PRA2 - Tipología y ciclo de vida de los datos: Limpieza y análisis de datos

Sergio Funes, David Ortiz

Junio de 2021

Contents

1. Descripción del dataset y problema de interés

El conjunto de datos es el 'Dataset recruitment data', disponible en el repositorio en Kaggle. Está formado por 11 características para un total de 614 personas, en un conjunto de datos que refleja el estado de contratación. Este conjunto de datos es interesante puesto que permite identificar las relaciones más directas con un proceso de contratación satisfactorio, lo que puede permitir actuar para mejorar las posibilidades de contratación, o simplemente para detectar relaciones entre distintos usuarios tipo y variables tan relevantes como el salario.

Las características de las que dispone el conjunto de datos son las siguientes:

- **Número de serie**. Número del registro.
- **Género**. Si el sujeto es hombre o mujer.
- **Experiencia con Python**. Si el sujeto tiene experiencia de programación en Python.
- **Años de experiencia**. Número de años de experiencia del sujeto.
- **Educación**. Si el sujeto es graduado universitario o no.
- **Prácticas**. Si el sujeto ha hecho unas prácticas o no.
- **Puntuación**. Puntuación del sujeto.
- **Salario** (*10e4). Salario del sujeto, en decenas de miles de rupias.
- **Histórico de ofertas**. Si el sujeto ha tenido ofertas laborales.
- **Localización**. Tipo de localización del sujeto en relación con la ciudad: urbana, semiurbana o rural.
- **Estado de contratación**. Si el sujeto fue contratado o no.

A partir de este conjunto de datos, se pretende determinar que variables influyen más a la hora de decidir si un candidato es contratado o no. Además, se podrán crear modelos predictivos para predecir la decisión final.

Este análisis adquiere importancia a la hora de reclutar nuevos empleados en una empresa principalmente de informática, ya que a partir de su experiencia de programación, su educación y su experiencia, tendremos una primera aproximación sobre si un empleado será contratado o no.

Asímismo, durante la realización de esta práctica nos proponemos dar respuesta a tres preguntas principales:

1. Si hay correlación entre la experiencia del usuario y la puntuación otorgada a cada uno de ellos y el salario que consigue.
2. Si las mujeres cobran distinto a los hombres y si, en caso de hacerlo, su salario es inferior al de estos.
3. Si hay factores de protección o riesgo importantes que afecten a la contratación y en qué grado.

2. Integración y selección de los datos de interés

Realizamos una lectura de los datos:

```
data <- read.csv("recruitment_decision_tree.csv", header = TRUE, stringsAsFactors = TRUE, na.strings =
head(data)
```

```
##   Serial_no Gender Python_exp Experience_Years    Education Internship Score
## 1         1   Male        Yes                0     Graduate         No  5139
## 2         2   Male         No                1     Graduate         No  4583
## 3         3   Male         No                0     Graduate        Yes  3000
## 4         4   Male         No                0 Not Graduate         No  2583
## 5         5   Male        Yes                0     Graduate         No  6000
## 6         6   Male         No                2     Graduate        Yes  5417
##   Salary...10E4 Offer_History Location Recruitment_Status
## 1             0             1    Urban                  Y
## 2           128             1    Rural                  N
## 3            66             1    Urban                  Y
```

```
## 4           120             1    Urban                  Y
## 5           141             1    Urban                  Y
## 6           267             1    Urban                  Y
```

```
sapply(data, function(x) class(x))
```

```
##          Serial_no             Gender         Python_exp   Experience_Years
##          "integer"           "factor"           "factor"          "integer"
##          Education         Internship              Score      Salary...10E4
##           "factor"           "factor"          "integer"          "integer"
##      Offer_History           Location Recruitment_Status
##          "integer"           "factor"           "factor"
```

Observamos que los tipos de datos asignados son los correctos. A continuación, podemos prescindir de la variable Serial_no, ya que no nos aporta información sobre el candidato, únicamente representa que número de candidato es.

```
data <- data[, -(1)]
```

3. Limpieza de los datos

3.1. Ceros y elementos vacíos

Comprobamos si nuestro conjunto de datos contiene elementos vacíos:

```
sapply(data, function(x) sum(is.na(x)))
```

```
##             Gender         Python_exp   Experience_Years          Education
##                 13                  3                 15                  0
##         Internship              Score      Salary...10E4      Offer_History
##                 32                  0                 21                 50
##           Location Recruitment_Status
##                  0                  0
```

Podemos observar que hay 134 elementos vacíos entre las variables Gender, Python_exp, Experience_Years, Internship, Salary...10E4 y Offer_History. Procederemos a emplear un método de imputación de valores basada en k vecinos más próximos. Utilizaremos la función kNN de la libreria VIM:

```
suppressWarnings(suppressMessages(library(VIM)))

data$Gender <- kNN(data)$Gender
data$Python_exp <- kNN(data)$Python_exp
data$Experience_Years <- kNN(data)$Experience_Years
data$Internship <- kNN(data)$Internship
data$Salary...10E4 <- kNN(data)$Salary...10E4
data$Offer_History <- kNN(data)$Offer_History

sapply(data, function(x) sum(is.na(x)))
```

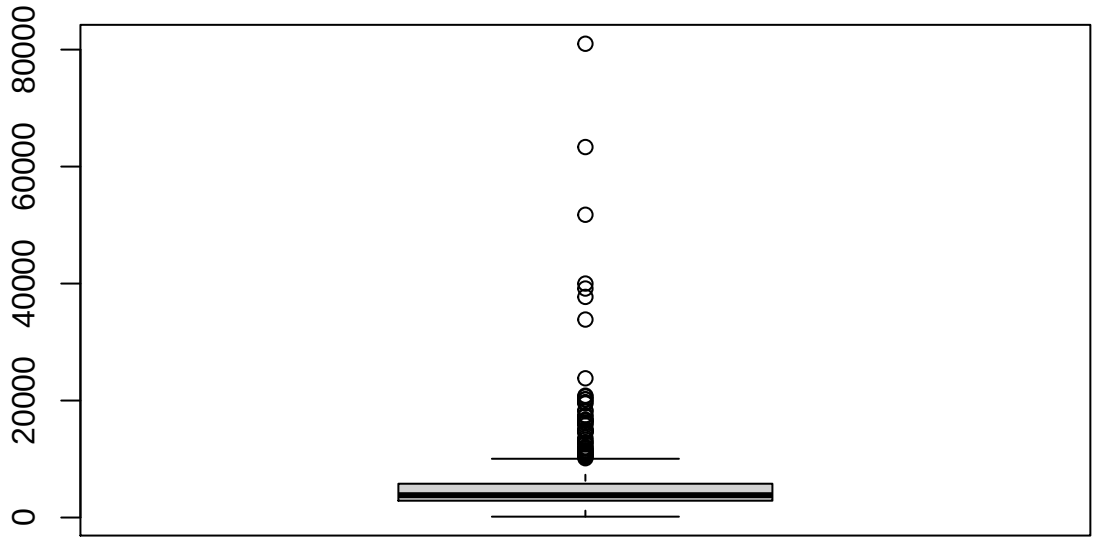
```
##             Gender         Python_exp   Experience_Years          Education
##                  0                  0                  0                  0
##         Internship              Score      Salary...10E4      Offer_History
##                  0                  0                  0                  0
##           Location Recruitment_Status
##                  0                  0
```

Después de la imputación de valores, podemos comprobar que ya no existen valores vacíos en nuestro conjunto de datos.

3.2. Valores extremos

Identificaremos los valores outliers de dos formas diferentes, utilizando un diagrama de caja y utilizando la función boxplots.stats() de R. Representaremos los datos de las variables numéricas Score y Salary...10E4. Primeramente representaremos la variable Score:

```
boxplot(data$Score)
```

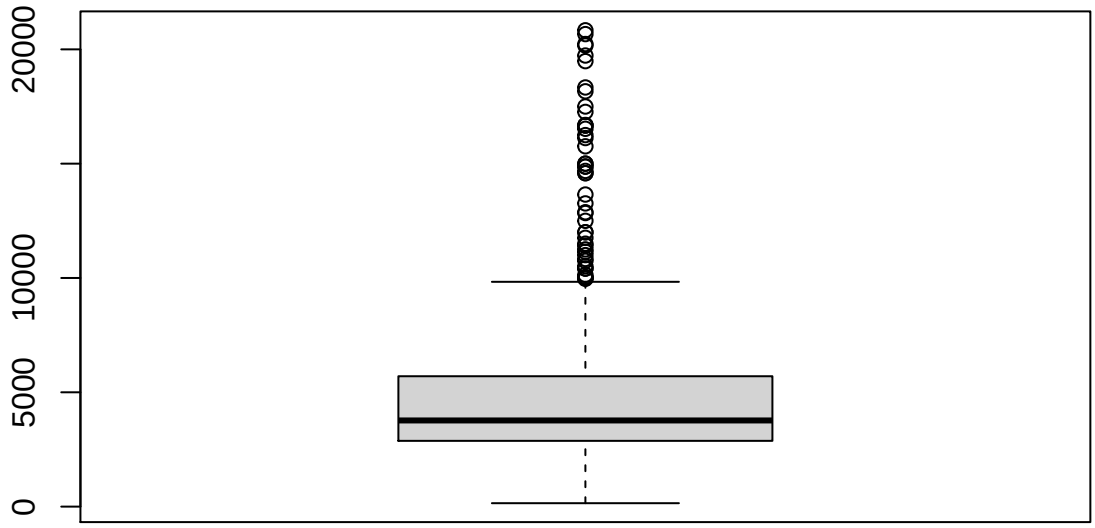



```
boxplot.stats(data$Score)$out
```

```
##  [1] 12841 12500 11500 10750 13650 11417 14583 10408 23803 10513 20166 14999
## [13] 11757 14866 39999 51763 33846 39147 12000 11000 16250 14683 11146 14583
## [25] 20667 20233 15000 63337 19730 15759 81000 14880 12876 10416 37719 16692
## [37] 16525 16667 10833 18333 17263 20833 13262 17500 11250 18165 10139 19484
## [49] 16666 16120 12000
```

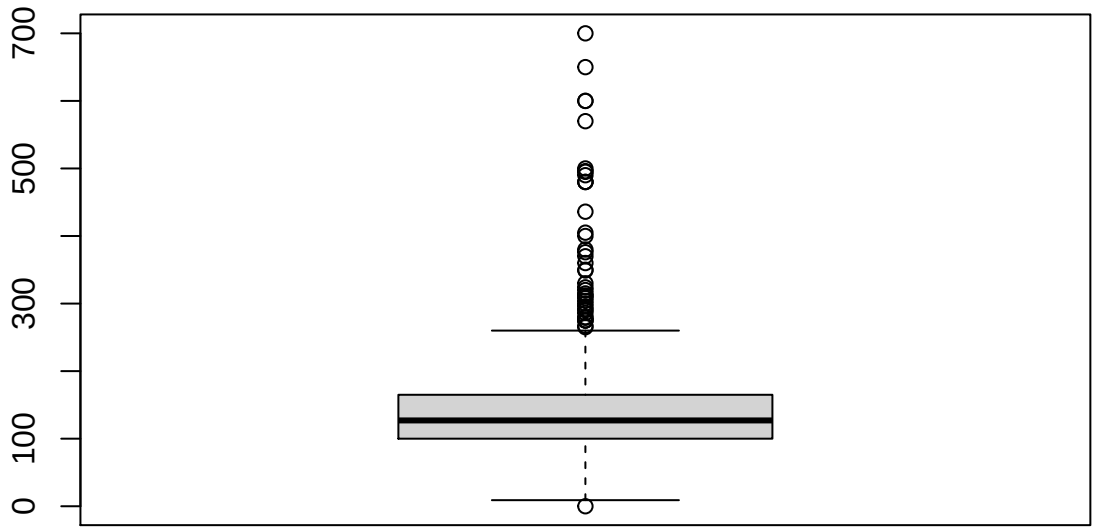
Podemos observar que la variable Score contiene muchos outliers. En este caso, podemos deshacernos de los valores más lejanos, ya que aunque haya valores fuera del rango, no podemos asumir que sean outliers y no personas altamente cualificadas para ser contratadas. Para deshacernos de los outliers, le daremos valor vacío a partir de cierto rango y después utilizaremos la función kNN nuevamente para la imputación de los valores:

```
data$Score[data$Score > 23000] <- NA

data$Score <- kNN(data)$Score

boxplot(data$Score)
```



Puede observarse que han desaparecido los outliers más lejanos. Continuaremos con la variable Salary...10E4:

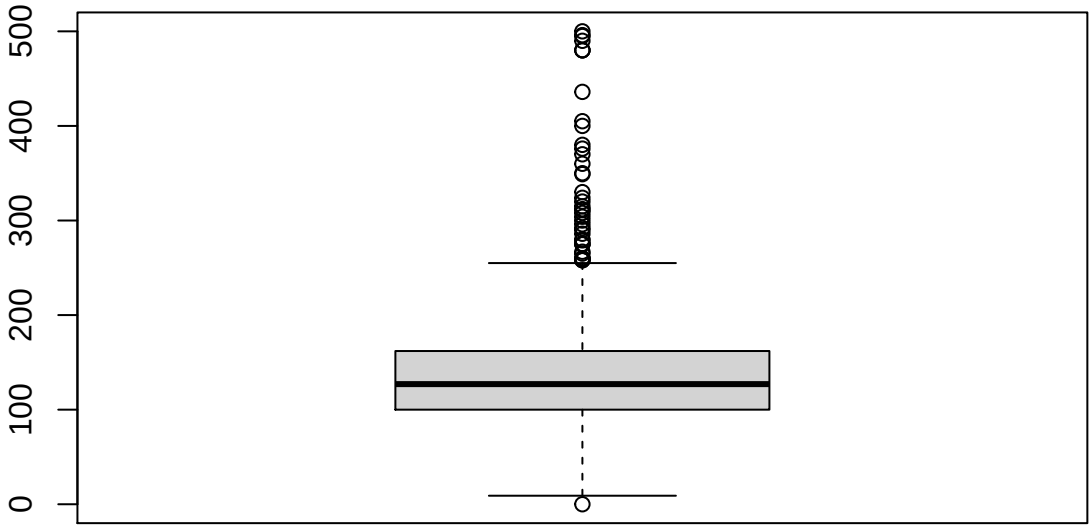
```
boxplot(data$Salary...10E4)
```




```
boxplot.stats(data$Salary...10E4)$out
```

```
##  [1]   0 267 349 315 320 286 312 265 370 650 290 600 275 700 495 280 279 304 330
## [20] 436 480 300 376 490 308 570 380 296 275 360 405 500 480 311 480 400 324 600
## [39] 275 292 350 496
```

```
data$Salary...10E4[data$Salary...10E4 > 500] <- NA

data$Salary...10E4 <- kNN(data)$Salary...10E4

boxplot(data$Salary...10E4)
```



En la variable Salary...10E4, hemos realizado el mismo proceso que en la variable anterior.

3.3. Exportación de los datos preprocesados

Después de leer y validar el conjunto de datos, limpieza de los elementos vacíos y extremos, procedemos a guardar los datos en un fichero denominado "recruitment_decision_tree_clean.csv":

```
write.csv(data, "recruitment_decision_tree_clean.csv")
```

4. Análisis de los datos

4.1. Selección de datos

Pasamos a la selección de las variables de interés para su análisis estadístico posterior. Encontramos aquí las variables categóricas más relevantes.

```
# Per gender
data.women <- data[data$Gender == "Female", ]
data.men <- data[data$Gender == "Male", ]

# Per location type
data.urban <- data[data$Location == "Urban", ]
data.semiurban <- data[data$Location == "Semiurban", ]
```

```r
data.rural <- data[data$Location == "Rural", ]

# Per education level
data.graduated <- data[data$Education == "Graduate", ]
data.not_graduated <- data[data$Education == "Not Graduate", ]

# Target var: recruitment status
data.recruited <- data[data$Recruitment_Status == "Y", ]
data.not_recruited <- data[data$Recruitment_Status == "N", ]
```

4.2. Comprobación de normalidad y homogeneidad de la varianza

Se comprueba ahora la normalidad de las variables cuantitativas, mediante el test de normalidad de Anderson-Darling. Si el p-valor de es superior al nivel de significancia de $\alpha = 0,05$, entonces podemos concluir que la variable en cuestión es normal.

```r
library(nortest)

num_vars <- c("Experience_Years", "Score", "Salary...10E4")
alpha <- 0.05

for (col in num_vars){
  p_value = ad.test(data[, col])$p.value
  if (p_value > alpha){
    print(paste0(col, ": Normal (p = ", p_value, ")"))
  } else {
    print(paste0(col, ": Not normal (p = ", p_value, ")"))
  }
}
```

```
## [1] "Experience_Years: Not normal (p = 3.7e-24)"
## [1] "Score: Not normal (p = 3.7e-24)"
## [1] "Salary...10E4: Not normal (p = 3.7e-24)"
```

Por tanto vemos que las tres variables numéricas siguen una distribución no normal.

Para estudiar la homogeneidad de las varianzas aplicamos el test de Fligner-Killeen. Se pretende comprobar esta en la relación del género de los candidatos con el salario asociado a estos. Para este test, la hipótesis nula, H_0, sostiene la igualdad de varianzas para ambos grupos. Por su parte, H_1 supondría varianzas diferentes.

Si el p-valor es superior a un nivel de significancia del 0,05, se aceptará dicha hipótesis nula y por tanto la igualdad de varianzas. Aplicamos el test sobre los salarios, una de las variables de interés a estudiar.

```r
# Salary
fligner.test(Salary...10E4 ~ Gender, data=data)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Salary...10E4 by Gender
## Fligner-Killeen:med chi-squared = 3.6005, df = 1, p-value = 0.05776
```

Puesto que el p-valor es mayor a 0,05, se acepta la hipótesis nula y por tanto se asume la igualdad de varianzas para ambas muestras.

4.3. Pruebas estadísticas

Realizamos ahora varios análisis estadísticos que nos permitan responder las preguntas de interés planteadas sobre el conjunto de datos.

4.3.1. Correlación de la experiencia y puntuación de los candidatos con el salario

Comprobamos la correlación de la experiencia y puntuación de los candidatos con el salario, con el fin de comprobar en qué medida están asociadas estas métricas con una mayor retribución económica.

```
# Experience vs salary
corr_exp_salary = cor(data$Experience_Years, data$Salary...10E4)
print(paste("Correlation of experience and salary:", round(corr_exp_salary, 3)))
```

```
## [1] "Correlation of experience and salary: 0.142"
```

```
# Score vs salary
corr_score_salary = cor(data$Score, data$Salary...10E4)
print(paste("Correlation of score and salary:", round(corr_score_salary, 3)))
```

```
## [1] "Correlation of score and salary: 0.495"
```

Vemos que la correlación entre la experiencia y el salario es más bien débil para las muestras recogidas en el conjunto de datos, con un valor de 0.142. No obstante, la puntuación tiene una correlación superior, aunque todavía moderada, de 0.495.

4.3.2. Contraste de hipótesis: salario de mujeres y hombres

Se pretende comprobar además si el salario de las mujeres y el de los hombres es diferente. Para ello, realizamos un contraste de hipótesis bilateral definiendo:

- $H_0 : \mu_{mujeres} = \mu_{hombres}$
- $H_1 : \mu_{mujeres} \neq \mu_{hombres}$

Donde la hipótesis nula, H_0, corresponde a la igualdad de salarios, y la hipótesis alternativa H_1, corresponde a la desigualdad de estos. Si bien la distribución no es normal, contamos con un buen tamaño muestral muy superior a 30, por lo que el teorema del límite central nos permite realizar test estadístico sobre la media asumiendo la normalidad de esta.

```
t.test(data.women$Salary...10E4, data.men$Salary...10E4,
       alternative="two.sided", conf.level=0.95)
```

```
##
##  Welch Two Sample t-test
##
## data:  data.women$Salary...10E4 and data.men$Salary...10E4
## t = -3.4887, df = 182.19, p-value = 0.0006086
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  -37.63997 -10.44483
## sample estimates:
## mean of x mean of y
##  121.6429  145.6853
```

Al obtener un `p-valor` inferior a 0.05, el valor de significancia para un nivel de confianza del 95%, rechazamos la hipótesis nula. Por tanto, podemos afirmar que los salarios de mujeres y hombres son diferentes.

Para concretar más, podemos aplicar una hipótesis unilateral que verifique si efectivamente y como se puede sospechar, los sueldos de las mujeres son inferiores a los de los hombres. Para ello, definimos ahora las hipótesis:

- $H_0 : \mu_{mujeres} = \mu_{hombres}$
- $H_1 : \mu_{mujeres} < \mu_{hombres}$

En este caso la hipótesis alternativa, H_1, corresponde al supuesto en el que las mujeres tienen un salario medio inferior al de los hombres. Realizamos el test estadístico.

```
t.test(data.women$Salary...10E4, data.men$Salary...10E4,
       alternative="less", conf.level=0.95)
```

```
##
##  Welch Two Sample t-test
##
## data:  data.women$Salary...10E4 and data.men$Salary...10E4
## t = -3.4887, df = 182.19, p-value = 0.0003043
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -12.64883
## sample estimates:
## mean of x mean of y
##  121.6429  145.6853
```

En este caso vemos, de nuevo, como el `p-valor` es inferior al nivel de significancia adoptado, por lo que se rechaza la hipótesis nula y se concluye que, la media de los salarios de las mujeres es inferior al de los hombres con una confianza del 95%.

4.3.3. Regresión logística

Puesto que buena parte de las variables con las que contamos son de tipo categórico, podemos comprobar la relación entre estas y si finalmente el candidato fue contratado o no.

Podemos comprobar si existe relación entre algunas variables y la variable objetivo mediante el test χ^2 de Pearson.

```
# Get relation for all variables
data.colnames <- colnames(data[, -(10)])
alpha <- 0.05
print("Checking against Recruitment_Status:")
```

```
## [1] "Checking against Recruitment_Status:"
```

```
for (col in data.colnames){
  p_value <- chisq.test(table(data$Recruitment_Status, data[, col]),
                        simulate.p.value=TRUE)$p.value
  p_value <- round(p_value, 4)
  if (p_value > 0.05){
    print(paste0(col, ": not related (p = ", p_value, ")"))
  } else {
    print(paste0(col, ": related (p = ", p_value, ")"))
  }
}
```

```
## [1] "Gender: not related (p = 0.7351)"
## [1] "Python_exp: related (p = 0.023)"
## [1] "Experience_Years: not related (p = 0.2919)"
## [1] "Education: related (p = 0.04)"
## [1] "Internship: not related (p = 1)"
## [1] "Score: not related (p = 0.2279)"
## [1] "Salary...10E4: not related (p = 0.2199)"
## [1] "Offer_History: related (p = 5e-04)"
## [1] "Location: related (p = 0.002)"
```

Vemos que las variables que mantienen relación con la contratación son, según dicho test la educación, el historial de ofertas y la localización. Asimismo, vemos que la experiencia con Python es marginalmente relevante, por lo que la añadimos por la información que pueda aportar al modelo. Reajustamos las referencias de estas variables, con el estado más bajo (no graduado, sin ofertas previas, localización rural y sin experiencia en Python) como tal.

```
# Relevel related variables
data$Education_R <- relevel(data$Education, ref="Not Graduate")
data$Offer_History_R <- relevel(as.factor(data$Offer_History), ref="0")
data$Location_R <- relevel(data$Location, ref="Rural")
data$Python_exp_R <- relevel(data$Python_exp, ref="No")

# Logistic regression
logistic_formula <- Recruitment_Status ~ Education_R + Offer_History_R + Location_R + Python_exp_R
logistic_model <- glm(formula=logistic_formula, data=data,
                      family=binomial(link = 'logit'))

summary(logistic_model)
```

```
##
## Call:
## glm(formula = logistic_formula, family = binomial(link = "logit"),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1475  -0.3553   0.5221   0.6998   2.5802
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -3.0082     0.4730  -6.360 2.02e-10 ***
```

```
## Education_RGraduate   0.2771     0.2548   1.088  0.27668
## Offer_History_R1      4.0132     0.4164   9.639  < 2e-16 ***
## Location_RSemiurban   0.9188     0.2717   3.382  0.00072 ***
## Location_RUrban       0.2146     0.2572   0.834  0.40408
## Python_exp_RYes      -0.5613     0.2196  -2.555  0.01061 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 762.89  on 613  degrees of freedom
## Residual deviance: 548.91  on 608  degrees of freedom
## AIC: 560.91
##
## Number of Fisher Scoring iterations: 5
```

Sobre las variables estudiadas, se observa una gran significancia para el historial de ofertas laborales registrado, así como la localización, en el caso de que esta esté calificada como semiurbana. Por último, vemos una significancia menor, aunque no despreciable, de la experiencia de programación en Python. Podemos calcular las *Odds-Ratio* (OR), para comprobar cuál es el efecto de estas tres variables:

```
## Odds-Ratio
exp(coefficients(logistic_model))
```

```
##         (Intercept) Education_RGraduate    Offer_History_R1 Location_RSemiurban
##          0.04938198          1.31935960         55.32563063          2.50626013
##     Location_RUrban     Python_exp_RYes
##          1.23934717          0.57048294
```

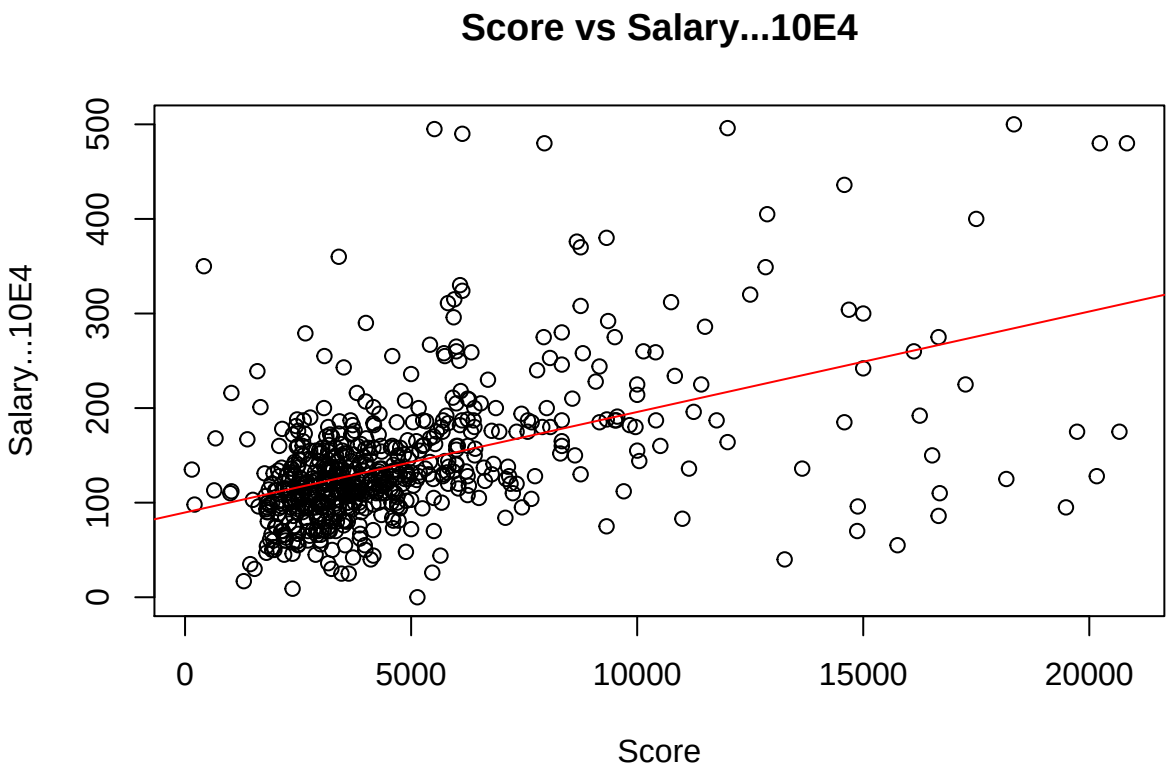
Vemos por tanto que respecto a la referencia multivariable tomada, vemos que corresponde al historial de ofertas laborales una probabilidad hasta 55 veces superior, por lo que supone un claro factor de riesgo en términos estadísticos. Por su parte, tener una localización semiurbana sería hasta 2,5 veces (+150%) más probable, de nuevo un factor de riesgo, para la contratación. Por su parte y sorprendentemente, contar con experiencia en Python reduce la probabilidad de contratación, para el conjunto de datos estudiado, en hasta un 43% respecto a los candidatos que no cuentan con ella, por lo que sería un factor de protección respecto a la contratación.

Es interesante tener en cuenta que, tal y como se indicó en la descripción del problema, una vez construido este modelo, puede ser utilizado para predecir la contratación, o no, de candidatos a ofertas de empleo, basándonos en las variables utilizadas para su construcción: la educación, el tipo de localización, el historial de ofertas y la experiencia de programación en Python.

5. Representación de resultados

Haremos una representación de los resultados para responder a varias de las preguntas planteadas inicialmente.

```
plot(Salary...10E4 ~ Score, data = data, xlab = "Score",
     ylab = "Salary...10E4", main="Score vs Salary...10E4")
MLatin <- lm(Salary...10E4 ~ Score, data = data)
abline(MLatin, col = "red")
```

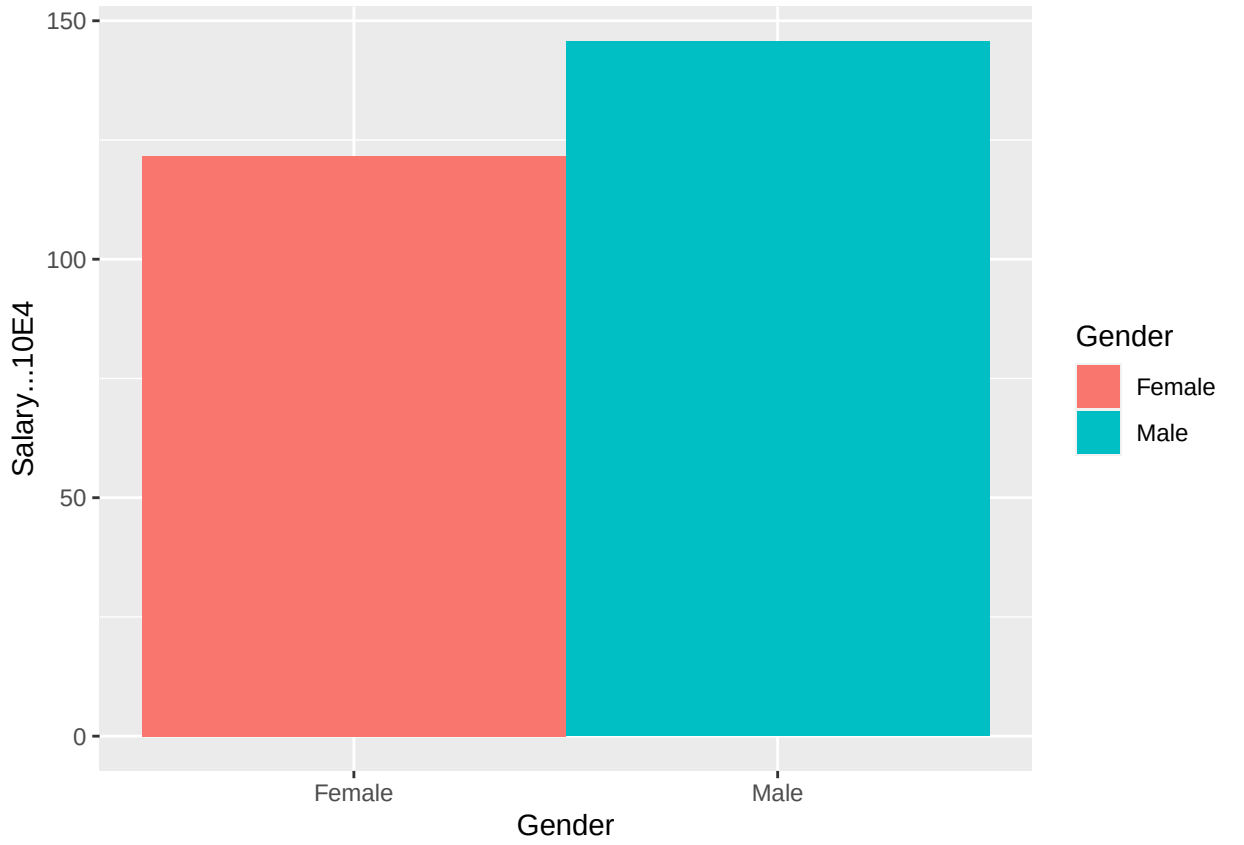


Como puede verse en la gráfica anterior, existe una relación entre el Score que recibe un candidato y su salario. Cuando mayor es el Score mayor suele ser su salario.

```
#mean(data$Salary...10E4[data$Gender=="Male"]) mean(data$Salary...10E4[data$Gender=="Female"])


ggplot( data, aes(x=Gender, y=Salary...10E4, fill=Gender)) +
geom_bar(width=1,stat="summary", fun.y="mean")
```

```
## Warning: Ignoring unknown parameters: fun.y
```

```
## No summary function supplied, defaulting to `mean_se()`
```



Según el gráfico de barras, puede verse que el salario medio de los hombres es superior al de las mujeres en el momento de contratación.

6. Resolución y conclusiones

En esta práctica se ha preparado y analizado un conjunto de datos, completando el ciclo de vida de un proyecto datos. Para ello, se ha partido de la exploración de un repositorio público, Kaggle, donde se ha localizado un conjunto de datos interesante e imperfecto.

Seleccionados los datos de relevancia e interés, se ha procedido a limpiar este conjunto de datos de elementos indeseados como ceros, elementos vacíos y ausentes, realizando imputación de valores utilizando modelos de minería de datos como *k-Nearest Neighbors*. Hecho esto, se han identificado y tratado los valores extremos.

A continuación, se ha realizado el análisis de datos propiamente, comenzando por la preselección de los datos categóricos a utilizar y la comprobación de normalidad y homogeneidad de varianza para los datos numéricos. Habiendo realizado dichas tareas de acondicionamiento, se ha exportado la versión preprocesada de los datos al fichero `recruitment_decision_tree_clean.csv`.

Se han aplicado tres métodos de análisis de datos diferentes, con los que se ha podido responder a las preguntas de interés planteadas. Hemos visto que sí hay correlación entre la experiencia del usuario y la puntuación otorgada a cada uno de ellos y el salario que consigue, aunque estas han resultado ser baja o moderada.

Asimismo, se ha determinado mediante contrastes de hipótesis que efectivamente, las mujeres no solo cobran distinto a los hombres sino que el salario de estas es inferior, con nivel de confianza del 95%.

También se han determinado las características cualitativas más relevantes para la contratación, en el marco del conjunto de datos analizado. Se ha visto que la experiencia en Python es algo relevante, actuando como

factor de protección respecto a esta (descendiendo la probabilidad de contratación en hasta un 43% respecto a quienes no cuentan con ella). Más relevante es la localización, que en el caso de ser de tipo semiurbana cuenta multiplica la probabilidad de contratación por 2,5. No obstante, el factor más importante es claramente contar con un historial de ofertas de empleo, el cual multiplica por 55 la probabilidad de contratación respecto a quienes no cuentan con él.

Finalmente se han representado gráficamente algunos de los resultados obtenidos, facilitando la comprensión y asimilación de las relaciones observadas. Estas son las asociadas al salario respecto a características como la puntuación del candidato, así como la comparación de los salarios de las mujeres frente a los de los hombres.

7. Código

El contenido de esta práctica se encuentra disponible en el repositorio Pra2DataAnalysis en GitHub, donde podemos encontrar el código en R utilizado para la realización de la misma en formato .Rmd, así como versiones en formato .pdf y .html del presente documento. En este repositorio encontramos además el *dataset* original, `recruitment_decision_tree.csv`, así como el conjunto de datos final tras realizar su limpieza, `recruitment_decision_tree_clean.csv`.

Contribuciones	Firma
Investigación previa	DO, SF
Redacción de las respuestas	DO, SF
Desarrollo código	DO, SF