Tarea 5: Introducción a Ciencia de Datos.

Rojas Gutiérrez Rodolfo Emmanuel

24 de septiembre de 2021

1. Ejercicio

Ejercicio 1:

> Considere el conjunto de datos 'datos.csv':
>
> a) Ajuste (en R o Python) un modelo de regresión lineal a estos datos haciendo uso del procedimiento de gradiente estocástico. Estudie diferentes alternativas para el tamaño de paso.
>
> b) Investigue acerca de versiones de gradiente estocástico que usan el método de momentos e implementelo en R o Python, aplicado a datos çsv".

a)

Solución. El archivo 'datos.csv' contiene 21454 observaciones de una variable respuesta, la cuál se denotará por y, además de que cuenta con el mismo número de observaciones para 22 variables predictoras.[1] Se destaca que todas las columnas en este conjunto de datos son de tipo numérico y que no existen $NA's$ en este conjunto, adicionalmente, es importante mencionar que todas las filas suman cero,[2] esto porque lo anterior indica que el modelo a ajustar no debe de considerar un intercepto. Lo anterior, se puede corroborar en el resumen realizado en R del mismo, el cual se encuentra al inicio del Script adjunto a este trabajo. Ahora, denote por X a una matriz en $M_{21454\times 22}(\mathbb{R})$ tal que

- Las 22 columnas de X coinciden con las columnas 2 a 23 de la tabla 'datos.csv'.

Y, sea y la matriz columna en $M_{21454\times 1}(\mathbb{R})$ cuya única columna, coincide con la última columna de la tabla 'datos.csv'. Entonces, se busca ajustar un modelo de regresión lineal de la forma

$$y = X\beta + \varepsilon,$$

donde, β es un vector columna de parámetros en $M_{22\times 1}(\mathbb{R})$ y ε es un vector columna de términos de error, que toma valores en $M_{22\times 1}(\mathbb{R})$. Ahora, en R se corroboró que X es una matriz de rango completo, pues, el rango de dicha matriz es 22. Por ende, se decidió estimar el vector de parámetros

[1] Ya que, se nos pidió remover la primera columna.

[2] Al quitar la primer columna del conjunto de datos y sin considerar la columna de la variable respuesta.

β por mínimos cuadrados con el objetivo de comparar resultados, además de servir como un criterio de paro óptimo para nuestros algoritmos, como se verá más adelante. Los resultados obtenidos por este método se exponen a continuación:[3]

$$\widetilde{\beta} = (X'X)^{-1}X'y = \begin{pmatrix} 0.306 & 0.028 & \dots & -0.083 & -0.063 \end{pmatrix}'. \tag{1}$$

Luego, esta estimación para β obtiene una suma de cuadrados residual de:

$$S_{opt} = f(\widetilde{\beta}) = \left\| y - X\widetilde{\beta} \right\|^2 = 21.646. \tag{2}$$

Por otro lado, se explicará el como se empleo el algoritmo de Descenso Por Gradiente Estocástico (SGD), primero, se debe tener una función de interés para minimizar, en este caso, la suma de cuadrados residual del modelo que se quiere ajustar:

$$\hat{\beta} = \text{argmin}_{\beta \in M_{22\times 1}(\mathbb{R})} \|y - X\beta\|, \text{ donde } \|\cdot\| \text{ representa a la norma Euclideana en } \mathbb{R}^{21454},$$

Con esto en mente, defínase la función $f : M_{22\times 1}(\mathbb{R}) \to \mathbb{R}_+$ como

$$f(\beta) = \|y - X\beta\|^2, \text{ donde } \beta = \begin{pmatrix} \beta_1 & \beta_2 & \dots & \beta_{22} \end{pmatrix}.$$

Luego, si se denota por X_{ij} y y_{i1} con $i \in \{1, \dots, 21454\}$ y $j \in \{1, \dots, 22\}$ a las entradas de X y y respectivamente, entonces, la ecuación anterior puede expresarse de manera equivalente como:

$$f(\beta) = \sum_{i=1}^{21454} \left[y_i - \sum_{j=1}^{22} \beta_j X_{ij} \right]^2.$$

Así, sea $k \in \{1, \dots, 22\}$ entonces derivando f con respecto a β_k, se obtiene:

$$\frac{\partial}{\partial \beta_k} f = -2 \sum_{i=1}^{21454} X_{ik} \left[y_i - \sum_{j=1}^{22} \beta_j X_{ij} \right].$$

Por lo tanto, el gradiente de la función f esta dado por

$$\nabla f(\beta)_{k1} = \sum_{i=1}^{21454} 2X_{ik} \left[-y_i + \sum_{j=1}^{22} \beta_j X_{ij} \right], \text{ con } k \in \{1, \dots, 22\},$$

Finalmente, el método SGD consiste en seleccionar al azar en cada iteración una observación $i^* \in \{1, \dots, 21454\}$, y actualizar la estimación hecha con las iteraciones pasadas de la siguiente forma:

$$\hat{\beta}_k^{nuevo} \leftarrow \hat{\beta}_k^{viejo} - 2\gamma X_{i^*k} \left[-y_i^* + \sum_{j=1}^{22} \beta_j^{viejo} X_{i^*j} \right], \text{ con } k \in \{1, \dots, 22\}.$$

Es decir, solo se evalúa una componente de la suma en el gradiente en cada iteración, la cual se selecciona al azar. Así, el algoritmo a emplear será el siguiente:

[3]Por cuestiones de espacio no se puso el vector completo, pero el mismo puede ser consultado en el script en R.

1. Si $n = 0$, se selecciona $\hat{\beta}_k^{(n)} = (X'y)_{k1}$ con $k \in \{1, \ldots, 22\}$.

2. Se elige al azar un índice i^* en el conjunto $\{1, \ldots, 21454\}$, que no se haya seleccionado en los n pasos anteriores, y se actualiza la estimación del vector de parámetros β de la siguiente forma:
$$\hat{\beta}_k^{(n+1)} \leftarrow \hat{\beta}_k^{(n)} - 2\gamma X_{i^*k} \left[-y_i^* + \sum_{j=1}^{22} \beta_j^{(n)} X_{i^*j} \right], \text{ con } k \in \{1, \ldots, 22\}.$$
donde, γ es el tamaño de paso el cual se debe determinar.

3. Se calcula la suma de cuadrados residual asociada a esta estimación de β, la cual se denotará por SCR_{n+1}, y esta dada por:
$$SCR_{n+1} = f(\hat{\beta}^{(n+1)}) = \sum_{i=1}^{21454} \left[y_i - \sum_{j=1}^{22} \hat{\beta}_j^{(n+1)} X_{ij} \right]^2,$$
donde, $(\hat{\beta}^{(n+1)})_{k1} = \hat{\beta}_k^{(n+1)}$ para cada $k \in \{1, \ldots, 22\}$. Si
$$|SCR_{n+1} - S_{opt}| < 0.01 \text{ o } n + 1 > 1500,$$
con S_{opt} como en (2). Entonces, el algoritmo acaba y la estimación para β es $\hat{\beta} = \hat{\beta}^{(n+1)}$, en otro caso, se repite todo desde el paso 2 intercambiando el papel de n por $n + 1$.

Finalmente, se verá que la elección del tamaño de paso es muy relevante. Para ello, se programó el algoritmo anteriormente estructurado en R y se probó el mismo con diversos tamaños de paso, obteniendo 3 casos que se cree son suficientemente relevantes como para ser comentados. El primero de ellos fue eligiendo $\gamma = 5$, lo que se encontró en este caso puede resumirse en el gráfico presentado en la Figura 1, en el cual se puede observar el valor de la suma de cuadrados residual, para cada una de las estimaciones de β hecha en cada una de las 1501 iteraciones,[4] en dicha gráfica se puede ver que la suma de cuadrados residual parece hacerse arbitrariamente grande conforme avanzan las iteraciones. En un principio no se encontró una explicación lógica a esto, pero, al intentar con otros tamaños de paso la situación se clarifico un poco como se verá más adelante.

Otra caso importante, fue cuando se eligió como tamaño de paso a $\gamma = 10^{-10}$, el resultado de esto puede ser ilustrado en el gráfico presentado en la Figura 2, en el cual se puede observar el valor de la suma de cuadrados residual, para cada una de las estimaciones de β hechas a través de 1501 iteraciones del algoritmo.[5] En dicho gráfico, puede observarse que en este caso el algoritmo parece permanecer estático, pues, la suma de cuadrados residual es casi constante a lo largo de las iteraciones. Esto puede deberse a que el tamaño de paso se selecciono tan pequeño, que realmente no se esta restando prácticamente nada en cada iteración del algoritmo, cuando este se encuentra actualizando el vector de parámetros en cada paso.[6] De hecho, se calculó la máxima diferencia en valor absoluto entre las entradas de la semilla inicial $\hat{\beta}^{(0)}$ y el último valor calculado por el algoritmo $\hat{\beta}^{(1501)}$, lo que dio un resultado de 0.353 que es una cantidad bastante cercana a 0. Lo anterior en conjunto con el primer caso analizado, sugieren que el elegir tamaños de paso muy pequeños harán

[4] Es decir, nunca se alcanzo el criterio de paro óptimo establecido, lo cuál es lógica en vista de los resultados obtenidos.

[5] Nuevamente, nunca se alcanzo el criterio de paro óptimo establecido.

[6] Paso 3. del algoritmo planteado.

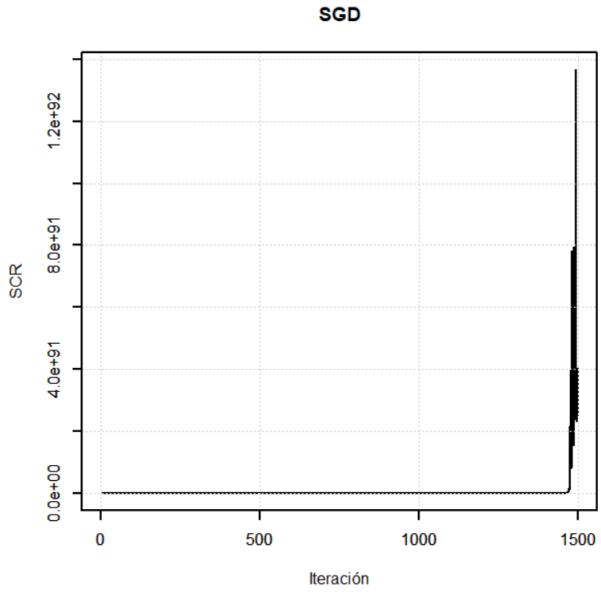

Figura 1: Método del Gradiente estocástico con tamaño de paso $\gamma = 5$.

que el método se quede estático, mientras que, el elegir tamaños de paso muy grandes harán que el método diverja. Por ende, se decidió elegir un tamaño de paso de $\gamma = 2 \cdot 10^{-1}$ el cual se encuentra entre los valores propuestos con anterioridad. El resultado se puede resumir en el gráfico presentado en la Figura 3, como lo sugiere dicho gráfico el método no convergió bajo el criterio de optimalidad establecido, no obstante, después de 1501 iteraciones se obtuvieron resultados mucho más acertados que en los casos anteriores, como se deduce la imagen. Por ejemplo, la suma de cuadrados residual obtenida por este método fue de.

$$SCR_{1501} = 23.6203.$$

Lo que es bastante similar al valor que se obtiene al hacer uso de Mínimos Cuadrados.[7] Por otro lado, la diferencia en valores absolutos más grande entre las entradas del vector $\hat{\beta}^{(1501)}$, estimado mediante el método SGD para este tamaño de paso, y el vector $\widetilde{\beta}$ construido con mínimos cuadrados es de 0.181,[8] lo que nos da una idea de la cercanía de estos vectores.

■

b)

Solución. Finalmente, para este segundo inciso se investigaron dos algoritmos adicionales, los cuales llevan por nombre Descenso por Gradiente Estocástico con Momentum (SGDM) y Método de Estimación Adaptada por Momentos (ADAM), los cuales agregan estimadores de la media y el segundo momento no centrado del gradiente al proceso iterativo. Dichos estimadores de los momentos, se

[7] Ver (2).

[8] Por cuestiones de espacio, no se incluyo en este documento el vector de parámetros estimado, pero, el mismo puede consultarse en el script adjunto.

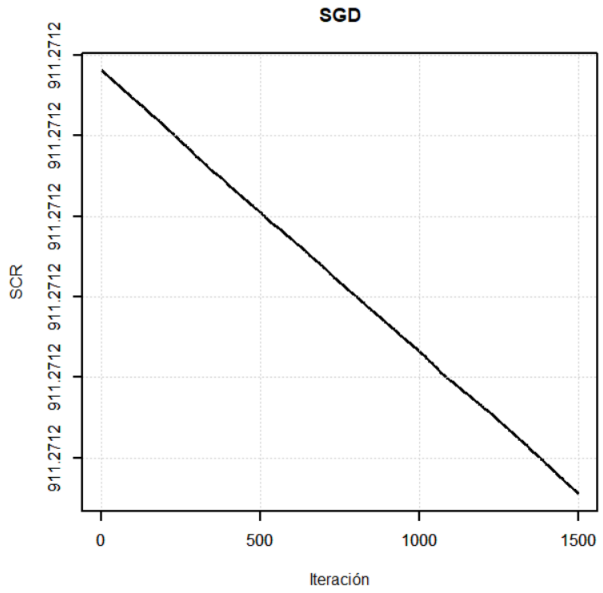

Figura 2: Método del Gradiente estocástico con tamaño de paso $\gamma = 10^{-10}$.

construyen con los rezagos de los gradientes estimados en las iteraciones anteriores. La idea que hay detrás de esto es sencilla, en SGD no se calcula exactamente el valor del gradiente, pues, como ya hemos visto anteriormente el gradiente solo se evalúa en cierto subconjunto del total de datos, por ende, al agregar las estimaciones mencionados en el proceso iterativo se puede mantener mayor precisión en la relación que existe entre el gradiente real, que debería ser calculado en la n-ésima iteración, y el gradiente que es calculado en realidad en dicha iteración del algoritmo, con lo que, al menos en principio se piensa que la convergencia del método debería ser más rápida. A continuación se lista, el como se aplicarían los algoritmos mencionados al conjunto de datos 'datos.csv'. Primero, empezaremos con SGDM:

1. Si $n = 0$, se selecciona $\hat{\beta}_k^{(n)} = (X'y)_{k1}$ con $k \in \{1, \ldots, 22\}$ y se define $m^{(0)} \in M_{22\times 1}(\mathbb{R})$ con todas sus entradas iguales a cero.

2. Se elige al azar un índice i^* en el conjunto $\{1, \ldots, 21454\}$, que no se haya seleccionado en los n pasos anteriores, y se calcula el gradiente aproximado en i^* como

$$g_k^{(n+1)} \leftarrow 2X_{i^*k}\left[-y_i^* + \sum_{j=1}^{22} \beta_j^{(n)} X_{i^*j}\right], \text{ con } k \in \{1, \ldots, 22\}.$$

   Posteriormente, se construye el estimador para la media del gradiente como:

$$m_k^{(n+1)} \leftarrow \iota_1 m_k^{(n)} + (1-\iota_1) g_k^{(n+1)}, \text{ con } k \in \{1, \ldots, 22\},$$

   donde, ι_1 es un hiper-parámetro en $(0, 1)$ que debe ser seleccionado al inicio del proceso.

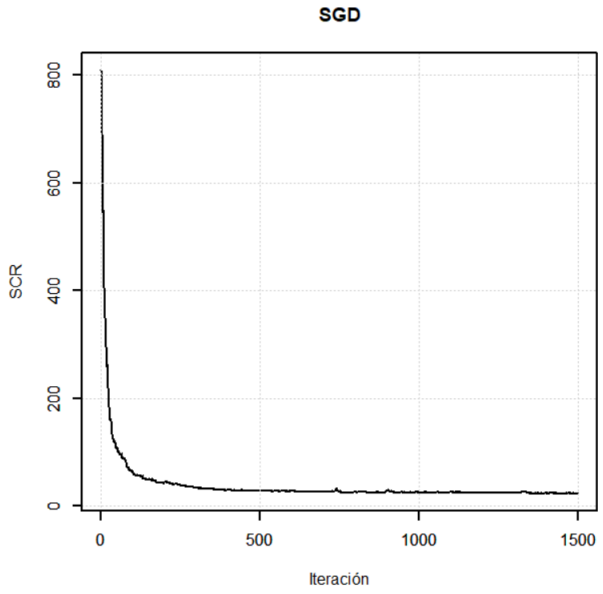

Figura 3: Método del Gradiente estocástico con tamaño de paso $\gamma = 2 \cdot 10^{-1}$.

Luego, se actualiza la estimación del vector de parámetros β de la siguiente forma:

$$\hat{\beta}_k^{(n+1)} \leftarrow \hat{\beta}_k^{(n)} - \gamma m_k^{(n+1)}, \text{ con } k \in \{1, \ldots, 22\},$$

donde, γ es el tamaño de paso cuyo valor se debe determinar al iniciar el algoritmo.

3. Se calcula la suma de cuadrados residual asociada a esta estimación de β, la cual se denotará por SCR_{n+1}, y esta dada por:

$$SCR_{n+1} = f(\hat{\beta}^{(n+1)}) = \sum_{i=1}^{21454} \left[y_i - \sum_{j=1}^{22} \hat{\beta}_j^{(n+1)} X_{ij} \right]^2,$$

donde, $(\hat{\beta}^{(n+1)})_{k1} = \hat{\beta}_k^{(n+1)}$ para cada $k \in \{1, \ldots, 22\}$. Si

$$|SCR_{n+1} - S_{opt}| < 0.01 \text{ ó } n + 1 > 1500,$$

donde, S_{opt} esta dado en (2). Entonces, el algoritmo acaba y la estimación para β es $\hat{\beta} = \hat{\beta}^{(n+1)}$, en otro caso, se repite todo desde el paso 2 intercambiando el papel de n por $n + 1$.

Mientras que, el algoritmo para el método ADAM quedaría descrito de la siguiente manera:

1. Si $n = 0$ se selecciona $\hat{\beta}_k^{(n)} = (X'y)_{k1}$ con $k \in \{1, \ldots, 22\}$ y se definen $m^{(0)}, v^{(0)} \in M_{22\times 1}(\mathbb{R})$, iguales al elemento 0 en $M_{22\times 1}(\mathbb{R})$.

2. Se elige al azar un índice i^* en el conjunto $\{1,\ldots,21454\}$, que no se haya seleccionado en los n pasos anteriores, y se calcula el gradiente aproximado en i^* como:

$$g_k^{(n+1)} \leftarrow 2X_{i^*k}\left[-y_i^* + \sum_{j=1}^{22}\beta_j^{(n)}X_{i^*j}\right],\ \text{con } k\in\{1,\ldots,22\}$$

Posteriormente, se construyen los estimadores para la media y el segundo momento del gradiente, como:

$$m_k^{(n+1)} \leftarrow \iota_1 m_k^{(n)} + (1-\iota_1)g_k^{(n+1)},\ \text{con } k\in\{1,\ldots,22\},$$
$$v_k^{(n+1)} \leftarrow \iota_2 v_k^{(n)} + (1-\iota_2)(g_k^{(n+1)})^2,\ \text{con } k\in\{1,\ldots,22\},$$

donde, ι_1 y ι_2 son hiper-parámetros en $(0,1)$ que deben ser seleccionados al inicio del proceso. Luego, los estimadores anteriores se corrigen al multiplicarlos por una cantidad denominada como factor de corrección del sesgo, de la siguiente manera:

$$\hat{m}_k^{(n+1)} \leftarrow \frac{m_k^{(n+1)}}{1-\iota_1^n},\ \text{con } k\in\{1,\ldots,22\},$$
$$\hat{v}_k^{(n+1)} \leftarrow \frac{v_k^{(n+1)}}{1-\iota_2^n},\ \text{con } k\in\{1,\ldots,22\}.$$

Finalmente, se actualiza la estimación del vector de parámetros β de la siguiente forma:

$$\hat{\beta}_k^{(n+1)} \leftarrow \hat{\beta}_k^{(n)} - \gamma\frac{\hat{m}_k^{(n+1)}}{\sqrt{\hat{v}_k^{(n+1)}}+\varepsilon},\ \text{con } k\in\{1,\ldots,22\}.$$

donde, γ es el tamaño de paso cuyo valor se debe determinar al iniciar el algoritmo y $\varepsilon>0$, es una cantidad muy pequeña que se elige con el objetivo de evitar divisiones por cero.

3. Se calcula la suma de cuadrados residual asociada a esta estimación de β, la cual se denotará por SCR_{n+1}, y esta dada por:

$$SCR_{n+1} = f(\hat{\beta}^{(n+1)}) = \sum_{i=1}^{21454}\left[y_i - \sum_{j=1}^{22}\hat{\beta}_j^{(n+1)}X_{ij}\right]^2,$$

donde, $(\hat{\beta}^{(n+1)})_{k1} = \hat{\beta}_k^{(n+1)}$ para cada $k\in\{1,\ldots,22\}$. Si

$$|SCR_{n+1} - S_{opt}| < 0.01 \text{ o } n+1 > 1500.$$

donde, S_{opt} esta dado en (2). Entonces, el algoritmo acaba y la estimación para β es $\hat{\beta} = \hat{\beta}^{(n+1)}$, en otro caso, se repite todo desde el paso 2 intercambiando el papel de n por $n+1$.

Ahora, hubo un problema al intentar aplicar ADAM, pues, este algoritmo requiere de una gran cantidad de hiper-parámetros para ser implementado. No obstante, fue relativamente fácil calibrar los hiper-parámetros para el método SGDM, la aplicación de este último método sobre los datos

en 'datos.csv' se realizó en R, el código de esta implementación puede consultarse en el script adjunto a este trabajo. Un breve resumen gráfico de lo acontecido al aplicar este algoritmo, similar al presentado en el inciso anterior cuando se selecciono el γ correcto, puede verse en la gráfica presentada en la figura 4.[9] Los hiper-parámetros que fueron usados fue $\iota_1 = 0.2$ y $\gamma = 5 \cdot 10^{-1}$. Adicionalmente, como el gráfico mencionado lo sugiere, el método no convergió bajo el criterio de optimalidad establecido después de 1501 iteraciones, no obstante, como se puede inferir de la imagen los resultados fueron alentadores, por ejemplo: La suma de cuadrados con el estimador $\hat{\beta}^{(1501)}$ arrojada por este método fue de:

$$SCR_{1501} = 23.529,$$

lo cual, es bastante cercano a la suma de cuadrados óptima (2). Finalmente, la diferencia en valor absoluto máxima entre las entradas de $\hat{\beta}^{(1501)}$ calculado con SGDM y $\widetilde{\beta}$ fue de 0.1085 lo cual es un valor cercano a cero. Por último, a pasar de que ninguno de los dos métodos convergió bajo

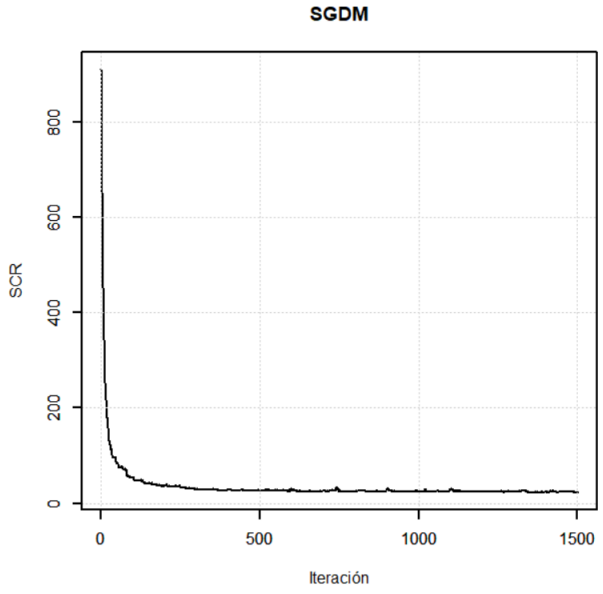

Figura 4: Método SGDM con $\iota_1 = 0.2$ y $\gamma = 5 \cdot 10^{-1}$.

el criterio de optimalidad establecido, se deja en la Figura 5 una gráfica comparativa entre ambos métodos, con el objetivo de ilustrar la diferencia en la velocidad en la que estos aproximan a la suma de cuadrados, más pequeña posible (2). En rojo puede ver los resultados obtenidos para SGD y en negro para SGDM, mientras que, en una linea verde se resalta el valor de la suma de cuadrados residual óptima (2). Observado este gráfico se podría pensar que SGDM en principio converge más rápido, no obstante parece que en el largo plazo la convergencia se vuelve un poco inestable, mientras que, con SGD la convergencia parece darse de manera más uniforme, no obstante, este

[9]Es importante resaltar que para poder hacer comparables los resultados de SGDM con los de SGD, se gráfico como en los casos anteriores la SCR de cada iteración más la SCR de la semilla inicial.

efecto puede deberse a la forma tan arbitraria en la que los hyper-parámetros de ambos modelos fueron elegidos. ■

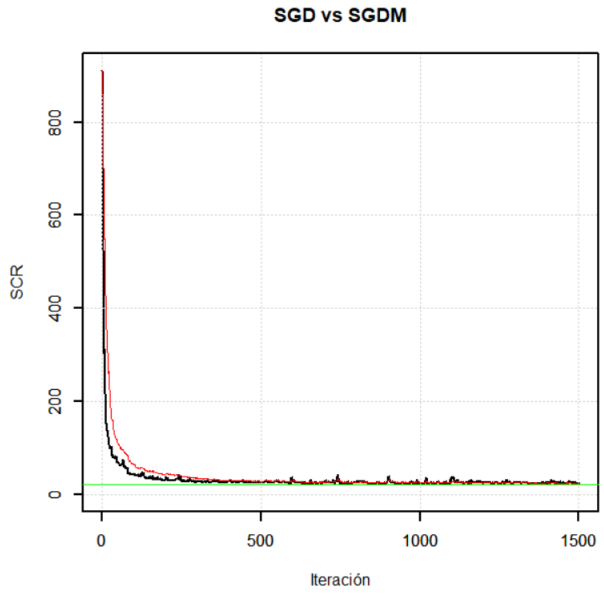

Figura 5: Método SGDM con $\iota_1 = 0.2$ y $\gamma = 5 \cdot 10^{-1}$ en linea negra, contra, método SGD con $\gamma = 2 \cdot 10^{-1}$ en linea roja, en linea verde se observa el valor de la suma de cuadrados residual óptima (2).

2. Anexo

En el script en R también puede encontrarse un intento adicional hecho con SGD, para la estimación de los parámetros del modelo, la diferencia radical entre el último caso reportado con un tamaño de paso $\gamma = 2 \cdot 10^{-1}$ en el inciso **a)** y el intento referido, es que en este último se eligió en lugar de un tamaño de paso γ fijo, un tamaño de paso que dependiera de la iteración en la que se encuentre el algoritmo tomando como $\gamma_{n+1} = \left|10^{-1} - 1/(n+1)\right|$ al tamaño de paso para la iteración $n+1$, $n \in \{0, \ldots, 1500\}$. Sin embargo, el motivo por el que no se reporto este caso en el trabajo principal, es porque no se obtuvieron mejorías sustanciales en el resultado al utilizar esta complicación. Para ver esto observe la Figura 6 en la cual se presentan las diferentes sumas de cuadrados residuales, producidas por la aplicación del método SGD al conjunto 'datos.csv' con tamaño de paso constante $\gamma = 2 \cdot 10^{-1}$ en linea roja, mientras que, en linea negra se presenta lo correspondiente a la aplicación del método con tamaño de paso variable, finalmente, en una linea de color verde se marca la suma de cuadrados óptima. En este gráfico puede verse que el método con el tamaño de paso constante, parece tener una convergencia más rápida al valor óptimo de la suma de cuadrados, más aún, el valor final para la suma de cuadrados residual obtenida, mediante

el modelo con tamaño de paso variable es

$$25.255,$$

lo cual es mayor a la suma de cuadrados reportada al final del inciso **a)**.

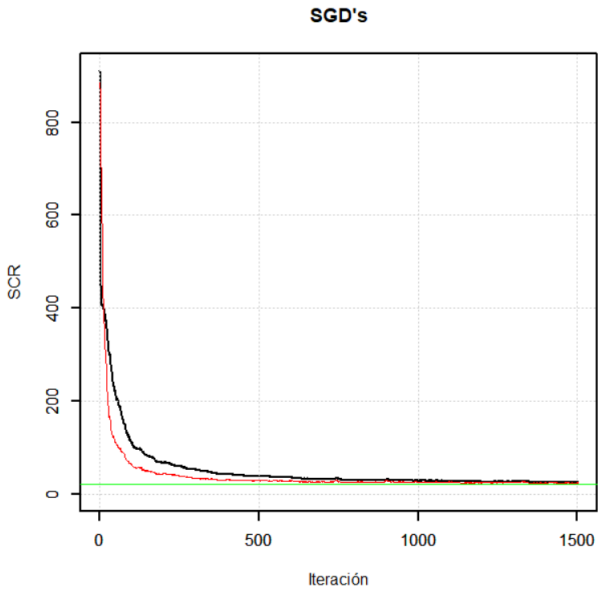

Figura 6: Método SGD con tamaño de paso constante $\gamma = 2 \cdot 10^{-1}$ en linea roja vs SGD con tamaño de paso variable en linea negra.

Referencias

Kingma, D. P. & Ba, J. (2017). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*. Consultado el 18 de septiembre de 2021, desde http://arxiv.org/abs/1412.6980