DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico Número 3

Ajustando suavemente

23 de marzo de 2023 — Métodos Numéricos

Integrante	LU	Correo electrónico
Fiorino Santiago	516/20	fiorinosanti@gmail.com
Salmun Daniel	108/19	salmundani@gmail.com
Valentini Nicolas	86/21	nicolasvalentini@hotmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - Pabellón I
Intendente Güiraldes 2160 - C1428EGA
Ciudad Autónoma de Buenos Aires - Argentina
Tel/Fax: (54 11) 4576-3359
http://exactas.uba.ar

Palabras claves: Regresión Local Ponderada, Loess, Suavizado de Scatterplots

En este informe se trabaja sobre el método de Regresión Local Ponderada, propuesto en el Paper [1]. Primero se desarrolla paso a paso el procedimiento. Luego se recrea la Sección 5 del mismo Paper, el cual contiene un caso de uso del algoritmo sobre datos meteorológicos. A esto le sigue experimentación propia, en la que generamos una gran variedad de datasets para analizar el comportamiento del algoritmo. Con cada uno de los datasets generados se corrió el algoritmo variando distintos parámetros del mismo, en búsqueda de la combinación óptima. Por último se explican diferentes herramientas de visualización usadas en el Paper, explicando cómo usarlas y qué información se puede extraer de ellas.

Índice

1. Introducción

En este informe se implementará y estudiará el procedimiento de Regresión Local Ponderada (loess), presentado en el Paper [1]. En primer lugar se explicará en qué consiste este procedimiento. Luego, se replicaran los gráficos de la sección cinco del paper mencionado. Finalmente, se llevarán a cabo nuevos experimentos con el objetivo de observar cómo se comporta frente a distintos datos de entrada, cambios de parámetros, entre otros.

1.1. Loess Univariado

En el Paper [2] se introduce la Regresión Local Ponderada. El objetivo de la misma es generar una curva suave que se ajuste a puntos en $\mathbb{R}^2$ que presentan ruido. El ruido en las observaciones es un fenómeno muy común, ya que en la práctica hay muchas variables externas al experimento que influyen al momento de realizar una medición. Este procedimiento asume que las observaciones se pueden explicar de la siguiente forma:

$$y_i = f(x_i) + \epsilon_i \tag{1}$$

Siendo ϵ_i variables independientes que siguen una distribución normal. Para poder estimar la función original, eliminando el ruido y generando una curva suave, se propuso el método Loess.

La principal idea de este procedimiento es que para cada valor de x, se estima el valor de $f(x)$ usando una cierta cantidad de vecinos de x para los cuales ya conocemos su valor de y. Luego a cada vecino se le asigna un peso basado en su distancia a x, con el objetivo de que los puntos más cercanos a x tengan mayor influencia en la estimación. Más adelante veremos cuál es la forma propuesta de calcular los pesos. Una vez que tenemos los vecinos de x, cada uno con su peso asignado, ya podemos calcular la estimación. Recordamos que las ecuaciones normales de regresión lineal están dadas por:

$$X^T X\beta = X^t Y \tag{2}$$

Lo único que debemos cambiar para tener en cuenta los pesos que les asignamos a los vecinos, es agregar una matriz W la cual contiene los pesos de cada vecino en su diagonal. Las ecuaciones normales con los pesos queda expresada como:

$$X^T W X\beta = X^t W Y \tag{3}$$

Resolviendo el sistema y haciendo el producto punto de β con x, obtenemos la estimación para el x dado.

1.2. Loess Multivariado

El Paper [1] se basa en el procedimiento de la sección 1.1 para generar un procedimiento muy similar pero para casos en los que la variable dependiente está dada por múltiples variables independientes. En este informe se trabajará y experimentará con este procedimiento.

La extensión a múltiples variables genera ciertas complicaciones. El principal problema que surge es que en la versión univariada encontrar los vecinos más cercanos es algo muy directo, sin embargo, cuando tenemos múltiples variables no lo es. La intuición es tomar la distancia euclidiana, pero al estar cada variable en diferentes escalas, se puede dar el caso de que la diferencia en magnitud entre ellas haga que una tenga mucho más efecto en la distancia que otra. Más adelante explicaremos las soluciones propuestas.

2. Desarrollo

2.1. Procedimiento Loess

Notamos $X \in \mathbb{R}^{n \times m}$ a la matriz de variables independientes observadas (n observaciones y m variables), $Y \in \mathbb{R}^n$ al vector de las n variables dependientes correspondientes. Dado un nuevo $x' \in \mathbb{R}^m$ a estimar, lo primero que debemos hacer es encontrar los q vecinos más cercanos a x' dentro de X. Calcular la distancia euclidiana de x' con cada fila de X puede ser problemático, ya que las variables independientes estarán en diferentes unidades y puede haber una que domine, sacándole relevancia a las demás. Para solucionar esto lo primero que hacemos es estandarizar los datos de la siguiente manera:

$$x_{i,j} = \frac{x_{i,j} - \mu_j}{\sigma_j} \tag{4}$$

Siendo μ_j el promedio y σ_j el desvío estándar de la columna j. De esta forma los datos van a seguir una distribución Normal con $\mu = 0$ y $\delta = 1$. Esto soluciona el problema de las unidades, y nos permite calcular las distancias sin discriminar a ninguna variable independiente. Una vez calculadas las distancias, creamos una matriz con los q vecinos más cercanos, ordenados ascendentemente por distancia a x', la cual llamaremos $N \in \mathbb{R}^{q \times m}$. Llamamos $Y' \in \mathbb{R}^q$ a las variables dependientes asociadas a N. Además, creamos una matriz diagonal $W \in \mathbb{R}^{q \times q}$. Esta matriz le asignará pesos a cada vecino, con el objetivo de que los vecinos más cercanos a x' tengan mayor efecto en la estimación. Definimos D al conjunto ordenado de distancias euclidianas entre cada fila de N y x'. Para calcular los pesos lo primero que hacemos es normalizar el conjunto. Luego, buscamos que las distancias menores tengan pesos mayores, por lo tanto les aplicamos la función tricubica, dada por la siguiente fórmula:

$$W(\mu) = \begin{cases} (1-\mu^3)^3 & \text{si } \mu \in [0,1] \\ 0 & \text{si } \mu \notin [0,1] \end{cases} \tag{5}$$

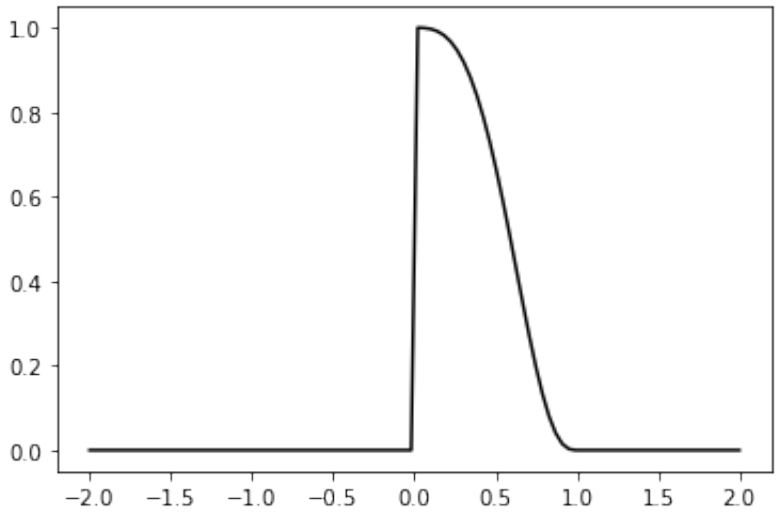


Figura 1: Gráfico de la función Tricubica

Como podemos observar en la Figura 1, al aplicar esta función a las distancias normalizadas (números positivos entre 0 y 1) vamos a obtener un peso mayor para las distancias más chicas. Así es como calculamos los elementos de la matriz diagonal W.

Ya con estas matrices definidas, podemos plantear las ecuaciones normales de cuadrados mínimos pesados de la siguiente forma:

$$N^T W N \beta = N^T W Y' \tag{6}$$

En el caso del fitting lineal, la matrix N simplemente contendrá las variables independientes. En el caso del fitting cuadrático, la matriz contendrá las variables independientes, su cuadrado, y el producto cruzado entre ellas. Lo único que nos queda es obtener el β que resuelva el sistema (6), y la estimación será $x' \cdot \beta$

3. Experimentación

3.1. Recreación de la Sección 5 del Paper [1]

En esta Sección del Paper se trabaja con datos meteorológicos de la ciudad de Nueva York. Se efectuaron un total de 111 medidas de cuatro variables (ozono, radiación solar, velocidad del viento y temperatura). En esta sección se analizan los datos para describir una dependencia del ozono con las otras variables, para que el ozono se pueda predecir usando los pronósticos meteorológicos. La Figura 2 es una matriz de scatterplots de los datos utilizados.

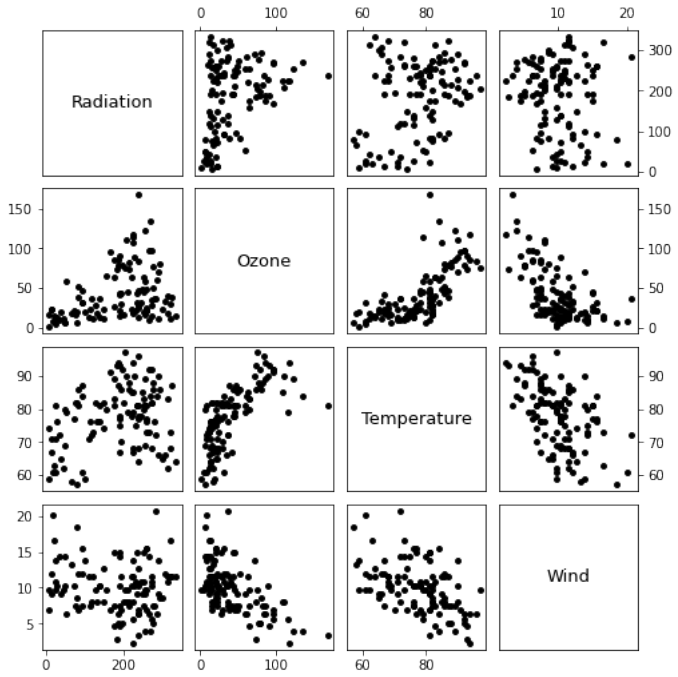
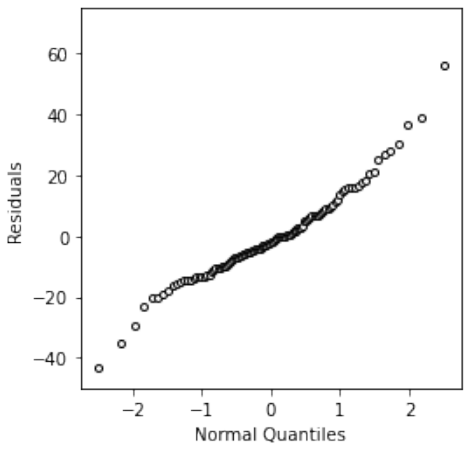
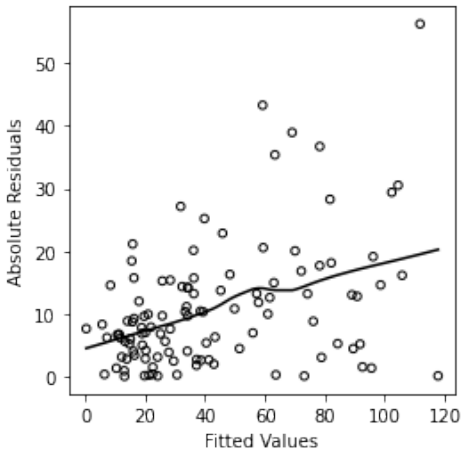


Figura 2: Datos Meteorológicos

El primer paso del análisis es aplicar el algoritmo Loess, siendo el ozono la variable dependiente, y las demás las independientes. El fitting será lineal con $f = 0{,}4$. Luego se generan diferentes gráficos para analizar distintas propiedades del algoritmo. En la Figura 3a se busca mostrar que ϵ_i sigue una distribución normal, tal como se plantea en una hipótesis anterior del Paper. En la Figura 3b se grafica $|\epsilon_i|$ contra y' para chequear que la varianza sea constante. Finalmente, en los cuadros de la Figura 4 se grafica ϵ_i contra las variables independientes, para chequear bias.



(a) QQ Plot



(b) Estimaciones vs. Residuos

Figura 3: Gráficos de diagnostico

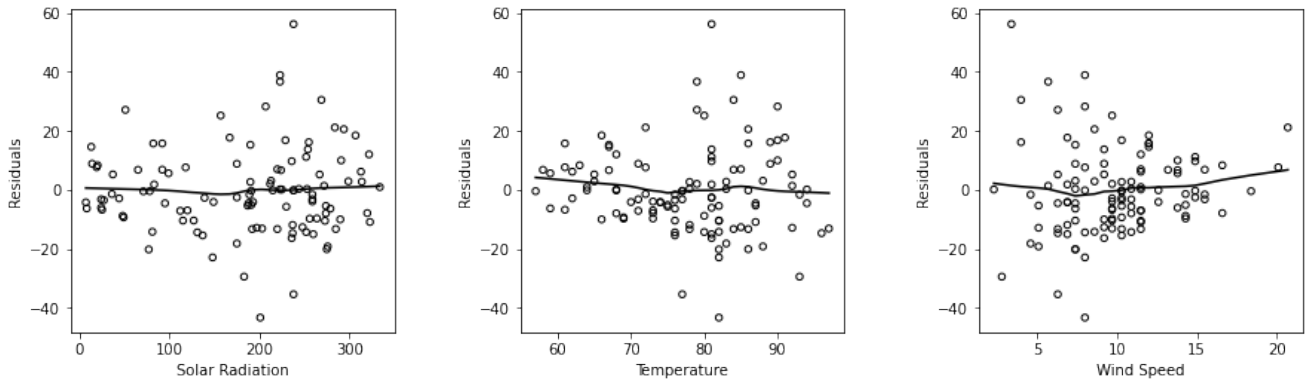


Figura 4: Residuos del Ozono vs. Variables Independientes

Más adelante, en la Sección 4: "Herramientas de visualización" analizaremos qué indicios o conclusiones se pueden extraer de estas Figuras.

Luego, para reducir la distorsión cambian a fitting cuadrático con $f = 0{,}8$. Con esto logran reducirla, pero los scatterplots de la Figura 3 seguían presentando deficiencias. Para solucionar esto, decidieron aplicar fitting cuadrático de nuevo, pero esta vez sobre la raíz cúbica del Ozono. Esta estimación pasó los chequeos de diagnostico. Por último, generaron las Figuras 5, 6 y 7, las cuales contienen curvas de nivel en las que se fijan dos variables, y se deja una libre, pudiendo observar mejor el comportamiento de la estimación.

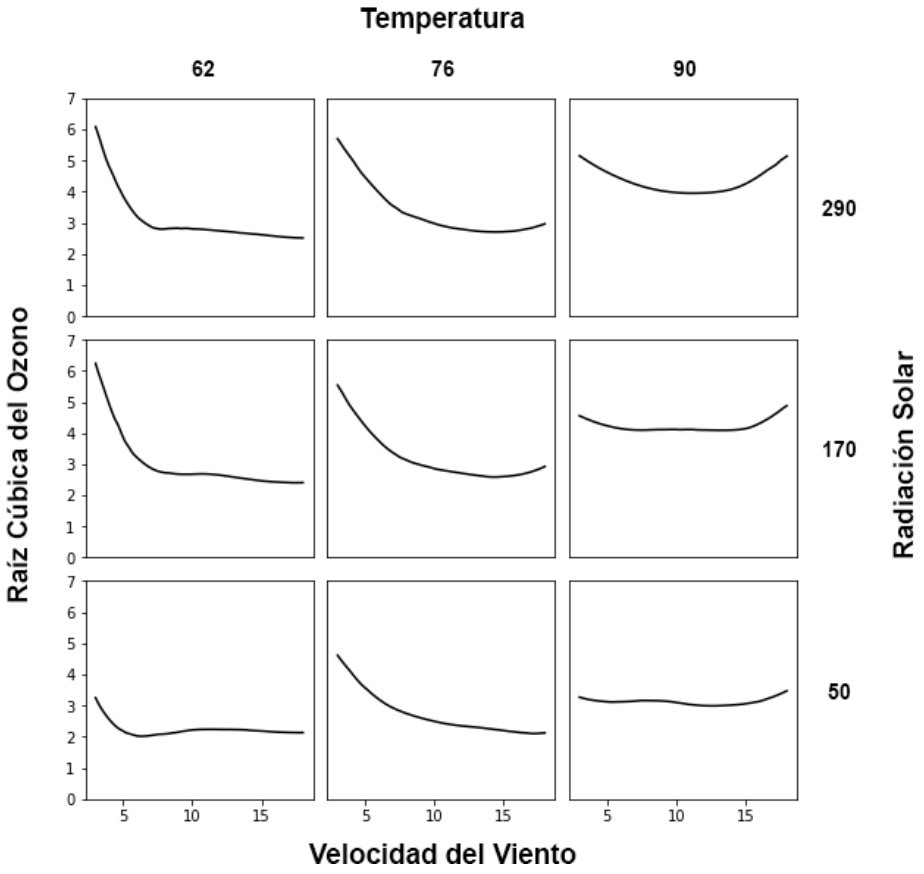


Figura 5: Temperatura y Radiación Solar fijas

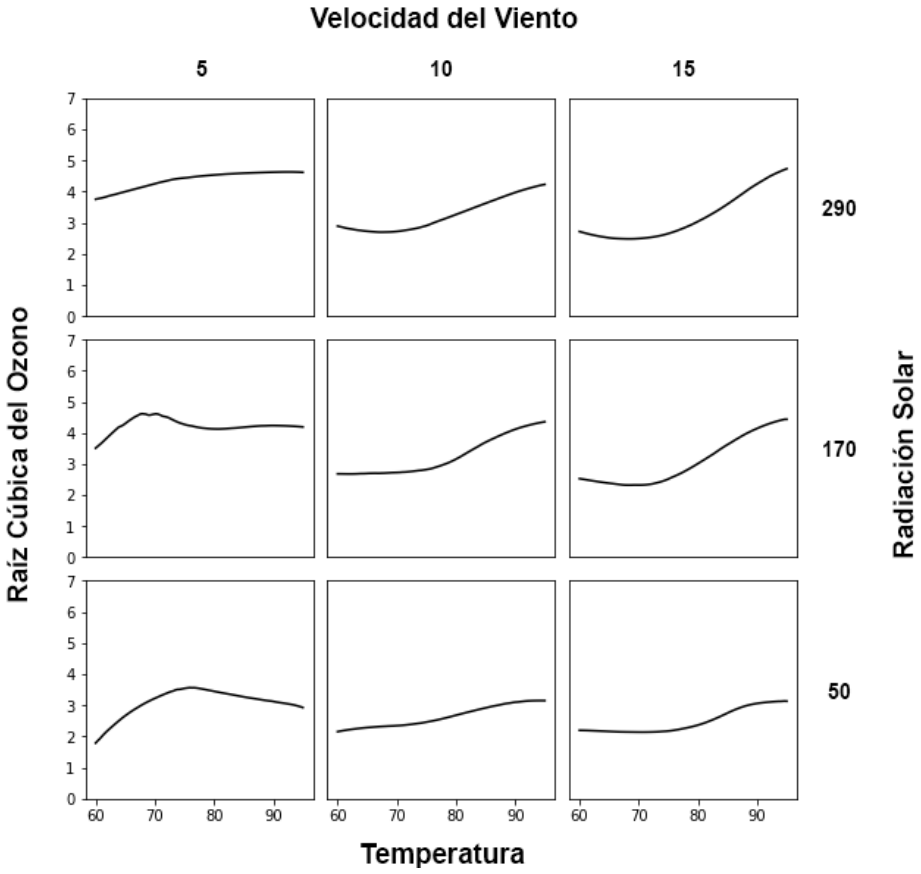


Figura 6: Velocidad del Viento y Radiación Solar fijas

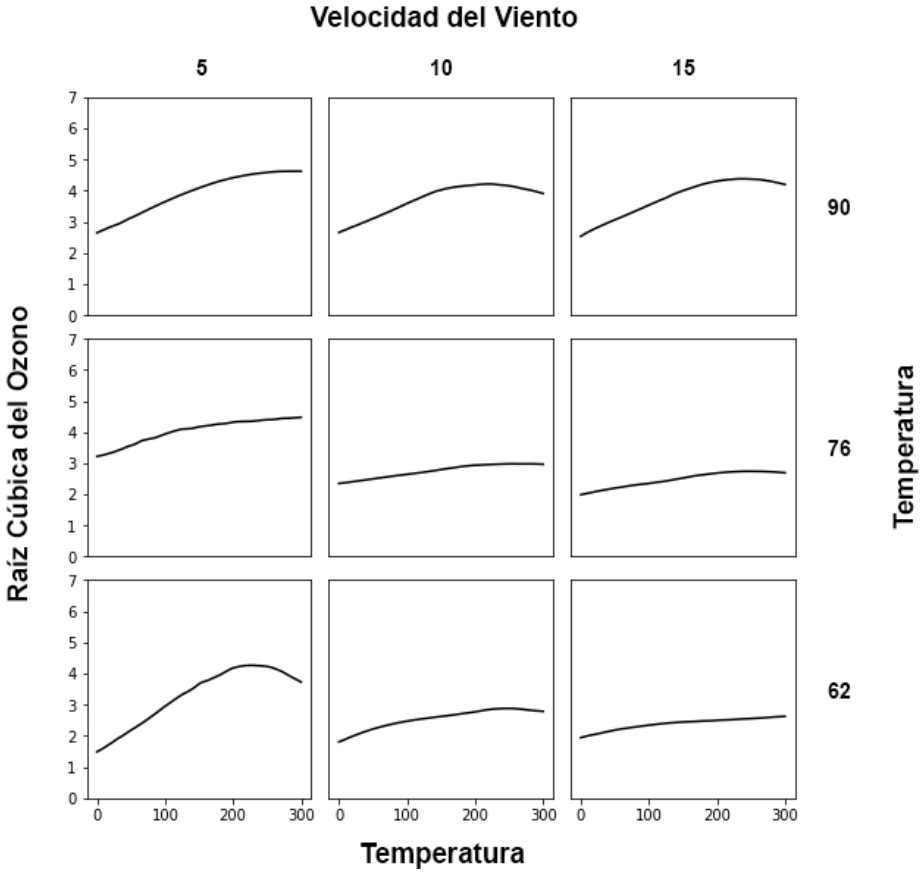


Figura 7: Velocidad del Viento y Temperatura fijas

3.2. Análisis con data sintética

En esta sección vamos a trabajar con datos sintéticos para poder observar el comportamiento del algoritmo ante diferentes datos de entrada y parámetros. Para poder visualizar mejor los resultados, creamos los datos en $\mathbb{R}^3$. Para empezar, generamos 4 superficies, dadas por las fórmulas y gráficos de la Figura 8. Elegimos estas superficies para tener ejemplos variados: una lineal, una cuadrática, una cúbica y una con bordes afilados. El plano xy está discretizado con una malla de 50×50 puntos, con $x \in [-5, 5]$, e $y \in [-5, 5]$.

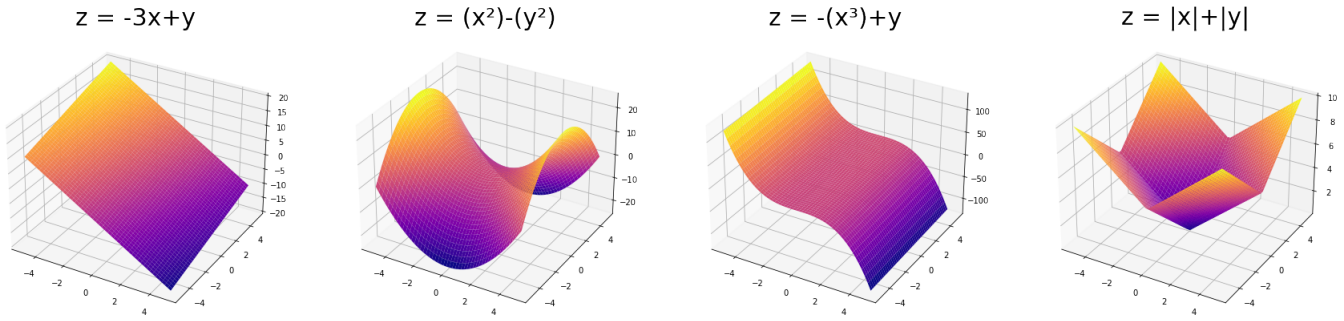


Figura 8: Superficies

Luego, le agregamos ruido a cada una de ellas. El ruido sigue una distribución normal $\mathbf{N}(0, \sigma/2)$, siendo σ el desvío de los puntos de la superficie. Esta distribución fue elegida porque en el procedimiento se asume que los datos están generados de la forma $y_i = g(x_i) + \epsilon_i$, y que los ϵ_i son variables independientes que siguen una distribución normal. En la Figura 9 podemos observar cómo quedan estas superficies con el ruido agregado.

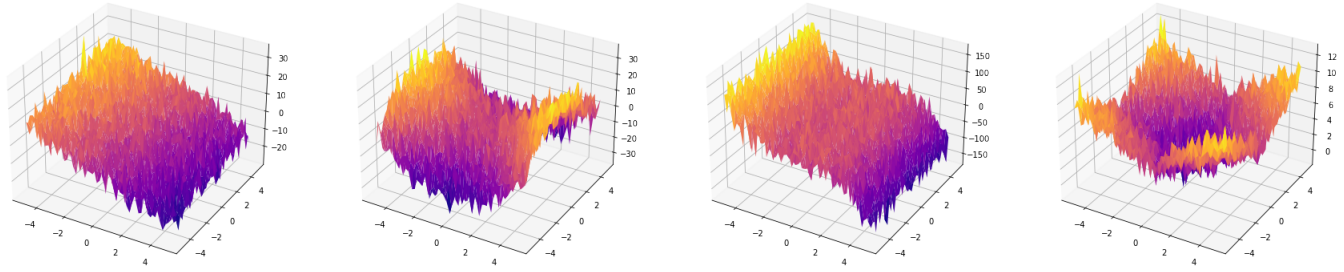


Figura 9: Superficies con ruido

Una vez generada nuestra data sintética, corrimos el algoritmo de Loess con diferentes parámetros. El primer parámetro con el que vamos a experimentar es el f, el cual determina la cantidad de vecinos que tiene en cuenta el algoritmo al momento de estimar. Mientras más grande sea el f, la superficie estimada se vuelve mas suave. Las Figuras 10, 11 y 12 son fitting cuadráticos, con $f = 0{,}1$, $f = 0{,}5$ y $f = 1$ respectivamente. En estas Figuras podemos observar cómo se suavizan las superficies cuando el f aumenta. Algo interesante a notar es que tanto en la superficie cuadrática como en la de módulos, el efecto del suavizado produce peores resultados.

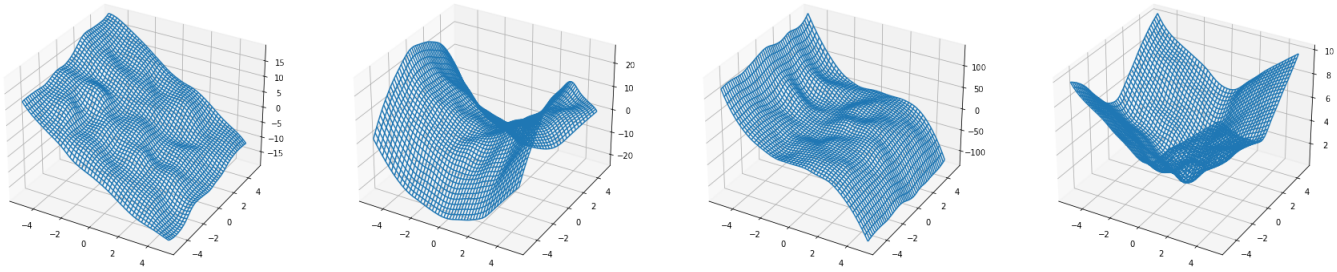


Figura 10: Estimaciónes obtenidas mediante un fitting cuadrático con $f = 0{,}1$

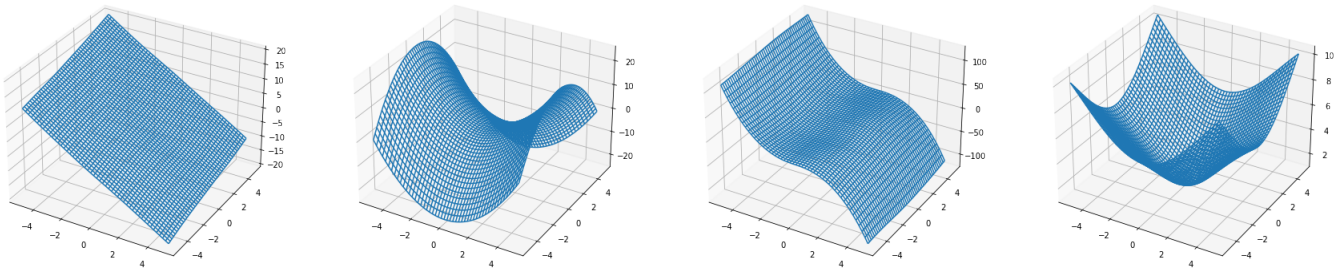


Figura 11: Estimaciónes obtenidas mediante un fitting cuadrático con $f = 0{,}5$

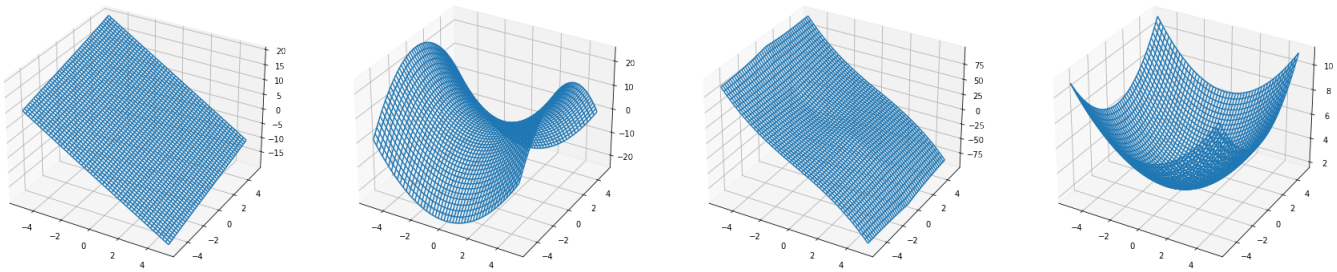


Figura 12: Estimaciónes obtenidas mediante un fitting cuadrático con $f = 1$

Para poder observar y comparar mejor la diferencia ante el cambio del parámetro f, decidimos graficar curvas de nivel. Para cada superficie generamos dos Figuras. La primera será la curva de nivel fijando el eje y en 0, y la segunda fijando el eje x en 0. Cada Figura tendrá dos gráficos. En el primero se comparan las curvas estimadas con la curva de la superficie original, para poder ver que tan parecidas quedan. En el segundo se comparan las curvas con las superficies ruidosas, para poder ver cómo estos afectan al resultado final.

3.2.1. Curvas de Nivel del Plano

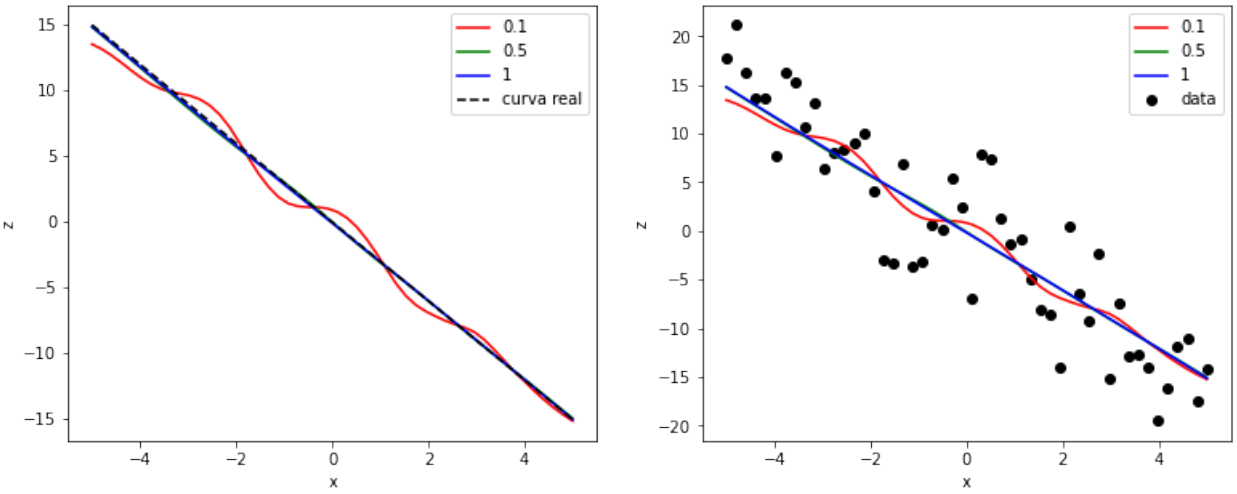


Figura 13: Curva de nivel del plano con y fijo

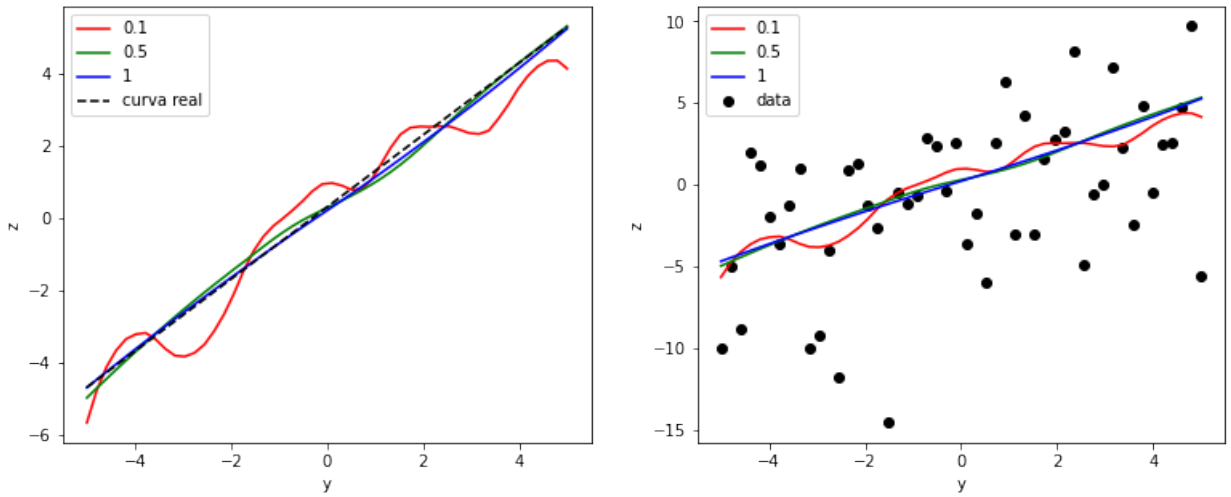


Figura 14: Curva de nivel del plano con y fijo

En las Figuras 13 y 14 podemos observar que en el caso del plano, el algoritmo fue más efectivo para $f = 1$. Las curvas generadas con $f = 0{,}1$ son sensibles a los outliers, por lo tanto si en uno de los vecindarios hay muchos puntos por debajo de la recta, se generará una curvatura para abajo en ese sector. Para $f = 0{,}5$ se comportó mejor, aunque se puede observar una leve oscilación en comparación con la curva de $f = 1$, que es casi completamente recta.

3.2.2. Curvas de Nivel de la Silla

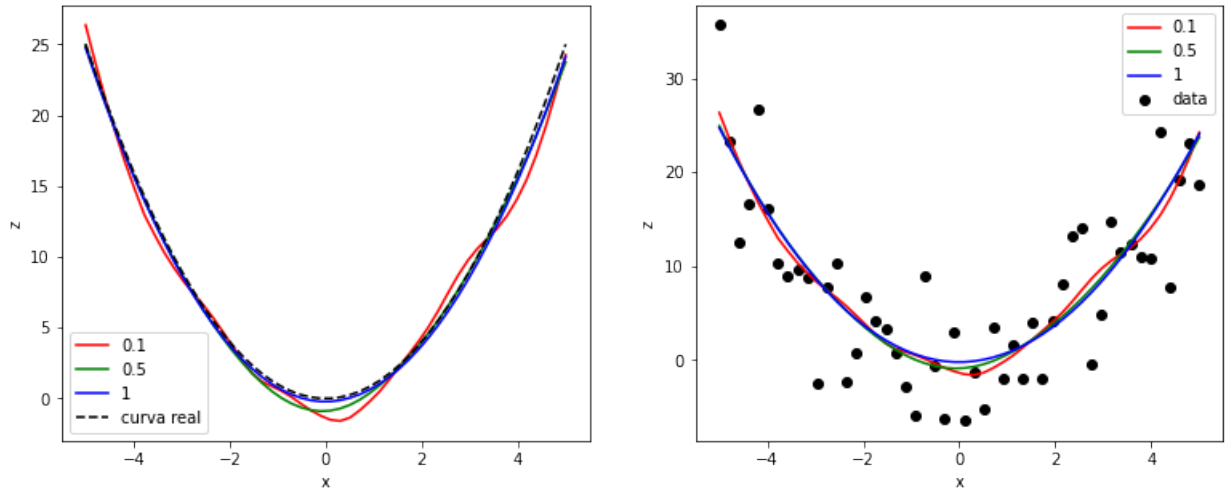


Figura 15: Curva de nivel de la silla con y fijo

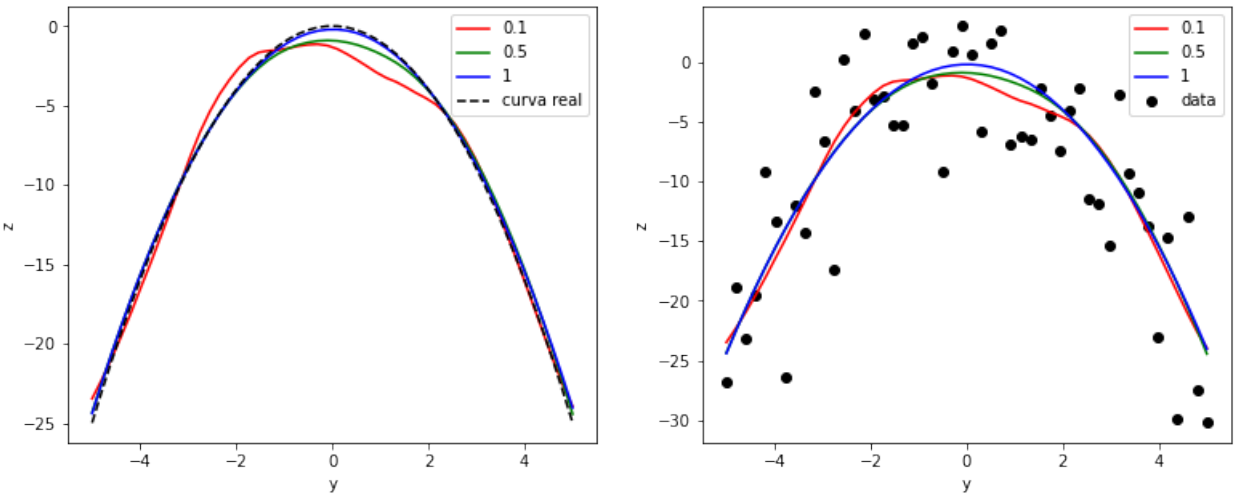


Figura 16: Curva de nivel de la silla con y fijo

En las Figuras 15 y 16 nos encontramos con una situación similar a la del plano. Nuevamente, la mejor estimación la obtuvimos con $f = 1$. La curva estimada con $f = 0{,}1$ es irregular y contiene varias imperfecciones con respecto a la curva original. Tomar muy pocos puntos hace que la estimación sea demasiada local, estimando de una peor manera la curva en cuestión. La curva estimada con $f = 0{,}5$ es lo suficientemente suave, pero en algunos sectores difiere con la original, obteniendo una estimación levemente peor que con $f = 1$.

3.2.3. Curvas de Nivel de la Cúbica

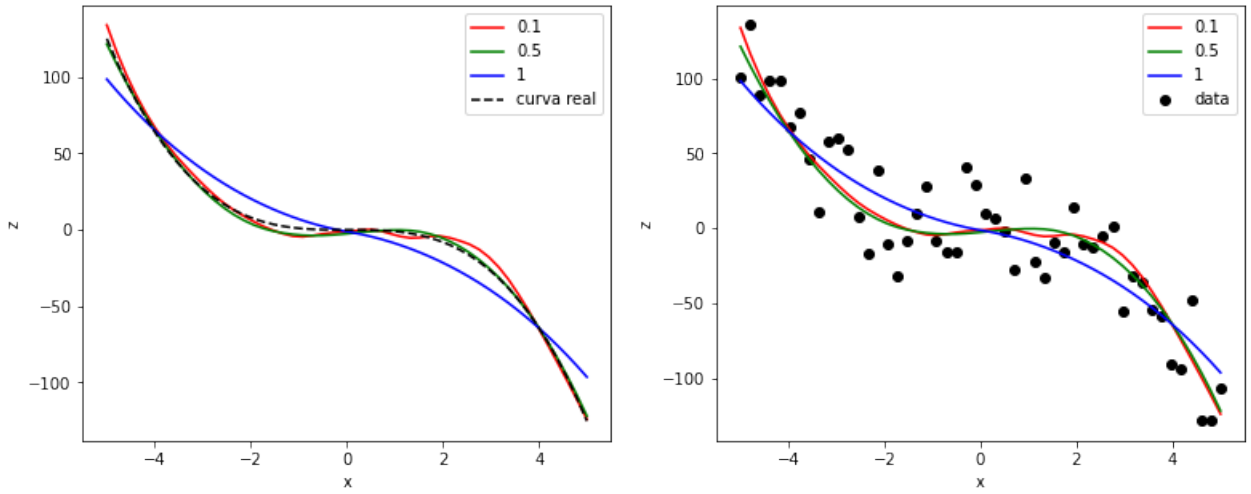


Figura 17: Curva de nivel de la cúbica con y fijo

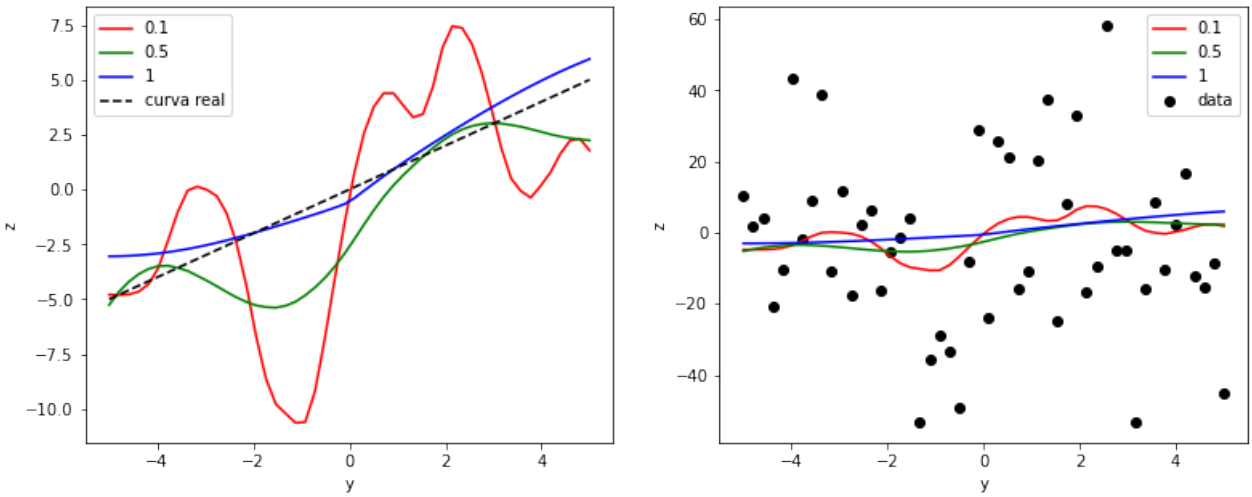


Figura 18: Curva de nivel de la cúbica con x fijo

En las Figuras 17 y 18 observamos algo interesante. La curva con el y fijo se comporta de forma cúbica, y en ese caso las estimaciones con $f = 0{,}1$ y $f = 0{,}5$ fueron superiores. Sin embargo, la curva con el x fijo se comporta de forma lineal, y en ese eje obtuvimos los mismos resultados que en el caso del plano, mientras más grande el f, mejor.

3.2.4. Curvas de Nivel de la Absoluta

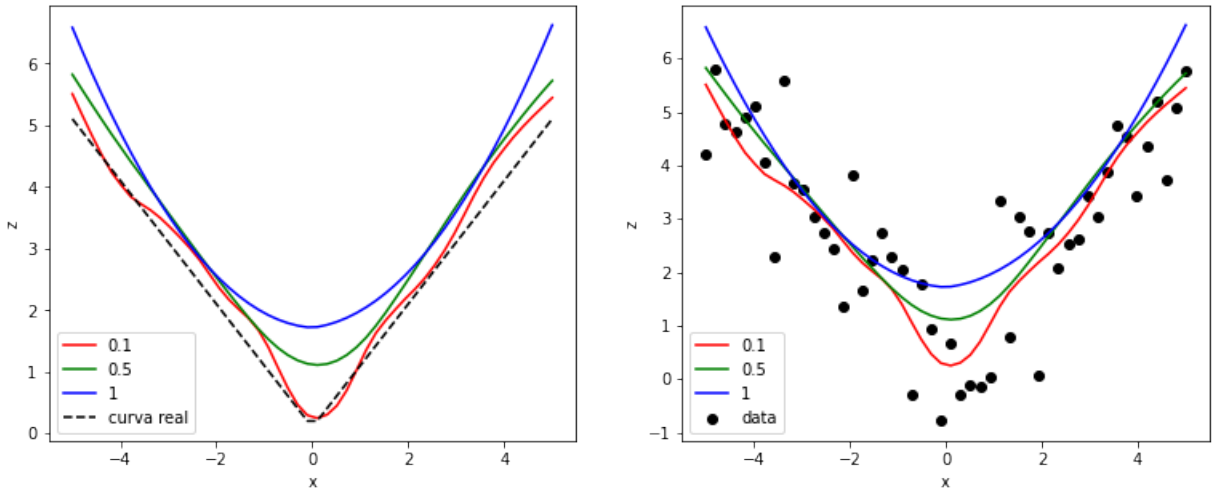


Figura 19: Curva de nivel de la cúbica con y fijo

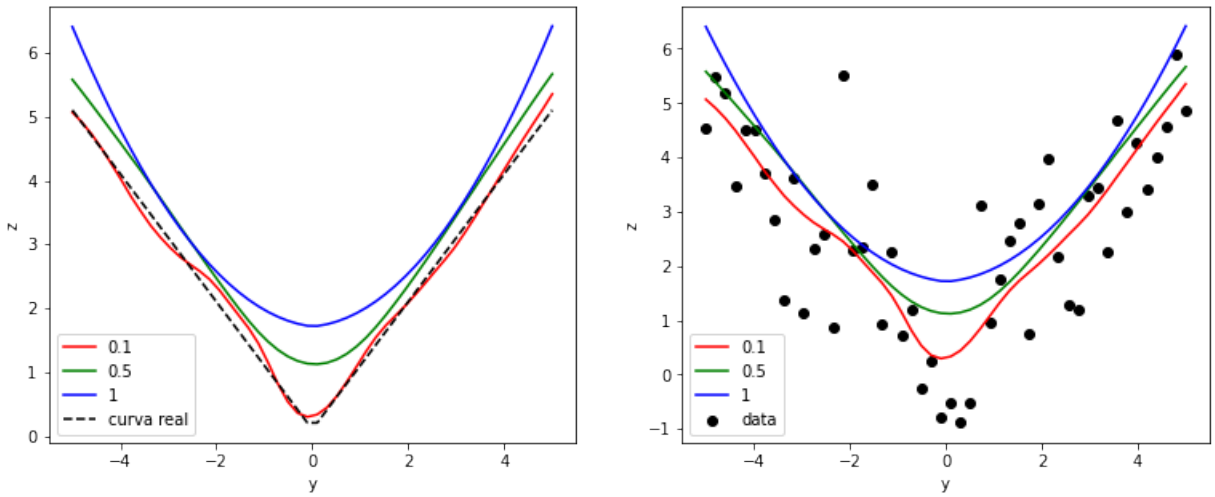
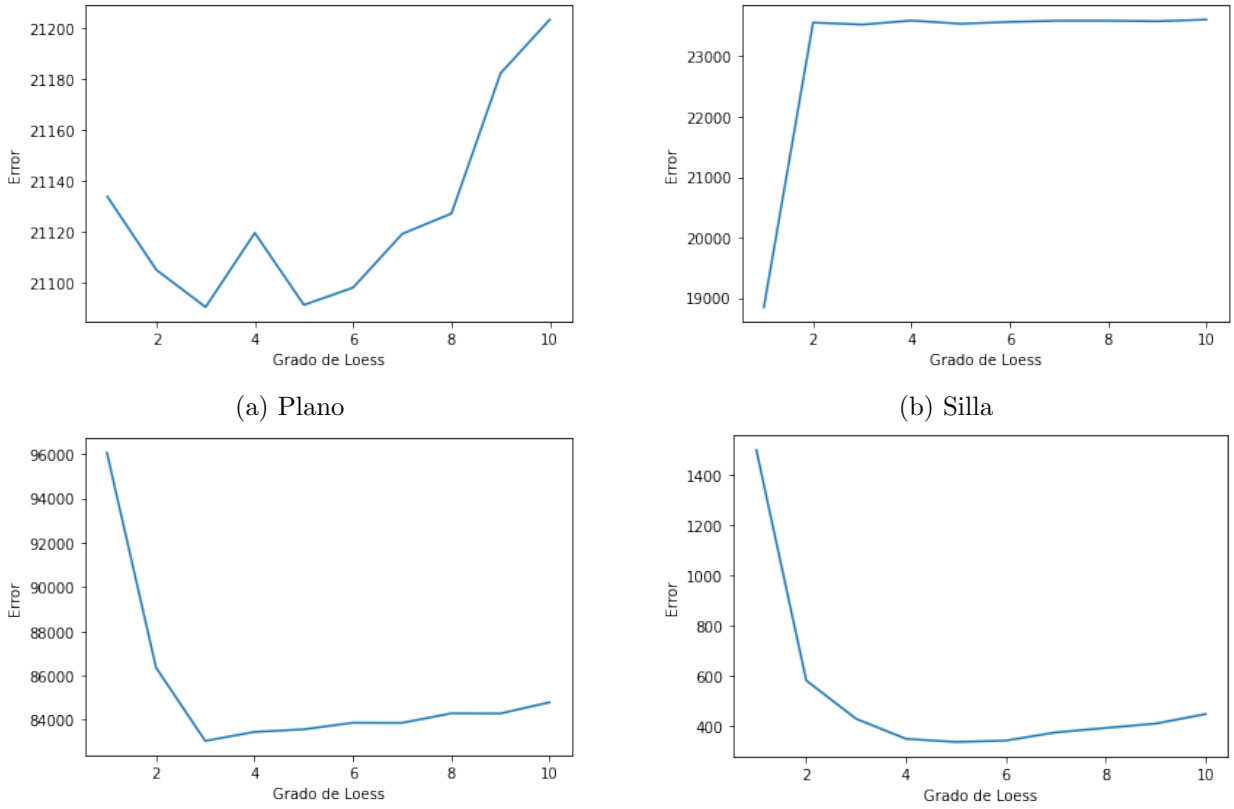


Figura 20: Curva de nivel de la cúbica con x fijo

En las Figuras 19 y 20 podemos ver que en el caso de la superficie de módulos, la cual contiene bordes afilados, aumentar el f nos da una peor estimación. Esto se debe a que, como mencionamos anteriormente, aumentar el f produce un efecto de suavizado sobre la superficie estimada, y al ser la original afilada, mientras más suave peor.

3.2.5. Medición de error al variar el grado de Loess

Fijando $f = 0{,}5$, medimos el error entre la función esperada y la predicción de Loess para las superficies mencionadas anteriormente. En particular, solo vamos a variar el grado de Loess para este experimento. Primero se debería definir una función de error: en nuestro caso la definimos arbitrariamente como la suma de la distancia euclidiana entre el punto esperado y el punto estimado por Loess.



(a) Plano

(b) Silla

(c) Cúbica

(d) Modulo

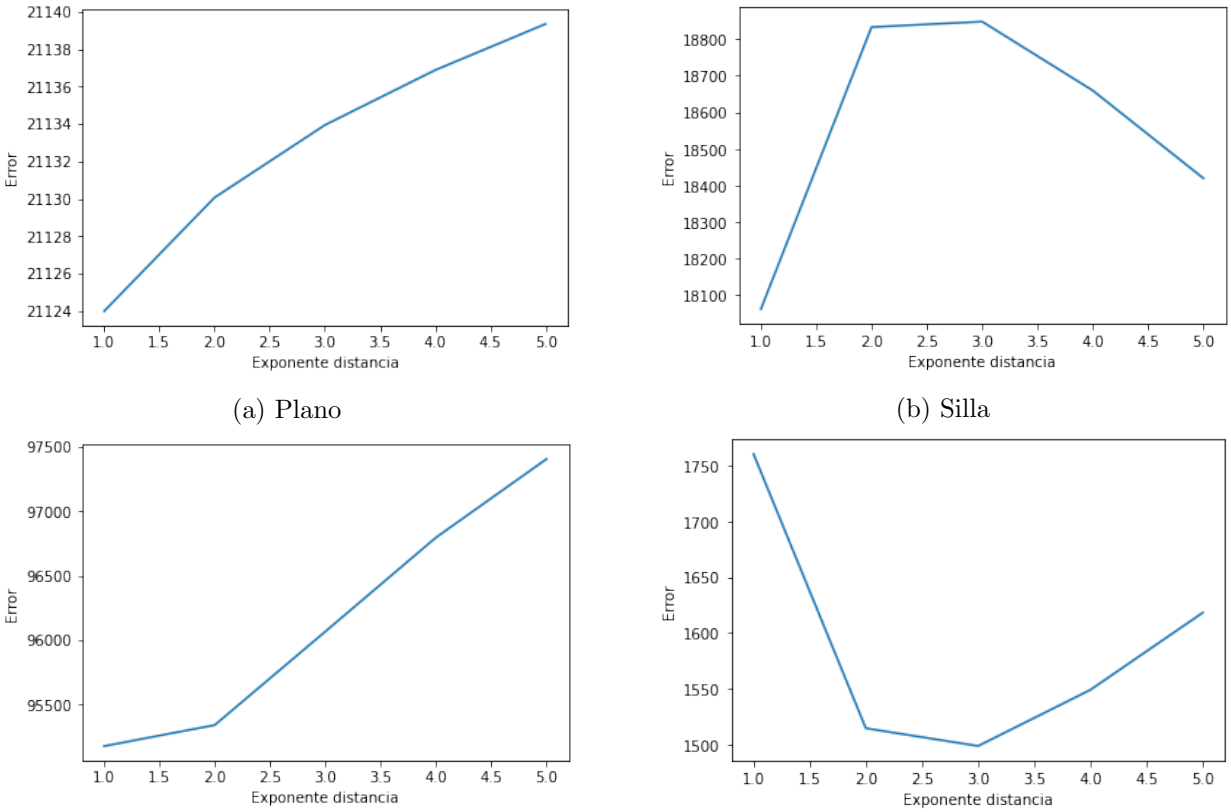
Figura 21: Evolución del error a medida que se va modificando el grado de Loess para distintas funciones

En la figura 21 vemos que la solución óptima para estas funciones simples suele estar en los primeros grados y a medida que aumentamos el grado suele empeorar notablemente una vez superado el valor óptimo. Suponemos que la solución óptima suele estar en grados bajos porque estamos expresando funciones relativamente simples. En particular, si tuviéramos que predecir los valores dada una función expresada por un polinomio de grado alto seguramente el grado óptimo resulte mayor que los vistos acá.

Algunos resultado fueron relativamente inesperados. Por ejemplo en la figura 21b la solución óptima es con grado 1 y luego crece el error considerablemente para grados mayores al óptimo. Uno supondría que estimando con grado 2 se tendría aproximadamente un error moderadamente equivalente al de grado 1. Otro caso es el de la figura 21a: vemos que estimando con grado 3 y 5 se tiene un error muy parecido, pero hay un pico creciente en la estimación con grado 4. Nuevamente no estamos seguros de por qué sucede esto.

3.2.6. Medición de error al modificar la función de distancia

En este experimento evaluamos cómo varía el error con distintas funciones de distancia. En particular, teníamos en el paper la función de distancia *tri-cúbica* definida por $W(u) = (1 - u^3)^3$ si $u \in [0, 1)$ y 0 sino. Decidimos probar las funciones de distancia definidas por $W_n(u) = (1 - u^n)^n$. Similarmente a como hicimos en el experimento anterior, vamos a experimentar con distintos valores de n para evaluar como va variando el error, tomando la misma función de error.



(a) Plano

(b) Silla

(c) Cúbica

(d) Modulo

Figura 22: Evolución del error para distintas funciones de error *n-cúbicas*

Observando los gráficos de la Figura 22, pareciera que no hay siquiera un candidato a mejor función de distancia. Pero si observamos los valores del error vemos que varía muy poco entre el mínimo y el máximo, apenas un pequeño porcentaje del error total para la mayoría de las funciones evaluadas. La única excepción es la Figura 22d, en la que hay una diferencia mayor al 10 % entre el error máximo evaluado y el menor. Curiosamente, la función de distancia óptima para este caso es la *tri-cúbica* que es justamente la que recomienda el paper visto.

4. Herramientas de visualización

4.1. QQ Plot

El QQ (Cuantil-Cuantil) plot es un gráfico utilizado para comparar dos distribuciones de probabilidad. En el eje de las Y se colocan los cuantiles de la distribución a analizar, mientras que en las X se la compara con los cuantiles de una distribución ya conocida. En caso de que ambas distribuciones tengan la misma forma, el gráfico de QQ plot responderá al de una ecuación lineal de $Y = X$. El sentido de este fenómeno es que en ambos, la distribución de los cuantiles sera la misma.

El principal objetivo de este método es entender qué forma toman los datos en base a distribuciones ya conocidas. Por ende mientras más lineal sea el gráfico de QQ plot, más cercana sera a la distribución a analizar a la propuesta en el eje de las X. Un ejemplo de esto es la Figura 3a , en donde se comparan los cuantiles residuales del ozono con los cuantiles de una distribución normal. Los resultados fueron los esperados, obteniendo una recta muy cercana a $Y = X$. Esto entonces significa que en ese caso los residuos del Ozono tienen un comportamiento similar al de una distribución normal, tal como se suponía. Por otro lado se puede mirar la Figura 23 [1], en la que se compara los cuantiles de una distribución exponencial contra los de una normal. Se puede ver claramente una lejanía con la recta Y=X, mostrando que una distribución normal no es apropiada para explicar esos datos.

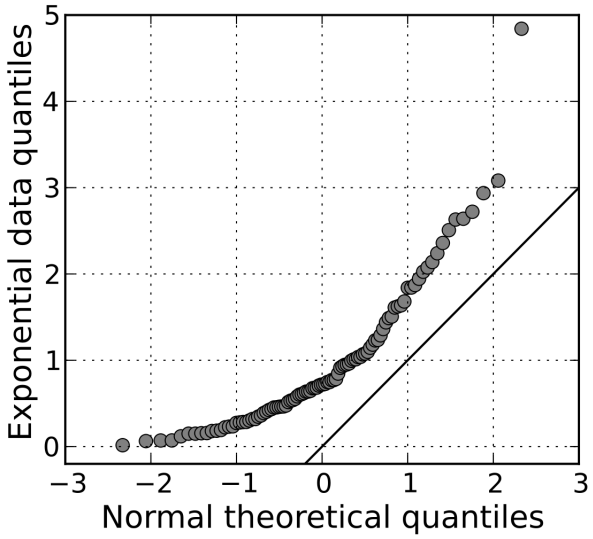


Figura 23: Distribución exponencial vs. Normal

Una aclaración que es importante hacer de estos gráficos es que su principal utilidad radica en sacar primeras conclusiones respecto de los datos que se están analizando, para tener una intuición sobre que forman toman, pero no proporcionan información mucho más profunda que esa.

[1] https://en.wikipedia.org/wiki/QQplot

4.2. Residual vs. Fitted

En los gráficos de Residual vs. Fitted se analiza los valores residual de unos datos en el eje de las Y, y se los compara con sus valores estimados en el eje de las X. El objetivo de estos gráficos es saber si la regresión propuesta para describir esos datos describe apropiadamente un comportamiento lineal, o presenta variaciones en diferentes puntos. En caso de que se trate de que la regresión lineal propuesta describa precisamente los datos analizados, un gráfico de Residual vs. Fitted debería tomar la forma de la Figura 24 [2]. Esta figura proviene de un experimento realizado tras encuestar a 50 personas para averiguar si existía una relación lineal entre cantidad de alcohol consumida y fuerza. De este experimento se realizo un estimador lineal que lo representase adecuadamente, y finalmente se realizo un grafico de residual vs. fitted dando como el resultado la Figura 24 . Lo que se puede destacar de esta figura es que los datos están completamente distribuidos a lo largo del gráfico de forma independiente y sin sesgo. Esto contrasta con el gráfico de la Figura 3b en donde se ve que los datos no están distribuidos uniformemente y por ende hay grados de varianza dentro de los datos.

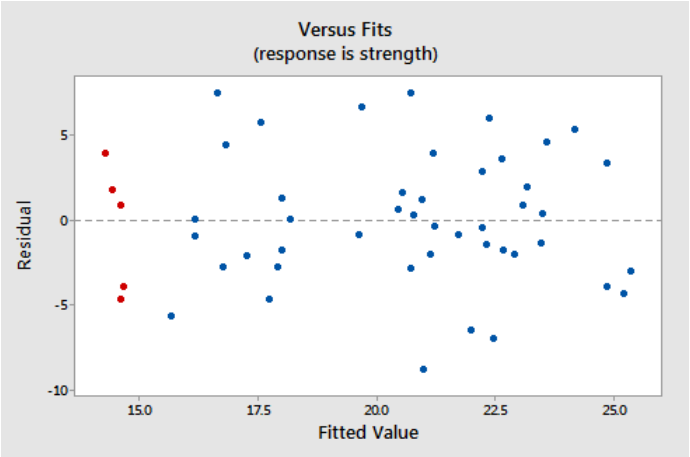


Figura 24: Residual vs. Fitted ideal

Una vez más, por fuera de la capacidad de saber cual es el grado de linealidad de los datos, estos gráficos sirven para tener una primera intuición sobre si es apropiado o no aplicar un método linear, pero no más que eso.

4.3. Component-residual Plot

En un gráfico de component-residual se intenta analizar la linealidad de los datos y valores dependientes del mismo. Por ende un factor que se busca ver es si el predictor lineal de los datos describe apropiadamente los valores de variables dependientes. La forma de armar este tipo de gráficos es poner en base al eje de las Y los valores residuales contra en el eje de las X los valores de las variables dependientes. Hasta este punto sería un partial residual plot, pero se le agrega una linea que indica donde se ubica el mejor estimador. En caso de que haya una discrepancia grande entre el estimador de la variable dependiente y el estimador del valor residual, quiere decir que el estimador propuesto no tiene una buena relación con respecto a la variable dependiente. Para entender mejor este fenómeno, se propone mirar el comportamiento en una función que se compone de tres variables, una con datos distribuidos de forma lineal, otra cuadrática y una logarítmica. En caso de que se quisiese hacer una component-residual plot, en donde el estimador residual es uno lineal, los gráficas tomarían la forma de la Figura 25 [3]. Este experimento fue realizado con funciones en R, para generar las variables dependiente con sus formas correspondientes.

[2] https://online.stat.psu.edu/stat501/lesson/4/4.2
[3] https://www.r-bloggers.com/2012/01/r-regression-diagnostics-part-1/

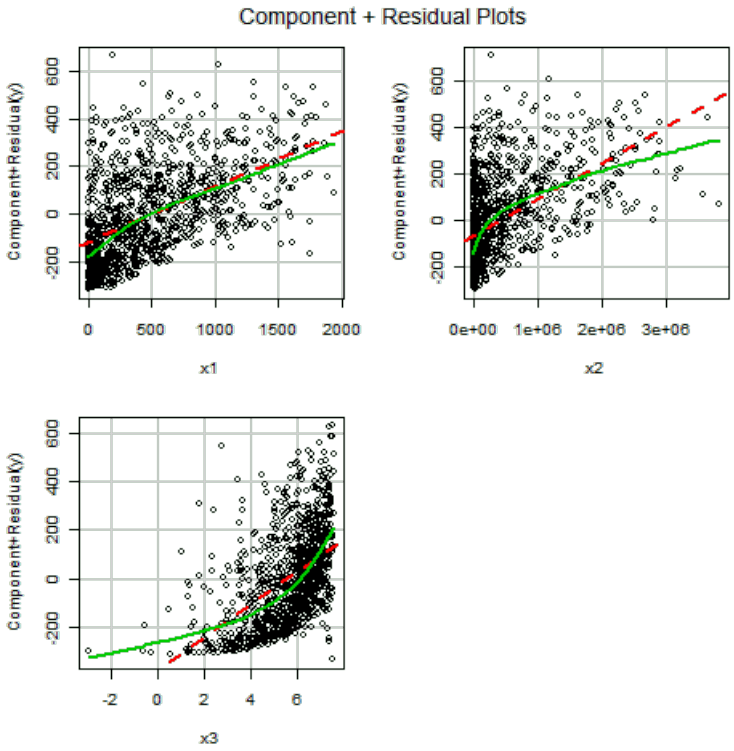


Figura 25: component-residual Plots

Como se puede observar, en los casos de la variable cuadrática y logarítmica, el estimador residual lineal presenta muchas discrepancias. Por ende una solución para esto puede ser aplicar transformaciones a los datos, para que estos se puedan acomodar más fácilmente al estimador residual. Como en este caso las distribuciones ya son conocidas, una solución podría ser aplicar la raíz a los datos de la variable x_2 y elevar a e por los datos de la variable x_3, de hacer así, un posible resultado sería la figura 26 [4]. Como se puede ver, en estos resultados la linealidad es mucho más evidente. Aunque no sea una superposición total, los datos de estas variables ahora son mucho mejor estimados por el estimador residual.

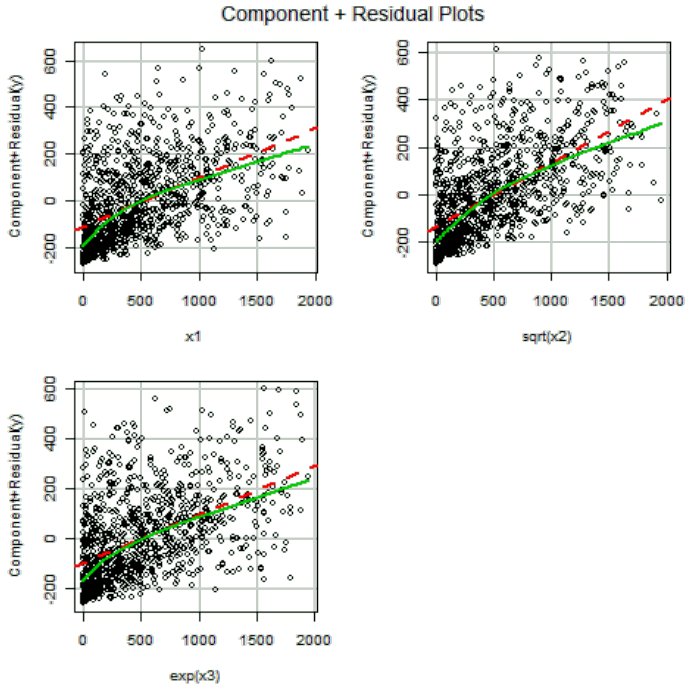


Figura 26: component-residual Plots modificados

[4] https://www.r-bloggers.com/2012/01/r-regression-diagnostics-part-1/

5. Conclusiones

La experimentación fue realizada con datos sintéticos, lo cual nos ayudó a observar el comportamiento del algoritmo ante diversos casos y cambios de parámetros. A continuación detallaremos las conclusiones que extraímos de dichos experimentos, y mencionaremos la investigación posterior que recomendamos efectuar para ampliar este informe.

Primero experimentamos con el parámetro f, el cuál indica qué porcentaje de datos (cantidad de vecinos) se usan para cada estimación. En todas las Figuras pudimos observar que tal como mencionan en el Paper [1], mientras más grande el f, la estimación resultante es más suave. Además, en esos experimentos notamos que tomar un f muy chico no suele ser buena idea. Las estimaciones que obtuvimos cuando usamos una poca cantidad de vecinos fueron muy ruidosas. Concluimos que con f chicos la estimación es demasiado local. Si dentro de un pequeño vecindario se da la casualidad de que haya varios outliers o uno muy grande, eso se verá reflejado en la estimación, obteniendo peores resultados. El único caso en el que usar un f muy chico dió mejores resultados fue en la superficie de módulos. Se podría decir que cuando las curvas de nivel de los datos de entrada son afiladas, el algoritmo se comporta mejor con un f chico, sin emabrgo este método está hecho para generar curvas suaves, por lo que tampoco sería muy sensato utilizar este algoritmo para esos casos. Por otro lado, cuando el f es muy grande, se puede dar el caso en el que la estimación generada sea demasiado suave. Esto lo observamos en el caso de la cúbica, en la cual obtuvimos mejores resultados para f más chicos. La ventaja de un f muy grande la encontramos en los casos donde las superficies o curvas de nivel eran lineales, ya que mientras más suave la estimación, mejor. Generalizando los resultados obtenidos, las mejores estimaciones las obtuvimos con $f = 0{,}5$, sin embargo, recomendamos observar el comportamiento de las mediciones, ya que dependiendo de la forma y la cantidad de ruido puede cambiar el f óptimo.

El segundo parámetro con el que experimentamos fue el grado. En los experimentos notamos que se suele comportar mejor con grados bajos. Sospechamos que esto puede ser debido a que las funciones con las que experimentamos son simples, y que con funciones más complejas los grados bajos no sean tan efectivos como los más altos. Basándonos en dichos experimentos no podemos afirmar que haya un grado para el cual el algoritmo se comporte de manera óptima, ya que el resultado fue muy diferente para cada caso. Proponemos investigar a futuro por qué sucede lo mencionado, evaluando nuevamente para distintas funciones más complejas, mayores dimensiones y distintos parámetros para ver si se encuentra una tendencia.

Para el último experimento consideramos interesante observar qué pasa cuando se cambia la función de distancia. En el Paper [1] se propuso la tricubica, lo cuál esperábamos obtener mejores estimaciones usando esa función. Nosotros propusimos funciones similares, en las cuales cambiamos los exponentes. Una vez calculado el error para cada uno de nuestros datasets, usando diferentes exponentes en la función de distancia, notamos que la función tricubica fue óptima en solo en uno de los cuatro casos. De todas formas, notamos que el cambio en el error al cambiar la función fue muy chico. Concluimos que no es un parámetro de los más influyentes en la estimación. No obstante, consideramos interesante continuar la investigación con más funciones, para confirmar que la tricubica sea la función óptima.

6. Bibliografía

[1] William S. Cleveland and Susan J. Devlin (1988) Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting

[2] William S. Cleveland (1979) Robust Locally Weighted Regression and Smoothing Scatterplots, Journal of the American Statistical Association, 74:368, 829-836