

Non-linear dimensionality reduction

Principal curves, local MDS, Isomap and t-SNE

Caballero Vergés Biel, Menzenbach Svenja and Reyes Illescas Kleber Enrique

2023-10-24

PART B. Local MDS, ISOMAP and t-SNE

2. Local MDS for ZERO digits

```
zip.train <- read.table("zip.train")
zip.train.0 <- zip.train[zip.train[,1] == 0, -1]

row.names(zip.train.0) <- NULL
dist.zip.0 <- dist(zip.train.0)

k <- 5
tau <- .05
q<-2 # 2-dim config

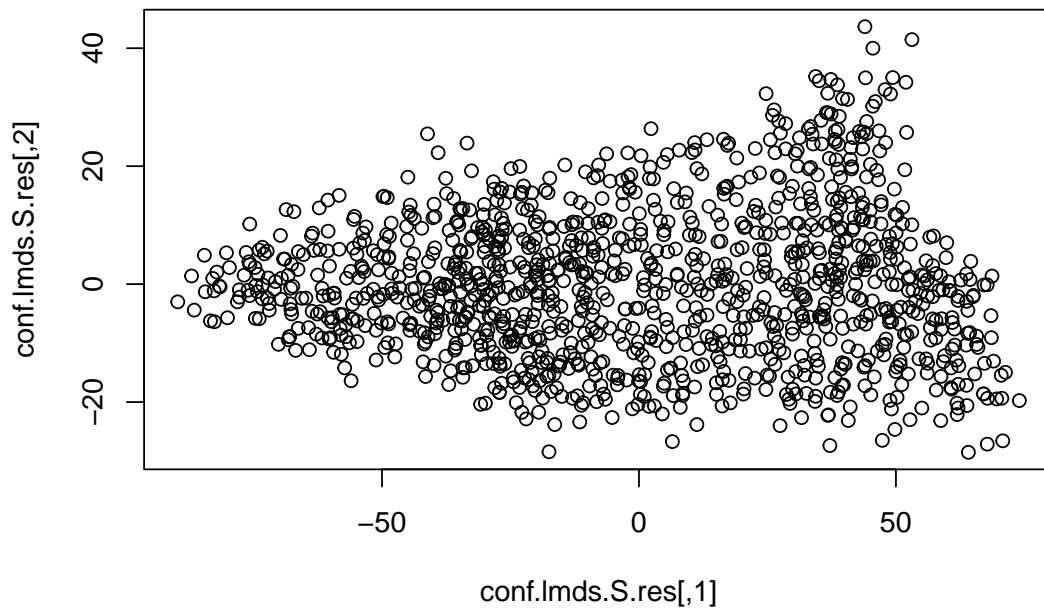
conf0 <- stats::cmdscale(dist.zip.0, k=q)

if (!file.exists("lmds1.RData")) {
  lmds.S.res <- lmds(as.matrix(dist.zip.0), init=conf0, ndim=q, k=k, tau=tau, itmax = 1000)
} else {
  load("lmds1.RData")
}

conf.lmds.S.res <- lmds.S.res$conf

plot(conf.lmds.S.res, main=paste0("Local MDS, k=",k," ", tau=",tau))
```

Local MDS, k=5, tau=0.05

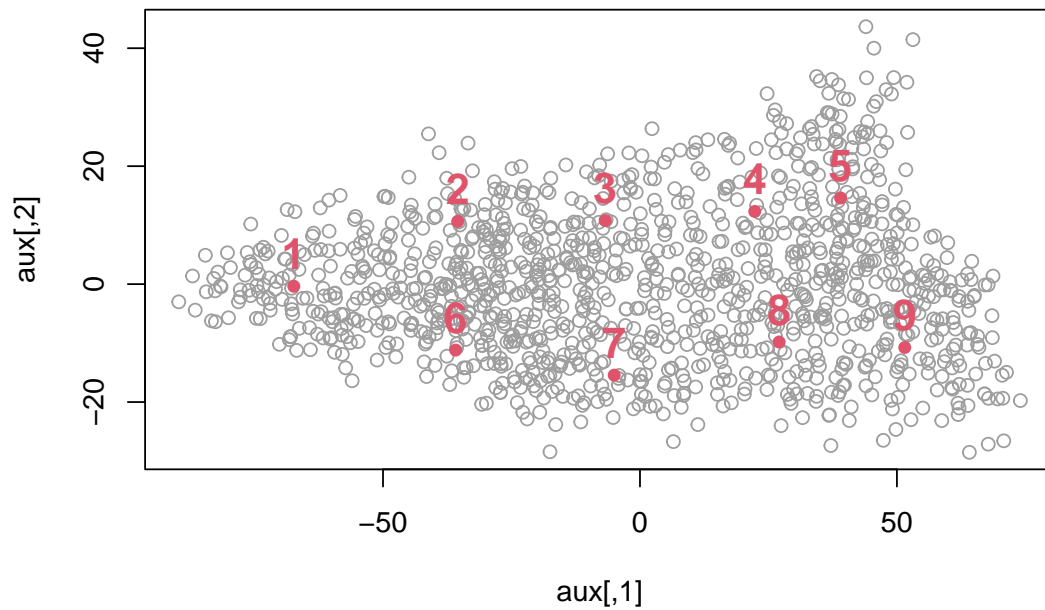


We've selected 9 points to encompass the data's variability. These points are marked in red, and each one has assigned a number in order to recognize each point with its representation using the *plot.zip* function.

```
#filt <- as.data.frame(conf.lmds.S.res)
#filt2 <- subset(filt, V1 > 40 & V1 < 55 & V2 < -7 & V2 > -14)
selected.points <- c(443, 575, 474, 291, 905, 731, 644, 242, 201)
aux <- conf.lmds.S.res[-selected.points,]
aux2 <- conf.lmds.S.res[selected.points,]

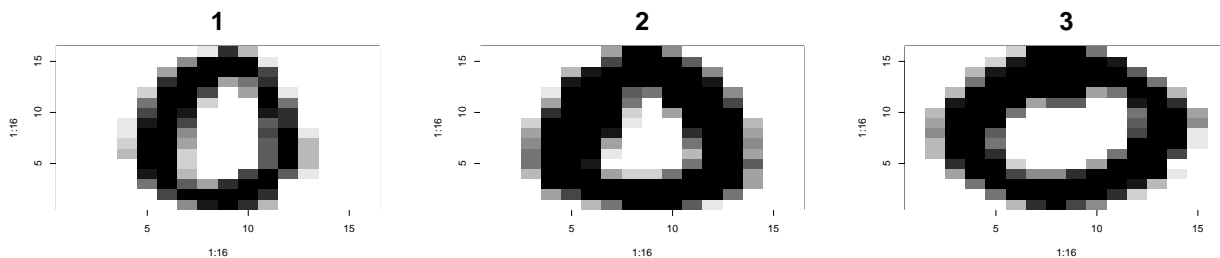
plot(aux, main=paste0("Local MDS, k=",k," tau=",tau), col=8)
points(aux2, pch = 16, col=2)
text(aux2[,1],aux2[,2],1:9, pos=3, col=2, font = 2, cex = 1.5)
```

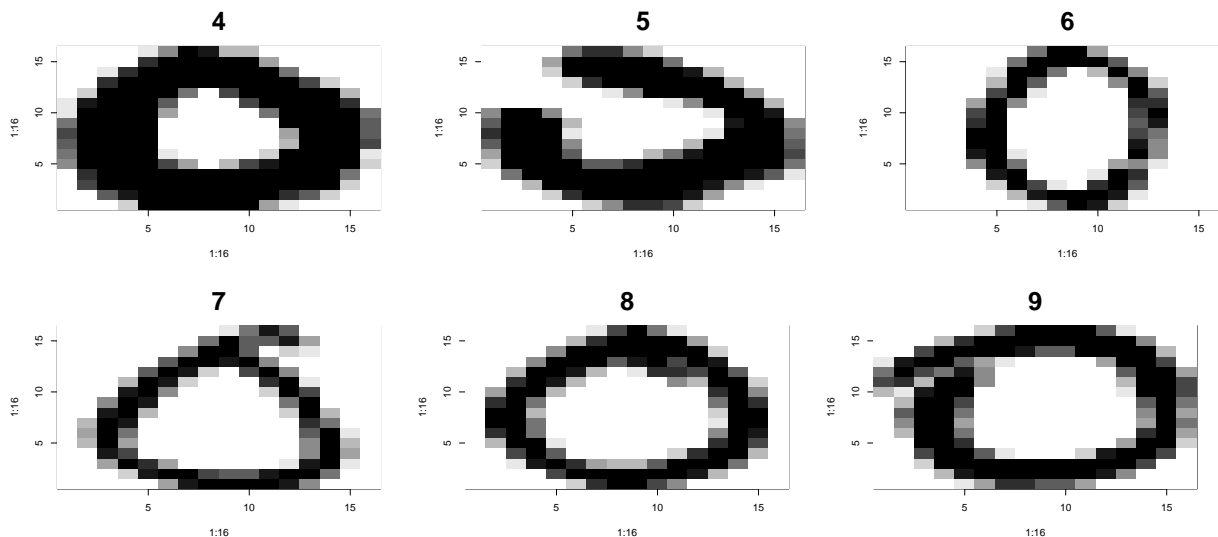
Local MDS, k=5, tau=0.05



We can observe 2 patterns in the printed zeros below. The first one (the y-axis) is related to thickness of the stroke, positive values are thicker (the higher the value, the higher the stroke) and vice versa (lower values result in thinner strokes). On the other hand, the x-axis could be associated with the roundness of the number (negative values resemble an oval shape, while positive values resemble a circle shape). Furthermore, it appears that positive values in x-axis tend to result in unfinished strokes, like fast writing.

```
for (i in 1:length(selected.points)) {
  index <- selected.points[i]
  plot.zip(zip.train.0[index,], use.first=TRUE)
  title(i, cex.main = 2.5)
}
```





```
#dist.zip.0 <- dist(zip.train.0)
q <- 2
Kp <- 10

K <- c(5,10,15)
Tau <- c(.1,.5,1)

if (!file.exists("lmds_k_tau.RData")) {
  LC <- matrix(0,nrow=length(K),ncol=length(Tau))
  lmds.k.tau <- array(vector("list",1),dim=dim(LC))

  for (i in 1:length(K)){
    for (j in 1:length(Tau)){
      lmds.k.tau[[i,j]] <- lmds(as.matrix(dist.zip.0), init=conf0,
                                ndim=q, k=K[i], tau=Tau[j], itmax=1000)$conf
      D2.k.tau <- dist(lmds.k.tau[[i,j]])
      LC[i,j] <- LCMC(dist.zip.0,D2.k.tau,Kp)$M.Kp.adj
    }
  }
} else {
  load("lmds_k_tau.RData")
  load("LC.RData")
}

ij.max <- arrayInd(which.max(LC),.dim=dim(LC))
k.max <- K[ij.max[1]]
tau.max <- Tau[ij.max[2]]
lmds.max <- lmds.k.tau[[ij.max[1],ij.max[2]]]

print(paste0("k.max=",k.max,"; tau.max=",tau.max, "; LC.max=", LC[ij.max[1], ij.max[2]]))

## [1] "k.max=5; tau.max=1; LC.max=0.2628070500589"
```

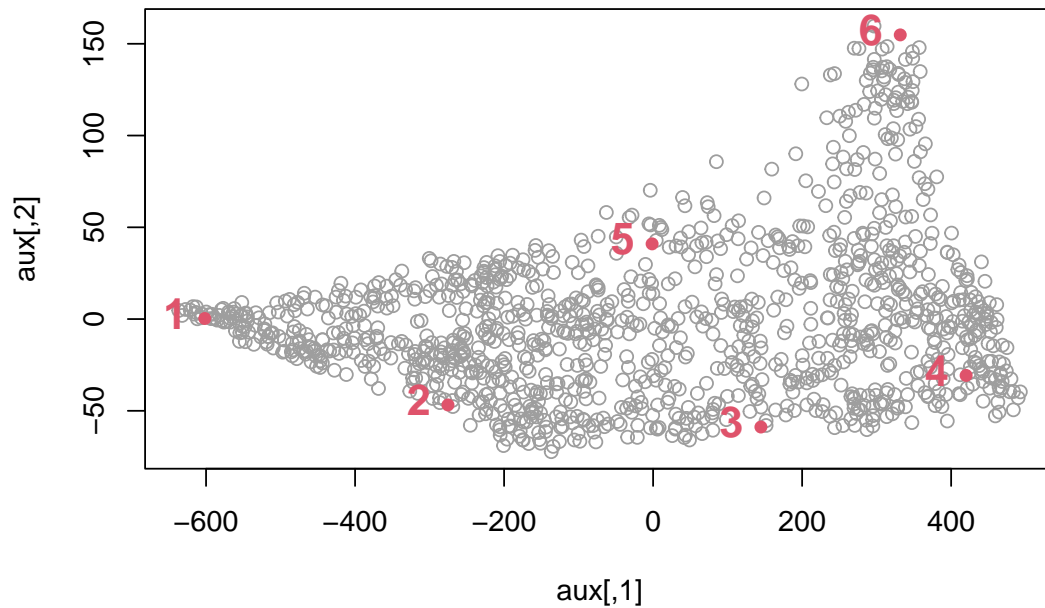
```

selected.points <- c(344, 962, 221, 530, 399, 440)
aux <- lmds.max[-selected.points,]
aux2 <- lmds.max[selected.points,]

plot(aux, main=paste0("Local MDS, k=",k.max," tau=",tau.max), col=8)
points(aux2, pch = 16, col=2)
text(aux2[,1],aux2[,2],1:6, pos=2, col=2, font = 2, cex = 1.5)

```

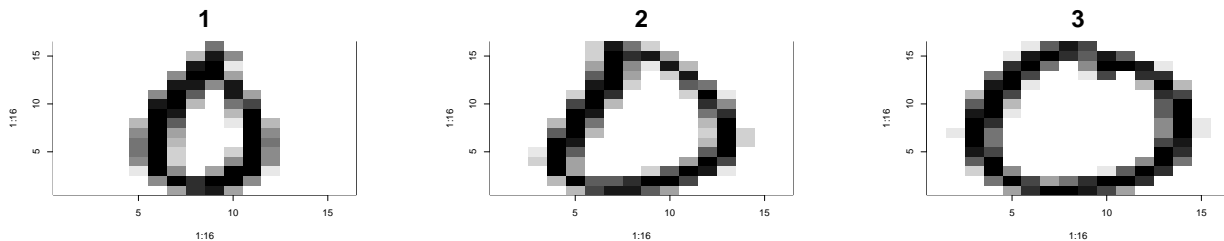
Local MDS, k=5, tau=1

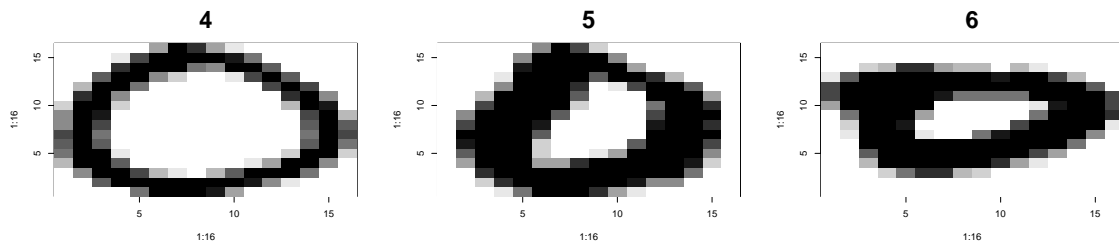


```

for (i in 1:length(selected.points)) {
  index <- selected.points[i]
  plot.zip(zip.train.0[index,], use.first=TRUE)
  title(i, cex.main=2.5)
}

```





3.

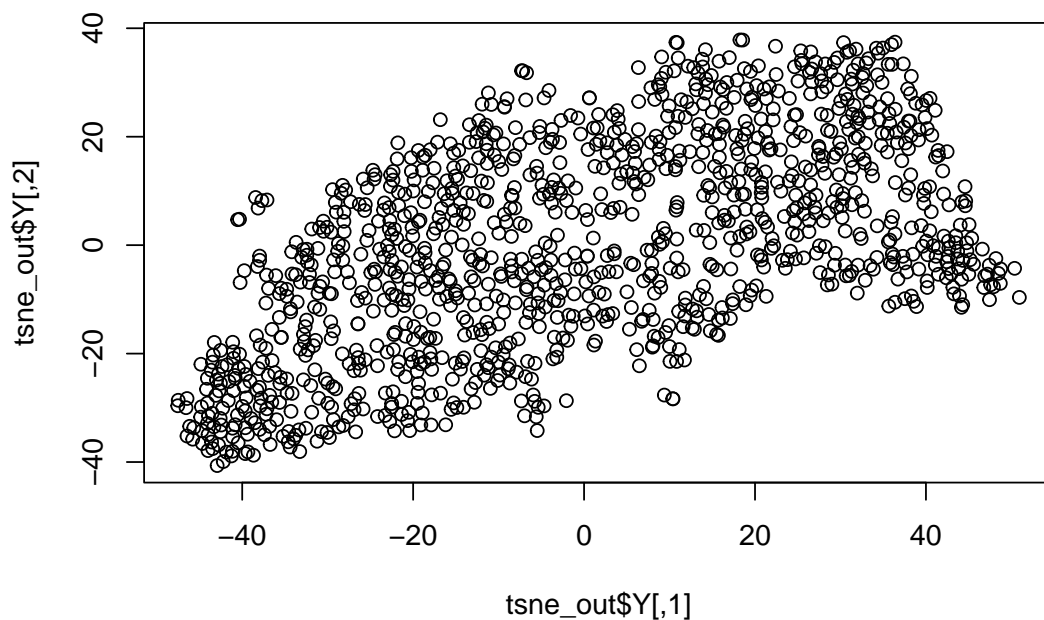
4.

In this last exercise we have tried t-sne

```
#row.names(zip.train.0) <- NULL
#zip_matrix <- as.matrix(zip.train.0)

if (!file.exists("tsne_out.RData")) {
  tsne_out <- Rtsne(dist.zip.0, dims=2, pca=FALSE, perplexity=40, theta=0.0, max_iter=1000, num_threads=4)
} else {
  load("tsne_out.RData")
}
```

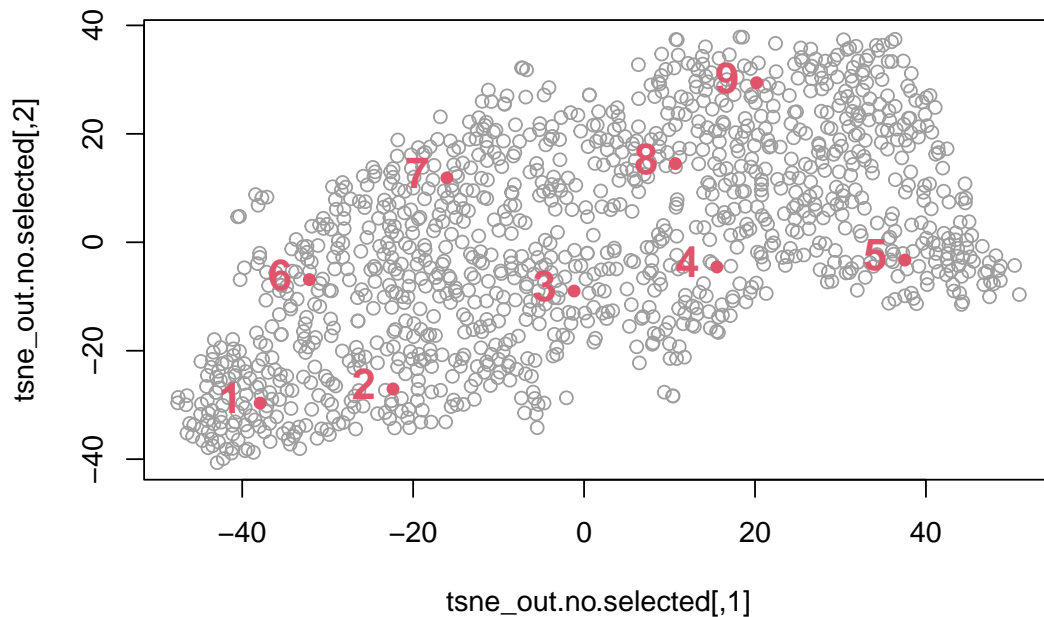
```
plot(tsne_out$Y)
```



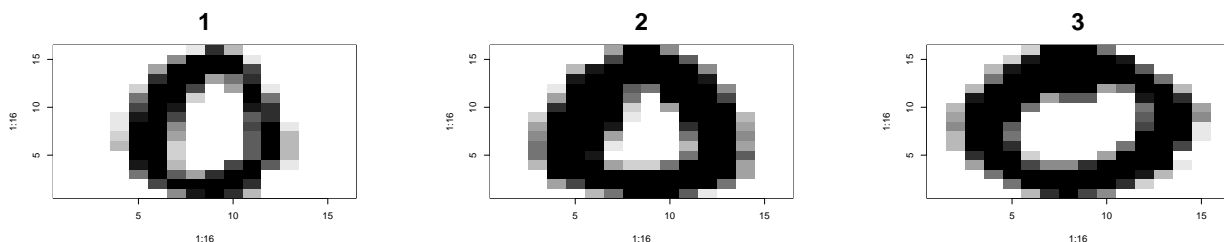
In this case we can notice that the y-axis tends to increase while x-axis increases, but the conclusions are similar to previous exercises. The y-axis is correlated with thickness and x-axis with roundness. We have to highlight the fact that in this case lower values are thickness but the overall interpretation is equal.

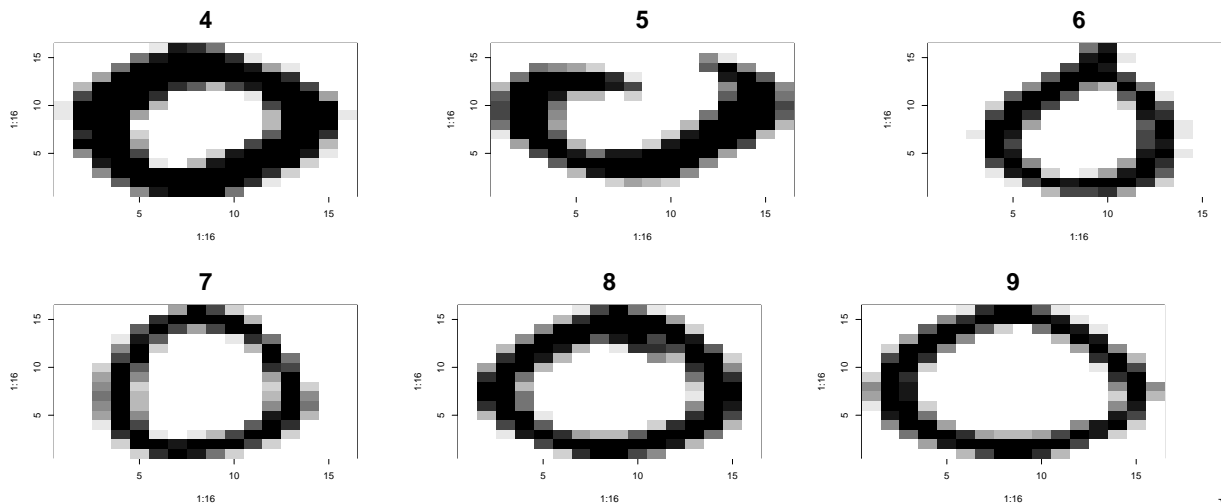
```
#filt <- as.data.frame(tsne_out$Y)
#filt2 <- subset(filt, V1 > -10 & V1 < 0 & V2 > -25 & V2 < -18)
selected.points <- c(443, 575, 474, 139, 489, 732, 198, 242, 699 )
tsne_out.no.selected <- tsne_out$Y[-selected.points,]
tsne_out.selected <- tsne_out$Y[selected.points,]

plot(tsne_out.no.selected, col=8)
points(tsne_out.selected, pch = 16, col=2)
text(tsne_out.selected[,1],tsne_out.selected[,2],1:9, pos=2, col=2, font = 2, cex = 1.5)
```



```
for (i in 1:length(selected.points)) {
  index <- selected.points[i]
  plot.zip(zip.train.0[index,], use.first=TRUE)
  title(i, cex.main=2.5)
}
```





We

apply the LCMC with the perplexity parameter.

```
Kp <- 10

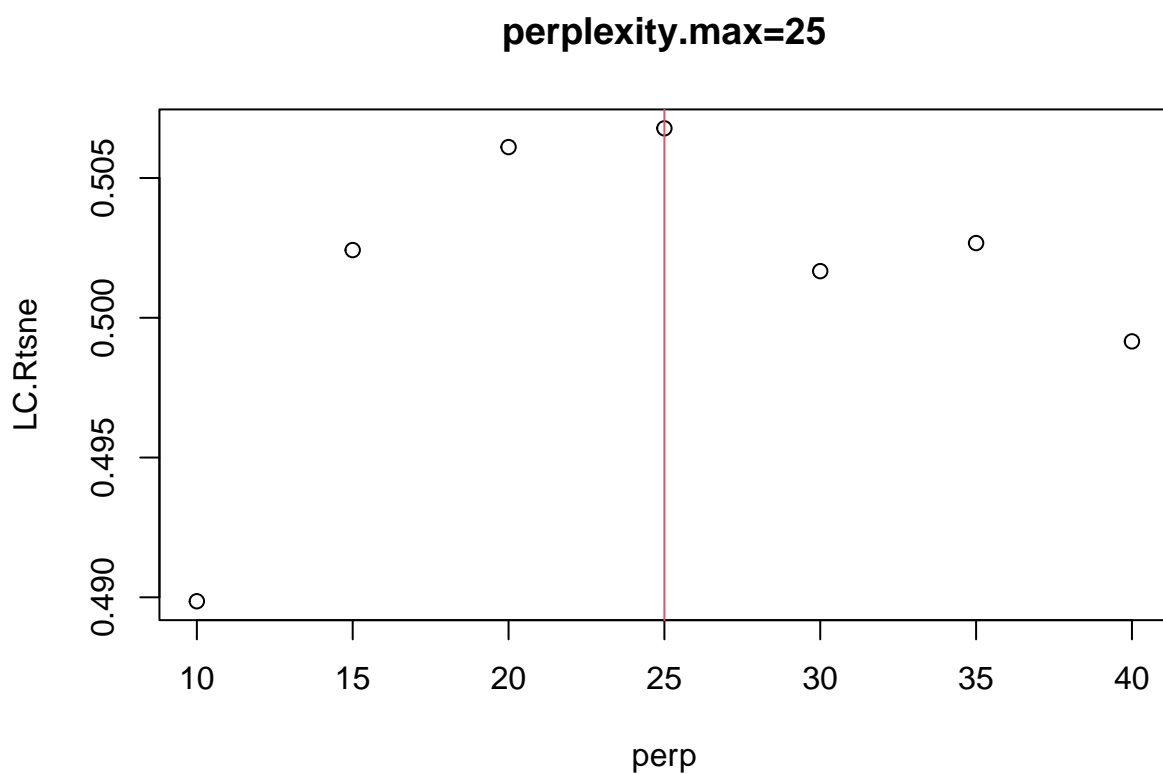
perp <- c(10,15,20,25,30,35,40)

if (!file.exists("Rtsne_perp.RData")) {
  LC.Rtsne <- numeric(length(perp))
  Rtsne.perp <- vector("list",length(perp))

  for (i in 1:length(perp)){
    Rtsne.perp[[i]] <- Rtsne(dist.zip.0, dims=2, pca=FALSE, perplexity=perp[i],theta=0.0, max_iter=1000)
    D2.perp <- dist(Rtsne.perp[[i]]$Y)
    LC.Rtsne[i] <- LCMC(dist.zip.0,D2.perp,Kp)$M.Kp.adj
  }
} else {
  load("LC_rtsne.RData")
  load("Rtsne_perp.RData")
}

i.max <- which.max(LC.Rtsne)
perplexity.max <- perp[i.max[1]]
Rtsne.max <- Rtsne.perp[[i.max]]

plot(perp, LC.Rtsne, main=paste0("perplexity.max=",perplexity.max))
abline(v=perp[i.max],col=2)
```

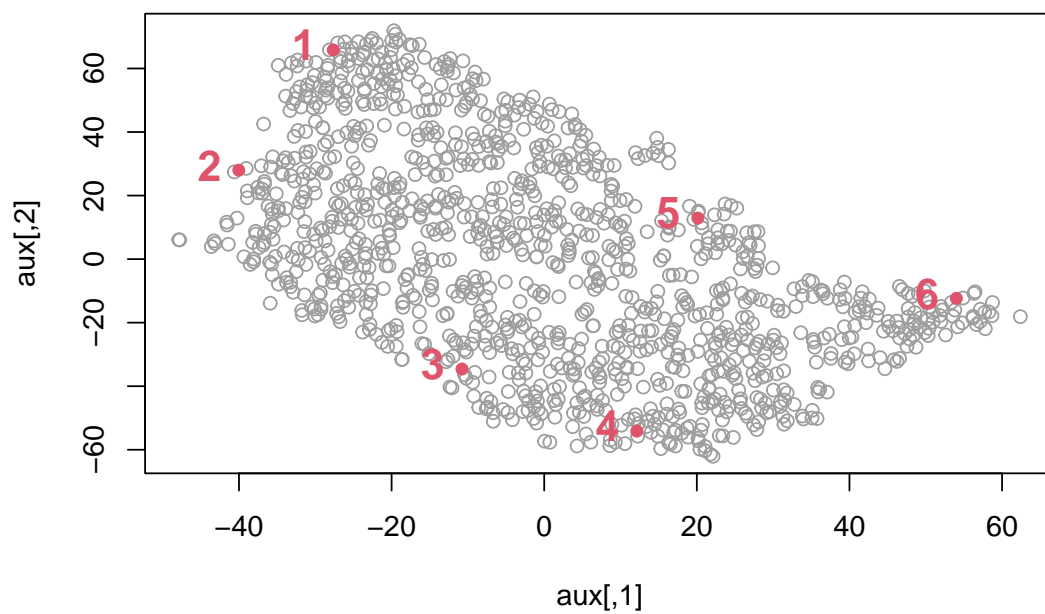



In this part we used the same points as in B2c part. The conclusions are similars

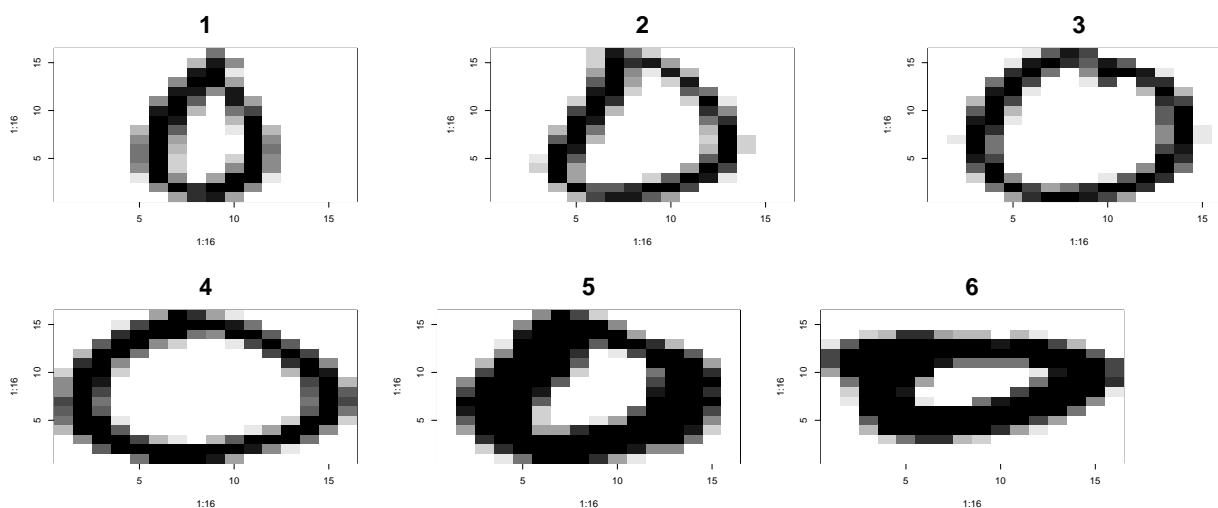
```
selected.points <- c(344, 962, 221, 530, 399, 440)
aux <- Rtsne.max$Y[-selected.points,]
aux2 <- Rtsne.max$Y[selected.points,]

plot(aux, main=paste0("Rtsne, perplexity=",perplexity.max), col=8)
points(aux2, pch = 16, col=2)
text(aux2[,1],aux2[,2],1:6, pos=2, col=2, font = 2, cex = 1.5)
```

Rtsne, perplexity=25



```
for (i in 1:length(selected.points)) {
  index <- selected.points[i]
  plot.zip(zip.train.0[index,], use.first=TRUE)
  title(i, cex.main=2.5)
}
```



5.