EE378B Homework 6 Solution

Due to: Isaac Gibbs

Part 1

If we had the the exact distance for $d_{ij}^2 = \|x_i - x_j\|_2^2 = \|x_i - \bar{x}\|_2^2 + \|\bar{x} - x_j\|_2^2 + 2\langle x_i - \bar{x}, \bar{x} - x_j \rangle$, then denoting $c_i = x_i - \bar{x}$,

$$D = \begin{bmatrix} \|c_1\|_2^2 \\ \|c_2\|_2^2 \\ \vdots \\ \|c_n\|_2^2 \end{bmatrix} 1^\top + 1 \begin{bmatrix} \|c_1\|_2^2 \\ \|c_2\|_2^2 \\ \vdots \\ \|c_n\|_2^2 \end{bmatrix}^\top - CC^\top$$

for $C = [c_1; c_2 \cdots ; c_n]$. Now since $1^\top C = 0$ and

$$\begin{bmatrix} \|c_1\|_2^2 \\ \|c_2\|_2^2 \\ \vdots \\ \|c_n\|_2^2 \end{bmatrix} 1^\top P^\perp = \begin{bmatrix} \|c_1\|_2^2 \\ \|c_2\|_2^2 \\ \vdots \\ \|c_n\|_2^2 \end{bmatrix} 1^\top (I - \frac{1}{n} 11^\top) = 0\,,$$

therefore

$$-P^\perp D P^\perp = P^\perp CC^\top P^\perp = (I - \frac{1}{n}11^\top) CC^\top (I - \frac{1}{n}11^\top) = CC^\top\,.$$

The eigen-decomposition then recovers $c_1, \cdots, c_n$ if they are all vectors in $\mathbb{R}^d$. Otherwise we recover the best rank-d approximation. When we don't have the exact distance and $\tilde{D}$ is a noisy version, this algorithm should also work up to some error term depending on the noise level.

Part 2

The code below computes the distance matrix.

```
library(geosphere)
```

```
## Warning: package 'geosphere' was built under R version 4.0.2
```

```
cities <- read.csv("citiesCleaned.csv")
cleanCities <- data.frame(City = cities$City,
                          longtitude = cities[,4] + cities[,5]/60,
                          latitude = cities[,2] + cities[,3]/60)
cityDists <- distm(cleanCities[,c(2,3)], fun= distCosine)
```

Part 3

Code for constructing the adjacency matrix is presented below.

```r
numCities <- nrow(cities)
AdjMat <- matrix(0,ncol=numCities,nrow=numCities)
for(i in 1:numCities){
  AdjMat[i,order(cityDists[,i])[1:6]] <- 1
}
```

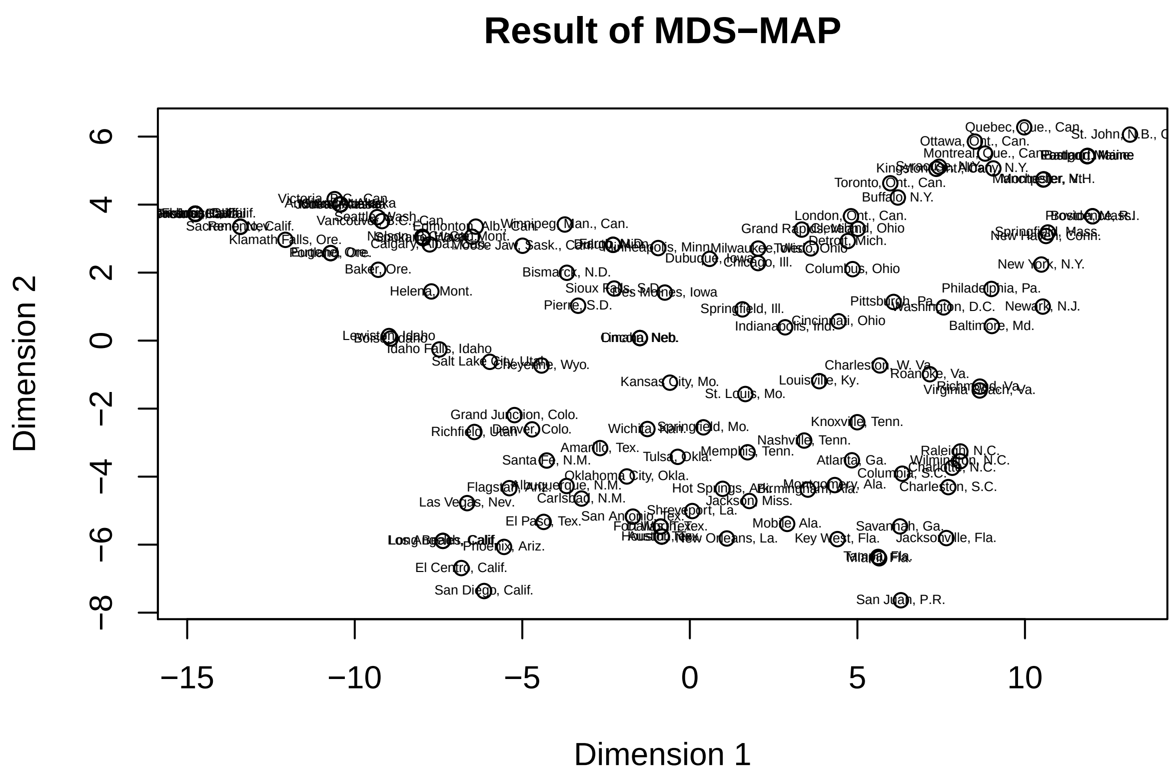
Part 4

A plot of the MDS results is shown below.

```r
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.0.2

##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

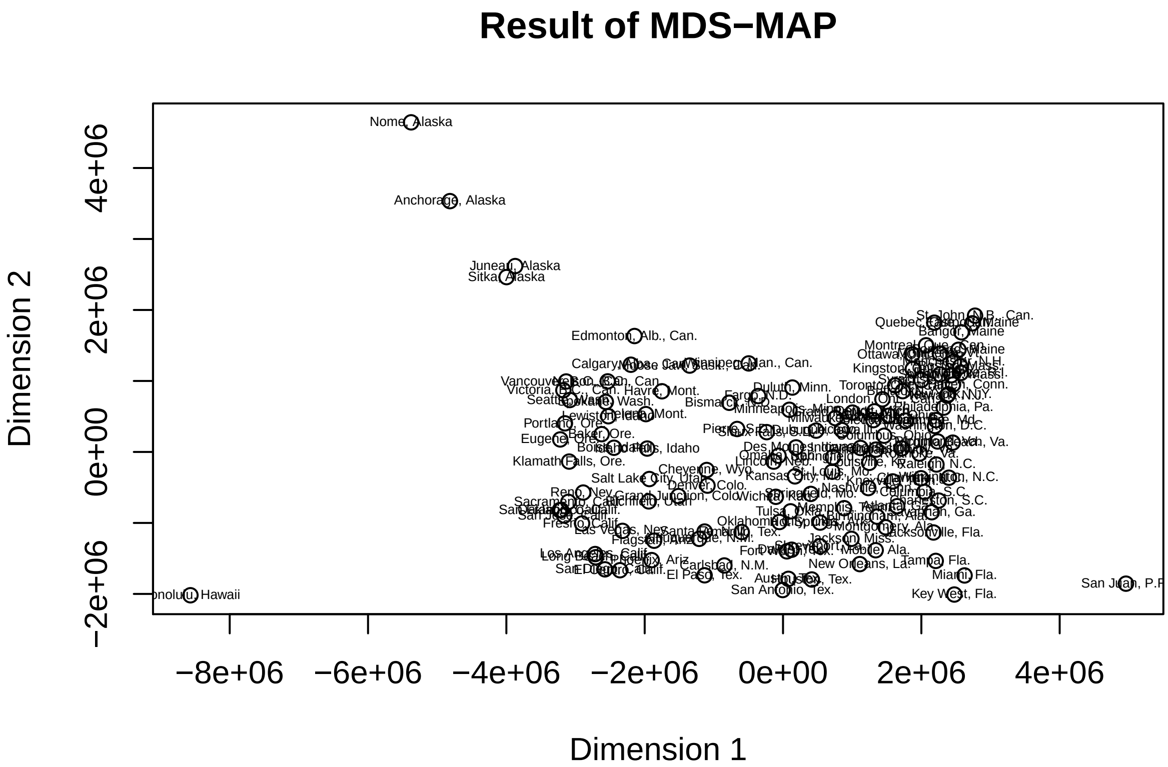
```r
MDSMAP <- function(distsMat){
  n <- nrow(distsMat)
  Pperp <- diag(1,nrow=n) - matrix(1/n,nrow=n,ncol=n)
  Q <- -Pperp %*% distsMat^2 %*% Pperp
  eigsDecomp <- eigen(Q)
  X <- eigsDecomp$vectors[,1:2] %*% diag(sqrt(eigsDecomp$values[1:2]))
  return(X)
}


edges <- c()
for(i in 1:(numCities-1)){
  for(j in (i+1):numCities){
    if(AdjMat[i,j] == 1){
      edges <- c(edges,i,j)
    }
  }
}
Dtilde <- shortest.paths(graph(edges,directed=FALSE,n=numCities))

X <- MDSMAP(Dtilde)

plot(X,xlab="Dimension 1",ylab="Dimension 2", main="Result of MDS-MAP")
text(X,labels=cities$City,cex=0.4)
```



Part 5

The results from MDS are once again plotted below.

```
X <- MDSMAP(cityDists)

plot(X,xlab="Dimension 1",ylab="Dimension 2", main="Result of MDS-MAP")
text(X,labels=cities$City,cex=0.4)
```


Result of MDS−MAP
Dimension 1
Dimension 2
−8e+06
−6e+06
−4e+06
−2e+06
0e+00
2e+06
4e+06
−2e+06
0e+00
2e+06
4e+06
Nome, Alaska
Anchorage, Alaska
Juneau, Alaska
Sitka, Alaska
Edmonton, Alb., Can.
Honolulu, Hawaii
San Juan, P.R.
Key West, Fla.
Miami, Fla.
Tampa, Fla.
Jacksonville, Fla.
New Orleans, La.
San Antonio, Tex.
El Paso, Tex.
Carlsbad, N.M.
Salt Lake City, Utah
Cheyenne, Wyo.
Denver, Colo.
Klamath Falls, Ore.
Portland, Ore.
Baker, Ore.
Duluth, Minn.
Fargo, N.D.
Bangor, Maine
Quebec, Que., Can.
Philadelphia, Pa.
Washington, D.C.
Raleigh, N.C.
Wilmington, N.C.
Charleston, S.C.
Mobile, Ala.
Jackson, Miss.
Kansas City, Mo.
Des Moines, Iowa
London, Ont., Can.