ETAPA 1: DATA PREPARATION

Ana Luisa Masetto Herrera

1. Limpieza de Datos

En el siguiente Markdown de R se llevan acabo los procesos correspondientes a la primera etapa del proyecto: *DataPreparation*, dicha etapa incluye: Limpieza de Datos de ambas bases de datos proporcionadas por la compañia, transformación de variables e ingeniería de caracteristicas.

Es importante mencionar como paso antecesor a cargar los datos, primero se utiliza **Notepad++** para determinar la codificación en la que vienen ambos documentos proporcionados por la empresa, y se cambia a UTF-8 para un manejo más sencillo de estas.

1.1 Limpieza de Datos: Catálogo de la compañía

Paso 1: Cargar el archivo con la información del catálogo de tiendas de la empresa en cuestión.

```
#Lectura de datos
Catalogo <- read_csv("catalogo.csv")
```

```
dim(Catalogo)
```

```
## [1] 1911   79
```

```
head(Catalogo, 2)
```

```
## # A tibble: 2 x 79
##   `NOMBRE DEL PDV` PDV   `NUEVO NOMBRE D~ `NUEVA CLAVE PD~ `REGIONES HOMOL~
##   <chr>            <chr> <chr>            <chr>            <chr>
## 1 Apizaco          ACRT~ ACR-CENTROAPIZA~ ACRTDA006        Sur
## 2 Atlixco Centro   ACRT~ ACR-ATLIXCOCENT~ ACRTDA009        Sur
## # ... with 74 more variables: CANAL <chr>, DIVISIÓN <chr>,
## #   SUBDIVISIÓN <chr>, `FECHA APERTURA` <chr>, AÑO_APERTURA <dbl>,
## #   ESTADO <chr>, `PLAZA PDV` <chr>, CIUDAD <chr>, LATITUD <chr>,
## #   LONGITUD <chr>, UBICACIÓN <chr>, `LUNES - VIERNES` <chr>,
## #   SÁBADO <chr>, DOMINGO <chr>, CVE_AGEE <chr>, LAT <chr>, LONG <chr>,
## #   LATITUD_NUM <dbl>, LONGITUD_NUM <dbl>, POBTOT <dbl>, POBMAS <dbl>,
## #   POBFEM <dbl>, P_12YMAS <dbl>, P_18A24 <dbl>, POB15_64 <dbl>,
## #   PEA <dbl>, PEA_M <dbl>, PEA_F <dbl>, POCUPADA <dbl>, POCUPADA_M <dbl>,
## #   POCUPADA_F <dbl>, P12YM_SOLT <dbl>, P12YM_CASA <dbl>,
## #   P12YM_SEPA <dbl>, TOTHOG <dbl>, HOGJEF_M <dbl>, HOGJEF_F <dbl>,
## #   VPH_PC <dbl>, VPH_TELEF <dbl>, VPH_CEL <dbl>, VPH_INTER <dbl>,
## #   Pob_2011 <dbl>, Pob_2012 <dbl>, Pob_2013 <dbl>, Pob_2014 <dbl>,
## #   Pob_2015 <dbl>, Pob_2016 <dbl>, Pob_2017 <dbl>, Pob_2018 <dbl>,
## #   Pob_2019 <dbl>, Pob_2020 <dbl>, GRAPROES <dbl>, `VIV_A/B` <dbl>,
## #   `VIV_C+` <dbl>, VIV_NC <dbl>, VIV_C <dbl>, `VIV_C-` <dbl>,
## #   VIV_D <dbl>, `VIV_D+` <dbl>, VIV_E <dbl>, I <dbl>, PERC_AB <dbl>,
## #   PERC_CPLUS <dbl>, PERC_C <dbl>, PERC_CMIN <dbl>, PERC_DPLUS <dbl>,
## #   PERC_D <dbl>, PERC_E <dbl>, PERC_NC <dbl>, PERC_NULL <lgl>,
## #   VIV_NUL <lgl>, H_M <dbl>, PERC_OCUP <dbl>, P10_P18 <dbl>
```

Paso 2: Seleccionar variables importantes por conservar.

```
catalogo_peque <- Catalogo %>% select(`NOMBRE DEL PDV`,
                                      `NUEVO NOMBRE DEL PDV (Operacione`,
                                      ESTADO, CIUDAD,
                                      LATITUD_NUM,
                                      LONGITUD_NUM)
```

```
dim(catalogo_peque)
```

```
## [1] 1911    6
```

```
head(catalogo_peque, 2)
```

```
## # A tibble: 2 x 6
##   `NOMBRE DEL PDV` `NUEVO NOMBRE DE~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>             <chr>  <chr>        <dbl>        <dbl>
## 1 Apizaco          ACR-CENTROAPIZAC~ Tlaxc~ Apiza~        19.4        -98.1
## 2 Atlixco Centro   ACR-ATLIXCOCENTR~ Puebla Atlix~        18.9        -98.4
```

Paso 3: Pasar a minúsculas.

```
catalogo_peque$`NOMBRE DEL PDV`<-tolower(catalogo_peque$`NOMBRE DEL PDV`)
catalogo_peque$`NUEVO NOMBRE DEL PDV (Operacione`<-tolower(catalogo_peque$`NUEVO NOMBRE DEL PDV (Operac
catalogo_peque$ESTADO <- tolower(catalogo_peque$ESTADO)
catalogo_peque$CIUDAD <- tolower(catalogo_peque$CIUDAD)
```

Paso 4: Remover caracteres especiales en cada columna.

```
catalogo_peque$`NOMBRE DEL PDV` <- str_replace(catalogo_peque$`NOMBRE DEL PDV`, "á", "a") %>%
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")%>%
  str_replace(" - ", " ")%>%
  str_replace("-"," ")%>%
  str_replace("  "," ")

catalogo_peque$`NUEVO NOMBRE DEL PDV (Operacione` <- str_replace(catalogo_peque$`NUEVO NOMBRE DEL PDV (
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")%>%
  str_replace(" - ", " ")%>%
  str_replace("-"," ")%>%
  str_replace("  "," ")

catalogo_peque$ESTADO <- str_replace(catalogo_peque$ESTADO, "á", "a") %>%
  str_replace("é","e")%>%
```

```
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")%>%
  str_replace(" - ", " ")%>%
  str_replace("-"," ")%>%
  str_replace("  "," ")

catalogo_peque$CIUDAD <- str_replace(catalogo_peque$CIUDAD , "á", "a") %>%
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")%>%
  str_replace(" - ", " ")%>%
  str_replace("-"," ")%>%
  str_replace("  "," ")
```

Paso 5: Homogeneizar la forma en la que están escritos los estados.

```
catalogo_peque$ESTADO <- str_replace(catalogo_peque$ESTADO, "ciudad de mexico", "cdmx")%>%
  str_replace("edo. mex.", "estado de mexico")%>%
  str_replace("matamoros","tamaulipas")
```

Paso 6: Homogeneizar las zonas en el documento.

Se observa que las zonas no estan bien delimitadas, por ende, se adquiere información de la página de CONABIO y se determinan las zonas de la siguiente manera. Al final se tiene una tabla que se va a pegar con el catálogo.

```
#knitr::include_graphics("project-objectstorage/mapa_nuevo.PNG")
```

```
#Lectura del archivo con las nuevas zonas delimitadas
regiones <- read_csv("regiones_economicas.csv")
names(regiones)[1]<-"ESTADO"

#Pasar a minúsculas
regiones$ESTADO<-tolower(regiones$ESTADO)
regiones$Zona<-tolower(regiones$Zona)

#Quitar caracteres especiales de este nuevo documento
regiones$ESTADO <- str_replace(regiones$ESTADO, "á", "a") %>%
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")

regiones$Zona <- str_replace(regiones$Zona, "á", "a") %>%
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")
```

Paso 7: Completar Puntos de Venta.

En el documento con los registros (SD.csv) existen puntos de venta que no concuerdan con ninguno de los registros en este catalogo, por ende, se hace un proceso sumamente artesal que involucra detectar las tiendas que no se encuentran en este archivo y agregarlas.

```
agregar_a_catalogo_1 <- read_csv("agregar_a_catalogo_1.csv")
agregar_a_catalogo_1 <- agregar_a_catalogo_1 %>% mutate("columna"== NA)
names(agregar_a_catalogo_1)[6]<-"NUEVO NOMBRE DEL PDV (Operacione"
agregar_a_catalogo_1 <- agregar_a_catalogo_1 %>% select(`NOMBRE DEL PDV`, `NUEVO NOMBRE DEL PDV (Operac
```

```
catalogo_peque <- rbind(catalogo_peque, agregar_a_catalogo_1)
```

```
agregar_a_catalogo_2 <- read_csv("agregar_a_catalogo_2.csv") #716
agregar_a_catalogo_2 <- agregar_a_catalogo_2 %>% mutate("columna"== NA)
names(agregar_a_catalogo_2)[6]<-"NUEVO NOMBRE DEL PDV (Operacione"
agregar_a_catalogo_2 <- agregar_a_catalogo_2 %>% select(`NOMBRE DEL PDV`, `NUEVO NOMBRE DEL PDV (Operac
```

```
catalogo_peque <- rbind(catalogo_peque, agregar_a_catalogo_2)
```

```
dim(catalogo_peque)
```

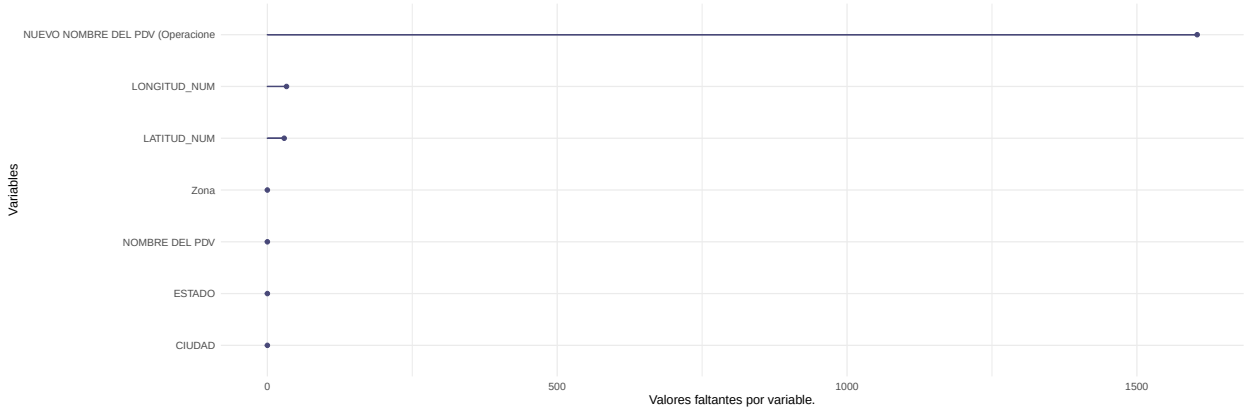
```
## [1] 2801    6
```

Ahora se tienen 2801 puntos de venta.

Paso 8: Imputar valores con nuevas zona.

```
nuevo_catalogo <- left_join(catalogo_peque, regiones, by="ESTADO")
```

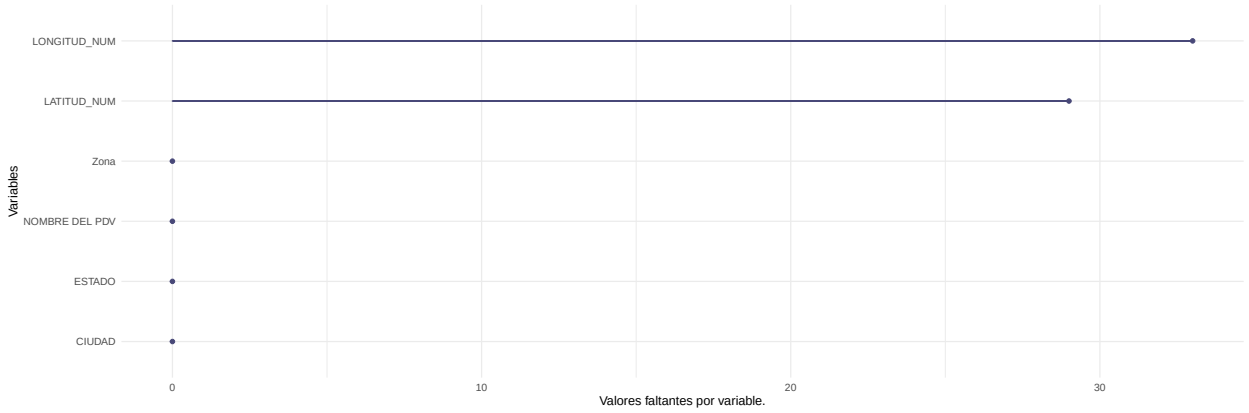
```
gg_miss_var(nuevo_catalogo) + labs(y = "Valores faltantes por variable.")
```



La variable con el mayor número de campos vacíos es la de "NUEVO NOMBRE DEL PUNTO DE VENTA", esto se debe a que no todos los puntos de venta cambiaron de nombre, por lo tanto, los valores faltantes corresponden al valor en la columna previa, situación que más adelante se va a abarcar.

Paso 9: Imputar valores faltantes de las variables: Longitud y Latitud

```
sin_longitud_y_latitud<-nuevo_catalogo%>%select(-`NUEVO NOMBRE DEL PDV (Operacione`)
gg_miss_var(sin_longitud_y_latitud) + labs(y = "Valores faltantes por variable.")
```




```
#valores faltantes longitud 33 faltantes
which(is.na(nuevo_catalogo$LONGITUD_NUM))
```

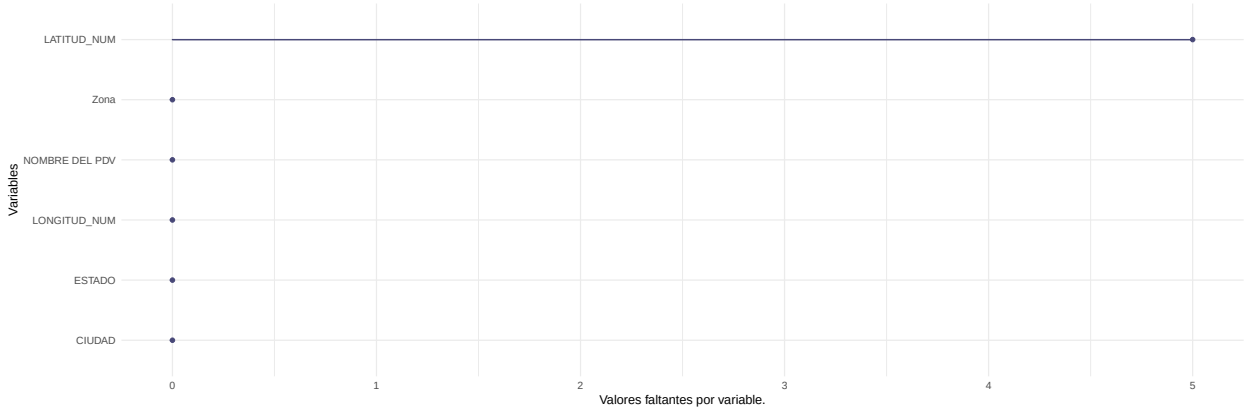
```
##  [1]  316  345  346  347  348  349  350  351  378  385  413  414  415  417
## [15]  420  423  480  483  485  487  597  637  812  849  865 1272 1297 1573
## [29] 1608 1664 1791 1801 1879
```

```
#Guardar dichas observaciones para comprobar que efectivamente no tienen valor en ese campo y algunas d
faltantes_longitud<-nuevo_catalogo[c(316, 345 , 346, 347, 348 , 349, 350, 351, 378, 385, 413, 414, 415,
```

```
#Imputar longitud y latitud en los 33 registros detectados con anterioridad (proceso sumamente artesana
nuevo_catalogo[316, c(5,6)] <- c("19.322907", "-99.106639")
nuevo_catalogo[345, c(5,6)] <- c("25.616183", "-100.273719")
nuevo_catalogo[346, c(5,6)] <- c("20.656168", "-103.317118")
nuevo_catalogo[347, c(5,6)] <- c("25.750311", "-100.256962")
nuevo_catalogo[348, c(5,6)] <- c("19.374657", "-99.123360")
nuevo_catalogo[349, c(5,6)] <- c("25.668530", "-100.312852")
nuevo_catalogo[350, c(5,6)] <- c("25.768912", "-100.299058")
nuevo_catalogo[351, c(5,6)] <- c("25.764158", "-100.191360")
nuevo_catalogo[378, c(5,6)] <- c("32.657752", "-115.413286")
nuevo_catalogo[385, c(5,6)] <- c("19.271492", "-99.601485")
nuevo_catalogo[413, c(5,6)] <- c("19.485674", "-99.092861")
nuevo_catalogo[414, c(5,6)] <- c("19.479683", "-99.095414")
nuevo_catalogo[415, c(5,6)] <- c("19.472291", "-99.120370")
nuevo_catalogo[417, c(5,6)] <- c("19.630151", "-99.123269")
nuevo_catalogo[420, c(5,6)] <- c("19.006252", "-98.239517")
nuevo_catalogo[423, c(5,6)] <- c("19.264139", "-98.896666")
nuevo_catalogo[480, c(5,6)] <- c("19.649018", "-99.206474")
nuevo_catalogo[483, c(5,6)] <- c("19.520872", "-99.251211")
nuevo_catalogo[485, c(5,6)] <- c("19.513555", "-96.859482")
nuevo_catalogo[487, c(5,6)] <- c("19.139566", "-96.105681")
nuevo_catalogo[597, c(5,6)] <- c("19.531056", "-96.892723")
nuevo_catalogo[637, c(5,6)] <- c("19.406433", "-99.168831")
nuevo_catalogo[812, c(5,6)] <- c("18.891172", "-96.937202")
nuevo_catalogo[849, c(5,6)] <- c("20.504322", "-86.956190")
```

```
nuevo_catalogo[865, c(5,6)] <- c("20.632762", "-103.244771")
nuevo_catalogo[1272, c(5,6)] <- c("17.062675", "-96.718006")
nuevo_catalogo[1297, c(5,6)] <- c("20.095259", "-98.770459")
nuevo_catalogo[1573, c(5,6)] <- c("17.989232", "-92.929308")
nuevo_catalogo[1608, c(5,6)] <- c("24.814153", "-107.399681")
nuevo_catalogo[1664, c(5,6)] <- c("20.290522", "-102.710445")
nuevo_catalogo[1791, c(5,6)] <- c("19.077216", "-98.295908")
nuevo_catalogo[1801, c(5,6)] <- c("18.933995", "-99.221997")
nuevo_catalogo[1879, c(5,6)] <- c("19.393622", "-99.166727")
```

```
sin_latitud<-nuevo_catalogo%>%select(-`NUEVO NOMBRE DEL PDV (Operacione`)
gg_miss_var(sin_latitud) + labs(y = "Valores faltantes por variable.")
```



Aún hay 5 valores faltantes ene la variable de latitud que se pueden imputar.

```
#valores faltantes latitud: 5faltantes
which(is.na(nuevo_catalogo$LATITUD_NUM))
```

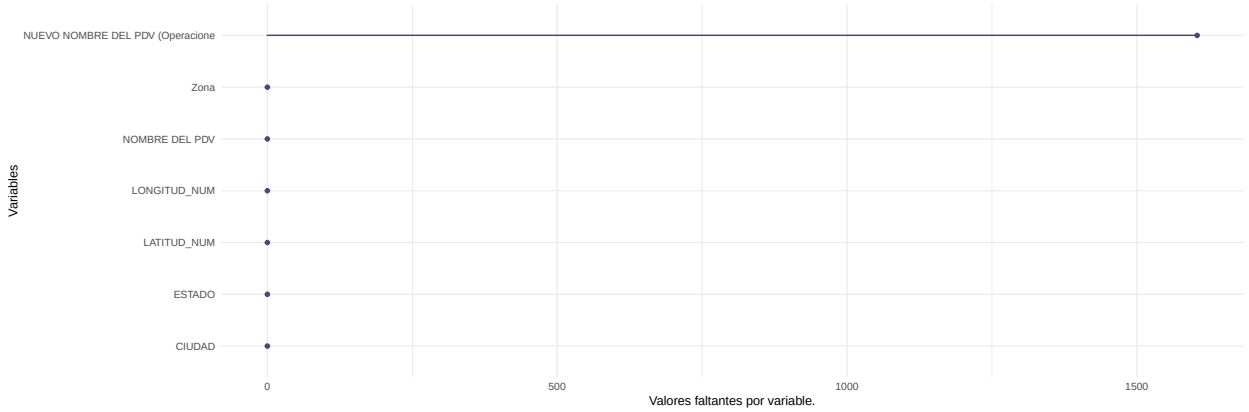
```
## [1]  521  523  527 1607 1900
```

```
#Guardar dichas observaciones para comprobar que efectivamente no tienen valor en ese campo.
faltantes_latitud<-nuevo_catalogo[c(521, 523, 527, 1607, 1900),]
```

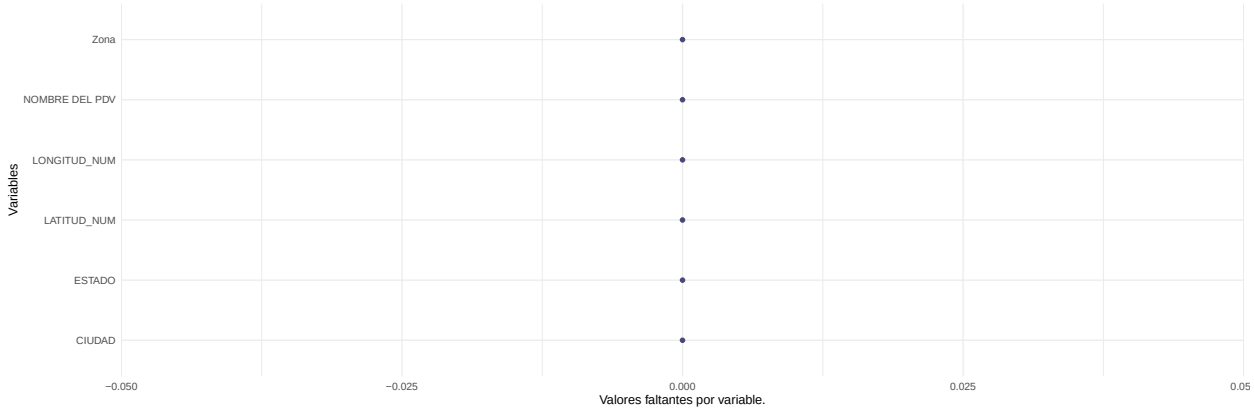
```
#Imputar latitud en los 5 registros detectados con anterioridad (proceso artesanal que consistio en bus
nuevo_catalogo[521, c(5,6)] <- c("25.766070", "-100.402907")
nuevo_catalogo[523, c(5,6)] <- c("25.694716", "-100.240128")
nuevo_catalogo[527, c(5,6)] <- c("25.618766", "-100.282577")
nuevo_catalogo[1607, c(5,6)] <- c("24.815505", "-107.387520")
nuevo_catalogo[1900, c(5,6)] <- c("20.674653", "-103.404025")
```

Ahora se puede observar que los únicos valores faltantes son los de la columna relacionada con el *nuevonombredelpuntodeventa*.

```
gg_miss_var(nuevo_catalogo) + labs(y = "Valores faltantes por variable.")
```




```
sin_faltantes<-nuevo_catalogo%>%select(-`NUEVO NOMBRE DEL PDV (Operacione`)
gg_miss_var(sin_faltantes) + labs(y = "Valores faltantes por variable.")
```



Paso 10: Asignar tipos de variables correctos.

Haciendo un resumen general de las variables se puede observar que todas las variables tienen formato *character*, lo cual debe cambiarse en el caso de las variables cuyos valores correspondan a números.

```
summary(nuevo_catalogo)
```

```
##  NOMBRE DEL PDV     NUEVO NOMBRE DEL PDV (Operacione    ESTADO
##  Length:2801        Length:2801                       Length:2801
##  Class :character   Class :character                  Class :character
##  Mode  :character   Mode  :character                  Mode  :character
##     CIUDAD          LATITUD_NUM        LONGITUD_NUM
##  Length:2801        Length:2801        Length:2801
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##      Zona
##  Length:2801
##  Class :character
##  Mode  :character
```

```
nuevo_catalogo$LATITUD_NUM <- as.numeric(nuevo_catalogo$LATITUD_NUM)
nuevo_catalogo$LONGITUD_NUM <- as.numeric(nuevo_catalogo$LONGITUD_NUM)
```

```
summary(nuevo_catalogo)
```

```
##  NOMBRE DEL PDV     NUEVO NOMBRE DEL PDV (Operacione    ESTADO
##  Length:2801        Length:2801                       Length:2801
##  Class :character   Class :character                  Class :character
##  Mode  :character   Mode  :character                  Mode  :character
##
##
##
##     CIUDAD            LATITUD_NUM        LONGITUD_NUM
##  Length:2801        Min.   :      15   Min.   :-117.12
##  Class :character   1st Qu.:      19   1st Qu.:-102.06
##  Mode  :character   Median :      20   Median : -99.24
##                     Mean   :   10404   Mean   : -99.44
##                     3rd Qu.:      22   3rd Qu.: -98.75
##                     Max.   :29081531   Max.   : 115.36
##      Zona
##  Length:2801
##  Class :character
##  Mode  :character
##
##
##
```

Paso 11: Corrección de registros erroneos en las columnas de longitud y latitud.

Una vez que se cambian los tipos de valores de las columnas numéricas se puede observar que hay valores mal registrados, en primer lugar, la columna relacionada con la variable de *longitud* tiene valores positivos lo cuál no se puede, dado que el territorio mexicano no abarca esas zonas, y en segundo lugar, valores en la columna de *latitud* están mal registrados ya que el valor máximo es de más de 20 millones. Es por eso que a continuación se corigen esos registros.

```
summary(nuevo_catalogo$LONGITUD_NUM)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -117.12 -102.06  -99.24  -99.44  -98.75  115.36
```

```
summary(nuevo_catalogo$LATITUD_NUM)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##       15       19       20    10404       22 29081531
```

```
# detectar Longitud positiva
which(nuevo_catalogo$LONGITUD_NUM>0)
```

```
##  [1]  164  524  525  526  528  590  862 1574 1604 1605 1901
```

```
longitud_positiva<-nuevo_catalogo[c(164, 524, 525, 526, 528, 590, 862, 1574, 1604, 1605, 1901),]
```

```r
#cambiar a negativo
nuevo_catalogo[164,6] <- "-103.38702"
nuevo_catalogo[524,6] <- "-103.41777"
nuevo_catalogo[525, 6] <- "-100.95139"
nuevo_catalogo[526, 6] <- "-115.35694"
nuevo_catalogo[528, 6] <- "-103.41558"
nuevo_catalogo[590, 6] <- "-99.23513"
nuevo_catalogo[862, 6] <- "-102.28584"
nuevo_catalogo[1574, 6] <- "-93.21964"
nuevo_catalogo[1604, 6] <- "-110.94573"
nuevo_catalogo[1605, 6] <-  "-109.89929"
nuevo_catalogo[1901, 6] <- "-103.28292"
```

```r
#latitud esta mal
which(nuevo_catalogo$LATITUD_NUM>33)
```

```
## [1]  643 1316
```

```r
latitud_exagerada <-nuevo_catalogo[c(643,1316),]
```

```r
nuevo_catalogo[643, c(5,6)] <- c("29.085553", "-110.994395")
nuevo_catalogo[1316, c(5,6)] <- c("19.287195", "-99.653878")
```

```r
nuevo_catalogo$LATITUD_NUM <- as.numeric(nuevo_catalogo$LATITUD_NUM)
nuevo_catalogo$LONGITUD_NUM <- as.numeric(nuevo_catalogo$LONGITUD_NUM)
summary(nuevo_catalogo)
```

```
##  NOMBRE DEL PDV     NUEVO NOMBRE DEL PDV (Operacione    ESTADO
##  Length:2801        Length:2801                      Length:2801
##  Class :character   Class :character                 Class :character
##  Mode  :character   Mode  :character                 Mode  :character
##
##
##
##     CIUDAD            LATITUD_NUM     LONGITUD_NUM         Zona
##  Length:2801        Min.   :14.87   Min.   :-117.12   Length:2801
##  Class :character   1st Qu.:19.35   1st Qu.:-102.26   Class :character
##  Mode  :character   Median :20.06   Median : -99.24   Mode  :character
##                     Mean   :21.58   Mean   :-100.29
##                     3rd Qu.:22.16   3rd Qu.: -98.76
##                     Max.   :32.67   Max.   : -86.81
```

Con esto las variables de latitud y longitud ya están listas para ser empleadas.

Paso 12: Quitar caracteres especiales una vez más.

Se pensaba que los caracteres especiales ya se habían quitado, sin embargo, faltó remover los signos de interrogación.

```r
nuevo_catalogo$`NOMBRE DEL PDV`<-gsub("[[:punct:]]", "", nuevo_catalogo$`NOMBRE DEL PDV`)
```

Paso 13: Detectar tiendas repetidas.

Hay 2770 Puntos de Venta únicos Y 2801 líneas en el catálogo, es por eso que se sabe que existen tiendas que se repiten, lo cual se debe de revisar. Si una tienda se repite más de una vez pero sus columnas adyacentes no son las mismas es porque corresponden a puntos de venta distintos que deben de poder distinguirse entre si, por lo tanto, si los registros son iguales, se eliminan los extras y se queda con un solo registro, pero si los registros difieren, se debe de asignar un nombre que distinga cada punto de venta.

```
pdv <- nuevo_catalogo%>%select(`NOMBRE DEL PDV`)%>%arrange(`NOMBRE DEL PDV`)%>%unique()
dim(pdv)
```

```
## [1] 2770    1
```

```
#agregar una columna con 1's para hacer conteo de repeticiones y enseguida revisar que tienen de difere
nuevo_catalogo <- nuevo_catalogo%>%mutate(repetidas = 1)
repetidas <- nuevo_catalogo%>%group_by(`NOMBRE DEL PDV`)%>%summarise(repeticiones = sum(repetidas))%>%a
repetidas
```

```
## # A tibble: 2,770 x 2
##    `NOMBRE DEL PDV`     repeticiones
##    <chr>                       <dbl>
##  1 digital 2000 dolores            4
##  2 chapultepec                     3
##  3 codite                          3
##  4 paseo interlomas                3
##  5 canek                           2
##  6 chapultepec slp                 2
##  7 chetumal                        2
##  8 city center                     2
##  9 la cima                         2
## 10 la cuspide                      2
## # ... with 2,760 more rows
```

1. Digital 2000 dolores

```
#no requiere modificaciones porque "nuevo nombre del pdv" esta completo y las 4 repeticiones se deben a
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="digital 2000 dolores")
```

```
## # A tibble: 4 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 digital 2000 do~ digital 2000 do~ guana~ dolor~        21.2        -101.
## 2 digital 2000 do~ corporativo2000~ jalis~ lagos~        21.4        -102.
## 3 digital 2000 do~ corporativo2000~ guana~ san f~        21.0        -102.
## 4 digital 2000 do~ corporativo2000~ guana~ dolor~        21.2        -101.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

2. Chapultepec

```
#cambiar "nuevo nombre" porque son diferentes establecimientos
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="chapultepec")
```

```
## # A tibble: 3 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 chapultepec      <NA>             nuevo~ monte~        25.7       -100.
## 2 chapultepec      <NA>             jalis~ guada~        20.7       -103.
## 3 chapultepec      <NA>             morel~ cuern~        18.9        -99.2
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="chapultepec")
```

```
## [1]  577  836 1278
```

```
#577 836 1278
```

```
#Renombrar puntos de venta para distinguirlos dado que corresponden a diferentes estados
nuevo_catalogo[577,2]<-"chapultepec mty"
nuevo_catalogo[836,2]<-"chapultepec gdl"
nuevo_catalogo[1278,2]<-"chapultepec mrls"
```

3. Codite

```
#no requiere modificaciones porque "nuevo nombre del pdv" esta completo y las 3 repeticiones se deben a
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="codite")
```

```
## # A tibble: 3 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 codite           codite jalisco ~ jalis~ arand~        20.7        -102.
## 2 codite           digitales tepic~ jalis~ arand~        20.7        -102.
## 3 codite           codite jalisco ~ jalis~ guada~        20.7        -103.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

4. Paseo interlomas

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="paseo interlomas")
```

```
## # A tibble: 3 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 paseo interlomas <NA>             estad~ huixq~        19.4        -99.3
## 2 paseo interlomas <NA>             estad~ huixq~        19.4        -99.3
## 3 paseo interlomas <NA>             estad~ huixq~        19.4        -99.3
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="paseo interlomas")
```

```
## [1] 1293 1786 1792
```

```
#1293 1786 1792 hay que remover los últimos 2 porque son iguales
```

5. Canek

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="canek")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 canek            <NA>             yucat~ merida        21.0        -89.7
## 2 canek            <NA>             yucat~ merida        21.0        -89.7
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="canek")
```

```
## [1]  905 1576
```

```
#905 1576 - hay que remover el último porque son lo mismo
```

6. Chapultepec SLP

```
#no requiere modificaciones porque "nuevo nombre del pdv" esta completo y las 2 repeticiones se deben a
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="chapultepec slp")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 chapultepec slp  bca parque espa~ san l~ san l~        22.2        -101.
## 2 chapultepec slp  bca chapultepec~ san l~ san l~        22.1        -101.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

7. Chetumal

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="chetumal")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 chetumal         <NA>             quint~ chetu~        18.5        -88.3
## 2 chetumal         <NA>             quint~ chetu~        18.5        -88.3
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="chetumal")
```

```
## [1] 1560 1666
```

```
#1560 1666 - hay que remover 1 porque son iguales
```

8. City center

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="city center") #quedarse con estado de mexico por pagina de i
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 city center      <NA>             yucat~ merida        21.0        -89.6
## 2 city center      <NA>             estad~ adolf~        19.5        -99.3
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="city center")
```

```
## [1] 1561 1763
```

```
#1561 1763 el segundo es el que corresponde al edo de mexico
```

9. La cima

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="la cima") #quedarse con las dos La cima GLD Y LA OTRA LA CIM
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 la cima          <NA>             jalis~ guada~        20.7        -103.
## 2 la cima          <NA>             jalis~ zapop~        20.7        -103.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="la cima")
```

```
## [1]  857 1894
```

```
#857 1894
```

```
nuevo_catalogo[857,2]<-"la cima gdl"
nuevo_catalogo[1894,2]<-"la cima"
```

10. La cuspide

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="la cuspide")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 la cuspide       business ca pla~ estad~ nauca~        19.5        -99.3
## 2 la cuspide       <NA>             estad~ nauca~        19.5        -99.3
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="la cuspide")
```

```
## [1]  103 1767
```

```r
#103 1767 - hay que remover 1 porque son iguales
```

11. Martinez de la torre

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="martinez de la torre")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 martinez de la ~ rax martinez de~ verac~ marti~        20.1        -97.1
## 2 martinez de la ~ <NA>             verac~ marti~        20.1        -97.1
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="martinez de la torre")
```

```
## [1] 1271 1857
```

```r
#1271 1857 remover 1 porque son iguales
```

12. Multiplaza lindavista

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="multiplaza lindavista")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 multiplaza lind~ <NA>             nuevo~ guada~        25.7        -100.
## 2 multiplaza lind~ <NA>             nuevo~ guada~        25.7        -100.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="multiplaza lindavista")
```

```
## [1] 632 655
```

```r
#632 655 remover 1 porque son iguales
```

13. Arboledas

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza arboledas") #quedarse con las dos
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza arboledas  <NA>             chiap~ tuxtl~        16.8        -93.1
## 2 plaza arboledas  <NA>             jalis~ zapop~        20.6       -103.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza arboledas")
```

```
## [1] 486 824
```

```r
nuevo_catalogo[486,2]<-"plaza arboledas chiapas"
nuevo_catalogo[824,2]<-"plaza arboledas jalisco"
```

14. Plaza comercial cosmopol

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza comercial cosmopol")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza comercial~ bca cosmopolimex estad~ coaca~        19.6        -99.1
## 2 plaza comercial~ bca cosmopolimex estad~ coaca~        19.6        -99.1
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza comercial cosmopol")
```

```
## [1] 122 123
```

```r
#122 123 - remover 1 porque son iguales
```

15. Plaza crystal

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza crystal")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza crystal    <NA>             verac~ verac~        19.2        -96.1
## 2 plaza crystal    <NA>             puebla puebla        19.0        -98.2
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza crystal")
```

```
## [1] 407 431
```

```r
nuevo_catalogo[407,2]<-"plaza crystal veracruz"
nuevo_catalogo[431,2]<-"ksk puebla plaza puebla"
```

16. Plaza esmeralda

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza esmeralda") #dejar las dos y especificar
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza esmeralda  <NA>             estad~ atiza~        19.6        -99.3
## 2 plaza esmeralda  <NA>             morel~ cuern~        18.9        -99.2
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza esmeralda")
```

```
## [1]  529 1304
```

```
#529 1304
```

```
nuevo_catalogo[529,2]<-"fgt plaza esmeralda"
```

17. Plaza palmas

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza palmas") # se quedan los dos porque son
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza palmas     <NA>             baja ~ tijua~        32.5       -117.
## 2 plaza palmas     <NA>             morel~ cuaut~        18.8        -98.9
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza palmas")
```

```
## [1] 578 843
```

```
nuevo_catalogo[843,2]<-"mgn plaza palmas"
```

18. Plaza real

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza real") # se quedan las dos
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza real       <NA>             campe~ cd. d~        18.7        -91.8
## 2 plaza real       <NA>             puebla san a~        19.0        -98.3
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza real") #renombrar porque no sale en documento de ventas
```

```
## [1]  883 1777
```

```
nuevo_catalogo[883,2]<-"plaza real campeche"
nuevo_catalogo[1777,2]<-"plaza real puebla"
```

19. Plaza sendero

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="plaza sendero") #quedarse con las 2 y especificar
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 plaza sendero    <NA>             yucat~ merida        21.0        -89.6
## 2 plaza sendero    <NA>             estad~ lerma         19.3        -99.6
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="plaza sendero")
```

```
## [1]  903 1294
```

```
#903 1294
```

```
nuevo_catalogo[903,2]<-"plaza sendero merida"
nuevo_catalogo[1294,2]<-"plaza sendero lerma"
```

20. Power center coacalco

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="power center coacalco") #se queda 1 son iguales
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 power center co~ bca coacalco po~ estad~ coaca~        19.6        -99.1
## 2 power center co~ <NA>             estad~ nezah~        19.6        -99.1
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="power center coacalco")
```

```
## [1] 126 442
```

```
#126 442 se debe de remover una
```

21. Power center tecamac

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="power center tecamac")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 power center te~ business corp a~ estad~ tecam~        19.7        -99.0
## 2 power center te~ <NA>             estad~ tecam~        19.7        -99.0
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="power center tecamac")
```

```
## [1] 128 337
```

```r
#128 337 se debe de remover una
```

22. Santa Ana

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="santa ana") #se quedan los dos pero no hay match en el otro
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 santa ana        <NA>             campe~ campe~        19.8        -90.5
## 2 santa ana        <NA>             estad~ toluca        19.3        -99.6
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="santa ana")
```

```
## [1] 1566 1779
```

```r
#1566 1779
```

```r
nuevo_catalogo[1566,2]<-"santa ana campeche"
nuevo_catalogo[1779,2]<-"santa ana toluca"
```

23. tepatitlan

```r
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="tepatitlan")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 tepatitlan       cdt tepatitlan   jalis~ tepat~        20.8        -103.
## 2 tepatitlan       <NA>             jalis~ tepat~        20.8        -103.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```r
which(nuevo_catalogo$`NOMBRE DEL PDV`=="tepatitlan")
```

```
## [1]  302 1625
```

```r
#302 1625 se debe de remover una
```

24. Texcoco

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="texcoco")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 texcoco          arsa texcoco     estad~ texco~        19.4        -98.9
## 2 texcoco          <NA>             estad~ texco~        19.5        -98.9
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="texcoco")
```

```
## [1]  63 435
```

```
#63 435 se debe de remover una
```

25. Tlajomulco

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="tlajomulco")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 tlajomulco       <NA>             jalis~ tlajo~        20.6        -103.
## 2 tlajomulco       <NA>             jalis~ tlajo~        20.5        -103.
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="tlajomulco")
```

```
## [1]  831 1618
```

```
#831 1618 se debe de remover una
```

26. Town center nicolas romero

```
nuevo_catalogo%>%filter(`NOMBRE DEL PDV`=="town center nicolas romero")
```

```
## # A tibble: 2 x 8
##   `NOMBRE DEL PDV` `NUEVO NOMBRE D~ ESTADO CIUDAD LATITUD_NUM LONGITUD_NUM
##   <chr>            <chr>            <chr>  <chr>        <dbl>        <dbl>
## 1 town center nic~ bca nicolas rom~ estad~ villa~        19.6        -99.3
## 2 town center nic~ <NA>             estad~ nicol~        19.6        -99.3
## # ... with 2 more variables: Zona <chr>, repetidas <dbl>
```

```
which(nuevo_catalogo$`NOMBRE DEL PDV`=="town center nicolas romero")
```

```
## [1] 135 446
```

```
#135 446
```

Paso 14: Eliminar tiendas repetidas

```
nuevo_catalogo <- nuevo_catalogo[-c(1857, 1792, 1786, 1767, 1666,  1625, 1618, 1576, 1561, 655,  446, 43
```

Paso 15: Renombrar variables con valores en la columna de "nuevos nombres"

```
#Detectar los renglones que tienen todos los campos llenos, es decir, detectar los puntos de venta cuyo
completo_nuevo_nombre <- nuevo_catalogo[complete.cases(nuevo_catalogo), ] #1213
```

```
#Quitar primera columna para quedarse con el nuevo nombre únicamente
completo_nuevo_nombre <-completo_nuevo_nombre %>% select(-`NOMBRE DEL PDV`, -repetidas)
#Renombrar las variables
names(completo_nuevo_nombre)<-c("punto_de_venta", "estado","ciudad", "latitud","longitud","zona")
```

```
#renglones restantes de puntos de venta que no tienen nuevo nombre
nombre_sin_cambio<- nuevo_catalogo[!complete.cases(nuevo_catalogo), ]
```

```
nombre_sin_cambio <-nombre_sin_cambio %>% select(-`NUEVO NOMBRE DEL PDV (Operacione`, -repetidas)
names(nombre_sin_cambio)<-c("punto_de_venta", "estado","ciudad", "latitud","longitud","zona")
```

```
#Con los dos casos anteriores ahora ya se pueden juntar los dos conjuntos de datos
catalogo_final <- rbind(nombre_sin_cambio, completo_nuevo_nombre) %>% arrange(punto_de_venta) #2786
```

```
#Detectar si algun punto de venta se repite, y si hay tres que se repiten
a <-catalogo_final %>% select(punto_de_venta) %>% group_by(punto_de_venta)%>%unique() #hay uno que se r
a <- catalogo_final %>% mutate("repetidas" = 1)
a <- a %>% select(punto_de_venta, repetidas) %>% group_by(punto_de_venta) %>% summarise(n = sum(repetida
head(a,3)
```

```
## # A tibble: 3 x 2
##   punto_de_venta                 n
##   <chr>                      <dbl>
## 1 bca plaza canada huehuetoca     2
## 2 ksk pachuca galerias            2
## 3 tda arandas                     2
```

```
#Determinar si se deben de conservar o eliminar los puntos de venta que se repiten
#which(catalogo_final$punto_de_venta=="tda arandas")
catalogo_final[c(2018, 2019),] #remover una
```

```
## # A tibble: 2 x 6
##   punto_de_venta estado  ciudad  latitud longitud zona
##   <chr>          <chr>   <chr>     <dbl>    <dbl> <chr>
## 1 tda arandas    jalisco arandas    20.7    -102. centro occidente
## 2 tda arandas    jalisco arandas    20.7    -102. centro occidente
```

```
#which(catalogo_final$punto_de_venta=="bca plaza canada huehuetoca")
catalogo_final[c(255, 256),] #remover una
```

```
## # A tibble: 2 x 6
##   punto_de_venta          estado        ciudad    latitud longitud zona
##   <chr>                   <chr>         <chr>       <dbl>    <dbl> <chr>
## 1 bca plaza canada huehu~ estado de me~ santa te~    19.8    -99.2 centro ~
## 2 bca plaza canada huehu~ estado de me~ huehueto~    21.1   -102.  centro ~
```

```
#which(catalogo_final$punto_de_venta=="ksk pachuca galerias")
catalogo_final[c(1022, 1023),] #remover una
```

```
## # A tibble: 2 x 6
##   punto_de_venta       estado  ciudad  latitud longitud zona
##   <chr>                <chr>   <chr>     <dbl>    <dbl> <chr>
## 1 ksk pachuca galerias hidalgo pachuca    20.1    -98.8 centro sur
## 2 ksk pachuca galerias hidalgo pachuca    20.1    -98.8 centro sur
```

```
#Remover 3 puntos de venta que se repiten
catalogo_final <- catalogo_final[c(-2019, -1022, -255),]
```

Paso 16: Guardar el catalogo ya limpio como "CATALOGO_FINAL.csv"

```
#write.csv(catalogo_final, file="1_CATALOGO_FINAL.csv", row.names = FALSE)
```

Paso 17: Hacer un resumen general de la información con la que se cuenta.

1. Hay 2,783 puntos de venta distintos.

```
puntos_de_venta <- catalogo_final%>%select(punto_de_venta)%>%unique()
nrow(puntos_de_venta)
```

```
## [1] 2783
```

2. Hay 32 estados.

```
estados <- catalogo_final%>%select(estado)%>%unique()
nrow(estados)
```

```
## [1] 32
```

3. Hay 301 ciudades.

```
ciudades <- catalogo_final%>%select(ciudad)%>%unique()
nrow(ciudades)
```

```
## [1] 301
```

4. Hay 8 regiones económicas.

```r
zonas <- catalogo_final%>%select(zona)%>%unique()
nrow(zonas)
```

```
## [1] 8
```

1.2 Limpieza de Datos: Registro de ventas

Paso 1: Cargar el archivo con la información de los registros de ventas de la empresa en cuestión.

```r
#Lectura de datos
SalesData <- read_csv("SD.csv") #10 variables - 1,048,575 observaciones
```

```r
dim(SalesData)
```

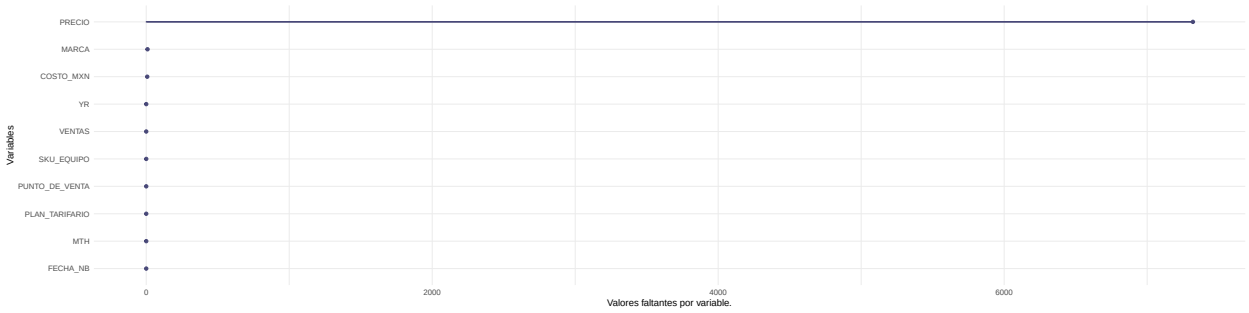
```
## [1] 1048575      10
```

```r
names(SalesData)
```

```
##  [1] "PUNTO_DE_VENTA" "PLAN_TARIFARIO" "SKU_EQUIPO"     "FECHA_NB"
##  [5] "PRECIO"         "COSTO_MXN"      "MARCA"          "VENTAS"
##  [9] "MTH"            "YR"
```

Paso 2: Detectar valores faltantes

```r
valores_faltantes_1 <- gg_miss_var(SalesData) + labs(y = "Valores faltantes por variable.")
valores_faltantes_1
```




```r
print("Valores faltantes de la variable Precio: ")
```

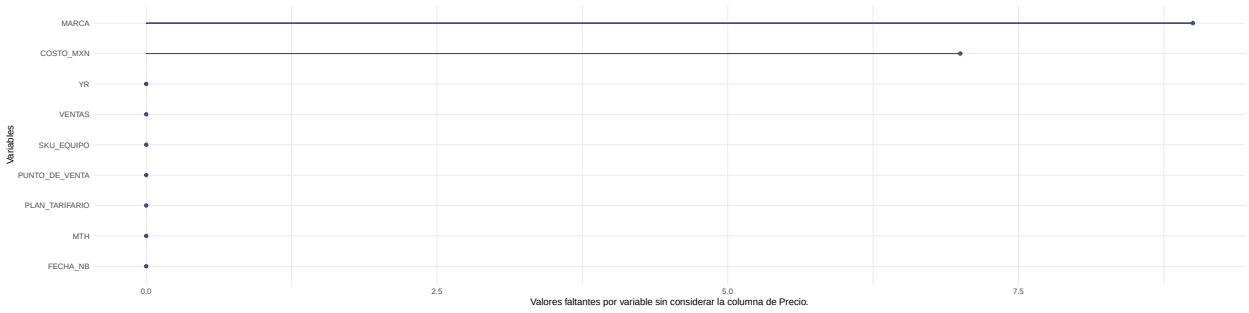
```
## [1] "Valores faltantes de la variable Precio: "
```

```r
sum(is.na(SalesData$PRECIO))
```

```
## [1] 7320
```

La gráfica considera las 10 variables y muestra que la variable de precio tiene varios valores faltantes (7,320). Sin embargo, es posible que dado el valor de faltantes de esta variable, los valores correspondientes a las demás variables no pueden apreciarse, es por eso que es necesario realizar otra gráfica.

```
sin_precio <- SalesData %>% select(-PRECIO)
valores_faltantes_2 <- gg_miss_var(sin_precio, show_pct = FALSE) + labs(y = "Valores faltantes por varia
valores_faltantes_2
```




```
print("Valores faltantes de la variable Marca: ")
```

```
## [1] "Valores faltantes de la variable Marca: "
```

```
sum(is.na(SalesData$MARCA))
```

```
## [1] 9
```

```
print("Valores faltantes de la variable Costo: ")
```

```
## [1] "Valores faltantes de la variable Costo: "
```

```
sum(is.na(SalesData$COSTO_MXN))
```

```
## [1] 7
```

Ahora se pueden apreciar los valores faltantes de las demás variables, dado que se excluyó la variable *Precio*. Se puede obsevar que al igual que la variable Precio, las variables *Marca* y *Costo* también tienen registros faltantes. Cabe mencionar que es de suma importancia identificar estos valores para poder tomar medidas màs adelantes y no agregar ruido o desechar información.

Paso 3: Resumen general de los datos tal cual están.

```
summary(SalesData)
```

```
##  PUNTO_DE_VENTA     PLAN_TARIFARIO      SKU_EQUIPO
##  Length:1048575     Length:1048575     Length:1048575
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    FECHA_NB             PRECIO           COSTO_MXN        MARCA
##  Length:1048575     Min.   :   -0.24   Min.   :    0   Length:1048575
```

```
##  Class :character   1st Qu.: 2400.00   1st Qu.: 2240   Class :character
##  Mode  :character   Median : 4500.00   Median : 3621   Mode  :character
##                     Mean   : 6486.82   Mean   : 5034
##                     3rd Qu.: 7200.00   3rd Qu.: 5391
##                     Max.   :37200.00   Max.   :32069
##                     NA's   :7320       NA's   :7
##      VENTAS        MTH              YR
##  Min.   :1   Min.   : 1.000   Min.   :2018
##  1st Qu.:1   1st Qu.: 6.000   1st Qu.:2018
##  Median :1   Median : 8.000   Median :2018
##  Mean   :1   Mean   : 7.447   Mean   :2018
##  3rd Qu.:1   3rd Qu.:11.000   3rd Qu.:2018
##  Max.   :1   Max.   :12.000   Max.   :2019
##
```

Paso 4: Detectar problemas de calidad en los datos

- La fechas no tienen formato de fecha y su formato de registro difiere.
- La segunda columna, correspondiente a la variable: Plan tarifario, posee caracteres especiales.
- Hay marcas escritas de diferente manera.
- Los registros están en mayúsculas y minúsculas.
- Existen valores faltantes que deben tratarse.
- Hay tiendas que no existen en el catálogo, es decir, que están registradas con un nombre erroneo.

Paso 5: Imputar valores faltantes: Marca y Costo

Tras una junta con el experto en la industria, se comentó que emplear se podía emplear cualquier columna entre "costo" y "precio", es por eso, que por facilidad, se escoge la variable de costo y se procede a imputar sus valores faltantes junto con los de la variable marca.

```
#Líneas con NA en la columna de Marca
which(is.na(SalesData$MARCA))
```

```
## [1] 143353 156986 156987 161483 161484 161485 161486 166702 169806
```

```
#Mostrar los sku a buscar para asociar una marca a estos en celdas con registros vacios
SalesData[c(143353, 156986, 156987, 161483, 161484, 161485, 161486, 166702, 169806), ] %>% group_by(SKU
```

```
## # A tibble: 5 x 1
## # Groups:   SKU_EQUIPO [5]
##   SKU_EQUIPO
##   <chr>
## 1 N.IPAP12256G
## 2 N.HL675PP
## 3 N.HL675PG
## 4 N.HL675PD
## 5 N.SS9P128A
```

```
#SKs a buscar
target <- c("N.IPAP12256G","N.HL675PP","N.HL675PG","N.HL675PD","N.SS9P128A")
#Tabla con marcas y SKUs
tail(SalesData%>%filter(SKU_EQUIPO %in% target) %>% group_by(SKU_EQUIPO) %>% select(SKU_EQUIPO, MARCA)
```

```
## # A tibble: 5 x 2
## # Groups:   SKU_EQUIPO [4]
##   SKU_EQUIPO   MARCA
##   <chr>        <chr>
## 1 N.HL675PG    Hisense
## 2 N.SS9P128A   Samsung
## 3 N.HL675PP    Hisense
## 4 N.IPAP12256G Apple
## 5 N.IPAP12256G Apple iP
```

```
#Imputar valores faltantes de marca
SalesData[(SalesData$SKU_EQUIPO=="N.IPAP12256G"), "MARCA"] <- "Apple"
SalesData[(SalesData$SKU_EQUIPO=="N.HL675PP"), "MARCA"] <- "Hisense"
SalesData[(SalesData$SKU_EQUIPO=="N.HL675PG"), "MARCA"] <- "Hisense"
SalesData[(SalesData$SKU_EQUIPO=="N.HL675PD"), "MARCA"] <- "Hisense"
SalesData[(SalesData$SKU_EQUIPO=="N.SS9P128A"), "MARCA"] <- "Samsung"
```

```
which(is.na(SalesData$MARCA))
```

```
## integer(0)
```

```
#Líneas con NA en la columna de costo
which(is.na(SalesData$COSTO_MXN))
```

```
## [1] 322071 322072 322073 325249 325250 325251 325252
```

```
#Mostrar los sku a buscar para asociar una marca a estos en celdas con registros vacios
SalesData[c(322071, 322072, 322073, 325249, 325250, 325251, 325252), ] %>% group_by(SKU_EQUIPO) %>% sel
```

```
## # A tibble: 2 x 1
## # Groups:   SKU_EQUIPO [2]
##   SKU_EQUIPO
##   <chr>
## 1 N.SNOTE9A
## 2 N.SNOTE9N
```

```
#SKs a buscar
target <- c("N.SNOTE9A","N.SNOTE9N")
#Tabla con marcas y SKUs y además se buscan por separado los diferentes puntos de venta y se intenta en
imputar_costos <- SalesData%>%filter(SKU_EQUIPO %in% target)%>%group_by(SKU_EQUIPO)
```

```
SalesData[322071, 6] <- 15932
SalesData[322072, 6] <- 18103.45
SalesData[322073, 6] <- 14369
SalesData[325249, 6] <- 15932
SalesData[325250, 6] <- 15697
SalesData[325251, 6] <- 15932
SalesData[325252, 6] <- 15697
```

```r
which(is.na(SalesData$COSTO_MXN))
```

```
## integer(0)
```

Paso 6: Homogeneizar registros de fecha.

```r
#Cambiar formato de registros
SalesData$FECHA_NB<-str_replace(SalesData$FECHA_NB,"jun", "06")%>%
  str_replace("jul", "07") %>%
  str_replace("AUG", "-08-") %>%
  str_replace("sep", "09") %>%
  str_replace("oct", "10") %>%
  str_replace("nov", "11") %>%
  str_replace("DEC", "-12-") %>%
  str_replace("JAN", "-01-") %>%
  str_replace("feb", "02") %>%
  str_replace("mar", "03") %>%
  str_replace("-18", "-2018") %>%
  str_replace("-19", "-2019")
```

```r
#Convertir a fecha
SalesData$FECHA_NB<-as.Date(SalesData$FECHA_NB, "%d-%m-%Y")
class(SalesData$FECHA_NB)
```

```
## [1] "Date"
```

Una vez que la columna de fecha ya esta en el formato necesario, se procede con ordenarlos de tal manera que todos los registros que se efectuaron en un día estén juntos.

```r
SalesData <- SalesData %>% arrange(FECHA_NB)
```

```r
head(SalesData, 2)
```

```
## # A tibble: 2 x 10
##   PUNTO_DE_VENTA PLAN_TARIFARIO SKU_EQUIPO FECHA_NB   PRECIO COSTO_MXN
##   <chr>          <chr>          <chr>      <date>      <dbl>     <dbl>
## 1 KSK PLAZA MAZ~ COMPÁRTELO IN~ N.A5054SD  2018-06-01   1200      633.
## 2 STR HERMOSILL~ COMPÁRTELO IN~ N.A5054SD  2018-06-01   1200      633.
## # ... with 4 more variables: MARCA <chr>, VENTAS <dbl>, MTH <dbl>,
## #   YR <dbl>
```

```r
tail(SalesData, 2)
```

```
## # A tibble: 2 x 10
##   PUNTO_DE_VENTA PLAN_TARIFARIO SKU_EQUIPO FECHA_NB   PRECIO COSTO_MXN
##   <chr>          <chr>          <chr>      <date>      <dbl>     <dbl>
## 1 TDA CDMX ATIZ~ COMPÁRTELO IN~ N.MOG6PLI  2019-03-31   5400      3750
## 2 TDA CDMX PERI~ COMPÁRTELO IN~ N.MOG6PLN  2019-03-31   5400      3750
## # ... with 4 more variables: MARCA <chr>, VENTAS <dbl>, MTH <dbl>,
## #   YR <dbl>
```

Con esto se pude observar que los registros van desde el primero de junio del 2018 hasta el 31 de marzo del año en curso, abarcando un total de 10 meses.

Paso 7: Homogeneizar la columna relacionada con la marca.

```
#Valores únicos de la columna marca_sencilla
marcas_original<- as.data.frame(unique(SalesData$MARCA))
#Cambiar nombre de columna
names(marcas_original)[1]<-"Marcas_original"
#convertir a lista para mejor visualización
as.list(marcas_original)
```

```
## $Marcas_original
##  [1] Alcatel  Huawei   Hisense  Apple    Lenovo   Lanix    Motorola
##  [8] Samsung  Sony     Affix    ZTE      LG       APPLE    HUAWEI
## [15] Huawei P Huawei Y Huawei M Apple iP Huawei N Apple Ip ZTE Blad
## [22] ZTE V8 M Lanix X5 Sony Xpe Huawei G LG X Max LG X Cam Huawei T
## [29] Sony XA  LG X Scr Sony M5  Affix V1
## 32 Levels: Affix Affix V1 Alcatel Apple APPLE Apple iP ... ZTE V8 M
```

Se tenía un total de 32 marcas,sin embargo, una vez identificado cuáles eran las marcas mal registradas se hace la homogenizaciòn de la siguente manera.

```
#Creación de una nueva variable en el dataframme
SalesData$MARCA_SENCILLA <- SalesData$MARCA
#Homogenizar Variables
SalesData$MARCA_SENCILLA<-str_replace(SalesData$MARCA_SENCILLA, "APPLE", "Apple") %>%
  str_replace("Apple iP", "Apple")%>%
  str_replace("Apple Ip", "Apple")%>%
  str_replace("HUAWEI", "Huawei")%>%
  str_replace("Huawei P", "Huawei")%>%
  str_replace("Huawei Y", "Huawei")%>%
  str_replace("Huawei M", "Huawei")%>%
  str_replace("Huawei N", "Huawei")%>%
  str_replace("Huawei T", "Huawei")%>%
  str_replace("Huawei G", "Huawei")%>%
  str_replace("ZTE Blad", "ZTE")%>%
  str_replace("ZTE V8 M", "ZTE")%>%
  str_replace("Lanix X5", "Lanix")%>%
  str_replace("Sony Xpe", "Sony")%>%
  str_replace("Sony XA", "Sony")%>%
  str_replace("Sony M5", "Sony")%>%
  str_replace("LG X Max", "LG")%>%
  str_replace("LG X Cam", "LG")%>%
  str_replace("LG X Scr", "LG")%>%
  str_replace("Affix V1", "Affix")
```

```
#Valores únicos de la columna marca_sencilla
marcas_sencillas<- as.data.frame(unique(SalesData$MARCA_SENCILLA))
#Cambiar nombre de columna
names(marcas_sencillas)[1]<-"Marcas_Sencillas"
as.list(marcas_sencillas)
```

```
## $Marcas_Sencillas
##  [1] Alcatel  Huawei   Hisense  Apple    Lenovo   Lanix    Motorola
##  [8] Samsung  Sony     Affix    ZTE      LG
## 12 Levels: Affix Alcatel Apple Hisense Huawei Lanix Lenovo LG ... ZTE
```

Ahora se cuenta con únicamente 12 marcas.

Paso 8: Pasar a minúsculas las columnas que lo ameriten.

```
SalesData$PUNTO_DE_VENTA <- tolower(SalesData$PUNTO_DE_VENTA)
SalesData$PLAN_TARIFARIO <- tolower(SalesData$PLAN_TARIFARIO)
SalesData$MARCA <- tolower(SalesData$MARCA)
SalesData$MARCA_SENCILLA <- tolower(SalesData$MARCA_SENCILLA)
```

Paso 9: Quitar caracteres especiales

```
#Remover caracteres especiales en cada columna
SalesData$PUNTO_DE_VENTA <- str_replace(SalesData$PUNTO_DE_VENTA , "á", "a") %>%
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")%>%
  str_replace(" - ", " ")%>%
  str_replace("-"," ")%>%
  str_replace("  "," ")

SalesData$PLAN_TARIFARIO <- str_replace(SalesData$PLAN_TARIFARIO , "á", "a") %>%
  str_replace("é","e")%>%
  str_replace("í", "i") %>%
  str_replace("ó","o")%>%
  str_replace("ú", "u")%>%
  str_replace("ñ","n")%>%
  str_replace(" - ", " ")%>%
  str_replace("-"," ")%>%
  str_replace("  "," ")
```

Paso 10: Renombrar Columnas

```
names(SalesData)<-c("punto_de_venta","plan_tarifario","sku_por_equipo","fecha","precio","costo","marca"
```

Paso 11: Cambiar el nombre de algunos puntos de venta

Este fue el segundo proceso más pesado de la limpieza de datos, el primero fue el de buscar los puntos de venta faltantes en el catalogo. Este paso fue pesado dado que gran parte de él tuvo que ser artesal, es decir, la búsqueda de los puntos de venta que no hacian match con el catalogo se tenían que revisar uno por uno para poderse asociar con un punto de venta ya registrado.

Para poder cumplir eso, lo primero que se hace es generar un archivo de excel independiente para sustituir los valores de la primera columna (valores que no hacian empate con el catalogo) por el valor de la segunda columna (el punto de venta al que correspondian en el catalogo).

```r
#leer archivo de excel independiente con los nombres de puntos de venta a sustituir
nuevos_nombres<- read_csv("nuevos_nombres.csv")
```

```r
dim(nuevos_nombres)
```

```
## [1] 2248    2
```

```r
nuevos_nombres <- nuevos_nombres  %>% arrange(punto_de_venta)
head(nuevos_nombres, 2)
```

```
## # A tibble: 2 x 2
##   punto_de_venta   punto_de_venta_nuevo
##   <chr>            <chr>
## 1 5 de mayo zmm    5 de mayo zmm
## 2 abasolo coahuila tda saltillo ii
```

```r
#ordenar el archivo con los registros de venta de acuerdo al punto de venta
SalesData <- SalesData %>% arrange(punto_de_venta)
SalesData <- as.data.frame(SalesData)
```

```r
#limpiar salesdata columna 1 de espacios extraños
pdv_limpios_sin_espacios <- SalesData %>% select(punto_de_venta)
pdv_limpios_sin_espacios <- as.vector(pdv_limpios_sin_espacios$punto_de_venta)
z <-gsub("\u00A0", " ", pdv_limpios_sin_espacios, fixed = TRUE)
showNonASCII(z)
punto_de_venta_limpio <- as.data.frame(z)

SalesData_limpio <- cbind(SalesData, punto_de_venta_limpio)
SalesData_limpio <- SalesData_limpio %>% select(z, plan_tarifario, sku_por_equipo, fecha, costo, marca,
names(SalesData_limpio)[1] <- "punto_de_venta"
SalesData_limpio <- SalesData_limpio %>% arrange(punto_de_venta)
```

```r
#Juntar el archivo de excel con los nuevos nombres a sustituit con el archivo que incluye los registros
salesdata_limpio_final <- left_join(SalesData_limpio, nuevos_nombres, by="punto_de_venta")
```

```r
#reacomodar variables
salesdata_limpio_final <- salesdata_limpio_final %>% select(punto_de_venta, punto_de_venta_nuevo, plan_
#salesdata_limpio_final
```

Una vez que eso ya esta listo, se procede a sustituir los puntos de venta por su nuevo nombre.

```r
#Seleccionar los renglones que tienen todas sus columnas llenas, dado que estos son los puntos de venta
todos_los_campos_cambiar_nombre<- salesdata_limpio_final[complete.cases(salesdata_limpio_final), ]
todos_los_campos_cambiar_nombre <- todos_los_campos_cambiar_nombre %>% select(-punto_de_venta)
names(todos_los_campos_cambiar_nombre)<-c("punto_de_venta", "plan_tarifario", "sku_por_equipo", "fecha"
```

```r
#campos que respetan nombre, campos que no requieren de modificaciones.
no_cambiar_nombre<- salesdata_limpio_final[!complete.cases(salesdata_limpio_final), ]
no_cambiar_nombre <- no_cambiar_nombre%>% select(-punto_de_venta_nuevo)
```

```r
#juntar ambos conjuntos de datos
ventas_limpio <- rbind(todos_los_campos_cambiar_nombre, no_cambiar_nombre)
ventas_limpio <- ventas_limpio %>% arrange(punto_de_venta)
head(ventas_limpio,10)
```

```
##    punto_de_venta                       plan_tarifario sku_por_equipo
## 1       1 poniente compartelo incluido voz, sms y datos      N.HUM10LN
## 2       1 poniente compartelo incluido voz, sms y datos      N.MZ2PLYG
## 3       1 poniente compartelo incluido voz, sms y datos       N.ZV8MNG
## 4       1 poniente compartelo incluido voz, sms y datos       N.ZV8MNG
## 5       1 poniente compartelo incluido voz, sms y datos      N.HUAP20N
## 6       1 poniente compartelo incluido voz, sms y datos     N.HUY9FLAZ
## 7       1 poniente compartelo incluido voz, sms y datos    N.HUY9FLADR
## 8       1 poniente compartelo incluido voz, sms y datos      N.MOTOCNG
## 9       1 poniente compartelo incluido voz, sms y datos       N.ZV8MNG
## 10      1 poniente compartelo incluido voz, sms y datos     N.HUY9FLAZ
##          fecha costo    marca ventas mes anio marca_modificada
## 1   2018-06-01  4300   huawei      1   6 2018           huawei
## 2   2018-06-01  6000 motorola      1   6 2018         motorola
## 3   2018-06-01  2622      zte      1   6 2018              zte
## 4   2018-06-01  2622      zte      1   6 2018              zte
## 5   2018-06-02  8740   huawei      1   6 2018           huawei
## 6   2018-06-02  3167   huawei      1   6 2018           huawei
## 7   2018-06-04  3090   huawei      1   6 2018           huawei
## 8   2018-06-04  1036 motorola      1   6 2018         motorola
## 9   2018-06-04  2622      zte      1   6 2018              zte
## 10 2018-06-05  3090   huawei      1   6 2018           huawei
```

```r
#se limpia de espacios extraños la primera columna
ventas_limpio_2 <- ventas_limpio %>% select(punto_de_venta)
ventas_limpio_2 <- as.vector(ventas_limpio_2$punto_de_venta)
z <-gsub("\u00A0", " ", ventas_limpio_2, fixed = TRUE)
showNonASCII(z)
punto_de_venta_limpio <- as.data.frame(z)
```

```r
#la columna limpia debe de juntarse en el dataframe
archivo_final_ventas_limpio <- cbind(ventas_limpio, punto_de_venta_limpio)
archivo_final_ventas_limpio <- archivo_final_ventas_limpio %>% select(z, plan_tarifario, sku_por_equipo
names(archivo_final_ventas_limpio)[1] <- "punto_de_venta"
```

A este punto uno pensaría que los datos ya están listos y que ya se puede **combinar la información** proporcionada tanto en el catálogo como en el archivo de los registros, sin embargo, al intentar hacer el join de estos dos documentos, se tienen valores faltantes relacionados a puntos de venta que aún no cuadran dentro del catálogo, es por eso que se ejecuta el siguiente código para renombrar manualmente las variables que se detectaron que no hacían match.

```r
archivo_final_ventas_limpio$punto_de_venta <- str_replace(archivo_final_ventas_limpio$punto_de_venta, "
  str_replace("ksk plaza camelinas morelias", "ksk plaza camelinas morelia")%>%
  str_replace("ovd multiplazaizcalli", "ovd multiplaza izcalli")%>%
  str_replace("pre cv sur campeche chedraui gobernadore", "pre cv sur campeche chedraui gobernadores")%
  str_replace("pre cv sur campeche chedraui gobernadoress", "pre cv sur campeche chedraui gobernadores"
  str_replace("chedraui anfora", "exp chedraui anfora")%>%
```

```
  str_replace("exp exp chedraui anfora", "exp chedraui anfora")%>%
  str_replace("chedraui mundo e", "arsa mundo e")%>%
  str_replace("fgt plaza sendero mazatlan", "ksk mazatlan plaza sendero")%>%
  str_replace("fgt plaza sahuaro", "mpdv hermosillo ley sahuaro")%>%
  str_replace("ksk cdmx televisa chapultepec", "asociados 5")%>%
  str_replace("ksk cdmx tvazteca", "ksk cdmx tv azteca")%>%
  str_replace("ksk leon plaza mayor isla", "ksk leon plaza mayor")%>%
  str_replace("ksk m de la torre veracruz chedraui", "tda m de la torre veracruz ignacio zaragoza")%>%
  str_replace("ksk m de la torre veracruz chedraui", "tda m de la torre veracruz ignacio zaragoza")%>%
  str_replace("mpdv cd del carmen walmart", "pre cvsur cd del carmen plaza real")%>%
  str_replace("mpdv cdmx ba los reyes", "business sendero ecatepec")%>%
  str_replace("mpdv cdmx chedraui heroes tecamac ii", "gnt macroplaza tecamac mex")%>%
  str_replace("mpdv cdmx sams ecatepec centro", "bca patio ecatepec")%>%
  str_replace("mpdv cdmx sams san jose tecamac", "cei cen power center tecamac mex")%>%
  str_replace("mpdv chihuahua soriana juventud", "tda chihuahua industrias plaza sendero")%>%
  str_replace("mpdv cuernavaca walmart jiutepec", "red celular jiutepec")%>%
  str_replace("mpdv culiacan sams culiacan", "ksk culiacan plaza forum")%>%
  str_replace("mpdv durango soriana city club", "tda durango i")%>%
  str_replace("mpdv morelia ba morelia este", "tda moreliami plaza")%>%
  str_replace("mpdv morelia walmart estadio", "ksk morelia plaza la huerta")%>%
  str_replace("mpdv tuxtla gutierrez walmart belisario", "str 11 pte 1173 tgz")%>%
  str_replace("mpdv tampico walmart", "mpdv tampico walmart alijadores")%>%
  str_replace("mpdv tampico walmart alijadores ciudad madero", "mpdv tampico walmart alijadores")%>%
  str_replace("mpdv tampico walmart alijadores alijadores", "mpdv tampico walmart alijadores")%>%
  str_replace("mpdv tula de allende ba", "codite hidalgo gerardo salinas calle sur")%>%
  str_replace("mpdv zamora soriana", "tda zamora jacona")%>%
  str_replace("pos boulevard circunvalacion", "epc oficinas corporativo pue")%>%
  str_replace("sid ley tres rios cul", "tda culiacan paseo san isidro")%>%
  str_replace("tda culiacan u de o", "ksk culiacan plaza forum")%>%
  str_replace("ksk los mochis plaza los mochis", "tda los mochis")%>%
  str_replace("mpdv cdmx ba fuentes del valle", "gnt plaza bella mexiquense")%>%
  str_replace("mpdv cdmx chedraui coacalco", "exp multiplaza coacalco")%>%
  str_replace("mpdv cdmx sams cortijo", "mpdv cdmx cm texcoco centro")%>%
  str_replace("mpdv cdmx sams las americas", "tda cdmx paseo ventura")%>%
  str_replace("mpdv cuernavaca comer mex jacarandas", "tda cuernavaca plaza norte")%>%
  str_replace("prm tda cuernavaca plaza norte", "tda cuernavaca plaza norte")%>%
  str_replace("mpdv culiacan bodega aurrera estadio", "fgt sendero culiacan")%>%
  str_replace("mpdv delicias soriana poniente", "tda delicias")%>%
  str_replace("mpdv merida sams aviacion", "ba itzaes")%>%
  str_replace("mpdv merida walmart itzimna", "macroplaza merida")%>%
  str_replace("mpdv san luis potosi walmart munoz", "mpdv san luis potosi walmart")%>%
  str_replace("mpdv veracruz walmart playa norte", "tda veracruz veracruz rafael cuervo")%>%
  str_replace("ovd multiplaza izcalli", "tda cdmx cuautitlan izcalli")%>%
  str_replace("ksk los mochis plaza los mochis", "tda los mochis")%>%
  str_replace("ksk los mochis plaza los mochis", "tda los mochis")%>%
  str_replace("ksk los mochis plaza los mochis", "tda los mochis")
```

Paso 12: Guardar el archivo ya limpio

```
#write.csv(archivo_final_ventas_limpio, file="SD_LIMPIO.csv", row.names = FALSE)
```

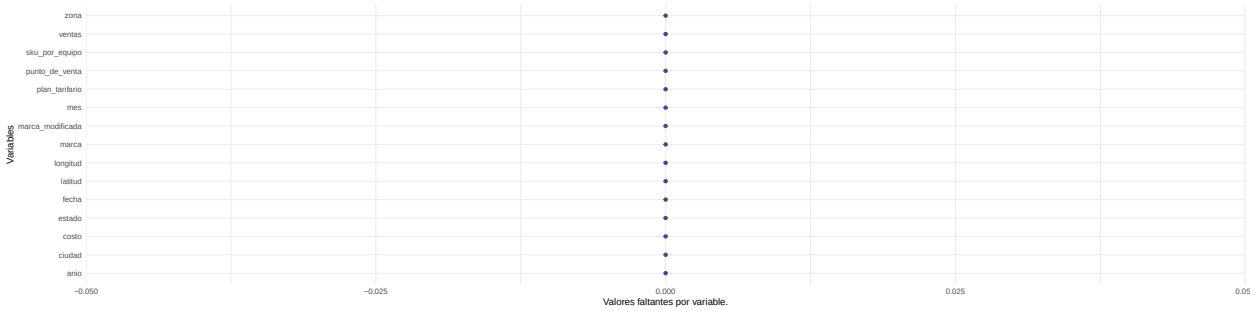
1.3 Juntar catálogo con registro de ventas

Paso 1: Hacer un **left_join** de ambos conjuntos de datos

```
#juntar archivos
a <- left_join(archivo_final_ventas_limpio, catalogo_final, by="punto_de_venta")
b <- a
```

Paso 2: Chear que no haya valores faltantes, comprobando que el join se hizo adecuadamente

```
#Comprobar que no hay valores faltantes
valores_faltantes_8 <- gg_miss_var(b) + labs(y = "Valores faltantes por variable.")
valores_faltantes_8
```



Paso 3: Tras tanto esfuerzo en la limpieza de datos, se notó que había ciudades mal escritas, por ende, se procede a homogeneizarlas.

```
a$ciudad<-str_replace(a$ciudad,"atlacomulco de fabela", "atlacomulco")%>%
  str_replace("cd. cuauhtemoc", "ciudad cuahutemoc") %>%
  str_replace("cd. del carmen", "ciudad del carmen") %>%
  str_replace("cd. guzman", "ciudad guzman") %>%
  str_replace("guzman", "ciudad guzman") %>%
  str_replace("cd. juarez", "ciudad juarez") %>%
  str_replace("juarez", "ciudad juarez") %>%
  str_replace("ciudad de mexico", "cdmx") %>%
  str_replace("coacalco de berriozabal", "coacalco") %>%
  str_replace("cuautitlan izcalli", "cuautitlan") %>%
  str_replace("dolores hidalgo", "hidalgo") %>%
  str_replace("edo. mex", "naucalpan") %>%
  str_replace("edo. mex.", "naucalpan") %>%
  str_replace("mexicali", "mexicalli") %>%
  str_replace("naucalpan de juarez", "naucalpan") %>%
  str_replace("ocozocoautla de espinosa", "ocozocoautla") %>%
  str_replace("pedras negras", "piedras negras") %>%
  str_replace("poza rrica de hidalgo", "poza rica") %>%
  str_replace("san francsico del rincon", "san francisco de rincon") %>%
  str_replace("san pedro graza garcia", "san pedro garza garcia") %>%
  str_replace("soledad de graciano", "soledad") %>%
  str_replace("texcoco de mora", "texcoco") %>%
```

```
  str_replace("tlajomulco de zuniga", "tlajomulco") %>%
  str_replace("tlalnepantla de baz", "tlalneplantla") %>%
  str_replace("tuxtla", "tuxtla gutierrez") %>%
  str_replace("zamora de hidalgo", "zamora") %>%
  str_replace("jalapa","xalapa")%>%
  str_replace("ecatepec de morelos", "ecatepec")
```

Paso 4: Descargar el archivo completo final y limpio

```
#write.csv(a,file="VENTAS_CON_UBICACION_LIMPIO.csv", row.names=FALSE)
```

```
names(a)
```

```
##  [1] "punto_de_venta"   "plan_tarifario"   "sku_por_equipo"
##  [4] "fecha"            "costo"            "marca"
##  [7] "ventas"           "mes"              "anio"
## [10] "marca_modificada" "estado"           "ciudad"
## [13] "latitud"          "longitud"         "zona"
```

Paso 5: Guardar una lista con los distintos puntos de venta de los que se tiene registro.

```
PUNTOS_DE_VENTA <- a%>%select(punto_de_venta)%>%group_by(punto_de_venta)%>%unique() #1909
#write.csv(PUNTOS_DE_VENTA, file="PUNTOS_DE_VENTA.csv", row.names=FALSE)
```