

# Simulación y Visualización de datos en Julia

## Material Complementario

Act. Enrique Lecuona Sierra

22 de enero de 2026

Universidad Nacional Autónoma de México

### 1. Teorema de la transformada inversa

El teorema de la transformada inversa nos da una estrategia general para generar simulaciones para cualquier variable aleatorias. Lo establecemos a continuación.

**Teorema 1.** Sea  $(\Omega, \mathcal{A}, \mathbb{P})$  un espacio de probabilidad y sea  $X$  una v.a. definida sobre dicho espacio con  $F_X$  su respectiva función de distribución de probabilidades. Si  $U$  es una variable aleatoria  $Unif(0, 1)$  y  $Y = F_X^{-1}(U)$  entonces  $F_Y = F_X$ .

*Demostración.* Obtengamos  $F_Y$ .

$$F_Y(y) = \mathbb{P}(Y \leq y) = \mathbb{P}\left(F_X^{-1}(U) \leq y\right) = \mathbb{P}\left(U \leq F_X(y)\right) = F_U\left(F_X(y)\right)$$

esto consecuencia de que la función de cuantiles  $F_X^{-1}$  es monótona creciente. Adicional a esto, recordemos que la F.d.p. de  $U$  queda como sigue

$$F_U(u) = u\mathbb{1}_{[0,1[}(u) + \mathbb{1}_{[1,\infty[}(u)$$

y como  $F_X(y)$  es un número en el intervalo  $[0, 1]$  se cumple que lo siguiente

$$F_Y(y) = F_U\left(F_X(y)\right) = F_X(y)$$

y esto se cumple  $\forall y \in \text{Ran } Y$ , cumpliéndose lo deseado.

□

Una vez sabemos esto, el algoritmo para simular valores aleatorios de cualquier modelo univariado queda como sigue.

**Algoritmo 1.1.** Simulación de una variable aleatoria mediante el método de la transformada inversa.

Sea  $X$  una variable aleatoria con función de distribución de probabilidades  $F_X$ . Para obtener una muestra de tamaño  $n$  para  $X$  realizamos lo siguiente:

1) Simulamos una muestra aleatoria de tamaño  $n$  para el modelo  $\text{Unif}(0,1)$ :

$$\{u_1, \dots, u_n\}$$

2) Aplicamos la función de cuantiles de  $X$  a cada simulación de nuestra  $\text{Unif}(0,1)$  quedando nuestra muestra de  $X$  como sigue:

$$\{x_1, \dots, x_n\}; \quad x_j = F_X^{-1}(u_j)$$

**Ejemplo 1.1.** Sea  $X$  una variable aleatoria cualquiera y sea  $Y = \sigma X + \mu$  con  $\sigma > 0$ . Obtengamos  $F_Y^{-1}$  en términos de  $F_X^{-1}$ .

Primeramente obtengamos  $F_Y$  en términos de  $F_X$

$$F_Y(y) = \mathbb{P}(Y \leq y) = \mathbb{P}(\sigma X + \mu \leq y) = \mathbb{P}\left(X \leq \frac{y - \mu}{\sigma}\right) = F_X\left(\frac{y - \mu}{\sigma}\right)$$

Ahora bien, para obtener  $F_Y^{-1}$  en términos de  $F_X^{-1}$  tomemos un  $u \in [0, 1]$  tal que  $F_Y(y) = u$  así  $y = F_Y^{-1}(u)$  pero también  $F_X\left(\frac{y - \mu}{\sigma}\right) = u$  por lo que  $y = \sigma F_X^{-1}(u) + \mu$  cumpliéndose la igualdad siguiente

$$F_Y^{-1}(u) = \sigma F_X^{-1}(u) + \mu$$

$\forall u \in [0, 1]$ .

El anterior ejemplo nos dice que si tenemos una v.a  $X$  y deseamos agregarle un parámetro de escala  $\sigma$  y uno de localización  $\mu$ , la generación de valores aleatorios para esta nueva v.a. se reduce a generar valores para la v.a. inicial y realizar la transformación siguiente de la información.

$$\{y_1, \dots, y_n\}; \quad y_j = \sigma x_j + \mu$$

**Ejemplo 1.2.** Sea  $X$  la variable aleatoria con la siguiente función de densidad de probabilidades para  $n \in \mathbb{N}$ :

$$f_X(x) = nx^{n-1} \mathbb{1}_{]0,1[}(x)$$

Para generar una simulación perteneciente a esta variable aleatoria necesitamos obtener  $F_X^{-1}$ . Para esto primeramente obtenemos  $F_X$ , la cual queda como sigue:

$$F_X(x) = x^n \mathbb{1}_{[0,1]}(x) + \mathbb{1}_{[1,\infty)}(x)$$

por lo que  $F_X^{-1}$  queda como sigue:

$$F_X^{-1}(u) = u^{\frac{1}{n}}$$

$\forall u \in [0, 1]$

## 2. Simulación de Vectores Aleatorios

Existen dos estrategias básicas para generar información de un vector aleatorio  $(X, Y)$ . Las presentamos a continuación.

## 2.1. Simulación de Vectores Aleatorios utilizando Distribución Condicional

Para todo vector aleatorio  $(X, Y)$  podemos realizar simulación utilizando la función de distribución condicional de probabilidades. Recordemos que la función de distribución bivariada tiene la siguiente descomposición.

$$F_{XY} \leftrightarrow F_{Y|\{X=x\}}, F_X$$

el proceso de simulación se establece en el siguiente algoritmo.

**Algoritmo 1.2.** *Simulación de un vector aleatorio mediante el método de la distribución condicional.*

Sea  $(X, Y)$  un vector aleatorio con función de distribución de probabilidades bivariada  $F_{XY}$ . Para obtener una muestra de tamaño  $n$  para el vector  $(X, Y)$  realizamos lo siguiente:

- 1) Simulamos una muestra aleatoria de tamaño  $n$  para la variable aleatoria  $X$ :

$$\{x_1, \dots, x_n\}$$

- 2) Para cada  $x_j$ , simulamos un dato del modelo condicional  $F_{Y|\{X=x_j\}}$ . Quedando la muestra de  $(X, Y)$  como sigue:

$$\{(x_1, y_1), \dots, (x_n, y_n)\}$$

**Ejemplo 1.3.** Sea  $(X, Y)$  el vector aleatorio con la siguiente función de densidad de probabilidades bivariada:

$$f_{XY}(x, y) = \mathbb{1}_T(x, y)$$

donde  $T = \{(x, y) \in \mathbb{R}^2 | |x| \leq 1; 0 \leq y \leq 1 - |x|\}$ . Para este ejemplo podemos concluir las distribuciones condicionales  $Y|X$  y  $X|Y$  y las distribuciones univariadas de  $X$  y  $Y$ :

$$\begin{aligned} f_X(x) &= (1 - |x|) \mathbb{1}_{-1 \leq x \leq 1}(x); & Y|\{X = x\} &\sim \text{Unif}(0, 1 - |x|) \\ f_Y(y) &= 2(1 - y) \mathbb{1}_{0 \leq y \leq 1}(y); & X|\{Y = y\} &\sim \text{Unif}(y - 1, 1 - y) \end{aligned}$$

se puede realizar simulación para este vector de dos formas distintas

- 1) Simulando información de  $X$  y luego utilizar la variable aleatoria condicional  $Y|\{X = x\}$ .
- 2) Simulando información de  $Y$  y luego utilizar la variable aleatoria condicional  $X|\{Y = y\}$ .

Utilizando el modelo univariado de  $Y$  podemos concluir que su función de cuantiles queda como sigue:

$$F_Y^{-1}(u) = 1 - \sqrt{1 - u}; \quad u \in [0, 1]$$

## 2.2. Simulación de Vectores Aleatorios utilizando Cópulas

Establecemos el teorema de Sklar a continuación.

### Teorema 2. Teorema de Sklar

Sea  $(\Omega, \mathcal{A}, \mathbb{P})$  un espacio de probabilidad cualquiera, entonces se cumple lo siguiente:

- 1) Para un vector aleatorio  $(X, Y)$  existe una función cópula  $C$  tal que

$$F_{XY}(x, y) = C(F_X(x), F_Y(y))$$

- 2) Si  $X$  y  $Y$  son variables aleatorias cualesquiera y  $C$  es una función cópula entonces la función  $H : \mathbb{R}^2 \rightarrow [0, 1]$  definida como sigue:

$$H(x, y) = C(F_X(x), F_Y(y))$$

es una función de distribución de probabilidades bivariada tal que sus marginales son  $F_X$  y  $F_Y$ , respectivamente.

Esto nos permite realizar la siguiente descomposición para el modelo bivariado  $(X, Y)$

$$F_{XY} \leftrightarrow C_{XY}, F_Y, F_X$$

donde  $F_X$  y  $F_Y$  se llevan la incertidumbre univariada de nuestra información mientras la cópula  $C_{XY}$  se lleva solamente la información de **dependencia o relación** entre las variables aleatorias  $X$  y  $Y$ . Podemos crear un algoritmo de simulación de un vector aleatorio mediante cópulas como sigue:

### Algoritmo 2.1. Simulación de un vector aleatorio mediante el método de cópulas.

Sea  $(X, Y)$  un vector aleatorio con función de distribución de probabilidades bivariada  $F_{XY}$  y cópula  $C_{XY}$ . Para obtener una muestra de tamaño  $n$  para el vector  $(X, Y)$  realizamos lo siguiente:

- 1) Simulamos una muestra aleatoria de tamaño  $n$  de la cópula  $C_{X,Y}$ :

$$\{(u_1, v_1), \dots, (u_n, v_n)\}$$

- 2) Para cada  $(u_j, v_j)$ , aplicamos el método de la transformada inversa con las funciones de cuantiles  $F_X^{-1}$  y  $F_Y^{-1}$ , respectivamente. Quedando la muestra de  $(X, Y)$  como sigue:

$$\{(x_1, y_1), \dots, (x_n, y_n)\}; \quad x_j = F_X^{-1}(u_j); \quad y_j = F_Y^{-1}(v_j)$$

## 2.3. Transformación de una muestra aleatoria

En el ámbito estadístico, es de interés transformar la información de una muestra aleatoria  $\mathbf{X} = (X_1, \dots, X_n)$ , es de decir, nos interesa conocer la  $F_T$  para  $T = t(X_1, \dots, t_n)$ . Obtener una simulación de la variable aleatoria  $T$  se realiza de la siguiente forma:

### Algoritmo 2.2. Simulación de una transformación de una muestra aleatoria.

Sea  $\mathbf{X} = (X_1, \dots, X_n)$  una muestra aleatoria con función de distribución de probabilidades  $F_{\mathbf{X}}$  y sea  $T = t(X_1, \dots, t_n)$  una transformación de dicha muestra aleatoria. Para obtener una muestra de tamaño  $m$  para la muestra aleatoria  $\mathbf{X}$  realizamos lo siguiente:

1) Realizamos una simulación de tamaño  $m$  para la muestra aleatoria  $\mathbf{X}$ :

$$\{(x_1^{(1)}, \dots, x_n^{(1)}), \dots, (x_1^{(m)}, \dots, x_n^{(m)})\}$$

donde cada  $(x_1^{(j)}, \dots, x_n^{(j)})$  es **una** observación de la muestra aleatoria  $\mathbf{X} = (X_1, \dots, X_n)$ .

2) Para cada  $(x_1^{(j)}, \dots, x_n^{(j)})$ , aplicamos la transformación  $t$ . Quedando la muestra de  $T$  como sigue:

$$\{t_1, \dots, t_m\}; \quad t_j = t(x_1^{(j)}, \dots, x_n^{(j)})$$

**Ejemplo 2.1.** Sea  $\mathbf{X} = (X_1, \dots, X_n)$  una muestra aleatoria de  $\text{Unif}(0, 1)$ , deseamos obtener una simulación de  $T = X_{(n)}$ . Recordemose que  $F_X$  se ve como sigue:

$$F_X(x) = x \mathbb{1}_{[0,1[}(x) + \mathbb{1}_{[1,\infty[}(x)$$

por lo que  $F_T$  queda como sigue:

$$F_T(t) = (F_X(t))^n = [t \mathbb{1}_{[0,1[}(t) + \mathbb{1}_{[1,\infty[}(t)]^n = t^n \mathbb{1}_{[0,1[}(t) + \mathbb{1}_{[1,\infty[}(t)$$

se puede realizar simulación para  $T$  de dos formas distintas

- 1) Simulando información de  $T$  directamente utilizando el teorema de la transformada inversa.
- 2) Simulando información de  $\mathbf{X} = (X_1, \dots, X_n)$  y aplicando la transformación  $T = t(X_1, \dots, X_n)$  a los datos de  $\mathbf{X}$ .

### 3. Aplicación de Simulación: Integración por Método Monte Carlo

Para poder entender la estrategia de integración por método Monte Carlo básica, es necesario hacer mención de un resultado clásico de la teoría probabilística, se omite la demostración pero se puede encontrar en *Jacod y Protter, Probability Essentials, Springer (2004)*.

**Teorema 3. La ley de los grandes números**

Sea  $(X_n)_{n \in \mathbb{N}}$  una sucesión de variables aleatorias independientes e idénticamente distribuidas en  $\mathcal{L}^2$  definidas sobre el mismo espacio de probabilidad  $(\Omega, \mathcal{A}, \mathbb{P})$  y sean  $\mu = \mathbb{E}(X_j)$  y  $\sigma^2 = \mathbb{V}(X_j)$ . Definimos la siguiente sucesión de variables aleatorias  $(S_n)_{n \in \mathbb{N}}$ :

$$S_n = \frac{1}{n} \sum_{j=1}^n X_j$$

entonces se cumple lo siguiente:

- 1) **Ley débil de los grandes números:** La sucesión  $(S_n)_{n \in \mathbb{N}}$  converge en probabilidad a la variable aleatoria constante  $\mu$ . Es decir, para cualquier  $\epsilon > 0$ ,

$$\lim_{n \rightarrow \infty} \mathbb{P}(\{w : |S_n(w) - \mu| > \epsilon\}) = 0$$

lo cual denotamos como  $S_n \xrightarrow{\mathbb{P}} \mu$ .

2) **Ley fuerte de los grandes números:** La sucesión  $(S_n)_{n \in \mathbb{N}}$  converge casi seguramente y en  $\mathcal{L}^2$  a la variable aleatoria constante  $\mu$ .

2.1)  $(S_n)_{n \in \mathbb{N}}$  converge casi seguramente a la variable aleatoria constante  $\mu$ . Es decir, si  $N$  es el siguiente conjunto,

$$N = \left\{ w : \lim_{n \rightarrow \infty} S_n(w) \neq \mu \right\}$$

entonces  $\mathbb{P}(N) = 0$ .

2.2)  $(S_n)_{n \in \mathbb{N}}$  converge en  $\mathcal{L}^2$  a la variable aleatoria constante  $\mu$ . Es decir, se cumple el siguiente límite,

$$\lim_{n \rightarrow \infty} \mathbb{E}(|S_n - \mu|^2) = 0$$

Como consecuencia de este teorema y del poder computacional actual, podemos aproximar computacionalmente cualquier integral definida sobre un conjunto del tipo  $[a, b]$ . Esto lo hacemos mediante el siguiente procedimiento. Sea  $f : [a, b] \rightarrow \mathbb{R}$  una función integrable. Deseamos aproximar la siguiente integral,

$$I = \int_a^b f(x) dx$$

Podemos expresar la integral anterior como sigue,

$$I = \int_a^b f(x) dx = (b - a) \int_a^b \frac{1}{b - a} f(x) dx = (b - a) \mathbb{E}(f(U))$$

donde  $U \sim \text{Unif}(a, b)$ . Como consecuencia de la ley de los grandes números, podemos aproximar la anterior esperanza mediante una observación de la variable aleatoria  $U$  y el valor de  $I$  queda como sigue,

$$I \approx (b - a) \sum_{j=1}^n f(u_j)$$

donde  $\{u_1, \dots, u_n\}$  es la observación de  $U$  simulada computacionalmente. La precisión de la integración se mejora conforme  $n \rightarrow \infty$ .

**Ejemplo 3.1.** Tomemos la siguiente función  $f(x) = \log(x)$  y realicemos computacionalmente la siguiente integral,

$$I = \int_1^e f(x) dx$$

mediante integración por partes, nosotros podemos resolver la anterior integral, quedando como sigue,

$$I = \int_1^e f(x) dx = \left( x \log(x) - x \right) \Big|_1^e = 1$$

Mediante una simulación de tamaño  $n = 1000000$  de una  $Unif(1, e)$ , podemos obtener una aproximación a esta integral la cual queda como sigue,

$$I = (e - 1)\mathbb{E}(f(U)) \approx (e - 1) \sum_{j=1}^{1000000} \log(u_j) = 0,99959$$

quedando nuestro resultado bastante cercano al resultado teórico exacto. Si se requiere de mayor precisión, se puede incrementar el tamaño de la muestra utilizada para aproximar  $\mathbb{E}(f(U))$  a costa de tiempo de procesamiento.