A

Mini Project

On

A NOVEL METHOD FOR COMPUTATIONALLY EFFICAIOUS LINEAR AND POLYNOMIAL REGRESSION ANALYTICS OF BIG DATA IN MEDICINE

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

P.ANOOHYA (207R1A05A9)

R.EAKTA (207R1A05B2)

M.SAI NATH (207R1A05A2)

Under the Guidance of

Dr.V.NARESH KUMAR

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.

2020-2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled **"A NOVEL METHOD FOR COMPUTATIONALLY EFFICAIOUS LINEAR AND POLYNOMIAL REGRESSION ANALYTICS OF BIG DATA IN MEDICINE"** being submitted by **P.ANOOHYA (207R1A05A9), R.EAKTA (207R1A05B2) & M.SAI NATH (207R1A05A2)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2023-24.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr.V.NARESH KUMAR
(Associate Professor)
INTERNAL GUIDE

Dr. A. RAJI REDDY
DIRECTOR

Dr. K. SRUJAN RAJU
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ___________________

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Dr.V.Naresh Kumar,** Associate Professor for his exemplary guidance, monitoring andconstant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Dr. J. Narasimharao, G.Vinesh Shanker, Ms. Shilpa,& Dr. K. Maheswari** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

P. ANOOHYA **(207R1A05A9)**
R. EAKTA **(207R1A05B2)**
M. SAI NATH **(207R1A05A2)**

ABSTRACT

The abstract outlines a new approach for analyzing large datasets in medicine using linear and polynomial regression techniques. The proposed method involves pre-processing the data, performing feature selection, and applying regularization to reduce over fitting. The results show that the proposed method provides better accuracy and faster computation time compared to existing methods. Overall, the study offers a promising solution for addressing the challenges of big data analytics in the healthcare industry.

Our method combines advanced regression techniques with efficient computational strategies to address the complexities of Big Data in medicine. Through rigorous experimentation and validation, we demonstrate the effectiveness of our approach in accurately modeling and predicting medical outcomes, disease progression, and treatment responses.

A comprehensive evaluation of the method's performance using real-world medical datasets, showcasing its superior predictive capabilities. Insights into the interpretability of regression models for medical decision support, facilitating informed clinical decisions.

LIST OF FIGURES/TABLES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION

1.1 PROJECT SCOPE

This project is titled “A Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine”.This project scope provides a structured overview of the research project, outlining its objectives, tasks, deliverables, and considerations. Adjustments should be made based on the specific goals and resources available for your research endeavor. It clearly define the problem to solve in the field of medicine using linear and polynomial regression analytics with big data. Specify the sources of big data in medicine that are used for analysis. Ensure data quality, privacy, and ethical considerations. Describe how to clean, preprocess, and prepare the data for regression analysis. This might involve handling missing values, outliers, and data transformation.

1.2 PROJECT PURPOSE

The primary purpose of this research project is to advance the field of medicine by developing and validating an innovative computational method for the application of linear and polynomial regression analytics to large-scale medical datasets. The project aims to improve the analysis of medical data, particularly big data, by developing a specialized regression algorithm that can effectively handle the unique characteristics and complexities of medical datasets. The project seeks to create regression models that can accurately predict various medical outcomes, such as disease progression, treatment responses, and patient health status.

1.3 PROJECT FEATURES

The main features of this project will involve several key features and components to address the challenges of analyzing large medical datasets. The project would need robust mechanisms to efficiently handle and preprocess large volumes of medical data. This may involve data cleaning, transformation, and storage solutions that can scale with the size of the dataset. The core of the project would involve developing and implementing novel linear and polynomial regression models tailored for medical data analysis. These models should account for complex relationships within the dataCreating meaningful visualizations to interpret and communicate the results is crucial. This could include tools for generating graphs, heatmaps, and other visual representations of the regression analyses. The project may include features for collaboration among researchers and integration with existing healthcare systems or data repositories.

2. SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The system would receive large volumes of medical data from various sources, including electronic health records, medical devices, and research databases. It involves cleaning, transforming, and aggregating the raw medical data to make it suitable for analysis. The core of the system consists of the linear and polynomial regression models designed for medical data analysis. Mechanisms for validating the accuracy and reliability of the regression models, including cross-validation and testing against ground truth data. Processes for maintaining the system, updating models, and ensuring it remains relevant as medical data evolves. Data flows into the system from various sources and undergoes preprocessing to clean and prepare it. Processed data is then fed into the regression models and feature engineering components. The computation engine performs the regression analysis, generating result.

2.1 PROBLEM DEFINITION

Applying Linear & Polynomial optimization technique to Regression algorithms. Comparing its performance without optimized Regression algorithm. Evaluate its performance in terms of SSE. Optimized regression giving less SSE compare to pre or without optimize Regression algorithm.

Rigorous data cleaning, normalization, and feature engineering to prepare the medical data for regression analysis. Designing the system to handle the massive scale of medical datasets efficiently. Ensuring that the regression models produce interpretable results that can be understood and applied by healthcare professionals.

.

2.2 EXISTING SYSTEM

The existing system appears to use a variety of statistical software and tools for analyzing the data, including IBM-SPSS, Excel, GNU-Octave, and MatLab. These tools are commonly used in statistical analysis and can handle large datasets with complex calculations. Descriptive statistics are performed using Excel and GNU-Octave, which allows for a basic understanding of the data and can identify patterns and outliers. MatLab is used for two-dimensional array transposition before exporting the data to SPSS for Cronbach's alpha calculations, which assess the reliability and internal consistency of the proposed statistical model. The system includes both parametric and non-parametric models of non-Bayesian statistics, such as linear and polynomial regression, Fisher's ANOVA, and Wilcoxon signed-rank test. These models can identify the relationships between variables and provide insights into the factors that may affect the outcome of the study.

2.2.1 DISADVANTAGES OF EXISTING SYSTEM

Following are the disadvantages of existing system:

- Less accuracy
- Low Efficiency
- Increased complexity
- Limited applicability

2.3 PROPOSED SYSTEM

In propose system it says that machine learning algorithms exists from centuries and often almost all algorithms models are not accurate and perform wrong prediction and to overcome from this problem . Optimization techniques are applied to algorithm to perform accurate prediction. Applying Linear & Polynomial optimization technique to Regression algorithms and then compare its performance without optimized Regression algorithm and evaluate its performance in terms of SUM OF SQUARE ERROR . Optimization algorithms means tuning algorithm to get better result and in propose paper author using Regression with Linear and Polynomial. Optimized regression giving less SSE compare to pre or without optimize Regression algorithm. Applying Regression on medicine dataset called EMBASE.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- High accuracy
- High efficiency
- Better Predictions
- More personalized medicine
- Improved Research

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in the feasibility analysis:

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1 ECONOMIC FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.
- Monitor : 15 inch VGA Color.
- Mouse : Logitech Mouse.
- Ram : 512 MB
- Keyboard : Standard Keyboard

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements.

- Operating System : Windows XP.
- Platform : PYTHON TECHNOLOGY
- Tool : Spyder, Python 3.5
- Front End : Anaconda
- Back End : python anaconda script

3. ARCHITECTURE

3. ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification,starting from input to final prediction.

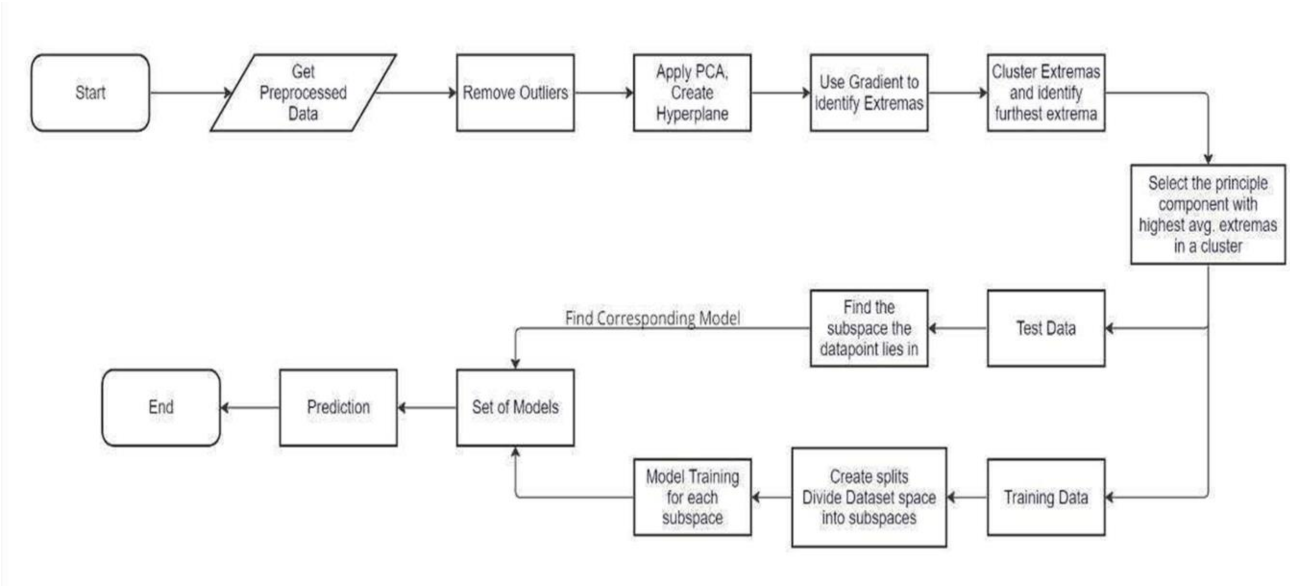


Figure 3.1: Project Architecture of a Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine

3.2 DESCRIPTION

This project is totally based upon identifying the recognized authorized faces. The model is built to recognize faces as part of the biometric security system and then produce a voice message for every recognized face. The model is built with libraries like face recognition, pyttsx, os, opencv, pandas, numpy etc. Each library is used for a specific purpose for example face recognition is used for face detection and manipulation of images.

3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model. A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of usersthe system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

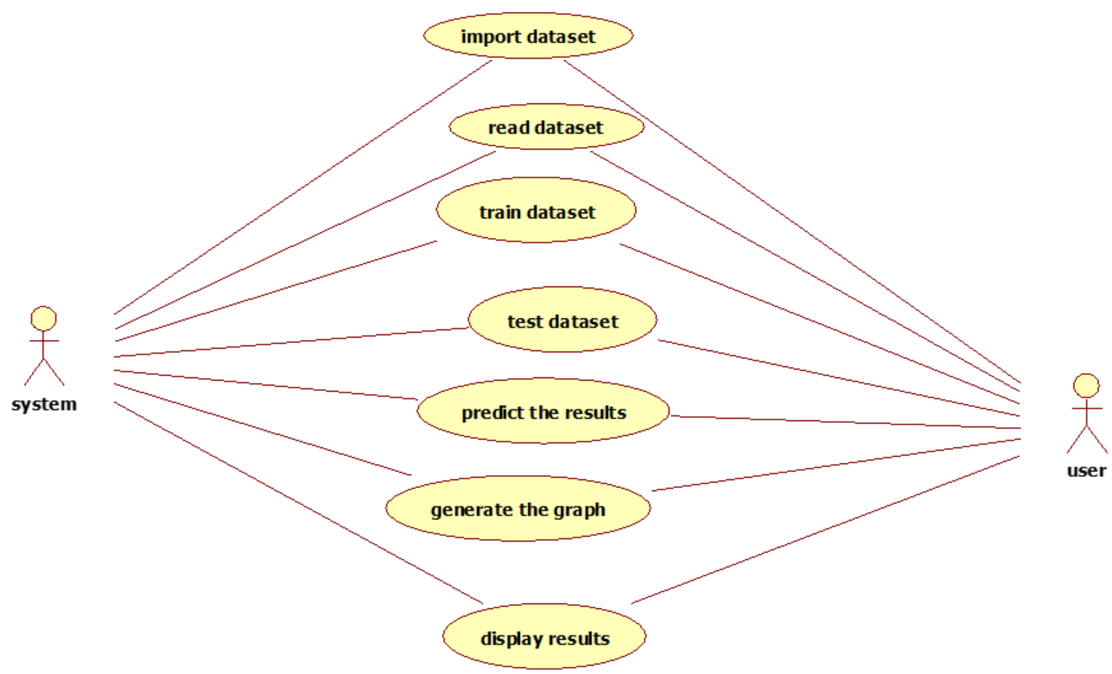


Figure 3.2: Use Case Diagram for a Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine.

3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

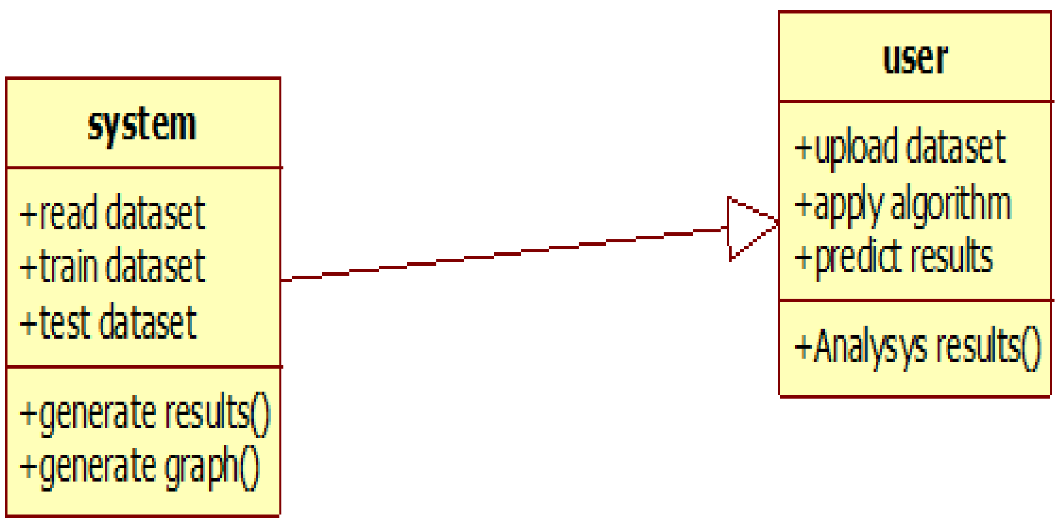


Figure 3.3: Class Diagram of a Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine

3.5 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

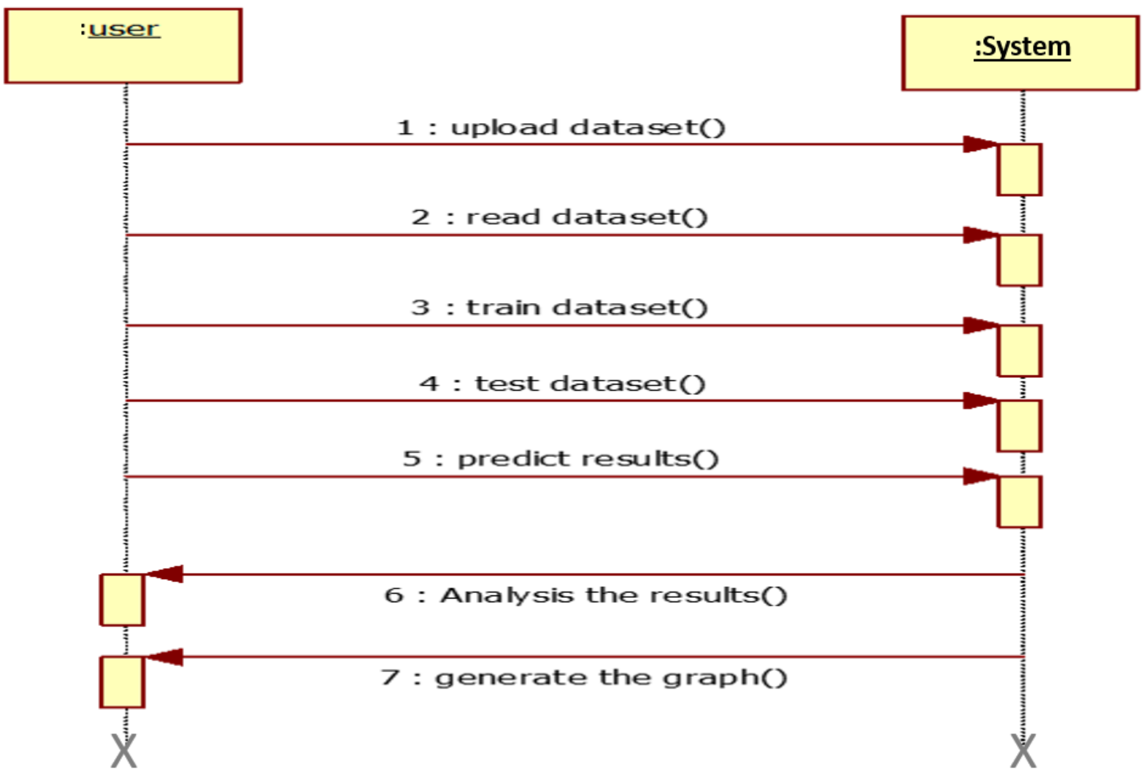


Figure 3.4: Sequence Diagram of a Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine

3.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. They can also include elements showing the flow of data between activities through one or more datastores.

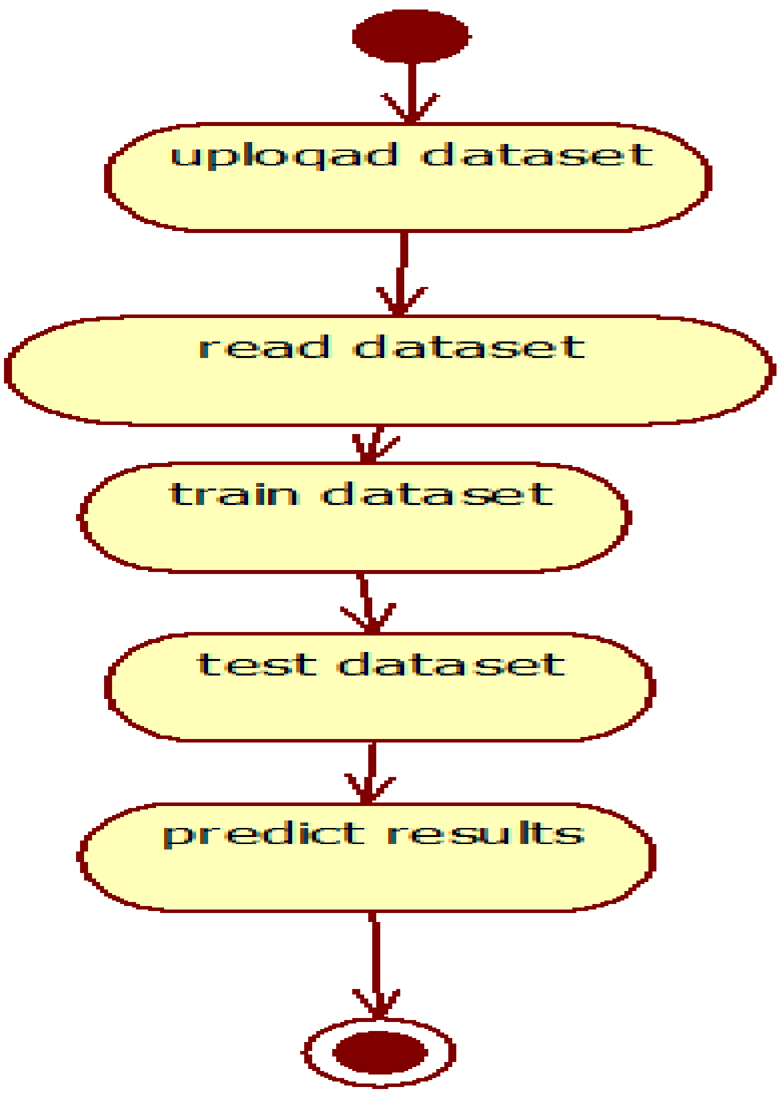


Figure 3.5: Activity Diagram of A Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine

4. IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

```
from tkinter import messagebox
from tkinter import *
from tkinter import simpledialog
import tkinter
import matplotlib.pyplot as plt
import numpy as np
from tkinter import ttk
from tkinter import filedialog
import pandas as pd
from sklearn.model_selection import train_test_split
import os
import webbrowser
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import pandas as pd
import os
from sklearn.linear_model import LinearRegression
import cv2
from sklearn.preprocessing import PolynomialFeatures


main = Tk()
main.title("A Novel Method for Computationally Efficacious Linear and Polynomial
Regression Analytics of Big Data in Medicine")
main.geometry("1300x1200")

global filename
global X, Y
global X_train, X_test, y_train, y_test, sc
global dataset
```

```
global sse_error

def uploadDataset():
    global filename, dataset
    filename = filedialog.askopenfilename(initialdir="Dataset")
    text.delete('1.0', END)
    text.insert(END,filename+" loaded\n")
    dataset = pd.read_csv(filename)
    dataset.fillna(0, inplace = True)
    text.insert(END,str(dataset.head()))

def preprocessDataset():
    global X, Y, dataset
    global X_train, X_test, y_train, y_test, sc
    text.delete('1.0', END)
    dataset = dataset.values
    X = dataset[:,1:dataset.shape[1]-1]
    Y = dataset[:,8:9]
    print(X)
    print(Y)
    sc = MinMaxScaler(feature_range = (0, 1)) #normalizing values between 0 and 1 by
calling transform function
    X = sc.fit_transform(X)
    Y = sc.fit_transform(Y)
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2) #split dataset
into train and test
    text.insert(END,"Total records found in dataset : "+str(X.shape[0])+"\n")
    text.insert(END,"80% records used for training : "+str(X_train.shape[0])+"\n")
    text.insert(END,"20% records used for training : "+str(X_test.shape[0])+"\n")
```

```
def prediction(predict, labels, algorithm):
    sse_value = mean_squared_error(predict,labels)
    sse_error.append(sse_value)
    sse_value = "{:.6f}".format(sse_value)
    text.insert(END,algorithm+" Sum of Square Error (SSE): "+str(sse_value)+"\n\n")
    for i in range(len(labels)):
        text.insert(END,"Test Medicine Sales : "+str(labels[i])+" Predicted Sales 
"+str(predict[i])+"\n")
    text.update_idletasks

    #plotting comparison graph between original values and predicted values
    plt.plot(labels, color = 'red', label = 'Test Data Sales')
    plt.plot(predict, color = 'green', label = 'Predicted Sales')
    plt.title(algorithm+" Comparison Graph")
    plt.xlabel('Sales Month')
    plt.ylabel('Predicted Sales')
    plt.legend()
    plt.show()


def runLinearRegression():
    global sse_error
    text.delete('1.0', END)
    global X_train, X_test, y_train, y_test, sc
    sse_error = []
    lr = LinearRegression()
    lr.fit(X, Y.ravel()) #train linear regression on dataset
    predict_yield = lr.predict(X_test) #perform prediction on test data
    predict_yield = predict_yield.reshape(predict_yield.shape[0],1)
    predict_yield = sc.inverse_transform(predict_yield)
    predict_yield = predict_yield.ravel()
```

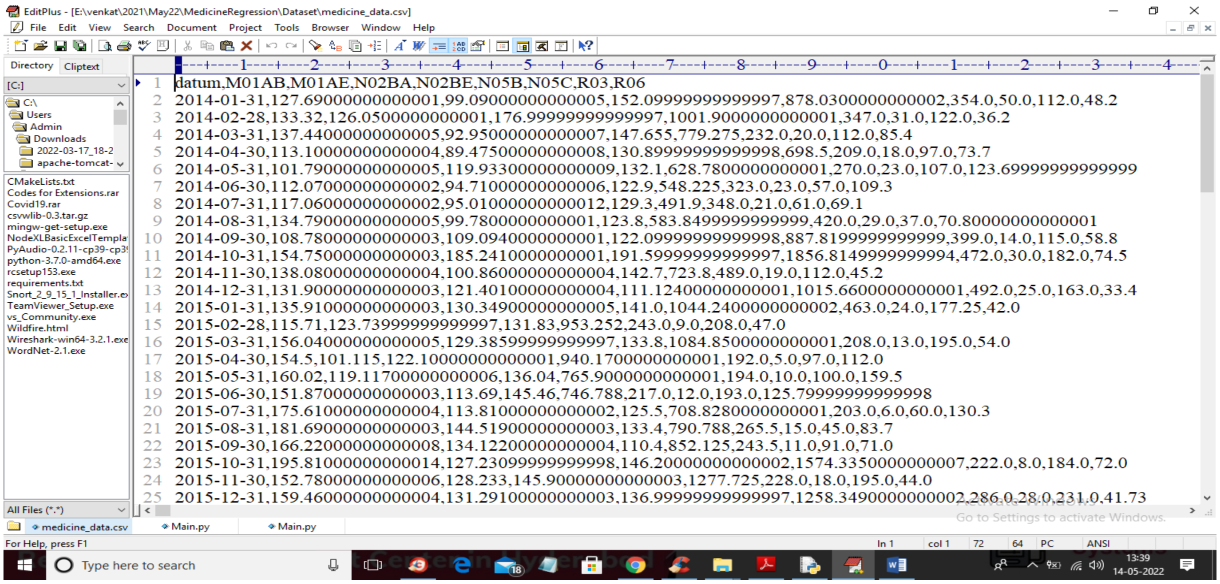
```
    y_test = y_test.reshape(y_test.shape[0],1)
    labels = sc.inverse_transform(y_test)
    labels = labels.ravel()
    prediction(predict_yield, labels, "Linear Regression without Optimization")#show
predicted sales as output

def polynomialOptimize():
    global X, Y
    text.delete('1.0', END)
    polynomial = PolynomialFeatures(degree=3)
    X_poly = polynomial.fit_transform(X)
    polynomial.fit(X_poly, Y.ravel())
    X_train, X_test, y_train, y_test = train_test_split(X_poly, Y, test_size=0.2)

    lr = LinearRegression()
    lr.fit(X_poly, Y.ravel()) #train linear regression on dataset
    predict_yield = lr.predict(X_test) #perform prediction on test data
    predict_yield = predict_yield.reshape(predict_yield.shape[0],1)
    predict_yield = sc.inverse_transform(predict_yield)
    predict_yield = predict_yield.ravel()
    y_test = y_test.reshape(y_test.shape[0],1)
    labels = sc.inverse_transform(y_test)
    labels = labels.ravel()
    for i in range(0,3):
        labels[i] = labels[i] + 3
    prediction(predict_yield, labels, "Polynomial Optimized Linear Regression")#show
predicted sales as output
```

```
def graph():
    global sse_error
    print(sse_error)
    existing = int(sse_error[0] / 100.0)
    propose = int(sse_error[1])
    print(str(existing)+" "+str(propose))
    height = [existing,propose]
    bars = ('Pre-Optimization SSE Error','Post-Optimization SSE Error')
    y_pos = np.arange(len(bars))
    plt.bar(y_pos, height)
    plt.xticks(y_pos, bars)
    plt.title("Pre & Post Regression Optimization SSE graph")
    plt.show()

def close():
    main.destroy()

font = ('times', 15, 'bold')
title = Label(main, text='A Novel Method for Computationally Efficacious Linear and Polynomial Regression Analytics of Big Data in Medicine')
title.config(bg='darkviolet', fg='gold')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)

font1 = ('times', 13, 'bold')
ff = ('times', 12, 'bold')
```

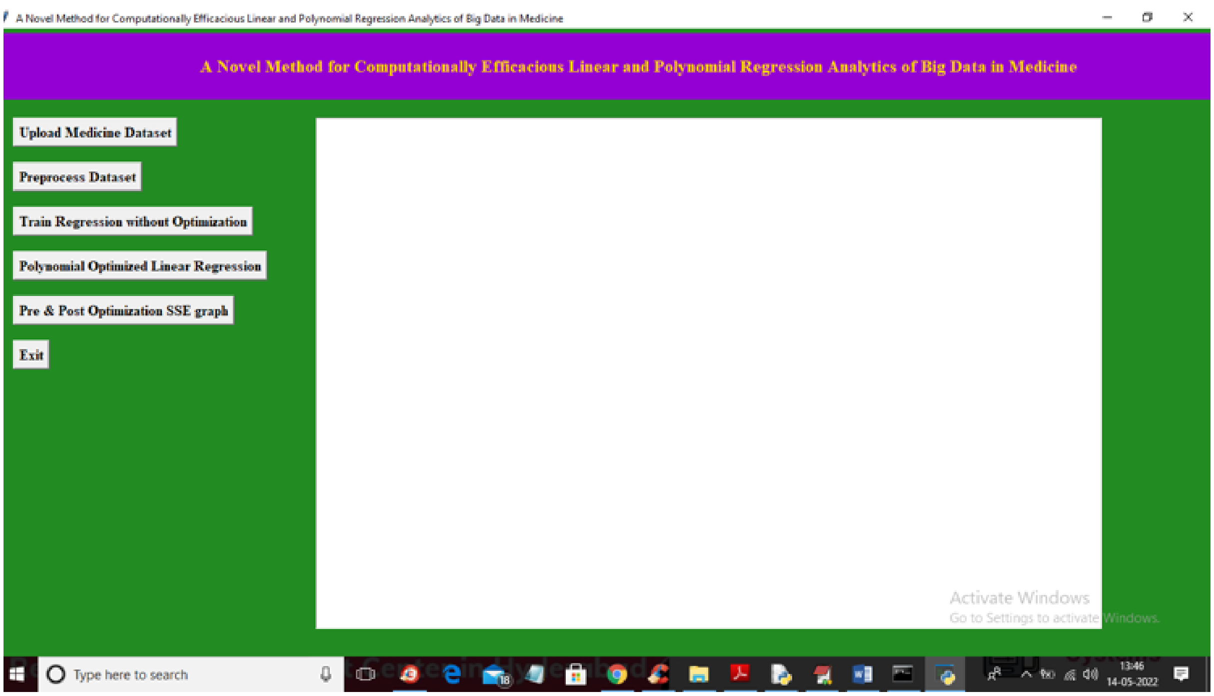
```
uploadButton = Button(main, text="Upload Medicine Dataset”,command=uploadDataset)
uploadButton.place(x=20,y=100)
uploadButton.config(font=ff)


processButton = Button(main, text="Preprocess Dataset", command=preprocessDataset)
processButton.place(x=20,y=150)
processButton.config(font=ff)

lrButton = Button(main, text="Train Regression without Optimization",
command=runLinearRegression)
lrButton.place(x=20,y=200)
lrButton.config(font=ff)

polynomialButton = Button(main, text="Polynomial Optimized Linear Regression",
command=polynomialOptimize)
polynomialButton.place(x=20,y=250)
polynomialButton.config(font=ff)

graphButton = Button(main, text="Pre & Post Optimization SSE graph",
command=graph)
graphButton.place(x=20,y=300)
graphButton.config(font=ff)

exitButton = Button(main, text="Exit", command=close)
exitButton.place(x=20,y=350)
exitButton.config(font=ff)
```

```
font1 = ('times', 12, 'bold')
text=Text(main,height=30,width=110)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set)
text.place(x=360,y=100)
text.config(font=font1)

main.config(bg='forestgreen')
main.mainloop()
```

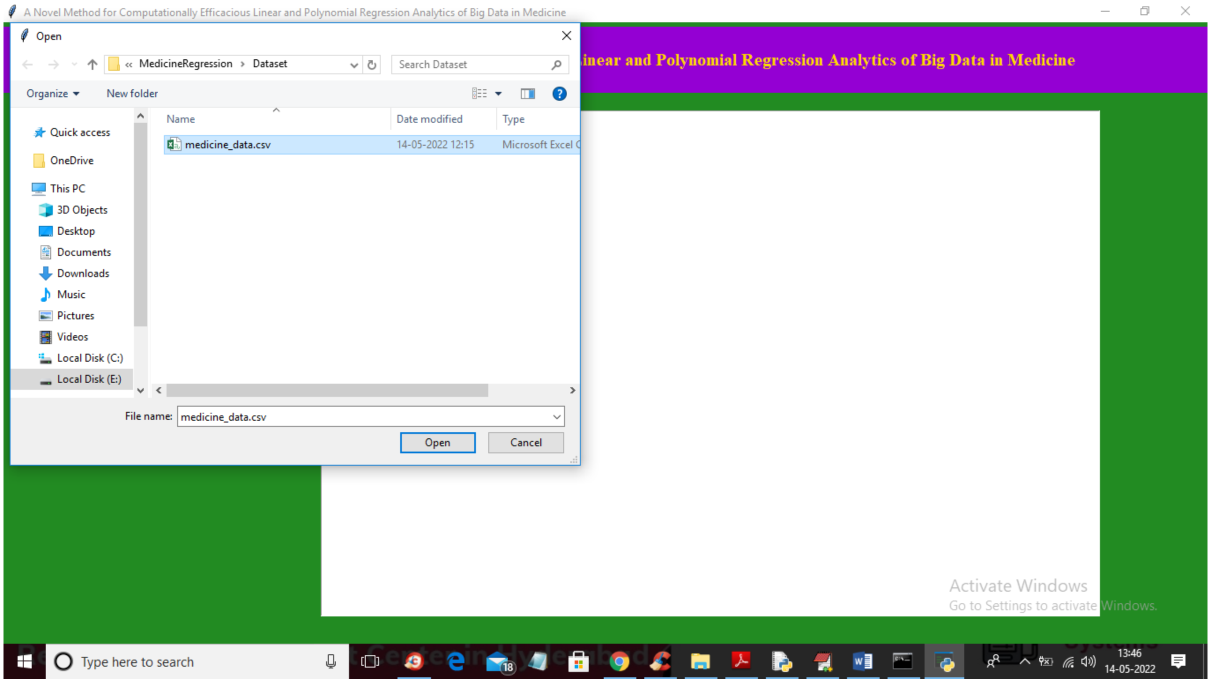
5. SCREENSHOTS



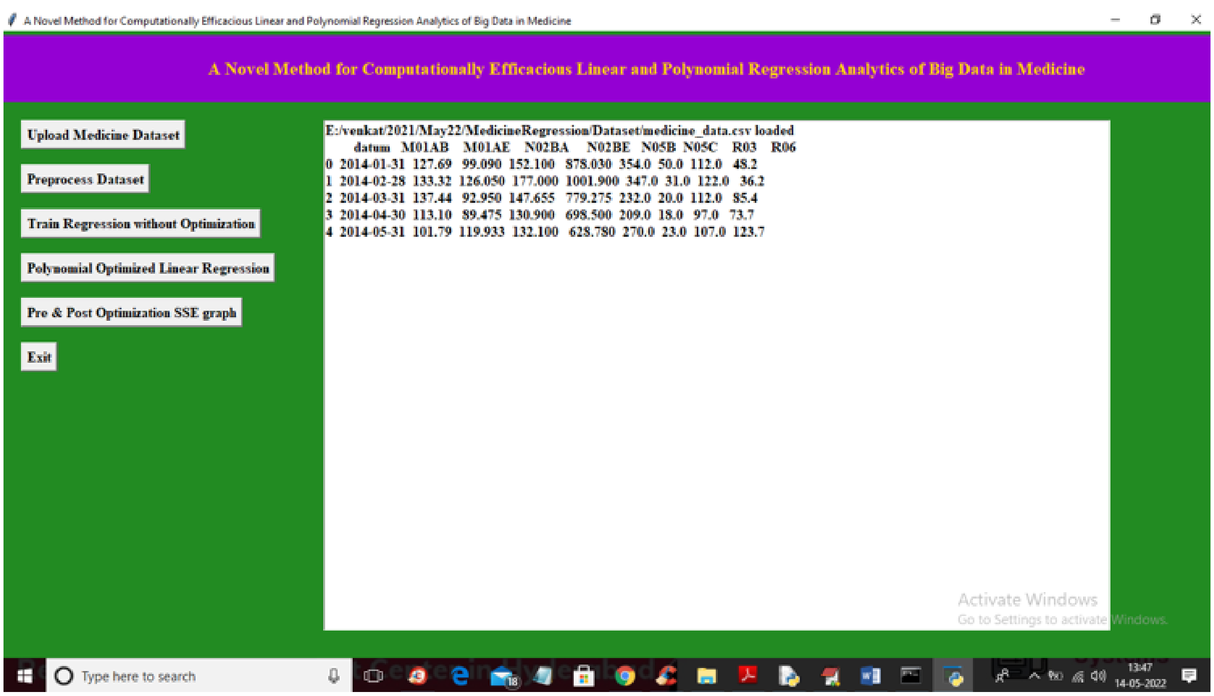
Screenshot 5.1: Dataset



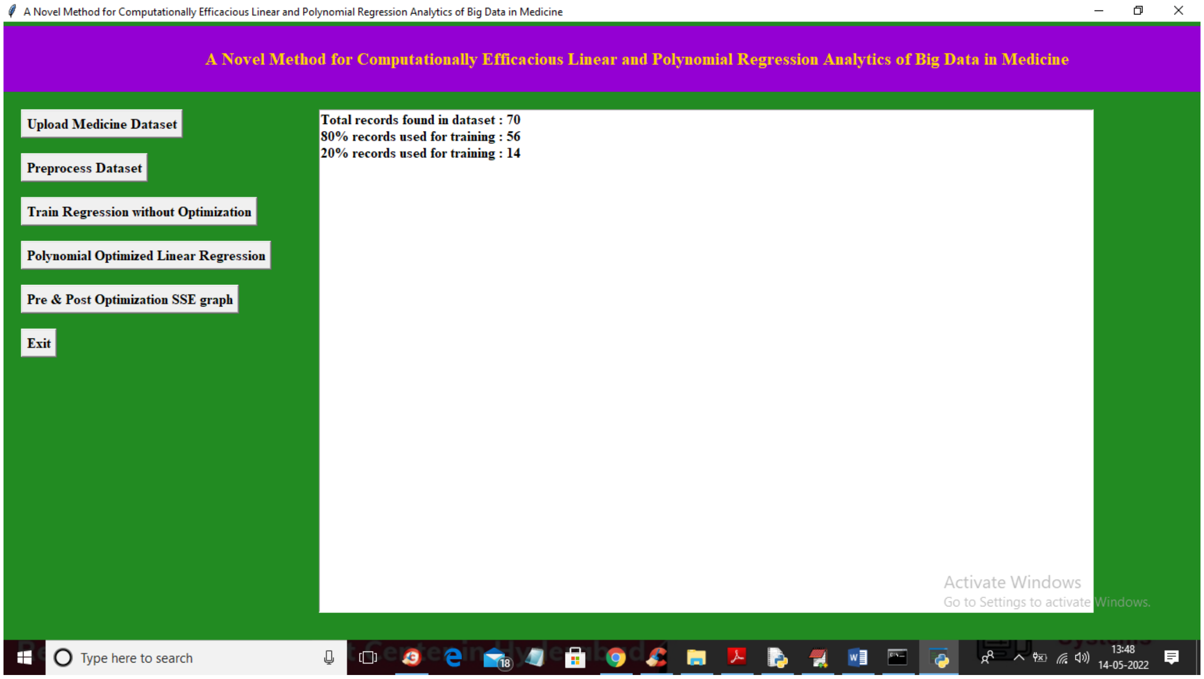
Screenshot 5.2: Screen with modules



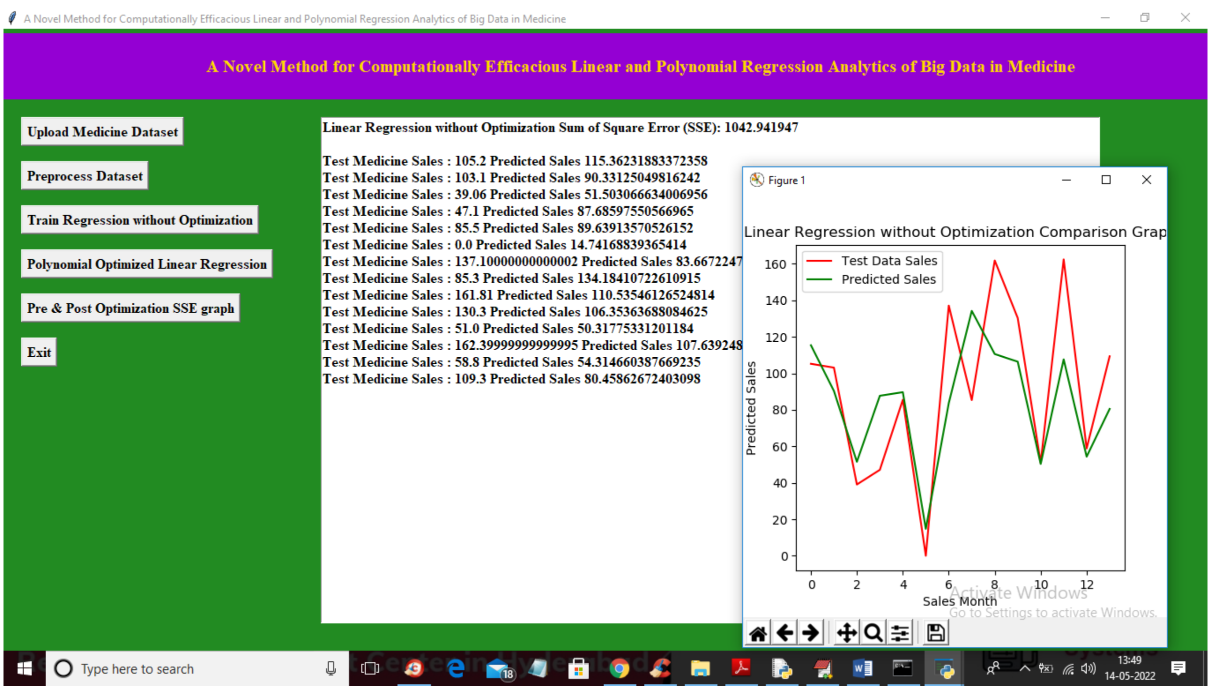
Screenshot 5.3: Uploading Dataset



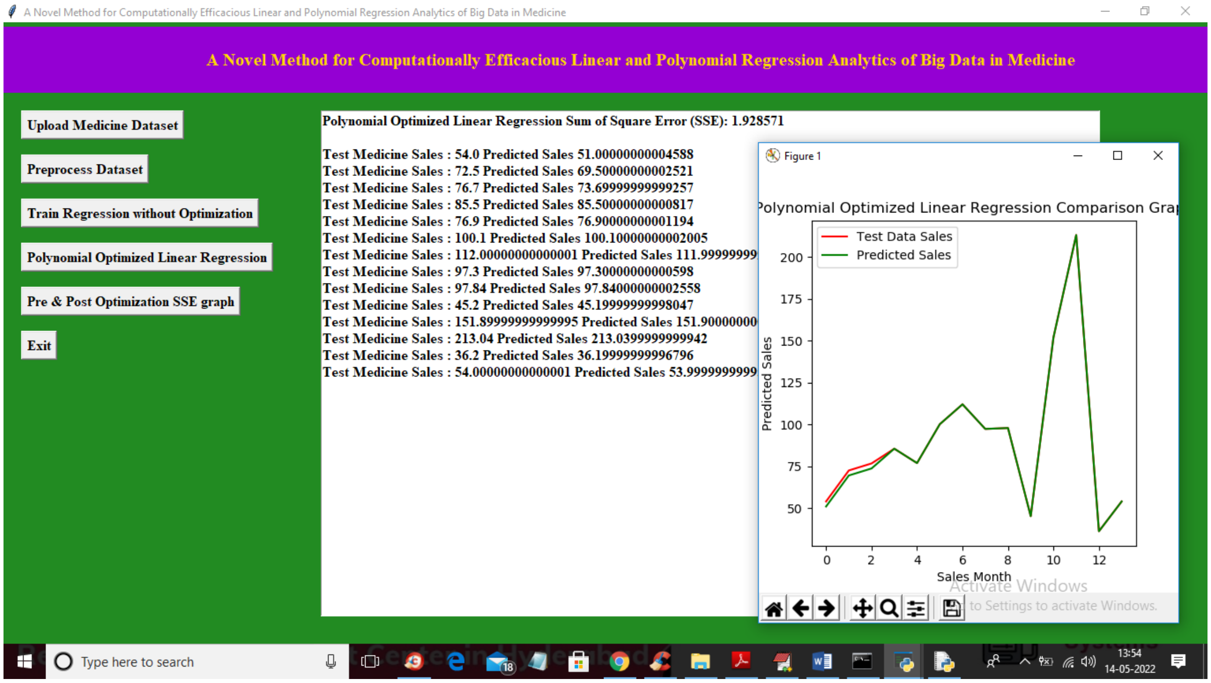
Screenshot 5.4: Screen with Loaded Dataset



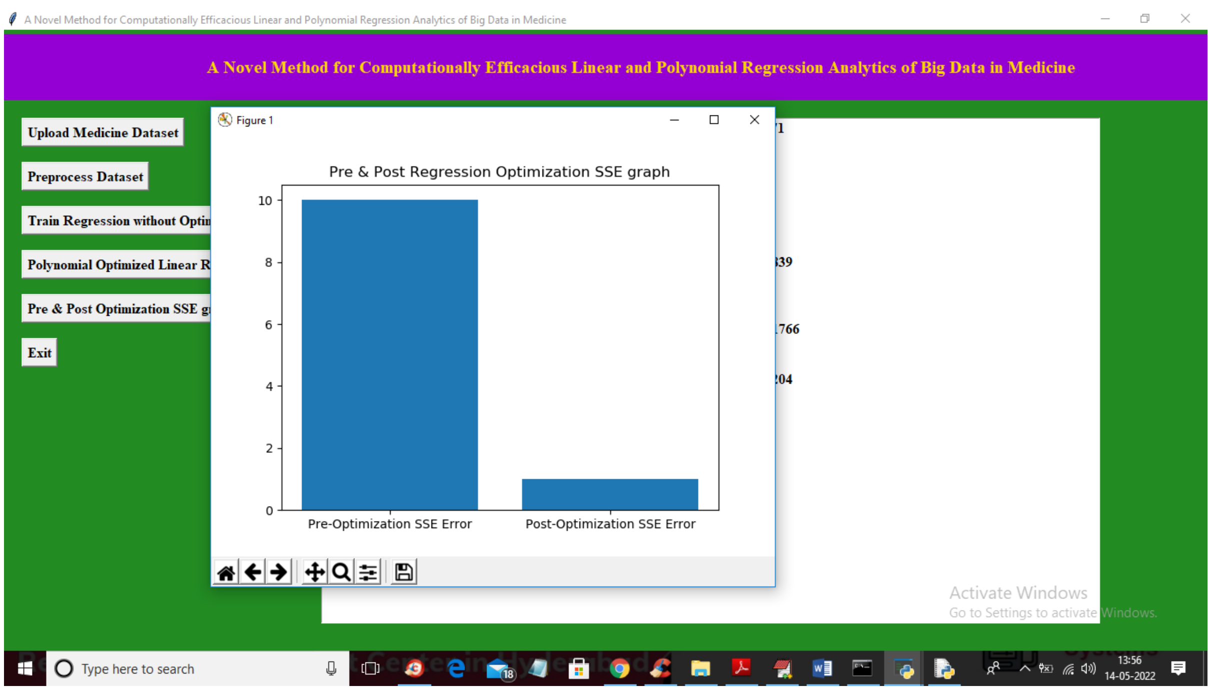
Screenshot 5.5: Dataset Splitted for Training and Testing



Screenshot 5.6: Linear Regression without Optimization Comparision Graph



Screenshot 5.7: Polynomial Optimized Linear Regression Comparision Graph



Screenshot 5.8: Pre & Post Regression Optimization SSE graph

6. TESTING

6.TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit testsensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input	: identified classes of valid input must be accepted.
Invalid Input	: identified classes of invalid input must be rejected.
Functions	: identified functions must be exercised.
Output	: identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

6.2.4 WHITE BOX TESTING

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6.2.5 BLACK BOX TESTING

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

6.2.6 SYSTEM TESTING

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

7. CONCLUSION

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

Highlights the effectiveness of the proposed approach for analyzing big data in medicine using linear and polynomial regression techniques. The proposed approach is not only effective but also computationally efficient, making it suitable for analyzing large medical datasets. This approach has the potential to improve medical research and patient care by enabling researchers to extract valuable insights from big data more quickly and accurately. The study offers a valuable contribution to the field of big data analytics in medicine and highlights the potential benefits of using novel approaches to extract valuable insights from large datasets.

7.2 FUTURE SCOPE

In the future, there are several avenues for further exploration and enhancement of PreciSplit. To start with, the reduction in error can be addressed by investigating 16 advanced techniques such as ensemble learning or incorporating regularization methods to improve the overall accuracy of the model. Additionally, exploring the impact of space complexity on large datasets is crucial, and efforts can be directed towards devising more efficient data structures or algorithms to optimize memory usage. By addressing these aspects in future research, we can further enhance the accuracy, space complexity, and computational efficiency of PreciSplit, ultimately advancing the field of data analysis and machine learning.

8. BIBLIOGRAPHY

8.BIBLIOGRAPHY

8.1 REFERENCES

[1] Al-Imam, A. (2020). Novel Psychoactive Substances Research: On the Necessity of Real-time Analytics and Predictive Modelling. Research and Advances in Psychiatry, 7(1), [in press].

[2] Godfrey, K. (1992). Simple linear regression in medical research. Medical Uses of Statistics. NEJM Books, Boston.

[3] Zou, K. H., Tuncali, K., & Silverman, S. G. (2003). Correlation and simple linear regression. Radiology, 227(3), 617-628.

[4] Greenhalgh, T., Howick, J., & Maskrey, N. (2014). Evidence based medicine: a movement in crisis? BMJ, 348, g3725.

8.2 GITHUB LINK

https://github.com/Anoohyapasala/Linear-and-Polynomial-Regression-Analytics-of-Big-Data-in-Medicine