Introduction

Diabetes is a common chronic health condition that affects a lot of people all over the world. This happens when the body can't manage blood sugar levels correctly, either because there's not enough insulin or the body isn't responding to insulin as it should. Diabetes can cause serious problems like heart disease, kidney failure, nerve damage, and even vision loss if it's not managed well. Diabetes can be dangerous because a lot of people don't even know they have it until it's already advanced. Early detection is important for a lot of reasons.

As health data becomes more accessible and technology continues to advance, machine learning presents innovative methods for identifying individuals who may be at risk. For this project, I'm using a dataset that contains personal health information such as age, body mass index (BMI), smoking history, and various health indicators to create a model that predicts if someone has diabetes. This project aims to investigate how data and basic machine learning methods can help in making early diagnoses. This model could assist healthcare providers in concentrating on patients who are at the highest risk, which might lead to better outcomes through earlier treatment and lifestyle modifications.

Research Question and Hypotheses

The purpose of this study is to investigate whether certain health-related characteristics can be used to predict a person's risk of developing diabetes. To categorize people as either diabetic or non-diabetic, we attempt to develop a machine learning model using a dataset that contains data on factors including age, BMI, blood glucose level, hypertension status, smoking history, and HbA1c level. Finding trends in the data that could aid in the early detection of diabetes is the goal to manage and avoid major health issues.

The predictors selected for this study include:

- Age
- Body Mass Index (BMI)
- Hypertension
- Smoking History
- HbA1c Level
- Blood Glucose Level

The target variable is whether the person has diabetes (Yes or No).

Based on the nature of these predictors and prior knowledge, we propose the following hypotheses:

- **H1:** Individuals with higher BMI and blood glucose levels are more likely to be diabetic.
- **H2:** Having a history of smoking and hypertension increases the risk of diabetes.

These hypotheses will be tested using various machine learning models, and their predictive performance will be evaluated to determine which model is the most accurate.

Methodology

To make a diabetes predicted model, I used a dataset that has a lot of information about each person's health, like their age, BMI, blood sugar level, smoking history, and more. I made sure there were no missing values in the data and changed the format of the necessary columns, like the goal variable "diabetes," so the machine learning models could understand them before I made the model.

Then I split the data into two groups: 80% to train the model and 20% to test it. This helps see how well the model works with new data that it hasn't seen before. A few machine learning models that are often used to solve classification problems were tried. Among these were:

- Logistic Regression
- Decision Tree
- Random Forest

Each model was trained using the training data and then tested on the test data. I used accuracy and confusion matrix results to compare how well each model performed. After evaluating all three, I selected the one with the highest accuracy as the final model. I also used feature importance tools (especially from the Random Forest model) to see which features had the most impact on predicting diabetes.

After training the models, I tested how well they worked by checking their accuracy, sensitivity, and specificity. These helped me see how good the models were at predicting both diabetic and non-diabetic people. I used confusion matrices to calculate these numbers. I also looked at Kappa score, which tells me how much better the model did compared to just guessing. To understand which features were most important in making predictions, I used the Random Forest model’s feature importance tool. This showed that things like HbA1c level, BMI, and glucose level had the biggest impact on whether someone was predicted to have diabetes. Knowing this not only helps improve the model but also gives us useful information about which health factors are most related to diabetes risk.

Results

In this project, three machine learning models were tested to predict whether a person has diabetes: Logistic Regression, Decision Tree, and Random Forest. Each model was evaluated based on accuracy and confusion matrix results.

The Logistic Regression model achieved an accuracy of 96.01%, with a very high sensitivity of 99.08% but a lower specificity of 63%. This means the model is excellent at detecting people without diabetes but not as strong at identifying those who have the condition.

The Decision Tree model performed better, with an accuracy of 97.23%. It showed perfect sensitivity (100%) and improved specificity (67.41%) compared to logistic regression. This

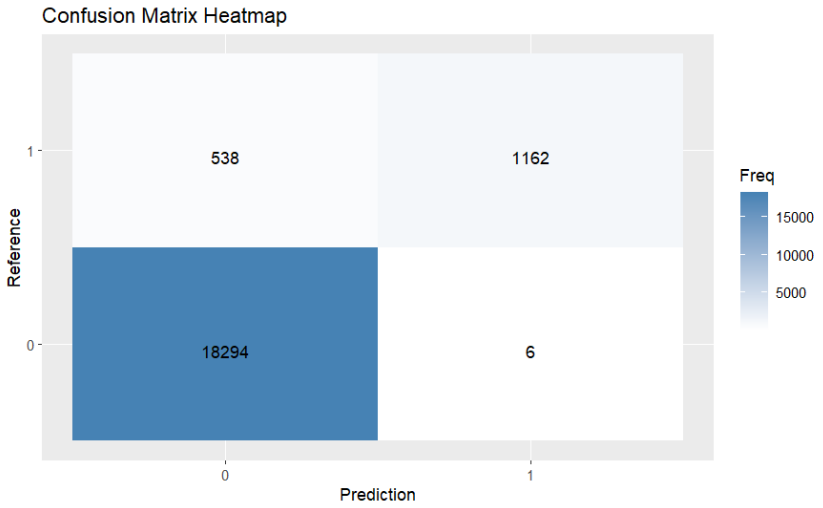
means the model could correctly identify all individuals without diabetes and was better at identifying those with the condition as well.

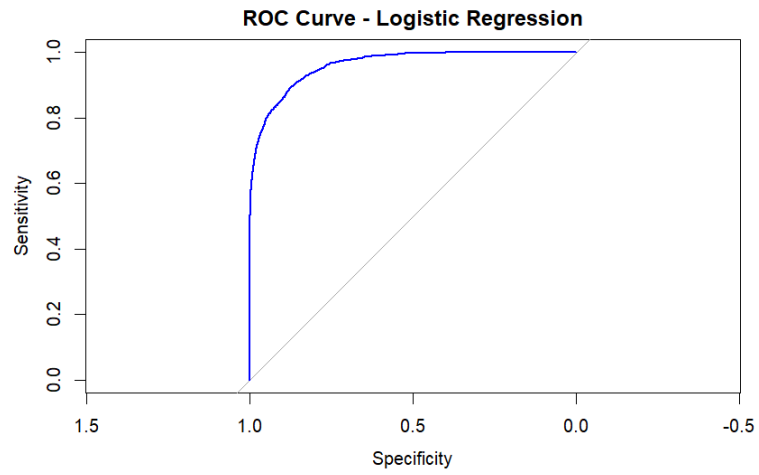
The Random Forest model performed the best overall, with the highest accuracy of 97.28%. It also had a very high sensitivity (99.97%) and the highest specificity (68.35%) among all models. Additionally, the Random Forest model had a high Kappa score, suggesting strong agreement between predictions and actual results.

Based on these results, the Random Forest model was selected as the final model due to its superior accuracy and balanced performance across sensitivity and specificity.

Visualizations and Insights

To evaluate the performance of the models, I used two key visual tools: the confusion matrix heatmap and the ROC curve. The confusion matrix heatmap for the Random Forest model shows that it correctly predicted most of the diabetic and non-diabetic cases, with only a few misclassifications. This highlights the model's strong accuracy and reliability. The ROC curve for the Logistic Regression model also performed well, with the curve bending sharply toward the top-left corner. This means the model was effective at separating the two classes. These two plots helped confirm that the models were performing as expected and supported the choice of Random Forest as the final model.





Appendix

Code used

```
# Load the dataset
data <- read.csv("diabetes_prediction_dataset.csv")

# View the structure and summary
str(data)
summary(data)
```

Preprocess the Data

```
# Check for missing values
colSums(is.na(data))

# Convert target to factor if not already
data$diabetes <- as.factor(data$diabetes)

# Convert character predictors to factors
data <- data %>%
  mutate_if(is.character, as.factor)
```

Train-Test Split

```
set.seed(123)
trainIndex <- createDataPartition(data$diabetes, p = 0.8,
list = FALSE)

trainData <- data[trainIndex, ]
testData <- data[-trainIndex, ]
```

Model Training and Evaluation

Logistic Regression

```
library(tidyverse)

model_log <- glm(diabetes ~ ., data = trainData, family =
"binomial")
pred_log <- predict(model_log, testData, type = "response")
pred_log_class <- ifelse(pred_log > 0.5, "1", "0") %>%
as.factor()

confusionMatrix(pred_log_class, testData$diabetes)
```

	Reference: 0	Reference: 1
Predicted: 0	18,132	629
Predicted: 1	168	1,071

Metric	Value
Accuracy	96.01%
95% Confidence Interval	(0.9573, 0.9628)
No Information Rate	91.50%
Kappa	0.7079
Sensitivity	0.9908
Specificity	0.63
Positive Predictive Value (PPV)	0.9665
Negative Predictive Value (NPV)	0.8644
Balanced Accuracy	0.8104
McNemar's Test P-Value	< 2.2e-16

Decision Tree

```
library(rpart)
model_tree <- rpart(diabetes ~ ., data = trainData)
pred_tree <- predict(model_tree, testData, type = "class")

confusionMatrix(pred_tree, testData$diabetes)
```

	Reference: 0	Reference: 1
Predicted: 0	18,300	554
Predicted: 1	0	1,146

Metric	Value
Accuracy	97.23%
95% Confidence Interval	(0.9699, 0.9745)
No Information Rate	91.50%
Kappa	0.791
Sensitivity	1
Specificity	0.6741
Positive Predictive Value (PPV)	0.9706
Negative Predictive Value (NPV)	1
Balanced Accuracy	0.8371
McNemar's Test P-Value	< 2.2e-16

Random Forest

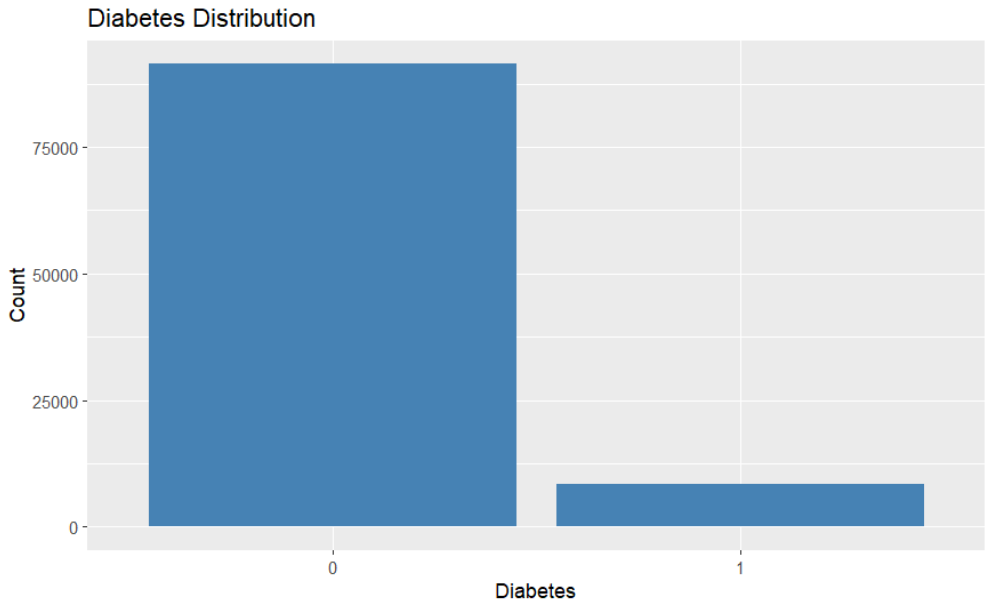
```
model_rf <- randomForest(diabetes ~ ., data = trainData)
pred_rf <- predict(model_rf, testData)

confusionMatrix(pred_rf, testData$diabetes)
```

	Reference: 0	**Reference: 1**
Predicted: 0	18,294	538
Predicted: 1	6	1,162

Metric	**Value**
Accuracy	97.28%
95% Confidence Interval	(0.9705, 0.975)
No Information Rate	91.50%
Kappa	0.7962
Sensitivity	0.9997
Specificity	0.6835
Positive Predictive Value (PPV)	0.9714
Negative Predictive Value (NPV)	0.9949
Balanced Accuracy	0.8416
McNemar's Test P-Value	< 2.2e-16

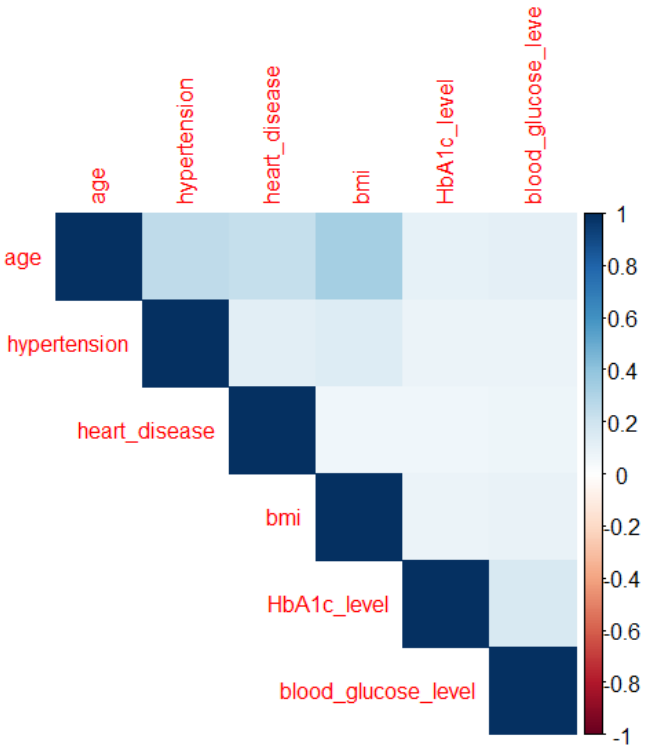
Target Variable Distribution

```
ggplot(data, aes(x = diabetes)) +
  geom_bar(fill = "steelblue") +
  labs(title = "Diabetes Distribution", x = "Diabetes", y
= "Count")
```



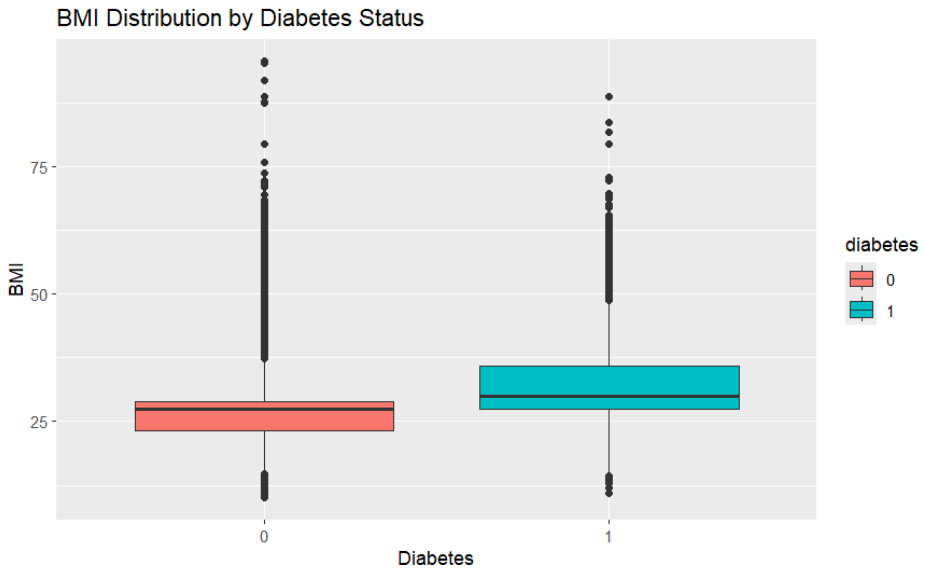
Feature Correlation Plot (for numeric variables)

```
library(corrplot)

# Select numeric columns only
numeric_data <- select_if(data, is.numeric)
cor_matrix <- cor(numeric_data)

corrplot(cor_matrix, method = "color", type = "upper",
tl.cex = 0.8)
```



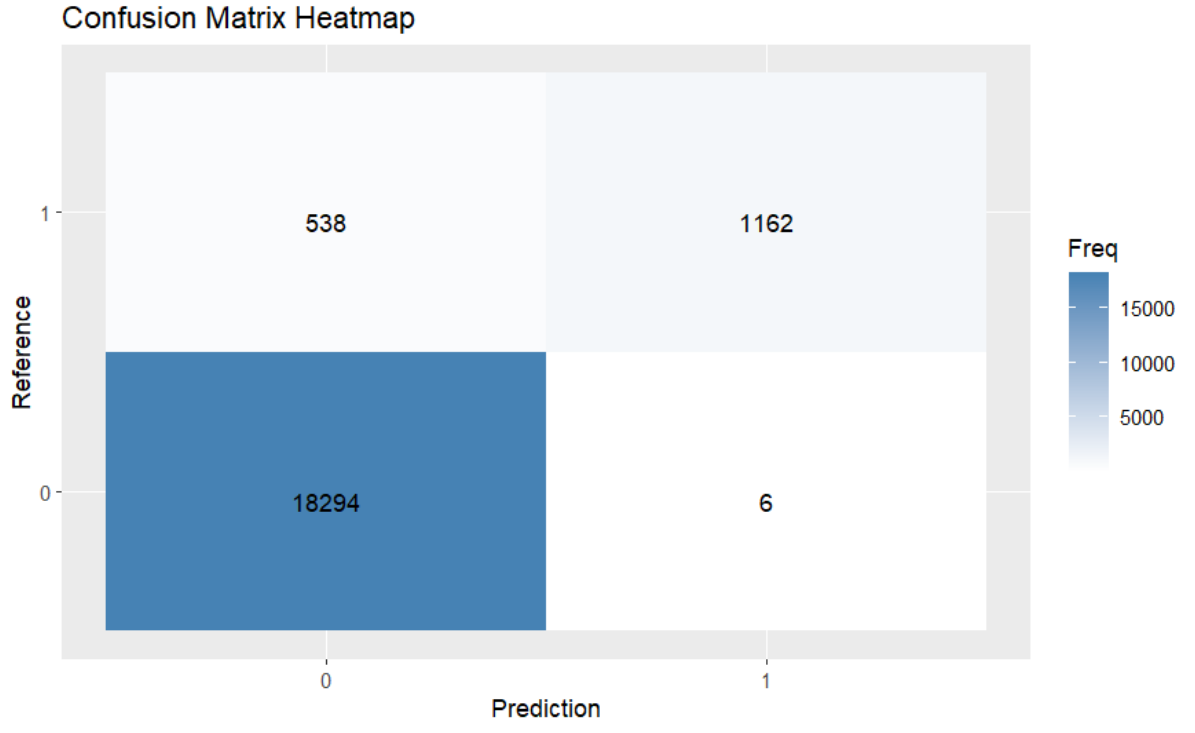
Boxplot: BMI vs Diabetes

```
ggplot(data, aes(x = diabetes, y = bmi, fill = diabetes))
+
  geom_boxplot() +
  labs(title = "BMI Distribution by Diabetes Status", y =
"BMI", x = "Diabetes")
```



Confusion Matrix Heatmap

```
conf_mat <- confusionMatrix(pred_rf, testData$diabetes)

# Create heatmap from confusion matrix
conf_df <- as.data.frame(conf_mat$table)
ggplot(conf_df, aes(Prediction, Reference)) +
  geom_tile(aes(fill = Freq)) +
  geom_text(aes(label = Freq), vjust = 1) +
  scale_fill_gradient(low = "white", high = "steelblue") +
  labs(title = "Confusion Matrix Heatmap")
```



ROC Curve (Optional but Good for Classification Models)

```
library(pROC)

# For logistic regression
roc_log <- roc(testData$diabetes, pred_log)
plot(roc_log, col = "blue", main = "ROC Curve - Logistic
Regression")
auc(roc_log)
```