Government of Kerala

COMPENDIUM OF PROJECT REPORTS

#AIFORALL

Capacity Building in Artificial Intelligence & Data Analytics

Conducted by

DIGITAL UNIVERSITY KERALA

From 28th Nov 2022 to 31st Jan 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 28th Nov 2022 to 31st Jan 2023

In collaboration with

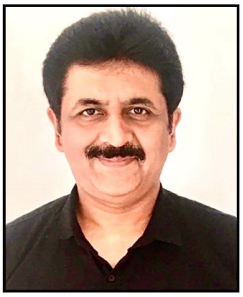
Department of Electronics & Information Technology Kerala
&
Kerala State Information Technology Mission

May 2023

DIRECTORATE OF ECONOMICS & STATISTICS
VIKAS BHAVAN, THIRUVANANTHAPURAM

Government of Kerala

PUNEET KUMAR IAS
ADDITIONAL CHIEF SECRETARY



Planning & Economic Affaris, PIE & M Dept.
Government Secretariat
Thiruvananthapuram-695 001
Phone : O471-2335452, 0471-2518356
Mobile : +91-9910816200
E-mail : prlsecy.plg@kerala.gov.in

Date : 19/05/2023

MESSAGE

It is learnt that a capacity building programme on Artificial Intelligence and Data Analytics has been conducted by Kerala University of Digital Sciences Innovation and Technology for statistical staff of the Department of Economics and Statistics under Planning & Economic Affairs Department. This training course was organized with the budgetary support of Department of Electronics and Information Technology Kerala and Kerala State Information Technology Mission. This 9 weeks course was exclusively conducted for statistical personnel in the cadre of Joint Director to Statistical Assistant working in Directorate and other departments. This course includes advanced learning topics- Machine Learning techniques like Regression Analysis. Neural Networks, Artificial Intelligence, Statistical Prediction etc. This Compendium contains the reports of projects done by each trainee as part of the capacity building programme by analyzing the datasets of various subjects which are available in the department and other socio economic sectors of the State.

I think this advanced learning programme on Data Science will be much useful to a department like Economics and Statistics, the Nodal Agency of official statistics in the State. This will lead to encourage the use of latest data science tools with the help of technology for analyzing and disseminating better results to the planners. In this occasion I thank Vice Chancellor Digital University Kerala, Director of Economics and Statistics and other officials in DUK and DES, who were behind the conduct of this programme very effectively.

I hope this Compendium of Project Reports will be a worthy reference for the statistical staff in DES, planners, academic institutions, research agencies, research aspirants and those working in the field of data science.

Puneet Kumar
19/05/2023

(PUNEET KUMAR IAS)

Government of Kerala

Dr. RATHAN U. KELKAR IAS
SECRETARY TO GOVERNMENT

Electronics & Information Technology
Taxes and Excise Department
Government Secretariat
Thiruvananthapuram, Govt. of Kerala

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

Date 18-05-2023

MESSAGE

Capacity building initiatives are of prime importance to develop in-house IT teams that are fully equipped to deal with the complexities of managing and implementing IT projects and to reduce dependency on external agencies. As an organisation like Department of Economics & Statistics which is dealing with official statistics, capacity building in Data Science is essential to equip the statistical personnel to the level of data scientists and to utilise them to build quality research outputs to cater the needs of Government planning purpose.

State IT department conducts Capacity Building programme in Artificial Intelligence and Data Analytics through Digital University Kerala with the objective of equipping a team of officers in Government system to understand the latest data science tools and utilise the technology in effective decision making. I am happy to learn that the Department of Economics & Statistics has participated the second batch of this advanced programme and a team of 19 officials from DES has successfully completed this course during the year 2022-23.

I hope that this learning initiative will lead to build a team of statistical personnel to conduct research in official statistics and thus to empower the Statistical System in Kerala. I congratulate the Vice Chancellor DUK and his team for arranging the second batch of Al & DA exclusively for DES staff and appreciate the effort taken by the Director and his team in DES for participating in it.

Dr. RATHAN U. KELKAR IAS

Prof (Dr.) Saji Gopinath
Vice Chancellor

Kerala University of Digital Sciences
Innovation & Technology
Technocity Campus, Mangalapuram
Thiruvananthapuram- 695317
Date: 25-05-2023

MESSAGE

In order to realize the Kerala's commitment to become a truly digital State where advanced technologies are used by Government for improving citizen services, Digital University Kerala (DUK) has started programs on imparting high end technology skills to Government employees in a structured manner. I am happy that twenty officers from the Department of Economics and Statistics (DES) with proven expertise in Statistics, have been trained in Artificial Intelligence and Data Analytics at DUK, as part of this initiative being implemented with the funding support from the Department of Electronics and IT, Government of Kerala. DUK's training and technology development services deliver skill development among Government and professionals to prepare them to lead the digital transformation of the future. DUK capacity building training nurtures them to build their own services to address specific challenges faced by the Department and build systems using cutting edge technologies like AI and data analytics. Fostering a knowledge economy, DUK continues to expand its offering to new courses and technologies which can match government and industries with a world class digital education experience.

As DES is the state nodal agency for collecting and disseminating data on various socio-economic sectors of Kerala, it is indeed a great pleasure that DES has taken an innovative step to leverage the technologies for building such services to transform the activities of DES to enhance its role in official statistics. Hope the participants of course could translate their newly acquired skills to improve your services and making better decision making with increased transparency and intelligence.

Wish you all the Best.

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: 25-05-2023

ACKNOWLEDGEMENT

Recognizing AI's potential to transform economics of Kerala, Digital University Kerala (DUK)'s pursuance in upskilling Government employees for current and future technologies position themselves as leaders with unique brand of #AIFORALL. #AIFORDES aims to empower human capabilities to enhance DES skilled expertise for effective implementation of AI to evolve scalable solutions for emerging economics. The capacity building imparted to DES was a great success with the effort of DES officials and DUK. A 45 days intensive customized training with hands-own experience using DES data for their services and applications enriched the programme with greater success. Remarkable contributions have been made by the participants by implementing AI projects in critical areas such as health, Agriculture, Consumer Price indexing etc using Machine Learning and Deep learning techniques. Forecasting of consumer price indexing in Kerala, Predictions on CRS Data – Live birth, Death, Maternal Death and Infant Death, Prediction of Cost of Cultivation of Important Crops, paddy cultivations in Kerala etc. are a few

Congratulations to all the participants who have successfully completed the course and projects and we are grateful to the Director Mr. Sajeevu P P and Deputy Director Mr. D S Shibukumar for constant interaction and cooperation for 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, Futures Studies of University of Kerala.

I am grateful to Dr. Saji Gopinath, Vice Chancellor of Digital University Kerala for his 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.

Elizabeth Sherly

Government of Kerala

SAJEEVU P.P.
DIRECTOR

Department of Economics & Statistics
Vikas Bhavan , Thiruvananthapuram- 695033
Phone: 0471- 2305318, Fax: 0471- 2305317
Email: ecostatdir@gmail.com

Date: 25-05-2023

PREFACE

Strengthening the statistical system in Kerala is the vision of the Department of Economics and Statistics and in line with this capacity building programmes and training programmes are being conducted every year through State Academy on Statistical Administration and Institute of Management in Government Kerala.

This project reports are prepared by statistical personnel as part of the capacity building programme on Artificial Intelligence and Data Analytics which was conducted by Digital University Kerala for 45 days from 28 November 2022 to 31 January 2023. This course was offered by DUK and sought nominations from DES. DES has sent nominations of 20 statistical staff and requested DUK to consider this programme as an exclusive batch of DES staff. DUK has agreed the request of DES and accordingly this adva learning programme was conducted at IIITM-K, Technopark for the convenience of DES.

This course was much useful for the staff in DES as the department is dealing in official statistics and several surveys and studies are being conducted in various socio-economic subjects under the guidance and technical support of Govt of India. In this occasion, I would like to express my sincere gratitude to DUK especially Dr. Saji Gopinath, Vice Chancellor DUK, Dr. Elizabeth Sherly, Programme Coordinator DUK and Sri. Surag M, Assistant Coordinator DUK for arranging the programme as an exclusive batch of DES staff and for their sincere and timely intervention on all matters for the successful completion of the course in an effective manner.

I also express my sincere thanks to Additional Director Smt. Lathakumari C.S. and Deputy Director Sri. D.S. Shibukumar for their painstaking efforts for the successful accomplishment of this two months programme in a very fruitful manner.

I would like to appreciate the officers who have participated and taken effort for completion of this capacity building initiative. I also express my thanks to the controlling officers who have given permission to attend this 45 days course without affecting the normal duties.

I hope that this compendium of reports will be of great useful for the department, research aspirants, academicians, colleges and universities and all those who are interested in data analytics and predictive learning in official statistics. Suggestions for improvement of the programme and contents of this report are most welcome.

Sajeevu P.P.

Trainees of the Programme

1 Smt. Sudarsha R., Joint Director, Survey & Design Division

2 Smt. Shailamma K., Deputy Director, Evaluation Division

3 Sri. Preeth V.S., Deputy Director, MCCD Division

4 Smt. Maya R., Deputy Director, O/o Chief Town Planner

5 Smt. Dhanya A., Deputy Director, O/o IDRB

6 Sri. Vijay R., Research Officer, Survey & Design Division

7 Sri. Sijith K.S., Research Officer, Labour & Housing Division

8 Sri. Abhilash K.V., Research Officer, ASI Division

9 Smt. Praseeda Gopan, Research Assistant, State Income

10 Smt. Suma S.A., Research Assistant, Evaluation Division

11 Sri. Prasanth B.R., Computer Supervisor, Computer Division

12 Sri. Rajesh R., Statistical Assistant Grade I, Computer Division

13 Sri. Ashad W.A., Statistical Assistant Grade I, Computer Division

14 Smt. Neethymol Kurian, Statistical Assistant Grade II, BSLLD Division

15 Smt. Chitra J.V., Statistical Assistant Grade II, District Office, Thiruvananthapuram

16 Smt. Brijila A.S., Statistical Assistant Grade II, District Office, Thiruvananthapuram

17 Kumari. Minu Merin Andrews, Statistical Assistant Grade II, PPC Division

18 Kumari. Baby Sindhu, Statistical Assistant Grade II, ASI Division

19 Sri.Nidhin Babu M., Statistical Assistant Grade II, District Office, Thiruvananthapuram

BEHIND THE PROGRAMME

Coordinators

Department of Economics & Statistics

1. Smt. Lathakumari C.S., Additional Director (General)
2. Sri. D.S. Shibukumar., Deputy Director, Computer Division

Digital University Kerala

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

Faculty Members

Digital University Kerala

3. Dr. Elizabeth Sherly, Distinguished Faculty DUK
4. Dr. Malu G., Research officer, DUK
5. Sri. Surag M., Scientific Associate (Training & Development) DUK
6. Smt. Nayana Uday, Research Scholar, DUK
7. Smt. Leena G. Pillai, Scientist, DUK
8. Smt. Judy K. George, Research Scholar, DUK
9. Smt. Sabitha Rani B. S., Research Scholar, DUK

Other Organizations

1. Sri. Prasad K. Nair, Project Manager, GBS Technologies Trivandrum
2. Dr. Satheeshkumar Krishnan Nair, Professor, Department of Futures Studies, University of Kerala
3. Smt. Gayatri Menon, Research Scholar, Kerala University
4. Sri. Somasekharan Pillai, Former HoD, Department of Statistics, University College, Thiruvananthapuram
5. Smt. Raji Gopinath, Information Officer, Entrance Commissionerate, Thiruvananthapuram

INTRODUCTION

The Department of Economics and Statistics is the Nodal Agency in Kerala for collection, compilation and analysis and dissemination of data on all socio-economic sectors of the State economy. Kerala has a well-established statistical system in India to organize and conduct sample surveys and census 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 details of schemes of surveys conducted by the department are mentioned in the official website https://www.ecostat.kerala.gov.in.

The department is publishing on an average 30 reports of various categories in each year and the digital copy in pdf format are available in official website. So far, the department has published more than 500 reports online. The digital publication started from 2004. In addition about 900 reports, which were published earlier, are available in the department library in printed format.

Conventional methods of statistical analysis are usually performed in the reports of the department. In this modern world, data is the oil for planning and Data Science provides latest techniques by blending the ICT technology for analysis the data more efficiently and to provide a best result for effective decision making in a quick and easier manner. 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.

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 2022-23. This programme of AI & DA is the second batch of the DUK with the funding of State IT department and Kerala State IT Mission. The programme was conducted in 45 working days, started from 28 November 2022 and ended on 31 January 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 45 days course was successfully completed on 31 January 2023 with project presentation by each participant. After the completion of the programme an orientation programme on this subject has been conducted by DES for the directorate staff in February 2023. Project presentations were arranged before the directorate staff during the months of March and April 2023 for understanding the data analysis done by the trainees and to give a motivation to the rest of the staff.

Evolution of the programme

DUK has called for nominations form DES for joining the programme of second batch AI & DA conducted by DUK with the budgetary support of Govt of Kerala. DES has sought willingness from the directorate staff. Willingness was sought from some staff working in line departments and district office of Thiruvananthapuram. 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. More than 20 nominations were obtained, after screening 20 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.

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, RStudio and data analysis in R.

Project Assignment and Evaluation

Two weeks project assignments were 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

Crop statistics published by DES under EARAS scheme, daily retail prices of essential commodities collected and published by DES, data on Indices of Industrial Production, industrial data under the scheme Annual Survey of Industries, cost of cultivation data under Cost of Cultivation Survey conducted by DES, Consumer Price Indices published by DES, Building construction cost data collected by DES quarterly from selected centers in the State, land utilization data under EARAS scheme, Birth and death statistics collected under Civil registration System, cause of death data under Medical Certification of Cause of Death (MCCD) scheme are the data 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), Artificial Neural Networks (ANN), Support Vector Machines (SVM), Random Forest etc. are the methods used by the trainees for analyzing and forecasting data.

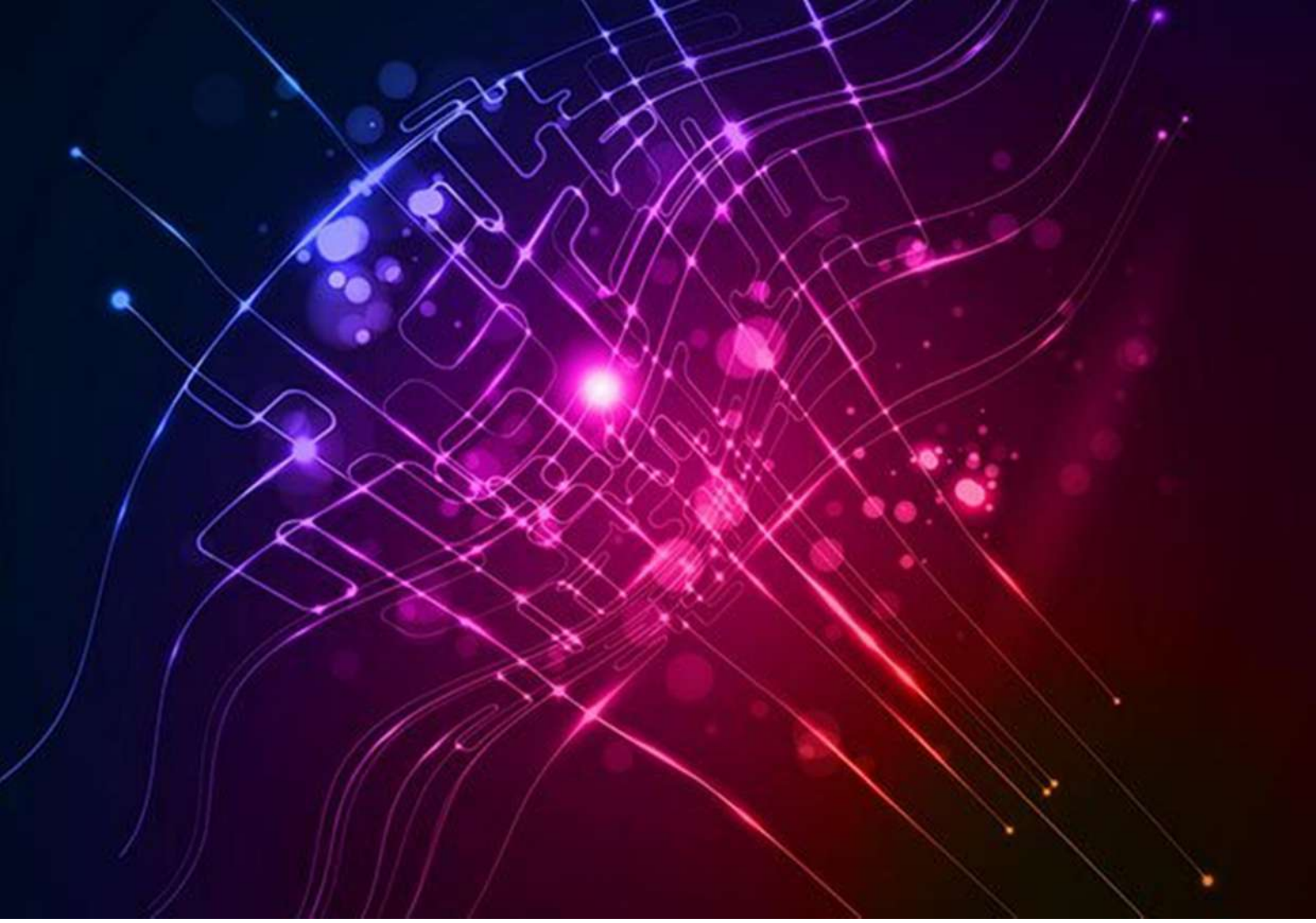
Project Reports

This compendium contains project reports submitted by the participants. In addition, some of the trainees have built user interface in Python for inputting data for applying in the model found by them and retrieving results dynamically.

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. Definitely this kind of initiatives will lead to set up a Data Analytics Unit in the department in the nearing future. Data Analytics Wing is essential to think like a data scientist and analyses 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 in the modern world of Data Science. DES is one of the prime departments as data is essential for planning; a transition from conventional style of data analysis is expected from DES by the planners community. Such a transition is inevitable.

INDEX OF PROJECT REPORTS



Machine Learning Models for predicting Rice and Tapioca yield in Kerala-

A comparative analysis

Submitted By

Smt. Chithra J.V., Statistical Assistant Grade II and
Sri. Ashad W.A., Statistical Assistant Grade I

1. Introduction

Analyzing data on the production, consumption, and trends of Kerala paddy and tapioca can provide insights into the agricultural sector's overall health and potential economic impact. Here are some potential areas of relevance for analyzing data on these crops:

- Agricultural policies and planning: Data on production and consumption can inform policymakers about the current status of paddy and tapioca farming in Kerala. This information can help them develop effective policies and plans for improving the agricultural sector, increasing crop yields, and supporting farmers.
- Market trends and opportunities: Analyzing data on prices, demand, and supply can help farmers and agricultural businesses identify market trends and opportunities for expanding their operations. They can use this information to make informed decisions about crop selection, marketing strategies, and pricing.
- Resource management: Data on crop yields, land use, and farming practices can help farmers make informed decisions about resource management. For example, they can use this information to determine the most efficient and sustainable methods for irrigation, fertilization, and pest control.
- Climate change adaptation: As climate change continues to affect the agricultural sector, analyzing data on crop production can help farmers and policymakers understand the impact of changing weather patterns on crop yields. They can use this information to develop strategies for adapting to these changes, such as implementing new farming practices or crop varieties that are more resilient to extreme weather conditions.

Overall, analyzing data on Kerala paddy and tapioca production and consumption can provide valuable insights into the agricultural sector's health and potential economic impact. This information can help policymakers, farmers, and businesses make informed decisions about resource management, market opportunities, and adaptation to climate change.

2. Objective

India's agricultural yield is heavily dependent on weather conditions, especially for rice cultivation that heavily relies on rainfall. This makes timely predictions and analyses of future crop productivity critical to help farmers maximize their yield. Unfortunately, predicting crop yield has been a significant agricultural challenge, as farmers typically rely on past experiences with yield to predict future harvests. To address this issue, various techniques and algorithms have been developed for crop yield prediction. The role of Big data analytics techniques in agriculture has also contributed to the effectiveness of these techniques.

Moreover, forecasting the yield of paddy and Tapioca is essential for planning purposes, and rice import policies should be based on such predictions. This study aims to investigate the past, present, and future trends of paddy and Tapioca production in Kerala. There are multiple ways to increase and improve the crop yield and the quality of the crops. Data mining is also useful for predicting crop yield production.

The main objectives are

- ❖ To Analyze the trends in area, production and productivity of Paddy and Tapioca in Kerala.
- ❖ To use machine learning techniques to predict crop yield.
- ❖ To analyze climatic parameter (Rainfall)

3. Methodology and Methods used

The proposed model makes use of two well-known machine learning algorithms applied to the customized input datasets for rice and Tapioca Separately. They are

3.1 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.

Important Points:

- There must be **linear relationship** between independent and dependent variables
- Multiple regression suffers from **multicollinearity, autocorrelation, heteroscedasticity**.
- Linear Regression is very sensitive to **Outliers**. It can terribly affect the regression line and eventually the forecasted values.
- Multicollinearity can increase the variance of the coefficient estimates and make the estimates very sensitive to minor changes in the model. The result is that the coefficient estimates are unstable.

3.2 Linear Support Vector Regression

It is the generalization of support vector machine (SVM) for regression problems. SVR can be linear or non-linear based on the kernel function. In case of regression problem, in place of finding hyperplane, SVR find epsilon in sensitive region around the function that have at most epsilon-deviation, known as epsilon tube.

The models were trained on the training set and evaluated using the cross validation technique to predict the crop yield.

4. Dataset Used

The dataset used for this project consists of Season-wise data on Rice and Tapioca, including information on Area, Rainfall, and Production of 14 districts in Kerala from the year 2005 to 2020. The dataset contains 1260 rows and 5 columns, including data on Area in hectares, Production in tons, and Rainfall in millimeters. Both numerical and categorical attributes are included in the dataset. The data were collected from the Agricultural Statistics of the Department of Economics and Statistics, Government of Kerala, and are considered as the main dataset for this study.

4.1 Data Partitioning

The Entire dataset is partitioned into 2 parts: for example, say, 75% of the dataset is used for training the model and 25% of the data is set aside to test the model.

5. Tools and Libraries Used

For the current project, Python is chosen as the programming language for all the implementations, starting from extracting the data, to evaluating the model. It has a huge library support for applications in the field of Machine Learning and Artificial Intelligence and this makes Python more suitable for solving problems in real world scenarios. Google Collab notebook were used to write the code for the project. All the exploratory data analysis was done using Python libraries like NumPy, Pandas, Matplotlib, and Seaborn. The selection, training and evaluating the model was done using the Scikit-Learn library and its classes. No specific operating system is required as Python is a portable language.

5.1 Performance Parameters

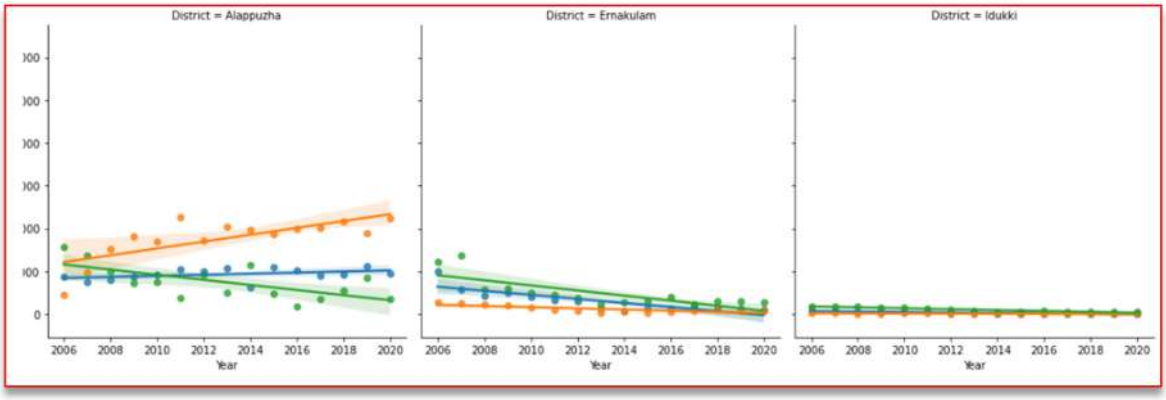
In this study, the performance parameters used are RMSE and R2 score. As a supervised learning algorithm is being used, the training instances have predefined labels. The RMSE is a typical performance measure for regression tasks, providing an estimate of the system's prediction error, with larger errors being given a higher weight. A higher RMSE value indicates lower efficiency of the model. Another evaluation metric used in this study is the R2 score, which is a popular measure of the model's accuracy. It represents how closely the data values align with the regression line. A higher value of R2 or (R) indicates a better fit of the model to the data.

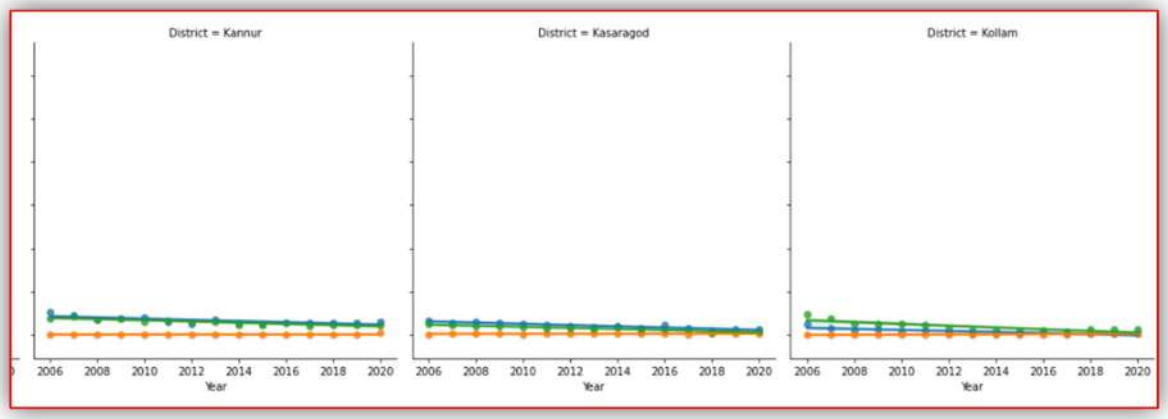
6. Results and Analysis

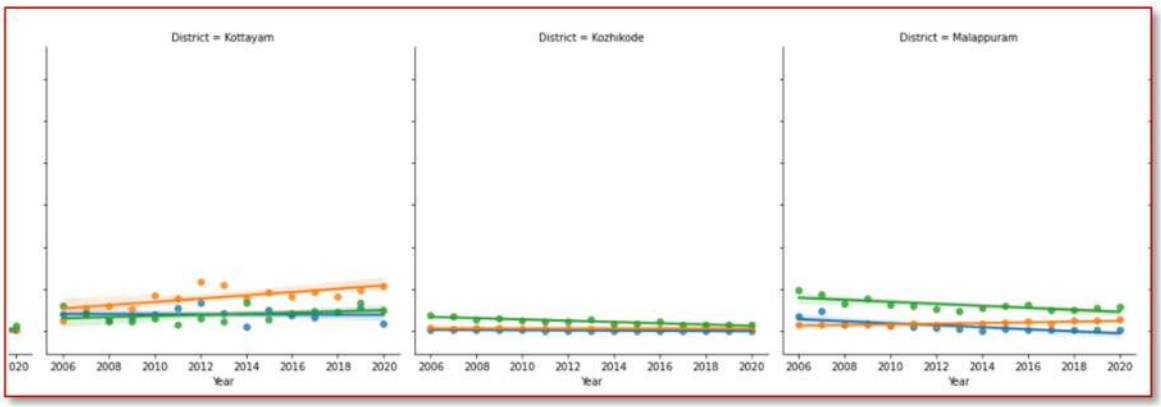
The following section outlines the findings of our study, which involved implementing the model discussed. Prior to implementation, we conducted a thorough analysis of the dataset to gain insights. Therefore, this section firstly provides the statistical results and followed by the performance analysis of the proposed model.

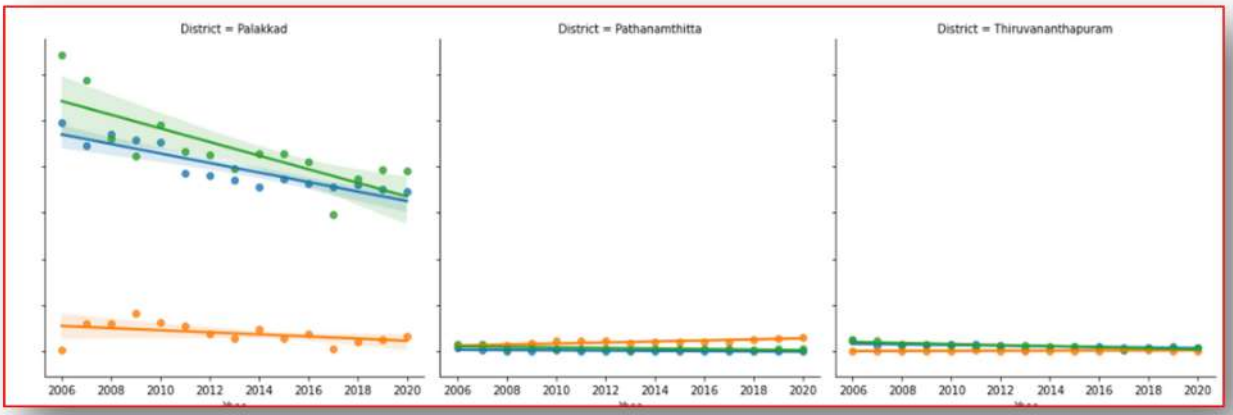
6.1 Statistical Analysis

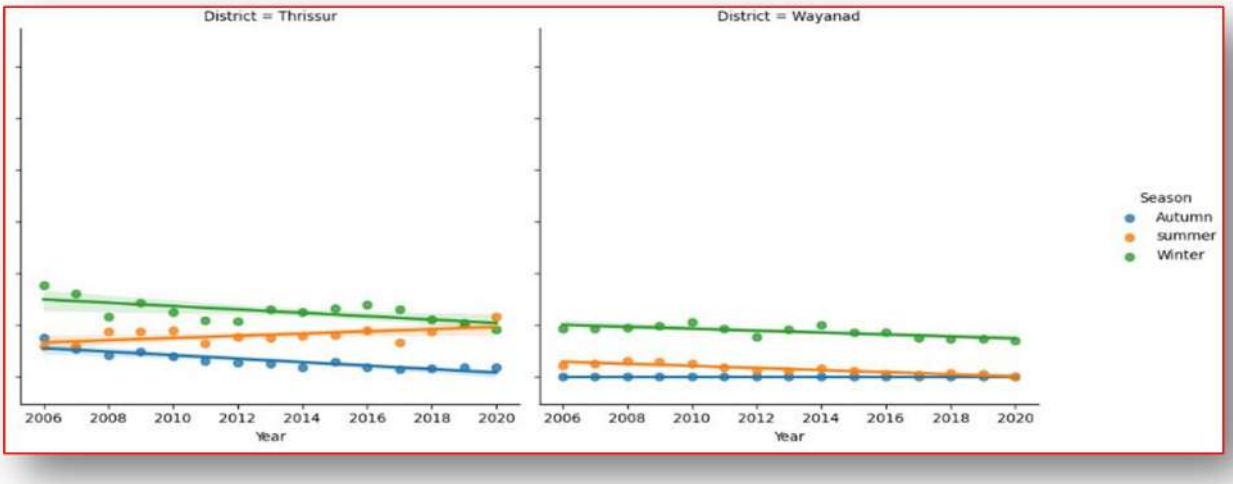
Analyzing the Area of Paddy Cultivation by District and Season between 2015-2020





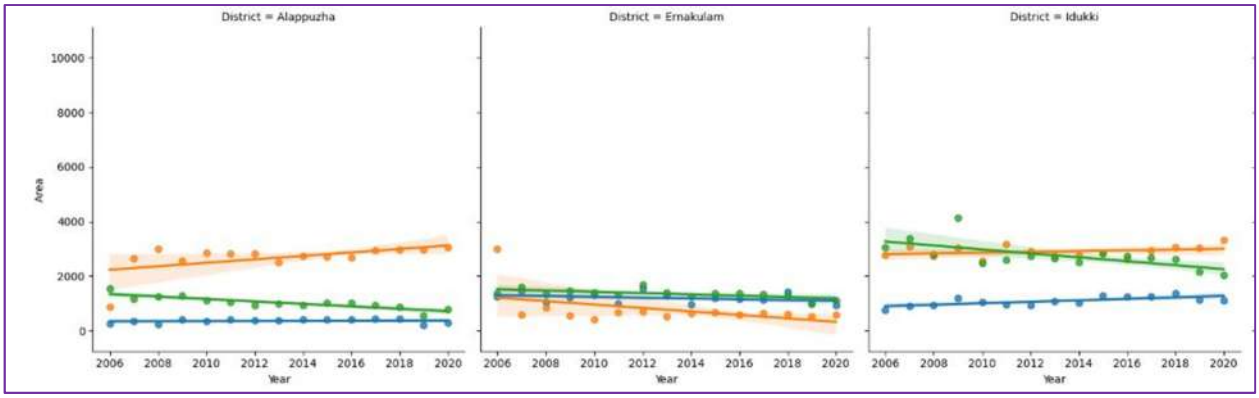


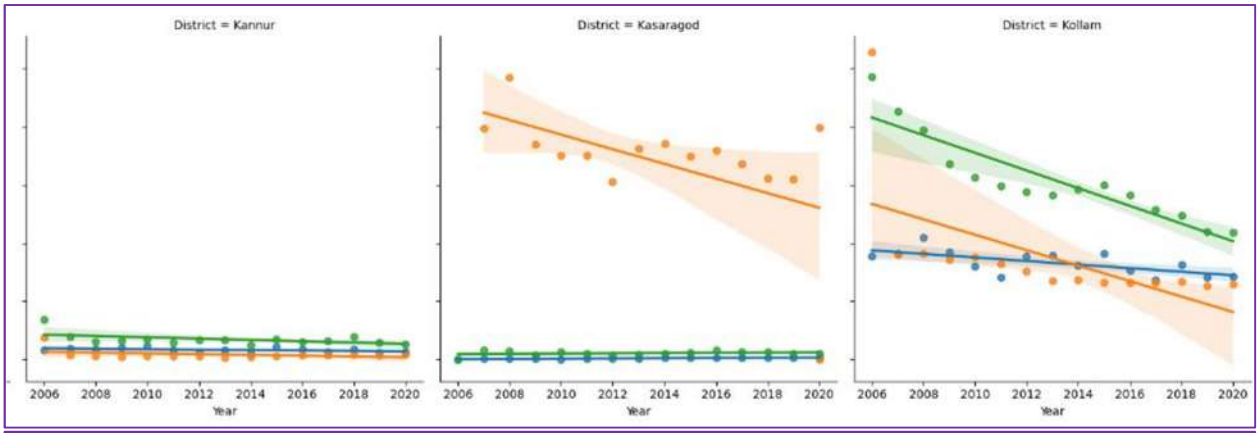


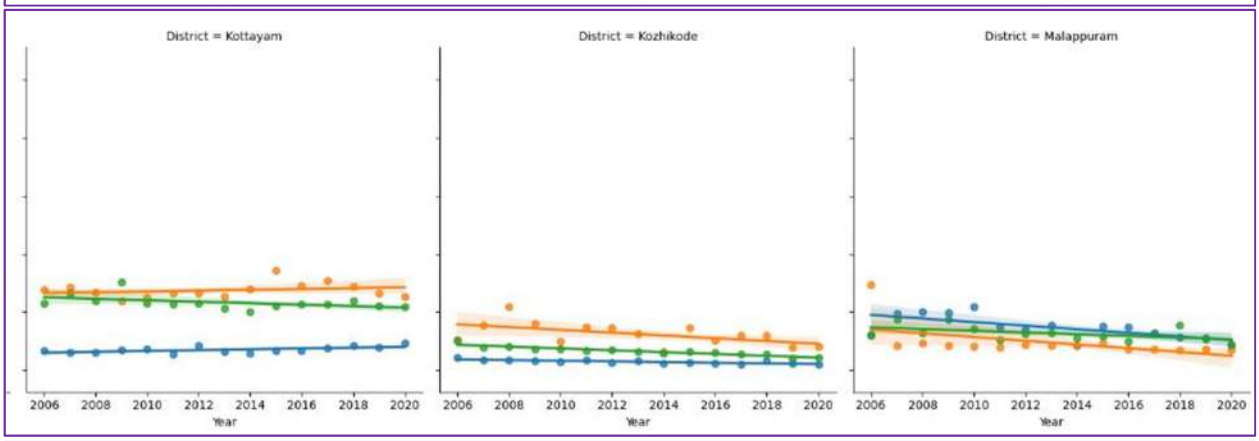


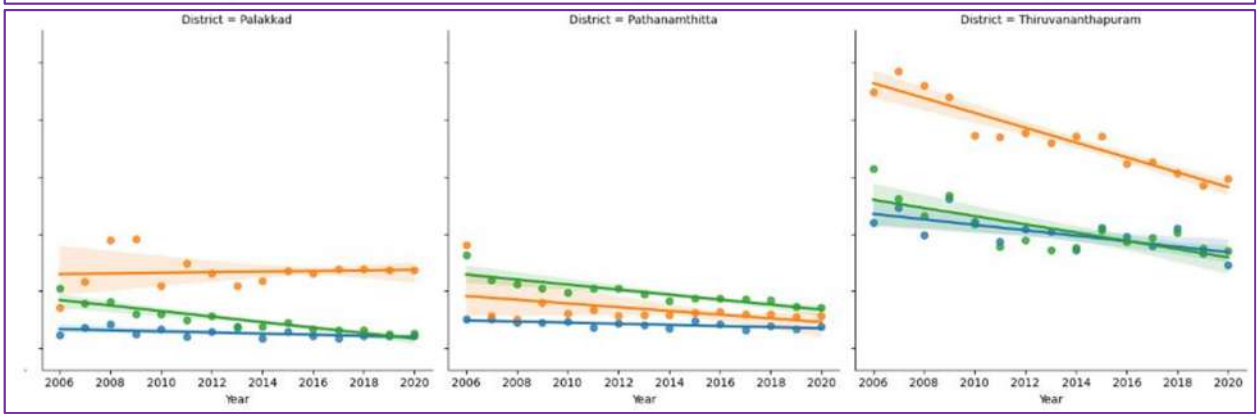
- Paddy cultivation in most districts typically occurs during the winter season, but in Alappuzha and kottayam, it takes place during the summer season, and there has been a progressive increase in cultivation.
- In kerala Palakkad has the largest paddy area over 15 years but the same time there has been a gradual reduction in overall paddy cultivation area during Autumn and winter.

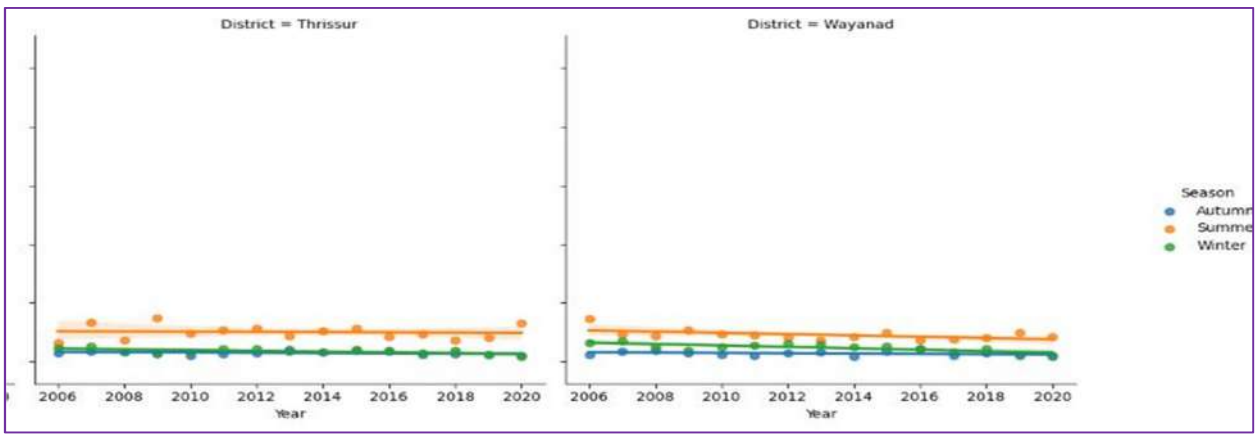
Analyzing the Area of Tapioca Cultivation by District and Season between 2015-2020











Tapioca cultivation occurs mostly during the summer season. Thiruvananthapuram and Kollam are the leading districts for tapioca cultivation. In Alappuzha, Idukki, and

Palakkad, there has been a slight increase in the area of tapioca cultivation during the summer season.

6.2 Statistical Description of the Dataset

Paddy

index	Area	Production	Rainfall
std	22456.5	59881.1	749.8
min	603.0	1467.0	43.7
mean	15234.8	39940.3	2873.7
max	113919.0	270103.0	5895.0
count	210.0	210.0	210.0
75%	17275.25	49053.0	3351.60
50%	6184.5	13472.5	2903.75
25%	2873.5	6392.5	2379.125

Tapioca

index	Area	Production	Rainfall
count	224.0	224.0	224.0
mean	5266.6	185509.56	2903.9
std	4979.66	160725.89	753.72
min	245.0	5672.0	43.7
25%	1815.75	61252.5	2385.42
50%	4088.5	134460.0	2940.45
75%	6128.75	241929.25	3413.82
max	23814.0	694984.0	5895.0

6.3 Correlation matrix for the training Set

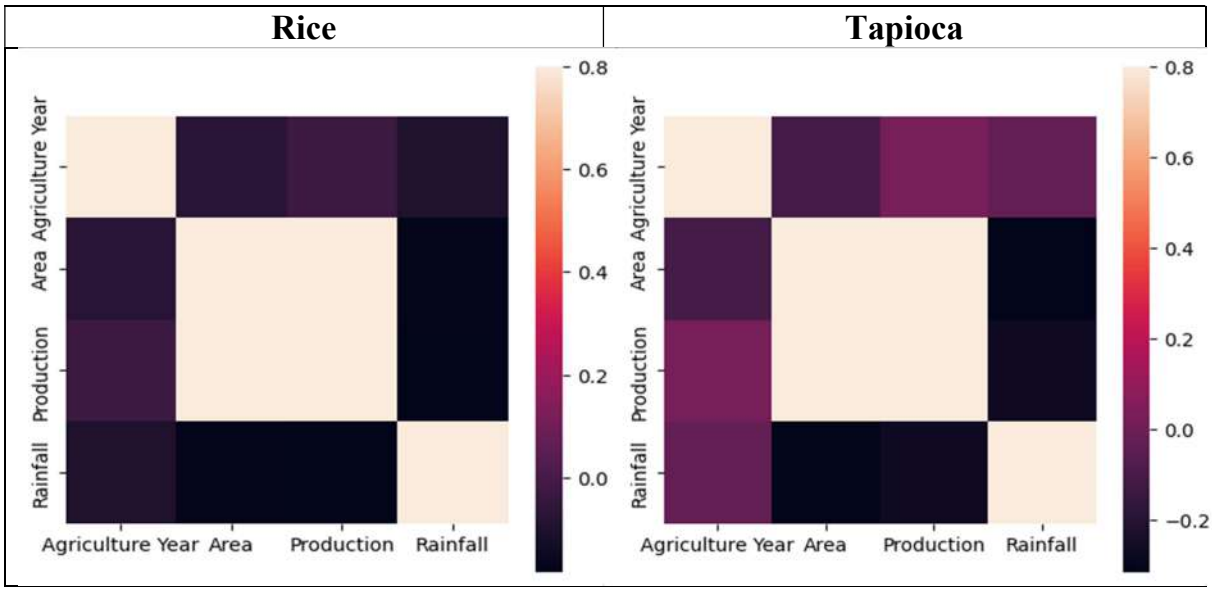
Rice

index	Area	Production	Rainfall
Rainfall	-0.17	-0.18	1.0
Production	0.99	1.0	-0.18
Area	1.0	0.99	-0.17

Tapioca

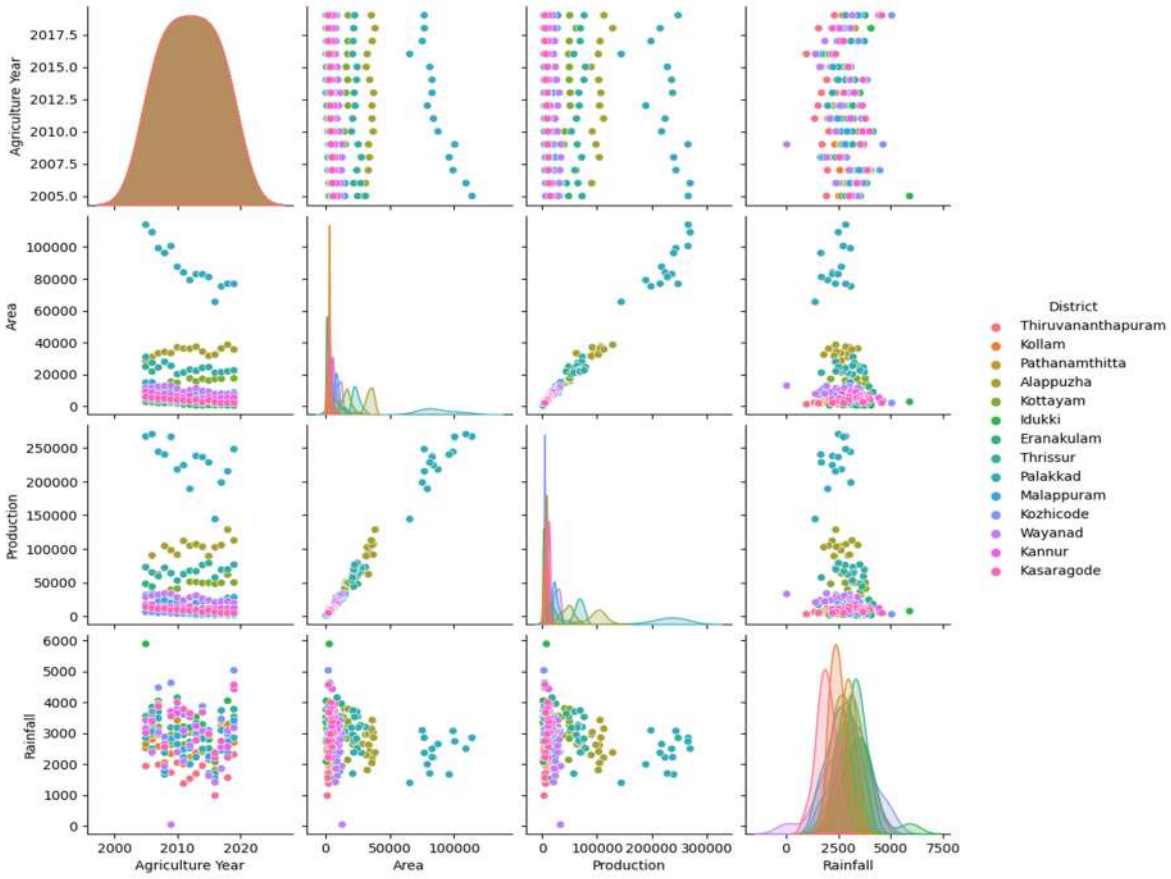
index	Area	Production	Rainfall
Area	1.0	0.96	-0.31
Production	0.96	1.0	-0.27
Rainfall	-0.31	-0.27	1.0

6.4 Heat map showing the correlation between numerical attributes training set

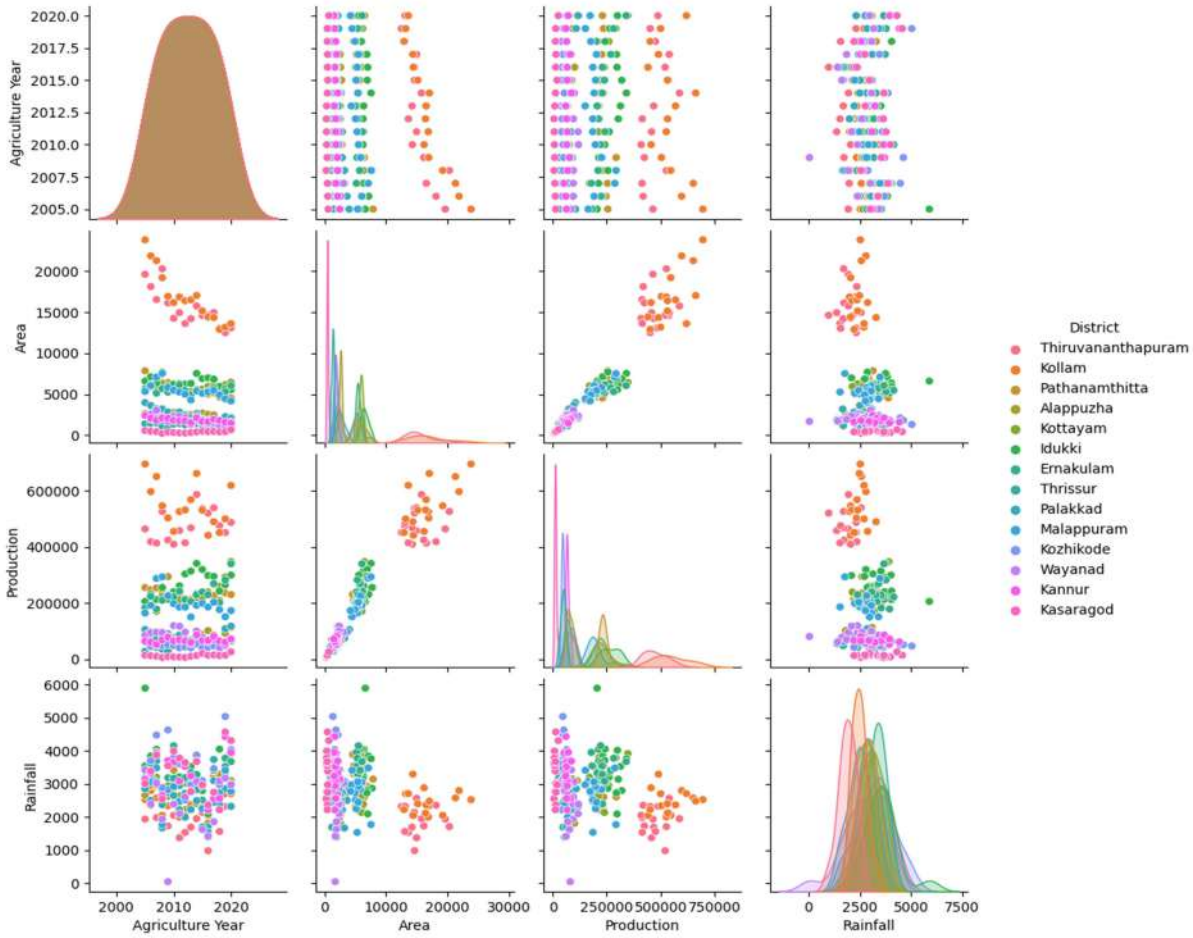


6.5 Scatter Matrix of the Training Set

Paddy



Tapioca



From all the above tables and diagrams correlation between the area, production of paddy, tapioca and rainfall indicates a weak negative correlation between the two variables. The relationship is not very strong. Also its important to note that correlation does not imply causation, and there may be other factors that affect the area of paddy, tapioca, such as soil type, temperature, or human factors like land use changes or agricultural practices.

6.6 Evaluation of Scores

In this paper, the algorithms such as Linear Regression and Support Vector Regressor are used to predict rice and Tapioca yield. The mean square error has been used to measure the loss incurred by the model.

Rice

Model	RMSE	R2
Linear Regression	10220	0.98
Support Vector Regression	52742	0.28

Tapioca

Model	RMSE	R2
Linear Regression	40570	0.93
Support Vector Regression	38747	0.04

This result clearly shows that Linear Regression fared much better than Support vector regression.

Conclusion

The Linear Regression is the basic regression algorithm which can be used only when the dataset is simple. It has been observed that that performance of linear regression has been affected by non-linearity of data, presence of outlier and high correlation among the features. Therefore, it may not work well for complex datasets. For this algorithms like Ridge Regression and Lasso Regression, which are more regularized forms of Linear Regression can be used when the dataset contains a large number of features or unwanted features. Predicting agricultural production involves analyzing various factors such as, weather patterns, soil quality, and agricultural practices. Statistical models can be used to incorporate data from multiple sources, and machine learning algorithms can be trained to make predictions about future production. However, it's important to note that predicting agricultural production is subject to uncertainties such as unforeseen weather events, pest outbreaks, or market fluctuations, and predictions should be regularly monitored and updated throughout the growing season. As mentioned in the introduction, if we have data on the consumption of rice and tapioca by Keralite, we can inform policymakers about the current status of paddy and tapioca farming in Kerala. This information can help them develop effective policies and plans to improve the agricultural sector, increase crop yields, and support farmers.

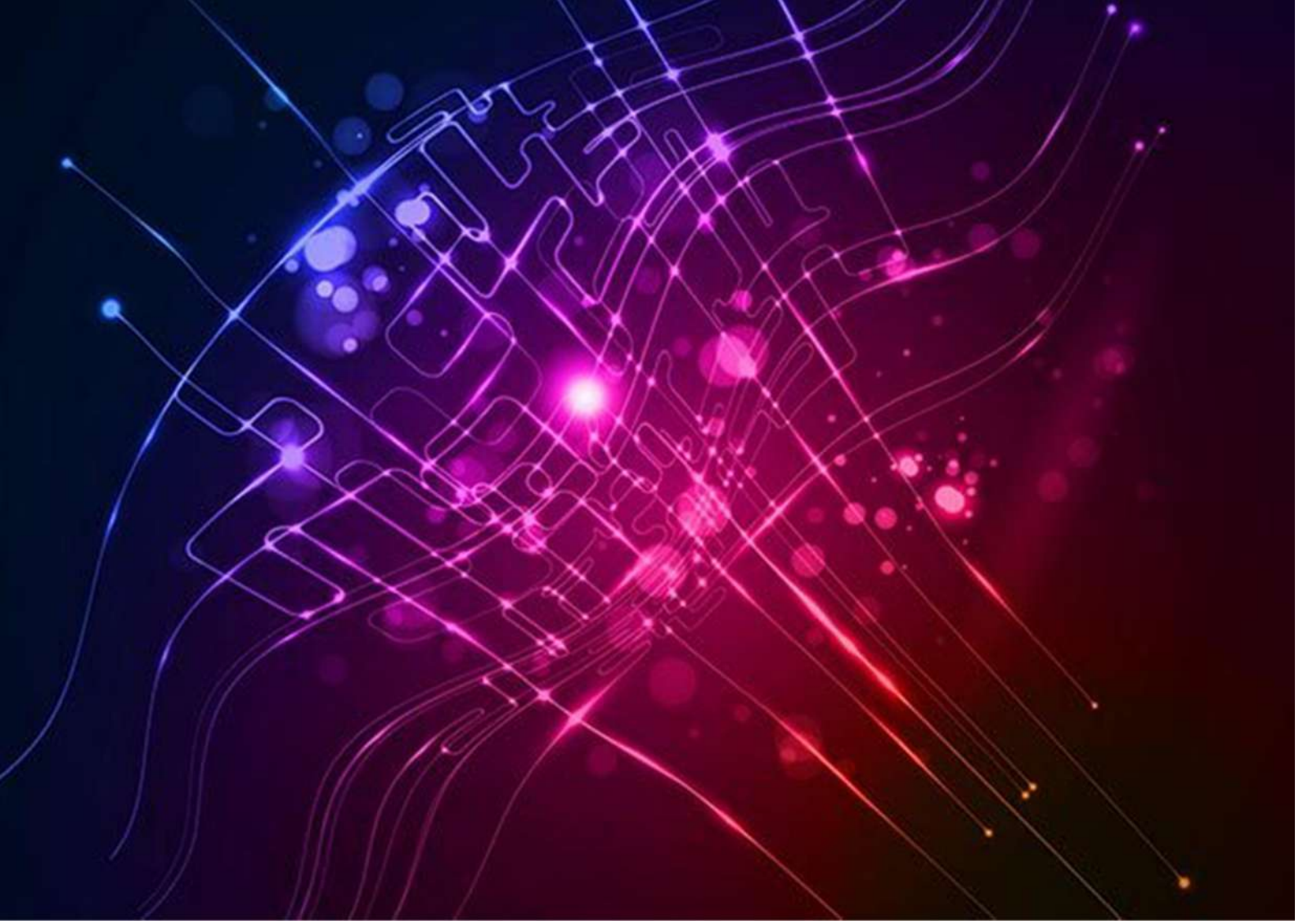
As the future aspects, we will work to evaluate non-linear techniques and more advanced deep learning methods can be used as they have the advantage during feature selection phase in machine learning.

References

[1] *A.P.S Manideep and Dr. Seema Kharb," A Comparative Analysis of Machine Learning Prediction Techniques for Crop Yield Prediction in India", Turkish Journal of Computer and Mathematics Education Vol.13 No.02 (2022), 120-133*

[2] *Dr. B M Sagar, Dr.N K Cauvery, Dr.Padmashree T. and Dr. R. Rajkumar, "Rice and Wheat yield prediction in india using decision tree and Random forest", Computational intelligence and Machine Learning, Vol3, issue2, 2022,PP 1-8.*

[3] *Mayank champaneri, Darpan Chachpara, Chaitanya Chandvidkar and mansing Rathod, "Crop Yield Prediction Using Machine Learning", Interational Journal on Science and Research, ISSN:2319-7064, 2019*



Prediction of Cost of cultivation of important crops in Kerala using Machine Learning

Submitted By
Smt. Shailamma K., Deputy Director

Introduction

India is predominantly an agricultural nation, with 60% of the population depending on agriculture for a living in one way or another. In emerging nations like India, where agriculture serves as the primary driver of economic growth, the development of agriculture is crucial. The development of agriculture is made feasible by increased investment to satisfy the rising capital requirements of contemporary agriculture. About 58% of India's population relies primarily on agriculture as a source of income, making India one of the key players in the global agricultural industry. Families are typically reliant on agriculture. Agriculture accounts for a significant portion of the nation's GDP. To achieve the strict standards, agricultural practices must be modernised. Prices for the crop have fluctuated significantly over the last few years. As a result the frequency of crop damage has increased. This prediction system's primary goal is to give farmers a better understanding of the cost of cultivation and how to maximise produce.

Objective

The most significant and prominent issue affecting Indian agriculture at the moment is the rise in cultivation costs. The cost of cultivation is made up of a variety of factors, including personnel costs, equipment rental fees, costs for seeds, fertilizer, and pesticides, as well as irrigation costs. Profitability and the cost-return ratio are directly impacted. Cost of cultivation of major crops are very essential for formulating proper support price policies, creating marketing facilities and assessing loss out of natural calamities, and the share of agriculture in GDP. We need reliable data on crop husbandry right from sowing to harvesting stages. For this, the department conducts an annual survey on Cost of Cultivation of Important Crops in Kerala during every agricultural year (July to June). The major components of the costs involved are seed/seedlings, fertilisers, labour, rent, equipment, irrigation charges etc. The main objective of the survey is to produce reliable estimates on production cost involved in major agricultural crops in the State. Major perennial, annual and seasonal crops are covered in the survey with a breakup of cost incurring during various stages from sowing to harvest.

Methodology

The survey covered all the districts in the state by considering taluk as a stratum. From each Taluk, required numbers of investigator zones were selected using circular systematic sampling method. From selected zones, cultivators and corresponding holdings are selected. The holdings are grouped under three size classes viz; small, medium and large according to the area. The investigators visited the selected holdings/cultivators and collected the required information in the prescribed format.

Cost incurred for growing the selected crops are classified under coat ‘A’, cost ‘B’, and cost ‘C’. Analysis of the data is carried out based on cost ‘A’. Cost ‘A’ is estimated based on all kind of expenses(paid out cost) actually incurred by the cultivators includes , Hired human labour, Animal labour, Machine labour, Seed/ seedlings, Farm yard Manure and Chemical fertilizers, Plant protection, Land tax and Irrigation Cess, Repair and maintenance charges of implements, machinery and buildings, Interest on working capital and Other expenses.

Proposed Methodology

Many factors are influential in agriculture, especially in cultivation and production. In this work, the impact of different factors that affect the cost of cultivation is considered and predicts how these factors affect the cost and forecast the cost. For that we use linear regression based prediction models. It is a machine learning algorithm based on supervised learning. It is used for finding out the relationship between variables and forecasting. Linear regression is a powerful tool for understanding and predicting the behavior of a variable. It is a statistical approach for modeling the relationship between a dependent variable and a given set of independent variables.

Method Used

The cost of cultivation is influenced by more than two items. So we use multiple linear regression based prediction models .Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several independent variables to predict the outcome of a dependent variable.

$$yi=\beta0+\beta1xi1+\beta2xi2+...+\beta pxip+\epsilon \quad \text{for } i=1,2,\ldots,n \text{ observations}$$

where, yi=dependent variable
xi=independent variables
β0=y-intercept (constant term)
βp=slope coefficients for each independent variable
ϵ=the model's error term (also known as the residuals)

Dataset Used

The actual cost incurred by the cultivators during 2019-20 is considered for the study.

Cost incurred for human labour, seed, fertilizer and other expenses are taken into consideration.

Farmer_id	Cost_fertilizer	Cost_seed	Cost_human	Cost_other expenses	Cost A
22797	16487.25	2964	30010.5	23465	72926.75
22799	10497.5	2315.63	31122	16981.25	60916.38
22800	5631.6	2667.6	5928	13585	27812.2
22801	12473.5	2593.5	39421.2	8645	63133.2
22802	11559.6	2964	33888.4	10374	58786
22803	11732.5	2717	35568	13173.33	63190.83
22805	11423.75	2778.75	51005.5	13585	78793
22806	12350	3334.5	13832	12350	41866.5
22807	11032.67	2717	39190.67	26676	79616.33
22810	18278	2964	82596.8	11362	115200.8
22811	21538.4	2964	41100.8	12844	78447.2

22969	258.83	1331.14	42049.16	0	43639.13
22970	2074.8	3334.5	24700	0	30109.3
22971	5086.37	3805.63	35037.41	2287.04	46216.44
22972	7782.75	2020.91	64556.82	8532.73	82893.2
22973	6249.1	2223	57427.5	10703.33	76602.93
22974	5705.7	2470	35074	3737.93	46987.63
22975	6570.2	2470	40960.83	6998.33	56999.37
22976	6995.04	2964	47177	4446	61582.04
22977	6007.04	2964	67431	988	77390.04
22978	6082.38	3087.5	10291.67	2058.33	21519.87
23004	6340.12	2774.94	16924.07	2363.27	28402.41
23005	5132.97	2881.67	10806.25	2315.63	21136.51
23006	6101.93	2593.5	14326	14614.17	37635.6
23007	7696.79	2761.6	39468.54	1440.83	51367.77
23008	9221.33	2815.8	34621.17	3087.5	49745.8
23009	6873.96	2851.01	21428.56	1182.45	32335.98
23010	5966.33	2861.79	28669.03	553.62	38050.78
23011	6923.72	2809.63	26019.91	14279.69	50032.94
23012	4802.78	2641.53	32212.92	1543.75	41200.97
23013	7512.92	2737.58	15725.67	926.25	26902.42
23014	7913.88	2849.76	11533.67	11713.98	34011.28
23015	4433.46	2836.64	26147.27	2026.17	35443.54
23016	10229.92	2881.67	27664	4528.33	45303.92
23017	7426.84	2475.61	28685.68	1571.82	40159.95
23018	8026.06	2613.6	37006.92	2584.88	50231.47
23063	17819.29	3920.63	2352.38	11467.86	35560.16
23065	0	4116.67	9495.78	8789.08	22401.53
23066	9297.64	4016.26	1606.5	22015.8	36936.21
23068	11866.19	4074.23	25973.2	12318.17	54231.78
23069	9756.5	3952	12967.5	24700	51376
23070	13959.03	3952	8962.57	11143.23	38016.83
23071	7431.48	4295.65	0	10943.17	22670.3
23072	11967.15	3892.12	8350.1	18510.03	42719.4
23073	2720.09	3952	29281.85	9555.81	45509.75
23074	11942.45	3952	11244.68	27089.73	54228.85
23075	15510.54	4249.46	15669.89	10437.74	45867.63
23077	23481.47	4610.67	0	11444.33	39536.47
23079	11761.32	3952	22271.17	25373.08	63357.56
23080	13667.33	4116.67	14964.08	10960.63	43708.71
23081	2346.5	3425.07	36007.11	5488.89	47267.57
23082	1743.11	2935.77	35709.14	8821.43	49209.46
23083	0	0	46899.13	9262.5	56161.63
23084	18125.17	2964	77774.13	4631.25	103494.5
23085	11826.36	3003.52	0	3705	18534.88
23086	4446	3249.42	34429.06	5488.89	47613.37
23087	18792.16	2903.51	29161.12	2520.41	53377.2

23088	1437.54	2845.44	40186.9	12350	56819.88
23089	1466.56	3260.4	35552.56	3087.5	43367.03
23090	12331.48	3062.8	40369.06	3087.5	58850.84
23091	19698.25	2964	48473.75	9262.5	80398.5
23092	16697.2	2529.28	36284.3	4940	60450.78
23093	1501.76	3003.52	36160.8	4940	45606.08
23094	16385.98	3319.68	25411.36	3952	49069.02
23095	1501.76	3003.52	39816.4	4940	49261.68
23188	6076.2	1704.3	19216.6	449.54	27446.64
23189	7558.2	1704.3	19043.7	5685.94	33992.14
23190	7551.14	1693.71	18313.29	6588.73	34146.87
23191	3705	2321.8	24041.33	2914.6	32982.73
23192	7753.06	2305.33	31698.33	5149.26	46905.99
23193	3705	2778.75	24082.5	2192.13	32758.38
23194	6058.76	1721.74	19760	6871.69	34412.19
23195	6175	2297.1	24453	6619.6	39544.7
23196	3705	2702.47	26734.12	12684.18	45825.76
23197	6422	2334.15	29393	6014.45	44163.6
23198	9962.33	2321.8	18936.67	444.6	31665.4
23199	6833.67	2766.4	30298.67	345.8	40244.53
23200	7934.88	2778.75	24082.5	339.63	35135.75
23201	12028.9	1704.3	19019	2820.74	35572.94
23202	5681	2717	32110	2799.33	43307.33
23280	5448.97	5987.88	65866.67	16092.42	93395.94
23281	6422	1915.51	58032.4	19029.08	85398.99
23282	2825.68	3952	54710.5	17784	79272.18
23283	5779.8	3952	51623	17784	79138.8
23284	12844	4149.6	54710.5	19918.08	91622.18
23285	6296.08	7749.02	53637.75	17435.29	85118.14
23286	8562.67	7978.1	37667.5	8892	63100.27
23287	7706.4	9386	40137.5	8892	66121.9
23288	2989.44	2815.8	40137.5	15067	61009.74
23289	7339.43	2681.71	35638.57	9668.29	55328
23290	9454.61	4665.56	37211.24	8892	60223.4
23291	2607.92	3009.14	12224.62	13541.12	31382.79
23292	6422	4347.2	37358.75	6158.53	54286.48
23293	4587.14	5645.71	35973.79	7621.71	53828.36
23294	6534.67	3293.33	36725	9360	55913
23332	91.08	1311.5	39651.15	0	41053.73
23623	7372.95	6402.24	18649.41	1469.65	33894.25
23624	7459.4	7706.4	31617.98	2692.3	49476.08
23625	9784.91	5157.36	32296.63	1512.88	48751.77
23626	10277.94	8892	0	1578.06	20748
23627	7788.38	5549.62	27959.54	6148.72	47446.26
23628	7968.22	6520.8	19143.44	1296.75	34929.21
23629	11271.43	5730.4	23644.4	1337.92	41984.15

23630	7254	7488	30854.88	2210	47806.88
23791	6103.15	5119.64	15045.2	1212.55	27480.52
23792	11411.4	0	16755.57	7698.17	35865.13
23825	9781.2	3293.33	53887.17	13132.17	80093.87
23828	8774.04	3226.12	28127.76	13443.86	53571.78
23838	9298.82	2847.76	74535.88	15037.94	101720.4
23841	5763.33	3046.33	66072.5	3972.58	78854.75
23854	4952.35	2881.67	52075.83	2202.42	62112.27
23855	11223.97	3359.93	75552.94	2324.71	92461.54
23857	9725.63	2701.56	61557.03	2470	76454.22
23859	2629.35	2948.06	31538.98	13651.4	50767.8
23862	3112.2	2855.32	53144.52	14128.4	73240.44
23864	4418.56	2881.67	65500.74	1738.15	74539.11
23866	5804.5	2881.67	0	2264.17	10950.33
23867	10003.5	3046.33	64014.17	3355.08	80419.08
23869	12903.62	2981.03	66093.79	2768.1	84746.55
23871	12926.33	2788.36	41715.56	1976	59406.24
23872	6327	2992.5	28500	1805	39624.5
23899	6125.6	2173.6	17425.85	14820	40545.05
23900	6379.04	2244.48	33973.24	23529.43	66126.2
23901	6520.8	2309.45	17425.85	15301.65	41557.75
23902	6298.5	2547.19	29832.2	17158.78	55836.67
23903	6520.8	2445.3	29769.68	24712.35	63448.13
23905	6549.91	2425.89	30023.73	16187.32	55186.86
23908	6276.71	2397.35	27818.01	11362	47854.07
23909	6257.33	2445.3	19899.97	20826.22	49428.82
23910	6520.8	2445.3	27393.54	17604.93	53964.56
23911	6422	2377.38	30498.33	25777.54	65075.24
23912	6520.8	2445.3	49350.6	10262.85	68579.55
23913	6586.67	2508	34341.09	18373	61808.75
24005	7765.8	3445.65	30779.7	1136.2	43127.35
24006	6648.42	3556.8	0	105427.8	115633.1
24007	8082.72	3968.07	30372.97	18449.7	60873.45
24008	7904	3445.65	27980.47	24230.7	63560.82
24012	8068.67	3747.08	26358.1	17518.7	55692.55
24013	8001.5	5190.9	28762.5	23868	65822.9
24015	7931.68	3570.43	41628.02	9633	62763.13
24016	8016.27	3563.99	21837.05	18188.18	51605.49
24021	7632.3	4075.5	23641.43	10091.71	45440.94
24024	8080.43	3952	39685.4	9728.27	61446.1
24103	4890.6	1852.5	24482.64	13338	44563.74
24104	6175	1789.86	34723.19	4474.64	47162.68
24105	3803.8	1914.25	26379.6	13338	45435.65
24106	4693	1837.68	25564.5	13585	45680.18
24122	3774.36	3981.94	22142.68	16573.95	46472.93
24123	2543.9	3919.61	30551.07	28166.1	65180.67

24124	2964	3952	13554.13	22971	43441.13
24125	2136.55	4149.6	41002	18599.1	65887.25
24126	2408.25	3952	10464.57	22806.33	39631.15
24127	1905.43	4327.44	30204.57	16549	52986.44
24128	2124.2	4149.6	43348.5	18426.2	68048.5
24129	2321.8	11065.6	27417	19019	59823.4
24130	1994.53	4149.6	31893.88	16437.85	54475.85
24131	3735.88	3952	17266.84	26398.13	51352.84
24132	1778.4	4149.6	59156.5	19537.7	84622.2
24133	1778.4	4034.33	11449.48	15787.99	33050.2
24134	1970.51	3872.96	22559.33	16933.22	45336.03
24135	2390.96	3651.65	47374.6	15264.6	68681.81
24136	3237.46	4149.6	35409.21	24700	67496.28
24291	8953.75	2223	27664	0	38840.75
24292	8424.46	2434.71	13408.57	4410.71	28678.46
24293	8205.89	2209.28	18223.11	0	28638.28
24294	9185.31	2223	12967.5	0	24375.81
24295	9553.09	2615.29	9298.82	4358.82	25826.03
24296	7958.89	2209.28	17454.67	0	27622.83
24297	8584.76	2216.98	21808.29	4819.51	37429.54
24298	7862.83	2470	22888.67	0	33221.5
24387	1818.4	1591.1	35958.96	20078.22	59446.69
24388	8595.6	1630.2	25490.4	33345	69061.2
24389	3705	1552.57	31492.5	7851.07	44601.14
24408	1852.5	1698.13	20377.5	1929.69	25857.81
24411	30737.78	1536.89	114388.4	2195.56	148858.7
24419	0	3175.71	0	793.93	3969.64
24423	14166.18	3196.47	19372.55	1816.18	38551.37
24425	12350	1605.5	42237	1852.5	58045
24426	7591.62	4358.82	37776.47	1343.97	51070.88
24507	10574.69	2223	21921.25	0	34718.94
24508	13210.34	2331.24	15819.1	0	31360.67
24792	6916	2815.8	26182	419.9	36333.7
24793	7853.33	1425	48133.33	285	57696.67
24794	9914.79	2435.21	25047.89	226.13	37624.01
24795	12411.75	2686.13	22230	308.75	37636.63
24796	7706.4	2213.12	41496	395.2	51810.72
24797	16631.33	1976	23876.67	205.83	42689.83
24798	10081.63	2117.14	30244.9	504.08	42947.76
24799	4322.5	2778.75	28713.75	370.5	36185.5
24800	7904	2305.33	16466.67	0	26676
24801	10492.56	2578.68	29047.2	395.2	42513.64
24802	8315.67	2223	16796	230.53	27565.2
24803	9445.71	2328.86	59985.71	271.43	72031.71
24943	2408.25	2686.13	28405	0	33499.38
24945	7059.26	2667.6	14573	247	24546.86

24966	6027.4	5142.36	3809.16	0	14978.91
24967	6452.88	5335.2	13832	6175	31795.08
24968	7821.67	5928	0	0	13749.67
24969	6211.32	6538.24	17798.53	11805.15	42353.24
24970	7671.25	5335.2	5890	9025	27921.45
24971	6100.9	5409.3	9880	9056.67	30446.87
24972	6257.33	5335.2	17652.27	12350	41594.8
24973	6399.55	5254.36	21197.09	12350	45201
24974	6768.75	3420	4560	11875	26623.75
24975	6704.29	12702.86	0	16937.14	36344.29
24976	6114.12	5009.58	21360.28	7827.46	40311.44
24977	5991.06	3783.83	0	10510.64	20285.53
24978	6526.2	4168.13	53124.3	13507.81	77326.44
24979	6801.45	5154.78	9593.62	9665.22	31215.07
24980	6650.51	5706.1	12997.22	9906.42	35260.24
25000	9255.09	2037.75	17808.7	13313.3	42414.84
25060	9818.25	2470	54447.39	2470	69205.64
25061	9818.25	2470	65208	2470	79966.25
25062	9818.25	2470	84844.5	2470	99602.75
25063	9818.25	2470	0	16584.29	28872.54
25064	9818.25	2470	68748.33	14820	95856.58
25065	9818.25	2470	133544.7	2470	148302.9
25066	9818.25	2470	56995.25	2470	71753.5
25067	9818.25	2470	50006.67	2470	64764.92
25068	9818.25	2470	65193.03	2470	79951.28
25069	9818.25	2470	5513.04	11757.2	29558.49
25070	9818.25	2470	100446.7	11801.11	124536
25071	9818.25	2470	72524.31	11413.1	96225.66
25072	9818.25	2470	61926.43	11644.29	85858.96
25073	9818.25	2470	41166.67	11879.52	65334.44
25074	9818.25	2470	41707.71	12350	66345.96
25164	9061.61	2766.4	24288.33	3869.67	39986.01
25165	7449.52	2558.92	18870.8	17413.5	46292.74
25166	13399.75	3087.5	30566.25	5094.38	52147.88
25167	23897.25	2766.4	18772	14869.4	60305.05
25168	4949.88	2766.4	10472.8	12646.4	30835.48
25169	12010.38	3087.5	34425.63	12890.31	62413.81
25170	6669	0	21406.67	10497.5	38573.17
25171	6669	2593.5	2305.33	11279.67	22847.5
25172	9600.07	2334.15	19142.5	15972.67	47049.38
25173	7281.56	2593.5	26774.8	1976	38625.86
25174	7133.36	2845.44	21637.2	12295.66	43911.66
25175	8336.25	2334.15	25243.4	8348.6	44262.4
25176	9880	2694.55	17514.55	14056.55	44145.64
25177	12610.72	3293.33	22778.89	12830.28	51513.22
25257	4663.36	4742.4	17981.6	38927.2	66314.56

25258	1963.65	5928	38532	12350	58773.65
25259	3290.04	6916	39520	30134	79860.04
25260	7904	4446	70642	20748	103740
25261	4683.12	7904	49400	33098	95085.12
25262	6051.5	5928	42484	26676	81139.5
25263	8892	7904	64714	30134	111644
25264	12350	3952	55328	44830.5	116460.5
25265	9484.8	4763.57	33874.29	7057.14	55179.8
25266	17191.2	4446	83980	37050	142667.2
25267	16466.67	4940	75746.67	10374	107527.3
25418	4831.03	3632.35	45767.65	27605.88	81836.91
25419	4364.5	3733	7839.29	18988.51	34925.3
25420	4285.45	4116.67	46106.67	27170	81678.78
25421	4237.17	3761.14	3929.55	29009.03	40936.88
25422	4701.23	3705	7492.33	27894.53	43793.1
25423	3798.86	3754.4	5532.8	28098.72	41184.78
25424	4363.67	3705	6916	28322.67	43307.33
25425	4024.2	3762	4788	28473.4	41047.6
25426	4010.22	3757.93	4940	28574.37	41282.52
25427	5051.15	3705	8645	28429.7	45830.85
25428	5219.93	3705	10374	27680.47	46979.4
25429	4664.18	3705	7492.33	19554.17	35415.68
25430	4557.15	3705	6916	28108.6	43286.75
25431	3835.91	3754.4	5532.8	27970.28	41093.39
25432	5878.6	3705	8068.67	28322.67	45974.93
25535	8637.59	2000.7	31214.63	9311.9	51164.82
25539	7150.65	1896.96	22674.6	49.4	31771.61
25541	9434.38	1909.79	42626.6	12477.32	66448.09
25542	0	2533.33	48070	0	50603.33
25544	9305.73	2521.46	22044.75	51.46	33923.39
25545	6800.52	1847.69	37916.1	12401.32	58965.64
25547	5063.5	2315.63	35104.88	185.25	42669.25
25548	10739.93	2515.74	37461.67	274.44	50991.78
25549	16631.33	2352.38	18230.95	235.24	37449.9
25550	7976.04	2408.25	37976.25	0	48360.54
25551	9262.5	2408.25	45942	164.67	57777.42
25552	6986.57	2470	32833.36	617.5	42907.43
25554	11115	2247.7	30875	247	44484.7
25562	2470	2470	20377.5	1852.5	27170
25566	22293.33	2964	22800	506.67	48564
25656	5335.2	5928	87932	8398	107593.2
25657	12844	3952	74100	21242	112138
25658	5310.5	3087.5	84597.5	6175	99170.5
25659	10406.93	0	50223.33	14243.67	74873.93
25798	12012	3952	47125	11050	74139
25799	8172	3952	35444.5	31282.55	78851.05

25800	8343.97	3952	54610.16	31801.25	98707.38
25801	8966.24	3952	0	31353.29	44271.54
25802	12254.45	3923.77	46102.55	23994.29	86275.05
25803	9305.73	3952	35444.5	34086	82788.23
25804	10324.35	3952	35451	32500	82227.35
25805	11844.93	3952	35285.71	33874.29	84956.93
25806	10151.44	3952	35435.49	32933.33	82472.27
25807	9071.32	3952	35444.5	21983	70450.82
25808	12526.43	3952	34844.64	25546.86	76869.93
25809	11584.3	3952	35321	36062	86919.3
25810	13128.67	3952	35043.13	21365.5	73489.29
25811	11998.19	3952	34965.94	19358.63	70274.75
25812	8916.7	3952	35444.5	34333	82646.2
26105	9163.7	4940	20501	5975.75	40580.45
26106	8284	5700	28215	6692.75	48891.75
26107	8645	4940	39211.25	7799.03	60595.28
26108	8645	4940	33437.63	7332.81	54355.44
26109	7330.96	4940	29244.8	2302.04	43817.8
26110	9447.11	5092.78	25379.04	19745.57	59664.51
26111	9163.7	4940	36556	8195.46	58855.16
26112	9027.05	5330.94	26210.43	6022.18	46590.6
26113	5873.11	5488.89	29228.33	6729.93	47320.26
26114	7780.5	4940	30504.5	6589.96	49814.96
26115	7330.96	4940	26676	26219.54	65166.5
26116	9163.7	4940	24823.5	7528.56	46455.76
26117	7868.71	5645.71	29675.29	7062.79	50252.5
26118	9163.7	5269.33	21900.67	8564.31	44898.01
26119	6125.6	4940	31430.75	26542.62	69038.97
26175	17868.91	0	42067.19	4564.35	64500.45
26176	10806.25	4322.5	41990	7286.5	64405.25
26177	7179.47	4610.67	25655.07	3392.13	40837.33
26178	14731.79	0	40508	16363.75	71603.54
26179	14608.29	0	71136	5469.29	91213.57
26180	14981.73	3675.6	32462.86	15158.15	66278.33
26181	14126.15	2694.55	26747.85	21329.57	64898.13
26182	18209.78	0	35283.36	21503.11	74996.26
26183	20289.29	0	34756.43	7210.05	62255.76
26184	11171.46	0	39971.66	13479.14	64622.26
26185	0	0	22543.27	0	22543.27
26186	14491.64	3051.18	31479.42	21358.24	70380.47
26187	15869.75	3952	26589.55	21242	67653.3
26537	3378.96	3458	23341.5	3952	34130.46
26539	3487.64	3458	17784	3458	28187.64
26544	3375.26	3458	13338	3458	23629.26
26545	3202.44	3458	18854.33	3952	29466.77
26546	2818.89	3396.25	21689.69	3859.38	31764.2

26547	3406.62	3458	24020.75	2964	33849.37
26549	3448.12	3458	22724	2881.67	32511.79
26550	3604.22	3458	26305.5	4446	37813.72
26552	3380.81	3458	21118.5	0	27957.31
26553	3371.55	3458	25564.5	3952	36346.05
26555	3029.87	3458	25276.33	2634.67	34398.87
26557	3195.56	3458	24206	2284.75	33144.31
26559	2252.64	3359.2	20105.8	2371.2	28088.84
26560	2704.65	3458	19760	3458	29380.65
26594	2872.26	3528.57	21594.86	2540.57	30536.26
26603	13008.67	2881.67	52899.17	4446	73235.5
26608	16820.8	2532.01	51673.64	11471.55	82498
26610	0	0	43661.37	0	43661.37
26611	13023.64	2619.7	44534.85	11638.94	71817.12
26613	13008.67	2881.67	49688.17	12918.1	78496.6
26615	12251.2	3112.2	43719	15363.4	74445.8
26617	13832	2881.67	52693.33	12802.83	82209.83
26618	14573	2377.38	44614.38	11763.38	73328.13
26621	13008.67	2881.67	52281.67	15799.77	83971.77
26623	16796	2939.3	60045.7	24774.1	104555.1
26625	16842.14	2375	51435.71	12024.29	82677.14
26628	14573	2377.38	39742.3	11763.38	68456.05
26631	16796	2401.39	67033.06	12061.83	98292.28
26634	16796	2161.25	66690	12380.88	98028.13
26636	14980.39	3143.64	49720.78	9905.66	77750.47
26884	11323.58	3293.33	50772.22	4147.77	69536.9
26888	11176.75	3087.5	49400	4145.48	67809.73
26892	11281.31	2964	48782.5	4146.31	67174.12
26896	11297.61	3163.82	43710.67	4146.27	62318.38
26900	11281.31	2964	49400	4146.31	67791.62
26904	11370.3	3113.45	49815.13	442.11	64740.98
26936	12751.38	2699.36	64925.71	9368.36	89744.8
26937	13187.06	2634.67	65866.67	5516.33	87204.72
26939	12725.51	2703.65	68425.68	9963.45	93818.28
26940	12839.88	2717	67184	9830.6	92571.48
26942	12813.13	2593.5	66690	12226.5	94323.13
26943	13056.99	257739.1	68014.49	11813.04	350623.7
26944	12870.46	2540.57	64572.86	11644.29	91628.18
26946	12653.14	2559.82	63321.82	11856	90390.77
26947	12813.13	2662.97	68696.88	12697.34	96870.31
26948	12587.5	2660	159663.3	12540	187450.8
27027	7883.34	4149.6	45497.4	19215.7	76746.04
27028	12328.59	4150.92	39717.6	18998.42	75195.53
27029	12325.3	4150.59	48646.65	20068.75	85191.29
27030	13529.01	4215.47	29640	21409.96	68794.44
27031	15.25	5.13	56.48	24.39	101.24

27032	12325.3	4150.42	45571.5	16816.58	78863.81
27033	7879.3	4149.6	45523.09	23427.95	80979.94
27034	12316.41	4150.39	45685.12	20069.24	82221.16
27035	12317.5	4150.47	43528.33	16849.08	76845.38
27036	12340.12	4150.26	46156.07	18093.57	80740.02
27037	12312.95	4150.42	45633.25	18394.64	80491.26
27038	7879.3	4149.6	45531.98	20068.75	77629.63
27039	12325.3	4150.42	58724.25	16816.58	92016.56
27040	12310.48	4150.48	45667.56	19244.59	81373.11
27041	12331.65	4150.45	45673.83	18516.53	80672.46
27087	13235.47	2990.41	39613.21	776.73	56615.82
27481	7125	3087.5	82792.5	1425	94430
27482	7410	2593.5	78299	1482	89784.5
27483	6062.73	2470	69204.91	8577.64	86315.27
27484	8027.5	2624.38	84937.13	1235	96824
27485	9115.48	2646.43	83744.76	17936.9	113443.6
27486	8468.57	2822.86	60956.07	19760	92007.5
27487	8336.25	2531.75	72865	14523.6	98256.6
27488	10034.38	3087.5	82204.69	1543.75	96870.31
27489	8159.82	2536.16	84906.25	21171.43	116773.7
27545	4350.39	2470	92505.48	14509.26	113835.1
27546	4310.15	2470	56717.38	35024.6	98522.13
27547	6567.29	2470	95203.97	35023.15	139264.4
27548	25137.54	2470	66460.64	37671.03	131739.2
27549	4322.5	2470	55155.1	35024.6	96972.2
27550	13290.95	2822.86	43166.19	37638.1	96918.1
27551	4310.39	2470	54654.8	34827	96262.2
27552	4293.31	2470	51448.98	35029.09	93241.38
27553	4314.78	2470	93242.5	35390.47	135417.8
27554	4324.4	4940	64752	35024.6	109041
27555	4878.25	2470	58230.25	35765.6	101344.1
27556	4322.5	2470	54932.8	35024.6	96749.9
27557	4310.15	2470	52034.67	35024.6	93839.42
27558	4318.24	2470	50784.05	35022.9	92595.19
27559	4314.78	2470	95017.81	35027.69	136830.3
27828	6029.4	790.4	23498.8	26260	56578.6
27836	7821.67	2161.25	16796	4116.67	30895.58
27837	11115	0	34394.75	6175	51684.75
27838	7014.8	0	24359.14	2470	33843.94
27839	7916.67	2216.67	28563.33	20583.33	59280
27840	6711.81	0	26037.09	1646.67	34395.57
27841	8710.87	988	20850.09	42283.11	72832.07
27842	5645.71	0	26287.86	44107.14	76040.71
27843	6962.31	1018.88	25826.94	3087.5	36895.63
27844	7014.8	988	31665.4	3087.5	42755.7
27845	9468.33	1097.78	30079.11	26072.22	66717.44

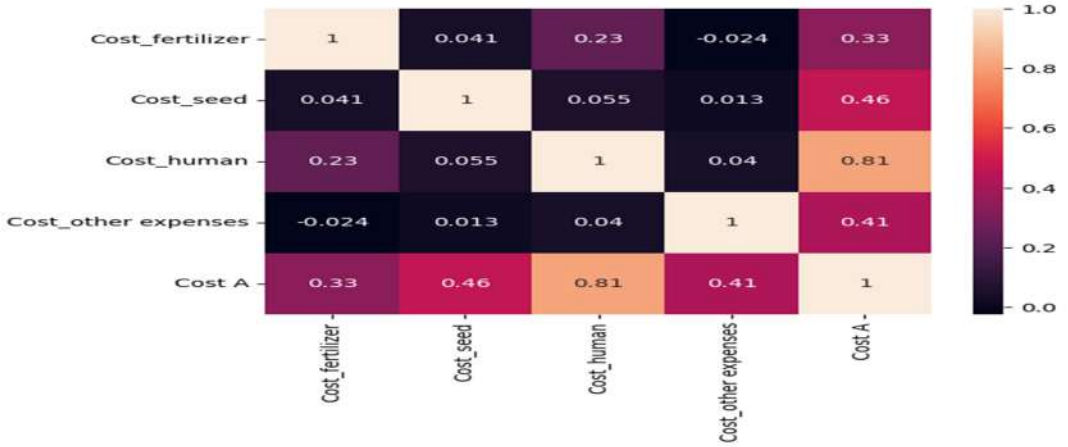
27846	4273.1	1729	38803.7	2470	47275.8
27847	8068.67	2223	16334.93	38696.67	65323.27
27848	6916	988	21406.67	2470	31780.67
27849	439.11	2401.39	11307.11	37050	51197.61
29626	9634.24	4273.1	17876.63	3099.85	34883.81
29627	9627.51	4259.38	16768.56	3098.48	33753.92
29628	9634.24	4273.1	16641.63	3096.76	33645.72
29629	9638.27	4268.16	11164.4	3099.03	28169.86
29630	9667.58	4281.33	17413.5	3103.97	34466.38
29631	9642.88	4262.51	18666.14	3102.32	35673.86
29632	9642.88	4262.51	16831.29	3098.09	33834.77
29633	9714.51	4248.4	31492.5	3097.38	48552.79
29634	9654.28	4271.2	15903	3098.14	32926.62
29635	9627.51	4259.38	13502.67	3098.48	30488.03
29636	9641.23	4281.33	70395	3099.85	87417.42
29637	9645.35	4281.33	55575	3098.48	72600.16
29638	9787.79	4215.47	50717.33	3108.91	67829.49
29639	9230.74	4234.29	67395.71	3098.09	83958.83
29640	9614.48	4199	72556.25	3118.38	89488.1

Tools and libraries used

- numpy
- matplotlib.pyplot
- pandas
- seaborn
- sklearn

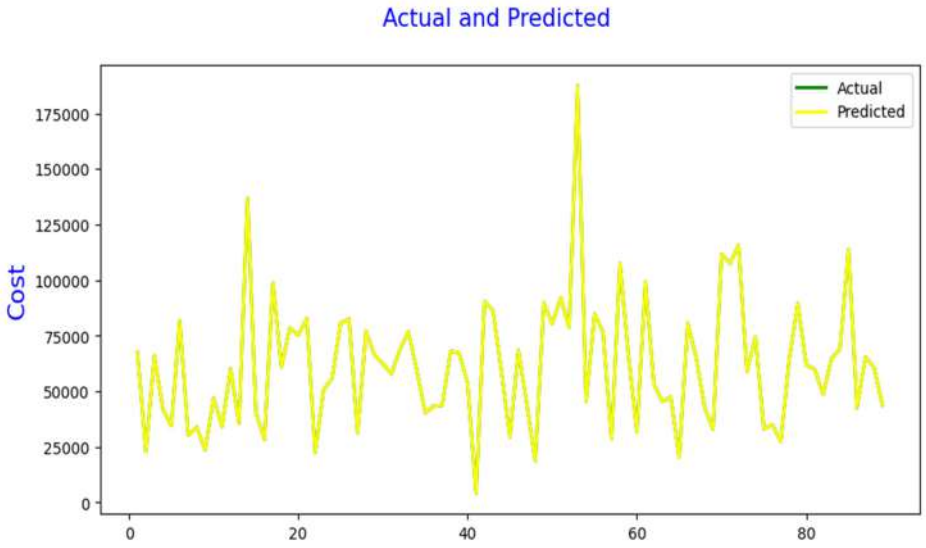
Data Processing and Result

Missing data and null values were treated. Study the correlation between the dependent variable (cost A) and independent variables (human labour, seed, fertilizer and other expenses). For that we use heatmap. From the heatmap it is clear that the dependent variable and independent variables are correlated.

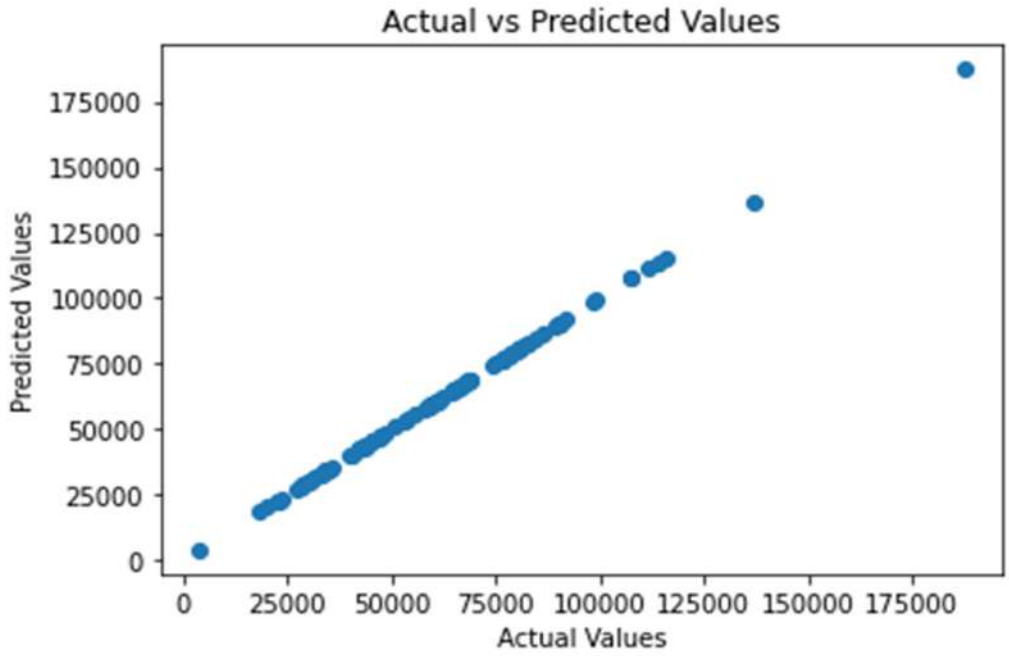


Split the data randomly in to train (80%) and test (20%).Using the train data, train the machine and construct a model .Fit the test value in the model and we got the future predicted value. Then calculate the mean square error and regression coefficients**. Mse =0.004932663589052374** and the regression coefficients are **0.99999989, 1.00000004, 1.00000001, 1.00000003.** It is evident that the model that we construct is correct and using it we can predict the future value.

Actual and Predicted Values		
	y_test	**y_pred**
327	67653.30	67653.298573
233	22847.50	22847.498535
122	66126.20	66126.189426
102	41984.15	41984.148164
71	34412.19	34412.188575
253	81678.78	81678.789970
12	30109.30	30109.298738
96	33894.25	33894.248445
330	23629.26	23629.258646
261	46979.40	46979.399467
328	34130.46	34130.458787
90	60223.40	60223.408733



To check the accuracy we draw scatter diagram with actual value and predicted value. From the diagram it is clear that almost all the values lying in a straight line. It means that actual value and predicted values are almost equal and the model that we create is a perfect one.

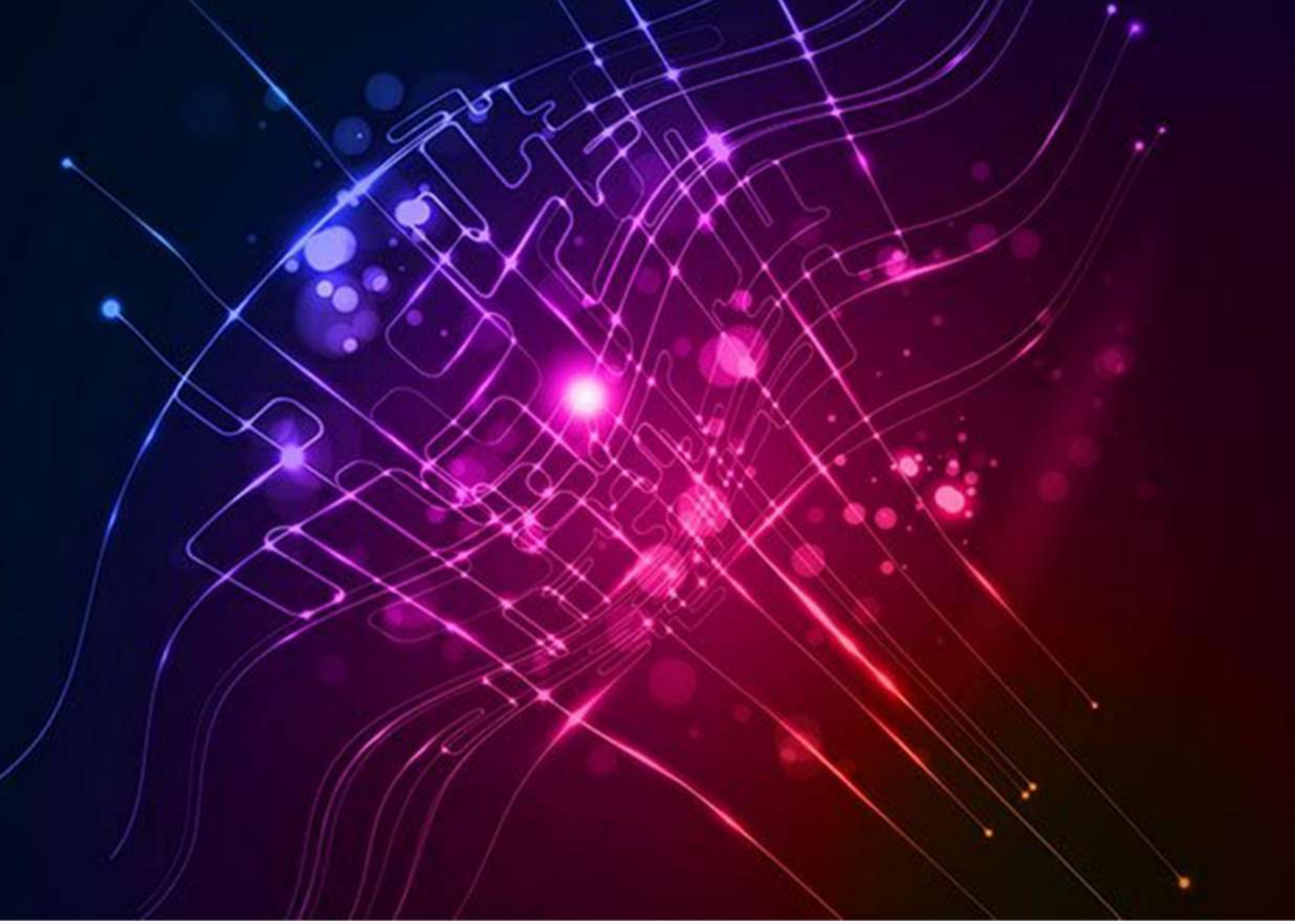


Conclusion

It is aimed to discuss the application of artificial intelligence in agriculture, even in the common case of small available datasets. The approach was applied and verified in the specific case of paddy. Also aims to use the model to predict other important crops.

REFERENCE

1. *Report on Cost of Cultivation 2011-12 to 2020-21 published by Directorate of Eonomics and Statistics*
2. *Multilinear regression model based studies*



Forecasting of Cement Price in Kerala

Submitted By
Sri.Sijith K.S., Research Officer

Introduction of the project

The construction sector is a key contributor to the economy of Kerala, with cement being a vital input for construction activities. The price of cement, have a significant impact on the profitability and competitiveness of construction firms, as well as the affordability and accessibility of housing for the general public. The price of cement in the state is subject to fluctuations due to various factors, including changes in demand, supply constraints, and economic conditions. Therefore, accurate forecasting of cement prices is essential for stakeholders in the construction industry, including cement manufacturers, distributors, retailers, contractors, and policymakers. Such forecasting can help these stakeholders make informed decisions regarding production, procurement, pricing, and investment. The motivation of this project is to develop and test a machine learning-based approach for forecasting the price of cement in Kerala, using historical data on cement price

Accurate forecasting of cement prices can be crucial in resolving disputes between contractors and house owners in the construction industry. When contractors agree to construct a building for a fixed price, they base their calculations on the estimated costs of materials, including cement. If the price of cement increases unexpectedly during the construction process, the contractor may face difficulties in meeting their obligations and may seek to pass on the additional costs to the house owner. On the other hand, if the price of cement decreases, the contractor may benefit from the cost savings without passing them on to the house owner. This can lead to disputes over payment, quality, and timelines.

By providing reliable forecasts of cement prices, stakeholders in the construction industry can mitigate the risks of such disputes. Contractors can use the forecasts to negotiate more accurate and transparent contracts with house owners, based on realistic estimates of material costs. House owners can use the forecasts to monitor the progress of construction, to ensure that the quality and quantity of materials used are consistent with the agreed terms. Additionally, policymakers can use the forecasts to identify potential bottlenecks in the supply chain and to implement measures to mitigate the impact of price fluctuations on the construction industry and the wider economy. In summary, accurate forecasting of cement prices can help to reduce the uncertainties and risks associated with construction projects, and can facilitate smoother and more efficient transactions between contractors and house owners.

Objectives of the project

The main objective of the study is to forecast the price of cement in construction sector of Kerala using historical data of cement prices through machine learning based approach. The purpose is to help the construction firm, building developers and householders to predict cement prices for their upcoming projects and hence construction cost of the project. The rising cost of building materials is also responsible for time overrun experienced on most construction projects. Another impact is the high dispute rate between contractor and owner due to the rising cost of building materials. In this study machine learning algorithms, may predict the changes in cement prices and forecast the possible prices. So the main objective is to develop a reliable model for forecasting the prices of cement in Kerala for the next 2 years.

Literature review

A study named as 'Application of machine learning in cement price prediction through a web based system', has conducted by A.O Afolabi, College of Science and Technology, Covenant University, Ota, Nigeria. This study aims to predict cement prices in the Nigerian construction industry through a machine learning based approach. The study aimed to develop a web based learning platform that uses machine learning algorithms on historical data of cement prices, petrol prices, diesel prices, interest rate, and exchange rate to predict future prices of cement products. The web based learning platform was developed using HTML, CSS, MySQL, and PHP. For building a reliable machine learning model, python language was used to train the system with the available data.

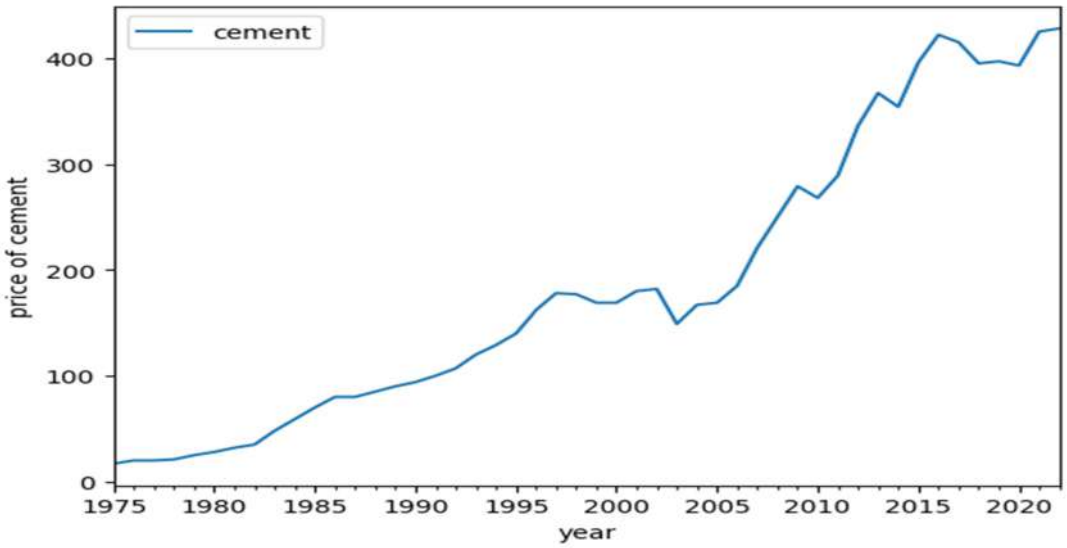
There is another study conducted by olga vainer finance director at ilanot tel aviv israel named as 'monthly sales of french champagne - arima models'. The aim of the study was to predict the number of monthly sales of champagne for the Perrin Freres label (named for a region in France).The dataset provides the number of monthly sales of champagne from January 1964 to September 1972.The validation dataset is about 11% of the original dataset. ARIMA model was developed by selecting the appropriate parameters, including the order of differencing (d), the autoregressive order (p), and the moving average order (q). The model was then validated using residual analysis, and the forecast accuracy was evaluated using RMSE.

Data set used

Labour and housing section of Department of economics and statistics collects price of building materials since 1974-75. The following is the price of cement in Kerala from 1974-75 to 2021-22 and is the annual average prices of cement in all districts.

year	price of cement in Rs for 50kg	year	price of cement in Rs for 50kg
1974-75	17	1998-99	169
1975-76	20	1999-00	169
1976-77	20	2000-01	180
1977-78	21	2001-02	182
1978-79	25	2002-03	149
1979-80	28	2003-04	167
1980-81	32	2004-05	169
1981-82	35	2005-06	185
1982-83	48	2006-07	221
1983-84	59	2007-08	250
1984-85	70	2008-09	279
1985-86	80	2009-10	268
1986-87	80	2010-11	289
1987-88	85	2011-12	336
1988-89	90	2012-13	367
1989-90	94	2013-14	354
1990-91	100	2014-15	396
1991-92	107	2015-16	422
1992-93	120	2016-17	415
1993-94	129	2017-18	395
1994-95	140	2018-19	397
1995-96	162	2019-20	393
1996-97	178	2020-21	425
1997-98	177	2021-22	428

Plot of cement price per 50 kg in Kerala from 1974-75 to 2021-22



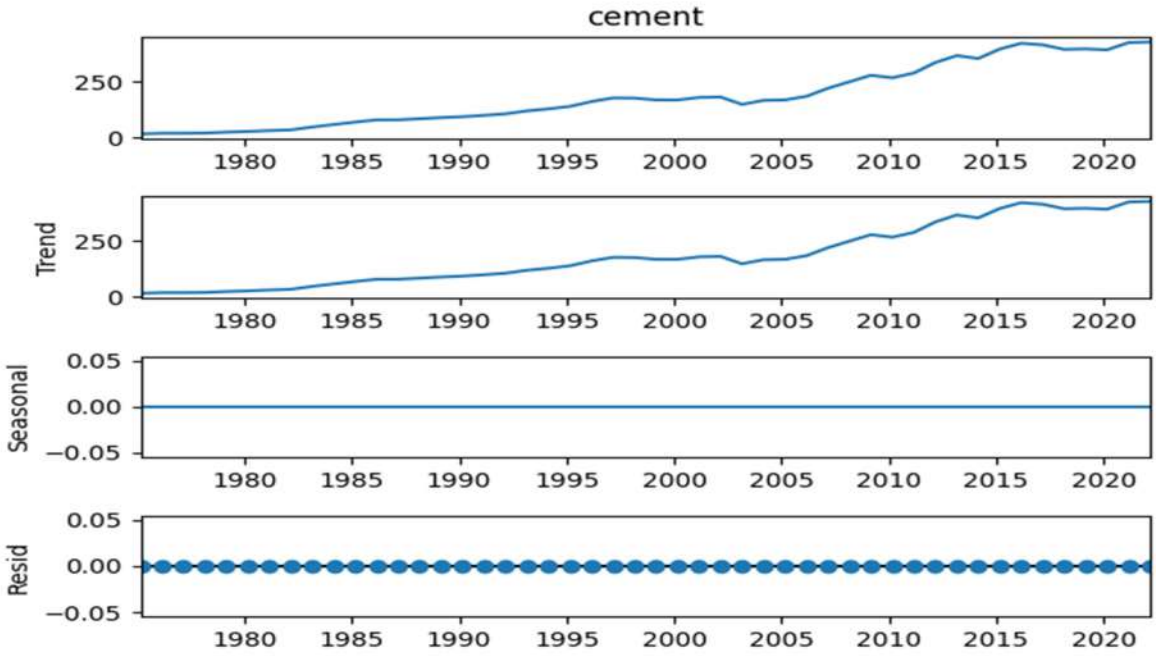
Method and Methodology

The following methodology was adopted in the study:

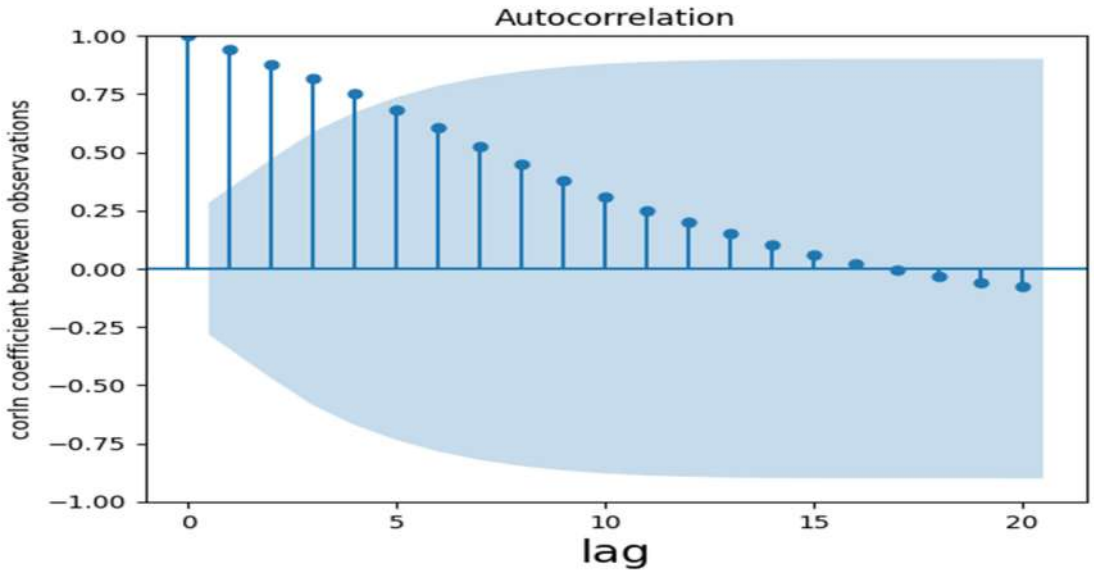
1. Collect data on cement prices from reliable sources government publications, or relevant websites
2. Data cleaning and pre processing
3. Divide the data into training data set and test data set.
4. Component analysis of the time series to examine the trends, seasonality, and other patterns in the data.
5. Choose appropriate time series forecasting model.
6. Estimate the model parameters by fitting the chosen model to the time series data.
7. Train the chosen model using the training data
8. Evaluate the performance of the model on the test data.
9. Check the accuracy of the model
10. Check the adequacy and validity of the model through residual analysis and conducting statistical tests.
11. Use the fitted model to make future predictions of cement prices.

The study has done using python machine language. The data has split into training dataset and testing dataset. The training data set contains price of cement from 1974-75 to 2017-18 and testing data set contains data from 2018-19 to 2021-22. We divide a data set into a training set and a testing set in order to evaluate the performance of a predictive model. The training set is used to fit or train the model, which involves estimating the model parameters using the data available. The model is then tested on the testing set, which is a subset of the data set that was not used in the training process. The purpose of using a separate testing set is to assess how well the model will perform on new, unseen data. By evaluating the model on the testing set, we can estimate the model's ability to generalize to new data. If the model performs well on the testing set, it is likely to perform well on new, unseen data.

Component analysis is a technique used in time series forecasting to break down a time series into its underlying components, which include trend, seasonality, and noise or error. Component analysis is a powerful tool in time series forecasting 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.On conducting component analysis of our time series we can understand that our time series has a trend but no seasonality and noise. So AR or ARIMA model time series forecasting is suitable.



Also, there is evidence of autocorrelation in the data. ARIMA models are commonly used for time series forecasting when there is evidence of autocorrelation in the data. Autocorrelation refers to the dependence of a time series on its past values.



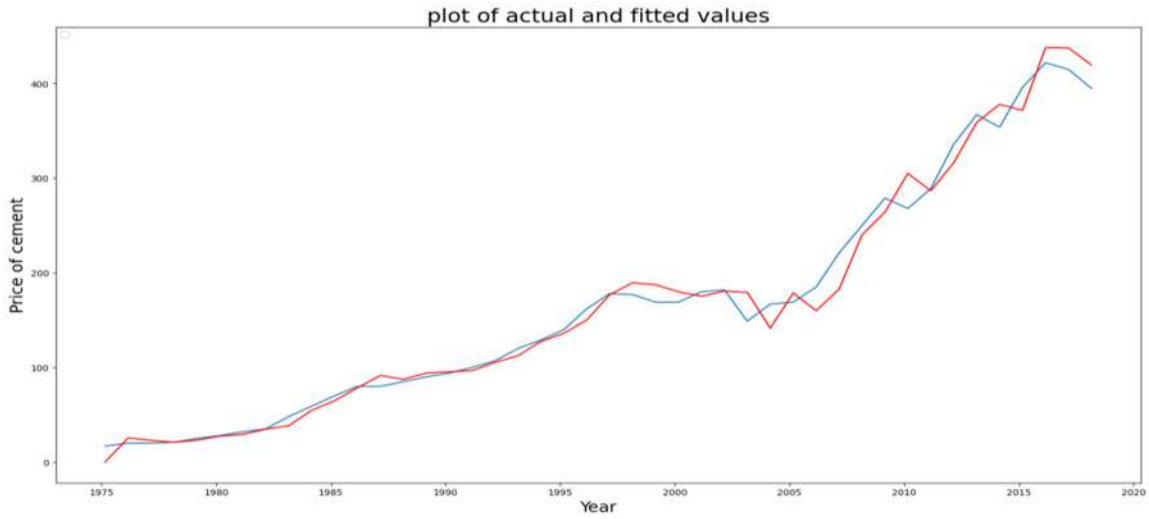
ARIMA (Auto Regressive Integrated Moving Average) is a popular time series modeling technique used for forecasting future values based on past observations and fitting errors. It is a combination of two models: the autoregressive (AR) model and the moving average (MA) model. There are 3 parameters p, d, q in an ARIMA model.

p-order of autoregressive terms
d-number of differences done to make data stationary
q-order of moving average terms

Stationarizing a time series is a common technique used in time series analysis to simplify and improve the accuracy of statistical models, facilitate interpretation and forecasting and reduce variability. A stationary time series is one whose statistical properties such as mean, variance, autocorrelation, etc. are constant over time. ADF (Augmented Dickey-Fuller) test is a statistical test used to determine whether a time series is stationary or not. 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.

The technique of differencing can be used to makes the data stationary. After first differencing, we conducted Augmented Dicky fuller test and got the p-value = 0.85, which is greater than 0.05, indicating it is non-stationary. After Second differencing, we again conducted Augmented Dicky fuller test and got, p-value = 0.004, which is less than 0.05 indicating now it is stationary.

Here we had done two times differencing to stationarize the data. So the value of parameter d of the ARIMA model can be taken as 2. Library Pmdarima in python has an Auto_arima function, using this we can find possible ARIMA models with different orders and corresponding AIC values.In general, a lower AIC value indicates a better model fit. (AIC stands for Akaike Information Criterion, which is a statistical measure used to compare the relative quality of different statistical models).In our study we got best model as ARIMA (3,2,0). Fitting of model was done using Python in test data set. Following plot is the fitted price (red color) and actual price (blue color) of cement.

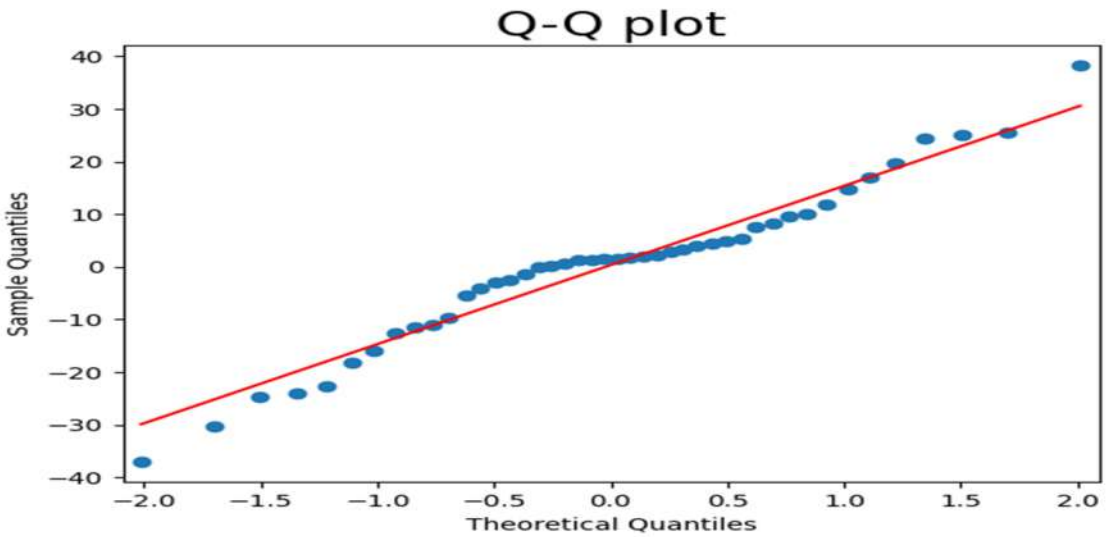


After fitting the ARIMA model to the data, the next step is to evaluate the model's performance. The accuracy of the model can be checked on test data by comparing the predicted values from the model to the actual values in the test set. MAPE (Mean Absolute Percentage Error) is a commonly used measure 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:

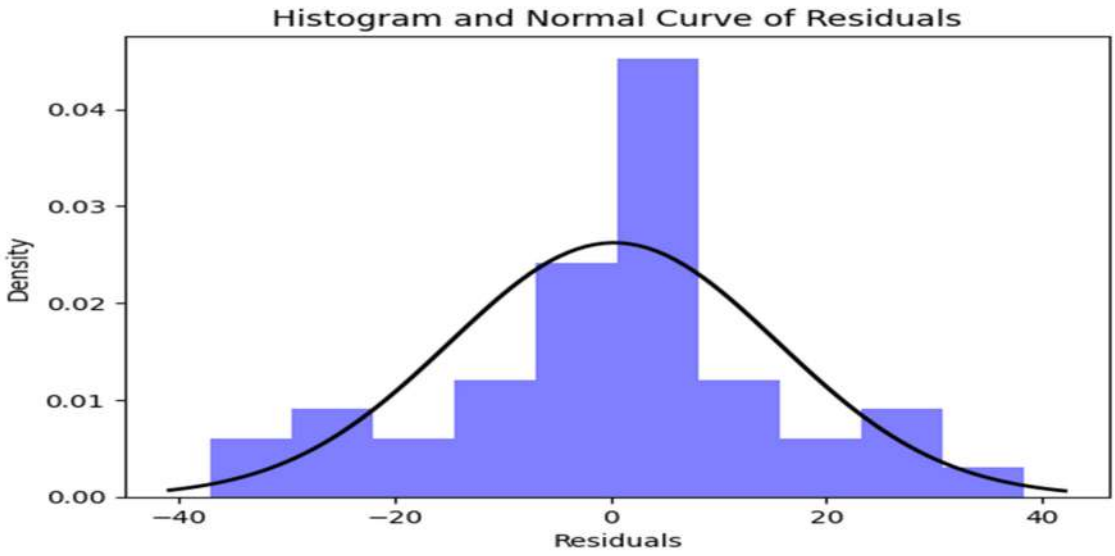
MAPE = (1/n) * ∑(|Actual - Forecast| / Actual) * 100

Where, n is the number of observations in the data set. In our case MAPE is 3.87, it means that on average, the forecasted values were off by 3.87% from the actual values. And R Sq= 0.54, it means that 54% of the variability observed in the price of cement is explained by the model.

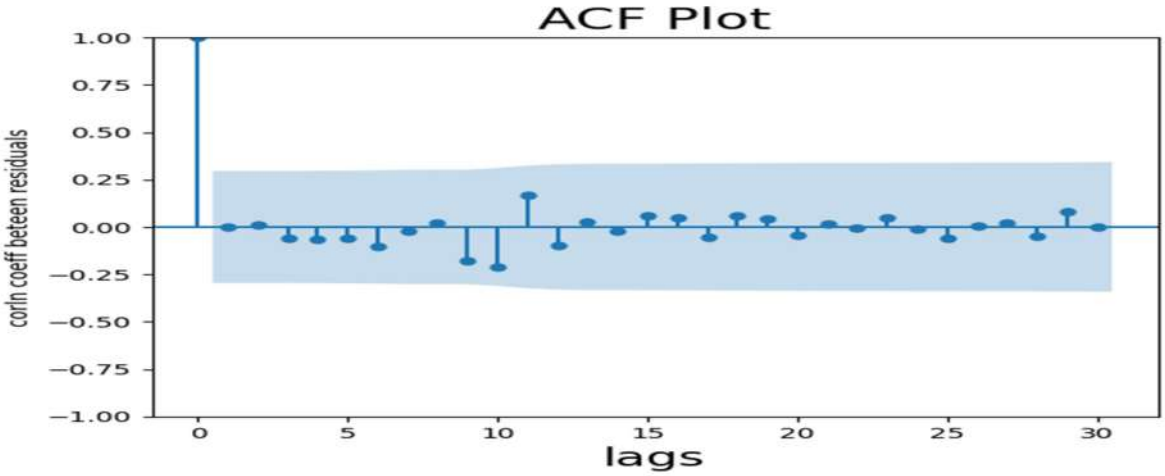
Finally we check the adequacy of the model. Residual analysis is an important technique for evaluating the adequacy of time series forecasting models. Residuals are the differences between the actual values and predicted values. Some of the key conditions that should be satisfied in residual analysis are residuals should be normally distributed and residuals should not have any auto correlation. Q-Q plots can be used to check whether the residuals are normally distributed or not. If the residuals fall approximately along the straight line on the Q-Q plot, then the residuals are normally distributed.



The figure shows that residuals fall approximately along a straight line on the Q-Q plot. So residuals are normally distributed. Shapiro-Wilk's test is well known statistical test for checking the normality. The null hypothesis of the Shapiro-Wilk's test is that the population or sample follows a normal distribution. As by the test if the probability of the test statistic, p-value is greater than chosen significance level (0.05), we accept the null hypothesis. In our case it was 0.58.so we accept the null hypothesis that residuals are normally distributed.



Next we have to check whether residuals have any significant auto correlation or not. The Autocorrelation Function (ACF) plot of residuals is a graphical tool used to diagnose the presence of autocorrelation in the residuals of a time series model. The plot shows the autocorrelation on the y-axis and the lag on the x-axis. If the autocorrelation values are within the confidence intervals (usually the blue shaded area on the plot), it suggests that there is no significant autocorrelation in the residuals. If the autocorrelation values fall outside the confidence intervals, it indicates that there is significant autocorrelation and further investigation is needed.



From the ACF plot it is clear that correlation coefficients between residuals at any lag except 0 are inside the blue band and hence statistically insignificant, means there is no significant auto correlation between residuals. So we can conclude that the chosen model is adequate.Hence we can make predictions using this model. Using the model we got the forecasted price of cement for next two years as;

Year	Forecasted price of cement
2022-23	429
2023-24	437

The results of this analysis suggest that the price of cement is likely to increase by 1-2% over the next two year.

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

Result

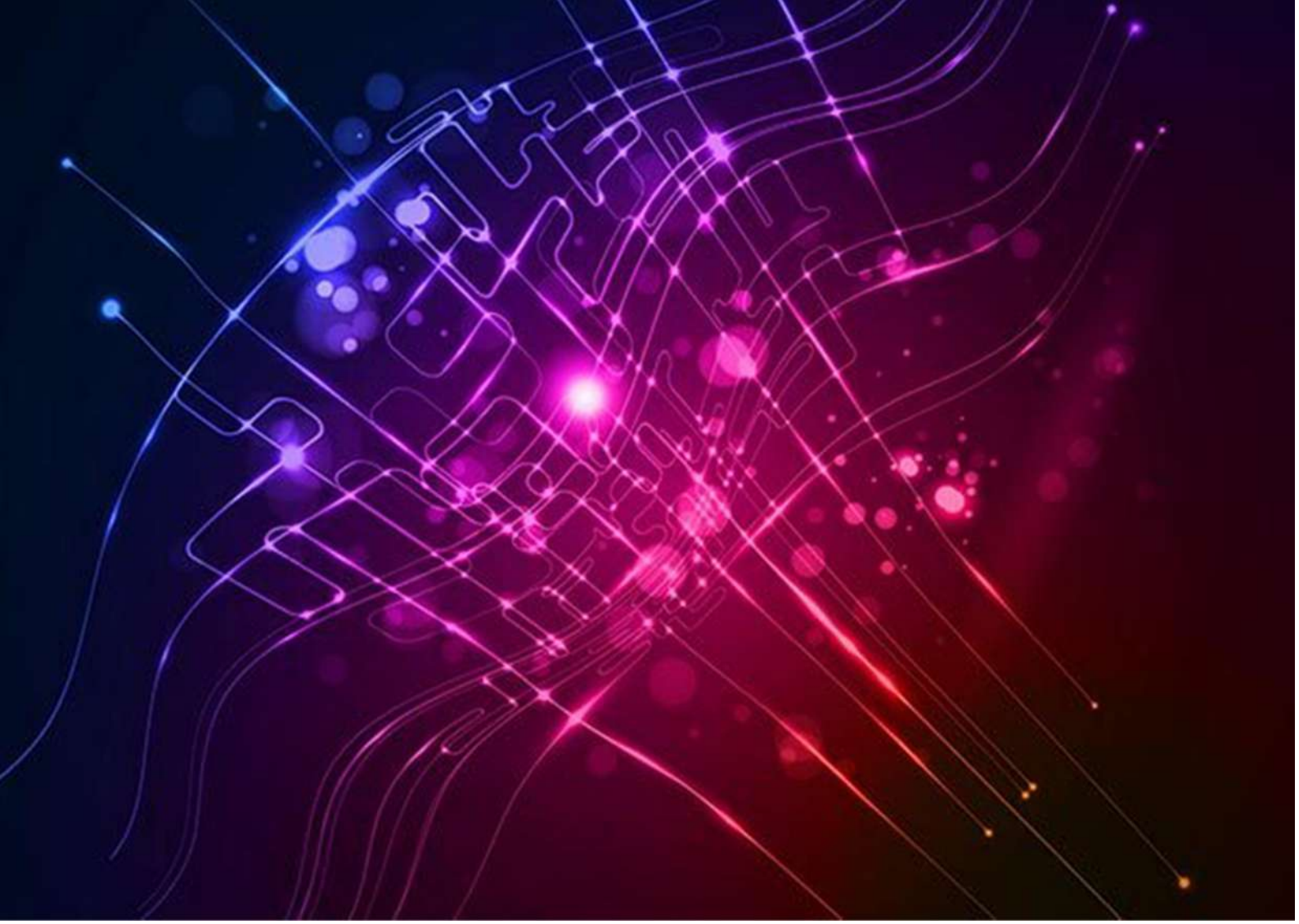
The aim of the study was to develop a model that predicts the future price of cement based on past prices of cement using python language. ARIMA(3,2,0) is the best model for the data and that predicts the average price of cement in Kerala for the next 2 years. The results of our analysis suggest that the price of cement is likely to increase by 1-2% over the next two year. However, there is a moderate level of uncertainty associated with the forecast due to factors such as fluctuations in raw material prices and changes in government policies that could impact the cement market. This study helped to forecast the future price of cement based on historical data at reasonable accuracy (MAPE=3.87). The results indicated that the ARIMA model could effectively capture the patterns and trends in the cement price data, providing accurate and reliable forecasts for the construction industry.

Conclusion

Statistical model of the project provides a useful tool for predicting the prices of cement in Kerala, but it is important to keep in mind the potential uncertainties and risks associated with any forecasting model. One of the limitations of the study was here only considered the past prices of cement for forecasting the future prices. But there are some other factors that affect the price of the cement, (eg. petrol price, govt.policies & regulations, currency exchange rate etc.) which are not considered in the study. However this study helped to forecast the future price of cement with a reasonable accuracy. For future research, other prediction model can be developed to consider other building materials such as steel reinforcement, bricks, paints, timbers etc. However, it should be noted that the availability of data may be impact the model's performance and the model may need to be updated to maintain its accuracy.It can conclude that price of any building materials and hence project cost can be forecast with a reasonable accuracy using python and machine learning approach. The study recommends the increased use of machine learning in predicting building materials prices to aid in project cost estimation.

References

1. *Monthly Sales of French Champagne - ARIMA Models, Olga Vainer Finance Director at Ilanot Tel Aviv-Yafo, Tel Aviv District, Israel*
2. *Application of machine learning in cement price prediction through a web based system, by A.O Afolabi, Oluwamayowa Abimbola Department of Building Technology, College of Science and Technology, Covenant University, Ota, Nigeria*
3. *Data of Economics and statistics department Kerala*



Index of Industrial Production

-An Overview

Submitted By
Smt. Sudarsha R., Deputy Director
and **Smt. Brijila R.,** Statistical Assistant Grade II

1. Introduction of the project
2. Objectives of the project
3. Methodology and method used
4. Datasets used
5. Tools and libraries used
6. Result with inference
7. Conclusion
8. Reference

Contents

IIP (Index of Industrial Production)

1. Introduction

The Index of Industrial Production (IIP) is an index that shows the growth rates in different industry groups of the economy in a fixed period of time. It is a composite indicator of the general level of industrial activity in the economy. In India, it is calculated and published by the Central Statistical Organisation (CSO) every month.

IIP measures the changes in the industrial production and the general level of industrial performance in the economy. It is used as a short term macro-economic indicator of industrial growth. IIP is considered to be one of the lead indicators for short term economic analysis because of its strong relationship with economic fluctuations in the economy. It is also used as major data source for the compilation of annual and quarterly estimation and forecasts of GDP. Index of industrial production is compiled for three major sectors.

a. Manufacturing
b. Mining and Quarrying
c. Electricity

2. Objectives

India being an Agrarian based Country in the pre independence period, Government was compelled to focus in Manufacturing sector so as to attain economic development and adapting most modern technologies in Agriculture. Hence, the Industrial development began immediately after independence. IIP is a measure to assess the same.

A trend in industrial sector is to be analysed and growth in each sector is to be evaluated so that policy makers can address the issues faced by the industrial production sector and support with appropriate infrastructure both physical as well as financial. Government already have taken more initiatives to promote the sector by easing he licensing procedure, etc…

3. Methodology

In case of 'manufacturing' sector, the base used for selection of items in the item basket for IIP is the item-wise production data of Annual Survey of Industries (ASI), 2011-12 provided by Central Statistical Organization (CSO). The item basket for manufacturing sector was identified with base year 2011-12. A total of **23** NIC 2- digit level item groups were selected for manufacturing sector. Since the number of items in 'mining & quarrying' sector in Kerala is very few, all the mining items in the item basket are covered. The Electricity sector consists of single item and total electricity generated in Kerala is taken.

Weighing Diagram

The relative importance of various economic activities is different and these differentials need to be reflected while measuring the performance of the entire industrial sector. With a view to achieving this, each item included in the item basket is given appropriate weight.

Total weight, which is taken as 1000, was apportioned first to three industrial sectors, i.e., Manufacturing, Mining & Quarrying and Electricity. The sectoral weight of manufacturing sector was then allocated to its 2 & 3- digit industry groups in proportion to their GVA figures. Finally, the weight allocation at item level has been done in proportion to item-wise GVO. In case of mining & quarrying sector the total weight of the sector was apportioned to the items included in the item basket in proportion to their value of output furnished by IBM.

4. Datasets used

The monthly production data of 'mining & quarrying' and 'electricity' for compiling State level IIP are collected from Indian Bureau of Mines (IBM), Nagpur and Central Electricity Authority respectively. In case of 'manufacturing' sector, after finalization of item basket (containing selected item groups and items within the group), item wise list of factories along with their production in the base year has been prepared from out of the list of factories from ASI survey 2011-12. A total of 190 registered manufacturing units are selected for data collection and the department collects monthly production data of 134 manufacturing items.

The data source is from the office of Directorate of Economics and Statistics, Government of Kerala.

Data used is as follows,

Year	Q1	Q2	Q3	Q4	QGen	Base year
1981-1982	175.11	180.32	195.33	171.26	180.51	1970-71
1982-1983	174.12	167.12	165.98	127.13	158.59	1970-71
1983-1984						
1984-1985						
1985-1986						
1986-1987	178.73	164.23	179.82	161	170.95	1970-71
1987-1988	180.74	187.36	169.91	162.73	175.19	1970-71
1988-1989	128.14	140.23	158.65	142.32	142.34	1980-81
1989-1990	139.41	160.74	179.87	187.88	166.98	1980-81
1990-1991						
1991-1992	180.89	236.68	226.79	220.48	216.21	1980-81
1992-1993	183.29	232.1	221.91	231.94	217.31	1980-81
1993-1994	210.46	272.22	272.64	305.97	265.32	1980-81
1994-1995	217.81	267.36	258.78	276.38	255.08	1980-81
1995-1996						
1996-1997						
1997-1998						
1998-1999						
1999-2000	243.91	328.21	412.83	420.03	351.24	1980-81
2000-2001					360.2	1980-81
2001-2002	237.46	344.74	307.23	319.71	302.29	1980-81
2002-2003	245.78	241.51	286.32	295.81	267.35	1980-81
2003-2004	212.36	220.41	310.87	306.72	262.59	1980-81
2004-2005	198.05	205.64	211.73	207.83	205.81	1993-94
2005-2006	202.83	209.72	210.79	209.72	208.27	1993-94
2006-2007	267.18	284.61	246.94	223.16	255.47	1993-94
2007-2008						
2008-2009						
2009-2010	93.73	99.27	108.87	115.57	104.36	2004-05
2010-2011	107.7	108.27	109.73	110.77	109.12	2004-05
2011-2012	130.4	124.3	127.8	135.13	129.41	2004-05
2012-2013	110.23	110.47	103.73	112.33	107.19	2004-05
2013-2014	102.7	130.43	123.07	111.6	116.95	2004-05
2014-2015	125.67	134.4	126.33	143.23	131.41	2004-05
2015-2016	90.1	91.8	79.3	83	86.05	2011-12
2016-2017	90.4	103	91.2	94.6	94.8	2011-12
2017-2018	81	93.87	119.91	127.93	105.68	2011-12
2018-2019	117.79	111.2	115.37	114.15	114.63	2011-12
2019-2020	100.91	116.57	102.3	90	102.45	2011-12
2020-2021	68.42	83.86	97.63	102.95	88.21	2011-12

Data processing-imputed the missing values

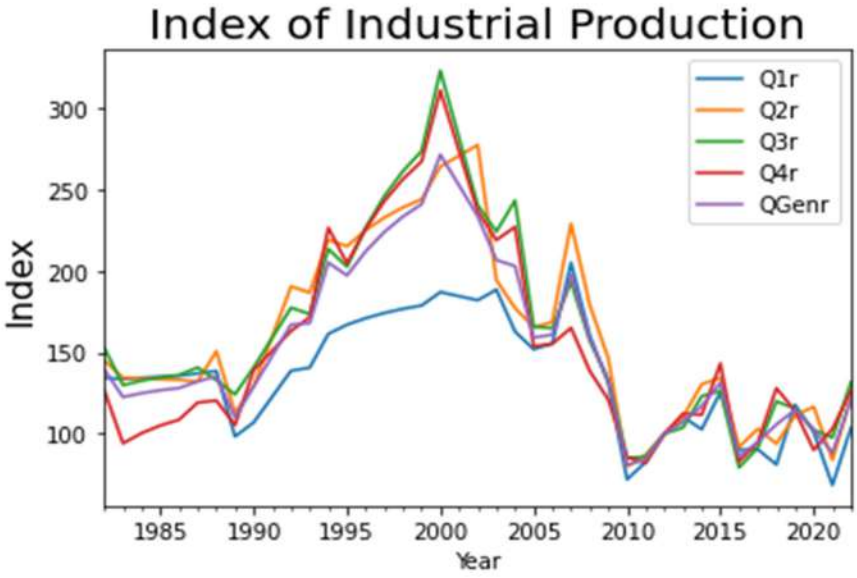
Year	Q1	Q2	Q3	Q4	QGen
1982	175.11	180.32	195.33	171.26	180.51
1983	174.12	167.12	165.98	127.13	158.59
1984	120.4	120.4	120.4	120.4	120.4
1985	130.7	130.7	130.7	130.7	130.7
1986	142.1	142.1	142.1	142.1	142.1
1987	178.73	164.23	179.82	161	170.95
1988	180.74	187.36	169.91	162.73	175.19
1989	128.14	140.23	158.65	142.32	142.34
1990	139.41	160.74	179.87	187.88	166.98
1991	160.15	198.71	203.33	204.18	212.6
1992	180.89	236.68	226.79	220.48	216.21
1993	183.29	232.1	221.91	231.94	217.31
1994	210.46	272.22	272.64	305.97	265.32
1995	217.81	267.36	258.78	276.38	255.08
1996	230.86	297.79	335.8	348.2	303.1
1997	230.86	297.79	335.8	348.2	304.6
1998	230.86	297.79	335.8	348.2	303.3
1999	230.86	297.97	335.8	348.2	303.5
2000	243.91	328.21	412.83	420.03	351.24
2001	240.685	336.475	360.03	369.87	360.2
2002	237.46	344.74	307.23	319.71	302.29
2003	245.78	241.51	286.32	295.81	267.35
2004	212.36	220.41	310.87	306.72	262.59
2005	198.05	205.64	211.73	207.83	205.81
2006	202.83	209.72	210.79	209.72	208.27
2007	267.18	284.61	246.94	223.16	255.47
2008	141.5	141.5	141.5	141.5	141.5
2009	145.6	145.6	145.6	145.6	145.6
2010	93.73	99.27	108.87	115.57	104.36
2011	107.7	108.27	109.73	110.77	109.12
2012	130.4	124.3	127.8	135.13	129.41
2013	110.23	110.47	103.73	112.33	107.19
2014	102.7	130.43	123.07	111.6	116.95
2015	125.67	134.4	126.33	143.23	131.41
2016	90.1	91.8	79.3	83	86.05
2017	90.4	103	91.2	94.6	94.8
2018	81	93.87	119.91	127.93	105.68
2019	117.79	111.2	115.37	114.15	114.63
2020	100.91	116.57	102.3	90	102.45
2021	68.42	83.86	97.63	102.95	88.21
2022	103.7	122.2	131.8	126.4	121

5. Tools and libraries used

```
import pandas as pd
import numpy as np
from statsmodels.tsa.stattools import adfuller
from pmdarima import auto_arima
from pandas import read_csv
from pandas import datetime
from matplotlib import pyplot
from statsmodels.tsa.arima.model import ARIMA
from sklearn.metrics import mean_squared_error
from math import sqrt
```

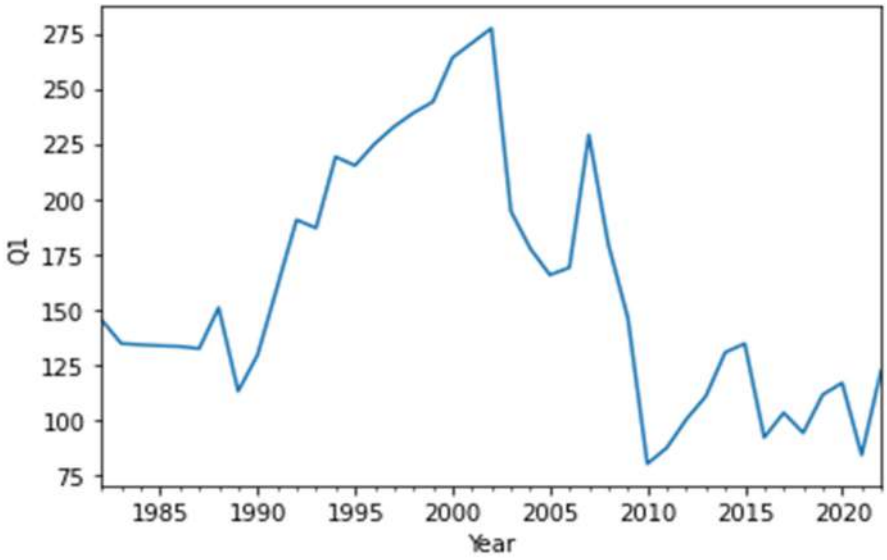
6. Result with inference

The pictorial representation of the data



Method 1: To predict the first Quartile Q1 using ARIMA Model

Pictorial representation of Q1



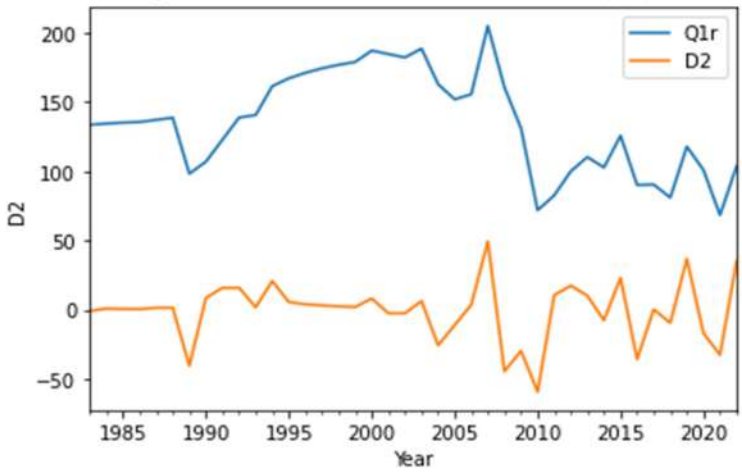
Check the Stationarity using Augmented Dickey-Fuller test

1. ADF Statistic: -1.8603658775718286
2. P-Value: 0.3509296637282311
3. n_lags: 0
4. Num of observations used for ADF Regression and Critical Values Calculation: 40
5. Critical Values:
   1%, -3.6055648906249997
   5%, -2.937069375
   10%, -2.606985625

Here the P-Value is greater than 0.05, the data is non-stationary.

Being non- stationary, took first difference and on testing stationarity is found.

Graph of Q1& First difference (D2)



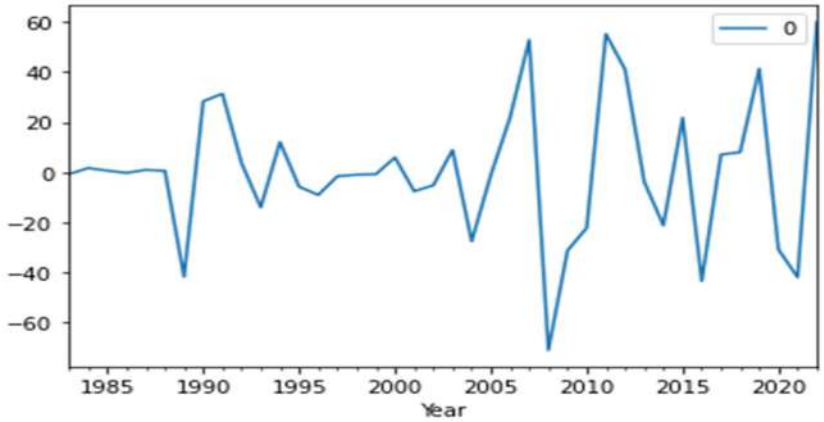
- 1.ADF Statistic: -6.437777441920769
- 2.P-Value: 1.6365132616833867e-08
- 3.n_lags: 0
- 4.Number of observations used for ADF Regression and Critical Values Calculation : 39
- 5.Critial Values:
  1%, -3.610399601308181
  5%, -2.939108945868946
  10%, -2.6080629651545038

It is Stationary

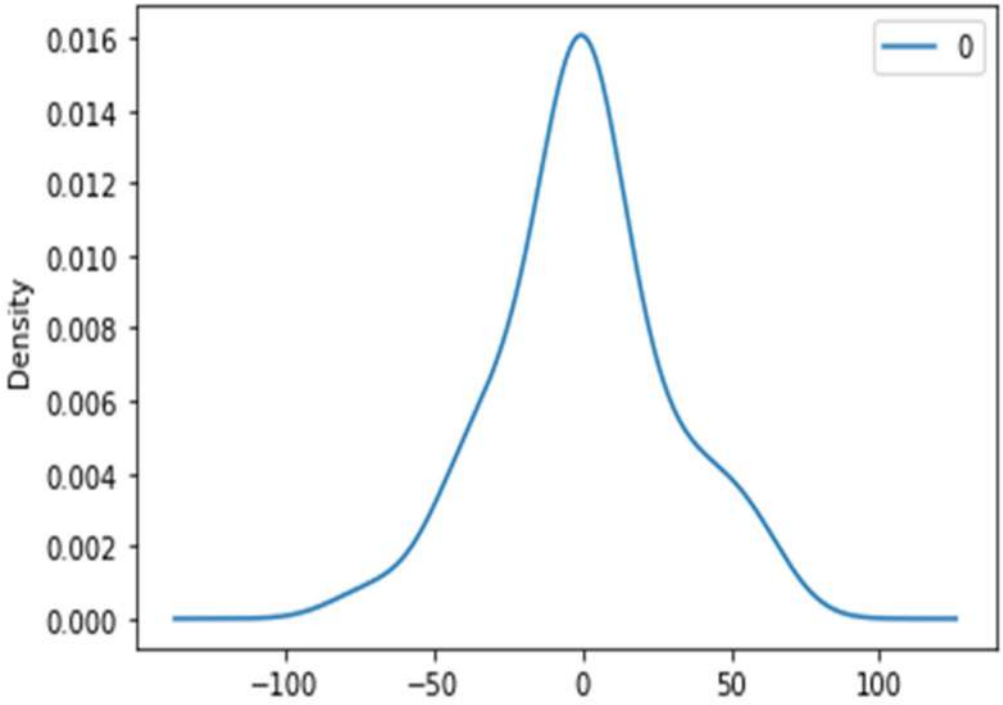
Using Auto ARIMA MODEL

- Best model: ARIMA(1,1,0)
  For verifying the best model, find its Residual and density graph of Residuals

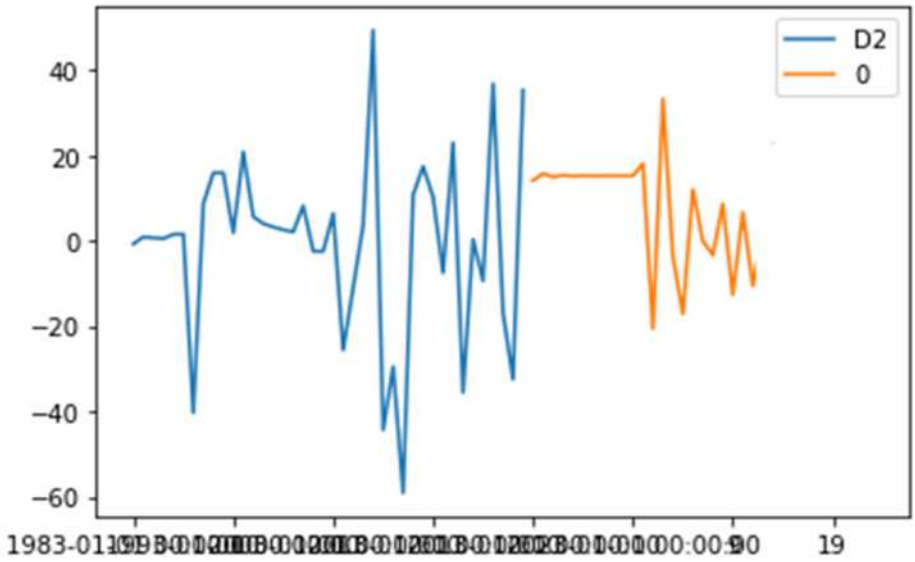
Residual



Density Plot of residual



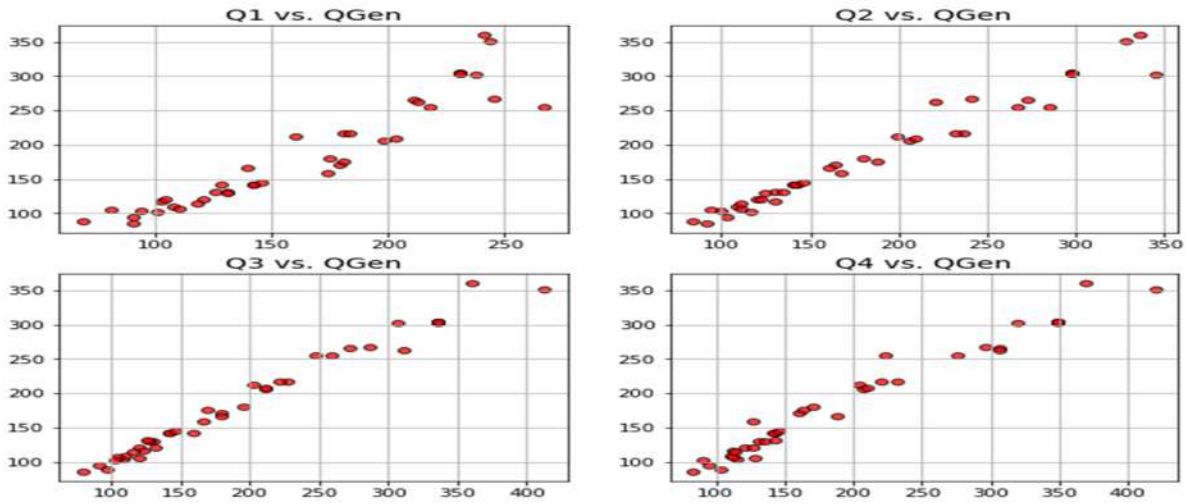
Prediction (Using the ARIMA model)



Method 2: Support Vector Regression Approach

- Linear regression model is developed to forecast the Qgen using SVR model.
- Qgen is a linear combination of Q1,Q2,Q3 and Q4.
- Data is plotted against each Qi with Gen Q as follows:

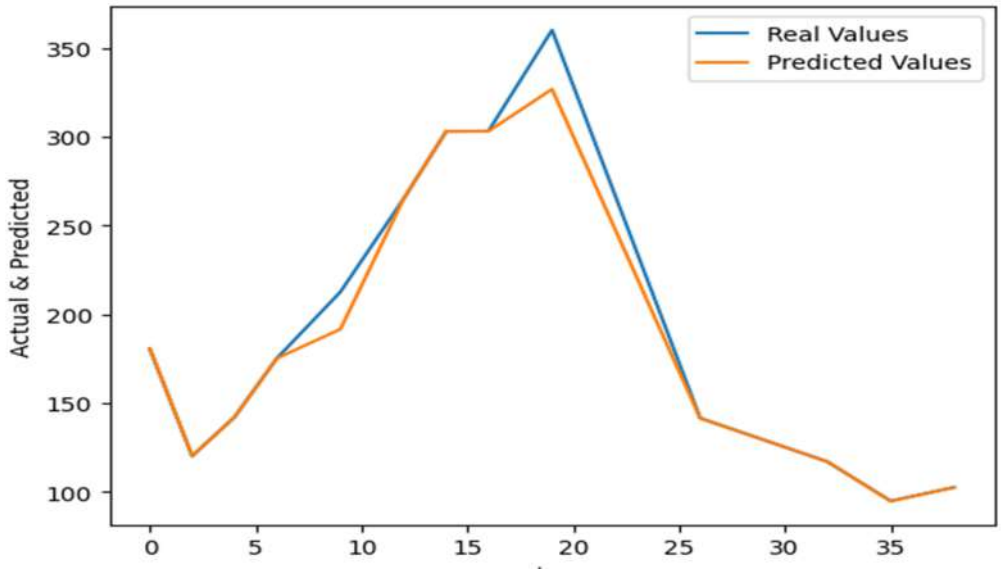
Pictorial representation



- Split the data in to test /train as
- X = df[['Q1','Q2','Q3','Q4']]
- y = df['QGen']
- Support vector regressor with linear kernel is chosen with test score as 0.9832
- Support vector regressor with Gaussian(radial basis function) kernel is also done with a test score of 0.03352.
- RMSE for linear SVR: 10.9163 RMSE for RBF kernelized SVR: 82.9108

Real Values Vs Predicted values (kernel='linear)

Sl no	Real Values	Predicted values
0	180.5	180.6
2	120.1	120.3
4	142.1	142
6	175.2	175.2
9	212.6	191.6
12	265.3	265.3
14	303.1	303.2
16	303.3	303.2
19	360.2	326.9
26	141.5	141.4
32	117	117
35	94.8	94.7
38	102.5	102.4

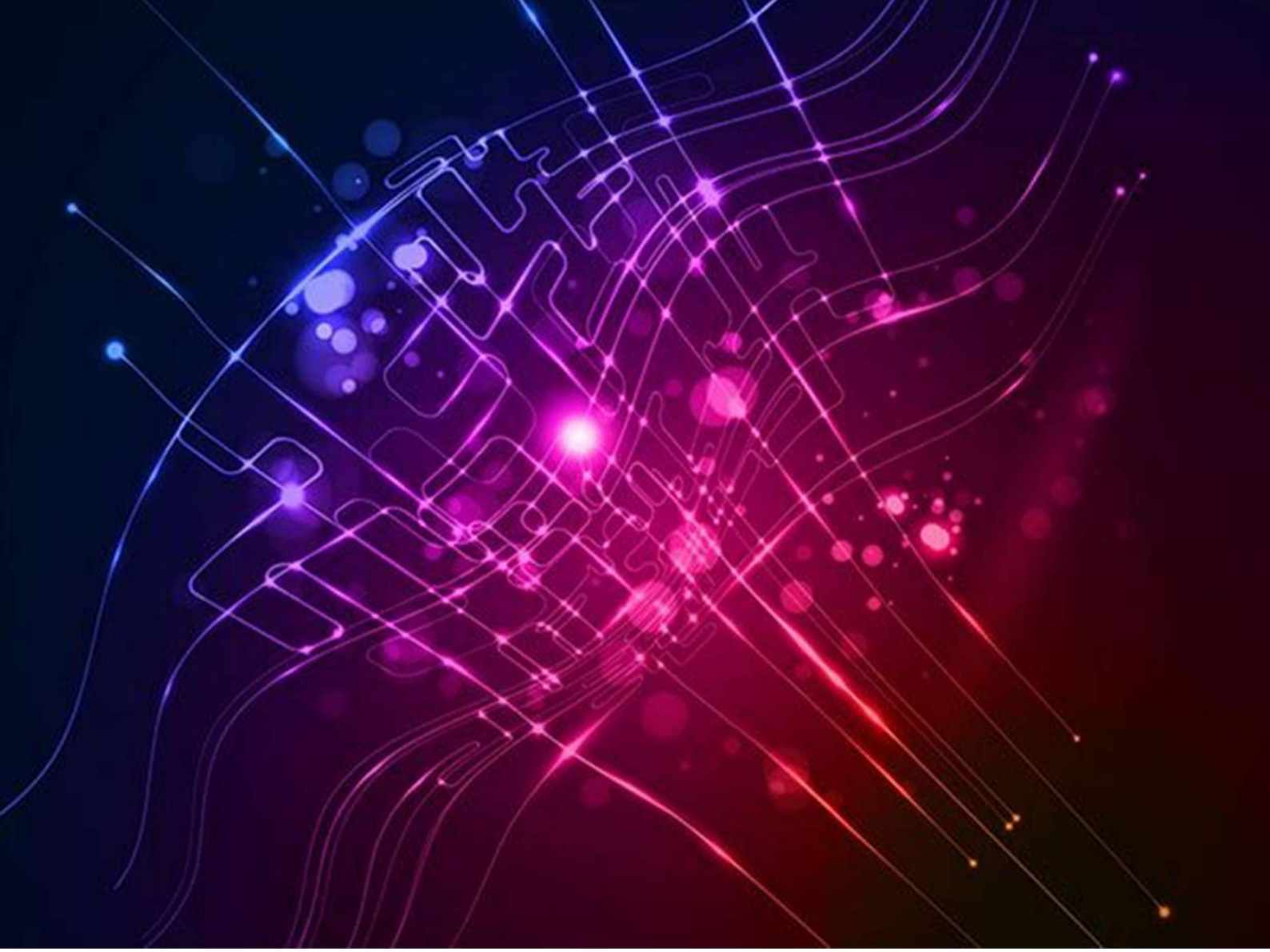


Conclusion

- Trend has got irregular pattern.
- Tested for stationarity using Augmented Dickey-Fuller test.
- Being nonstationary, took first difference and on testing stationarity is found.
- ARIMA(1,1,0) the model suitable to the dataset.
- Support Vector Method is the best method for the prediction of this dataset.
- Analysis of prediction will be helpful for estimation of GDP.

Reference

1. *Hand book from Central Statistical Organisation on IIP.*
2. *Data from Directorate of Economics and Statistics, Government of Kerala. (Annexure-1)*
3. *'Support Vecor Regression' by Dr.Thirthjyoti Sarkar,Fremont,CA94536*
4. *Machine Learning: Algorithms, Real-World Applications and Research Directions by Iqbal H Sarkar in 'SN Computer Science2,' AN:160(2021)*
5. *Review of ML and AutoML Solutions to Forecast Time-Series Data*
6. *Ahmad Alsharef, Karan Aggarwal, Sonia, Manoj Kumar & Ashutosh Mishra in' Archives of Computational Methods in Engineering' 29,2022*
7. *ARIMA and SARIMA forecasting in Neptune.ai https://neptune.ai/blog/arima-sarima-real-world-time-series-forecasting-guide#:~:text=forecasting%20have%20been-,some,-of%20the%20key*



Prediction of Cost of Cultivation of important Crops in Kerala

A Machine Learning Approach

Submitted By
Smt. Suma S.A., Research Assistant

INTRODUCTION

In our country, agriculture is the principal pillar of the economy. The majority of families are dependent on agriculture. The country's GDP is primarily focused on agriculture. More than half of the land is used for agriculture to meet the needs of the population of the region. It is necessary to modernize agricultural practices to meet the demanding requirements. In the past few years, a lot of fluctuation in the prices of the crop has been seen. This has increased the rate of crop damage produced each year. The main aim of this prediction system is to ensure that the farmers get a better idea about the cost of cultivation and how to maximize the production.

OBJECTIVE OF THE PROJECT

Cost of cultivation of major crops are very essential for formulating proper support price policies, creating marketing facilities and assessing loss out of natural calamities, and the share of agriculture in GDP. We need reliable data on crop husbandry right from sowing to harvesting stages. For this, the department conducts an annual survey on Cost of Cultivation of Important Crops in Kerala during every agricultural year (July to June). The major components of the costs involved are seed/seedlings, fertilisers, labour, rent, equipment, irrigation charges etc.The main objective of the survey is to produce reliable estimates on production cost involved in major agricultural crops in the State. Major perennial, annual and seasonal crops are covered in the survey with a breakup of cost incurring during various stages from sowing to harvest.

METHODOLOGY AND METHOD USED

The survey covered all the districts in the state by considering taluk as a stratum. From each Taluk, required numbers of investigator zones were selected using circular systematic sampling method. From selected zones, cultivators and corresponding holdings are selected. The holdings are grouped under three size classes viz; small, medium and large according to the area. . The investigators visited the selected holdings/cultivators and collected the required information in the prescribed format.

Many factors are influential in agriculture, especially in cultivation and production. In this work, the impact of different factors that affect the cost of cultivation is considered and predicts how these factors affect the cost and forecast the cost. For that we use linear regression based prediction models. It is a machine learning algorithm based on supervised learning. It is used for finding out the relationship between variables and forecasting. Linear regression is a powerful tool for understanding and predicting the behavior of a variable. It is a statistical approach for modeling the relationship between a dependent variable and a given set of independent variables.

Machine learning is a popular technique for finding patterns and linear and non-linear relationships between multiple variables (ML).From the static point of view, a model is counted as linear if the model’s parameters are linear. Classification, Regression, and Clustering are a few of the sub categories of ML that can be utilized for analysis and decision making. Machine learning is becoming more and more popular in agriculture, where it is usually necessary to examine complex linkages in order to address difficult agro-engineering difficulties.

DATA SET USED

The actual cost incurred by the cultivators for cultivating Tapioca during 2019-20 is considered for the study. Cost incurred for human labour, seed, fertilizer and other expenses are taken into consideration. The data set have 469 rows and 5 columns .

Farmer_ID	human_labour	fertilizer	Other_Expences	seed	Total_cost
18690	34717.22222	12034.38889	28816.66667	5008.611111	80576.88889
18692	45283.33333	11252.22222	35677.77778	6861.111111	99074.44444
18694	62510	33820	0	7030	103360
18696	50388	0	0	6051.5	56439.5
18698	60515	11197.33333	0	6669	78381.33333
18700	41397.2	13276.25	33345	4322.5	92340.95
18702	54290.6	23450.18	0	4890.6	82631.38
18703	49449.4	10991.5	27170	2667.6	90278.5
18705	41835.625	11747.9375	33962.5	2315.625	89861.6875
18707	44636.42857	8733.214286	8997.857143	2593.5	64961
18709	40382.25455	13540.09091	24924.54545	3817.272727	82664.16364
18711	64590.5	16479.01667	29228.33333	4462.466667	114760.3167
18712	47053.5	14602.64	12597	4722.64	78975.78
18713	35057.53333	11197.33333	24700	5705.7	76660.56667
18714	39245.55556	10072.11111	41166.66667	5543.777778	96028.11111
18956	105222	52827.125	1235	29640	188924.125
18961	77064	38645.62	2470	0	118179.62
18965	58390.8	17722.25	3705	29491.8	109309.85
18966	28322.66667	17590.51667	2470	30875	79258.18333
18967	87602.66667	564930.1667	3293.333333	30628	686454.1667
18968	29640	18722.6	2470	29640	80472.6
18969	62573.33333	28116.83333	4116.666667	30875	125681.8333
18970	62320	18453.75	6175	29212.5	116161.25
19094	28816.66667	38285	6861.111111	548.8888889	74511.66667
19095	64064	26416	26000	520	117000
19096	50684.4	41730.65	2470	555.75	95440.8
19098	105057.3333	16466.66667	8233.333333	55.98666667	129813.32
19130	161538	56810	3705	19760	241813
19131	135603	47918	4940	29640	218101
19257	130169	48988.33333	12350	19760	211267.3333
19258	55328	40343.33333	6586.666667	26346.66667	128604.6667
19259	58539	46930	9880	29640	144989
19260	171171	45448	12350	29640	258609
19261	82251	56711.2	7410	29640	176012.2
19262	136344	50223.33333	28816.66667	29640	245024
19263	166058.1	46930	2964	19760	235712.1
19264	166725	38285	4940	19760	229710
19265	171665	49400	4116.666667	19760	244941.6667
19266	139900.8	106111.2	2964	29640	278616
19267	152349.6	75088	3952	19760	251149.6

19268	156104	49400	32933.33333	29640	268077.3333
19269	106704	37050	9880	29640	183274
19661	108680	59280	4199	10003.5	182162.5
19666	19760	20377.5	4199	10003.5	54340
19669	30875	14820	4199	10003.5	59897.5
19671	27170	14820	4199	10003.5	56192.5
19676	96970.37037	14637.03704	4253.888889	10017.22222	125878.5185
19680	70148	21859.5	4199	10028.2	106234.7
19684	4940	22847.5	4199	9941.75	41928.25
19687	82745	22847.5	4199	10003.5	119795
19689	3458	14820	9151.35	7521.15	34950.5
19693	113620	26552.5	4199	10003.5	154375
19699	15561	14820	9151.35	10028.2	49560.55
19702	21736	13832	4199	10028.2	49795.2
19703	37050	15437.5	4199	10003.5	66690
19704	20583.33333	14820	4199	10003.5	49605.83333
19821	11263.2	16321.76	2964	13832	44380.96
19822	65702	18278	3087.5	6792.5	93860
19823	45052.8	17161.56	2223	11065.6	75502.96
19824	136097	21538.4	2593.5	6792.5	167021.4
19825	119671.5	20501	7039.5	13585	160797
19963	50684.4	20871.5	16919.5	10868	99343.4
19965	70202.88889	24420.06667	25660.55556	10977.77778	131261.2889
19966	54182.81818	29943.13636	24363.18182	6848.636364	115337.7727
19968	46930	22736.35	2161.25	6792.5	78620.1
19969	78216.66667	12452.91667	2058.333333	6792.5	99520.41667
19971	70082.13333	23802.56667	988	13585	108457.7
19973	54193.25294	22814.80882	25353.82353	10926.11765	113288.0029
19974	81658.2	58032.65	1605.5	6792.5	148088.85
19976	62573.33333	63369.22222	1921.111111	10977.77778	138841.4444
20030	42813.33333	33657.86667	27170	1235	104876.2
20031	132741.9167	40343.33333	15272.83333	2964	191322.0833
20032	0	33345	34580	4446	72371
20033	46312.5	25276.33333	6463.166667	3499.166667	81551.16667
20034	0	33345	31286.66667	4693	69324.66667
20035	21736	25984.4	15561	2470	65751.4
20036	64220	20741.66667	14566.66667	3166.666667	102695
20037	21736	23415.6	14943.5	4075.5	64170.6
20039	39520	13356.525	17290	4168.125	74334.65
20040	110676.5833	20274.58333	40508	3952	175411.1667
20042	63490.22727	17645.5303	17514.54545	3181.060606	101831.3636
20043	106366.4333	18730.83333	20171.66667	4075.5	149344.4333
20044	138715.2	38976.6	20583.33333	2840.5	201115.6333
20396	50943.75	14820	3952	10497.5	80213.25
20397	131733.3333	8562.666667	3211	10291.66667	153798.6667
20398	87067.5	7113.6	27170	9262.5	130613.6

20399	83362.5	5804.5	2470	9262.5	100899.5
20400	64837.5	12350	3396.25	10806.25	91390
20742	46312.5	16252.6	1482	2223	66270.1
20744	44305.625	19797.05	13214.5	2223	79540.175
20754	27092.8125	19142.5	1482	2223	49940.3125
20771	41852.77778	19691.38889	1482	2223	65249.16667
20774	45108.375	15363.4	13214.5	1852.5	75538.775
20778	65240.21739	17808.7	7844.934783	2223	93116.85217
20781	42144.375	15561	864.5	1852.5	60422.375
20826	61132.5	15437.5	4940	592.8	82102.8
20829	48341.42857	12173.57143	4410.714286	564.5714286	65490.28571
20830	50388	19809.4	27170	559.8666667	97927.26667
20834	45882.12121	22649.15152	65118.18182	591.3030303	134240.7576
20837	95977.14286	7410	0	555.75	103942.8929
20839	31813.6	27861.6	2470	1037.4	63182.6
20844	36414.85714	19834.1	3175.714286	1287.928571	60712.6
20845	75852.90323	44101.45161	3983.870968	2788.709677	126726.9355
20846	62604.11215	46456.7757	2885.514019	1442.757009	113389.1589
20847	49297.08333	20531.875	2572.916667	568.1	72969.975
20877	87771.45	11317.54	17537	2741.7	119367.69
20878	121735.7143	11673.69048	14702.38095	2858.142857	150969.9286
20879	115892.4	12251.2	18772	2815.8	149731.4
20880	88623.6	12238.85	7904	2741.7	111508.15
20881	119301	11073.83333	14820	2717	147911.8333
20882	122141.5	12257.375	14820	2964	152182.875
20883	129400.5556	13630.74074	0	2881.666667	145912.963
20884	129974.3939	11564.09091	14221.21212	2919.090909	158678.7879
20885	88055.5	11621.35	14326	2758.99	116761.84
20886	123088.3333	10209.33333	32933.33333	2791.1	169022.1
20887	80599.1875	11701.625	27787.5	2732.4375	122820.75
20888	80050.45455	10430.13636	20209.09091	2761.909091	113451.5909
20889	85783.1	12379.64	20254	2741.7	121158.44
20890	130795.1163	11344.76744	14360.46512	2843.372093	159343.7209
20891	121573.4	12201.8	18772	2781.22	155328.42
20922	91595.83333	37616.04167	13379.16667	0	142591.0417
20923	44460	34394.75	18525	0	97379.75
20924	102505	36020.83333	23465	0	161990.8333
20925	101270	33670.21667	27581.66667	0	162521.8833
20926	105386.6667	35362.16667	24974.44444	0	165723.2778
20927	49400	35609.16667	23670.83333	0	108680
20928	104925.6	33846.41	27417	0	166189.01
20929	69160	35537.125	15437.5	0	120134.625
20930	69160	35490.8125	2778.75	0	107429.5625
20932	96947.5	34541.40625	2624.375	0	134113.2813
20933	99623.33333	34857.875	3087.5	0	137568.7083
20934	99381.17647	35451.76471	2760.588235	0	137593.5294

20935	91390	34166.275	15437.5	0	140993.775
20936	112693.75	34721.51042	3087.5	0	150502.7604
21201	0	10571.6	2667.6	0	13239.2
21203	19760	10374	2717	12350	45201
21204	127205	8274.5	3705	11115	150299.5
21205	127205	13585	4199	12350	157339
21207	55081	7508.8	3754.4	9880	76224.2
21210	32418.75	5829.2	27911	9880	76038.95
21213	67153.125	8892	3087.5	10806.25	89938.875
21463	40343.33333	118683.5	11320.83333	2470	172817.6667
21464	150747.1875	25626.25	0	2470	178843.4375
21466	86958.52941	10868	1452.941176	2411.882353	101691.3529
21595	51870	10880.35	22230	2593.5	87573.85
21596	22791.36364	9414.068182	785.9090909	2357.727273	35349.06818
21598	21612.5	9639.175	988	2470	34709.675
21600	51870	9945.866667	823.3333333	2634.666667	65273.86667
21603	51870	11197.33333	823.3333333	2470	66360.66667
21604	51870	11197.33333	2881.666667	2387.666667	68336.66667
21605	40343.33333	11650.16667	9262.5	2470	63726
21606	92393.4375	11933.1875	17135.625	2593.5	124055.75
21607	46106.66667	9934.888889	4116.666667	2524.888889	62683.11111
21608	24378.9	9361.3	52858	2568.8	89167
21609	46106.66667	9934.888889	7410	2524.888889	65976.44444
21796	255645	59280	3087.5	0	318012.5
21797	228228	44460	4940	0	277628
21798	213408	50635	4940	11856	280839
21799	251940	50282.14286	8821.428571	10938.57143	321982.1429
21800	205504	105798.3333	3293.333333	11856	326451.6667
21801	281580	29640	6175	11362	328757
21802	167960	83980	26758.33333	11362	290060.3333
21803	196701.8182	70507.27273	2245.454545	10778.18182	280232.7273
21804	222300	67925	4940	11362	306527
21805	222892.8	70148	3952	11856	308848.8
21806	247000	56810	4116.666667	11856	319782.6667
21807	243048	50635	32110	11609	337402
21820	41249	41496	4940	4100.2	91785.2
21822	50388	43048.57143	4234.285714	4128.428571	101799.2857
21823	51652.05882	42498.52941	3632.352941	4140.882353	101923.8235
21825	0	35074	4940	3359.2	43373.2
21826	0	40137.5	6175	3334.5	49647
21828	82745	42731	9880	3359.2	138715.2
21831	0	41743	9262.5	3211	54216.5
22088	19595.33333	10209.33333	19760	22641.66667	72206.33333
22089	27875.71429	11820.71429	42342.85714	22935.71429	104975
22090	57760	18810	0	22800	99370
22091	11945.81818	7320.181818	39295.45455	22454.54545	81016

22094	27664	4347.2	6175	17784	55970.2
22096	29640	10868	0	21406.66667	61914.66667
22097	29640	7410	5292.857143	15525.71429	57868.57143
22100	27664	23218	0	22230	73112
22101	31051.42857	10444.57143	4940	19407.14286	65843.14286
22102	25688	10670.4	4940	22230	63528.4
22397	27787.5	7718.75	9262.5	35506.25	80275
22401	17100	9500	8550	35625	70775
22411	32521.66667	10291.66667	6586.666667	46312.5	95712.5
22412	29640	6175	8645	38593.75	83053.75
22417	24700	10291.66667	9056.666667	43225	87273.33333
22420	18525	15437.5	8645	38593.75	81201.25
22704	79534	9880	3293.333333	5763.333333	98470.66667
22708	98800	69160	2470	13832	184262
22710	44954	16302	12350	5187	78793
22712	82004	11856	988	13832	108680
22714	60515	29516.5	7410	8645	106086.5
22716	56480.66667	13585	11526.66667	5763.333333	87355.66667
22717	77805	7619.95	3087.5	10806.25	99318.7
22752	77805	6010.333333	0	1482	85297.33333
22754	113414.1667	14820	20583.33333	2675.833333	151493.3333
22755	104710.3571	9527.142857	17642.85714	2117.142857	133997.5
22759	70395	10538.66667	0	2634.666667	83568.33333
22761	109297.5	5038.8	0	1976	116312.3
22762	67184	15017.6	0	1284.4	83486
22763	42907.42857	11115	35285.71429	1270.285714	90578.42857
22764	79781	8768.5	18525	2161.25	109235.75
22765	88384.83333	14161.33333	12350	1893.666667	116789.8333
22766	57489.25	8398	18525	1482	85894.25
22924	66690	25688	17290	11115	120783
22926	160550	45695	4940	17290	228475
22927	146965	27417	19760	12967.5	207109.5
22929	87067.5	23526.75	4013.75	12967.5	127575.5
22931	81734.54545	24879.63636	15942.72727	19198.63636	141755.5455
22932	138814	42879.2	11856	19760	213309.2
22933	100198.6375	25163.125	15283.125	12350	152994.8875
22934	0	103122.5	0	0	103122.5
22935	107856.6667	21200.83333	13585	14820	157462.5
22937	89414	25885.6	14326	9089.6	138715.2
22938	0	18525	14820	21612.5	54957.5
22951	0	22847.5	15746.25	12350	50943.75
22952	213655	26552.5	16672.5	22230	279110
22953	101270	25317.5	14820	17290	158697.5
23427	7410	9213.1	6175	4446	27244.1
23428	19760	11559.6	617.5	4199	36136.1
23429	5903.3	15264.6	1432.6	3853.2	26453.7

23430	22649.9	11053.25	12547.6	3912.48	50163.23
23431	20315.75	11003.85	3458	4446	39223.6
23432	8398	22081.8	3952	3952	38383.8
23433	4940	10308.13333	1646.666667	4083.733333	20978.53333
23434	14820	16055	4446	4446	39767
23435	17290	14523.6	3952	3754.4	39520
23437	0	15906.8	4446	4693	25045.8
23438	8645	22600.5	3211	3952	38408.5
23439	0	22057.1	2717	4149.6	28923.7
23441	6817.2	10275.2	2568.8	3754.4	23415.6
23983	63016.863	49400	0	8645	121061.863
23985	42011.242	37050	0	7410	86471.242
23988	95319.62471	29058.82353	0	7410	131788.4482
23990	52514.0525	0	0	7410	59924.0525
24177	32933.33333	54718.73333	0	9880	97532.06667
24179	24700	60515	0	9880	95095
24183	18525	34147.75	617.5	9880	63170.25
24185	54340	61354.8	0	9880	125574.8
24187	28816.66667	38202.66667	1235	9880	78134.33333
24189	25688	40014	1976	9880	77558
24192	49400	62342.8	0	9880	121622.8
24194	61750	70395	0	9880	142025
24196	37050	45373.9	0	9880	92303.9
24198	49400	57616.86667	0	9880	116896.8667
24201	30875	38068.875	1235	9880	80058.875
24203	37050	42138.2	2470	4940	86598.2
24205	27444.44444	43515.91111	1097.777778	9880	81938.13333
24207	36388.39286	44007.90179	1102.678571	9262.5	90761.47321
24209	33592	41377.44	1235	9262.5	85466.94
24706	69160	44460	741	9880	124241
24707	69160	44460	0	19760	133380
24708	74100	41990	1976	14820	132886
24709	102999	49400	889.2	18525	171813.2
24710	108927	46831.2	494	9880	166132.2
24713	106704	49400	3309.8	11856	171269.8
24715	103740	39520	1432.6	24700	169392.6
24716	86450	31122	543.4	14820	132935.4
24856	15808	30628	9880	2470	58786
24857	52693.33333	34888.75	24700	8233.333333	120515.4167
24858	67184	31369	17290	7904	123747
24859	26346.66667	51046.66667	12350	4116.666667	93860
24860	98800	40137.5	18525	7718.75	165181.25
24861	93860	25729.16667	20583.33333	5763.333333	145935.8333
24862	71136	56513.6	29640	7410	164699.6
24863	22880	52260	20800	4550	100490
24864	13173.33333	343988.6667	20583.33333	4528.333333	382273.6667

24865	29640	46189	18525	4322.5	98676.5
24866	34580	35969.375	18525	9262.5	98336.875
24867	72453.33333	25935	20583.33333	8233.333333	127205
24868	92213.33333	32933.33333	16466.66667	8233.333333	149846.6667
24869	19760	53599	24700	3705	101764
24870	29640	45695	24700	5557.5	105592.5
24902	44460	21964	1672	3420	71516
24903	46106.66667	0	50058.66667	3705	99870.33333
24904	43754.28571	21877.14286	1834.857143	4940	72406.28571
24910	17784	0	2766.4	5928	26478.4
24912	21171.42857	299.9285714	0	35.28571429	21506.64286
24916	30875	30463.33333	0	10291.66667	71630
24918	15808	23316.8	0	4940	44064.8
24920	7904	1729	1778.4	19.76	11431.16
24921	15437.5	14279.6875	694.6875	4631.25	35043.125
24922	49811.66667	16672.5	1008.583333	3293.333333	70786.08333
24923	18833.75	78249.6	1420.25	19760	118263.6
24924	26111.42857	0	0	0	26111.42857
25013	192660	0	34580	3705	230945
25014	40508	5434	13832	3705	63479
25015	62244	3260.4	6669	3705	75878.4
25016	115596	4693	13832	2470	136591
25018	53352	16450.2	15808	5928	91538.2
25078	85956	7706.4	8151	3087.5	104900.9
25080	44460	13585	22230	2470	82745
25083	133380	7410	16055	2470	159315
25084	87932	10291.66667	10703.33333	2470	111397
25085	53352	4841.2	15808	1811.333333	75812.53333
25086	65866.66667	14408.33333	6586.666667	2744.444444	89606.11111
25087	22230	7718.75	21303.75	5557.5	56810
25088	119795	0	14408.33333	5763.333333	139966.6667
25090	44460	17784	6175	4940	73359
25337	24206	10769.2	3705	2173.6	40853.8
25338	13832	15923.26667	3705	2107.733333	35568
25339	31122	172692.52	3705	2074.8	209594.32
25341	34580	14326	3705	2124.2	54735.2
25348	34580	10868	3705	2173.6	51326.6
25350	11970	16739	28500	2128	59337
25353	31122	9287.2	3705	2074.8	46189
25355	34580	16845.4	3705	2124.2	57254.6
25358	21612.5	20346.625	4322.5	2161.25	48442.875
25364	17290	22946.3	4322.5	2161.25	46720.05
25365	34580	18747.3	28405	2074.8	83807.1
25452	81345.33333	14881.75	2470	1729	100426.0833
25453	58381.81818	14881.75	2470	1729	77462.56818
25454	53516.66667	14881.75	2470	1729	72597.41667

25455	61651.2	14881.75	2470	1729	80731.95
25456	121459.5652	14881.75	2470	1729	140540.3152
25457	58381.81818	14881.75	2470	1729	77462.56818
25458	57923.92157	14881.75	12156.27451	1729	86690.94608
25459	77064	14881.75	2470	1729	96144.75
25460	58868.33333	14881.75	2470	1729	77949.08333
25461	57798	14881.75	2470	1729	76878.75
25462	72247.5	14881.75	2470	1729	91328.25
25463	73050.25	14881.75	2470	1729	92131
25464	86697	14881.75	14820	1729	118127.75
25465	71926.4	14881.75	2470	1729	91007.15
25466	62079.33333	14881.75	2470	1729	81160.08333
25564	128440	35815	12350	7410	184015
25568	104975	31863	12350	9880	159068
25569	79040	16302	7410	5557.5	108309.5
25571	115266.6667	33756.66667	10291.66667	4631.25	163946.25
25572	100776	28652	4940	22230	156598
25573	125146.6667	34580	8233.333333	5187	173147
25574	209950	30875	6861.111111	4116.666667	251802.7778
25575	126587.5	19945.25	33962.5	23156.25	203651.5
25576	59280	0	12350	7286.5	78916.5
25577	65866.66667	31904.16667	10291.66667	9365.416667	117427.9167
25579	39520	27170	8645	12350	87685
25580	44460	27170	8645	8398	88673
25586	37050	41166.66667	0	10291.66667	88508.33333
25869	57633.33333	13412.1	20583.33333	1976	93604.76667
25870	69160	14375.4	0	1976	85511.4
25871	60515	14227.2	12350	1976	89068.2
25872	57633.33333	12720.5	0	1646.666667	72000.5
25873	69160	13930.8	0	1976	85066.8
25874	60515	13486.2	0	1976	75977.2
25875	57633.33333	13634.4	0	1976	73243.73333
25905	60515	12844	14408.33333	1976	89743.33333
25906	57633.33333	13930.8	12350	1976	85890.13333
25907	58209.66667	14227.2	13996.66667	1976	88409.53333
25908	56192.5	12893.4	2470	1976	73531.9
25909	69160	14375.4	0	1976	85511.4
25910	55328	13930.8	12350	1976	83584.8
25911	60515	13782.6	0	1976	76273.6
25912	59280	14057.82857	0	1976	75313.82857
26343	74100	25416.3	18590.86667	3077.62	121184.7867
26345	74100	25218.7	18599.1	3008.46	120926.26
26347	67925	26610.13333	3770.866667	3008.46	101314.46
26350	74100	25218.7	18599.1	3008.46	120926.26
26352	74100	25218.7	3779.1	3008.46	106106.26
26354	83362.5	25218.7	18599.1	3008.46	130188.76

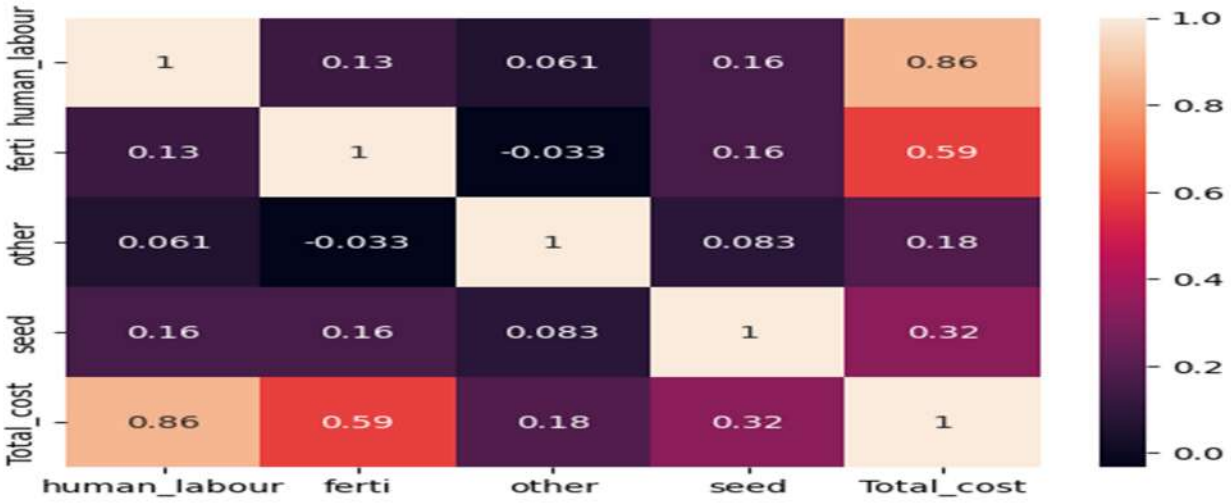
26356	74100	25218.7	3779.1	3008.46	106106.26
26358	71250	25897	18582	3032.4	118761.4
26360	74100	25539.8	3770.866667	3008.46	106419.1267
26361	69468.75	25700.35	3766.75	3008.46	101944.31
26364	69468.75	25700.35	3766.75	3008.46	101944.31
26367	74100	25539.8	3770.866667	3008.46	106419.1267
26370	64837.5	25539.8	18557.93333	2991.17	111926.4033
26372	83362.5	25712.7	18599.1	3008.46	130682.76
26373	72041.66667	27206.59259	3769.037037	2939.3	105956.5963
26732	59280	10497.5	3087.5	5094.375	77959.375
26733	79040	10127	4116.666667	4734.166667	98017.83333
26734	70571.42857	10850.35714	4410.714286	4719.464286	90551.96429
26735	72453.33333	10127	4116.666667	4693	91390
26736	98800	10127	4116.666667	4693	117736.6667
26737	108680	11609	3705	4723.875	128717.875
26738	119600	11440	3900	4712.5	139652.5
26739	79040	11115	4116.666667	4693	98964.66667
26740	74100	10868	3705	4693	93366
26741	81765.51724	11498.27586	4258.62069	4727.068966	102249.4828
26742	26346.66667	11115	4116.666667	4693	46271.33333
26743	63232	11312.6	4940	4693	84177.6
26744	59280	10991.5	3705	4693	78669.5
26745	69160	11732.5	3705	4693	89290.5
26746	46106.66667	111479.3333	4116.666667	4693	166395.6667
26782	46683	26972.4	0	0	73655.4
26784	69160	21242	0	0	90402
26785	69160	21933.6	0	0	91093.6
26786	77805	21242	0	0	99047
26788	74923.33333	21900.66667	0	0	96824
26789	69160	23712	0	0	92872
26791	64220	20042.28571	0	0	84262.28571
26792	57633.33333	22065.33333	0	0	79698.66667
26794	60515	23712	0	0	84227
26795	57633.33333	22065.33333	0	0	79698.66667
26797	69160	18772	0	0	87932
26798	60515	23712	0	0	84227
26800	51870	20254	0	0	72124
26801	69160	21736	10585.71429	0	101481.7143
26803	46106.66667	26099.66667	14408.33333	0	86614.66667
27143	44460	57304	0	2964	104728
27145	49400	44089.5	0	2964	96453.5
27147	83362.5	18833.75	0	2593.5	104789.75
27150	93860	32933.33333	4116.666667	3293.333333	134203.3333
27152	83980	18722.6	4940	3705	111347.6
27155	59280	41372.5	0	4075.5	104728
27157	113620	79040	0	5681	198341
27160	123500	69160	0	5928	198588
27162	64220	55575	4940	3556.8	128291.8

27163	37544	21143.2	4940	2173.6	65800.8
27165	60206.25	27633.125	3087.5	3581.5	94508.375
27168	128440	44624.66667	0	4281.333333	177346
27186	81510	23794.33333	4116.666667	2881.666667	112302.6667
27189	91390	33345	0	3705	128440
27191	106210	33098	0	3458	142766
27267	30708.4186	15004.96279	7310.625581	14820	67844.00698
27272	30381	15042.3	4841.2	14820	65084.5
27275	31122	14754.13333	4841.2	14820	65537.33333
27276	31122	15314	4841.2	14820	66097.2
27278	31122	14754.13333	4841.2	14820	65537.33333
27279	31122	14978.08	4841.2	14820	65761.28
27280	26676	15153.45	4841.2	14820	61490.65
27281	31122	10682.75	4841.2	14820	61465.95
27282	31122	14754.13333	4841.2	14820	65537.33333
27283	31122	15314	4841.2	14820	66097.2
27284	31122	14754.13333	4841.2	14820	65537.33333
27285	31122	14754.13333	4841.2	14820	65537.33333
27288	31122	15832.7	4841.2	14820	66615.9
27289	30486.85714	15003.48571	4841.2	14820	65151.54286
27291	31122	15001.13333	4824.733333	14820	65767.86667
27397	211333.2	20105.8	17092.4	22526.4	271057.8
27398	174036.9841	15913.85714	33019.5873	19007.2381	241977.6667
27399	185579.3333	15931.5	42628.08333	18969.6	263108.5167
27400	205936.25	20089.33333	24339.79167	22559.33333	272924.7083
27401	159879.5714	26517.21429	26810.08571	19004.88571	232211.7571
27402	139752.6	15931.5	26799.5	18969.6	201453.2
27403	232109.4286	17607.57143	17092.4	19336.57143	286145.9714
27404	190313.5	20229.3	17092.4	22559.33333	250194.5333
27405	168791.25	15917.25	26818.5	19000	230527
27406	190719.2857	20095.21429	20642.14286	22524.04762	253980.6905
27407	176111	20229.3	26815.96667	22526.4	245682.6667
27408	175542.9	37173.5	26587.08	18969.6	258273.08
27409	193154	123944.6	18772	22585.68	358456.28
27410	226450.9722	20103.05556	17091.02778	22312.33333	285957.3889
27657	49400	40755	43225	29640	163020
27659	29640	18525	30875	24700	103740
27660	26346.66667	29640	28816.66667	16466.66667	101270
27661	187720	41990	29640	14820	274170
27663	108680	33345	12350	14820	169195
27666	59280	102505	49400	15437.5	226622.5
27668	94848	0	9880	19760	124488
27671	65866.66667	41825.33333	41166.66667	12350	161208.6667
27743	235885	44608.2	2223	8299.2	291015.4
27744	222300	44213	2223	8299.2	277035.2
27745	216125	42422.25	2161.25	8336.25	269044.75
27746	157462.5	43406.13333	2140.666667	8274.5	211283.8
27747	214066.6667	44130.66667	2470	8274.5	268941.8333

27748	247000	44410.6	2223	8299.2	301932.8
27749	182780	43136.08	2173.6	8269.56	236359.24
27750	196828.125	42175.25	2161.25	8289.9375	249454.5625
27751	222300	44410.6	2099.5	8521.5	277331.6
27752	81664.375	42903.9	2161.25	8280.675	135010.2
27753	223250	43206	2185	8265	276906
27754	198835	44608.2	2223	8299.2	253965.4
27755	175987.5	46139.6	2161.25	8336.25	232624.6
27756	395200	0	2223	8892	406315
27757	222300	42418.13333	2140.666667	8299.2	275158

DATA PREPARATION

- Missing data and null values are treated
- Study the correlation between the dependent variable (total cost) and independent variables(human labour, seed, fertilizer and other expenses).
- For that we use heatmap.
- From the heatmap it is clear that the dependent variable and independent variable are correlated.



TOOLS AND LIBRARIES USED

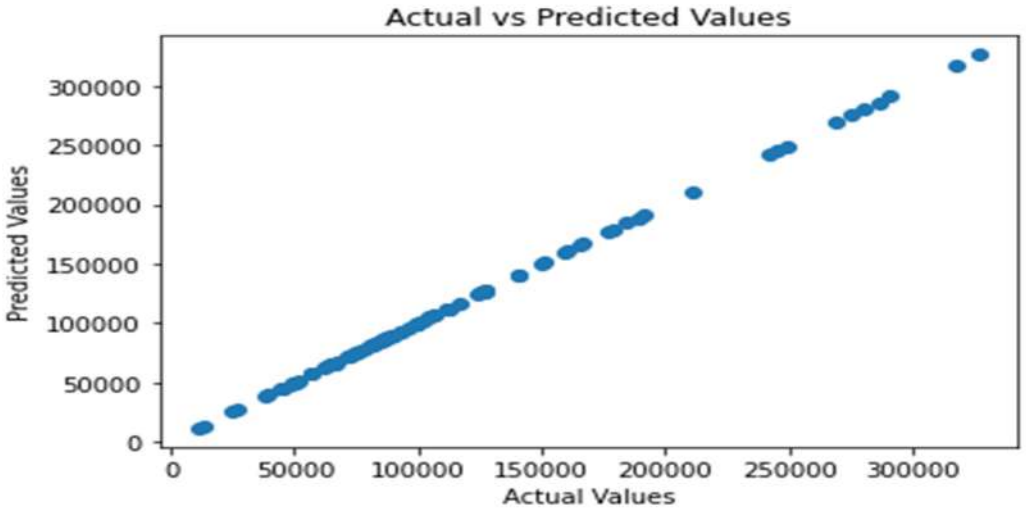
- numpy
- matplotlib.pyplot
- pandas
- seaborn
- sklearn

RESULT WITH INFERENCE

- In this work, the impact of different factors that affect the cost of cultivation is considered and predicts how these factors affect the cost and forecast the cost.
- For that we use linear regression based prediction models. . Linear regression is a powerful tool for understanding and predicting the behavior of a variable. It is a

statistical approach for modeling the relationship between a dependent variable and a given set of independent variables.

- To check the accuracy we draw scatter diagram with actual value and predicted value.
- From the diagram it is clear that almost all the values lying in a straight line.
- It means that actual value and predicted values are almost equal.
- The model that we create is a perfect one.

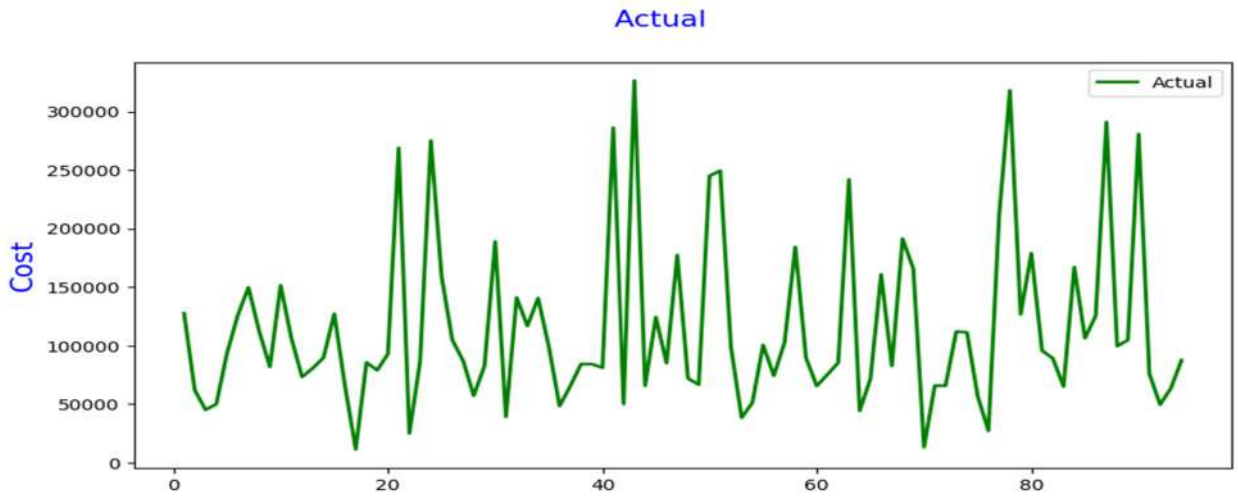


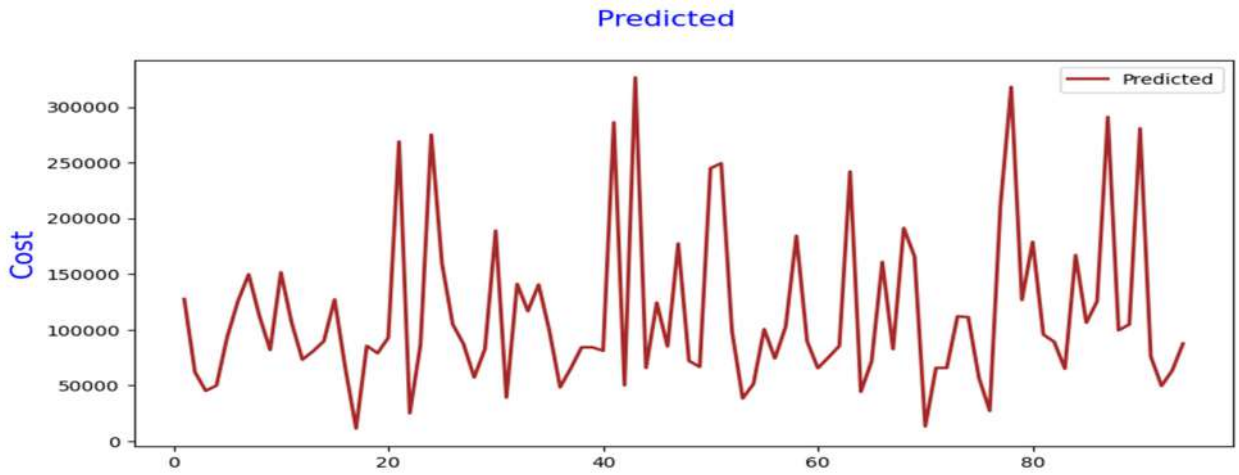
- Calculating the regression coefficient and fit the data in the model
- We got the actual value.
- predictedy = regr.predict([[34717.22222, 12034.38889, 28816.666670, 5008.611111]])
  The predicted values corresponding to the above costs is ([80576.88889239])
- For this we take 80% of the data randomly for training the machine and remaining data for testing.
- Using training data train the machine and construct a linear regression model .
- Fit the test value in the model and we got the future predicted value.
- In order to find the accuracy of the model we calculate the error value- mean squared error.
- We got mse=1.9770776563037933e-05
- It is evident that the model that we construct is correct and using it we can predict the future value

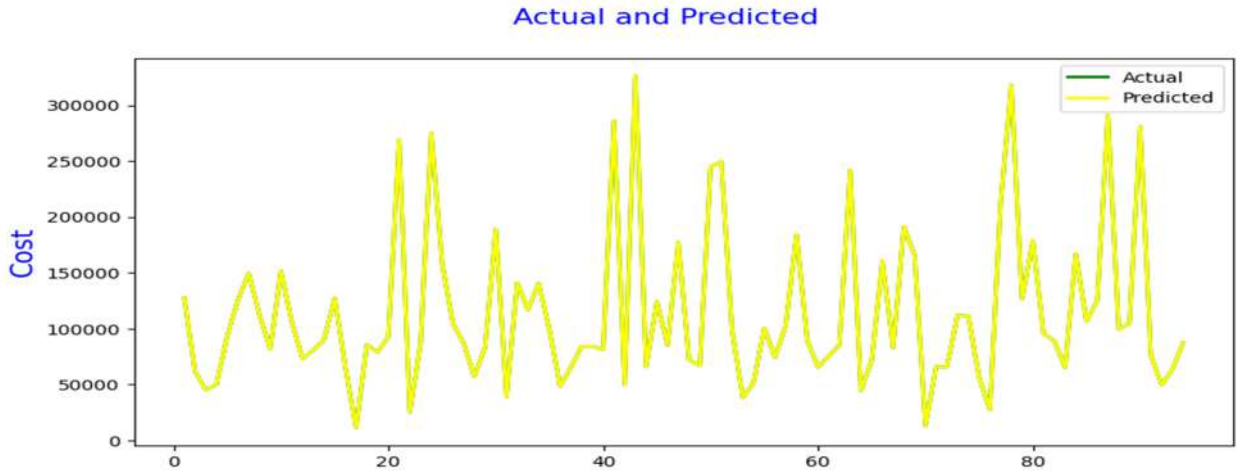
Actual Value and Predicted Value

Sl.No	Y_test	Y_pred
211	127575.50000	127575.50000053
180	61914.66667	61914.66666849
136	45201.0000	45200.99999913
91	49940.31250	49940.3124998
380	93366.00000	93366.00000076
158	280839.00000	280839.00000316
347	75977.20000	75977.20000032
56	49605.83333	49605.83332943
290	63479.00000	63479.00000111
145	87573.85000	87573.85000219

The graphical representations are given below.





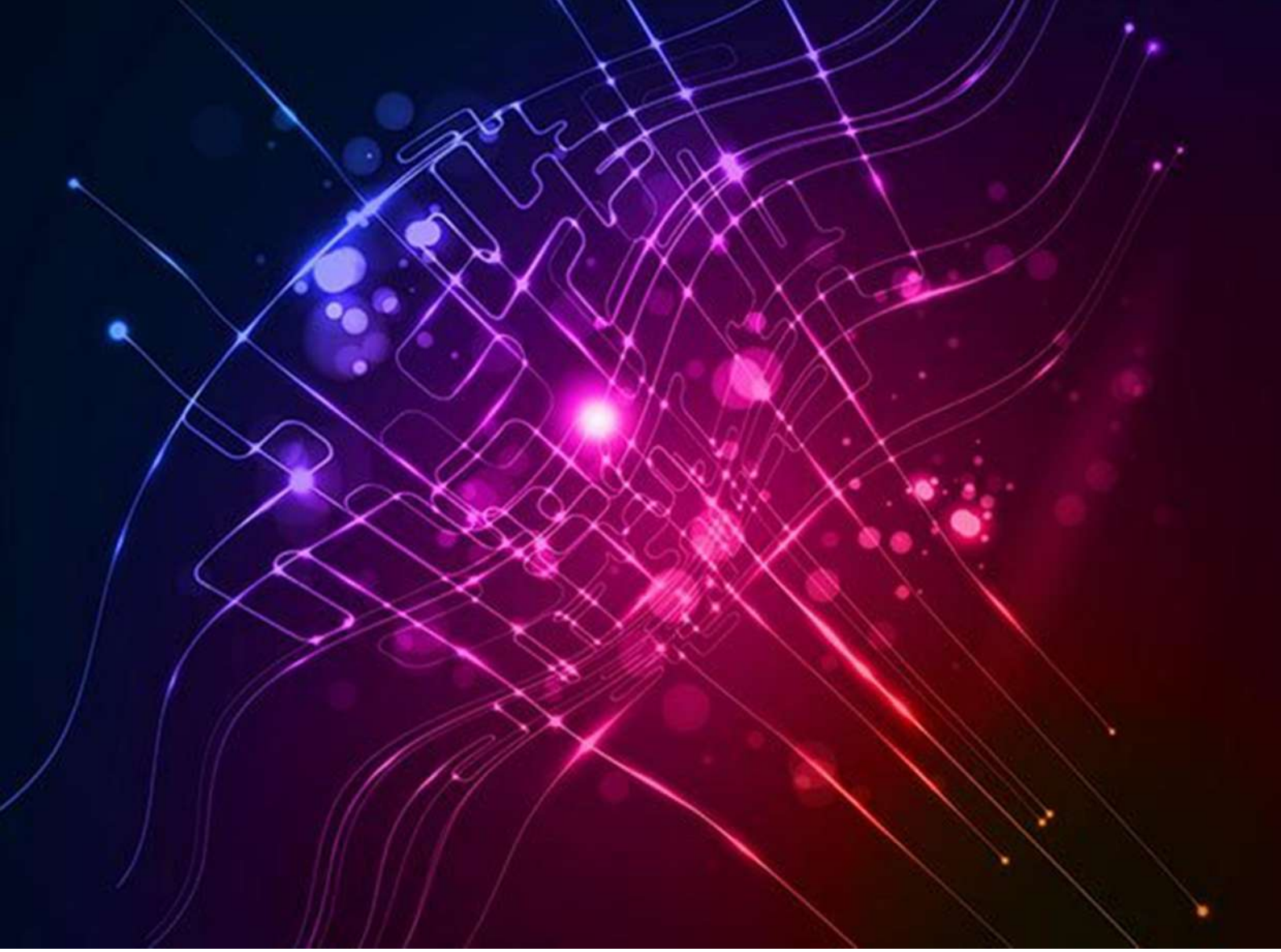


CONCLUSION

This paper was aimed to discuss the application of artificial intelligence in agriculture, even in the common case of small available datasets. The approach was applied and verified in the specific case of tapioca. Also aims to use the model to predict other important crops.

REFERENCE

1. *Report on Cost of Cultivation 2011-12 to 2020-21 published by Directorate of Economics and Statistics*
2. *Multilinear regression model based studies*



Utilising AI for accurate forecasting of Kerala's GSVA and NSVA

A methodological Analysis and Evaluation of benefits

Submitted By
Smt. Praseeda Gopan, Research Assistant

Introduction

Forecasting the Gross State Value Added (GSVA) and Net State Value Added (NSVA) of Kerala is crucial for economic planning and policymaking. Accurate projections of these figures for the upcoming financial year and the subsequent year provide insights into the state's economic growth and development prospects. Traditionally, such projections are made based on the analysis of historical data, economic indicators, and other relevant factors. However, with the advent of Artificial Intelligence (AI) and its advanced analytical capabilities, there is a growing interest in exploring its potential for more accurate and efficient projections. In this report, I examine how AI can be used to project the GSVA/NSVA of Kerala for the next five years and analyze the benefits of using AI-based forecasting techniques over traditional methods.

Objectives

The primary objective of this report is to utilize Artificial Intelligence (AI) techniques for the projection of the Gross State Domestic Product (GSDP) and Net State Domestic Product (NSDP) of Kerala for the next five financial years. Specifically, I aim to achieve the following:

1. Develop an AI-based forecasting model to project the GSDP/NSDP of Kerala for the upcoming financial years.
2. Assess the impact of unforeseen conditions on the various sectors of the economy and evaluate the effect of these conditions on the projected GSDP/NSDP figures.
3. Analyze the benefits of using AI-based forecasting techniques over traditional methods for accurate and efficient projections of economic indicators.
4. Provide recommendations for policymakers on how to utilize the AI-based forecasting model to make informed decisions for the economic development of Kerala.

Methodology and method used

The methodology employed in this report involved the following steps:

1. **Data Collection:** Historical economic data of Kerala spanning from 1970 to 2019 was collected from the Department.
2. **Data Standardization:** To ensure consistency in the data, all economic indicators were brought to a common base year of 2011-12.
3. **Correlation Analysis:** To identify the key drivers of economic growth, heatmaps were used to study the correlation between dependent and independent variables for both the primary sector and total GSVA.
4. **Model Building:** Linear regression models were built to calculate the regression coefficient and fit the data for both the primary sector and total GSVA. The models were evaluated using scatter diagrams to compare the predicted values with the actual values.
5. **Predictions:** The models were used to predict the GSVA/NSVA figures for the next five financial years, both sector-wise and for the total GSVA. The mean squared error was calculated for both models using Linear Regression to evaluate the accuracy of the predictions.

Datasets used

For the purpose of this report, I have used historical economic data of Kerala spanning from 1970 to 2019. It is important to note that the base year for the economic data has been shifted several times during this period. Therefore, I first standardized the data by bringing all the economic indicators to a common base year of 2011-12. The data was sourced from the Department of Economics and Statistics.

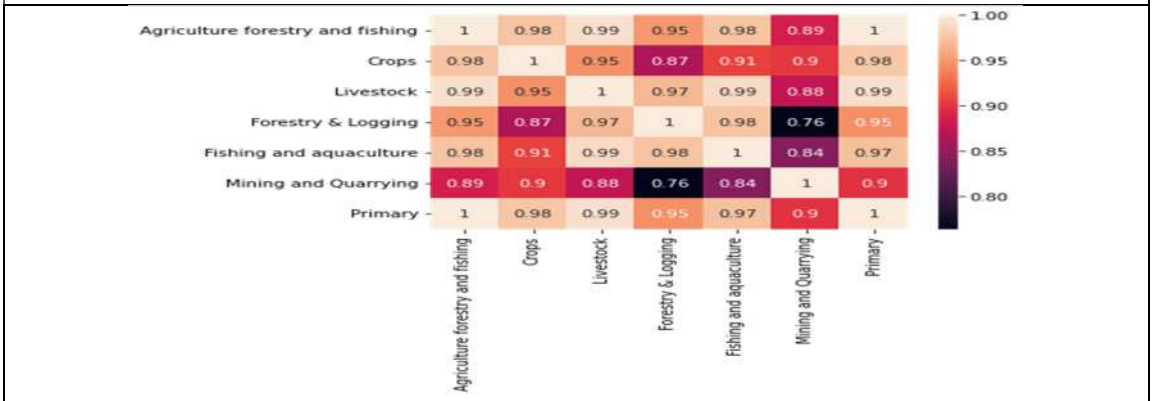
Tools and libraries used

To achieve the objective of projecting the Gross State Value Added (GSVA) and Net State Value Added (NSVA) of Kerala using Artificial Intelligence (AI), I have employed a variety of techniques. In particular, I utilized machine learning algorithms, such as regression analysis and time-series forecasting, to build and test my models. I also used data visualization techniques, such as heatmaps, to study the correlation between dependent and independent variables. By examining the correlation between various economic indicators, I was able to identify the key drivers of economic growth in Kerala.

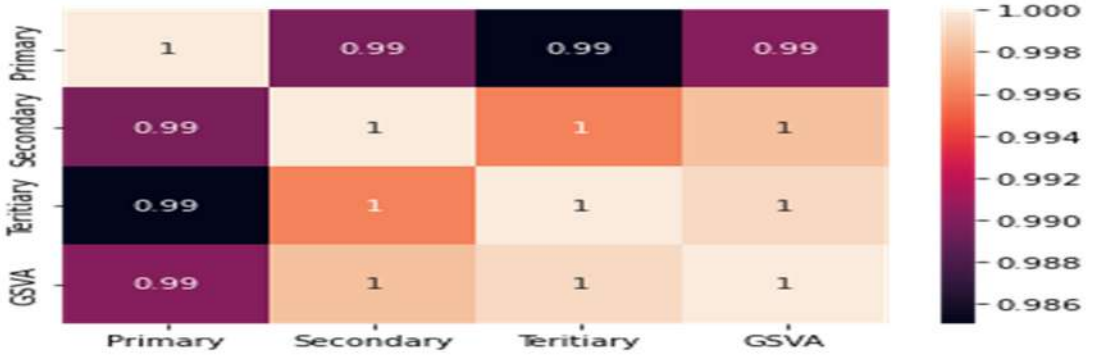
Result with inference

Firstly, I used a heatmap to study the correlation between the dependent variable (GSVA/NSVA) and independent variables for both the primary sector and total GSVA. My analysis revealed a positive correlation between the dependent variable and independent variables in both sectors.

Primary Sector

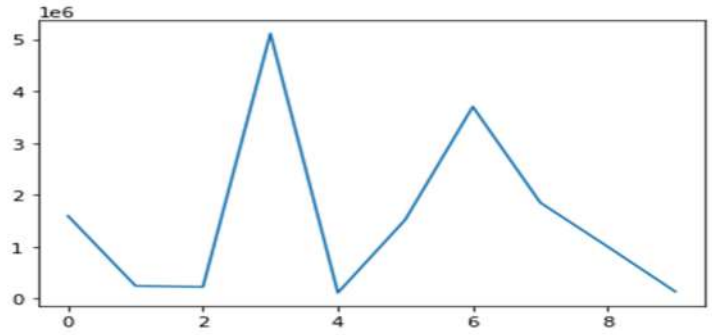


Total GSVA

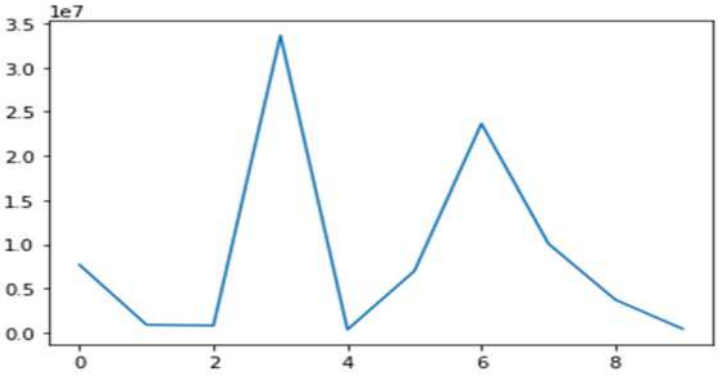


Next, I built a linear regression model to calculate the regression coefficient and fit the data for both the primary sector and total GSVA.

Primary Sector

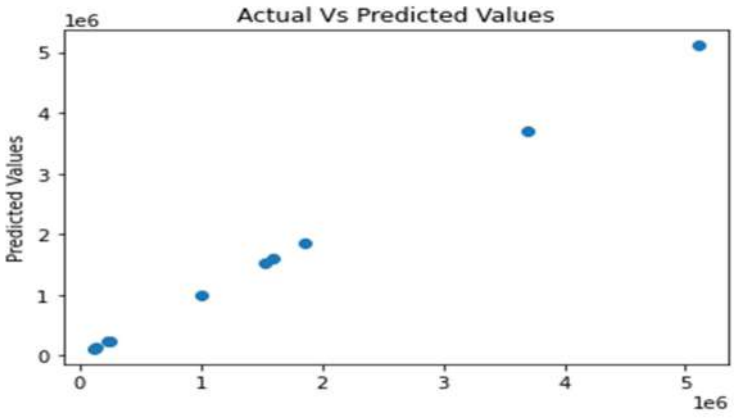


Total GSVA

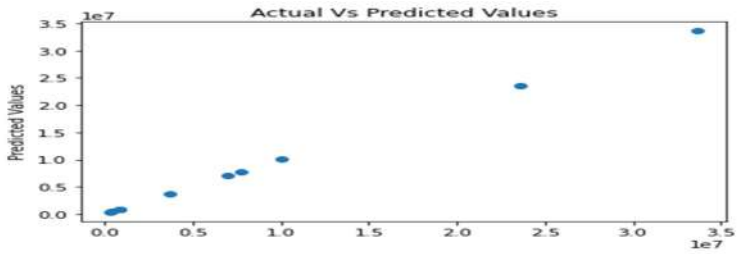


To evaluate the accuracy of my predictions, I used scatter diagrams to compare the predicted values with the actual values. My analysis showed that the plotted values were in a straight line, *indicating* that the model was a perfect fit.

Primary Sector

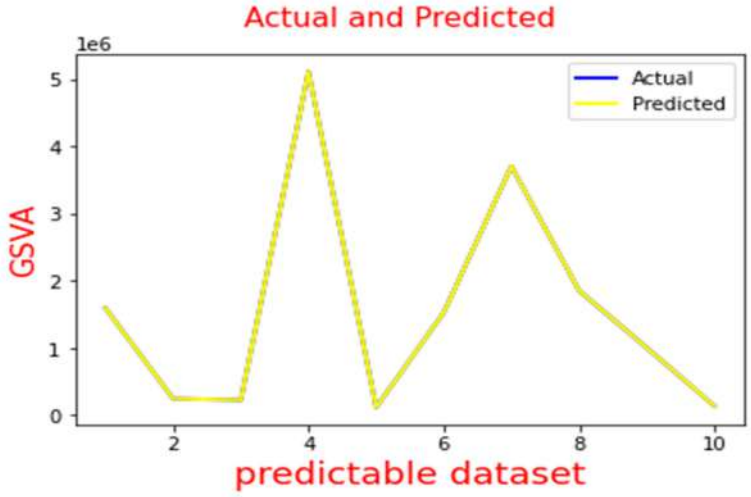


Total GSVA

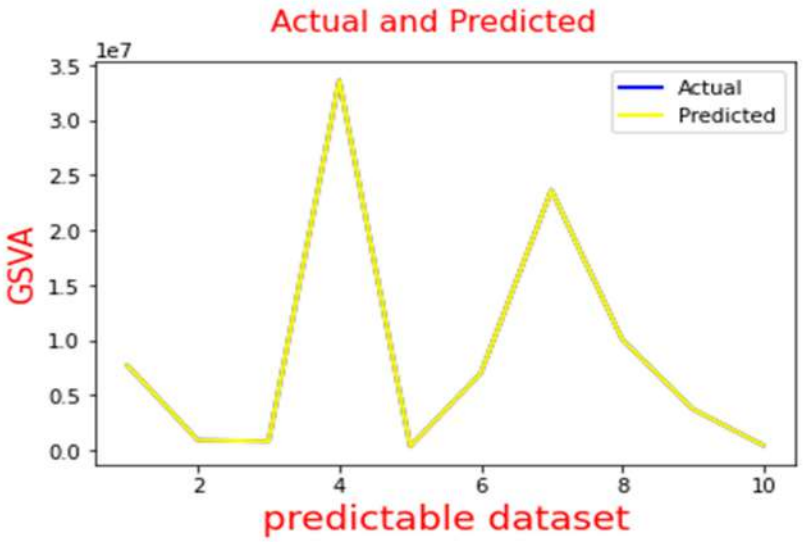


Finally, I conducted sector-wise predictions as well as predictions for the total GSVA, and calculated the mean squared error for both models using Linear Regression. My analysis revealed that there was no significant change in mean squared error for both models, indicating that the predictions were highly accurate.

Primary Sector



Total GSVA



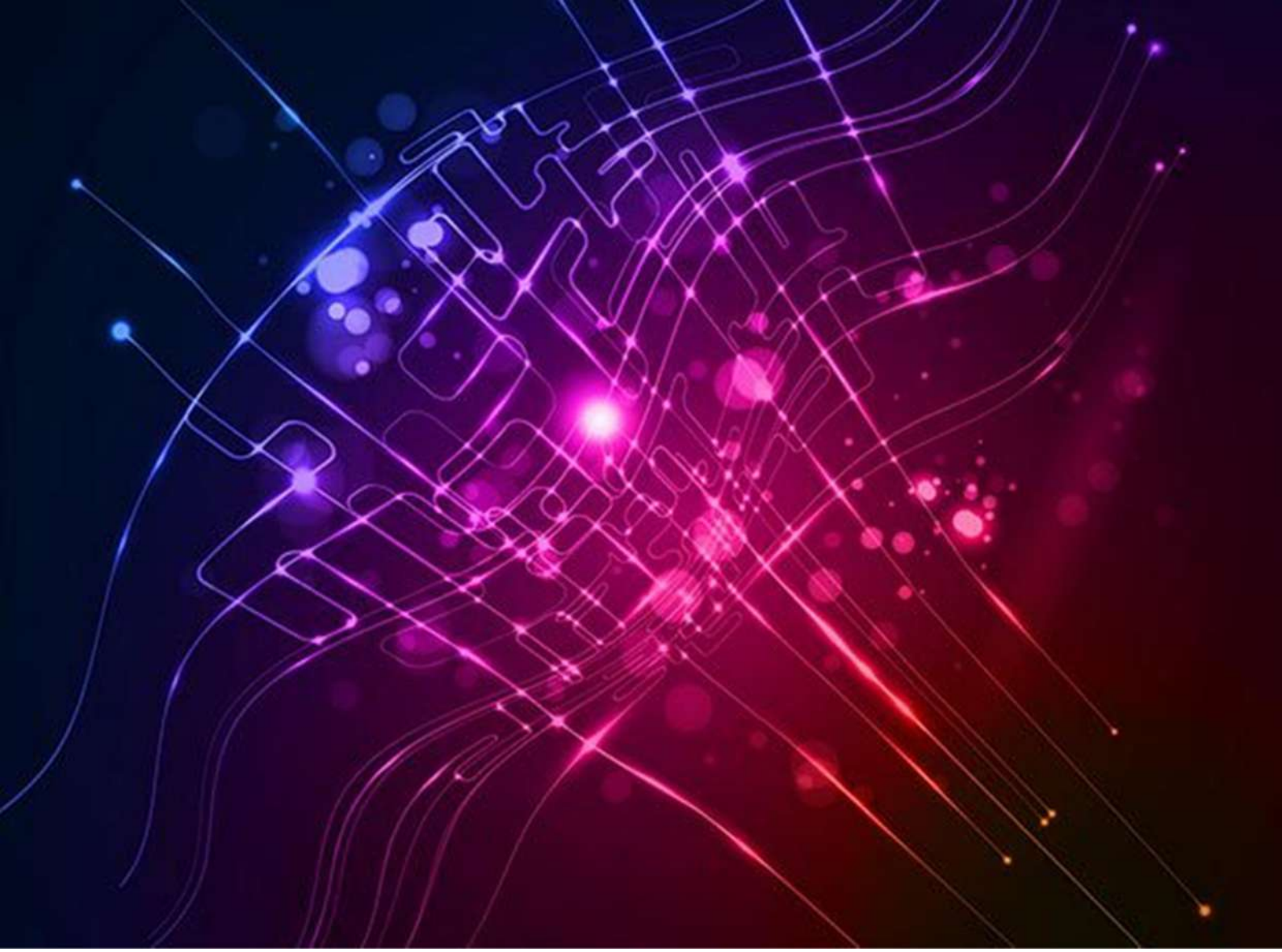
Conclusion

Overall, the methodology allowed me to identify the key drivers of economic growth in Kerala and build an AI-based forecasting model that accurately projected the GSVA/NSVA figures for the next five financial years.

References

https://www.geeksforgeeks.org/linear-regression-python-implementation/

https://www.analyticsvidhya.com/blog/2022/02/linear-regression-with-python-implementation/



Analyzing and predicting the market price of consumer products

Submitted By
Sri. Nidhin Babu, Statistical Assistant Grade II

Abstract

Using machine learning algorithms, this study proposes a methodology for analysing and predicting consumer product prices. The aim of the study is to analyse the factors that influence the pricing of consumer products and to develop a predictive model that can be used to estimate the prices of the consumer products. As part of the methodology, historical pricing data is collected from our department data library, data is cleaned and pre-processed, and machine learning algorithms are applied to build a predictive model. The study demonstrates the effectiveness of the proposed methodology by applying it to a case study of a consumer product. Based on the product's features and the market conditions, the predictive model is capable of accurately estimating the product's price.

Introduction

A complex task like analysing and predicting the price of consumer products involves various factors such as consumer behavior, supply and demand, competition, and macroeconomic influences. In order to analyse and predict the market price of consumer products, we should start by collecting information about historical prices, sales volumes, and market trends in order to establish a baseline. As a result of studying this data, we will be able to recognize patterns and correlations that can help us to understand the factors that influence prices in a more meaningful way.

Next, the opportunity to make predictions about future prices using statistical models and machine learning algorithms based on past data as well as current market conditions in order to make predictions about the future. A number of modeling techniques, such as linear regression, time-series analysis, and neural networks, are commonly used. The accuracy of the predictions will depend greatly on the quality and relevance of the data as well as the complexity of the market dynamics that are trying to model the quality and relevance of the data we use as a starting point. Also, it is crucial to consider other factors that may have an impact on prices, such as changes in regulations, as well as consumer preferences, that may be affecting them. In general, analysing and predicting the market price of consumer products is a challenging but important process for businesses and investors who are interested in making informed decisions about pricing and strategic decisions.

Grouping of consumer products

Depending on the price point of the product, consumers can be categorized into various categories, including the following:

Economical or budgetary products are products that tend to have a low price point and cater to customers who are looking for products that are affordable. The basic features they have are often lacking and they may also compromise on the quality or design of their product.

A mid-range product is one that offers a great balance between affordability and quality. Usually, these products are more expensive than economy products and they also offer more features and a better quality than economy products.

In the premium category, you will find products with high-quality features, high-quality design, and advanced technology that are more expensive than mid-range products. In order to be competitive, they need to cater to customers who are willing to spend more on luxury and exclusivity.

The Luxury Product category is comprised of high-end products that provide premium features, superior quality, and exclusive designs to their users. These products are usually offered at a premium price and are geared towards customers who place a high value on luxury, status, and exclusivity

Objective:-

- Analyze the underlying patterns in consumer prices for various products.
- to build a system capable of finding patterns in data
- A trend analysis examines the direction of the data.
- Short term Forecasting of prices of consumer products

Literature review

A similar study has been done in user-inputted text descriptions of its products, including details like product category name, brand name, and item condition. Using this data, they have created a model that predicts the price of a product listed on Mercari[1].

Another paper[2] addresses the problem of predicting direction of movement of stock and stock price index for Indian stock markets. The study compares four prediction models, Artificial Neural Network (ANN), Support Vector Machine (SVM), random forest and naive-Bayes with two approaches for input to these models. The first approach for input data involves computation of ten technical parameters using stock trading data (open, high, low & close prices) while the second approach focuses on representing these technical parameters as trend deterministic data. Accuracy of each of the prediction models for each of the two input approaches is evaluated.

A study [3] that forecasts CPI which is calculated in monthly periods each year to anticipate the possibility of a spike in the inflation rate. Forecasting the CPI makes use of past values, commonly known as time-series data (TSD). One method to assist the forecasting process on TSD is the Autoregressive Integrated Moving Average (ARIMA). Another method that can also be used for forecasting with linear data problems is the Artificial Neural Network (ANN). This study compared the two forecasting methods between ARIMA and ANN by predicting the Indonesian CPI value from January - December 2018. The TSD used is in data on the Indonesian CPI value between January 2009 and December 2017. This study indicates that the ANN method is better than ARIMA because it produces a smaller MSE of 59.23. However, ARIMA is also good because the two methods' forecast results are in the range of the CPI value.

Methodology

The objective of this study is to develop a machine learning model for predicting the short term forecast of price of consumer products in Thiruvananthapuram District, as well as to identify a machine learning technique that can be used to classify items into various predefined categories.

As part of the implementation process, the following steps will be taken:

- Collect a dataset containing time series data containing weekly price of consumer products items as well as a dataset containing the names of items and their labels.

Table 1

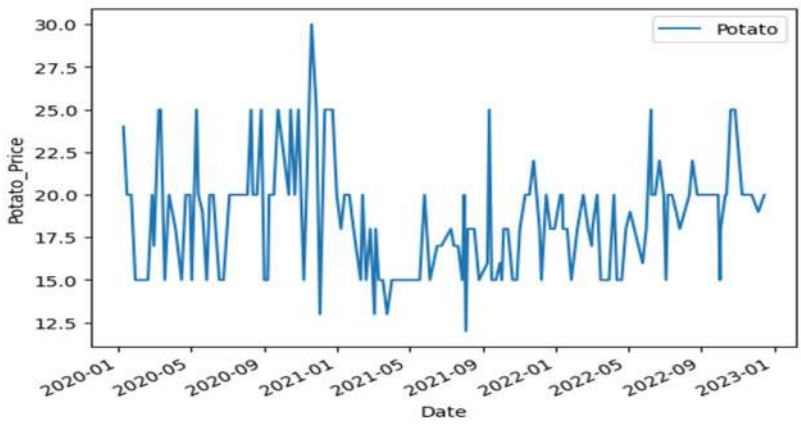
TEM	Potato	Onion	Onion- Sma	Carrot	Garlic	Ginger
07-01-2020	15	40	80	40	20	9
14-01-2020	20	30	70	40	20	8
21-01-2020	20	35	70	40	18	10
28-01-2020	15	35	60	40	24	8
30-01-2020	20	25	40	40	18	8
06-02-2020	15	25	40	40	24	10
13-02-2020	15	20	30	40	20	12
20-02-2020	20	20	30	40	18	10
27-02-2020	15	20	30	40	16	10
05-03-2020	13	20	30	40	14	8
12-03-2020	15	20	30	30	14	8
19-03-2020	20	17	30	30	12	8
26-03-2020	15	20	40	30	12	8
02-04-2020	20	15	30	20	12	14
08-04-2020	15	15	30	30	16	12
16-04-2020	20	15	30	20	14	8
23-04-2020	20	10	30	30	16	8
30-04-2020	20	10	30	24	12	8
07-05-2020	20	10	25	20	10	8
14-05-2020	19	10	24	20	12	8
21-05-2020	15	10	22	20	12	8
28-05-2020	18	10	20	20	12	8
04-06-2020	15	10	20	20	12	9
11-06-2020	15	10	20	20	12	8
18-06-2020	15	10	25	20	12	8
25-06-2020	15	10	23	20	12	8

- Prepare the data by cleaning, formatting, and possibly removing irrelevant information.
- To begin with, I have selected an item for which I want to predict the price as a first step.

Table 2

	Date	Price_Potato
	07-01-2020	15
	14-01-2020	20
	21-01-2020	20
	28-01-2020	15
	04-02-2020	20
	11-02-2020	15
	18-02-2020	15
	25-02-2020	20
0	03-03-2020	15
1	10-03-2020	13
2	17-03-2020	15
3	24-03-2020	20
4	05-05-2020	15
5	12-05-2020	20
6	19-05-2020	15
7	26-05-2020	20
8	02-06-2020	20
9	09-06-2020	20
0	16-06-2020	20
1	23-06-2020	19
2	30-06-2020	15
3	21-07-2020	18

Figure 1



- Understanding the underlying pattern of time series data by decomposing the data into various components of time series

Figure 2

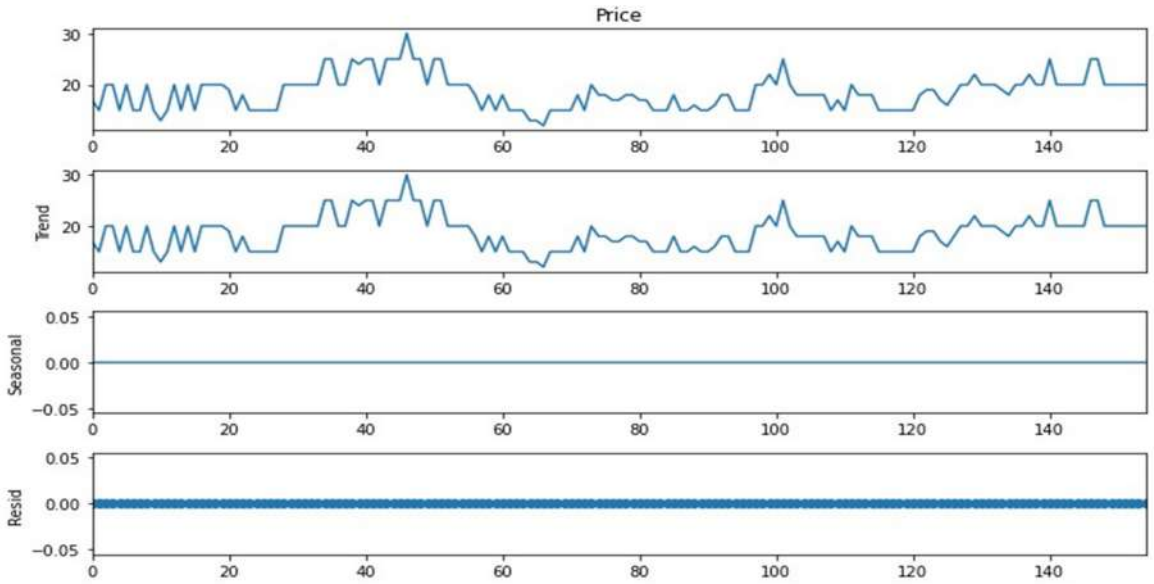


Figure 2 clearly shows that there is no residual (noise) component or seasonality component in the data, so we have to forecast using the ARIMA model. ARIMA forecasting requires stationary time series for the data for the forecast. The Dickey Fuller test is used to determine whether the data follows the condition of stationary. With null hypothesis as the data is stationary

1	.ADF Statistic:	-3.451925665842075
2	P-Value:	0.009311069858978352
3	n_lags	1
4	n_lags 4.Num of observations used for ADF Reggression and CriticalValues calculation	152
5	Critial Values:	1%, -3.474714913481481

The null hypothesis has been at 1% level of Significance

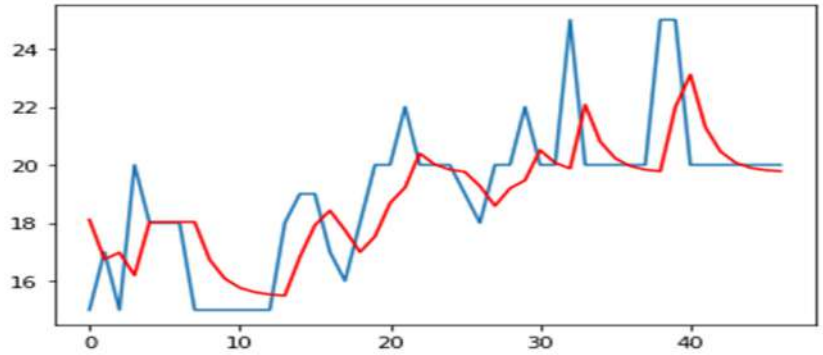
It is possible to find the best model using the autoarima function present in the pmdarima library. The auto arima function calculates the Akakike Information Criteria (AIC) for an ARIMA model based on the different parameters that are present in the model. It is decided that the best model is the one with the lowest AIC value. It is possible to find the best model using the autoarima function present in the pmdarima library. The auto arima function calculates the Akakike Information Criteria (AIC) for an ARIMA model based on the different parameters that are present in the model. It is decided that the best model is the one with the lowest AIC value.

- Best model: ARIMA(1,0,1) intercept Total fit time: 10.640 seconds p=1,d=0,q=1
- With minimum AIC value of 688.738688

To evaluate the model's performance, divide the data into training and test sets.

- 70% (106 data points)data selected for Training the model
- Remaining 30%(46 data points) for Testing the model
- Test RMSE: 1.909

Figure-3



Validity of the test

A final check is to analyze residual errors of the model. Ideally, Any plot for residuals must be uncorrelated and the distribution of the residuals should follow a Gaussian distribution with a zero mean. We can calculate residuals by substracting predicted values from actuals as below.

Figure 4

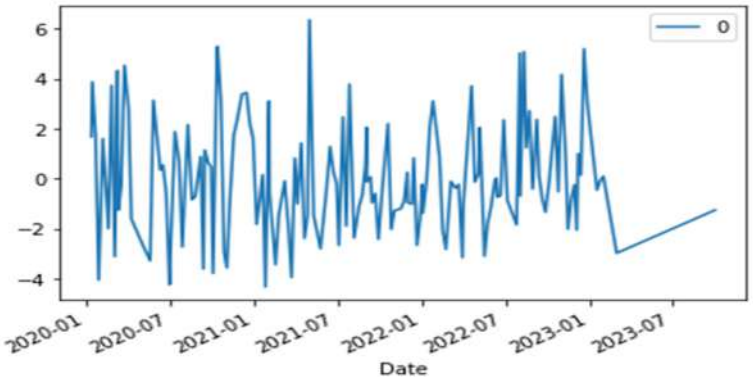
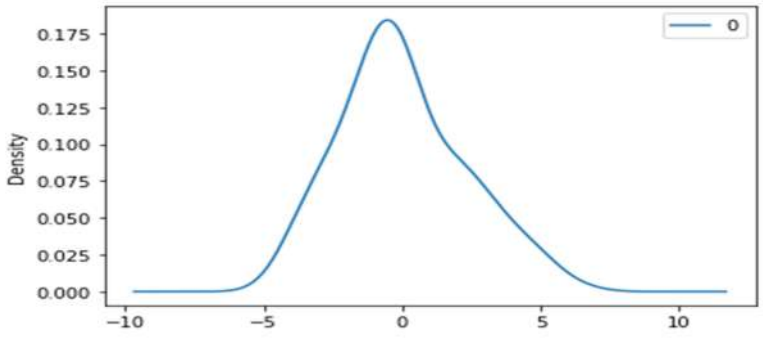


Figure 5



Using the model once it has been validated, it can be used to predict the price for new extended period. It would be helpful if you could provide the model with the relevant features of the product and obtain a price prediction as a result.

In order to make an accurate prediction out-of-sample, it is important to keep in mind that it will be determined by the quality of the data, the relevance of the selected features, and the performance of the machine learning algorithm used to make the prediction. Additionally, the model should be continuously monitored and updated as new data becomes available, in order to ensure that the model remains accurate and relevant in the future.

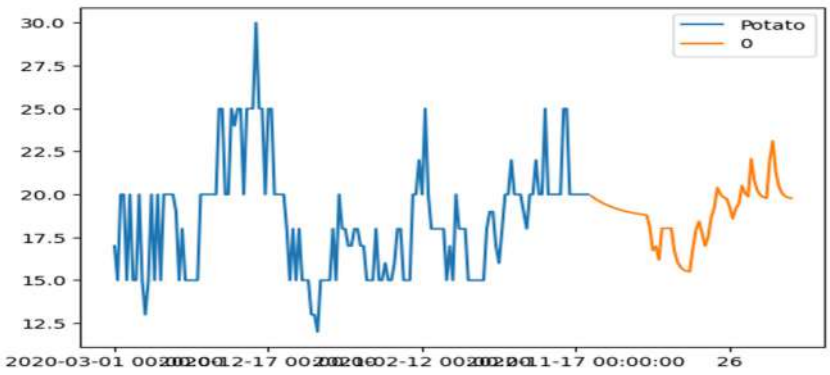
Libraries and Tools used

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

1. **Scikit-learn:** A popular machine learning library in Python for tasks such as classification, regression, and clustering.
2. **Pandas:** A data manipulation library in Python that is often used to clean, transform, and manipulate data for machine learning.
3. **NumPy:** A Python library for numerical computing that is often used for linear algebra and array operations in machine learning.
4. **Matplotlib:** A Python plotting library that is often used to create visualizations of data for machine learning.
5. **Seaborn:** A data visualization library in Python that is built on top of Matplotlib and provides a high-level interface for creating statistical graphics.
6. **pmdarima :**is a Python library for time series analysis and forecasting. It is built on top of the statsmodels library and provides an interface for automating the process of selecting the best ARIMA model for a given time series.

Results and Inference

Figure 6

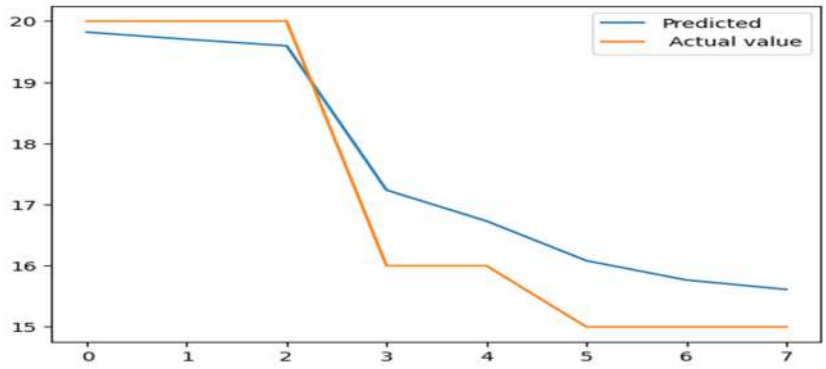


A comparison has been made between the date available after out of sample prediction and the date available after the prediction. the value displayed in the table 2.

Table 3

Date	Predicted value	Actual Value
07-02-2023	19.8204	20
14-02-2023	19.7029	20
21-02-2023	19.5964	20
28-02-2023	17.2371	16
07-03-2023	16.7309	16
14-03-2023	16.0825	15
21-03-2023	15.7663	15

Figure 7



Conclusion

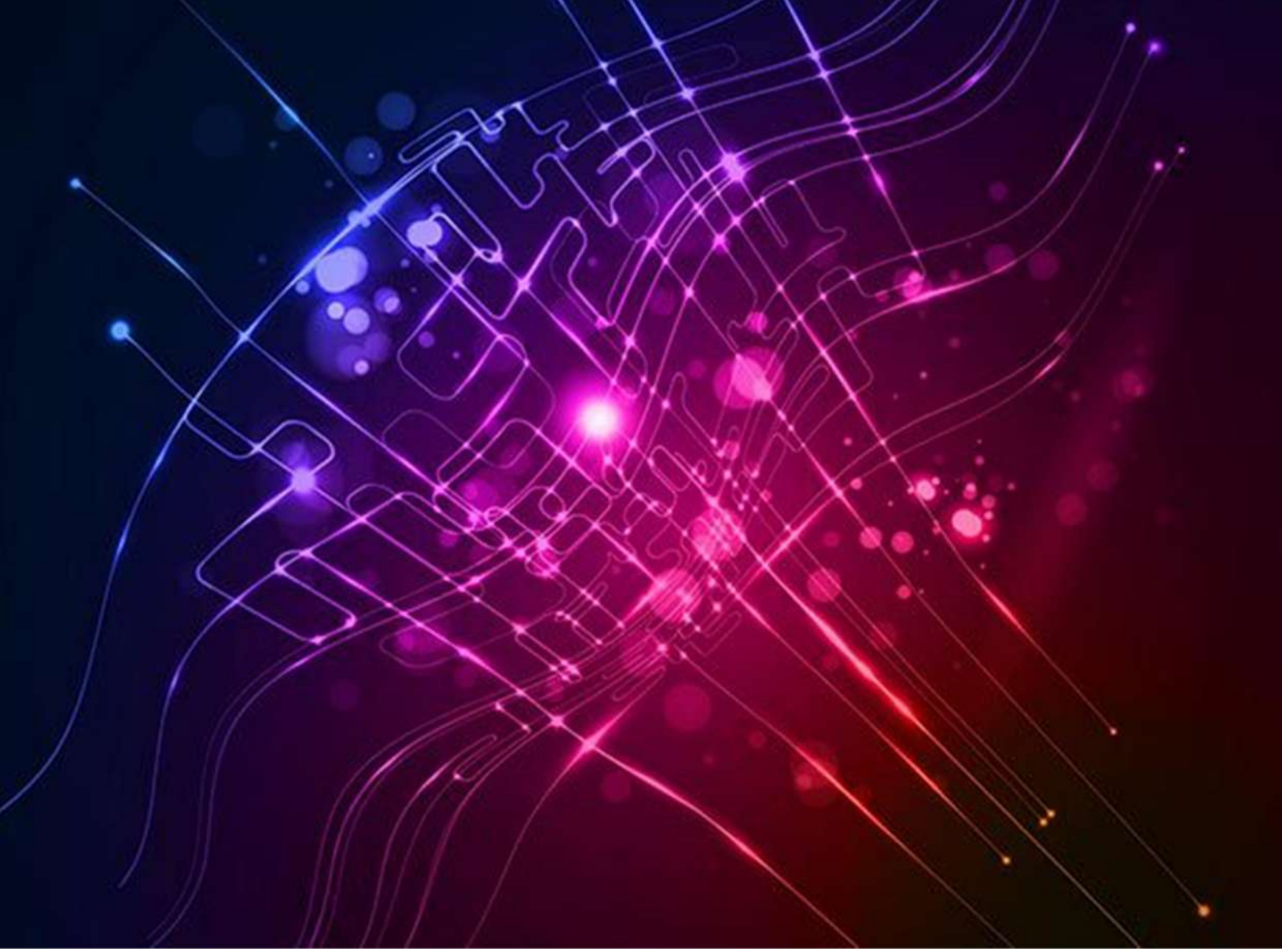
Although the performance of these methods depends on the data set and the specific features chosen, it is difficult to draw definitive conclusions from them. The potential of machine learning techniques for predicting the Consumer products price under a variety of scenarios requires further research.

Reference:

[1] Kumar, A. (2020).Price Prediction using Machine Learning Regression — a case study

[2] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. Expert Systems with Applications, 42(1), 259-268.

[3] Time-series Modeling for Consumer Price Index Forecasting using Comparison Analysis of AutoRegressive Integrated Moving Average and Artificial Neural Network Intan Yuniar Purbasari1 , Fetty Tri Anggraeny1 and Nindy Apsari Ardiningrum2 1 Department of Informatics, Faculty of Computer Science, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Raya Rungkut Madya, Surabaya, Indonesia



Crop yield prediction using Random Forest Algorithm for different crops in Kerala

Submitted By
Smt. Neethymol Kurian, Statistical Assistant Grade II

Introduction

Crop yield prediction is a critical aspect of agriculture that can help farmers make informed decisions regarding planting, harvesting, and crop management. Accurate crop yield prediction can improve crop productivity and minimize crop losses, which can significantly impact a farmer's income and food security.

In Kerala, agriculture is an essential sector of the economy, contributing significantly to the state's GDP. The state's topography and climatic conditions make it ideal for cultivating a wide variety of crops such as paddy, coconut, rubber, spices, and vegetables. However, due to unpredictable weather conditions and various pest and disease attacks, crop yield in Kerala is often volatile.

To address this issue, we propose using the Random Forest algorithm to predict the crop yield of different crops like Coconut, Tapioca, Banana, Areca nut, Pepper, Ginger, Turmeric in Kerala. The Random Forest algorithm is a supervised machine learning technique that has been shown to be effective in predicting yield in various crops. This algorithm can handle large datasets with many variables and is robust to outliers and missing data.

Objective

The aim of this project is to develop a predictive model using the Random Forest algorithm to estimate crop yield in Kerala. The model will be trained on crop area data, weather data, soil data, and other relevant factors. The output of the model will provide farmers with an estimate of their crop yield for the upcoming year, allowing them to plan and manage their crops more efficiently.

The project's outcomes are expected to help farmers in Kerala make data-driven decisions, improve crop yield, and increase their income. The proposed model could also be extended to other regions to enhance crop productivity and food security.

Methodology

A. Dataset

Data plays an important role in Machine Learning. To design and perform crop yield prediction system data is taken from various districts of Kerala state. The description of the attributes mentioned in the dataset are mentioned in the table below. The data used in the project is taken from the Agricultural Statistics 2005-2020 Report published by DES, Kerala. (*Dataset added as Annexure*)

Table 1: Attributes from district wise crop production dataset

Sl. No.	Attribute	Description
1.	Agriculture Year	Five agriculture years data are taken starting from 2015-16 to 2019-20.
2.	District	14 Districts of Kerala
3	Crop	Coconut, Tapioca, Banana, Areca nut, Pepper, Ginger, Turmeric
4.	Area (Hectare)	Area considered for various crops
5.	Production (Tonnes) Coconut Production (Nuts)	Production of crops in the area specified.
6.	Cost of Cultivation (Rs/hectare)	Cost required for the cultivation of various crops per hectare
7.	Rainfall (mm)	Actual rainfall(mm) of all districts in Kerala
8.	Soil Composition	Nitrogen, Phosphorous and Potassium values

B. Proposed System

A multiple linear regression model cannot be used in this case as the assumption of linearity is not satisfied.

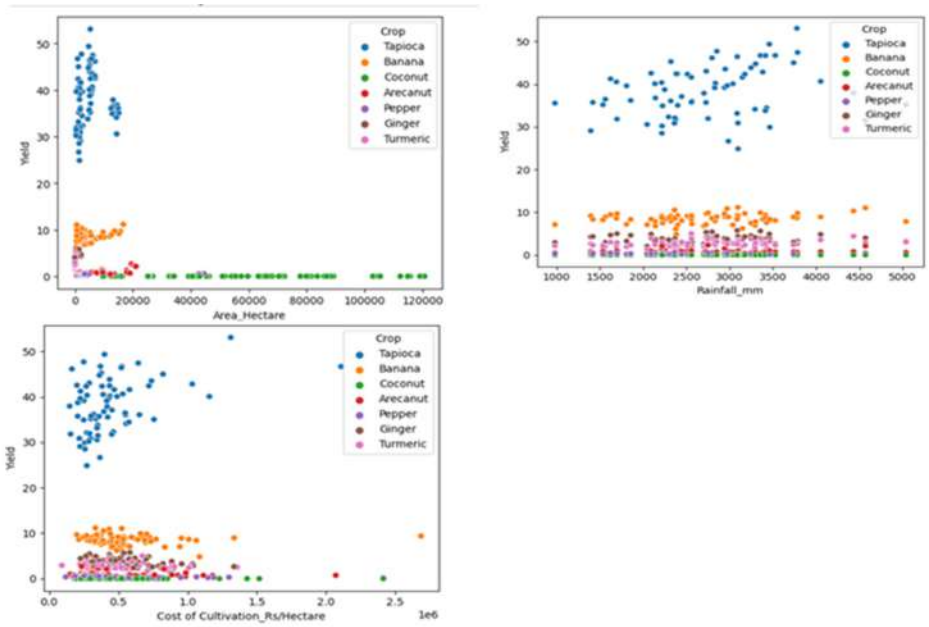


Fig.1: Checking linear relationship between yield (dependant variable) and Area, Rainfall, Cost of Cultivation(independent variables)

There are many non linear machine learning models used for classification and regression. Here we use the Random Forest approach for Crop yield prediction. Random forest is a basically supervised learning algorithm. It creates decision trees on different data samples and then predict the data from each subset and then by voting gives better the solution for the system. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of over fitting.Given below is a pictorial representation of the model.

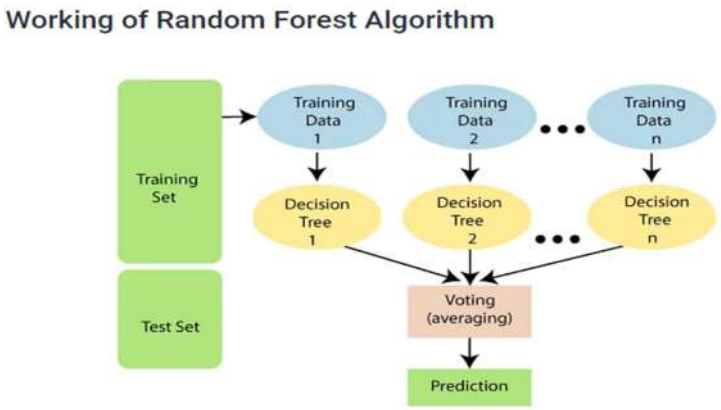


Fig.2: Proposed diagram for Crop Yield Prediction

Before applying the model to the data set apply the Data Preprocessing Techniques.

- Detect missing values in the dataset. These missing values are replaced with the mode of soil composition (N) since it is an ordinal variable.
- Code the categorical variables – N, P, K.
- We have Area and Production in our data set . Calculate yield since our objective is to estimate the yield of various crops.

	Agriculture Year	District	Crop	Area_Hectare	Cost of Cultivation_Rs/Hectare	Rainfall_mm	N	P	K	Yield
0	2015	Thiruvananthapuram	Tapioca	14585	395217	2142.1	1	2	2	36.770312
1	2015	Thiruvananthapuram	Banana	2676	489241	2142.1	1	2	2	6.852765
2	2015	Thiruvananthapuram	Coconut	72340	371454	2142.1	1	2	2	0.008750
3	2015	Thiruvananthapuram	Arecanut	1036	263626	2142.1	1	2	2	0.620656
4	2015	Thiruvananthapuram	Pepper	2293	472349	2142.1	1	2	2	0.423899

- Label Encoding is applied to Districts and Crops.
- The dataset will be split to two datasets, the training and test dataset. The data is usually tend to split unequally because training the model usually requires as much data points as possible. The common splits are 70/30 or 80/20 for train/test.

The training dataset is the initial dataset used to train ML algorithm to learn and produce right predictions. (80% of dataset is training dataset).

The test dataset, however, is used to assess how well ML algorithm is trained with the training dataset. You can't simply reuse the training dataset in the testing stage because ML algorithm will already "know" the expected output, which defeats the purpose of testing the algorithm. (20% of dataset is testing dataset)

Tools and Libraries used

- Python Programming
- Libraries – numpy, pandas, matplotlib, sklearn, seaborn, tkinter

Result and Analysis

```
In [72]: from sklearn.ensemble import RandomForestRegressor
         regr=RandomForestRegressor(max_depth=2,random_state=0,n_estimators=100)
         regr.fit(X_train, Y_train)
         y_pred=regr.predict(X_test)

         from sklearn.metrics import mean_squared_error as mse
         from sklearn.metrics import mean_absolute_error as mae
         from sklearn.metrics import r2_score
         print('MSE=',mse(y_pred,Y_test))
         print('MAE=', mae(y_pred,Y_test))
         print('R2 score=',r2_score(y_pred,Y_test))
```

- The evaluation metric is based on R^2 (coefficient of determination) regression score function, that will represent the proportion of the variance for items (crops) in

the regression model. R^2 score shows how well terms (data points) fit a curve or line.R^2 is a statistical measure between 0 and 1 which calculates how similar a regression line is to the data it's fitted to. If it's a 1, the model 100% predicts the data variance; if it's a 0, the model predicts none of the variance.

```
R2 score= 0.9114926858300401
```

Note: 91% of the proportion of variance is explained by the Random Forest model for Crop yield Prediction.

The graph below shows the most important factors that affect crop yield. The importance of this graph is that we can eliminate the irrelevant variables based on its importance, from the model and improve the accuracy of the model.

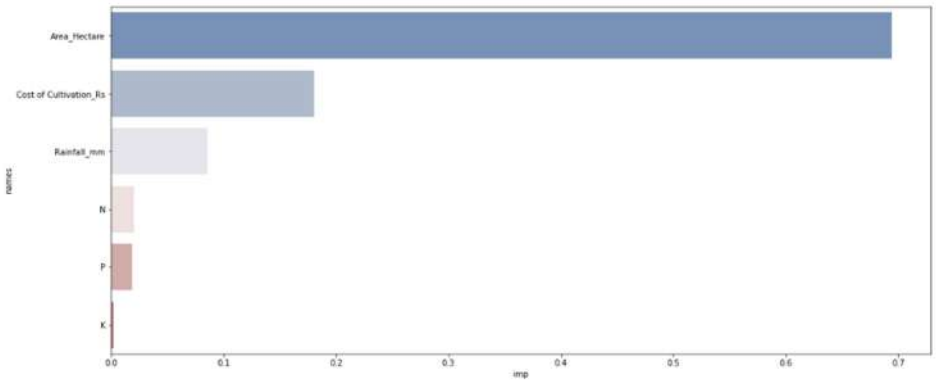


Fig. 3: Feature importance of variables in the model

The goal of the project is to build a graphical user interface which will help the users, policy planners to predict the crop yield based on area, cost of cultivation, rainfall, soil composition.

Fig.4: Graphical User Interface that shows the predicted crop yield after selection of different parameters by user.

As shown in figure 4, users of the application can see the home page and will able to enter the details such as the district code, crop code, area, cost, rainfall, soil

composition details are filled on the home page, user press the predict button then the request will send toward the server and the system gives a prediction using the model and trained under the random forest algorithm. The result of the prediction of the crop yield which is sent to the respective user and the unit of the crop yield is considered in tons.

References

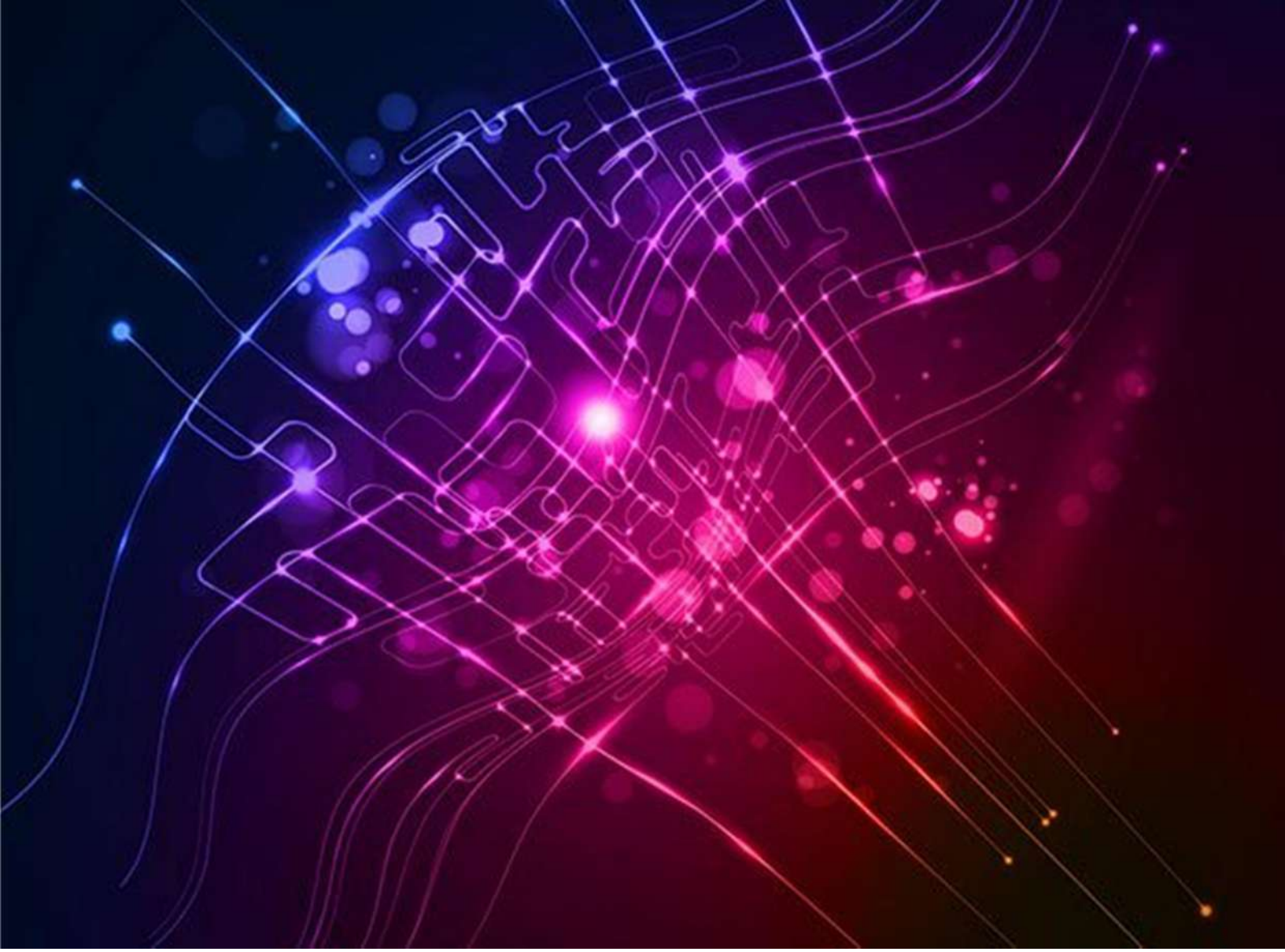
[1] Kiran Moraye, Suyog Nikam, Smit Thakkar, Crop Yield Prediction Using Random Forest Algorithm for Major Cities in Maharashtra State, International Journal of Innovative Research in Computer Science & Technology (IJIRCST).

[2] Sivanandhini P, Prakash „.Crop Yield Prediction Analysis using Feed Forward and Recurrent Neural Network, International Journal of Innovative Research in Computer Science & Technology.

[3] Crop Yield Prediction, https://www.kaggle.com/code/kushagranull/crop-yield-prediction

Sample dataset

Agriculture Year	District	Crop	Area _Hectare	Production _Tonnes	Cost of Cultivation_ Rs/Hectare	Rainfall _mm	N	P	K
2015	Thiruvananthapuram	Tapioca	14585	536295	395217	2142.1	Low	Medium	Medium
2015	Thiruvananthapuram	Banana	2676	18338	489241	2142.1	Low	Medium	Medium
2015	Thiruvananthapuram	Coconut	72340	633	371454	2142.1	Low	Medium	Medium
2015	Thiruvananthapuram	Arecanut	1036	643	263626	2142.1	Low	Medium	Medium
2015	Thiruvananthapuram	Pepper	2293	972	472349	2142.1	Low	Medium	Medium
2015	Thiruvananthapuram	Ginger	95	346	554412	2142.1	Low	Medium	Medium
2015	Thiruvananthapuram	Turmeric	73	185	487430	2142.1	Low	Medium	Medium
2016	Thiruvananthapuram	Tapioca	14628	520143	422946	981	Low	Medium	Medium
2016	Thiruvananthapuram	Banana	2776	19826	467942	981	Low	Medium	Medium



Changing trends in Cause of Deaths in Kerala

An analysis of cause of death data over the years from 2012 to 2021

Submitted By
Kumari. Minu Merin Andrews, Statistical Assistant Grade II

Introduction

The scheme of Medical Certification of Cause of Death (MCCD) was introduced in the country under the provisions of Registration of Births and Deaths (RBD) Act, 1969. Medical Practitioner attending the deceased at the time of death fill the medical certification form recommended by World Health Organization. The forms are sent to concerned Registrar of Births and deaths for tabulation . The cause of death reported are translated in to medical codes contained in the International Classification of diseases published by the World Health Organisation. After transmission to Additional chief Registrars office the State subsequently sends data to the Office of the Registrar General of India in the standardised format for National level consolidation.

In Kerala Medical Certification of cause of death scheme has been introduced in Thiruvananthapuram, Kollam, Kochi and Kozhikode corporations and Alappuzha Municipality.

Objectives

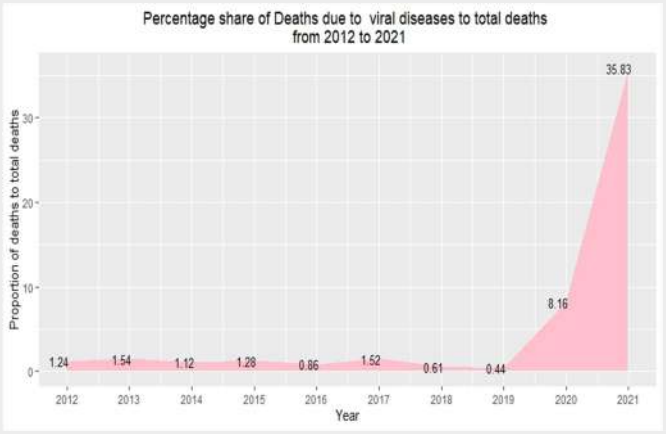
Analysis aimed to understand how causes of death changed over years from 2012 to 2021. Analysis has been made on the incidence of mortality due to nineteen causes of death disaggregated by gender and age.

Objective was to understand

- How the causes of death changed over time.
- Biggest leading cause of death associated with each age group
- Break down of deaths due to each cause in various age group

Leading Causes of Death

Diseases of circulatory system accounted for highest share among different causes of deaths from year 2012 to 2020 . There has been a change in this trend in the year 2021 with deaths due to certain infectious and parasitic diseases becoming biggest killer. Year 2021 observed surging share of deaths due to infectious and parasitic diseases accounting for 38.05% of total deaths among which viral diseases accounting for 94.16% . Diseases of circulatory system ranked second in 2021 with 20.69% of deaths. Endocrine nutritional and metabolic diseases accounted for 11.57% of total deaths in 2021 and is the third leading cause of death. Deaths due to neoplasms which ranked second from year 2012 to 2019 and third in 2020 was the fourth leading cause of death in 2021 responsible for 9.65% of total deaths. Diseases of respiratory system was the fifth leading cause in 2021 with 4.83% of total deaths in 2021. Diseases in the digestive system led to 4.12% of total deaths in 2021 and is the sixth leading cause. Injury, Poisoning and certain other external causes accounted for 2.52% and diseases of genitourinary system accounted 2.15% in 2021.



According to data from Medical Certification of Cause of death viral diseases accounted for 35.83% of total deaths in year 2021. 35.46% and 36.42% of total male and total female deaths respectively.

Age Wise Analysis

- From 2012 to 2020 leading cause of death in age group 70 and above was diseases of circulatory system. In 2021 leading cause was infectious and parasitic diseases. 43.47% of total mortality due to infectious and parasitic diseases was in people above 70 years of age. 44.63% of total deaths due to circulatory system occurred in this age group. In 2021 42.47% of total deaths in this age group occurred due to infectious and parasitic diseases, 23.7% of total deaths were due to diseases of circulatory system and 12.45% occurred due to endocrine, nutritional and metabolic diseases. Percentage share of deaths due to diseases of circulatory system in years 2012 and 2021 are 39.19% and 23.7% respectively.
- In 65-69 age group also leading cause of death in 2021 is infectious and parasitic diseases. In this age group during 2021 percentage share of deaths due to infectious and parasitic diseases is 38.34%. 14.78% of total deaths due to infectious and parasitic diseases was in this age group in 2021. Diseases of circulatory system which was the leading cause of death in this age group from 2012 to 2020 is the second leading cause in 2021. Diseases of circulatory system accounts for 34.61% and 23.08% of total deaths in this age group for years 2012 and 2021 respectively. Endocrine nutritional and metabolic diseases is the third leading cause in 2021 whose percentage share is 13.81%. Neoplasms which was the second leading cause from 2012 to 2019 and third leading cause in 2020 has become fourth leading cause in 2021, with 9.91% of total deaths in this age group.
- Those aged between 55 and 64 years diseases of circulatory system which was the leading cause from 2012 to 2020 was replaced by infectious and parasitic diseases in 2021 accounting for 36.71% of total deaths in that age group. Second leading cause is diseases of circulatory system in 2021 with share of 21.08%. Endocrine nutritional and metabolic diseases is the third leading cause in 2021 whose share is 13.1%. Neoplasms which was the second leading cause from 2012 to 2019 and third leading cause in 2020 has become fourth leading cause in 2021 with 12.36% share.
- Certain infectious and parasitic diseases which was the fifth leading cause of death in 45-54 age group with a share of 8.94% in 2012 has become leading cause of death in 2021 accounting for 36.42% deaths. Neoplasms was the second leading

cause of death in years 2012 to 2016 and 2019 to 2020. In 2017 and 2018 Neoplasms was the leading cause of death in this age group. Share of Neoplasm is 19.51%, 22.19%,21.86%,16.75% and 11.76% for years 2012,2018,2019,2020 and 2021 respectively. Diseases of digestive system was the third leading cause of death from years 2012 to 2017 and fourth leading cause in years 2019 and 2020. During 2021, Diseases of circulatory system, Neoplasms, Endocrine nutritional and metabolic diseases, Diseases of digestive system ranked second, third, fourth and fifth respectively.

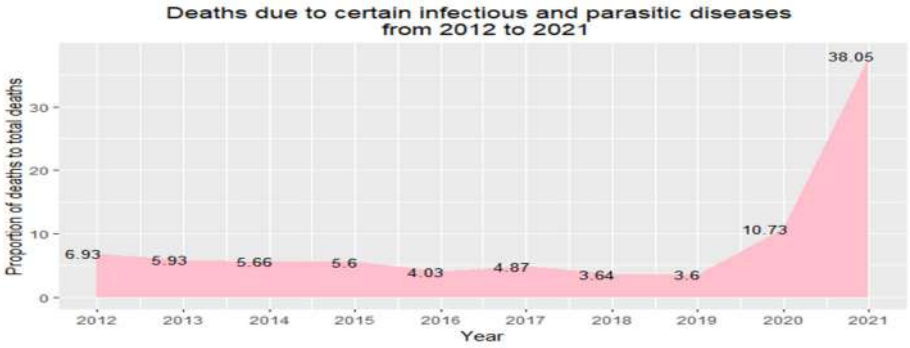
- For those aged between 35-44 leading causes of deaths were neoplasms and diseases of circulatory system. In 2016 leading cause was diseases of digestive system. Diseases of digestive system ranked third for all other years from 2012 to 2020. During 2021 Certain infectious and parasitic diseases, diseases of circulatory systems, neoplasms and diseases of digestive system ranked first, second, third, fourth and fifth respectively.
- In age group 25-34 leading cause of death for years 2012,2013,2015 to 2020 and second leading cause of death in 2014 was injury poisoning and certain other external causes. In 2012 percentage share of this cause was 18.82 % and in 2021 share is 16.74%, 20.26% of total male deaths and 11.83% of total female deaths. Neoplasms was the second leading cause in 2013 and 2015 to 2020. Share of neoplasm in years 2012, 2021 is 14.8% and 16.74% respectively. Diseases of circulatory system ranked second in 2012 and third from 2013 to 2017 and fourth from 2018 to 2021. In 2012 and 2021 percentage share due to diseases of circulatory system in this age group was 16.76% and 8.37% respectively. In 2021 leading cause was certain infectious and parasitic diseases accounting for 30.34% of total deaths in that age group.

In the year 2021 Certain infectious and parasitic diseases was the leading cause of death for all above 25 years age.

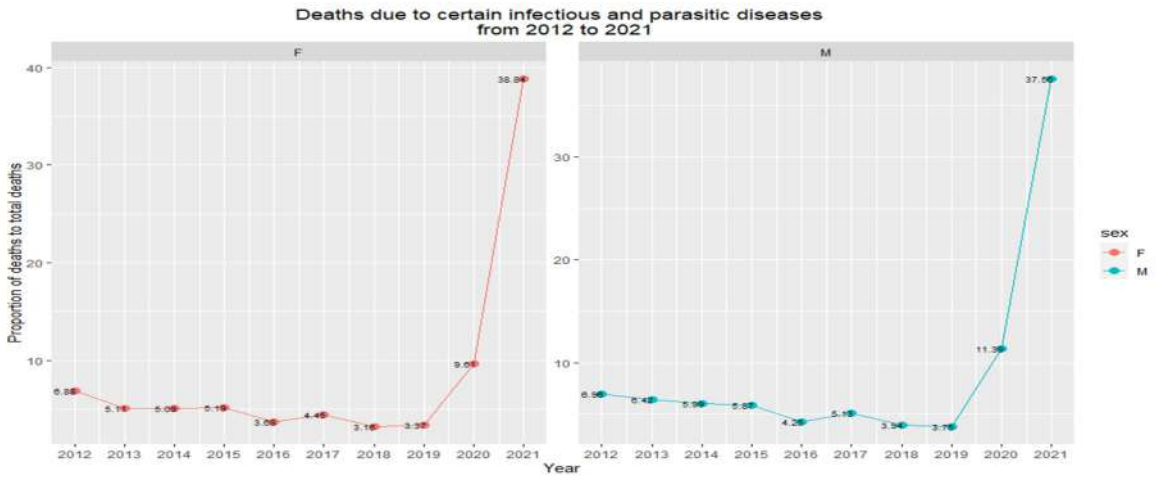
- In age group 15-24 leading cause of death is injury, poisoning and certain other external causes for all years from 2012 to 2021. During year 2021 injury, poisoning and certain other external causes accounted for 21.01% of total deaths in this age group, 29.91% of total male deaths and 10.53% of total female deaths in this age group occurred due to injury, poisoning and certain other external causes. There is significant difference between male and female proportion of deaths due to injury, poisoning and certain other external causes. In 2021 second leading cause of death is neoplasms (19.32% of total deaths) and third leading cause is infectious and parasitic diseases (17.87% of total deaths).
- In 5-14 age group leading cause of death is neoplasms for all years from 2012 to 2021. During 2021 percentage share of neoplasms to total deaths was 23.22%. Leukaemia accounts for 12.8% of total deaths in 5-14 age group (19.05% of total male deaths and 6.6% of total female deaths). In 2021 second leading cause is certain infectious and parasitic diseases (16.11% deaths) and third leading cause is injury, poisoning and certain other external causes (13.27% deaths).

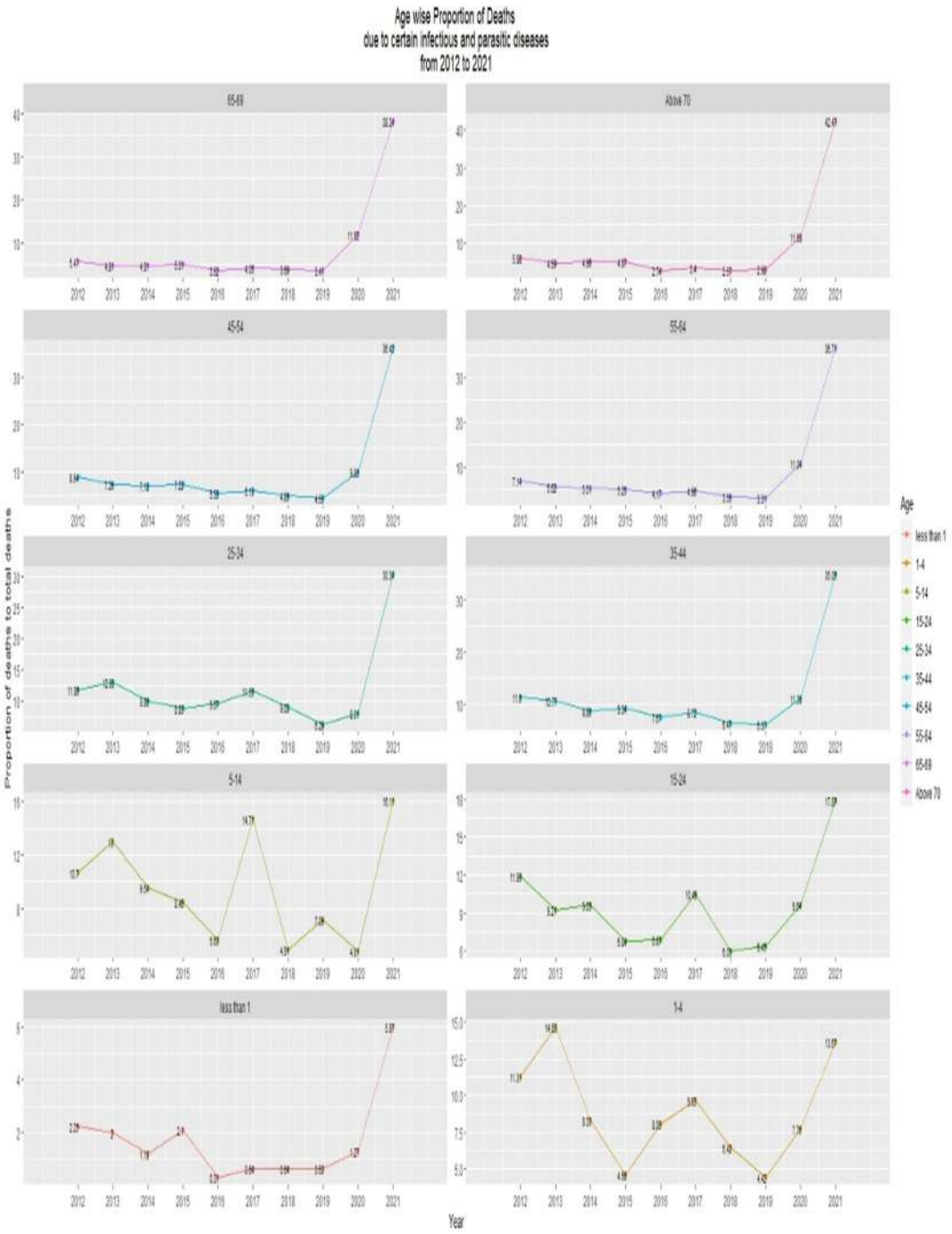
- For children of age group 1-4 Congenital Malformations, Deformations and Chromosomal Abnormalities was the leading cause from year 2015 to 2021. Certain Infectious and parasitic diseases was the third leading cause during year 2021.
- For children below 1 year certain conditions occurring in perinatal period is leading cause and second leading cause was congenital malformations, deformations and chromosomal abnormalities for years from 2012 to 2021. Certain Infectious and parasitic diseases has become the third leading cause of death in 2021 in this age group.

CERTAIN INFECTIOUS AND PARASITIC DISEASES

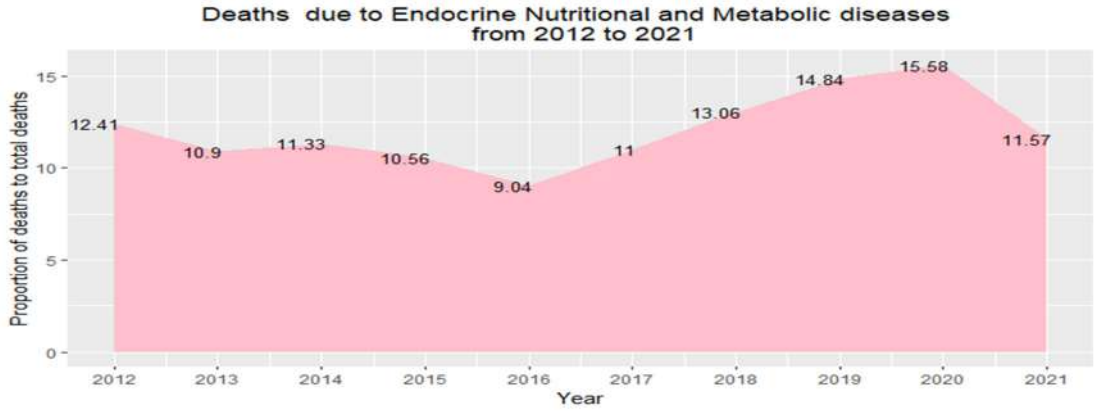


Proportion of deaths due to certain infectious and parasitic diseases which accounted for 6.93% of total deaths in 2012 has increased to 10.73% in 2020 and 38.05% in 2021. Intestinal infectious diseases, Tuberculosis, Other bacterial diseases, Infections with a predominantly sexual mode of transmission , Viral diseases, Protozoal diseases, Other certain infectious & parasitic diseases and late effects of infectious & parasitic diseases are the causes coming under this group. In 2020 and 2021 viral diseases accounted for 8.16% and 35.83% of total deaths . Not much gender gap is observed in the percentage share of deaths to total deaths in 2021, 37.56% of total male deaths and 38.84% of total female deaths . Percentage share of deaths increased in all age groups in year 2021 compared to 2020 and has become the leading cause of death for all those above 25 years of age. Out of total deaths due to infectious and parasitic diseases majority share is for people above 70 years, 43.47%. 14.78% in 65-69 age group, 21.24% in 55-64 age group, 10.83% in 45-54 age group and 4.47 % in 35-44 years of age.

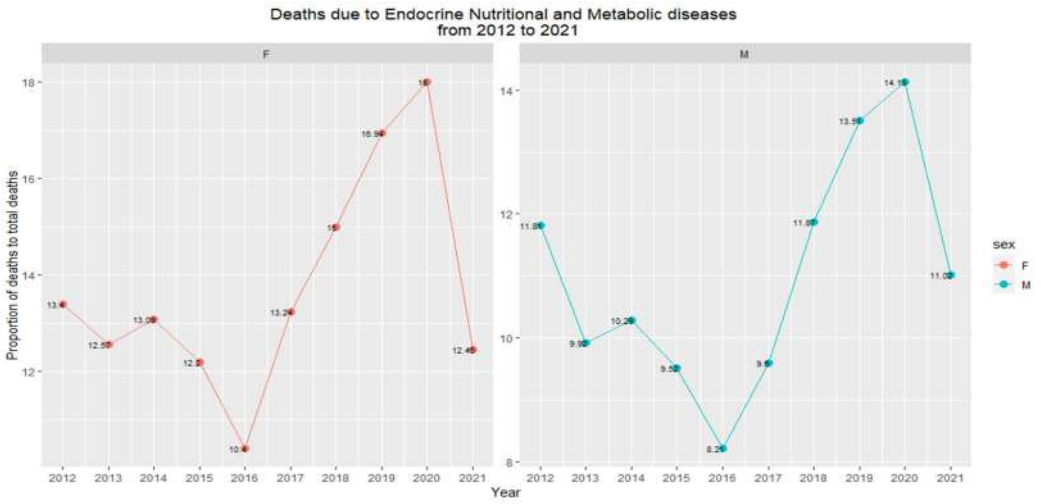




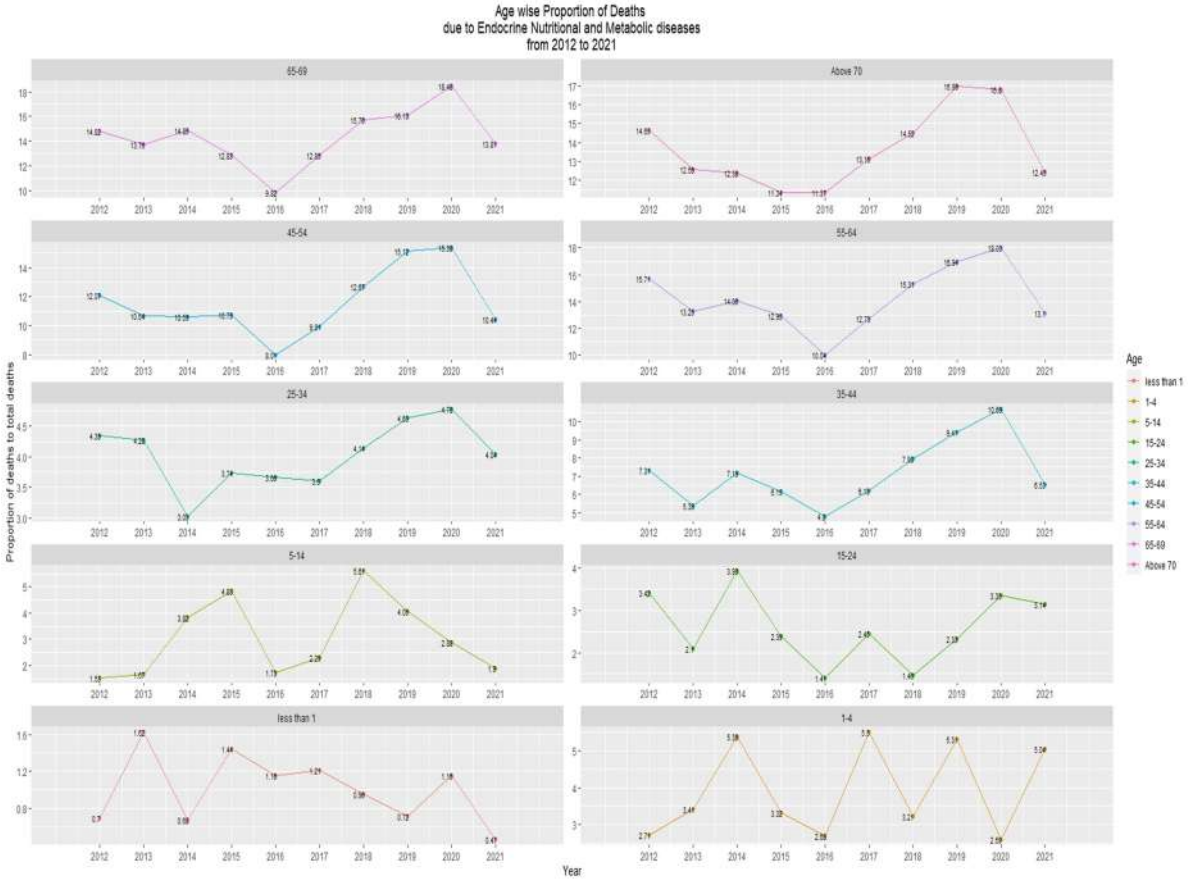
Endocrine, Nutritional and Metabolic diseases



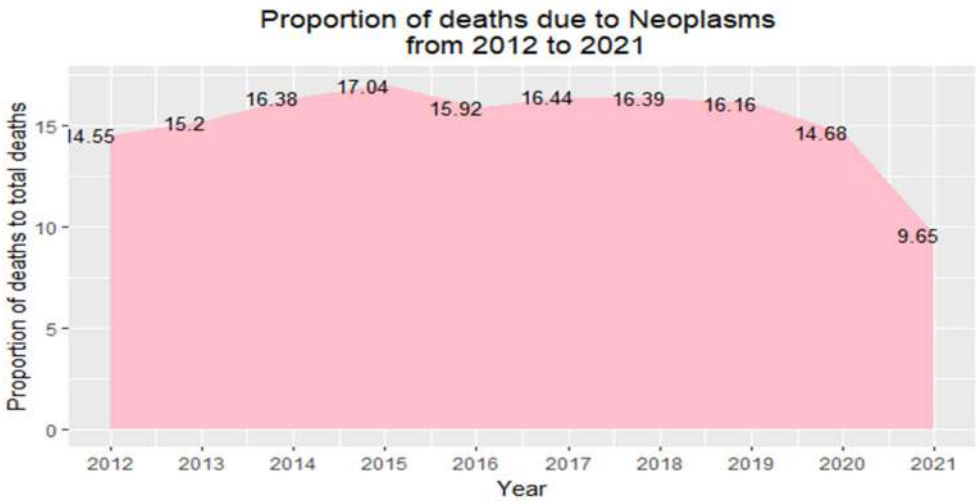
Percentage share of deaths due to Endocrine, Nutritional and Metabolic diseases has been steadily increasing from year 2016 to 2020. In 2021 share decreased from 15.58% to 11.57%.



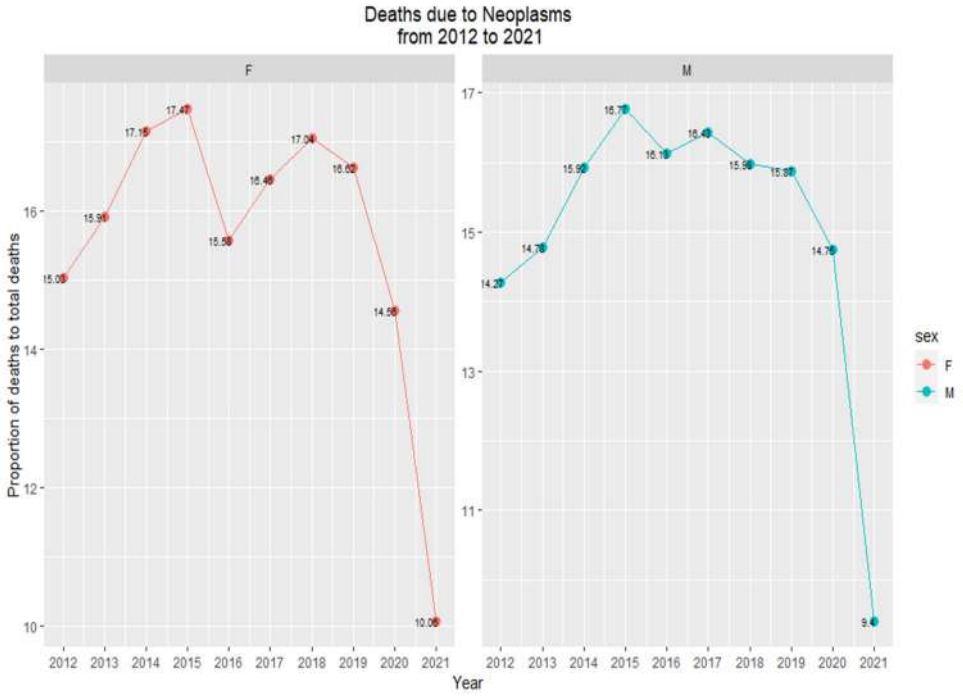
Percentage of deaths due to Endocrine, Nutritional and Metabolic diseases to total deaths has been increasing for both gender from 2016 to 2020. From year 2016 to 2020 there has been an increase from 10.4% to 18% (increase of 7.6 percentage points) for female and an increase from 8.2% to 14.13% (increase of 5.93 percentage points) for male . During year 2021 Diabetes Mellitus accounts for 94.23% of total deaths due to endocrine, nutritional and metabolic diseases . Share of diabetes Mellitus to total deaths in the year 2021 is 10.9%.



Neoplasm



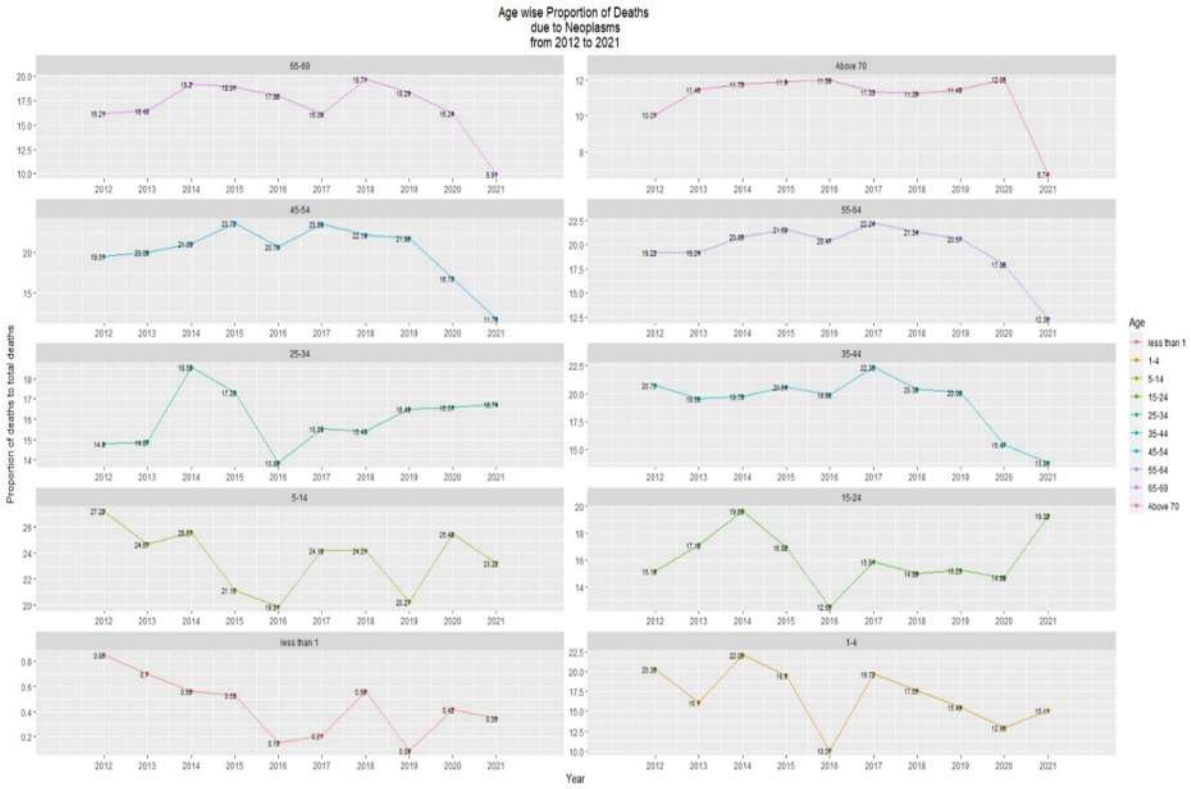
Percentage of deaths due to Neoplasms to total deaths in 15-24 age group increased in 2021 compared to 2020. Percentage share of deaths due to neoplasms in 15-24 age group was 14.68% and 19.32% for years 2020 and 2021 respectively. Neoplasms was the second leading cause of death in age group 25-34 for years 2013 and years 2015 to 2020. In 2021 neoplasms accounted for 16.74% of total deaths in age group 25-34 (12.05% of total male deaths and 23.3% of total female deaths).



During the year 2021 among the 'Neoplasm' deaths, 'Malignant Neoplasm of Digestive Organs' accounts for the highest mortality (29.71%), followed by 'Malignant Neoplasm of Respiratory and Intrathoracic Organs' (17.65%),'Malignant Neoplasms of Lymphoid, Haematopoietic& other related tissue' (17.45%),'Malignant neoplasms of bone, mesothelial and soft tissue, skin and breast' (10.42 %), 'Malignant neoplasms of genitourinary organs' (9.73%) and 'Malignant neoplasms of lip, oral cavity and pharynx'

(5.79 %) and 'Malignant neoplasms of other unspecified sites' (5.47 %) are other major causes.

In 2021, 27.18% of deaths due to Neoplasms occurred in those above 70 years of age , 15.06 % in 65-69 years age group, 28.19% in 55-64 years age group and 13.79% in 45-54, 6.97 % in 35-44 years, 3.22% in 25-34 age group , 2.3% in 15-24 age group, 1.41% in 5-14 age group.



Methodology Used

Data validation.
Data processing for finding proportions and percentage share in each age group
Exploratory and descriptive data aanalysis

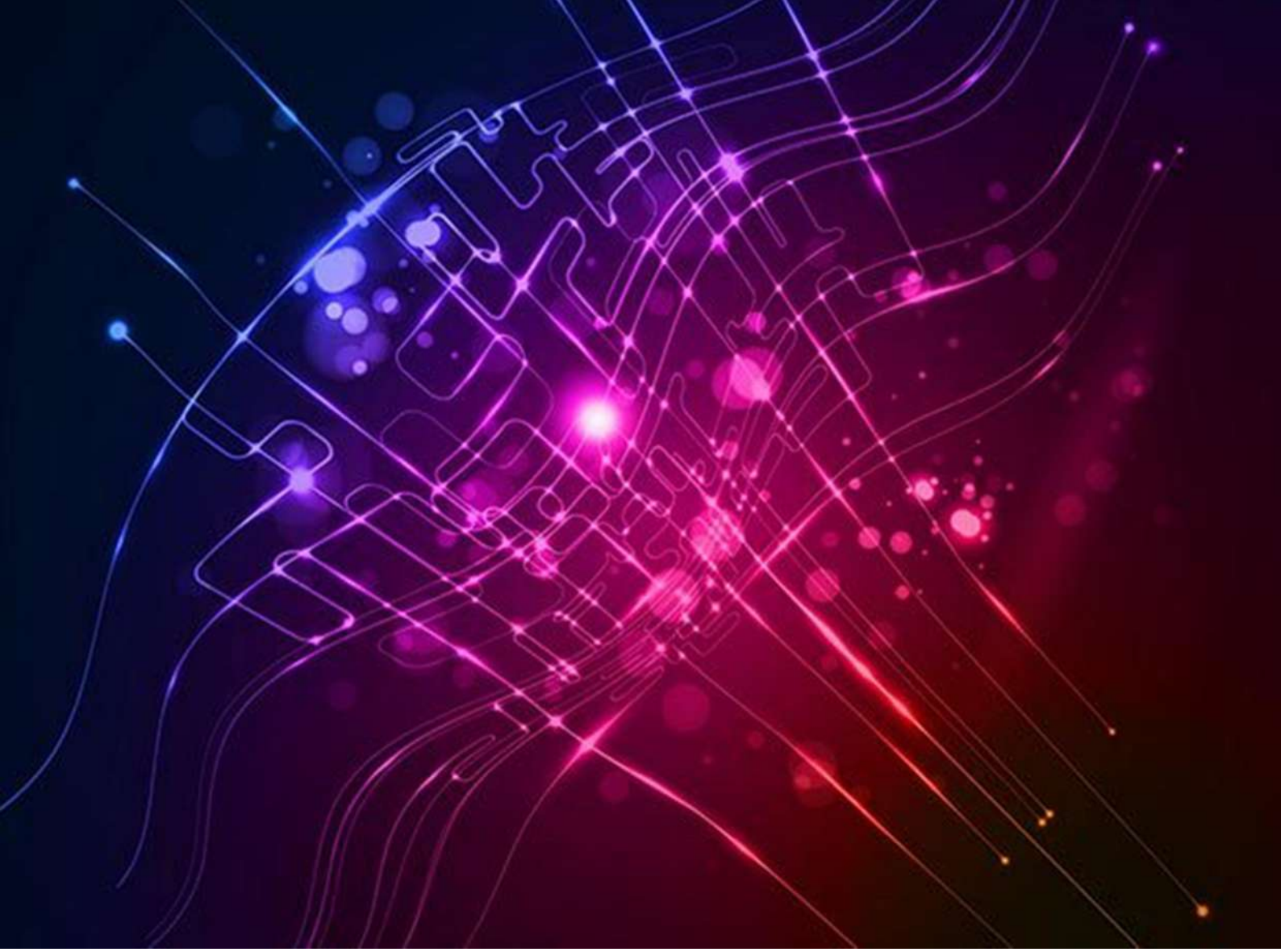
Tools and Libraries used

Project was done in R Markdown.

- Conditional statements and loops were used for validating data.
- Functions in readxl, openxlsx, dplyr, tidyr packages were used for importing, processing and exporting data .
- Exploratory data analysis was done using ggplot2 package.

Conclusion

In the year 2021 infectious and parasitic diseases has become the leading cause of death in Kerala for people aged above 25 years. This shows an increasing share of communicable diseases like viral diseases to total deaths.



Time series Analysis of Gross Value Added of Organized Manufacturing Sector of India and Kerala

Submitted By
Kumari. Baby Sindhu, Statistical Assistant Grade II

Abstract

This project aims to provide an analysis on the time series data of Gross Value Added (GVA) of Organized Manufacturing Sector of India and Kerala using the estimated data of Annual Survey of Industries conducted by Ministry of Statistics and Programme Implementation from 1980-81 to 2019-20. The Box-Jenkins procedure is used to determine the ARIMA (Auto Regressive Integrated Moving Average) model of the time series data. The model is used to forecast the performance of the GVA in the upcoming years.

1. Introduction

Industries plays a pertinent role in the economic growth of the nation. In particular, the contribution of Organised Manufacturing sector are imperative in the country's overall, and especially its economic development. The expansion of a country's manufacturing industries is used to gauge its economic strength. Gross Value Added (GVA) is an economic productivity metric that measures the contribution of the manufacturing sector to the economy. It is the difference of the Gross Output and Gross Input of the manufacturing industries. Analyzing time series data and Forecasting economic model is essential for the policy makers in developing new strategies for further development of economy.

2. Objective

The objective of this project work was to develop forecasting models for predicting the Gross Value Added (GVA) of the Organized Manufacturing Sector of India and Kerala from the historical data. Box-Jenkins methodology has been followed to model the GVA through time series with the ARIMA univariate approach. The purpose of this time series analysis is to extract information about the time series observations and use these information for decision making purpose. The dataset used in this study comprises of GVA of organized manufacturing sector in India and Kerala from 1980-81 to 2019-20 (40 years) and the entire historical data is used to build up a regression model and forecast the future. This provides the consequences and insights of features of the given dataset that changes over time. The analysis has been done using Python Programming.

3. Methodology of Time series analysis

Time series analysis consists of various techniques which is used to reproduce the observations with a mathematical model and hence predicts the estimate of the variable using the relevant historical data. In this study Box Jenkins Auto Regressive Integrated Moving Average (ARIMA) method is adopted for building model and forecasting. A usual time series data comprises of the following components which are the factors affecting the values of the phenomenon:

a. **Secular Trend or Long-Term Movement:** By trend we mean the general tendency of the data to increase or decrease during a long period of time.
b. **Seasonal variation:** These variations in a time series are due to forces which operate in a regular and periodic manner over a span of less than a year.

c. **Cyclic**: The oscillatory movements in time series with period of oscillation more than one year.
d. **Irregular component**: Apart from the regular variation a series may contain random or irregular factors which are not counted by secular trends, seasonal and cyclic variations.

The ARIMA models which is built in this study assumes that the new observation depend purely on the weighted linear combination of past values. It assumes that the series is stationary i.e., mean, variance and autocorrelation are constant over time. If the series is not stationary differencing method is adopted to transform the series in to stationary. The following are the models:

Autoregressive (AR) Model

In an autoregressive mode we forecast the variable of interest using a linear combination of past values of the variable. The term autoregressive indicates that it is a regression of the variable against itself:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t$$

where ε_t is white noise. This is like a multiple regression but with lagged values of y_t as predictors. This as an AR(p) model, an autoregressive model of order p.

Moving Average (MA) Model

Rather than using past values of the forecast variable in a regression, a moving average model uses past forecast errors in a regression-like model:

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$$

This as an MA(q) model, a moving average model of order q (fpp2).

Autoregressive Integrated Moving Average (ARIMA) Model

If we combine differencing with auto regression and a moving average model, we obtain a non seasonal ARIMA model:

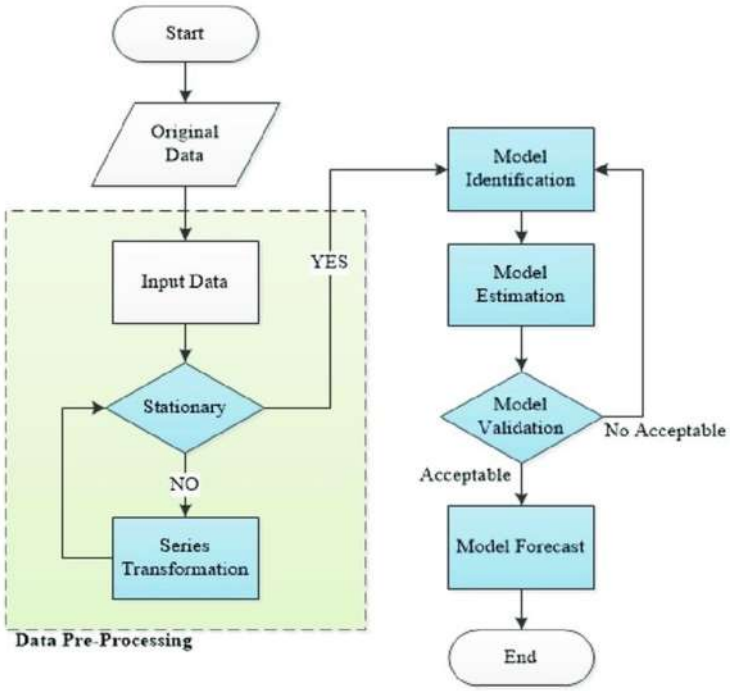
$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t$$

Where y_t' is the differenced series. This ARIMA (p, d, q) model, where p is the number of autoregressive terms, d is the degree of differencing and q is the number of moving average terms.

3.1 Time series Analysis process

Using Box-Jenkins methodology following procedures were followed for identification of model, Parameter estimation/calibration and Model testing/validation.

The basic steps followed is depicted in the flow chart below:



The following procedures is adopted for the analysis:

i. P**lotting time series data**: Time series data was plotted to identify the nature and the trend of the variable.

ii. Stationarity of time series: A time series is said to be stationary if mean, variance and autocorrelation remains constant over time. Graphically, the time series is stationary if the correlogram/ACF (Auto Correlation Function) (a plot of autocorrelation coefficients on the vertical axis with different lags on horizontal axis) dies down fairly quickly and the series is non-stationary if decay is very slow.

Stationarity can also be checked using Augmented Dickey-Fuller (ADF) test where,

H0: **The null hypothesis:** It is a statement about the population that either is believed to be true or is used to put forth an argument unless it can be shown to be incorrect beyond a reasonable doubt.

H1: **The alternative hypothesis:** It is a claim about the population that is contradictory to H_0 and what we conclude when we reject H_0.

#Ho: It is non-stationary
#H1: It is stationary

If p value >0.05, Fail to reject H0
If p value <0.05, Accept H1

Methods for transforming non-stationary data to stationary data

The major methods used for converting non-stationary to stationary for time series modelling are:

- Detrending
- Differencing
- Transformation

In this study differencing method is used for transforming data to stationary data.

iii. Model Identification

The model identification is done using Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF).

Auto Correlation Function (ACF)/Correlogram

ACF indicates the memory of the stochastic process. It indicates the dependence of the value of the variable on the previous value. It is a plot of autocorrelation coefficients on the vertical axis with different lags on horizontal axis.

Partial Correlation Coefficient Function (PACF)

PACF indicates the dependence of value of the variable on the previous values when the dependence on all other variables are removed.

Auto Regressive Model

AR(p) models can be identified as follows:

i. Exponential or sinusoidal decay in ACF.
ii. PACF cuts off after a few say, 'p' lags.

Moving Average Model

MA(q) models can be identified as follows:

i. Decaying of PACF (either exponential or in a dampened sine wave)
ii. ACF cuts off after a few say, 'q' lags.

Auto Regressive Integrated Moving Average Model

If both ACF and PACF decay gradually ARIMA model is used. In ARIMA (p,d,q) parameters p, q and d are non-negative integers, p is the order (number of time lags) of the autoregressive model, d is the degree of differencing (number of times the data have had past values subtracted), and q is the order of the moving average model.

iv. Model estimation

Akaike Information criterion (AIC) and Bayesian Information Criterion (BIC) are used to test how well the model fit the data. The best model is identified based on high value of AIC and low value of BIC. The AIC and BIC is described mathematically as follows:

$$AIC = log\ \hat{\sigma}_k^2 + \frac{n+2k}{n}$$

$$BIC = \log \hat{\sigma}_k^2 + \frac{k \log n}{n}$$

where $\hat{\sigma}_k^2$ is the maximum likelihood estimate of variance, n is the number of dates, and, k is the number of parameters of the model.

v. Model Validation

Residual Analysis is done to evaluate the accuracy of the model. The accuracy of the model can be assessed by measuring the difference between predicted value and actual value. In ARIMA the above difference is the error term and it behave as white noise, i.e, zero mean, a constant variance and no correlation. The error term should also be normally distributed.

Box-Ljung test

Box-Ljung test is a diagnostic tool applied to residuals to test the lack of fit of time series. It examines the autocorrelations of the residuals. The null hypothesis is that series is uncorrelated. If p value >0.05 then the residuals are uncorrelated.

Shapiro –Wilk test for Normality check

The Shapiro-Wilk test for normality is a diagnostic tool of statistics to assess the normality of the data. The null hypothesis is taken as H0: The residuals are normally distributed. If p value >0.05 then normality assumption is satisfied.

vi. Forecasting

After the validation procedure the model can be used for forecasting the future values.

vii. Measure of Accuracy

To measure the accuracy of the forecasted data **Mean Absolute Percentage Error (MAPE)** is used.

The **mean absolute percentage error (MAPE)** is commonly used to measure the predictive accuracy of models. It is calculated as:

MAPE = (1/n) * Σ(|actual – prediction| / |actual|) * 100

where:

- **Σ** – a symbol that means "sum"
- **n** – sample size
- **actual** – the actual data value
- **prediction** – the predicted data value

3.2 Time series Analysis

The time series analysis is conducted on the data comprising of the Gross Value Added (GVA) of industries of India and Kerala in the organised manufacturing sector from 1980-81 to 2019-20. The GVA is expressed in Rupee lakhs. The data has been procured from the official site of Ministry of Statistics and Programme Implementation (MOSPI).

3.2.1 Time Series Analysis for predicting the GVA of Organised manufacturing sector of India

1. Plotting Time series Data

The trend of time series data of GVA of India in 40 years is shown in *Figure 1.*

Figure 1

2. Decomposition of time series data

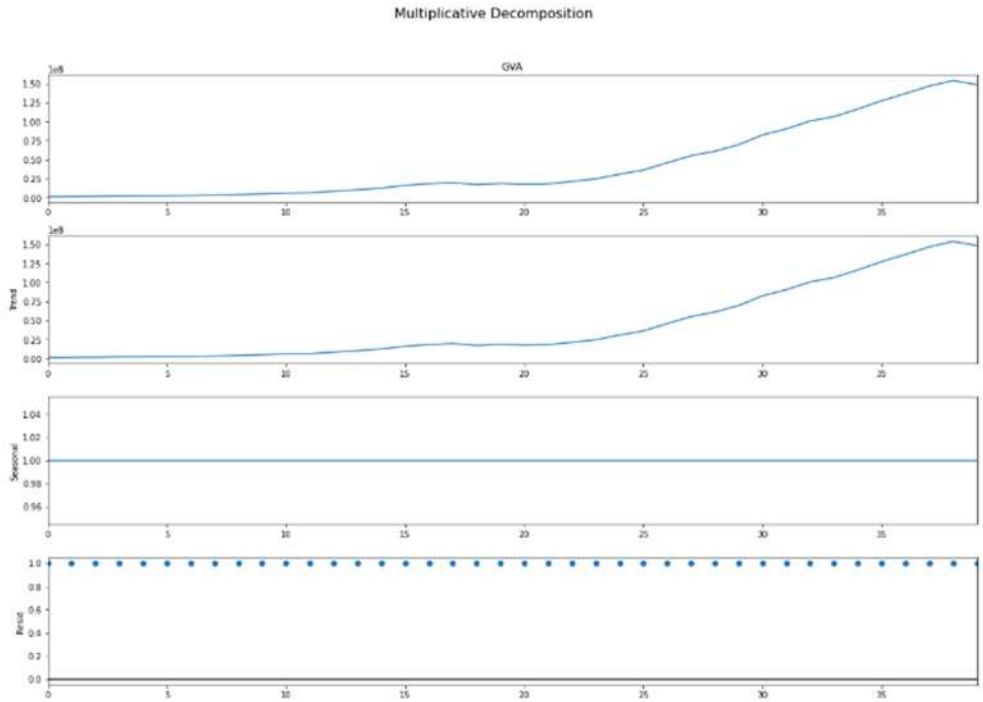


Figure 2

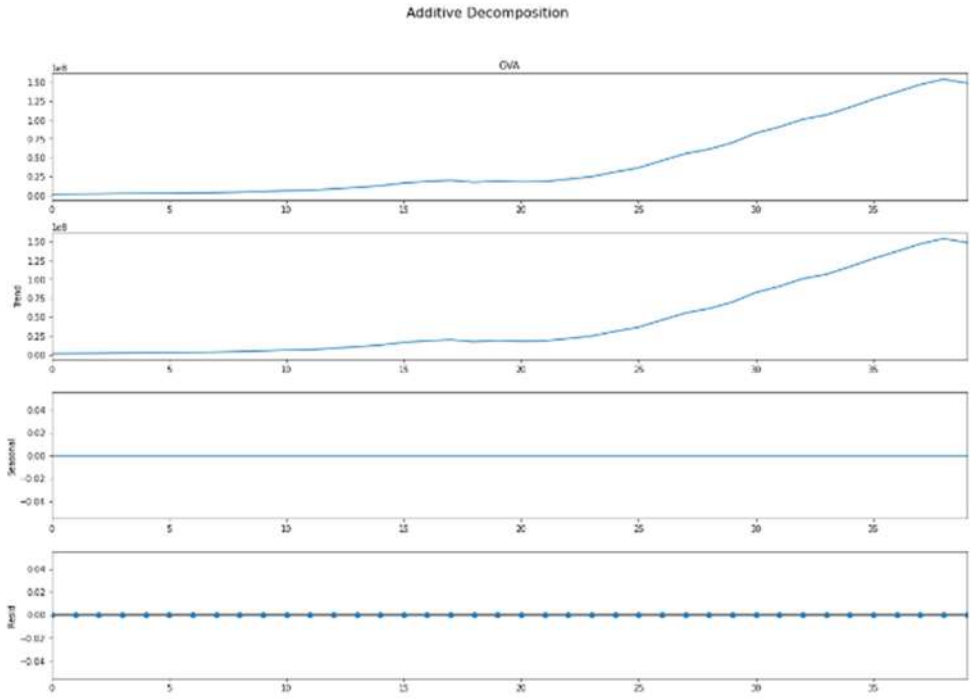


Figure 3

3. Training and Test Data

The whole data set is split into training and test data. The data from 1980-81 to 2015-16 is selected as train data and 2016-17 to 2019-20 as test data. The whole ARIMA methodology is performed in the train data to build a model and further the model is tested in the test data and after the validation and accuracy checking the model is used to forecast the future data.

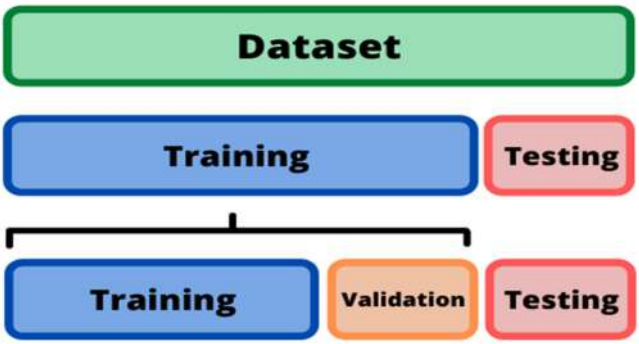


Figure 4

4. Stationarity Check

Auto Correlation function

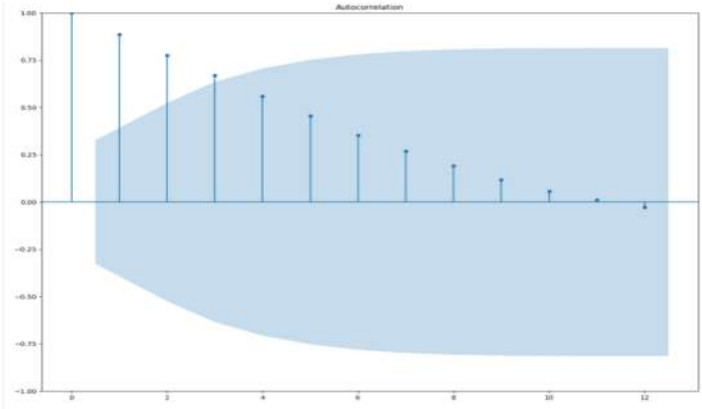


Figure 5

Partial Autocorrelation Function

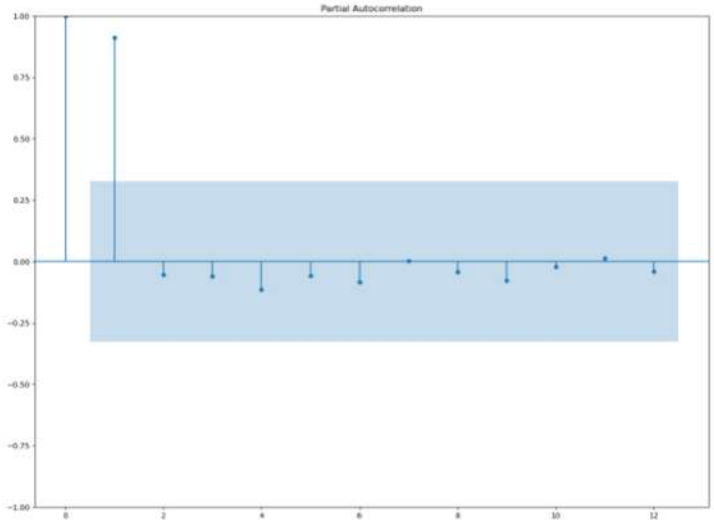


Figure 6

Dickey-Fuller Test

```
Observations of Dickey-fuller test
Test Statistic                  -1.881921
p-value                          0.340613
#lags used                       2.000000
number of observations used     37.000000
critical value (1%)             -3.620918
critical value (5%)             -2.943539
critical value (10%)            -2.610400
dtype: float64
```

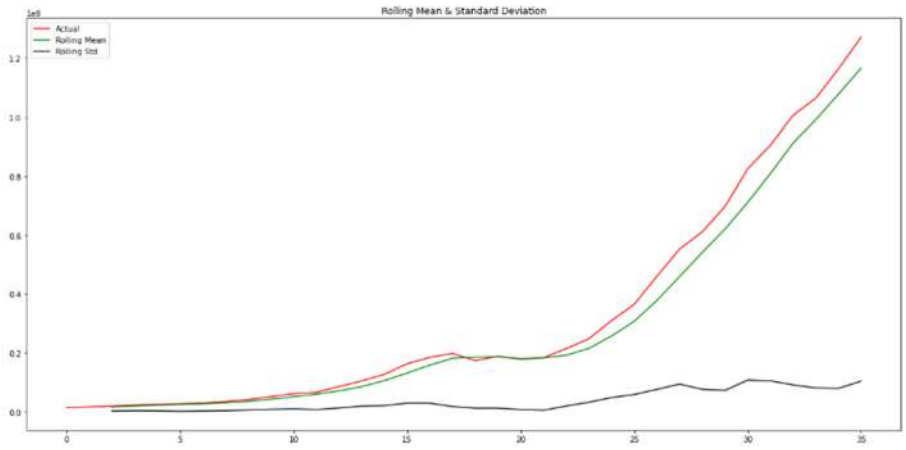


Figure 7

The ACF, PACF curve, Rolling mean and Standard deviation curve and Dickey-Fuller Test shows that the data is non-Stationary. Further differencing method was adopted to transform the data in to stationary data.

Differencing for transforming the data into stationary data.

The data was undergone differencing twice to transform to stationary data. The Dickey- Fuller test result after differencing is as follows:

```
Observations of Dickey-fuller test
Test Statistic                -9.214863e+00
p-value                        1.845513e-15
#lags used                     0.000000e+00
number of observations used    3.300000e+01
critical value (1%)           -3.646135e+00
critical value (5%)           -2.954127e+00
critical value (10%)          -2.615968e+00
dtype: float64
```

The results show that p value <0.05. So the data is stationary.

Distribution of Data after Second Differencing

Figure 8

Rolling Mean & Standard Deviation

Figure 9

5. Identifying the model using ACF and PACF

As per the ACF and PACF curve the model identified was ARIMA (1, 2, 1)

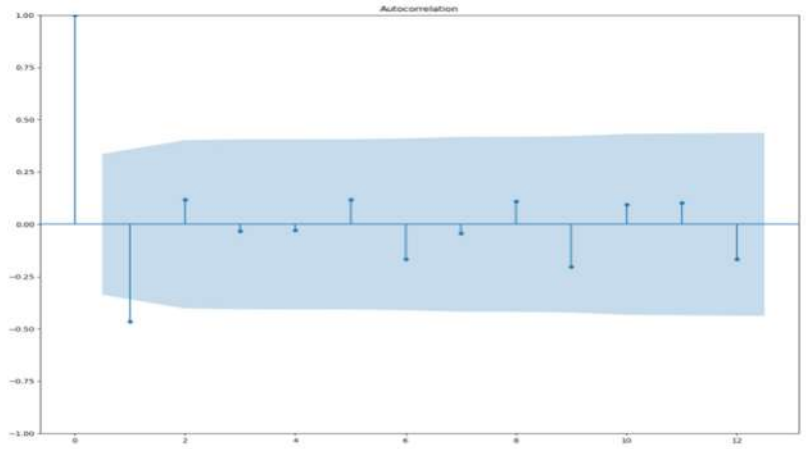


Figure 10

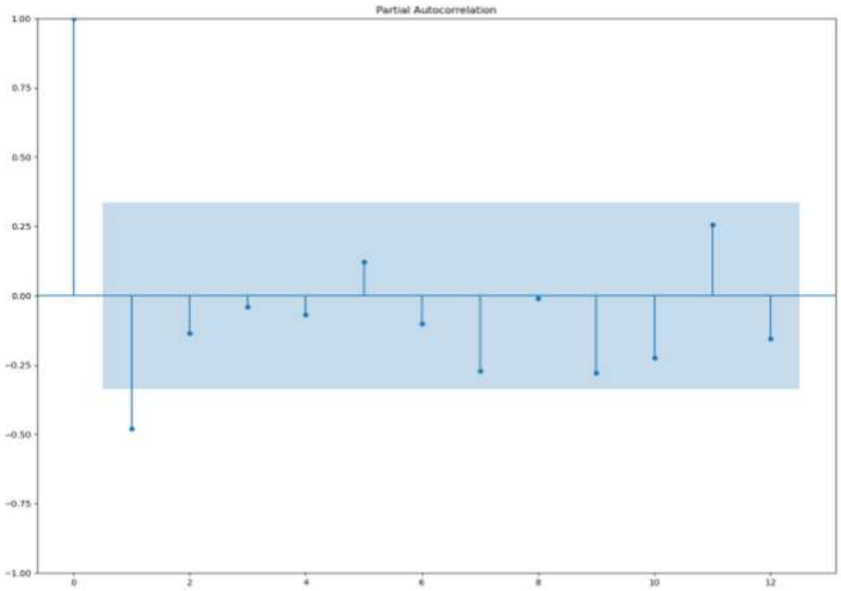


Figure 11

6. Plotting actual and predicted value of trained data

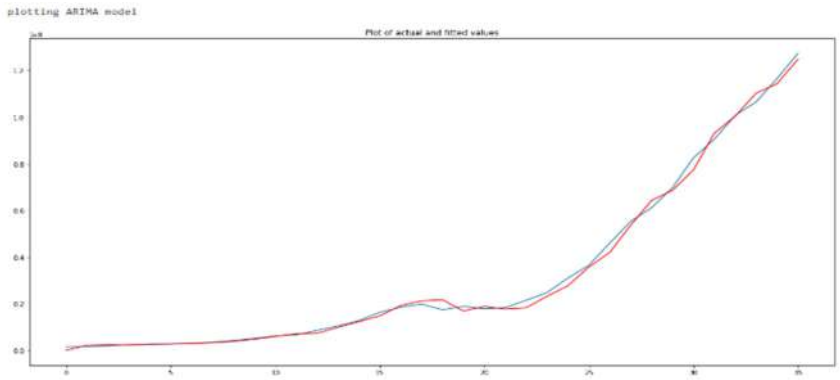


Figure 12

Dep. Variable:	GVA	No. Observations:	36
Model:	ARIMA(1, 2, 1)	Log Likelihood	-542.646
Date:	Thu, 27 Apr 2023	AIC	1091.292
Time:	09:15:07	BIC	1095.871
Sample:	0	HQIC	1092.854
	- 36		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.2810	0.332	-0.847	0.397	-0.931	0.369
ma.L1	-0.2011	0.330	-0.610	0.542	-0.847	0.445
sigma2	4.795e+12	2.18e-14	2.2e+26	0.000	4.8e+12	4.8e+12

Ljung-Box (L1) (Q):	0.10	Jarque-Bera (JB):	0.52
Prob(Q):	0.75	Prob(JB):	0.77
Heteroskedasticity (H):	37.42	Skew:	-0.19
Prob(H) (two-sided):	0.00	Kurtosis:	3.47

7.Model Validation using Box-Ljung Test

The test shows that p value >0.05 and hence the residuals are uncorrelated.

8. In sample forecast of GVA

Year	Actual GVA (in Rupee lakhs)	Predicted GVA (in Rupee Lakhs)
2016-17	136805409	137428200
2017-18	146697043	147741200
2018-19	153801928	157994500
2019-20	148574512	168264500

9. Residual Analysis

Shapiro –Wilk Test for Normality

The test shows that p value >0.05 therefore the residuals are normally distributed.

```
ShapiroResult(statistic=0.9682322144508362, pvalue=0.3792402446269989)
```

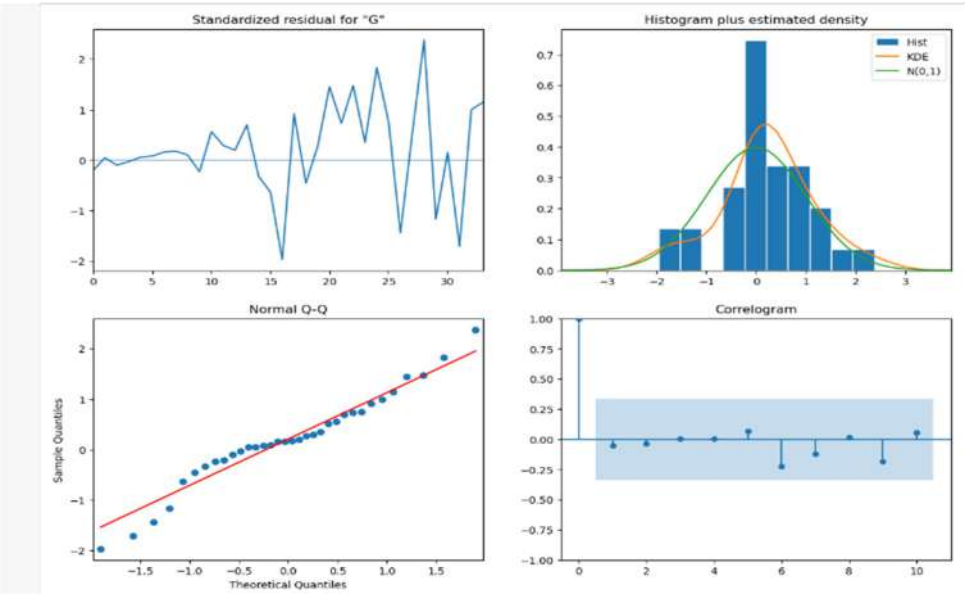


Figure 13

10. Measure of Accuracy

To measure the accuracy of the forecasted data **Mean Absolute Percentage Error (MAPE)** is used. The MAPE value is as follows which shows 4% error in predicted and actual data.

```
4.286453980737848
```

11. Out Sample Forecasting

Year	Predicted GVA (in Rupee Lakhs)
2020-21	178529800
2021-22	188796500
2022-23	199062800
2023-24	209329200

3.2.2 Time Series Analysis for predicting the GVA of Organised manufacturing sector of Kerala

1. Plotting the GVA of Kerala

Figure 14

2. Decomposition of time series data

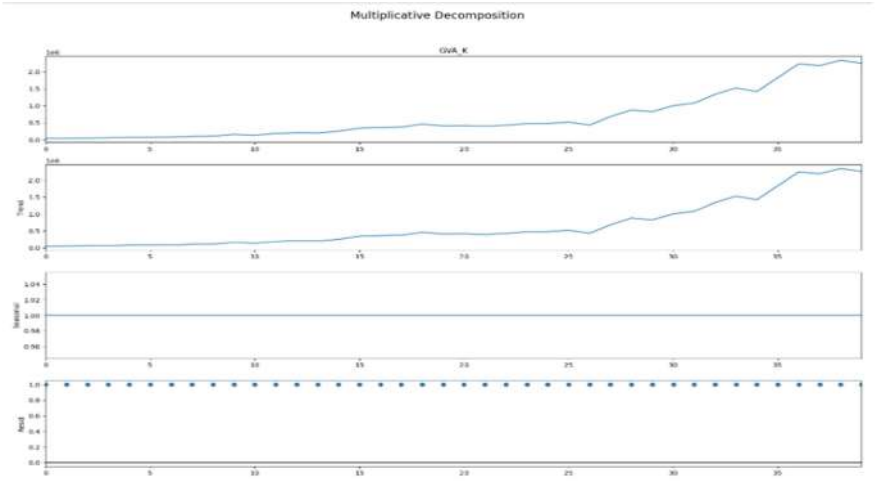


Figure 15

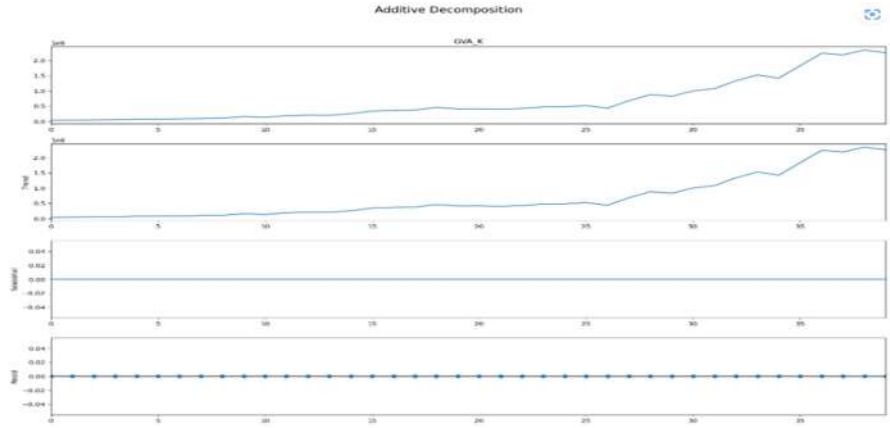


Figure 16

3. Trained Data and Test Data

The data from 1980-81 to 2015-16 is selected as train data and 2016-17 to 2019-20 as test data.

Figure 17

12. Stationarity check

Auto Correlation function

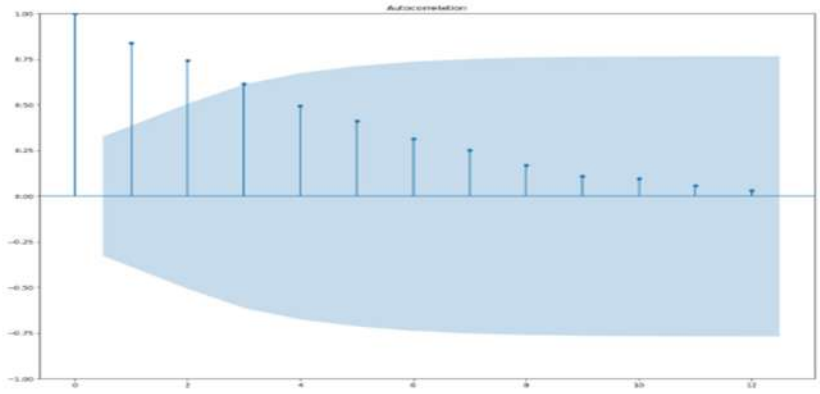


Figure 18

Partial Autocorrelation Function

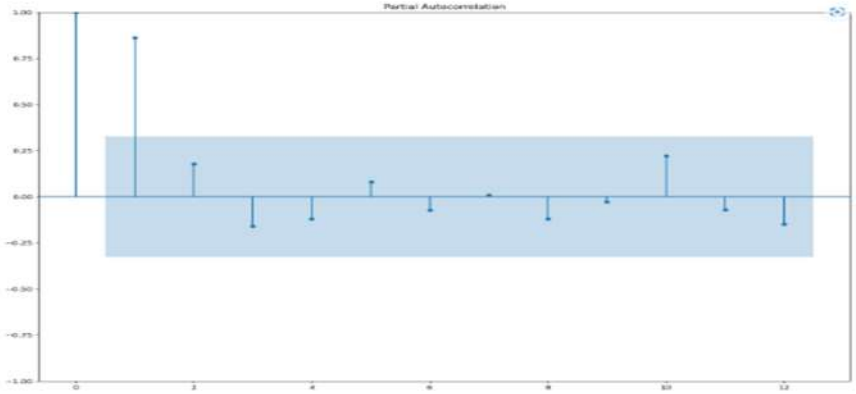


Figure 19

Dickey-Fuller Test

```
Observations of Dickey-fuller test
Test Statistic                  0.868662
p-value                         0.992667
#lags used                      7.000000
number of observations used    32.000000
critical value (1%)            -3.653520
critical value (5%)            -2.957219
critical value (10%)           -2.617588
dtype: float64
```

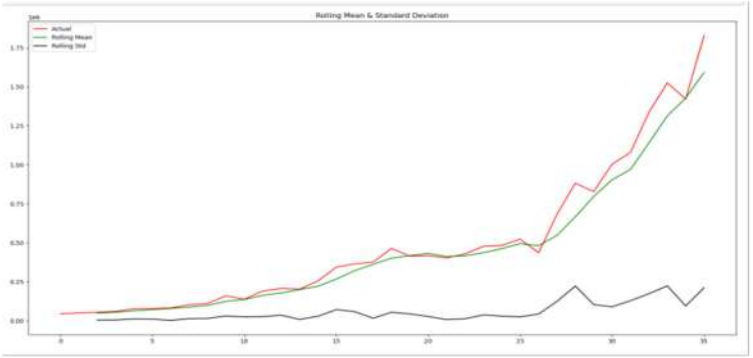


Figure 20

The ACF, PACF curve, Rolling mean and Standard deviation curve and Dickey-Fuller Test shows that the data is non-Stationary. Further differencing method was adopted to transform the data in to stationary data.

Differencing for transforming the data into stationary data.

The data was undergone differencing thrice to transform to stationary data. The Dickey- Fuller test result after differencing is as follows:

```
Observations of Dickey-fuller test
Test Statistic                  -4.783848
p-value                          0.000058
#lags used                       5.000000
number of observations used     27.000000
critical value (1%)             -3.699608
critical value (5%)             -2.976430
critical value (10%)            -2.627601
dtype: float64
```

The results show that p value <0.05. So the data is stationary.

Distribution of Data after Third Differencing

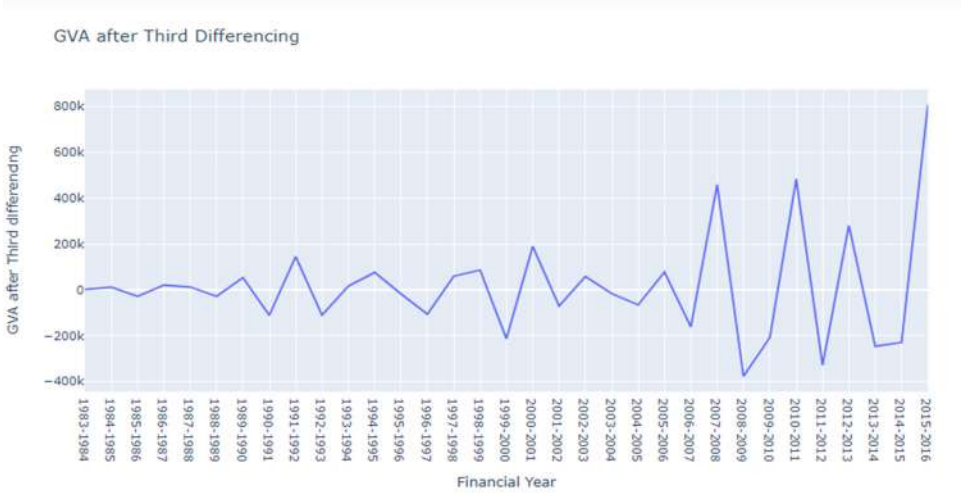


Figure 21

Rolling Mean & Standard Deviation

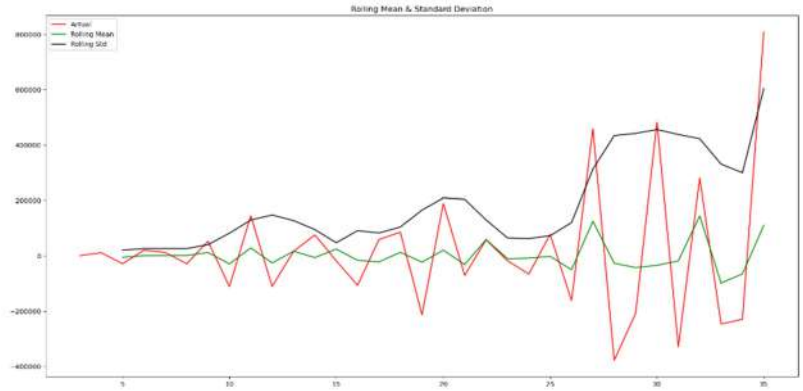


Figure 22

13. Identifying the model using ACF and PACF

As per the ACF and PACF curve the model identified was ARIMA (4, 3, 1)

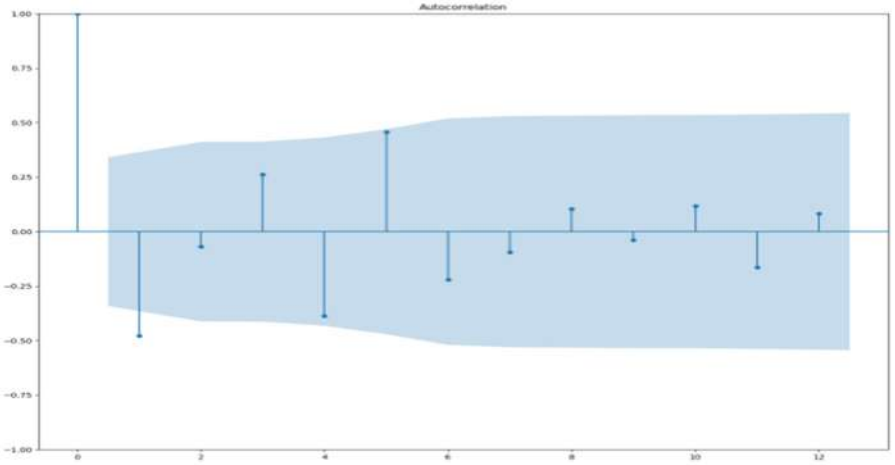


Figure 23

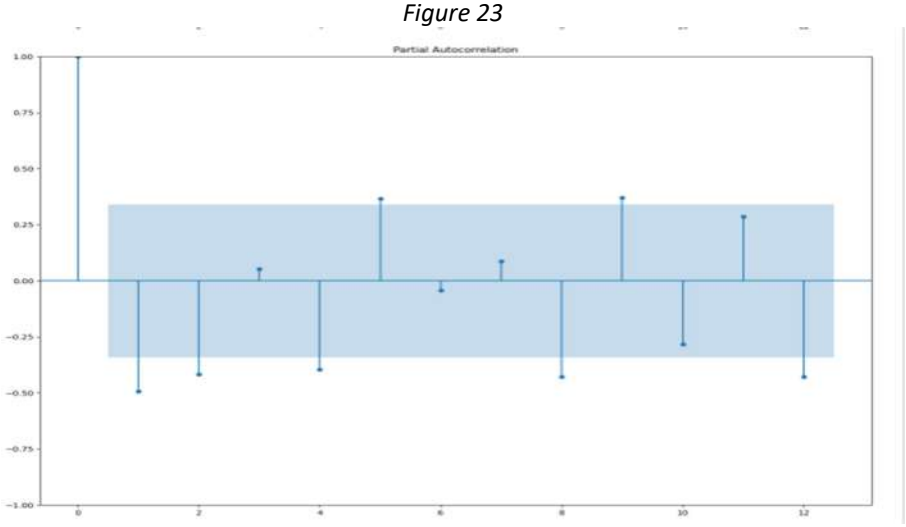


Figure 24

14. Plotting actual and predicted value of trained data

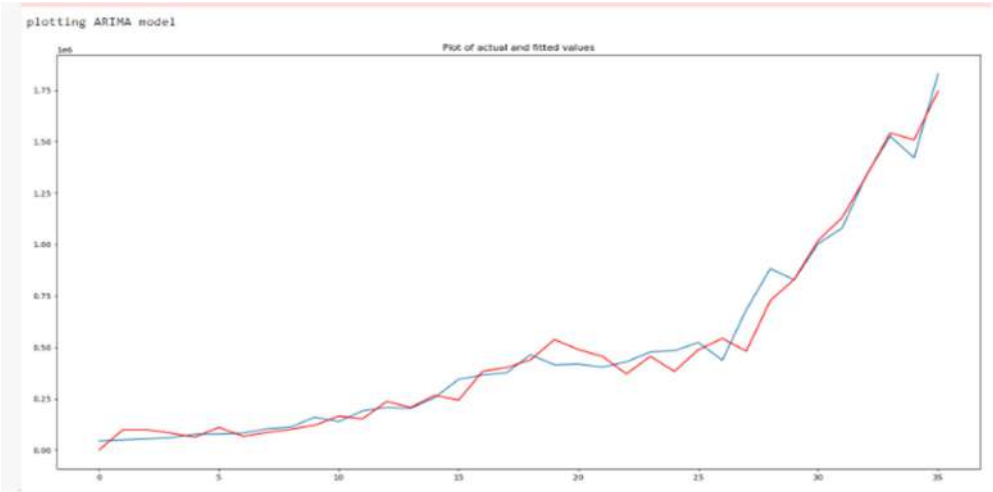


Figure 25

Dep. Variable:	GVA_K	No. Observations:	36
Model:	ARIMA(4, 3, 1)	Log Likelihood	-415.511
Date:	Fri, 28 Apr 2023	AIC	843.022
Time:	16:26:12	BIC	852.001
Sample:	0	HQIC	846.043
	- 36		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-1.3520	0.180	-7.524	0.000	-1.704	-1.000
ar.L2	-1.5857	0.286	-5.547	0.000	-2.146	-1.025
ar.L3	-1.1387	0.336	-3.384	0.001	-1.798	-0.479
ar.L4	-0.8607	0.210	-4.097	0.000	-1.272	-0.449
ma.L1	-0.6813	0.328	-2.080	0.038	-1.323	-0.039
sigma2	6.129e+09	4.47e-11	1.37e+20	0.000	6.13e+09	6.13e+09

Ljung-Box (L1) (Q):	0.06	Jarque-Bera (JB):	4.38
Prob(Q):	0.81	Prob(JB):	0.11
Heteroskedasticity (H):	14.85	Skew:	0.73
Prob(H) (two-sided):	0.00	Kurtosis:	4.04

15. Model Validation using Box-Ljung Test

The test shows that p value >0.05 and hence the residuals are uncorrelated.

	lb_stat	lb_pvalue
1	0.093441	0.759848

16. In sample forecast of GVA

Year	Actual GVA (in Rupee lakhs)	Predicted GVA (in Rupee Lakhs)
2016-17	2236188	2004238
2017-18	2183551	2146227
2018-19	2337876	2444895
2019-20	2252678	2479974

17. Residual Analysis

Shapiro –Wilk Test for Normality

The test shows that p value >0.05 therefore the residuals are normally distributed.

```
ShapiroResult(statistic=0.9545350074768066, pvalue=0.14523519575595856)
```

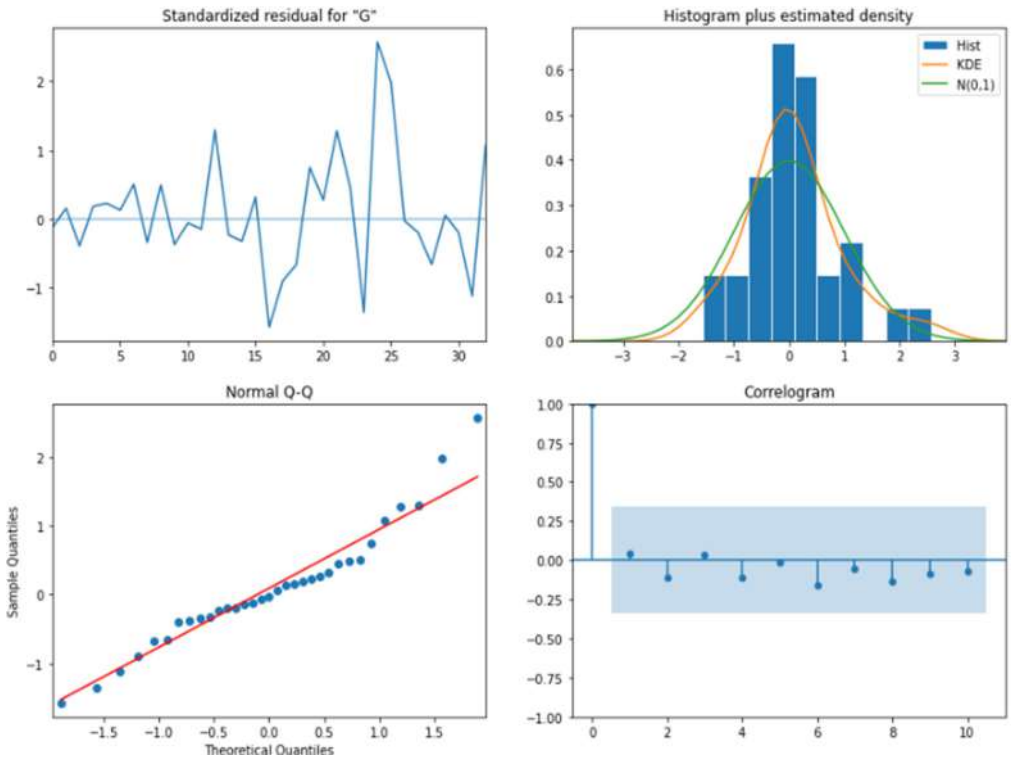


Figure 26

18. Measure of Accuracy

To measure the accuracy of the forecasted data **Mean Absolute Percentage Error (MAPE)** is used. The MAPE value is as follows which shows 6.8% error in predicted and actual data.

```
6.687375791304937
```

19. Out Sample Forecasting

Year	Predicted GVA (in Rupee Lakhs)
2020-21	2933417
2021-22	3160725
2022-23	3267800
2023-24	3718621

4. Dataset used

Gross Value Added (GVA) of Organized Manufacturing sector of Kerala from 1980-81 to 2019-20

Financial Year	Gross Value Added (GVA) of Kerala (in Rupee lakhs)
1980-1981	44453
1981-1982	49663
1982-1983	54563
1983-1984	59976
1984-1985	77272
1985-1986	77841

1986-1987	82298
1987-1988	102975
1988-1989	111358
1989-1990	160235
1990-1991	138461
1991-1992	190971
1992-1993	207178
1993-1994	203119
1994-1995	254675
1995-1996	344068
1996-1997	364840
1997-1998	376065
1998-1999	463724
1999-2000	414630
2000-2001	418162
2001-2002	403543
2002-2003	429000
2003-2004	477401
2004-2005	482950
2005-2006	523307
2006-2007	436893
2007-2008	682240
2008-2009	882107
2009-2010	828506
2010-2011	1004010
2011-2012	1079692
2012-2013	1335966
2013-2014	1526293
2014-2015	1421515
2015-2016	1829940
2016-2017	2236188
2017-2018	2183551
2018-2019	2337876
2019-2020	2252678

Source: www.mospi.gov.in

Gross Value Added (GVA) of Organized Manufacturing sector of India from 1980-81 to 2019-20

Year	Gross Value Added (GVA) in Rupee lakhs)
1980-1981	1407076
1981-1982	1668146
1982-1983	1914118
1983-1984	2352049
1984-1985	2494163
1985-1986	2696866
1986-1987	3019912
1987-1988	3458579
1988-1989	4176048

1989-1990	5132650
1990-1991	6157753
1991-1992	6616782
1992-1993	8567098
1993-1994	10488907
1994-1995	12719229
1995-1996	16302305
1996-1997	18489354
1997-1998	19823745
1998-1999	17372692
1999-2000	18857370
2000-2001	17835033
2001-2002	18322914
2002-2003	21437562
2003-2004	24777726
2004-2005	30962009
2005-2006	36469705
2006-2007	46018006
2007-2008	55275622
2008-2009	61131148
2009-2010	69718259
2010-2011	82513335
2011-2012	90520894
2012-2013	100727950
2013-2014	106511164
2014-2015	116470249
2015-2016	127327968
2016-2017	136805049
2017-2018	146697043
2018-2019	153801928
2019-2020	148574512

Source: www.mospi.gov.in

5. Tools and Libraries

The analysis was done using Python Programming. The technique used for time series analysis was ARIMA. The libraries used were numpy, pandas, matplotlib, plotly.graph_objects, seaborn, sklearn, statsmodels.tsa.stattools, statsmodels.graphics. tsaplots, statsmodels.tsa.arima.model, scipy.stats, statsmodels.api.

6. Result with inference

The GVA of Organized Manufacturing Sector of India and Kerala for 40 years from 1980-81 to 2019-20 were analysed using ARIMA model of time series analysis. The model fitted for data of India and Kerala were used for in-sample and out-sample forecast. The MAPE was used to measure the accuracy of the results. The following table shows the results of the analysis.

In-sample Forecast of GVA of India

Year	Actual GVA (in Rupee lakhs)	Predicted GVA (in Rupee Lakhs)
2016-17	136805409	137428200
2017-18	146697043	147741200
2018-19	153801928	157994500
2019-20	148574512	168264500

Out-sample Forecast of GVA of India

Year	Predicted GVA (in Rupee Lakhs)
2020-21	178529800
2021-22	188796500
2022-23	199062800
2023-24	209329200

In-sample forecast of GVA of Kerala

Year	Actual GVA (in Rupee lakhs)	Predicted GVA (in Rupee Lakhs)
2016-17	2236188	2004238
2017-18	2183551	2146227
2018-19	2337876	2444895
2019-20	2252678	2479974

Out-sample Forecast of GVA of Kerala

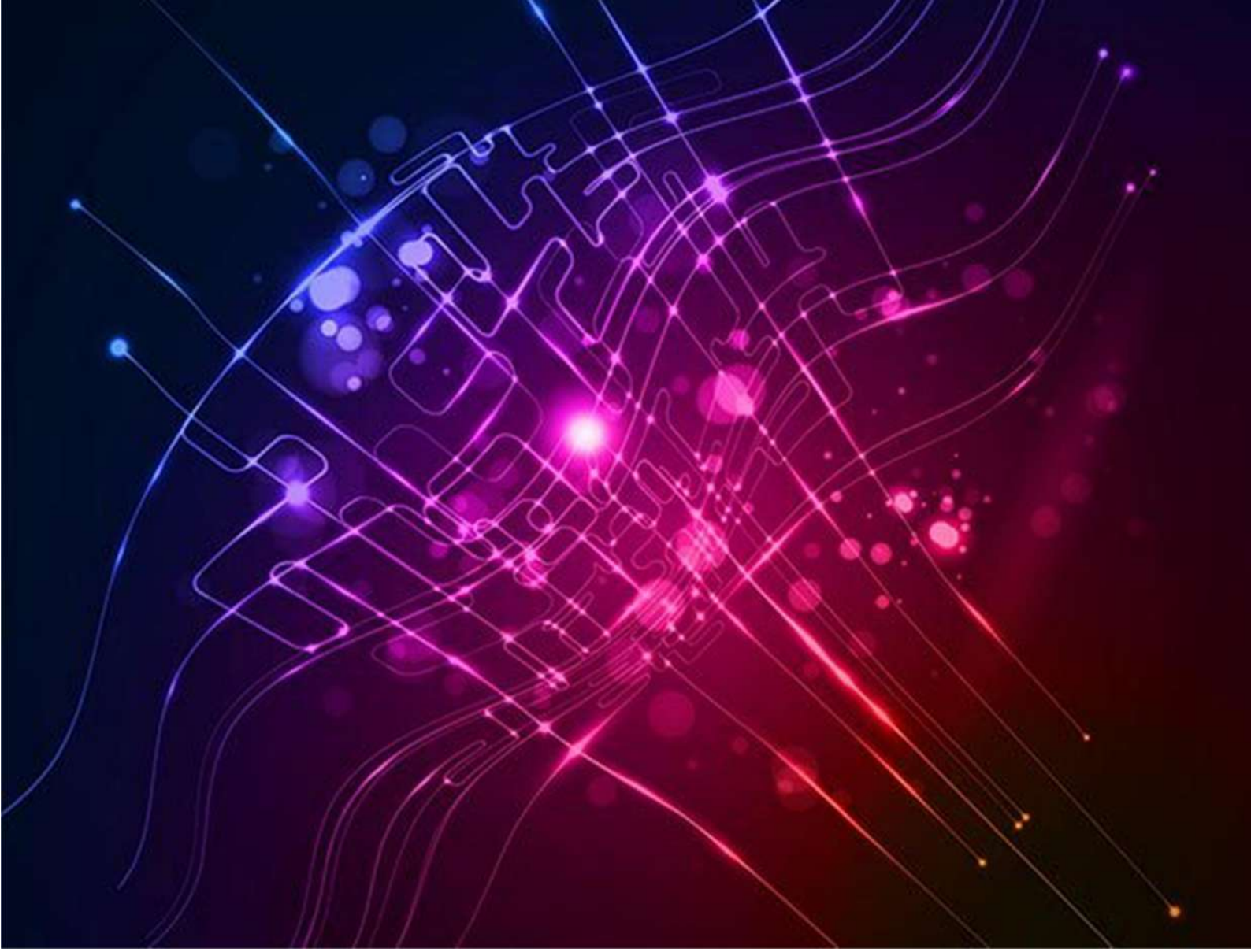
Year	Predicted GVA (in Rupee Lakhs)
2020-21	2933417
2021-22	3160725
2022-23	3267800
2023-24	3718621

Conclusion

This study was conducted to analyse the trend of GVA of India and Kerala over a period of 40years. ARIMA method used is applicable only if the data is stationary (mean, varaiance and autocorrelation remains constant through time). The study discusses the changes in the GVA for the period of 1980-81 to 2019-20. The results of the study provides useful information for identifying the trend of GVA.

References:

1. *Fundamentals of Applied Statistics, S.C.Gupta, V.K.Kapoor, Sultan Chand & Sons Publication.*
2. *Barzola_monteses, Monica Mite-Leon, Mayken Espinoza-Andaluz, Juan Gomez-Romero and Wald Fajardo, et al.(2019).Time Series Analysis for Predicting Hydroelectric Power Production: The Ecuador Case.*
3. *Habib Ahmed Elsayir, et al (2018), An Econometric Time Series GDP Model Analysis: Statistical Evidences and Investigations.*



Prediction on Live Births

Submitted By
Sri. Prasanth B.R., Computer Supervisor

Introduction

The live birth data for the Indian state of Kerala from 1962 to 2020 represents the number of live births that occurred in the state each year during this period. This dataset can be analyzed to identify trends and patterns in the number of live births over time and to make predictions about future live birth rates in the state.

The term live birth refers to the delivery of a baby that is born with a heartbeat and breathing, indicating that the baby has viable life outside the womb.

Objectives

The objective of this analysis is to predict the number of live births in Kerala for future years using four different methods: Simple linear regression, Simple linear regression with log, ARIMA, and LSTM. Simple linear regression is a commonly used statistical method that can help us understand the relationship between two variables, in this case, the year and the number of live births. By fitting a linear regression model to the data, we can estimate how much the number of live births changes for each additional year. In the Simple linear regression with log method, we will take the logarithm of the number of live births before fitting the regression model. This technique can help to stabilize the variance of the data and improve the accuracy of the model.

ARIMA stands for Autoregressive Integrated Moving Average, which is a time series model that can help us to predict future values based on past observations. This method takes into account both the trend and the seasonality of the data, which can help to improve the accuracy of the predictions.

Finally, LSTM stands for Long Short-Term Memory, which is a type of neural network model that is particularly suited to time series data. LSTM models can capture long-term dependencies in the data and are often used in applications such as natural language processing and speech recognition.

By comparing the performance of these four methods, we can identify which technique is most effective for predicting the number of live births in Kerala in the future. The results of this analysis could be useful for policymakers and healthcare professionals in Kerala who need to plan for future resource allocation and healthcare infrastructure based on the expected number of live births in the state.

Methodolody and method used

The objective of this analysis is to predict the live birth rates for the Indian state of Kerala from 1962 to 2020 using four different models: Simple Linear Regression, Simple Linear Regression with Logarithmic Transformation, Autoregressive Integrated Moving Average (ARIMA), and Long Short-Term Memory (LSTM) neural networks.

Methodology:

1. **Data Cleaning:** The first step is to clean the dataset, which involves checking for any missing or erroneous values and correcting them if necessary. The dataset appears to be complete, so no further cleaning is required.
2. **Exploratory Data Analysis:** The second step is to conduct exploratory data analysis to identify any trends or patterns in the data. This can be done by creating visualizations such

as line plots, scatter plots, and histograms. This step helps in understanding the characteristics of the data and identifying any outliers or anomalies that may need to be removed.

3. Feature Selection: The third step is to select the relevant features for the model. In this case, the only feature is the year, as it is the only independent variable in the dataset.

4. Model Selection and Training: The fourth step is to select and train the models. The four models selected for this analysis are Simple Linear Regression, Simple Linear Regression with Logarithmic Transformation, ARIMA, and LSTM neural networks.

a. **Simple Linear Regression:** This model assumes a linear relationship between the dependent variable (live birth rates) and the independent variable (year). The model is trained using the historical data and used to make predictions for future years.

b. Simple Linear Regression with Logarithmic Transformation: This model assumes that the relationship between the dependent variable and independent variable is not linear, but can be transformed using a logarithmic function to make it linear. The model is trained on the transformed data and used to make predictions for future years.

c. ARIMA: This model is a time series analysis technique that takes into account the autocorrelation and stationarity of the data. It involves identifying the order of differencing, the order of autoregression, and the order of moving average. The model is trained using historical data and used to make predictions for future years.

d. **LSTM:** This model is a type of recurrent neural network that can handle sequence data. It is well-suited for time series analysis as it can capture long-term dependencies in the data. The model is trained using historical data and used to make predictions for future years.

5. Model Evaluation: The final step is to evaluate the performance of the models. This can be done using metrics such as mean squared error, root mean squared error, and mean absolute error. The model with the lowest error is selected as the best model.

Overall, this methodology involves cleaning the data, conducting exploratory data analysis, selecting relevant features, training and evaluating the models, and selecting the best model based on the evaluation metrics.

Dataset

Year	LiveBirth		Year	LiveBirth		Year	LiveBirth
1962	395620		1982	560859		2002	581925
1963	398244		1983	546192		2003	558369
1964	424244		1984	548998		2004	563153
1965	415109		1985	587966		2005	559082
1966	428207		1986	567210		2006	537915
1967	402151		1987	570283		2007	545154
1968	415132		1988	574312		2008	535738
1969	378716		1989	564066		2009	542154
1970	305435		1990	568981		2010	546964
1971	398159		1991	557697		2011	560268
1972	468076		1992	523455		2012	550411

1973	466445		1993	530470		2013	536352
1974	471424		1994	533209		2014	534458
1975	530425		1995	557787		2015	516013
1976	529522		1996	586253		2016	496292
1977	494888		1997	607727		2017	503597
1978	498357		1998	591508		2018	488174
1979	527839		1999	596948		2019	480113
1980	532422		2000	593724		2020	446891
1981	535746		2001	579063			

Tools and libraries used

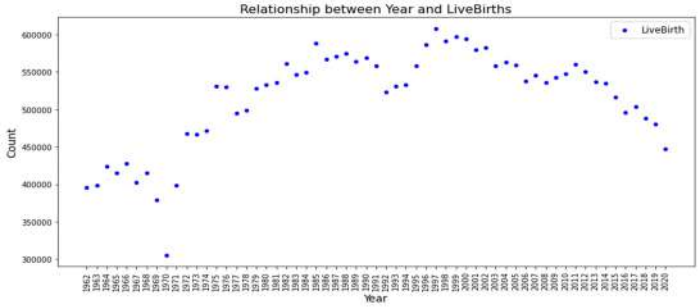
To predict live birth data for Kerala from 1962 to 2020 using different techniques, you can use several Python libraries and tools. Here are some common ones:

1. Pandas: Pandas is a Python library used for data manipulation and analysis. You can use it to load and preprocess the live birth data for Kerala.
2. NumPy: NumPy is a library used for numerical computing in Python. It provides support for large, multi-dimensional arrays and matrices, as well as a variety of mathematical functions. You can use it to perform mathematical operations on the live birth data.
3. Matplotlib: Matplotlib is a data visualization library that allows you to create various types of charts and graphs. You can use it to visualize the trends and patterns in the live birth data.
4. StatsModels: StatsModels is a Python library used for statistical modeling and analysis. You can use it to perform linear regression analysis and time series analysis on the live birth data.
5. Scikit-learn: Scikit-learn is a machine learning library that provides various algorithms for data analysis and modeling. You can use it to perform machine learning techniques such as linear regression and LSTM on the live birth data.

Once you have loaded and preprocessed the live birth data using Pandas, you can use StatsModels to perform simple linear regression and ARIMA analysis on the data. You can use Scikit-learn to perform simple linear regression with log and LSTM analysis on the data. You can use Matplotlib to visualize the predictions and evaluate the performance of each model.

Result with inference

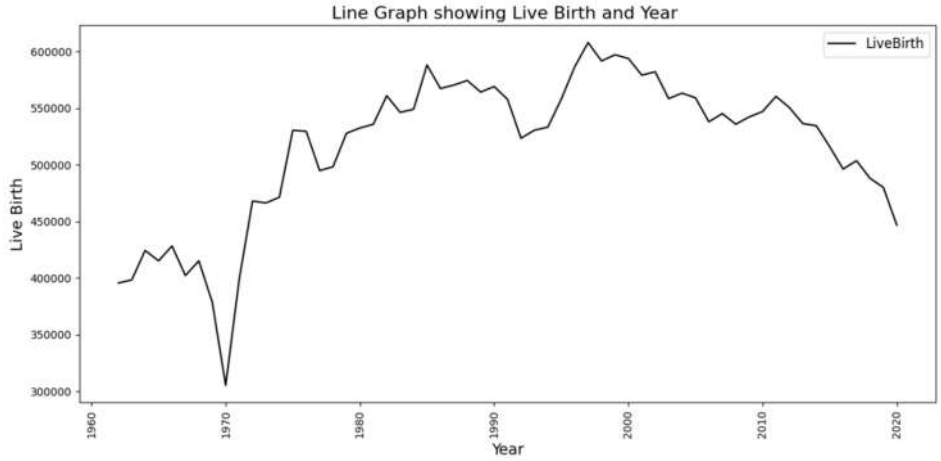
Diagram 01 represents the scattered diagram of live birth.



The scatter plot would show the year on the x-axis and the number of live births on the y-axis. By plotting the data points for each year, we can observe the trend of live births over the years.

Upon observing the scatter plot, we can see that the number of live births has been fluctuating over the years. In the 1960s and 1970s, the number of live births was relatively low and gradually increased in the 1980s and 1990s. However, after 2000, the trend shows a gradual decline in the number of live births in Kerala.

Diagram 02 Represents a Line Graph showing the relationship between LiveBirth and Year



The Line Graph also shows some years where there were significant spikes or drops in the number of live births, such as in the years 1972, 1975, 1996, and 2002. These spikes and drops may be attributed to various factors such as changes in policies, social norms, or economic conditions.

Overall, the scatter plot and Line Graph shows live births in Kerala from 1962 to 2020 shows a fluctuating trend with a recent decline in the number of live births.

Prediction with Live birth dataset

1. Prediction based on Simple Linear Regression

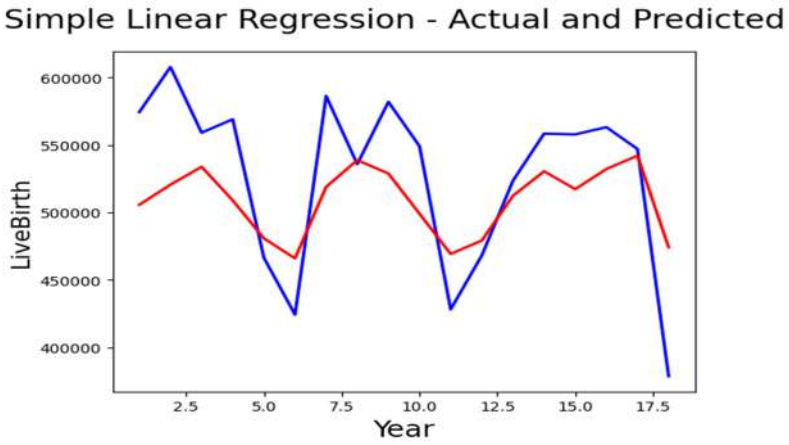
A simple linear regression model was used with a training data set of 0.7 and a testing data set of 0.3. The Mean Squared Error (MSE) is a metric that measures the average squared difference between the predicted values and the actual values in the test data set. A lower MSE value indicates better performance of the model. In this case, the MSE value is 2380213597, which is a relatively high value. This suggests that the model may not be accurately predicting the dependent variable based on the independent variable, and there may be other factors that are influencing the dependent variable that are not captured by the model.

The R squared value is a metric that indicates the proportion of variance in the dependent variable that can be explained by the independent variable in the model. An R squared value of 0.4 indicates that 40% of the variance in the dependent variable can be explained by the independent variable in the model. This value is not very high, which suggests that there may be other variables that are influencing the dependent variable that are not included in the model.

Overall, based on the high MSE value and the relatively low R squared value, it appears that the simple linear regression model may not be the best fit for the data. It may be worth exploring other types of models or incorporating additional variables to improve the accuracy and performance of the model.

Based on this model

Diagram 04



	2020	2021	2022
Live Birth Prediction using SLR	558577	560232	561887

2. Prediction based on Simple Linear Regression with Log

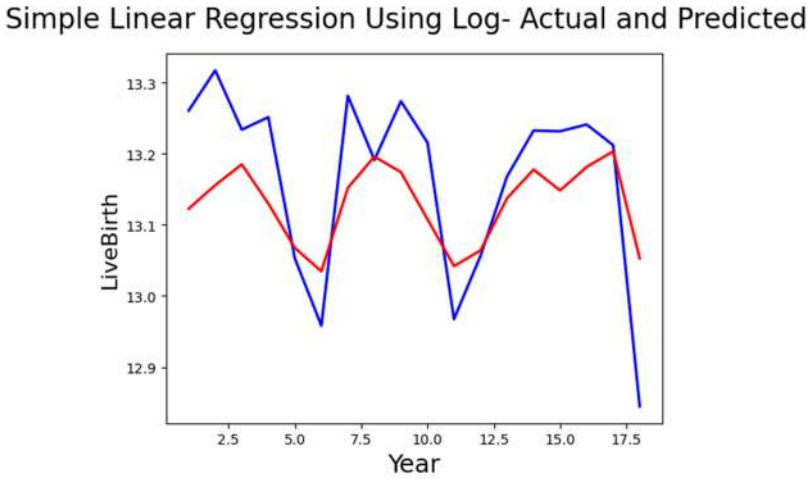
A simple linear regression model with log transformation was used for prediction, with a training data set of 0.7 and a testing data set of 0.3. The performance of the model was evaluated using two metrics: Mean Squared Error (MSE) and R squared value.

The use of a log transformation on the independent variable is often done to help normalize the data and reduce the influence of outliers. The performance of the model, as indicated by the MSE and R squared value, was as follows:

The Mean Squared Error (MSE) was 0.009, which is a very low value. This indicates that the model's predictions are very close to the actual values in the test data set, suggesting that the model is performing well.

The R squared value was 0.43, which indicates that 43% of the variance in the dependent variable can be explained by the independent variable in the model. This is higher than the R squared value of 0.4 obtained in the previous model, suggesting that the log transformation of the independent variable has improved the performance of the model.

Overall, the simple linear regression model with log transformation appears to be performing well based on the low MSE and relatively high R squared value. However, it is important to keep in mind that there may still be other variables or factors that are influencing the dependent variable that are not captured by the model. Further analysis and exploration may be needed to improve the accuracy and robustness of the model.



3. Prediction based on Simple Linear Regression with moving average

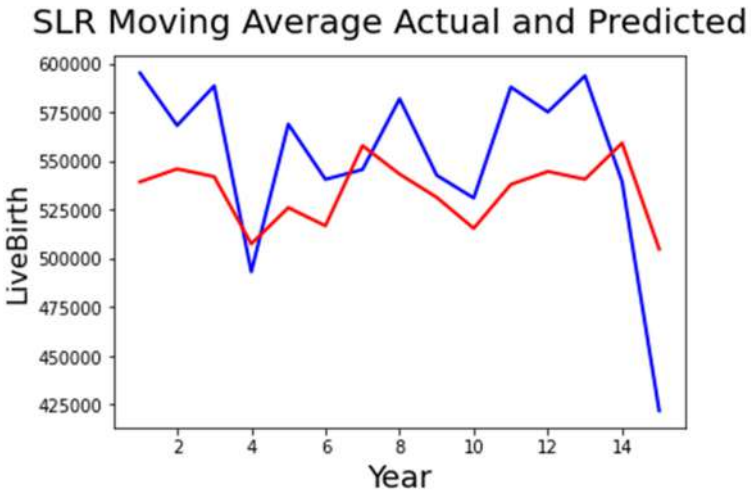
A simple linear regression model with a moving average was used for prediction, with a training data set of 0.7 and a testing data set of 0.3. The performance of the model was evaluated using two metrics: Mean Squared Error (MSE) and R squared value.

The use of a moving average on the independent variable is often done to smooth out fluctuations in the data and make it easier to identify underlying trends. The performance of the model, as indicated by the MSE and R squared value, was as follows:

The Mean Squared Error (MSE) was 1596213570, which is a relatively high value. This indicates that the model's predictions have a higher degree of error when compared to the actual values in the test data set.

The R squared value was 0.19, which is a low value. This indicates that the variation in the dependent variable that can be explained by the independent variable in the model is only 19%.

Overall, the performance of the simple linear regression model with moving average appears to be weaker compared to the other models described earlier. The relatively high MSE and low R squared value suggest that the model is not accurately capturing the relationship between the dependent and independent variables. It is important to note that while moving average can be helpful in identifying trends, it may not be the best choice of transformation for all datasets. Further exploration and experimentation with different transformation techniques and models may be necessary to improve the accuracy and reliability of the predictions.



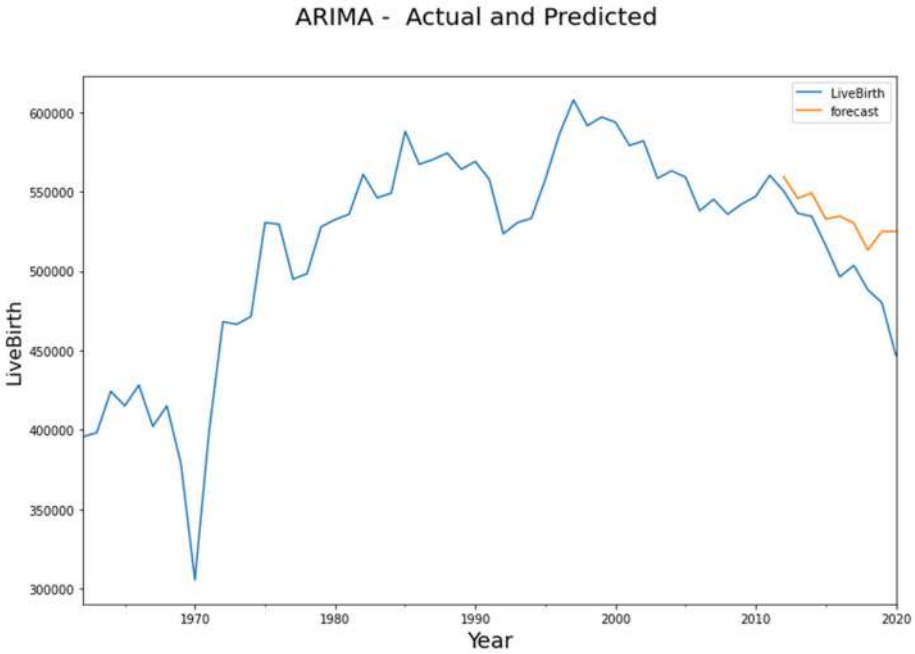
4. Prediction based on ARIMA Auto Regression Model

The ARIMA (AutoRegressive Integrated Moving Average) model is a commonly used time series model that is suitable for forecasting data with trend and/or seasonal patterns. In this case, the ARIMA model has been used for prediction with a training data set of 0.7 and a testing data set of 0.3.

The ARIMA model works by identifying patterns in the time series data and using these patterns to make predictions about future values. This is typically done by analyzing the autocorrelation and partial autocorrelation plots of the time series data to determine the appropriate parameters for the model.

It is also worth noting that visual inspection of the graph generated by the ARIMA model can be helpful in understanding the trend and seasonality in the data, and assessing the accuracy of the predictions.

Overall, the ARIMA model is a useful tool for modeling and forecasting time series data, and can provide valuable insights into trends and patterns in the data. However, it is important to carefully evaluate the performance of the model using appropriate evaluation metrics and to consider alternative models if necessary.



```
model_fit.summary()
```

ARIMA Model Results

Dep. Variable:	D.LiveBirth	**No. Observations:**	58
Model:	ARIMA(1, 1, 1)	**Log Likelihood**	-669.163
Method:	css-mle	**S.D. of innovations**	24750.186
Date:	Mon, 30 Jan 2023	**AIC**	1346.326
Time:	19:09:01	**BIC**	1354.568
Sample:	01-01-1963 - 01-01-2020	**HQIC**	1349.536

	coef	std err	z	P>|z|	[0.025	0.975]
const	642.0359	3706.502	0.173	0.862	-6622.575	7906.647
ar.L1.D.LiveBirth	-0.5880	0.182	-3.232	0.001	-0.945	-0.231
ma.L1.D.LiveBirth	0.8127	0.118	6.862	0.000	0.581	1.045

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	-1.7005	+0.0000j	1.7005	0.5000
MA.1	-1.2304	+0.0000j	1.2304	0.5000

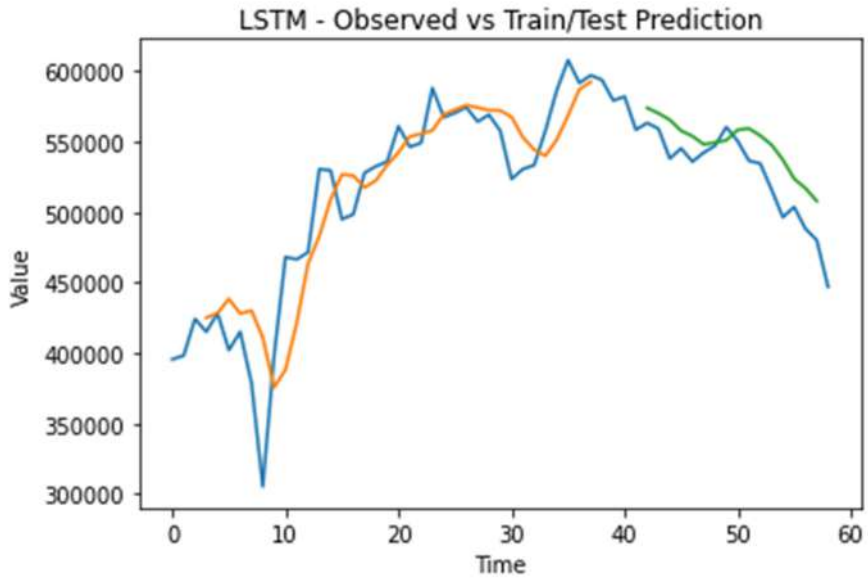
5. Prediction based on LSTM

LSTM (Long Short-Term Memory) is a type of neural network model that is commonly used for modeling and forecasting sequential data, such as time series data. In this case, the LSTM model has been trained using a training data set of 0.7 and a testing data set of 0.3.

LSTM models are particularly well-suited to capturing long-term dependencies in time series data, and can be used to model both linear and non-linear relationships. They work by using a network of interconnected nodes that allow information to flow through the network over time, and are capable of remembering information from previous time steps.

However, it is important to note that LSTM models can be complex and computationally intensive to train, and may require significant amounts of data to achieve good performance. Additionally, the selection of appropriate hyper parameters, such as the number of nodes and the learning rate, can be crucial for obtaining good results.

Overall, LSTM models are a powerful tool for modeling and forecasting time series data, and can provide valuable insights into trends and patterns in the data. However, they require careful selection and tuning of hyperparameters, and their performance should be carefully evaluated using appropriate evaluation metrics.

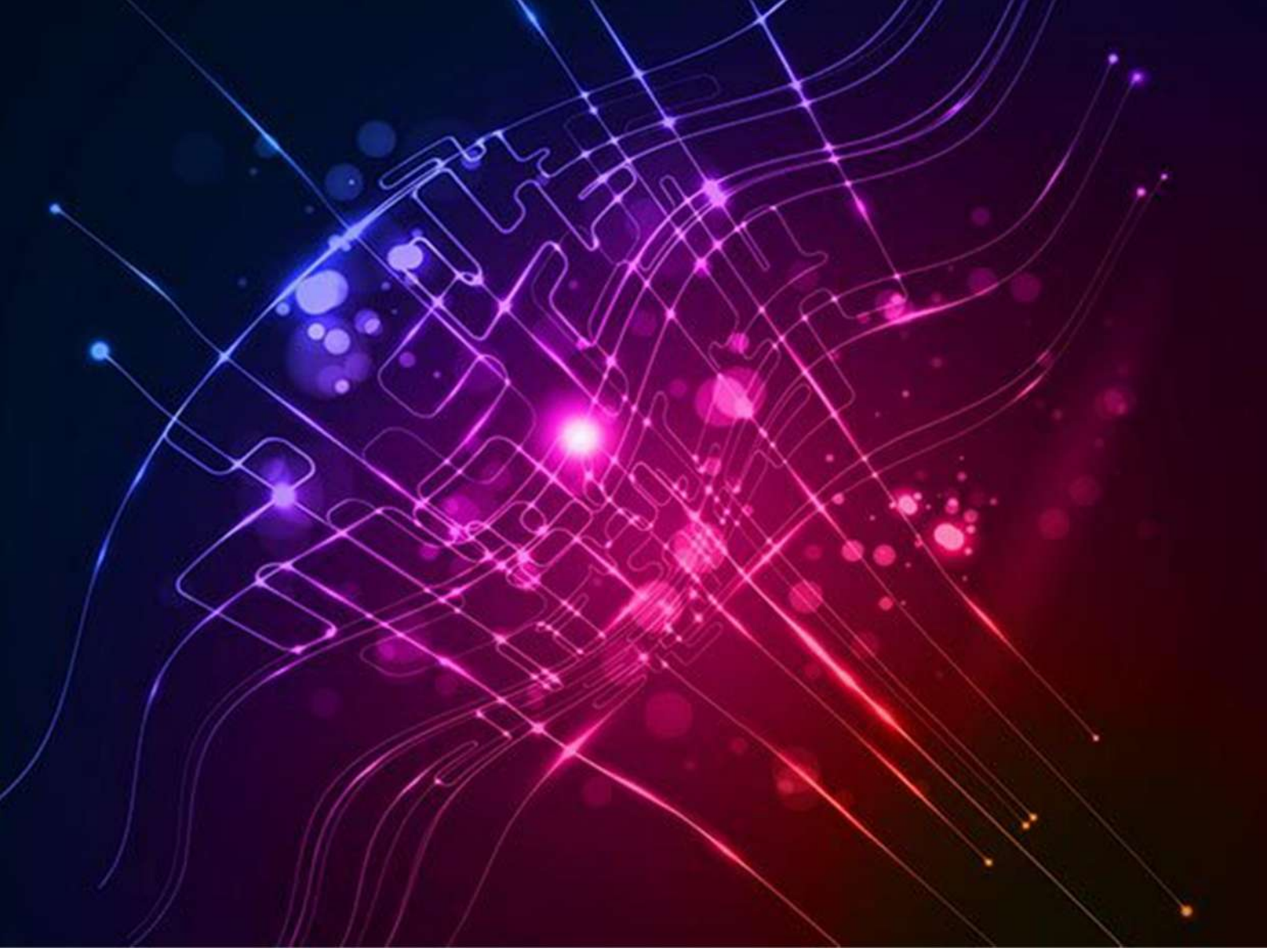


Conclusion

Types of Model used	Training Data	Testing Data	Mean Squared Error	R square value	Remarks
Simple Linear Regression	0.7	0.3	2380213597	0.4	Got Prediction based on input
Simple Linear Regression with Log	0.7	0.3	0.009	0.43	Got Prediction based on input
Simple Linear Regression with moving average	0.7	0.3	1596213570	0.19	Got Prediction based on input
ARIMA Auto Regression Model	0.7	0.3			Graph
LSTM	0.7	0.3			Graph

Reference

- *Introduction to Time Series Forecasting with Python: https://machinelearningmastery.com/time-series-forecasting/*
- *Simple Linear Regression: https://www.statisticssolutions.com/simple-linear-regression/*
- *ARIMA: https://www.statisticshowto.com/arima/*
- *LSTM: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/*



Analysing the correlation between Birth weight and Maternal factors

A review of CRS data using Machine Learning Approach

Submitted By
Sri. Rajesh R, Statistical Assistant Grade I

1.Introduction

Civil registration is the process of recording vital events, such as births, deaths, and marriages, and maintaining official records of these events for legal and statistical purposes. In the Indian state of Kerala, the civil registration system is an essential component of the state's administrative machinery, responsible for recording and maintaining accurate and up-to-date records of vital events occurring in the state. The Civil Registration System (CRS) is the continuous, regular and mandatory registration of important events (births and deaths) of a population. The Births and Deaths Registration Act 1969 (RBD Act) mandated the registration of births, deaths and stillbirths in India on the basis of place of occurrence. The Act went into effect in the state along with several other states on April 1, 1970 through an official gazette notice dated March 21, 1970. The Act requires the registration of births and deaths and the deadline for notification of birth or death to the registrar is 21 days for registration. Registration can be extended beyond 21 days under the terms of article 13 of the Law. The reporting form for data is divided into two sections: legal and statistical. The statistical report for births, deaths, and stillbirths is created using data that is electronically registered through the Sevana/ILGMS software, developed by IKM in Kerala. The statistical report of registered births and deaths is prepared by Department of Economics and Statistics (DES). The "Annual Vital Statistics Report" published annually by the Department is the main publication that describes the situation of CRS in the State. This includes compiling data on births, deaths, infant deaths and stillbirths recorded from civil registries.

Child birth is a complex process that can be influenced by a wide range of factors, including the child's sex, the mother's age at the time of delivery, the number of children the mother has already, the duration of pregnancy, and the manner of delivery. One of the most critical indicators of a child's health at birth is their birth weight, which can have long-term implications for their growth and development. With the advent of machine learning techniques, it has become possible to analyze large datasets and identify complex patterns and relationships between different variables. In this context, a machine learning approach can be used to identify any correlation between a child's birth weight and various maternal and delivery-related factors, such as the child's sex, the mother's age, the number of previous births, the duration of pregnancy, and the manner of delivery. Such an analysis can provide valuable insights into the factors that influence a child's birth weight and enable healthcare providers to develop targeted interventions to improve outcomes for mothers and newborns. By leveraging the power of machine learning, this approach can help to identify previously unknown relationships and patterns that may have a significant impact on maternal and child health. Ultimately, this can lead to more effective and efficient healthcare interventions and improve outcomes for mothers and their newborns.

2. OBJECTIVE

To assess and illustrate the relationship between a child's birth weight and other maternal characteristics including child’s sex, the mother's age at delivery, the number of children the mother has already, gestational age, and the manner of delivery, an efficient machine learning technique should be developed. The analysis has been done using Python Programming.

METHODOLOGY AND METHOD USED

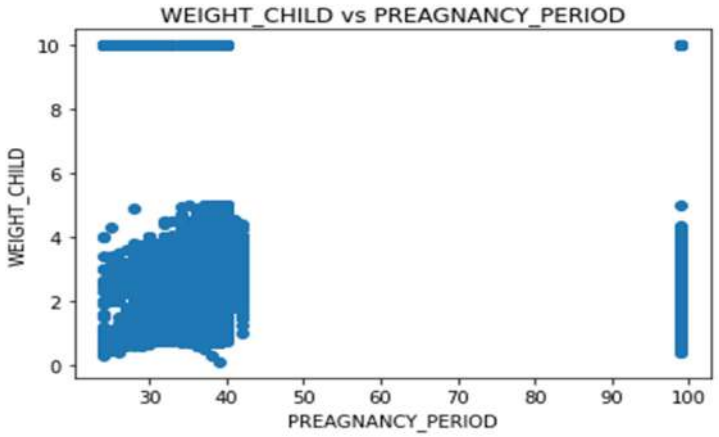
Trying to apply multiple regression method to analyses the data. Before that we have to make the data suitable for the study Which typically involves following steps.

1. **Data collection:** Collect the dataset that includes the predictor variables (also called independent variables or features) and the target variable (also called dependent variable or response variable)

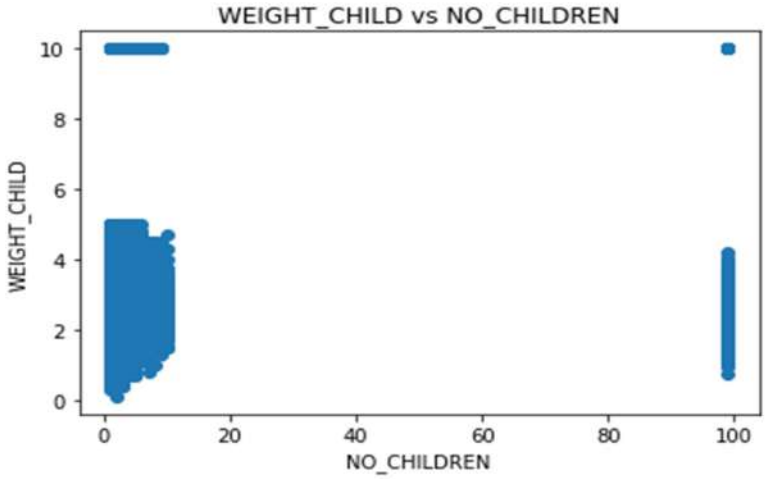
The dataset used for the study is Civil Registration System data - Statistical part from 2008 to 2019. Data consist of a total of 629272 (more than 62 lakhs) records.

2. Data cleaning and preprocessing: Perform data cleaning and preprocessing tasks such as removing missing values, handling outliers, and scaling the data.

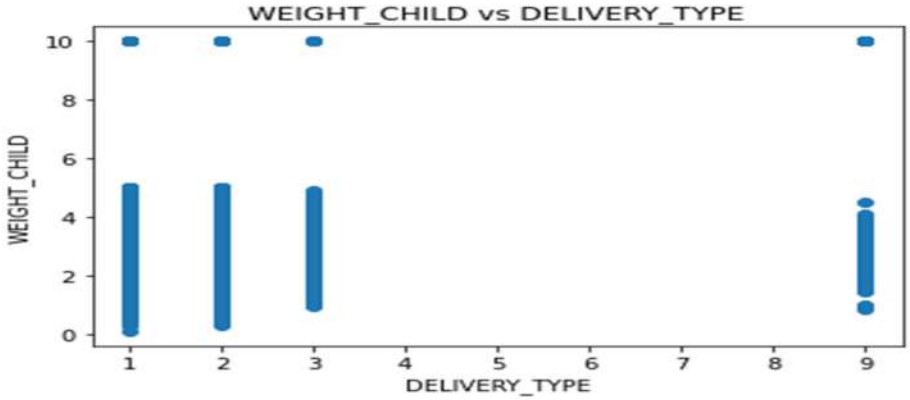
Here in our data by checking the data with outliers and when visualizing it found that there are certain outliers which affect the quality of the data and our study so have to remove it first.



By analysing the graph it is clear that weight of the child versus pregnancy period graph has outliers on weight as 9kg (which is actually the code used for unknown weight) and pregnancy period as 99 months (which is actually the the code used for unknown months) any way have to remove such outliers.



From this data, It is clear that weight of the child versus number of children the mother have graph has outliers on number of children as 99 (which is actully the code used for unspecified number of children the mother had) any way have to remove such outliers.



By analysing the above data, It is clear that weight of the child versus Delivery type graph has some outliers on delivery type as 9(which is actualy the code used for unspecified delivery type)any way have to remove such outliers.

In order to remove outliers, run inter quartile range in python and that removes the outliers stated above and get a clean data and since we used published dataset for the study so there is no scope for missing values on all columns.

3. Splitting the data: Split the dataset into training and testing sets using a holdout method or cross-validation.

In order to train the system we split data set into training dataset and testing dataset. Commonly we split the data into 80:20 ratios. But here, since the dataset was very large (contains more than 62 lakhs of data) we split the data set into 99:1 ratio (only for plotting).

4. Building the model: Build a multilinear regression model using a suitable Python library such as scikit-learn or statsmodels etc.

5. Model training: Train the model on the training set.

Train the system with training dataset and for the evaluation we apply that on testing dataset and check the accuracy.

6. Model evaluation: Evaluate the model's performance on the testing set using appropriate evaluation metrics such as mean squared error (MSE), root mean squared error (RMSE), and R-squared.

4. DATASET USED

The dataset used for the study was Civil Registration System – Birth data from the year 2008 to the year 2019. The data consist of a total of 629272 (more than 62 lakhs) records. Dataset has column headings as

REG_YEAR 1093403 non-null int64
1 SEX 1093402 non-null float64
2 MOTHERS_AGE_DELIVERY 1093400 non-null float64
3 NO_CHILDREN 1093396 non-null float64
4 PREAGNANCY_PERIOD 1093257 non-null float64
5 DELIVERY_TYPE 1093393 non-null float64
6 WEIGHT_CHILD 1093393 non-null float64

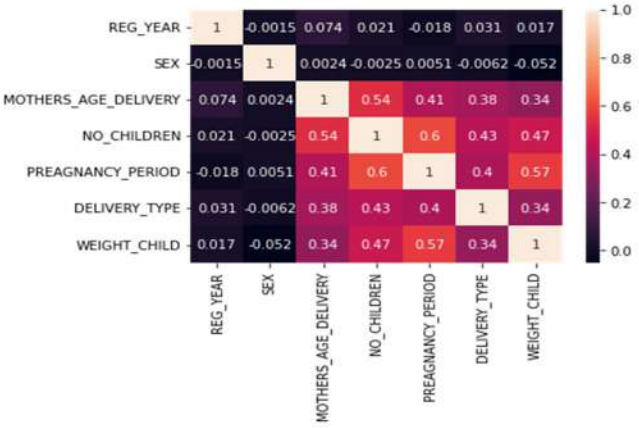
The row data is of the form

REG_ YEAR	SEX	MOTHERS_ AGE_ DELIVERY	NO_ HILDREN	PREAGNANCY_ PERIOD	DELIVERY_ TYPE	WEIGHT_ CHILD
2019	2	28	1	39	2	3.67
2019	2	21	1	40	1	2.77
2019	1	28	4	37	2	2.59
2019	2	33	3	38	2	3
2019	1	31	2	38	2	3.27
2019	2	30	3	39	2	2.51
2019	2	31	2	38	2	3
2019	2	27	4	40	2	3.7
2019	1	28	3	38	2	4.05
2019	1	19	1	40	1	2.7
2019	2	28	2	39	2	3.37
2019	1	19	1	38	1	2.53
2019	1	26	1	39	1	3.28
2019	1	25	2	40	1	4
2019	1	27	2	38	1	2.95
2019	2	29	2	39	2	3.5
2019	1	29	2	40	2	2.75
2019	2	25	1	40	2	3
2019	1	36	3	40	2	3
2019	2	31	2	37	2	2.8
2019	2	27	1	39	2	3.5
2019	2	26	2	39	2	2.4
2019	2	32	2	40	2	3
2019	2	23	1	40	2	3.2
2019	2	26	1	38	1	3
2019	2	26	1	40	2	3.25
2019	2	24	1	39	1	2.9
2019	1	25	2	38	2	3.05
2019	2	30	2	36	2	3.2
2019	1	25	1	36	1	3.04
2019	1	27	1	39	1	3.25
2019	2	27	1	40	2	2.75
2019	1	26	2	38	2	3.5
2019	2	31	4	38	1	3
2019	1	33	2	39	2	2.47
2019	1	28	2	37	1	2.6
2019	2	27	1	39	1	2.87
2019	1	33	3	37	1	3
2019	2	29	2	40	1	3.11
2019	2	31	1	37	2	2.84
2019	2	21	1	38	1	3.22
2019	2	31	4	39	2	3.65
2019	2	20	1	40	2	2.5
2019	2	28	2	39	1	3
2019	2	35	2	39	2	3.3
2019	1	23	1	40	2	2.5
2019	1	29	2	37	2	3
2019	1	25	1	38	2	2.75
2019	2	23	1	38	1	2.4
2019	1	23	2	38	2	3

And the data set after cleaning is of the form

	Reg-Year	Sex	Months- Age - Delivery	Pregnancy period	Delivery Type	Weight Child
6292723	2019	2	23	1	37.0	2
6292724	2019	1	36	3	40.0	2
6292725	2019	2	19	1	39.0	2
6292726	2019	2	29	1	39.0	2

In order to apply multi linear regression model. We consider weight of the child as dependent variable and Sex of the child, Mother's age at delivery, number of children, pregnancy period, delivery type as independent variables.



From heat map of correlation it is clear that sex of the child ,Registration year have no correlation with the weight of the child so sex of the child does not consider for further study. Now the weight of the child as dependent variable and Mother's age at delivery, number of children, pregnancy period, delivery type as independent variables. And now fit the data into multi linear regression model.

5.TOOLS AND LIBRARIES USED:

In python, there are several libraries available for machine learning using multi-linear regression, and here we use the following**.**

3.1 NumPy: A fundamental scientific computing library for Python that provides support for large multi-dimensional arrays and matrices, and a wide range of mathematical functions.

3.2 Pandas: A data analysis library that provides data structures and tools for manipulating and analyzing numerical tables and time series data.

3.3 Statsmodels: A statistical library that provides tools for exploring data, estimating statistical models, and performing statistical tests.

3.4 Seaborn

Data visualization: Seaborn provides a wide range of functions for visualizing data, such as scatter plots, histograms, box plots, heatmaps, and more. These functions can be used to explore the relationships between variables, identify patterns in the data, and detect outliers or anomalies.

Feature engineering: For feature engineering, which involves transforming the raw data into a format that is suitable for machine learning algorithms? Seaborn provides functions for scaling, encoding, and transforming data, which can help to improve the performance of machine learning models.

Model evaluation: For evaluating the performance of machine learning models. For example, Seaborn provides functions for generating confusion matrices, classification reports, and ROC curves, which can be used to assess the accuracy, precision, recall, and other performance metrics of a model.

3.5 Matplotlib.pyplot

It is a useful tool for visualizing and analyzing the data and results

3.6 Sklearn.linear_model:

Is a module in Scikit-learn library in python. The linear regression class in sklearn.linear_model can be used to fit a linear regression model to a dataset and make predictions on new data.

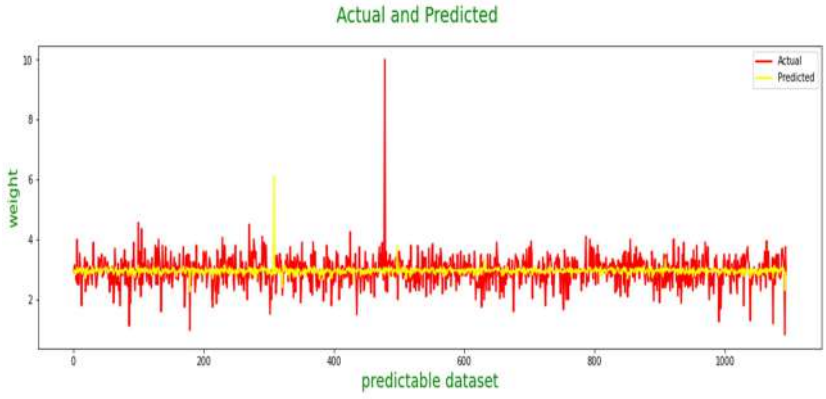
6. RESULT AND INFERENCE:

Calculated the mean squared error value

Mean Absolute Error: 0.3503968182783409

The value .035 represents the average absolute difference between the predicted and actual values of the target variable, the closer MAE value to zero the better the model making predictions.

Based on our model we plot the graph



The prediction made is :
This the actual data in the data set

	REG_YEAR	SEX	MOTHERS_AGE_DELIVERY	NO_CHILDREN	PREAGNANCY_PERIOD	DELIVERY_TYPE	WEIGHT_CHILD
546948	2019	1.0	22.0	1.0	39.0	1.0	3.43

And our predicted value is
predict_final=model.predict([[22.0,1.0,39.0,1.0]])

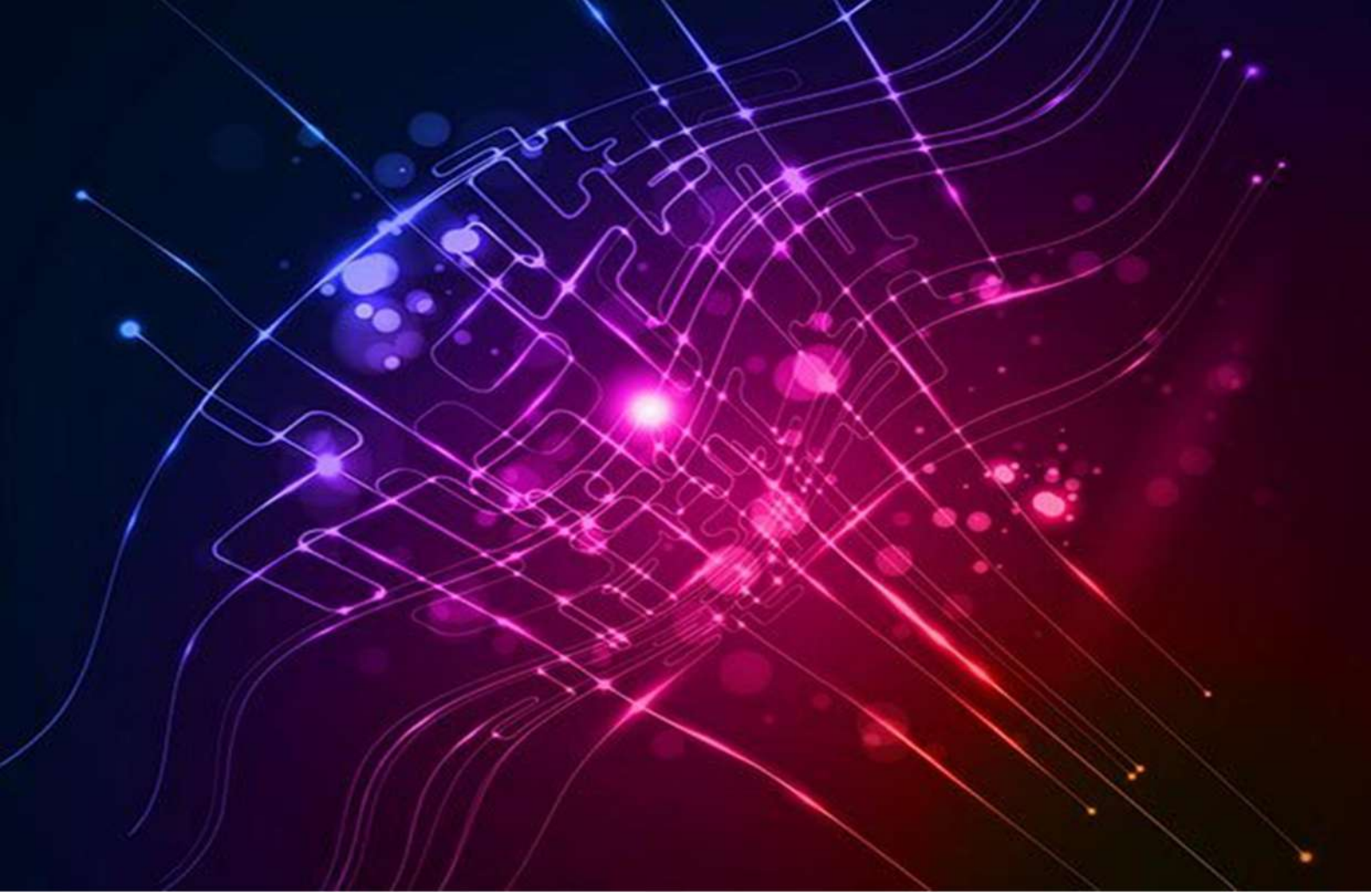
OUTPUT IS 3.86kg

CONCLUSION:

There is a difference between actual and predicted value (3.43-3.86= - .43 kg). Hence more studies are needed to identify an accurate model to fit the data and make predictions and may be other factors like nutrition's etc. are to be considered to predict the weight of the child in a better manner.

REFERENCE:

- *Journal of Obstetrics and Gynaecology, used decision tree and random forest algorithms to analyse data from over 33,000 deliveries: https://journals.sbmu.ac.ir/aab/article/download/15412/13220*
- *Early Prediction of Weight at Birth Using Support Vector Regression: https://link.springer.com/chapter/10.1007/978-3-030-30648-9_5*
- *Sex differences in outcomes of very low birth weight infants: https://fn.bmj.com/content/83/3/F182*



A Machine Learning Approach to classification of cause of death

Submitted By
Sri. Preeth V.S., Deputy Director (Nosologist)

1. Introduction

Reliable cause-specific mortality statistics is required on a regular basis by Administrators, Policy Planners, Researchers and other Professionals for evidence-based decision-making with regard to resource allocation, monitoring of indicators, identifying the priorities for programs and other related activities in the area of Public Health. Keeping this in view, the scheme of Medical Certification of Cause of Death (MCCD) was introduced in the country under the provisions of the Registration of Births and Deaths (RBD) Act, 1969. In Kerala the scheme is presently implemented only in four Corporations viz. – Thiruvananthapuram, Kollam, Ernakulum and Kozhikode and in Alappuzha Municipality. Department of Economics and Statistics (DES) cross tabulates the data on medically certified cause of deaths reported from the above-mentioned 5 centres in conformity with the International Classification of Diseases (ICD) by age and sex of the deceased.

The necessary data is collected in the prescribed forms (Form 4 for Hospital deaths and Form 4A for Non-institutional deaths). Both these forms have been designed by the World Health Organization (WHO). The forms are filled-up by the medical professionals attending to the deceased at the time of terminal illness. Thereafter, these forms are to be sent to the concerned Registrars of Births and Deaths for onward transmission to the Chief Registrar Office for tabulation as per the National List of Causes of Death based on Tenth Revision of International Classification of Disease (ICD-10). The coding is done by Deputy Health Officers (DHO) in the corporation/Municipality offices. Nosologist in DES cross tabulates the data by cause of death, age and sex in excel format and prepares consolidation statements in prescribed format. The States/UTs subsequently send it to the Office of RGI in the form of Statistical Table-11 for consolidation at the National level.

2. Objective

To create a model that will be able to recognize and determine handwritten characters from scanned images of Medical certification of Cause of death forms filled by doctors and to classify the cause of deaths as per international classification of Diseases (ICD-10).

3. Methodology

3.1. **Data collection**: This step involves collecting a dataset of scanned handwritten death certificates. The dataset should be labelled with the cause of death for each certificate. This means that each certificate should be associated with a specific category of cause of death, such as heart disease, cancer, or respiratory illness.

3.2. **Text extraction**: After collecting the dataset, the next step is to use Optical Character Recognition (OCR) or handwriting recognition methods to extract the text from the scanned documents. The extracted text should be cleaned and pre-processed to remove any inaccuracies or inconsistencies.

3.3. **Feature extraction**: The next step is to extract relevant features from the text that can be used for classification. These features may include words, phrases, or other

information related to the cause of death. For example, if a death certificate mentions the phrase "heart attack", this can be used as a feature to classify the cause of death as heart disease.

3.4. **Model training**: After feature extraction, the next step is to train a machine learning model on the labelled dataset using the extracted features. This can be done using supervised learning techniques such as Support Vector Machines, Random Forest, Naive Bayes, or Decision Trees. The goal is to develop a model that can accurately classify the cause of death based on the extracted features.

3.5. **Model evaluation**: Once the model is trained, the next step is to evaluate its performance using appropriate metrics such as accuracy, precision, recall, and F1-score. This helps to assess the effectiveness of the model in correctly classifying the cause of death from the scanned handwritten death certificates.

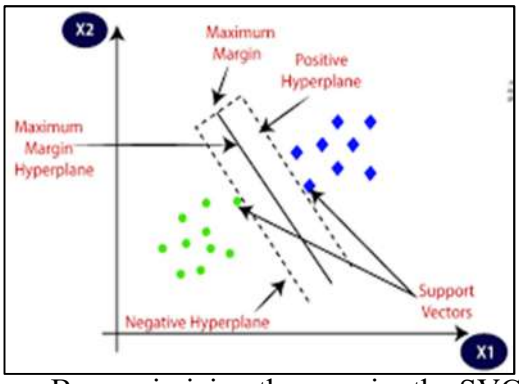
3.6. **Model optimization**: If the model is not performing well, it may be necessary to perform model optimization by adjusting the parameters or using different feature extraction methods. This helps to improve the accuracy of the model and make it more effective in classifying the cause of death.

3.7. **Model deployment**: Once the model is trained and optimized, it can be deployed for use in classifying the cause of death from new, unseen scanned handwritten death certificates. This can be done by integrating the model into an application or system that can automatically classify the cause of death based on the scanned handwritten death certificate input.

4. Models used

4.1. Support Vector Classifier (SVC)

The Support Vector Classifier (SVC) is a popular supervised learning algorithm used in machine learning for classification tasks. It is a type of Support Vector Machine (SVM) algorithm that is effective in solving both linear and non-linear classification problems..



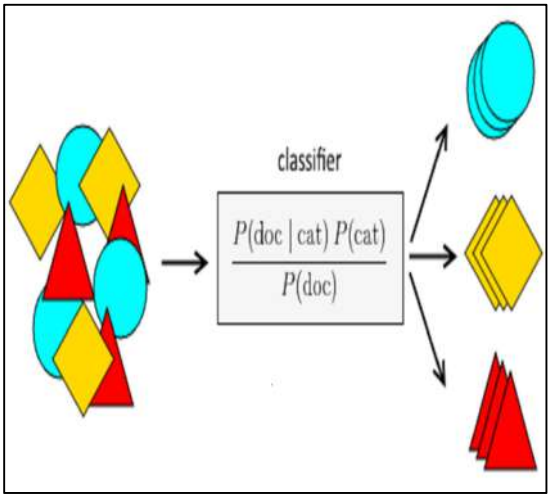
The SVC algorithm works by finding the hyperplane that maximally separates the different classes in the input data. This hyperplane is determined by the support vectors, which are the data points that lie closest to the decision boundary. The SVC algorithm tries to find the hyperplane that maximizes the margin, which is the distance between the support vectors and the decision boundary. By maximizing the margin, the SVC algorithm is able to minimize the generalization error, which is the error rate on new, unseen data.

One of the benefits of using the SVC algorithm is that it is effective in handling high-dimensional data, which makes it suitable for use in text classification, image recognition, and other types of classification tasks. Additionally, SVC is also able to handle both linearly separable and non-linearly separable datasets by using different types of kernels. Some common kernels used

with the SVC algorithm include linear, polynomial, radial basis function (RBF), and sigmoid kernels.

In terms of implementation, the SVC algorithm can be used in Python using the scikit-learn library. The SVC class in scikit-learn provides various hyperparameters that can be tuned to improve the performance of the algorithm, such as the kernel type, regularization parameter, and gamma parameter.

4.2. Naïve Bayes Classifier



The Naïve Bayes Classifier is a simple yet powerful algorithm used in machine learning for classification tasks. It is based on the Bayes theorem of probability theory and is commonly used in text classification, sentiment analysis, spam filtering, and other applications where data is represented as a collection of features.The Naïve Bayes Classifier works by assuming that the presence of a particular feature in a class is independent of the presence of other features. This is known as the "naive" assumption, which simplifies the probability calculations and makes the algorithm computationally efficient. Based on this assumption, the Naïve Bayes Classifier calculates the probability of a data point belonging to a particular class based on the conditional probability of each feature given the class. The class with the highest probability is then assigned to the data point.

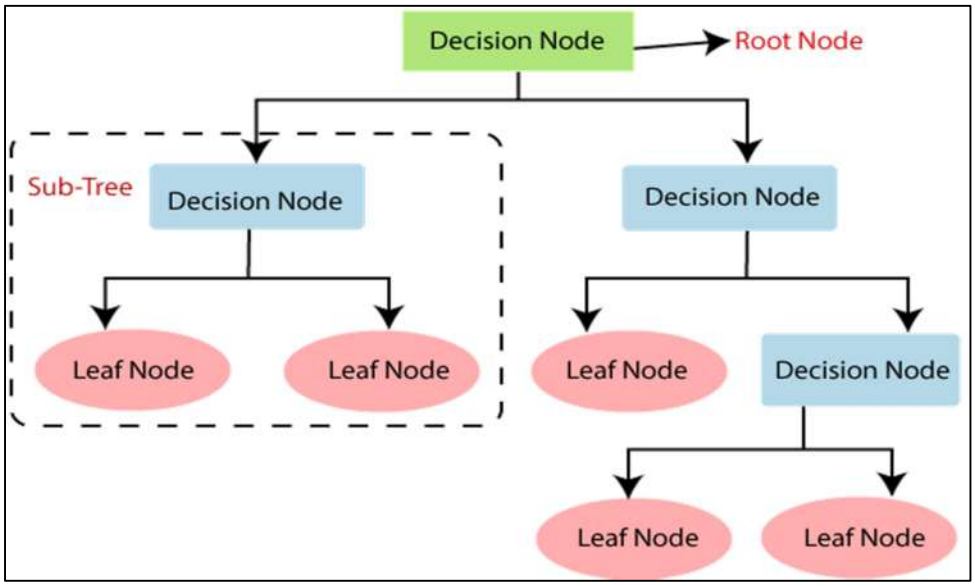
One of the benefits of using the Naïve Bayes Classifier is its simplicity and speed. The algorithm requires only a small amount of training data and can be trained quickly even on large datasets. Additionally, the Naïve Bayes Classifier is robust to irrelevant features, meaning that it can effectively classify data even when there are features that are not useful for classification.

There are three types of Naïve Bayes Classifier: Bernoulli, Multinomial, and Gaussian. The Bernoulli Naïve Bayes Classifier is used for binary classification problems where the features are binary variables. The Multinomial Naïve Bayes Classifier is used for multi-class classification problems where the features are counts of occurrences. The Gaussian Naïve Bayes Classifier is used for continuous features that are assumed to follow a Gaussian (normal) distribution.

In terms of implementation, the Naïve Bayes Classifier can be used in Python using the scikit-learn library. The Gaussian Naïve Bayes Classifier is available in scikit-learn's GaussianNB class, while the Multinomial Naïve Bayes Classifier and Bernoulli Naïve Bayes Classifier are available in the MultinomialNB and BernoulliNB classes, respectively.

4.3. Decision Tree Classifier

A decision tree classifier is a type of machine learning algorithm that is commonly used for classification tasks. It builds a tree-like model of decisions and their possible consequences, based on a set of training data.



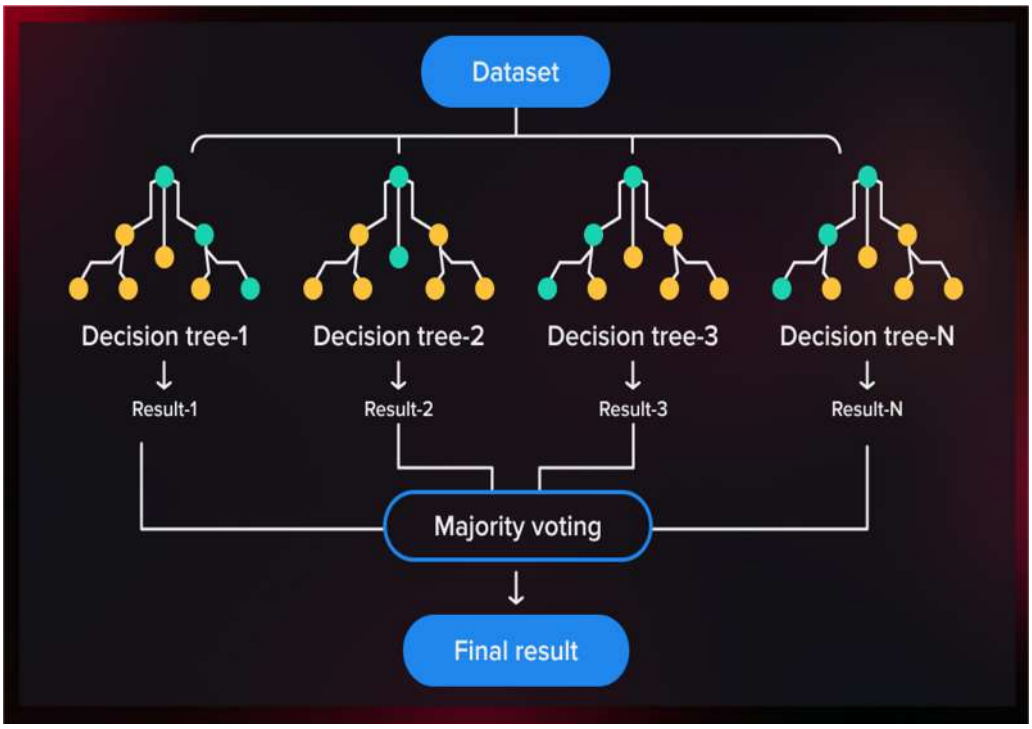
The decision tree starts with a root node and branches out into different paths, with each path corresponding to a possible decision based on a specific feature or attribute. Each decision or split in the tree is made based on the feature that provides the most information gain, meaning it separates the data into the most distinct groups.

The process continues recursively until a stopping criterion is met, such as when all instances in a branch belong to the same class or when the tree reaches a maximum depth. Once the tree is built, it can be used to classify new instances by traversing down the tree from the root to a leaf node, which corresponds to a particular class label.

Decision trees have several advantages, including their interpretability and ability to handle both numerical and categorical data. However, they may suffer from over fitting and instability, and may not perform well on datasets with high dimensionality or noisy data. Ensemble methods, such as random forests and gradient boosting, can help overcome some of these limitations.

4.4. Random Forest Classifier

Random Forest Classifier is an ensemble machine learning algorithm that is used for classification tasks. It is an extension of the decision tree algorithm and is based on the concept of building multiple decision trees and combining their results to make predictions.



The algorithm works by randomly selecting a subset of features and building decision trees based on these subsets. This process is repeated multiple times to create a forest of decision trees. During training, each decision tree in the forest is trained on a random subset of the training data, with replacement (i.e., bootstrap samples). The final prediction is made by aggregating the predictions of all the decision trees in the forest, either by taking the majority vote or averaging the predictions.

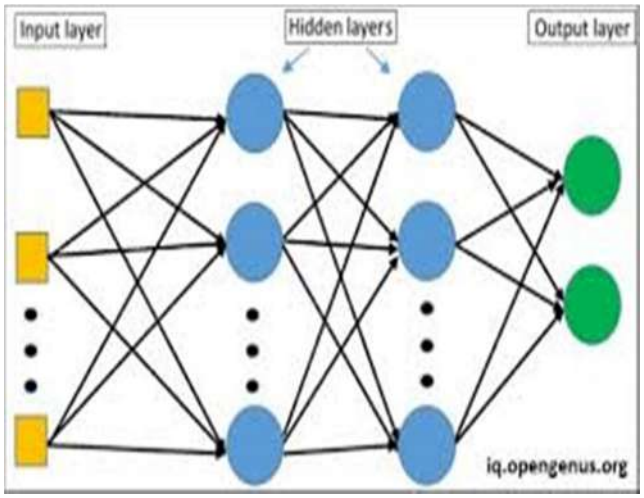
The use of multiple decision trees in a random forest has several advantages. First, it reduces the risk of over fitting, as the average prediction of many trees is often more stable and accurate than the prediction of a single tree. Second, the algorithm can handle both numerical and categorical data and can identify important features for classification. Third, it can deal with missing values in the dataset.

Random forest classifiers have been shown to be effective in a wide range of classification problems, including text classification, image classification, and bioinformatics. However, they can be computationally expensive and may require tuning of hyper parameters such as the number of trees in the forest and the size of the subsets used for feature selection.

4.5. **Multilayer Perceptron (MLP) classifier**

Multilayer Perceptron (MLP) classifier is a type of artificial neural network that is used for classification tasks. It is a feedforward neural network that consists of multiple layers of nodes or neurons, with each layer connected to the next in a sequence. The input layer of the MLP receives the input data, which is then passed through one or more hidden layers, each of which consists of multiple neurons. The neurons in each hidden layer are fully connected to the neurons in the previous layer,

and each neuron in a hidden layer computes a weighted sum of the inputs from the previous layer and applies an activation function to produce its output. The output layer produces the final classification result.



The MLP classifier is trained using a supervised learning algorithm, such as backpropagation. During training, the weights and biases of the neurons are adjusted iteratively to minimize the error between the predicted output and the actual output. The error is measured using a loss function, such as cross-entropy or mean squared error.

MLP classifiers are widely used in many applications, including image recognition, speech recognition, and natural language processing. They are powerful and flexible models that can learn complex patterns in the input data. However, they can be computationally expensive and may require tuning of hyperparameters such as the number of hidden layers, the number of neurons in each layer, and the learning rate.

5. Data sets used

A total of 100 scanned images of MCCD forms for the year 2022 were gathered to extract text information. Additionally, the MCCD data for the year 2022 in Excel format consists of 17000 cases was utilized to train the machine learning models mentioned above.

6. Tools and Libraries used

6.1 Tesseract OCR- Tesseract OCR is an open-source optical character recognition (OCR) engine developed by Google. OCR is a technology that allows machines to recognize printed or handwritten text within an image and convert it into machine-readable text. In the project, the Pytesseract library is utilized to integrate with Tesseract.

6.2 Pandas - for data manipulation and analysis

6.3. scikit-learn (sklearn for short)- is a free and open-source machine learning library for the Python programming language. It provides tools for data analysis and preprocessing, classification, regression, clustering, model selection and evaluation, and more. The library is built on top of NumPy, SciPy, and Matplotlib, which are popular Python libraries for scientific computing and data visualization.

6.4. TfidfVectorizer - for converting text data into numerical feature vectors using the term frequency-inverse document frequency (TF-IDF) approach

6.5. SVC (Support Vector Classification) - for training the SVM (Support Vector Machine) classifier model

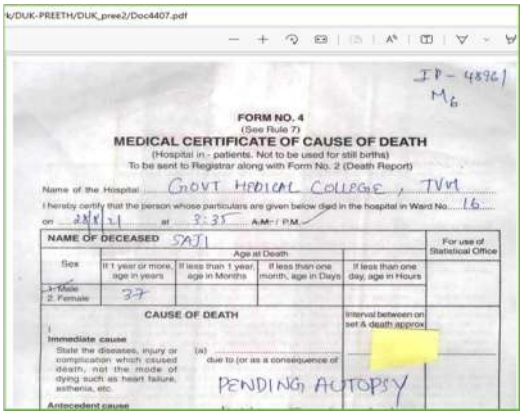
6.6. MultinomialNB -is a class in the sklearn.naive_bayes module of scikit-learn library in Python, which implements the Naive Bayes algorithm for multinomially distributed data.

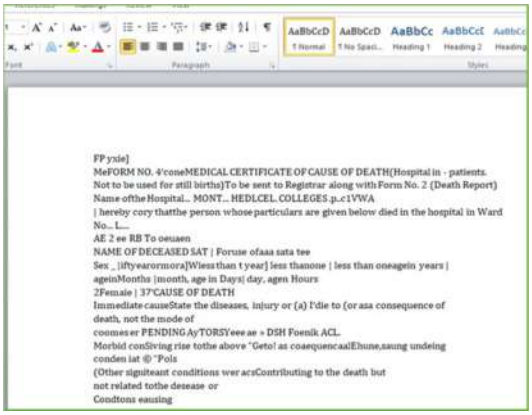
6.7. DecisionTreeClassifier- a class for creating Decision tree classifier
6.8. MLPClassifier - for creating a multi-layer perceptron (MLP) neural network classifier
6.9. Pipeline - for chaining together multiple machine learning steps into a single workflow
6.10. train_test_split - for splitting the dataset into training and testing sets
6.11. GridSearchCV - for performing hyperparameter tuning of the SVM classifier using grid search
6.12. accuracy_score - for calculating the accuracy of the classifier
6.13. precision_score - for calculating the precision of the classifier
6.14. recall_score - for calculating the recall of the classifier
6.15. f1_score - for calculating the F1-score of the classifier
6.16. confusion_matrix - for creating a confusion matrix to evaluate the performance of the classifier
6.17. classification_report - for generating a report of the classification performance, including precision, recall, and F1-score
6.18. seaborn - for creating visualizations, such as heatmaps
6.19. matplotlib.pyplot - for creating visualizations, such as line plots and scatter plots.
6.20. RandomForestClassifier: a class for creating a random forest classifier, which is an ensemble of decision trees.

7. Results

7.1. Text Extraction

Tesseract OCR was not successful in correctly recognizing the text owing to the unfamiliarity of the engine with the handwritten characters of doctors. One of such result is shown below. Although Google Cloud API Vision was utilized to enhance the results, it too failed to deliver accurate outcomes.





7.2. Evaluation of Models

As the text extraction process failed, the models (SVC classifier, Multinomial NB classifier, Decision Tree Classifier, Random Forest Classifier, and MLP classifier) were trained using MCCD data in an Excel format. To determine the optimal model for the classification problem, the accuracy, F1-score, recall, and precision of the models under consideration, were computed. Below are brief definitions of these terms.

Accuracy: This is the proportion of correctly classified instances out of all instances in the dataset. It is calculated by dividing the number of correctly classified instances by the total number of instances.

F1-score: This is a weighted average of the precision and recall metrics. It is a measure of a model's accuracy that considers both the precision and recall. The F1-score is often used in cases where there is an imbalance in the class distribution (i.e., when one class is much more prevalent than the other).

Recall: This is the proportion of positive instances that were correctly classified by the model out of all actual positive instances. It is also known as sensitivity or true positive rate.

Precision: This is the proportion of positive instances that were correctly classified by the model out of all instances classified as positive by the model. It is also known as positive predictive value.

Table 7.2.1 Accuracy of the Models under consideration

Model	accuracy	precision	recall	F1-score
SVC	98.36	98.11	98.36	98.06
Multinomial NB	96.36	94.98	96.35	95.4
Decision Tree	98.36	98.11	98.36	98.05
Random Forest	98.36	98.11	98.36	98.05
MLP	98.43	98.18	98.43	98.13

Table 7.2.1 depicts the accuracy, F1-score, recall, and precision of the models under consideration. it seems that the MLP model has the highest F1-score of 98.13%, which combines both precision and recall into a single metric. The MLP model also has the highest accuracy of 98.43%, indicating that it correctly classified a high percentage of the samples.

Although the SVC, Decision Tree, and Random Forest models also have high accuracy, precision, and recall values, their F1-scores are slightly lower than the MLP model. This indicates that while they correctly classified a high percentage of samples, they may have had slightly more false positives or false negatives than the MLP model.

The MultinomialNB model has a significantly lower accuracy, precision, recall, and F1-score compared to the other models, indicating that it may not be the best choice for this classification problem.

Overall, based on the provided table, the MLP model appears to be the best choice for this classification problem, as it has the highest F1-score and accuracy among the models under consideration.

7.3 Model optimization

Even though all the models have achieved high accuracy, we attempted to optimize theMLP model by fine-tuning its hyperparameters. Below are the specific parameters that were fine-tuned for the MLP classifier.

Hidden layers: The number of hidden layers in the neural network. Each hidden layer is a layer of neurons between the input and output layers. By default, MLPClassifier uses a single hidden layer with 100 neurons.

Alpha: A regularization parameter that controls the strength of L2 regularization on the weights of the neural network. L2 regularization helps prevent overfitting by shrinking the weights towards zero. By default, alpha is set to 0.0001.

Solver: The algorithm used to optimize the weights and biases of the neural network during training. The possible values are 'lbfgs', 'sgd', and 'adam'. By default, MLPClassifier uses 'adam'.

Verbose: Controls the amount of information printed during training. The possible values are True or False. If set to True, the training progress is printed to the console. By default, verbose is set to False.

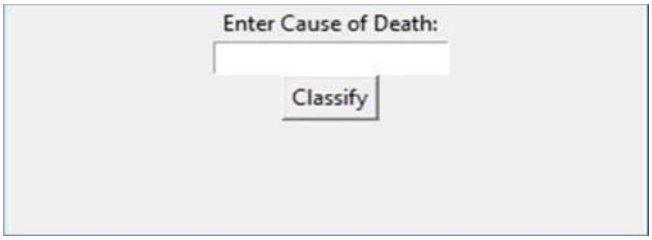
Learning_rate_init: The initial learning rate used by the optimizer. This determines the step size taken during weight updates. By default, learning_rate_init is set to 0.001.

Table 7.3.1 Accuracy of the model for different values of hyper parameters.

Trial	Hidden layers	alpha	verbose	learning_rate_init	solver	Accuracy
1	(10,10)	0.0001	10	0.1	sgd	98.36
2	(50,50)	0.0001	10	0.1	sgd	98.36
3	(100,100)	0.0001	10	0.1	sgd	98.43
4	(200,200)	0.0001	10	0.1	sgd	98.36
5	(100,100)	0.001	10	0.1	sgd	98.43
6	(100,100)	0.01	10	0.1	sgd	98.43
7	(100,100)	0.1	10	0.1	sgd	98.36
8	(100,100)	0.001	10	0.01	sgd	98.61
9	(100,100)	0.001	10	0.001	sgd	98.36
10	(100,100)	0.001	10	0.01	lbfgs	98.36
11	(100,100)	0.001	10	0.01	adam	98.36

Based on the table, the best parameters for the MLPClassifier are: Hidden layers: (100,100), Alpha: 0.001,Verbose: 10,Learning_rate_init: 0.01,Solver: 'sgd'.These parameters were associated with the highest accuracy of 98.61%.

7.4 Model Deployment



After the model is trained and optimized, it is deployed to classify the cause of death from new cases. To facilitate this process, a Graphical User Interface is developed to make it easy to classify the cause of death. In addition, a module is prepared to enable users to input the cause of death as a CSV file, and the model will predict the corresponding ICD codes and add them to a new column labeled "code" in the CSV file.

Conclusion

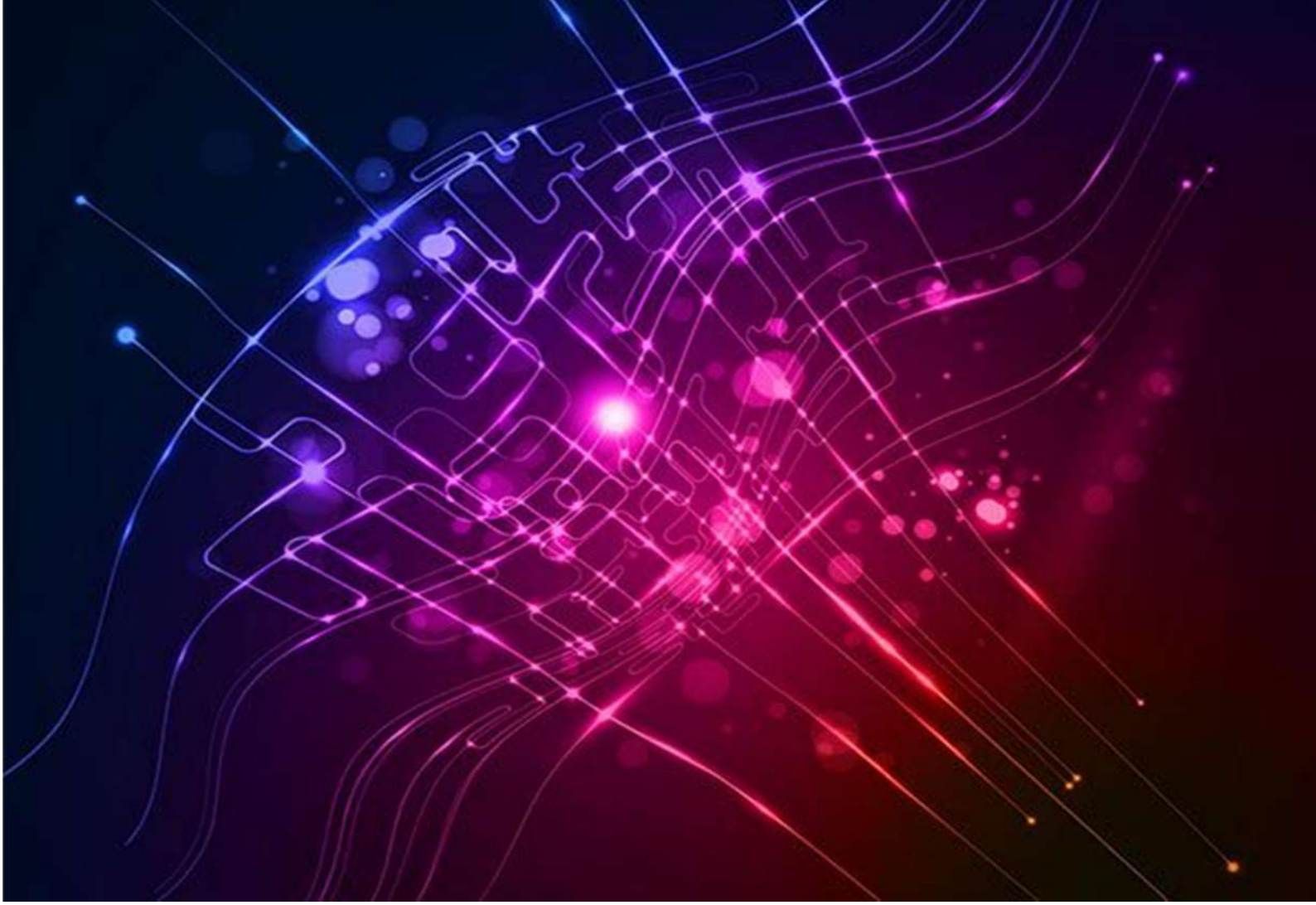
Based on the results and model optimization, it can be concluded that a machine learning approach using MLPClassifier is an effective method for classification of cause of death. Despite the unsuccessful extraction of handwritten text from scanned documents, the results suggest that the model can be improved through fine-tuning with a large dataset of handwritten characters commonly used by doctors. The MLP classifier model trained based on MCCD data set in excel format for the year 2022 showed high accuracy, precision, recall, and F1-score values, indicating that it correctly classified a high percentage of samples. The best parameters for the MLPClassifier were determined to be Hidden layers: (100,100), Alpha: 0.001, Verbose: 10, Learning_rate_init: 0.01, and Solver: 'sgd', with an accuracy of 98.61%.

To facilitate the classification process, a Graphical User Interface was developed, and a module was prepared to enable users to input the cause of death as a CSV file, and the model will predict the corresponding ICD codes and add them to a new column labeled "code" in the CSV file. This can be a valuable tool for administrators, policy planners, researchers, and other professionals in the area of public health.

However, it is important to note that the model's accuracy may be affected by the quality of the input data, and it may require continuous updates and improvements to stay relevant and accurate. Overall, this project provides a promising approach to improving the accuracy and efficiency of cause of death classification.

Reference

i. *Ghulam Mujtaba, Liyana Shuib, Ram Gopal Raj, Retnagowri Rajandram, Khairunisa Shaikh, Mohammed Ali Al-Garadi :"Automated classification of cause of death in verbal autopsy data using machine learning" by Inoue et al. (2018).*
   The paper can be found in the journal "PLoS ONE" and can be accessed at the following link: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0193890
ii. *Organization WHO (1979): Medical Certification of Cause of Death: instructions to physicians on use of international form of medical certification of cause of death*
iii. *Department of Economics and Statistics (2020): Report on Medical Certification of cause of death*



Exploration of land utilization in Kerala- Forecasting

A Machine Learning Approach

Submitted By
Sri. Vijay R., Research Officer

Introduction

In the past two decades, Kerala has experienced significant changes in land use, with a noticeable decline in area used for cultivation, known as net sown area. Non-agricultural land use has gradually increased while agricultural activities have decreased. This conversion of agricultural land for non-agricultural purposes not only poses a threat to food security but also raises ecological concerns. The factors that affect land use are complex, including things like the surrounding environment, topography, climate, and policies, etc. Here, we focus on the impact of different types of land use on the area used for farming. Examining the changes in land use patterns during a specific timeframe allows us to obtain valuable information regarding the present condition of agriculture land utilization.

Predicting time series data can be challenging due to significant fluctuations, changing trends, and limited information. Traditionally, a number of techniques, including univariate Autoregressive, univariate Moving Average, Simple Exponential Smoothing, and in particular Autoregressive Integrated Moving Average with its numerous variations, have been used to predict the next lag of time series data. The other most used forecasting methods, multivariate ARIMA models and vector auto-regression models, generalize univariate ARIMA models and univariate autoregressive models by allowing several evolving variables. However, with the advancements in computational power and the development of more advanced machine learning algorithms, such as deep learning, new methods have emerged for analyzing and forecasting time series data. Deep learning approaches excel at identifying patterns and structures in data, including non-linearity and complexity, in time series forecasting.

In this study, a deep learning approach using neural networks is used to analyze land utilization patterns in Kerala and predict the net sown area across different categories of land use. By analyzing numerical data on past land usage, a model is built to understand the underlying patterns and make predictions about the future proportion of land used for cultivation.

2. Objectives

- To identify the net sown area use changes in Kerala.
- To make future prediction of net sown use area.
- To understand the transformation of land.
- To restrict land use that affects human survival by implementing effective policies and regulations.

3. Methodology and method used

The modeling procedure can be broken down into three simple steps as given below.

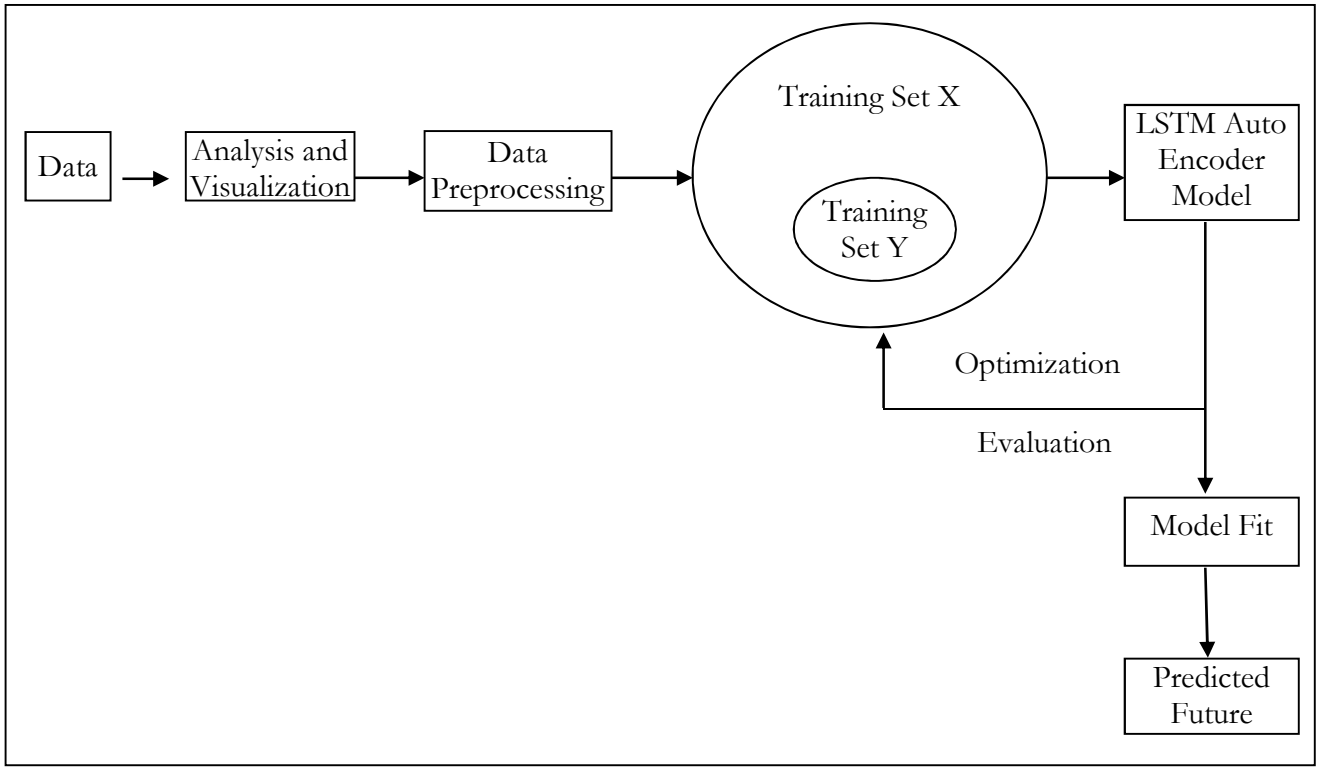
1. Data collection: Gathering the required data from various sources.
2. Data preprocessing: Cleaning and organizing the collected data to make it suitable for analysis.
3. Model training: Using the prepared data to train the model and enable it to learn patterns and make predictions.

The Directorate of Economics and Statistics (DES), Government of Kerala, has published reports containing secondary data on land use patterns in Kerala. This data, spanning from 2005-06 to 2020-21, was utilized to create a neural network model for predicting the future net sown area in the state. Various methods are applied for data preprocessing to ensure that the data is in a suitable format for analysis. For this study, we used the Long Short-Term Memory (LSTM) techniques to build a model for the selected dataset. LSTM is a special case of Recurrent Neural Network (RNN) method in deep learning developed to address the challenges related to the forecasting models.

RNNs are a type of neural network that can process sequences of data. They work by taking input from the current time step and combining it with information from previous time steps, allowing them to capture temporal dependencies in the data. However, traditional RNNs suffer from the "vanishing gradient" problem, where the gradients that flow through the network become too small to be useful for learning long-term dependencies. **LSTMs** were designed to overcome the vanishing gradient problem by introducing a memory cell, which can store information over time and selectively forget or update it based on the input. LSTMs also have separate gates for controlling the flow of information through the cell, allowing them to learn long-term dependencies more effectively.

In short, LSTMs are a powerful type of RNN that excel at capturing long-term dependencies in data. For time series forecasting, different algorithms are available based on the characteristics of the dataset. In this study, we utilized the Encoder-Decoder model, also known as the Autoencoder model, which is a multi-step LSTM model. **Figure 1** serves as a visual representation of the sequential process followed in the study.

Figure 1: Work flowchart



4. Data Set

4.1 Collection

The secondary data on land use area patterns for the period from 2005-06 to 2020-21 was obtained from the official website of the Department of Economics and Statistics, Kerala. The dataset covers a total geographical area of 3,886,287 hectares in the state. The land is classified into thirteen distinct categories, including Forest, Land put to non-agricultural use, Barren and uncultivable land, Permanent pastures and other grazing land, Land under miscellaneous tree crops, Cultivable waste, Fallow other than current fallow, Current fallow, Marshy land, Still water, Waterlogged area, Social forestry, and Net area sown. This classification system enables a comprehensive analysis and categorization of the various land uses within the region.

4.2 Preprocessing

In order to develop an effective model, it is necessary to preprocess and transform the raw data into a suitable format. For this study, we have chosen the LSTM Network as the main model framework due to its ability to effectively handle numerical input. In this part, the original classification of land use area into 13 categories has been reconstructed into 8 basic categories based on the nature of the data. Some classifications have consistently maintained negligible values throughout the entire timeframe and thus remain unchanged. As a result of this data transformation, a dataset consisting of 144 data points (16x8 samples) has been generated. The dataset is structured as follows:

Table 1: Land use area pattern in Kerala

Agriculture year	Barren and uncultivable land	Cultivable waste	Fallow land	Forest area	Still water	Others	Non-agricultural use	Net area sown
2005-06	26457	69165	113263	1081509	74286	19202	369922	2132483
2006-07	26125	90298	128795	1081509	82702	16743	358684	2101431
2007-08	25527	92764	128167	1081509	84829	12904	371558	2089029
2008-09	24931	96193	113714	1081509	92876	11954	376155	2088955
2009-10	22046	98014	122319	1081509	101547	10231	371906	2078715
2010-11	19573	91665	127971	1081509	100111	9777	384174	2071507
2011-12	17552	95437	134726	1081509	107181	9826	399924	2040132
2012-13	16354	96596	132579	1081509	99789	8784	402567	2048109
2013-14	13655	97069	128322	1081509	99673	9239	405826	2050994
2014-15	12952	100676	120070	1081509	100453	8618	419128	2042881
2015-16	13100	99489	125261	1081509	100589	8620	434646	2023073
2016-17	11780	101379	127538	1081499	98343	8332	441934	2015482
2017-18	10894	96491	106983	1081509	98889	8065	443041	2040415
2018-19	10276	96489	102991	1081509	99323	8035	454040	2033624
2019-20	10619	99810	104128	1081509	100160	8100	455897	2026064
2020-21	9530	93975	97008	1081509	100033	8184	460919	2035129

This dataset, derived from the reconstructed land use area classifications, will serve as the basis for training and evaluating the LSTM Network model.

5. Model Construction, Training and Optimization

The LSTM network consists of input, hidden, and output layers and we need to specify parameters such as the number of layers and nodes, learning rate, activation function, loss function, and optimizer. To begin, we construct the LSTM autoencoder model by defining its architecture and parameters. Once the model is set up, we train it using a known training set, which in this case is the original data set. During training, the model learns to encode the input data into a compressed representation and decode it back to reconstruct the original input.

During the training process, evaluation functions are employed to calculate metrics such as loss and accuracy, which help assess the performance of the model. Based on these evaluation results, we can make adjustments to the model's parameters to improve its performance. The optimization phase involves iteratively refining the model by adjusting the parameters, such as the number of layers and nodes, learning rate, activation function, loss function, and optimizer, until we achieve satisfactory results in terms of the model's performance and the quality of data reconstruction.

6. Learning Process Analysis

To understand how the model learns, we regularly check its predictions on the training set at specific epoch intervals. By comparing these predictions over time, we can analyze the changing trends and patterns in the learning process.

7. Tools and Libraries Used

The tool used for building a Neural Network model is Google Colab, which is a Python-based platform for performing various tasks, including machine learning and deep learning. Colab offers a convenient and collaborative environment where Python code can be written and executed interactively. It provides a wide range of libraries and frameworks that are commonly used for machine learning and deep learning tasks. A brief explanation of the libraries and modules used in the project for data handling, model building, and visualization are detailed below. When creating an LSTM autoencoder model, you can utilize several tools and libraries to simplify the implementation process. Here are some commonly used ones:

Python: Python is a popular programming language for machine learning and deep learning tasks. It offers a wide range of libraries and tools that can be used for building LSTM autoencoder models.

NumPy: It's a library that helps with numerical computations and provides functions to work with large arrays of numbers efficiently.

TensorFlow and Keras: They are libraries used for building and training machine learning models. TensorFlow is the underlying framework, and Keras is a user-friendly interface for TensorFlow. The code uses them to create neural network models.

Pandas: It's a library for data manipulation and analysis. It provides data structures to handle structured data and perform operations like filtering and merging.

Matplotlib: It's a library for creating visualizations. It offers functions to create different types of plots like line plots and bar plots.

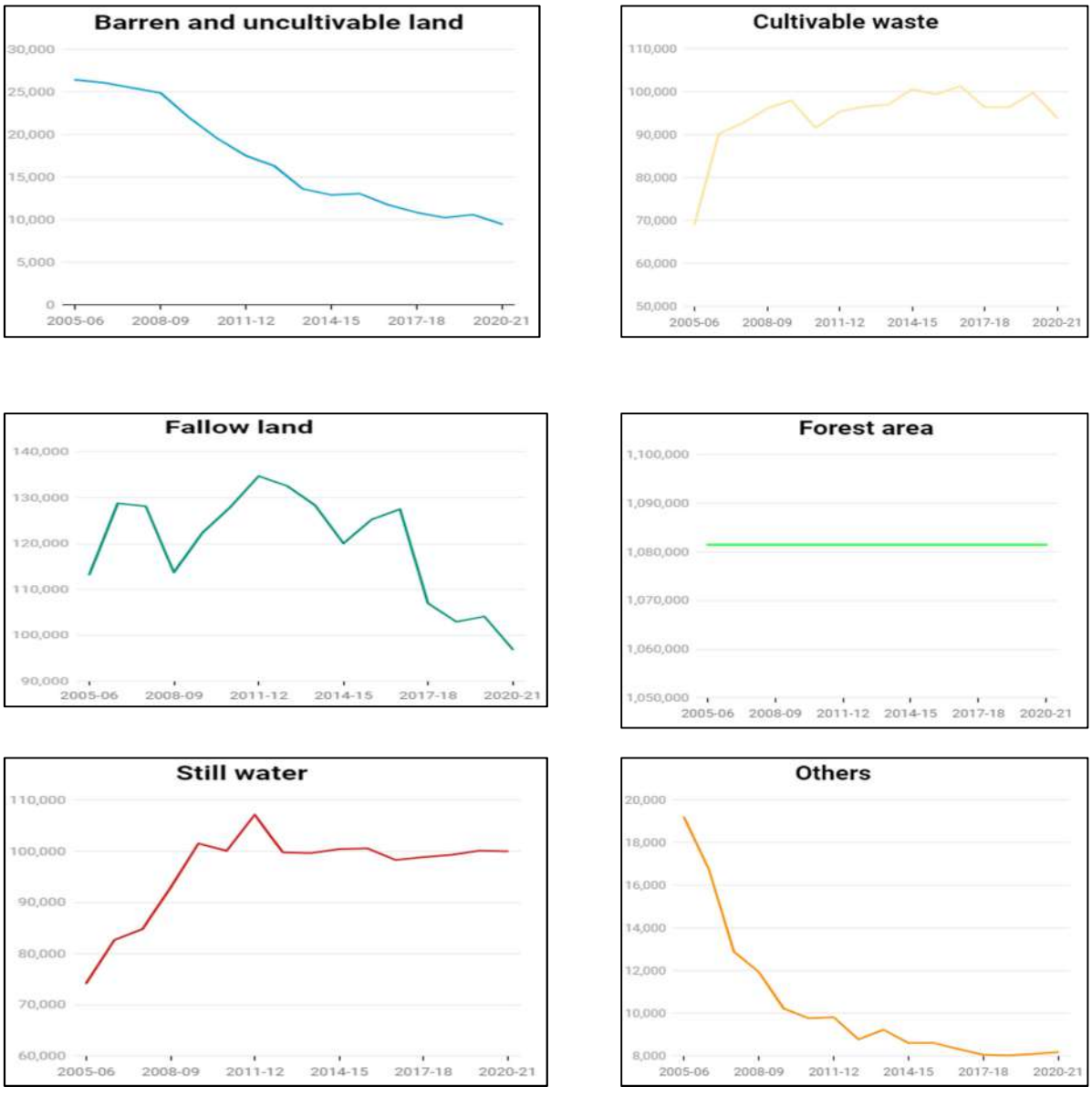
Scikit-learn: It's a machine learning library that provides tools for data preprocessing and model evaluation. The code uses the StandardScaler class to scale or normalize the data.

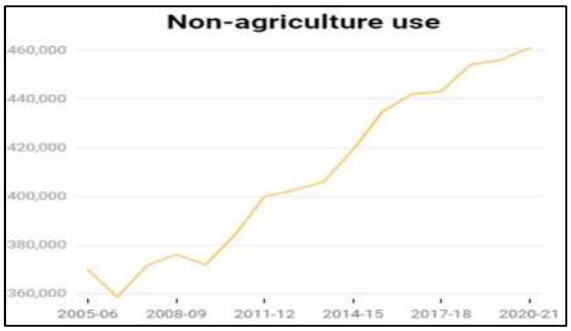
Seaborn: It's a library that works on top of Matplotlib and helps create visually appealing statistical graphics.

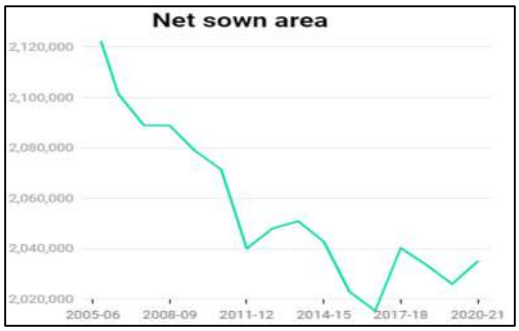
Datetime: It's a module for working with dates and times. The code imports the datetime class from this module.

8. Data visualization

Figure 2: Graphical format of dataset







9. Model working rule

In the model architecture, the working rule involves using the previous three steps of the data to predict the next step. The objective is to predict the net area sown for the next time step. The model takes the land use data from three consecutive time steps as input and trains to learn the patterns and dependencies within the data. It then uses this learned information to make predictions for the net area sown in the next time step. For instance, to predict the net area sown for the year 2008-09, the model would utilize the land use data from the previous three years (2005-06, 2006-07 and 2007-08). It would analyze the patterns and relationships between the lands use categories over these three years and generate a prediction for the net area sown in 2008-09. By applying this working rule, the LSTM autoencoder model aims to capture the temporal dependencies and trends in the data, allowing for accurate predictions of the net area sown based on the previous three steps of land use information. Here is the table representation of the working rule mentioned above.

Table 2: Working rule of the model

Training Set X									Training Set Y
Agriculture year	**Barren and uncultivable land**	**Cultivable waste**	**Fallow land**	**Forest area**	**Still water**	**Others**	**Non-agricultural use**	**Net area sown**	**Net area sown**
2005-06	26457	69165	113263	1081509	74286	19202	369922	2132483	
2006-07	26125	90298	128795	1081509	82702	16743	358684	2101431	
2007-08	25527	92764	128167	1081509	84829	12904	371558	2089029	
2008-09	24931	96193	113714	1081509	92876	11954	376155	2088955	2088955
2009-10	22046	98014	122319	1081509	101547	10231	371906	2078715	2078715
2010-11	19573	91665	127971	1081509	100111	9777	384174	2071507	2071507
2011-12	17552	95437	134726	1081509	107181	9826	399924	2040132	2040132
2012-13	16354	96596	132579	1081509	99789	8784	402567	2048109	2048109
2013-14	13655	97069	128322	1081509	99673	9239	405826	2050994	2050994
2014-15	12952	100676	120070	1081509	100453	8618	419128	2042881	2042881
2015-16	13100	99489	125261	1081509	100589	8620	434646	2023073	2023073
2016-17	11780	101379	127538	1081499	98343	8332	441934	2015482	2015482
2017-18	10894	96491	106983	1081509	98889	8065	443041	2040415	2040415
2018-19	10276	96489	102991	1081509	99323	8035	454040	2033624	2033624
2019-20	10619	99810	104128	1081509	100160	8100	455897	2026064	2026064
2020-21	9530	93975	97008	1081509	100033	8184	460919	2035129	2035129
2021-22									**?**

10. Hyper Parameters Used

The model begins with random parameters and undergoes fine-tuning for improved performance. We trained the model using the following hyperparameters:

- LSTM layers with 64 and 32 units respectively.
- Dropout regularization to prevent overfitting.
- Adam optimizer for adaptive learning rate.
- Mean squared error (MSE) loss function for training.
- 60 epochs for training iterations.
- Batch size of 24 data points per batch.
- Divided into 13 iterations per epoch.

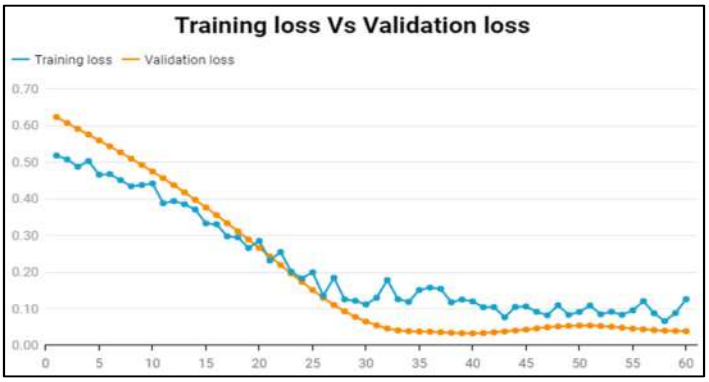
Here are simplified definitions of each hyperparameter used in the model.

- **LSTM layers:** Memory cells in the model that captures long-term dependencies in the data.
- **Dropout regularization:** Technique to prevent overfitting by randomly ignoring some input units during training.
- **Adam optimizer:** Algorithm that adjusts the learning rate for each parameter during training.
- **Mean squared error (MSE) loss function:** Measures the average squared difference between predicted and actual values.
- **Epochs:** Number of times the model goes through the entire dataset during training.
- **Batch size:** Number of data samples processed together in each training iteration.
- **Iterations:** Number of times the model processes the dataset in one epoch.

11. Training loss and Validation loss

The training loss assesses how well the model reconstructs the training data, whereas the validation loss assesses how well it performs on new data. To achieve accurate reconstruction and good generalization, we strive to minimize both losses. Monitoring the validation loss helps in identifying instances of overfitting, and modifying the model or stopping training can improve reconstruction quality. The following graph depicts the training and validation loss during the training procedure. It shows how the loss of the model lowers over time, showing increasing performance and convergence.

Figure 3: Loss curve



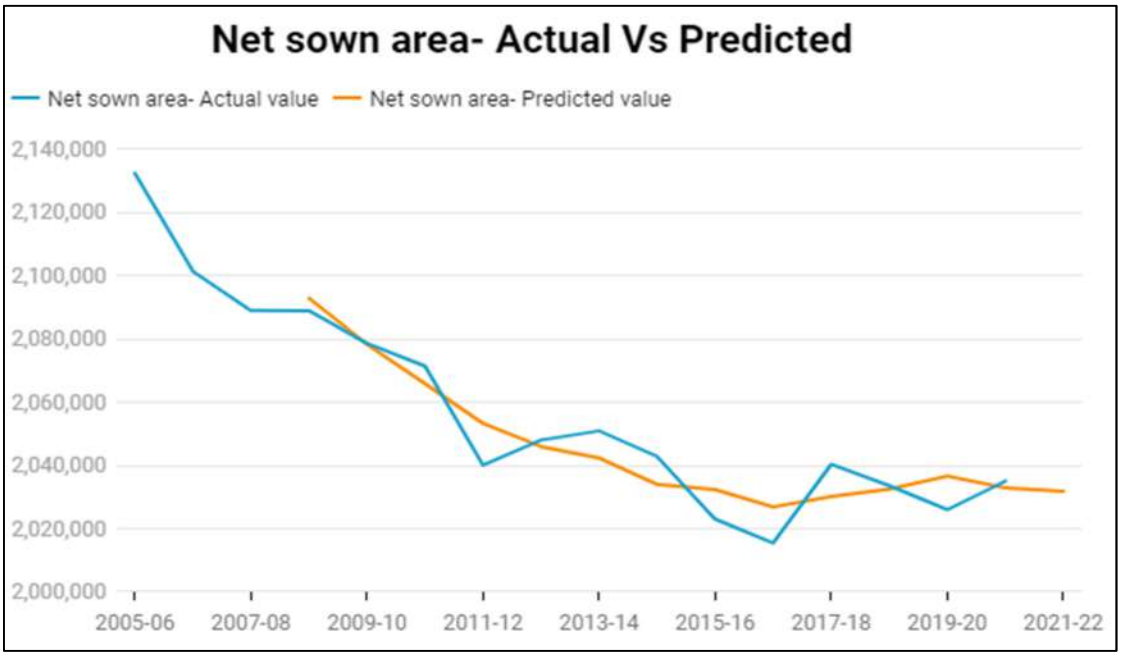
12. Results with Inference

The hyperparameters used considerably enhanced the model's training efficiency, resulting in accurate and efficient data analysis. The loss (mse) demonstrates that the model learns by defining a wide value range and making precise modifications. Overall, the model is thought to be valid and effective. The prediction results are presented in the table below, and a visual representation is provided by the following graph.

Table 3: Predicted values by the model

Agriculture year	Net sown area- Actual value	Net sown area- Predicted value
2005-06	2132483	
2006-07	2101431	
2007-08	2089029	
2008-09	2088955	2092886
2009-10	2078715	2078375.2
2010-11	2071507	2065916.9
2011-12	2040132	2053325.1
2012-13	2048109	2046021.9
2013-14	2050994	2042406
2014-15	2042881	2034051.8
2015-16	2023073	2032418.8
2016-17	2015482	2026918.9
2017-18	2040415	2030204.1
2018-19	2033624	2032539.2
2019-20	2026064	2036676.2
2020-21	2035129	2032929.2
2021-22		2031854.9

Figure 4: Predicted Vs Actual value curve



The dataset used for the model is small, limiting its ability to learn complicated patterns. As a result, based on the rules it has learned, the model creates predictions. To enhance the accuracy of future predictions, it is advisable to expand the dataset by including more features and forecasting each feature value. The projected outcomes show a large decrease in the share of land used for cultivation in the agricultural year 2021-22. The anticipated cultivable land area for 2021-22 is 2031854.9 hectares. This reduction in land usage could potentially have adverse effects on food production, raising concerns about meeting the growing demand.

Conclusion

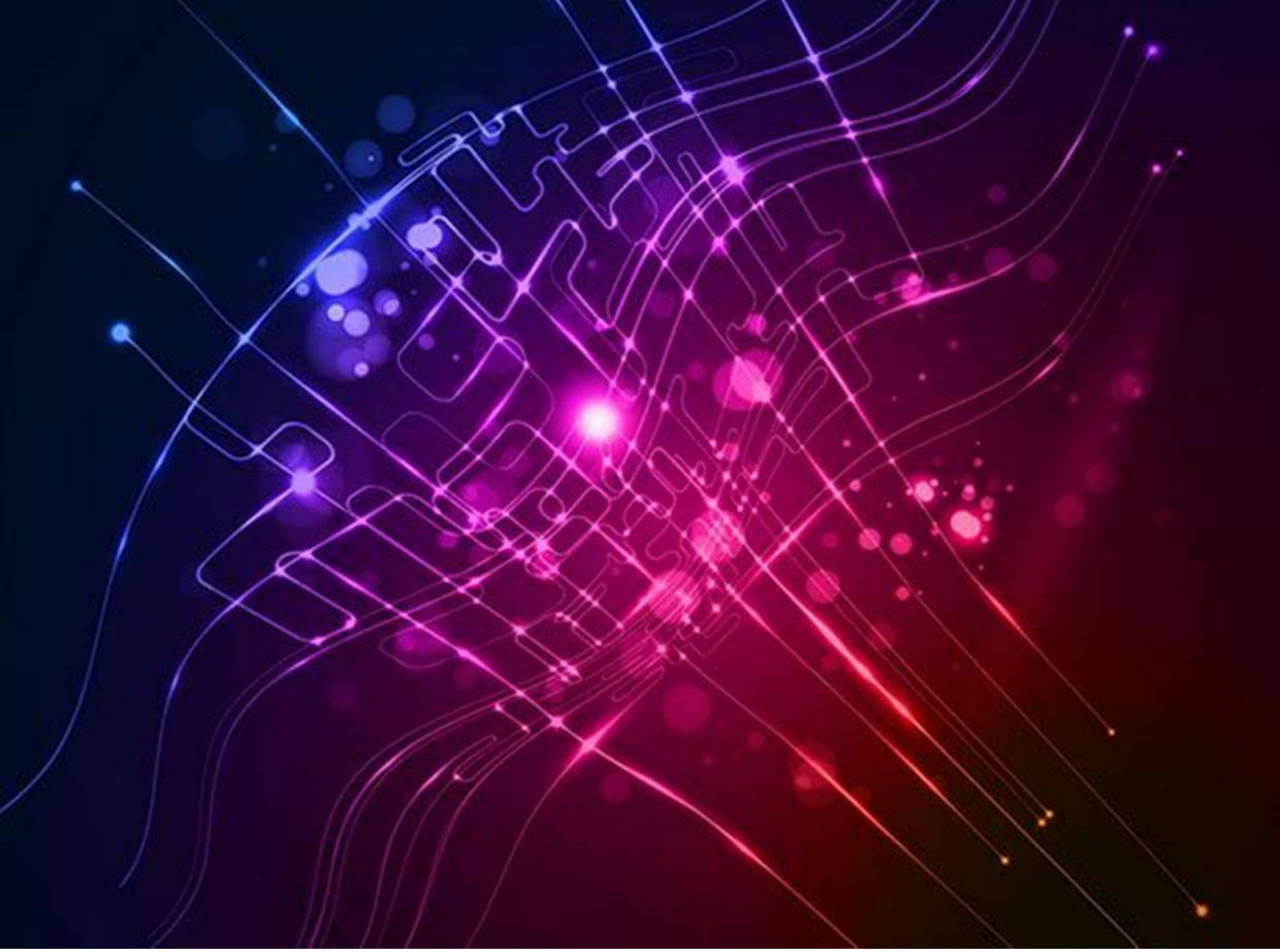
This study emphasizes the advantages of forecasting land use data using deep learning techniques. The algorithm can more accurately forecast the net sown area based on other land uses by examining a broader and more diverse dataset. Although the model has revealed some patterns, it's critical to keep in mind that real-world situations are intricate and dynamic. By experimenting with various modifications, we may optimize the hyperparameters to improve the performance of the model. Altering the model parameters and adding more affecting variables to the dataset are additional steps in enhancing the findings. The study was able to effectively forecast the outcomes to a significant extent. However, the remaining goals were temporarily put on hold because there weren't enough feature values available.

Reference:

1. *Imad Basheer and M. Hajmeer: 2001: Artificial Neural Networks- fundamentals, computing, design and application.*

2. *Sima Siami-Namini, Neda Tavakoli, Akbar Siami Namin: 2018- A Comparison of ARIMA and LSTM in Forecasting Time Series- 17th IEEE International Conference on Machine Learning and Applications.*

   Various deep learning architectures, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and generative adversarial networks (GANs), have been applied on various numerical historical data and demonstrated the superiority of deep learning methods over traditional machine learning methods in terms of accuracy in these papers.



Building Construction Cost and Index Prediction in Kerala

A Machine Learning Approach

Submitted By
Smt. Dhanya A., Deputy Director

INTRODUCTION

The construction industry is a vital sector in any economy, and construction cost estimation is an essential aspect of the industry. In Kerala, the building construction cost index (BCCI) and the construction cost are significant determinants of the viability of construction projects. Building construction is a complex and multi-disciplinary process that requires the integration of various factors such as labor, materials, and time. The cost of construction is an important aspect of the process, as it affects the feasibility of the project and the return on investment for the stakeholders. In recent years, the building construction industry in Kerala has seen significant growth, with an increasing demand for new construction projects. However, the estimation of construction costs can be challenging, as it is influenced by many variables such as inflation, changes in market conditions, and advances in construction technologies. The traditional methods of estimating construction costs are often time-consuming, inaccurate, and subjective.

In order to address these challenges, the field of construction cost prediction has seen significant progress in recent years. The integration of machine learning (ML) techniques in construction cost estimation can provide a more accurate and efficient solution. Machine learning techniques have been used to develop cost prediction models that can provide more accurate and reliable estimates of construction costs. The objective of this project is to utilize these advanced techniques to develop a machine learning model that can be used to forecast the Building Construction Cost Index (BCCI) and the cost of building construction in Kerala. The project also aims to develop a graphical user interface (GUI) that can be used by stakeholders to predict the cost of construction for future projects. By providing an accurate and reliable tool for construction cost prediction, this project aims to support the growth of the construction industry in Kerala and provide valuable insights for stakeholders in the industry.

The results of this project will provide valuable insights into the potential of ML in construction cost estimation and will benefit the construction industry in Kerala.

OBJECTIVE

The objective of this study is to utilize Machine Learning techniques to predict the future Building Construction Cost Index (BCCI) and building construction cost in Kerala. This will help in providing better cost estimates for builders and contractors, thus improving the efficiency of the construction industry. The study will also develop a graphical user interface (GUI) for ease of use by construction professionals. The results of this study will be useful for future reference and as a starting point for further research in this field.

METHODOLOGY

i. Data Collection:

Building material price and Labour wage rates from 2011-12 to 2021-22 were collected from Department of Economics and Statistics.

ii. Data Preparation:

Districts in Kerala were given codes from Thiruvananthapuram to Kasaragod as 1 to 14 respectivey. Then computed construction cost from building material price, labour charges and other charges collected. After that prepared the weighing diagram and the construction cost index for all districts.

In the building construction cost index data, the variables are year, district code and BCCI. There are no null values in this dataset.

In the building construction cost data, the variables are year, district code, area in square feet, material costs, labour costs, other charges and the total construction cost. There are no null values in this dataset.

iii. Model Selection:

In this study, the goal was to predict building construction costs and cost indices using machine learning techniques. After exploring various models, multiple linear regression was selected as the most appropriate model for this problem.

Multiple linear regression is a statistical method that models the linear relationship between a dependent variable (building construction cost or cost index) and independent variables (features such as location, materials cost, labour costs, etc.). The model estimates the coefficients for each feature, which can then be used to make predictions based on new data.

The equation for multiple linear regression can be represented as:

$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n$

where:

y is the dependent variable (the one being predicted)

$X_1, X_2, ..., X_n$ are the independent variables (the predictors)

β_0 is the intercept term (the value of y when all X's are 0)

$\beta_1, \beta_2, ..., \beta n$ are the coefficients, representing the strength and direction of the relationship between each independent variable and the dependent variable.

The goal of multiple linear regression is to estimate the values of the coefficients $\beta_0, \beta_1, \beta_2, ..., \beta_n$ that best fit the observed data. These estimates can be obtained using various techniques, such as ordinary least squares or maximum likelihood estimation.

The choice of multiple linear regression was based on several factors, including:

- Simplicity: Multiple linear regression is a simple and well-understood model that can be easily interpreted and applied to this problem.
- Linearity: The relationship between the dependent variable and the independent variables was assumed to be linear, which is a reasonable assumption for this problem.
- Availability of data: Adequate data was available to train the model and validate its performance.
- Performance: In initial experiments, multiple linear regression was found to perform well in terms of accuracy and stability compared to other models such as decision trees, random forests or ANN.

In conclusion, multiple linear regression was selected as the model for prediction in this study, as it offered a good balance between performance and interpretability, and was well-suited to the problem and available data.

iv. Model Training and Validation:

The multiple linear regression model was trained on a dataset of building construction costs and cost indices. The dataset was divided into two parts: 80% for training and 20% for validation.

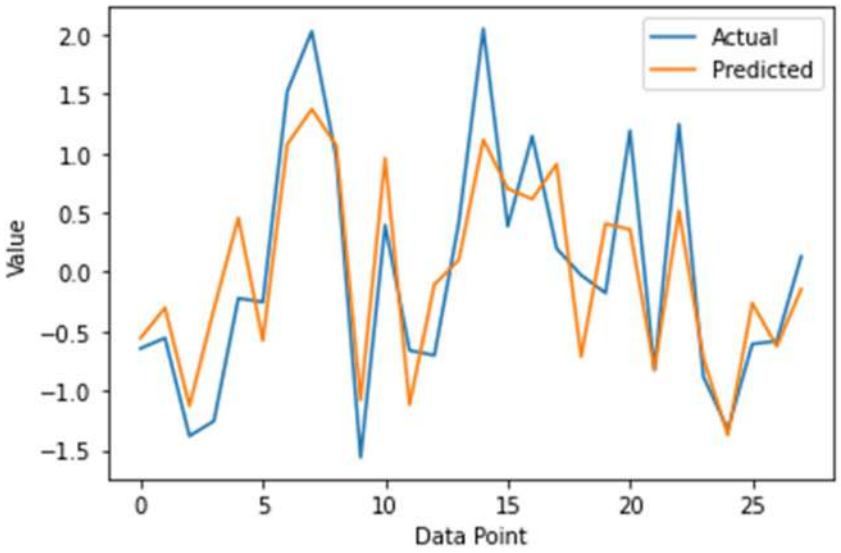
The training process involved estimating the coefficients for each feature in the model using the training data. This was done using the ordinary least squares method, which minimizes the sum of the squared residuals between the observed and predicted values.

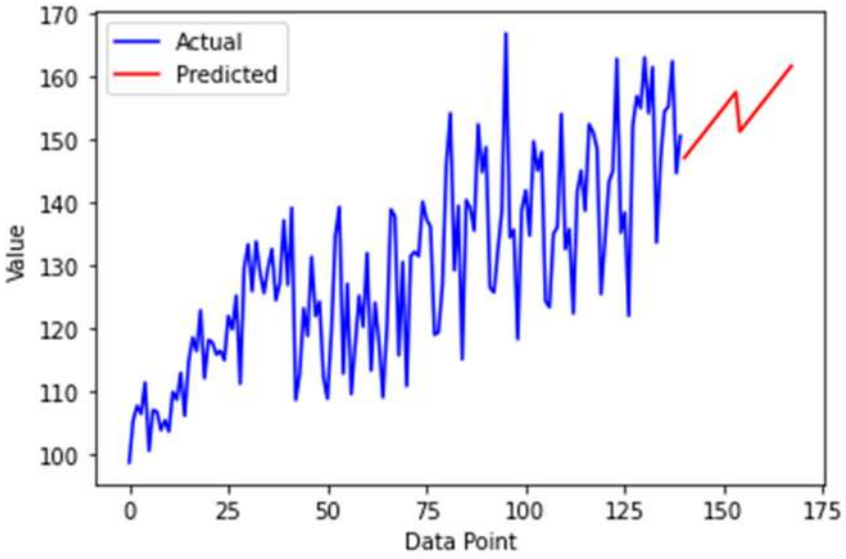
Once the model was trained, its performance was evaluated on the validation data using two metrics: mean squared error (MSE) and coefficient of determination (R-squared). The MSE measures the average difference between the observed and predicted values, while the R-squared measures the proportion of variance in the dependent variable that can be explained by the independent variables.

a. Building Construction Cost Index (BCCI):

The results showed that the multiple linear regression model had a MSE of 0.2667 and an R-squared of 0.733 on the validation data, indicating that the model had good performance in terms of both accuracy and goodness-of-fit. These results confirmed that multiple linear regression was a suitable model for this problem and provided a good basis for making predictions on new data.

In conclusion, the multiple linear regression model was trained and validated on the data, and its performance was evaluated using two appropriate metrics. The results showed that the model had good performance in terms of accuracy and goodness-of-fit, and was ready for testing on new data.

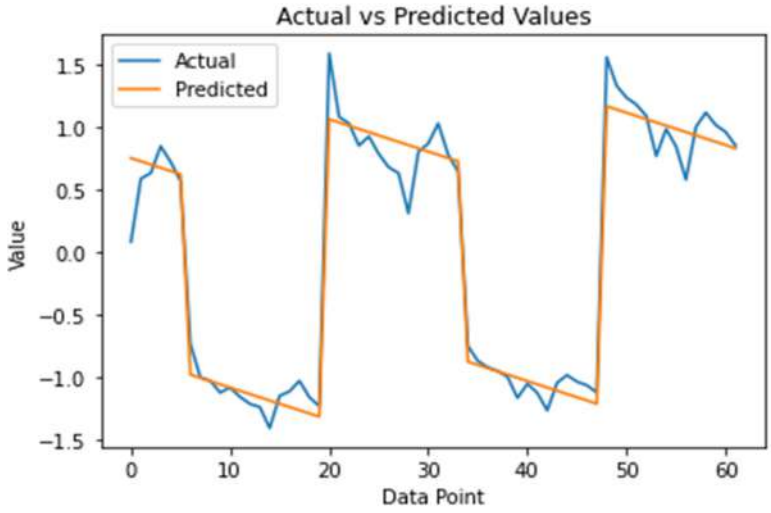


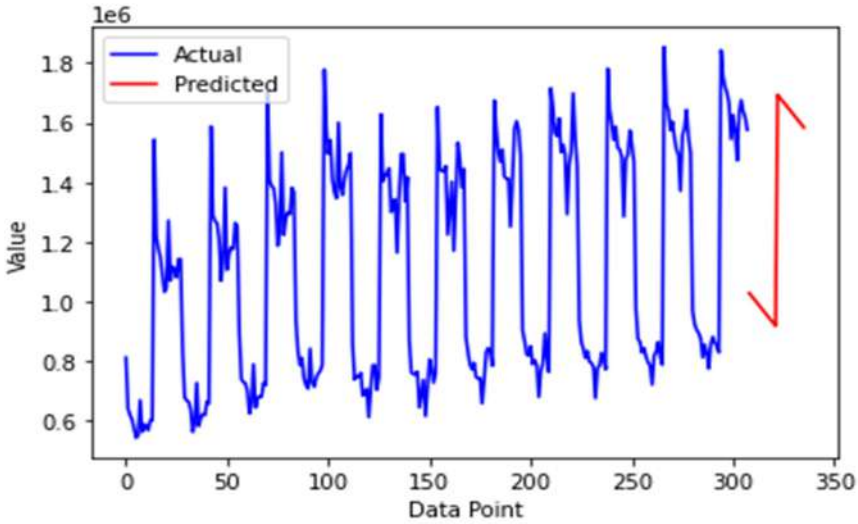


b. Building Construction Cost:

The model was trained on 80% of the data and tested on the remaining 20%. The mean squared error (MSE) and R-squared metrics were used to evaluate the model's performance on the testing set.

The mean squared error (MSE) on the testing set was found to be 0.00688, and the R-squared was 0.918. These values indicate that the model has a good fit to the data, with a small difference between the actual and predicted values (MSE) and a high proportion(91.8%) of the variance in the dependent variable explained by the independent variables in the model (R-squared). This is a high value, indicating that the model has a good fit to the data.

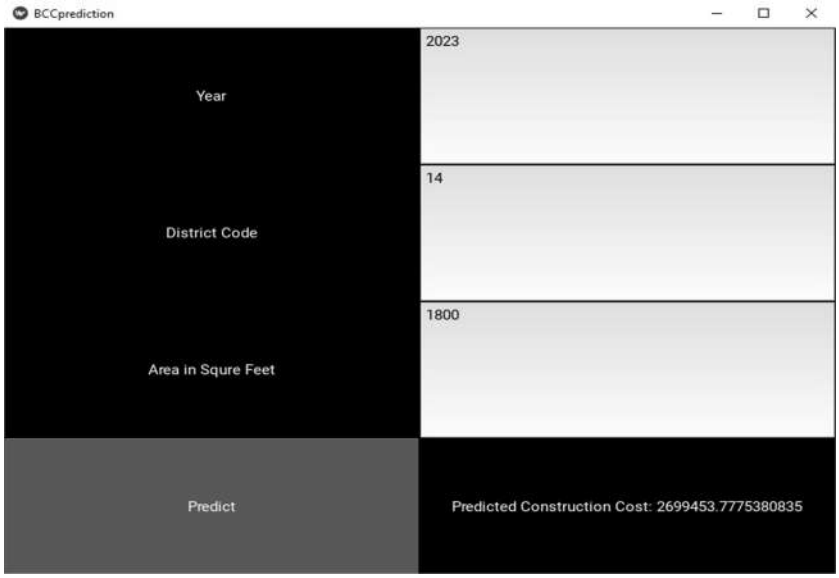




v. Model deployment:

The result of 2938154.66 that got from testing the model with new values such as year - 2022, district code - 14 (Kasaragod), and area in square feet - 2000 indicates that the model has made a prediction for the dependent variable based on the given independent variables. This prediction is a good one, suggests that the model is likely to make accurate predictions in similar scenarios.

Also developed a Graphical User Interface(GUI) for predicting the Building Construction Cost using 'tkinter' package and developed a mobile application using 'kivy' package in python.



CONCLUSION:

The results of the model testing demonstrate that the developed machine learning model has good performance in terms of both accuracy and fit to the data.

Here also developed a graphical user interface (GUI) and mobile application for ease of use by construction professionals. The results of this study will be useful for providing better cost estimates for builders and contractors, thus improving the efficiency of the construction industry.

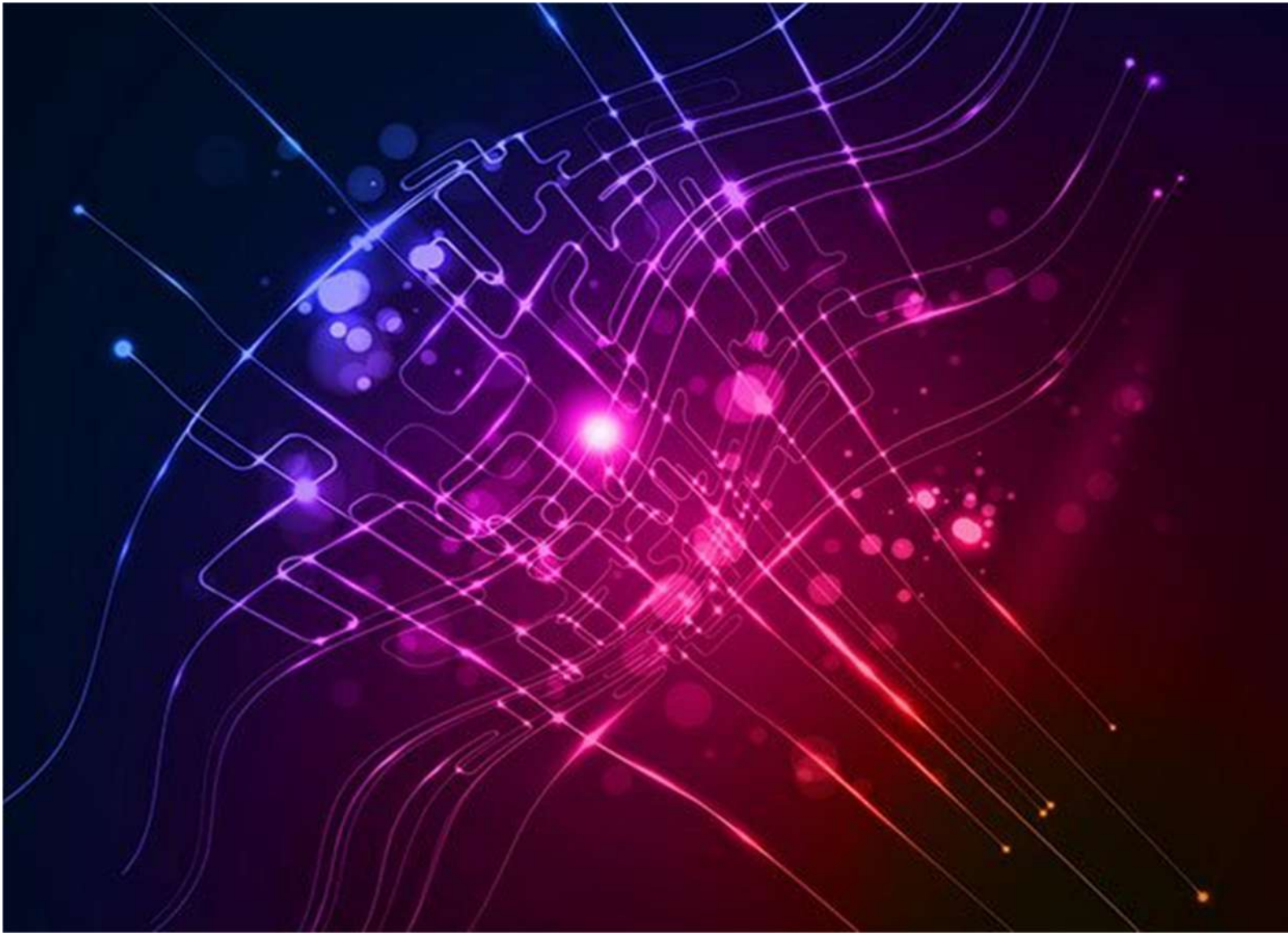
Overall, the results of this study fulfil the objective of utilizing machine learning techniques to predict the future BCCI and building construction cost in Kerala. The model's performance and the developed GUI provide a starting point for future reference and further research in this field.

Future work could include expanding the dataset to include additional factors that may influence construction costs, such as the size and complexity of the building or the economic conditions at the time of construction. Additionally, further research could be conducted to compare the performance of different machine learning models and to evaluate the long-term effectiveness of the tool in predicting construction costs.

Predictions like this can be made and published on the website of Department of Economics and Statistics in future.

REFERENCE:

- *Website of*
    1. *Department of Economics and Statistics, Kerala*
    2. *Department of Public Works Kerala*
    3. *Kerala State Housing Board*
    4. *Central Public Works Department*
- *International Journal of Civil Engineering and Technology (IJCIET), Construction costs in affordable housing in kerala: relative significance of the various elements of costs of affordable housing projects, Manoj P K.*
- *International Journal of Earth Sciences and Engineering, Study on factors Affecting the performance of construction projects and developing a cost prediction model using ANN, Melba Alias, Dhanya R and Gangapathy Ramaswamy, October, 2015.*



To predict daily prices of essential commodities

Submitted By
Sri. Abhilash K.V., Research Officer

Introduction

Predicting the daily price of an essential commodity can be a valuable tool for businesses, investors, and governments. The price of essential commodities such as oil, rice, pulses can have a significant impact on the economy and can be influenced by a variety of factors such as supply and demand, economic conditions, and geopolitical events.

Accurate predictions of the daily price of an essential commodity can help businesses make informed decisions about production, pricing, and purchasing. It can also help investors make informed decisions about buying and selling stocks or commodities. Governments may also use price predictions to make policy decisions and to manage the impact of commodity price changes on the economy.

There are a number of statistical methods that can be used to predict the daily price of an essential commodity, including the LSTM Model (Long Short -Term Memory). In this approach, a statistical model is built using data on the prices of the commodity over time and a number of independent variables that are believed to influence the price. The model is then used to make predictions about the price of the commodity in the future.

Objectives

- To create a model that will be able to analyze the relationship between district wise monthly average price data of some important essential commodities.
- The level of accuracy of this model may help the government to make decisions to ensure the buffer stocks for commodities in the market and minimize the artificial shortages.
- To prepare charts and animated maps for the better visualization of data.

Literature Survey

There has been a significant amount of research on predicting the monthly average price of essential commodities. Here are a few examples of studies that have used regression analysis to predict commodity prices:

- Predicting Commodity Prices Using a Hybrid Neural Network Model" (Wang et al., 2013) - In this study, the authors develop a hybrid neural network model to predict the prices of four commodities (crude oil, natural gas, corn, and soybeans) over a five-year period. They find that the model is able to make accurate short-term and medium-term predictions for all four commodities.
- Commodity Price Forecasting: A Review of the Literature" (Bilal and Sun, 2018) - This review paper surveys the literature on commodity price forecasting and discusses the various statistical and machine learning methods that have been used to predict prices. The authors also discuss the challenges and limitations of commodity price forecasting and suggest directions for future research.

Proposed Methodology

To predict the daily price of an essential commodity, use a LSTM (Long Short-Term Memory) model.

LSTM is a type of Recurrent Neural Network (RNN) architecture specifically designed to handle sequential data, such as time series, natural language processing, and speech recognition.

To build a regression model to predict the daily average price of an essential commodity, collect data on the prices of the commodity over time. This collected data includes the month, the year, and the price of the commodity for each month.

Analysis

We will use the following techniques to analyze the data and make the price predictions:

- Time series analysis to identify trends and patterns in the historical price data
- Using LSTM Model (Long Short -Term Memory) to develop the statistical model for forecasting future prices

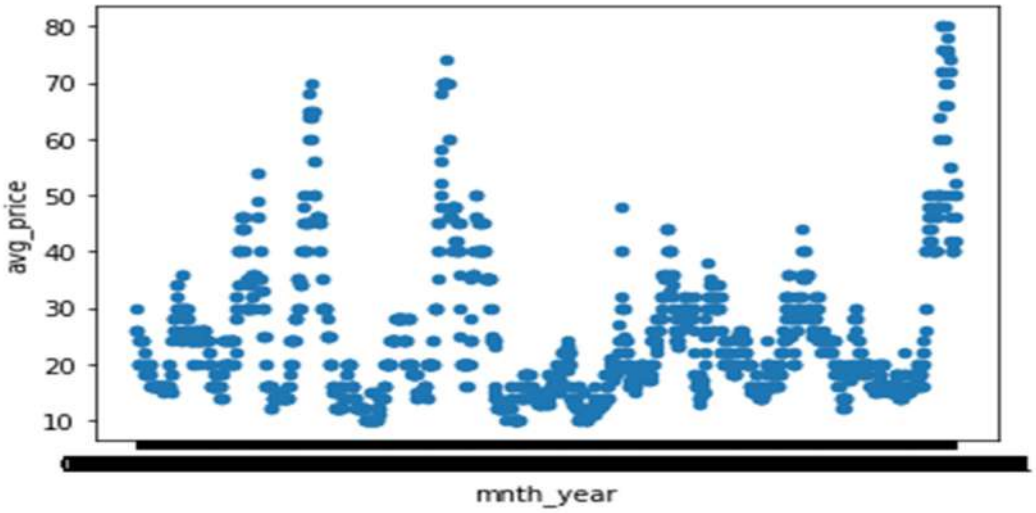
Daily Price Data (Jan 2015- Jan2022)

distid	itemcode	mnth_year	avg_price
01	A001	01-2015	30.00
01	A001	01-2016	30.17
01	A001	01-2017	30.96
01	A001	01-2018	35.00
01	A001	01-2019	34.88
01	A001	01-2020	34.20
01	A001	01-2021	33.00
02	A001	02-2015	29.91
02	A001	02-2016	30.61
02	A001	02-2017	35.41
02	A001	02-2018	35.00
02	A001	02-2019	34.50
02	A001	02-2020	33.50
02	A001	02-2021	31.57
03	A001	02-2022	37.00

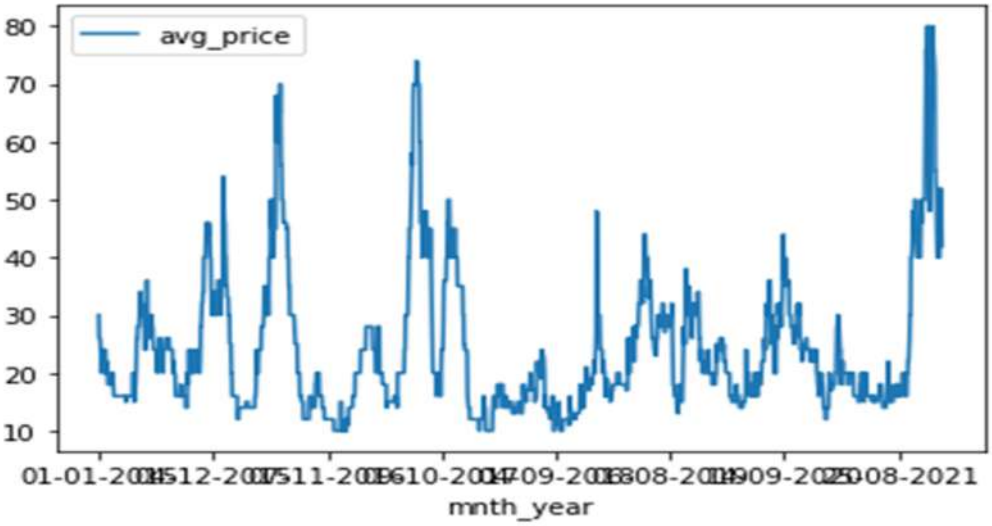
Districts Name and Codes

Distid	District_name
01	Kasargod
03	Wayanad
04	Kozhikkode
06	Palakkad
08	Ernakulam
14	Thiruvananthapuram
Item Names and Codes	
itemcode	item_name
A001	Rice(red) - Matta
A008	Rice(white)- Andhra Vella
B006	Tur Dhall
D006	Vanaspathi (Dalda)
E002	Chillies dry
F003	Potato
G001	Onion Big
G014	Tomato

Scatter Plots



Line Plots



LSTM (Long Short-Term Memory)

data.shape - (85595,4)

Data columns (total 4 columns):

#	Column	Non-Null Count	Dtype
0	distid	89595 non-null	int64
1	itemid	89595 non-null	object
2	mnth_year	89595 non-null	object
3	avg_price	89595 non-null	float64

```
# G014-Tomato
options = ['G014']
# selecting rows based on condition
data = data.loc[ (data['distid'] == 1)  &
data['itemid'].isin(options)]
df=data[['mnth_year','avg_price']]
df.shape
(1846, 2)
```

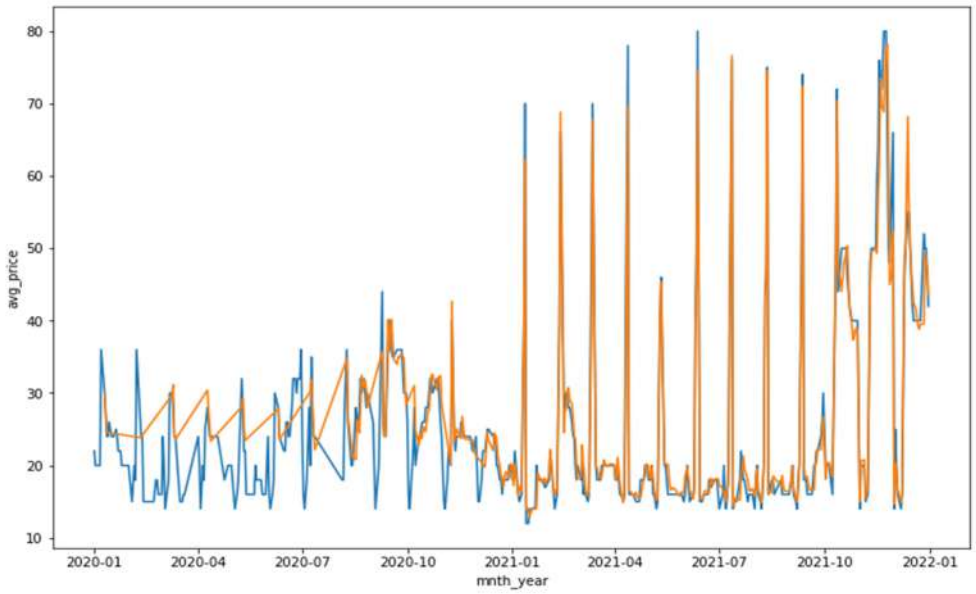
In this model with a batch size of 512 and 400 epochs, the mean absolute error achieved is 1.771214348561055.

- In an LSTM model, batch size and epochs are two important hyper parameters that need to be set when training the model.
- Batch size refers to the number of samples that are processed in one forward/backward pass. It is a hyper parameter that determines how many samples are processed at a time before the weights of the model are updated. A larger batch size will reduce the number of weight updates, but also increases the memory requirements and can slow down the training process.
- Epochs refer to the number of times the entire training dataset is passed through the model. One epoch means that the model has seen all the samples in the training dataset once. In each epoch, the model uses the training data to update its weights. Increasing the number of epochs allows the model to see more of the training data, which can improve the model's performance, but also increases the training time.

The mean absolute error can be used to evaluate the performance of an LSTM model.

The mean absolute error (MAE) is a measure of the difference between the predicted values and the true values in a regression problem. In the case of an LSTM model with a mean absolute error of 1.771214348561055, it indicates that on average, the predictions made by the model are off by approximately 1.77 units (the units depend on the scale of the target variable). A lower mean absolute error indicates that the model is making more accurate predictions.

Actual - Predicted (LSTM)

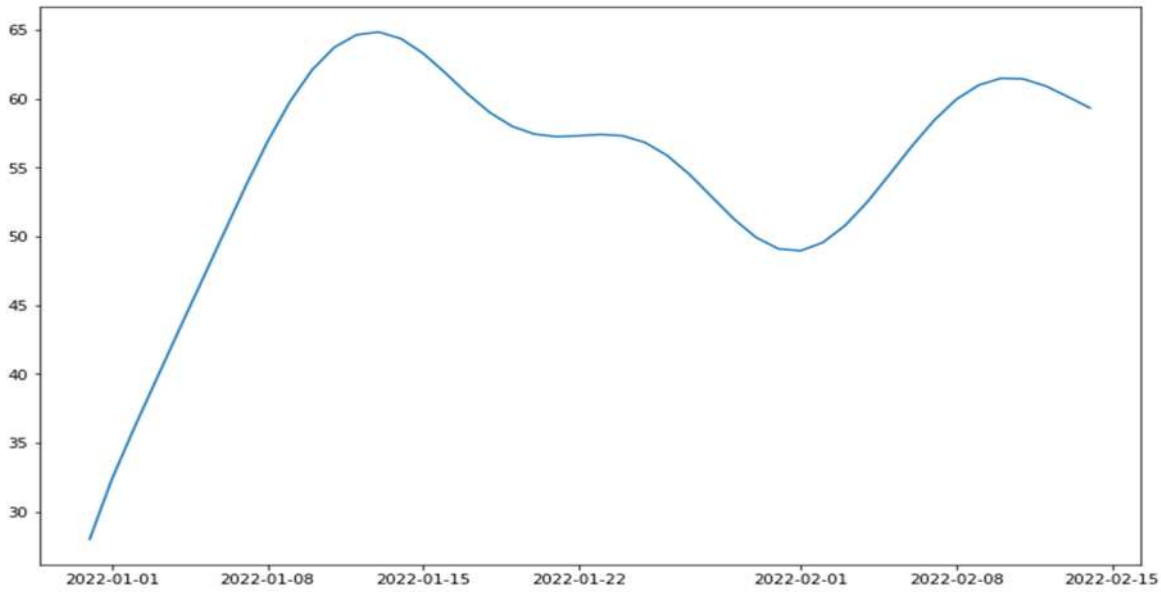


Future Prediction

Look Back - 90 days

No . of Future Predictions - 45 days

[[28.]
[32.36255264]
[36.07043266]
[39.60783195]
[43.12387085]
[46.62206268]
[50.12095261]
[53.60870361]
[56.889431]
[59.77647781]
[62.09675598]
[63.73423004]
[64.64855194]
[64.85329437]
[64.36707687]
[63.30108643]
[61.87174606]
[60.35391617]
[59.0037384]
[58.00448227]
[57.43686295]
[57.25453186]
[57.31728363]



The "look back" in an LSTM model refers to the number of time steps that the model uses as input to make a prediction. The look back is an important hyper parameter that needs to be set when training an LSTM model

Expected Outcome

The expected outcome of predicting the daily price of an essential commodity is to have an estimate of the expected price of the commodity for the coming month. This prediction can be used for a variety of purposes, such as:

Planning and budgeting : A consumer or a business that relies on the commodity, knowing the expected price can help you plan and budget for your expenses.

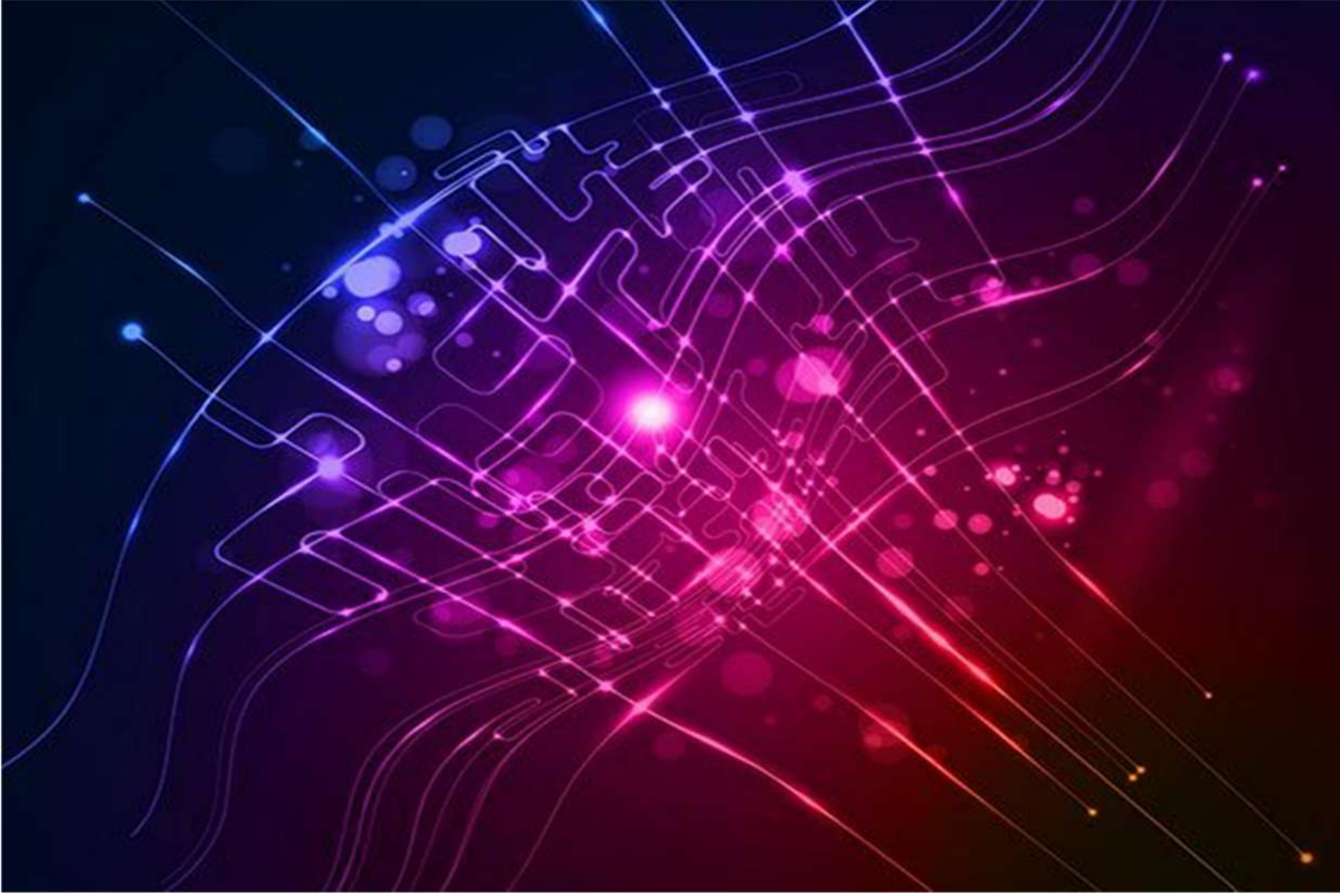
Investment : An investor, knowing the expected price can help you make informed decisions about whether to buy or sell the commodity.

Overall, the expected outcome of predicting the monthly average price of an essential commodity is to have a better understanding of the market and make more informed decisions.

Conclusion

General conclusion about predicting the monthly average price of an essential commodity is a tedious task, as the specific commodity and market conditions will play a large role in the accuracy and usefulness of any predictions. However, some general points that could be made include:

- Accurate price predictions can be very valuable, as they can help Government, businesses and investors make informed decisions and manage risks. However, it is important to remember that no prediction is ever 100% certain, and there is always a degree of uncertainty and risk involved.
- It is important to regularly review and update predictions as new information becomes available and market conditions change. This can help ensure that your predictions remain relevant and accurate.



Forecasting of Consumer Price Index in Kerala

A Machine Learning approach

Submitted By
Smt. Maya R., Deputy Director

Abstract

This research paper presents a comparative analysis of two popular machine learning models, namely the Auto Regressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM), for the purpose of forecasting Consumer Price Index (CPI) values. The study uses historical CPI data from a particular region and employs both ARIMA and LSTM models to predict the CPI values for upcoming years. The accuracy of the models is measured using the Mean Squared Error (MSE) metric, and the results show that the LSTM model is more efficient than the ARIMA model, with an MSE of 1.77 compared to 3.27 for SARIMA. The study thus provides a valuable comparison of these two popular models and highlights the potential benefits of using LSTM in forecasting time series data for CPI values.

INTRODUCTION

The Consumer Price Index (CPI) is a measure of the average change in prices of goods and services consumed by households over time. It is used to track changes in the cost of living and is calculated by comparing the price of a basket of goods and services in a base period to the current period. The basket of goods and services is designed to represent what households typically purchase. The CPI is used for various purposes including adjusting wages, pensions, DA and other payments to account for changes in the cost of living. It is also used to adjust the value of money in financial contracts such as mortgages and bonds. Central banks often use the CPI as a benchmark for inflation targeting and also used for adjusting tax brackets and other economic variables.

CPI is an index of prices of commodities in a fixed quantity over a fixed period It typically includes essential items such as food, education, medicine, housing, fuel, and other necessities. The prices of these goods and services are measured in relation to a reference period, known as the base period, where the value of the index is set at 100. The rate at which the CPI increases or decreases indicates the overall change in the level of prices. The CPI is the most widely used measure of inflation, closely followed by policymakers, financial markets, businesses, and consumers. Using predicted CPI, inflation rate can be calculated for the future. Forecasting the Consumer Price Index (CPI) is an important task as it provides insight into the future movements of inflation. By forecasting the CPI, policymakers, businesses, and individuals can make informed decisions and plan for the future. For example, central banks use the CPI to inform their monetary policy decisions and businesses use it to set prices and make investment decisions. Additionally, forecasting the CPI can also help in identifying potential economic trends and detecting any signs of economic instability. In general, the ability to forecast the CPI is essential for any economic decision-making, as it provides a clear picture of the direction in which the economy is heading and allows for appropriate actions to be taken. the state government of Kerala uses the CPI to set the minimum wages for workers and to adjust the subsidies and other welfare schemes. Businesses in Kerala, particularly those in the retail and wholesale trade sectors, use the CPI to make pricing and investment decisions.

The CPI (IW)(CPI Industrial workers and Agricultural Labourers and CPI (R\U\C) (CPI Rural/Urban /Combined) are released by the Department of Economics and Statistics

on a monthly basis and are widely used as an economic indicator. The Consumer Price Index Numbers for Agricultural and Industrial workers with base 2011-12 in the state of Kerala are being computed monthly for seventeen centers. Apart from this CPI (R\U\C) with base 2018 is also calculated for rural and urban areas of all districts and the state monthly.

In recent years, machine learning techniques have been used to forecast the CPI. These techniques can take into account a wide range of variables and patterns in historical data and can produce more accurate forecasts than traditional methods.

To the best of my knowledge, there is currently no study that has been conducted on the use of machine learning for forecasting the Consumer Price Index (CPI) in Kerala. However, it is possible that machine learning techniques could be applied in this context to improve the accuracy of CPI forecasting in the state.

This study aims to identify an appropriate forecasting model for the Consumer Price Index (CPI) in the Indian state of Kerala. It will use various statistical and machine learning techniques to forecast the CPI and evaluate the performance of different models to determine the most suitable one.

OBJECTIVE

To implement machine learning models to forecast the consumer price index Kerala.

LITERATURE SURVEY

There have been various studies on using machine learning techniques for forecasting consumer price indices in India and other countries and regions. These studies have employed techniques such as time series analysis and various machine learning algorithms, such as linear regression, decision trees, and neural networks, Long Short Term Memory, to analyse historical data on consumer prices in order to make predictions about future price movements.

[1] Dr P K Sarangi, Dr Deepthi Sinha, Dr Sachin Sinha, Dr Meenakshi Sharma in their study “Forecasting of Consumer Price Index Using Neural Networks Model” suggested ANN as abetter choice for predicting CPI taking into account two network architecture. The first one is 12-12-1, and the other one is 8-8-1. The MAE and MSE for 12-12-1 is 0.40446 and 0.28256 respectively. While the MAE and MSE for 8-8-1 is 0.3998 and 0.30943 respectively.

[2] The study "Machine Learning Approach for the Prediction of Consumer Food Price Index" by P. K. Sarangi, D. Sinha, and Mittal (2021) concluded that a simple Artificial Neural Network (ANN) model with backpropagation is highly capable in forecasting Consumer Food Price Index (CFPI).

[3] S Zahara et al 2020 conducted a study on Consumer Price Index Numbers and suggested a forecasting model, Long Short Term Memory with nonlinear parameter input to predict the next move in Indonesia. They used various optimization algorithms to obtain best accuracy in the LSTM method. Stochastic Gradient Descent, Root Mean Square Propagation, Adaptive Gradient, Adaptive moment, Adadelta, Nesterov Adam,Adamax optimization algorithms were used to improve the accuracy ,Nesterov Adam got the best result with RMSE value 4.088.But they conclude that the accuracy in this model was still

far from expectations ,there were several aspect of evaluation could be implemented in the next study like variation of epoch ,hidden layer, batch size input variable could be tested.

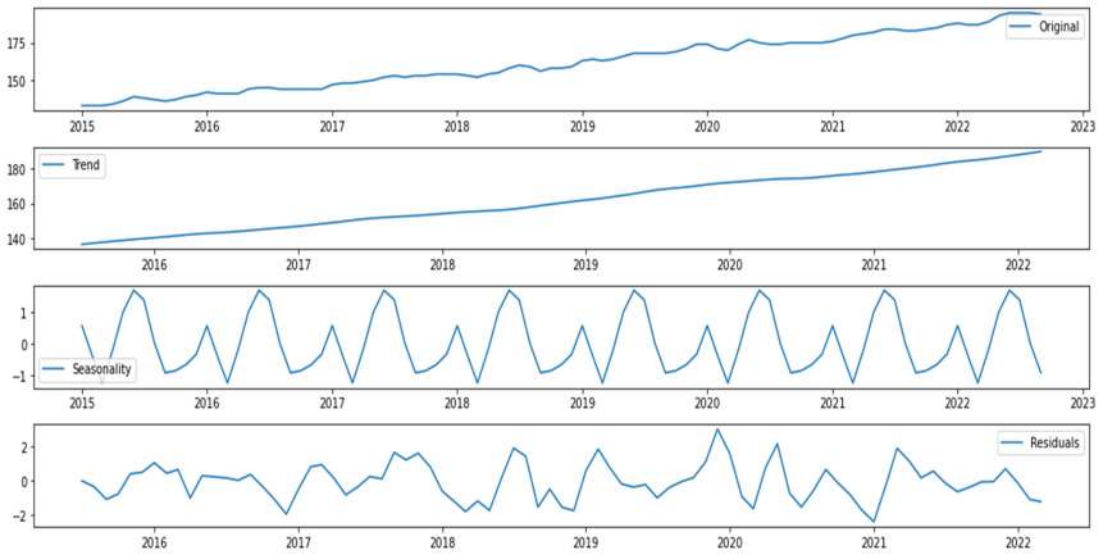
[4] Cheng Yang and Shuhua Guo 2021 in their research article "Inflation Prediction Method Based on Deep Learning" presents a method for forecasting inflation using a GRU-RNN (gated RNN) model which shows superior performance in predicting the Consumer Price Index (CPI) compared to several comparison methods, as evidenced by lower MSE, MAPE, and SMAPE values. GRU-RNN is an improved version of RNN (recurrent neural network) and has been shown to have better performance in sequence prediction performance. Furthermore, the GRU-RNN model outperforms the BP method, which is a shallow-layer neural network, highlighting the advantages of using a deep network structure and the improved capabilities of the GRU-RNN model, which includes gate nodes to enhance its performance. In this method, Consumer Price Index (CPI) is used as the indicator for inflation and multiple economic-related indexes are included as features for the model. The proposed method uses historical data as input and trains a GRU-RNN model to optimize its parameters.

DATA SET AND ANALYSIS

The dataset used in this study consists of monthly Consumer Price Index (CPI) data for the state of Kerala, India. The data spans from April 2015 to September 2022, covering a period of over seven years. By analyzing this dataset, the study aims to identify trends and patterns in CPI values over time and to develop accurate forecasting models using machine learning techniques such as ARIMA and LSTM.

Month	CPI
2015-01	133
2015-02	133
2015-03	133
^^^^^^^^^^^^^^^^^^^^	
2022-05	193
2022-06	195
2022-07	195
2022-08	195
2022-09	194

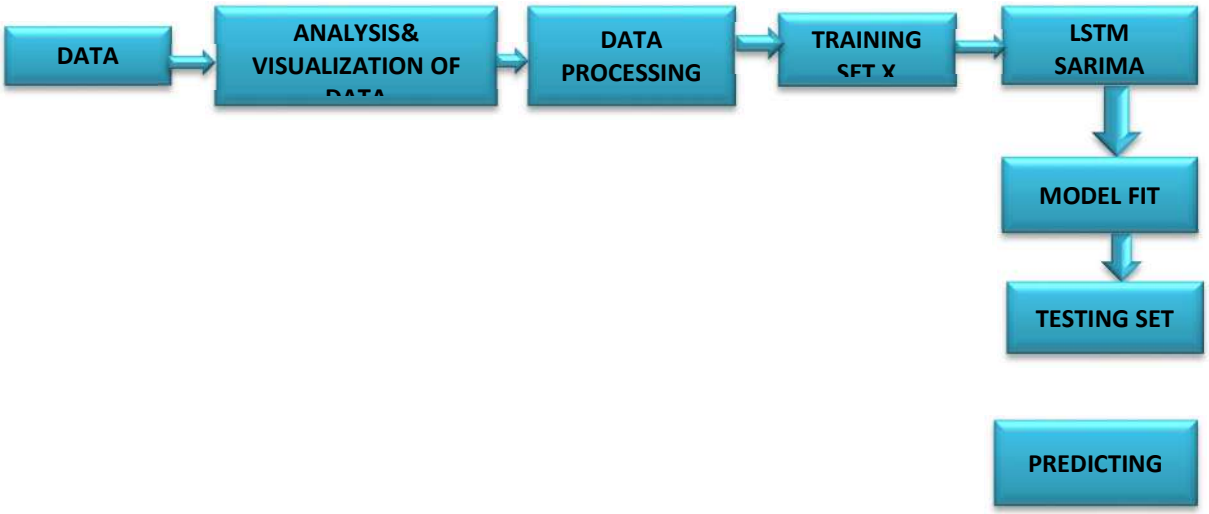
Seasonal decomposition graph



METHODOLOGY AND METHODS USED

Two models are selected (ARIMA and LSTM) and compare these models in terms of their ability to analyze and make predictions based on the given dataset.The following steps are implemented:

- Collect a dataset containing time series data on the Consumer Price Index
- Pre-process the data by cleaning, formatting, and potentially removing any irrelevant or redundant information.
- Divide the data into training and test sets or use cross-validation methods to evaluate the model's performance.
- Train the chosen model using the training data and fine-tune the model's parameters to improve its performance.
- Evaluate the model's performance using the test set.



Libraries and Tools

The analysis was done using Python Programming. The technique used for time series analysis was SARIMA and LSTM.

Libraries are,

- pandas
- matplotlib.pyplot
- numpy
- tensorflow
- tensorflow.keras.models
- tensorflow.keras.layers
- sklearn.preprocessing
- sklearn.metrics
- seaborn
- statsmodels.tools.eval_measures
- matplotlib.pylab
- statsmodels.tsa.seasonal
- pmdarima.arima

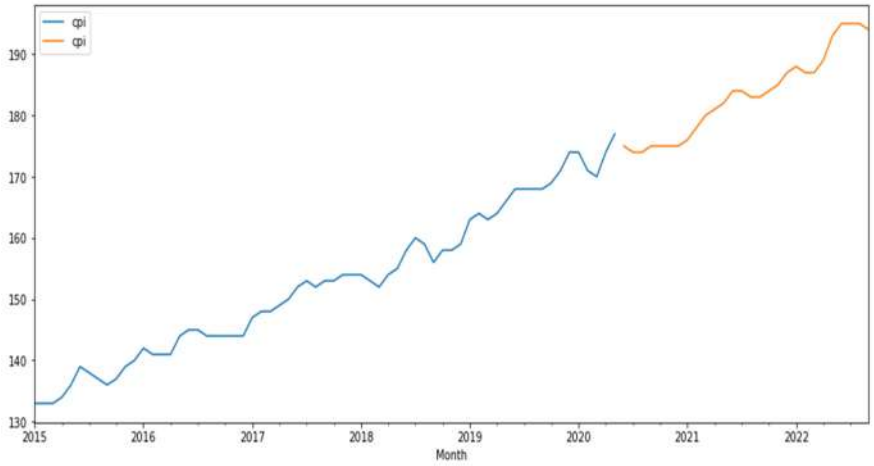
RESULT WITH INFERENCE

ARIMA MODEL

Train -Test Split

In machine learning, it is important to divide the available data into two separate sets: training data and testing data. The purpose of this division is to use the training data to develop a predictive model that can accurately capture the underlying patterns and trends in the data. Once the model is fitted to the training data, it can be evaluated and tested using the testing data to measure its effectiveness in predicting unseen data.

For the specific case of the ARIMA (Seasonal Auto Regressive Integrated Moving Average) model, the first step would be to split the available CPI data into training and testing datasets. The training dataset would be used to fit the SARIMA model and tune its parameters, while the testing dataset would be used to evaluate the performance of the model in predicting future CPI values.



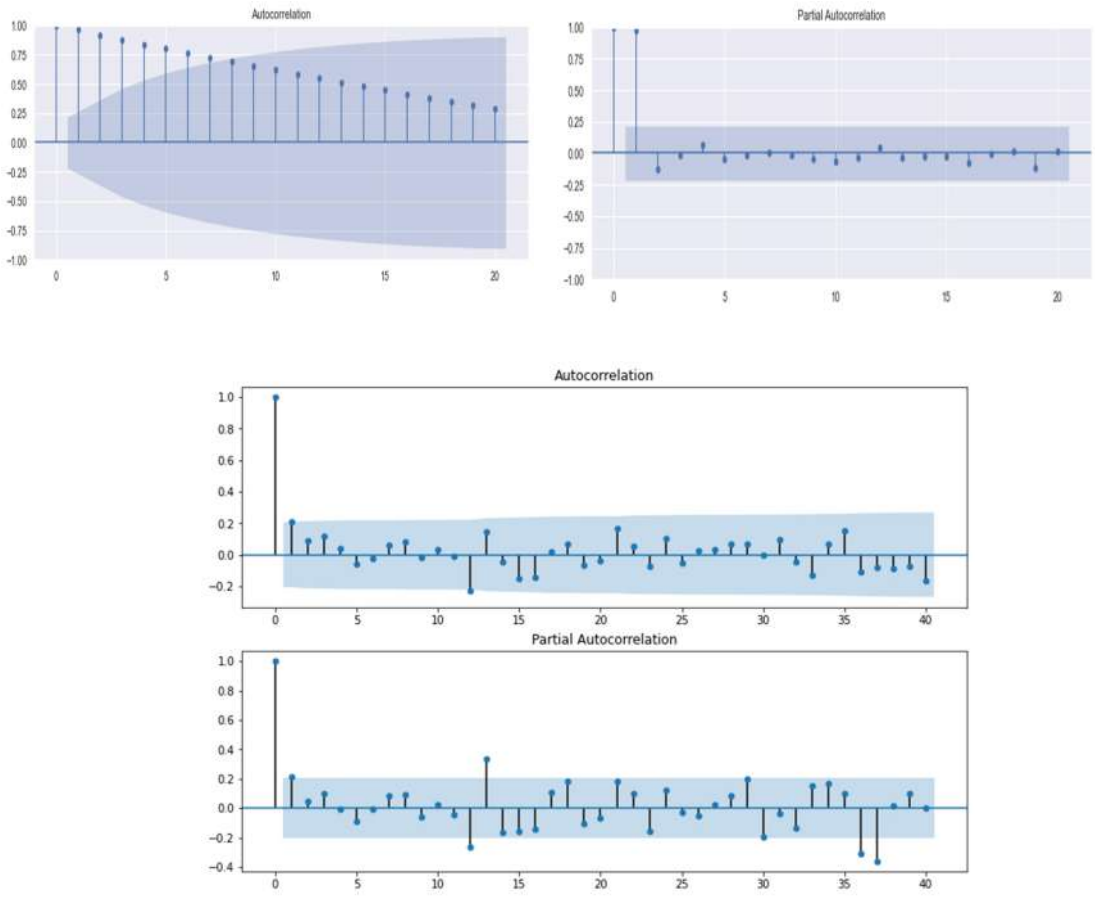
Training data in blue colour and test shown in orange colour

In time series analysis, a series is considered to be stationary if its mean, variance, and autocorrelation remain constant over time. A stationary time series is characterized by a correlogram or ACF (Auto Correlation Function) that quickly dies down, while a non-stationary series exhibits a slow decay. To determine stationarity, the Augmented Dickey-Fuller (ADF) test is commonly used. The test has two hypotheses: H0, the null hypothesis, and H1, the alternative hypothesis. H0 assumes that the series is non-stationary, while H1 suggests that the series is stationary. If the p-value is greater than 0.05, we fail to reject H0, indicating that the series is non-stationary. If the p-value is less than 0.05, we accept H1, implying that the series is stationary.Here ADFuller test gives Test Statistic 0.575829 and p-value .987004 that implies this data set is non stationary.

Before

The method used for converting non-stationary to stationary for time series is differencing.the below graph shows ACF and PACF before and after of differencing. The model identification is done using Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF).

After



Graph of auto correlation and partial correlation before and after differencing

The pmdarima.arima module is a machine learning library used to identify accurate ARIMA models for time series forecasting. The pmdarima.arima module provides a number of functions that simplify the process of identifying the most accurate ARIMA model for a given time series. These functions automate the process of fitting and evaluating multiple ARIMA models, selecting the best model based on statistical metrics such as AIC (Akaike Information Criterion).

The auto_arima function in the pmdarima.arima module is particularly useful for automatically identifying the optimal ARIMA model for a given time series. This function performs a stepwise search algorithm over multiple ARIMA models, testing each model's performance against the available data. The function then returns the ARIMA model with the lowest AIC value, indicating the best model for predicting future values of the time series.

Performing stepwise search to minimize AIC

ARIMA(2,1,2)(1,1,1)[12] : AIC=186.023, Time=0.31 sec
ARIMA(0,1,0)(0,1,0)[12] : AIC=206.697, Time=0.02 sec
ARIMA(1,1,0)(1,1,0)[12] : AIC=187.692, Time=0.05 sec
ARIMA(0,1,1)(0,1,1)[12] : AIC=inf, Time=0.11 sec
ARIMA(2,1,2)(0,1,1)[12] : AIC=184.636, Time=0.31 sec
ARIMA(2,1,2)(0,1,0)[12] : AIC=206.953, Time=0.06 sec
ARIMA(2,1,2)(0,1,2)[12] : AIC=186.049, Time=0.39 sec
ARIMA(3,1,3)(0,1,1)[12] : AIC=188.291, Time=0.35 sec
ARIMA(2,1,2)(0,1,1)[12] : AIC=inf, Time=0.31 sec

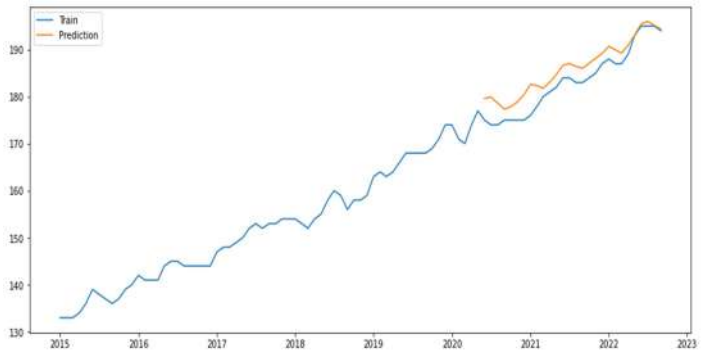
Best model: ARIMA(2,1,2)(0,1,1)[12]

- 2 the number of autoregressive (AR) terms
- 1 differencing operation (to make the time series stationary)
- 2 moving average terms
- 1 seasonal autoregressive term
- 1 seasonal differencing operation
- 1 seasonal seasonal moving average term
- A seasonal cycle of 12 (i.e., monthly data)

This model is used to forecast time series data that exhibit seasonal patterns.

Train model used is auto_arima(train, trace=True, error_action='ignore', suppress_warnings=True)

After fitting the model using training data, it is essential to evaluate the model's performance using test data. The graph shown the predicted values of test(orange) and actual values (blue).



Graph of actual predicted test data

Performance testing

RMSE stands for Root Mean Squared Error, which is a commonly used metric to evaluate the performance of a time series forecasting model. It measures the difference between the predicted and actual values of the time series.

This error value is calculated by taking the square root of the average of the squared differences between the predicted and actual values.

A lower RMSE value indicates that the model is better at predicting the future values of the time series. Here RMSE is 3.27.

Predicted CPI using SARIMA Model

Date	Predicted future values
2022-10-01	195.011124
2022-11-01	195.951605
2022-12-01	197.103139

Graph of future data

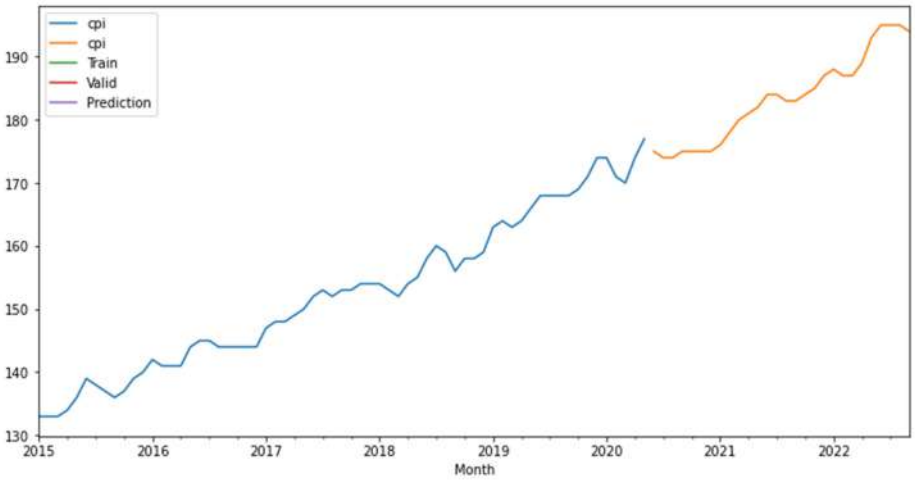
LSTM Model

Data pre-processing

Data pre-processing is a crucial step in data analysis and machine learning. It involves cleaning and transforming raw data into a format that is suitable for analysis and modeling. One common pre-processing step is to fix the date column as the index column.In time series analysis, the time dimension is critical and often serves as the basis for forecasting and modeling. By setting the date column as the index column, we make it easier to slice, filter, and manipulate the data by date.

Train Test Split

Splitting a dataframe into training and test sets, The "train_test_split" function from the sklearn library is being used to split the data into two sets: 80% of the data is being assigned to the training set and 20% is being assigned to the test set. The "shuffle" argument is set to "False.



Train Test Split of data

Prepare Data

Prepare the data for a time series prediction problem with a look-back window size of 3. The data is converted into input (X) and output (y) pairs. X represents the window of previous n_features (3) time steps, while y represents the next time step following X. Reshapes the training and test data into a 3-dimensional format, which is required as input for LSTMs models.

In time series prediction problems, the input data is often transformed into input (X) and output (y) pairs. Define a look-back window size of 3, where X represents the window of the previous 3 time steps, and y represents the next time step following X. This allows us to use the previous values to predict future values.

When using LSTM models for time series prediction, we need to reshape the data into a 3-dimensional format that is suitable for the LSTM input shape. The 3 dimensions are (samples, time steps, features), where samples refer to the number of observations in the dataset, time steps refer to the number of time steps in each input sequence, and features refer to the number of features in each time step.

Train Model

- model = Sequential()
- model.add(LSTM(8, input_shape=(1, look_back=3),))
- model.add(Dense(1))
- model.compile(loss='mean_absolute_error', optimizer='adam’)
- model.fit(trainX, trainY, epochs=100, batch_size=12, verbose=1)

Test Model

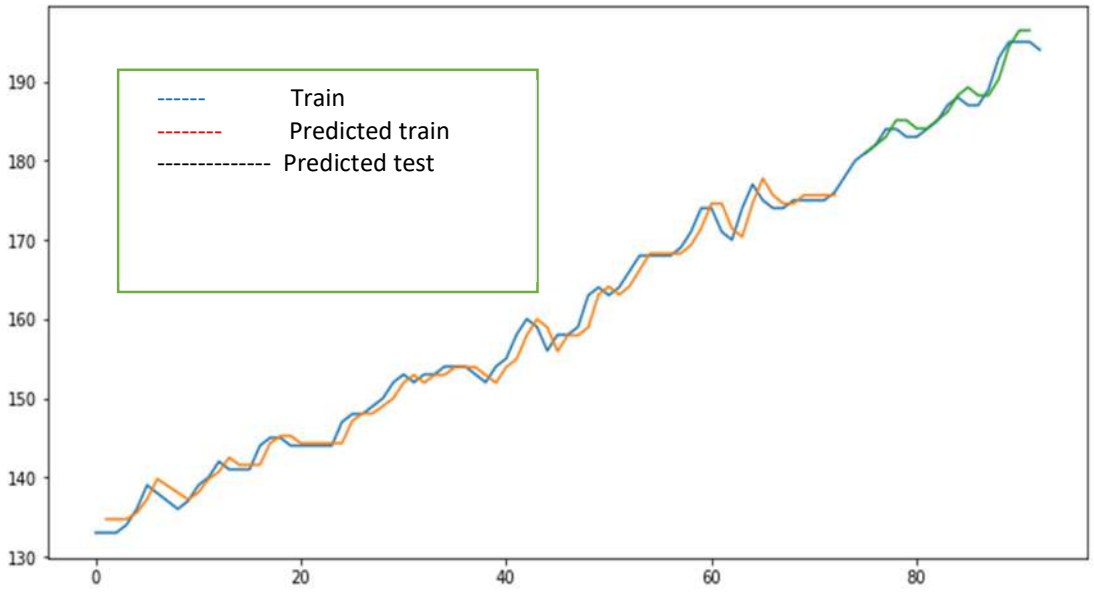
testPredict = model.predict(testX)

LSTM model was trained using the Keras deep learning library in Python. The model was trained with a look-back window size of 3, where X represents the window of previous 3 time steps and y represents the next time step following X. The training data was fed to the model with 100 epochs and a batch size of 12. The loss function used for the model was mean absolute error and the optimizer was Adam. The trained model was then tested on the test data.

Validation of Model

In order to optimize the performance of the LSTM model, we experimented with different values of the epoch and look back parameters. After testing the model with different values of these parameters, we found that the lowest root mean squared error (RMSE) was obtained with a look back window of 3 and 100 epochs.

LSTM model was trained and evaluated using the root mean squared error (RMSE) metric. The trained model was evaluated on both the training and test datasets. The RMSE of the model on the training set was found to be 1.55, indicating that the model performed reasonably well on the training data. Similarly, the RMSE of the model on the test set was found to be 1.27, indicating that the model was able to generalize well on the unseen data. The lower RMSE on the test set compared to the training set suggests that the model was not overfitting the training data. Overall, the results suggest that the LSTM model was able to effectively capture the underlying patterns in the time series data and make accurate predictions.



Graph of actual and predicted data

Both ARIMA and LSTM models are applied to the dataset, which represents the Consumer Price Index (CPI) for Kerala. The ARIMA model involves a series of steps, including data pre-processing, training/test data split, identifying and fitting the model, and evaluating its performance using metrics such as RMSE. The best ARIMA model is selected based on its AIC value and is then used to forecast future CPI values. The LSTM model involves a similar set of steps, including data pre-processing, train/test data split, and model fitting. The LSTM model is a neural network-based model that can capture complex temporal dependencies in time series data. The LSTM model is trained and evaluated based on its ability to accurately predict future CPI values.

The study comparing ARIMA and LSTM models for forecasting consumer price index, it was found that the LSTM model had a lower RMSE value of 1.77 compared to the SARIMA model's RMSE value of 3.27. This indicates that the LSTM model is more accurate in predicting the future values of the time series data.

CONCLUSION

Compared to traditional time series models such as ARIMA, LSTM models can handle non-linear relationships and can automatically learn relevant features from the input data. However, they require significantly more data and computational resources for training and may suffer from overfitting if not properly regularized. Overall, LSTM models are a powerful tool for time series analysis and are widely used in many real-world applications. The study only evaluated the performance of the ARIMA and LSTM models on one dataset .Using a longer time frame could potentially provide more insights into these patterns and trends.

In conclusion, the study demonstrated the effectiveness of both ARIMA and LSTM models in time series forecasting. However, the study is limited by the short duration of the dataset used. To address this limitation, future studies can incorporate longer time series with more historical data. Furthermore, other deep learning models and evaluation metrics can be explored to compare the performance of the models. By using more robust evaluation techniques, the accuracy and reliability of the models can be enhanced.

REFERENCES

[1] Fundamentals of Applied Statistics , S.C.Gupta, V.K.Kapoor, Sultan Chand & Sons Publication.

[2] Dr P K Sarangi, Dr Deepthi Sinha, Dr Sachin Sinha, Dr Meenakshi Sharma "Forecasting of Consumer Price Index Using Neural Networks Model".Innovative Practises in Operations Management & Information Technology (ISBN 978-93-84562-11-3)

[3] P. K. Sarangi, D. Sinha, and Mittal (2021) The study "Machine Learning Approach for the Prediction of Consumer Food Price Index" . 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) DOI: 10.1109/ICRITO51393.2021

[4] S Zahara1 , Sugianto1and M B Ilmiddaviq1 2020 Consumer price index prediction using Long Short Term Memory (LSTM) based cloud computing Journal of Physics: Conference Series 1456 (2020) 012022 doi:10.1088/1742-6596/1456/1/012022

[5] Cheng Yang and Shuhua Guo 2021 "Inflation Prediction Method Based on Deep Learning" Hindawi Computational Intelligence and Neuroscience Volume 20

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