EmployeePromotion

2023-02-18

Este proyecto está realizado por el grupo 3 formado por José Arturo Espaillat, Johnsiel Castaños, José Delgado, Salama Mohamed-fadel Sidna

El dataset elegido es llamado HR Analytics: Employer Promotion Data (https://www.kaggle.com/datasets/arashnic/hr-ana?select=test.csv) sacado de Kaggle, el cual cuenta con 13 columnas y 54808 filas. Estos datos parten de una empresa la cual tiene un problema debido a que las promociones definitivas no se anuncian hasta después de evaluar empleado por empleo haciendo de esto un proceso lento y tedioso. Gracias al análisis de estos históricos podemos entender cuales son las variables que más afectan a la promoción para aumentar la eficacia del proceso ya que ahorran mucho más tiempo al tener claro cuales son los potenciales candidatos. También los candidatos obtienen un punto de vista de que es lo que más repercute en su promoción, pudiendo así mejorar cierto aspecto de cara a la evaluación.

De esta manera conseguimos derribar el muro de las emociones evitando asi promociones no merecidas a empleados por el mero hecho de caer bien, conseguiendo así una criticidad para que todos los empleados entiendan por que son o no promocionados.Esto es un problema a dia de hoy y vemos muchas publicaciones de como evitar conflictos entre compañeros https://www.ieie.eu/como-ascender-a-un-empleado-sin-generar-conflictos/

Personalmente hemos querido abordar este problema porque esto es un problema real de nuestro dia a dia y nos pareció bastante interesante la idea de poder analizar este caso. Es cierto que esto es muy relativo a cada tipo de empresa la cual se fija más en unas variables que en otras pero nos puede aportar el procedimiento para extrapolarlo a otros campos Lo que queremos lograr es plantear el análisis de problemas de este estilo como que visualizacion nos puede ayudar, correlaciones, saber si nos podemos ahorrar pasos a través de sacar conclusiones tempranas...

En resumen creemos que es bastante interesante el estudio de este dataset ya que tiene cierta importancia a la hora de poder entender porque las personas son ascendidas en su trabajo y desde el punto de vista de la empresa les beneficia en el tiempo ahorrado en estos procesos de promoción.

Para organizarnos, hemos decidido crear un repositorio en GitHub (https://github.com/Josdelser/DataScienceEmployeePromotion/tree/develop) para trabajar simultáneamente. También hemos creado una bolsa de preguntas y las hemos asignado según las capacidades y aspiraciones de cada persona. En el caso de que algún miembro quiera obtener una calificación más alta, deberá realizar un mayor número de preguntas individuales. Todos los miembros realizarán una pregunta para la parte grupal y se ha intentado que estas preguntas estén relacionadas con las preguntas individuales, para poder afrontar mejor el desafío.

A continuación, se detalla la asignación de cada miembro:

Jose Delgado:

-Grupal

```
¿Podemos sacar conclusiones tempranas de que variables afectan prediciendo si será promovido un emplead
```

```
¿Quitando variables mejorará el arbol?
```

-Individual:

¿Nos puede ayudar la visualización a sacar conclusiones temprana?

Uso de la correlacion cramer para responder a ¿Existe una correlación significativa entre las variables

Salama: Johnsiel: Jose Espaillat:

Cada miembro realizará la documentación de ambas partes asignadas. Además en el documento se indicará donde empieza y acaba la parte de cada miembro para que así sea mas sencilla su evaluación.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

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

```
## Loading required package: tibble

## Loading required package: bitops

## Rattle: A free graphical interface for data science with R.
## Versión 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.
## Escriba 'rattle()' para agitar, sacudir y  rotar sus datos.
```

```
library(dplyr)
library(ggplot2)
library(RColorBrewer)
library(ggfortify)
library(vcd)
```

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

```
library(caret)
```

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

```
require(corrplot)
```

```
## Loading required package: corrplot

## corrplot 0.92 loaded
```

```r
set.seed(28)
```

Información general del Dataset

```r
data<- read.csv("train.csv")
data <- rename(data, awards_won = awards_won.)
head(data)
```

```
##   employee_id        department    region        education gender
## 1       65438 Sales & Marketing  region_7 Master's & above      f
## 2       65141        Operations region_22       Bachelor's      m
## 3        7513 Sales & Marketing region_19       Bachelor's      m
## 4        2542 Sales & Marketing region_23       Bachelor's      m
## 5       48945        Technology region_26       Bachelor's      m
## 6       58896         Analytics  region_2       Bachelor's      m
##   recruitment_channel no_of_trainings age previous_year_rating
## 1           sourcing                1  35                    5
## 2              other                1  30                    5
## 3           sourcing                1  34                    3
## 4              other                2  39                    1
## 5              other                1  45                    3
## 6           sourcing                2  31                    3
##   length_of_service awards_won avg_training_score is_promoted
## 1                 8          0                 49           0
## 2                 4          0                 60           0
## 3                 7          0                 50           0
## 4                10          0                 50           0
## 5                 2          0                 73           0
## 6                 7          0                 85           0
```

```r
colnames(data)
```

```
##  [1] "employee_id"          "department"           "region"
##  [4] "education"            "gender"               "recruitment_channel"
##  [7] "no_of_trainings"      "age"                  "previous_year_rating"
## [10] "length_of_service"    "awards_won"           "avg_training_score"
## [13] "is_promoted"
```

```r
attach(data)
```

```r
str(data)
```

```
## 'data.frame':    54808 obs. of  13 variables:
##  $ employee_id         : int  65438 65141 7513 2542 48945 58896 20379 16290 73202 28911 ...
##  $ department          : chr  "Sales & Marketing" "Operations" "Sales & Marketing" "Sales & Marketing" ...
##  $ region              : chr  "region_7" "region_22" "region_19" "region_23" ...
##  $ education           : chr  "Master's & above" "Bachelor's" "Bachelor's" "Bachelor's" ...
##  $ gender              : chr  "f" "m" "m" "m" ...
##  $ recruitment_channel : chr  "sourcing" "other" "sourcing" "other" ...
##  $ no_of_trainings     : int  1 1 1 2 1 2 1 1 1 1 ...
##  $ age                 : int  35 30 34 39 45 31 31 33 28 32 ...
```

```
##  $ previous_year_rating: num  5 5 3 1 3 3 3 3 4 5 ...
##  $ length_of_service   : int  8 4 7 10 2 7 5 6 5 5 ...
##  $ awards_won          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ avg_training_score  : int  49 60 50 50 73 85 59 63 83 54 ...
##  $ is_promoted         : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
dim(data)
```

```
## [1] 54808    13
```

```
object.size(data)/1024 #KB
```

```
## 4073.3 bytes
```

```
object.size(data)/1024^2 #MB
```

```
## 4 bytes
```

```
object.size(data)/1024^3 #GB
```

```
## 0 bytes
```

Vemos que el número de bytes del dataset es 4073,3 KB o 4,1 MB una vez lo tenemos cargado

Es verdad que el dataset está presentable pero aún así tenemos que hacerle una etapa de transformaciones y limpieza. Primero vamos a quitar todos los NA y duplicados, luego cada mienbro para responder a sus preguntas hará mas transformaciones.

```
print(paste("Número de valores faltantes totales:", sum(is.na(data))))
```

```
## [1] "Número de valores faltantes totales: 4124"
```

```
data <- na.omit(data)
data <- unique(data)
```

*Aquí empieza la parte de Jose Delgado

Una vez cargado el dataset y tratado mínimamente vamos a pasar a hacer responder a las preguntas, primero la pregunta grupal:

¿Cómo predecir si será promovido un empleado?

La cual se plantea responder mediante una predicción utilizando validación cruzada (cross validation), para ello podemos tirar de regresión logística o un árbol de decisión. En este caso usaré árboles de decisión para así también ver qué variables influyen y poder comparar este resultado con las preguntas individuales

Primero le hacemos preprocesamiento de manera preparatoria a la predicción, luego dividimos el dataset en conjunto de prueba(test) y entrenamiento(train) para obtener resultado

```
datajose <- data
```

Luego convertimos las variables categóricas en factores y definimos explícitamente los niveles de cada categoría.

```
dep_levels <- unique(datajose$department)
gen_levels <- unique(datajose$gender)
recru_levels <- unique(datajose$recruitment_channel)
promo_levels <- unique(datajose$is_promoted )
award_levels <- unique(datajose$awards_won)


datajose$department <- factor(datajose$department, levels = dep_levels)
datajose$gender  <- factor(datajose$gender , levels = gen_levels)
datajose$recruitment_channel <- factor(datajose$recruitment_channel, levels = recru_levels)
datajose$is_promoted <- factor(datajose$is_promoted, levels = promo_levels)
datajose$awards_won <- factor(datajose$awards_won, levels = award_levels)
```

Partimos de un dataset exclusivo para la predicción y que no afecte a otros cálculos

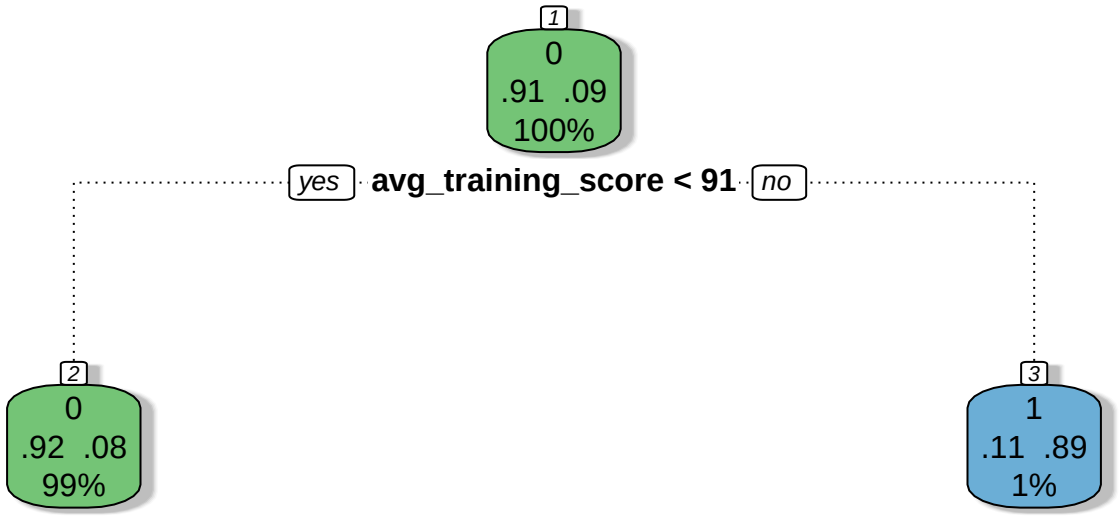
```
datajosepred <-datajose
```

```
division <- createDataPartition(datajosepred$is_promoted, p = .7, list = FALSE, times = 1)
train <- datajosepred[division, ]
test  <- datajosepred[-division, ]
```

Entrenamos el modelo y visualizamos para tener una primera imagen de las variables decisivas

```
arbol <- rpart(formula =  is_promoted  ~ ., data = train, method = 'class')

fancyRpartPlot(arbol)
```



Podemos observar que la variable que más afecta es la puntuación en el rango superior a 91, donde con un 99% de posibilidades serás promocionado. Al tener una variable con tanta importancia se nos viene a la cabeza otra pregunta: Quitando esta variable, ¿Como sería el árbol final? Se verá mas adelante.

Pasamos a predecir según clasificación

```
prediccion <-predict(arbol, test, type = "class")
```

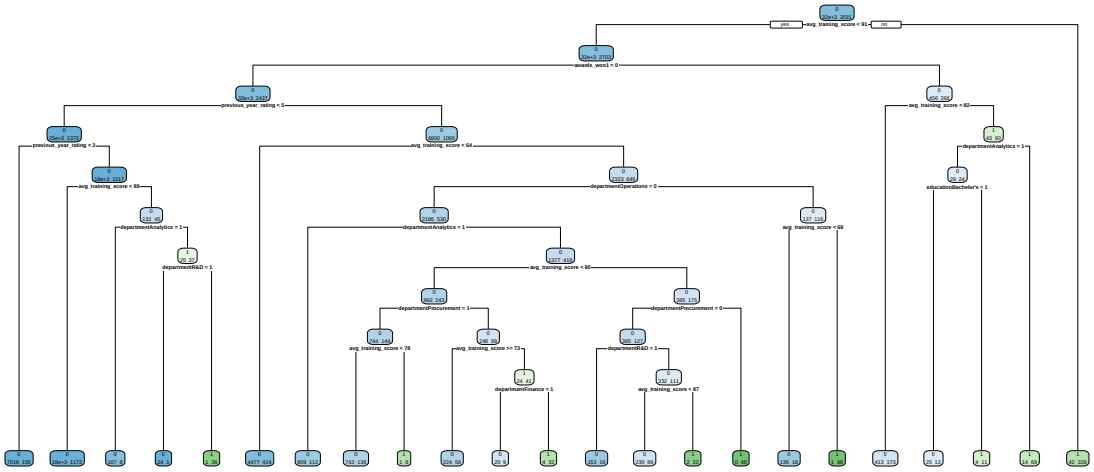
Ahora toca evaluar el rendimiento según validación cruzada.

```
valcruz <- trainControl(method = "cv", number = 10)

arbolfit <- train(is_promoted ~ ., data = train, method = "rpart",
                  trControl = valcruz, tuneLength = 10)
```

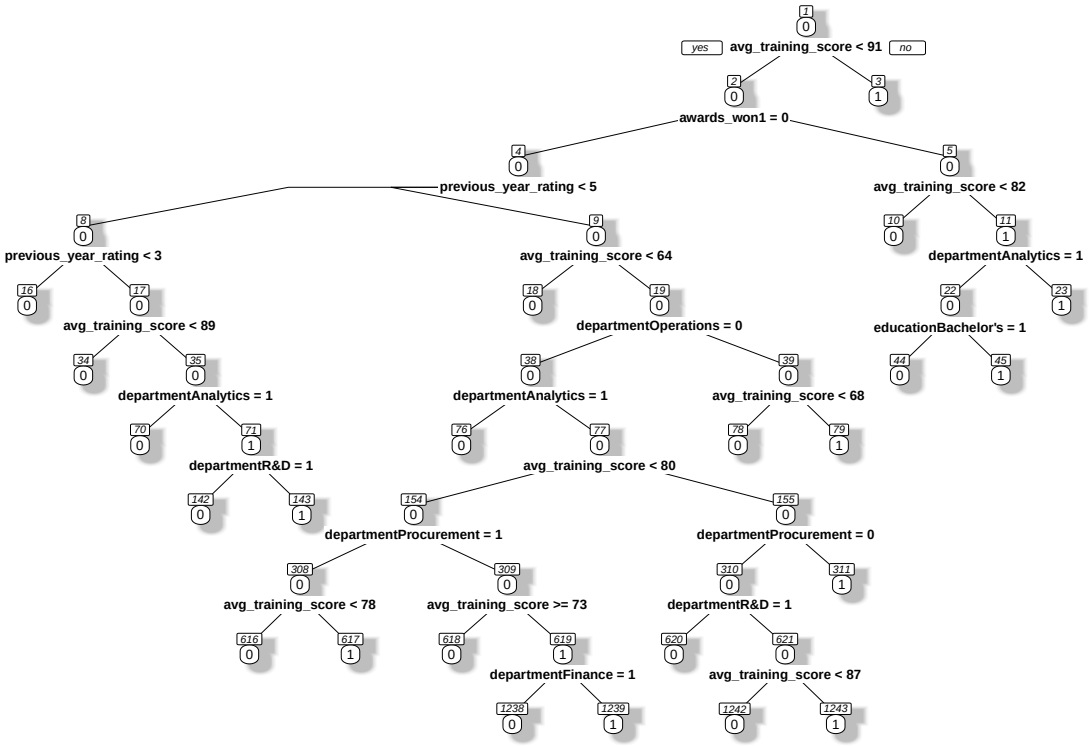
Graficamos

```
rpart.plot(arbolfit$finalModel, type = 2, extra = 1)
```



Mejoramos la visualización para ver mejor el nombre de las variable sy al estructura del árbol

```
prp(arbolfit$finalModel, type = 2, nn = TRUE,
    fallen.leaves = FALSE,
    varlen = 0,  shadow.col = "gray")
```




```
prediccionTest <- predict(arbolfit, newdata = test)
confusionMatrix(prediccionTest, test$is_promoted)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 13861  1044
##          1    45   254
##
##                Accuracy : 0.9284
##                  95% CI : (0.9242, 0.9324)
##     No Information Rate : 0.9146
##     P-Value [Acc > NIR] : 2.733e-10
##
##                   Kappa : 0.2956
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9968
##             Specificity : 0.1957
##          Pos Pred Value : 0.9300
##          Neg Pred Value : 0.8495
##              Prevalence : 0.9146
##          Detection Rate : 0.9117
```

```
##    Detection Prevalence : 0.9803
##       Balanced Accuracy : 0.5962
##
##        'Positive' Class : 0
##
```

Con estos datos ya podemos sacar conclusiones sobre el modelo:

- Hay 13,861 verdaderos negativos y 254 verdaderos positivos
- Hay 1,044 falsos negativos y 45 falsos positivos

Ya con estos valores vemos que el modelo está bien ajustado. Con estos datos obtenemos una precisión del 92,8% y una tasa de error del 7,16%. También observamos que tenemos mayor capacidad para acertar, es decir predecir cuándo se va a promover que cuando no se va a promover.

Además observando el arbolfit vemos que ya aparecen más variables que son relevantes como award_won y previus_year_rating. Añadir que vemos una fuerte tendencia en el departamento de analiticas, donde superando los 82 puntos serás promocionado con alta seguridad.

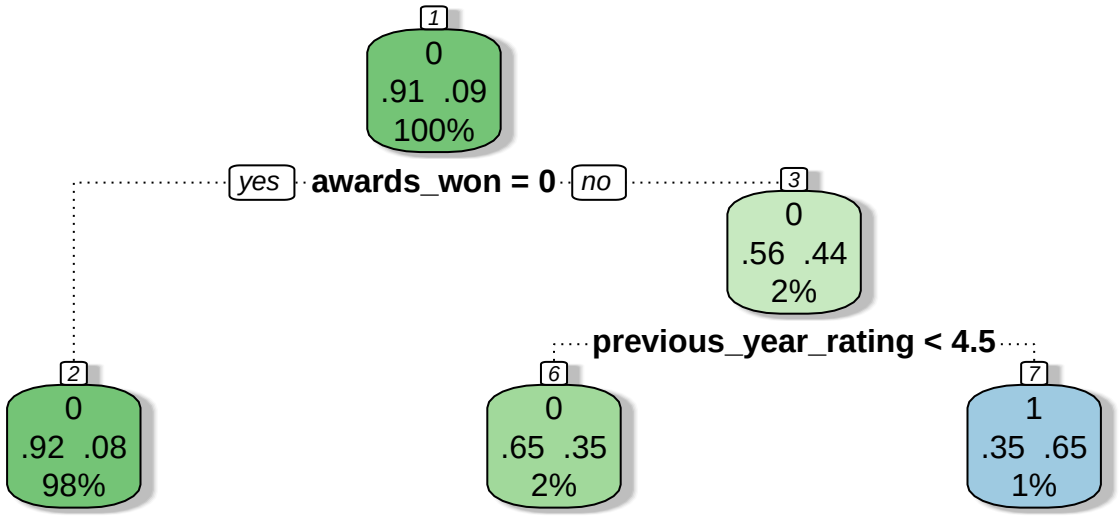
Segunda pregunta

```
¿Quitando variables mejorará el árbol?
```

Hemos visto que avg_training_score tiene mucho peso, por lo que me surgió la pregunta de que si eliminando esta variable que otras variables afectan, además de ver si mejoramos el modelo.

```
datajosearbol2 <-datajose
```

```
arbol2 <- rpart(is_promoted ~ . - avg_training_score, data = datajosearbol2, method = "class")
fancyRpartPlot(arbol2)
```



Rattle 2023−mar.−12 21:01:00 josed

Directamente vemos que nos da resultados parecidos al árbol entrenado, volviendo a afirmar que el modelo está bien ajustado. Con esta conclusión nos ahorramos de volver a predecir ya que intuimos que en el mejor de los casos va a igualar a los resultados anteriores. Gracias a un análisis previo nos hemos ahorrado esta parte.

Jose preguntas individuales:

Una vez hecha la predicción, me pregunté si había alguna gráfica que me hubiese podido ayudar a sacar una conclusión temprana de las variables que más afectan. Para ello voy a realizar gráficas tanto para variables numéricas como categóricas.

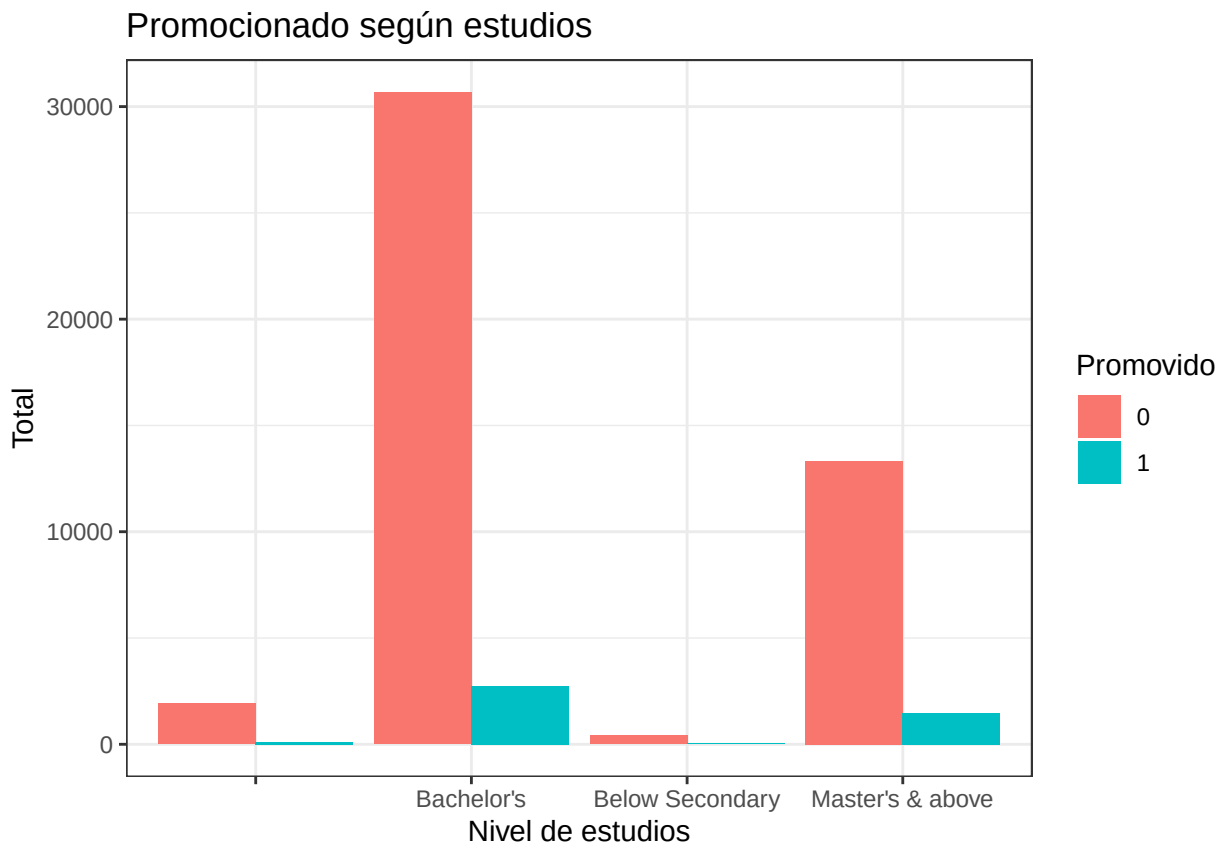
Pondré el caso de que no hemos predicho todavía y no se qué variables afectan, de esta manera poder simular un problema real ya que mi tendencia será ir a por las variables que creo que puedan afectar más.

Volvemos a partir de un dataset exclusivo para esta parte

```
datajosegraficas <- datajose
```

Primero vamos a ver las variables categóricas. Empezamos por ver si los estudios influyen, vamos a graficar la distribución de empleados promovidos según sus estudios

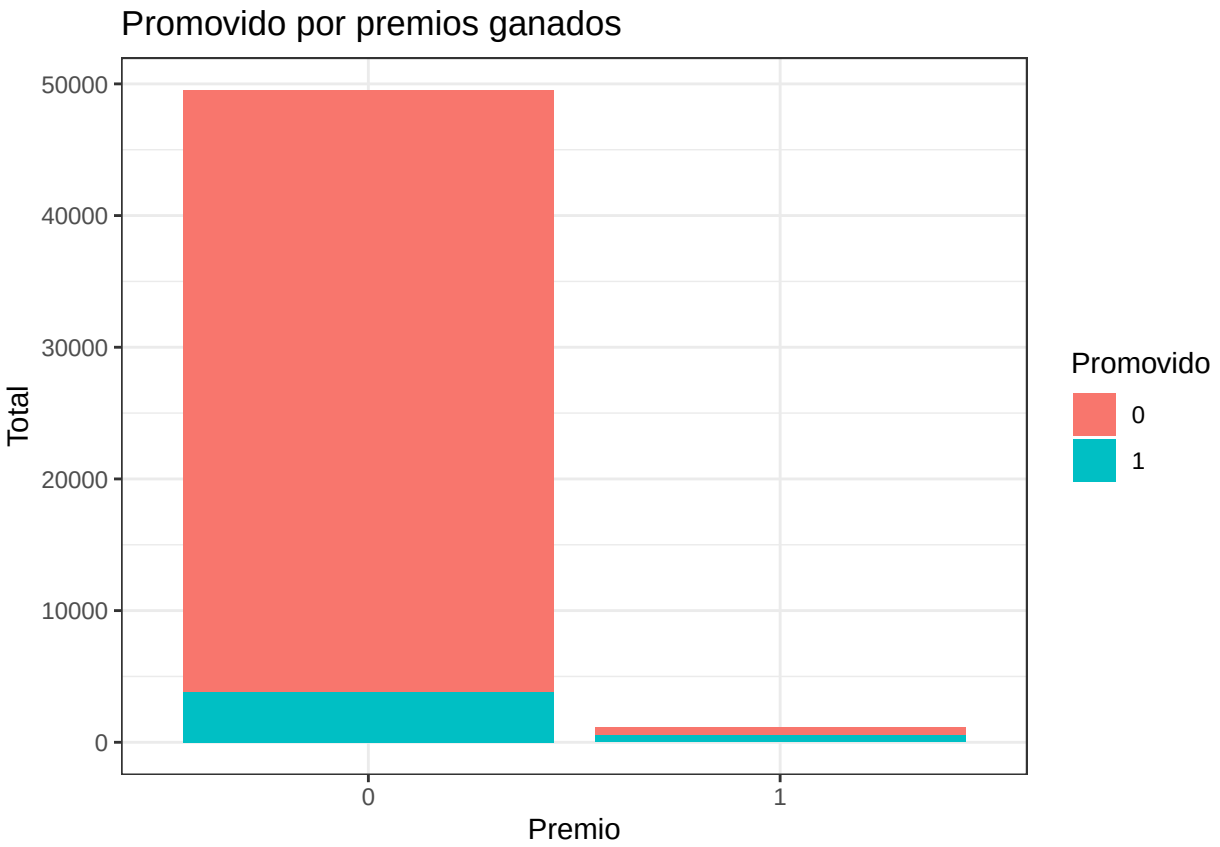
```
ggplot(data = datajosegraficas, aes(x = education, fill = is_promoted)) +
  geom_bar(position = "dodge") +
  labs(title = "Promocionado según estudios",
       x = "Nivel de estudios",
       y = "Total",
       fill = "Promovido") +
  theme_bw()
```



Observamos que la mayoría tienen como mínimo el bachillerato y luego el Master. Podemos pensar que es una variable que afecta debido a las diferencias entre promovido o no pero no es decisiva o tiene una importancia mayor.

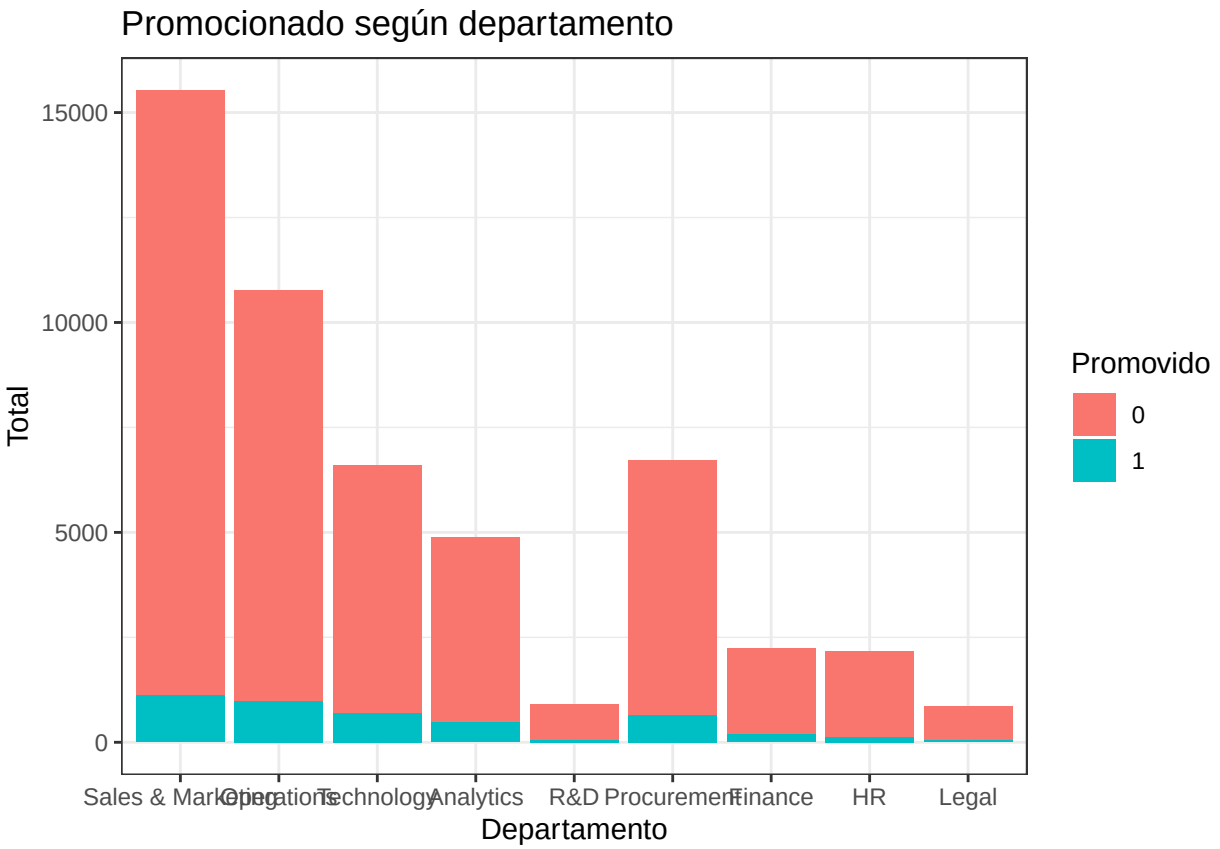
Repetimos pero con los premios ganados

```
ggplot(data = datajosegraficas, aes(x = awards_won, fill = is_promoted)) +
  geom_bar(position = "stack") +
  labs(title = "Promovido por premios ganados",
       x = "Premio",
       y = "Total",
       fill = "Promovido") +
  theme_bw()
```



Se observa que en los que están promovidos tiene mayor relevancia tener un premio, ya que el número de promovidos con premios es notablemente mayor aunque a simple vista no se observe.

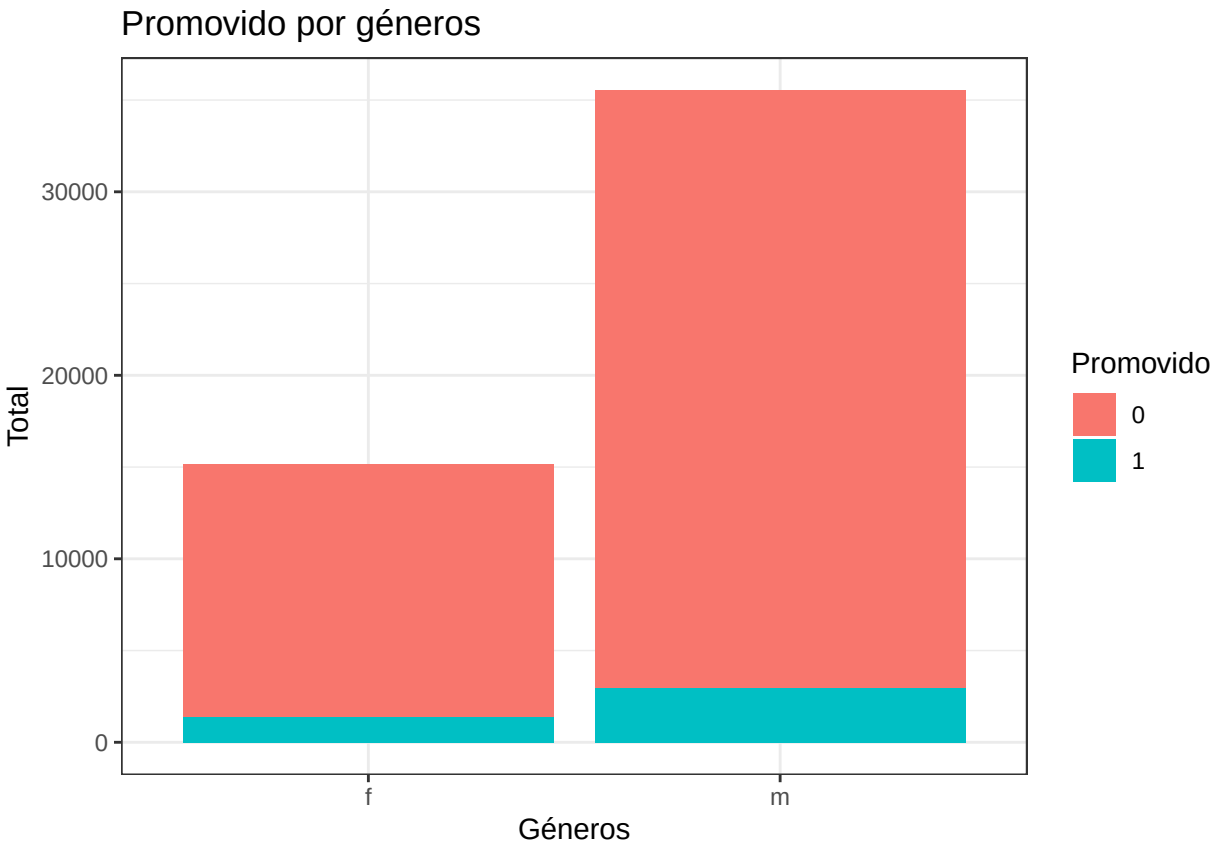
```
ggplot(data = datajosegraficas, aes(x = department, fill = is_promoted)) +
  geom_bar(position = "stack") +
  labs(title = "Promocionado según departamento",
       x = "Departamento",
       y = "Total",
       fill = "Promovido") +
  theme_bw()
```



Con este gráfico de barras apiladas vemos que esta variable no afecta fuertemente a la promoción ya que cada departamento tiene resultados iguales si no contamos en proporción porque al no saber a qué se dedica la empresa no podemos justificar el número de empleados por cada departamento.

Por último vamos a ver si el género afecta.

```
ggplot(data = datajosegraficas, aes(x = gender, fill = is_promoted)) +
  geom_bar(position = "stack") +
  labs(title = "Promovido por géneros",
       x = "Géneros",
       y = "Total",
       fill = "Promovido") +
  theme_bw()
```

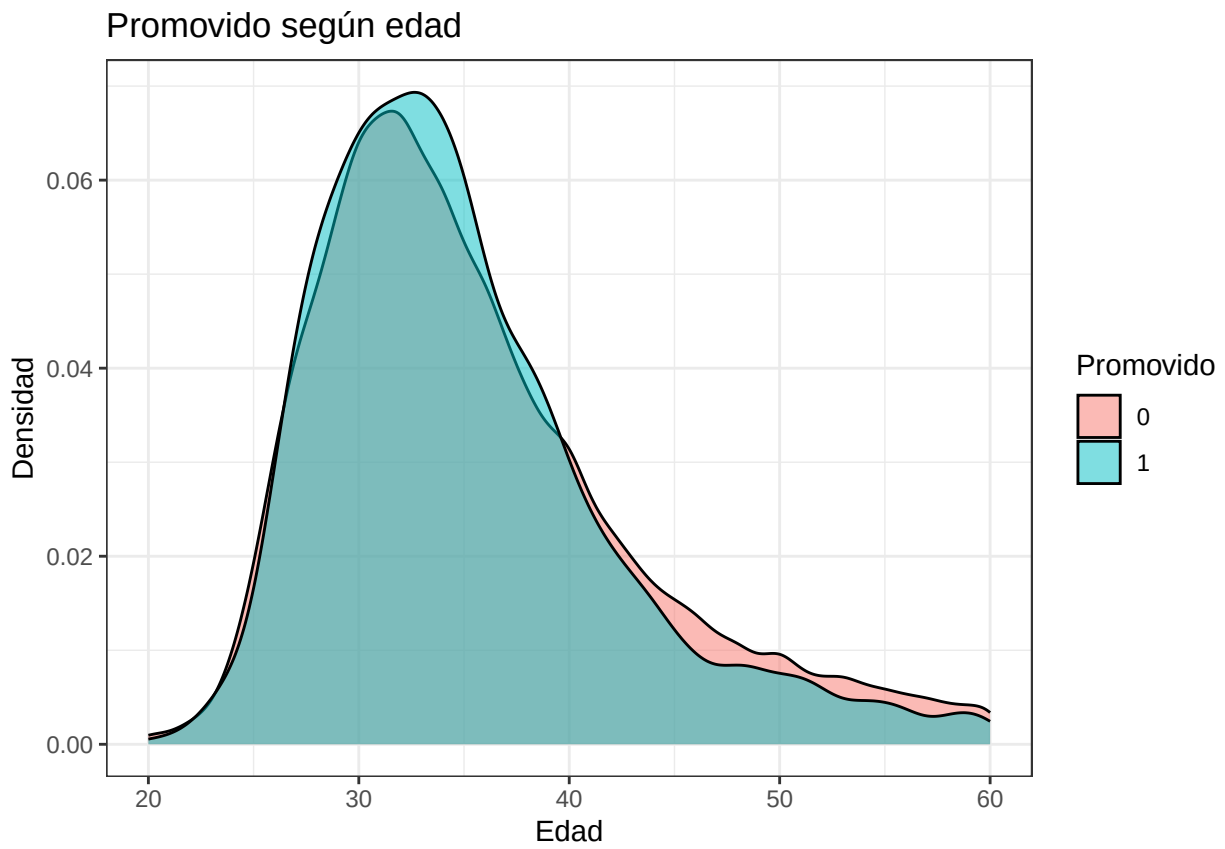


Aqui vemos que en proporcion que no afecta el género, ya que a simple vista los porcentajes de promocio segun el sexo parecen cercanoss.

Ahora pasamos a las variables numéricas.

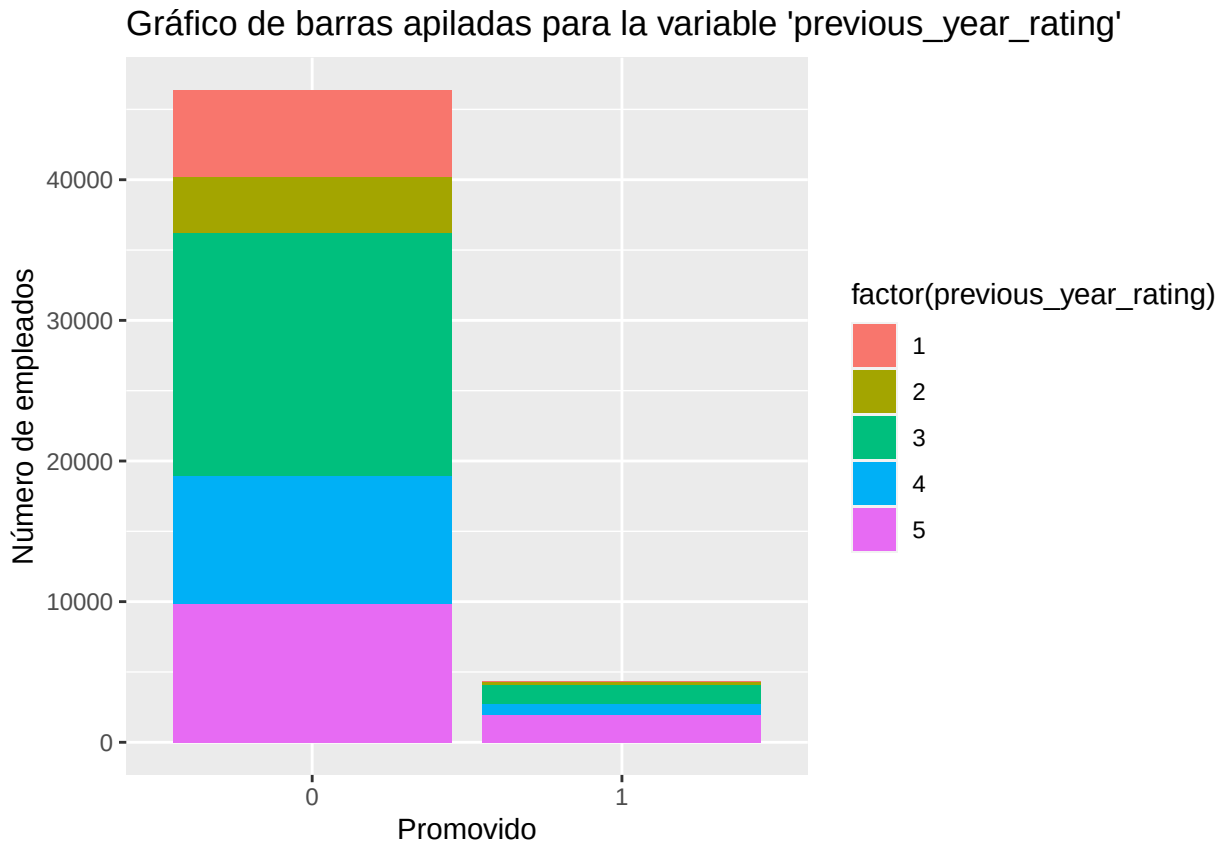
Empezamos con la edad representada en un gráfico de densidad para ver donde se concentran la mayoría de promocionados por si vemos que la edad pueda afectar ya que también está relacionada con el tiempo que lleves en la empresa

```
ggplot(data = datajosegraficas, aes(x = age, fill = is_promoted)) +
  geom_density(alpha = 0.5) +
  labs(title = "Promovido según edad",
       x = "Edad",
       y = "Densidad",
       fill = "Promovido") +
  theme_bw()
```



El rango es bastante amplio por lo que no vemos que la edad concreta pueda afectar, pero si puede afectar dependiendo de los años que lleve en la empresa, es decir, una persona con 40 años tiene más probabilidad de llevar más tiempo en la empresa que una persona de 20 años. Por eso vamos a graficar ahora si el tiempo en la empresa afecta

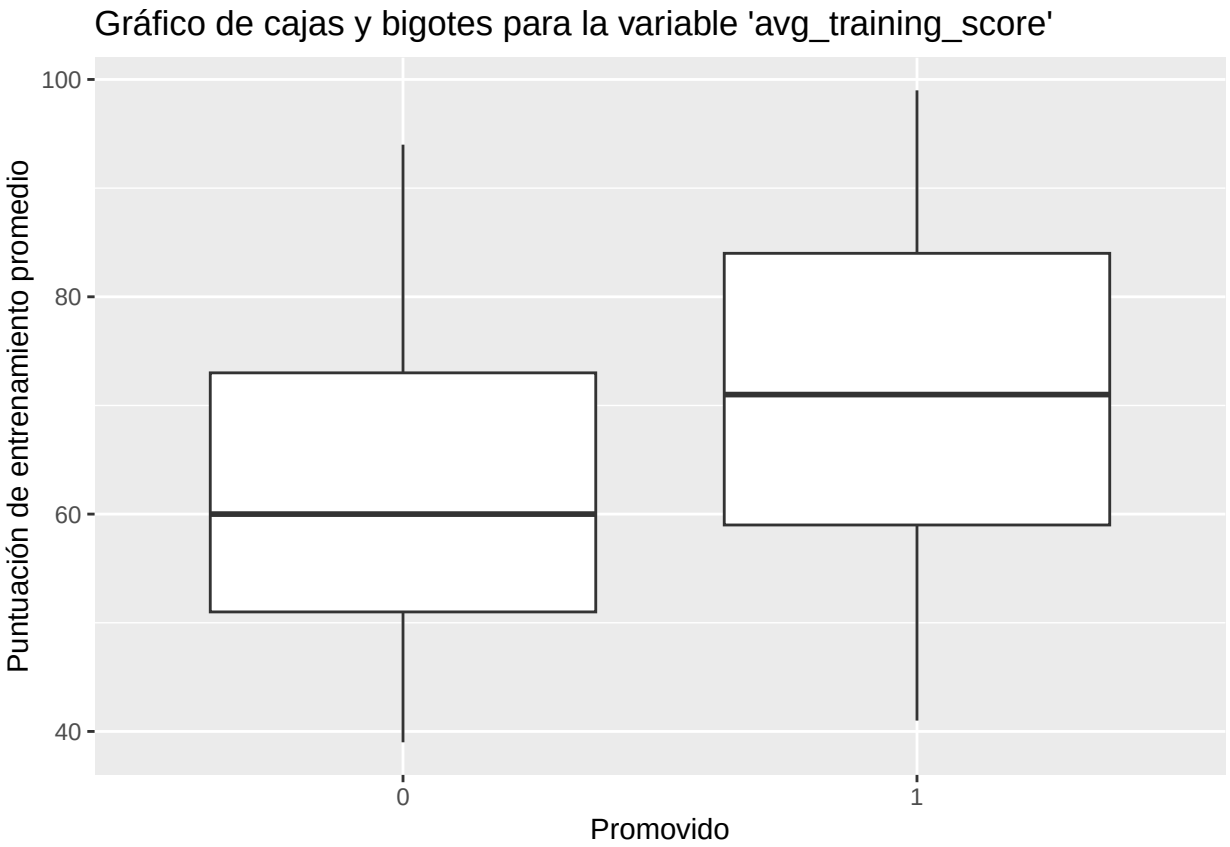
```
ggplot(data = datajosegraficas, aes(x = is_promoted, fill = factor(previous_year_rating))) +
  geom_bar(position = "stack") +
  labs(x = "Promovido", y = "Número de empleados") +
  ggtitle("Gráfico de barras apiladas para la variable 'previous_year_rating'")
```



Ahora si encontramos dos grupos que afectan a la promoción, los que llevan 5 años y 3.

Por último vamos a probar con la variable avg_training_score para ver si esta nos afecta.Para ello vamos con un gráfico de cajas y bigotes

```
ggplot(data = datajosegraficas, aes(x = is_promoted, y = avg_training_score)) +
  geom_boxplot() +
  labs(x = "Promovido", y = "Puntuación de entrenamiento promedio") +
  ggtitle("Gráfico de cajas y bigotes para la variable 'avg_training_score'")
```



Observamos que la variable av_training_score es bastante importante ya que por encima de cierto rango se concentra todos los que han sido promovidos.

La conclusión final que sacamos de estas gráficas está bastante relacionada con la de la predicción. Vemos que ciertas variables tienden a afectar de manera directa a la promoción y tanto en gráficas como en predicción concuerdan unas con otras.

Segunda pregunta individual:

```
Uso de la correlación cramer para responder a ¿Existe una correlación significativa entre las variables
```

La correlación de cramer es un coeficiente que nos indica el grado de relación entre dos variables categóricas, de esta manera si probamos todas la variables categorías frente a is_promoted podremos analizar cuáles son las que mayor relación tienen. Obviamos las variables como región, género y recuitment_channel ya que previamente hemos visto que no tienen una relación importante. Se realizará con departamentos, estudios, género y premios ganados

Para ello primero tenemos que crear tablas de contingencias con las variables

```
tabladepa <- table(datajose$department, datajose$is_promoted)
tablagen <- table(datajose$gender, datajose$is_promoted)
tablaedu <- table(datajose$education, datajose$is_promoted)
tablawards <- table(datajose$awards_won, datajose$is_promoted)
```

Luego con la función assocstats() calculamos diferentes estadísticas como la correlación de Pearson, chi-cuadrado y el estadístico V de cramer, que es el que buscamos

```
cramerdepa<-assocstats(tabladepa)
cramergen<-assocstats(tablagen)
crameredu<-assocstats(tablaedu)
cramerawards<-assocstats(tablawards)
```

```
cramerdepa$cramer
```

```
## [1] 0.05307495
```

```
cramergen$cramer
```

```
## [1] 0.01243931
```

```
crameredu$cramer
```

```
## [1] 0.03884881
```

```
cramerawards$cramer
```

```
## [1] 0.1974461
```

Estos resultados son la V de cramer que nos da la correlación. Observamos, como también hemos demostrado arriba con las gráficas, que tener un premio ganado tiene cierta importancia y sobre todo respecto al departamento y educación. Volvemos a corroborar que el género no influyen en nada a la hora de ser promovido.

Finalmente después de realizar todo este proceso de análisis he llegado a la conclusión de que un buen estudio de los datos y conocimiento del desafío puede ahorrarnos horas de investigación y elección de caminos que no llevan a nada. Luego de realizar todo esto, podríamos volver a limpiar el dataset eliminando columnas bajo un criterio de correlación, es decir prescindir de variables que no afectan a la promoción y centrarnos en las que sí

- Aquí empieza la parte de Salama
- Aquí empieza la parte de Johnsiel
- Aquí empieza la parte de Jose Espaillat