

Faim dans le monde +9 millions de morts par an

Etude de santé publique FAO

Projet 3 - Nalron Septembre 2019

OpenClassrooms - ENSAE-ENSAI Formation Continue

Population Mondiale

Une vague d'humanité...

2013

6,9
Milliards

2019

7,7
Milliards

2050

9,8
Milliards



Population en sous-nutrition

Et demain ?

+12% de la population mondiale est en sous-nutrition.

880
Millions

2013

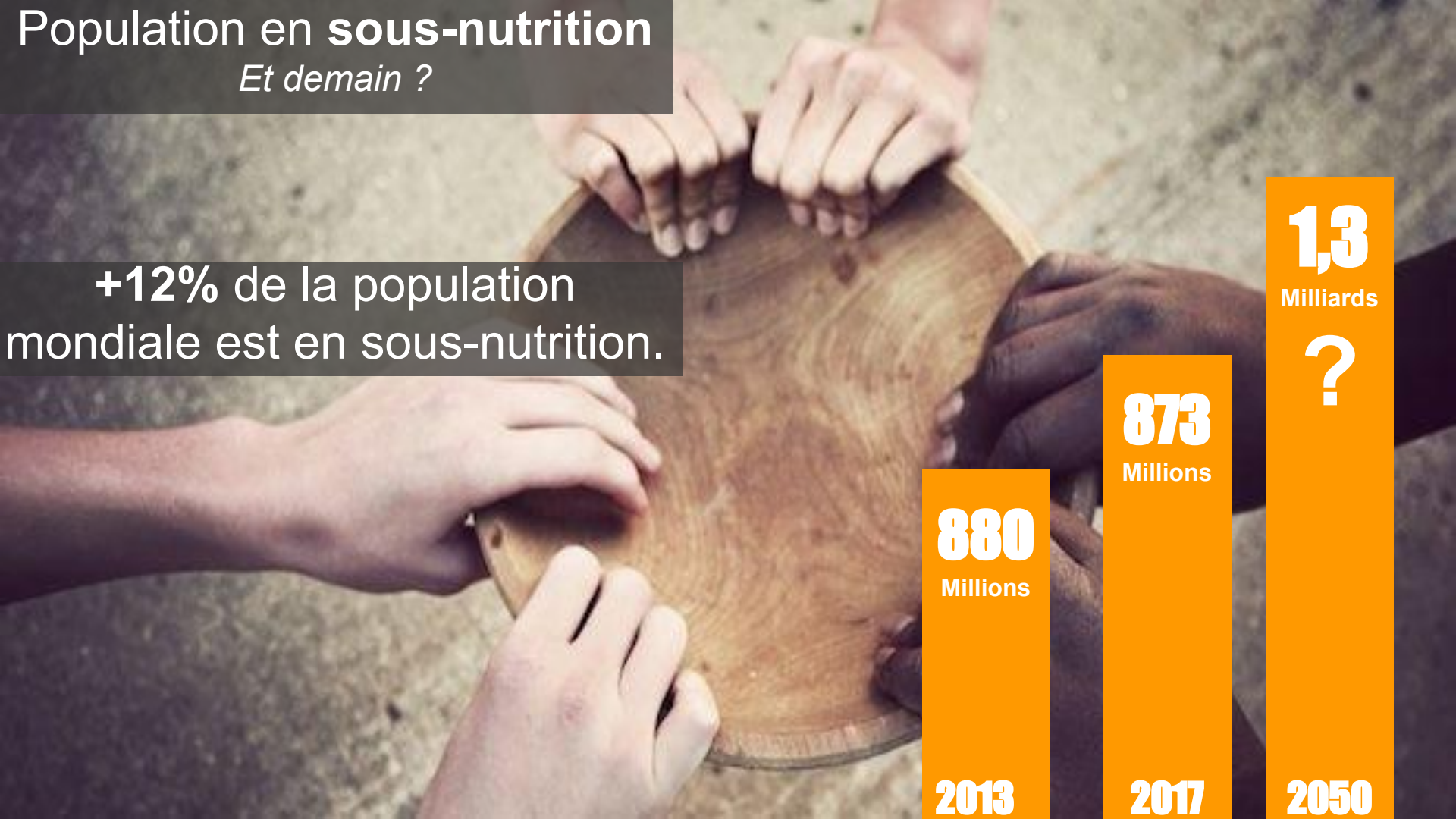
873
Millions

2017

1,3
Milliards

?

2050



Définition de l'Insécurité Alimentaire (source FAO):

Capacité de tout temps de fournir à toute personne un accès physique, social et économique à une alimentation suffisante, saine et nutritive afin de satisfaire ses besoins et préférences alimentaires pour une vie active et saine.

Plusieurs causes de sous-nutrition :

- . Pauvreté
- . Problèmes d'accès à la nourriture
- . Maladies
- . Conflits
- . Changements climatiques
- . Manque d'eau potable


Disponibilité alimentaire mondiale cause ou atout ?

La disponibilité est nécessaire. La disponibilité est nécessaire mais insuffisante pour assurer l'accès, qui est également nécessaire mais insuffisant pour assurer une utilisation efficace.



+8 milliards d'individus auraient pu être nourris uniquement avec la **disponibilité alimentaire mondiale** sur l'année 2013.

**Disponibilité intérieure mondiale
issue de produits végétaux.**



+14 milliards d'individus auraient pu être nourris
en 2013. Soit **plus de 200%** de la population !

Une **Production suffisante** mais pas efficace,
alors que faire ???



+1 milliard d'adultes en surpoids
+300 millions sont obèses

source : OMS, FAO



Une situation intolérable !

- > Changer nos habitudes...
- > Manger autrement et de façon durable



36% de la disponibilité mondiale des céréales est destinée à l'alimentation animale.

Exemples d'un déséquilibre des répartitions :

- 3 calories de céréales pour 1 calorie de volaille.
- 30 Kg de céréales pour 1 Kg de viande de bœuf comestible.



Réduire notre consommation en protéines animales: une nécessité !

USA un exemple
saisissant...

**Si baisse de 10% :
Production animale**



**+14 milliards de tonnes de
céréales disponibles !**

Données utilisées pour l'étude, site de la FAO :

<http://www.fao.org/faostat/fr/#data>

Extraction des données en 5 fichiers .csv :

- > Population mondiale en 2013
- > Bilan alimentaire animal, végétal et cérééal
- > Sécurité alimentaire (population en sous-nutrition)

DOWNLOAD DATAVISUALIZE DATAMETADATAREPORT

PAYS

RÉGIONS

GROUPES SPÉCIAUX

⚙

Q

Filterer les résultats par ex. afghanistan

☐

Afghanistan

☐

Afrique du Sud

☐

Albanie

☐

Algérie

☐

Allemagne

☐

Angola

Tout sélectionner

Tout effacer

PRODUITS

GROUPES DE PRODUITS

Q

Filterer les résultats par ex. population

☐

Population

☐

Blé

☐

Riz (Eq Blanc)☐☐☐

Tout sélectionner

Tout effacer

ÉLÉMENTS

Q

Filterer les résultats par ex. population totale

☐

Population totale

☐

Production - Quantité

☐

Importations - Quantité☐☐☐

Tout sélectionner

Tout effacer

ANNÉES

Q

Filterer les résultats par ex. 2013

☐

2013

☐

2012

☐

2011☐☐☐

Tout sélectionner

Tout effacer

Bilans Alimentaires

Food Balance Sheet presents a comprehensive picture of the pattern of a country's food supply during a specified reference period. The food balance... [Show More](#)

Food and Agriculture Organization of the United Nations (FAO)

⬇

Bulk Downloads

Toutes les Données

25.65 MB

Toutes les Données Normalisé

70.32 MB

Tous les groupes de pays

3.39 MB

Afrique

2.72 MB

Amériques

2.53 MB

Asie

4.66 MB

Europe

2.94 MB

Océanie

503 KB

Dernière mise à jour

12 décembre 2017

Documentation

📄

Update history

☰

Definitions and standa...


```
# Chargement du dataframe de la population mondiale, fichier 'population.csv'
population = pd.read_csv('population.csv')
population.head()
```

	Code Domaine	Domaine	Code Pays	Pays	Code Élément	Élément	Code Produit	Produit	Code Année	Année	Unité	Valeur	Symbole	Description du Symbole
0	FBS	Bilans Alimentaires	2	Afghanistan	511	Population totale	2501	Population	2013	2013	1000 personnes	30552	NaN	Donnée officielle
1	FBS	Bilans Alimentaires	202	Afrique du Sud	511	Population totale	2501	Population	2013	2013	1000 personnes	52776	NaN	Donnée officielle

Dataframe des données (disponibilité alimentaire, pertes, ratio énergétique) par Pays & Produits

Élément	index	Pays	Produit	Origine	Disponibilité alimentaire (Kcal/an)	Disponibilité alimentaire en quantité (Kg/an)	Disponibilité de protéines en quantité (Kg/an)	Aliments pour animaux	Pertes	Disponibilité intérieure Kg	Ratio énergie/poids (Kcal/Kg)
0	0	Arménie	Blé	Végétale	1.112684e+09	388796.20	33163.18460	93.0	32.0	554000000.0	2861.868300
2	2	Arménie	Maïs	Végétale	0.000000e+00	89.31	10.86605	96.0	7.0	102000000.0	0.000000
3	3	Arménie	Seigle	Végétale	1.086605e+06	357.24	21.73210	1.0	0.0	1000000.0	3041.666667

Création d'un dataframe (cereal)à partir des données extraites de la FAO

	Code Domaine	Domaine	Code Pays	Pays	Code Élément	Élément	Code Produit	Produit	Code Année	Année	Unité	Valeur	Symbole	Description du Symbole
0	FBS	Bilans Alimentaires	5000	Monde	5511	Production	2511	Blé	2013	2013	Milliers de tonnes	708443	A	Agrégat, peut inclure des données officielles, semi-officielles, estimées ou calculées
1	FBS	Bilans Alimentaires	5000	Monde	5511	Production	2805	Riz (Eq Blanchi)	2013	2013	Milliers de tonnes	495447	A	Agrégat, peut inclure des données officielles, semi-officielles, estimées ou calculées

L'agrégation m'a permis d'effectuer facilement des opérations sur l'ensemble des éléments Series et Dataframes.

> Exemple d'un `.groupby()` pour agréger les moyennes des protéines par produits.

5.2 Liste des 5 aliments parmi les 20 aliments les plus riches en protéines

```
# Création du dataframe (df32) par agrégation  
# Retrait des valeurs 0 pour éviter de fausser le calcul de la moyenne  
df32 = df2.groupby(['Produit']).mean()  
df32 = df32.replace([np.inf, -np.inf], np.nan).dropna(subset=["%Protéines"], how="all")  
df32 = df32['%Protéines'].sort_values(ascending=False)  
df32.head(20)
```

Produit	
Soja	25.338211
Arachides Decortiquees	24.790816
Pois	22.108112
Légumineuses Autres	22.025876
Haricots	21.719055
Sésame	17.828134
Abats Comestible	17.630901

> Exemple d'un `.pivot_table()` pour agréger les valeurs des disponibilités alimentaires par pays et par produits.

3. Disponibilité alimentaire en kcal et kg de protéines pour chaque pays et chaque produit

```
# Synthèse des données du df pour obtenir les valeurs de disponibilité alimentaire... par pays et produit
# .pivot_table() pour synthétiser et agréger les résultats en somme
df1 = df.pivot_table(index=['Code Pays', 'Pays', 'Code Produit', 'Produit', 'Origine', 'Population', 'Année'], \
                      columns='Élément', values='Valeur', aggfunc=sum).reset_index()
df1.head()
```

Élément	Code Pays	Pays	Code Produit	Produit	Origine	Population	Année	Aliments pour animaux	Autres Utilisations	Disponibilité alimentaire (Kcal/personne/jour)	...	Disponibilité de protéines en quantité (g/personne/jour)	Disponibilité intérieure	Exportat - Quali
0	1	Arménie	2511	Blé	Végétale	2977	2013	93.0	0.0	1024.0	...	30.52	554.0	
1	1	Arménie	2513	Orge	Végétale	2977	2013	137.0	26.0	0.0	...	0.00	198.0	
2	1	Arménie	2514	Maïs	Végétale	2977	2013	96.0	NaN	0.0	...	0.01	102.0	

La **Restriction** m'a permis de sélectionner que **certaines lignes** d'un dataframe.

> Exemple d'une **Restriction** sur condition booléenne qui permet d'obtenir uniquement les produits céréaliers. Ou encore selon une année choisie.

11 : Quelle proportion (en termes de poids) est destinée à l'alimentation animale ?

En ne prenant en compte que les céréales destinées à l'alimentation (humaine et animale).

```
# Création du dataframe (df100) selon la restriction des produits céréaliers
df100 = df1[df1.is_cereal == True]
```

11.4 Sélection des données bilans alimentaires concernant uniquement les pays qui recensent des personnes en sous-nutrition (source FAO).

```
# Création d'un dataframe complet selon les conditions restrictives des personnes en sous-nutrition
# Réduction du dataframe relatif à la sous-nutrition (df_sousnut) sur l'année 2013
df111 = df_sousnut[df_sousnut['Code année'] == 20122014]
df111.head()
```

	Code Domaine	Domaine	Code zone	Zone	Code Élément	Élément	Code Produit	Produit	Code année	Année	Unité	Valeur	Symbole	Description du Symbole	Note
0	FS	Données de la sécurité alimentaire	2	Afghanistan	6132	Valeur	210011	Nombre de personnes sous- alimentées (millions) (moyenne sur 3 ans)	20122014	2012- 2014	millions	7.9	F	Estimation FAO	NaN

La **Projection** m'a permis de sélectionner que **certaines colonnes** d'un dataframe.

> Exemple d'une **Projection** pour **réduire et filtrer** uniquement les colonnes *Pays*, *Produits* et *Importations*.

```
# Projection sur les catégories Pays, Produit et Importations
# Mais pour la suite de l'étude nous devons conserver un df avec une projection plus large...
df113 = df113[['Pays', 'Produit', 'Importations - Quantité', 'Autres Utilisations', 'Disponibilité intérieure', 'Alimen'
```

```
# Filtre sur les quantités importées
# Sélection des 200 plus grandes importations sur ces produits
df113 = df113.sort_values(by='Importations - Quantité', ascending=False)[:200]
df1130 = df113[['Pays', 'Produit', 'Importations - Quantité']].reset_index()
df1130
```

Élément	index	Pays	Produit	Importations - Quantité
0	2509	Chine, continentale	Soja	63381.0
1	2496	Chine, continentale	Manioc	29046.0
2	6965	Japon	Maïs	14403.0

La **Jointure** m'a permis d'assembler deux dataframes selon une condition commune pays.

> Exemple d'une **Jointure Interne** selon la condition Pays.

```
# Méthode .append puis .merge pour agréger nos données produits à celles de la population  
df = vegetal.append(animal)  
df = pd.merge(df, population, on='Pays')  
df.head()
```

	Code Domaine	Domaine	Code Pays_x	Pays	Code Élément	Élément	Code Produit	Produit	Code Année	Année_x	Unité	Valeur_x	Symbole	Description du Symbole	Origine	Ci Pay
0	FBS	Bilans Alimentaires	2	Afghanistan	5511	Production	2511	Blé	2013	2013	Milliers de tonnes	5169.0	S	Données standardisées	Végétale	
1	FBS	Bilans Alimentaires	2	Afghanistan	5611	Importations - Quantité	2511	Blé	2013	2013	Milliers de tonnes	1173.0	S	Données standardisées	Végétale	
2	FBS	Bilans Alimentaires	2	Afghanistan	5072	Variation de stock	2511	Blé	2013	2013	Milliers de tonnes	-350.0	S	Données standardisées	Végétale	

Intégration et formatage des données en base de données relationnelle SQLite :

> Utilisation de la librairie Python pour créer la connection avec SQLite.

```
import sqlite3
conn = sqlite3.connect('base_fao.db')
c = conn.cursor()
```

> Préparation des données à partir des dataframes sous Python.

```
# Préparation de la table population à partir du dataframe (population)
df_pop = population
df_pop = df_pop[['Code Pays', 'Pays', 'Année', 'Valeur']].sort_values(by='Code Pays')
df_pop = df_pop.rename(columns={'Code Pays': 'code_pays', 'Pays': 'pays', 'Année': 'annee', 'Valeur': 'population'})
```

>>> La clé primaire retenue pour la table population sera le **code pays**.

> Création / Intégration des **tables** *population*, *dispo_alim*, *equilibre_prod*, *sous_nutrition* via Python.

```
# Création de la table population
c.execute('''CREATE TABLE population(
            code_pays INTEGER PRIMARY KEY,
            pays TEXT,
            annee INTEGER,
            population REAL)''')
conn.commit()
```

```
df_pop.to_sql('population', conn, if_exists='append', index=False)
```


Requête SQL qui rassemble plusieurs règles d'algèbre relationnelle :

- > **Projection** des données par pays via **SELECT**.
- > **Restriction** des données par année via **WHERE**.
- > **Agrégation** des données par pays via la clause **GROUP BY**.
- > **Clé primaire** de la table `dispo_alim` : `id` (clé artificielle).

19.1 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

```
# Ratio disponibilité alimentaire en protéines (en kg et kcal) par habitant
c.execute("""
SELECT pays, ROUND(SUM(dispo_prot)/1000, 2) AS dispo_prot_kg, SUM(dispo_alim_kcal_p_j) AS dispo_kcal
FROM dispo_alim
WHERE année = 2013
GROUP BY pays
ORDER BY(dispo_prot_kg)
DESC LIMIT 10
""")
c.fetchall()
```

```
[('Chine, continentale', 49587.02, 3112.0),
 ('Inde', 27380.73, 2454.0),
 ('États-Unis d'Amérique', 12782.29, 3682.0),
 ('Brésil', 6948.28, 3262.0),
 ('Indonésie', 5672.71, 2776.0),
 ('Fédération de Russie', 5359.94, 3360.0),
 ('Pakistan', 4351.26, 2438.0),
 ('Japon', 4067.62, 2726.0),
 ('Nigéria', 4041.07, 2700.0),
 ('Mexique', 3910.55, 3068.0)]
```

Requête SQL pour obtenir les 10 pays dont le ratio Dispo. alimentaire / Habitant (Kg protéines) est le plus faible.

19.2 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 disponibles

```
c.execute("""
SELECT PAYS, ROUND(SUM(dispo_prot)/1000, 2) AS dispo_prot_kg
FROM dispo_alim
WHERE année = 2013
GROUP BY pays
ORDER BY(dispo_prot_kg)
ASC LIMIT 10
""").fetchall()
```

```
[('Saint-Kitts-et-Nevis', 1.43),
 ('Dominique', 2.0),
 ('Bermudes', 2.14),
 ('Grenade', 2.68),
 ('Kiribati', 2.73),
 ('Antigua-et-Barbuda', 2.74),
 ('Saint-Vincent-et-les Grenadines', 3.43),
 ('Sao Tomé-et-Principe', 3.74),
 ('Sainte-Lucie', 5.7),
 ('Samoa', 5.93)]
```

Requête SQL pour obtenir les produits perdus par pays sur l'année 2013.

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

```
[('Chine, continentale', 89575.0),
 ('Brésil', 75914.0),
 ('Inde', 55930.0),
 ('Nigéria', 19854.0),
 ('Indonésie', 13081.0),
 ('Turquie', 12036.0),
 ('Mexique', 8289.0),
 ('Égypte', 7608.0),
 ('Ghana', 7442.0),
 ('États-Unis d'Amérique", 7162.0),
 ('Viet Nam', 6743.0),
 ('Pakistan', 5897.0),
 ('Thaïlande', 5749.0),
 ('Iran (République islamique d')", 5450.0),
 ('Fédération de Russie', 4997.0),
 ('Angola', 4799.0),
 ('Ukraine', 4138.0),
 ('Bangladesh', 4080.0),
 ('Cameroun', 3981.0),
 ('Allemagne', 3781.0)]
```


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

```
[('Inde', 1020.7),  
 ('Chine', 638.1999999999999),  
 ('Chine, continentale', 633.1),  
 ('Pakistan', 196.0),  
 ('Bangladesh', 126.10000000000001),  
 ('Éthiopie', 115.29999999999998),  
 ('Indonésie', 111.7),  
 ('Nigéria', 93.0),  
 ('République-Unie de Tanzanie', 84.9),  
 ('Ouganda', 75.4)]
```

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

```
[('Alcool, non Comestible', 112.0),  
 ('Huile de Palme', 96.57),  
 ('Huile Plantes Oleif Autr', 85.36),  
 ('Graisses Animales Crue', 46.64),  
 ('Huile de Palmistes', 44.87),  
 ('Huile de Colza&Moutarde', 42.56),  
 ('Plantes Aquatiques', 40.51),  
 ('Huile de Coco', 35.97),  
 ('Manioc', 28.17),  
 ('Edulcorants Autres', 23.44)]
```

Huile de Palme

Autres utilisations possibles :

- > Usages Industriels
- > Oléochimie
- > Agro-carburants



Huile de Coco

Autres utilisations possibles :

- > Cosmétique
- > Hygiène (dentifrice)
- > Préparation de cordes en jute



«Notre maison brûle et nous regardons ailleurs...»

Jacques Chirac

Notre maison brûle et nous regardons ailleurs. La nature, mutilée, surexploitée, ne parvient plus à se reconstituer, et nous refusons de l'admettre. L'humanité souffre. Elle souffre de mal-développement, au Nord comme au Sud, et nous sommes indifférents. La Terre et l'humanité sont en péril, et nous en sommes tous responsables.