PRA2: Limpieza y análisis de datos

Alberto García y Alejandro Floriano

5/1/2021

Índice

1. Detalles de la práctica

1.1. Presentación

Los procesos de carga, limpieza y transformación de datos establecen una de las partes fundamentales en el ciclo de vida de un proyecto de análisis de datos. Constituyen toda una serie de métodos y técnicas que nos permiten corregir irregularidades y posibles errores con el objetivo de presentar un conjunto de datos final de calidad listo para ser analizado mediante modelos estadísticos.

En este contexto se enmarca esta segunda práctica de la asignatura **Tipología y ciclo de vida de los datos**. Se propone la realización de un caso práctico en el cual se precisan llevar a cabo los procedimientos de limpieza de datos estudiados en la asignatura para posteriormente realizar análisis estadísticos que nos permitan optimizar la toma de decisiones basadas en datos. Para ello, partiremos de un conjunto de datos que deberemos explorar y que presentará una problemática de negocio concreta. Toda la práctica se desarrollará utilizando el lenguaje de programación `R`, con la gama de herramientas disponibles y que iremos mostrando a lo largo del desarrollo.

1.2. Competenecias

En esta práctica se desarrollan, por tanto, las siguientes competencias del **Máster en Ciencia de Datos (Data Science)** de la **Universitat Oberta de Catalunya**:

- Capacidad de analizar un problema en el nivel de abstracción adecuado a cada situación y aplicar las habilidades y conocimientos adquiridos para abordarlo y resolverlo.
- Capacidad para aplicar las técnicas específicas de tratamiento de datos (integración, transformación, limpieza y validación) para su posterior análisis.

1.3. Objetivos

Los objetivos específicos de esta práctica son:

- Aprender a aplicar los conocimientos adquiridos y desarrollar una capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios o multidisciplinares.
- Saber identificar los datos relevantes y los tratamientos necesarios (integración, limpieza y validación) para llevar a cabo un proyecto analítico de datos.
- Aprender a analizar los datos adecuadamente para abordar la información contenida en los datos.
- Identificar la mejor representación de los resultados para aportar conclusiones sobre el problema planteado en el proceso analítico.
- Actuar con los principios éticos y legales relacionados con la manipulación de datos en función del ámbito de aplicación.
- Desarrollar las habilidades de aprendizaje que permitan continuar estudiando de un modo autodirigido o autónomo.
- Desarrollar la capacidad de búsqueda, gestión y uso de información y recursos en el ámbito de la ciencia de datos.

2. Realización de la práctica

2.1. Descripción del conjunto de datos

El conjunto de datos seleccionado para la realización del caso práctico se denomina **Lead Scoring Dataset**. Se encuentra alojado en Kaggle, la plataforma de ciencia de datos conocida por albergar algunas de las competiciones más conocidas de aprendizaje automático.

`https://www.kaggle.com/amritachatterjee09/lead-scoring-dataset`

En primer lugar, detallaremos el contexto general del conjunto de datos, explicando la importancia del mismo y el problema de negocio que se plantea y que se pretende resolver. Tras ello, se hará una descripción del conjunto de variables que conforman el archivo.

Este conjunto de datos contiene información de una empresa de educación (denominada *X Education*) que se dedica a vender cursos formativos vía internet a través de distintas páginas webs. Cuando una persona está interesada en alguno de los cursos, rellena un formulario con cierta información personal (teléfono, email, ...) y pasa, en terminología de la empresa, a convertirse en un *lead*.[1]

Sobre cada *lead*, el departamento de ventas de la empresa comienza una campaña de comunicación personal cuyo objetivo final es la compra del curso por el que se mostró interés. La empresa *X Education* ha detectado que la tasa de conversión de *lead* a alumno inscrito con el plan de negocio actual es de alrededor un 30%, por lo que está interesada en conocer, a partir de la información recogida, qué subgrupo de *leads* son los que tienen una mayor probabilidad de convertirse finalmente en alumnos. La empresa se ha planteado, como objetivo final, alcanzar una tasa de conversión de *lead* a alumno de aproximadamente un 80%.

Para dar solución a dicha problemática, se dispone de un conjunto de datos compuesto por la informacíon de **9240 registros**, de manera que cada registro corresponde a un único *lead*, sobre los cuales se han recogido **37 variables** diferentes en el proceso de comunicación. Este conjunto de datos sería la base para resolver dicho problema a través de, por ejemplo, algún modelo de minería de datos de clasificación supervisada, especialmente útil a la hora de optimizar los esfuerzos y recursos del departamento de ventas y aumentar, a su vez, el número de alumnos inscritos.

A continuación, realizaremos una breve descripción de cada una de las variables recogidas.

- **Prospect ID**: Identificador único del usuario (variable de cadena).
- **Lead Number**: Identificador numérico único para cada *lead* (variable de cadena).
- **Lead Origin**: Origen del *lead*: la API, *landing page submission*,... (cualitativa nominal multicategórica).
- **Lead Source**: Fuente del *lead* en internet: Facebook, Google... (cualitativa nominal multicategórica).
- **Do Not Email**: Opción dada al usuario sobre si quiere recibir información vía email del curso (cualitativa nominal dicotómica).
- **Do Not Call**: Opción dada al usuario sobre si quiere recibir información vía llamada telefónica (cualitativa nominal dicotómica).
- **Converted**: La variable objetivo. Indica si tras el proceso de comunicación con el *lead*, éste se transfomó en alumno de algún curso (cualitativa nominal dicotómica).
- **TotalVisits**: Número de visitas realizadas por el usuario *lead* (cuantitativa discreta).
- **Total Time Spent on Website**: Tiempo total del *lead* en la web (cuantitativa discreta).
- **Page Views Per Visit**: Número promedio de páginas vistas durante las visitas (cuantitativa continua).
- **Last Activity**: Última actividad llevada a cabo por el *lead* (cualitativa nominal multicategórica).

[1] Un *lead* es un usuario que ha entregado sus datos a una empresa y que, como consecuencia, pasa a ser un registro de su base de datos con el que la organización puede interactuar. Este registro puede realizarse de forma física, con papel y boli, o de manera online, a través de un formulario. (https://www.inboundcycle.com/blog-de-inbound-marketing/que-es-un-lead)

- **Country**: País del *lead* (cualitativa nominal multicategórica).
- **Specialization**: Área de la industria en la que el *lead* trabajó previamente (cualitativa nominal multicategórica).
- **How did you hear about X Education**: ¿Cómo llegó a saber el *lead* de *X Education*? (cualitativa nominal multicategórica).
- **What is your current occupation**: Ocupación actual del *lead*(cualitativa nominal multicategórica).
- **What matters most to you in choosing this course**: Motivación para realizar el curso seleccionado (cualitativa nominal multicategórica).
- **Search**: ¿Se llegó a *X Education* a través de búsqueda en internet? (cualitativa nominal dicotómica).
- **Magazine**: ¿Se llegó a *X Education* a través de alguna revista? (cualitativa nominal dicotómica).
- **Newspaper Article**: ¿Se llegó a *X Education* a través de algún artículo en periódico? (cualitativa nominal dicotómica).
- **X Education Forums**: ¿Se llegó a *X Education* a través de algún foro? (cualitativa nominal dicotómica).
- **Newspaper**: ¿Se llegó a *X Education* a través de un periódico? (cualitativa nominal dicotómica).
- **Digital Advertisement**: ¿Se llegó a *X Education* a través de anuncios digitales? (cualitativa nominal dicotómica).
- **Through Recommendations**: ¿Se llegó a *X Education* a través de una recomendación? (cualitativa nominal dicotómica).
- **Receive More Updates About Our Courses**: Indica si el usuario quiere recibir actualizaciones sobre cursos (cualitativa nominal dicotómica).
- **Tags**: Etiqueta asignada por el departamento de ventas sobree el último estado del *lead* (cualitativa nominal multicategórica).
- **Lead Quality**: Indica la calidad del *lead* para convertirse en estudiante basado en la intuición del servicio de ventas (cualitativa ordinal multicategórica).
- **Update me on Supply Chain Content**: Indica si el usuario quiere actualización del contenido *Supply Chain* (cualitativa nominal dicotómica).
- **Get updates on DM Content**: Indica si el usuario quiere una actualización del contenido a través de un mensaje directo (cualitativa dicotómica).
- **Lead Profile**: Nivel asignado por el departamento de ventas al *lead* basado en su perfil (cualitativa ordinal multicategórica).
- **City**: Ciudad del *lead* (cualitativa nominal multicategórica).
- **Asymmetrique Activity Index**: Índice asignado al *lead* en función de su actividad (cualitativa ordinal multicategórica).
- **Asymmetrique Profile Index**: Índice asignado al *lead* en función de su perfil (cualitativa ordinal multicategórica).
- **Asymmetrique Activity Score**: Puntuación asignada al *lead* en función de la actividad (cuantitativa discreta).
- **Asymmetrique Profile Score**: Puntuación asignada al *lead* en función del perfil (cuantitativa discreta).
- **I agree to pay the amount through cheque**: Indica si el usuario estaría dispuesto a pagar con cheque (cualitativa nominal dicotómica).

- **A free copy of Mastering The Interview**: Indica si el usuario quiere una copia de la entrevista realizada (cualitativa nominal dicotómica).
- **Last Notable Activity**: Última actividad llevada a cabo por el estudiante (cualitativa nominal multicategórica).

Dada la información disponible, se han planteado las siguientes hipótesis iniciales que, mediante el uso de técnicas estadísticas y de minería de datos, pretenden ser contrastadas.

- El tiempo total empleado en la web de la página de la empresa no varía en función de si el *lead* compró finalmente alguno de los cursos ofrecidos.
- El hecho de que un *lead* compre o no alguno de los cursos es independiente del grado de interés mostrado en el formulario rellenado en el proceso de captación.
- Es posible predecir mediante un modelo de clasificación si un nuevo *lead* acabará comprando alguno de los cursos de la empresa en función de las características extraídas en el proceso de comunicación, ahorrando así un tiempo al departamente de marketing de la empresa para localizar a los potenciales usuarios.

2.2. Integración y selección de los datos de interés a analizar

Previamente a cualquier tipo de acción, y puesto que vamos a utilizar `R` como herramienta principal, realizaremos la carga de las librerías necesarias para el correcto desarrollo del caso práctico.

```
library(tidyverse)
library(DataExplorer)
library(caret)
library(VIM)
library(patchwork)
library(ggpubr)
library(FactoMineR)
library(factoextra)
library(pROC)
library(plotROC)
```

A su vez, fijaremos una semilla para obtener resultados reproducibles en la práctica.

```
set.seed(0) # Fijamos semilla para reproducibilidad de experimentos
```

Una vez cargadas las librerías, el siguiente paso de nuestro proyecto consiste en la carga de datos en el entorno para poder empezar a trabajar. En este punto, sería necesario llevar a cabo todos los procesos de **integración** de datos en caso de que se precisaran, con el objetivo de centralizar la información disponible en un único conjunto de datos. En nuestro caso práctico, puesto que toda la información se encuentra disponible en un único archivo, la fase de integración de datos es prescindible y no merece especial atención.

Aunque parece una fase trivial, es necesario prestar especial atención al proceso de lectura de datos en `R`, puesto que hay que tener en cuenta diferentes aspectos como por ejemplo el formato de las variables almacenadas.

Teniendo esto en cuenta, realizamos la carga del archivo a través de la función `read.csv()`. La información quedará guardada en un único `dataframe` de nombre `leads`. Como vemos, incluímos en la función el argumento `na.strings()`, que permite definir que valores del conjunto de datos deben ser interpretados en la carga de datos como valores perdidos. Este aspecto se encuentra detallado en el apartado 2.3.1. de la práctica.

```
leads <- read.csv("LeadScoring.csv",
                  sep = ";",
```

```
                  header = TRUE,
                  na.strings = c("", "Select", "Unknown"))
```

El siguiente paso es la revisión de las características generales del archivo y las variables correspondientes. Realizaremos esto a través de la función `summary()`, que nos proporcionará un resumen de las variables y nos informará del tipo de dato interpretado.

```
leads %>%
  summary()
```

```
##  Prospect.ID         Lead.Number     Lead.Origin        Lead.Source
##  Length:9240        Min.   :579533   Length:9240        Length:9240
##  Class :character   1st Qu.:596485   Class :character   Class :character
##  Mode  :character   Median :615479   Mode  :character   Mode  :character
##                     Mean   :617188
##                     3rd Qu.:637387
##                     Max.   :660737
##
##  Do.Not.Email       Do.Not.Call          Converted       TotalVisits
##  Length:9240        Length:9240        Min.   :0.0000   Min.   :  0.000
##  Class :character   Class :character   1st Qu.:0.0000   1st Qu.:  1.000
##  Mode  :character   Mode  :character   Median :0.0000   Median :  3.000
##                                        Mean   :0.3854   Mean   :  3.445
##                                        3rd Qu.:1.0000   3rd Qu.:  5.000
##                                        Max.   :1.0000   Max.   :251.000
##                                                         NA's   :137
##  Total.Time.Spent.on.Website Page.Views.Per.Visit Last.Activity
##  Min.   :   0.0              Min.   : 0.000       Length:9240
##  1st Qu.:  12.0              1st Qu.: 1.000       Class :character
##  Median : 248.0              Median : 2.000       Mode  :character
##  Mean   : 487.7              Mean   : 2.363
##  3rd Qu.: 936.0              3rd Qu.: 3.000
##  Max.   :2272.0              Max.   :55.000
##                              NA's   :137
##    Country          Specialization     How.did.you.hear.about.X.Education
##  Length:9240        Length:9240        Length:9240
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  What.is.your.current.occupation What.matters.most.to.you.in.choosing
##  Length:9240                     Length:9240
##  Class :character                Class :character
##  Mode  :character                Mode  :character
##
##
##
##
##     Search            Magazine         Newspaper.Article  X.Education.Forums
##  Length:9240        Length:9240        Length:9240        Length:9240
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

```
##
##
##
##
##  Newspaper         Digital.Advertisement Through.Recommendations
##  Length:9240       Length:9240           Length:9240
##  Class :character  Class :character      Class :character
##  Mode  :character  Mode  :character      Mode  :character
##
##
##
##
##  Receive.More.Updates.About.Our.Courses     Tags           Lead.Quality
##  Length:9240                            Length:9240        Length:9240
##  Class :character                       Class :character   Class :character
##  Mode  :character                       Mode  :character   Mode  :character
##
##
##
##
##  Update.me.on.Supply.Chain.Content Get.updates.on.DM.Content Lead.Profile
##  Length:9240                       Length:9240               Length:9240
##  Class :character                  Class :character          Class :character
##  Mode  :character                  Mode  :character          Mode  :character
##
##
##
##
##      City           Asymmetrique.Activity.Index Asymmetrique.Profile.Index
##  Length:9240        Length:9240                 Length:9240
##  Class :character   Class :character            Class :character
##  Mode  :character   Mode  :character            Mode  :character
##
##
##
##
##  Asymmetrique.Activity.Score Asymmetrique.Profile.Score
##  Min.   : 7.00               Min.   :11.00
##  1st Qu.:14.00               1st Qu.:15.00
##  Median :14.00               Median :16.00
##  Mean   :14.31               Mean   :16.34
##  3rd Qu.:15.00               3rd Qu.:18.00
##  Max.   :18.00               Max.   :20.00
##  NA's   :4218                NA's   :4218
##  I.agree.to.pay.the.amount.through.cheq A.free.copy.of.Mastering.The.Interview
##  Length:9240                            Length:9240
##  Class :character                       Class :character
##  Mode  :character                       Mode  :character
##
##
##
##
##  Last.Notable.Activity
##  Length:9240
```

```
##  Class :character
##  Mode  :character
##
##
##
##
```

Como vemos, todas las variables numéricas han sido interpretadas correctamente. Sin embargo, todas las demás variables, que deberían de haber sido interpretadas como `factor` (salvo `Lead.Number`, que se corresponde con una variable `character`), han sido interpretadas con un tipo de datos erróneo. Por tanto, realizamos la modificación correspondiente sobre estas variables y volvemos a mostrar el resumen.

```
well_interpreted <- c("Prospect.ID",
                      "TotalVisits",
                      "Total.Time.Spent.on.Website",
                      "Page.Views.Per.Visit",
                      "Asymmetrique.Activity.Score",
                      "Asymmetrique.Profile.Score")

leads <- leads %>%
  mutate_at(vars(-all_of(well_interpreted)), factor)

leads <- leads %>%
  mutate_at(vars(Lead.Number), as.character)

leads %>%
  summary()
```

```
##  Prospect.ID        Lead.Number                          Lead.Origin
##  Length:9240        Length:9240        API                     :3580
##  Class :character   Class :character   Landing Page Submission:4886
##  Mode  :character   Mode  :character   Lead Add Form           : 718
##                                        Lead Import             :  55
##                                        Quick Add Form          :   1
##
##
##          Lead.Source   Do.Not.Email Do.Not.Call Converted  TotalVisits
##  Google        :2868   No :8506     No :9238    0:5679    Min.   :  0.000
##  Direct Traffic:2543   Yes: 734     Yes:   2    1:3561    1st Qu.:  1.000
##  Olark Chat    :1755                                      Median :  3.000
##  Organic Search:1154                                      Mean   :  3.445
##  Reference     : 534                                      3rd Qu.:  5.000
##  (Other)       : 350                                      Max.   :251.000
##  NA's          :  36                                      NA's   :137
##  Total.Time.Spent.on.Website Page.Views.Per.Visit
##  Min.   :   0.0              Min.   : 0.000
##  1st Qu.:  12.0              1st Qu.: 1.000
##  Median : 248.0              Median : 2.000
##  Mean   : 487.7              Mean   : 2.363
##  3rd Qu.: 936.0              3rd Qu.: 3.000
##  Max.   :2272.0              Max.   :55.000
##                              NA's   :137
##                  Last.Activity                  Country
##  Email Opened           :3437   India              :6492
##  SMS Sent               :2745   United States      :  69
```

```
##  Olark Chat Conversation: 973   United Arab Emirates:  53
##  Page Visited on Website: 640   Singapore           :  24
##  Converted to Lead      : 428   Saudi Arabia        :  21
##  (Other)                : 914   (Other)             : 120
##  NA's                   : 103   NA's                :2461
##                    Specialization     How.did.you.hear.about.X.Education
##  Finance Management       : 976   Online Search        : 808
##  Human Resource Management: 848   Word Of Mouth        : 348
##  Marketing Management     : 838   Student of SomeSchool: 310
##  Operations Management    : 503   Other                : 186
##  Business Administration  : 403   Multiple Sources     : 152
##  (Other)                  :2292   (Other)              : 186
##  NA's                     :3380   NA's                 :7250
##      What.is.your.current.occupation
##  Businessman         :   8
##  Housewife           :  10
##  Other               :  16
##  Student             : 210
##  Unemployed          :5600
##  Working Professional: 706
##  NA's                :2690
##         What.matters.most.to.you.in.choosing Search     Magazine
##  Better Career Prospects  :6528              No :9226   No:9240
##  Flexibility & Convenience:   2              Yes:  14
##  Other                    :   1
##  NA's                     :2709
##
##
##
##  Newspaper.Article X.Education.Forums Newspaper  Digital.Advertisement
##  No :9238          No :9239           No :9239   No :9236
##  Yes:   2          Yes:   1           Yes:   1   Yes:   4
##
##
##
##
##
##  Through.Recommendations Receive.More.Updates.About.Our.Courses
##  No :9233                No:9240
##  Yes:   7
##
##
##
##
##
##                                    Tags                Lead.Quality
##  Will revert after reading the email:2072   High in Relevance: 637
##  Ringing                             :1203   Low in Relevance : 583
##  Interested in other courses         : 513   Might be         :1560
##  Already a student                   : 465   Not Sure         :1092
##  Closed by Horizzon                  : 358   Worst            : 601
##  (Other)                             :1276   NA's             :4767
##  NA's                                :3353
##  Update.me.on.Supply.Chain.Content Get.updates.on.DM.Content
```

```
##  No:9240                          No:9240
##
##
##
##
##
##
##                       Lead.Profile                            City
##  Dual Specialization Student:  20   Mumbai                     :3222
##  Lateral Student            :  24   Other Cities               : 686
##  Other Leads                : 487   Other Cities of Maharashtra: 457
##  Potential Lead             :1613   Other Metro Cities         : 380
##  Student of SomeSchool      : 241   Thane & Outskirts          : 752
##  NA's                       :6855   Tier II Cities             :  74
##                                     NA's                       :3669
##  Asymmetrique.Activity.Index Asymmetrique.Profile.Index
##  01.High  : 821              01.High  :2203
##  02.Medium:3839              02.Medium:2788
##  03.Low   : 362              03.Low   :  31
##  NA's     :4218              NA's     :4218
##
##
##
##  Asymmetrique.Activity.Score Asymmetrique.Profile.Score
##  Min.   : 7.00               Min.   :11.00
##  1st Qu.:14.00               1st Qu.:15.00
##  Median :14.00               Median :16.00
##  Mean   :14.31               Mean   :16.34
##  3rd Qu.:15.00               3rd Qu.:18.00
##  Max.   :18.00               Max.   :20.00
##  NA's   :4218                NA's   :4218
##  I.agree.to.pay.the.amount.through.cheq A.free.copy.of.Mastering.The.Interview
##  No:9240                                   No :6352
##                                            Yes:2888
##
##
##
##
##
##              Last.Notable.Activity
##  Modified               :3407
##  Email Opened           :2827
##  SMS Sent               :2172
##  Page Visited on Website: 318
##  Olark Chat Conversation: 183
##  Email Link Clicked     : 173
##  (Other)                : 160
```

Tras las modificaciones, todas las variables se encuentran formateadas con el tipo de dato correspondiente y ahora el resumen proporcionado por la función `summary()` puede ser interpretado de manera fiable.

Hecho esto, pasamos a realizar la **selección** de aquellas características de los *leads* que merezcan ser estudiadas y que nos proporcionen una fuente de información útil para resolver nuestro problema de negocio. Para llevar a cabo esta selección, en primer lugar prestaremos atención a la naturaleza de las variables recogidas, y excluiremos aquellas que no proporcionen ninguna información de interés.

Por ejemplo, las dos primeras variables presentan un valor único para cada registro del conjunto de datos, por lo que únicamente pueden utilizarse como identificadores. Como un único identificador es suficente, se elimina uno de ellos del conjunto de datos.

```
leads <- leads %>%
  select(-Prospect.ID)
```

Dado que uno de los objetivos del problema de negocio planteado es la construcciòn de un modelo de clasificación, no excluiremos ninguna de las variables restantes del estudio basándonos únicamente en su naturaleza y definición. En otras palabras, cada una de las variables es susceptible de aportar informacón útil a la hora de predecir si un *lead* finalmente comprará alguno de los cursos o no.

Por otra parte, existe la posibilidad de que por sus características numéricas o por su distribución, algunas variables no aporten información alguna y finalmente no sean utilizadas para ningún análisis. No obstante, esta fase de exploración y selección/exclusión de nuevas variables la llevaremos a cabo en la limpieza de los datos.

En ocasiones, puede ser interesante la definición en nuestro conjunto de datos de nuevas variables construídas en base a las que ya disponemos, en un proceso conocido como ***feature engineering***. Para nuestro caso práctico, y tras examinar el conjunto de datos y su contexto, hemos decidio incorporar una nueva variable que justificamos a continuación.

Como ya hemos comentado, algunas de las variables recogidas para cada uno de los registros correspondían a respuestas sobre un formulario proporcionado a cada uno de los *leads*. Existe la posibilidad de que el interés de los usuarios por los cursos ofrecidos pueda estar relacionado con la probabilidad de convertirse finalmente en un comprador. Una manera de medir dicho interés puede plantearse en función de la cantidad de respuestas personales que completó en el formulario.

En consecuencia, hemos decidido incorporar a nuestro conjunto de datos la variable `total_answered`, que define el número total de preguntas referentes al cuestionario contestadas por un usuario en concreto, y que tomará valores entre 0 (ninguna pregunta contestada) y 4 (todas las preguntas contestadas). Como estamos interpretando esta nueva variable como el grado de interés del usuario, la trataremos como una variable categórica ordinal (`factor` en `R`).

Para el cálculo, se han incluído las cuatro preguntas del cuestionario que, por su naturaleza, tenemos certeza de que correspondieron al cuestionario proporcionado al usuario y que no han sido eliminadas por nuestra parte en el análisis realizado a lo largo del proceso de limpieza.

```
leads$ans_specialization = !is.na(leads$Specialization)
leads$ans_how.hear = !is.na(leads$How.did.you.hear.about.X.Education)
leads$ans_occupation = !is.na(leads$What.is.your.current.occupation)
leads$ans_matters = !is.na(leads$What.matters.most.to.you.in.choosing)

leads <- leads %>%
  mutate(total_answered = ans_specialization + ans_how.hear + ans_occupation + ans_matters)

leads <- leads %>%
  mutate_at(vars(total_answered), factor)
```

Como vemos, en el proceso hemos creado adicionalmente cuatro nuevas variables intermedias que definen si el usuario en cuestión contestó o no cada una de las cuatro preguntas del cuestionario. Puesto que nuestro objetivo es evaluar el interés de manera global en función del número total de preguntas contestadas, excluímos estas cuatro variables de nuestro conjunto de datos final.

```
leads <- leads %>%
  select(-starts_with("ans"))
```

Otra opción puede ser prescindir de aquellas variables en las que variabilidad es prácticamente nula. Por ejemplo, podemos observar que para la variable `Newspaper`, unicamente un registro presenta el valor de `1`,

mientras que los 9239 restantes presentan el valor `0`. Esta ausencia de varianza hace que dicha variable pierda el sentido de ser analizada. Al no haber variabilidad, no presenta ningún tipo de efecto en un modelo predictivo ni puede ser utilizada en una prueba estadística.

En consecuencia, todas las variables cuya variabilidad sea igual o muy próxima a 0 serán excluidas del estudio. No obstante, es posible que alguna variable presente una falsa variabilidad alta debido a que algún valor debería haber sido interpretado como nulo o perdido. Por ello, será necesario previamente identificar qué valores son susceptibles de ser interpretados como valores perdidos o nulos. Lo veremos en el apartado a continuación.

Por útlimo, nos parece necesario recalcar que **no se llevará a cabo ninguna reducción de la cantidad de registros** de nuestro conjunto de datos. Entendemos que presenta un tamaño lo suficientemente grande como para construir un modelo de clasificación que pueda ser validado con un volumen de datos adecuado y lo suficientemente pequeño como para que los cálculos computacionales (imputación de valores perdidos, entrenamiento de algoritmos) no se demoren demasiado en el tiempo.

2.3. Limpieza de datos

2.3.1. Identificación y tratamiento de valores nulos o perdidos

Tal y como decíamos en el aparto anterior, existe la posibilidad de que alguna variable tome algún valor que, por el contexto y lo que representa dicho valor, deba ser interpretado como `NA` (valor perdido). Tras realizar una exploración básica del conjunto de datos (y basándonos además en el resultado de la función `summary()` anterior), hemos observado algunos valores que se deben ser identificados como valores nulos o desconocidos para las variables en cuestión.

- **Casilla vacía (`""`)**: Esta es la forma más generalizada a la hora de indicar la ausencia de un dato en nuestro conjunto de datos.
- `Unknown`: En el caso de la variable `Country`, existe esta opción entre las posibilidades seleccionadas, indicando que el país del *lead* no se conoce.
- `Select`: Esta opción está presente en algunas de las variables como `Specialization`, `How.did.you hear.about.X.Education`, `Lead Profile` o `City`. Parece que este valor aparece en los casos en los que el *lead* dejó la opción por defecto a la hora de responder, indicando por tanto que no contestó a dicha cuestión.

Por tanto, lo primero que debemos hacer es unificar la forma en la que aparecen los registros desconocidos en el conjunto de datos. En este sentido, a la hora de cargar el conjunto de datos con la función `read.csv()`, hemos incluido un argumento adicional especificando qué valores del conjunto de datos debían ser interpretados como valores nulos, tal y como mencionábamos en la carga de datos.

Antes de discutir la manera de tratar los valores nulos de las demás variables, parece adecuado estudiar la variabilidad de las demás variables del conjunto de datos, tal y como comentábamos en el apartado anterior de la práctica. Para ello, utilizamos la función `nearZeroVar()`, de la librería `caret`, que permite analizar de manera automática qué variables presentan una varianza igual a 0 o muy próxima a dicho valor.

```
near_zero_var <- nearZeroVar(leads, saveMetrics = TRUE)
near_zero_var
```

```
##                                  freqRatio percentUnique zeroVar   nzv
## Lead.Number                       1.000000  100.00000000   FALSE FALSE
## Lead.Origin                       1.364804    0.05411255   FALSE FALSE
## Lead.Source                       1.127802    0.22727273   FALSE FALSE
## Do.Not.Email                     11.588556    0.02164502   FALSE FALSE
## Do.Not.Call                    4619.000000    0.02164502   FALSE  TRUE
## Converted                         1.594777    0.02164502   FALSE FALSE
## TotalVisits                       1.302976    0.44372294   FALSE FALSE
## Total.Time.Spent.on.Website     115.421053   18.73376623   FALSE FALSE
```

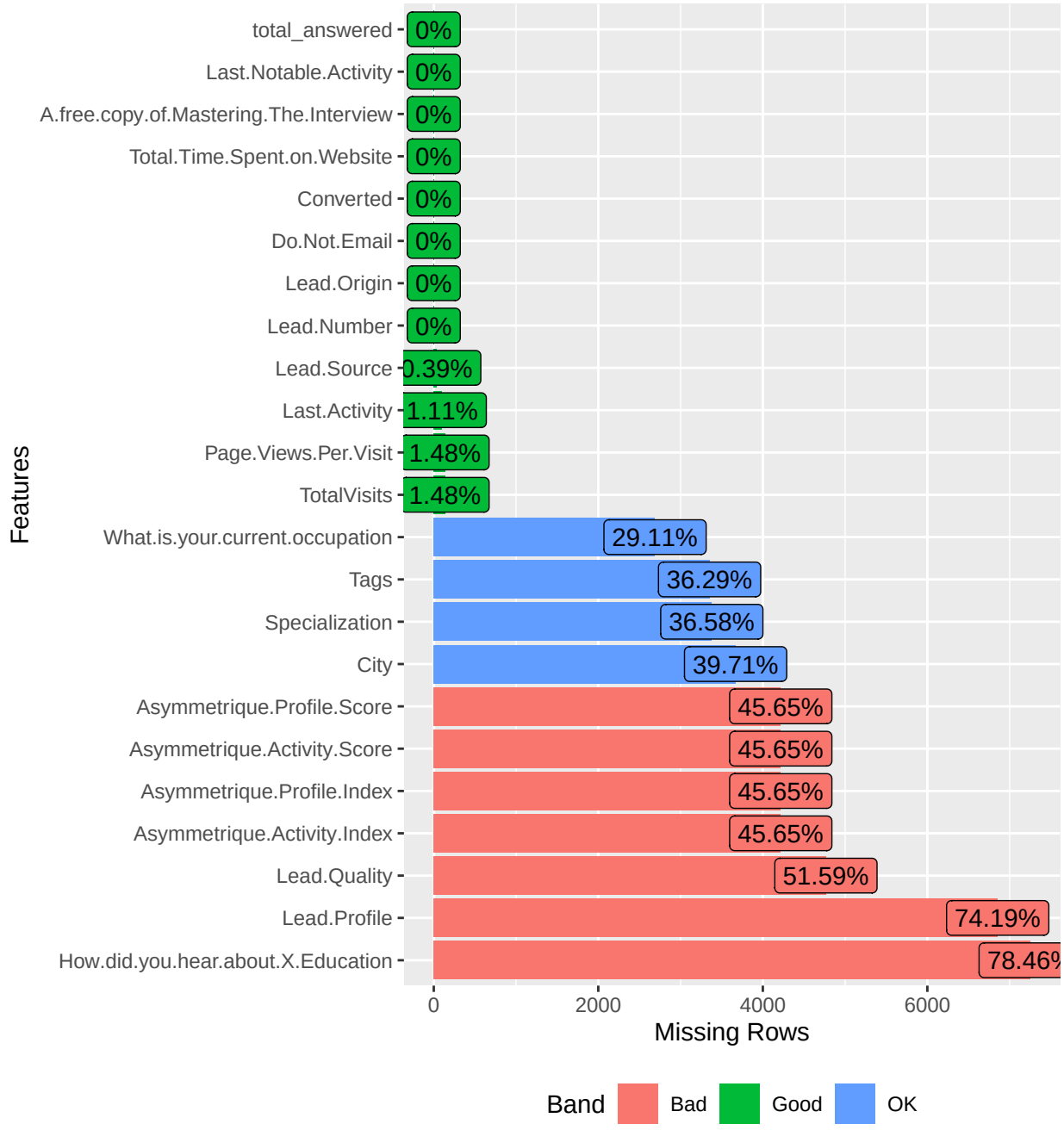
```
## Page.Views.Per.Visit                        1.219499    1.23376623   FALSE FALSE
## Last.Activity                               1.252095    0.18398268   FALSE FALSE
## Country                                    94.086957    0.41125541   FALSE  TRUE
## Specialization                              1.150943    0.19480519   FALSE FALSE
## How.did.you.hear.about.X.Education          2.321839    0.09740260   FALSE FALSE
## What.is.your.current.occupation             7.932011    0.06493506   FALSE FALSE
## What.matters.most.to.you.in.choosing     3264.000000    0.03246753   FALSE  TRUE
## Search                                    659.000000    0.02164502   FALSE  TRUE
## Magazine                                    0.000000    0.01082251    TRUE  TRUE
## Newspaper.Article                        4619.000000    0.02164502   FALSE  TRUE
## X.Education.Forums                       9239.000000    0.02164502   FALSE  TRUE
## Newspaper                                9239.000000    0.02164502   FALSE  TRUE
## Digital.Advertisement                    2309.000000    0.02164502   FALSE  TRUE
## Through.Recommendations                  1319.000000    0.02164502   FALSE  TRUE
## Receive.More.Updates.About.Our.Courses      0.000000    0.01082251    TRUE  TRUE
## Tags                                        1.722361    0.28138528   FALSE FALSE
## Lead.Quality                                1.428571    0.05411255   FALSE FALSE
## Update.me.on.Supply.Chain.Content           0.000000    0.01082251    TRUE  TRUE
## Get.updates.on.DM.Content                   0.000000    0.01082251    TRUE  TRUE
## Lead.Profile                                3.312115    0.05411255   FALSE FALSE
## City                                        4.284574    0.06493506   FALSE FALSE
## Asymmetrique.Activity.Index                 4.676005    0.03246753   FALSE FALSE
## Asymmetrique.Profile.Index                  1.265547    0.03246753   FALSE FALSE
## Asymmetrique.Activity.Score                 1.369683    0.12987013   FALSE FALSE
## Asymmetrique.Profile.Score                  1.642390    0.10822511   FALSE FALSE
## I.agree.to.pay.the.amount.through.cheq      0.000000    0.01082251    TRUE  TRUE
## A.free.copy.of.Mastering.The.Interview      2.199446    0.02164502   FALSE FALSE
## Last.Notable.Activity                       1.205164    0.17316017   FALSE FALSE
## total_answered                              1.326360    0.05411255   FALSE FALSE
```

Como vemos, un gran número de variables presentan una varianza muy próxima (en muchos casos igual) a 0. En la práctica, estas variables no nos serán muy útiles para nuestros análisis y a la hora de extraer conclusiones bien fundamentadas, por lo que decidimos excluirlas de nuestro conjunto de datos. Se eliminarán por tanto aquellas variables que en el análisis anterior tienen el valor `TRUE` en `nzv`.

```
near_zero_var <- nearZeroVar(leads)
leads <- leads[, -near_zero_var]
```

A continuación, parece adecuado conocer el porcentaje de `NA` presente en cada una de las variables que no han sido eliminadas en los procesos previos del conjunto de datos. Para ello, utilizamos la función `plot_missing()` de la librería `DataExplorer`.

```
leads %>%
  plot_missing()
```



Como vemos en el gráfico anterior, el conjunto de datos, tras eliminar las variables sin variabilidad, presenta **15 variables con valores perdidos o nulos**. No obstante, el porcentaje de este tipo de valores no es uniforme en cada una de ellas, y en función de la cantidad de valores perdidos, cada variable deberá ser tratada de una forma en particular.

Para las variables marcadas en rojo, el porcentaje de valores perdidos es tan grande (en el mejor de los casos de casi la mitad de los registros) que realizar una imputación de los mismos conlleva el riesgo de falsear los datos de manera significativa y, en consecuencia, las conclusiones extraídas de sus análisis. Por ello, parece razonable no utilizar estas variables para nuestros análisis y modelos, por lo que las excluimos de nuestros conjunto de datos.

```
leads <- leads %>%
  select(-c(How.did.you.hear.about.X.Education,
            Lead.Profile,
            Lead.Quality,
            Asymmetrique.Activity.Index,
            Asymmetrique.Profile.Index,
            Asymmetrique.Activity.Score,
            Asymmetrique.Profile.Score))
```

Para las demás variables, hemos llevado a cabo una evaluación exhaustiva en cada una de ellas para decidir la mejor manera de proceder para solucionar la presencia de valores nulos. Tras una exploración básica del conjunto de datos, y teniendo en cuenta la variabilidad de cada una de las variables, hemos tomado las siguientes decisiones.

- `City`. Esta variable cualitativa presenta un porcentaje muy alto de valores nulos. Cabe mencionar que hay una categoría, que corresponde a la ciudad de Mumbay, mucho más presente que todas las demás. Con el objetivo de utilizar la mayor cantidad de información posible, realizaremos una imputación de valores perdidos mediante el algoritmo `KNN` en base a las demás variables.
- `Specialization`. A diferencia de la variable anterior, en este caso la distribución se presenta mucho más equilibrada entre categorías. Por otra parte, dado el número tan alto de categorías que presenta, junto con el alto porcentaje de valores nulos, se plantea difícil realizar una imputación de estos (sea por el método que sea) sin que dicha acción conlleve introducir información falsa en buena parte de los registros. Por tanto, hemos decidido prescindir de dicha variable para las fases posteriores del estudio.

```
leads <- leads %>%
  select(-Specialization)
```

- `Tags`. Esta variable presenta unas características similares a la anterior: un porcentaje de valores nulos significativo, muchas categorías y, además, una variabilidad alta que dificulta la imputación. En consecuencia, también hemos decidio excluir esta variable del estudio.

```
leads <- leads %>%
  select(-Tags)
```

- `Whats.is.your.current.occupation`. En este caso, el porcentaje de valores perdidos es bastante menor que en los casos anteriores. Además, el número de categorías es bastante inferior que, por ejemplo, el caso de la variable `Tags`. Por ello, y con el objetivo de perder la menor cantidad información posible presente en el conjunto de datos, hemos decidido utilizar para la imputación de valores perdidos el algoritmo `KNN`.
- `TotalVisits`, `Page.Views.Per.Visit`, `Last.Activity`, `Lead.Source`. Para estas cuatro variables, el porcentaje de valores nulos es ínfimo (próximo al 1%). Por ello, parece razonable imputar los pocos valores nulos que presentan con el método de `KNN` y perder la menor cantidad de información posible.

Por tanto, en total realizaremos la imputación de cinco variables a través de la función `kNN()` del paquete `VIM`. Para ello vamos a asumir que existe algún tipo de correlacción/asociación entre las variables del conjunto de datos de manera que se asignará un valor aproximado a los `NA` en función del valor del resto de variables. La única columna que se elimina para la imputación es `Lead.Number`, ya que al ser un identificador único del *lead* no aportaría ninguna información en el proceso de imputación.

Para nuestro conjunto de datos actual, con un número sustancialmente alto de registros, parece adecuado utilizar el valor por defecto de la función `kNN()`, de manera que se utilizarán los 5 vecinos más cercanos al registro sobre el cual se vaya a realizar la imputación para predecir el valor de la variable en cuestión. También cabe mencionar que, puesto que nuestro conjunto de datos presenta tanto variables numéricas como categóricas, parece adecuado utilizar la distancia por defecto, es decir, la distancia de Gower, que permite calcular distancias entre individuos que presentan ambos tipos de variables. Además, por como está definida la distancia, no será neceario normalizar las variables continuas involucradas en el cálculo [1].

```
imputation_vars <- c("City",
                     "Lead.Source",
                     "TotalVisits",
                     "Page.Views.Per.Visit",
                     "Last.Activity",
                     "What.is.your.current.occupation")

# Imputación de valores nulos con KNN
leads <- leads %>%
  kNN(variable = imputation_vars,
      dist_var = colnames(leads[2:14]),
      imp_var = FALSE)
```

Tras la imputación, podemos comprobar que ninguna de las variables presenta valores perdidos.

```
colSums(is.na(leads))
```

```
##                             Lead.Number                             Lead.Origin
##                                       0                                       0
##                             Lead.Source                            Do.Not.Email
##                                       0                                       0
##                               Converted                            TotalVisits
##                                       0                                       0
##           Total.Time.Spent.on.Website                    Page.Views.Per.Visit
##                                       0                                       0
##                           Last.Activity          What.is.your.current.occupation
##                                       0                                       0
##                                    City A.free.copy.of.Mastering.The.Interview
##                                       0                                       0
##                  Last.Notable.Activity                           total_answered
##                                       0                                       0
```

2.3.2 Identificación y tratamiento de los valores extremos

Los valores extremos de un conjunto de datos son aquellos que se encuentran muy alejados de la distribución general de una variable. Su presencia, en caso de no ser valores legítimos, puede afectar a los resultados obtenidos en los análisis posteriores, por lo que se requiere una evaluación individualizada de estos para poder identificarlos y decidir el tratamiento que se realizará.

En nuestro caso, evaluaremos la presencia de valores extremos en las tres variables numéricas del conjunto de datos con la función `boxplots.stats()` de `R`, que utiliza como criterio para considerar valores extremos a todos aquellos que no queden definidos en el rango establecido por los *bigotes* del *boxplot*.

- Número de valores extremos en la variable `Total.Time.Spent.on.Website`.

```
length(boxplot.stats(leads$Total.Time.Spent.on.Website)$out)
```

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

- Número de valores extremos en la variable `TotalVisits`.

```
length(boxplot.stats(leads$TotalVisits)$out)
```

```
## [1] 267
```

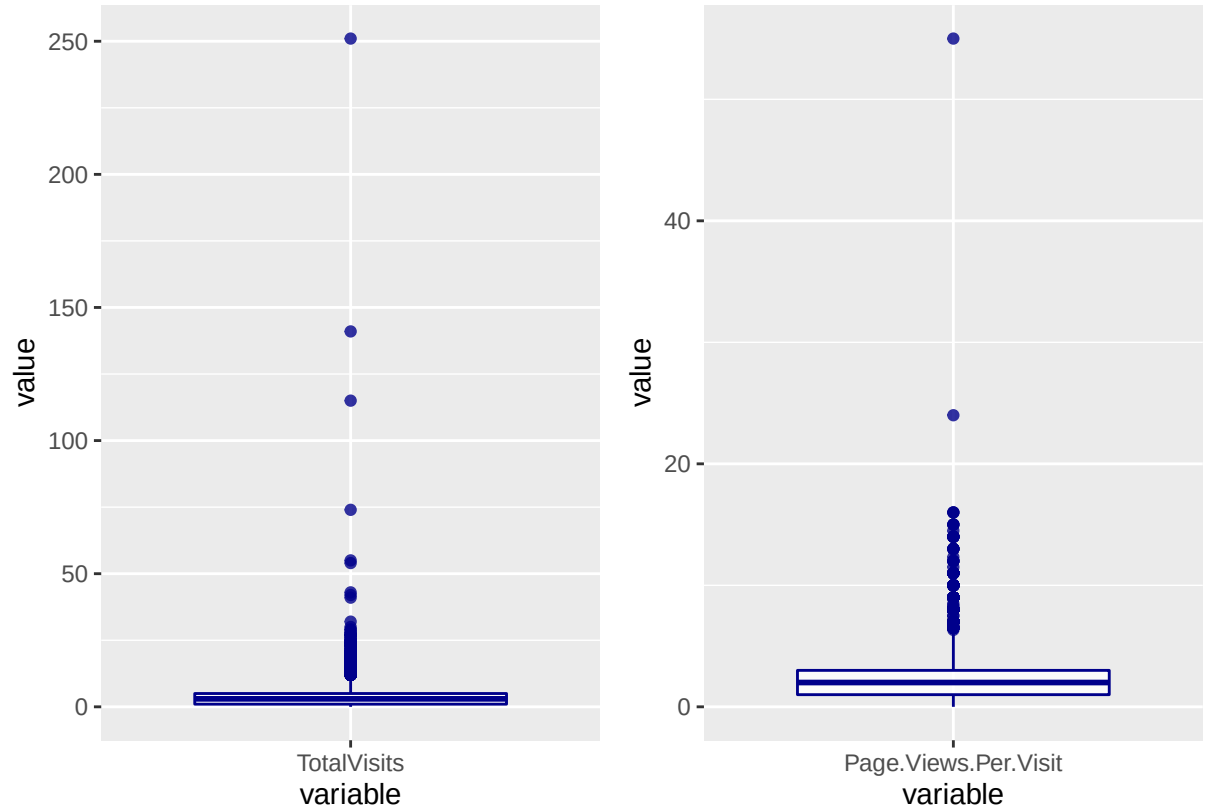
- Número de valores extremos en la variable `TotalVisits`.

```
length(boxplot.stats(leads$Page.Views.Per.Visit)$out)
```

```
## [1] 360
```

Podemos observar que en dos de las variables presenciamos la existencia de valores extremos. Para estudiar la validez y legitimidad de los valores extremos, puede ser útil conocer qué valores toman. Para ello, dibujamos los *boxplots* asociados a ambas variables.

```
vars = c("TotalVisits", "Page.Views.Per.Visit")

# Dataset para facilitar la construcción de gráficos
boxplot_data <- leads %>%
  gather(unlist(vars), key = "variable", value = "value")

bp1 <- boxplot_data %>%
  filter(variable == "TotalVisits") %>%
  ggplot(boxplot_data, mapping = aes(x = variable, y = value)) +
  geom_boxplot(col = "darkblue", alpha = 0.8)

bp2 <- boxplot_data %>%
  filter(variable == "Page.Views.Per.Visit") %>%
  ggplot(boxplot_data, mapping = aes(x = variable, y = value)) +
  geom_boxplot(col = "darkblue", alpha = 0.8)

# Composición de figura final con ambos gráficos
bp1 + bp2
```



Tras observar qué valores toman los valores extremos, se llega a la conclusión de que dichos valores pueden considerarse legítimos de la distribución (muy relacionado con el hecho de la no normalidad de la distribución

que veremos en el próximo apartado) y que pertenecen al dominio de las variables, por lo que el tratamiento decidido fue mantenerlos en la conjunto de datos.

El conjunto de datos definitivo para comenzar a explotar la información mediante técnicas y modelos analíticos presenta la siguiente dimensión.

```
dim(leads)
```

```
## [1] 9240   14
```

Como vemos, mantenemos los 9240 registros iniciales del conjunto de datos original, mientras que por otra parte hemos reducido el número total de variables de 37 a 14.

Procedemos por último a generar un archivo '.csv donde guardaremos el conjunto de datos generado tras los distintos procesos de limpieza y selección realizados en este apartado, nombrándolo como `LeadScoring_clean.csv`.

```
# Creación del archivo con los datos definitivos tras el proceso de limpieza y selección:
write.csv(leads, "LeadScoring_clean.csv")
```

Dicho archivo se ecnocntrará generado en el directorio de trabajo en el que se encuentre el presente documento.

2.4. Análisis de los datos

2.4.1. Selección de los grupos de datos que se quieren analizar/comparar

Puesto que estamos interesados en conocer qué información puede estar asociada con el hecho de que un *lead* se convierta o no en un comprador de cursos, únicamente crearemos los grupos definidos por la variable `Converted`.

```
converted <- leads %>%
  filter(Converted == "1")

non_converted  <- leads %>%
  filter(Converted == "0")
```

Además de esta selección, se ha llevado a cabo una exclusión de las diferentes variables que por sus propiedades numéricas y/o su distribución no proporcionaban información para los análisis posteriores (apartado 2.3.1.).

2.4.2. Comprobación de la normalidad y homgeneidad de la varianza

Para la resolución de este caso práctico, el conjunto de datos escogido se caracteriza por presentar un número muy reducido de variables numéricas a la hora de estudiar si un *lead* contrató alguno de los cursos. A pesar de esto, para las variables numéricas presentadas, se estudiará si dichas variables se distribuyen segun la distribución normal y si su varianza es constante con el objetivo de poder elegir el test estadístico más adecuado en cada caso.

Normalidad

Primeramente, vamos a verificar si las variables numéricas del conjunto de datos siguen o no una distribución normal en cada uno de los grupos definidos anteriormente. Usaremos el test de *Kolmogorov-Smirnov*, ya que el de *Shapiro-Wilk* sólo es posible utilizarlo con muestras de un tamaño inferior a 5000.

- Normalidad de la variable `TotalVisits` por grupos.

```
ks.test(converted$TotalVisits, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  converted$TotalVisits
```

```
## D = 0.67565, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
ks.test(non_converted$TotalVisits, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  non_converted$TotalVisits
## D = 0.68882, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

- Normalidad de la variable `Total.Time.Spent.on.Website` por grupos.

```
ks.test(converted$Total.Time.Spent.on.Website, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  converted$Total.Time.Spent.on.Website
## D = 0.74108, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
ks.test(non_converted$Total.Time.Spent.on.Website, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  non_converted$Total.Time.Spent.on.Website
## D = 0.77088, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

- Normalidad de la variable `Page.Views.Per.Visit` por grupos.

```
ks.test(converted$Page.Views.Per.Visit, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  converted$Page.Views.Per.Visit
## D = 0.58301, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

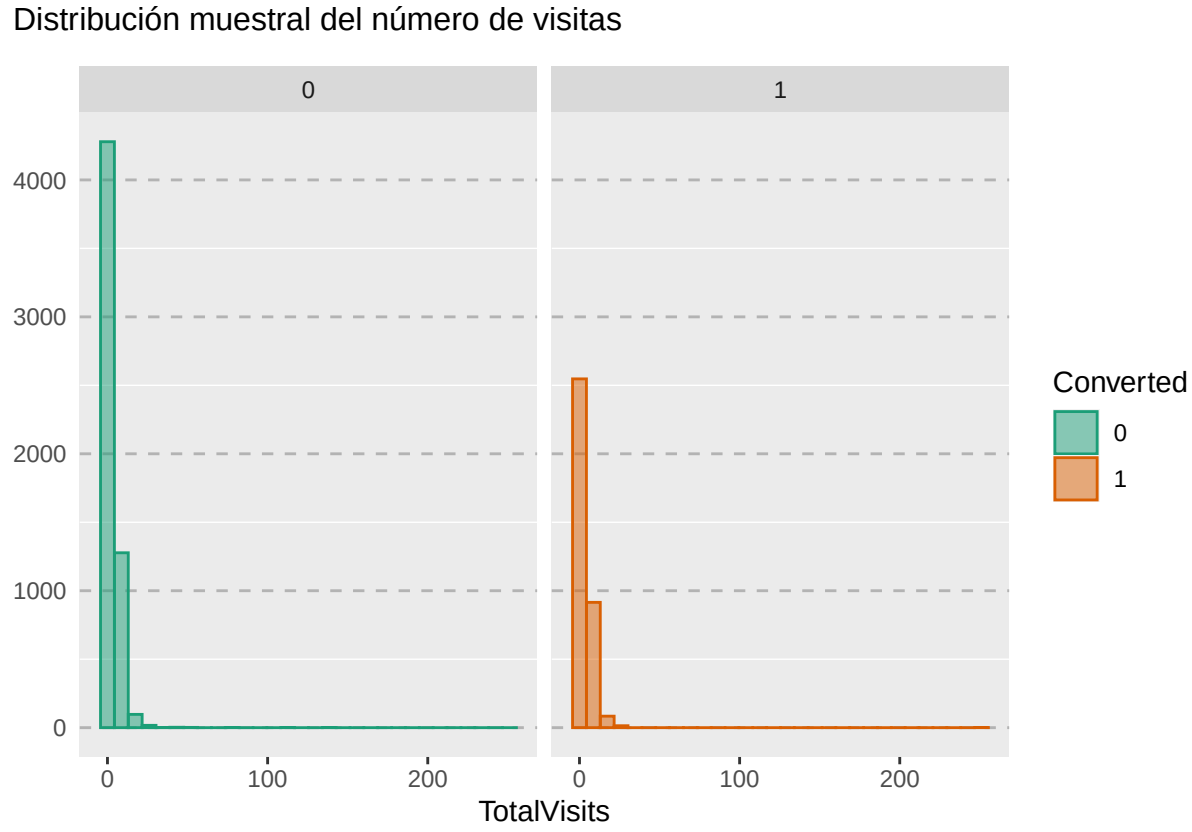
```
ks.test(non_converted$Page.Views.Per.Visit, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  non_converted$Page.Views.Per.Visit
## D = 0.61208, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

Como se observa en todos los casos, el p-valor reportado se encuentra muy por debajo del valor de significación de $\alpha = 0.05$. Por este motivo, podemos concluir que las variables en ambos grupos no se distribuyen segun una distribución normal. Este hecho podemos verificarlo a su vez de manera gráficamente a través del histograma de las variables:

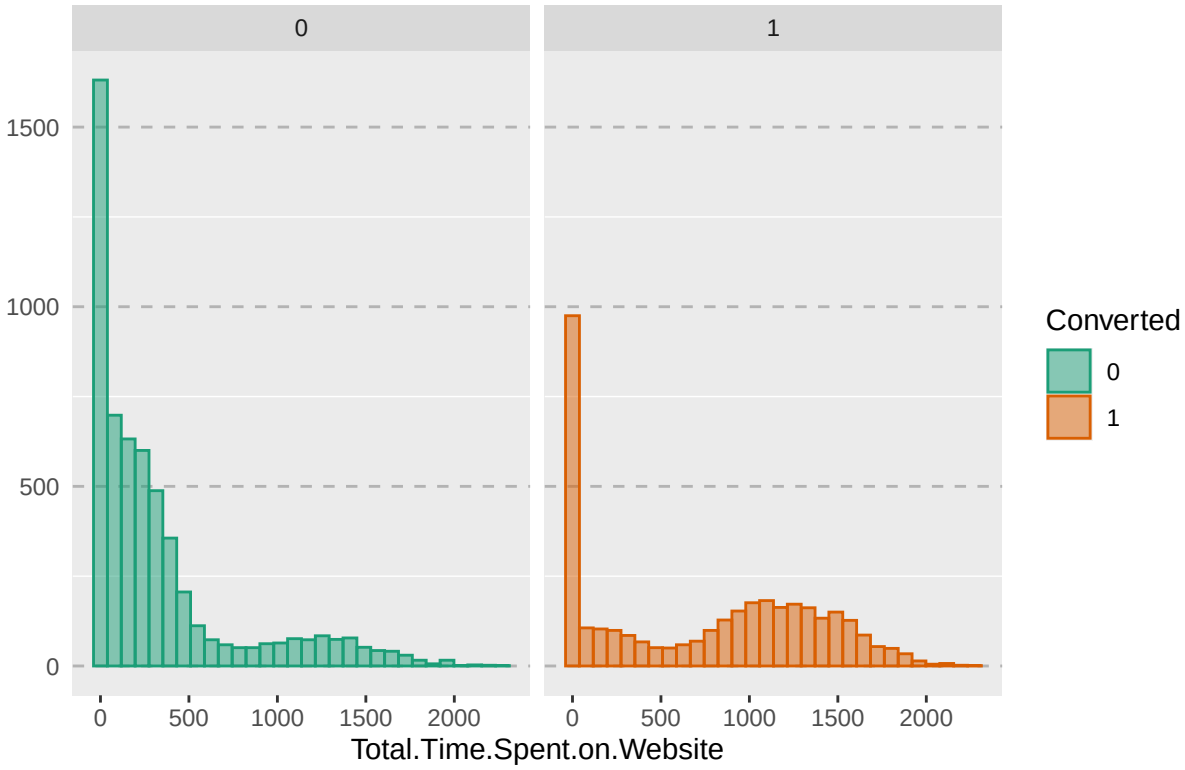
```
p1 <- ggplot(data = leads) +
  aes(x = TotalVisits, color = Converted, fill = Converted) +
```

```
  geom_histogram(alpha = 0.5, position = "identity", bins = 30) +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2") +
  theme_cleveland()

p1 + facet_grid(cols = vars(Converted)) +
  plot_annotation("Distribución muestral del número de visitas",
  theme = theme(plot.title = element_text(size = 12)))
```

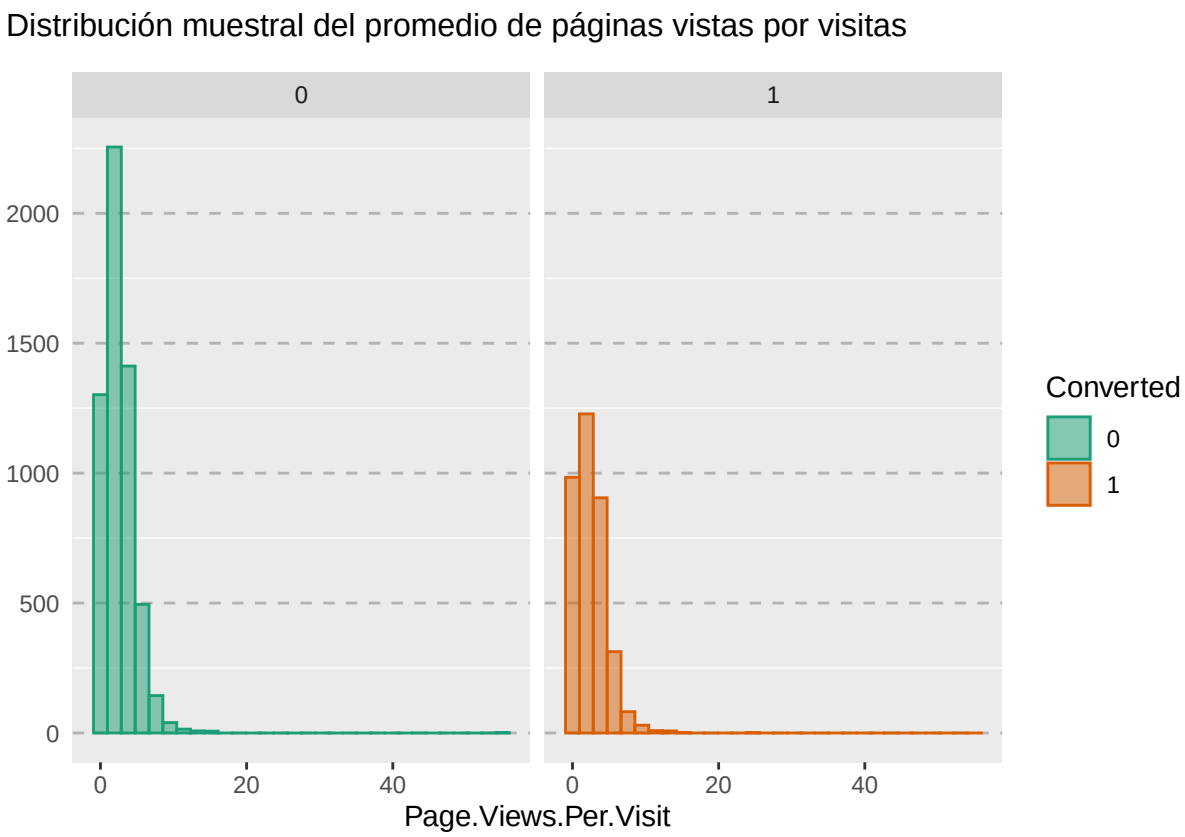



```
p2 <- ggplot(data = leads) +
  aes(x = Total.Time.Spent.on.Website, color = Converted, fill = Converted) +
  geom_histogram(alpha = 0.5, position = "identity", bins = 30) +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2") +
  theme_cleveland()

p2 + facet_grid(cols = vars(Converted)) +
  plot_annotation("Distribución muestral del tiempo total empleado en la web",
  theme = theme(plot.title = element_text(size = 12)))
```

Distribución muestral del tiempo total empleado en la web




```
p3 <- ggplot(data = leads) +
  aes(x = Page.Views.Per.Visit, color = Converted, fill = Converted) +
  geom_histogram(alpha = 0.5, position = "identity", bins = 30) +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2") +
  theme_cleveland()

p3 + facet_grid(cols = vars(Converted)) +
  plot_annotation("Distribución muestral del promedio de páginas vistas por visitas",
  theme = theme(plot.title = element_text(size = 12)))
```



Como vemos, en todos los casos nos encontramos con distribuciones con grandes colas a la derecha, donde los valores más pequeños son los que se observan con mayor frecuencia. Tendremos en cuenta dicha ausencia de normalidad para los análisis posteriores.

Cabe mencionar que, en caso de que uno de los objetivos del caso práctico fuera utilizar de manera globlal las variables numéricas para algún análisis en particular (p.e. analizar la correlación entre dos de las variables numéricas), sería necesario contrastar la normalidad de las mismas sin tener en cuenta la división en grupos. De manera ilustrativa, contrastaremos además la normalidad global de las variables numéricas de manera análoga a cómo lo hemos hecho antes.

```
ks.test(leads$TotalVisits, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  leads$TotalVisits
## D = 0.68374, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
ks.test(leads$Total.Time.Spent.on.Website, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  leads$Total.Time.Spent.on.Website
## D = 0.75938, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```r
ks.test(leads$Page.Views.Per.Visit, pnorm)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  leads$Page.Views.Per.Visit
## D = 0.59824, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

Como vemos, tenemos evidencias para afirmar que las variables numéricas no se ditribuyen globalmente segun una normal puesto que los p-valores reportados no son superiores a la significación habitual $\alpha = 0.05$.

Homocedasticidad

En relación a la igualdad de varianzas, utilizaremos el test de Fligner-Killen, útil para nuestro caso puesto que hemos verificado la ausencia de normalidad en nuestras variables. Utilizaremos el mismo nivel de significación, y en caso de obtener p-valores menores a 0.05 podremos concluir que existen evidencias para rechazar la hipótesis nula de varianzas iguales (homocedasticidad).

- Homocedasticidad de la variable `TotalVisits`.

```r
fligner.test(TotalVisits ~ Converted, data = leads)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  TotalVisits by Converted
## Fligner-Killeen:med chi-squared = 40.832, df = 1, p-value = 1.659e-10
```

- Homocedasticidad de la variable `Total.Time.Spent.on.Website`.

```r
fligner.test(Total.Time.Spent.on.Website ~ Converted, data = leads)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Total.Time.Spent.on.Website by Converted
## Fligner-Killeen:med chi-squared = 1076.2, df = 1, p-value < 2.2e-16
```

- Homocedasticidad de la variable `Page.Views.Per.Visit`.

```r
fligner.test(Page.Views.Per.Visit ~ Converted, data = leads)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Page.Views.Per.Visit by Converted
## Fligner-Killeen:med chi-squared = 11.037, df = 1, p-value = 0.0008933
```

Como podemos observar en los p-valores reportados, en todos los casos tenemos valores inferiores a 0.05, por lo que se concluye que las varianzas no son homogéneas entre los grupos definidos para ninguna de las variables numéricas analizadas.

En consecuencia, en caso de realizar algún test estadístico, habría que tener en cuenta dichas características. En nuestro caso, como únicamente buscamos contrastar la diferencia por grupos en el tiempo total empleado en la página web, abordaremos la aplicación de test estadísticos sobre dicha variable, tal y como veremos en el apartado a continuación.

2.4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos

Comparación entre grupos del tiempo total empleado en la web

La primera hipótesis que planteábamos al inicio de la práctica era si el tiempo total empleado en la web de *X Education* era el mismo indendientemente de si el *lead* compró finalmente alguno de los cursos. Para resolver esta cuestión, planteamos un contraste de hipótesis estadístico de comparación de dos muestras independientes.

Para ello, denotamos formalmente:

$$X_1 = \text{Tiempo total empleado en la web de usuarios convertidos}$$
$$X_2 = \text{Tiempo total empleado en la web de usuarios no convertidos}$$

Antes de plantear el contraste de hipótesis, es necesario especificar si dicho contraste será paramétrico o no parámetrico. Como ya hemos visto, la variable numérica a estudiar en ambos grupos presentaba evidencias para rechazar la hipótesis de normalidad. No obstante, dado el gran tamaño muestral que disponemos para ambos grupos, podemos asumir por el Teorema del Límite Central que:

$$\overline{X}_1 \sim N(\mu_1, \frac{\sigma_1}{\sqrt{n_1}}) \quad \text{y} \quad \overline{X}_2 \sim N(\mu_2, \frac{\sigma_2}{\sqrt{n_2}})$$

El contraste de hipótesis planteado es el siguiente:

$$H_0 : \mu_1 = \mu_2$$
$$H_1 : \mu_1 \neq \mu_2$$

En consecuencia, y dado que adicionalmente hemos contrastado la ausencia de varianzas iguales, parece adecuado utilizar el test de Welch, un caso particular del test T para varianzas distintas (y desconocidas).

```
t.test(converted$Total.Time.Spent.on.Website, non_converted$Total.Time.Spent.on.Website)
```

```
##
##  Welch Two Sample t-test
##
## data:  converted$Total.Time.Spent.on.Website and non_converted$Total.Time.Spent.on.Website
## t = 34.576, df = 5755.9, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  385.0018 431.2828
## sample estimates:
## mean of x mean of y
##  738.5468  330.4045
```

Como podemos observar del resultado del contraste de hipótesis, el p-valor asociado es menor que la significación habitual $\alpha = 0.05$. Por tanto, podemos afirmar que existen evidencias suficientes para afirmar que el tiempo total empleado en la web es diferente en función de si el *lead* compró alguno de los cursos o no.

Asociación entre el grado de interés y si el *lead* se convirtió

Por otra parte, se requiere estudiar si el grado de interés que el *lead* mostró en el formulario proporcionado presenta alguna asociación con el hecho de que dicho *lead* finalmente comprara alguno de los cursos. Para contrastar dicha hipótesis, plantamos un test chi-cuadrado.

Dicho contraste se plantea de la siguiente manera:

H_0 : *Convertirse en alumno es independiente del grado de interés al completar el cuestionario*

H_1 : *Convertirse en alumno es dependiente del grado de interés al completar el cuestionario*

Para realizar el test estadísctico, construímos primeramente la tabla de contigencia entre las variables `Converted` y `total_answered`.

```
# Tabla de contingencia
table_lead <- table(leads$total_answered, leads$Converted)
table_lead
```

```
##
##        0    1
##   0 1268  151
##   1  668  133
##   2 1492  898
##   3 1546 1624
##   4  705  755
```

Realizamos el test chi-cuadrado a través de la función `chisq-test`, que compara las frecuencias observadas con las frecuencias que en teoría se deberían de obtener si ambas variables fueran independientes.

```
chisq.test(table_lead)
```

```
##
##  Pearson's Chi-squared test
##
## data:  table_lead
## X-squared = 952.42, df = 4, p-value < 2.2e-16
```

Como vemos, el p-valor asocidado al contraste de hipótesis está por debajo del valor de significación habitual de $\alpha = 0.05$. En consecuencia, rechazamos la hipótesis nula, y concluímos que convertirse en alumno sí depende del grado de interés mostrado al completar las preguntas del formulario.

Una vez contrastada la dependencia entre ambas variables, podemos estar interesados en conocer cómo es dicha asociación y explorar analíticamente la relación descubierta. Para ello, existen técnicas factoriales específicas como el Análisis Factorial de Correspondencias (AFC) [2].

En concreto, el AFC es una técnica factorial multivariante muy conocida y que podría considerse como una extensión del Análisis de Componentes Principales para variables cualitativas. En concreto, este método extrae un espacio de dimensión reducida en el que se representan las categorías de las variables cualitativas involucradas en el análisis. El procedimiento habitual consiste en extraer el primer plano factorial e interpretar la proximidad de las categorías en el espacio. Aquellas categorías más próximas en el plano se interpretarán con una asociación positiva, mientras que las que más se distancien se interpretarán con una asociación negativa.

Para realizar el AFC en `R`, utilizaremos las funciones disponibles en los paquetes `FactoMineR` y `factoextra`, habituales en este tipo de análisis. En primer lugar, realizamos el análisis y lo almacenamos en un objeto que contenga toda la información del procedimiento.

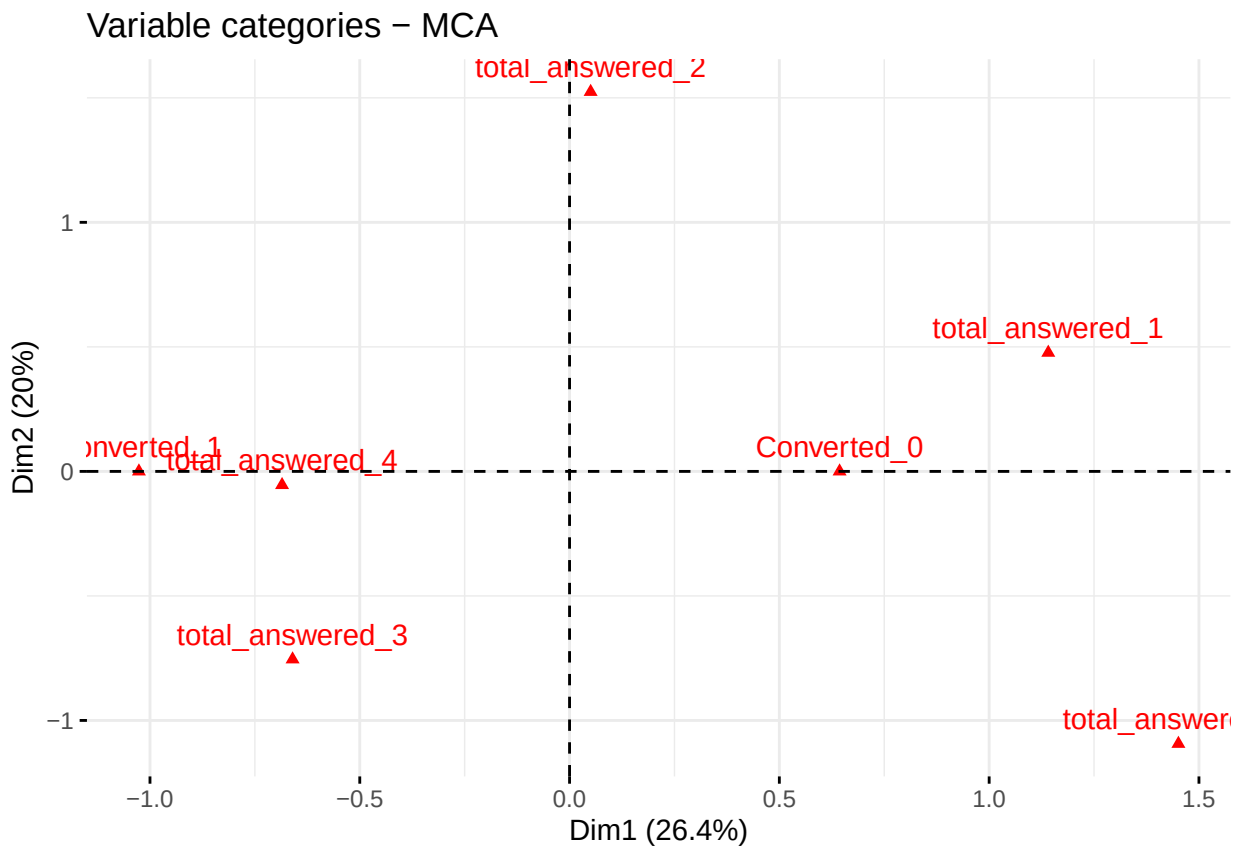
```
afc_vars <- leads %>%
  select(Converted, total_answered)

# AFC
afc <- MCA(afc_vars, graph = FALSE)
afc
```

```
## **Results of the Multiple Correspondence Analysis (MCA)**
## The analysis was performed on 9240 individuals, described by 2 variables
## *The results are available in the following objects:
```

```
##
##    name              description
## 1  "$eig"            "eigenvalues"
## 2  "$var"            "results for the variables"
## 3  "$var$coord"      "coord. of the categories"
## 4  "$var$cos2"       "cos2 for the categories"
## 5  "$var$contrib"    "contributions of the categories"
## 6  "$var$v.test"     "v-test for the categories"
## 7  "$ind"            "results for the individuals"
## 8  "$ind$coord"      "coord. for the individuals"
## 9  "$ind$cos2"       "cos2 for the individuals"
## 10 "$ind$contrib"    "contributions of the individuals"
## 11 "$call"           "intermediate results"
## 12 "$call$marge.col" "weights of columns"
## 13 "$call$marge.li"  "weights of rows"
```

Una vez llevado a cabo el análisis, procedemos a visualizar el plano factorial junto con las categorías de ambas variables. Esto lo deberemos de especificar con la función correspondiente puesto que también existe la posibilidad de visualizar las categorías de cada variable por separado. Utilizamos en este sentido la funcíon `fviz_mca_var()`.

```
fviz_mca_var(afc)
```



Observando el plano factorial resultante del AFC, podemos proceder a interpretarlo. Como se podía intuir, podemos concluir que mostrar un alto grado de interés (contestar 3 o 4 preguntas en el cuestionario) está muy asociado con que el *lead* se convierta finalmente en usuario de la empresa.

Por otra parte, las categorías asociadas a un bajo grado de interés (contestar 0 o 1 pregunta en el cuestionario)

está asociado con no convertirse *lead*.

El gráfico del plano factorial nos ha permitido examinar mas a fondo cómo era la asociación descubierta del test chi-cuadrado. De hecho, ambas técnicas se usan generalmente de manera complementaria.

Modelo de clasificación para predecir si un *lead* comprará o no alguno de los cursos

Como último ejercicio de análisis, buscamos ajustar un modelo de clasificación supervisado que sea capaz de predecir si un nuevo *lead* comprará finalmente alguno de los cursos ofrecidos por *X Education* en funcíon de las variables recogidas en el proceso de comunicación.

Para ello, en primer lugar será necesario identificar, de las 14 variables que forman nuestro conjunto de datos final, cuáles entrarán a formar parte del modelo. En el contexto de la construcción del modelo de clasificación, mantendremos 13 de las 14 variables, y únicamente despreciaremos la variable `Lead Number`, que recordemos servía exclusivamente como identificador de los registros del conjunto de datos.

```
leads <- leads %>%
  select(-Lead.Number)
```

Trabajaremos como habitualmente se trabaja bajo la metodología de aprendizaje supervisado. Para ello, utilizaremos uno de los paquetes más conocidos en `R` para dicho propósito como es `caret`.

En primer lugar, y dado el tamaño de nuestro conjunto de datos, lo dividiremos en dos subconjuntos: uno que llamaremos `train`, que servirá para el entrenamiento y desarrollo del algoritmo (o algoritmos), y otro que llamaremos `test`, que utilizaremos exclusivamente para evaluar la capacidad de generalización del modelo resultante.

Para ello, utilizamos la función `createDataPartition()`, que sirve justamente para dicho propósito. Emplearemos 2/3 del conjunto de datos para el subconjunto de `train` y el 1/3 restante para el subconjunto de `test`. Por defecto, ambos subconjuntos presentarán la misma proporción para la variable objetivo.

```
train.index <- createDataPartition(leads$Converted, p = 2/3, list = FALSE)

train <- leads[train.index,]
test <- leads[-train.index,]
```

A continuación, verificamos que el reparto de los registros de ambas categorías de la variable objetivo es proporcional es cada uno de los subconjuntos.

```
# Tabla de frecuencias relativas
prop.table(table(train$Converted))
```

```
##
##         0         1
## 0.6146104 0.3853896
```

```
prop.table(table(test$Converted))
```

```
##
##         0         1
## 0.6146104 0.3853896
```

Una vez establecidos los subconjuntos, será necesario definir una metodología específica para la construcción de los modelos de clasificación (por ejemplo, si vamos a realizar ajuste de hiperparámetros o si vamos probar más de un algoritmo distinto).

Dado el contexto de esta práctica, no prestaremos atención a ningún proceso de optimización de hiperparámetros, ya que queda fuera del alcance de la misma, pero nos pareció interesante comparar al menos dos algoritmos diferentes:

- Regresión logística.

- Random Forest [3].

Como hemos dicho, el subconjunto de `test` únicamente lo emplearemos para evaluar la capacidad de generalización del modelo final, por lo que no podrá ser utilizado para escoger uno de los dos algoritmos presentados. En consecuencia, será necesario definir una comparación justa de algoritmos utilizando únicamente el subconjunto `train`.

En este sentido, puede ser útil emplear procedimientos de validación cruzada que permitan evaluar los algoritmos utilizando las mismas particiones y de manera que cada instancia se utilice tanto para el entrenamiento como para la validación. Para ello, definiremos un único método de entrenamiento mediante la función `trainControl()`, de manera que será el que emplearemos para ambos algoritmos. Concretamente, dado el volumen de datos disponible, se estableció una validación cruzada con 3 *folds*.

Al definir un único método de validación cruzada, los *folds* creados serán los mismos para ambos algoritmos, lo que permitirá una comparación justa de su rendimiento a la hora de decidir entre uno u otro.

```
train_control <- trainControl(method = "cv", number = 3)
```

Una vez definido el método de entrenamiento, ajustamos los modelos y evaluamos los resultados obtenidos.

```
# glm -> implementación en caret de regresión logística
lr <- train(Converted ~ ., data = train, method = "glm", trControl = train_control)
lr
```

```
## Generalized Linear Model
##
## 6160 samples
##   12 predictor
##    2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 4107, 4106, 4107
## Resampling results:
##
##   Accuracy   Kappa
##   0.8196426  0.61317
```

```
# ranger -> implementación en caret de Random Forest
rf <- train(Converted ~ ., data = train, method = "ranger", trControl = train_control)
rf
```

```
## Random Forest
##
## 6160 samples
##   12 predictor
##    2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 4107, 4107, 4106
## Resampling results across tuning parameters:
##
##   mtry  splitrule   Accuracy   Kappa
##    2    gini        0.6975644  0.2580204
##    2    extratrees  0.6769476  0.1949916
##   38    gini        0.8426948  0.6651281
##   38    extratrees  0.8467522  0.6729526
```

```
##   74    gini        0.8357143  0.6503998
##   74    extratrees  0.8373364  0.6539566
##
## Tuning parameter 'min.node.size' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were mtry = 38, splitrule = extratrees
##  and min.node.size = 1.
```

Aunque el algoritmo Random Forest haya ajustado de manera automática algunos hiperparámetros, no prestaremos especial atención a este aspecto. Dicho lo cual, podemos comprobar que este algoritmo ha obtenido una exactitud de 0.847, un valor ligeramente superior al obtenido por la regresión logística (0.820). En consecuencia, elegimos Random Forest como modelo final para realizar las predicciones en el subconjunto `test` y evaluar su capacidad de generalización.

Para ello, será necesario atender a la matriz de confusión del modelo Para reportarla, junto con algunas métricas de interés, utilizamos la función `confusionMatrix()`.

```r
# Predicciones en test y matriz de confusión
pred <- predict(rf, newdata = test)
confusionMatrix(pred, test$Converted, positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1663  270
##          1  230  917
##
##                Accuracy : 0.8377
##                  95% CI : (0.8242, 0.8505)
##     No Information Rate : 0.6146
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.6552
##
##  Mcnemar's Test P-Value : 0.08114
##
##             Sensitivity : 0.7725
##             Specificity : 0.8785
##          Pos Pred Value : 0.7995
##          Neg Pred Value : 0.8603
##              Prevalence : 0.3854
##          Detection Rate : 0.2977
##    Detection Prevalence : 0.3724
##       Balanced Accuracy : 0.8255
##
##        'Positive' Class : 1
##
```

Como puede observarse, hemos obtenido una exactitud en el subconjunto de `test` prácticamente igual al subconjunto de `train`, alrededor de 0.84. Asumiendo por tanto que el subconjunto de `test` sea representativo de la población a estudiar, podríamos estar seguros de que nuestro modelo podría clasificar correctamente como futuro estudiante o no, al 84% de los nuevos *leads* que entren a formar parte del proceso de comunicación de *X Education*.

Adicionalmente, a raíz del reporte de clasificación, podemos evaluar otras métricas sobre el rendimiento de nuestro modelo. Por ejemplo, podemos destacar la sensibilidad y la especificidad, que presentan unos

valores de 0.77 y 0.88, respectivamente. Así, podemos concluir que nuestro modelo es capaz de identificar correctamente como usuario comprador al 77% de los *leads* que realmente comprarían alguno de los cursos. Por otra parte, de los *leads* que no comprarían ningun curso, nuestro modelo sería capaz de clasificar correctamente al 88%.

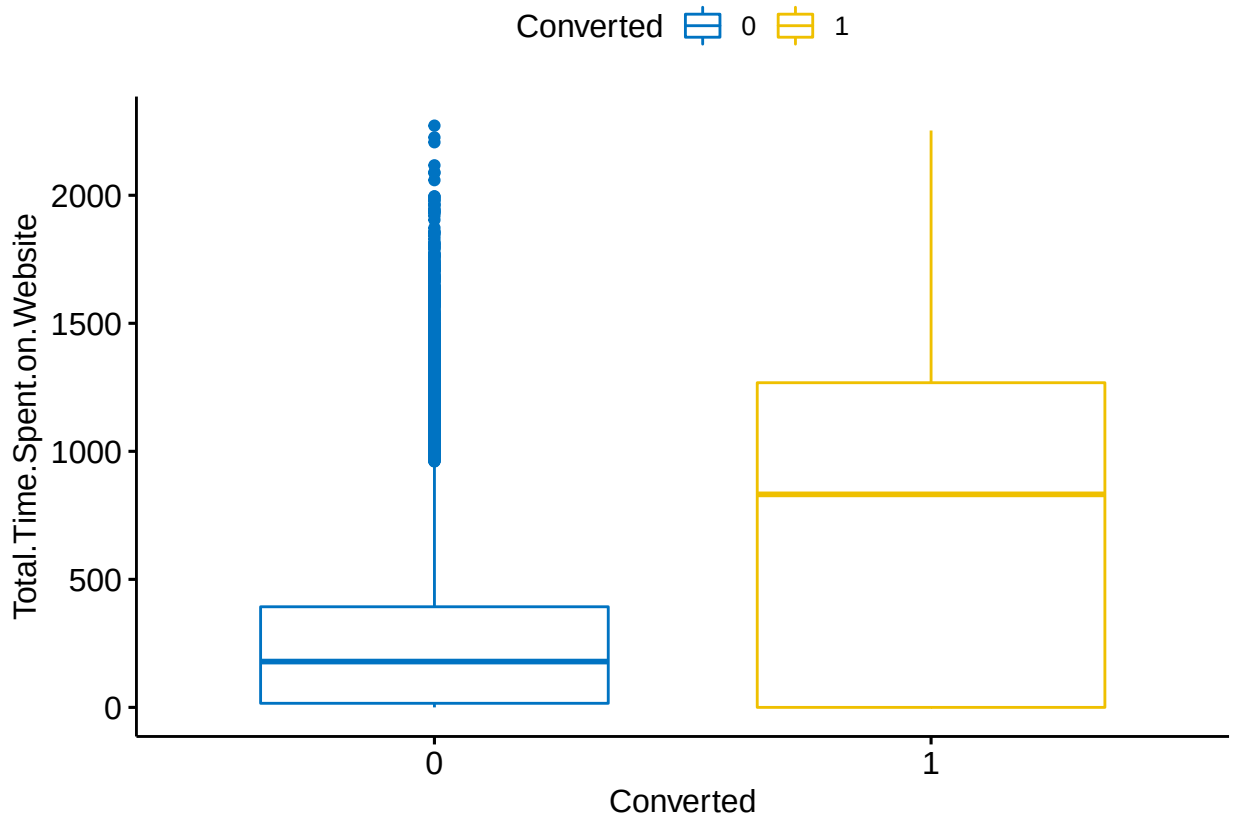
2.5. Representación de los resultados a partir de tablas y gráficas

Tras realizar todos los análisis planteados, será necesario reportar los resultados obtenidos mediante gráficas y tablas. Cabe mencionar que algunas de estas gráficas y tablas ya han sido mostradas a lo largo de la práctica por lo que simplemente serán referenciadas.

En primer lugar, se contrastó la normalidad de las variables numéricas involucradas en nuestro conujunto de datos. Una vez se comprobó la ausencia de normalidad, se representaron las distribuciones de las variables mediante histogramas (véase 2.4.2, Normalidad), donde se pudieron evidenciar grandes colas principalmente a la derecha de las distribuciones, identificando así la principal causa de la ausencia de normalidad.

Se contrastó la diferencia en el tiempo total empleado en la web en función del grupo considerado, tal y como puede apreciarse en el gráfico a continuación.

```
p4 <- ggboxplot(leads, x = "Converted",
                y = "Total.Time.Spent.on.Website",
                color = "Converted",
                palette = "jco")

p4
```

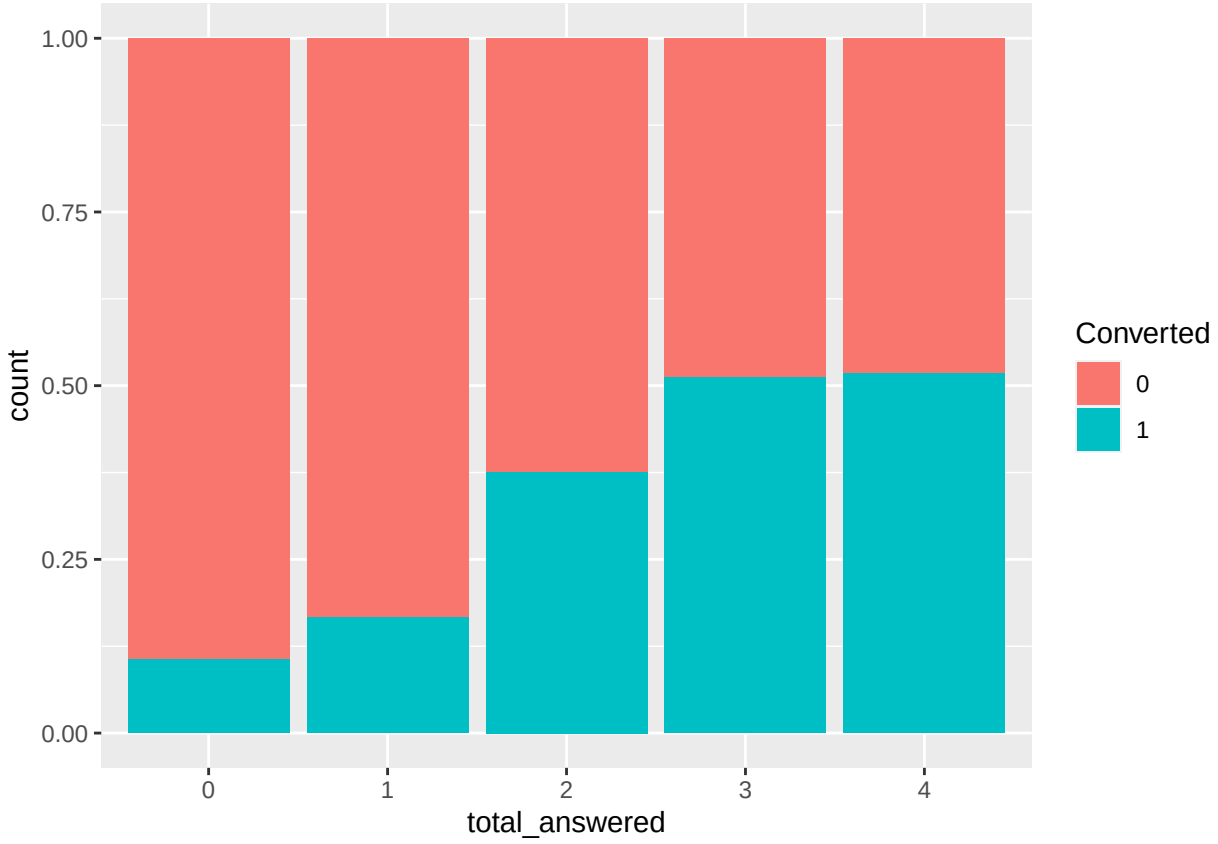


Como vemos, la media de tiempo empleado en la web era bastante superior en los individuos convertidos.

A la hora de comprobar el tipo de asocación entre el grado de interés y el hecho de convertirse o no en usuario,

el gráfico que mejor permite comprender dicha asociación se presentó en el apartado 2.4.3, a través del primer plano factorial del AFC. En este sentido, adicionalmente podemos visualizar cómo varía la frecuencia relativa de convertidos en función del número de respuestas a través de un gráfico de barras apiladas.

```
ggplot(data = leads) +
  aes(x = total_answered, fill = Converted) +
  geom_bar(position = "fill")
```



Podemos apreciar lo que ya exploramos con el AFC: a mayor número de respuestas contestadas se oberva un mayor número de *leads* que finalmente comparon alguno de los cursos.

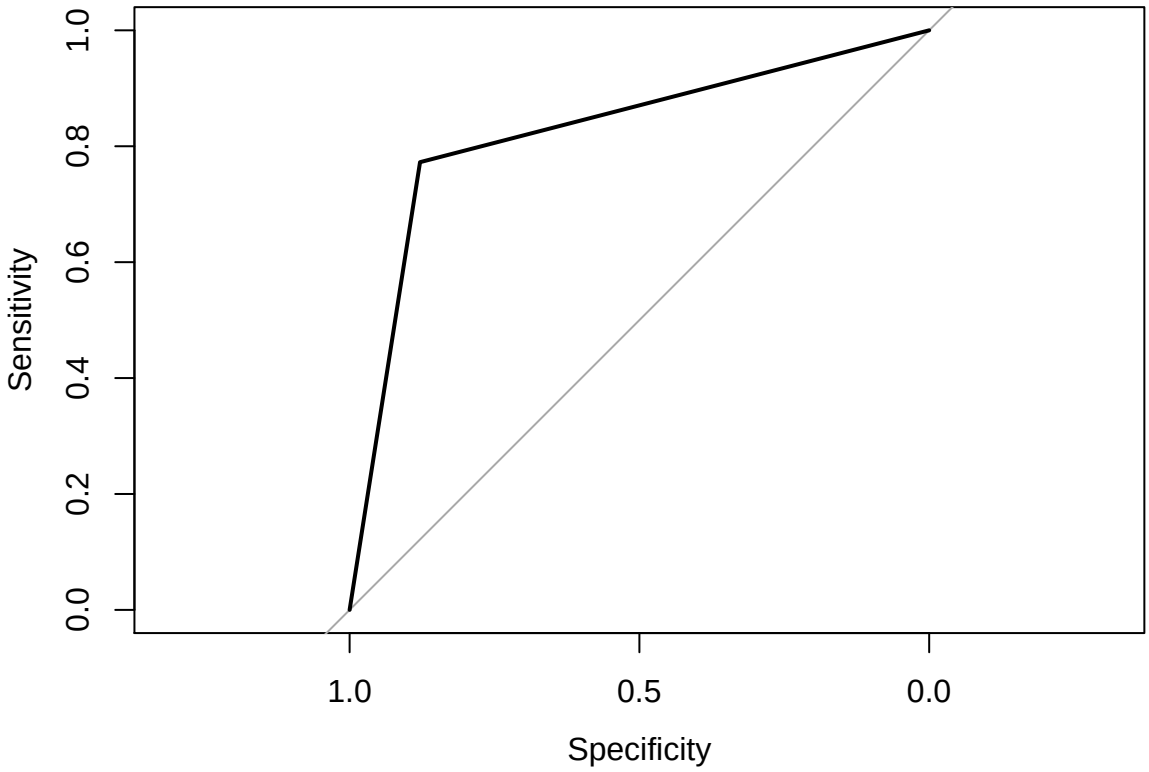
Finalmente, para resumir el rendimiento del modelo clasificación ajustado, se presento un reporte de clasificación para el subconjunto de `test`, donde se pudo observar la matriz de confusión y algunas métricas relevantes derivadas de la misma. Adicionalmente, este tipo de modelos suele evaluarse con la curva ROC, que representa todos los resultados posibles de sensibilidad y especificidad para diferentes umbrales de clasificación [4]. La elección del mejor umbral de clasificación, que depende del propósito específico del modelo y del contexto del caso de uso, queda fuera del alcance de esta práctica.

```
roc_obj <- roc(as.numeric(test$Converted), as.numeric(pred))
```

```
## Setting levels: control = 1, case = 2
```

```
## Setting direction: controls < cases
```

```
plot(roc_obj)
```



2.6. Resolución del problema

Recordemos del apartado 2.1 que la problemática global planteada a través de este conjunto de datos era la mejora en la identificación de los *leads* que tienen una mayor probabilidad de convertirse en futuros alumnos para que, a través de dicha información, ayudar al departamento de ventas a la hora de optimizar los recursos y aumentar con ello el % de *leads* que se terminan apuntando a alguno de los cursos ofertados.

Con esto presente, podemos considerar que los resultados obtenidos en la práctica tras la limpieza del conjutno de datos y los tres análisis posteriores nos han permitido responder al problema planteado. De manera resumida:

- Se determinó un primer indicador de conversión a estudiante relacionado con el tiempo total empleado en navegar en la página web por el *lead*. Como ha podido comprobarse, el tiempo total empleado en la web era mucho mayor en los casos en los que el *lead* finalemente se decidió a comprar alguno de los cursos ofertados.
- Se obtuvo un nuevo parámetro relacionado con el grado de interés mostrado por el *lead* identificado con el número de preguntas personales del cuestionario contestadas. Se comprobó que dicho parámtetro también presentaba una relación con el grado de conversión a estudiante. A mayor número de preguntas contestadas, mayor probabilidad de que el *lead* compre alguno de los cursos ofertados.
- Finalmente, se desarrolló un modelo de clasificación con un grado de exactitud del 84% a la hora de identificar qué *lead* se convertirá en estudiante y cual no. Este modelo permitirá ahorrar recursos a la empresa en cuestión y permitirá identificar fácilmente a usuarios potenciales. Por ejemplo, algunas de las variables recogidas incialmente y que luego no han formado parte del modelo que hacían referencia a puntuaciones y perfiles asignados a los *lead* en base al criterio del departamentento de ventas de la

empresa. Este paso podría ahorrarse en el proceso de identificación si se hiciera uso del modelo final que ha sido desarrollado.

Por tanto, concluímos que el conjunto de análisis realizados contribuyen significativamente a una mejor identificación de los futuros estudiantes a partir de las variables recogidas de los *leads*.

2.7. Código

Todo el código `R` utilizado en la práctica se encuentra contenido en el presente documento, que ha sido generado mediante `rmarkdown`.

3. Recursos y bibliografía

[1]. Kowarik, A., & Templ, M. (2016). Imputation with the R Package VIM. Journal of Statistical Software, 74(7), 1-16.

[2]. Kassambara, A. (2017). Kassambara, A. (2017). Practical guide to principal component methods in R: PCA, M (CA), FAMD, MFA, HCPC, factoextra (Vol. 2). STHDA.

[3]. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[4]. Kuhn, M., y Johnson, K. (2013). Applied predictive modeling. New York: Springer.

4. Tabla de contribuciones

- Investigación previa: Alberto García Galindo y Federico Alejandro Floriano Pardal.
- Redacción de las respuestas: Alberto García Galindo y Federico Alejandro Floriano Pardal.
- Desarrollo del código: Alberto García Galindo y Federico Alejandro Floriano Pardal.