



Instituto Tecnológico
de Buenos Aires

TRABAJO PRÁCTICO N°1

EAC1: ESTADÍSTICA DESCRIPTIVA EN ENFERMEDADES CORONARIAS

93.24 Probabilidad y Estadística

Grupo N°5

Legajo N°	Nombre
61428	Kevin Amiel Wahle
61430	Francisco Basili
61431	Nicolás Bustelo
61433	Sergio Andrés Peralta
61463	Bautista Schneeberger

12/04/2022

Índice

A. Primeras impresiones	2
B. Características de los datos	2
C. Boxplots	2
D. Polígonos de frecuencia	3
E. Gráfico de dispersión (sbp, ldl)	3
F. Enfermedad cardíaca coronaria	4
G. División por grupos A, B y C	5
H. Boxplots de los grupos	5
I. Histograma de los grupos	6
J. Análisis final de los grupos	7
K. Anexo	8
K.1. Código del ejercicio A	8
K.2. Código del ejercicio B	9
K.3. Código del ejercicio C	10
K.4. Código del ejercicio D	11
K.5. Código del ejercicio E	12
K.6. Código del ejercicio F	13
K.7. Código del ejercicio G	14
K.8. Código del ejercicio H	15
K.9. Código del ejercicio I	16
K.10. Código del ejercicio J	17

A. Primeras impresiones

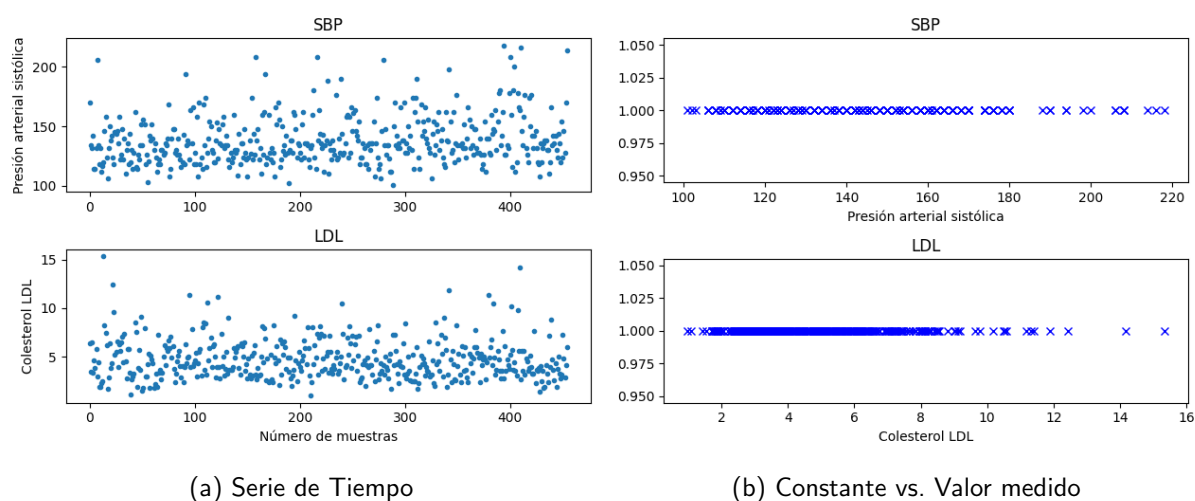


Figura 1: Representación de los valores SBP y LDL

En cuanto a las primeras impresiones podemos notar una similitud en los gráficos de la presión y el colesterol, teniendo este último una dispersión menor como se puede apreciar a simple vista en todos los gráficos. En particular, se puede notar una ligera asimetría positiva gracias a [1b](#) en el cual se compara una constante contra los valores medidos.

El código del programa se encuentra en el anexo apartado [K.1](#).

B. Características de los datos

El código del programa se encuentra en el anexo apartado [K.2](#).

Aspecto	SBP	LDL
Mínimo Valor	101	0.98
Máximo Valor	218	15.33
Media	138.34	4.73
Mediana	134.0	4.33
Desvío Estandar	20.43	2.05
Cuartil 1	124.0	3.27
Cuartil 3	148.0	5.8
Rango Interquartílico	24.0	2.52
Coef. Simetría	0.09	0.28
Coef. de Kurtosis	-1.15	-0.12

C. Boxplots

En cuanto a la representación de los datos en boxplots, podemos ver una marcada presencia de outliers por encima del límite de Tuckey superior. En particular hay un 3.3 % de outliers en el gráfico de SBP y un 2.86 % en el gráfico de LDL.

El código del programa se encuentra en el anexo apartado [K.3](#).

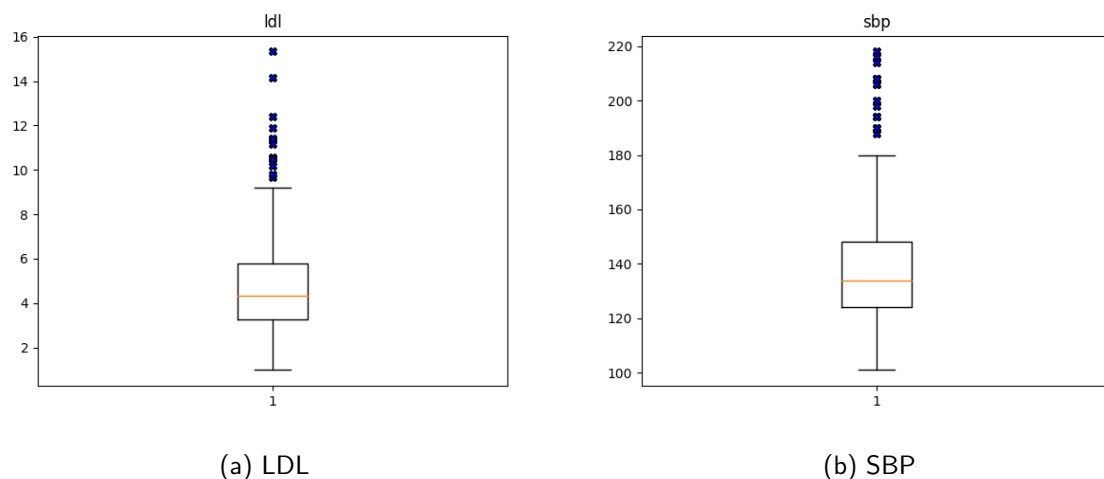


Figura 2: Representación de los valores SBP y LDL con Boxplots.

D. Polígonos de frecuencia

El código del programa se encuentra en el anexo apartado [K.4](#).

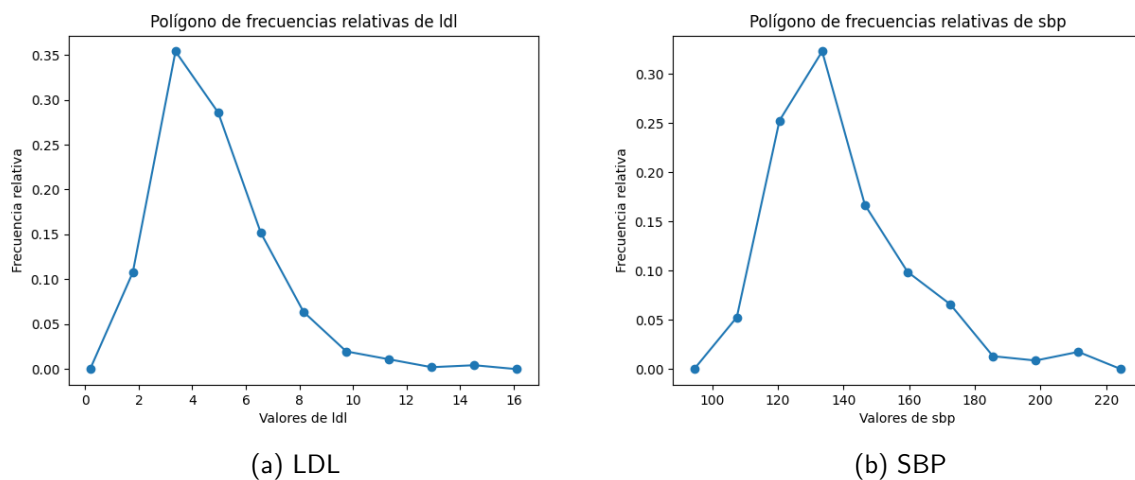


Figura 3: Representación de los valores SBP y LDL con polígonos de frecuencias relativas.

E. Gráfico de dispersión (sbp, ldl)

En el gráfico de la figura 4 podemos observar la relación de la presión arterial sistólica y el colesterol de baja densidad de los pacientes encuestados. En particular se indicó en rojo los valores para los cuales el valor $sbp = 130$ y $ldl = 8$. Debajo de esos valores nos encontramos con una zona a la cual muchos médicos consideran adecuada o saludable. De todas las personas presentes en el estudio, el 33.85 % se encuentran en esa zona.

El código del programa se encuentra en el anexo apartado [K.5](#).

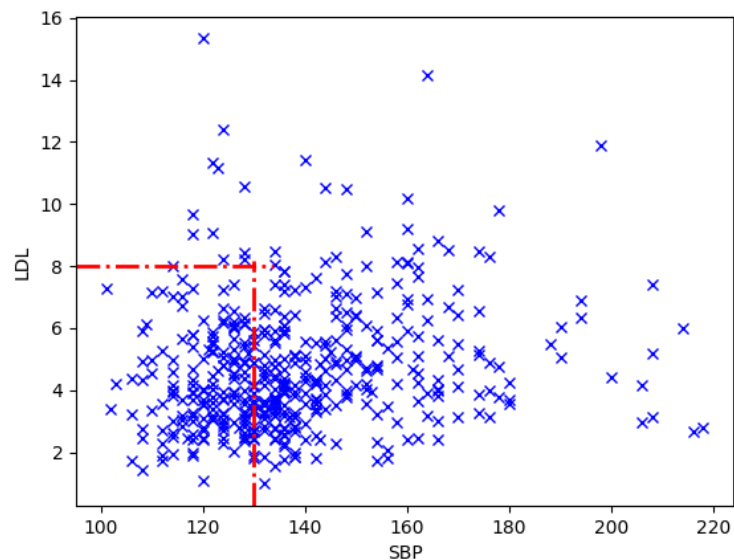


Figura 4: SBP vs. LDL

F. Enfermedad cardíaca coronaria

A continuación se puede observar los datos según si las personas tienen enfermedad cardíaca coronaria (*chd*) o no.

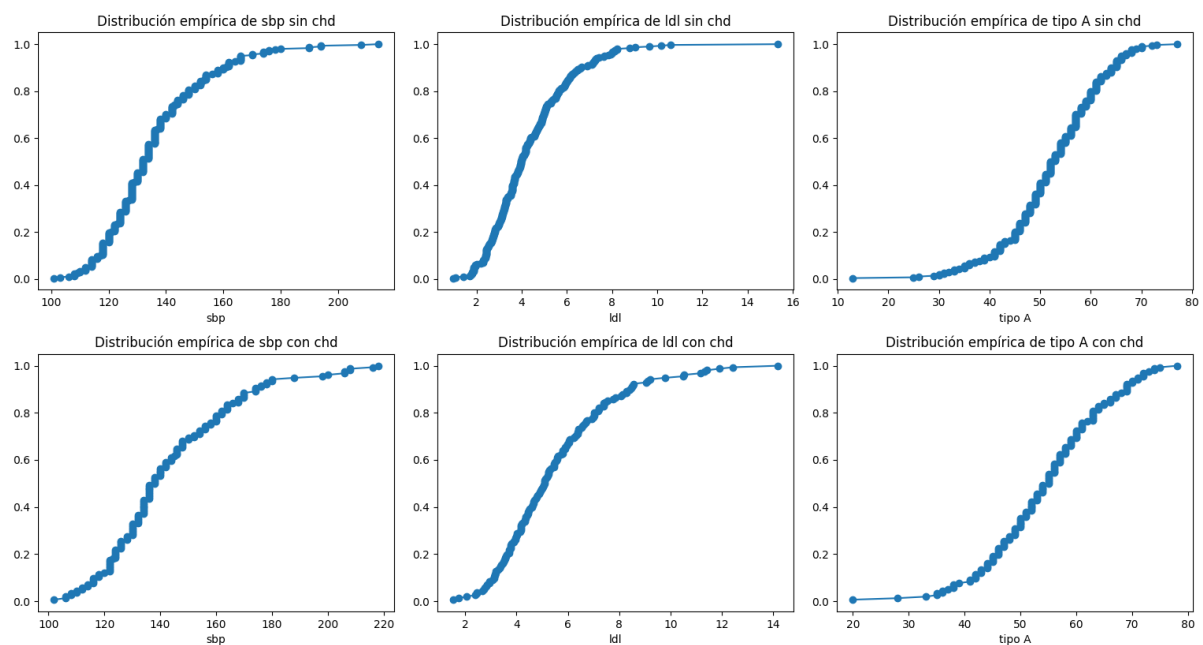
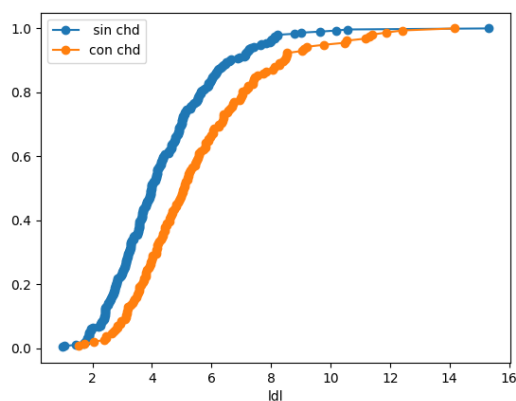
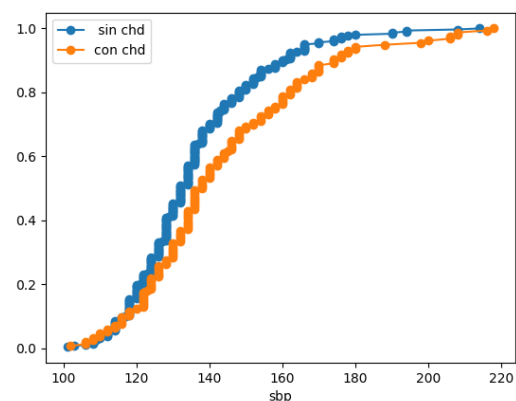


Figura 5: Frecuencias acumuladas con datos no agrupados.

En particular se puede notar que las personas que tienen *chd* presentan una tendencia a tener mayor colesterol o una presión mas elevada, como se puede observar en la figura 6



(a) Comparación de LDL con y sin chd.



(b) Comparación de SBP con y sin chd.

Figura 6: Comparación de frecuencias acumuladas

El código del programa se encuentra en el anexo apartado [K.6](#).

G. División por grupos A, B y C

A continuación, en lo que respecta del informe, vamos a analizar los datos separándolos en tres grupos: A) No tienen enfermedad coronaria y no tienen antecedentes familiares, B) Tienen una de las dos condiciones, C) tienen ambas condiciones.

Primero se calcularon los cuartiles 1 y 3 de los datos que se indican a continuación.

Grupo A:

Aspecto	Cuartil 1	Cuartil 3
sbp	126	145
ldl	3	5
tipo A	48	58

Grupo B:

Aspecto	Cuartil 1	Cuartil 3
sbp	124	148
ldl	3	6
tipo A	47	60

Grupo C:

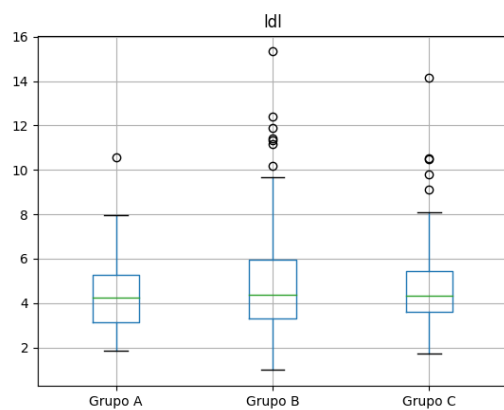
Aspecto	Cuartil 1	Cuartil 3
sbp	126	151
ldl	4	5
tipo A	47	61

El código del programa se encuentra en el anexo apartado [K.7](#).

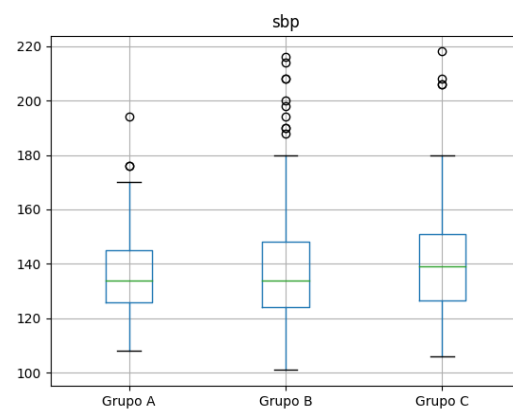
H. Boxplots de los grupos

Acá ponemos una explicación/chamuyo de los gráficos

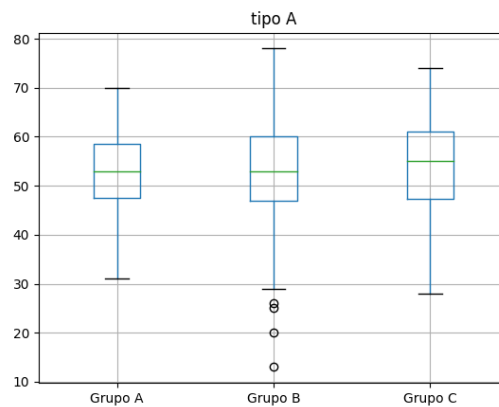
El código del programa se encuentra en el anexo apartado [K.8](#).



(a) Boxplot LDL



(b) Boxplot SBP



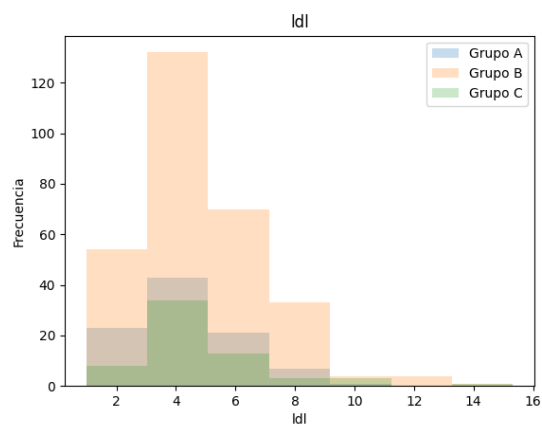
(c) Boxplot Tipo A

Figura 7: Boxplots con grupo A, B o C

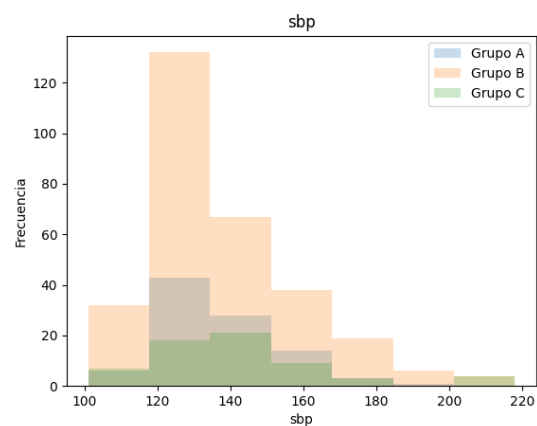
I. Histograma de los grupos

En los gráficos de la figura 8 podemos apreciar los histogramas de las variables sbp, ldl y tipoA separando los datos en los tres grupos A, B y C. En

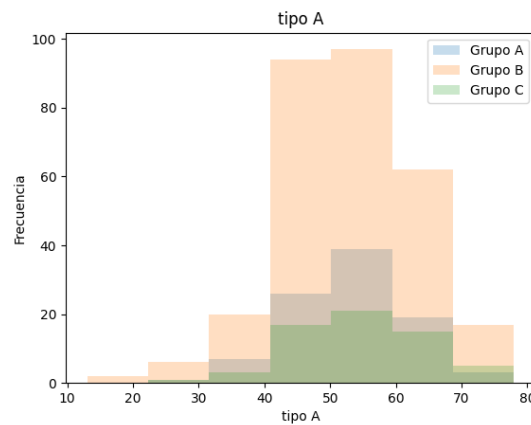
El código del programa se encuentra en el anexo apartado K.9.



(a) Histograma LDL



(b) Histograma SBP



(c) Histograma Tipo A

Figura 8: Histogramas con grupo A, B o C

J. Análisis final de los grupos

El código del programa se encuentra en el anexo apartado [K.10](#).

Grupo A:

Aspecto	Media	Coef. Simetria
sbp	136.68	0.98
ldl	4.42	0.79
tipo A	52.81	-0.33

Grupo B:

Aspecto	Media	Coef. Simetria
sbp	138.06	1.13
ldl	4.79	1.19
tipo A	52.88	-0.38

Grupo C:

Aspecto	Media	Coef. Simetria
sbp	142.21	1.23
ldl	4.94	1.9
tipo A	54.6	-0.2

PONER UNA CONCLUSIÓN FINAL DE LOS DATOS CORRESPONDIENTE A LOS GRUPOS

K. Anexo

K.1. Código del ejercicio A

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
5             'chd'],
6             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, '
7             chd': bool})
8
9 sbp = data['sbp']
10 ldl = data['ldl']
11
12 # Gráfico de constante vs variables
13 y = 1
14 for x in range(0, len(sbp)):
15     axx = plt.subplot(2,1,1)
16     plt.plot(sbp[x], y, 'x', color='blue')
17     plt.title("SBP")
18     axx.set_xlabel('Presión arterial sistólica')
19     axx2 = plt.subplot(2,1,2)
20     plt.plot(ldl[x], y, 'x', color='blue')
21     axx2.set_xlabel('Colesterol LDL')
22     plt.title("LDL")
23     plt.tight_layout()
24 plt.savefig("A Gráficos\\" + "CteVsVM" + ".png")
25 plt.show()
26
27 # Gráfico de variables vs. numero de muestra
28 ax = plt.subplot(2,1,1)
29 plt.plot(sbp, '.')
30 plt.title("SBP")
31 ax.set_ylabel('Presión arterial sistólica')
32 ax2 = plt.subplot(2,1,2)
33 plt.plot(ldl, '.')
34 ax2.set_xlabel('Número de muestras')
35 ax2.set_ylabel('Colesterol LDL')
36 plt.title("LDL")
37 plt.tight_layout()
38 plt.savefig("A Gráficos\\" + "SdTiempo" + ".png")
39 plt.show()
```

K.2. Código del ejercicio B

```
1 import pandas as pd
2 from tabulate import tabulate
3
4 file = open("B Resultados.txt","w")
5
6 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
7             'chd'],
8             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, '
9             chd': bool})
10
11 file.write("Resultados:\n")
12 table = [["Aspecto", "Mínimo Valor", "Máximo Valor", "Media", "Mediana", "Desvío Estandar",
13           "Cuartil 1", "Cuartil 3", "Rango Intercuartílico", "Coef. Simetría", "Coef. de
14           Kurtosis"]]
15
16 # Calcula los parametros de cada variable
17 for aspect in ['sbp', 'ldl']:
18     mV = data[aspect].min()
19     MV = data[aspect].max()
20     m = data[aspect].mean()
21     M = data[aspect].median()
22     DE = data[aspect].std()
23     q1, q3 = data[aspect].quantile([0.25, 0.75])
24     RI = q3 - q1
25     CS = (q3 - q1)/(q3 + q1)
26     K = data[aspect].kurtosis()-3
27     table.append([aspect, round(mV, 2), round(MV,2), round(m, 2), round(M, 2), round(DE,
28         2), round(q1, 2), round(q3, 2), round(RI, 2), round(CS, 2), round(K, 2)])
29
30 file.write(tabulate(table))
31 file.write("\n\n")
```

K.3. Código del ejercicio C

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
5             'chd'],
6             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, 'chd': bool})
7
8 file = open("C Resultados.txt", "w") # Se guardan los porcentajes de Outliers en un txt
9 file.write("Porcentajes de Outliers:\n")
10 outliersColor = dict(markerfacecolor='b', marker='X')
11 for aspecto in ['sbp', 'ldl']:
12     q1, q3 = data[aspecto].quantile([0.25, 0.75])
13     RI = q3 - q1
14     count = 0
15     for y in range(0, len(data[aspecto])):
16         if data[aspecto][y] > q3 + 1.5*RI or data[aspecto][y] < q1 - 1.5*RI:
17             count = count + 1
18     file.write(aspecto + ": " + str(round(count/len(data[aspecto])*100, 2)) + "%\n")
19     plt.boxplot(data[aspecto], flierprops=outliersColor)
20     plt.title(aspecto)
21     plt.savefig("C Gráficos\\"+aspecto+".png")
22     plt.show()
```

K.4. Código del ejercicio D

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4
5 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
6             'chd'],
7             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': int, 'chd': bool},
8             skipinitialspace=True)
9
10 N = data.shape[0]
11
12 bins = {'sbp': 9, 'ldl': 9}
13
14 for aspecto in ['sbp', 'ldl']:
15     h, f = np.histogram(data[aspecto], bins=bins[aspecto])
16
17     amp = f[1] - f[0]
18
19     h = h / N
20
21     f -= amp / 2 # Puntos en el medio
22     f = np.append(f, f[-1] + amp)
23
24     h = np.append(h, 0) # Agregado de 0s en los bordes
25     h = np.append([0], h)
26
27     plt.title("Polígono de frecuencias relativas de " + aspecto)
28     plt.plot(f, h, 'o-')
29     plt.xlabel("Valores de " + aspecto)
30     plt.ylabel("Frecuencia relativa")
31     plt.savefig("D Gráficos\\" + aspecto)
32     plt.show()
```

K.5. Código del ejercicio E

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
5             'chd'],
6             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, '
7             chd': bool})
8
9 sbp = data['sbp']
10 ldl = data['ldl']
11 count = 0
12 for j in range(0, len(sbp)):
13     if (sbp[j] < 130 and ldl[j] < 8):
14         count = count + 1
15 file = open("E Resultados.txt", "w")
16 file.write("Proporción de muestras:\n")
17 file.write(str(round(count/len(sbp)*100, 4)) + "%\n")
18
19 for i in range(0, len(sbp)):
20     plt.plot(sbp[i], ldl[i], 'x', color='blue')
21 plt.xlabel('SBP')
22 plt.ylabel('LDL')
23
24 plt.axhline(y=8, xmin=0, xmax=0.3, color='r', linestyle='-.', linewidth=2) # Plot a
25                                     horizontal line using axhline() in pyplot
26 plt.axvline(x=130, ymin=0, ymax=0.5, color='r', linestyle='-.', linewidth=2) # Plot a
27                                     vertical line using axvline() in pyplot
28 plt.savefig("E Gráficos\\"+"SBPvsLDL"+" .png")
29 plt.show()
```

K.6. Código del ejercicio F

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4
5 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
6             'chd'],
7             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': int, 'chd': bool},
8             skipinitialspace=True)
9 N = data.shape[0]
10
11 i = 1
12
13 for aspecto in ['sbp', 'ldl', 'tipo A']:
14     for cond in [False, True]:
15
16         x = data[data.chd == cond][aspecto].values
17
18         x.sort()      # Ordenar los datos
19         y = np.array(range(1, len(x)+1)) / len(x)      # y = i/n
20
21         plt.subplot(2, 3, i)
22         plt.title("Distribución empírica de " + aspecto + (" con" if cond else " sin") +
23             " chd")
24         plt.xlabel(aspecto)
25         plt.plot(x, y, 'o-', label=("con" if cond else " sin") + " chd")
26         # plt.savefig("F Gráficos\\" + aspecto + '_' + str(cond))
27         # plt.show()
28         i += 3
29     # plt.legend()
30     # plt.savefig("F Gráficos\\" + aspecto)
31     # plt.show()
32     i -= 5
33
34 # plt.get_current_fig_manager().full_screen_toggle()
35 plt.get_current_fig_manager().window.showMaximized()
36 plt.tight_layout()
37 # plt.savefig("F Gráficos\\Total")
38 plt.show()
```

K.7. Código del ejercicio G

```
1 import pandas as pd
2 from tabulate import tabulate
3
4 file = open("G Resultados.txt","w")
5
6 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
7             'chd'],
8             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, '
9             chd': bool})
10
11 GrupoA=data[(data['chd'] == False) & (data['famhist'] == True)]      # No tiene enfermedad
12                               ni antecedentes
13 GrupoB=data[((data['chd'] == False) & (data['famhist'] == False)) | # No tiene enfermedad
14             ((data['chd'] == True) & (data['famhist'] == True))]    # Tiene enfermedad
15                               pero sí antecedentes
16                               pero no antecedentes
17 GrupoC=data[(data['chd'] == True) & (data['famhist'] == False)]    # Tiene antecedentes
18                               y la enfermedad
19
20 for Group in [GrupoA, GrupoB, GrupoC]:
21     if Group.equals(GrupoA):
22         file.write("Grupo A:\n")
23     elif Group.equals(GrupoB):
24         file.write("Grupo B:\n")
25     else:
26         file.write("Grupo C:\n")
27
28     table = [["Aspecto", "Cuartil 1", "Cuartil 3"]]
29
30     for aspect in ['sbp', 'ldl', 'tipo A']:
31         q1, q3 = Group[aspect].quantile([0.25, 0.75])
32         table.append([aspect, round(q1), round(q3)])
33
34     file.write(tabulate(table))
35     file.write("\n\n")
```

K.8. Código del ejercicio H

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
5               'chd'],
6               dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, 'chd': bool})
7 GrupoA=data[(data['chd'] == False) & (data['famhist'] == True)]      # No tiene enfermedad
8                               ni antecedentes
9 GrupoB=data[((data['chd'] == False) & (data['famhist'] == False)) | # No tiene enfermedad
10              ((data['chd'] == True) & (data['famhist'] == True))]   # Tiene enfermedad
11                               pero sí antecedentes
12                               pero no antecedentes
13 GrupoC=data[(data['chd'] == True) & (data['famhist'] == False)]    # Tiene antecedentes
14                               y la enfermedad
15
16 for aspecto in ["sbp", "ldl", 'tipo A']:
17     data = pd.concat([GrupoA[aspecto].rename("Grupo A"), GrupoB[aspecto].rename("Grupo B"),
18                       GrupoC[aspecto].rename("Grupo C")],
19                       ignore_index=False, axis=1)
20     boxplot=data.boxplot()
21     boxplot.plot()
22     plt.title(aspecto)
23     plt.savefig("H Gráficos\\"+aspecto+".png")
24     plt.show()
```


K.9. Código del ejercicio I

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import math
4
5 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
6             'chd'],
7             dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, '
8             chd': bool})
9
10 GrupoA=data[(data['chd'] == False) & (data['famhist'] == True)]      # No tiene enfermedad
11                               ni antecedentes
12 GrupoB=data[((data['chd'] == False) & (data['famhist'] == False)) | # No tiene enfermedad
13             ((data['chd'] == True) & (data['famhist'] == True))]    # Tiene enfermedad
14                               pero sí antecedentes
15                               pero no antecedentes
16 GrupoC=data[(data['chd'] == True) & (data['famhist'] == False)]    # Tiene antecedentes
17                               y la enfermedad
18
19 for aspecto in ["sbp","ldl",'tipo A']:
20     data = pd.concat([GrupoA[aspecto].rename("Grupo A"), GrupoB[aspecto].rename("Grupo B"),
21                     GrupoC[aspecto].rename("Grupo C")],
22                     ignore_index=False, axis=1)
23     shape = data.shape
24     hist=data.plot.hist(bins=math.ceil(math.log(shape[0])), alpha=0.25)
25     hist.set_ylabel("Frecuencia")
26     hist.set_xlabel(aspecto)
27     hist.plot()
28     plt.title(aspecto)
29     plt.savefig("I Gráficos\\" + aspecto + ".png")
30     plt.show()
```

K.10. Código del ejercicio J

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 from tabulate import tabulate
4
5 file = open("J Resultados.txt","w")
6
7 data = pd.read_table("heart2022.txt", sep=' ', names=['sbp', 'ldl', 'famhist', 'tipo A',
8 'chd'],
9 dtype={'sbp': int, 'ldl': float, 'famhist': bool, 'tipo A': float, '
10 chd': bool})
11
12 GrupoA=data[(data['chd'] == False) & (data['famhist'] == True)] # No tiene enfermedad
13 ni antecedentes
14 GrupoB=data[((data['chd'] == False) & (data['famhist'] == False)) | # No tiene enfermedad
15 pero sí antecedentes
16 ((data['chd'] == True) & (data['famhist'] == True))] # Tiene enfermedad
17 pero no antecedentes
18 GrupoC=data[(data['chd'] == True) & (data['famhist'] == False)] # Tiene antecedentes
19 y la enfermedad
20
21 for Group in [GrupoA, GrupoB, GrupoC]:
22
23     if Group.equals(GrupoA):
24         file.write("Grupo A:\n")
25     elif Group.equals(GrupoB):
26         file.write("Grupo B:\n")
27     else:
28         file.write("Grupo C:\n")
29
30     table = [["Aspecto", "Media", "Coef Simetria"]]
31
32     for aspect in ['sbp', 'ldl', 'tipo A']:
33         media = Group[aspect].mean()
34         simetria = Group[aspect].skew(axis = 0, skipna = True)
35         table.append([aspect, round(media, 2), round(simetria,2)])
36
37     file.write(tabulate(table))
38     file.write("\n\n")
```