BioData Lab <Visualization : Heatmap3 >

김성민, 2018.6.22

# 필요한 library

**library(heatmap3)**

**library(RColorBrewer)**

**library(gplots)**

# read data

**od<-read.csv("D:/git/SungminCode/HGU/BioData/HowTo/heatmap/sample.csv",header = T,sep = ',')**

**dim(od)**

# ## data processing

**data<-od**

**rownames(data)<-data$patient** #data 의 row names을 data의 patient이름으로

**data<-subset(data,select = -c(patient,index))** # heatmap3에는 index와 patient이름이 필요가 없다.

# ##basic heatmap3

#heatmap에 들어가는 data는 다 numeric data로 되어야 함.

#그래서 cancer\_code를 지우고, result를 gene interaction을 보는 heatmap 에는 필요가 없다.

#나중에 복잡한 heatmap 그릴때 필요!

**input<-subset(data,select = -c(cancer\_code,result))**

**input <-as.matrix(input)**

#처음에는 일단 기본적인 heatmap을 그린다.

png("파일이름",파일 크기)

--figure--

dev.off() 이것을 해야 plot이 close되고 파일을 최종 저장한다. plot을 저장할때 항상 필요

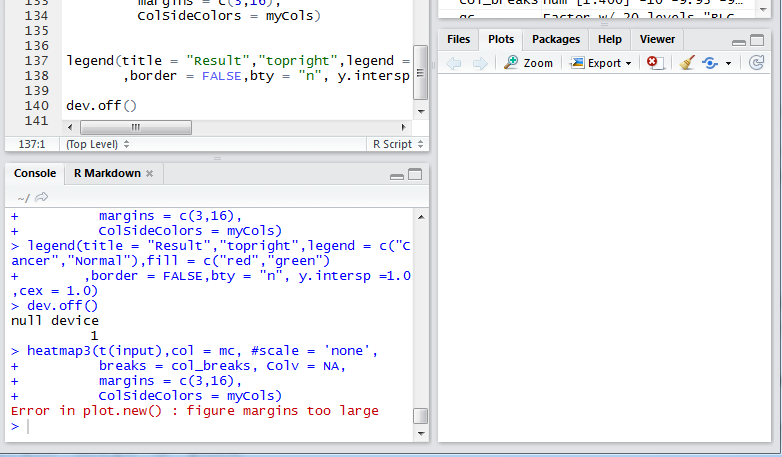
#그리고 plot을 다시 그릴대 dev.off를 하면 plot 창에 그림이 다 지워진다.

#이렇게 안해도 rstudio 옆에 plot에 보여주긴 한다.

#예) 그냥 pnt() dev.off()를 빼고 heatmap3(input)만 run! 하면 오른쪽아래 plot크기만 크게 하면 될 것 보일 것이다.

#하지만 가끔 오류가 뜨게 되면, 이 방법을 통해 확인 가능

#일단 주석처리 하고 질문이 있으면.. 저에게..



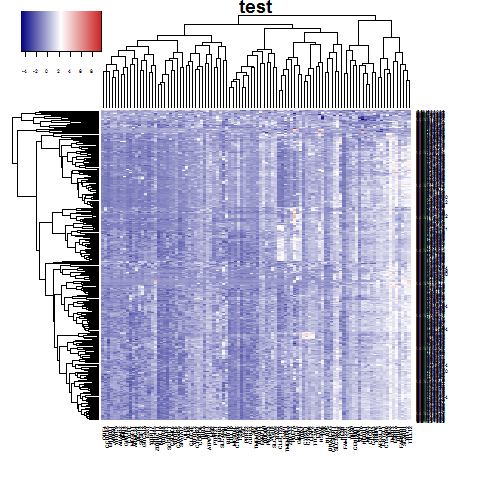
(빨간줄) 경고가 뜨면 rstudio를 전체 화면을 하고 파란색 클릭!

**#png("basic\_heatmap3.png")**

#main은 plot 이름을, cexRow = ROW 글자 크기, cexCol = COL 글자 크기. 변경해보면서 확인!

**heatmap3(input,main = "test",cexRow = 0.5, cexCol = 0.5)**

**dev.off()**



# ## heatmap ColV RowV

#지금 그림을 보면 가로 = 사람, 세로 = gene이름이 있다.

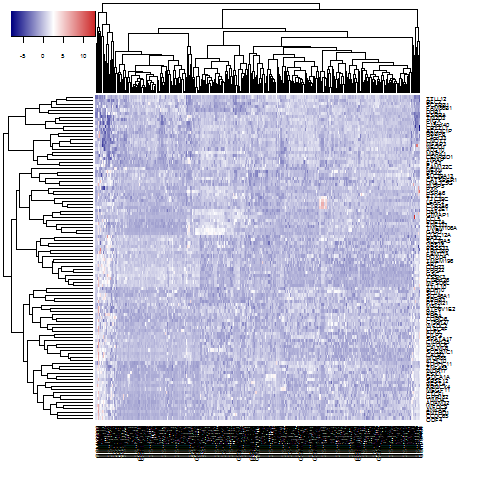
# 그리고 자동으로 clustering (위에 나뭇가지 처럼 되어 있는 그림)이 된다.

# 이것을 조절 하는 것이 Colv, Rowv parameter 이다.

# 그리고 우리 기존 data.frame이 row = patient, col = variables이다.

# t(input)을 통해 row, col의 위치를 바꾸어 준다.

# 그리고 ColV = NA로 설정하면 col의 나뭇가지가 없어진다.

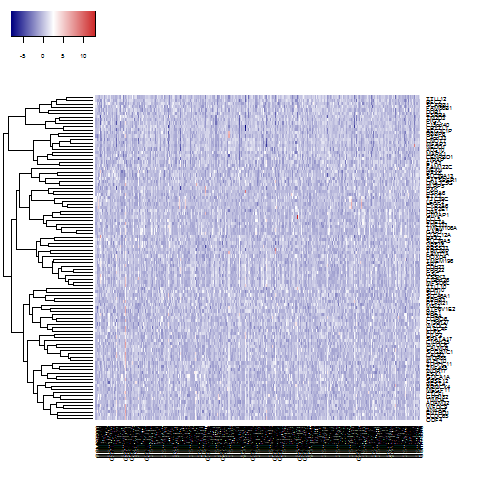


#colv = T와 비교해보면 확실하게 느낌이 올것이다.

**#png("heatmap\_colv\_F.png")**

**heatmap3(t(input),Colv = F)**

**dev.off()**

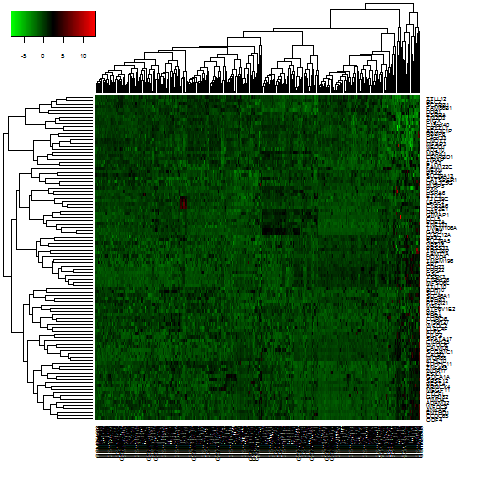


**#png("heatmap\_colv\_NA.png")**

#colv = NA는 나뭇가지 지우기

**heatmap3(t(input),Colv = NA)**

**dev.off()**



# ##color and breaks

**#png("heatmap\_color.png")**

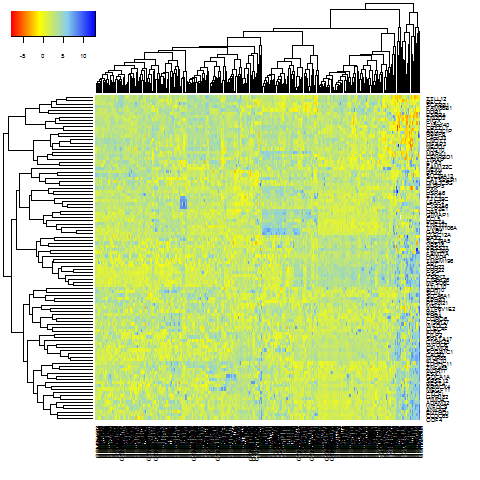
**heatmap3(t(input),col = greenred(100))** # col = 색 설정

**dev.off()**

#breaks 는 색 range를 설정해 주는 것이다.

#여러가지 색을 사용하기 위해 새로운 color를 저장한다.

**mc <- colorRampPalette(c("red", "yellow" ,"skyblue" ,"blue"))(n = 399)**

#col\_breaks 에 저장을 한다.

**col\_breaks = c(seq(-10,-3,length=100),**

**seq(-2.99,0,length=100),**

**seq(0.01,3,length=100),**

**seq(5,10,length=100))**

#여기서 중요한 점은 color의 갯수(n)이 breaks length의 갯수보다 1 작아야 한다.

#length의 합 = n + 1 # 400 = 399 + 1

**#png("heatmap\_color\_break.png")**

**heatmap3(t(input),col = mc, breaks = col\_breaks)**

**dev.off()**

# ## scale

scale = 'none'으로 설정 하면 orginal data 값으로 구분함.

#그전에는 heatmap에서 scale을 조절함(default = 'row')

**#png("heatmap\_scale.png")**

**heatmap3(t(input),col = mc, breaks = col\_breaks,scale = 'none')** # scale (왼쪽 위 key value 값 확인)

**dev.off()**

# ##heatmap 위에 원하는 index bar 추가!

# heatmap에 cancer code와 result 값을 추가!

#이전에 orginat data를 다시 가공해야함

**model <-od**

**model <- model[order(-model$result,model$cancer\_code),]**

**gc <- factor(model$cancer\_code)**

**cancer\_code<-model$cancer\_code**

**num\_cancercode <- length(levels(model$cancer\_code))**

**CancerCode\_color <- rainbow(num\_cancercode)[as.integer(gc)]**

**fresult <-function(result){**

**if(result == 1) {"#CC0000"} #red**

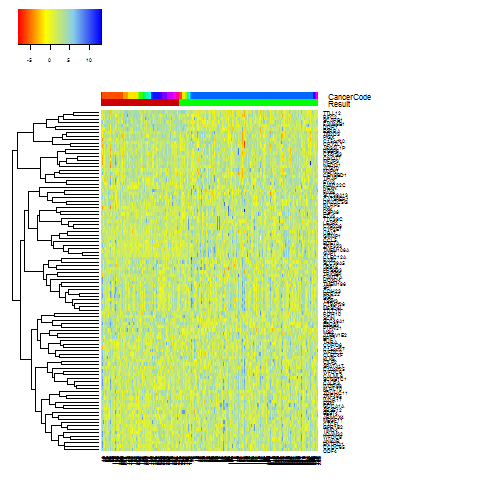
**else{"#00FF00"} #green**

**}**

**result\_color<-unlist(lapply(model$result,fresult))**

**myCols <- cbind(result\_color,CancerCode\_color)**

**colnames(myCols)[1] <- "Result"**

**colnames(myCols)[2] <- "CancerCode"**

**model<-subset(model,select = -c(patient,cancer\_code,result,index))**

**input<-data.matrix(model)**

**png("heatmap\_extra\_colbar.png")**

#Scale, colv 값을 조절하면서 보면 차이를 확일 할 수 있다.

**heatmap3(t(input),col = mc, #scale = 'none',**

**breaks = col\_breaks, Colv = NA,**

**margins = c(3,16),**

**ColSideColors = myCols)**

**dev.off()**

# ##legend 추가

**png("heatmap\_extra\_colbar\_with\_legend.png")**

#Scale, colv 값을 조절하면서 보면 차이를 확일 할 수 잇다.

**heatmap3(t(input),col = mc, #scale = 'none',**

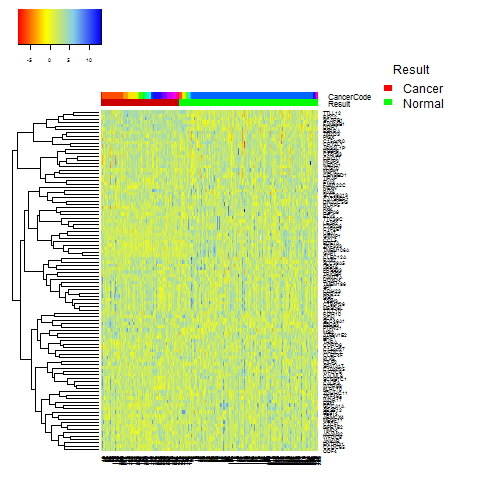
**breaks = col\_breaks, Colv = NA,**

**margins = c(3,16),**

**ColSideColors = myCols)**

**legend(title = "Result","topright",legend = c("Cancer","Normal"),fill = c("red","green")**

**,border = FALSE,bty = "n", y.intersp =1.0,cex = 1.0)**

**dev.off()**