

Formation Data Analyst

Projet 3 : Réalisez une étude de santé publique

24/01/2021

Aïssa MOUACHA



SOMMAIRE

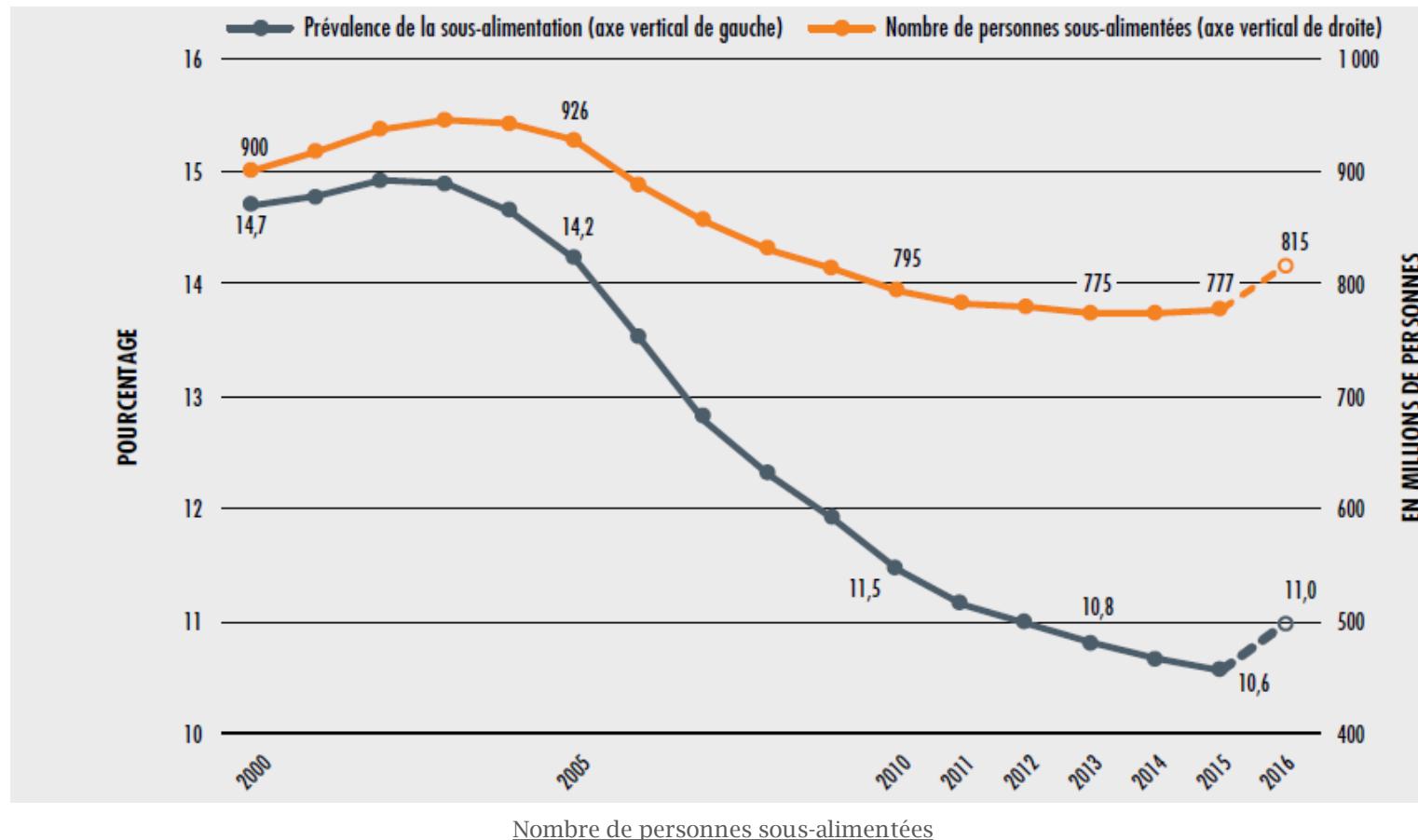
- Eléments de contexte
- La faim dans le monde
- Annexes

Ma mission au sein de la FAO :

Réaliser une étude de grande ampleur sur le thème de la sous-nutrition dans le monde.

→ Statut en 2013

→ multiples causes de la faim



Le 25 Septembre 2015, les 193 états membres des Nations Unies ont adopté le Programme de développement durable à l'horizon 2030. Il comprend 17 objectifs avec 169 buts et 230 indicateurs.

"LEAVE NO ONE BEHIND"



Quelques chiffres récents :



2013



2030



2050



Projection : Population mondiale

THEMES
↓



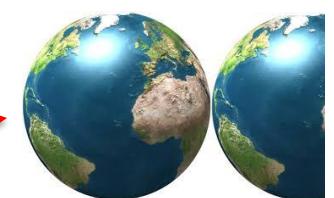
+19%

~8.3
Milliards



+17%

~9.7
Milliards



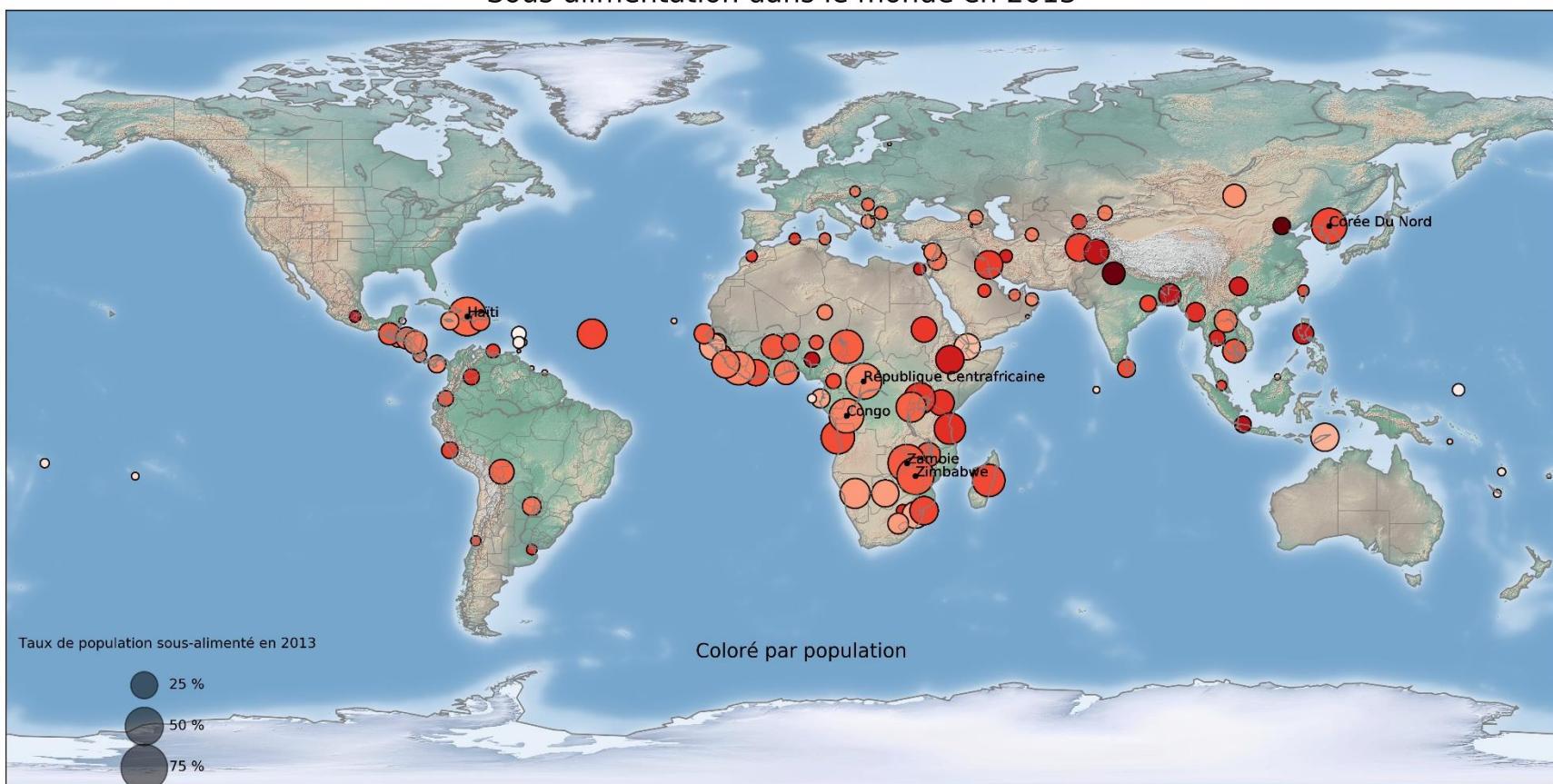
DEMOGRAPHIE

**BESOINS
ALIMENTAIRES**

ENJEU

Répondre au besoin croissant
Éradiquer la faim

Sous-alimentation dans le monde en 2013



- Afrique subsaharienne
- Asie du Sud Est
- Amérique du Sud

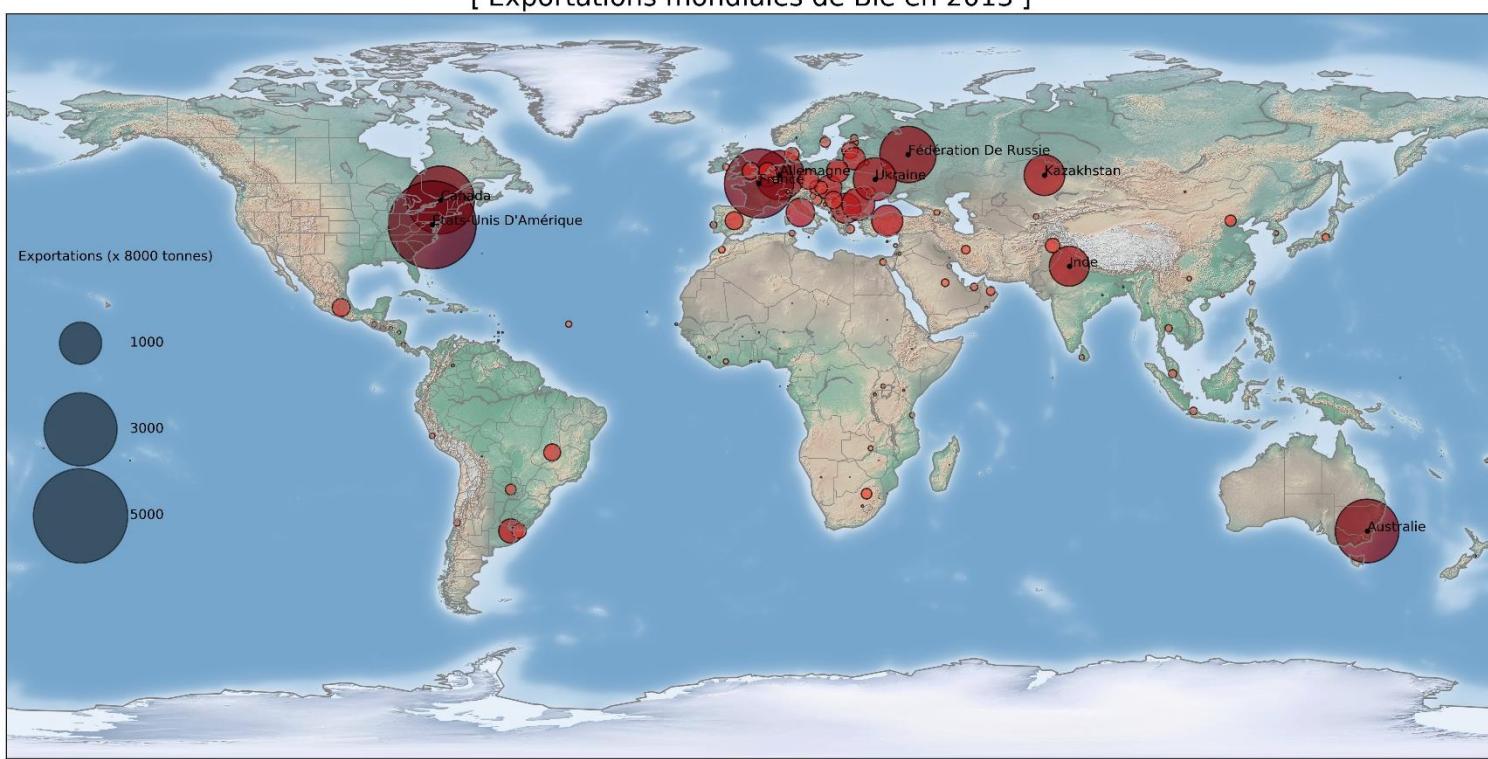
746 Millions de personnes en
2013 souffrent de sous-nutrition

```
# Echelle de couleur par densité de population + taille par sous-alimentation
m.scatter(lon,lat,latlon=True,c=np.log10(population),s=15*100*pourcentage,cmap='Reds',edgecolors = 'k',alpha=0.7)
# Affection Légende avec dummy points
for pourcentage in [25, 50, 75]:
    plt.scatter([],[],c='k', edgecolors = 'k', alpha=0.5, s=15*pourcentage, label=str(pourcentage) + '%') #marker oversizing
plt.legend(scatterpoints=1, frameon=False,labelspacing=2, loc='lower left',title='Taux de population sous-alimenté en 2013');
plt.title("Sous-alimentation dans le monde en 2013",fontsize=20)
```

Source :
http://www.fao.org/PDF/02_nutrition_en.pdf

['Exportations mondiales de Blé en 2013']

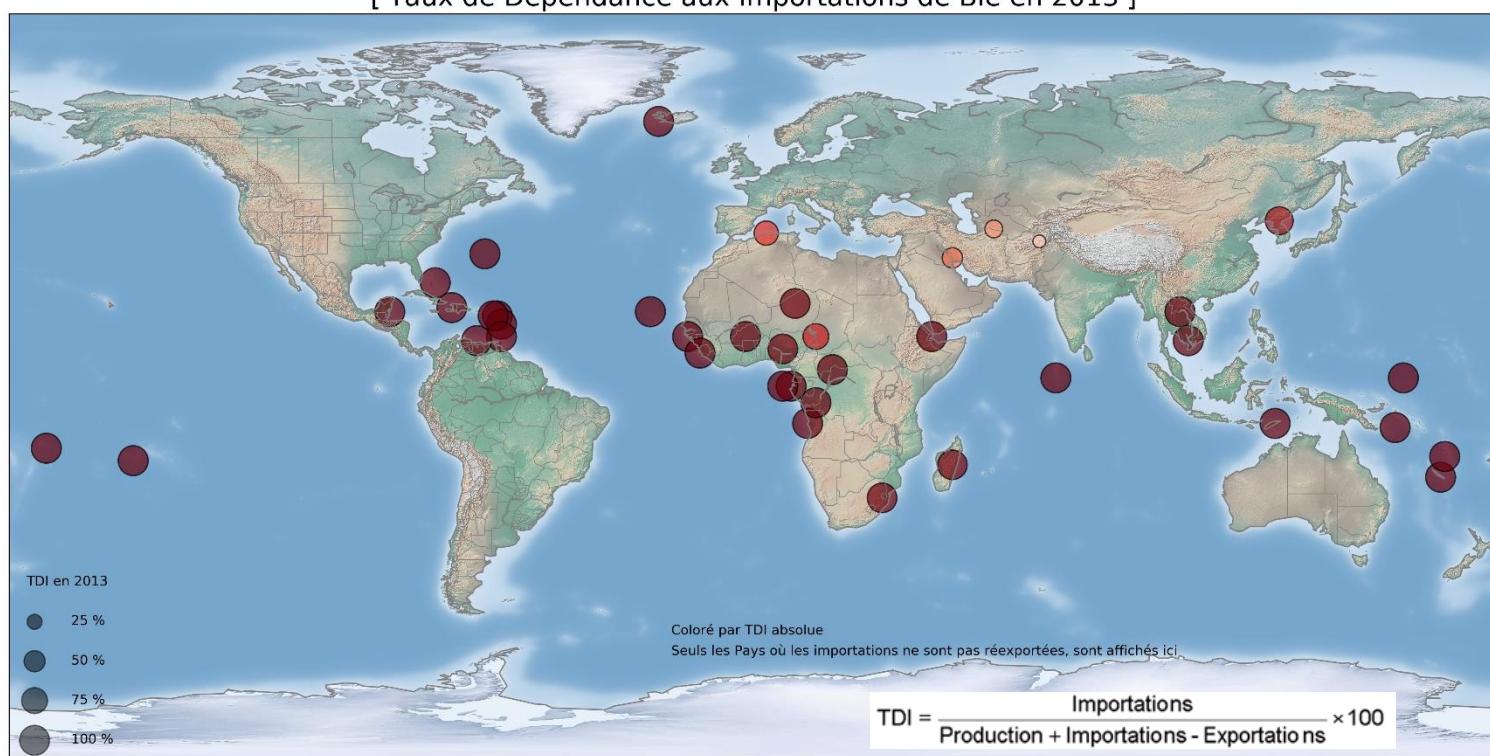
Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucre Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987



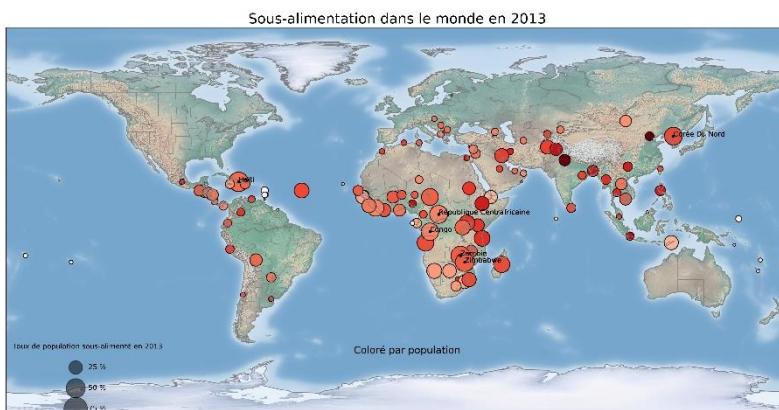
- Hémisphère Nord + Australie : 1^{ers} exportateurs de blé
- Hémisphère Sud en déficit de ce produit

['Taux de Dépendance aux Importations de Blé en 2013']

Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucre Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987

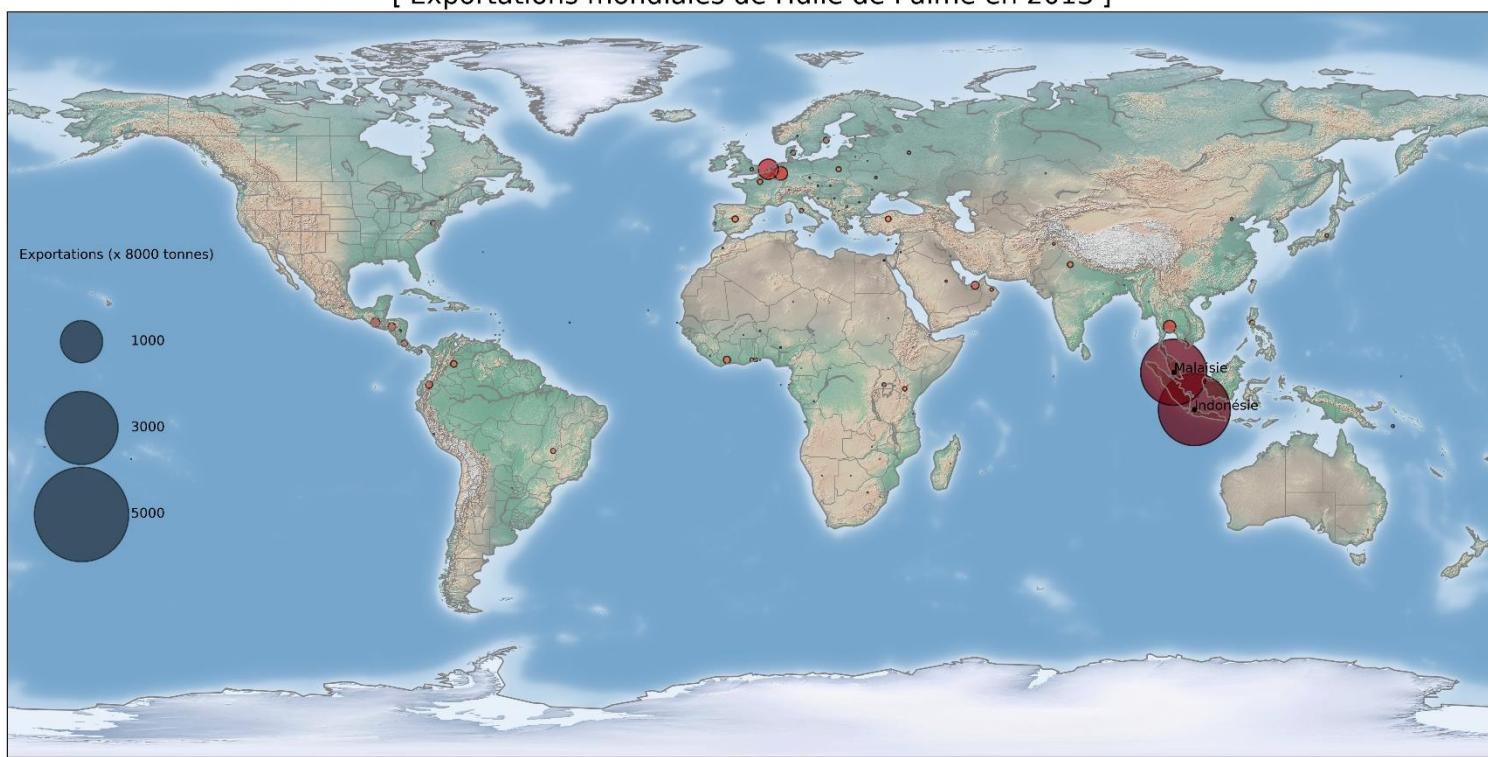


- En réponse à ces exportations, le TDI montre que parmi les pays importateurs, une grande majorité fait partie des pays recensés en sous-alimentation.



['Exportations mondiales de Huile de Palme en 2013']

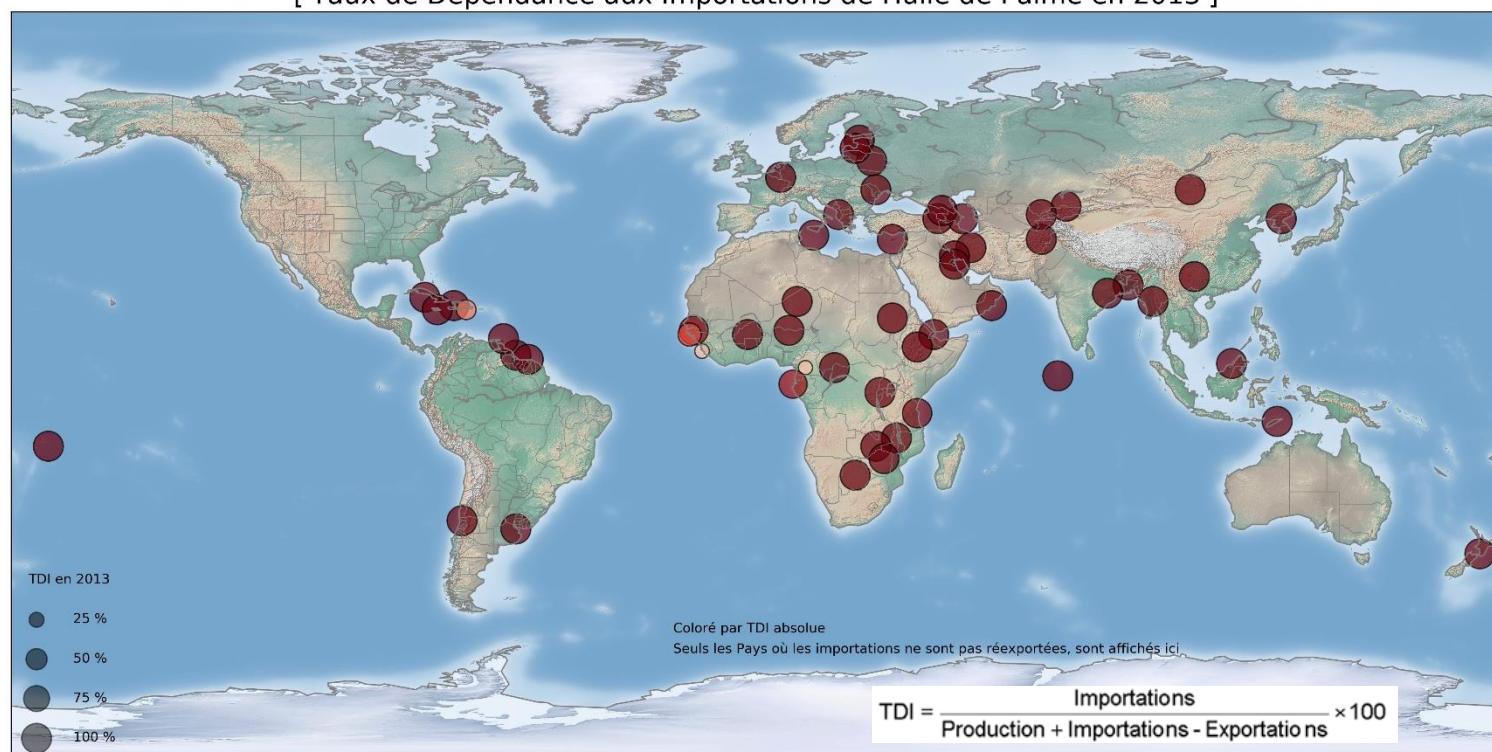
Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucré Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987



- Malaisie et Indonésie : 1^{ers} et exclusifs exportateurs d'huile de palme

'[Taux de Dépendance aux Importations de Huile de Palme en 2013]'

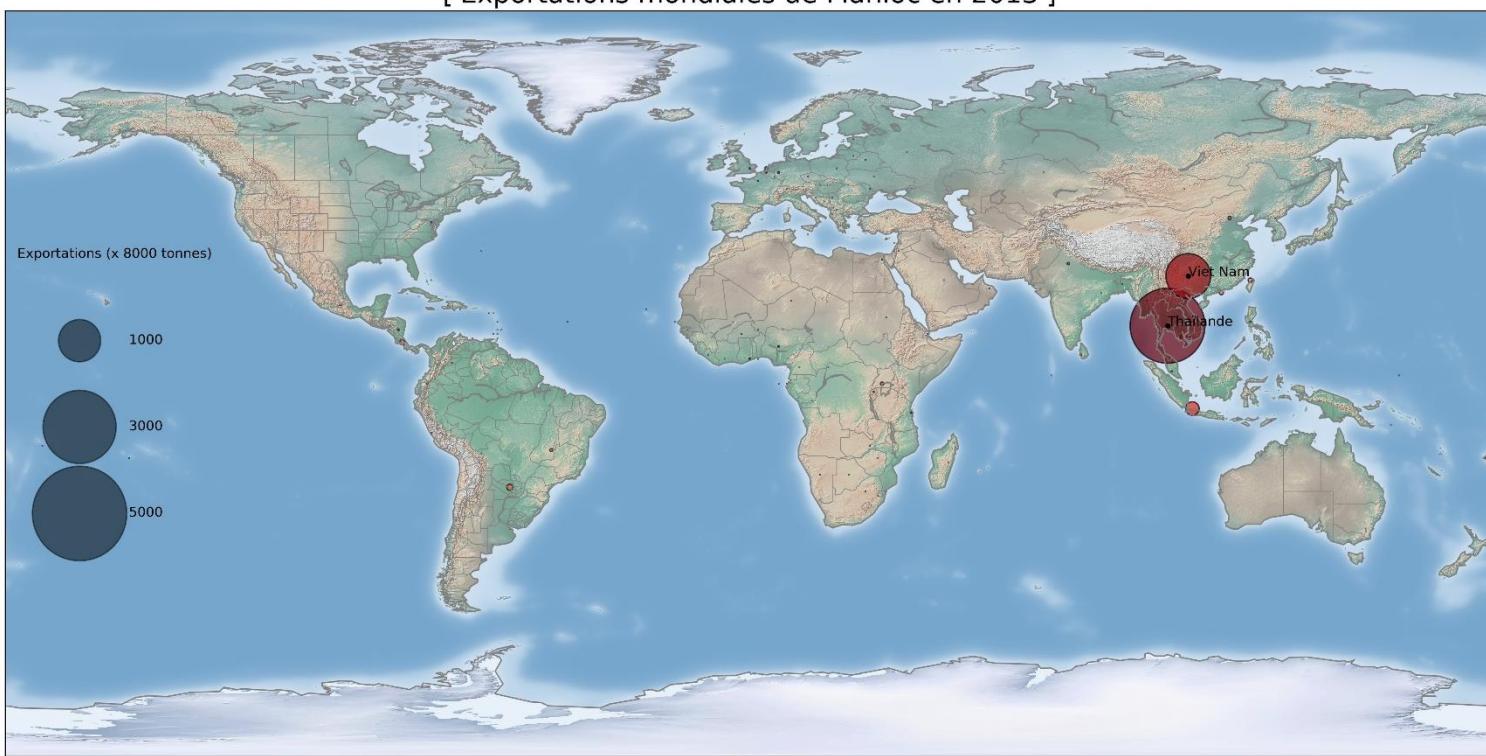
Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucré Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987



- Amérique du sud, Afrique et Asie sont particulièrement représentés pour la consommation seule d'huile de palme.

['Exportations mondiales de Manioc en 2013']

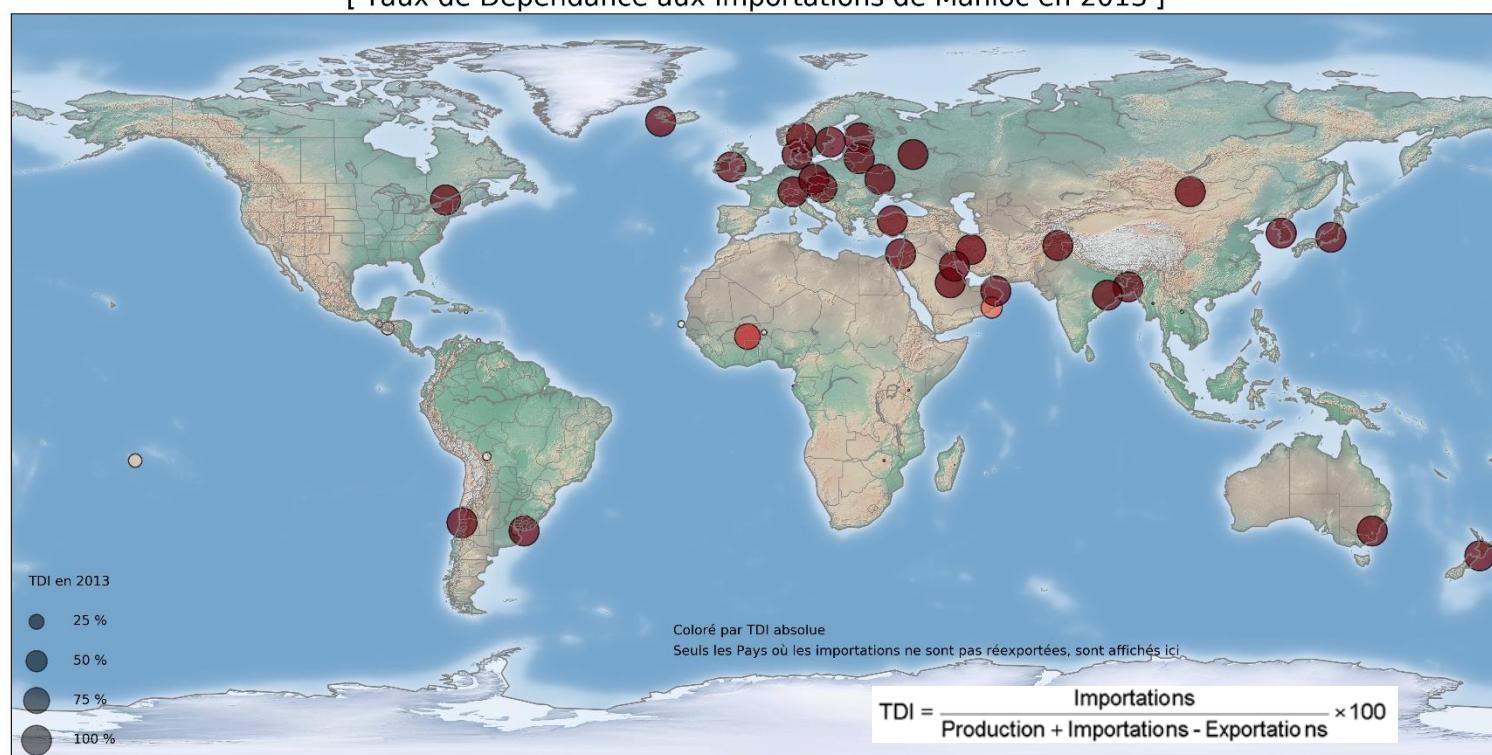
Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucré Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987



- Thaïlande et Viet-Nam : 1^{ers} et exclusifs exportateurs de manioc

['Taux de Dépendance aux Importations de Manioc en 2013']

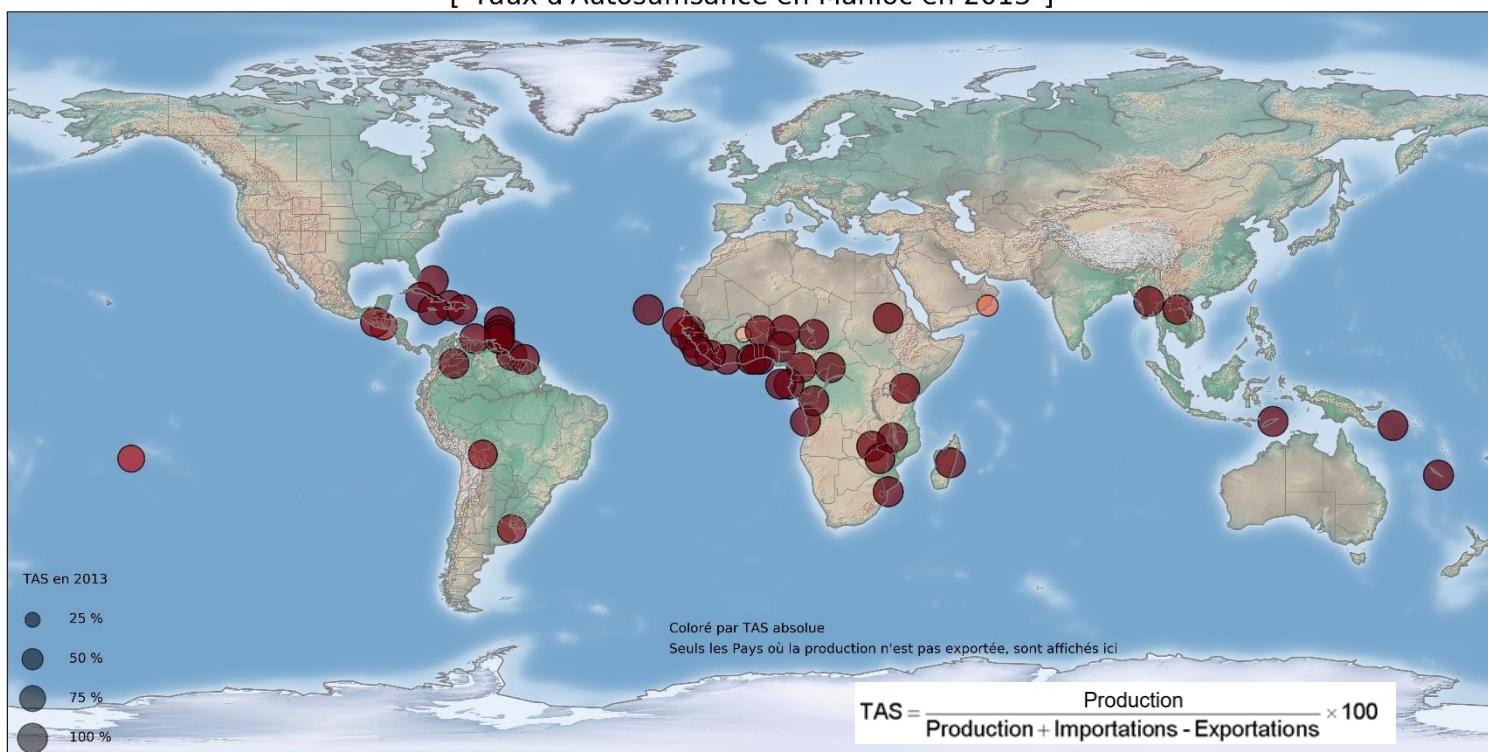
Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucre Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987



- Europe, US, Moyen-Orient sont les principaux importateur de manioc pour leur consommation.

["Taux d'Autosuffisance en Manioc en 2013"]

Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soja	106912
Sucré Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987

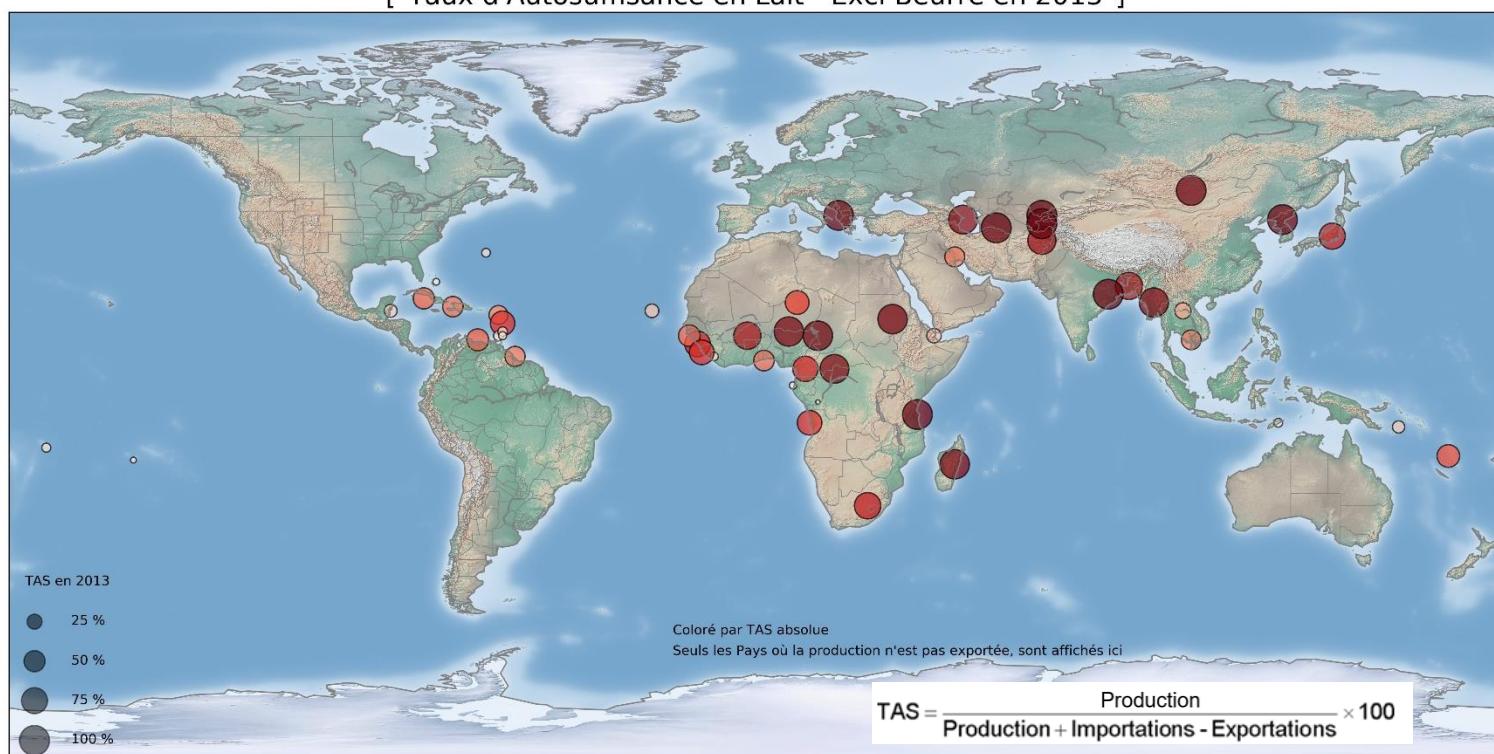


- De nombreux pays dits sous-alimentés sont autosuffisants en manioc (Afrique et Caraïbes)
- L'aspect excédentaire des pays n'est pas représenté ici (exportations nulles)

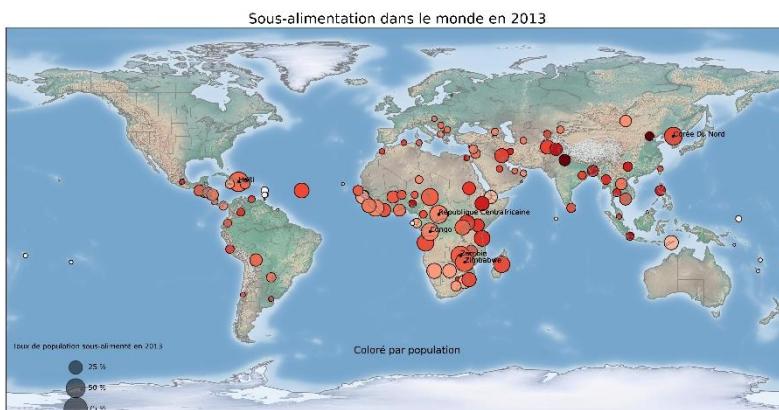


["Taux d'Autosuffisance en Lait - Excl Beurre en 2013"]

Produit_Z_ssai	Exportations_Z_ssai
Blé	198380
Mais	128787
Lait - Excl Beurre	125098
Soya	106912
Sucre Eq Brut	68320
Huile de Palme	50673
Légumes, Autres	44535
Orge	41585
Riz (Eq Blanchi)	41445
Oranges, Mandarines	37489
Manioc	36052
Fruits, Autres	30165
Pommes de Terre	29259
Poissons Pelagiques	23134
Tomates	21987



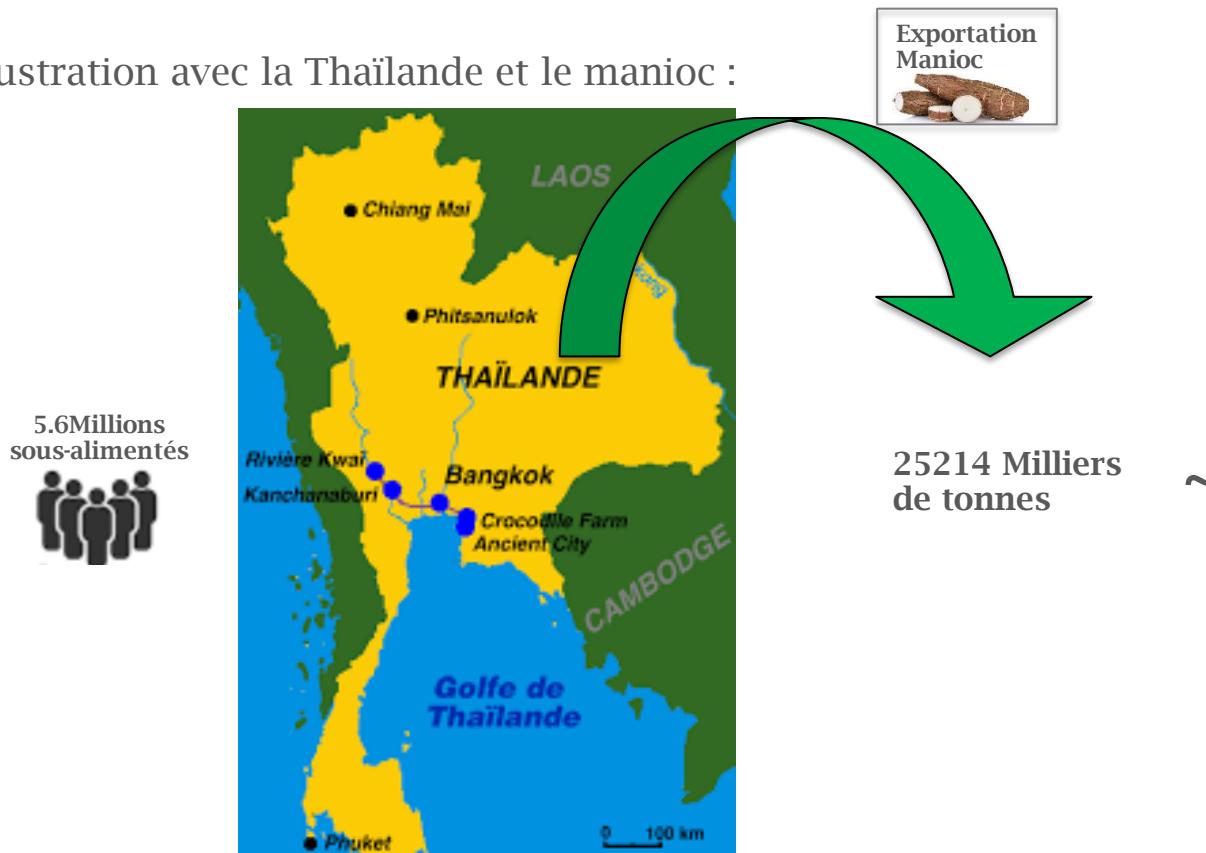
- De nombreux pays dits sous-alimentés sont autosuffisants en lait (Afrique , Asie et Caraïbes)
- L'aspect excédentaire des pays n'est pas représenté ici (exportations nulles)





En multipliant l'exercice précédent avec d'autres (groupements de) produits, on démontre aisément que des pays considérés en sous-alimentation peuvent subvenir aux besoins de leur propre population. C'est le **paradoxe de la faim**.

Illustration avec la Thaïlande et le manioc :

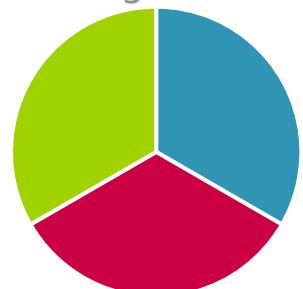


* : déterminé via le ratio énergétique du manioc trouvé en Question 4 (1123 kcal/kg) + besoin nutritionnel moyen ~2500 kcal/jr

```
#détermintaion des exportations de manioc en Thaïlande from df2 vegetaux
thai_exp = df2.loc[(df2['Code zone']==216) & (df2['Code Produit']==2532) & (df2['Code Élément']==5911),['Valeur']].sum()
print('\x1b[6;31;40m' + "exportations de manioc en Thaïlande : %.f" %thai_exp + " Milliers de tonnes"+ '\x1b[0m')
exportations de manioc en Thaïlande : 25214 Milliers de tonnes
```

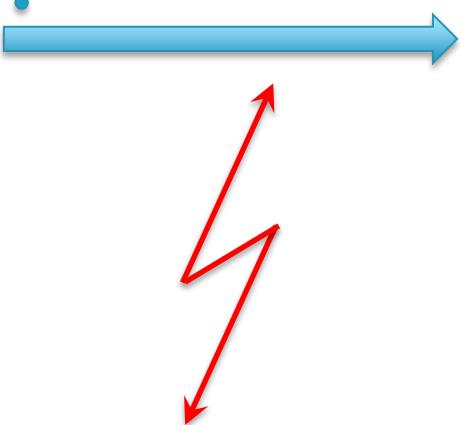
Source :
https://doi.org/10.5281/zenodo.3230606

Production Agricole Mondiale

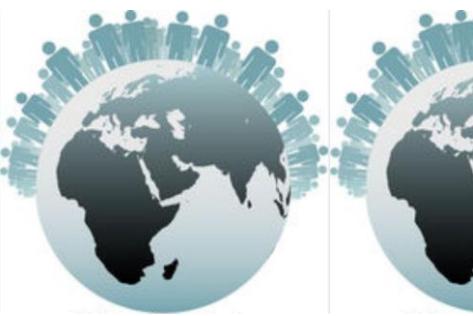


- Alimentation animale
- Alimentation humaine d'origine végétale
- Non-alimentaire, pertes

Dans un
monde
idéal...

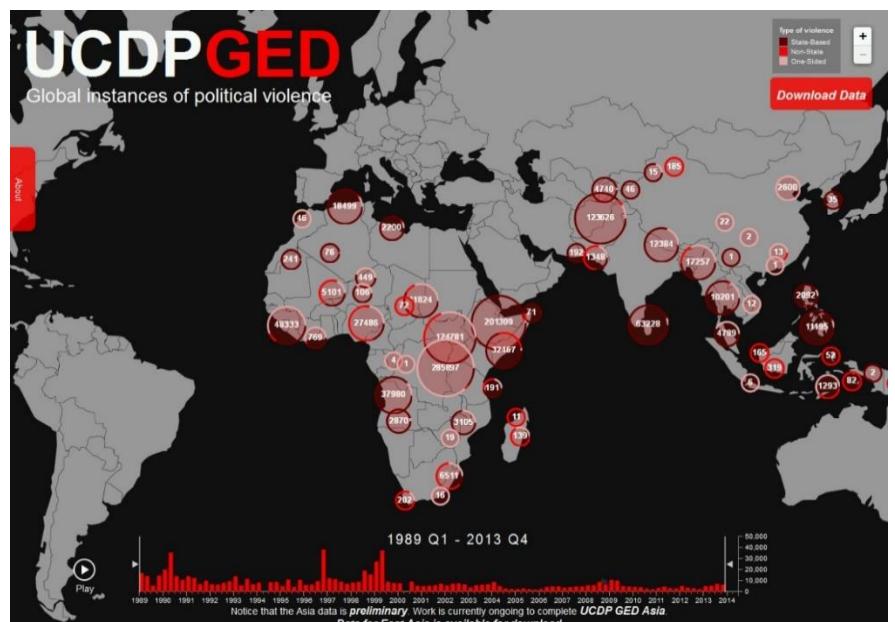
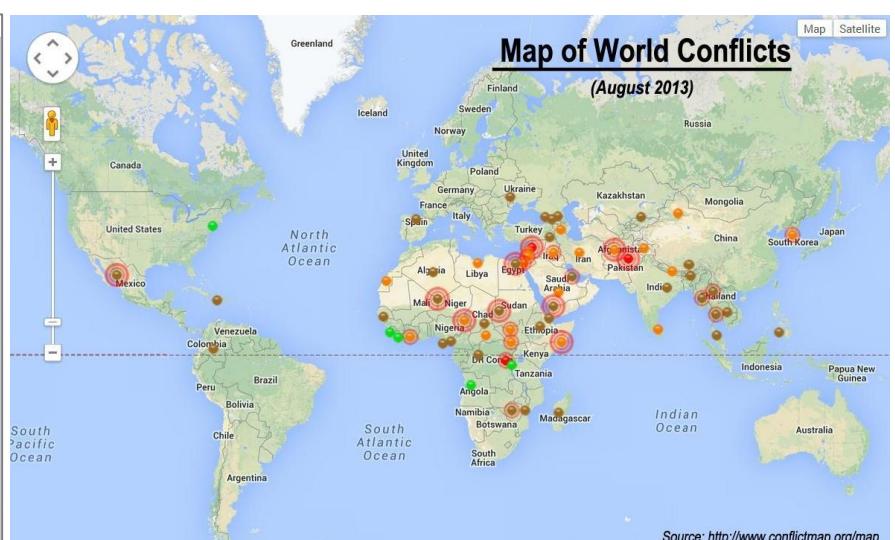
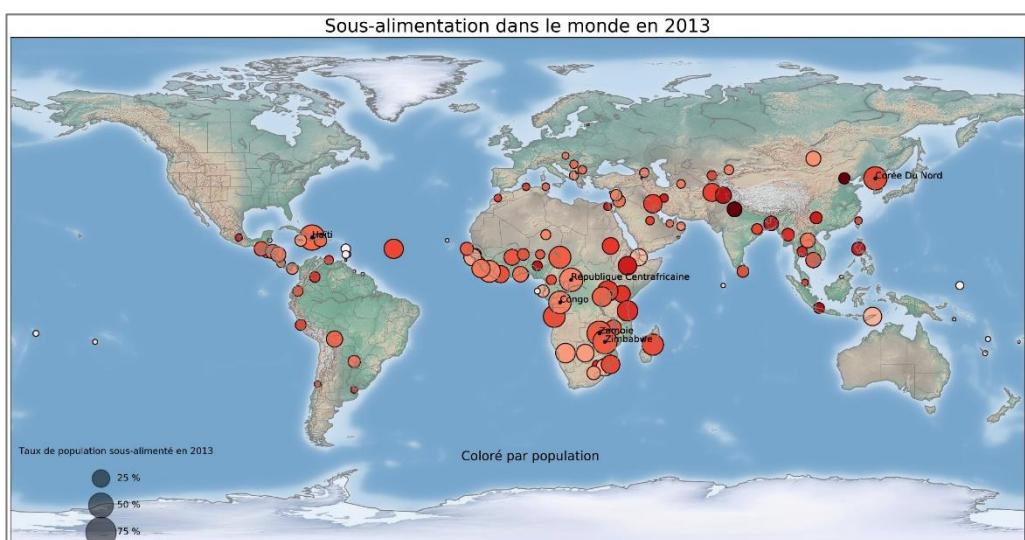


10 Milliards
d'humains nourris



Il ne s'agit pas d'un problème de production mais de **causes** qui bouleversent cet idéal

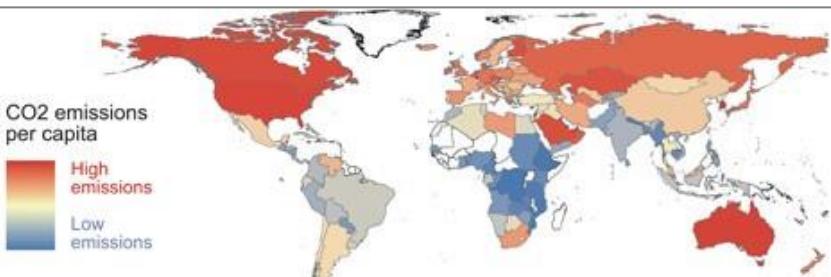
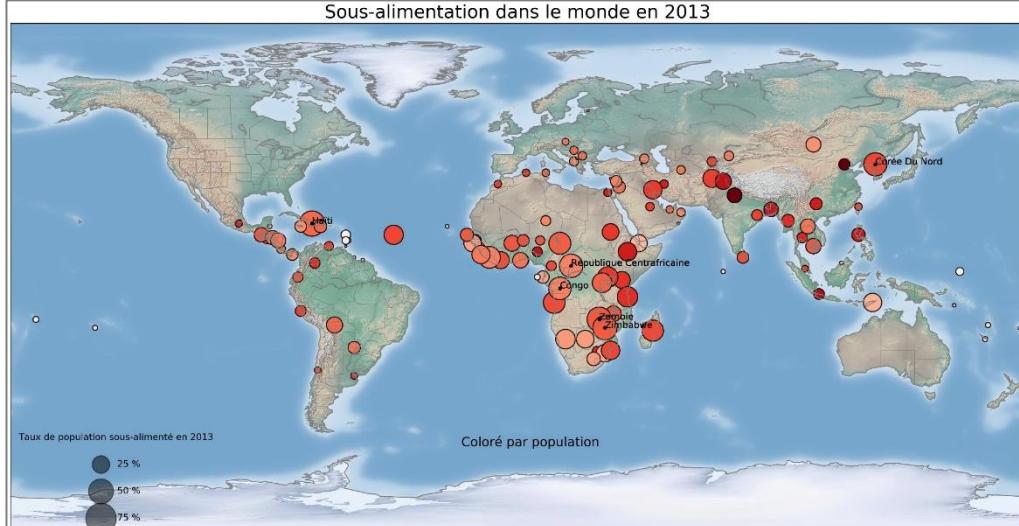
```
#info:quantité de calories nécessaires chaque jour à l'adulte en moyenne, de 2400 à 2600 kcal par jour.
#partons sur un besoin nutritionnel moyen => 2500 kcal/jr
nb_humains = somme_q8_kca / (2500^365)
Disponibilité intérieure mondiale (Q8) : 9.17e+15 kcal
Nb humains nourris : 10049419319
Rapporté à la pop Mondiale : 144 %
```



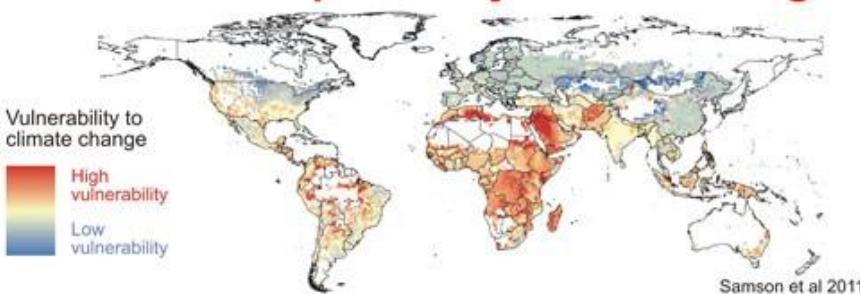
Les zones touchées par la sous-nutrition le sont aussi par des **conflits** divers.

- lien entre conflit et insécurité alimentaire
- prévalence des cas de sous-alimentation 2 fois + élevée dans pays affectés par un conflit et une crise prolongée

Sous-alimentation dans le monde en 2013



Those who contribute the least greenhouse gases will be most impacted by climate change



Les zones touchées par la sous-nutrition le sont aussi par des **changements climatiques** majeurs.

- sécheresses, inondations, cyclones, canicules...
- impacts de ces changements plus importants dans les pays émettant peu de gaz à effet de serre
- injustice entre causes et conséquences de ces changements climatiques
- conséquences : réduction voire destruction des récoltes (cultures anéanties, sols appauvris, augmentation des prix des aliments)
- Augmentation du nombre de réfugiés climatiques

➤ Politiques agricoles

- très forte disparité pour le **budget** alloué par les gouvernements à l'agriculture
➔ US, UE vs. pays comme Bolivie, Congo, Ethiopie
- une majorité de pays africains suit une politique de **culture exportatrice** au détriment d'une culture destinée à nourrir sa population

café



sucre



cacao



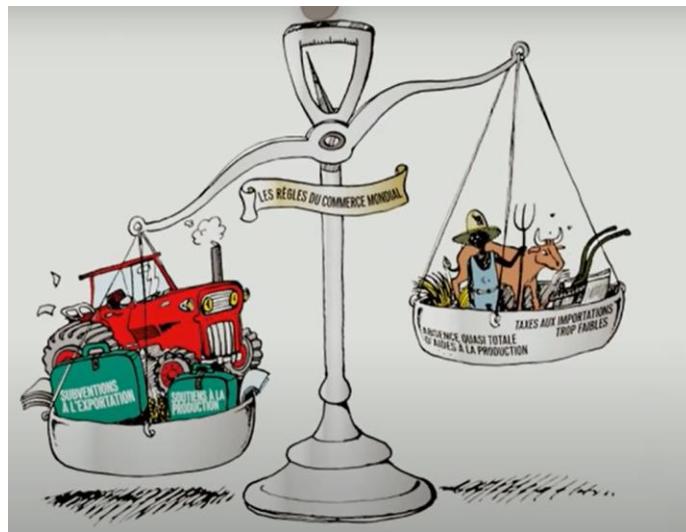
coton



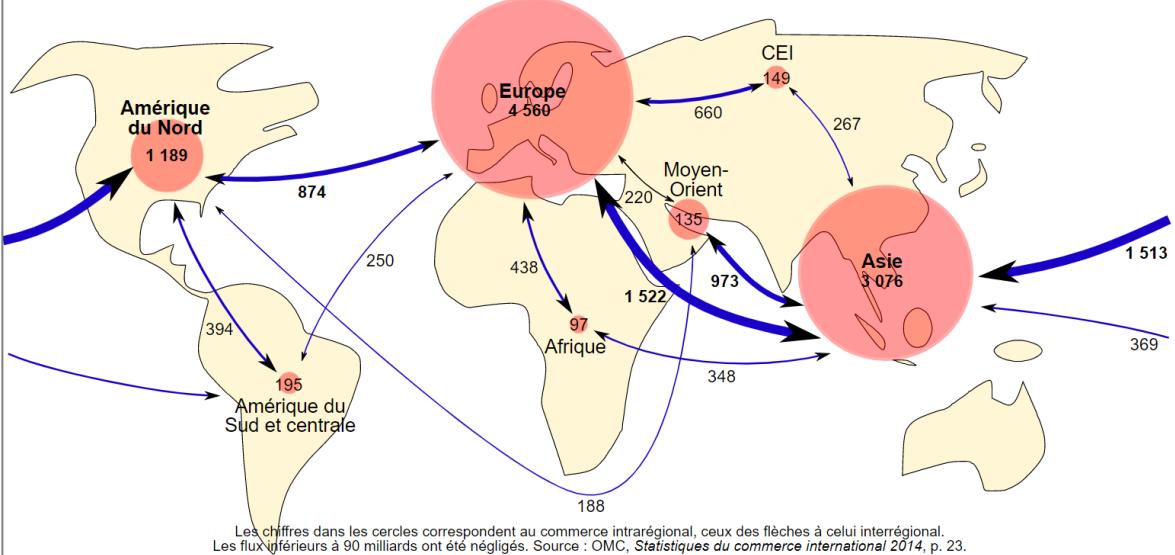
- **dépendance à l'importation** pour se nourrir (et là aussi la pauvreté creuse les écarts)

➤ Politiques commerciales

- avec la baisse des taxes & quotas : libéralisation des échanges facilitée
- fortes disparités Nord/Sud (accès semences/terre, motorisation parc, subventions, financements)
- ➔ l'Afrique n'est pas un acteur majeur



Les flux internationaux de marchandises, en milliards de dollars pour 2013



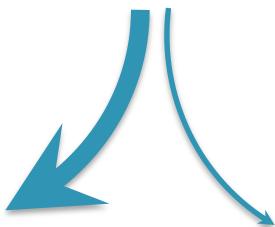
➤ Politiques énergétiques

- valorisation des agrocarburants appelle une production croissante au détriment de l'alimentation des populations locales

multiplication de surfaces consacrées aux futurs biocarburants



demande croissante



alimentation défavorisée



L'étude « How biofuels could starve the poor » a montré que la quantité de maïs nécessaire pour produire 95 litres d'éthanol (le réservoir d'une grosse voiture) fournirait suffisamment de calories à un individu pendant un an.

2013

2017

2020

2030

2050



Projections / Estimations

THEMES

DEMOGRAPHIE

~7
Milliards

FAIM

~690
Millions

REFUGIES CLIMATIQUES

~19 Millions
~23 Millions

DECES DE LA FAIM / AN

~9 Millions
(des causes associées,
maladie, manque
d'eau potable et
hygiène)

~9.7
Milliards

~840
Millions

~143
Millions

Objectif Zéro faim 2030 difficile à atteindre

Renouveler les **efforts**

Accroître les **investissement**

Mettre en place de **nouvelles façons de travailler**

Répartir plus **équitablement** la production mondiale **entre riches et pauvres**.

Maintenir les opérations de **développement**.

Redistribuer les recettes de la **taxe carbone**.

Multiplier les **efforts de paix**.

Sensibiliser les populations à l'agro-écologie, l'agro-foresterie, l'hydroponie.

Transformer les systèmes alimentaires
Rendre l'alimentation saine plus **abordable**

Harmoniser les actions d'**aide humanitaire**.

Diminuer le coût des aliments nutritifs

Entrez vos données dans une base de données relationnelle (préparation & export des csv)

Export table population

```
#on repart du dataset "population" (df1)
#on conserve les colonnes suivantes
df1=df1[['Zone','Code zone','Année','Valeur']]
#on renomme les colonnes pour coller à l'énoncé
df1.rename(columns={'Zone':'pays','Code zone':'code_pays','Année':'annee','Valeur':'population'}, inplace=True)
#conversion unité population
df1['population']=df1['population']*1000
#export du csv
df1.to_csv("Documents/Dossier_AISSA/POLE_EMPLOI/OPENCLASSROOM/P3_mouacha_aissa/table1_population.csv", index = False)
```

* Voir Annexes pour le code d'exportation de toutes les tables



Structure								
	Données		Contraintes		Index		Déclencheurs	
	Table	Formulaire						
	Nom de la table :			WITHOUT ROWID				
1	pays	TEXT	Primary Key	Foreign Key	Unique	Contrôle	Not NULL	Collate
2	code_pays	INTEGER						NULL
3	annee	DATE						NULL
4	population	INTEGER						NULL

Structure				
Table		Formulaire		
	Structure	Données	Contraintes	Index
1	pays	code_pays	annee	population
2	Arménie		1	2013
2	Afghanistan		2	2013
3	Albanie		3	2013
4	Algérie		4	2013
5	Angola		7	2013
6	Antigua-et-Barbuda		8	2013
7	Argentine		9	2013
8	Australie		10	2013

Primary keys:

- table population : [code_pays] (+année dans un cadre optionnel)
- table dispo_alim : [code_pays + code_produit] (+année dans un cadre optionnel)
- table equilibre_prod : [code_pays + code_produit] (+année dans un cadre optionnel)
- table sous_nutrition : [code_pays] (+année dans un cadre optionnel)



Question 19 : Écrivez les requêtes SQL permettant de connaître...

- • Les 10 pays ayant le plus haut ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant, **puis** en termes de kcal par habitant.
- Pour l'année 2013, les 10 pays ayant le plus faible ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant.
- La quantité totale (en kg) de produits perdus par pays en 2013.
- Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.
- Les 10 produits pour lesquels le ratio **Autres utilisations/Disponibilité intérieure** est le plus élevé.

```
CREATE TABLE SQL_Q19_a_high_ratio_kg AS
SELECT pays, CAST(ROUND(SUM(dispo_prot*365/1000),2) AS REAL) AS ratio_prot_kg
/*ratio annuel d'où (x365)*/
FROM dispo_alim
GROUP BY pays
ORDER BY ratio_prot_kg DESC
LIMIT 10
```



	pays	ratio_prot_kg
1	Islande	48.57
2	Chine - RAS de Hong-Kong	47.11
3	Israël	46.72
4	Lituanie	45.39
5	Maldives	44.65
6	Finlande	42.91
7	Luxembourg	41.48
8	Monténégro	40.84
9	Pays-Bas	40.68
10	Albanie	40.65



Question 19 : Écrivez les requêtes SQL permettant de connaître...

- • Les 10 pays ayant le plus haut ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant, **puis** en termes de kcal par habitant.
- Pour l'année 2013, les 10 pays ayant le plus faible ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant.
- La quantité totale (en kg) de produits perdus par pays en 2013.
- Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.
- Les 10 produits pour lesquels le ratio **Autres utilisations/Disponibilité intérieure** est le plus élevé.

```
CREATE TABLE SQL_Q19_b_high_ratio_kcal AS
SELECT pays, CAST(ROUND(SUM(dispo_alim_kcal_p_j*365),0) AS REAL) AS ratio_kal
/*ratio annuel d'où (x365)*/
FROM dispo_alim
GROUP BY pays
ORDER BY ratio_kal DESC
LIMIT 10
```



	pays	ratio_kal
1	Autriche	1376050
2	Belgique	1364005
3	Turquie	1353420
4	États-Unis d'Amérique	1343930
5	Israël	1317650
6	Irlande	1314730
7	Italie	1305970
8	Luxembourg	1292100
9	Égypte	1284070
10	Allemagne	1278595



Question 19 : Écrivez les requêtes SQL permettant de connaître...

- Les 10 pays ayant le plus haut ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant, **puis** en termes de kcal par habitant.
- • Pour l'année 2013, les 10 pays ayant le plus faible ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant.
- La quantité totale (en kg) de produits perdus par pays en 2013.
- Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.
- Les 10 produits pour lesquels le ratio **Autres utilisations/Disponibilité intérieure** est le plus élevé.

```
CREATE TABLE SQL_Q19_c_low_ratio_kg AS
  SELECT pays, CAST(ROUND(SUM(dispo_prot*365/1000),2) AS REAL) AS ratio_prot_kg
  /*ratio annuel d'où (x365)*/
  FROM dispo_alim
  GROUP BY pays
  ORDER BY ratio_prot_kg ASC
  LIMIT 10
```



	pays	ratio_prot_kg
1	Libéria	13.75
2	Guinée-Bissau	16.08
3	Mozambique	16.67
4	République centrafricaine	16.8
5	Madagascar	17.04
6	Haïti	17.41
7	Zimbabwe	17.64
8	Congo	18.76
9	Ouganda	19.21
10	Sao Tomé-et-Principe	19.38



Question 19 : Écrivez les requêtes SQL permettant de connaître...

- Les 10 pays ayant le plus haut ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant, **puis** en termes de kcal par habitant.
- Pour l'année 2013, les 10 pays ayant le plus faible ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant.
- La quantité totale (en kg) de produits perdus par pays en 2013.
- Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.
- Les 10 produits pour lesquels le ratio **Autres utilisations/Disponibilité intérieure** est le plus élevé.

```
CREATE TABLE SQL_Q19_d_pertes_tot_kg AS
  SELECT pays, CAST(ROUND(SUM(pertes*365*1e6),0) AS REAL) AS Q_tot_pertes_kg
  /*total annuel d'où (x365)*/
  FROM equilibre_prod
  GROUP BY pays
  ORDER BY Q_tot_pertes_kg DESC
```



	pays	Q_tot_pertes_kg
1	Chine, continentale	32694875000000.
2	Brésil	27708610000000.
3	Inde	20414450000000.
4	Nigéria	7246710000000.
5	Indonésie	47745650000000.
6	Turquie	43931400000000.
7	Mexique	30254850000000.

165	Polynésie française	1095000000.
166	Bahamas	730000000.
167	Barbade	730000000.
168	Chine - RAS de Macao	365000000.
169	Antigua-et-Barbuda	0
170	Bermudes	0
171	Islande	0
172	Kiribati	0
173	Maldives	0
174	Saint-Kitts-et-Nevis	0



Question 19 : Écrivez les requêtes SQL permettant de connaître...

- Les 10 pays ayant le plus haut ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant, **puis** en termes de kcal par habitant.
- Pour l'année 2013, les 10 pays ayant le plus faible ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant.
- La quantité totale (en kg) de produits perdus par pays en 2013.
- • Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.
- Les 10 produits pour lesquels le ratio **Autres utilisations/Disponibilité intérieure** est le plus élevé.

```
CREATE TABLE SQL_Q_JOINTURE_operation_part1 AS
SELECT * FROM population
NATURAL JOIN sous_nutrition;
```

Jointure nécessaire pour déterminer la proportion attendue

```
CREATE TABLE SQL_Q19_e_high_ssal_proportion AS
SELECT pays, CAST(ROUND(MAX(100*(nb_personnes/population)),0) AS REAL) AS proportion
FROM SQL_Q_JOINTURE_operation_part1
WHERE nb_personnes <> ''
GROUP BY pays
ORDER BY proportion DESC
LIMIT 10
```



	pays	proportion **
1	Dominique	139
2	Kiribati	98
3	Saint-Vincent-et-les Grenadines	92
4	Samoa	53
5	Sao Tomé-et-Principe	52
6	Haiti	50
7	Zambie	48
8	Zimbabwe	47
9	République centrafricaine	43
10	République populaire démocratique de Corée	43

** : Proportion donnée avec l'hypothèse de population sous-alimentée étant égale à 100 000 (cf. <0.1millions contenu dans le dataset initial). En cross-checkant ces mêmes ratios à ceux présent sur les sites officiels, on peut affiner cette hypothèse et aboutir au vrai chiffre sur la population sous-alimentée par pays.



Question 19 : Écrivez les requêtes SQL permettant de connaître...

- Les 10 pays ayant le plus haut ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant, **puis** en termes de kcal par habitant.
- Pour l'année 2013, les 10 pays ayant le plus faible ratio **disponibilité alimentaire/habitant** en termes de protéines (en kg) par habitant.
- La quantité totale (en kg) de produits perdus par pays en 2013.
- Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.
- Les 10 produits pour lesquels le ratio **Autres utilisations/Disponibilité intérieure** est le plus élevé.



```

CREATE TABLE SQL_Q19_f_10_prod_part1 AS
SELECT produit,code_produit, CAST(SUM(autres_utilisations) AS REAL) AS others_Mt , CAST(SUM(dispo_int) AS REAL) AS dispint_Mt
FROM equilibre_prod
WHERE dispo_int AND autres_utilisations IS NOT NULL
GROUP BY code_produit
ORDER BY autres_utilisations DESC

CREATE TABLE SQL_Q19_f_10_prod_part2 AS
SELECT produit,code_produit,others_Mt,dispint_Mt, ROUND((100*(others_Mt/dispint_Mt)),1) AS ratio_pct
FROM SQL_Q19_f_10_prod_part1
WHERE others_Mt AND dispint_Mt IS NOT NULL
ORDER BY ratio_pct DESC
LIMIT 10
  
```

Veuillez à bien créer la table 'part1' pour requêter



	produit	code_produit	others_Mt	dispint_Mt	ratio_pct
1	Alcool, non Comestible	2659	21769	21741	100.1
2	Huil Plantes Oleif Autr	2586	4733	6271	75.5
3	Huile de Palmistes	2576	4625	6570	70.4
4	Huile de Palme	2577	35191	50409	69.8
5	Girofles	2642	84	130	64.6
6	Huile de Colza&Moutarde	2574	12909	23416	55.1
7	Graisses Animales Crue	2737	10792	22951	47
8	Huiles de Poissons	2781	371	828	44.8
9	Huile de Soja	2571	17463	41961	41.6
10	Plantes Aquatiques	2775	8934	23554	37.9

Exemple de Jointure

```
CREATE TABLE SQL_Q_JOINTURE_operation_part1 AS
SELECT * FROM population
NATURAL JOIN sous_nutrition
GROUP BY pays
HAVING population > 1e9;
```

	pays	code_pays	annee	population	année	nb_personnes
1	Arménie	1	2013	2977000	2012-2014	100000
2	Afghanistan	2	2013	30552000	2012-2014	7900000
3	Albanie	3	2013	3173000	2012-2014	200000
4	Algérie	4	2013	39208000	2012-2014	1700000
5	Angola	7	2013	21472000	2012-2014	8100000
6	Antigua-et-Barbuda	8	2013	90000	2012-2014	
7	Argentine	9	2013	41446000	2012-2014	1500000
8	Australie	10	2013	23343000	2012-2014	
9	Autriche	11	2013	8495000	2012-2014	
10	Bahamas	12	2013	377000	2012-2014	
	⋮			⋮		

	pays	code_pays	annee	population	année	nb_personnes
1	Inde	100	2013	1252140000	2012-2014	216300000
2	Chine, continentale	41	2013	1385567000	2012-2014	136600000

Annexes



Question 20 : pour quelques uns des produits identifiés dans cette dernière requête SQL, supposez quelles sont ces "autres utilisations" possibles (recherchez sur internet !)

Produits	Ratio (%)	Autres utilisations
Alcool, non Comestible	100.1	médical (antiseptique), désinfection
Huil Plantes Oleif Autr	75.5	
Huile de Palmistes	70.4	
Huile de Palme	69.8	80% pour l'alimentation, 19% en oléochimie (cosmétiques, etc), 1% en agrocarburant
Girofles	64.6	anti-inflammatoire , anesthésiant local et antiseptique efficace, agit contre les bactéries, les microbes et les virus
Huile de Colza&Moutarde	55.1	
Graisses Animales Crue	47	base du savon, cosmétique & parfumerie
Huiles de Poissons	44.8	
Huile de Soja	41.6	
Plantes Aquatiques	37.9	stabilisation physicochimique du milieu, support et abris à une multitude d'organismes aquatiques

Concaténation

```
url = 'INPUTS_FORMATION/fr_vegetaux.csv'
data = pd.read_csv(url, sep=',',encoding='utf-8')
df2 = data.copy()
#on supprime La ligne correspondant à Zone = Chine (cf. doublons)
df2 = df2.drop(df2[df2['Code zone'] ==351].index, inplace=False)

url = 'INPUTS_FORMATION/fr_animaux.csv'
data = pd.read_csv(url, sep=',',encoding='utf-8')
df3 = data.copy()
#on supprime La ligne correspondant à Zone = Chine (cf. doublons)
df3 = df3.drop(df3[df3['Code zone'] ==351].index, inplace=False)
```



Concaténation des datasets vegetx et animx de même structure

```
#on concatène les 2 datasets
frames = [df2, df3]
result = pd.concat(frames, keys=['vegetx','animx'])
```

Jointure



```
#on concatène les 2 datasets
frames = [df2, df3]
result = pd.concat(frames, keys=['vegetx','animx'])
```

```
url = 'INPUTS_FORMATION/fr_population.csv'
data = pd.read_csv(url, sep=',',encoding='utf-8')
df1 = data.copy()
#on supprime la Ligne correspondant à Zone = Chine (cf. doublons)
df1 = df1.drop(df1[df1['Code zone'] ==351].index, inplace=False)
```

Merge de ce df avec le dataset population

```
#on récupère l'info population mondiale que l'on intègre au df issu de la concaténation
table = pd.merge(result, df1, how="left", on="Zone")
#on renomme les []
table.rename(columns={'Code Domaine_x':'Code Domaine', 'Domaine_x':'Domaine', 'Code zone_x':'Code zone',
                     'Code Élément_x':'Code Élément', 'Élément_x':'Élément', 'Code Produit_x':'Code Produit',
                     'Produit_x':'Produit', 'Code année_x':'Code année', 'Année_x':'Année', 'Unité_x':'Unité', 'Valeur_x':'Valeur',
                     'Symbole_x':'Symbole', 'Description du Symbole_x':'Description du Symbole', 'Unité_y':'Unité_population',
                     'Valeur_y':'Valeur_populationx1000'}, inplace=True)
#on sélectionne les [] que l'on veut conserver
table=table[['Zone', 'Code zone', 'Élément', 'Année', 'Code Produit', 'Produit', 'Unité', 'Valeur', 'Unité_population',
             'Valeur_populationx1000']]
```

Conversion

Remplacement

Agrégation

Tri

```
#conversion numérique de [RATIO_KCAL_KG]
table['RATIO_KCAL_KG'] = pd.to_numeric(table['RATIO_KCAL_KG'])
#on remplace les 0 par des NaN pour ne pas fausser la moyenne
table['RATIO_KCAL_KG'] = table['RATIO_KCAL_KG'].replace(0 ,np.nan)
#agrégation & création d'une nouvelle table moyennant chaque produit sur tous les pays
table_agg = table.groupby(['Produit']).agg({'RATIO_KCAL_KG':np.mean})
#tri [RATIO_KCAL_KG] par ordre décroissant
table_agg_sorted_desc = table_agg.sort_values(['RATIO_KCAL_KG'], ascending=0)
```

Restrictions

```
#création set contenant les différents produits existant dans le dataset vegetaux
list_vegtx = tuple(df2['Produit'].unique())

n = len(table)
for i, row in table.iterrows():
    val_prod = table.iloc[i,table.columns.get_loc('Produit')] #on relève la valeur pour chaque ligne de [Produit]
    if val_prod in list_vegtx : #si valeur dans la liste des produits vegetaux précédemment établie
        #alors création [] = [RATIO_KCAL_KG] x [disponibilité intérieure]
        table.loc[i,'DISP_IN_MOND_KCAL_Q6'] = table.iloc[i,table.columns.get_loc('RATIO_KCAL_KG')]*table.iloc[i,table.columns.get_loc('Disponibilité intérieure')]*1000000
somme_q6_kca = table['DISP_IN_MOND_KCAL_Q6'].sum() #on somme les valeurs de cette []
print('\x1b[6;31;40m' + "Disponibilité intérieure mondiale (Q6) : %.2e" %somme_q6_kca + " kcal" + '\x1b[0m')
```

```
#determination des exportations de manioc en Thaïlande from df2 vegetaux
thai_exp = df2.loc[(df2['Code zone']==216) & (df2['Code Produit']==2532) & (df2['Code Élément']==5911),['Valeur']].sum()
print('\x1b[6;31;40m' + "exportations de manioc en Thaïlande : %.f" %thai_exp + " Milliers de tonnes"+ '\x1b[0m')
```

Basemap toolkit

Sous-alimentation dans le monde en 2013



```

# Extraction des data à visualiser
lat = df_vf['Latitude'].values
lon = df_vf['Longitude'].values
population = df_vf['Valeur_populationx1000'].values*1000
sous_alimentation = df_vf['Valeur_sousalx1e6'].values*1e6
pourcentage = sous_alimentation/population

# création [Pourcentage]
for i, row in df_vf.iterrows():
    try:
        df_vf.loc[i,'Pourcentage_ssal'] = (1e6*row['Valeur_sousalx1e6'])/(1e3*row['Valeur_populationx1000'])
    except ZeroDivisionError:
        df_vf.loc[i,'Pourcentage_ssal'] = 0

# Préparation du background de La map
fig = plt.figure(figsize=(20, 15))
# m = Basemap(projection='cyl',llcrnrlat=-45,urcrnrlat=45,llcrnrlon=-15,urcrnrlon=60,resolution='c') # by zone
m = Basemap(projection='cyl', resolution='c', lon_0=0) # whole map
m.etopo(scale=0.5, alpha=0.5)
m.shadedrelief()
m.drawcoastlines(color='gray')
m.drawcountries(color='gray')
m.drawstates(color='gray')

# Echelle de couleur population par pays/capitale + taille pour sous-alimentation
m.scatter(lon,lat,latlon=True,c=np.log10(population),s=15*100*pourcentage,cmap='Reds',edgecolors = 'k',alpha=0.7)

# création colorbar + Légende
# plt.colorbar(orientation="horizontal", label=r'$\log_{10}(\{\rm population\})$')
bornm = np.log10(1000*df_vf['Valeur_populationx1000'].min())
bornp = np.log10(1000*df_vf['Valeur_populationx1000'].max())
plt.clim(bornm,bornp)

# Affection Légende avec dummy points
for pourcentage in [25, 50, 75]:
    plt.scatter([],[],c='k', edgecolors = 'k', alpha=0.5, s=15*pourcentage, label=str(pourcentage) + ' %') #marker oversizing
plt.legend(scatterpoints=1, frameon=False,labelspacing=2, loc='lower left',title='Taux de population sous-alimenté en 2013');
plt.title("Sous-alimentation dans le monde en 2013",fontsize=20)

#affichage de qlq données significatives (nom zone)
df_vf_sample = df_vf[df_vf['Pourcentage_ssal'] > 0.40]
n=len(df_vf_sample)
for i in range (n) :
    longi = df_vf_sample.iloc[i,df_vf_sample.columns.get_loc('Longitude')]
    lati = df_vf_sample.iloc[i,df_vf_sample.columns.get_loc('Latitude')]
    zo = df_vf_sample.iloc[i,df_vf_sample.columns.get_loc('Zone')]
    plt.plot(longi,lati, 'ok', markersize=3)
    plt.text(longi,lati, zo, fontsize=10);

plt.text(-20,-60,'Coloré par population', fontsize=14, color='k')
plt.savefig('Documents/Dossier_AISSA/POLE_EMPLOI/OPENCLASSROOM/P3_mouacha_aissa/jpg/image_ssal.jpg',dpi=300)
plt.show()

```

Code export

Export table population

```
#on repart du dataset "population" (df1)
#on conserve les colonnes suivantes
df1=df1[['Zone','Code zone','Année','Valeur']]
#on renomme les colonnes pour coller à l'énoncé
df1.rename(columns={'Zone':'pays','Code zone':'code_pays','Année':'annee','Valeur':'population'}, inplace=True)
#conversion unité population
df1['population']=df1['population']*1000
#export du csv
df1.to_csv("Documents/Dossier_AISSA/POLE_EMPLOI/OPENCLASSROOM/P3_mouacha_aissa/table1_population.csv", index = False)
```

Code export

Export table dispo_alim

```
#on repart de la table issue de la Question 3 avec (table_q15 = table.copy())

#listing des types produits (vegetaux vs animaux) et création colonne [origin]
list_prod_v = df2['Code Produit'].unique()
list_prod_a = df3['Code Produit'].unique()

n = len(table_q15)
for i, row in table_q15.iterrows():
    #on relève la valeur pour chaque Ligne de [Produit]
    cod_prod = table_q15.iloc[i,table_q15.columns.get_loc('Code Produit')]
    #si valeur contenue dans la liste des céréales précédemment établie alors création [] contenant true or False
    if cod_prod in list_prod_v :
        table_q15.loc[i,'origin'] = 'vegetal'
    elif cod_prod in list_prod_a :
        table_q15.loc[i,'origin'] = 'animal'

#on conserve les colonnes suivantes
table_q15=table_q15[['Zone','Code zone','Année','Produit','Code Produit','origin','Dispo_TOT_KG',
                     'Disponibilité alimentaire (Kcal/personne/jour)', 'Disponibilité de protéines en quantité (g/personne/jour)',
                     'Disponibilité de matière grasse en quantité (g/personne/jour)']]

#on renomme les colonnes pour coller à l'énoncé
table_q15.rename(columns={'Zone':'pays','Code zone':'code_pays','Année':'année','Produit':'produit',
                          'Code Produit':'code_produit','Dispo_TOT_KG':'dispo_alim_tonnes',
                          'Disponibilité alimentaire (Kcal/personne/jour)':'dispo_alim_kcal_p_j',
                          'Disponibilité de protéines en quantité (g/personne/jour)':'dispo_prot',
                          'Disponibilité de matière grasse en quantité (g/personne/jour)':'dispo_mat_gr'}, inplace=True)

#conversion des kg en tonnes
table_q15['dispo_alim_tonnes']=table_q15['dispo_alim_tonnes']*1e-3

#export du csv
table_q15.to_csv("Documents/Dossier_AISSA/POLE_EMPLOI/OPENCLASSROOM/P3_mouacha_aissa/table2_dispo_alim.csv", index = False)
```

Code export

Export table équilibre_prod

```
#on repart de la table issue de la Question 3 avec (table_q16 = table.copy())

#on conserve les colonnes suivantes
table_q16=table_q16[['Zone','Code zone','Année','Produit','Code Produit','Disponibilité intérieure',
                     'Aliments pour animaux','Semences','Pertes','Traitement','Nourriture',
                     'Autres utilisations (non alimentaire)']]

#on renomme les colonnes pour coller à l'énoncé
table_q16.rename(columns={'Zone':'pays','Code zone':'code_pays','Année':'année','Produit':'produit','Code Produit':'code_produit'
                          'Disponibilité intérieure':'dispo_int',
                          'Aliments pour animaux':'alim_ani',
                          'Semences':'semences',
                          'Pertes':'pertes',
                          'Traitement':'transfo',
                          'Nourriture':'nourriture',
                          'Autres utilisations (non alimentaire)':'autres_utilisations'}, inplace=True)

#export du csv
table_q16.to_csv("Documents/Dossier_AISSA/POLE_EMPLOI/OPENCLASSROOM/P3_mouacha_aissa/table3_equilibre_prod.csv", index = False)
```

Code export

Export table sous_nutrition

```
#on repart du dataset "sous-alimentation" (df5)

#formatage + filtre sur l'année qui nous intéresse ici 2013
df5['Valeur'] = df5['Valeur'].replace('<0.1' , '0.1')
df5['Valeur'] = pd.to_numeric(df5['Valeur'])
df5_filter_year = df5.groupby(['Année']).get_group(('2012-2014'))

#on conserve les colonnes suivantes
df5_filter_year=df5_filter_year[['Zone','Code zone','Année','Valeur']]

#conversion unité population sous-alimentée
df5_filter_year['Valeur']=df5_filter_year['Valeur']*1e6

#on renomme les colonnes pour coller à l'énoncé
df5_filter_year.rename(columns={'Zone':'pays','Code zone':'code_pays','Année':'année','Valeur':'nb_personnes'}, inplace=True)

#export du csv
df5_filter_year.to_csv("Documents/Dossier_AISSA/POLE_EMPLOI/OPENCLASSROOM/P3_mouacha_aissa/table4_sous_nutrition.csv", index = False)
```

Merci

