Human analysis on ITGB4:Negative and Positive

yejg

2018/1/29

### Load the packages

library(Seurat)  
library(data.table)  
library(NMF)  
library(rsvd)  
library(Rtsne)  
library(ggplot2)  
library(cowplot)  
library(sva)  
library(igraph)  
library(cccd)  
library(KernSmooth)  
library(beeswarm)  
library(stringr)  
library(formatR)  
source("../tools.R")  
library(ggthemes)  
library(DESeq2)

### The function will be used in the follow

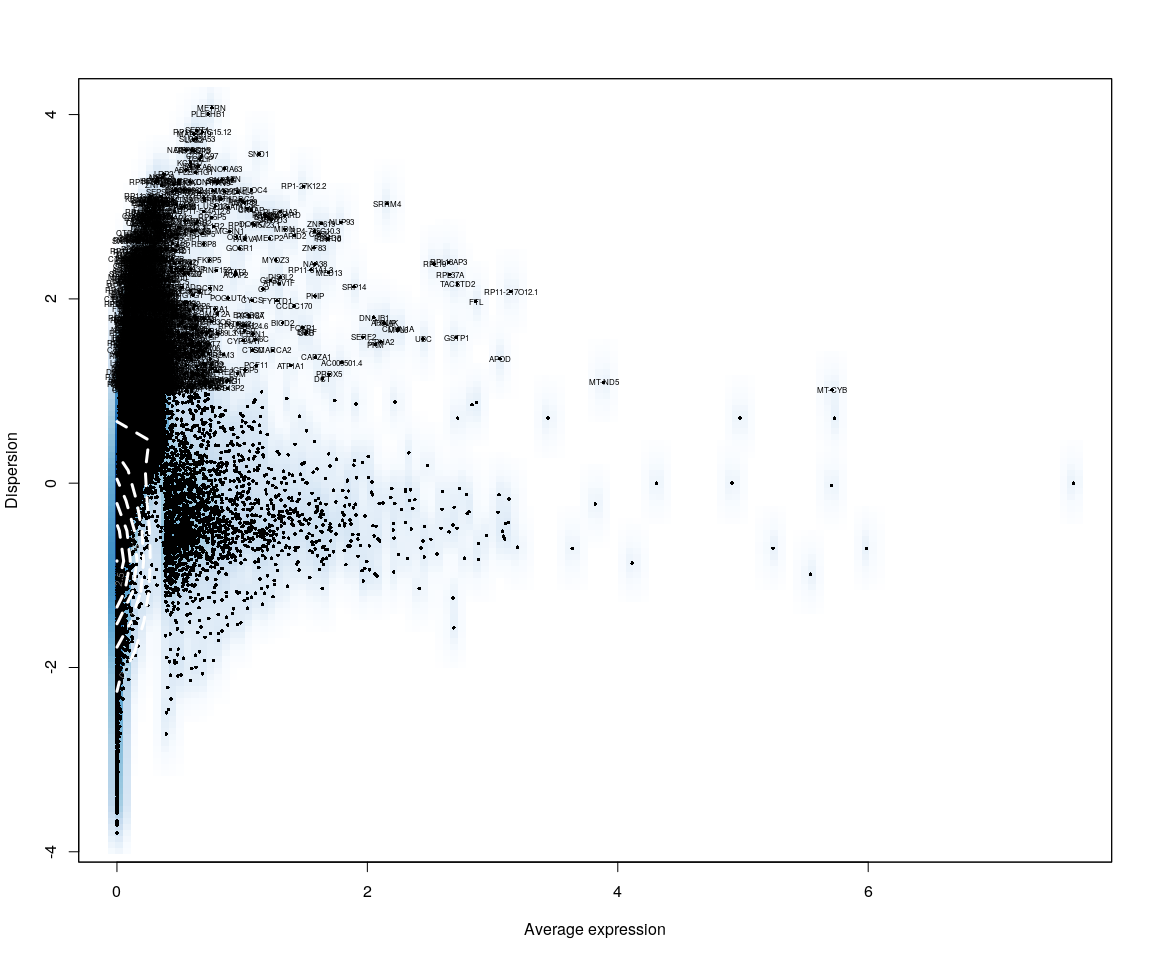
## Analysis based on cell size

According to the previous analysis on sample group,remove the group **hc001** and cell size **2um** ### Read data ### Data QA

human.only.pro <- Load\_data(data\_dir = "../data/human.txt")  
important.genes <- c("ITGB4", "ABCB5", "KRT19", "ACTB", "KRT12", "KRT5", "GAPDH",   
 "KRT3", "PAX6", "WNT7A", "KRT14", "TP63", "KRT10")  
human.only.pro <- human.only.pro[, colnames(human.only.pro)[unlist(lapply(colnames(human.only.pro),   
 function(x) return(str\_split(x, "\_")[[1]][2]))) %in% c("10um", "20um", "6um")]]  
human.only.pro <- human.only.pro[, colnames(human.only.pro)[!unlist(lapply(colnames(human.only.pro),   
 function(x) return(str\_split(x, "\_")[[1]][1]))) %in% c("hc001", "shoutiao")]]  
# human.only.pro<-Simplify\_Select(human.only.pro)

human.all.DESeq <- DESeq\_SeuratObj(X = human.only.pro, DESq = FALSE, min.cells = 10,   
 min.genes = 2)

## [1] "Scaling data matrix"  
##   
 |   
 | | 0%  
 |   
 |=================================================================| 100%



human.imp.lognorm <- data.frame(FetchData(human.all.DESeq, vars.all = important.genes[important.genes %in%   
 rownames(human.all.DESeq@raw.data)]))

### Split the data according to whether the gene ITGB4 is Negative or negative

library(ggplot2)  
library(reshape2)  
  
ITGB4 <- as.numeric(human.imp.lognorm[, "ITGB4"])  
Positive.idx <- which(ITGB4 > 0)  
Negative.idx <- which(ITGB4 == 0)  
Positive.data <- human.imp.lognorm[Positive.idx, , drop = FALSE]  
Negative.data <- human.imp.lognorm[Negative.idx, , drop = FALSE]  
Positive.data <- Positive.data[, -1] # remove ITGB4  
Negative.data <- Negative.data[, -1]

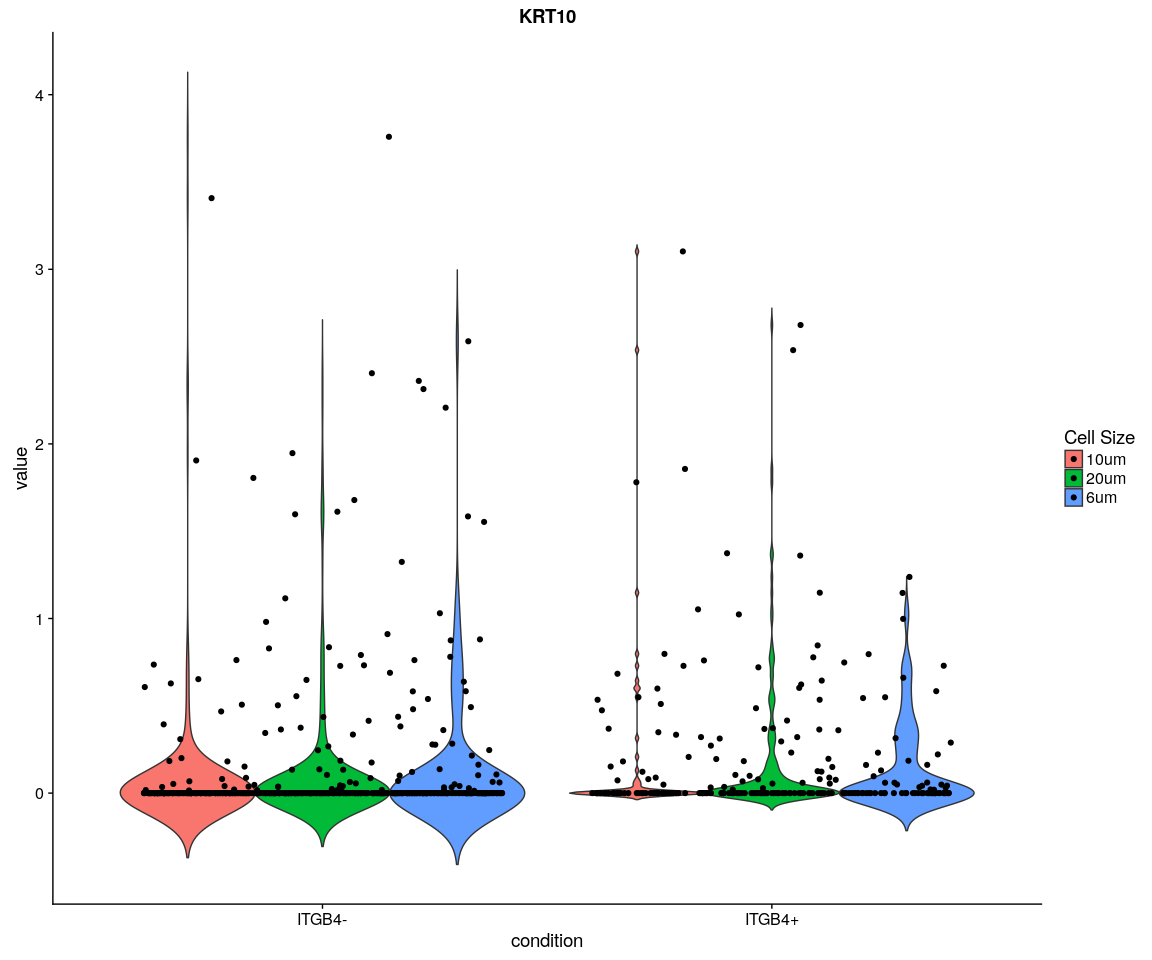
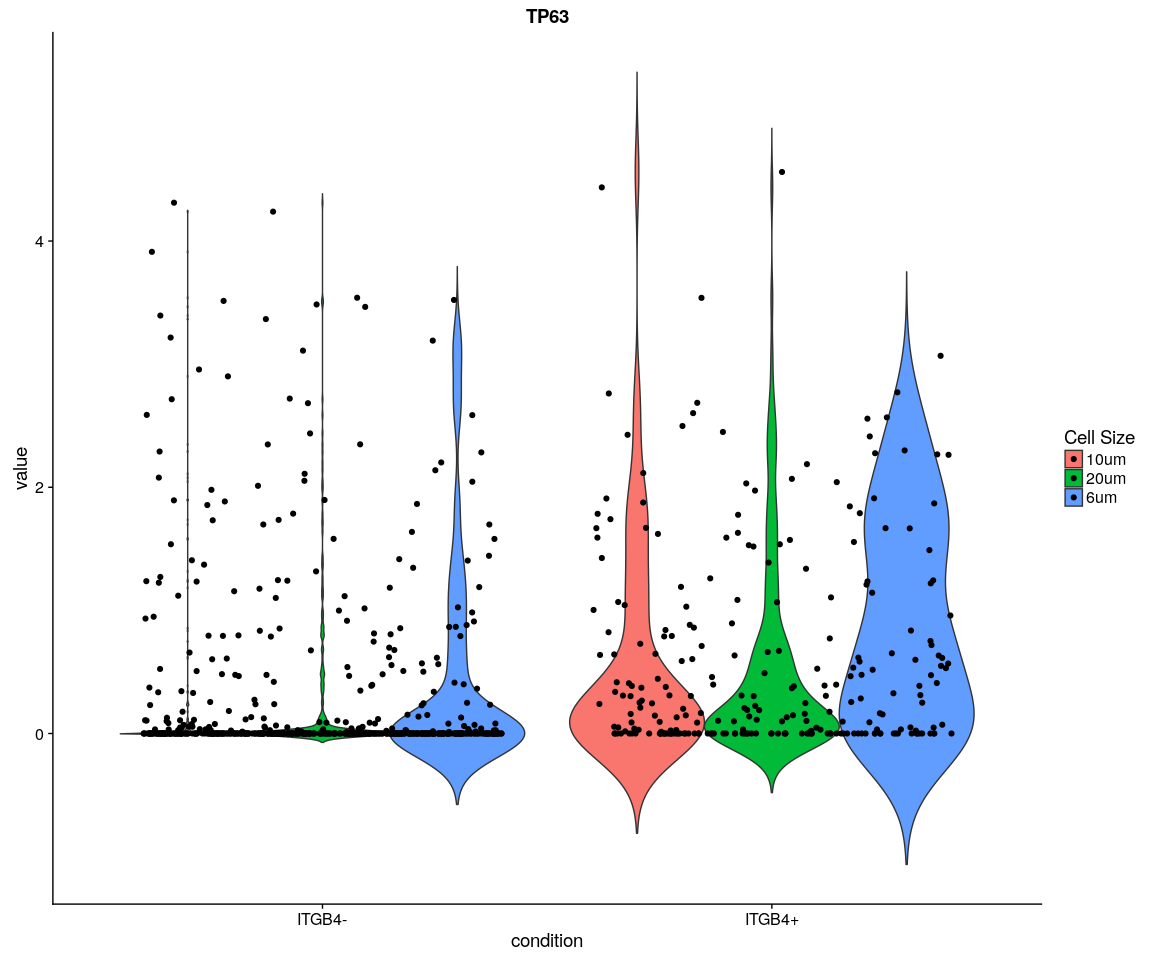
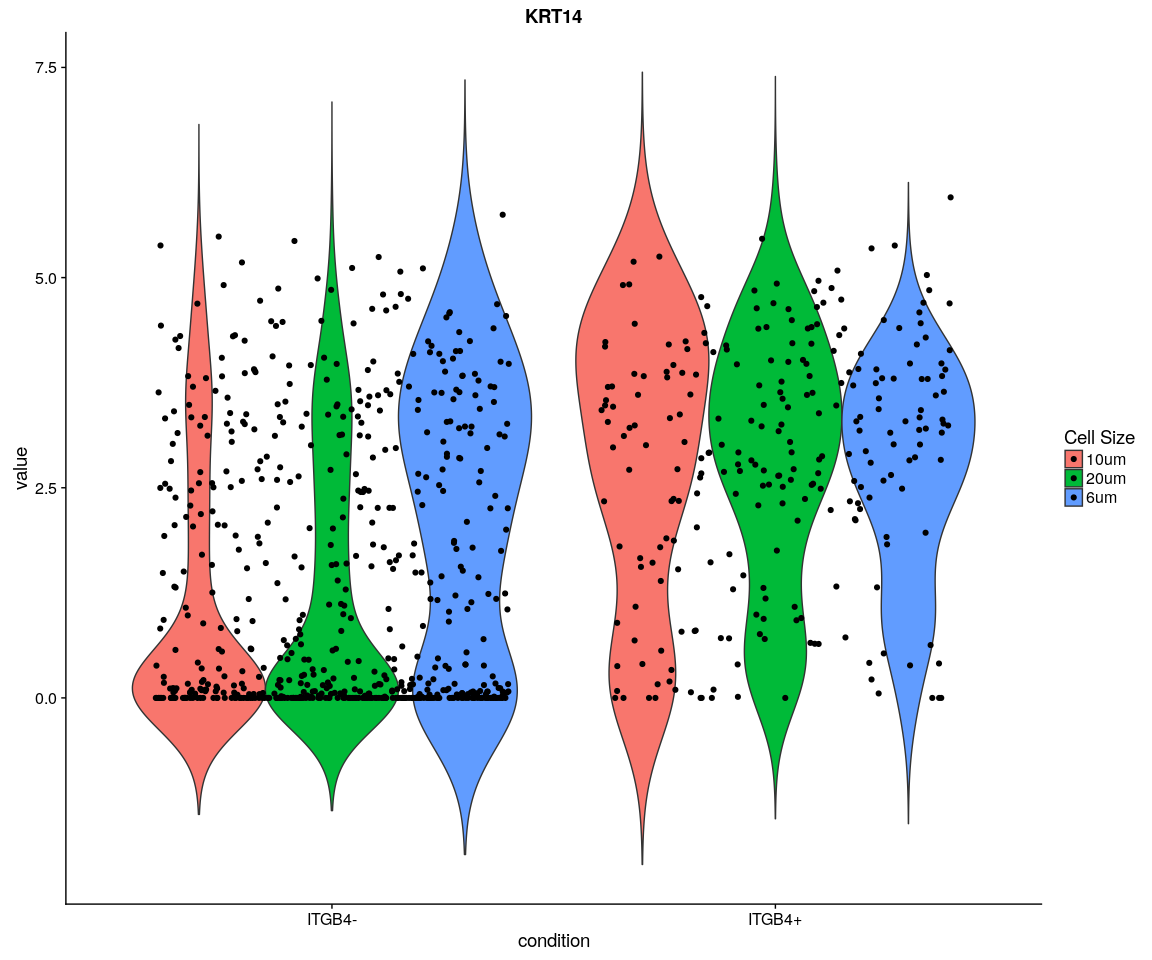
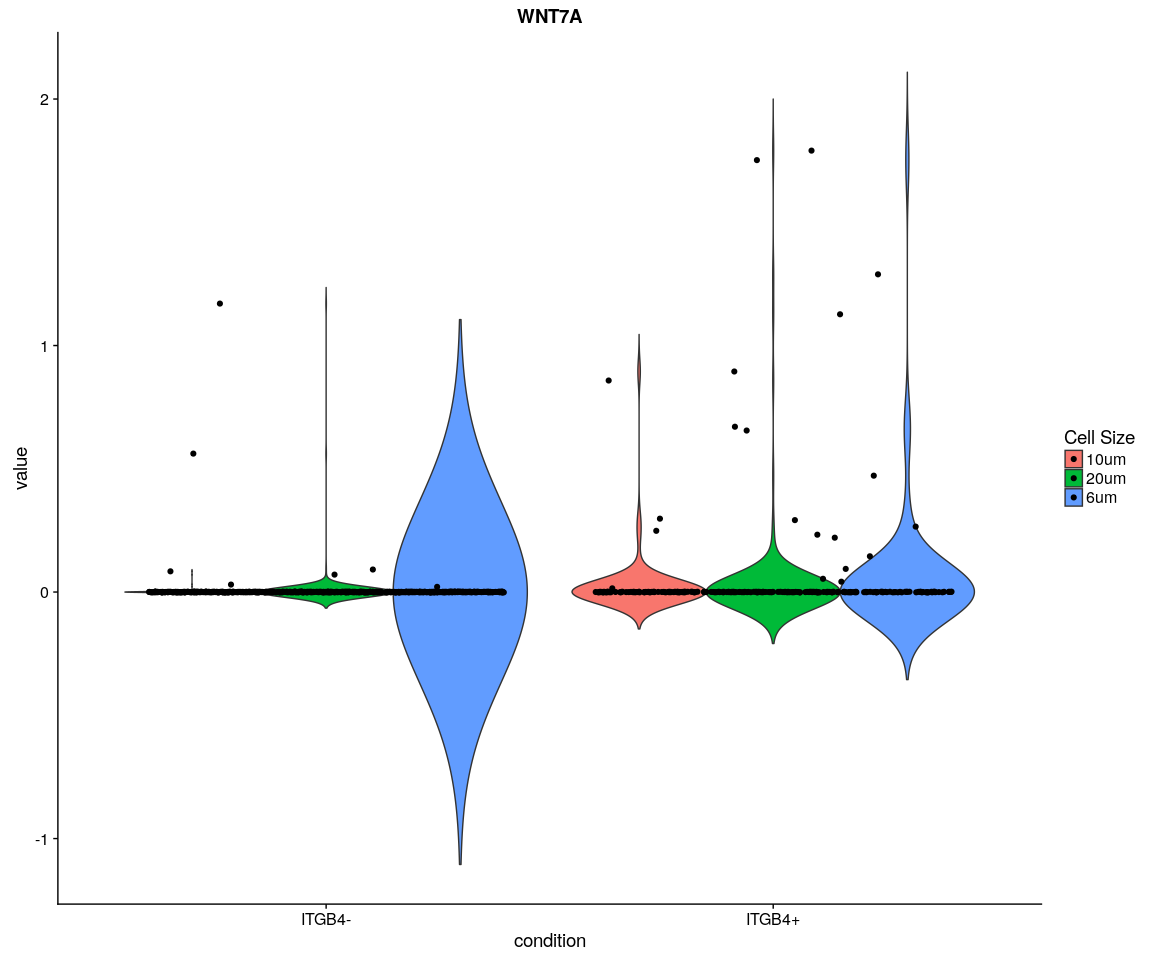
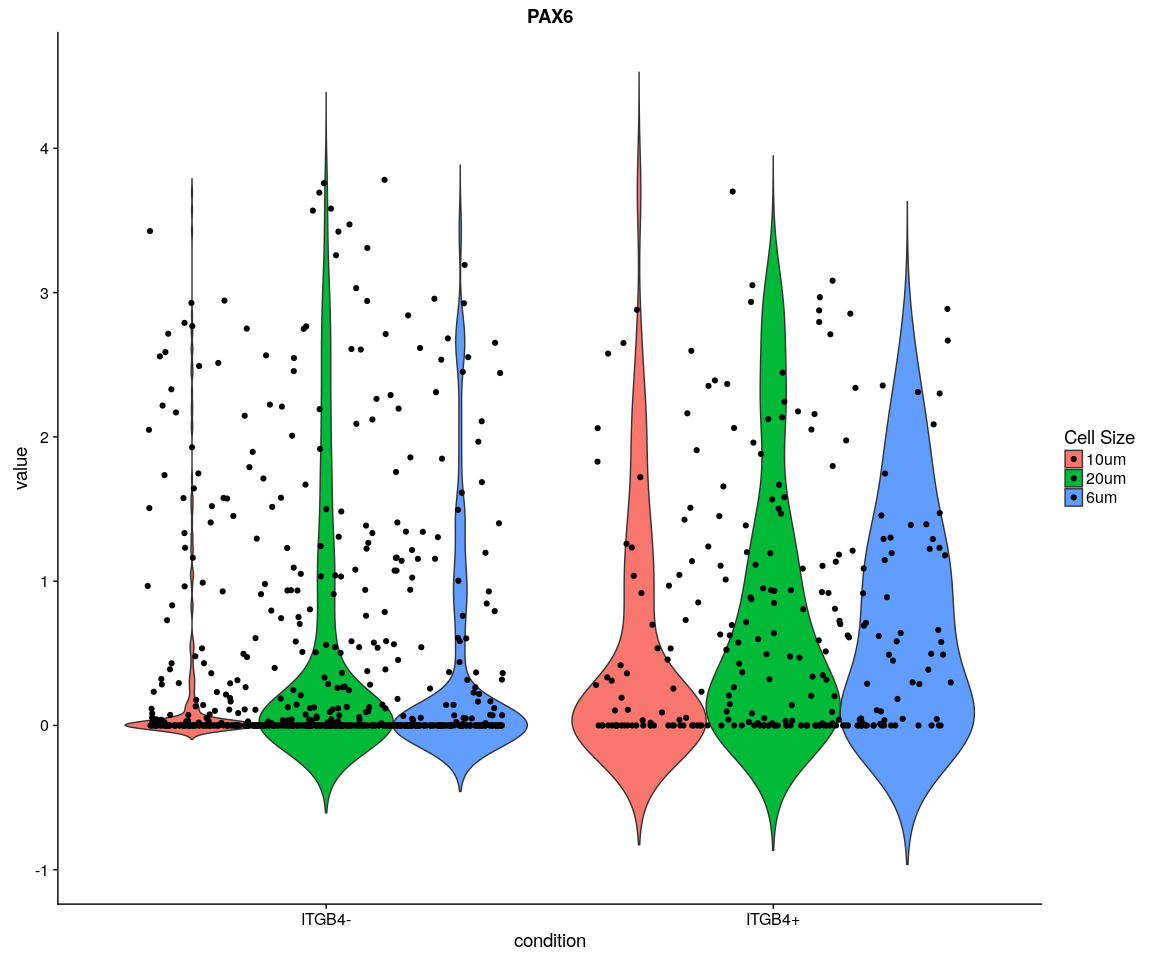
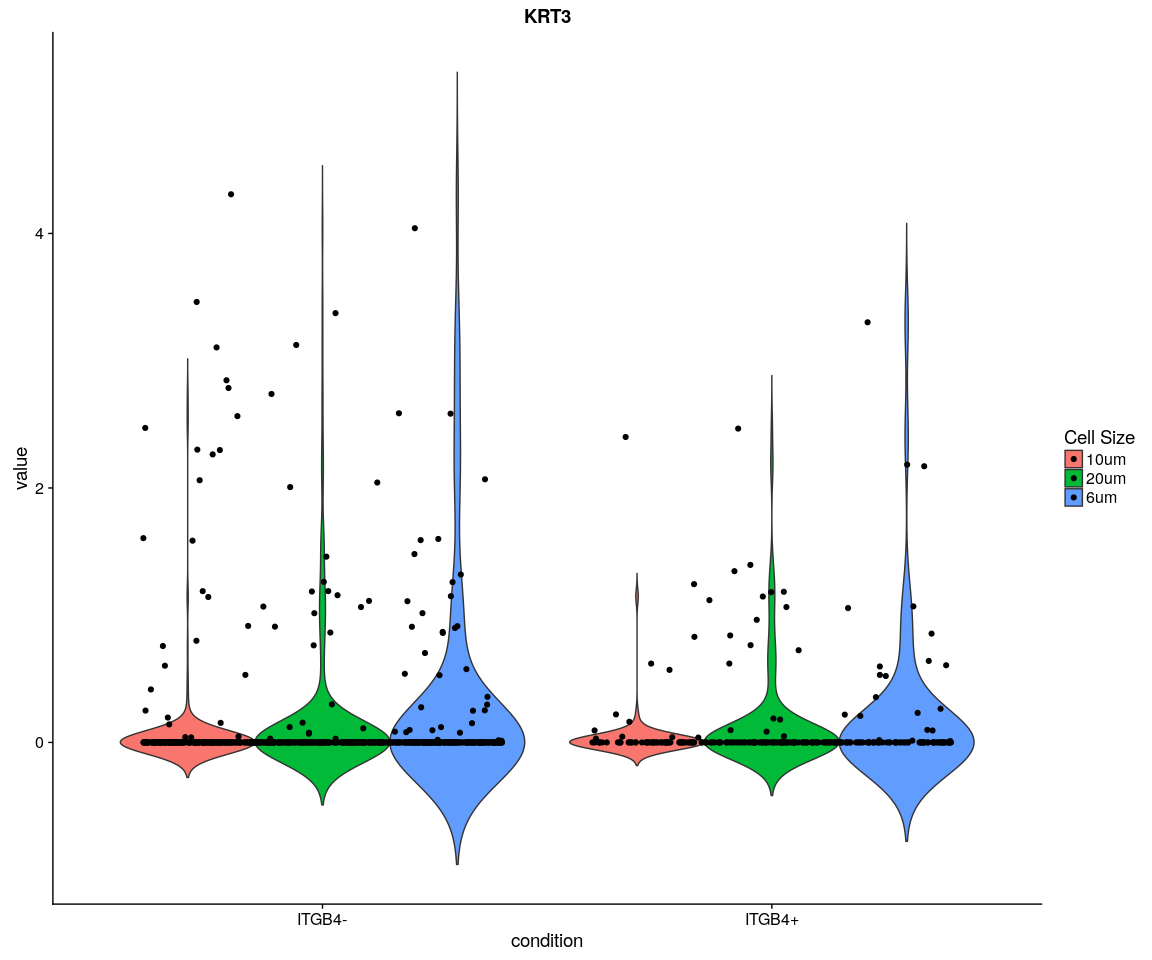
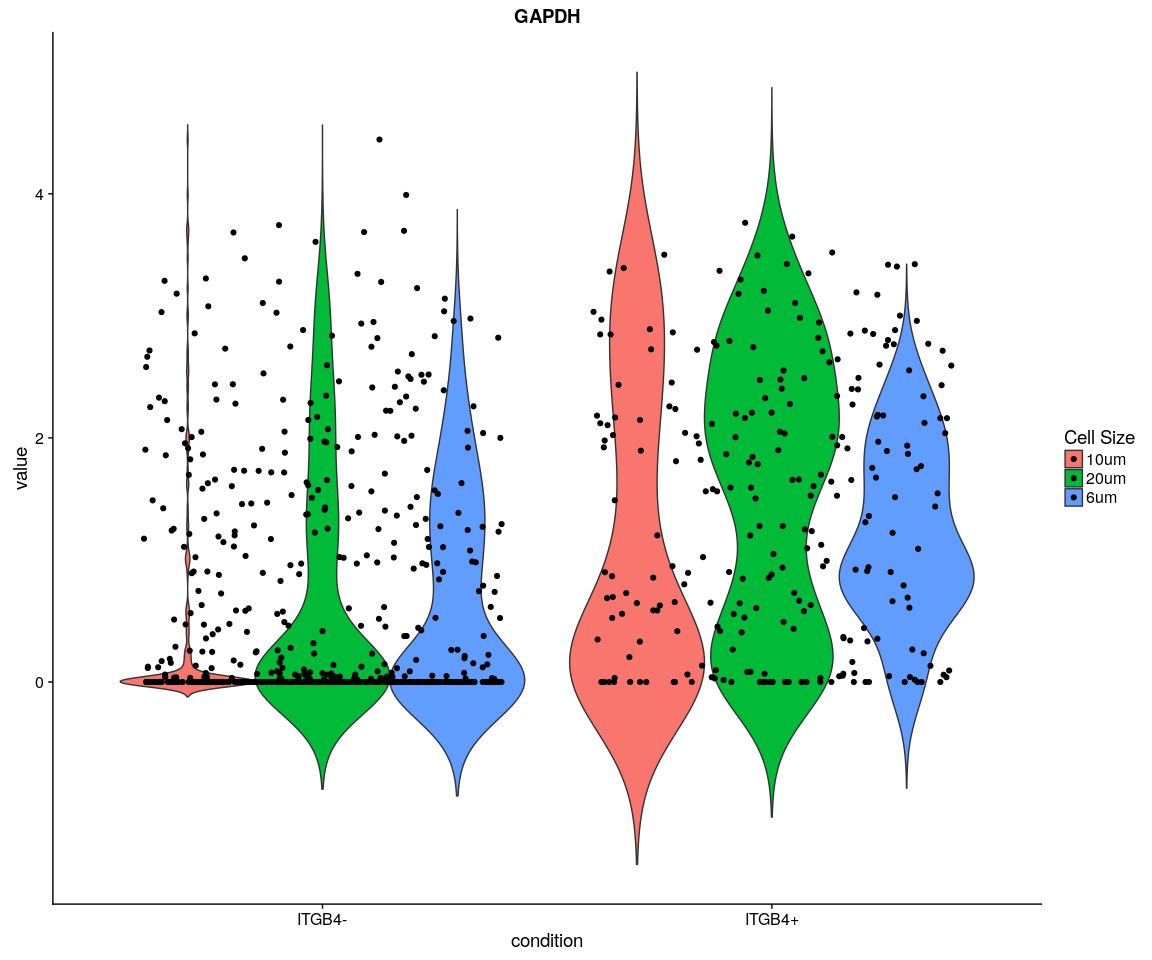
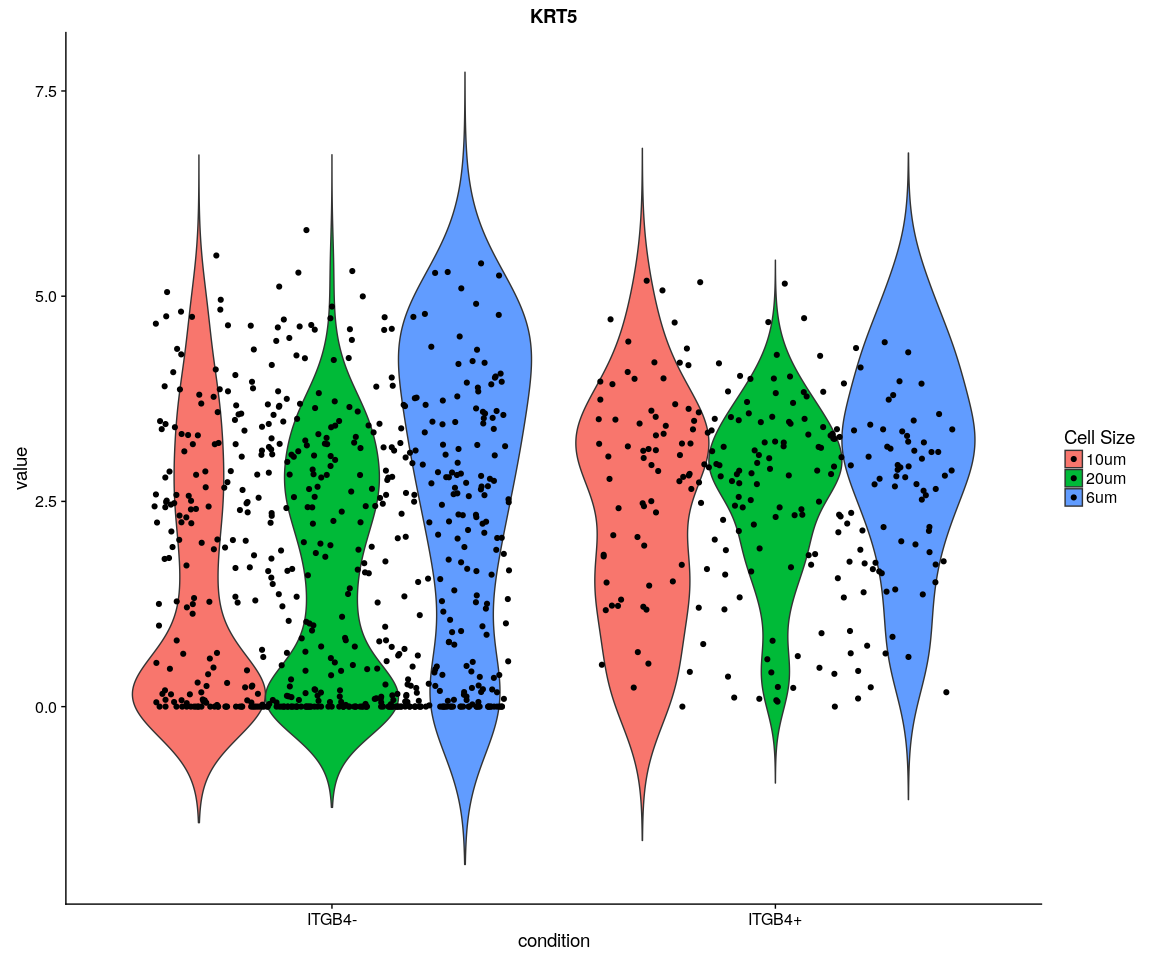
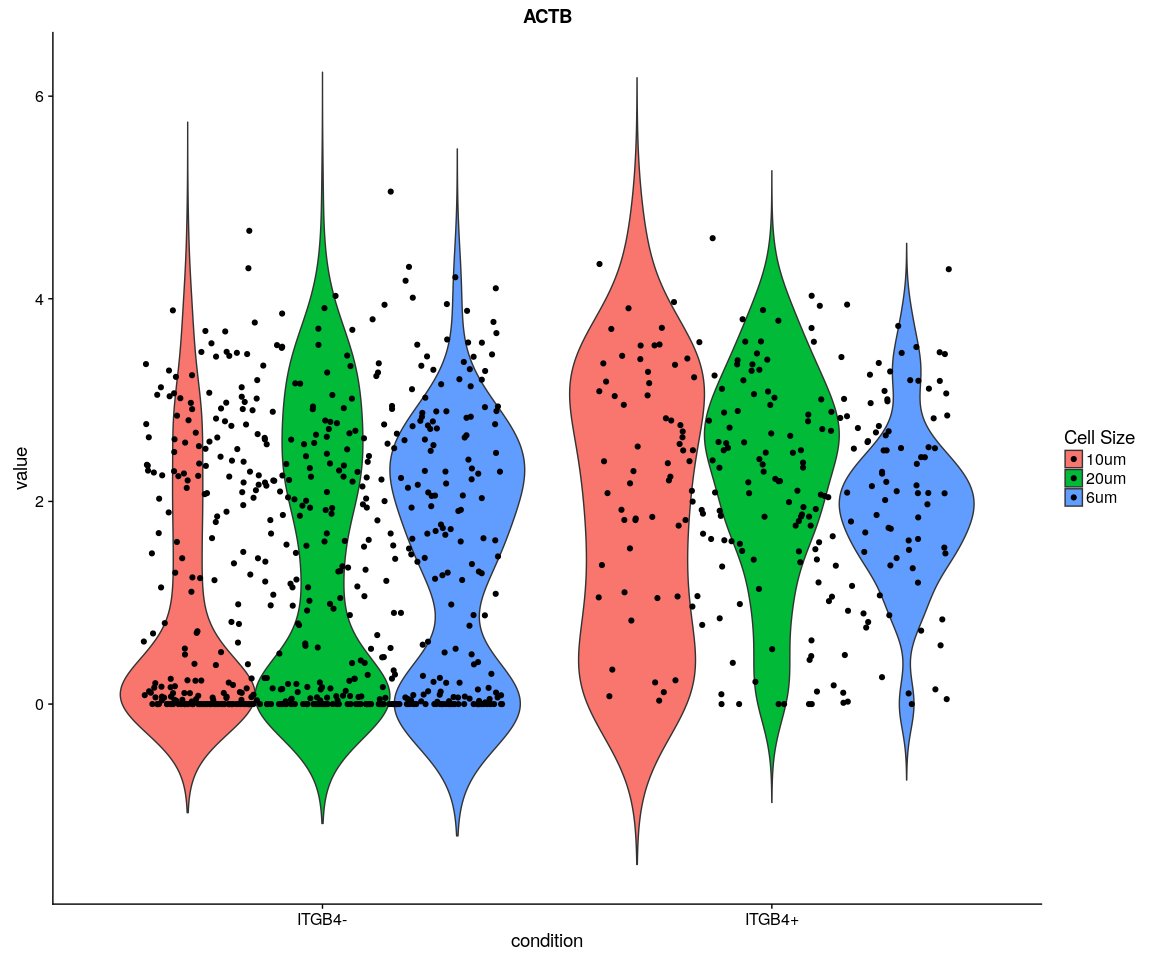
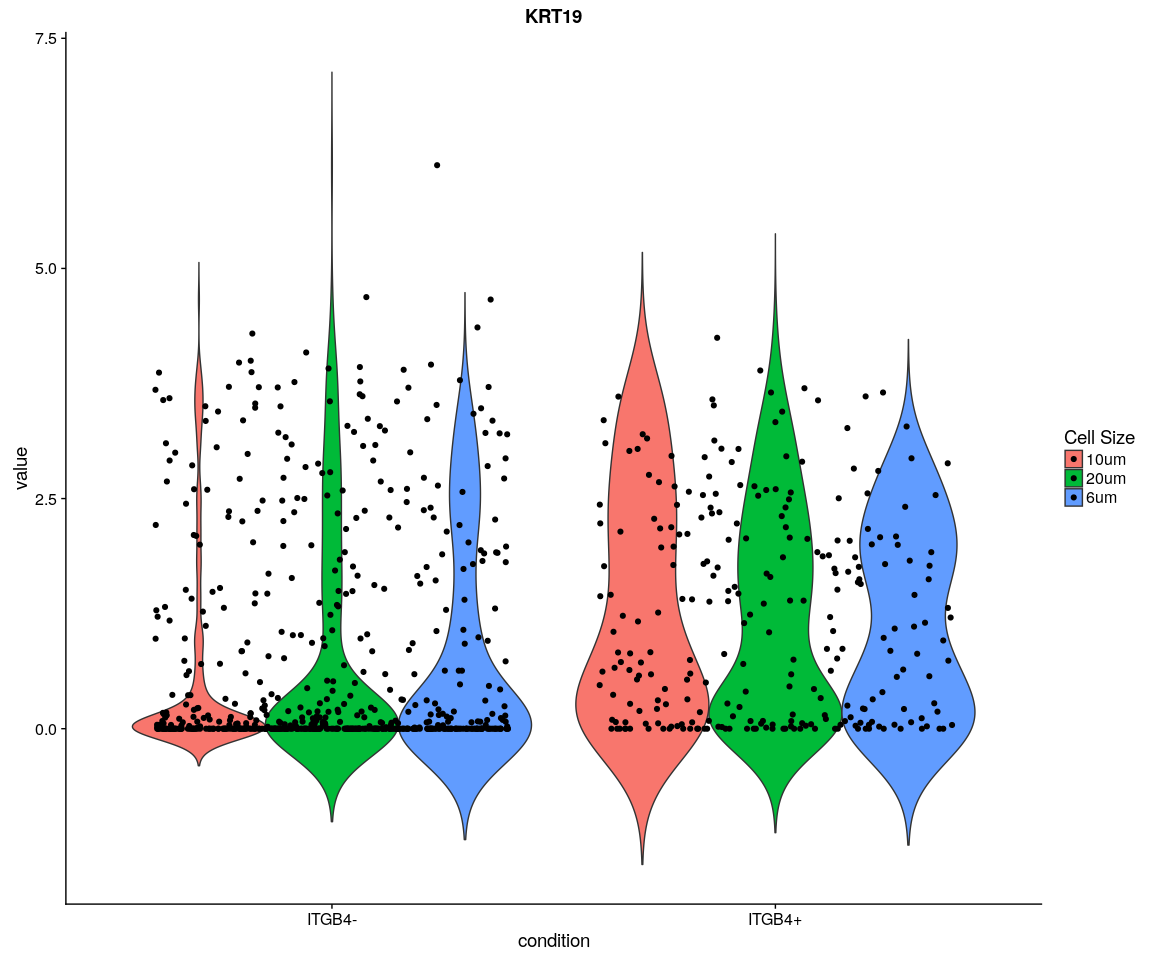
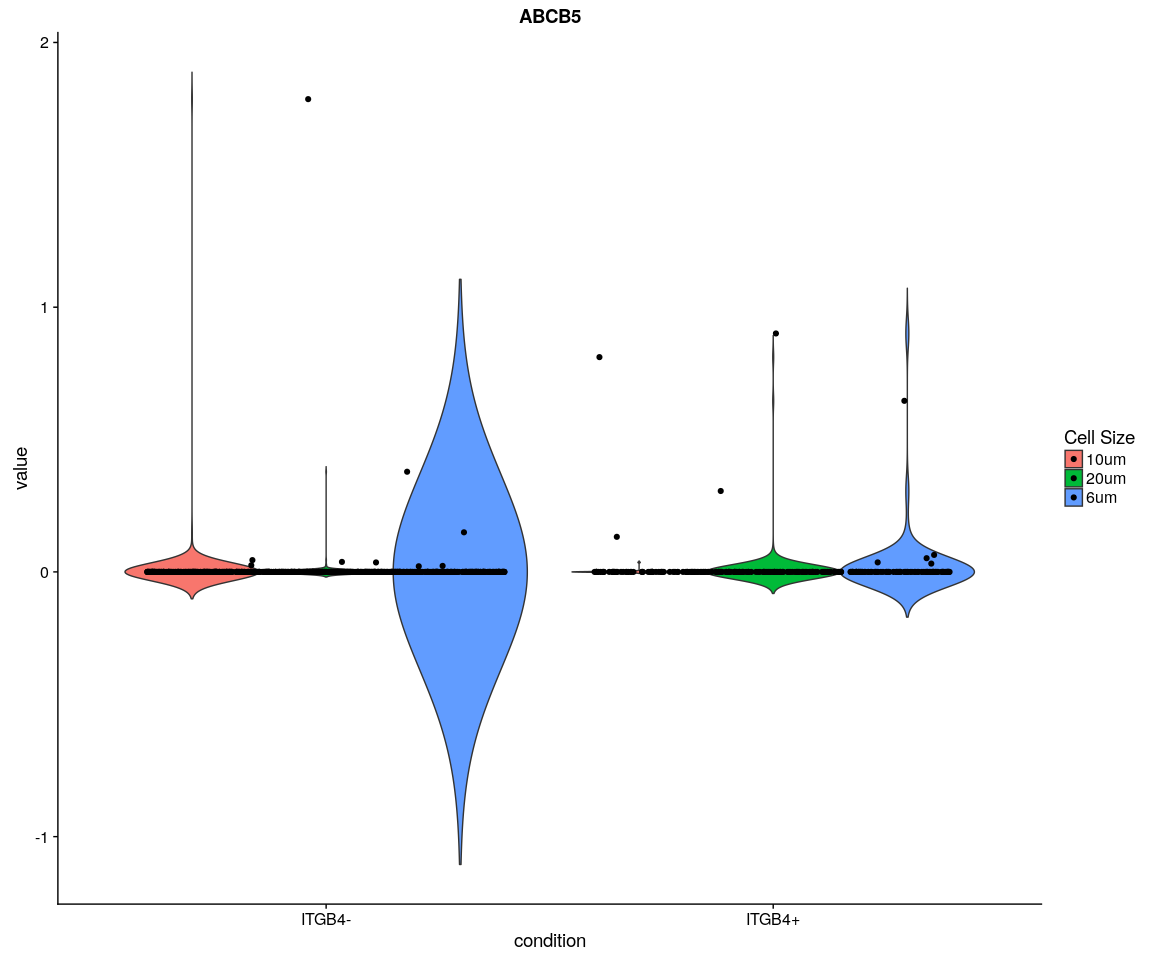
## Figure Explore.1

# Positive.t<-data.frame(as.matrix(LogNormalize(Positive.data,display.progress  
# = FALSE)))  
# Negative.t<-data.frame(as.matrix(LogNormalize(Negative.data,display.progress  
# = FALSE))) Positive.t<-data.frame(t(Positive.t[important.genes,]))  
# Negative.t<-data.frame(t(Negative.t[important.genes,]))  
plot.data <- rbind(Positive.data, Negative.data)  
plot.data$condition <- rep(c("ITGB4+", "ITGB4-"), times = c(dim(Positive.data)[1],   
 dim(Negative.data)[1]))  
cell.size <- c(unlist(lapply(rownames(Positive.data), function(x) return(str\_split(x,   
 "\_")[[1]][2]))), unlist(lapply(rownames(Negative.data), function(x) return(str\_split(x,   
 "\_")[[1]][2]))))  
  
  
plot.data$cell.size <- cell.size  
X <- melt(plot.data)

### Melt the data

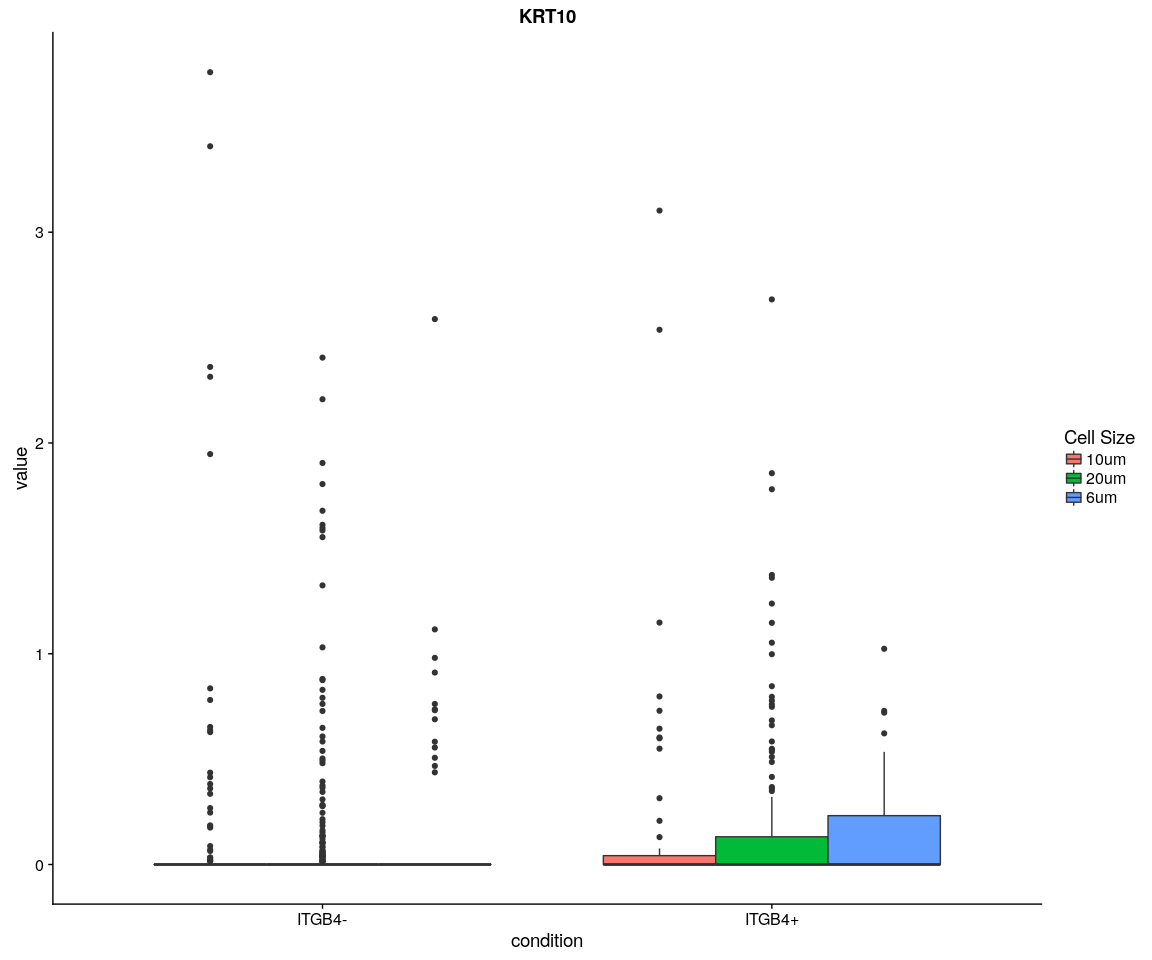
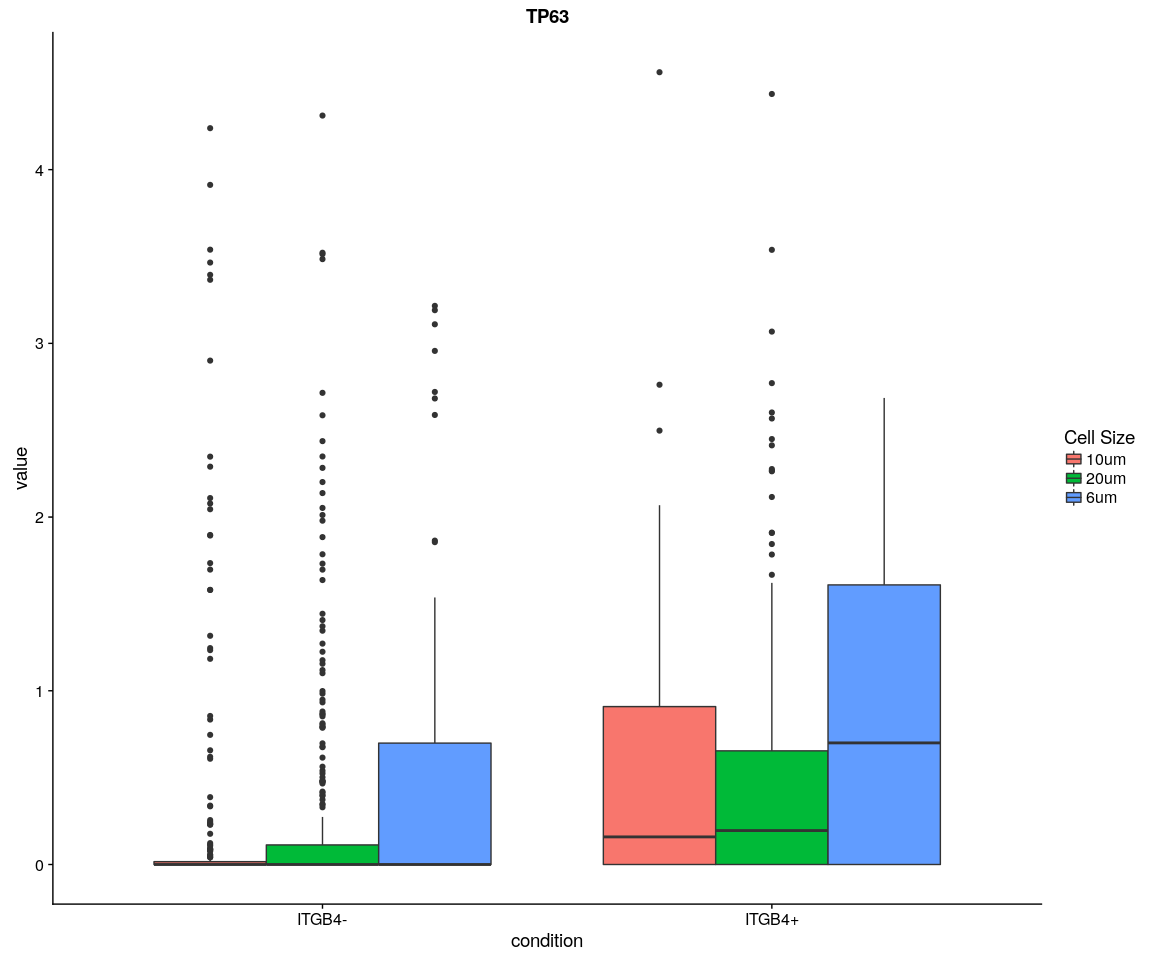
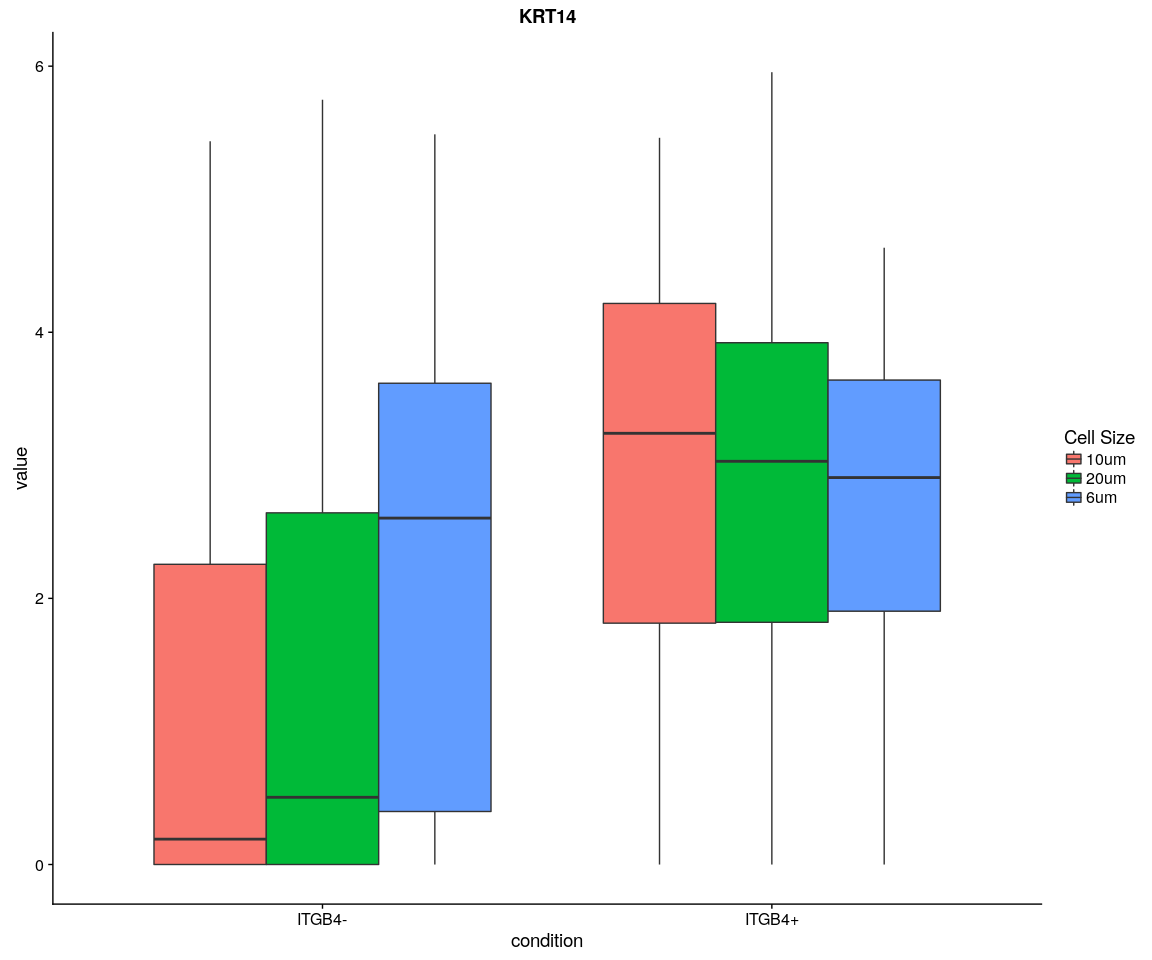
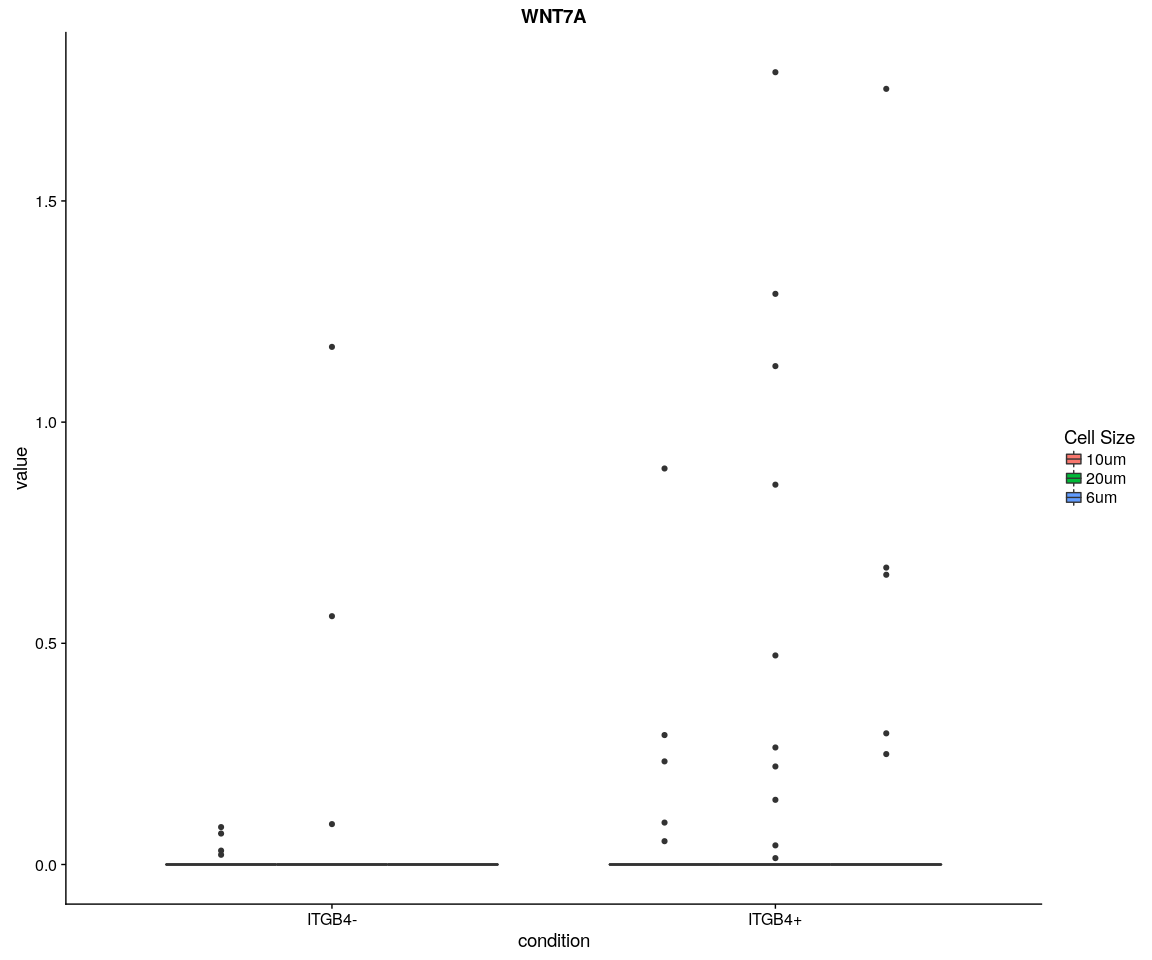
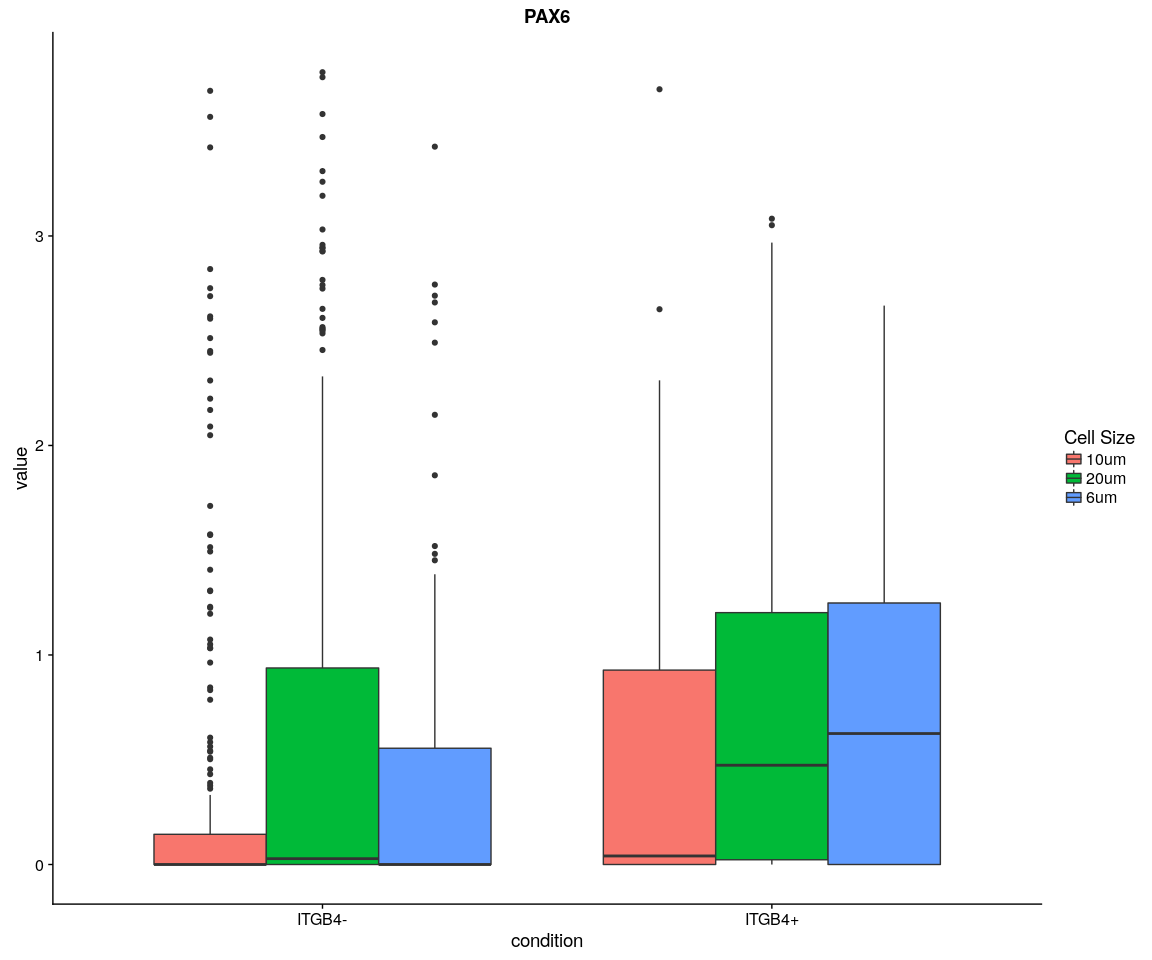
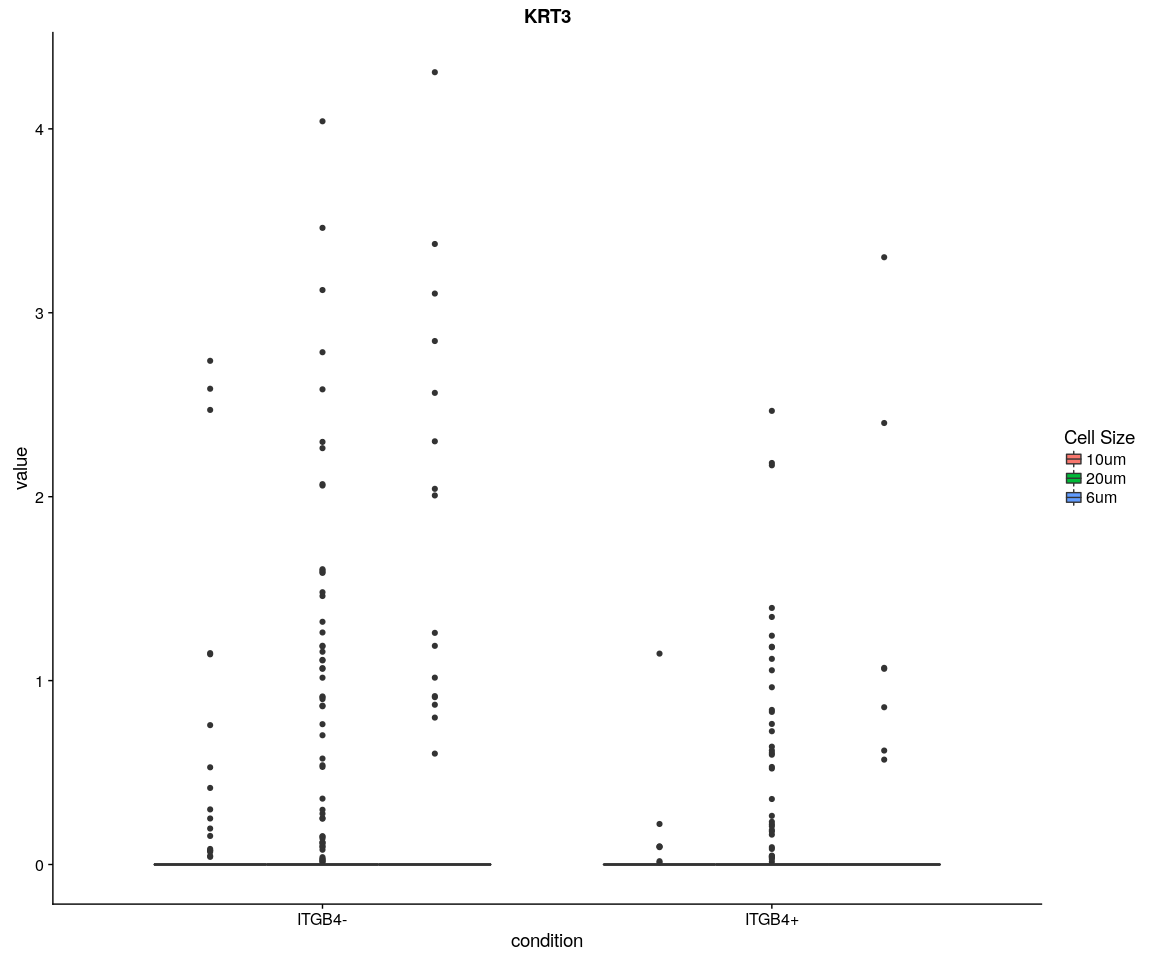
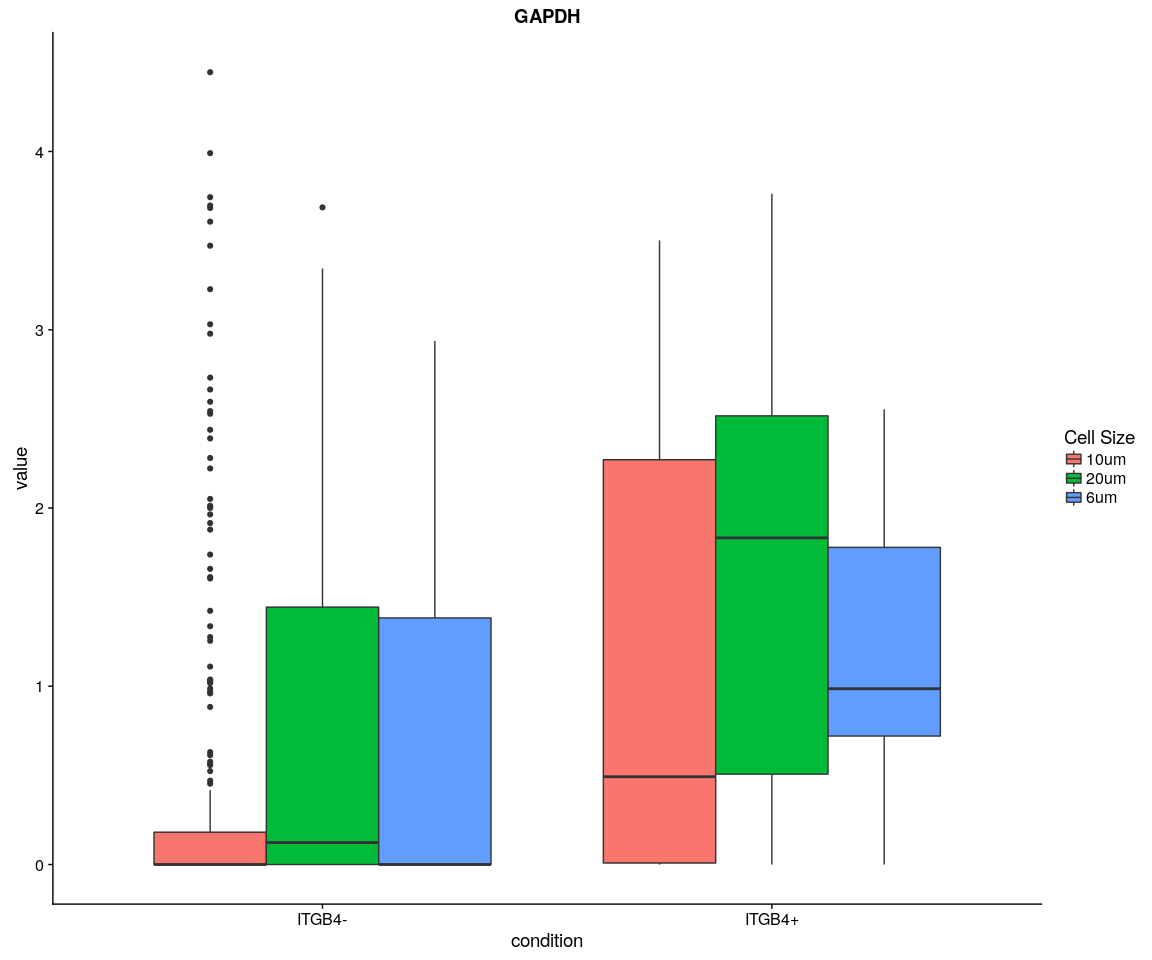
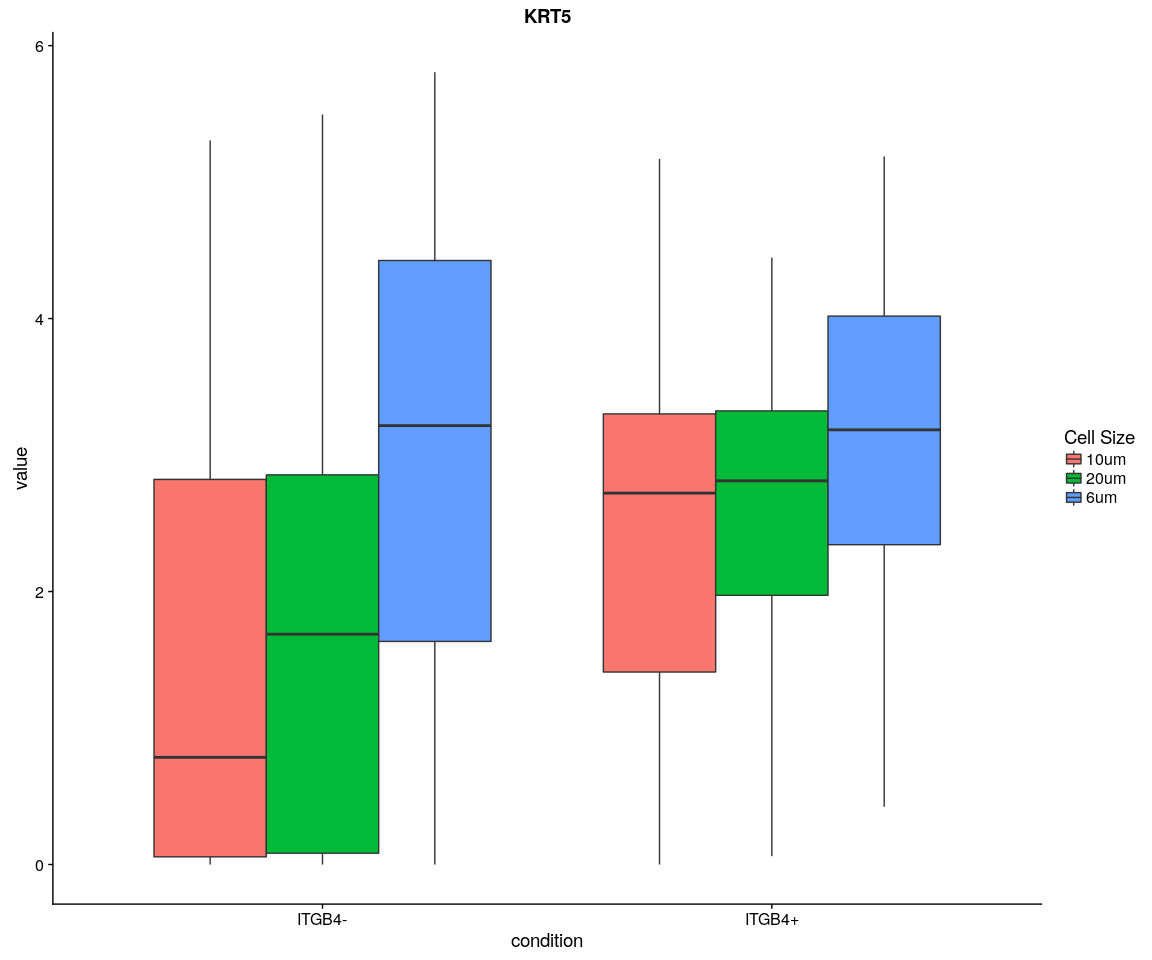
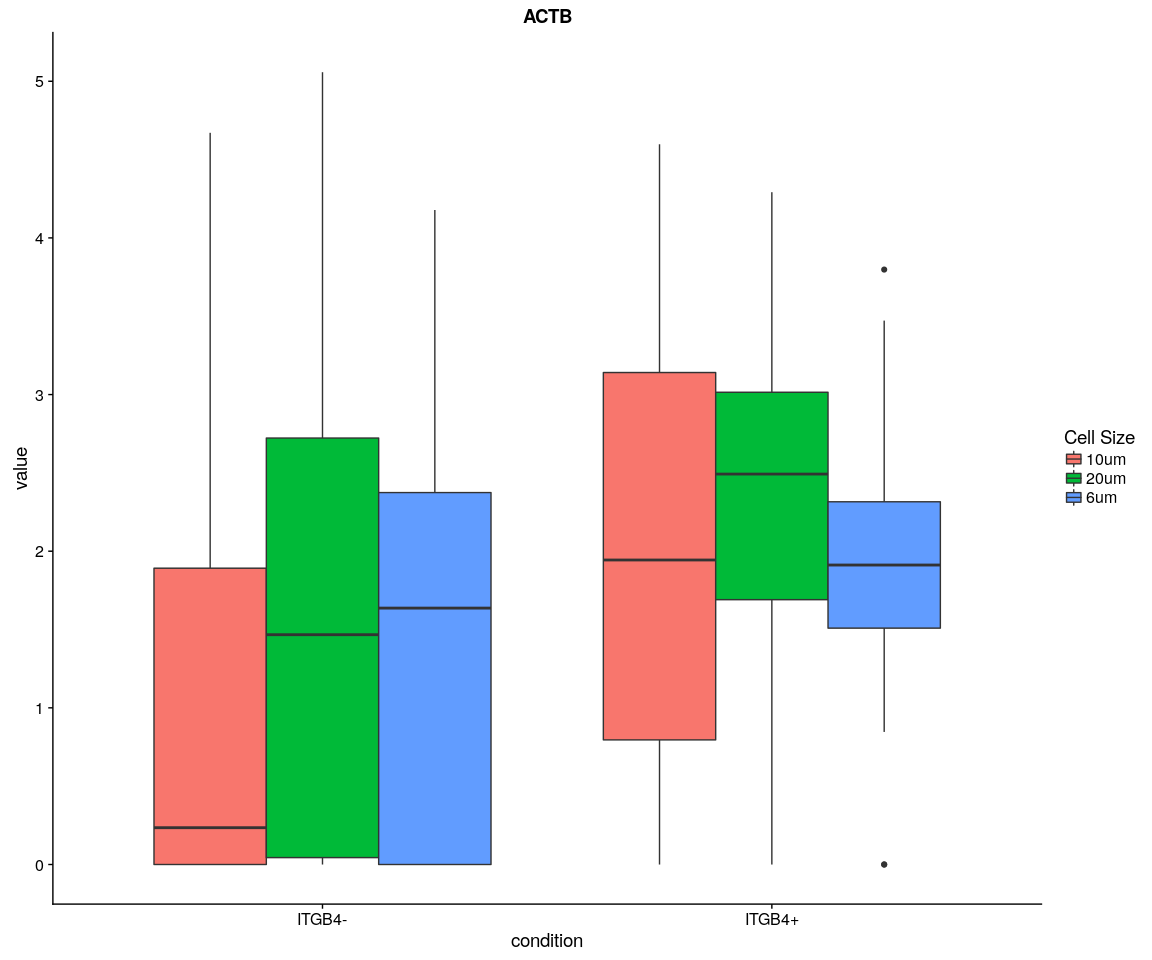
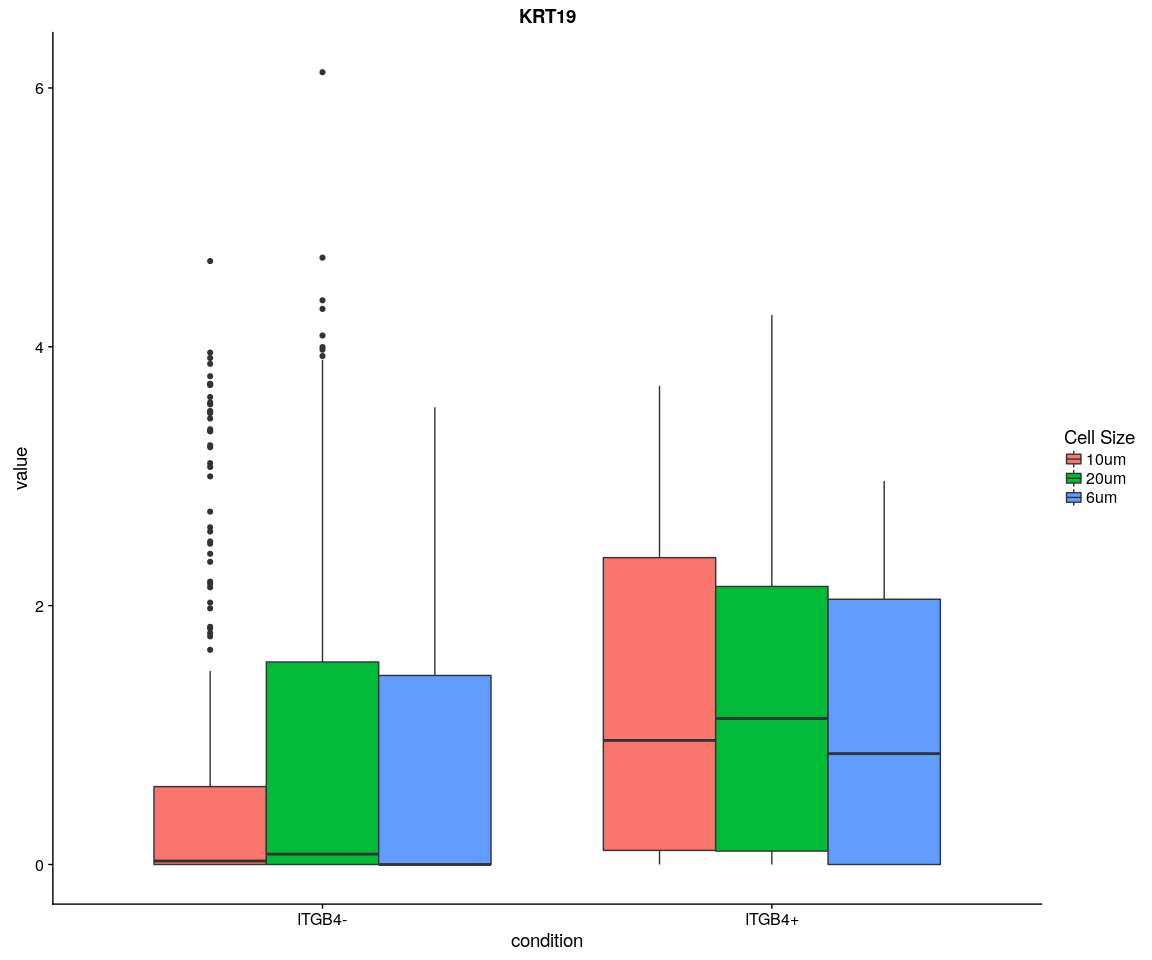
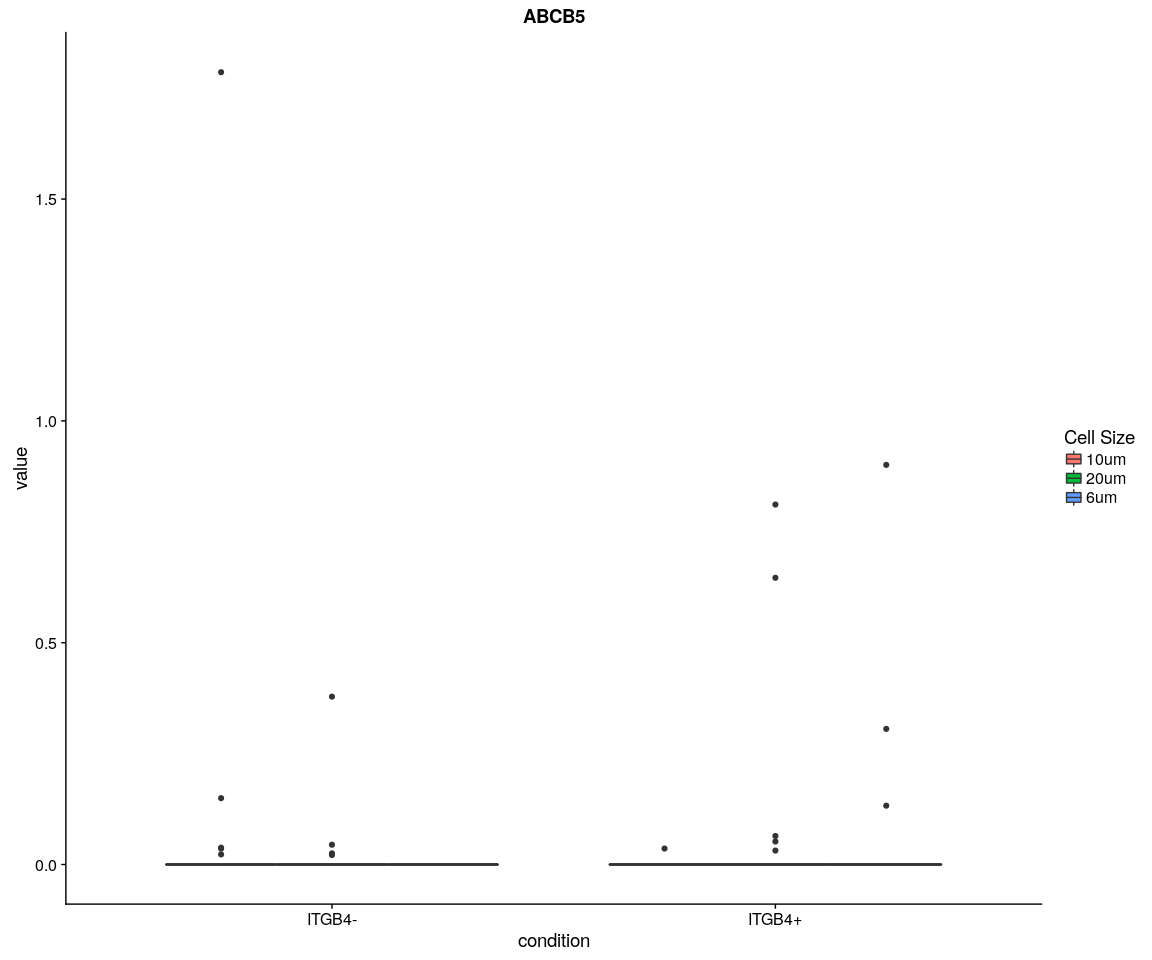
#### Violin

# p<-ggplot(data = X,aes(y=value,x=condition,fill=cell.size))  
# p+geom\_violin(trim = FALSE,scale =  
# 'width')+facet\_wrap(~variable+condition)+  
# geom\_jitter()+guides(fill=guide\_legend(title='Cell Size'))  
  
for (var in as.character(unique(X$variable))) {  
 p <- ggplot(data = X[X$variable == var, ], aes(y = value, x = condition,   
 fill = cell.size))  
 print(p + geom\_violin(trim = FALSE, scale = "width") + geom\_jitter() + guides(fill = guide\_legend(title = "Cell Size")) +   
 ggtitle(label = var))  
}



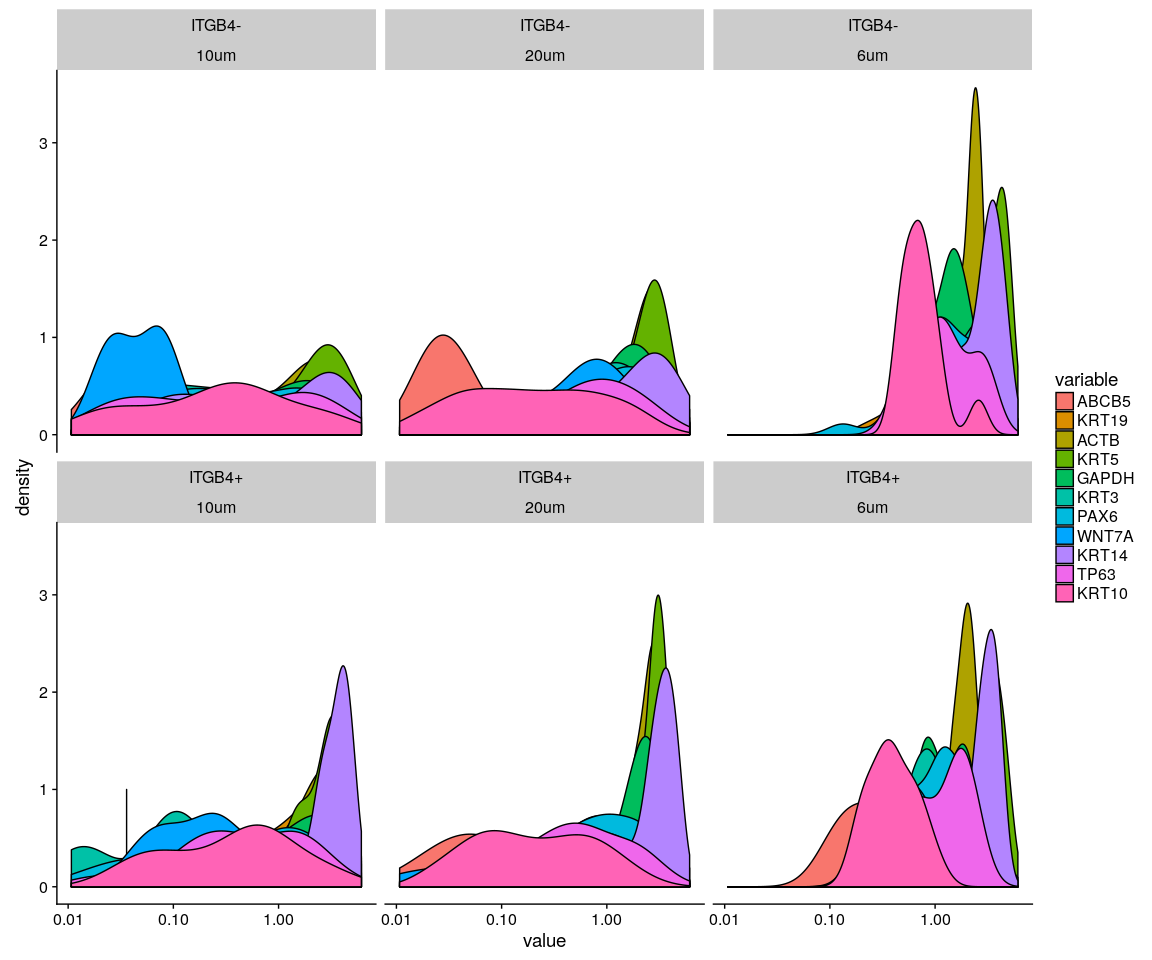
#### Boxplot

# p<-ggplot(data = X,aes(y=value,x=condition,fill=cell.size))  
# p+geom\_boxplot()+guides(fill=guide\_legend(title='Cell  
# Size'))+facet\_wrap(~variable+condition)  
for (var in as.character(unique(X$variable))) {  
 p <- ggplot(data = X[X$variable == var, ], aes(y = value, x = condition,   
 fill = cell.size))  
 print(p + geom\_boxplot() + guides(fill = guide\_legend(title = "Cell Size")) +   
 ggtitle(label = var))  
}

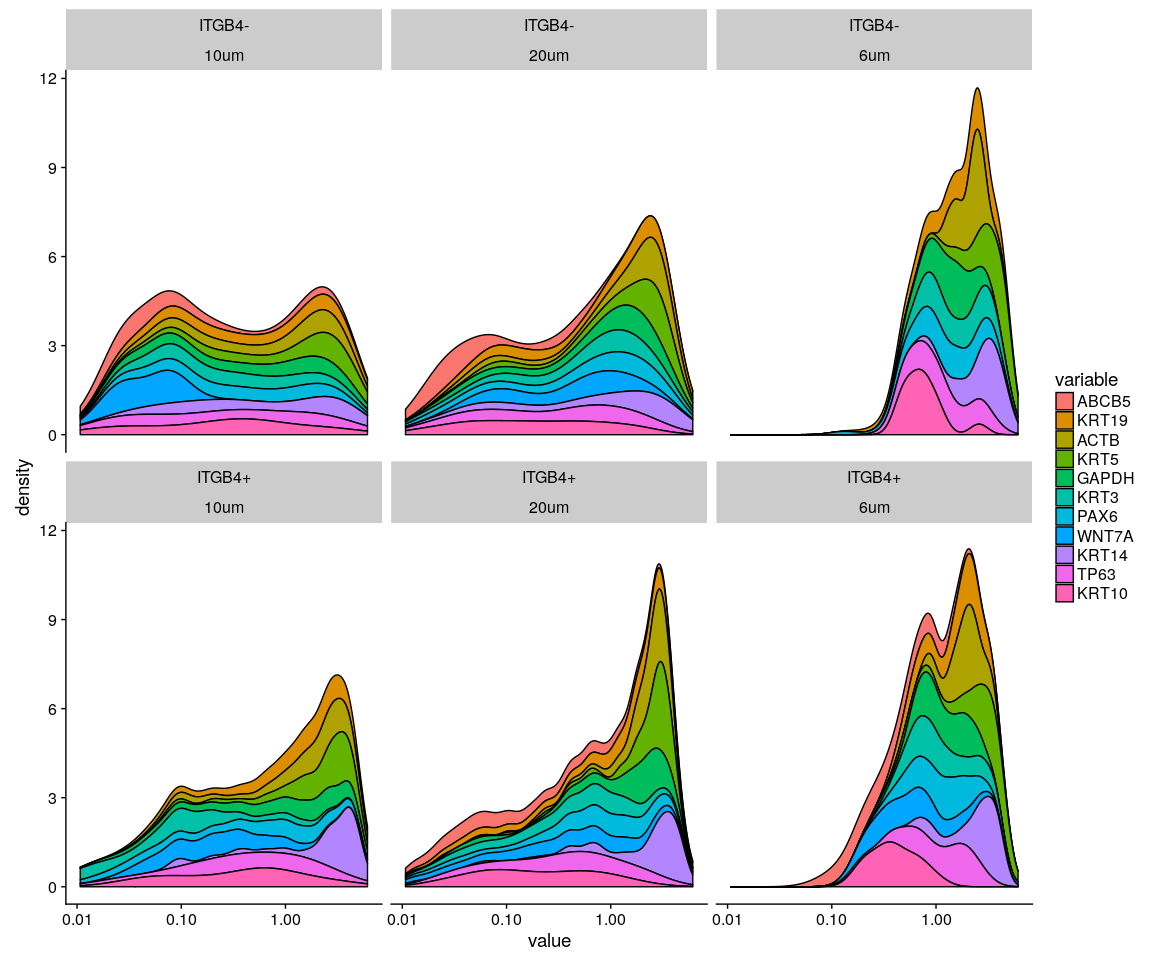


#### Density,histogram

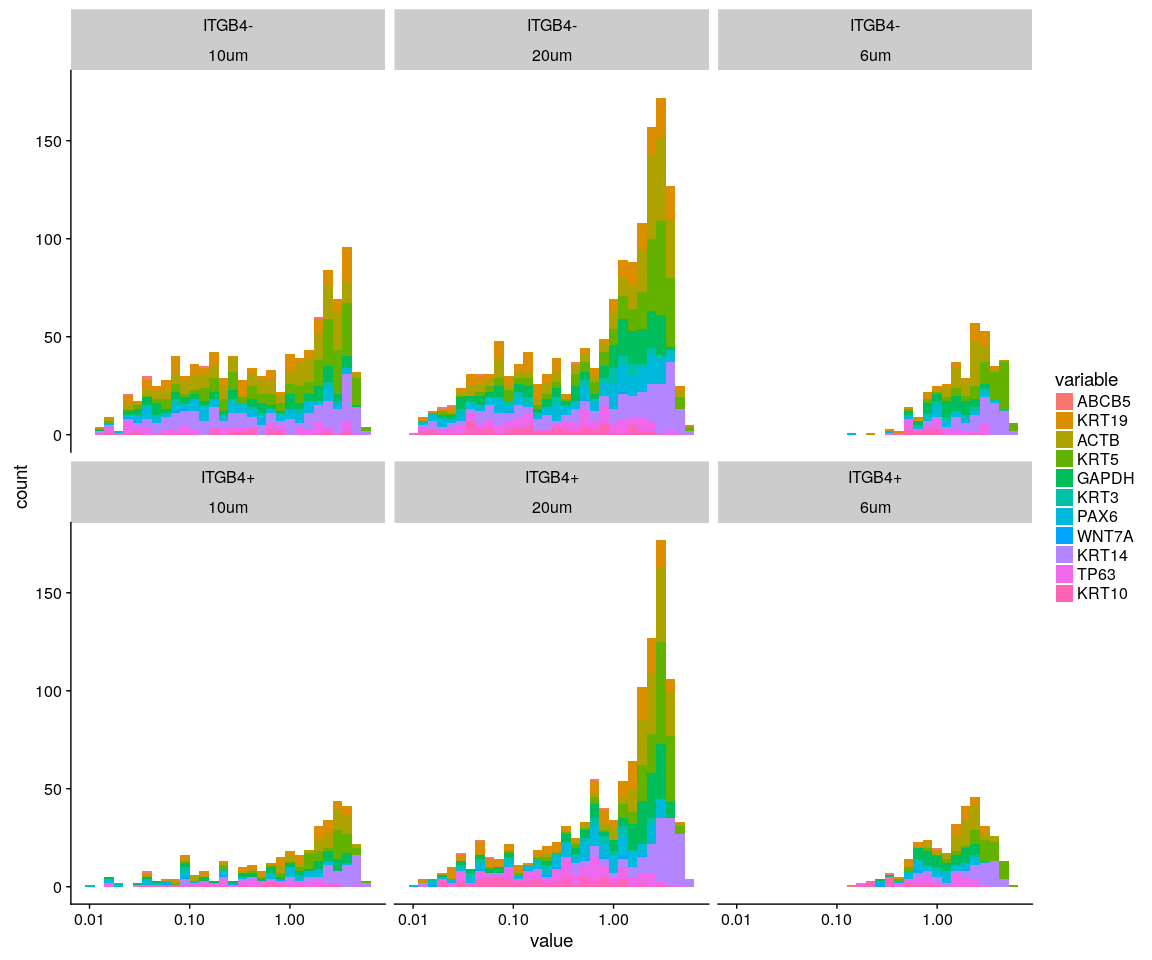
ggplot(data = X, aes(x = value, fill = variable)) + geom\_density(kernel = "gaussian") +   
 scale\_x\_log10() + facet\_wrap(~condition + cell.size)



ggplot(data = X, aes(x = value, fill = variable)) + geom\_density(kernel = "gaussian",   
 position = "stack") + scale\_x\_log10() + facet\_wrap(~condition + cell.size)



ggplot(data = X, aes(x = value, fill = variable)) + geom\_histogram() + scale\_x\_log10() +   
 facet\_wrap(~condition + cell.size)

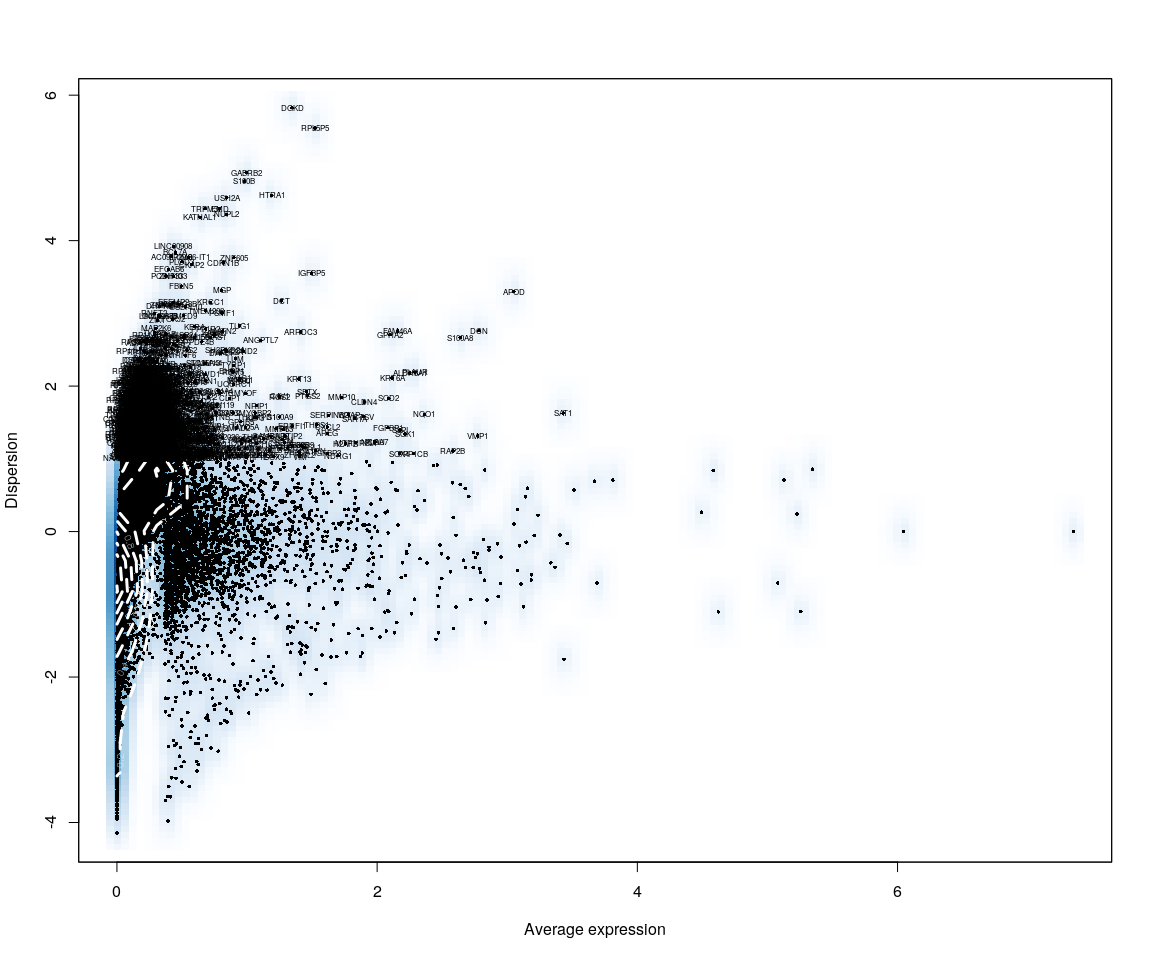


ITGB4 <- as.integer(human.only.pro["ITGB4", ])  
Positive.idx <- which(ITGB4 > 0)  
Negative.idx <- which(ITGB4 == 0)  
Positive.data <- human.only.pro[, Positive.idx, drop = FALSE]  
Negative.data <- human.only.pro[, Negative.idx, drop = FALSE]

### Create Seurat object and not caculate DESeq

Positive.pbmc <- DESeq\_SeuratObj(X = Positive.data, min.cells = 10, min.genes = 2)

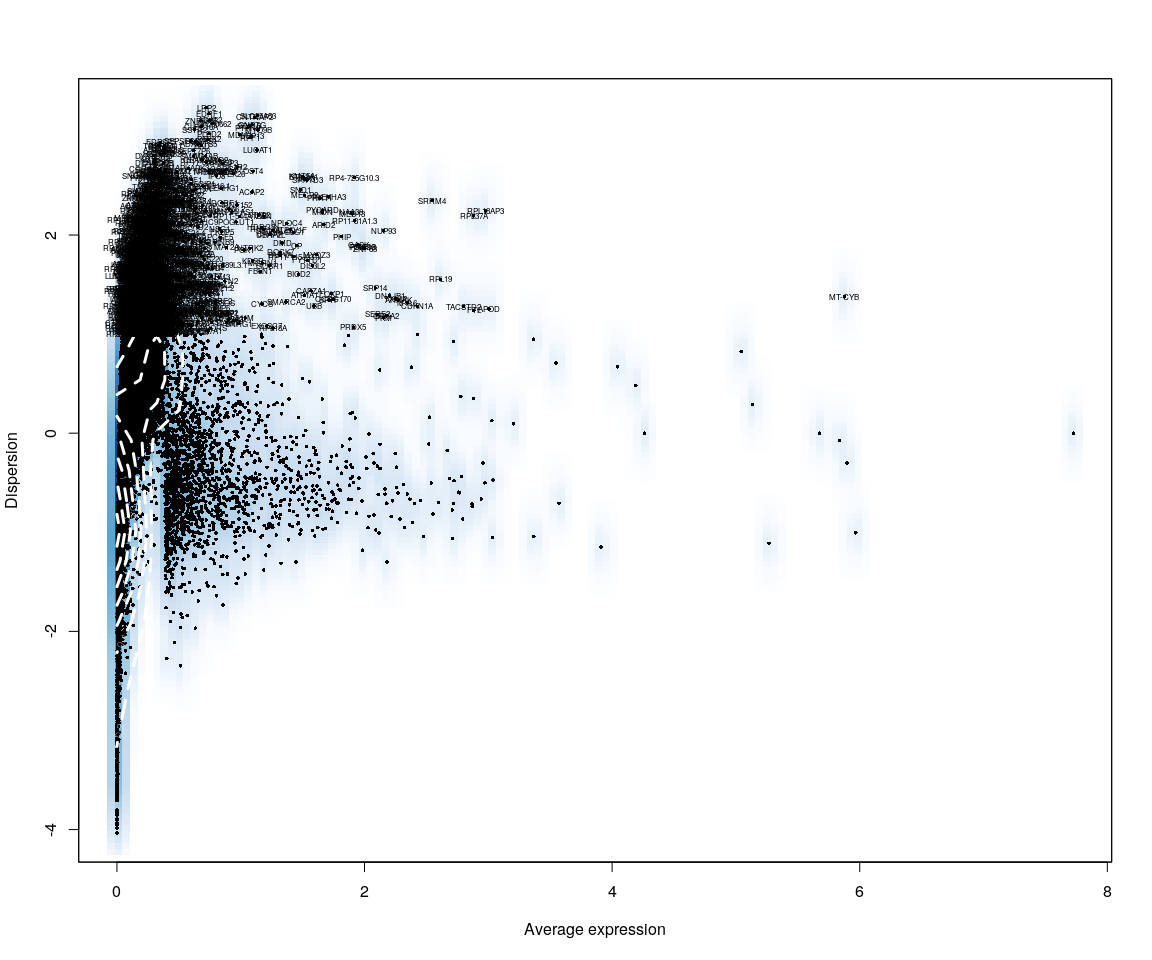
## [1] "Scaling data matrix"  
##   
 |   
 | | 0%  
 |   
 |=================================================================| 100%



Positive.sample.group <- unlist(lapply(Positive.pbmc@cell.names, function(x) return(str\_split(x,   
 "\_")[[1]][1])))  
Positive.sample.cellsize <- unlist(lapply(Positive.pbmc@cell.names, function(x) return(str\_split(x,   
 "\_")[[1]][2])))  
  
Positive.pbmc <- SetIdent(Positive.pbmc, cells.use = Positive.pbmc@cell.names,   
 ident.use = Positive.sample.cellsize)

Negative.pbmc <- DESeq\_SeuratObj(X = Negative.data, min.cells = 10, min.genes = 2)

## [1] "Scaling data matrix"  
##   
 |   
 | | 0%  
 |   
 |=================================================================| 100%



Negative.sample.group <- unlist(lapply(Negative.pbmc@cell.names, function(x) return(str\_split(x,   
 "\_")[[1]][1])))  
Negative.sample.cellsize <- unlist(lapply(Negative.pbmc@cell.names, function(x) return(str\_split(x,   
 "\_")[[1]][2])))  
  
Negative.pbmc <- SetIdent(Negative.pbmc, cells.use = Negative.pbmc@cell.names,   
 ident.use = Negative.sample.cellsize)

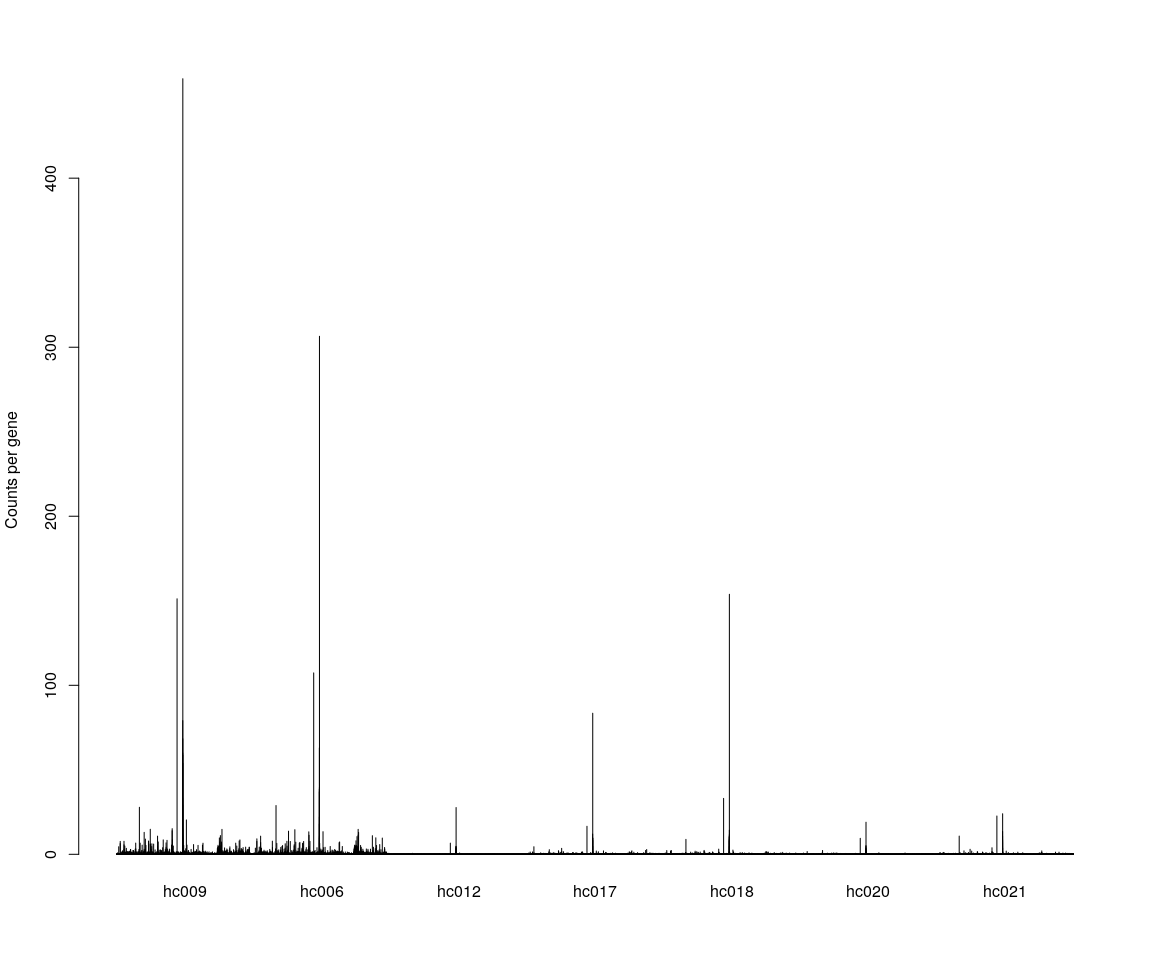
Accordind to the Dispersion vs Avearge expression of Positive and Negative data on ITGB4,they tell us that the although they have similar shape and trend,dispersion of Positive data is more significant than Negative in some genes.

## Step 1: analysis on Positive data

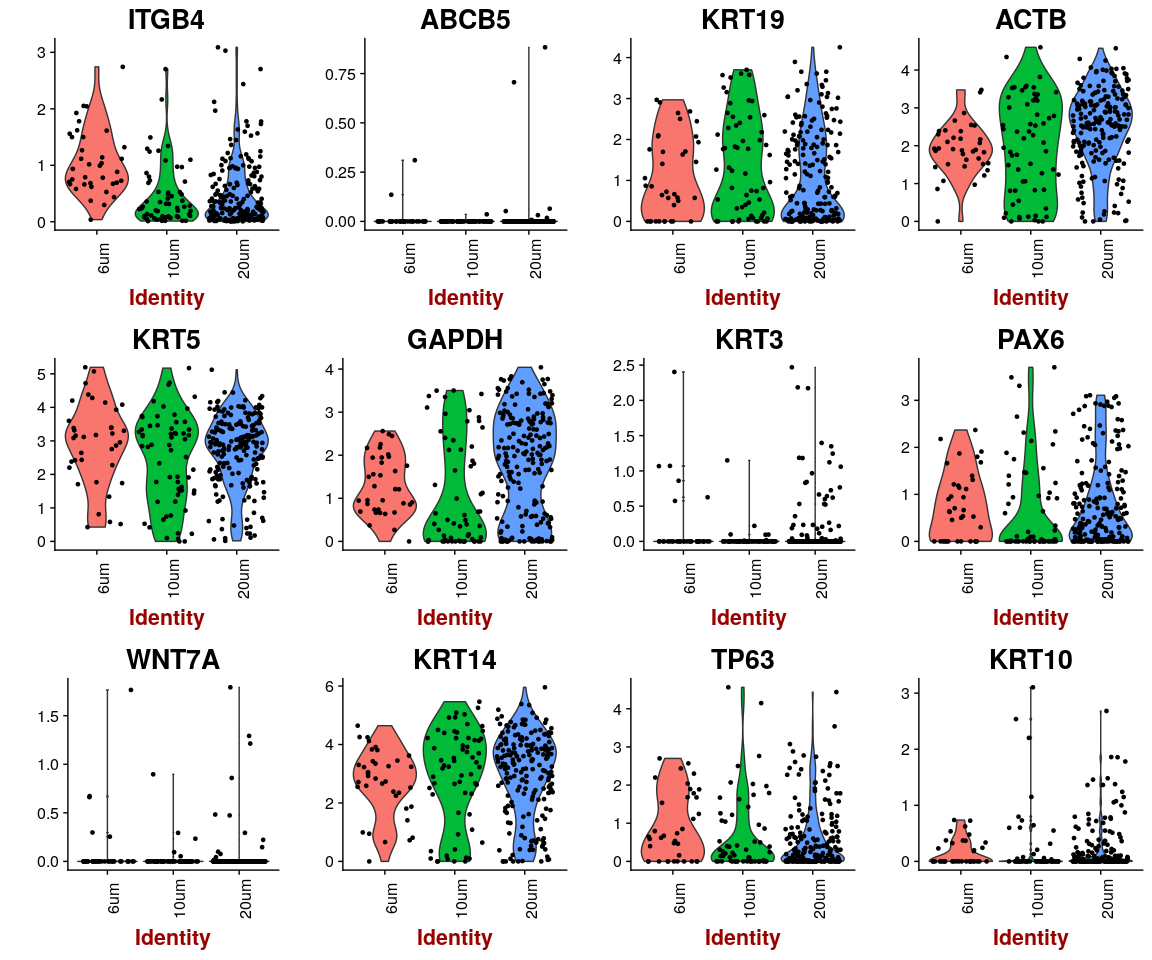
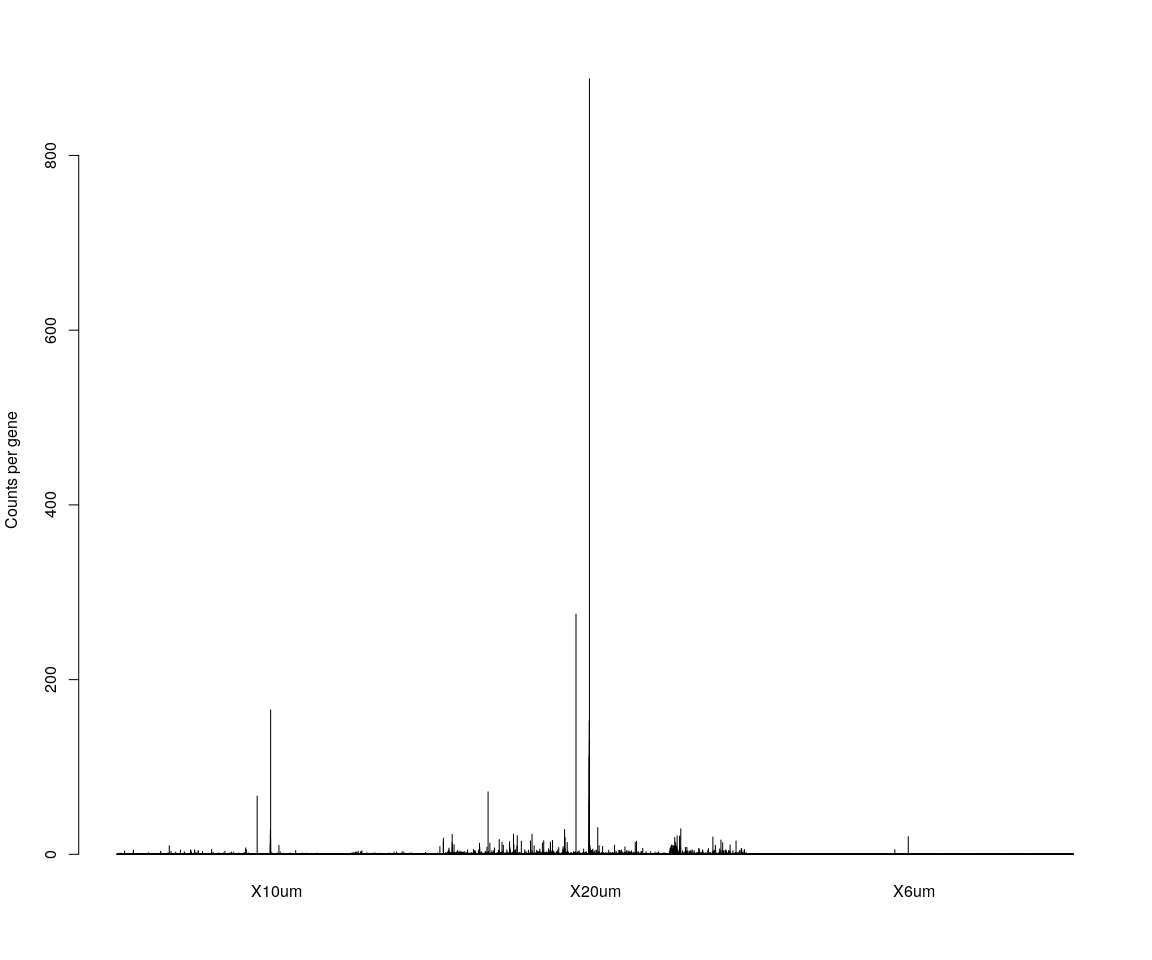
## Figure Explore.2

### First,use the plot,eg. Barplot,Violin…,we can explore some message from sample

Group\_Bar(Positive.pbmc@raw.data, group = Positive.sample.group)



Group\_Bar(Positive.pbmc@raw.data, group = Positive.sample.cellsize)  
  
VlnPlot(Positive.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(Positive.pbmc@raw.data)], y.lab.rot = 90) # Violinn plot of gene ITGB in all sample



## Dimensionality reduction

### **PCA** and **tSNE**

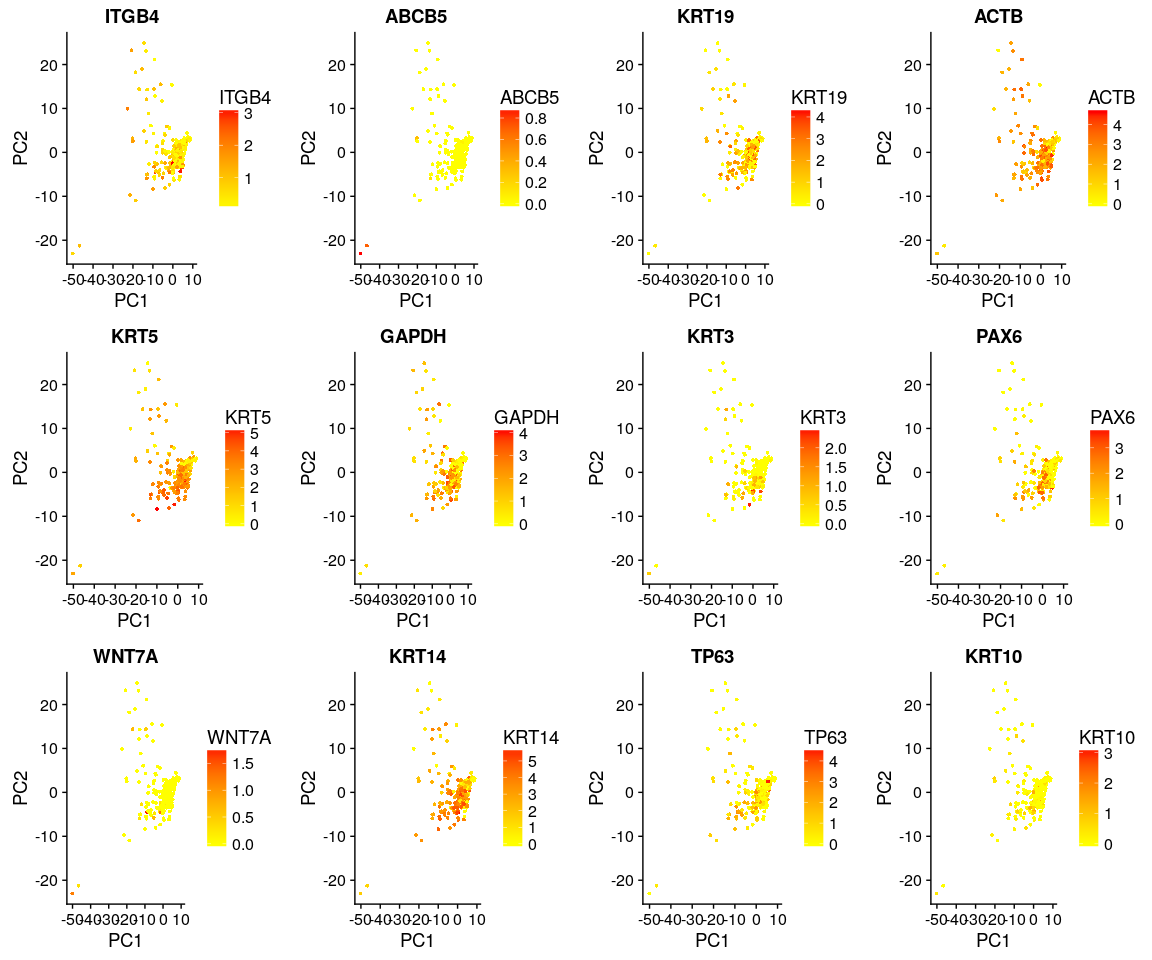
Here,do the dimensionality reduction using the PCA, tSNE method

*It will take a long time to caculate significant pcs.So,here we use the default value*

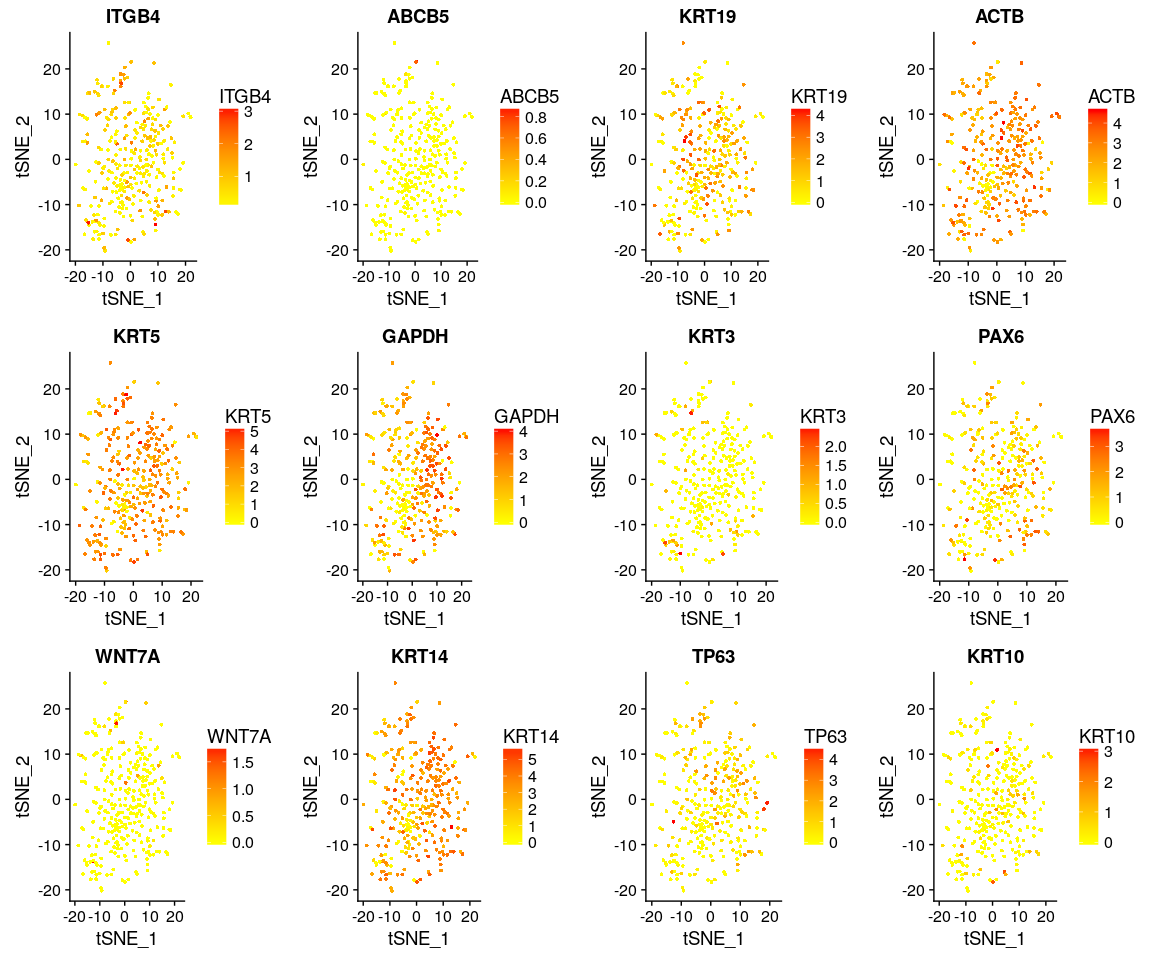
Positive.pbmc <- PCA.TSNE(object = Positive.pbmc, pcs.compute = FALSE, num.pcs = 28)

### After the PCA and tSNE,try plot: Featureplot of **ITGB4**,four var.genes,PCA plot,tSNE plot…

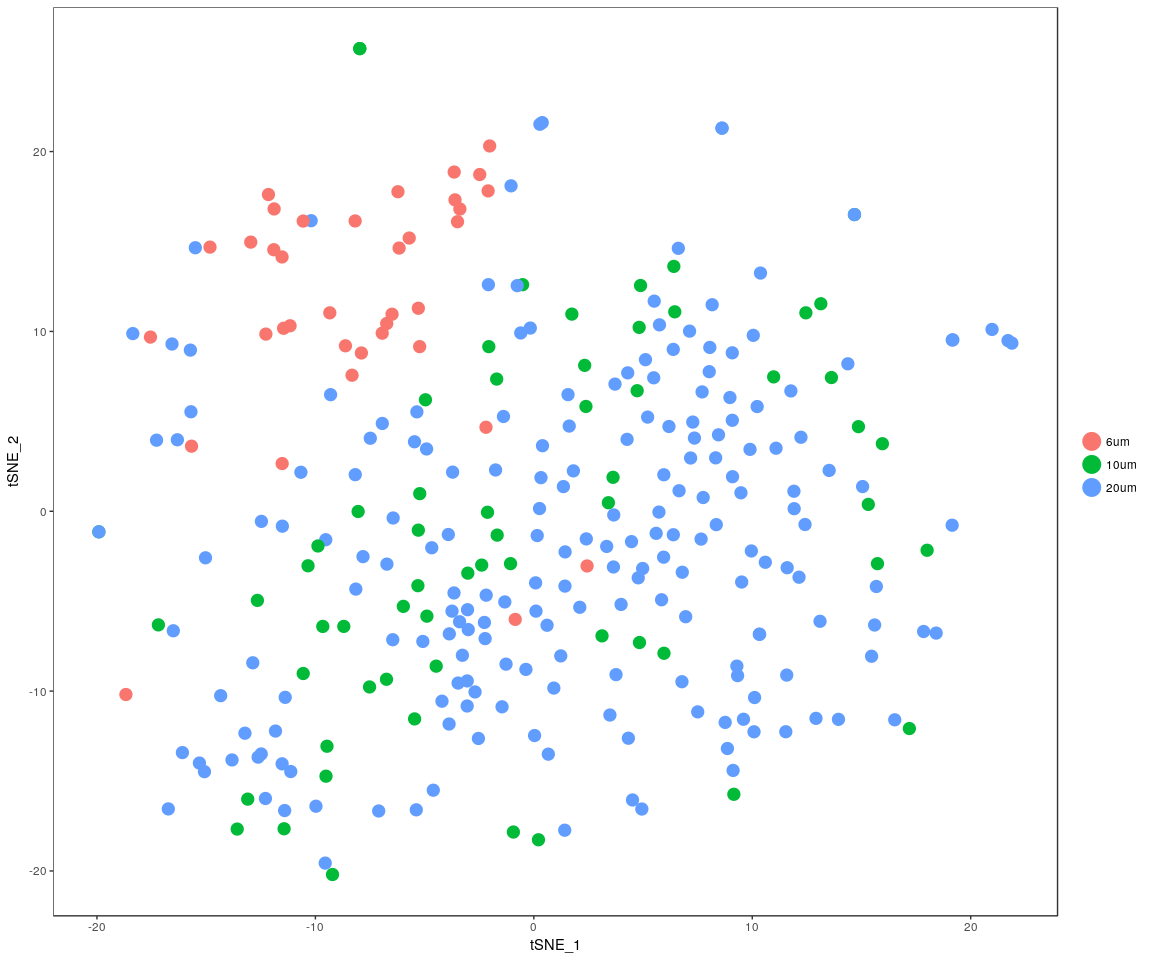
FeaturePlot(object = Positive.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(Positive.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "pca") # ITGB4 gene in part dataset



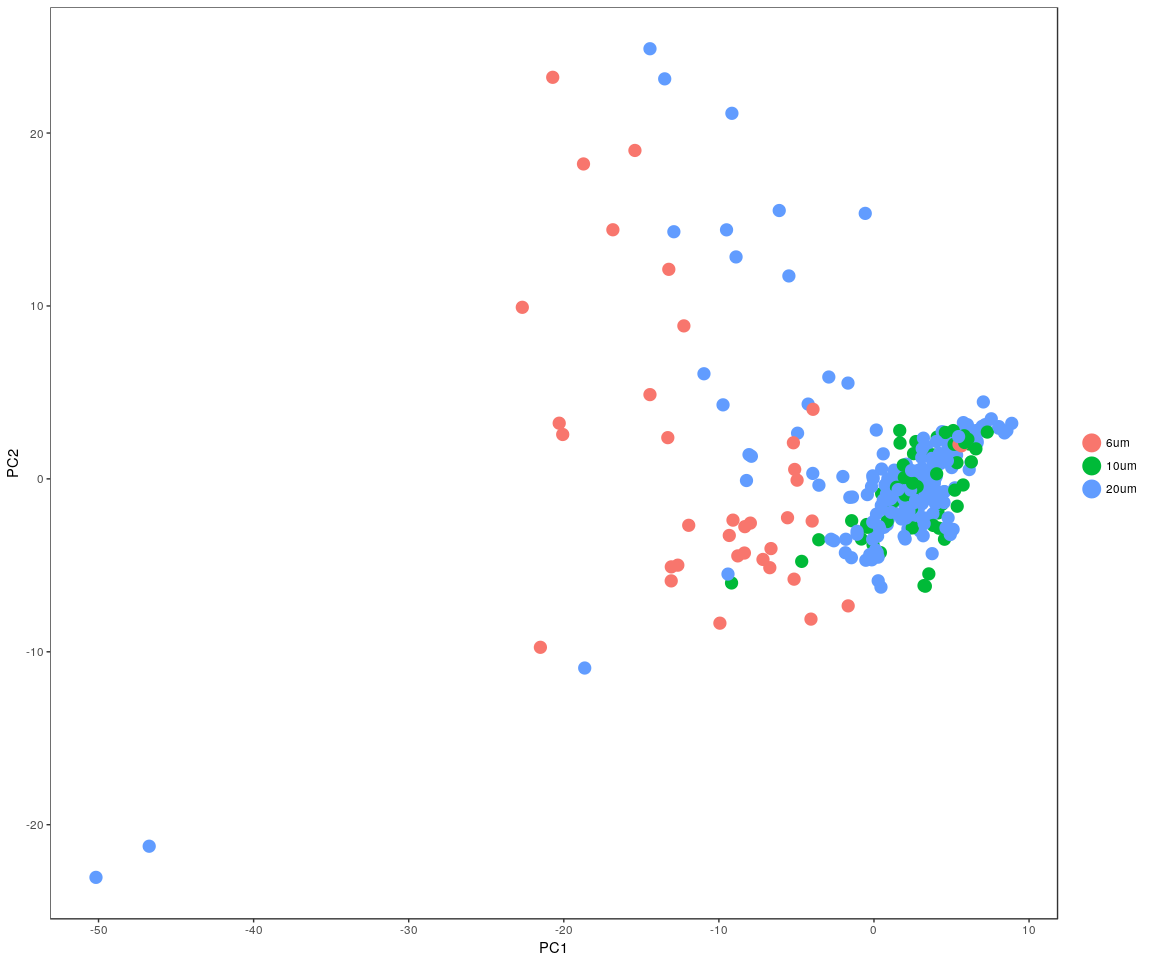
FeaturePlot(object = Positive.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(Positive.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "tsne") # ITGB4 gene in part dataset



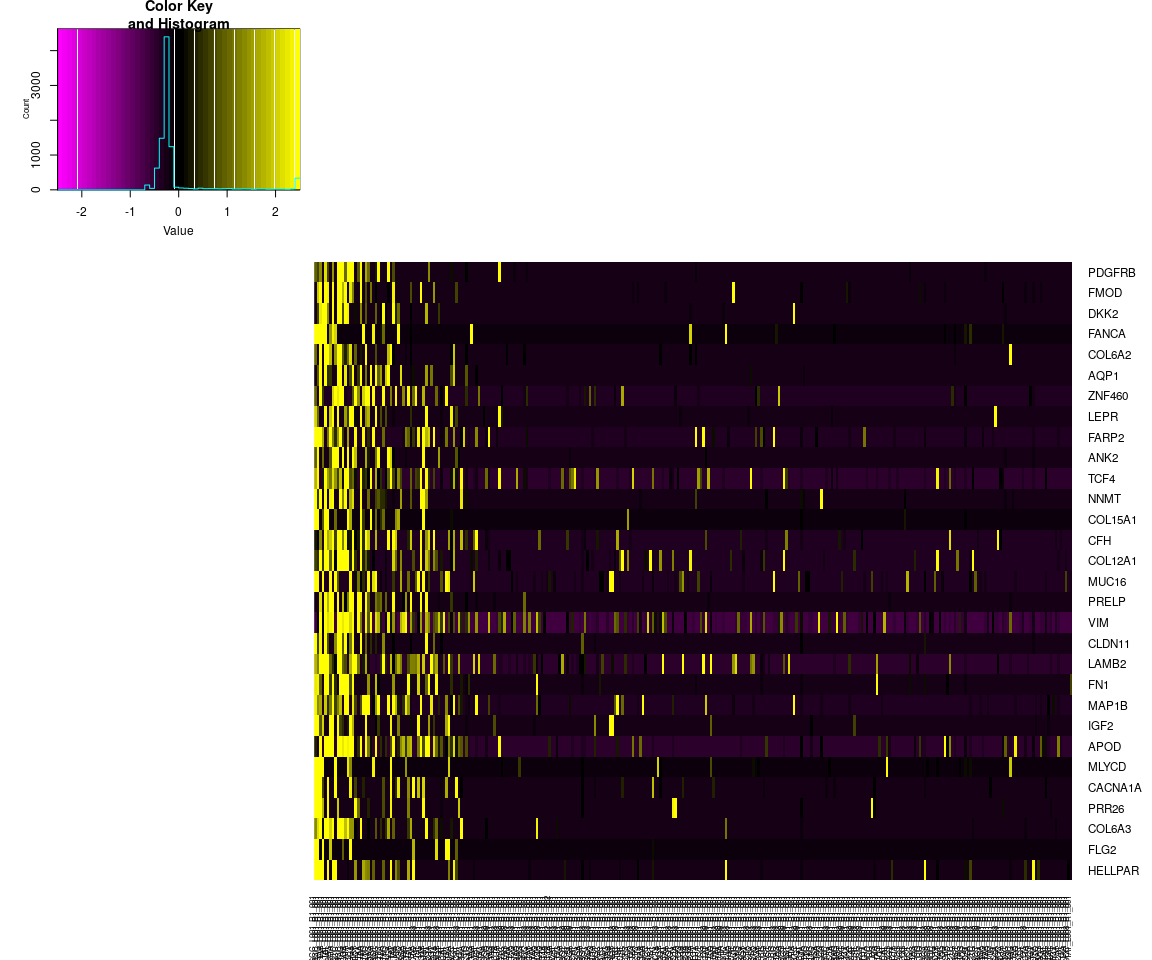
DimPlot(Positive.pbmc, reduction.use = "tsne", pt.size = 4) # grour by sample



DimPlot(Positive.pbmc, reduction.use = "pca", pt.size = 4) # grour by sample



DimHeatmap(Positive.pbmc, reduction.type = "pca", check.plot = FALSE)



The Faetureplot of ITGB4, ABCB5, KRT19, ACTB, KRT12, KRT5, GAPDH, KRT3, PAX6, WNT7A, KRT14, TP63, KRT10based on **PCA** shows that,they only has high expression level in few samples,and expresss lowly in most sample.It means that may be these important genes express differently across sample.The plot also tell us the gene **KRT5,GAPDH,PAXX6,KRT14** have more higher expression level than the other important genes.It is consistent with the result of violin plot. About the heatmap,we only show the gene **ITGB4** And the FeatureHeatmap and Heamap also comfirm this phenomeno.We try the other four variable genes,which has the similar result as gene *ITGB4* But the *tSNE* and \* PCA \* plot show that, the sample can not be split apparently.The result may be is not good based on the PCA and tSNE method.

## Differential expression

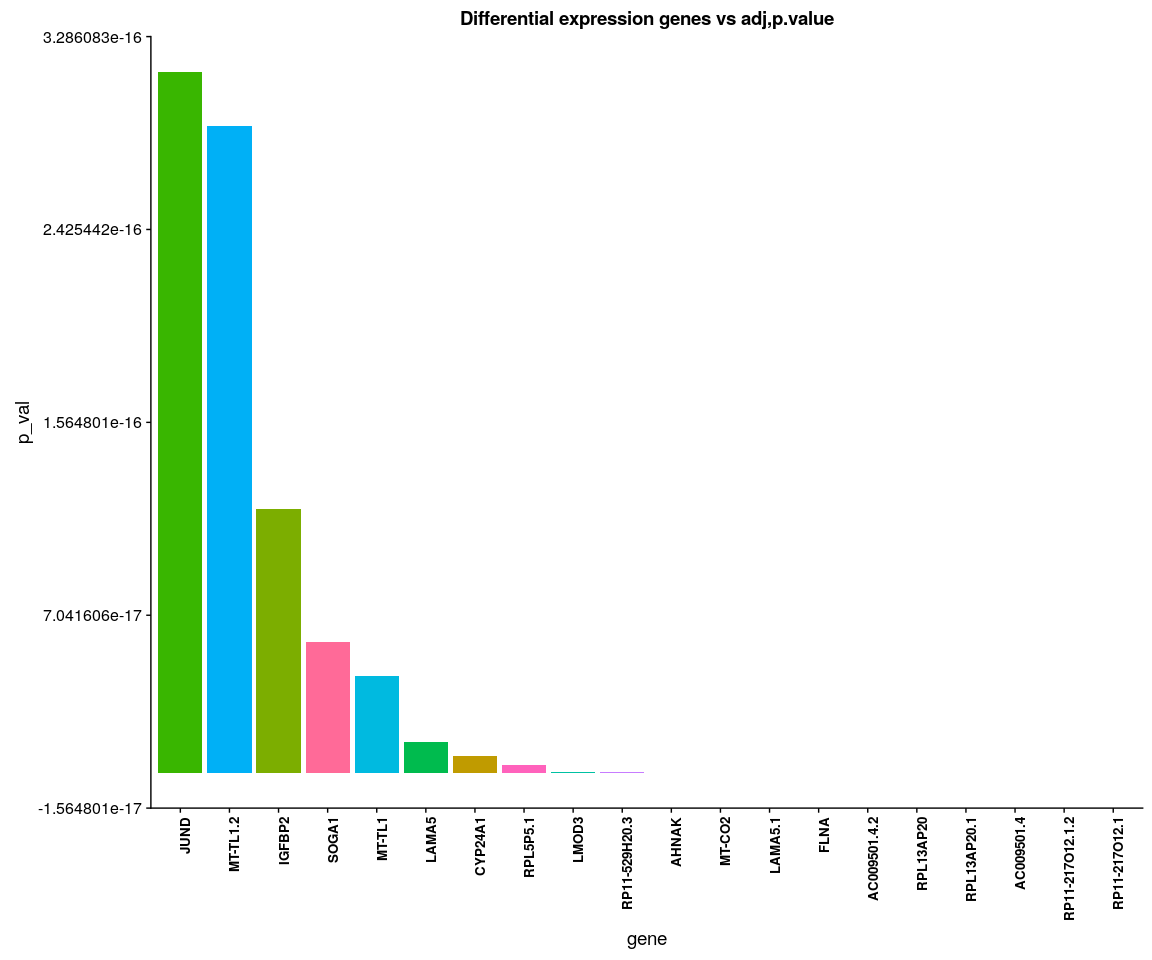
Next,we will have analysis on gene differential expression.Find maker genes across sample.We use the method: \*\*wilcox test\*\*

# Finds markers (differentially expressed genes) for each of the identity  
# classes in a dataset  
Positive.markers <- FindAllMarkers(Positive.pbmc, test.use = "bimod", print.bar = FALSE)  
head(Positive.markers)

## p\_val avg\_logFC pct.1 pct.2 p\_val\_adj cluster  
## RP11-217O12.1 6.454120e-95 2.3017192 1.000 1.000 1.014071e-90 6um  
## AC009501.4 9.144325e-59 2.1192914 0.946 0.322 1.436756e-54 6um  
## FLNA 7.333467e-27 1.1575461 0.811 0.269 1.152234e-22 6um  
## MT-CO2 9.767696e-25 -1.7782097 1.000 1.000 1.534700e-20 6um  
## AHNAK 1.157653e-24 1.5878616 0.973 0.902 1.818904e-20 6um  
## RP11-529H20.3 1.863462e-23 -0.3520236 0.054 0.871 2.927871e-19 6um  
## gene  
## RP11-217O12.1 RP11-217O12.1  
## AC009501.4 AC009501.4  
## FLNA FLNA  
## MT-CO2 MT-CO2  
## AHNAK AHNAK  
## RP11-529H20.3 RP11-529H20.3

We check whether the important genes are still in the marker genes we found from the DESeq analysis. the genes:ITGB4, KRT19, ACTB, GAPDH, KRT10 are still in the marker genes.

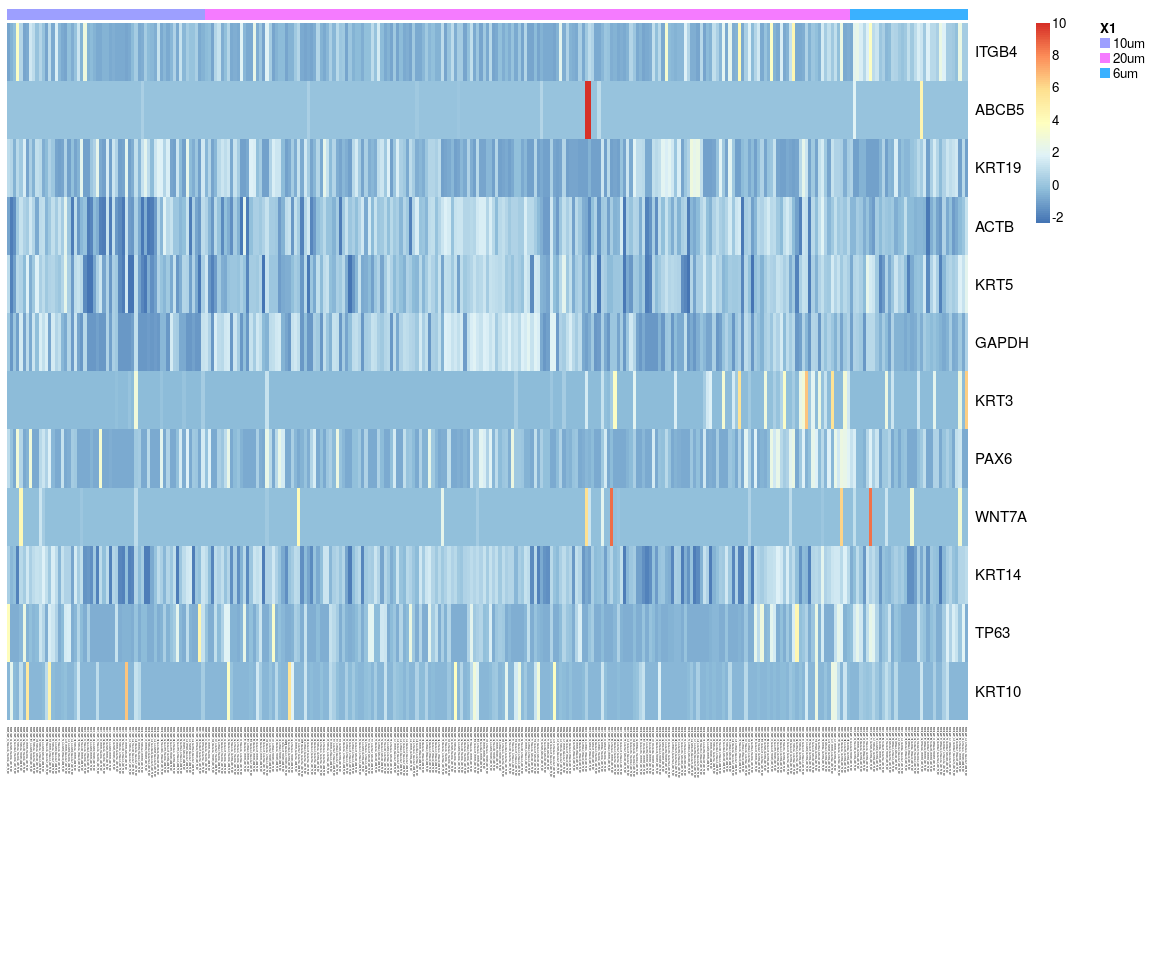
### Bar plot of gene’s p.val



Positive.heatmap <- Heatmap\_fun(genes = important.genes[important.genes %in%   
 rownames(Positive.pbmc@raw.data)], tpm.data = Positive.pbmc@scale.data,   
 condition = unique(as.character(Positive.pbmc@ident)), all.condition = as.character(Positive.pbmc@ident))

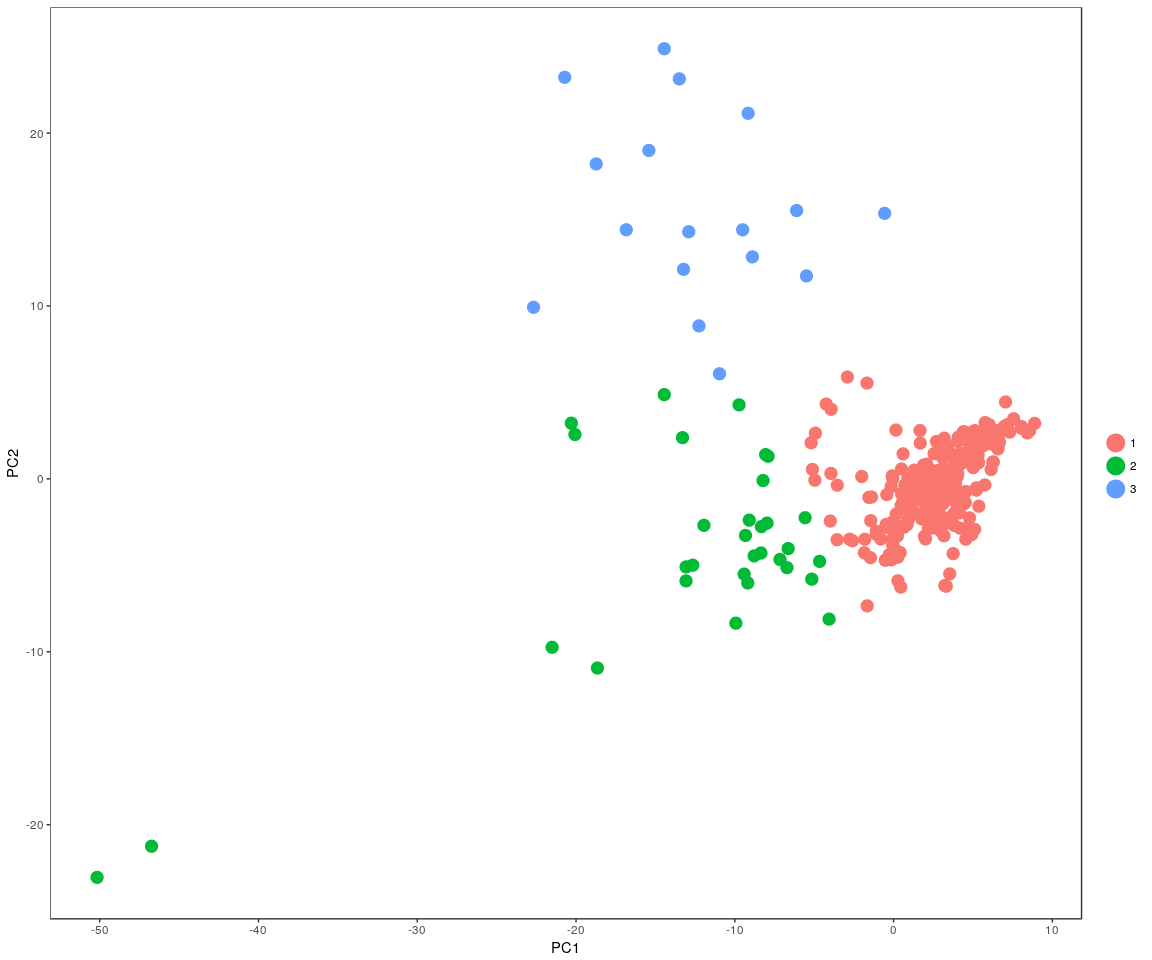
## There ara 3 conditions  
## Whether creat data accurate 0

NMF::aheatmap(Positive.heatmap[[2]], Rowv = NA, Colv = NA, annCol = Positive.heatmap[[1]],   
 scale = "none")

 We have find all marker genes across sample,there are 937 significant genes(adjust p-value <0.05) in all marker genes.

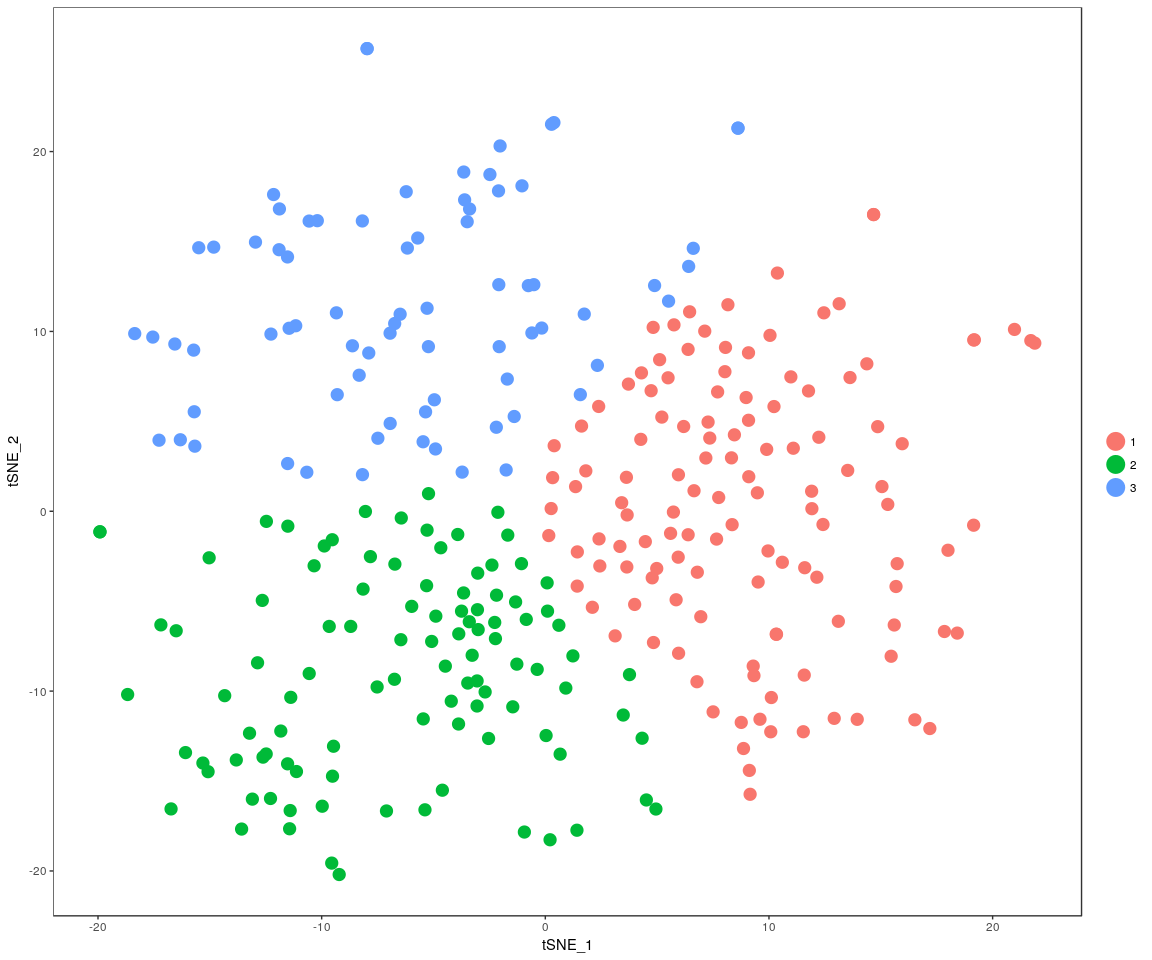
### Next,Spectral k-means clustering on single cells based on PCA

Positive.pbmc <- KClustDimension(Positive.pbmc, reduction.use = "pca", k.use = 3)  
clusters.pca <- Positive.pbmc@meta.data$kdimension.ident  
DimPlot(Positive.pbmc, pt.size = 4, group.by = "kdimension.ident")



### Spectral k-means clustering on single cells based on tSNE

Positive.pbmc <- KClustDimension(Positive.pbmc, reduction.use = "tsne", k.use = 3)  
clusters.tsne <- Positive.pbmc@meta.data$kdimension.ident  
DimPlot(Positive.pbmc, pt.size = 4, group.by = "kdimension.ident", reduction.use = "tsne")

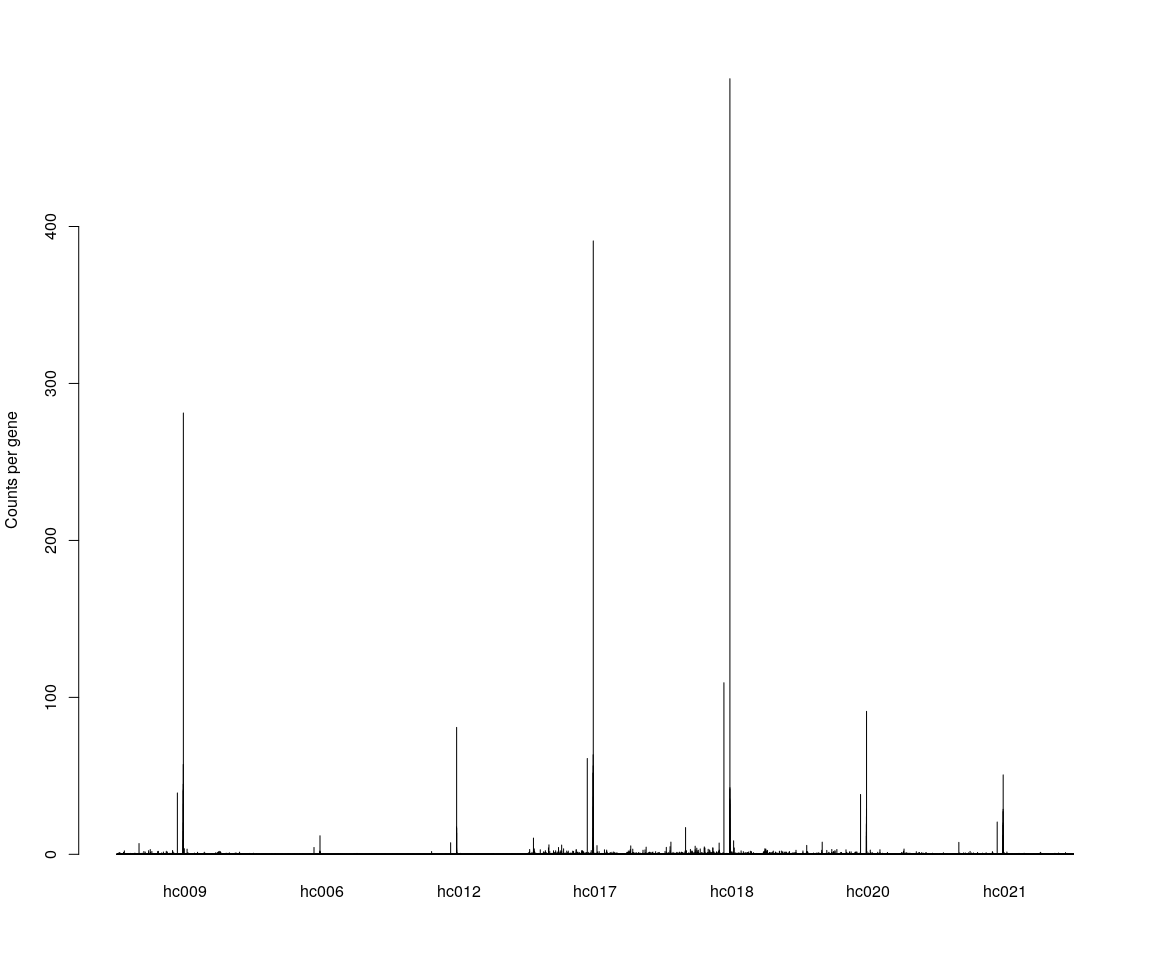


## Step 2: analysis on Negative data

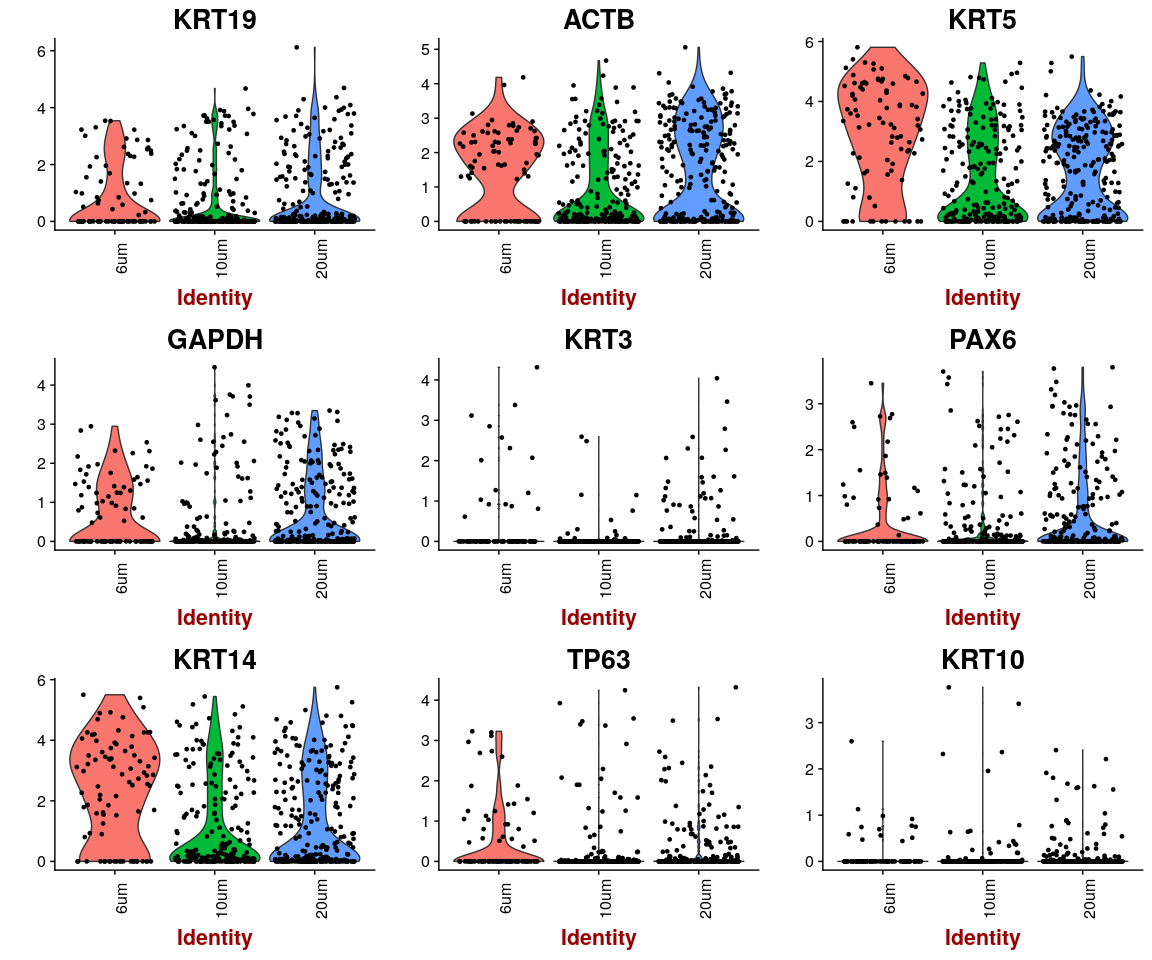
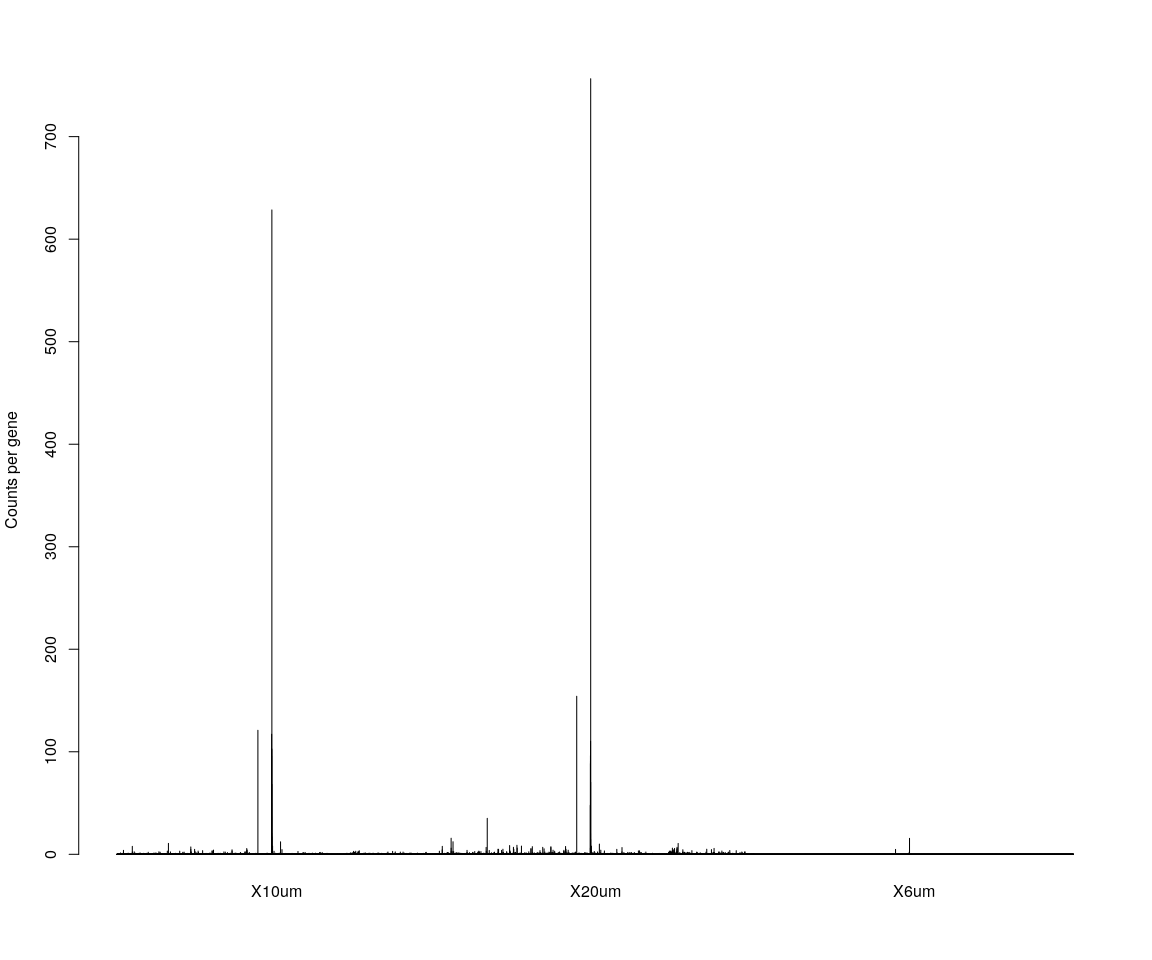
## Figure Explore

### First,use the plot,eg. Barplot,Violin…,we can explore some message from sample

Group\_Bar(Negative.pbmc@raw.data, group = Negative.sample.group)



Group\_Bar(Negative.pbmc@raw.data, group = Negative.sample.cellsize)  
  
VlnPlot(Negative.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(Negative.pbmc@raw.data)], y.lab.rot = 90) # Violinn plot of gene ITGB in all sample



## Dimensionality reduction

### **PCA** and **tSNE**

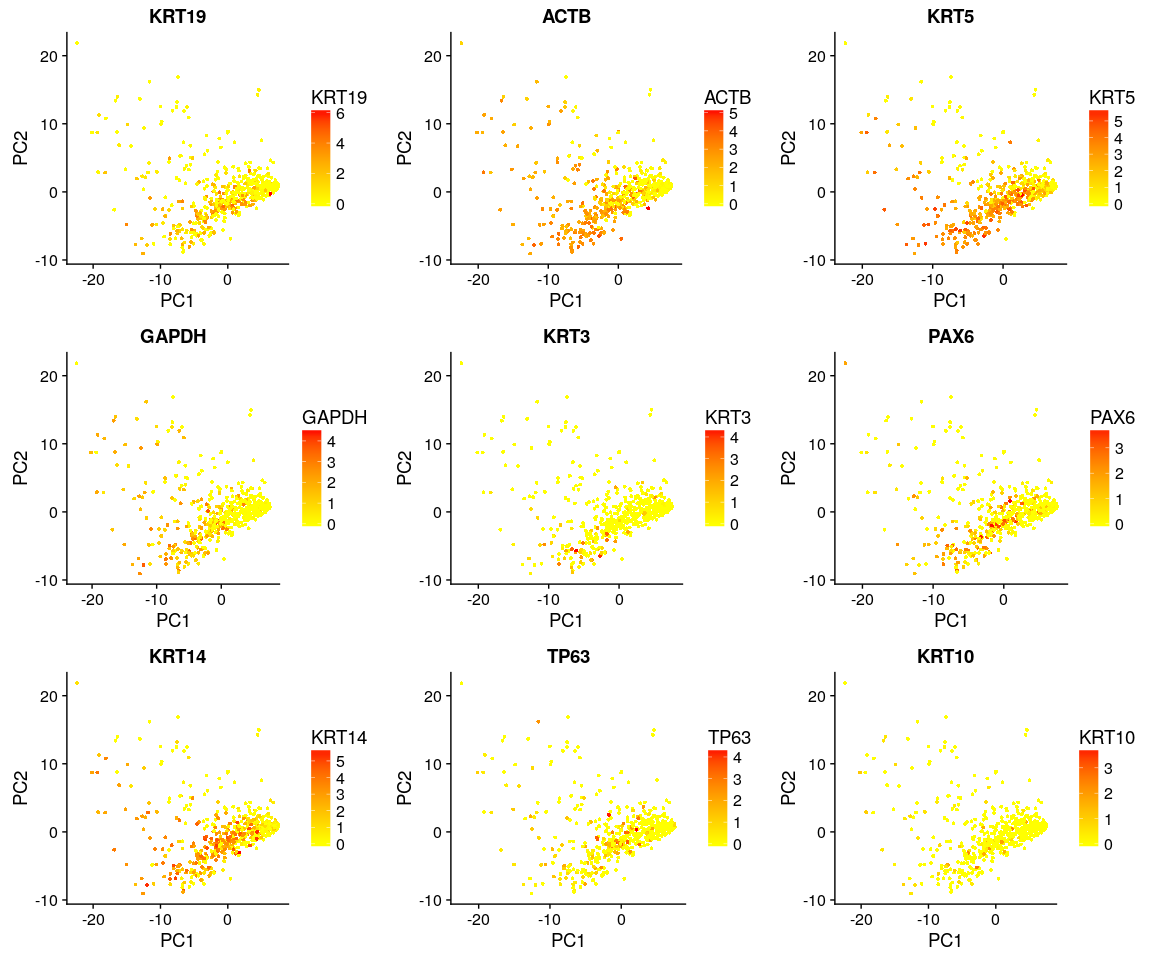
Here,do the dimensionality reduction using the PCA, tSNE method

*It will take a long time to caculate significant pcs.So,here we use the default value*

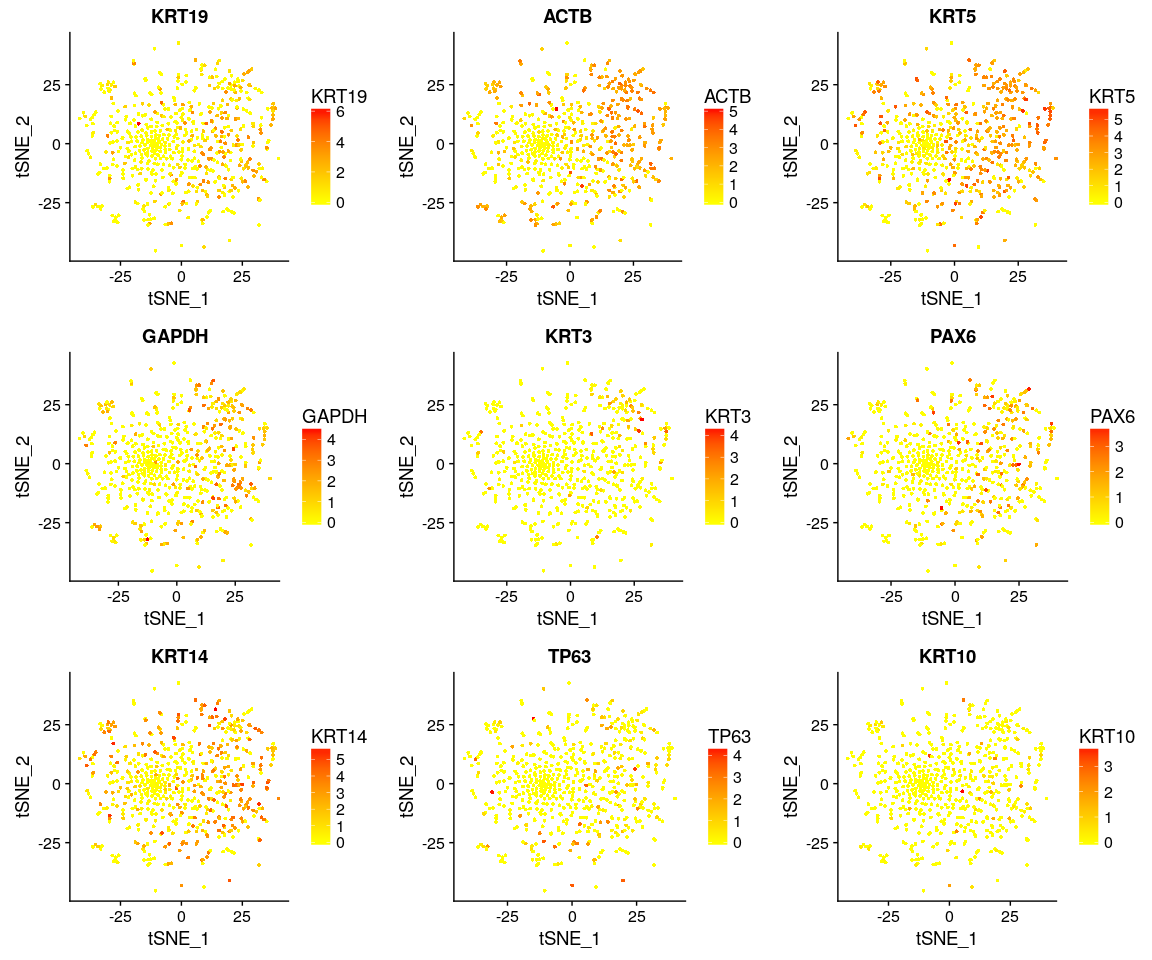
Negative.pbmc <- PCA.TSNE(object = Negative.pbmc, pcs.compute = FALSE, num.pcs = 28)

### After the PCA and tSNE,try plot: Featureplot of **ITGB4**,four var.genes,PCA plot,tSNE plot…

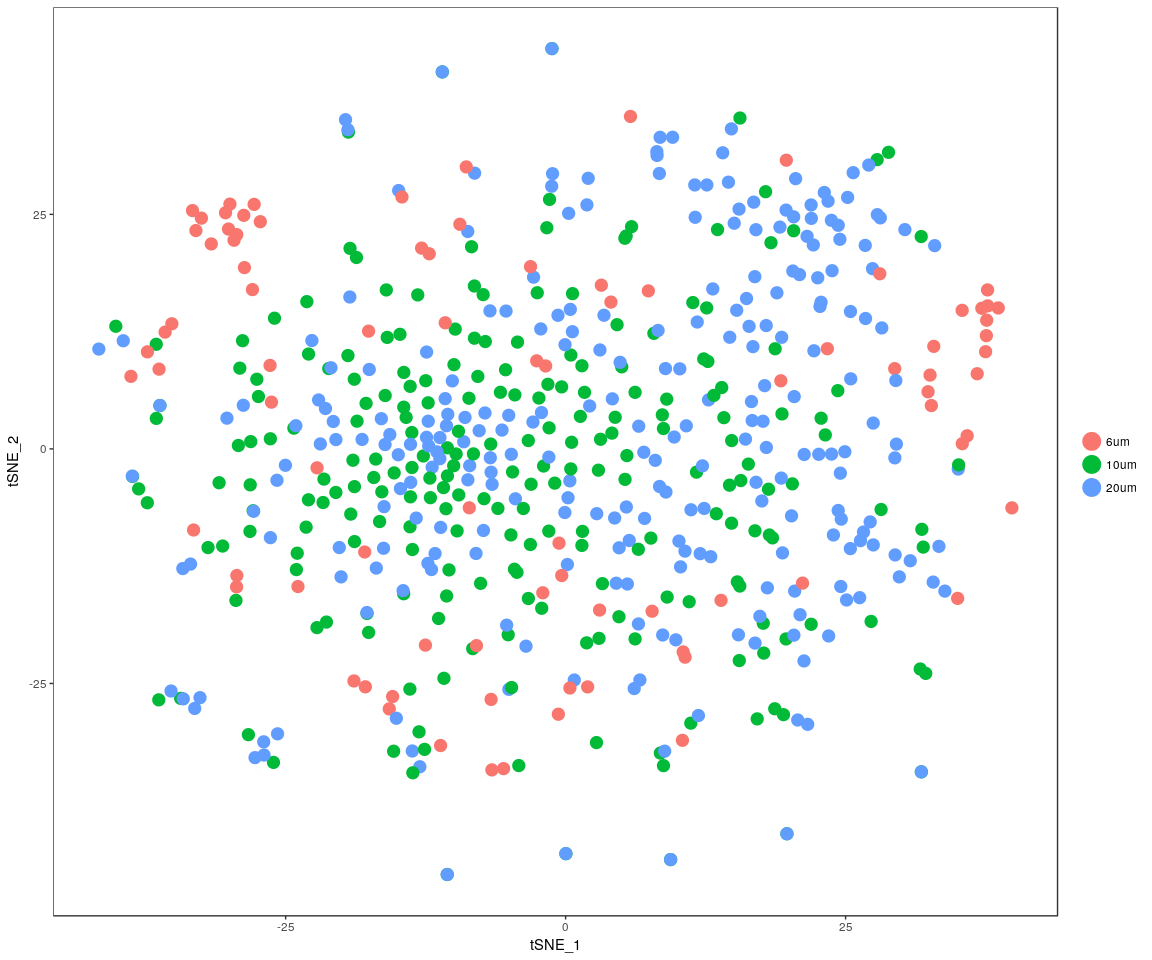
FeaturePlot(object = Negative.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(Negative.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "pca")



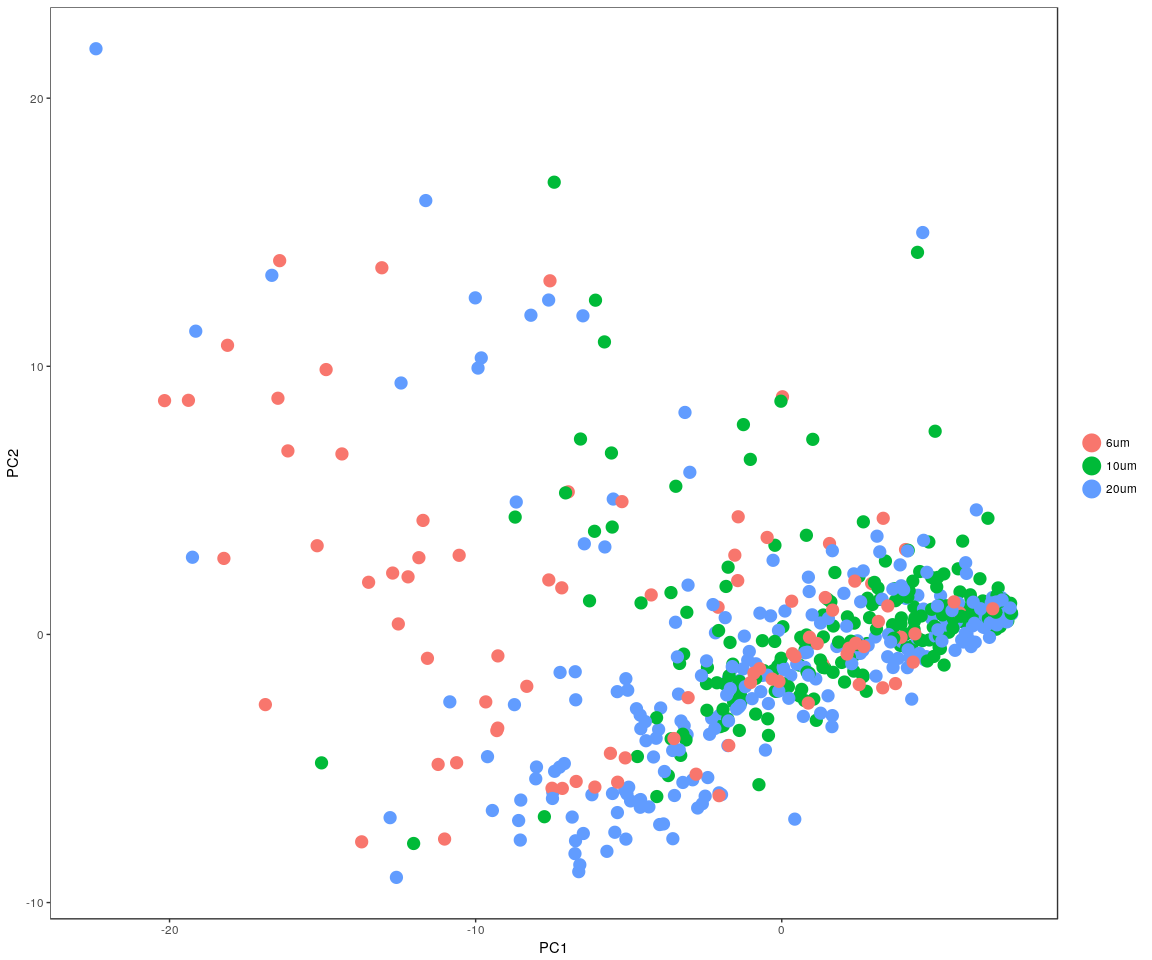
FeaturePlot(object = Negative.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(Negative.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "tsne")



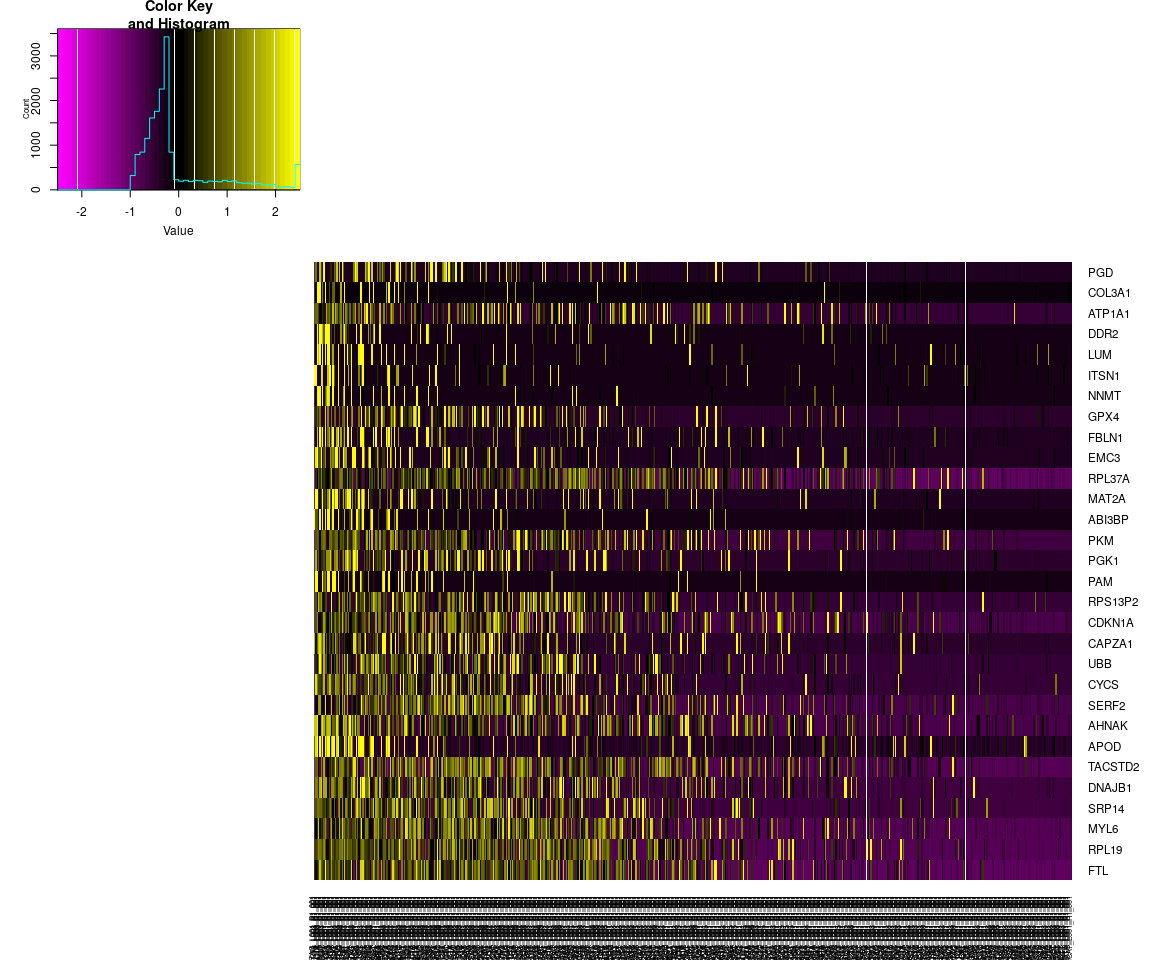
DimPlot(Negative.pbmc, reduction.use = "tsne", pt.size = 4) # grour by sample



DimPlot(Negative.pbmc, reduction.use = "pca", pt.size = 4) # grour by sample



DimHeatmap(Negative.pbmc, reduction.type = "pca", check.plot = FALSE)



The Faetureplot of ITGB4, ABCB5, KRT19, ACTB, KRT12, KRT5, GAPDH, KRT3, PAX6, WNT7A, KRT14, TP63, KRT10based on **PCA** shows that,they only has high expression level in few samples,and expresss lowly in most sample.It means that may be these important genes express differently across sample.The plot also tell us the gene **KRT5,GAPDH,PAXX6,KRT14** have more higher expression level than the other important genes.It is consistent with the result of violin plot. About the heatmap,we only show the gene **ITGB4** And the FeatureHeatmap and Heamap also comfirm this phenomeno.We try the other four variable genes,which has the similar result as gene *ITGB4* But the *tSNE* and \* PCA \* plot show that, the sample can not be split apparently.The result may be is not good based on the PCA and tSNE method.

## Differential expression

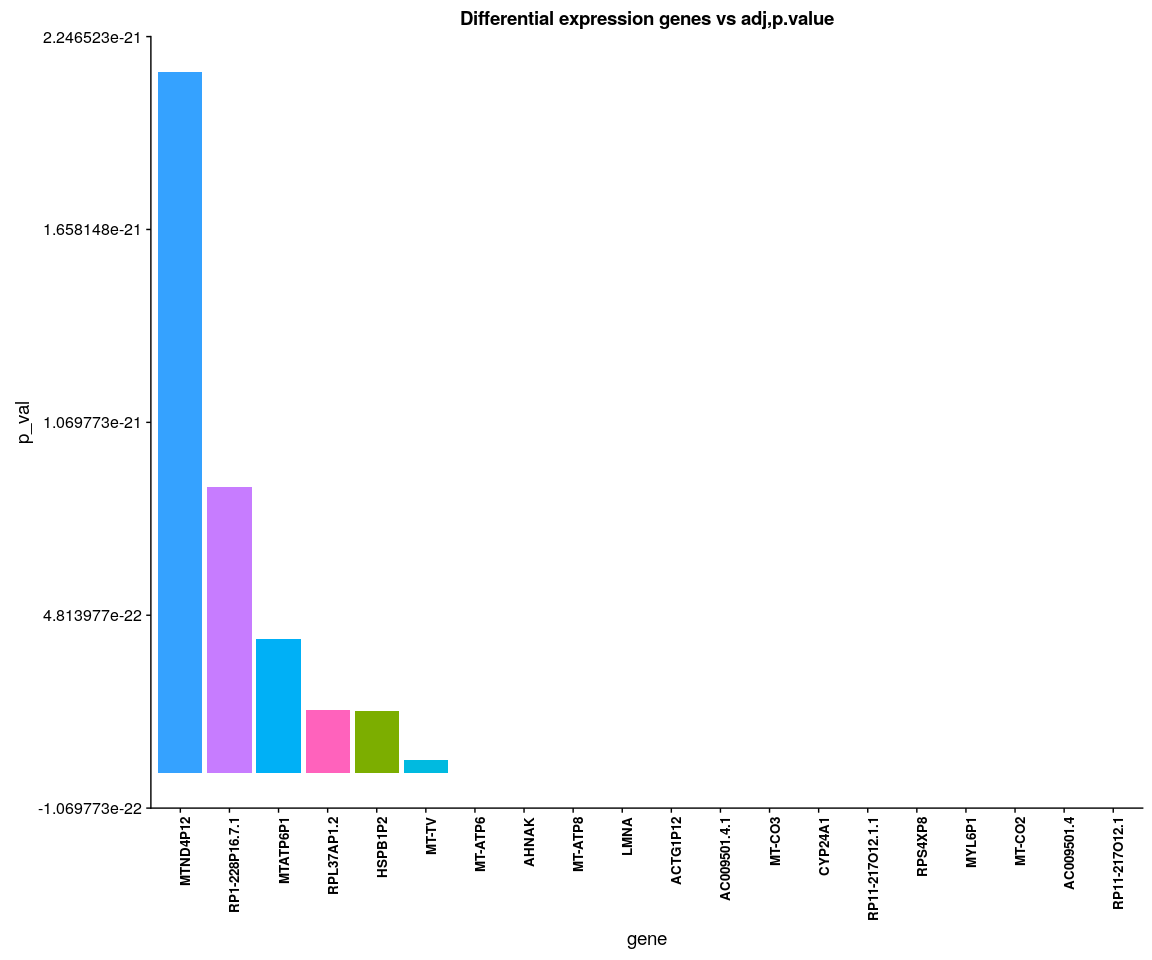
Next,we will have analysis on gene differential expression.Find maker genes across sample.We use the method: \*\*wilcox test\*\*

# Finds markers (differentially expressed genes) for each of the identity  
# classes in a dataset  
Negative.markers <- FindAllMarkers(Negative.pbmc, test.use = "bimod", print.bar = FALSE)  
head(Negative.markers)

## p\_val avg\_logFC pct.1 pct.2 p\_val\_adj cluster  
## RP11-217O12.1 1.169891e-91 3.0194034 0.989 0.949 1.636561e-87 6um  
## AC009501.4 5.001195e-55 3.3911245 0.759 0.093 6.996172e-51 6um  
## MT-CO2 7.052448e-42 -2.2905234 0.770 0.968 9.865670e-38 6um  
## MYL6P1 1.498695e-39 0.4207640 0.034 0.291 2.096525e-35 6um  
## RPS4XP8 1.829150e-38 0.2555633 0.034 0.145 2.558798e-34 6um  
## CYP24A1 9.961353e-36 2.7452522 0.586 0.053 1.393494e-31 6um  
## gene  
## RP11-217O12.1 RP11-217O12.1  
## AC009501.4 AC009501.4  
## MT-CO2 MT-CO2  
## MYL6P1 MYL6P1  
## RPS4XP8 RPS4XP8  
## CYP24A1 CYP24A1

We check whether the important genes are still in the marker genes we found from the DESeq analysis. the genes:KRT19, ACTB, KRT5, KRT3, PAX6, KRT14 are still in the marker genes.

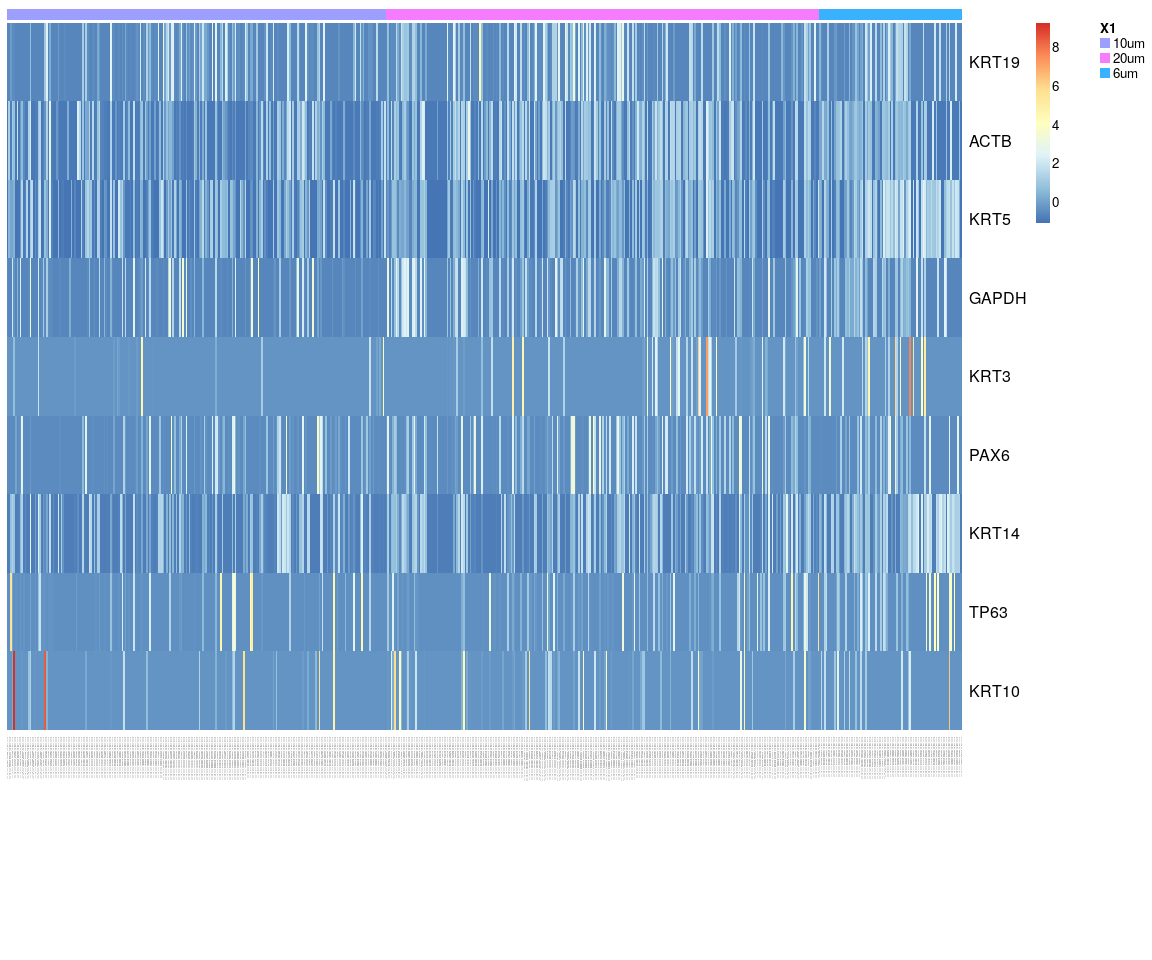
### Bar plot of gene’s p.val



Negative.heatmap <- Heatmap\_fun(genes = important.genes[important.genes %in%   
 rownames(Negative.pbmc@raw.data)], tpm.data = Negative.pbmc@scale.data,   
 condition = unique(as.character(Negative.pbmc@ident)), all.condition = as.character(Negative.pbmc@ident))

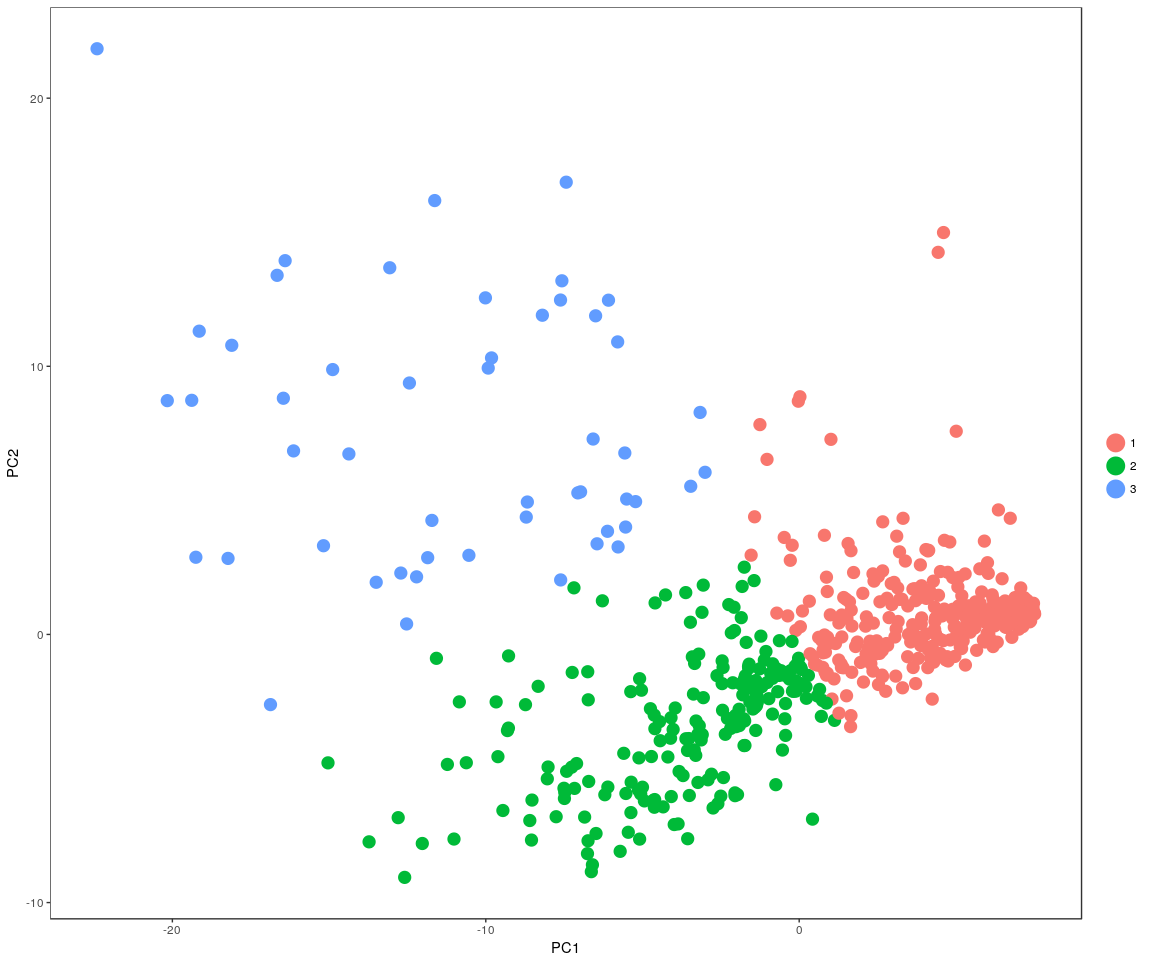
## There ara 3 conditions  
## Whether creat data accurate 0

NMF::aheatmap(Negative.heatmap[[2]], Rowv = NA, Colv = NA, annCol = Negative.heatmap[[1]],   
 scale = "none")

 We have find all marker genes across sample,there are 824 significant genes(adjust p-value <0.05) in all marker genes.

### Next,Spectral k-means clustering on single cells based on PCA

Negative.pbmc <- KClustDimension(Negative.pbmc, reduction.use = "pca", k.use = 3)  
clusters.pca <- Negative.pbmc@meta.data$kdimension.ident  
DimPlot(Negative.pbmc, pt.size = 4, group.by = "kdimension.ident")



### Spectral k-means clustering on single cells based on tSNE

Negative.pbmc <- KClustDimension(Negative.pbmc, reduction.use = "tsne", k.use = 3)  
clusters.tsne <- Negative.pbmc@meta.data$kdimension.ident  
DimPlot(Negative.pbmc, pt.size = 4, group.by = "kdimension.ident", reduction.use = "tsne")

