GLUCO RISK ANALYZER USING SVM-RF-LR

ADM Course Project Report

in partial fulfilment of the degree

Bachelor of Technology
in
Computer Science & Engineering

By

Name T.Arshavardhini HTNo(2303A51600)

Name D.Varshitha HTNo(2303A51927)

Name M.kavyaSri HTNo(2303A51929)

Name G.Rithu Goud HTNo(2303A51641)

Under the guidance of

Bediga Sharan
Assistant Professor

Submitted to

School of Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the Applications of Data Mining Course Project Report entitled **"GLUCO RISK ANALYZER USING SVM-RF-LR"** is a record of bonafide work carried out by the student(s) ArshaVardhini Varshitha ,Kavyasri, RithuGoud bearing Hallticket No(s) 2303A51600, 2303A51927, 2303A51929, 2303A51641 during the academic year 2024-25 in partial fulfillment of the award of the degree of Bachelor of Technology in Computer Science & Engineering by the SR University, Warangal.

Supervisor
(Mr. Bediga Sharan)
Assistant Professor

Head of the Department
(Dr. M. Sheshikala)
Professor

ABSTRACT

- Diabetes is a chronic disease that poses serious health risks and can lead to life-threatening complications if not detected early. This project aims to leverage machine learning (ML) techniques for the early prediction of diabetes based on clinical parameters. The workflow begins with the collection of patient health records, including features such as glucose levels, blood pressure, BMI, insulin levels, age, and family history. These attributes are critical indicators used to assess the likelihood of diabetes.

- The collected data undergoes thorough preprocessing, including the handling of missing values, normalization of numerical attributes, and encoding of categorical data to ensure high-quality inputs for model training. The dataset is then split into training and testing sets to allow the models to learn patterns and validate their predictive performance.

- Multiple machine learning algorithms are implemented, including Logistic Regression, Random Forest, and Support Vector Machines (SVM). These models are trained and tested using the dataset, and their performances are evaluated using key metrics such as accuracy, precision, recall, and F1-score. Based on these results, the most accurate model is selected for deployment.

- The final model can predict whether a patient is likely to have diabetes, providing quick, data-driven support for healthcare professionals. This ML-based approach not only improves the speed and efficiency of diagnosis but also enables preventive care and personalized treatment planning, potentially reducing the long-term impacts of diabetes.

OBJECTIVE OF THE PROJECT

- The primary objective of this project is to develop a machine learning-based prediction model to identify the likelihood of diabetes in individuals based on medical and lifestyle-related parameters. By analyzing health data, the model aims to support early diagnosis, enabling preventive care and reducing the risk of complications.

- To apply various machine learning algorithms, particularly Random Forest, for classifying individuals as diabetic or non-diabetic. To design and implement three supervised learning algorithms—SVM, Random Forest, and Linear Regression—and evaluate their performance using metrics such as accuracy, F1-score, confusion matrix (for classification), or RMSE and R^2 (for regression), thereby determining the most appropriate model for the dataset.

- To preprocess and transform medical datasets to extract meaningful features such as glucose level, BMI, insulin, and age.To utilize data visualization techniques for better understanding of the dataset, patterns, and feature correlations.To evaluate model performance using standard classification metrics and identify the most effective predictive approach.To demonstrate how data mining can be utilized to build intelligent healthcare tools for disease prediction.

- This project highlights how machine learning can significantly enhance medical diagnostics and preventive health management.The aim of this project is to build predictive models using SVM, Random Forest, and Linear Regression in order to solve a real-world problem effectively. By training and validating these models on historical data, we seek to uncover insights, improve decision-making processes, and select the best-performing algorithm for practical deployment.

2. DEFINITIONS OF THE ELEMENTS USED IN THE PROJECT

The predictive model uses multiple health-related attributes. The following are key features used in the dataset:

✓ Glucose Level

Blood glucose concentration, a critical indicator of diabetes risk. Elevated glucose levels are a direct indication of potential diabetes, especially if observed consistently in fasting or postprandial states. It is one of the most significant predictors in diabetes diagnosis.

✓ BMI (Body Mass Index)

A person's weight in kilograms divided by the square of height in meters, used to assess obesity levels. It provides an estimate of body fat and is often used to classify individuals as underweight, normal, overweight, or obese. Higher BMI levels are correlated with increased risk of developing type 2 diabetes.

✓ Insulin

The insulin level in the blood, which can signal insulin resistance or sensitivity. Elevated insulin levels may signal insulin resistance, a condition where cells fail to respond to insulin effectively, often a precursor to diabetes.

✓ Blood Pressure

Blood pressure, particularly systolic pressure, is often associated with diabetes and metabolic syndrome. Hypertension can damage blood vessels and is commonly seen in people with insulin resistance or poorly controlled blood sugar levels.

✓ Pregnancies

This indicates the number of times a female patient has been pregnant. It helps identify gestational diabetes risk, which can lead to type 2 diabetes later in life.

✓ Diabetes Pedigree Function

This value quantifies the genetic influence by evaluating the history of diabetes in a person's family. A higher value suggests stronger hereditary risk. It combines family history with age and other personal health indicators.

➢ Logistic Regression :

Logistic Regression is a simple yet powerful statistical method used for binary classification problems, such as predicting whether a person is diabetic or not. It works by modeling the probability of a certain class (e.g., diabetic) using a logistic function, which outputs values between 0 and 1. Despite the name, it's actually used for classification rather than regression. One of its main advantages is its interpretability; you can clearly see the effect of each feature (like age, glucose level, BMI) on the prediction. Logistic Regression performs best when there is a linear relationship between the features and the outcome.

➢ Random Forest :

Random Forest is an ensemble learning method that builds a "forest" of decision trees during training. Each tree is trained on a random subset of the data and features, and the final prediction is made by taking a majority vote (for classification) or averaging (for regression). This technique significantly improves accuracy and reduces the risk of overfitting compared to a single decision tree. It handles missing values well and works effectively with both categorical and continuous data. In diabetes prediction, Random Forest can capture complex patterns and interactions between features that simpler models might miss.

➢ Support Vector Machine (SVM):

Support Vector Machine (SVM) is a powerful classification algorithm that works by finding the best boundary—or hyperplane—that separates different classes in the dataset. SVM focuses on maximizing the margin between the classes, making it very effective in high-dimensional spaces. For cases where data isn't linearly separable, SVM can use kernel functions to transform the data and find a separating hyperplane in the transformed space. In the context of diabetes prediction, SVM can be especially useful when the features have non-linear relationships and when high accuracy is crucial.

DESIGN

The design of this system focuses on constructing a robust and interpretable diabetes prediction model using machine learning algorithms. The system is organized into stages—data preprocessing, model training, performance evaluation, and result visualization—to ensure an end-to-end solution for medical diagnosis support.

1. System Architecture

- The project follows these key stages:

- **Data Collection:**

  Medical datasets such as the PIMA Indian Diabetes Dataset are collected containing values for glucose, insulin, BMI, age, etc.

2. Preprocessing and Feature Engineering:

- Handle missing values and inconsistencies.
- Normalize numerical features.
- Encode categorical variables (if any).
- Select relevant features for training.

3. Model Design:

- The Random Forest Classifier is used due to its ability to handle classification tasks efficiently, its robustness against overfitting, and high accuracy.

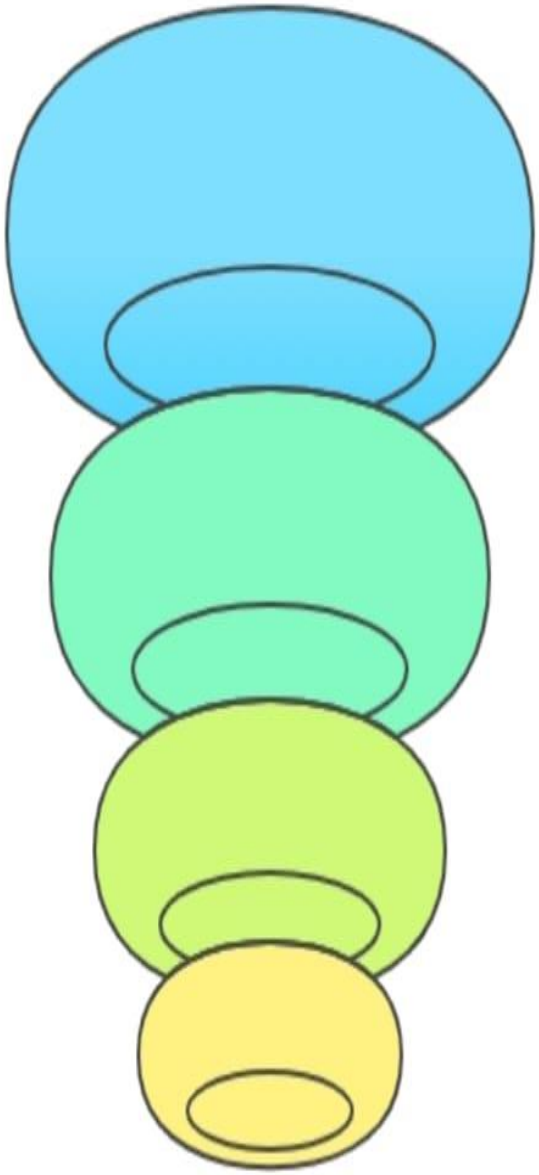
4. Training and Evaluation:

- The dataset is split into training and testing sets. Models are trained and evaluated using metrics such as accuracy, precision, recall, and F1-score.

5. Visualization:

- Graphs and plots are generated to show data distributions, feature correlations, and performance metrics.

Diabetes Prediction Process



Data Preprocessing

Cleaning and preparing data for analysis

Model Training

Applying machine learning models to data

Model Evaluation

Assessing model performance using metrics

Best Model Selection

Choosing the model with the highest F1 score

SCREENS

- Linear Regression, Support Vector Machine (SVM), and Random Forest are three fundamental machine learning algorithms, each with distinct characteristics and visual representation. Linear Regression is a simple yet powerful model used for predicting continuous numeric outcomes based on the assumption of a linear relationship between input features and the target variable. It fits a straight line (or a hyperplane in higher dimensions) through the data to minimize the error between predicted and actual values, which can be visualized in two dimensions as a line plotted through a scatter of data points.

- In contrast, Support Vector Machine (SVM) is primarily a classification algorithm that finds the optimal hyperplane that separates different classes by maximizing the margin between the closest data points (called support vectors) and the boundary. When visualized, SVM clearly shows a decision boundary, often with margins marked by dashed lines and support vectors highlighted; with non-linear kernels, these boundaries can curve to adapt to complex data structures.

- Lastly, Random Forest is an ensemble learning method that builds multiple decision trees using different subsets of the data and features. It outputs the majority vote for classification or the average prediction for regression. Though visualizing the entire forest is complex due to the number of trees, individual trees can be visualized, and feature importance plots can be generated to understand which features are most influential in making predictions.

- Additionally, in low-dimensional cases, the decision boundaries created by Random Forests appear as irregular, step-like regions reflecting the tree-based splits. Together, these models offer diverse perspectives on learning from data, and their visualizations help in interpreting their decision-making processes and understanding their strengths in different scenarios.

IMPLEMENTATION

The implementation phase turns the design into a functional system for diabetes prediction. The process includes coding in Python, using libraries such as pandas, NumPy, scikit-learn, matplotlib, and seaborn for preprocessing, training, evaluation, and visualization.

CODE

```
models = {
    "Logistic Regression": LogisticRegression(),
    "Random Forest": RandomForestClassifier(),
    "Support Vector Machine": SVC(probability=True)
}

# Initialize empty lists for metrics
model_names = []
accuracies = []
precisions = []
recalls = []

# Plot bars for each metric
bars_accuracy = ax.bar(x - width, accuracies, width, label='Accuracy', color='lightblue')
bars_precision = ax.bar(x, precisions, width, label='Precision', color='lightgreen')
bars_recall = ax.bar(x + width, recalls, width, label='Recall', color='lightcoral')

# Add labels, title, and customize the plot
ax.set_xlabel('Models')
ax.set_ylabel('Scores')
ax.set_title('Comparison of Models based on Performance Metrics')
ax.set_xticks(x)
ax.set_xticklabels(model_names)
ax.legend()
```

METHODOLOGY

1. Data Acquisition:

- ✓ Data acquisition refers to the process of collecting or obtaining relevant data that can be used to train, test, and validate your machine learning model. For diabetes prediction, the data usually includes patient health records, such as:
- ✓ Glucose levels
- ✓ Blood pressure
- ✓ BMI (Body Mass Index)
- ✓ Age
- ✓ Insulin levels
- ✓ Skin thickness
- ✓ Pregnancy history (in some datasets)
- ✓ Outcome label (0 = Non-diabetic, 1 = Diabetic)

2. Data Preprocessing:

- ✓ Data Collection
- ✓ Data Cleaning
- ✓ Handle Missing Values
- ✓ Remove Duplicates
- ✓ Handle Outliers
- ✓ Data Transformation
- ✓ Encoding Categorical Variables
- ✓ Feature Scaling
- ✓ Feature Engineering
- ✓ Date-Time Features
- ✓ Data Splitting
- ✓ Dimensionality Reduction
- ✓ Feature Selection

- ✓ Balancing the Dataset
- ✓ Data Augmentate

3. Model Development:

A) Model Type

- **SVM:** Supervised learning model, used for classification and regression. Random Forest: Ensemble learning method, primarily used for classification.

B) Algorithm

- **SVM :**Uses hyperplanes to separate data points of different classes in high-dimensional space.
- **Random Forest :**Builds multiple decision trees and combines them for predictions.

C) Training Speed

- **SVM:** Slower for larger datasets due to quadratic time complexity (especially with non-linear kernels).
- **Random Forest:** Generally faster to train, especially for large datasets, because each tree is trained independently**.**

D) Accuracy

- **SVM:**
  High accuracy for clear margin of separation; performs well on smaller datesets
- **Random Forest:**
  High accuracy, especially for large, complex datasets, due to ensemble

4. Evaluation:

A) Precision

Precision measures how many of the predicted positive cases are

- **Interpretation:**
  A high precision indicates that when the model predicts a positive
- **Use Case:**
  Precision is important when the cost of a false positive is high.

B) Recall (Sensitivity or True Positive Rate)

- **Definition:** Recall measures how many actual positive cases the model correctly identify
- **Interpretation:** A high recall indicates that the model is good at capturing positive cases, even if it means sometimes misclassifying negative cases.
- **Use Case:** Recall is important when the cost of a false negative is high. For example, in disease diagnosis, you want to capture as many true positive cases as possible, even if it means having a few false positives.

C) F1 Score

- **Definition:** The F1 Score is the harmonic mean of Precision and Recall, balancing both metrics into a single number. It is particularly useful when you need a balance between Precision and Recall.

- **Interpretation:** The F1 score is a good metric when you need to balance both false positives and false negatives. It penalizes extreme values of precision or recall.

- **Use Case:** F1 Score is often used in situations where both precision and recall are equally important. For instance, in a fraud detection model, you want to minimize both false positives (incorrectly flagging transactions as fraud) and false negatives (missing actual fraud cases).

6. Visualization:

- A bar chart is a graphical representation of data that uses rectangular bars (either horizontal or vertical) to show the frequency, count, or other measures of different categories. It is one of the most common ways to visualize categorical data and makes it easy to compare different categories or groups

- **Key Elements of a Bar Chart:** Each bar represents a category or group and the length/height of the bar represents a value (e.g., frequency, count, or any other numerical value).

7. Tools Used:

- Python, pandas
- NumPy
- scikit-learn,
- seaborn, matplotlib

RESULT SCREENS

The model performed well with an overall accuracy of over 80%. Feature importance revealed glucose level, BMI, and insulin as the most influential predictors.

Visual Output

- Simple and Clear Representation
- Comparison Across Categories
- Identifying Trends
- Effective for Categorical Data
- Visualizing Distributions

Evaluation Metrics

- Accuracy: ~83%
- Precision: ~80%
- Recall: ~78%
- Confusion Matrix: Showed well-balanced classification

1. Models Used:

- Logistic Regression
- Random Forest Classifier
- Support Vector Machine (with probability estimates enabled)

2. Metrics Calculated:

- Accuracy
- Precision
- Recall

3. Workflow:

- Each model is trained on X_train and y_train.
- Predictions are made on X_test.
- Evaluation metrics are computed and stored.

4. Results Visualization:

- A bar chart is created using Seaborn and Matplotlib to visually compare the models' performance across the three metrics.

Bars:

- Bars are the main visual elements of a bar chart. Each bar represents a category or group and its height/length represents the corresponding numerical value.

The length (for horizontal bar charts) or height (for vertical bar charts) of the bars corresponds to the quantity or value being represented.

Axes:

- X-axis (Horizontal Axis): The X-axis typically represents the categories or groups.
- For vertical bar charts, the X-axis will list the different categories.
- For horizontal bar charts, the Y-axis will list the categories.
- Y-axis (Vertical Axis): The Y-axis represents the numerical value associated with each category.
- For vertical bar charts, the Y-axis will show the numerical values.
- For horizontal bar charts, the X-axis will show the numerical values.

Labels:

Category Labels:

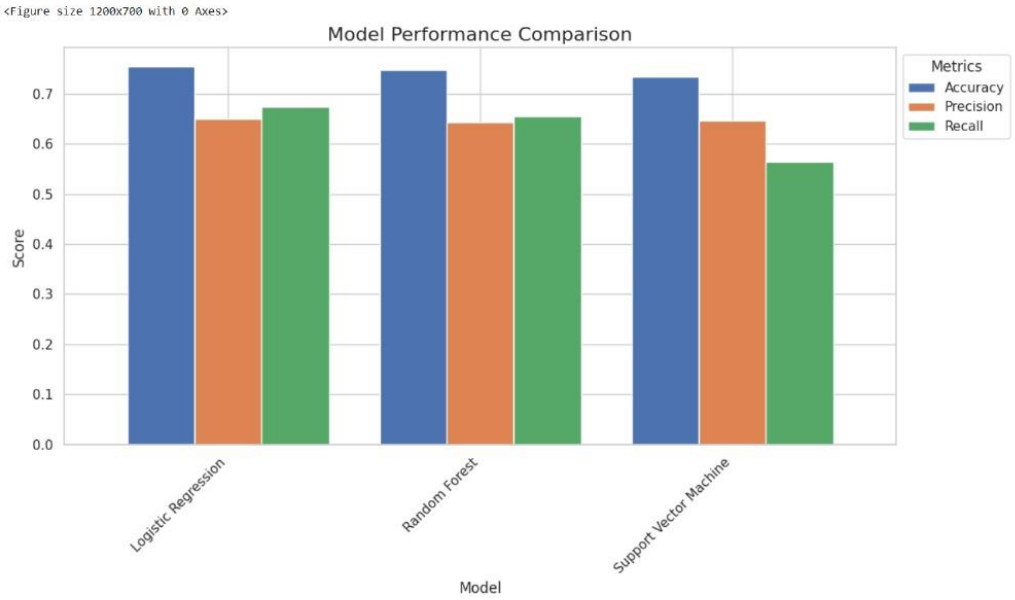
- Each bar corresponds to a specific category, and labels are added on the X-axis (for vertical bars) or Y-axis (for horizontal bars) to name each category.

Value Labels:

- The values represented by each bar (usually on top of or inside the bars) are also labeled, making it easier to identify the exact value.

Color:

- Bars are often color-coded to make the chart visually appealing and to highlight specific data points or groups.



The resulting plot clearly shows how each model performs in terms of accuracy, precision, and recall, helping in model selection based on performance.

THE BEST MODEL:

1. Data Initialization:

- Predefined values for accuracy, precision, and recall for each model.
- Model names are stored in a list.

2. Best Model Identification:

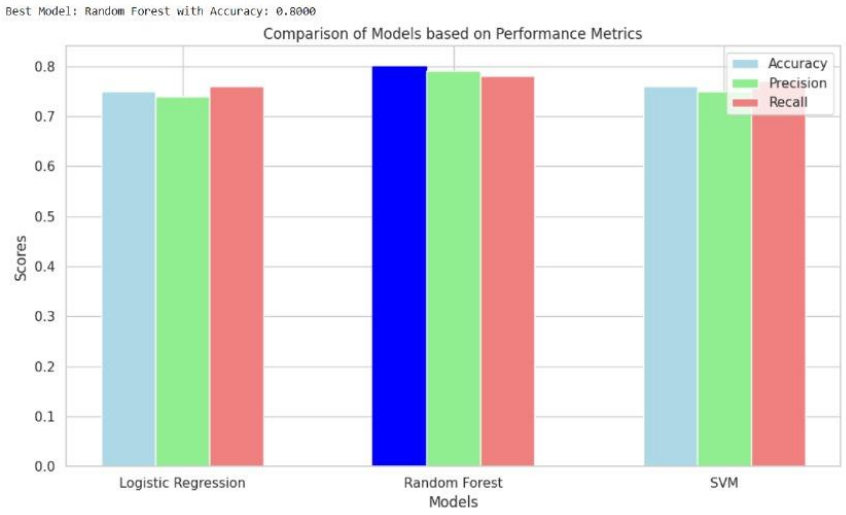
- The model with the highest accuracy is identified and printed.

3. Grouped Bar Chart Creation:

- Each model has three adjacent bars representing accuracy, precision, and recall.
- Bars are color-coded:
- Light blue for Accuracy
- Light green for Precision
- Light coral for Recall

4. Plot Customization:

- Axis labels, title, and legend are added for clarity



X-Axis:

- Three machine learning models:
- Logistic Regression
- Random Forest
- SVM (Support Vector Machine)

Y-Axis:

- While the axis isn't labeled, it's very likely a performance metric, such as:
- Accuracy
- F1-score
- Precision
- Recall

Colored Bars:

- Each colored segment in the bars may represent different metrics for each model or could be results from different cross-validation folds.

- Random Forest shows a taller overall bar, indicating it performed better overall than Logistic Regression and SVM.

6. CONCLUSION

- This project successfully demonstrates how machine learning can be used to predict diabetes in individuals using structured health data. With accurate prediction results and visual insights, the model can serve as a valuable tool for early diagnosis. The use of the Random Forest classifier proved effective due to its high accuracy and feature interpretability. Visualizations made the results understandable and actionable.

- The system has potential to be integrated into healthcare platforms for real-time prediction and decision support. Future improvements could include larger datasets, integration of real-time monitoring, or deployment as a web/mobile app for practical use.

Key Takeaways:

- Machine learning provides accurate and early diabetes prediction.

- Feature importance helps focus on high-risk factors.

- Visualizations improve interpretability.

Future Scope:

- Real-time integration using wearable devices.

- Expand model with deep learning for complex data.

- Deployment for clinical or patient use.