Maestría en Ciencia de Datos

Introducción a la Estadística Usando Software

Curso 2023

Ejercicios obligatorios

22/05/2023

GRUPO 2

Esteban García

Diego Velasco

Ejercicio 1

Se considera la muestra de 25 datos presentada a continuación:

869.39	177.11	735.20	721.46	850.72
636.72	624.48	539.24	4388.68	1376.25
1656.88	1397.89	523.88	254.30	304.92
263.11	584.39	360.07	139.33	88.86
567.65	948.83	918.23	1328.33	3112.35

Tabla 1. Datos de entrada - Ejercicio 1

Parte a)

En primer lugar, se realiza la carga del paquete de estadística y los datos a utilizar. El código implementado en Octave es mostrado a continuación:

```
### Parte A ####
pkg load statistics
datos =[869.39 177.11 735.20 721.46 850.72 636.72 624.48 539.24 4388.68 1376.25
1656.88 1397.89 523.88 254.30 304.92 263.11 584.39 360.07 139.33 88.86 567.65
948.83 918.23 1328.33 3112.35]
```

Una vez cargados los datos, los mismos son redondeados con la función **round** y se genera el diagrama de tallos y hojas con la función **stemleaf**:

```
led_round= round(datos)
stemleaf(sort(led_round), "Horas Led")
```

El diagrama de tallos y hojas obtenido es presentado a continuación:

8 | 9

13 | 9

17 | 7

25 | 4

26 | 3

30 | 5

36 | 0

52 | 4

53 | 9

56 | 8

58 | 4

62 | 4

63 | 7

72 | 1

73 | 5

85 | 1

86 | 9

91 | 8

94 | 9

132 | 8

137 | 6

139 | 8

165 | 7

311 | 2

438 | 9

Una vez generado el diagrama de tallos y hojas, se genera el gráfico de caja para los datos sobre duración de lámparas LED (en horas). El código implementado en Octave es presentado a continuación:

```
figure
labels = {"Led"}
boxplot(datos,"Labels",labels,"BoxStyle", "filled");
axis([0,2])
title("Horas Duración",'fontsize', 28)
set(gca, 'fontsize', 24)
ylabel('Horas', 'fontsize', 24)
grid
```

El diagrama de caja obtenido es el siguiente:

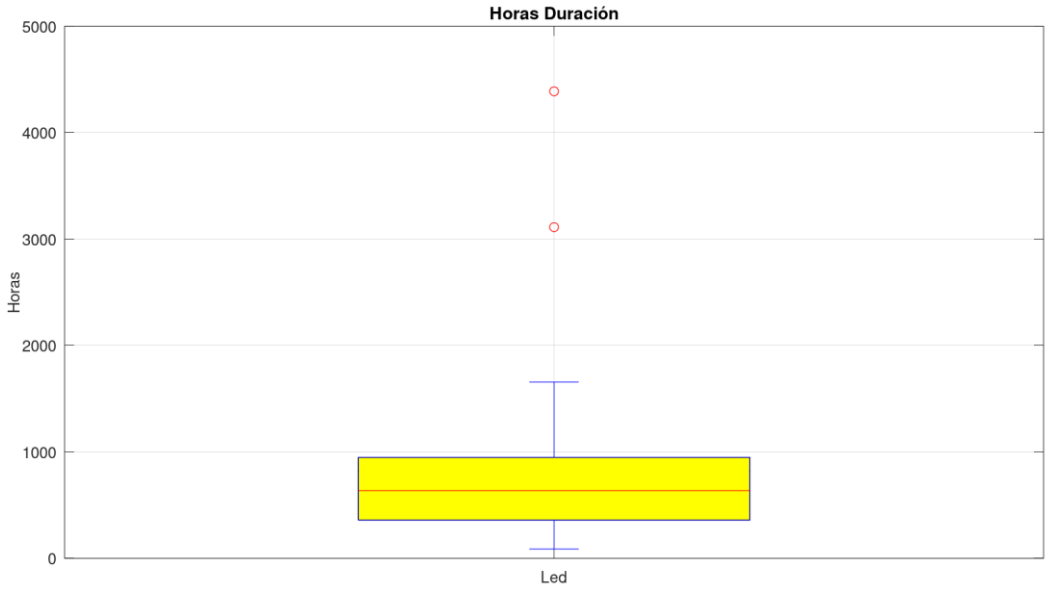

Figura 1. Diagrama de caja - Ejercicio 1

Es posible observar que la mayoría de los datos (más del 75%) se encuentran por debajo de 1000 horas.

Parte b)

Se presume que la distribución de los datos es $\varepsilon(\lambda)$, y se desea realizar el siguiente test bilateral:

$$\begin{cases} H_0: \lambda = \dfrac{1}{1000} \\ H_1: \lambda \neq \dfrac{1}{1000} \end{cases}$$

Para poder definir la región crítica, se utiliza el resultado de que, siendo $X_1, X_2, \dots, X_n$ una muestra de variables aleatorias iid provenientes de una distribución distinta de la normal, la variable aleatoria $Z_n = \sqrt{n}\left(\frac{\overline{X_n}-\mu}{\sigma_n}\right)$ tiende a tener una distribución normal estándar a medida que el tamaño de la muestra crece, siendo n el tamaño de la muestra, $\overline{X_n}$ el promedio de la muestra, μ la esperanza de X y σ_n la desviación estándar de los datos.

De esta manera, se considera $d = \sqrt{n}\left(\frac{\overline{X_n}-\mu}{\sigma_n}\right)$ como el estadístico del test de hipótesis a implementar. Considerando H_0, dado que se presume que los datos están distribuidos de manera exponencial, se tiene que $\mu_0 = \frac{1}{\lambda_0} = 1000$, lo que permite calcular el estadístico de la muestra en cuestión. Esto es implementado en Octave mediante el siguiente código:

```
# TEST BILATERAL
# Parámetros de entrada
n = length(datos);
media = mean(datos);
```

```
sigma_n = std(datos);

lambda0 = 1/1000; # Valor de lambda bajo Ho
mu0 = 1/lambda0; # Esperanza de VA exponencial bajo Ho

d = sqrt(n)*((media-mu0)/sigma_n); # Estadístico del test
alfa = 0.05; # Nivel de significancia del test
```

Para el caso en cuestión, se tiene que $n = 100$, $\overline{X_n} = 934.73$ y $\sigma_n = 961.46$, lo que arroja un valor del estadístico de $d = -0.3394$.

Luego, dado que es esperable que este estadístico siga una distribución normal estándar, es posible calcular la probabilidad de que dicho estadístico tome un valor tanto o más extremo que el obtenido para la muestra en cuestión (p-valor) y así contrastar dicha probabilidad con el nivel de significancia deseado, el cual se toma como 0.05. Siendo F_Φ la función de distribución normal estándar, dicha probabilidad es calculada como $P(Z_n \notin [-d, d]) = 1 - (F_\Phi(-d) - F_\Phi(d))$ ya que el estadístico d es negativo en este caso. Esto es implementado en Octave mediante el siguiente código:

```
# Test
p1 = normcdf(d);
p2 = normcdf(-d);

p_value = 1-(max(p1,p2)-min(p1,p2))

if p_value < alfa
    sprintf('TEST BILATERAL: p-valor = %.2f. Ho es rechazada con alfa = %d',
p_value, alfa)
else
    sprintf('TEST BILATERAL: p-valor = %.2f. Ho no puede ser rechazada con alfa =
%d', p_value, alfa)
end
```

El p-valor obtenido para el caso en cuestión es de 0.7343. Dado que dicho valor es mayor a 0.05, se deduce que no se tiene suficiente evidencia para poder rechazar H_0. Cabe destacar que la conclusión sería la misma para cualquier valor de α en el rango usual (0.01 – 0.10).

Parte c)

En esta parte se realiza el mismo test que en la parte anterior, pero por simulación. Para ello, se simulan 10000 muestras de 25 datos cada una, distribuidas de manera exponencial con parámetro λ=1/1000. Luego, para cada muestra se calcula el estimador de máxima verosimilitud de dicho parámetro. El código implementado en Octave es presentado a continuación.

```
#### Parte C #####
# SIMULACIÓN
```

```
cant_muestras = 10000;

# Generación de datos simulados
datos_sim = exprnd(mu0, cant_muestras, n);
medias = mean(datos_sim,2); # EMV de lambda para distribución exponencial
(promedio)
sigma_nn = std(datos_sim, [], 2);
```

Cabe destacar que se trabaja con μ = 1/λ por comodidad.

Luego, se calcula la distancia L entre el valor de μ estimado a partir de la muestra original con respecto a μ_0 (valor usado para simular las muestras).

Con estos valores se define el intervalo $[\mu_0 - L\,,\ \mu_0 + L]$ y se calcula la cantidad de muestras cuyo estimador cae por fuera del mismo. El cociente entre esta cantidad y la cantidad total de muestras dará la proporción de muestras cuyo estimador toma un valor tanto o más extremo que el estimado a partir de la muestra original, por lo que se espera que se aproxime al p-valor calculado en la parte anterior. Esto es implementado mediante el siguiente código:

```
desvio = mu0 - media;
m1 = media;
m2 = mu0 + desvio;
pseudo_p_val = (size(find(medias<=min(m1,m2)))(1) +
size(find(medias>=max(m1,m2)))(1))/cant_muestras

if pseudo_p_val < alfa
    sprintf('TEST SIMULADO: pseudo p-valor = %.2f. Ho es rechazada con alfa =
%d', pseudo_p_val, alfa)
else
    sprintf('TEST SIMULADO: pseudo p-valor = %.2f. Ho no puede ser rechazada con
alfa = %d', pseudo_p_val, alfa)
end
```

Para el caso en cuestión, la cantidad de muestras cuyo estimador cae por fuera del intervalo es de 7398, lo que da una proporción de 0.7398, valor muy cercano al p-valor de 0.7343 obtenido en la parte anterior. Nuevamente, este valor es mayor al nivel de significancia de 0.05 considerado, por lo que no hay evidencia suficiente para rechazar H_0. De esta manera, se concluye que el valor obtenido del estimador de λ a partir de la muestra original es aceptable. A continuación, se realiza un histograma sobre las medias de cada una de las muestras simuladas, indicando entre líneas verticales de color rojo el intervalo utilizado para el cálculo del p-valor aproximado (“pseudo_p_val” en el código presentado).

```
# Histograma de medias
figure
hist(medias, floor(sqrt(cant_muestras)))
```

```
tit = sprintf('Histograma de Medias');
title(tit, 'fontsize', 35) # plot title
set(gca, 'fontsize', 35) # set axis fontsize
ylabel('Count', 'fontsize', 35, 'fontweight', 'bold') # set ylabel
xlabel('Media', 'fontsize', 35, 'fontweight', 'bold') # set xlabel
grid
hold on
line([m1,m1], ylim, 'LineWidth', 2, 'Color', 'r')
line([m2,m2], ylim, 'LineWidth', 2, 'Color', 'r')
```

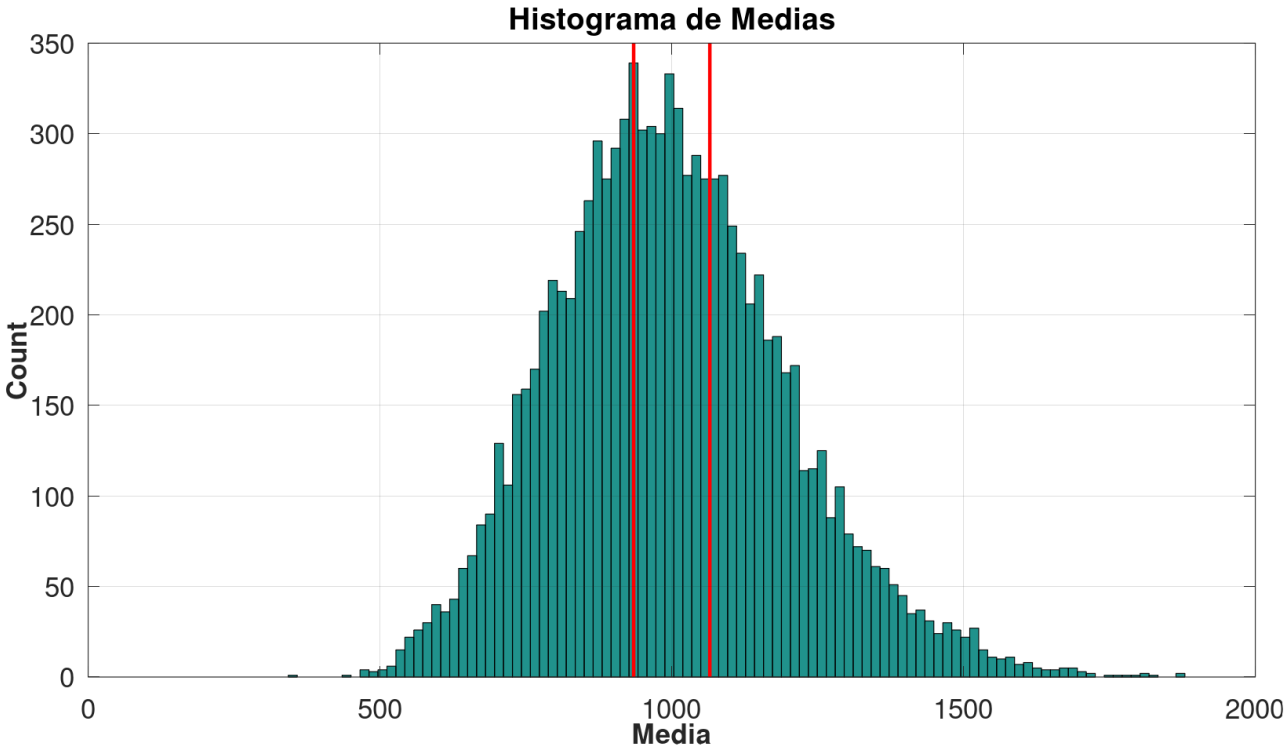

Figura 2. Histograma de medias - Ejercicio 1

Ejercicio 2

Se considera la muestra de 100 datos contenida en el archivo grupo2.mat. Los datos son presentados a continuación en formato de tabla:

3.3886	6.3747	4.0627	3.0814	5.0643	2.6848	3.2637	4.6122	6.8612	8.9373
5.6922	4.1663	5.0513	-2.7141	2.0681	2.5801	8.4102	11.7354	3.0586	7.0123
3.2604	3.8175	2.6682	6.4051	8.2268	7.8622	3.0709	2.5617	4.8077	2.6665
6.6238	4.9443	8.5872	6.3333	2.3689	6.0945	8.8882	4.0163	3.505	8.0935
6.8338	3.9139	-1.391	2.5431	3.2082	2.7563	5.9767	5.9053	4.9613	-0.2036
2.4489	4.6571	3.6211	4.9314	5.7754	1.9263	10.063	6.0157	7.305	7.2098
3.919	5.5957	0.9538	3.0351	-2.1623	9.5778	7.2541	4.8025	3.4117	3.3604
7.2265	5.5613	3.4753	7.5836	2.8903	5.4195	1.2064	5.7064	3.59	6.9578
5.5374	3.1819	3.3963	5.1659	5.372	7.4476	11.6858	6.0031	7.0742	2.6624
4.6466	3.0723	6.1583	6.4682	4.8875	6.222	4.6411	2.7346	6.4033	7.0973

Tabla 2. Datos de entrada - Ejercicio 2

Parte a)

Se plantea la realización de las pruebas de aleatoriedad de Rachas y de Spearman de manera de ver si es razonables afirmar que los datos son independientes e idénticamente distribuidos (i.i.d).

Test de rachas

Se considera una racha como una secuencia de observaciones en orden creciente o decreciente. Una racha creciente se termina cuando el dato en posición *i+1* es menor al dato en posición *i*, mientras que una racha decreciente se termina cuando el dato en posición *i+1* es mayor al dato en posición *i.* Tomando dos datos cualesquiera de un set de datos i.i.d, la probabilidad de que el segundo dato sea mayor o menor que el primero es de 0.5, lo que implica que es esperable que la cantidad de rachas en un set de datos iid esté razonablemente alejado tanto de la unidad (datos ordenados) como del tamaño de la muestra *n*.

Teniendo *n* realizaciones de una variable aleatoria, se tienen *n!* posibles configuraciones en las cuales los datos pueden estar ordenados. Cada posible configuración está asociada con una probabilidad de ocurrencia, la cual es igual al producto de la probabilidad de ocurrencia de cada observación si la muestra es iid. Dado que varias configuraciones distintas tendrán la misma cantidad de rachas, es posible obtener la probabilidad de ocurrencia de cada número de rachas para cada valor de *n,* pudiendo así contrastar la cantidad de rachas de una muestra cualquiera con su probabilidad de ocurrencia y así poder decidir si es razonable afirmar que los datos de dicha muestra son efectivamente iid. La tabla presentada en el curso presenta estas probabilidades en formato $P(R \leq R')$ y $P(R \geq R')$ para cada valor de *n*, siendo *R'* los posibles valores de rachas para el valor de *n.* Dichas probabilidades corresponden a los p-valores del test de rachas.

Para el caso en cuestión, la cantidad de rachas es determinada mediante la función *rachas* implementada en Octave. El código es presentado a continuación:

```
function r = rachas(datos)
    n = length(datos);
    Y = datos(2:n) > datos(1:n-1);
    rachas = (Y(2:n-1) != Y (1:n-2));
    r = sum(rachas) + 1;
```

Esto arroja un total de 65 rachas. Dado que la tabla mencionada cuenta con p-valores para *n≤25*, se utiliza el ajuste de Levene, calculando entonces el p-valor del test a partir del número de rachas *R* como se indica a continuación:

$$Si\ R < \frac{2n-1}{3} \rightarrow p = P(Z \leq z_L), con\ Z \sim N(0,1)\ y\ z_L = \frac{R + 0.5 - \frac{2n-1}{3}}{\sqrt{\frac{(16n-29)}{90}}}$$

$$Si\ R > \frac{2n-1}{3} \rightarrow p = P(Z \geq z_R), con\ Z \sim N(0,1)\ y\ z_R = \frac{R - 0.5 - \frac{2n-1}{3}}{\sqrt{\frac{(16n-29)}{90}}}$$

Esto es implementado en Octave en el archivo .m entregado. La fracción de código en la cual se realiza este ajuste es mostrada a continuación:

```
# Ajuste de Levene
zL_r = (R + 0.5 - (2*n-1)/3)/(sqrt((16*n-29)/90));
zR_r = (R - 0.5 - (2*n-1)/3)/(sqrt((16*n-29)/90));

if R < (2*n-1)/3
    p_value_rachas = normcdf(zL_r);
else
    p_value_rachas = 1 - normcdf(zR_r);
end
```

Dado que $R = 65 < \frac{2n-1}{3} = 66.33$, se utiliza el primer renglón del ajuste planteado, lo que arroja un p-valor de 0.4210. Este valor representa la probabilidad de obtener un número de rachas igual a 65 o más extremo (entendiendo "más extremo" como "menor" en este caso particular). Dado que dicho valor es mayor a 0.05, se considera que la muestra supera el test de rachas. Nuevamente, se destaca que el test se consideraría superado para cualquiera de los valores usuales de α.

Test de correlación de rangos de Spearman

Se consideran las siguientes definiciones:

- Rango: posición que ocuparía el dato X_i si se ordenan los datos de menor a mayor
- Índice: posición que ocupa el dato X_i en la configuración inicial.

El test de correlación de Spearman consiste en hallar el coeficiente de correlación entre ambos vectores, el cual corresponde a la pendiente de la recta que mejor ajusta los puntos de ambos vectores. Este coeficiente es determinado mediante la función *spearman* implementada en Octave. El código es presentado a continuación:

```
function RS = spearman(datos)
    n = length(datos);
    [a b] = sort(datos);
    [indices rangos] = sort(b);

    RS = 1 - 6*(sum((rangos-indices).^2))/(n*(n^2-1));
```

Una vez hallado dicho coeficiente, se determina el p-valor del test en base a valores tabulados y se lo contrasta con el nivel de significancia elegido, pudiendo así decidir si se tiene suficiente evidencia como para rechazar la hipótesis de aleatoriedad. En este caso, dado que el tamaño de la muestra es mayor a 30, se utiliza nuevamente un ajuste, calculando el p-valor de la siguiente manera:

$$Si\ RS < 0 \rightarrow p = P(Z \leq z_L), con\ Z \sim N(0,1)\ y\ z_L = RS\sqrt{n-1}$$

$$Si\ RS > 0 \rightarrow p = P(Z \geq z_R), con\ Z \sim N(0,1)\ y\ z_R = RS\sqrt{n-1}$$

Esto es implementado en Octave en el archivo .m entregado. La fracción de código en la cual se realiza este ajuste es mostrada a continuación:

```
# TEST DE SPEARMAN
RS = spearman(datos);

# Ajuste normal
z_sp = RS*sqrt(n-1);

if RS < 0
    p_value_sp = normcdf(z_sp)
else
    p_value_sp = 1 - normcdf(z_sp)
end
```

Dado que $RS = 0.094905 > 0$, se utiliza el segundo renglón del ajuste planteado, lo que arroja un p-valor de 0.1725. Dado que dicho valor es mayor a 0.05, se considera que la muestra supera el test de correlación de rangos de Spearman. Nuevamente, se destaca que el test se consideraría superado para cualquiera de los valores usuales de α.

Dado que el p valor obtenido para ambos tests es mayor al nivel de significancia considerado, se deduce que no se tiene suficiente evidencia para rechazar la hipótesis de aleatoriedad, de manera que es razonable afirmar que los datos son iid.

Parte b)

El test de Kolmogorov y Smirnov se basa en el teorema de Glivenko-Cantelli, el cual establece que, siendo $X_1, X_2..,X_n$ una sucesión de variables aleatorias iid con distribución *F* y siendo *Fn* la dunción de distribución empírica para la muestra de tamaño *n,* entonces se cumple que:

$$sup_{x \in R} |F_n(x) - F(x)| \to 0$$

con probabilidad 1.

El test consiste en suponer una distribución *F(x)* definida (en este caso, una distribución normal con μ=3 y σ=2), calcular la función de distribución empírica para la muestra en cuestión, y calcular el supremo de la diferencia entre ambas funciones. Luego, se compara dicho valor con los valores límites tabulados, calculados para el nivel de significancia deseado, y se decide si la hipótesis nula es rechazada o no. Cabe destacar que, para cada nivel de significancia α, los valores tabulados corresponden a supremos *KS'* tal que *P(KS>KS') = α.*

Para el caso en cuestión, el supremo mencionado se calcula mediante la función *kolsmi_normal,* implementada en Octave. El código se presenta a continuación:

```
function KS = kolsmi_normal(datos, mu, sigma)
    n = length(datos);
    x = sort([datos datos]);
    Fn = sort([0 (1:n-1)/n (1:n-1)/n 1]);

    F = normcdf((x-mu)/sigma);

    KS = max(abs(Fn - F));
```

Luego, dado que el tamaño de la muestra es mayor a 35, se utiliza el resultado asintótico en función de *n*. La implementación en Octave de la búsqueda del supremo y la comparación contra valores tabulados es presentada a continuación:

```
mu = 3;
sigma = 2;

KS = kolsmi_normal(datos, mu, sigma)

# Ajuste para n>35
if alfa == 0.20
    KS_lim = 1.07/sqrt(n)
elseif alfa == 0.15
    KS_lim = 1.14/sqrt(n)
elseif alfa == 0.10
    KS_lim = 1.22/sqrt(n)
elseif alfa == 0.05
    KS_lim = 1.36/sqrt(n)
elseif alfa == 0.01
    KS_lim = 1.63/sqrt(n)
end
```

El valor del supremo KS vale 0.3599 para el caso en cuestión. Asimismo, dado que se trabaja al 5%, el valor límite de KS vale $\frac{1.36}{\sqrt{n}} = 0.136$. Dado que el valor del supremo es mayor al valor límite, se rechaza la hipótesis nula, de manera que no es razonable afirmar que los datos tienen distribución normal de parámetros μ=3 y σ=2. Cabe destacar que se también se realizó el test con niveles de significancia del 10% y 1% respectivamente, llegando a la misma conclusión. Los resultados son resumidos a continuación.

α	KS	KS_{lim}	Conclusión
0.01	0.3599	0.1630	Se rechaza H_0
0.05	0.3599	0.1360	Se rechaza H_0
0.10	0.3599	0.1220	Se rechaza H_0

Tabla 3. Resumen de resultados para test de Kolmogorov y Smirnov - Ejercicio 2

Parte c)

La prueba de normalidad de Lilliefors utiliza el mismo estadístico que la prueba de Kolmogorov y Smirnov, en el caso en que la media y el desvío de la distribución no son conocidas y por tanto son estimados utilizando toda la muestra. Dado que estimar estos parámetros en base a la muestra tiende a sesgar los resultados hacia no rechazar la hipótesis nula, se utilizan valores límites del estadístico más exigentes para esta prueba. La implementación en Octave de esta prueba es mostrada a continuación. En ella, se estiman los parámetros μ y σ en base a la muestra, se aplica el test de Kolmogorov y Smirnov con dichos parámetros, y se compara el estadístico obtenido con los valores más exigentes obtenidos de la tabla de Lilliefors para el ajuste normal.

```
# PARTE c)
mu = mean(datos);
sigma = std(datos);

KSL = kolsmi_normal(datos, mu, sigma)

# Ajuste para n>35
if alfa == 0.20
    KSL_lim = 0.736/sqrt(n)
elseif alfa == 0.15
    KSL_lim = 0.768/sqrt(n)
elseif alfa == 0.10
    KSL_lim = 0.805/sqrt(n)
elseif alfa == 0.05
    KSL_lim = 0.886/sqrt(n)
elseif alfa == 0.01
    KSL_lim = 1.031/sqrt(n)
end
```

Para el caso en cuestión, se tiene que μ = 4.8787, σ = 2.5625 y KSL = 0.083680. Luego, dado que se trabaja al 5%, el valor límite de KSL vale $\frac{0.886}{\sqrt{n}} = 0.0886$. Dado que el valor del supremo es menor al valor límite,

no se tiene suficiente evidencia para rechazar la hipótesis nula, de manera que se deduce que es razonable afirmar que los datos tienen distribución normal. Cabe destacar que se también se realizó el test con niveles de significancia del 10% y 1% respectivamente. En este caso, si se trabaja al 1% o 5%, se tiene que $KSL<KSL_{lim}$, de manera que no se tiene evidencia suficiente para rechazar H_0. Sin embargo, si se trabaja al 10%, la hipótesis nula debe ser rechazada. Los resultados son resumidos a continuación

α	KSL	KSL_{lim}	Conclusión
0.01	0.083680	0.1031	No hay suficiente evidencia para rechazar H_0
0.05	0.083680	0.0886	No hay suficiente evidencia para rechazar H_0
0.10	0.083680	0.0805	Se rechaza H_0

Tabla 4. Resumen de resultados para test de normalidad de Kolmogorov y Smirnov y Lilliefors - Ejercicio 2