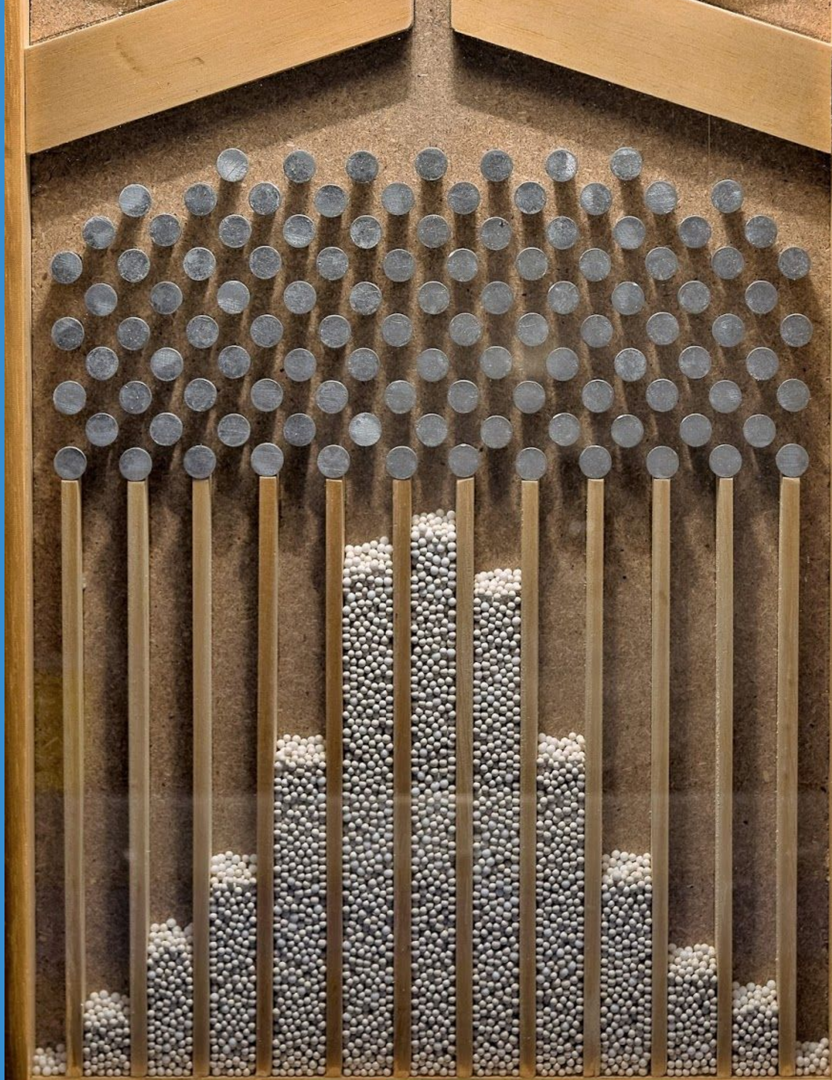
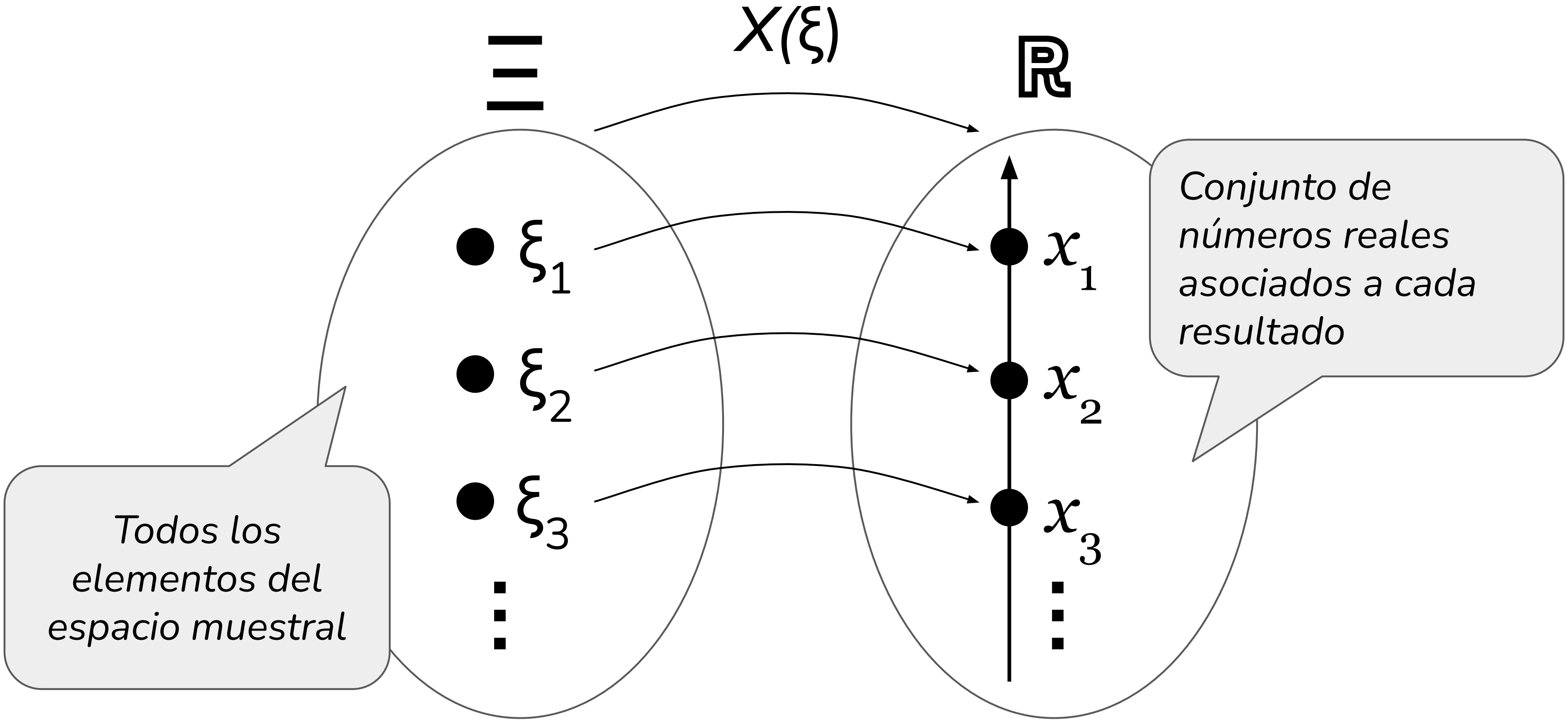
Procesos estocásticos (86.09)

- Variables y vectores aleatorios



Variables Aleatorias

Variables Aleatorias (VA)

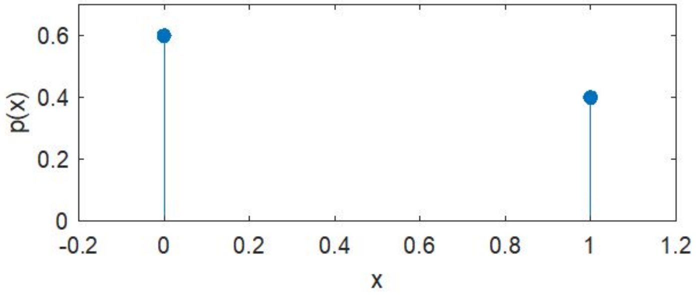


Distribuciones de probabilidad

Distribuciones de probabilidad – Funciones de masa

Bernoulli: $X \sim \mathrm{Ber}(p)$

$$p_X(x) = \begin{cases} 1-p & \text{si } x = 0, \\ p & \text{si } x = 1. \end{cases} \qquad p \in [0,1]$$



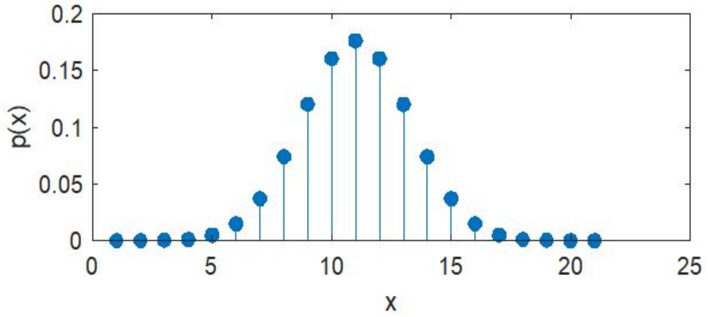
- Lanzamiento de una moneda.
- Estado de un Bit.

$$\mathrm{E}[X] = p \qquad \mathrm{Var}(X) = p(1-p)$$

Distribuciones de probabilidad – Funciones de masa

Binomial: $X \sim \mathrm{Bin}(n, p)$

$$p_X(x) = \binom{n}{x} p^x (1-p)^{n-x} \qquad p \in [0,1] \qquad x = 0, 1, \ldots, n$$



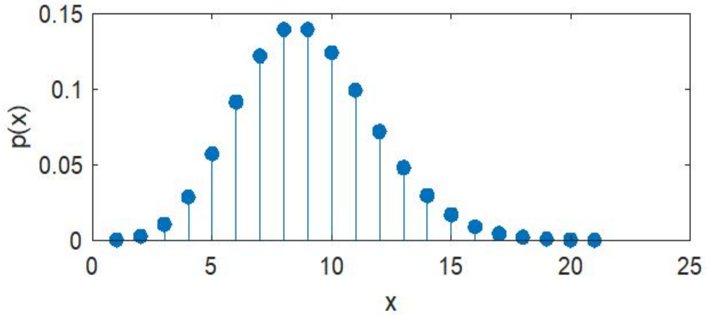
- Cantidad de productos fallados en un lote de 80 unidades.
- Cantidad de personas que responden una encuesta de 50 consultadas.

$$\mathrm{E}[X] = np \qquad \mathrm{Var}(X) = np(1-p)$$

Distribuciones de probabilidad – Funciones de masa

Poisson: $X \sim \text{Poisson}(\lambda)$

$$p_X(x) = \frac{\lambda^x e^{-\lambda}}{x!} \qquad \begin{array}{l} \lambda > 0 \\ x = 1, 2 \ldots \end{array}$$



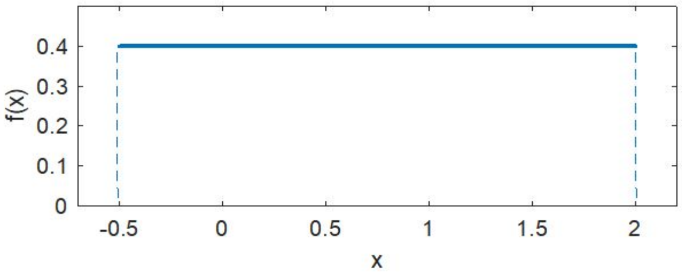
- Cantidad de clientes que llegan a un local durante 1 hora.
- Cantidad de fallas en una tela de 5 metros de largo.

$$E[X] = \lambda \qquad \text{Var}(X) = \lambda$$

Distribuciones de probabilidad – Funciones de densidad

Poisson: $X \sim U(a, b)$

$$f_X(x) = \frac{1}{b-a}, \qquad a < x < b.$$



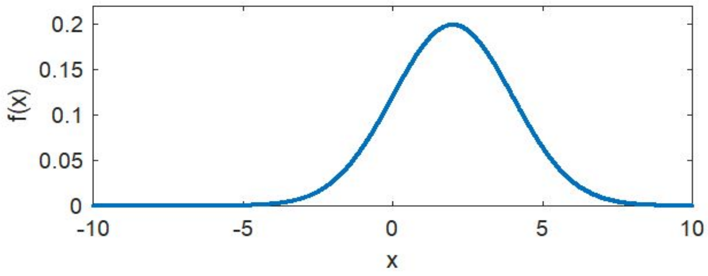
- Tiempo de espera en un semáforo.
- Posición en la que cae una gota de lluvia en un recipiente.

$$\mathrm{E}[X] = \frac{a+b}{2} \qquad \mathrm{Var}(X) = \frac{(b-a)^2}{12}$$

Distribuciones de probabilidad – Funciones de densidad

Poisson: $X \sim N(\mu, \sigma^2)$

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \qquad x \in \mathbb{R}$$



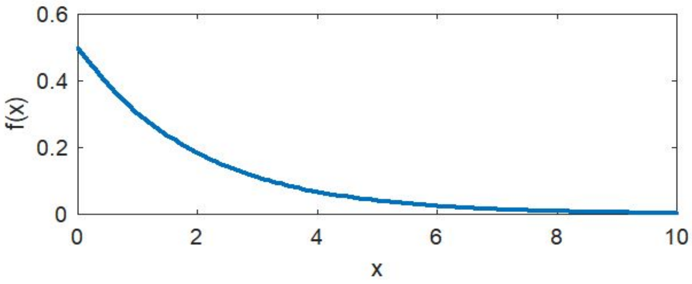
- Ruido térmico en señales electrónicas
- Peso de productos manufacturados

$$\mathrm{E}[X] = \mu \qquad\qquad \mathrm{Var}(X) = \sigma^2$$

Distribuciones de probabilidad – Funciones de densidad

Poisson: $X \sim \mathrm{Exp}(\lambda)$

$$f_X(x) = \lambda\, e^{-\lambda x}, \qquad x \geq 0$$



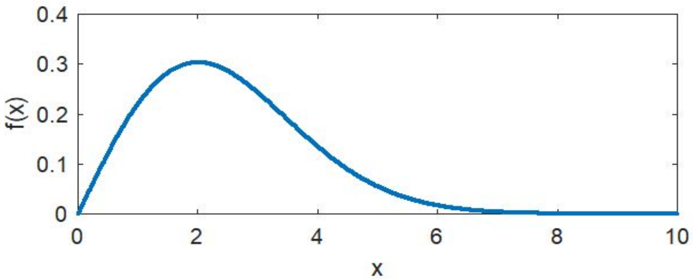
- Tiempo entre dos clientes que ingresan a un banco.
- Distancia entre fragmentos de meteorito caídos en tierra.

$$\mathrm{E}[X] = \frac{1}{\lambda} \qquad\qquad \mathrm{Var}(X) = \frac{1}{\lambda^2}$$

Distribuciones de probabilidad – Funciones de densidad

Rayleigh: $X \sim \mathrm{Rayleigh}(\sigma)$

$$f_X(x) = \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}, \qquad x \geq 0 \quad \sigma > 0$$



- Velocidad del viento en meteorología.
- Amplitud de una señal inalámbrica al propagarse.

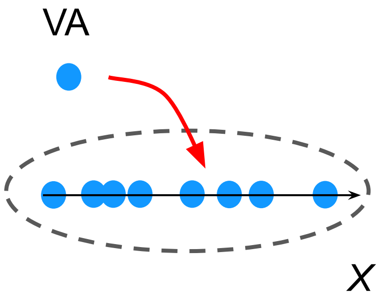
$$E[X] = b\sqrt{\pi/2} \qquad \mathrm{Var}(X) = \frac{4-\pi}{2} b^2$$

Simulación de Variables Aleatorias

Simulación de Variables Aleatorias

Funciones de Matlab/Octave para generar muestras de distribuciones comunes:

```
x = rand(1,N);                  % Uniforme estándar
x = unifrnd(a, b, 1, N);        % Uniforme
x = randn(1,N);                 % Normal estándar
x = normrnd(mu, sig, 1, N);     % Normal
x = binornd(n, p, 1, N);        % Binomial
x = poissrnd(mu, 1, N);         % Poisson (mu = lambda)
x = exprnd(mu, 1, N);           % Exponencial(mu = 1/lambda)
x = raylrnd(b, 1, N);           % Rayleigh
```

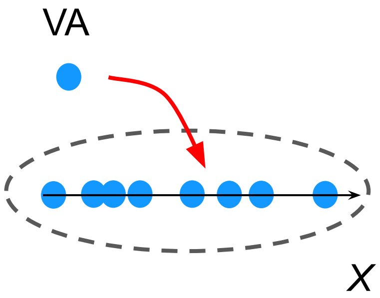


N muestras con cierta distribución (realizaciones)

Simulación de Variables Aleatorias

Funciones de Python para generar muestras de distribuciones comunes:

```
import numpy as np

x = np.random.uniform(a, b, N)                  # Uniforme

x = np.random.normal(mu, sig, N)                # Normal

x = np.random.binomial(n, p, N)                 # Binomial

x = np.random.poisson(mu, N)                    # Poisson

x = np.random.exponential(mu, N)                # Exponencial

x = np.random.rayleigh(scale=b, size=N)         # Rayleigh
```



N muestras con cierta distribución (realizaciones)

Simulación de Variables Aleatorias (Matlab)

```
>> x = rand(1,5)

x =

    0.0975    0.2785    0.5469    0.9575    0.9649
```

Realizaciones independientes

```
>> x = randn(1,6)

x =

   -1.3499    3.0349    0.7254   -0.0631    0.7147   -0.2050
```

Simulación de Variables Aleatorias (Python)

```
import numpy as np
x = np.random.uniform(0, 1, 5) # x = np.random.rand(5)
print(x)
```

```
[0.24230626 0.56564437 0.14763606 0.97714927 0.31140779]
```

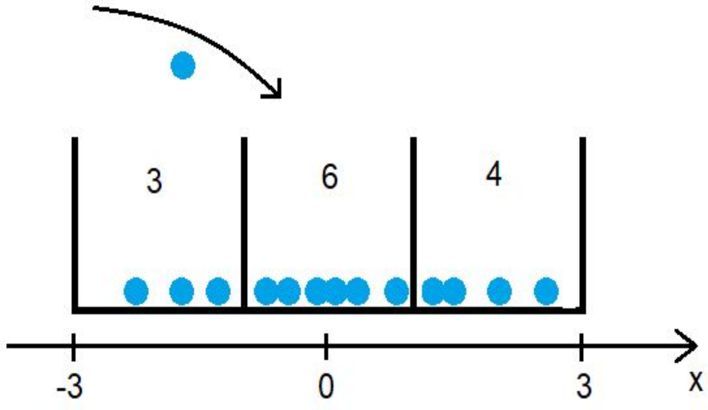
```
x = np.random.normal(0, 1, 6) # x = np.random.randn(6)
print(x)
```

```
[ 1.03918166  0.50593943 -0.35094316  1.12961749 -0.73663994 -0.6805176 ]
```

Histogramas

Histogramas

- Permite representar una aproximación de la función de **densidad**
- Representa la **frecuencia de ocurrencia** de una dada VA dentro de cada bin
- Divide el eje de la VA en **intervalos (bins)**
- **Debe** representarse **normalizada** con área unitaria.



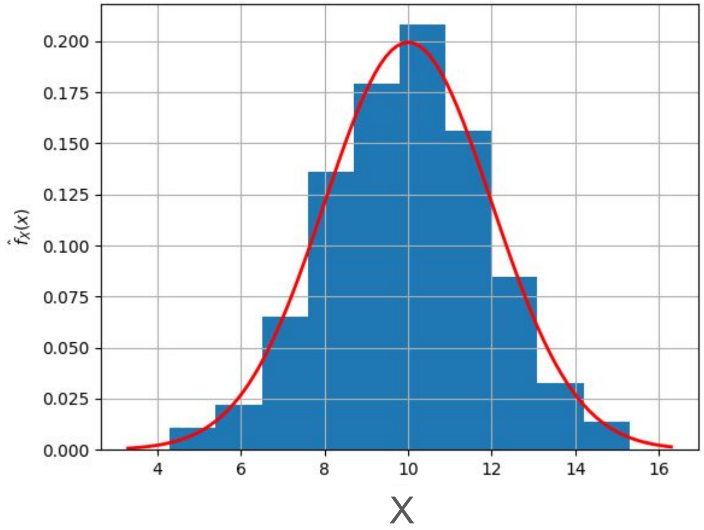
$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} \sum_{j} \mathbf{1}\{x_i \in B_j\} \cdot \mathbf{1}\{x \in B_j\},$$

$$B_j = [m_j - \frac{h}{2}, m_j + \frac{h}{2}]$$

h y m_j son la longitud y centro del intervalo.

Histogramas

- Permite representar una aproximación de la función de **densidad**
- Representa la **frecuencia de ocurrencia** de una dada VA dentro de cada bin
- Divide el eje de la VA en **intervalos (bins)**
- **Debe** representarse **normalizada** con área unitaria.



Histogramas

Matlab

```
histogram(x)              % Por defecto gráfica frecuencia de ocurrencia con
                          % bins en automático(se ajusta)

histogram(x, bins)        % Se puede especificar la cantidad de bins

histogram(x,bins,'Normalization','pdf') % Normalización (para comparar
                                        % con la función de densidad)
```

Matlab/Octave

```
h = hist(x, bins);        % Guardar en una variable los valores de hist()

[h, xc] = hist(x, bins); % Guardar histograma normalizado y graficar
bar(xc, h /(sum(h)*(xc(2)-xc(1))));
```

Histogramas

Python

```python
import matplotlib as plt

plt.hist(x) # Por defecto gráfica frecuencia de ocurrencia con 20 bins

plt.hist(x, bins) # Se puede especificar la cantidad de bins

plt.hist(x, bins, density = True)  # Histograma normalizado

                                  # PDF aprox. (el área cierra a 1)
```

Momentos de una variable aleatoria

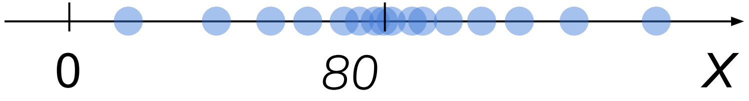
Momentos de una variable aleatoria

Una ferretero quiere saber el peso promedio de los bulones que compró a dos proveedores distintos

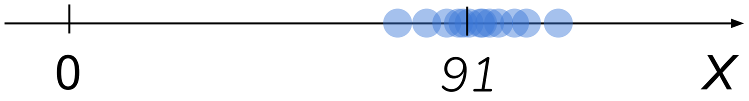
Variable aleatoria X:
peso de un bulón

Ejemplos, suponiendo 16 muestras al azar

Proveedor 1 (sin control de calidad):



Proveedor 2 (con control de calidad):



Momentos de una variable aleatoria

Esperanza

$$\mathbb{E}[X] = \mu_X$$

Varianza (medida de dispersión)

$$\mathbb{V}[X] = \sigma_X^2 = \mathbb{E}\left[(X - \mu_X)^2\right]$$

Covarianza entre dos VA

$$\mathrm{Cov}(X, Y) = \mathbb{E}[(X - \mu_X)(Y - \mu_Y)]$$

Coeficiente de correlación

$$\rho(X, Y) = \frac{\mathrm{Cov}(X, Y)}{\sigma_X \, \sigma_Y} \; ; \; -1 \leq \varrho \leq 1$$

Estimación de algunas medidas estadísticas (Matlab)

Funciones de Matlab

```matlab
mean(x);           % Media de x

var(x);            % Varianza de x

std(x);            % Desvío de x

corrcoef(x, y);    % Coeficiente de correlación entre x e y (columnas)
```

Estimación de algunas medidas estadísticas (Matlab)

Funciones de Matlab/Octave

```
>> mean(x)

ans =

    0.6462
```

```
>> var(x)

ans =

    0.1242
```

```
>> std(x)

ans =

    0.3524
```

```
>> corrcoef(x,y)

ans =

    1.0000    0.2038
    0.2038    1.0000
```

Estimación de algunas medidas estadísticas (Python)

Funciones de Python

```
import numpy as np

mean_x = np.mean(x)                   # Media

var_x = np.var(x)                     # Varianza

std_x = np.std(x)                     # Desvío

corr_coef = np.corrcoef(x, y)        # Coeficiente de correlación
```

Estimación de algunas medidas estadísticas (Python)

```
import numpy as np
x = np.random.normal(0, 1, 1000)
y = np.random.normal(0, 1, 1000)
```

```
mean_x = np.mean(x)
print(mean_x)
```

```
-0.014969036499220583
```

```
std_x = np.std(x)
print(std_x)
```

```
1.000321491280756
```

```
var_x = np.var(x)
print(var_x)
```

```
1.0006430859181559
```

```
corr_coef = np.corrcoef(x, y)
print(corr_coef)
```

```
[[ 1.         -0.04528167]
 [-0.04528167  1.        ]]
```

Problema 1

(Guía 1, Ejercicio 2.7)

Problema 1

Genere N experimentos de una variable aleatoria Rayleigh con parámetro $b = 0.5$. Grafique su histograma para los siguientes parámetros:

1. $N = 100$, bins = 10
2. $N = 100$, bins = 30
3. $N = 10000$, bins = 30

Problema 2

(Guía 1, Ejercicio 2.5)

Problema 2

Una empresa estudia la cantidad de respuestas X que los usuarios responden antes de cortar el teléfono en una encuesta telefónica. Se asume que la cantidad de respuestas contestadas responde a una distribución de Poisson. Además, se sabe que es tres veces más probable que una llamada tenga 2 respuestas a que tenga 4 respuestas. Halle el valor esperado de X.

X = "respuestas a la encuesta antes de cortar"

$X \sim Poi(\lambda)$

$p(x) = \frac{\lambda^x e^{-x}}{x!} \mathbb{1}\{x \in \mathbb{N}\}$

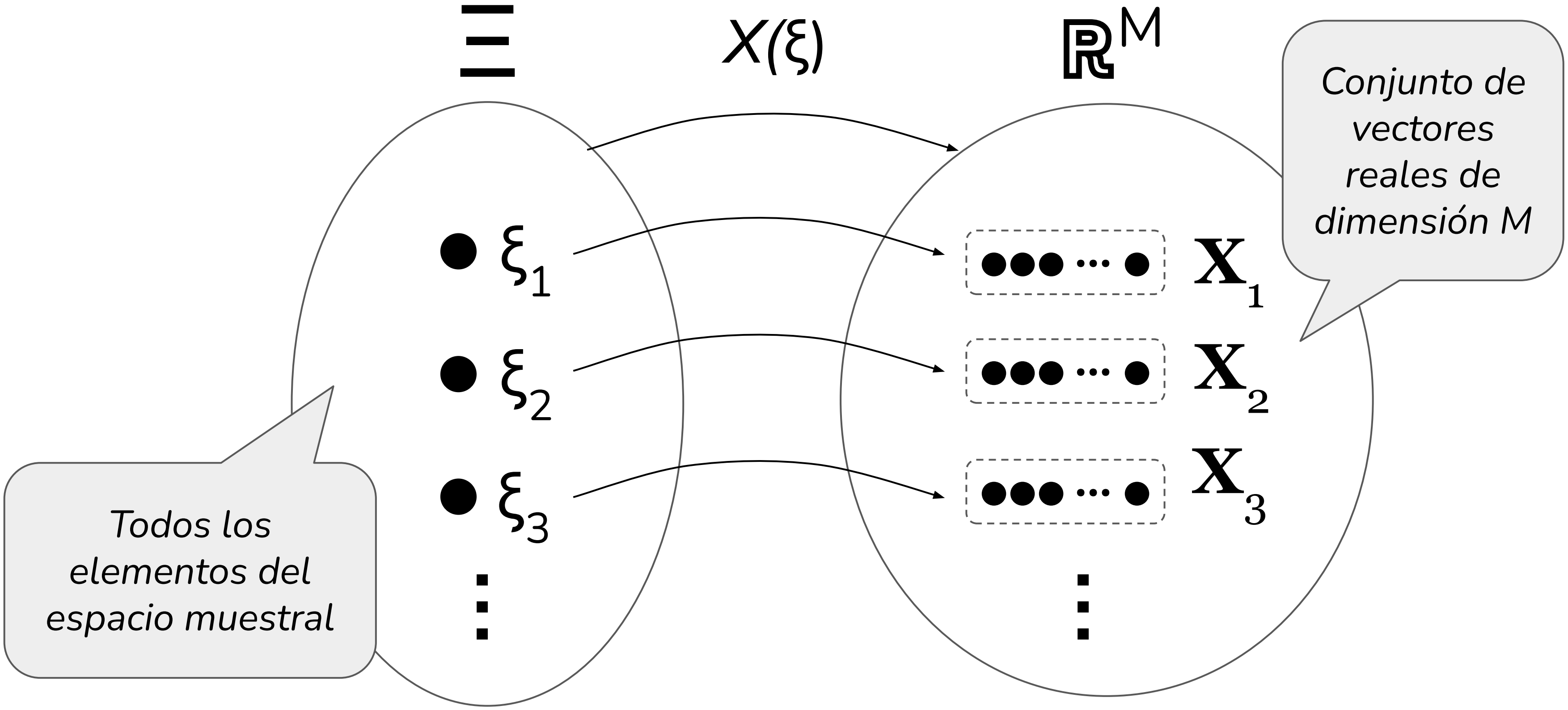
$\sum_{x=0}^{\infty} p_X(x) = 1$

$\mathbb{P}(X=2) = 3\mathbb{P}(X=4) \quad p_X(2) = 3 p_X(4)$

$\frac{\cancel{\lambda^2} \cancel{e^{-\lambda}}}{2!} = \frac{3 \cancel{\lambda^4}^{\lambda^2} \cancel{e^{-\lambda}}}{4!} \Rightarrow \lambda^2 = \frac{4!}{2! \cdot 3} = 4 \rightarrow \boxed{\lambda = 2}$

Distribuciones multivariadas

Vectores Aleatorios (VeA)



Momento de primer orden de un Vector Aleatorio

Vector aleatorio

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix}$$

VA $\in \mathbb{R}$ (escalar)

VA $\in \mathbb{R}^M$ (vectorial)

Media de un vector aleatorio $\mathbf{X}$

$$\mathbb{E}[\mathbf{X}] = \mu_{\mathbf{X}} = \mathbb{E}\left[\begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}\right] = \begin{bmatrix} \mathbb{E}[X_1] \\ \vdots \\ \mathbb{E}[X_n] \end{bmatrix}$$

Simulación de Vectores aleatorios

Simulación de un Vector Aleatorio (Matlab)

Realizaciones iid de un VeA uniforme de dimensión 1x5

```
>> x = rand(1,5)

x =

    0.0975    0.2785    0.5469    0.9575    0.9649
```

Realizaciones iid de un VeA normal de dimensión 1x5

```
>> x = randn(1,5)

x =

   -1.3499    3.0349    0.7254   -0.0631    0.7147
```

Simulación de un Vector Aleatorio (Matlab)

2 realizaciones iid de un VeA uniforme de dimensión 1x5

```
>> x = rand(2,5)

x =

    0.4519    0.8644    0.5398    0.6779    0.7095

    0.7685    0.7278    0.7395    0.5265    0.3678
```

También puede verse como **5 realizaciones** iid de un VeA uniforme de dimensión 2x1

```
>> x = rand(2,5)

x =

    0.4519    0.8644    0.5398    0.6779    0.7095

    0.7685    0.7278    0.7395    0.5265    0.3678
```

Simulación de un Vector Aleatorio (Matlab)

2 realizaciones iid de un VeA uniforme de dimensión 1x5

```
>> x = rand(2,5)

x =

    0.4519    0.8644    0.5398    0.6779    0.7095

    0.7685    0.7278    0.7395    0.5265    0.3678
```

Realizaciones de un VeA

hola soy cata

También puede verse como **5 realizaciones** iid de un VeA uniforme de dimensión 2x1

```
>> x = rand(2,5)

x =

    0.4519    0.8644    0.5398    0.6779    0.7095

    0.7685    0.7278    0.7395    0.5265    0.3678
```

Simulación de un Vector Aleatorio (Matlab)

2 realizaciones iid de un VeA uniforme de dimensión 1x5

Hola Cola

```
>> x = rand(2,5)

x =

    0.4519    0.8644    0.5398    0.6779    0.7095

    0.7685    0.7278    0.7395    0.5265    0.3678
```

Practicá

Hola SaNa

También puede verse como **5 realizaciones** iid de un VeA uniforme de dimensión 2x1

```
>> x = rand(2,5)

x =

    0.4519    0.8644    0.5398    0.6779    0.7095

    0.7685    0.7278    0.7395    0.5265    0.3678
```

Realizaciones de un VeA

Problema 3

Problema 3

Generar N = 200 muestras para definir los siguientes vectores aleatorios.:

1. Para el vector $\boldsymbol{U} = [U_1\ U_2]^T$, genere dos variables uniformes, $U_1 \sim U(-1;1)$ y $U_2 \sim U(-1;1)$.

2. Para el vector $\boldsymbol{X} = [X_1\ X_2]^T$ genere muestras de las variables X_1 y X_2 a partir de U_1 y U_2, tal que $X_1 = 0.5\,U_1 - 0.3\,U_2$ y $X_2 = 0.7\,U_1 + 0.2\,U_2$.
3. Para el vector $\boldsymbol{Y} = [Y_1\ Y_2]^T$, genere muestras de las variables Y_1 y Y_2 a partir de U_1 y U_2, tal que $Y_1 = 1.2\,U_1 - 0.1\,U_2$ y $Y_2 = U_1 + 0.1\,U_2$.

Para cada caso, hacer un gráfico de dispersión y calcular el coeficiente de correlación.

Nota: definir los límites del gráfico en [-2, 2] para ambos ejes.

Problema 4

(Guía 2, Ejercicio 1.2)

Problema 4

Sea $Y = X + N$, con X y N variables aleatorias independientes.

(a) Demostrar que $f_{Y|X}(y|x) = f_N(y - x)$.

(b) Demostrar que $f_Y(y) = f_X(y) * f_N(y)$.

(c) Si $X \in \{0, 1\}$ es una variable aleatoria Bernoulli con $P(X = 0) = p$ y $\mathbb{P}(X = 1) = 1 - p$, expresar $f_Y(y)$.

(d) Suponiendo que $p = 0.3$ y $N \sim N(0, 0.1)$, grafique (en python/matlab) la curva de $f_Y(y)$.

$$F_{Y|X}(y|x) = \mathbb{P}(Y \leq y | X = x) = \frac{\mathbb{P}(Y \leq y, X = x)}{\mathbb{P}(X = x)} = \frac{\mathbb{P}(X + N \leq x + n, X = x)}{\mathbb{P}(X = x)}$$

$$\frac{\mathbb{P}(x + N \leq x + n)}{\mathbb{P}(X = x)} = \frac{\mathbb{P}(N \leq n)}{\cancel{\mathbb{P}(X = x)}}$$

$Y = X + N \qquad X, N$ Ind.

$$f_{Y|X}(y|x) = \frac{f_{XY}(x,y)}{f_X(x)}$$

$X = Y - N$

$$f_N(n) \cdot f_X(x) = f_{XN}(x,n)$$

$$F_{Y|X}(y|x) = \mathbb{P}(Y \leq y | X = x) = \frac{\mathbb{P}(Y \leq y, X = x)}{\mathbb{P}(X = x)} = \frac{\mathbb{P}(X + N \leq x + n, X = x)}{\mathbb{P}(X = x)}$$

$$\frac{\mathbb{P}(x + N \leq x + n)}{\mathbb{P}(X = x)} = \frac{\mathbb{P}(N \leq n)}{\underbrace{\mathbb{P}(X = x)}_{\neq 0}} \Rightarrow \frac{\int_{-\infty}^{n} f_N(t)\,dt}{\int_{-\infty}^{x} f_X(t)\,dt}$$

Profe en pizarrón

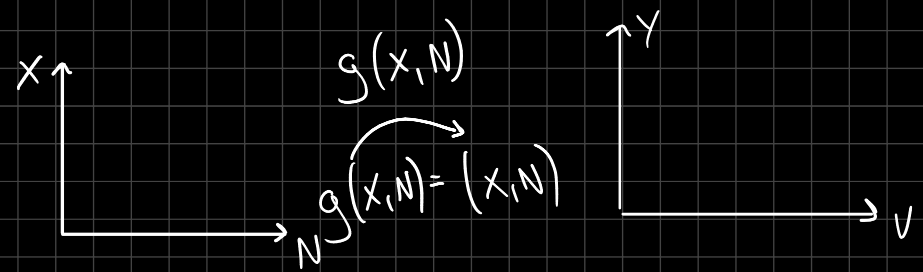
Asumo X discreta

$$F_{Y|X=x} = \mathbb{P}(Y \leq y | X = x) = \mathbb{P}(X + N \leq y | X = x) \overset{\Downarrow}{=} \frac{\mathbb{P}(X + N \leq y, X = x)}{\mathbb{P}(X = x)}$$

$$= \frac{\mathbb{P}(N \leq y - x, X = x)}{\mathbb{P}(X = x)} \overset{\substack{N, X \text{ Ind} \\ \Downarrow}}{=} \frac{\mathbb{P}(N \leq y - x)\mathbb{P}(X = x)}{\mathbb{P}(X = x)} = \mathbb{P}(N \leq y - x)$$

$$= \int_{-\infty}^{y-x} f_N(t)\,dt \Rightarrow \frac{d}{d(y-x)} = f_N(y - x)$$

b) $\begin{cases} Y = X + N \\ V = X \end{cases}$



$$f_{YV}(y,v) = \frac{f_{XN}(x,n)}{|J_g|}\Bigg|_{x = v,\, n = y - v}$$

$$\det(J_g) = \begin{vmatrix} \frac{dy}{dx} & \frac{dy}{dr} \\ \frac{dv}{dx} & \frac{dv}{dr} \end{vmatrix} = \begin{vmatrix} 1 & 1 \\ 1 & 0 \end{vmatrix} = -1$$

$$f_{YV}(y,v) = f_{XN}(v, y-v)$$

$$f_Y(y) = \int_{-\infty}^{\infty} f_{YV}(y,v)\,dv = \int_{-\infty}^{\infty} f_{XN}(v, y-v)\,dv =$$

$$\int_{-\infty}^{\infty} f_X(v) \cdot f_N(y-v)\,dv = f_X(y) * f_N(y)$$

c) $X \sim Ber(p)$

$$\mathbb{P}(Y \leq y) = \mathbb{P}(N+X \leq y) = \mathbb{P}(N+X \leq y, X=0) + \mathbb{P}(X+N \leq y, X=1)$$

$$= \mathbb{P}(N \leq y, X=0) + \mathbb{P}(N \leq y-1, X=1)$$

$$= \mathbb{P}(N \leq y)\,\mathbb{P}(X=0) + \mathbb{P}(N \leq y-1)\,\mathbb{P}(X=1)$$

$$F_Y(y) = F_N(y)p + F_N(y-1)(1-p)$$

$$f_Y(y) = f_N(y)p + f_N(y-1)(1-p)$$