TDVI_TP1

2025-03-23

Trábajo Práctico 1

1. Introduccion al problema: origen y variables principales del conjunto de datos

El conjunto de datos elegido proviene de Kaggle y se utiliza para predecir si un prestamo solicitado por una persona sera aprobado o rechazado, basandose en distintas caracteristicas del solicitante.El conjunto fue enriquecido con variables adicionales basadas en *Riesgo Financiero* para la Aprobación de Préstamos. Además, se aplicó *SMOTENC (Synthetic Minority Over-sampling Technique for Nominal and Continuous)* para generar nuevos puntos de datos y ampliar el conjunto de instancias. El conjunto de datos contiene 45,000 registros y 14 variables. A continuación se describen las variables clave:

Table 1: Resumen de los atributos del dataset

Columna	Descripción	Tipo.de.Dato
person_age	Edad de la persona	Float
person_gender	Género de la persona	Categórico
person_education	Nivel de educación más alto alcanzado	Categórico
person_income	Ingreso anual de la persona	Float
person_emp_exp	Años de experiencia laboral	Entero
person_home_ownership	Estado de propiedad de la vivienda (ej., alquiler, propia)	Categórico
loan_amnt	Monto del préstamo solicitado	Float
loan_intent	Propósito del préstamo	Categórico
loan_int_rate	Tasa de interés del préstamo	Float
loan_percent_income	Monto del préstamo como porcentaje del ingreso anual	Float
cb_person_cred_hist_length	Longitud del historial de crédito en años	Float
credit_score	Puntaje de crédito de la persona	Entero
previous_loan_defaults_on_file	Indicador de los anteriores incumplimientos de préstamo	Categórico
loan_status (objetivo)	Estado del préstamo: 1 = aprobado; 0 = rechazado	Entero (Binario)

Utilizar un árbol de decisión para modelar este problema es adecuado por varias razones. Primero, permite realizar tareas de clasificación binaria, como predecir si un préstamo será aprobado o rechazado, ya que divide los datos en función de las características más importantes. Segundo, maneja eficientemente datos mixtos, es decir, tanto numéricos (como ingresos y puntajes de crédito) como categóricos (como género y estado civil), sin necesidad de transformaciones complicadas. Tercero, la estructura de un árbol de decisión permite dividir los datos en subgrupos homogéneos, lo que ayuda a identificar clientes con características similares que pueden tener un comportamiento parecido, como los que tienen un puntaje de crédito bajo o altos ingresos. Esto es importante porque las relaciones entre características, como ingresos y puntaje de crédito, no siempre son lineales, y los árboles pueden manejar estas interacciones. Finalmente, su gran ventaja es la interpretabilidad; los árboles de decisión son fáciles de entender y explican claramente las razones detrás de cada clasificación, lo cual es fundamental en el ámbito bancario, donde se necesitan justificar las decisiones de aprobación o rechazo de préstamos.

2. Preparación de los datos

Es necesario preprocesar los datos, para lo cual realizaremos un análisis que incluya la verificación de las variables y su clasificación, la normalización o escalado de las variables, la detección de valores faltantes, la identificación de posibles anomalías y balanceado de datos.

Como se mencionó anteriormente, las variables están correctamente clasificadas según su tipo (categórico o numérico). Dado que vamos a utilizar árboles de decisión, no es necesario normalizar ni escalar los datos, ya que este modelo no depende de las magnitudes de las variables. Los árboles de decisión dividen los datos basándose en los valores específicos de las características, por lo que la escala de las variables no influye en el desempeño del modelo. Asimismo, detallaremos que no existen valores faltantes en el conjunto de datos, como se puede observar en la Tabla 2.

Table 2: Resumen de valores nulos

Cantidad.de.atributos.verificados	Atributos.con.0.nulos
14	14

También corroboramos que no huebiese una cantidad excesiva de filas repetidas, de hecho, nos dió cero.

```
duplicados<-duplicated(data)
cant<-sum(duplicados)
cant
```

```
## [1] 0
```

Habiendo mencionado lo anterior, continuaremos con el análisis de las distribuciones de los distintos atributos. Como se puede observar en el gráfico XX, se aplicó una transformación logarítmica a tres atributos clave: Edad, Ingresos e Historial de Crédito. Esta transformación es relevante porque, al ser creciente, no altera la distribución subyacente de los datos, pero sí comprime el rango de los valores grandes y expande el de los valores pequeños, lo que facilita su análisis.

Es importante destacar la relevancia de analizar la distribución de cada atributo. En este caso, observamos que atributos como Edad, Monto del Préstamo e Ingresos presentan una distribución sesgada a la derecha, mientras que el Puntaje de Crédito está sesgado a la izquierda. Esta asimetría podría generar predicciones erróneas para los valores extremos en cada uno de estos atributos. Además, notamos que el Historial de Crédito no sigue una distribución bien definida, lo que añade complejidad a su análisis.

Distribución de Variables Numéricas

Distribución de Variables Categóricas

Luego de analizar la distribución de las variables categóricas, no se identifican patrones relevantes ni desequilibrios significativos que requieran mención.

Anomalías

```
##   person_age person_gender person_education person_income person_emp_exp
## 1        144          male         Bachelor        300616            125
## 2        144          male        Associate        241424            121
## 3        144        female        Associate       7200766            124
##   person_home_ownership loan_amnt loan_intent loan_int_rate loan_percent_income
## 1                  RENT      4800     VENTURE         13.57                0.02
## 2              MORTGAGE      6000   EDUCATION         11.86                0.02
## 3              MORTGAGE      5000    PERSONAL         12.73                0.00
##   cb_person_cred_hist_length credit_score previous_loan_defaults_on_file
## 1                          3          789                             No
```

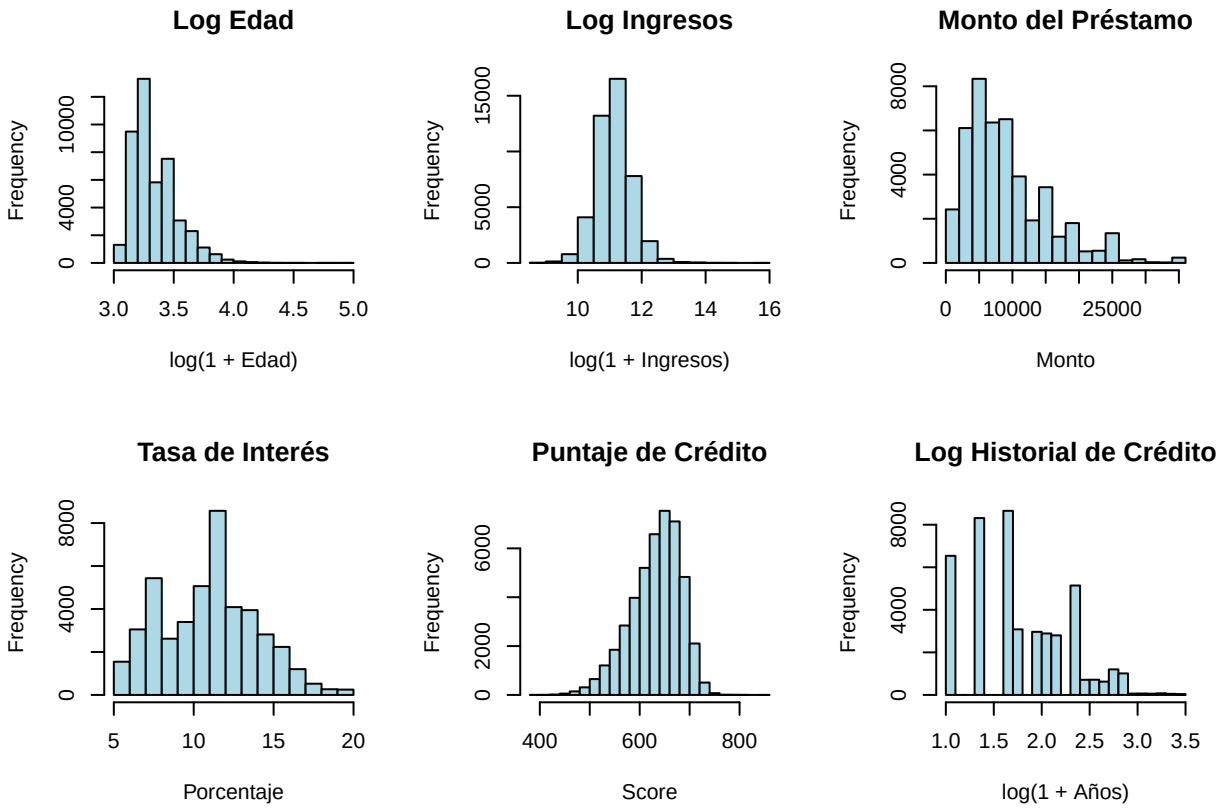


Figure 1: Distribución de Variables Numéricas

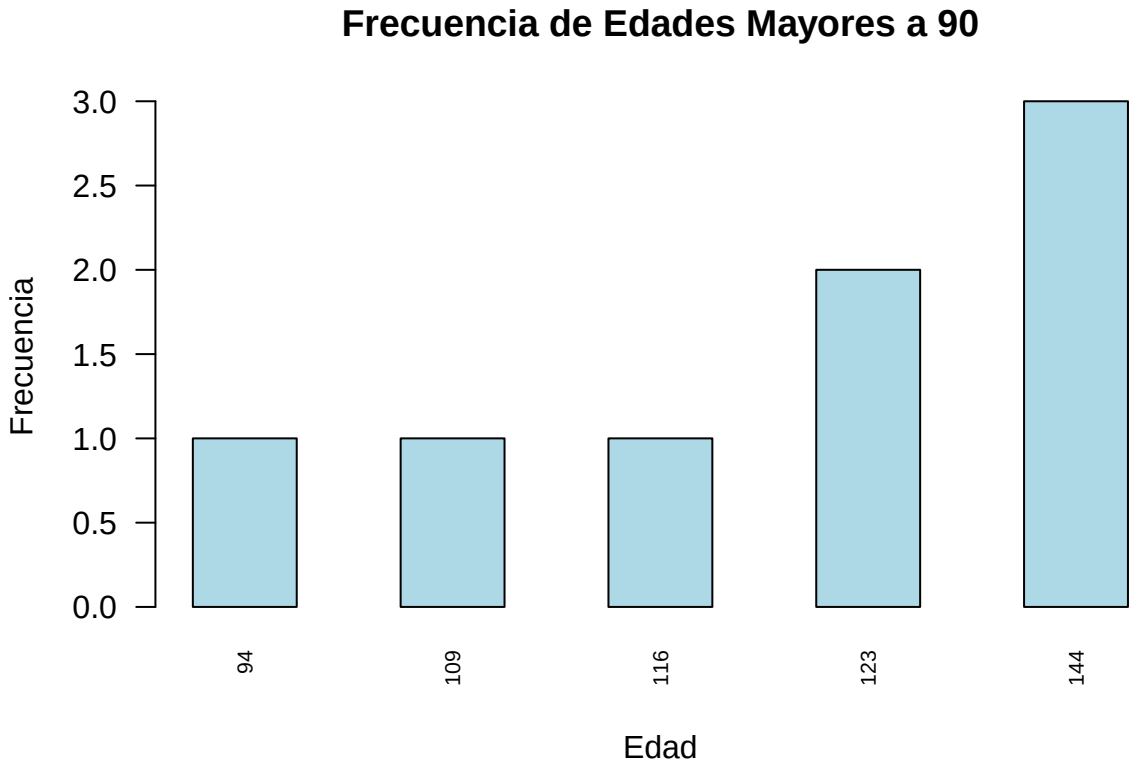


Figure 2: Distribución de Edades Mayores a 90

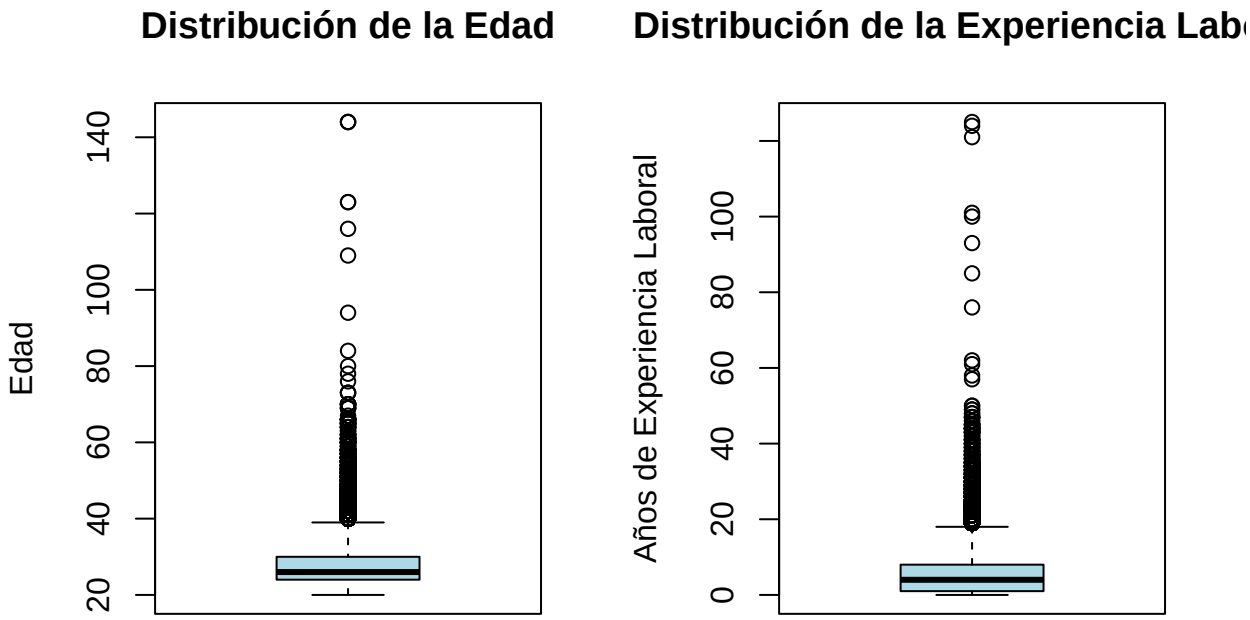


Figure 3: Boxplots de Experiencia Laboral y Edad

```
## 2                          2          807                              No
## 3                         25          850                              No
##   loan_status log_person_age log_person_income log_cb_person_cred_hist_length
## 1           0       4.976734          12.61359                        1.386294
## 2           0       4.976734          12.39431                        1.098612
## 3           0       4.976734          15.78970                        3.258097
```

```
## Cantidad de personas con experiencia laboral mayor a su edad: 0
```

Chequeo de datos balanceados

Para evaluar el balance de nuestro conjunto de datos, analizamos las proporciones de las categorías de la variable de interés, en este caso, el estado del préstamo (aceptado o rechazado). A continuación, creamos un gráfico de barras que muestra la proporción de "Aceptado" y "Rechazado" en el conjunto de datos.

Al examinar el gráfico y las proporciones obtenidas, observamos que aproximadamente el 22% de los registros corresponden a "Aceptados" y el 77% a "Rechazados". Esto indica que el conjunto de datos no está completamente balanceado, aunque la desproporción no es extremadamente alta. La desproporción en el conjunto de datos podría generar un sesgo hacia la clase mayoritaria ("Rechazados"), lo que podría resultar en un mejor desempeño del modelo para predecir esta clase y un desempeño inferior para la clase minoritaria ("Aceptados").

Sin embargo, decidimos no intervenir en el conjunto de datos eliminando registros o utilizando técnicas de balanceo como sobremuestreo o submuestreo. Esto se debe a que tales técnicas podrían introducir sesgos adicionales o alterar la representatividad del conjunto de datos. En cambio, optamos por mantener la estructura original y tener en cuenta las proporciones de las clases al entrenar, validar y evaluar el modelo.

3. Construcción de árbol de decisión básico

```
library(caret)
```

```
## Loading required package: lattice
```

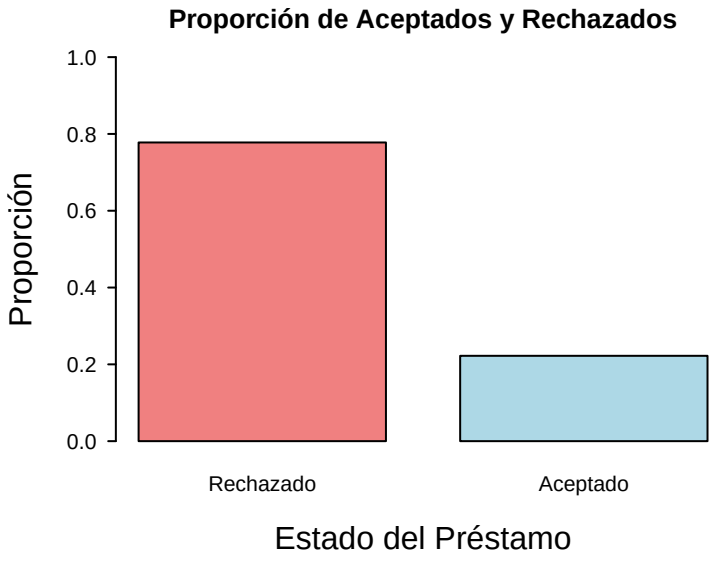


Figure 4: Proporción de Aceptados y Rechazados

```
library(rpart)
library(rpart.plot)
```

```
## Warning: package 'rpart.plot' was built under R version 4.2.3
```

```
# Fijar semilla para replicabilidad
set.seed(123)

# Crear índices estratificados para entrenamiento (70%)
train_index <- createDataPartition(data$loan_status, p = 0.70, list = FALSE)

# Conjunto de entrenamiento
train_data <- data[train_index, ]

# Resto de los datos (30% para validación y testeo)
remaining <- data[-train_index, ]

# Dividir el 30% restante en validación (15%) y testeo (15%), manteniendo la proporción de loan_status
val_index <- createDataPartition(remaining$loan_status, p = 0.50, list = FALSE)
val_data <- remaining[val_index, ]
test_data <- remaining[-val_index, ]

# Verificar la distribución de loan_status en cada conjunto
prop.table(table(train_data$loan_status))  # Proporción en entrenamiento
```

```
##
##         0         1
## 0.7786349 0.2213651
```

```
prop.table(table(val_data$loan_status))    # Proporción en validación
```

```
##
##         0         1
## 0.7739259 0.2260741
```

```
prop.table(table(test_data$loan_status))   # Proporción en testeo
```

```
##
##         0         1
## 0.7776296 0.2223704
```

```
# Imprimir tamaños
cat("Tamaño de Entrenamiento:", nrow(train_data), "\n")
```

```
## Tamaño de Entrenamiento: 31500
```

```
cat("Tamaño de Validación:", nrow(val_data), "\n")
```

```
## Tamaño de Validación: 6750
```

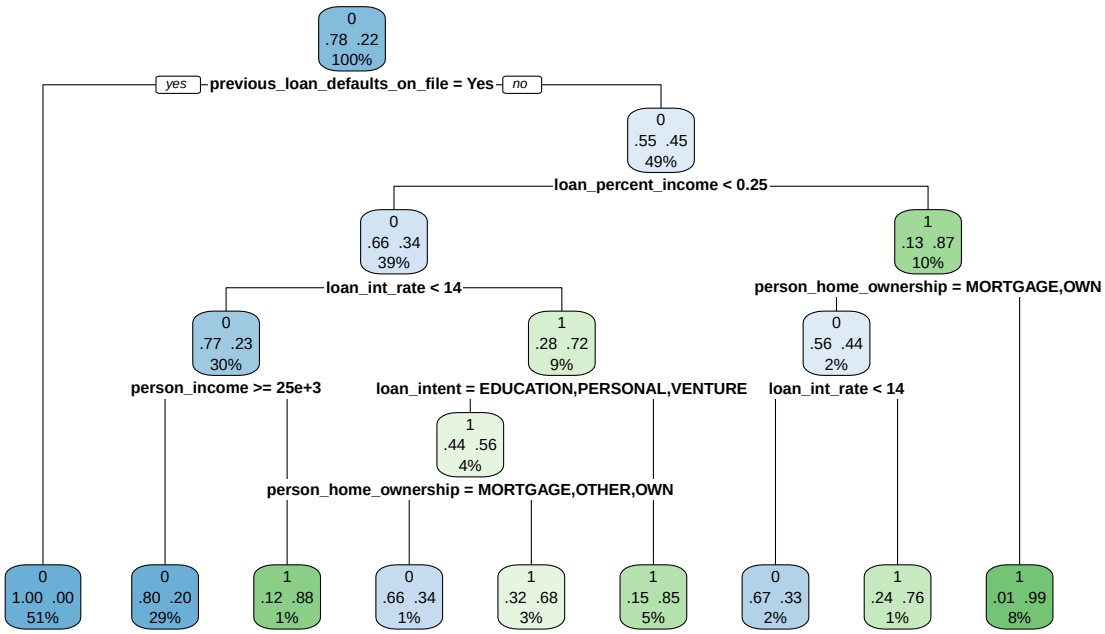
```
cat("Tamaño de Testeo:", nrow(test_data), "\n")
```

```
## Tamaño de Testeo: 6750
```

```
tree <- rpart(formula = loan_status ~ person_age + person_gender + person_education + person_income + pe
              data = train_data,
              method = "class")

rpart.plot(tree, main = "Árbol de Decisión", type = 2, extra = 104)
```

Árbol de Decisión



Al realizar el análisis, mantuvimos la proporción de otorgamiento de préstamos (78%) y rechazo (22%) a lo largo de las etapas de entrenamiento, validación y testeo. En la descripción del árbol de decisión, observamos que cada nodo se compone de tres filas:

La primera fila indica la categoría de "no" (rechazo del préstamo).

La segunda fila muestra la proporción de casos dentro del subgrupo.

La tercera fila indica cuántos datos se concentran en ese nivel del árbol.

El árbol de decisión muestra que el primer factor que se utiliza para determinar si se le otorga un préstamo es la existencia de defaults previos (previous_loan_defaults_on_file), donde la presencia de antecedentes implica una altísima probabilidad de no pago. En ausencia de defaults, el siguiente factor relevante es la relación entre el monto solicitado y el ingreso (loan_percent_income), siendo los valores altos indicativos de mayor riesgo. Además, la tasa de interés del préstamo (loan_int_rate) cumple un rol clave: tasas superiores al 14% incrementan significativamente la probabilidad de incumplimiento. El nivel de ingresos (person_income) también influye, ya que a menor ingreso, el riesgo aumenta, especialmente combinado con tasas altas. La intención del préstamo (loan_intent), cuando es para educación, fines personales o emprendimientos, aparece asociada a mayor riesgo. Finalmente, la tenencia de vivienda propia o bajo hipoteca (person_home_ownership) contribuye a elevar la probabilidad de default en combinación con tasas altas y un elevado porcentaje préstamo/ingreso. En conjunto, estas variables permiten identificar perfiles de alto riesgo de manera clara.

Como conclusión, utilizando esta instancia del modelo entrenado con el conjunto de datos train se le rechazaría al 83% el préstamo y al resto se le otorgaría.

Hiperprámetros por defecto del árbol

```
tree$control
```

```
## $minsplit
## [1] 20
##
## $minbucket
## [1] 7
##
## $cp
## [1] 0.01
##
## $maxcompete
## [1] 4
##
## $maxsurrogate
## [1] 5
##
## $usesurrogate
## [1] 2
##
## $surrogatestyle
## [1] 0
##
## $maxdepth
## [1] 30
##
## $xval
## [1] 10
```

A continuación se detallan los hiperparámetros utilizados por defecto en la construcción del árbol y el impacto que tiene cada uno en la estructura del modelo:

minsplit = 20: Define el tamaño mínimo de observaciones en un nodo para que el árbol considere realizar una partición. Un valor de 20 limita las divisiones a grupos con un tamaño suficientemente grande, evitando sobreajuste en nodos con pocos datos.

minbucket = 7: Indica el tamaño mínimo que pueden tener las hojas terminales. Esto asegura que cada hoja final contenga al menos 7 observaciones, promoviendo estabilidad en las predicciones.

cp = 0.01: Es el parámetro de complejidad que regula el proceso de poda. Solo se aceptan divisiones que logren mejorar la calidad del ajuste en al menos un 1%. Un valor de 0.01 representa un control intermedio, balanceando entre un árbol complejo y uno demasiado simple.

maxcompete = 4: Almacena hasta 4 splits "competidores" cercanos al mejor split en cada nodo. Esto es útil para analizar qué otras variables casi logran ser seleccionadas.

maxsurrogate = 5: Controla la cantidad máxima de variables sustitutas utilizadas cuando hay datos faltantes en la variable principal de división. La presencia de 5 surrogates mejora la capacidad del modelo para manejar datos incompletos. En nuestro caso, al no haber datos faltantes, este hiperparámetro pasa a ser irrelevante.

usesurrogate = 2: Define cómo se utilizan los surrogates. Con un valor de 2, el árbol emplea surrogate splits incluso si la variable principal está disponible, siempre que aporten mejora en el ajuste.

surrogatestyle = 0: Indica que la selección de variables sustitutas se realiza utilizando un índice de concordancia simple, priorizando velocidad y simplicidad.

maxdepth = 30: Fija la profundidad máxima que puede alcanzar el árbol. Un valor alto de 30 permite que, si los datos y el pruning lo permiten, el árbol tenga una gran profundidad.

xval = 10: Determina que la validación cruzada interna para el proceso de poda se realice en 10 particiones, aportando mayor robustez a la selección del tamaño óptimo del árbol.

4. Evaluación del árbol de decisión básico

```
##              0         1
## 1  0.007786885 0.9922131
## 23 0.007786885 0.9922131
## 37 0.007786885 0.9922131
## 42 0.119101124 0.8808989
## 46 1.000000000 0.0000000
## 63 0.803889073 0.1961109
```

```
##  1 23 37 42 46 63
##  1  1  1  1  0  0
## Levels: 0 1
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction         0         1
##          0 75.703704  6.977778
##          1  2.059259 15.259259
##
##                Accuracy : 0.9096
##                  95% CI : (0.9025, 0.9164)
##     No Information Rate : 0.7776
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.7163
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9735
##             Specificity : 0.6862
##          Pos Pred Value : 0.9156
##          Neg Pred Value : 0.8811
##              Prevalence : 0.7776
```

```
##          Detection Rate : 0.7570
##    Detection Prevalence : 0.8268
##       Balanced Accuracy : 0.8299
##
##        'Positive' Class : 0
##
```

```
## Accuracy: 0.9096296
```

```
## Precision: 0.9156065
```

```
## Recall: 0.9735188
```

```
## F1-score: 0.943675
```

```
## AUC-ROC: 0.9415207
```