*Final Doc P2*

Gabrielle Salamanca

August 14, 2023

This document is Part 2 of the final doc research. This will explain the work with the full dataset but with Y1. Everything is the same up until the columns part.

### *SKAT*

Now after the columns, we will make three subsets based on Y1. This had caused a problem when trying to run the for loops, which we’ll dive in during that part. Right now, we need to fill in the NAs for Y1.

Now, we will make three subsets and null models based on Y1’s values: active, recovered, and healthy. The reason why we’re doing three pairs is because SKATBinary only takes 1’s and 0’s. So the pairs are: active/recovered (actRec), active/healthy (actHea), and recovered/healthy (recHea). The 1 is active for actRec, active for actHea, and recovered for recHea.

Finally, we can run the for loops. Now, as I mentioned, there were problems I ran into when trying to run them initially. So, I did each one individually to see what were the warnings that were popping up. The first one was this: **missing value where TRUE/FALSE needed**. I’m not sure what this meant, but the rest of the warnings had to do with dimension problems.

The dimension problem possibly occurred, because the original subsets had only the gene dataset in mind, so that had the for loop stopping at one p-value. So, after some fiddling from the professor, she figured out that I needed add in a column index for the subset to finally run smoothly and fix the dimension problem.

To note, the p-values must be less than .

## Active/Recovered

Here’s the v gene for actRec:

## vgene.idx pvalue  
## 1 1 2.397077e-05  
## 2 2 4.158143e-02  
## 3 3 7.755451e-05  
## 4 4 2.273850e-06  
## 5 5 2.734692e-04  
## 6 6 1.240747e-01  
## 7 7 4.201495e-01  
## 8 8 7.854085e-07  
## 9 9 1.000000e+00  
## 10 10 1.086708e-02  
## 11 11 3.908792e-03  
## 12 12 1.906858e-03  
## 13 13 1.533240e-05  
## 14 14 4.757829e-04  
## 15 15 1.883714e-06  
## 16 16 1.863261e-06  
## 17 17 1.595830e-03  
## 18 18 4.144692e-03  
## 19 19 9.087526e-03  
## 20 20 8.627452e-05  
## 21 21 5.469046e-04  
## 22 22 3.790496e-05  
## 23 23 2.265788e-02  
## 24 24 1.111375e-04  
## 25 25 3.951094e-02  
## 26 26 1.000000e+00  
## 27 27 7.528594e-02  
## 28 28 1.762465e-01  
## 29 29 2.189914e-04  
## 30 30 9.109265e-05  
## 31 31 1.535079e-05  
## 32 32 2.987368e-03  
## 33 33 9.139160e-05  
## 34 34 7.006954e-04  
## 35 35 1.592233e-06  
## 36 36 8.379929e-02  
## 37 37 6.488731e-05  
## 38 38 5.968856e-01  
## 39 39 2.984939e-03  
## 40 40 2.834240e-07  
## 41 41 4.769847e-05  
## 42 42 2.540612e-05  
## 43 43 1.979830e-01  
## 44 44 1.485182e-04  
## 45 45 4.825608e-04  
## 46 46 3.327397e-04  
## 47 47 1.282150e-03  
## 48 48 8.707685e-07  
## 49 49 7.675289e-03  
## 50 50 6.241048e-02

The p-values are far better when we do these kinds of pairs. The notable p-values are: colv1, colv2, colv3, colv4, colv5, colv6, colv7, colv8, colv10, colv11, colv12, colv13, colv14, colv15, colv16, colv17, colv18, colv20, colv21, colv22, colv23, colv24, colv25, colv27, colv28, colv29, colv30, colv31, colv32, colv33, colv34, colv35, colv36 colv37, colv38, colv39, colv40, colv41, colv42, colv43, colv44, colv45, colv46, colv47, colv48, colv49, and colv50.

Here’s the j gene for actRec:

## jgene.idx p-value  
## 1 1 7.525604e-08  
## 2 2 6.960908e-07  
## 3 3 1.484912e-06  
## 4 4 1.598473e-07  
## 5 5 1.339650e-05  
## 6 6 3.059248e-06  
## 7 7 1.162100e-06  
## 8 8 1.303055e-06  
## 9 9 2.665907e-06  
## 10 10 7.264480e-08  
## 11 11 1.059571e-05  
## 12 12 1.345202e-08  
## 13 13 3.884085e-06

Here, all the j genes were notable.

## 

## Active/Healthy

Here’s the v gene for actHea:

## vgene.idx pvalue  
## 1 1 0.23682149  
## 2 2 1.00000000  
## 3 3 1.00000000  
## 4 4 1.00000000  
## 5 5 0.94804748  
## 6 6 0.24017458  
## 7 7 0.77870961  
## 8 8 1.00000000  
## 9 9 1.00000000  
## 10 10 0.10412652  
## 11 11 0.09321457  
## 12 12 0.37420537  
## 13 13 0.01932227  
## 14 14 1.00000000  
## 15 15 1.00000000  
## 16 16 0.37456487  
## 17 17 1.00000000  
## 18 18 0.52327622  
## 19 19 1.00000000  
## 20 20 1.00000000  
## 21 21 1.00000000  
## 22 22 1.00000000  
## 23 23 0.84539219  
## 24 24 1.00000000  
## 25 25 1.00000000  
## 26 26 1.00000000  
## 27 27 0.10898509  
## 28 28 0.30954129  
## 29 29 1.00000000  
## 30 30 1.00000000  
## 31 31 1.00000000  
## 32 32 1.00000000  
## 33 33 1.00000000  
## 34 34 0.09927081  
## 35 35 1.00000000  
## 36 36 0.05624608  
## 37 37 0.01042770  
## 38 38 0.59203424  
## 39 39 0.02868995  
## 40 40 1.00000000  
## 41 41 1.00000000  
## 42 42 1.00000000  
## 43 43 0.36253066  
## 44 44 1.00000000  
## 45 45 1.00000000  
## 46 46 1.00000000  
## 47 47 1.00000000  
## 48 48 1.00000000  
## 49 49 1.00000000  
## 50 50 0.10923141

The notable p-values are: colv13, colv36, colv37, colv39.

Here’s the j gene for actHea:

## jgene.idx p-value  
## 1 1 0.40897049  
## 2 2 0.10922864  
## 3 3 0.03129443  
## 4 4 0.33987566  
## 5 5 0.40275036  
## 6 6 0.33931857  
## 7 7 0.58933709  
## 8 8 0.02091644  
## 9 9 0.48938937  
## 10 10 0.01843833  
## 11 11 1.00000000  
## 12 12 0.09061047  
## 13 13 0.07906318

Here, the notable j genes are: colj3, colj8, and colj10.

## Recovered/Healthy

Here’s the v gene for recHea:

## vgene.idx pvalue  
## 1 1 1.210706e-02  
## 2 2 1.285203e-01  
## 3 3 2.408672e-04  
## 4 4 2.889555e-03  
## 5 5 1.220039e-02  
## 6 6 1.270984e-01  
## 7 7 2.465860e-01  
## 8 8 2.025142e-03  
## 9 9 6.810633e-02  
## 10 10 5.390648e-06  
## 11 11 2.608766e-02  
## 12 12 6.035357e-03  
## 13 13 1.451353e-01  
## 14 14 3.625424e-03  
## 15 15 1.301455e-04  
## 16 16 8.456416e-06  
## 17 17 4.053111e-03  
## 18 18 5.899806e-04  
## 19 19 2.988405e-05  
## 20 20 3.362903e-02  
## 21 21 1.383218e-02  
## 22 22 5.915148e-02  
## 23 23 1.162577e-02  
## 24 24 2.017836e-04  
## 25 25 4.220383e-02  
## 26 26 4.401381e-01  
## 27 27 1.596282e-05  
## 28 28 2.152684e-01  
## 29 29 7.445946e-01  
## 30 30 5.801731e-06  
## 31 31 8.160436e-05  
## 32 32 2.476571e-02  
## 33 33 1.281031e-04  
## 34 34 3.394787e-01  
## 35 35 6.119150e-02  
## 36 36 3.778931e-04  
## 37 37 2.702099e-01  
## 38 38 6.282658e-01  
## 39 39 1.542016e-01  
## 40 40 5.696814e-04  
## 41 41 9.435286e-03  
## 42 42 2.774806e-03  
## 43 43 2.661045e-01  
## 44 44 1.019783e-01  
## 45 45 4.221335e-01  
## 46 46 1.569324e-03  
## 47 47 1.404360e-03  
## 48 48 2.501305e-01  
## 49 49 5.080258e-05  
## 50 50 1.745098e-02

The notable p-values are all of them.

Here’s the j gene for recHea:

## jgene.idx p-value  
## 1 1 5.180186e-05  
## 2 2 1.555973e-03  
## 3 3 5.555566e-03  
## 4 4 1.637188e-03  
## 5 5 1.304339e-01  
## 6 6 2.687000e-03  
## 7 7 4.655433e-03  
## 8 8 4.948688e-02  
## 9 9 8.194130e-04  
## 10 10 1.895542e-05  
## 11 11 7.903545e-02  
## 12 12 2.529557e-05  
## 13 13 1.560728e-04

Here, all the j genes are notable.

Now, let’s do the p-value adjustments on each one.

## actRec: P-value Adjustment

Here’s for the actRec’s v genes:

## vgene.idx p-value  
## 1 1 9.828014e-04  
## 2 2 4.741313e-01  
## 3 3 2.791962e-03  
## 4 4 1.000494e-04  
## 5 5 7.930608e-03  
## 6 6 8.685228e-01  
## 7 7 1.000000e+00  
## 8 8 3.848502e-05  
## 9 9 1.000000e+00  
## 10 10 1.521392e-01  
## 11 11 7.035825e-02  
## 12 12 4.004402e-02  
## 13 13 6.592930e-04  
## 14 14 1.284614e-02  
## 15 15 8.571001e-05  
## 16 16 8.571001e-05  
## 17 17 3.510826e-02  
## 18 18 7.045976e-02  
## 19 19 1.363129e-01  
## 20 20 3.019608e-03  
## 21 21 1.367261e-02  
## 22 22 1.478293e-03  
## 23 23 2.945524e-01  
## 24 24 3.556400e-03  
## 25 25 4.741313e-01  
## 26 26 1.000000e+00  
## 27 27 6.775735e-01  
## 28 28 1.000000e+00  
## 29 29 6.569741e-03  
## 30 30 3.097150e-03  
## 31 31 6.592930e-04  
## 32 32 5.969878e-02  
## 33 33 3.097150e-03  
## 34 34 1.681669e-02  
## 35 35 7.483495e-05  
## 36 36 6.775735e-01  
## 37 37 2.400830e-03  
## 38 38 1.000000e+00  
## 39 39 5.969878e-02  
## 40 40 1.417120e-05  
## 41 41 1.812542e-03  
## 42 42 1.016245e-03  
## 43 43 1.000000e+00  
## 44 44 4.604063e-03  
## 45 45 1.284614e-02  
## 46 46 9.316711e-03  
## 47 47 2.948945e-02  
## 48 48 4.179689e-05  
## 49 49 1.228046e-01  
## 50 50 6.241048e-01

The notable v gene columns are: 1, 2, 3, 4, 5, 6, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 27, 29, 30, 31, 32, 33, 34, 35, 36, 37, 39, 40, 41, 42, 44, 45, 46, 47, 48, 49, and 50.

And here’s for the j genes:

## jgene.idx p-value  
## 1 1 8.717376e-07  
## 2 2 6.264817e-06  
## 3 3 9.296800e-06  
## 4 4 1.598473e-06  
## 5 5 2.119142e-05  
## 6 6 1.332954e-05  
## 7 7 9.296800e-06  
## 8 8 9.296800e-06  
## 9 9 1.332954e-05  
## 10 10 8.717376e-07  
## 11 11 2.119142e-05  
## 12 12 1.748762e-07  
## 13 13 1.332954e-05

The notable j gene columns are all of them.

## actHea: P-value Adjustment

Here’s for the actHea’s v genes:

## vgene.idx p-value  
## 1 1 1.0000000  
## 2 2 1.0000000  
## 3 3 1.0000000  
## 4 4 1.0000000  
## 5 5 1.0000000  
## 6 6 1.0000000  
## 7 7 1.0000000  
## 8 8 1.0000000  
## 9 9 1.0000000  
## 10 10 1.0000000  
## 11 11 1.0000000  
## 12 12 1.0000000  
## 13 13 0.9467912  
## 14 14 1.0000000  
## 15 15 1.0000000  
## 16 16 1.0000000  
## 17 17 1.0000000  
## 18 18 1.0000000  
## 19 19 1.0000000  
## 20 20 1.0000000  
## 21 21 1.0000000  
## 22 22 1.0000000  
## 23 23 1.0000000  
## 24 24 1.0000000  
## 25 25 1.0000000  
## 26 26 1.0000000  
## 27 27 1.0000000  
## 28 28 1.0000000  
## 29 29 1.0000000  
## 30 30 1.0000000  
## 31 31 1.0000000  
## 32 32 1.0000000  
## 33 33 1.0000000  
## 34 34 1.0000000  
## 35 35 1.0000000  
## 36 36 1.0000000  
## 37 37 0.5213850  
## 38 38 1.0000000  
## 39 39 1.0000000  
## 40 40 1.0000000  
## 41 41 1.0000000  
## 42 42 1.0000000  
## 43 43 1.0000000  
## 44 44 1.0000000  
## 45 45 1.0000000  
## 46 46 1.0000000  
## 47 47 1.0000000  
## 48 48 1.0000000  
## 49 49 1.0000000  
## 50 50 1.0000000

The notable v gene column is 37.

And here’s for the j genes:

## jgene.idx p-value  
## 1 1 1.0000000  
## 2 2 0.8738291  
## 3 3 0.3442387  
## 4 4 1.0000000  
## 5 5 1.0000000  
## 6 6 1.0000000  
## 7 7 1.0000000  
## 8 8 0.2509973  
## 9 9 1.0000000  
## 10 10 0.2396982  
## 11 11 1.0000000  
## 12 12 0.8154943  
## 13 13 0.7906318

The notable j gene columns are none.

## recHea: P-value Adjustment

Here’s for the recHea’s v genes:

## vgene.idx p-value  
## 1 1 0.3147835245  
## 2 2 1.0000000000  
## 3 3 0.0096346899  
## 4 4 0.0924657552  
## 5 5 0.3147835245  
## 6 6 1.0000000000  
## 7 7 1.0000000000  
## 8 8 0.0688548116  
## 9 9 1.0000000000  
## 10 10 0.0002695324  
## 11 11 0.5478409555  
## 12 12 0.1750253574  
## 13 13 1.0000000000  
## 14 14 0.1123881574  
## 15 15 0.0055084316  
## 16 16 0.0004059080  
## 17 17 0.1215933157  
## 18 18 0.0218292816  
## 19 19 0.0013746662  
## 20 20 0.6725806227  
## 21 21 0.3319724134  
## 22 22 1.0000000000  
## 23 23 0.3138957817  
## 24 24 0.0082731264  
## 25 25 0.8018728223  
## 26 26 1.0000000000  
## 27 27 0.0007502527  
## 28 28 1.0000000000  
## 29 29 1.0000000000  
## 30 30 0.0002842848  
## 31 31 0.0035905917  
## 32 32 0.5448456264  
## 33 33 0.0055084316  
## 34 34 1.0000000000  
## 35 35 1.0000000000  
## 36 36 0.0147378326  
## 37 37 1.0000000000  
## 38 38 1.0000000000  
## 39 39 1.0000000000  
## 40 40 0.0216478928  
## 41 41 0.2641880181  
## 42 42 0.0915685846  
## 43 43 1.0000000000  
## 44 44 1.0000000000  
## 45 45 1.0000000000  
## 46 46 0.0549263548  
## 47 47 0.0505569428  
## 48 48 1.0000000000  
## 49 49 0.0022861161  
## 50 50 0.4013726129

The notable v gene columns are: 3, 10, 15, 16, 18, 19, 24, 27, 30, 31, 33, 36, 40, 46, 47, and 49.

And here’s for the j genes:

## jgene.idx p-value  
## 1 1 0.0005698204  
## 2 2 0.0124477844  
## 3 3 0.0232771665  
## 4 4 0.0124477844  
## 5 5 0.1580709041  
## 6 6 0.0161220009  
## 7 7 0.0232771665  
## 8 8 0.1484606508  
## 9 9 0.0073747167  
## 10 10 0.0002464205  
## 11 11 0.1580709041  
## 12 12 0.0003035469  
## 13 13 0.0015607281

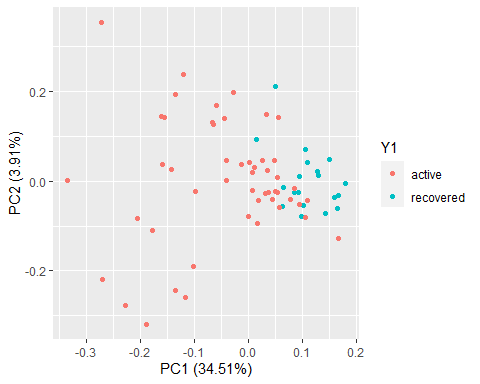
The notable j gene columns are: 1, 2, 3, 4, 6, 7, 9, 10, 12, and 13.

### 

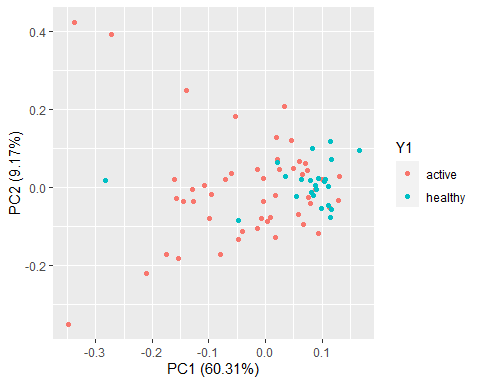
### *PCA Plots*

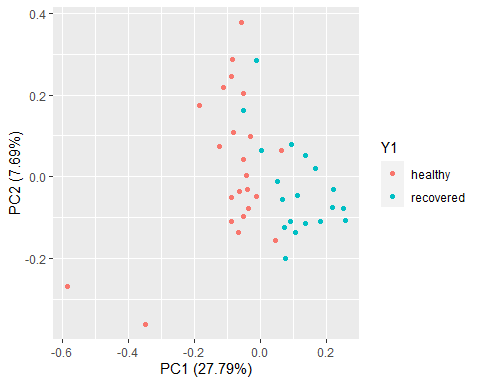
Now, we will make PCA plots for each subset and the entire dataset to compare. We will use the p-adjusted values for this one. As usual, we dio some prep for the plots before we can do the v and j gene plots for each pair.

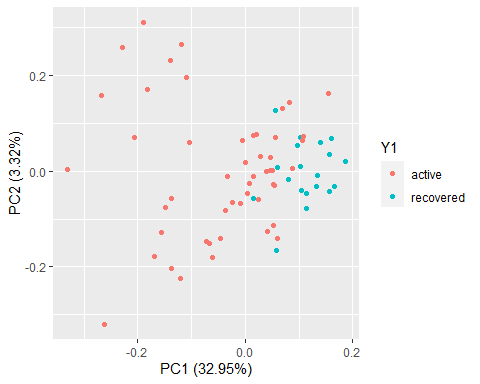
**PCA: Active vs Recovered (V Gene)**

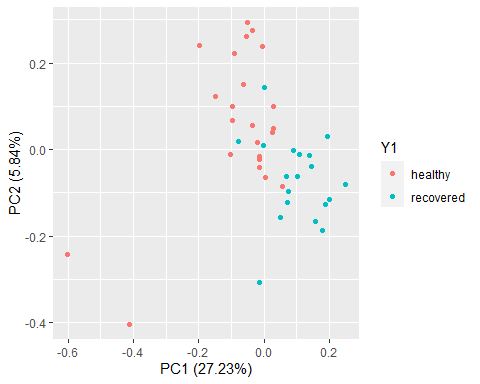


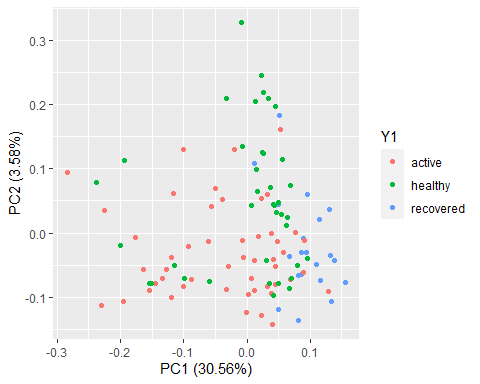
**PCA: Active vs Healthy (V Gene)**



**PCA: Recovered vs Healthy (V Gene)**

**PCA: Active vs Recovered (J Gene)**

**PCA: Recovered vs Healthy (J Gene)**

**PCA: Full Y1 Genes**

**Code Appendix**

knitr::opts\_chunk$set(echo = FALSE)  
library(dplyr)  
library(factoextra)  
library(ggfortify)  
library(ggplot2)  
library(gridExtra)  
library(psych)  
library(readr)  
library(readxl)  
library(SKAT)  
library(tidyr)  
gene <- read\_excel("~/TCR-Project/Datasets/Disease Data/fullgenes.xlsx")  
attach(gene)  
# v gene  
stringv1 <- "TRBV10-1"  
stringv2 <- "TRBV10-2"  
stringv3 <- "TRBV10-3"  
stringv4 <- "TRBV11-1"  
stringv5 <- "TRBV11-2"  
stringv6 <- "TRBV11-3"  
stringv7 <- "TRBV12-1"  
stringv8 <- "TRBV12-3"  
stringv9 <- "TRBV12-4"  
stringv10 <- "TRBV12-5"  
  
stringv11 <- "TRBV13"  
stringv12 <- "TRBV14"  
stringv13 <- "TRBV15"  
stringv14 <- "TRBV18"  
stringv15 <- "TRBV19"  
stringv16 <- "TRBV2"  
stringv17 <- "TRBV20-1"  
stringv18 <- "TRBV21-1"  
stringv19 <- "TRBV23-1"  
stringv20 <- "TRBV24-1"  
  
stringv21 <- "TRBV25-1"  
stringv22 <- "TRBV27"  
stringv23 <- "TRBV28"  
stringv24 <- "TRBV29-1"  
stringv25 <- "TRBV3-2"  
stringv26 <- "TRBV30"  
stringv27 <- "TRBV4-1"  
stringv28 <- "TRBV4-2"  
stringv29 <- "TRBV4-3"  
stringv30 <- "TRBV5-1"  
  
stringv31 <- "TRBV5-3"  
stringv32 <- "TRBV5-4"  
stringv33 <- "TRBV5-5"  
stringv34 <- "TRBV5-6"  
stringv35 <- "TRBV5-7"  
stringv36 <- "TRBV5-8"  
stringv37 <- "TRBV6-1"  
stringv38 <- "TRBV6-2"  
stringv39 <- "TRBV6-3"  
stringv40 <- "TRBV6-4"  
  
stringv41 <- "TRBV6-5"  
stringv42 <- "TRBV6-6"  
stringv43 <- "TRBV6-7"  
stringv44 <- "TRBV6-8"  
stringv45 <- "TRBV6-9"  
stringv46 <- "TRBV7-2"  
stringv47 <- "TRBV7-3"  
stringv48 <- "TRBV7-4"  
stringv49 <- "TRBV7-5"  
stringv50 <- "TRBV7-6"  
  
# j gene  
stringj1 <- "TRBJ1-1"  
stringj2 <- "TRBJ1-2"  
stringj3 <- "TRBJ1-3"  
stringj4 <- "TRBJ1-4"  
stringj5 <- "TRBJ1-5"  
stringj6 <- "TRBJ1-6"  
stringj7 <- "TRBJ2-1"  
stringj8 <- "TRBJ2-2"  
stringj9 <- "TRBJ2-3"  
stringj10 <- "TRBJ2-4"  
  
stringj11 <- "TRBJ2-5"  
stringj12 <- "TRBJ2-6"  
stringj13 <- "TRBJ2-7"  
# v gene  
colv1 <- grep(stringv1, names(gene), value = TRUE)  
colv2 <- grep(stringv2, names(gene), value = TRUE)  
colv3 <- grep(stringv3, names(gene), value = TRUE)  
colv4 <- grep(stringv4, names(gene), value = TRUE)  
colv5 <- grep(stringv5, names(gene), value = TRUE)  
colv6 <- grep(stringv6, names(gene), value = TRUE)  
colv7 <- grep(stringv7, names(gene), value = TRUE)  
colv8 <- grep(stringv8, names(gene), value = TRUE)  
colv9 <- grep(stringv9, names(gene), value = TRUE)  
colv10 <- grep(stringv10, names(gene), value = TRUE)  
  
colv11 <- grep(stringv11, names(gene), value = TRUE)  
colv12 <- grep(stringv12, names(gene), value = TRUE)  
colv13 <- grep(stringv13, names(gene), value = TRUE)  
colv14 <- grep(stringv14, names(gene), value = TRUE)  
colv15 <- grep(stringv15, names(gene), value = TRUE)  
colv16 <- grep(stringv16, names(gene), value = TRUE)  
colv17 <- grep(stringv17, names(gene), value = TRUE)  
colv18 <- grep(stringv18, names(gene), value = TRUE)  
colv19 <- grep(stringv19, names(gene), value = TRUE)  
colv20 <- grep(stringv20, names(gene), value = TRUE)  
  
colv21 <- grep(stringv21, names(gene), value = TRUE)  
colv22 <- grep(stringv22, names(gene), value = TRUE)  
colv23 <- grep(stringv23, names(gene), value = TRUE)  
colv24 <- grep(stringv24, names(gene), value = TRUE)  
colv25 <- grep(stringv25, names(gene), value = TRUE)  
colv26 <- grep(stringv26, names(gene), value = TRUE)  
colv27 <- grep(stringv27, names(gene), value = TRUE)  
colv28 <- grep(stringv28, names(gene), value = TRUE)  
colv29 <- grep(stringv29, names(gene), value = TRUE)  
colv30 <- grep(stringv30, names(gene), value = TRUE)  
  
colv31 <- grep(stringv31, names(gene), value = TRUE)  
colv32 <- grep(stringv32, names(gene), value = TRUE)  
colv33 <- grep(stringv33, names(gene), value = TRUE)  
colv34 <- grep(stringv34, names(gene), value = TRUE)  
colv35 <- grep(stringv35, names(gene), value = TRUE)  
colv36 <- grep(stringv36, names(gene), value = TRUE)  
colv37 <- grep(stringv37, names(gene), value = TRUE)  
colv38 <- grep(stringv38, names(gene), value = TRUE)  
colv39 <- grep(stringv39, names(gene), value = TRUE)  
colv40 <- grep(stringv40, names(gene), value = TRUE)  
  
colv41 <- grep(stringv41, names(gene), value = TRUE)  
colv42 <- grep(stringv42, names(gene), value = TRUE)  
colv43 <- grep(stringv43, names(gene), value = TRUE)  
colv44 <- grep(stringv44, names(gene), value = TRUE)  
colv45 <- grep(stringv45, names(gene), value = TRUE)  
colv46 <- grep(stringv46, names(gene), value = TRUE)  
colv47 <- grep(stringv47, names(gene), value = TRUE)  
colv48 <- grep(stringv48, names(gene), value = TRUE)  
colv49 <- grep(stringv49, names(gene), value = TRUE)  
colv50 <- grep(stringv50, names(gene), value = TRUE)  
  
# j gene  
colj1 <- grep(stringj1, names(gene), value = TRUE)  
colj2 <- grep(stringj2, names(gene), value = TRUE)  
colj3 <- grep(stringj3, names(gene), value = TRUE)  
colj4 <- grep(stringj4, names(gene), value = TRUE)  
colj5 <- grep(stringj5, names(gene), value = TRUE)  
colj6 <- grep(stringj6, names(gene), value = TRUE)  
colj7 <- grep(stringj7, names(gene), value = TRUE)  
colj8 <- grep(stringj8, names(gene), value = TRUE)  
colj9 <- grep(stringj9, names(gene), value = TRUE)  
colj10 <- grep(stringj10, names(gene), value = TRUE)  
  
colj11 <- grep(stringj11, names(gene), value = TRUE)  
colj12 <- grep(stringj12, names(gene), value = TRUE)  
colj13 <- grep(stringj13, names(gene), value = TRUE)  
set.na1 <- c(22)  
set.na2 <- c(94:109)  
Y1 <- gene$Y1  
Y1[set.na1] <- "recovered"  
Y1[set.na2] <- "healthy"  
# subsets  
actRec <- subset(gene, Y1 == "active" | Y1 == "recovered")  
Y.ar <- rep(0, length(actRec$Y1))  
Y.ar[which(actRec$Y1 == "active")] = 1  
  
actHea <- subset(gene, Y1 == "active" | Y1 == "healthy")  
Y.ah <- rep(0, length(actHea$Y1))  
Y.ah[which(actHea$Y1 == "active")] = 1  
  
  
recHea <- subset(gene, Y1 == "recovered" | Y1 == "healthy")  
Y.rh <- rep(0, length(recHea$Y1))  
Y.rh[which(recHea$Y1 == "recovered")] = 1  
  
# null models  
obj.ar <- SKAT\_Null\_Model(Y.ar ~ 1, out\_type = "D")  
obj.ah <- SKAT\_Null\_Model(Y.ah ~ 1, out\_type = "D")  
obj.rh <- SKAT\_Null\_Model(Y.rh ~ 1, out\_type = "D")  
p.ar <- rep(0,50)  
ar.val <- rep(0,13)  
  
# loop  
for (i in 1:50) {  
 col.idx <- get(paste0("colv", i,sep=""))  
 sub <- as.matrix(actRec[,col.idx])  
 out <- SKATBinary(sub, obj.ar, kernel = "linear.weighted")  
 p <- out$p.value  
 p.ar[i] <- p  
}  
ar.v <- data.frame(cbind(c(1:50), p.ar))  
colnames(ar.v) <- c("vgene.idx","pvalue")  
ar.v  
for (i in 1:13) {  
 col.idx <- get(paste0("colj", i,sep=""))  
 sub <- as.matrix(actRec[,col.idx])  
 out <- SKATBinary(sub, obj.ar, kernel = "linear.weighted")  
 p <- out$p.value  
 ar.val[i] <- p  
}  
ar.j <- data.frame(cbind(c(1:13),ar.val))  
colnames(ar.j) <- c("jgene.idx","p-value")  
ar.j  
p.ah <- rep(0,50)  
ah.val <- rep(0,13)  
  
# loop  
for (i in 1:50) {  
 col.idx <- get(paste0("colv", i,sep=""))  
 sub <- as.matrix(actHea[,col.idx])  
 out <- SKATBinary(sub, obj.ah, kernel = "linear.weighted")  
 p <- out$p.value  
 p.ah[i] <- p  
}  
ah.v <- data.frame(cbind(c(1:50), p.ah))  
colnames(ah.v) <- c("vgene.idx","pvalue")  
ah.v  
for (i in 1:13) {  
 col.idx <- get(paste0("colj", i,sep=""))  
 sub <- as.matrix(actHea[,col.idx])  
 out <- SKATBinary(sub, obj.ah, kernel = "linear.weighted")  
 p <- out$p.value  
 ah.val[i] <- p  
}  
ah.j <- data.frame(cbind(c(1:13),ah.val))  
colnames(ah.j) <- c("jgene.idx","p-value")  
ah.j  
p.rh <- rep(0,50)  
rh.val <- rep(0,13)  
  
# loop  
for (i in 1:50) {  
 col.idx <- get(paste0("colv", i,sep=""))  
 sub <- as.matrix(recHea[,col.idx])  
 out <- SKATBinary(sub, obj.rh, kernel = "linear.weighted")  
 p <- out$p.value  
 p.rh[i] <- p  
}  
rh.v <- data.frame(cbind(c(1:50), p.rh))  
colnames(rh.v) <- c("vgene.idx","pvalue")  
rh.v  
for (i in 1:13) {  
 col.idx <- get(paste0("colj", i,sep=""))  
 sub <- as.matrix(recHea[,col.idx])  
 out <- SKATBinary(sub, obj.rh, kernel = "linear.weighted")  
 p <- out$p.value  
 rh.val[i] <- p  
}  
rh.j <- data.frame(cbind(c(1:13),rh.val))  
colnames(rh.j) <- c("jgene.idx","p-value")  
rh.j  
# v gene  
ar.pv <- ar.v$pvalue  
ar.paV <- p.adjust(ar.pv, method = p.adjust.methods, n = length(ar.pv))  
ar.vRes <- data.frame(cbind(c(1:50), ar.paV))  
colnames(ar.vRes) <- c("vgene.idx","p-value")  
ar.vRes  
# j gene  
ar.pj <- ar.j$`p-value`  
ar.paJ <- p.adjust(ar.pj, method = p.adjust.methods, n = length(ar.pj))  
ar.jRes <- data.frame(cbind(c(1:13), ar.paJ))  
colnames(ar.jRes) <- c("jgene.idx","p-value")  
ar.jRes  
# v gene  
ah.pv <- ah.v$pvalue  
ah.paV <- p.adjust(ah.pv, method = p.adjust.methods, n = length(ah.pv))  
ah.vRes <- data.frame(cbind(c(1:50), ah.paV))  
colnames(ah.vRes) <- c("vgene.idx","p-value")  
ah.vRes  
# j gene  
ah.pj <- ah.j$`p-value`  
ah.paJ <- p.adjust(ah.pj, method = p.adjust.methods, n = length(ah.pj))  
ah.jRes <- data.frame(cbind(c(1:13), ah.paJ))  
colnames(ah.jRes) <- c("jgene.idx","p-value")  
ah.jRes  
# v gene  
rh.pv <- rh.v$pvalue  
rh.paV <- p.adjust(rh.pv, method = p.adjust.methods, n = length(rh.pv))  
rh.vRes <- data.frame(cbind(c(1:50), rh.paV))  
colnames(rh.vRes) <- c("vgene.idx","p-value")  
rh.vRes  
# j gene  
rh.pj <- rh.j$`p-value`  
rh.paJ <- p.adjust(rh.pj, method = p.adjust.methods, n = length(rh.pj))  
rh.jRes <- data.frame(cbind(c(1:13), rh.paJ))  
colnames(rh.jRes) <- c("jgene.idx","p-value")  
rh.jRes  
# dataframe  
gene$Y1 <- Y1  
dfull <- gene[2:677]  
  
# v gene  
ar.v <- actRec[, c(colv1, colv2, colv3, colv4, colv5, colv6, colv8, colv10,   
 colv11, colv12, colv13, colv14, colv15, colv16, colv17,   
 colv18, colv19, colv20, colv21, colv22, colv23, colv24,   
 colv25, colv27, colv29, colv30, colv31, colv32, colv33,   
 colv34, colv35, colv36, colv37, colv39, colv40, colv41,   
 colv42, colv44, colv45, colv46, colv47, colv48, colv49,   
 colv50)]   
  
ah.v <- actHea[, colv37]   
  
rh.v <- recHea[, c(colv3, colv10, colv15, colv16, colv18, colv19, colv24,   
 colv27, colv30, colv31, colv33, colv36, colv40, colv46,   
 colv47, colv49)]  
  
# j gene  
ar.j <- actRec[, c(colj1, colj2, colj3, colj4, colj5, colj6, colj7, colj8,   
 colj9, colj10, colj11, colj12, colj13)]  
ar.j <- ar.j[, which(apply(ar.j, 2, var) != 0)]  
  
rh.j <- recHea[, c(colj1, colj2, colj3, colj4, colj6, colj7, colj9, colj10,   
 colj12, colj13)]  
rh.j <- rh.j[, which(apply(rh.j, 2, var) != 0)]  
  
# pca res  
pcaFull <- prcomp(dfull, scale. = TRUE)  
# v gene  
pca.arV <- prcomp(ar.v, scale. = TRUE)  
pca.ahV <- prcomp(ah.v, scale. = TRUE)  
pca.rhV <- prcomp(rh.v, scale. = TRUE)   
  
# j gene  
pca.arJ <- prcomp(ar.j, scale. = TRUE)  
pca.ahJ <- prcomp(ah.j, scale. = TRUE)  
pca.rhJ <- prcomp(rh.j, scale. = TRUE)

autoplot(pca.arV, data = actRec, colour = 'Y1')  
autoplot(pca.ahV, data = actHea, colour = 'Y1')  
autoplot(pca.rhV, data = recHea, colour = 'Y1')  
autoplot(pca.arJ, data = actRec, colour = 'Y1')  
autoplot(pca.rhJ, data = recHea, colour = 'Y1')  
autoplot(pcaFull, data = gene, colour = 'Y1')