# CASO: “USO/ACTIVACIÓN DE TARJETAS DE CRÉDITO”

# CONTEXTO

Antonio entró a trabajar al City Bank hace unos pocos meses como Gerente de la Banca Personas y está muy entusiasmado con entender el funcionamiento del negocio, con la intención de poder proponer mejoras que generen valor a los clientes y al banco.

A la fecha, el segmento de tarjetahabientes del City Bank está conformado por 700 mil personas y se cuenta con una participación de mercado de alrededor del 23%, ubicándose como líder en el segmento. Sin embargo se observa que las ventas nuevas han venido disminuyendo progresivamente, ubicándonos como el 4to banco en participación de mercado en venta de nuevas tarjetas.

Así, uno de los encargos de Antonio consiste en encontrar una forma de generar eficiencias en costos en el segmento de tarjetas de crédito (vender más y mejores tarjetas (que se usen y generen ingresos). Específicamente, se ha observado que aproximadamente un 40% de las tarjetas que el banco vende a los clientes nunca se utilizan.

Una tarjeta no utilizada genera una serie de costos, como los relacionados con la impresión, la venta y entrega de la misma. Este monto asciende aproximadamente a 160 soles por tarjeta. Dado que una tarjeta sin uso no genera ingresos para el banco, la rentabilidad del producto se ve afectada, más aun teniendo en consideración que el banco vende aproximadamente 200 000 tarjetas semestralmente. Por otro lado cada tarjeta que si se utiliza le genera al banco un ingreso neto (ya descontado los costos detallados anteriormente) diferenciado por el perfil del cliente.



El saber con anticipación qué tipo de clientes podría tener una mayor propensión a usar la tarjeta podría hacer que el equipo de Antonio decida a cuales clientes otorgarles una tarjeta y a cuáles no.

Adicionalmente, existe una posibilidad de incentivar el uso natural de las tarjetas a través de campañas específicas. El costo histórico promedio de una campaña para cada tarjeta asciende a 40 soles y la efectividad de la misma hace que la probabilidad de uso de la tarjeta se incremente en un 50%.

En vista de la situación actual, Antonio decidió contactarse con usted para evaluar qué opciones analíticas podría utilizar para solucionar su problemática actual. Específicamente, le encantaría contar al menos con una estimación aproximada de la probabilidad de uso de una tarjeta de crédito que le permita identificar con mayor precisión quienes serían los clientes que realizarán un primer consumo con su tarjeta de crédito. Asimismo, les ha provisto información histórica sobre las ventas reales de tarjetas de crédito de los últimos años.

Considere que usted es uno de los consultores que ha sido contactado por Antonio para colaborar en este proyecto y trabaje una propuesta analítica que sea capaz de cubrir las necesidades de este cliente. Además del desarrollo de una herramienta analítica para estimar la probabilidad de uso de una tarjeta de crédito (utilizando como máximo 50 variables explicativas. Para ello **deberá de utilizar la base “Base de Datos”** para mostrar sus resultados. Al menos Antonio espera que el modelo le ayude a formular una estrategia que maximice la utilidad indicándole a cuales clientes otorgar (o no) la tarjeta. Además, Antonio valorará otras propuestas de uso novedosas y diferentes a las que tiene en mente (piense en todos los usos potenciales para el banco).

Finalmente, considere que usted debe presentarle de manera efectiva sus resultados a Antonio al término del proyecto.

# INFORMACIÓN

La información disponible para desarrollar el caso es la siguiente:

* Base de datos para entrenar el modelo: El archivo “Base de Datos” contiene **7700** registros con información del identificador de la solicitud (CODSOLICITUD) de la tarjeta de crédito (aleatorio), una variable objetivo dicotómica (FLG\_ACTIVACION\_0\_12) de uso en los próximos 12 meses de la tarjeta de crédito (1 si usó y 0 si no usó) e información sobre 397 variables potencialmente explicativas. Sobre esta base se debe construir el modelo de uso de la tarjeta de crédito.
* Diccionario de variables: El archivo “Diccionario Variables.xlsx” contiene un detalle de referencia sobre los descriptivos de los campos de las tablas anteriores.
* Queda a potestad del candidato definir las bases train y test para validar el modelo.

# ENTREGABLE

El entregable es una presentación que contenga, al menos, los siguientes puntos:

* 1. Diseño metodológico de la estrategia analítica: Pasos seguidos en el desarrollo del modelo (depuración de variables, tratamiento de datos, si se realizaron varias pruebas, indicarlas y explicar la racional detrás de cada decisión).
  2. Explicación breve de la técnica estadística utilizada.
  3. Resultados del modelo en la muestra de desarrollo
  4. Propuestas de posibles usos del modelo.

DESARROLLO

1. Metodología Analítica.

El proceso analítico contemplas las siguientes etapas:

* + 1. Carga del dataset
    - Contamos con un set de datos de 7700 observaciones con 398 variables entre las cuales está “flg\_activacion\_0\_12” que será nuestra variable de objetivo (target).
    - Entre las variables categóricas están “cat\_zona1” y “tipo\_cliente” con 30 y 3 variables respectivamente
    1. Revisamos que variables tienen mayor presencia de NA (análisis por columnas)
    - Las variables con muchos NA (vacios) no aportan información por lo cual es mejor retirarlas
    1. Quitamos las observaciones que tienen presencia de NAs (Análisis por filas)
    2. Re-categorizamos las variables categóricas (variables dummy)
    - Es necesario “dummisar” las variables categóricas para ver que categoría influyen más en el target.
    1. Preprocesamiento que contempla, escalamiento, centrado, y retiramos las variables con varianza cercana a cero.
    - Dado que no todas las variables tienen la misma escala (unas son ratios, otros montos, etc) ponemos a todas en una misma escala (normalizamos)
    - Centramos los datos para que no estén sesgados
    - Las variables con varianza cercana a cero no aportan información para el modelo, por lo cual se optó por retirarlas.
    1. Retiramos variables correlacionadas
    - Las variables correlacionadas aportan la misma información por lo cual no es recomendable tenerlas en el set de datos para entrenar a los modelos.
    1. Seleccionamos variables importantes
    - La importancia de variables la realizamos con el algoritmo “randomforest” el cual nos indica que variables productoras están más asociadas al target.

La métrica usada es la reducción de la exactitud (Accuracy)

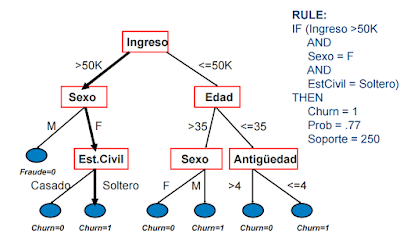
Nos quedamos con las primeras variables.

* + 1. Balanceamos el target
    2. Competencia de modelos con Validación cruzada
    - Se emplea validación cruzada para comparar los modelos, se crea 10 folds.
    1. Selección del mejor modelo

1. Árboles de clasificación y bosques aleatorios.

**Árboles de clasificación:** Los árboles de clasificación son un algoritmo de la clasificación supervisada, la idea surge de la estructura de un árbol que se compone de una raíz, nodos (las posiciones donde las ramas se dividen), ramas y hojas; de manera similar, un árbol de clasificación se construye a partir de nodos que representan los círculos y las ramas son representadas por los segmentos que conectan los nodos.

Un árbol de clasificación se inicia con un nodo raíz, en el cual está la variable que mayor asociación tienen con la variable objetivo, y las demás variables en orden de importancia van apareciendo en los demás nodos, cada nodo intenta discriminar las clases de la categoría objetivo. (Ali, Khan, Ahmad y Maqsood, 2012).

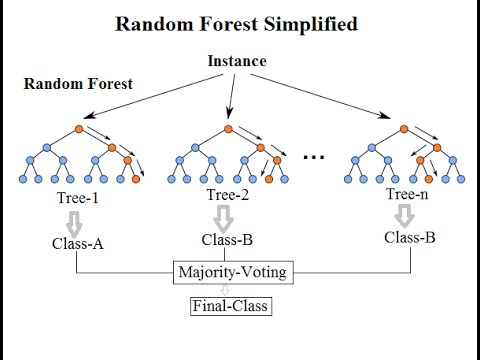


**Bosques aleatorios:** También conocidos en castellano como '"Bosques Aleatorios"' es una combinación de árboles de clasificación tal que cada árbol depende de las variables predictores que son una submuestra del total de variables. Es una modificación sustancial de bagging que construye una larga colección de árboles no correlacionados y luego los promedia.

El algoritmo para inducir un random forest fue desarrollado por Leo Breiman1​ y Adele Cutler y Random forests es su marca de fábrica. El término aparece de la primera propuesta de Random decision forests, hecha por Tin Kam Ho de Bell Labs en 1995. El método combina la idea de bagging de Breiman y la selección aleatoria de atributos, introducida independientemente por Ho,2​3​ Amit y Geman,4​ para construir una colección de árboles de decisión con variación controlada.

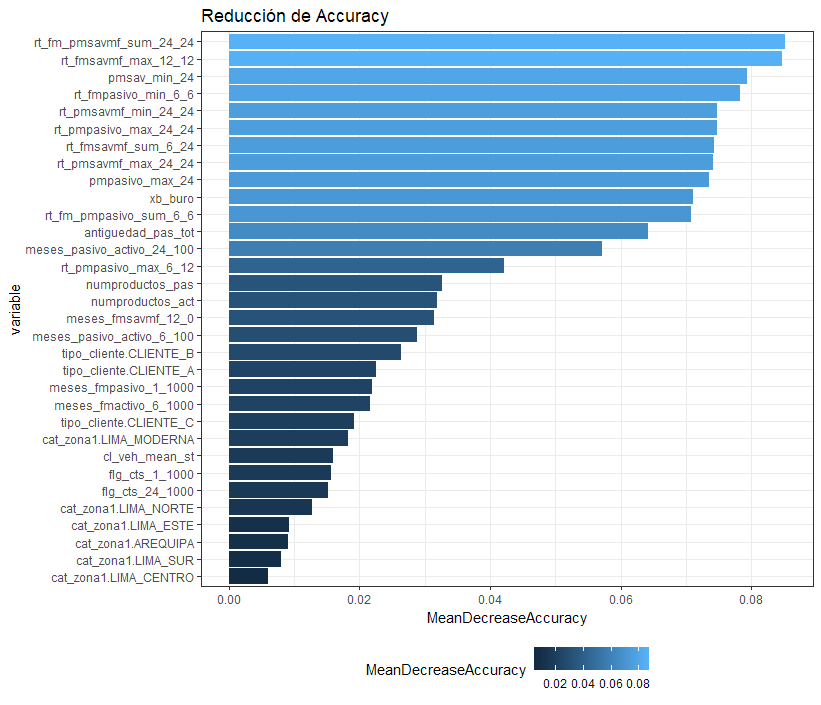
La selección de un subconjunto aleatorio de atributos es un ejemplo del método random subspace, el que, según la formulación de Ho, es una manera de llevar a cabo la discriminación estocástica5​ propuesta por Eugenio Kleinberg.

En muchos problemas el rendimiento del algoritmo random forest es muy similar a la del boosting, y es más simple de entrenar y ajustar. Como consecuencia, el Random forest es popular y ampliamente utilizado.



1. Resultado del Train.

En el proceso de selección de variables nos quedamos con 14 predictores que son los primeros del siguiente gráfico.



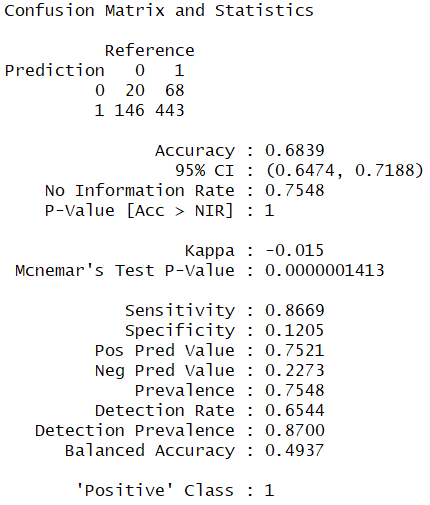
La validación cruzada con 12 grupos nos da los siguientes resultados.

El script de desarrollo entre los modelos está en el anexo.

Presentamos el performance de la validación cruzada en la cual podemos notar que el modelo que más se ajusta a nuestros datos, es el “random forest” con una exactitud cercana al 70%.



Además de la exactitud también es necesario ver como se comparta la matriz de confusión, por lo cual adjuntamos una salida de la matriz de confusión de este modelo.



Observamos que el modelo es bueno prediciendo aquellos clientes que usarán su crédito, es decir podemos predecir los positivos (pos\_pred\_value 75%).

1. Propuestas de uso.

Con la salida de la matriz de confusión observamos que el modelo puede ayudarnos a saber que cliente usará o no el crédito otorgado, además podemos crear un rankinf según la probabilidad de usa que será nuestra probabilidad de uso del crédito. Con este score se puede priorizar a los clientes a la hora de otorgar los créditos.

Es recomendable desarrollar una segmentación previa al modelo predictivo, con esto se consigue que la exactitud se más precisa.

Anexo

rm(list = ls())

# Análisis

library(tidyverse)

library(caret)

library(rpart)

library(randomForest)

library(nnet)

library(ROSE)

# Carga de datos

datos <- read.csv("Base de datos.csv")

dim(datos)

datos <- datos %>% select(-codsolicitud, -rank\_tip)

# Variables con poca información

pMiss <- function(x){sum(is.na(x))/length(x)\*100}

aux <- as.data.frame(apply(datos,2,pMiss))

aux$variable <- row.names(aux)

names(aux) <- c("porc\_perdidos", "variables")

# Nos quedamos con las variables que tienen menos de cantidad de datos perdidos

# punto de corte será 0.10

plot(density(aux$porc\_perdidos))

table(ifelse(aux$porc\_perdidos>15,0,1))

quantile(aux$porc\_perdidos, seq(0.4,1,0.05))

variables <- aux %>% filter(porc\_perdidos<15) %>% select(variables)

datos <- datos[, variables$variables]

# Quitamos las observaciones con NAs

datos <- na.omit(datos)

# Recategorizamos las variables categoricas en dummys

datos$flg\_activacion\_0\_12 <- as.factor(datos$flg\_activacion\_0\_12)

dummies <- dummyVars(flg\_activacion\_0\_12 ~ ., data = datos)

head(predict(dummies, newdata = datos))

datos\_x <- data.frame(predict(dummies, newdata = datos))

flg\_activacion\_0\_12 <- datos$flg\_activacion\_0\_12

datos <- cbind(datos\_x, flg\_activacion\_0\_12)

# Pre procesamientp

pp\_no\_nzv <- preProcess(datos %>% select(-flg\_activacion\_0\_12),

method = c("center", "scale", "YeoJohnson", "nzv"))

datos\_final <- predict(pp\_no\_nzv, newdata = datos)

hist(datos\_final$flg\_cts\_24\_1000)

# QUitamos las variables correlacionadas

descrCor <- cor(datos\_final %>% select(-flg\_activacion\_0\_12))

highlyCorDescr <- findCorrelation(descrCor, cutoff = .75)

datos\_final2 <- datos\_final[,-highlyCorDescr]

descrCor2 <- cor(datos\_final2 %>% select(-flg\_activacion\_0\_12))

summary(descrCor2[upper.tri(descrCor2)])

# Selección de variables importantes

library(randomForest)

set.seed(666)

datos\_rf <- ovun.sample(flg\_activacion\_0\_12 ~ ., data = datos\_final2, method = "over")$data

datos\_rf <- map\_if(.x = datos\_rf, .p = is.character, .f = as.factor) %>%

as.data.frame()

modelo\_randforest <- randomForest(formula = flg\_activacion\_0\_12 ~ . ,

data = na.omit(datos\_rf),

mtry = 3,

importance = TRUE,

ntree = 500)

modelo\_randforest

importancia <- as.data.frame(modelo\_randforest$importance)

importancia <- rownames\_to\_column(importancia,var = "variable")

p1 <- ggplot(data = importancia, aes(x = reorder(variable, MeanDecreaseAccuracy),

y = MeanDecreaseAccuracy,

fill = MeanDecreaseAccuracy)) +

labs(x = "variable", title = "Reducción de Accuracy") +

geom\_col() +

coord\_flip() +

theme\_bw() +

theme(legend.position = "bottom")

p2 <- ggplot(data = importancia, aes(x = reorder(variable, MeanDecreaseGini),

y = MeanDecreaseGini,

fill = MeanDecreaseGini)) +

labs(x = "variable", title = "Reducción de pureza (Gini)") +

geom\_col() +

coord\_flip() +

theme\_bw() +

theme(legend.position = "bottom")

library(ggpubr)

ggarrange(p1, p2)

p1

importancia <- importancia %>% arrange(-MeanDecreaseAccuracy)

variables <- importancia$variable[1:12]

datos\_final3 <- datos\_final2[, c(variables, "flg\_activacion\_0\_12")]

# borrar <- ls()

# borrar <- borrar[-which(borrar == "datos\_final3")]

# rm(list = borrar)

# rm(borrar)

# VALIDACI?N CRUZADA

# Creamos los grupos(K-Folds), separa a la data en 10 grupos diferentes

set.seed(1) # Prefijamos una semilla para tener resultados reproducibles

folds <- 10 # 10 Grupos

datos\_final3$kfold <- sample(1:folds, nrow(datos\_final3), replace = T)

ind\_accuracy <- data.frame(NULL)

ind\_kappa <- data.frame(NULL)

for (i in 1:folds){

#i=1

test <- subset(datos\_final3, kfold == i)

train <- subset(datos\_final3, !kfold == i)

train <- ovun.sample(flg\_activacion\_0\_12 ~ ., data = train, method = "over")$data

# REGRESION LOGISTICA

logistica <- glm(flg\_activacion\_0\_12 ~ .,data = train %>% select(-kfold),family=binomial(link="logit"))

logist\_pred <- predict(logistica, test, type = "response") # Predicci?n sobre test

logist\_pred\_clase <- factor(ifelse(logist\_pred>0.5,1,0))

# Matriz de confusi?n

table(logist\_pred\_clase,test$flg\_activacion\_0\_12)

confusionMatrix(logist\_pred\_clase,test$flg\_activacion\_0\_12,positive="1")

kap\_logist <- confusionMatrix(logist\_pred\_clase,test$flg\_activacion\_0\_12,positive="1")$overall[2]

kap\_logist

auc\_logist <- confusionMatrix(logist\_pred\_clase,test$flg\_activacion\_0\_12,positive="1")$overall[1]

# BOSQUE ALEATORIO

bosque <- randomForest(flg\_activacion\_0\_12 ~ ., data=train %>% select(-kfold), ntree=10, mtry= 5)

bosque\_pred <- predict(bosque, test, type = "response") # Predicci?n sobre test

# Matriz de confusi?n

table(bosque\_pred,test$flg\_activacion\_0\_12)

confusionMatrix(bosque\_pred,test$flg\_activacion\_0\_12,positive="1")

kap\_BA <- confusionMatrix(bosque\_pred,test$flg\_activacion\_0\_12,positive="1")$overall[2]

kap\_BA

auc\_BA <- confusionMatrix(bosque\_pred,test$flg\_activacion\_0\_12,positive="1")$overall[1]

# ARBOL DE CLASIFICACION

arbol <- rpart(flg\_activacion\_0\_12 ~ ., data=train %>% select(-kfold), method = "class")

arb\_pred <- predict(arbol,test, type = "class")

# Matriz de confusi?n

table(arb\_pred,test$flg\_activacion\_0\_12)

confusionMatrix(arb\_pred,test$flg\_activacion\_0\_12,positive="1")

kap\_AC <- confusionMatrix(arb\_pred,test$flg\_activacion\_0\_12,positive="1")$overall[2]

kap\_AC

auc\_AC <- confusionMatrix(arb\_pred,test$flg\_activacion\_0\_12,positive="1")$overall[1]

# REDES NEURONALES

red\_neu <- nnet(flg\_activacion\_0\_12~., data = train %>% select(-kfold),size = 4, rang = 0.1,decay = 5e-4, maxit = 1000)

red\_pred <- predict(red\_neu,test)

red\_pred\_clase <- factor(ifelse(red\_pred>0.5,1,0))

# Matriz de confusi?n

table(red\_pred\_clase,test$flg\_activacion\_0\_12)

confusionMatrix(red\_pred\_clase,test$flg\_activacion\_0\_12,positive="1")

kap\_RN <- confusionMatrix(red\_pred\_clase,test$flg\_activacion\_0\_12,positive="1")$overall[2]

kap\_RN

auc\_RN <- confusionMatrix(red\_pred\_clase,test$flg\_activacion\_0\_12,positive="1")$overall[1]

ind\_kappa <- rbind(ind\_kappa, data.frame(iteracion = i, Logistica = kap\_logist, BosqueAleatorio = kap\_BA, ArbolClasificacion = kap\_AC, RedNeuronal = kap\_RN))

ind\_accuracy <- rbind(ind\_accuracy, data.frame(iteracion = i, Logistica = auc\_logist, BosqueAleatorio = auc\_BA,ArbolClasificacion = auc\_AC, RedNeuronal = auc\_RN))

print(paste("Fin de Iteraci?n",i))

}

ind\_kappa <- round(ind\_kappa,2)

row.names(ind\_kappa) <- NULL

ind\_kappa

ind\_accuracy <- round(ind\_accuracy,2)

row.names(ind\_accuracy) <- NULL

# Medida de precisi?n Sensibilidad

kappa\_final <- apply(ind\_kappa[,-1],2,mean)

{windows()

plot(ind\_kappa[,c(1,2)], type = "l", ylim = c(-0.1,0.1), main = "Evoluci?n del indice kappa para todos los modelos",

cex.axis = .7,cex.lab = .7,cex.main = .8,

xlab ="No. de Iteraciones", ylab="Sensibilidad")

lines(ind\_kappa$BosqueAleatorio, col="green",pch=22)

lines(ind\_kappa$ArbolClasificacion, col="red")

lines(ind\_kappa$RedNeuronal, col="blue")

abline(h = kappa\_final[1], col = "black", lty = 2)

abline(h = kappa\_final[2], col = "green", lty = 2)

abline(h = kappa\_final[3], col = "red", lty = 2)

abline(h = kappa\_final[4], col = "blue", lty = 2)

legend("bottomright", ncol=4,c("Logistica","BosqueAleatorio","Arbol de clasificacion", "Red Neuronal"),

cex=0.6,bty="n",fill=c("black","green","red","blue"))

}

# Medida de precisi?n Gini

accuracy\_final <- apply(ind\_accuracy[,-1],2,mean)

{windows()

plot(ind\_accuracy[,c(1,2)], type = "l", ylim = c(0,1), main = "Evoluci?n del accuracy para todos los modelos",

cex.axis = .7,cex.lab = .7,cex.main = .8,

xlab ="No. de Iteraciones", ylab="Indicador Gini")

lines(ind\_accuracy$BosqueAleatorio, col="green",pch=22)

lines(ind\_accuracy$ArbolClasificacion, col="red")

lines(ind\_accuracy$RedNeuronal, col="blue")

abline(h = accuracy\_final[1], col = "black", lty = 2)

abline(h = accuracy\_final[2], col = "green", lty = 2)

abline(h = accuracy\_final[3], col = "red", lty = 2)

abline(h = accuracy\_final[4], col = "blue", lty = 2)

legend("bottomright", ncol=4,c("Logistica","BosqueAleatorio","Arbol de clasificacion", "Red Neuronal"),

cex=0.6,bty="n",fill=c("black","green","red","blue"))}

**CONSULTAS**

- En el punto 2. y 3. donde se retiran las columnas y filas que presentan muchos NAs, cuales son las que quedan?

Las variables que quedan son 54 y se enumeran a continuación:

|  |
| --- |
| xb\_buro |
| cl\_veh\_mean\_st |
| cat\_zona1 |
| numproductos\_pas |
| numproductos\_act |
| numproductos |
| meses\_fmactivo\_24\_0 |
| meses\_fmactivo\_6\_1000 |
| meses\_fmpasivo\_6\_0 |
| meses\_fmpasivo\_12\_100 |
| meses\_fmpasivo\_6\_100 |
| meses\_fmpasivo\_24\_1000 |
| meses\_fmpasivo\_12\_1000 |
| meses\_fmpasivo\_1\_1000 |
| meses\_fmsavmf\_24\_0 |
| meses\_fmsavmf\_12\_0 |
| meses\_pmsavmf\_12\_0 |
| fmjud\_count\_24 |
| pmsav\_min\_24 |
| pmsav\_med\_24 |
| pmpasivo\_max\_24 |
| pmpasivo\_min\_24 |
| pmpasivo\_med\_24 |
| fmvj\_count\_12 |
| pmpasivo\_min\_12 |
| fmpasivo\_min\_6 |
| fmsav\_min\_6 |
| pmpasivo\_min\_6 |
| pmpasivo\_med\_6 |
| meses\_pasivo\_activo\_6\_100 |
| meses\_pasivo\_activo\_12\_1000 |
| meses\_pasivo\_activo\_24\_100 |
| meses\_pasivo\_activo\_24\_1000 |
| antiguedad\_pas\_tot |
| maxantiguedad\_act\_pas\_vig |
| rt\_pmpasivo\_max\_6\_12 |
| rt\_fmsavmf\_sum\_6\_24 |
| rt\_pmsav\_sum\_1\_1 |
| flg\_cts\_1\_1000 |
| rt\_fmpasivo\_max\_6\_6 |
| rt\_fmpasivo\_min\_6\_6 |
| rt\_fm\_pmpasivo\_sum\_6\_6 |
| flg\_cts\_6\_1000 |
| rt\_pmpasivo\_max\_12\_12 |
| rt\_fmpasivo\_min\_12\_12 |
| rt\_fmsavmf\_max\_12\_12 |
| rt\_pmpasivo\_max\_24\_24 |
| rt\_pmsavmf\_max\_24\_24 |
| rt\_pmsavmf\_min\_24\_24 |
| rt\_fm\_pmsavmf\_sum\_24\_24 |
| rt\_fmsav\_sum\_24\_24 |
| flg\_cts\_24\_1000 |
| tipo\_cliente |
| flg\_activacion\_0\_12 |

- En el punto 4. donde sale "Recategorizamos las variables categóricas (variables dummy) ¿En qué consiste ese proceso? y cómo se sabe cuales son las que influyen mas?

\* Al tener una variable que tiene categorías, como lo es cat\_zona1 que tiene 30 niveles distintos, no todas las categorías se comportan igual, más aún existen categorías en las que por tener cierta categoría ya existe un sesgo en la variable objetivo. Pongámoslo así, supongamos que solo existen tres categorías de zona que sean Lima, Callao y provincias. Puede ser que todas las personas de Lima hacen compras inmediatas con su tarjeta por lo cual esa es una categoría de influencia positiva, mientras que en callao la mayoría no hace compras inmediatas, es decir, tienen influencia negativa y provincias se comportan mitad y mitad. Entonces sabes que Lima está a favor y callao en contra de tu variable objetivo.

- En el punto 5. donde sale que se ponen las variables a una misma escala ¿A qué escala se ponen? y ¿A qué variables se le cambia la escala, a todas? También donde sale que se centran los datos para que no esten sesgados ¿En qué consiste ese proceso de centrar?

\* Primero escalar, hace referencia al hecho que las variables no tienen la misma naturaleza. Pongamos un ejemplo supongamos que dos variables importantes son el número de hijos de una persona y su sueldo, mientras que la primera ira de 0, 1, 2 o hasta cinco exagerando el sueldo ira desde 500 hasta 30000 soles es decir no están medidas en la misma escala y si entran así al modelo la que tiene mayor variabilidad pesará más. Aquí el proceso es que todas tengan la misma variabilidad.

\* Respecto al centrado, lo que se busca es que las variables traten de seguir una distribución normal, done la mayoría de información está en el centro.

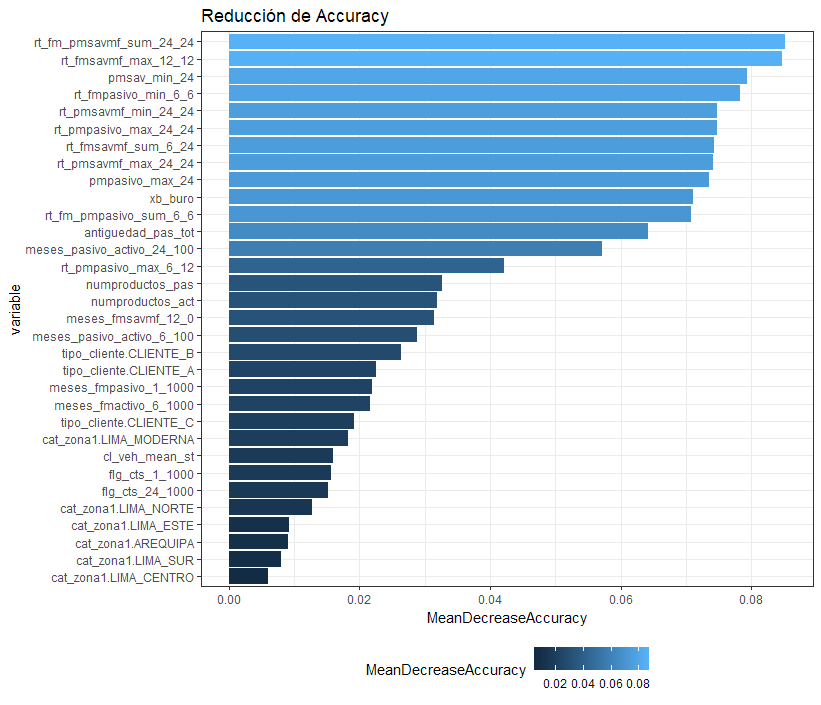
Juntando las dos (escalado y centrado) tratamos de que los datos sigan una distribución normal estándar (con media 1 y varianza 0).

- En el punto 7. ¿Qué son las variable productoras? ¿Por qué se utiliza el random forest y no otro? La métrica Accuracy utilizada siempre se usa con Random Forest? ¿Cuáles son las primeras variables que se elijen?

\* En este sentido la explicación es un poco más extensa. Las variables predictoras son aquellas que no ayudan a entender el comportamiento de la variable objetivo, es decir, son todas menos la que queremos predecir.

\* Sobre la utilización del randon forest, sucede que ese algoritmo trata de utilizar las variables (predictoras) que ayuden a entender más a la variable objetivo, es decir dentro de todas va probando una a una quien de todas las predictoras está más asociada a la variable objetivo y así construye una importancia de variables. No siempre se usa el randomforest, se puede usar otros como algoritmos de regresión penalizados, algoritmo por filtrado, algoritmos genéticos, boosting, etc. Usamos el random por simpleza y fácil entendimiento. Sobre la métrica, accuracy, hace referencia a la exactitud del modelo al predecir, es decir en cuántos casos acierta.

\* Las primeras variables o las más importantes a la hora de predecir si un cliente usará o no su tarjeta son las del gráfico.



- En el punto 8. ¿Cómo se balancea el target y cual sería el resultado?

Lo que se hizo para balancearlos es duplicar los casos de menor porcentaje, es decir, si los NO eran el 80% y los SI eran solo el 20% lo que se hace es duplicar los SI hasta que igualen en número a los NO. Eso es Oversampling, no es el único método existen otro como Undersampling (bajar la categoría mayor) o SMOOTE con data sintética equiparar las cantidades.

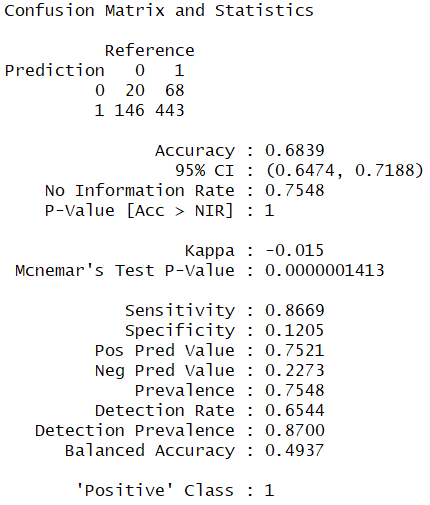
- ¿Por qué se uso esos 4 modelos?

El Modelar es un arte, y no hay un único camino, por eso siempre se trata de probar la mayor cantidad de modelos posibles, en este caso solo probamos 4, podríamos hacer más de 271 modelos, pero como solo se quiere entender el proceso analítico se puso 4. Piden uno solo así que se supera lo pedido.

- Y al final si veo que sale que el modelo es bueno prediciendo, pero quería saber como eso responde la pregunta que sale en el texto que dice:

"Específicamente, le encantaría contar al menos con una estimación aproximada de la probabilidad de uso de una tarjeta de crédito que le permita identificar con mayor precisión quienes serían los clientes que realizarán un primer consumo con su tarjeta de crédito". Si bien si responde a la primera pregunta sobre indicar la probabilidad de uso, la otra sería saber que clientes son los que realizarían el consumo. "Al menos Antonio espera que el modelo le ayude a formular una estrategia que maximice la utilidad indicándole a cuales clientes otorgar (o no) la tarjeta". También otra pregunta sería saber si se usarían las campañas mencionadas en el texto para aumentar la probabilidad de uso y que decir a que clientes se les daría una tarjeta.

Para esta parte usamos la matriz de confusión.



De lo marcado podemos ver que el modelo es bueno prediciendo quienes usarán la tarjeta, es por eso que el modelo nos dirá la probabilidad de uso de la tarjeta (ese es el score) mientras más alta es más probable que el cliente use la tarjeta de crédito.

Sobre la estrategia, la idea es entregar la tarjeta a aquellos clientes que el modelo bote una probabilidad mayor a 0.5. O si se quiere mayor precisión en el uso de las tarjetas de crédito podríamos decir que entreguemos las tarjetas a aquellos que tienen probabilidad mayor a 0.7 o mayor.

Sobre estrategias para incrementar podemos usar las del texto presentado y acompañar con experiencia que tenga el alumno en el rubro.