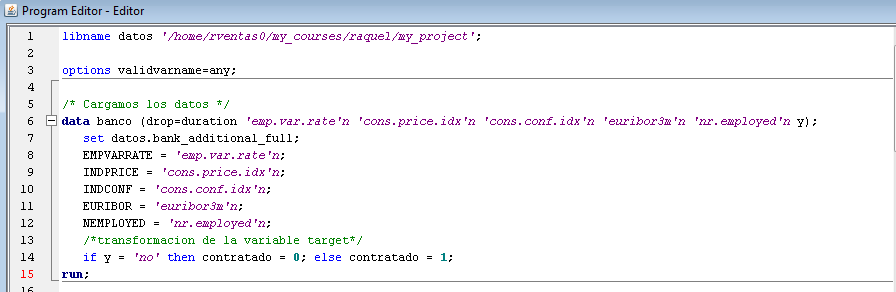
El objetivo de la práctica es construir un modelo que nos prediga el éxito de una campaña de marketing telemático. Deberemos además distinguir qué clientes son los más propensos a la contratación de un depósito.

## Primera toma de contacto con los datos

Tenemos un dataset de 41188 filas y 20 variables. Podemos distinguir 3 grupos de datos:

* Datos que describen al cliente: edad, trabajo, estado civil, nivel de estudios, si tiene una deuda pendiente y si tiene contratado un préstamo hipotecario.
* Datos relacionados con la campaña de marketing: a través de qué medio se contactó con el cliente (fijo o móvil), mes del año en el que fue contactado, qué día de la semana era, duración de la última llamada, número de contactos durante la última campaña, número de días transcurridos desde que el cliente fue contactado por última vez en la campaña anterior, si tuvo éxito.
* Datos socioeconómicos: tasa de empleo, índice de precios al consumidor, índice de confianza, tasa de Euribor a 3 meses, número de empleados.

Fijándonos en las descripciones de las variables de nuestra tabla, dejamos fuera la variable duration en el análisis, ya que según la descripción de esta variable tenemos que puede usarse como benchmark, pero no debe ser tenida en cuenta para un modelo predictivo realista.

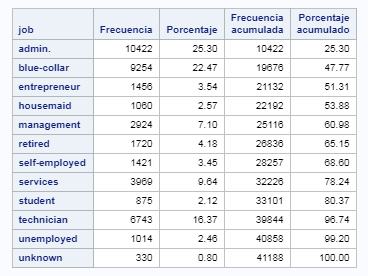


También aprovecho la carga de los datos para renombrar las variables referentes a los datos socioeconómicos para que sea más sencillo a la hora de operar con ellas. Otra cosa que hago es transformar la variable objetivo en valores numéricos 0 y 1.

**Nota**: He alternado la ejecución del código entre SAS Miner, SAS Guide y SAS Studio. Los resultados que muestro están sacados de las distintas herramientas, según he ido probando.

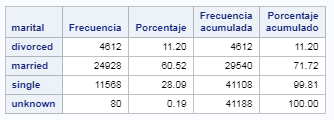
#### Análisis de frecuencias

**proc freq** data=banco; **run;**

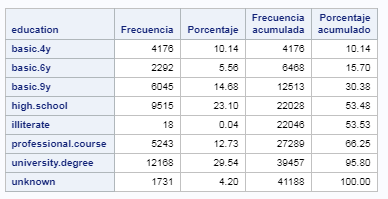
* Job: Observamos que casi el 50% de las observaciones corresponden a los trabajos admin (25.30%) y blue-collar(obreros) (22.47%). Le siguen en importancia los técnicos, que tienen una representación del 16.37%. 

También observamos que tenemos un 0,8% de desconocidos (unknown).

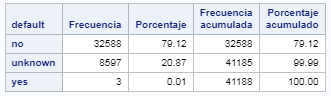
* Marital: La mayoría (60.52%) están casados. Hay un pequeño porcentaje (0.19%) de desconocidos.



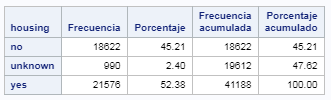
* Education: Algo más del 50% de los datos están repartidos entre “high.school” (23.10%) y “university.degree” (29.54%). Encontramos un 4.20% de desconocidos.



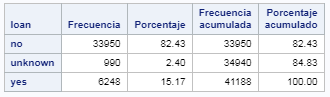
* Default: Aquí vemos que la mayoría de los datos (99.99%) se corresponden con clientes que no tienen deuda pendiente o cuyo dato es desconocido. Sólo el 0.01% de clientes tienen una deuda pendiente.



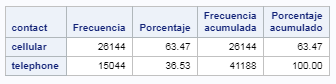
* Housing: La distribución de esta variable es bastante equilibrada entre clientes que tienen un crédito hipotecario y aquellos que no lo tienen. Aquí encontramos un 2.40% de valores desconocidos.



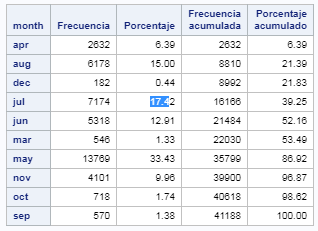
* Loan: En este caso, la mayoría de clientes (82.43%) no tienen un préstamo personal frente al 15.17% que sí lo tiene. Aquí de nuevo encontramos un 2.40% de valores desconocidos.



* Contact: El 63.47% de los contactos se realiza a través de teléfono móvil frente al 36.53% de contactos a través del fijo. En teste caso no tenemos valores desconocidos.



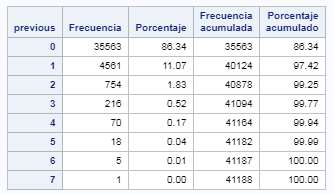
* Month: Los meses que más contactos acumulan son mayo, junio, julio y agosto, con un porcentaje acumulado del 78.76%.



* Day\_of\_week: Aquí vemos que hay una distribución uniforme entre los distintos días de la semana.



* Campaign: La mayoría de las veces (81.46%) se han realizado 1, 2 o 3 contactos durante la campaña.
* Las variables previous, poutcome y pdays se refieren a datos de campañas previas. Si observamos el valor de la variable previous, podemos ver que hay 35563 clientes que no han sido contactados con anterioridad, lo cual coincide con el valor **nonexistent** de la variable poutcome.



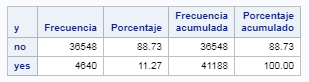
Esto tiene sentido ya que si no se contactó al cliente en la campaña previa no podemos clasificar como ‘éxito’ o `fracaso’ el resultado de la misma.



En el caso de pdays vemos que hay 39673 valores 999, que según la descripción de las variables indican que el cliente no ha sido contactado previamente. Si tenemos en cuenta que en la variable previous había 35563 clientes no contactados vemos que algunos de estos valores, 4110 (39673 – 35563) se corresponden con desconocidos, es decir, que no se conocen los días que pasaron desde que el cliente fue contactado con anterioridad.



* Variable objetivo: Es bastante representativo el desbalanceo de los valores de la variable target, tenemos un 88.73% de resultados **no** frente a un 11.27% de **sí**.



También podemos observar que no hay valores missing.

Casi todas las variables analizadas hasta aquí son categóricas, a excepción de campaing, pdays y previous. Además nos encontramos con las variables numéricas age, emp.var.rate, cons.price.idx, cons.conf.idx y euribor3m.

Voy a analizar con gráficos la distribución de estas variables:

ods listing close;

ods listing gpath="/home/rventas0/my\_courses/raquel/data\_output";

proc univariate data=banco normal plot;

var age EMPVARRATE INDPRICE INDCONF EURIBOR NEMPLOYED;

qqplot age EMPVARRATE INDPRICE INDCONF EURIBOR NEMPLOYED / NORMAL (MU=EST SIGMA=EST COLOR=RED L=1);

HISTOGRAM /NORMAL(COLOR=MAROON W=4) CFILL = BLUE CFRAME = LIGR;

INSET MEAN STD /CFILL=BLANK FORMAT=5.2;

run;

ods listing close;

Obtengo los siguientes gráficos:

|  |  |  |
| --- | --- | --- |
|  |  | |
|  |  | |
|  |  | |
|  |  | |
|  |  | |
|  | |  |
|  | |  |
|  | |  |
|  | |  |

A la vista de los resultados y aunque las variables son continuas observo que tienen un comportamiento “discreto”, por lo que voy a agrupar los valores en distintas categorías, según los rangos por los que se mueven sus valores.

## Preprocesamiento

Con las conclusiones que he sacado del apartado anterior voy a hacer el tratamiento de los datos para posteriormente proceder a la modelización. Como la variable target tiene valores desbalanceados (88.73% de resultados **no** frente a un 11.27% de **sí**) voy a tomar una muestra de los datos que contenga tantos valores positivos como negativos, para que no haya una mayor influencia de los valores negativos en el modelo. Para ello, me quedo con los 4.640 resultados positivos y con una muestra aleatoria de 4.640 valores negativos. Para intentar que la proporción de los valores categóricos se mantenga en la muestra seleccionada, voy a estratificar por aquellas variables que contienen más categorías (no puede hacerse por todas las variables porque salen más “estratos” que valores que tiene la muestra).

proc sort data=banco\_no out=banco\_no\_ordenado;

by job marital education month day\_of\_week;

run;

proc surveyselect data=banco\_no\_ordenado out=banco\_no\_sample seed=1979

method=srs sampsize=4640;

strata job marital education month day\_of\_week / alloc=prop;

run;

Podemos ver la distribución de los valores de las variables según la variable objetivo contratado.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Aquí a simple vista podemos ver que aparentemente las variables housing, loan, day\_of\_week y age no influyen en el valor de la variable objetivo. Voy a dejar fuera la variable default, ya que como vimos antes el 99.99% de los datos se corresponden con valores negativos y unknown, por lo que este dato es de mala calidad. Además, para el caso de las variables job y marital, voy a incluir los unknown dentro de la categoría mayoritaria (admin y married respectivamente) por representar un porcentaje muy bajo dentro del total. Así consigo un nivel menos en estas variables que resta complejidad.

También voy a dividir en tramos las variables campaign, pdays, previous, emp.var.rate, cons.price.idx, cons.conf.idx, euribor3m y nr.employed, para así “categorizar” sus valores y además eliminar outliers.

* Variable age: Tramifico los valores:

data banco\_sample;

set banco\_sample;

if age <= 25 then age = 1;

else if 25 < age <= 35 then age = 2;

else if 35 < age <= 45 then age = 3;

else if 45 < age <= 55 then age = 4;

else if 55 < age <= 65 then age = 5;

else age = 6; /\* mayor de 65 \*/

run;

* Variable campaign: Divido los datos en 1, 2, 3, o más de tres contactos:

data banco\_sample;

set banco\_sample;

if campaign > 3 then campaign = 4;

run;

* Variable pdays: Voy a dividir en dos categorías, no contactado o contactado:

data banco\_sample;

set banco\_sample;

if pdays = 999 then pdays = 0; /\*no contactado\*/

else pdays = 1; /\*contactado\*/

run;

* Variable previous: Al igual que con pdays divido en nocontactado o contactado:

data banco\_sample;

set banco\_sample;

if previous > 0 then previous = 1; /\*contactado\*/

run;

* Variable emp.var.rate: Observando los gráficos he decidio dividir en tres grupos, emp.var.rate menor o igual a -1.9, entre -1.9 y -0.1 y mayor a -1.9.

data banco\_sample;

set banco\_sample;

if EMPVARRATE <= -1.9 then EMPVARRATE = 1;

else if -1.9 < EMPVARRATE <= -0.1 then EMPVARRATE = 2;

else EMPVARRATE = 3;

run;

* Variable cons.price.idx: En este caso he establecido los tramos en los que la variable vale menos o igual a 93, entre 93 y 94.2 y mayor o igual a 94.2.

data banco\_sample;

set banco\_sample;

if INDPRICE <= 93 then INDPRICE = 1;

else if 93 < INDPRICE < 94.2 then INDPRICE = 2;

else INDPRICE = 3;

run;

* Variable cons.conf.idx: Voy a dividir en valores menores o iguales a -46.8, entre --46.8 y -34.8 y por último valores mayores o iguales a -34.8.

data banco\_sample;

set banco\_sample;

if INDCONF <= -46.8 then INDCONF = 1;

else if -46.8 < INDCONF < -34.8 then INDCONF = 2;

else INDCONF = 3;

run;

* Variable euribor3m: En este caso las divisiones son valores menores a 1.25, entre 1.25 y 3.95, entre 3.95 y 4.85 y mayores a 4.85.

data banco\_sample;

set banco\_sample;

if EURIBOR < 1.25 then EURIBOR = 1;

else if 1.25 <= EURIBOR < 3.95 then EURIBOR = 2;

else if 3.95 <= EURIBOR < 4.85 then EURIBOR = 3;

else EURIBOR = 4;

run;

* Variable nr.employed: Los rangos que he establecido son valores menores de 5091, entre 5091 y 5181, entre 5181 y 5217 y valores mayores a 5217.

data banco\_sample;

set banco\_sample;

if NEMPLOYED < 5091 then NEMPLOYED = 1;

else if 5091 <= NEMPLOYED <= 5181 then NEMPLOYED = 2;

else if 5181 < NEMPLOYED <= 5217 then NEMPLOYED = 3;

else NEMPLOYED = 4;

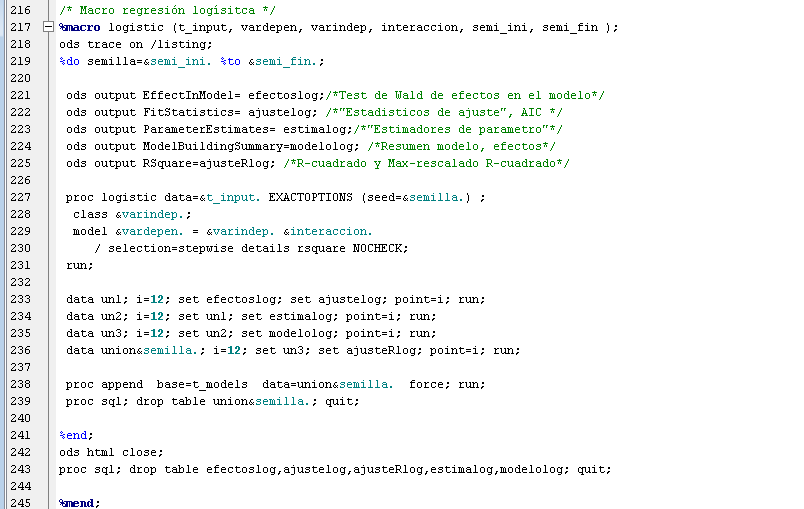
run;

## Selección del modelo

##### Regresión logística

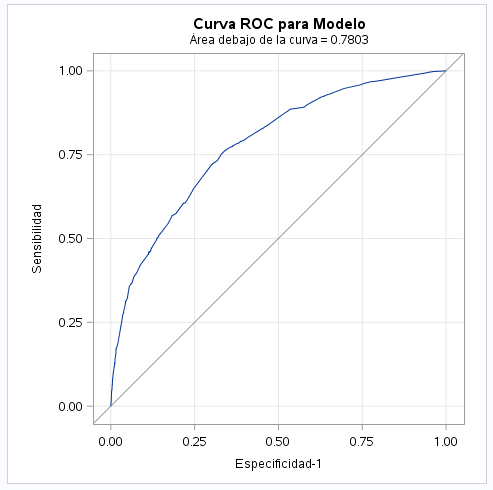
Voy a empezar con un modelo de regresión logística, dada la naturaleza de la variable target.

Defino una macro que he nombrado como logistic que se encarga de hacer regresiones logísticas para distintas semillas que se pasan como parámetros.



Empiezo con una primera aproximación, ejecutando la llamada con las 14 variables que he dejado. Después voy a seguir añadiendo interacciones. Tras ejecutar 100 modelos, los resultados que he obtenido son los que se pueden ver en el documento *PracticaRaquelRegLogistica-resultados.pdf*.

Voy a coger todas las variables que me han salido, ya que todas presentan más o menos los mismos valores de R2 y tienen errores bajos. El modelo que me sale tiene la siguiente curva ROC:



##### Modelos lineales generalizados

No voy a hacer un modelo de regresión lineal para el tipo de problema que nos ocupa, ya que la variable objetivo es dicotómica.

He creado una macro que ejecuta el procedimiento glmselect para un rango de semillas pasadas por parámetro y tomando un conjunto de variables (con o sin interacciones) para el modelo. Además, para cada semilla se ejecuta tres veces con fracciones diferentes. Lanzo esta macro tres veces con 30 semillas para obtener 270 modelos. Los resultados los he guardado en una tabla llamada t\_models que guarda por un lado los efectos del modelo, con el ASE, la fracción y la semilla con la que fueron generados. Según los resultados obtenidos (se pueden ver en el documento *Informe\_mgmselect.pdf*, los dos modelos que más se repiten son:

*Intercept job marital education contact month pdays previous NEMPLOYED*

*Intercept marital contact month pdays previous NEMPLOYED job\*education EURIBOR\*job*

Voy a usar primero la instrucción PROC GLMSELECT para seleccionar el modelo.

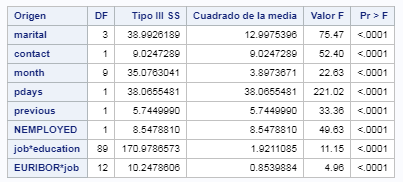
|  |  |
| --- | --- |
|  |  |

A la vista de los resultados, me voy a quedar con el segundo modelo, el de la derecha. Tiene valores de R2 y de R2 ajustado mayores y además los valores AIC, AICC, BIC, PRESS, SBC y ASE son menores.

Voy a hacer el análisis específico con el procedimiento GLM. Obtengo una bondad de ajuste de 0.31.

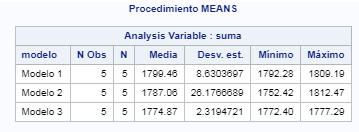


Además, todos los efectos son significativos porque tienen un p-valor menor a 0.05:



##### Redes neuronales

De nuevo voy a usar una macro que recibe como parámetros la función de activación, el número de nodos y un rango con semillas. Voy a limitar la ejecución de estos modelos a 5 semillas ya que son modelos que tardan más y dispongo de poco tiempo. Además, he ejecutado tres veces la macro, en la primera ejecución que guardaré posteriormente en la tabla final de resultados con el valor Modelo1 en la columna modelo, utilizo como función de activación la tangente hiperbólica, con 4 nodos. En la segunda ejecución (Modelo2), de nuevo la función de activación tangente hiperbólica, esta vez con 6 nodos. Por último, he cambiado la función de activación por una lineal y 4 nodos. Los resultados vuelco en una tabla:



Podemos observar que, de las tres opciones, la peor es la segunda. Curiosamente funciona mejor el modelo con 4 nodos que con 6. Por último, veo que el modelo con la función de activación lineal es el mejor de los tres. Me quedo con este último modelo y lo ejecuto:

## Conclusiones