

Universidad laica Eloy Alfaro de Manabi

Facultad de Ciencias Informaticas

Carrera de Tecnologia de la Informacion

Nombre: Vera Rivera Jefferson Jesus

Curso: Sexto "B"

Materia: Modelamiento y Simulacion

Docente: Ing. Jorge Anibal Moya Delgado

Caso de estudio: Importaciones dentro de la aduana del ecuador en el año 2021(enero)

SISTEMA ESTADISTICO DE IMPORTACIONES DE LA ADUANA EN EL ECUADOR

En el proyecto presente se desarrollara un sistema para el control del centro importaciones de la aduana del Ecuador permitiendo controlar la adquisición de productos, el objetivo es implementar una metodología física basada en los datos obtenidos donde se presentan datos numéricos de los cuales se implementará medidas estadísticas, el contexto en que realizo fue en la aduana del Ecuador, su unidad de análisis son las importaciones que tiene la aduana (SENAE).

Con este estudio se estima mejorar la comprensión de personas comunes así los temas de interés como son las importaciones que se realizan al país, con este trabajo se contribuirá un informe detallado sobre los análisis y proceso estadísticos claros y precisos además como a contar con un documento que respalde la información presentada.

El modelo que se ha utilizado para darle solución a estos problemas a sido el modelo físico ya que con este se podrá realizar una construcción teórica y matemática para verificar el comportamiento de algunas entidades, reduciendo así muchas dificultades dentro de la aduana. Se pretende establecer una posible solución del problema antes detallado mediante el estudio de varios métodos estadísticos para lograr la obtención de resultados acorde a la petición necesaria para entender detalladamente cada uno de los procesos que conllevan las importaciones dentro de la entidad aduanera.

Ejercicio

Este problema se realiza de acuerdo a la falta de procedimientos documentados en áreas que no son operativas afectando el conocimiento de los organismos de control sobre el sistema aduanero, su justificación es importante ya que los diferentes países de origen envían sus productos y estoy no esta manera controlada genera pérdidas tanto en los productos como en la economía del país, por esto se realizará un sistema que ayuda al control de cuales, de la verificación del producto con los datos obtenidos, para resolver este problema se ha realizado una mejora continua de controles aduaneros en zonas primaria y secundaria y el crecimiento del fortalecimiento de la capacidad operativa de control posterior.

Datos obtenidos a mostrar

En esta parte se mostraran los datos obtenidos de la pagina de la Aduana en formato excel guardado dentro de los documentos

Histogramas

Histograma que mostrara el pago de la moneda diariamente

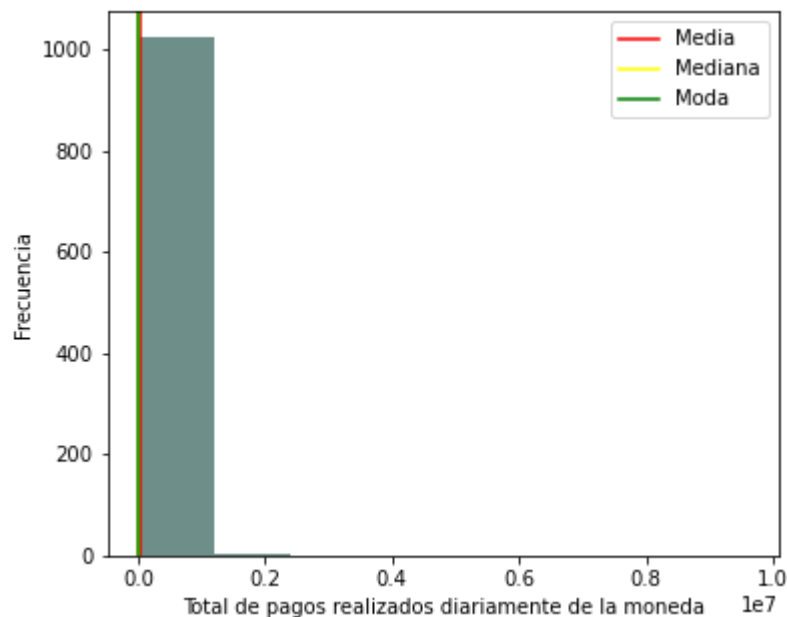
- 1 A continuacion se mostrara un histograma de las cantidades fisicas importadas y recibidas dentro de nuestro Pais

In [57]:

```

1 #Gráficos
2 x=datos["CANTIDAD FÍSICA"]
3 #modificacion del grafico anchura,altura
4 plt.figure(figsize=(6,5))
5 plt.hist(x,bins=8,color='#6E8E8A')
6 plt.axvline(x.mean(),color='red',label='Media')
7 plt.axvline(x.median(),color='yellow',label='Mediana')
8 plt.axvline(x.mode()[0],color='green',label='Moda')
9 #titulo de las ejes de la X
10 plt.xlabel('Total de pagos realizados diariamente de la moneda')
11 #titulo de las ejes de la y
12 plt.ylabel('Frecuencia')
13 plt.legend()
14 plt.show()

```



Histograma que mostrara el precio de la moneda en el mercado

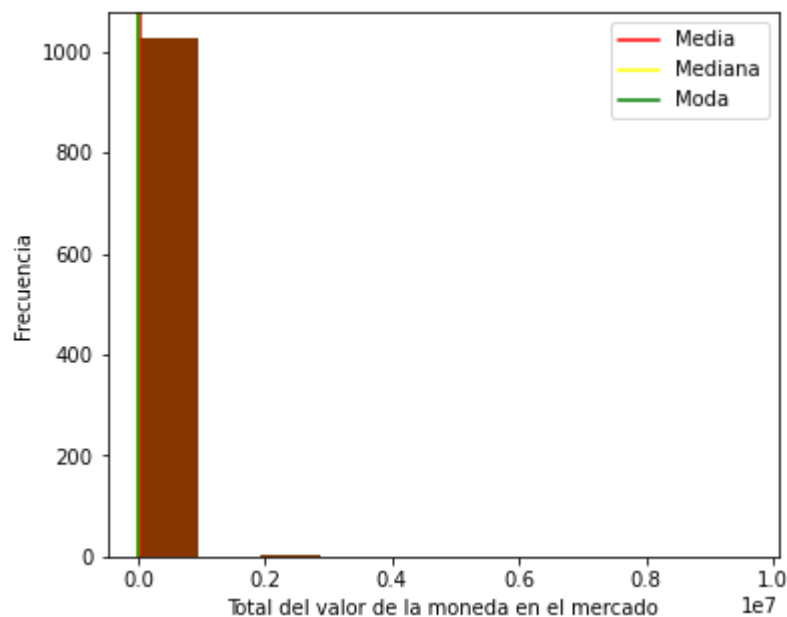
A continuacion se mostrara un histograma de el peso neto en kilogramos de los productos importados y recibidos dentro de nuestro Pais

In [58]:

```

1 x=datos["PESO NETO (KG)"]
2 #modificacion del grafico anchura,altura
3 plt.figure(figsize=(6,5))
4 plt.hist(x,bins=None,color='#873600')
5 plt.axvline(x.mean(),color='red',label='Media')
6 plt.axvline(x.median(),color='yellow',label='Mediana')
7 plt.axvline(x.mode()[0],color='green',label='Moda')
8 #titulo de las ejes de la X
9 plt.xlabel('Total del valor de la moneda en el mercado')
10 #titulo de las ejes de la y
11 plt.ylabel('Frecuencia')
12 plt.legend()
13 plt.show()

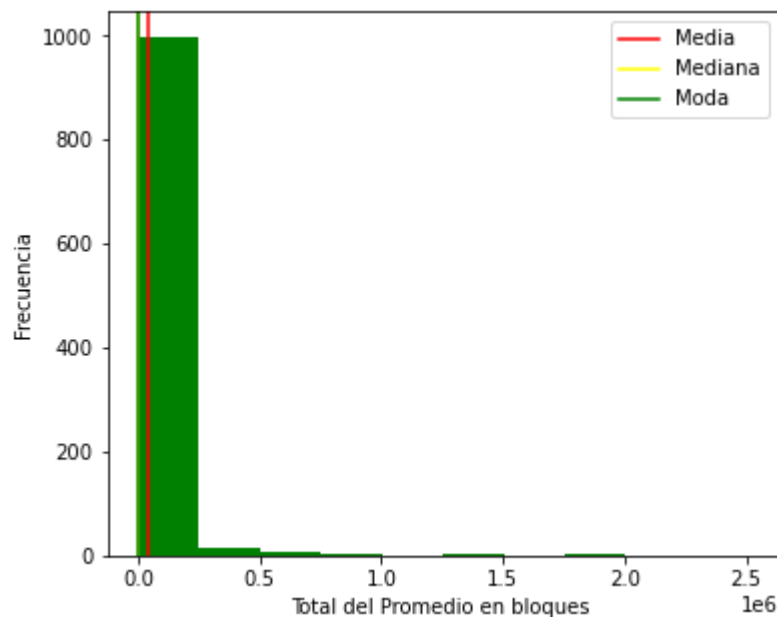
```



Histograma que mostrara el promedio de la moneda por bloques

A continuacion se mostrara un histograma de la cantidad en moneda en este caso en dolares de los productos que han sido importados y recibidos dentro de nuestro Pais

```
In [59]: 1 x=datos["CIF (DÓLARES)"]
2 #modificacion del grafico anchura,altura
3 plt.figure(figsize=(6,5))
4 plt.hist(x,bins=None,color='green')
5 plt.axvline(x.mean(),color='red',label='Media')
6 plt.axvline(x.median(),color='yellow',label='Mediana')
7 plt.axvline(x.mode()[0],color='green',label='Moda')
8 #titulo de las ejes de la X
9 plt.xlabel('Total del Promedio en bloques ')
10 #titulo de las ejes de la y
11 plt.ylabel('Frecuencia')
12 plt.legend()
13 plt.show()
```



Media

La media es el valor que se obtienen de los datos, si todos ellos han sido o serán iguales.

Dentro de este sistema se calcula la media de las columnas nombradas a continuación:

- CIF en dolares
- Peso Neto de los productos en KiloGramos
- Cantidad Fisica de los productos enviados

```
In [63]: 1 #Calculo de la media
2 print("MEDIA:")
3 #Entrada de los datos por medios del array
4
5 t4 =datos["CIF (DÓLARES)"].mean()
6 t5 =datos["PESO NETO (KG)"].mean()
7 t6 =datos["CANTIDAD FÍSICA"].mean()
8 #Mostreo de los datos
9 print( "\nLa Media de los pagos en moneda realizados en dolares --> ", t4)
10 print( "\nLa Media del peso neto en kilogramos de los productos importados
11 print( "\nLa Media de la cantidad fisica de cad auno de lor productos import
12
```

MEDIA:

La Media de los pagos en moneda realizados en dolares --> 37926.634884578074

La Media del peso neto en kilogramos de los productos importados --> 24687.474123181382

La Media de la cantidad fisica de cad auno de lor productos importados --> 35200.99416515033

Media Aritmetica

La media aritmetica es el valor que ocupa la posición central dependiendo si el número de datos es par, la mediana es la suma de ambos dividido para dos.

Dentro de este sistema se calcula la media aritmetica de las columnas nombradas a continuacion:

- CIF en dolares
- Peso Neto de los productos en KiloGramos
- Cantidad Fisica de los productos enviados

```
In [73]: 1 #Calculo de la Mediana
2 print("Mediana:")
3 #Entrada de los datos por medios del array
4 m1 =datos["CIF (DÓLARES)"].median()
5 m2 =datos["PESO NETO (KG)"].median()
6 m3 =datos["CANTIDAD FÍSICA"].median()
7 #Mostreo de los datos
8 print( "\nLa Mediana de los pagos en moneda realizados en dolares --> ", m1
9 print( "\nLa Mediana del peso neto en kilogramos de los productos importados
10 print( "\nLa Mediana de la cantidad fisica de cad auno de lor productos impo
```

Mediana:

La Mediana de los pagos en moneda realizados en dolares --> 2345.105

La Mediana del peso neto en kilogramos de los productos importados --> 68.843

La Mediana de la cantidad fisica de cad auno de lor productos importados --> 52.0

Moda

La moda es el valor que más se repite o el que tiene una mayor frecuencia.

Dentro de este sistema se calcula la moda de las columnas nombradas a continuacion:

- CIF en dolares
- Peso Neto de los productos en KiloGramos
- Cantidad Fisica de los productos enviados


```
In [74]: 1 #Calculo de La Moda
2 print("Moda:")
3 #Entrada de Los datos por medios del array
4 mo1 = datos["CIF (DÓLARES)"].mode()
5 mo2 = datos["PESO NETO (KG)"].mode()
6 mo3 = datos["CANTIDAD FÍSICA"].mode()
7 #Mostreo de Los datos
8 print( "\nLa Moda de los pagos en moneda realizados en dolares --> \n\n", m
9 print( "\nLa Moda del peso neto en kilogramos de los productos importados -
10 print( "\nLa Moda de la cantidad fisica de cad auno de lor productos importa
11 #agregandolas a La Libreria panda
12 pd.DataFrame(mo1)
13 pd.DataFrame(mo2)
14 pd.DataFrame(mo3)
15
16
17
```

Moda:

La Moda de los pagos en moneda realizados en dolares -->

```
0      68.080
1     190.945
dtype: float64
```

La Moda del peso neto en kilogramos de los productos importados -->

```
0      20.0
1     300.0
dtype: float64
```

La Moda de la cantidad fisica de cad auno de lor productos importados -->

```
0      1.0
dtype: float64
```

Out[74]:

```
0
--
0  1.0
```

Medidas de tendencias

Se mostrara alguna de las medidas de tendencia central entre esos estan:

--> Valores iniciales de salvamiento

--> Periodos de recuperacion

--> Cuartiles.

Cuartiles

```
In [69]: 1 #Medidas de tendencia central
2
3 datos[['PESO NETO (KG)', 'CIF (DÓLARES)', 'CANTIDAD FÍSICA']].describe()
```

Out[69]:

	PESO NETO (KG)	CIF (DÓLARES)	CANTIDAD FÍSICA
count	1.031000e+03	1.031000e+03	1.031000e+03
mean	2.468747e+04	3.792663e+04	3.520099e+04
std	3.226989e+05	1.613966e+05	3.445899e+05
min	1.000000e-03	3.600000e-01	2.000000e-03
25%	8.160000e+00	3.143310e+02	5.840000e+00
50%	6.884300e+01	2.345105e+03	5.200000e+01
75%	8.468470e+02	1.586017e+04	1.092158e+03
max	9.600000e+06	2.505414e+06	9.600000e+06

Luego se ingresan los datos dentro de un array con el total de expansion

```
In [71]: 1 df_1 = datos[:7]
2 print ("-----")
3 print ("---Datos estadísticos de acuerdo a la Subpartida---")
4 print ("-----")
5 df_1[['PESO NETO (KG)', 'CIF (DÓLARES)', 'CANTIDAD FÍSICA']].describe()
```

```
-----
---Datos estadísticos de acuerdo a la Subpartida---
-----
```

Out[71]:

	PESO NETO (KG)	CIF (DÓLARES)	CANTIDAD FÍSICA
count	7.000000	7.000000	7.000000
mean	3983.431143	17318.399429	4081.721000
std	10462.004751	37388.447607	10752.519624
min	0.429000	47.490000	1.000000
25%	2.815000	219.744500	3.000000
50%	4.140000	403.080000	4.000000
75%	82.541500	9713.173500	47.500000
max	27708.736000	100912.390000	28466.047000

Graficos adicionales

Uno de los graficos utilizados en la estadística es el siguiente donde se muestra los puntos de acuerdo a la subpartida de la cantidad física, del peso, de la cantidad en dolares

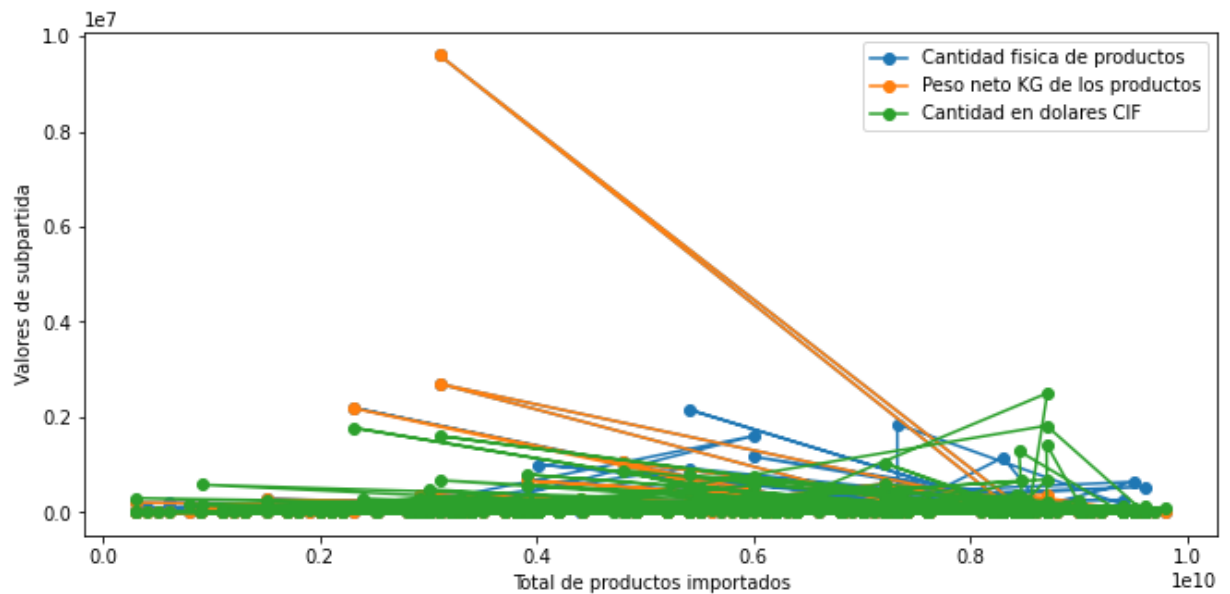
In [75]:

```

1
2 #Ingreso de las categoria en las ejes de la X
3 x = datos["SUBPARTIDA"]
4 #Ingreso de los datos al Array
5 t1 = datos["CANTIDAD FÍSICA"]
6 t2 = datos["PESO NETO (KG)"]
7 t3 = datos["CIF (DÓLARES)"]
8 #Modificacion del grafico
9 plt.figure(figsize=(11,5))
10 plt.plot(x,t1,x,t2,x,t3, marker='o')
11 #Imprecion de los datos y modificacion del grafico
12 plt.xlabel('Total de productos importados')
13 plt.ylabel('Valores de subpartida')
14 plt.legend(('Cantidad fisica de productos', 'Peso neto KG de los productos',

```

Out[75]: <matplotlib.legend.Legend at 0x1d3d778b7f0>



Otra grafica

Grafica para mostrar el crecimiento o los cambios realizados de las columnas de datos obtenidos en las cuales se han utilizado los siguientes:

Primer Grafico

- Cantidad fisica: que es la entrada de los datos de Array
- Subpartida que es el nombre de los datos en rttotal ya que es un dato unico

Segundo Grafico

-Peso Neto de los productos importados mostrados en la unidad de Kilogramos

Tercer Grafico

-Cantidad en dolares de cada uno de los productos importados

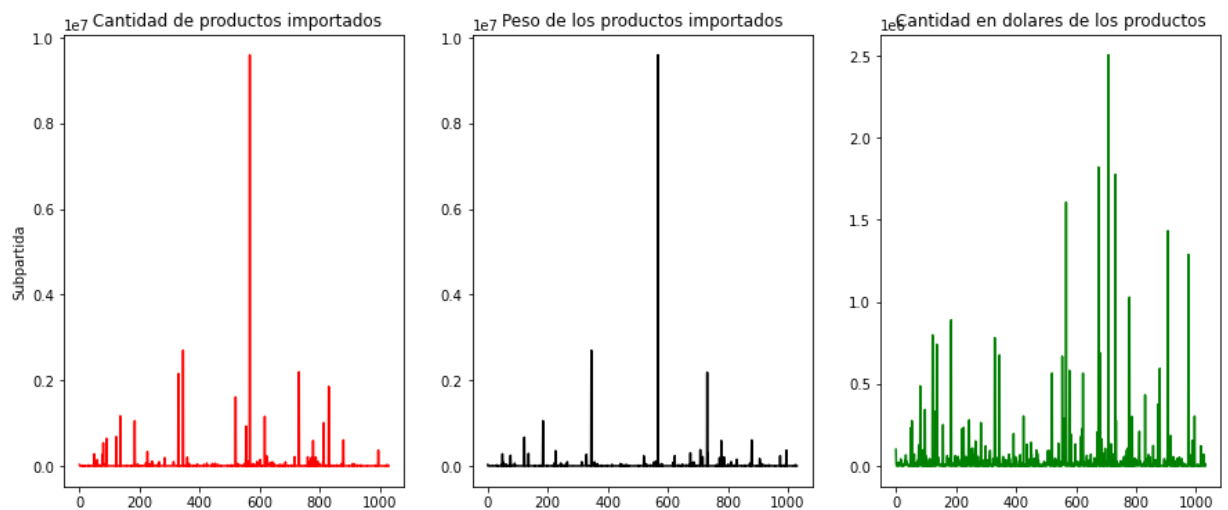
In [76]:

```

1
2
3 #Entra del rango al Array con Los datos que estan en la fuente de datos
4 x = range(1031)
5 #Configuracion de Las dimensiones que el grafico optendra
6 plt.figure(figsize=(15,6))
7 plt.subplot(131)
8 #Entrada de Los datos a Las array
9 t1 = datos["CANTIDAD FÍSICA"]
10 #'r'=> (rojo) es el color que se le puede agregar es una simbolo
11 p1 = plt.plot(x,t1, 'red')
12 plt.ylabel('Subpartida')
13 plt.title(' Cantidad de productos importados')
14 plt.subplot(132)
15 t2 = datos["PESO NETO (KG)"]
16 #'k'=> (negro)
17 p1 = plt.plot(x,t2,'black')
18 plt.title('Peso de los productos importados')
19 plt.subplot(133)
20 t3 = datos["CIF (DÓLARES)"]
21 #'g'=> (verde)
22 p1 = plt.plot(x,t3,'green')
23 plt.title('Cantidad en dolares de los productos')
24

```

Out[76]: Text(0.5, 1.0, 'Cantidad en dolares de los productos')



Datos unicos de la tabla

En esta parte se mostraran los datos unicos de la columna de cantidad en dolares por cada uno

de los productos importados lo cual sera guardado en la variable dfclases

- Se utilizo la columna nombrada CIF (DÓLARES)

In [77]:

```
1
2 # OBTENER LOS DATOS UNICOS DE LA TABLA
3 #Agregar la columna de valor inicial a una variable
4 lis = datos["CIF (DÓLARES)"].unique()
5 #Para poder sacar los valores unicos de la tabla
6 dfclases=pd.DataFrame(lis,columns=["CIF (DÓLARES)"])
7 dfclases
```

Out[77]:

	CIF (DÓLARES)
0	100912.390
1	17282.740
2	170.320
3	269.169
4	47.490
...	...
1024	17811.260
1025	464.690
1026	1073.800
1027	16818.670
1028	68.114

1029 rows × 1 columns

Frecuencias absolutas

La frecuencia absoluta es una medida estadística que muestra la cantidad de veces que se repite un suceso al realizar un número determinado de experimentos aleatorios.

Para obtener los datos de las frecuencias absolutas debe seguir los datos a continuación:

- Crear una lista con los valores de las frecuencias
- Agregar la columna al dataframe nombrada Fi
- Mostrar datos unicos guardados anteriormente

```
In [79]: 1
2 datafi=pd.crosstab(index=datos["CIF (DÓLARES)"], columns = "fi")
3 # Creamos una lista con los valores de las frecuencias
4 li = datafi.values
5 # agregamos una columna al dataframe
6 dfclases["fi"] = li
7 #observamos dfclase
8 dfclases
```

Out[79]:

	CIF (DÓLARES)	fi
0	100912.390	1
1	17282.740	1
2	170.320	1
3	269.169	1
4	47.490	1
...
1024	17811.260	1
1025	464.690	1
1026	1073.800	1
1027	16818.670	1
1028	68.114	1

1029 rows × 2 columns

Muestra una suma de los datos anteriores

```
In [80]: 1 total = dfclases.sum(axis=0)
2 total
3
```

Out[80]: CIF (DÓLARES) 3.910210e+07
fi 1.031000e+03
dtype: float64

Frecuencia Relativa

La frecuencia relativa es una medida estadística que se calcula como el cociente de la frecuencia absoluta de algún valor de la población/muestra (fi) entre el total de valores que componen la población/muestra (N).

Para obtener los datos de la frecuencia relativa debe seguir los siguientes pasos: se calcula y se agrega la columna adicional con el resultado de los datos utilizando algunos datos de la Frecuencia Absoluta

- Se calcula la frecuencia
- Se agrega una nueva columna
- Muestra los datos anteriores y los de la nueva columna

```
In [81]: 1 # Columna de Frecuencia relativa
2 total = dfclases.sum(axis=0)
3 datahi = dfclases["fi"]/total["fi"] # aqui calculamos la frecuencia
4 datahi.values
5 # agregamos nueva columna de frecuencia relativa
6 dfclases["hi"] = datahi
7 dfclases
```

Out[81]:

	CIF (DÓLARES)	fi	hi
0	100912.390	1	0.00097
1	17282.740	1	0.00097
2	170.320	1	0.00097
3	269.169	1	0.00097
4	47.490	1	0.00097
...
1024	17811.260	1	0.00097
1025	464.690	1	0.00097
1026	1073.800	1	0.00097
1027	16818.670	1	0.00097
1028	68.114	1	0.00097

1029 rows × 3 columns

Mostramos la suma de los datos utilizados anteriormente con las siguientes lineas de codigo

```
In [82]: 1 total1 = dfclases.sum(axis=0)
2 total1
```

Out[82]: CIF (DÓLARES) 3.910210e+07
fi 1.031000e+03
hi 1.000000e+00
dtype: float64

Suma de frecuencias relativas

Se realizo una suma de frecuencias relativas utilizando algunos de los datos anteriores los cuales seran guardados y se mostraran cada uno de los datos, mediante un for para que se puedan sumar los datos y mostrarlos de manera rapida

```

In [83]: 1 # La suma de las frecuencias Relativas nos da 1
2 # aquí vamos a calcular la frecuencia absoluta
3 FA = dfclases["fi"].values
4 # obtenemos FA
5 a=[]
6 b=0
7 for c in FA:
8     b = c + b
9     a.append(b)
10 dfclases["FA"] = a
11 HI = dfclases["hi"].values
12 # obtenemos HI
13 a=[]
14 b=0
15 for c in HI:
16     b = c + b
17     a.append(b)
18 dfclases["HI"] = a
19 dfclases

```

Out[83]:

	CIF (DÓLARES)	fi	hi	FA	HI
0	100912.390	1	0.00097	1	0.00097
1	17282.740	1	0.00097	2	0.00194
2	170.320	1	0.00097	3	0.00291
3	269.169	1	0.00097	4	0.00388
4	47.490	1	0.00097	5	0.00485
...
1024	17811.260	1	0.00097	1027	0.99612
1025	464.690	1	0.00097	1028	0.99709
1026	1073.800	1	0.00097	1029	0.99806
1027	16818.670	1	0.00097	1030	0.99903
1028	68.114	1	0.00097	1031	1.00000

1029 rows × 5 columns

Se realiza una suma entre los valores anteriores y se los muestra

```

In [84]: 1 total2 = dfclases.sum(axis=0)
2 total2

```

```

Out[84]: CIF (DÓLARES)    3.910210e+07
fi                    1.031000e+03
hi                    1.000000e+00
FA                    5.317070e+05
HI                    5.157197e+02
dtype: float64

```


Se muestra toda la informacion de la tabla con sus respectivas columnas

In [85]:

```
1 datos.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1031 entries, 0 to 1030
Data columns (total 8 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   SUBPARTIDA                           1031 non-null   int64
1   DESCRIPCIÓN ARANCELARIA              1031 non-null   object
2   DISTRITO                             1031 non-null   object
3   PAÍS ORIGEN                          1031 non-null   object
4   TIPO UNIDAD FÍSICA                   1031 non-null   object
5   CANTIDAD FÍSICA                      1031 non-null   float64
6   PESO NETO (KG)                      1031 non-null   float64
7   CIF (DÓLARES)                       1031 non-null   float64
dtypes: float64(3), int64(1), object(4)
memory usage: 64.6+ KB
```

In []:

```
1
```