TP :3 MOCA LU2IN021

Lazrak Ali 28605235

Exercice1 :

Moyenne des X:44.22

Moyenne des Y:804.93

Sxy:1193020.73

Sxx:64666.67

b:18.45

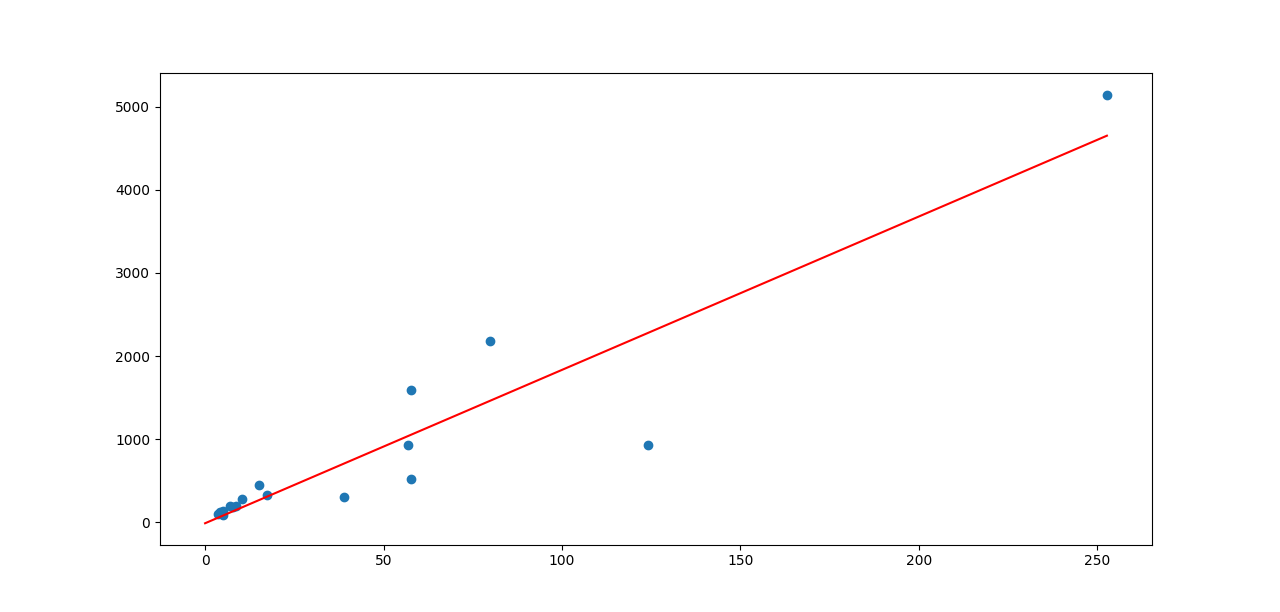
a:-10.94

Estimation de la consommation du Brésil pour 160 millions d'hab:2940.86

Coefficient de corrélation lineaire:0.93

Coefficient de détermination :0.87

Résidus normalisés [0.040399047585775026, 0.1019313272344398, 0.20745121025594102, 0.11787214951365026, 0.02358757401863383, -0.2361473811803882, 1.5312869280944008, 0.09094468518606168, -1.1196958084107187, -2.8246477703926898, 0.3719632113546402, 0.10809510162582239, -0.8592327589075968, 0.11025528514534591, 0.16504529417429079, 1.1362281751046266, 1.0346637295977628]



1.2) On peut en conclure que visuellement est numériquement que la droite de régression linéaire est pertinente.

1.3) Estimation de la consommation du Brésil pour 160 millions d'hab:2940.86 au millions de dollars

1.4)

Coefficient de corrélation lineaire:0.93

Coefficient de détermination :0.87

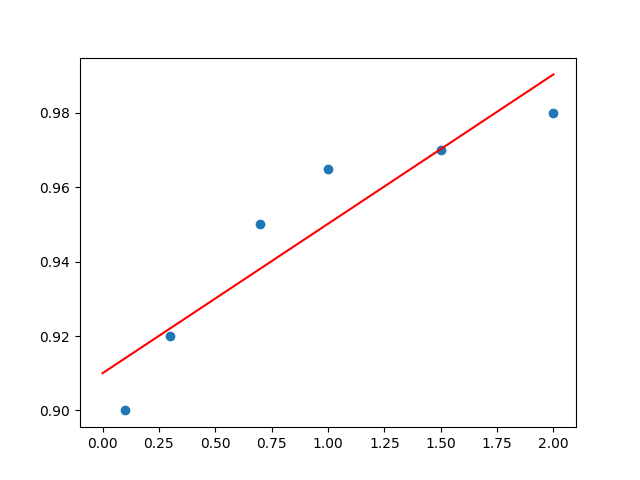
Résidus normalisés [0.040399047585775026, 0.1019313272344398, 0.20745121025594102, 0.11787214951365026, 0.02358757401863383, -0.2361473811803882, 1.5312869280944008, 0.09094468518606168, -1.1196958084107187, -2.8246477703926898, 0.3719632113546402, 0.10809510162582239, -0.8592327589075968, 0.11025528514534591, 0.16504529417429079, 1.1362281751046266, 1.0346637295977628]

Le coefficient de corrélation et de détermination confirme le fais que la droite de régression est pertinente, de plus les résidus normalisés quant on une justesse supérieure a 95%.

1.5)

Pour affiner nos analyses il est préférable de prendre un sous ensemble sans le japon est la France.

Exercice 2 :



Moyenne des X :0.93

Moyenne des Y :0.95

Sxy:0.10

Sxx:2.61

b:0.04

a:0.91

Coefficient de corrélation lineaire:0.93

Coefficient de détermination :0.86

Résidus normalisés [-1.0841259198583642, -0.1588209946289394, 0.9183996645933697, 1.1462732655827048, -0.020715781908130743, -0.8010102337807005]

2.1) Les résidus est les coefficient (corrélation et normalisation) nous indique que la droite de régression est pertinente).

CODE EXERCICE 1 :

import matplotlib.pyplot as plt

import numpy as np

import math

file=open("./confiserie.csv",'r')

texte=file.readlines()

file.close()

x=[]

y=[]

pays=[]

for f in texte:

    a=f.split(',')

    pays.append(a[0])

    x.append(float(a[1]))

    y.append(float(a[2]))

#equation de la droite linéaire est agale a y\*=a+bx

    #a = moyenneY-b\*moyenneX

    #b= Sxy/Sxx

#moyenne des x

moyenneX=0

i=0

for j in x:

    moyenneX+=j

    i=i+1

moyenneX=moyenneX/i

moyenneY=0

#moyenne des y

i=0

for j in y:

    moyenneY+=j

    i=i+1

moyenneY=moyenneY/i

#Sxy

Sxy=0

for p in range(len(x)):

    Sxy+=(x[p]-moyenneX)\*(y[p]-moyenneY)

#Sxx

Sxx=0

for j in x:

    Sxx+=((j-moyenneX)\*(j-moyenneX))

#Syy

Syy=0

for j in y:

    Syy+=((j-moyenneY)\*(j-moyenneY))

#b

b=Sxy/Sxx

#a

a=moyenneY-b\*moyenneX

#Estimation de consomation du Bresil en 1991 (160m hab)

ConsBrs=a+b\*160

#Coefficient de correlation lineaire

r=Sxy/math.sqrt(Sxx\*Syy)

#Coefficient de determination

R2=r\*r

#y\*

Yetoile=[]

for i in x:

    Yetoile.append(a+b\*i)

#SR erreur quadratique

SR=0

for i in range(len(y)):

    SR+=pow((y[i]-Yetoile[i]),2)

#Racine de l'estimateur sans biais

Resb=math.sqrt(SR/(len(x)-2))

#Residus normalisé

Rn=[]

for i in range(len(x)):

    Rn.append((y[i]-Yetoile[i])/Resb)

#-------------------affichage des differentes valeurs

print("moyenne des X:{0:.2f}".format(moyenneX))

print("moyenne des Y:{0:.2f}".format(moyenneY))

print("Sxy:{0:.2f}".format(Sxy))

print("Sxx:{0:.2f}".format(Sxx))

print("b:{0:.2f}".format(b))

print("a:{0:.2f}".format(a))

print("Estimation de la consomation du Bresil pour 160 milions d'hab:{0:.2f}".format(ConsBrs))

print("Coefficient de corrélation lineaire:{0:.2f}".format(r))

print("Coefficient de determination :{0:.2f}".format(R2))

print("Residus normalisés",Rn)

#--------------------affichage du graphe

xx=np.linspace(0,max(x),100)

yy=a+b\*xx

plt.plot(xx,yy,'-r')

plt.scatter(x,y)

plt.show()

CODE EXERCICE 2 :

import matplotlib.pyplot as plt

import numpy as np

import math

file=open("./reactions.csv",'r')

texte=file.readlines()

file.close()

x=[]

y=[]

for f in texte:

    a=f.split(',')

    x.append(float(a[1]))

    y.append(float(a[2]))

#equation de la droite linéaire est agale a y\*=a+bx

    #a = moyenneY-b\*moyenneX

    #b= Sxy/Sxx

#moyenne des x

moyenneX=0

i=0

for j in x:

    moyenneX+=j

    i=i+1

moyenneX=moyenneX/i

moyenneY=0

#moyenne des y

i=0

for j in y:

    moyenneY+=j

    i=i+1

moyenneY=moyenneY/i

#Sxy

Sxy=0

for p in range(len(x)):

    Sxy+=(x[p]-moyenneX)\*(y[p]-moyenneY)

#Sxx

Sxx=0

for j in x:

    Sxx+=((j-moyenneX)\*(j-moyenneX))

#Syy

Syy=0

for j in y:

    Syy+=((j-moyenneY)\*(j-moyenneY))

#b

b=Sxy/Sxx

#a

a=moyenneY-b\*moyenneX

#Coefficient de correlation lineaire

r=Sxy/math.sqrt(Sxx\*Syy)

#Coefficient de determination

R2=r\*r

#y\*

Yetoile=[]

for i in x:

    Yetoile.append(a+b\*i)

#SR erreur quadratique

SR=0

for i in range(len(y)):

    SR+=pow((y[i]-Yetoile[i]),2)

#Racine de l'estimateur sans biais

Resb=math.sqrt(SR/(len(x)-2))

#Residus normalisé

Rn=[]

for i in range(len(x)):

    Rn.append((y[i]-Yetoile[i])/Resb)

"""

Pour evaluer alpha et beta on peut faire un systeme avec les valeurs precedentes

    ainsi nous avons

        p+p\*e\*\*(alpha+beta\*t)=e\*\*(alpha+beta\*t)

        p=e\*\*(alpha+beta\*t)\*(1-p)

        p/(1-p)=e\*\*(alpha+beta\*t)

        ln(19)=alpha+beta\*0.7

        ln(0.965/0.035)=alpha+beta

        (7\*ln(0.965/0.035)-10\*ln(19))/-3=alpha

"""

alpha=(math.log(11.5)-3\*math.log(9))/-2

beta=(math.log(9)-alpha)/0.3

alpha1=(7\*math.log(0.965/0.035)-10\*math.log(19))/-3

beta1=(math.log(19)-alpha)/0.7

print(alpha)

print(beta)

print(alpha1)

print(beta1)

#-------------------affichage des differentes valeurs

print("moyenne des X:{0:.2f}".format(moyenneX))

print("moyenne des Y:{0:.2f}".format(moyenneY))

print("Sxy:{0:.2f}".format(Sxy))

print("Sxx:{0:.2f}".format(Sxx))

print("b:{0:.2f}".format(b))

print("a:{0:.2f}".format(a))

print("Coefficient de corrélation lineaire:{0:.2f}".format(r))

print("Coefficient de determination :{0:.2f}".format(R2))

print("Residus normalisés",Rn)

#--------------------affichage du graphe

"""

xx=np.linspace(0,max(x),100)

yy=a+b\*xx

plt.plot(xx,yy,'-r')

plt.scatter(x,y)

plt.show()

"""