

Nom, prénom : EL MOUTAQUI Hicham

Filière : Statistique Démographie

Principes d'analyse démographique :

Enoncé :

Exercice 6.

Deux groupes atteints par une maladie Z ont été suivis dans une clinique. Le groupe A a été atteint au mois de mars 2010 et le groupe B a été atteint au mois de septembre 2010. Ils étaient suivis depuis le moment où ils ont contracté la maladie. Au groupe A, un traitement A a été administré et au groupe B, un traitement B a été administré.

Nous savons que la guérison ne peut avoir lieu que durant la première année où le malade ait contracté la maladie. Aucun phénomène ne peut avoir lieu durant l'année (ni décès, ni autres phénomènes).

L'effectif du groupe A au départ est 2350 et l'effectif du groupe B est 4650. On donne ci-dessous les nombres de guérisons pour chacun des deux groupes de malades et ce pour chacun des mois de suivi au sein de la clinique.

Nombre de mois révolus passés dans la clinique	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)
0	120	131
1	140	172
2	157	226
3	162	297
4	163	319
5	169	323
6	163	333
7	161	323
8	150	321
9	114	311
10	87	277
11	66	236

- a- Calculez les quotients de guérison pour chacun des deux groupes A et B
- b- Elaborez les deux tables de guérison
- c- Calculez les intensités moyennes, calendriers moyens et variances de calendrier
- d- Comparez l'efficacité des deux traitements, illustrez votre méthodologie
- e- Comparez la guérison dans le premier trimestre de l'année et la guérison dans la seconde moitié de l'année, et ce pour chacun des deux groupes A et B. que peut-on conclure ?

b

Récupération des données du fichier excel dans le programme:

Entrée [168]:

```
import pandas as pd
import numpy as np
```

Importation du fichier excel

Entrée [169]:

```
df = pd.read_excel(
    'TD.Etat_pur (2).xlsx',
    sheet_name='Exo 6',
    skiprows=range(1),
    skipfooter=2,
)
```

Entrée [170]:

```
df
```

Out[170]:

	Nombre de mois révolus passés dans la clinique	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)	Effectif de malades (GA)	Quotients de guérison (GA)	Effectif de malades (GB)	Quotients de guérison (GB)	Quotients de guérison (GA).1	Probabilité de survie (GA)	Mi (GA)	Guérisons (GA)	1Li (GA)	Quotients de guérison (GB).1	Probabilité de survie (GB)	Mi (GB)
0	0.0	120.0	131	2350	NaN	4650.0	NaN	NaN	NaN	100000.0	NaN	NaN	NaN	NaN	100000.0
1	1.0	140.0	172	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	2.0	157.0	226	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	3.0	162.0	297	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	4.0	163.0	319	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	5.0	169.0	323	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
6	6.0	163.0	333	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
7	7.0	161.0	323	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
8	8.0	150.0	321	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9	9.0	114.0	311	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10	10.0	87.0	277	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
11	11.0	66.0	236	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
12	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
13	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
14	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

preprocessing

Entrée [171]:

```
df = df[['Nombre de mois révolus passés dans la clinique', 'Nombre de guéris (Traitement A)', 'Nombre de guéris (Traitement B)']]
```

Entrée [172]:

```
df = df.dropna(axis=0)
```

Entrée [173]:

```
df
```

Out[173]:

	Nombre de mois révolus passés dans la clinique	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)
0	0.0	120.0	131
1	1.0	140.0	172
2	2.0	157.0	226
3	3.0	162.0	297
4	4.0	163.0	319
5	5.0	169.0	323
6	6.0	163.0	333
7	7.0	161.0	323
8	8.0	150.0	321
9	9.0	114.0	311
10	10.0	87.0	277
11	11.0	66.0	236

Entrée [174]:

```
df.dtypes
```

Out[174]:

Nombre de mois révolus passés dans la clinique	float64
Nombre de guéris (Traitement A)	float64
Nombre de guéris (Traitement B)	object
dtype:	object

Entrée [175]:

```
df['Nombre de guéris (Traitement B)'] = df['Nombre de guéris (Traitement B)'].astype(int)
df['Nombre de guéris (Traitement A)'] = df['Nombre de guéris (Traitement A)'].astype(int)
```

b

Ajout des deux colonnes (effectifs des malades A/B) :

Entrée [176]:

df

Out[176]:

	Nombre de mois révolus passés dans la clinique	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)
0	0.0	120	131
1	1.0	140	172
2	2.0	157	226
3	3.0	162	297
4	4.0	163	319
5	5.0	169	323
6	6.0	163	333
7	7.0	161	323
8	8.0	150	321
9	9.0	114	311
10	10.0	87	277
11	11.0	66	236

Calcul des effectifs de malades des deux groupes A et B.

Entrée [177]:

```
effectifA = 2350
A = [effectifA]
for i in range(df['Nombre de guéris (Traitement A)'].shape[0]-1):
    A.append(effectifA-df['Nombre de guéris (Traitement A)'][i])
    effectifA = effectifA-df['Nombre de guéris (Traitement A)'][i]
effectifB = 4650
B = [effectifB]
for i in range(df['Nombre de guéris (Traitement B)'].shape[0]-1):
    B.append(effectifB-df['Nombre de guéris (Traitement B)'][i])
    effectifB = effectifB-df['Nombre de guéris (Traitement B)'][i]
```

Entrée [178]:

```
df['effectif des malades GA'] = A
df['effectif des malades GB'] = B
```

b

Entrée [179]:

df

Out[179]:

	Nombre de mois révolus passés dans la clinique	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)	effectif des malades GA	effectif des malades GB
0	0.0	120	131	2350	4650
1	1.0	140	172	2230	4519
2	2.0	157	226	2090	4347
3	3.0	162	297	1933	4121
4	4.0	163	319	1771	3824
5	5.0	169	323	1608	3505
6	6.0	163	333	1439	3182
7	7.0	161	323	1276	2849
8	8.0	150	321	1115	2526
9	9.0	114	311	965	2205
10	10.0	87	277	851	1894
11	11.0	66	236	764	1617

Calcul des quotients de guérison pour les deux groupes A et B

Entrée [180]:

```
df['Quotient de guérison GA'] = df['Nombre de guéris (Traitement A)']/df['effectif des malades GA']
df['Quotient de guérison GB'] = df['Nombre de guéris (Traitement B)']/df['effectif des malades GB']
```

Entrée [181]:

df

Out[181]:

	Nombre de mois révolus passés dans la clinique	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)	effectif des malades GA	effectif des malades GB	Quotient de guérison GA	Quotient de guérison GB
0	0.0	120	131	2350	4650	0.051064	0.028172
1	1.0	140	172	2230	4519	0.062780	0.038082
2	2.0	157	226	2090	4347	0.075120	0.051990
3	3.0	162	297	1933	4121	0.083808	0.072070
4	4.0	163	319	1771	3824	0.092038	0.083421
5	5.0	169	323	1608	3505	0.105100	0.092154
6	6.0	163	333	1439	3182	0.113273	0.104651
7	7.0	161	323	1276	2849	0.126176	0.113373
8	8.0	150	321	1115	2526	0.134529	0.127078
9	9.0	114	311	965	2205	0.118135	0.141043
10	10.0	87	277	851	1894	0.102233	0.146251
11	11.0	66	236	764	1617	0.086387	0.145949

Elaboration de la table d'extinction pour le groupe A :

b

Elaboration de la table de guérison Groupe A

```

Entrée [182]: table_guérison_GA = pd.DataFrame(columns=['quotients de guérison lqi (GA)', 'Probabilité de survie lpi (GA)', 'Mi (GA)', 'lLi (GA)'])

Entrée [183]: table_guérison_GA['quotients de guérison lqi (GA)'] = df['Quotient de guérison GA']
table_guérison_GA['Probabilité de survie lpi (GA)'] = 1 - df['Quotient de guérison GA']

Entrée [184]: a = 100000
G = [a]
for i in range(table_guérison_GA['Probabilité de survie lpi (GA)'].shape[0]-1):
    G.append(table_guérison_GA['Probabilité de survie lpi (GA)'][i]*a)
    a=table_guérison_GA['Probabilité de survie lpi (GA)'][i]*a

Entrée [185]: table_guérison_GA['Mi (GA)'] = G

Entrée [186]: '[i] + table_guérison_GA['Mi (GA)'][i+1] for i in range(table_guérison_GA.shape[0]-1)] + [table_guérison_GA['Mi (GA)'][11]*0.5]

Entrée [187]: table_guérison_GA['lLi (GA)'] = N

Entrée [188]: range(table_guérison_GA.shape[0]-1)] + [table_guérison_GA['quotients de guérison lqi (GA)'][11]*table_guérison_GA['Mi (GA)'][11]]

Entrée [189]: table_guérison_GA['Guérisons lGi (GA)'] = Q

Entrée [190]: table_guérison_GA

```

Out[190]:

	quotients de guérison lqi (GA)	Probabilité de survie lpi (GA)	Mi (GA)	lLi (GA)	Guérisons lGi (GA)
0	0.051064	0.948936	100000.000000	97446.808511	5106.382979
1	0.062780	0.937220	94893.617021	91914.893617	5957.448809
2	0.075120	0.924880	88936.170213	85595.744681	6680.851064
3	0.083808	0.916192	82255.319149	78808.510638	6883.617021
4	0.092038	0.907962	75361.702128	71893.617021	6936.170213
5	0.105100	0.894900	68425.531915	64829.787234	7191.489362
6	0.113273	0.886727	61234.042553	57765.957447	6936.170213
7	0.126176	0.873824	54297.872340	50872.340426	6851.063830
8	0.134529	0.865471	47446.808511	44255.319149	6382.978723
9	0.118135	0.881865	41063.829787	38638.297872	4851.063830
10	0.102233	0.897767	36212.765957	34361.702128	3702.127960
11	0.086387	0.913613	32510.838298	16255.319149	2808.510638

Elaboration de la table d'extinction pour le groupe B :

b

Elaboration de la table de guérison Groupe B

```

Entrée [191]: table_guérison_GB = pd.DataFrame(columns=['quotients de guérison 1qi (GB)', 'Probabilité de survie 1pi (GB)', 'Mi (GB)', '1Li (GB)'])

Entrée [192]: table_guérison_GB['quotients de guérison 1qi (GB)'] = df['Quotient de guérison GB']
table_guérison_GB['Probabilité de survie 1pi (GB)'] = 1 - df['Quotient de guérison GB']

Entrée [193]: a = 100000
G = [a]
for i in range(table_guérison_GB['Probabilité de survie 1pi (GB)'].shape[0]-1):
    G.append(table_guérison_GB['Probabilité de survie 1pi (GB)'][i]*a)
    a=table_guérison_GB['Probabilité de survie 1pi (GB)'][i]*a

Entrée [194]: table_guérison_GB['Mi (GB)'] = G

Entrée [195]: '[i] + table_guérison_GB['Mi (GB)'][i+1] for i in range(table_guérison_GB.shape[0]-1)] + [table_guérison_GB['Mi (GB)'][11]*0.5'

Entrée [196]: table_guérison_GB['1Li (GB)'] = N

Entrée [197]: 'rison_GB['Mi (GB)'][i] - table_guérison_GB['Mi (GB)'][i+1] for i in range(table_guérison_GB.shape[0]-1)] + [table_guérison_GB['Mi (GB)'][11]*0.5'

Entrée [198]: table_guérison_GB['Guérisons 1Gi (GB)'] = Q

Entrée [199]: table_guérison_GB

```

Out[199]:

	quotients de guérison 1qi (GB)	Probabilité de survie 1pi (GB)	Mi (GB)	1Li (GB)	Guérisons 1Gi (GB)
0	0.028172	0.971828	100000.000000	98591.397849	2817.204301
1	0.038082	0.961918	97182.795969	95333.333333	3698.924731
2	0.051990	0.948010	93483.870998	91053.763441	4890.215054
3	0.072070	0.927930	88823.655914	85430.107527	6387.096774
4	0.083421	0.916579	82236.559140	78808.451613	6890.215054
5	0.092154	0.907846	75376.344086	71903.225806	6948.236559
6	0.104851	0.895149	68430.107527	64849.402386	7101.290323
7	0.113373	0.886627	61268.817204	57795.969925	6948.236559
8	0.127078	0.872922	54322.580645	50870.967742	6903.225806
9	0.141043	0.858957	47419.354839	44075.268817	6888.172043
10	0.148251	0.853749	40731.182796	37752.888172	5956.989247
11	0.145949	0.854051	34774.193548	17387.096774	5075.268817

Calcul de la variance et du calendrier moyen :

b

Calcul du calendrier numérateur et de la variance numératrice pour les groupes A et B

Entrée [200]: `= pd.DataFrame(columns = ['calendrier (numér) (GA)', 'calendrier (numér) (GB)', 'variance (numér) (GA)', 'variance (numér) (GB)'])`

Calcul du calendrier moyen

Entrée [201]: `er (numér) (GA)' = [(i + 0.5)*table_guérison_GA['Guérisons 1G1 (GA)'][i] for i in range(table_guérison_GA.shape[0])]
er (numér) (GB)' = [(i + 0.5)*table_guérison_GB['Guérisons 1G1 (GB)'][i] for i in range(table_guérison_GB.shape[0])]
calen_vari['calendrier (numér) (GA)'] = (er (numér) (GA)).sum()/table_guérison_GA['Guérisons 1G1 (GA)'].sum()
calen_vari['calendrier (numér) (GB)'] = (er (numér) (GB)).sum()/table_guérison_GB['Guérisons 1G1 (GB)'].sum()
(numér) (GA)' = [(i+0.5 - calen_vari['calendrier (numér) (GA)'])**2*table_guérison_GA['Guérisons 1G1 (GA)'][i] for i in range(table_guérison_GA.shape[0])]
(numér) (GB)' = [(i+0.5 - calen_vari['calendrier (numér) (GB)'])**2*table_guérison_GB['Guérisons 1G1 (GB)'][i] for i in range(table_guérison_GB.shape[0])]`

Entrée [202]: `calen_vari`

Out[202]:

	calendrier (numér) (GA)	calendrier (numér) (GB)	variance (numér) (GA)	variance (numér) (GB)
0	2553.191489	1408.602151	130894.484206	99197.884759
1	8936.170213	5548.387097	96343.120142	90045.193708
2	16702.127880	12150.537634	62677.654753	75215.540773
3	24127.659574	22354.838710	29337.711836	54979.582455
4	31212.765957	30870.967742	7836.978933	25857.849731
5	39553.191489	38204.301075	28.501350	6058.614737
6	45085.106383	46548.387097	6090.340429	31.265757
7	51382.978723	52096.774194	25706.200136	7894.512071
8	54255.319149	58677.419355	55061.101497	29467.571838
9	46085.106383	63537.634409	75193.096249	62874.284472
10	38872.340426	62548.387097	90237.226515	96486.712352

Calcul du calendrier moyen pour A et B

Entrée [203]: `calendrier_moyen_A = calen_vari['calendrier (numér) (GA)'].sum()/table_guérison_GA['Guérisons 1G1 (GA)'].sum()
calendrier_moyen_B = calen_vari['calendrier (numér) (GB)'].sum()/table_guérison_GB['Guérisons 1G1 (GB)'].sum()`

Entrée [204]: `print('le calendrier moyen pour le groupe A est : ',calendrier_moyen_A)`

le calendrier moyen pour le groupe A est : 5.562953995157385

Entrée [205]: `print('le calendrier moyen pour le groupe B est : ',calendrier_moyen_B)`

le calendrier moyen pour le groupe B est : 6.433924747629243

Calcul de la variance du calendrier pour le groupe A et le groupe B

Entrée [206]: `variance_calendrier_A = calen_vari['variance (numér) (GA)'].sum()/table_guérison_GA['Guérisons 1G1 (GA)'].sum()
variance_calendrier_B = calen_vari['variance (numér) (GB)'].sum()/table_guérison_GB['Guérisons 1G1 (GB)'].sum()`

Entrée [207]: `print('la variance du calendrier pour le groupe A est : ',variance_calendrier_A)`

la variance du calendrier pour le groupe A est : 9.678845511280743

Entrée [208]: `print('la variance du calendrier pour le groupe B est : ',variance_calendrier_B)`

la variance du calendrier pour le groupe B est : 9.675846725447512

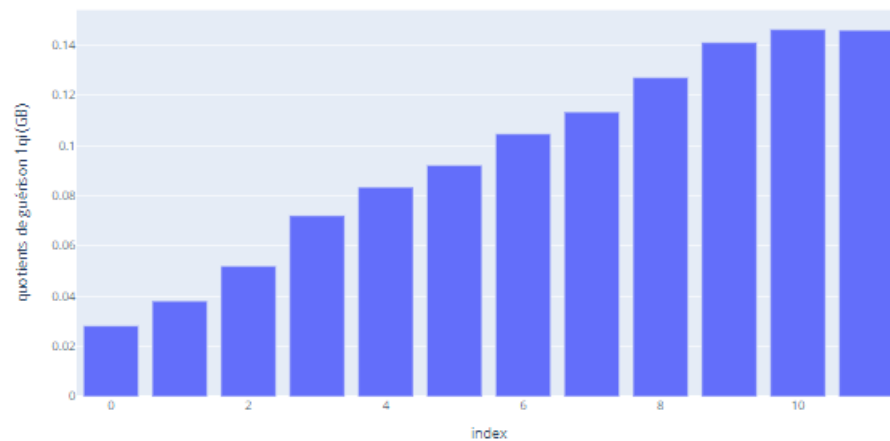
Entrée [209]: `!pip install plotly`

****Bar chart****

b

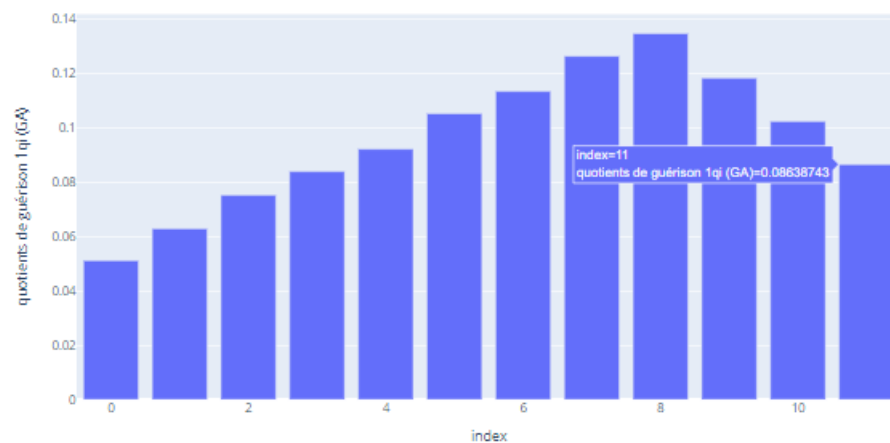
```
Entrée [210]: import plotly.express as ex
```

```
Entrée [211]: fig = ex.bar(table_guérison_GB, y="quotients de guérison lqi (GB)")  
fig.show()  
print('Le mode: ', table_guérison_GB['quotients de guérison lqi (GB)'].max())
```



Le mode: 0.14625131995776136

```
Entrée [212]: fig = ex.bar(table_guérison_GA, y="quotients de guérison lqi (GA)")  
fig.show()  
print('Le mode: ', table_guérison_GB['quotients de guérison lqi (GB)'].max())
```



Le mode: 0.14625131995776136

Interprétation des résultats :

b

Interprétation des résultats

Calendrier moyen: le traitement A est plus efficace étant donné que la durée moyenne de guérison est 0,87 mois de moins par rapport au traitement B

Entrée []:

Des visualisations :

des visualisations

```
Entrée [111]: df1 = df[['Nombre de guéris (Traitement A)', 'Nombre de guéris (Traitement B)']]
```

```
Entrée [114]: df1
```

```
Out[114]:
```

	Nombre de guéris (Traitement A)	Nombre de guéris (Traitement B)	fréquence relative de guérison groupe A	fréquence relative de guérison groupe B
0	120	131	0.072639	0.040073
1	140	172	0.084746	0.052615
2	157	226	0.095036	0.069134
3	162	297	0.098063	0.090853
4	163	319	0.098668	0.097583
5	169	323	0.102300	0.098807
6	163	333	0.098668	0.101866
7	161	323	0.097458	0.098807
8	150	321	0.090799	0.098195
9	114	311	0.069007	0.095136
10	87	277	0.052663	0.084735
11	66	236	0.039952	0.072193

```
Entrée [116]: df2 = df1[['fréquence relative de guérison groupe A', 'fréquence relative de guérison groupe B']]
```

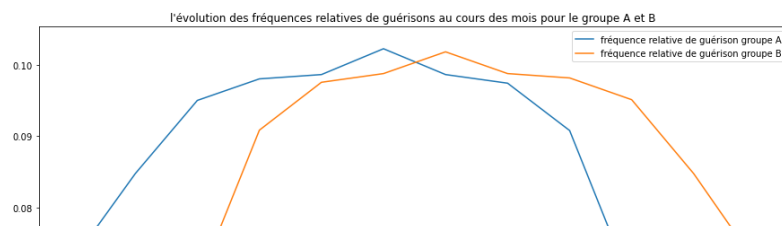
```
Entrée [118]: df2
```

```
Out[118]:
```

	fréquence relative de guérison groupe A	fréquence relative de guérison groupe B
0	0.072639	0.040073
1	0.084746	0.052615
2	0.095036	0.069134
3	0.098063	0.090853
4	0.098668	0.097583
5	0.102300	0.098807
6	0.098668	0.101866
7	0.097458	0.098807
8	0.090799	0.098195
9	0.069007	0.095136
10	0.052663	0.084735
11	0.039952	0.072193

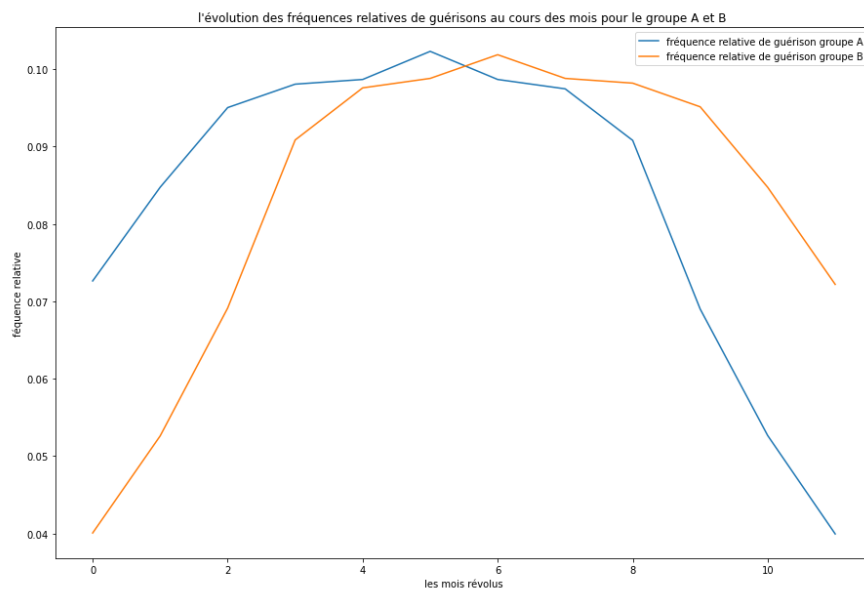
```
Entrée [119]: fig = df2.plot(kind = 'line', figsize = (15,10))  
fig.set_xlabel('les mois révolus')  
fig.set_ylabel('fréquence relative')  
fig.set_title("l'évolution des fréquences relatives de guérisons au cours des mois pour le groupe A et B ")
```

```
Out[119]: Text(0.5, 1.0, "l'évolution des fréquences relatives de guérisons au cours des mois pour le groupe A et B ")
```



b

plt.plot(x, y, 'o') # création des fréquences relatives de guérisons au cours des mois pour le groupe A et B



Entrée [128]:

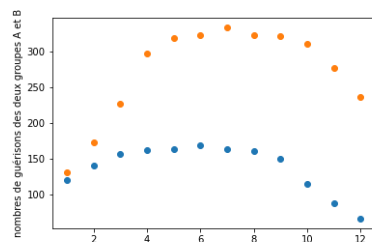
```
daf= df[['Nombre de mois révolus passés dans la clinique','Nombre de guéris (Traitement A)','Nombre de guéris (Traitement B)'] ]
```

Standardisation des guérisons pour pouvoir les comparer

Entrée [152]:

```
import matplotlib.pyplot as plt
plt.scatter(np.arange(1,13), daf[['Nombre de guéris (Traitement A)']-daf[['Nombre de guéris (Traitement A)']].mean()/daf[['Nombre de guéris (Traitement B)']-daf[['Nombre de guéris (Traitement B)']].mean()/daf[['Nombre de guéris (Traitement B)']].mean()/daf[['Nombre de guéris (Traitement B)']].mean()))
plt.ylabel('nombres de guérisons des deux groupes A et B', fontsize =10)
```

Out[152]: Text(0, 0.5, 'nombres de guérisons des deux groupes A et B')



La guérison chez le groupe A est plus importante que celle chez le groupe B.

Entrée []: