

Guía de Ejercicios Probabilidad y Estadística para I.O.

Investigación Operativa, Universidad de San Andrés

Si encuentran algún error en el documento o hay alguna duda, mandenm  un mail a rodriguezf@udesa.edu.ar y lo revisamos.

1. Ejercicios

1.1. Conversión de Usuarios

Una startup de e-commerce tiene tres canales de adquisición de usuarios: redes sociales (40 % de los usuarios), Google Ads (35 %) y referidos (25 %). Se sabe que la tasa de conversión a compra es del 2 % para usuarios de redes sociales, 3 % para Google Ads y 4 % para referidos. Si se selecciona un usuario al azar, ¿cuál es la probabilidad de que realice una compra?

1.2. A/B Testing

La app de delivery de cierto emprendimiento botánico de alto calibre está probando una nueva funcionalidad de checkout. Se muestran 100 usuarios la nueva versión y se rechaza el cambio si más de 3 usuarios abandonan el proceso de compra. Si la probabilidad de abandono con la nueva versión es 0.02, ¿cuál es la probabilidad de que el test sea exitoso?

1.3. Engagement en Redes Sociales

Una startup de contenido digital publica 20 posts por semana en Instagram. Cada post tiene una probabilidad del 5 % de volverse viral. ¿Cuál es la probabilidad de que exactamente 2 posts se vuelvan virales en una semana?

1.4. Retención de Usuarios

Una app de fintech tiene una tasa de retención del 85 % después del primer mes. Si 15 usuarios nuevos se registran, ¿cuál es la probabilidad de que al menos 12 usuarios permanezcan activos después del primer mes?

1.5. Tráfico Web

Un sitio de e-commerce recibe en promedio 6 visitas por minuto durante las horas pico. ¿Cuál es la probabilidad de que lleguen exactamente 4 visitas en el próximo minuto?

1.6. Soporte Técnico

Un chatbot de una fintech recibe en promedio 10 consultas por hora durante el horario pico. ¿Cuál es la probabilidad de que lleguen más de 12 consultas en una hora?

1.7. Tiempo de Carga

El tiempo de carga de una página web sigue una distribución normal con media 2.5 segundos y desviación estándar 0.4 segundos. ¿Cuál es la probabilidad de que una página cargue en menos de 3 segundos?

1.8. Costos de Adquisición

Los costos de adquisición de usuarios (CAC) de una startup siguen una distribución normal con media \$120 y desviación estándar \$15. Un inversor exige que el CAC no supere los \$140. ¿Cuál es la probabilidad de cumplir con el requisito?

1.9. Plataforma de E-learning

Una plataforma de e-learning tiene una tasa de abandono del 3% en sus cursos. Se analizan 50 estudiantes y se considera que el curso falló si más de 2 estudiantes abandonan. Además, el tiempo de carga de cada video sigue una distribución normal con media 2 segundos y desviación estándar 0.5 segundos. ¿Cuál es la probabilidad de que un curso sea considerado exitoso y que el tiempo total de carga sea menor a 90 segundos?

1.10. Sistema de Streaming

Una plataforma de streaming tiene una tasa de llegada de 8 usuarios por hora (distribución de Poisson) y el tiempo de procesamiento de cada video

sigue una distribución exponencial con media 6 minutos. ¿Cuál es la probabilidad de que lleguen más de 10 usuarios en una hora y que el tiempo de procesamiento sea menor a 5 minutos?

2. Soluciones

2.1. Conversión de Usuarios

Planteo:

- $P(\text{Redes}) = 0,40, P(\text{Google}) = 0,35, P(\text{Referidos}) = 0,25$
- $P(\text{Compra}|\text{Redes}) = 0,02, P(\text{Compra}|\text{Google}) = 0,03,$
 $P(\text{Compra}|\text{Referidos}) = 0,04$
- Queremos: $P(\text{Compra}) = P(\text{Compra}|\text{Redes})P(\text{Redes}) + P(\text{Compra}|\text{Google})P(\text{Google}) + P(\text{Compra}|\text{Referidos})P(\text{Referidos})$

Resolución:

$$P(\text{Compra}) = 0,02 \times 0,40 + 0,03 \times 0,35 + 0,04 \times 0,25 = 0,008 + 0,0105 + 0,01 = 0,0285$$

Código Python:

```
1 # Probabilidades de los canales
2 P_redes = 0.40
3 P_google = 0.35
4 P_referidos = 0.25
5
6 # Probabilidades de conversion por canal
7 P_compra_redes = 0.02
8 P_compra_google = 0.03
9 P_compra_referidos = 0.04
10
11 # Probabilidad total de conversion
12 P_compra = P_compra_redes * P_redes + P_compra_google * P_google + P_compra_referidos * P_referidos
13 print(f"Probabilidad de conversion: {P_compra:.4f}")
```

2.2. A/B Testing

Planteo:

- $n = 100, p = 0,02$
- $X \sim \text{Binomial}(100, 0,02)$
- Queremos: $P(X \leq 3) = 1 - P(X > 3)$

Resolución:

$$P(X \leq 3) = \sum_{k=0}^3 \binom{100}{k} (0,02)^k (0,98)^{100-k}$$

Código Python:

```
1 from scipy.stats import binom
2
3 n = 100
4 p = 0.02
5
6 # Probabilidad de que el test sea exitoso (maximo 3 abandonos
7 )  
7 prob_exitoso = binom.cdf(3, n, p)
8 print(f"Probabilidad de test exitoso: {prob_exitoso:.4f}")
```

2.3. Engagement en Redes Sociales

Planteo:

- $n = 20, p = 0,05$
- $X \sim \text{Binomial}(20, 0,05)$
- Queremos: $P(X = 2)$

Resolución:

$$P(X = 2) = \binom{20}{2} (0,05)^2 (0,95)^{18} = 190 \times 0,0025 \times 0,3972 \approx 0,1887$$

Código Python:

```
1 from scipy.stats import binom
2
3 n = 20
4 p = 0.05
5
6 # Probabilidad de exactamente 2 posts virales
7 prob_exacta = binom.pmf(2, n, p)
8 print(f"Probabilidad de exactamente 2 posts virales: {prob_exacta:.4f}")
```

2.4. Retención de Usuarios

Planteo:

- $n = 15, p = 0,85$
- $X \sim \text{Binomial}(15, 0,85)$
- Queremos: $P(X \geq 12) = 1 - P(X \leq 11)$

Resolución:

$$P(X \geq 12) = 1 - P(X \leq 11) = 1 - \sum_{k=0}^{11} \binom{15}{k} (0,85)^k (0,15)^{15-k}$$

Código Python:

```
1 from scipy.stats import binom
2
3 n = 15
4 p = 0.85
5
6 # Probabilidad de al menos 12 usuarios retenidos
7 prob_al_menos_12 = 1 - binom.cdf(11, n, p)
8 print(f"Probabilidad de al menos 12 usuarios retenidos: {
    prob_al_menos_12:.4f}")
```

2.5. Tráfico Web

Planteo:

- $\lambda = 6$ (tasa promedio de pedidos por hora)
- $X \sim \text{Poisson}(6)$
- Queremos: $P(X = 4)$

Resolución:

$$P(X = 4) = \frac{6^4 e^{-6}}{4!} = \frac{1296 \times 0,0025}{24} \approx 0,1339$$

Código Python:

```

1 from scipy.stats import poisson
2
3 lambd = 6
4
5 # Probabilidad de exactamente 4 visitas
6 prob_exacta = poisson.pmf(4, lambd)
7 print(f"Probabilidad de exactamente 4 visitas: {prob_exacta
      :.4f}")

```

2.6. Soporte Técnico

Planteo:

- $\lambda = 10$ (tasa promedio de llamadas por hora)
- $X \sim \text{Poisson}(10)$
- Queremos: $P(X > 12) = 1 - P(X \leq 12)$

Resolución:

$$P(X > 12) = 1 - P(X \leq 12) = 1 - \sum_{k=0}^{12} \frac{10^k e^{-10}}{k!}$$

Código Python:

```

1 from scipy.stats import poisson
2
3 lambd = 10
4
5 # Probabilidad de mas de 12 consultas
6 prob_mas_12 = 1 - poisson.cdf(12, lambd)
7 print(f"Probabilidad de mas de 12 consultas: {prob_mas_12:.4f}
      ")

```

2.7. Tiempo de Carga

Planteo:

- $\mu = 45, \sigma = 8$
- $X \sim N(45, 8^2)$

- Queremos: $P(X < 50)$

Resolución:

$$P(X < 50) = P\left(Z < \frac{50 - 45}{8}\right) = P(Z < 0,625) \approx 0,7340$$

Código Python:

```

1 from scipy.stats import norm
2
3 mu = 45
4 sigma = 8
5
6 # Probabilidad de cargar en menos de 3 segundos
7 prob_menos_3 = norm.cdf(3, loc=mu, scale=sigma)
8 print(f"Probabilidad de cargar en menos de 3 segundos: {prob_menos_3:.4f}")

```

2.8. Costos de Adquisición

Planteo:

- $\mu = 120, \sigma = 15$
- $X \sim N(120, 15^2)$
- Queremos: $P(X \leq 140)$

Resolución:

$$P(X \leq 140) = P\left(Z \leq \frac{140 - 120}{15}\right) = P(Z \leq 1,33) \approx 0,9082$$

Código Python:

```

1 from scipy.stats import norm
2
3 mu = 120
4 sigma = 15
5
6 # Probabilidad de cumplir requisito (CAC <= 140)
7 prob_cumplir = norm.cdf(140, loc=mu, scale=sigma)
8 print(f"Probabilidad de cumplir requisito: {prob_cumplir:.4f}")

```

2.9. Plataforma de E-learning

Planteo:

- Rechazo del lote: $X \sim \text{Binomial}(50, 0,03)$, $P(X > 2)$
- Tiempo de inspección: $Y \sim N(2, 0,5^2)$, $P(\text{tiempo total} < 90)$
- Eventos independientes: $P(\text{rechazo}) \times P(\text{tiempo} < 90)$

Resolución:

$$P(\text{rechazo}) = 1 - P(X \leq 2) = 1 - \sum_{k=0}^2 \binom{50}{k} (0,03)^k (0,97)^{50-k}$$
$$P(\text{tiempo total} < 90) = P(Y < 1,8) = P\left(Z < \frac{1,8 - 2}{0,5}\right)$$

Código Python:

```
1 from scipy.stats import binom, norm
2
3 # Parametros del curso
4 n = 50
5 p_abandono = 0.03
6
7 # Parametros del tiempo de carga
8 mu_tiempo = 2 # segundos por video
9 sigma_tiempo = 0.5
10 tiempo_total = 90 # segundos
11
12 # Probabilidad de que el curso sea exitoso
13 prob_exitoso = 1 - binom.cdf(2, n, p_abandono)
14
15 # Probabilidad de tiempo total menor a 90 segundos
16 # Tiempo promedio total = 50 * 2 = 100 segundos
17 # Desviacion total = sqrt(50) * 0.5 = 3.54 segundos
18 mu_total = n * mu_tiempo
19 sigma_total = (n**0.5) * sigma_tiempo
20 prob_tiempo = norm.cdf(tiempo_total, loc=mu_total, scale=sigma_total)
21
22 # Probabilidad conjunta (eventos independientes)
23 prob_conjunta = prob_exitoso * prob_tiempo
24
25 print(f"Probabilidad de curso exitoso: {prob_exitoso:.4f}")
```

```

26 print(f"Probabilidad de tiempo < 90 seg: {prob_tiempo:.4f}")
27 print(f"Probabilidad conjunta: {prob_conjunta:.4f}")

```

2.10. Sistema de Streaming

Planteo:

- Llegadas: $X \sim \text{Poisson}(8)$, $P(X > 10)$
- Tiempo de servicio: $Y \sim \text{Exponencial}(\lambda = 1/6)$, $P(Y < 5)$
- Eventos independientes: $P(X > 10) \times P(Y < 5)$

Resolución:

$$P(X > 10) = 1 - P(X \leq 10) = 1 - \sum_{k=0}^{10} \frac{8^k e^{-8}}{k!}$$

$$P(Y < 5) = 1 - e^{-\lambda \cdot 5} = 1 - e^{-5/6}$$

Código Python:

```

1 from scipy.stats import poisson, expon
2
3 # Parametros del sistema
4 lambd_llegadas = 8 # clientes por hora
5 lambd_servicio = 1/6 # tasa de servicio (clientes por minuto
   )
6
7 # Probabilidad de mas de 10 usuarios
8 prob_mas_10 = 1 - poisson.cdf(10, lambd_llegadas)
9
10 # Probabilidad de tiempo de procesamiento menor a 5 minutos
11 prob_procesamiento_rapido = expon.cdf(5, scale=1/
   lambd_servicio)
12
13 # Probabilidad conjunta
14 prob_conjunta = prob_mas_10 * prob_procesamiento_rapido
15
16 print(f"Probabilidad de mas de 10 usuarios: {prob_mas_10:.4f}
   ")
17 print(f"Probabilidad de procesamiento < 5 min: {
   prob_procesamiento_rapido:.4f}")
18 print(f"Probabilidad conjunta: {prob_conjunta:.4f}")

```