Détection de régions statistiquement peu probables dans un nuage de points

Noé Aubin-Cadot

12 mars 2020

But

<ins>But :</ins> Détecter des régions statistiquement peu probables dans un nuage de points à valeurs binaires.

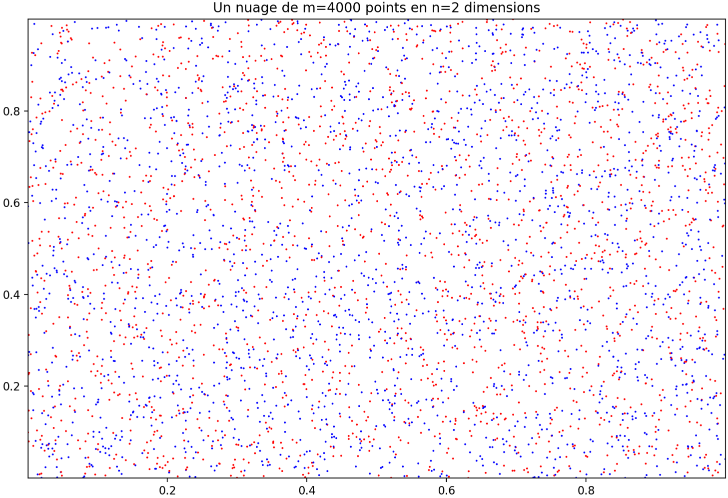
Plan

Plan :

1. Visualiser le problème.
2. Algorithme de détections.
3. Utilisation de l’algorithme.
4. Réflexions à propos de l’algorithme.

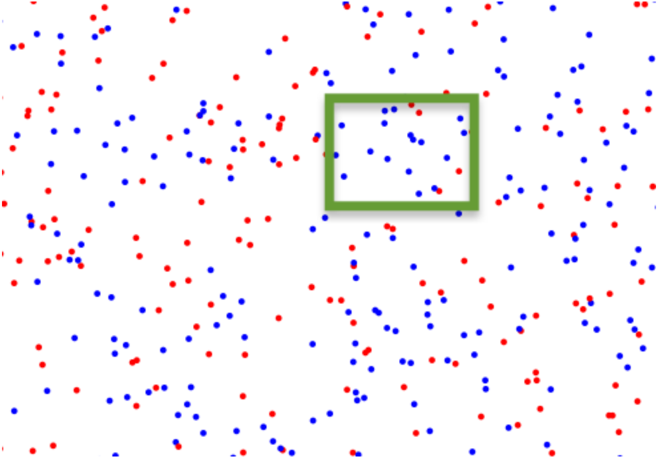
Visualiser le problème

On considère un nuage de m points dans un espace de dimension n. Chaque point est à valeurs binaires $\{0, 1\}$.



Visualiser le problème

On cherche des régions où la proportion locale de `0` et de `1` est éloignée de la proportion globale de `0` et de `1`.



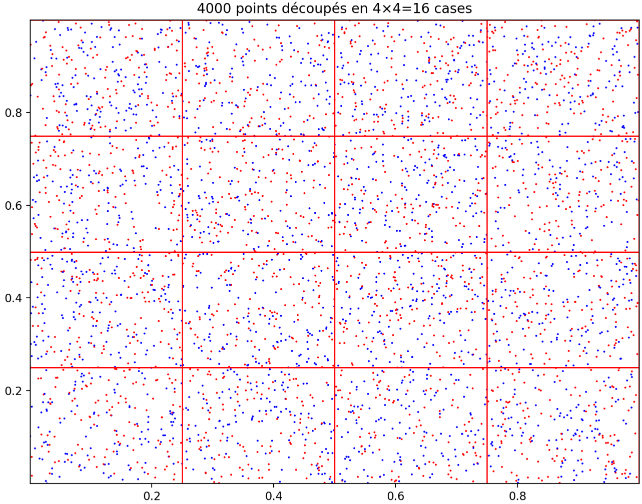
Algorithme de détections

Il existe plusieurs manières d'aborder ce problème.

En voici une.

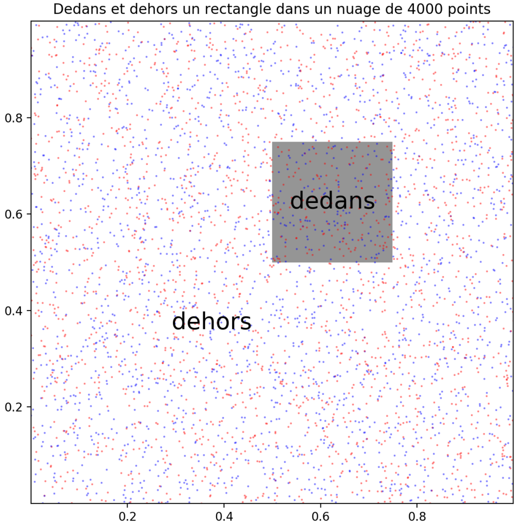
Algorithme de détections

D’abord, on découpe le nuage de points selon une grille de résolution $k_1 \times k_2 \times \ldots \times k_n$.



Algorithme de détections

On choisit une case de la grille et on dénombre les points `0` et `1` en dedans et en dehors de cette case.



Algorithme de détections

On place ces dénombrements dans un tableau de contingence O des valeurs observées :

	0	1	*total*
dedans	122	140	262
dehors	1904	1834	3738
total	2026	1974	4000

Algorithme de détections

On se donne comme *hypothèse nulle* H_0 qu'il n'existe pas de relation entre « le nombre de `0` et de `1` » et « le dedans et le dehors ».

Le tableau O induit un tableau de contingence T des valeurs théoriques correspondantes à l'hypothèse nulle :

	`0`	`1`	*total*
dedans	$262 \cdot 2026/4000$	$262 \cdot 1974/4000$	262
dehors	$3738 \cdot 2026/4000$	$3738 \cdot 1974/4000$	3738
total	2026	1974	4000

Algorithme de détections

On évalue alors le χ^2 entre les valeurs observées O et les valeurs théoriques T de l'hypothèse nulle :

$$\chi^2(O,T) = \sum_{\substack{i=\text{dedans,dehors} \\ j=0,1}} \frac{(T_{i,j} - O_{i,j})^2}{T_{i,j}} \approx 1.872$$

Ici il y a $\nu = (2-1)\cdot(2-1) = 1$ *degré de liberté*.

Pour déterminer si l'hypothèse nulle est rejetée ou non par l'observation on doit se donner un *risque* α_{case} de rejeter à tort H_0 par un test de χ^2. Lequel prendre ?

Algorithme de détections

La probabilité de détecter "quelque chose" pour du bruit dans une case est α_{case}.

La probabilité de détecter "quelque chose" pour du bruit sur toute la grille est :

$$\alpha_{\text{grille}} = 1 - (1 - \alpha_{\text{case}})^{\text{résolution utile}}$$

Ici, la résolution utile est le nombre de cases de la grille qui possèdent "assez de points".

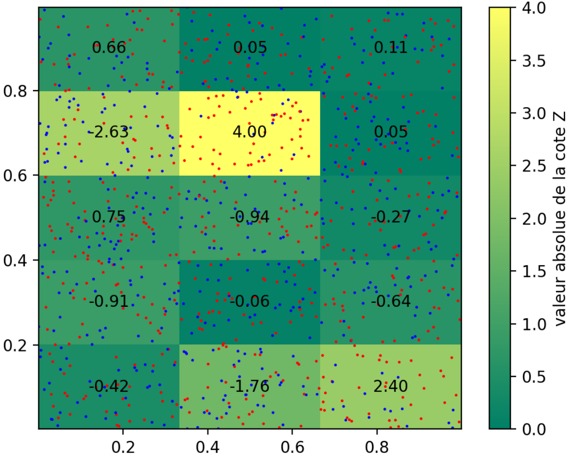
La probabilité de détecter "quelque chose" pour du bruit selon toutes les grilles de résolution $k_1 \times ... \times k_n$, $k_i \leq K$, est :

$$\alpha_{\text{total}} = 1 - (1 - \alpha_{\text{grille}})^{K^n - 1}$$

Le nombre de grilles différentes utilisées est le nombre de résolutions possibles moins la grille $1 \times ... \times 1$. On prend $\alpha_{\text{total}} = 0.05$.

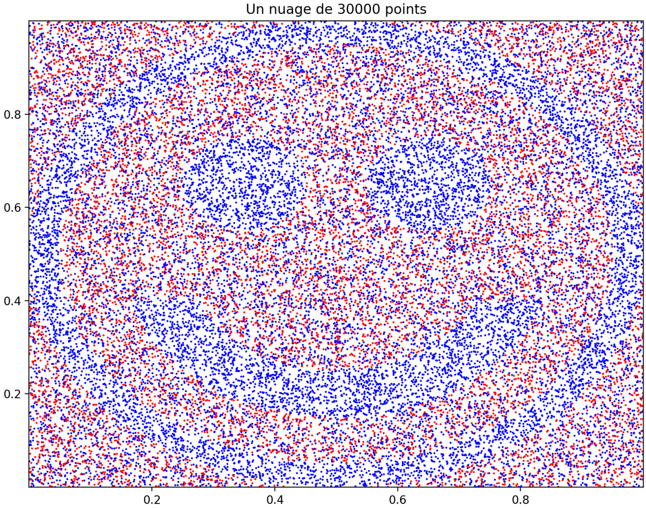
Algorithme de détections

Un autre critère de détection équivalent est de se donner une borne sur la cote Z entre la proportion de `0` et de `1` dans une case versus la proportion globale du nuage de points.



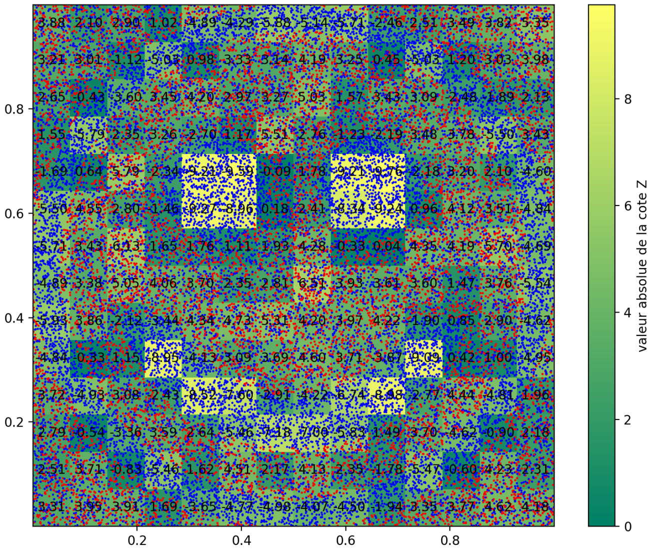
Utilisation de l'algorithme

Considérons un nuage de points.



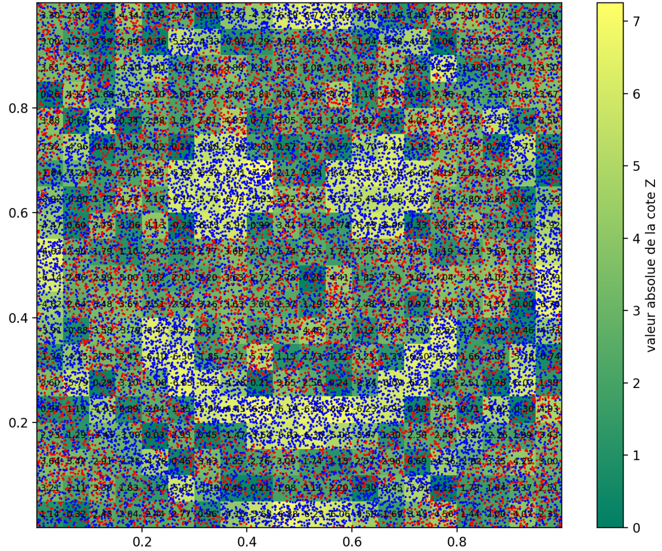
Utilisation de l'algorithme

La cote Z varie d'une case à l'autre.



Utilisation de l’algorithme

On peut faire varier la résolution du détecteur.



Réflexions à propos de l'algorithme

En dimension $n = 2$ et pour une résolution maximale de $K \times K$, l'algorithme roule rapidement sur un ordinateur portable, même pour un très grand nombre m de points. En particulier, il existe au plus m cases qui contiennent des points. Ce faisant, les cases sont les composantes d'un array `numpy` *sparse* d'au plus m valeurs non nulles.

En dimension $n \gg 2$, la complexité algorithmique est énorme :

$$\mathcal{O}(mK^{2n})$$

Un mK^n est dû à la classification des m points en les cases et l'autre K^n est dû à toutes les résolutions $k_1 \times \ldots \times k_n$, $k_i \leq K$.

Enfin, l'algorithme n'est pas utilisable sur des données incomplètes où les coordonnées de certains points sont inconnues.

Merci de votre attention ☺