CentraleSupélec

OPENCLASSROOMS

# Analysez des données nutritionnelles

## Contenu




MINÉ Nicolas

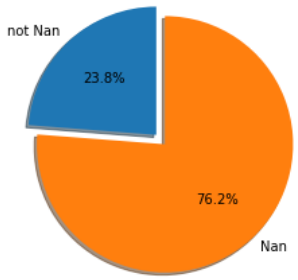
# Introduction

Le site "Lamarmite" souhaite mettre en place un générateur de recettes saines. Pour ce faire, elle a besoin d'une analyse de données pour comprendre les paramètres permettant d'obtenir un plat sain ou non. Pour ce faire, elle nous met à disposition une base de données recensant divers produits de consommation avec différentes informations nutritionnelles.

Ce rapport va donc présenter le nettoyage des données avec la sélection des features, leur simplification si nécessaire puis un nettoyage des données aberrantes. Par la suite une analyse univariée et multivariée va être effectuée pour mettre en relief les paramètres important à une recette équilibrée nutritionnellement.

# Nettoyage

Les données originales sont issues d'Openfoodfacts.org. Le dataset est assez complexe à traiter car il est constitué de 321 000 produit regroupant 162 paramètres (allant du nom à sa composition par 100g en passant par le pays de production ou encore le CO2 généré lors de sa fabrication)




Le gros problème de ce dataset est le nombre de données manquantes. En effet, près de 75% du dataset ne possède pas de données.

L'objectif de ce nettoyage sera donc de supprimer les features inutiles et inexploitable tout en gardant un nombre de points le plus important possible. Pour ce faire, le nettoyage a été fait en 3 phases :

## Sélection des features :

Lors de cette phase, l'ensemble des colonnes a été exploré pour visualiser le type de données, les valeurs et leur fréquence. Lorsque celle-ci n'était pas utile à l'étude, elles ont été supprimées. Les critères sont :

- Données temporelles :
  - la date d'ajout ou de mise à jour n'est pas utile
- Données de référence:
  - Nom du produit
  - Origine
  - Magasin les vendant

Ce 1er nettoyage a permis de réduire à environ 80 features le dataset.

## Simplification de features :

Pour les features de type "liste", les données ont été agrégées par le compte au lieu de chaque élément. C'est le cas pour les labels par exemple. Générer 1 colonne par label aurait fait exploser le nombre de features sans pour autant permettre une meilleur interprétation. De ce fait, cette colonne a été remplacée par le nombre de labels. Certaines features avaient déjà ce paramètre (par exemple allergens_n)

Pour finir la simplification, certaines features n'avaient besoin que de nettoyage. C'est le cas des pnns_groups qui avaient des orthographes différents et créant des surplus de valeurs pour rien. L'objectif était donc d'agréger les données.

## Nettoyage des données aberrantes :

En parallèle à l'exploration, une suppression des données aberrantes a été faite. Cela s'est fait en 3 critères pour les données lié à la constitution du produit.

### Critère 1:Suppression des valeurs négatives.

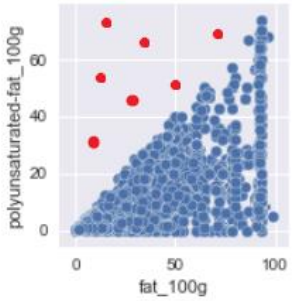
De manière évidente, il n'est pas possible d'avoir une masse négative d'un constituant.

### Critère 2: Suppression des outliers

Pour cette phase, l'étude univariée a été très utile. En effet, certains produits possédaient des quantités aberrantes d'un constituant (par exemple le sel contenait 100g de sel par 100g). La majorité des produits étaient bien en dessous. Pour cette simplification j'ai juste gardé les produits dont leur valeurs est inclus dans un interval de +/- 5 fois la Standard Déviation autour de la médiane.

### Critère 3: Suppression des valeurs impossibles

Ce critère de nettoyage a été trouvé lors de l'étude des différents types de graisses. En effet lorsque l'on affiche la quantité de certaines graisses en fonction de la graisse totale, on pouvoir voir des points ou la graisse spécifique était supérieurs à la graisse normale (cf. les points rouge sur l'image ci-contre)




Pour terminer le nettoyage, toutes les feature ayant moins de 1000 données ont été supprimé. Cela a permis de réduire de 162 à 44 features et conserver 77 % des produits. Au niveau des données manquantes, la réduction de donnée manquante est de 19 % pour atteindre 56% de Nan. Une fois les données nettoyé, un dataset a été sauvegardé pour passer à la phase d'exploration.

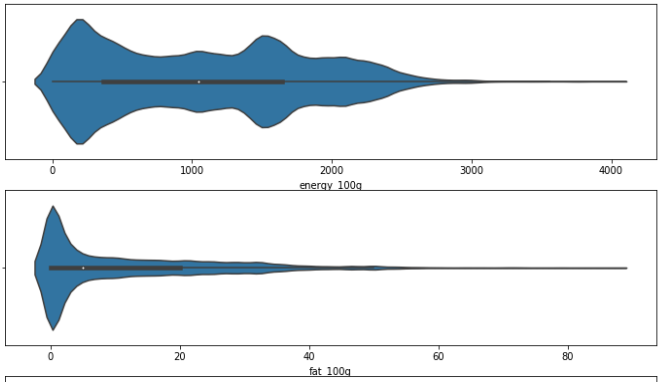
# Exploration

## Analyse Univariée

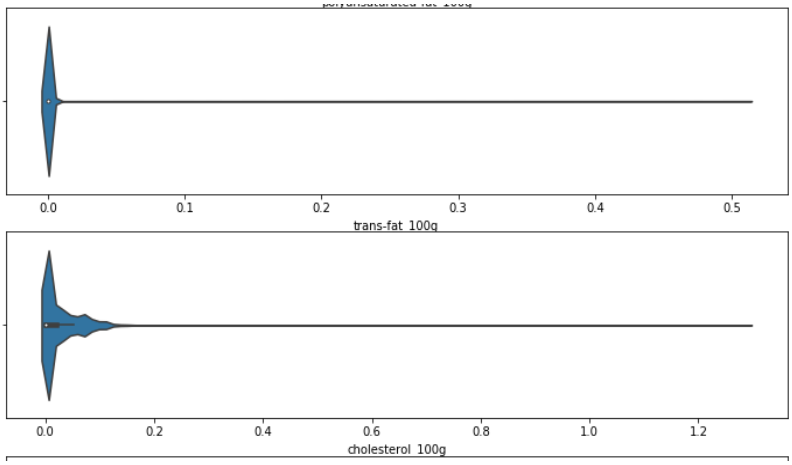
Pour l'analyse univariée, 3 principaux diagrammes ont été utilisé pour visualiser les features :

### Données Continues : Violinplot

Pour les données continues (plus globalement les features finissant par "_100g"), l'utilisation du violinplot est le plus censé. Il permet d pouvoir visualiser la répartition des valeurs et ainsi de peaufiner les règles de nettoyage (cf. le critère 2 et 3 de nettoyage). A chaque fois, les règles ont été ajoutées au Notebook de Nettoyage pour nettoyer le dataset.

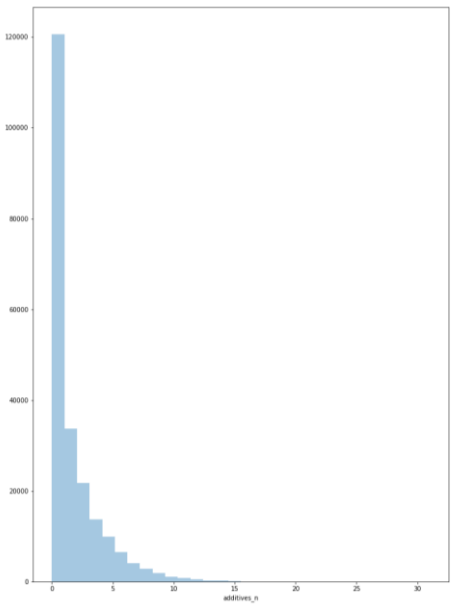



Malgré cette règle pour les outliers, il reste des violinplot peu balancé (cf. l'illustration ci-dessous avec la répartition des gras dits "trans" et le cholestérol). La raison de cette balance sera expliquée dans la partie parlant de l'intérêt de cette étude dite univariée.



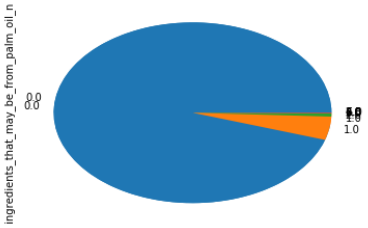

## Données discrètes : Histogramme

Pour les données discrètes, un diagramme de distribution a été utilisé. L'objectif étant de visualiser aussi la présence de chaque paramètre de features dans le dataset.



## Données quasi binaires : Tarte

Pour les données binaires ou quasi binaires (très peu de classes), le diagramme en tarte a été utilisé pour visualiser aussi leur répartition sur l'ensemble du dataset. C'est le cas notamment pour les ingrédients venant des huiles de palmes. On peut remarquer que la très grande majorité des produits n'ont pas d'huile de palme




## Intérêt de cette étude

L'intérêt de cette étude a été majoritairement le nettoyage des outliers comme expliqué précédemment. En effet, si on regarde par ingrédient la distribution. On a par exemple pour le sel :

| | index | count | mean | std | min | 25% | 50% | 75% | max | cleanup |
|---|---|---|---|---|---|---|---|---|---|---|
| **10** | salt_100g | 255510.0 | 2.028624 | 128.269454 | 0.0 | 0.0635 | 0.58166 | 1.37414 | 64312.8000 | avant |
| **44** | salt_100g | 251684.0 | 1.655391 | 7.142057 | 0.0 | 0.0635 | 0.58928 | 1.36906 | 604.7613 | apres |
| **78** | salt_100g | 243786.0 | 1.122654 | 2.404598 | 0.0 | 0.0635 | 0.57404 | 1.34366 | 37.4650 | apres2 |

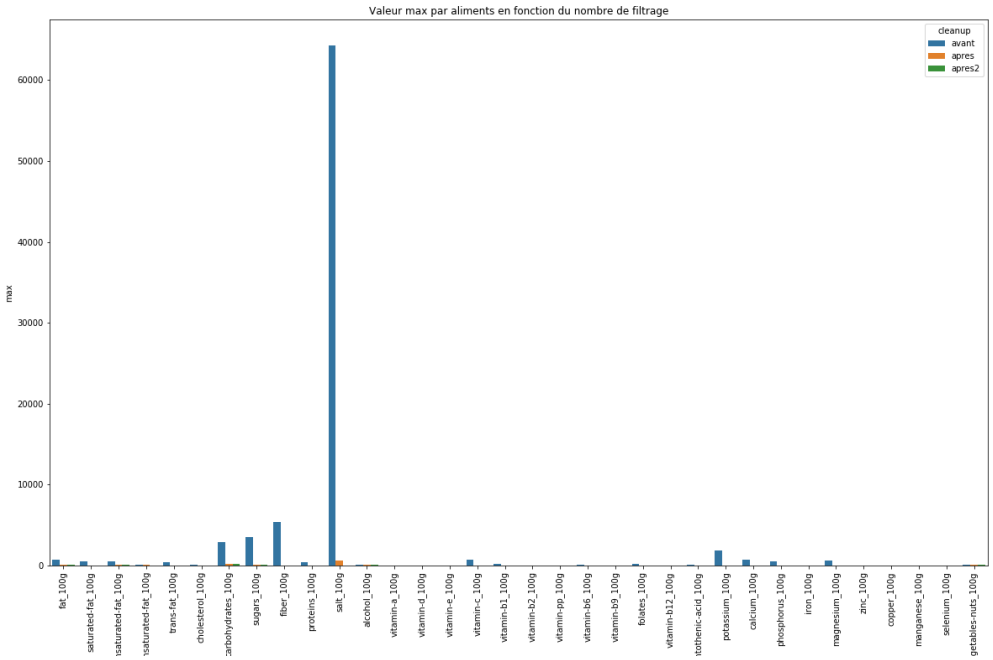
La 1ere ligne est avant un premier filtrage avec +/- 5 fois la Standard Déviation. La seconde ligne est après le filtrage et la 3ème est après un second filtrage identique.

On remarque qu'en l'absence de ce filtrage on a en max 64 312 g de sel par 100g. Cela est irréaliste. C'est tellement grand que l'impact sur la moyenne et la Standard déviation qu'après le 1$^{er}$ filtrage, on est toujours à 604g de sel par 100g. Il faut appliquer une seconde fois ce filtre pour avoir une valeur raisonnable (37 g tout de même). Ce filtrage a notamment supprimé 12 000 points aberrants.

Par contre du côté des fruits, on a une variation très faible et de ce fait on ne supprime que 37 points.

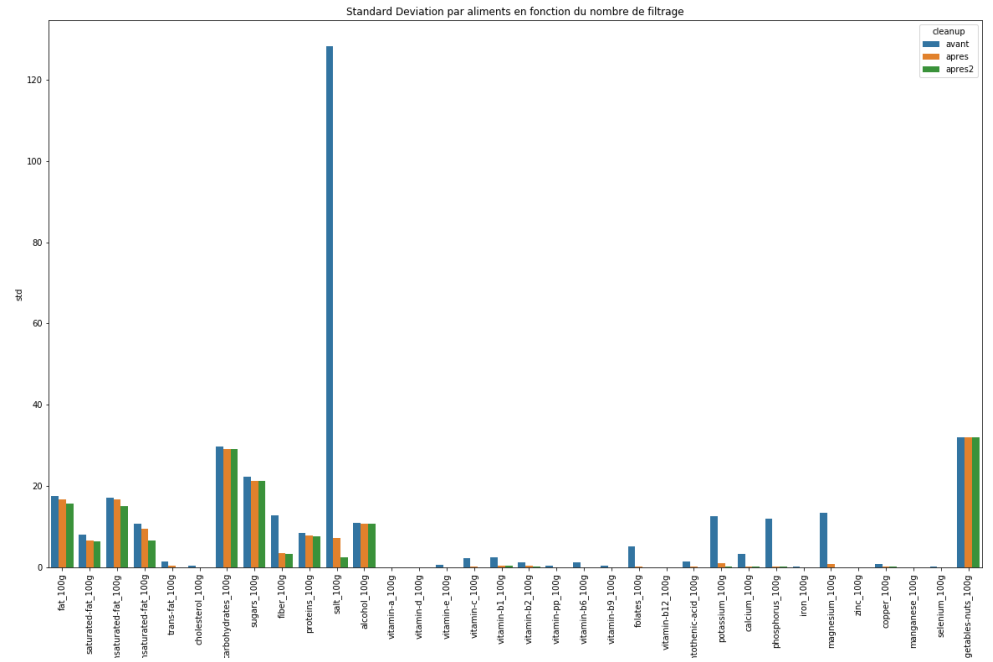
| | index | count | mean | std | min | 25% | 50% | 75% | max | cleanup |
|---|---|---|---|---|---|---|---|---|---|---|
| **33** | fruits-vege... | 3036.0 | 31.458587 | 31.967918 | 0.0 | 0.0 | 23.0 | 51.0 | 100.0 | avant |
| **67** | fruits-vege... | 3015.0 | 31.594783 | 31.935997 | 0.0 | 0.0 | 23.6 | 51.0 | 100.0 | apres |
| **101** | fruits-vege... | 2999.0 | 31.650640 | 31.913197 | 0.0 | 0.0 | 24.0 | 51.0 | 100.0 | apres2 |

Si on regarde l'évolution du maximum par filtrage et par produits, on a :

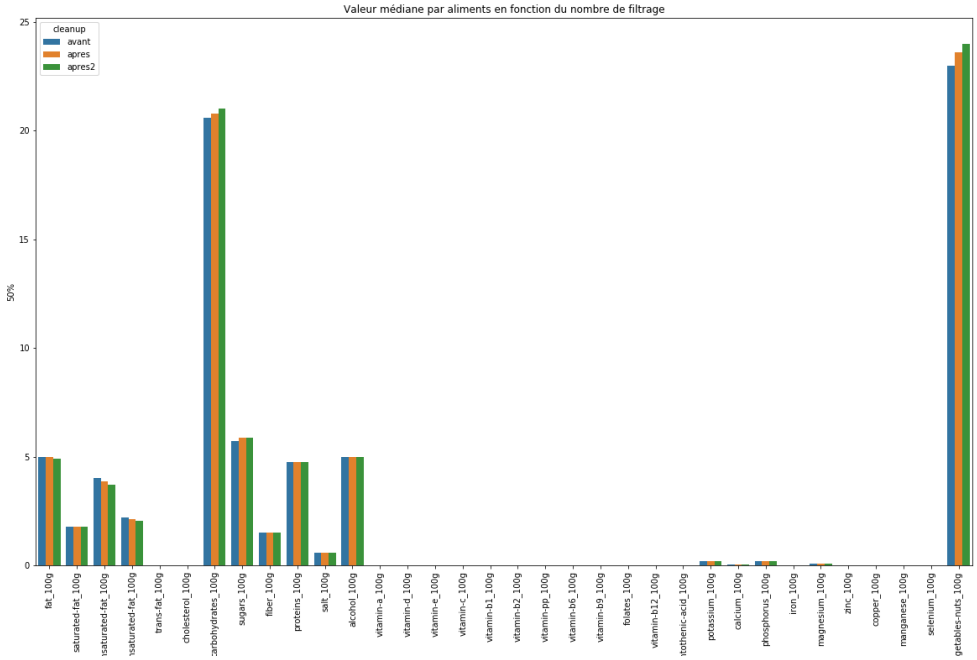



A cause de l'échelle, on a du mal à voir les valeurs post filtrage mais on remarque que celui-ci est utile pour ce nettoyage.

Si on regarde la Standard Déviation,



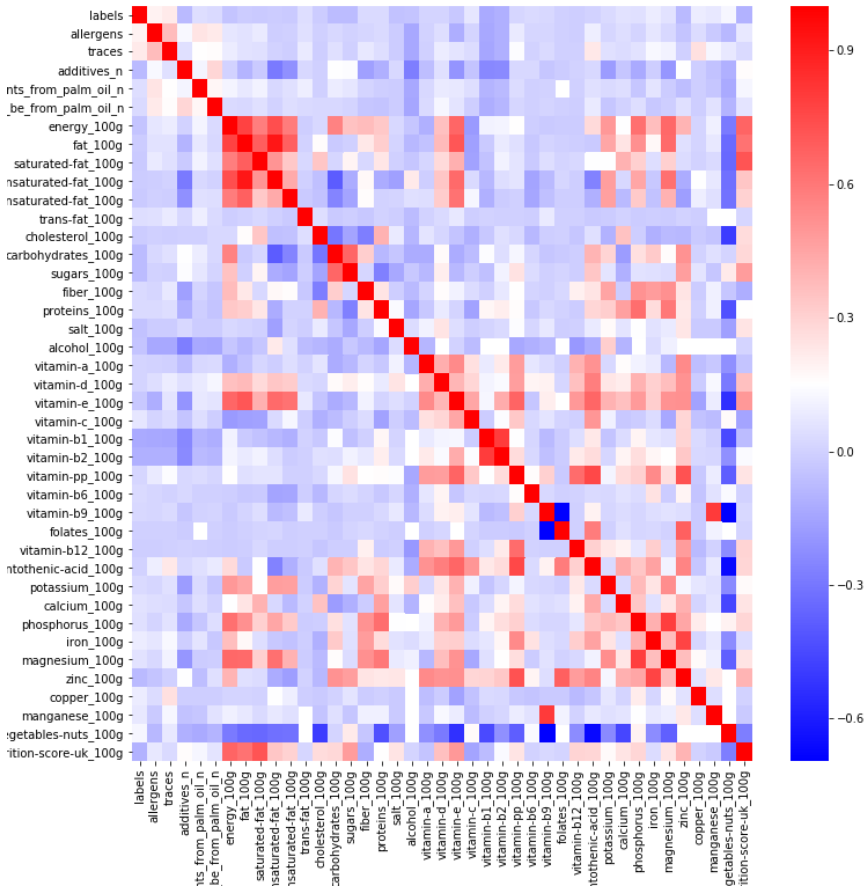

on remarque aussi une chute importante après le 1$^{er}$ filtrage pour les features ayant un fort outlier. S'il n'y en a pas, la valeur reste stable. Et finalement, si on regarde du côté de la médiane, elle reste très similaire car on ne supprime que peu de points




Cela est un grand intérêt de ce type de filtrage. Cependant, ayant filtré précédemment les valeurs négatives et supérieurs à 100g, je n'ai appliqué qu'un seul filtrage.

## Analyse Multivariée

L'analyse multivariée a été faite en 2 parties. La première était basée sur la matrice de corrélation suivante :



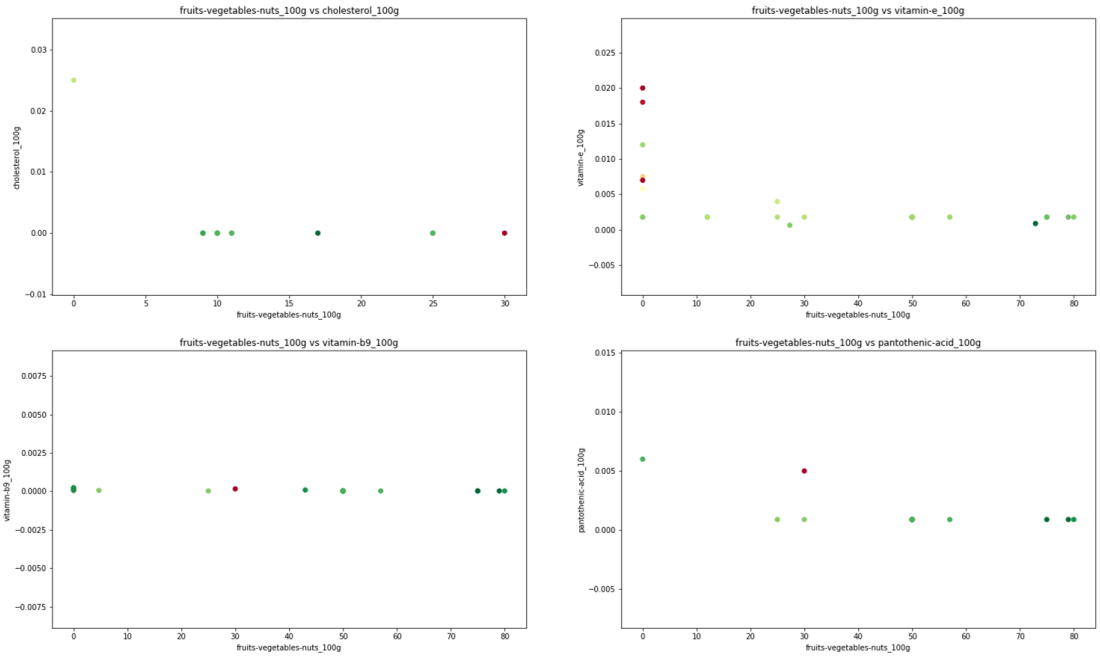
Sur celle-ci on notera pour la suite 6 corrélation/indépendance:

### Indépendance forte : Vitamine b9 & folates

Cette indépendance forte s'avère être due à un manque de points. En effet, le dataset n'a que 3 données possédant une valeur pour les vitamines b9 ET les folates. De ce fait, on ne peut pas dire que les 2 variables sont indépendantes mais on ne peut pas dire si elles sont corrélées

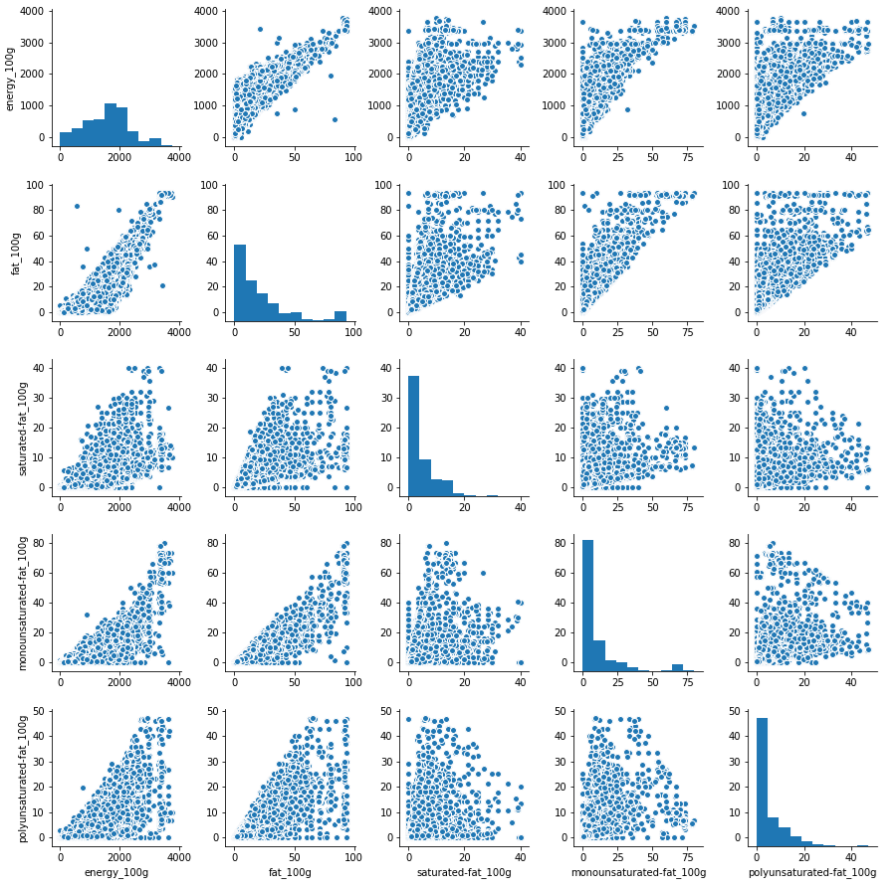
### Indépendance forte : Fruit, Légume, nuts & les autres paramètres

On remarque que la ligne des fruits et légumes par 100g est indépendant avec la majorité des valeurs. Si on trace pour les principales indépendances un la répartition des valeurs on remarque que comme pour les folates et la vitamine b9, on manque surtout de points.



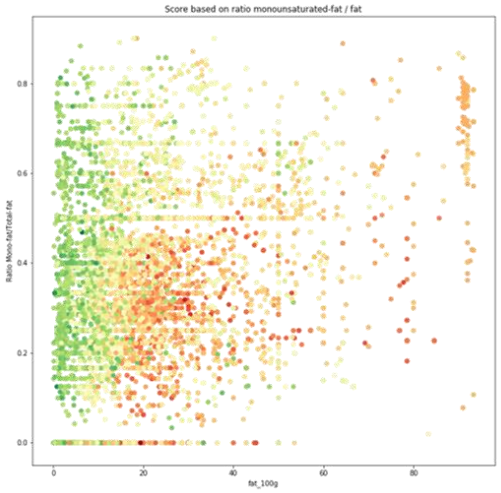

### Corrélation forte : Graisses

Une des plus grandes corrélations des donnés se situe entre les différents type de graisses, mais aussi avec l'énergie. Cela est logique car la majorité de l'énergie vient en très grande partie de la brûlure des graisses. On peut donc tracer les "pairplot" entre chacun de ces 5 features.



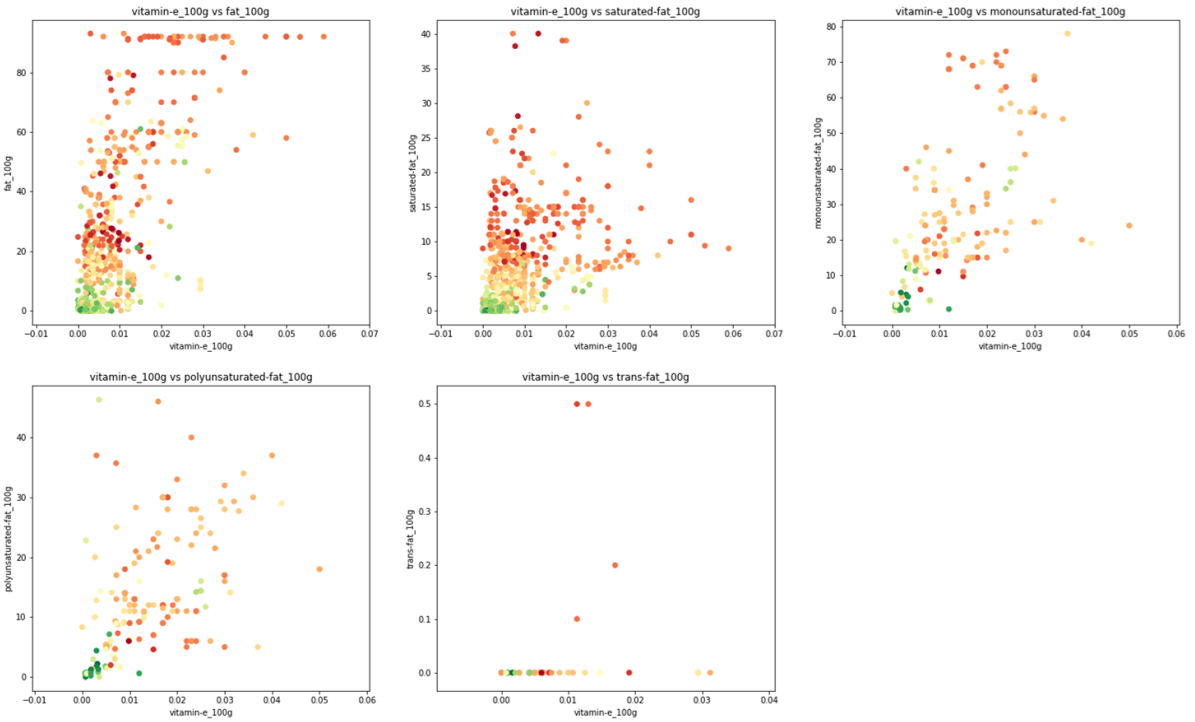

Et on peut voir sur le graphe (ligne 2/colonne 1) que même sans graisse, le produit est calorifique mais très rapidement seul la graisse est l'apport d'énergie car le tracé est en "pointe".

Pour rentrer plus en détail, on peut regarder s'il y a une graisse spécifique à privilégier. Pour ce faire on peut tracer des nuages de points avec les différents ratios de graisse spécifique par rapport à la graisse totale. La couleur est liée à son score nutritionnel. Ci-dessous on a le ratio de monounsaturated fat par rapport à la graisse totale. On peut voir que plus la portion de cette graisse est importante, plus on peut avoir de quantité de graisse en gardant un score correct. C'est la même conclusion avec le poly-unsaturated-fat. Cela s'explique car les graisses non saturées sont les premières à être utilisé comme combustible par le corps pour en faire de l'énergie.



## Corrélation forte : Vitamine E et graisses

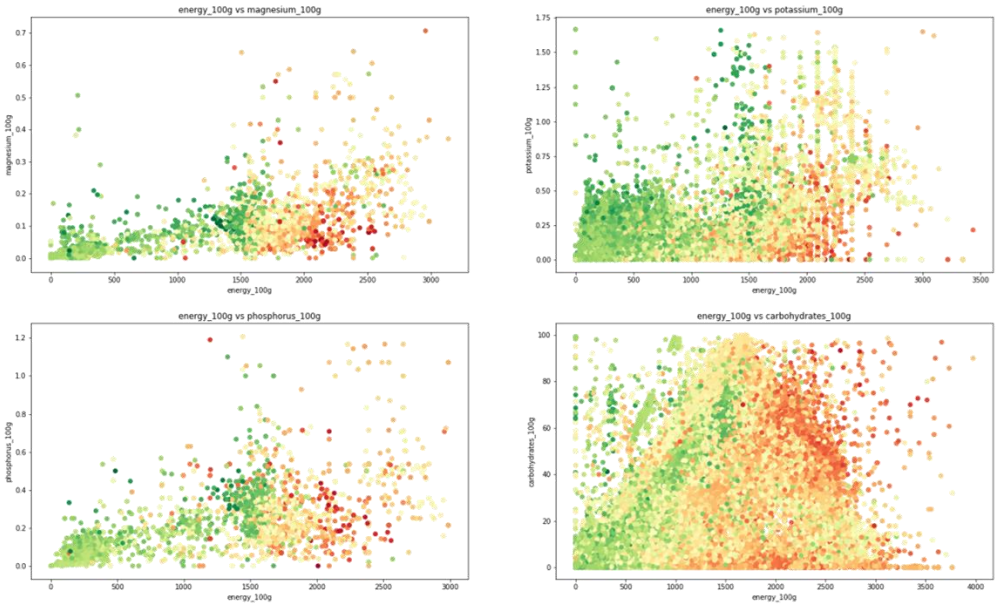
Une corrélation un peu particulière est celle entre les vitamines E et les graisses. Si on fait le même nuage de point on remarque qu'en effet il y a une corrélation forte




Après une recherche, cela s'explique car la vitamine E est en fait soluble dans la graisse. De ce fait, plus on mange gras, plus on assimile de vitamine E. Cependant, arrivé un certain seuil, les produit ayant trop de vitamine E se retrouve avec un mauvais score car il a beaucoup de graisse et parce que l'apport journalier recommandé n'est que de 15mg/j.

## Corrélation forte : Energie et magnésium, potassium, phosphore, carbohydrates

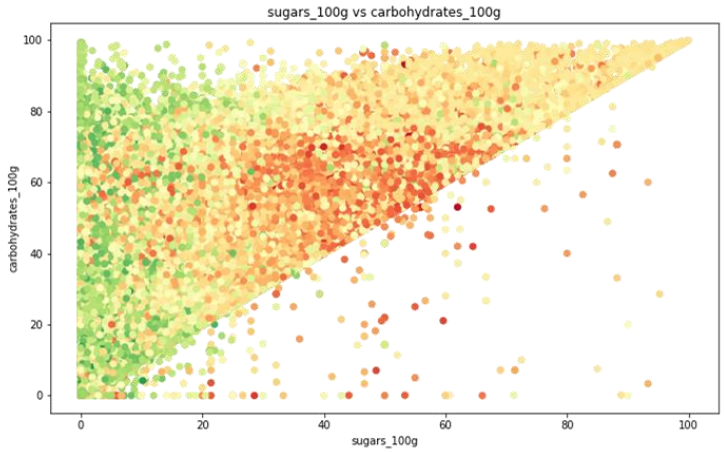
On peut aussi voir une corrélation entre l'énergie et le magnésium, potassium, phosphore, carbohydrates. De la même manière on peut tracer le nuage de points et on a :



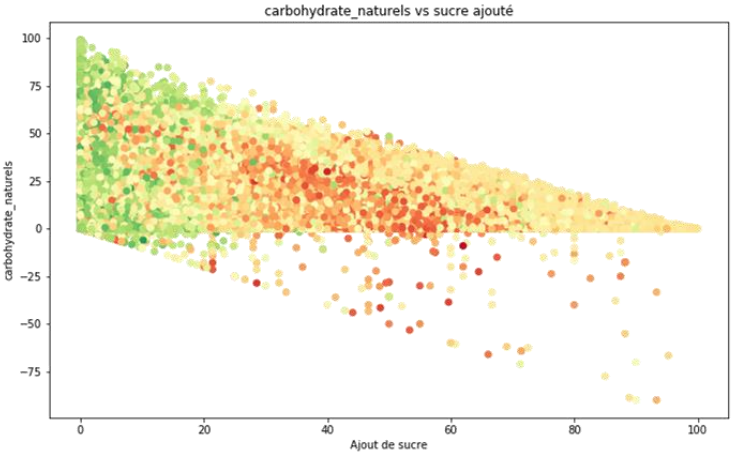
On peut remarquer en effet une certaine relation entre les features. Cependant on peut aussi remarquer 2 choses. Les bonnes choses sont peu calorifiques et il y a un groupe de bons aliments autour de 1500 kcal particulièrement visible sur le graphique avec le potassium. On verra cette marque lors d'une future analyse. Cependant, je n'ai pas trouvé de raisons particulières à ce phénomène.

### Corrélation forte : Sucres et carbohydrates

Finalement on peut aussi s'attarder sur la relation entre les sucres et carbohydrates. Ceux-ci sont de la même famille (glucides) et l'apport de sucre apporte directement des carbohydrates comme on peut le voir sur le nuage ci-dessous:




On remarque que les meilleurs produits ont peu de sucres mais une répartition homogène en carbohydrates. Cela s'explique car les carbohydrates sont présents à l'état naturel et le sucre est un ajout. De ce fait on peut aussi faire un nuage de point des carbohydrates naturels vs le sucre ajouté et on trouve:

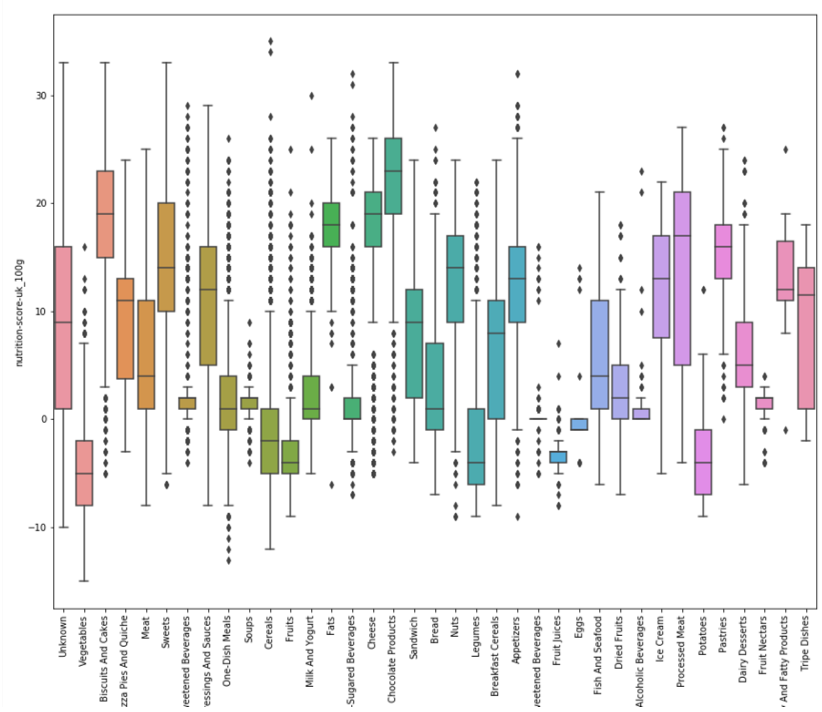


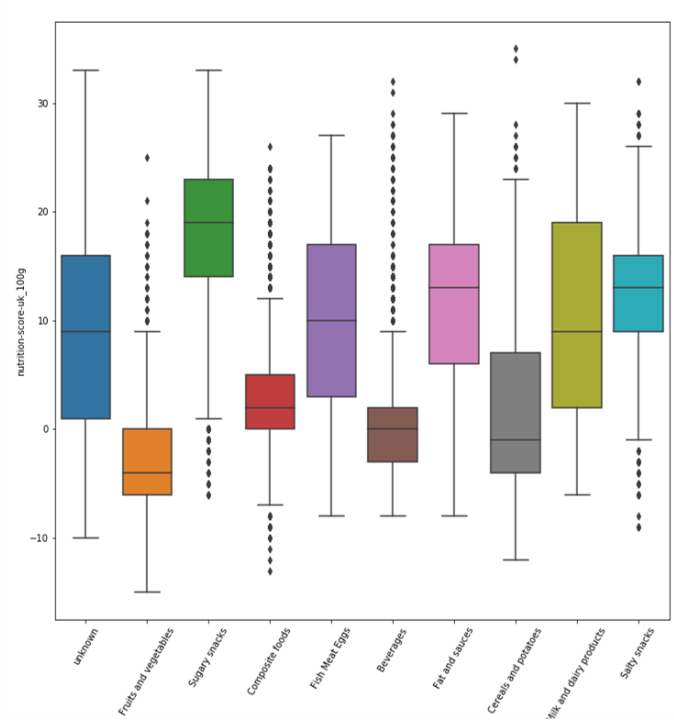

On trouve aussi sur ce graphe une possible règle additionnelle de nettoyage mais n'étant pas sûr, je n'ai pas ajouté ce filtrage (supprimer si carbohydrate < sucres). On voit ainsi plus facilement que les carbohydrates naturels n'impacte pas le score du produit.

La seconde façon d'analyser a été de regarder manuellement certaines features. On trouve notamment :

### Score par groupe PNNS

On peut afficher par groupe PNNS 1 et 2 les boxplot des scores et on trouve :

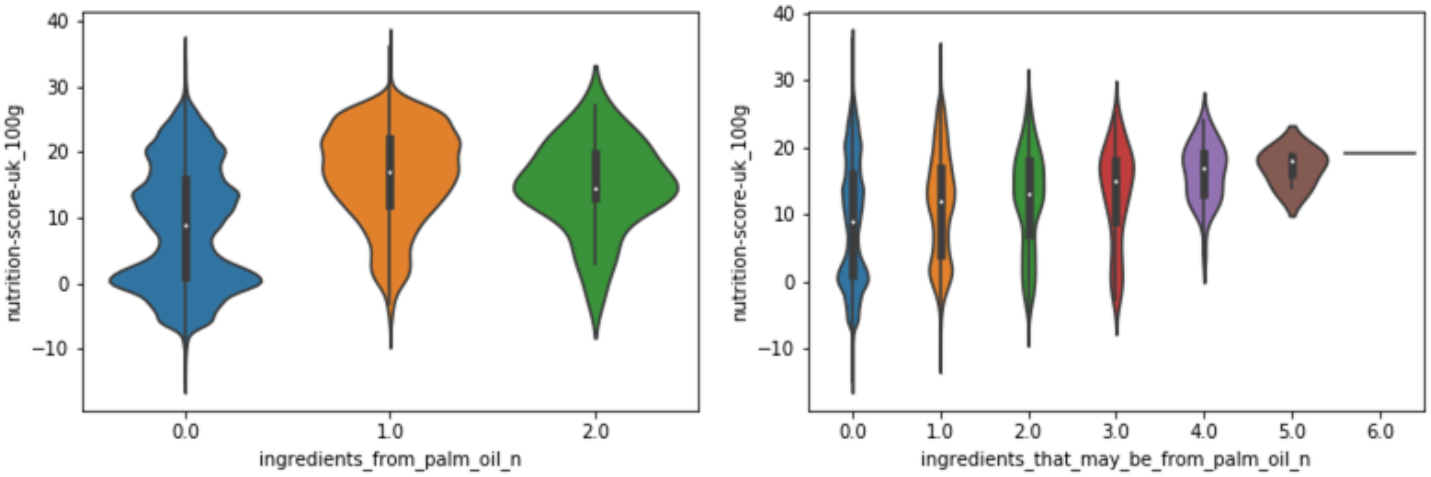



On remarque avec ces graphes certains points très connues. Par exemple on peut voir que les snacks sucrées sont globalement très mauvais. Il en existe des bons mais sont rare (on pensera notamment aux barres énergétiques pour les sportifs). On peut aussi voir que les fruits et légumes sont les meilleurs sauf certains outliers. C'est la même conclusion avec les boissons ou les sodas sont donc les outliers avec un mauvais score.

### Evolution du score en fonction de la présence d'huile de palme

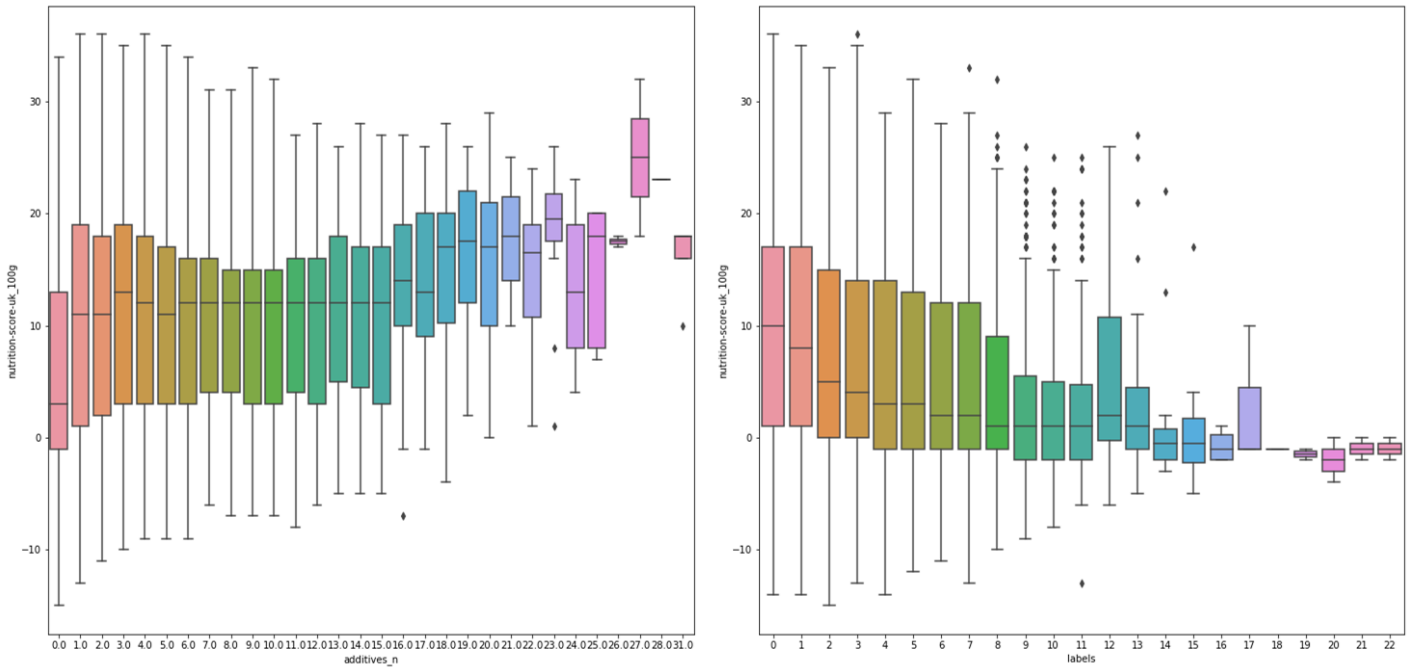
On peut aussi regarder l'évolution du score en fonction du nombre d'ingrédient venant de l'huile de palme qui est très controversée




On remarque que les produits sans huile de palmes sont répartis presque égalitairement en fonction du score. Par contre dès que les produits ont des ingrédients venant d'huile de parle, le score augmente sensiblement.

### Evolution du score en fonction de la présence d'huile de palme

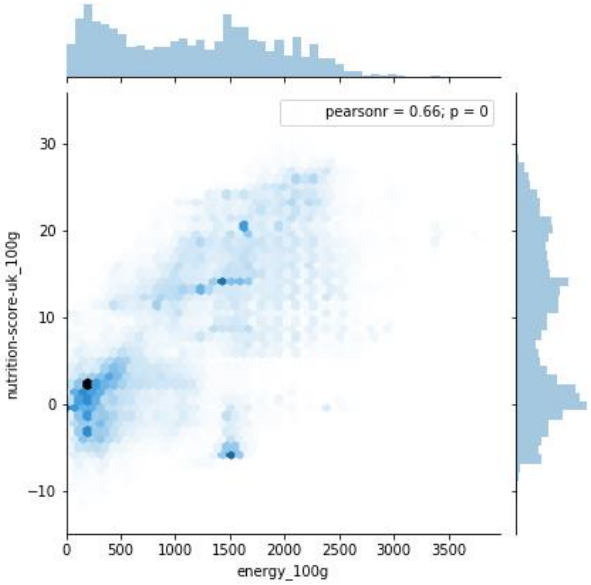
On peut faire de même avec les données que l'on a agrégé (nombres de labels ou d'additifs)




On remarque que plus le produit a d'additifs, moins il est naturel donc moins il est bon. Inversement, on remarque que plus le produit possède de labels, meilleur il est car il a été produit dans de bonne conditions.

### Répartition notes et score

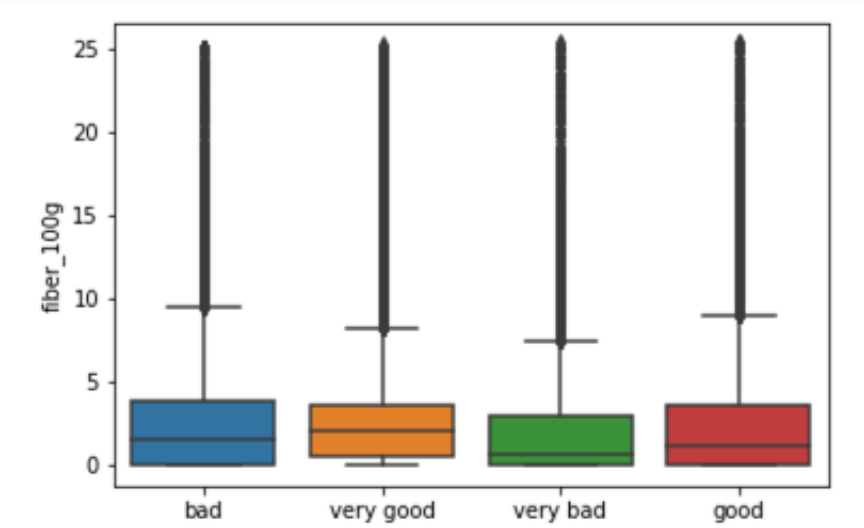
On a vu précédemment qu'il y a un groupe de bons produits autour de 1500 kcal. Si on trace la répartition des produits par énergie et score on trouve :




On remarque aussi le pic inexpliqué de produits aux alentours de 1500 kcal et un score faible. Hormis cette zone, on peut voir une relation entre le score et l'énergie assez forte.

### Répartition des fibres

Une des dernières analyses a été la répartition des fibres en fonction du score. Pour simplifier, les produits ont été groupés par quartile pour avoir un groupe de Très bon produit, bon, mauvais ou très mauvais. Si on regarde le taux de fibres pour chaque groupe on trouve :




On remarque donc que l'apport de fibre à peu près constant pour tous les types de produits. Cependant les produits très mauvais possèdent légèrement moins de fibres car ils sont majoritairement faits à base de sucre

## Synthèse

En synthèse, on peut dire que l'énergie calorifique des produits est très liée à la quantité de graisses. Parmi les graisses, il faut mieux privilégier les graisses non saturées. Les produit à privilégier sont les produits peu calorifique, si possible labélisés, avec peu d'allergènes et d'additifs. Concernant les types de produits, il faut privilégier les produits faits à base de fruits, légumes, œufs, céréales. Les produits à éviter sont les produits avec beaucoup de sucres ajoutés (gâteaux, sodas) ou les fromages et produits gras.

Concernant les vitamines, on a peu de critères spécifiques. Il est difficile avec les valeurs par 100g d'arriver à déterminer les apports journaliers à avoir. Avoir plus de données pourrait permettre cette analyse.

## Evolution possible

Concernant l'évolution, on a actuellement trop de données manquantes pour faire un modèle de prédiction de la qualité nutritionnelle d'une recette. Cependant avec plus de données, on pourrait potentiellement mettre en place

### Un Regresseur

L'objectif serait de prédire le score en fonction des données nutritionnelles. Pour ce faire on pourrait tester:

- Un modèle linéaire : SGD Regressor car on a beaucoup trop de points pour faire la régression classique.
- Un modèle de proximité : KNN Regressor mais sur 200 000 points ce n'est pas sûr de passer
- Un modèle non linéaire : SVM pour mettre en place le kernel trick pour linéariser le problème

### Un Classifieur

L'objectif serait de prédire si le produit est bon ou pas bon (il est aussi possible de créer plus de classes comme bon, moyen et mauvais et faire une classification OVO ou OVA). Pour ce faire on pourrait utilise:

- Un modèle linéaire : Regression logistique pour avoir une probabilité d'être bon ou mauvais
- Un modèle de proximité : KNN Classifieur mais sur 200 000 points ce n'est pas sûr de passer non plus
- Un modèle non linéaire : SVC pour mettre en place le kernel trick pour linéariser le problème pour le classifier

# Conclusion

En conclusion, le nettoyage a été plutôt efficace car il a permis de conserver une grande partie des données et ainsi pouvoir extraire diverses tendances. Cependant, il n'est actuellement pas possible de pouvoir mettre en place un modèle de prédiction de qualité nutritionnelle automatique.

Divers paramètres ont été mis en relief pour concevoir des produits sains (notamment les graisses, l'énergie, les additifs, les labels, …) ainsi que des produits privilégier pour ce faire (fruits, légumes, pommes de terres, …)

Un seul bémol est cependant à mettre en relief. Les scores nutritionnels ont été utilisés pour déterminer les features importantes mais la note est aussi basée sur ces features. De ce fait, on a plus fait du reversed engineering que du feature engineering.