Ejercicio Árbol Gender Discrimination

Jorge Fuertes Argüello

modelo a estimar Gender ~ Experience + Salary

1- Instalar y cargar librerías necesarias para el ejercicio

```
#install.packages("tree")
library(tree)
#install.packages("rpart")
library(rpart)
#install.packages("rpart.plot")
library(rpart.plot)
#install.packages("partykit")
library(partykit)
```

```
## Loading required package: grid
```

```
#install.packages("party")
library(party)
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: modeltools
```

```
## Loading required package: stats4
```

```
## Loading required package: strucchange
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
##
## Attaching package: 'party'
```

```
## The following objects are masked from 'package:partykit':
##
##     cforest, ctree, ctree_control, edge_simple, mob, mob_control,
##     node_barplot, node_bivplot, node_boxplot, node_inner,
##     node_surv, node_terminal
```

2- Establecer Work Directory.

Establecemos un work directory el cual contiene el documento csv que vamos a utilizar para la estimación del modelo.

A partir de esto, vamos a definir un DataFrame (al que llamaremos “gender”“) de las 208 observaciones de tres variables: sexo (Gender), años de experiencia (Experience) y salario (Salary).

Hacemos un head(), para que nos muestre los primeros y así observar que se han descargado correctamente.

```
setwd("C:/Users/jorge/Desktop/JORGE/CUNEF/MASTER/TECNICAS DE CLASIFICACION/Clase03/Practica a
rbol decision/Ejercicio Gender Discrimination")

gender <- read.csv("http://www.biz.uiowa.edu/faculty/jledolter/DataMining/GenderDiscriminatio
n.csv")

head(gender, 6) # comprobamos el encabezado de gender
```

```
##   Gender Experience Salary
## 1 Female         15  78200
## 2 Female         12  66400
## 3 Female         15  61200
## 4 Female          3  61000
## 5 Female          4  60000
## 6 Female          4  68000
```

```
attach(gender)
```

3- Variable a estimar (Gender)

La variable a estimar será el sexo en función de los años de experiencia y el salario.

Previo a estimar el modelo, determinaremos el número de hombres y mujeres del DataFrame. Para una mejor visualización lo haremos mediante un histograma.

El resultado nos indica que dentro de nuestro DataFrame hay 140 mujeres y 68 hombres.

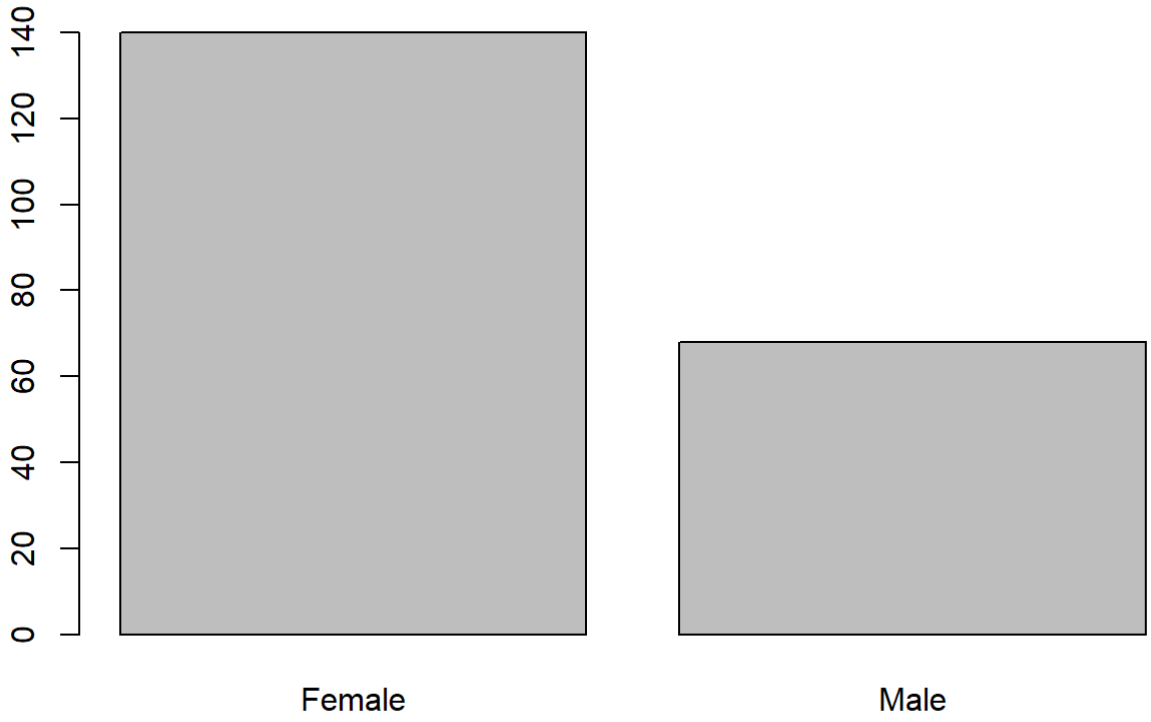
```
sum(gender$Gender=="Female")
```

```
## [1] 140
```

```
sum(gender$Gender=="Male")
```

```
## [1] 68
```

```
plot(gender$Gender)
```



4- Muestra de entrenamiento y validación

4.1- Definimos semilla aleatoria.

```
set.seed((124))
```

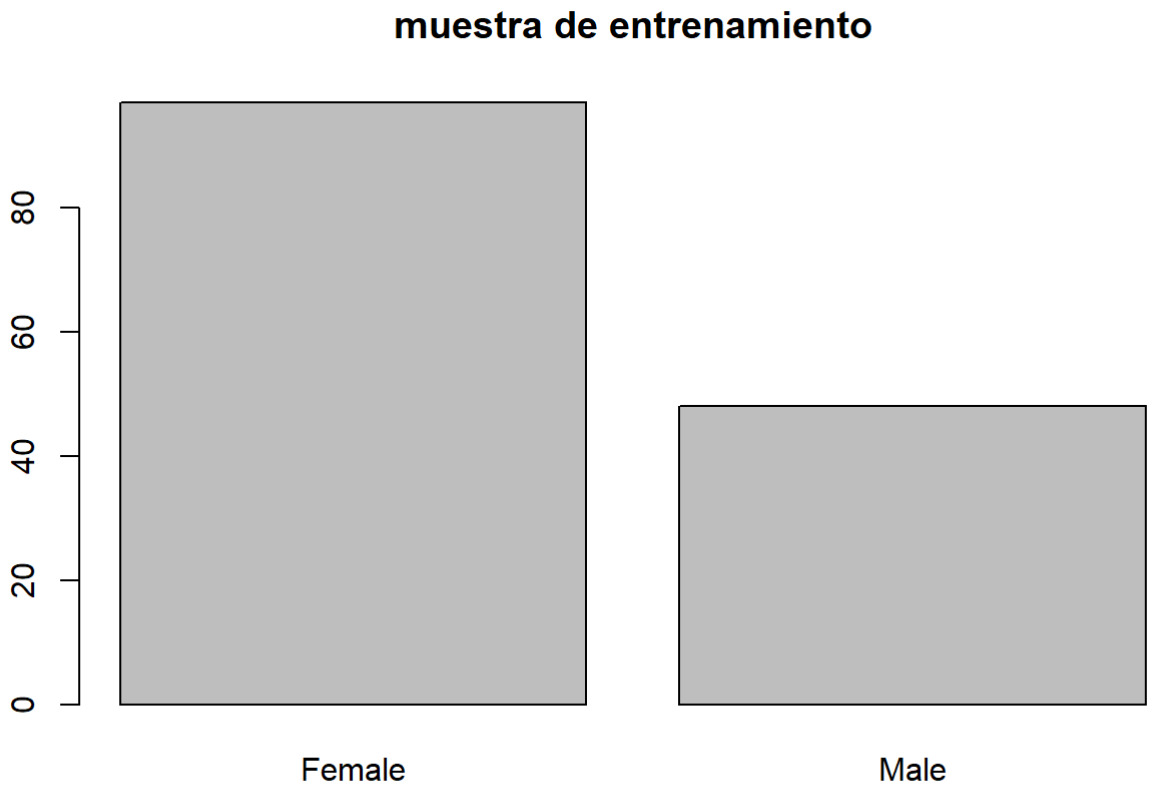
4.2- Muestra aleatoria de aprendizaje del árbol

Definimos una muestra aleatoria de aprendizaje del árbol, esta será el 70% de los datos que tenemos en el DataFrame:

```
train <- sample(nrow(gender), 0.7*nrow(gender))
par(mfrow=c(1,2))
```

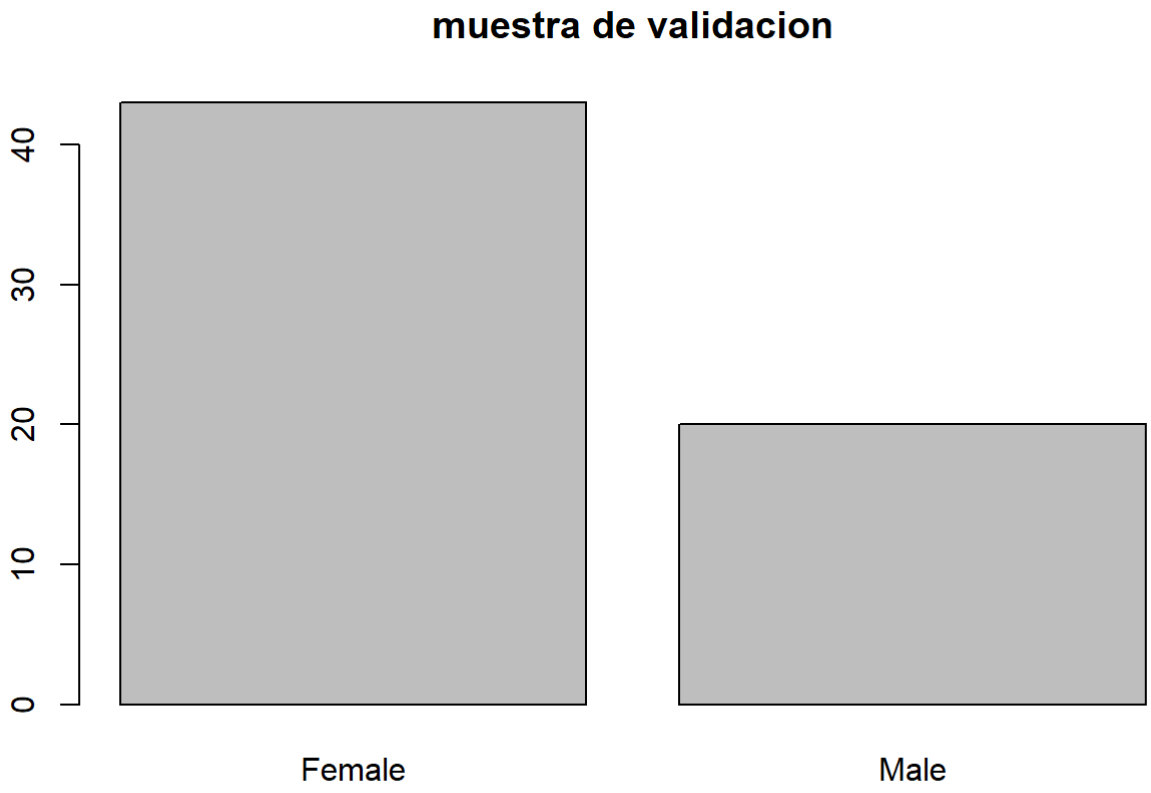
4.3- Muestra de entrenamiento (70%)

```
gender.train <- gender[train,]
plot(gender.train$Gender, main="muestra de entrenamiento")
```



4.4- Muestra de validación (30%)

```
gender.validate <- gender[-train,]
plot(gender.validate$Gender, main="muestra de validacion")
```



5- Creamos árbol de clasificación

Primero vamos a crear el árbol de clasificación con todos los nodos posibles:

```
arbol <- rpart(Gender~ ., data=gender.train,method="class", parms=list(split="information"))

print(arbol)
```

```
## n= 145
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 145 48 Female (0.6689655 0.3310345)
##    2) Salary< 90800 119 30 Female (0.7478992 0.2521008)
##      4) Experience>=6.5 89 14 Female (0.8426966 0.1573034) *
##      5) Experience< 6.5 30 14 Male (0.4666667 0.5333333)
##       10) Salary< 60950 7  1 Female (0.8571429 0.1428571) *
##       11) Salary>=60950 23  8 Male (0.3478261 0.6521739)
##         22) Experience< 4.5 9  4 Female (0.5555556 0.4444444) *
##         23) Experience>=4.5 14  3 Male (0.2142857 0.7857143) *
##    3) Salary>=90800 26  8 Male (0.3076923 0.6923077) *
```

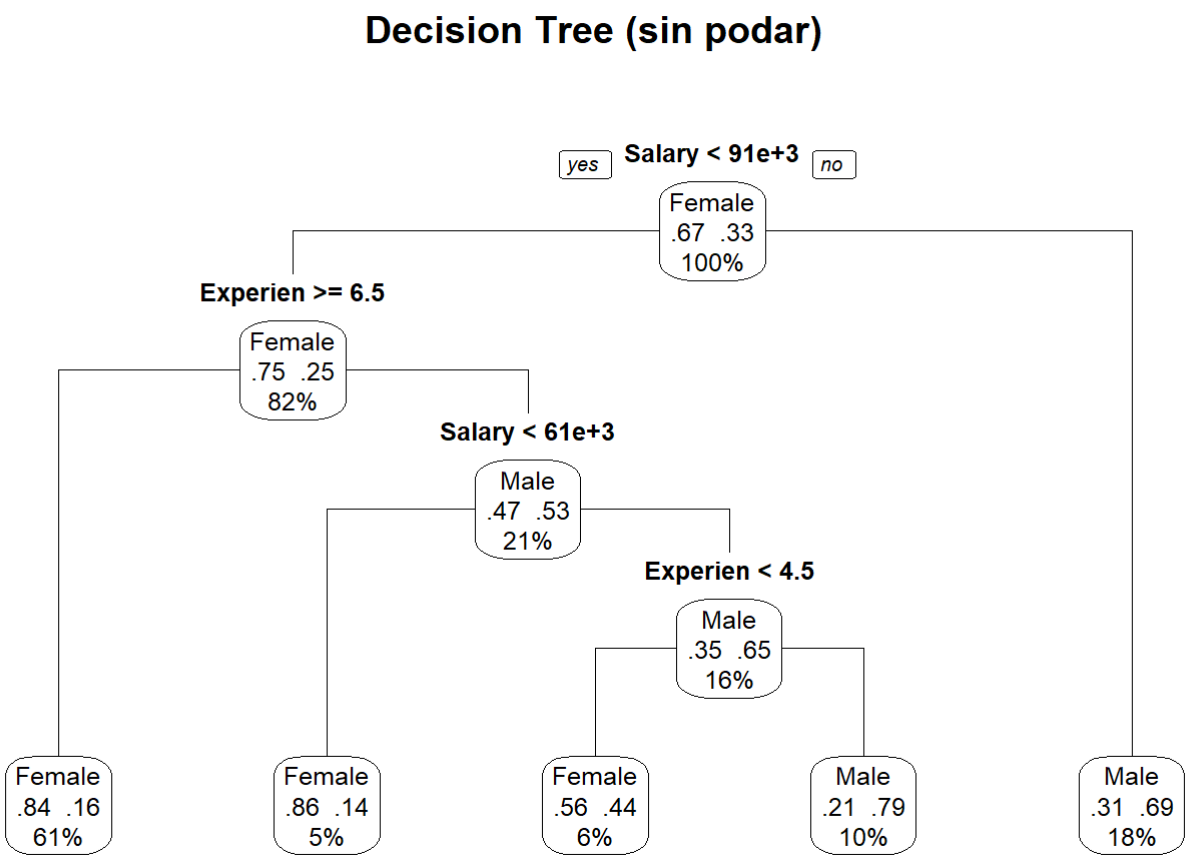
Vamos a interpretar los datos de la siguiente manera:

1.) Se obtiene 145 observaciones donde el 66.89% son mujeres y el 33.11% son hombres

```
1.1) La condición salario<90800, nos indica que es cumplida por 119 observaciones: 74.79% muj
eres y 25.21% son hombres.

  1.1.1) Años de experiencia igual o mayor a 6.5 años, es cumplido por 89 observaciones de la
s que el 84.27% son mujeres y el 15.73% son hombres.

  1.1.2) Años de experiencia menor a 6.5 años, es cumplido por 30 observaciones de las que el
46.67% son mujeres y el 53.33% son hombres.

    1.1.2.1) Dentro de estos últimos: La condición salario<60950, nos indica que es cumplida
 por 7 observaciones: 85.71% mujeres y 14.29% son hombres.

    1.1.2.2) 23 observaciones tienen un salario mayor o igual a 60950, de las que el 34.78% s
on mujeres y 65.22% son hombres.

      1.1.2.2.1) De estas 23 observaciones 9, tienen menos de 4.5 años de experiencia (55.55%
mujeres y 44.45% hombres)

      1.1.2.2.2) 14 observaciones tienen una experiencia igual o superior a 4.5 años, de las
 que el 21.43% son mujeres y el 78.57% son hombres.

1.2) La condición salario>=90800, nos indica que es cumplida por 26 observaciones: 30.77% muj
eres y 69.23% son hombres.
```

5.1- Dibujamos el arbol.

```
prp(arbol, type = 1, extra = 104, fallen.leaves = TRUE, main="Decision Tree (sin podar)")
```



5.2- Complejidad del árbol.

Dibujamos la gráfica con el CP asociado al menor error.

```
plotcp(arbol)
```

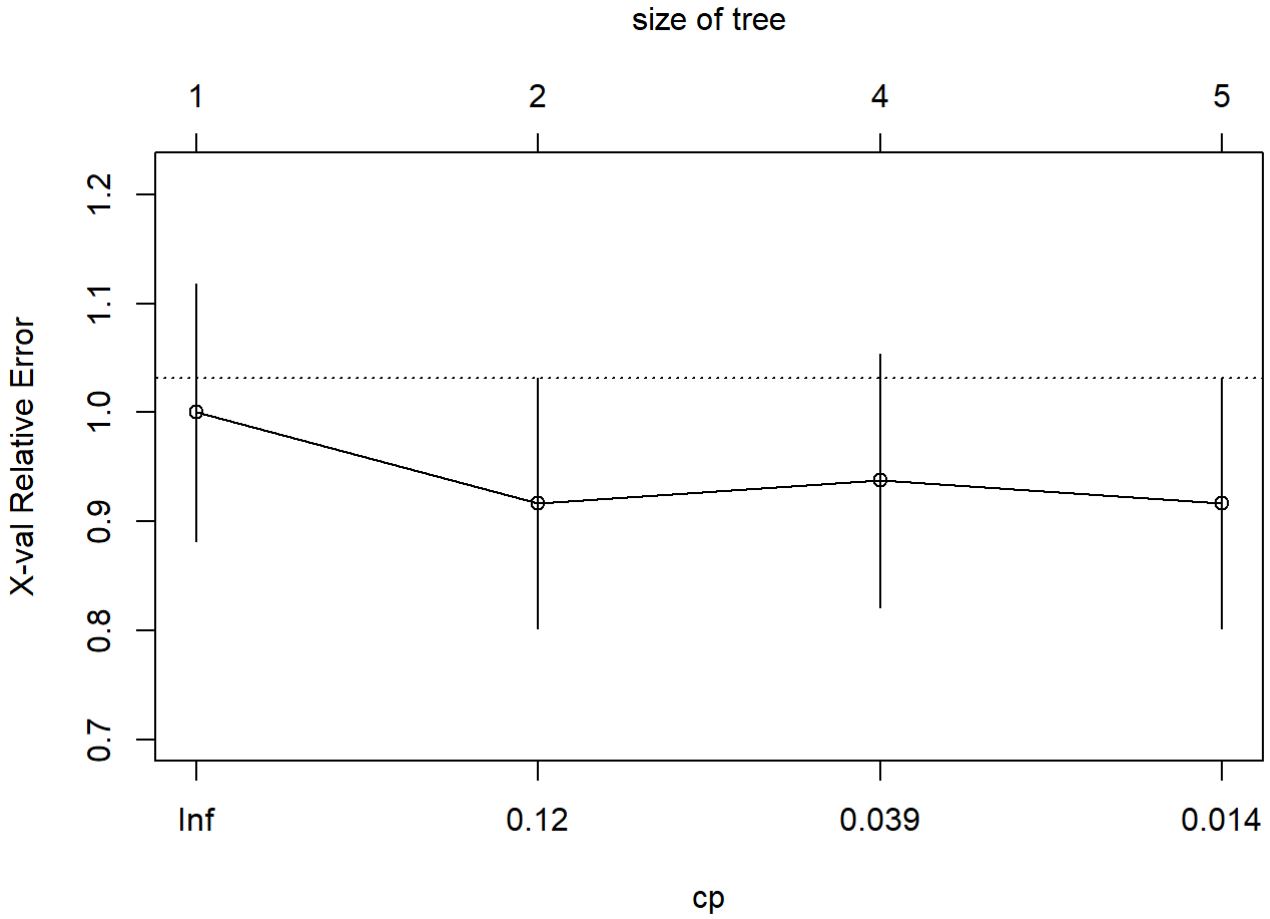


Tabla de complejidades:

```
arbol$cptable
```

```
##           CP nsplit rel error    xerror      xstd
## 1 0.20833333      0 1.0000000 1.0000000 0.1180541
## 2 0.07291667      1 0.7916667 0.9166667 0.1153352
## 3 0.02083333      3 0.6458333 0.9375000 0.1160596
## 4 0.01000000      4 0.6250000 0.9166667 0.1153352
```

La columna xerror indica el momento en el que el árbol deja de descender su error. Se puede observar que existen dos xerror con el mismo valor asociado 0.9166667. Si se escoge el último y penúltimo xerror, se obtendría el mismo árbol que se ha estimado anteriormente con todos los nodos.

6- Podamos el árbol de clasificación.

Por las conclusiones sacadas anteriormente, vamos a podar el árbol por el nodo 2:

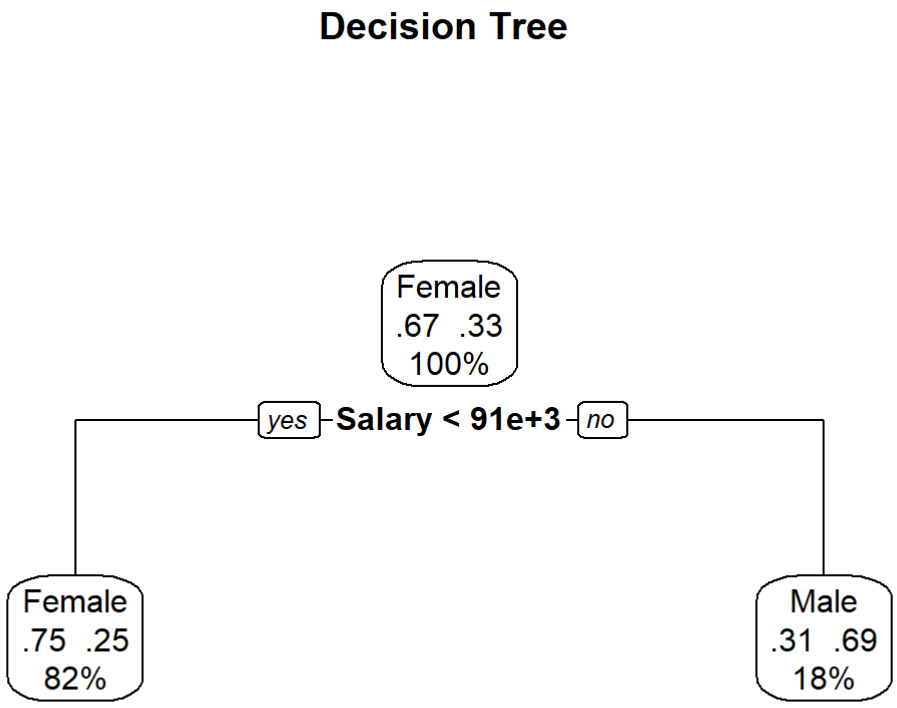
Cogemos por xerror=0.9166667 y cp=0.07291667

```
arbol.podado <- prune(arbol, cp=0.07291667)
arbol.podado
```

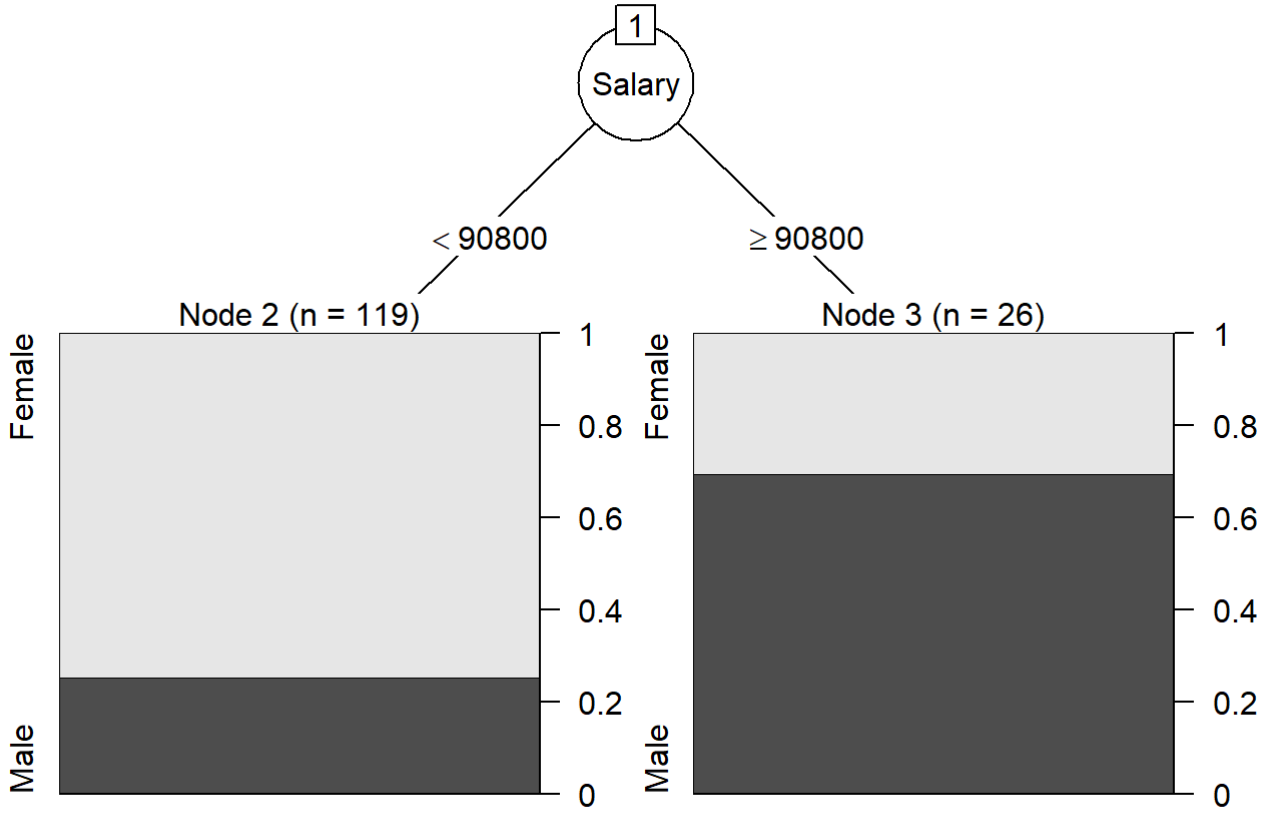
```
## n= 145
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
## 1) root 145 48 Female (0.6689655 0.3310345)
##   2) Salary< 90800 119 30 Female (0.7478992 0.2521008) *
##   3) Salary>=90800 26  8 Male (0.3076923 0.6923077) *
```

6.1- Dibujamos el árbol

```
prp(arbol.podado, type = 2, extra = 104, fallen.leaves = TRUE, main="Decision Tree ")
```




```
#Otra forma de graficarlo.
plot(as.party(arbol.podado))
```



7- Comprobación.

Una vez realizado el árbol con la muestra de entrenamiento, vamos a estimarlo con la muestra de validación para ver si está correctamente estimado.

7.1- Árbol sin podar.

```
arbol.pred <- predict(arbol, gender.validate, type="class")
arbol.perf <- table(gender.validate$Gender, arbol.pred, dnn=c("Actual", "Predicted"))
arbol.perf
```

```
##         Predicted
## Actual   Female Male
##   Female     32   11
##   Male       10   10
```

Con los resultados observamos que el modelo a predecido correctamente que el género era femenino para 32 casos y que el masculino para 10 casos.

Sin embargo, ha predecido mal 21 observaciones. Si en total había 63 observaciones en la muestra de validación, se ha estimado bien el 66.67% de las observaciones. (42/63)*100

7.2- Árbol podado.

```
arbol.pred <- predict(arbol.podado, gender.validate, type="class")
arbol.perf <- table(gender.validate$Gender, arbol.pred, dnn=c("Actual", "Predicted"))
arbol.perf
```

```
##         Predicted
## Actual   Female Male
##   Female     40    3
##   Male       11    9
```

Con los resultados observamos que el modelo podado a predecido correctamente que el género era femenino para 40 casos y que el masculino para 9 casos.

Sin embargo, ha predecido mal 14 observaciones. Si en total había 63 observaciones en la muestra de validación, se ha estimado bien el 77.78% de las observaciones. (49/63)*100.

8- Conclusiones.

El árbol podado, finalmente, observamos como ha hecho una mejor estimación del modelo con un porcentaje de 77.78% frente a un porcentaje de 66.67% del árbol sin podar.

Interpretando el árbol podado, observamos que la mayoría de las personas con un salario superior a 90800 euros son hombres.

Dentro de nuestro DataFrame, observamos que de los 68 hombres, 27 tienen un suelto mayor o igual 90800€ y únicamente 11 mujeres de las 140.

Por tanto, existiendo mayor proporción de mujeres con un salario inferior y además siendo un porcentaje significativamente mayor a los hombres, es consecuente haber concluído estimando éste árbol ya que aunque el 59.7% de los hombres tenga un salario inferior a 90800€, representa un porcentaje pequeño en relación con la muestra total.

En el caso de distinguir entre hombre y mujer, en esta muestra, no afecta de forma significativa la variable experiencia. Además en el árbol sin podar veíamos todos los nodos posibles, donde observabamos que con un menor salario y una menor experiencia, el árbol siempre acababa concluyendo que el sexo era femenino.