Actividad 1

En esta actividad se presenta un estudio sobre regresiones con el programa R. La base de datos se puede descargar del siguiente repositorio de GitHub.

1. Base de datos

La base de datos seleccionada para esta actividad se puede encontrar en el portal de datos abiertos "Open Data" del estado de Connecticut de Estados Unidos. En particular, se selecciona el conjunto de datos ***Real Estate Sales 2001-2020 GL***[1]. Consta de un listado de todas las ventas de bienes inmuebles se produjeron entre los años 2001 y 2020. El conjunto de datos está formado por un total de 997.213 entradas y 14 columnas; y cada entrada representa una venta. A continuación se detalla cada una de las columnas:

1. **Serial Number**: Número de serie (Número).
2. **List Year**: Año en que la propiedad se puso en venta (Número).
3. **Date Recorded**: Fecha en que se registró la venta localmente (Fecha y Hora).
4. **Town**: Nombre de la ciudad (Texto).
5. **Address**: Dirección (Texto).
6. **Assessed Value**: Valor de la propiedad utilizado para la evaluación de impuestos locales (Número).
7. **Sale Amount**: Valor por el cual se vendió la propiedad (Número).
8. **Sales Ratio**: Proporción del precio de venta respecto al valor tasado (Texto).
9. **Property Type**: Tipo de propiedad, incluyendo: Residencial, Comercial, Industrial, Apartamentos, Vacante, etc. (Texto).
10. **Residential Type**: Indica si la propiedad es residencial unifamiliar o multifamiliar (Texto).
11. **Non Use Code**: Código de venta que indica que el precio de venta no es confiable para determinar el valor de una propiedad (Texto).
12. **Assessor Remarks**: Observaciones del tasador (Texto).
13. **OPM remarks**: Observaciones de OPM (*Other People's Money*) (Texto).
14. **Location**: Coordenadas de latitud / longitud (Coordenadas).

2. Objetivo

Una vez analizados los datos, se establece como objetivo la **predicción del precio de venta del inmueble**, representado por la variable "Sale.Amount", a partir de varias características del conjunto de datos, como la tasación inicial del inmueble.

Para realizar esta predicción se hace un estudio de varios tipos de regresión: regresión lineal, regresión lineal multidimensional, regresión polinómica multidimensional, regresión ridge (L2), regresión lasso (L1), regresión Elastic-Net (L1 y L2), y finalmente, una prueba de regresión logística.

---

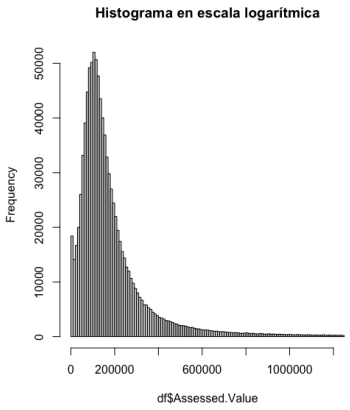
[1] https://data.ct.gov/Housing-and-Development/Real-Estate-Sales-2001-2020-GL/5mzw-sjtu

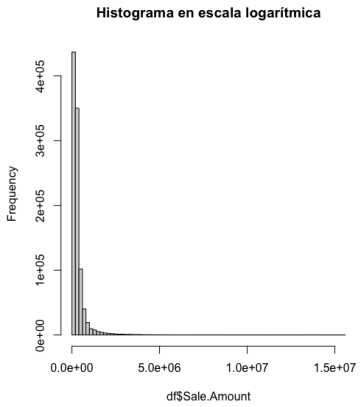
3. Análisis y procesado

En primer lugar se realiza una visualización de los datos para determinar el tipo y formato de cada columna. A partir de este estudio se pueden observar un total de **997.213 entradas**, dónde cada columna presenta características estadísticas muy distintas.

Se estudia el número total de valores faltantes inicial, así como una primera visualización de la distribución de las columnas numéricas más relevantes (“Sale.Amount” y “Assessed.Value”):

	NA_Count
Serial.Number	0
List.Year	0
Date.Recorded	2
Town	0
Address	51
Assessed.Value	0
Sale.Amount	0
Sales.Ratio	0
Property.Type	382446
Residential.Type	388309
Non.Use.Code	707532
Assessor.Remarks	847349
OPM.remarks	987279
Location	799516





(a) Valores NA, (b) Histograma de “Assessed.Value”, y (c) Histograma de “Sale.Amount“

1. Para realizar un estudio acotado y razonable sobre la regresión, se realiza una primera limpieza de los datos, dónde **se eliminan las columnas no relevantes**. Las columnas “Non.Use.Code", "Assessor.Remarks", "OPM.remarks" y "Location" se eliminan debido a la gran cantidad de valores faltantes, respecto al número total de columnas. Por otro lado, las columnas “Date.Recorded” y "Serial.Number" no aportan información relevante para la predicción, ya que el año de venta ya está presente en otra variable. Las columnas de “Town" y “Address" son eliminadas para la simplificación de la actividad, ya que requieren de un procesamiento de NLP adicional no relevante para esta actividad. Finalmente, "Sales.Ratio" es eliminada ya que no es una variable que podamos usar para la predicción, ya que precisamente representa la relación entre una de las variables predictoras (“Assessed.Value”) y la variable a predecir (“Sale.Amount”). Podemos observar el estado del data-frame y el número de valores NA.

	List.Year	Assessed.Value	Sale.Amount	Property.Type	Residential.Type
1	2020	133000	248400	Residential	Single Family
2	2020	110500	239900	Residential	Three Family
3	2020	150500	325000	Commercial	<NA>
4	2020	127400	202500	Residential	Two Family
5	2020	217640	400000	Residential	Single Family
6	2020	528490	775000	Residential	Single Family

	NA_Count
List.Year	0
Assessed.Value	0
Sale.Amount	0
Property.Type	371891
Residential.Type	377282

2. Se procede a la **eliminación de las filas con elementos faltantes**, así como de la eliminación de aquellos inmuebles con valores atípicos, dónde el precio de venta es de $0.

3. También se procede a **transformar las variables categóricas a variables ficticias o *dummies*** para una correcta interpretación de las regresiones lineales. Con este cambio

obtenemos las siguientes columnas: "List.Year", "Assessed.Value", "Sale.Amount", “Property.Type_Four Family", "Property.Type_Residential", "Property.Type_Single Family", "Property.Type_Three Family", "Property.Type_Two Family", "Residential.Type_Four Family" "Residential.Type_ Single Family", “Residential.Type_Three Family", "Residential.Type_Two Family".
Este proceso permite **evitar la multicolinealidad**, ya que si tenemos k niveles de una variable categórica y creamos k nuevas variables “dummies”, podemos caer en la trampa de variables “dummies”. Esta trampa se observa cuando una variable puede predecirse exactamente por el valor de otras variables (multicolinealidad). Si bien la multicolinealidad no afecta al estimador directamente, lo hace impreciso y poco fiable, ya que puede ser difícil determinar cómo las variables independientes influyen individualmente en la variable dependiente. Por lo tanto, necesitamos excluir una variable ficticia mientras construimos el modelo de regresión. Como resultado, para k niveles de una variable categórica, necesitamos crear k-1 variables ficticias.

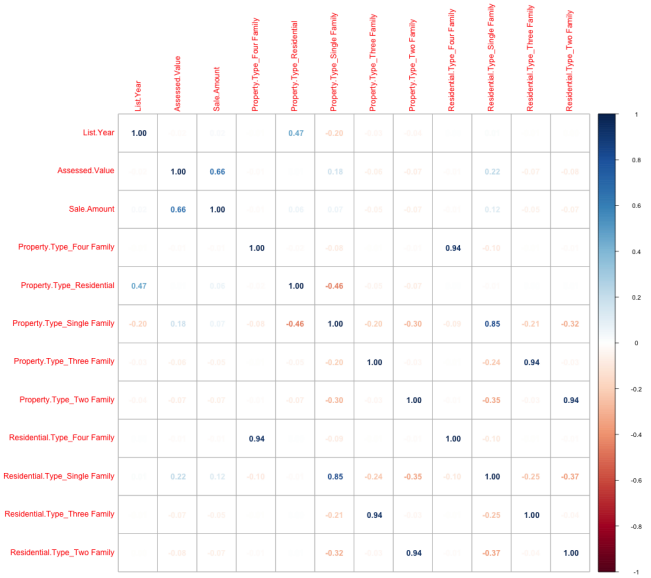
4. A continuación, se procede a **convertir la variable “List.Year”**, que contiene el año de venta (de 2001 a 2022) **a una lista de enteros del 1 al 15**, dónde cada año es asignado un número entero (por ejemplo, 2006 → 1, 2007 → 2, etc.). Esta conversión permite introducir el concepto de inflación a los datos, ya que a medida que avanza el tiempo los precios tienden a ascender.

5. Se procede a la **estandarización de las variables numéricas** “Assessed.Value” y “Sale.Amount” a partir de la media y desviación estándar de “Sale.Amount”. Se utilizan estos valores para mantener la homogeneidad entre las dos variables, ya que son representadas en la misma escala. Una vez realizado este procesado inicial obtenemos el siguiente data-frame:

	List.Year	Assessed.Value	Sale.Amount	Property.Type_Four Family	Property.Type_Residential	Property.Type_Single Family	Property.Type_Three Family	Property.Type_Two Family
353507	9	-1.0437642	-0.8117611	0	0	0	0	0
294541	8	-0.2426689	-0.2334028	0	0	0	0	0
35808	15	-0.5629545	-0.3314296	0	1	0	0	0
291942	7	-0.5920447	-0.4210541	0	0	0	0	0
145426	3	0.7191265	0.3827659	0	0	0	0	0
78582	1	0.1121028	0.1951145	0	0	0	0	0

	Residential.Type_Four Family	Residential.Type_Single Family	Residential.Type_Three Family	Residential.Type_Two Family
353507	0	0	0	0
294541	0	0	0	0
35808	0	0	0	0
291942	0	0	0	0
145426	0	0	0	0
78582	0	0	0	0

6. A continuación se analizan las **correlaciones** y **multicolinealidad**, dónde se pueden observar algunas correlaciones elevadas entre las variables *dummy* de “Residential.Type” y “Property.Type” cuando tienen el mismo número de habitaciones. En este caso, no se elimina ninguna variable ya que la correlación máxima es de 0.94 y podríamos estar perdiendo información.



7. Antes del entrenamiento se hace un procesado de los datos en el que **se mezclan los datos de forma aleatoria** para evitar el orden temporal ascendiente de los precios. A continuación se crea un conjunto de train (478255 filas) y de test (119562 filas).

8. Finalmente, **se definen las funciones para automatizar el entrenamiento, creación de gráficas y evaluación de resultados** con las funciones “perform_regression_and_save_plots” y “eval_results”.

4. Regresiones y resultados

Tal y como se ha descrito anteriormente, se procede a la **predicción de la variable “Sale.Amount”** a partir de las otras variables predictoras.

4.1. Regresión logística

Para el caso de la regresión logística unidimensional, se procede a la predicción con la variable “Assessed.Amount”. En este caso obtenemos los siguientes resultados para el conjunto de train (izquierda) y test (derecha):

```
R-squared: 0.6710605
Mean Absolute Error: 61061.09
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 92988.83
```

```
R-squared: 0.4256257
Mean Absolute Error: 78869.92
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 275956.6
```

Se puede observar *overfitting*, y un error medio absoluto de $78869.92 en la predicción de test. Este error es muy relevante en la venta de inmuebles, ya que es un margen que ser’ia inaceptable. Se obtiene una pendiente de 0.62, y un *intercept* de -0.032, lo que nos indica que la relación entre las dos variables se encuentra en un rango de valores similar. Finalmente, podemos observar MAPE como infinito, la cual puede no ser una buena medida, ya que indica que los valores originales son muy pequeños o tienden a 0.

4.2. Regresión logística multidimensional

En este caso se utilizan el resto de columnas para la predicción de “Sale.Amount”. Obtenemos los siguientes resultados para el conjunto de train (izquierda) y test (derecha):

```
R-squared: 0.688558
Mean Absolute Error: 58967.91
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 90481.83
```

```
R-squared: 0.4303467
Mean Absolute Error: 76722.86
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 274806
```

Podemos observar una mejora en los resultados pero tenemos las mismas características que con la regresión lineal anterior.

4.3. Regresión polinómica multidimensional

Se considera el caso en que las relaciones no sean lineales, y para ello se usa una regresión polinómica de grado 2. Se obtienen los siguientes resultados:

```
R-squared: 0.7073064
Mean Absolute Error: 56682.56
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 87716.12
```

```
R-squared: 0.4339741
Mean Absolute Error: 75111.76
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 273974.7
```

Podemos observar una mejora en los resultados, ya que se aprecia un R2 de 0.7 para el conjunto de train, lo que indica cómo las variables independientes (predictores) explican la variabilidad en la variable dependiente. Aun así, continuamos con un claro *overfitting*.

4.4. Regresión ridge (L2)

La regresión lineal funciona seleccionando coeficientes para cada variable independiente que minimizan una función de pérdida. Sin embargo, si los coeficientes son grandes, pueden provocar *overfitting*, y dicho modelo no generalizará bien en los datos de test. Para superar esta deficiencia, utilizamos regularización, la cual penaliza los coeficientes grandes.

```
R-squared: 0.6885477
Mean Absolute Error: 59004.03
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 90483.84
```

```
R-squared: 0.4303135
Mean Absolute Error: 76777.43
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 274843.2
```

Se puede observar el efecto de la regularización, y cómo el conjunto de train ha aumentado su error. Desafortunadamente, el conjunto de test no ha mostrado mejora con respecto a la regresión polinómica ni multidimensional, pero sí respecto a la regresión lineal inicial.

4.5. Regresión lasso (L1)

Se utiliza el valor absoluto de los coeficientes para determinar la penalización, a diferencia de la penalización ridge, que utiliza la segunda norma del vector de coeficientes. Obtenemos los siguientes resultados para el conjunto de train (izquierda) y test (derecha):

```
R-squared: 0.6839884
Mean Absolute Error: 69391.03
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 96968.87
```

```
R-squared: 0.4288045
Mean Absolute Error: 89707.1
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 281394.7
```

Podemos observar una reducción de overfitting, pero no se han mejorado los resultados del conjunto de test.

4.6. Elastic-Net (L1+ L2)

Para esta regresión se realiza un proceso iterativo que permite determinar el mejor valor para *alpha* y *lambda*. Una vez establecidos estos valores obtenemos los resultados:
Podemos observar una mejora considerable en la reducción de overfitting, pero aun así el

```
R-squared: 0.6839884
Mean Absolute Error: 69391.03
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 96968.87
```

```
R-squared: 0.4288045
Mean Absolute Error: 89707.1
Mean Absolute Percentage Error: Inf
Root Mean Squared Error: 281394.7
```

mejor resultado se ha obtenido con la regresión polinómica de grado 2. Esto indica que, si bien la regularización debe ser usada para reducir el overfitting, **las relaciones entre la variable a predecir y las variables predictoras parecen no ser lineales**.

4.7. Regresión logística

Se utiliza una variable categórica binaria aleatoria (“Property.Type_Residential”) para testear la regresión logística. Cómo se trata de una variable dummy, el modelo puede predecir el resultado con un 100% de exactitud. A continuación se muestra la matriz de confusión:

binary_predictions	0	1
0	107671	0
1	0	11891

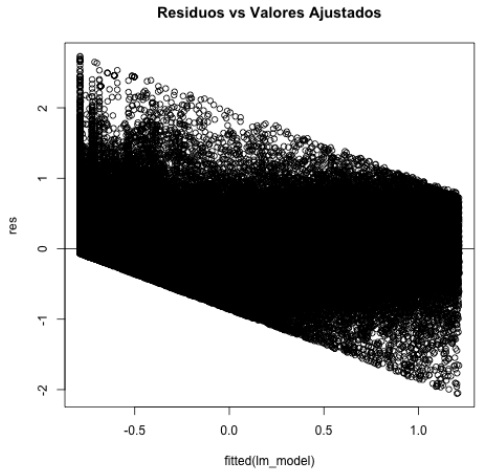
Matriz de confusión

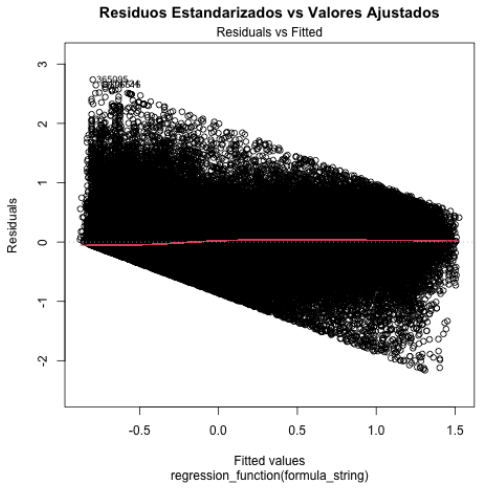
Finalmente, tanto para la regresión logística unidimensional, multidimensional y polinómica, automáticamente se generan y almacenan las gráficas de "*Residuos vs. Valores Ajustados*", "*Residuos Estándarizados vs. Valores Ajustados*", "*Q-Q*", "*Escala-Localización*" y "*Palanca*":

1. Gráfico de Residuos vs. Valores Ajustados:
Este gráfico compara los valores ajustados (predichos) por el modelo con los residuos (diferencias entre los valores observados y los valores predichos). La línea horizontal en el centro (abline) es una referencia para verificar si los residuos están distribuidos aleatoriamente alrededor de cero. Hay que recordar que el eje X está en el rango de datos estandarizado, y no el rango original, razón por la que las predicciones van de 0 a 5.

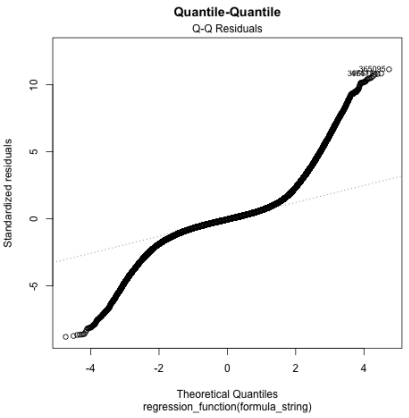
El patrón diagonal negativo que se observa sugiere que, en general, a medida que las predicciones aumentan, los residuos disminuyen en magnitud, lo que podría indicar que el modelo tiende a subestimar las observaciones con valores más altos y sobrestimar las observaciones con valores más bajos. Esto coincide con la teoría anterior de que existe una relación de no linealidad. Una solución a este problema, podría ser el **añadir variables que aporten información sobre el tamaño del inmueble**, ya que se podría establecer con mayor precisión si se trata de un inmueble de más alto o bajo valor. A pesar de la tendencia en diagonal, podemos ver que la mayor parte de valores se encuentran alrededor del 0, siguiendo una distribución normal.

2. Gráfico de Residuos Estándarizados vs. Valores Ajustados:
Este gráfico es similar al anterior, pero los residuos están estandarizados para verificar si siguen una distribución constante a lo largo de los valores ajustados. También se muestra una línea suave que indica la tendencia general de los residuos.



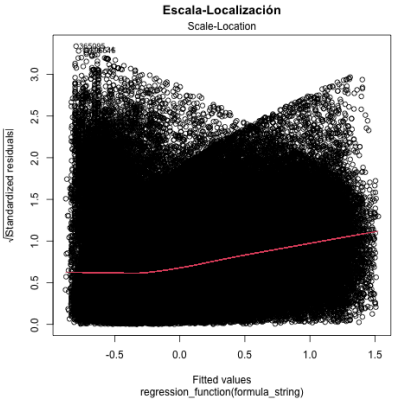


3. Gráfico Q-Q (Quantile-Quantile):
El gráfico Q-Q compara la distribución de los residuos con una distribución normal teórica. Si los puntos siguen aproximadamente una línea recta, sugiere que los residuos se distribuyen de manera normal. Desviaciones de la línea indican desviaciones de la normalidad.



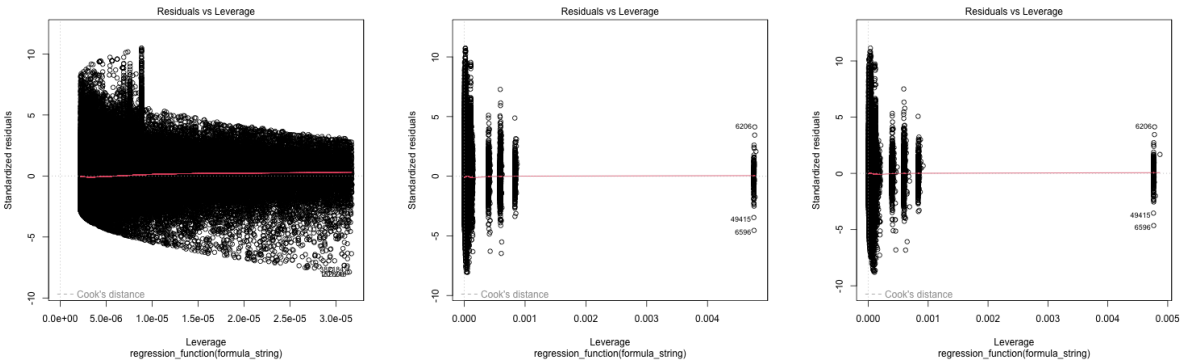
En este caso podemos observar que los datos no siguen una línea recta, y los gráficos resultantes parecen reforzar la idea de que el modelo tiende a subestimar las observaciones con valores más altos y sobrestimar las observaciones con valores más bajos.

4. Gráfico de Escala-Localización:
Este gráfico muestra la raíz cuadrada de los residuos estandarizados frente a los valores ajustados. Puede ayudar a detectar heteroscedasticidad en los residuos (es decir, la varianza de los errores no es constante en todos los niveles de los valores ajustados).



Podemos observar una varianza muy variable, principalmente para las predicciones en los rangos inferior y superior. Con la regresión polinómica podemos ver como esta tendencia sigue existiendo, pero tiende a disminuir.

5. Gráfico de Palanca (Leverage):
El gráfico de palanca muestra la influencia de cada observación en el modelo. Las observaciones con una alta palanca pueden influir en gran medida en la estimación de los coeficientes. Es una forma de identificar valores atípicos en términos de su influencia en el modelo. Podemos observar los resultados para la regresión lineal (izquierda), multidimensional (centro) y polinómica (derecha):



En este caso, no existe ninguna muestra particular que tenga una gran influencia sobre el modelo. Esto se puede deber al gran número de muestras del modelo y a la limpieza de datos realizada.