Teoría de Filas

Investigación Operativa

Introducción a la Teoría de Filas

¿Qué es la Teoría de Filas?

Rama de la Investigación Operativa que estudia el comportamiento de sistemas en los que entidades (clientes) deben esperar para recibir un servicio.

Ejemplos en la vida real:

- Bancos y supermercados: clientes esperando ser atendidos
- Sistemas de comunicaciones: paquetes de datos esperando ser transmitidos
- Hospitales: pacientes esperando atención médica
- Centros de llamadas: llamadas esperando ser atendidas
- Sistemas de manufactura: trabajos esperando ser procesados

Componentes de un Sistema de Filas

Un sistema de filas típico consta de los siguientes elementos:

1. **Proceso de llegada**: Cómo los clientes llegan al sistema (tasa λ)
2. **Mecanismo de servicio**: Cómo se atienden los clientes (tasa μ)
3. **Disciplina de la cola**: Orden en que se atienden los clientes (FIFO, LIFO, etc.)
4. **Capacidad del sistema**: Número máximo de clientes que puede contener
5. **Número de servidores**: Cantidad de recursos disponibles para atender

Objetivos del Análisis

El análisis de sistemas de filas busca responder preguntas como:

Objetivos del Análisis

El análisis de sistemas de filas busca responder preguntas como:

- ¿Cuánto tiempo esperará un cliente en promedio?

Objetivos del Análisis

El análisis de sistemas de filas busca responder preguntas como:

- ¿Cuánto tiempo esperará un cliente en promedio?
- ¿Cuántos clientes habrá en el sistema en un momento dado?

Objetivos del Análisis

El análisis de sistemas de filas busca responder preguntas como:

- ¿Cuánto tiempo esperará un cliente en promedio?
- ¿Cuántos clientes habrá en el sistema en un momento dado?
- ¿Cuál es la probabilidad de que el sistema esté vacío u ocupado?

Objetivos del Análisis

El análisis de sistemas de filas busca responder preguntas como:

- ¿Cuánto tiempo esperará un cliente en promedio?
- ¿Cuántos clientes habrá en el sistema en un momento dado?
- ¿Cuál es la probabilidad de que el sistema esté vacío u ocupado?
- ¿Cuántos servidores se necesitan para mantener un nivel de servicio aceptable?

Objetivos del Análisis

El análisis de sistemas de filas busca responder preguntas como:

- ¿Cuánto tiempo esperará un cliente en promedio?
- ¿Cuántos clientes habrá en el sistema en un momento dado?
- ¿Cuál es la probabilidad de que el sistema esté vacío u ocupado?
- ¿Cuántos servidores se necesitan para mantener un nivel de servicio aceptable?
- ¿Cómo optimizar el balance entre costos de servicio y costos de espera?

Distribución de Poisson

Uso: Modelar el número de llegadas en un intervalo de tiempo fijo.

Si las llegadas ocurren con una tasa λ, la probabilidad de que ocurran exactamente n llegadas es:

$$P(N = n) = \frac{e^{-\lambda}\lambda^n}{n!}, \quad n = 0, 1, 2, \ldots$$

Propiedades:

- Media: $E[N] = \lambda$
- Varianza: $\mathrm{Var}(N) = \lambda$
- Las llegadas son independientes entre sí
- El proceso es sin memoria (propiedad markoviana)

Distribución Exponencial

Uso: Modelar los tiempos entre llegadas y los tiempos de servicio.

Función de densidad con parámetro λ:

$$f(t) = \lambda e^{-\lambda t}, \quad t \geq 0$$

Función de distribución acumulada:

$$F(t) = P(T \leq t) = 1 - e^{-\lambda t}$$

Propiedades:

- Media: $E[T] = \frac{1}{\lambda}$
- Varianza: $\text{Var}(T) = \frac{1}{\lambda^2}$
- $P(T > s + t \mid T > s) = P(T > t)$ (falta de memoria)
- El mínimo de variables exponenciales independientes es exponencial

Relación entre Poisson y Exponencial

Existe una relación fundamental entre ambas distribuciones:

- Si el número de llegadas sigue una distribución de Poisson con tasa λ, entonces los tiempos entre llegadas siguen una distribución exponencial con parámetro λ
- Recíprocamente, si los tiempos entre llegadas son exponenciales con parámetro λ, el número de llegadas sigue una distribución de Poisson con tasa λ

Distribución General

En algunos modelos (como M/G/1), el tiempo de servicio puede seguir una distribución general (no necesariamente exponencial).

En estos casos se caracteriza por:

- Media: $E[S] = \frac{1}{\mu}$
- Varianza: $\mathrm{Var}(S) = \sigma^2$
- Segundo momento: $E[S^2]$

Procesos de Nacimiento y Muerte

Clase especial de cadenas de Markov en tiempo continuo que modelan sistemas donde la población puede aumentar (nacimientos) o disminuir (muertes) de a una unidad a la vez.

Características:

- **Estados**: $n = 0, 1, 2, \ldots$ (número de clientes en el sistema)
- **Tasas de nacimiento** λ_n: tasa a la que el sistema pasa del estado n al estado $n + 1$
- **Tasas de muerte** μ_n: tasa a la que el sistema pasa del estado n al estado $n - 1$

Diagrama de Transiciones

$$
\begin{tikzcd}
0 \arrow[r, bend left, "\lambda_0"] & 1 \arrow[l, bend left, "\mu_1"] \arrow[r, bend left, "\lambda_1"] & 2 \arrow[l, bend left, "\mu_2"] \arrow[r, bend left, "\lambda_2"] & 3 \arrow[l, bend left, "\mu_3"] \arrow[r, bend left, "\lambda_3"] & \cdots
\end{tikzcd}
$$

Ecuaciones de Balance

En el estado estacionario, la tasa de entrada a cada estado debe igualar la tasa de salida.

Para el estado 0:

$$\lambda_0 \pi_0 = \mu_1 \pi_1$$

Para el estado $n \geq 1$:

$$(\lambda_n + \mu_n)\pi_n = \lambda_{n-1}\pi_{n-1} + \mu_{n+1}\pi_{n+1}$$

Solución para Distribuciones de Estado Estacionario

Resolviendo las ecuaciones de balance, obtenemos:

$$\pi_n = \pi_0 \prod_{i=0}^{n-1} \frac{\lambda_i}{\mu_{i+1}}, \quad n = 1, 2, 3, \ldots$$

donde π_0 se obtiene de la condición de normalización:

$$\sum_{n=0}^{\infty} \pi_n = 1$$

Aplicación a Teoría de Filas

En teoría de filas, los procesos de nacimiento y muerte modelan:

- **Nacimientos**: Llegadas de clientes (tasa λ_n)
- **Muertes**: Salidas de clientes después del servicio (tasa μ_n)

Para sistemas M/M/1 y M/M/1/c:

- $\lambda_n = \lambda$ (tasa de llegada constante)
- $\mu_n = \mu$ (tasa de servicio constante)

Notación de Kendall

Notación estándar para clasificar sistemas de filas:

$$\boxed{A/B/c/K/N/D}$$

- **A**: Distribución de los tiempos entre llegadas
- **B**: Distribución de los tiempos de servicio
- **c**: Número de servidores en paralelo
- **K**: Capacidad máxima del sistema (opcional, por defecto ∞)
- **N**: Tamaño de la población de clientes (opcional, por defecto ∞)
- **D**: Disciplina de la cola (opcional, por defecto FIFO)

Símbolos Comunes para Distribuciones

- **M** (Markoviana): Distribución exponencial (o Poisson para llegadas)
- **D** (Determinística): Tiempos constantes
- **G** (General): Distribución arbitraria
- E_k (Erlang): Distribución Erlang con k fases

Ejemplos Comunes

- **M/M/1**: Llegadas Poisson, servicio exponencial, 1 servidor, capacidad infinita
- **M/M/1/c**: Llegadas Poisson, servicio exponencial, 1 servidor, capacidad máxima c
- **M/M/s**: Llegadas Poisson, servicio exponencial, s servidores en paralelo
- **M/G/1**: Llegadas Poisson, servicio con distribución general, 1 servidor
- **M/D/1**: Llegadas Poisson, servicio determinístico, 1 servidor

Disciplinas de Cola

- **FIFO** (First In First Out): El primero en llegar es el primero en ser atendido
- **LIFO** (Last In First Out): El último en llegar es el primero en ser atendido
- **SIRO** (Service In Random Order): Se atiende en orden aleatorio
- **Priority**: Se atiende según prioridades asignadas

Parámetros

- λ: Tasa de llegadas
- μ: Tasa de servicio
- L: Número de clientes en el sistema
- L_q: Número de clientes en la cola
- L_s: Número de clientes en servicio
- W: Valor medio esperado del tiempo de espera en el sistema
- W_q: Valor medio esperado del tiempo de espera en la cola
- W_s: Valor medio esperado del tiempo de servicio

Fórmula para ρ

Tanto para modelos M/M/1 como M/M/1/c, la fórmula para ρ es la misma:

$$\rho = \frac{\lambda}{\mu}$$

Interpretación: ρ representa el factor de utilización del sistema.

Fórmula para π_0

Para modelos M/M/1:

$$\pi_0 = 1 - \rho$$

Para modelos M/M/1/c:

$$\pi_0 = \frac{1 - \rho}{1 - \rho^{c+1}}$$

Fórmula para L

Para modelos M/M/1:

$$L = \frac{\rho}{1-\rho}$$

Para modelos M/M/1/c:

$$L = \frac{\rho\left[1-(c+1)\rho^{c}+c\rho^{c+1}\right]}{(1-\rho^{c+1})(1-\rho)}$$

Fórmula para L_q

Para modelos M/M/1:

$$L_q = \frac{\rho^2}{1-\rho}$$

Para modelos M/M/1/c:

$$L_q = L - (1 - \pi_0)$$

Fórmula para L_s

Para modelos M/M/1:

$$L_s = \rho$$

Para modelos M/M/1/c:

$$L_s = 1 - \pi_0$$

Fórmula para W

Para modelos M/M/1:

$$W = \frac{1}{\mu - \lambda}$$

Para modelos M/M/1/c:

$$W = \frac{L}{\lambda(1 - \pi_c)}$$

Fórmula para W_q

Para modelos M/M/1:

$$W_q = \frac{\lambda}{\mu(\mu - \lambda)}$$

Para modelos M/M/1/c:

$$W_q = \frac{L_q}{\lambda(1 - \pi_c)}$$

Fórmula para W_s

Tanto para modelos M/M/1 como M/M/1/c:

$$W_s = \frac{1}{\mu}$$

Ejercicio 1 - Consigna

Suponga que en una estación con un solo servidor llegan en promedio 45 clientes por hora. Se tiene capacidad para atender en promedio a 60 clientes por hora. Se sabe que los clientes esperan en promedio 3 minutos en la cola.

Se solicita:

a) Tiempo promedio que un cliente pasa en el sistema

b) Número promedio de clientes en la cola

c) Número promedio de clientes en el sistema en un momento dado

Ejercicio 1 - Datos

Datos:

- $\lambda = 45$ clientes/hora
- $\mu = 60$ clientes/hora
- $W_q = 3$ minutos $= 0{,}05$ horas

Este es un modelo M/M/1.

Ejercicio 1 - Solución (a)

a) Tiempo promedio en el sistema (W):

Sabemos que $W = W_q + W_s$, donde $W_s = \frac{1}{\mu}$:

$$W_s = \frac{1}{60} \text{ horas} = 1 \text{ minuto}$$

Por lo tanto:

$$W = W_q + W_s = 3 + 1 = 4 \text{ minutos} = \frac{1}{15} \text{ horas}$$

Alternativamente, usando la fórmula directa:

$$W = \frac{1}{\mu - \lambda} = \frac{1}{60 - 45} = \frac{1}{15} \text{ horas} = 4 \text{ minutos}$$

Ejercicio 1 - Solución (b)

b) Número promedio de clientes en la cola (L_q):

Usando la Ley de Little: $L_q = \lambda W_q$

$$L_q = 45 \times 0{,}05 = 2{,}25 \text{ clientes}$$

Alternativamente, usando la fórmula directa:

$$\rho = \frac{\lambda}{\mu} = \frac{45}{60} = 0{,}75$$

$$L_q = \frac{\rho^2}{1-\rho} = \frac{(0{,}75)^2}{1-0{,}75} = \frac{0{,}5625}{0{,}25} = 2{,}25 \text{ clientes}$$

Ejercicio 1 - Solución (c)

c) Número promedio de clientes en el sistema (L):

Usando la Ley de Little: $L = \lambda W$

$$L = 45 \times \frac{1}{15} = 3 \text{ clientes}$$

Alternativamente, usando la fórmula directa:

$$L = \frac{\rho}{1-\rho} = \frac{0{,}75}{1-0{,}75} = \frac{0{,}75}{0{,}25} = 3 \text{ clientes}$$

Ejercicio 2 - Consigna

Suponga un restaurante de comidas rápidas al cual llegan en promedio 100 clientes por hora. Se tiene capacidad para atender en promedio a 150 clientes por hora.

Calcule las medidas de desempeño del sistema:

a) ¿Cuál es la probabilidad que el sistema esté ocioso?

b) ¿Cuál es la probabilidad que un cliente llegue y tenga que esperar?

c) ¿Cuál es el número promedio de clientes en la cola?

d) ¿Cuál es la probabilidad que hayan 10 clientes en la cola?

Ejercicio 2 - Datos

Datos:

- $\lambda = 100$ clientes/hora
- $\mu = 150$ clientes/hora

Este es un modelo M/M/1.

Primero calculamos ρ:

$$\rho = \frac{\lambda}{\mu} = \frac{100}{150} = \frac{2}{3} \approx 0{,}667$$

Ejercicio 2 - Solución (a)

a) Probabilidad de que el sistema esté ocioso (π_0):

$$\pi_0 = 1 - \rho = 1 - \frac{2}{3} = \frac{1}{3} \approx 0{,}333$$

La probabilidad de que el sistema esté ocioso es del 33.3 %.

Ejercicio 2 - Solución (b)

b) Probabilidad de que un cliente tenga que esperar:

Un cliente tiene que esperar cuando el sistema está ocupado, es decir, cuando hay al menos 1 cliente en el sistema. Esta probabilidad es:

$$P(\text{esperar}) = 1 - \pi_0 = \rho = \frac{2}{3} \approx 0{,}667$$

La probabilidad de que un cliente tenga que esperar es del 66.7 %.

Ejercicio 2 - Solución (c)

c) Número promedio de clientes en la cola (L_q):

Usando la fórmula para modelo M/M/1:

$$L_q = \frac{\rho^2}{1-\rho} = \frac{(2/3)^2}{1-2/3} = \frac{4/9}{1/3} = \frac{4}{3} \approx 1{,}33 \text{ clientes}$$

Alternativamente, primero calculamos W_q:

$$W_q = \frac{\lambda}{\mu(\mu-\lambda)} = \frac{100}{150(150-100)} = \frac{100}{7500} = \frac{1}{75} \text{ horas} = 0{,}8 \text{ minutos}$$

Y luego aplicamos la Ley de Little:

$$L_q = \lambda W_q = 100 \times \frac{1}{75} = \frac{4}{3} \approx 1{,}33 \text{ clientes}$$

Ejercicio 2 - Solución (d)

d) Probabilidad de que haya 10 clientes en la cola:

En un sistema M/M/1, la probabilidad de que haya exactamente n clientes en el sistema es:

$$\pi_n = (1 - \rho)\rho^n$$

Si hay 10 clientes en la cola, hay 11 clientes en el sistema (10 esperando + 1 siendo atendido):

$$\pi_{11} = (1 - \rho)\rho^{11} = \frac{1}{3} \times \left(\frac{2}{3}\right)^{11} \approx 0{,}333 \times 0{,}00568 \approx 0{,}00189$$

La probabilidad de que haya exactamente 10 clientes en la cola es aproximadamente 0.189 %.

Ejercicio 3 - Consigna

Una cabina de peaje tiene una ventanilla y puede atender como máximo a 4 autos. Los autos llegan a razón de 20 por hora, y el tiempo medio de servicio es de 2 minutos por auto.

Se desea calcular:

a) La probabilidad de que el sistema esté vacío

b) La probabilidad de que haya 4 autos (sistema lleno)

c) El número promedio de autos en el sistema

d) El número promedio de autos en la cola

e) El tiempo promedio total y en cola que pasa un auto en el sistema

Ejercicio 3 - Datos

Datos:

- $\lambda = 20$ autos/hora
- Tiempo medio de servicio $= 2$ minutos $\Rightarrow \mu = 30$ autos/hora
- Capacidad máxima del sistema: $c = 4$ autos

Este es un modelo M/M/1/4.

Primero calculamos ρ:

$$\rho = \frac{\lambda}{\mu} = \frac{20}{30} = \frac{2}{3}$$

Ejercicio 3 - Solución (a)

a) Probabilidad de que el sistema esté vacío (π_0):

Para un modelo M/M/1/c:

$$\pi_0 = \frac{1-\rho}{1-\rho^{c+1}} = \frac{1-\frac{2}{3}}{1-(\frac{2}{3})^5} = \frac{\frac{1}{3}}{1-\frac{32}{243}} = \frac{\frac{1}{3}}{\frac{211}{243}} = \frac{243}{3 \times 211} = \frac{81}{211} \approx 0{,}384$$

La probabilidad de que el sistema esté vacío es aproximadamente 38.4 %.

Ejercicio 3 - Solución (b)

b) Probabilidad de que el sistema esté lleno (π_4):

Para un modelo M/M/1/c, la probabilidad de estado *n* es:

$$\pi_n = \pi_0 \rho^n$$

Por lo tanto:

$$\pi_4 = \pi_0 \rho^4 = \frac{81}{211} \times \left(\frac{2}{3}\right)^4 = \frac{81}{211} \times \frac{16}{81} = \frac{16}{211} \approx 0{,}076$$

La probabilidad de que haya 4 autos (sistema lleno) es aproximadamente 7.6 %.

c) Número promedio de autos en el sistema (L):

Para un modelo M/M/1/c:

$$L = \frac{\rho\left[1 - (c+1)\rho^{c} + c\rho^{c+1}\right]}{(1-\rho^{c+1})(1-\rho)}$$

Sustituyendo con $\rho = \frac{2}{3}$ y $c = 4$:

$$L = \frac{\frac{2}{3}\left[1 - 5\left(\frac{2}{3}\right)^{4} + 4\left(\frac{2}{3}\right)^{5}\right]}{\left(1 - \left(\frac{2}{3}\right)^{5}\right)\left(1 - \frac{2}{3}\right)}$$

Ejercicio 3 - Solución (c) - Parte 2

Continuando con el cálculo:

$$L = \frac{\frac{2}{3}\left[1 - 5 \times \frac{16}{81} + 4 \times \frac{32}{243}\right]}{\frac{211}{243} \times \frac{1}{3}} = \frac{\frac{2}{3}\left[1 - \frac{80}{81} + \frac{128}{243}\right]}{\frac{211}{729}}$$

$$L = \frac{\frac{2}{3}\left[\frac{243-240+128}{243}\right]}{\frac{211}{729}} = \frac{\frac{2}{3} \times \frac{131}{243}}{\frac{211}{729}} = \frac{\frac{262}{729}}{\frac{211}{729}} = \frac{262}{211} \approx 1{,}242$$

El número promedio de autos en el sistema es aproximadamente 1.24 autos.

d) Número promedio de autos en la cola (L_q):

Para un modelo M/M/1/c:

$$L_q = L - (1 - \pi_0) = L - L_s$$

donde $L_s = 1 - \pi_0 = 1 - \frac{81}{211} = \frac{130}{211}$

$$L_q = \frac{262}{211} - \frac{130}{211} = \frac{132}{211} \approx 0{,}626$$

El número promedio de autos en la cola es aproximadamente 0.63 autos.

e) Tiempo promedio total y en cola:

La tasa efectiva de llegadas es:

$$\lambda_{ef} = \lambda(1 - \pi_c) = 20 \times \left(1 - \frac{16}{211}\right) = 20 \times \frac{195}{211} \approx 18{,}48 \text{ autos/hora}$$

Tiempo promedio en el sistema:

$$W = \frac{L}{\lambda_{ef}} = \frac{\frac{262}{211}}{20 \times \frac{195}{211}} = \frac{262}{3900} \approx 0{,}0672 \text{ horas} \approx 4{,}03 \text{ minutos}$$

Tiempo promedio en la cola:

$$W_q = \frac{L_q}{\lambda_{ef}} = \frac{\frac{132}{211}}{20 \times \frac{195}{211}} = \frac{132}{3900} \approx 0{,}0338 \text{ horas} \approx 2{,}03 \text{ minutos}$$

M/G/1

El modelo M/G/1 es un sistema de filas donde:

- **M**: Las llegadas siguen un proceso de Poisson (distribución Markoviana)
- **G**: El tiempo de servicio sigue una distribución general (no necesariamente exponencial)
- **1**: Hay un solo servidor

Este modelo generaliza el M/M/1 al permitir cualquier distribución de tiempo de servicio.

M/G/1 - Parámetros del Sistema

- λ: Tasa de llegadas (llegadas por unidad de tiempo)
- $E[S]$: Tiempo medio de servicio
- $\mu = \frac{1}{E[S]}$: Tasa media de servicio
- $\mathrm{Var}(S)$: Varianza del tiempo de servicio
- $E[S^2]$: Segundo momento del tiempo de servicio
- $\rho = \frac{\lambda}{\mu} = \lambda E[S]$: Factor de utilización

Para estabilidad del sistema, se requiere $\rho < 1$.

Fórmula de Pollaczek-Khinchin

La fórmula de Pollaczek-Khinchin proporciona el número promedio de clientes en la cola:

$$L_q = \frac{\lambda^2 E[S^2]}{2(1-\rho)} = \frac{\rho^2 + \lambda^2 \mathrm{Var}(S)}{2(1-\rho)}$$

También puede expresarse en términos del coeficiente de variación del tiempo de servicio $C_s^2 = \frac{\mathrm{Var}(S)}{E[S]^2}$:

$$L_q = \frac{\rho^2(1 + C_s^2)}{2(1-\rho)}$$

El coeficiente de variación del tiempo de servicio mide la variabilidad relativa del tiempo de servicio respecto de la media.

M/G/1 - Otras Medidas de Desempeño

Tiempo promedio en la cola:

$$W_q = \frac{L_q}{\lambda} = \frac{\lambda E[S^2]}{2(1-\rho)}$$

Número promedio de clientes en el sistema:

$$L = L_q + \rho$$

Tiempo promedio en el sistema:

$$W = W_q + E[S] = \frac{L}{\lambda}$$

M/M/1 (Servicio Exponencial)

Si el servicio es exponencial: $E[S^2] = \frac{2}{\mu^2}$ y $C_s^2 = 1$

$$L_q = \frac{\rho^2}{1 - \rho}$$

que coincide con la fórmula del modelo M/M/1.

M/D/1 (Servicio Determinístico)

Si el servicio es determinístico (constante): $\mathrm{Var}(S) = 0$, $E[S^2] = E[S]^2$ y $C_s^2 = 0$

$$L_q = \frac{\rho^2}{2(1-\rho)}$$

Nótese que para el mismo ρ, el sistema M/D/1 tiene la mitad de clientes en cola que M/M/1, debido a la ausencia de variabilidad en el servicio.

Interpretación

La fórmula de Pollaczek-Khinchin muestra que:

- El número de clientes en cola aumenta con la variabilidad del tiempo de servicio
- Reducir la variabilidad del servicio (manteniendo la media constante) reduce la congestión
- Cuando $C_s^2 = 0$ (servicio determinístico), se minimiza L_q
- Cuando $C_s^2 = 1$ (servicio exponencial), se recupera el modelo M/M/1

Simulación - Ejercicio 1

Un banco recibe en promedio ($\lambda = 4$) clientes por hora (llegadas Poisson) y atiende cada cajero a razón de ($\mu = 2$) clientes por hora (servicio exponencial). El banco puede contratar s cajeros paralelos.

- El costo de espera es de \$10 por cliente-hora en cola
- El costo de servicio es de \$15 por cajero-hora

Se busca determinar el número óptimo de cajeros s que minimice el costo total:

$$C(s) = 10\, L_q(s) + 15\, s$$

donde $L_q(s)$ es el número promedio de clientes en cola en el sistema M/M/s.

Código Python - Parte 1

```
import numpy as np
import matplotlib.pyplot as plt

# Parametros
rate_llegada   = 4    # lambda (llegadas/hora)
rate_atencion  = 2    # mu (servicios por servidor por hora)
s              = 2    # numero de servidores
n_pasos        = 100  # numero de intervalos
costo_espera   = 10   # costo por cliente en cola
costo_servidor = 15   # costo por servidor
```

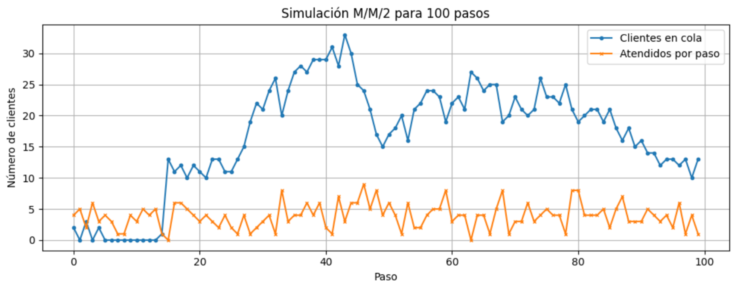
Código Python - Parte 2

```
# Historial
hist_cola      = []
hist_atendidos = []

# Estado inicial
cola = 0

for paso in range(n_pasos):
    # 1) Llegadas: Poisson(lambda * dt)
    llegadas = np.random.poisson(rate_llegada)
    cola += llegadas

    # 2) Atenciones: Poisson(s * mu * dt)
    capacidad = np.random.poisson(s * rate_atencion)
    atendidos = min(cola, capacidad)
    cola -= atendidos

    # 3) Guardar historial
    hist_cola.append(cola)
    hist_atendidos.append(atendidos)
```

Código Python - Parte 3

```
# 4) Costo final
cola_final = hist_cola[-1]
costo_total = costo_espera * cola_final + costo_servidor * s

print(f"Clientes en cola al final: {cola_final}")
print(f"Costo total = {costo_total:.2f}")

# 5) Grafica
plt.figure(figsize=(10,4))
plt.plot(hist_cola, label="Clientes en cola", marker='o',
    markersize=3)
plt.plot(hist_atendidos, label="Atendidos por paso", marker=
    'x', markersize=3)
plt.xlabel("Paso")
plt.ylabel("Numero de clientes")
plt.title(f"Simulacion M/M/{s} para {n_pasos} pasos")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Resultado Simulación 1



- Clientes en cola al final: 13
- Costo total = 10*13 + 15*2 = 160.00

¿Y si tengo diferentes S? - Parte 1

```python
import numpy as np
import matplotlib.pyplot as plt

# Parametros base
rate_llegada   = 4
rate_atencion  = 2
s_max          = 5     # numero maximo de servidores a probar
n_pasos        = 100
costo_espera   = 10
costo_servidor = 15
```

¿Y si tengo diferentes S? - Parte 2

```python
def M_M_s(rate_llegada, rate_atencion, s, n_pasos,
    costo_espera, costo_servidor):
    hist_cola      = []
    hist_atendidos = []
    cola = 0

    for paso in range(n_pasos):
        # 1) Llegadas: Poisson(rate_llegada)
        llegadas = np.random.poisson(rate_llegada)
        cola += llegadas

        # 2) Atenciones: Poisson(s * rate_atencion)
        capacidad = np.random.poisson(s * rate_atencion)
        atendidos = min(cola, capacidad)
        cola -= atendidos
```

¿Y si tengo diferentes S? - Parte 3

```
        # 3) Guardar historial
        hist_cola.append(cola)
        hist_atendidos.append(atendidos)

    media_cola    = np.mean(hist_cola)
    costo_total   = costo_espera * media_cola +
        costo_servidor * s

    return hist_cola, hist_atendidos, costo_total
```

¿Y si tengo diferentes S? - Parte 3

```python
# Simular para s = 1 ... s_max
costos_totales = []
for i in range(1, s_max + 1):
    hist_cola, hist_atendidos, costo_total = M_M_s(
        rate_llegada, rate_atencion, i, n_pasos,
            costo_espera, costo_servidor)
    costos_totales.append(costo_total)
```

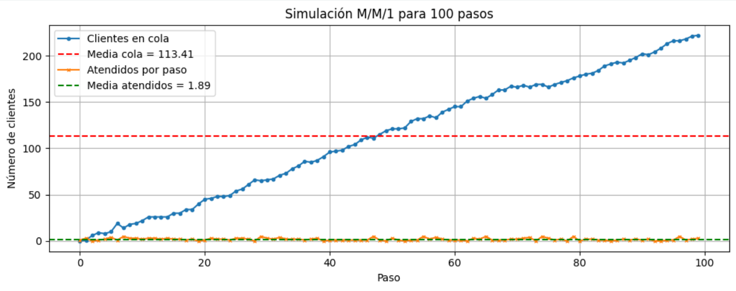
¿Y si tengo diferentes S? - Parte 4

```
# Grafico de evolucion para cada s
plt.figure(figsize=(10,4))
plt.plot(hist_cola, label="Clientes en cola", marker='o'
    , markersize=3)
plt.axhline(np.mean(hist_cola), color='r', linestyle='--
    ',
            label=f"Media cola = {np.mean(hist_cola):.2f
                }")
plt.plot(hist_atendidos, label="Atendidos por paso",
    marker='x', markersize=3)
plt.axhline(np.mean(hist_atendidos), color='g',
    linestyle='--',
            label=f"Media atendidos = {np.mean(
                hist_atendidos):.2f}")
plt.xlabel("Paso")
plt.ylabel("Numero de clientes")
plt.title(f"Simulacion M/M/{i} para {n_pasos} pasos")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

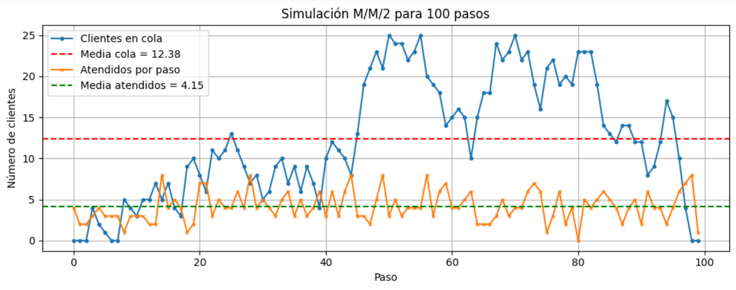
¿Y si tengo diferentes S? - Parte 4

```
# Grafico de costos totales vs numero de servidores
plt.figure(figsize=(8,4))
plt.plot(range(1, s_max+1), costos_totales, '-o')
plt.xlabel("Numero de servidores $s$")
plt.ylabel("Costo total por intervalo")
plt.title("Costo total vs. numero de servidores")
plt.grid(True)
plt.tight_layout()
plt.show()
```

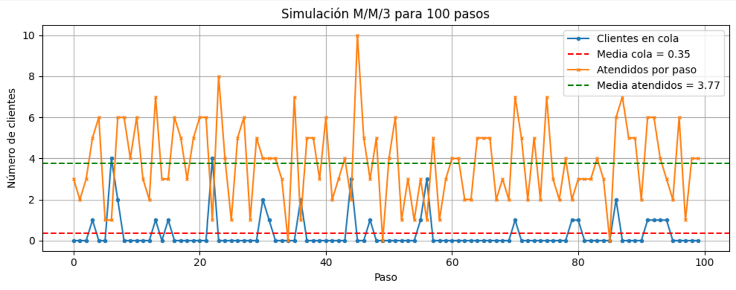
Resultado Simulación 2 - s=1



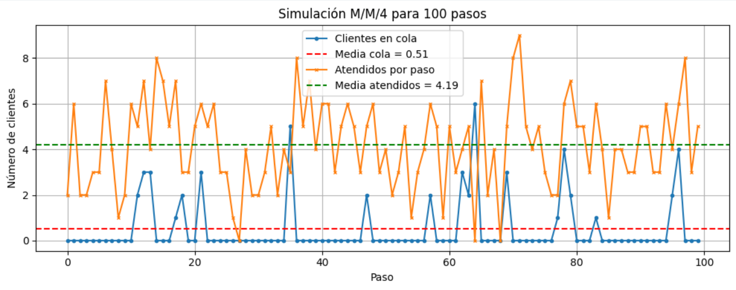
Resultado Simulación 2 - s=2



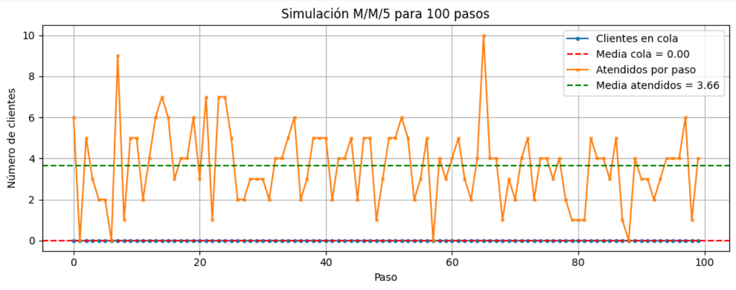
Resultado Simulación 2 - s=3



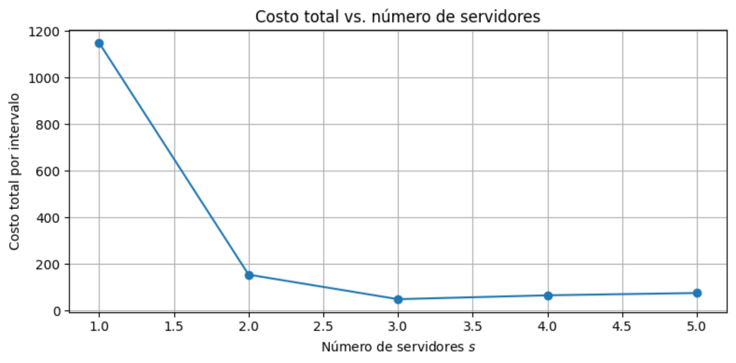
Resultado Simulación 2 - s=4



Resultado Simulación 2 - s=5



Resultado Simulación 2 - Costos



¿Y si hacemos Montecarlo? - Parte 1

```python
import numpy as np
import matplotlib.pyplot as plt

# Parametros base
rate_llegada   = 4
rate_atencion  = 2
s              = 3    # fijamos s=3
n_pasos        = 100
costo_espera   = 10
costo_servidor = 15
n_rep          = 1000  # numero de replicas Monte Carlo
```

¿Y si hacemos Montecarlo? - Parte 2

```
def M_M_s(rate_llegada, rate_atencion, s, n_pasos,
    costo_espera, costo_servidor):
    hist_cola = []
    hist_atendidos = []
    cola = 0

    for _ in range(n_pasos):
        llegadas  = np.random.poisson(rate_llegada)
        capacidad = np.random.poisson(s * rate_atencion)
        cola += llegadas
        atendidos = min(cola, capacidad)
        cola -= atendidos

        hist_cola.append(cola)
        hist_atendidos.append(atendidos)

    media_cola  = np.mean(hist_cola)
    costo_total = costo_espera * media_cola + costo_servidor
         * s
    return np.array(hist_cola), np.array(hist_atendidos),
        costo_total
```

¿Y si hacemos Montecarlo? - Parte 3

```
# Almacenar todos los historicos
all_colas      = np.zeros((n_rep, n_pasos))
all_atendidos  = np.zeros((n_rep, n_pasos))
all_costos     = np.zeros(n_rep)

for i in range(n_rep):
    hc, ha, ctot = M_M_s(rate_llegada, rate_atencion, s,
        n_pasos, costo_espera, costo_servidor)
    all_colas[i]     = hc
    all_atendidos[i] = ha
    all_costos[i]    = ctot

# Calcular estadisticos
mean_cola      = all_colas.mean(axis=0)
p5_cola, p95_cola = np.percentile(all_colas, [5,95], axis=0)

mean_atendidos    = all_atendidos.mean(axis=0)
p5_att, p95_att   = np.percentile(all_atendidos, [5,95],
    axis=0)
```

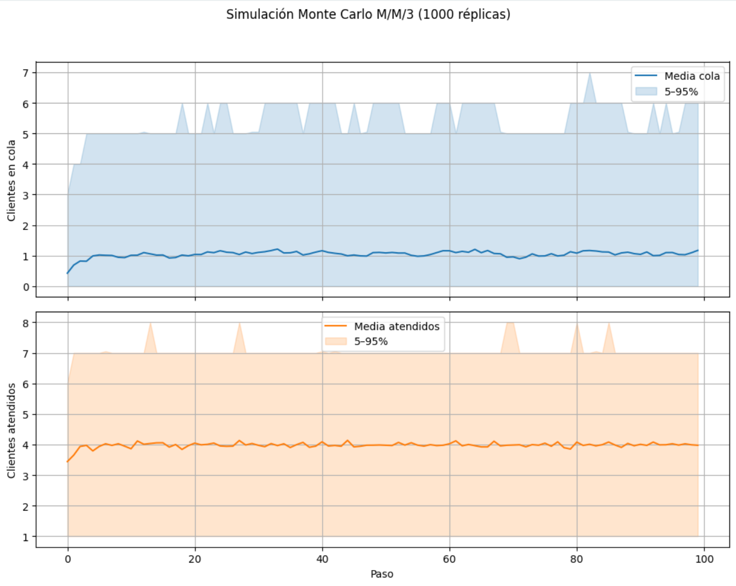
Trayectorias promedio con bandas 5-95 %

```
fig, axes = plt.subplots(2,1, figsize=(10,8), sharex=True)

axes[0].plot(mean_cola, color='C0', label='Media cola')
axes[0].fill_between(range(n_pasos), p5_cola, p95_cola,
    color='C0', alpha=0.2, label='5-95\%')
axes[0].set_ylabel('Clientes en cola')
axes[0].legend()
axes[0].grid(True) # Cola

axes[1].plot(mean_atendidos, color='C1', label='Media
    atendidos')
axes[1].fill_between(range(n_pasos), p5_att, p95_att, color=
    'C1', alpha=0.2, label='5-95\%')
axes[1].set_xlabel('Paso')
axes[1].set_ylabel('Clientes atendidos')
axes[1].legend()
axes[1].grid(True) # Atendidos

plt.suptitle(f'Simulacion Monte Carlo M/M/{s} ({n_rep}
    replicas)')
plt.tight_layout(rect=[0,0,1,0.95])
plt.show()
```

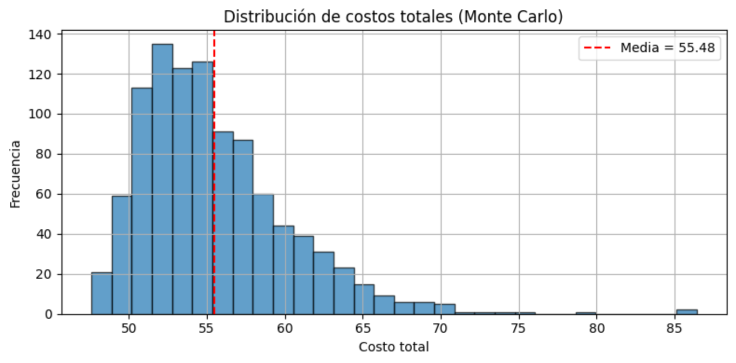
¿Y si hacemos Montecarlo? - Parte 5

```
# 2) Histograma de costos
plt.figure(figsize=(8,4))
plt.hist(all_costos, bins=30, edgecolor='k', alpha=0.7)
plt.axvline(all_costos.mean(), color='r', linestyle='--',
    label=f'Media = {all_costos.mean():.2f}')
plt.xlabel('Costo total')
plt.ylabel('Frecuencia')
plt.title('Distribucion de costos totales (Monte Carlo)')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Resultado Montecarlo 1 - Evolución



Resultado Montecarlo 1 - Costos



¿Y con diferentes S? - Parte 1

```python
import numpy as np
import matplotlib.pyplot as plt

# Parametros base
rate_llegada   = 4
rate_atencion  = 2
s_max          = 5
n_pasos        = 100
costo_espera   = 10
costo_servidor = 15
n_rep          = 1000
```

¿Y con diferentes S? - Parte 2

```
def M_M_s(rate_llegada, rate_atencion, s, n_pasos,
    costo_espera, costo_servidor):
    hist_cola = np.zeros(n_pasos)
    hist_att  = np.zeros(n_pasos)
    cola = 0
    for t in range(n_pasos):
        cola += np.random.poisson(rate_llegada)
        atend = min(cola, np.random.poisson(s *
            rate_atencion))
        cola -= atend
        hist_cola[t] = cola
        hist_att[t]  = atend
    media_cola = hist_cola.mean()
    costo_total = costo_espera * media_cola + costo_servidor
         * s
    return hist_cola, hist_att, costo_total
```

¿Y con diferentes S? - Parte 3

```python
# Arrays para almacenar resultados
cola_means   = np.zeros((s_max, n_pasos))
att_means    = np.zeros((s_max, n_pasos))
costos_means = np.zeros(s_max)

# Bucle sobre s
for s in range(1, s_max + 1):
    all_colas = np.zeros((n_rep, n_pasos))
    all_atts  = np.zeros((n_rep, n_pasos))
    all_costs = np.zeros(n_rep)

    for i in range(n_rep):
        hc, ha, ctot = M_M_s(rate_llegada, rate_atencion, s,
             n_pasos, costo_espera, costo_servidor)
        all_colas[i] = hc
        all_atts[i]  = ha
        all_costs[i] = ctot

    cola_means[s-1]   = all_colas.mean(axis=0)
    att_means[s-1]    = all_atts.mean(axis=0)
    costos_means[s-1] = all_costs.mean()
```

¿Y con diferentes S? - Parte 4

```
# 1) Plot de medias de cola
plt.figure(figsize=(8,4))
for idx in range(s_max):
    plt.plot(cola_means[idx], label=f's={idx+1}')
plt.xlabel('Paso')
plt.ylabel('Media clientes en cola')
plt.title('Evolucion media de la cola para distintos s')
plt.legend()
plt.grid(True)
plt.tight_layout()
```

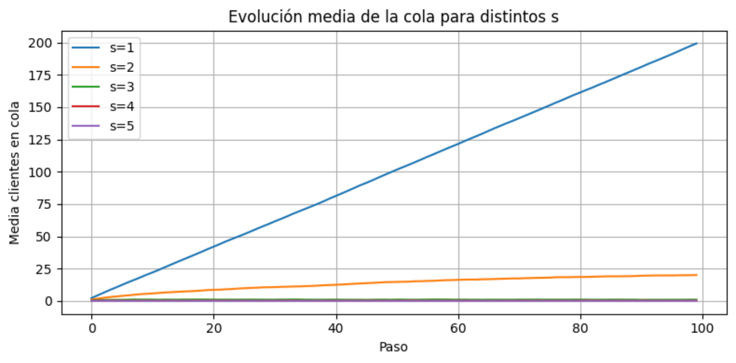
¿Y con diferentes S? - Parte 5

```
# 2) Plot de medias de atendidos
plt.figure(figsize=(8,4))
for idx in range(s_max):
    plt.plot(att_means[idx], label=f's={idx+1}')
plt.xlabel('Paso')
plt.ylabel('Media clientes atendidos')
plt.title('Evolucion media de atendidos para distintos s')
plt.legend()
plt.grid(True)
plt.tight_layout()
```

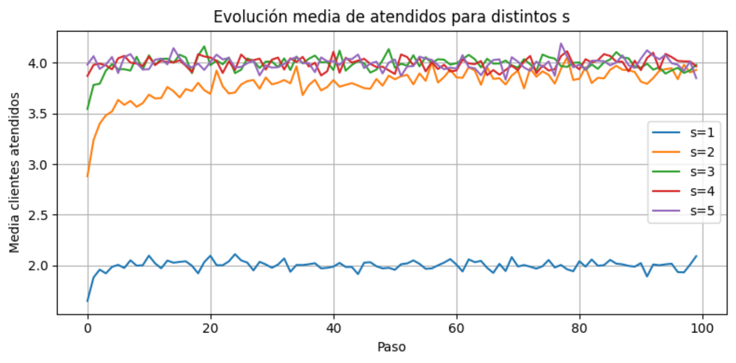
¿Y con diferentes S? - Parte 6

```
# 3) Plot del costo medio final vs s
plt.figure(figsize=(6,4))
plt.plot(range(1, s_max+1), costos_means, '-o')
plt.xlabel('Numero de servidores $s$')
plt.ylabel('Costo medio por intervalo')
plt.title('Costo medio vs numero de servidores')
plt.grid(True)
plt.tight_layout()
plt.show()
```

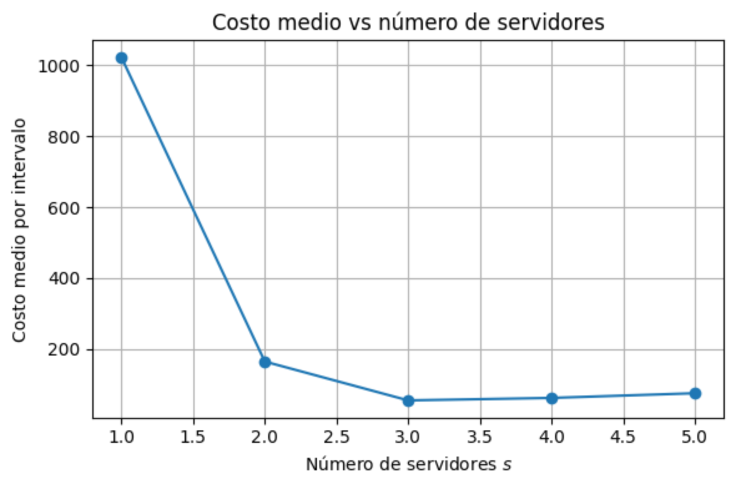
Resultado Montecarlo 2 - Cola



Resultado Montecarlo 2 - Atendidos



Resultado Montecarlo 2 - Costos



¿Dudas?

¿Consultas?