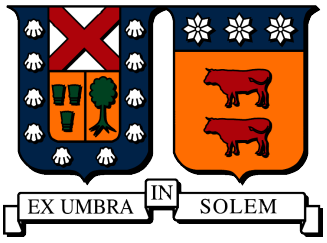

UNIVERSIDAD TÉCNICA
FEDERICO SANTA MARÍA

Departamento de Matemática

MAT281

Segundo Semestre 2023

Tarea 1

Aplicaciones de las matemáticas a la ingeniería

Nombre: Diego Alejandro Astaburuaga Corveleyn

Rol: 202010018-7

Valparaíso, Septiembre de 2023

1. Preliminares

A continuación se presenta la respuesta y desarrollo a la Tarea 1 del ramo aplicaciones de las matemáticas a la ingeniería. Las imágenes y códigos mostrados son de propia autoría con fuerte apoyo de herramientas computacionales disponibles en internet y del material de apoyo entregado por el profesor.

Se pueden encontrar todos los códigos utilizados en mi repositorio de GitHub (<– cuyo enlace está aquí). Aquí sólo se mostrarán algunos referenciales.

2. Tarea 1

Considere el conjunto de datos 'column_2C' disponible en AULA que contiene valores para seis características biomecánicas utilizadas para clasificar a los pacientes ortopédicos en 2 clases (normal o anormal). Más precisamente se tienen las siguientes covariables:

- X_1: incidencia pélvica.
- X_2: inclinación pélvica.
- X_3: ángulo de lordosis lumbar.
- X_4: pendiente sacra.
- X_5: radio pélvico.
- X_6: grado de espondilolistesis.

donde la última columna corresponde a la variable de respuesta (nombrada como `status`) que corresponde a la etiqueta `NO` y `AB`, que indica si el paciente es normal o anormal respectivamente.

a. ¿Cuántas observaciones hay en el conjunto de datos? ¿Cuántas observaciones corresponden a cada etiqueta?

Respuesta: Utilizando el código

```
df["status"].value_counts()
```

Listing 1: Variables por etiqueta.

donde `df` es la variable con el dataframe relacionado a los datos y `status` es el nombre puesto a la columna de las etiquetas.

Para la etiqueta `AB` hay 210 datos mientras que para `NO` hay 100. En total hay 310 observaciones. Se observa que las clases no están balanceadas:

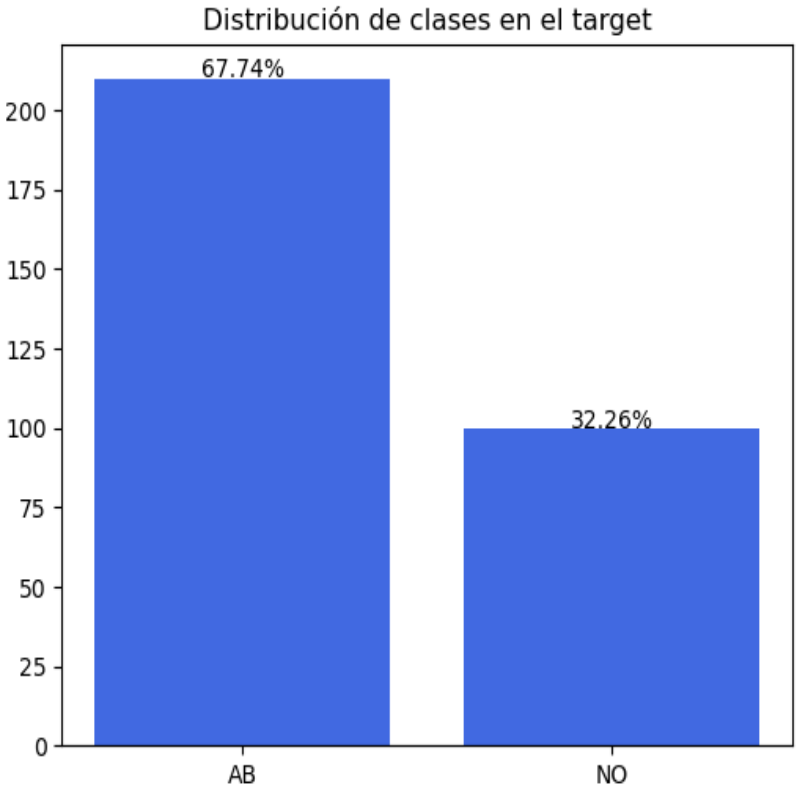

Figura 1: Distribución del target.

b. Realice histogramas para cada una de las covariables (separando las observaciones de acuerdo a las distintas etiquetas). Además, reporte las medidas de tendencia central, dispersión y forma. Comente los resultados.

Respuesta: Utilizando el siguiente código se obtienen las medidas de tendencia central, dispersión y forma para ambas etiquetas.

```
    for status in pd.unique(df['status']):
    print(f'Medidas de tendencia central,
        dispersion y forma para: {status}')
    temp = df[df['status']==status].drop(['status'], axis=1)
    info = temp.describe().T
    info['skew'] = temp.skew()
    info['curtosis'] = temp.kurt()
    display(info)
```

Listing 2: Medidas de tendencia.

Medidas de tendencia central, dispersión y forma para: AB

	count	mean	std	min	25%	50%	75%	max	skew	curtosis
incidencia pélvica	210.0	64.692143	17.661807	26.15	50.1050	65.275	77.5975	129.83	0.274841	0.202589
inclinación pélvica	210.0	19.791048	10.515653	-6.55	13.0475	18.795	24.8125	49.43	0.543336	0.165615
ángulo de lordosis lumbar	210.0	55.925190	19.668972	14.00	41.1175	56.150	68.1050	125.74	0.314309	-0.173225
pendiente sacra	210.0	44.901524	14.515133	13.37	34.3800	44.640	55.1425	121.43	0.650361	2.754752
radio pélvico	210.0	115.077381	14.090965	70.08	107.3075	115.650	123.1350	163.07	0.111110	1.034906
grado de espondilolistesis	210.0	37.777571	40.696738	-10.68	7.2600	31.945	55.3750	418.54	4.274374	35.952586

Medidas de tendencia central, dispersión y forma para: NO

	count	mean	std	min	25%	50%	75%	max	skew	curtosis
incidencia pélvica	100.0	51.6856	12.367900	30.74	42.8200	50.125	61.4725	89.83	0.747352	0.495784
inclinación pélvica	100.0	12.8218	6.778658	-5.85	8.8025	13.485	16.7875	29.89	-0.194265	0.189884
ángulo de lordosis lumbar	100.0	43.5423	12.361581	19.07	35.0000	42.640	51.6025	90.56	0.750793	1.220533
pendiente sacra	100.0	38.8638	9.623776	17.39	32.3425	37.060	44.6050	67.20	0.431472	0.125237
radio pélvico	100.0	123.8912	9.013755	100.50	118.1800	123.875	129.0400	147.89	0.010833	0.117276
grado de espondilolistesis	100.0	2.1870	6.307020	-11.06	-1.5100	1.155	4.9675	31.17	1.691499	5.596433

Figura 2: Medidas de tendencias.

Para tener más información también se realizan gráficos de caja e histogramas para cada covariable separando según la etiqueta.

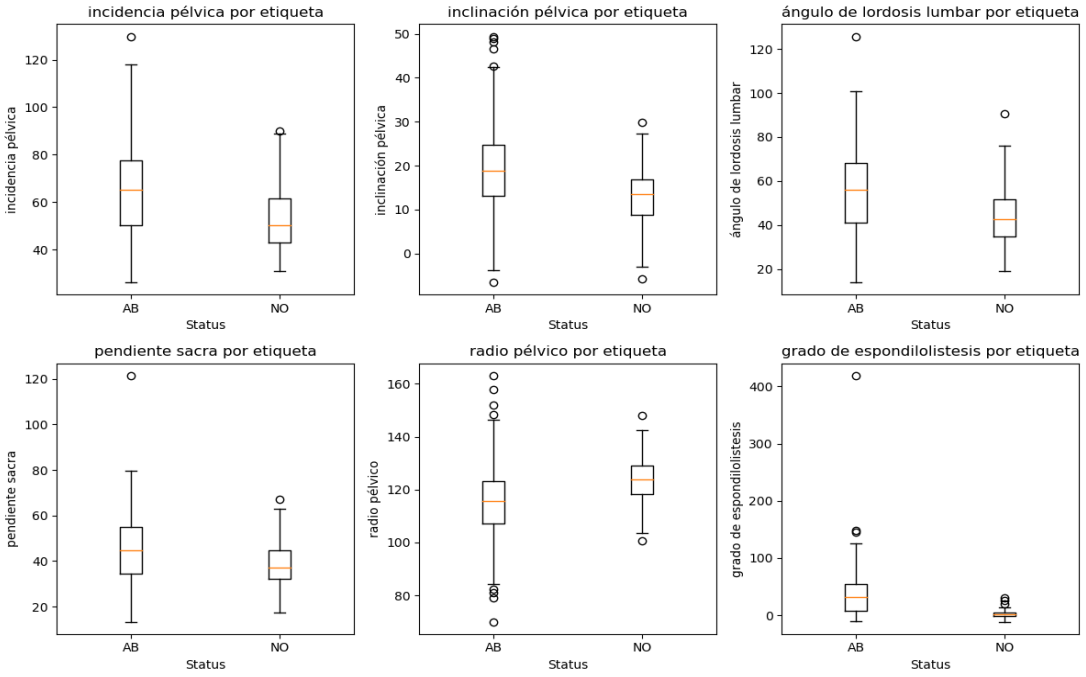

Figura 3: Boxplot.

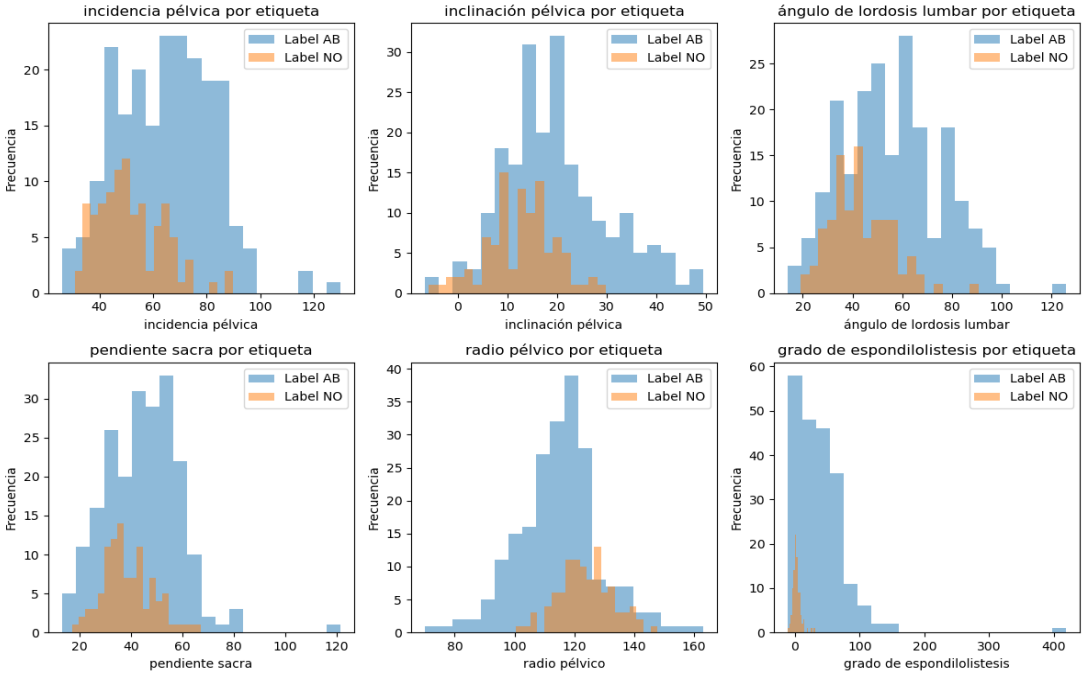

Figura 4: Histogramas por covariable.

Sobre las medidas de tendencia y boxplot se pueden expresar las siguientes ideas:

- Se observa que en términos generales el grupo `AB` posee una desviación estándar más alta en todas las características.
- En esto se nota que `grado de espondilolistesis` representa una gran diferencia entre un grupo y otro respecto a la desviación estandar y principalmente en que toma valores muy altos para `AB`.
- Como regla general parece ser que `AB` posee valores más altos en sus carácteristicas respecto a `NO` salvo talvez en `radio pélvico`.
- Por simple inspección no se descarta que las distribuciones no sean normales, a excepción de `grado de espondilolistesis`.

Con los gráficos se refuerza la hipótesis de normalidad expresada en el último punto, además en conjunto con el análisis de las medidas de tendencia central se puede intuir que las distribuciones por parecidas que parezcan pueden diferir en su media.

c. Para diferentes parejas de covariables, reporte los gráficos de dispersión (es decir, grafique X_i versus X_j) usando diferentes colores/símbolos de acuerdo a las distintas etiquetas. Comente los patrones que observa en estos gráficos.

Respuesta: Este código y los anterior pueden ser vistos a detalle en el repositorio de GitHub por su longitud. Primero estudiando los datos del conjunto en general y luego separando por las clases de `status`:

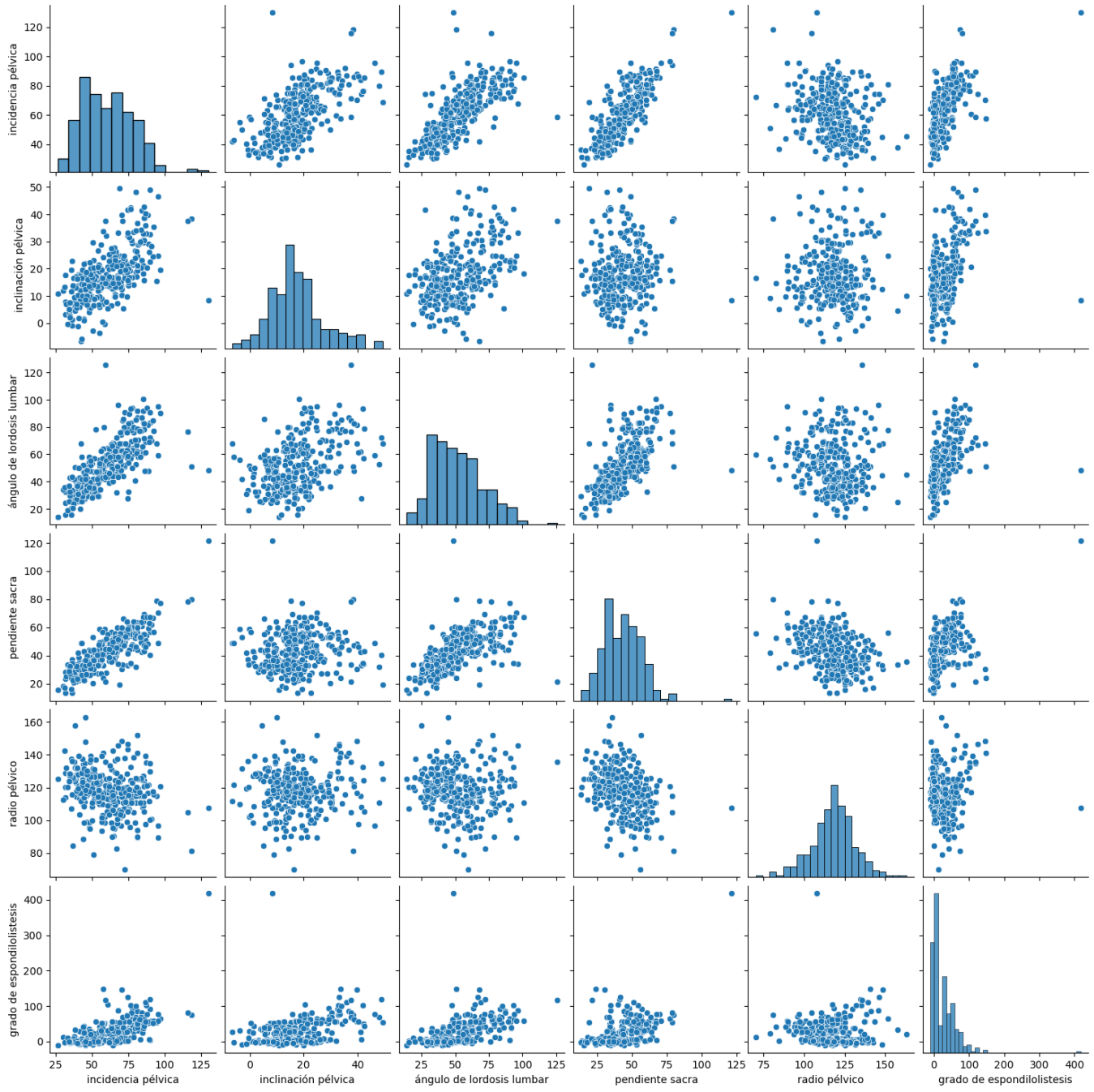

Figura 5: Gráficos de dispersión e histograma.

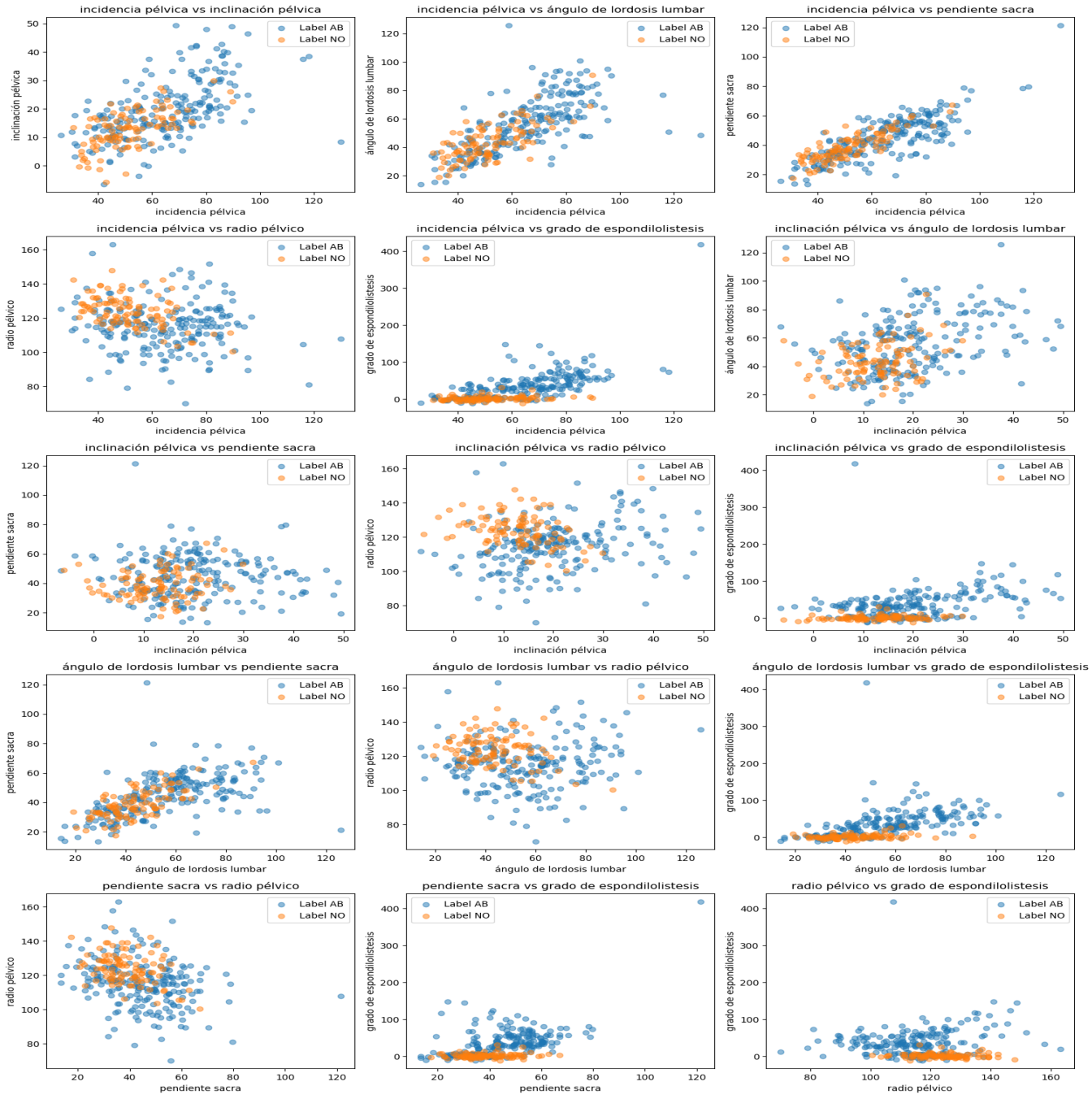

Figura 6: Gráficos de dispersión.

Observando los primeros 3 gráficos (izquierda a derecha) se puede observar cierta relación proporcional que puede indicar una fuerte correlación que se estudiará con un mapa de correlación.

Por la diferencia de magnitud entre los `grados de espondilolistesis` de los grupos `AB` y `NO` no

se puede observar si existe o no algún tipo de relación para los grupos AB en dicha característica respecto al resto.

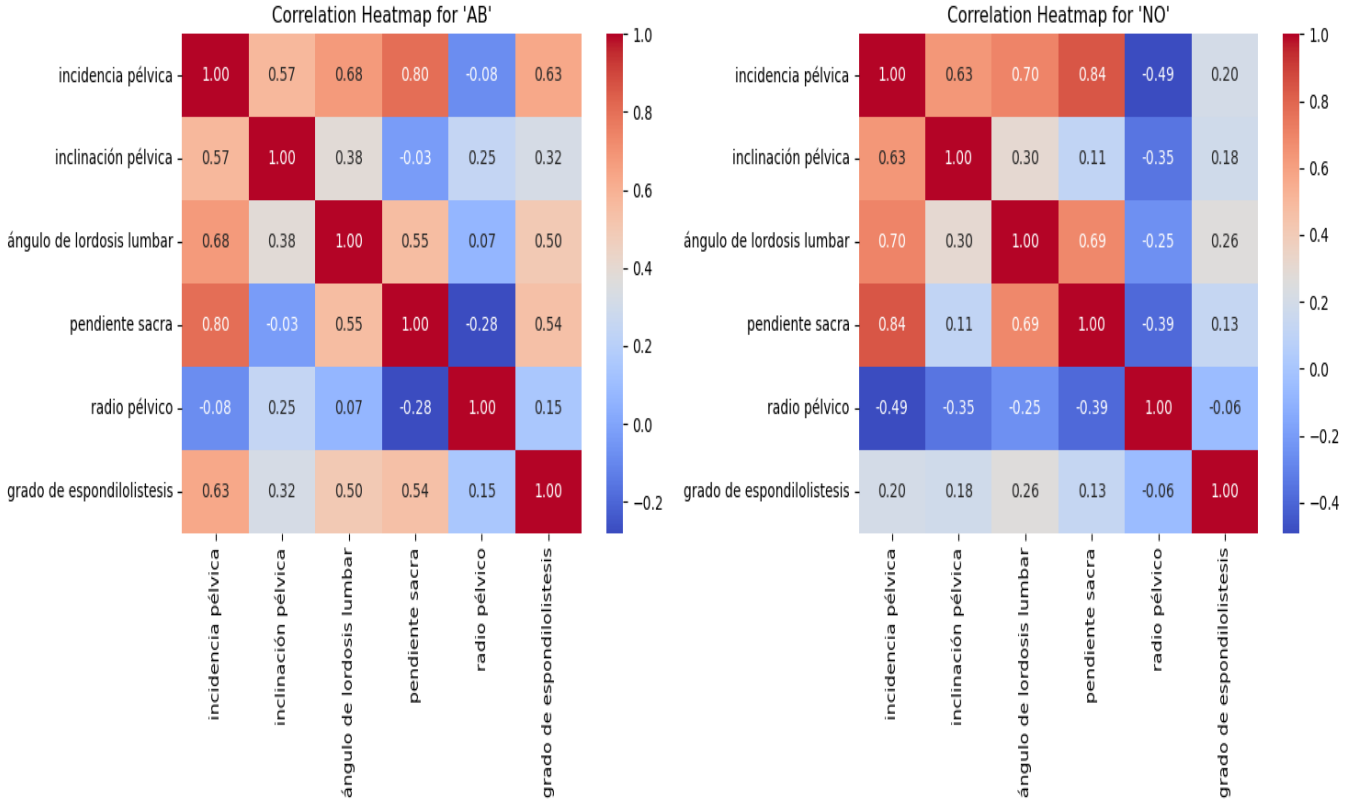

Figura 7: Mapas de correlación por etiqueta.

Note que las hipótesis fueron correctas, existen datos fuertemente correlaciones como por ejemplo `pendiente sacra` y `incidencia pélvica` entre otros, lo cuál puede suponer problemas para modelos que trabajen bajo supuestos de independencia y puede motivar a la eliminación de características para disminuir el riesgo de colinealidad o bien el uso de métodos de reducción de dimensionalidad.

Desde un análisis de datos, se recalca que en el grupo `AB` existe una correlación moderada de `grado de espondilolistesis` con respecto al resto de variables, la cual no se ve presente en el grupo `NO`.

Estos casos mencionados no son los únicos, estas dos características se repiten para otros features, notase el caso del `radio pélvico` como `incidencia pélvica`.

Vale notar que si el problema a estudiar fuera un problema de regresión y no de clasificación, sería más adecuado incluir la etiqueta y estudiar que covariable tiene más relación lineal con la variable objetivo.

d. Ajuste un modelo de regresión logística con todas las covariables. Reporte y comente los resultados obtenidos.

Respuesta: Utilizando la librería sklearn, se realiza la separación de conjunto de entrenamiento y conjunto de prueba:

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

X = df.drop('status', axis=1)  # Dado que 'status' es la etiqueta objetivo
    y = df['status']

# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=7) # random state for reproducibility
```

Listing 3: Obtener el conjunto de entrenamiento.

con esto se entrena el modelo con el siguiente código

```
# Crear y entrenar el modelo de regresion logistica
model = LogisticRegression()
model.fit(X_train, y_train)

# Predecir las etiquetas en el conjunto de prueba
y_pred = model.predict(X_test)
```

Listing 4: Entrenar el modelo.

y se entrega la matriz de confusión para evaluar el rendimiento del modelo.

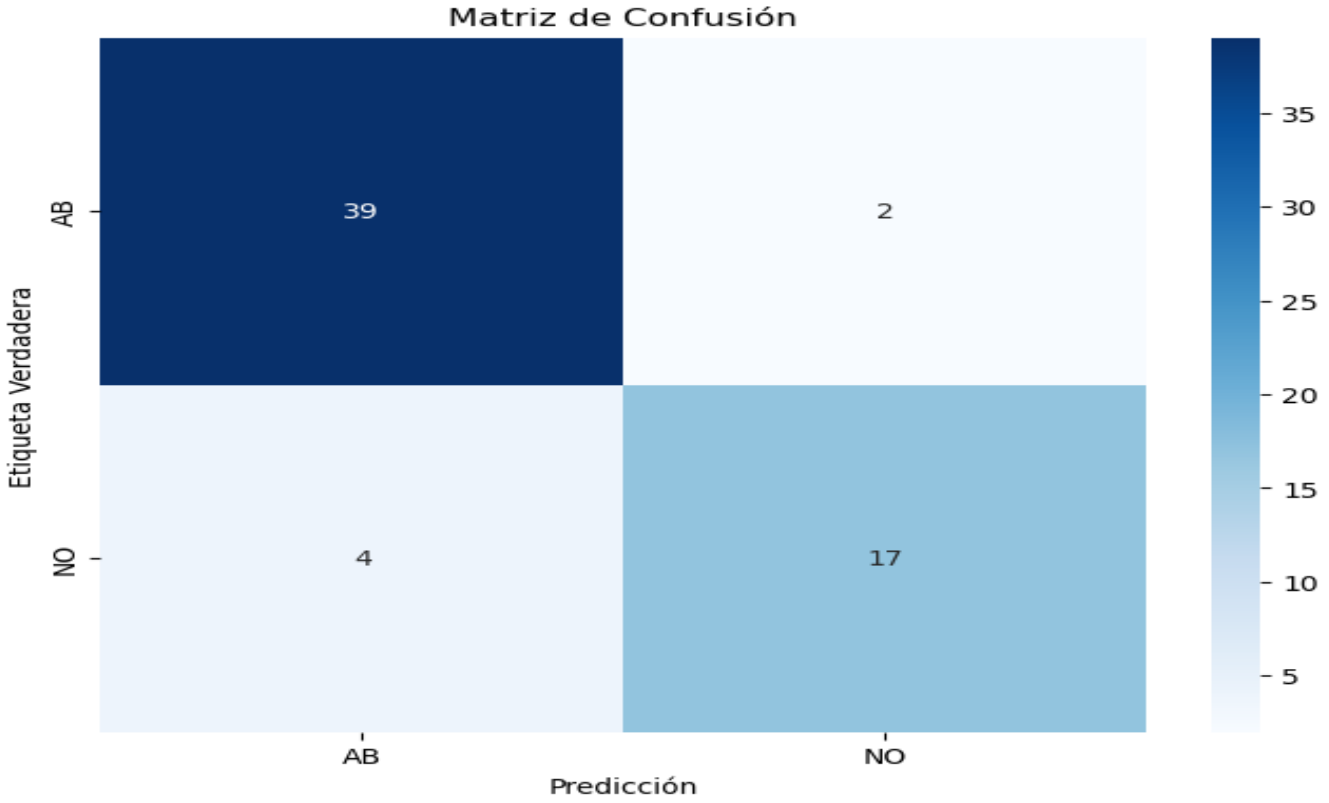

Figura 8: Matriz de confusión de regresión logística.

Pese a los problemas de colinealidad parece ser que la regresión logistica tiene un buen desempeño observado en la matriz de confusión.

Mediante la librería `Statsmodels` se entrena el modelo para obtener información más descriptiva:

```
import statsmodels.api as sm

y_trainstats = y_train.replace({'NO':0, 'AB':1})

logit_model = sm.Logit(y_trainstats, sm.add_constant(X_train)).fit()

print(logit_model.summary())
```

Listing 5: Logit_model in Statsmodels.

obteniendo lo siguiente

```
Optimization terminated successfully.
         Current function value: 0.301000
         Iterations 9
                           Logit Regression Results
==============================================================================
Dep. Variable:                 status   No. Observations:                  248
Model:                          Logit   Df Residuals:                      241
Method:                           MLE   Df Model:                            6
Date:                Tue, 05 Sep 2023   Pseudo R-squ.:                 0.5190
Time:                        12:58:55   Log-Likelihood:                -74.648
converged:                       True   LL-Null:                       -155.19
Covariance Type:            nonrobust   LLR p-value:                 3.486e-32
================================================================================================
                                   coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                           14.1506      3.405      4.156      0.000       7.477      20.824
incidencia pélvica             -26.5099     38.924     -0.681      0.496    -102.799      49.779
inclinación pélvica             26.5906     38.928      0.683      0.495     -49.706     102.887
ángulo de lordosis lumbar       -0.0233      0.025     -0.915      0.360      -0.073       0.027
pendiente sacra                 26.4230     38.925      0.679      0.497     -49.869     102.715
radio pélvico                   -0.0993      0.024     -4.134      0.000      -0.146      -0.052
grado de espondilolistesis       0.1605      0.025      6.358      0.000       0.111       0.210
================================================================================================

Possibly complete quasi-separation: A fraction 0.11 of observations can be
perfectly predicted. This might indicate that there is complete
quasi-separation. In this case some parameters will not be identified.
```

Figura 9: Estadísticas del modelo.

e. Con el modelo ajustado en el inciso anterior prediga la etiqueta de un paciente con las siguientes características:

$$(X_1, X_2, X_3, X_4, X_5, X_6) = (60, 20, 50, 50, 100, 10)$$

Respuesta:

```python
# Nueva observacion en el mismo formato que tus datos originales
new_observation = [[60, 20, 50, 50, 100, 10]]

# Crear un DataFrame temporal para la nueva observacion
new_df = pd.DataFrame(new_observation, columns=X.columns)

# Realizar la prediccion
predicted_label = model.predict(new_df)

print(f'Etiqueta predicha
    para la nueva observacion: {predicted_label[0]}')
```

Listing 6: Predecir con regresión logística.

```
Etiqueta predicha para la nueva observación: AB
```

f. ¿Es razonable disminuir el número de covariables? Justifique su análisis rigurosamente.

Respuesta: Es razonable disminuir el número de covariables dado que existen pares de covariables altamente correlacionadas sin importar el grupo al que pertenecen, por lo cuál para un modelo de regresión esta característica puede ser redundante mientras que para otro modelo que asuma independencia, puede afectar su capacidad predictiva, en ambos casos, la presencia de covariables con alta correlación tiene un efecto negativo y por ende es recomendable eliminar columnas de manera conveniente.

Para decidir que columna eliminar o que método aplicar para solventar este problema se utilizará el test de Wald. Para esto notemos que en un punto anterior se observa que `pendiente sacra` entrega un mayor valor de `P>|z|`, por lo tanto eliminando dicha columna:

```
X_trainstats = X_train.drop(columns=['pendiente sacra'])
logit_model = sm.Logit(y_trainstats, sm.add_constant(X_trainstats)).fit()

print(logit_model.summary())
```

Listing 7: Eliminar primera columna.

se obtiene las nuevas estadísticas:

```
Optimization terminated successfully.
         Current function value: 0.301934
         Iterations 9
                           Logit Regression Results
==============================================================================
Dep. Variable:                 status   No. Observations:                  248
Model:                          Logit   Df Residuals:                      242
Method:                           MLE   Df Model:                            5
Date:                Tue, 05 Sep 2023   Pseudo R-squ.:                 0.5175
Time:                        12:58:55   Log-Likelihood:                -74.880
converged:                       True   LL-Null:                       -155.19
Covariance Type:            nonrobust   LLR p-value:                 7.291e-33
================================================================================================
                                   coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                           14.1441      3.422      4.133      0.000       7.437      20.851
incidencia pélvica              -0.0885      0.033     -2.648      0.008      -0.154      -0.023
inclinación pélvica              0.1667      0.041      4.043      0.000       0.086       0.248
ángulo de lordosis lumbar       -0.0212      0.025     -0.841      0.400      -0.071       0.028
radio pélvico                   -0.0991      0.024     -4.094      0.000      -0.146      -0.052
grado de espondilolistesis       0.1602      0.025      6.360      0.000       0.111       0.210
================================================================================================

Possibly complete quasi-separation: A fraction 0.11 of observations can be
perfectly predicted. This might indicate that there is complete
quasi-separation. In this case some parameters will not be identified.
```

Figura 10: Estadísticas del modelo eliminando una columna.

notese que nuevamente se puede elimininar una columna

```
X_trainstats = X_train.drop(columns=['pendiente sacra', 'angulo de
    lordosis lumbar'])
logit_model = sm.Logit(y_trainstats, sm.add_constant(X_trainstats)).fit()

print(logit_model.summary())
```

Listing 8: Eliminar dos columnas.

con lo que se obtiene

```
Optimization terminated successfully.
         Current function value: 0.303380
         Iterations 9
                           Logit Regression Results
==============================================================================
Dep. Variable:                 status   No. Observations:                  248
Model:                          Logit   Df Residuals:                      243
Method:                           MLE   Df Model:                            4
Date:                Tue, 05 Sep 2023   Pseudo R-squ.:                 0.5152
Time:                        12:58:55   Log-Likelihood:                -75.238
converged:                       True   LL-Null:                       -155.19
Covariance Type:            nonrobust   LLR p-value:                1.532e-33
================================================================================================
                                   coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                          14.5251      3.409      4.261      0.000       7.844      21.206
incidencia pélvica             -0.1085      0.024     -4.493      0.000      -0.156      -0.061
inclinación pélvica             0.1756      0.040      4.361      0.000       0.097       0.254
radio pélvico                  -0.1019      0.024     -4.250      0.000      -0.149      -0.055
grado de espondilolistesis      0.1561      0.025      6.349      0.000       0.108       0.204
================================================================================================

Possibly complete quasi-separation: A fraction 0.12 of observations can be
perfectly predicted. This might indicate that there is complete
quasi-separation. In this case some parameters will not be identified.
```

Figura 11: Estadísticas del modelo eliminando dos columnas.

teniendo finalmente un modelo con todas las covariables significativas. Notese que se debe hacer la eliminación de un valor y un valor dado que la significancia dada por el test depende de las otras covariables.

g. ¿Es adecuado utilizar el método de Bayes ingenuo en este conjunto de datos? Justifique su respuesta.

Respuesta: No es adecuado utilizar este modelo ya que asume independencia entre las covariables, lo cual en los puntos anteriores se vió fuertemente criticado por la alta presencia de correlación.

h. Independientemente de la respuesta del inciso anterior, ajuste un modelo de Bayes ingenuo. Reporte las funciones de densidad estimadas. Comente sus resultados.

Respuesta: En el código se puede ver más detallamente como realizar esto en `Python` y `R`. Desde `R` se puede entrenar el modelo de Bayes ingenuo estimando las densidades mediantes distribuciones normales basado en lo visto en los gráficos y estadísticos anteriores:

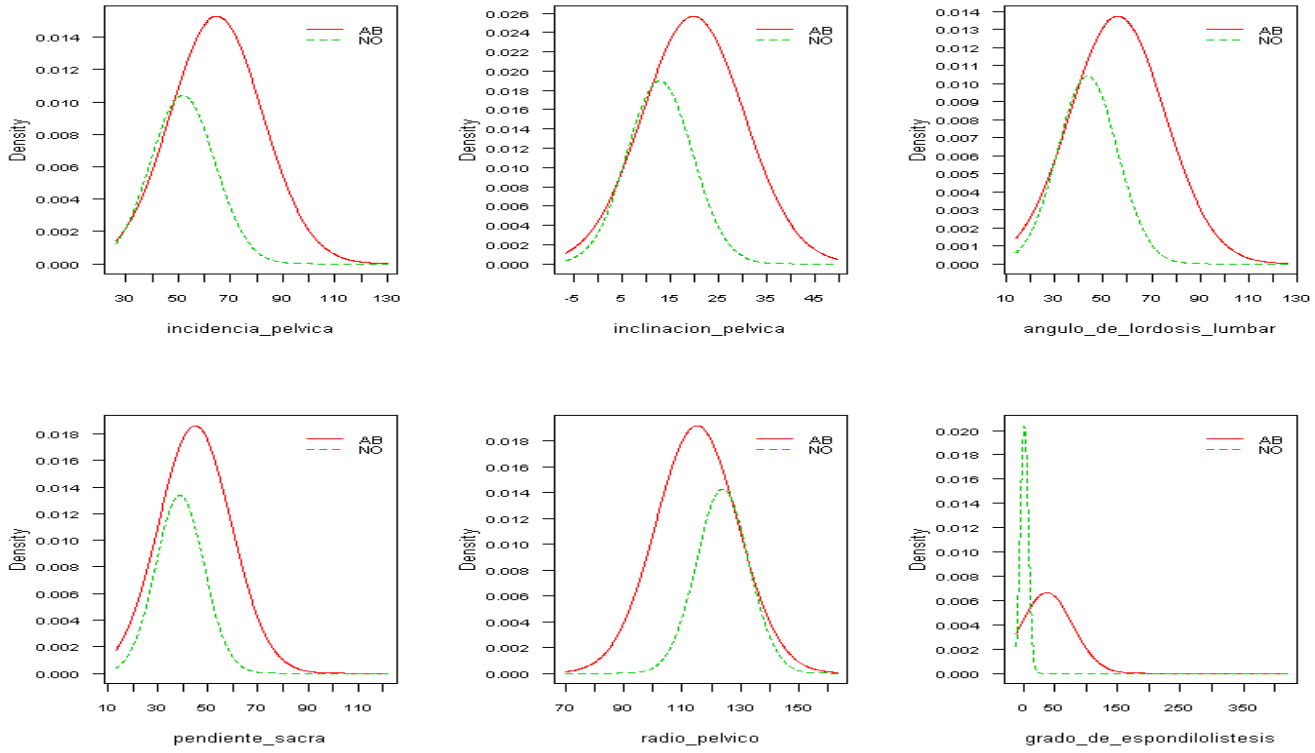

Figura 12: Densidades estimadas.

Observando los gráficos, se podría especular que el modelo puede fallar al tener densidades parecidas en localización (salvo la última variable).

i. Ajuste un árbol de clasificación. Reporte y comente sus resultados.

Respuesta: Nuevamente realizando los procedimientos necesarios en `R` el modelo entrenado corresponde a

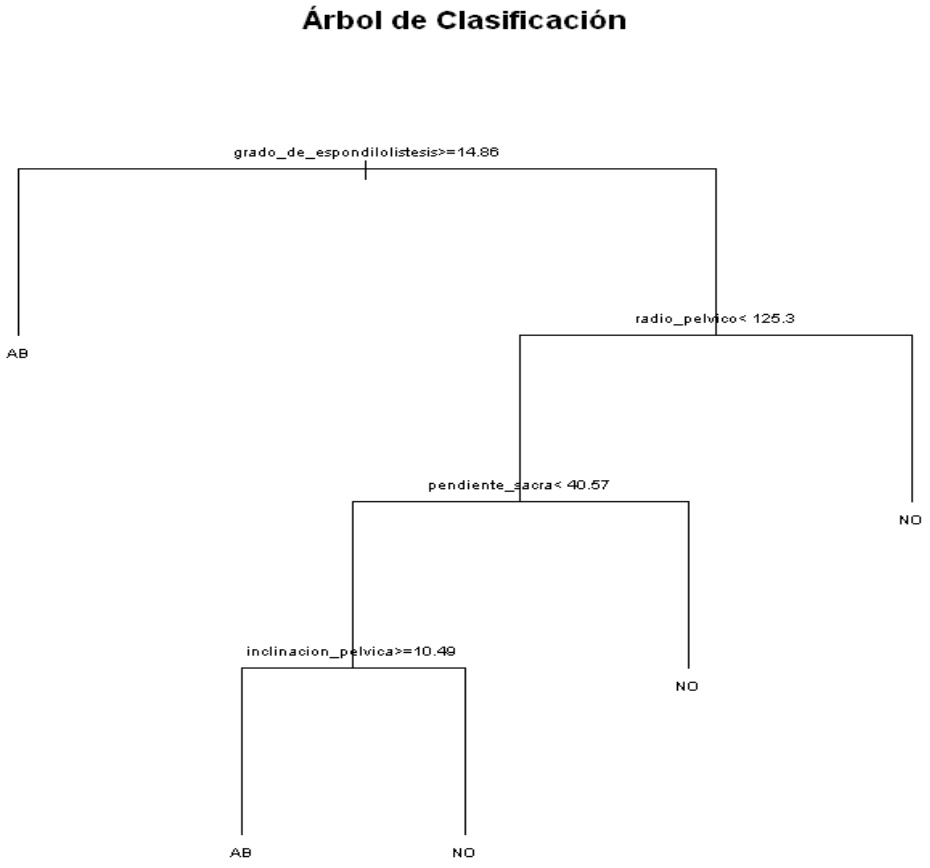

Figura 13: Árbol de decisión generado.

Note que el modelo ajustado no utiliza todas las covariables al momento de decidir, pero estas si fueron ocupadas al momento de construir el modelo.

j. Use los ajustes de los incisos h. y i. para predecir la etiqueta de un paciente con las mismas características que en el inciso e. ¿La conclusión coincide con la regresión logística?

Respuesta: Para esto usaremos los modelos entrenados en `Python` mediante procedimientos más comunes, estos arrojaron el siguiente resultado:

```
Etiqueta predicha mediante logistic regression: AB
Etiqueta predicha mediante Naive Bayes: AB
```

```
Etiqueta predicha mediante Decision Tree: NO
```

Notese que las predicciones no son iguales, sólo Naive Bayes coincide con la regresión logística, mientras que el árbol de decisión discrepa en la solución.

3. Anexo

```r

#install.packages("naivebayes")

library("naivebayes")# Ruta del archivo

# Ajustar un modelo de Bayes Ingenuo
modelo <- naive_bayes(status ~ incidencia_pelvica
            + inclinacion_pelvica
            + angulo_de_lordosis_lumbar
            + pendiente_sacra + radio_pelvico
            + grado_de_espondilolistesis, data = datos, usekernel = F
               )
# Particionar el espacio de graficos
par(mfrow = c(2, 3))

# Generar los graficos del modelo
plot(modelo)
```

Listing 9: Naive Bayes R.

```r
#install.packages("rpart")

# Cargar la libreria necesaria para arboles de clasificacian
library("rpart")

# Ajustar el arbol de clasificacion
arbol <- rpart(status ~ incidencia_pelvica + inclinacion_pelvica +
    angulo_de_lordosis_lumbar +
                pendiente_sacra + radio_pelvico + grado_de_
                   espondilolistesis, data = datos)

# Personalizar el estilo del grafico
par(cex=0.8)
plot(arbol, uniform = TRUE, main = "Arbol de Clasificacion", margin =
     0.1)
text(arbol, cex = 0.6)
```

Listing 10: Classification Tree R.