Curso de Especialização em *Data Science* e Estatística Aplicada
Módulo III - *Big Data Analytics*
Atividade Avaliativa

Profa. Dra. Juliana Scudilio

04/12/2024

Instruções

- O desenvolvimento desta atividade deve ser realizada de forma individual ou em dupla.
- Deve-se completar o arquivo Rmd enviado na atividade.
- É necessário devolver o arquivo em Rmd e em pdf.
- Valor da atividade: 10 pontos.
- Use o código em anexo como base, mantenha a mesma semente do código (set.seed(42)).
- A atividade deve conter todas as etapas abaixo:

1. Definir do problema (descrito abaixo).
2. Realizar a Análise Exploratória dos Dados (EDA).
3. Realizar o pré-processamento dos dados.
4. Construir e treinar o modelo.
5. Avaliar o modelo.
6. Elaborar a conclusão.

Atividade

Construir um modelo de classificação utilizando a Árvore de Decisão usando a linguagem R para prever a ocorrência de diabetes com base no dataset diabetes.csv. A atividade deverá ser entregue em formato de relatório no R Markdown e PDF, htlm ou Word, contendo todas as etapas do processo de modelagem. Os dados foram extraídos do seguinte link: https://www.kaggle.com/datasets/mathchi/diabetes-data-set

Etapas do projeto

1. Definir do Problema

O objetivo principal do problema é prever a ocorrência de diabetes (coluna Outcome) utilizando um modelo baseado em Árvore de Decisão.

2. Análise Exploratória de Dados (EDA)

Realize a Análise Exploratória de Dados (EDA). Apresente os seguintes itens:

(i) Número de linhas e colunas do dataset;
(ii) Tipos de variáveis do dataset;
(iii) Presença e número de valores ausentes;
(iv) Estatísticas descritivas (média, mediana, variância, frequências, etc.);
(v) Geração de gráficos para entender a relação entre as variáveis independentes e a variável-alvo (se aplicável).

3. Realizar o pré-processamento dos dados

Realize as seguintes etapas para o pré-processamento dos dados:

(i) Tratamento de valores ausentes: identificar e substituir ou remover valores nulos.
(ii) Escalonamento: se necessário, normalize variáveis com escalas muito distintas.
(iii) Divisão dos dados:

- Separe o dataset em conjunto de treino (70%) e teste (30%).
- Use uma semente aleatória para garantir a reprodutibilidade.

OBS: use o código abaixo com a mesma semente (set.seed(42))

4. Constuir e treinar o modelo

Realize a construção do modelo seguindo os passos abaixo:

(i) Utilize a biblioteca rpart para criar a Árvore de Decisão.
(ii) Detalhe os parâmetros escolhidos para o treinamento.
(iii) Geração do gráfico da Árvore.

5. Avaliar o Modelo

Realize a avaliação do modelo seguindo os passos abaixo:

(i) Calcule as métricas de desempenho: Acurácia; Matriz de confusão; Precisão, Recall e F1-Score.
(ii) Compare o desempenho nos conjuntos de treino e teste.

6. Elaborar a conclusão

Apresente os principais insights do modelo:

(i) A importância das variáveis no modelo.
(ii) Interpretação dos resultados obtidos.
(iii) Limitações do modelo e sugestões de melhorias.

Apêndice

Use o código a seguir como base e faça ajustes se necessário:

```r
# Carregando os pacotes necessários
library(dplyr)        # Manipulação de dados
library(ggplot2)      # Visualização de dados
library(rpart)        # Criação do modelo de Árvore de Decisão
library(rpart.plot)   # Plot da Árvore de Decisão
library(caret)        # Avaliação do modelo

# Carregando o dataset
diabetes <- read.csv("diabetes.csv")

# Inspecionando os dados
summary(diabetes)  # Estatísticas descritivas das variáveis
str(diabetes)      # Estrutura do dataset

# Análise Exploratória: Distribuição da variável dependente (Outcome)
ggplot(diabetes, aes(x = as.factor(Outcome))) +
  geom_bar(fill = "skyblue") +
  labs(x = "Outcome", y = "Frequência", title = "Distribuição de Casos de Diabetes") +
  theme_minimal()

# Tratamento de valores ausentes (substituindo zeros por médias nas variáveis contínuas)
diabetes_clean <- diabetes %>%
  mutate(across(c(Glucose, BloodPressure, SkinThickness, Insulin, BMI), ~ ifelse(. == 0, mean(., na.rm =

# Confirmando o tratamento
summary(diabetes_clean)

# Divisão do dataset em treino (70%) e teste (30%)
set.seed(42)  # Garantindo reprodutibilidade
index <- sample(1:nrow(diabetes_clean), 0.7 * nrow(diabetes_clean))
train_data <- diabetes_clean[index, ]
test_data <- diabetes_clean[-index, ]

# Criação do modelo de Árvore de Decisão
model <- rpart(Outcome ~ ., data = train_data, method = "class")

# Visualizando a Árvore
rpart.plot(model, type = 3, extra = 106, under = TRUE, tweak = 1.2)

# Realizando previsões no conjunto de teste
predictions <- predict(model, test_data, type = "class")

# Avaliação do modelo com matriz de confusão
confusion <- confusionMatrix(as.factor(predictions), as.factor(test_data$Outcome))
print(confusion)

# Métricas de desempenho
accuracy <- confusion$overall["Accuracy"]
precision <- confusion$byClass["Pos Pred Value"]
recall <- confusion$byClass["Sensitivity"]
f1_score <- 2 * ((precision * recall) / (precision + recall))

cat("Acurácia:", accuracy, "\n")
```

```
cat("Precisão:", precision, "\n")
cat("Recall:", recall, "\n")
cat("F1-Score:", f1_score, "\n")
```