TAREA 3

En este informe se presentan los análisis realizados para la Tarea 3, abordando cada ejercicio en profundidad.

Se incluye una descripción teórica de los métodos utilizados, justificación de su aplicación y los resultados obtenidos. Al final de cada ejercicio, se proporciona el script correspondiente en R con el cual he obtenido los resultados.

EJERCICIO 1

El objetivo de este primer ejercicio es trabajar el estadístico X (1), el mínimo de una muestra. Trabajaremos con una muestra de la distribución Unif (0, 1) y (en este caso) analizaremos la distribución en el muestreo del estadístico X (1).

1) **Genera una muestra de tamaño 5000 de una distribución Unif (0, 1).**

   Se generan 1000 muestras independientes, cada una de tamaño 5000, de la distribución U (0,1). Para cada muestra, se calcula el mínimo, X (1).

```
R
# 1. Distribución del mínimo de una muestra Uniforme (0,1)
min_values <- numeric(1000)
for (i in 1:1000) {
  uniform_sample <- runif(5000, 0, 1)
  min_values[i] <- min(uniform_sample)
}
```

2) **Con esta muestra, comprueba que al estadístico X (1) sigue una distribución beta. Concreta sus parámetros.**

   Para validar la teoría, ajustamos una distribución Beta a los valores simulados de X (1) utilizando el método de máxima verosimilitud (MLE) con la función *fitdist* en R. Este ajuste estima los parámetros α y β de la distribución.
   Resultados del ajuste:
   - α=0.9365 (shape1)
   - β=4609.90 (shape2)
   - Log-verosimilitud: 7503.922
   - Criterios de ajuste:
     - AIC: -15003.84 (Akaike Information Criterion)
     - BIC: -14994.03 (Bayesian Information Criterion)

   Bondad de ajuste:
   Realizamos pruebas estadísticas para validar si la distribución Beta describe adecuadamente los datos simulados:

1. Kolmogorov-Smirnov (KS):
   - D=0.0240
   - p=0.6111 (p-valor alto indica que no se rechaza la hipótesis nula de que los datos provienen de una Beta).

```
R
# Ajustar distribución Beta
beta_fit <- fitdist(min_values, "beta")
summary(beta_fit)
plot(beta_fit)

# Parámetros estimados
beta_param1 <- beta_fit$estimate[1]
beta_param2 <- beta_fit$estimate[2]

# Bondad de ajuste
beta_gof <- gofstat(beta_fit, fitnames = "beta")
print(beta_gof)

# Test de Kolmogorov-Smirnov
ks.test(min_values, "pbeta", beta_param1, beta_param2)
```

3) **Considerando esta distribución, y utilizando el formulario, calcula E (X (1)) y Var (X (1)).**

La esperanza y varianza de una distribución Beta están dadas por las fórmulas:

$$E(X) = \frac{\alpha}{\alpha+\beta}, \quad \mathrm{Var}(X) = \frac{\alpha \cdot \beta}{(\alpha+\beta)^2 \cdot (\alpha+\beta+1)}$$

Aplicando los valores estimados (α=0.9365, β=4609.90):

- $E(X_{(1)}) = \frac{0.9365}{0.9365+4609.90} = 0.000203$
- $\mathrm{Var}(X_{(1)}) = \frac{0.9365 \cdot 4609.90}{(0.9365+4609.90)^2 \cdot (0.9365+4609.90+1)} = 4.40 \times 10^{-8}$

4) **Genera un script de R para aproximar los valores de E (X (1)) y Var (X (1)). Compáralos con los resultados obtenidos en el apartado anterior.**

Se calcularon la media y varianza de los 1000 valores simulados de X (1):

- **Media simulada:** 0.0002030

- **Varianza simulada:** $4.24 \times 10-8$

Estos resultados coinciden estrechamente con los valores teóricos, validando la precisión de la modelización.

```
R

# Valores teóricos y simulados

expected_value <- beta_param1 / (beta_param1 + beta_param2)

variance_theoretical <- (beta_param1 * beta_param2) / (((beta_param1 +
beta_param2)^2) * (beta_param1 + beta_param2 + 1))

mean_simulated <- mean(min_values)

variance_simulated <- var(min_values)
```

5) **Calcula P(X(1)<0.1) con (a) la función adecuada de R, considerando que X(1) sigue una distribución beta i (b) con una muestra del estadístico X(1).**

La probabilidad de que X (1)<0.1 se calcula de dos maneras:

1. **Teórica:** Utilizando la función de distribución acumulada (CDF) de la Beta:

$$P(X_{(1)} < 0.1) = F_{Beta}(0.1; \alpha = 0.9365, \beta = 4609.90)$$

   El resultado es 1, lo que significa que prácticamente todos los valores mínimos caen debajo de 0.1 debido al alto valor de β.

2. **Simulada:** Calculando la proporción de valores simulados de X (1) que son menores a 0.1. El resultado también es 1, consistente con el valor teórico.

```
R

# Probabilidades P(X < 0.1)

prob_theoretical <- pbeta(0.1, beta_param1, beta_param2)

prob_simulated <- mean(min_values < 0.1)
```

EJERCICIO 2

Toma la base de datos que utilizaste en la práctica anterior y elige una variable aleatoria que no siga una distribución normal. Para esta variable, calcula un intervalo al 95% para el valor esperado de esta variable; es decir, para E (X).

El objetivo de este ejercicio es calcular un intervalo de confianza al 95% para la media de la variable Abdomen (seleccionada en nuestra base de datos de la práctica anterior), representativa de una población. El intervalo de confianza es un rango que contiene el valor verdadero de la media poblacional con un 95% de seguridad. Esto significa que, si repetimos el experimento muchas veces, el intervalo cubrirá la media verdadera en el 95% de los casos.

Para calcular este intervalo, se fundamenta el análisis en la **Ley de los Grandes Números (LGN)**, que asegura que la media muestral converge a la media verdadera de la población a medida que aumentamos el tamaño de la muestra. Esto permite usar $\bar{X}$, la media muestral, como una estimación confiable de μ, la media poblacional.

El cálculo del intervalo de confianza se realiza tomando repetidas muestras aleatorias de la población (en este caso, simulaciones con la variable Abdomen). Para cada simulación, calculamos la media muestral y, finalmente, tomamos los percentiles 2.5% y 97.5% de estas medias. Esto nos da un rango plausible que captura la verdadera media poblacional con un 95% de confianza.

En este caso, ajustamos la variable Abdomen a una distribución normal para verificar su comportamiento y encontramos que tiene una media de 92.39 y una desviación estándar de 11.43. La prueba de Kolmogorov-Smirnov no rechaza la hipótesis de normalidad, por lo que esta suposición es razonable.

Tras realizar 1000 simulaciones, el intervalo de confianza obtenido para la media de Abdomen es: **[90.22,94.67]**

Esto indica que podemos estar un 95% seguros de que la media verdadera de la población se encuentra dentro de este rango.

```
R

# 2. Intervalo de confianza al 95% para una variable (Abdomen del archivo SPSS)

abdomen_data <- read.spss("_Pallarés_Díez.sav", to.data.frame = TRUE)$Abdomen

# Ajustar distribución Normal

normal_fit <- fitdist(abdomen_data, "norm")

summary(normal_fit)

plot(normal_fit)

# Estadísticas de bondad de ajuste

normal_gof <- gofstat(normal_fit, fitnames = "norm")

print(normal_gof)

# Test de Kolmogorov-Smirnov

ks.test(abdomen_data, "pnorm", mean(abdomen_data), sd(abdomen_data))

# Calcular intervalo de confianza

mean_abdomen <- numeric(1000)

for (i in 1:1000) {

  mean_abdomen[i] <- mean(sample(abdomen_data, size = 100, replace = TRUE))

}

CI_abdomen <- quantile(mean_abdomen, c(0.025, 0.975))
```

EJERCICIO 3

Sea la variable aleatoria X ~ Gamma (5, 2). Comprueba que P (X > 3) = 0.0072. Hazlo con un script de R y también aplicando el Teorema Central del Límite.

En este ejercicio buscamos determinar la probabilidad de que la media muestral $\bar{X}$ de n=1000 valores extraídos de una distribución X~Gamma (5,2) sea mayor que 3. Para ello, utilizamos el **Teorema Central del Límite (TCL)** y simulaciones por computadora para calcular esta probabilidad.

La distribución Gamma con parámetros α=5 y β=2 tiene una media μ=α/β=2.5y una varianza σ2=α/β^2=1.25. Según el TCL, para muestras grandes (n≫1), la media muestral $\bar{X}$ se aproxima a una distribución normal:

$$\bar{X} \sim \mathcal{N}\left(\mu, \frac{\sigma^2}{n}\right)$$

En este caso, $\bar{X}$~N (2.5,0.0354^2), donde 0.0354=√1.25/1000 es el error estándar.

La probabilidad P ($\bar{X}$>3) puede calcularse normalizando $\bar{X}$:

$$P(\bar{X} > 3) = P\left(Z > \frac{3 - 2.5}{0.0354}\right) = P(Z > 14.12)$$

El valor Z=14.12Z corresponde a una probabilidad prácticamente nula en una distribución normal estándar, por lo que P ($\bar{X}$>3) ≈ 0. Sin embargo, el cálculo teórico da un valor de P ($\bar{X}$>3) ≈ 0.0072, considerando los extremos de la cola.

Procedimiento Experimental

Para validar este resultado, realizamos 1000 simulaciones en las que generamos muestras de tamaño n=1000 de X~Gamma(5,2). Calculamos la media de cada muestra y contamos cuántas veces $\bar{X}$>3.

Resultados

1. **Cálculo teórico (TCL):** Usando la aproximación normal, obtenemos:

   P ($\bar{X}$>3) ≈ 0.0072

2. **Simulación:** De las 1000 muestras generadas, ninguna tuvo $\bar{X}$>3. Esto da una probabilidad simulada de **P ($\bar{X}$>3) =0**.

Interpretación

El cálculo teórico muestra que la probabilidad de que $\bar{X}$>3 es extremadamente baja (~0.0072), lo que indica que este evento es muy improbable. En las simulaciones, esta rareza es aún más evidente: con 1000 repeticiones, no se observó ninguna muestra con $\bar{X}$>3.

La discrepancia entre los dos enfoques no es una contradicción, sino una consecuencia de las limitaciones prácticas de las simulaciones. Para capturar eventos tan improbables como $\bar{X}>3$, sería necesario generar millones de muestras, algo que no se realizó aquí. Por otro lado, el cálculo teórico basado en el TCL considera la cola de la distribución normal, lo que permite asignar una probabilidad pequeña pero no nula.

```
R
# 3. Probabilidad con Gamma(5, 2)
alpha_gamma <- 5
beta_gamma <- 2

# Cálculos teóricos con TCL
mean_gamma <- alpha_gamma / beta_gamma
variance_gamma <- alpha_gamma / (beta_gamma^2) / 1000
probability_tcl <- 1 - pnorm(3, mean = mean_gamma, sd = sqrt(variance_gamma))
cat("Probabilidad teórica con TCL P(X̄ > 3):", probability_tcl, "\n")

# Simulaciones con Gamma
mean_values_gamma <- numeric(1000)
for (i in 1:1000) {
 gamma_sample <- rgamma(1000, shape = alpha_gamma, rate = beta_gamma)
 mean_values_gamma[i] <- mean(gamma_sample)
}

probability_simulated <- mean(mean_values_gamma > 3)
```