Universidad de Buenos Aires

Especialización en Explotación de Datos y Descubrimiento del Conocimiento.

Trabajo Práctico Final de la Especialización en Explotación de Datos y Descubrimiento del Conocimiento.

Análisis de bajas/permanencias de líneas de telefonía móvil

Alumno: Dawoon Choi

Supervisión: Marcelo Soria

Fecha: Marzo 2016

Contenido

Introducción

El presente trabajo tiene como objetivo principal aplicar técnicas de minería de datos sobre una base de datos que almacena información relacionada al uso de líneas móviles (dadas de alta en el mismo rango de periodo) y el estado de activación (baja / permanencia) luego de un tiempo determinado, de una operadora de telefonía móvil.

Los usuarios de servicios hoy en día, ya sea de comunicación u otra, pretenden contratar y usar servicios de alta calidad y si esta necesidad no es cumplida cambian de proveedor fácilmente en busca de una mejor prestación. Lo cual hace absolutamente crucial y necesario poner en marcha una estrategia sostenible y robusta de retención de clientes para preservar el valor económico que representa cada cliente.

Para ello, es fundamental estudiar, analizar y conocer cada uno de los datos relacionados al cliente ya sea información demográfica como información de uso de línea móvil en busca de conocimientos y patrones relacionados a la decisión de no seguir como cliente de la operadora móvil que afecta negativamente los ingresos del negocio en cuestión.

En este contexto, usaremos distintas técnicas de minería de datos:

- Clasificación (Análisis discriminante y RandomForest) que nos permita predecir quienes se darán de baja o no.
- Clustering (K-means y PAM) que nos permita caracterizar distintos grupos de comportamiento.

La base de datos (archivo de formato .csv) que utilizaremos para el desarrollo del presente trabajo práctico fue extraída del sitio web BigML (https://bigml.com/) [1] y corresponde a datos de usuarios de una operadora de telefonía móvil de EE.UU, donde se observan:

- 3333 filas que corresponden a la información histórica de los usuarios.
- 20 columnas entre datos demográficos y de uso de línea móvil.

Los softwares utilizados para el desarrollo del trabajo práctico son:

- SPSS Statistics versión 22.0.0.0.
- R versión 3.2.0

[1] La base puede ser descargada desde esta url: https://bigml.com/user/bigml/gallery/dataset/4f89bff4155268645c000030

Estadística descriptiva

A continuación se describen y se analizan de forma univariada las variables que conforman la base de datos. El dataset contiene 3333 casos y 20 variables:

1. STATE :Estado en donde vive el usuario (Categórica)
2. ACCOUNT: Identificador único por cliente (Categórica)
3. AREACODE: Código de Área (Categórica)
4. INT_PLAN: Plan a llamadas internacionales, SI/NO (Binaria)
5. VMAIL_PLAN: Plan a voice mail SI/NO (Binaria)
6. NMB_VMAIL_PLAN: Cantidad de voice mail (Numérica)
7. TOT_DAY_MIN: Minutos usados durante la mañana (Numérica)
8. TOT_DAY_CALLS: Cantidad de llamadas realizadas durante la mañana (Numérica)
9. TOT_DAY_CHARGE: Crédito gastado en llamadas realizadas durante la mañana (Numérica)
10. TOT_EVE_MIN: Minutos usados durante la tarde (Numérica)
11. TOT_EVE_CALLS: Cantidad de llamadas realizadas durante la tarde(Numérica)
12. TOT_EVE_CHARGE: Crédito gastado en llamadas realizadas durante el día (Numérica)
13. TOT_NIGHT_MIN: Minutos usados durante la noche(Numérica)
14. TOT_NIGHT_CALLS: Cantidad de llamadas realizadas durante la noche(Numérica)
15. TOT_NIGHT_CHARGE: Crédito gastado en llamadas realizadas durante la noche(Numérica)
16. TOT_INT_MIN: Minutos usados en llamadas internacionales(Numérica)
17. TOT_INT_CALLS : Cantidad de llamadas internacionales(Numérica)
18. TOTAL_INT_CHARGE: Crédito gastado en llamadas internacionales(Numérica)
19. CUST_SERV_CALLS : Cantidad de llamadas a centro de atención a clientes (Numérica)
20. CHURN: Baja=1 /Permanece= 0, es la variable que deseamos predecir (Binaria).

Estadística descriptiva de las variables mencionadas (con la excepción del atributo “Account” ya que es un identificador único):

Estadísticos descriptivos

	N	Mínimo	Máximo	Media	Mediana	Desviación	Varianza
TOTAL_DAY_MIN	3333	,00	350,80	179,775	179,4	54,467	2966,696
NMB_VMAIL_PLAN	3333	0	51	8,10	0	13,688	187,371
TOTAL_DAY_CALLS	3333	0	165	100,44	101	20,069	402,768
TOTAL_DAY_CHARGE	3333	,00	59,64	30,562	30,5	9,259	85,737
TOTAL_EVE_MIN	3333	,00	363,70	200,980	201,4	50,713	2571,894
TOTAL_EVE_CALLS	3333	0	170	100,11	100	19,923	396,911
TOTAL_EVE_CHARGE	3333	,00	30,91	17,083	17,12	4,310	18,582
TOTAL_NIGHT_MIN	3333	23,20	395,00	200,872	201,2	50,573	2557,714
TOTAL_NIGHT_CALLS	3333	33	175	100,11	100	19,569	382,930
TOTAL_NIGHT_CHARGE	3333	1,04	17,77	9,039	9,05	2,275	5,180
TOTAL_INT_MIN	3333	,00	20,00	10,237	10,3	2,791	7,794
TOTAL_INT_CALLS	3333	0	20	4,48	4	2,461	6,058
TOTAL_INT_CHARGE	3333	,00	5,40	2,7646	2,78	,75377	,568
CUST_SERV_CALLS	3333	0	9	1,56	1	1,315	1,731

No se observan valores faltantes para las variables de interés.

Tabla de frecuencia[2]:

INT_PLAN	Frecuencia	Porcentaje
no	3010	90,3
yes	323	9,7
Total	3333	100,0

AREACODE	Frecuencia	Porcentaje
408	838	25,1
415	1655	49,7
510	840	25,2
Total	3333	100,0

VMAIL_PLAN	Frecuencia	Porcentaje
no	2411	72,3
yes	922	27,7
Total	3333	100,0

CHURN	Frecuencia	Porcentaje
0	2850	85,5
1	483	14,5
Total	3333	100,0

ESTADO	Frecuencia	Porcentaje
AK	52	1,6
AL	80	2,4
AR	55	1,7
AZ	64	1,9
CA	34	1,0
CO	66	2,0
CT	74	2,2
DC	54	1,6
DE	61	1,8
FL	63	1,9
GA	54	1,6
HI	53	1,6
IA	44	1,3
ID	73	2,2
IL	58	1,7
IN	71	2,1
KS	70	2,1
KY	59	1,8
LA	51	1,5
MA	65	2,0
MD	70	2,1
ME	62	1,9

[2] Se adjunta en la seccion anexo los gráficos de barras.

MI	73	2,2
MN	84	2,5
MO	63	1,9
MS	65	2,0
MT	68	2,0
NC	68	2,0
ND	62	1,9
NE	61	1,8
NH	56	1,7
NJ	68	2,0
NM	62	1,9
NV	66	2,0
NY	83	2,5
OH	78	2,3
OK	61	1,8
OR	78	2,3
PA	45	1,4
RI	65	2,0
SC	60	1,8
SD	60	1,8
TN	53	1,6
TX	72	2,2
UT	72	2,2
VA	77	2,3
VT	73	2,2
WA	66	2,0
WI	78	2,3
WV	106	3,2
WY	77	2,3
Total	3333	100,0

Como se puede observar en la tabla anterior (ESTADO), la frecuencia de cada estado es bastante similar para la mayoría de los casos con excepciones en “WV” (Virginia Occidental) con 106 apariciones y en “CA” (California) con 34 observaciones.

La variable AREACODE presenta únicamente 3 valores que corresponden a las siguientes ciudades[3]:

[3] Código de área en Estados Unidos : http://www.mejortarjetatelefonica.com/usa/codigos-area-estados-unidos.

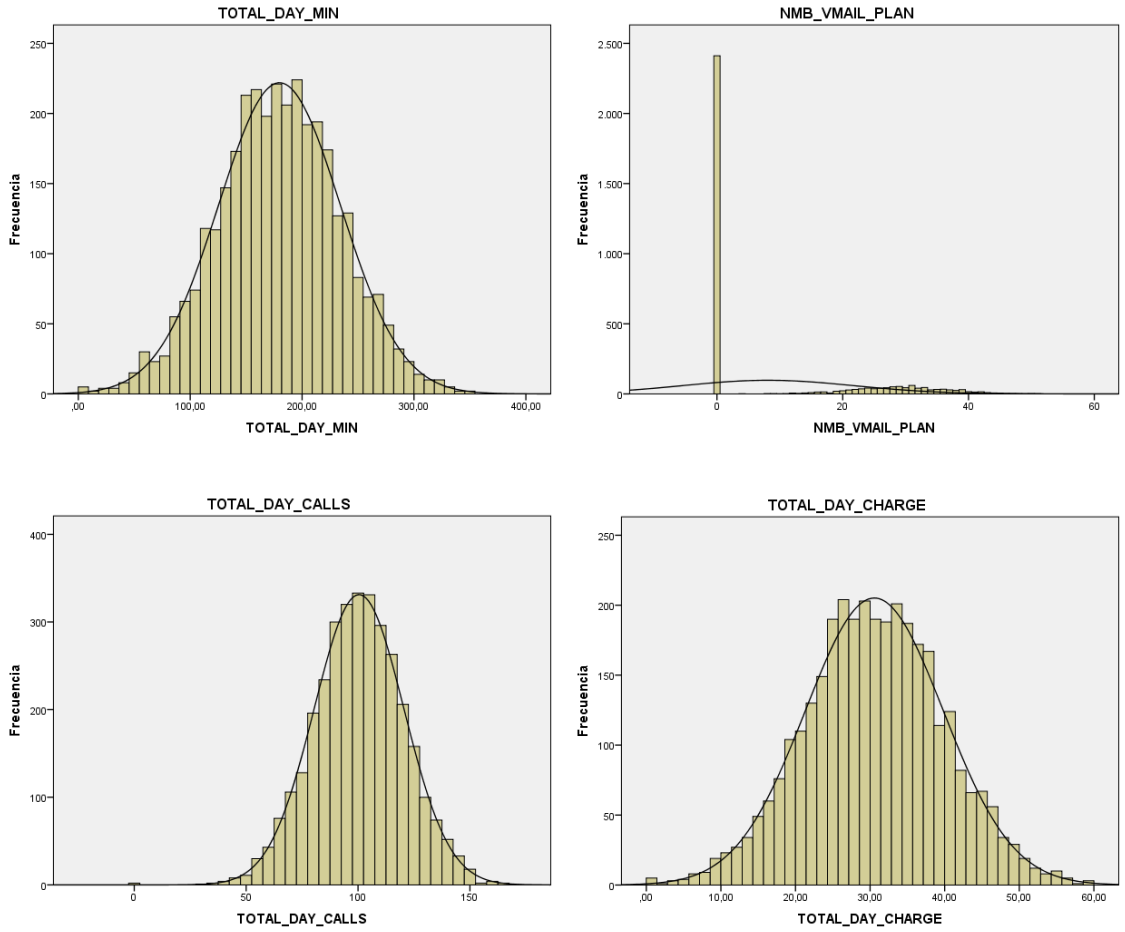
- 408 : San José, Santa Clara, Sunnyvale
- 415 : Novato, San Francisco, San Rafael
- 510 : Fremont, Hayward, Oakland

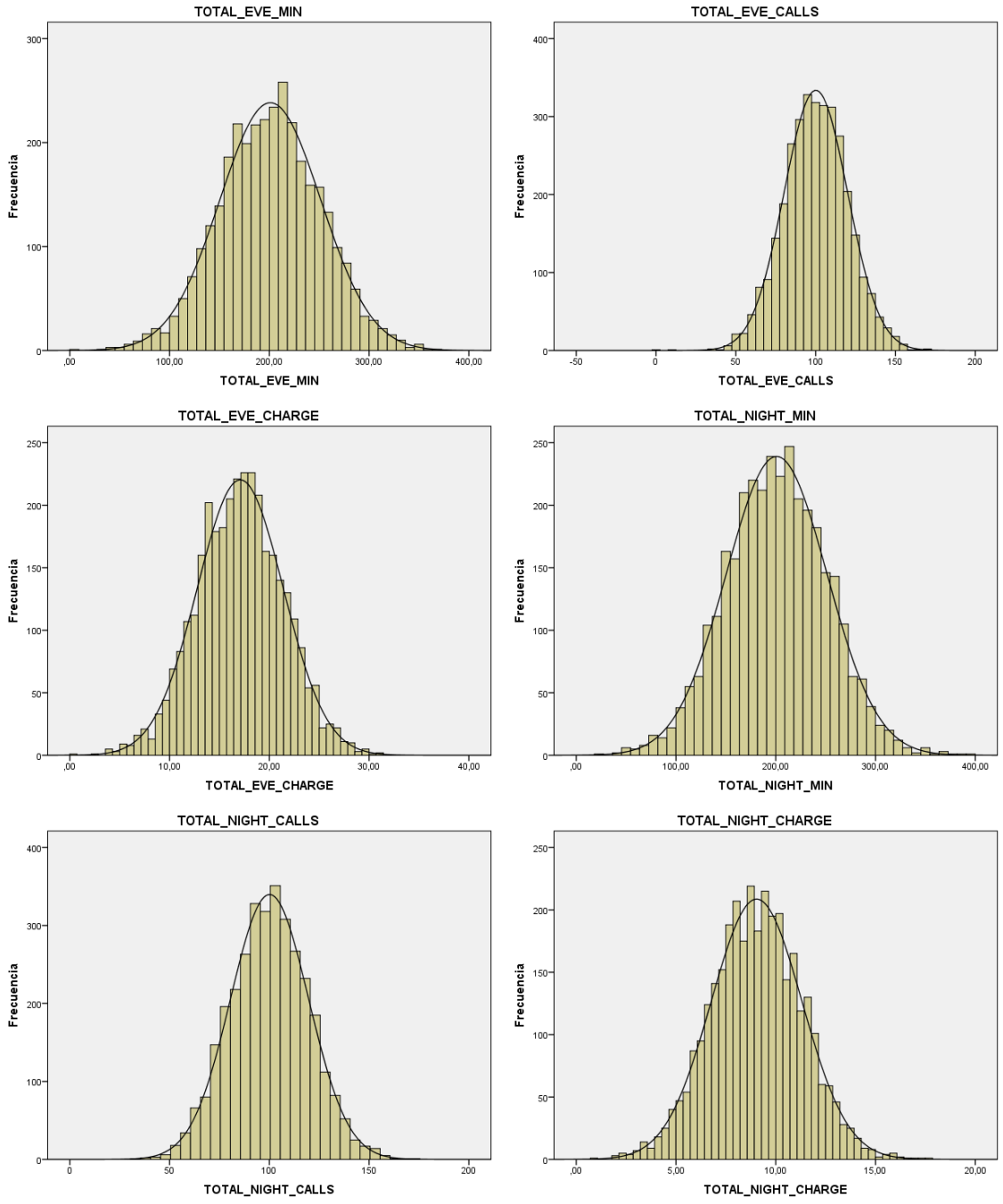
Dado que los valores anteriores corresponden únicamente al estado de California, y no concuerdan con los valores de la columna “STATE”, se sospecha que “AREACODE” fue cargada incorrectamente por lo cual se decide excluirla.

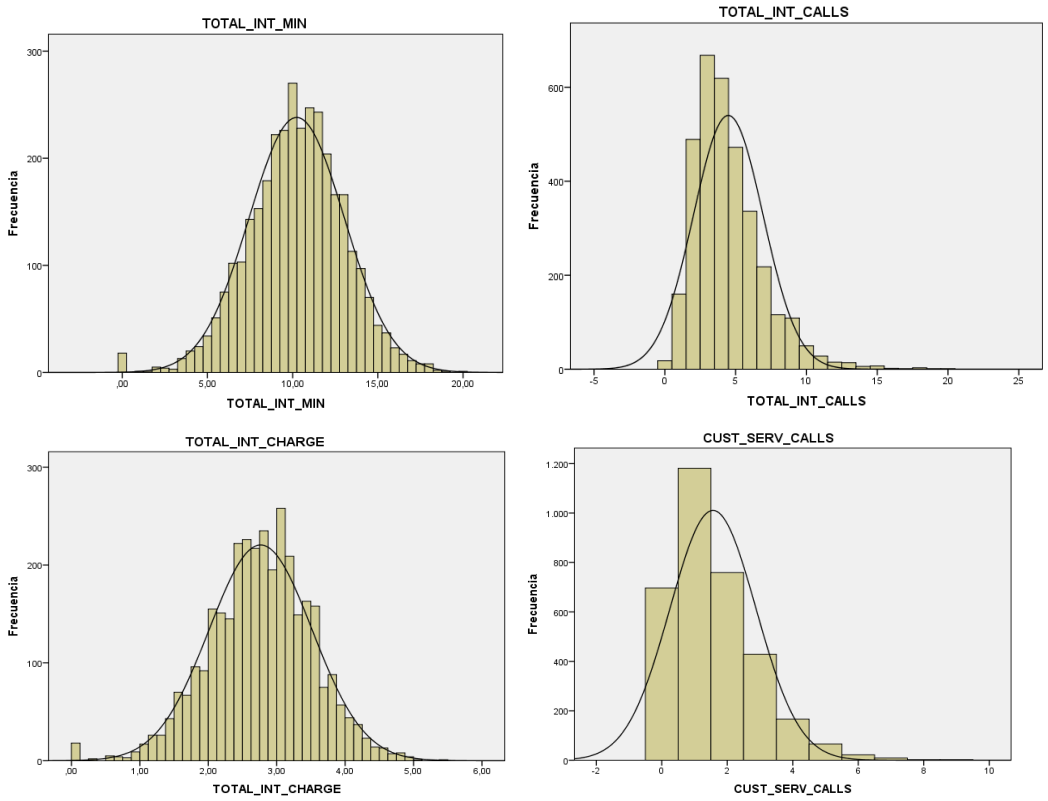
En cuanto a las variables INT_PLAN y VMAIL_PLAN notamos que están bastante desbalanceadas con alta frecuencia en “no”.

También se observa que la variable “CHURN” presenta un desbalanceo notorio en valor “Permanece” con 85,5% contra 14,5% en “Baja”.

Histograma de las variables numéricas:




TOTAL_EVE_MIN
Frecuencia
TOTAL_EVE_MIN
TOTAL_EVE_CALLS
Frecuencia
TOTAL_EVE_CALLS
TOTAL_EVE_CHARGE
Frecuencia
TOTAL_EVE_CHARGE
TOTAL_NIGHT_MIN
Frecuencia
TOTAL_NIGHT_MIN
TOTAL_NIGHT_CALLS
Frecuencia
TOTAL_NIGHT_CALLS
TOTAL_NIGHT_CHARGE
Frecuencia
TOTAL_NIGHT_CHARGE



Según los gráficos anteriores en la mayoría de los casos las variables consideradas parecieran seguir una distribución normal, comprobaremos dicha hipótesis con el test de Shapiro-Wilk[4]:

Variable	P-value
TTL.DAY.MIN	0.640
NMB.VMAIL.PLAN	3.4494E-65
TTL.DAY.CHARGE	0.640
TTL.EVE.MIN	0.712
TTL.DAY.CHARGE	0.640
TTL.NGHT.MIN	0.627
TTL.NGHT.CHARGE	0.623
TTL.INTL.MIN	7.998E-11
TTL.INTL.CHARGE	7.5345E-11

En la tabla anterior podemos observar que salvo 3 variables (P-value < 0.05) todas no rechazan la hipótesis nula (los datos provienen de una distribución normal) del test.

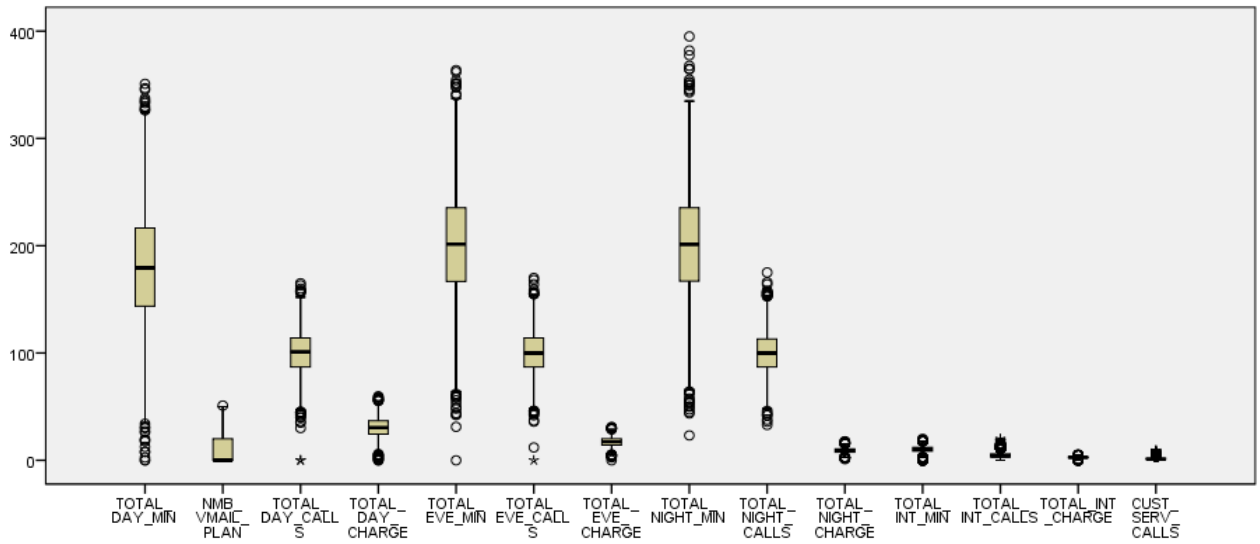
[4] Se adjunta en la sección anexo el código en R utilizado para el desarrollo del punto. Se tiene en cuenta únicamente las variables continuas.

La variable NMB_VMAIL_PLAN presenta el valor 0 con alta frecuencia, esta observación se debe a que las líneas que no cuentan con el plan “voice mail”, VMAIL_PLAN = “no”, (ca si el 90% de los usuarios) lógicamente no tienen ningún correo de voz. Por esta razón el v alor P de la variable resulta demasiado chico. Sin Embargo si excluimos los valores 0 la v ariable resulta tener un valor-p igual a 0.218, siendo esta normal.

Para las variables TTL.INTL.MIN y TTL.INTL.CHARGE la situación es similar a la observa ción del párrafo anterior. Los valores P de ambas variables dan muy chicos ya que hay lín eas que no poseen plan a llamadas internacionales; si excluimos dichos valores en el Test de Shapiro-Wilk las variables resultan normales (0.352 y 0.325 respectivamente).

Las variables que tienen como valor mínimo igual a 0 (ver tabla de estadística descriptiva) significan que existen usuarios que no realizaron ninguna llamada en el horario correspondiente.

Box-Plot:

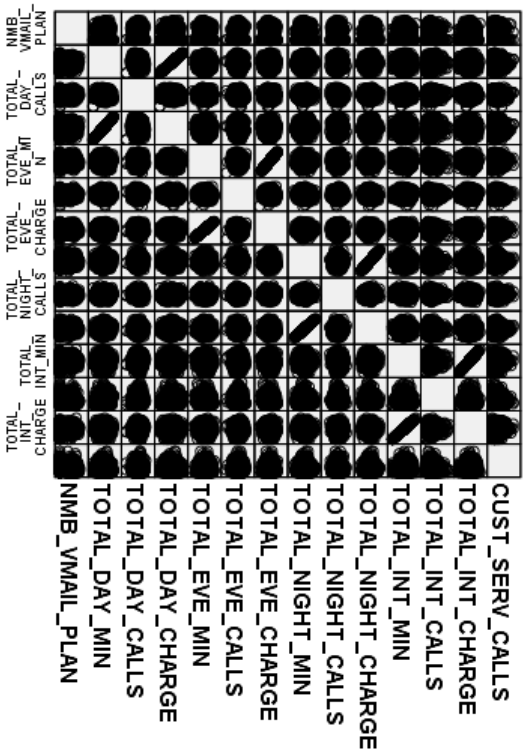


En los diagramas de cajas anteriores se puede observar la existencia de outiliers desde el punto de vista estadístico, sin embargo conociendo el negocio en cuestión, dichos valores son naturalmente posibles, por lo cual se decide no excluirlos del análisis.

Análisis de componentes principales

A continuación realizaremos un análisis de componentes principales (ACP) de nuestras variables continuas (con matriz de correlación) con el objetivo de estudiar si es posible reducir la dimensionalidad de las mismas.

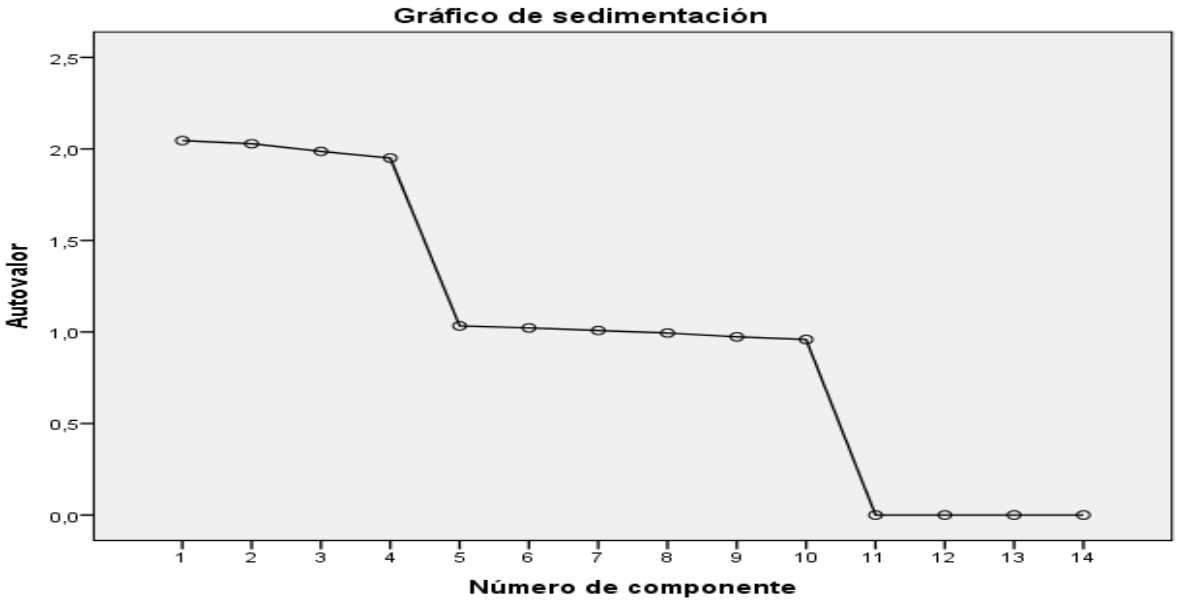
Dado que el uso de ACP tiene sentido cuando las variables estén fuertemente correlacionadas entre sí, verificaremos el mismo con un grafico de dispersión matricial:



Como se puede visualizar en la matriz anterior las variables correlacionadas parecerían ser:

- TOTAL_DAY_CHARGE con TOTAL_DAY_MIN
- TOTAL_EVE_CHARGE con TOTAL_EVE_MIN
- TOTAL_NIGHT_CHARGE con TOTAL_NIGHT_MIN
- TOTAL_INT_CHARGE con TOTAL_INT_MIN

Dado que se observa cierta correlación entre algunas variables, se procede con el ACP utilizando la matriz de correlaciones:



Varianza total explicada

Componente	Autovalores iniciales			Sumas de extracción de cargas al cuadrado		
	Total	% de varianza	% acumulado	Total	% de varianza	% acumulado
1	2,045	14,610	14,610	2,045	14,610	14,610
2	2,028	14,486	29,097	2,028	14,486	29,097
3	1,987	14,190	43,286	1,987	14,190	43,286
4	1,950	13,929	57,215	1,950	13,929	57,215
5	1,033	7,380	64,596	1,033	7,380	64,596
6	1,022	7,300	71,896	1,022	7,300	71,896
7	1,008	7,201	79,097	1,008	7,201	79,097
8	,994	7,101	86,198	,994	7,101	86,198
9	,973	6,953	93,151	,973	6,953	93,151
10	,959	6,848	100,000	,959	6,848	100,000
11	7,247E-6	5,176E-5	100,000	7,247E-6	5,176E-5	100,000
12	7,833E-7	5,595E-6	100,000	7,833E-7	5,595E-6	100,000
13	2,236E-7	1,597E-6	100,000	2,236E-7	1,597E-6	100,000
14	4,774E-8	3,410E-7	100,000	4,774E-8	3,410E-7	100,000

Como se puede observar en el grafico de sedimentación anterior hay dos rupturas muy notorias entre los componentes 4:5 y 10:11.Dado que la componente 10 presenta un auto valor cercano a 1 podemos considerar hasta dicha dimensión como importantes.

Sin embargo en la tabla de varianza explicada se observa que con 6 componentes se puede explicar casi el 72% de la varianza.

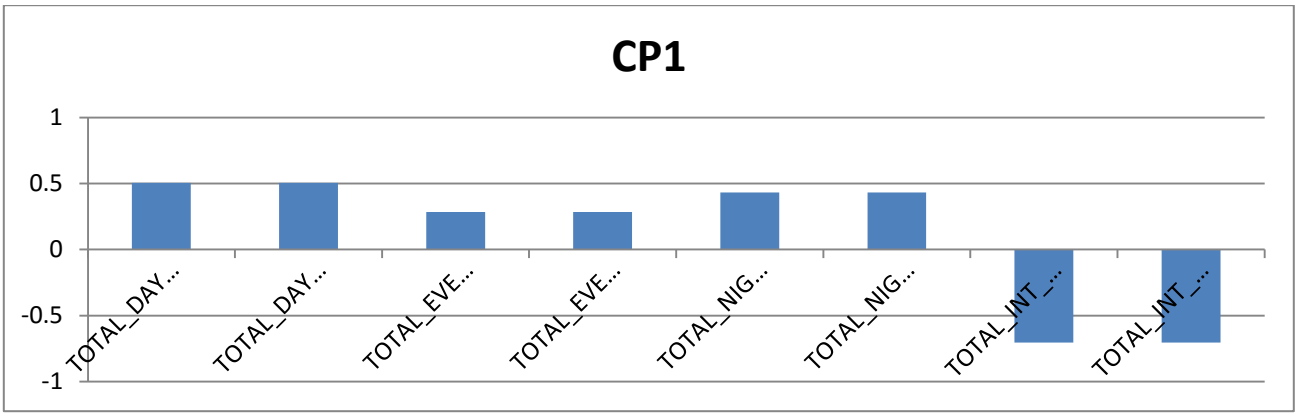
Matriz de componente

	Componente					
	1	2	3	4	5	6
NMB_VMAIL_PLAN	,013	,017	-,006	,034	,432	,332
TOTAL_DAY_MIN	,505	,159	,846	-,054	-,025	-,002
TOTAL_DAY_CALLS	-,016	-,058	,039	,029	,134	-,732
TOTAL_DAY_CHARGE	,505	,159	,846	-,054	-,025	-,002
TOTAL_EVE_MIN	,285	,738	-,265	,550	-,014	-,024
TOTAL_EVE_CALLS	-,005	-,009	,043	-,007	,247	-,131
TOTAL_EVE_CHARGE	,285	,738	-,265	,550	-,014	-,024
TOTAL_NIGHT_MIN	,432	-,665	-,090	,602	-,009	,008
TOTAL_NIGHT_CALLS	,054	,005	,021	,005	,280	,559
TOTAL_NIGHT_CHARGE	,432	-,665	-,090	,602	-,009	,008
TOTAL_INT_MIN	-,705	,005	,442	,552	-,029	,029
TOTAL_INT_CALLS	-,044	,022	,045	,025	,580	-,110
TOTAL_INT_CHARGE	-,705	,005	,442	,552	-,029	,029
CUST_SERV_CALLS	-,014	-,010	-,025	-,038	-,591	,177

Los 6 componentes considerados son de “forma” ya que poseen puntuaciones positivas y negativas.

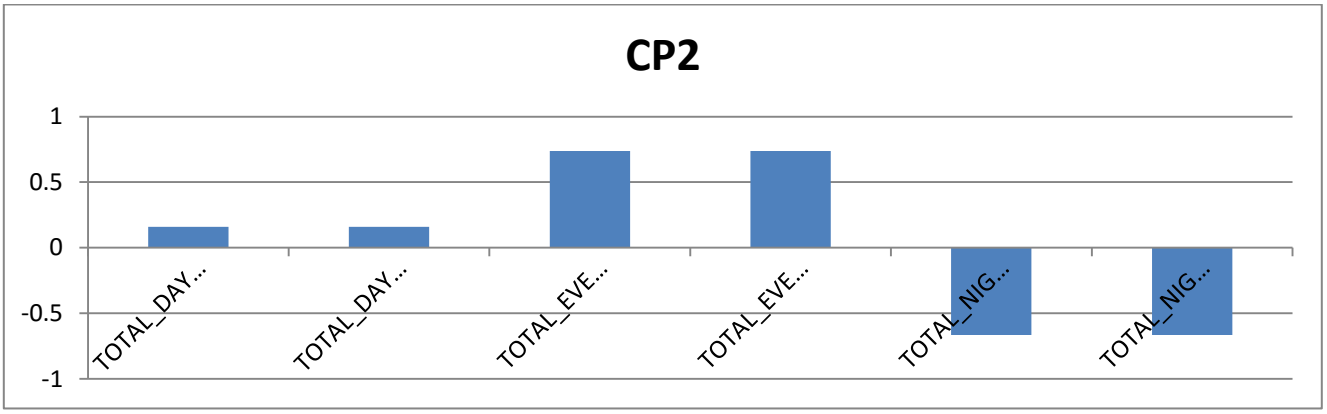
A continuación se describe cada una de las componentes (para dicho análisis se excluyeron puntuaciones inferiores a 0.1 en valor absoluto):

Componente 1:



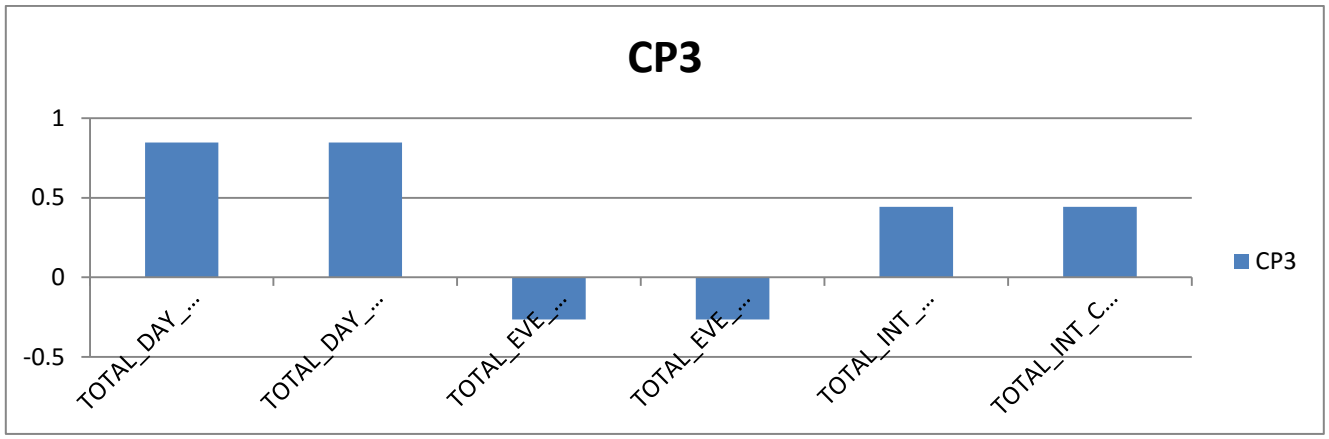
Un individuo que presenta un valor alto en la componente 1 significa que habla muchos minutos y gasta mucho crédito durante la mañana, tarde y noche y realiza pocas llamadas internacionales.

Componente 2:



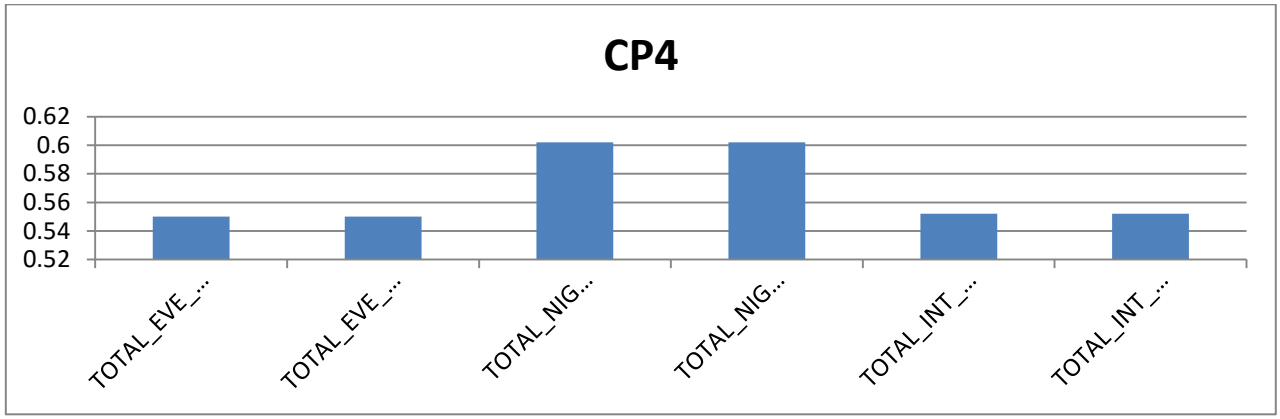
Un individuo que presenta un valor alto en la componente 2 significa que habla muchos minutos y gasta mucho crédito durante la mañana y tarde; y muestra un comportamiento opuesto durante la noche.

Componente 3:



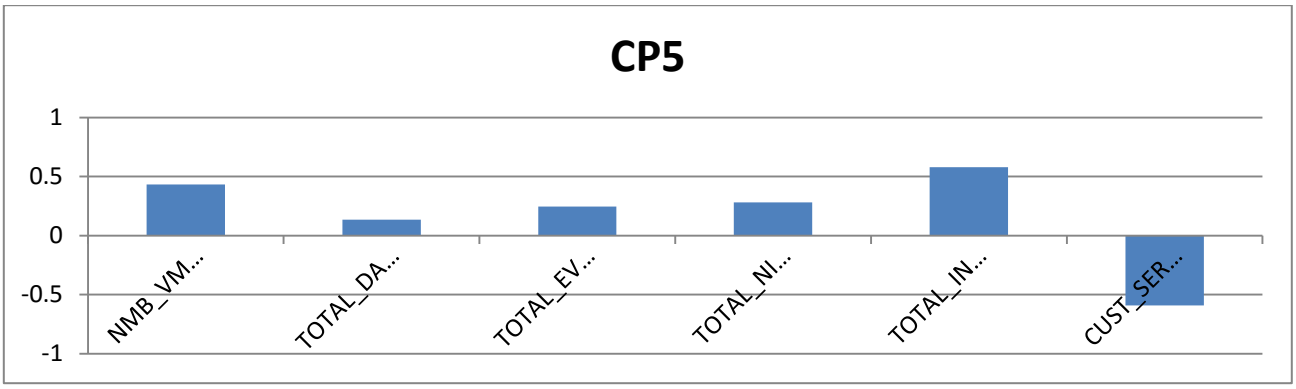
Un individuo que presenta un valor alto en la componente 3 significa que habla muchos minutos y gasta mucho crédito durante la mañana y noche; y muestra un comportamiento opuesto durante la tarde.

Componente 4 :



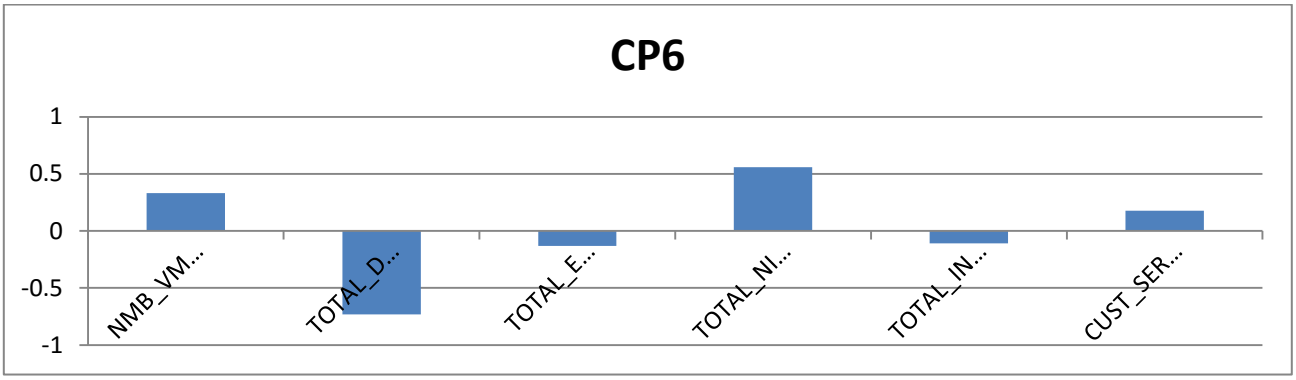
Un individuo que presenta un valor alto en la componente 4 significa que habla muchos minutos y gasta mucho crédito durante la tarde y noche y en las llamadas internacionales.

Componente 5:



Un individuo que presenta un valor alto en la componente 5 significa que utiliza frecuentemente el voice mail; realiza mucha llamadas durante la mañana, tarde y noche; y también en internacionales y al servicio de clientes.

Componente 6:



Un individuo que presenta un valor alto en la componente 6 significa que utiliza frecuentemente el mensaje de voz; realiza pocas llamadas durante la mañana, tarde y en internacionales y presenta una cantidad elevada en llamadas realizadas durante la noche y a atención a clientes.

Un aspecto interesante es observar si existe correlación, ya sea positiva o negativa, entre las componentes obtenidas y la variable “Churn”, se utiliza para ello el coeficiente de correlación de Pearson:

CP vs Churn	Coef. Pearson
CP1 vs Churn	0.094
CP2 vs Churn	0.074
CP3 vs Churn	0.174
CP4 vs Churn	0.096
CP5 vs Churn	-0.197
CP6 vs Churn	0.001

Como se observa en la tabla anterior, los valores son cercanos a 0 lo cual significa que no hay correlación alguna entre las componentes y la variable Churn.

Clasificación

Análisis discriminante

En esta sección analizaremos si es posible encontrar una función discriminante capaz de clasificar, con un nivel aceptable de aciertos, en “Baja” o “Permanece”.[6]

Antes de generar la función discriminante corroboraremos que se rechaza la hipótesis de igualdad de medias entre los grupos (utilizando variables numéricas):

Ho: Las medias de los dos grupos son iguales;

H1: Las medias de ambos grupos son distintas;

Pruebas multivalente

Efecto		Valor	F	Gl de hipótesis	gl de error	Sig.
CHURN	Traza de Hotelling	,128	30,432b	14,000	3318,000	,000

El cuadro anterior nos arroja un p-valor < 0.05 rechazando de esta manera la Ho, en conclusión se puede inferir que las medias de los dos grupos (Baja/Permanece) son diferentes lo cual se justifica realizar el análisis discriminante:

Resultados de pruebas

M de Box		596,870
F	Aprox.	10,772
	df1	55
	df2	2406469,283
	Sig.	,000

Como se observa en la tabla M de box anterior el valor p es < 0,05 (se rechaza el supuesto de homocedasticidad de que Σ1= Σ2) por lo cual utilizaremos el método de la distancia estandarizada de la puntuación discriminante del dato a los centroides.

[6] Se utilizan únicamente variables numéricas.

Coeficientes de función discriminante canónica estandarizadas

	Función 1
TOTAL_DAY_CALLS	,039
TOTAL_EVE_CALLS	,061
TOTAL_NIGHT_CALLS	-,014
TOTAL_INT_CALLS	-,212
CUST_SERV_CALLS	,623
NMB_VMAIL_PLAN	-,287
TOTAL_DAY_MIN	,665
TOTAL_EVE_MIN	,279
TOTAL_NIGHT_MIN	,164
TOTAL_INT_MIN	,191

Funciones en centroides de grupo

CHURN	Función 1
0	-,152
1	,865

Las funciones discriminantes canónicas sin estandarizar se han evaluado en medias de grupos

Con el objetivo de probar la efectividad discriminante de la regla se decide dividir el dataset original en dos subdatasets:

- 70% de datos para traning y;
- 30% de datos para testing.

Obteniendo de esta forma, el siguiente resultado:

			Pertenencia a grupos pronosticada		
		CHURN	0	1	Total
training	Recuento	0	1429	541	1970
		1	99	247	346
	%	0	72,5	27,5	100,0
		1	28,6	71,4	100,0
testing	Recuento	0	651	229	880
		1	46	91	137
	%	0	74,0	26,0	100,0
		1	33,6	66,4	100,0

Observaciones:

- Con el set de training se clasificó correctamente 72,5% para clase 0 (Permanece) y 71.4% para clase 1 (Baja).
- Con el set de validación se clasificó correctamente 74% para clase 0 (Permanece) y 66.4% para clase 1 (Baja).

Si bien el porcentaje de acierto de la clase 1 en el set de validación se redujo un 5% en general la clasificación obtenida es aceptable ya que ronda en 70%.

Árbol (Random Forest)

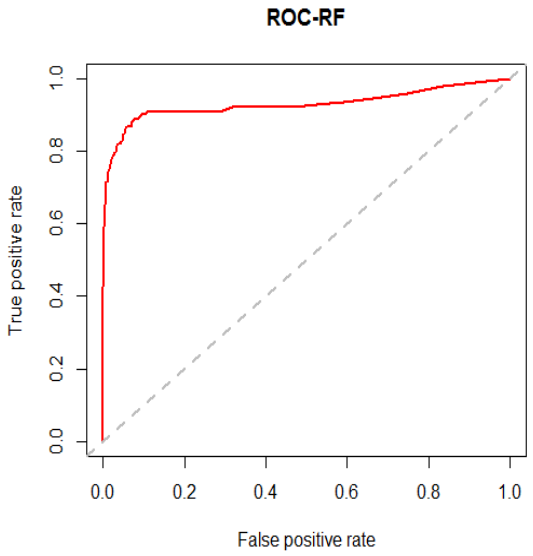
Otro algoritmo de clasificación muy utilizado en la actualidad son los arboles de decisión y en particular “Random Forest”[7].

En esta sección analizaremos el funcionamiento del “Random Forest” en nuestra base, tomando distintas medida de performance como área bajo la curva ROC, matriz de confusión, precisión, especificidad y recall.

Se divide de manera aleatoria el dataset inicial en dos subdatasets donde el 70% se utilizó para entrenar y el 30% para validar, siendo “CHURN” la variable objetivo.

Con el objetivo de encontrar los óptimos parámetros que permita obtener el mejor árbol se probó (sobre el dataset de validación) variando de 5 a 500 la cantidad de árboles a considerar:

Cantidad de Arboles	ROC
5	0.9003115
10	0.9199077
50	0.9293468
100	0.9288478
150	0.9297605
200	0.9249416
250	0.92972
300	0.928138
350	0.9266858
400	0.9262721
450	0.9278824
500	0.9265398



[7] Se adjunta como anexo (ScriptR.txt) el código en R utilizado para el desarrollo del punto.

Como se observa en la tabla, si bien todos los valores son muy similares, el Random Forest generado por 150 árboles presenta el ROC más alto. Siendo éste 0.9297605 podemos afirmar que es un número más que aceptable. Comprobaremos dicha cantidad de arboles con otras medidas de performance:

- Matriz de confusión:

	PREDICHO	
	FALSO	VERDADERO
FALSO	850	6
VERDADERO	42	102

- Recall(Sensibilidad) : 0.708
- Especificidad:0.991
- Precision:0.944

Como observamos en general, salvo la sensibilidad que presenta una leve baja, las medid as de performance para el parámetro elegido son buenas.

Además podemos comparar las matrices de confusión del análisis discriminante con el Ra ndom Forest anterior y concluir que el segundo método es claramente superior al primero.

Clustering

K-Means

A continuación, con el objetivo de identificar distintos grupos de comportamiento de usuarios, se lleva a cabo el análisis de conglomerados con todas las variables (estandarizadas) del negocio.

El método de clasificación utilizado es K-medias (en R).

Teniendo en cuenta que K-medias acepta únicamente variables numéricas, se convierte las categóricas a valores numéricos.

Dado que a priori se desconoce un posible valor óptimo de "K", se prueba de manera empírica distinta cantidad de clústeres, luego se utiliza como criterio de performance el coeficiente de Silhouette para determinar el número de clúster a considerar.[8]

A continuación el valor del coeficiente de cada k:

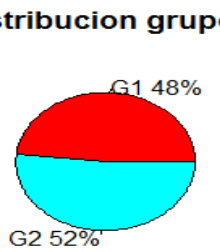
K	Silhouette
2	0.09409951
3	0.009706656
4	0.03163529
5	0.01525658
6	0.005859431
7	0.005103091
8	0.01892225
9	0.01667195
10	-0.001624427

Como se puede observar en la tabla anterior los valores obtenidos en general son muy pequeños, lo cual significa que los agrupamientos no son muy de buena calidad sin embargo el coeficiente de Silhouette correspondiente a k=2 presenta una leve mejora entre los K probados, con lo cual analizaremos en forma particular el clustering formado por 2 grupos.

Si analizamos los coeficientes de Sihouettes por grupo (G1 = 0.0900 y G2 = 0.0979) observamos que los valores no presentan diferencias notorias con lo cual la calidad de cada clúster obtenido es similar al valor promedio obtenido.

[8] Se adjunta en la sección anexo el código en R utilizado para el desarrollo del punto.

Los 3333 individuos de la base fueron clasificados con la siguiente distribución:



A continuación las medias (estandarizadas) de cada variable por grupo:

Variable\Grupo	G1	G2
ESTADO	0.03086987	-0.02894947
ACCOUNT	-0.02202188	0.02065192
INTL.PLAN	0.03286447	-0.03081999
VMAIL.PLAN	0.290721	-0.2726354
NMB.VMAIL.MSSG	0.28485	-0.2671296
TTL.DAY.MIN	0.4949294	-0.4641402
TTL.DAY.CALLS	-0.03041858	0.02852626
TTL.DAY.CHARGE	0.4949291	-0.46414
TTL.EVE.MIN	0.4224633	-0.3961821
TTL.EVE.CALLS	-0.04438698	0.0416257
TTL.EVE.CHARG	0.42247	-0.3961885
TTL.NGHT.MIN	0.2927346	-0.2745238
TTL.NGHT.CALLS	0.00834055	-0.007821691
TTL.NGHT.CHARGE	0.2927033	-0.2744945
TTL.INTL.MIN	-0.3234017	0.3032832
TTL.INTL.CALLS	0.01880393	-0.01763415
TTL.INTL.CHARGE	-0.3233661	0.3032497
CUST.SERV.CALLS	-0.02728099	0.02558386

- Grupo 1:

  Los individuos que pertenecen a este clúster son caracterizados por tener las medias de “ESTADO” , ”INTL.PLAN”,”VMAIL.PLAN” , ”NMB.VMAIL.MSSG” , ”TTL.DAY.MIN” , ”TTL.DAY.CHARGE” , ”TTL.EVE.MIN” , ”TTL.EVE.CHARG” , ”TTL.NGHT.MIN”,”TTL.NGHT.CALLS”,”TTL.NGHT.CHARGE” y ”TTL.INTL.CALLS” más elevadas que el Grupo 2.Lo cual se traduce a que son líneas con plan de llamadas a internacionales, plan de voice mail, cantidad elevada de voice mail,

muchos minutos y créditos usados durante la mañana y tarde, muchos minutos, créditos y llamadas durante la noche y alta frecuencia en llamadas internacionales.

- Grupo 2:

  Los pertenecientes a este grupo son usuarios que presentan medias levemente bajas para las variables mencionadas en el grupo anterior.

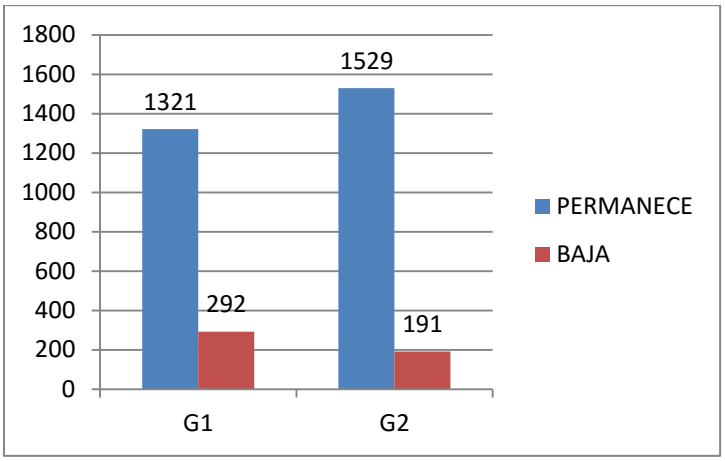
  Cabe destacar que las líneas de este grupo presentan medias superiores que el Grupo 1 en las variables “ACCOUNT”,”TTL.DAY.CALLS”, “TTL.EVE.CALLS”, “TTL.INTL.MIN”, “TTL.INTL.CHARGE” y “CUST.SERV.CALLS”.Lo cual significa que son usuarios que realizan llamadas frecuentes durante la mañana, tarde; usan muchos minutos y créditos en llamadas internacionales y llaman más veces al centro de atención a clientes que el grupo anterior.

Teniendo en cuenta las medias de cada variable por grupo podríamos indagar si dichos valores están relacionados de alguna manera con los valores de las componentes de PCA:

	C1	C2	C3	C4	C5	C6
G1	0.9083	0.8130	0.1289	0.1824	-0.4829	-0.6153
G2	-0.3999	-0.3194	0.4705	-0.0987	-0.2399	-0.1268

Al analizar las correlaciones entre las medias y los componentes se observan valores interesantes como las correlaciones positivas entre “Grupo1” y “Componente 1” (0.9083); “Grupo1” y “Componente2” (0.8130); y en menor medida “Grupo2” y “Componente 3” (0.4705).También podemos ver correlaciones negativas entre “Grupo 1” y “Componente 5” (-0.4829) ; “Grupo1” y “Componente 6” (-0.6153) ; “Grupo 2” y “Componente1” ; “Grupo 2” y “Componente 2”.

Otro aspecto interesante es ver como se distribuye la variable clase (CHURN) en los grupos obtenidos anteriormente:



Como se observa en el gráfico de barras, el Grupo 1 muestra un número elevado de caso s (292 líneas) de BAJA con respecto al Grupo 2(191).Con lo cual podemos sospechar que los individuos que pertenecen al Grupo 1 tienen una probabilidad más elevada a darse d e BAJA.

Comprobaremos la observación anterior con un test de independencia (Chi cuadrado):

- Ho: Los grupos son independientes de la variable Churn
- H1: Los grupos no son independientes de la variable Churn

El valor P que obtenemos del test es igual a 1.296e-08 siendo menor a 0.05 con lo cual r echazamos la Ho y concluimos que ambas variable (Grupo vs Churn) son dependientes.

Partitioning Around Medoids (PAM)

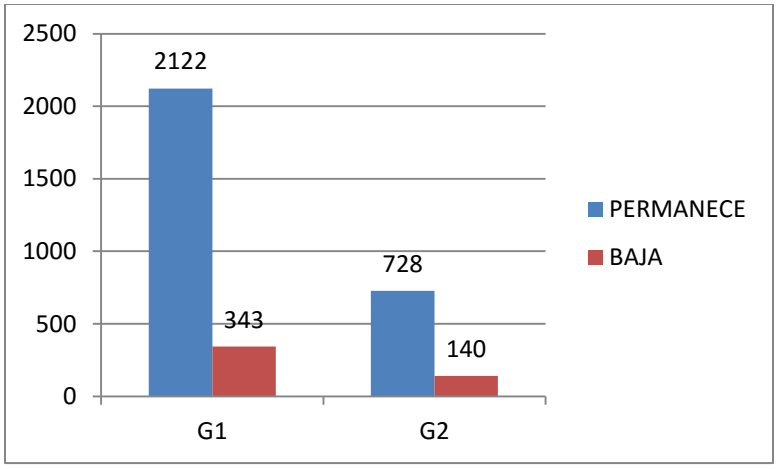
En este punto vamos a probar el método de clustering por partición y compararemos la eficiencia del mismo con el obtenido con K-medias

Como al igual que con K-medias, a priori se desconoce el valor óptimo de K, con lo cual se prueba desde K = 2 a K = 10 y se analiza el coeficiente de Silhouette de cada K[9]:

K	Silhouette
2	0.168
3	0.145
4	0.130
5	0.116
6	0.116
7	0.116
8	0.110
9	0.104
10	0.104

Como se observa en la tabla anterior, nuevamente, el valor óptimo de K es 2 ya que dicho clustering presenta el valor del coeficiente más alto entre los K probados. Cabe destacar que el valor del coeficiente obtenido con el método de partición (0.168756) es claramente mayor al obtenido con K-medias (0.09409951), con lo cual podemos afirmar que el clustering por PAM es más eficiente.

La variable CHURN en los grupos obtenidos:



[9] Se adjunta en la seccón anexo el código en R utilizado para el desarrollo del punto.

Como se observa en el grafico anterior en el grupo 1 se concentra el 73% de los usuarios siendo los casos de BAJA similares en magnitud porcentual en ambos grupos (13 y 16%).

Al igual que con K-medias, comprobaremos la observación anterior con un test de independencia:

- Ho: Los grupos son independientes de la variable Churn
- H1: Los grupos no son independientes de la variable Churn

El valor P es igual a 0.1241 siendo mayor a 0.05 con lo cual no rechazamos la Ho de que ambas variables son independientes.

Conclusión

Con el desarrollo del presente trabajo práctico se pudo comprobar la existencia de una tendencia/caracterización de usuarios que permite discriminar, con un porcentaje aceptable, entre las líneas que continuarán y las que se darán de baja. Los modelos construidos con Análisis discriminante y Random Forest servirán desde el punto de vista de negocios a anticipar a las posibles bajas y de esta manera planear distintas estrategias de retención de clientes.

A su vez, con las técnicas de clustreing, empleando métodos de K-medias y PAM, se pudo reconocer 2 grupos de comportamiento distintos de usuarios. Con la información obtenida podríamos ofrecerles a los usuarios distintos paquetes personalizados de llamadas según su modo de usar la línea.

Como una futura idea de desarrollo seria interesante poder enriquecer el dataset, además de las variables que ya poseemos, incorporando información demográfica (Edad, profesión, sexo, ingresos, etc.) de los usuarios, aumentando de esta manera la eficiencia de los algoritmos utilizados en el presente desarrollo.

Bibliografía

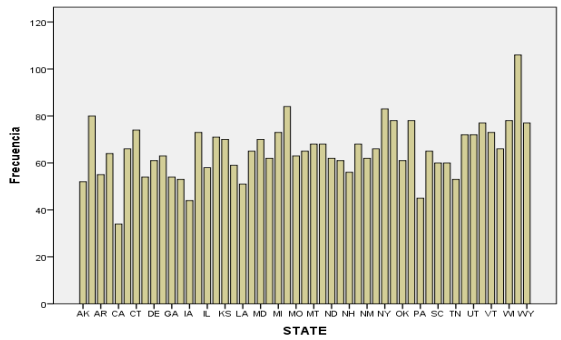
Apuntes de la catedra de Análisis inteligente de datos.

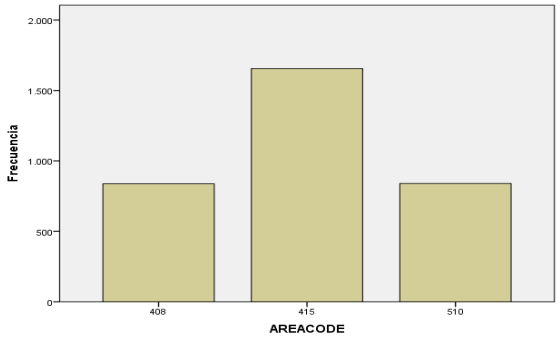
Apuntes de la catedra de Datamining en ciencias y tecnologías.

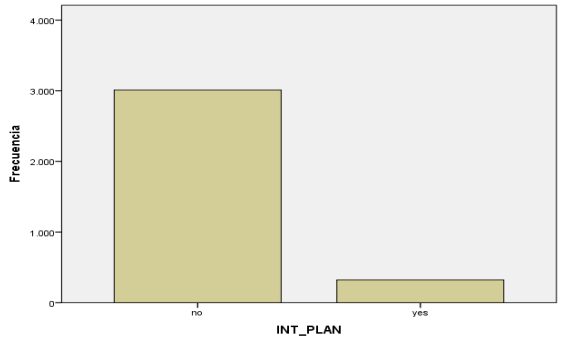
Apuntes de la catedra de Enfoque Estadístico del Aprendizaje.

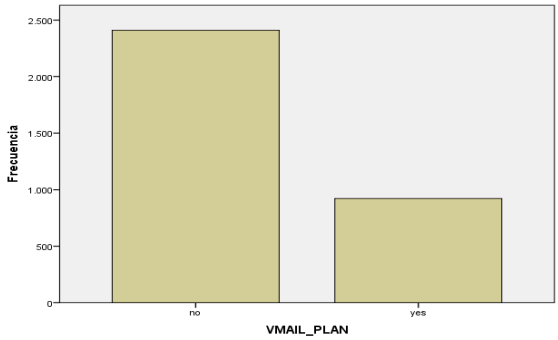
Anexo

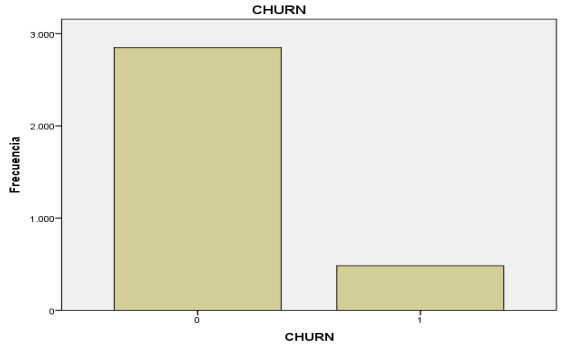
Gráfico de barras de las variables categóricas:











```
library(randomForest)
library(ROCR)
library(cluster)
set.seed(19900330)
dataTelecomChurn <- read.table("Churn.csv", dec = ",",sep = ";",header = T)
dataTelecomChurn$ESTADO<-as.numeric(dataTelecomChurn$ESTADO)

##TEST DE NORMALIDAD SHAPIRO
a1<-shapiro.test(dataTelecomChurn$TTL.DAY.MIN)
a1$p.value
a2<-shapiro.test(dataTelecomChurn$NMB.VMAIL.MSSG)
a2$p.value
a3<-shapiro.test(dataTelecomChurn$TTL.DAY.CALLS)
a3$p.value
a4<-shapiro.test(dataTelecomChurn$TTL.DAY.CHARGE)
a4$p.value
a5<-shapiro.test(dataTelecomChurn$TTL.EVE.MIN)
a5$p.value
a6<-shapiro.test(dataTelecomChurn$TTL.EVE.CALLS)
a6$p.value
a7<-shapiro.test(dataTelecomChurn$TTL.DAY.CHARGE)
a7$p.value
a8<-shapiro.test(dataTelecomChurn$TTL.NGHT.MIN)
a8$p.value
a9<-shapiro.test(dataTelecomChurn$TTL.NGHT.CALLS)
a9$p.value
a10<-shapiro.test(dataTelecomChurn$TTL.NGHT.CHARGE)
a10$p.value
a11<-shapiro.test(dataTelecomChurn$TTL.INTL.MIN)
a11$p.value
a12<-shapiro.test(dataTelecomChurn$TTL.INTL.CALLS)
a12$p.value
a13<-shapiro.test(dataTelecomChurn$TTL.INTL.CHARGE)
a13$p.value
a14<-shapiro.test(dataTelecomChurn$CUST.SERV.CALLS)
a14$p.value
#TEST SHAPIRO NMB.VMAIL.MSSG
NMB.VMAIL.MSSG<- shapiro.test(dataTelecomChurn$NMB.VMAIL.MSSG[dataTelecomChurn$NMB.VMAIL.MSSG > 0])
NMB.VMAIL.MSSG$p.value
#TEST SHAPIRO TOTAL_INT_MIN
TTL.INTL.MIN<- shapiro.test(dataTelecomChurn$TTL.INTL.MIN[dataTelecomChurn$TTL.INTL.MIN > 0])
TTL.INTL.MIN$p.value
#TEST SHAPIRO TOTAL_INT_CHARGE
TTL.INTL.CHARGE<- shapiro.test(dataTelecomChurn$TTL.INTL.CHARGE[dataTelecomChurn$TTL.INTL.CHARGE > 0])
TTL.INTL.CHARGE$p.value
dataTelecomChurn$CUST.SERV.CALLS
a<-fitdistr(dataTelecomChurn$CUST.SERV.CALLS,"Poisson")
a
mean(dataTelecomChurn$CUST.SERV.CALLS)
CUST.SERV.CALLS <-
poisson.test(sum(dataTelecomChurn$CUST.SERV.CALLS),length(dataTelecomChurn$CUST.SERV.CALLS))
CUST.SERV.CALLS$p.value
##RANDOM FOREST
set.seed(19900330)
dataTelecomChurn.regs <- sample(1:nrow(dataTelecomChurn), size = nrow(dataTelecomChurn) * 0.7)
dataTelecomChurnTrain <- dataTelecomChurn[dataTelecomChurn.regs,]
dataTelecomChurnTest <-  dataTelecomChurn[-dataTelecomChurn.regs,]
rf0 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=5,nodesize=5)
rf1 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=10,nodesize=5)
rf2 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=50,nodesize=5)
rf3 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=100,nodesize=5)
rf4 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=150,nodesize=5)
rf5 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=200,nodesize=5)
rf6 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=250,nodesize=5)
rf7 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=300,nodesize=5)
rf8 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=350,nodesize=5)
rf9 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=400,nodesize=5)
```

```
rf10 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=450,nodesize=5)
rf11 <- randomForest(factor(CHURN)~.,data=dataTelecomChurnTrain,ntree=500,nodesize=5)

trainResultrf0_prob<-predict(rf0,newdata=dataTelecomChurnTest,type="prob")
trainResultrf1_prob<-predict(rf1,newdata=dataTelecomChurnTest,type="prob")
trainResultrf2_prob<-predict(rf2,newdata=dataTelecomChurnTest,type="prob")
trainResultrf3_prob<-predict(rf3,newdata=dataTelecomChurnTest,type="prob")
trainResultrf4_prob<-predict(rf4,newdata=dataTelecomChurnTest,type="prob")
trainResultrf5_prob<-predict(rf5,newdata=dataTelecomChurnTest,type="prob")
trainResultrf6_prob<-predict(rf6,newdata=dataTelecomChurnTest,type="prob")
trainResultrf7_prob<-predict(rf7,newdata=dataTelecomChurnTest,type="prob")
trainResultrf8_prob<-predict(rf8,newdata=dataTelecomChurnTest,type="prob")
trainResultrf9_prob<-predict(rf9,newdata=dataTelecomChurnTest,type="prob")
trainResultrf10_prob<-predict(rf10,newdata=dataTelecomChurnTest,type="prob")
trainResultrf11_prob<-predict(rf11,newdata=dataTelecomChurnTest,type="prob")

#ROC
forestpred = prediction(trainResultrf4_prob[,2], dataTelecomChurnTest$CHURN)
auc <- performance(forestpred,"auc")
auc <- unlist(slot(auc, "y.values"))
auc
forestperf = performance(forestpred, "tpr", "fpr")
plot(forestperf,main="ROC-RF",col=2,lwd=2)
abline(a=0,b=1,lwd=2,lty=2,col="gray")
#OTRAS MEDIDAS
trainResultrf4_prob<-predict(rf4,newdata=dataTelecomChurnTest,type ="response")
sb.m2.tc <- table(dataTelecomChurnTest$CHURN, trainResultrf4_prob)
sb.m2.tc
recall <- sb.m2.tc[2,2] / ( sb.m2.tc[2,2] +  sb.m2.tc[2,1])
recall
especificidad <- sb.m2.tc[1,1] / (sb.m2.tc[1,1] + sb.m2.tc[1,2])
especificidad
precision <- sb.m2.tc[2,2] / ( sb.m2.tc[2,2] +  sb.m2.tc[1,2])
precision

##K-MEDIAS
dataTelecomChurnKmean<-dataTelecomChurn
dataTelecomChurnKmean$CHURN<-NULL
dataTelecomChurnKmean$INTL.PLAN <- as.numeric(dataTelecomChurnKmean$INTL.PLAN)
dataTelecomChurnKmean$VMAIL.PLAN <- as.numeric(dataTelecomChurnKmean$VMAIL.PLAN)

dataTelecomChurnKmean.nrm <- data.frame(scale(dataTelecomChurnKmean, scale = TRUE, center = TRUE))
dataTelecomChurnKmean.std <- scale(dataTelecomChurnKmean, scale = TRUE, center = TRUE)
dataTelecomChurnKmean.nrm.dist <- as.matrix(dist(dataTelecomChurnKmean, diag=T, upper=T))

kmeans.2 <- kmeans(dataTelecomChurnKmean.nrm, centers = 2)
dataTelecomChurnKmean.kmeans.2 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.2$cluster)
dataTelecomChurnKmean.kmeans.2
kmeans.3 <- kmeans(dataTelecomChurnKmean.nrm, centers = 3)
dataTelecomChurnKmean.kmeans.3 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.3$cluster)
dataTelecomChurnKmean.kmeans.3
kmeans.4 <- kmeans(dataTelecomChurnKmean.nrm, centers = 4)
dataTelecomChurnKmean.kmeans.4 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.4$cluster)
kmeans.5 <- kmeans(dataTelecomChurnKmean.nrm, centers = 5)
dataTelecomChurnKmean.kmeans.5 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.5$cluster)
kmeans.6 <- kmeans(dataTelecomChurnKmean.nrm, centers = 6)
dataTelecomChurnKmean.kmeans.6 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.6$cluster)
kmeans.7 <- kmeans(dataTelecomChurnKmean.nrm, centers = 7)
dataTelecomChurnKmean.kmeans.7 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.7$cluster)
kmeans.8 <- kmeans(dataTelecomChurnKmean.nrm, centers = 8)
dataTelecomChurnKmean.kmeans.8 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.8$cluster)
kmeans.9 <- kmeans(dataTelecomChurnKmean.nrm, centers = 9)
dataTelecomChurnKmean.kmeans.9 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.9$cluster)
kmeans.10 <- kmeans(dataTelecomChurnKmean.nrm, centers = 10)
dataTelecomChurnKmean.kmeans.10 <- data.frame(dataTelecomChurnKmean.nrm, kmeans.10$cluster)

dataTelecomChurnKmean.kmeans.2.sil <- silhouette(kmeans.2$cluster, dataTelecomChurnKmean.nrm.dist)
```

```
dataTelecomChurnKmean.kmeans.3.sil <- silhouette(kmeans.3$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.4.sil <- silhouette(kmeans.4$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.5.sil <- silhouette(kmeans.5$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.6.sil <- silhouette(kmeans.6$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.7.sil <- silhouette(kmeans.7$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.8.sil <- silhouette(kmeans.8$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.9.sil <- silhouette(kmeans.9$cluster, dataTelecomChurnKmean.nrm.dist)
dataTelecomChurnKmean.kmeans.10.sil <- silhouette(kmeans.10$cluster, dataTelecomChurnKmean.nrm.dist)


summary(dataTelecomChurnKmean.kmeans.2.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.2.sil)$clus.avg.widths
summary(dataTelecomChurnKmean.kmeans.3.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.4.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.5.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.6.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.7.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.8.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.9.sil)$avg.width
summary(dataTelecomChurnKmean.kmeans.10.sil)$avg.width


kmeans.2$centers
lbls <- c("G1","G2")
pct <- round(table(kmeans.2$cluster)/sum(table(kmeans.2$cluster))*100)
lbls <- paste(lbls, pct) # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels
pie(table(kmeans.2$cluster),labels = lbls, col=rainbow(length(lbls)),main="Distribucion grupos")
table(kmeans.2$cluster)
tablaObservada=table(dataTelecomChurn$CHURN,kmeans.2$cluster)
tablaObservada
chisq.Observada=chisq.test(tablaObservada)
chisq.Observada
##PAM
pam.2 <- pam(dataTelecomChurnKmean.nrm.dist, k = 2, diss = T)

dataTelecomChurnKmean.pam.2 <- data.frame(dataTelecomChurnKmean.nrm, pam.2$cluster)
tablaObservada=table(dataTelecomChurn$CHURN,pam.2$clustering)
chisq.Observada=chisq.test(tablaObservada)
chisq.Observada
dataTelecomChurn$CHURN[pam.2$id.med]
pam.3 <- pam(dataTelecomChurnKmean.nrm.dist, k = 3, diss = T)
dataTelecomChurnKmean.pam.3 <- data.frame(dataTelecomChurnKmean.nrm, pam.3$cluster)
pam.4 <- pam(dataTelecomChurnKmean.nrm.dist, k = 4, diss = T)
dataTelecomChurnKmean.pam.4 <- data.frame(dataTelecomChurnKmean.nrm, pam.4$cluster)
pam.5 <- pam(dataTelecomChurnKmean.nrm.dist, k = 5, diss = T)
dataTelecomChurnKmean.pam.5 <- data.frame(dataTelecomChurnKmean.nrm, pam.5$cluster)
pam.6 <- pam(dataTelecomChurnKmean.nrm.dist, k = 6, diss = T)
dataTelecomChurnKmean.pam.6 <- data.frame(dataTelecomChurnKmean.nrm, pam.6$cluster)
pam.7 <- pam(dataTelecomChurnKmean.nrm.dist, k = 7, diss = T)
dataTelecomChurnKmean.pam.7 <- data.frame(dataTelecomChurnKmean.nrm, pam.7$cluster)
pam.8 <- pam(dataTelecomChurnKmean.nrm.dist, k = 8, diss = T)
dataTelecomChurnKmean.pam.8 <- data.frame(dataTelecomChurnKmean.nrm, pam.8$cluster)
pam.9 <- pam(dataTelecomChurnKmean.nrm.dist, k = 9, diss = T)
dataTelecomChurnKmean.pam.9 <- data.frame(dataTelecomChurnKmean.nrm, pam.9$cluster)
pam.10 <- pam(dataTelecomChurnKmean.nrm.dist, k = 10, diss = T)
dataTelecomChurnKmean.pam.10 <- data.frame(dataTelecomChurnKmean.nrm, pam.10$cluster)
pam.11 <- pam(dataTelecomChurnKmean.nrm.dist, k = 11, diss = T)
dataTelecomChurnKmean.pam.11 <- data.frame(dataTelecomChurnKmean.nrm, pam.11$cluster)
pam.12 <- pam(dataTelecomChurnKmean.nrm.dist, k = 12, diss = T)
dataTelecomChurnKmean.pam.12 <- data.frame(dataTelecomChurnKmean.nrm, pam.12$cluster)
# Silhouette
summary(silhouette(pam.2))$avg.width
summary(silhouette(pam.3))$avg.width
summary(silhouette(pam.4))$avg.width
summary(silhouette(pam.5))$avg.width
summary(silhouette(pam.6))$avg.width
```

```
summary(silhouette(pam.7))$avg.width
summary(silhouette(pam.8))$avg.width
summary(silhouette(pam.9))$avg.width
summary(silhouette(pam.10))$avg.width
summary(silhouette(pam.11))$avg.width
summary(silhouette(pam.12))$avg.width
```