Projet estimation non parametrique

N'DOYE EL Hadrami et RAMDÉ Ismaïl

2021-12-08

A. Données circulaire

Loi uniforme sur le cercle

Une loi uniforme circulaire est une loi de probabilité sur le cercle unité dont la densité f est uniforme pour tous les angles $\theta \in [0, 2\pi]$,

$$f(z) = \frac{1}{2\pi},$$

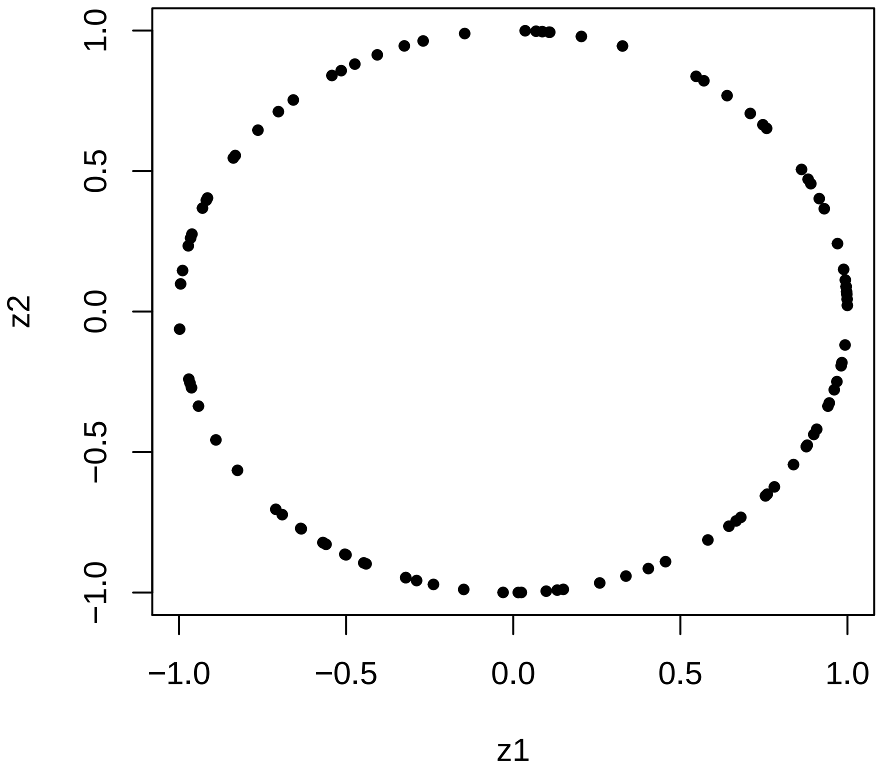
z est un point du cercle unitaire, z est donc un point du plan de coordonnées $(\cos\theta, \sin\theta)$ pour $\theta \in [0, 2\pi]$.

Question (A1)

On rapelle que si U suit la loi uniforme sur l'intervalle $[0, 1]$ alors le couple aléatoire $(\cos(2\pi U), \sin 2\pi U)$ est de loi uniforme sur le cercle unitaire.

(A1) Utilisons ce résultat pour simuler n points $z_1 \dots z_n$ uniformément distribués sur le cercle unitaire.

```
z = function(n_points) {
    u = runif(n_points, 0, 1)
    x = cos(2 * pi * u)
    y = sin(2 * pi * u)
    z = matrix(c(x, y), nrow = n_points, ncol = 2)
}
n = 100
zi = z(n)
plot(
    zi[, 1],
    zi[, 2],
    xlab = "z1",
    ylab = "z2",
    pch = 20,
    xlim = c(-1,1)
)
```



On remarque que les trajectoires suivies par les différents points définissent un cercle. Ces points sont tous sur le cercle unitaire.

Loi de Cauchy sur le cercle

Soit ϕ fixé dans [0,1]. La loi de Cauchy sur le cercle, notée $C^*(\varphi)$

Question (A2)

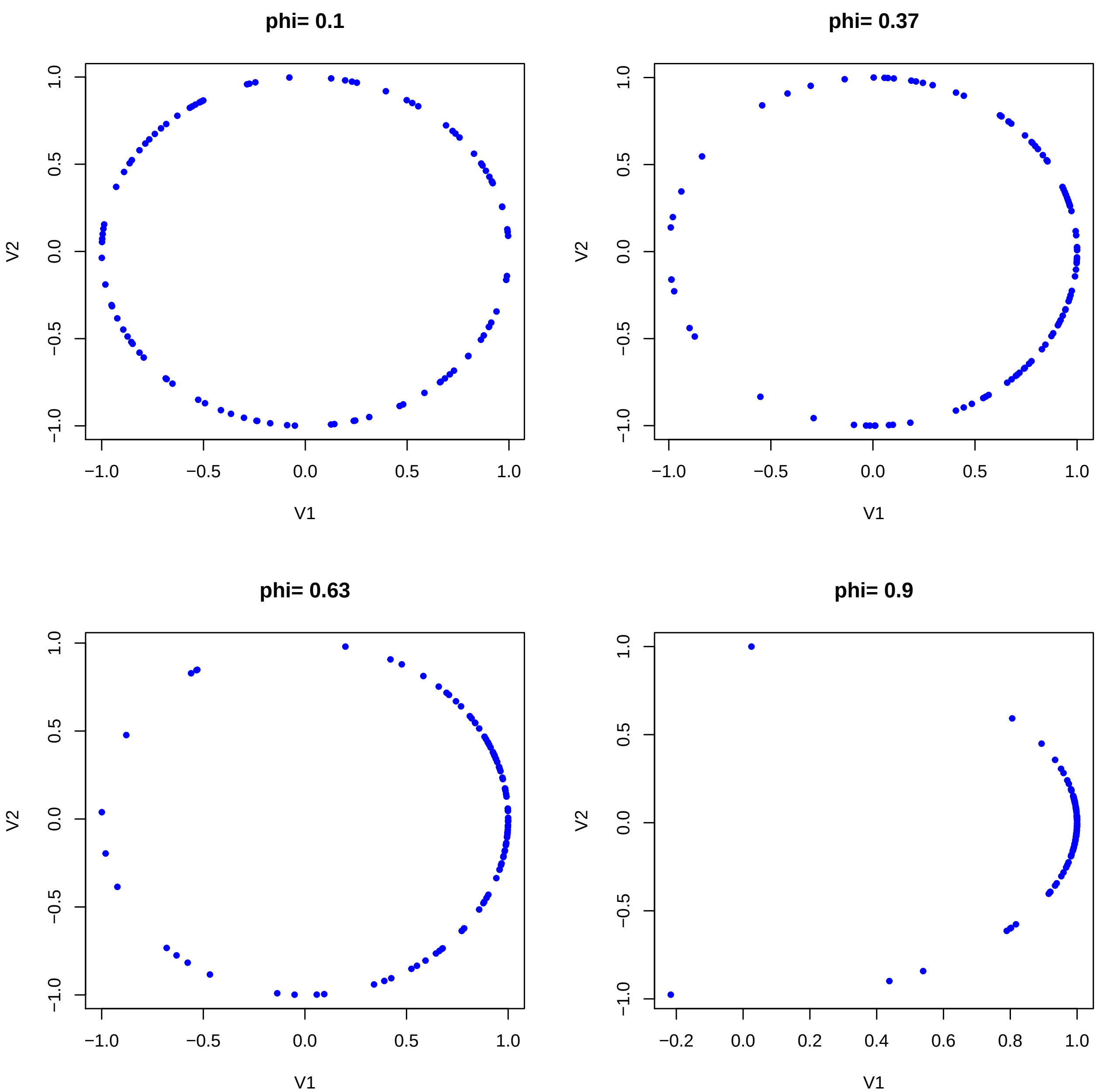
On admet le résultat suivant : si U suit la loi uniforme sur l'intervalle $[0, 1]$ alors le couple $W = (V1, V2)$ avec $V1 = \frac{2\varphi+\cos(2\pi U)+\varphi^2\cos(2\pi U)}{1+\varphi^2+2\varphi\cos(2\pi U)}$, $V2 = \frac{\sin(2\pi U)+\varphi^2\sin(2\pi U))}{1+\varphi^2+2\varphi\cos(2\pi U)}$ suit la loi $C^*(\varphi)$.

A2 En utilisant ce résultat et en fixant une valeur de φ, simulons n points $w_1 \dots w_n$ issus de la loi $C^*(\varphi)$.

```
loi_cauchy = function(n_points, phi) {
    u = runif(n_points)
    cos2piu = cos(2 * pi * u)
    sin2piu = sin(2 * pi * u)
    # Pour la loi V1
    num_V1 = 2 * phi +  (cos2piu) + (phi ^ 2 * cos2piu)
    denom_V1 = 1 + (phi ^ 2) + (2 * phi * cos2piu)
    V1 = num_V1 / denom_V1
    num_V1 = sin2piu - ((phi ^ 2) * sin2piu)
    num_V2 = 1 + (phi ^ 2) + (2 * phi * cos2piu)
    V2 = num_V1 / num_V2
    W = matrix(c(V1, V2), nrow = n_points, ncol = 2)

    return(W)
}
phi = seq(0.1,0.9, length.out = 4)
par(mfrow = c(2,2))
```

```r
for (i in phi) {
    W = loi_cauchy(n, i)
    plot(
    W[, 1],
    W[, 2],
    col = "blue",
    xlab = "V1",
    ylab = "V2",
    pch = 20
    )
    title(paste("phi=", round(i,2)))
}
```



On obtient des points qui sont situés sur le cercle unitaire. En faisant varier φ (0.1, 0.37, O.63, 0.9), on observe une différence dans la distribution des 4 cas de figure. Pour $\varphi = 0.1$, on observe des points qui se répartissent bien sur la quasi-totalité du cercle unitaire. Au fur à mesure que φ augmente les points ont tendance à se regrouper sur une partie du cercle. Pour $\varphi = 0.9$ presque tous les points se regroupent sur un seul côté du cercle. Le paramètre φ a donc un effet sur la répartition ou la distribution des points.

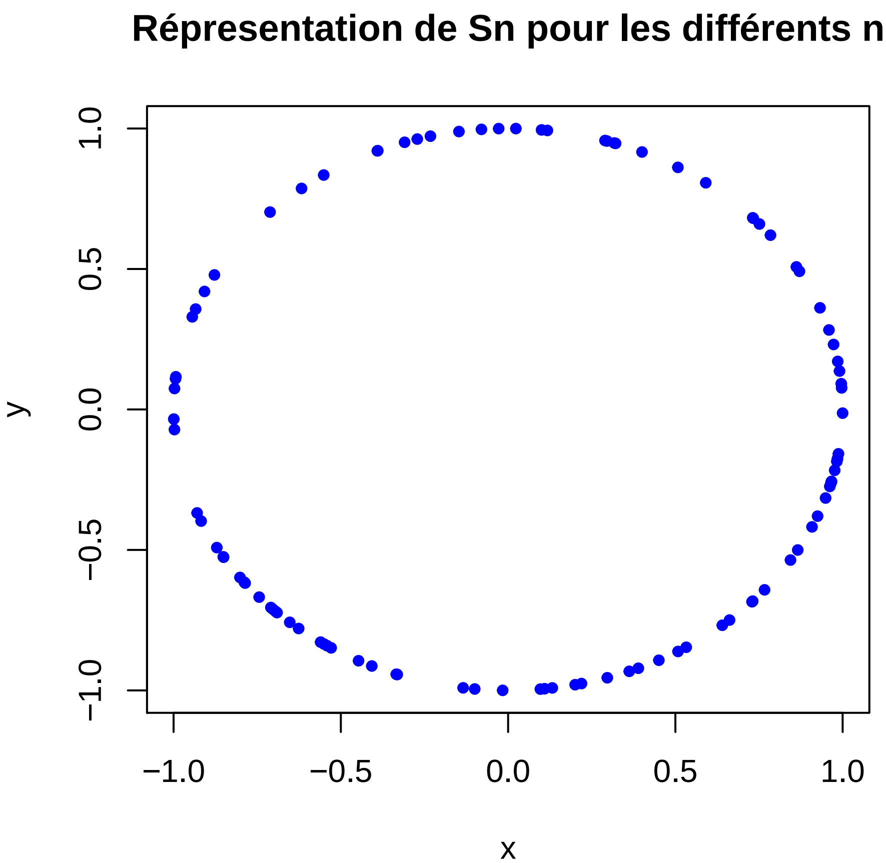
Simulation des données

Question (A3)

Simulons cet ensemble et répresenton S_n pour des différentes valeurs de n.

```
S_n = function(n_points) {
    Sn = matrix(0, nrow = n_points, ncol = 2)
    zi = z(n_points)
    P0 = zi[1, ]
    eps = loi_cauchy(n_points,phi=0.5)
    eps1 = W[1, ]
    a = P0[1]
    b = P0[2]
    c = eps1[1]
    d = eps1[2]
    # Calcul de P1
    Sn[1, 1] = (a * c) - (d * b)
    Sn[1, 2] = (a * d) + (b * c)
    for (i in 2:n_points) {
        a_sn = Sn[i - 1, 1]
        b_sn = Sn[i - 1, 2]
        c_sn = eps[i, 1]
        d_sn = eps[i, 2]
        Sn[i, 1] = (a_sn * c_sn) - (d_sn * b_sn)
        Sn[i, 2] = (a_sn * d_sn) + (b_sn * c_sn)
    }
    return(Sn)
}
```

```
n = 100
Sn = S_n(n)
plot(
    Sn[, 1],
    Sn[, 2],
    xlab = "x",
    ylab = "y",
    col = "blue",
    pch = 20,
)
title("Répresentation de Sn pour les différents n")
```

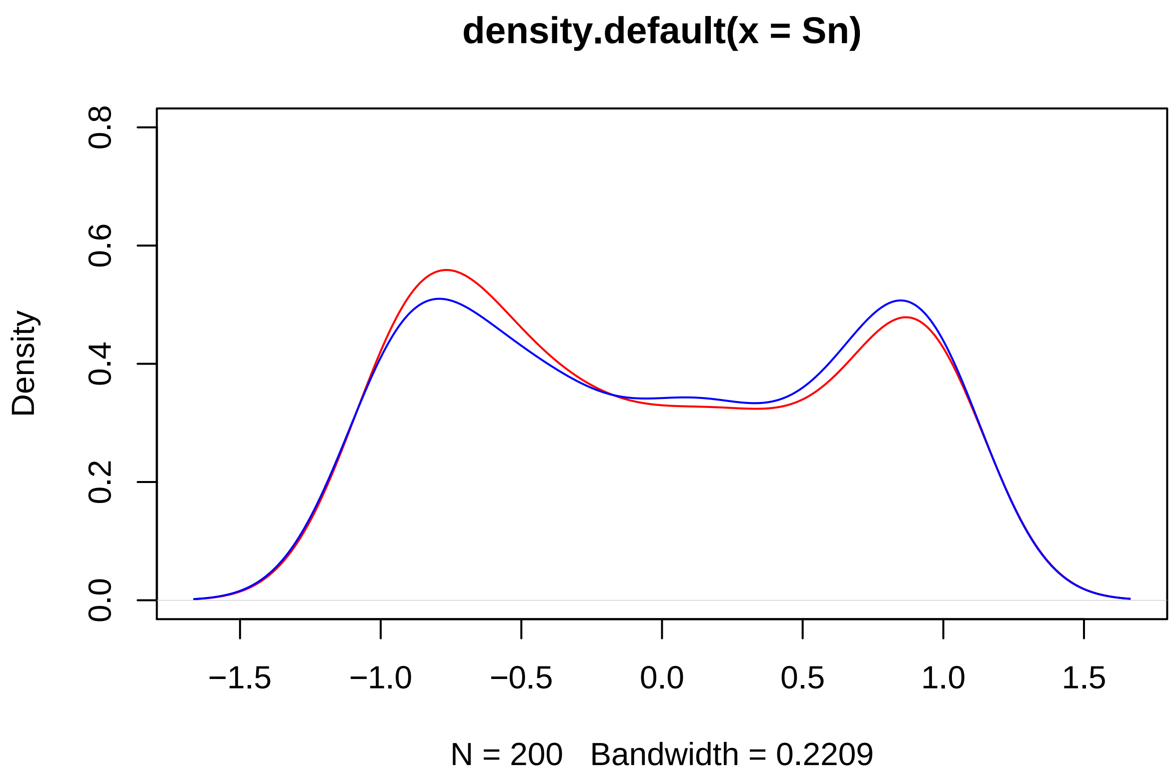


On constate que les points sont sur le cercle unitaire.

Question (A4)

Vérifions, par des illustrations graphiques, que S_n s'approche d'un ensemble déterministe M que l'on déterminera.

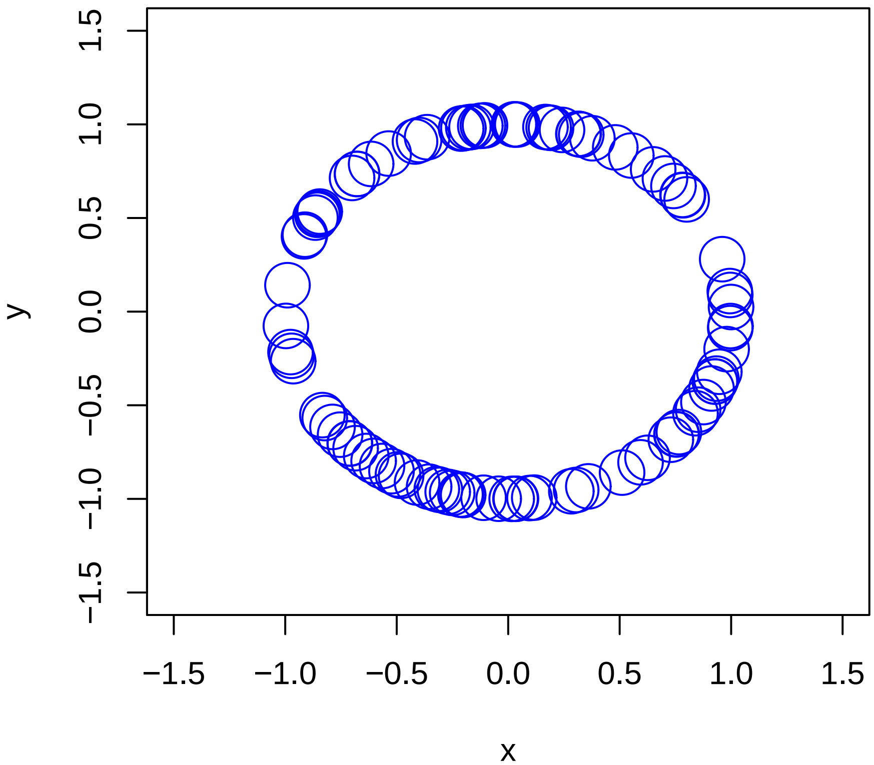
```
plot(density(Sn), col = "red", ylim = c(0, 0.8))
lines(density(zi), col = "blue")
```



Question (A5)

Répresentons pour $\epsilon < 0.2$ et pour des différents valeurs de n l'ensemble $\mathbb{U}_{P \in S_n} B(P, \epsilon)$

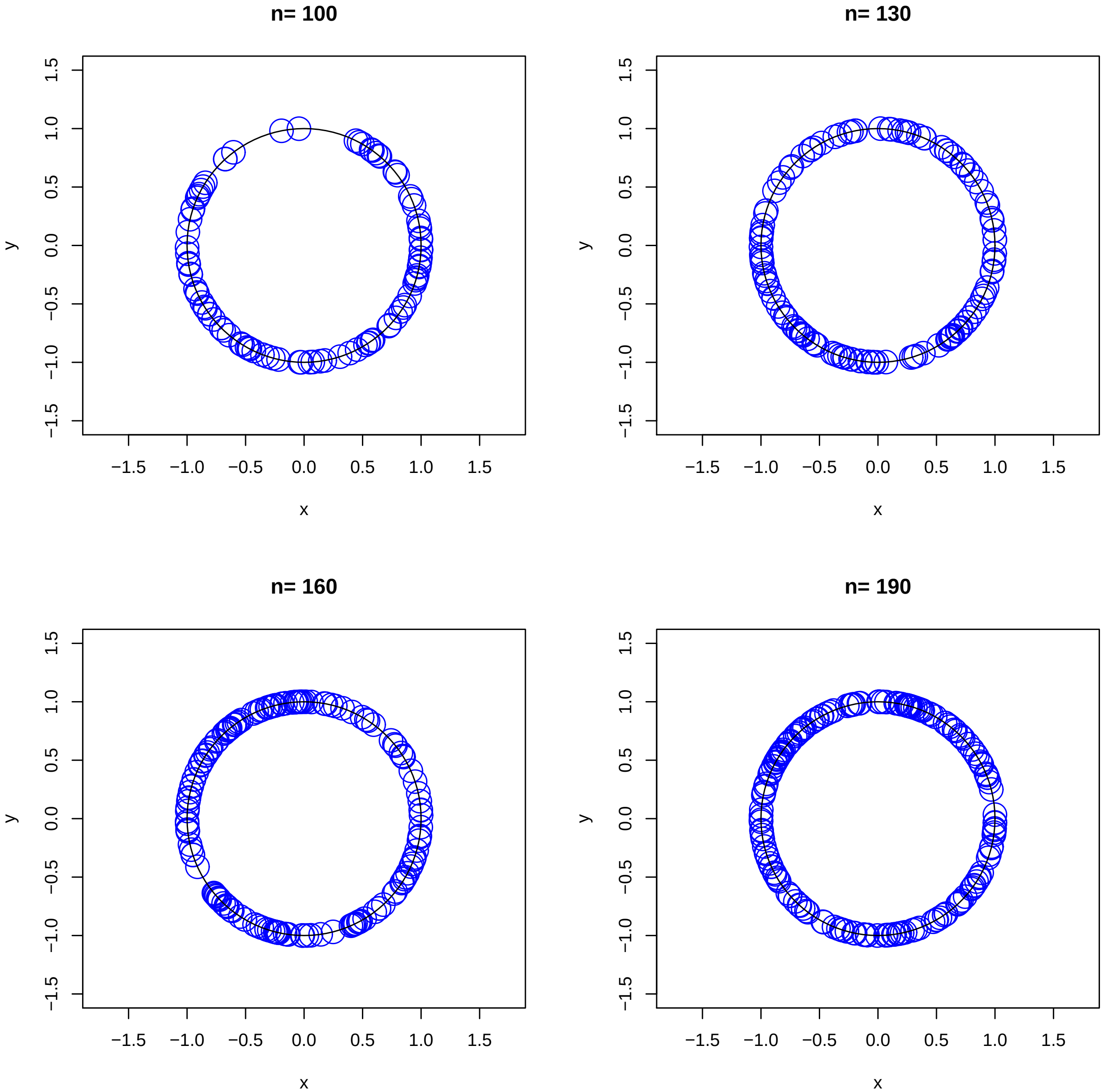
```
library(plotrix)
epsilon = 0.1
x = seq(-1.5, 1.5, length.out = n)
y = x
plot(x, y, type = "n")
Sn = S_n(n)
for (i in 1:n) {
    draw.circle(Sn[i, 1], Sn[i, 2], epsilon, border = "blue")
}
```



Question (A6)

Voyons à partir de quelle valeur de n, $\mathbb{U}_{P \in S_n} B(P, \epsilon)$ recouvre $\mathbb{M}$

```
par(mfrow = c(2, 2))
np = seq(100,190,length.out=4)
for (i in np) {
    x = seq(-1.5, 1.5, length.out = i)
    y = x
    plot(
        x,
        y,
        type = "n",
        xlim = c(-1.5, 1.5),
        ylim = c(-1.5, 1.5),
        main = paste("n=", i),
        asp=1
    )
    Sn = S_n(i)
    for (j in 1:i) {
        draw.circle(Sn[j, 1], Sn[j, 2], epsilon, border = "blue")
    }
    draw.circle(0, 0, 1.0)
}
```



Pour n = 100, On remarque $\mathbb{U}_{P\in S_n} B(P,\epsilon)$ ne recouvre pas complètement $\mathbb{M}$. À partir de n = 130 $\mathbb{U}_{P\in S_n} B(P,\epsilon)$ recouvre $\mathbb{M}$.

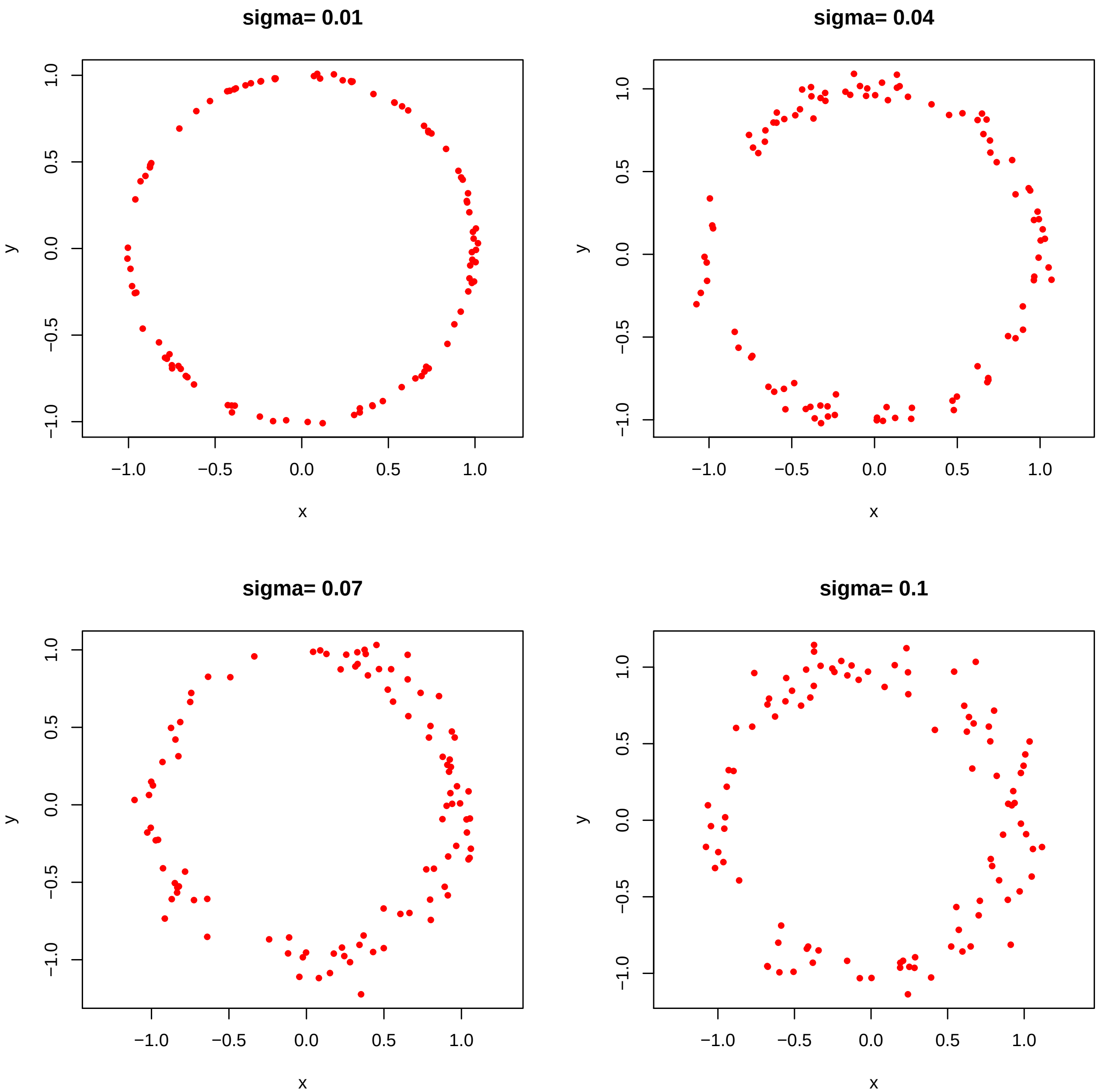
B. Données bruitées

Question (B1) :

Simulons et représentons le nuage des points $X_1 \dots X_n$ pour une valeur de σ qu'on a choisi et pour des différentes valeurs de n (les données $P_1 \dots P_n$ étant celles simulées lors de la question (A3)).

```
library(MASS)
par(mfrow = c(2, 2))
simul_Xi = function(n, sigma,alpha_i) {
    P_i = S_n(n)
    X_i = P_i + (sigma * alpha_i)
```

```r
    return(X_i)
}
sigma = seq(0.01, 0.1, length.out = 4)
alpha_i = mvrnorm(n, mu = c(0, 0), Sigma = diag(2))
for (i in 1:length(sigma)) {
    X_i = simul_Xi(n,sigma[i],alpha_i)
    plot(
        X_i[, 1],
        X_i[, 2],
        pch = 20,
        col = "red",
        xlab = "x",
        ylab = "y",
        main = paste("sigma=", sigma[i]),
        asp = 1
    )
}
```



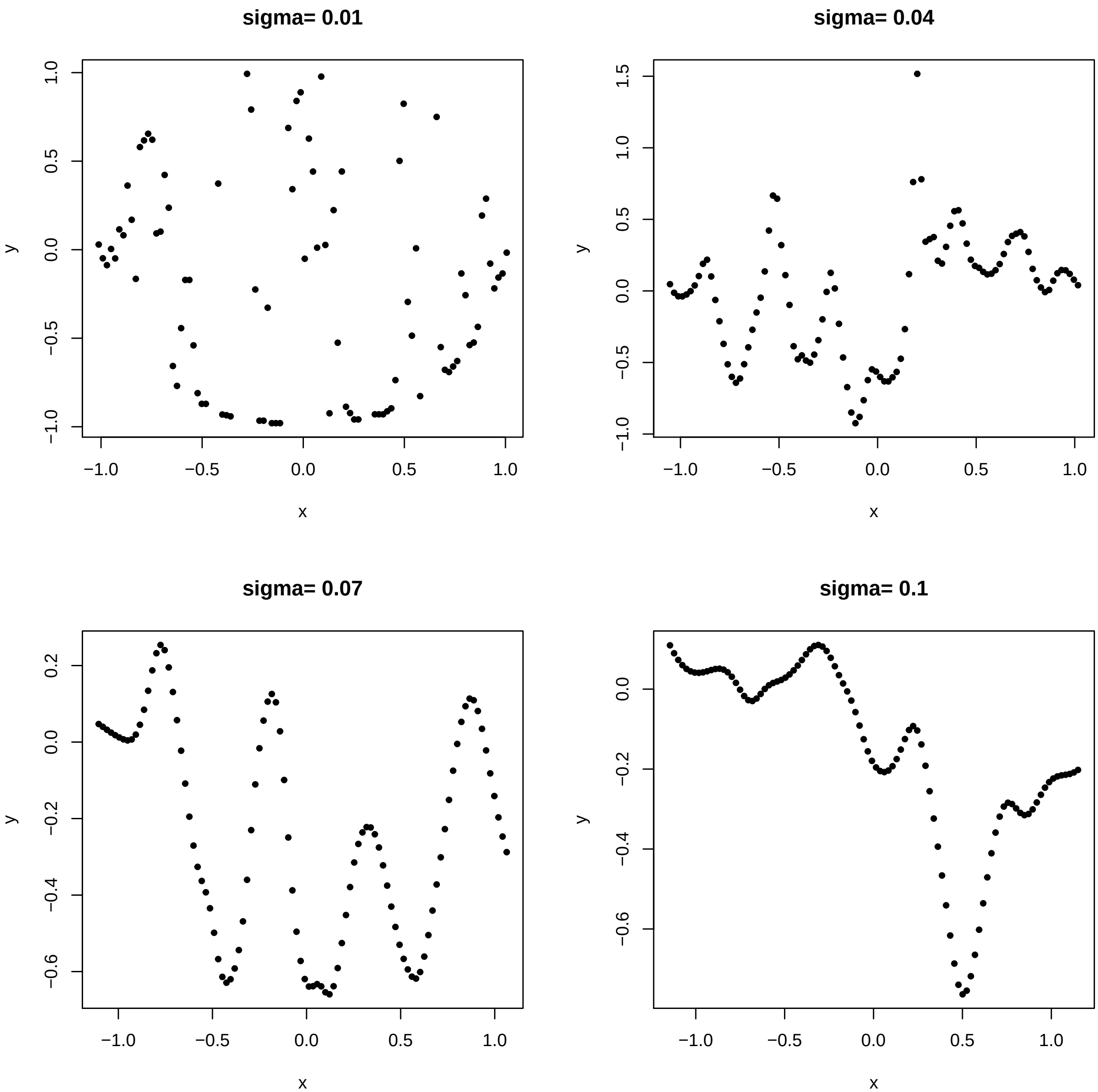
C. Lien avec la régression non paramétrique

Question (C1)

```r
par(mfrow=c(2,2))
library(KernSmooth)
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```r
for(i in 1:length(sigma)){
    Xi = simul_Xi(n,sigma[i],alpha_i)
    plot(locpoly(Xi[,1],Xi[,2],bandwidth=sigma[i],gridsize = n),pch=20,main = paste("sigma=",sigma[i]),
}
```



Question (C2)

Conclure en rapport avec l'ensemble M.