36-669 HW6

Daniel Nason

11/22/2021

Question 0

I plan to do a data analysis project on the Zeisel data set for the final with Anirban Chowdhury. We haven't reviewed the data set yet but would like to focus the analysis on clustering the data.

```
# setwd("C:/Users/Owner/CMU/Fall/36-669/HW/HW6")
library(PMA)
```

```
## Warning: package 'PMA' was built under R version 4.1.1
```

```
tmp <- read.csv("https://raw.githubusercontent.com/xuranw/
469_public/master/hw6/darmanis_preprocessed.csv",row.names = 1)
expr_mat <- as.matrix(tmp[,-1])
cell_types <- as.factor(tmp[,1])
source("https://raw.githubusercontent.com/xuranw/469_public/
master/hw6/hw6_functions.R")
# dim(expr_mat)
# length(cell_types)
# table(cell_types)
# expr_mat[1:5,1:5]
```

Question 1

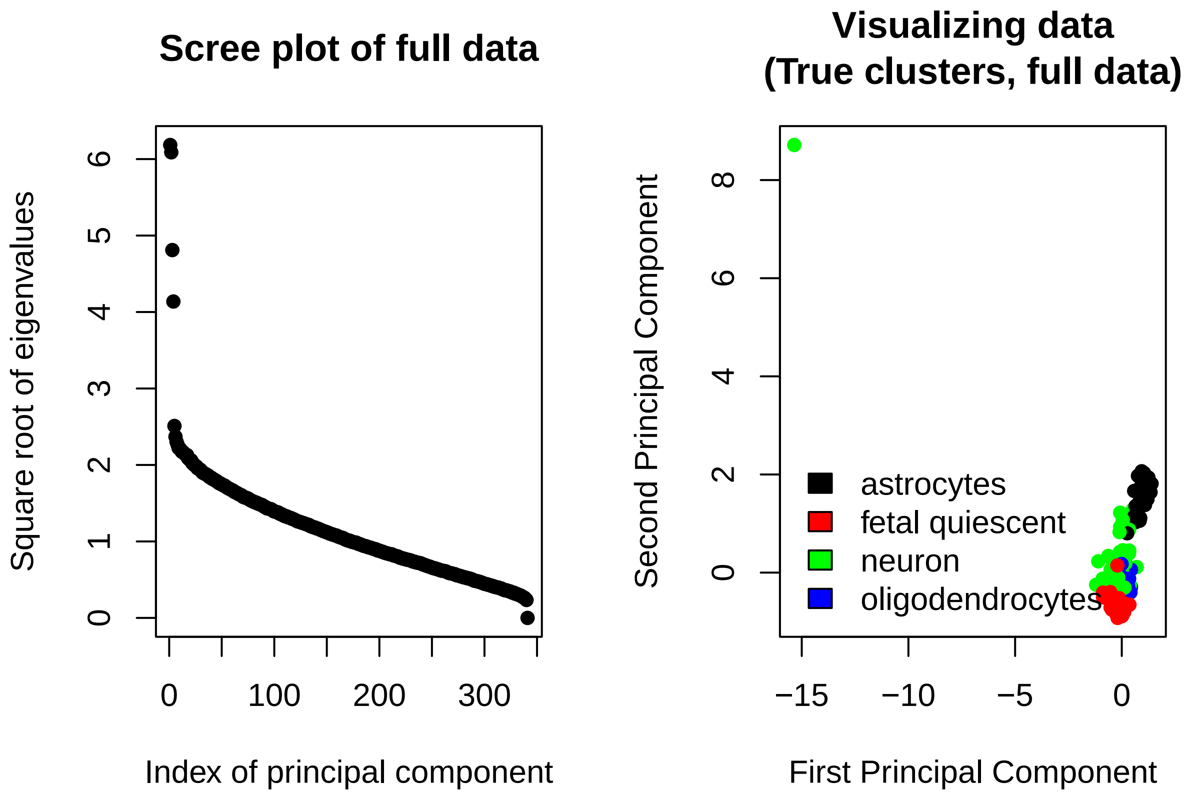
1.A

```
expr_mat <- 10**4*scale(t(expr_mat), center = FALSE, scale = rowSums(expr_mat))
expr_mat <- log2((t(expr_mat) + 1 ))
expr_mat <- scale(expr_mat)
expr_mat[1:5,1:5]
```

```
##                ATP1A2.       GJA1.   AGXT2L1.      GPR98.       AQP4.
## GSM1657871 -0.6414620 -0.5841153 -0.5243599 -0.6513356 -0.6184208
## GSM1657873 -0.6414620  0.4581023 -0.5243599 -0.5945028 -0.6184208
## GSM1657876  0.7081271  1.5889826  0.7705642 -0.1266670  0.6987435
## GSM1657877 -0.3378199 -0.5841153 -0.5243599 -0.6513356 -0.6184208
## GSM1657881  0.8066816 -0.5841153 -0.5243599 -0.6513356 -0.1482069
```

1.B

```
pca_res <- stats::prcomp(expr_mat, center = T, scale. = T)
expr_pca <- pca_res$x[,1:4]
expr_pca <- scale(expr_pca)

par(mfrow = c(1,2))
plot(pca_res$sdev, pch = 16, ylab = "Square root of eigenvalues", xlab = "Index of principal component"
plot(x = expr_pca[,1], y = expr_pca[,2], type = "p", pch = 16, xlab = "First Principal Component", ylab
legend("bottomleft", legend = c("astrocytes", "fetal quiescent", "neuron", "oligodendrocytes"), fill = 
```



```
par(mfrow = c(1,1))
```

Based on the Scree plot, we see that the first 4 principle components have the largest square rooted eigenvalues, so these would be reasonable to use for the analysis since they account for more variation in the data. The scatterplot shows that the majority of the points of the first 2 principle components cluster around 0 for each of the categories of the 4 cell types. The majority of the points in the clusters for neuron, fetal quiescent, and oligodendrocytes cells are all roughly at the origin, and the cluster for astrocytes points are slightly different in that they are mostly around (1, 2). However, there is one point for neuron that is far outside the cluster for the other points and does not fit with the rest of the clusters of the data.

1.C

```
set.seed(10)
kmean_res <- stats::kmeans(expr_mat, centers = length(unique(cell_types)))
table(kmean_res$cluster, cell_types)
```

```
##    cell_types
##     astrocytes fetal_quiescent neurons oligodendrocytes
##   1          0             109       5                1
##   2          1               1     122                1
##   3         61               0       3               36
##   4          0               0       1                0
```

```
compute_misclustering_rate(kmean_res$cluster, cell_types)
```

```
## [1] 0.143695
```

We see that the clustering error is 14.3695%, which suggests that using PCA for all of the genes is only somewhat effective in clustering the data. From these results we also see that the clustering mislabeled an entire cell type since cluster 3 contains astrocytes and oligodenrocytes while cluster 4 does not cluster any of the cell types, illustrating that multiple cell types are incorrectly clustered into a single cluster.

1.D

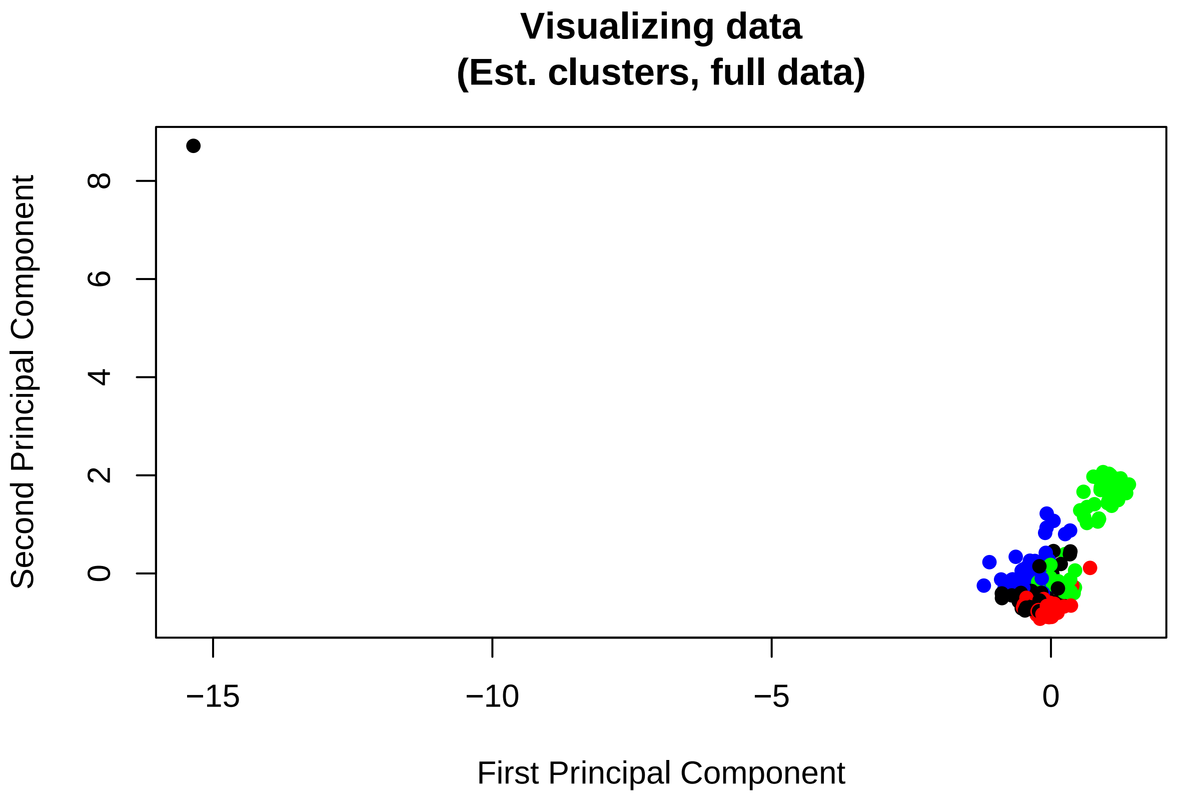
```
set.seed(10)
kmean_res2 <- stats::kmeans(expr_pca, centers = length(unique(cell_types)))
table(kmean_res2$cluster, cell_types)
```

```
##    cell_types
##     astrocytes fetal_quiescent neurons oligodendrocytes
##   1          0              21      38                1
##   2          0              89       4                0
##   3         61               0       1               37
##   4          1               0      88                0
```

```
compute_misclustering_rate(kmean_res2$cluster, cell_types)
```

```
## [1] 0.2991202
```

```
plot(x = expr_pca[,1], y = expr_pca[,2], type = "p", pch = 16, xlab = "First Principal Component", ylab
```



We see that the clustering error is 29.91202%, which illustrates that using the top 4 principle principle components for all of the genes is relatively less effective in clustering the data compared to using all of the PCAs. This is because information is lost by arbitrarily selecting only the top 4 principle components. The results show that clusters 1 contains the cell types for fetal_quiescent and neurons while cluster 3 contains cell types for astrocytes and oligodendrocytes, illustrating that multiple cell types are incorrectly clustered into a single cluster.

1.E

```
set.seed(10)
spca_cv_res <- PMA::SPC.cv(expr_mat, sumabsvs = seq(1.2, sqrt(ncol(expr_mat))/2, length.out = 10))
```

```
## Fold 1 out of 5
## Fold 2 out of 5
## Fold 3 out of 5
## Fold 4 out of 5
## Fold 5 out of 5
```

```
spca_res <- PMA::SPC(expr_mat, sumabsv = spca_cv_res$bestsumabsv1se, K = 4)
```

```
## 1234567891011121314151617181920
## 12345
## 1234
## 1234
```

```
dim(spca_res$v)
```

```
## [1] 600   4
```

```
gene_idx <- unique(sort(unlist(lapply(1:ncol(spca_res$v), function(i){
  which(spca_res$v[,i]!=0)
}))))
length(gene_idx)
```

```
## [1] 8
```

```
expr_mat_screened <- expr_mat[,gene_idx]
dim(expr_mat_screened)
```

```
## [1] 341   8
```

```
expr_mat_screened <- scale(expr_mat_screened, center = T, scale = T)
head(expr_mat_screened)
```

```
##                  GJA1.   AGXT2L1.     ERMN.   OPALIN.    GABRB2.    GABRA1.
## GSM1657871 -0.5841153 -0.5243599 2.0522604 2.7505257 -0.6271884 -0.7487121
## GSM1657873  0.4581023 -0.5243599 3.2782705 3.0318012 -0.8398126  0.7478212
## GSM1657876  1.5889826  0.7705642 1.6528755 0.3238351  0.9613630  0.1266895
## GSM1657877 -0.5841153 -0.5243599 2.8049611 2.8979837 -0.8398126 -0.2879080
## GSM1657881 -0.5841153 -0.5243599 2.9933269 2.5795206 -0.7866966 -0.7891715
## GSM1657885  2.4449522  1.9533421 0.7706621 1.4316583 -0.8398126 -0.7891715
##               C10orf95.   FLJ25363.
## GSM1657871 -0.07444386 -0.07125703
## GSM1657873 -0.07444386  1.74390724
## GSM1657876 -0.07444386 -0.07125703
## GSM1657877 -0.07444386 -0.07125703
## GSM1657881 -0.07444386 -0.07125703
## GSM1657885 -0.07444386 -0.07125703
```

```
#expr_mat_screened[1:3, 1:3]
```

1.F

```
pca_res2 <- stats::prcomp(expr_mat_screened, center = T, scale. = T)
expr_spca <- pca_res2$x[,1:4]
expr_spca <- scale(expr_spca)
dim(expr_spca)
```

```
## [1] 341   4
```

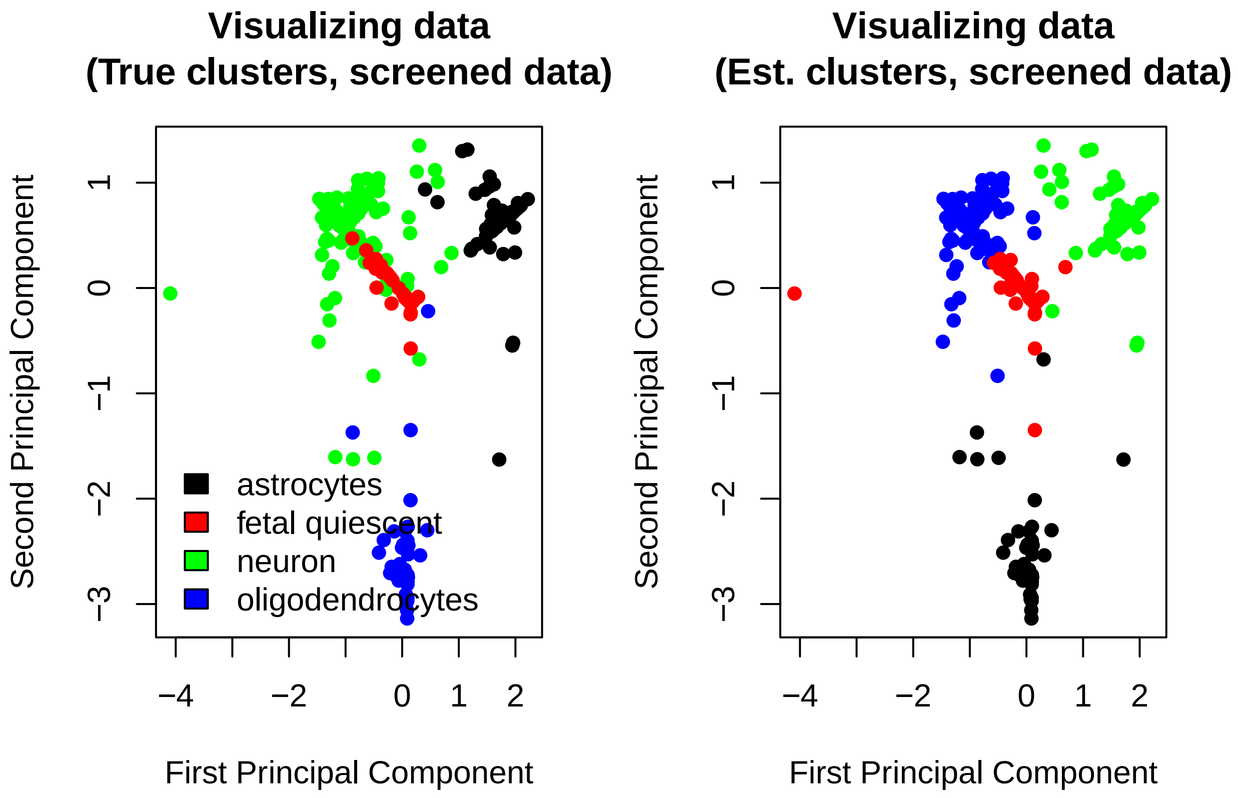
```
#plot(pca_res2$sdev, pch = 16, ylab = "Square root of eigenvalues", xlab = "Index of principal componen

set.seed(10)
kmean_res3 <- stats::kmeans(expr_spca, centers = 4)
table(kmean_res3$cluster, cell_types)
```

```
##    cell_types
##     astrocytes fetal_quiescent neurons oligodendrocytes
##   1          1               0       4               35
##   2          0             108      14                1
##   3         61               0       5                1
##   4          0               2     108                1
```

```
compute_misclustering_rate(kmean_res3$cluster, cell_types)
```

```
## [1] 0.08504399
```

```
par(mfrow = c(1,2))
plot(x = expr_spca[,1], y = expr_spca[,2], type = "p", pch = 16, xlab = "First Principal Component", yla
legend("bottomleft", legend = c("astrocytes", "fetal quiescent", "neuron", "oligodendrocytes"), fill = c
plot(x = expr_spca[,1], y = expr_spca[,2], type = "p", pch = 16, xlab = "First Principal Component", yla
```



```
par(mfrow = c(1,1))
```

The clustering results improved after deploying sparse PCA for 1.F compared to 1.B and 1.C because of the difference between PCA and sparce PCA in generating principal components. Unlike PCA, sparse PCA imposes a penalty for nonzero loadings to screen for relevant features, so only a small number of features have non-zero loadings that can be selected. Since sparce PCA takes this additional screening step to find the most important features, it has clusters that more accurately reflect the true clusters and therefore results in a smaller misclustering rate. That is, each of the four clusters contain mostly 1 of each cell type unlike the results for PCA in 1.B and 1.D, where at least one cluster contains multiple cell types and as a result have higher misclustering rates.