# Guía de Trabajos Prácticos Nº 6

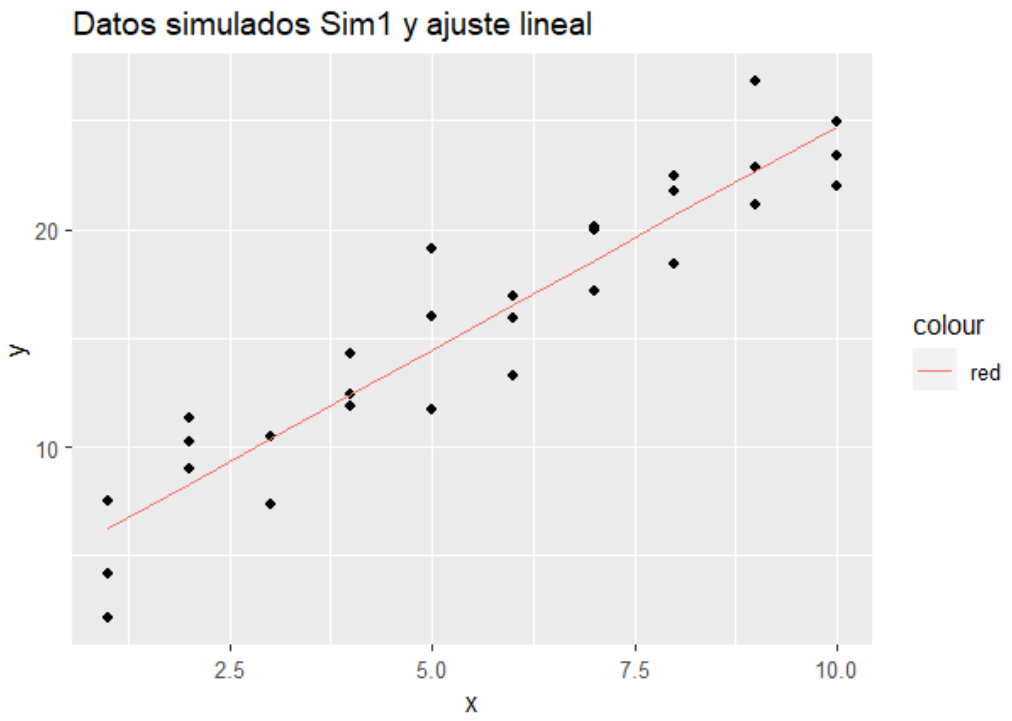
Sebastián D'Amelio, Lucas Oliaro, Julián Fraga

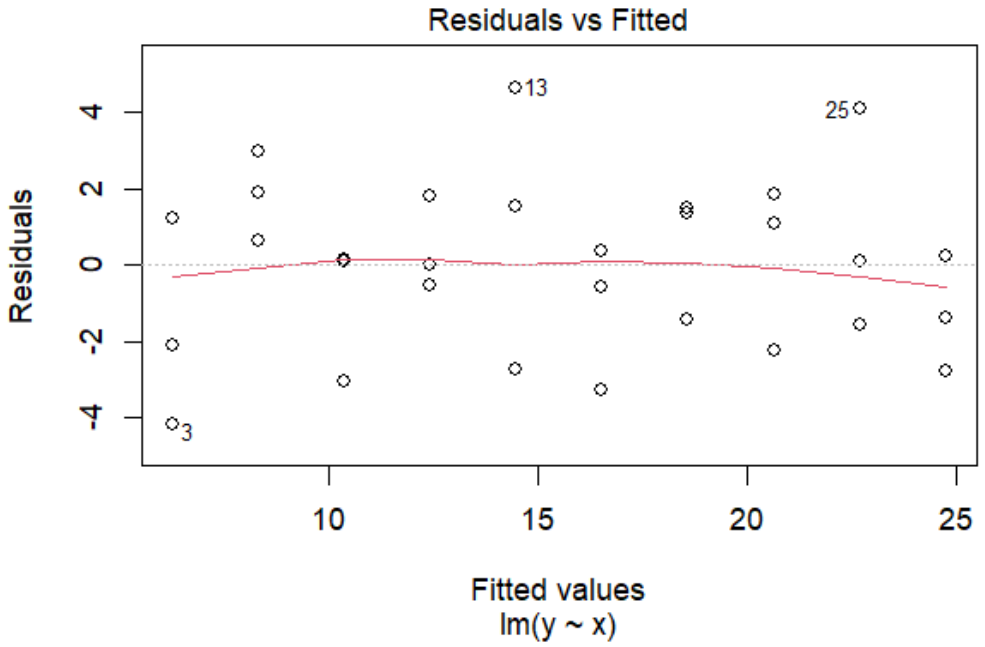
## Tarea 1

Realizamos una exploración del dataset SIM1 y un ajuste lineal. Calculamos los residuos y graficamos. Este último gráfico parece ir de la mano con lo explicado en clase en el sentido de que la tendencia de los residuos debería ser alrededor de cero. Además, están contenidos entre aproximadamente -2.5 y 3, una banda casi simétrica.

El resumen del modelo utilizando la función *Summary* es el siguiente:

- Ordenada al origen: **t value** 4.85 ; **p value** 4.09e-05 ***
- Coeficiente lineal: **t value** 15.65 ; **p value** 1.17e-14 ***

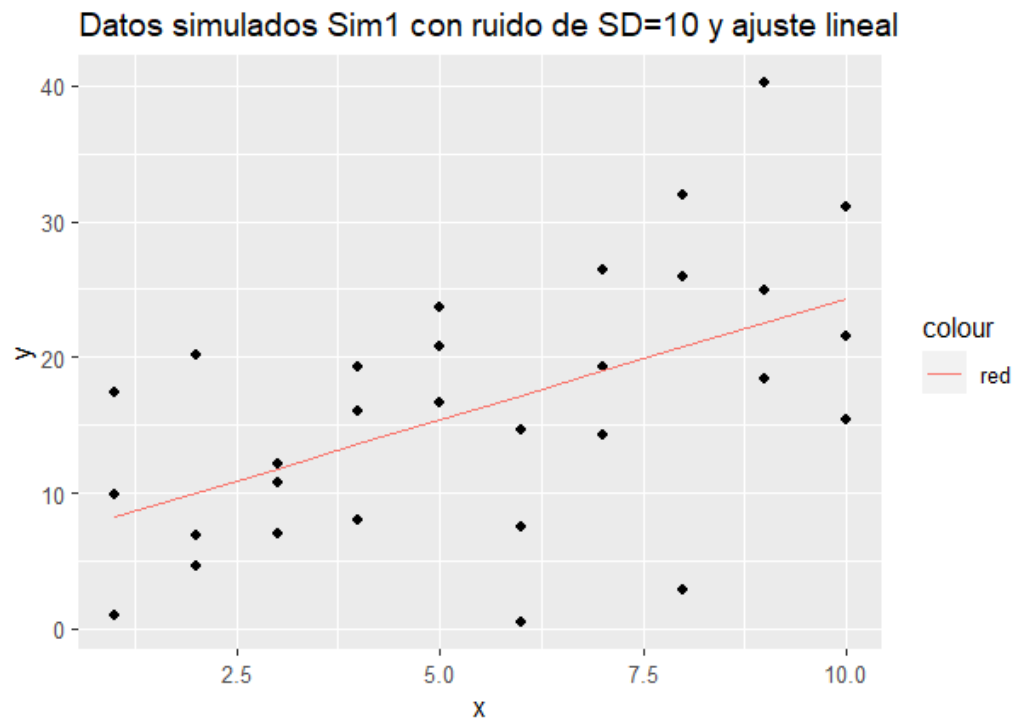


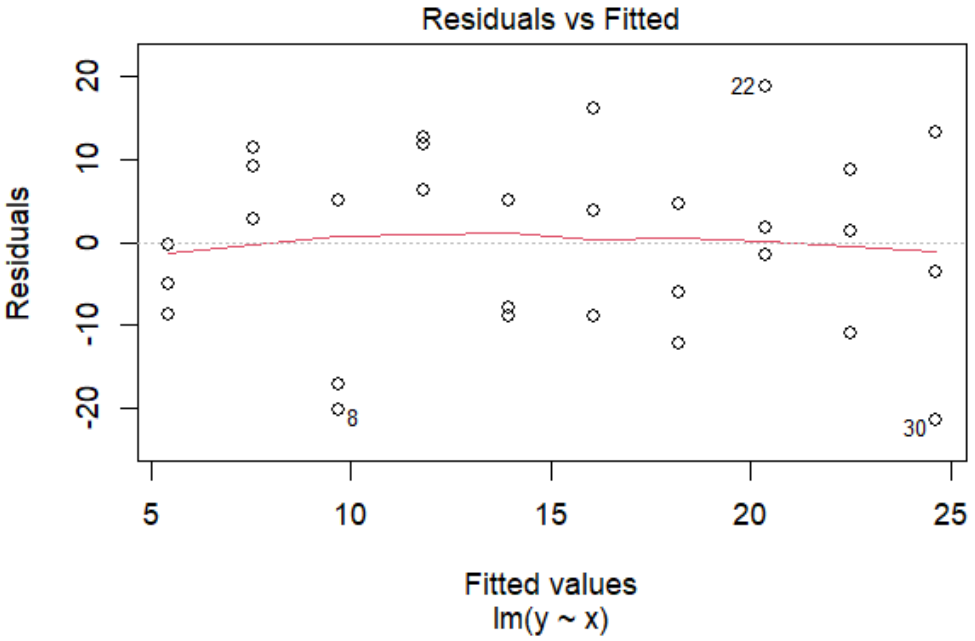



Luego generamos un vector de ruido y lo sumamos al dataset. Vemos que con un ruido de media cero y desviación 10, los residuos pasan a estar contenidos en una banda cuatro veces más grande (entre -10 y 10 aproximadamente). El resumen del modelo ahora es:

- Ordenada al origen: **t value** 0.775 ; **p value** 0.44 (sin asteriscos)
- Coeficiente lineal: **t value** 3.079 ; **p value** 4.62e-3 **

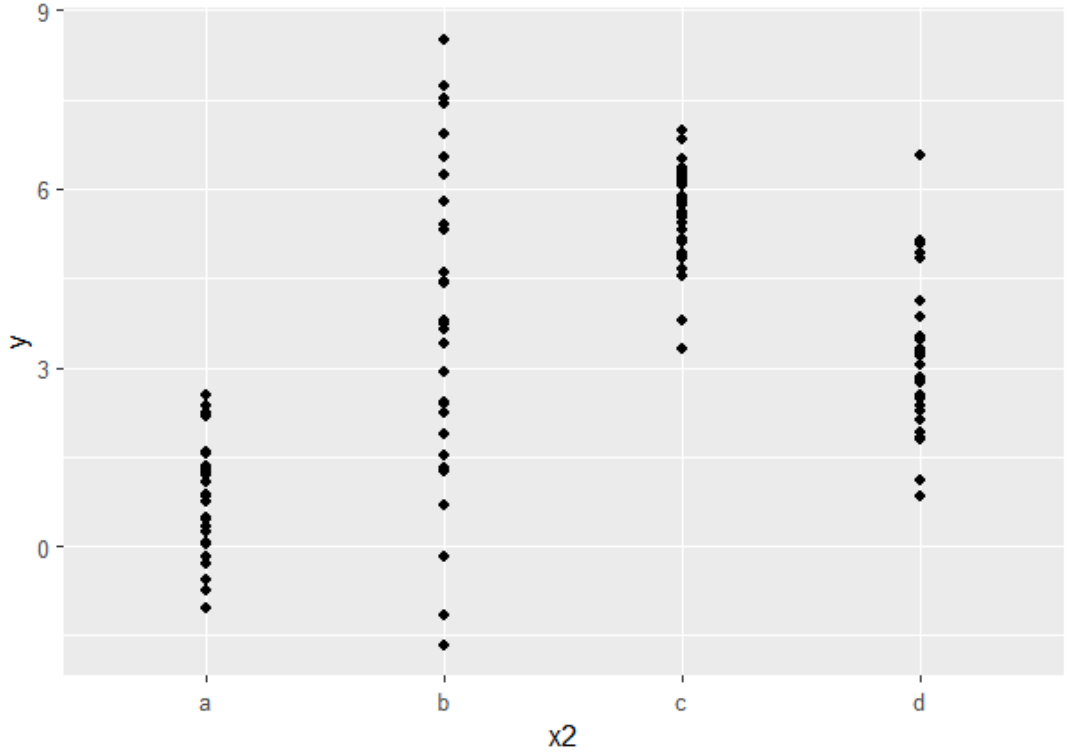
Con ruido de desviación estándar de 30, la pendiente ya no es significativa


Datos simulados Sim1 con ruido de SD=10 y ajuste lineal
40
30
20
10
0
y
2.5
5.0
7.5
10.0
x
colour
red


Residuals vs Fitted
Residuals
20
10
0
-10
-20
22
8
30
5
10
15
20
25
Fitted values
lm(y ~ x)

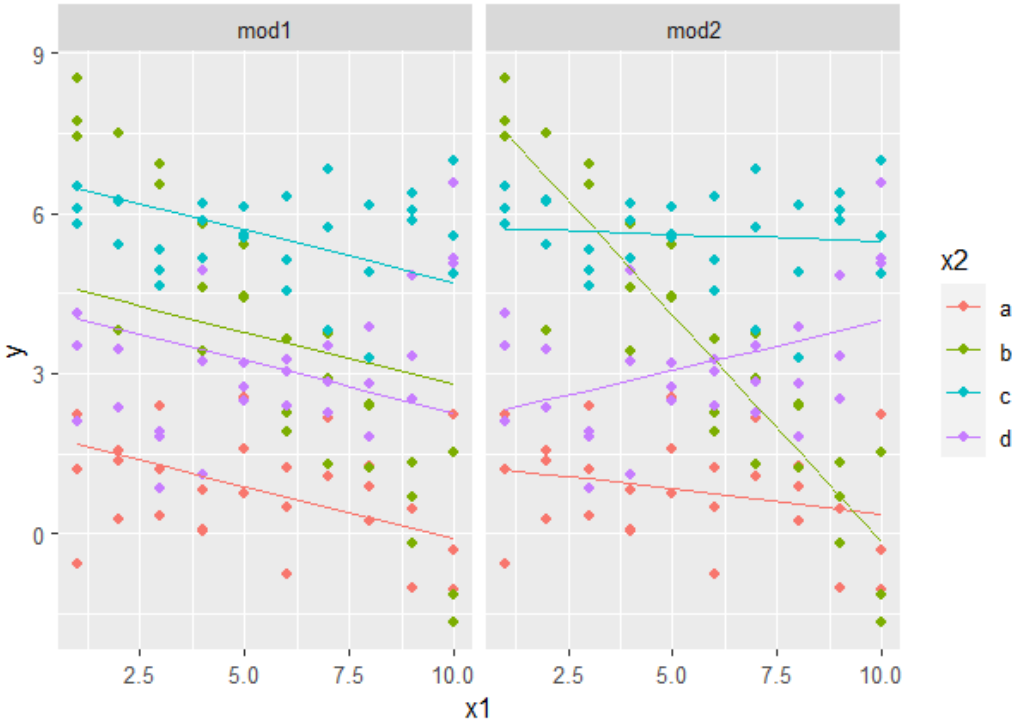
## Tarea 2

Importamos el dataset **sim3** e identificamos las variables usando *glimpse*. Vemos que la variable **x2** es la variable categórica e **y** la variable contínua. Realizamos gráfico de dispersión en el que observamos 4 categorías.




Investigamos la diferencia entre las formula **y ~ x1 + x2** e **y ~ x1 * x2** utilizando model_matrix. Es evidente que el segundo modelo es el que genera más parámetros.

Obtenemos la grilla de ajuste utilizando *gather_predictions* para generar las predicciones de ambos modelos y visualizamos los resultados geom_points y ralizando líneas de tendencia para cada variable categórica. A juzgar por las líneas de tendencia verde y cian, el modelo “mod2” (el que lleva términos de interacción) pareciera ser el que mejor ajusta a los datos.

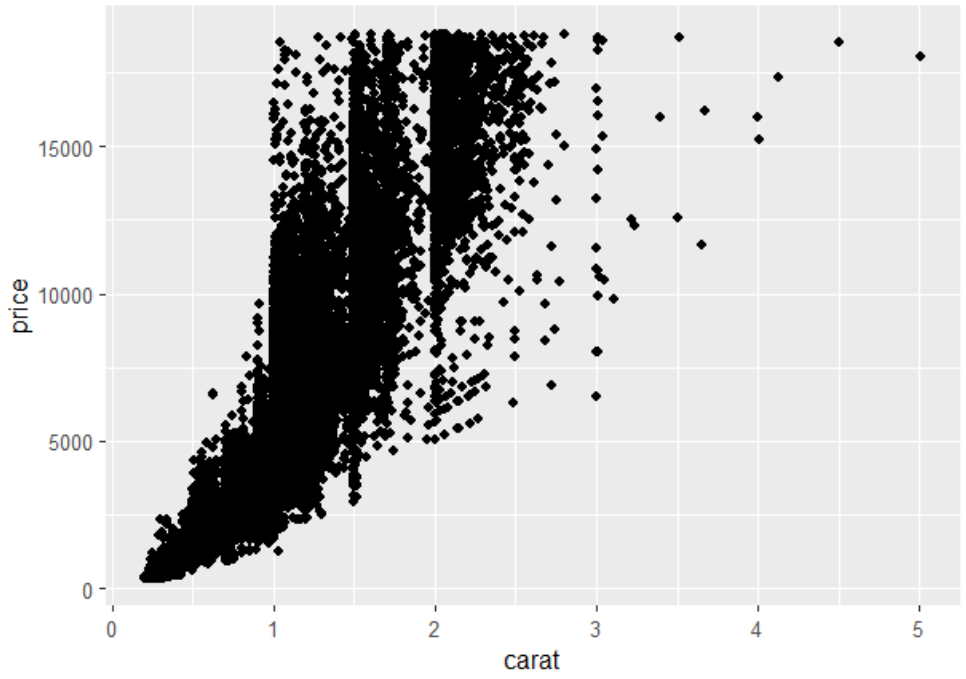


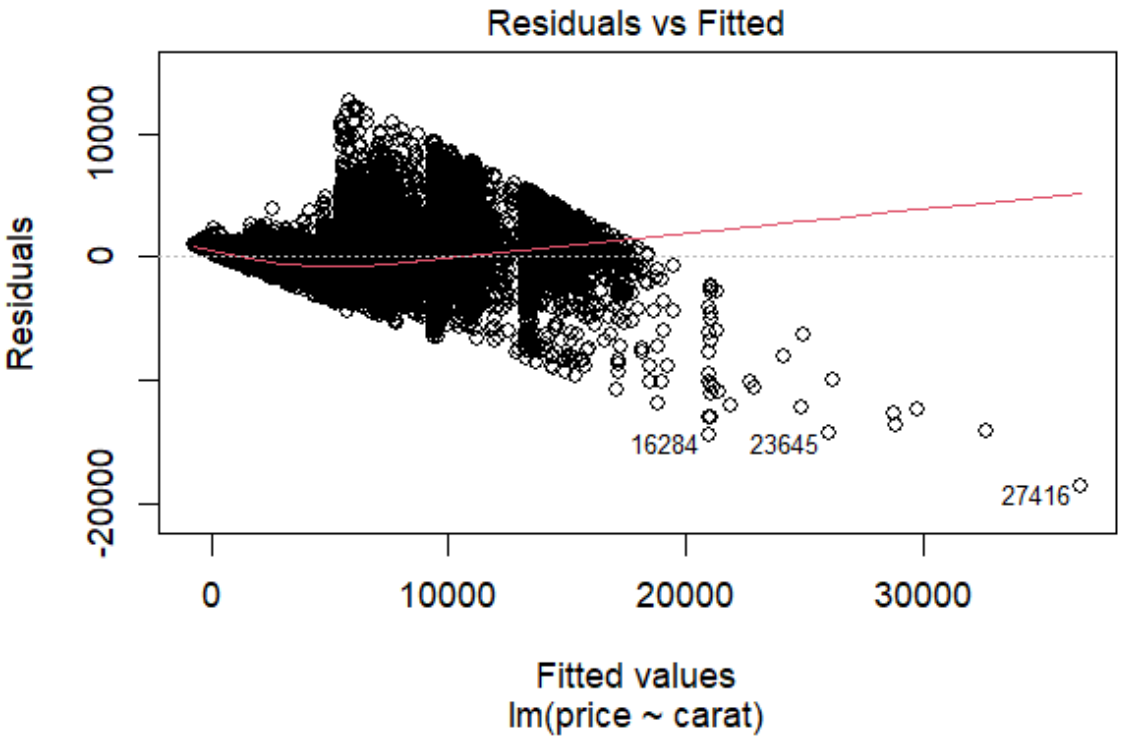

Un ejemplo donde un modelo con términos de interacción es útil podría ser, por ejemplo, si se quiere medir una magnitud física como la energía cinética o una fuerza como por ejemplo la energía con la que dispara un rifle de aire comprimido. Teniendo como datos la masa de cada proyectil y la velocidad de salida de la boca del cañón (*muzzle velocity*) podrías calcular la energía cinética como un ajuste del estilo **T~m*vc** donde **m** es la masa y **vc** la velocidad al cuadrado.

## Tarea 3

Cargamos el dataset de diamantes e hicimos un primer scatterplot con la variable **precio** en función de los **kilates** (**carat** vs **price**). A primera vista pareciera haber una relación del tipo **precio~exp(kilates).**

Al tratar de modelar con esta hipótesis, observamos que los residuos son inmensos (del orden de 10k)



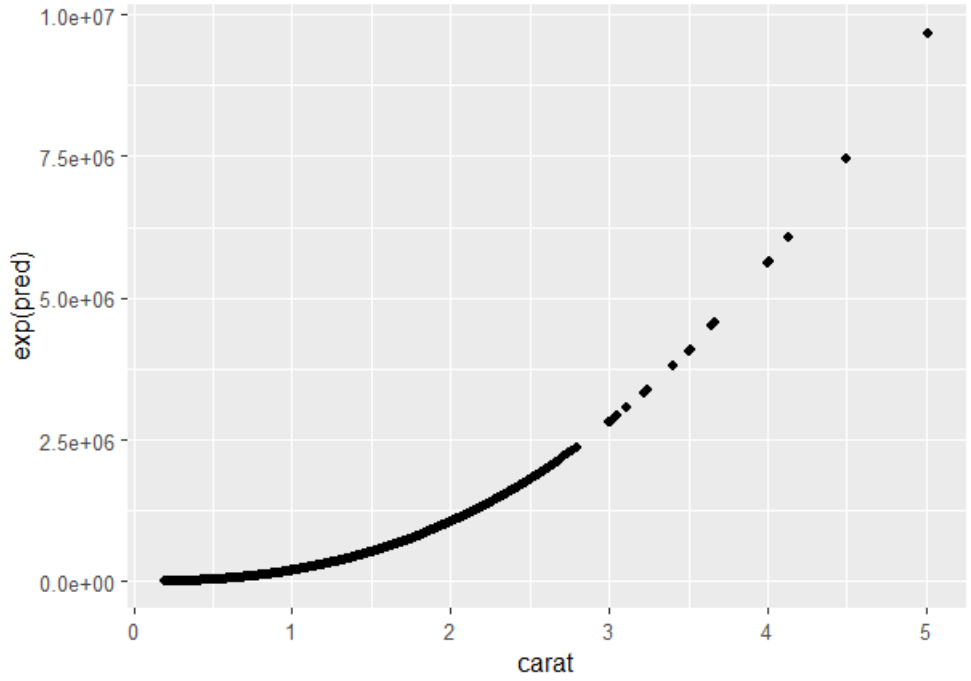



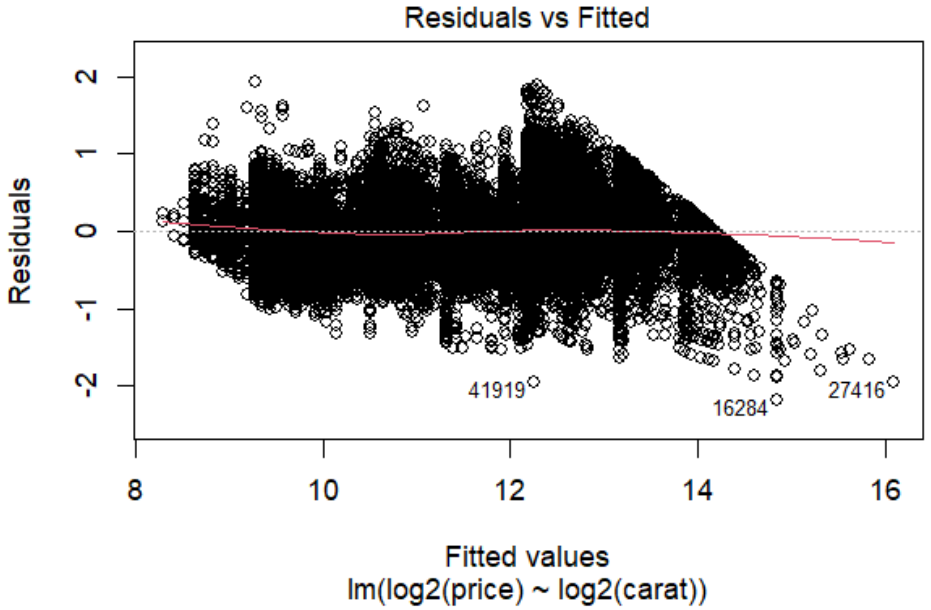
Generamos un nuevo modelo en relación ***log-log*** y generamos predicciones. Graficamos un nuevo scatterplot con **carat~exp(predicciones)**.

No es excelente: de entrada vemos que las unidades de lo que vendría a ser el precio toma valores de alrededor de 2.5M a los 3 kilates mientras que en el dataset original, a la altura de los 3 kilates el precio es de entre 10k y 15k. Sin embargo, por alguna razón que no entendemos bien los residuos parecen no ser tan terribles. Al menos están contenidos entre -2 y 2.

El resumen del modelo es el siguiente:

- Ordenada al origen: **t value** ~ 6200 ; **p value** 2e-16 ***
- Coeficiente lineal: **t value** 866 ; **p value** 2e-16 ***



## Tarea 4

Llegamos a hacer un boxplot visualizando el precio en función de la claridad para un color determinado de diamante. También agregamos la variable dicotómica al dataset discriminando entre buena y mala calidad según su claridad (decisión completamente arbitraria, como primera exploración)