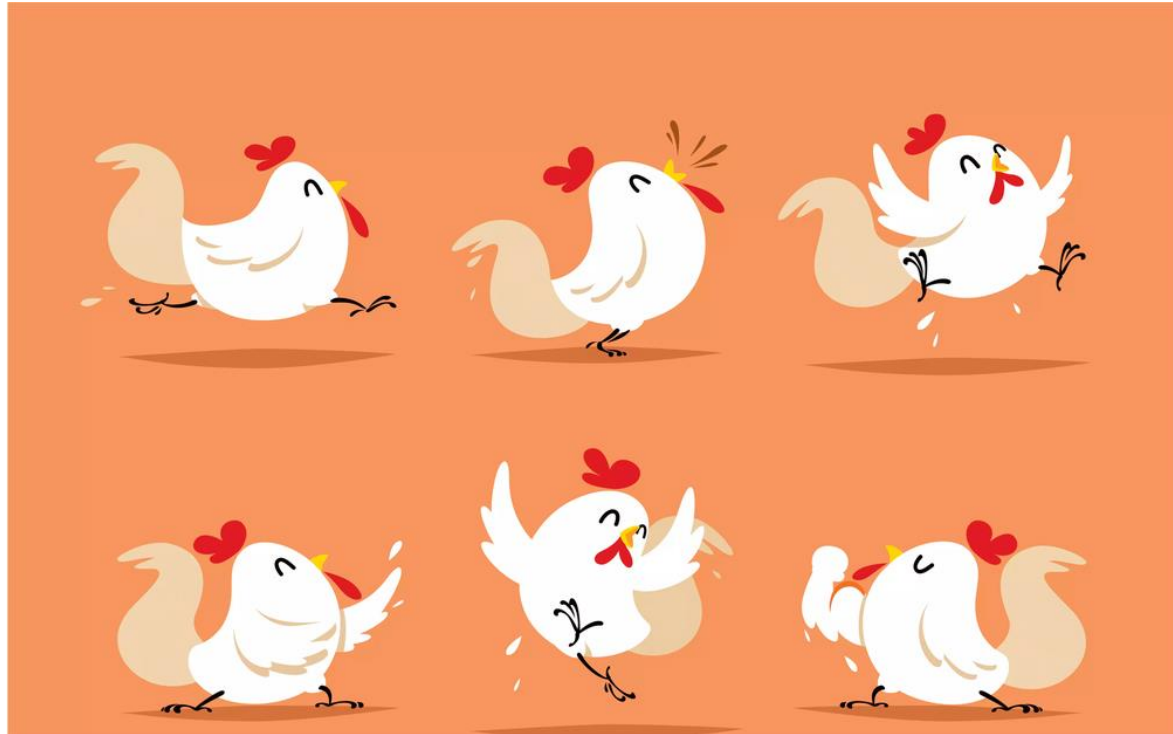


Projet 5 : Produisez une étude de marché



Introduction



Introduction

4 Variables :

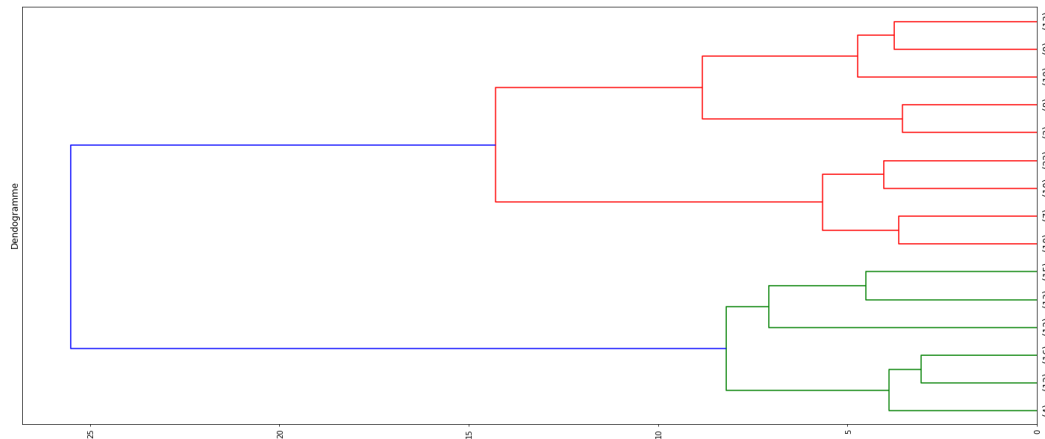
- Evolution de la population
- Proportion de protéines d'origine animale par rapport à la quantité totale de protéines dans la disponibilité alimentaire du pays
- Disponibilité alimentaire en protéines par habitant par jour
- Disponibilité alimentaire en calories par habitant par jour

Introduction

Classification hiérarchique



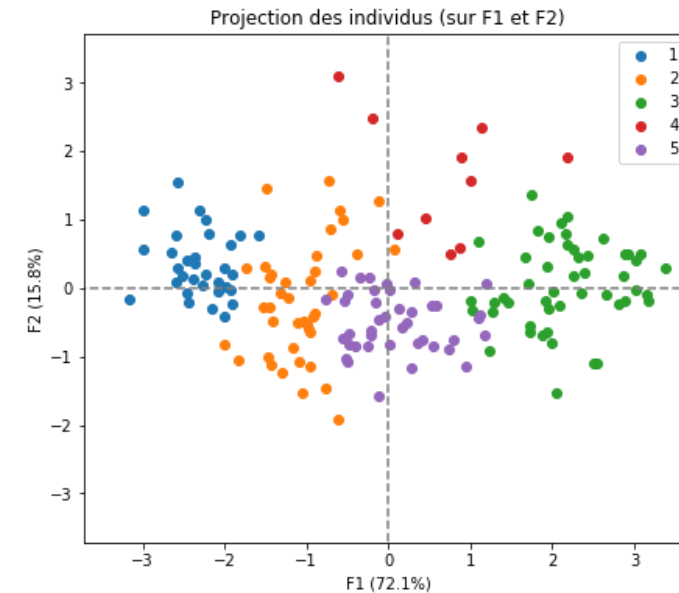
Partitionner les données



Analyse en composante principale (ACP)



Visualiser les partitions



I. Création du DataFrame

```
dispo_Kcal=dispo_Kcal.rename(columns={"Valeur":"dispo_Kcal"})
dispo_prot_total=dispo_prot_total.rename(columns={"Valeur":"dispo_prot_total"})

pop2010=pop[pop.Année==2010]
pop2010=pop2010.reset_index()
#print(pop2010[pop2010["Code Pays"]==151])
#Le pays "Antilles néerlandaises (ex)" est présent sur le df pop2010 mais pas sur pop2013

pop2010=pop2010[pop2010["Code Pays"]!=151]
pop2010=pop2010.reset_index()

pop2013=pop[pop.Année==2013]
pop2013=pop2013.reset_index()

ratio_pop={"ratio":((pop2013.Valeur*1000)/(pop2010.Valeur*1000)*100-100).tolist()}
df_ratio_pop=pd.DataFrame(ratio_pop)
```

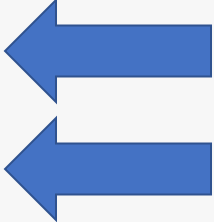
I. Création du DataFrame

```
data_4={"Pays":dispo_Kcal.Pays.tolist(),"dispo_Kcal":dispo_Kcal.dispo_Kcal.tolist(),"dispo_prot_total":dispo_prot_total.dispo_prot_total.tolist()}
df=pd.DataFrame(data_4)

print(df[df.Pays=="Oman"]) #Le pays Oman est un outlier
df=df[df.Pays!="Oman"]

print(df[df.Pays=="Soudan"]) #Le pays Soudan est un outlier
df=df[df.Pays!="Soudan"]

df=df.reset_index(drop=True)
df=df.set_index("Pays")
```



Le DataFrame est prêt à être analysé

II. Classification hiérarchique

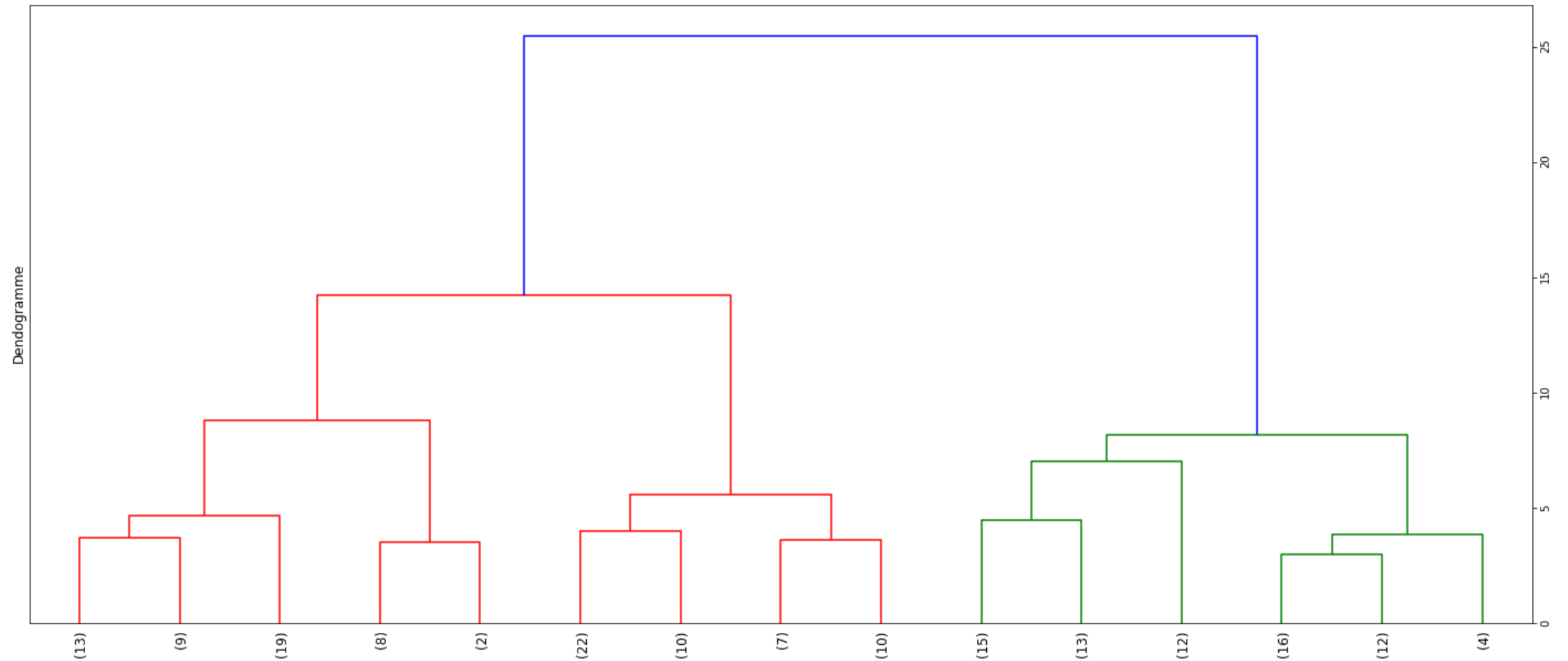


```
def plot_dendrogram(Z, names):  
    plt.figure(figsize=(10,25))  
    plt.title('Dendrogramme')  
    dendrogram(  
        Z,  
        labels = names,  
        truncate_mode='lastp',  
        p=15,  
    )  
    plt.show()  
  
# préparation des données pour le clustering  
X = df.values  
names = df.index  
  
# Centrage et Réduction  
std_scale = preprocessing.StandardScaler().fit(X)  
X_scaled = std_scale.transform(X)  
  
# Clustering hiérarchique  
Z = linkage(X_scaled, 'ward')  
  
# Affichage du dendrogramme  
plot_dendrogram(Z, names)
```



II. Classification hiérarchique

Découpage en 15 clusters



III. Analyse des partitions

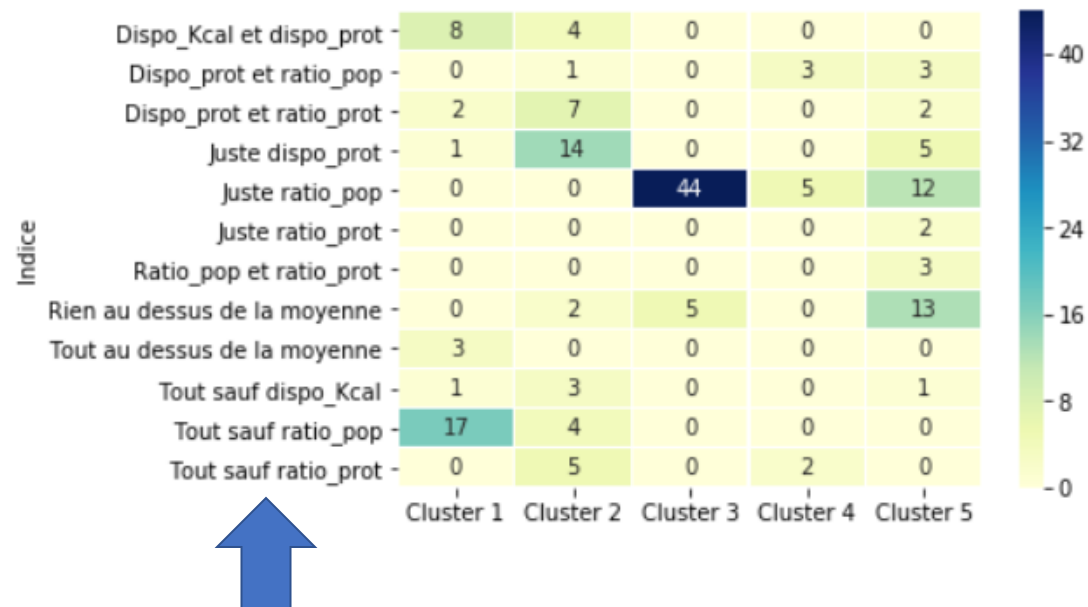
```
# Découpage du dendrogramme en 5 clusters
clusters = fcluster(Z, 5, criterion='maxclust')

# Comparaison des clusters trouvés

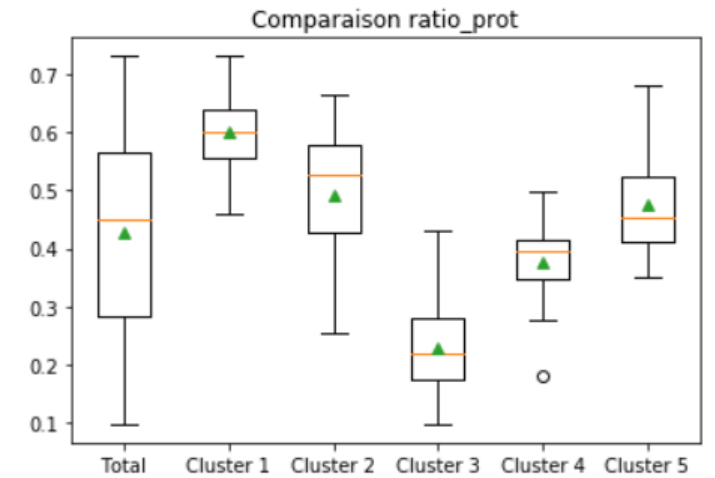
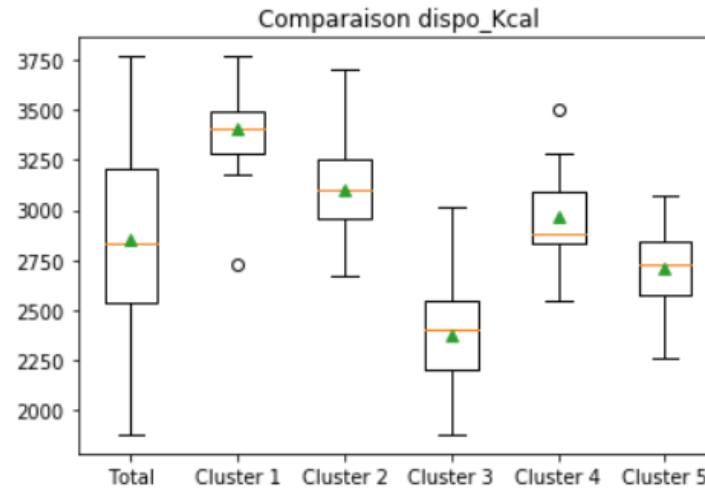
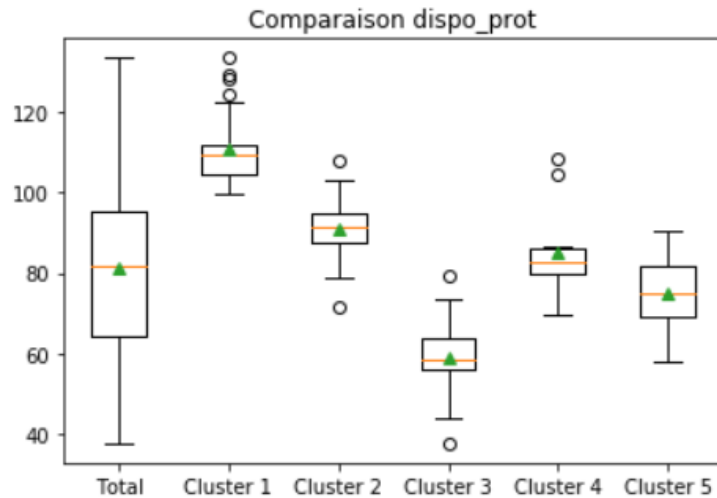
courses = pd.DataFrame({"cluster": clusters, "Pays":df.index, "Indice": indice.Indice})

sns.heatmap(courses.pivot_table(index="Indice" , columns="cluster", aggfunc=len, fill_value=0), annot=True, xticklabels=["Cluster 1", "Cluster 2", "Cluster 3", "Cluster 4", "Cluster 5"])

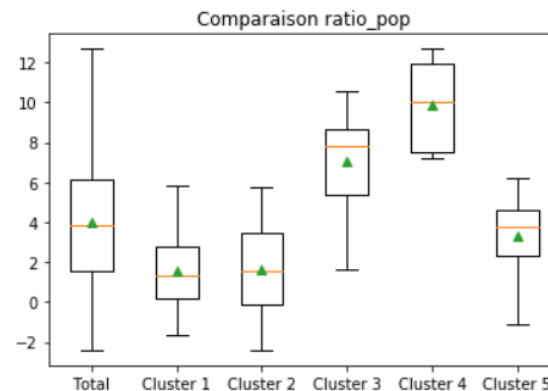
<matplotlib.axes._subplots.AxesSubplot at 0x42f88eeac8>
```



III. Analyse des partitions



Les variables dispo_prot, dispo_Kcal et ratio_prot en général plus haute pour le cluster 1.



Cependant, les clusters 1 et 2 sont les plus bas au niveau de la variable ratio_pop

IV. Analyse en composante principale

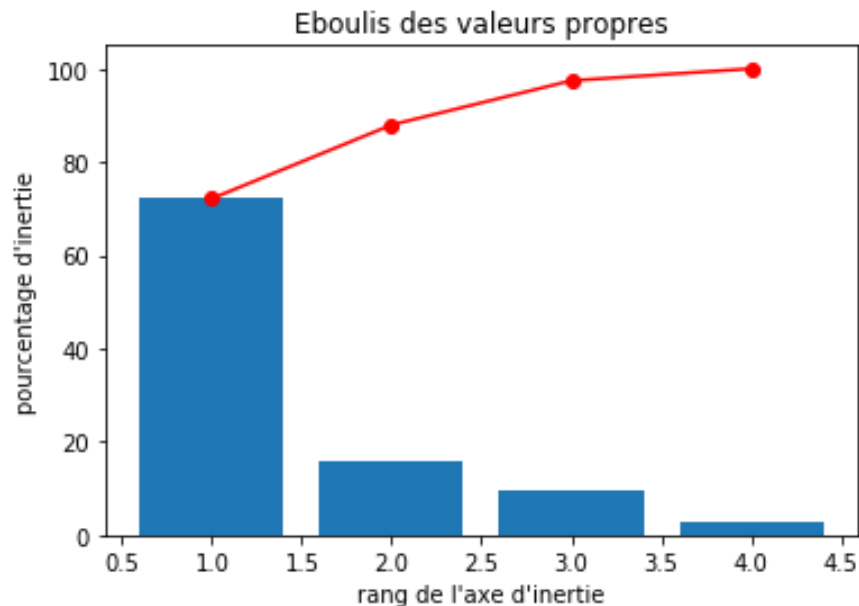
```
# choix du nombre de composantes à calculer
n_comp = 4

# import de l'échantillon
X = df.drop(columns=["Indice", "Pays", "cluster"]).values

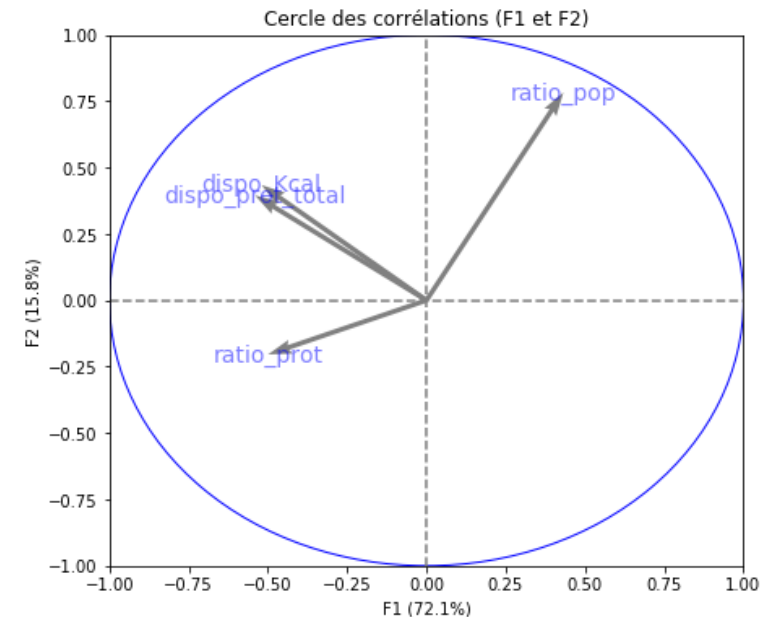
# Centrage & réduction
std_scale = preprocessing.StandardScaler().fit(X)
X_scaled = std_scale.transform(X)

# Calcul des composantes principales
pca = decomposition.PCA(n_components= n_comp)
pca.fit(X_scaled)
```

```
# Eboulis des valeurs propres
display_scee_plot(pca)
```

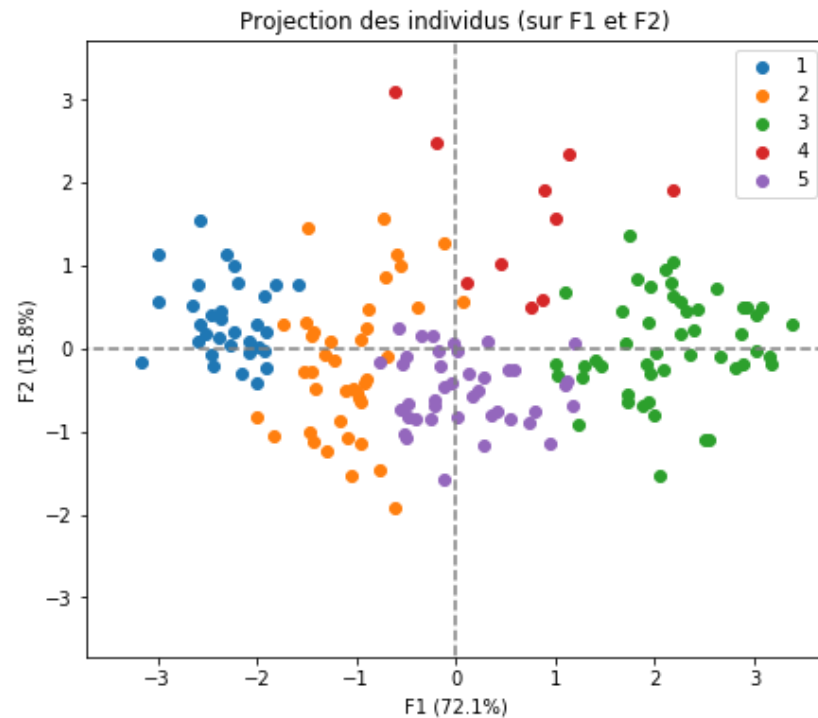


```
# Cercle des corrélations
pcs = pca.components_
display_circles(pcs, n_comp, pca, [(0,1)], labels = np.array(df.drop(columns=["Indice", "Pays"]).columns))
```

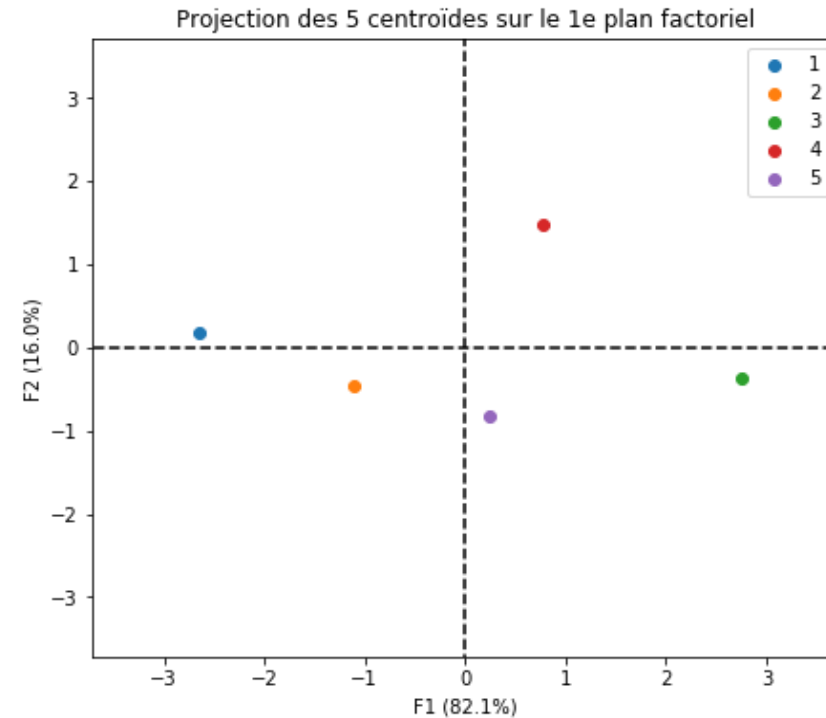


IV. Analyse en composante principale

```
# Projection des individus  
X_projected = pca.transform(X_scaled)  
display_factorial_planes(X_projected, n_comp, pca, [(0,1)], illustrative_var=df.cluster)
```



```
#Projection des centroïdes
```



V. Qualité de représentation

Contribution dans l_inertie totale	
id	
Zambie	11.905066
Islande	11.602863
Koweït	10.748100
Lituanie	10.561226
Chine - RAS de Hong-Kong	10.551092
Libéria	10.427880
Madagascar	10.053566
Ouganda	9.992069
Rwanda	9.587161
Mozambique	9.294506

Contribution dans l_inertie totale	
id	
Kirghizistan	0.050425
Turkménistan	0.210350
Viet Nam	0.244451
Ouzbékistan	0.282423
Afrique du Sud	0.384145
Pérou	0.393910
Mexique	0.413540
Fidji	0.458271
Kiribati	0.464336
Guyana	0.528963



COS2_1er_plan_factoriel			
id	COS2_F1	COS2_F2	
Eswatini	0.998246	0.879300	0.118947
France	0.998091	0.972659	0.025433
Saint-Vincent-et-les Grenadines	0.997698	0.631653	0.366045
Norvège	0.997161	0.883915	0.113246
Jamaïque	0.995503	0.078885	0.916618
Iraq	0.995431	0.877040	0.118391
Portugal	0.993950	0.993142	0.000808
République-Unie de Tanzanie	0.992079	0.963739	0.028341
Danemark	0.991633	0.988750	0.002883
Sierra Leone	0.991587	0.968950	0.022636

COS2_1er_plan_factoriel			
id	COS2_F1	COS2_F2	
Panama	0.003760	0.000993	0.002768
Kiribati	0.064521	0.062795	0.001725
Kirghizistan	0.067423	0.006121	0.061302
Mongolie	0.102084	0.000775	0.101310
Costa Rica	0.105337	0.037995	0.067341
Bahamas	0.145905	0.128146	0.017760
Malaisie	0.188292	0.157295	0.030997
Iran (République islamique d')	0.215854	0.004158	0.211695
Venezuela (République bolivarienne du)	0.245990	0.094883	0.151107
Antigua-et-Barbuda	0.250713	0.046662	0.204051

V. Qualité de représentation

id	CTR_F1
Zambie	0.022976
Libéria	0.020350
Lituanie	0.020058
Madagascar	0.020044
Ouganda	0.018990
Rwanda	0.018376
Mozambique	0.018317
Chine - RAS de Hong-Kong	0.018059
Islande	0.017986
Tchad	0.017216

Top 10

id	CTR_F1
Kirghizistan	6.219170e-07
Saint-Kitts-et-Nevis	6.893884e-07
Panama	1.075693e-06
Mongolie	2.803287e-06
Iran (République islamique d')	1.285789e-05
Gabon	2.362841e-05
Algérie	2.460875e-05
Fidji	2.858141e-05
Viet Nam	3.081229e-05
Grenade	3.245125e-05

Min 10

$$1/172 = 0.005814$$

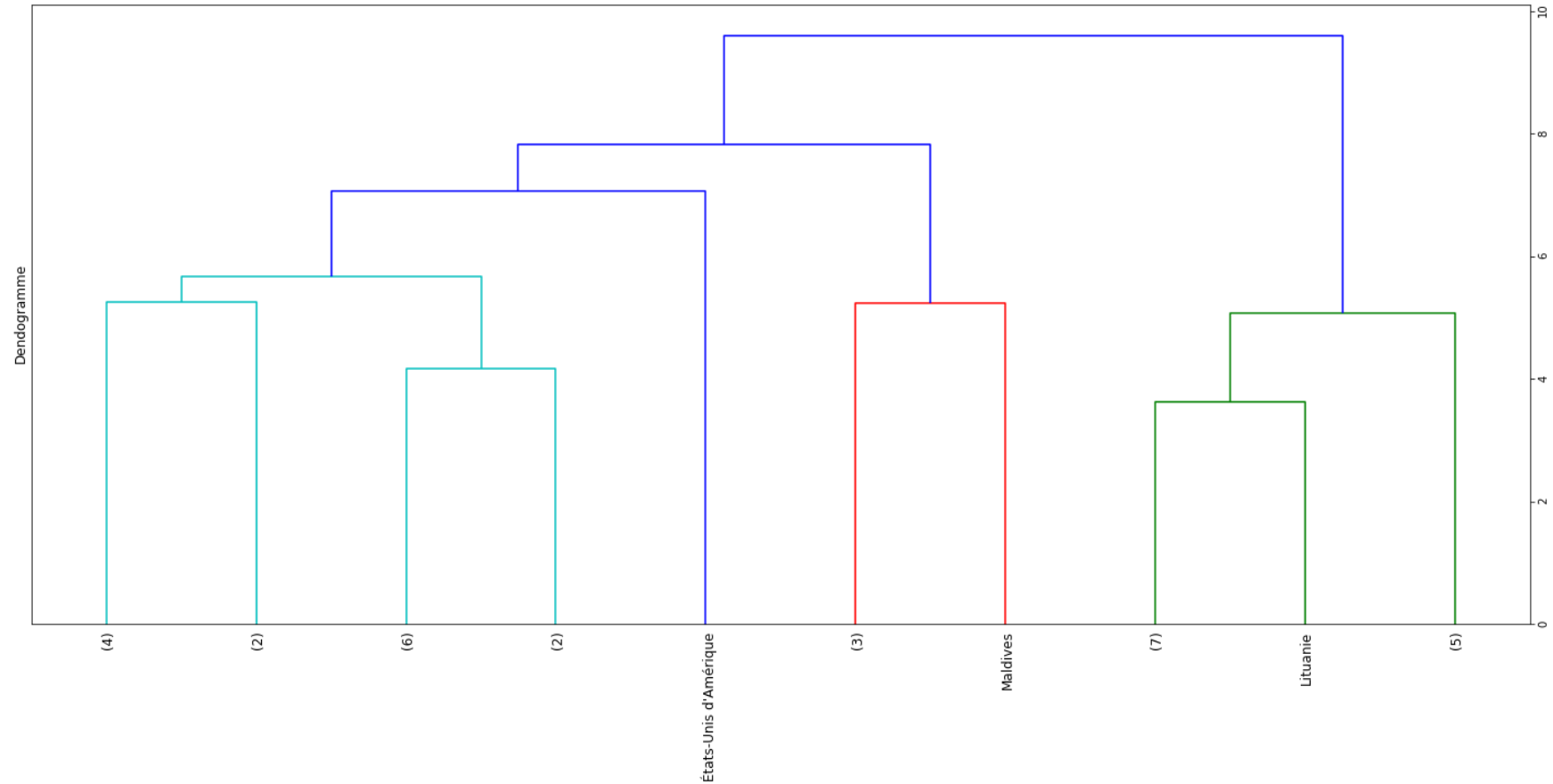
VI. Choix des pays

Choix du focus sur le cluster 1

32 Pays

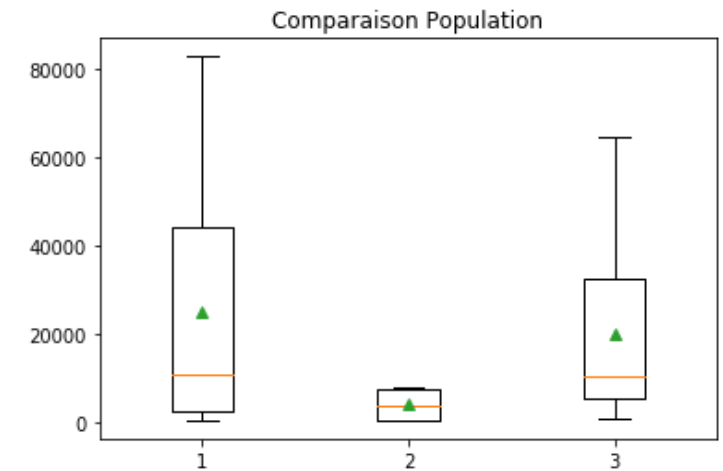
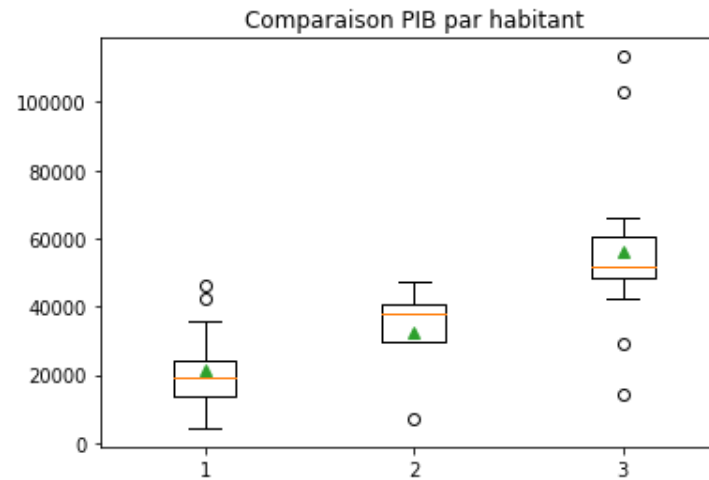
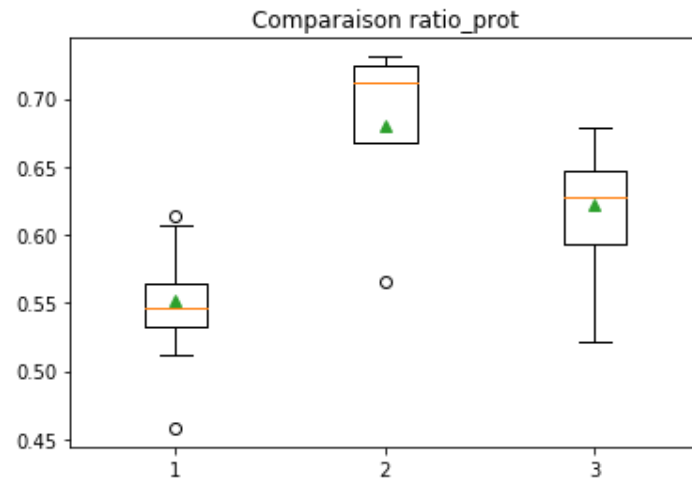
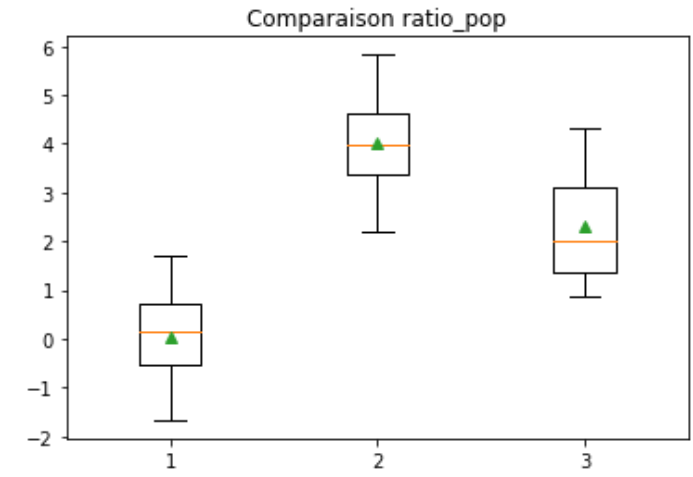
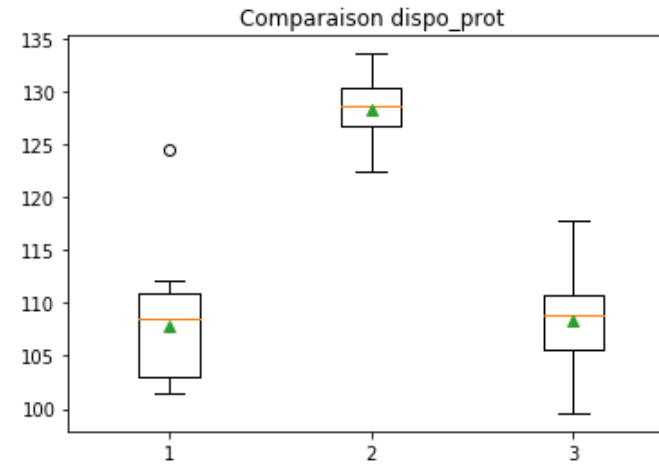
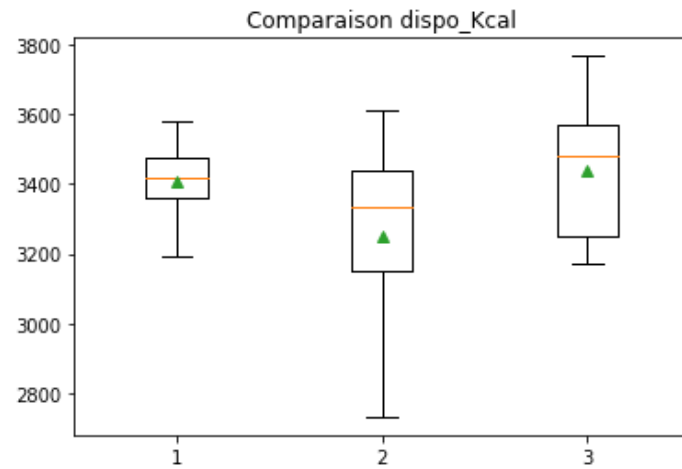
	dispo_Kcal	dispo_prot_total	ratio_pop	ratio_prot	\
Pays					
Albanie	3193	111.42	0.730159	0.533297	
Allemagne	3499	101.59	-0.349326	0.605178	
Argentine	3229	102.64	2.655174	0.652182	
Australie	3276	106.28	4.191216	0.674445	
Autriche	3768	106.21	1.106879	0.591846	
Belgique	3733	99.59	1.489809	0.583593	
Canada	3494	104.95	3.094415	0.521201	
Chine - RAS de Hong-Kong	3290	129.18	2.184397	0.730841	
Danemark	3367	108.88	1.225005	0.641256	
Espagne	3174	104.88	1.613183	0.621186	
Estonie	3253	103.90	-0.923788	0.512416	
États-Unis d'Amérique	3682	109.60	2.499303	0.636679	
Fédération de Russie	3361	102.84	-0.545893	0.546285	
Finlande	3368	117.72	1.080477	0.620370	
France	3482	110.52	1.676393	0.627488	
Grèce	3400	108.80	0.162016	0.544485	
Irlande	3600	110.02	3.558639	0.589166	
Islande	3380	133.54	3.773585	0.722480	
Israël	3610	128.14	4.218329	0.565553	
Italie	3579	108.51	0.794923	0.536725	
Lituanie	3417	124.49	-1.662321	0.614427	
Luxembourg	3539	113.88	4.330709	0.633298	
Maldives	2732	122.43	5.828221	0.702115	
Malte	3378	110.36	0.941176	0.559442	
Monténégro	3491	112.07	0.161290	0.563130	
Norvège	3485	110.90	3.107749	0.595041	
Pays-Bas	3228	111.72	0.866687	0.678213	
Pologne	3451	101.47	0.047122	0.524983	
Portugal	3477	110.88	0.169972	0.606782	
Roumanie	3358	103.02	-0.741046	0.458261	
Royaume-Uni	3424	103.21	1.723640	0.564674	
Suède	3179	107.72	2.014496	0.657538	

VI. Choix des pays





Découpage en 10 clusters

VI. Choix des pays



VI. Choix des pays

	Pays	Population	score1	PIB	score2	total
	États-Unis d'Amérique	320051	15	52898.817379	10	25
	Australie	23343	10	66301.306788	13	23
	Canada	35182	11	52264.959967	9	20
	Suède	9571	7	60190.029595	11	18
	France	64291	14	42493.606628	3	17
	Pays-Bas	16759	9	51466.478134	8	17
	Danemark	5619	5	60942.805627	12	17
	Norvège	5043	3	103110.441896	14	17
	Luxembourg	530	1	113341.237463	15	16
	Espagne	46927	13	29163.288417	2	15
	Argentine	41446	12	14417.421454	1	13
	Belgique	11104	8	46713.462219	4	12
	Autriche	8495	6	50137.519579	6	12
	Finlande	5426	4	49659.588178	5	9
	Irlande	4627	2	51130.077752	7	9

Choix des pays Suède, Pays-Bas, Danemark et Norvège

VII. Tests statistiques

```
from scipy.stats import kstest

#Hypothèse  $H_0$  : la disponibilité en Kcal suit une distribution gaussienne

print(kstest(X_scaled[:,0], 'norm'))

#La p-value est supérieure au seuil de 5%. L'hypothèse  $H_0$  n'est ainsi pas rejeté.
#La distribution du ratio de protéines est donc gaussienne

KstestResult(statistic=0.05819703271366561, pvalue=0.6004195326847048)
```



VII. Tests statistiques

```
#Hypothèse H0 : Les deux clusters suivent la même loi de distribution pour la disponibilité en Kcal  
from scipy.stats import bartlett  
bartlett(cluster_1_scaled[:,0],cluster_2_scaled[:,0])  
#La p-value est supérieure au seuil de 5%. On ne rejette donc pas l'égalité des variances.
```

```
BartlettResult(statistic=0.7266592689821892, pvalue=0.39396819736177313)
```

```
from scipy.stats import ttest_ind  
ttest_ind(cluster_1_scaled[:,0],cluster_2_scaled[:,0], equal_var=True)  
# La p-value est bien inférieure au seuil de 5%. On rejette l'hypothèse H0.  
#Les deux clusters ne suivent pas la même loi de distribution pour la disponibilité en Kcal
```

```
Ttest_indResult(statistic=5.823178675834855, pvalue=1.6026269596095416e-07)
```

URL des images utilisées

- <https://openclassrooms.com/fr/projects/produisez-une-etude-de-marche>
- <https://www.cuisineaz.com/recettes/poulet-entier-au-cookeo-98217.aspx>
- <http://www.ecommercemag.fr/Thematique/management-1225/Breves/Commerce-Paris-2014-conseils-experts-start-bien-exporter-245853.htm>