OPENCLASSROOMS

Projet 3

Concevez une application au service de la santé publique

Kévin

Parcours Data Scientist

Contexte

- Appel à projet par Santé publique France
- Les enjeux de l’alimentation
- Essentiel pour notre santé

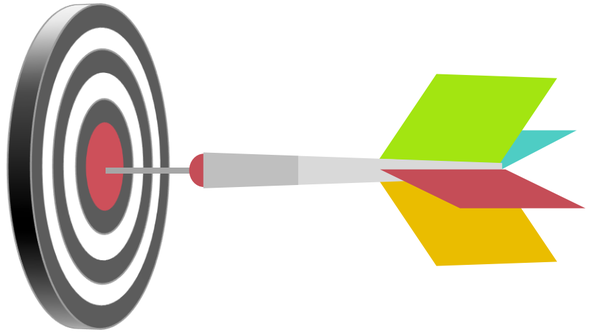
Application

Expérience personnelle

- Objectif de perdre du poids
- Limité par mes connaissances
- Lassitude
- Difficulté à interpreter les valeurs nutritionnelles

Apports

- Comparer les produits
- Répondre à un besoin



Jeu de données et nettoyage

Exploration et Exploitation

Résultats

1

Jeu de données et nettoyage

Jeu de données

Source

Open Food Facts

Format

320 772 lignes

162 variables

Diversité des données

- Informations générales du produit
- Tags
- Ingrédients
- Informations nutritionnelles

Produits

221 347 produits uniques

Catégories

36 982 catégories

Marques

58 784 marques

Données manquantes

Produits

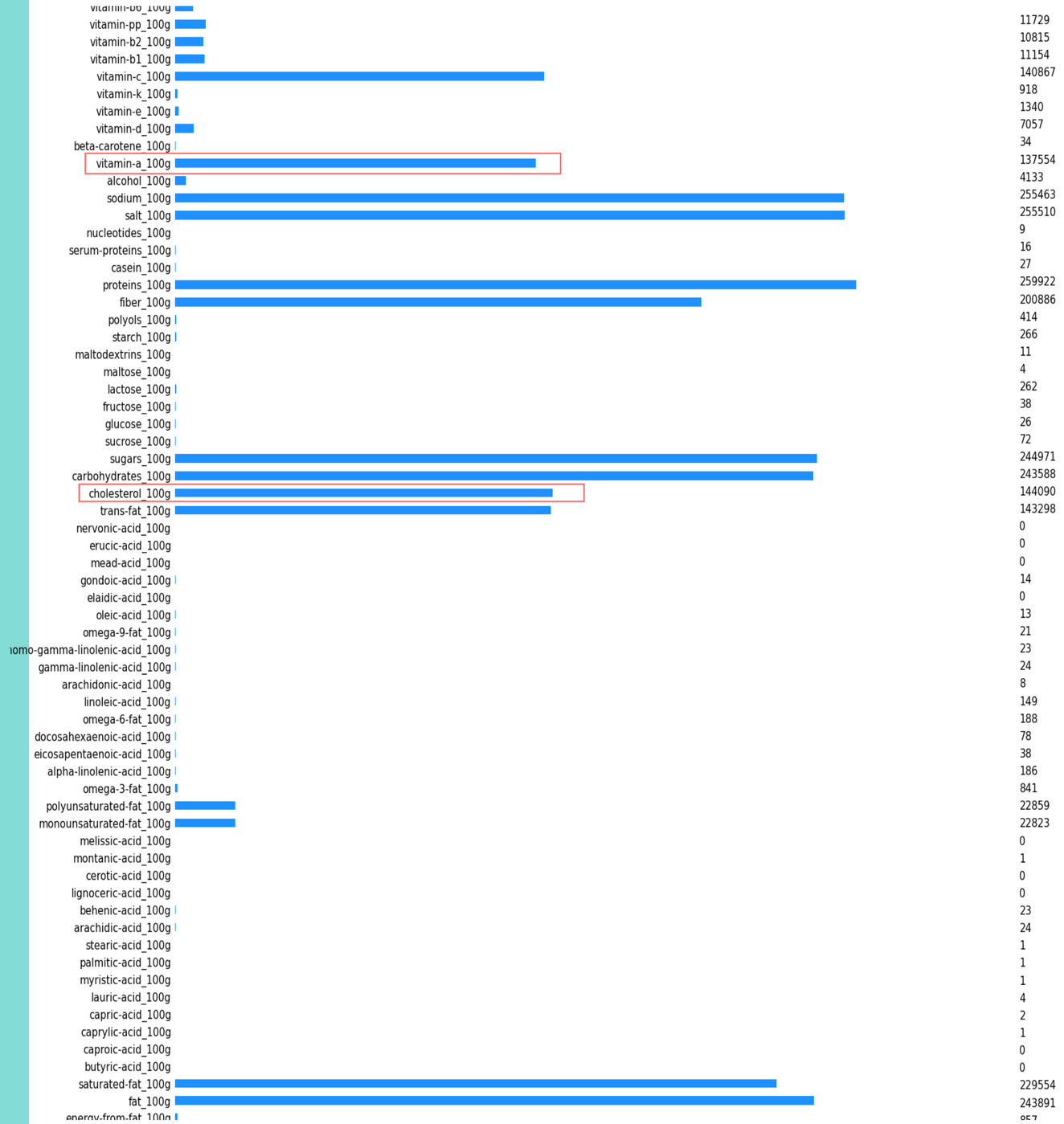
320 772 produits

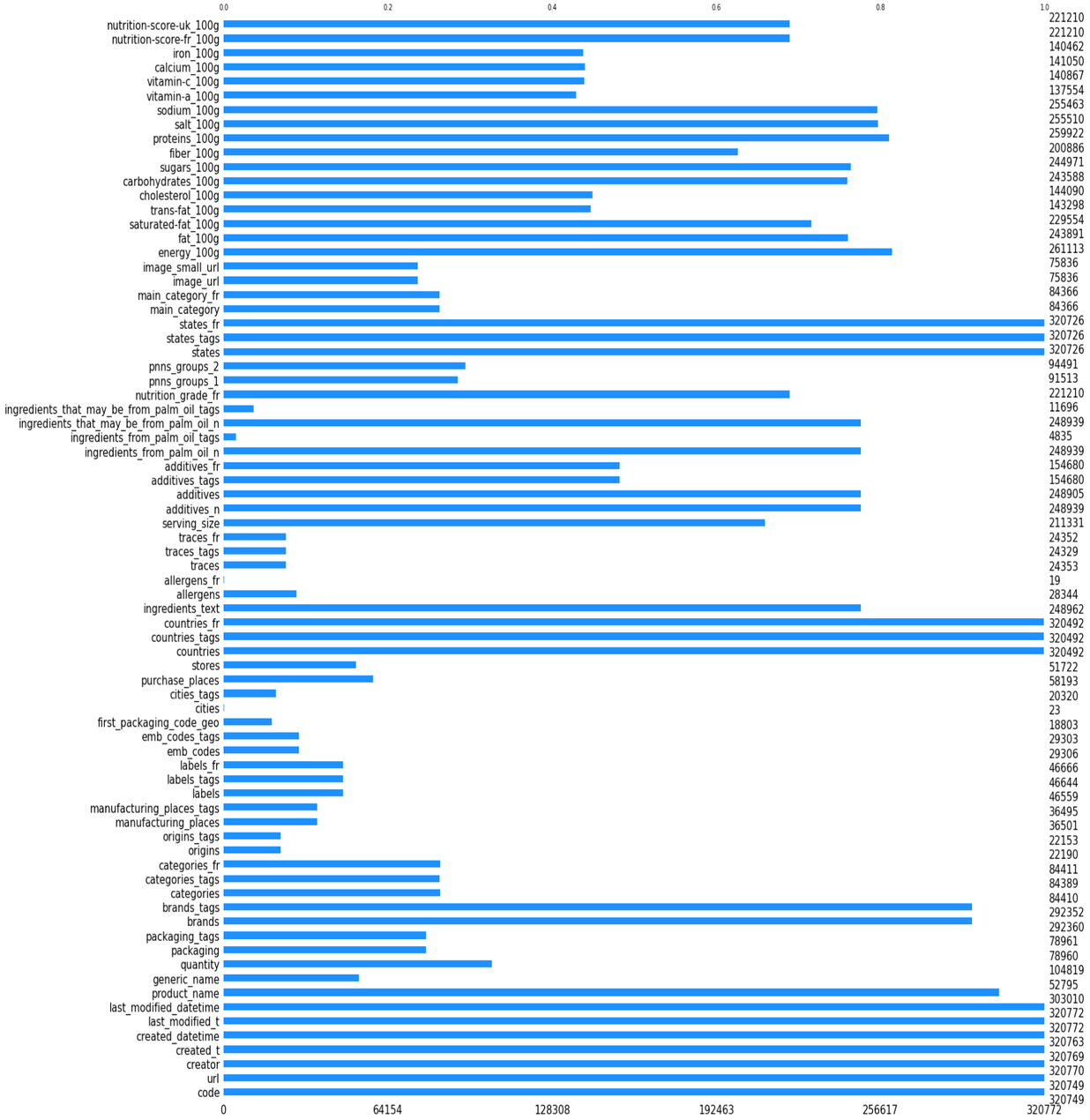
Distinction entre variables

Caractéristiques assez différentes

Action prise

Supprimer les variables avec plus de 80% de données manquantes





Après traitement…

Variables peu renseignées supprimées…

- Majorité des variables avec peu d’informations supprimées
- Premier nettoyage réussi

… mais des exceptions sont toujours présentes

- Données textuelles préservées
- Indispensable pour identifier des particularités

Doublons

- 320 000 lignes mais 221 000 produits ?
- 99 424 doublons
- Garder le produit le plus récent

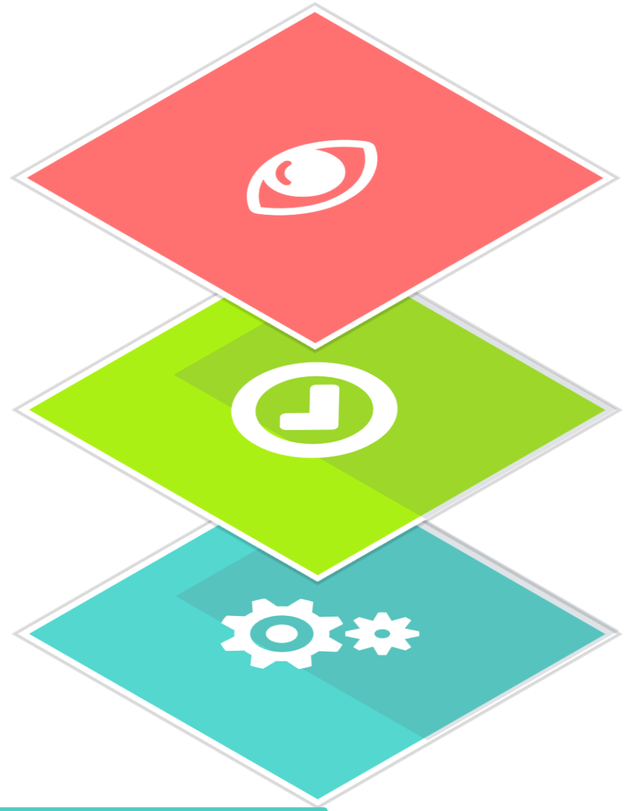
```
df_data_clean = df_data_clean.sort_values('created_t').drop_duplicates("product_name", keep = "last")
```

Outliers

Détection d'outliers

```
: count    243891.000000
  mean         12.730379
  std          17.578747
  min           0.000000
  25%           0.000000
  50%           5.000000
  75%          20.000000
  max         714.290000
  Name: fat_100g, dtype: float64
```

```
count    259922.000000
mean          7.075940
std           8.409054
min        -800.000000
25%           0.700000
50%           4.760000
75%          10.000000
max         430.000000
Name: proteins_100g, dtype: float64
```

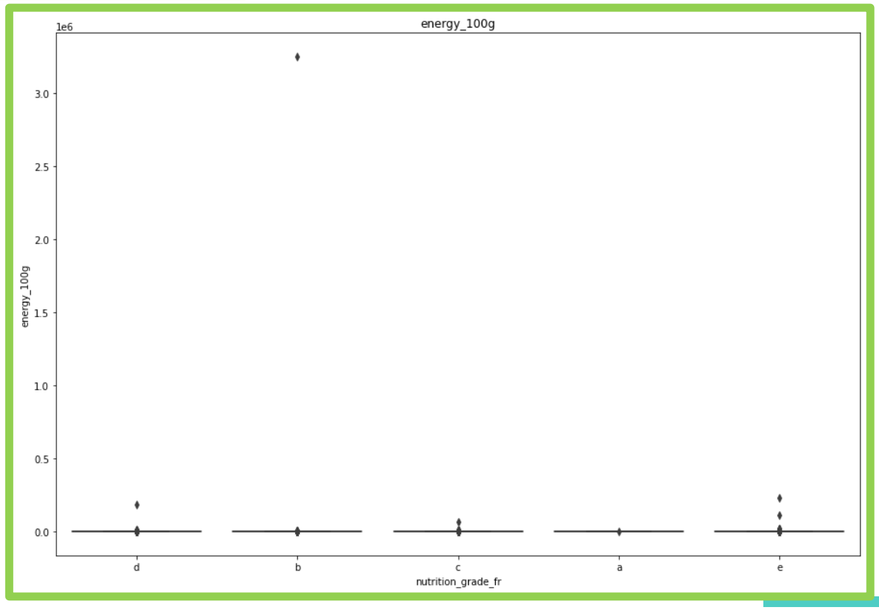


Solution

Supprimer les valeurs negatives et supérieures à 100

```
def delete_outliers_nutrition(data, columns):
# Les colonnes nutrition sont pour 100g. Il ne peut donc, ni y avoir valeur négative, ni valeur au-dessus de 100g
# where supprime les valeurs où la condition est fausse
    for col in columns:
        data.loc[:,col] = data[col].where(data[col] >= 0)
        data.loc[:,col] = data[col].where(data[col] <= 100)
    return(data)
```

Confirmation des outliers



Variables

01 **Date**
Création / Modification

02 **Le produit**
Nom / Marque / Catégorie

03 **Origine du produit**
Pays

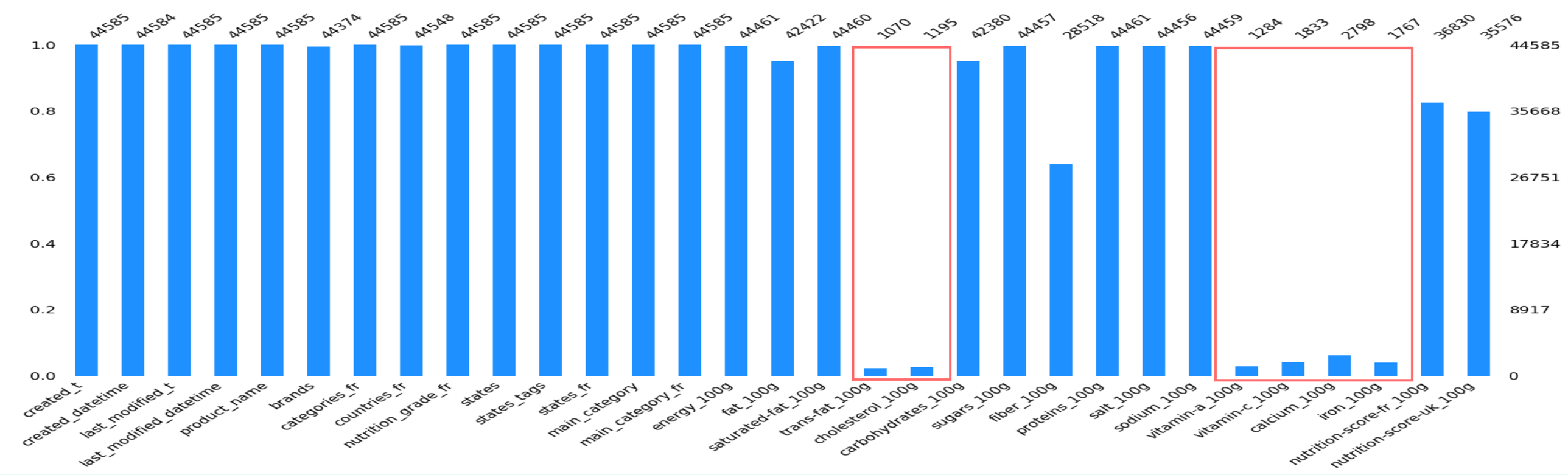
04 **Valeurs nutritionnelles**
Matières graisses, protéines, sucres...

	created_t	created_datetime	last_modified_t	last_modified_datetime	product_name	brands	categories_fr	
187478	0	NaN	1488992055	2017-03-08T16:54:15Z	Lulu la barquette (Fraise)	LU	Snacks sucrés,Biscuits et gâteaux,Biscuits,Bar...	
251761	1328021038	2012-01-31T14:43:58Z	1482511099	2016-12-23T16:38:19Z	Caramels tendres au beurre salé au sel de Guér...	Carabreizh	Epicerie,Snacks sucrés,Confiseries,Caramels	
188901	1328783696	2012-02-09T10:34:56Z	1482511099	2016-12-23T16:38:19Z	Jacquet Les bouchées créatives à garnir	Jacquet	Snacks salés,Apéritif,Biscuits apéritifs	
195704	1328986318	2012-02-11T18:51:58Z	1403887806	2014-06-27T16:50:06Z	Cookies tout chocolat Biocoop	Biocoop	Snacks sucrés,Biscuits et gâteaux,Biscuits,Bis...	
302186	1328993181	2012-02-11T20:46:21Z	1490646628	2017-03-27T20:30:28Z	Eau minérale gazeuse	San Pellegrino	Boissons,Boissons gazeuses,Eaux,Eaux minérales...	Belgique,
...	...	...	...	...	...	...	...	
211769	1492737303	2017-04-21T01:15:03Z	1492737304	2017-04-21T01:15:04Z	NaN	NaN	NaN	
189404	Brétigny-sur-Orge,Marseille 5°,France	Auchan,Super U	Suisse,France	en:france,en:switzerland	France,Suisse	NaN	6	
189068	France	NaN	Belgique,France	en:belgium,en:france	Belgique,France	NaN	6	er
189109	France	NaN	Belgique,France, en:switzerland	en:belgium,en:france,en:switzerland	Belgique,France,Suisse	NaN	4	er
189379	NaN	NaN	France	en:france	France	NaN	3	er

221345 rows × 31 columns

Erreurs lexicales

- Erreur probablement d'inattention
- Supprimer les produits n'ayant ni catégorie, ni nutriscore



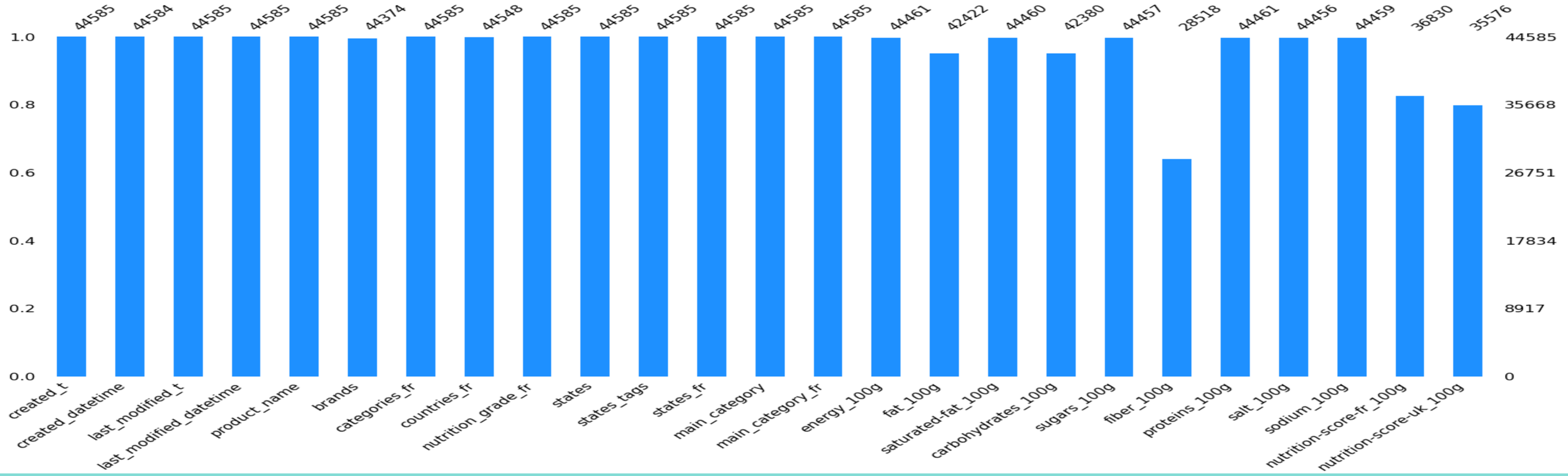
Traitement des dernières variables inutiles

Suppression

Sed at agam dignissim disputationi, nec erroribus maiestatis disputando id.

2

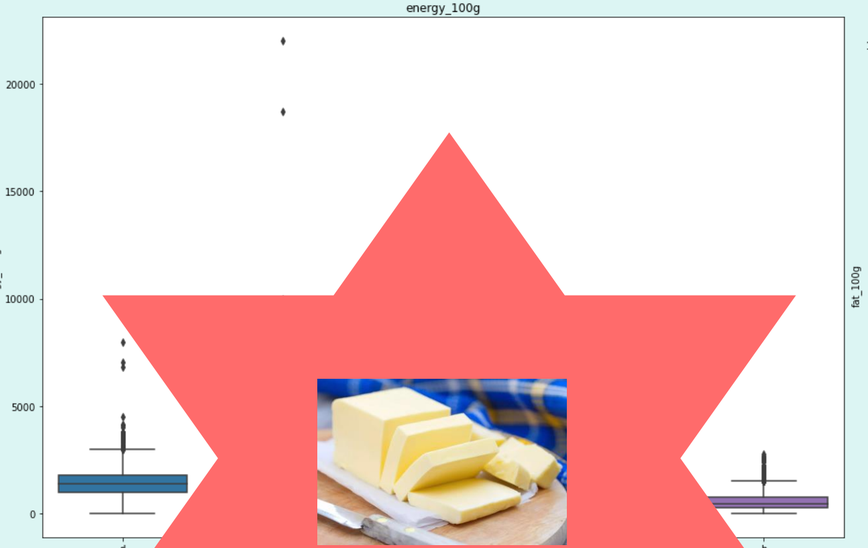
Exploration et Exploitation

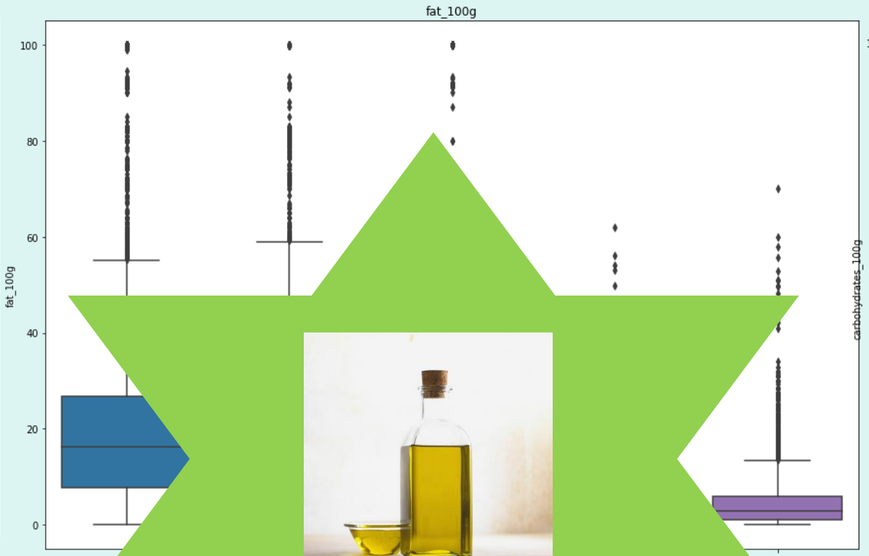


Suppression

Jeu de données

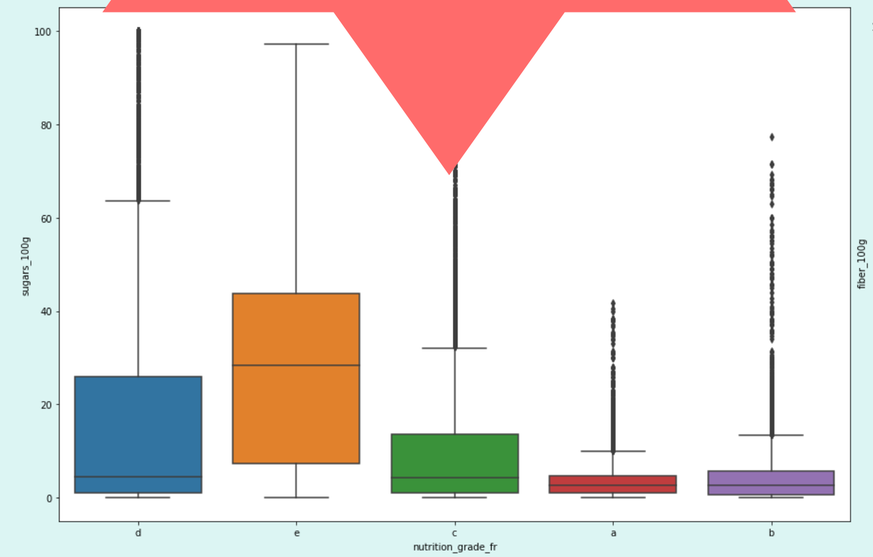
Nutriscore

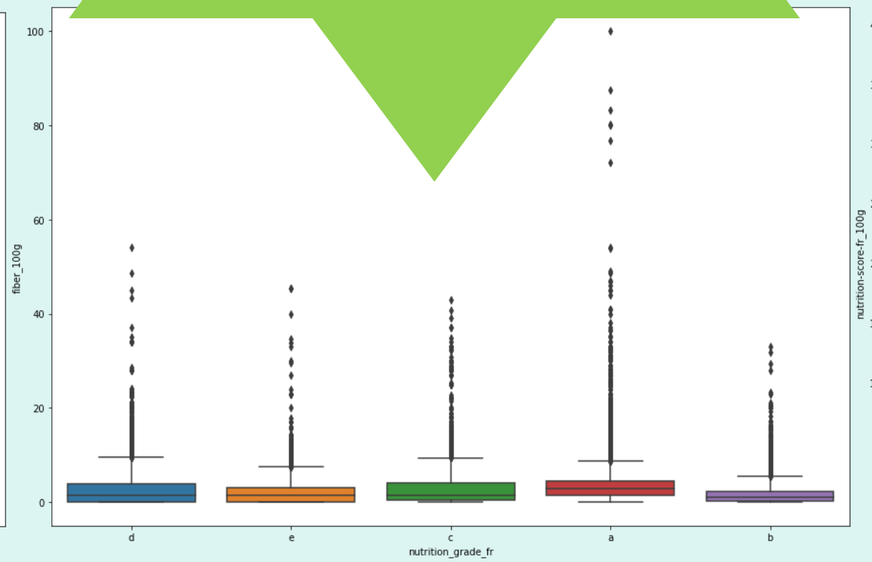

energy_100g
20000
15000
10000
5000
0
d
b

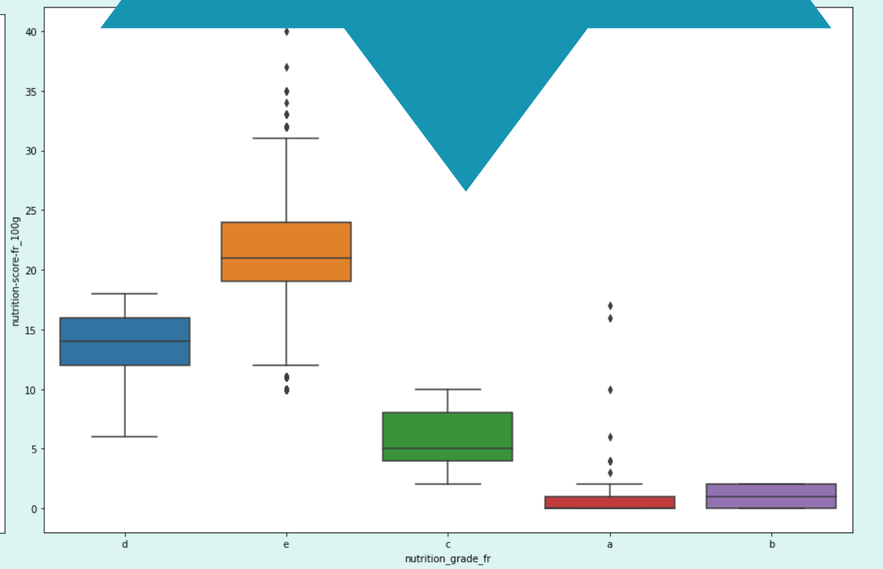

fat_100g
100
80
60
40
20
0
d
b


carbohydrates_100g
100
80
60
40
20
0
d
b

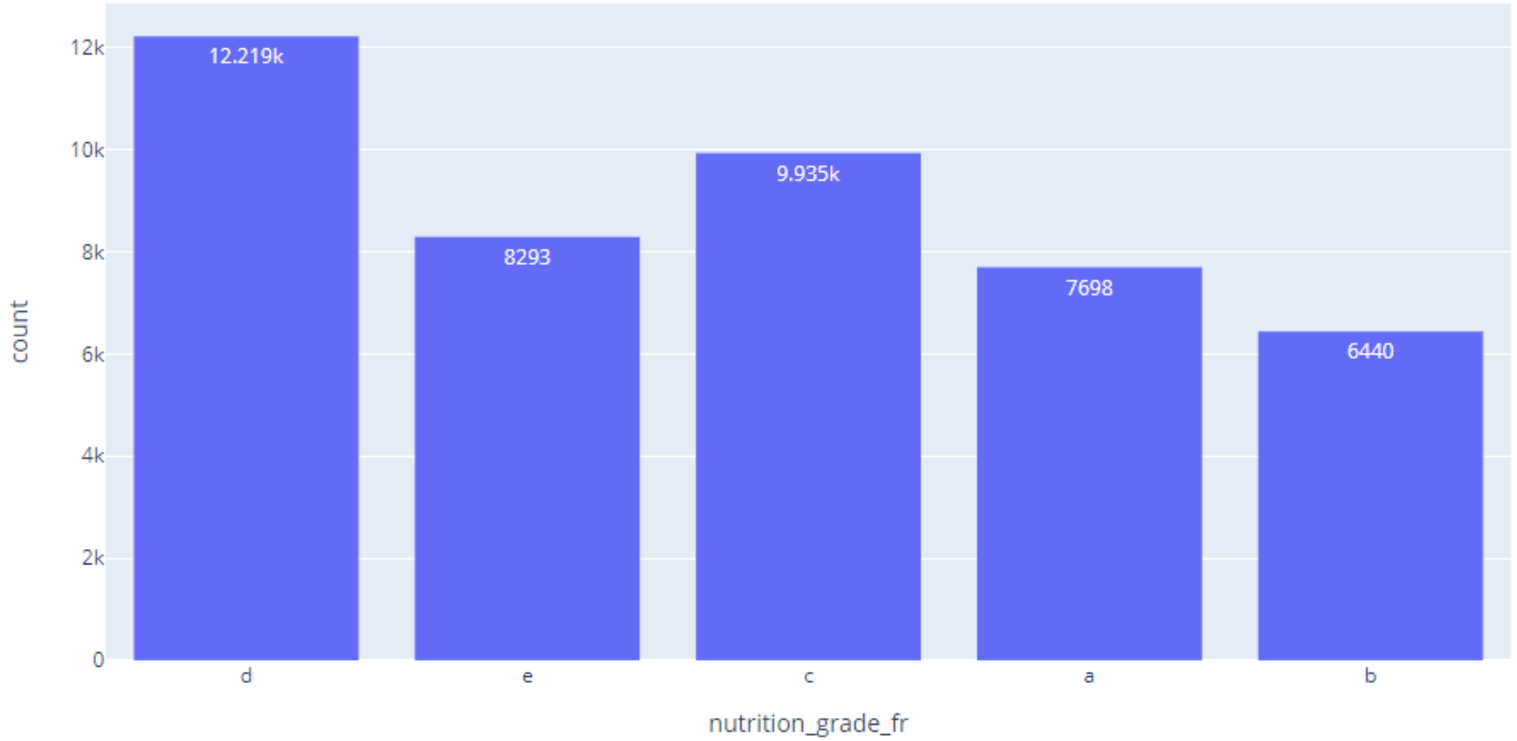
Cadbury
Dairy Milk
Silk
Bubbly


sugars_100g
100
80
60
40
20
0
d
e
c
a
b
nutrition_grade_fr


fiber_100g
100
80
60
40
20
0
d
e
c
a
b
nutrition_grade_fr


nutrition-score-fr_100g
40
35
30
25
20
15
10
5
0
d
e
c
a
b
nutrition_grade_fr

Nutriscore



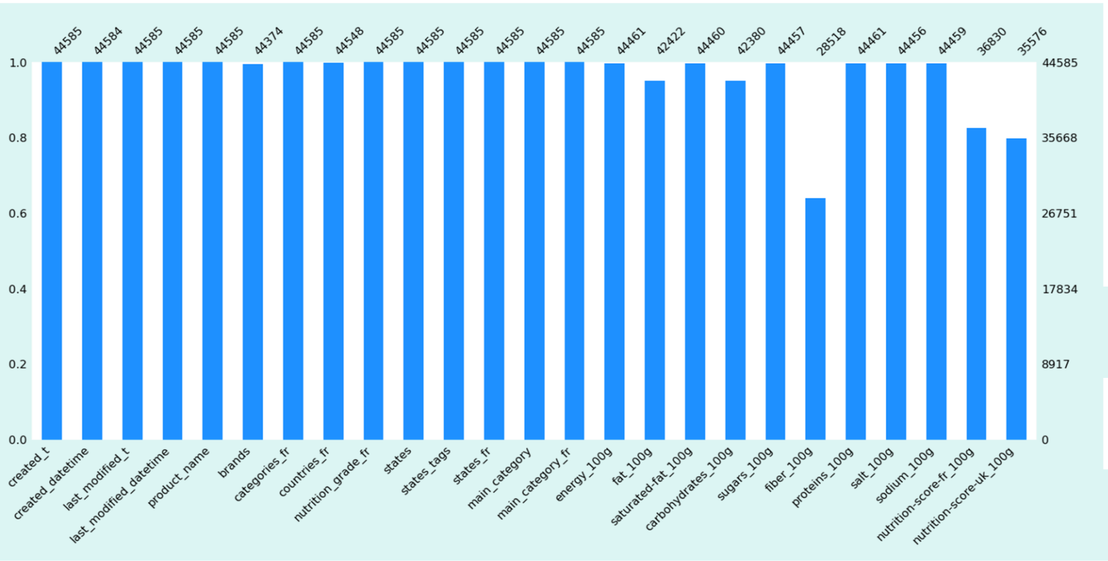
Nombre de produits

44500 produits

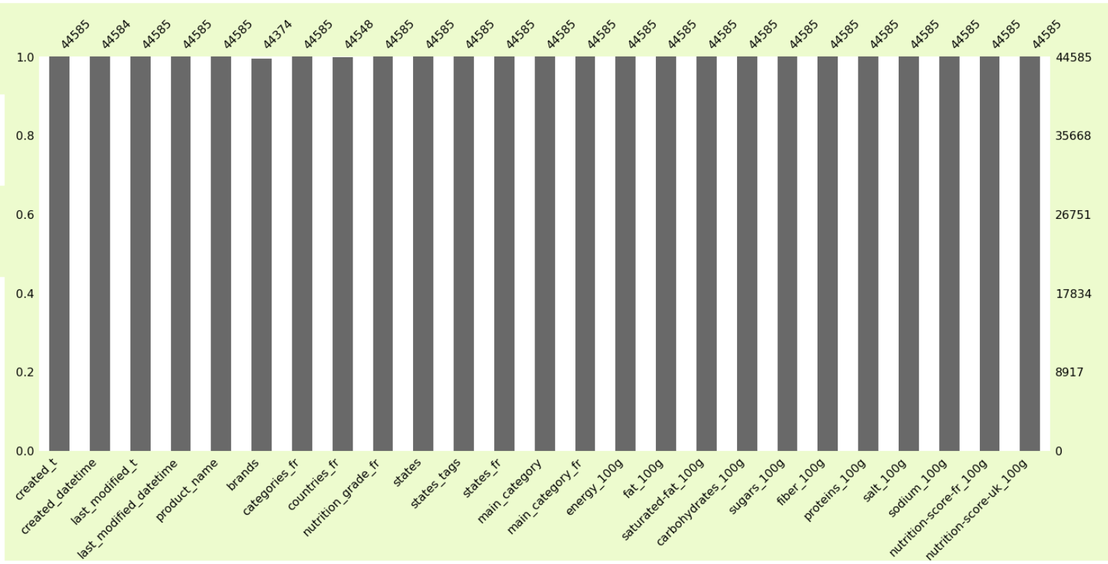
Distribution assez partagée

L'échantillon est assez diversifié

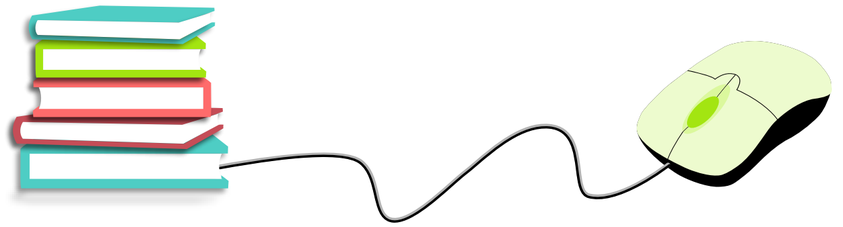
Remplissage par KNN



Avant KNN



Après KNN



ACP

Isoler les variables

```
# selection des colonnes à prendre dans l'ACP
colonnes_nutrition_KNN = data_final_KNN.filter(regex='_100g').
drop(['nutrition-score-fr_100g', 'nutrition-score-uk_100g'],
                                                    axis=1)
```

Centrage et réduction

```
# Centrage et réduction

# La fonction fit_transform()
# renvoie en sortie les coordonnées factorielles Fik que nous collectons dans la variable
std_scale = preprocessing.StandardScaler().fit(colonnes_nutrition_KNN.values)
X_scaled = std_scale.transform(colonnes_nutrition_KNN.values)
```

Vérification…

```
print(X_scaled)
```

executed in 12ms, finished 00:37:35 2022-04-03

```
[[ 0.47893576 -0.6809331  -0.6016112  ... -0.44827713 -0.26334966
  -0.25032246]
 [ 0.78061691 -0.13953767  0.19805013 ... -0.61152308  0.21162574
   0.19761054]
 [ 1.01101475  0.14346449 -0.50319134 ...  0.13668751  0.22704778
   0.21215454]
 ...
 [ 0.77807108  0.05118118 -0.50319134 ... -0.09457758 -0.09622869
  -0.0927164 ]
 [-0.91236168 -0.72399864 -0.56470375 ... -0.61152308 -0.24282604
  -0.23096733]
 [ 0.55531074  0.80790434  1.31757599 ... -0.35305033 -0.19884683
  -0.18949205]]
```

Composantes

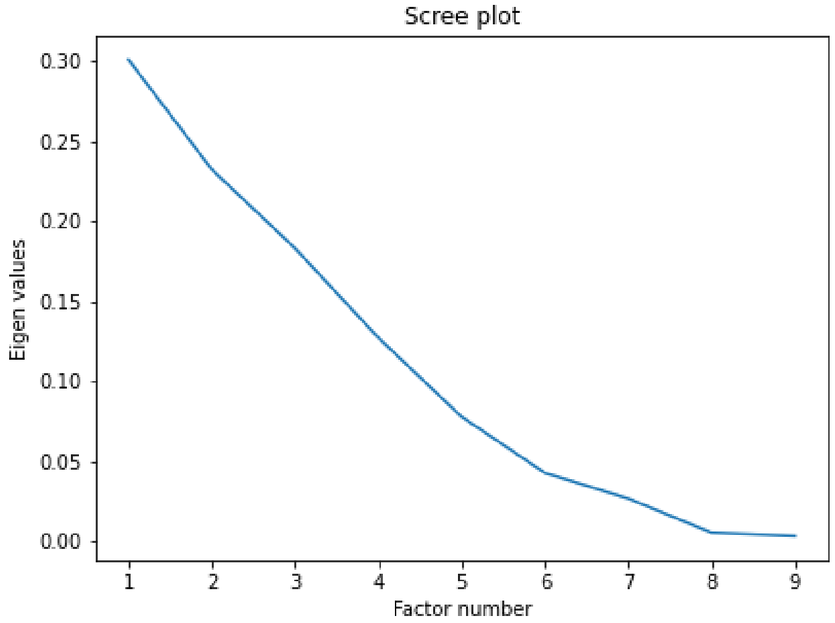
```
acp = PCA(svd_solver='full')

coord = acp.fit_transform(X_scaled)

# nombre de composants à prendre :
n_comp = acp.n_components_
print(n_comp)
```

executed in 10ms, finished 00:37:35 2022-04-03

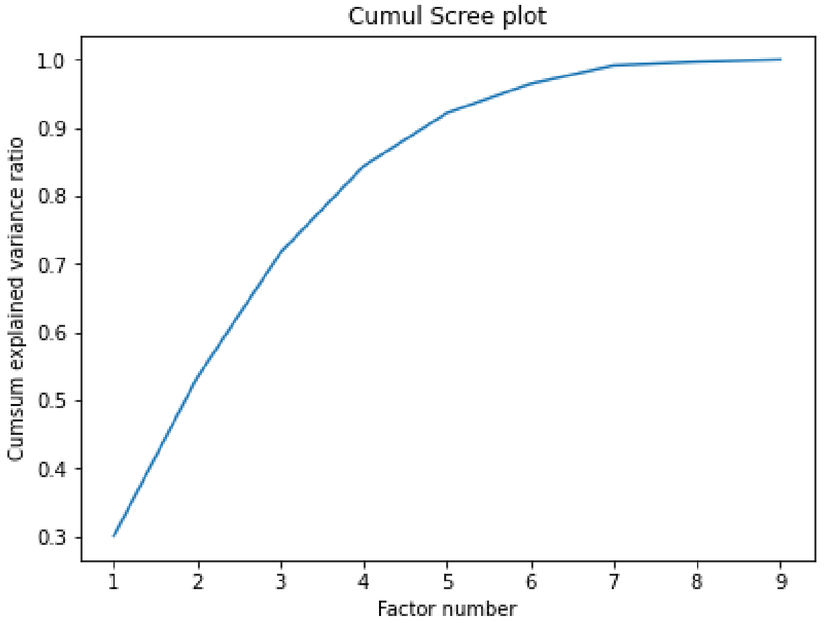
```
9
```

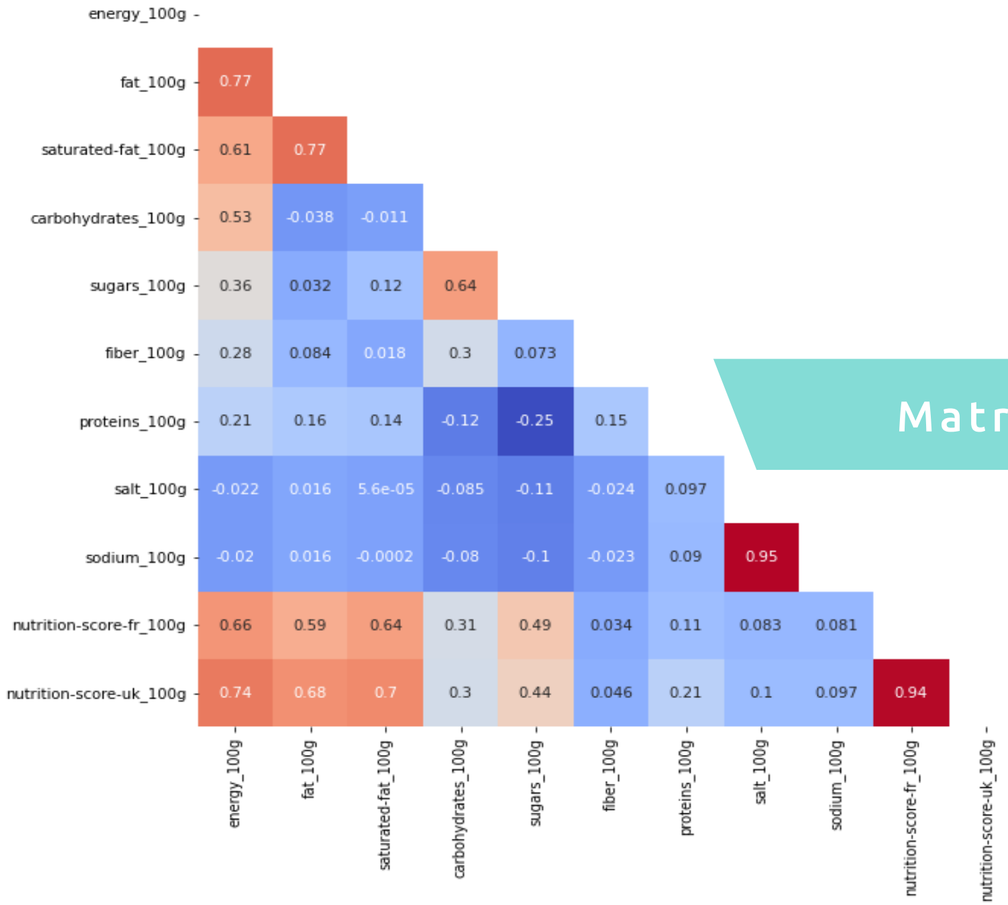
Valeurs propres et scree plot

Information disponible par axe



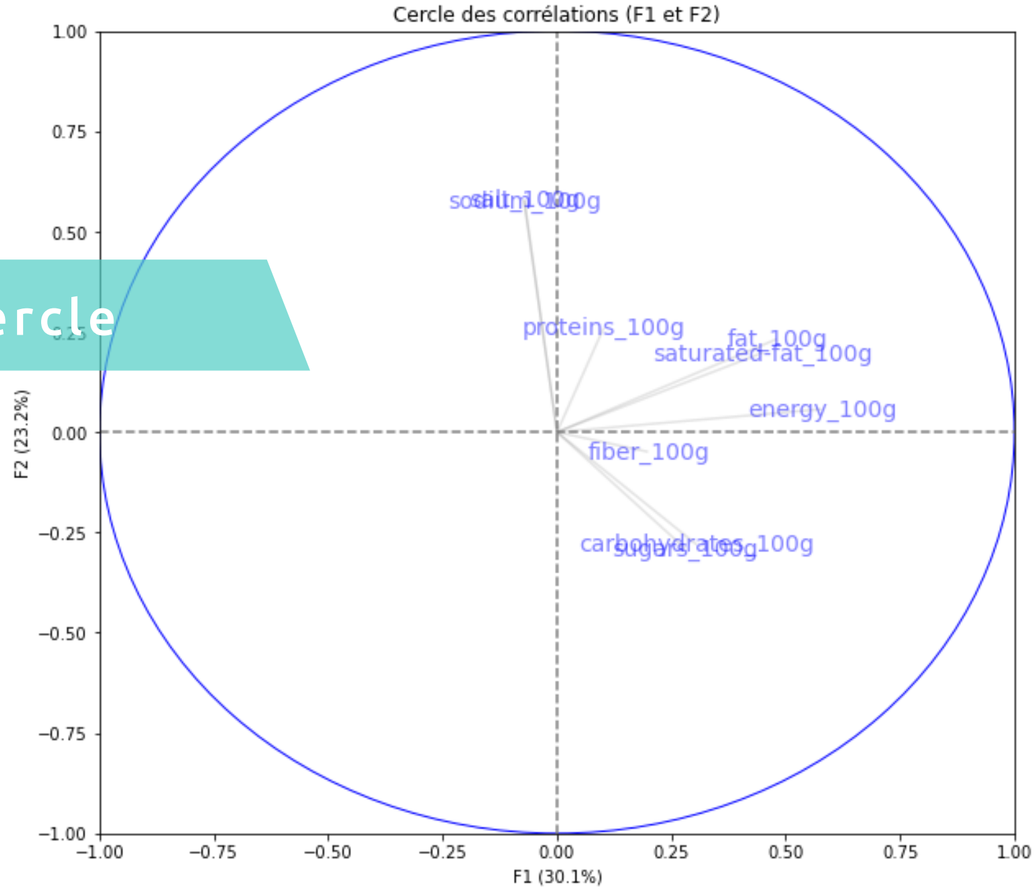
Cumul de l'information disponible





Matrice

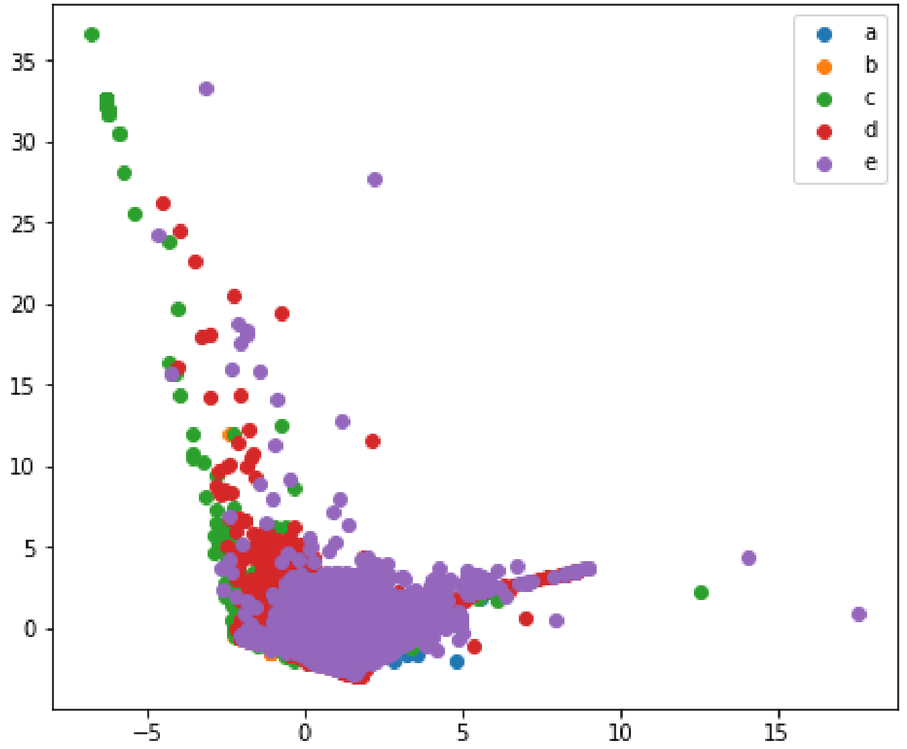
Cercle



Corrélation

Représentation des individus

Plan



Aucune catégorie ne semble se démarquer….

Anova

	index	X	Y	SS	DF	MS	F	p-unc	np2
0	0	nutrition_grade_fr	energy_100g	8339028996.629916	4	2084757249.157479	4846.297435	0.0	0.303058
1	1	Within	energy_100g	19177212997.422577	44580	430175.257905	NaN	NaN	NaN
2	0	nutrition_grade_fr	fat_100g	3278012.525504	4	819503.131376	4297.327127	0.0	0.278282
3	1	Within	fat_100g	8501435.548027	44580	190.700663	NaN	NaN	NaN
4	0	nutrition_grade_fr	saturated-fat_100g	979506.07962	4	244876.519905	5551.862642	0.0	0.332509
5	1	Within	saturated-fat_100g	1966294.190296	44580	44.107093	NaN	NaN	NaN
6	0	nutrition_grade_fr	carbohydrates_100g	1834650.153164	4	458662.538291	667.05949	0.0	0.056473
7	1	Within	carbohydrates_100g	30652702.272671	44580	687.588656	NaN	NaN	NaN
8	0	nutrition_grade_fr	sugars_100g	2864387.409921	4	716096.85248	2689.277265	0.0	0.194392
9	1	Within	sugars_100g	11870697.787218	44580	266.278551	NaN	NaN	NaN
10	0	nutrition_grade_fr	fiber_100g	17488.999265	4	4372.249816	387.001266	0.0	0.033559
11	1	Within	fiber_100g	503654.416693	44580	11.297766	NaN	NaN	NaN
12	0	nutrition_grade_fr	proteins_100g	52931.24179	4	13232.810448	250.36515	0.0	0.021971
13	1	Within	proteins_100g	2356233.248113	44580	52.854043	NaN	NaN	NaN
14	0	nutrition_grade_fr	salt_100g	6574.349074	4	1643.587269	143.085789	0.0	0.012676
15	1	Within	salt_100g	512078.250496	44580	11.486726	NaN	NaN	NaN
16	0	nutrition_grade_fr	sodium_100g	1038.999626	4	259.749907	129.595863	0.0	0.011495
17	1	Within	sodium_100g	89352.009877	44580	2.004307	NaN	NaN	NaN
18	0	nutrition_grade_fr	nutrition-score-fr_100g	2223840.99835	4	555960.249587	64395.620465	0.0	0.852463
19	1	Within	nutrition-score-fr_100g	384881.887117	44580	8.63351	NaN	NaN	NaN
20	0	nutrition_grade_fr	nutrition-score-uk_100g	2033642.130501	4	508410.532625	25081.72297	0.0	0.692354
21	1	Within	nutrition-score-uk_100g	903643.723827	44580	20.27016	NaN	NaN	NaN

Anova

Source = Factor names

SS = Sums of Squares

DF = Degrees of freedom

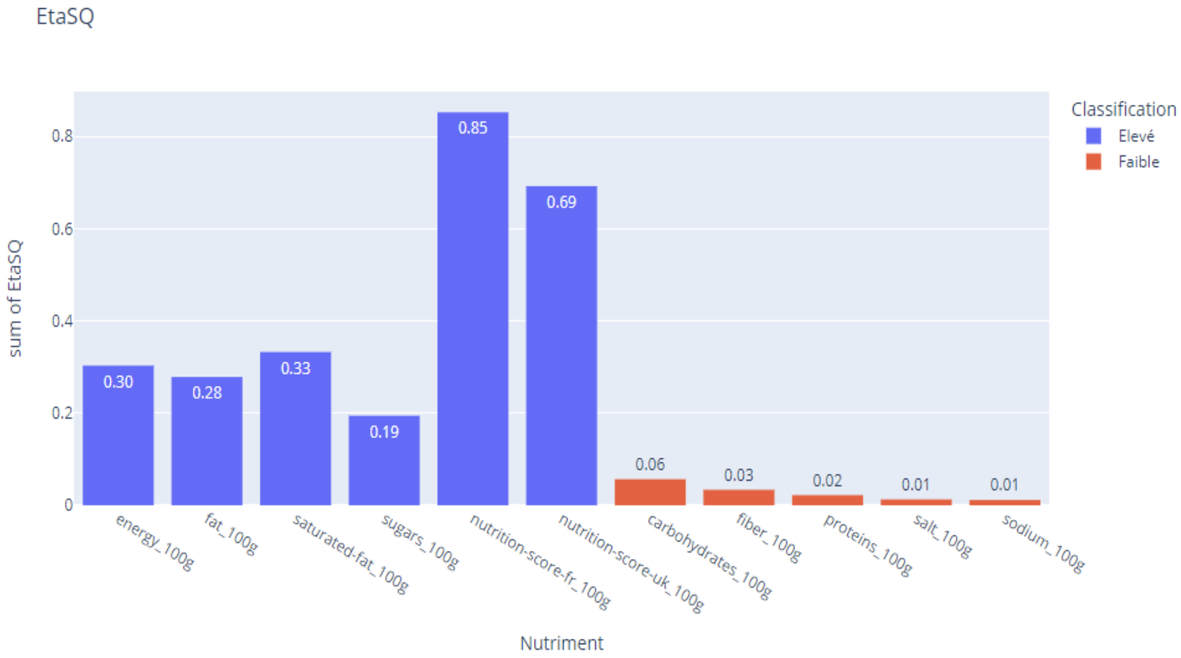
MS = Mean Squares

F = F-values

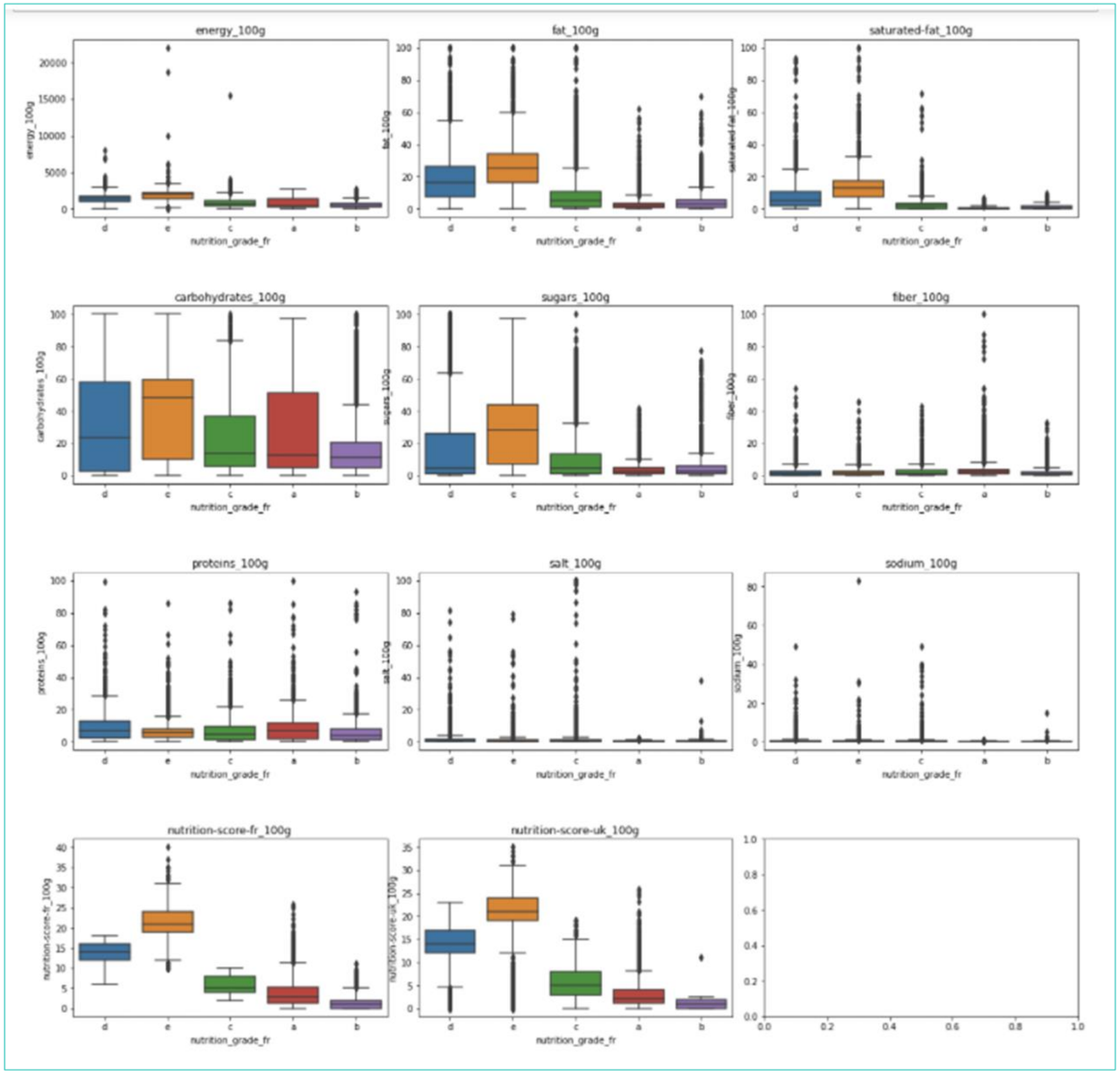
p-unc : uncorrected p-values

np2 : Partial eta-square effect sizes

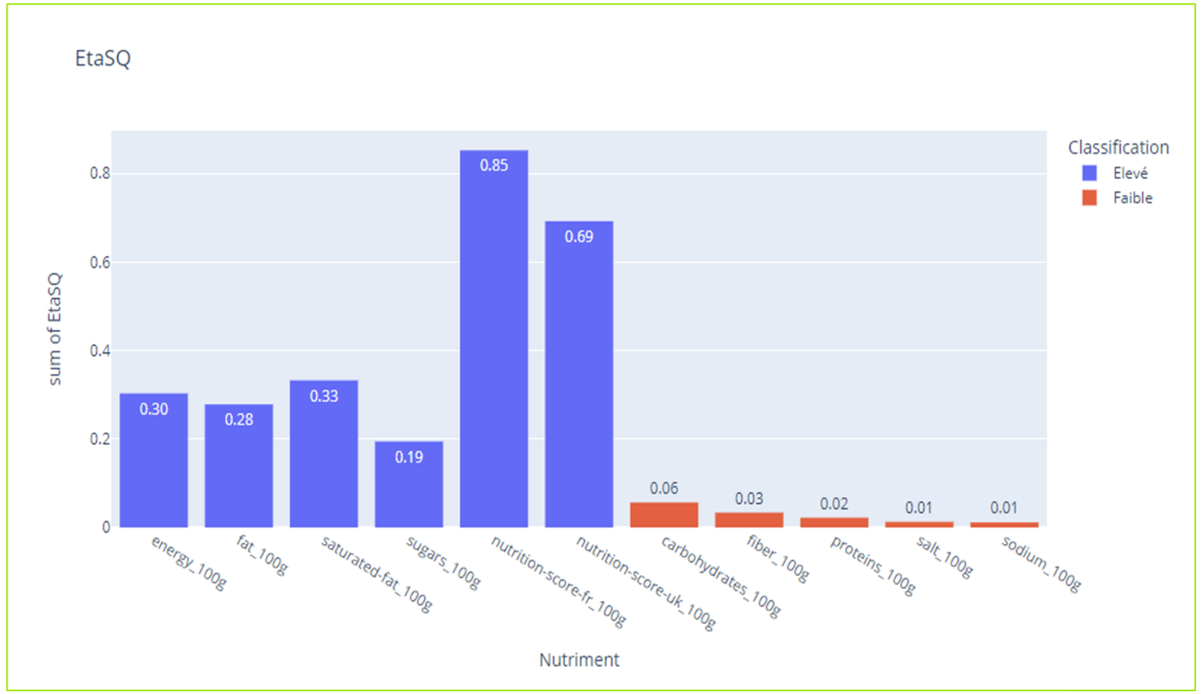
Eta_square



Corrélation

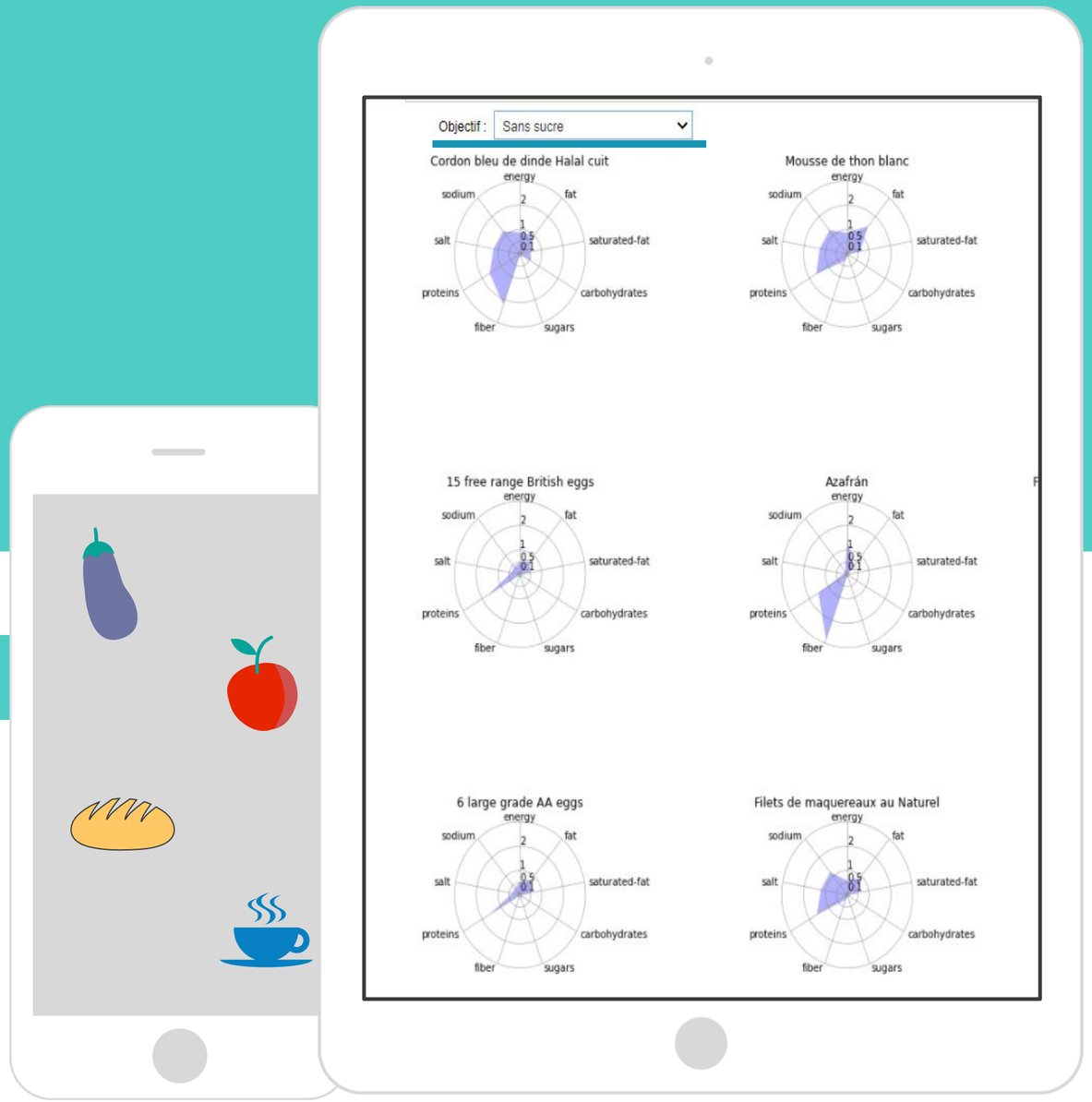


Comparaison



3

Résultats



Application

- Objectif : Repas sans sucre