and here C=0

*AC, Religion2 (Islam), Income from Pravasi, Income from Business, and Floor type (Cement, Tiles) are binary variables. This means they take the value of 1 if the corresponding facility is available, and 0 if it is not available.

ii. Interpretation

The coefficients in the linear regression model indicate the impact of some respective feature in the electricity bill. Here's the interpretation of the key features:

Number of Fans (FanN): With all other factors held constant, each additional fan in a household is associated with an estimated increase of approximately 74.03 rupees in the electricity bill.

Family Members (famMemb): With all other factors held constant, An additional family member is associated with an estimated increase of about 108.13 rupees in the electricity bill.

AC Availability (ACYN_1): Holding all other factors constant, the presence of an air conditioner (AC) is associated with a significant impact, contributing to an estimated increase of around 349.25 rupees in the electricity bill.

Square Feet Area (SqrFeet): Keeping all other factors constant, each additional square foot of living space corresponds to a modest increase of approximately 0.15 rupees in the electricity bill.

These interpretations help understand the direction and magnitude of the impact each variable has on the electricity consumption prediction.

11. Conclusion:

In conclusion, the application of AI approaches has significantly enhanced our understanding of the features influencing residential electricity consumption. The insights gained from this analysis provide a foundation for informed decision-making in the energy sector. This dataset is derived from census data and specifically pertains to Cherukunnu Gramapanchayath in Kannur. It's important to note that the dataset may not capture all the characteristics of the entire population. Instead, it focuses on the information relevant to Cherukunnu Gramapanchayath, providing a snapshot of the features and variables within this specific area.

When interpreting and generalizing findings from this dataset, it's crucial to consider its limited scope and applicability to the broader population. The model's R-squared value

of 49.19% indicates that approximately 49.19% of the variability in the dependent variable (ElecBill) is explained by the independent variables included in the model. While R-squared is a measure of the model's goodness of fit, it's important to acknowledge that a portion of the variability remains unaccounted for.

Regrssion model's explains about 49.19% of the factors influencing electricity bills in Cherukunnu Gramapanchayath. The remaining 50.81% of the variability is not captured by the model and could be attributed to other factors or inherent complexities that are not considered in the current set of independent variables.

This limitation highlights the need for further investigation, potential inclusion of additional relevant variables, or acknowledgment that there may be inherent variability that cannot be explained solely by the chosen predictors.

12. Acknowledgments:

I would like to sincerely express our gratitude to our Director for affording me the opportunity to participate in the training. Our heartfelt appreciation goes to all the teaching and non-teaching staff of DUK who played an active role in both the project and training, enabling the success of this analysis. Their invaluable efforts have been pivotal to the accomplishments of this project.

13. References:

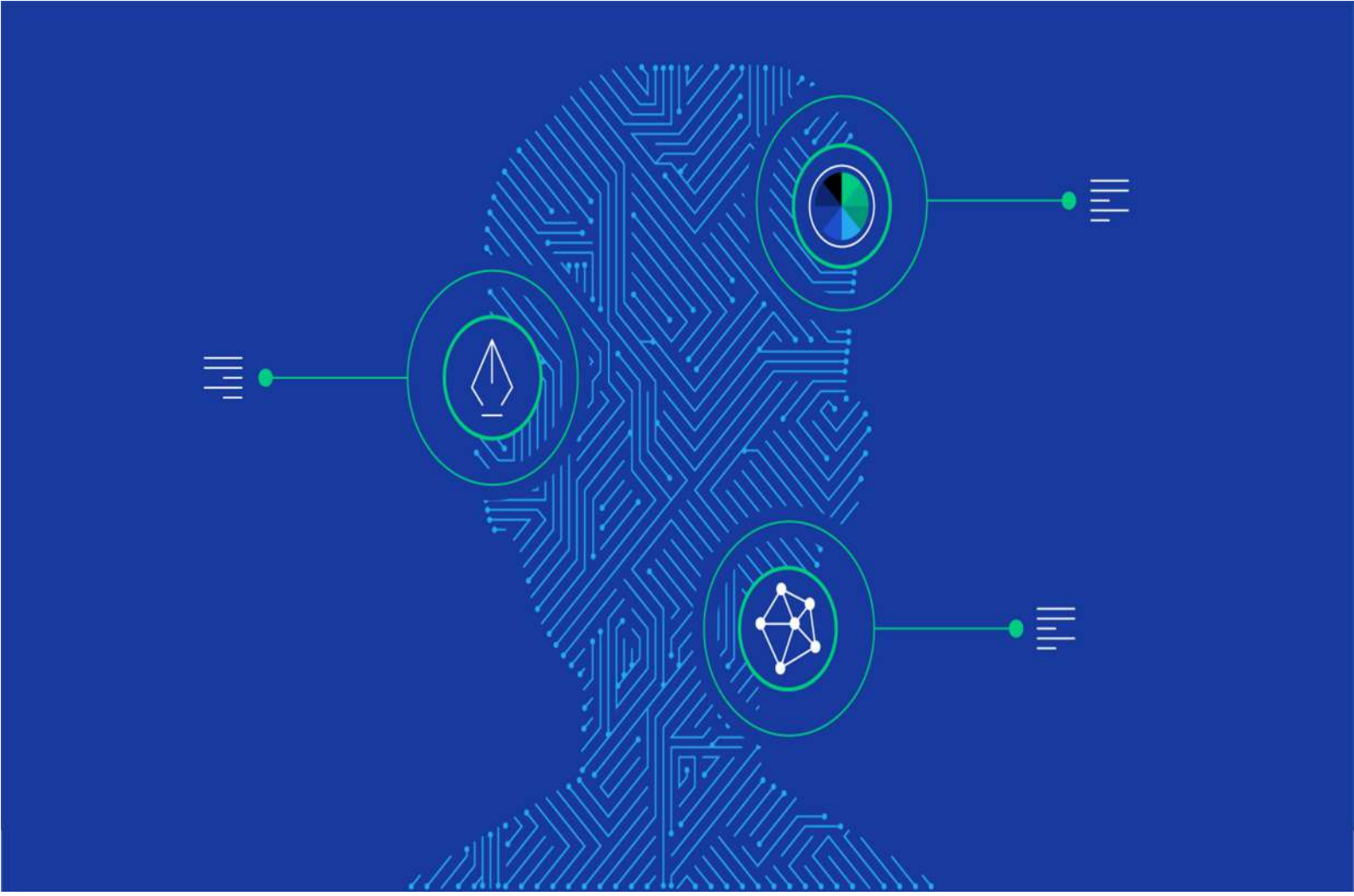
1. Predicting House Prices with Machine Learning – authored by Eric Kim and shared on kaggle.

14. Source:

1. Dataset - Cherukunnu `Gramapanchayath Socio Economic Survey 2021data used

15. Libraries and Tools Used

1. Scikit-learn: Used for machine learning model implementation.
2. Pandas: Utilized for data manipulation and analysis.
3. Matplotlib and Seaborn: Employed for data visualization.
4. TensorFlow: Applied for deep learning tasks.
5. Jupyter Notebooks: Used as the primary environment for code development and experimentation.



Forecasting Milk Production Trends in Kerala using AI Model

Submitted by

Sri. Adarsh R.S,

Statistical Assistant Grade I

1.Introduction

The animal husbandry sector holds significant importance in Kerala, contributing to various aspects of the economy, nutrition, and livelihood. Dairy farming, in particular, plays a crucial role in rural development by providing employment opportunities and supporting local economies. It serves as a means of diversification for agricultural practices in Kerala, allowing farmers to integrate dairy farming with other activities, thereby reducing dependence on a single crop. Moreover, milk forms the foundation for a range of diversified dairy products such as ghee, butter, cheese, and various traditional dairy items, adding value to agricultural produce and catering to diverse consumer preferences.

2.Objective

The objective of this project was to analyses thirty years of milk production data for Kerala and develop a predictive model for future milk production. The data included season – wise and category milk production for cattle buffalo and goat from 1990 to 2021.

3.Data set used

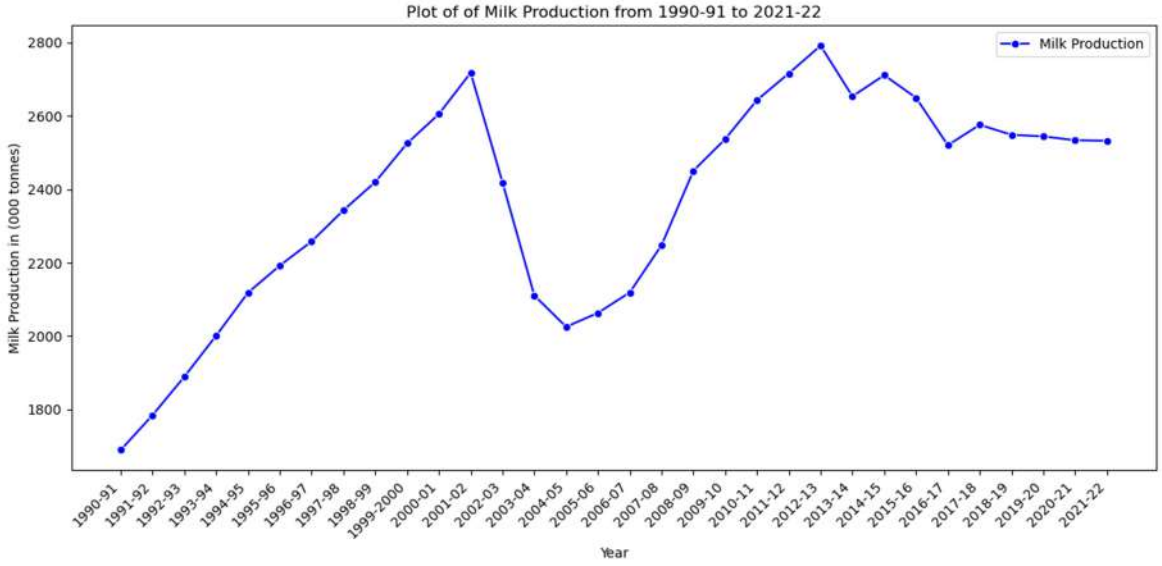
The dataset was obtained from Integrated Sample Survey results, conducted from March to February. The survey period was divided into three seasons:

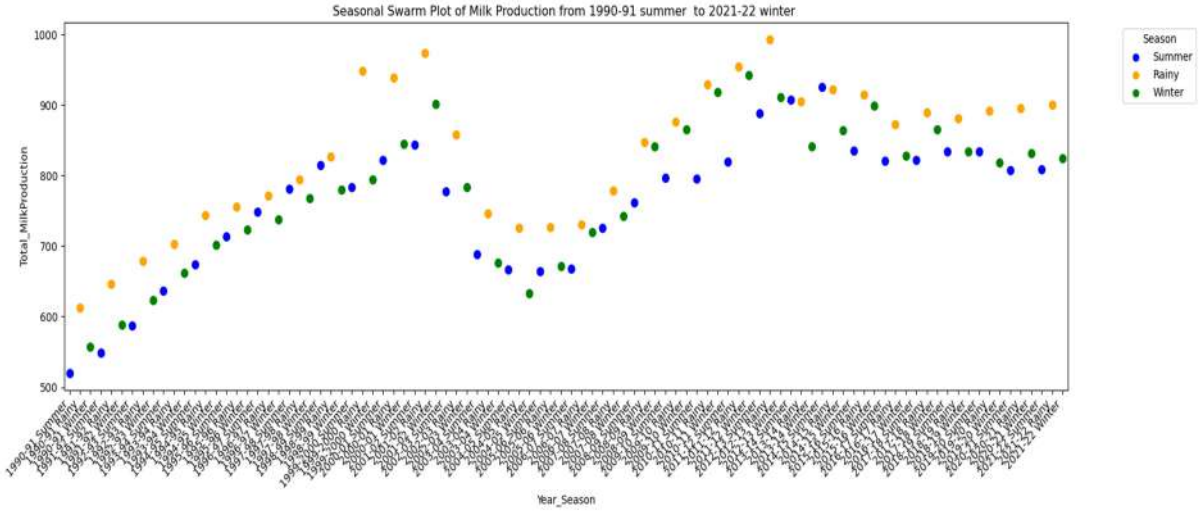
- Summer Season: 1st March to 30th June
- Rainy Season: 1st July to 31st October
- Winter Season: 1st November to 28th or 29th February

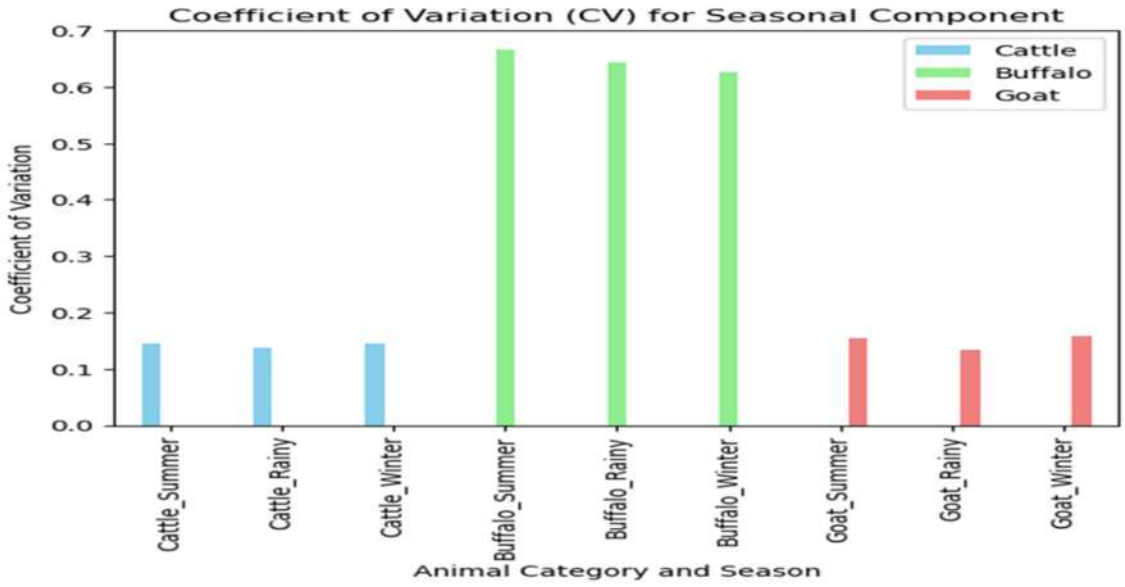
Season wise and category wise milk production of Kerala since 1990 (in 000 tonnes)

Year	Cattle				Buffalo				Goat				
	Summer	Rainy	Winter	Total	Summer	Rainy	Winter	Total	Summer	Rainy	Winter	Total	Grand Total
1990-91	455.11	540.18	496.56	1491.85	34.07	36.25	30.86	101.18	30.93	35.72	29.89	96.54	1689.57
1991-92	481.50	571.12	523.07	1575.69	35.76	38.16	32.96	106.88	32.06	37.33	32.57	101.96	1784.53
1992-93	517.13	602.29	555.37	1674.79	36.47	38.32	34.51	109.30	33.30	37.98	34.06	105.34	1889.43
1993-94	564.87	625.76	590.91	1781.54	36.67	38.55	35.01	110.23	34.98	39.11	35.47	109.56	2001.33
1994-95	601.85	665.67	629.60	1897.12	35.93	37.84	34.82	108.59	35.75	39.85	36.94	112.54	2118.25
1995-96	648.79	684.92	658.55	1992.26	28.89	29.58	26.82	85.29	36.12	40.75	37.80	114.67	2192.22
1996-97	683.46	700.15	674.22	2057.83	26.97	29.00	25.85	81.82	38.33	42.10	38.01	118.44	2258.09
1997-98	716.13	725.52	705.83	2147.48	24.65	25.82	23.58	74.05	39.87	42.73	38.80	121.40	2342.93
1998-99	745.31	762.06	712.30	2219.67	24.99	26.01	23.80	74.80	44.06	38.34	43.17	125.57	2420.04
1999-2000	722.24	873.77	731.51	2327.52	21.65	26.67	21.71	70.03	39.2	48.00	40.52	127.72	2525.27
2000-01	762.60	872.42	784.81	2419.83	19.90	22.84	20.71	63.45	38.87	43.87	39.38	122.12	2605.40
2001-02	784.47	909.66	843.48	2537.61	20.12	22.53	20.34	62.99	38.52	41.83	36.93	117.28	2717.88
2002-03	725.62	804.63	734.46	2264.71	15.77	16.57	15.69	48.03	36.15	37.35	32.74	106.24	2418.98
2003-04	649.18	703.97	638.84	1991.99	12.34	14.61	12.84	39.79	26.65	27.95	24.17	78.77	2110.55
2004-05	630.35	683.85	598.23	1912.43	11.86	14.24	11.83	37.93	25.06	27.10	22.70	74.86	2025.22
2005-06	629.11	685.02	633.39	1947.52	10.79	13.58	12.10	36.47	24.63	28.46	26.12	79.21	2063.20
2006-07	631.56	689.51	681.87	2002.94	10.16	10.99	8.70	29.85	26.25	30.16	29.68	86.09	2118.88
2007-08	687.31	734.91	701.17	2123.39	9.36	9.82	8.17	27.35	29.24	34.14	33.52	96.90	2247.64
2008-09	718.34	802.47	790.32	2311.13	10.20	10.39	15.72	36.31	33.58	34.84	34.67	103.09	2450.53
2009-10	741.47	816.69	808.20	2366.36	12.82	15.46	16.17	44.45	41.91	43.56	40.83	126.30	2537.11
2010-11	754.66	875.51	882.13	2512.30	6.96	8.60	5.43	20.99	33.64	45.48	30.93	110.05	2643.34
2011-12	777.89	898.74	905.36	2581.98	8.97	10.61	7.44	27.02	32.7	44.53	29.99	107.22	2716.22
2012-13	842.12	943.45	862.96	2648.53	9.01	14.21	14.39	37.61	36.65	35.53	33.29	105.47	2791.61
2013-14	851.58	844.69	787.29	2483.56	20.95	21.50	15.60	58.05	35.30	38.91	37.86	112.07	2653.68
2014-15	866.11	860.73	807.12	2533.96	21.83	21.77	18.81	62.41	37.73	39.21	37.86	114.80	2711.17
2015-16	788.13	867.11	852.65	2507.89	4.53	4.95	3.47	12.95	42.61	42.99	43.28	128.88	2649.72
2016-17	772.99	826.88	781.52	2381.39	4.55	4.18	3.48	12.21	43.05	40.92	42.72	126.69	2520.29
2017-18	778.86	842.87	820.81	2442.54	4.22	4.65	3.80	12.67	38.96	41.54	40.27	120.77	2575.98
2018-19	790.20	834.91	789.94	2415.05	4.37	4.01	3.79	12.17	38.99	42.11	40.36	121.46	2548.68
2019-20	790.24	845.09	767.94	2403.27	3.03	4.16	4.76	11.95	40.68	42.67	45.78	129.13	2544.35
2020-21	759.66	857.13	785.45	2402.24	3.75	2.90	5.78	12.43	43.80	34.80	40.61	119.21	2533.88
2021-22	760.09	860.33	775.55	2395.97	4.11	3.12	5.98	13.21	44.07	36.87	42.34	123.28	2532.46

4.Data Analysis







The dataset provided detailed information on milk production in Kerala across different seasons (summer, Rainy, and Winter) and categories (Cattle, Buffalo, Goat) for each year. Initial exploration revealed increasing trends in overall milk production, with variations across categories and seasons. There was also seasonality in the data, with production being higher in the rainy season. Based the coefficient of variation Buffalo milk production shows the highest variability in seasonality changes. Cattle milk production shows relatively low variability in seasonality changes. Goat milk production falls in between, with a moderate level of variability. These interpretations provide insights into how each category of animals responds to seasonal changes in terms of milk production.

5. Methodology

The study utilized machine learning models and Python programming for data analysis. The **SARIMA model**, an extension of the ARIMA model, was employed for time series forecasting, considering the presence of seasonal patterns in the data. Components of SARIMA included Seasonal, Autoregressive, Integrated, and Moving Average components, each defined by specific parameters.

Seasonal (S): This indicates the presence of seasonality in the data.

Autoregressive (AR): This component models the relationship between an observation and a number of lagged observations (autoregression).

Integrated (I): This component accounts for the differencing needed to make the time series stationary, i.e., to remove trends from the data.

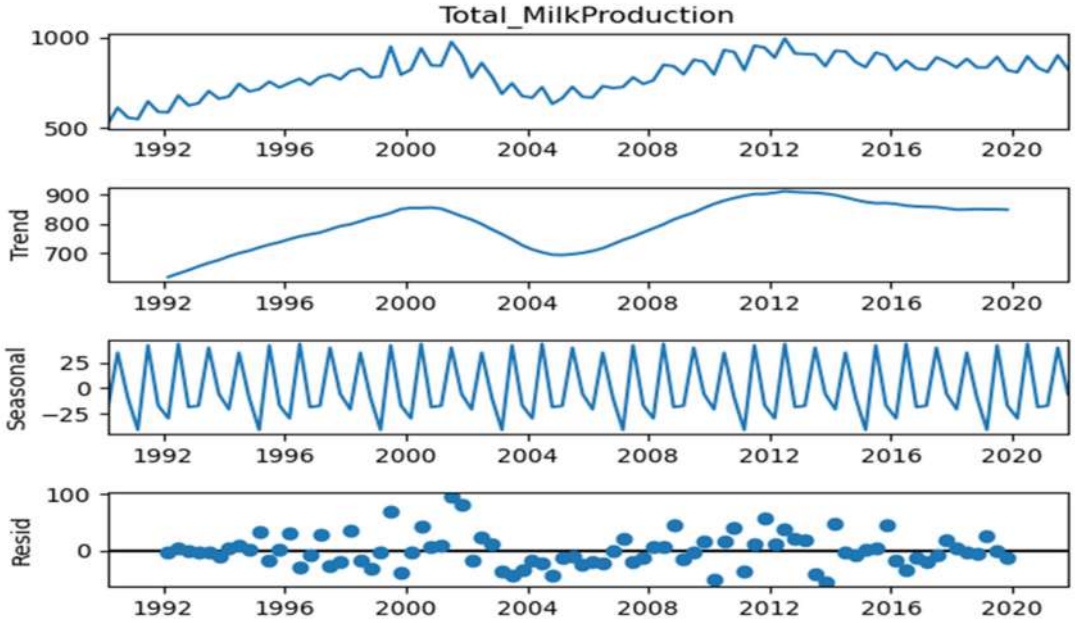
Moving Average (MA): This component models the relationship between an observation and a residual error from a moving average model.

The SARIMA model is specified by three main sets of parameters:

p, d, q: These parameters are for the non-seasonal components of the time series data. They represent the order of the autoregressive (p), differencing (d), and moving average (q) terms, respectively.

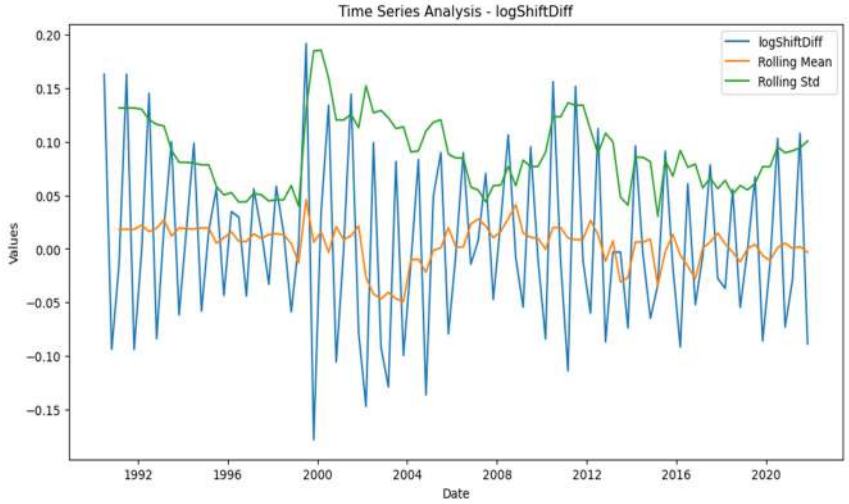
P, D, Q: These parameters are similar to p, d, and q, but they operate on the seasonal component of the time series.

s: This parameter represents the number of time steps in each season



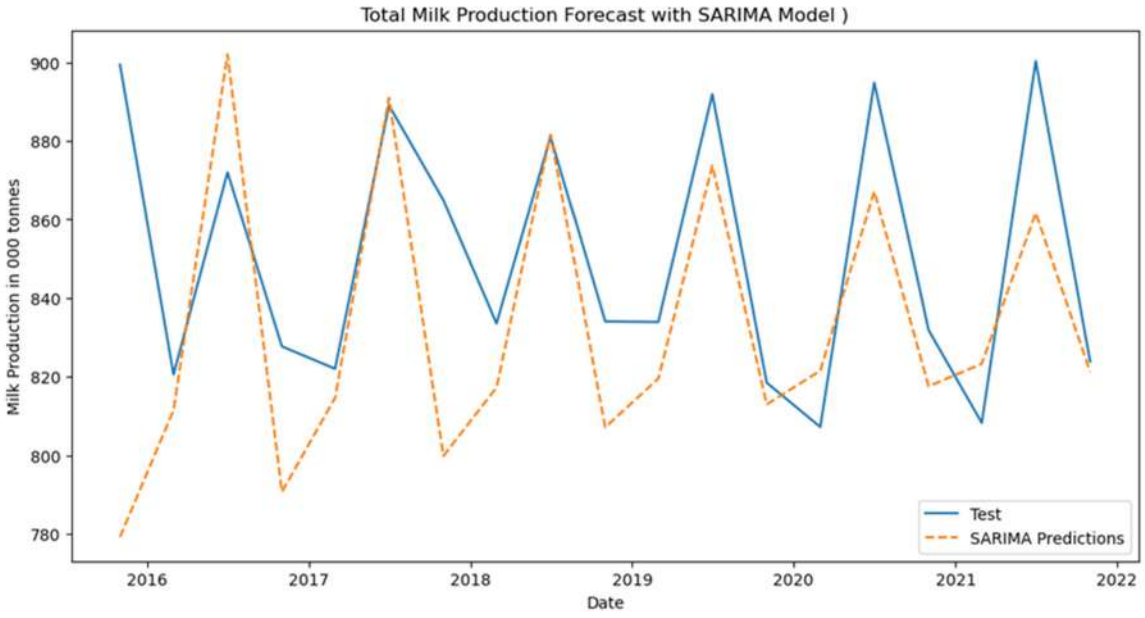
Check Stationarity using Augmented Dickey-Fuller Test

The Augmented Dickey-Fuller (ADF) test is a statistical test used to determine whether a unit root is present in a time series dataset. A unit root indicates that a time series data has a stochastic trend, making it non-stationary. The null hypothesis of the ADF test is that the time series data has a unit root, meaning it is non-stationary. The alternate hypothesis is that the time series data is stationary, meaning it does not have a unit root. In the case of our time series p-value = 0.10262423774638108>0.05, indicates that time series is non-stationary. After applying log transformation and differencing, p-value = 0.000095<0.05, the data became stationary.



Model Fitting:

We took 80% data as train data set and 20 % as test data set. Using the Auto_arima function in the Pdarima library, the best-fitted SARIMA model was determined. In general, lower AIC (Akaike Information Criterion) value indicate a model better fit. The selected model was ARIMA(3,0,1)(0,0,0)[4].



Milk Production Actual vs Prediction using SARIMA Model

Year	Total Milk Production (in 000	Prediction (using SARIMA)
Summer 2017	822.04	814.602475
Rainy 2017	889.06	890.94
Winter 2017	864.88	799.83423
Summer 2018	833.56	817.251008
Rainy 2018	881.03	881.580105
Winter 2018	834.09	807.125008
Summer 2019	833.95	819.538415
Rainy 2019	891.92	873.714269
Winter 2019	818.48	812.940531
Summer 2020	807.21	821.537516
Rainy 2020	894.83	867.134186
Winter 2020	831.84	817.58027
Summer 2021	808.27	823.296254
Rainy 2021	900.32	861.64602
Winter 2021	823.87	821.2811

Model Evaluation:

The predictive model's performance was evaluated using Mean Absolute Percentage Error (MAPE) and Mean Absolute Percentage Error (MAPE):

MAPE measures the average absolute percentage difference between actual and forecasted values, expressed as a percentage. It is calculated using the formula:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{|Actual_i - Forecasted_i|}{|Actual_i|}\ X\ 100$$

Where, n is the number of observations in the data set, $Actual_i$ is the actual value at time i and $Forecasted_i$ is the predicted value at time i

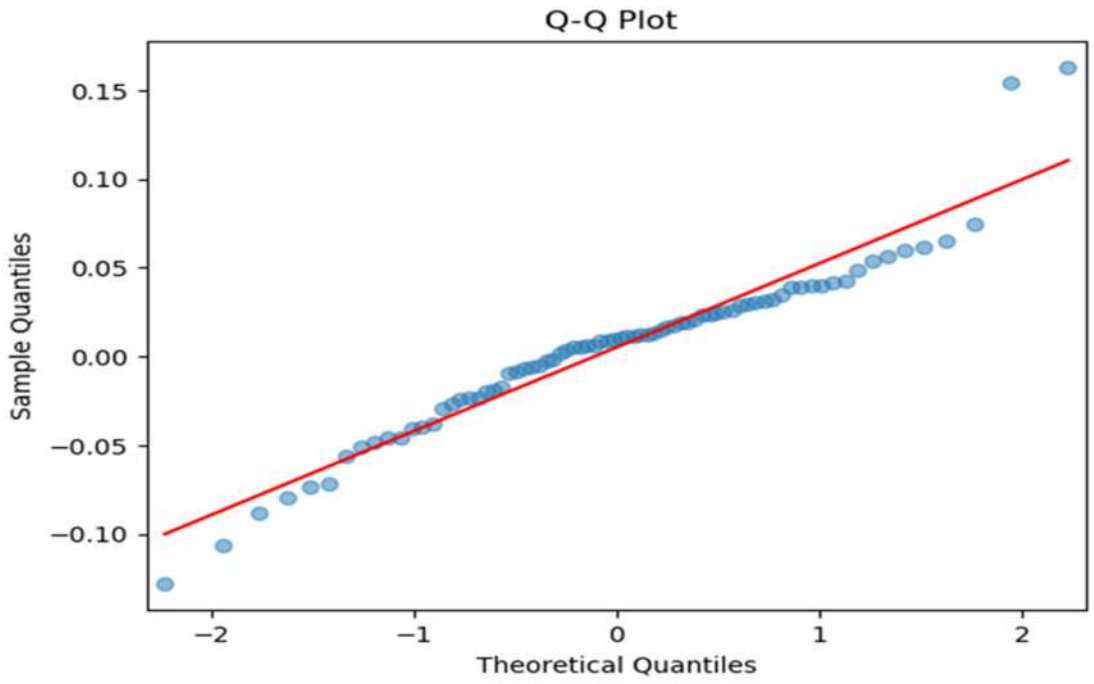
Root Mean Squared Error (RMSE).

RMSE measures the square root of the average of the squared differences between actual and forecasted values.

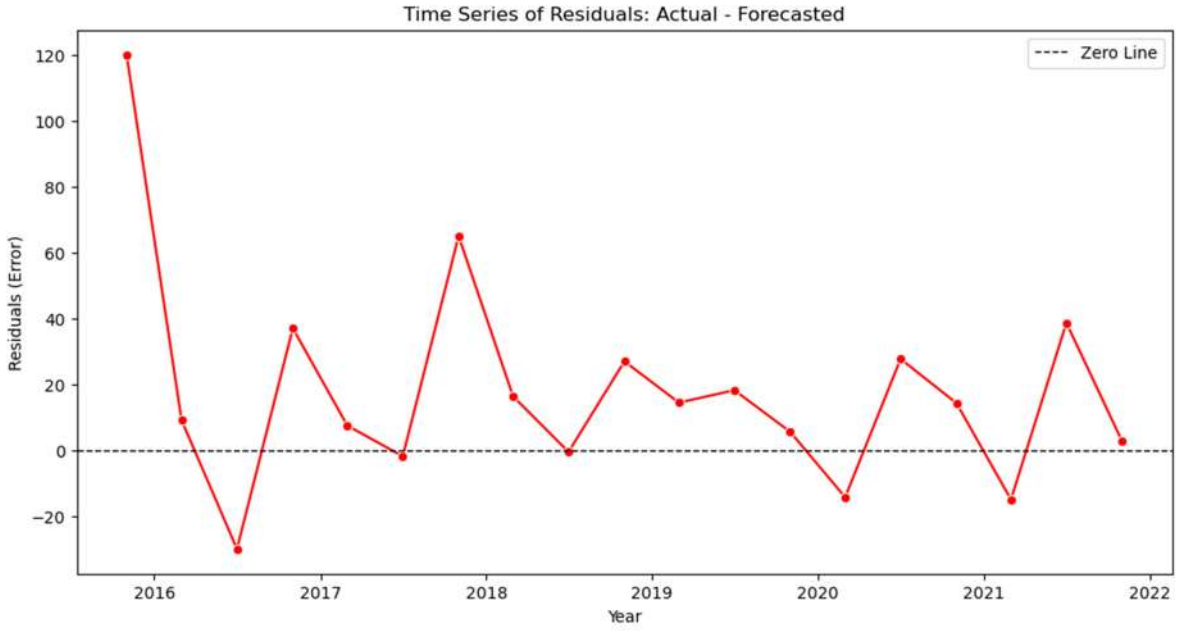
The obtained MAPE value of 2.83% indicates a minimal average percentage difference between predicted and actual values, while the RMSE value of 36.38 signifies the typical magnitude of errors. These metrics affirm the model's effectiveness in capturing underlying data patterns and providing accurate predictions.

Adequacy Check and Visualization:

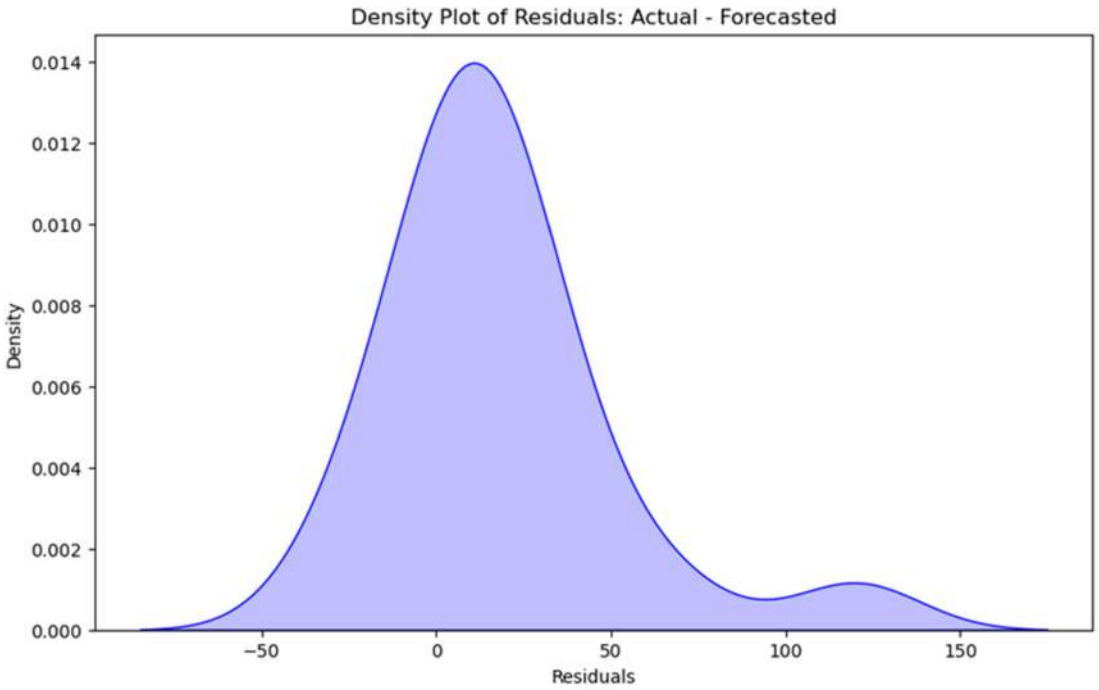
Q-QPlot



The adequacy of the model is checked by using a Q-Q plot. A Q-Q plot is a valuable visualization to assess the normality of the model residuals. The plot compares the quantiles of the residuals against the quantiles of a theoretical normal distribution. The Q-Q plot falls approximately along a straight line, it suggests that the residuals follow a normal distribution.

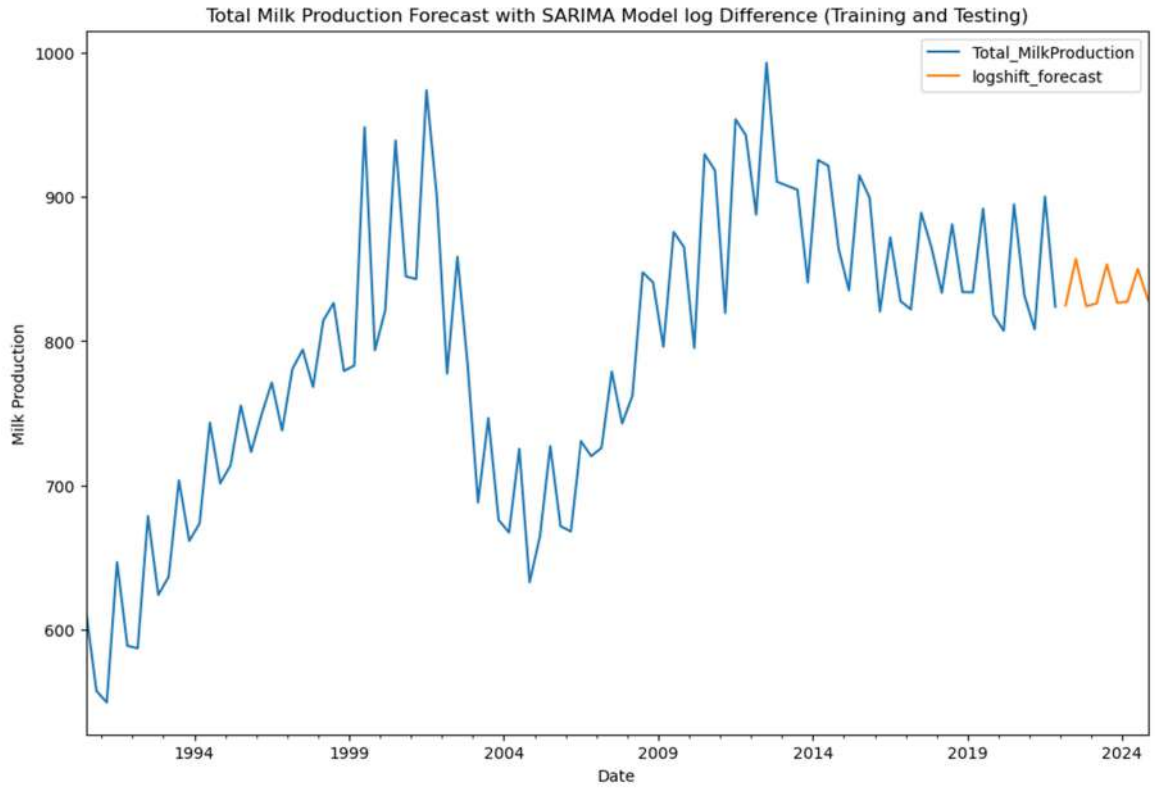


The line graph appears to show a mostly horizontal line around the zero line, with some variability. There are some data points above and below the zero line, but none deviate far from it. Overall, the image suggests that the residuals from the time series model being used are relatively small and evenly distributed around the zero line. This is a good indication that the model is fitting the data well and making accurate predictions.



The density plot below illustrates the distribution of residuals obtained from our predictive model. A density plot offers a visual representation of the probability density function of the residuals, enabling insights into their distributional characteristics

Future Prediction



Future Prediction using SARIMA	
Year	Future Prediction (in 000 tonnes)
Summer 2022	824.848021
Rainy 2022	857.078835
Winter 2022	824.231194
Summer 2023	826.217732
Rainy 2023	853.284715
Winter 2023	826.580567
Summer 2024	827.425327
Rainy 2024	850.137245
Winter 2024	828.449142

6.Tools and Libraries Used

- Pandas, Numpy- libraries used for data analysis and manipulation.
- Mathplotlib- libraries used for creating high-quality visualizations, including charts,
- plots, and graphs.
- Seaborn- data visualization library that is built on top of Matplotlib
- Statsmodels-library for statistical modeling and analysis
- Sklearn- library for machine learning
- Scipy-library for scientific computing and technical computing.
- Pmdarima- library for automatic time series forecasting with ARIMA models

7.Conclusion

In conclusion, the SARIMA predictive model effectively forecasts milk production in Kerala based on historical data. The model's accuracy, as indicated by low MAPE and RMSE values, suggests its reliability for future predictions. Additionally, the adequacy of the model was confirmed through a Q-Q plot analysis, which indicated that residuals follow a normal distribution.

However, it's essential to acknowledge certain limitations inherent in this study. Firstly, the predictive model's accuracy heavily relies on the quality and completeness of the dataset used for training. Incomplete or inaccurate data could lead to biased predictions and undermine the model's effectiveness. Secondly, while SARIMA is a robust method for time series forecasting, it may not capture all complex underlying factors influencing milk production, such as changes in environmental conditions, disease outbreaks, or policy interventions.

Furthermore, the model's performance may vary when extrapolating beyond the observed data range, especially in the face of unforeseen events or systemic shifts in the agricultural sector. Additionally, the SARIMA model assumes stationarity and may not adequately account for structural changes or long-term trends in milk production patterns.

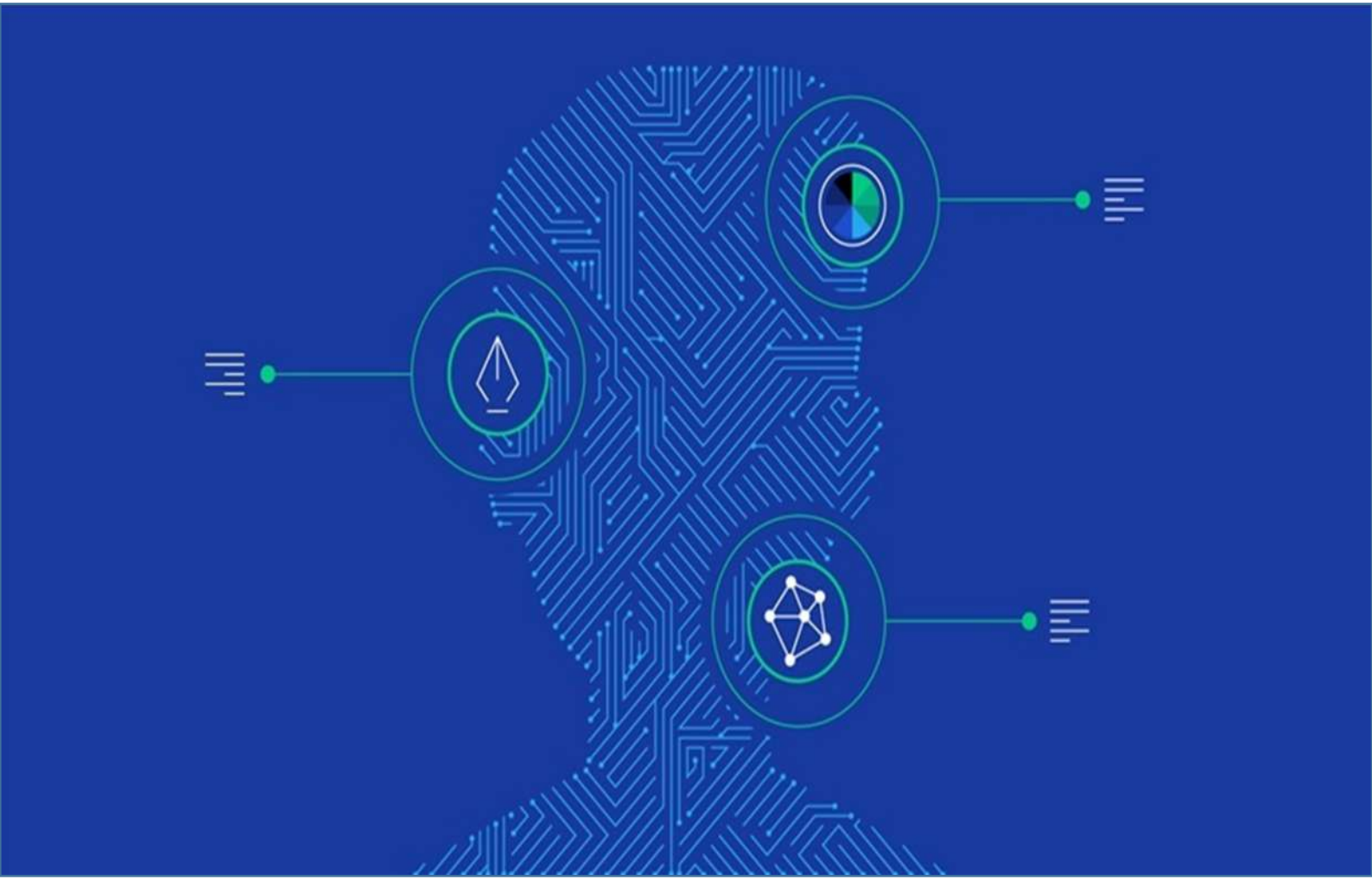
Despite these limitations, the project underscores the importance of leveraging advanced statistical methods for forecasting and planning within the animal husbandry

sector. Future research could explore integrating additional data sources and employing more sophisticated modeling.

References used

- *Integrated Sample Survey Report from 1990 to 2021 from Department of Animal Husbandry*
- *Compendium of Project Reports #AIFORALL Capacity Building in Artificial Intelligence & Data Analytics from Department of Economics & Statistics*
- *https://www.kaggle.com/code/sunaysawant/air-passengers-time-series-arima*

.



AI Model for Forecasting Farm Prices of Coconut in Kerala

Submitted by

Smt. Deepa S.A,

Deputy Director

1. Introduction

India is the largest coconut-producing country in the world and ranked third in the list of global coconut-producing nations. Although Kerala is known as 'the land of kera (coconut)', the present scenario with respect to production of coconut is not promising. In Kerala Interior places with fertile soils and plain regions also give good growth to coconut trees. During 2021-22, Kerala had the largest area under coconut (36%) followed by Karnataka (28%) and Tamil Nadu (21%). But during this period, Kerala was in the third position with respect to production (25%), with Karnataka in the first position (31%) and Tamil Nadu in second position (26%).

The agricultural sector plays a pivotal role in the economic landscape of Kerala, with coconut cultivation standing out as a cornerstone of the state's agrarian economy. As a major contributor to both rural livelihoods and the overall economy, the coconut industry's sustainable growth is essential for ensuring the well-being of countless farmers and stakeholders.

The price behaviour of coconut and its products has profound influence on the rural economy of Kerala. Large variation in the prices of coconut and coconut oil within a year is the major problem faced by farmers, consumers as well as planners. Farmers depending on income from coconut should get remunerative price throughout the year. Hence analysis of time series data of farm price of coconut is of prime importance.

2. Scope and Objectives

The scope of this project encompasses the development, validation, and application of ARIMA (Autoregressive Integrated Moving Average) and SARIMA (Seasonal ARIMA) models for forecasting the farm prices of coconuts in the state of Kerala. The project will focus on analysing historical price data, identifying temporal patterns, and incorporating seasonality to enhance the accuracy of predictions. The temporal nature of agricultural markets, influenced by factors such as climate conditions, demand fluctuations, and harvesting seasons, makes time-series analysis an apt approach for forecasting coconut prices.

The primary objectives of the project are:

- To develop accurate forecasting model to predict the temporal patterns in the farm price of coconut
- Generate timely and accurate forecasts for future coconut farm prices, aiding stakeholders in making informed decisions related to cultivation, trading, and policy formulation.
- Contribute to the reduction of financial risks for coconut farmers by offering insights into potential price fluctuations and enabling proactive risk management strategies.
- Provide a valuable tool for farmers, traders, and policymakers to make informed decisions related to planting, harvesting, pricing, and market interventions.

3. Literature Review

An article named 'Price Behaviour in India's Coconut Sector' by Dr. Prafulla K. Das. In this article, an attempt has been made to predict the prices of copra with the use of coconut oil prices; and the prices of coconut with the uses of prices of either copra or coconut oil in the same market or in different markets.

An article in the 'International Journal of Chemical Studies' named as 'Price behaviour of coconut in major markets of Kerala: A time series analysis' by Preethi VP, M.Sc. (Ag.) Student, Jessy Thomas K, Professor and Anil Kuruvila, Associate Professor, Department of Agricultural Economics, College of Horticulture, Thrissur, Kerala Agricultural University. In this journal they analysed the price behaviour of coconut in major markets of Kerala (Alappuzha & Kozhikode) for two periods and the variations in price to trend, cyclical, seasonal and irregular fluctuations were calculated.

A study named as 'Price forecasting models for coconut and coconut oil', has conducted by Indraji KN, Kerala Agriculture University, Thrissur. In this study aims to evaluate different time series forecast models for prices of coconut oil, copra and coconut and to suggest the suitable forecast models for all.

4. Data used for modelling and validation with data source

The dataset used in this project is the monthly average farm wholesale price of coconut (Rs / 100 numbers) in Kerala spanning from January 2000 to December 2020. The data was collected from various years of "Price Statistics", a publication of Department of Economics and Statistics. The department has been collecting the Farm price of major agricultural commodities produced in the State since 1954. The farm wholesale price is being collected fortnightly from 77 Taluk Centres in the State. District Average and State Average of Agricultural commodities is generated on monthly basis.

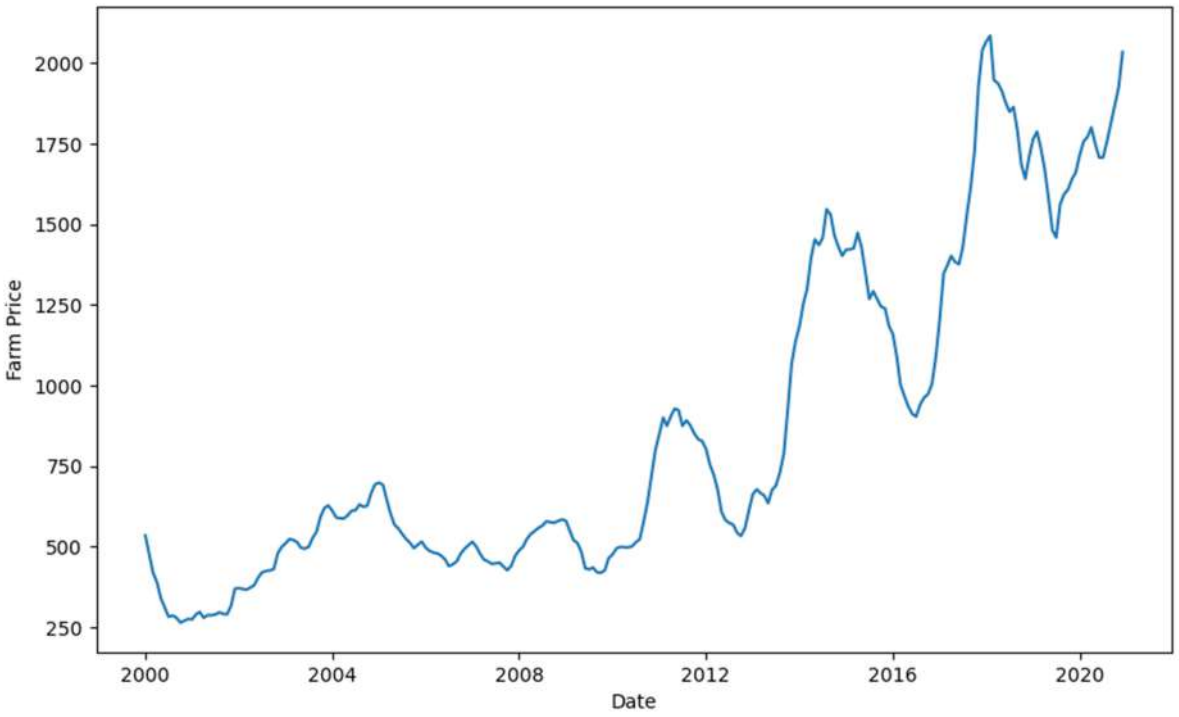
The dataset contains 252 rows and few rows are given below.

Sl No	Month	Farm price
0	2000-01	534.00
1	2000-02	474.00
2	2000-03	419.00
3	2000-04	389.00
4	2000-05	340.00
...	...	...
247	2020-08	1756.25
248	2020-09	1813.08

249	2020-10	1869.46
250	2020-11	1925.62
251	2020-12	2033.73

252 rows × 2 columns

Visualization of farm price over time 2000 to 2020



5. Methods and Methodology used

The time series values of the variable namely farm price of coconut in Kerala recorded at monthly interval is used in this study.

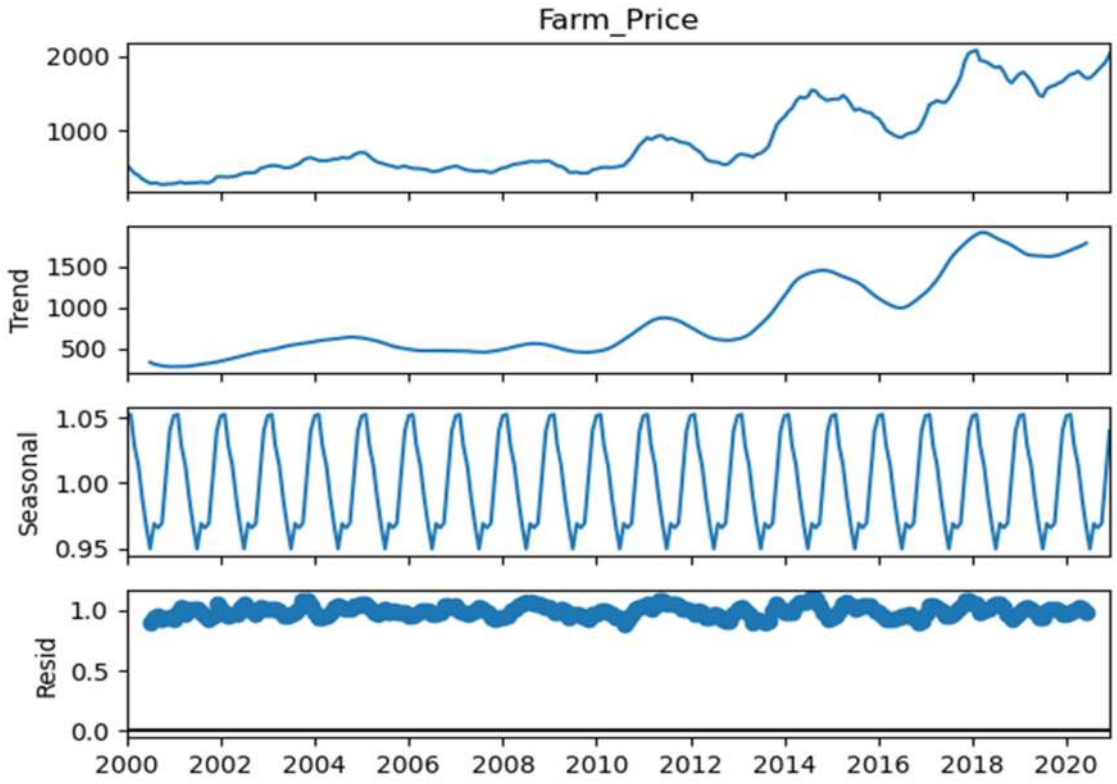
The forecasting of farm prices is a critical aspect of agricultural management, providing insights for both producers and stakeholders in navigating market dynamics. In this time series analysis, specifically the Auto Regressive Integrated Moving Average (ARIMA) and Seasonal ARIMA (SARIMA) models are used to predict farm prices of coconut in Kerala.

The general processes used for ARIMA models is the following:

- Gathered historical farm price data from various Price Statistics of Dept. of Economics & Statistics and compiled the dataset covering a substantial timeframe to capture temporal patterns and trends
- Smoothed outliers, and applied time series-specific techniques (differencing, logarithmic transformations) for stationarity.
- Utilized EDA techniques, including time series plots, ACF, and PACF, to identify temporal patterns and seasonality.

- Choose ARIMA and SARIMA models for their ability to capture temporal dependencies and seasonality.
- Fine-tuned model parameters through differencing and identification of AR and MA components.
- Split dataset into training and validation sets.
- Utilized metrics such as MAE for model accuracy assessment.
- Ensured models' robustness through rigorous validation processes.
- ARIMA and SARIMA models demonstrated effective forecasting capabilities.
- Validation metrics indicated reliable predictive accuracy.
- Use the fitted model to make future prediction of farm price.

The seasonal decompose function is used for time series decomposition, where a time series is decomposed into its trend, seasonal, and residual components that can help us understand the underlying behaviour of a time series and make more accurate predictions. By separating out the different components, we can better capture the trends, patterns, and seasonality in the data and reduce the impact of noise or error. The seasonal decomposition chart is given below.



On conducting component analysis of our time series, we can understand that our time series has an increasing trend, seasonality and noise. So, SARIMA model time series forecasting is suitable.

SARIMA (Seasonal ARIMA) Model

ARIMA (Auto Regressive Integrated Moving Average) is a popular time series modeling technique used for forecasting future values based on past observations and fitting errors. It is a combination of two models: the autoregressive (AR) model and the

moving average (MA) model. By addressing seasonal pattern, SARIMA extends ARIMA to handle seasonality.

The key steps involved in developing ARIMA & SARIMA models are given below:

1. Selection of differencing order to achieve stationarity

Stationarity is a desirable property for time series data because it simplifies the modelling process. A Stationary series is one whose statistical properties such as mean, variance, covariance, and standard deviation do not vary with time. Time series-specific preprocessing techniques, such as differencing and logarithmic transformations, were applied to stabilize variance and achieve stationarity. The order of differencing (d in ARIMA and D in SARIMA) represents the number of times differencing is needed to achieve stationarity.

Augmented Dickey-Fuller (ADF) test is a statistical test used to determine whether a unit root is present in a univariate time series dataset. The presence of a unit root indicates that the time series data is non-stationary. The null hypothesis of the test is that time series is non-stationery and alternative hypothesis is that the series is stationary. If the probability of the test statistic; p value <0.05(chosen significance level), null hypothesis can be rejected i.e.; time series will be stationery. In the case of our time series p-value = 0.998>0.05, indicates that time series is non-stationary.

After first differencing, the ADF test was conducted and the following result obtained:

ADF Test Statistic = - 6.503673674983843

p-value = 1.1452×10^{-8}

Which is less than 0.05 and hence the time series is stationary.

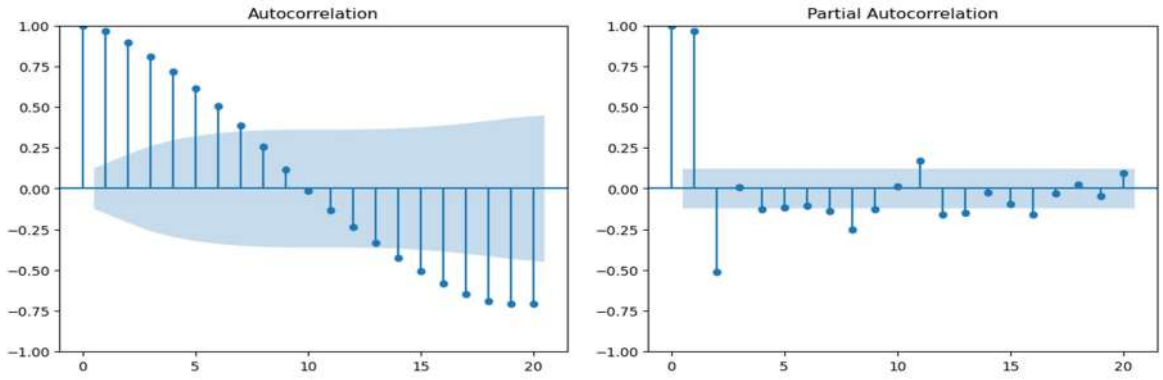
Here we had done one time differencing to make the data stationary. So, the value of parameter 'D' of the SARIMA model can be taken as 1.

2. Identification of AR and MA components to capture temporal dependencies

Autoregressive (AR) and Moving Average (MA) components capture dependencies between current and past observations. ACF (Auto Correlation Function) and PACF (Partial Auto Correlation Function) analyses are used to identify significant lags for inclusion in the model. ACF helps to identify AR component and PACF helps to identify MA component. SARIMA extends ARIMA to handle seasonality. It includes additional seasonal parameters (P, D, Q) to model seasonal dependencies. Seasonal orders (P, D, Q) represent the order of the seasonal AR, seasonal differencing, and seasonal MA components, respectively.

Exploratory Data Analysis (EDA) was conducted to gain insights into the temporal patterns and seasonality inherent in the coconut price data. Visualizations, such as time series plots, autocorrelation functions (ACF), and partial autocorrelation functions (PACF), were employed to identify potential patterns and inform the choice of model parameters 'P' & 'Q'.

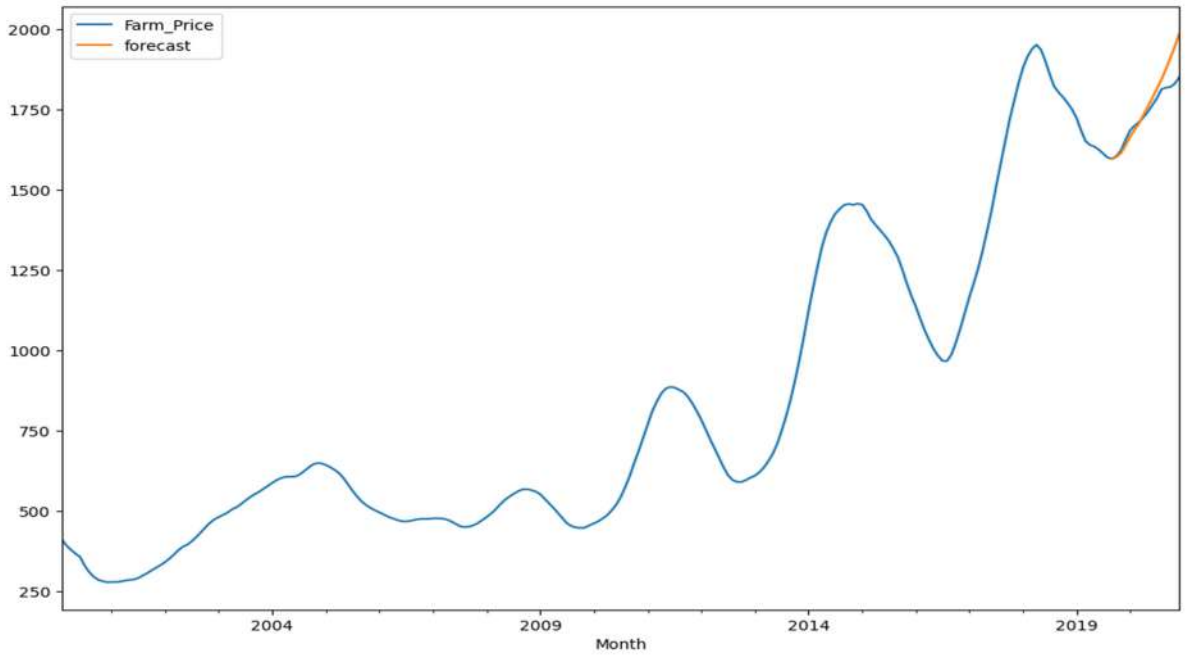
ACF and PACF plots for model order selection



The lag where PACF drops off can suggest the order of the AR component 'P' and the lag where ACF drops off can suggest the order of the MA component 'Q'. Use decomposition techniques to identify the periodicity of seasonality in the data. In this study, the season is a 'month'.

In this study, we get the best SARIMA model with seasonal order (P, D, Q, seasonal period) = (2, 1, 10, 12) and the fitted model is given below.

Plot for actual and predicted farm price



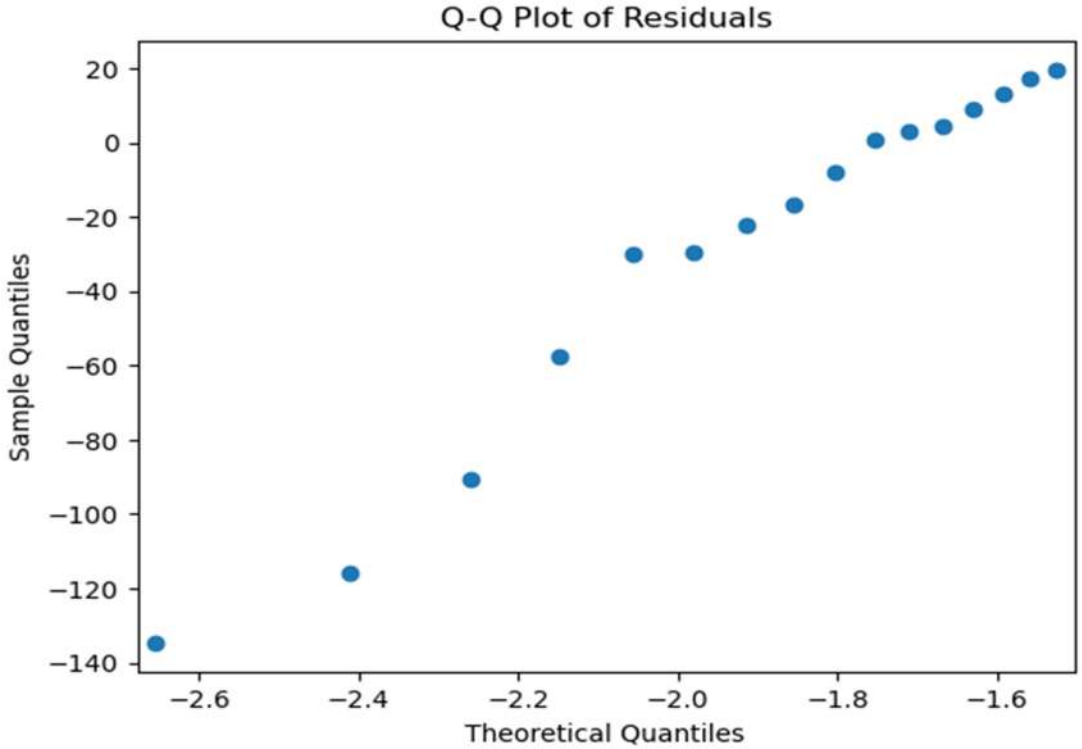
Once a Seasonal Auto Regressive Integrated Moving Average (SARIMA) model is fitted, it's crucial to assess its performance and validate its predictive capabilities. The accuracy of the model can be checked on test data by comparing the predicted values from the model to the actual values in the test set. MAPE (Mean Absolute Percentage Error) is a commonly used measure for evaluating the accuracy of a model. It measures the average percentage difference between the forecasted values and the actual values, and is expressed as a percentage. The formula for calculating MAPE is:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{|Actual_i - Forecasted_i|}{|Actual_i|}\ X\ 100$$

Where, n is the number of observations in the data set, $Actual_i$ is the actual value at time i and $Forecasted_i$ is the predicted value at time i. In our case MAPE is 1.98, that indicates good accuracy in your forecasting model.

A Root Mean Squared Error (RMSE) is another measure of the accuracy of SARIMA model. The RMSE represents the square root of the average of the squared differences between the actual and predicted values. Here RMSE= 53.99, which means the differences between the predicted and actual values are around 53.99 in the same unit as the actual data.

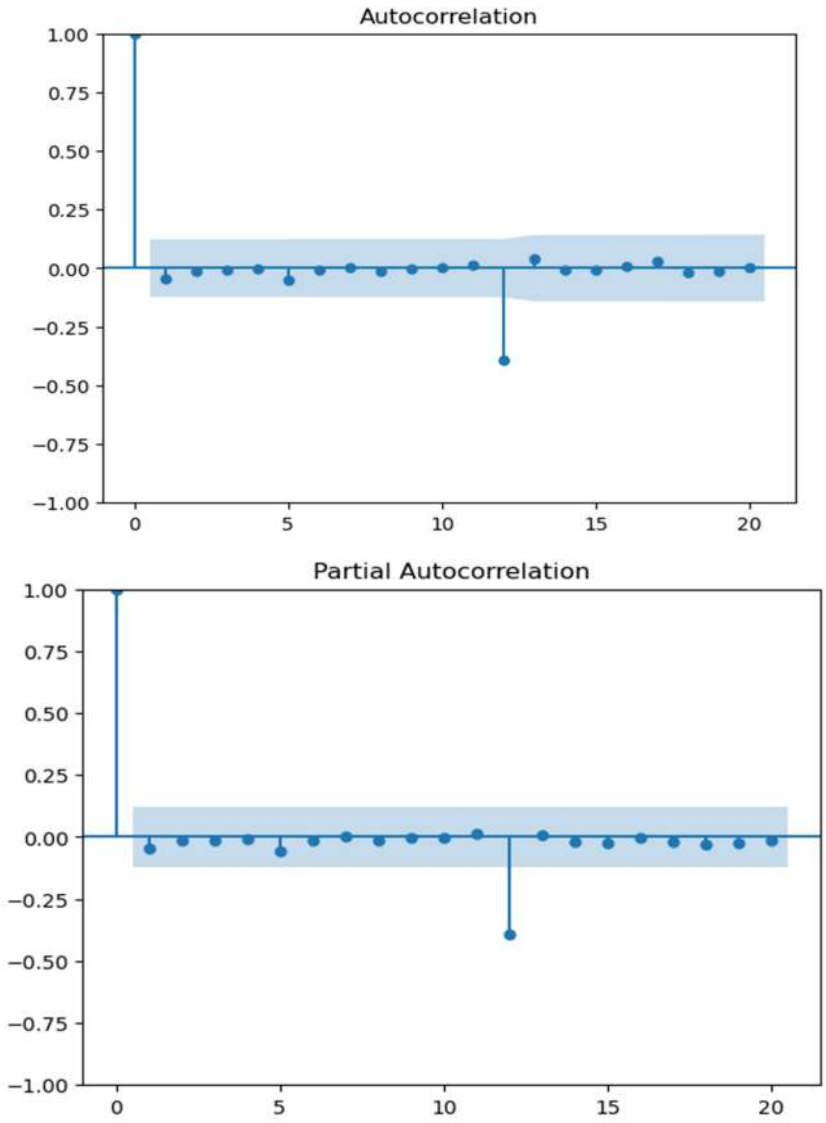
Finally, we check the adequacy of the model. Residual analysis is an important technique for evaluating the adequacy of time series forecasting models. Residuals are the differences between the actual values and predicted values. A quantile-quantile (Q-Q) plot is used to assess the normality of residuals. It's a valuable diagnostic tool for understanding the distributional characteristics of residuals and assessing the model's assumptions.



Here the points are approximately along with the diagonal line and hence, the residuals are approximately normally distributed.

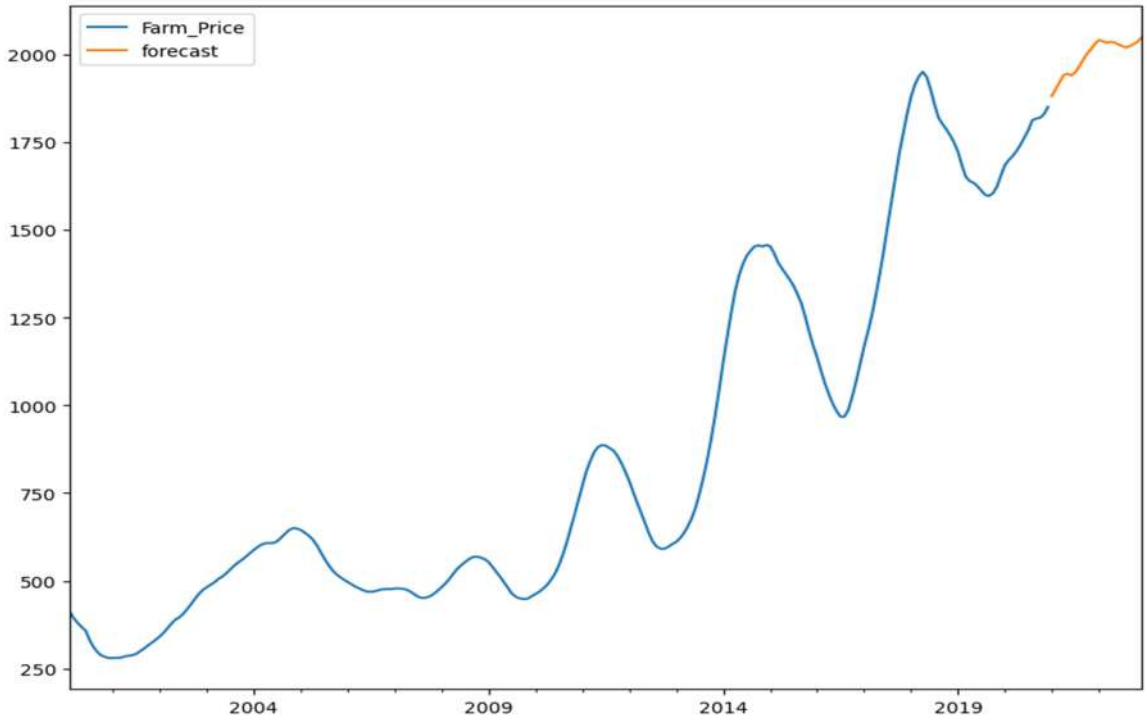
Autocorrelation and partial autocorrelation of residuals are important tools in the analysis of time series data, particularly in the context of modelling and forecasting using techniques like autoregressive integrated moving average (ARIMA) or seasonal ARIMA (SARIMA) models. To check for autocorrelation in residuals, the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots are used. Autocorrelation measures the correlation between the residuals at different time points and partial autocorrelation of residuals helps identify the direct relationship between the residuals at two different time points, excluding the influence of other lags.

ACF and PACF plots of residuals



From the ACF plot it is clear that correlation coefficients between residuals at any lag except 0 and 12 are inside the blue band (indicating a confidence interval) and hence statistically insignificant, means there is no significant auto correlation between residuals. So, we can conclude that, this time series model adequately captures the temporal dependencies in the data. Hence, we can make predictions using this model. Using the model, we got the forecasted the monthly farm price of coconut for next one year as;

SARIMA future data forecast



The future data predicted for 12 months of 2021 is given below.

Month	Predicted Farm Price (in Rs)
Jan -2021	1882.48
Feb – 2021	1902.81
Mar – 2021	1922.39
Apr – 2021	1941.56
May – 2021	1945.46
June – 2021	1940.85
July – 2021	1950.12
Aug – 2021	1966.37
Sep – 2021	1985.70
Oct – 2021	2002.67
Nov – 2021	2015.54
Dec - 2021	2030.75

6. Packages and libraries used

The libraries and tools used in this project are given below.

- Pandas (pd): library used for data manipulation and analysis. This library provides data structures like Data Frame and Series, and functions for data cleaning, filtering, and analysis.
- Numpy (np): library used for numerical computing in python. It offers support for large, multi-dimensional arrays and matrices, along with mathematical functions.
- Matplotlib.pyplot (plt): A plotting library used for creating static, animated and interactive visualizations. Commonly used for creating charts and plots.
- Statsmodels: the library used for statistical modelling and analysis.
- Seaborn (sns): is a data visualization library based on Matplotlib, and it provides a high-level interface for creating attractive and informative statistical graphics.
- Sklearn.metrics: This module in scikit-learn (sklearn) provides a variety of metrics for evaluating the performance of machine learning models. These metrics cover different aspects of model performance, depending on the type of task.
- Scipy: is a scientific library in Python that builds on the capabilities of NumPy and provides additional functionality for scientific computing.

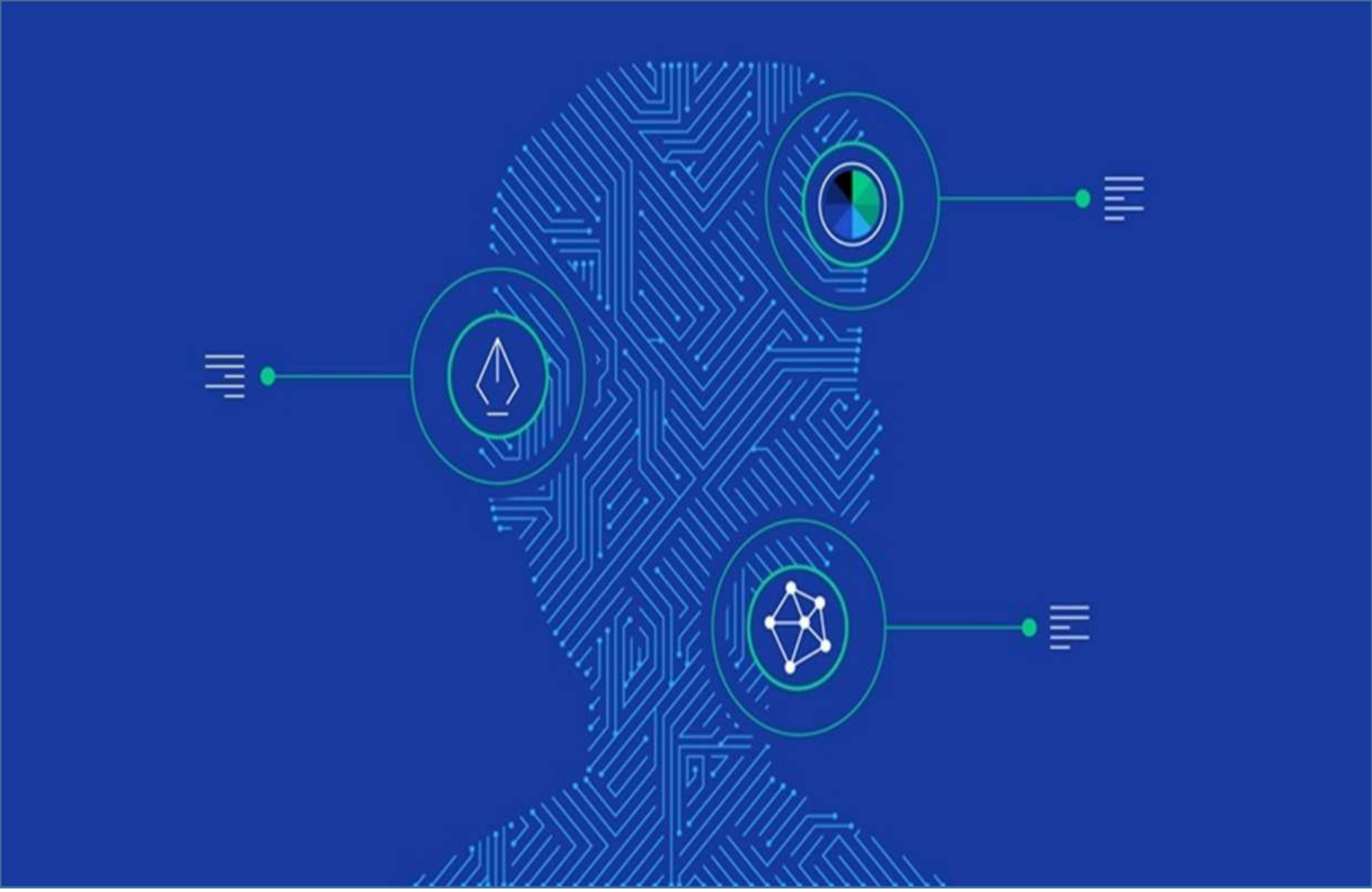
7. Final output of the project

This study was undertaken to analyses the seasonal variations in the farm price of coconut in Kerala and to develop a model that predicts the future farm prices of coconut. Here SARIMA (2, 1, 10, 12) is the best model for this data and that predicts the monthly average farm price of coconut in Kerala for the next 1 year. The values seem to show an increasing trend over the months, with some fluctuations. This study helped to forecast the future price of coconut based on historical data at reasonable accuracy (MAPE=1.98). The results indicated that the SARIMA based farm price prediction model has proven to be a valuable tool for forecasting agricultural prices in Kerala.

The forecasted prices align closely with observed values, indicating that the model has successfully learned from historical data and is capable of making reliable predictions. While the model has shown satisfactory performance, continuous refinement and updates may further enhance its accuracy. Future iterations could consider incorporating additional factors that might influence farm prices. Further research and iterative improvements will contribute to the sustained effectiveness of such forecasting models.

8. References used

i. Price Statistics from 2000 to 2020 from Department of Economics & Statistics
ii. Compendium of Project Reports #AIFORALL Capacity Building in Artificial Intelligence & Data Analytics from Department of Economics & Statistics
iii. Official website of Department of Economics & Statistics
iv. Agriculture Statistics at a glance 2022 from Ministry of Agriculture and Farmers' Welfare, Govt. of India.



Forecasting Paddy Yield Using Sample Check Data

Submitted by

Kum. Lakshmi S,

Statistical Assistant Grade II

Abstract

This study utilizes sample check survey data on paddy cultivation to develop an Artificial Neural Network (ANN) model focused on predicting yield weight. The dataset includes various variables such as district, manuring status, irrigation, fertilizer quantities (N, P, K), seed variety, and yield weight. The primary objective is to leverage this data to construct an ANN model that accurately predicts paddy yield weight. The model aims to offer valuable insights into the factors influencing yield, contributing to the advancement of predictive models in agricultural contexts.

Introduction.

The program for sample check on area enumeration involves the identification and location of four clusters within each selected Investigator zone. Differences in crop reporting and recording across survey numbers during various seasons are pinpointed by supervisory officers from NSO or DES through comprehensive analysis.

Concurrently, the inspection of crop cutting experiments during the harvest stage entails evaluating the adherence of investigators to prescribed General Crop Estimation Survey procedures. This assessment encompasses multiple facets, including the selection of fields and random coordinates, marking of experimental plots with specified dimensions, harvesting procedures, weighing of the produce, among others. Additionally, the examination encompasses scrutinizing the provision and utilization of equipment for these experiments, the training received by primary workers, crop conditions, input usage, and related factors.

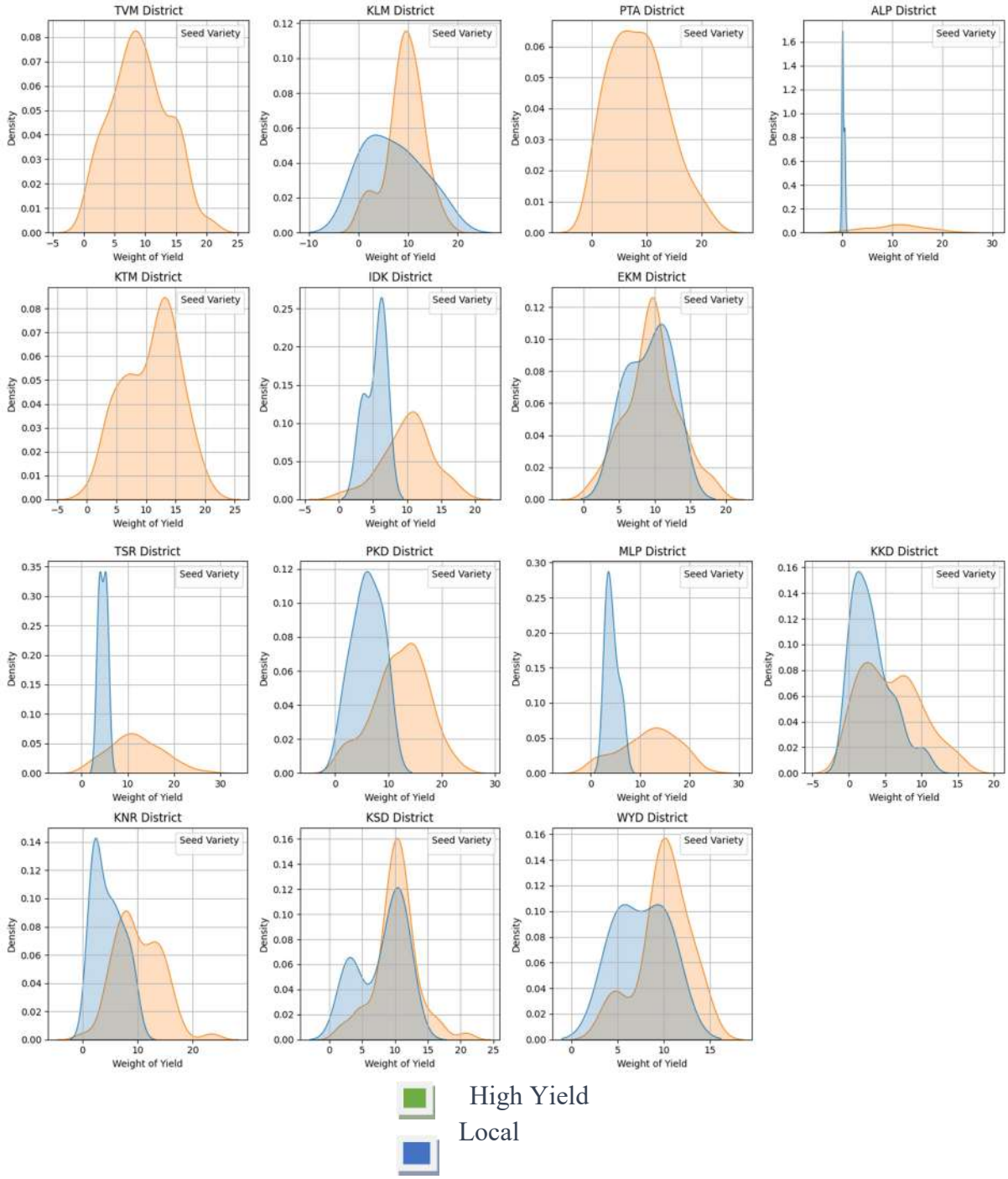
Utilizing the data acquired from this survey, I've gathered comprehensive information on paddy cultivation, encompassing district-specific details, seed varieties, irrigation methods, manure application status, and specific quantities of N, P, K fertilizers used. Additionally, the dataset includes the recorded weights of the paddy yield.

Machine learning models, known for their versatility and predictive capabilities, find extensive applications across various industries. These models, designed to analyze patterns and make predictions from data, have a significant potential for application within the agricultural sector. For instance, in agriculture, these models can be employed to predict crucial outcomes such as crop yield. By leveraging data encompassing diverse variables like district-specific details, seed varieties, irrigation methods, manure application status, and specific quantities of fertilizers (N, P, K), machine learning models can effectively forecast paddy yield weight. This predictive ability enables farmers and agricultural experts to anticipate yield trends, optimize resource allocation, and make informed decisions to enhance crop productivity. Moreover, machine learning models can aid in disease prediction in crops, optimize irrigation schedules, suggest suitable crop varieties based on environmental conditions, and even predict market demand, thereby assisting farmers in strategic planning and maximizing agricultural output. The utilization of machine learning models in agriculture presents a promising prospect for revolutionizing farming practices, improving crop yields, and ensuring global food security by enabling data-driven decision-making in this vital sector.

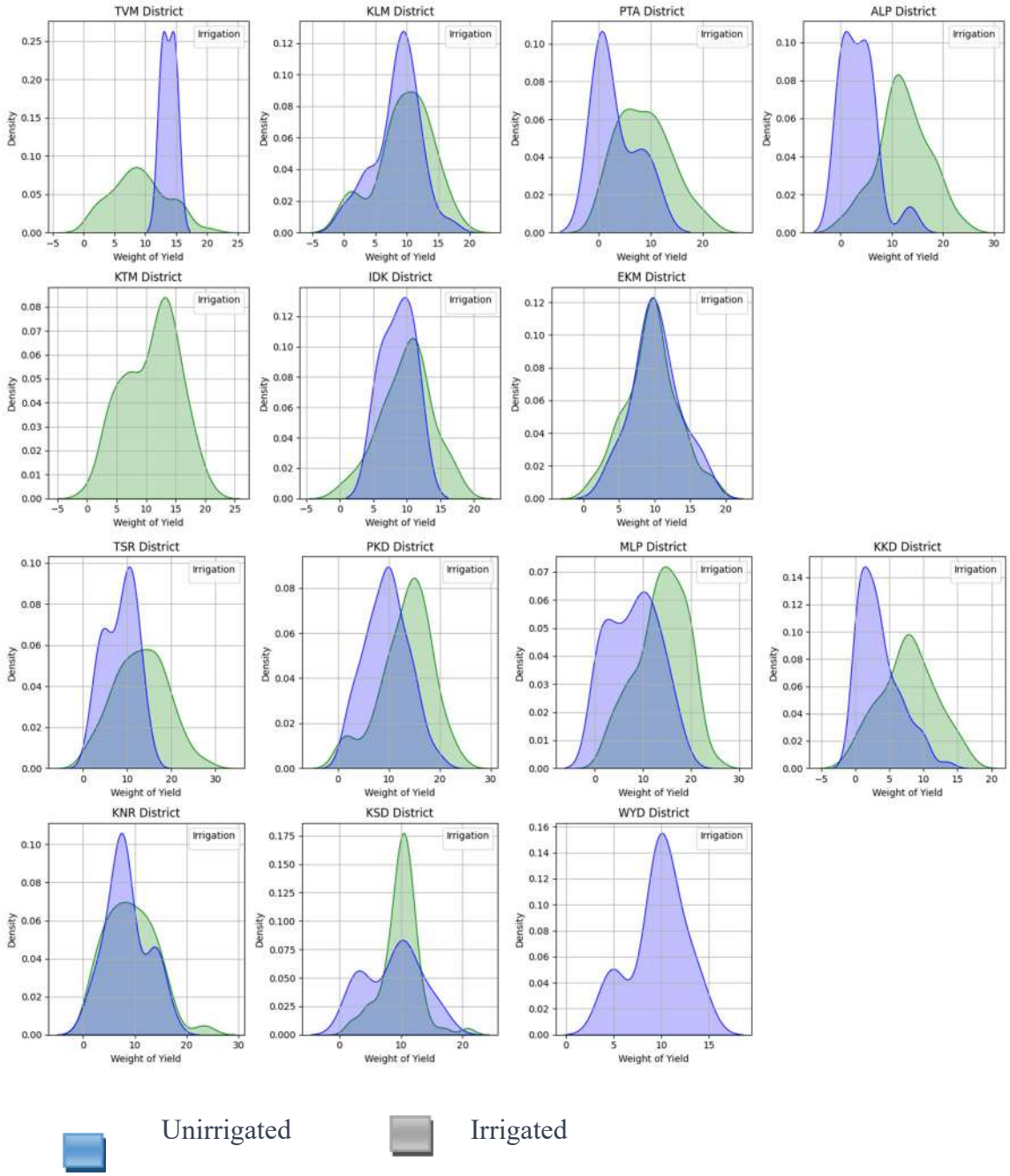
Objective

To construct an Artificial Neural Network (ANN) model using paddy sample data, comprising district, whether manure was applied or not, irrigation status, fertilizer quantities (N, P, K), seed variety, and yield weight. The objective is to accurately predict the yield weight based on these variables.

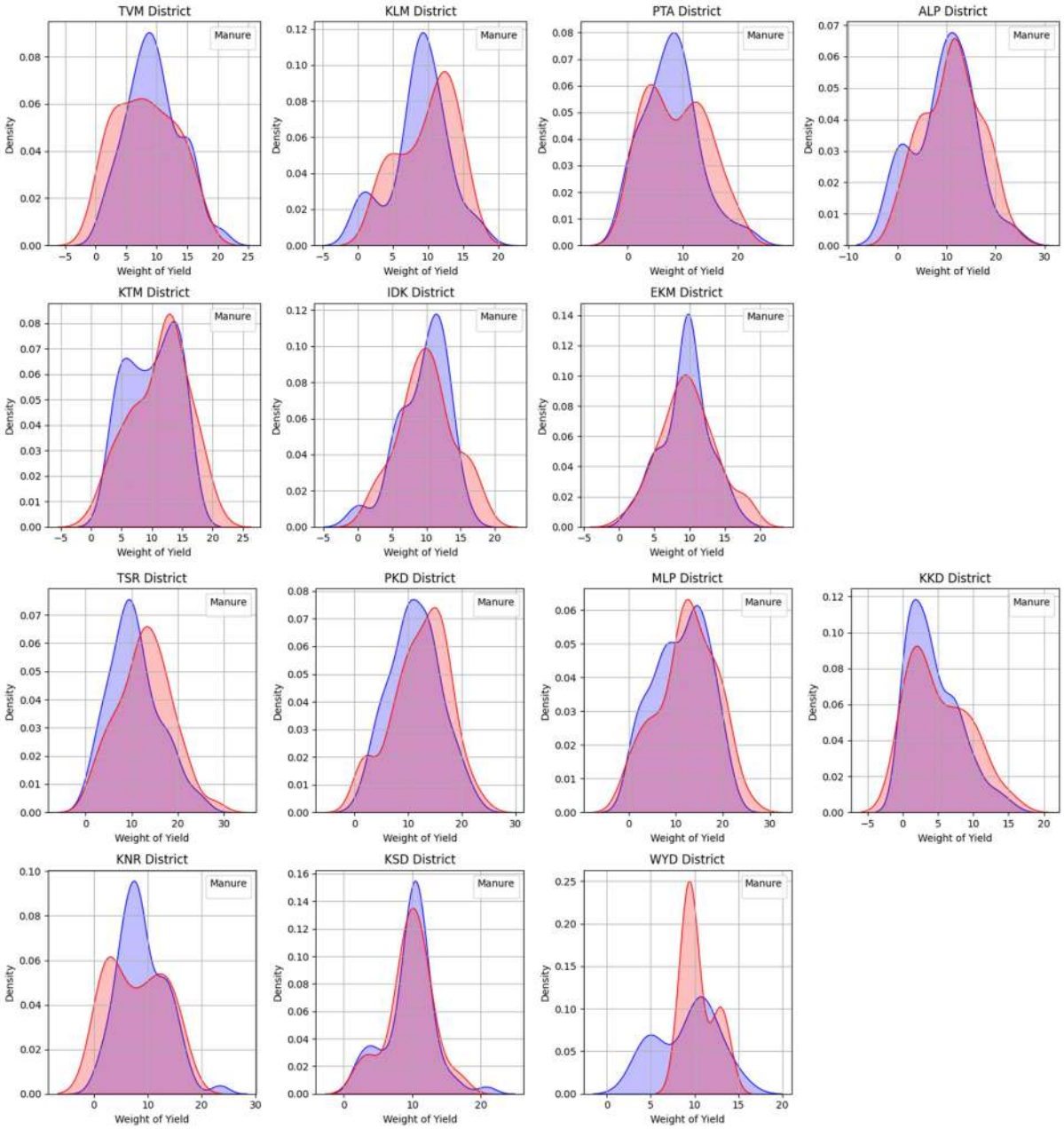
Distribution of weight of yield according to Seed Variety.



Distribution of weight of yield according to Irrigation



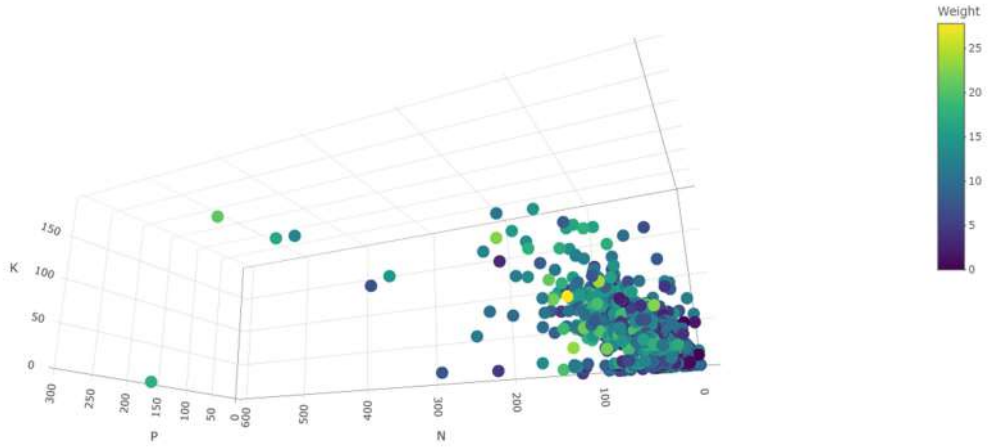
Distribution of Weight of Yield according to Manure



Not manured

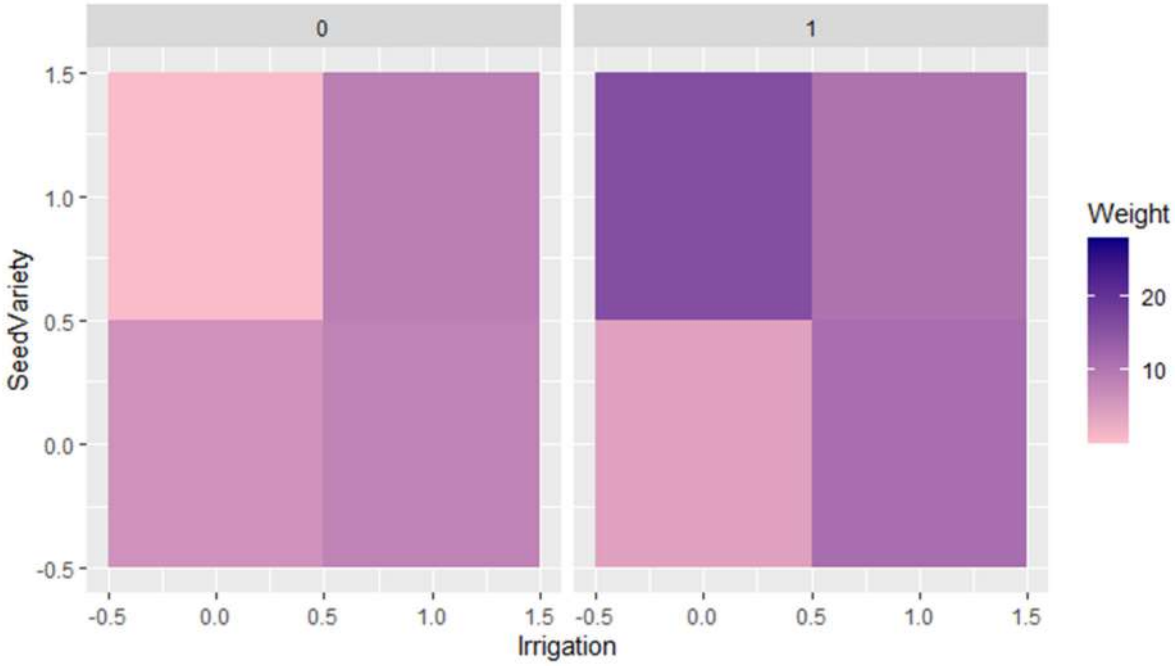
Manured

Visualization of Fertilizer composition impact on Paddy Yield



Maximum weight was obtained for **N** :117, **P**:60, **K**:74

Combined influence of Manure, Irrigation and Seed variety on Paddy yield weight



Methodology

1. Data Collection and Preprocessing

- Data Sources

The data used in this project was gathered from various zones across 14 districts within Kerala state. It comprised 1198 observations, encompassing diverse parameters relevant to paddy yield prediction.

- Data Preprocessing

Conversion of Categorical Data: District names, a categorical variable, were transformed into numerical representations (1-14) to enable their use in predictive modeling.

Yield Weight Classification: The weight of paddy yield was classified into six distinct categories (1-5 kg, 5-10 kg, 10-15 kg, 15-20 kg, 20-25 kg, and 25-30 kg). This categorization enabled the modelling of yield ranges.

2. Model Development using Artificial Neural Networks (ANN)

- Feature Engineering

Normalization/Scaling: Input features underwent normalization or scaling to ensure uniformity in their ranges and prevent bias during model training.

- Categorical Variable Encoding

Techniques such as one-hot encoding or label encoding were employed to handle categorical variables (e.g., irrigation, manure status, seed variety) for compatibility with the neural network.

- Model Architecture

Neural Network Configuration: An Artificial Neural Network (ANN) architecture was constructed, comprising layers:

Input Layer: Configured to accommodate the number of input features.
Hidden Layers: Comprised 64 and 32 neurons respectively, employing 'relu' activation functions.
Output Layer: Consisted of 7 neurons employing 'softmax' activation for multi-class classification, predicting yield range categories.

- Model Training and Evaluation

Data Splitting: The dataset was divided into training (80%) and testing (20%) sets for training and validating the model's performance.

Model Compilation: The ANN was compiled using 'categorical_crossentropy' as the loss function and the 'adam' optimizer.

Training Process: The model was trained over 64 epochs with a batch size of 32, and a 10% validation split was used to monitor and mitigate overfitting during training.

Model Accuracy: The trained model achieved an accuracy of 87% on the test dataset, indicating its ability to predict paddy yield ranges effectively.

- Model Improvement Strategies To enhance the model's performance:

Hyperparameter Tuning: Experimentation was conducted with different epochs, batch sizes, and learning rates.
Regularization Techniques: Techniques like dropout were explored to prevent overfitting.
Feature Engineering: Additional feature engineering and selection methods were considered to improve the model's predictive power.

3. Result Analysis and Conclusion

The methodology section concludes with an analysis of the model's predictions and performance metrics, such as accuracy, precision, recall, and F1-score. The iterative nature of refining the model based on dataset understanding and achieving an accuracy of 87% was highlighted. Additionally, potential avenues for future enhancements or alternative modelling approaches were proposed.

Prediction of Unseen Data

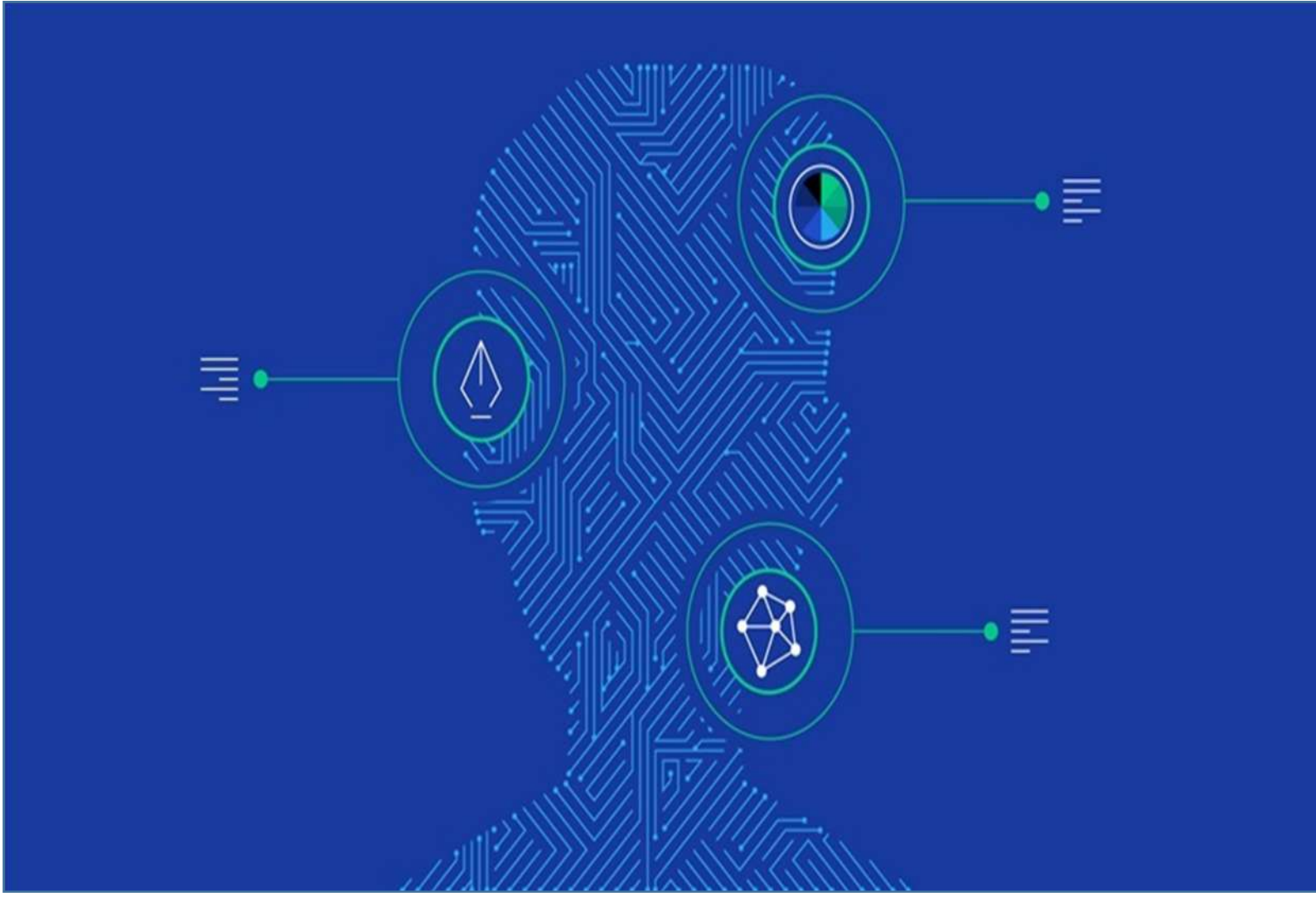
Based on the model's prediction, for the given conditions in the Autumn season in Pathanamthitta district, employing irrigation, using N:P:K fertilizers in the proportion of 15,20,15 respectively, without manure, and planting local seed variety, the model predicts a yield between 5 and 10 kilograms of paddy.
This incorporation of a specific prediction for an unseen scenario not present in the training dataset demonstrates the practicality and usefulness of the model in making predictions for real-world agricultural conditions.

Reference

Journal Articles:
1.Thomas, R., & Bhat, R. (2018). Predictive Modeling of Crop Yield using Machine Learning Techniques. International Journal of Computer Applications, 180(15), 32-39.

2.Das, S., Mishra, P., & Mohanty, S. P. (2020). Impact of Irrigation and Fertilizer Application on Crop Yield: A Case Study in Eastern India. Journal of Agricultural Science and Technology, 22(5), 1101-1115.

Online Sources:
1.United States Department of Agriculture (USDA): National Agricultural Statistics Service. (n.d.). https://www.nass.usda.gov/
2.Agricultural Research Service (ARS): United States Department of Agriculture. (n.d.). https://www.ars.usda.gov/
3.International Food Policy Research Institute (IFPRI).
https://www.ifpri.org/
4.Global Open Data for Agriculture and Nutrition (GODAN). https://www.godan.info/
5.Food and Agriculture Organization (FAO) Statistical Databases. http://www.fao.org/faostat/en/#data
6.AgriTech Hub: Agricultural Technology Resources. https://www.agritechhub.com/



Analysis and Forecasting of Tapioca Production in Kerala

Submitted by

Kum. Reshmi S,

Statistical Assistant Grade II

Introduction

In the lush landscapes of Kerala, India, where agriculture plays a pivotal role in the economy, tapioca production stands as a significant contributor to the agricultural tapestry. The subtropical climate and fertile soil of Kerala provide an ideal environment for cultivating tapioca, a starchy root crop widely utilized in various culinary traditions. This analysis delves into forecasting tapioca production, leveraging ARIMA modelling techniques. Recognizing the importance of accurate production forecasts for informed agricultural planning, the study aims to unravel patterns and trends in historical data. By examining the intricacies of tapioca cultivation in Kerala, this analysis not only seeks to enhance predictive modelling but also to offer valuable insights into the factors influencing the region's tapioca production. The unique agro-ecological conditions of Kerala, coupled with the prominence of tapioca in its agricultural landscape, make this study particularly relevant for stakeholders in the region, providing a foundation for informed decision-making and sustainable agricultural practices.

Objectives

The primary objective of this report is to employ ARIMA modeling techniques for forecasting tapioca production in Kerala, India. By leveraging time series analysis, the study aims to discern patterns, trends, and potential influencing factors in historical data related to tapioca cultivation in the region. The specific objectives include identifying optimal orders (p, d, and q) for the ARIMA model, generating accurate forecasts, and evaluating the performance of the forecasting model. Through this analysis, the report strives to provide valuable insights into the temporal dynamics of tapioca production in Kerala, facilitating informed decision-making for agricultural planning and resource management.

Datasets Used

To conduct this analysis of tapioca production in Kerala, i used a comprehensive dataset that contains relevant informations about area of cultivation of tapioca, annual tapioca production, annual rainfall of 37 years from 1985 to 2022.

Data Source:

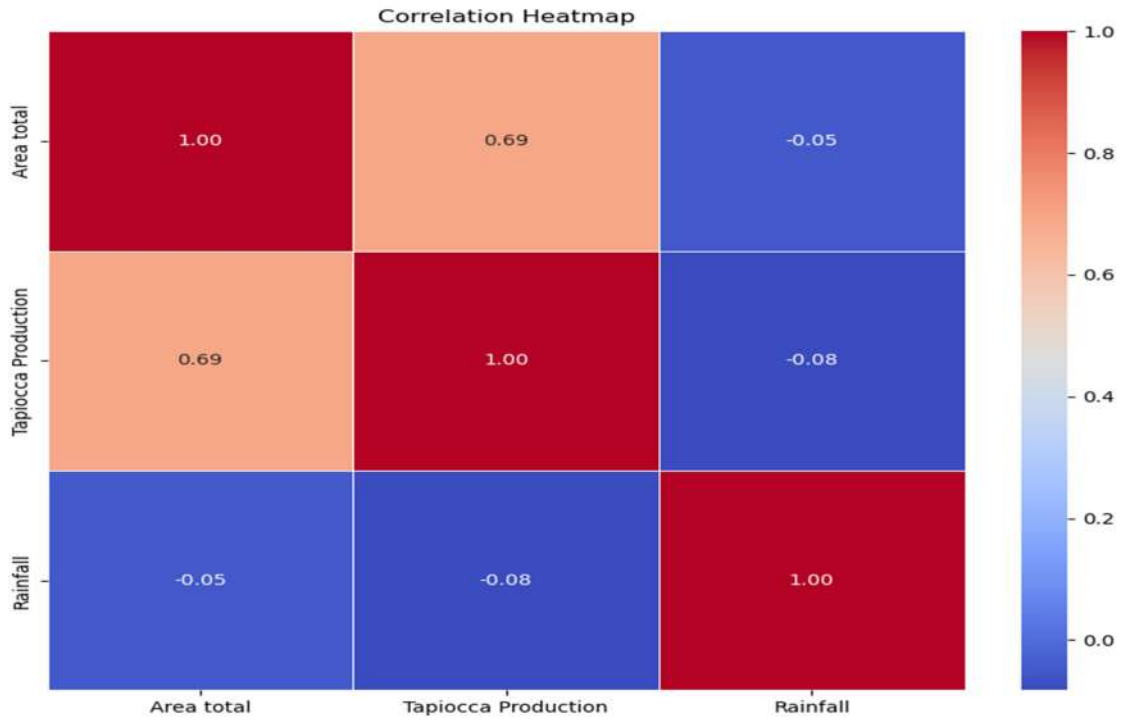
Dataset is collected from our publication – "Agricultural Statistics, Department of Economics and Statistics, Govt. Of Kerala".

Methods used for analyse the relationship between the factors such as area and rainfall.

Tools and Libraries used

Python is used for the analyse the correlation and time series analysis of the data. For this python and google collab notebook is used. The data analysis is done using python libraries such as Matplotlib, Pandas, Numpy etc.The ARIMA model for prediction is done using R-Programming. Statsmodels libraries are used.

Visualisation of the correlation using Heat Map

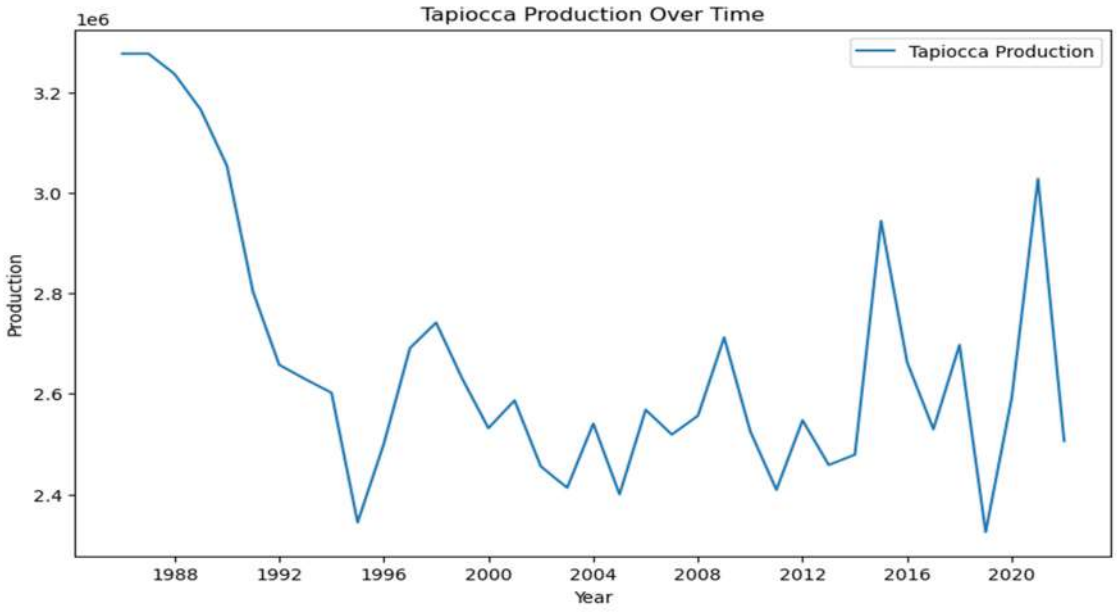


Correlation Matrix

	Area total	Tapioca Production	Rainfall
Area total	1.000000	0.688104	-0.046007
Tapioca Production	0.688104	1.000000	-0.082406
Rainfall	-0.046007	-0.082406	1.000000

Area total has positive relation. Rainfall shows negative relation.

Graphical representation of Tapioca production over time

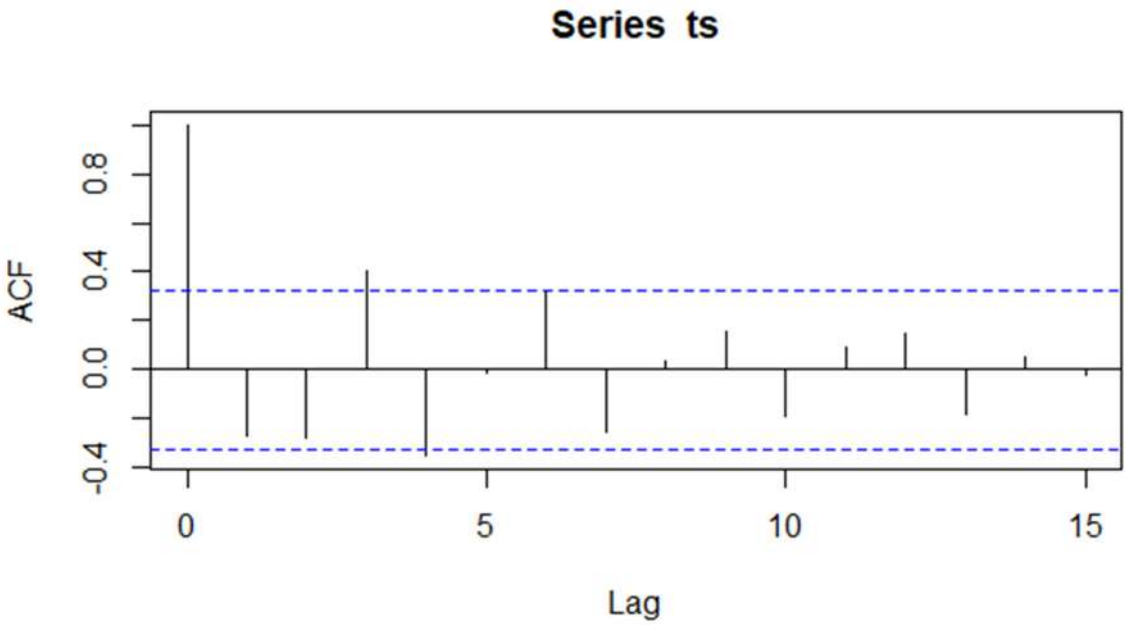


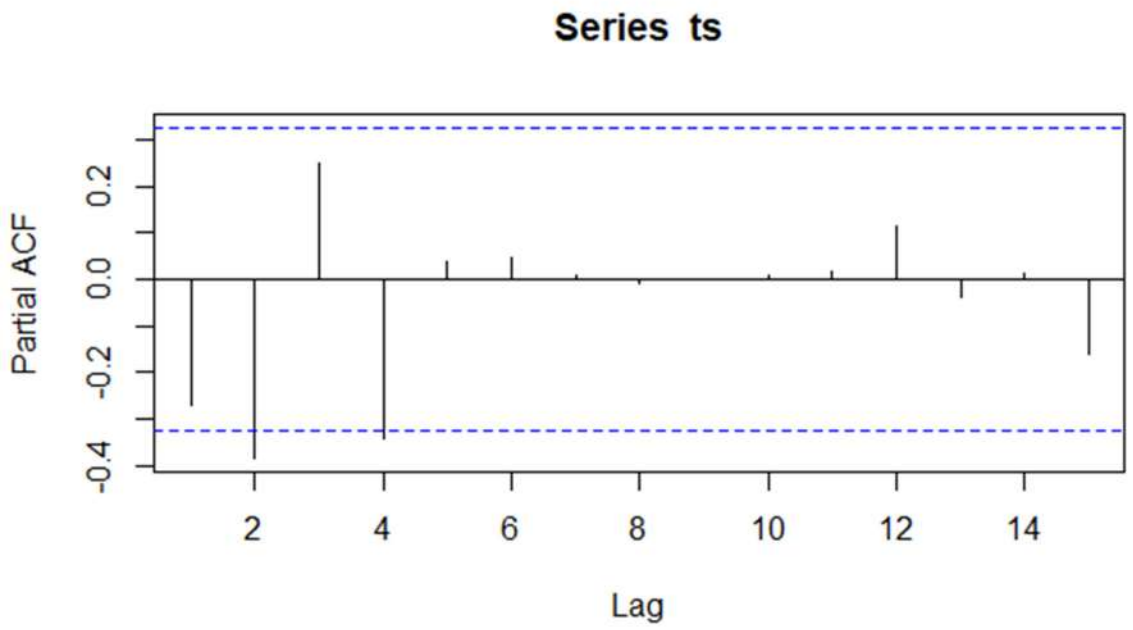
Auto Regressive Integrated Moving Average [ARIMA Model]

An ARIMA is a statistical analysis model that uses time series data to either better understand the dataset to predict the future trends. It predicts the future values based on past values.

Autocorrelation Functions (ACF) and Partial Autocorrelation Functions (PACF) plots are useful tools in time series analysis to identify the autocorrelation structure of the data. From these graph we can understand the relationship between observations at different lags.

ACF and PACF





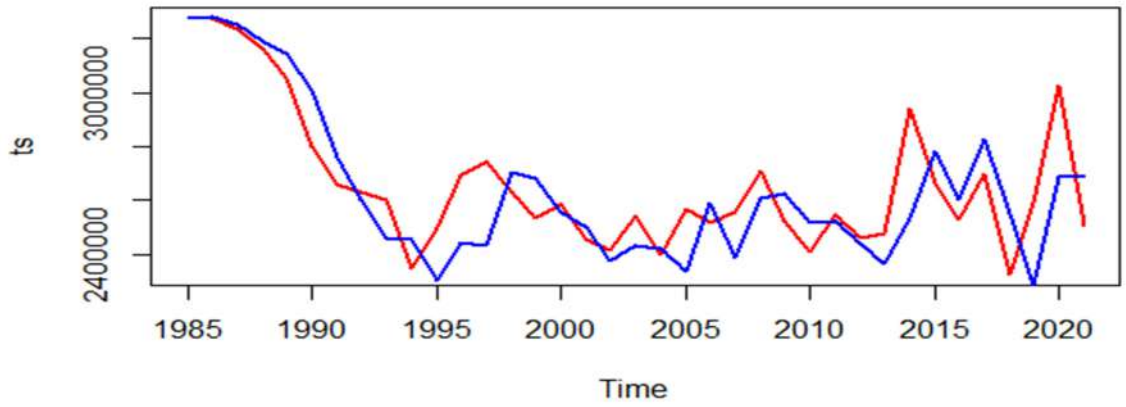
gmented Dickey-Fuller Test

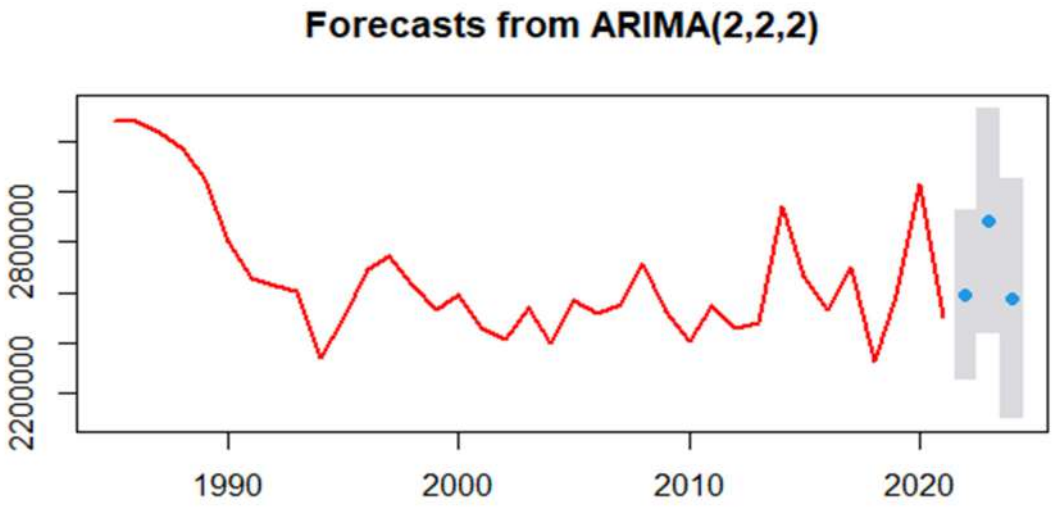
Dickey-Fuller = -4.6098
Lag order = 3
p-value = 0.01
Alternative hypothesis: stationary

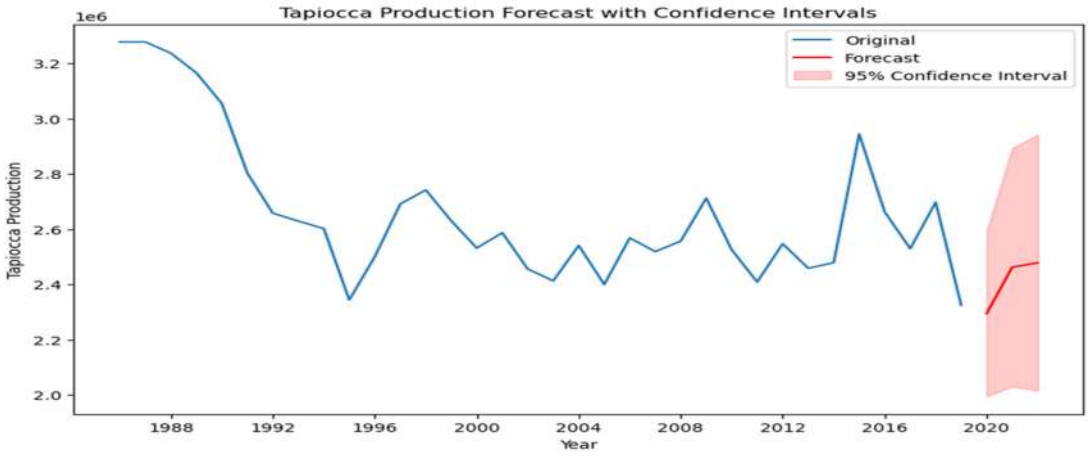
Considering different models using Auto ARIMA

ARIMA(2,2,2) : 951.7914
ARIMA(0,2,0) : 986.9408
ARIMA(1,2,0) : 981.0933
ARIMA(0,2,1) : Inf
ARIMA(1,2,2) : Inf
ARIMA(2,2,1) : 957.0147
ARIMA(3,2,2) : 952.5215
ARIMA(2,2,3) : Inf
ARIMA(1,2,1) : 961.7851
ARIMA(1,2,3) : 955.9423
ARIMA(3,2,1) : 954.6358
ARIMA(3,2,3) Inf

Best model: ARIMA(2,2,2)


ts
3000000
2400000
1985
1990
1995
2000
2005
2010
2015
2020
Time


Forecasts from ARIMA(2,2,2)
2800000
2200000
1990
2000
2010
2020


Tapiocca Production Forecast with Confidence Intervals
1e6
Original
Forecast
95% Confidence Interval
Tapiocca Production
3.2
3.0
2.8
2.6
2.4
2.2
2.0
1988
1992
1996
2000
2004
2008
2012
2016
2020
Year

Result

Forecasted production for next 3 years(2023-2025)

	Point Forecast	Lo 95	Hi 95
2022	2590419	2257713	2923125
2023	2882707	2437173	3328241
2024	2578168	2102395	3053941

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	25396.12	155375.2	116605.2	0.8623782	4.412618	0.7440292	0.022996

Conclusion

In conclusion, this project aimed to enhance our understanding of tapioca production in Kerala through the application of ARIMA modeling techniques. By leveraging time series analysis on historical data, we successfully identified optimal orders (p, d, q) for the ARIMA model, enabling us to generate accurate forecasts for tapioca production. The results, validated through key metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared, provided insights into the temporal dynamics of tapioca cultivation in the region.

The analysis revealed patterns and trends, shedding light on the factors influencing tapioca production in Kerala. The unique agro-ecological conditions, climate variations, and potentially other contributing variables were considered in interpreting the forecasting results. While acknowledging the challenges associated with forecasting agricultural commodities, the findings of this study contribute to the ongoing efforts in sustainable agricultural planning and resource management in Kerala.

This project not only deepens our knowledge of tapioca production but also underscores the importance of accurate forecasting for informed decision-making in agriculture. The insights gained can guide policymakers, farmers, and other stakeholders in optimizing resource allocation, improving crop yield predictions, and fostering resilience in the face of changing agricultural landscapes. Looking ahead, the lessons learned from this study can serve as a foundation for further research and refinement of forecasting models.

Consumer Price Index (R/U/C)
A Machine Learning Approach to Cluster the Markets

Submitted by

Sri. Sajin Gopi,

Assistant Director

1. Introduction

The Consumer Price Index (CPI) stands out as a key and widely utilized economic indicator for assessing the cost of living for specific segments of the population at a given point in time. As implied by its name, the CPI is intricately linked to the price fluctuations of commodities and the purchasing behaviour of the population, with these aspects often serving as reflections of other economic parameters. The Consumer Price Index (Rural/Urban/Combined) holds key importance as it represents all strata of the population, including the Poor, Middle, and Affluent. The dynamics of the market, whether Rural or Urban, play a pivotal role in shaping the value of CPI(R/U/C) since rural and urban markets may exhibit significant variations in the aforementioned price fluctuations and purchasing patterns. The objective of the study is to assess the significance of classifying markets into Rural and Urban categories in Kerala for the generation of CPI(R/U/C).

2.Scope and Structure of the problem

The Laspeyres Formula in formulating CPI(R/U/C), relying on the weights assigned to different commodities grouped into six primary Groups and their corresponding Subgroups. These weight assignments are derived from the outcomes of the 68th round of the National Sample Survey's Consumer Expenditure Survey (CES). There are total of 149 markets across all districts, comprising 78 urban and 71 rural markets, where prices of these commodities are regularly collected. The pricing of each commodity is subject to common factors such as local availability, transportation, and demand, which are considered usual in nature. As highlighted earlier, the prices of approximately four hundred items categorized into groups and subgroups may exhibit variations between rural and urban areas. The approach involves clustering markets based on these price variations.

Since Kerala is a consumer state, it heavily relies on neighbouring states for almost all essential commodities. The lifestyle, occupations of the people, and various development indicators in the state suggest that it is undergoing rapid urbanization, distinguishing it as one of the states in the country experiencing substantial urbanization. Here comes the attention to a machine learning approach to cluster the markets in Kerala to enable the impact of Rural and Urban classification on CPI.

The objective is employing a machine learning approach to cluster markets in Kerala based on the influence of Rural and Urban classifications on the Consumer Price Index CPI(R/U/C). This initiative holds the potential to offer valuable insights into the distinctions between Rural and Urban markets considering the price values of commodities. Key steps include identifying relevant features, pre-processing the data, selecting a suitable clustering algorithm, feature engineering, model training, evaluation, interpretation of results, visualization, and a detailed analysis of the impact of Rural and Urban classifications on the formed clusters. The ultimate goal is to provide a comprehensive understanding of how CPI(R/U/C) item basket price values contribute to market clustering and the subsequent differentiation between Rural and Urban classifications.

3. K-Means Clustering

Clustering is an unsupervised learning technique designed to partition data into distinct groups where the elements within each group exhibit similarities. The primary objective of clustering is to unveil meaningful and significant patterns within the data. These groups can serve various purposes, such as direct analysis, in-depth exploration, or utilization as features or outcomes for predictive regression or classification models. K-means, being the first clustering method developed, remains widely used due to its algorithmic simplicity and effectiveness in handling large datasets.

In our scenario, the application of the K-Means algorithm to cluster markets into two groups could yield satisfactory model accuracy measures. Comparing these clusters with the actual market labels of Rural and Urban may reveal relevant insights. If the algorithm successfully generates two clusters resembling the actual rural and urban labels, it implies a distinct separation between markets in rural and urban areas. Essentially, in the context of Kerala, there exists a significant divergence between markets situated in rural and urban areas.

4. Data Pre processing

The monthly market data, initially unstructured, underwent a three-stage data pre-processing to facilitate model building. With approximately 400 commodities featuring different codes related to CPI (R/U/C) and additional characteristics, the goal was to extract commodities sharing similar attributes across markets, creating a robust dataset. This process involved the development of Python code comprising three user-defined functions:

1. Phase 1 Function:
   - To extract each commodity with a unique item code from the Excel file. Included features such as unit, observed quantity, base price, and monthly prices. The output was a dataset encompassing 298 commodities and their corresponding price values for a span of 6 months, from January 2020 to June 2020.
2. Phase 2 Function:
   -To Unify units (e.g., Kilogram, Numbers, Packets) based on a Unit Master. Updated unit information consistently across markets for the dataset generated in Phase 1.
3. Phase 3 Function:
   - To standardize price values of commodities according to the unique unit and observed quantity. This standardization enabled meaningful market-to-market comparisons.

These three functions significantly reduced the workload in pre-processing data for 149 markets. The efforts resulted in successfully obtaining processed data for 139 markets, while the remaining markets exhibited substantial inconsistencies that proved challenging to pre-process. The resultant dataset showcases a structured and standardized representation of market data, facilitating efficient comparisons across 139 markets. Upon further examination of the data, it became evident that the price values of the initial 298 commodities were significantly influenced by the diverse selection of products

during the item basket fixation process, leading to substantial variations across markets. Recognizing the potential challenges posed by the highly inconsistent dataset, a decision was made to narrow the scope of the dataset to focus exclusively on six varieties of rice.

Rice, being a fundamental commodity, undergoes close monitoring of its price fluctuations across markets. Further, uniformity and comparability of attributes such as unit, observed quantity, and base price across all markets leads to streamline the dataset to include only rice varieties. This deliberate restriction enhances the dataset's coherence with the study's objectives, ensuring a more meaningful and reliable foundation for model building. The data set looked like

Unnamed: 0	Months	Rice1	Rice2	Rice3	Rice4	Rice5	CODE
0	Jn-20	47.000000	60.0	36.0	36.00	35.0	U
1	Fb-20	47.000000	60.0	36.0	36.00	35.0	U
2	Mr-20	47.000000	60.0	36.0	36.00	35.0	U
3	Ap-20	47.000000	60.0	36.0	36.00	35.0	U
4	My-20	47.800000	60.0	38.0	39.00	35.0	U
...	...	...	...	...	...	...	...
829	Fb-20	41.897112	36.0	37.0	38.00	29.0	R
830	Mr-20	41.897112	36.0	37.0	38.00	29.0	R
831	Ap-20	41.897112	36.0	37.5	37.75	29.0	R
832	My-20	41.897112	36.0	37.5	37.75	29.0	R
833	Ju-20	41.897112	36.0	39.0	38.00	29.0	R

5. Tools an Packages used

The tools and libraries utilized to construct the model in Python include the following libraries:

- `numpy`
- `pandas`
- `matplotlib`
- `seaborn`
- `sklearn`

These libraries were instrumental in various stages of the model development process, providing essential functionalities for data manipulation, visualization, and machine learning tasks.

6. Graphical analysis of distribution of rice varieties over Urban and Rural markets

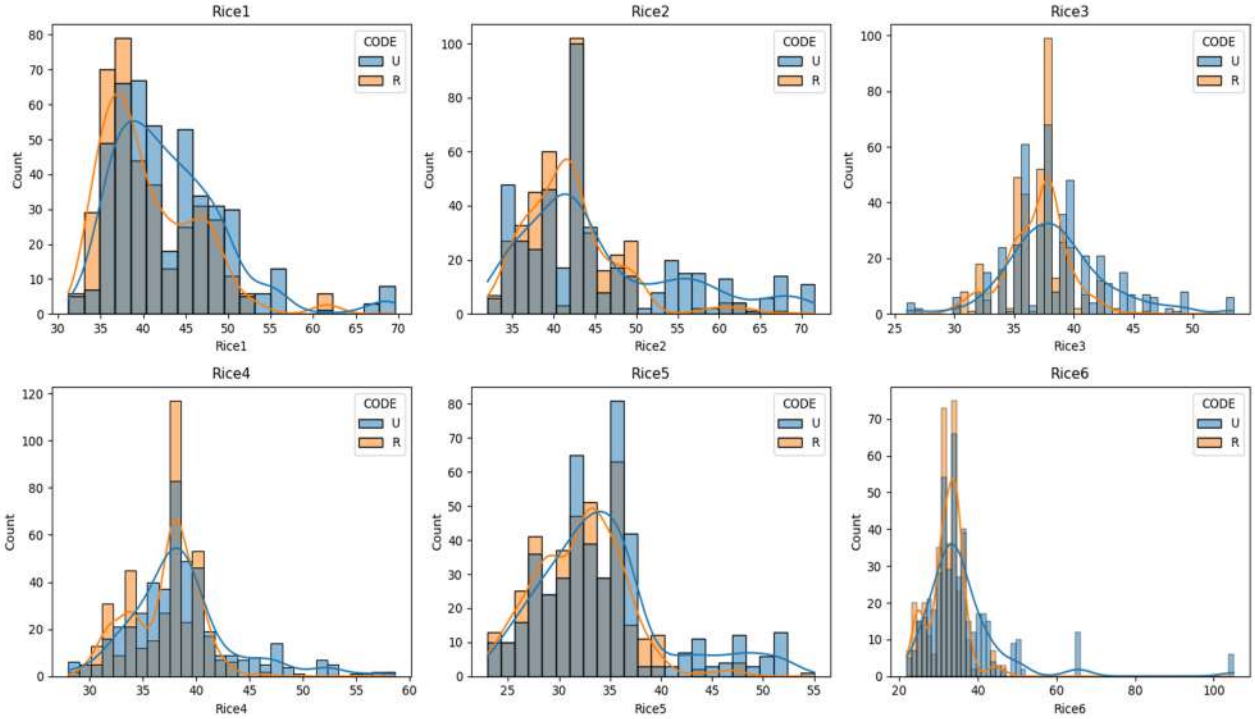


Figure1

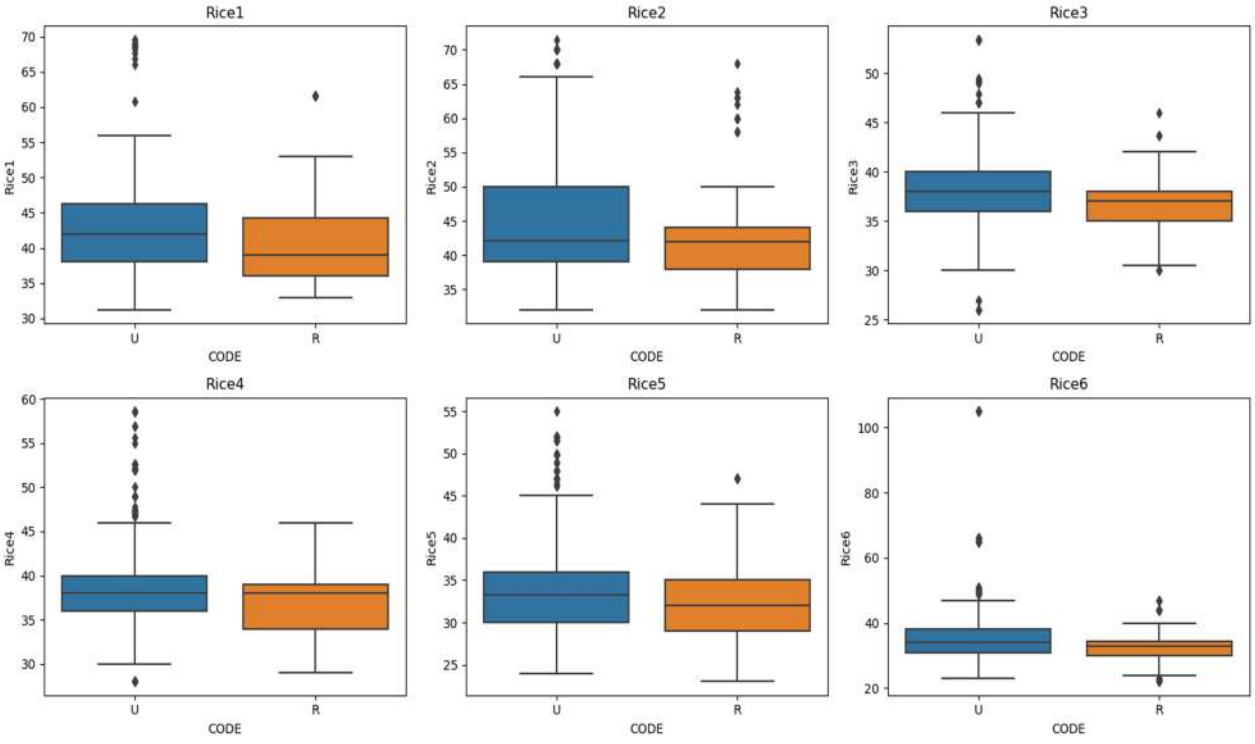


Figure 2

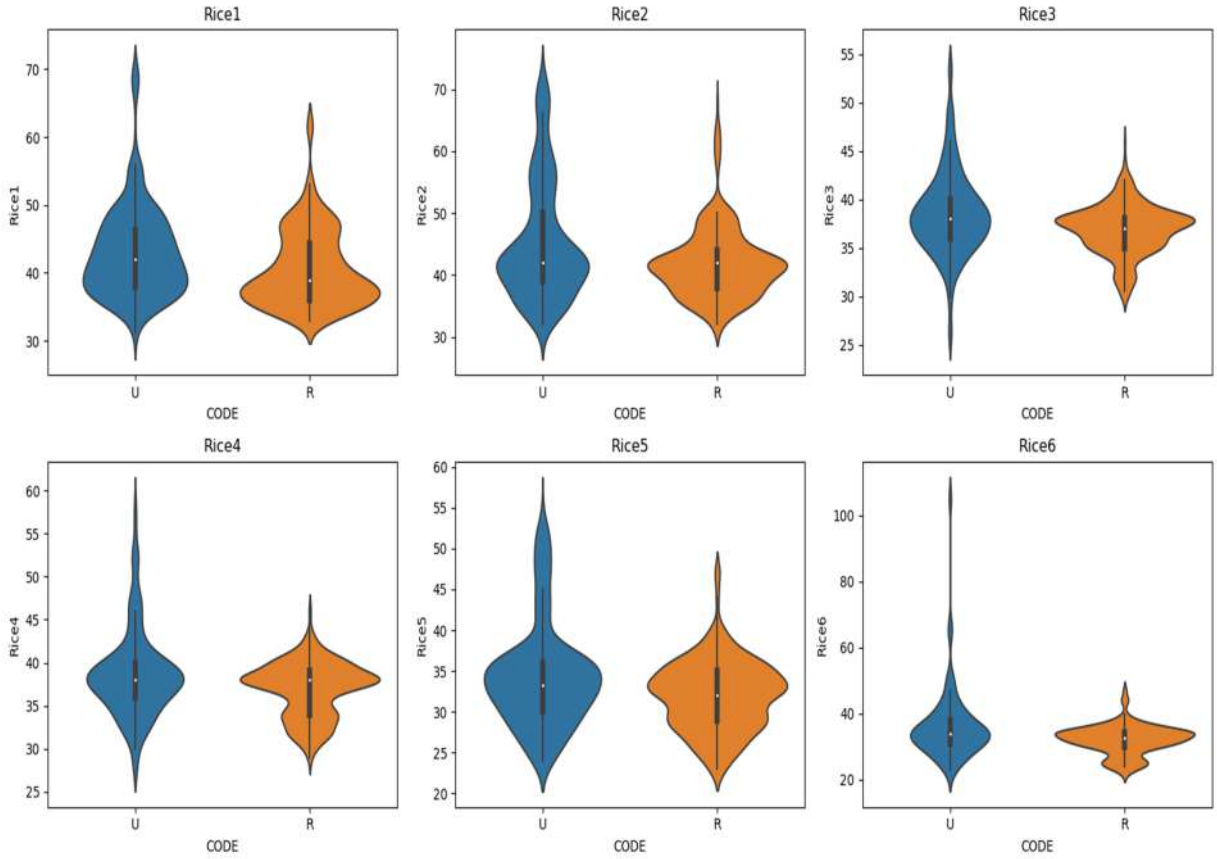


Figure 3

The graphical analysis explains the distribution of six rice varieties across Rural and Urban markets. Notably, all varieties exhibit a nearly identical pattern, leaning slightly towards the right tail. Both box plots and violin plots validate this observation. It is evident that the behaviour of all six varieties remains consistent across both Rural and Urban categories. While certain varieties may demonstrate higher distribution values in Urban markets compared to Rural ones, such differences are insufficient to conclude that Urban markets are generally more expensive than their rural counterparts.

7. Model Building

K-means algorithm precisely applied to cluster the all 834 data points two cluster with their under laying similarities. The below given plot exhibit the two clusters clearly.

18	366
97	353

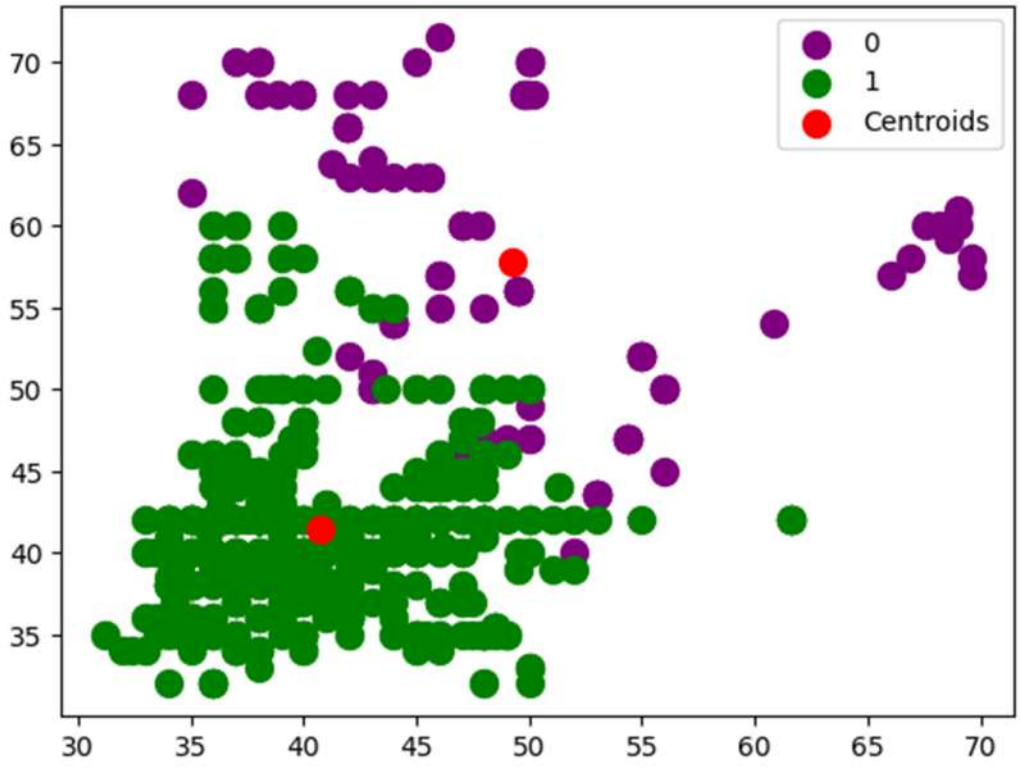


Figure 4

The figure visually indicates the absence of a definitive classifier or boundary to exclusively distinguish between the two clusters, despite their formation. The silhouette score, measuring the cohesion and separation of clusters, is only 0.482. This score suggests that the model does not demonstrate a significant difference between the two obtained clusters, emphasizing the challenge of establishing a clear distinction based on the available features. The provided confusion matrix is,

This implies that out of the total 834 data points, only 371 are correctly assigned to their respective labels, Rural or Urban. The precision values for each cluster, at 16% and 48%, further underscore the model's subpar performance. These precision values indicate the proportion of correctly identified instances within each cluster, highlighting the challenges in accurately classifying data points into their correct categories.

The model's poor performance in clustering the dataset into two categories underscores the challenge of determining the actual number of clusters that can effectively subdivide the dataset. The Elbow method is employed to address this issue. This technique, commonly used in data analysis and machine learning, aims to identify the optimal number of clusters for k-means clustering. The method entails executing the k-means clustering algorithm across a range of k values (number of clusters) and plotting the sum of squared distances from each point to its assigned centre against the number of clusters. The "elbow" in the plot signifies a point where the rate of decrease in the sum of squared distances slows down, forming a bend resembling an elbow. The objective is to select the k value at the elbow point because, beyond that point, additional clusters do not significantly reduce the sum of squared distances, and extra clusters may not yield meaningful separation in the data.

Here is the elbow plot

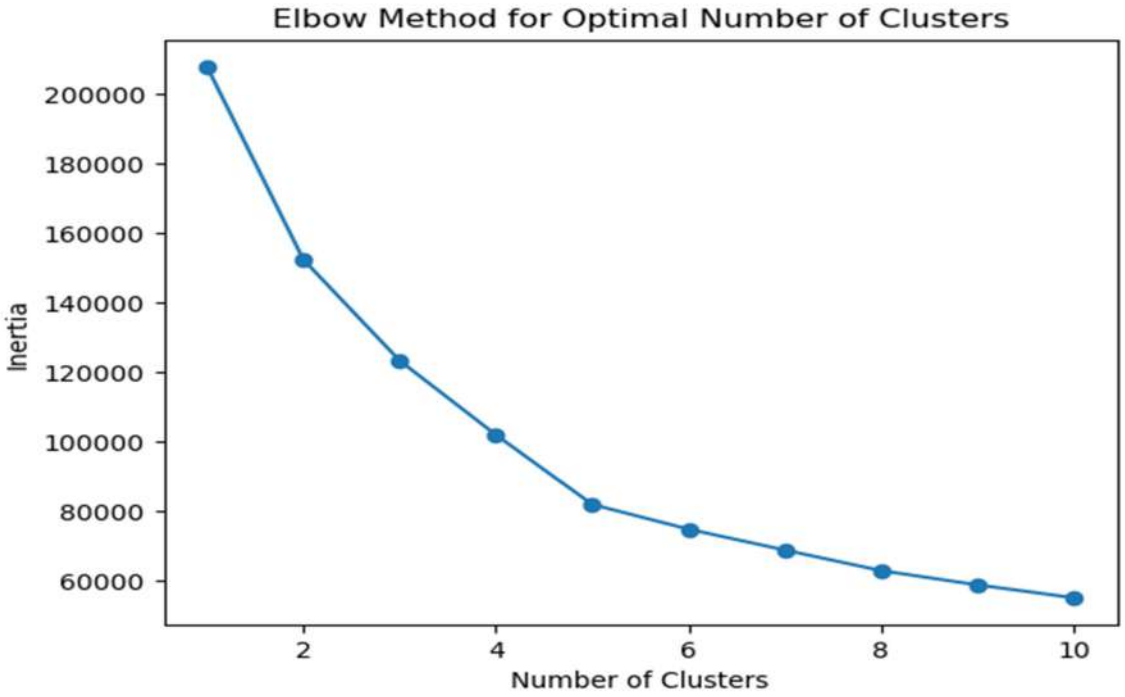


Figure 5

In the provided elbow plot (Figure 5), it is evident that no distinct elbow point is present. This absence suggests that there is no optimal number for subdividing the dataset into clusters. In other words, the subgroups within the dataset exhibit significant similarity in nature, making it challenging to identify a meaningful number of clusters. For instance the attempt to cluster the data set into five clusters gave poor clusters as shown in figure below.

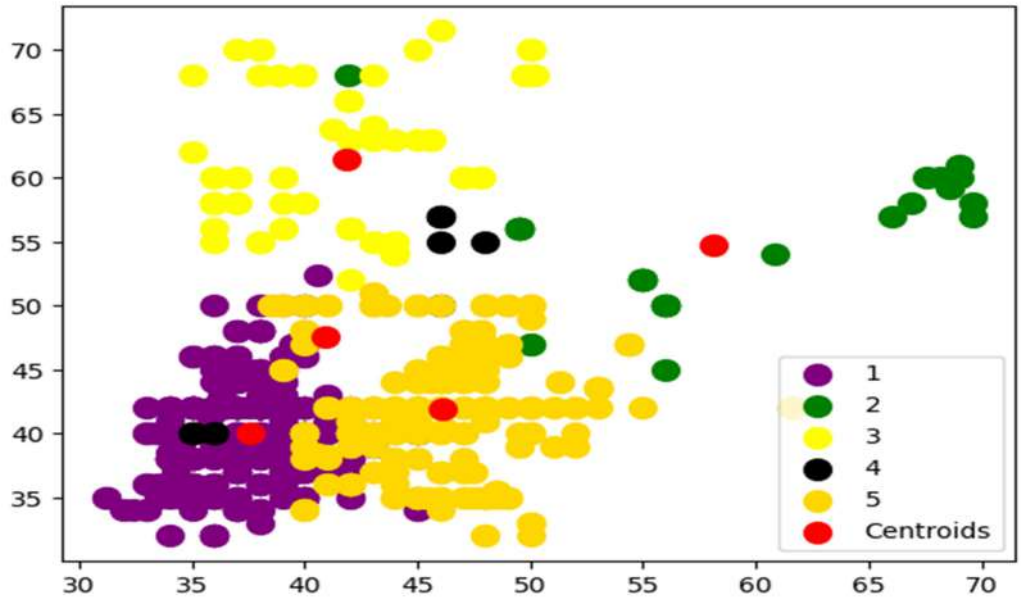


Figure 6

The figure visually demonstrates that the clusters formed exhibit overlap, indicating a lack of reliability in the clustering results.

8. Inference and Conclusion

The objective was to cluster markets in Kerala based on the price values of six rice varieties. The outcomes of the K-Means clustering analysis imply that there is no significant difference in terms of price values between rural and urban markets. Simply put, the prices in rural and urban areas do not vary significantly. This observation aligns with the earlier mentioned point, supporting the inference that rural areas in Kerala are undergoing a substantial rate of urbanization.

9. Opportunities for Further Investigation

Expanding the model to include a broader range of commodities over an extended time frame could yield valuable insights for market selection, replacement, and other analyses related to price values in the computation of Consumer Price Index (CPI) for Rural, Urban, and Combined areas. This enhanced scope may provide a more comprehensive understanding of market dynamics, facilitating more informed decision-making in terms of commodity trends, market fluctuations, and economic scrutiny

10. References

- Peter Bruce, Andrew Bruce and Peter Gedeck. 2020. Practical Statistics for Data Scientists, O'REILLY, Second edition 283-302.

- Methodology for Compilation of Consumer Price Index Numbers(Rural/Urban/Combined) at District Level In Kerala, Price Division, Directorate of Economics and Statistics, Kerala

11. Appendix

Code for Phase1, Phase2, Phase3 functions written for data processing

1.Phse1 Function

```
import numpy as np
import pandas as pd
```

```
def ps1(d,i,c,m):
    df=d
    itemUpdtd_N=i
    M_code=c
    market=m

    # Specify the column numbers you want to drop
    columns_to_drop = [0,1,2, 3,4, 5,6,7,8,9,10,11,12,14,15,16,18]

    # Drop the specified columns
    df_drpd = df.drop(df.columns[columns_to_drop], axis=1)

    df_drpd1 = pd.DataFrame(df_drpd)


    itemUpdtd_N['ITEM CODE'] = itemUpdtd_N['ITEM CODE'].astype(str)

    # Create a new DataFrame with the concatenated values
    itemUpdtd = pd.DataFrame({'ITEM CODE': itemUpdtd_N['ITEM CODE'].apply(lambda x: M_code + x)})

    # Step 1: Filter rows based on 'ITEM CODE'
    filtered_Data = df_drpd1[df_drpd1['ITEM CODE'].isin(itemUpdtd['ITEM CODE'])]
```

```
    # Step 2: Get the unique 'ITEM CODE' values from itemUpdtd
    unique_item_codes = itemUpdtd['ITEM CODE'].unique()

    # Step 3: Create a new DataFrame with the unique 'ITEM CODE' values
    blank_data = pd.DataFrame({'ITEM CODE': unique_item_codes})

    # Step 4: Merge the filtered_Data with blank_data to keep only the rows with matching 'ITEM CODE'
    result_df = pd.merge(blank_data, filtered_Data, how='left', on='ITEM CODE')

    result_df=result_df.iloc[:, 0:11]
    result_df.columns= ['ITEM CODE', 'ITEM', 'Unit','Obsd qnty','Base price','Jn-20','Fb-20','Mr-20','Ap-20','My-20','Ju-20']

    result_df.to_excel(market +'Sorted_P1.xlsx', index=False)
    print("successfully srted")
```

```
: df=pd.read_excel('065 Kalpetta affluent CPI (r_u_c) schedule.xlsx')
```

```
: itemUpdtd_N=pd.read_excel('itemUpdtd_Code.xlsx')
```

```
: ps1(df,itemUpdtd_N,'212','KPTA_A')
```

```
successfully srted
```

2.Phse2 Function

```
import numpy as np
import pandas as pd
```

```
dt=pd.read_excel('CHENGSorted_P1.xlsx')
dt = dt.reset_index(drop=True)
```

```
ut=pd.read_excel('Unit_replace.xlsx')
ut = ut.reset_index(drop=True)
```

```
Unit_Master=pd.read_excel('Unit_New_Master.xlsx')
```

```
def ps2(d,um,ur,m):

    dt_New=d
    Unit_Master=um
    ut=ur
    market=m
```

```
    Unit = dt.loc[:, 'Unit'].to_frame()

    Unit_N=Unit.drop_duplicates()

    not_in_Unit_Master = Unit_N[~Unit_N['Unit'].isin(Unit_Master.values.flatten())]
    # Check if the DataFrame is blank
    is_blank = not_in_Unit_Master.isna().all().all()
    if is_blank:
        print("Unit Is Updated")
    else:
        not_in_Unit_Master.to_excel('Unit'+ market+'.xlsx', index=True)
        print("Please Check Unit Data")
```

```
: ps2(dt,Unit_Master,ut,'CHENG')
```

```
Unit Is Updated
```

3.Phse3 Function

```
import numpy as np
import pandas as pd
```

```
dt=pd.read_excel('ALAPSorted_P1.xlsx')
dt = dt.reset_index(drop=True)
ur=pd.read_excel('Unit_replace.xlsx')
ur = ur.reset_index(drop=True)
Unit_Master=pd.read_excel('Unit_New_Master.xlsx')
```

```
def ps3(d,um,ur,m):
    dt_New=dt
    Unit_Master=um
    ut=ur
    market=m
    # converting prices to numeric
    columns_to_convert = ['Jn-20', 'Fb-20', 'Mr-20','Ap-20','My-20','Ju-20']
    dt_New[columns_to_convert] = dt_New[columns_to_convert].apply(pd.to_numeric, errors='coerce')

    #converting observed qty numeric only when 10kg type
    dt_New['Obsd qnty'] = dt_New['Obsd qnty'].astype(str)
    dt_New['Obsd qnty'] = dt_New['Obsd qnty'].str.extract('(\d+)', expand=False)
    dt_New['Obsd qnty'] = pd.to_numeric(dt_New['Obsd qnty'], errors='coerce')
```

```
    # Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Kg'

    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Base price'] = dt_New['Base price'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Obsd qnty'] = 1
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Kg']), 'Unit'] ='Kg'

    # Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Gm'

    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']*100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Obsd qnty'] = 100
    dt_New.loc[dt_New['Unit'].isin(Unit_Master['Gm']), 'Unit'] ='Gm'
```

```
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'No'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['No']), 'Unit'] ='No'
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Ml'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']*100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Obsd qnty'] = 100
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ml']), 'Unit'] ='Ml'

# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Ltr'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ltr']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ltr']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ltr']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ltr']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Ltr']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
```

```
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Unit'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Unit']), 'Unit'] ='Unit'


# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Pck'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Pck']), 'Unit'] ='Pck'
```

```
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Mtr'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Ju-20'] = dt_New['Ju-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mtr']), 'Unit'] ='Mtr'


# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Day'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Ju-20'] = (dt_New['Ju-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Day']), 'Unit'] ='Day'
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Mnth'
```

```
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Mnth'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Ju-20'] = (dt_New['Ju-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Mnth']), 'Unit'] ='Mnth'

# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Hr'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Fb-20'] = dt_New['Fb-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Mr-20'] = dt_New['Mr-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Jn-20'] = dt_New['Jn-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Ju-20'] = (dt_New['Ju-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Hr']), 'Unit'] ='Hr'
```

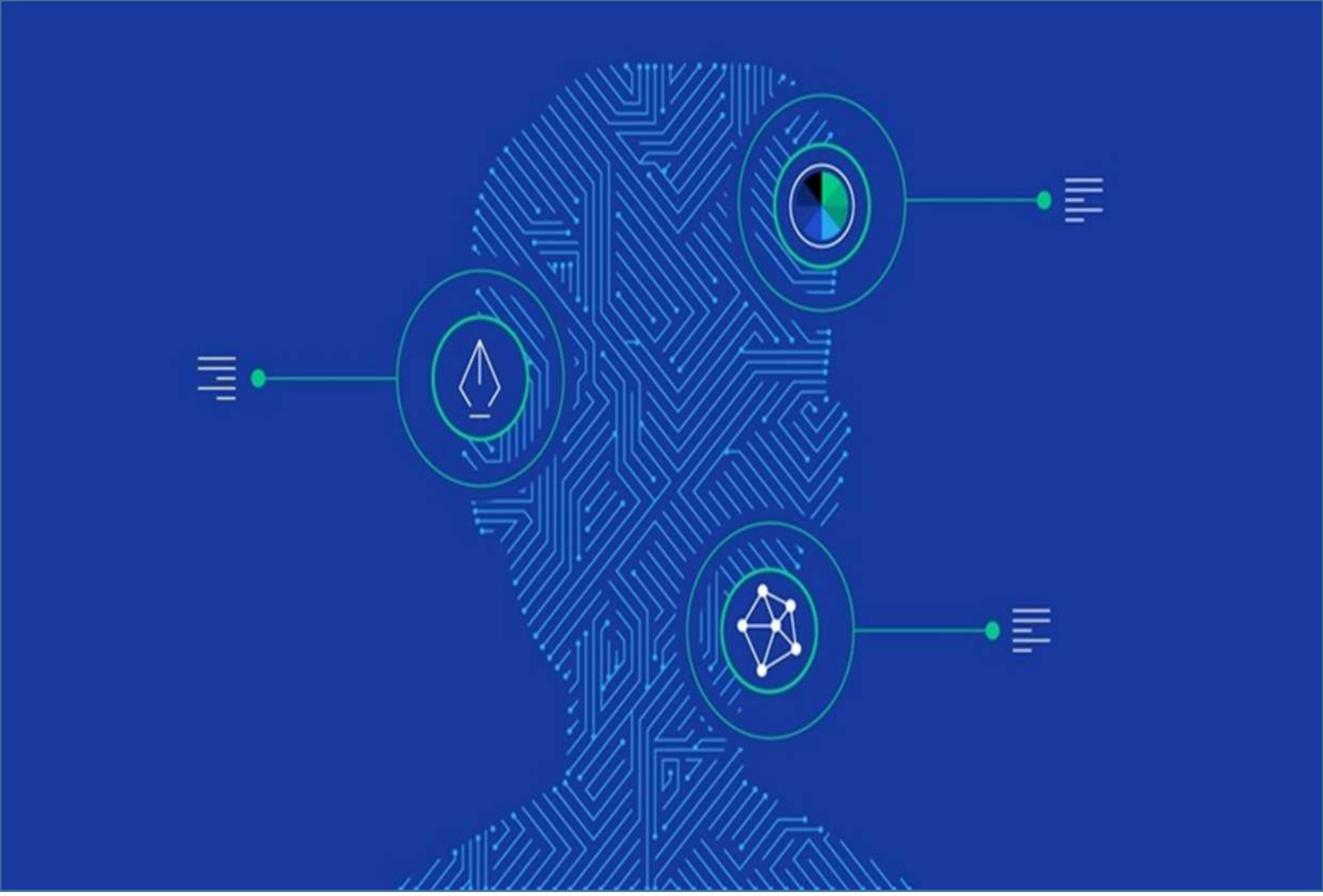
```
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Yr'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Fb-20'] = (dt_New['Fb-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Mr-20'] = (dt_New['Mr-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Ju-20'] = (dt_New['Ju-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Yr']), 'Unit'] ='Yr'


# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Km'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Fb-20'] = (dt_New['Fb-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Mr-20'] = (dt_New['Mr-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Ju-20'] = (dt_New['Ju-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Km']), 'Unit'] ='Km'
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Set'
```

```
# Update 'Obsd qnty' and 'Base price' for rows where 'Unit' is 'Set'

dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Base price'] = (dt_New['Base price'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Jn-20'] = (dt_New['Jn-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Fb-20'] = (dt_New['Fb-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Mr-20'] = (dt_New['Mr-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Ap-20'] = dt_New['Ap-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'My-20'] = dt_New['My-20'] / dt_New['Obsd qnty']
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Ju-20'] = (dt_New['Ju-20'] / dt_New['Obsd qnty'])
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Obsd qnty'] = 1
dt_New.loc[dt_New['Unit'].isin(Unit_Master['Set']), 'Unit'] ='Set'

# updating rows without unit with std unit in unit_replace
dt_New['Unit'] = dt_New['Unit'].fillna(dt_New['ITEM CODE'].map(ut.set_index('ITEM CODE')['Unit']))


selected_Months = dt_New.loc[:, ['ITEM CODE', 'Jn-20', 'Fb-20','Mr-20','Ap-20','My-20','Ju-20']]
Tp= selected_Months.reset_index(drop=True).transpose()

Tp.columns = Tp.iloc[0]  # Set the first row as column headers
Tp = Tp[1:]  # Reset the index, excluding the first row
```

```
    Tp.columns = Tp.iloc[0]  # Set the first row as column headers
    Tp = Tp[1:]  # Reset the index, excluding the first row

    # Ensure the column names are strings
    Tp.columns = Tp.columns.astype(str)
    # Assuming Mndy is your DataFrame
    Tp.loc[:, 'Market'] =market

    # Check for NaN values
    nan_columns = Tp.columns[Tp.isna().any()].tolist()

    # Replace NaN values with the average of each column
    for column in nan_columns:
        column_mean = Tp[column].mean()
        Tp[column].fillna(column_mean, inplace=True)

    #saving cinal matrix
    Tp.to_excel(market+'_final_Jan-Ju20.xlsx', index=True)
    print("Matrix Is Updated")
```

```
: ps3(dt,Unit_Master,ur,'ALAP')
```

```
Matrix Is Updated
```



Prediction of Farm Wholesale Price of Paddy Cost of Cultivation as Predictor

Submitted by

Sri. Saju K,

Research Assistant

1. Executive Summary

The project aims to develop predictive models for forecasting the wholesale price of paddy with cost of cultivation of paddy as predictor. Two machine learning algorithms, Decision Tree Regression and Random Forest, are employed for prediction. The project seeks to provide farmers and stakeholders with valuable insights into future paddy prices, enabling informed decision-making.

2. Introduction

2.1 Background

Agriculture plays a crucial role in the state's economy, providing livelihoods to a significant portion of the population. Paddy cultivation is a cornerstone of agriculture in Kerala, contributing substantially to the state's food production. The success of paddy farming directly impacts the food security and economic well-being of the state. The farm price of paddy is a critical factor for farmers, influencing their income and overall financial stability. Fluctuations in paddy prices can significantly affect the livelihoods of farmers, making accurate price prediction essential. The need for an efficient and reliable system for predicting farm wholesale prices of paddy in Kerala is evident. Agricultural markets are dynamic, and predicting the prices of crops is essential for farmers and traders to make strategic decisions. This project focuses on predicting the wholesale price of paddy using machine learning models.

2.2 Objectives

- Develop a Decision Tree Regression model for predicting paddy prices.
- Implement a Random Forest model for comparison.
- Evaluate and compare the performance of both models.

3. Data Collection and Pre-processing

3.1 Data Source

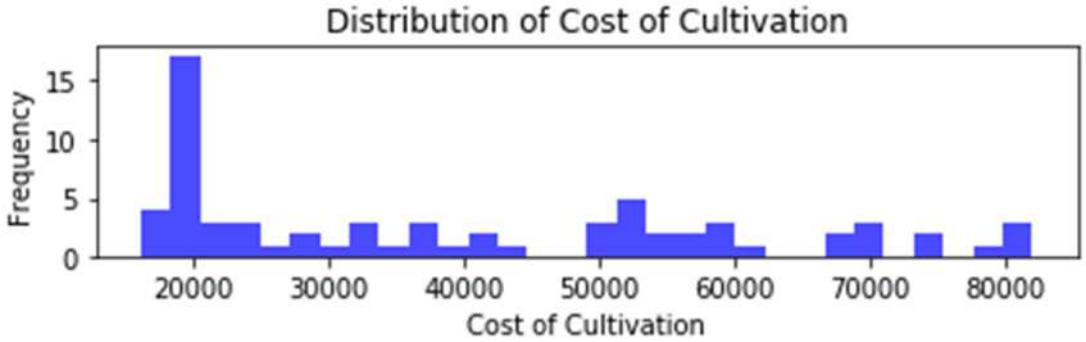
The dataset used for this project is collected from the annual publications from two sections namely Cost of cultivation and Prices of the Department of Economics and Statistics directorate.

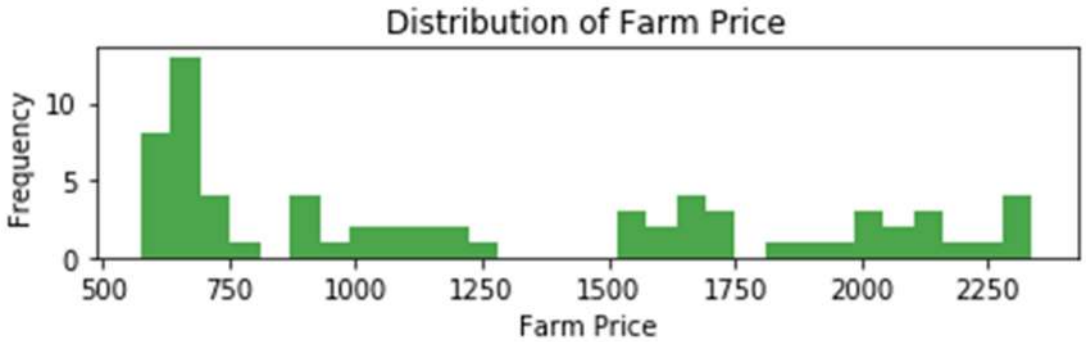
3.2 Data Pre-processing

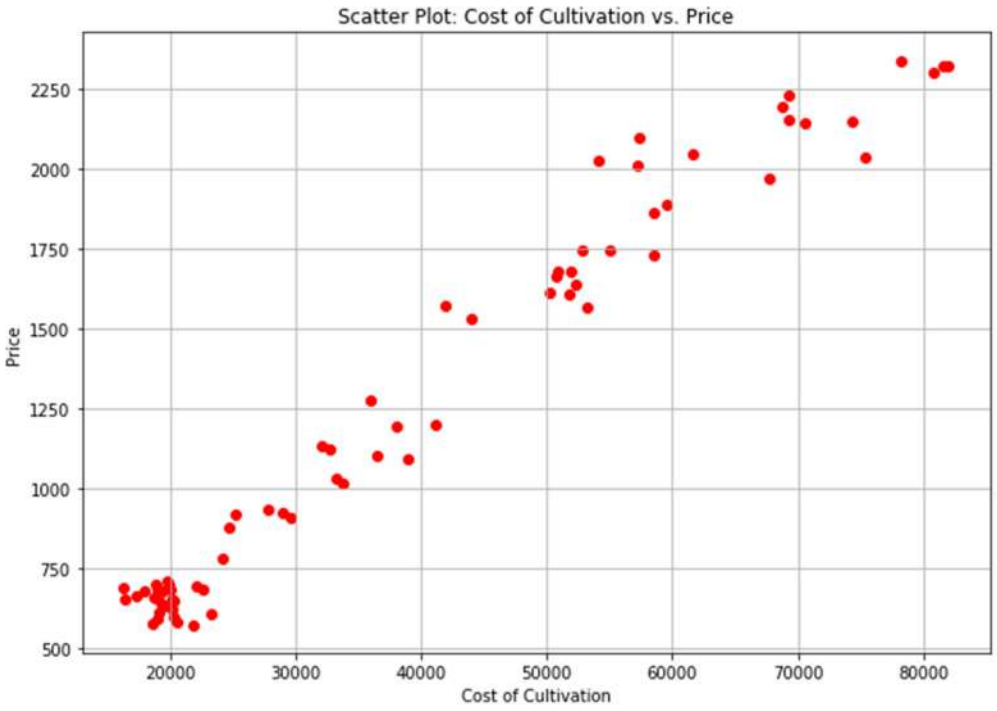
The dataset available in the prices section is in the form of monthly data while the cost of cultivation is in the form of season wise data (Autumn, Winter, Summer). So the first step of the data pre-processing is to convert the monthly data in to season wise data by taking the average of four months. That is taking average of July, August, September and October to get the autumn season Farm price. After the pre-processing 23 years of 3 season data from 2000 to 2022 is available for the project. The dataset used and visualization are given below.

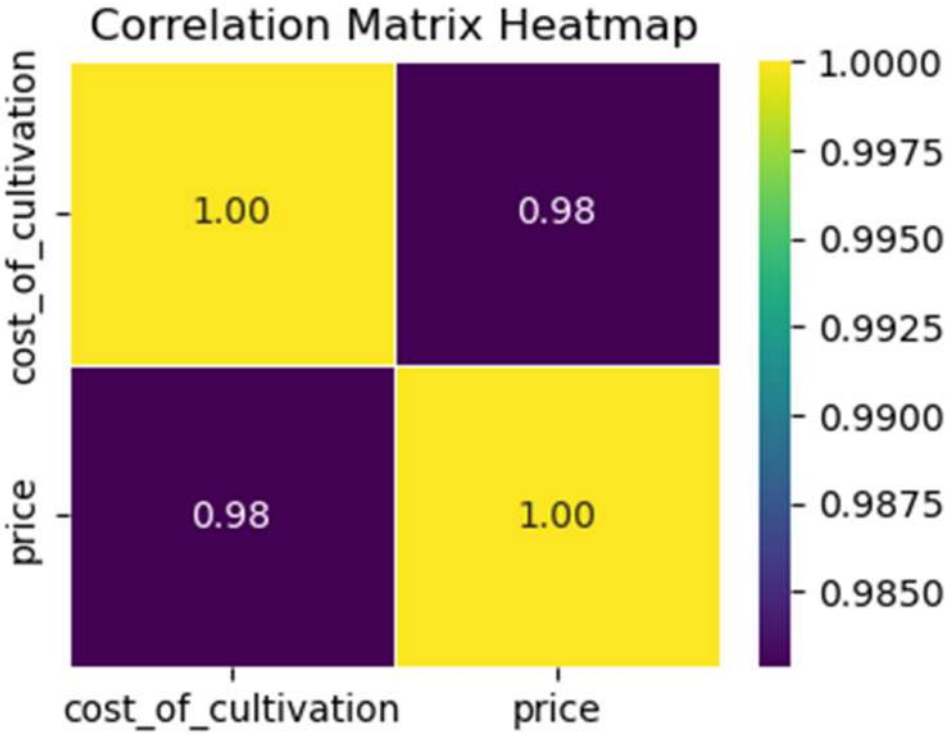
Sl.No	cost_of_cultivation	price
1	16218	688.25
2	17916	680.50
3	19354	678.25
4	17210	666.50
5	16318	654.00
6	20270	601.25
7	20484	584.00
8	18614	576.82
9	21754	576.02
10	19777	636.26
11	18715	658.23
12	20111	657.43
13	19932	683.81
14	18925	688.09
15	19694	711.13
16	18835	702.86
.	.	.
.	.	.
67	75430	2038.41
68	82025	2322.86
69	80802	2302.13

Cost_of_Cultivation -Rs/Ha
Price -Rs/Qtl









The scatter plot shows a positive correlation between cost of cultivation and farm wholesale price of paddy. Also the heatmap shows correlation coefficient of 0.98 between the cost of cultivation and the farm wholesale price of paddy indicates a very strong positive correlation.

4. Methodology

4.1 Tools and Libraries used

The predictive models were fitted using Python Programming. The techniques used were Decision Tree Regression and Random Forest. The libraries used were

- **pandas**
- **joblib**
- **sklearn**
- **matplotlib**
- **math**

4.2 Decision Tree Regression

Decision Tree Regression is employed to build a model that recursively splits the dataset based on features, creating a tree-like structure. This model is suitable for capturing non-linear relationships in the data.

Advantages:

- *Interpretability:* Decision trees offer a clear and interpretable representation.
- *Handling Non-linearity:* Effective in capturing non-linear relationships in data.
- *Feature Importance:* Provides insights into the importance of different features.

4.3 Random Forest

Random Forest is an ensemble learning method that combines multiple Decision Trees to improve predictive accuracy and control over fitting. It works by training several trees on different subsets of the data and averaging their predictions.

Advantages:

- *High Accuracy:* Random Forest tends to provide accurate predictions.
- *Reduced Over fitting:* Randomization helps mitigate over fitting seen in individual decision trees.
- *Feature Importance:* It can assess the importance of different features in making predictions.

5. Model Training and Evaluation

5.1 Decision Tree Regression Model

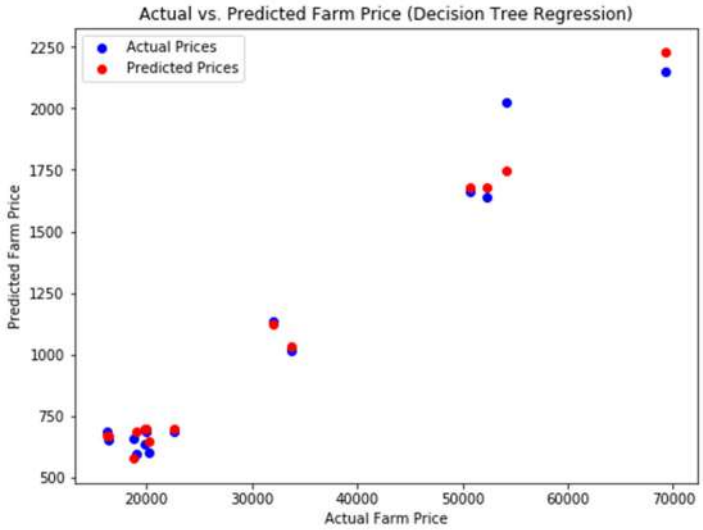
- Trained the model on the training dataset. (80% of the Dataset)
- Evaluated the model on the testing dataset. (20% of the Dataset)
- Measured performance using metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared.

Observed error values are given below.

Mean Squared Error : 7782.09083714875
Mean Absolute Error : 56.074275071428566
Root Mean Squared Error : 88.21615972795885
R-squared : 0.9739485254884002

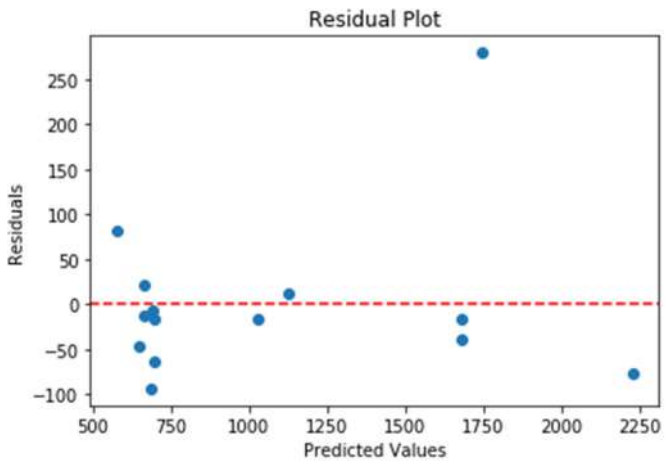
Actual v/s Predicted Values

A graph is plotted with actual and predicted values of the Farm wholesale price of paddy. By analyzing the graph we can see that the predicted values are approximately nearer to the actual values.



Residual Plot

A residual plot is plotted to assess the goodness of fit of the fitted model. It helps to identify patterns or trends in the residuals, which are the differences between the observed and predicted values of the dependent variable. Analyzing residual plots is crucial for understanding whether a regression model captures the underlying patterns in the data adequately.



Random Prediction

A random prediction is done using the value **80,000** as cost of cultivation. The predicted value of the Farm wholesale price is **2302.13375.**

5.2 Random Forest Model

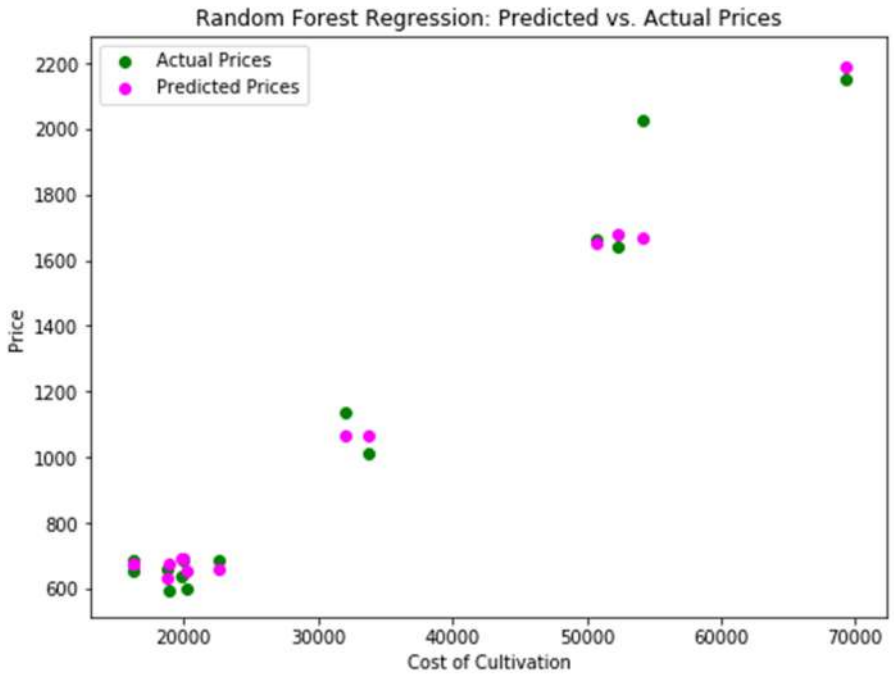
- Train the Random Forest model on the training dataset. (80% of the Dataset)
- Evaluate the model on the testing dataset. (20% of the Dataset)
- Measured performance using metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared.

Observed error values are given below.

Mean Squared Error : 9552.055205388411
Mean Absolute Error : 57.51485139500009
Root Mean Squared Error : 97.73461620832411
R-squared : 0.9680233592842841

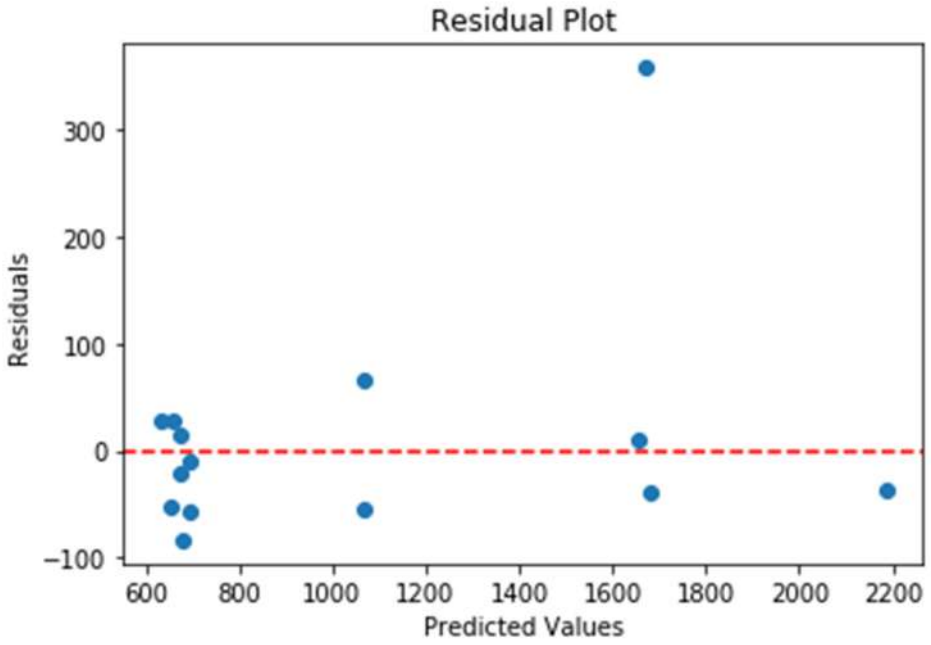
Actual v/s Predicted Values

By analyzing the graph we can see that the predicted values are approximately nearer to the actual values.



Residual Plot

Analyzing the residual plot residuals are randomly scattered around zero, with no discernible pattern, it indicates that the model's predictions are unbiased and capturing the underlying relationships in the data.



Random Prediction

A random prediction is done using the value **80,000** as cost of cultivation. The predicted value of the Farm wholesale price is **2300.9263**

6. Comparison of the Results

A comparison of the results of Decision Tree Regression and Random Forest model evaluations by the performance metrics are given below.

Decision Tree Regression	
Mean Squared Error	: 7782.09
Mean Absolute Error	: 56.07
Root Mean Squared Error	: 88.21
R-squared	: 0.9739

Random Forest	
Mean Squared Error	: 9552.05
Mean Absolute Error	: 57.51
Root Mean Squared Error	: 97.73
R-squared	: 0.9680

MSE and RMSE:

Decision Tree has lower MSE and RMSE values, indicating better performance in terms of average squared and absolute differences. The RMSE of the Decision Tree Regression model is 88.2, which is higher. But comparing the Farm wholesale price of paddy which is nearer to 2300, Rs.88 is a negotiable price.

MAE:

Decision Tree also has a lower MAE, suggesting smaller absolute differences between predicted and actual values.

R^2:

Decision Tree has a higher R^2, indicating a better overall fit in explaining the variance in the target variable.

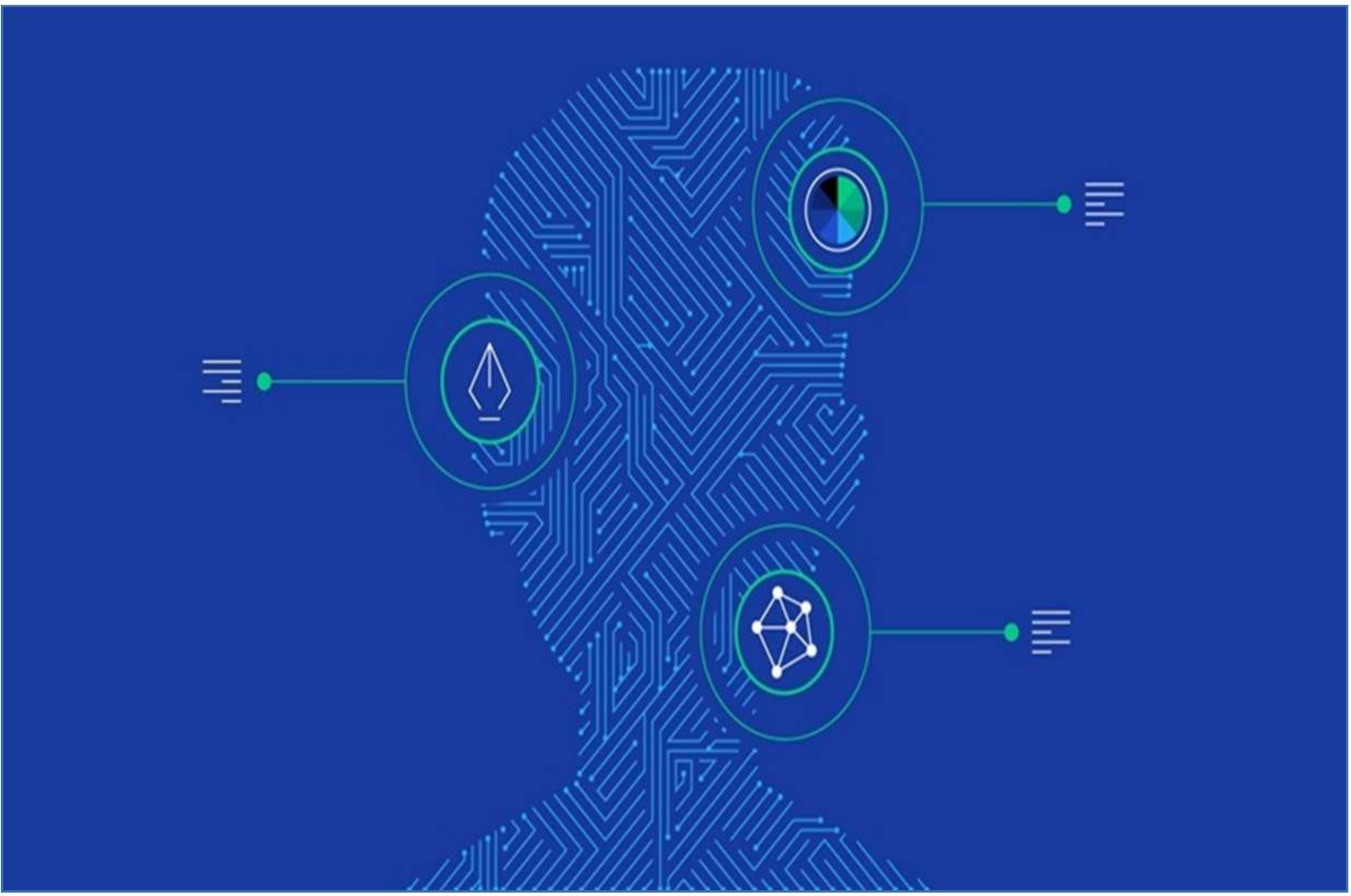
7. Conclusion

After comparing the performance metrics of two models we can find that both model perform approximately the same. Even though Decision Tree Regression model with 97.39% accuracy is more accurate than the Random Forest model. Based on the above values, the Decision Tree Regression model appears to outperform the Random Forest model for the given dataset.

8. References

Publication *1. Report on Cost of cultivation of important crops in Kerala 2000 to 2022*
2.Price Bulletin 2000 to 2022
3.Regression model based studies

This project report provides a comprehensive overview of the process of predicting the wholesale price of paddy using Decision Tree Regression and Random Forest models. It encompasses data collection, pre-processing, model development, evaluation, and comparison, offering valuable insights for stakeholders in the agriculture sector



An Analysis of Suicides in Kerala for the past 10 years

Submitted by

Sri. Shibu B.T, Research Assistant

and

Sri. Brijesh C.J, Statistical Assistant Grade II

Introduction

Suicide is a multidimensional and serious social problem that affects people all around the world, regardless of where they live. Kerala, a state renowned for its beautiful scenery and high human development index, is not exempt from the alarmingly high rate of suicides. The goal of this research is to provide light on the underlying causes, patterns, and trends related to suicides that occurred in Kerala between 1996 and 2021.

Kerala has an impressive standard of living (life expectancy, literacy, health care, and other areas), so it makes sense that issues of socioeconomic development have received a lot of attention. However, there is another startling trend in Kerala that receives little attention from social science research: Kerala has the fourth-highest rate of suicides in India. It is crucial to take a close look at the less talked-about subject of mental health and wellbeing. Suicide needs to be thoroughly investigated in order to develop effective preventive efforts because it is frequently believed that suicide is a symptom of underlying emotional, social, and economic difficulties.

According to the publication of National Crime Records Bureau "Accidental Deaths & Suicides in India 2022" A total of 1,70,924 suicides were reported in the country during 2022 showing an increase of 4.2% in comparison to 2021 and the rate of suicides has increased by 3.3% during 2022 over 2021. Number of suicides reported in Kerala during 2022 is 10162 (5.9% of Total suicides in India) showing an increase of 6.4% in comparison to 2021.

Rate of suicides i.e. the number of suicides per one lakh population, has been widely accepted as a standard yardstick for comparison. All India rate of suicides was 12.4 during the year 2022. Sikkim reported the highest rate of suicide (43.1) followed by A & N Islands (42.8), Puducherry (29.7), Kerala (28.5) and Chhattisgarh (28.2). The rate of suicides in Kerala for the years 2020, 2021 & 2022 are 24.0, 26.9 & 28.5 respectively. Notably, Kerala's suicide rate has exhibited an upward trend, increasing from 24.0 in 2020 to 28.5 in 2022, surpassing the national averages of 11.3, 12.0, and 12.4 for the same respective years. Despite its smaller size, Kerala's consistently higher rates underscore a significant concern for the state.

Scope & Objectives of the study

The scope of this study encompasses a comprehensive examination of suicide trends in Kerala from 1996 to 2021. The focus will extend to analysis the gender wise and age group wise causes of suicides.

Objectives

1. **Gender-wise Analysis:**

   The study aims to examine suicide data with a specific focus on gender. This involves analyzing the number of suicides among males and females separately. It seeks to identify if there are any significant differences in suicide numbers between genders, and if so, what factors may contribute to these differences.

2. **Cause-wise Analysis:**

   This objective involves categorizing and studying suicides based on their causes. Causes may include factors such as health issues, socio-economic problems, family conflicts, or other specific reasons mentioned in the available data. The goal is to understand the predominant causes behind suicides in Kerala and any variations in these causes over the years.

3. **Age Group-wise Analysis:**

   The study will examine suicide data by categorizing it into different age groups. This objective seeks to identify whether certain age groups are more vulnerable to suicide and to understand the patterns and trends in suicide across different age groups. This analysis can provide insights into age-specific risk factors.

4. **Year-wise Analysis:**

   This objective involves examining the suicide data on a yearly basis. The study aims to identify any temporal trends, fluctuations, or significant changes in suicides over the years from 1996 to 2021. By analyzing the data on a yearly basis, the study can capture the dynamics and evolution of suicide patterns in Kerala.

In summary, the study's primary focus is to conduct a detailed analysis by considering multiple dimensions such as gender, cause, age group, and year. This approach enables a comprehensive understanding of the various factors influencing suicides in Kerala, leading to insights that can inform targeted interventions, policy recommendations, and mental health initiatives tailored to specific demographics and contexts.

Limitations of the study

The study focused on suicide data in Kerala from 1996 to 2021, considering factors such as sex, age groups, and causes. Using artificial intelligence, an attempt was made to predict total suicides using the Auto Regressive Integrated Moving Average (ARIMA) model. Due to the limited 26-year dataset, accuracy was generally low in most AI models. Also the study could be more effective with the inclusion of factors like profession, income, and educational qualification.

Data set

Data set for the study is provided by the Statistical wing of State Crime Records Bureau. The dataset contains the following columns.

Year :- This contain information from 1996 – 2021.

Cause:- This column tell us about the reason of committing suicide like Family Problems, illness, Bankruptcy, unemployment etc.

Age group :- This column contains the cause wise total suicide committed by the male & female age groups from 0-14 to 60+.

Tools Used

Python Tools

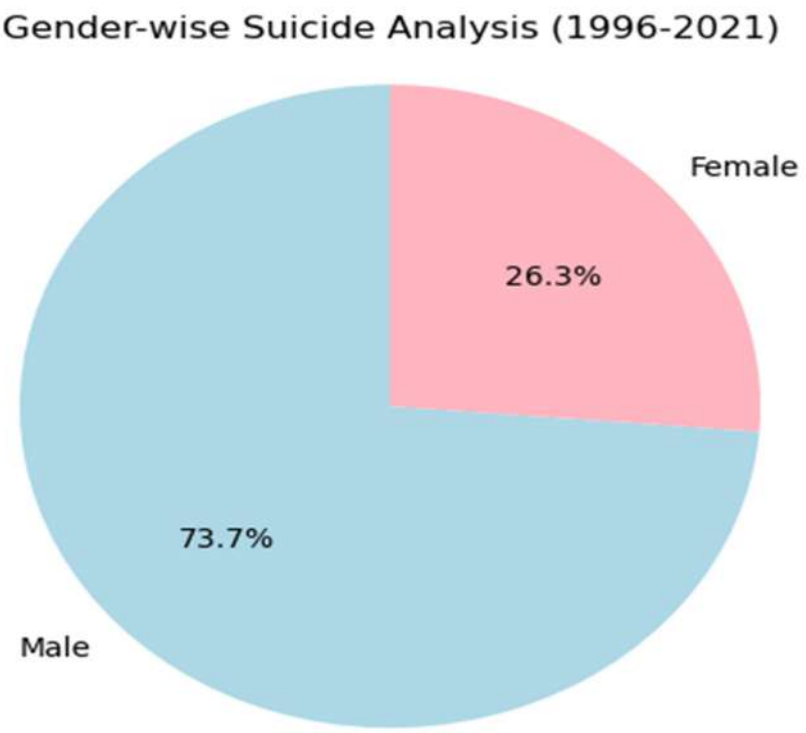
- Pandas
- Numpy
- Seaborn
- SK Learn
- Statsmodels

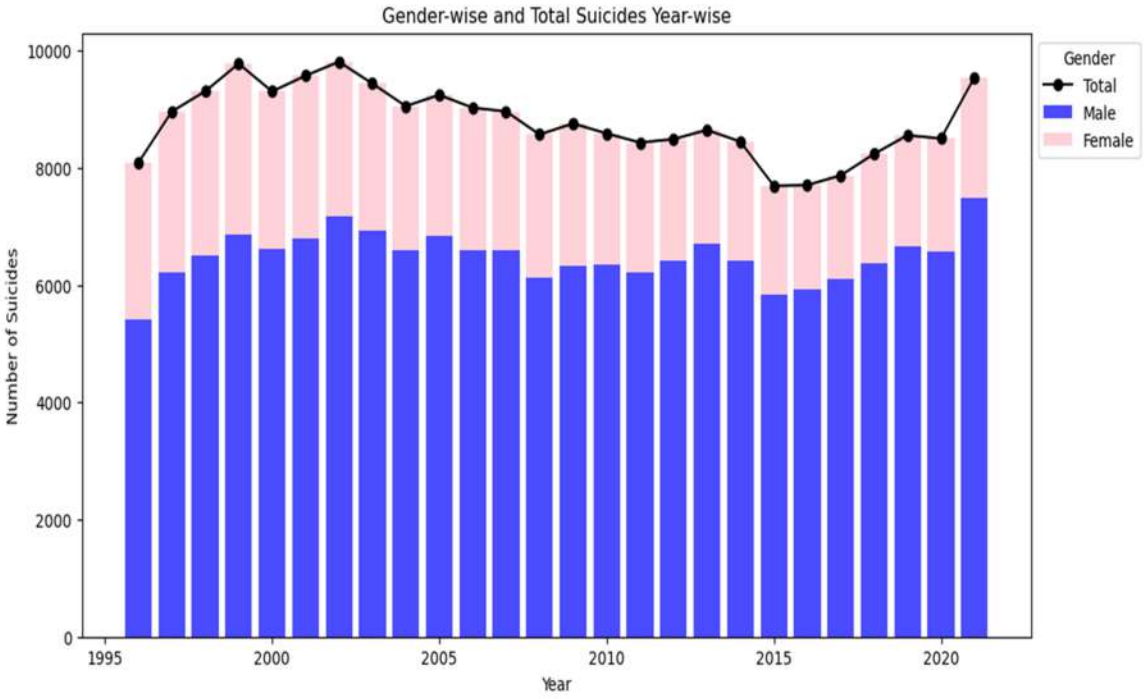
Machine Language

- ARIMA

Gender wise Analysis

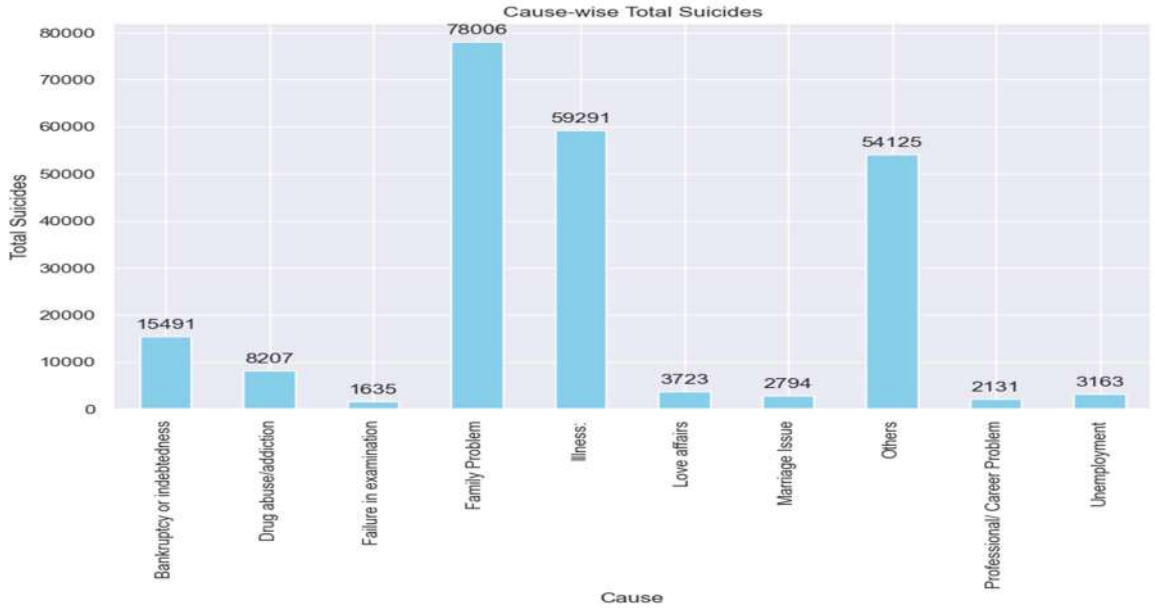
Among the total suicides, a significant gender skew is evident. A staggering 73.7% of the reported suicides were males, totalling 1,68,562, while females accounted for 26.3%, with 60,004 cases.

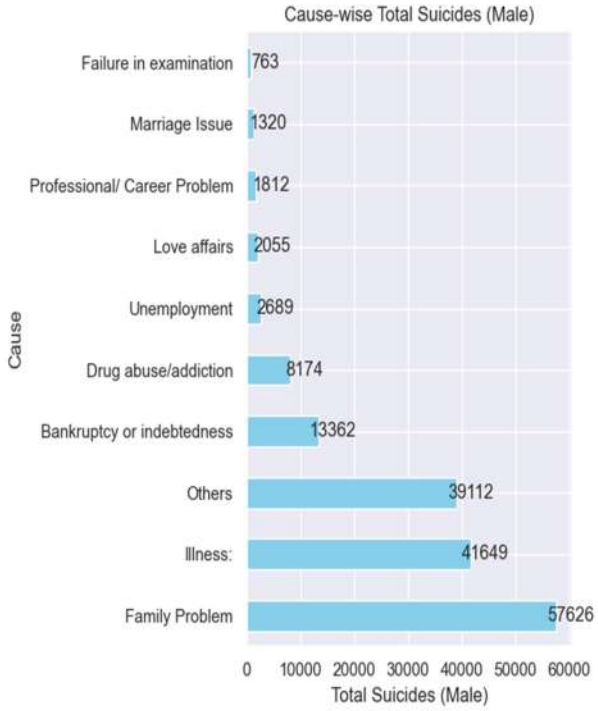

Gender-wise Suicide Analysis (1996-2021)
Female
26.3%
73.7%
Male

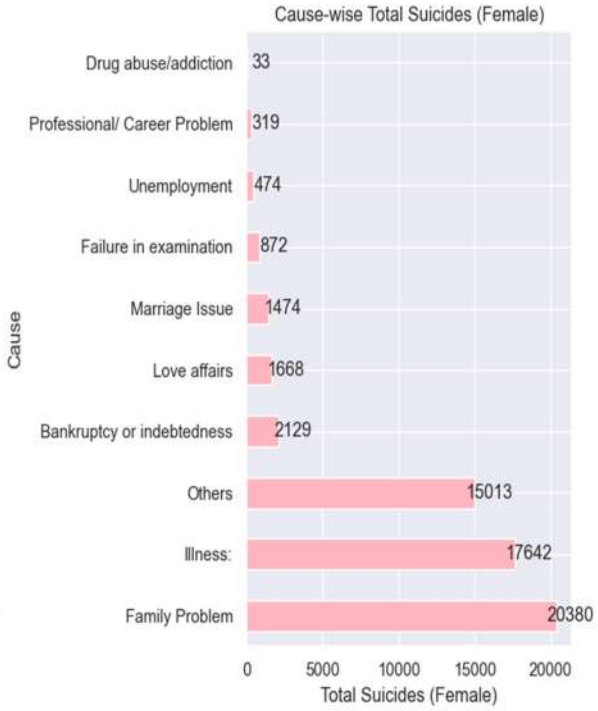

Gender-wise and Total Suicides Year-wise
Number of Suicides
Year
Gender
Total
Male
Female
0
2000
4000
6000
8000
10000
1995
2000
2005
2010
2015
2020

Cause wise Analysis

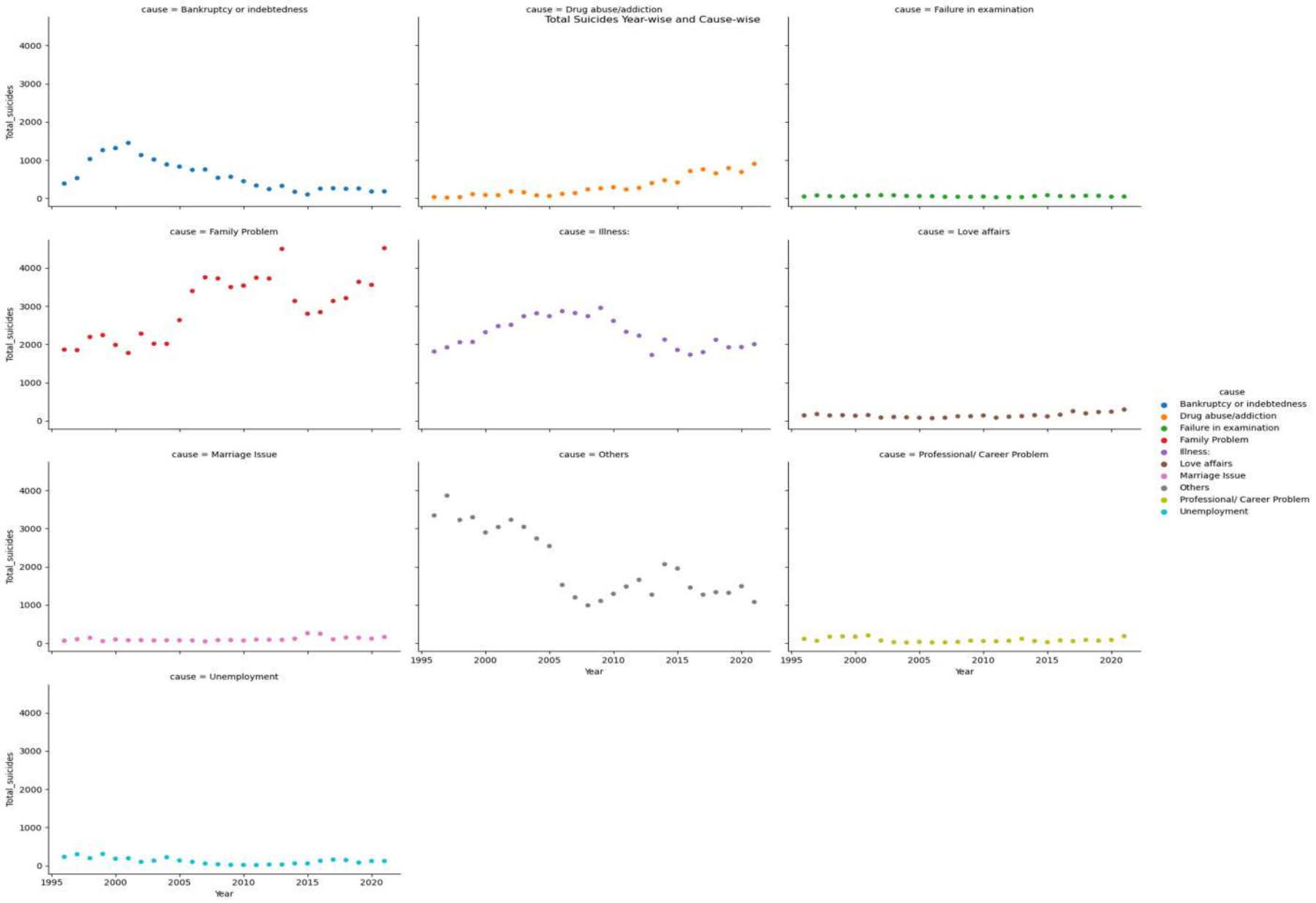
Around 34% of the total suicides committed during the study period (1996-2021) is conducted due to family problem and next is due to illness (26%). While analysing male & female suicides separately, it is noticed that 34% of male suicides were committed due to family problem and 25% is due to illness. In case of female it is 34% and 29% respectively.





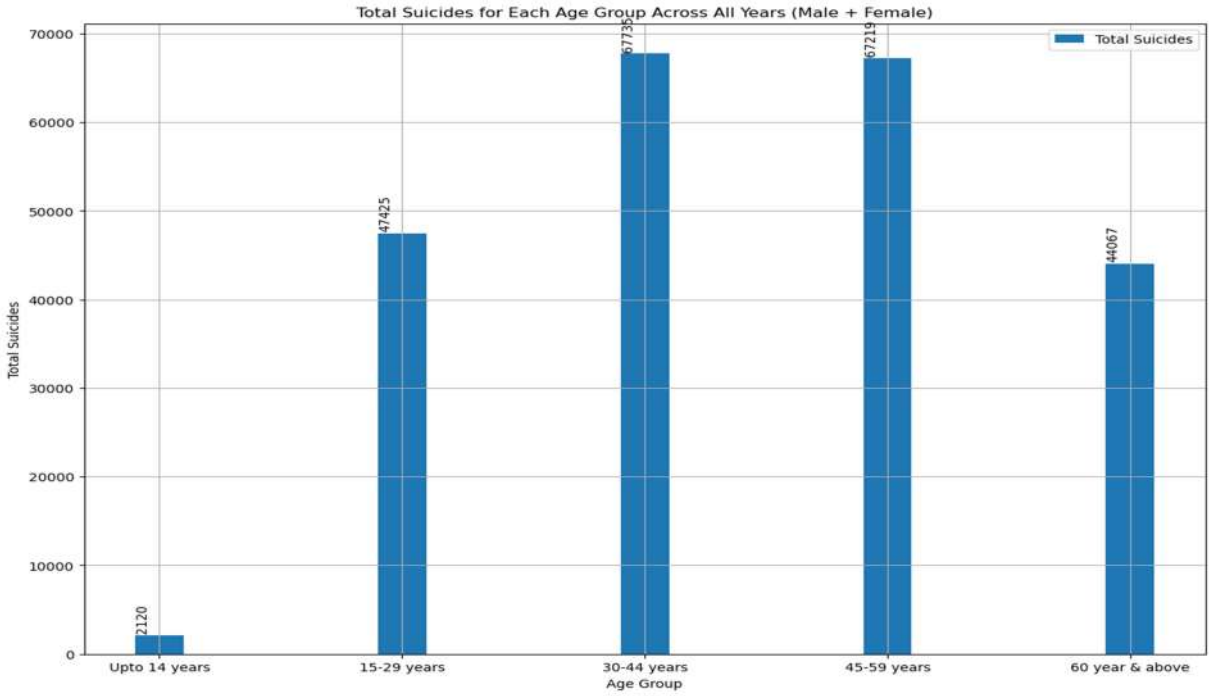


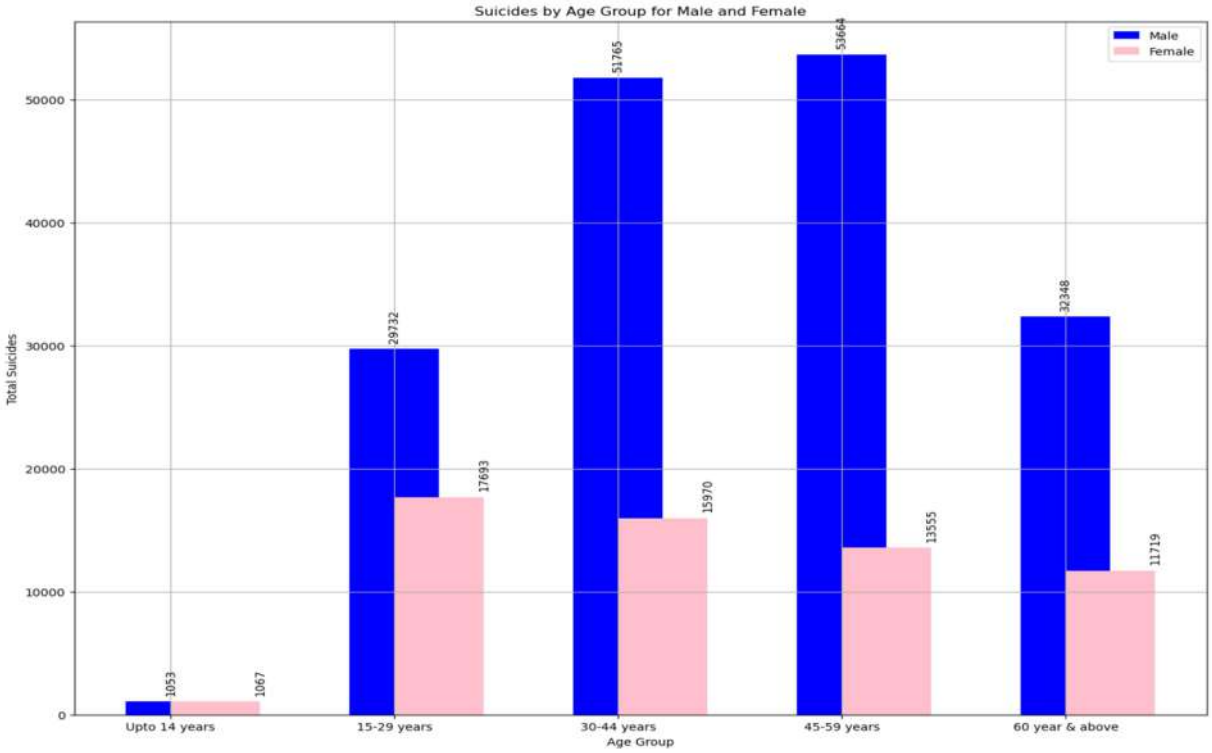
Cause wise Trends

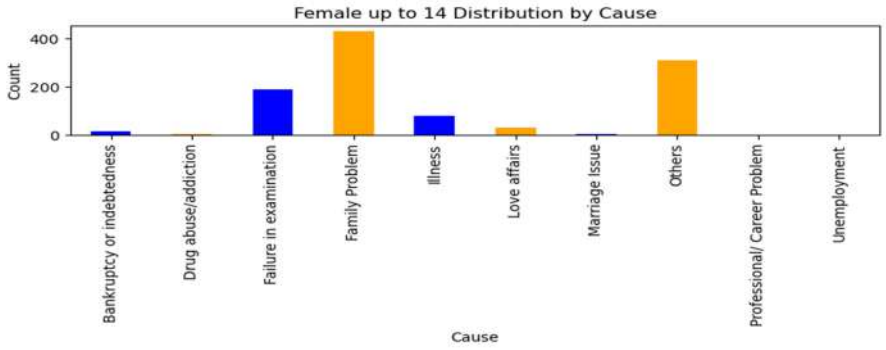


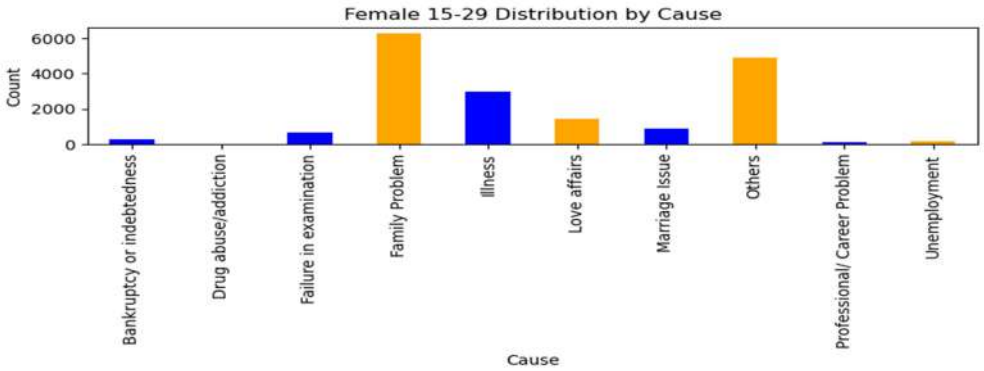
Age Group-wise Analysis

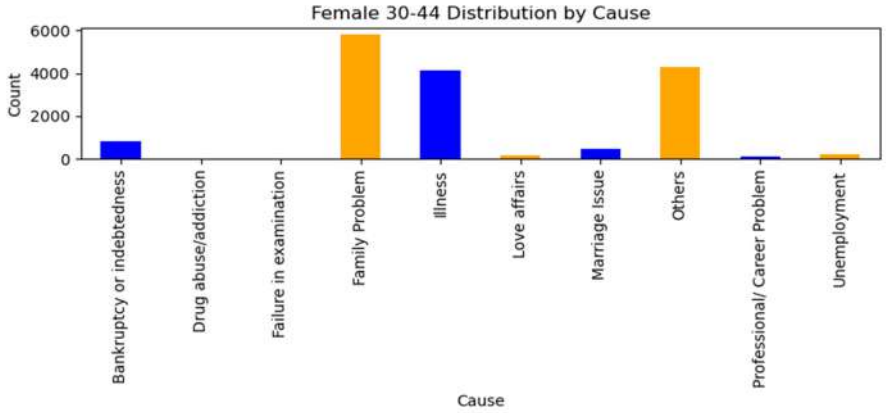
It is examined that the age group committing more suicides are 30-44 & 45-59. In case of male it is 45-59 and in case of female it is 15-29.

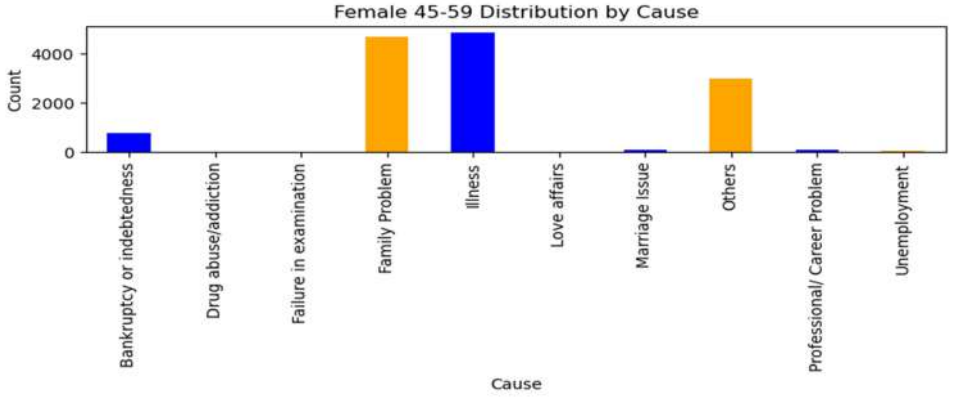


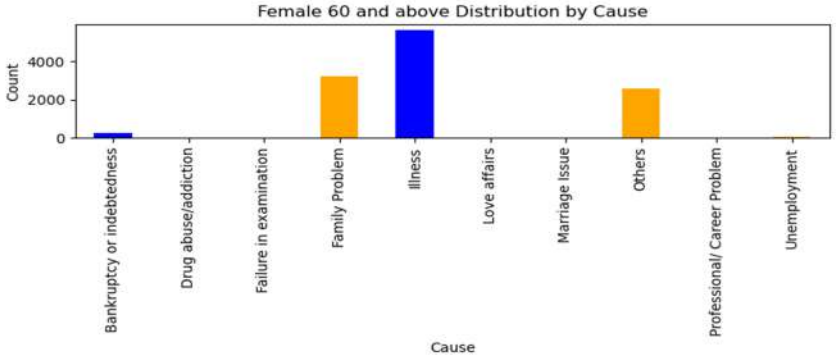


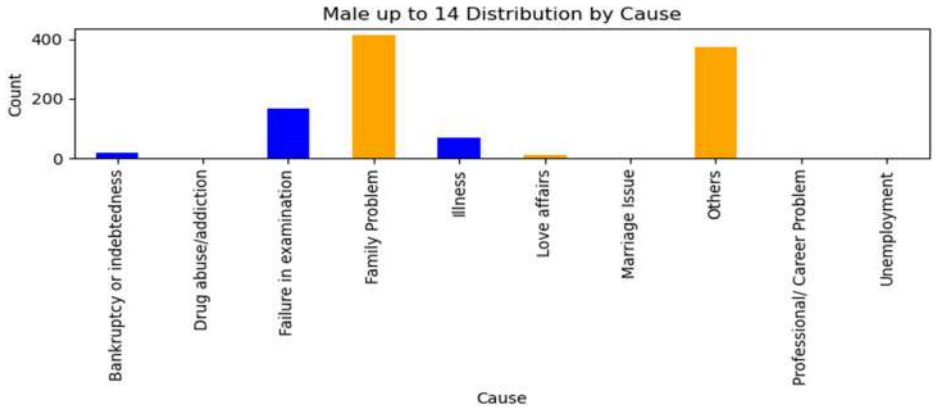

Female up to 14 Distribution by Cause
Count
400
200
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

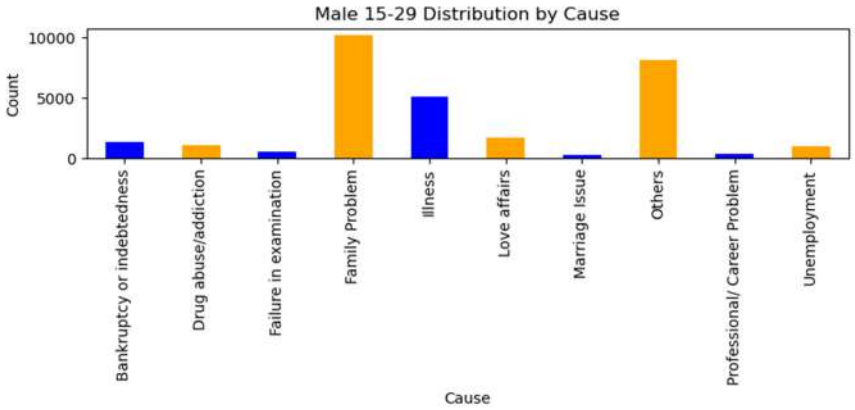

Female 15-29 Distribution by Cause
Count
6000
4000
2000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

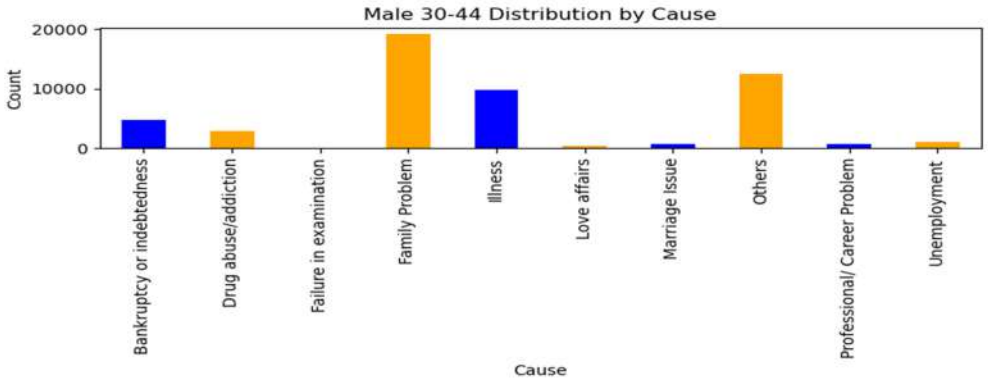

Female 30-44 Distribution by Cause
Count
6000
4000
2000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

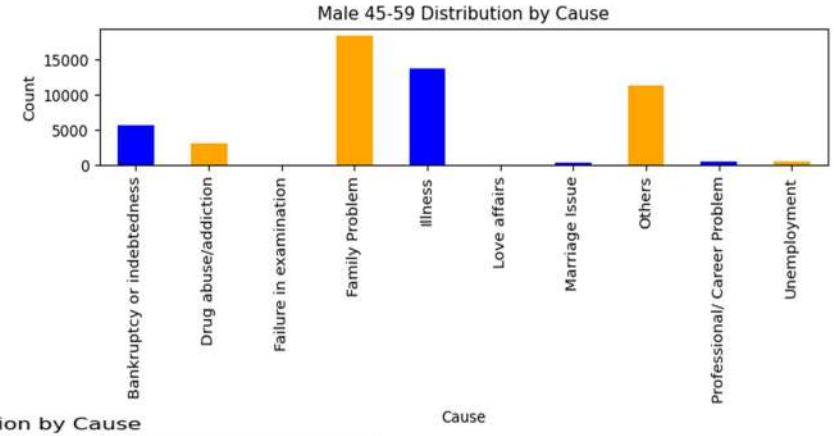

Female 45-59 Distribution by Cause
Count
4000
2000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

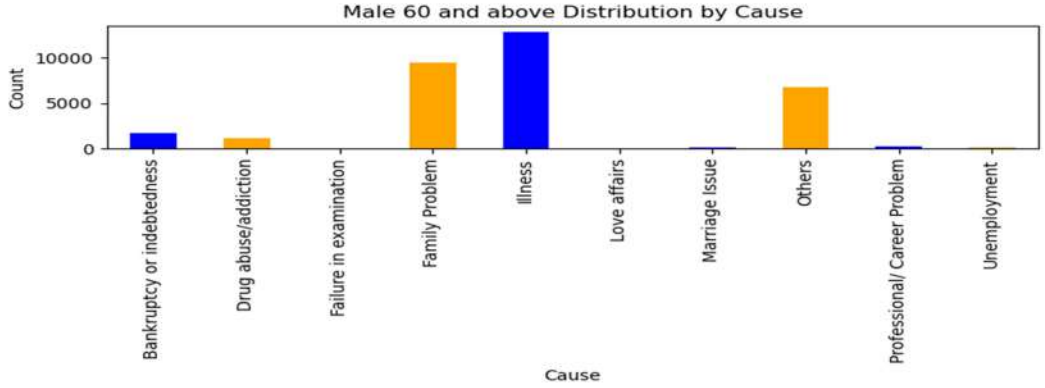

Female 60 and above Distribution by Cause
Count
4000
2000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause


Male up to 14 Distribution by Cause
Count
400
200
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause


Male 15-29 Distribution by Cause
Count
10000
5000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

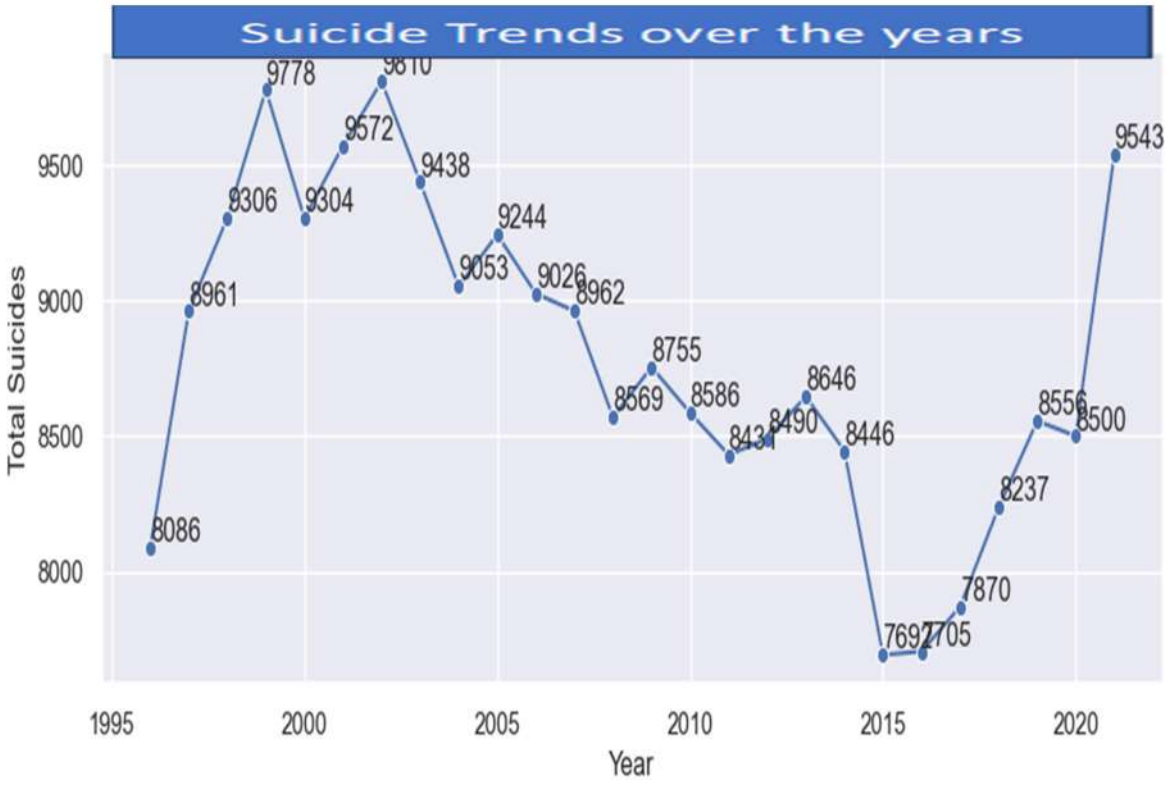

Male 30-44 Distribution by Cause
Count
20000
10000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

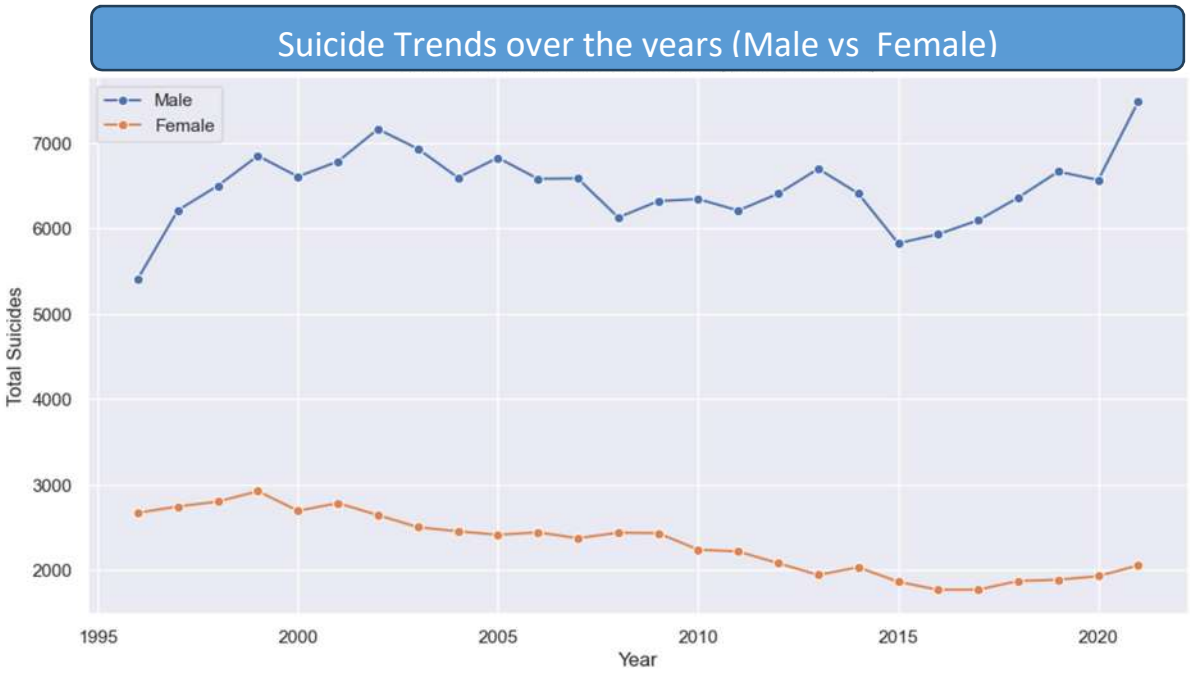

Male 45-59 Distribution by Cause
Count
15000
10000
5000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

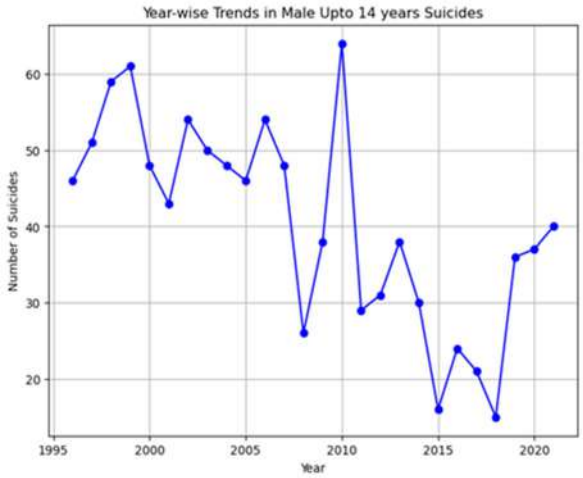

Male 60 and above Distribution by Cause
Count
10000
5000
0
Bankruptcy or indebtedness
Drug abuse/addiction
Failure in examination
Family Problem
Illness
Love affairs
Marriage Issue
Others
Professional/ Career Problem
Unemployment
Cause

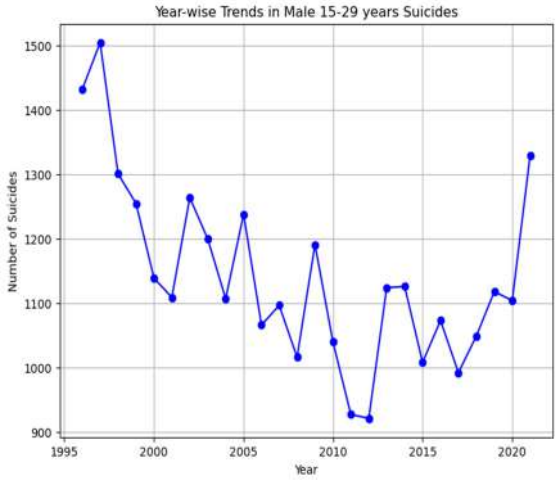
Year-wise Analysis

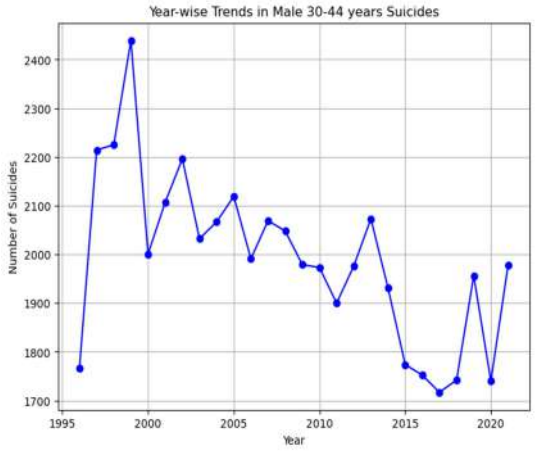
While analysing the yearwise suicides from 1996 to 2021 it is found that there is no notable trend in total suicides. More Suicides are committed in the year 2002 (9810) & Least Suicides are committed in the year 2015 (7692).

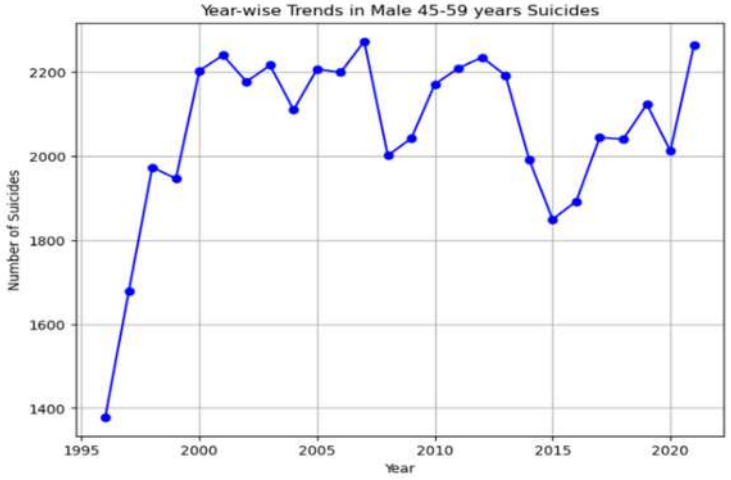


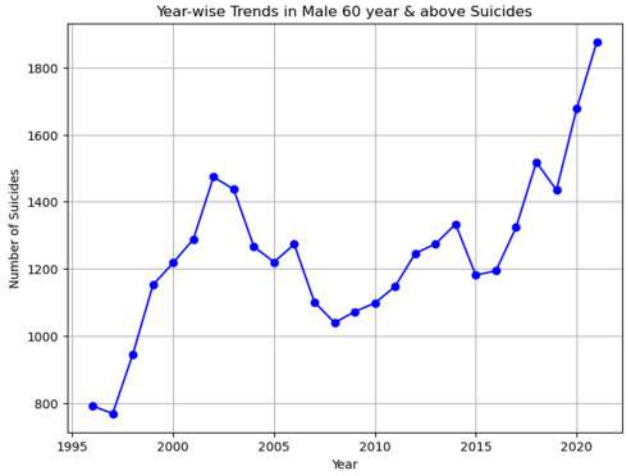


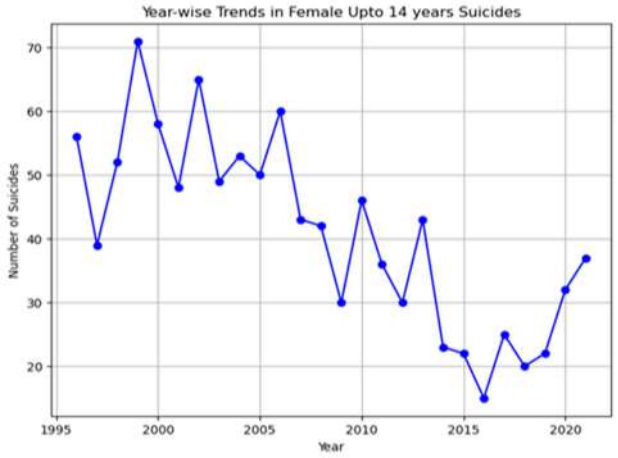

Year-wise Trends in Male Upto 14 years Suicides
Number of Suicides
Year

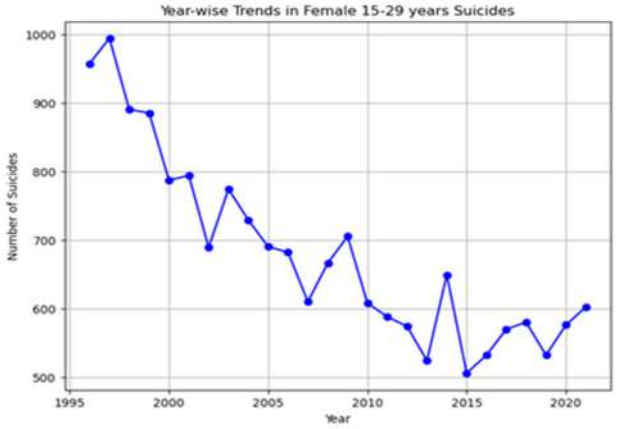

Year-wise Trends in Male 15-29 years Suicides
Number of Suicides
Year

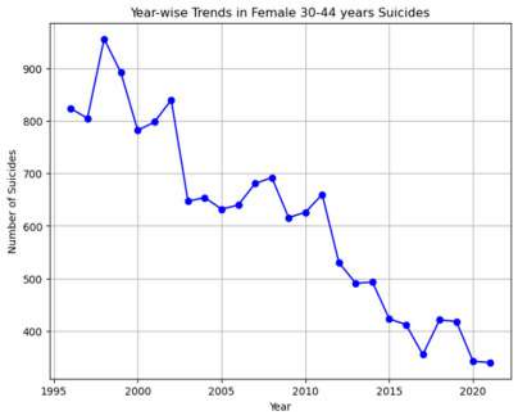

Year-wise Trends in Male 30-44 years Suicides
Number of Suicides
Year

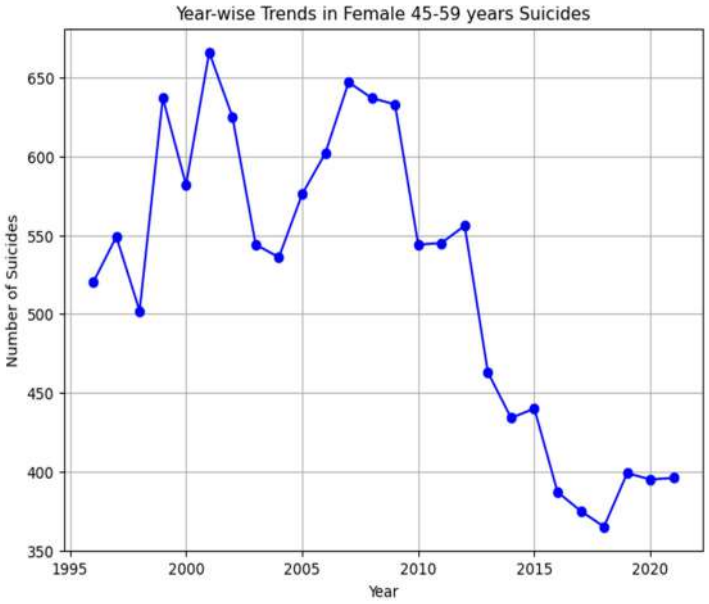

Year-wise Trends in Male 45-59 years Suicides
Number of Suicides
Year

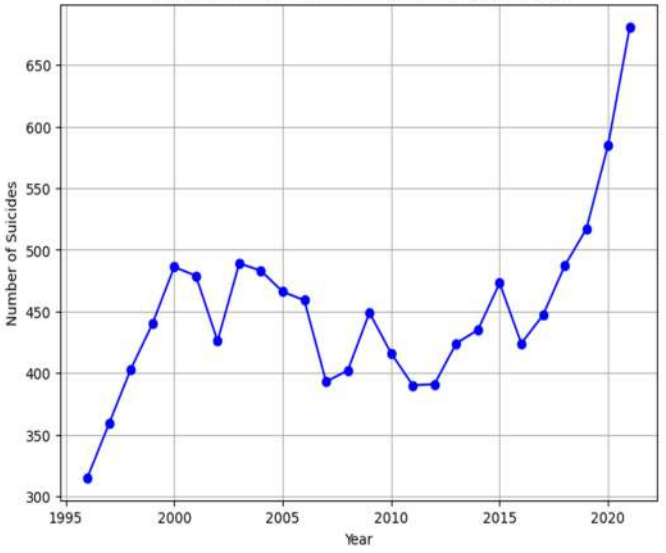

Year-wise Trends in Male 60 year & above Suicides
Number of Suicides
Year

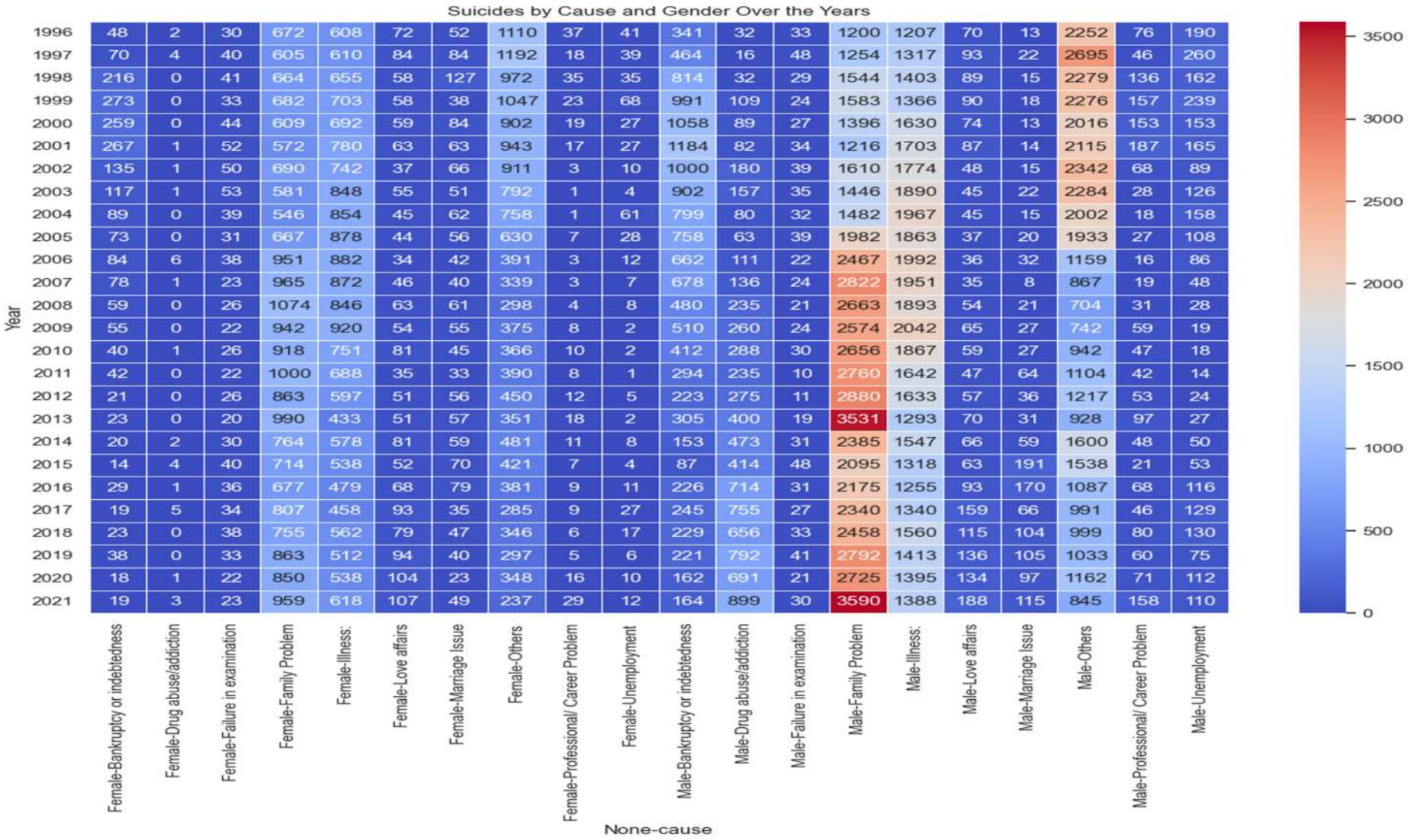

Year-wise Trends in Female Upto 14 years Suicides
Number of Suicides
Year


Year-wise Trends in Female 15-29 years Suicides
Number of Suicides
Year


Year-wise Trends in Female 30-44 years Suicides
Number of Suicides
Year


Year-wise Trends in Female 45-59 years Suicides
Number of Suicides
Year


Year-wise Trends in Female 60 year & above Suicides
Number of Suicides
Year

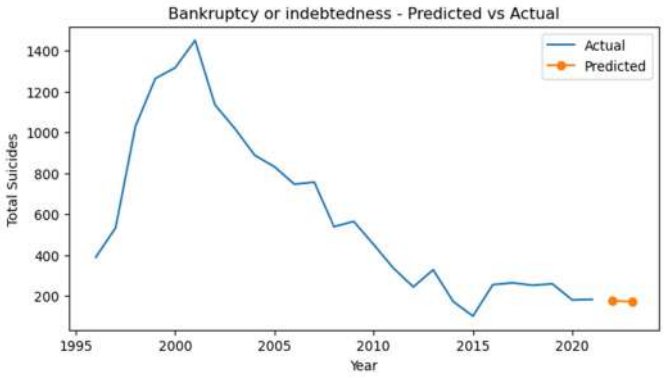

Suicides by Cause and Gender Over the Years
Year
1996 48 2 30 672 608 72 52 1110 37 41 341 32 33 1200 1207 70 13 2252 76 190
1997 70 4 40 605 610 84 84 1192 18 39 464 16 48 1254 1317 93 22 2695 46 260
1998 216 0 41 664 655 58 127 972 35 35 814 32 29 1544 1403 89 15 2279 136 162
1999 273 0 33 682 703 58 38 1047 23 68 991 109 24 1583 1366 90 18 2276 157 239
2000 259 0 44 609 692 59 84 902 19 27 1058 89 27 1396 1630 74 13 2016 153 153
2001 267 1 52 572 780 63 63 943 17 27 1184 82 34 1216 1703 87 14 2115 187 165
2002 135 1 50 690 742 37 66 911 3 10 1000 180 39 1610 1774 48 15 2342 68 89
2003 117 1 53 581 848 55 51 792 1 4 902 157 35 1446 1890 45 22 2284 28 126
2004 89 0 39 546 854 45 62 758 1 61 799 80 32 1482 1967 45 15 2002 18 158
2005 73 0 31 667 878 44 56 630 7 28 758 63 39 1982 1863 37 20 1933 27 108
2006 84 6 38 951 882 34 42 391 3 12 662 111 22 2467 1992 36 32 1159 16 86
2007 78 1 23 965 872 46 40 339 3 7 678 136 24 2822 1951 35 8 867 19 48
2008 59 0 26 1074 846 63 61 298 4 8 480 235 21 2663 1893 54 21 704 31 28
2009 55 0 22 942 920 54 55 375 8 2 510 260 24 2574 2042 65 27 742 59 19
2010 40 1 26 918 751 81 45 366 10 2 412 288 30 2656 1867 59 27 942 47 18
2011 42 0 22 1000 688 35 33 390 8 1 294 235 10 2760 1642 47 64 1104 42 14
2012 21 0 26 863 597 51 56 450 12 5 223 275 11 2880 1633 57 36 1217 53 24
2013 23 0 20 990 433 51 57 351 18 2 305 400 19 3531 1293 70 31 928 97 27
2014 20 2 30 764 578 81 59 481 11 8 153 473 31 2385 1547 66 59 1600 48 50
2015 14 4 40 714 538 52 70 421 7 4 87 414 48 2095 1318 63 191 1538 21 53
2016 29 1 36 677 479 68 79 381 9 11 226 714 31 2175 1255 93 170 1087 68 116
2017 19 5 34 807 458 93 35 285 9 27 245 755 27 2340 1340 159 66 991 46 129
2018 23 0 38 755 562 79 47 346 6 17 229 656 33 2458 1560 115 104 999 80 130
2019 38 0 33 863 512 94 40 297 5 6 221 792 41 2792 1413 136 105 1033 60 75
2020 18 1 22 850 538 104 23 348 16 10 162 691 21 2725 1395 134 97 1162 71 112
2021 19 3 23 959 618 107 49 237 29 12 164 899 30 3590 1388 188 115 845 158 110
Female-Bankruptcy or indebtedness
Female-Drug abuse/addiction
Female-Failure in examination
Female-Family Problem
Female-Illness:
Female-Love affairs
Female-Marriage Issue
Female-Others
Female-Professional/ Career Problem
Female-Unemployment
Male-Bankruptcy or indebtedness
Male-Drug abuse/addiction
Male-Failure in examination
Male-Family Problem
Male-Illness:
Male-Love affairs
Male-Marriage Issue
Male-Others
Male-Professional/ Career Problem
Male-Unemployment
None-cause
3500
3000
2500
2000
1500
1000
500
0

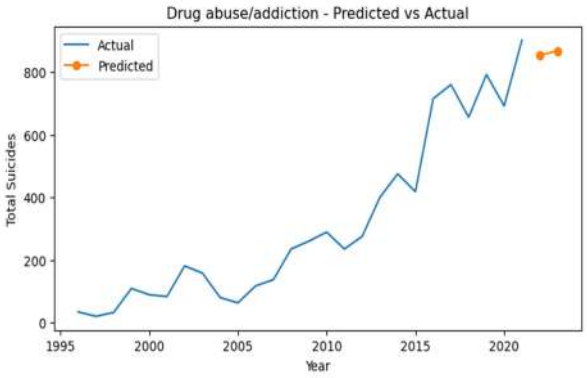
Prediction of Suicides

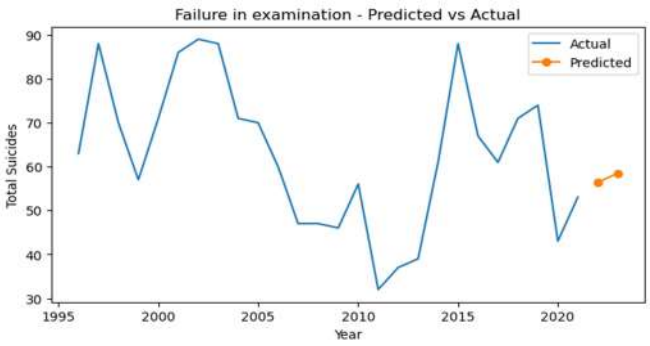
We experimented with various machine learning models for suicide rate estimation, but encountered significant challenges with low accuracy, impeding our progress. Nevertheless, we have opted for the Autoregressive Integrated Moving Average (ARIMA) model, which has demonstrated comparatively good, enabling us to proceed with our prediction efforts.

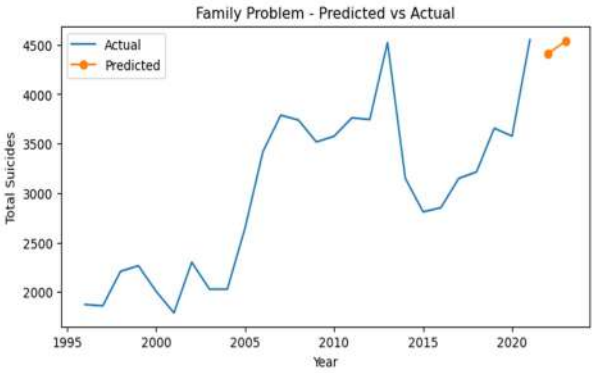
Prediction of Suicides using Time Series ARIMA Model

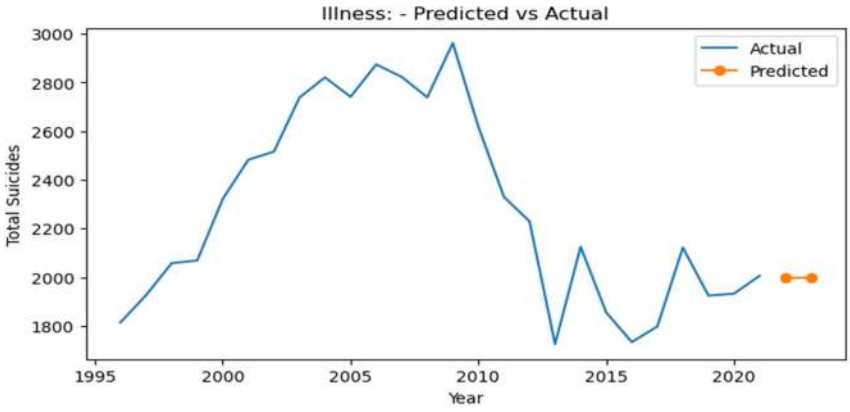
Cause	Actual Suicides-2020	Actual Suicides-2021	Predicted Suicides-2022	Predicted Suicides-2023
Bankruptcy or indebtedness	180	183	176	172
Drug abuse/addiction	692	902	854	868
Failure in examination	43	53	56	58
Family Problem	3575	4549	4407	4535
Illness:	1933	2006	1996	1999
Love affairs	238	295	277	282
Marriage Issue	120	164	153	148
Others	1510	1082	1065	1060
Professional/ Career Problem	87	187	196	188
Unemployment	122	122	132	124
Total Suicides	**8500**	**9543**	**9311**	**9436**

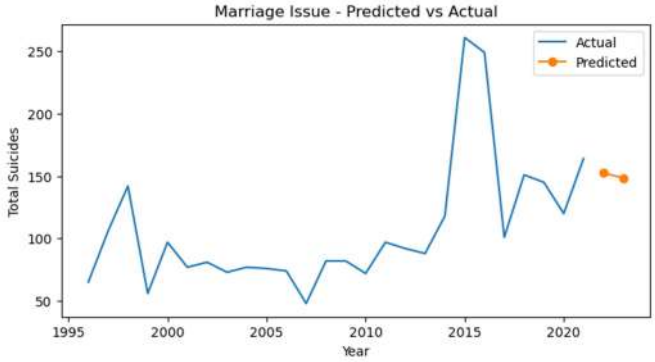

Bankruptcy or indebtedness - Predicted vs Actual
Actual
Predicted
Total Suicides
Year

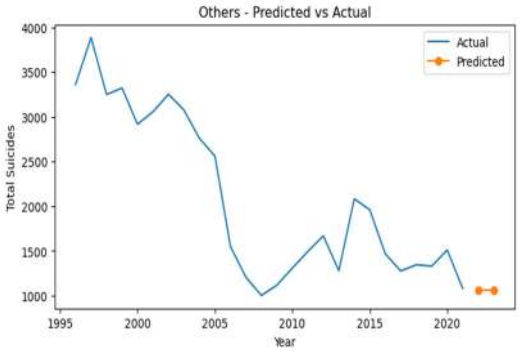

Drug abuse/addiction - Predicted vs Actual
Actual
Predicted
Total Suicides
Year

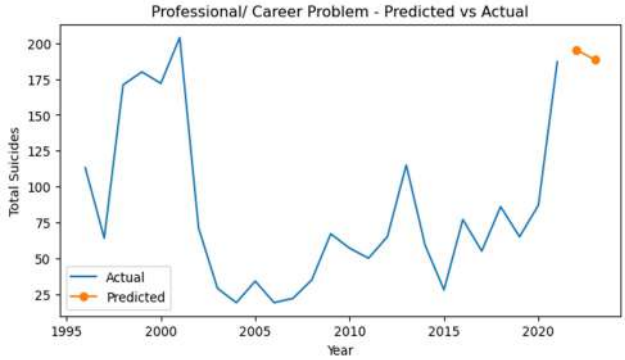

Failure in examination - Predicted vs Actual
Actual
Predicted
Total Suicides
Year

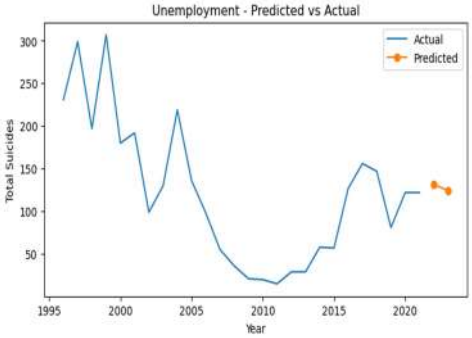

Family Problem - Predicted vs Actual
Actual
Predicted
Total Suicides
Year


Illness: - Predicted vs Actual
Actual
Predicted
Total Suicides
Year


Marriage Issue - Predicted vs Actual
Actual
Predicted
Total Suicides
Year
250
200
150
100
50
1995
2000
2005
2010
2015
2020


Others - Predicted vs Actual
Actual
Predicted
Total Suicides
Year
4000
3500
3000
2500
2000
1500
1000
1995
2000
2005
2010
2015
2020


Professional/ Career Problem - Predicted vs Actual
Actual
Predicted
Total Suicides
Year
200
175
150
125
100
75
50
25
1995
2000
2005
2010
2015
2020


Unemployment - Predicted vs Actual
Actual
Predicted
Total Suicides
Year
300
250
200
150
100
50
1995
2000
2005
2010
2015
2020

Major Findings of the Study

- 74% of Suicides in Kerala are committed by men.
- Major cause of suicides in Kerala is family problem & illness.
- Age group committing more suicide is 30-44 & 45-59.
- Male of age group 45-59 are committing more suicides, in case of female it is 15-29.
- Major cause of suicide of men upto age 59 is due to family Problem but above 60 years is due to illness.
- Major cause of suicides of female upto 44 years is family problem & above 45 is illness.
- There are no Specific trends in Suicides over the Years, but there is slight increase in suicide of men.
- In 2015 there is a decline in suicide of all male age groups.
- There is a decline in suicide of female age group 30-44 & increase in age 60 above.
- Suicide due to bankruptcy & indebtedness is declining.
- Most affected group of suicide by usage of drugs is male of age group 45-59 & 30-44.
- Suicide due to drug abuse is very low in case of female (Below 10 each year).
- Male suicide due to drug abuse is increasing.

Conclusion

Government should consider comprehensive mental health programs, strengthening mental health infrastructure, and promoting awareness campaigns targeting different age and gender demographics. Proper interventions should address the root causes identified in the study, including family problems, illness, and the rise in male suicides due to drug abuse. Collaborative efforts between government agencies, healthcare professionals, and community organizations are essential for effective suicide prevention initiatives in the state. It is seen that suicide is done due to one's mental disorder or physical disability, as a result of excessive alcohol/drug use, due to excess of mental tension, due to love disappointment, or due to a disturbance in economic status. If there is a strong intervention of the government to adopt ways to follow the lifestyle, we can save ourselves from the big challenge of suicide that is facing the modern society. While this project was in progress, an article was published in the editorial column of Mathrubhumi daily regarding the huge increase in male suicide in Kerala which reveals the relevance of this project. We still have a long way to go. We need to have a mentally and physically healthy generation. May this report increase the attention of the government on this issue. "Creating Hope Through Action" is the triennial theme of the World Health Organisation for the World Suicide Prevention Day from 2021-2023. This theme serves as a powerful call to action and reminder that there is an alternative to suicide and that through our actions we can encourage hope and strengthen prevention.

"When you feal like giving up, just remember the reason why you held on for so long"

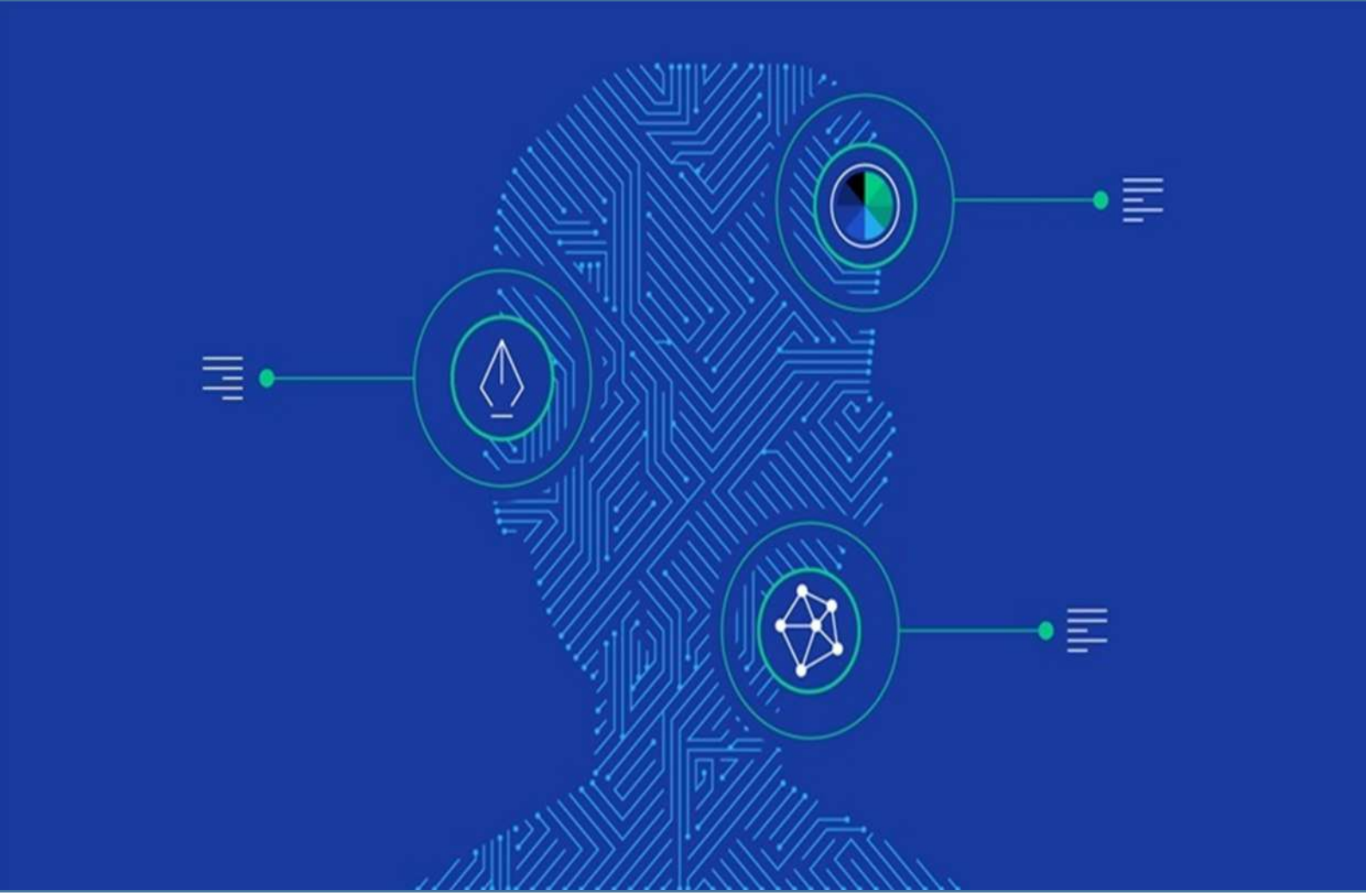
"Because if you kill yourself, you're also going to kill the people who love you"

References

"Suicides in India - Basic Data Analysis" - By Shavilya Rajput

"Prediction of Suicide Causes in India using Machine Learning"- By Imran Amin, Sobia Syed

"Accidental Deaths & Suicides in India" – Published by National Crime Records Bureau (NCRB)



Analysis and Prediction of Monthly Percapita Expenditure (MPCE) of a Family in Kerala

Submitted by

Sri. Sreekumar G,

Research Officer

Introduction

The Household Consumer Expenditure Surveys of National Statistical Office (NSO) are the primary source of data on various indicators of level of living, pattern of consumption and wellbeing of individuals at national and state levels. NSO and DES conducted NSS 68th round survey on Household Consumer Expenditure with 2:3 matching ratio sample size basis during July 2011 to June 2012.

The NSS consumer expenditure survey aims at generating estimates of household Monthly Per Capita Consumer Expenditure (MPCE) and the distribution of households and persons over the MPCE range separately for States and Union Territories, and for different socio-economic groups. These indicators are among the most important measures of the level of living of the relevant domains of the population. The distribution of MPCE highlights the differences in level of living of the different segments of the population and is an effective tool to study the prevalence of poverty and inequality. These numbers thus enable the apex planning and decision-making

Objectives

The MPCE play a critical role in the estimation of poverty line and living standard of the people in the state. Moreover the vigilance department usually request to the DES to make available the average expenditure of some families involved in cases. In this context, this study has more importance.

The main objectives of the study.

- To analyse the data of household consumer expenditure.
- To build a system capable of finding underlying patterns in the data.
- Predict the MPCE.

Literate review

A similar study has been done in Forecasting and analysing the real estate market can evaluate the stability of the real estate market. On the one hand, it can facilitate the government to carry out macro control on house prices and maintain the healthy and stable development of China's national economy. On the other hand, the prediction of house prices can also provide a certain basis for real estate investors to formulate investment strategies and avoid losses.[4].

Another paper [5] discusses various work by different researchers on linear regression and polynomial regression and compare their performance using the best approach to optimize prediction and precision.

Through this project, I try to develop a machine learning model for predicting the MPCE of a family in Kerala.

Programming Language

I use ***Python*** as my programming language. It is a versatile, high-level programming language known for its readability and ease of use. It supports multiple programming paradigms, including procedural, object-oriented and functional programming. Python's extensive standard library and active community make it suitable for diverse applications, from web development and data analysis to artificial intelligence and automation.

Steps

- Understand the Data
- Handle Missing Values
- Remove Duplicates
- Correct Data Types
- Standardize/Normalize Data
- Handle Outliers
- Address Inconsistencies
- Check for Accuracy

Libraries and Tools used

There are a vast number of tools and libraries available for machine learning. Here are some of the Libraries used for the study.

1. **Pandas:** A data manipulation library in Python that is often used to clean, transform, and manipulate data for machine learning.
2. **NumPy:** A Python library for numerical computing that is often used for linear algebra and array operations in machine learning.
3. **Matplotlib:** A Python plotting library that is often used to create visualizations of data for machine learning.
4. **Seaborn:** A data visualization library in Python that is built on top of Matplotlib and provides a high-level interface for creating statistical graphics.
5. **Scikit-learn**: A popular machine learning library in Python for tasks such as classification, regression, and clustering.
6. **Statsmodels -**library for statistical modeling and analysis

Data Set used

Pooled Data of 68th Round Socio-Economic Survey conducted by National Sample Survey (NSS) [July 2011 to June 2012] and dummy data. The model of data is given below. It has 6723 rows and 6 columns.

Table :1

	HH Size	HH type	Religion	Social Group	MPCE in Rs
0	6	9	2	3	2602.36
1	4	2	3	9	1642.00
2	5	5	2	3	1646.85
3	7	1	2	3	1569.71
4	6	9	2	3	2610.67
...	...	...	...	...	...
6719	4	1	1	3	2887.00
6720	3	2	1	3	4144.00
6721	4	3	1	3	2403.75
6722	4	3	1	3	1808.50
6723	2	2	1	3	1374.00

Missing values and duplicates are removed from the data set.

Codes used in the data

Housed type:

Self-employed in agriculture - 1
Self-employed in non-agriculture - 2
Regular wage/salary earning - 3
Casual labour in agriculture - 4
Casual labour in non-agriculture - 5
Others-9

Religion:

Hinduism-1
Islam-2
Christianity-3
Sikhism-4
Jainism-5
Budhism-6
Others-9

Social group:

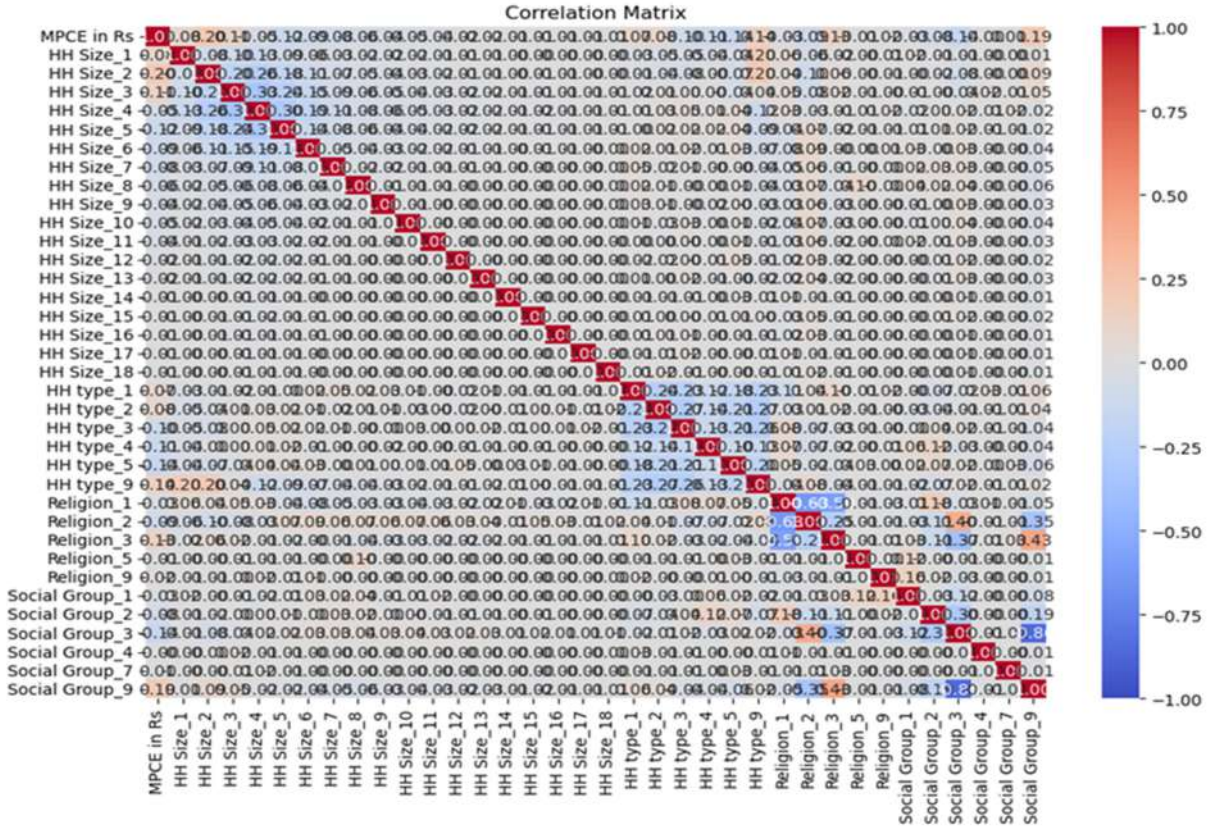
Scheduled tribes - 1
Scheduled castes - 2
Other backward classes - 3
Others - 9

The household type, religion and social group are all represented by codes, the system will be treated these as numerical values. So they are expanded along each code with two entries – ‘0 and 1’.

Table 2: The Model of the Expanded Columns

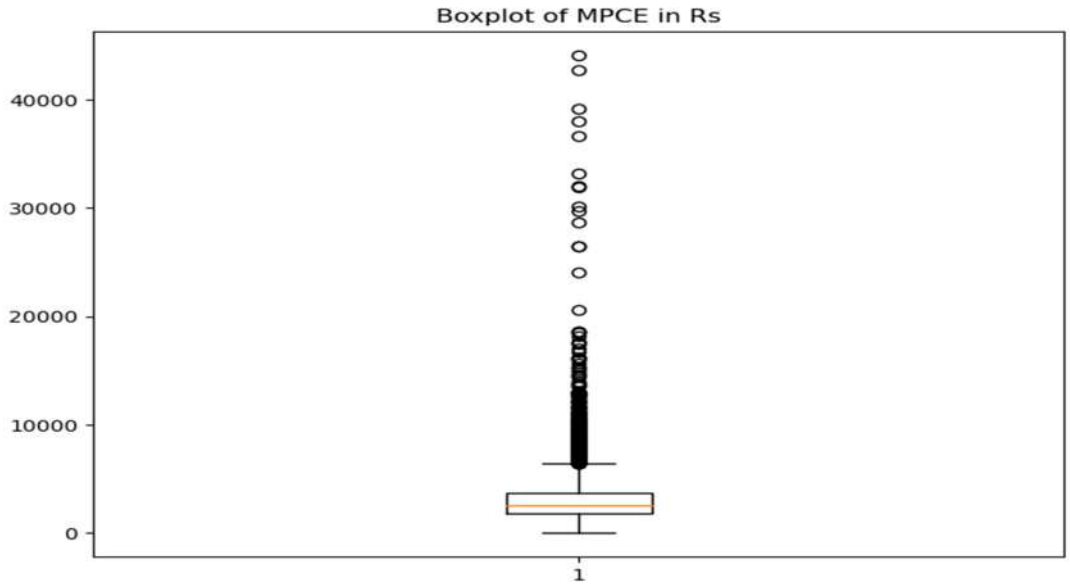
	MPCE in Rs	HH Size_1	HH Size_2	HH Size_3	HH Size_4	HH Size_5	HH Size_6	HH Size_7	HH Size_8	HH Size_9	...	Religion_2	Religion_3	Religion_5	Religion_9	Social Group_1	Social Group_2	Social Group_3	Social Group_4	Social Group_7	Social Group_9
0	2602	0	0	0	0	0	1	0	0	0	...	1	0	0	0	0	0	1	0	0	0
1	1642	0	0	0	1	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	1
2	1646	0	0	0	0	1	0	0	0	0	...	1	0	0	0	0	0	1	0	0	0
3	1569	0	0	0	0	0	0	1	0	0	...	1	0	0	0	0	0	1	0	0	0
4	2610	0	0	0	0	0	1	0	0	0	...	1	0	0	0	0	0	1	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
6719	2887	0	0	0	1	0	0	0	0	0	...	0	0	0	0	0	0	1	0	0	0
6720	4144	0	0	1	0	0	0	0	0	0	...										

Figure 1 : Correlation Matrix of Expanded Table Data



Handling Outliers

Figure 2 : Boxplot of Monthly Per Capita Expenditure



Outliers are data points that deviate significantly from the rest of the data in a dataset. Whether or not to remove outliers depends on the nature of the data and the goals of the analysis. In some cases, it might be appropriate not to remove outliers.

Here outliers represent valid and genuine variability in the data. Removing such outliers may result in the loss of important information.

Since we have a small sample size, removing outliers may significantly impact the representativeness of the data.

So we have to retain the outliers.

<u>Data at a Glance</u>

Figure: 3

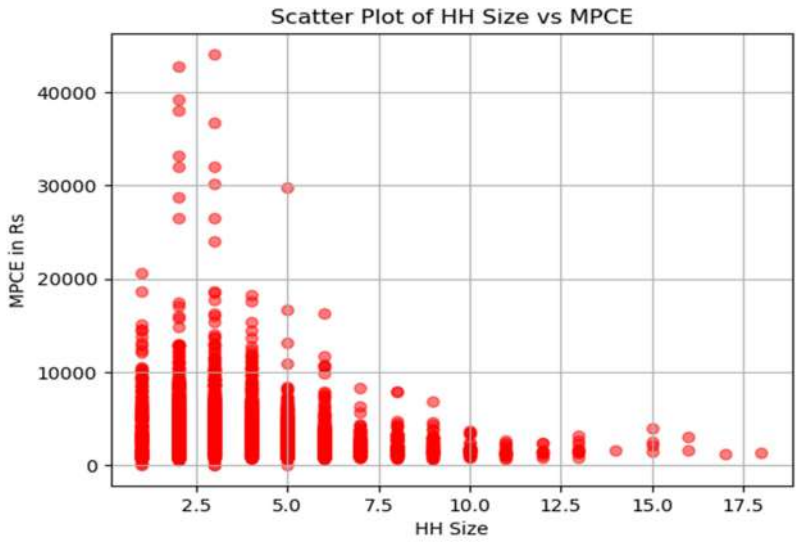


Figure: 4

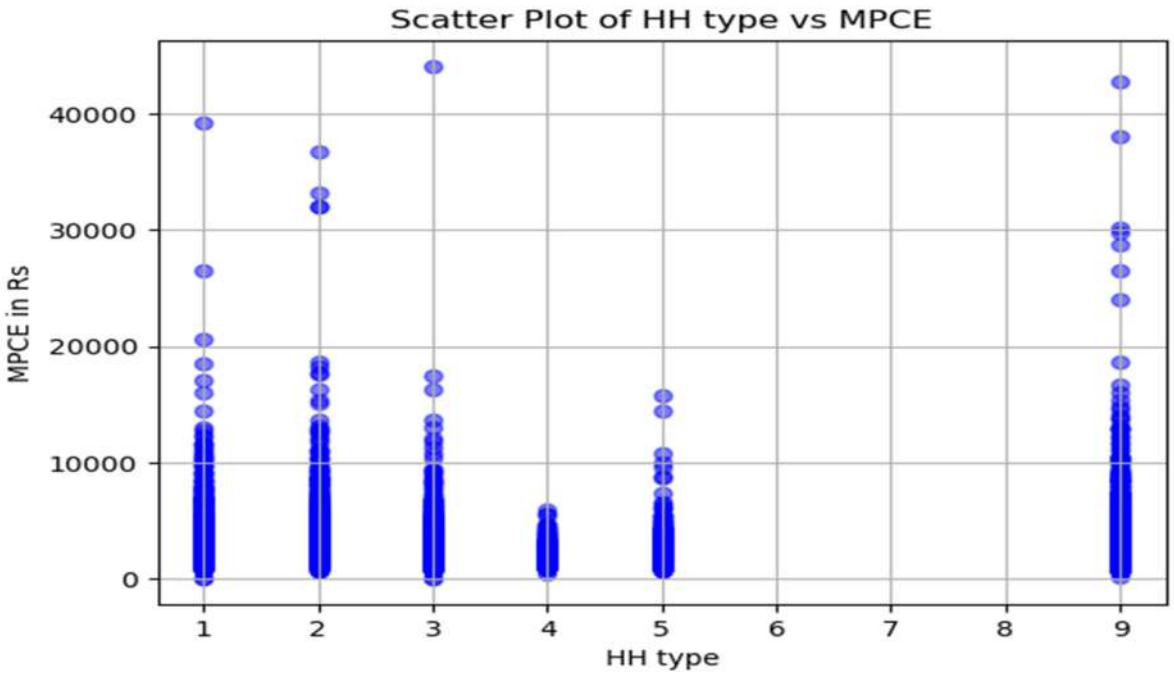
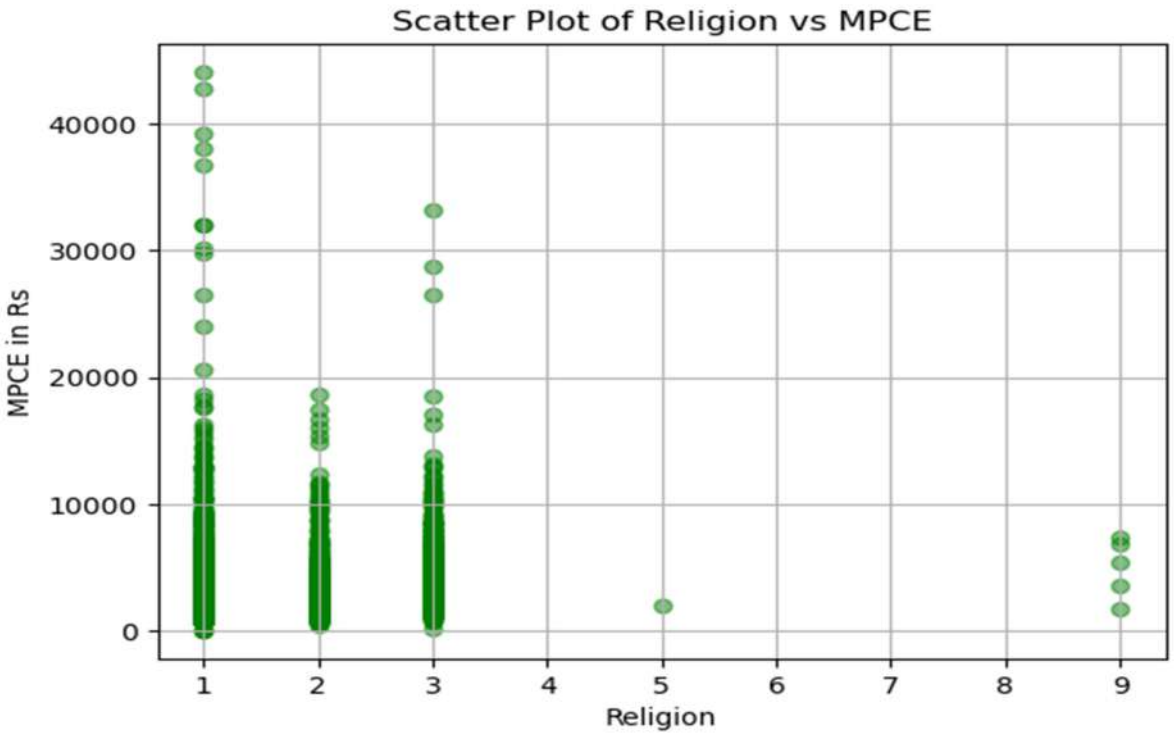


Figure: 5



The following machine learning algorithms are used.

Ordinary Least Square (OLS)

Ordinary Least Squares (OLS) regression is a statistical method used for estimating the relationship between one or more independent variables and a dependent variable.

OLS Regression Results

==========

Dep. Variable:	MPCE in Rs	R-squared:	0.151
Model:	OLS	Adj. R-squared:	0.147
Method:	Least Squares	F-statistic:	38.50
Date:	Mon, 18 Dec 2023	Prob (F-statistic):	1.70e-211
Time:	22:21:13	Log-Likelihood:	-61425.
No. Observations:	6724	AIC:	1.229e+05
Df Residuals:	6692	BIC:	1.231e+05
Df Model:	31		
Covariance Type:	non-robust		

==========

R-squared and Adjusted R-squared:

- **R-squared (R^2):** This value (0.151) represents the proportion of the variance in the dependent variable (MPCE in Rs) that is explained by the independent variables in the model. Approximately 15.1% of the variability in MPCE can be explained by the model.
- **Adjusted R-squared:** This is a modified version of R-squared that accounts for the number of predictors in the model. The adjusted R-squared (0.147) is slightly lower than R-squared but still provides a measure of goodness of fit.

F-statistic:

The F-statistic (38.50) is used for testing the overall significance of the regression model. A high F-statistic suggests that at least one independent variable is significantly related to the dependent variable. The F-statistic is relatively high.

Prob (F-statistic):

The p-value associated with the F-statistic (1.70e-211) is extremely low. This indicates that the overall model is statistically significant, meaning that there is strong evidence that at least one independent variable is related to the dependent variable.

In summary, the model has statistical significance (low p-value for F-statistic), but the R-squared value is relatively low (15.1%), suggesting that the model explains only a modest proportion of the variance in the dependent variable. Additionally, we may want to **explore potential improvements to the model**, such as adding relevant variables or considering alternative model specifications.

To get a better result I try the next model ***linear regression***

Linear Regression

Linear Regression establishes a relationship between dependent variable (Y) and one or more independent variables (X) using a best fit straight line (also known as regression line). It is represented by an equation Y=a+bX + e, where a is intercept, b is slope of the line and e is error term. This equation can be used to predict the value of target variable based on given predictor variable(s).

The difference between simple linear regression and multiple linear regression is that, multiple linear regression has (>1) independent variables, whereas simple linear regression has only 1 independent variable.

We have more than one independent variables, we used multiple linear regression which is expressed as follows.

Y=B0+B1X1+B2X2+B3X3+B4X4+Ui

Where Y=MPCE of the family

X1=Household size

X2=Household type

X3=Religion

X4=Social group

B0,B1,B2,B3,B4 are the Regression Coefficients

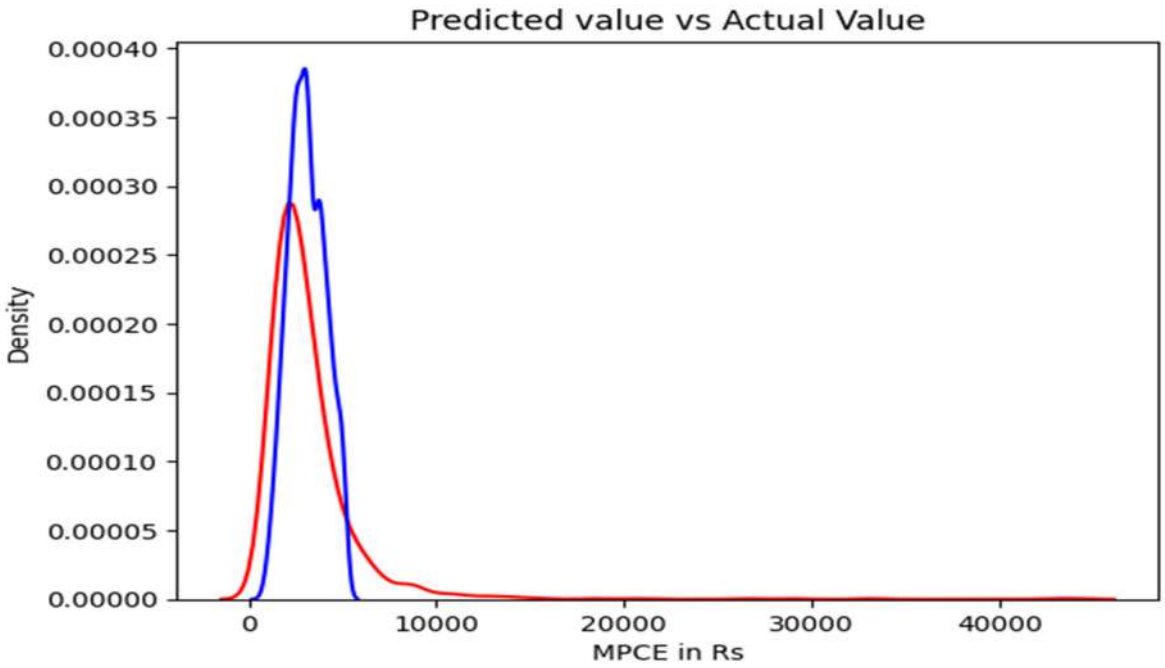
Ui=Disturbance term(Error term)

Data Partitioning

The Entire dataset is partitioned into 2 parts: 80% dataset is used for training the model and remaining 20% data is used for test the model.

Results and analysis

Figure: 6

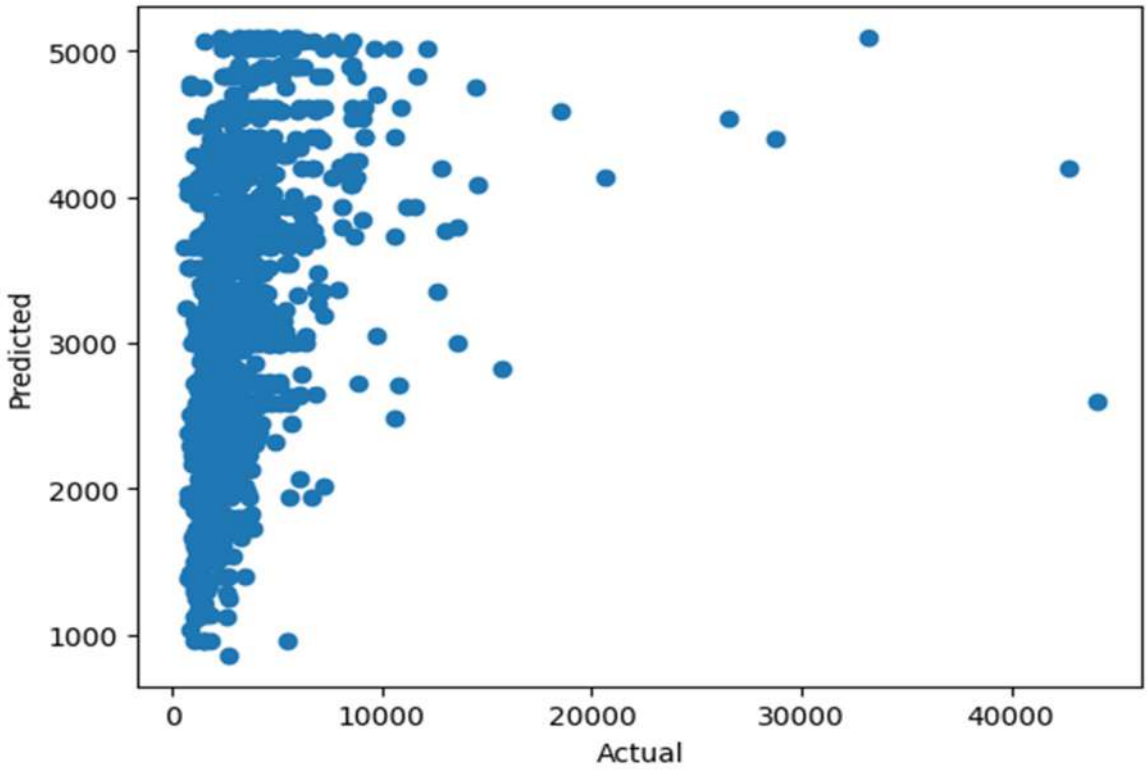


R-squared: 0.12167642964911363

Red: Predicted

Blue: Actual

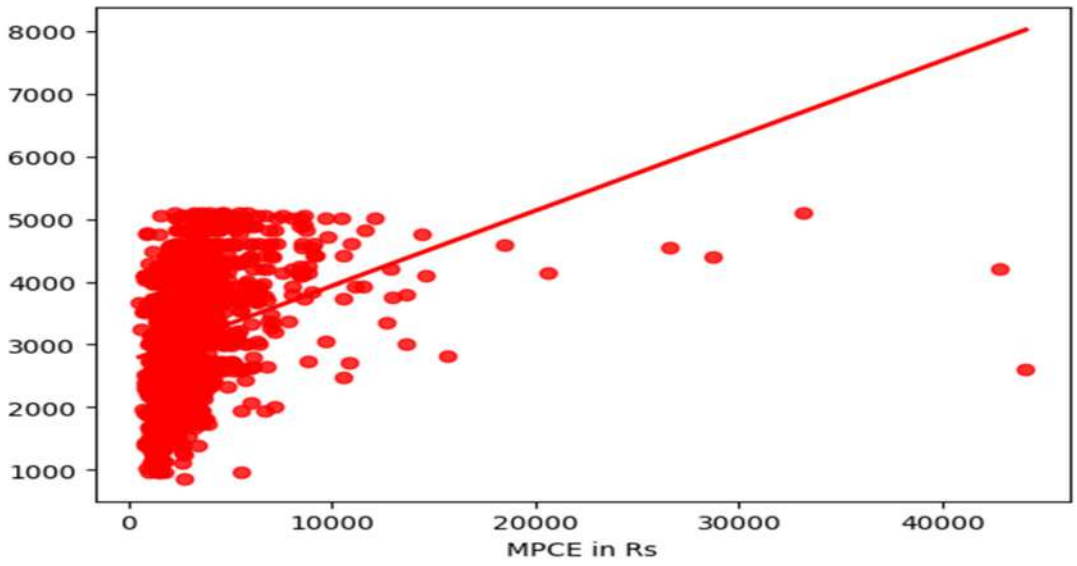
Figure 7 : The scatter plot between actual values and predicted values.



Regression plot of the model.

A regression plot is useful to understand the linear relationship between two parameters. It creates a regression line in-between those parameters and then plots a scatter plot of those data points.

Figure: 8



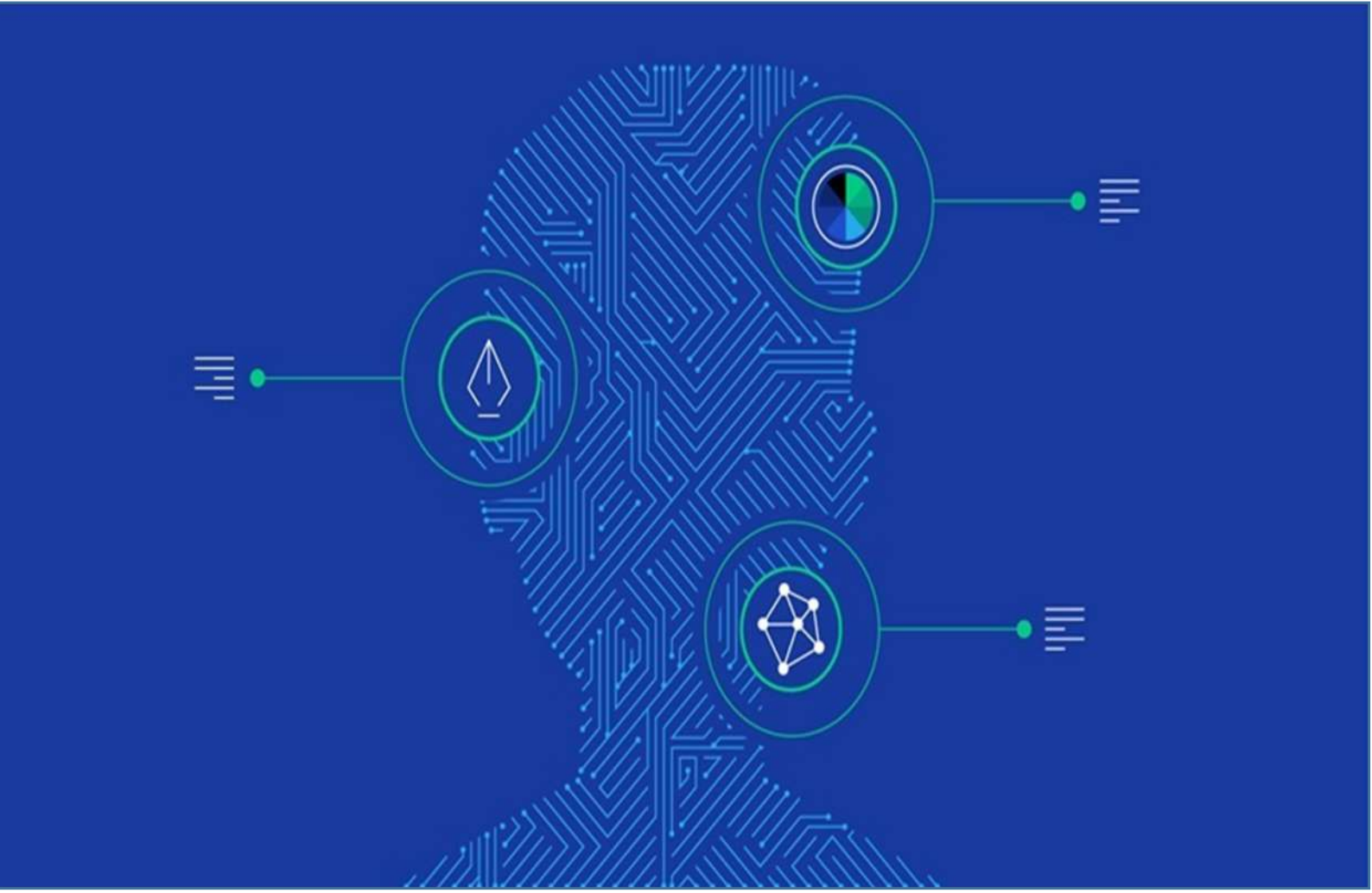
Conclusion

Now with the help of above Linear Regression model, we can predict the MPCE of a family if the characteristics (household size, household type, religion and social group) of that family are given.

The model is about 88% accurate to predict the MPCE. Statistical model of the project provides a useful tool for predicting the MPCE. It does not depend on any economic parameter. No such data are available from the socio-economic survey conducted by the NSO. So further set of data linked with economic conditions of the families are necessary for predicting MPCE which reveals the real picture of the expenditure of the families. The potential of machine learning techniques for predicting MPCE under variety of scenarios requires further research and analysis.

REFERENCE:

1. *Report on Household Consumer Expenditure in Kerala, Based on NSS 68th Round July 2011- June 2012 Central and State sample pooled data, Department of Economics & Statistics, Government of Kerala.*
2. *Report on Household Consumption of various Goods and Services in Kerala, (NSS 66th Round July 2009- June 2010) Department of Economics & Statistics, Government of Kerala.*
3. *Various websites.*
4. *Research on Prediction and Analysis of Real Estate Market Based on the Multiple Linear Regression Model, Sheng Bin , 2022*
5. *A review on Linear regression comprehensive in machine learning, Dastan Maulod, Adnan N AbdulAzeez 2020*



Times Series Forecasting Study for Death (1962 - 2020) using AI

Submitted by

Smt. Yamuna A.R,

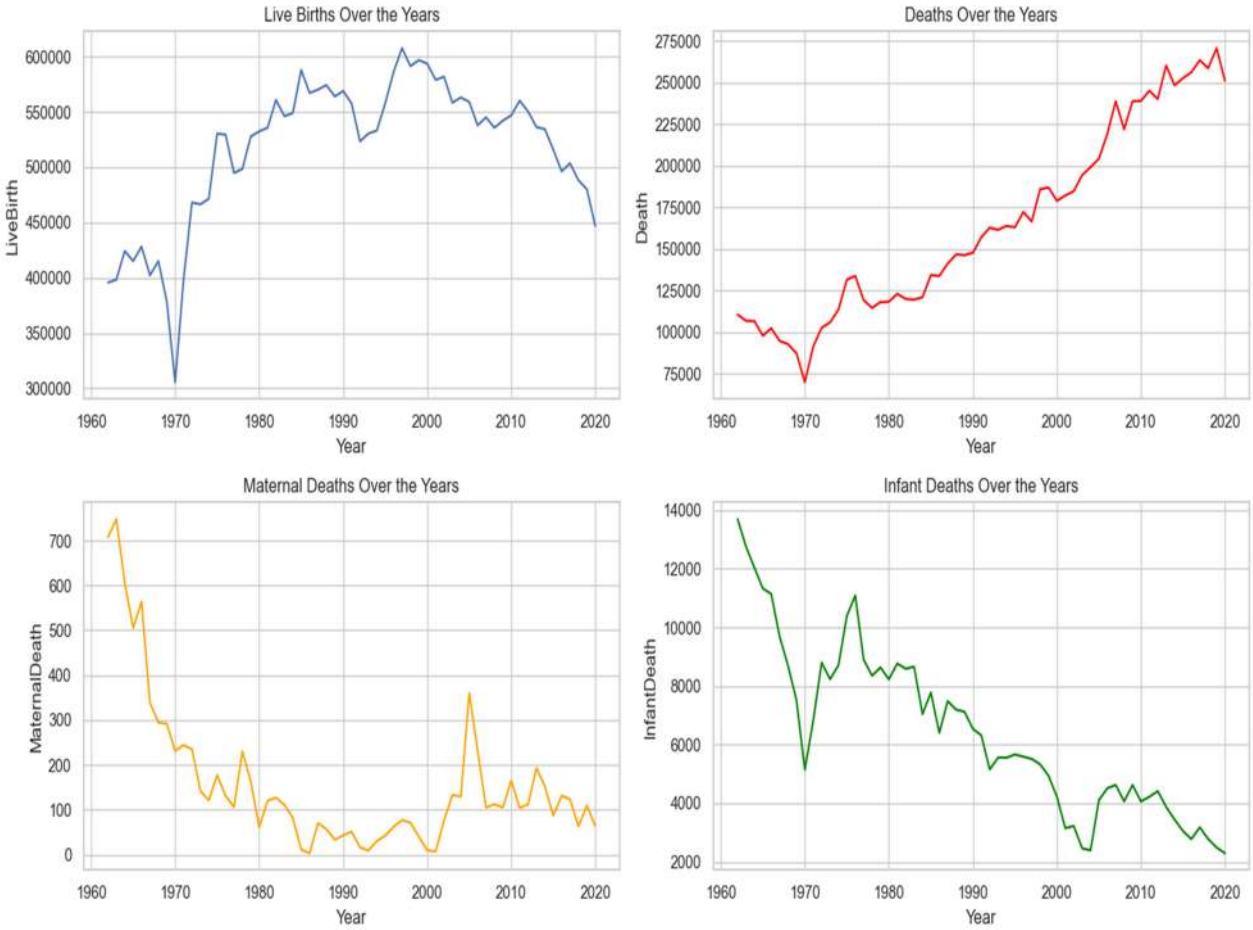
Deputy Director

1. Introduction:

Time series forecasting is a crucial aspect of data analysis, especially in the context of understanding and predicting trends in mortality rates over the years. In this study, we employ Artificial Intelligence (AI) techniques to analyze and forecast death data spanning from 1962 to 2020. The objective is to gain insights into historical patterns and trends and use AI models to predict future mortality rates. Predicting future deaths is a critical challenge in public health and demography. Understanding mortality trends helps governments, healthcare providers, and policymakers to allocate resources effectively.

2. Data Collection:

The dataset used in this study encompasses death statistics from the year 1962 to 2020. The data is sourced from civil registration database, and it includes various demographic indicators related to live births, deaths, maternal deaths and infant deaths from the year 1962 to 2020.



3. Preprocessing:

To prepare the data for time series forecasting, preprocessing steps are implemented. This includes handling missing values, converting data into a time series format, and addressing any anomalies or outliers detected during the EDA phase. For our time series forecasting analysis, we focus exclusively on the "Year" and "Death" columns within the dataset. This targeted approach allows us to streamline the preprocessing steps and concentrate on the essential components relevant to forecasting. The "Year" column is converted into a

datetime format and set as the index, and any missing values in the "Death" column are add ressed, ensuring a complete and coherent time series.

a) Handling Missing Values:

The dataset is inspected for any missing values in both the "Year" and "Death" co lumns.Appropriate strategies, such as forward-filling, are employed to address and fill in m issing values, ensuring a continuous and complete time series.

b) Converting Data into Time Series Format:

The "Year" column, originally presented as a standard numerical format, is conve rted into a datetime type. The "Year" column is designated as the index of the dataset, trans forming it into a time series format. This step is fundamental for chronological data analysi s.

Year	Death
1962-01-01	110456
1963-01-01	106667
1964-01-01	106496
1965-01-01	97709
1966-01-01	102245
1967-01-01	94552
1968-01-01	92669
1969-01-01	87186
1970-01-01	69745
1971-01-01	91272
1972-01-01	102515
1973-01-01	105721
1974-01-01	113447
1975-01-01	131473
1976-01-01	133602

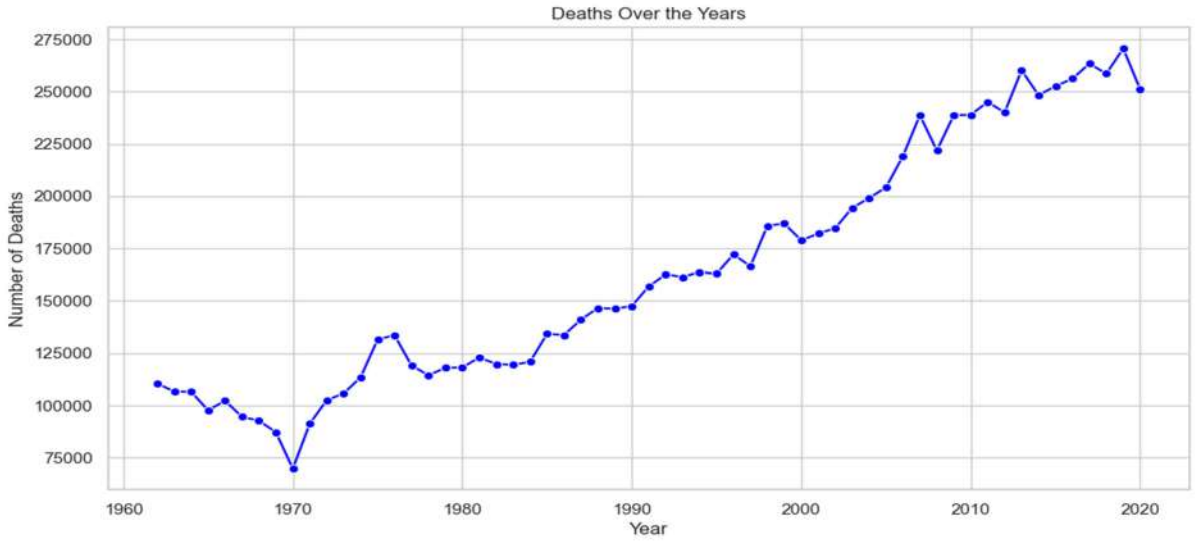
c) Addressing Anomalies or Outliers Detected during EDA:

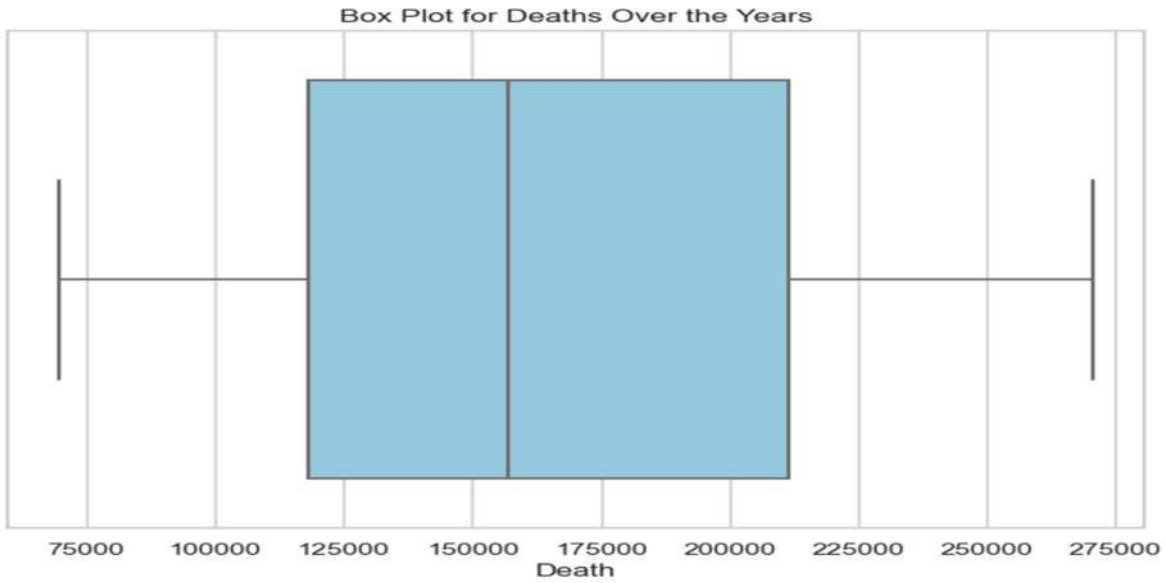
Anomalies and outliers identified during the Exploratory Data Analysis (EDA) p hase are specifically targeted for resolution.Techniques like Z-score analysis or other statist ical methods are applied to detect and address outliers, ensuring that extreme values do not unduly influence the forecasting models.

d) Visualizing Preprocessed Data:

The preprocessed time series data is visualized to confirm that the conversion, ha ndling of missing values, and outlier resolution have been successfully executed. Plots and visual representations help in assessing the effectiveness of the preprocessing steps and un derstanding the data distribution. Before delving into time series forecasting, an EDA is co nducted to understand the distribution, trends, and seasonality in the death data. This phase

involves visualizing the time series, identifying any outliers, and exploring any discernible patterns.





We employ outlier detection techniques, such as the boxplot method, to identify a nd handle any extreme values that might impact the forecasting models. The resulting prep rocessed time series data, visualized forms the foundation for our subsequent time series fo recasting analysis..

By undertaking these preprocessing steps, the dataset is optimized for time series forecasting, ensuring that it aligns with the prerequisites of accurate and reliable predictive modeling.

4. Model Selection:

Several AI models are considered for time series forecasting, with a focus on thos e designed for sequential data. Common choices include Autoregressive Integrated Moving Average (ARIMA), Seasonal-Trend decomposition using LOESS (STL), and more recentl y, machine learning techniques like Long Short-Term Memory (LSTM) networks, which a re well-suited for capturing long-term dependencies in sequential data.

a) Train-Test Split:

The dataset is split into training and validation sets. Typically, earlier years are us ed for training, and later years are reserved for validation.

b) Scaling Data:

Feature scaling is applied to normalize the data. Min-Max scaling or Standard sca ling is used for LSTM model.

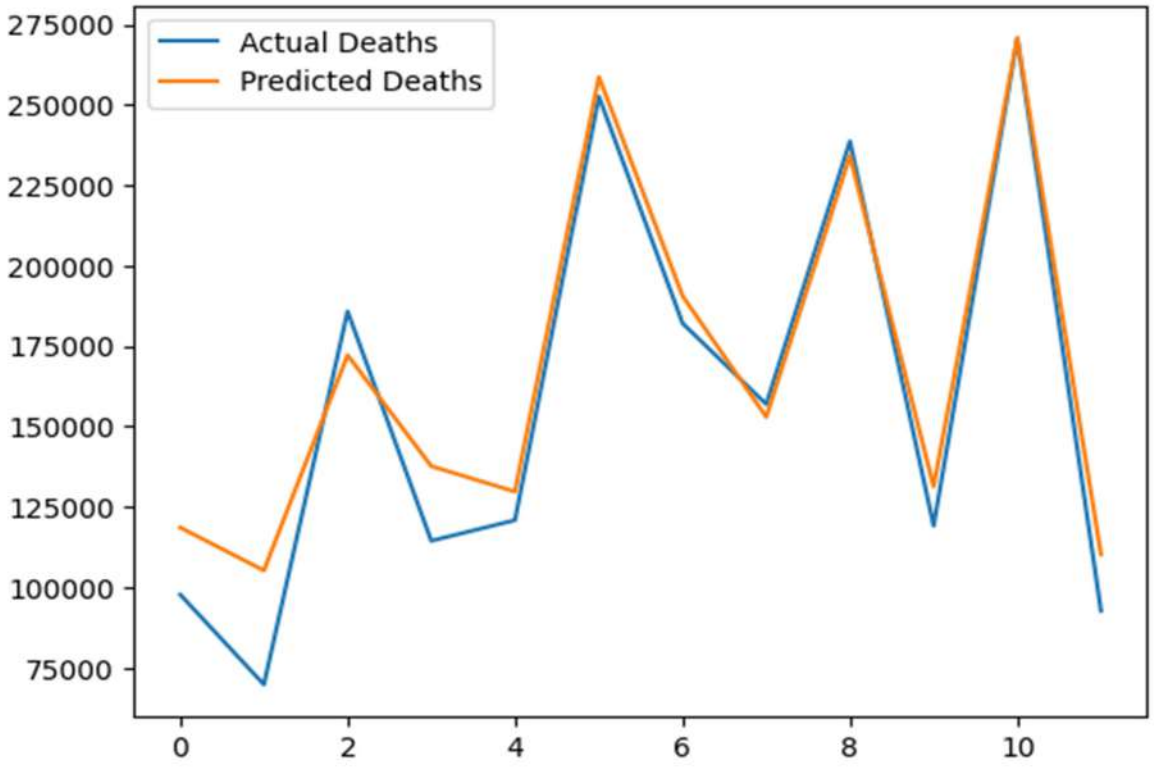
c) Model Training:

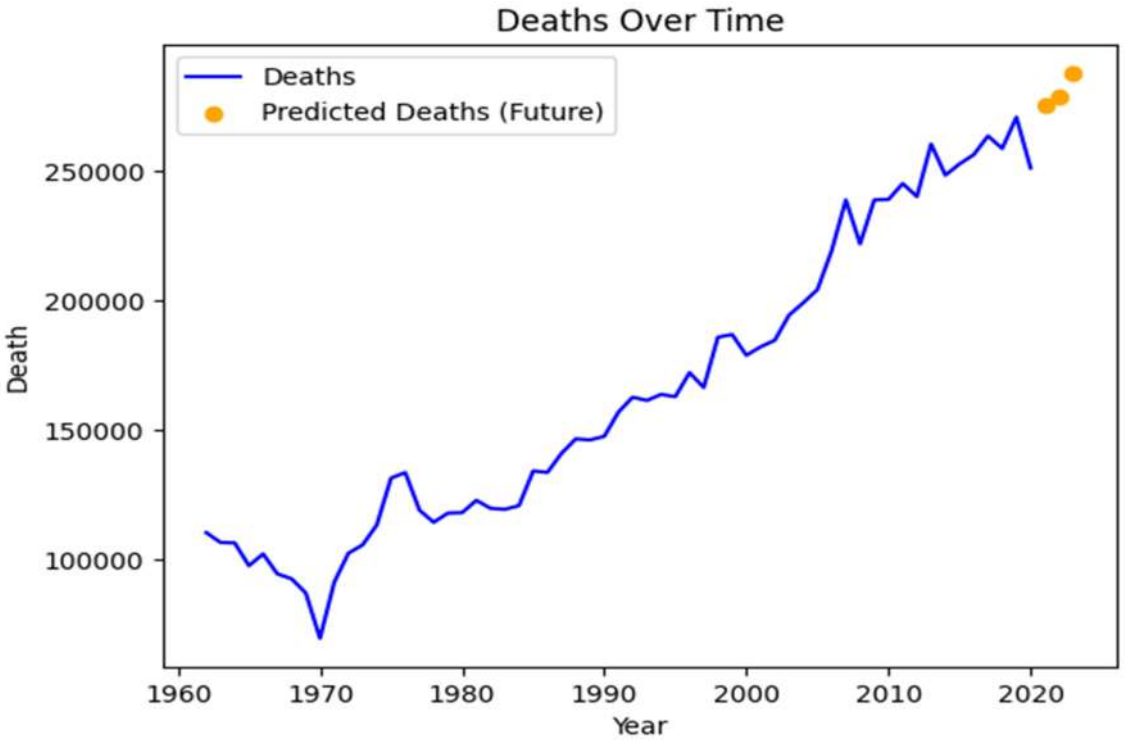
1. Long Short-Term Memory (LSTM) networks

Initially, the Long Short-Term Memory (LSTM) network is selected as the model for time series forecasting. The chosen AI model undergoes training using historical death data.

Validation and Evaluation:

The trained model is validated using the validation set to assess its accuracy.Various evaluation metrics such as Mean Squared Error (MSE) or Root Mean Squared Error (RMSE) are computed to gauge model performance. Mean Squared Error: 269443718.62611395 and RMSE (Root Mean Squared Error):16414.7408





2. Autoregressive Integrated Moving Average (ARIMA)

Next, we explore the ARIMA(3,2,5) by inspecting ACF and PACF plots after differencing the time series data inorder to make it stationary. The ARIMA model stands for Auto Regressive Integrated Moving Average, and the numbers (p, d, q) represent the order of the autoregressive, integrated, and moving average components, respectively.The ARIMA(3,2,5) model represents an AutoRegressive Integrated Moving Average model with three autoregressive (AR) terms, two differencing (I) terms, and five moving average (MA) terms.

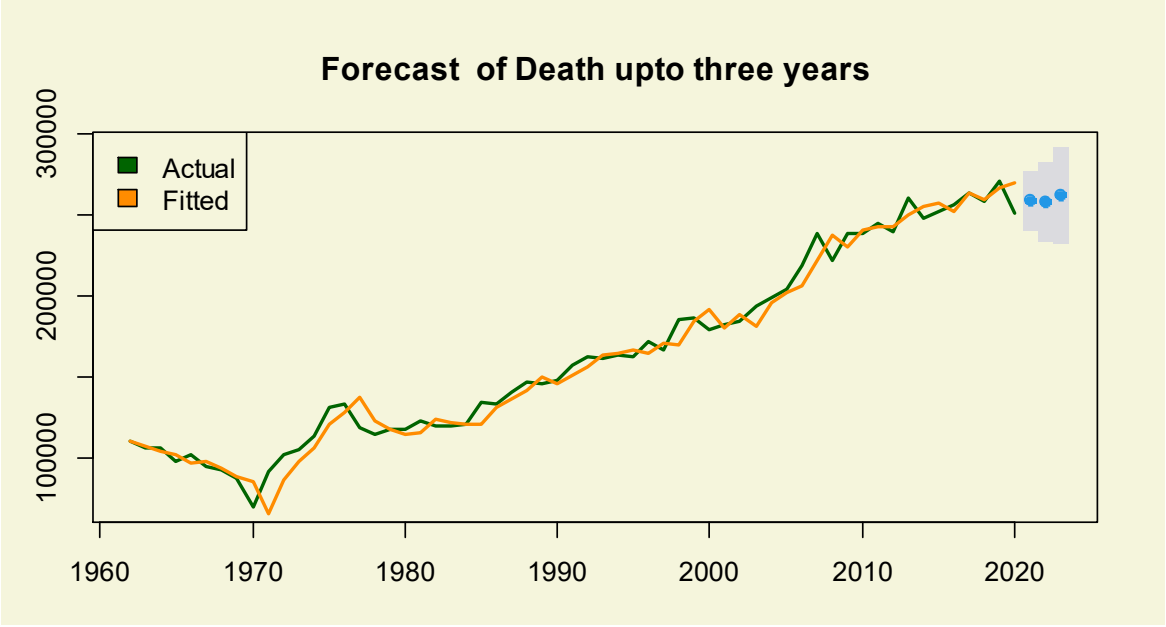
Validation and Evaluation:

The AI model's forecasting performance is analyzed, and predictions for future de ath counts are generated. Visualizations of the forecasted time series data are created to de monstrate the model's ability to capture trends and patterns.

RMSE (Root Mean Squared Error): The RMSE value is 8537.676. This value rep resents the average magnitude of the error. MAE (Mean Absolute Error): The MAE value i s 6378.316.MPE (Mean Percentage Error): The MPE value is 1.049009. A MPE close to ze ro is generally desirable, indicating a low average percentage difference between predicted and actual values. MAPE (Mean Absolute Percentage Error): The MAPE value is 4.45826 5. Like MPE, a lower MAPE is preferred. This metric provides the average percentage diff erence between predicted and actual values. MASE (Mean Absolute Scaled Error): The M ASE value is 0.8674745. A MASE close to 1 suggests that the ARIMA model is performin g well compared to a naive forecast.

Year	Actual	Fitted
1962	110456	110406.6026
1963	106667	106823.664
1964	106496	104391.7208
1965	97709	102517.5753
1966	102245	96516.79461
1967	94552	97442.66006
1968	92669	93397.43246
1969	87186	88089.43266
1970	69745	85712.07919
1971	91272	65851.55688
1972	102515	86558.98614
1973	105721	98135.11035
1974	113447	105856.0146
1975	131473	120527.1884
1976	133602	127656.2841
1977	119113	137034.0829
1978	114434	122557.2126
1979	117961	117604.5951
1980	118140	114312.5699
1981	122904	116046.4438
1982	119827	124393.2793
1983	119414	121717.9708
1984	120841	120398.1615
1985	134230	121339.0565
1986	133654	131407.0327
1987	141047	136527.893
1988	146596	141929.737
1989	146194	149869.0503
1990	147551	145786.5311
1991	156919	151310.5592
1992	162644	155953.6241
1993	161403	163929.2134
1994	163711	164500.3616
1995	162868	166462.3883
1996	172103	164580.8401
1997	166428	171345.9153
1998	185788	170266.774
1999	186828	184320.4002
2000	178795	191355.8657

2001	182059	180505.222
2002	184597	188203.0949
2003	194264	181372.2592
2004	199017	195513.4959
2005	204157	201646.9465
2006	219094	206550.8613
2007	238691	221986.8416
2008	221769	237337.7915
2009	238691	230444.5631
2010	238864	240417.4254
2011	245002	242543.0694
2012	239982	242352.7981
2013	260195	250253.2754
2014	248242	255357.5514
2015	252576	257366.9326
2016	256130	251679.5837
2017	263342	263036.3889
2018	258530	259219.4433
2019	270567	266949.5301
2020	250983	269699.9342



5. Conclusion:

This time series forecasting study for death data using AI provides valuable insig hts into historical trends and future predictions. After comparing the Root Mean Squared E rror (RMSE) of the two models, namely the Long Short-Term Memory (LSTM) and the A RIMA(3,2,5), the ARIMA(3,2,5) model demonstrates superior performance and is selected as the preferred model. The decision is based on the evaluation metric, where a lower RM

SE signifies better accuracy. In this case, the ARIMA(3,2,5) model exhibits a lower RMSE, indicating that it provides more accurate predictions compared to the LSTM model. Therefore, the ARIMA(3,2,5) model is deemed the better choice for the given time series forecasting task.The chosen AI model demonstrates its efficacy in capturing patterns within the data, offering a tool for policymakers and healthcare professionals to anticipate and address evolving mortality trends. Further research and continuous model refinement can enhance the accuracy and applicability of such forecasting methodologies.

Forecast of Death upto three years obtained from ARIMA(3,2,5)

Year	Forecast
2021	258927
2022	257882
2023	262216

5. Libraries and Tools Used

6. Scikit-learn: Used for machine learning model implementation.
7. Pandas: Utilized for data manipulation and analysis.
8. Matplotlib and Seaborn: Employed for data visualization.
9. TensorFlow: Applied for deep learning tasks.
10. Jupyter Notebooks: Used as the primary environment for code development and experimentation

ARTIFICIAL INTELLIGENCE & DATA ANALYTICS

DATA FORECASTING USING
MACHINE LEARNING MODELS
ARIMA, ANN, LSTM, SVM, RANDOM FOREST

DATA ANALYSIS USING PYTHON AND R

Government of Kerala

DIRECTORATE OF ECONOMICS & STATISTICS
Vikas Bhavan, Thiruvananthapuram- 695 033
Phone: 0471-2305318, Fax: 0471-2305317
email: ecostatdir@gmail.com, dgdir.des@kerala.gov.in
website: www.ecostat.kerala.gov.in