UNIVERSITÀ DI PISA

Department of Computer Science

Master in Data Science & Business Informatics

“Heart Disease Prediction With Pyspark”

Distributed Data Analysis - Big Data

Submission Date: 07/12/2024

Group ID_09

Sana Afreen (681744) | Zhiqi Zui (702295)

Prof. Valentina Panesula

Heart Disease Prediction On A Medical Dataset With PySpark

Sana Afreen & Zhiqi Zhu
Distributed Data Analysis and Mining, University of Pisa

December 7, 2024

Contents

1 Introduction

1.1 Research Background

Heart disease prediction is crucial for public health, as early identification of at-risk individuals can improve healthcare outcomes. This project applies distributed machine learning techniques to a large-scale dataset, demonstrating their effectiveness in predictive health analysis.

This project uses a synthetic version of the Statlog Heart dataset, originally from the UCI Machine Learning Repository [1], to predict the presence of heart disease. The dataset, sourced from OpenML [2], contains over one million records and is designed for scalable Distributed Data Analysis and Mining (DDAM).

1.2 Research Objective

The main objective of this study is to apply machine learning models with spark techniques to predict the presence or absence of the heart disease. In this case, our target variable is:

- **Class 0:** Absence of heart disease.
- **Class 1:** Presence of heart disease.

In addition to exploring supervised learning approaches, this study also aims to investigate potential unsupervised patterns within the data. By doing so, we seek to uncover underlying structures or clusters that may provide further insights into the prediction process and contribute to the development of more accurate and robust models.

1.3 Description of the Variables

The dataset comprises various numerical and categorical variables, all of which have been smoothed or expanded into numerical ranges during the synthesis process.

1.3.1 Input Variables

- **Age:** Numerical, representing the age of the individual.
- **Sex:** Numerical, encoded as 0.0 for female and 1.0 for male.
- **Chest Pain Type:** Numerical, derived from original categorical values (e.g., typical angina, atypical angina, asymptomatic) and expanded to a continuous range.
- **Resting Blood Pressure:** Numerical, measured in mmHg.
- **Serum Cholesterol:** Measured in mg/dl, smoothed for realistic distribution.
- **Fasting Blood Sugar:** Encoded as 1.0 if >120 mg/dl, otherwise 0.0.
- **Resting Electrocardiographic Results:** Encoded as:
  - 0.0: Normal
  - 1.0: ST-T wave abnormality

  - 2.0: Indicating left ventricular hypertrophy.

- **Maximum Heart Rate Achieved:** Numerical, expanded for higher resolution.
- **Exercise-Induced Angina:** Encoded as 1.0 (Yes) or 0.0 (No).
- **Oldpeak:** Numerical, representing ST depression induced by exercise.
- **Slope:** Encoded as:
  - 1.0: Upsloping
  - 2.0: Flat
  - 3.0: Downsloping.
- **Number of Major Vessels:** Numerical, ranging from 0.0 to 3.0.
- **Thalassemia (Thal):** Encoded as:
  - 3.0: Normal
  - 6.0: Fixed defect
  - 7.0: Reversible defect.

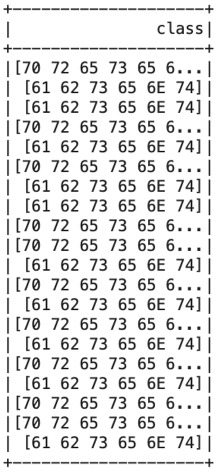
1.3.2 Target Variable

- **Class:** Encoded as 0.0 (absence of disease) or 1.0 (presence of disease).

2 Data Preprocessing

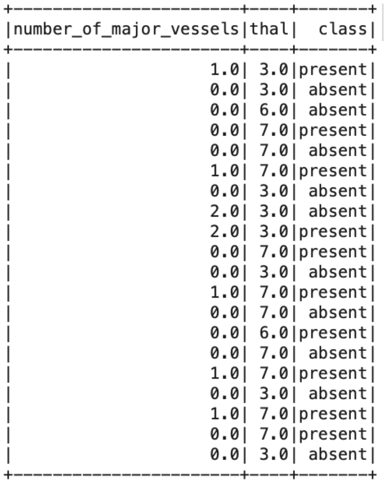
2.1 Data Transformation

When initially loading our dataset, we encountered an issue where the 'class' column was represented as byte sequences instead of meaningful categorical values due to PySpark automatically interpreting non-numeric categorical data as bytes during import (as shown in Figures 1 and 2).



```
+--------------------+
|               class|
+--------------------+
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
|[70 72 65 73 65 6...|
|[70 72 65 73 65 6...|
| [61 62 73 65 6E 74]|
+--------------------+
```

Figure 1: Class as bytes.



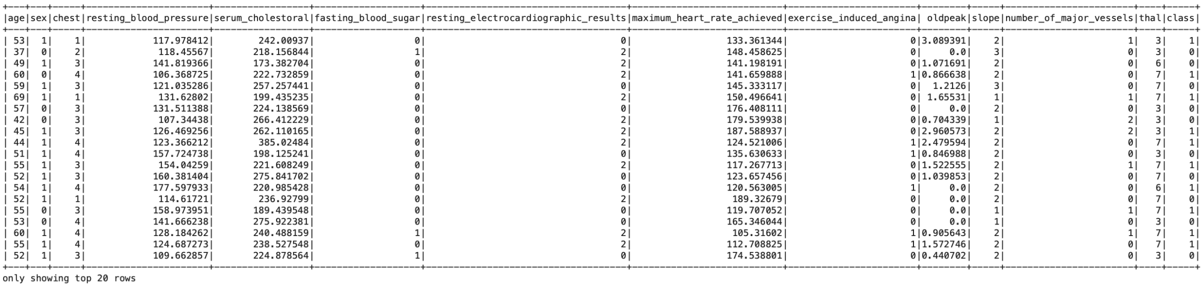
```
+-----------------------+----+-------+
|number_of_major_vessels|thal|  class|
+-----------------------+----+-------+
|                    1.0| 3.0|present|
|                    0.0| 3.0| absent|
|                    0.0| 6.0| absent|
|                    0.0| 7.0|present|
|                    0.0| 7.0| absent|
|                    1.0| 7.0|present|
|                    0.0| 3.0| absent|
|                    2.0| 3.0| absent|
|                    2.0| 3.0|present|
|                    0.0| 7.0|present|
|                    0.0| 3.0| absent|
|                    1.0| 7.0|present|
|                    0.0| 7.0| absent|
|                    0.0| 6.0|present|
|                    0.0| 7.0| absent|
|                    1.0| 7.0|present|
|                    0.0| 3.0| absent|
|                    1.0| 7.0|present|
|                    0.0| 7.0|present|
|                    0.0| 3.0| absent|
+-----------------------+----+-------+
```

Figure 2: Class as categories.

One interesting insight we discovered in this dataset was that the 'age' variable, which should be integer, was stored as a float. Similarly, some other categorical variables, despite being encoded, were represented as floats with unnecessarily long decimal tails.This section presents an initial analysis of the dataset, including summary statistics and visualizations to identify patterns, correlations, and potential outliers.

2.2 Data Cleaning & Encoding

To address this issue, we removed the extraneous decimal points from variables such as `age`, `sex`, `exercise_induced_angina`, and `number_of_major_vessels`, aiming to make the data more consistent and logically coherent. This cleaning step helped improve the overall quality of the dataset and ensured that the variables better aligned with their intended data types. The cleaned dataset is shown in Figure 3.



```
+---+---+-----+----------------------+-----------------+-------------------+------------------------------------+---------------------------+-----------------------+--------+-----+-----------------------+----+-----+
|age|sex|chest|resting_blood_pressure|serum_cholestoral|fasting_blood_sugar|resting_electrocardiographic_results|maximum_heart_rate_achieved|exercise_induced_angina| oldpeak|slope|number_of_major_vessels|thal|class|
+---+---+-----+----------------------+-----------------+-------------------+------------------------------------+---------------------------+-----------------------+--------+-----+-----------------------+----+-----+
| 53|  1|    1|            117.978412|        242.00937|                  0|                                   0|                 133.361344|                      0|3.089391|    2|                      1|   3|    1|
| 37|  0|    2|             118.45567|       218.156844|                  1|                                   2|                 148.458625|                      0|     0.0|    3|                      0|   3|    0|
| 49|  1|    3|            141.819366|       173.382704|                  0|                                   2|                 141.198191|                      0|1.071691|    2|                      0|   6|    0|
| 60|  0|    4|            106.368725|       222.732859|                  0|                                   2|                 141.659888|                      1|0.866638|    2|                      0|   7|    1|
| 59|  1|    3|            121.035286|       257.257441|                  0|                                   0|                 145.333117|                      0|  1.2126|    3|                      0|   7|    0|
| 69|  1|    1|             131.62802|       199.435235|                  0|                                   2|                 150.496641|                      0| 1.65531|    1|                      1|   7|    1|
| 57|  0|    3|            131.511388|       224.138569|                  0|                                   0|                 176.408111|                      0|     0.0|    2|                      0|   3|    0|
| 42|  0|    3|             107.34438|       266.412229|                  0|                                   2|                 179.539938|                      0|0.704339|    1|                      2|   3|    0|
| 45|  1|    3|            126.469256|       262.110165|                  0|                                   2|                 187.588937|                      0|2.960573|    2|                      2|   3|    1|
| 44|  1|    4|            123.366212|        385.02484|                  0|                                   2|                 124.521006|                      1|2.479594|    2|                      0|   7|    1|
| 51|  1|    4|            157.724738|       198.125241|                  0|                                   0|                 135.630633|                      1|0.846988|    2|                      0|   3|    0|
| 55|  1|    3|             154.04259|       221.608249|                  0|                                   2|                 117.267713|                      0|1.522555|    2|                      1|   7|    1|
| 52|  1|    3|            160.381404|       275.841702|                  0|                                   0|                 123.657456|                      0|1.039853|    2|                      0|   7|    0|
| 54|  1|    4|            177.597933|       220.985428|                  0|                                   0|                 120.563005|                      1|     0.0|    2|                      0|   6|    1|
| 52|  1|    1|             114.61721|        236.92799|                  0|                                   2|                  189.32679|                      0|     0.0|    2|                      0|   7|    0|
| 55|  0|    3|            158.973951|       189.439548|                  0|                                   0|                 119.707052|                      0|     0.0|    1|                      1|   7|    1|
| 53|  0|    4|            141.666238|       275.922381|                  0|                                   0|                 165.346044|                      0|     0.0|    1|                      0|   3|    0|
| 60|  1|    4|            128.184262|       240.488159|                  1|                                   2|                  105.31602|                      1|0.905643|    2|                      1|   7|    1|
| 55|  1|    4|            124.687273|       238.527548|                  0|                                   2|                 112.708825|                      1|1.572746|    2|                      0|   7|    1|
| 52|  1|    3|            109.662857|       224.878564|                  1|                                   0|                 174.538801|                      0|0.440702|    2|                      0|   3|    0|
+---+---+-----+----------------------+-----------------+-------------------+------------------------------------+---------------------------+-----------------------+--------+-----+-----------------------+----+-----+
only showing top 20 rows
```

Figure 3: Cleaned dataset after removing extraneous decimal points.

3 Exploratory Data Analysis

This section presents an initial analysis of the dataset, including summary statistics and visualizations to identify patterns, correlations, and potential outliers.

3.1 Feature Correlations

To analyze the relationships between features in our dataset using PySpark, we leveraged several key functionalities. First, we used the VectorAssembler to combine all numerical feature columns into a single vector column, which is a prerequisite for correlation analysis in PySpark. Next, we utilized the Correlation module from 'pyspark.ml.stat' to compute the Pearson correlation matrix, allowing us to quantify pairwise relationships between features. The resulting correlation matrix is visualized in Figure 4.

From the heatmap, it is evident that there are no strong correlations between the variables. Thus, there is no need for dimensionality reduction.

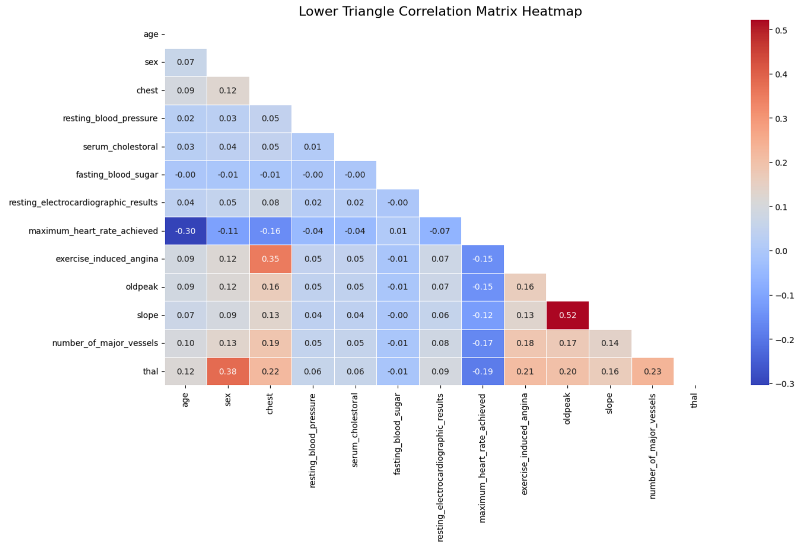


Figure 4: Heatmap of feature correlations based on the Pearson correlation matrix.

3.2 Distribution Analysis

We also analyzed the distribution of features to better understand the dataset. Most of them are categorical and numeric features. Below are the histograms of these features showing their distributions. Notably, the `age` histogram highlighted that most individuals were aged 55–65, with a noticeable spike at 60 (Figure 5).

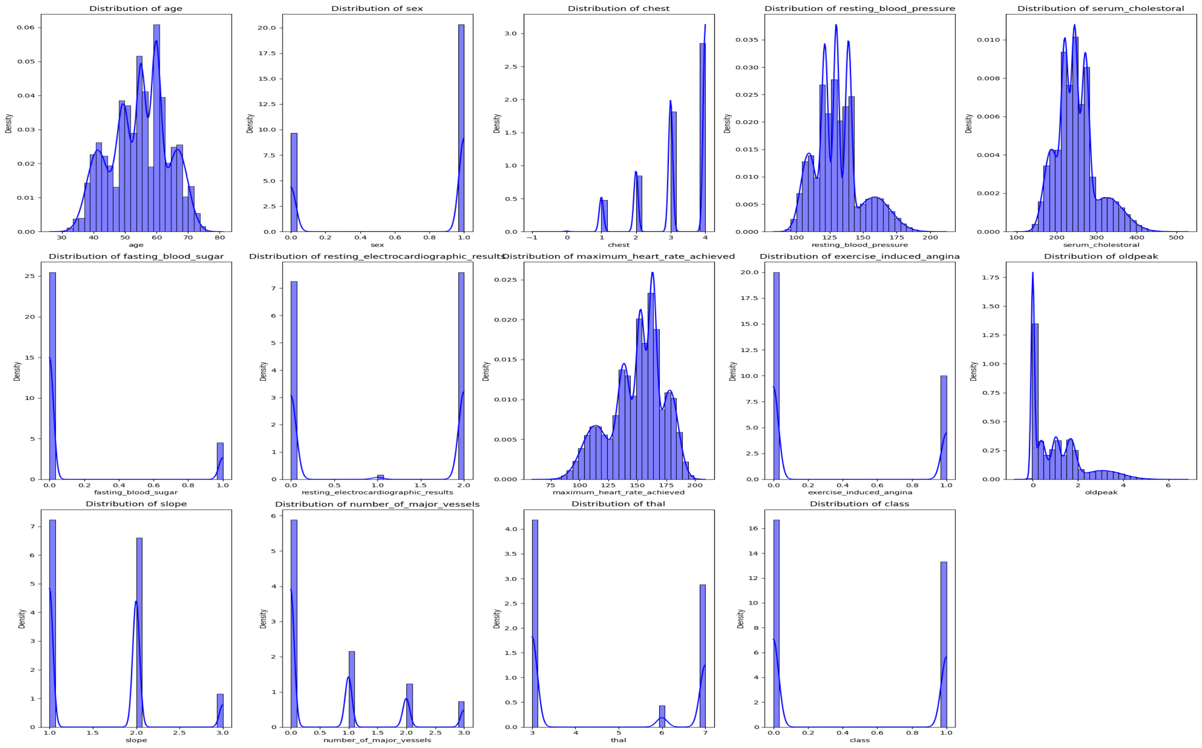
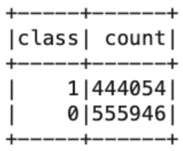


Figure 5: Histograms showing the distributions of numerical features.

3.3 Balanced Data Classification

Fortunately, our data is perfectly balanced, as shown in Figure 6. This means there is no need to perform imbalance handling techniques before applying machine learning models.

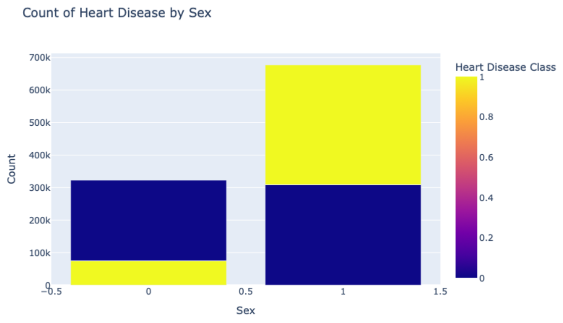


```
+-----+------+
|class| count|
+-----+------+
|    1|444054|
|    0|555946|
+-----+------+
```

Figure 6: Visualization of the perfectly balanced dataset.

3.4 Demographic Insights on Heart Disease

We also explored the relationships between age groups, sex, and other features in relation to heart disease. Below are some interesting insights, as illustrated in Figures 7 and 8.



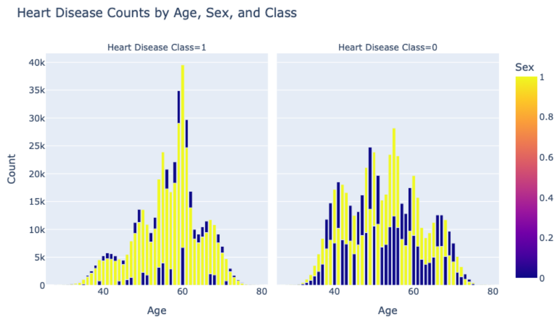


Figure 7: Heart Disease by sex

Figure 8: Heart Disease by Age & Sex

This analysis highlights the relationship between sex and heart disease. In our dataset, sex = 1 represents males and sex = 0 represents females. The findings indicate that males have a higher likelihood of facing heart disease compared to females.

4 Machine Learning

We primarily used three-fold cross-validation combined with grid search to train the model on randomly split training sets and identify the optimal parameter combination for the best model performance.

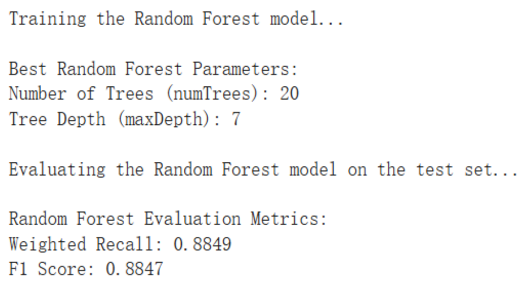
Regarding model evaluation, our main focus is on recall, as our objective is to minimize missed diagnoses (false negatives), particularly cases where patients truly have the condition but are predicted as healthy by the model. Recall serves as the primary evaluation metric because it reflects the model's ability to correctly identify actual positive cases. Additionally, when necessary, we also consider the F1 score to balance precision and recall for a more comprehensive assessment.

4.1 Supervised Learning

4.1.1 Random Forest

- Model Performance

For the Random Forest model, we found that the best performance in terms of recall was achieved when the number of trees was set to 20 and the depth was set to 7. Under these parameters, the recall and F1 score are shown in Figure 9, and the confusion matrix is presented in Figure 10.



```
Training the Random Forest model...

Best Random Forest Parameters:
Number of Trees (numTrees): 20
Tree Depth (maxDepth): 7

Evaluating the Random Forest model on the test set...

Random Forest Evaluation Metrics:
Weighted Recall: 0.8849
F1 Score: 0.8847
```

Figure 9: Recall and F1 Score for the Random Forest Model.

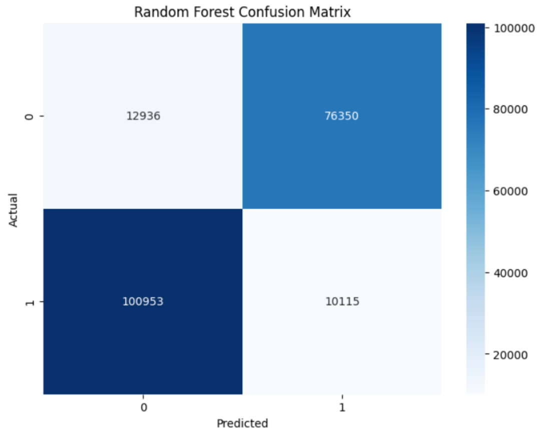
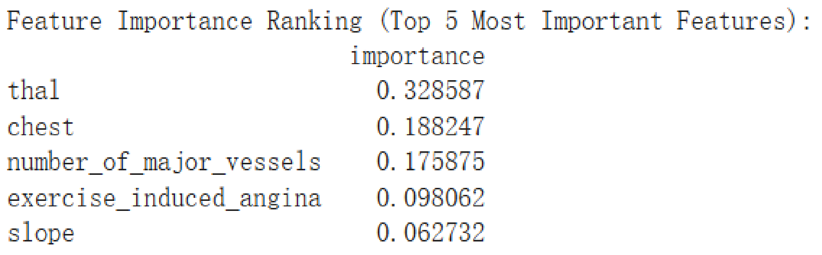


Figure 10: Confusion Matrix for the Random Forest Model.

- Feature Importance Ranking

In addition, based on the Random Forest model, we identified the most important variables that have the greatest impact on the target variable in this dataset. The specific details are shown in Figure 11.



```
Feature Importance Ranking (Top 5 Most Important Features):
                          importance
thal                        0.328587
chest                       0.188247
number_of_major_vessels     0.175875
exercise_induced_angina     0.098062
slope                       0.062732
```

Figure 11: Important Features for Predicting Heart Disease (Random Forest Model).

4.1.2 Logistic Regression

Next, we used the Logistic Regression model to perform the prediction task, as it is one of the primary algorithms best suited for binary classification problems.

- Feature scaling

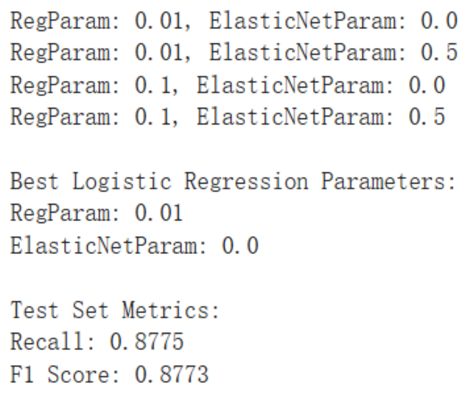
However, since logistic regression is based on gradient optimization, it is sensitive to the range of features. Therefore, we needed to perform feature scaling on some variables. In this dataset, most of the features have a narrow range, so we primarily standardized the following numerical variables: age, blood pressure, cholesterol, maximum heart rate, and ST depression.

- One-hot coding

Additionally, we applied one-hot encoding to the chest variable, as it is an unordered categorical variable. After performing these feature engineering steps, we proceeded with training the model and tuning the parameters.

- Result

For the Logistic Regression model, we found that the best performance in terms of recall was achieved with the following optimal parameters: RegParam set to 0.01 and ElasticNetParam set to 0.0. Under these conditions, the recall and F1 score are shown in Figure 12, and the confusion matrix is presented in Figure 13.



```
RegParam: 0.01, ElasticNetParam: 0.0
RegParam: 0.01, ElasticNetParam: 0.5
RegParam: 0.1, ElasticNetParam: 0.0
RegParam: 0.1, ElasticNetParam: 0.5

Best Logistic Regression Parameters:
RegParam: 0.01
ElasticNetParam: 0.0

Test Set Metrics:
Recall: 0.8775
F1 Score: 0.8773
```

Figure 12: Recall and F1 Score for the Logistic Regression Model.

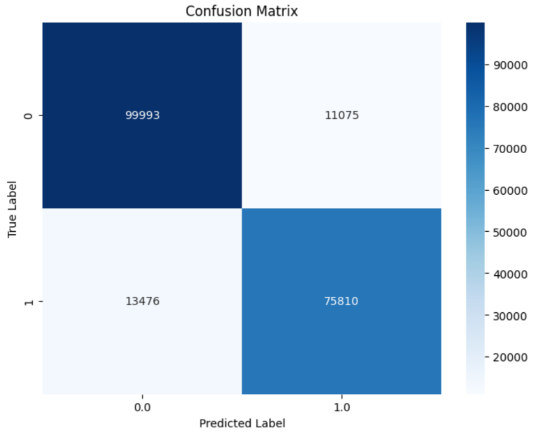


Figure 13: Confusion Matrix for the Logistic Regression Model.

4.2 Unsupervised Learning – Clustering

For the unsupervised learning model, we primarily selected K-means clustering due to its simplicity, efficiency, and widespread use in segmenting data into distinct groups based on feature similarity.

We first plotted the Elbow Method graph to determine the optimal number of clusters. Based on the Elbow Method, we initially identified that $k = 3$ and $k = 4$ were potential optimal cluster numbers. The resulting plot is shown in Figure 14.

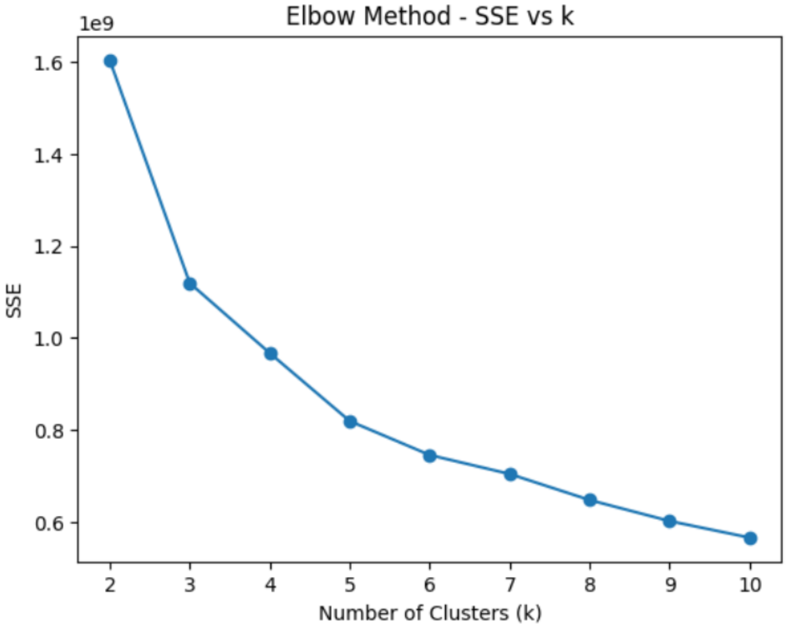


Figure 14: Elbow Method for Optimal Number of Clusters.

Next, we calculated the silhouette score and used it to identify the best parameter combination. Based on this score, we trained the final model using the optimal parameters.

5 Conclusions

Both supervised learning algorithms performed quite well. Under the optimal parameter settings for each model, both Random Forest and Logistic Regression achieved recall and F1 scores around 0.87. However, the Random Forest model slightly outperformed Logistic Regression, with a recall score of 0.8849 compared to 0.8775 for Logistic Regression.

Next, for the unsupervised learning task with K-means clustering, we initially observed that $k = 3$ or $k = 4$ yielded the best results. We then attempted to evaluate the quality of these cluster numbers to identify the optimal parameters.

References

[1] UCI Machine Learning Repository. *Statlog Heart Dataset*. Retrieved from: `https://archive.ics.uci.edu/dataset/145/statlog+heart`.

[2] OpenML Synthetic Dataset. *Dataset ID: 267*. Retrieved from: `https://www.openml.org/d/267`.