1. **Input le jeu des données**, fichier “AV34\_V3.csv" (package : gdata):

https://www.data.gouv.fr/fr/datasets/transparence-sante-1/

data <- read.csv("AV36\_100.csv", header=T) ; dim(data)

N=dim(data)[1]; M=dim(data)[2]; N; M

### *Installer des packages : qrmtools, plyr, evir :*

library(qrmtools); library(evir); library(plyr)

**2. Nettoyer des données**: supprimer les variables identifiants - colonne 3 & 35 :

et **k-anonymiser des données** : k-anonymiser les 5 variables qui ont les plus probabilité de re-identification - colonne 7,11,12,13,25

Input: des données brutes, k-anonymisation

Output: des données k-anonymisées

function : anonymise(inp,k)

inp <- data[,c(7,11,12,13,25)]

outp <- anonymise(inp,3)

newdata <- cbind(data[,c(1:6)],outp[,1],data[,c(8:10)], outp[,c(2,3,4)],data[,c(14:24)],outp[,5],data[,c(26:36)])[,-c(3,35)]

dim(newdata)

**4. Calculer la probabilité de re-identification** de chaque h-uplet combinatoire (et binning k)

Input: des données, h-uplet

Output: des probabilité de re-identification

function : prob\_re(inp, h)

h=2

x <- prob\_re(newdata,h)$prob; length(x)

**5. Calculer l’indicateur de dangerosité** – « x\_hat» avec le 5% des valeurs les plus grandes de la probabilité de re-identification (alpha = 0.05) et à un quantile très élevé, par exemple : v=0.99 ou v=0.99 (Méthode GPD, Peaks Over Threshold - POT)

*### Calculer l’indicateur de dangerosité*

Input: des probabilité de re-identification, le percentage alpha(%) des valeurs les plus grandes, le niveau de confidence

Output: l’indicateur de dangerosité

function: id.danger(x, v, alpha)

LOGIT Méthode

function: id.danger.logit(x, v, alpha)

v1=0.99; v2=0.999; alpha=0.05

id.danger.logit(x,v1,v2,alpha)

**6. Faire des graphiques**

*### Histogram de la probabilities*

> hist(x, main="Histogram of prob of re-identification",

xlab="Probability", ylab="Frequency")

> legend("topright",col=c(2,4,5),legend=c(paste("threshold",alpha\*100,"%"),

paste("threshold",(1-v)\*100,"%"), paste("threshold",(1-v2)\*100,"%")),lty=c(1,1,1),lwd=c(2,2,2))

*### Points de la probabilities*

> plot(x, ylab="Probability",xlab = paste(length(x),"combinations"),

main="prob of re-identification")

*### “Mean excess over threshold”*

> u <- id.danger(x,v,alpha)$threshold

> u1 <- id.danger(x,0.99,0.01)$threshold; ### at level 0.99

> u2 <- id.danger(x,0.999,0.001)$threshold; ### at level 0.999

> mean\_excess\_plot(x, main=paste("Threshold of",alpha\*100,"% extremes values u =",round(u,4)),xlim=c(0,max(x)))

> segments(u,0,u,1,col=2,lwd=2)

> segments(u1,0,u1,1,col=4,lwd=2)

> segments(u2,0,u2,1,col=5,lwd=2)

*### “GPD Fit” et “POT estimator”*

> tms <- id.danger(x,v,alpha)

> xi <- round(tms$shape,4); beta <- round(tms$scale,4)

> u <- tms$threshold; ex <- tms$value.extreme

> xhat1<-tms$POT; xhat2<-id.danger(x,0.999,alpha)$POT

> xx <- seq(from = 0, to = 1, length = 1000)

> z <- qgpd(xx, xi, u, beta)

> y <- pgpd(z, xi, u, beta)

> plot(sort(ex),ppoints(sort(ex)),type="p",ylab="Fu(x-u)",

main= paste("GPD Fit u =",round(u,4)), ylim=c(0,1),

xlab=paste("The",alpha\*100,"% extremes values of prob of re-identification (x>=u)"))#,log="x")

> lines(z[y >= 0], y[y >= 0], col=2, lwd=3)

> legend("bottomright",legend=c(paste("xi =",xi),paste("beta =",beta)))

> plot(sort(x),ppoints(sort(x)),type="l",ylab="F(x)",

main= paste("POT estimator"), ylim=c(0,1),

xlab=paste("The probability of re-identification (x)"))#,log="x")

> legend("bottomright",col=c(4,5),lty=c(1,1),

legend=c(paste("id\_99% = ",round(xhat1,4)),paste("id\_99.9% =",round(xhat2,4))))

> segments(xhat1,0,xhat1,v,col=4,lwd=2)

> segments(xhat1,v,0,v,col=4,lwd=2)

> segments(xhat2,0,xhat2,v,col=5,lwd=2)

> segments(xhat2,v2,0,v2,col=5,lwd=2)

par(oma=c(0,0,2,0))

par(mfrow=c(2,2))

Plot\_id(x,v1,v2,alpha)

title(main=paste("AV36 : N =",N,"observations, M =",M,"attributes, h =",h,"keys"),col=4,outer=T)