Carrera de Computación
Fundamentos de Análisis de Datos
Examen Bimestral B2 – Práctica

Respuestas

PREGUNTA 1: CLASIFICACIÓN (NIVEL DE OBESIDAD)

Datos resumen

Indique los datos generales sobre el modelo generado y sus principales indicadores

TÉCNICA APLICADA: Random Forest
VARIABLES INDEPENDIENTES SELECCIONADAS: Age, FCVC, TUE, FAF, NCP, CH2O, Gender
ACCURACY: 0.8278
SENSITIVITY (Promedio): 0.7544
SPECIFICITY (Promedio): 0.9640

Script R

En el cuadro verde pegue el texto de los comandos R usados para lo solicitado en la Pregunta 1

```
library(dplyr)
df = read.csv("Obesidad_Data.csv")

str(df)
names(df)
head(df)

unique(df$SCC)
# Clasificacion de variables
# Categoricas
# - Gender - family_with_overweight - FAVC - CAEC - SMOKE - SCC - CALC - MTRANS -
NObeyesdad
# Numericas
# - Age - Height - Weight - FCVC - NCP - CH2O - FAF - TUE
# Bimodales
# -

# Media de valores faltantes y nulos
sapply(df, function(x) sum(is.na(x) | !nzchar(x)))

# Limpieza
df = df %>% filter(nzchar(family_with_overweight))
df = df %>% filter(!is.na(FCVC))

# Verficamos
sapply(df, function(x) sum(is.na(x) | !nzchar(x)))
```

```
##############################
# CLASIFICACION
# Predecir: nivel de obesidad
##############################

# Random Forest

# Librerias Nescesarias
library(dplyr)
library(caret)
library(randomForest)

unique(df$NObeyesdad)
# Seleccionamos las variables
df_analisis1 = df %>% select(NObeyesdad, FCVC, NCP, Age, TUE, FAF, CH2O, Gender)

# Convertimos a factores
df_analisis1$NObeyesdad = as.factor(df_analisis1$NObeyesdad)
df_analisis1$Gender = as.factor(df_analisis1$Gender)

# Dividir los datos de entrenamiento (80%) y de Prueba (20%)
set.seed (88)
train_index = createDataPartition(df_analisis1$NObeyesdad, p = 0.8, list = FALSE)
train_data = df_analisis1[train_index, ]

#Datos de entrenamiento
test_data <- df_analisis1[-train_index, ]

#Datos de prueba
# Ajustar el modelo Random Forest
set.seed (88)

rf_model = randomForest(NObeyesdad ~ Age + FCVC + TUE + FAF + NCP + CH2O + Gender,
                        data = train_data,
                        ntree = 500,
                        mtry = 2)

print(rf_model)

# Importancia de las variables
var_imp = importance(rf_model)
print(var_imp)

# Visualizar la importancia de las variables
varImpPlot(rf_model)

# EVALUAR EL MODELO
# Predicciones en el conjunto de prueba
rf_pred = predict(rf_model, test_data)

# Matriz de confusión
conf_matrix <- confusionMatrix(rf_pred,
                               test_data$NObeyesdad)
print(conf_matrix)
```

Matriz de confusión y otras salidas del modelo final generado

Pegue aquí capturas de la matriz de confusión y otras salidas de acuerdo a la técnica usada (resumen, arbol, importancia de variables, etc.)

- ## Modelo Random Forest

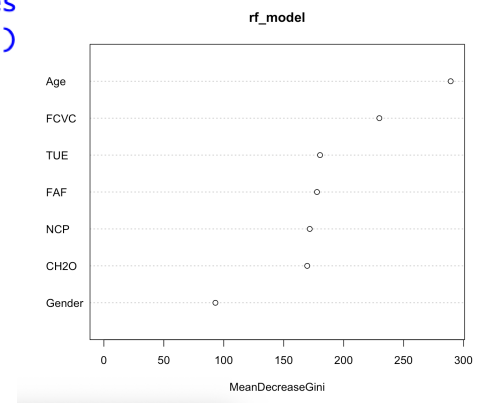
```
> rf_model = randomForest(NObeyesdad ~ Age + FCVC + TUE + FAF + NCP + CH2O + Gender,
+                         data = train_data,
+                         ntree = 500,
+                         mtry = 2)
> print(rf_model)

Call:
 randomForest(formula = NObeyesdad ~ Age + FCVC + TUE + FAF +      NCP + CH2O + Gender, data = train_data,
ntree = 500, mtry = 2)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 19.94%
```

- ## Importancia de variables

```
> # Importancia de las variables
> var_imp = importance(rf_model)
> print(var_imp)
       MeanDecreaseGini
Age           289.30405
FCVC          229.78901
TUE           180.33834
FAF           177.76878
NCP           171.74507
CH2O          169.62238
Gender         93.04247
```



- ## Matriz de confusion

```
> print(conf_matrix)
Confusion Matrix and Statistics

                     Reference
Prediction            Insufficient_Weight Normal_Weight Obesity_Type_I Obesity_Type_II Obesity_Type_III
  Insufficient_Weight                  46             1              0               0                0
  Normal_Weight                         2            48              9               4                1
  Obesity_Type_I                        2             1             54               0                0
  Obesity_Type_II                       0             0              2              53                0
  Obesity_Type_III                      0             0              0               0               63
  Overweight_Level_I                    2             4              3               1                0
  Overweight_Level_II                   2             3              2               1                0
                     Reference
Prediction            Overweight_Level_I Overweight_Level_II
  Insufficient_Weight                  1                   3
  Normal_Weight                        7                   5
  Obesity_Type_I                       4                   3
  Obesity_Type_II                      1                   2
  Obesity_Type_III                     0                   0
  Overweight_Level_I                  39                   1
  Overweight_Level_II                  5                  43

Overall Statistics

               Accuracy : 0.8278
                 95% CI : (0.7881, 0.8627)
    No Information Rate : 0.1675
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7989
```

Interprete los resultados

¿Qué análisis puede realizar respecto a los resultados del modelo final obtenido, y del proceso que conllevo llegar a ese modelo final?

El modelo Random Forest logró una precisión del 82.78% lo que indica un buen desempeño en la predicción del nivel de obesidad.
Las variables más influyentes fueron la

- Edad
- Consumo de alimentos saludables (FCVC)
- Tiempo frente a pantallas (TUE)

El modelo mostró buena sensibilidad (75.44%) y alta especificidad (96.40%), lo que significa que identifica correctamente tanto los casos positivos como negativos. El proceso incluyó limpieza de datos, selección de variables relevantes y ajuste del modelo, logrando resultados confiables y útiles para aplicaciones reales.

PREGUNTA 2: REGRESIÓN (IMC)

Datos resumen

Indique los datos generales sobre el modelo generado y sus principales indicadores

TÉCNICA APLICADA: Random Forest (Regresion)
VARIABLES INDEPENDIENTES SELECCIONADAS: `Age, FCVC, TUE, CAEC, FAF, NCP, CALC, Gender`
Mean Absolute Error (MAE): 2.11
Mean Squared Error (MSE): 8.22
Root Mean Squared Error (RMSE): 2.87

Script R

En el cuadro verde pegue el texto de los comandos R usados para lo solicitado en la Pregunta 1

```
########################################
# REGRESION
# Predecir: indice de masa corporal(IMC)
########################################

# Random Forest
library(dplyr)
library(caret)
library(randomForest)

# Calculamos el IMC
df_analisis2 = df %>%
  mutate(IMC = Weight / (Height^2)
)

names(df_analisis2)
unique(df_analisis2$CALC)
```

```r
# Seleccionamos las variables
df_analisis2 = df_analisis2 %>%
  select(IMC, Age, FCVC, TUE, CAEC, FAF, NCP, CALC, Gender)

# Convertimos a factores
df_analisis2$CAEC = as.factor(df_analisis2$CAEC)

df_analisis2$CALC = as.factor(df_analisis2$CALC)
df_analisis2$Gender = as.factor(df_analisis2$Gender)

# Dividir los datos de entrenamiento (80%) y de Prueba (20%)
set.seed (88)
train_index = createDataPartition(df_analisis2$IMC,
                                  p = 0.8,
                                  list = FALSE)

train_data = df_analisis2[train_index, ]

#Datos de entrenamiento
test_data <- df_analisis2[-train_index, ]

#Datos de prueba
# Ajustar el modelo Random Forest
set.seed (88)

rf_model = randomForest(IMC ~ Age + FCVC + TUE + CAEC + FAF + NCP + CALC + Gender,
                        data = train_data,
                        ntree = 300,
                        mtry = 3)

print(rf_model)

# Importancia de las variables
var_imp = importance(rf_model)
print(var_imp)

# Visualizar la importancia de las variables
varImpPlot(rf_model)

# EVALUAR EL MODELO
# Predicciones en el conjunto de prueba
rf_pred = predict(rf_model, test_data)

# Error medio absoluto (MAE)
mae = mean(abs(rf_pred - test_data$IMC))
print(paste("Mean Absolute Error (MAE):", round(mae, 2)))

# Error cuadrático medio (MSE)
mse = mean((rf_pred - test_data$IMC)^2)
print(paste("Mean Squared Error (MSE):", round(mse, 2)))

# Raíz del error cuadrático medio (RMSE)
rmse = sqrt(mse)
print(paste("Root Mean Squared Error (RMSE):", round(rmse, 2)))
```

MAE, MSE, RMSE y otras salidas del modelo final generado

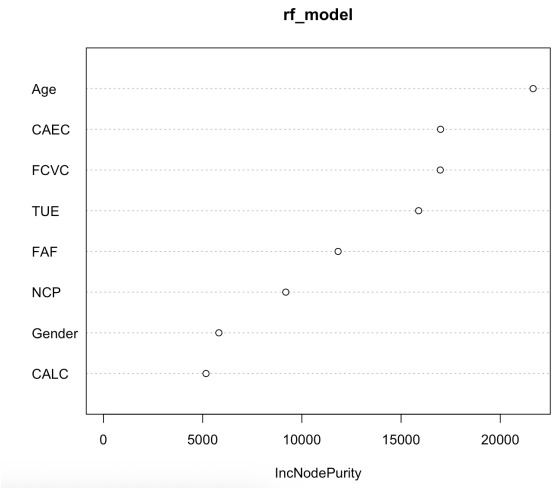
Pegue aquí capturas de los indicadores MAE, MSE, y RMSE, y otras salidas de acuerdo a la técnica usada (resumen, arbol, importancia de variables, etc.)

- **MAE, MSE, RMSE**

```
> # Visualizar la importancia de las variables
> varImpPlot(rf_model)
> # EVALUAR EL MODELO
> # Predicciones en el conjunto de prueba
> rf_pred = predict(rf_model, test_data)
> # Error medio absoluto (MAE)
> mae = mean(abs(rf_pred - test_data$IMC))
> print(paste("Mean Absolute Error (MAE):", round(mae, 2)))
[1] "Mean Absolute Error (MAE): 2.11"
> # Error cuadrático medio (MSE)
> mse = mean((rf_pred - test_data$IMC)^2)
> print(paste("Mean Squared Error (MSE):", round(mse, 2)))
[1] "Mean Squared Error (MSE): 8.22"
> # Raíz del error cuadrático medio (RMSE)
> rmse = sqrt(mse)
> print(paste("Root Mean Squared Error (RMSE):", round(rmse, 2)))
[1] "Root Mean Squared Error (RMSE): 2.87"
```

- **Importancia de variables**

```
> # Importancia de las variables
> var_imp = importance(rf_model)
> print(var_imp)
       IncNodePurity
Age         21655.940
FCVC        16975.548
TUE         15884.920
CAEC        16989.058
FAF         11825.398
NCP          9191.248
CALC         5168.824
Gender       5816.862
```



Interprete los resultados

¿Qué análisis puede realizar respecto a los resultados del modelo final obtenido, y del proceso que conllevo llegar a ese modelo final?

El modelo logró predecir el IMC con una precisión alta, explicando el 82.99% de la variabilidad de los datos. El RMSE fue de 2.87, lo que indica un buen ajuste y predicciones cercanas a los valores reales (se logró cumplir con el objetivo de mantener el RMSE por debajo de 3).
Las 3 variables más influyentes fueron la

- Edad
- Consumo de alimentos saludables (FCVC)

- Tiempo frente a pantallas (TUE)

Durante el proceso se realizó:

- Limpieza y selección de variables.
- Conversión de variables categóricas a factores.
- División en conjuntos de entrenamiento y prueba (80/20).
- Entrenamiento con 300 árboles y selección de 3 variables por división (mtry = 3).

Este proceso permitió obtener un modelo robusto, útil para la predicción del IMC en función de hábitos y características personales.

Disculpara ingeniero no hubo como con menos variables.

.-