HW3

Napon Jatusripitak

2/17/2018

1 Distance/Similarity

Use the dataset NIPS, posted on Canvas. This dataset includes the words from all papers presented at the Neural Information Processing Systems (NIPS) Conference from 1987–2015. The data comprise 11,463 words across 5,811 unique NIPS papers, where columns are year-paperID.

```
# Load the data
NIPS <- read.csv("NIPS_1987-2015.csv", stringsAsFactors=F)
```

1. Create a new matrix that aggregates each year into a single column, so that the final matrix will contain counts of every word by the year in which the paper was presented.

```
patterns <- 1987:2015

NIPS_by_year <- sapply(patterns, function(xx) rowSums(NIPS[, grep(xx, names(NIPS)), drop=F]))
colnames(NIPS_by_year) <- patterns
rownames(NIPS_by_year) <- NIPS$X
head(NIPS_by_year)
```

```
##             1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998
## abalone        0    0    0    0    0    0    0    0    0    0    2    1
## abbeel         0    0    0    0    0    0    0    0    0    0    0    0
## abbott         0    0    0    8    2    5    6    0    0    4    6   30
## abbreviate     0    0    0    0    0    0    0    1    1    0    0    1
## abbreviated    1    0    1    0    2    3    3    1    0    2    2    0
## abc            3    3    0    3    2    4    0    1    1    5    3    2
##             1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
## abalone        5    6    1   10    7    4    2    5   11   11    0    1
## abbeel         0    0    0    0    5    4    4    8   37    0    4   32
## abbott        19    6    2    6    3    6    9    4    2   10   13   13
## abbreviate     0    4    1    1    5    1    6    6    0    1    4    6
## abbreviated    1    0    1    3    6    1    0    5    4    3    2    7
## abc            0    0    3    9    8   19   32   16    7   10   19    9
##             2011 2012 2013 2014 2015
## abalone        2   23    5    9    6
## abbeel         7    9    7   14   16
## abbott         8   11   10    6    6
## abbreviate     4    4    2    5    4
## abbreviated    5    4    5    4    4
## abc           24    9   28   26   44
```

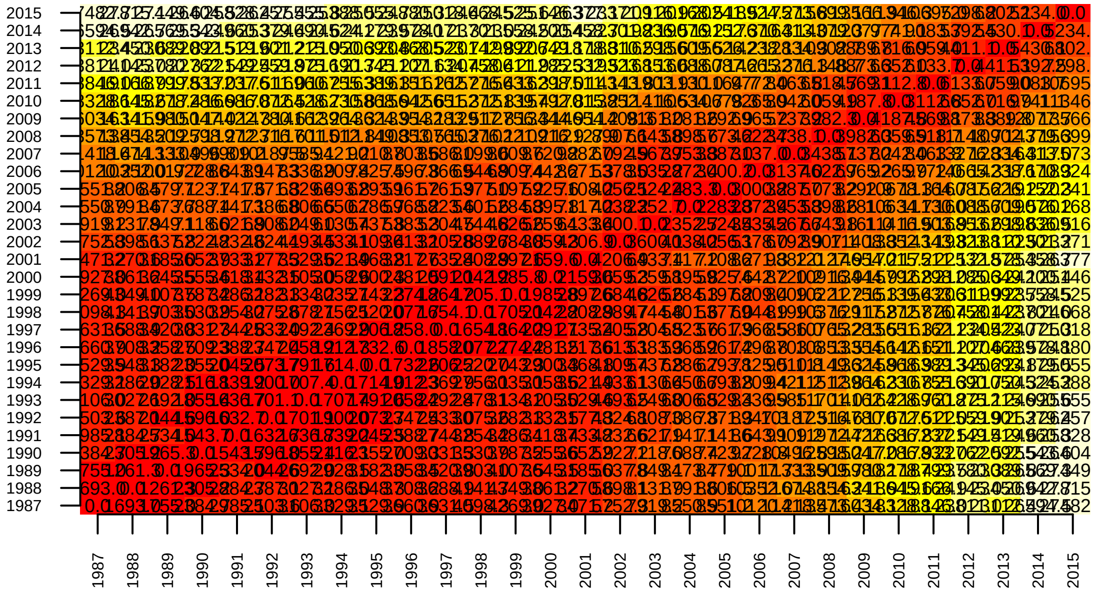
2. Measure the Euclidean distance between years; present your results either in a table or graphically.

```
sim <- dist(t(NIPS_by_year), method="euclidean")

# Heatmap (https://stackoverflow.com/questions/3081066/what-techniques-exists-in-r-to-visualize-a-dista
dst <- data.matrix(sim)
dim <- ncol(dst)
image(1:dim, 1:dim, dst, axes = FALSE, xlab="", ylab="")
```

```r
axis(1, 1:dim, colnames(NIPS_by_year), cex.axis = 0.5, las=3)
axis(2, 1:dim, colnames(NIPS_by_year), cex.axis = 0.5, las=1)

text(expand.grid(1:dim, 1:dim), sprintf("%0.1f", dst), cex=0.6)
```



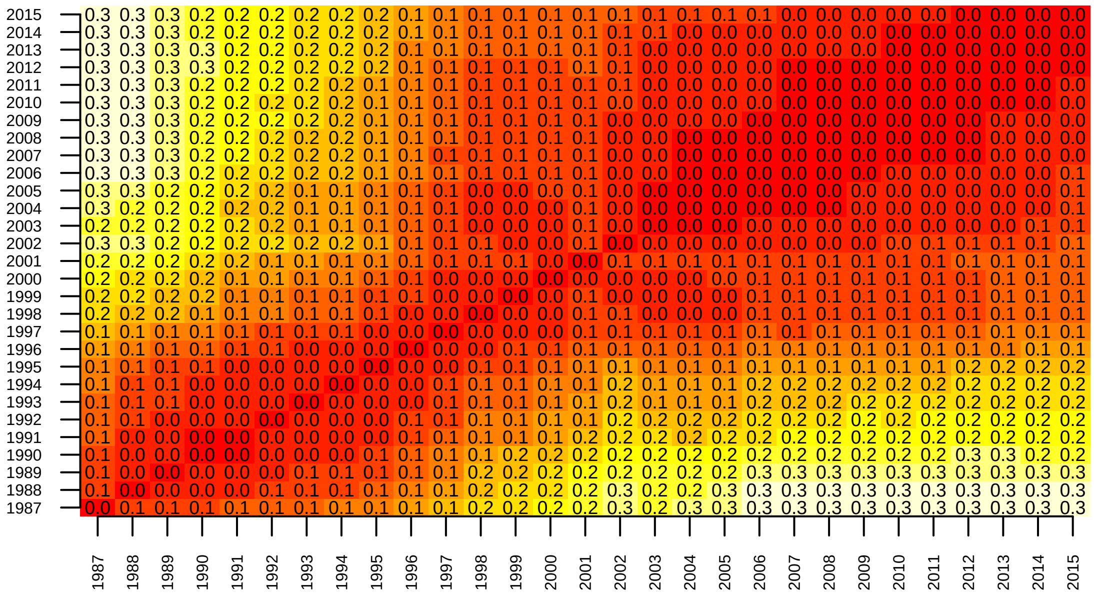
3. Measure cosine distance between years; present your results in a confusion matrix (graphical or with values).

```r
sim <- dist(t(NIPS_by_year), method="cosine")


# Heatmap (https://stackoverflow.com/questions/3081066/what-techniques-exists-in-r-to-visualize-a-dista
dst <- data.matrix(sim)
dim <- ncol(dst)
image(1:dim, 1:dim, dst, axes = FALSE, xlab="", ylab="")

axis(1, 1:dim, colnames(NIPS_by_year), cex.axis = 0.5, las=3)
axis(2, 1:dim, colnames(NIPS_by_year), cex.axis = 0.5, las=1)

text(expand.grid(1:dim, 1:dim), sprintf("%0.1f", dst), cex=0.6)
```



4. What conclusions can you draw about variation in NIPS papers over time?

Both Euclidean distance and cosine distance measures indicate that NIPS papers that are published few years apart are more similar to one another than those that are published many years apart. However, with Euclidean distance, the heatmap shows that NIPS papers published in earlier years are more similar to one another than those that are published in more recent years, hence the smaller area of shades of red towards the top right corner. With cosine distance, on the other hand, we see the reverse. That is, papers that are publushed in earlier years are less similar to one another than papers that are published in more recent years, hence the smaller area of shades of red at the bottom left corner.

2 Clustering

Use the Complaints Against Police dataset from Philadelphia (csv on Canvas).

```
# Load the data
CAP <- read.csv("ppd_complaints.csv", stringsAsFactors = F)
CAP$date_received <- as_date(CAP$date_received)
CAP$dist_occurrence <- as.factor(CAP$dist_occurrence)
```

1. Determine the number of unique "classifications" the police department uses for complaints.

```
CAP$general_cap_classification %>% n_distinct()
```

```
## [1] 13
```

```
k.value <- CAP$general_cap_classification %>% n_distinct()
```

2. Use k-means to cluster these complaints, specifying k = the number you found in part 1. Cluster with respect to date and district.

```
set.seed(12)
date.dist <- CAP %>% transmute(date = date_received, dist = dist_occurrence, label = general_cap_classi
date.dist$dist[date.dist$dist == "UNK"] <- NA
date.dist <- date.dist[complete.cases(date.dist), ]
date.dist$date.label <- date.dist$date
date.dist$date <- seq_along(date.dist$date)
```

```r
kml <- kmeans(select(date.dist, date, dist), k.value, nstart = 10)
```

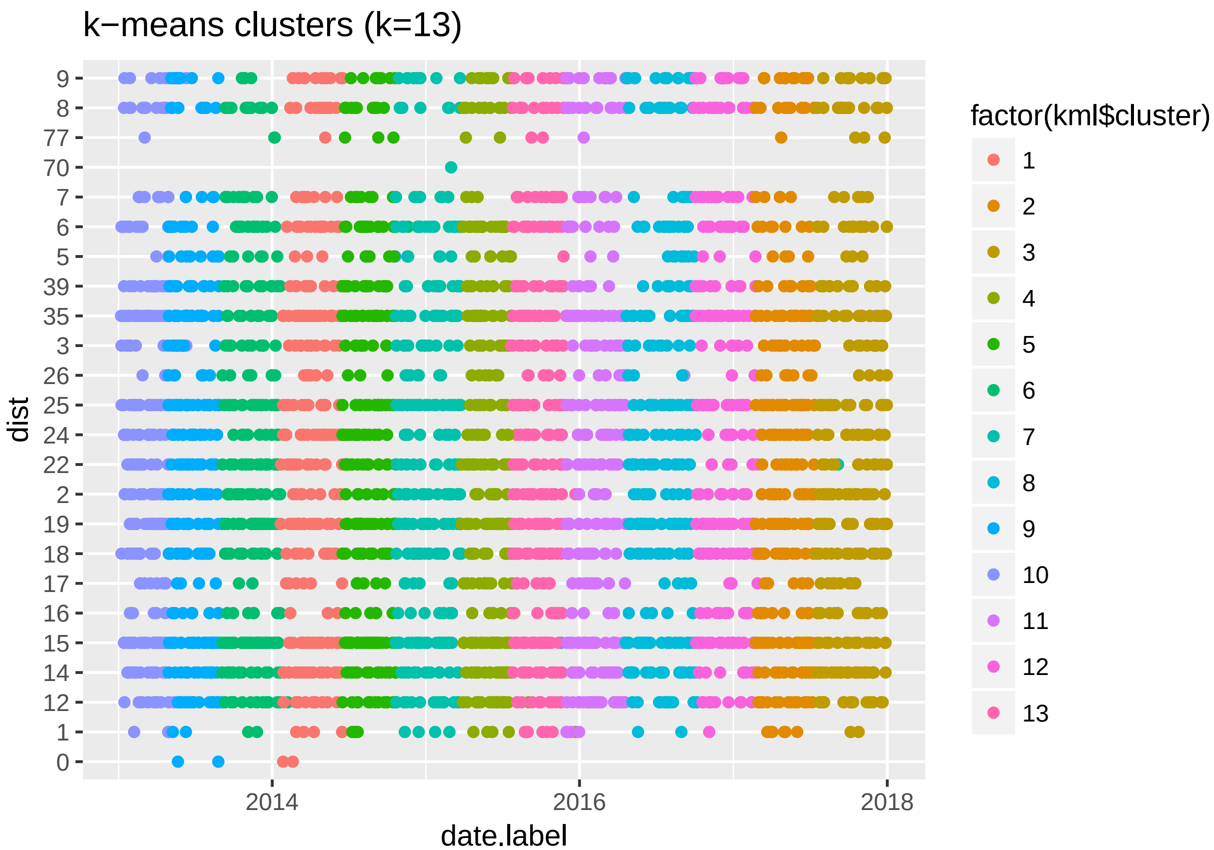
3. Plot your findings. How well did k-means perform? What do your results indicate?

First, we plot with the original labels.

```r
# Original
ggplot(date.dist, aes(date.label, dist)) + geom_point(aes(color=factor(label))) + ggtitle("Plot with ori
```

Next, we compare this to k-means clusters with k = 13.

```r
# K-means
ggplot(date.dist, aes(date.label, dist)) + geom_point(aes(color=factor(kml$cluster))) + ggtitle("k-means
```



K-means clusters with k=13 does not appear to capture the actual variation that exists between types of complaints. Here, it indicates that complaints are generally clustered by date.

4. Repeat these steps, but set k = 3. How different do your results look?

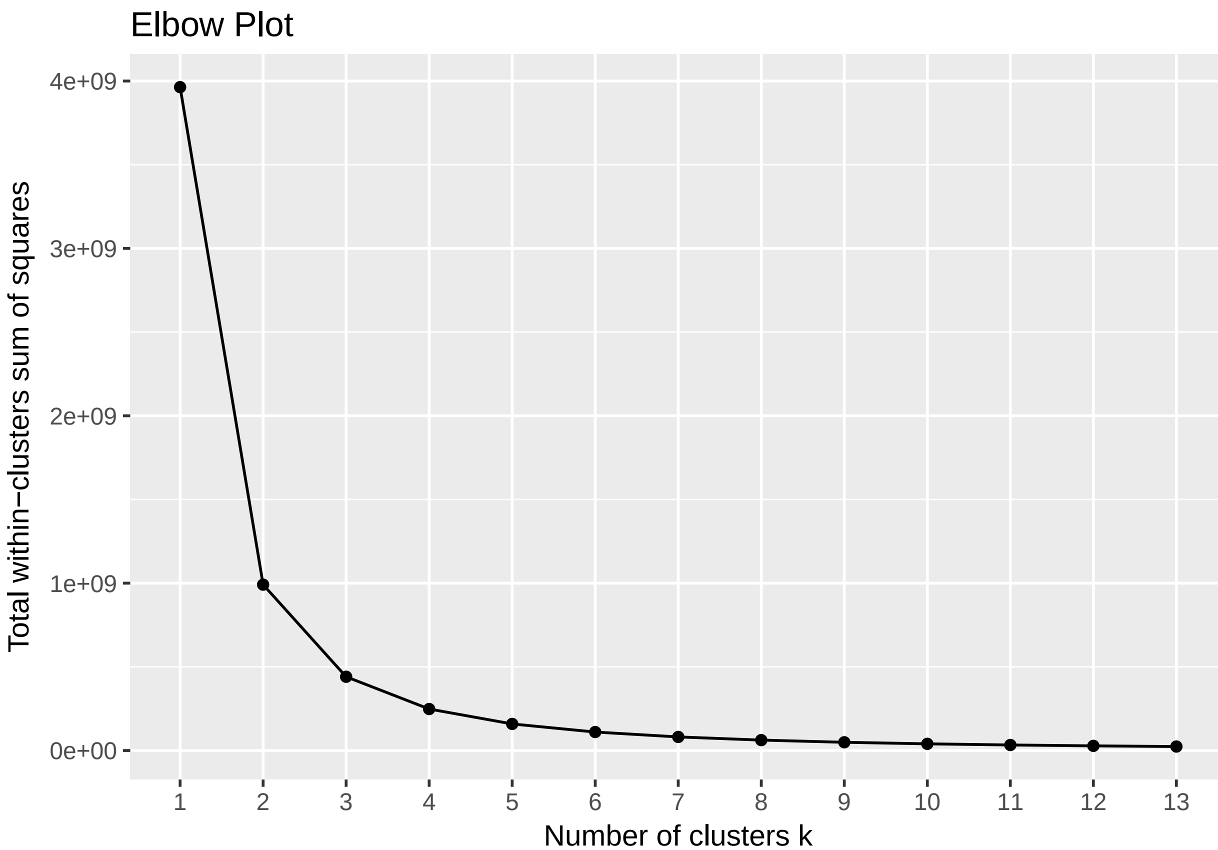
```
kml <- kmeans(select(date.dist, date, dist), 3, nstart = 10)
ggplot(date.dist, aes(date.label, dist)) + geom_point(aes(color=factor(kml$cluster))) + ggtitle("k-means
```

With k = 3, we lose much of the variation that we estimate using k=13. Compared to the original labels, this still does not do a good job at capturing the variations between types of complaints, though it produces a plot that is simpler for interpretation.

5. Create an elbow plot assessing what an optimal value for k should be in this analysis. What do you find?

```
wss <- sapply(1:k.value, function(k){kmeans(select(date.dist, date, dist), k, nstart=10, iter.max = 13)
elbowplot <- data.frame(k=1:k.value, wss=wss)

ggplot(elbowplot, aes(k, wss)) + geom_point() + geom_line() + scale_x_discrete(limits=1:13, labels=1:13
```



The optimal value for k is 3. This is the point at which the tradeoff between total within-clusters sum of squares and the number of clusters is smallest. After this point, the total within-clusters sum of squares does not decrease significantly.

3 EM

fire_data.csv is a random sample from the UK government's datasets on fire incidents and responses. The dataset contains two variables: emergency response time, and the extent of the damage caused by the fire. (csv on Canvas)

```
# Load the data
fire_data <- read.csv("fire_data.csv", stringsAsFactors = F)
```

1. Using the EM algorithm implementation in the mixtools package, evaluate the data as a function of response time and total damage (i.e., as though these data contain clusters drawn from 2 multivariate Gaussians).

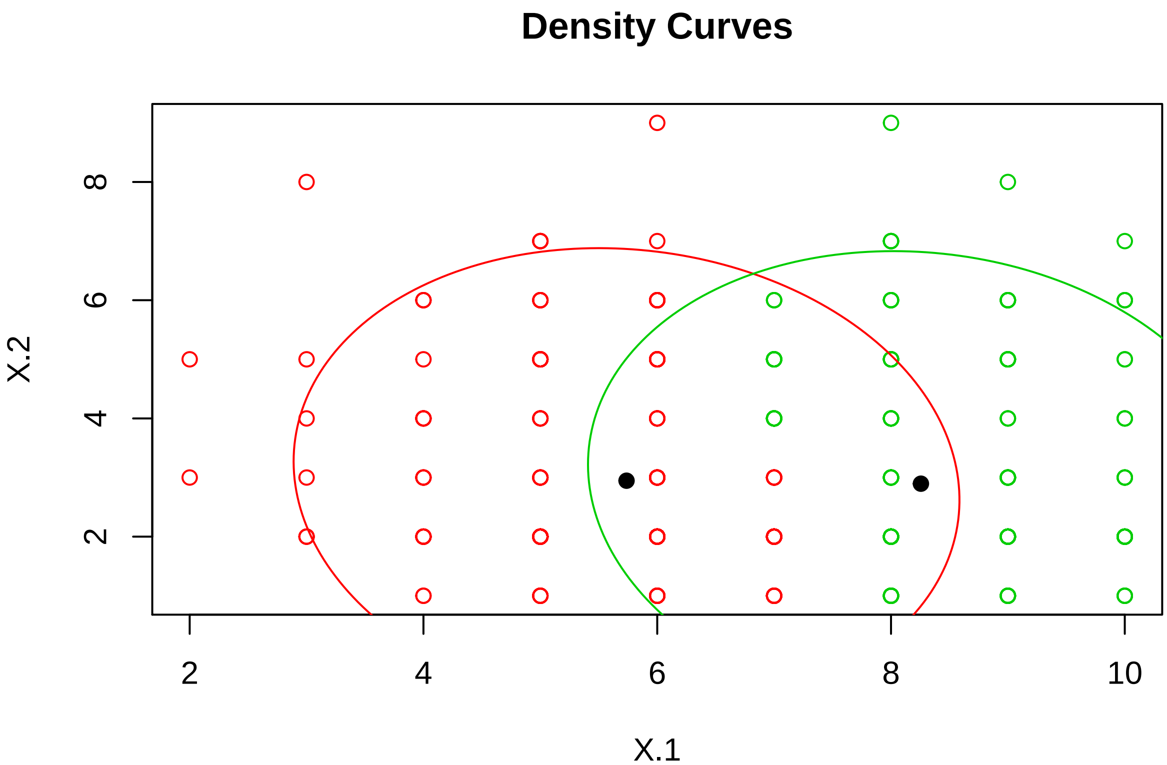
```
set.seed(123)

time.damage <- fire_data %>% select(response_time, total_damage_extent)
time.damage.mat <- as.matrix(time.damage)
em.time.damage = mvnormalmixEM(time.damage.mat, k=2, arbvar = F)
```

```
## number of iterations= 87
```

2. Plot your results.

```r
plot(em.time.damage, whichplots = 2)
```

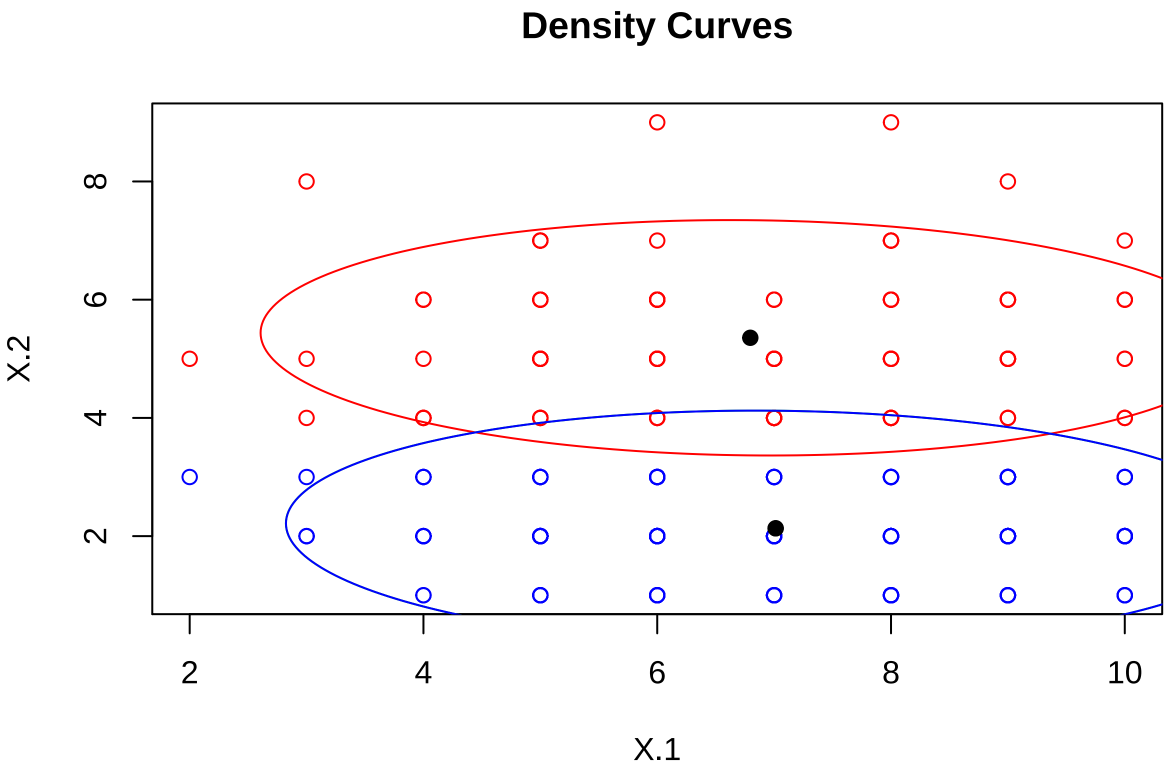


3. What happens if instead you perform the same analysis, but with k = 3? Which model is preferable for these data?

```r
em.time.damage2 = mvnormalmixEM(time.damage.mat, k=3, arbvar = F)
```

```
## number of iterations= 440
```

```r
plot(em.time.damage2, whichplots = 2)
```



It seems that there are still two clusters. However, with k=3, the overlapping regions between clusters is significantly smaller.