## Outils informatiques de la statistique (STA115)

### Nom de l’auditeur: Samir Chergui

### date: “14/09/2019”

### Courriel: [chergsam@yahoo.fr](mailto:chergsam@yahoo.fr)

### Tel: 0658122841

Lignes de code projet R:

Q1/Ski <- read.table('C:/Users/Samir Chergui/Desktop/STA115/ENONCES PROJET/ski\_sta115.txt',header= TRUE,sep="\t",row.names=1)

Ski\_quantitatif <- Ski[,c(2:20)]

X <- scale(Ski\_quantitatif)

Q2/Ski\_Visualisation <- Ski\_quantitatif[,c(1:10)]

summary(Ski\_Visualisation)

hist(Ski\_Visualisation$Altstation, col="blue",border="green", main=paste("Altitude des",nrow(Ski\_Visualisation),"stations de ski"), xlab="Altitude en mètres", ylab="Effectifs", xlim=c(0,2500), ylim=c(0,50), labels=TRUE)

V=Ski\_Visualisation[2:3]

plot(V, type="p", xlab="Altitude du sommet des pistes", ylab="nombre de kilomètres de piste alpin", main="Altitude des pistes par rapport au nombre de kilomètres de piste")

boxplot(Nbvertes~tlpheriques,data=Ski\_Visualisation, ylab="nombre de pistes vertes(faciles)", xlab="nombre de téléphériques")

boxplot(Nbbleues~tlpheriques,data=Ski\_Visualisation, ylab="nombre de pistes bleues(moyennes)", xlab="nombre de téléphériques")

boxplot(Nbrouges~tlpheriques,data=Ski\_Visualisation, ylab="nombre de pistes rouges(difficiles)", xlab="nombre de téléphériques")

boxplot(Nbnoires~tlpheriques,data=Ski\_Visualisation, ylab="nombre de pistes noires(très difficile)", xlab="nombre de téléphériques")

library(lattice)

a=Ski\_Visualisation$tlcabines

b=Ski\_Visualisation$tlsieges

densityplot(~b|a, xlab="nombre de télésièges", col="blue")

Q3/Ski.pca=princomp(Ski\_Visualisation,cor=TRUE)

Ski.pca

summary(Ski.pca)

Q4/Composantes\_PCA = Ski.pca$scores

Composantes\_PCA

Z= Composantes\_PCA[,c(1:2)]

Z

Q5/groupes.kmeans = kmeans(Z,centers=3)

groupes.kmeans

Q6/plot(Z, col=groupes.kmeans$cluster)

Q7/D = dist(Z)

D

Q8/class\_hierarchique = hclust(D)

class\_hierarchique

Q9/plot(class\_hierarchique)

Q10/f <- function(X) {

if (typeof(X) == typeof(Z))

if (typeof(X) != typeof(Z)) stop

G= dist(X)

P= hclust(G)

return (P)

}

f(Z)

plot(f(Z))