M2 probabilités et statistiques des nouvelles donnés

Régression non paramétriques

Djamila AZZOUZ
RAZAFIMANDIMBY Henimpitahiana

Professeur référent
Mr Christophe Denis

Année académique 2022 - 2023

I Simuler un échantillon selon un modèle de régression donné :

Nous avons $X \in \mathbf{R}^d$ distribué selon une loi uniforme sur l'hypercube $[0,1]^d$, avec chaque $(X^j)_{1\leq j\leq d}$ iid suivant une loi uniforme sur $[0,1]$.

On pose la variable aléatoire suivante :

$$Y = 5 * exp(||X||_2) + \varepsilon$$

où ε est indépendante de X et suit une loi normale centré réduite.

1. Écrivons une fonction prenant en argument n et renvoyant un n-échantillon de même loi que (X, Y).

```
#Question 1: écrire une fonction qui prend en paramètre n
n=100
#Fonction qui nous permet de calculer la norme
X=matrix(data=runif(n*d,0,1),nrow=n,ncol=d)
norme=function(X){
  norme_2=rep(0,n)
  for(i in 1:n){
    norme_2[i]= sqrt(sum(X[i,]^2))
  }
  return(norme_2)
}
#On construit la fonction qui prend en paramètre n
d=1
fon=function(n){
  epsi=rnorm(n,0,1)
  Y=5*exp(norme(X))+epsi
  return(Y)
}
```

2. Dans le cas où d=1, nous allons représenter sur un même graphique un n-échantillon de même loi que (X, Y) ainsi que la fonction de régression f^*.

```
#Question 2:
#projection  de n-échantillon de même loi que (X,Y)
plot(X,fon(n),main="Graphe de n-éch et la fonction de régression")
```

   — Calculons la fonction f^*

   — On sait que $f^* = E(Y|X)$

$$f^* = E(E(Y|X)) = 5 * E(exp(norme_2|X)) = 5 * exp(x)$$

```
f_et=function(x){
  f=5*exp(x)
  return(f)
}
curve((f_et(x)),lwd="2",col="red", add=TRUE)
```

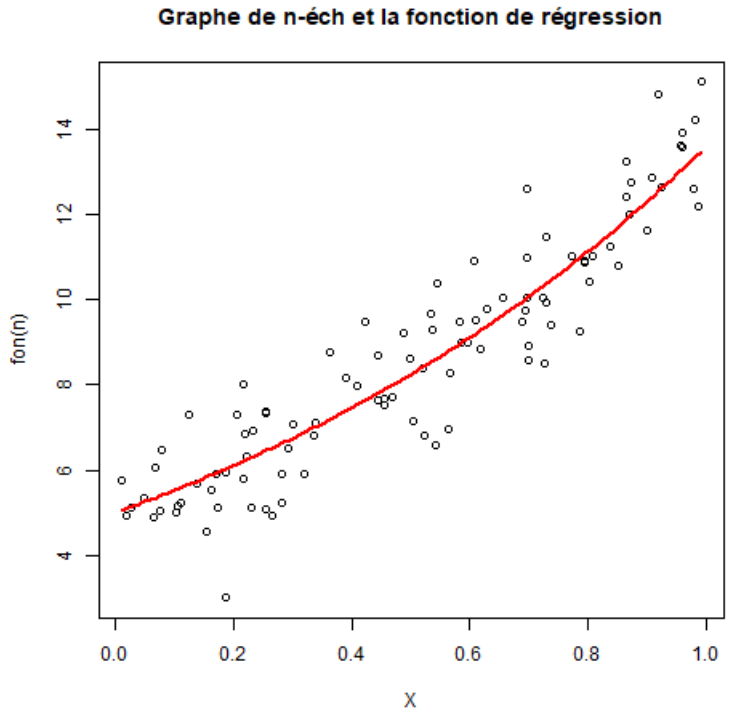


FIGURE 1 – Représentation de n-échantillons et la fonction de régression

II Estimation par k-plus proches voisins cas où $d = 1$:

Dans cette partie, nous allons calculer le prédicteur des k-plus proches voisins $\hat{f}_n$ et nous allons faire varier les paramètres k et n pour savoir s'ils ont un impact sur notre prédicteur et f^*.
Le tableau ci dessus contient les k,taille des échantillons choisis pour réaliser les simulations de notre fonction de régression et son prédicteur.

k choisis	Échantillons choisis
1	100
4	500
20	1000
80	1500

(a) Calculons le risque L_2 du prédicteur optimal f^* défini en cours :
On sait que

$$R(f^*) = E((Y - f^*(x))^2)$$

Nous avons donc :

$$R(f^*) = E((5 * \exp(x) + \varepsilon - 5 * exp(x))^2)$$

Ainsi

$$R(f^*) = E(\varepsilon^2) = Var(\varepsilon) = 1$$

(b) Estimation par variation de k.

```
#Construction d'un nouveau échantillon
n=100
#Fonction qui nous permet de calculer la norme
```

```
X=matrix(data=runif(n,0,1),nrow=n,ncol=1)
Xtest=X
norme=function(Xtest){
  norme_2=rep(0,n)
  for(i in 1:n){
    norme_2[i]= sqrt(sum(Xtest[i,]^2))
  }
  return(norme_2)
}
tes=function(n){
  epsi=rnorm(n,0,1)
  Ytest=5*exp(norme(Xtest))+epsi
  return(Ytest)
}
library(FNN)
par(mfrow=c(2,2))
f_chap=knn.reg(X,Xtest,tes(n),k=1)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="green",main="k=1",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)
#Variation de k
f_chap=knn.reg(X,Xtest,tes(n),k=4)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="green",main="k=4",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)
##############################################"
f_chap=knn.reg(X,Xtest,tes(n),k=20)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="green",main="k=20",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)
########################################
f_chap=knn.reg(X,Xtest,tes(n),k=80)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="green",main="k=80",lwd=3)
```

```
curve((f_et(x)),lwd="2",col="red", add=TRUE)
```

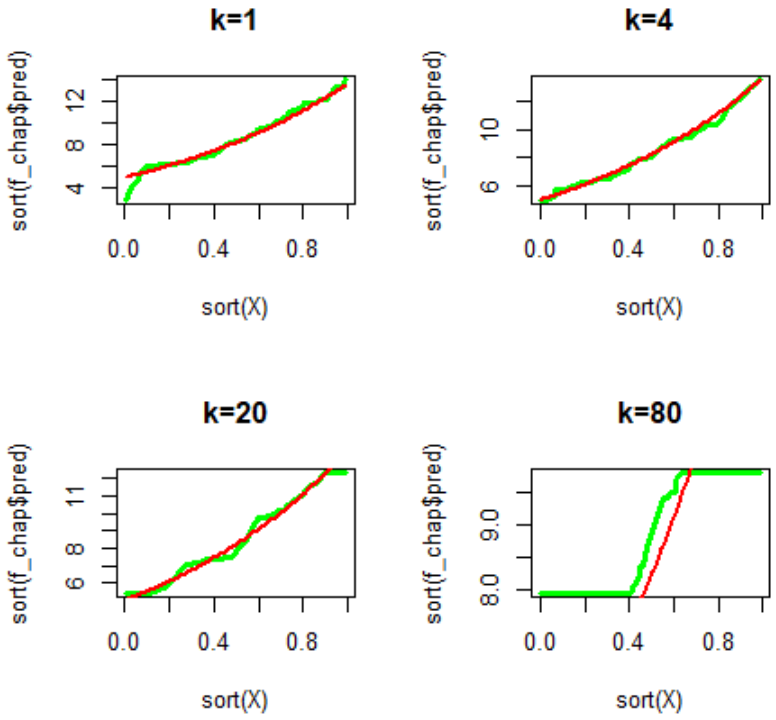


FIGURE 2 – Simulation du prédicteur f^* et $\hat{f}_n$

Observation : On remarque que le prédicteur optimale s'éloigne de plus en plus du prédicteur des k-plus proches voisins quand k devient de plus en plus petit. Nous pouvons donc dire que l'estimateur des k-plus proches voisins dépendent du choix de k.

(c) Estimation par variation de l'échantillon

Nous allons faire quatre simulations avec 4 échantillons différents et observons ce qui se passe :

```
#Variation de l'echantillon
library(FNN)
par(mfrow=c(2,2))
n=100
X=matrix(runif(n,0,1),nrow=n,ncol=1)
Xtest=X
f_chap=knn.reg(X,Xtest,tes(n),k=3)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="blue",main="n=100",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)

#Variation de n
n=500
X=matrix(runif(n,0,1),nrow=n,ncol=1)
```

```
Xtest=X
f_chap=knn.reg(X,Xtest,tes(n),k=3)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="blue",main="n=500",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)
##############################################"
n=1000
X=matrix(runif(n,0,1),nrow=n,ncol=1)
Xtest=X
f_chap=knn.reg(X,Xtest,tes(n),k=3)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="blue",main="n=1000",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)
########################################
n=1500
X=matrix(runif(n,0,1),nrow=n,ncol=1)
Xtest=X
f_chap=knn.reg(X,Xtest,tes(n),k=3)
f_chap$pred
f_et=function(x){
  f=5*exp(x)
  return(f)
}
plot(sort(X),sort(f_chap$pred),type='l',col="blue",main="n=1500",lwd=3)
curve((f_et(x)),lwd="2",col="red", add=TRUE)
```

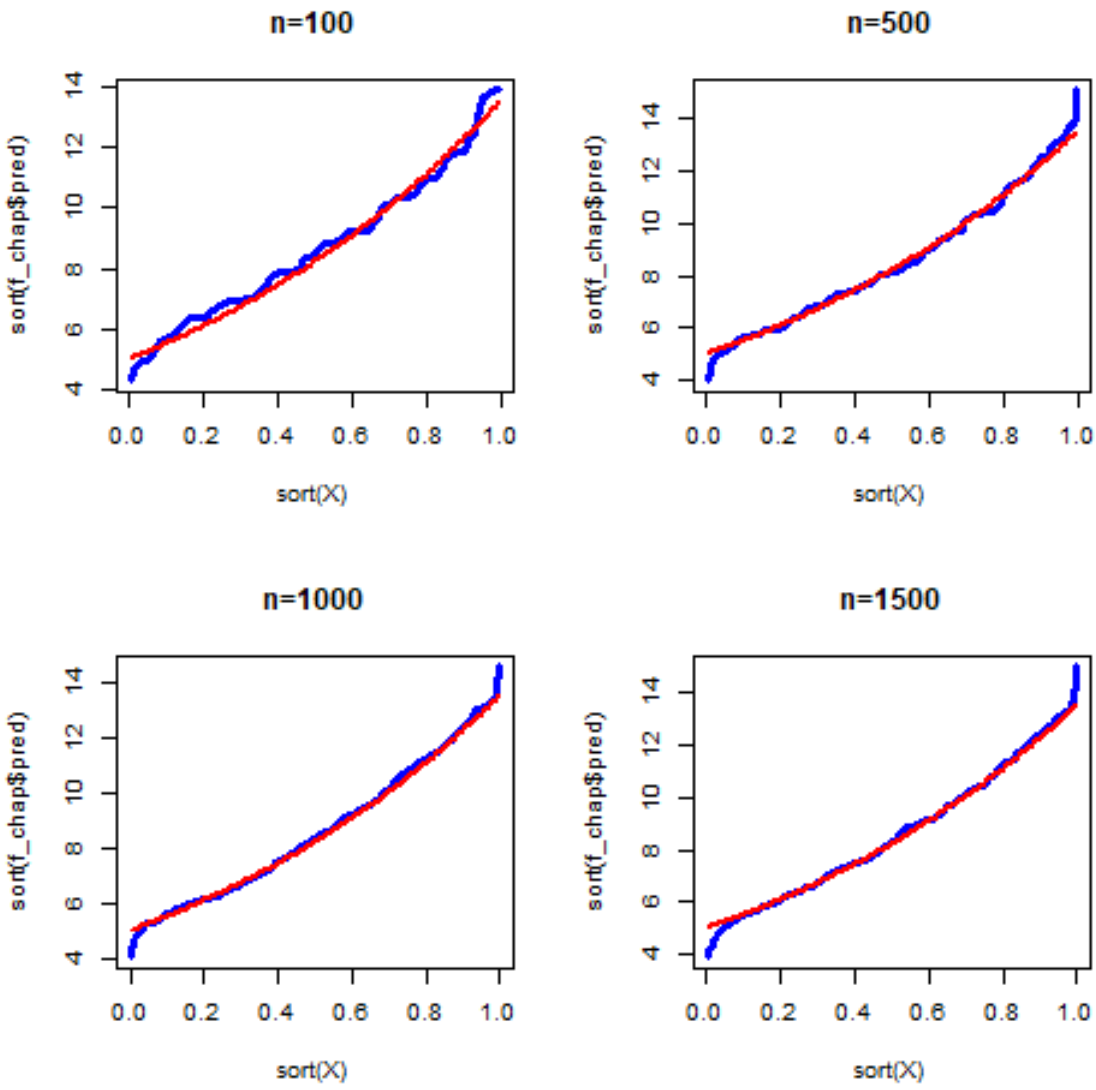


FIGURE 3 – Simulation du prédicteur f^* et $\hat{f}_n$

Observation On remarque que : plus que l'échantillon est grand plus que notre prédicteur optimal est plus proche de notre prédicteur des k-plus proches voisins.

III Évaluation du risque L_2 de $\hat{f}_n$

Dans cette partie, nous allons évaluer en fonction de n le risque L_2 de $\hat{f}_n$ et son excès de risque. Pour cela nous allons réaliser ceci à l'aide de la méthode Monte Carlo en suivant les étapes suivantes :

— Tout d'abord nous allons commencer par simuler D_n et $((X_{n+1}, Y_{n+1}), ...(X_{n+M}, Y_{n+M}))$: Le code obtenu avec R est le suivant :

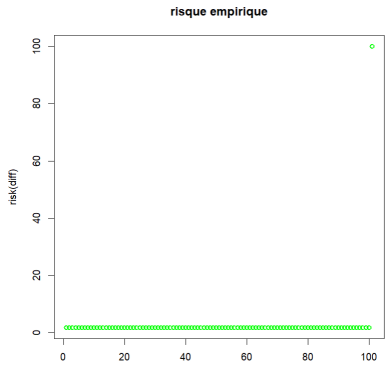
```
#Question 3
#  1/ Simulation des D_n
#Fonction qui nous permet de calculer la norme
n=5
X=matrix(data=runif(n,0,1),nrow= n,ncol=1)
Xtest=X
```

```
norme=function(Xtest){
  norme_2=rep(0,n)
  for(i in 1:n){
    norme_2[i]= sqrt(sum(Xtest[i,]^2))
  }
  return(norme_2)
}
tes=function(n){
  epsi=rnorm(n,0,1)
  Ytest=5*exp(norme(Xtest))+epsi
  return(Ytest)
}
#Simulation de (X_n+1,Y_n+1).....(X_n+M,Y_n+M)
#Fonction qui nous permet de calculer la norme
n=30
M=2
X_1=matrix(data=runif((n+M),0,1),nrow=(n+M),ncol=1)
Xtest1=X_1

norme1=function(Xtest1){
  norme_2=rep(0,(n+M))
  for(i in 1:(n+M)){
    norme_2[i]= sqrt(sum(Xtest1[i,]^2))
  }
  return(norme_2)
}

tes1=function(n,M){
  epsi=rnorm((n+M),0,1)
  Ytest1=5*exp(norme(Xtest1))+epsi
  return(Ytest1)
}
```

— On construit l'estimateur $\hat{f}_n$ des k-ppv et $\hat{f}_n(X_{n+1})$.....$\hat{f}_n(X_{n+M})$:
Le code est le suivant :

```
M=100
X_1=matrix(data=runif((n+M),0,1),nrow=(n+M),ncol=1)
Xtest1=X_1
norme(Xtest1)
Y_1=tes(n+M)
Ytest=Y_1

#on liste  les couples de variable aléatoires (X_{n+1}, Y_{n+1}),....(X_{n+M}, Y_{n+M

Donn= data.frame(don1,don2)

library(FNN)
#pour déterminer \hat{f}_n
```

```
f_chap1=knn.reg(X_1,Xtest1,Y_1, k=3)
f_chap1$pred
# on liste et on range les Y_{n+m} et les \hat{f}_{n+m} pour faciliter le calcul de l

data.frame(sort(Ytest),sort(f_chap1$pred))

#on calcul d'abord la différence entre les deux
diff= sort(Ytest)- sort(f_chap1$pred)
#puis le risque
Nrep= 100
risk= function(diff){
  risque= rep(0:M)
  for (j in 1:Nrep){
    risque[j]= (1/M)* sum(diff^2)
}
  return(risque)
}
risk(diff)
mean(risk(diff))
```




```
  [1]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
  [9]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [17]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [25]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [33]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [41]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [49]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [57]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [65]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [73]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [81]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [89]   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377   1.828377
 [97]   1.828377   1.828377   1.828377   1.828377 100.000000
```