Método de Máxima Verosimilitud como problema de Optimización

Rodrigo Suárez Segovia - 191351
José Reyes Garza - 142207
Miguel López Cruz - 197967
José Luis Zárate Cortés - 183347
Maestría en Ciencia de Datos
Instituto Tecnológico Autónomo de México

Diciembre 2020

Índice

Resumen

En estadística, uno de los principales problemas es determinar el valor de un parámetro (o los valores de varios parámetros) alternativos. En este sentido, existen diversas metodologías que nos permiten calcular estimadores distintos de un mismo parámetro para una población, tales como el método de momentos y el método de máxima verosimilitud. Esencialmente, dichas metodologías, realizan un proceso de Optimización, por lo que naturalmente podemos adecuar este proceso para analizarlo a través de los métodos de Descenso en Gradiente y Newton-Raphson."

1. Introducción

El método de máxima verosimilitud es una herramienta estadística que nos permite estimar estadísticos o parámetros de una distribución a partir de los datos observados en una muestra. Este método implica construir una función objetivo, plantear un problema de maximización y obtener el punto óptimo que representa el valor de los estimadores. En este trabajo, se describirá a detalle el método de máxima verosimilitud y su relación con la estadística. Después, se presentan dos diferentes métodos de optimización que son usado para obtener los estimadores de máxima verosimilitud. Continúa con la implementación del Método de máxima verosimilitud para una distribución Normal y para una distribución Poisson y su modelo de regresión. En esta sección se plantea el problema de maximización y su solución analítica. Por último, se aplica este método utilizando conjuntos de datos y aplicando los métodos de optimización mencionados anteriormente.

2. Función de Verosimilitud

La función de verosimilitud (o, simplemente, verosimilitud) es una función de los parámetros de un modelo estadístico que permite realizar inferencias acerca de su valor a partir de un conjunto de observaciones. En otras palabras, dada una muestra observada

$$x_1, x_2, ..., x_n$$

y su función de densidad conjunta como sigue:

$$f(x) = (x_1, x_2, ..., x_n)$$

la función de verosimilitud cuantifica la probabilidad de que dichas observaciones en la muestra provengan efectivamente de una muestra teórica.

En la práctica el uso en la estimación de parámetros es para maximizar la probabilidad de obtener justamente la muestra observada; es decir, selecciona como estimaciones, valores de los parámetros que maximizan la función de densidad conjunta de la muestra observada.

La función de verosimilitud se define como sigue:
Sean $X_1, \ldots, X_n$ una muestra de una densidad $f(x;\theta)$ y sean $x_1, x_2, \ldots, x_n$ los valores observados.

La función de verosimilitud del parámetro de interés θ está definida por:

$$\mathcal{L}(\theta; x_1, \ldots, x_n) = \prod_{i=1}^{n} f(x_i; \theta)$$

Esta función nos dice qué tan creíble es el valor del parámetro θ dada la muestra observada. A veces también la denotamos por $\mathcal{L}n(\theta)$.

2.1. Método de Máxima Verosimilitud

Máxima verosimilitud es un proceso intuitivo, y consiste en aprender o estimar valores de parámetros desconocidos suponiendo para los datos su explicación más probable. Para esto, usando supuestos y modelos requeriremos calcular la probabilidad de un conjunto de observaciones. En otras palabras, la estimación de máxima verosimilitud maximiza la probabilidad de los parámetros de las funciones de densidad que dependen de la distribución de probabilidad y las observaciones de la muestra.

De manera formal, un estimador de máxima verosimilitud lo denotamos por $\hat{\theta}_{\mathsf{MLE}}$ y es un valor que satisface:

$$\hat{\theta}_{\mathsf{MLE}} = \arg\max_{\theta \in \Theta} \mathcal{L}(\theta; x_1, \ldots, x_n)$$

donde θ denota el espacio perimetral. Es decir, el espacio válido de búsqueda congruente con la definición del modelo.

Obsérvese que para construir la verosimilitud y en consecuencia buscar por estimadores de máxima verosimilitud necesitamos:

- Un modelo teórico de cómo es la población con parámetros.

- Información de cómo se extrajo la muestra.

y entonces podemos resolver nuestro problema de estimación convirtiéndolo en uno de optimización.

Ahora notemos que:
• Maximizar la verosimilitud es lo mismo que maximizar la log-verosimilitud, pues el logaritmo es una función creciente. Si x_max es el máximo de f, tenemos que $f(x_max) > f(x)$ para cualquier x, entonces tomando logaritmo,

$$log(f(x_max)) > log(f(x)),$$

para cualquier x. Pues el logaritmo respeta la desigualdad por ser creciente.
• Usualmente usamos la log-verosimilitud para encontrar el estimador de máxima verosimilitud.
• Hay razones teóricas y de interpretación por las que también es conveniente hacer esto.

La log-verosimilitud la denotamos usualmente por

$$\ell_n(\theta) = \log\left(\mathcal{L}_n(\theta)\right)$$

donde hemos suprimido la dependencia en la muestra por conveniencia.

Es decir, al igual que otros métodos, la estimación de máxima verosimilitud se basa en la iteración. Es decir, repetir una operación determinada tantas veces como requiera para encontrar el valor máximo o mínimo de una función, como un problema de optimización. Este proceso puede estar sujeto a restricciones en los valores finales de los parámetros. Por ejemplo, que el resultado sea superior o igual a cero o que la suma de dos parámetros tiene que ser inferior a uno[1]:

3. Métodos de Optimización

Dado que el método de máxima verosimilitud exige maximizar una función objetivo, es necesario entender los métodos numéricos que permiten encontrar este valor máximo. En esta sección, se definen los métodos de descenso y se exponen dos de ellos, el método de descenso en gradiente y método de Newton Raphson.

3.1. Métodos de Descenso

Los métodos de descenso son aquellos que producen una secuencia de minimización $x^{(k)}$, $k = 1, 2, \ldots$ donde:

$$(1)x^{(k+1)} = x^{(k)} + t^{(k)}\Delta x^{(k)}$$

y $t^{(k)} > 0$ (excepto cuando $x^{(k)}$ es óptimo). La entidad $\Delta x^{(k)}$ es un vector R^n llamado "**paso**.º "**dirección de búsqueda**", donde $k = 1, 2, \ldots$ es el índice de iteración. El escalar $t^{(k)} \geq 0$ se denomina el "**tamaño de paso**.en la iteración k. En algunas ocasiones es conveniente utilizar una notación simplificada sin superíndices $x := x + t\Delta x$ en lugar de (1). Los métodos de descenso implican lo siguiente:

$$f(x^{k+1}) < f(x^k),$$

excepto cuando x^k es óptimo.

Un método de descenso es aquel que genera la secuencia de minimización $x^{(0)}, x^{(1)}, \cdots \in \text{dom} f_o$ la cual cumple con la desigualdad: $f_o(x^{(k+1)}) < f_o(x^{(k)})$ excepto para $x^{(k)}$ óptimo y $f_o(x^{(k)}) \to p^*$.

La idea de los métodos de optimización es calcular direcciones Δx de búsqueda que sean de descenso, esto es, que al movernos de un punto a otro en tal dirección, el valor de f_o decrece.

Si el paso o dirección de búsqueda satisface: $\nabla f_o(x)^T \Delta x < 0$ se le nombra dirección de descenso, siendo $\nabla f_o(x)^T \Delta x$ la derivada direccional en de f_o en x en la dirección de Δx

[1]El modelo simétrico GARCH y sus diferentes extensiones se aplica la Estimación de Máxima Verosimilitud para encontrar el valor estimado de los parámetros que maximiza la probabilidad de los parámetros de las funciones de densidad.

3.2. Método de Descenso en Gradiente

Este método es un algoritmo de optimización iterativo de primer orden que busca encontrar el mínimo local de una función diferenciable.

Una selección natural para la búsqueda de dirección es el negativo del gradiente $\Delta x = -\nabla f(x)$, lo que da lugar al método de descenso en gradiente:

Algoritmo: Dada una función objetivo f_o y un punto inicial $x \in dom f_o$, conceptualmente el método consiste en realizar el siguiente proceso de forma iterativa:

Repetir:

1. $\Delta x := -\nabla f(x)$.
2. Búsqueda de línea. Seleccionar un tamaño de paso t . Existen algunas metodologías conocidas, "vía exactaz "backtracking", que ayudan a seleccionar el tamaño de paso adecuado, lo que facilita o acelera la convergencia garantizando no quedar en atrapado dentro de un ciclo. Dado el objetivo de nuestro problema, utilizaremos un tamaño de paso definido en nuestros algoritmos.
3. Actualizar.

Hasta satisfacer el criterio de paro.

El criterio de paro tiene la forma $\|\nabla f(x)\|_2. \leq \eta$, donde η es un valor positivo y "pequeño". Esta condición se evalúa posterior a la primer iteración.

3.3. Método de Newton Raphson

En análisis numérico, el método de Newton (conocido también como el método de Newton-Raphson) es un algoritmo para encontrar aproximaciones de los ceros o raíces de una función real. También puede ser usado para encontrar el máximo o mínimo de una función, encontrando los ceros de su primera derivada.

El método de Newton-Raphson, permite hallar una raíz de una ecuación no-lineal siempre y cuando se parta de una buena estimación inicial de la misma. El esquema iterativo de Newton puede derivarse del desarrollo de Taylor de la función alrededor de la estimación inicial.

$$f(x) = 0 = f(x_o) + f'(x_o)(x - x_o) + O(h^2)$$

Ahora bien, la recta tangente a la función, que pasa por el punto $[x0, f(x0)]$, se encuentra definida por la siguiente expresión:

$$g(x) = f(x_0) + f'(x_0) * (x - x_0)$$

Si denominamos x_1 a la intersección de $g(x)$ con el eje x (esto es, la raíz de $g(x)$), resolviendo dicha ecuación obtenemos, la siguiente expresión:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

y generalizando este esquema de aproximaciones sucesivas a la raíz, obtenemos:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n}$$

Para que el método de Newton-Raphson converja deben cumplirse ciertas condiciones de convergencia. Aún partiendo de un punto cercano a la raíz buscada, en un caso el método converge y en otro caso no.

4. Implementación del Método de Máxima Verosimilitud

En esta sección, se plantea el método de máxima verosimilitud para obtener los estimadores de una distribución Normal, de una distribución Poisson y de un modelo de regressión Poisson. Se realiza un plenteamiento teórico de estimación por máxima verosimilitud desde un punto de vista de un problema de optimización.

4.1. Distribución Normal

La distribución de probabilidad de una Normal tiene la siguiente forma:

$$f_x(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Se puede estimar los parámetros μ y σ desconocidos utilizando el método de máxima verosimilitud dado un conjunto de datos independientes e idénticamente distribuidos:

$$X_n = x_1, x_2, ..., x_n \sim N(\mu,\sigma)$$

Sea $\ell(\mu,\sigma|X_n)$ la función de verosimilitud de la distribución normal. Se tiene que:

$$\ell(\mu,\sigma|X_n) = \prod_{i=1}^{n} f_x(x_i|\mu,\sigma)$$

$$\ell(\mu,\sigma|X_n) = \prod_{i=1}^{n} (2\pi\sigma^2)^{\frac{-1}{2}} e^{\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2}$$

$$\ell(\mu,\sigma|X_n) = (2\pi\sigma^2)^{\frac{-n}{2}} exp\big(\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\big)$$

Calculando el logaritmo de la función de verosimilitud, se obtiene:

$$ln(\ell(\mu,\sigma|X_n)) = -\frac{n}{2}ln(2\pi) - \frac{n}{2}ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2$$

La transformación logarítmica es una transformación monótona, por lo que se mantiene la concavidad de la función verosimilitud.

4.1.1. Problema de Maximización

Buscamos encontrar los parámetros μ y σ tal que maximicen la función de log-verosimilitud. El problema de maximizacion esta dado por[2]:

$$max_{\mu,\sigma} ln(\ell(\mu, \sigma|X_n)) = -\frac{n}{2}ln(2\pi) - \frac{n}{2}ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2$$

Para determinar que los los puntos μ^* y σ^* son puntos óptimos, deben cumplir con lo siguiente:

Condición Necesaria de Primer Orden:

$$\nabla(\ell(\mu^*, \sigma^*|X_n)) = 0$$

Condición Suficiente de Segundo Orden [3]:

$$\nabla^2(\ell(\mu^*, \sigma^*|X_n)) \in -S_-$$

Para esta función de log-verosimilitud, el gradiente está dado por:

$$\nabla(\ell(\mu, \sigma|X_n)) = \begin{bmatrix} \frac{\partial \ell(\mu,\sigma|X_n)}{\partial \mu} \\ \\ \frac{\partial \ell(\mu,\sigma|X_n)}{\partial \sigma} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i - \mu) \\ \\ -\frac{n}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{n}(x_i - \mu)^2 \end{bmatrix}$$

Para resolver el problema de forma analítica,se tiene un sistema de ecuaciones no lineales con dos incógnitas que están dadas por la condición necesaria de primer orden.

1. $\frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i - \mu) = 0$
2. $-\frac{n}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{n}(x_i - \mu)^2 = 0$

Resolviendo dicho sistema, tenemos que los estimadores de máxima verosimilitud son:

$$\mu_{mle} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\sigma_{mle} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_{mle})^2}$$

[2]Un problema de maximización de $f(x)$ es equivalente a un problema de minimización si la función a minimizar es igual a $-f(x)$

[3]En caso de un problema de minimización, la condición suficiente de segundo orden es

$$\nabla^2(\ell(\mu^*, \sigma^*|X_n)) \in S_+$$

Para determinar que efectivamente estos valores son un valor máximo de la función de log-verosimilitud, es necesario satisfacer la condición suficiente de segundo orden. Para esto, se calcula la matriz Hessiana:

$$\nabla^2(\ell(\mu,\sigma|X_n)) = \begin{bmatrix} \frac{\partial^2\ell(\mu,\sigma|X_n)}{\partial^2\mu} & \frac{\partial^2\ell(\mu,\sigma|X_n)}{\partial\mu\partial\sigma} \\ \frac{\partial^2\ell(\mu,\sigma|X_n)}{\partial\sigma\partial\mu} & \frac{\partial\ell^2(\mu,\sigma|X_n)}{\partial^2\sigma} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\sigma^2}n & -\frac{2}{\sigma^3}\sum_{i=1}^n(x_i-\mu) \\ -\frac{2}{\sigma^3}\sum_{i=1}^n(x_i-\mu) & \frac{n}{\sigma^2}-\frac{4}{\sigma^4}\sum_{i=1}^n(x_i-\mu)^2 \end{bmatrix}$$

En este caso, para que sea un máximo local, la matriz debe ser semi definida negativa. Para garantizar que el punto critico encontrado es un máximo global, es necesario analizar las propiedades de la función log-verosimilitud.

Se define a una función $f(x)$ como cuasicóncava si para todo $x \in S$, para todo $x' \in S$ con S un conjunto convexo, $x \neq x'$ y $\lambda \in [0,1]$ se tiene que si $f(x) \geq f(x')$ entonces $f(\lambda x + (1-\lambda)x') \geq f(x')$ [4]. Si la desigualdad se cumple de forma estricta, se dice que es estrictamente cuasicóncava. Si una función es estrictamente cuasicóncava, entonces se garantiza que el máximo local es igual al máximo global[5]. Esto tambien implica que el contorno superior de una función cuasicóncava es un conjunto convexo.[6]. En la siguiente sección se proveerán elementos gráficos para demostrar la cuasiconcavidad de la función log-verosimilitud.

4.2. Distribución Poisson

Decimos que X es una variable aleatoria de *Poisson* con parámetro λ, denotado por $X \sim Poisson(\lambda)$, si su rango es $R_X = \{0,1,2,...\}$, y su función de masa de probabilidad está dada por:

$$P_X(k) = \begin{cases} \frac{e^{-\lambda}\lambda^k}{k!} & \text{si } k \in R_X \\ 0 & \text{si } k \notin R_X \end{cases}$$

Estimaremos el parámetro λ utilizando el método de máxima verosimilitud, similar al empleado para el caso de una distribución normal. Dado un conjunto de valores observados $X_1 = x_1$, ..., $X_n = x_n$, asumiendo que las variables aleatorias X_1, ..., X_n son independientes e idénticamente distribuidas, la función de verosimilitud viene dada por:

$$\mathcal{L}(x_1,x_2,...,x_n;\theta) = \prod_{j=1}^n f_{X_j}(x_j;\theta) = \prod_{j=1}^n \frac{e^{-\theta}\theta^{x_j}}{x_j!} = \frac{e^{-n\theta}\theta^{x_1+\cdots+x_n}}{\prod_{j=1}^n(x_j!)} = \frac{e^{-n\theta}\theta^S}{c}$$

donde $S = \sum_{j=1}^n x_j$ y $c = \prod_{j=1}^n(x_j!)$.

[4] https://mjo.osborne.economics.utoronto.ca/index.php/tutorial/index/1/qcc/t

[5] http://www.pitt.edu/ luca/ECON2001

[6] Otro resltado importante, es que si una funcion $f(x)$ es cuasicóncava, entonces $-f(x)$ es cuasiconvexa. Si una funcion es cuasiconvexa podemos asegurar que habra un mínimo.

De ahí que,

$$\ln \mathcal{L}(x_1, x_2, ..., x_n; \theta) = -n\theta + s\ln(\theta) - \ln(c)$$

luego,

$$\ln \mathcal{L}(x_1, x_2, ..., x_n; \theta)^{'} = -n + \frac{S}{\theta}$$

Igualando la última derivada a cero y despejando θ nos queda:

$$\hat{\theta}_{ML} = \frac{1}{n}\sum_{j=1}^{n} x_j$$

Ésto sugiere que el estimador de máxima verosimilitud (MLE) puede ser escrito de la siguiente manera:

$$\hat{\theta}_{ML} = \frac{1}{n}\sum_{j=1}^{n} X_j$$

.

Modelo de Regresión

Si $\mathbf{x} \in R^n$ es un vector de variables independientes, entonces el modelo toma la forma

$$\log(\mathrm{E}(Y \mid \mathbf{x})) = \alpha + \beta'\mathbf{x}$$

donde $\alpha \in R$ y $\beta \in R^n$.

También, donde x es ahora un vector $(n+1)$ -dimensional que consta de n variables independientes concatenadas al número uno.

Por lo tanto, cuando se le da un modelo de regresión de Poisson con parámetro θ y un vector de entrada x, la media predicha de la distribución de Poisson asociada viene dada por

$$\mathrm{E}(Y \mid \mathbf{x}) = e^{\theta^{'} x}$$

Si Y_i son observaciones independientes con los valores correspondientes x_i de las variables predictoras, entonces θ puede estimarse por máxima verosimilitud. Las estimaciones de máxima verosimilitud carecen de una expresión de forma cerrada y deben calcularse mediante métodos numéricos. La superficie de probabilidad para la regresión de Poisson de máxima verosimilitud es siempre cóncava, lo que hace que Newton-Raphson u otros métodos basados en gradientes, técnicas de estimación apropiadas.

5. Aplicación

5.1. Estimación de Parámetros para una Distribución Normal

Para la estimación de parámetros por el método de máxima verosimilitud, se desarrollo una serie de algoritmos que implementan métodos numéricos necesarios resolver el problema. Para ilustrar el funcionamiento de estos métodos numéricos, se utiliza una base de datos de rendimiento de trigo por unidad de tierra. Esta base de datos es de un estudio históricamente relevante por Mercer and Hall en Rothamsted Experimental Station en Gran Bretaña en el año de 1910[7]. Esta estación experimental se fundó en 1834 con el objetivo de experimentar con diferentes cultivos y analizar su rendimiento contrastando el uso de distintos fertilizantes.

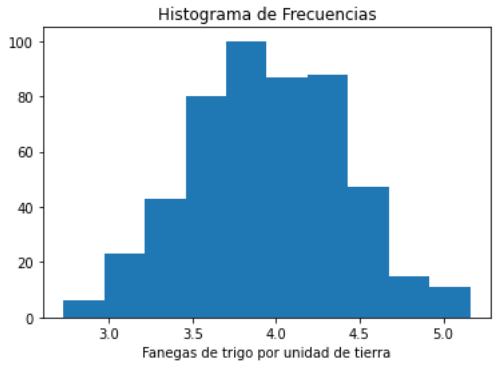

Figura 1: De acuerdo con el histograma de frecuencias, el rendimiento de tipo podría provenir de una distribución normal.

En primer lugar, es necesario programar la función de log-verosimilitud de una distribución normal que funcione para cualquier conjunto de datos. Para resolver el problema de maximización de forma numérica, es necesario cambiar el planteamiento del problema como uno de minimización. Por lo tanto, la función de verosimilitud programada regresa el valor negativo de la función evaluada en los parámetros.

```
def normal_loglikelihood(params,*args):
    mu = params[0]
    sigma = params[1]
    x = data
    n=len(data)
```

[7]Mercer, W.B., and A.D. Hall. 1911. The experimental error of field trials. Journal of Agricultural Science (Cambridge) 4: 107-132

```
    loglikelihood=-(n/2)*np.log(2*np.pi)-
                  (n/2)*np.log(sigma**2)-(1/(2*sigma**2))*np.sum((x-mu)**2)

    return -1*loglikelihood
```

Dado que esta es una función $f : R \times R_+ \to R$ podemos explorar de forma visual si la función es cuasi convexa para determinar si existe un mínimo global. La **figura 2** sugiere que la función es cuasiconvexa, por lo tanto, de encontrar un mínimo local podemos deducir que también sera un mínimo global.

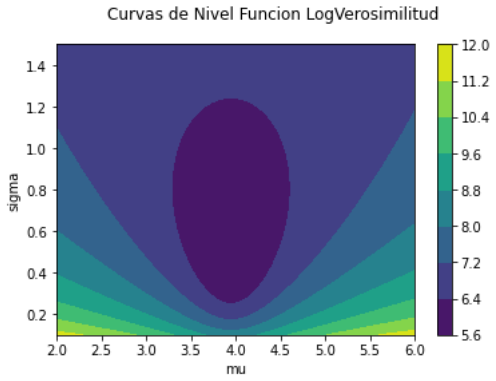

Figura 2: El contorno inferior de la función verosimilitud es convexo, por lo tanto es cuasiconvexa. Nota: Se aplico una transformación logarítmica para reducir el nivel de la función, sin embargo el mapeo de curvas de nivel no se ve afectado.

Para obtener los estimadores de máxima verosimilitud μ_{mle} y σ_{mle}, se implementan dos algoritmos: descenso en gradiente y Newton Raphson. En ambos casos, es necesario programar el gradiente de la log-verosimilitud.

```
def gradient_normal_loglike(params,*args):
    mu = params[0]
    sigma = params[1]
    x=data
    n=len(data)
    dmu= (1/(sigma**2))*np.sum(x-mu)
    dsigma=-(n/2)*((2*sigma)/sigma**2)+(1/sigma**3)*np.sum((x-mu)**2)
    return -1*np.array([dmu,dsigma])
```

Una vez calculado el gradiente de la funcion log-verosimilitud, se utiliza el algoritmo de descenso en gradiente presentado a continuación.

```
x=np.array([1,1])
x_old=x+10
```

```
i=0
points=list()
while ((np.linalg.norm(x-x_old)>=.001)):
    if i==1000:
        break
    points.append(x)
    x_old=x
    x=x-.0001*gradient_normal_loglike(x)
    i+=1
```

Para asegurar la convergencia, es necesario tener un tamaño de paso lo suficientemente pequeño. En este caso se probo que un tamaño igual a 0,0001 era suficiente para converger en menos de mil iteraciones.

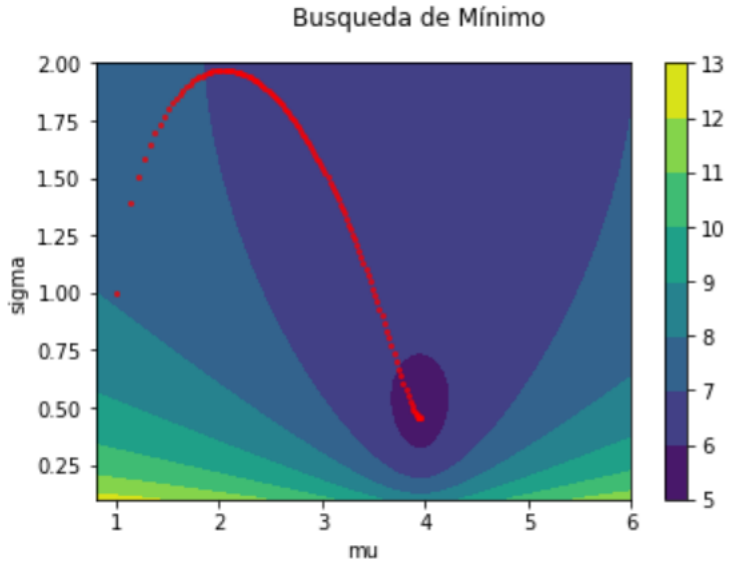

Figura 3: Trayectoria de descenso para la búsqueda del mínimo con el algoritmo de descenso en gradiente.

Los valores resultados del algoritmo de descenso están dados en la siguiente tabla:

Además se puede contrastar con los valores obtenidos resolviendo el problema con la solución analítica:

	μ	σ
Solución por DG	3.94832449	0.45782199
Solución Analítica	3.9486399	0.45782108
Error Relativo	-7.99e-05	1.97e-06

5.2. Estimación de Parámetros para una Distribución Poisson

En este caso, vamos a estudiar el comportamiento de la implementación desarrollada del método de optimización "Newton-Raphson"para obtener el parámetro de una distribución Poisson. Para ello, se seleccionó el caso analizado en el artículo de Daniel Treisman (2016), que concluye que Rusia tiene un número mayor de multimillonarios de lo que predicen factores económicos como el tamaño del mercado y la tasa impositiva [8]. Para ello, el autor se interesó en estimar el número de los multimillonarios en diferentes países y utilizó la información contenida en los datos del artículo "Forbes' annual rankings of billionaires and their estimated net worth".

En este caso, la variable a analizar, se valora en números enteros por lo que debemos consideramos distribuciones que toman valores solo en los enteros no negativos, por lo que se considera que el comportamiento de esta variable sigue una Distribución de Poisson. El autor construye un modelo basado en cuatro variables explicativas con las que busca predecir el número de multimillonarios en un determinado país, por lo que condiciona su variable de regresión a tales variables en un modelo conocido como un Regresión de Poisson. Para estimar su modelo utilizando MLE, lo que desea es maximizar la verosimilitud de que su estimador $\hat{\beta}$ es el verdadero parámetro β, el vector de las 4 variables seleccionadas para describir su modelo. Dentro de las soluciones planteadas, considera el realizar su análisis considerando métodos numéricos o paqueterías como "statmodels".

Para nuestro caso, realizamos una adaptación a la solución planteada por el autor, en la cual buscamos demostrar el uso del método de Newton-Raphson en la determinación del vector que maximiza $\hat{\beta}$.

Codificamos la función de log-verosimilitud asociada a una distribución Poisson. Crearemos una clase llamada Poisson para que podamos volver a calcular fácilmente los valores de la probabilidad de cada uno de los registros, el gradiente y la hessiana para cada iteración.

Finalmente, comparamos el vector $\hat{\beta}$ (excepto para el intercepto) obtenido con el método de optimización hemos desarrollado Newton-Raphson y comparando con los valores obtenidos por el autor con la paquete "statmodels"

[8] https://python.quantecon.org/mle.html

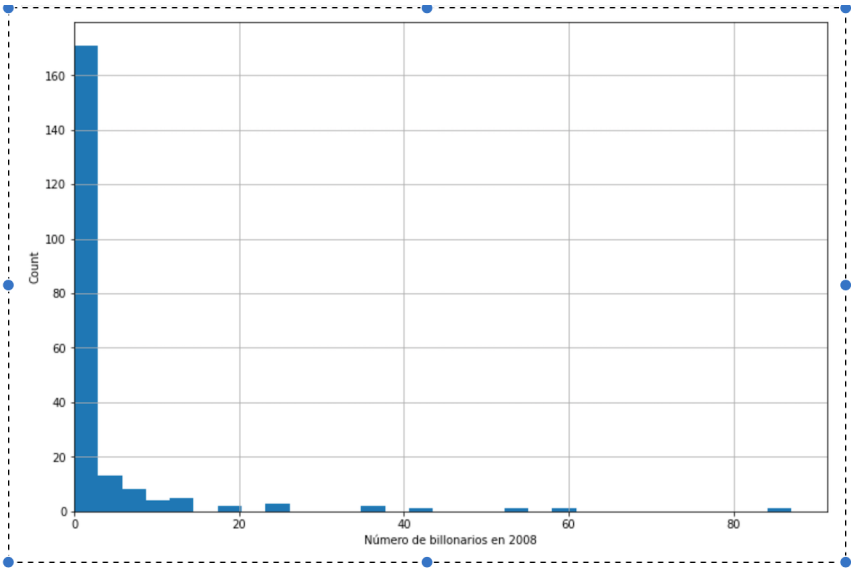

Figura 4: De acuerdo con el histograma de frecuencias, el número de billionarios debe provenir de una distribución Poisson.

```
Iteration_k  Log-likelihood  θ
-------------------------------------------------------------------------------
0            -inf            ['-0.932', '0.102', '0.103', '0.0994']
1            -inf            ['-2.02', '0.108', '0.111', '0.0978']
2            -inf            ['-3.26', '0.124', '0.132', '0.0937']
3            -inf            ['-4.93', '0.166', '0.188', '0.0837']
4            -inf            ['-7.81', '0.267', '0.321', '0.0632']
5            -inf            ['-13.3', '0.476', '0.582', '0.0354']
6            -inf            ['-21.0', '0.759', '0.894', '0.016']
7            -inf            ['-26.4', '0.969', '1.09', '0.00917']
8            -inf            ['-28.5', '1.06', '1.15', '0.00711']
9            -inf            ['-28.7', '1.07', '1.16', '0.00683']
10           -inf            ['-28.7', '1.07', '1.16', '0.00683']
Number of iterations: 11
β_hat = [-2.87359323e+01  1.07004016e+00  1.15985257e+00  6.82561588e-03]
```

Figura 5: Se obtienen los estimadores para una regresión Poisson.

6. Conclusiones

En conclusión, el método de máxima verosimilitud es una herramienta que nos permite obtener estimadores puntuales de una distribución dada una muestra de datos. Dado que este método implica un proceso de maximización, su implementación depende de métodos numéricos que permitan encontrar los valores máximos de la función. En este caso, se realizó la implementación en una distribución Normal y un modelo de regresión Poisson. En ambos casos, los estimadores obtenidos utilizando el método de descenso en gradiente y de Newton Raphson son muy similares a los resultados obtenidos de forma analítica.

```
Optimization terminated successfully.
         Current function value: 2.226090
         Iterations 9
                          Poisson Regression Results
==============================================================================
Dep. Variable:                numbil0   No. Observations:                  197
Model:                        Poisson   Df Residuals:                      193
Method:                           MLE   Df Model:                            3
Date:                Sun, 06 Dec 2020   Pseudo R-squ.:                  0.8574
Time:                        01:26:41   Log-Likelihood:                -438.54
converged:                       True   LL-Null:                       -3074.7
Covariance Type:                 HC0   LLR p-value:                     0.000
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const        -29.0495      2.578    -11.268      0.000     -34.103     -23.997
lngdppc        1.0839      0.138      7.834      0.000       0.813       1.355
lnpop          1.1714      0.097     12.024      0.000       0.980       1.362
gattwto08      0.0060      0.007      0.868      0.386      -0.008       0.019
==============================================================================
```

Figura 6: Se vuelve a estimar nuestro modelo simple con modelos de la paquetería Statmodels para confirmar que obtenemos los mismos coeficientes y valor logarítmico de probabilidad.

Entre las conclusiones generales de este trabajo podemos resumir las siguientes:

- La implementación y estimación de parámetros, se puede generalizar para cualquier distribución estadística.
- Los paquetes estadísticos más utilizados como: statmodels, linalg, etc. utilizan métodos de descenso en la optimización debido a la naturaleza algorítmica.
- Adicional a éste esfuerzo, se puede comparar el costo computacional sobre los algoritmos de optimización como descenso en gradiente y newton-raphson.
- En particular, para el método de newton-raphson, existe una complejidad para la generalización del método de optimización para dimensiones superiores.
- Para poder acelerar la convergencia de los métodos se pueden adicionar metodologías como "backtrackingƨ "vía exacta.[en] las iteraciones.