# hw7

*Hari Rajan*

## Question 1

harirajan0

```
# load the R.matlab package
library(R.matlab)
```

```
## R.matlab v3.6.2 (2018-09-26) successfully loaded. See ?R.matlab for help.

##
## Attaching package: 'R.matlab'

## The following objects are masked from 'package:base':
##
##     getOption, isOpen
```

```
# read the .mat file and store contents in fmri.p1
fmri.p1 <- readMat("data/data-science-P1.mat")
# unlist each element in the dataset and use rbind to combine into a data fram
fmri <- do.call(rbind, lapply(fmri.p1$data, unlist))
# store the coumn names for the fmri data
colnames <- 1:dim(fmri)[2]
```

```
dim(fmri)
```

```
## [1]   360 21764
```

As indicated above, the diminesions are `360 X 21764` as expected.

```
fmri[172, 2014]
```

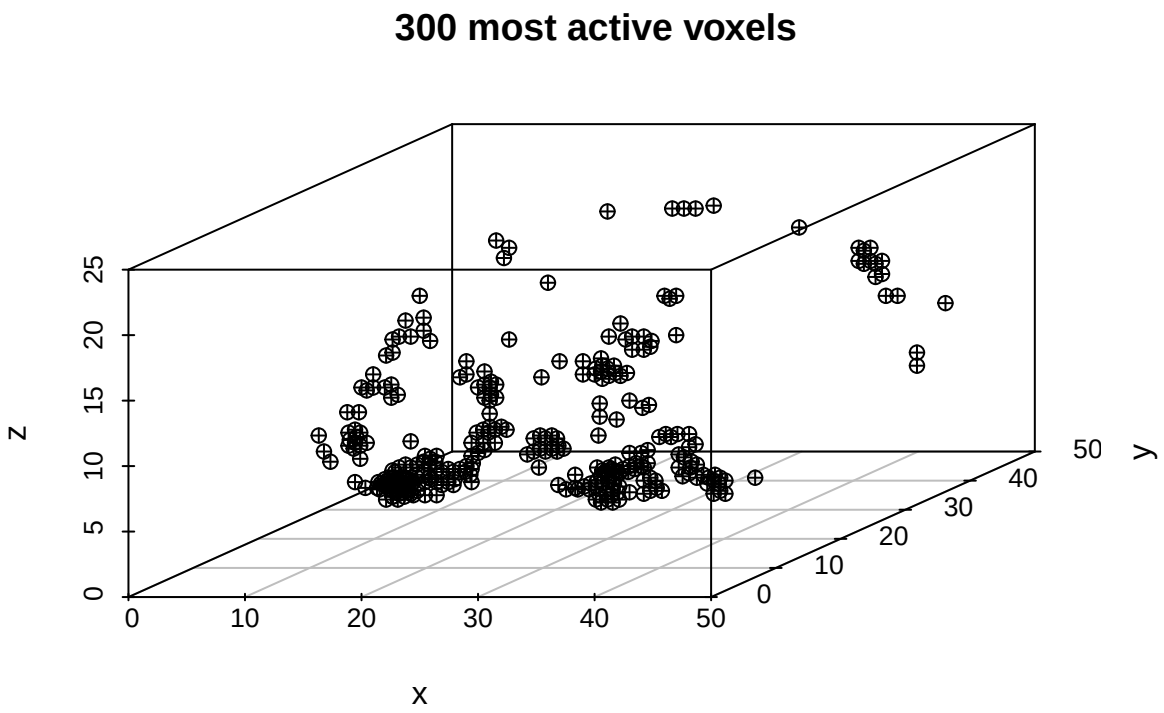
```
##
## -0.06624457
```

As indicated above, the value of `fmri[172,2014]` is $\approx -0.07$.

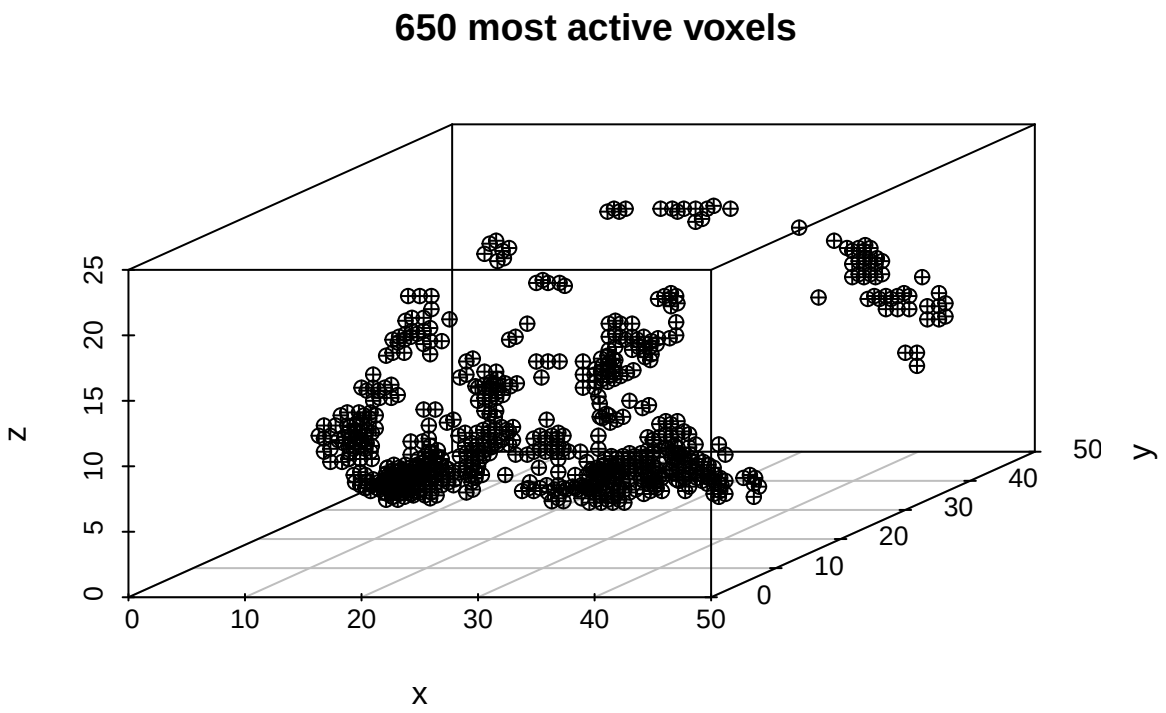
## Question 2

harirajan0

```r
# load the scatterplot3d package
library(scatterplot3d)
# total number of features
p <- 21764
# store the coordinates needed for the scatterplot in col2coord
col2coord <- fmri.p1$meta[[8]]
# compute the means of each column in the fmri data
ave.colMeans <- colMeans(fmri)
# finds the rank of each voxel
rank.fmri <- rank(ave.colMeans)
# we go in reverse order due to the rank() function to get the  300 most active voxels
scatterplot3d(x=col2coord[rank.fmri>(p-300),],pch=10,
  xlab="x", ylab="y", zlab="z", main="300 most active voxels")
```




```r
# repeat as above to plot the 650 most active voxels
scatterplot3d(x=col2coord[rank.fmri>(p-650),],pch=10,
  xlab="x", ylab="y", zlab="z", main="650 most active voxels")
```




## Question 3

calvin298, audreyametz

```
# Doing the PCA
fmri.pca = prcomp(fmri)

# Obtaining the loading vectors for each PCA
fmri.latent.sem = fmri.pca$rotation

# To obtain the variances, we first find the standard deviations for all of the principal components. W
variances = fmri.pca$sdev^2

# To get the fraction of variance for PC1, we divide the variance for PC1 by the sum of variances acros
variances[1]/sum(variances)
```

```
## [1] 0.1284251
```

## Question 4

calvinm298

The number of principal components is determined with min(n-1, p). In this case, we need the min of 360-1 and 21764, which is 359.

## Question 5

calvinm298

A biplot would be very bad because our data is very high dimensional, so we have a lot of principle components. The vectors representing the principal components would overlap each other and we wouldn't be able to tell what's going on around the source of the vectors.

## Question 6

giraudp, harirajan0

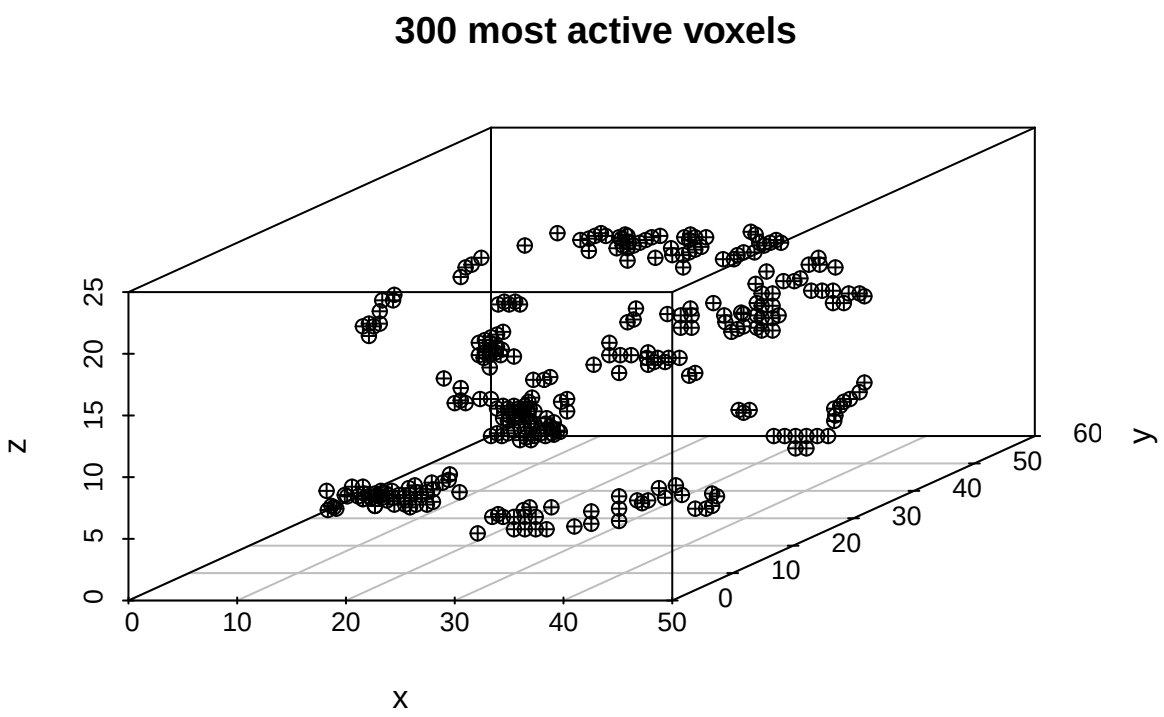
```
# This code is finding the number 300th highest loading value for the first PC. We sort the absolute va

sort(abs(fmri.latent.sem[,1]), decreasing = TRUE)[[300]]
```

```
## [1] 0.01972067
```

```
# We plot the 300 voxel coordinates which have indices matching the top 300 absolute values of the PC1

scatterplot3d(x=col2coord[(abs(fmri.latent.sem[,1]) >= 0.01972066),],pch=10,
  xlab="x", ylab="y", zlab="z", main="300 most active voxels")
```



## Question 7

giraudp

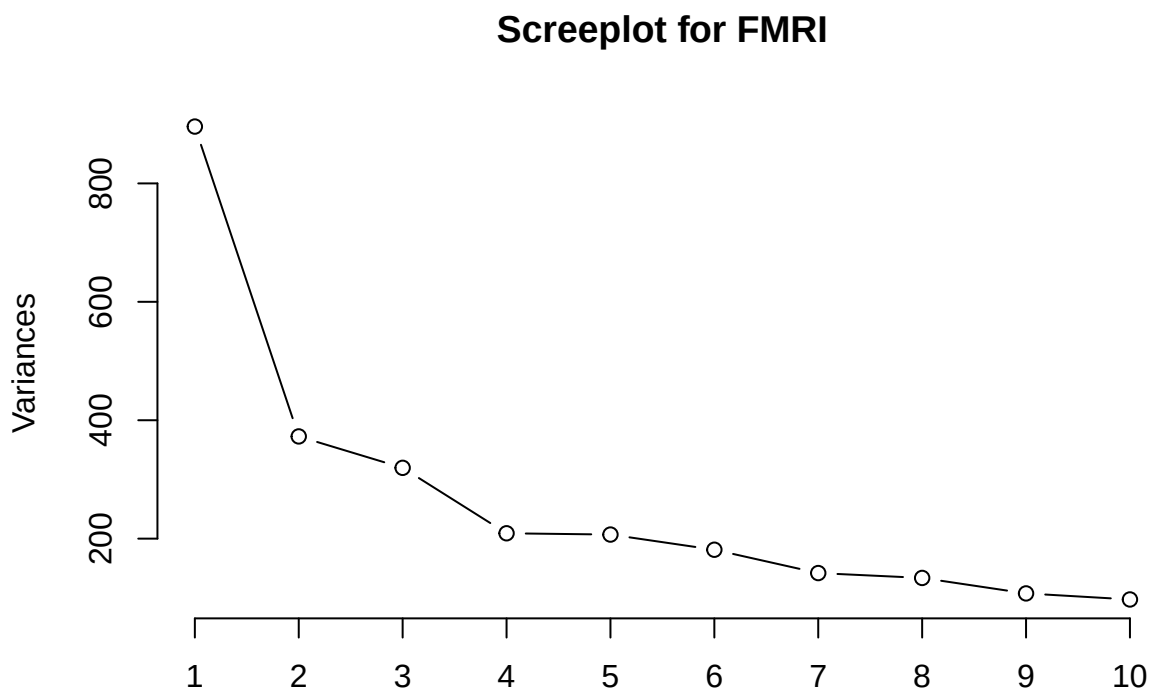
The plots from 2 and 6 are different. This is because they are graphing different things. The plot from 2 is simply graphing the voxels that have the 300 top average row values from fmri. The plot from 6 is graphing the voxels with the top 300 most extreme loadings, which relate to the top 300 most correlated coefficients to PC1, so this is clearly distinct from plotting the voxels that have the highest average values from the fmri data.

## Question 8

audreyametz

The optimal number of principal components is two, since this is the smallest number of principle components at which an adequate amount of variability is explained.

```
screeplot(fmri.pca, type= "lines", main="Screeplot for FMRI") # create a screeplot for FMRI with lines
```




audreyametz, calvinm298

## Question 9

The plot is a mass centered approximately around (25, 30, 12.5).

```
colors= c("red", "blue", "green", "orange", "purple", "pink", "blueviolet", "powderblue", "yellow", "cya
bins= cut(fmri.latent.sem[,1],10, labels=colors) # divide the first PC into 10 bins and assign each bin
scatterplot3d(x=col2coord,pch=10,
  xlab="x", ylab="y", zlab="z", main="Voxels binned by PC1", color = bins) # create3d plot of first PC
```