Actividad 3 - Actividad grupal: Detección de anomalías y técnicas de agrupamiento

6/22/2020

Grupo 10: Integrantes:

Cortés Forero Leydi Milena

Saavedra Coneo Richard Camilo

Rodríguez Angarita Ramón

Zapata Llano Juan Sebastián

```
#install.packages("corrplot")
#install.packages("gmodels")
#install.packages("tidyverse")
#install.packages("readr")
#install.packages("caret")
#install.packages('Hmisc')
#install.packages("modeest")

# install.packages("e1071")
# install.packages("caTools")
# install.packages("h2o")
# install.packages("dygraphs")
# install.packages("dplyr")
# install.packages("DT")
```

```
library( h2o )
library(caTools)
library(e1071)
library(gmodels)
library(caret)
library(tidyverse)
library(Hmisc)
library(modeest)
library(cluster)
library(corrplot)
```

Se cargan los datos a la variable `data`

```
set.seed(1234)
data <- read_csv("data/datos.csv")
```

```
## Parsed with column specification:
## cols(
##   Merchant_id = col_double(),
##   `Transaction date` = col_logical(),
##   `Average Amount/transaction/day` = col_double(),
```

```
##   Transaction_amount = col_double(),
##   `Is declined` = col_character(),
##   `Total Number of declines/day` = col_double(),
##   isForeignTransaction = col_character(),
##   isHighRiskCountry = col_character(),
##   Daily_chargeback_avg_amt = col_double(),
##   `6_month_avg_chbk_amt` = col_double(),
##   `6-month_chbk_freq` = col_double(),
##   isFradulent = col_character()
## )
```

Se revisa si existen valores nulos.

```
str(data)
```

```
## tibble [3,075 x 12] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Merchant_id                   : num [1:3075] 3.16e+09 3.16e+09 3.16e+09 3.16e+09 3.16e+09 ...
##  $ Transaction date              : logi [1:3075] NA NA NA NA NA NA ...
##  $ Average Amount/transaction/day: num [1:3075] 100 100 186 186 500 ...
##  $ Transaction_amount            : num [1:3075] 3000 4300 4823 5008 26000 ...
##  $ Is declined                   : chr [1:3075] "N" "N" "Y" "Y" ...
##  $ Total Number of declines/day  : num [1:3075] 5 5 5 8 0 0 0 20 20 20 ...
##  $ isForeignTransaction          : chr [1:3075] "Y" "Y" "N" "N" ...
##  $ isHighRiskCountry             : chr [1:3075] "Y" "Y" "N" "N" ...
##  $ Daily_chargeback_avg_amt      : num [1:3075] 0 0 0 0 800 800 900 0 0 0 ...
##  $ 6_month_avg_chbk_amt          : num [1:3075] 0 0 0 0 677 ...
##  $ 6-month_chbk_freq             : num [1:3075] 0 0 0 0 6 6 7 0 0 0 ...
##  $ isFradulent                   : chr [1:3075] "Y" "Y" "Y" "Y" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Merchant_id = col_double(),
##   ..   `Transaction date` = col_logical(),
##   ..   `Average Amount/transaction/day` = col_double(),
##   ..   Transaction_amount = col_double(),
##   ..   `Is declined` = col_character(),
##   ..   `Total Number of declines/day` = col_double(),
##   ..   isForeignTransaction = col_character(),
##   ..   isHighRiskCountry = col_character(),
##   ..   Daily_chargeback_avg_amt = col_double(),
##   ..   `6_month_avg_chbk_amt` = col_double(),
##   ..   `6-month_chbk_freq` = col_double(),
##   ..   isFradulent = col_character()
##   .. )
```

Se hallan de las variables numéricas el valor mínimo máximo, la mediana y la media.

Se hallan de las variables categóricas las diferentes categorías y la frecuencia de cada una de ellas.

Se eliminó la variable Transaction date por tener todos los datos nulos.

```
data$`Transaction date` <- NULL
```

Volver las columnas categóricas a númericas

```
data$isFradulent <- as.factor(data$isFradulent)
data$isForeignTransaction <- as.factor(data$isForeignTransaction)
data$isHighRiskCountry <- as.factor(data$isHighRiskCountry)
data$`Is declined` <- as.factor(data$`Is declined`)
```

Se ejecuta la función summary para ver los valores máximos y mínimos de cada una de las variables numéricas, así como la media, se muestra también la frecuencia de las variables categóricas.

```
summary(data)
```

```
##   Merchant_id        Average Amount/transaction/day Transaction_amount
##  Min.   :3.160e+09   Min.   :   4.011               Min.   :     0
##  1st Qu.:4.171e+09   1st Qu.: 269.788               1st Qu.:  2409
##  Median :5.026e+09   Median : 502.550               Median :  6699
##  Mean   :5.027e+09   Mean   : 515.027               Mean   :  9876
##  3rd Qu.:5.890e+09   3rd Qu.: 765.273               3rd Qu.: 14423
##  Max.   :6.666e+09   Max.   :2000.000               Max.   :108000
##  Is declined Total Number of declines/day isForeignTransaction
##  N:3018      Min.   : 0.0000              N:2369
##  Y:  57      1st Qu.: 0.0000              Y: 706
##              Median : 0.0000
##              Mean   : 0.9574
##              3rd Qu.: 0.0000
##              Max.   :20.0000
##  isHighRiskCountry Daily_chargeback_avg_amt 6_month_avg_chbk_amt
##  N:2870            Min.   :  0.00           Min.   :  0.00
##  Y: 205            1st Qu.:  0.00           1st Qu.:  0.00
##                    Median :  0.00           Median :  0.00
##                    Mean   : 55.74           Mean   : 40.02
##                    3rd Qu.:  0.00           3rd Qu.:  0.00
##                    Max.   :998.00           Max.   :998.00
##  6-month_chbk_freq isFradulent
##  Min.   :0.0000    N:2627
##  1st Qu.:0.0000    Y: 448
##  Median :0.0000
##  Mean   :0.3919
##  3rd Qu.:0.0000
##  Max.   :9.0000
```

Se cambian los valores que estaban en caracter de Y y N reeemplazandolos por 1 y 0 respectivamente

```
data$isFradulent <- ifelse(data$isFradulent == "N",0,1)
data$`Is declined` <- ifelse(data$`Is declined` == "N",0,1)
data$isHighRiskCountry <- ifelse(data$isHighRiskCountry == "N",0,1)
data$isForeignTransaction<- ifelse(data$isForeignTransaction == "N",0,1)

str(data)
```

```
## tibble [3,075 x 11] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Merchant_id                   : num [1:3075] 3.16e+09 3.16e+09 3.16e+09 3.16e+09 3.16e+09 ...
##  $ Average Amount/transaction/day: num [1:3075] 100 100 186 186 500 ...
##  $ Transaction_amount            : num [1:3075] 3000 4300 4823 5008 26000 ...
##  $ Is declined                   : num [1:3075] 0 0 1 1 0 0 0 1 1 1 ...
##  $ Total Number of declines/day  : num [1:3075] 5 5 5 8 0 0 0 20 20 20 ...
##  $ isForeignTransaction          : num [1:3075] 1 1 0 0 1 1 0 0 0 0 ...
##  $ isHighRiskCountry             : num [1:3075] 1 1 0 0 1 1 0 0 0 0 ...
##  $ Daily_chargeback_avg_amt      : num [1:3075] 0 0 0 0 800 800 900 0 0 0 ...
##  $ 6_month_avg_chbk_amt          : num [1:3075] 0 0 0 0 677 ...
##  $ 6-month_chbk_freq             : num [1:3075] 0 0 0 0 6 6 7 0 0 0 ...
##  $ isFradulent                   : num [1:3075] 1 1 1 1 1 1 1 1 1 1 ...
##  - attr(*, "spec")=
```

```
##   .. cols(
##   ..   Merchant_id = col_double(),
##   ..   `Transaction date` = col_logical(),
##   ..   `Average Amount/transaction/day` = col_double(),
##   ..   Transaction_amount = col_double(),
##   ..   `Is declined` = col_character(),
##   ..   `Total Number of declines/day` = col_double(),
##   ..   isForeignTransaction = col_character(),
##   ..   isHighRiskCountry = col_character(),
##   ..   Daily_chargeback_avg_amt = col_double(),
##   ..   `6_month_avg_chbk_amt` = col_double(),
##   ..   `6-month_chbk_freq` = col_double(),
##   ..   isFradulent = col_character()
##   .. )
```

```
summary(data)
```

```
##   Merchant_id        Average Amount/transaction/day Transaction_amount
##  Min.   :3.160e+09   Min.   :   4.011               Min.   :     0
##  1st Qu.:4.171e+09   1st Qu.: 269.788               1st Qu.:  2409
##  Median :5.026e+09   Median : 502.550               Median :  6699
##  Mean   :5.027e+09   Mean   : 515.027               Mean   :  9876
##  3rd Qu.:5.890e+09   3rd Qu.: 765.273               3rd Qu.: 14423
##  Max.   :6.666e+09   Max.   :2000.000               Max.   :108000
##   Is declined      Total Number of declines/day isForeignTransaction
##  Min.   :0.00000   Min.   : 0.0000              Min.   :0.0000
##  1st Qu.:0.00000   1st Qu.: 0.0000              1st Qu.:0.0000
##  Median :0.00000   Median : 0.0000              Median :0.0000
##  Mean   :0.01854   Mean   : 0.9574              Mean   :0.2296
##  3rd Qu.:0.00000   3rd Qu.: 0.0000              3rd Qu.:0.0000
##  Max.   :1.00000   Max.   :20.0000              Max.   :1.0000
##  isHighRiskCountry Daily_chargeback_avg_amt 6_month_avg_chbk_amt
##  Min.   :0.00000   Min.   :  0.00           Min.   :  0.00
##  1st Qu.:0.00000   1st Qu.:  0.00           1st Qu.:  0.00
##  Median :0.00000   Median :  0.00           Median :  0.00
##  Mean   :0.06667   Mean   : 55.74           Mean   : 40.02
##  3rd Qu.:0.00000   3rd Qu.:  0.00           3rd Qu.:  0.00
##  Max.   :1.00000   Max.   :998.00           Max.   :998.00
##  6-month_chbk_freq  isFradulent
##  Min.   :0.0000    Min.   :0.0000
##  1st Qu.:0.0000    1st Qu.:0.0000
##  Median :0.0000    Median :0.0000
##  Mean   :0.3919    Mean   :0.1457
##  3rd Qu.:0.0000    3rd Qu.:0.0000
##  Max.   :9.0000    Max.   :1.0000
```

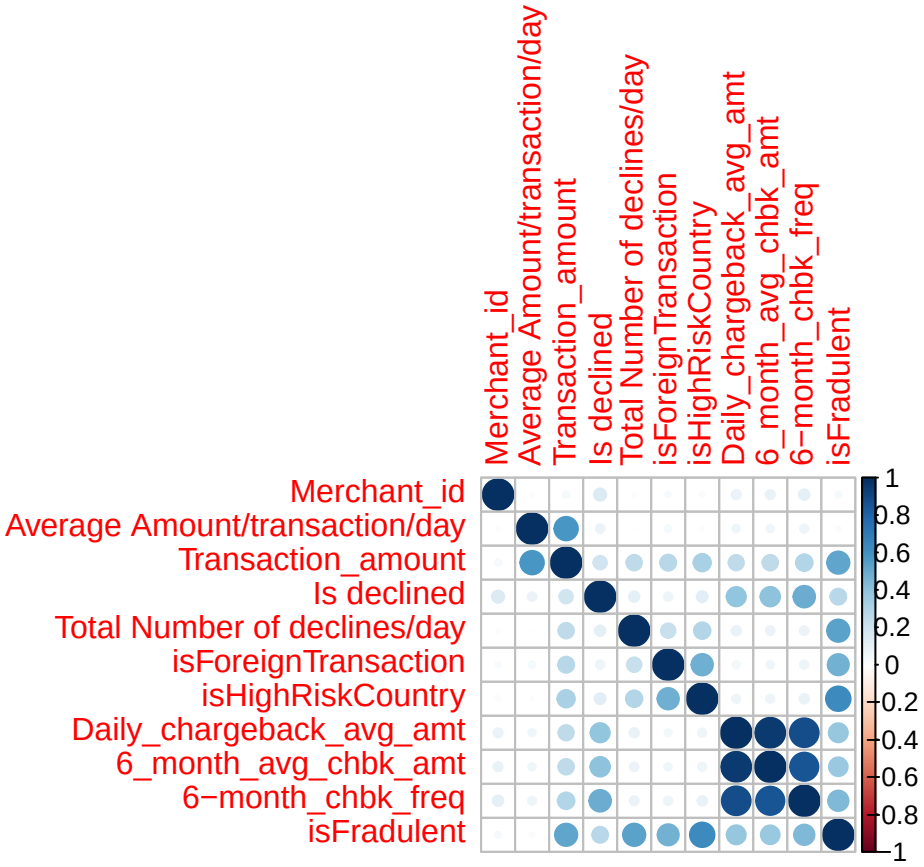
Se hallan la correlaciones existentes entre las variables del conjunto de datos mediante la matriz de correlación

```
corr <- cor(data[,])
corrGreater <- findCorrelation(corr, cutoff=0.8)
corrGreater
```

```
## [1] 10  8
```

Se grafica la Matriz de correlación

```
Mat_Correlacion <- cor(data, method = "pearson")
round(Mat_Correlacion, digits = 2)
corrplot(Mat_Correlacion)
```



Conjunto de datos de modelización y validación.

Se toman los datos para la modelización del 80'%' para train y él 20% para test.

```
train_data <- sample(nrow(data), 0.8 * nrow(data))
data_train <- data[train_data, ]
data_test <- data[-train_data, ]
```

```
h2o.init()
```

```
allData_hex = as.h2o( data_train )
```

```
## Warning in use.package("data.table"): data.table cannot be used without R
## package bit64 version 0.9.7 or higher. Please upgrade to take advangage of
## data.table speedups.
```

```
##   |                                                                      |
```

```
str(allData_hex)
```

```
## Class 'H2OFrame' <environment: 0x7fe5557e4888>
##  - attr(*, "op")= chr "Parse"
##  - attr(*, "id")= chr "data_train_sid_82b0_1"
##  - attr(*, "eval")= logi FALSE
```

```
##  - attr(*, "nrow")= int 2460
##  - attr(*, "ncol")= int 11
##  - attr(*, "types")=List of 11
##   ..$ : chr "int"
##   ..$ : chr "real"
##   ..$ : chr "real"
##   ..$ : chr "int"
##   ..$ : chr "int"
##   ..$ : chr "int"
##   ..$ : chr "int"
##   ..$ : chr "int"
##   ..$ : chr "real"
##   ..$ : chr "int"
##   ..$ : chr "int"
##  - attr(*, "data")='data.frame': 10 obs. of  11 variables:
##   ..$ Merchant_id                   : num  6.66e+09 4.03e+09 3.54e+09 6.52e+09 6.19e+09 ...
##   ..$ Average Amount/transaction/day: num  240 775 231 423 270 ...
##   ..$ Transaction_amount            : num  2164 4648 5080 4234 0 ...
##   ..$ Is declined                   : num  0 0 0 0 0 0 0 0 0 0
##   ..$ Total Number of declines/day  : num  0 0 0 0 0 6 3 1 0 0
##   ..$ isForeignTransaction          : num  0 0 0 0 0 0 1 0 0 0
##   ..$ isHighRiskCountry             : num  0 0 0 0 0 0 0 0 0 0
##   ..$ Daily_chargeback_avg_amt      : num  0 0 0 0 0 0 754 0 0 0
##   ..$ 6_month_avg_chbk_amt          : num  0 0 0 0 0 0 585 0 0 0
##   ..$ 6-month_chbk_freq             : num  0 0 0 0 0 0 7 0 0 0
##   ..$ isFradulent                   : num  0 0 0 0 0 0 1 0 0 0
```

Isolation Forest

Es un método no supervisado para identificar anomalías (outliers) cuando los datos no están etiquetados, es decir, no se conoce la clasificación real (anomalía - no anomalía) de las observaciones. • Su funcionamiento está inspirado en el algoritmo de clasificación y regresión Random Forest. • Un modelo Isolation Forest está formado por la combinación de múltiples árboles llamados isolation trees. • El modelo Isolation Forest se obtiene al combinar múltiples isolation tree, cada uno entrenado con una muestra distinta generada por bootstrapping a partir de los datos de originales.

Se aplica la técnica de detección de anomalías h2o.isolationforest

```
dataModel = h2o.isolationForest( training_frame = allData_hex,
                                 x              = colnames(allData_hex)[-11],
                                 sample_rate    = 0.9,
                                 max_depth      = 100,
                                 ntrees         = 500
                                )
```

```
##   |                                                                              |
```

```
dataModel
```

```
## Model Details:
## ==============
##
## H2OAnomalyDetectionModel: isolationforest
## Model ID:  IsolationForest_model_R_1593205015465_16
## Model Summary:
##   number_of_trees number_of_internal_trees model_size_in_bytes min_depth
```

```
## 1             500                       500            12398462        19
##   max_depth mean_depth min_leaves max_leaves mean_leaves
## 1        33   26.82200        337       2155  1965.95600
##
##
## H2OAnomalyDetectionMetrics: isolationforest
## ** Reported on training data. **
## ** Metrics reported on Out-Of-Bag training samples **
```

Se genera la predicción

```
allData_hex_test = as.h2o(data_test)
```

```
score = h2o.predict( dataModel, allData_hex_test )
```

```
##   |                                                                              |
```

```
result_pred = as.vector( score$predict )
```

```
predicciones_h2o <- h2o.predict(
                      object  = dataModel,
                      newdata = allData_hex
                    )
```

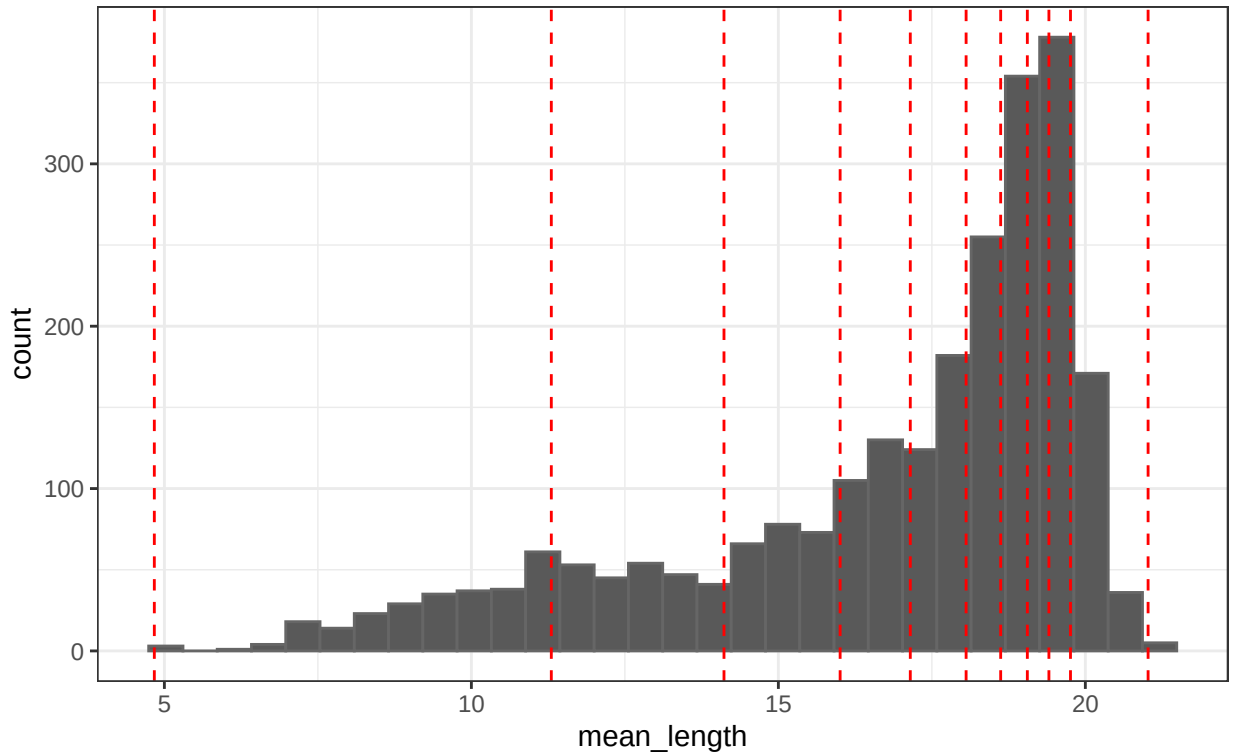
```
##   |                                                                              |
```

```
predicciones <- as.data.frame(predicciones_h2o)
head(predicciones)
```

```
##     predict mean_length
## 1 0.1846040      18.034
## 2 0.1324602      18.878
## 3 0.1029285      19.356
## 4 0.1644631      18.360
## 5 0.1698999      18.272
## 6 0.3386878      15.540
```

```
library(ggplot2)
ggplot(data = predicciones, aes(x = mean_length)) +
  geom_histogram(color = "gray40") +
  geom_vline(
    xintercept = quantile(predicciones$mean_length, seq(0, 1, 0.1)),
    color      = "red",
    linetype   = "dashed") +
  labs(
    title = "Distribución de las distancias medias del Isolation Forest",
    subtitle = "Cuantiles marcados en rojo"  ) +
  theme_bw()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
cuantiles <- quantile(x = predicciones$mean_length, probs = seq(0, 1, 0.05))
cuantiles
```

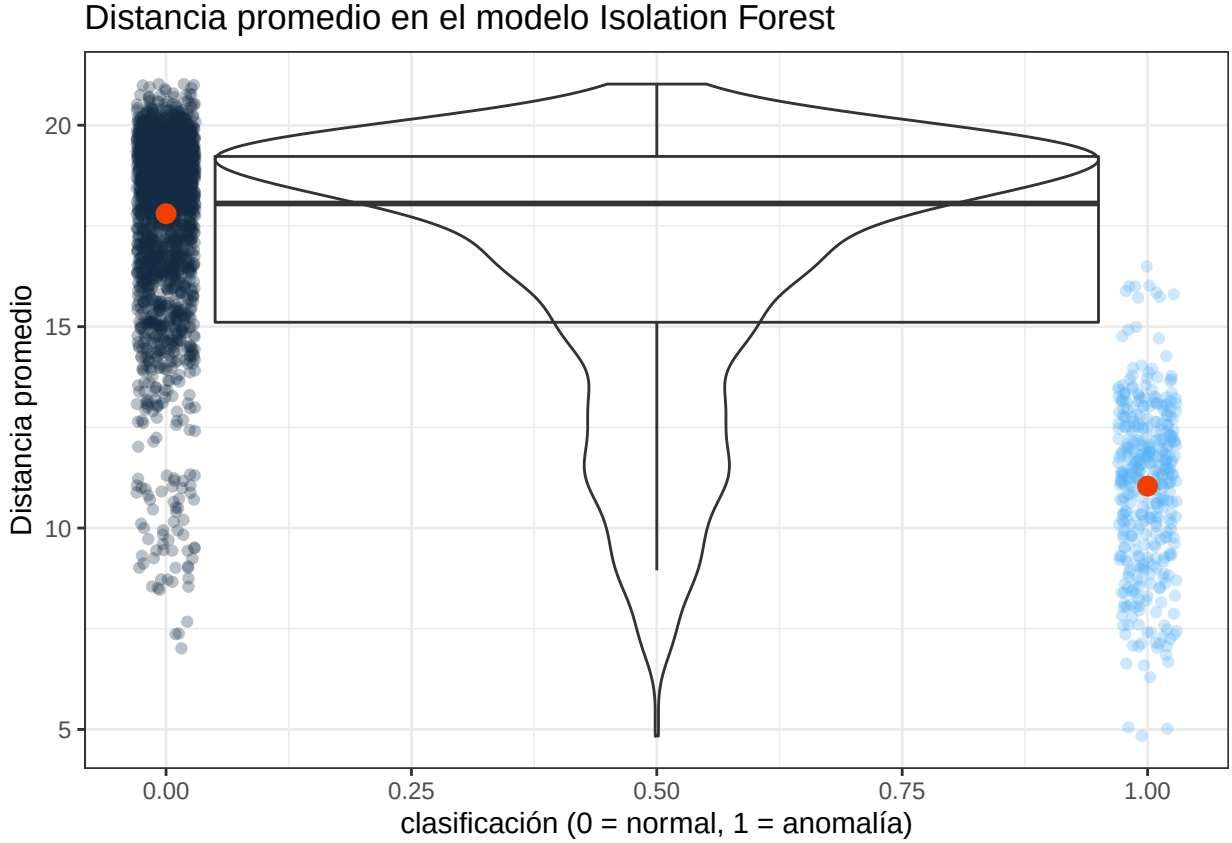
```
##      0%      5%     10%     15%     20%     25%     30%     35%     40%     45%
##  4.8360  9.7116 11.3020 12.6434 14.1144 15.1080 16.0060 16.6106 17.1488 17.7391
##     50%     55%     60%     65%     70%     75%     80%     85%     90%     95%
## 18.0580 18.3578 18.6208 18.8500 19.0552 19.2260 19.4064 19.5460 19.7582 20.0361
##    100%
## 21.0220
```

```
datos <- data_train %>%
         bind_cols(predicciones)
head(datos)
```

```
## # A tibble: 6 x 13
##   Merchant_id `Average Amount~ Transaction_amo~ `Is declined` `Total Number o~
##         <dbl>            <dbl>            <dbl>         <dbl>            <dbl>
## 1  6662015632             240.            2164.             0                0
## 2  4034539813             775.            4648.             0                0
## 3  3540872906             231.            5080.             0                0
## 4  6524453525             423.            4234.             0                0
## 5  6188615028             270.               0              0                0
## 6  3943474211             218.            3272.             0                6
## # ... with 8 more variables: isForeignTransaction <dbl>,
## #   isHighRiskCountry <dbl>, Daily_chargeback_avg_amt <dbl>,
## #   `6_month_avg_chbk_amt` <dbl>, `6-month_chbk_freq` <dbl>, isFradulent <dbl>,
## #   predict <dbl>, mean_length <dbl>
```

```
ggplot(data = datos,
       aes(x = isFradulent, y = mean_length)) +
geom_jitter(aes(color = isFradulent), width = 0.03, alpha = 0.3) +
geom_violin(alpha = 0) +
geom_boxplot(width = 0.2, outlier.shape = NA, alpha = 0) +
stat_summary(fun = "mean", colour = "orangered2", size = 3, geom = "point") +
labs(title = "Distancia promedio en el modelo Isolation Forest",
     x = "clasificación (0 = normal, 1 = anomalía)",
     y = "Distancia promedio") +
theme_bw() +
theme(legend.position = "none")
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```



Acorde a la información del dataset, contiene 380 anomalías.

Se muestra matriz de confusión resultante si se clasifican como anomalías, las 380 observaciones con menor distancia predicha.

```
resultados <- datos %>%
              select(isFradulent, mean_length) %>%
              arrange(mean_length) %>%
              mutate(clasificacion = if_else(row_number() <= 380, "1", "0"))
```

```
mat_confusion <- MLmetrics::ConfusionMatrix(
                    y_pred = resultados$clasificacion,
                    y_true = resultados$isFradulent
                 )
```

```
mat_confusion
```

```
##       y_pred
## y_true    0    1
##      0 2007   74
##      1   73  306
```

```
a = 310 / 380
falsos_positivos = 1 - a
falsos_positivos * 100
```

```
## [1] 18.42105
```

Técnica de agrupamiento K-means

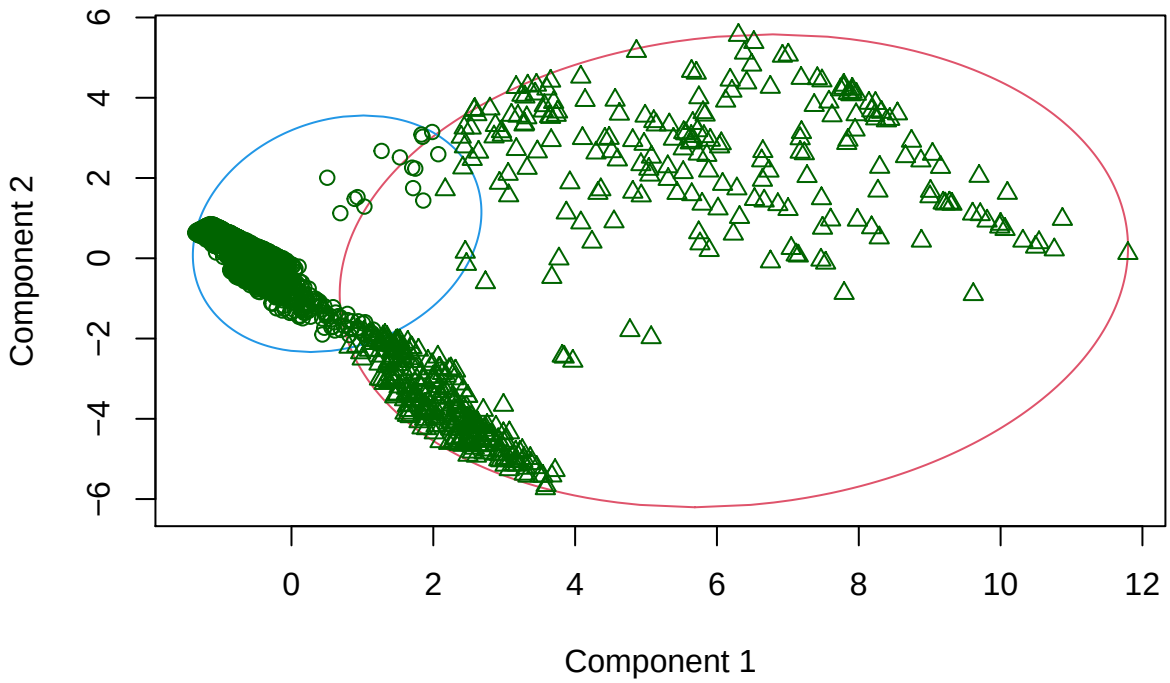
K-means es un algoritmo de clasificación no supervisada (clusterización) que agrupa objetos en k grupos basándose en sus características.

El agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo o cluster. Se suele usar la distancia cuadrática.

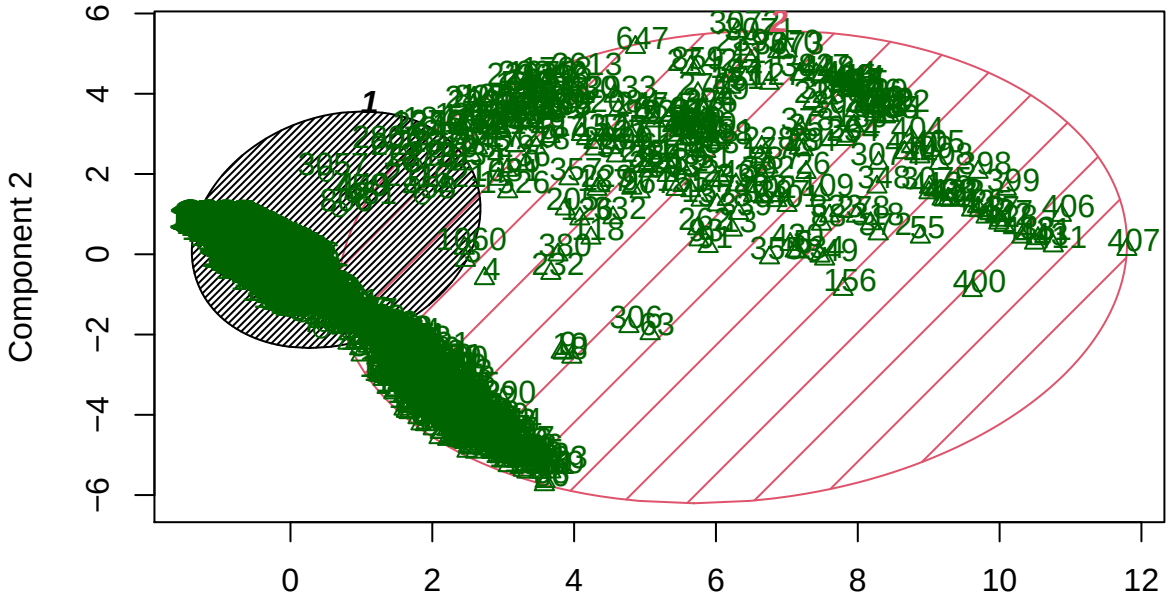
El algoritmo k-means resuelve un problema de optimización, siendo la función a optimizar (minimizar) la suma de las distancias cuadráticas de cada objeto al centroide de su cluster.

```
set.seed(1234)
data_2 <- as.data.frame(lapply(data, scale))
```

```
clusters <- kmeans(data_2,2)
clusplot(data_2,
         clusters$cluster,
         color = TRUE
         )
```




```
clusplot(data_2,
         clusters$cluster,
         color = TRUE,
         col.clus=c(1:2)[unique(clusters$cluster)],
         shade = TRUE,
         labels = 2,
         lines=0,
         main = "Bivariate Cluster Plot")
```




```
#library()
#plotcluster(data_2, clusters$cluster)
clusters$size
```

```
## [1] 2592  483
```

```
clusters$centers
```

```
##   Merchant_id Average.Amount.transaction.day Transaction_amount Is.declined
## 1 -0.01839243                    -0.02583824         -0.2196326  -0.1288269
## 2  0.09870223                     0.13865988          1.1786495   0.6913445
##   Total.Number.of.declines.day isForeignTransaction isHighRiskCountry
## 1                   -0.2139091           -0.1835336        -0.2672178
## 2                    1.1479348            0.9849254         1.4340134
##   Daily_chargeback_avg_amt X6_month_avg_chbk_amt X6.month_chbk_freq isFradulent
## 1               -0.2580927            -0.2494565         -0.2448457  -0.3800924
## 2                1.3850440             1.3386981          1.3139547   2.0397507
```

```
data_2$cluster <- clusters$cluster

data_2[1:10,c("cluster","isFradulent","isForeignTransaction","isHighRiskCountry"
              )]
```

```
##    cluster isFradulent isForeignTransaction isHighRiskCountry
## 1        2    2.421143            1.8315111         3.7410489
## 2        2    2.421143            1.8315111         3.7410489
## 3        2    2.421143           -0.5458197        -0.2672178
## 4        2    2.421143           -0.5458197        -0.2672178
## 5        2    2.421143            1.8315111         3.7410489
## 6        2    2.421143            1.8315111         3.7410489
```

```
## 7        2    2.421143          -0.5458197       -0.2672178
## 8        2    2.421143          -0.5458197       -0.2672178
## 9        2    2.421143          -0.5458197       -0.2672178
## 10       2    2.421143          -0.5458197       -0.2672178
```

```
aggregate(data = data_2,isFradulent ~ cluster,mean)
```

```
##   cluster isFradulent
## 1       1  -0.3800924
## 2       2   2.0397507
```

```
aggregate(data = data_2,isForeignTransaction ~ cluster,mean)
```

```
##   cluster isForeignTransaction
## 1       1           -0.1835336
## 2       2            0.9849254
```

```
aggregate(data = data_2,isHighRiskCountry ~ cluster,mean)
```

```
##   cluster isHighRiskCountry
## 1       1        -0.2672178
## 2       2         1.4340134
```

Conclusiones

- Se realizó el análisis de los datos del data frame cargando y revisando cada una de las variables observando que se tiene 12 variables con 3075 observaciones. En el análisis de los datos se observó que se cuenta con 1 variable lógica, 7 numéricas y 4 tipo char.
- Al realizar la correlación de las variables se encuentra alta correlación entre la varible isFradulen con Transaction_amount, Total Number of declines/day, isForeignTransaction, isHighRiskCountry, 6-month_chbk_freq, tambien de la variable Transaction_amount con Average Amount/transaction/day, is declined con Daily_chargeback_avg_amt , 6_month_avg_chbk_amt, 6-month_chbk_freq, de la variable isForeignTransaction con isHighRiskCountry, entre las más relevantes.
- La detección de anomalías para el conjunto de datos la predicción permite inferir que la información de la data cargada contiene 380 anomalías clasificadas de manera adecuada por el método.
- Se observó que para el ejemplo con K-means no queda tan claro el concepto, por lo que se sugeriría realizarla con otro método para ver si mejora. En cuanto a los clusters usados se notó que los datos se situaron a la periferia de los círculos que identifican el grupo, y otro tanto dispersos sobre el círculo mayor, lo que no se deja ver muy claro el concepto, como sí aparece en el ejemplo propuesto por el profesor en las magistrales.