**Exercici .- 1 Eliminar els identificadors del conjunt de dades (0.5p)**

Responeu a les següent qüestions:

a. Quins són els atributs identificadors en el conjunt de dades?

Els atributs que identifiquen una persona de manera única i inequívocament són el DNI i el número de la SS.

b. Descriu les comandes que has emprat, indicant breument la funció de cadascun dels paràmetres

que has utilitzat.

dades\_subset = dades[,c(3,4,5)]. Amb aquesta comanda el que fem es que creem una nova variable “dades\_subset” on hi guardarem el contingut de les columnes 3, 4 i 5 que es corresponen a CP, Edat i salari.

**Exercici 2.- Re-identificació (0.5p)**

Responeu a les següent qüestions:

a. Quins d’aquests atributs o combinacions poden conduir a la identificació única d’un registre del conjunt de dades?

Hem trobat que tant per Edat, CP i Salari hi ha persones que tenen valors únics i llavors qualsevol dels 3 atributs poden identificar de manera única a una persona concreta. A més a més també hem trobat que les combinacions CP-Edat, CP-Salari, Edat-Salari i CP-Edat-Salari identifiquen a una persona concretament.

b. Indica quines comandes has fet servir per obtenir aquesta informació, explicitant el significat de les comandes i dels seus paràmetres.

dades\_CP\_freq = table(dades\_subset$CP)

Amb aquesta comanda creem una nova taula a partir del subset que hem treballat en l’apartat anterior per a la columna CP. Com a sortida d’aquesta comanda rebrem una taula amb els diferents valors que pren el CP i la freqüència amb la que aquests valors surten.

Si ara volem fer una combinació, la comanda resultant es molt similar:

dades\_CP\_Edat\_freq = table(dades\_subset$CP,dades\_subset$Edat)

Ara tenim una taula que ens mesura per a la variable1(CP) i per a la variable2(Edat) les combinacions possibles i quants cops apareixen.

Un cop hem probat totes les combinacions i tots els camps que considerem que poden ser quasi-identificadors utilitzarem:

'1' %in% TAULA

on TAULA és el nom de la taula generada anteriorment com per exemple dades\_CP\_freq i ens mostrarà TRUE si hi ha cap individu que es pot identificar de manera única.

**Exercici 3.- Aplicació de soroll additiu (additive noise) (0.5p)**

Responeu a les següent qüestions:

a. Indica la comanda que has fet servir, junt amb els paràmetres emprats i el significat de cadascun d’ells.

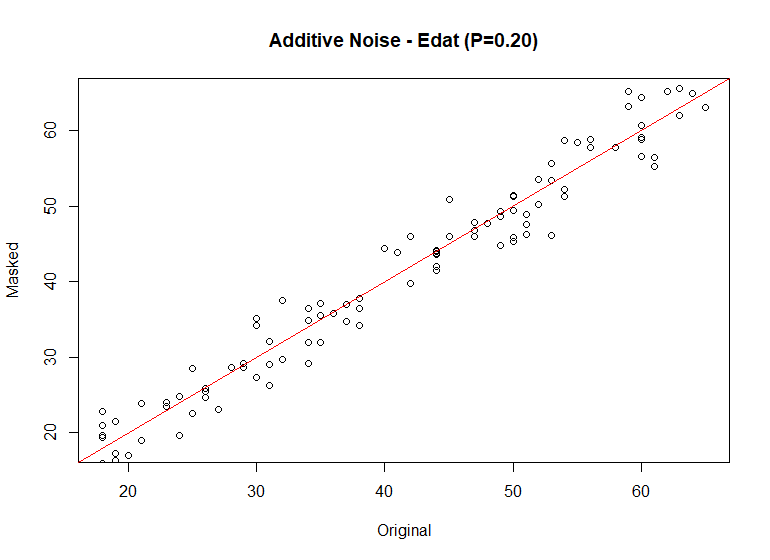
Nosaltres ho hem fet en 2 passos:

1: edat\_soroll <- addNoise(dades\_subset,'Edat',20). Amb això creem un objecte de tipus sdcMicroObj-class en el qual guardarem el soroll que hem afegit a la variable edat, el primer parametre és el data frame a partir del qual es treballa, el segon parametre és la variable o variables a les quals es vol afegir soroll i el tercer parametre es la quantitat de soroll que afegim

2: dades.an = dades\_subset[,c(1,3)].

dades.an$Edat <- edat\_soroll$xm. En aquestes dues comandes el que fem es crear un data frame “dades.an” on guardem el Salari i el CP de “dades\_subset” i després afegim el camp xm del objecte sdcMicroObj-class “edat\_soroll” que hem creat abans al data frame “dades.an”.

b. Executa el següent codi adjunt i explica els resultats obtinguts, justificant la resposta.

Obtenim aquesta gràfica, en l’eix d’abscisses tenim l’edat real de la persona i en l’eix d’oordenades l’edat després d’afegir el soroll. La línia vermella és per on haurien d’estar els punts en cas de no haver soroll, llavors tots aquells punts que es troben per sobre de la líniea vermella son persones a les quals se’ls ha afegit edat (fins un 20%) i aquells que es troben per sota son persones a les quals se’ls ha tret edat (fins un 20%).

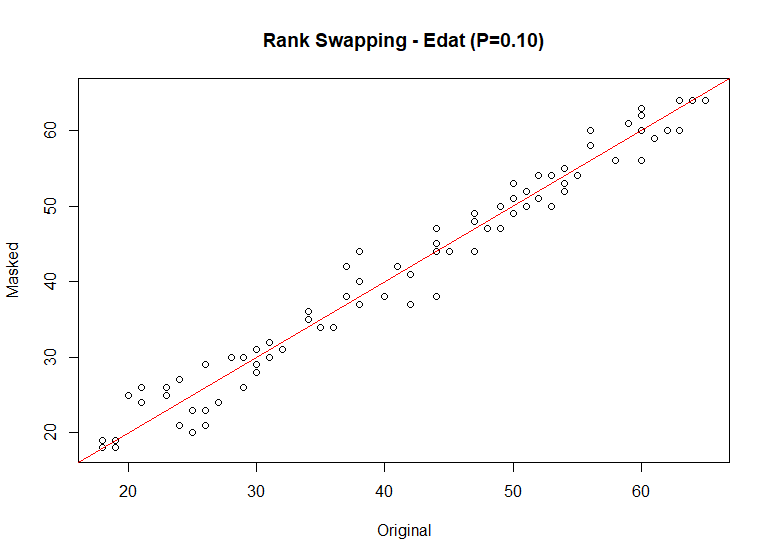
**Exercici 4.- Aplicació del mètode d’intercanvi de rang additiu (rang swap) (0.5p)**

Responeu a les següent qüestions:

a. Indica la comanda que has fet servir, junt amb els paràmetres emprats i el significat de cadascun d’ells.

dades.rs <- rankSwap(dades\_subset,'Edat',P=10). primer parametre és el data frame a partir del qual es treballa, el segon parametre és la variable o variables a les quals es vol fer el rank swapping i el tercer parametre(P) es el percentatge amb el qual es fara el rank swap

b. Genera una gràfica similar a la que hem emprat en l’exercici anterior – apartat b), on es pugui veure de forma visual i ràpida la dispersió o alteració de l’atribut Edat després d’aplicar el mètode d’intercanvi de rang.

Igual que en el cas anterior, en l’eix d’abscisses tenim l’edat real de la persona i en l’eix d’oordenades l’edat després de fer el rank swapping. També com en el cas, la línia vermella és per on haurien d’estar els punts en cas original, llavors tots aquells punts que es troben per sobre de la líniea vermella son persones a les quals se’ls ha afegit edat i aquells que es troben per sota son persones a les quals se’ls ha tret edat.

**Exercici 5.- Càlcul de la utilitat (0.5p)**

Responeu a les següent qüestions:

a. Utilitza la comanda dUtility de la llibreria sdcMicro per a estimar la pèrdua de informació que s’ha produït en generar la versió pertorbada de l’atribut Edat emprant els mètodes de soroll additiu i intercanvi de rang (generats en els dos exercicis anteriors). Indica la comanda que has fet servir, junt amb els paràmetres emprats i el significat de cadascun d’ells.

Per additive noise:

dUtility(obj=dades\_subset[2], xm=dades.an[3]). El que fem es que a partir del data frame dades\_subset agafem el camp 2 que es correspon a l’edat i el comparem amb el camp 3 del dataframe modificat dades.an.

Per rank swapping:

dUtility(obj = dades\_subset[2], xm=dades.rs[2]). El que fem es que a partir del data frame dades\_subset agafem el camp 2 que es correspon a l’edat i el comparem amb el camp 2 del dataframe modificat dades.rs.

b. Comenta els resultats obtinguts i descriu el significat dels valors obtingut i com es calculen.

Els resultats son molt similars però com era d’esperar el rank swapping ens dona menys pèrdua d’informació perquè hem afegit menys pertorbació.

Pèrdua de informació per soroll aditiu: 12.46861

Pèrdua de informació per rank swapping: 11.27276

Mesura la distància entre les dades originals i les pertorbades, escalades per la desvicacio estandar. Per a valors més alts significa que hi ha una major dispersio.

**Exercici 6.- Càlcul del nivell de privacitat (0.5p)**

Responeu a les següent qüestions:

a. Utilitza la comanda dRisk de la llibreria sdcMicro per a estimar el risc o nivell de privacitat en la versió pertorbada de l’atribut Edat, emprant els mètodes de soroll additiu i intercanvi de rang (generats en els dos exercicis anteriors). Indica la comanda que has fet servir, junt amb els paràmetres emprats i el significat de cadascun d’ells.

En ambdós casos la comanda és molt similar a l’anterior

Per additive noise:

dRisk(obj = dades\_subset[2], xm=dades.an[3]). El que fem es que a partir del data frame dades\_subset agafem el camp 2 que es correspon a l’edat i el comparem amb el camp 3 del dataframe modificat dades.an.

Per rank swapping:

dRisk(obj = dades\_subset[2], xm=dades.rs[2]). El que fem es que a partir del data frame dades\_subset agafem el camp 2 que es correspon a l’edat i el comparem amb el camp 2 del dataframe modificat dades.rs.

b. Comenta els resultats obtinguts i descriu el significat dels valors obtingut i com es calculen.

En ambdós casos obteninm dRisk = 1. dRisk és el valor en tant per 1 de la quantitat de registres que poden ser reidentificats. Això ho fa definint un interval per cada valor pertorbat. Aquest interval el calcula segons la desviació estàndard de l'atribut. Si el valor original està dins d'aquest interval, considera que pot ser reidentificat. Per tant, un valor 1 és el més dolent perquè significa que el 100% dels registres es podrien arribar a reidentificar.

**Exercici 7.- Micro-agregació univariant i multivariant (1p)**

Responeu a les següent qüestions:

a. Aplica el mètode de micro-agregació univariant amb un nivell d’agregació igual a 3 als

atributs Edat i Salari, de forma independent, sobre conjunt de dades original. Indica les

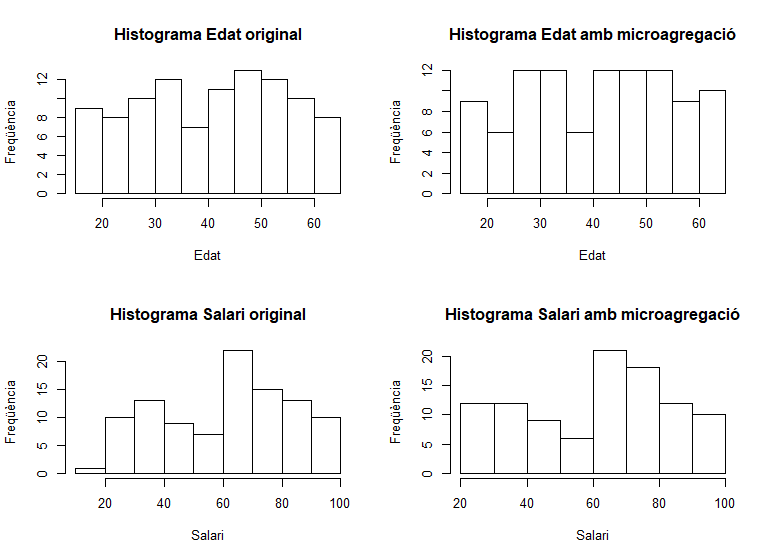
comandes i paràmetres que has fet servir i la comenta la seva funció.

microa <- microaggregation(dades\_subset[,c('Edat','Salari')],aggr = 3, method = "onedims")

Em utilitzat aquesta comanda per guardar en microa la micro-agregació univariant, el primer parametre és l’atribut original al qual es vol fer la micro-agregació, el segon parametre es el nivell d’agregacio que tal com s’ens diu ha de ser de 3 i per ultim el mètode i que en aquest cas es univariant.

b. Els histogrames són gràfiques que presenten la freqüència d’aparició segons els valors. Per tant, són molt interessants per veure quins són els valors més freqüents i quins són únics. Crea quatre gràfiques que presentin la informació de l’atribut Edat i Salari abans i després del procés de micro-agregació, i que permetin comparar de forma visual i senzilla, com s’han modificat el valors d’aquests dos atributs.

Indica el codi que has emprat, detallant la funció dels diferents Nota: la funció par(mfrow=c(2,2)) us pot ajudar a posar les 4 gràfiques en una sola imatge, de manera que simplifica la visualització.



par(mfrow=c(2,2))

hist(microa$x$Edat, main = "Histograma Edat original", xlab = "Edat", ylab = "Freqüència")

hist(microa$mx$Edat, main = "Histograma Edat amb microagregació", xlab = "Edat", ylab = "Freqüència")

hist(microa$x$Salari, main = "Histograma Salari original" , xlab = "Salari", ylab = "Freqüència")

hist(microa$mx$Salari, main = "Histograma Salari amb microagregació", xlab = "Salari", ylab = "Freqüència")

Hem creat 4 histogrames:

Histograma Edat original

Histograma Edat amb microagregació

Histograma Salari original

Histograma Salari amb microagregació

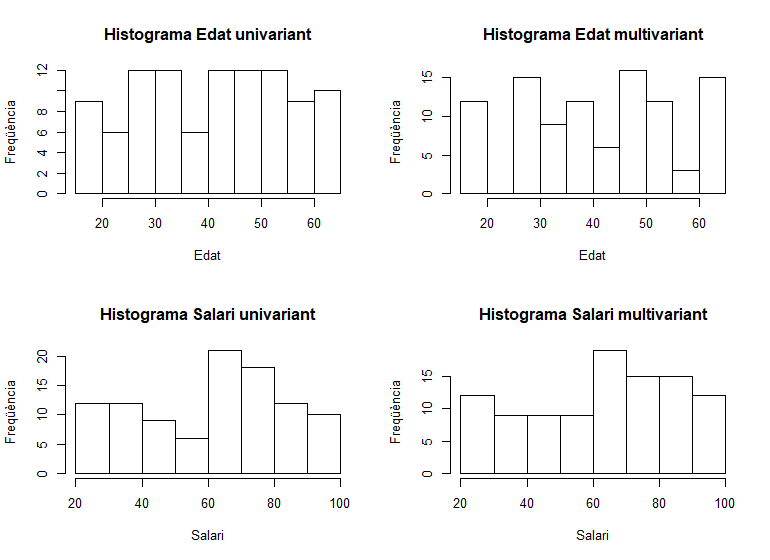
Per a tots els histogrames hem utilitzat com a primer parametre els valors els quals volem que siguin a l’eix d’abscisses en aquest cas Edat o Salari, el segon parametre és el títol de l’histograma, el tercer parametre és el nom de l’eix horitzontal i el quart parametre el nom de l’eix vertical.

c. Aplica el mètode de micro-agregació multivariant MDAV amb un nivell d’agregació igual a 3 als atributs Edat i Salari sobre conjunt de dades original. Indica les comandes i paràmetres que has fet servir i la comenta la seva funció.

microa\_mv <- microaggregation(dades\_subset[,c('Edat','Salari')],aggr = 3, method = "mdav")

Em utilitzat aquesta comanda per guardar en microa\_mv la micro-agregació multivariant, el primer parametre és l’atribut original al qual es vol fer la micro-agregació, el segon parametre es el nivell d’agregacio que tal com s’ens diu ha de ser de 3 i per ultim el mètode i que en aquest cas es multivariant.

d. Crea les gràfiques que presentin la informació dels atributs Edat i Salari anonimitzats en la seva versió multivariant, i compara els resultats obtinguts amb els obtinguts en el cas univariant.



hist(microa$mx$Edat, main="Histograma Edat univariant",xlab="Edat",ylab="Freqüència")

hist(microa\_mv$mx$Edat, main="Histograma Edat multivariant",xlab="Edat",ylab="Freqüència")

hist(microa$mx$Salari, main="Histograma Salari univariant",xlab="Salari",ylab="Freqüència")

hist(microa\_mv$mx$Salari, main="Histograma Salari multivariant",xlab="Salari",ylab="Freqüència")

Veiem que per al cas de l’Edat amb el mètode univariant les edats estan més repartides mentre que per el cas multivariant hi ha grups que concentren molts usuaris. En canvi en el cas del Salari veiem que ens dona uns valors més repartits per al cas del mètode multivariant que no per el mètode univariant.

e. Realitza una estimació del risc i de la utilitat en els dos casos (univariant i multivariant).

Comenta i justifica els resultats obtinguts.

Per a la utilitat:

Calcul de la utilitat per a l'edat univariant: 1.760311

Calcul de la utilitat per a l'edat multivariant: 10.35875

Calcul de la utilitat per al salari univariant: 2.03656

Calcul de la utilitat per al salari multivariant: 9.754602

Com veiem els resultats son els esperats, es pertorben més les dades en el cas del mètode multivariant que no pas en el mètode univariant.

Per al risc:

De nou obtenim que el valor és 1.

**Exercici 8.- Generalització d’atributs (1p)**

Responeu a les següent qüestions:

a. Crea una funció pròpia que permeti generalitzar els codis postals de població, convertintlos en codis de província. Detalle el codi de la funció, així com alguns exemples del seu funcionament i valors de retorn.

ProvinciaCP <- function(CP) {

value = substr(CP,1,2)

padding = "000"

value = paste(value,padding,sep="")

return (value)

}

El que fem amb aquesta funció és a partir d’un String d’entrada CP. Agafem els 2 primers dígits que son els que ens indiquen la provincia i els afegim un padding de 000 perque els codis postals tenen longitud de 5 digits i els concatenem i retornem el nou String.

Per exemple algu amb el codi postal 08018 sabem que es de Barcelona ciutat i ens retorna 08000 i ara només sabem que és Barcelona província

Altre exemple seria 25610 (només surt un cop i és fàcil d’identificar) que sabem que és Os de Balaguer i ens retorna 25000 i ara només sabem que és de Lleida província

b. Aplica la funció que has creat a l’atribut CP del conjunt de dades, per a tots els individus (registres) existents. Comenta el resultat obtingut.

Obtenim un nou dataframe però que els CP es corresponen només a les províncies.

c. Aplica la funció que has creat a l’atribut CP del conjunt de dades, però només a aquells

registres que siguin únics, és a dir, que la seva freqüència d’aparició sigui igual a 1 en tot el conjunt de dades. Comenta el resultat obtingut i compara-ho amb els resultats de l’apartat anterior.

ProvinciaCPValorsUnics <- function(CP) {

counter = 1

for(CPindv in CP){

if('1' %in% dades\_CP\_freq[CPindv]){

CP[counter] = ProvinciaCP(CPindv)

}

counter= counter + 1

}

return (CP)

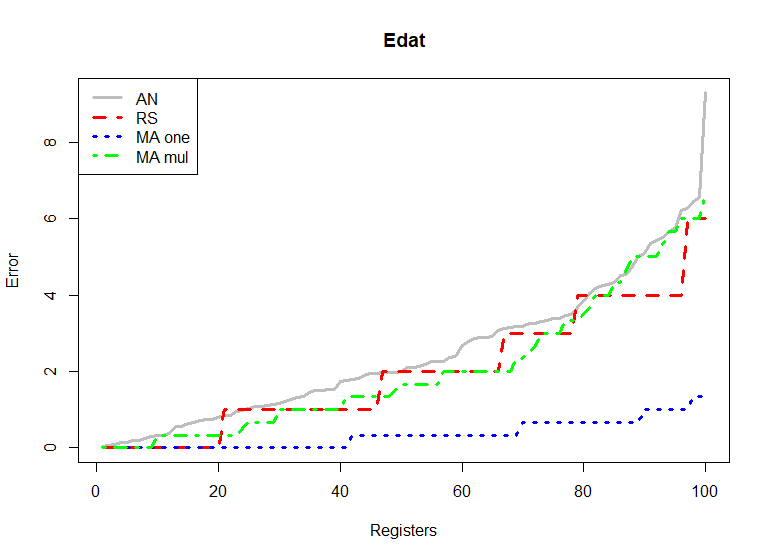
}

En aquest cas hi ha diversos CP hi ha de ciutats si hi ha més d’un individu i hi ha de províncies com en el cas de 25600 i 25610 que es transformen en 25000

**Exercici 9.- Alternatives per mesurar la pèrdua de informació (1p)**

Responeu a les següent qüestions:

a. Mostra la gràfica obtinguda.



b. Comenta i justifica els resultats obtinguts.

En aquesta gràfica es mostra l’error en els registres segons les pertorbacions que hem utilitzat. La que introdueix un valor major d’error és Additive Noise en part perquè hem posat un valor més elevat (20%). Rank Swapping es veu clarament quan introdueix error i és quan efectua un canvi de rang per això és veu tan esglaonat. Per micro-agregacio univariant vegem que és la que menor error introdueix i que també està esglaonat perquè també fa intercanvis en les dades. Per micro-agregacio multivariant es veu que esta esglaonada però que puja més ràpidament ja que introdueix més pertorbació.