OPTOMETRIST ENSEMBLE

Assignment : Intelligent Systems

Luis Rubén Mejia García

Abstract

To have a better understanding of this project it's important to proportionate some context related to the most common eye ametropia, some of their symptoms, diseases that can reduce visual capacity.

Astigmatism

Astigmatism is cause by a deformation on the cornea curvature. Normally the cornea has a curvature like a ball. When astigmatism increase the pols of our eyes starts to become flat, making the cornea sharp at the front of the eye, dispersing the light that goes to our eyes and by consequence blurring the image on the retina.

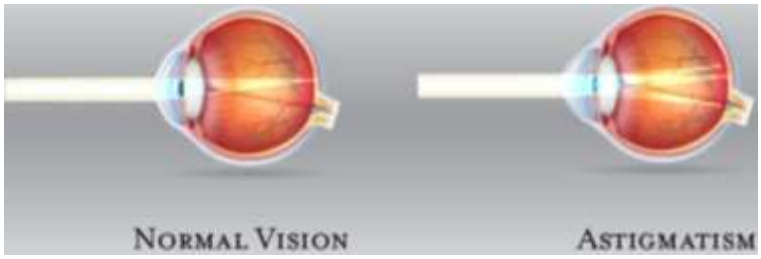

Fig. 1: Comparison between an eye without and with astigmatism

It's estimated that more than 35% percent of the global population presents this ametropia. Which some of the times comes its combined with myopia or farsightedness. Often its diagnosed wrong or confused with other ametropias. Due to the symptoms, that goes from blurred vision on near and far objects. Astigmatism can cause us to have a wrong distinction of distances, so objects can be seen too thin, short, wide, etc.

Fig. 2: This is how someone with astigmatism see

Symptoms:

Blurred vision or distorted

Headaches

Night Vision difficulty

Cross eyed

Vision discomfort

Myopia

When your eyeball or your cornea is too long and the protective layer of the eye is too curved, the light that enters your eye wont focus correctly. This cause difficulties to seeing distant objects.

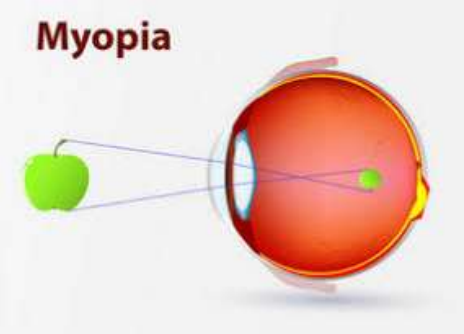

Fig. 3: Eye with myopia

Symptoms:

Faraway objects look blurred or fuzzy

Headaches

Eye strain

Squinting

Farsightedness

The eye does not refract light properly, it under focuses the light and this cause an incomplete image on the retina.

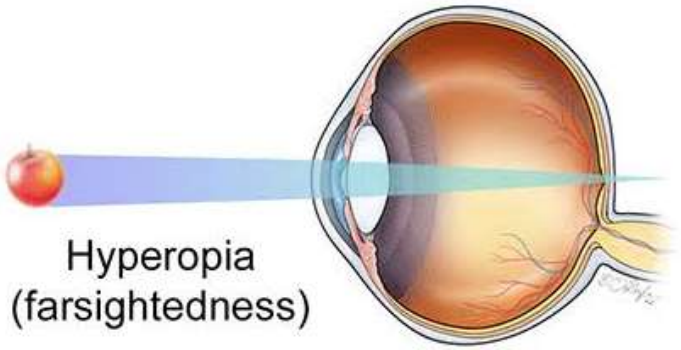

Fig. 4: Eye with farsightedness

Symptoms:

Nearby objects may appear blurry

Aching in or around the eyes

Need to squint to see clearly

Emmetrope

This eye doesn't have visual defect.

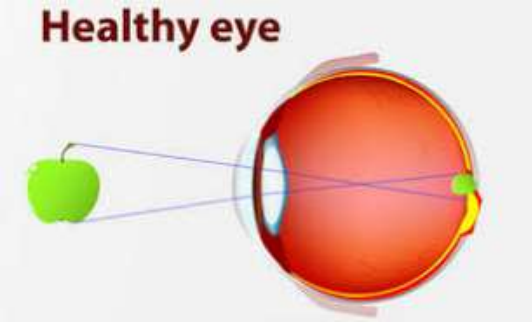

Fig. 5 Healthy eye

Symptoms:

None

Pruritus:

Medical term that means itching, it can be associated to dry eye or skin. It can be painful or irritating and could be localized to an area of your body.

Fig. 6 Eye with pruritus

Hyperemia:

An increased amount of blood flow in the vessels of an organ or tissue in the body.

Fig.7 Eye with hyperemia

Photophobia:

Light sensitivity a condition in which bright lights hurt your eyes.

Fig. 8 Person with photophobia

Diplopia:

Perception of a double vision. Can be cause by squint, retinal diseases, muscular or neurological alterations.

Fig. 9 Double vision

Nyctalopia:

Inability to see at night or in poor light. This can be associated with an inability to quickly adapt from a well illuminated to a poorly illuminated environment.

Fig. 10 Comparation with normal vision vs night blindness

Secretion:

A discharge of oil, mucus and skin cells coming from the eyes, are secreted by the ocular glands; one of their functions its to protect the eyes by getting rid of particles that can damage the eyes.

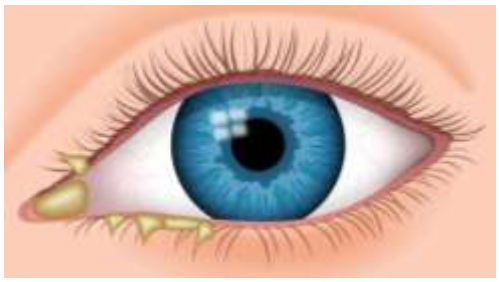

Fig. 11 Eye with secretion

Epiphora:

Over production or bad discharge of tears, can be cause by conjunctivitis, trauma, dry eye or bad function of the nasolacrimal system.

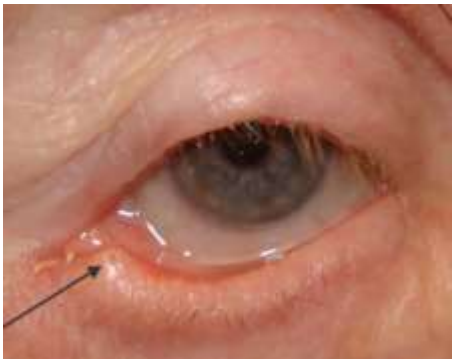

Fig. 12 Eye with overproduction of tears

Asthenopia:

Fatigue or weakness in the eyes, can cause epiphora, headache, ocular pain, blurry vision and ocular sensibility.

Fig. 13 Person with ocular fatigue

Factor that can simulate or create ametropias

- Diabetes
- Hypertension

- Hormonal disturbances
- Pregnancy or lactation
- Glaucoma
- Contraceptive methods
- Medicines
- Drugs, alcohol, smoke
- Exposure to blue lights
- Etc.

Introduction

It's important to implement the acquired knowledge on projects that really make the difference, projects than can solve any problematic. That's why I consider important to make an algorithm that can be an auxiliar to Opticas Mega View workers, which can help them to reduce time, perform both their accuracy. Its important to have in mine that this software won't replace the criteria of an optometrist, it will just help him out to diagnostic the patients. This idea came due to rise of clients in OMV, we attribute this to COVID-19 and the pandemic, because most of our clients during the past 2 years have been students, workers doing home office. The use of electronic devices has reduced the visual capacities of our users. Scared of the eye health of our current and future clients we have develop an algorithm capable of diagnose 3 of the most common ametropias. The other part of the software has the objective to proportionate an approximate quotation for the type of lens they will require. In a future we hope we have finished the client segmentation to proportionate the better lens treatment for each case. During this document we would try to measure the accuracy or the percentage of error we have, the possible solutions and everything regarding this project.

Development

Let's take about the first part of the algorithm. In this segment the objective is to obtain the symptoms, activities, and occupation the patient. These questions were taken from Opticas Mega View as part of their medical history records.

Fig. 14: Opticas Mega View medical history

We had to create our own dataset with all the anamnesis we have since 2019. We archive to gather more than 400 instances of clients records. This is how it looks like more less:

	Edad	Prurito	Hiperemia	Fotofobia	Diplopia	Nictalopia	Secrecion	Epifora	Astenopia	Vision distorcionada	Baja vision lejana
0	54.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0
1	19.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0
2	52.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
3	66.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	1.0	1.0
4	21.0	1.0	1.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	1.0
...	...	...	...	...	...	...	...	...	...	...	...
295	85.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	0.0	0.0	1.0
296	22.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
297	28.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0
298	19.0	1.0	1.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0
299	28.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0

Fig. 15: Part 1 of the dataset

Baja vision cercana	Activities	Ocupacion	Diagnostico	Tratamiento	Gasto	Arbol
1.0	Dispositivos	Ama de casa	Miopia con Astigmatismo y presvicie	Bifocal	1500.0	8
0.0	Dispositivos	Estudiante	Miopia con Astigmatismo	BlueBlock	1500.0	7
1.0	Leer	Trabajador	Miopia con Astigmatismo y presvicie	Bifocal	2500.0	8
0.0	Dispositivos	Trabajador	Miopia con Astigmatismo	Blanco	600.0	7
0.0	Leer	Ama de casa	Miopia con Astigmatismo	Blanco	600.0	7
...	...	...	...	...	...	...
1.0	Dispositivos	Pensionado	Hipermetropia con presvicie	Bifocal	980.0	5
0.0	Estudiar	Estudiante	Miopia con Astigmatismo	BlueBlock	1200.0	7
0.0	Leer	Trabajador	Miopia con Astigmatismo	BlueBlock	1200.0	7
0.0	Dispositivos	Estudiante	Miopia con Astigmatismo	BlueBlock	1600.0	7
0.0	Dispositivos	Trabajador	Miopia con Astigmatismo	Fotocromatico	1300.0	7

Fig. 16: Part 2 of the dataset

In our algorithm we wanted to know if the patient has present pruritus, hyperemia, photophobia, diplopia, nyctalopia, secretion, epiphora, asthenopia, blurred vision, low far vision, low near vision, in addition we ask the client for their daily activities and their occupation. This to know if they can be in danger, related to their eyes. This is the part of the code we used:

```
from numpy import array
#Obtencion de datos del paciente
#Para poder obtener una predicción hay que agregar realizar la anamnesis y
Empezar=input ("¿Quiere conocer que ametropía presentas? (Escriba Si o No): ")
if Empezar=='Si':
  print ("Bienvenido, para poder conocer el estado de sus ojos es importante que conteste las siguientes preguntas")
  Edad=input ("¿Qué edad tienes?: ")
  Prurito=input ("¿Ha tenido comezón en los ojos?: ")
  Hiperemia=input ("¿Ha notado que se enrojece su esclerótica (parte blanca del ojo) ?: ")
  Fotofobia=input ("¿Le molesta la luz? (Ya sea interior o exterior): ")
  diplopía=input ("¿Ha notado que ve doble?: ")
  nictalopía=input ("¿En la noche siente que ve menos?: ")
  secreción=input ("¿Se le forman lagañas?: ")
  Epifora=input ("¿Le lagrimean los ojos?: ")
  Astenopia=input ("¿Tiene fatiga visual?: ")
  Vision_Distorcionada=input ("¿Ha notado que ve borroso o no les agarra forma a las cosas?: ")
  Baja_Vision_Lejana=input ("¿Considera que no ve bien de lejos?: ")
```

```
  Baja_Vision_Cercana=input ("¿Le cuesta enfocar objetos que tiene cerca?: ")
   Print ("De la siguiente lista, ¿Cual considera que es la que más realiza con
frecuencia")
  print ("- Bailar")
  print ("- Cocinar")
  print ("- construcción")
  print ("- Costura")
  print ("- Deportes")
  print ("- Dispositivos")
  print ("- Estudiar")
  print ("- Leer")
  print ("- música")
  print ("- Pintar")
  Actividades=input (": ")
  Print ("De la siguiente lista, ¿Cual se adapta más a su ocupación?")
  Print ("- Ama de casa")
  Print ("- Conductor")
  Print ("- Estudiante")
  Print ("- Pensionado")
  Print ("- Profesor")
  Print ("- Secretaria")
  Print ("- Trabajador")
  ocupación=input (": ")
   Anamnesis_Paciente= {'Edad': [Edad],"Prurito": [Prurito], 'Hiperemia': [Hiperemia],
"Fotofobia":        [Fotofobia],        "diplopía":[diplopía],"nictalopía":[nictalopía],
"secreción":[secreción],       "Epifora":[Epifora],       "Astenopia":[Astenopia],"visión
distorsionada":[Vision_Distorcionada], "Baja visión lejana":[Baja_Vision_Lejana], "Baja
visión        cercana":[Baja_Vision_Cercana],        'Activities'        :[Actividades],
"ocupación":[ocupación]}
  Anamnesis_Paciente=pd.DataFrame(Anamnesis_Paciente)
  Anamnesis_Paciente
else:
  print("ópticas Mega View le agradece su visita")
```

This is how the information its shown:

	Edad	Prurito	Hiperemia	Fotofobia	Diplopia	Nictalopia	Secrecion	Epifora	Astenopia	Vision distorcionada	Baja vision lejana	Baja vision cercana	Activities	Ocupacion
0	22	1	1	0	0	0	1	0	1	1	1	1	Dispositivos	Estudiante

Figure 17: Info given by the patient

We use the second part of the algorithm for decision trees and random forest, in which we would obtain the corresponding ametropia. We display the decision tree to how the dataset was performing. With a brief look and help of an optometrist, we conclude that the way it was created corresponds to the protocol they follow to diagnose their patients. To apply

decision trees, we had to change a little bit our dataset and later apply the function dummies to convert categorical columns. So, this is the dataset we use:

	Edad	Prurito	Hiperemia	Fotofobia	Diplopia	Nictalopia	Secrecion	Epifora	Astenopia
0	54.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0
1	19.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
2	52.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	66.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
4	21.0	1.0	1.0	1.0	0.0	1.0	0.0	1.0	1.0
...	...	...	...	...	...	...	...	...	...
297	28.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0
298	19.0	1.0	1.0	0.0	0.0	0.0	1.0	0.0	1.0
299	28.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0
300	22.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0	1.0
301	18.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0

Vision distorcionada	Baja vision lejana	Baja vision cercana	Activities	Ocupacion	Arbol
0.0	1.0	1.0	Dispositivos	Ama de casa	8.0
0.0	1.0	0.0	Dispositivos	Estudiante	7.0
0.0	1.0	1.0	Leer	Trabajador	8.0
1.0	1.0	0.0	Dispositivos	Trabajador	7.0
0.0	1.0	0.0	Leer	Ama de casa	7.0
...	...	...	...	...	...
0.0	1.0	0.0	Leer	Trabajador	7.0
0.0	0.0	0.0	Dispositivos	Estudiante	7.0
0.0	0.0	0.0	Dispositivos	Trabajador	7.0
1.0	1.0	1.0	Dispositivos	Estudiante	7.0
0.0	1.0	0.0	Dispositivos	Estudiante	7.0

Figure 18: Visualization of the tree dataset

The code we use was the following:

```
from sklearn.tree import DecisionTreeClassifier
import pandas as pd

datos=pd.read_excel("Trees.xlsx")
datos.to_csv("Trees.csv")
datos
```

```
#Convertimos nuestras columnas string a float
df1 = pd.get_dummies(data=datos, drop_first=True)
explicativas=df1.drop (columns="Arbol")
objetivo=df1.Arbol

#hyper parameter max depth
tree_clf = DecisionTreeClassifier (max_depth = 4)
tree_clf.fit(explicativas,objetivo)

print("tree classiefier configuration")
tree_clf
df1
from sklearn.tree import plot_tree
from sklearn import tree
import matplotlib.pyplot as plt
import graphviz
plt.figure(figsize=(50,15))
plot_tree
(decision_tree=tree_clf, feature_names=explicativas.columns, filled=True, fontsiz
e=8);
```

On the 2.1 part, rewarding random forest, in this part we had a clear vision of how the dataset was distributed, the results will be on the part of results. This is the code we use to apply random forest:

```
import numpy as np
import os

# To plot pretty figures
import matplotlib as mpl
import matplotlib.pyplot as plt

from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
explicativas_train, explicativas_test, objetivo_train, objetivo_test = train_test
_split(explicativas, objetivo, test_size=0.33, random_state=1, stratify=objetivo)

#Imprimos el tamaño de nuestras muestras, aqui podemos ver en que ametropia podem
os mejorar
print("Numero de muestras en objetivo:", np.bincount(objetivo))
print ("Numero de muestras en objetivo_train:", np.bincount(objetivo_train))
print ("Numero de muestras en objetivo_test:", np.bincount(objetivo_test))

tree_clf1 = DecisionTreeClassifier (max_depth = 14)
```

```
tree_clf1.fit(explicativas_train,objetivo_train)

print ("tree classifier configuration")
tree_clf1
from sklearn.metrics import accuracy_score
objetivo_pred_rf= tree_clf1.predict(explicativas_test)
print ("tree score", accuracy_score(objetivo_test, objetivo_pred_rf))
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier

sc=StandardScaler()
sc.fit(explicativas_train)

explicativas_train_std=sc.transform(explicativas_train)
explicativas_test_std=sc.transform(explicativas_test)

bosque= RandomForestClassifier
(n_estimators=500, criterion='entropy', max_features='sqrt', max_depth=10)
bosque.fit(explicativas_train_std, objetivo_train)

print ("Train Acurrac: %.5f" % bosque.
score(explicativas_train_std, objetivo_train))
print ("Test Acurracy: %.5f" % bosque.
score(explicativas_test_std, objetivo_test))
#Aqui vamos a realizar la predicción

Prediccion=df1.iloc[[301], : ]
Prediccion=Prediccion.drop(['Arbol'], axis=1)
print(Prediccion)
proba=bosque.predict_proba(Prediccion)
print('Probabilidad de todas las clases', proba)

pred=bosque.predict(Prediccion)
print("\n")
print(pred)
Enfermedad=pred [0]

if Enfermedad == 0:
  print ("\Usted podría estar padeciendo Astigmatismo")

if Enfermedad == 1:
  print ("\Usted podría estar padeciendo Astigmatismo con Presbicia")

if Enfermedad == 2 :
  print ("\Usted podría estar padeciendo Hipermetropía")
```

```
if Enfermedad == 3:
  print ("\nUsted podría estar padeciendo Hipermetropia con Astigmatismo")

if Enfermedad == 4:
  print
("\nUsted podria estar padeciendo Hipermetropia con Astigmatismo y Presbicia")

if Enfermedad == 5:
  print ("\Usted podría estar padeciendo Astigmatismo con Presbicia")

if Enfermedad == 6:
  print ("\Usted podría estar padeciendo miopía")

if Enfermedad == 7:
  print ("\n Usted podría estar padeciendo miopía con Astigmatismo")

if Enfermedad == 8:
  print ("\nUsted podria estar padeciendo Miopia con Astigmatismo y Presbicia")

if Enfermedad == 9:
  print ("\Usted podría estar padeciendo miopía con Presbicia")

if Enfermedad == 10:
  print ("\Usted podría estar padeciendo Presbicia")

if Enfermedad == 11:
  print ("\Usted podría estar padeciendo emétrope")

else:
  print ("\n")

from sklearn.metrics import confusion_matrix
from sklearn.metrics import ConfusionMatrixDisplay

objetivo_pred = bosque.predict(explicativas_test_std)
cm = confusion_matrix(objetivo_test, objetivo_pred)

cm_display = ConfusionMatrixDisplay (cm, display_labels=
['Astigmatismo','Astigmatismo con presbicie','Hipermetropia', 'Hipermetropia con
Astigmatismo', 'Hipermetropia con Astigmatismo y Presbicia', 'Hipermetropia con P
resbicia', "Miopia", "Miopia con Astigmatismo", "Miopia con Astigmatismo y Presbi
cia", "Miipia con Presbicia", "Presbicia", "Émetrope"])
cm_display.plot()
```

```
cm_display.ax_.set
(title='Desempeño', xlabel='Clases predichas', ylabel='Clases verdaderas')
```

In the third part of this algorithm, we wanted to be regression, we analyze the information and we found out that a linear regression would fit appropriately. Applying regression without a framework gives the same result as if we apply it with a framework, although I would change it to a framework when the beta comes out. I wanted a better interpretation and appliance of the regression, so that's why i made a single regression for each ametropia and lens treatment recommended, given us a large code, causing us to make 96 regressions. For this delivery I made the following code:

```
import matplotlib.pyplot as plt

from mpl_toolkits.mplot3d import Axes3D

import numpy as np

import pandas as pd


#Obtencion del numero de instancias

info=df[['Edad', 'Gasto']]

grad=info

n=info.shape

n=n[0]


#Obtenemos la edad del consumidor

ed=info['Edad'][299]

print(ed)

info=df[['Edad', 'Diagnostico','Tratamiento', 'Gasto']]

ed=float(ed)

print(info)

Enfermedad=7

Trat=1



if Enfermedad==0:

  if Trat==0:

    info=info[info['Diagnostico'] == 'Astigmatismo']

    info=info[info['Tratamiento'] == 'Bifocal']

    info['Edad']= info['Edad'].astype(float)
```

```
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
```

```
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx
```

```
 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
```

```
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
```

```
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==5:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Astigmatismo']
  info=info[info['Tratamiento'] == 'Progresivo']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
```

```
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo']
 info=info[info['Tratamiento'] == 'Antirreflejante']
```

```
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
```

```
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean
```

```
    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==1:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Astigmatismo con presvicie']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()
```

```
 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo con presvicie']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
```

```
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo con presvicie']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)
```

```
  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==3:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Astigmatismo con presvicie']
  info=info[info['Tratamiento'] == 'Fotocromatico']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
```

```
 x=info.Edad
 y=info.Gasto
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
```

```
info=info[info['Diagnostico'] == 'Astigmatismo con presvicie']
info=info[info['Tratamiento'] == 'Lente de contacto']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
```

```
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo con presvicie']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
```

```
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo presvicie']
 info=info[info['Tratamiento'] == 'Antirreflejante']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
```

```
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Astigmatismo presvicie']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
```

```
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==2:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Hipermetropia']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
```

```
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
```

```
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
```

```
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
```

```
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()
```

```
 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
```

```
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==6:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia']
  info=info[info['Tratamiento'] == 'Antirreflejante']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
```

```
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia']
```

```
info=info[info['Tratamiento'] == 'Polarizado']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
```

```
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==3:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
```

```
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
```

```
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
```

```
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
```

```
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
```

```
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
```

```
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Antirreflejante']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx
```

```
 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
```

```
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==4:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto
    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
```

```
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==1:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
  info=info[info['Tratamiento'] == 'BlueBlock']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
```

```
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Blanco']
```

```
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
```

```
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean
```

```
 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()
```

```
  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==5:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
  info=info[info['Tratamiento'] == 'Progresivo']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()
```

```
  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==6:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
  info=info[info['Tratamiento'] == 'Antirreflejante']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
  print(info)
  print(Tam)
```

```
 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con Astigmatismo y presvicie']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
```

```
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==5:
  if Trat==0:
```

```
info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
info=info[info['Tratamiento'] == 'Bifocal']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
```

```
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==1:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
  info=info[info['Tratamiento'] == 'BlueBlock']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx
```

```
 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
```

```
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==3:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
  info=info[info['Tratamiento'] == 'Fotocromatico']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
```

```
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==4:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
  info=info[info['Tratamiento'] == 'Lente de contacto']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
```

```
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
```

```
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
 info=info[info['Tratamiento'] == 'Antirreflejante']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
```

```
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Hipermetropia con presvicie']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
```

```
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==6:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Miopia']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()
```

```
  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==1:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Miopia']
  info=info[info['Tratamiento'] == 'BlueBlock']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
  print(info)
  print(Tam)
```

```
 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
```

```
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
```

```
info=info[info['Diagnostico'] == 'Miopia']
info=info[info['Tratamiento'] == 'Fotocromatico']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
```

```
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
```

```
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
```

```
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia']
 info=info[info['Tratamiento'] == 'Antirreflejante']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
```

```
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
```

```
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==7:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
```

```
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto
 Tam=info.shape
 print(info)
 print(Tam)



 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
```

```
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx
```

```
 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
```

```
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
```

```
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==5:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
  info=info[info['Tratamiento'] == 'Progresivo']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
```

```
  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==6:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
  info=info[info['Tratamiento'] == 'Antirreflejante']
```

```
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
```

```
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean
```

```
    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==8:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()
```

```
 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
```

```
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)
```

```
 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
```

```
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')

if Trat==4:
```

```
info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
info=info[info['Tratamiento'] == 'Lente de contacto']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
```

```
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==5:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
  info=info[info['Tratamiento'] == 'Progresivo']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
```

```
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
 info=info[info['Tratamiento'] == 'Antirreflejante']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
```

```
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Astigmatismo y Presbicia']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
```

```
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==9:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Miopia con Presbicia']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
```

```
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Presbicia']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
```

```
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Presbicia']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
```

```
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Presbicia']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
```

```
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Presbicia']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()
```

```
  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==5:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Miopia con Presbicia']
  info=info[info['Tratamiento'] == 'Progresivo']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto

  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
```

```
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==6:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Miopia con Presbicia']
  info=info[info['Tratamiento'] == 'Antirreflejante']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
```

```
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Miopia con Presbicia']
```

```
info=info[info['Tratamiento'] == 'Polarizado']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
```

```
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==10:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Presbicia']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
    #CALCULAMOS SSxx
    xmean = x.mean()
    info['diffx']=xmean-x
    info['diffx_squared']=info.diffx**2
    SSxx=info.diffx_squared.sum()

    #CALCULAMOS SSxy
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
```

```
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Presbicia']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
```

```
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Presbicia']
 info=info[info['Tratamiento'] == 'Blanco']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
```

```
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Presbicia']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
```

```
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
  info['diffx_squared']=info.diffx**2
  SSxx=info.diffx_squared.sum()

  #CALCULAMOS SSxy
  ymean=y.mean()
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==4:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Presbicia']
  info=info[info['Tratamiento'] == 'Lente de contacto']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
```

```
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto

Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

```
if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Presbicia']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
```

```
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==6:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Presbicia']
 info=info[info['Tratamiento'] == 'Antirreflejante']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx
```

```
 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Presbicia']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto

 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
```

```
    ymean=y.mean()
    info['diffy']=ymean-y
    SSxy=(info.diffx*info.diffy).sum()

    #CALCULAMOS M
    m=SSxy/SSxx

    #CALCULAMOS B
    b=ymean - m*xmean

    #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
    predict=m*ed+b
    predict=round(predict,2)
    print("\nUsted pagara alrededor de: ", predict)
    print("\n")
    plt.scatter(x,y)
    plt.plot(x, m*x+b, 'r')

if Enfermedad==11:
  if Trat==0:
    info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
    info=info[info['Diagnostico'] == 'Émetrope']
    info=info[info['Tratamiento'] == 'Bifocal']
    info=pd.DataFrame(info)
    info['Edad']= info['Edad'].astype(float)
    info['Gasto']= info['Gasto'].astype(float)
    x=info.Edad
    y=info.Gasto

    Tam=info.shape
    print(info)
    print(Tam)

    #Comenzamos con la regresion lineal
```

```
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==1:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Émetrope']
 info=info[info['Tratamiento'] == 'BlueBlock']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto
```

```
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==2:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Émetrope']
```

```
info=info[info['Tratamiento'] == 'Blanco']
info=pd.DataFrame(info)
info['Edad']= info['Edad'].astype(float)
info['Gasto']= info['Gasto'].astype(float)
x=info.Edad
y=info.Gasto
Tam=info.shape
print(info)
print(Tam)

#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
```

```
 plt.plot(x, m*x+b, 'r')

if Trat==3:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Émetrope']
 info=info[info['Tratamiento'] == 'Fotocromatico']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
```

```
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==4:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Émetrope']
 info=info[info['Tratamiento'] == 'Lente de contacto']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx
```

```
 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==5:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Émetrope']
 info=info[info['Tratamiento'] == 'Progresivo']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto
 Tam=info.shape
 print(info)
 print(Tam)

 #Comenzamos con la regresion lineal
 #CALCULAMOS SSxx
 xmean = x.mean()
 info['diffx']=xmean-x
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
```

```
  info['diffy']=ymean-y
  SSxy=(info.diffx*info.diffy).sum()

  #CALCULAMOS M
  m=SSxy/SSxx

  #CALCULAMOS B
  b=ymean - m*xmean

  #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
  predict=m*ed+b
  predict=round(predict,2)
  print("\nUsted pagara alrededor de: ", predict)
  print("\n")
  plt.scatter(x,y)
  plt.plot(x, m*x+b, 'r')

if Trat==6:
  info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
  info=info[info['Diagnostico'] == 'Émetrope']
  info=info[info['Tratamiento'] == 'Antirreflejante']
  info=pd.DataFrame(info)
  info['Edad']= info['Edad'].astype(float)
  info['Gasto']= info['Gasto'].astype(float)
  x=info.Edad
  y=info.Gasto
  Tam=info.shape
  print(info)
  print(Tam)

  #Comenzamos con la regresion lineal
  #CALCULAMOS SSxx
  xmean = x.mean()
  info['diffx']=xmean-x
```

```
 info['diffx_squared']=info.diffx**2
 SSxx=info.diffx_squared.sum()

 #CALCULAMOS SSxy
 ymean=y.mean()
 info['diffy']=ymean-y
 SSxy=(info.diffx*info.diffy).sum()

 #CALCULAMOS M
 m=SSxy/SSxx

 #CALCULAMOS B
 b=ymean - m*xmean

 #LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
 predict=m*ed+b
 predict=round(predict,2)
 print("\nUsted pagara alrededor de: ", predict)
 print("\n")
 plt.scatter(x,y)
 plt.plot(x, m*x+b, 'r')

if Trat==7:
 info=df[['Edad','Diagnostico','Tratamiento', 'Gasto']]
 info=info[info['Diagnostico'] == 'Émetrope']
 info=info[info['Tratamiento'] == 'Polarizado']
 info=pd.DataFrame(info)
 info['Edad']= info['Edad'].astype(float)
 info['Gasto']= info['Gasto'].astype(float)
 x=info.Edad
 y=info.Gasto
 Tam=info.shape
 print(info)
 print(Tam)
```

```
#Comenzamos con la regresion lineal
#CALCULAMOS SSxx
xmean = x.mean()
info['diffx']=xmean-x
info['diffx_squared']=info.diffx**2
SSxx=info.diffx_squared.sum()

#CALCULAMOS SSxy
ymean=y.mean()
info['diffy']=ymean-y
SSxy=(info.diffx*info.diffy).sum()

#CALCULAMOS M
m=SSxy/SSxx

#CALCULAMOS B
b=ymean - m*xmean

#LA PREDICCION DEL PASIENTE ES LA SIGUIENTE
predict=m*ed+b
predict=round(predict,2)
print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

In part 3.1 we apply gradient descent and I use the following code:

```
import NumPy
x=info[['Edad']]
x=x['Edad'].tolist()
x
```

```
y=info[['Gasto']]
y=list(y['Gasto'])
y
params = [0,0]
samples = x
y = y
__errors__= [];
def h(params, sample):
  acum = 0
  for i in range(len(params)):
    acum = acum + params[i]*sample[i]
  return acum;
def show_errors(params, samples,y):
  global __errors__
  error_acum =0
  print("transposed samples")
  print(samples)
  for i in range(len(samples)):
    hyp = h(params,samples[i])
    print( "hyp  %f  y %f " % (hyp,  y[i]))
    error=hyp-y[i]
    error_acum=+error**2 # this error is the original cost function, (the one use
d to make updates in GD is the derivated verssion of this formula)
  mean_error_param=error_acum/len(samples)
  __errors__.append(mean_error_param)
def GD(params, samples, y, alfa):
  temp = list(params)
  general_error=0
  for j in range(len(params)):
    acum =0; error_acum=0
    for i in range(len(samples)):
      error = h(params,samples[i]) - y[i]
      acum = acum + error*samples[i][j]  #Sumatory part of the Gradient Descent f
ormula for linear Regression.
    temp[j] = params[j] -
 alfa*(1/len(samples))*acum  #Subtraction of original parameter value with learni
ng rate included.
  return temp
def scaling(samples):
  acum =0
  samples = numpy.asarray(samples).T.tolist()
  for i in range(1,len(samples)):
    for j in range(len(samples[i])):
      acum=+ samples[i][j]
    avg = acum/(len(samples[i]))
```

```
    max_val = max(samples[i])
    #print("avg %f" % avg)
    #print(max_val)
    for j in range(len(samples[i])):
      #print(samples[i][j])
      samples[i][j] = (samples[i][j] - avg)/max_val  #Mean scaling
  return numpy.asarray(samples).T.tolist()
alfa =0.44  #  learning rate
for i in range(len(samples)):
  if isinstance(samples[i], list):
    samples[i]=  [1]+samples[i]
  else:
    samples[i]=  [1,samples[i]]
print ("original samples:")
print (samples)
samples = scaling(samples)
print ("scaled samples:")
print (samples)
epochs = 0
while True:  #  run gradient descent until local minima is reached
  oldparams = list(params)
  print (params)
  params=GD(params, samples,y,alfa)
  show_errors(params, samples, y)  #only used to show errors, it is not used in c
alculation
  print (params)
  epochs = epochs + 1
  if(oldparams == params or epochs == 200):   #  local minima is found when there
 is no further improvement
    print ("samples:")
    print(samples)
    print ("final params:")
    print (params)
    break
import matplotlib.pyplot as plt  #use this to generate a graph of the errors/loss
 so we can see whats going on (diagnostics)
plt.plot(__errors__)
plt.show()
```

In the last part of the algorithm, I stablish it to be the output that will get the user, in this part we will know the ametropia, the treatment and the approx. cost. Basically, is a resume of all the outputs we have been given on the past codes That's why I had to generate the following code:

```
print('De acuerdo a su edad: ', ed)
```

```
if Enfermedad == 0:
  print("\nUsted podría estar padeciendo Astigmatismo")

if Enfermedad == 1:
  print("\nUsted podria estar padeciendo Astigmatismo con Presbicia")

if Enfermedad == 2:
  print("\nUsted podría estar padeciendo hipermetropía")

if Enfermedad == 3:
  print("\nUsted podria estar padeciendo Hipermetropia con Astigmatismo")

if Enfermedad == 4:
  print("\nUsted podria estar padeciendo Hipermetropia con Astigmatismo y Presbic
ia")

if Enfermedad == 5:
  print("\nUsted podria estar padeciendo Astigmatismo con Presbicia")

if Enfermedad == 6:
  print("\nUsted podría estar padeciendo miopía")

if Enfermedad == 7:
  print("\nUsted podria estar padeciendo Miopia con Astigmatismo")

if Enfermedad == 8:
  print("\nUsted podria estar padeciendo Miopia con Astigmatismo y Presbicia")

if Enfermedad == 9:
  print("\nUsted podría estar padeciendo miopía con Presbicia")

if Enfermedad == 10:
  print("\nUsted podría estar padeciendo Presbicia")

if Enfermedad == 11:
  print("\nUsted podría estar padeciendo emétrope")

if Trat== 0:
  print('\nEl tratamiento adecuado para usted es Bifocal')

if Trat== 1:
  print('\nEl tratamiento adecuado para usted es Blueblock')

if Trat== 2:
```

```
  print('\nEl tratamiento adecuado para usted es Blanco')

if Trat== 3:
  print('\nEl tratamiento adecuado para usted es fotocromático')

if Trat== 4:
  print('\nEl tratamiento adecuado para usted es Lente de contacto')

if Trat== 5:
  print('\nEl tratamiento adecuado para usted es Progresivo')

if Trat== 6:
  print('\nEl tratamiento adecuado para usted es Antireflejante')

if Trat== 7:
  print('\nEl tratamiento adecuado para usted es Polarizado')

print("\nUsted pagara alrededor de: ", predict)
print("\n")
plt.scatter(x,y)
plt.plot(x, m*x+b, 'r')
```

Results

To prevent this document to be that long, we are just going to analyze the result that our clients are going to receive. The results are going to include the age, the ametropia, the right lens treatment and the price of them. This is how they are going to be display:

```
De acuerdo a su edad:  28.0

Usted podria estar padeciendo Miopia con Astigmatismo

El tratamiento adecuado para usted es Blueblock

Usted pagara alrededor de:  1560.21


[<matplotlib.lines.Line2D at 0x7fd0c9eadc50>]
```

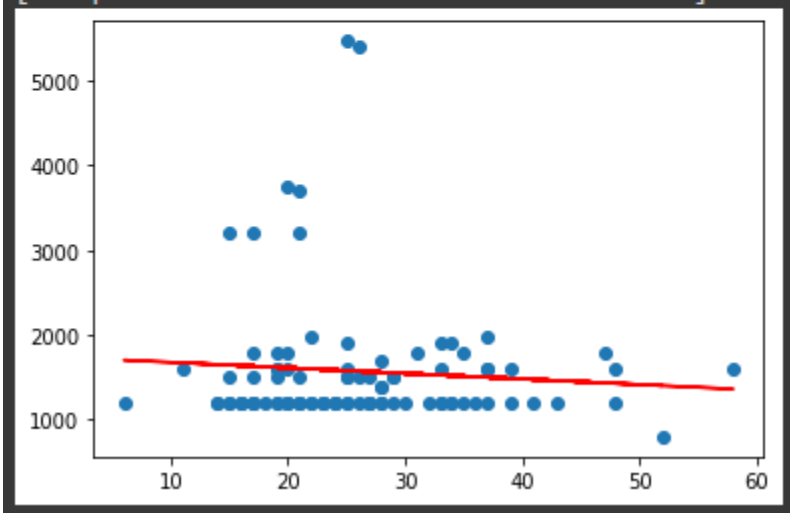

Fig 19: Clients results

The age is a value given by the patient, so we just print the value. The ametropia it's given by the random forest algorithm. In this part we have 3 things to have in mind. The first issue we found it was that the dataset wasn't perfectly balance, what do I mean with this, the number of instances between each class its different, which cause the algorithm to fail when trying to predict a class different of "Myopia with astigmatism, which is the class with the mayor number of instances. In fact, it's the most common ametropia. This is the number of instances of each class:

```
Numero de muestras en objetivo: [ 37   6   2  17  19   8  23 148  17   3  10  12]
Numero de muestras en objetivo_train: [25  4  1 11 13  5 16 99 11  2  7  8]
Numero de muestras en objetivo_test: [12  2  1  6  6  3  7 49  6  1  3  4]


DecisionTreeClassifier(max_depth=14)
```

Fig. 20: The first row corresponds to the number of instances in each class, the second and third one corresponds to the number of instance for each class ether training o test.

The second part is the error. I estimate that the cause of the low accuracy ratio we are getting, ratio of 0.33, same that will increase ones we run the forest part, changing to a

value of 0.55, its mainly cause by the number of instances that each class has. Let's look to the decision tree we obtain:

Decision tree with deep of 4

We notice that a tree with a deep of 4 gives a inconclud result, what do I mean with this, in deep 4, the tree gives a anfibios result, cause more than one ametropia it can be diagnost, this cause astigmatismo can be mix with myopia and hyperopia, so I found that a tree with 14 of deep, the tree can diagnost only one myopia.

Decision tree with deep of 14

The last part we need to analyze regarding trees and forest, I need to know the performance of them, and here is where I found out that the performance was critical, in which we can see that our algorithm achieve 45 predictions right, we can see as well that there is a confusion with the ametropias in whichever astigmatism its present. This may be because the symptoms are the same. I would consider the performance for astigmatism as acceptable.

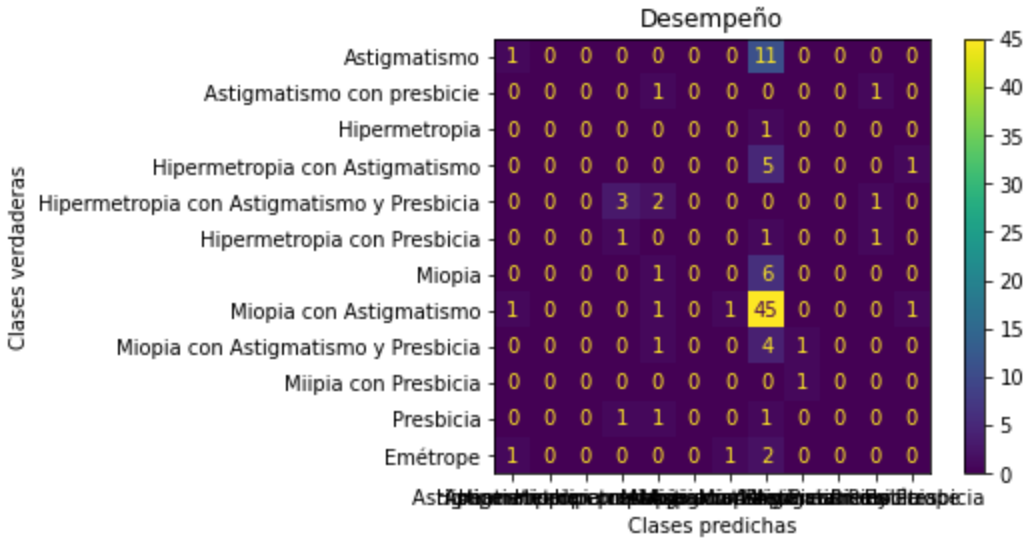

Fig. 21 : Confusion table

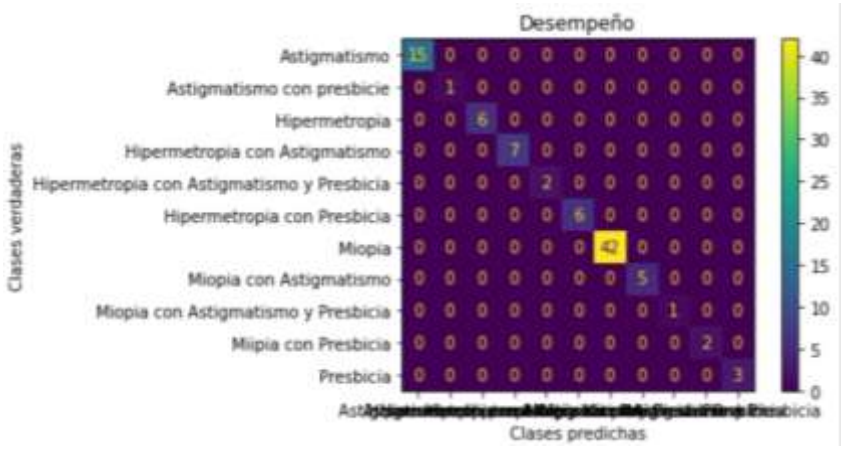

Figure 22: Confusion table obtain one time

For the last part of this algorithm, we need to analyze the results we obtain on the linear regression. Remember that we obtain the approximate price of the lens you will require. The client enters an age of 28, and the software recommended him a blueblock lens, diagnose with myopia with astigmatism, printing a price of 1560.21, which correspond to the price of blue block.

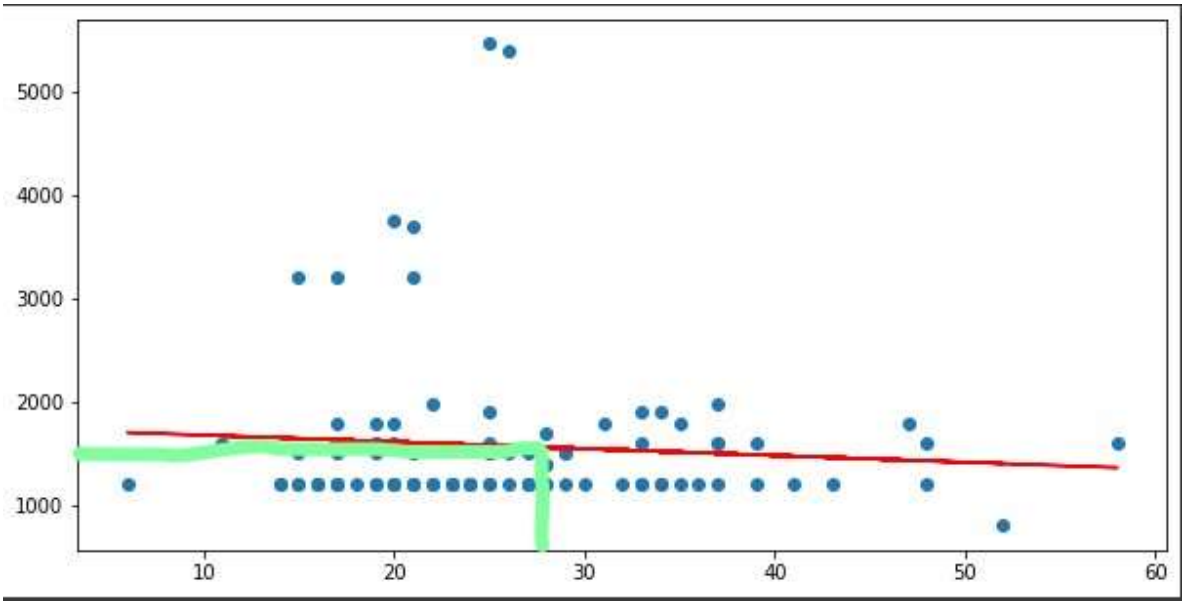

Fig. 23: Graph representing the prices versus the age.

The last part where we compare the algorithm in real life, this to solve the problematic that Opticas Mega View have been struggling with. Please check it out on the following video.

Video testing the algorithm

Conclusion

As an individual, AI has been the greatest invention on human history. The limits are almost non-existent. During this curse I learn how this type of the algorithm could help us solve our daily problems. That´s why I got inspired to think in a way we could reduce the time we invest in each client, without reducing the service that define Opticas Mega View. It was difficult for me to find a dataset that had the information I wanted or was functional for me. I didn't expect to have my dataset way too un-balanced. And obviously I didn't expect this to cause a diminution on the accuracy. Now that I have analyze in dep and related it with the short optometrist knowledge I have. As I explain in the abstract, each ametropia has their symptoms and if we pay attention, we can notice that most of the ametropias share the same symptoms. So, I estimate that the error is cause by the few instances that most classes have, the symptoms that the patients have manifested. My mom told me that the symptoms are kind of subjective. Cause blurry vision, double vision, diplopia is similar, and detecting the difference as a user its difficult, all of them can be cause by astigmatism, but myopia and hyperopia also present those symptoms. In the next anamnesis we will try to focus in describing clearly each patients how to distinguish each one.

Bibliography

Asthenopia - EyeWiki. (2021, 12 agosto). https://eyewiki.aao.org/Asthenopia

Clinic, A. E. (2020, 29 diciembre). *What Is Epiphora and What Causes It?* Armadale Eye Clinic.

https://armadale-eye.com.au/what-is-epiphora/

Delgado, A. (2019, 2 octubre). *What Causes Light Sensitivity?* Healthline.

https://www.healthline.com/health/photophobia

EMMÉTROPE - Definition and synonyms of emmétrope in the French dictionary. (s. f.).

https://educalingo.com/en/dic-fr/emmetrope

Institut Català de Retina. (2021, 10 noviembre). *Visión doble o diplopía. Causas y tratamiento*. ICR.

https://icrcat.com/enfermedades-oculares/vision-doble-diplopia/

Miranza. (2020, 28 octubre). *Astigmatismo ¿en qué consiste y cómo se produce?*

https://miranza.es/patologias/astigmatismo/

Night Blindness (Nyctalopia): Definition, Causes & Symptoms. (s. f.). Cleveland Clinic.

https://my.clevelandclinic.org/health/symptoms/10118-night-blindness-nyctalopia

Pruritus: Causes & Treatments for Itchy Skin. (s. f.). Cleveland Clinic. https://my.clevelandclinic.org/health/diseases/11879-pruritus

¿Qué es la diplopía? ¿Cómo se produce? (2022, 25 marzo). https://fernandez-vega.com/blog/diplopia-como-se-produce/

Watson, S. (2018, 29 agosto). *Hyperemia*. Healthline. https://www.healthline.com/health/hyperemia

What Is Hyperemia? (2021, 17 junio). WebMD. https://www.webmd.com/a-to-z-guides/what-is-hyperemia

Whitaker, J. (2021, 7 julio). *Astigmatism*. Eyecare Optical Knoxville. https://www.eyecareoptical-knoxville.com/astigmatism/

Your Skin, Pruritus, and Itching. (2006, 18 enero). WebMD. https://www.webmd.com/skin-problems-and-treatments/guide/skin-conditions-pruritus

Fig 1. astigmatism - Google Zoeken. (s. f.-a). https://www.google.com/search?q=astigmatism

Fig 2. astigmatism - Google Zoeken. (s. f.-b). https://www.google.com/search?q=astigmatism

Fig 3. miopia - Google Zoeken. (s. f.). https://www.google.com/search?q=miopia

Fig 4. Farsightedness - Google Zoeken. (s. f.). https://www.google.com/search?q=Farsightedness

Fig 5. Healthy eye - Google Zoeken. (s. f.). https://www.google.com/search?q=miopia

Fig 6. ocular Pruritus - Google Zoeken. (s. f.). https://www.google.com/search?q=ocular+Pruritus

Fig 7. Freepik. (2021, 20 diciembre). *Irritación de ojo. konyuktevitis, queratitis, alergias, uveítis. Vector Premium*. https://www.freepik.es/vector-premium/irritacion-ojo-konyuktevitis-queratitis-alergias-uveitis_21556890.htm

Fig 8. photophobia - Google Zoeken. (s. f.). https://www.google.com/search?q=photophobia

Fig 9. Diplopia - Google Zoeken. (s. f.). https://www.google.com/search?q=Diplopia

Fig 10. Nyctalopia - Google Zoeken. (s. f.). https://www.google.com/search?q=Nyctalopia

Fig 11. ocular Secretion - Google Zoeken. (s. f.). https://www.google.com/search?q =ocular+ Secretion

Fig 12. Epiphora - Google Zoeken. (s. f.). https://www.google.com/search?q=Epiphora

Fig 13. ocular asthenopia - Google Zoeken. (s. f.). https://www.google.com/search?q=ocular+asthenopia