

Reminder on Markov chains - Stochastic gradient descentExercise 1

1) On va utiliser la méthode la fonctionnelle.

Soit h continue bornée: $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$

$$\begin{aligned} E_{x,y} [h(x,y)] &= E_{R,\theta} [h(R\cos\theta, R\sin\theta)] \\ &= \int_{\mathbb{R}^2 \times [0,2\pi]} h(r\cos\theta, r\sin\theta) f_R(r) f_\theta(\theta) dr d\theta \quad \text{car } R \perp\!\!\!\perp \theta \\ &= \int_{\mathbb{R} \times \mathbb{R}} h(r\cos\theta, r\sin\theta) f_R(r) \mathbf{1}_{(r \geq 0)} f_\theta(\theta) \mathbf{1}_{(\theta \in [0,2\pi])} dr d\theta \end{aligned}$$

On pose $\underline{\Phi}$: $\begin{cases} \mathbb{R}^2 \times [0,2\pi] = \mathbb{R} \times \mathbb{R} \\ (r, \theta) \mapsto r\cos\theta, r\sin\theta \end{cases}$ $\underline{\Phi}$ est C^1 , bijective

$$\text{Jac}[\underline{\Phi}](r, \theta) = \begin{pmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{pmatrix}$$

$$|\text{Jac}[\underline{\Phi}](r, \theta)| = r\cos^2\theta + r\sin^2\theta = r > 0$$

Avec $x = r\cos\theta$ et $y = r\sin\theta$, on a $|\text{Jac}|^{-1} = \frac{1}{r} = \frac{1}{\sqrt{x^2+y^2}}$

$$\text{Il vient } E_{x,y} [h(x,y)] = \int_{\mathbb{R} \times \mathbb{R}} h(x,y) \underbrace{\frac{1}{\sqrt{x^2+y^2}}}_{|\text{Jac}|^{-1}} \times \underbrace{\frac{1}{2\pi}}_{f_\theta(\theta)} \times \underbrace{\int_0^\infty r e^{-\frac{r^2}{2}} dr}_{f_R(r)} dx dy$$

$$\text{Soit } E_{x,y} [h(x,y)] = \int_{\mathbb{R} \times \mathbb{R}} h(x,y) \times \exp\left(-\frac{x^2}{2}\right) \exp\left(-\frac{y^2}{2}\right) \times \frac{1}{\sqrt{2\pi}} \times \frac{1}{\sqrt{2\pi}} dx dy$$

On reconnaît $E_{X,Y} [h(x,y)] = \int_{\mathbb{R} \times \mathbb{R}} h(x,y) f_X(x) f_Y(y) dx dy$

d'où $X \perp\!\!\!\perp Y$

et $f_X(x) \sim \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x^2}{2}\right)$ soit $x \sim \mathcal{U}(0,1)$. Pareil pour Y .

2. On cherche la loi d'une distribution de Rayleigh de paramètre 1.

$$F_R(r) = \int_0^{+\infty} f_R(r) dr = \int_0^{+\infty} r \exp\left(-\frac{r^2}{2}\right) dr = \left[-\exp\left(\frac{-r^2}{2}\right) \right]_0^{+\infty} = 1 - \exp\left(-\frac{r^2}{2}\right)$$

On cherche $F^{-1}(u)$ pour $u \in [0,1]$ soit r tq $F(r) = u$.

$$1 - \exp\left(-\frac{r^2}{2}\right) = u \iff -\frac{r^2}{2} = \ln(1-u) \iff r = \sqrt{-2\ln(1-u)}$$

on vérifie bien $\ln(1-u) < 0$ et $r > 0$.

Algorithme: on prend $U \sim \mathcal{U}([0,1])$, $\tilde{U} \sim \mathcal{U}([0,1])$ et $\theta = 2\pi \times \tilde{U}$
avec $U \perp\!\!\!\perp \tilde{U}$.

- Sample U and \tilde{U} . Get θ
- $r = \sqrt{-2\ln(1-U)}$
- return $r \cos \theta$ and $r \sin \theta$.

3. a. On a $U_1 \perp\!\!\!\perp U_2$, $V_1 = 2U_1 - 1$ et $V_2 = 2U_2 - 1$.

Clairement, $V_1 \perp\!\!\!\perp V_2$ et $V_1, V_2 \sim \mathcal{U}([-1,1])$

On cherche la loi de (V_1, V_2) à la fin de la boucle while

$$E_{V_1, V_2} [h(V_1, V_2)] = \int_{(-1,1) \times (-1,1)} h(V_1, V_2) f(V_1, V_2) \mathbf{1}_{\{V_1^2 + V_2^2 \leq 1\}} dV_1 dV_2$$

La contrainte $V_1^2 + V_2^2 \leq 1$ nous fait décrire un cercle centré en $\mathbf{0}$ de rayon 1.

$$\text{Soit } f(V_1, V_2) = \frac{1}{\pi} \mathbf{1}_{\{V_1^2 + V_2^2 \leq 1\}}$$

b) Pour montrer l'indépendance de (T_1, T_2) et V , on va montrer

$$T_1 \perp\!\!\!\perp V \text{ puis } T_2 \perp\!\!\!\perp V \Rightarrow (T_1, T_2) \perp\!\!\!\perp V.$$

On posera les changements de variable suivants : $T_1 = \frac{V_1}{\sqrt{V}}$, $V_2^2 = V(1-T_1^2)$

$$\text{On aura donc } T_2 = \pm \sqrt{1-T_1^2} \text{ en disant } T_2 = \frac{V_2}{\sqrt{V}}.$$

$$\left\{ \begin{array}{l} V_2 = \pm \sqrt{V(1-T_1^2)} \end{array} \right.$$

Ce changement de variable n'est pas bijectif, il faudra donc "couper" le dosage en $V_2 > 0$ et $V_2 \leq 0$. (resp. \mathbb{D}_1^+ et \mathbb{D}_1^-).

$$\bullet \mathbb{E}[h(T_1, V)] = \int_{\mathbb{D}_1} h\left(\frac{V_1}{\sqrt{V_1^2 + V_2^2}}, \frac{V_2^2 + V_1^2}{\sqrt{V_1^2 + V_2^2}}\right) f_{(V_1, V_2)}(V_1, V_2) dV_1 dV_2$$

$\underbrace{f_{(V_1, V_2)}}_{1/\pi}$

$$= \int_{\mathbb{D}_1^+} \dots + \int_{\mathbb{D}_1^-}$$

$$\text{On pose } \underline{\Phi}(V_1, V_2) = \left(\frac{V_1}{\sqrt{V_1^2 + V_2^2}}, \frac{V_2^2 + V_1^2}{\sqrt{V_1^2 + V_2^2}} \right).$$

$$\text{On sait que } |\text{Jac}[\underline{\Phi}](u)|^{-1} = |\text{Jac}[\underline{\Phi}^{-1}](\underline{\Phi}(u))|.$$

On calcule $\underline{\Phi}^{-1}$, qui n'est pas bijection.

$$\text{Sur } \mathbb{D}_1^+: \underline{\Phi}^{-1}(T_1, V) = \left(\underbrace{T_1 \sqrt{V}}_{V_1}, \underbrace{\sqrt{V(1-T_1^2)}}_{V_2} \right)$$

$$\text{sur } \mathbb{D}_1^-: \underline{\Phi}^{-1}(T_1, V) = \left(T_1 \sqrt{V}, -\sqrt{V(1-T_1^2)} \right).$$

$$\text{Jac}[\underline{\Phi}^{-1}](T_1, V) = \begin{pmatrix} \sqrt{V} & \frac{T_1}{2\sqrt{V}} \\ \frac{-\sqrt{V}T_1}{\sqrt{V(1-T_1^2)}} & \frac{\pm(1-T_1^2)}{2\sqrt{V(1-T_1^2)}} \end{pmatrix}$$

sur \mathbb{D}_1^+ sur \mathbb{D}_1^-

$$\text{Sur } \mathcal{D}_1^+: |\text{Jac}[\Phi^{-1}](T_1, V)| = \frac{\sqrt{V}(1-T_1^2)}{2\sqrt{V(1-T_1^2)}} + \frac{\sqrt{T_1}}{\sqrt{V(1-T_1^2)}} \times \frac{T_1}{2\sqrt{V}} \\ = \frac{\sqrt{V}(1-T_1^2) + \sqrt{V}T_1^2}{2\sqrt{V(1-T_1^2)}} = \frac{1}{2\sqrt{1-T_1^2}} > 0$$

$$\text{Sur } \mathcal{D}_1^-: |\text{Jac}[\Phi^{-1}](T_1, V)| = \left| \frac{-\sqrt{V}(1-T_1^2)}{2\sqrt{V(1-T_1^2)}} - \frac{\sqrt{T_1}}{\sqrt{V(1-T_1^2)}} \times \frac{T_1}{2\sqrt{V}} \right| \\ = \left| \frac{-\sqrt{V}(1-T_1^2) - \sqrt{V}T_1^2}{2\sqrt{V(1-T_1^2)}} \right| = \left| \frac{-1}{2\sqrt{1-T_1^2}} \right| = \frac{1}{2\sqrt{1-T_1^2}}$$

On admet : $\mathbb{E}[h(T_1, V)] = \int_{\mathcal{D}_1^+} h(t_1, v) \frac{1}{\pi} \times \frac{1}{2\sqrt{1-t_1^2}} dt_1 dv$

$$+ \int_{\mathcal{D}_1^-} h(t_1, v) \frac{1}{\pi} \times \frac{1}{2\sqrt{1-t_1^2}} dt_1 dv$$

Suit $\mathbb{E}[h(T_1, V)] = \frac{2}{\pi} \int_{\mathcal{D}_1^+} h(t_1, v) \frac{1}{2\sqrt{1-t_1^2}} dt_1 dv$

Il vient que $T_1 \perp\!\!\!\perp V$ car on a $f_{T_1}(t_1) = \frac{1}{2\sqrt{1-t_1^2}}$ et $f_V(v) = \frac{1}{\pi}$

En remettant sur \mathbb{R}^2 : $V \sim \mathcal{U}([0, 1])$ et T_1 une loi de

densité $f_{T_1}(t) = \frac{1}{\pi\sqrt{1-t^2}} \quad t \in [-1, 1]$

- On sait que V_1 et V_2 sont indépendants, et il vient que T_1 et T_2 jouent même rôle par rapport à V donc $T_2 \perp\!\!\!\perp V$, T_2 suit une loi $f_{T_2}(t) = \frac{1}{\pi\sqrt{1-t^2}} \quad t \in [-1, 1]$.

- On cherche à montrer que T_1 a une distribution en cos θ .

On sait que $f_{T_1}(t) = \frac{1}{\pi \sqrt{1-t^2}} \mathbf{1}_{[-1,1]}$.

On pose le changement de variable $t_1 = \cos \theta \Leftrightarrow \theta = \arccos t_1$
 t_1 est bien défini sur $[-1,1]$

Donc $d\theta = -\frac{1}{\sqrt{1-t_1^2}} dt_1 \Leftrightarrow dt_1 = -\sqrt{1-t_1^2} d\theta$.

On a alors $E[h(T_1)] = \int_{\arccos(-1)}^{\arccos(1)} -h(t_1) \times \frac{\sqrt{1-t_1^2}}{\pi \sqrt{1-t_1^2}} d\theta$

soit $E[h(T_1)] = \int_{-\pi}^0 -h(\cos \theta) \frac{1}{\pi} d\theta = \frac{1}{\pi} \int_0^\pi h(\cos \theta) d\theta$

On sait que $\int_0^\pi h(\cos \theta) d\theta = \int_{\pi}^{2\pi} h(\cos \theta) d\theta$ par symétrie.

$$\begin{aligned} \text{Donc } E[h(T_1)] &= E[h(\cos \theta)] = \frac{1}{\pi} \times \frac{1}{2} \left[\int_0^\pi h(\cos \theta) d\theta + \int_{\pi}^{2\pi} h(\cos \theta) d\theta \right] \\ &= \frac{1}{2\pi} \int_0^{2\pi} h(\cos \theta) d\theta \\ &= \int_0^{2\pi} h(\cos \theta) \times \frac{1}{2\pi} d\theta. \end{aligned}$$

On a donc que T_1 suit la même loi que θ avec $\theta \sim U([0, 2\pi])$.

On sait que $T_2^2 = 1 - T_1^2$ donc $T_1^2 + T_2^2 = 1$

Il en résulte que T_2 suit la même loi que θ (celui de T_1 qui a $U([0, 2\pi])$)

c. On sait que $S = \sqrt{2\ln(V_1^2 + V_2^2)}$ et $V_1^2 + V_2^2 \sim \mathcal{U}([0,1])$

On sait aussi que pour la fdp de Rayleigh(1), on a $F^{-1}(u) = \sqrt{-2\ln(1-u)}$ $\forall u \in [0,1]$

On peut donc assimiler S à $F^{-1}(1 - (V_1^2 + V_2^2)) = F^{-1}(1 - r)$

D'où $S \sim R(1)$

$$X = ST_1 \sim R \cos \theta \quad Y = ST_2 \sim R \sin \theta$$

On se retrouve avec les variables aléatoires de la question 1, d'où $X \sim \mathcal{N}(0,1)$, $Y \sim \mathcal{N}(0,1)$ et $X \perp\!\!\!\perp Y$.

d. On définit la VA $Z \in \mathbb{N}^*$ le nombre de tirages nécessaires avant de sortir de la boucle while.

$$\text{D'où } E(Z) = \sum_{n \in \mathbb{N}^*} P(Z=n)n.$$

On définit une seconde VA $(B_n)_{n \in \mathbb{N}^*}$ = $\begin{cases} 1 & \text{si sortie de boucle à l'étape } n \\ 0 & \text{sinon.} \end{cases}$

La loi de B_n pour l'événement $\{Z=n\}$

$$\text{est donc : } \left\{ \prod_{i=1}^{n-1} B_i = 0, B_n = 1 \right\}$$

Comme chaque tirage en entrée de boucle est indépendant des boucles précédentes, les $(B_n)_{n \in \mathbb{N}^*}$ sont iid

$$\text{D'où } P(Z=n) = P\left(\prod_{i=1}^{n-1} B_i = 0, B_n = 1\right) = \prod_{i=1}^{n-1} P(B_i = 0) \times P(B_n = 1)$$

Si on revient à l'interprétation géométrique: on tire des points dans un carré de côté 2, centré en 0. On répète la boucle while tant que le point est à l'extérieur du disque centré en 0 de rayon 1.

Et $\text{carre} = 4$ et $\text{disque} = \pi$. On a donc une probabilité $\frac{\pi}{4}$ de sortir de la boucle à chaque boucle, tandis que la zone de rejet est de $1 - \frac{\pi}{4}$.

$$\text{On a donc } \left\{ \begin{array}{l} \mathbb{P}(B_i^o = 0) = \mathbb{P}(B_1 = 0) = 1 - \frac{\pi}{q} \quad \forall i \in \mathbb{N}^*. \\ \mathbb{P}(B_i^o = 1) = \mathbb{P}(B_1 = 1) = \frac{\pi}{q} \quad \forall i \in \mathbb{N}^* \end{array} \right.$$

$$\begin{aligned} \text{On revient à } \mathbb{P}(Z=n) &= \prod_{i=1}^{n-1} \mathbb{P}(B_i^o = 0) \times \mathbb{P}(B_n = 1) \\ &= \left(\mathbb{P}(B_1 = 0) \right)^{n-1} \times \mathbb{P}(B_n = 1) \\ &= \left(1 - \frac{\pi}{q} \right)^{n-1} \times \frac{\pi}{q} \end{aligned}$$

On reconnaît une loi géométrique de paramètre $\frac{\pi}{q}$, dont l'espérance est donc $\left(\frac{\pi}{q}\right)^{-1} \approx 1,27$. $E(Z) = 1,27$

Le nombre d'étapes en moyenne est de 1,27.

Exercise 2

On a $E[h(X_{n+1}) | X_n = x] = \int_{\mathbb{R}} h(y) f(y_0, y) \mathbf{1}_{\{y_0=x\}} dy$
 $(X_n)_{n \geq 0}$ est à valeurs dans $[0, 1]$ donc on peut restreindre l'intégrale à cet intervalle.

$$= \int_{[0, 1]} h(y) f_x(y) dy.$$

Il vient que le noyau de transition $P(x, A) = \int_{A \cap [0, 1]} h(y) f_x(y) dy$.

Soit $m \in \mathbb{N}^*$.

- Si $X_n = x \neq \frac{1}{m}$, on sait que $X_{n+1} \sim U([0, 1])$

soit $P(x, A) = \int_{A \cap [0, 1]} \frac{1}{[0, 1]^2} dt = \int_{A \cap [0, 1]} dt \quad x \neq \frac{1}{m}$

- Si $X_n = x = \frac{1}{m}$.

On note $(B_n)_{n \in \mathbb{N}^*}$ les variables de Bernoulli iid de paramètres x_n^2 .

$$\begin{aligned} E[h(X_{n+1}) | X_n = x] &= \int h(x_{n+1}) p(\mu_{n+1} | x_n = x) dx_{n+1} \\ &\equiv \int h(x_{n+1}) p(\mu_{n+1}, B_n = 0 | x_n) dx_{n+1} \\ &\quad + \int h(x_{n+1}) p(\mu_{n+1}, B_n = 1 | x_n) dx_{n+1} \\ &\equiv \int h(x_{n+1}) p(\underbrace{\mu_{n+1} | x_n, B_n = 1}_{U([0, 1])}) p(B_n = 1 | x_n) dx_{n+1} \\ &\quad + \int h(x_{n+1}) p(\underbrace{\mu_{n+1} | x_n, B_n = 0}_{1/m+1}) p(B_n = 0 | x_n) dx_{n+1} \\ &= \int h(x_{n+1}) \prod_{y \in [0, 1]} x_n^2 dx_{n+1} + \int h(x_{n+1}) \sum_{m=1}^M (1 - x_n^2) dx_{n+1} \end{aligned}$$

On prend un $\sum_{m=1}^M$ car il y a une probabilité infinie d'obtenir $\frac{1}{m+1}$ si $B_n = 0$, et on veut une mesure de probabilité.

$$\text{On a } \mathbb{E}[h(x_{n+1}) | X_n = x] = x^2 \int_{[0,1]} h(x_{n+1}) du_{n+1} + \int h(u_{n+1}) \sum_{m=1}^{n+1} (1-u_m^2) du_{n+1}$$

En prenant $h = \frac{1}{dx}$ on obtient:

$$P(x, A) = \begin{cases} \int_{A \cap [0,1]} (1-u^2) \sum_{m=1}^{n+1} (A) dt & \text{si } n = \frac{1}{m} \\ \int_{A \cap [0,1]} dt & \text{sinon.} \end{cases}$$

2. $\Pi \sim \mathcal{U}([0,1])$. Par définition, $\Pi(A) = \int_{A \cap [0,1]} dy -$
 Π invariant pour P si $\Pi P = \Pi$.

$$\Pi P(A) = \int_{[0,1]} \Pi(dx) P(x, A) = \int_{[0,1]} P(x, A) dx$$

$$= \int_{[0,1] \times [0,1] \cap A} dt dx \text{ car } P \text{ moyen de transition de mesure nulle si } x = \frac{1}{m}.$$

$$= \int_{[0,1] \cap A} dt = \Pi(A).$$

Π est bien invariant pour P .

$$3 \quad n \notin \left\{ \frac{1}{m}, m \in \mathbb{N}^* \right\}$$

$$P_f(n) = \mathbb{E}[f(X_1) | X_0 = n] = \int f(y) P(x, dy) = \int f(y) \Pi(y) dy$$

(car $n \notin \left\{ \frac{1}{m}, m \in \mathbb{N}^* \right\}$)

$$\text{Il vient } P_f(n) = \int f(y) \Pi(y) dy = \int_0^1 f(y) dy$$

$$\text{On cherche } P^t f(u) = P(Pf(u))$$

Comme on a pris f quelconque, bornée mesurable. On peut dire

$$Pf(x) = f(y) \text{ vu plus haut.}$$

$$\text{D'où } P^2 f(u) = \int_0^1 Pf(y) dy = \iint_0^1 f(t) dt dy = \int_0^1 f(t) dt = Pf(t).$$

$$\text{On pose l'hypothèse que } P^n f(x) = \int_0^1 f(y) dy \quad \forall n \in \mathbb{N}^*.$$

$$\text{Par récurrence alors } P^{n+1} f(u) = P(P^n f(u))$$

$$\begin{aligned} &= \int_{\mathbb{R}} P(x, dy) P^n f(y) \\ &= \int_0^1 P^n f(y) dy \\ &= \int_0^1 \int_0^1 f(t) dt dy \\ &= \int_0^1 f(t) dt. \end{aligned}$$

On a prouvé notre hypothèse.

$$\lim_{n \rightarrow \infty} P^n f(x) = \int_0^1 f(y) dy = \int_{\mathbb{R}} f(y) \pi(y) dy = \int f(u) \pi(u) du.$$

$$g_j \stackrel{a}{=} \kappa = \frac{1}{m}, \quad m \geq 2 -$$

$$P^1 \left(1, \frac{1}{m+1} \right) = 1 - \kappa^2 = 1 - \frac{1}{m^2}$$

$$P^2\left(x, \frac{1}{m+2}\right) = P\left(P\left(x, \frac{1}{m+2}\right)\right)$$

$$= \int P(x, dy) P\left(y, \frac{1}{m+2}\right)$$

on sait $\mu = \frac{1}{m}$ donc d'après 1), $P(x, dy) = \underbrace{\int_0^x dt}_{dy \in [0, 1]} + (1-\mu^2) \sum_{\frac{1}{m+1}} (dy)$

$$\text{D'où } P^2\left(x, \frac{1}{m+2}\right) = \int (1-\mu^2) \sum_{\frac{1}{m+1}} (dy) P\left(y, \frac{1}{m+2}\right)$$

$$= (1-\mu^2) \int \sum_{\frac{1}{m+1}} (dy) P\left(y, \frac{1}{m+2}\right)$$

$$= (1-\mu^2) P\left(\frac{1}{m+1}, \frac{1}{m+2}\right) = \frac{(1-\mu^2) \times (1 - \left(\frac{1}{m+1}\right)^2)}{0 \left[1 - \left(\frac{1}{m+1}\right)^2\right]}$$

Vue la définition de $P(\mu, A)$, il semble en effet que pour passer de μ à $\frac{1}{m+n}$ il faille passer successivement par $\frac{1}{m+1}, \frac{1}{m+2}, \dots$ les $U([0, 1])$ ne permettant pas cela.

$$\text{On peut donc conjecturer } P^n\left(\mu, \frac{1}{m+n}\right) = \prod_{i=0}^{n-1} P\left(\frac{1}{m+i}, \frac{1}{m+i+1}\right)$$

$$= \prod_{i=0}^{n-1} \left(1 - \left(\frac{1}{m+i}\right)^2\right).$$

On suppose P^n vraie pour un $n \in \mathbb{N}^*$.

$$\text{Alors } P^{n+1}\left(x, \frac{1}{m+n+1}\right) = \int P^n(x, dy) P\left(y, \frac{1}{m+n+1}\right)$$

$$= \int \prod_{i=0}^{n-1} \left(1 - \left(\frac{1}{m+i}\right)^2\right) \sum_{\frac{1}{m+n+1}} (dy) P\left(y, \frac{1}{m+n+1}\right)$$

$$= \prod_{i=0}^{n-1} \left(1 - \left(\frac{1}{m+i}\right)^2\right) P\left(\frac{1}{m+n}, \frac{1}{m+n+1}\right)$$

$$= \prod_{i=0}^{n-1} \left(1 - \left(\frac{1}{m+i}\right)^2\right) \times \left(1 - \left(\frac{1}{m+n}\right)^2\right)$$

$$\text{Il vient bien que } P^{n+1}\left(x, \frac{1}{m+n+1}\right) = \prod_{i=0}^n \left(1 - \left(\frac{1}{m+i}\right)^2\right)$$

vérifiant l'hypothèse de récurrence.

b) Pour $A = \bigcup_{q \in \mathbb{N}} \left[\frac{1}{m+1+q}, \frac{1}{m+q} \right]$ et $x = \frac{1}{m}$, $m \in \mathbb{N}^*$, $m \geq 2$

$$\text{On a } \pi(A) = \int_{A \cap [0,1]} dt = \int_{\bigcup_{q \in \mathbb{N}} \left[\frac{1}{m+1+q}, \frac{1}{m+q} \right] \cap [0,1]} dt = \sum_{q \in \mathbb{N}} \int_{\left[\frac{1}{m+1+q}, \frac{1}{m+q} \right] \cap [0,1]} dt = 0$$

$$\begin{aligned} \text{On a } P^n(x, A) &= P^n\left(\frac{1}{m}, \bigcup_{q \in \mathbb{N}} \left[\frac{1}{m+1+q}, \frac{1}{m+q} \right]\right) \\ &= \sum_{q \in \mathbb{N}} P^n\left(\frac{1}{m}, \left[\frac{1}{m+1+q}, \frac{1}{m+q} \right]\right) \\ &= \sum_{q \in \mathbb{N} \setminus \{n-1\}} P^n\left(\frac{1}{m}, \left[\frac{1}{m+1+q}, \frac{1}{m+q} \right]\right) + P^n\left(\frac{1}{m}, \left[\frac{1}{m}, \frac{1}{m+n} \right]\right) \end{aligned}$$

Un noyau de transition est toujours ≥ 0

$$\text{donc } P^n(x, A) \geq P^n\left(x, \frac{1}{m+n}\right) = \prod_{i=0}^{n-1} \left(1 - \left(\frac{1}{m+i}\right)^2\right)$$

$$\text{On a } m \geq 2 \text{ donc } \forall i \in \mathbb{N}, \quad 1 - \left(\frac{1}{m+i}\right)^2 \geq 1 - \left(\frac{1}{i+2}\right)^2$$

$$\text{Il vient } P^n(x, A) \geq \prod_{i=0}^{n-1} 1 - \left(\frac{1}{i+2}\right)^2$$

$$\geq \prod_{i=0}^{n-1} \frac{(i+2)^2 - 1}{(i+2)^2}$$

$$\geq \frac{n+1}{n+2} \frac{a^2 - 1}{a^2}$$

$$\geq \frac{n+1}{2} \frac{(a-1)(a+1)}{a^2}$$

On pose $a = i+2$

$$\text{Soit } P^n(x, A) \geq \frac{1 \times 3}{2^2} \times \frac{2 \times 4}{3^2} \times \cdots \times \frac{n \times (n+2)}{(n+1)^2}$$

$$\geq \frac{1}{2} \times \frac{n+2}{n+1}$$

Tout se simplifie.

$$\text{D'où } \lim_{n \rightarrow +\infty} P^n(x, A) \geq \lim_{n \rightarrow \infty} \left(\frac{1}{2} \times \frac{n+2}{n+1} \right)$$

$$\text{i.e. } \lim_{n \rightarrow +\infty} P^n(x, A) \geq \frac{1}{2}$$

Donc $P^n(x, A)$ ne converge pas vers $\pi(A)$ quand $n \rightarrow \infty$

HW1_DECHARRIN

October 25, 2023

de Charrin Théotime - TP1

```
[1]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sn
import sklearn
import math
import scipy.stats as stats
from sklearn.preprocessing import Normalizer
from sklearn.model_selection import train_test_split
```

1 Exercice 1 - Implémentation des algorithmes de Box-Müller et Marsaglia-Bray

```
[2]: def box_muller(n_sample):
    U=np.random.rand(n_sample,2)
    u=U[:,0]
    v=U[:,1]
    theta=2*np.pi*v
    r=np.sqrt(- 2*np.log(1-u))
    x=r*np.cos(theta)
    y=r*np.sin(theta)
    return x,y
```

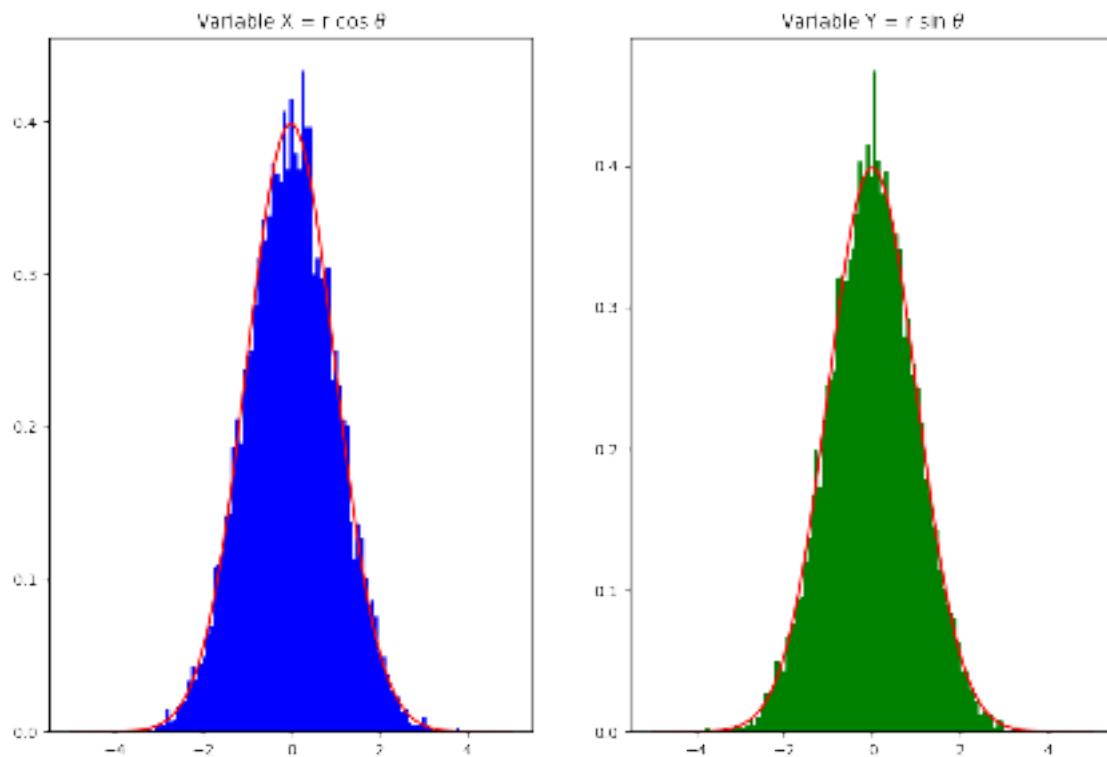
```
[9]: n=10000
x,y=box_muller(n)
ax1=plt.subplot(121)
ax2=plt.subplot(122)
ax1.hist(x,bins=int(n/100),density=True,color='b')
ax2.hist(y,bins=int(n/100),density=True,color='g')
ax1.set_title(r'Variable X = r cos $\theta$')
ax2.set_title(r'Variable Y = r sin $\theta$')

mu = 0
variance = 1
sigma = math.sqrt(variance)
```

```

x = np.linspace(mu - 5*sigma, mu + 5*sigma, 100)
ax1.plot(x, stats.norm.pdf(x, mu, sigma), 'r')
ax2.plot(x, stats.norm.pdf(x, mu, sigma), 'r')
plt.show()

```



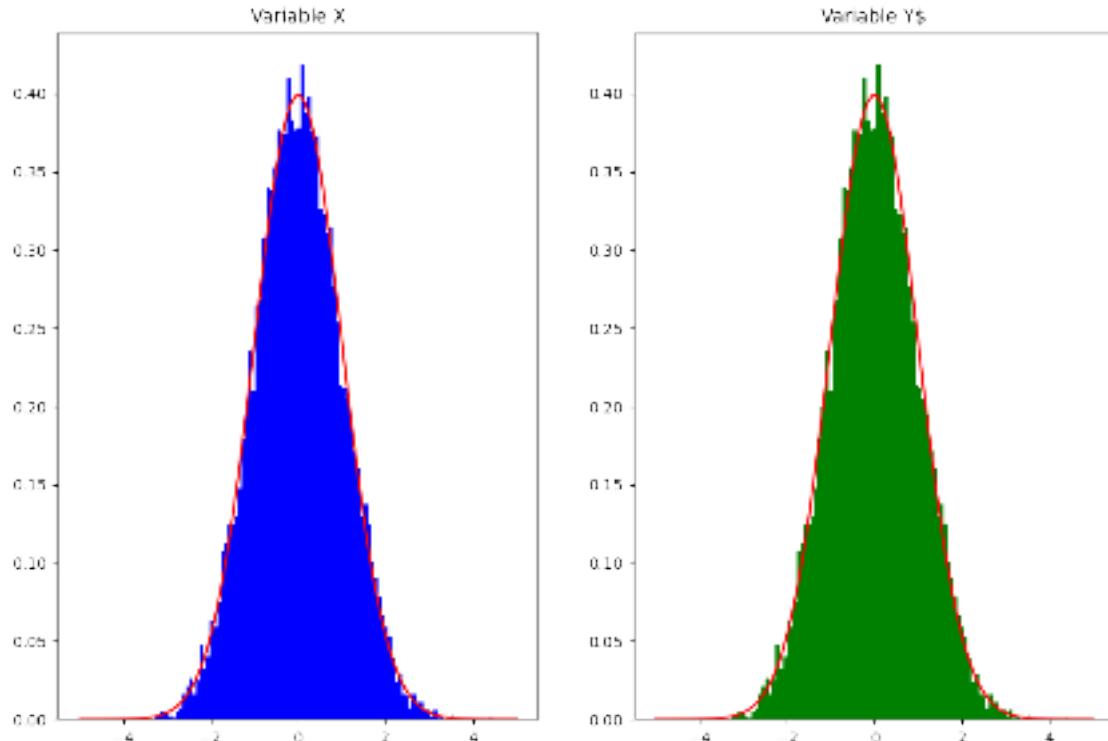
```

[7]: def marsaglia_bray(n_sample):
    X=np.zeros(n_sample)
    Y=X
    for i in range(n_sample):
        V_1=np.random.uniform(-1,1)
        V_2=np.random.uniform(-1,1)
        while V_1**2+V_2**2 > 1:
            U_1=np.random.uniform(0,1)
            U_2=np.random.uniform(0,1)
            V_1=2*U_1-1
            V_2=2*U_2-1
        S=np.sqrt(-2*np.log(V_1**2+V_2**2))
        X[i]=S*(V_1/np.sqrt(V_1**2+V_2**2))
        Y[i]=S*(V_2/np.sqrt(V_1**2+V_2**2))
    return X,Y

```

```
[10]: n=10000
x,y=marsaglia_bray(n)
ax1=plt.subplot(121)
ax2=plt.subplot(122)
ax1.hist(x,bins=int(n/100),density=True,color='b')
ax2.hist(y,bins=int(n/100),density=True,color='g')
ax1.set_title(r'Variable X')
ax2.set_title(r'Variable Y$')

mu = 0
variance = 1
sigma = math.sqrt(variance)
x = np.linspace(mu - 5*sigma, mu + 5*sigma, 100)
ax1.plot(x, stats.norm.pdf(x, mu, sigma), 'r')
ax2.plot(x, stats.norm.pdf(x, mu, sigma), 'r')
plt.show()
```



2 Exercice 3 - Stochastic Gradient learning in gradient descent

```
[11]: plt.rcParams['figure.figsize'] = [12, 8]
plt.rcParams['figure.dpi'] = 100
```

2.0.1 Q1 - Describe the stochastic gradient descent algorithm for minimizing the empirical risk and implement it

La descente de gradient stochastique est une approximation de la descente de gradient classique, où l'on essaie de minimiser la fonction de risque. La seule différence est qu'au lieu de calculer le gradient "réel", on l'approxime grâce aux échantillons observés.

On va supposer :

$$\exists J \text{ différentiable}, \nabla J(\theta) = \mathbb{E}(g(\theta, X)), X \text{ de loi } \mathbb{P}_X \text{ connue et suivant } X^*$$

Notre étape de descente de gradient classique pour optimiser le paramètre θ donne ceci :

$$\theta^{k+1} = \theta^k - \eta_{k+1} \nabla J(\theta^k) \quad (1)$$

$$= \theta^k - \eta_{k+1} \mathbb{E}(g(\theta, X)) \quad (2)$$

Avec η_k le Learning Rate à l'étape k. En descente de gradient stochastique, on va poser :

$$\mathbb{E}(g(\theta, X)) \approx \frac{1}{n} \sum_n g(\theta, X_k), X_k \text{ n copies iid de X}$$

Ainsi, si l'on fait des pas suffisamment petits, on peut poser :

$$\theta^{k+1} = \theta^k - \frac{c}{n} \sum_n g(\theta, X_{k+1})$$

Ici, on veut minimiser le risque sur l'ensemble des w, dans l'implémentation stochastique on introduit donc le risque empirique que l'on veut minimiser :

$$\min_w R_n(w) = \min_w \frac{1}{n} \sum_n (y_i - w^t x_i)^2$$

Le pseudo-algorithme est donc le suivant : > - On part d'un vecteur aléatoire $w_0 \in \mathbb{R}^d$ > - pour $k = (0, 1, \dots, \text{fin} = n_{\text{iter}})$: > > - on choisit un learning rate $(\epsilon_k)_{k \geq 0} > 0$ et un $i \in \mathbb{N}$ aléatoire (MC) > > - On calcule $\nabla_w j(w^k, z_i)$ > > -

$$w_{k+1} = w_k - \epsilon_k \nabla R_n(w_k) \quad (3)$$

$$= w_k - \epsilon_k \mathbb{E}(\nabla_w j(w_k, z_i)) \quad (4)$$

$$= w_k - \epsilon_k \nabla_w (y_i - w_k^t x_i)^2 \quad (5)$$

$$= w_k + 2\epsilon_k (y_i - w_k^t x_i) x_i \quad (6)$$

En pratique, on prendra un $(\epsilon_k)_{k \geq 0}$ tel que $\lim_{+\infty} \epsilon_k = 0$ mais pas trop vite, i.e. $\sum_{+\infty} \epsilon_k = +\infty$. On prendra souvent

$$\epsilon_k = \frac{1}{k^\alpha}, \alpha \in [0.1, 1]$$

```
[12]: def stochastic_gd(x,y, w_0,alpha, niter):
    w_old=w_0
    k=0
    for k in range(1,niter+1):
        i=np.random.randint(0, x.shape[0])
        w_new=w_old+ (2./(k**alpha)) * (y[i] - x[i,:].dot(w_old))*x[i,:]
        w_old=w_new
        #On normalise le vecteur normal
    return w_new/np.sqrt(w_new.dot(w_new))
```

```
[13]: def sample(n,w):
    x=np.random.rand(n,2)
    a=np.dot(x,w)
    a[a>0]=1
    a[a<0]=-1
    return x, a

w=np.array([1,-1])
#On normalise pour comparer les distances
w=w/np.sqrt(w.dot(w))
n=10000
x,y=sample(n,w)
print(x.shape)
```

(10000, 2)

```
[14]: alpha=0.8
w_est=stochastic_gd(x,y,w,alpha,10000)
print(w_est)
y_est=x.dot(w_est)
y_est[y_est>0]=1
y_est[y_est<0]=-1
dist=np.linalg.norm(w_est-w)
print(f"La distance entre les deux vecteurs est de {dist}")
```

[0.70561589 -0.70859453]

La distance entre les deux vecteurs est de 0.0021062194486795873

Le vecteur estimé est toujours très proche de w^* , mais jamais égal.

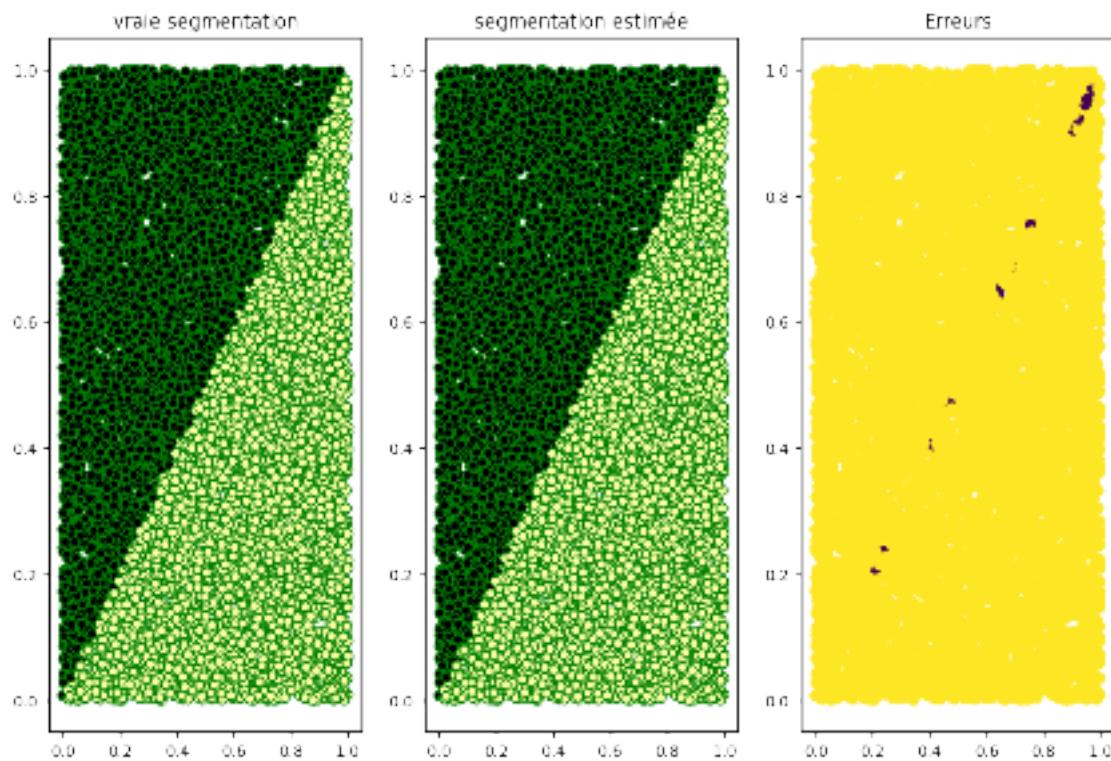
```
[16]: ax1=plt.subplot(131)
ax1.scatter(x[:,0],x[:,1],c=y,cmap='inferno',facecolors='none',edgecolors='g')
ax1.set_title("vraie segmentation")
ax2=plt.subplot(132)
ax2.scatter(x[:,0],x[:,
    ↵,1],c=y_est,cmap='inferno',facecolors='none',edgecolors='g')
ax2.set_title("segmentation estimée")
y_diff=y_est-y
```

```

ax3=plt.subplot(133)
ax3.scatter(x[:,0],x[:,1],c=y_diff)
ax3.set_title("Erreurs")

```

[16]: `Text(0.5, 1.0, 'Erreurs')`



[17]:

```

n=10000
noise=np.random.normal(loc=0,scale=0.125,size=(n,2))
#Comment estimer l'écart-type de la normale? On veut que 95% du bruit soit
#entre -0.25 et 0.25 environ (25% de la variance)
#donc on prend loc=0.125
w_0=np.array([4,-1])
w_0=w_0/np.sqrt(w_0.dot(w_0))
x,y=sample(n,w_0)
x_noised=x+noise
w_est=stochastic_gd(x,y,w_0,alpha,10000)
y_est=x.dot(w_est)
y_est[y_est>0]=1
y_est[y_est<0]=-1
w_noised=stochastic_gd(x_noised,y,w_0,alpha,10000)
y_noised=x.dot(w_noised)
y_noised[y_noised>0]=1
y_noised[y_noised<0]=-1

```

```

print(f"La valeur estimée sans bruit est : {w_est}\n Alors que l'estimation avec bruit est (learning rate alpha de {alpha} ): {w_noised} ) ")
dist=np.linalg.norm(w_est-w_noised)
print(f"La distance entre les deux vecteurs (bruité versus non bruité) est de {dist}")
dist_t=np.linalg.norm(w_noised-w_0)
print(f"La distance entre les deux vecteurs (bruité versus vrai) est de {dist_t}")

```

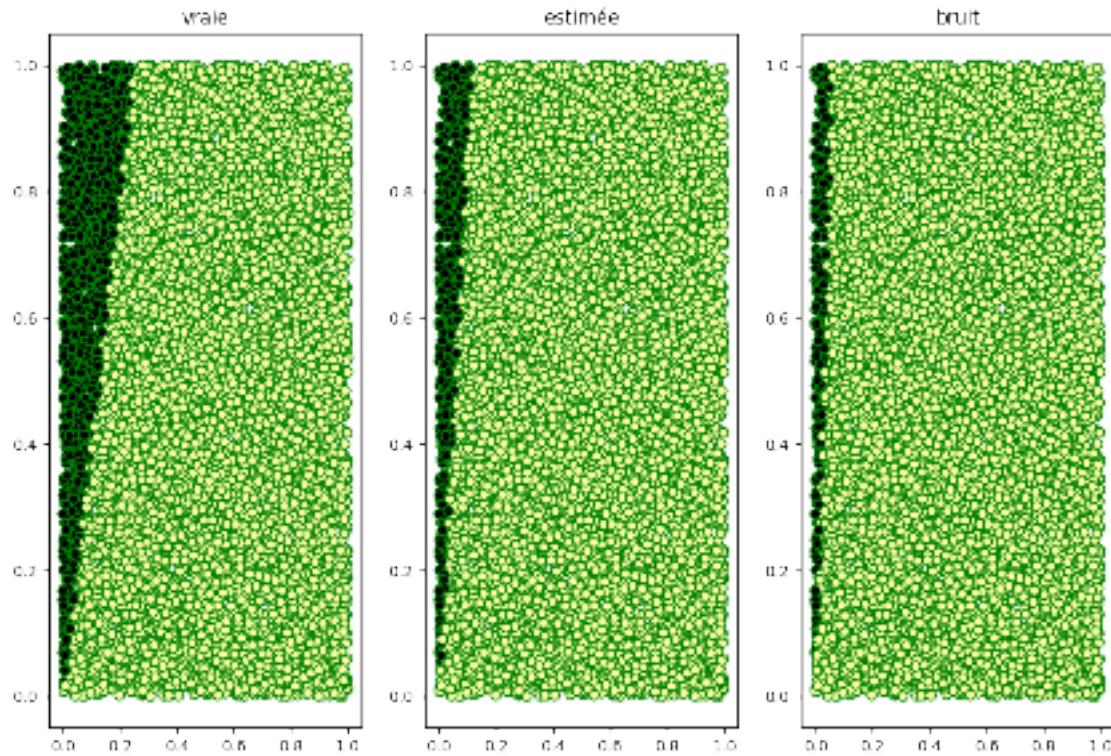
La valeur estimée sans bruit est : [0.99344162 -0.11434047]
Alors que l'estimation avec bruit est (learning rate alpha de 0.8): [0.99831147 -0.05808796])
La distance entre les deux vecteurs (bruité versus non bruité) est de 0.05646291037013009
La distance entre les deux vecteurs (bruité versus vrai) est de 0.18658625586090044

```

[18]: ax3=plt.subplot(131)
ax3.scatter(x[:,0],x[:,1],c=y,cmap='inferno',facecolors='none',edgecolors='g')
ax3.set_title("vraie")
ax1=plt.subplot(132)
ax1.scatter(x[:,0],x[:,1],c=y_est,cmap='inferno',facecolors='none',edgecolors='g')
ax1.set_title("estimée")
ax2=plt.subplot(133)
ax2.scatter(x[:,0],x[:,1],c=y_noised,cmap='inferno',facecolors='none',edgecolors='g')
ax2.set_title("bruit")

```

[18]: Text(0.5, 1.0, 'bruit')



```
[19]: data=pd.read_csv('./data/breast-cancer-wisconsin.data', delimiter=",",  
                     index_col=False, header=None).replace("?", np.nan).dropna()  
print(data.shape)  
column_names=[["id"]+[ "x%d" %i for i in range(1,10)]+["class"]]  
data.columns=column_names  
data.index=data.iloc[:,0]  
data=data.iloc[:,1:]  
data
```

(683, 11)

	x1	x2	x3	x4	x5	x6	x7	x8	x9	class
id										
1000025	5	1	1	1	2	1	3	1	1	2
1002945	5	4	4	5	7	10	3	2	1	2
1015425	3	1	1	1	2	2	3	1	1	2
1016277	6	8	8	1	3	4	3	7	1	2
1017023	4	1	1	3	2	1	3	1	1	2
...
776715	3	1	1	1	3	2	1	1	1	2
841769	2	1	1	1	2	1	1	1	1	2
888820	5	10	10	3	7	3	8	10	2	4
897471	4	8	6	4	3	4	10	6	1	4

```
897471      4     8     8     5     4     5    10     4     1      4
```

```
[683 rows x 10 columns]
```

```
[20]: x=data.iloc[:,0:-1]
y=np.where(data["class"]==2,1,-1)
#y=1 veut dire cancer bénin
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.25,random_state=0)
scaler = Normalizer().fit(x_train)
normalized_x_train= scaler.transform(x_train)
normalized_x_test = scaler.transform(x_test)
normalized_x_train.shape
```

```
[20]: (512, 9)
```

```
[21]: w_0=np.random.rand(normalized_x_train.shape[1])
w_0=w_0/(np.sqrt(w_0.dot(w_0)))
w_est=stochastic_gd(normalized_x_train,y_train,w_0,alpha=0.8,niter=100000)
```

```
### Confusion matrix
y_pred=np.where(normalized_x_test.dot(w_est)>0,1,-1)
accuracy_score=sklearn.metrics.accuracy_score(y_test,y_pred,normalize=True)
cm=sklearn.metrics.confusion_matrix(y_test, y_pred, normalize='true')
df_cm = pd.DataFrame(cm, columns = ["benign", "malignant"],
                      index = ["Pred_ben", "Pred_mal"])
plt.figure(figsize = (10,7))
sn.heatmap(df_cm, annot=True)
plt.title(f"Matrice de confusion, le score de précision est de {accuracy_score*100:.1f}%")
```

```
[21]: Text(0.5, 1.0, 'Matrice de confusion, le score de précision est de 87.1%)
```

Matrice de confusion, le score de précision est de 87.1%

