Cómputo científico para probabilidad y estadística. Tarea 7. MCMC: Metropolis-Hastings II

Juan Esaul González Rangel

Octubre 2023

Con el algoritmo Metropolis-Hastings (MH), simular lo siguiente:

1. Sean $x_i \sim Ga(\alpha, \beta); i = 1, 2, ..., n$. Simular datos x_i con $\alpha = 3$ y $\beta = 100$ considerando los casos $n = 4$ y 30.
   Con $\alpha \sim U(1, 4), \beta \sim exp(1)$ distribuciones a priori, se tiene la posterior

$$f(\alpha, \beta|\bar{x}) \propto \frac{\beta^{n\alpha}}{\Gamma(\alpha)^n} r_1^{\alpha-1} e^{-\beta(r_2+1)} \mathbb{1}_{1\leq\alpha\leq 4} \mathbb{1}_{\beta>0},$$

   con $r_2 = \sum_{i=1}^n x_i$ y $r_1 = \prod_{i=1}^n x_i$.
   En ambos casos, grafica los contornos para visualizar dónde está concentrada la posterior.
   Utilizar la propuesta

$$q\left(\begin{pmatrix}\alpha_p \\ \beta_p\end{pmatrix} \middle| \begin{pmatrix}\alpha \\ \beta\end{pmatrix}\right) = \begin{pmatrix}\alpha \\ \beta\end{pmatrix} + \begin{pmatrix}\varepsilon_1 \\ \varepsilon_2\end{pmatrix},$$

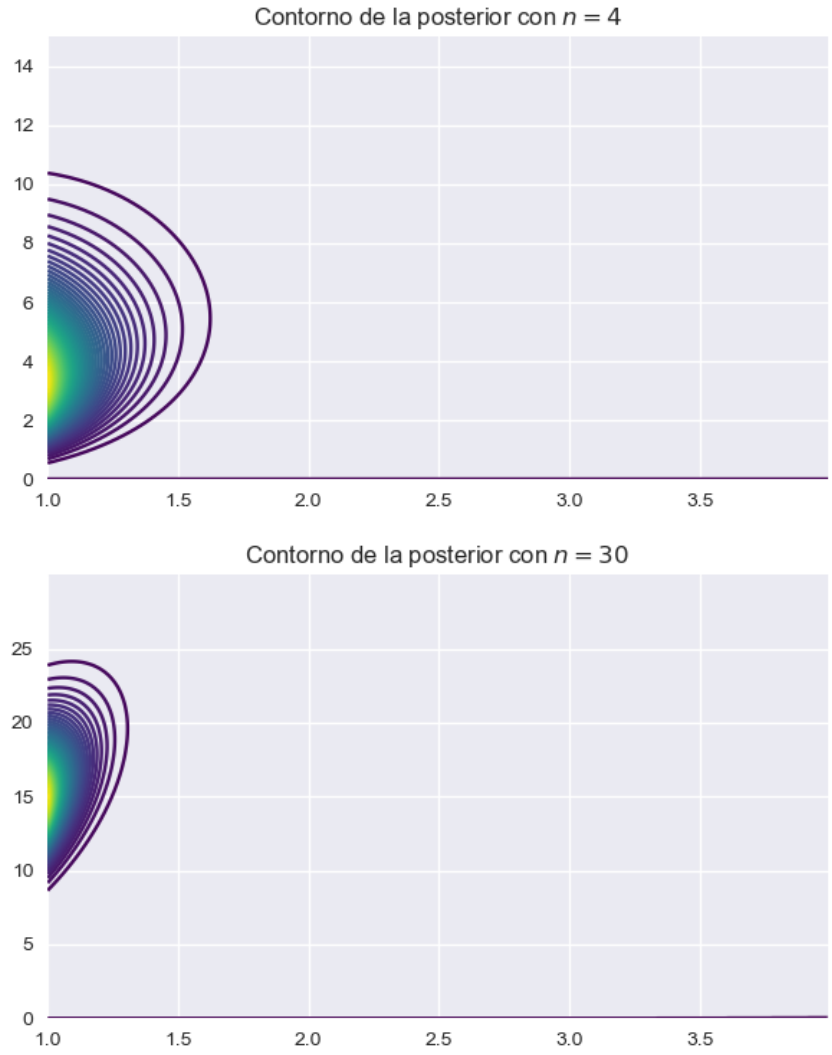
   donde

$$\begin{pmatrix}\varepsilon_1 \\ \varepsilon_2\end{pmatrix} \sim \mathcal{N}_2\left(\begin{pmatrix}0 \\ 0\end{pmatrix}, \begin{pmatrix}\sigma_1^2 & 0 \\ 0 & \sigma_2^2\end{pmatrix}\right).$$

   Solución. La simulación de los datos Gamma(α, β) se realizó mediante Numpy, y se obtuvieron los siguientes valores,

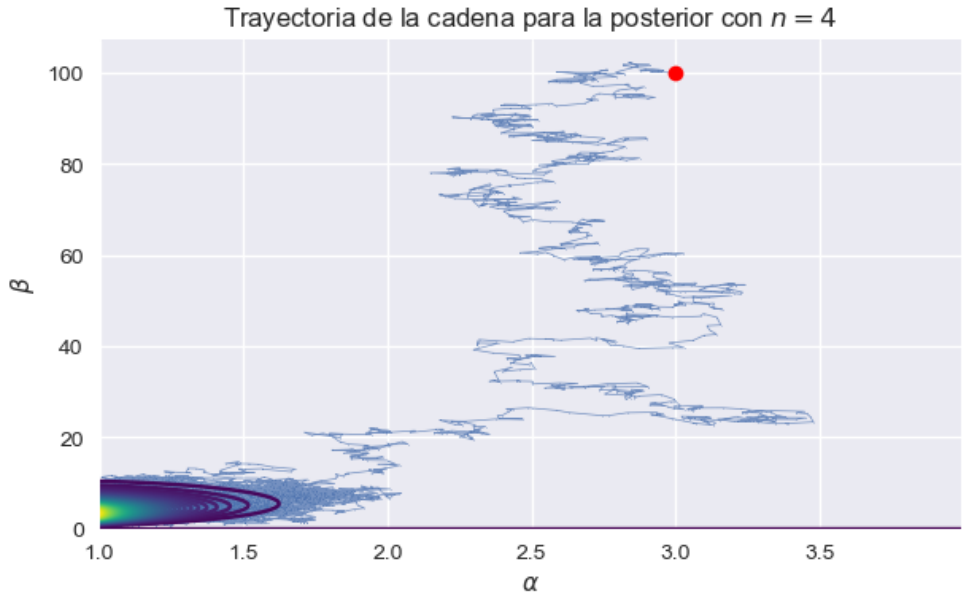
```
data4 = [0.02458333 0.02352249 0.0602106  0.08239561]

data30 = [0.00271467 0.04123339 0.06655271 0.00750071 0.01409199 0.02545264
0.01677018 0.02412853 0.02313011 0.02021658 0.07463565 0.01992959
0.03853146 0.04940094 0.02968845 0.00975932 0.05164652 0.04375947
0.03710245 0.01255773 0.03329843 0.02368755 0.04556456 0.01107533
0.03635922 0.02969227 0.07078983 0.02011851 0.0694928  0.05073943]
```

   A continuación, graficamos los contornos



Notamos que la densidad de la posterior parece estar concentrada en $0 \leq \beta \leq 10$ y $0 \leq \alpha \leq 1.75$ para el caso $n = 4$, y en $7.5 \leq \beta \leq 25$ y $0 \leq \alpha \leq 1.5$ para $n = 30$.

Para muestrear de la posterior se implementó el algoritmo de Metropolis-Hastings usando la densidad de transición propuesta. Como distribución inicial se usó la constante con $\alpha = 3, \beta = 100$ con el objetivo de poder observar la evolución de la cadena desde un punto distante. La trayectoria de la cadena se muestra en la siguiente Figura.



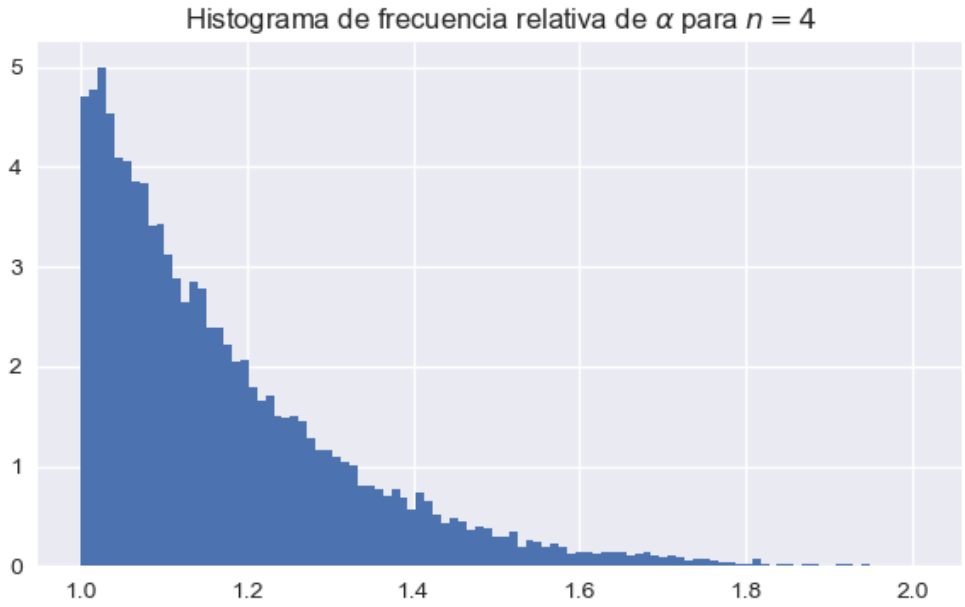
El punto que se muestra en color rojo es el punto inicial de la cadena.

Podemos notar varias cosas, la primera es que en etapas tempranas de la cadena el movimiento es principalmente en sentido vertical, y esto es porque proporcionalmente, la componente α de la cadena se encuentra más cerca de la densidad que la componente β, lo que cause que se acepten las transiciones que hacen a β más próximo, aunque no necesariamente lo hagan para α. Eventualmente, la cadena parece moverse con la misma intensidad en las direcciones α y β.

Para poder llegar a un comportamiento óptimo de la cadena, se ajustaron las varianzas σ_1^2 y σ_2^2, de manera que la cadena avanzara lo suficientemente rápido para llegar a la densidad en un tiempo corto, pero lo suficientemente lento, como para permanecer en el dominio de concentración de la densidad una vez que se llegó ahí. Las desviaciones estándar que se usaron después de varios intentos son 0.05 y 0.5.

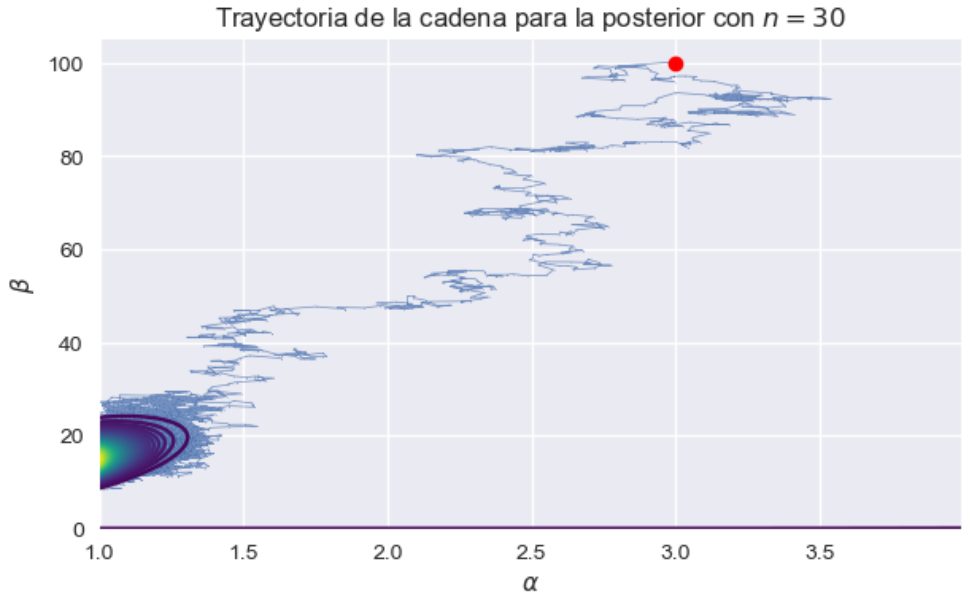
Evidentemente, en esta cadena hay puntos que no corresponden a muestrear desde la propuesta, por lo que es necesario encontrar el *burn-in* y eliminar los primeros datos para contar con una muestra más fiel. Podemos estimar el momento en que la cadena se estabiliza al observar la siguiente gráfica de $\log(f(\alpha_n, \beta_n))$, donde f es la posterior.

Existe un punto en la evolución del proceso en el que el logaritmo de la densidad se estabiliza y podemos considerar que este es el momento a partir del cuál estamos muestreando desde nuestra distribución objetivo. Entonces, para obtener una muestra de la posterior con la cadena, nos quedamos únicamente con las observaciones después de la 2,500. El histograma de dichas observaciones para α es el siguiente,



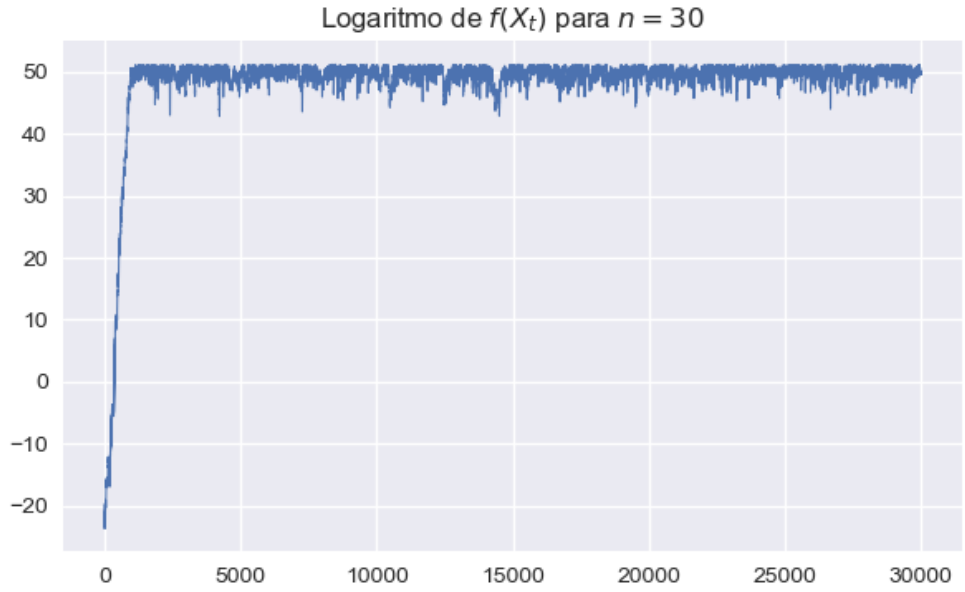
El histograma correspondiente para β es

Para el caso $n = 30$ se tienen resultados simiares, la trayectoria de la cadena es la siguiente



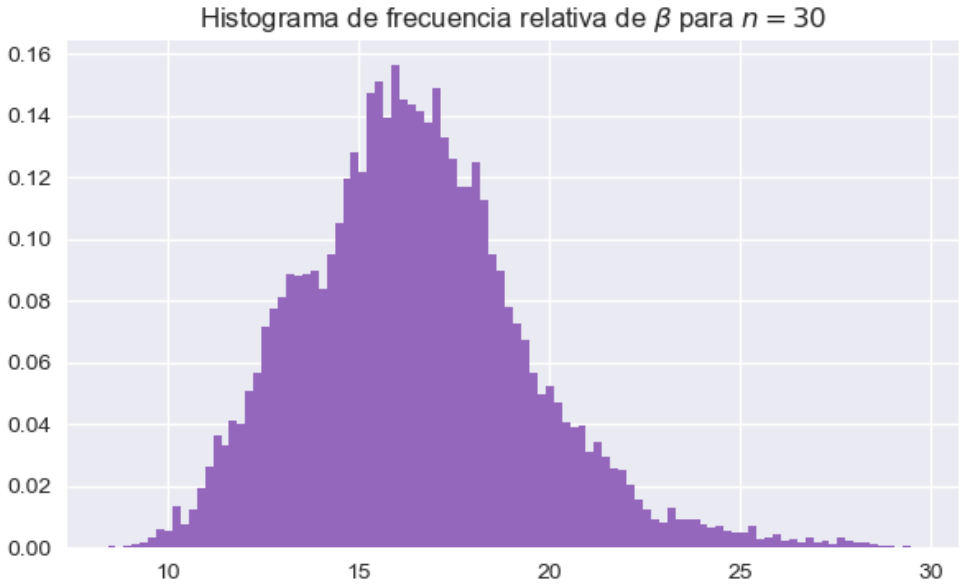
De igual manera, el punto rojo es el punto inicial de la cadena. Notemos que en este caso la cadena se mueve en dirección a la densidad más rápidamente que en el anterior, y esto se puede deber a que ahora la densidad está más cerca del punto inicial, lo que causa que se rechace la transición con menos frecuencia.

La gráfica de la log-densidad evaluada en la trayectoria es la siguiente,



En este caso, la log-densidad parece estabilizarse antes, por lo que es posible muestrear desde el punto 1500 en adelante. Al quedarnos únicamente con los valores pertinentes encontramos el siguiente histograma para α

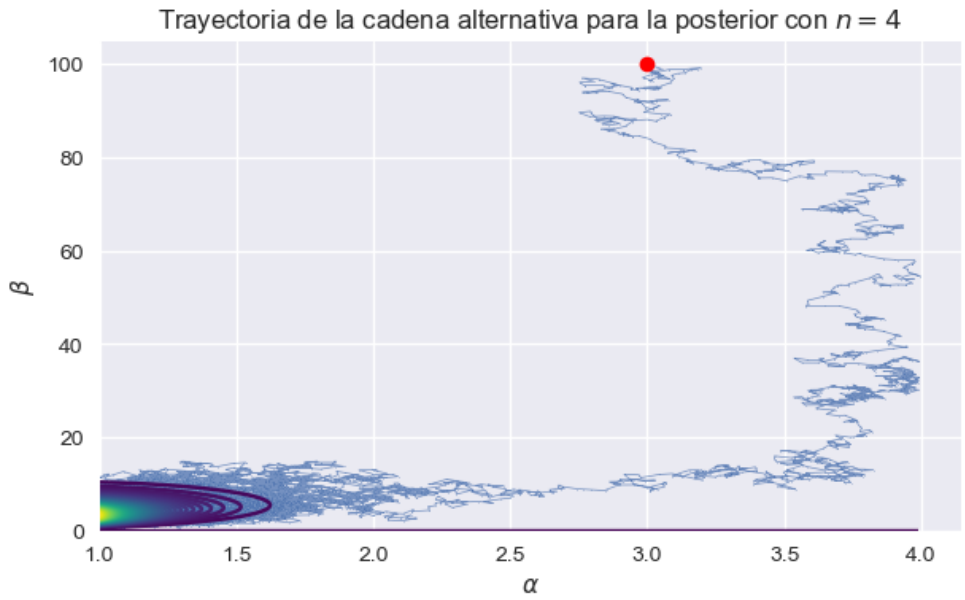
El histograma correspondiente para β es



En ambos casos, los resultados son similares a los que se obtienen para $n = 4$.

De un total de 30,000 pasos de la cadena que simulamos, se utilizaron alrededor de 2,500 o 1,500 para comenzar a muestrear de la distribución objetivo. Teniendo en cuenta que el costo computacional de 2,500 pasos es relativamente bajo, tenemos que la cadena es eficiente para la simulación de la posterior, pero debemos tener en cuenta que esto se debe a que las varianzas fueron ajustadas para que la convergencia pueda ser los mejor posible. Así mismo, es posible ajustar aún más las varianzas o cambiar la distribución inicial para obtener una convergencia aún mejor a la distribución objetivo.

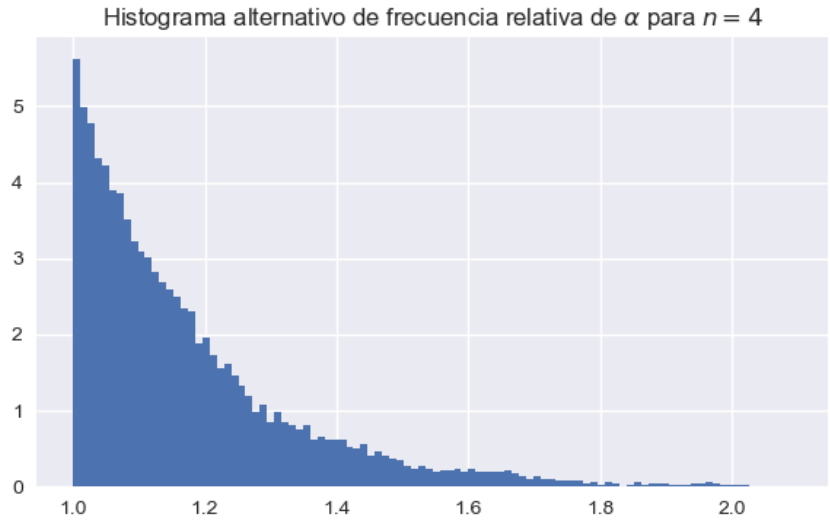
Una alternativa a la distribución normal como transición es la distribución semicircular de Wigner. Al implementar el algoritmo de Metropolis-Hastings para el mismo problema con una semicircular obtenemos la siguiente trayectoria para $n = 4$,

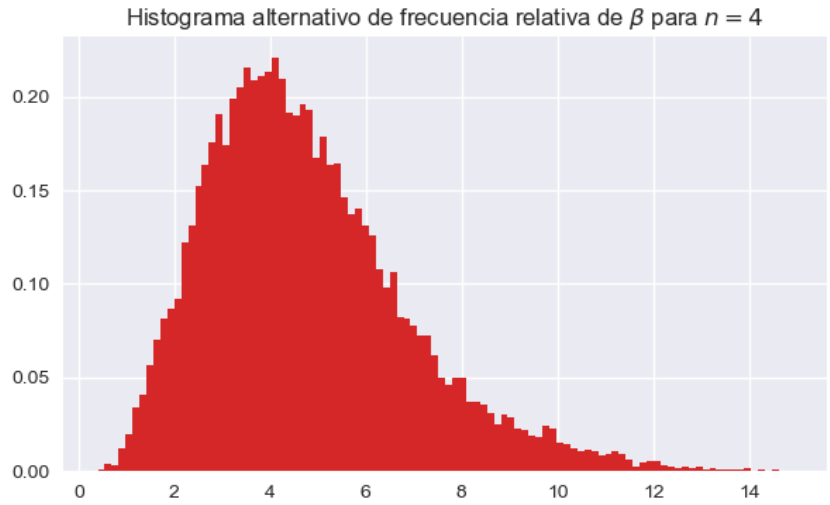


El comportamiento que observamos es similar al de la cadena con transición normal. Podemos encontrar su tasa de *burn-in* usando la gráfica de log-densidad,

Al igual que en el caso con densidad de transición normal, podemos considerar que es seguro tomar la muestra a partir del punto 2,500.

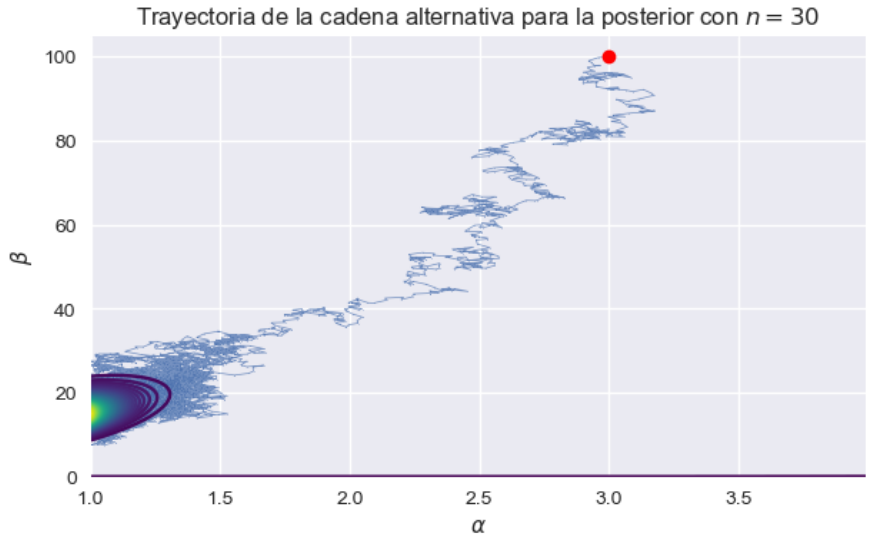
Los histogramas que obtenemos para α y β al eliminar los puntos de *burn-in* son los siguientes,





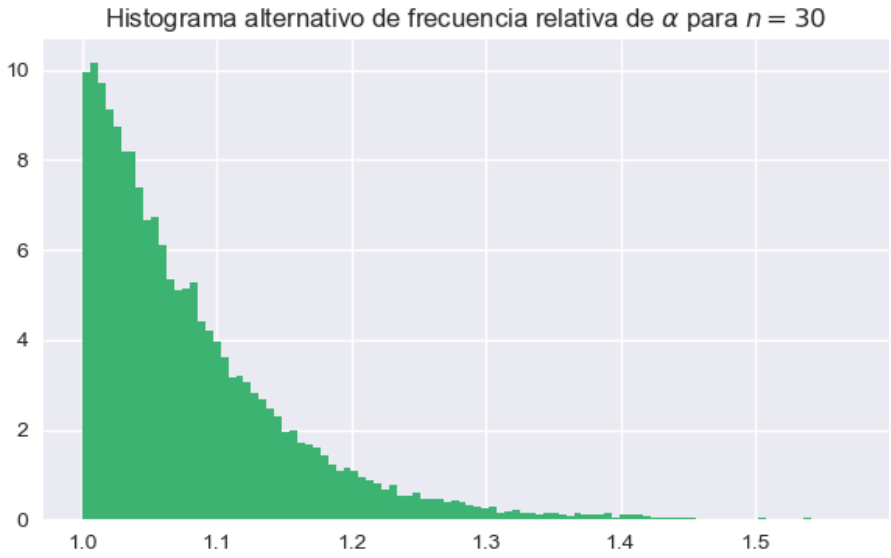
Los resultados son similares al algoritmo con transición normal.

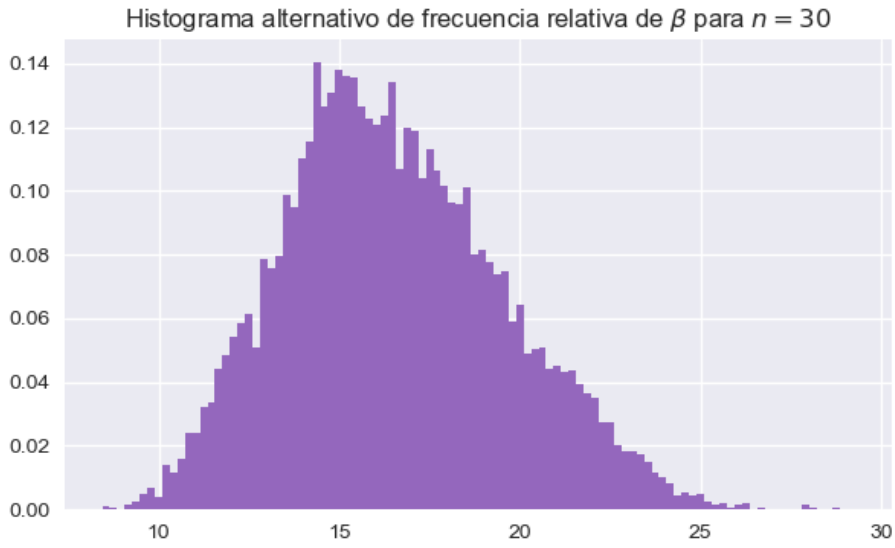
Para el caso $n = 30$ tenemos la siguiente trayectoria,



Al igual que en el caso de kernel normal, la cadena parece converger más rápido a la densidad que cuando $n = 4$. En la siguiente figura se observa la log-densidad evaluada en la historia de la cadena,

Notamos que también en este caso es posible considerar el muestreo a partir de $1,500$ iteraciones. Los histogramas para α y β son los siguientes,





Nuevamente, los resultados son similares a los análogos normales, por lo que concluimos que la distribución semicircular es una propuesta viable para este problema en particular.

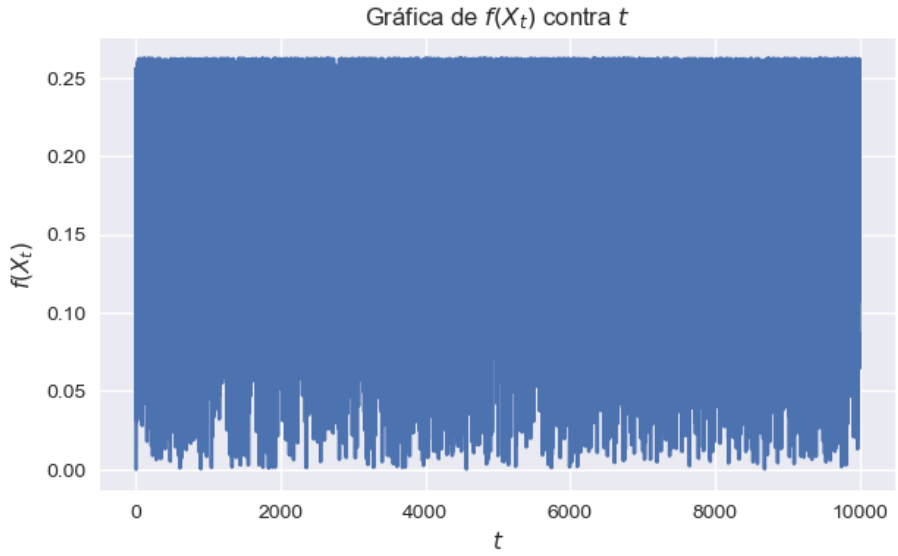
□

2. Simular de la distribución Gamma$(\alpha, 1)$ con la propuesta Gamma$([\alpha], 1)$, donde $[\alpha]$ denota la parte entera de α. Además, realizar el siguiente experimento: poner como punto inicial $x_0 = 900$ y graficar la evolución de la cadena, es decir, $f(X_t)$ vs t.

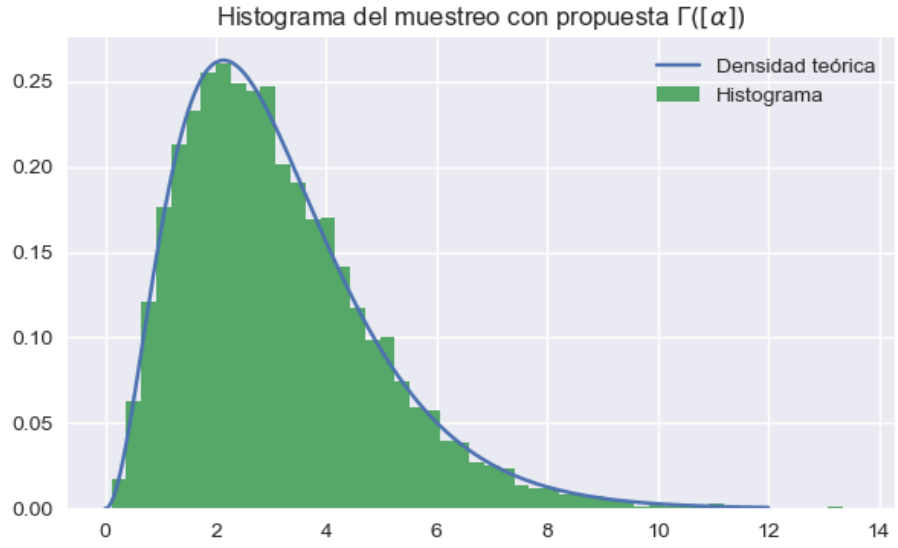
Solución. Primero debemos proponer un valor de α. Es evidente que si $\alpha \in \mathbb{Z}$ el problema no tiene sentido, pues en tal caso estaríamos usando simulaciones de $Ga(\alpha)$ para simular $Ga(\alpha)$, por lo que necesitamos que α no sea entero, por ejemplo $\alpha = \pi$. Como lo dicta el ejercicio, establecemos el punto inicial $x_0 = 900$, y obtenemos la siguiente trayectoria,

Como podemos notar, la cadena empieza en un punto muy alejado de la masa de la distribución, pero inmediantamente en un paso cambia a encontrarse dentro de él, y esto se debe a que, como la transición no depende del punto anterior, no importa dónde nos encontremos, siempre podemos llegar al soporte de la densidad en un paso (de hecho con probabilidad muy alta).

La gráfica de $f(X_t)$ contra t, donde f es la densidad objetivo, es la siguiente



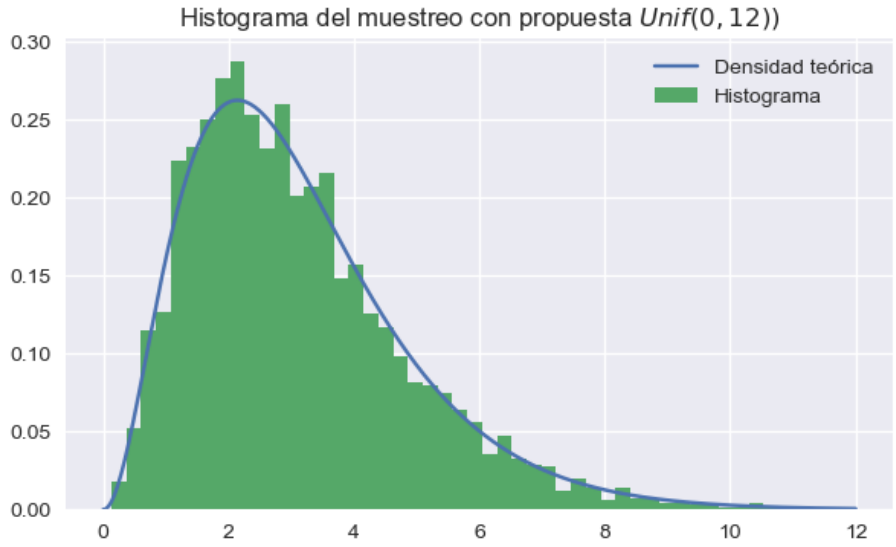
Como sabemos que la densidad propuesta es muy cercana a la densidad objetivo, y la transición no depende del punto anterior, no es necesario realizar una gráfica de log-densidad para conocer el *burn-in*, de hecho, es suficiente descartar el primer punto porque es el único que se encuentra fuera de las zonas donde la distribución objetivo concentra masa. Al descartar el primer punto obtenemos el siguiente histograma,



El ajuste es bastante parecido a la densidad teórica. Esta cadena presenta una convergencia muy rápida, y ella se debe a que contamos con una densidad de transición que es muy similar a nuestra densidad objetivo, lo que ocasiona que la transición se acepte con más frecuencia. No siempre es posible contar con una densidad parecida a la densidad objetivo, por lo que implementar una cadena como esta para muestrear desde cualquier distribución, en general no es posible.

Observando el soporte de la distribución, nos damos cuenta de que la mayor parte de la masa se concentra en el intervalo $(0, 12)$, por lo que otra propuesta de transición razonable sería una uniforme en este intervalo. Al implementar el algoritmo con esta densidad de transición encontramos la siguiente trayectoria,

Como en el caso anterior, el hecho de que la propuesta no dependa del punto anterior, ocasiona que la trayectoria exhiba un comportamiento muy regular a partir del primer paso, pues se puede avanzar por todo el soporte de la densidad, importando únicamente la regla de rechazo de la transición. Similarmente, el *burn-in* también consiste en quitar únicamente el primer punto, el histograma que obtenemos al hacer esto es,



El histograma ahora presenta un nivel de ajuste menor al de la cadena con propuesta $Ga([\alpha])$, ya que la distribución uniforme no es tan similar a la $Ga(\alpha)$. Aunque el diseño de la cadena garantiza la convergencia a la distribución objetvo, una mala elección de la densidad de transición puede ocasionar que esta convergencia sea más lenta. Podemos ver que cuando no tenemos una densidad de transición muy parecida a nuestra densidad objetivo, no es eficiente hacer una cadena cuya transición no dependa del punto anterior. □

3. Implementar Random Walk Metropolis Hasting (RWMH) donde la distribución objetivo es $\mathcal{N}_2(\mu, \Sigma)$, con

$$\mu = \begin{pmatrix} 3 \\ 5 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}.$$

Utilizar como propuesta $\varepsilon_t \sim \mathcal{N}_2(\mathbf{0}, \sigma I)$. ¿Cómo elegir σ para que la cadena sea eficiente? ¿Qué consecuencias tiene la elección de σ?

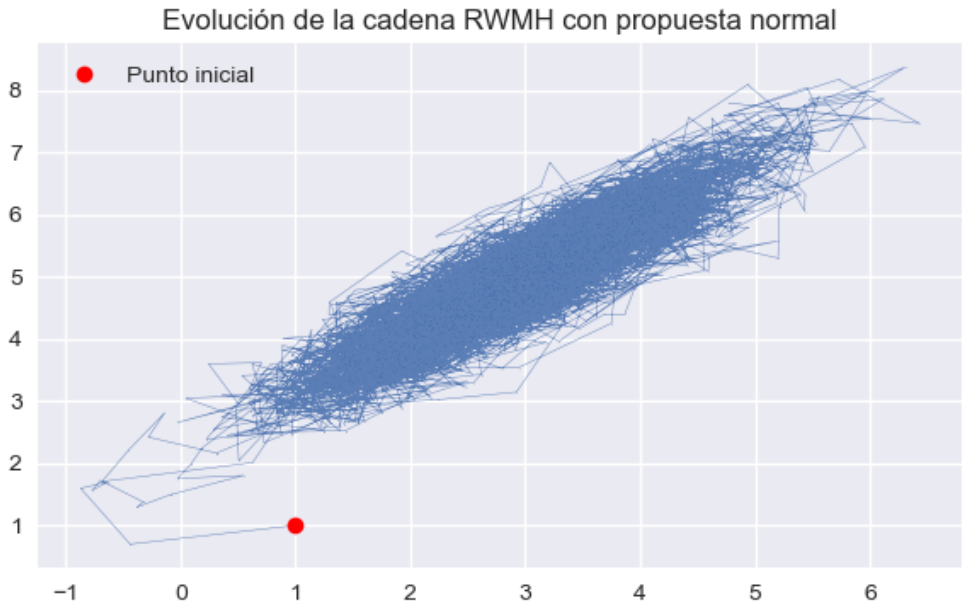
Como experimento, elige como punto inicial $x_o = \binom{1000}{1}$ y comenta los resultados.

Solución. Al implementar el algoritmo con un σ arbitrario, notamos que la longitud de los "pasos" de la cadena está en función de σ. Un σ muy chico ocasiona que la cadena se aleje muy poco del punto inicial, cubriendo una menor parte del soporte, o gastando demasiadas iteraciones en llegar a él. Un σ grande permite que la cadena llegue más rápidamente a los puntos en que se concentra la masa de la distribución objetivo, pero también ocasiona que se tienda a abandonar esa zona con mucha frecuencia, lo que causa un muestreo más lento. Una elección de σ adecuada permite que la cadena alcance la zona de muestreo de la distribución rápidamente y que permanezca en ella durante más tiempo.

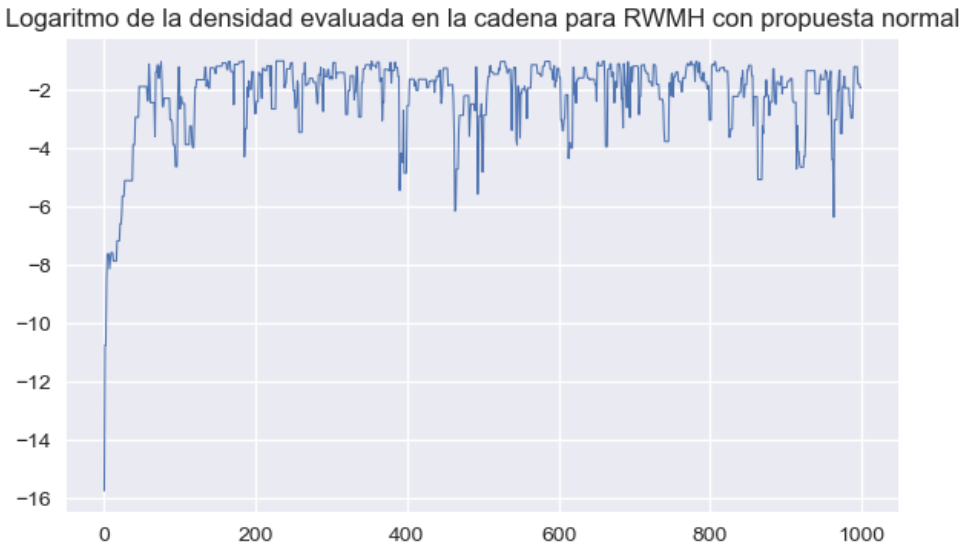
Mi elección de σ se basó en probar con varios valores y observar el comportamiento de la cadena al cubrir el plano, finalmente decidí utilizar $\sigma = 1/2$, pues con este valor se logra un equilibrio entre la velocidad con la que se muestrea, y la permanencia en el soporte de la cadena.

Cuando intentamos iniciar la cadena en el punto $(1000, 1)$ obtenemos un error de división por 0, debido a que el punto $(1000, 1)$ está lo suficientemente lejos de la media de la densidad objetivo como para que la densidad evaluada en ese punto sea equivalente a cero en números de punto flotante. Con esto observamos que no cualquier punto dentro del dominio de la objetivo es una buena elección de punto inicial

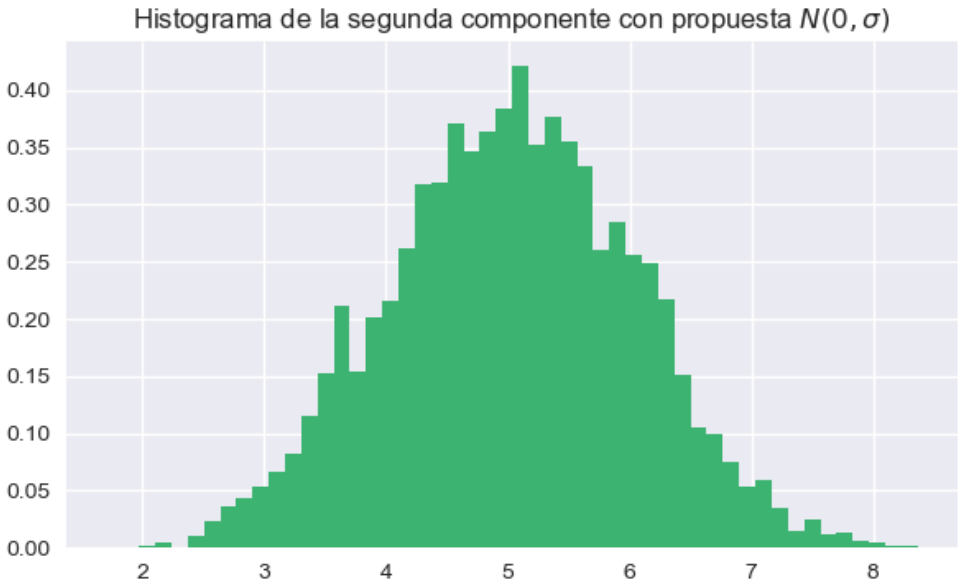
La trayectoria de la cadena con la densidad de transición propuesta y punto inicial $(1, 1)$ es la siguiente,



Notemos que a pesar de comenzar en una zona fuera de la mayor concentración de masa de la objetivo, la cadena rápidamente se concentra en una elipse alrededor de la media (3,5), y pasa la mayor parte de la trayectoria en esta zona. Se realizaron 10,000 iteraciones de esta cadena, pero debido a que llega muy rápido al dominio de la densidad objetivo, su función de log-densidad tiende a estabilizarse rápidamente, por lo que a continuación se muestra esta función evaluada únicamente en los primeros 1,000 puntos,



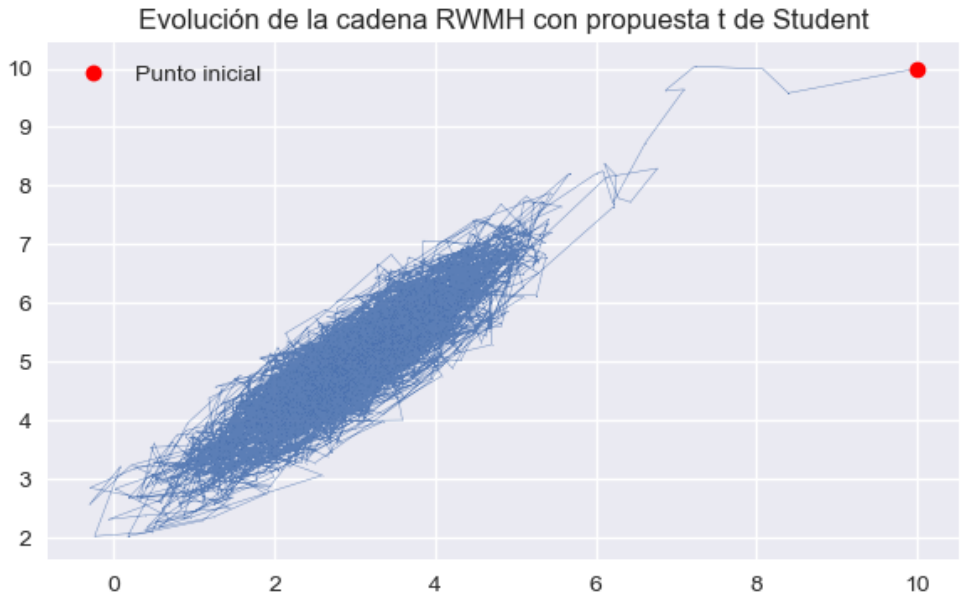
Podemos notar que es aproximadamente alrededor de la iteración 100 que la cadena comienza a encontrarse alrededor de la elipse que concentra la mayoría de la masa de la distribución. Los histogramas para ambas coordenadas al descartar en *burn-in* son los siguientes,



Como se trata de un vector normal multivariado, sus marginales deben ser normales, aunque en este caso no son independientes.

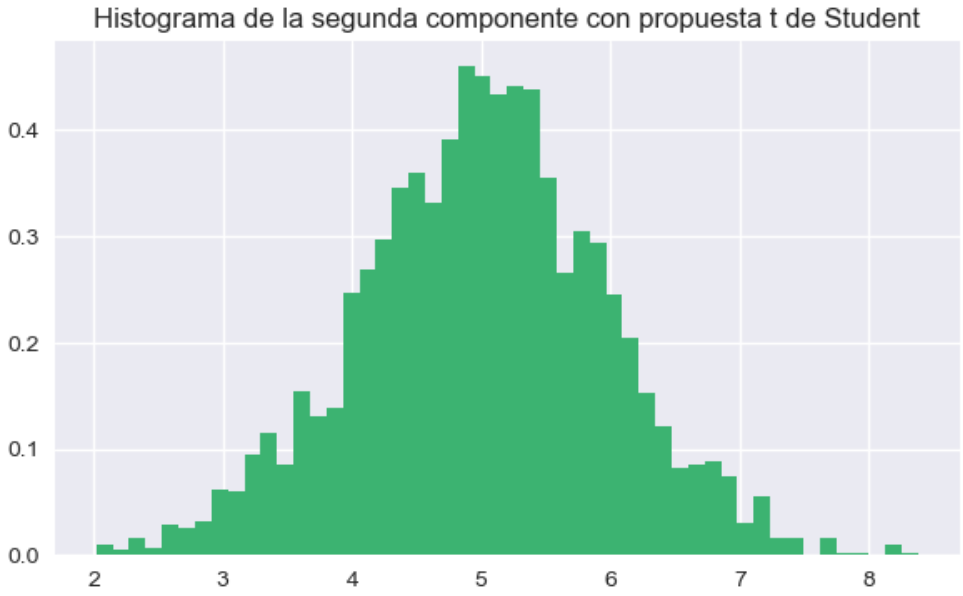
La velocidad de convergencia que observamos en las Figuras anteriores se debe tanto a la elección de la tasa como al punto inicial. Podemos decir que estos dos factores causan que esta cadena sea muy eficiente al muestrear de la densidad objetivo.

Como alternativa a la función normal podemos usar una densidad t de Student como transición en el algoritmo RWMH, al implementarlo con punto inicial $(10, 10)$ obtenemos la siguiente trayectoria,



Como vemos, en este caso también se cumple que la cadena entra en el conjunto de concentración de masa de la objetivo de manera relativamente rápida, y permanece ahí la mayor parte de la trayectoria. Para encontrar el *burn-in* también graficamos únicamente la log-densidad de los primeros 1,000 puntos,

Nuevamente, podemos considerar que la cadena ha llegado a la zona de concentración de masa de la objetivo en 100 pasos, o incluso menos. Al descartar las primeras 100 iteraciones de la cadena obtenemos los siguientes histogramas para las coordenadas de la distribución objetivo,



Ambos histogramas son muy parecidos a los que se tenían en el caso en que la propuesta era una normal multivariada, lo cual se justifica si consideramos que la distribución t de Student multivariada es una aproximación a la distribución normal. □

Para todos los incisos del ejercicio anterior:

- Establece cual es tu distribución inicial.
- Grafica la evolución de la cadena.
- Indica cuál es el Burn-in.
- Comenta qué tan eficiente es la cadena.
- Implementa el algoritmo MH considerando una propuesta diferente.

Como conclusión, en todos los ejercicios anteriores notamos que existen decisiones sencillas en apariencia que pueden hacer una gran diferencia en el desempeño de un algoritmo de Metropolis-Hastings, por lo que el uso de esta técnica no se basa únicamente en el conocimiento de la teoría, sino en la práctica necesaria para saber cuáles parámetros o distribuciones podrían resultar mejor en cada caso.