Final Project for Advanced Applied Statistics

Yu Han

1. Introduction

It is common for researchers to use high dimensional matrices to store large collections of data and extract information from the covariance matrices. Therefore, research on high dimensional covariance matrices means a lot to analyzing big data and putting it to application. Researchers from home and abroad have studied a lot and enjoyed several mature results on the estimation of static matrices, for example, see Bickel(2008) and Cai(2011). However, in some real cases, covariance may vary with other variables, for instance, time. In medical imaging, if we statistically store MRI signals of a patient at every particular time, then its covariance matrix will be time-dependent.

Detection of dynamic property has been attracting more and more attention by researchers from different fields. It will be of great value if we can detect the dynamic property of covariance matrices and apply appropriate methods to analysis the covariances through data. In neurology research, schizophrenia and bipolar disorder are clinically hard to diffentiate, for they both often present with psychotic symptoms. Research shows that some bipolar disorder patients can go years misdiagnosed as much as 45% of the time (Meyer, 2009). It is evident that the consequence of miscategorization is costly both economically and in terms of human suffering (DiLuca, 2014). Fortunately, recent researches have shown that incorporation of dynamics may provide a more sensitive and specific marker of disease than static connectivity (Calhoun, 2014), which means that the detection of dynamicity may help a lot in differentiating schizophrenia and bipolar clinically.

2. Data Description

I am gonna analyze the fMRI data of the attention deficit hyperactivity disorder (ADHD) patients. The preprocessed data set was made available by Neuro Bureau and the ADHD-200 consortium and can be downloaded from https://www.nitrc.org/plugins/mwiki/index.php/neurobureau:AthenaPipeline. Plus, this data set has been made to hold a data analysis competition, so descriptions and other information of the data can be found on the competition website http://fcon_1000.projects.nitrc.org/indi/adhd200/.

The data set consists 776 resting-state fMRI and anatomical datasets aggregated across 8 independent imaging sites, 491 of which were obtained from typically developing individuals and 285 in children and adolescents with ADHD (ages: 7-21 years old). The phenotype most related to our analysis is the sympotoms, where 0 denotes Typically Developing Children, 1 denotes ADHD-Combined, 2 denotes ADHD-Hyperactive/Impulsive and 3 denotes ADHD-Inattentive.

First, I read description data as follows.

```
library(readr)
library(reshape2)
library(ggplot2)
phenotype <- read_csv(file = "/Users/HanY/Desktop/ADHD/adhd200_preprocessed_phenotypics.csv")
```

```
## Parsed with column specification:
## cols(
##   `ScanDir ID` = col_double(),
##   Site = col_double(),
##   Gender = col_double(),
##   Age = col_double(),
##   Handedness = col_character(),
##   DX = col_character(),
```

```
##   `Secondary Dx` = col_character(),
##   `ADHD Measure` = col_character(),
##   `ADHD Index` = col_character(),
##   Inattentive = col_character(),
##   `Hyper/Impulsive` = col_character(),
##   `IQ Measure` = col_character(),
##   `Verbal IQ` = col_character(),
##   `Performance IQ` = col_character(),
##   `Full2 IQ` = col_character(),
##   `Full4 IQ` = col_character(),
##   `Med Status` = col_character(),
##   QC_Athena = col_character(),
##   QC_NIAK = col_character()
## )
```

```
head(phenotype)
```

```
## # A tibble: 6 x 19
##   `ScanDir ID`  Site Gender   Age Handedness DX    `Secondary Dx`
##          <dbl> <dbl>  <dbl> <dbl> <chr>      <chr> <chr>
## 1      2371032     3      0 10.7  1          0     <NA>
## 2      2026113     3      0 13.0  1          1     <NA>
## 3      3434578     3      0  8.12 1          0     <NA>
## 4      8628223     3      0 10.8  1          0     Simple phobia
## 5      1623716     3      0 12.6  1          1     <NA>
## 6      1594156     3      1 12.9  1          0     Simple Phobia
## # ... with 12 more variables: `ADHD Measure` <chr>, `ADHD Index` <chr>,
## #   Inattentive <chr>, `Hyper/Impulsive` <chr>, `IQ Measure` <chr>,
## #   `Verbal IQ` <chr>, `Performance IQ` <chr>, `Full2 IQ` <chr>, `Full4
## #   IQ` <chr>, `Med Status` <chr>, QC_Athena <chr>, QC_NIAK <chr>
```

Then I get some infomation of the data. Each individual data consists 172 scanned data of 190 different regions of interest (ROI).

```
NYU1000804_0 = read_csv(file = "/Users/HanY/Desktop/ADHD/ADHD200_CC200_TCs_filtfix/NYU/1000804/sfnwmrda
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   File = col_character(),
##   `Sub-brick` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
dim(NYU1000804_0)
```

```
## [1] 172 192
```

3. Detection of Dynamic Property

To check the dynamic characteristics of the data, I split the 172 observations into two sets, each consisting 86 observations. Then, I plot the heatmaps of the first 20 ROIs for individual 0010002 (DX=3).

```
MyData <- read_csv(file="/Users/HanY/Desktop/ADHD/ADHD200_CC200_TCs_filtfix/NYU/0010002/sfnwmrda0010002
```

```
## Parsed with column specification:
```

```
## cols(
##   .default = col_double(),
##   File = col_character(),
##   `Sub-brick` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

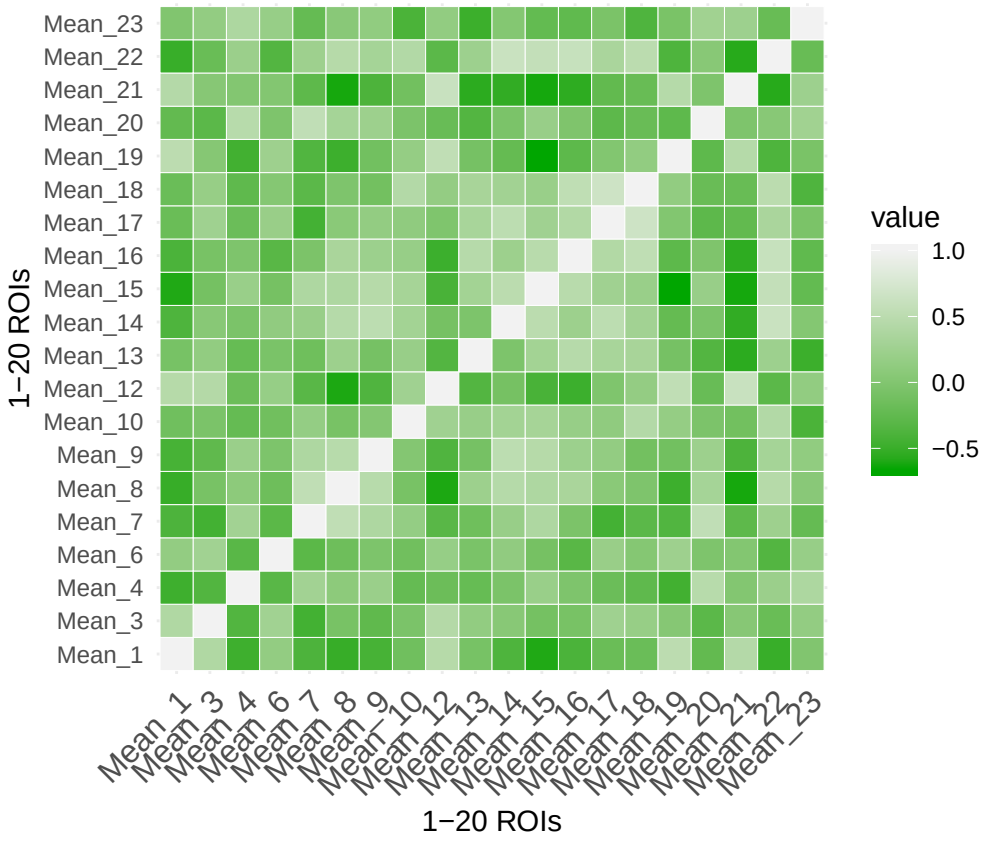
```
dim(MyData)
```

```
## [1] 172 192
```

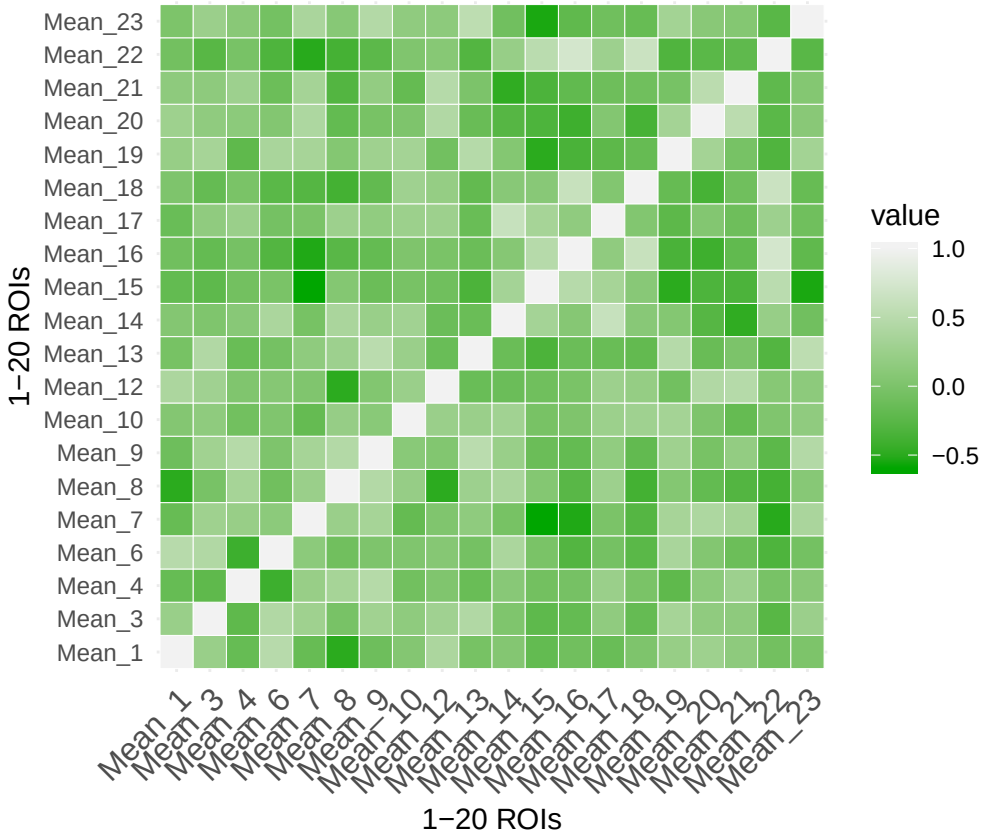
```
MyData[1:5,1:5]
```

```
## # A tibble: 5 x 5
##   File                                `Sub-brick`   Mean_1   Mean_3   Mean_4
##   <chr>                               <chr>          <dbl>    <dbl>    <dbl>
## 1 sfnwmrda0010002_session_1_rest_1.~ 0[?]         0.0155   0.0432  -1.39e-4
## 2 sfnwmrda0010002_session_1_rest_1.~ 1[?]        -0.0207   0.0365  -2.52e-2
## 3 sfnwmrda0010002_session_1_rest_1.~ 2[?]        -0.0541   0.00426 -7.34e-2
## 4 sfnwmrda0010002_session_1_rest_1.~ 3[?]        -0.0466  -0.0444  -1.14e-1
## 5 sfnwmrda0010002_session_1_rest_1.~ 4[?]         0.00670 -0.0862  -1.07e-1
```

```
half_data1 = MyData[1:86,3:22]
half_data2 = MyData[87:172,3:22]
corr_data1 = cor(half_data1)
corr_data2 = cor(half_data2)
melt_data1 = melt(corr_data1)
melt_data2 = melt(corr_data2)
ggplot(data = melt_data1, aes(Var2, Var1, fill = value))+
  xlab('1-20 ROIs')+ylab("1-20 ROIs")+
  geom_tile(color = "white")+
  scale_fill_gradientn(colours = terrain.colors(2)) +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1))+
  coord_fixed()
```




```
ggplot(data = melt_data2, aes(Var2, Var1, fill = value))+
  xlab('1-20 ROIs')+ylab("1-20 ROIs")+
  geom_tile(color = "white")+
  scale_fill_gradientn(colours = terrain.colors(2)) +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1))+
  coord_fixed()
```



I also apply a similar procedure to a typical-developed individual 1000804 (DX=0), the heatmaps of the first 20 ROIs for it are shown below:

```
MyData <- read_csv(file="/Users/HanY/Desktop/ADHD/ADHD200_CC200_TCs_filtfix/NYU/1000804/sfnwmrda1000804_
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   File = col_character(),
##   `Sub-brick` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

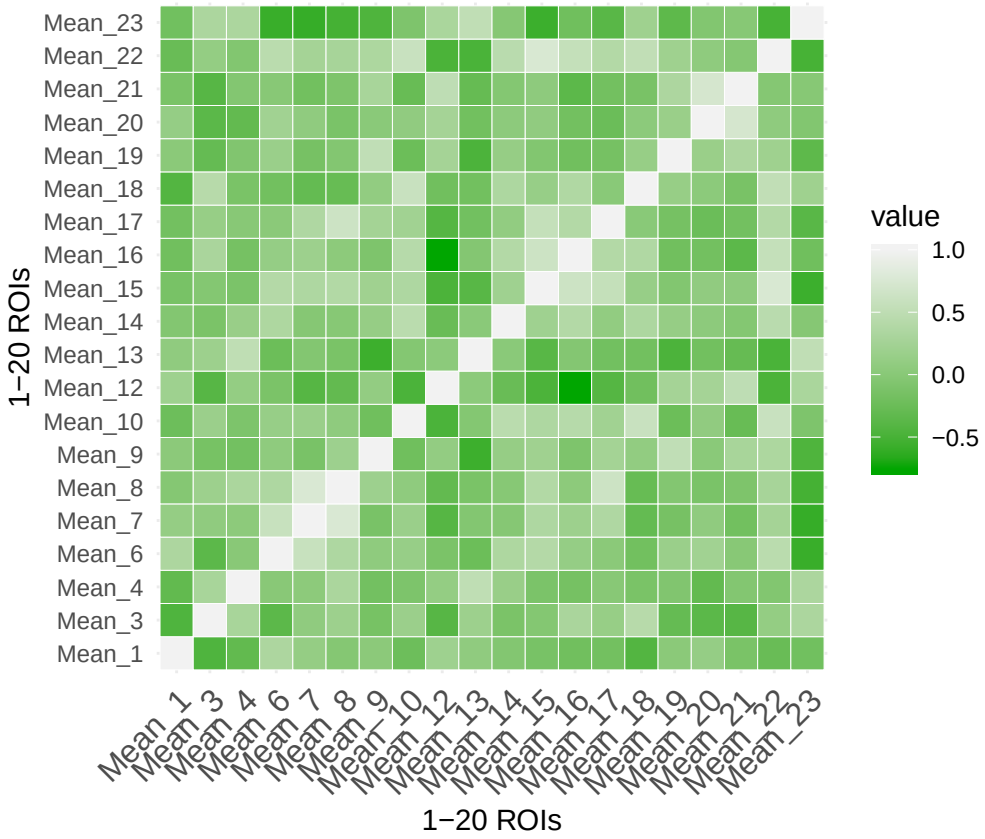
```
dim(MyData)
```

```
## [1] 172 192
```

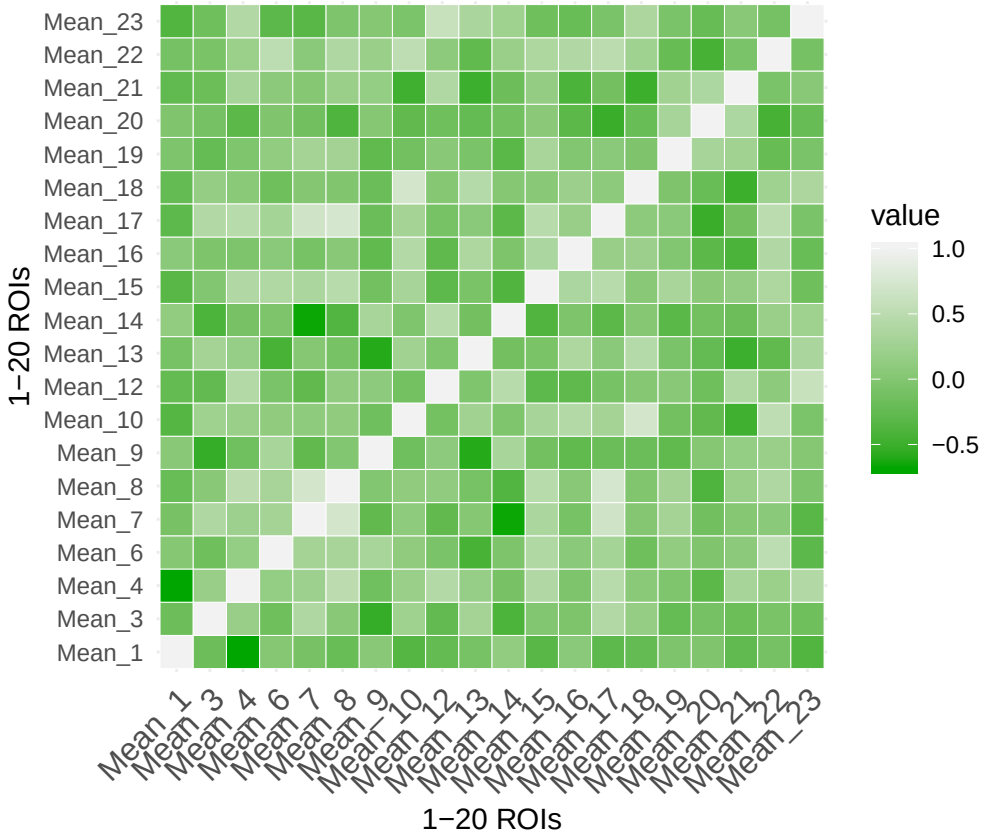
```
MyData[1:5,1:5]
```

```
## # A tibble: 5 x 5
##   File                                  `Sub-brick`  Mean_1 Mean_3  Mean_4
##   <chr>                                 <chr>         <dbl>  <dbl>   <dbl>
## 1 sfnwmrda1000804_session_1_rest_1.nii.~ 0[?]         0.202  -0.197  0.172
## 2 sfnwmrda1000804_session_1_rest_1.nii.~ 1[?]         0.132  -0.292  0.220
## 3 sfnwmrda1000804_session_1_rest_1.nii.~ 2[?]         0.0335 -0.299  0.189
## 4 sfnwmrda1000804_session_1_rest_1.nii.~ 3[?]        -0.0330 -0.219  0.0824
## 5 sfnwmrda1000804_session_1_rest_1.nii.~ 4[?]        -0.0288 -0.104 -0.0512
```

```
half_data1 = MyData[1:86,3:22]
half_data2 = MyData[87:172,3:22]
corr_data1 = cor(half_data1)
corr_data2 = cor(half_data2)
melt_data1 = melt(corr_data1)
melt_data2 = melt(corr_data2)
ggplot(data = melt_data1, aes(Var2, Var1, fill = value))+
  xlab('1-20 ROIs')+ylab("1-20 ROIs")+
  geom_tile(color = "white")+
  scale_fill_gradientn(colours = terrain.colors(2)) +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1))+
  coord_fixed()
```




```
ggplot(data = melt_data2, aes(Var2, Var1, fill = value))+
  xlab('1-20 ROIs')+ylab("1-20 ROIs")+
  geom_tile(color = "white")+
  scale_fill_gradientn(colours = terrain.colors(2)) +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1))+
  coord_fixed()
```



It can be seen clearly in the heatmaps that the correlations among the first 20 ROIs vary a lot with time going. So it may be reasonable for us to capture the dynamic property of the data.

4 Dynamic Model for Covaricance Matrices

Chen(2016) has proposed an effective dynamic model to estimation the dynamic covariance matrix. Let $Y = (Y_1, \cdots, Y_p)^T$ be a p-dimensional random vector and U denote time. Suppose that $\{Y_i, U_i\}$ with $Y_i = (Y_{i1}, \cdots, Y_{ip})^T$ is a random sample from the population $\{Y, U\}$, for $i = 1, \cdots, n$. The empirical sample conditional covariance matrix based on kernel smoothing is

$$\begin{aligned}\hat{\Sigma}(u) := &\left\{\sum_{i=1}^{n} K_h(U_i - u) Y_i Y_i^T\right\} \cdot \left\{\sum_{i=1}^{n} K_h(U_i - u)\right\}^{-1} \\ &- \left\{\sum_{i=1}^{n} K_h(U_i - u) Y_i\right\} \left\{\sum_{i=1}^{n} K_h(U_i - u) Y_i^T\right\} \\ &\times \left\{\sum_{i=1}^{n} K_h(U_i - u)\right\}^{-2}\end{aligned}$$

Under the model from Chen(2016), we plot some correlations of individual 1023964 (DX=3):

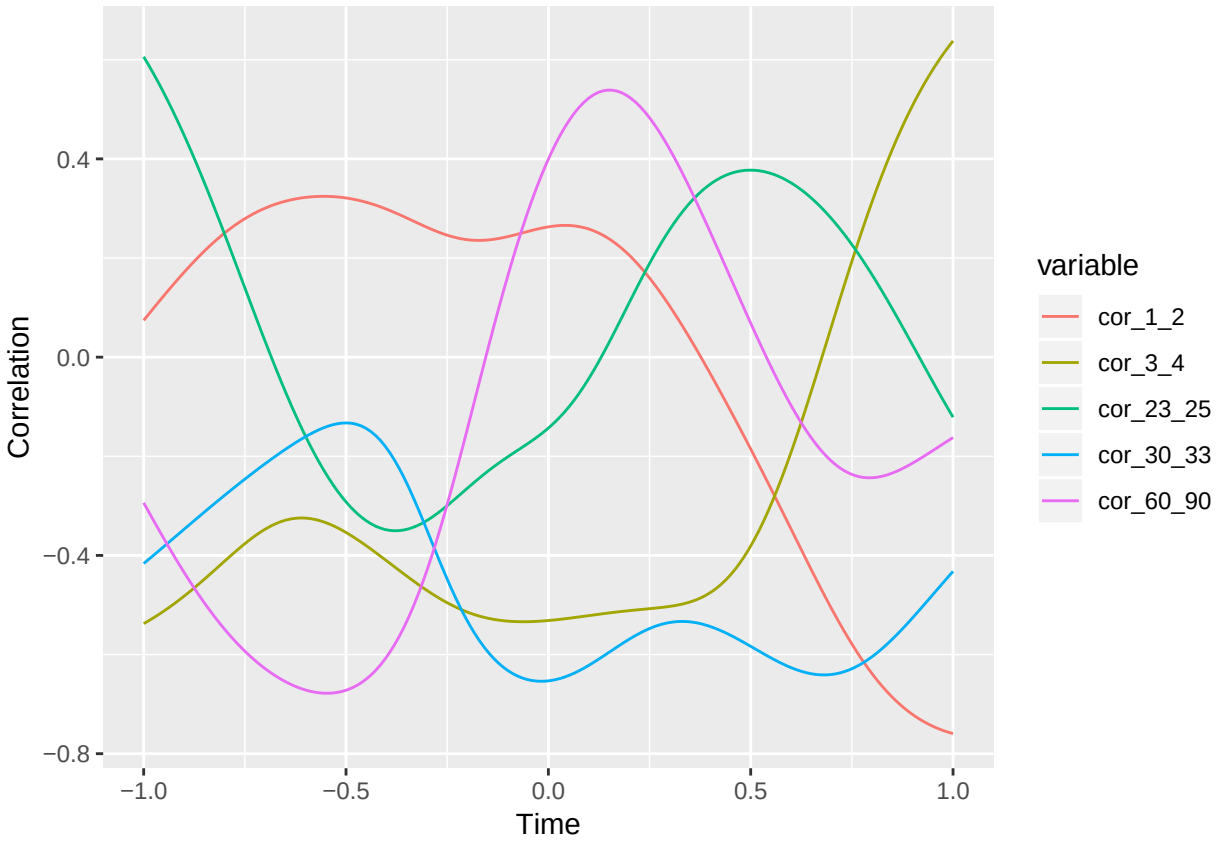
```
n = 172;
p = 190;
U = seq(-1,1,length.out = n)
h = 0.2
thd = 2.5 * sqrt(log(p)/(n*h))
```

```
kern <- function(x, h, type = "normal"){ return(dnorm(x/h)/h)}
dcmEst <- function(Y, u_pred){
  ## double loop, rows and columns of covariance matrix
  dcmVar <- matrix(0, p, p)
  Y_prod_k <- matrix(0, p, p)
  kernel <- kern(U - u_pred, h)
  for(i in 1:n) {
      Y_prod_k = Y_prod_k + kernel[i] * outer( Y[i,],  Y[i,] )
  }
  Yk <- matrix(kernel,1) %*% Y
  dcmVar <- Y_prod_k / sum(kernel)
            - outer(Yk,Yk) / (sum(kernel)) ^ 2
  return(dcmVar)
}
corEst <- function(Y, i, j){
  corE = rep(0,n)
  for(k in 1:n){
    corE[k] = cov2cor(dcmEst(Y,U[k]))[i,j]
  }
  return(corE)
}
NYU1023964_3 = read_csv(file = "/Users/HanY/Desktop/ADHD/ADHD200_CC200_TCs_filtfix/NYU/1023964/sfnwmrda
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   File = col_character(),
##   `Sub-brick` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
NYU1023964_3 = as.matrix(NYU1023964_3[,3:192])
cor_1_2 = corEst(NYU1023964_3,1,2)
cor_3_4 = corEst(NYU1023964_3,3,4)
cor_23_25 = corEst(NYU1023964_3,23,25)
cor_30_33 = corEst(NYU1023964_3,30,33)
cor_60_90 = corEst(NYU1023964_3,60,90)
ggplot(melt(data.frame(cor_1_2,cor_3_4,cor_23_25,cor_30_33,cor_60_90,U),id.vars = "U"), aes(U, value),
  geom_line(aes(color=variable)) +
  xlab('Time')+ylab("Correlation")
```



From what has been discussed above, we can safely draw a conclusion that correlations among each ROIs do vary a lot with time. It will be of great value if we can apply some dynamic model to capture the dynamicity of the covariance so that it can be applied to help settle some intractable problems in various fields.

5 Future Expectation

It is expected that voxels of some or all ROIs may vary with DX and have similarity among people with the same sympotom. This will be an open problem. To get some inspiration, the scatter plots of absolute sums of individuals 1000804 (DX=0) and 0010005 (DX=2) are shown below

```
absfun <- function(x){
  sum(abs(x))
}
NYU1000804_0 = as.matrix(NYU1000804_0[,3:192])
sumNYU1000804_0 = apply(NYU1000804_0,2,absfun)

NYU1023964_3 = read_csv(file = "/Users/HanY/Desktop/ADHD/ADHD200_CC200_TCs_filtfix/NYU/1023964/sfnwmrda
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   File = col_character(),
##   `Sub-brick` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

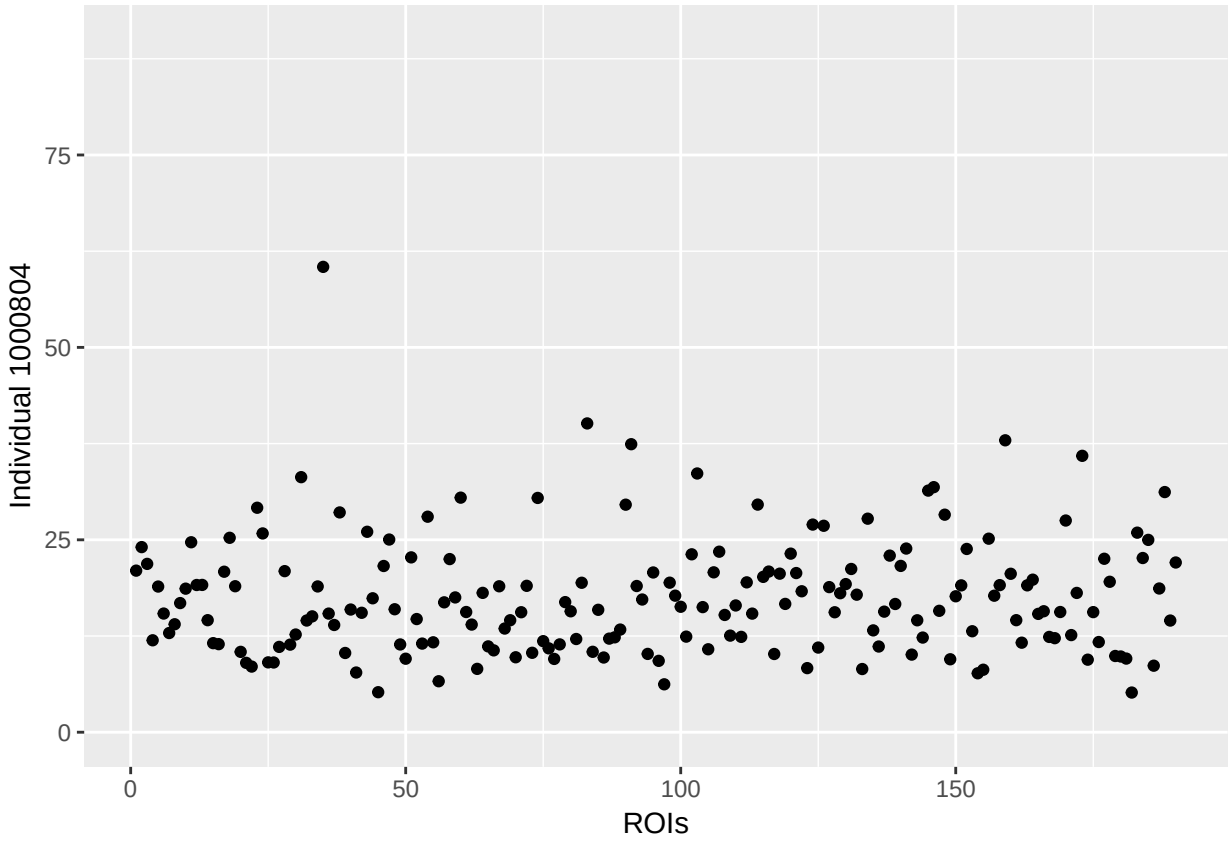
```
NYU1023964_3 = as.matrix(NYU1023964_3[,3:192])
sumNYU1023964_3 = apply(NYU1023964_3,2,absfun)

NYU0010005_2 = read_csv(file = "/Users/HanY/Desktop/ADHD/ADHD200_CC200_TCs_filtfix/NYU/0010005/sfnwmrda
```

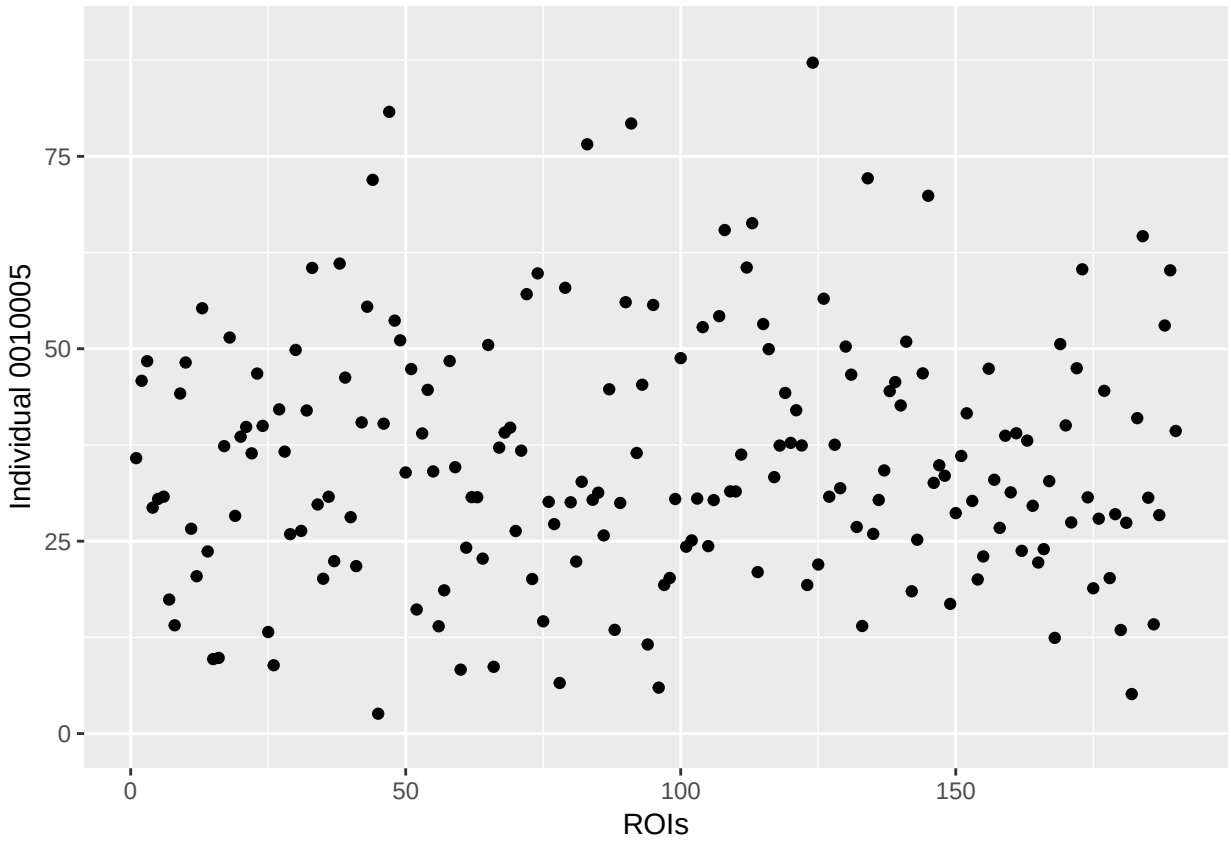
```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   File = col_character(),
##   `Sub-brick` = col_character()
## )
## See spec(...) for full column specifications.
```

```
NYU0010005_2 = as.matrix(NYU0010005_2[,3:192])
sumNYU0010005_2 = apply(NYU0010005_2,2,absfun)

ggplot(data = melt(data.frame(sumNYU1000804_0,c(1:p)),id.vars = 2), aes(c(1:p), sumNYU1000804_0)) +
  geom_point() +
  xlab('ROIs')+ylab("Individual 1000804") +
  ylim(0,90)
```

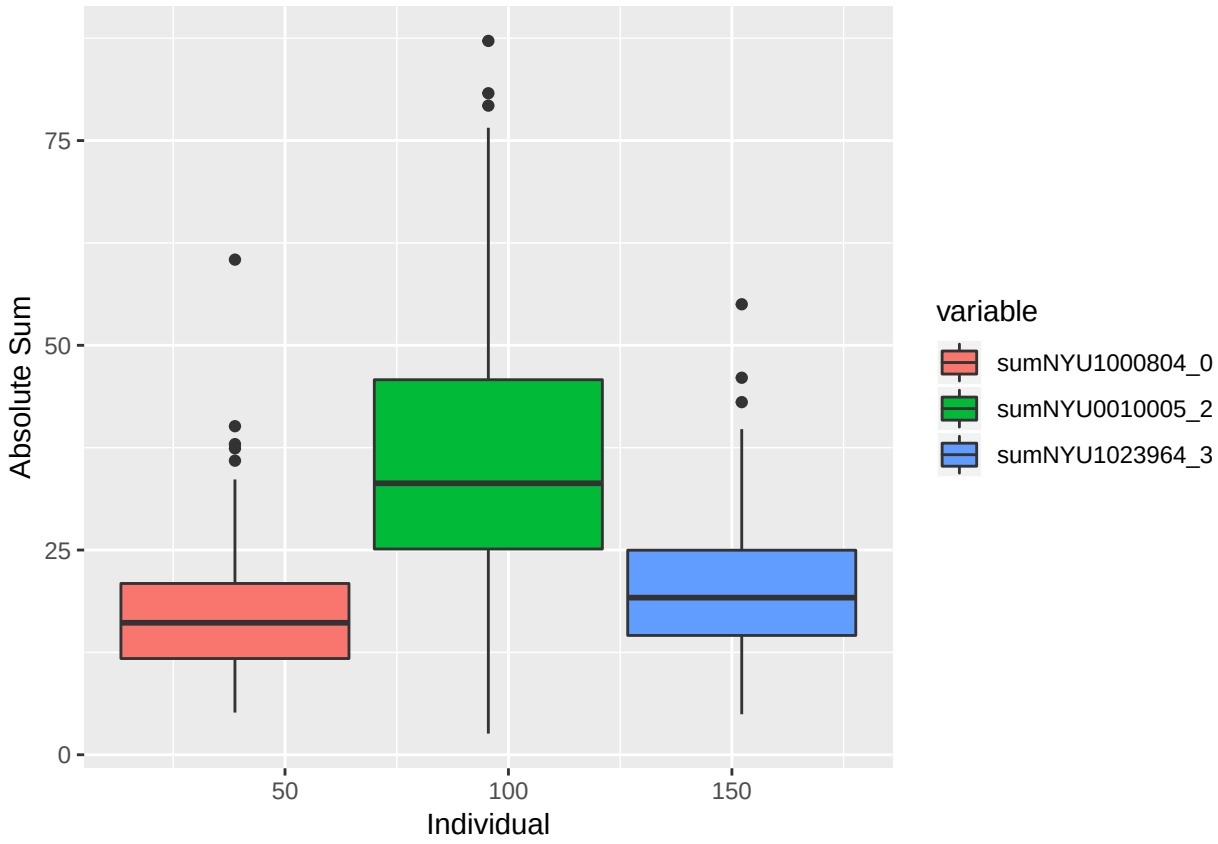



```
ggplot(data = melt(data.frame(sumNYU0010005_2,c(1:p)),id.vars = 2), aes(c(1:p), sumNYU0010005_2)) +
  geom_point() +
  xlab('ROIs')+ylab("Individual 0010005") +
  ylim(0,90)
```



Boxplots for absolute sums of individuals 1000804 (DX=0), 0010005 (DX=2) and 1023964 (DX=3) are shown below

```
ggplot(melt(data.frame(sumNYU1000804_0,sumNYU0010005_2,sumNYU1023964_3,y = 1:p),id.var = 'y'), aes(y, va
  geom_boxplot(aes(fill=variable)) +
  xlab('Individual')+ylab('Absolute Sum')
```



From what has been shown in this section, it is reasonable to futher classify people with different symptoms.