UCM - Mineria de Datos

SEGUNDO TRABAJO SEMMA 2

TÉCNICAS Y METODOLOGÍA DE LA MINERÍA DE DATOS (SEMMA)

CAIO FERNANDES MORENO

1. Enunciado del trabajo

Segundo Trabajo SEMMA: Regresión

Fecha de entrega: 19 de enero de 2016.

Se trata de construir un modelo de predicción a partir de regresión lineal (variable dependiente continua u ordinal con más de 7 valores diferentes).

Se trabajará sobre 2 archivos de datos:

1) Los datos del concurso de Kaggle.

2) Un archivo de datos escogido a voluntad de entre a) y b):

a) Uno entre los ocho que se aportan en el archivo datosregresion.zip . La explicación de cada archivo está en un txt o en un doc.

b) Un archivo libre de cualquier otra cosa, pero de complejidad parecida a los de a). Una web de donde se pueden descargar es:

http://archive.ics.uci.edu/ml/datasets.html?format=&task=reg&att=&area=&numAtt=&numIns=&type=&sort=attup&view=table

---

Contenidos del trabajo

Se trata de establecer el mejor modelo predictivo posible, considerando como "mejor" el criterio de performance en validación cruzada repetida, intentando buscar estabilidad (baja variabilidad) del modelo además de bajo error promedio.

1) Se pide, con un cierto espíritu de síntesis:

- Gráficos
- Construcción de interacciones
- Métodos Stepwise
- Razonamientos sobre creación de variables nuevas, problemas varios y otras consideraciones.
- Validación cruzada.

- Conclusiones y comentarios

2) Se pide emplear la mayor parte del tiempo (el 80%) en la construcción de regresión y en la redacción (número de páginas) del segundo archivo, pues es más personal. Para el primer archivo (el de Kaggle) hay que concursar al menos con 7 modelos diferentes y poner en el trabajo la puntuación conseguida. No se pide más, salvo explicar muy básicamente como se han encontrado los modelos. Otra cosa es que lo trabajéis más por puro vicio, lo cual es muy bueno también.

2. Ejecución del trabajo

Para el trabajo he utilizado en fichero llamado **vino.sas7bdat**.

Conforme el archivo **wine_quality.txt** de descripción del *dataset* utilizado las variables son:

Input variables (based on physicochemical tests):

% 1 - fixed acidity

% 2 - volatile acidity

% 3 - citric acid

% 4 - residual sugar

% 5 - chlorides

% 6 - free sulfur dioxide

% 7 - total sulfur dioxide

% 8 - density

% 9 - pH

% 10 - sulphates

% 11 - alcohol

% Output variable (based on sensory data):

% 12 - quality (score between 0 and 10)

No hay valores *missing* en las variables lo que facilita el trabajo.

Código SAS para cargar los datos:

```
/* Carga los datos SAS en la libreria disco que estan en la carpeta */
libname trabajo2 'C:\Users\win\Documents\GitHub\ucm\semma\trabajo2';

data uno; set trabajo2.Vino; run;
```

Es necesario hacer una transformación inicial en los datos para crear la variable tipovino donde las primeras instancias hasta el 1599 son *red wine* y de la observación 1600 hasta la 6497 son *white wine*. *(Number of Instances: red wine - first 1599 instances; white wine - instances 1600 to 6497.)*

Para esta transformación he utilizado el código SAS a bajo:

```
data uno;
      set trabajo2.Vino;
      if _n_ < 1600  then tipovino='red';
    else tipovino='white';
run;

proc print data=uno;run;
```

A bajo se ensena las 10 primeras líneas del *dataset* cargado y transformado:

The SAS System

Obs	fixed	volatile	citric	sugar	chlorides	freesulfur	totalsulfur	density	pH	sulphates	alcohol	quality	tipovino
1	7.40	0.700	0.00	1.90	0.076	11.0	34.0	0.99780	3.51	0.56	9.4000	5	red
2	7.80	0.880	0.00	2.60	0.098	25.0	67.0	0.99680	3.20	0.68	9.8000	5	red
3	7.80	0.760	0.04	2.30	0.092	15.0	54.0	0.99700	3.26	0.65	9.8000	5	red
4	11.20	0.280	0.56	1.90	0.075	17.0	60.0	0.99800	3.16	0.58	9.8000	6	red
5	7.40	0.700	0.00	1.90	0.076	11.0	34.0	0.99780	3.51	0.56	9.4000	5	red
6	7.40	0.660	0.00	1.80	0.075	13.0	40.0	0.99780	3.51	0.56	9.4000	5	red
7	7.90	0.600	0.06	1.60	0.069	15.0	59.0	0.99640	3.30	0.46	9.4000	5	red
8	7.30	0.650	0.00	1.20	0.065	15.0	21.0	0.99460	3.39	0.47	10.0000	7	red
9	7.80	0.580	0.02	2.00	0.073	9.0	18.0	0.99680	3.36	0.57	9.5000	7	red
10	7.50	0.500	0.36	6.10	0.071	17.0	102.0	0.99780	3.35	0.80	10.5000	5	red

El código en SAS a bajo nos ayuda conocer mejor los tipos de variables en el *dataset*:

```
/*  List all the variables in the dataset */
proc contents data=uno out=sa;
data;set sa;if _n_=1 then put 'LISTA DE VARIABLES CONTINUAS';if type=1
then put name @@;run;
```

Alphabetic List of Variables and Attributes			
#	**Variable**	**Type**	**Len**
11	alcohol	Num	8
5	chlorides	Num	8
3	citric	Num	8
8	density	Num	8
1	fixed	Num	8
6	freesulfur	Num	8
9	pH	Num	8
12	quality	Num	8
4	sugar	Num	8
10	sulphates	Num	8
13	tipovino	Char	3
7	totalsulfur	Num	8
2	volatile	Num	8

Con los datos cargados vamos crear nuestro primer modelo donde llamaremos de **modelo1**.

Utilizando el material "**SEMMA Tema 5 macros trucos regresión 2015-16.pdf**" fornecido por el Prof. Javier Portela es posible aplicar una serie de macros y trucos con el objetivo principal de encontrar el mejor modelo en regresión.

Los próximos pasos son utilizar técnicas avanzadas, macros y trucos en selección de modelos en regresión y llegar a un resultado final y una conclusión.

Primera parte: modelos tentativos iniciales

Tenemos un total de 6497 variables y 13 variables.

El objetivo es predecir la calidad del vino utilizando la variable dependiente (variable de salida) *quality* y las otras variables como independientes (variables de entrada).

La calidad del vino se mide con una escala de 0 (muy mala) a 10 (muy excelente).

Utilizando Regresión linear múltiple con SAS, empezaremos seleccionando el método pasos sucesivos (*stepwise*).

Se puede utilizar análisis de regresión lineal múltiple para explorar y cuantificar la relación entre una variable llamada dependiente o criterio (Y) y una o más variables llamadas independientes o predictoras (X1, X2, ..., Xp). El análisis de regresión lineal múltiple estudia la relación entre variables cuantitativas.

El método Pasos sucesivos (stepwise) lo que hace es ir eligiendo que variables metes y que variables sacas en función del valor de probabilidad asociado al estadístico F. Es decir

que al incluir una variables esta variable no tiene un valor de probabilidad que sea significativo no será incluida en el modelo y si el valor de probabilidad asociada a esta variable tiene un valor que está por encima de 0.01 esta variable aun que un determinado momento estuviera incluido en el modelo saldría fuera.

Pasos sucesivos (*stepwise*) nos servirían para hacer una analice exploratoria, una regresión linear exploratoria, donde no sabemos muy bien cual variables puede ser importantes o no, la idea es seleccionar este método de introducir variables, porque el introduce y quita variables con un criterio puramente matemático.

Modelo 1 utilizando pasos sucesivos (stepwise)

```
proc glmselect data=uno;
class tipovino;
model quality=freesulfur citric alcohol chlorides density volatile
totalsulfur sulphates pH sugar fixed
/selection=stepwise(select=SBC choose=SBC);
run;
```

Resultados (he removido algunas partes de la salida de SAS para simplicar los resultados):

The GLMSELECT Procedure

Data Set	WORK.UNO
Dependent Variable	quality
Selection Method	Stepwise
Select Criterion	SBC
Stop Criterion	SBC
Choose Criterion	SBC
Effect Hierarchy Enforced	None

The GLMSELECT Procedure

Stepwise Selection Summary

Step	Effect Entered	Effect Removed	Number Effects In	SBC
0	Intercept		1	-1753.2635
1	alcohol		2	-3173.3204
2	volatile		3	-3687.5574
3	sulphates		4	-3764.8201
4	sugar		5	-3835.4742
5	totalsulfur		6	-3851.0030
6	freesulfur		7	-3907.0508
7	chlorides		8	-3907.0637*

*** Optimal Value Of Criterion**

Selection stopped at a local minimum of the SBC criterion.

Stop Details				
Candidate For	**Effect**	**Candidate SBC**		**Compare SBC**
Entry	pH	-3904.7262	>	-3907.0637
Removal	chlorides	-3907.0508	>	-3907.0637

The GLMSELECT Procedure

Selected Model

The selected model, based on SBC, is the model at Step 7.

Effects: Intercept freesulfur alcohol chlorides volatile totalsulfur sulphates sugar

Analysis of Variance				
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**
Model	7	1431.20237	204.45748	376.64
Error	6489	3522.48333	0.54284	
Corrected Total	6496	4953.68570		

Root MSE	0.73678
Dependent Mean	5.81838
R-Square	0.2889
Adj R-Sq	0.2881
AIC	2537.70358
AICC	2537.73132
SBC	-3907.06366

Parameter Estimates				
Parameter	**DF**	**Estimate**	**Standard Error**	**t Value**
Intercept	1	2.508329	0.119898	20.92
freesulfur	1	0.006132	0.000749	8.18
alcohol	1	0.328188	0.008920	36.79
chlorides	1	-0.946501	0.319300	-2.96
volatile	1	-1.370462	0.064389	-21.28
totalsulfur	1	-0.002411	0.000263	-9.18
sulphates	1	0.677816	0.068319	9.92
sugar	1	0.021504	0.002322	9.26

El mejor paso ha sido el 7, donde la función *stepwise* ha probado distintos modelos quitando y añadiendo variables y ha elegido el paso (step) 7, donde quedaríamos con las variables: freesulfur alcohol chlorides volatile totalsulfur sulphates sugar.

Pero esto ha sido un primero pasos y aún tenemos que probar muchísimos modelos para llegar a al modelo seleccionado.

Comparativo utilizando atras (backward) normal

Ahora con el método atrás (backward) normal (SLE=0.05,SLS=0.10), en el enunciando se pide para utilizar el stepwise pero con el objetivo de aprender y comparar pongo unas pruebas con el backward normal, exigente y muy exigente como se ve en las transparencias vistas en clase.

```
/*  selection=backward(select=SL ); */
proc glmselect data=uno;
class tipovino;
model quality=freesulfur citric alcohol chlorides density volatile
totalsulfur sulphates pH sugar fixed
/selection=backward(select=SL );
run;
```

Resultados:

The selected model is the model at the last step (Step 1).

Effects: Intercept freesulfur alcohol chlorides density volatile totalsulfur sulphates pH sugar fixed

Analysis of Variance				
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**
Model	10	1446.12864	144.61286	267.41
Error	6486	3507.55706	0.54079	
Corrected Total	6496	4953.68570		

Root MSE	0.73538
Dependent Mean	5.81838
R-Square	0.2919
Adj R-Sq	0.2908
AIC	2516.11452
AICC	2516.16263
SBC	-3908.31543

Comparativo utilizando atras (backward) exigente

Ahora con el metodo atras (backward).

```
/*  Ahora con el metodo atras backward exigente:
/selection=backward(select=SL ) sle=0.01 sls=0.01;  */
```

```
proc glmselect data=uno;
class tipovino;
model quality=freesulfur citric alcohol chlorides density volatile
totalsulfur sulphates pH sugar fixed
/selection=backward(select=SL ) sle=0.01 sls=0.01;
run;
```

Comparativo utilizando atras (backward) muy exigente

```
/*  Ahora con el metodo atras backward muy exigente:
/selection=backward(select=SL ) sle=0.001 sls=0.001;  */
proc glmselect data=uno;
class tipovino;
model quality=freesulfur citric alcohol chlorides density volatile
totalsulfur sulphates pH sugar fixed
/selection=backward(select=SL ) sle=0.001 sls=0.001;
run;
```

Una primera comparación con validación cruzada

Algo muy importante para que te funcione el código SAS a bajo, es que hay que ejecutar antes la macro %cruzada en el archivo macro cruzada regresion v 4.0.sas.

El modelo 1 tiene todas las variables cuantitativas y la variable categórica tipovino:

```
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur citric
alcohol chlorides density volatile totalsulfur sulphates pH sugar
fixed, listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final1;set final;modelo=1;
```

El modelo 2 tiene todas las variables cuantitativas, pero no tiene la variable categórica tipovino:

```
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur citric
alcohol chlorides density volatile totalsulfur sulphates pH sugar
fixed, listclass=,ngrupos=4,sinicio=12345,sfinal=12385); data
final2;set final;modelo=2;
```

El modelo 3 hemos quitado la variable citric de las variables cuantitativas y hemos dejado la variable categórica tipovino:

```
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur alcohol
chlorides density volatile totalsulfur sulphates pH sugar fixed,
listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final3;set final;modelo=3;

data union;set final1 final2 final3;run;
proc boxplot data=union;plot media*modelo;run;
```

En la figura a bajo se puede ver los resultados de los 3 modelos probados.

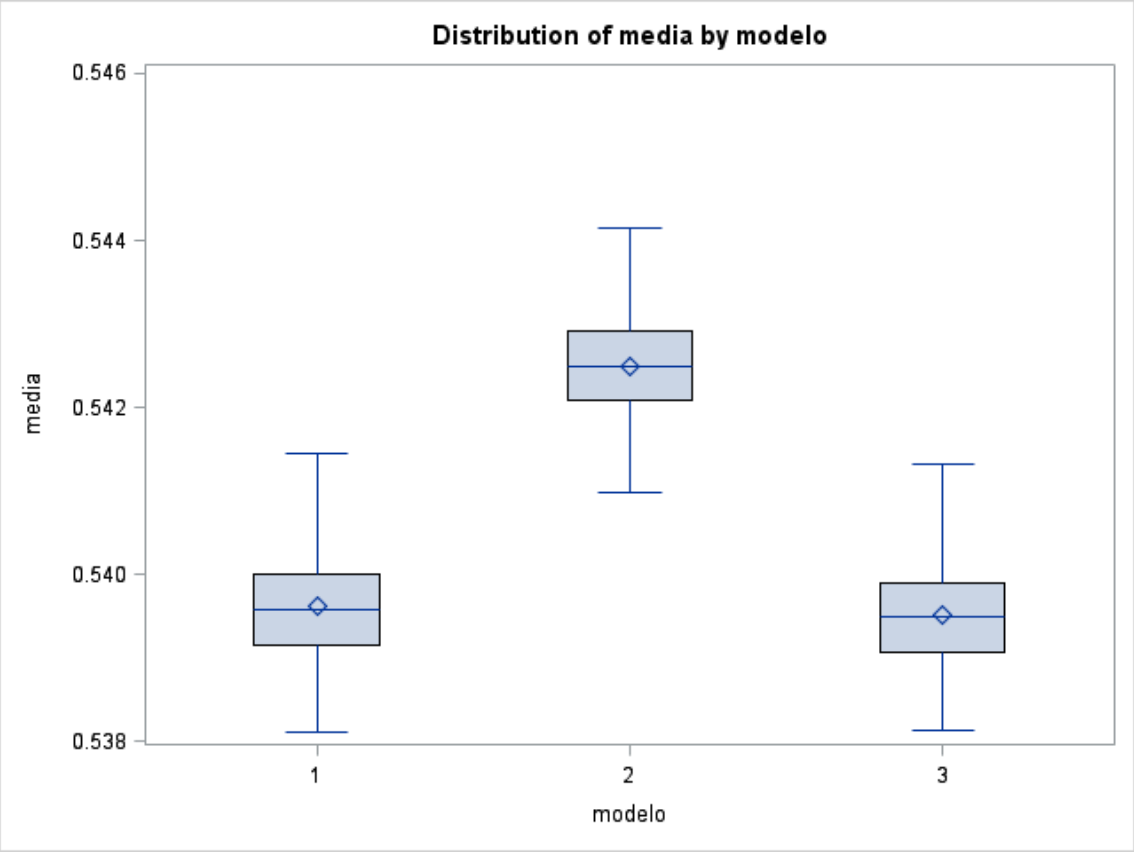

Figura 1: Comparativo entre los 3 modelos.

Ahora hacemos lo mismo pero añadiendo un 4 modelo donde dejamos menos variables solo las variables que nos ha sugerido en stepwise que hemos hecho en el inicio de todo donde nos ha dicho para quedar con el paso (step) 7. Las variables que se quedan son freesulfur alcohol chlorides volatile totalsulfur sulphates sugar.

```
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur citric
alcohol chlorides density volatile totalsulfur sulphates pH sugar
fixed, listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final1;set final;modelo=1;

%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur citric
alcohol chlorides density volatile totalsulfur sulphates pH sugar
fixed, listclass=,ngrupos=4,sinicio=12345,sfinal=12385); data
final2;set final;modelo=2;

%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur alcohol
chlorides density volatile totalsulfur sulphates pH sugar fixed,
listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final3;set final;modelo=3;

%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur alcohol
chlorides volatile totalsulfur sulphates sugar,
```

```
listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final4;set final;modelo=4;

data union;set final1 final2 final3 final4; run;
proc boxplot data=union;plot media*modelo;run;
```

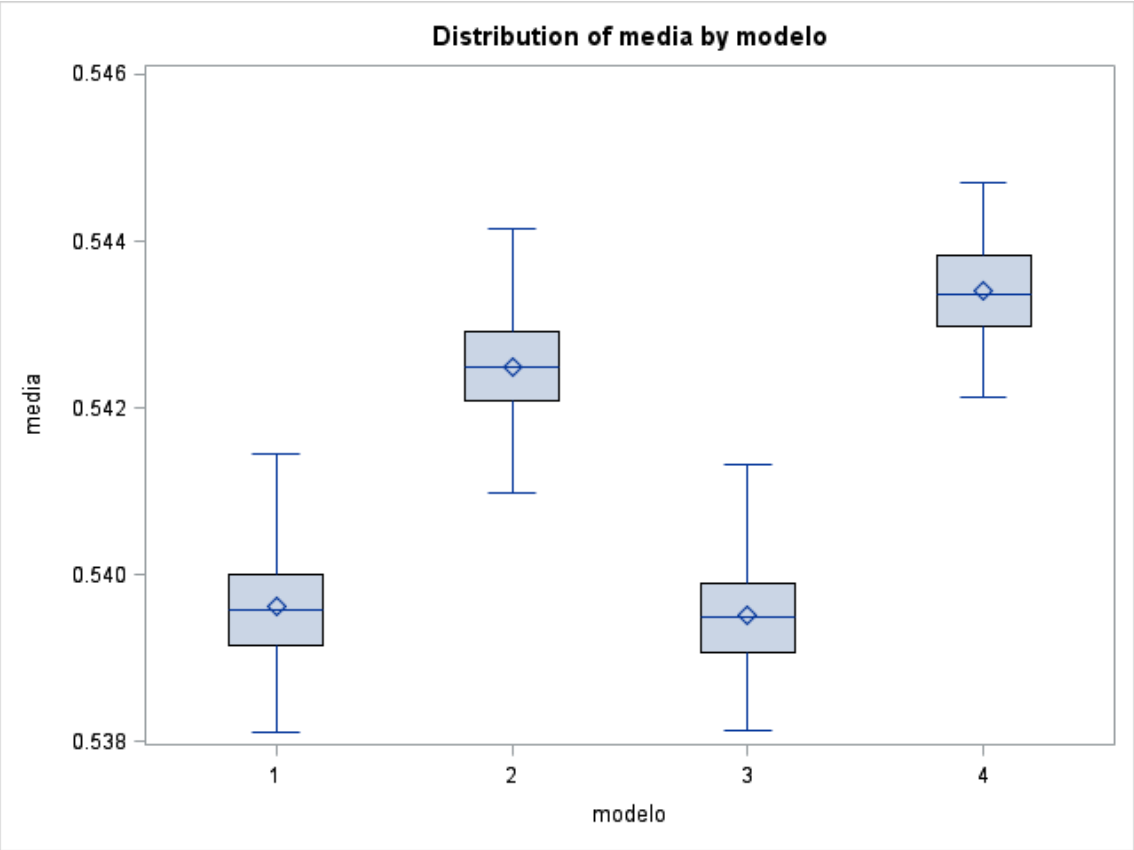

Figura 2: Comparativo entre los 4 modelos.

Los modelos más bajos y con menos longitud (que la caja no sea muy larga) son los mejores.

Se puede decir que los modelos 1 y 3 son los mejores.

Segunda parte: retocar-recodificar categóricas

El dataset de vinos solo tiene una variable categórica la variable tipovino y hay una buena distribución entre los valores red y white, no hay datos nulos y tampoco hay que retocar-recodificar esta variable.

No tenemos variables categóricas poco representadas, por esto no hace falta utilizar la macro AgruparCategorias para mejorar la variable tipovino, pero se quieres puedes probar con el código SAS a bajo por una cuestión de aprendizaje:

```
%AgruparCategorias(archivo=uno,vardep=quality,vardeptipo=I,listclass=t
ipovino,
criterio=,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\
);
```

Tercera parte: Incorporar interacciones

En esta parte se utiliza la macro interacttodo que se encuentra en el archivo interacttodo v 5.0.sas para …

Código SAS:

```
/*
Hay que ejecutar antes la macro %interacttodo en el archivo
interacttodo v 5.0.sas
*/
%interacttodo(archivo=uno,vardep=quality,
listclass=tipovino,listconti=freesulfur citric alcohol chlorides
density volatile totalsulfur sulphates pH sugar fixed,
interac=1,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\
);
```

En el log tenemos el listado de variables para copiar y pegar.

```
tipovino*alcohol alcohol density tipovino*density tipovino*volatile volatile
tipovino*chlorides
chlorides tipovino*totalsulfur tipovino*sulphates tipovino*pH tipovino
tipovino*freesulfur
tipovino*sugar tipovino*fixed tipovino*citric citric fixed freesulfur totalsulfur
sulphates sugar
pH
```

En el listado a bajo se puede ver las variables que aportan más y las que aportan menos.

Obs	variable	AIC	FValue	ProbF
1	tipovino*alcohol	3230.24869	851.26	<.0001
2	alcohol	3312.12141	1597.64	<.0001
3	density	4102.83268	670.31	<.0001
4	tipovino*density	4104.83237	335.10	<.0001
5	tipovino*volatile	4239.45036	261.64	<.0001
6	volatile	4265.29915	493.35	<.0001
7	tipovino*chlorides	4464.55228	142.16	<.0001
8	chlorides	4473.93211	272.50	<.0001
9	tipovino*totalsulfur	4534.84378	105.69	<.0001
10	tipovino*sulphates	4612.06032	66.08	<.0001
11	tipovino*pH	4618.57391	62.76	<.0001

Obs	variable	AIC	FValue	ProbF
12	tipovino	4647.78798	93.81	<.0001
13	tipovino*freesulfur	4657.35327	43.07	<.0001
14	tipovino*sugar	4658.32322	42.57	<.0001
15	tipovino*fixed	4672.50019	35.40	<.0001
16	tipovino*citric	4677.74241	32.76	<.0001
17	citric	4693.25278	47.87	<.0001
18	fixed	4702.58009	38.48	<.0001
19	freesulfur	4720.94087	20.04	<.0001
20	totalsulfur	4729.82011	11.14	0.0008
21	sulphates	4731.32739	9.63	0.0019
22	sugar	4732.06633	8.89	0.0029
23	pH	4738.48502	2.47	0.1159

Ahora probamos con las nuevas variables con interacciones.

```
/* selection=stepwise(select=AIC choose=AIC); */
proc glmselect data=uno;
class tipovino;
model quality=tipovino*alcohol alcohol density tipovino*density
tipovino*volatile volatile tipovino*chlorides
chlorides tipovino*totalsulfur tipovino*sulphates tipovino*pH tipovino
tipovino*freesulfur
tipovino*sugar tipovino*fixed tipovino*citric citric fixed freesulfur
totalsulfur sulphates sugar
pH
/selection=stepwise(select=AIC choose=AIC);
run;
```

The selected model, based on AIC, is the model at Step 12.

Effects: Intercept alcohol*tipovino density*tipovino volatile*tipovino chlorides*tipovino totalsulfur*tipovino sulphates*tipovino pH*tipovino tipovino freesulfur*tipovino sugar*tipovino fixed*tipovino pH

```
/* selection=stepwise(select=SBC choose=SBC); */
proc glmselect data=uno;
class tipovino;
model quality=tipovino*alcohol alcohol density tipovino*density
tipovino*volatile volatile tipovino*chlorides
chlorides tipovino*totalsulfur tipovino*sulphates tipovino*pH tipovino
tipovino*freesulfur
tipovino*sugar tipovino*fixed tipovino*citric citric fixed freesulfur
totalsulfur sulphates sugar
pH
/selection=stepwise(select=SBC choose=SBC);
run;
```

The selected model, based on SBC, is the model at Step 8.

Effects: Intercept alcohol*tipovino density volatile*tipovino freesulfur*tipovino fixed*tipovino sulphates sugar pH

Probamos nuevos modelos para ver los resultados.

```
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur citric
alcohol chlorides density volatile totalsulfur sulphates pH sugar
fixed, listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final1;set final;modelo=1;
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur citric
alcohol chlorides density volatile totalsulfur sulphates pH sugar
fixed, listclass=,ngrupos=4,sinicio=12345,sfinal=12385); data
final2;set final;modelo=2;
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur alcohol
chlorides density volatile totalsulfur sulphates pH sugar fixed,
listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final3;set final;modelo=3;
%cruzada(archivo=uno, vardepen=quality, listconti=freesulfur alcohol
chlorides volatile totalsulfur sulphates sugar,
listclass=tipovino,ngrupos=4,sinicio=12345,sfinal=12385); data
final4;set final;modelo=4;


%cruzada(archivo=uno, vardepen=quality, listconti=alcohol*tipovino
density*tipovino volatile*tipovino chlorides*tipovino
totalsulfur*tipovino sulphates*tipovino pH*tipovino tipovino
freesulfur*tipovino sugar*tipovino fixed*tipovino pH,
listclass=tipovino,ngrupos=4, sinicio=12345,sfinal=12385); data
final5;set final;modelo=5;
%cruzada(archivo=uno, vardepen=quality, listconti=alcohol*tipovino
density volatile*tipovino freesulfur*tipovino fixed*tipovino sulphates
sugar pH , listclass=tipovino,ngrupos=4, sinicio=12345,sfinal=12385);
data final6;set final;modelo=6;
title 'Modelos con y sin interacciones'; data union;set final1 final2
final3 final4 final5 final6;run; proc boxplot data=union;plot
media*modelo;run;
```

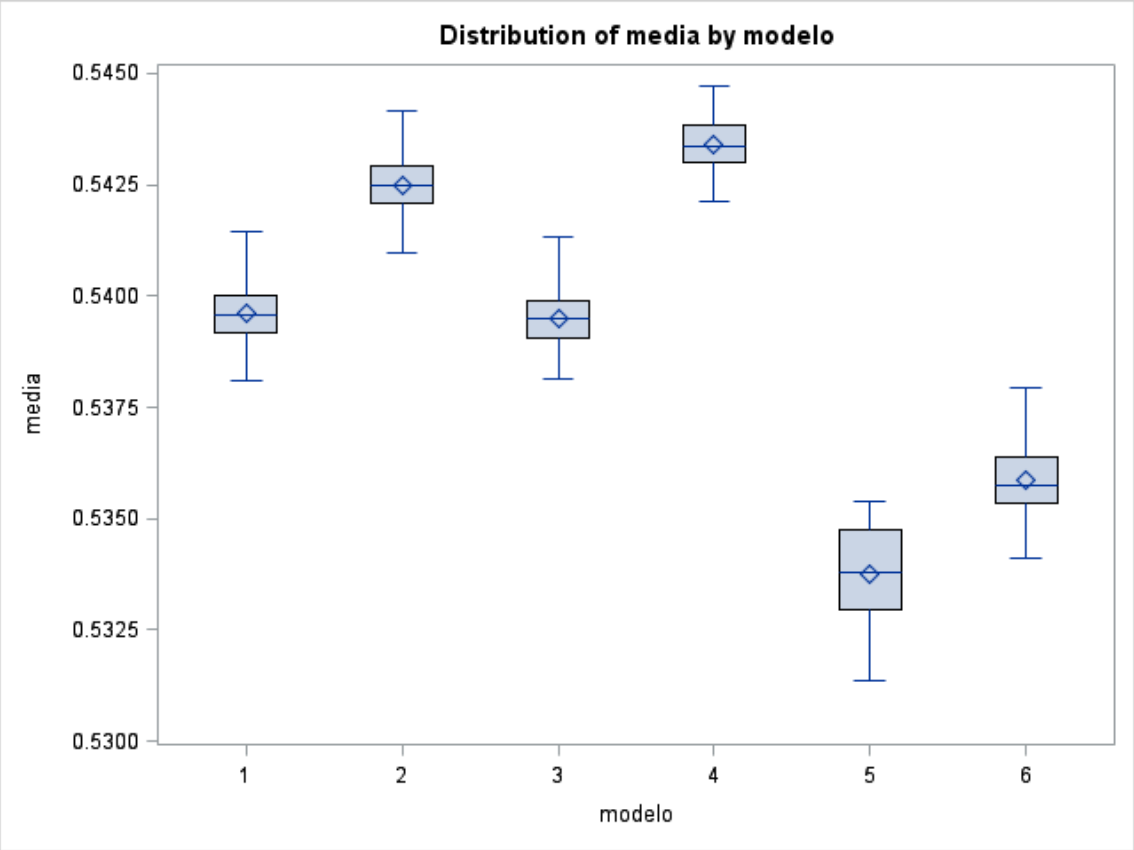

Figura 3: Comparativo entre los 6 modelos.

Se puede ver que el modelo 5 es el mejor hasta el momento.

Para observar mejor la tabla y parámetros de la regresión se puede ejecutar el código SAS a bajo para el modelo 5.

```
proc glm data=uno; class tipovino; model quality=alcohol*tipovino
density*tipovino volatile*tipovino chlorides*tipovino
totalsulfur*tipovino sulphates*tipovino pH*tipovino tipovino
freesulfur*tipovino sugar*tipovino fixed*tipovino pH /solution; run;
```

Cuarta parte: Refinamiento de las variables e interacciones: creación de dummys

La macro %nombresmodbien construye columnas de dummys e interacciones correspondientes a los efectos-categorías con más de un cierto número de observaciones. El archivo de salida pretest es una versión modificada del original.

```
data uno;set uno;if quality=. then delete;run; options mprint;
%nombresmodbien(archivo=uno,depen=quality,
modelo=alcohol*tipovino density volatile*tipovino freesulfur*tipovino
fixed*tipovino sulphates sugar pH
,listclass=tipovino,
listconti=freesulfur alcohol chlorides density volatile totalsulfur
sulphates pH sugar fixed,
corte=15,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\)
;

/* output
alcovinored alcovinowhi density fixenored fixenowhi freeipovinored
freeipovinowhi pH sugar
sulphates volaovinored volaovinowhi
*/

ods output SelectedEffects=efectos;/* Para crear un archivo con el
nombre de los efectos seleccionados */
proc glmselect data=pretest;
model quality=alcovinored alcovinowhi density fixenored fixenowhi
freeipovinored freeipovinowhi pH sugar
sulphates volaovinored volaovinowhi
/selection=stepwise(select=AIC choose=AIC);/* aqui se cambia la medida
*/ run;
data;set efectos; put effects @@;run;/* Para obtener en LOG la lista
de los efectos seleccionados */

/* output
alcovinored alcovinowhi density fixenored fixenowhi freeipovinored
freeipovinowhi pH sug
ar sulphates volaovinored volaovinowhi
*/


data uno;set uno;if quality=. then delete;run; options mprint;
%nombresmodbien(archivo=uno,depen=quality,
modelo=alcohol*tipovino density*tipovino volatile*tipovino
chlorides*tipovino totalsulfur*tipovino sulphates*tipovino pH*tipovino
tipovino freesulfur*tipovino sugar*tipovino fixed*tipovino pH
,listclass=tipovino,
listconti=freesulfur alcohol chlorides density volatile totalsulfur
sulphates pH sugar fixed,
corte=15,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\)
;

/* output
alcovinored alcovinowhi chlopovinored chlopovinowhi densvinored
densvinowhi fixenored fixenowhi
freeipovinored freeipovinowhi pH pHtipovinored pHtipovinowhi suganored
suganowhi sulppovinored
sulppovinowhi tipovinored tipovinowhi totatipovinored totatipovinowhi
volaovinored volaovinowhi
*/


ods output SelectedEffects=efectos;/* Para crear un archivo con el
nombre de los efectos seleccionados */
proc glmselect data=pretest;
```

```
model quality=alcovinored alcovinowhi chlopovinored chlopovinowhi
densvinored densvinowhi fixenored fixenowhi
freeipovinored freeipovinowhi pH pHtipovinored pHtipovinowhi suganored
suganowhi sulppovinored
sulppovinowhi tipovinored tipovinowhi totatipovinored totatipovinowhi
volaovinored volaovinowhi
/selection=stepwise(select=SBC choose=SBC);/* aqui se cambia la medida
*/ run;
data;set efectos; put effects @@;run;/* Para obtener en LOG la lista
de los efectos seleccionados */


%cruzada(archivo=pretest, vardepen=quality, listconti=alcovinored
alcovinowhi density fixenored fixenowhi freeipovinored freeipovinowhi
pH sugar sulphates volaovinored volaovinowhi,
listclass=,ngrupos=4,sinicio=12345,sfinal=12385); data final7;set
final;modelo=7;
%cruzada(archivo=pretest, vardepen=quality, listconti=alcovinored
alcovinowhi chlopovinored chlopovinowhi densvinored densvinowhi
fixenored fixenowhi freeipovinored freeipovinowhi pH pHtipovinored
pHtipovinowhi suganored suganowhi sulppovinored sulppovinowhi
tipovinored tipovinowhi totatipovinored totatipovinowhi volaovinored
volaovinowhi, listclass=,ngrupos=4,sinicio=12345,sfinal=12385); data
final8;set final;modelo=8;


title 'Modelos con nombresmod-dummys'; data union;set final1 final2
final3 final4 final5 final6 final7 final8;run; proc boxplot
data=union;plot media*modelo;run;
```

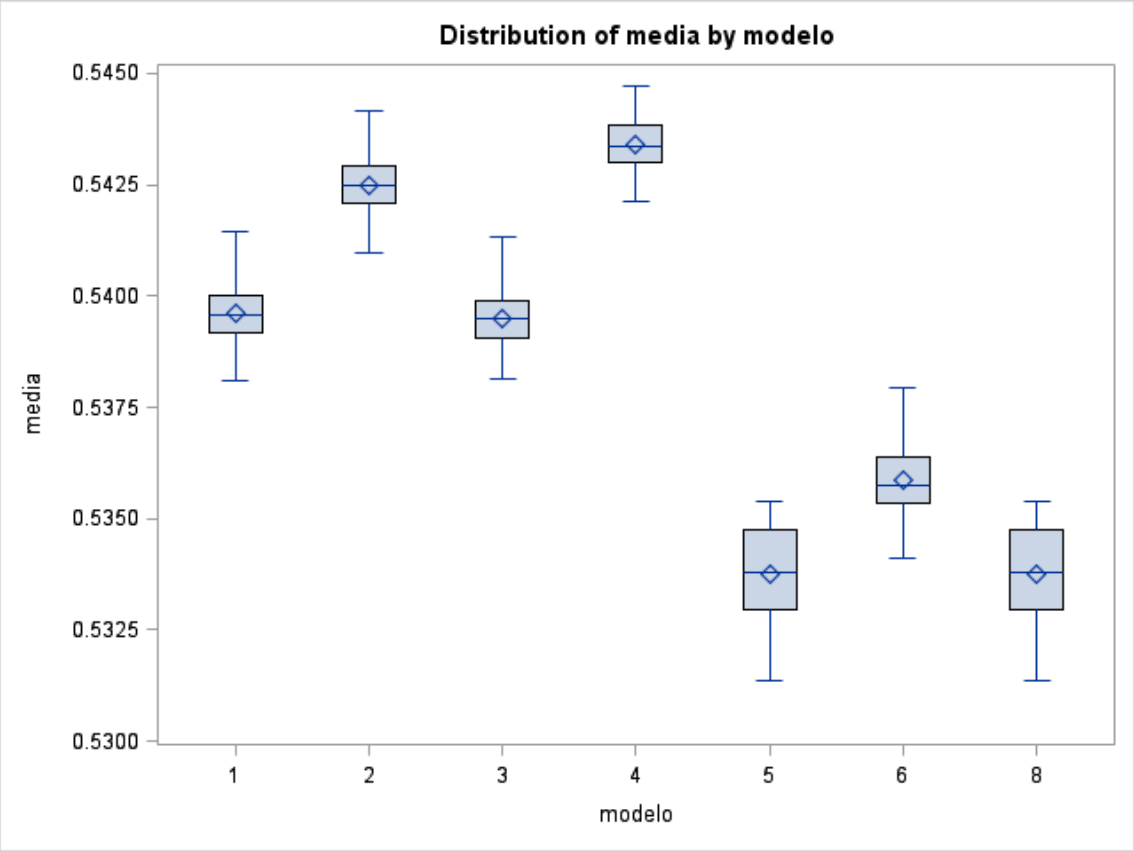

Figura 4: Comparativo entre los 8 modelos, porque no me ha salido el modelo 7?

Quinta parte: Sensibilidad de los modelos escogidos vía stepwise a alteraciones en los datos

En esta parte intentamos crear nuevos modelos utilizando la macro randonselect.

```
/*
Hay que ejecutar antes la macro %randomselect en el archivo macro
randomselect v 4.0.sas
*/


/*
randonselect con las variables del modelo 7
*/

%randomselect(data=pretest, listclass=, vardepen=quality,
modelo=alcovinored alcovinowhi density fixenored fixenowhi
freeipovinored freeipovinowhi pH sugar sulphates volaovinored
```

```
volaovinowhi,criterio=SBC,sinicio=12345,sfinal=12700,fracciontrain=0.9
0,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\);


/* modelos para probar - modelo=9
Intercept alcovinored alcovinowhi chlopovinored fixenowhi
freeipovinowhi suganowhi sulppovinored sulppovinowhi totatipovinored
volaovinored volaovinowhi
*/

/*
randonselect con las variables del modelo 8
*/

%randomselect(data=pretest, listclass=, vardepen=quality,
modelo=alcovinored alcovinowhi chlopovinored fixenowhi freeipovinowhi
suganowhi sulppovinored sulppovinowhi totatipovinored volaovinored
volaovinowhi,criterio=SBC,sinicio=12345,sfinal=12700,fracciontrain=0.9
0,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\);

/*  modelos para probar - modelo=10,11 e 12
Intercept alcovinored alcovinowhi chlopovinored fixenowhi
freeipovinowhi suganowhi sulppovinored sulppovinowhi totatipovinored
volaovinored volaovinowhi
Intercept alcovinored alcovinowhi fixenowhi freeipovinowhi suganowhi
sulppovinored sulppovinowhi totatipovinored volaovinored volaovinowhi
Intercept alcovinored alcovinowhi chlopovinored fixenowhi
freeipovinowhi suganowhi sulppovinored sulppovinowhi volaovinored
volaovinowhi

*/


%cruzada(archivo=pretest, vardepen=quality, listconti=alcovinored
alcovinowhi chlopovinored fixenowhi freeipovinowhi suganowhi
sulppovinored sulppovinowhi totatipovinored volaovinored volaovinowhi,
listclass=,ngrupos=4,sinicio=12345,sfinal=12346); data final9;set
final;modelo=9;

%cruzada(archivo=pretest, vardepen=quality, listconti=alcovinored
alcovinowhi chlopovinored fixenowhi freeipovinowhi suganowhi
sulppovinored sulppovinowhi totatipovinored volaovinored volaovinowhi,
listclass=,ngrupos=4,sinicio=12345,sfinal=12346); data final10;set
final;modelo=10;
%cruzada(archivo=pretest, vardepen=quality, listconti=alcovinored
alcovinowhi fixenowhi freeipovinowhi suganowhi sulppovinored
sulppovinowhi totatipovinored volaovinored volaovinowhi,
listclass=,ngrupos=4,sinicio=12345,sfinal=12346); data final11;set
final;modelo=11;
%cruzada(archivo=pretest, vardepen=quality, listconti=alcovinored
alcovinowhi chlopovinored fixenowhi freeipovinowhi suganowhi
sulppovinored sulppovinowhi volaovinored volaovinowhi,
listclass=,ngrupos=4,sinicio=12345,sfinal=12346); data final12;set
final;modelo=12;

title 'Modelos con randomselect tambien'; data union;set final1 final2
final3 final4 final5 final6 final7 final8 final9 final10 final11
final12;run;
proc boxplot data=union;plot media*modelo;run;
```

Algo he hecho mal que no me ha salido el plot con los modelos de 1 a 12.

Sexta parte: Trabajar con la variable dependiente transformada

Esta parte no me ha funcionado.

```
/*
Esta parte no he entendio mucho como hacer y por que

data uno; set trabajo2.Vino; run;


*/


data pretest2;set pretest;logquality=log(quality); logpH=log(pH);

/*  List all the variables in the dataset */
proc contents data=pretest2 out=sa;
data;set sa;if _n_=1 then put 'LISTA DE VARIABLES CONTINUAS';if type=1
then put name @@;run;

/*
alcovinored alcovinowhi chlopovinored chlopovinowhi densvinored
densvinowhi fixenored fixenowhi freeipovinored freeipovinowhi logpH
logquality pH pHtipovinored pHtipovinowhi quality suganored suganowhi
sulppovinored sulppovinowhi tipovinored tipovinowhi
totatipovinored totatipovinowhi volaovinored volaovinowhi
*/

proc print data=pretest2;run;


proc gplot data=pretest2;
plot quality*pH logquality*pH logquality*logpH; run;
```

Séptima parte: Creación tentativa de otras variables

Crear variables con log para ver se mejora el modelo.

Para no tener que cambiar una por una las variables continuas a logaritmo se utilizan arrays.

Las interacciones no se transforman de momento pero podría hacerse (en el archivo pretest).

Las continuas, por simplificar y evitar el problema del log(0), se transforman a log(x+1):

```
/*
Hago la transformacion del dataset pretest para logaritimo.
*/


data pretest2 (drop=i);
array x{26} alcovinored alcovinowhi chlopovinored chlopovinowhi
densvinored densvinowhi fixenored fixenowhi freeipovinored
freeipovinowhi logpH logquality pH pHtipovinored pHtipovinowhi quality
suganored suganowhi sulppovinored sulppovinowhi tipovinored
tipovinowhi
totatipovinored totatipovinowhi volaovinored volaovinowhi;
array z{26}; set pretest;logquality=log(quality);
do i=1 to 26;z{i}=log(x{i}+1);end;
run;

/*
Hago la transformacion del dataset original (uno) para logaritimo.
*/

data vinolog (drop=i);
array x{11} freesulfur citric alcohol chlorides density volatile
totalsulfur sulphates pH sugar fixed;
array z{11}; set uno;logquality=log(quality);
do i=1 to 11;z{i}=log(x{i}+1);end;
run;
```

Probamos

```
proc glmselect data=pretest2;
model logquality= z1-z26 alcovinored alcovinowhi chlopovinored
chlopovinowhi densvinored densvinowhi fixenored fixenowhi
freeipovinored freeipovinowhi logpH logquality pH pHtipovinored
pHtipovinowhi quality suganored suganowhi sulppovinored sulppovinowhi
tipovinored tipovinowhi
totatipovinored totatipovinowhi volaovinored volaovinowhi
/selection=stepwise(select=sbc choose=sbc);
run;
```

```
proc glmselect data=vinolog;
model logquality= z1-z11 freesulfur citric alcohol chlorides density
volatile totalsulfur sulphates pH sugar fixed
/selection=stepwise(select=sbc choose=sbc);
run;
```

```
%cruzadabis(archivo=pretest2, vardepen=logquality, listconti=z1-z26
alcovinored alcovinowhi chlopovinored chlopovinowhi densvinored
densvinowhi fixenored fixenowhi freeipovinored freeipovinowhi logpH
logquality pH pHtipovinored pHtipovinowhi quality suganored suganowhi
sulppovinored sulppovinowhi tipovinored tipovinowhi
totatipovinored totatipovinowhi volaovinored volaovinowhi ,
listclass=, ngrupos=4,sinicio=12345,sfinal=12385); data final13;set
final;modelo=13;

%cruzadabis(archivo=vinolog, vardepen=logquality, listconti=z1-z11
freesulfur citric alcohol chlorides density volatile totalsulfur
sulphates pH sugar fixed , listclass=,
ngrupos=4,sinicio=12345,sfinal=12385); data final13;set
final;modelo=14;
```

```
title 'Modelos con randomselect tambien'; data union;set final1 final2
final3 final4 final5 final6 final7 final8 final9 final10 final11
final12 final13 final14;run; proc boxplot data=union;plot
media*modelo;run;
```

No me ha salido el grafico:

Error:

```
WARNING: No output destinations active.
NOTE: Processing beginning for PLOT statement number 1.
NOTE: There were 295 observations read from the data set WORK.UNION.
NOTE: PROCEDURE BOXPLOT used (Total process time):
      real time           0.04 seconds
      cpu time            0.03 second
```

Solo con esto también no me ha funcionado:

```
title 'Modelos con randomselect tambien'; data union;set final1 final2
final3 final4 final14;run; proc boxplot data=union;plot
media*modelo;run;
```

Creo que hay un error en el modelo final14.

3. Kaggle

Una de las partes de este trabajo era intentar de predecir los precios de los apartamientos en el concurso creado por el Prof. Javier en la página Kaggle.

https://inclass.kaggle.com/c/housing-price-prediction

Para este ejercicio he probado quitar y añadir variables, utilizar la macro intere, normalizar las variables, crear nuevas variables con log, etc.

El mejor modelo que he conseguido sacar tiene un resultado de:

24 ↓2 caiomoreno 132274.02394 10 Mon, 18 Jan 2016 20:48:15 (-44.6h)

Enlace para el Leaderboard:

https://inclass.kaggle.com/c/housing-price-prediction/leaderboard

Código SAS para cargas los datos:

```
/* Carga los datos SAS en la libreria disco que estan en la carpeta */
libname discoc 'C:\Users\win\Documents\GitHub\ucm\semma\kaggle';

data union;set discoc.preditrain discoc.testalumnos;run;
```

Código SAS para normalizar los datos:

```
proc standard data=union out=salida_standard mean=0 std=1; var age
baths beds lot sqft tax;
run;
```

Código SAS para generar variables nuevas:

```
data union; set union;
      logatax=log(tax);
      expotax=tax**2;
      logsqft=log(sqft);
      expqft=sqft**2;
      logage=log(age);
      expage=age**2;
      logbaths=log(baths);
      expbaths=baths**2;
      logbeds=log(beds);
      expbeds=beds**2;
run;
```

Código SAS para ejecutar el mejor modelo que he conseguido:

```
/* Root MSE = 128982.9 */
proc glm data=salida_standard;
      model price=age baths beds expotax logatax lot sqft tax logsqft
expqft logage expage logbaths expbaths logbeds expbeds
logsqft*expqft*logage*expage*logbaths*expbaths*logbeds*expbeds
logsqft*expqft*expotax*logatax logsqft*expqft
beds*expotax*logatax*lot*sqft*tax
age*baths*beds*expotax*logatax*lot*sqft*tax
expotax*logatax*lot*sqft*tax age*baths*beds*sqft tax*lot*sqft tax*lot
age*lot tax*sqft sqft*baths*beds baths*beds expotax*logatax;
      output out=salida p=predi;
run;
```

Código SAS para generar el fichero en .csv para enviar a Kaggle.

```
data salida;set salida;if price=. then output;run;
data ;set salida;
file 'C:\Users\win\Documents\GitHub\ucm\semma\kaggle\Submit-18enero16-
01-attempt.csv' dlm=',';
if _n_=1 then put 'Id,Prediction';
put id predi;
run;
```

Código SAS con la macro **%*interacttodo***

```
%interacttodo(archivo=union_original,vardep=price,
listclass= ,listconti=beds baths sqft lot age tax,
interac=1,directorio=C:\Users\win\Documents\GitHub\ucm\semma\trabajo2\
);


/* output interacttodo

tax sqft baths beds age lot

*/
```

Yo he probado muchísimos otros modelos pero no voy poner en este trabajo para no quedar demasiado grande el trabajo.

Envió también junto el código SAS con las muchas pruebas, no está muy organizado pero con un poco de conocimiento de SAS se puede probar los distintos modelos caso sea necesario.