```yaml
---
title: "A Machine Learning Framework for Early Detection of Cardiovascular
Risk Using Diabetes-Related Indicators: A Predictive Modeling Data-Driven Research"
author: "Julian Borges"
date: "2024-08-23"
output:
  pdf_document: default
  html_document: default
---
```

```r
# Load necessary libraries
library(tidyverse)
library(caret)
library(randomForest)
library(pROC)
library(corrplot)

# URLs for the datasets
diabetes_url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/00529/diabetes_data_upload.cs
heart_failure_url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/00519/heart_failure_cli

# Local file paths for saving
diabetes_csv <- "diabetes_data_upload.csv"
heart_failure_csv <- "heart_failure_clinical_records_dataset.csv"

# Download the datasets
download.file(diabetes_url, diabetes_csv, mode = "wb")
download.file(heart_failure_url, heart_failure_csv, mode = "wb")

# Load the datasets
diabetes_data <- read_csv(diabetes_csv)
heart_failure_data <- read_csv(heart_failure_csv)

# Data Cleaning: Handle missing values and convert categorical variables
diabetes_data <- diabetes_data %>%
  mutate(across(where(is.numeric), ~ifelse(is.na(.), median(., na.rm = TRUE), .))) %>%
  mutate(Gender = ifelse(Gender == "Male", 1, 0))

# Rename columns with spaces in diabetes_data
colnames(diabetes_data) <- make.names(colnames(diabetes_data))

# Convert "Yes"/"No" to 1/0 for relevant columns
binary_columns <- c("Polyuria", "Polydipsia", "sudden.weight.loss", "weakness",
                    "Polyphagia", "Genital.thrush", "visual.blurring", "Itching",
                    "Irritability", "delayed.healing", "partial.paresis",
                    "muscle.stiffness", "Alopecia", "Obesity", "class")

diabetes_data[binary_columns] <- lapply(diabetes_data[binary_columns], function(x) ifelse(x == "Yes", 1

# Convert the class column to a factor
diabetes_data$class <- as.factor(diabetes_data$class)

# Convert the target column in the heart failure dataset to binary
heart_failure_data$DEATH_EVENT <- as.factor(heart_failure_data$DEATH_EVENT)
```

```r
# Summary of the datasets
summary(diabetes_data)
summary(heart_failure_data)
```

Exploratory Data Analysis (EDA)

```r
# Age Distribution in Diabetes Dataset
ggplot(diabetes_data, aes(x = Age)) +
  geom_histogram(binwidth = 5, fill = "blue", color = "black") +
  labs(title = "Age Distribution in Diabetes Dataset", x = "Age", y = "Count")

# Age Distribution in Heart Failure Dataset
ggplot(heart_failure_data, aes(x = age)) +
  geom_histogram(binwidth = 5, fill = "red", color = "black") +
  labs(title = "Age Distribution in Heart Failure Dataset", x = "Age", y = "Count")
```

Correlation Analysis

```r
# Correlation analysis on the diabetes dataset
correlation_matrix_diabetes <- cor(diabetes_data %>% select(where(is.numeric)), use = "complete.obs")
corrplot(correlation_matrix_diabetes, method = "circle", title = "Correlation Matrix - Diabetes Dataset

# Correlation analysis on the heart failure dataset
correlation_matrix_heart_failure <- cor(heart_failure_data %>% select(where(is.numeric)), use = "comple
corrplot(correlation_matrix_heart_failure, method = "circle", title = "Correlation Matrix - Heart Failu
```

Feature Engineering

```r
# Create a symptom severity score in the diabetes dataset
diabetes_data <- diabetes_data %>%
  mutate(symptom_severity = Polyuria + Polydipsia + `sudden.weight.loss` + weakness + Polyphagia)

# Normalize features if necessary
diabetes_data <- diabetes_data %>%
  mutate(across(where(is.numeric), scale))

heart_failure_data <- heart_failure_data %>%
  mutate(across(where(is.numeric), scale))

# Review the newly engineered features
head(diabetes_data)
head(heart_failure_data)
```

Model Development and Evaluation

Data Splitting

```r
# Split the diabetes data into training and test sets
set.seed(123)
diabetes_train_index <- createDataPartition(diabetes_data$class, p = 0.8, list = FALSE)
diabetes_train_data <- diabetes_data[diabetes_train_index, ]
diabetes_test_data <- diabetes_data[-diabetes_train_index, ]

# Split the heart failure data into training and test sets
```

```
set.seed(123)
train_index <- createDataPartition(heart_failure_data$DEATH_EVENT, p = 0.8, list = FALSE)
heart_failure_train <- heart_failure_data[train_index, ]
heart_failure_test <- heart_failure_data[-train_index, ]
```

Logistic Regression on Heart Failure Data

```
# Logistic Regression on Heart Failure Data
logit_model <- glm(DEATH_EVENT ~ ., data = heart_failure_train, family = binomial)
logit_predictions <- predict(logit_model, newdata = heart_failure_test, type = "response")
logit_predicted_classes <- ifelse(logit_predictions > 0.5, 1, 0)
logit_predicted_factors <- as.factor(logit_predicted_classes)
logit_conf_matrix <- confusionMatrix(logit_predicted_factors, heart_failure_test$DEATH_EVENT)
print(logit_conf_matrix)
```

Random Forest on Heart Failure Data

```
# Random Forest on Heart Failure Data
rf_model <- randomForest(DEATH_EVENT ~ ., data = heart_failure_train, importance = TRUE, ntree = 500)
rf_predictions <- predict(rf_model, newdata = heart_failure_test)
rf_conf_matrix <- confusionMatrix(rf_predictions, heart_failure_test$DEATH_EVENT)
print(rf_conf_matrix)
```

Feature Importance

```
# Feature Importance from the Random Forest model
importance <- importance(rf_model)
print(importance)
varImpPlot(rf_model)
```

Model Evaluation

```
# ROC Curve and AUC for Random Forest model
rf_probs <- predict(rf_model, newdata = heart_failure_test, type = "prob")[,2]
roc_curve <- roc(heart_failure_test$DEATH_EVENT, rf_probs)
plot(roc_curve, col = "blue", main = "ROC Curve - Random Forest")
auc_value <- auc(roc_curve)
print(paste("AUC - Random Forest:", auc_value))
```

Conclusion

In this project, I explored the potential link between early-stage diabetes symptoms and cardiovascular disease (CVD) risk by analyzing two distinct datasets. Through the use of logistic regression and Random Forest models, I demonstrated that it is possible to predict the risk of heart failure using symptoms commonly associated with diabetes. The Random Forest model, in particular, showed promising results, with a strong AUC, indicating good model performance.

Future Work

Further analysis could explore the relationship between specific diabetes symptoms and other cardiovascular conditions, such as coronary artery disease, to build more generalized predictive models. Additionally, using larger datasets and more advanced machine learning techniques may yield even more accurate predictions.

References

1. Harrell, Frank E. “Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis.” Springer, 2015. https://doi.org/10.1007/978-3-319-19425-7.

2. Breiman, Leo. “Random Forests.” Machine Learning 45, no. 1 (2001): 5-32. https://doi.org/10.1023/A:1010933404324.

3. Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. “The Elements of Statistical Learning: Data Mining, Inference, and Prediction.” Springer Science & Business Media, 2009. https://doi.org/10.1007/978-0-387-84858-7. “‘