Detecting Basic Anomalies in Bank Transactions

A PROJECT REPORT

Submitted by

SUDHAKARAN M (2303811724321111)

in partial fulfillment of requirements for the award of the course

AGI1252 - FUNDAMENTALS OF DATA SCIENCE USING R

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

JUNE- 2025

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"DETECTING BASIC ANOMALIES IN BANK TRANSACTIONS"** is the bonafide work of **SUDHAKARAN M (2303811724321111)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

SIGNATURE

Dr.T. AVUDAIAPPAN, M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

PROFESSOR

Department of Artificial Intelligence

K.Ramakrishnan College of Technology (Autonomous)

Samayapuram–621112.

SIGNATURE

Ms.S.Murugavalli., M.E.,(Ph.D).,

SUPERVISOR

ASSISTANT PROFESSOR

Department of Artificial Intelligence

K.Ramakrishnan College of Technology (Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on 02.06.2025

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

I declare that the project report on **"DETECTING BASIC ANOMALIES IN BANK TRANSACTIONS"** is the result of original work done by me and best of my knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF TECHNOLOGY**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **AGI1252 - FUNDAMENTALS OF DATA SCIENCE USING R.**

Sudhakaran.

Signature

SUDHAKARAN M

Place: Samayapuram

Date:02.06.2025

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

I would like to thank **Dr. N. VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. T. AVUDAIAPPAN, M.E.,Ph.D.,** Head of the department, **ARTIFICIAL INTELLIGENCE** for providing his encourage pursuing this project.

I express our deep expression and sincere gratitude to our project supervisor **Ms.S.Murugavalli., M.E.,(Ph.D).,** Department of **ARTIFICIAL INTELLIGENCE,** for her incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

INSTITUTE

Vision:

- To serve the society by offering top-notch technical education on par with global standards.

Mission:

- Be a center of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all – round personalities respecting moral and ethical values.

DEPARTMENT

Vision:

- To excel in education, innovation, and research in Artificial Intelligence and Data Science to fulfil industrial demands and societal expectations.

Mission

- To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.
- To collaborate with industry and offer top-notch facilities in a conducive learning environment.
- To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.
- To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

PROGRAM EDUCATIONAL OBJECTIVES (PEO)

- **PEO1:** Compete on a global scale for a professional career in Artificial Intelligence and Data Science.
- **PEO2:** Provide industry-specific solutions for the society with effective communication and ethics.
- **PEO3** Enhance their professional skills through research and lifelong learning initiatives.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Capable of finding the important factors in large datasets, simplify the data, and improve predictive model accuracy.
- **PSO2:** Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals, and an engineering specialization to develop solutions to complex engineering problems.

2. **Problem analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.

3. **Design/development of solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required.

4. **Conduct investigations of complex problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions.

5. **Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems.

6. **The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment.

7. **Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws.

8. **Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

9. **Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.

10. **Project management and finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

11. **Life-long learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.

ABSTRACT

This project presents a user-friendly and interactive **Shiny web application** designed for detecting anomalies in bank transaction data. The goal is to identify unusual or suspicious transactions that may indicate fraudulent activity using statistical methods. The application enables users to upload transaction datasets in CSV format and apply real-time filtering based on date range, Z-score thresholds, and Interquartile Range (IQR) coefficients.

The system leverages two common outlier detection techniques: **Z-Score** and **IQR-based filtering**. Z-Score detects anomalies based on deviations from the mean, while IQR captures transactions that fall significantly outside the typical transaction range. The application visualizes the distribution of transaction amounts using histograms with dynamic boundary lines to highlight potential outliers.

Users can explore raw data, view detected anomalies, and analyze key summary statistics, such as the number of anomalies and unique account identifiers. The tool is valuable for banking professionals and data analysts seeking to automate and visualize the process of anomaly detection in financial datasets.

ABSTRACT WITH POs AND PSOs MAPPING

CO 5 : BUILD DATA SCIENCE USING R PROGRAMMING FOR SOLVING REAL-TIME PROBLEMS.

ABSTRACT	POs MAPPED	PSOs MAPPED
This project is an interactive **Shiny web application** for detecting anomalies in bank transactions. Users can upload CSV data and apply filters to identify suspicious transactions using **Z-Score** and **IQR-based** methods. The app displays raw data, highlights anomalies, and provides visualizations like histograms and summary statistics. It offers a simple, effective tool for financial anomaly detection and fraud analysis.	**PO1 -3** **PO2 -3** **PO3 -3** **PO4 -3** **PO5 -3** **PO9 -2** **PO10 -2** **PO11-2**	**PSO1 -3** **PSO2 -3**

Note: 1- Low, 2-Medium, 3- High

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 Objective

To build an interactive Shiny web application using R that detects anomalies in bank transactions. The app enables users to upload CSV files and filter data by date and statistical thresholds. It applies Z-Score and IQR techniques to flag suspicious transactions. The system offers clear visualizations such as histograms and summary tables. This tool aims to assist financial institutions in early fraud detection. It also highlights the application of data science for solving real-world problems effectively.

1.2 Overview

This project develops a Shiny app in R for anomaly detection in banking transactions. Users can upload their transaction data and apply filters like date range, Z-score threshold, and IQR coefficient. The app uses statistical methods to identify outliers that may indicate fraudulent or unusual activities. Results are shown through interactive tables and informative histograms. The user-friendly interface makes it accessible for analysts and auditors. It provides real-time feedback and helps in quick decision-making. This tool demonstrates how data science can be applied practically in finance. By visualizing anomalies, it supports better financial oversight. Overall, it is an efficient solution for fraud detection using R programming.

1.3 R Programming Concepts Used

This project incorporates various R programming concepts to build an effective and interactive anomaly detection system. The **Shiny framework** is used to design a dynamic web application, enabling real-time interaction between the user and data. **Reactive programming** plays a key role, with reactive() and eventReactive() ensuring outputs update only when relevant inputs change. **File handling** is managed using fileInput() and read.csv() to load user-uploaded transaction data. **Data manipulation** is performed using the **dplyr** package, leveraging functions like filter() and mutate() for efficient processing. Date inputs are managed with as.Date() and dateRangeInput() to filter transactions by time range. The application detects anomalies using **statistical calculations** like mean, standard deviation (Z-Score), and interquartile range (IQR), calculated via built-in R functions. Conditional logic is used to flag anomalies based on thresholds set by the user.

For **data visualization**, **ggplot2** is used to generate histograms with visual markers for IQR bounds. Tables are rendered interactively using the **DT** package, allowing for user-friendly browsing of both raw and filtered data. The UI is organized using functions like fluidPage(), sidebarLayout(), and mainPanel() for a clean layout. User actions, such as running detection, are triggered through actionButton() and handled reactively. The app also uses req() to validate inputs like file uploads before proceeding. Different types of outputs, including plots, tables, and summary text, are rendered using renderPlot(), renderDT(), and renderPrint(). Integration of multiple R packages (shiny, dplyr, ggplot2, DT) demonstrates modular design and robust package usage. Additionally, numeric inputs (numericInput()) allow users to fine-tune parameters, ensuring a flexible and effective anomaly detection process.

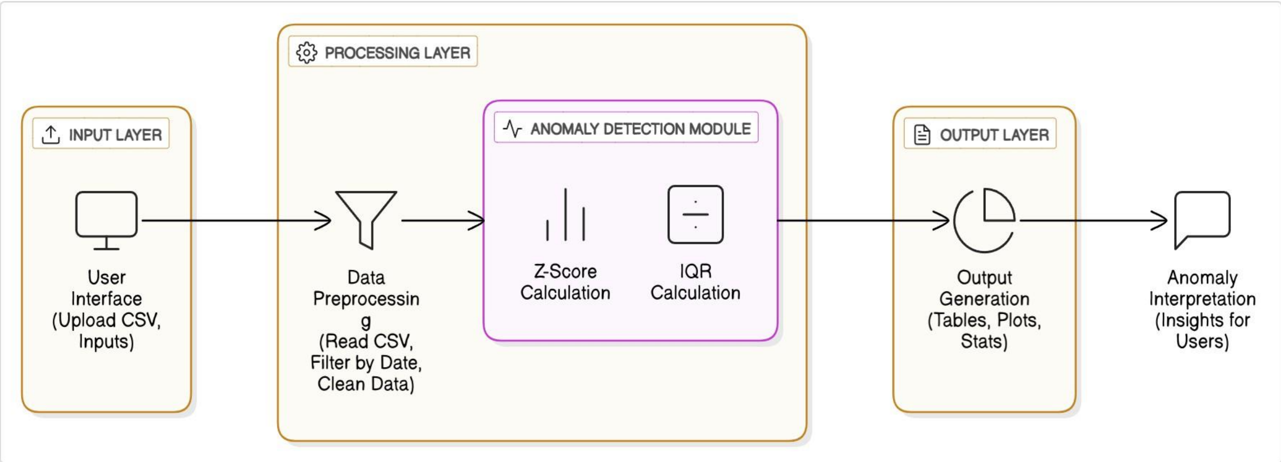
CHAPTER 2
PROJECT METHODOLOGY

2.1 Proposed Work

The proposed work aims to design and implement a user-interactive Shiny web application in R for detecting anomalies in bank transaction data. The system is intended to assist users, such as auditors or analysts, in identifying potentially fraudulent or unusual financial transactions. Users can upload a CSV file containing transaction records and apply filters such as date range and statistical thresholds to analyze the data. The app applies two key statistical methods: **Z-Score**, which measures how far a transaction deviates from the mean, and **Interquartile Range (IQR)**, which identifies outliers beyond typical value ranges.

The application uses R’s powerful data manipulation capabilities via the **dplyr** package and dynamic visualizations through **ggplot2** to enhance interpretability. Data is presented in structured tables using the **DT** package, with separate views for raw and anomalous records. The real-time responsiveness is achieved through **Shiny's reactive programming**, which updates visual and tabular outputs as users adjust parameters. Additionally, summary statistics and histograms are provided to offer quick overviews of data patterns and outliers. This project serves as a practical demonstration of applying data science techniques in R to a real-world problem, offering both analytical power and ease of use through an intuitive interface.

2.2 Block Diagram



CHAPTER 3
MODULE DESCRIPTION

3.1 Data Upload Module

This module allows the user to upload a CSV file containing bank transaction data. It uses Shiny's fileInput() feature to select and load the file from the user's local system. The data typically includes fields such as date, account ID, and transaction amount. This module validates the uploaded file and ensures the correct structure is available for further processing.

3.2 Preprocessing & Cleaning Module

Once the data is uploaded, this module performs data cleaning and preprocessing tasks. It converts date fields to proper Date formats, filters transactions within the user-specified date range, and handles any missing or inconsistent values. This step ensures the dataset is clean, consistent, and ready for anomaly detection operations.

3.3 Z-score Anomaly Detection Module

This module implements statistical anomaly detection using the Z-score method. It calculates the mean and standard deviation of transaction amounts, and then computes the Z-score for each transaction. Transactions with Z-scores beyond a user-defined threshold are flagged as anomalies. This technique is effective in detecting values that deviate significantly from the average.

3.4 IQR Outlier Detection Module

In this module, the Interquartile Range (IQR) method is used to identify outliers. It calculates the first (Q1) and third (Q3) quartiles and determines the IQR. Any transaction amounts that fall below Q1 - k×IQR or above Q3 + k×IQR are marked as outliers, where *k* is a coefficient chosen by the user. This method is robust for skewed data and non-normal distributions.

3.5 Visualization Module

This module is responsible for visually presenting the data and anomalies. It uses the ggplot2 package to create histograms that illustrate the distribution of transaction amounts, along with reference lines indicating outlier thresholds. It also uses the DT package to render interactive tables for both raw data and detected anomalies, enhancing interpretability.

3.6 Summary & Reporting Module

The final module provides a textual summary of the analysis. It displays key metrics such as the total number of transactions, number of anomalies detected by each method, and count of unique accounts. This concise report helps users quickly understand the results of the anomaly detection process and supports informed decision-making.

CHAPTER 4
CONCLUSION & FUTURE SCOPE

Conclusion

This project successfully demonstrates the use of R programming and Shiny to create an interactive and user-friendly web application for detecting anomalies in bank transaction data. By implementing both Z-score and IQR-based methods, the system effectively identifies outliers and potentially suspicious transactions. The use of visualizations such as histograms and dynamic tables enhances the user's ability to interpret the results. Overall, the application provides a valuable tool for financial data analysis and supports early detection of irregularities that could indicate fraud or errors.

Future Scope

In the future, this project can be extended in several ways:

- Integration of Machine Learning Models: Incorporate supervised or unsupervised learning algorithms to enhance the accuracy of anomaly detection.
- User Authentication: Add login and role-based access to secure sensitive financial data.
- Real-time Data Streaming: Enable support for live transaction feeds instead of static CSV files.
- Dashboard Enhancements: Include additional visualizations like time-series graphs and account-based filtering.
- Alert System: Implement automatic email or SMS notifications when anomalies are detected.

These improvements would make the system more scalable, intelligent, and applicable to real-world banking environments.

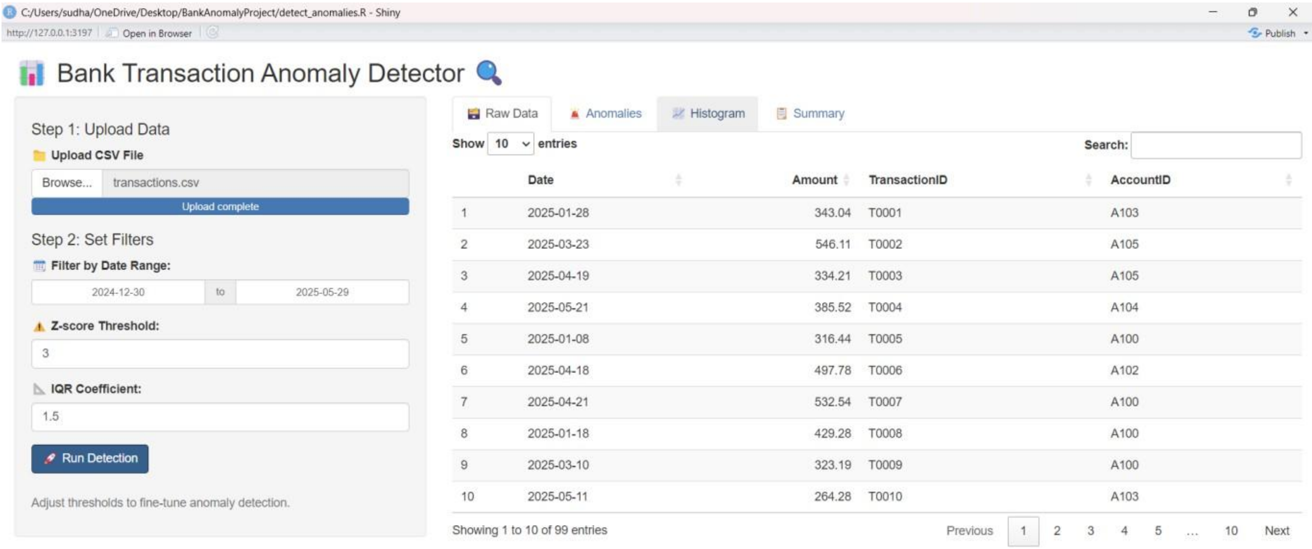
CHAPTER 5
APPENDIX A – SOURCE CODE

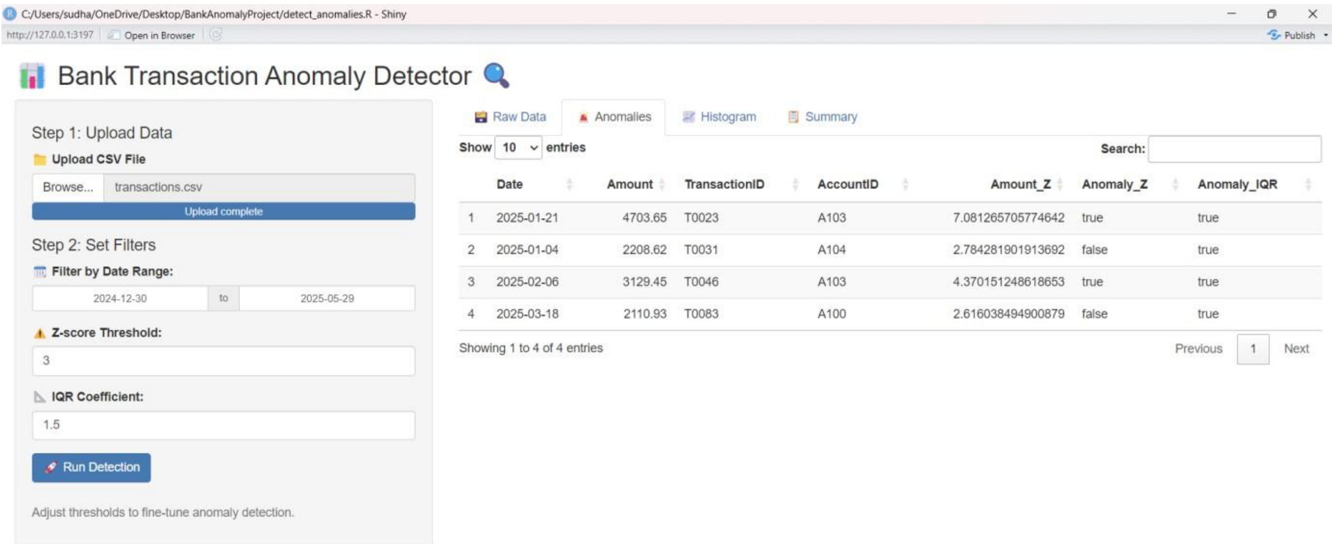
```
# app.R: Creative Shiny App for Bank Transaction Anomaly Detection
# Place this file and your transactions.csv in the same directory, then run `shiny::runApp(".")`

library(shiny)
library(dplyr)
library(ggplot2)
library(DT)

ui <- fluidPage(
  titlePanel("📊 Bank Transaction Anomaly Detector 🔍"),
  sidebarLayout(
    sidebarPanel(
      h4("Step 1: Upload Data"),
      fileInput("file", "📁 Upload CSV File", accept = c(".csv")),
      h4("Step 2: Set Filters"),
      dateRangeInput("dates", "📅 Filter by Date Range:", start = Sys.Date() - 150, end =
Sys.Date()),
      numericInput("z_thresh", "⚠️ Z-score Threshold:", value = 3, min = 0, step = 0.5),
      numericInput("iqr_coef", "📐 IQR Coefficient:", value = 1.5, min = 0, step = 0.1),
      actionButton("run", "🚀 Run Detection", class = "btn-primary"),
      br(), br(),
      helpText("Adjust thresholds to fine-tune anomaly detection.")
    ),
    mainPanel(
      tabsetPanel(
        tabPanel("🗃️ Raw Data", DTOutput("table")),
        tabPanel("🚨 Anomalies", DTOutput("anomalyTable")),
        tabPanel("📈 Histogram", plotOutput("histPlot")),
        tabPanel("📋 Summary", verbatimTextOutput("summaryText"))
      )
    )
  )
)

server <- function(input, output, session) {
  data <- eventReactive(input$run, {
    req(input$file)
    df <- read.csv(input$file$datapath, stringsAsFactors = FALSE)
    df$Date <- as.Date(df$Date)
```

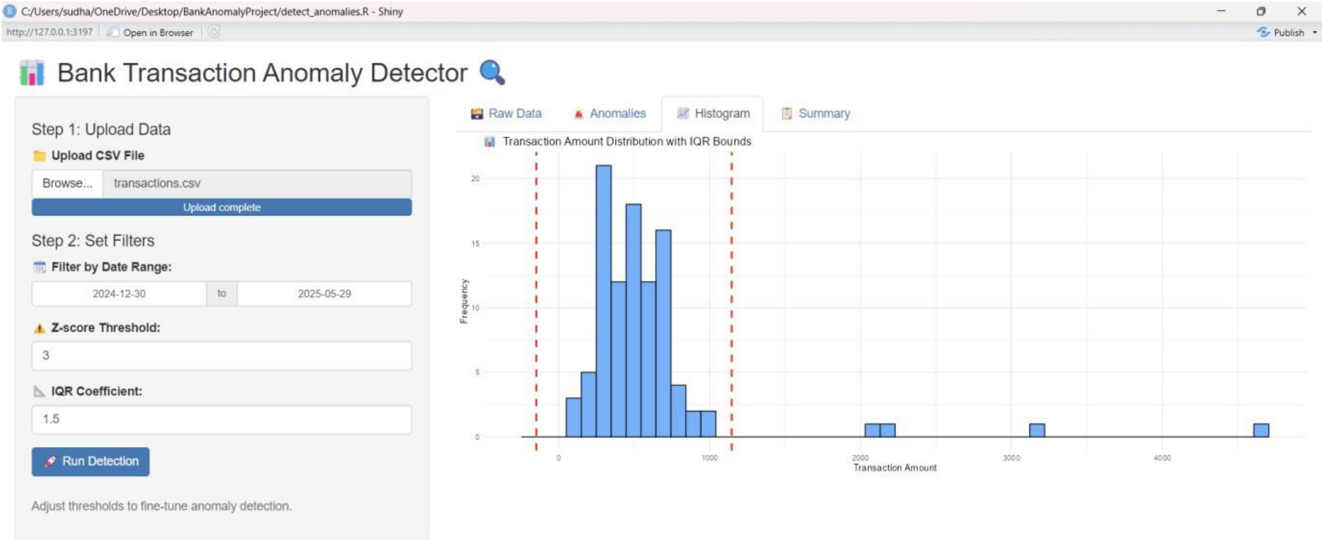
```
    if (!is.null(input$dates)) {
      df <- df %>% filter(Date >= input$dates[1] & Date <= input$dates[2])
    }
    df
  })

  processed <- reactive({
    df <- data()
    mean_amt <- mean(df$Amount)
    sd_amt <- sd(df$Amount)
    df <- df %>%
      mutate(
        Amount_Z = (Amount - mean_amt) / sd_amt,
        Anomaly_Z = abs(Amount_Z) > input$z_thresh
      )
    Q1 <- quantile(df$Amount, 0.25)
    Q3 <- quantile(df$Amount, 0.75)
    IQRv <- Q3 - Q1
    lower <- Q1 - input$iqr_coef * IQRv
    upper <- Q3 + input$iqr_coef * IQRv
    df <- df %>%
      mutate(
        Anomaly_IQR = (Amount < lower) | (Amount > upper)
      )
    df
  })

  output$table <- renderDT({
    datatable(data(), options = list(pageLength = 10, scrollX = TRUE))
  })

  output$anomalyTable <- renderDT({
    df <- processed()
    datatable(df %>% filter(Anomaly_Z | Anomaly_IQR), options = list(pageLength = 10,
scrollX = TRUE))
  })

  output$histPlot <- renderPlot({
    df <- processed()
    Q1 <- quantile(df$Amount, 0.25)
    Q3 <- quantile(df$Amount, 0.75)
    IQRv <- Q3 - Q1
    lower <- Q1 - input$iqr_coef * IQRv
    upper <- Q3 + input$iqr_coef * IQRv
    ggplot(df, aes(x = Amount)) +
```

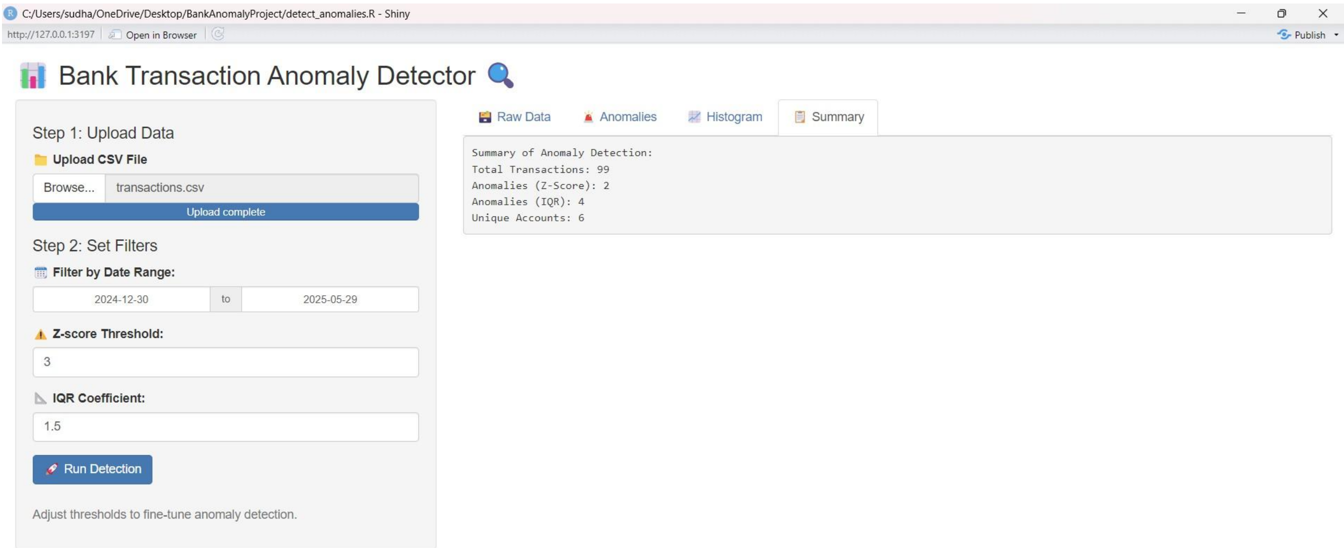
```
      geom_histogram(bins = 50, fill = "#66b2ff", color = "black") +
      geom_vline(xintercept = c(lower, upper), linetype = "dashed", color = "red", size = 1) +
      labs(
        title = "📊 Transaction Amount Distribution with IQR Bounds",
        x = "Transaction Amount",
        y = "Frequency"
      ) +
      theme_minimal()
  })

  output$summaryText <- renderPrint({
    df <- processed()
    cat("Summary of Anomaly Detection:\n")
    cat("Total Transactions:", nrow(df), "\n")
    cat("Anomalies (Z-Score):", sum(df$Anomaly_Z), "\n")
    cat("Anomalies (IQR):", sum(df$Anomaly_IQR), "\n")
    cat("Unique Accounts:", length(unique(df$AccountID)), "\n")
  })
}

shinyApp(ui = ui, server = server)
```

Appendix B – Screenshots






Bank Transaction Anomaly Detector
Step 1: Upload Data
Upload CSV File
Browse...
transactions.csv
Upload complete
Step 2: Set Filters
Filter by Date Range:
2024-12-30
to
2025-05-29
Z-score Threshold:
3
IQR Coefficient:
1.5
Run Detection
Adjust thresholds to fine-tune anomaly detection.
Raw Data
Anomalies
Histogram
Summary
Transaction Amount Distribution with IQR Bounds
Frequency
Transaction Amount


Bank Transaction Anomaly Detector
Step 1: Upload Data
Upload CSV File
Browse...
transactions.csv
Upload complete
Step 2: Set Filters
Filter by Date Range:
2024-12-30
to
2025-05-29
Z-score Threshold:
3
IQR Coefficient:
1.5
Run Detection
Adjust thresholds to fine-tune anomaly detection.
Raw Data
Anomalies
Histogram
Summary
Summary of Anomaly Detection:
Total Transactions: 99
Anomalies (Z-Score): 2
Anomalies (IQR): 4
Unique Accounts: 6

REFERENCES

1. Aggarwal, C. C. (2017). *Outlier Analysis* (2nd ed.). Springer.

2. Chang, W., Cheng, J., Allaire, J. J., Xie, Y., & McPherson, J. (2021). *Shiny: Web Application Framework for R*. R package version 1.7.1. https://shiny.rstudio.com

3. Wickham, H., & Grolemund, G. (2016). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. O'Reilly Media.