Práctica Gender - Álvaro de Prada

Enunciado de la tarea: "La tarea que deben realizar del tema Arboles consistirá en la resolución del arbol Gender Discrimination que tienen en el material de dicho tema. Tendrán que elaborar un script con la programación del arbol y su resolución así como la elaboración de un informe en donde expliciten los resultados principales y el análisis e interpretación exhaustiva desde un punto de vista económico de los resultados obtenidos. Deben elaborar también dentro de dicho informe aquellas conclusiones que les parezcan oportunas a partir del análisis realizado."

1 - Descargamos los datos

```
#Al tratarse de un proyecto nuevo, tenemos establecida como carpeta ruta
la carpeta en la que está el proyecto, en donde hemos guardado los demas
archivos, por lo que no hace falta volver a definir la ruta
getwd()
```

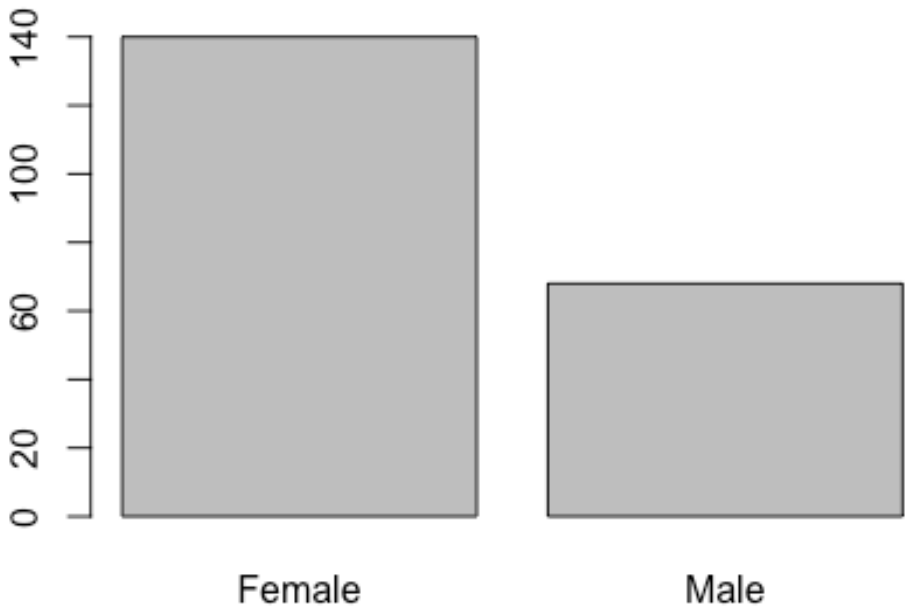
```
## [1] "/Users/alvarodeprada/Documents/CUNEF/Tecnicas de
clasificación/Practicas a entregar/Practica01 Gender"
```

```
gender <-
read.csv("http://www.biz.uiowa.edu/faculty/jledolter/DataMining/GenderDis
crimination.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
```

2 - A continuación dibujamos para ver la proporcion de masculino y femenino antes de proceder a establecer la semilla

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




```
table(gender$Gender)
```

```
##
## Female   Male
##    140     68
```

```
length(gender$Gender[gender$Gender=="Female"])
```

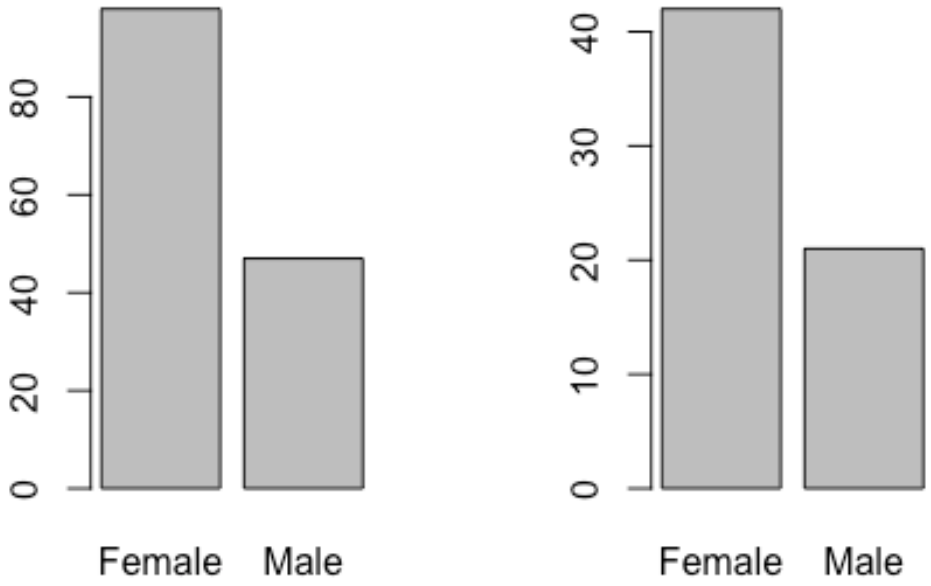
```
## [1] 140
```

```
# Hay 140 female y 68 male, por lo que el 67.3% de la muestra es female
```

3 - Establecemos la semilla aleatoria

```
# Establezco la semilla aleatoria
set.seed(246)
train <- sample(nrow(gender), 0.7*nrow(gender))

# Creo los data frames para la muestra de entrenamiento y validación
df.train <- gender[train,]

df.validate <- gender[-train,]

# Volvemos a hacer la gráfica para comprobar si mantiene la proporción
```

```r
par(mfrow=c(1,2))    #esta línea realiza ambos graficos juntos, lo que nos facilita la comparación visual.

plot(df.train$Gender)
table(df.train$Gender)

##
## Female   Male
##     98     47

# comprobamos que en la muestra hay 98 female y 47 male.

plot(df.validate$Gender)
```




```r
table(df.validate$Gender)

##
## Female   Male
##     42     21

# tenemos un total de 42 female y 21 male.

# Los resultados tanto de la muestra de entrenamiento como de la muestra
```

```
de validación parecen estar bien distribuidos, por lo que trabajaremos
con ellas.
```

4 - Definimos el arbol

```
library(rpart)

# Estimamos el arbol

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

```
## n= 145
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 145 47 Female (0.67586207 0.32413793)
##    2) Salary< 87900 109 23 Female (0.78899083 0.21100917)
##      4) Experience>=5.5 82 11 Female (0.86585366 0.13414634) *
##      5) Experience< 5.5 27 12 Female (0.55555556 0.44444444)
##       10) Experience< 4.5 17  6 Female (0.64705882 0.35294118) *
##       11) Experience>=4.5 10  4 Male (0.40000000 0.60000000) *
##    3) Salary>=87900 36 12 Male (0.33333333 0.66666667)
##      6) Salary< 109300 24 11 Male (0.45833333 0.54166667)
##       12) Experience>=10.5 12  3 Female (0.75000000 0.25000000) *
##       13) Experience< 10.5 12  2 Male (0.16666667 0.83333333) *
##      7) Salary>=109300 12  1 Male (0.08333333 0.91666667) *
```

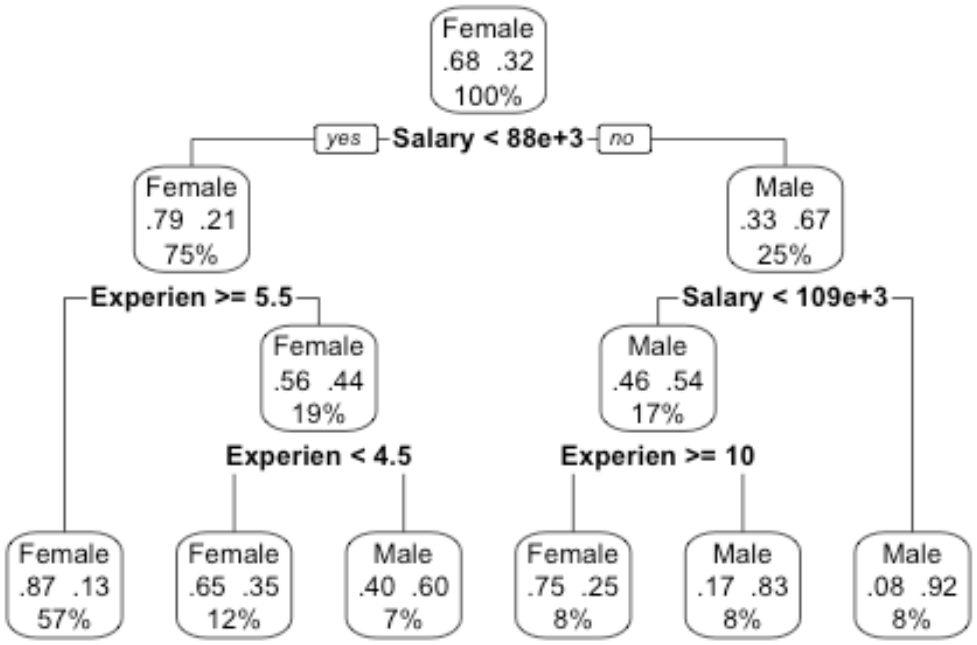
```
summary(arbol)
```

```
## Call:
## rpart(formula = Gender ~ ., data = df.train, method = "class",
##     parms = list(split = "information"))
##   n= 145
##
##           CP nsplit rel error    xerror      xstd
## 1 0.25531915      0 1.0000000 1.0000000 0.1199168
## 2 0.06382979      1 0.7446809 0.8085106 0.1126683
## 3 0.02127660      3 0.6170213 0.7446809 0.1096348
## 4 0.01000000      5 0.5744681 0.8297872 0.1136066
##
## Variable importance
##     Salary Experience
##         57         43
##
## Node number 1: 145 observations,    complexity param=0.2553191
##   predicted class=Female  expected loss=0.3241379  P(node) =1
##     class counts:    98    47
##    probabilities: 0.676 0.324
```

```
##   left son=2 (109 obs) right son=3 (36 obs)
##   Primary splits:
##       Salary     < 87900  to the left,  improve=12.26145, (0 missing)
##       Experience < 22.5   to the left,  improve= 1.86282, (0 missing)
##   Surrogate splits:
##       Experience < 22.5   to the left,  agree=0.779, adj=0.111, (0
split)
##
## Node number 2: 109 observations,    complexity param=0.0212766
##   predicted class=Female  expected loss=0.2110092  P(node) =0.7517241
##     class counts:    86    23
##    probabilities: 0.789 0.211
##   left son=4 (82 obs) right son=5 (27 obs)
##   Primary splits:
##       Experience < 5.5    to the right, improve=5.2948630, (0 missing)
##       Salary     < 79200  to the right, improve=0.4989729, (0 missing)
##
## Node number 3: 36 observations,    complexity param=0.06382979
##   predicted class=Male    expected loss=0.3333333  P(node) =0.2482759
##     class counts:    12    24
##    probabilities: 0.333 0.667
##   left son=6 (24 obs) right son=7 (12 obs)
##   Primary splits:
##       Salary     < 109300 to the left,  improve=2.920376, (0 missing)
##       Experience < 10.5   to the right, improve=1.207137, (0 missing)
##   Surrogate splits:
##       Experience < 18.5   to the left,  agree=0.778, adj=0.333, (0
split)
##
## Node number 4: 82 observations
##   predicted class=Female  expected loss=0.1341463  P(node) =0.5655172
##     class counts:    71    11
##    probabilities: 0.866 0.134
##
## Node number 5: 27 observations,    complexity param=0.0212766
##   predicted class=Female  expected loss=0.4444444  P(node) =0.1862069
##     class counts:    15    12
##    probabilities: 0.556 0.444
##   left son=10 (17 obs) right son=11 (10 obs)
##   Primary splits:
##       Experience < 4.5    to the left,  improve=0.7806239, (0 missing)
##       Salary     < 70500  to the left,  improve=0.7806239, (0 missing)
##   Surrogate splits:
##       Salary < 68600  to the left,  agree=0.778, adj=0.4, (0 split)
##
## Node number 6: 24 observations,    complexity param=0.06382979
##   predicted class=Male    expected loss=0.4583333  P(node) =0.1655172
##     class counts:    11    13
##    probabilities: 0.458 0.542
##   left son=12 (12 obs) right son=13 (12 obs)
```

```
##   Primary splits:
##       Experience < 10.5   to the right, improve=4.3973460, (0 missing)
##       Salary     < 94500  to the right, improve=0.6745846, (0 missing)
##   Surrogate splits:
##       Salary < 89300  to the right, agree=0.667, adj=0.333, (0 split)
##
## Node number 7: 12 observations
##   predicted class=Male    expected loss=0.08333333  P(node)
=0.08275862
##     class counts:     1    11
##    probabilities: 0.083 0.917
##
## Node number 10: 17 observations
##   predicted class=Female  expected loss=0.3529412  P(node) =0.1172414
##     class counts:    11     6
##    probabilities: 0.647 0.353
##
## Node number 11: 10 observations
##   predicted class=Male    expected loss=0.4  P(node) =0.06896552
##     class counts:     4     6
##    probabilities: 0.400 0.600
##
## Node number 12: 12 observations
##   predicted class=Female  expected loss=0.25  P(node) =0.08275862
##     class counts:     9     3
##    probabilities: 0.750 0.250
##
## Node number 13: 12 observations
##   predicted class=Male    expected loss=0.1666667  P(node) =0.08275862
##     class counts:     2    10
##    probabilities: 0.167 0.833

#dibujamos el arbol sin podar
library(rpart.plot)
prp(arbol, type=2, extra=104, fallen.leaves= TRUE, main="arbol general")
```

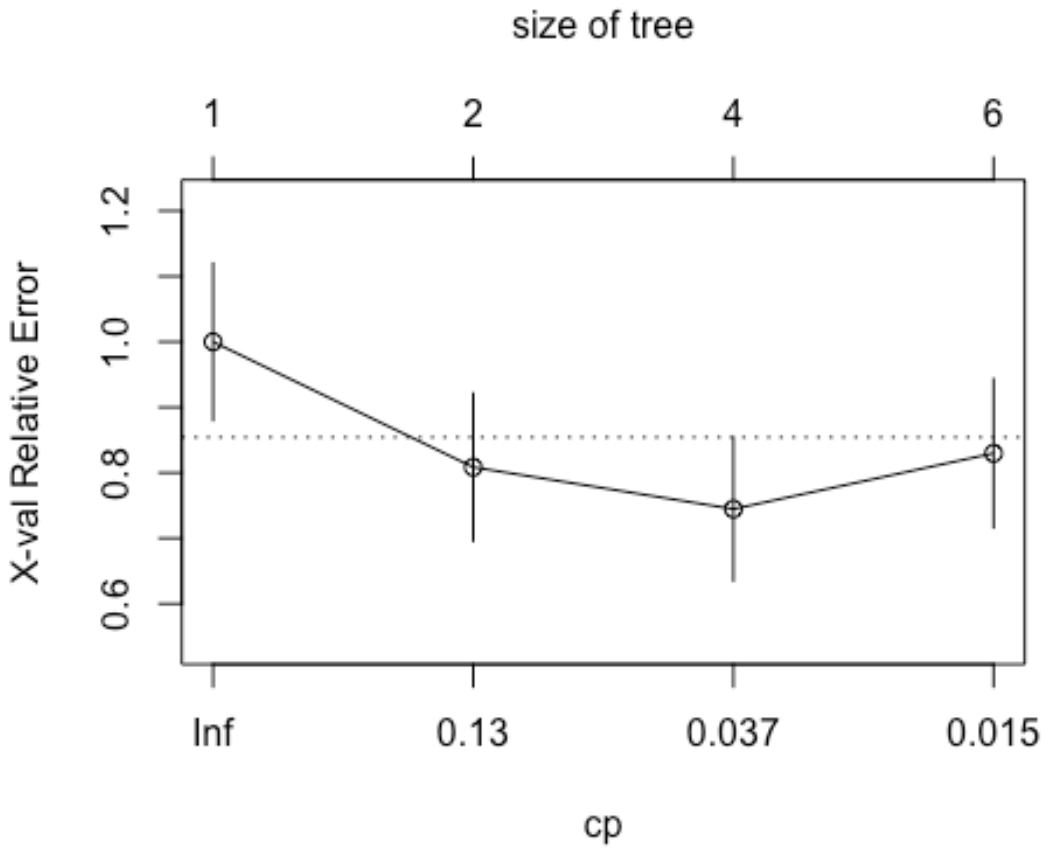
arbol general




```
#miramos el cp y escogemos el punto de corte donde tenga el menor error
que en este caso es el terero.

arbol$cptable
```

```
##           CP nsplit rel error    xerror      xstd
## 1 0.25531915      0 1.0000000 1.0000000 0.1199168
## 2 0.06382979      1 0.7446809 0.8085106 0.1126683
## 3 0.02127660      3 0.6170213 0.7446809 0.1096348
## 4 0.01000000      5 0.5744681 0.8297872 0.1136066
```

```
# Representamos gráficamente la curva cp

plotcp(arbol)
```



5 -

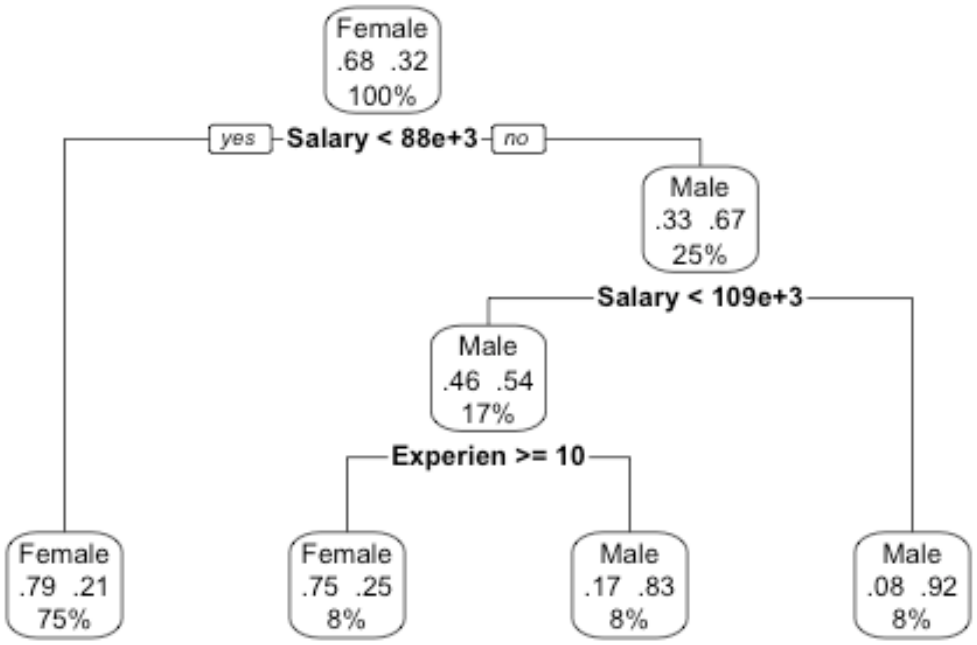
A continuación 'podamos` el arbol

```
arbol.podado <- prune(arbol, cp= 0.02127660 )

print(arbol.podado)
```

```
## n= 145
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 145 47 Female (0.67586207 0.32413793)
##    2) Salary< 87900 109 23 Female (0.78899083 0.21100917) *
##    3) Salary>=87900 36 12 Male (0.33333333 0.66666667)
##      6) Salary< 109300 24 11 Male (0.45833333 0.54166667)
##       12) Experience>=10.5 12  3 Female (0.75000000 0.25000000) *
##       13) Experience< 10.5 12  2 Male (0.16666667 0.83333333) *
##      7) Salary>=109300 12  1 Male (0.08333333 0.91666667) *
```

```
# y volvemos a pintar el arbol una vez podado
prp(arbol.podado, type = 2, extra = 104,
    fallen.leaves = TRUE, main="Arbol final")
```

Arbol final



6 -

Realizamos la predicción tanto del arbol podado como del arbol sin podar, y las comparamos

```
# predicción con la muestra de validacion
arbol.pred <- predict(arbol.podado, df.validate, type="class")

arbol.perf <- table(df.validate$Gender, arbol.pred,
                    dnn=c("Actual", "Predicted"))

arbol.perf
```

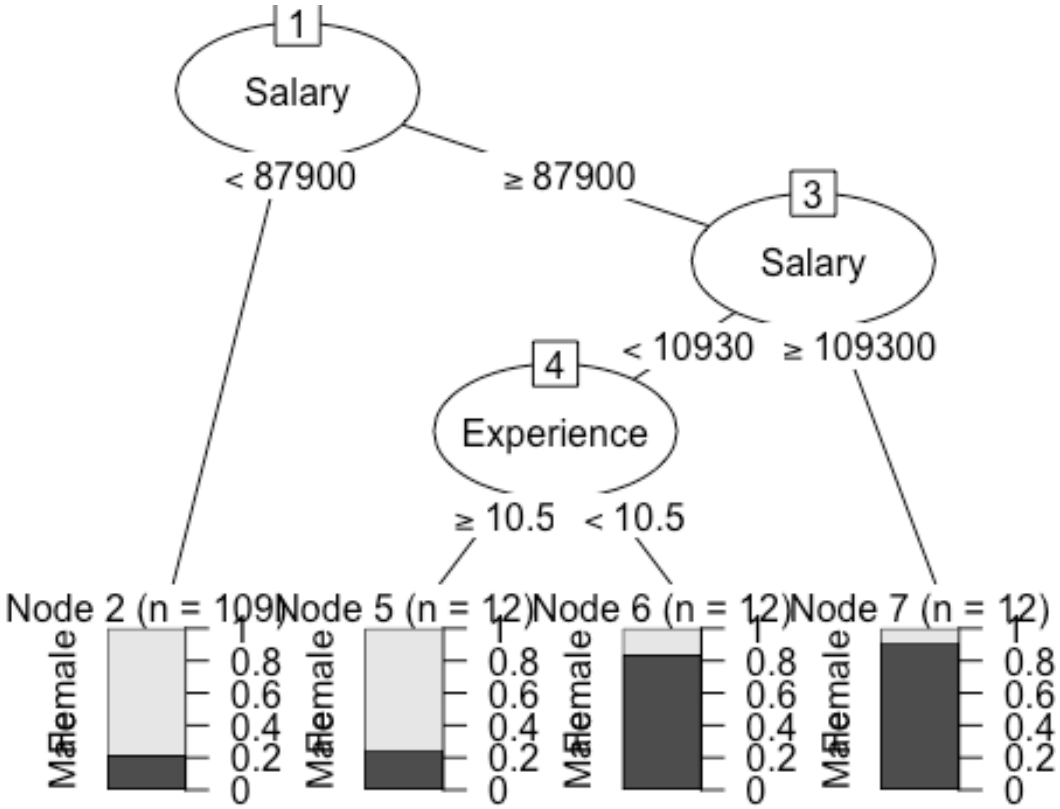
```
##         Predicted
## Actual   Female Male
##   Female     40    2
##   Male       15    6
```

```
#vamos a comparar que hubiera pasasdo si no lo hubiéramos podado:
arbol.pred1 <- predict(arbol, df.validate, type="class")

arbol.perf1 <- table(df.validate$Gender, arbol.pred1,
                    dnn=c("Actual", "Predicted"))

arbol.perf1
```

```
##          Predicted
## Actual   Female Male
##   Female     38    4
##   Male       14    7

# En conclusion vemos que nuestro arbol podado sale igual que si no lo
podamos, es decir, hemos ahorrado en complejidad
```

-CONCLUSIONES-

```
#lo representamos en un grafico diferente.
library(partykit)

## Loading required package: grid

plot(as.party(arbol.podado))
```



Conclusiones: Cuanto más homogéneo sea el color del nodo mejor será el resultado. A la vista de los resultados: -En el primer nodo se anaizan los salarios menores de 87.900, estableciéndose una probabilidad mayor para las mujeres que para los hombres, por lo que las mujeres tienen mayor probabilidad que los hobres de estar en el rango salarial mas bajo. - En el segungundo nodo, se analizan los salarios comprendidos entre 87.900 y 109.300, con una experiencia MAYOR a 10.5 años, dándose nuevamente mayor probabilidad de este suceso a las mujeres que a los

hombres. - En el tercer nodo se analizan los salarios comprendidos entre 87.900 y 109.300, con una experiencia MENOR a 10.5 años, dándose más probabilidad de este suceso a los hombres que a las mujeres. En el último nodo se analizan los salarios superiores a 109.300 independientemente de la experiencia, Estableciendose superior la priobabilidad en hombres que en mujeres.

Por lo tanto se concluye que existe una diferencia salarial entre hombres y mujeres en la que las mujeres tienden a ganar menos dinero aun teniendo más experiencia que los hombres.