Caso_Churn

Bryan Velazco

2024-12-21

1. Análisis

```
Base <- read_excel("UV6696-XLS-ENG.xlsx", sheet = "Case Data")
names (Base) = c("ID", "Customer.Age", "Churn", "CHI.Score.M0",
                 "CHI.Score.0.1","Support.Cases.M0", "Support.Cases.0.1",
                 "SP.M0","SP.0.1","Logins.0.1","Blog.Articles.0.1",
                 "Views.0.1","X.DaysLL.0.1")
Base <- Base %>% select(-ID)
attach(Base)
glimpse(Base)
```

```
## Rows: 6,347
## Columns: 12
## $ Customer.Age      <dbl> 67, 67, 55, 63, 57, 58, 57, 46, 56, 56, 53, 56, 57, …
## $ Churn             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ CHI.Score.M0      <dbl> 0, 62, 0, 231, 43, 138, 180, 116, 78, 78, 91, 40, 21…
## $ CHI.Score.0.1     <dbl> 0, 4, 0, 1, -1, -10, -5, -11, -7, -37, -1, 14, 15, 0…
## $ Support.Cases.M0  <dbl> 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ Support.Cases.0.1 <dbl> 0, 0, 0, -1, 0, 0, 1, 0, -2, 0, 0, 0, 0, 0, 0, 0, 0,…
## $ SP.M0             <dbl> 0, 0, 0, 3, 0, 0, 3, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ SP.0.1            <dbl> 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
## $ Logins.0.1        <dbl> 0, 0, 0, 167, 0, 43, 13, 0, -9, -7, 14, 0, 71, 0, 5,…
## $ Blog.Articles.0.1 <dbl> 0, 0, 0, -8, 0, 0, -1, 0, 1, 0, 3, 0, 9, 0, 1, 0, 0,…
## $ Views.0.1         <dbl> 0, -16, 0, 21996, 9, -33, 907, 38, 0, 30, 0, 15, 865…
## $ X.DaysLL.0.1      <dbl> 31, 31, 31, 0, 31, 0, 0, 6, 7, 14, 0, 31, 0, 31, 0, …
```

- Actualizando los tipos de variables

```
Base$Churn<-as.factor(Base$Churn)
levels(Base$Churn)<-c("No","Si")
attach(Base)
table(Churn)
```

```
## Churn
##   No   Si
## 6024  323
```

- Validando la estructura y las estadísticas básicas

```
tibble(Base)
```

```
## # A tibble: 6,347 × 12
##    Customer.Age Churn CHI.Score.M0 CHI.Score.0.1 Support.Cases.M0
##           <dbl> <fct>        <dbl>         <dbl>            <dbl>
##  1           67 No               0             0                0
##  2           67 No              62             4                0
##  3           55 No               0             0                0
##  4           63 No             231             1                1
##  5           57 No              43            -1                0
##  6           58 No             138           -10                0
##  7           57 No             180            -5                1
##  8           46 No             116           -11                0
##  9           56 No              78            -7                1
## 10           56 No              78           -37                0
## # ℹ 6,337 more rows
## # ℹ 7 more variables: Support.Cases.0.1 <dbl>, SP.M0 <dbl>, SP.0.1 <dbl>,
## #   Logins.0.1 <dbl>, Blog.Articles.0.1 <dbl>, Views.0.1 <dbl>,
## #   X.DaysLL.0.1 <dbl>
```

```
df_status(Base)
```

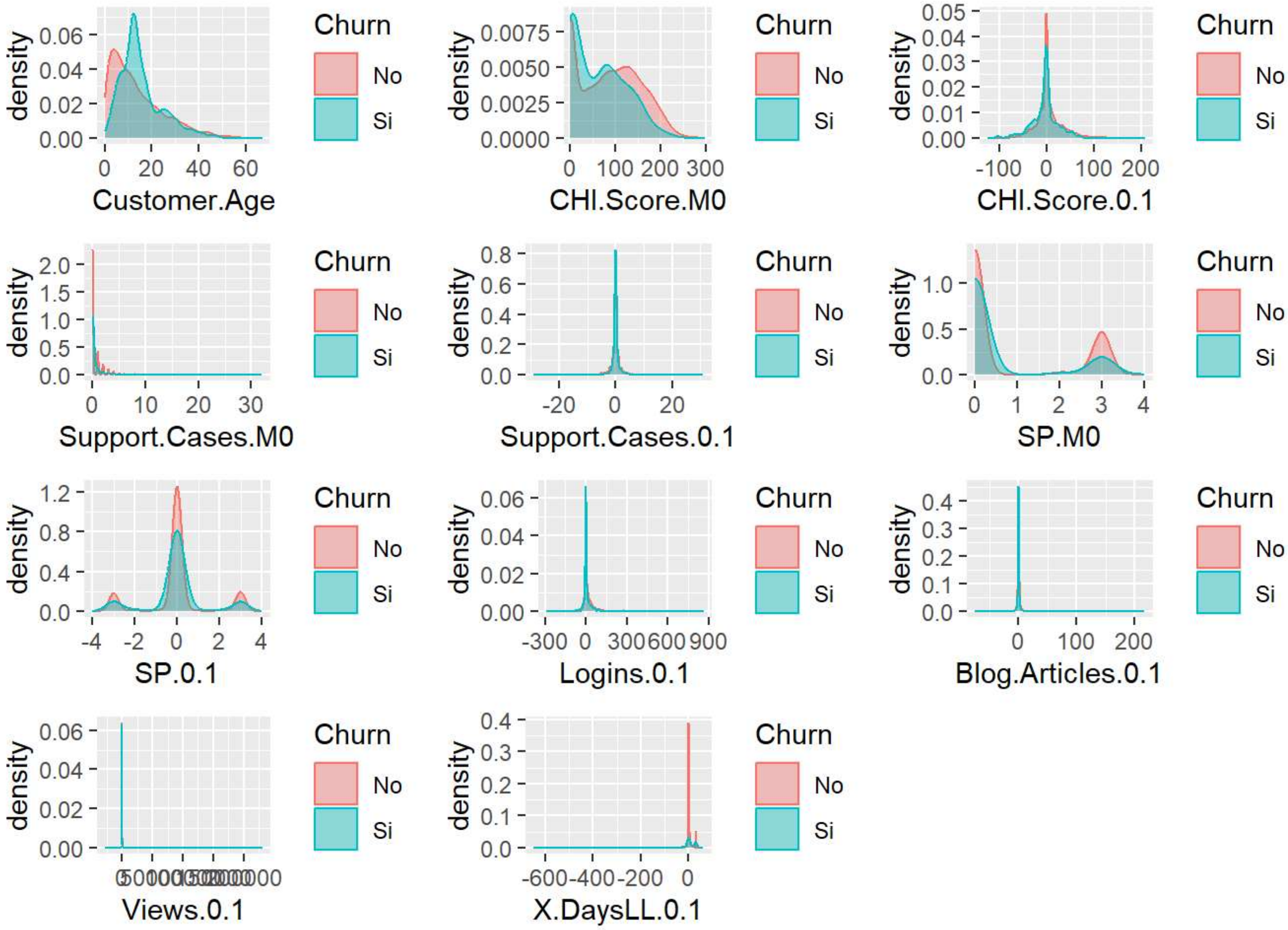
```
##             variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1       Customer.Age       1    0.02    0    0     0     0 numeric     61
## 2              Churn       0    0.00    0    0     0     0  factor      2
## 3       CHI.Score.M0    1193   18.80    0    0     0     0 numeric    263
## 4      CHI.Score.0.1    1408   22.18    0    0     0     0 numeric    242
## 5   Support.Cases.M0    4556   71.78    0    0     0     0 numeric     21
## 6  Support.Cases.0.1    4062   64.00    0    0     0     0 numeric     37
## 7              SP.M0    4588   72.29    0    0     0     0 numeric     27
## 8             SP.0.1    4561   71.86    0    0     0     0 numeric     81
## 9         Logins.0.1    1289   20.31    0    0     0     0 numeric    294
## 10 Blog.Articles.0.1    3632   57.22    0    0     0     0 numeric     57
## 11         Views.0.1    1925   30.33    0    0     0     0 numeric   1360
## 12      X.DaysLL.0.1    2665   41.99    0    0     0     0 numeric    143
```

```
summary(Base)
```

```
##   Customer.Age  Churn      CHI.Score.M0    CHI.Score.0.1      Support.Cases.M0
##  Min.   : 0.0   No:6024   Min.   :  0.00   Min.   :-125.000   Min.   : 0.0000
##  1st Qu.: 5.0   Si: 323   1st Qu.: 24.50   1st Qu.:  -8.000   1st Qu.: 0.0000
##  Median :11.0             Median : 87.00   Median :   0.000   Median : 0.0000
##  Mean   :13.9             Mean   : 87.32   Mean   :   5.059   Mean   : 0.7063
##  3rd Qu.:20.0             3rd Qu.:139.00   3rd Qu.:  15.000   3rd Qu.: 1.0000
##  Max.   :67.0             Max.   :298.00   Max.   : 208.000   Max.   :32.0000
##  Support.Cases.0.1        SP.M0            SP.0.1            Logins.0.1
##  Min.   :-29.000000   Min.   :0.0000   Min.   :-4.00000   Min.   :-293.00
##  1st Qu.:  0.000000   1st Qu.:0.0000   1st Qu.: 0.00000   1st Qu.:  -1.00
##  Median :  0.000000   Median :0.0000   Median : 0.00000   Median :   2.00
##  Mean   : -0.006932   Mean   :0.8128   Mean   : 0.03017   Mean   :  15.73
##  3rd Qu.:  0.000000   3rd Qu.:2.6667   3rd Qu.: 0.00000   3rd Qu.:  23.00
##  Max.   : 31.000000   Max.   :4.0000   Max.   : 4.00000   Max.   : 865.00
##  Blog.Articles.0.1    Views.0.1          X.DaysLL.0.1
##  Min.   :-75.0000   Min.   :-28322.00   Min.   :-648.000
##  1st Qu.:  0.0000   1st Qu.:   -11.00   1st Qu.:   0.000
##  Median :  0.0000   Median :     0.00   Median :   0.000
##  Mean   :  0.1572   Mean   :    96.31   Mean   :   1.765
##  3rd Qu.:  0.0000   3rd Qu.:    27.00   3rd Qu.:   3.000
##  Max.   :217.0000   Max.   :230414.00   Max.   :  61.000
```

1.1 Análisis descriptivo

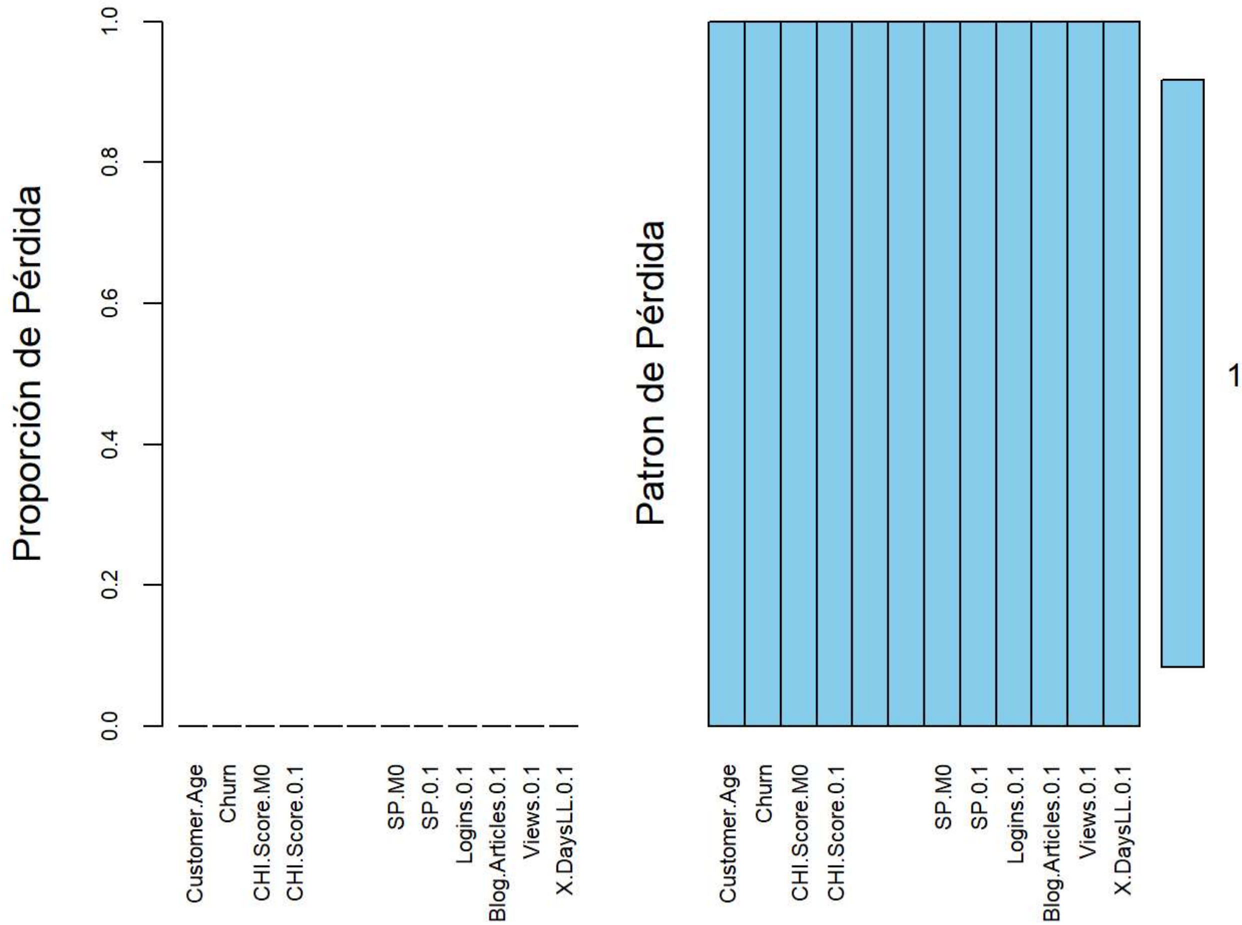
```
Base$Churn=as.factor(Base$Churn)
attach(Base)

p1=ggplot(Base, aes(x=Customer.Age, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p2=ggplot(Base, aes(x=CHI.Score.M0, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p3=ggplot(Base, aes(x=CHI.Score.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p4=ggplot(Base, aes(x=Support.Cases.M0, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p5=ggplot(Base, aes(x=Support.Cases.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p6=ggplot(Base, aes(x=SP.M0, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p7=ggplot(Base, aes(x=SP.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p8=ggplot(Base, aes(x=Logins.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p9=ggplot(Base, aes(x=Blog.Articles.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p10=ggplot(Base, aes(x=Views.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)
p11=ggplot(Base, aes(x=X.DaysLL.0.1, group=Churn, color=Churn, fill=Churn)) +
  geom_density(alpha=0.4)

grid.arrange(p1, p2, p3, p4,p5,p6,p7,p8,p9,p10,p11, ncol=3)
```



2. Análisis de Datos Perdidos

Proporción de valores perdidos

```
aggr(Base , numbers = TRUE,
     sortVars = TRUE,
     cex.axis= 0.7, gap = 3,
     ylab = c("Proporción de Pérdida", "Patron de Pérdida"))
```




```
head(sort((colSums(is.na(Base))/nrow(Base))*100 , decreasing = TRUE ), 10)
```

```
##      Customer.Age             Churn      CHI.Score.M0     CHI.Score.0.1
##                 0                 0                 0                 0
##  Support.Cases.M0 Support.Cases.0.1             SP.M0             SP.0.1
##                 0                 0                 0                 0
##       Logins.0.1 Blog.Articles.0.1
##                 0                 0
```

Valores perdidos por Variable:

```
head(sort( apply(Base, MARGIN=2, function(x){sum(is.na(x))})  , decreasing = TRUE ) , 10)
```

```
##      Customer.Age             Churn      CHI.Score.M0     CHI.Score.0.1
##                 0                 0                 0                 0
##  Support.Cases.M0 Support.Cases.0.1             SP.M0             SP.0.1
##                 0                 0                 0                 0
##       Logins.0.1 Blog.Articles.0.1
##                 0                 0
```

```
table(apply(Base, MARGIN=2, function(x){sum(is.na(x))}) ,  dnn="Frecuencia de Variables con N va
lores faltantes")
```

```
## Frecuencia de Variables con N valores faltantes
##  0
## 12
```

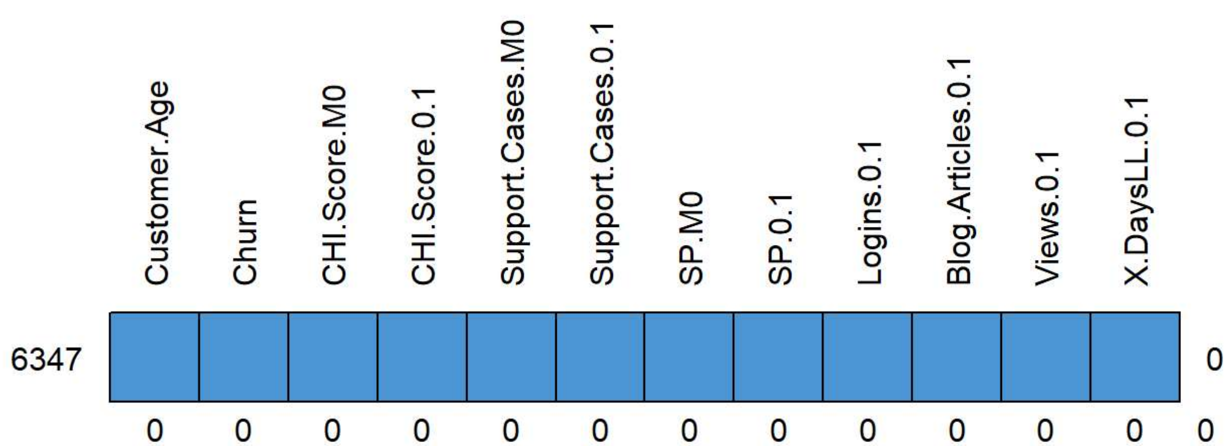
```
table(apply(Base, MARGIN=1 , function(x){sum(is.na(x))}) ,  dnn="Frecuencia de filas con N valor
es faltantes")
```

```
## Frecuencia de filas con N valores faltantes
##    0
## 6347
```

Patron de Pérdida

Nos muestra el patrón de los valores pedidos dándonos el número de filas que poseen "n" valores perdidos y en que variables se encuentran estos datos perdidos.

```
md.pattern(Base, plot = TRUE , rotate.names = TRUE)
```



3. Modelos por segmentos

```
* Clientes con menos de 6 meses de antigüedad
```

```
Base <- read_excel("UV6696-XLS-ENG.xlsx", sheet = "Case Data")
names (Base) = c("ID", "Customer.Age", "Churn", "CHI.Score.M0",
                 "CHI.Score.0.1","Support.Cases.M0", "Support.Cases.0.1",
                 "SP.M0","SP.0.1","Logins.0.1","Blog.Articles.0.1",
                 "Views.0.1","X.DaysLL.0.1")

attach(Base)

Base$Churn<-as.factor(Base$Churn)

Base1<-Base%>%
  filter(Customer.Age<=6)
Base1$Churn<-as.factor(Base1$Churn)
summary(Base1)
```

```
##        ID        Customer.Age  Churn     CHI.Score.M0 CHI.Score.0.1
##  Min.   : 165   Min.   :0.00   0:2007   Min.   :  0   Min.   :-93.00
##  1st Qu.:4810   1st Qu.:2.00   1:  44   1st Qu.:  0   1st Qu.:  0.00
##  Median :5322   Median :3.00            Median : 18   Median :  0.00
##  Mean   :5164   Mean   :3.45            Mean   : 48   Mean   : 16.03
##  3rd Qu.:5834   3rd Qu.:5.00            3rd Qu.: 91   3rd Qu.: 30.00
##  Max.   :6347   Max.   :6.00            Max.   :248   Max.   :208.00
##  Support.Cases.M0  Support.Cases.0.1      SP.M0            SP.0.1
##  Min.   : 0.0000   Min.   :-29.0000   Min.   :0.0000   Min.   :-3.6667
##  1st Qu.: 0.0000   1st Qu.:  0.0000   1st Qu.:0.0000   1st Qu.: 0.0000
##  Median : 0.0000   Median :  0.0000   Median :0.0000   Median : 0.0000
##  Mean   : 0.9371   Mean   :  0.1204   Mean   :0.8726   Mean   : 0.1702
##  3rd Qu.: 1.0000   3rd Qu.:  0.0000   3rd Qu.:3.0000   3rd Qu.: 0.0000
##  Max.   :32.0000   Max.   : 31.0000   Max.   :4.0000   Max.   : 3.5000
##    Logins.0.1      Blog.Articles.0.1    Views.0.1         X.DaysLL.0.1
##  Min.   :-293.00   Min.   :-75.0000   Min.   :-1887.00   Min.   :-648.0000
##  1st Qu.:   0.00   1st Qu.:  0.0000   1st Qu.:    0.00   1st Qu.:   0.0000
##  Median :   0.00   Median :  0.0000   Median :    0.00   Median :   0.0000
##  Mean   :  18.53   Mean   :  0.5768   Mean   :   81.26   Mean   :   0.7991
##  3rd Qu.:  26.50   3rd Qu.:  0.0000   3rd Qu.:    2.00   3rd Qu.:   0.0000
##  Max.   : 496.00   Max.   :217.0000   Max.   :14148.00   Max.   :  46.0000
```

```
modelo2 <- glm(Churn ~ Customer.Age  + CHI.Score.0.1 +
                 Support.Cases.0.1  + Logins.0.1 +
                 X.DaysLL.0.1, data = Base1,
               family = binomial(link='logit'))
summary(modelo2)
```

```
##
## Call:
## glm(formula = Churn ~ Customer.Age + CHI.Score.0.1 + Support.Cases.0.1 +
##     Logins.0.1 + X.DaysLL.0.1, family = binomial(link = "logit"),
##     data = Base1)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -5.292706   0.478507 -11.061  < 2e-16 ***
## Customer.Age       0.356347   0.103332   3.449 0.000564 ***
## CHI.Score.0.1     -0.021474   0.005799  -3.703 0.000213 ***
## Support.Cases.0.1  0.118865   0.060340   1.970 0.048847 *
## Logins.0.1         0.006056   0.002839   2.133 0.032900 *
## X.DaysLL.0.1       0.040409   0.018312   2.207 0.027340 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 425.14  on 2050  degrees of freedom
## Residual deviance: 385.44  on 2045  degrees of freedom
## AIC: 397.44
##
## Number of Fisher Scoring iterations: 7
```

```
# Calculando la matriz de confusión

summary(Base1)
```

```
##        ID        Customer.Age  Churn     CHI.Score.M0 CHI.Score.0.1
##  Min.   : 165   Min.   :0.00   0:2007   Min.   :  0   Min.   :-93.00
##  1st Qu.:4810   1st Qu.:2.00   1:  44   1st Qu.:  0   1st Qu.:  0.00
##  Median :5322   Median :3.00            Median : 18   Median :  0.00
##  Mean   :5164   Mean   :3.45            Mean   : 48   Mean   : 16.03
##  3rd Qu.:5834   3rd Qu.:5.00            3rd Qu.: 91   3rd Qu.: 30.00
##  Max.   :6347   Max.   :6.00            Max.   :248   Max.   :208.00
##  Support.Cases.M0  Support.Cases.0.1      SP.M0            SP.0.1
##  Min.   : 0.0000   Min.   :-29.0000   Min.   :0.0000   Min.   :-3.6667
##  1st Qu.: 0.0000   1st Qu.:  0.0000   1st Qu.:0.0000   1st Qu.: 0.0000
##  Median : 0.0000   Median :  0.0000   Median :0.0000   Median : 0.0000
##  Mean   : 0.9371   Mean   :  0.1204   Mean   :0.8726   Mean   : 0.1702
##  3rd Qu.: 1.0000   3rd Qu.:  0.0000   3rd Qu.:3.0000   3rd Qu.: 0.0000
##  Max.   :32.0000   Max.   : 31.0000   Max.   :4.0000   Max.   : 3.5000
##    Logins.0.1      Blog.Articles.0.1    Views.0.1         X.DaysLL.0.1
##  Min.   :-293.00   Min.   :-75.0000   Min.   :-1887.00   Min.   :-648.0000
##  1st Qu.:   0.00   1st Qu.:  0.0000   1st Qu.:    0.00   1st Qu.:   0.0000
##  Median :   0.00   Median :  0.0000   Median :    0.00   Median :   0.0000
##  Mean   :  18.53   Mean   :  0.5768   Mean   :   81.26   Mean   :   0.7991
##  3rd Qu.:  26.50   3rd Qu.:  0.0000   3rd Qu.:    2.00   3rd Qu.:   0.0000
##  Max.   : 496.00   Max.   :217.0000   Max.   :14148.00   Max.   :  46.0000
```

```
prop.table(table(Base1$Churn))
```

```
## 
##          0          1 
## 0.97854705 0.02145295
```

```
Base1$Churn<-as.factor(Base1$Churn)
pred1 <- predict(modelo2,Base1,type="response")
pred1_Est <- ifelse(pred1>0.02,"1","0")
head(data.frame(Base1,pred1,pred1_Est))
```

```
##    ID Customer.Age Churn CHI.Score.M0 CHI.Score.0.1 Support.Cases.M0
## 1 165            1     0            0             0                0
## 2 244            1     0            0             0                0
## 3 254            1     0            0             0                0
## 4 277            5     0          112             3                0
## 5 315            6     0           20            -1                0
## 6 327            5     0          118             6                0
##   Support.Cases.0.1 SP.M0 SP.0.1 Logins.0.1 Blog.Articles.0.1 Views.0.1
## 1                 0     0      0          0                 0         0
## 2                 0     0      0          0                 0         0
## 3                 0     0      0          0                 0         0
## 4                -2     0     -3         36                -2      1490
## 5                 0     0      0         -2                 0        15
## 6                -1     0     -3        -29                 5       191
##   X.DaysLL.0.1       pred1 pred1_Est
## 1            0 0.007129501         0
## 2            0 0.007129501         0
## 3            0 0.007129501         0
## 4            0 0.026724299         1
## 5            4 0.048170942         1
## 6           -5 0.015729521         0
```

```
xtab1 <- table(Base1$Churn,pred1_Est)   # Matriz de Confusi?n
xtab1
```

```
##    pred1_Est
##        0    1
##   0 1403  604
##   1   12   32
```

```
acc_1 <- confusionMatrix(xtab1,positive = "1")
acc_1
```

```
## Confusion Matrix and Statistics
##
##    pred1_Est
##        0    1
##   0 1403  604
##   1   12   32
##
##                Accuracy : 0.6997
##                  95% CI : (0.6793, 0.7194)
##     No Information Rate : 0.6899
##     P-Value [Acc > NIR] : 0.1761
##
##                   Kappa : 0.0562
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.05031
##             Specificity : 0.99152
##          Pos Pred Value : 0.72727
##          Neg Pred Value : 0.69905
##              Prevalence : 0.31009
##          Detection Rate : 0.01560
##    Detection Prevalence : 0.02145
##       Balanced Accuracy : 0.52092
##
##        'Positive' Class : 1
##
```

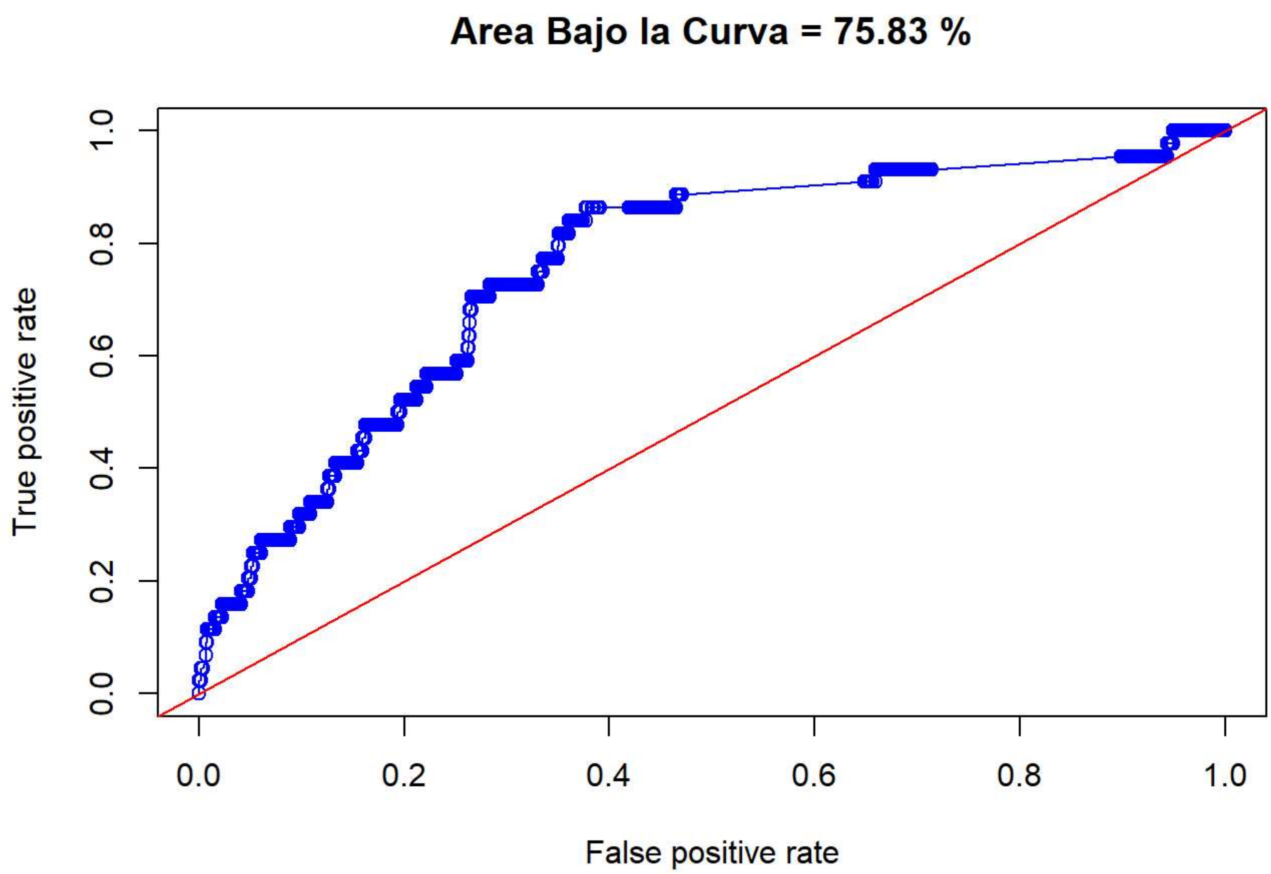
```
pred_1 <- prediction(pred1,Base1$Churn)
perf_1=ROCR::performance(pred_1,"tpr","fpr")
auc <- 100*as.numeric(ROCR::performance(pred_1 ,"auc")@y.values)
auc
```

```
## [1] 75.83005
```

```
gini <- 2*(auc-50)
gini
```

```
## [1] 51.6601
```

```
plot(perf_1,type='o', main = paste('Area Bajo la Curva =',round(auc,2),'%'),col="blue")
abline(a=0, b= 1,col="red")
```




```
* Clientes con antigüedad entre 7 y 13 meses
```

```
Base2<-Base%>%
  filter(Customer.Age>=7 & Customer.Age<=13)

Base2 <- Base2 %>%
  mutate(FlagAge12 = case_when(Base2$Customer.Age==12 ~ 1,
                               TRUE ~ 0))
Base2$FlagAge12 <- as.factor(Base2$FlagAge12)

attach(Base2)
summary(Base2)
```

```
##        ID        Customer.Age    Churn     CHI.Score.M0   CHI.Score.0.1
##  Min.   :  66   Min.   : 7.000   0:1565   Min.   :  0.0   Min.   :-116.00000
##  1st Qu.:2078   1st Qu.: 8.000   1: 131   1st Qu.: 54.0   1st Qu.: -12.00000
##  Median :3526   Median :10.000            Median :106.0   Median :   0.00000
##  Mean   :2977   Mean   : 9.929            Mean   :101.4   Mean   :   0.06014
##  3rd Qu.:3999   3rd Qu.:12.000            3rd Qu.:147.0   3rd Qu.:  12.00000
##  Max.   :4651   Max.   :13.000            Max.   :279.0   Max.   : 114.00000
##  Support.Cases.M0  Support.Cases.0.1      SP.M0            SP.0.1
##  Min.   : 0.0000   Min.   :-17.0000   Min.   :0.0000   Min.   :-4.0000
##  1st Qu.: 0.0000   1st Qu.: -0.2500   1st Qu.:0.0000   1st Qu.: 0.0000
##  Median : 0.0000   Median :  0.0000   Median :0.0000   Median : 0.0000
##  Mean   : 0.6946   Mean   : -0.1834   Mean   :0.8929   Mean   :-0.1402
##  3rd Qu.: 1.0000   3rd Qu.:  0.0000   3rd Qu.:3.0000   3rd Qu.: 0.0000
##  Max.   :18.0000   Max.   : 13.0000   Max.   :4.0000   Max.   : 4.0000
##    Logins.0.1      Blog.Articles.0.1     Views.0.1        X.DaysLL.0.1
##  Min.   :-115.00   Min.   :-28.00000   Min.   :-28322.0   Min.   :-127.00
##  1st Qu.:  -3.00   1st Qu.: -1.00000   1st Qu.:   -38.0   1st Qu.:  -1.00
##  Median :   4.00   Median :  0.00000   Median :     0.0   Median :   0.00
##  Mean   :  15.87   Mean   : -0.07075   Mean   :   173.3   Mean   :   1.98
##  3rd Qu.:  25.25   3rd Qu.:  0.00000   3rd Qu.:    45.0   3rd Qu.:   4.00
##  Max.   : 513.00   Max.   : 44.00000   Max.   :230414.0   Max.   :  31.00
##  FlagAge12
##  0:1425
##  1: 271
##
##
##
##
```

```
modelo3 <- glm(Churn ~ CHI.Score.M0 +
                 SP.M0  + Views.0.1+
                 X.DaysLL.0.1+FlagAge12, data = Base2,
               family = binomial(link='logit'))
summary(modelo3)
```

```
##
## Call:
## glm(formula = Churn ~ CHI.Score.M0 + SP.M0 + Views.0.1 + X.DaysLL.0.1 +
##     FlagAge12, family = binomial(link = "logit"), data = Base2)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.062e+00  1.946e-01 -10.599  < 2e-16 ***
## CHI.Score.M0 -9.814e-03  1.952e-03  -5.028 4.95e-07 ***
## SP.M0        -2.086e-01  9.737e-02  -2.142  0.03217 *
## Views.0.1    -1.245e-04  5.323e-05  -2.339  0.01935 *
## X.DaysLL.0.1  1.820e-02  6.382e-03   2.852  0.00434 **
## FlagAge121    1.572e+00  2.013e-01   7.808 5.83e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 922.55  on 1695  degrees of freedom
## Residual deviance: 782.98  on 1690  degrees of freedom
## AIC: 794.98
##
## Number of Fisher Scoring iterations: 6
```

```
# Calculando la matriz de confusión


prop.table(table(Base2$Churn))
```

```
##
##          0          1
## 0.92275943 0.07724057
```

```
Base2$Churn<-as.factor(Base2$Churn)
pred1 <- predict(modelo3,Base2,type="response")
pred1_Est <- ifelse(pred1>0.07,"1","0")
head(data.frame(Base2,pred1,pred1_Est))
```

```
##    ID Customer.Age Churn CHI.Score.M0 CHI.Score.0.1 Support.Cases.M0
## 1  66           13     1          122            31                6
## 2 206           13     0          121           -15                0
## 3 207           13     0            3             0                2
## 4 208            7     0          141            22                2
## 5 211           12     1          143            37                0
## 6 216           13     0          118            -1                1
##   Support.Cases.0.1 SP.M0 SP.0.1 Logins.0.1 Blog.Articles.0.1 Views.0.1
## 1                 4     3      0         94                 4        -1
## 2                 0     0      0          1                -1       266
## 3                 1     3      3         19                 0         0
## 4                 1     3      0         70                 0        72
## 5                 0     0      0         50                 3        18
## 6                 1     3      3         41                -2         5
##   X.DaysLL.0.1 FlagAge12      pred1 pred1_Est
## 1           -9         0 0.01714223         0
## 2            0         0 0.03616757         0
## 3          -16         0 0.04703784         0
## 4           -3         0 0.01574687         0
## 5           28         1 0.20002677         1
## 6            0         0 0.02090650         0
```

```
xtab1 <- table(Base2$Churn,pred1_Est)   # Matriz de Confusi?n
xtab1
```

```
##    pred1_Est
##        0    1
##   0 1087  478
##   1   39   92
```

```
acc_1 <- confusionMatrix(xtab1,positive = "1")
acc_1
```

```
## Confusion Matrix and Statistics
## 
##    pred1_Est
##        0    1
##   0 1087  478
##   1   39   92
##                                           
##                Accuracy : 0.6952          
##                  95% CI : (0.6726, 0.717) 
##     No Information Rate : 0.6639          
##     P-Value [Acc > NIR] : 0.003287        
##                                           
##                   Kappa : 0.1565          
##                                           
##  Mcnemar's Test P-Value : < 2.2e-16       
##                                           
##             Sensitivity : 0.16140         
##             Specificity : 0.96536         
##          Pos Pred Value : 0.70229         
##          Neg Pred Value : 0.69457         
##              Prevalence : 0.33608         
##          Detection Rate : 0.05425         
##    Detection Prevalence : 0.07724         
##       Balanced Accuracy : 0.56338         
##                                           
##        'Positive' Class : 1               
## 
```

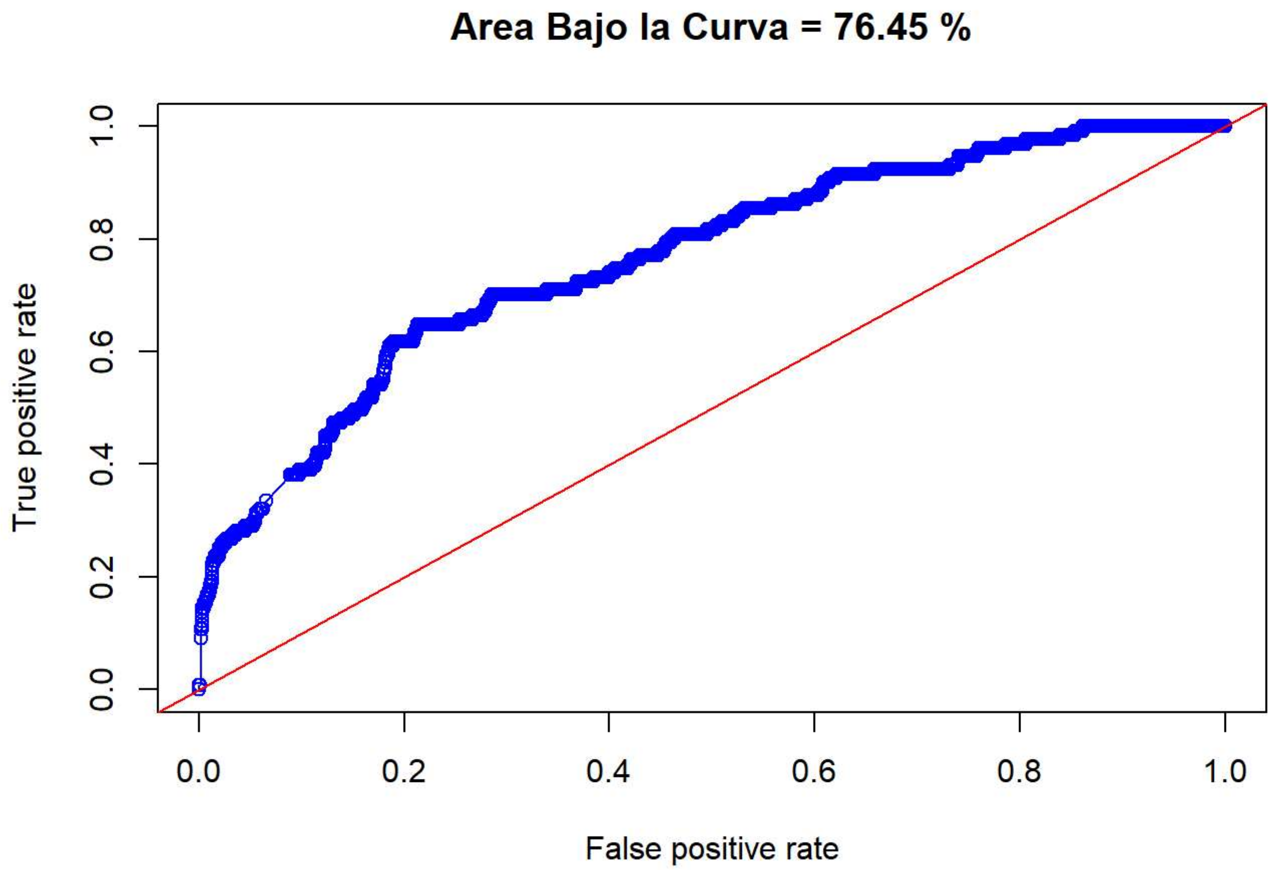
```
pred_1 <- prediction(pred1,Base2$Churn)
perf_1=ROCR::performance(pred_1,"tpr","fpr")

auc <- 100*as.numeric(ROCR::performance(pred_1 ,"auc")@y.values)
auc
```

```
## [1] 76.44514
```

```
gini <- 2*(auc-50)
gini
```

```
## [1] 52.89028
```

```
plot(perf_1,type='o', main = paste('Area Bajo la Curva =',round(auc,2),'%'),col="blue")
abline(a=0, b= 1,col="red")
```



4. Clientes con más de 13 meses de antigüedad

```
Base3<-Base%>%
  filter(Customer.Age>13)



attach(Base3)
summary(Base3)
```

```
##        ID        Customer.Age   Churn     CHI.Score.M0   CHI.Score.0.1
##  Min.   :   1   Min.   :14.00   0:2452   Min.   :  0.0   Min.   :-125.0000
##  1st Qu.:1073   1st Qu.:17.00   1: 148   1st Qu.: 66.0   1st Qu.: -11.0000
##  Median :1770   Median :22.00            Median :110.0   Median :   0.0000
##  Mean   :1732   Mean   :24.73            Mean   :109.2   Mean   :  -0.3365
##  3rd Qu.:2457   3rd Qu.:30.00            3rd Qu.:154.0   3rd Qu.:  10.0000
##  Max.   :4499   Max.   :67.00            Max.   :298.0   Max.   : 154.0000
##  Support.Cases.M0  Support.Cases.0.1        SP.M0            SP.0.1
##  Min.   : 0.0000   Min.   :-10.000000   Min.   :0.0000   Min.   :-4.00000
##  1st Qu.: 0.0000   1st Qu.:  0.000000   1st Qu.:0.0000   1st Qu.: 0.00000
##  Median : 0.0000   Median :  0.000000   Median :0.0000   Median : 0.00000
##  Mean   : 0.5319   Mean   :  0.007692   Mean   :0.7133   Mean   : 0.03088
##  3rd Qu.: 0.0000   3rd Qu.:  0.000000   3rd Qu.:0.0000   3rd Qu.: 0.00000
##  Max.   :24.0000   Max.   : 14.000000   Max.   :4.0000   Max.   : 4.00000
##    Logins.0.1      Blog.Articles.0.1   Views.0.1          X.DaysLL.0.1
##  Min.   :-115.00   Min.   :-49.000   Min.   :-27768.00   Min.   :-190.000
##  1st Qu.:  -2.00   1st Qu.:  0.000   1st Qu.:   -56.25   1st Qu.:  -1.000
##  Median :   4.00   Median :  0.000   Median :     0.00   Median :   0.000
##  Mean   :  13.43   Mean   : -0.025   Mean   :    57.97   Mean   :   2.386
##  3rd Qu.:  21.00   3rd Qu.:  0.000   3rd Qu.:    49.00   3rd Qu.:   6.000
##  Max.   : 865.00   Max.   :118.000   Max.   : 21996.00   Max.   :  61.000
```

```
modelo4 <- glm(Churn ~ Customer.Age+CHI.Score.M0, data = Base3,
               family = binomial(link='logit'))
summary(modelo4)
```

```
##
## Call:
## glm(formula = Churn ~ Customer.Age + CHI.Score.M0, family = binomial(link = "logit"),
##     data = Base3)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.848857   0.278434  -3.049  0.00230 **
## Customer.Age -0.033836   0.010377  -3.261  0.00111 **
## CHI.Score.M0 -0.012963   0.001565  -8.284  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1135.8  on 2599  degrees of freedom
## Residual deviance: 1049.4  on 2597  degrees of freedom
## AIC: 1055.4
##
## Number of Fisher Scoring iterations: 6
```

```
# Calculando la matriz de confusión


prop.table(table(Base3$Churn))
```

```
##
##          0          1
## 0.94307692 0.05692308
```

```
Base3$Churn<-as.factor(Base3$Churn)
pred1 <- predict(modelo4,Base3,type="response")
pred1_Est <- ifelse(pred1>0.056,"1","0")
head(data.frame(Base3,pred1,pred1_Est))
```

```
##   ID Customer.Age Churn CHI.Score.M0 CHI.Score.0.1 Support.Cases.M0
## 1  1           67     0            0             0                0
## 2  2           67     0           62             4                0
## 3  3           55     0            0             0                0
## 4  4           63     0          231             1                1
## 5  5           57     0           43            -1                0
## 6  6           58     0          138           -10                0
##   Support.Cases.0.1 SP.M0 SP.0.1 Logins.0.1 Blog.Articles.0.1 Views.0.1
## 1                 0     0      0          0                 0         0
## 2                 0     0      0          0                 0       -16
## 3                 0     0      0          0                 0         0
## 4                -1     3      0        167                -8     21996
## 5                 0     0      0          0                 0         9
## 6                 0     0      0         43                 0       -33
##   X.DaysLL.0.1       pred1 pred1_Est
## 1           31 0.042457844         0
## 2           31 0.019463413         0
## 3           31 0.062395911         1
## 4            0 0.002535146         0
## 5           31 0.034392852         0
## 6            0 0.009949552         0
```

```
xtab1 <- table(Base3$Churn,pred1_Est)   # Matriz de Confusi?n
xtab1
```

```
##    pred1_Est
##        0    1
##   0 1603  849
##   1   49   99
```

```
acc_1 <- confusionMatrix(xtab1,positive = "1")
acc_1
```

```
## Confusion Matrix and Statistics
##
##    pred1_Est
##        0    1
##   0 1603  849
##   1   49   99
##
##                Accuracy : 0.6546
##                  95% CI : (0.636, 0.6729)
##     No Information Rate : 0.6354
##     P-Value [Acc > NIR] : 0.02155
##
##                   Kappa : 0.0912
##
##  Mcnemar's Test P-Value : < 2e-16
##
##             Sensitivity : 0.10443
##             Specificity : 0.97034
##          Pos Pred Value : 0.66892
##          Neg Pred Value : 0.65375
##              Prevalence : 0.36462
##          Detection Rate : 0.03808
##    Detection Prevalence : 0.05692
##       Balanced Accuracy : 0.53738
##
##        'Positive' Class : 1
##
```

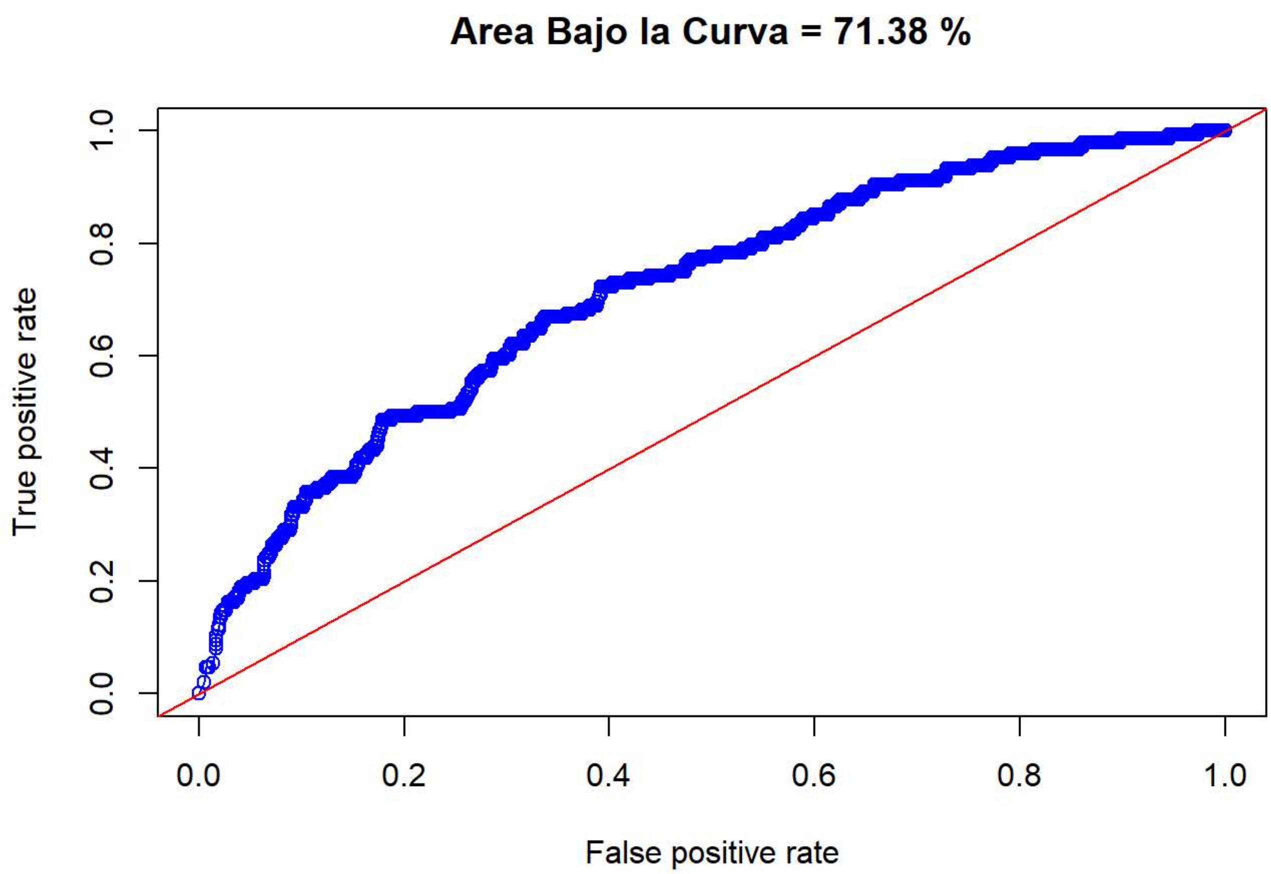
```
pred_1 <- prediction(pred1,Base3$Churn)
perf_1=ROCR::performance(pred_1,"tpr","fpr")

auc <- 100*as.numeric(ROCR::performance(pred_1 ,"auc")@y.values)
auc
```

```
## [1] 71.38133
```

```
gini <- 2*(auc-50)
gini
```

```
## [1] 42.76266
```

```
plot(perf_1,type='o', main = paste('Area Bajo la Curva =',round(auc,2),'%'),col="blue")
abline(a=0, b= 1,col="red")
```



1. División entre entrenamiento y prueba

```
split =0.8
trainIndex <- createDataPartition(Base3$Churn, p = split, list = FALSE, times = 1)
train <- Base3[trainIndex, ]
test <- Base3[-trainIndex, ]
table(train$Churn)
```

```
##
##    0    1
## 1962  119
```

```
prop.table(table(train$Churn))
```

```
##
##          0          1
## 0.94281595 0.05718405
```

```
table(test$Churn)
```

```
##
##   0   1
## 490  29
```

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

```
##
##          0          1
## 0.94412331 0.05587669
```

2. Balanceo de datos para entrenamiento y testeo

```
# Balanceo de datos de entrenamiento
train_bal <- ROSE(Churn ~ ., data = train, seed = 2021)$data

# Balanceo de datos de testeo
test_bal <- ROSE(Churn ~ ., data = test, seed = 2021)$data

table(train_bal$Churn)
```

```
##
##    0    1
## 1061 1020
```

```
prop.table(table(train_bal$Churn))
```

```
##
##        0        1
## 0.509851 0.490149
```

```
table(test_bal$Churn)
```

```
##
##   0   1
## 272 247
```

```
prop.table(table(test_bal$Churn))
```

```
##
##         0         1
## 0.5240848 0.4759152
```

3. Selección de variables significativas

```
levels(train_bal$Churn)<- c("No_Churn", "Churn")


fitControl = trainControl(method = "cv",number = 4,classProbs = TRUE, summaryFunction = twoClass
Summary)

fit_rf = train(x=select(train_bal, -Churn) , y = train_bal$Churn , method = "rf",
               trControl = fitControl,verbose = FALSE, metric = "ROC")

fit_gbm = train(x=select(train_bal, -Churn),y = train_bal$Churn ,
                method = "gbm", trControl = fitControl, verbose = FALSE, metric = "ROC")
```

```
var_imp_rf=data.frame(varImp(fit_rf, scale=T)["importance"]) %>%
  dplyr::mutate(variable=rownames(.)) %>% dplyr::rename(importance_rf=Overall) %>%
  dplyr::arrange(-importance_rf) %>%
  dplyr::mutate(rank_rf=seq(1:nrow(.)))

var_imp_gbm=as.data.frame(varImp(fit_gbm, scale=T)["importance"])  %>%
  dplyr::mutate(variable=rownames(.)) %>% dplyr::rename(importance_gbm=Overall) %>%
  dplyr::arrange(-importance_gbm) %>%
  dplyr::mutate(rank_gbm=seq(1:nrow(.)))
final_res=merge(var_imp_rf, var_imp_gbm, by="variable")

kable(final_res)
```

variable	importance_rf	rank_rf	importance_gbm	rank_gbm
Blog.Articles.0.1	100.0000000	1	100.000000	1
CHI.Score.0.1	4.0486224	9	4.791065	8
CHI.Score.M0	31.4978263	3	26.085251	3
Customer.Age	12.9611581	6	15.415995	5
ID	0.9501361	11	0.000000	12
Logins.0.1	12.8232129	7	2.937648	10
SP.0.1	0.0000000	12	2.111910	11
SP.M0	6.2806854	8	9.086306	7
Support.Cases.0.1	15.0679271	5	12.485000	6
Support.Cases.M0	37.4246987	2	32.489307	2
Views.0.1	3.8555241	10	3.134809	9
X.DaysLL.0.1	24.4797209	4	16.576158	4

```
modelo <- glm(Churn ~ Customer.Age + CHI.Score.M0 + CHI.Score.0.1 +
                Support.Cases.M0 + Support.Cases.0.1 + SP.M0 +
                SP.0.1 + Logins.0.1 + Blog.Articles.0.1 +
                Views.0.1 + X.DaysLL.0.1, data = train_bal,
              family = binomial(link='logit'))
summary(modelo)
```

```
##
## Call:
## glm(formula = Churn ~ Customer.Age + CHI.Score.M0 + CHI.Score.0.1 +
##     Support.Cases.M0 + Support.Cases.0.1 + SP.M0 + SP.0.1 + Logins.0.1 +
##     Blog.Articles.0.1 + Views.0.1 + X.DaysLL.0.1, family = binomial(link = "logit"),
##     data = train_bal)
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        1.555e+00  1.444e-01  10.766  < 2e-16 ***
## Customer.Age      -2.977e-02  4.647e-03  -6.408 1.48e-10 ***
## CHI.Score.M0      -8.363e-03  7.484e-04 -11.175  < 2e-16 ***
## CHI.Score.0.1      1.823e-03  1.690e-03   1.079  0.28077
## Support.Cases.M0   7.766e-03  4.409e-02   0.176  0.86019
## Support.Cases.0.1  1.152e-02  3.933e-02   0.293  0.76964
## SP.M0             -9.739e-02  4.390e-02  -2.218  0.02655 *
## SP.0.1             2.914e-02  3.709e-02   0.785  0.43218
## Logins.0.1        -4.661e-03  1.632e-03  -2.857  0.00428 **
## Blog.Articles.0.1 -6.575e-03  1.348e-02  -0.488  0.62565
## Views.0.1         -2.721e-05  3.462e-05  -0.786  0.43189
## X.DaysLL.0.1      -1.972e-03  1.771e-03  -1.113  0.26557
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2884.1  on 2080  degrees of freedom
## Residual deviance: 2627.2  on 2069  degrees of freedom
## AIC: 2651.2
##
## Number of Fisher Scoring iterations: 4
```

```
var_selected <- c("Customer.Age", "CHI.Score.M0","SP.M0", "Logins.0.1", "Churn")
data_final <- select(train_bal, var_selected)
test_final <- select(test_bal, var_selected)
```

Ajuste del Modelo

- Regresion Logistica

Utilizaremos el modelo de Regresion Logistica ya que contempla predictores numericos como categoricos y la respuesta es binaria, el Riesgo o No Riesgo Cardiaco.

```
logit <- glm(Churn ~., data = data_final , family = binomial(link = "logit"))
summary(logit)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = binomial(link = "logit"), data = data_final)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.5188209  0.1399455  10.853  < 2e-16 ***
## Customer.Age -0.0295990  0.0046048  -6.428 1.29e-10 ***
## CHI.Score.M0 -0.0082405  0.0007176 -11.484  < 2e-16 ***
## SP.M0        -0.0736410  0.0364282  -2.022  0.04322 *
## Logins.0.1   -0.0042332  0.0015695  -2.697  0.00699 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2884.1  on 2080  degrees of freedom
## Residual deviance: 2631.8  on 2076  degrees of freedom
## AIC: 2641.8
##
## Number of Fisher Scoring iterations: 4
```

4. Prediciones *Predicción Data de Entrenamiento*

```
train_pred <- predict(logit, newdata = train_bal, type = "response")
head(train_pred)
```

```
##         1         2         3         4         5         6
## 0.5848182 0.7109686 0.5485075 0.1957339 0.2984064 0.4593118
```

```
train_pred_clase <- factor(ifelse(train_pred > 0.5, 1, 0))
levels(train_pred_clase) <- c("No_Churn", "Churn")

head(train_pred_clase)
```

```
##        1        2        3        4        5        6
##    Churn    Churn    Churn No_Churn No_Churn No_Churn
## Levels: No_Churn Churn
```

```
summary(train_pred_clase)
```

```
## No_Churn    Churn
##     1082      999
```

Predicción para data de testeo

```
test_pred <- predict(logit, newdata = test_bal, type = "response")
head(test_pred)
```

```
##         1         2         3         4         5         6
## 0.6335774 0.5753934 0.7228761 0.4155544 0.6734587 0.6159108
```

```
test_pred_clase <- factor(ifelse(test_pred > 0.5, 1, 0))
levels(test_pred_clase) <- c("No_Churn", "Churn")

head(test_pred_clase)
```

```
##        1        2        3        4        5        6
##    Churn    Churn    Churn No_Churn    Churn    Churn
## Levels: No_Churn Churn
```

```
summary(test_pred_clase)
```

```
## No_Churn    Churn
##      258      261
```

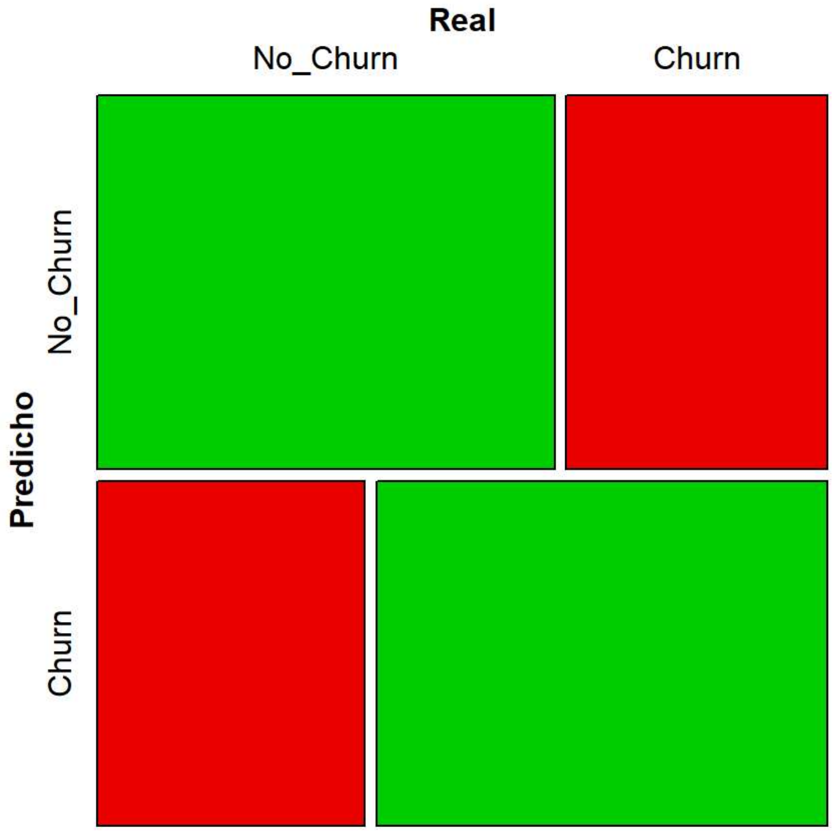
Matriz de confusion

- Para data de entrenamiento

```
data_final$Churn <- as.factor(data_final$Churn)
levels(data_final$Churn) <- c("No_Churn", "Churn")

matriz_confusion_train <- table(Predicho = train_pred_clase, Real = data_final$Churn)
matriz_confusion_train
```

```
##           Real
## Predicho   No_Churn Churn
##   No_Churn      689   393
##   Churn         372   627
```

```
mosaic(matriz_confusion_train, shade = T, colorize = T,
       gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2, 2)))
```

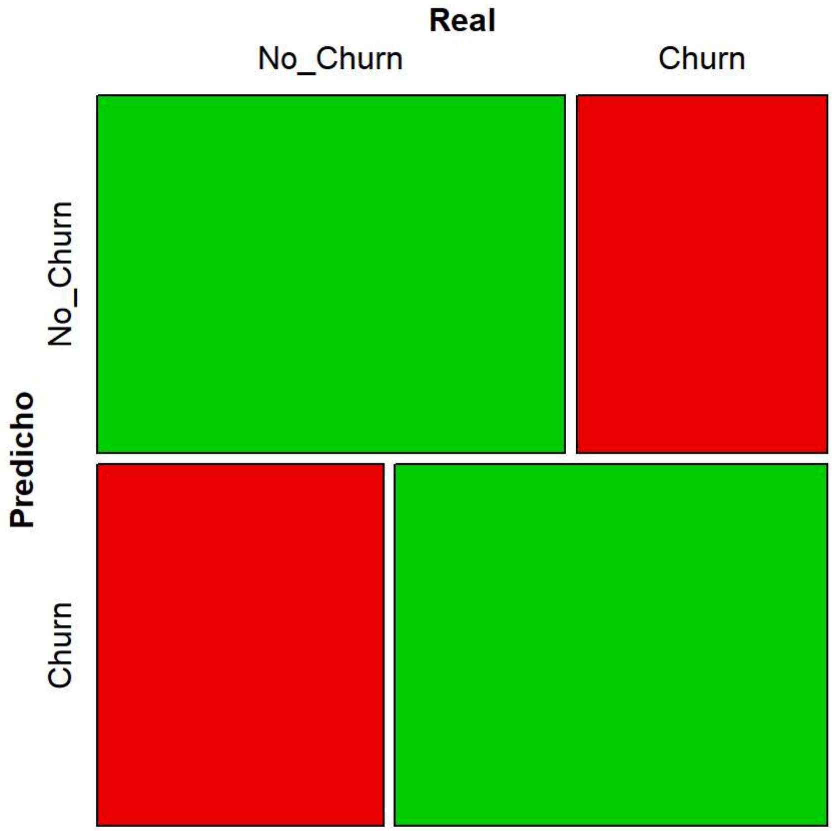


- Para la data de testeo

```
test_final$Churn <- as.factor(test_final$Churn)
levels(test_final$Churn) <- c("No_Churn", "Churn")

matriz_confusion_test <- table(Predicho = test_pred_clase, Real = test_final$Churn)
matriz_confusion_test
```

```
##           Real
## Predicho   No_Churn Churn
##   No_Churn      168    90
##   Churn         104   157
```

```
mosaic(matriz_confusion_test, shade = T, colorize = T,
       gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2, 2)))
```



5. Métricas de evaluación

- Entrenamiento

```
confusionMatrix(train_pred_clase, data_final$Churn)
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction No_Churn Churn
##   No_Churn      689   393
##   Churn         372   627
##                                           
##                Accuracy : 0.6324          
##                  95% CI : (0.6113, 0.6531)
##     No Information Rate : 0.5099          
##     P-Value [Acc > NIR] : <2e-16          
##                                           
##                   Kappa : 0.2642          
##                                           
##  Mcnemar's Test P-Value : 0.4696          
##                                           
##             Sensitivity : 0.6494          
##             Specificity : 0.6147          
##          Pos Pred Value : 0.6368          
##          Neg Pred Value : 0.6276          
##              Prevalence : 0.5099          
##          Detection Rate : 0.3311          
##    Detection Prevalence : 0.5199          
##       Balanced Accuracy : 0.6320          
##                                           
##        'Positive' Class : No_Churn        
## 
```

- Testeo

```r
library(caret)
confusionMatrix(test_pred_clase, test_final$Churn)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction No_Churn Churn
##   No_Churn      168    90
##   Churn         104   157
##
##                Accuracy : 0.6262
##                  95% CI : (0.583, 0.668)
##     No Information Rate : 0.5241
##     P-Value [Acc > NIR] : 1.706e-06
##
##                   Kappa : 0.2526
##
##  Mcnemar's Test P-Value : 0.3506
##
##             Sensitivity : 0.6176
##             Specificity : 0.6356
##          Pos Pred Value : 0.6512
##          Neg Pred Value : 0.6015
##              Prevalence : 0.5241
##          Detection Rate : 0.3237
##    Detection Prevalence : 0.4971
##       Balanced Accuracy : 0.6266
##
##        'Positive' Class : No_Churn
##
```

AUC

- Datos de Entrenamiento

```
pred_train <- prediction(train_pred, data_final$Churn)

AUC_train <- ROCR::performance(pred_train ,measure="auc")

AUCaltura_train <- AUC_train@y.values
cat("AUC:", AUCaltura_train[[1]])
```

```
## AUC: 0.3098085
```

- Datos de Testeo

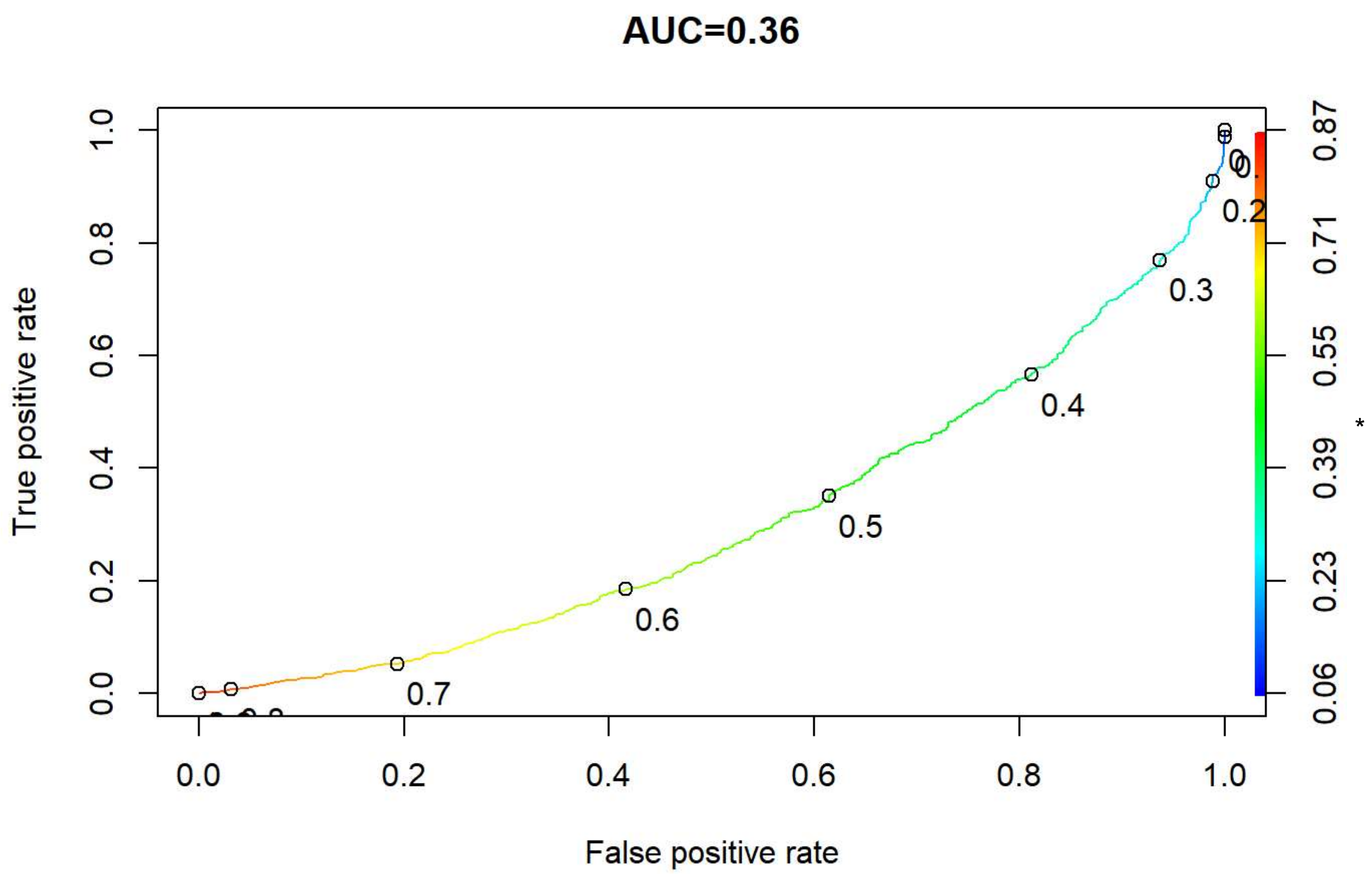
```
pred_test <- prediction(test_pred, test_final$Churn)

AUC_test <- ROCR::performance(pred_test ,measure="auc")

AUCaltura_test <- AUC_test@y.values
cat("AUC:", AUCaltura_test[[1]])
```

```
## AUC: 0.3152239
```

ROC

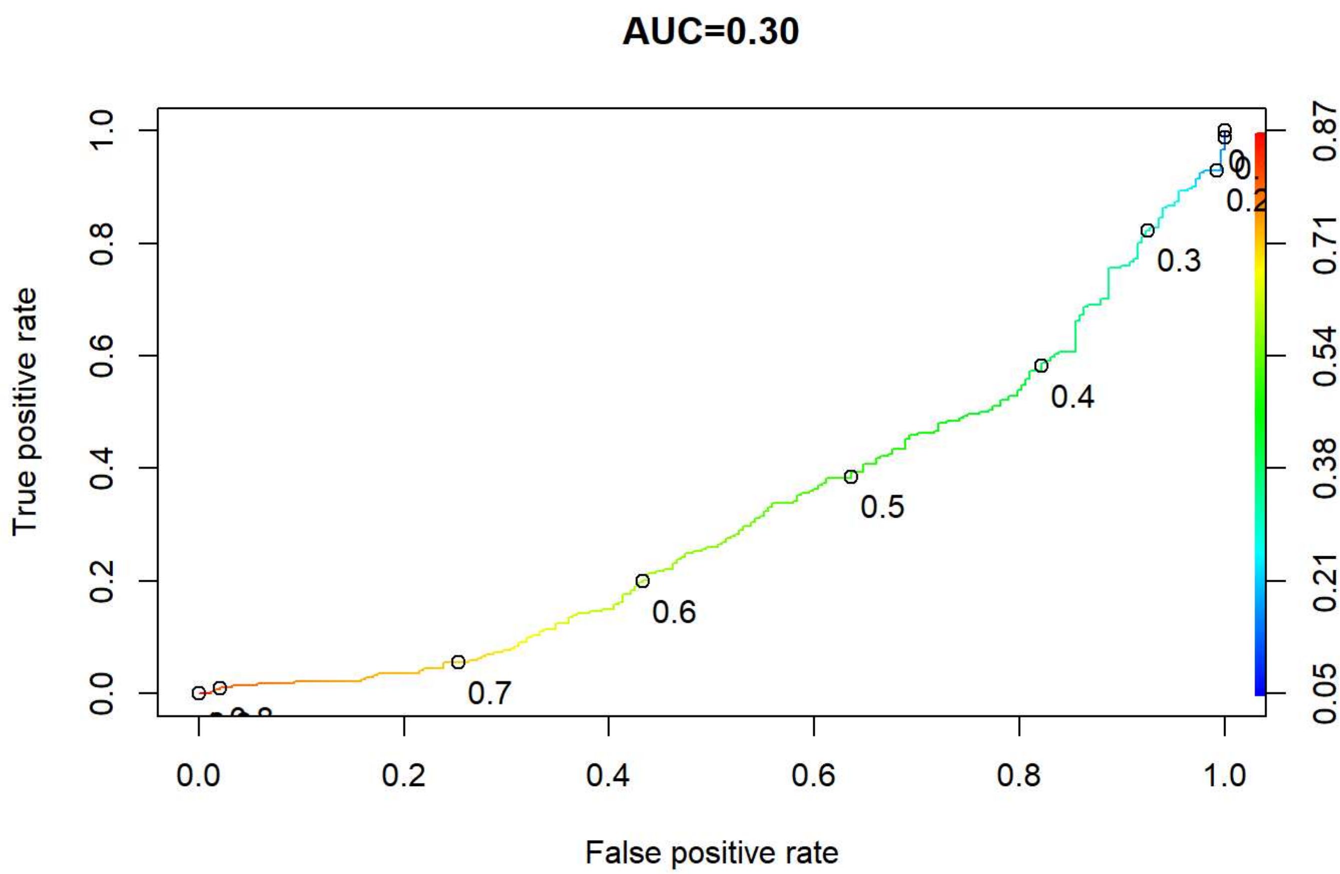
- Entrenamiento

```
ROCRpred1 <- prediction(train_pred, data_final$Churn)
ROCRperf1 <- ROCR::performance(ROCRpred1, "tpr", "fpr")
plot(ROCRperf1 , main="AUC=0.36" ,  colorize = TRUE, print.cutoffs.at = seq(0, 1, by = 0.1), tex
t.adj =c(-0.2,1.7))
```



Testeo

```
ROCRpred2 <- prediction(test_pred, test_final$Churn)
ROCRperf2 <- ROCR::performance(ROCRpred2 , "tpr", "fpr")
plot(ROCRperf2 , main="AUC=0.30" ,  colorize = TRUE, print.cutoffs.at = seq(0, 1, by = 0.1), tex
t.adj =c(-0.2,1.7))
```



- Modelo: Solo variables significativas

```
modelo1 <- glm(Churn ~ Customer.Age + CHI.Score.M0 +
                 SP.M0 + Logins.0.1, data = data_final,
               family = binomial(link='logit'))
summary(modelo1)
```

```
##
## Call:
## glm(formula = Churn ~ Customer.Age + CHI.Score.M0 + SP.M0 + Logins.0.1,
##     family = binomial(link = "logit"), data = data_final)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.5188209  0.1399455  10.853  < 2e-16 ***
## Customer.Age -0.0295990  0.0046048  -6.428 1.29e-10 ***
## CHI.Score.M0 -0.0082405  0.0007176 -11.484  < 2e-16 ***
## SP.M0        -0.0736410  0.0364282  -2.022  0.04322 *
## Logins.0.1   -0.0042332  0.0015695  -2.697  0.00699 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2884.1  on 2080  degrees of freedom
## Residual deviance: 2631.8  on 2076  degrees of freedom
## AIC: 2641.8
##
## Number of Fisher Scoring iterations: 4
```

- Proporción target

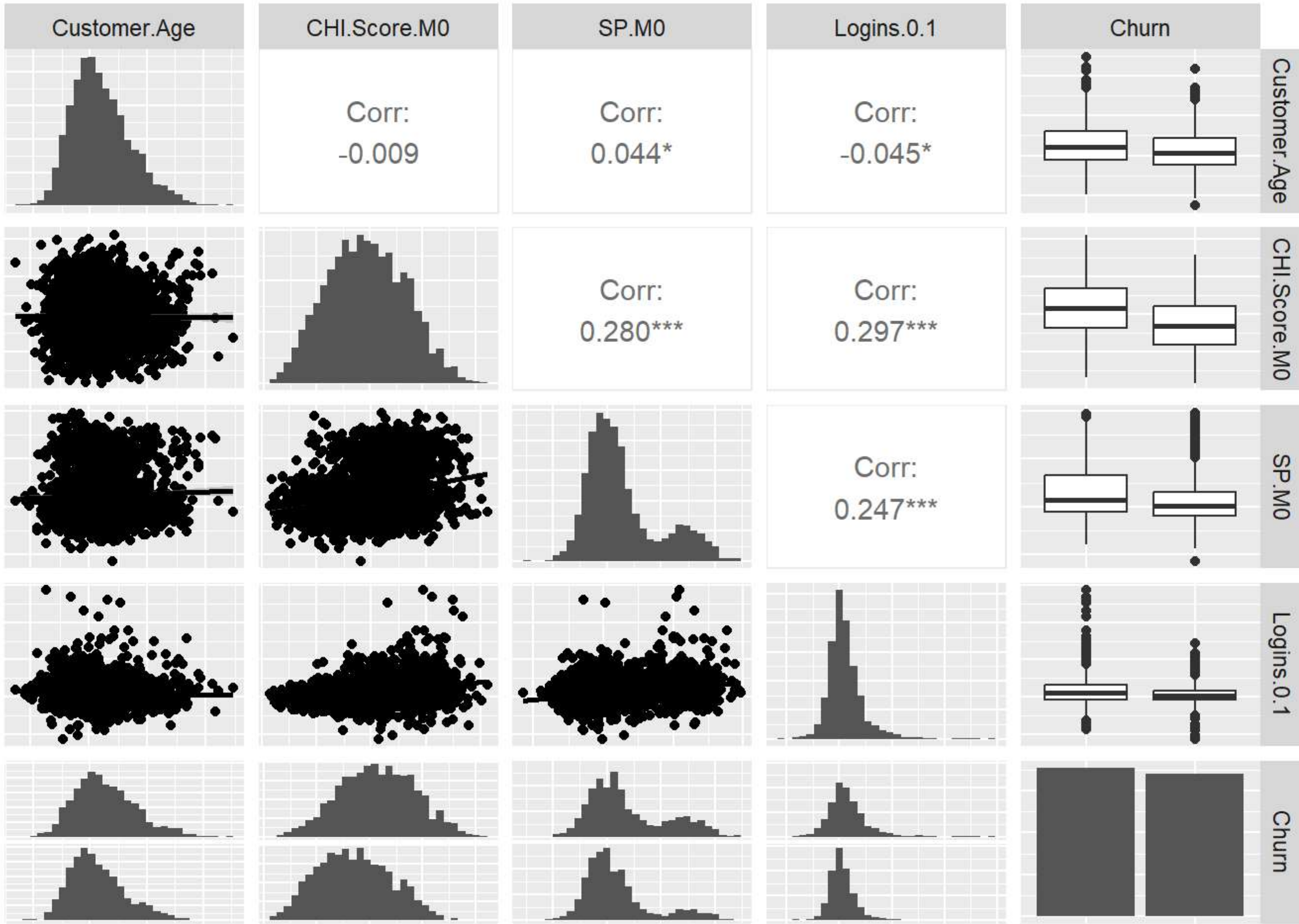
```
prop.table(table(Churn))
```

```
## Churn
##          0          1
## 0.94307692 0.05692308
```

Usando un punto de corte de 0.05:

- Multicolinealidad
- Analizar la relación entre variables

```
ggpairs(data_final, lower = list(continuous = "smooth"),
        diag = list(continuous = "barDiag"), axisLabels = "none")
```




```
  *  Interpretando coeficientes en el modelo
```

Ahora que estamos seguros de que nuestro modelo se ajusta razonablemente bien a los datos, Podemos proceder a interpretarlo. Los coeficientes:log(p/(1-p))=b0+b1x1+…+b3x3

Cuanto mayor sean las probabilidades logarítmicas, más probable será el evento de referencia. Por lo tanto, los coeficientes positivos indican que el evento se vuelve más probable y los coeficientes negativos indican que el evento se vuelve menos probable.

```
kable(summary(modelo1)$coefficients, digits=4, scientific=FALSE)
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.5188	0.1399	10.8529	0.0000
Customer.Age	-0.0296	0.0046	-6.4279	0.0000
CHI.Score.M0	-0.0082	0.0007	-11.4838	0.0000
SP.M0	-0.0736	0.0364	-2.0215	0.0432
Logins.0.1	-0.0042	0.0016	-2.6971	0.0070

```
exp(coefficients(modelo1))
```

```
##  (Intercept) Customer.Age CHI.Score.M0        SP.M0   Logins.0.1
##    4.5668373    0.9708348    0.9917934    0.9290052    0.9957758
```

- Interpretando coeficientes

exp (β1) = 0.98. Por cada punto adicional de antigüedad del cliente, la probabilidad de disminuye disminuye en 2%

exp (β2) = 0.99. Por cada punto adicional en el indice de felicidad, la probabilidad de disminuye disminuye en 1%

exp (β3) = 0.91. Por cada punto adicional de SP.M0, la probabilidad de disminuye disminuye en 9%

exp (β4) = 0.99. Por cada punto adicional de Logins.0.1, la probabilidad de disminuye disminuye en 1%