

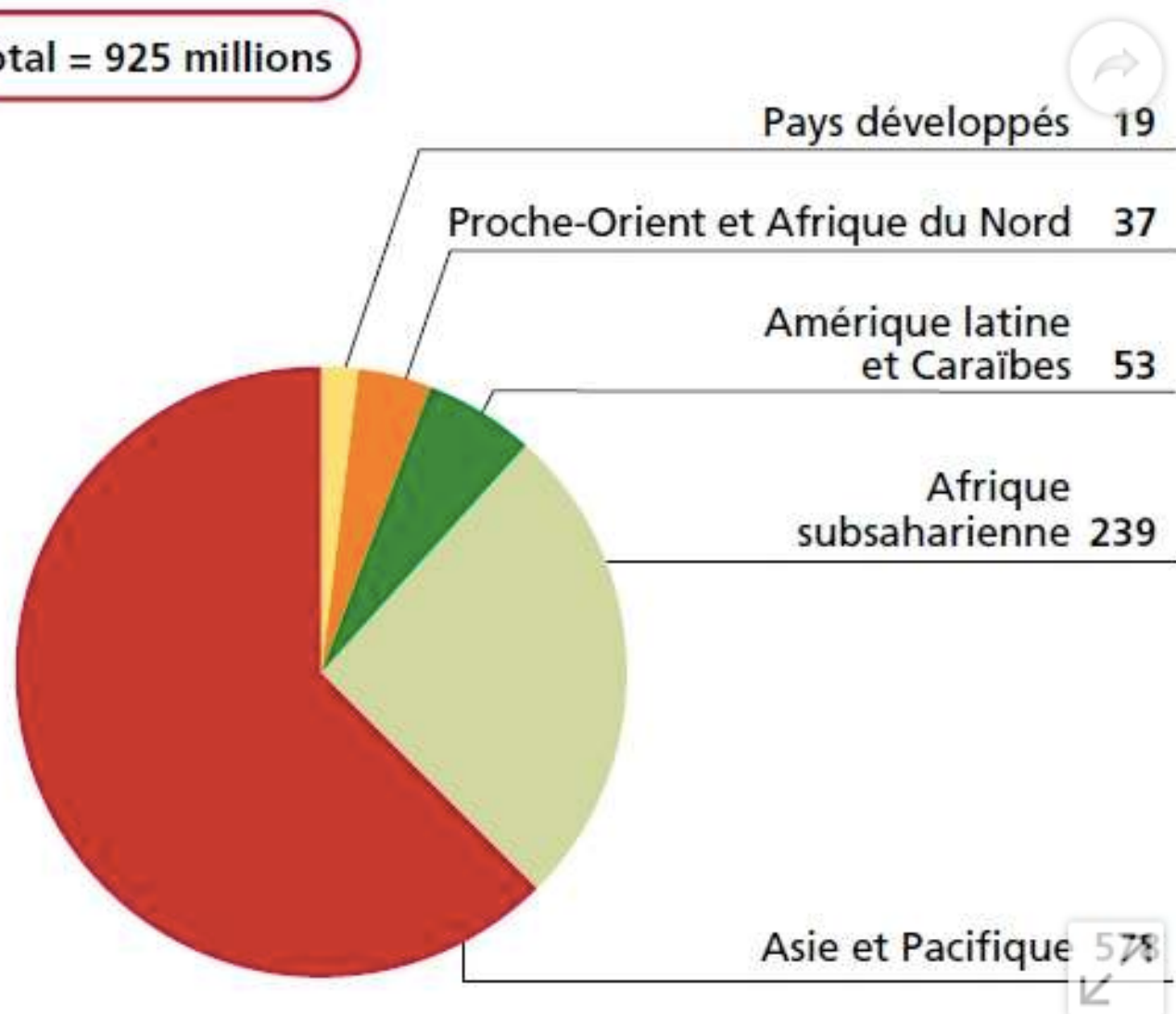


La question de la sous-nutrition dans le monde

Date

- ❖ Pour la FAO, la sous-alimentation est « définie comme le pourcentage de la population qui n'a pas accès à une quantité de nourriture ».
- ❖ En 2015, il y aurait eu 9,1 millions de morts dus à la malnutrition dans le monde.
- ❖ Contre 900 millions il y a quinze ans
- ❖ 815 millions de personnes demeurent sous-alimentées dans le monde en 2016.
- ❖ Ces résultats sont confirmés par nos calculs, car pour l'année 2016, nous considérons que 9,85% de la population mondiale était considérée comme étant en sous-nutrition.
- ❖ Si les ressources sont bien utilisées, on peut imaginer dans le futur une disparition de la sous-nutrition

Total = 925 millions



Répartition géographique des personnes sous-alimentées dans le monde en 2010. Les valeurs arrondies sont exprimées en millions. © FAO

- ❖ De nombreuses raisons expliquent les difficultés alimentaires :
- ❖ population mondiale en hausse constante
- ❖ catastrophes naturelles
- ❖ niveau d'éducation faible
- ❖ guerres
- ❖ corruption des élites politiques
- ❖ lobbying économique
- ❖ pandémies...

- ❖ Nous avons calculé la population mondiale pour les années 2013 et 2017. Celle-ci est en augmentation, elle passe de 7 213 211 612 à 7 550 036 044.
- ❖ Elle devrait atteindre 8,6 milliards en 2030, 9,8 milliards en 2050 et 11,2 milliards en 2100, selon rapport des Nations Unies
- ❖ Aura t'on besoin d'augmenter drastiquement la production alimentaire ?
- ❖ Pas obligatoirement
- ❖ Plusieurs études le confirment, on peut nourrir 9 milliards de personnes à l'horizon 2050. Les potentiels en surfaces agricoles et en rendement sont largement suffisants.
- ❖ Résultats question 7 : si toute la disponibilité intérieure mondiale de produits végétaux était utilisée pour de la nourriture, elle répondrait aux besoins en kcal de 1,87 fois la population mondiale et aux besoins en protéines de 1,57 fois la population mondiale.
- ❖ Nous ne manquerons donc pas de nourriture en 2050, mais à une condition : le modèle économique en cours doit changer.

- ❖ Infos sur les données téléchargées :
 - ❖ la source a été le site de la FAO
 - ❖ liste de fichiers téléchargés :
 - ❖ - population_2013.csv : infos relatives à la population mondiale triées par pays
 - ❖ - bilan_2013.csv : reprend toutes les infos relatives à la nourriture sur le plan international aussi bien au niveau de la production, de l'exportation, de la consommation, des pertes ...
 - ❖ - securite_2013.csv : reprend les infos relatives à la sous-nutrition dans le monde
 - ❖ - 2nd temps : bilan_veg : nous allons faire une sélection lors du téléchargement des données, dans le cas présent, nous nous ne sélectionnerons, que les infos relatives aux produits végétaux
 - ❖ les fichiers représentent des échantillons, sont organisés par attributs (= colonnes) et par lignes (= individus)
 - ❖ dans certains cas, nous serons amenés à faire un formatage du dataframe principal


```

In [9]: 1 # formatage du dataframe principal
2 bilan_2013 = bilan_2013.pivot_table(values = ["Valeur"], index = ["Code Pays", "Pays", "Code Produit", "Produit",
3                                     "Année"], columns = ["Élément"], aggfunc = sum)
4 bilan_2013.columns = ['Aliments pour animaux', 'Autres Utilisations',
5                       'Disponibilité alimentaire (Kcal/personne/jour)',
6                       'Disponibilité alimentaire en quantité (kg/personne/an)',
7                       'Disponibilité de matière grasse en quantité (g/personne/jour)',
8                       'Disponibilité de protéines en quantité (g/personne/jour)', 'Disponibilité intérieure',
9                       'Exportations - Quantité', 'Importations - Quantité', 'Nourriture', 'Pertes',
10                      'Population totale', 'Production', 'Semences', 'Traitement', 'Variation de stock']
11 bilan_2013 = bilan_2013.reset_index()
12 bilan_2013.head()

```

Out[9]:

	Code Pays	Pays	Code Produit	Produit	Année	Aliments pour animaux	Autres Utilisations	Disponibilité alimentaire (Kcal/personne/jour)	Disponibilité alimentaire en quantité (kg/personne/an)	Disponibilité de matière grasse en quantité (g/personne/jour)	...	Disponibilité intérieure	Exportations - Quantité	Imp
0	1	Arménie	2501	Population	2013	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	
1	1	Arménie	2511	Blé	2013	93.0	0.0	1024.0	130.60	3.6	...	554.0	1.0	
2	1	Arménie	2513	Orge	2013	137.0	26.0	0.0	0.00	0.0	...	198.0	0.0	
3	1	Arménie	2514	Maïs	2013	96.0	NaN	0.0	0.03	NaN	...	102.0	NaN	
4	1	Arménie	2515	Seigle	2013	1.0	NaN	1.0	0.12	0.0	...	1.0	NaN	

5 rows x 21 columns

1 agrégation

```
In [22]: 1 # création d'un nouveau dataframe où les lignes égales à 0 dans la variable 'Ratio kcal/kg' sont supprimées
2 bilan_2013_cal = bilan_2013.loc[ bilan_2013['Ratio "énergie/poids" (kcal/kg)'] != 0, :]
```

```
In [23]: 1 # création d'un objet de type DataFrameGroupBy, que nous appelons 'gb', celui-ci prend en attribut de
2 # partitionnement la variable 'Produit' de la table 'bilan_2013_cal'
3 gb = bilan_2013_cal.groupby('Produit')
```

```
In [24]: 1 # sur cet objet gb, on peut ensuite appliquer la fonction d'agrégation .mean()
2 m = gb['Ratio "énergie/poids" (kcal/kg)',].mean()
3
4 # représentation du dataframe
5 grouppe = pd.concat([m], axis=1)
6 grouppe.columns = ['moy Ratio "énergie/poids" (kcal/kg)']
```

```
In [25]: 1 # tri descendant au sein du dataframe donnant les 20 aliments les plus caloriques
2 grouppe.sort_values(by = 'moy Ratio "énergie/poids" (kcal/kg)', ascending = False)[0:20]
```

Out[25]:

	moy Ratio "énergie/poids" (kcal/kg)
Produit	
Huiles de Foie de Poisso	9982.932500
Huile de Sésame	9433.744729
Huile de Son de Riz	9361.549883
Huiles de Poissons	9003.715504
Huile d'Arachide	8935.765811
Huile d'Olive	8931.223447
Huile de Germe de Maïs	8871.762566
Huil Plantes Oleif Autr	8848.222893
Huile de Palmistes	8789.540277

1 jointure interne

ers (SQLite 3)
lite 3)

_alim
ore_prod
ation
nutrition

Requête Historique

```
1 SELECT pays, annee, (nb_personnes/population) as ratio
2 FROM population
3 INNER JOIN sous_nutrition
4 ON (population.pays=sous_nutrition.pays)
5 AND (population.code_pays=sous_nutrition.code_pays)
6 AND (population.annee=sous_nutrition.annee)
7 WHERE population.annee = '2013'
8 ORDER BY ratio DESC LIMIT 10
```

Table Formulaire

Nombre de lignes chargées : 10

	pays	annee	ratio
1	Haïti	2013	0.49847696116174
2	République centrafricaine	2013	0.46670265462692
3	Zambie	2013	0.4619483264602
4	Zimbabwe	2013	0.44504947555237
5	République populaire démocratique de Corée	2013	0.41623349033874
6	Congo	2013	0.37883626970028
7	Libéria	2013	0.37328310187059
8	Tchad	2013	0.37308918377147
9	Madagascar	2013	0.35712503199971
10	Ouganda	2013	0.33551930373034

1 restriction

```
9 # suppression de l'individu Chine en doublon  
10 bilan_2013 = bilan_2013.loc[ bilan_2013.Pays != 'Chine', :]  
11
```


Les 10 pays ayant le plus haut ratio disponibilité alimentaire/habitant en termes de protéines (en kg) par habitant.

Requete Historique

```
1 SELECT pays,sum((dispo_prot/1000)*365) as dispo_prot_modif
2 FROM dispo_alim
3 GROUP BY pays
4 ORDER BY dispo_prot_modif DESC LIMIT 10
```

Table Formulaire

Nombre de lignes chargées : 10

	pays	dispo_prot_modif
1	Islande	48.566900000000001
2	Chine - RAS de Hong-Kong	47.110550000000001
3	Israël	46.719999999999999
4	Lituanie	45.3914
5	Maldives	44.646800000000001
6	Finlande	42.909400000000001
7	Luxembourg	41.478600000000001
8	Monténégro	40.8435
9	Pays-Bas	40.682900000000002
10	Albanie	40.65005

Les 10 pays ayant le plus haut ratio disponibilité alimentaire/habitant en termes de kcal par habitant.

```
1 SELECT pays,sum(dispo_alim_kcal_p_j*365) as dispo_alim_modif
2 FROM dispo_alim
3 GROUP BY pays
4 ORDER BY dispo_alim_modif DESC LIMIT 10
```

Table Formulaire

Nombre de lignes chargées : 10

	pays	dispo_alim_modif
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

Pour chaque année disponible, les 10 pays ayant le plus faible ratio disponibilité alimentaire/habitant en termes de protéines (en kg) par habitant. Le nombre de lignes de la table renvoyée sera donc égal à 10 fois le nombre d'années

Requete

Historique

1 SELECT annee, pays, sum((dispo_prot/1000)*365) as dispo_prot_modif

2 FROM dispo_alim

3 GROUP BY annee, pays

4 ORDER BY dispo_prot_modif ASC LIMIT 10

Table

Formulaire

1

Nombre de lignes chargées : 10

	annee	pays	dispo_prot_modif
1	2013	Libéria	13.7459
2	2013	Guinée-Bissau	16.07825
3	2013	Mozambique	16.6732
4	2013	République centrafricaine	16.8046
5	2013	Madagascar	17.04185
6	2013	Haïti	17.410500000000001
7	2013	Zimbabwe	17.636799999999999
8	2013	Congo	18.76465
9	2013	Ouganda	19.213600000000001
10	2013	Sao Tomé-et-Principe	19.3815

La quantité totale (en kg) de produits perdus par pays et par année. La table renvoyée contiendra donc une ligne par couple (pays, année).

```
1 SELECT pays, annee, sum(pertes) as pertes
2 FROM equilibre_prod
3 GROUP BY pays, annee
```

Table Formulaire

Nombre de lignes chargées : 174

	pays	annee	pertes
1	Afghanistan	2013	1135000000.
2	Afrique du Sud	2013	2193000000.
3	Albanie	2013	276000000.
4	Algérie	2013	3753000000.
5	Allemagne	2013	3781000000.
6	Angola	2013	4799000000.
7	Antigua-et-Barbuda	2013	0
8	Arabie saoudite	2013	1040000000.
9	Argentine	2013	3522000000.
10	Arménie	2013	228000000.
11	Australie	2013	520000000.
12	Autriche	2013	459000000.
13	Azerbaïdjan	2013	232000000.

Les 10 pays pour lesquels la proportion de personnes sous-alimentées est la plus forte.

```
1 SELECT pays, annee, (nb_personnes/population) as ratio
2 FROM population
3 INNER JOIN sous_nutrition
4 ON (population.pays=sous_nutrition.pays)
5 AND (population.code_pays=sous_nutrition.code_pays)
6 AND (population.annee=sous_nutrition.annee)
7 WHERE population.annee = '2013'
8 ORDER BY ratio DESC LIMIT 10
```

Table Formulaire

Nombre de lignes chargées : 10

	pays	annee	ratio
1	Haïti	2013	0.49847696116174
2	République centrafricaine	2013	0.46670265462692
3	Zambie	2013	0.4619483264602
4	Zimbabwe	2013	0.44504947555237
5	République populaire démocratique de Corée	2013	0.41623349033874
6	Congo	2013	0.37883626970028
7	Libéria	2013	0.37328310187059
8	Tchad	2013	0.37308918377147
9	Madagascar	2013	0.35712503199971
10	Ouganda	2013	0.33551930373034

Les 10 produits pour lesquels le ratio Autres utilisations/Disponibilité intérieure est le plus élevé.

Requête

Historique

1 SELECT produit, AVG(autres_utilisations/dispo_int) as ratio

2 FROM equilibre_prod

3 GROUP BY produit

4 ORDER BY ratio DESC LIMIT 10

Table

Formulaire

✓

✗

⏪

⏩

1

⏴

⏵

🖨

Nombre de lignes chargées : 10

	produit	ratio
1	Alcool, non Comestible	0.96551724137931
2	Plantes Aquatiques	0.92066115702479
3	Huile de Palme	0.65252019281057
4	Huil Plantes Oleif Autr	0.54369391758518
5	Huile de Palmistes	0.53418347963942
6	Huile de Colza&Moutarde	0.46264905520177
7	Huiles de Poissons	0.4027338041343
8	Huile de Coco	0.36329602348459
9	Graisses Animales Crue	0.30480920784929
10	Manioc	0.23284581146062

- ❖ Question 20 : pour quelques uns des produits identifiés dans cette dernière requête SQL, supposez quelles sont ces "autres utilisations" possibles (recherchez sur internet !).
- ❖ - alcool : antiseptique, désinfectant
- ❖ - plante aquatique : décoration dans les aquariums
- ❖ huile de palme : production d'agrocarburant
- ❖ huile plante : huile essentielle
- ❖ huile de palmiste : fabrication d'aliments transformés (beurre, saindoux ...)
- ❖ huile de colza : biocarburant
- ❖ huile de poissons : industrie pharmaceutique
- ❖ huile de coco : produit comestique
- ❖ graisse animale crue : cosmétique et parfumerie