

**2.3.**

Considérese la población de los grupos de la materia Introducción a la Estadística que se imparte en cierta universidad. La universidad tiene 647 estudiantes de esta materia repartidos en 15 grupos con  $M_i$  estudiantes cada grupo según la tabla siguiente:

<i>Grupos (Población)</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$M_i$	44	33	26	22	76	63	20	44	54	34	46	24	46	100	15

Extraemos una muestra de cinco grupos con probabilidades proporcionales a los tamaños de los grupos con reemplazo y anotamos el total de horas durante una semana que todos los estudiantes de cada grupo han empleado para estudiar la materia de Introducción a la Estadística. Los datos se recogen en la siguiente tabla:

<i>Grupos (Muestra)</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>Horas</i>	120	203	100	90	40

Si se ha selecciona la muestra  $\{a, b, c, d, e\}$  por el método del tamaño acumulativo, estimar la cantidad promedio de tiempo semanal que un estudiante empleó para estudiar la materia Introducción a la Estadística midiendo la calidad de la estimación. Estimar por intervalos al 95%.

Primero cargaremos la informacion de la poblacion y la des muestras, con ellos obtendremos la lista de intervalos, la cual se muestra a continuacion

```
In [155]: import pandas as pd
import numpy as np
poblacion = pd.read_csv('tabla.csv', sep=';', na_values=".")
muestras = pd.read_csv('muestra.csv', sep=';', na_values=".")
total_alumnos = poblacion['tamaño'].sum()

intervals_list = []

linf = 0
lsup = 0
for x in poblacion['tamaño']:
    linf = lsup+1
    lsup = lsup + x
    intervals_list.append([linf,lsup,])

intervals_df = pd.DataFrame(intervals_list, columns=['linf','lsup'])
intervals_df
```

Out[155]:

	linf	lsup
0	1	44
1	45	77
2	78	103
3	104	125
4	126	201
5	202	264
6	265	284
7	285	328
8	329	382
9	383	416
10	417	462
11	463	486
12	487	532
13	533	632
14	633	647

ahora crearemos una lista de errores y los ubicaremos en la lista de intervalos para encontrar su clase perteneciente

```
In [156]: e_list = [389,324,88,123,509]
          solution_list = []

          for e in e_list:
              i = 0
              for x in intervals_df.values:
                  i = i+1
                  if x[0] <= e <= x[1]:
                      # print(i, x[0], x[1])
                      solution_list.append([e,i,poblacion['tamaño'][i-1]])
          solution_df = pd.DataFrame(solution_list, columns=['error','clase','tamaño'])
          solution_df
```

Out[156]:

	error	clase	tamaño
0	389	10	34
1	324	8	44
2	88	3	26
3	123	4	22
4	509	13	46

A continuacion calculare la tabla de muestras y probabilidades

```
In [157]: tabla2 = []
          i = 0
          for e in solution_df['clase']:
              grupo = e
              tamaño = solution_df['tamaño'][i]
              xi = muestras['horas'][i]
              xi_barra = muestras['horas'][i]/solution_df['tamaño'][i]
              pi = tamaño/total_alumnos
              tabla2.append([grupo,tamaño,xi,xi_barra,pi])
              i = i+1
          # tabla1 = pd.DataFrame(muestras['horas']/poblacion['tamaño'],poblacion['tamaño']/total_alumnos, columns=['grupo','tamaño','xi','xi_','pi'])
          tabla3 = np.around(tabla2, decimals=4)

          tabla_df = pd.DataFrame(tabla3,columns=['grupo','tamaño','xi','xi_','pi'])
          tabla_df
```

Out[157]:

	grupo	tamaño	xi	xi_	pi
0	10.0	34.0	120.0	3.5294	0.0526
1	8.0	44.0	203.0	4.6136	0.0680
2	3.0	26.0	100.0	3.8462	0.0402
3	4.0	22.0	90.0	4.0909	0.0340
4	13.0	46.0	40.0	0.8696	0.0711

```
In [158]: n = len(tabla_df['xi_'])
          suma = 0
          for x in tabla_df['xi_']:
              suma = suma + x
          print(suma/n, "horas")
```

3.38994 horas

Una estimacion insesgada del promedio de horas/semana de un estudiante es 3.4 hrs/semana