



Método de la transformada inversa

Rodrigo Maranzana

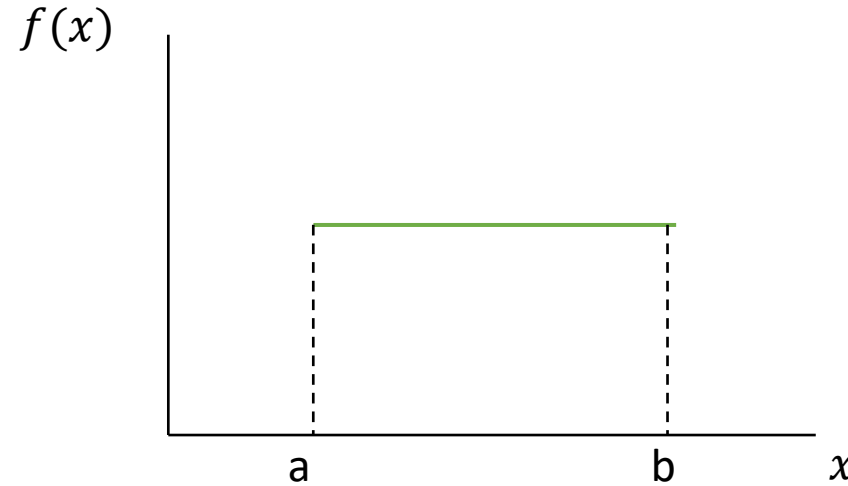
Método de la transformada inversa

Es un método de sampleo de variables aleatorias.

- Ventaja: muy simple de obtener, requiere invertir la función acumulada de densidad de probabilidad.
- Desventaja: es necesario conocer la solución analítica de la acumulada. Y muchos casos no existe.

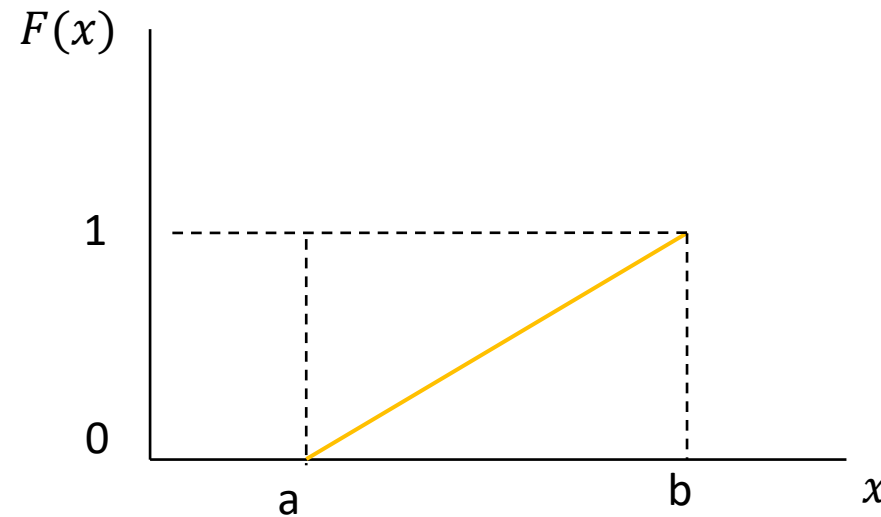
Herramientas: distribución uniforme

Función de densidad:



$$f(x) = \frac{1}{b - a}$$

Función acumulada/
Probabilidad:



$$f(x) = \frac{x - a}{b - a} \quad (\text{proporción})$$

Procedimiento

Dada una función acumulada de probabilidad Uniforme:

$$F_U(X) = U(X)$$

Dada una función acumulada de probabilidad Target:

$$F_T(S) = T(S)$$

- Buscamos **samplear** valores del **dominio de la función target**
- Usamos valores generados con $U(X)$ como la imagen de $T(X)$.

Entonces:

Los valores sampleados se obtienen de: **$S = T^{-1}(U(X))$**

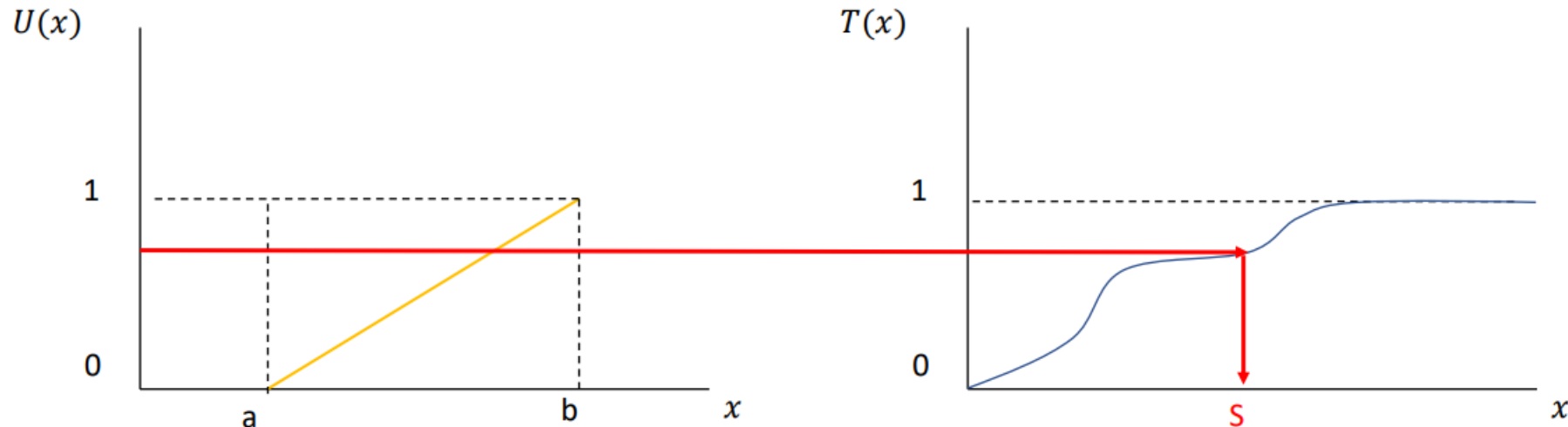
Procedimiento de sampleo de un valor

1) Sampleo de Variable Aleatoria uniforme

$$u = U(X).$$

2) Valor sampleado uniforme como input de la inversa de la acumulada target.

$$s = T^{-1}(U(X))$$



Transformada inversa en función piecewise constant

Una empresa de E-Commerce quiere simular su sistema de pedidos para conocer el comportamiento. En un estudio ABC, surgieron tres grandes rubros que generan pedidos similares en monto y frecuencia: repuestos de automotor, consumibles de computadora, comida y supermercado.

La probabilidad de recibir un pedido de cada rubro es la siguiente:

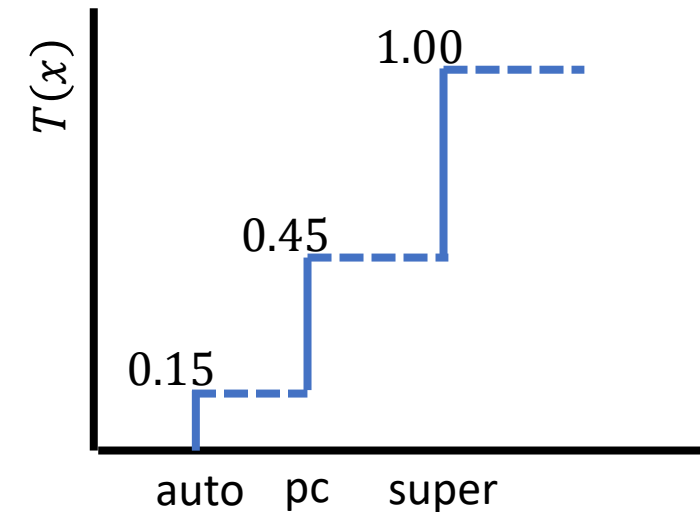
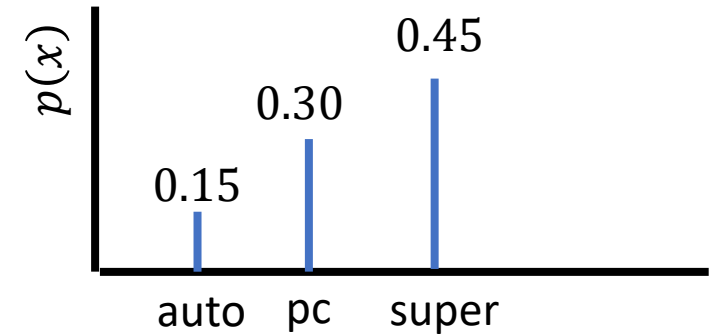
$$p(x) = \begin{cases} 0.15 & \text{si } x = \text{repuestos de automotor} \\ 0.30 & \text{si } x = \text{consumibles computadora} \\ 0.65 & \text{si } x = \text{comida y supermercado} \end{cases}$$

Transformada inversa en función piecewise constant

$$p(x) = \begin{cases} 0.15 & \text{si } x = \text{repuestos de automotor} \\ 0.30 & \text{si } x = \text{consumibles computadora} \\ 0.55 & \text{si } x = \text{comida y supermercado} \end{cases}$$

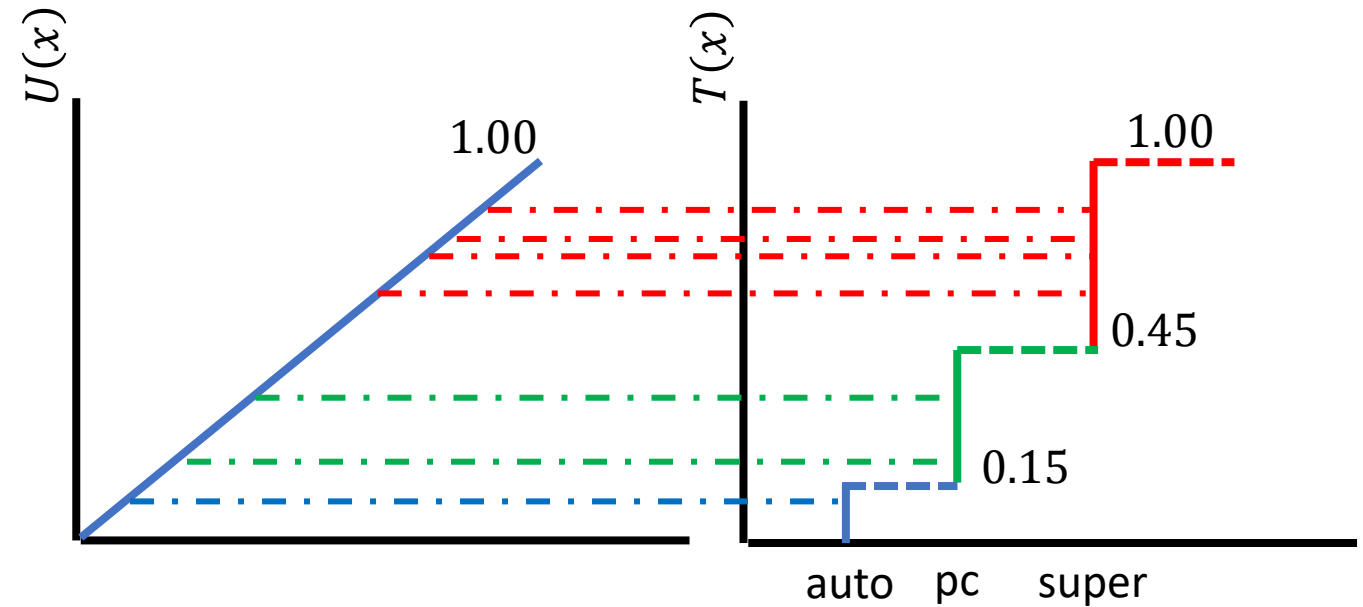
Calculamos la función acumulada de probabilidad:

$$T(x) = \begin{cases} [0.00, 0.15] & \text{si } x = \text{repuestos de automotor} \\ (0.15, 0.45] & \text{si } x = \text{consumibles computadora} \\ (0.45, 1.00] & \text{si } x = \text{comida y supermercado} \end{cases}$$



Transformada inversa en función piecewise constant

Iteración	$U(X)$	$T^{-1}(U(X))$
0	0.11	auto
1	0.52	super
2	0.68	super
3	0.19	pc
4	0.89	super
5	0.32	pc
6	0.76	super



Distribución exponencial

Función de densidad de probabilidad:

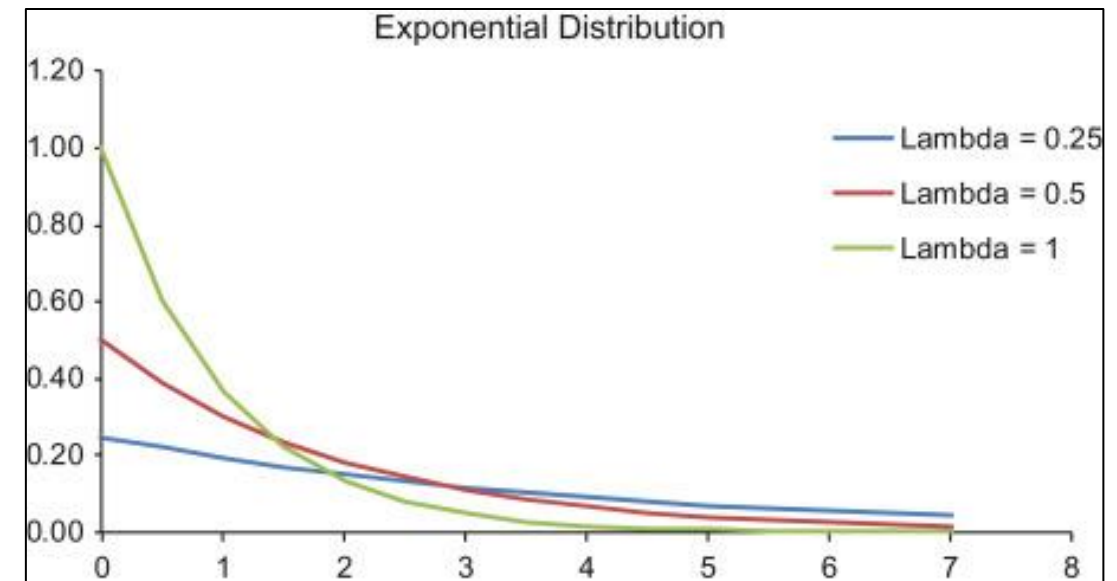
$$f(x) = \lambda e^{-\lambda x}$$

Acumulada de densidad de probabilidad:

$$F(x) = 1 - e^{-\lambda x}$$

Esperanza:

$$\mathbb{E}[x] = \frac{1}{\lambda}$$



<https://www.sciencedirect.com/topics/mathematics/exponential-distribution>

Transformada inversa con distribución exponencial

Partimos de la acumulada:

$$F_{exp}(x) = 1 - e^{-\lambda x}$$

Despejamos x , la variable aleatoria que queremos samplear:

$$x = -\left(\frac{1}{\lambda}\right) \ln(1 - F_{exp}(x))$$

Transformada inversa con distribución exponencial

Despejamos x , la variable aleatoria que queremos samplear:

$$x = -\left(\frac{1}{\lambda}\right) \ln(1 - F_{exp}(x))$$

Llegamos a la inversa. El paso siguiente es generar valores aleatorios para $F_{exp}(x)$.

Reemplazamos por un generador uniforme: $u \sim U(0,1)$

$$x = -\left(\frac{1}{\lambda}\right) \ln(1 - u)$$

Transformada inversa con distribución exponencial

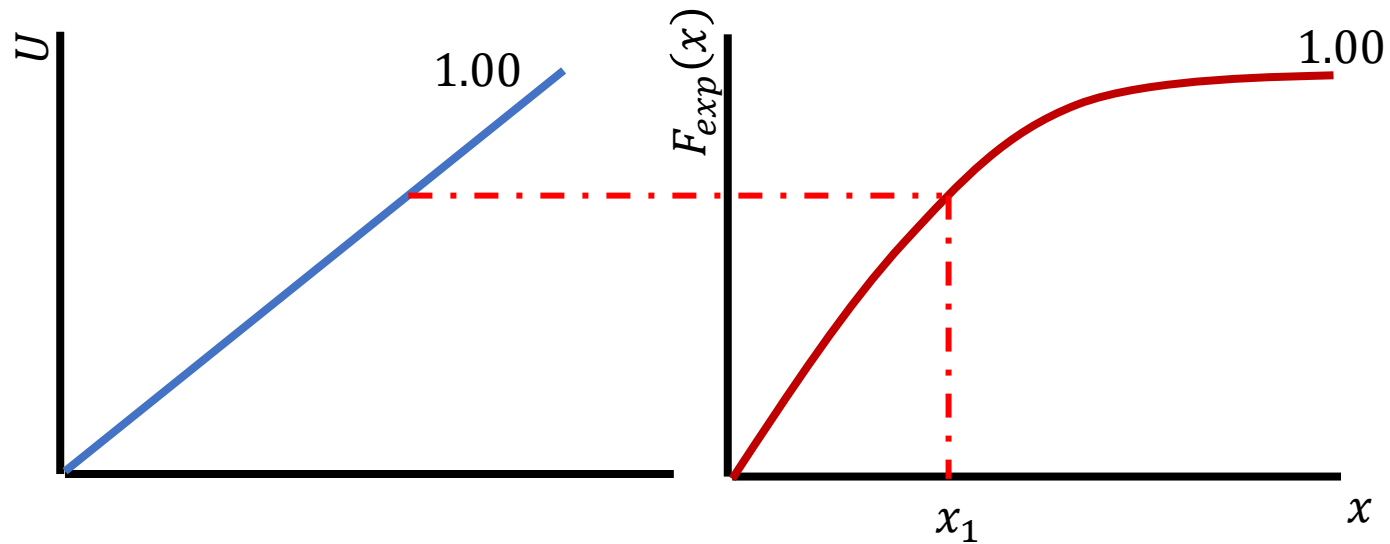
Dado que $(1 - u)$ es el complemento de una variable uniforme, u lo es también. En sampleo es lo mismo.

$$x = -\left(\frac{1}{\lambda}\right) \ln(u)$$

$$u \sim U(0,1)$$

$$x \sim \text{Exp}(\lambda)$$

Representación gráfica



Transformada inversa exponencial con Python

```
import numpy as np

def samplear_transformada_inversa_exp(lam):

    # Sampleo de v.a. uniforme.
    u = np.random.uniform(0, 1)

    # Retornar sampleo de v.a. exponencial.
    return -(1 / lam) * np.log(u)
```

```
samplear_transformada_inversa_exp(0.5)

>> 1.1064004539708514
```

$$x = -\left(\frac{1}{\lambda}\right) \ln(u)$$

$$u \sim U(0,1)$$

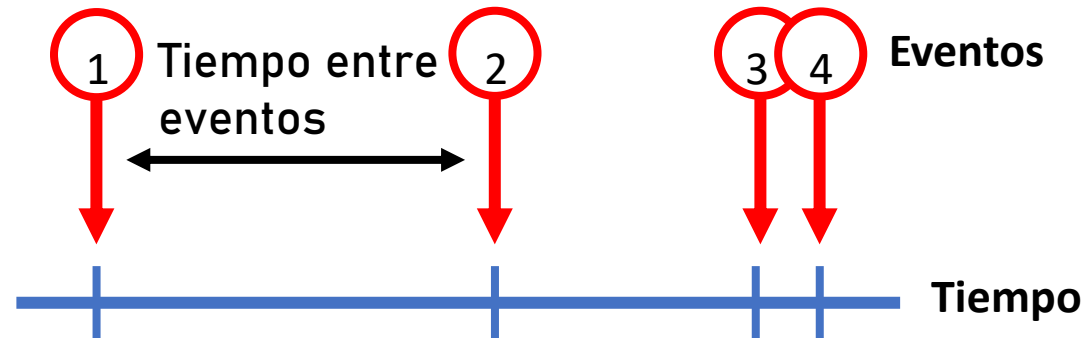
$$x \sim \text{Exp}(\lambda)$$

Usos comunes de sampleo exponencial

Cálculo de “tiempo entre eventos”

Por ejemplo:

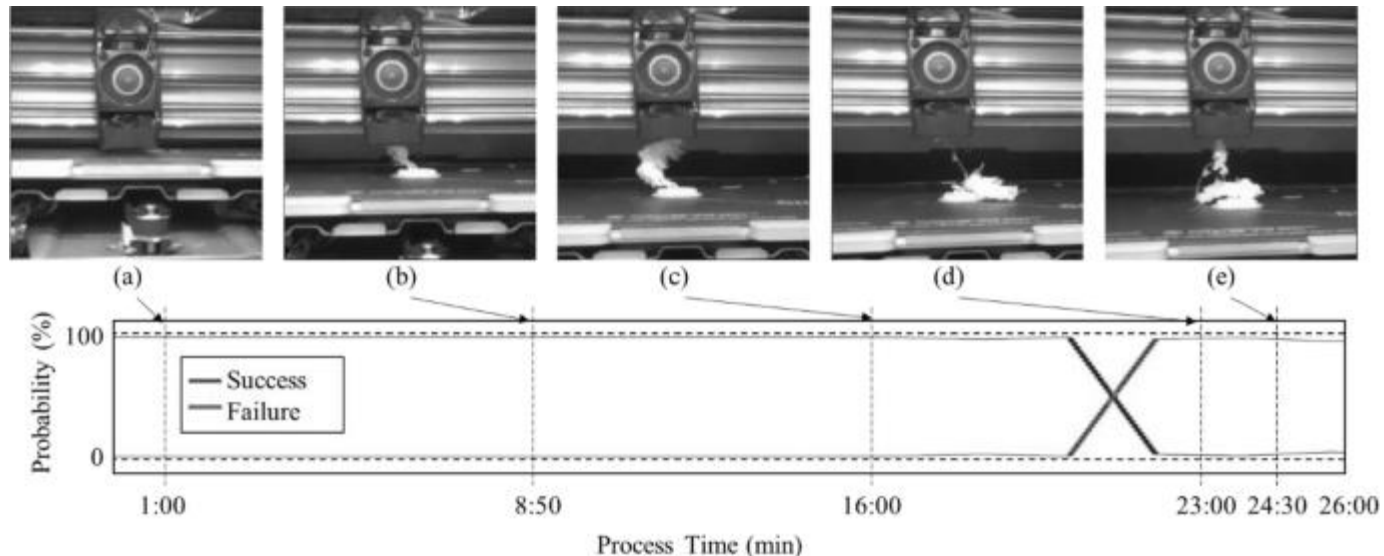
- Tiempo entre arribos de productos, personas.
- Tiempo entre fallas de máquinas.
- Tiempo entre llamados a un callcenter.
- Tiempo entre requests a una API.



Ejemplo transformada inversa exponencial

Una cámara ubicada en punta de línea de extrusión envía eventos de falla de forma remota a un servidor de monitoreo.

Se desea **simular** la performance del sistema completo. Por lo tanto, es necesario generar registros de **eventos aleatorios, que se suponen exponenciales**. Se sabe que la media es de 5 fallas / hora.



Kim et. al (2020), Image-based failure detection for material extrusion process using a convolutional neural network

Ejemplo transformada inversa exponencial

Datos:

$x \sim \text{Exp}(\lambda)$ variable aleatoria exponencial con $\lambda = 5$ fallas/hora

x representa “Tiempo entre fallas”

Unidad temporal $[t] = \text{horas}$

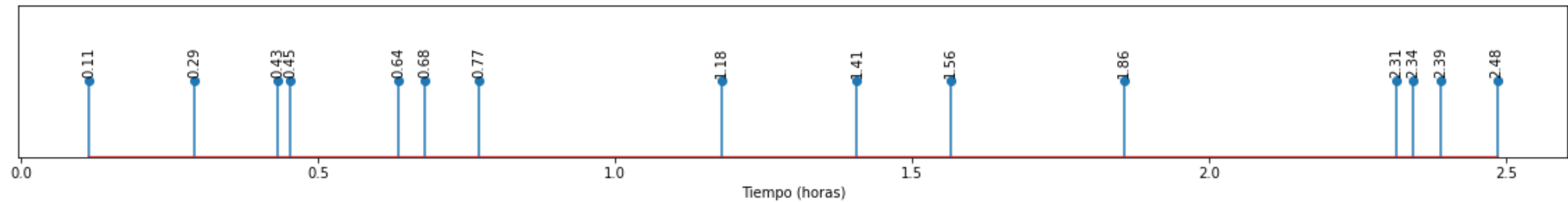
falla	U	Tiempo entre fallas de máquina (hrs)	Cronómetro (tiempo del evento en hrs)
1	0.6715		
2	0.0283		
3	0.6541		
4	0.2395		
5	0.3240		

$$x = -\left(\frac{1}{\lambda}\right) \ln(u)$$

$$u \sim U(0,1)$$

$$x \sim \text{Exp}(\lambda)$$

Ejemplo con 15 valores generados



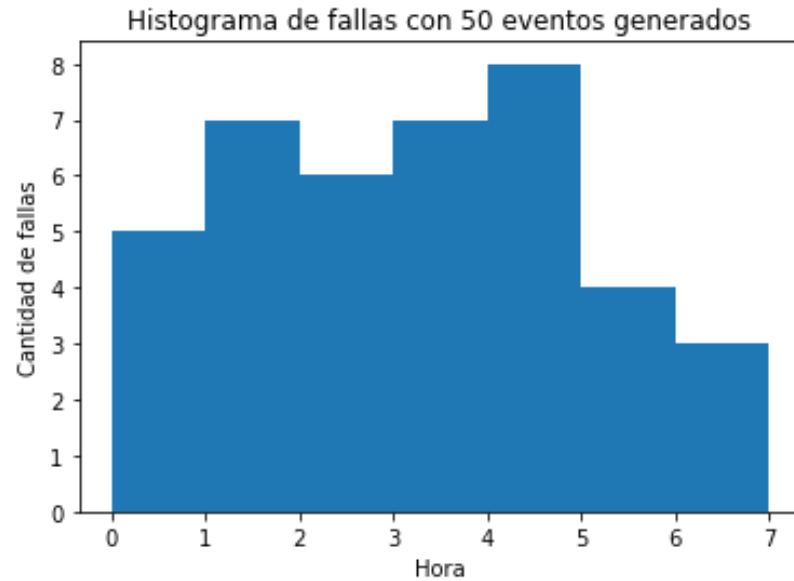
Ejemplo con 10 valores generados

Calculado con python:

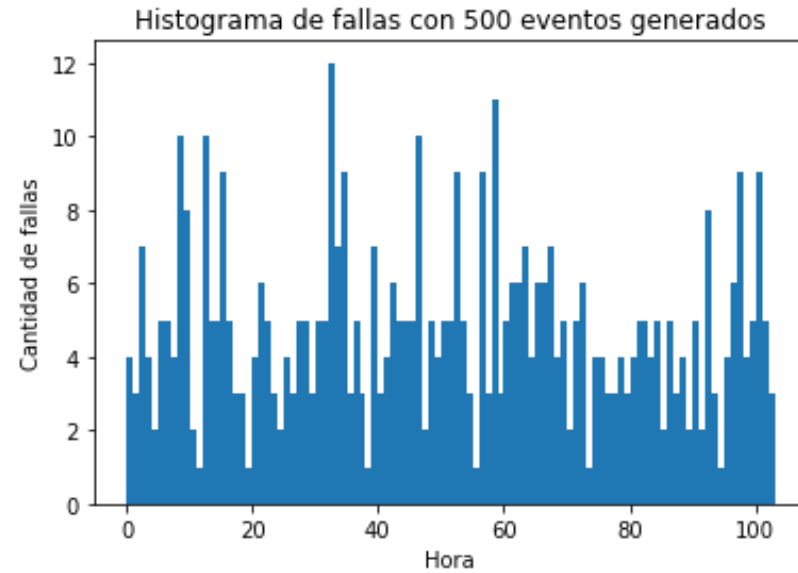
falla	U	Tiempo entre fallas de máquina (hrs)	Cronómetro (tiempo del evento en hrs)	Hora	
1	0.08762491	0.48693799	0.48693799	0	2 fallas
2	0.45027983	0.15957721	0.64651519	0	
3	0.08892822	0.48398516	1.13050035	1	8 fallas
4	0.51890566	0.13120664	1.26170699	1	
5	0.8117183	0.04172038	1.30342738	1	
6	0.65619456	0.08425959	1.38768696	1	
7	0.51140286	0.13411952	1.52180649	1	
8	0.72340748	0.06475652	1.58656301	1	
9	0.81206083	0.04163601	1.62819902	1	
10	0.18952834	0.33264335	1.96084236	1	

Validación de λ de la muestra simulada

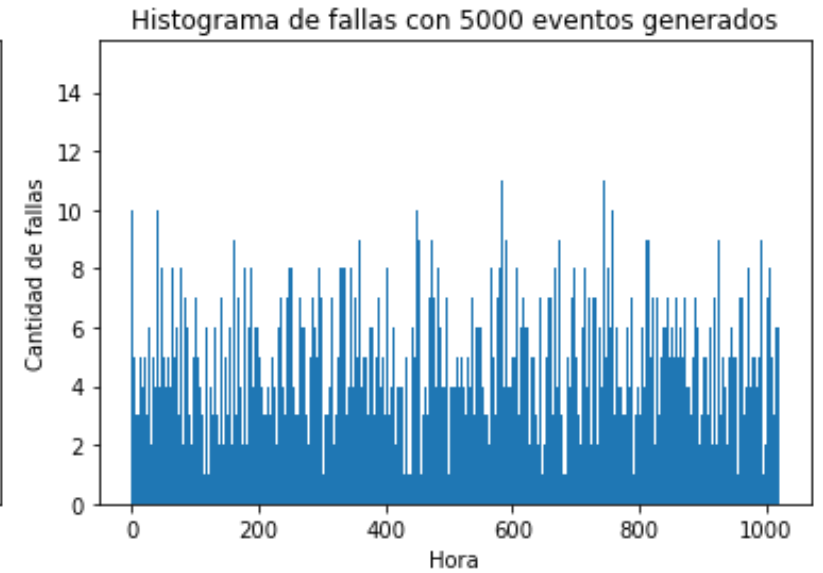
$$\hat{\lambda} = \frac{\sum_i \text{Cantidad eventos}_{\text{hora } i}}{\text{Cantidad de horas simuladas}}$$



$$\hat{\lambda} = 5.71$$

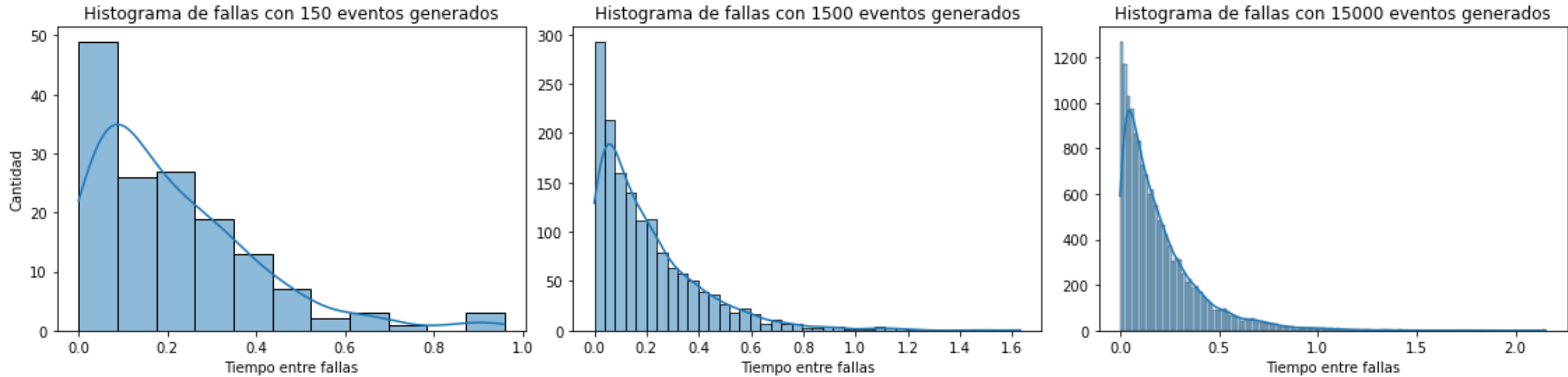


$$\hat{\lambda} = 4.75$$



$$\hat{\lambda} = 4.89$$

Visualización de la exponencial



Arribos y despachos en una línea industrial

Una máquina recibe productos con un tiempo entre arribos que sigue una distribución exponencial de $\lambda = 6 \text{ u/min}$.

El tiempo de procesamiento de la misma también se supone exponencial con una media de $\mu = 7 \text{ u/min}$.

Simulando tiempos de arribos y despachos, se busca conocer el nivel de stock.

Se pide simular un horizonte temporal de 0.60 minutos.

Arribos y despachos en una línea industrial

Datos:

Arribos: $t_A \sim \text{Exp}(\lambda)$ variable aleatoria exponencial con $\lambda = 6 \text{ u/min}$

t_A representa "Tiempo entre arribos de productos"

Despachos: $t_D \sim \text{Exp}(\mu)$ variable aleatoria exponencial con $\mu = 7 \text{ u/min}$

t_D representa "Tiempo entre procesamiento exitoso de productos"

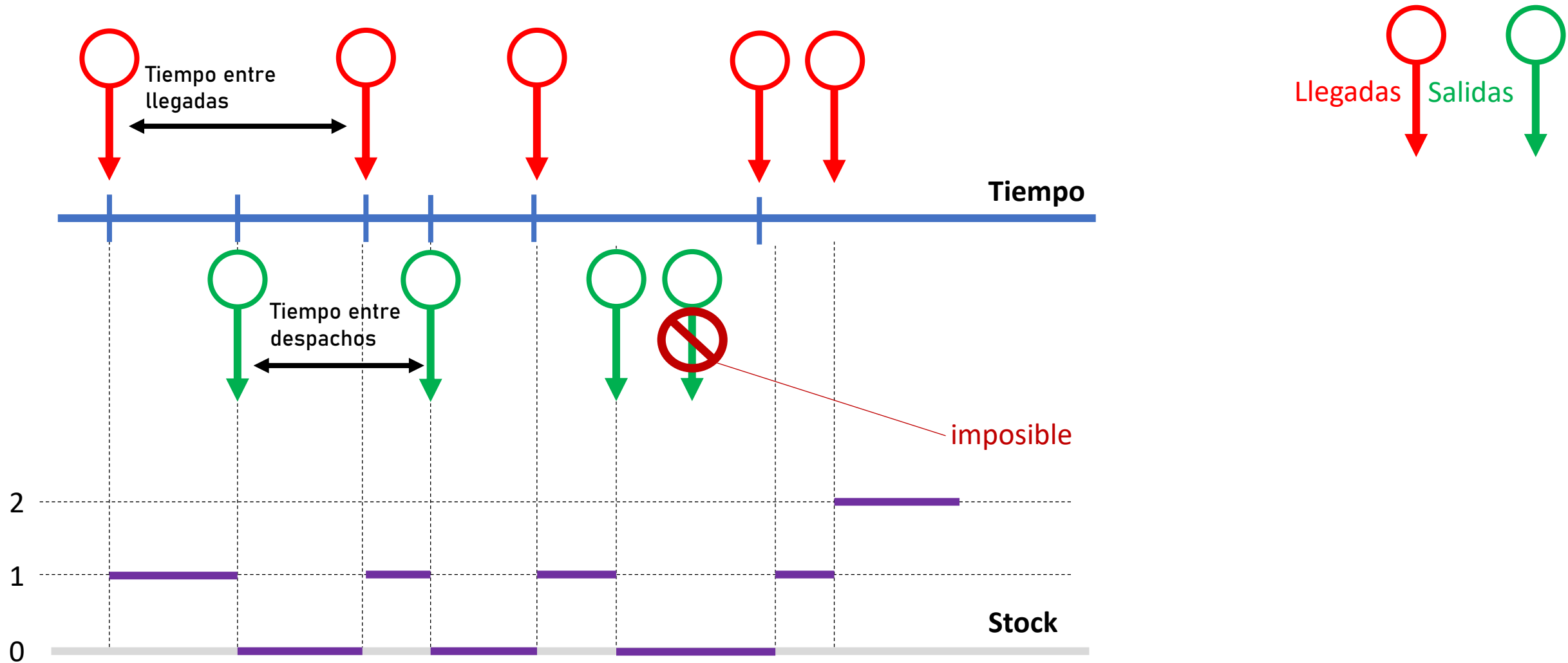
Unidad temporal $[t] = \text{minutos}$

Arribos y despachos de una línea industrial

Simulación simplificada:

- Se simulan ambos eventos por separado, a pesar de ser dependientes, y luego se eliminan inconsistencias en salidas.
- No se busca individualizar registros, sino el acumulado de stock. No es posible conocer producto a producto su historia.

Arribos y despachos de una línea industrial



Arribos y despachos de una línea industrial

1- Generamos valores para cada tipo de evento, con un horizonte temporal de 0.60.

Arribos: $t_A \sim \text{Exp}(\lambda)$ variable aleatoria exponencial con $\lambda = 6 \text{ u/min}$

Despachos: $t_D \sim \text{Exp}(\mu)$ variable aleatoria exponencial con $\mu = 7 \text{ u/min}$

Arribo	U	t entre ev.	Cronómetro
1	0.33912384		
2	0.69487153		
3	0.52248686		
4	0.52882801		
5	0.48778478		

Despacho	U	t entre ev.	Cronómetro
1	0.31468644		
2	0.81579292		
3	0.59140531		
4	0.97468275		
5	0.85049757		
6	0.87347892		
7	0.31670127		
8	0.87871853		

Arribos y despachos de una línea industrial

1- Generamos valores para cada tipo de evento.

Arribos: $t_A \sim \text{Exp}(\lambda)$ variable aleatoria exponencial con $\lambda = 6 \text{ u/min}$

Despachos: $t_D \sim \text{Exp}(\mu)$ variable aleatoria exponencial con $\mu = 7 \text{ u/min}$

Arribo	U	t entre ev.	Cronómetro
1	0.33912384	0.18023166	0.18023166
2	0.69487153	0.06067138	0.24090304
3	0.52248686	0.10819257	0.34909561
4	0.52882801	0.106182	0.45527762
5	0.48778478	0.11964683	0.57492445

Despacho	U	t entre ev.	Cronómetro
1	0.31468644	0.19269643	0.19269643
2	0.81579292	0.03393245	0.22662888
3	0.59140531	0.08754228	0.31417117
4	0.97468275	0.00427387	0.31844504
5	0.85049757	0.02698895	0.34543399
6	0.87347892	0.02254521	0.36797921
7	0.31670127	0.19163272	0.55961192
8	0.87871853	0.02154844	0.58116037

Arribos y despachos de una línea industrial

2- Ordenamos ambos eventos en la misma tabla, indicando el tipo. Eliminamos imposibles

Tipo	Cronómetro	Stock	Decisión
arribo	0.180232	1	
despacho	0.192696	0	
despacho	0.226629	0	Eliminar
arribo	0.240903	1	
despacho	0.314171	0	
despacho	0.318445	0	Eliminar
despacho	0.345434	0	Eliminar
arribo	0.349096	1	
despacho	0.367979	0	
arribo	0.455278	1	
despacho	0.559612	0	
arribo	0.574924	1	
despacho	0.58116	0	

Arribos y despachos de una línea industrial

¿Qué efecto tiene sobre el μ de muestreo el método simplificado?

- El μ del ejercicio es teórico, la **capacidad máxima** de la máquina.
- Al simular y eliminar eventos, obtenemos el μ de trabajo, considerando capacidad ociosa.