



**y otros**

Árboles de decisión con SAS

# Índice

1. Presentaciones
2. Rentabilidad empieza con R
3. ¿Sólo R?
4. R y otros
5. Árboles de decisión con SAS

# 1. Presentaciones

Raúl Vaquerizo

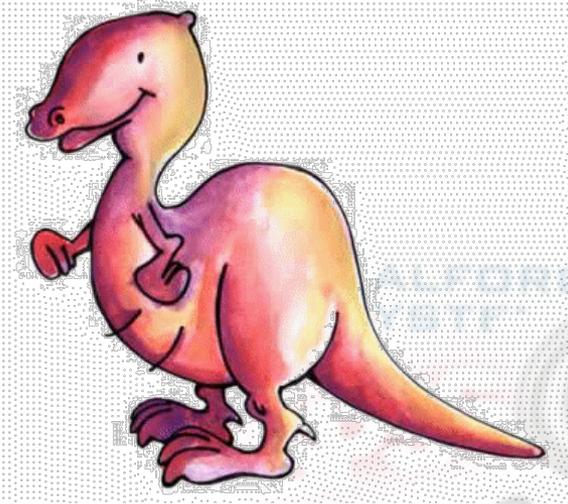
[www.analisydecision.es](http://www.analisydecision.es)

## 2. Rentabilidad empieza con R

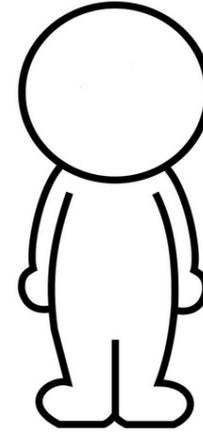
Software comercial implica ¡¡€!!

**R no cuesta €**

### 3. ¿Sólo R?



Dinosaurios:  
Extintos, duraron 165  
millones de años



Homo sapiens:  
En vías de extinción  
¿200.000 años?

# 4. R y otros



R

## SAS® Analytics Pro

SAS Analytics Pro provides all of the functionality needed to access, manipulate, analyze and present information. For more information, click [here](#).

How many user licenses?

**5.920 €**

First-year license fee\*\* for Desktop Version of Windows  
\* Includes Technical Support  
\* Includes Documentation  
\* Includes Software Upgrades

You can add other products to your order in the next step

\*\*ANNUAL LICENSE: SAS software licenses are renewable on an annual basis. Annual renewal fees are approximately 28 percent of the then-current first-year license fee. The fee shown does not include applicable taxes.

<http://cran.r-project.org/mirrors.html>

# 5. Árboles de decisión con SAS

## Modelización y creación de gráficos con R

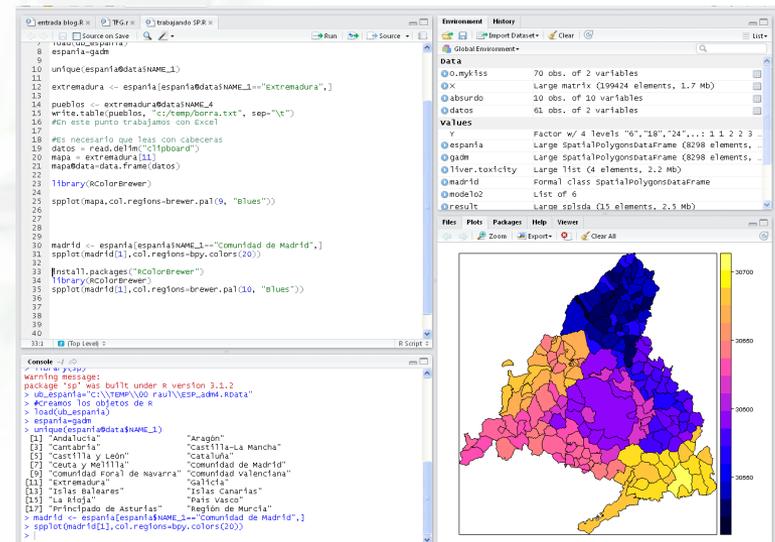
```
SAS - [Log - (Sin título)]
Archivo Edición Ver Herramientas Soluciones Ventanas Ayuda

Explorador
Librerías activas
Emblem Maps
Mapsgfk Mapssas
Sashelp Sasuser
Work

19 if edad<30 then potencial=potencial+0.05*ranuni(13);
20 else if edad<40 then potencial=potencial+0.03*ranuni(13);
21 else if edad<45 then potencial=potencial+0.04*ranuni(13);
22 else if edad<55 then potencial=potencial+0.05*ranuni(13);
23 else potencial=potencial+0.02*ranuni(13);
24 *RECIBOS;
25 if recibos<=2 then potencial=potencial+0.01*ranuni(13);
26 else if recibos<=4 then potencial=potencial+0.03*ranuni(13);
27 else potencial=potencial+0.05*ranuni(13);
28 *VINCULACION;
29 if vinculacion<=5 then potencial=potencial+0.01*ranuni(13);
30 else if vinculacion<=8 then potencial=potencial+0.05*ranuni(13);
31 else if vinculacion<=12 then potencial=potencial+0.2*ranuni(13);
32 else potencial=potencial+0.4*ranuni(13);
33 *COMPRAS;
34 if compras<100 then potencial=potencial-0.01*ranuni(13);
35 else if compras<250 then potencial=potencial+0.03*ranuni(13);
36 else if compras<700 then potencial=potencial+0.05*ranuni(13);
37 else if compras<2000 then potencial=potencial+0.06*ranuni(13);
38 else potencial=potencial+0.07*ranuni(13);
39 *PASIVO;
40 if pasivo<50000 then potencial=potencial-0.02*ranuni(13);
41 else if pasivo<100000 then potencial=potencial-0.01*ranuni(13);
42 else if pasivo<200000 then potencial=potencial+0.05*ranuni(13);
43 else if pasivo<300000 then potencial=potencial+0.1*ranuni(13);
44 else if pasivo<500000 then potencial=potencial+0.1*ranuni(13);
45 else potencial=potencial+1;
46 *PROVINCIA;
47 if mod(provincia,5)=0 then potencial=potencial-0.02*ranuni(13);
48 if potencial<0.4 then contrata=1;
49 else contrata=0;
50 run;

NOTE: There were 200000 observations read from the data set WORK.DAT05.
NOTE: The data set SASUSER.DAT05 has 200000 observations and 9 variables.
NOTE: Sentencia DATA used (Total process time):
      real time      2.03 seconds
      cpu time       0.18 seconds
```

## Gestión de datos con SAS



## 5. Árboles de decisión con SAS



Podemos:

1. Trabajar con una u otra herramienta indistintamente
2. Automatizar procesos desde SAS

# 5. Árboles de decisión con SAS

## Automatizar el proceso con SAS

```
*EXPORTACION A CSV DEL DS;  
PROC EXPORT DATA= entreno  
OUTFILE= "c:\temp\elimina.csv"  
DBMS=CSV REPLACE;  
RUN;
```

Exporta el conjunto de datos  
SAS a CSV

```
data ejecucion_R;  
infile datalines dlm='@';  
input lineas: $200.;  
lineas = tranwrd(lineas,'names[i]','\n','names[i];\n');  
datalines4;
```

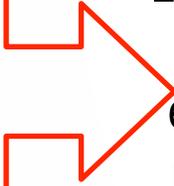
Vamos a crear un conjunto de datos SAS con  
donde insertamos el código en R

# 5. Árboles de decisión con SAS

## Automatizar el proceso con SAS

```
data ejecucion_R;  
infile datalines dlm='@';  
input lineas: $200.;  
lineas = tranwrd(lineas,'names[i]","\n"', 'names[i]","\n");  
Datalines4:
```

```
setwd('c:/temp') @  
dfsas <- read.csv('elimina.csv') @  
library(rpart) @  
arbol=rpart(as.factor(contrata)~ pasivo + edad + @  
recibos + vinculacion + compras , @  
data=dfsas,method="anova", @  
control=rpart.control(minsplit=30, cp=0.0008) ) @  
arbol$frame @
```

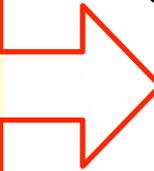


Lee en R los datos de SAS que hemos exportado en CSV y realiza un árbol con ellos

# 5. Árboles de decisión con SAS

## Automatizar el proceso con SAS

```
#####@
#Ubicación de salida del modelo @
fsalida = "C:\\temp\\reglas_arbol.txt" @
#Función que identifica las reglas @
reglas.rpart <- function(model) @
{frm <- model$frame @
names <- row.names(frm) @
cat("\n",file=fsalida) @
for (i in 1:nrow(frm)) @
{if (frm[i,1] == "") @
{cat("\n",file=fsalida,append=T) @
cat(sprintf("IF ", names[i]),file=fsalida, @
append=T) @
pth <- path.rpart(model, nodes=as.numeric(names[i]),@
print.it=FALSE) @
cat(sprintf(" %s\n", unlist(pth)[-1]), sep=" AND ", @
file=fsalida, append=T) @
cat(sprintf("THEN NODO= "),names[i],";\n", @
file=fsalida,append=T)}} @
#####@
reglas.rpart(arbol) @
;;;
run;
```



Genera una función en R que incluso permite crear el código SAS que aplica las reglas creadas

# 5. Árboles de decisión con SAS

## Automatizar el proceso con SAS

```
data _null_;  
set ejecucion_R;  
file "c:\temp\pgm.R";  
put lineas;  
run;  
  
data _null_;  
file "c:\temp\ejecucion.bat";  
*TENEMOS QUE TENER EN CUENTA DONDE ESTA  
R Y QUE VERSION  
TENEMOS;  
put "'C:\Archivos de programa\R\R-2.12.0\bin  
\R.exe'"@@;  
put ' CMD BATCH --no-save "'@@;  
put "c:\temp\pgm.R"@@;  
put "";  
call sleep (150);  
run;  
*EJECUTAMOS EL PROCESO;  
options noxwait;  
x "C:\temp\ejecucion.bat";
```



SAS guarda el programa creado previamente y se ejecuta R en modo BATCH

# 5. Árboles de decisión con SAS

## Automatizar el proceso con SAS

```
*EJECUTAMOS EL CODIGO GENERADO;  
data validacion;  
set validacion;  
%include "C:\temp\reglas_arbol.txt";  
run;  
*PODEMOS ANALIZAR SU RESULTADO;  
proc sql;  
select nodo,  
count(*) as cli,  
sum(contrata)/count(*) as tasa  
from validacion  
group by 1;  
quit;
```



Podemos usar las reglas generadas por el modelo creado con R

# 5. Árboles de decisión con SAS

Ejemplo reproducible

Escribimos en google árboles, decisión, SAS



Muchas gracias