Mouse analysis v2

yejg

2018/1/26

### 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(DESeq2)

## Step 1: All data: Analysis based on sample group

### Read data

### Data QA

mouse.only.pro <- Load\_data(data\_dir = "../data/mouse.txt")  
rownames(mouse.only.pro) <- unlist(lapply(rownames(mouse.only.pro), str\_to\_upper))  
important.genes <- c("ITGB4", "ABCB5", "KRT19", "ACTB", "KRT12", "KRT5", "GAPDH",   
 "KRT3", "PAX6", "WNT7A", "KRT14", "TRP63", "KRT10")  
  
table(unlist(lapply(colnames(mouse.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][2]))))

##   
## 10um 20um 6um   
## 195 543 184

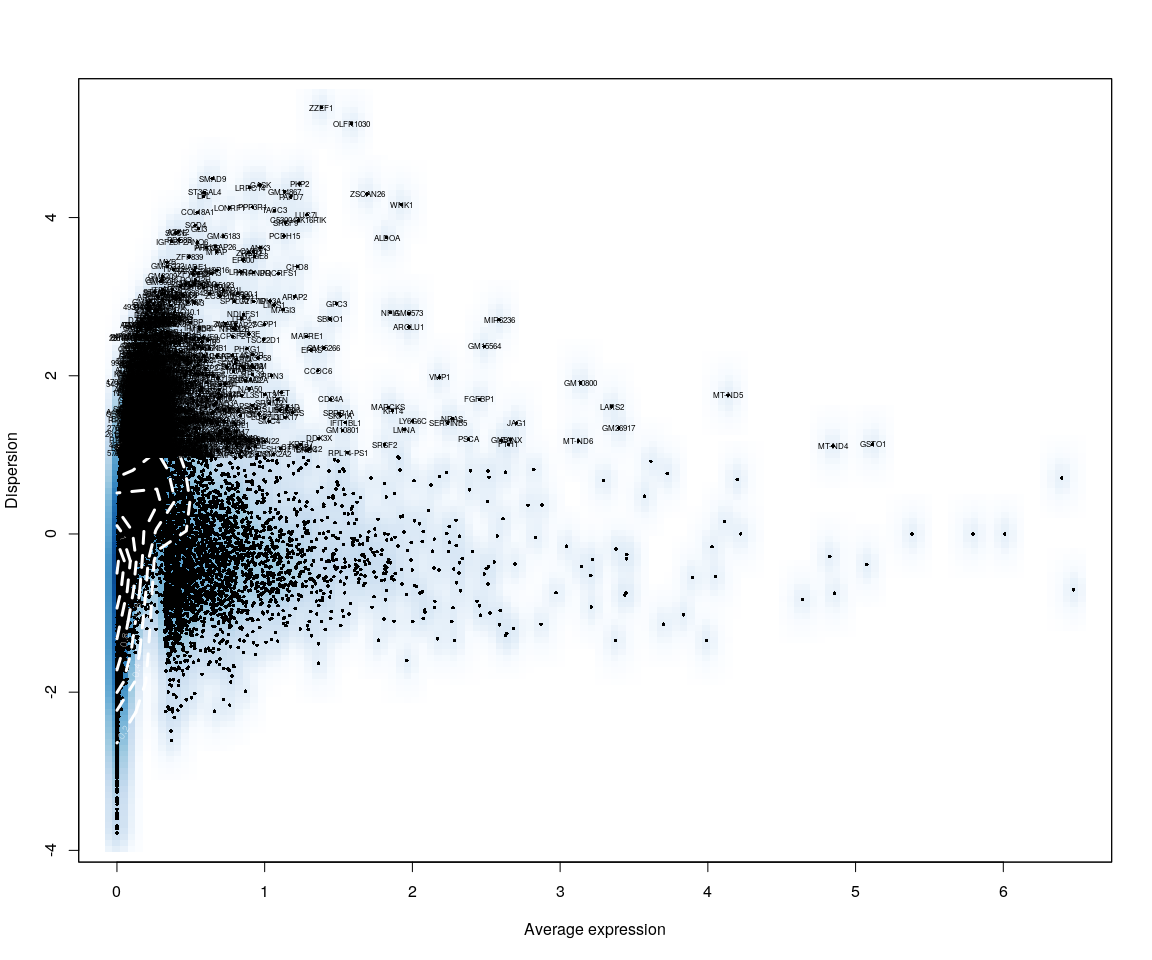
table(unlist(lapply(colnames(mouse.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][1]))))

##   
## mc002 mc004 mc005 mc006 mc007 mc008 mc009 mc010 mc011 mc012 mc013 mc014   
## 91 54 64 67 70 53 61 92 92 94 31 31   
## mc015 mc016 mc017 mc018   
## 31 31 30 30

### Create Seurat object and not caculate DESeq,but set **min.cells=10** and **min.genes=2**

# only select the cells contain 10 genes expressed at least,select the genes  
# must be expressed in two cells at least  
mouse.all.pbmc <- DESeq\_SeuratObj(X = mouse.only.pro, DESq = FALSE, min.cells = 10,   
 min.genes = 2)

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



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

## Figure Explore

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

mouse.imp.lognorm <- data.frame(FetchData(mouse.all.pbmc, vars.all = important.genes[important.genes %in%   
 rownames(mouse.all.pbmc@raw.data)]))  
mouse.imp.lognorm$cell.size <- unlist(lapply(rownames(mouse.imp.lognorm), function(x) return(str\_split(x,   
 "\_")[[1]][2])))  
mouse.imp.lognorm.melt <- melt(mouse.imp.lognorm)

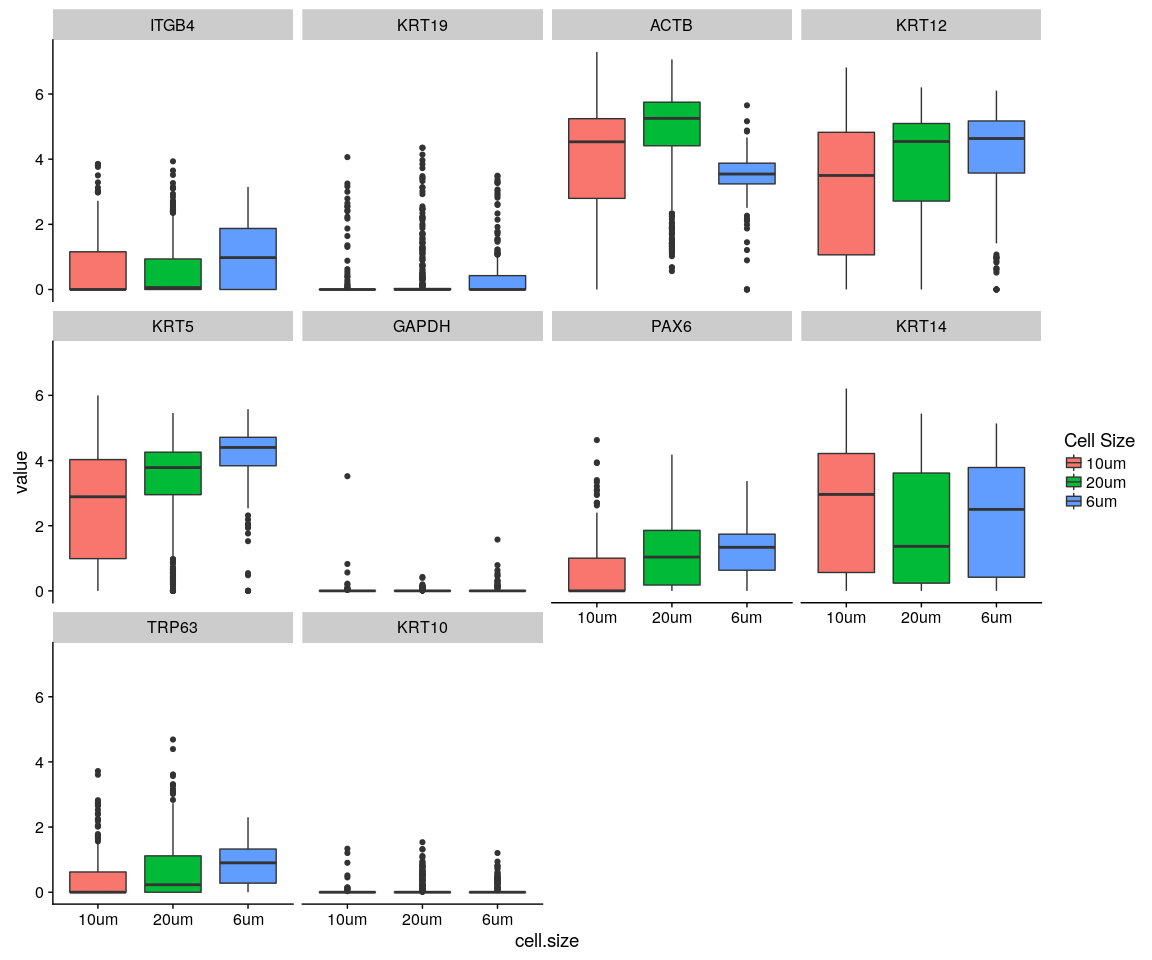
### Figure Explore.1

#### Violin

p <- ggplot(data = mouse.imp.lognorm.melt, aes(y = value, x = cell.size, fill = cell.size))  
p + geom\_violin(trim = FALSE, scale = "width") + facet\_wrap(~variable) + geom\_jitter() +   
 guides(fill = guide\_legend(title = "Cell Size"))

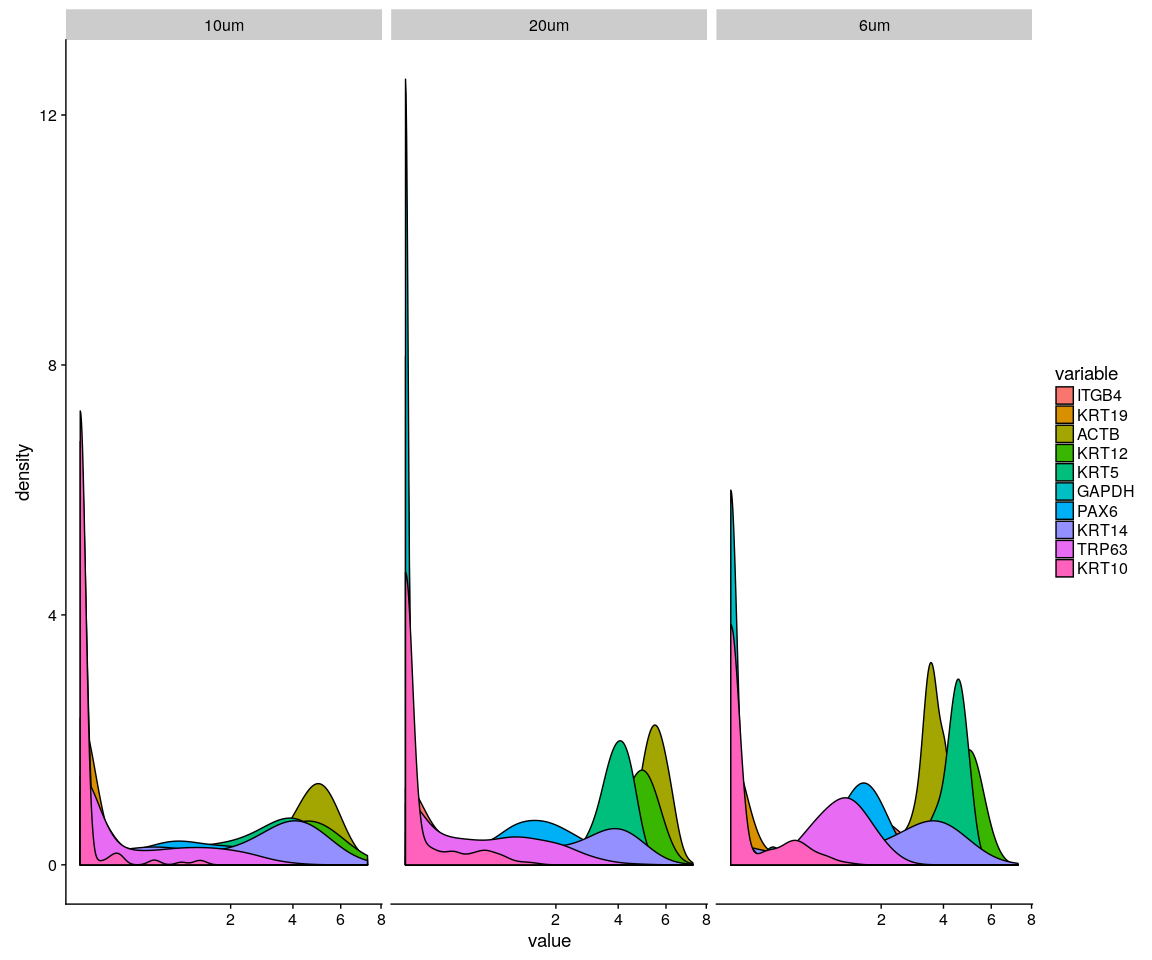
#### Boxplot

p <- ggplot(data = mouse.imp.lognorm.melt, aes(y = value, x = cell.size, fill = cell.size))  
p + geom\_boxplot() + guides(fill = guide\_legend(title = "Cell Size")) + facet\_wrap(~variable)

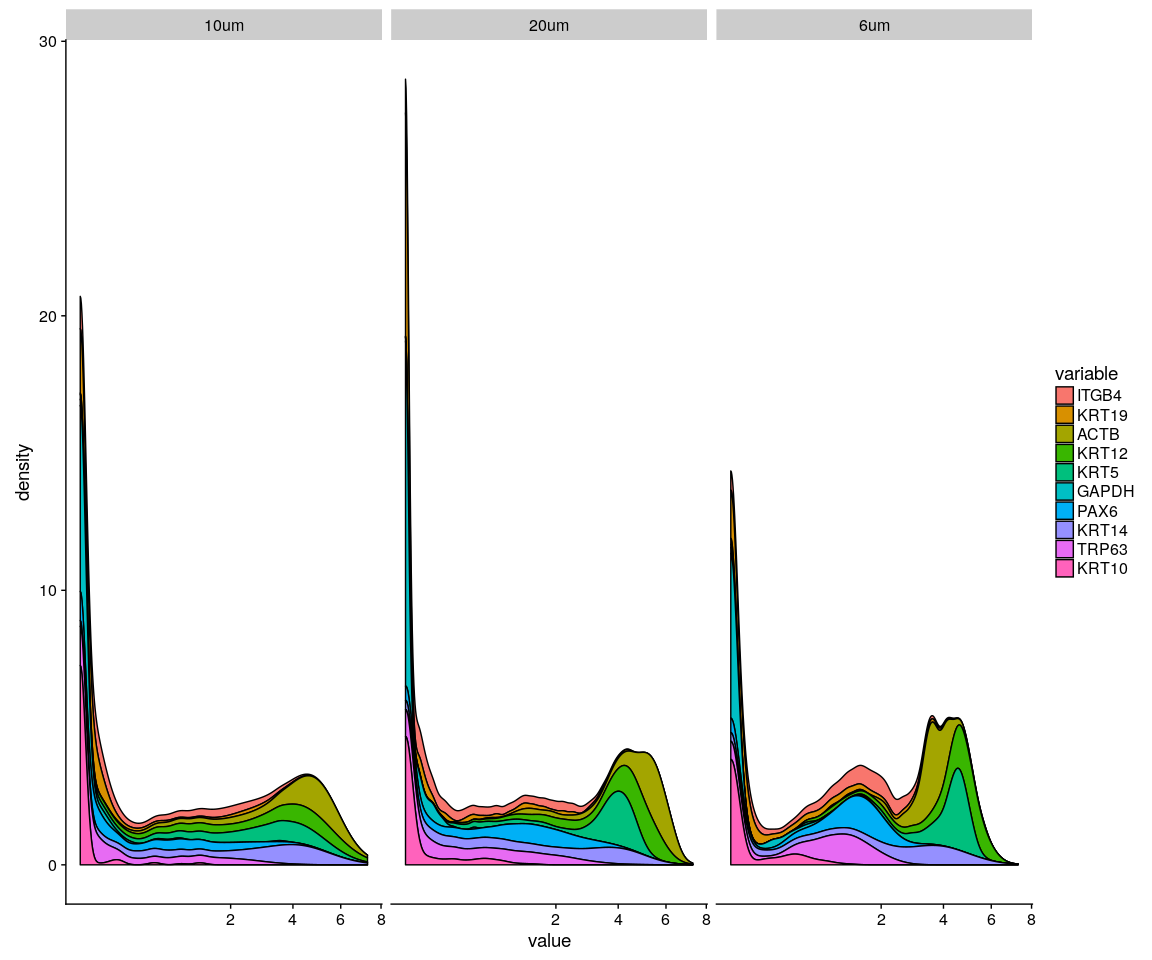


#### Density,histogram

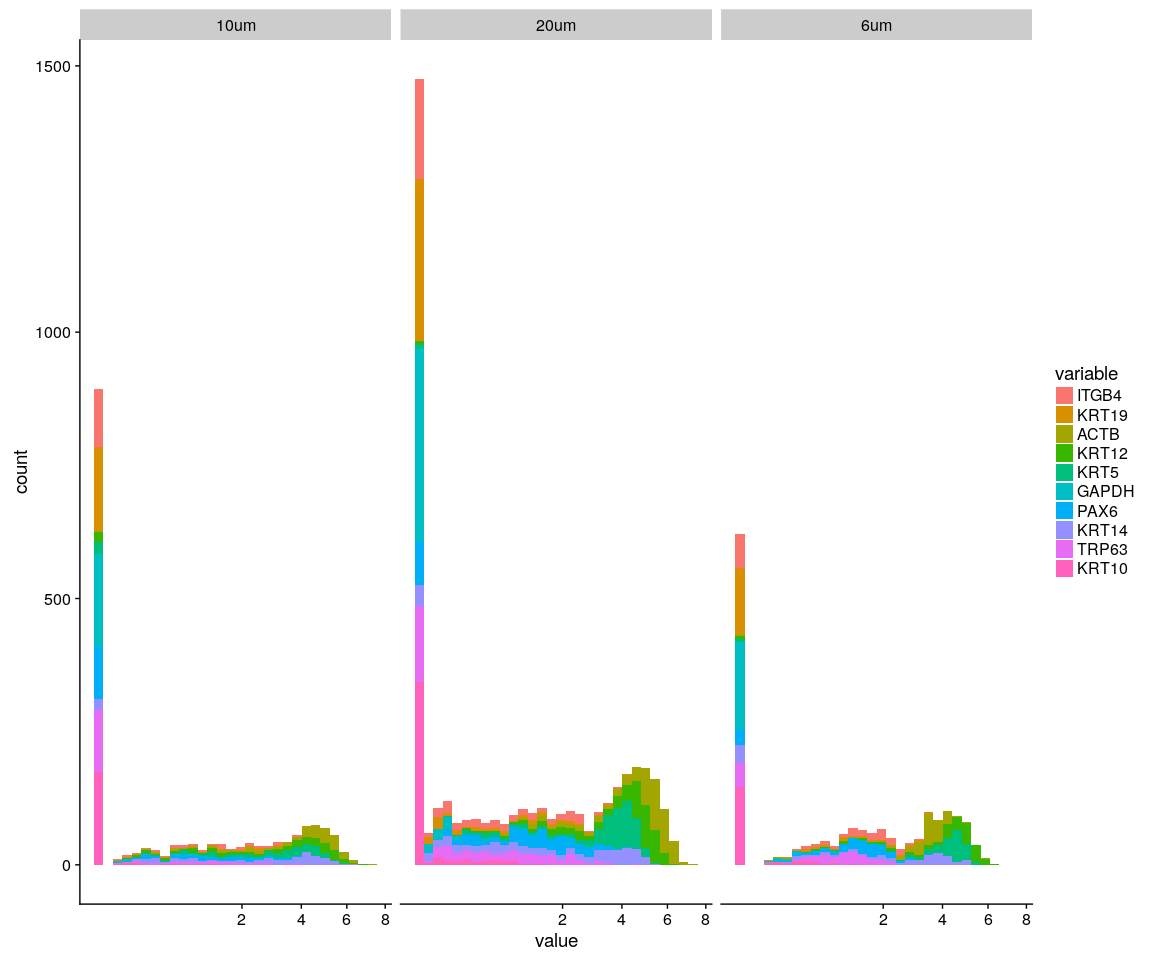
ggplot(data = mouse.imp.lognorm.melt, aes(x = value, fill = variable)) + geom\_density(kernel = "gaussian") +   
 scale\_x\_sqrt() + facet\_wrap(~cell.size)



ggplot(data = mouse.imp.lognorm.melt, aes(x = value, fill = variable)) + geom\_density(kernel = "gaussian",   
 position = "stack") + scale\_x\_sqrt() + facet\_wrap(~cell.size)

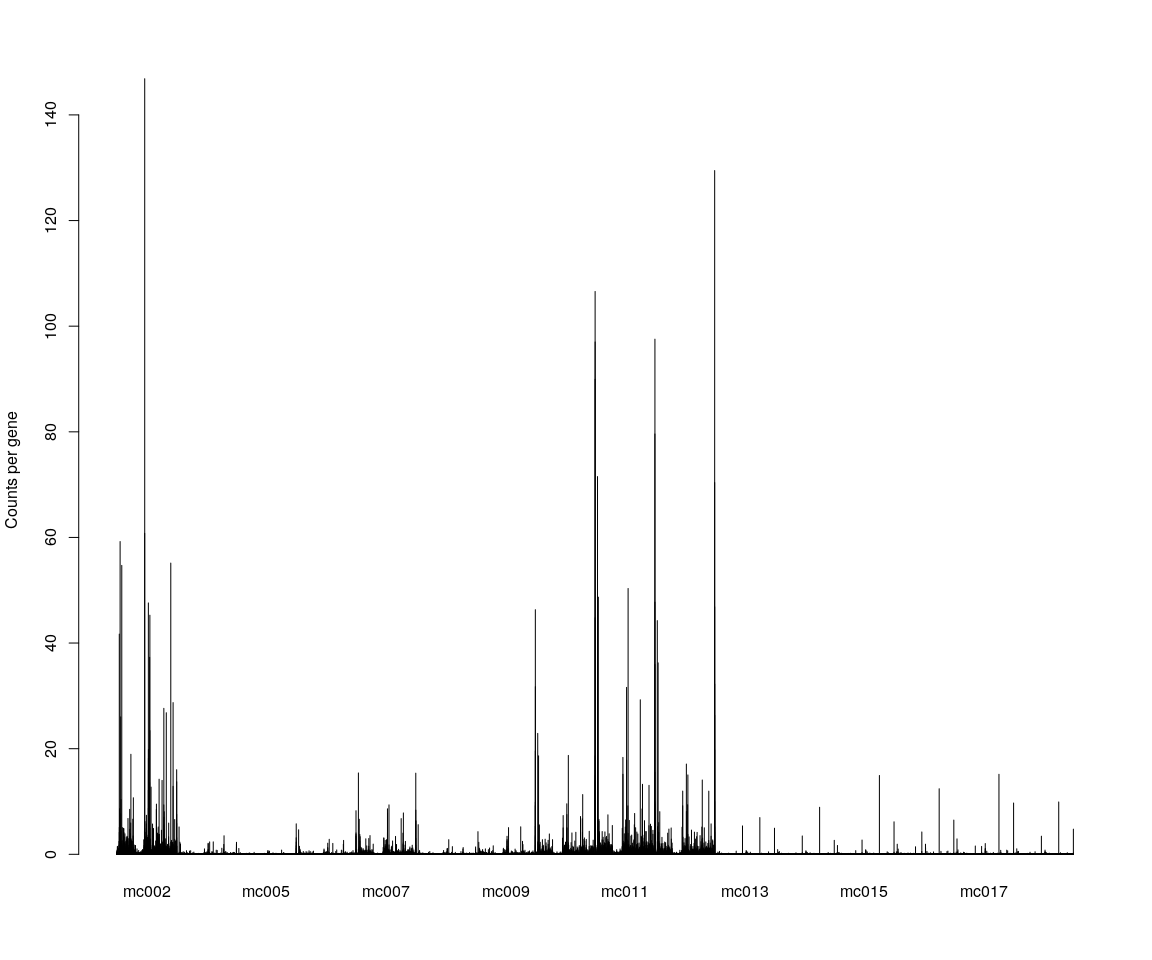


ggplot(data = mouse.imp.lognorm.melt, aes(x = value, fill = variable)) + geom\_histogram() +   
 scale\_x\_sqrt() + facet\_wrap(~cell.size)

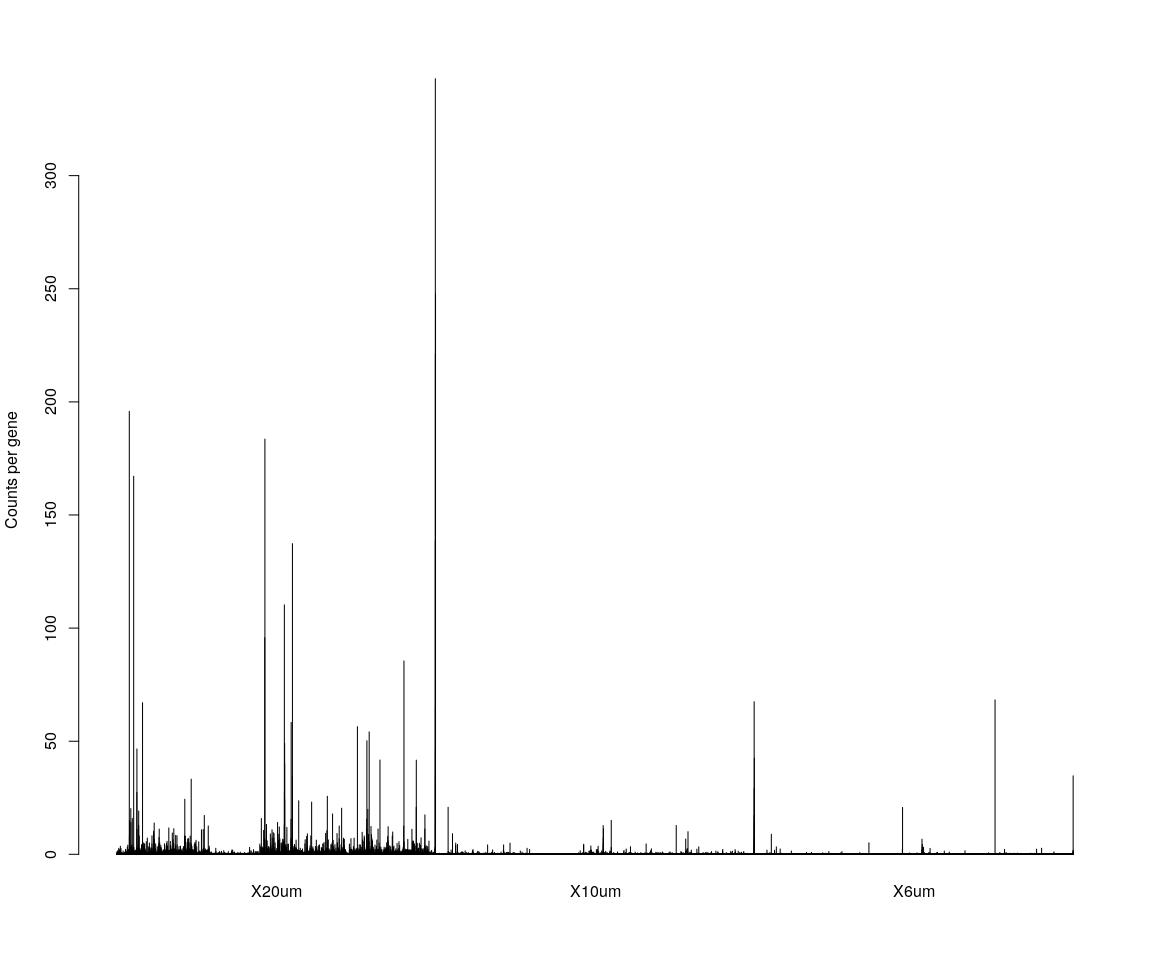


### Figure explore.2

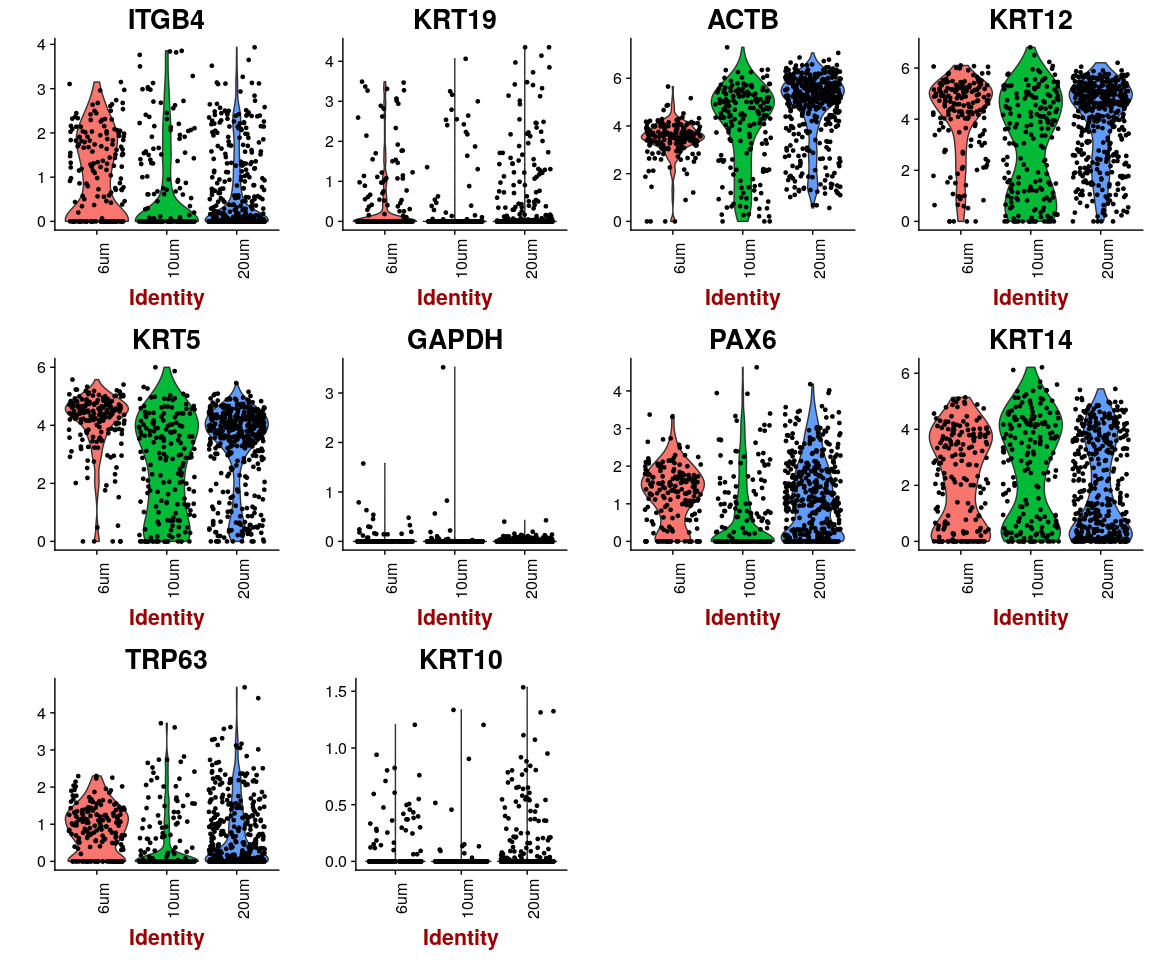
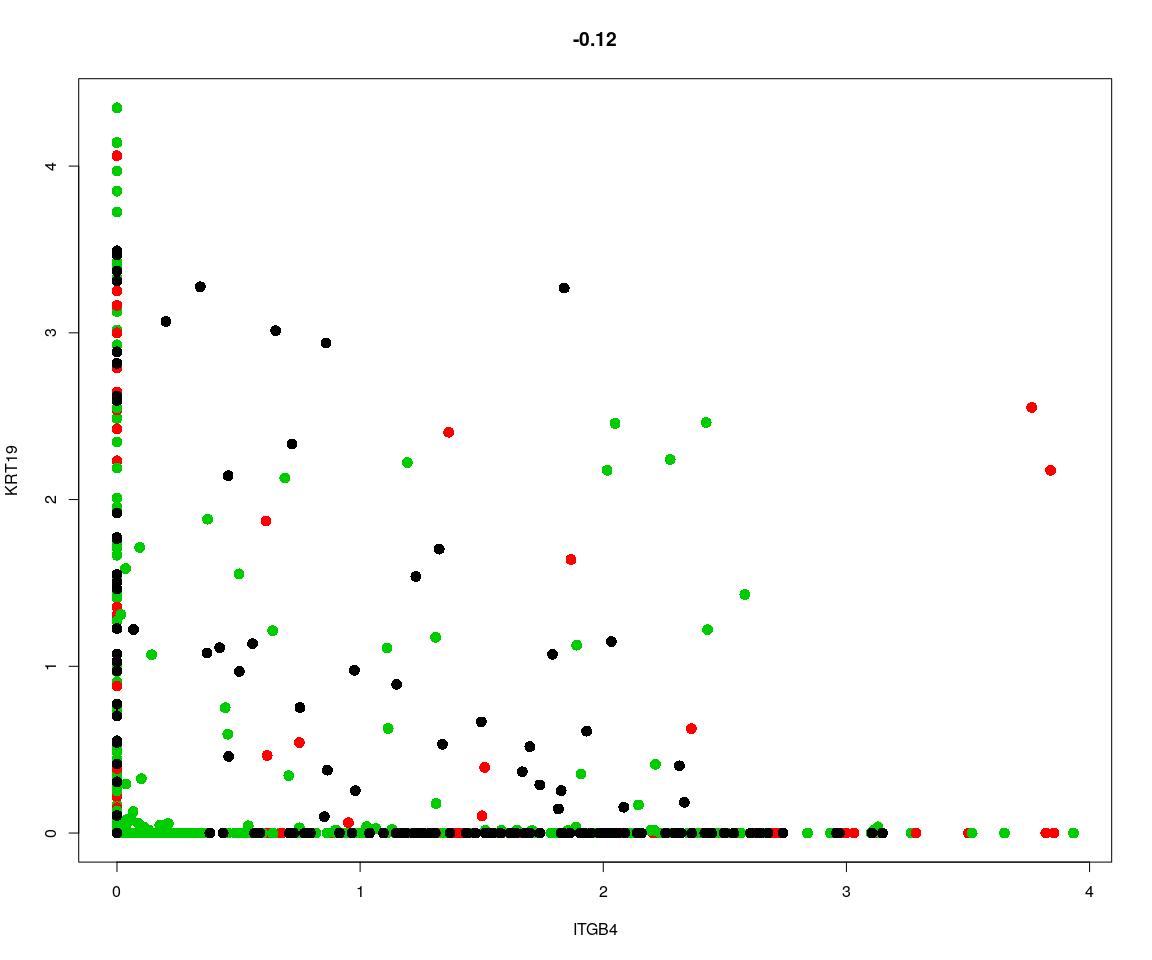
Group\_Bar(mouse.all.pbmc@raw.data, group = all.sample.group)



Group\_Bar(mouse.all.pbmc@raw.data, group = all.sample.size)



# We are interested in the gene ITGB4  
GenePlot(mouse.all.pbmc, gene1 = "ITGB4", gene2 = important.genes[3])  
# VlnPlot(mouse.all.pbmc,features.plot = 'ITGB4',y.lab.rot = 90) # Violinn  
# plot of gene ITGB in all sample  
VlnPlot(mouse.all.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(mouse.all.pbmc@raw.data)], y.lab.rot = 90) # Violinn plot of gene ITGB in all sample



After create seurat objectITGB4, KRT19, ACTB, KRT12, KRT5, GAPDH, PAX6, KRT14, TRP63, KRT10 are still in the sample data.Seurat object will remove low expression genes because set min.cells and min.genes.According to the plot above,they tell us that genes have higher expression in **mc002,mc007,mc001** if group by sample name identity;have higher expression in **20um** if group by cell size.And the gene **GAPDH,KRT10** have lower expression across sample,compared to other genes detected.

## Dimensionality reduction

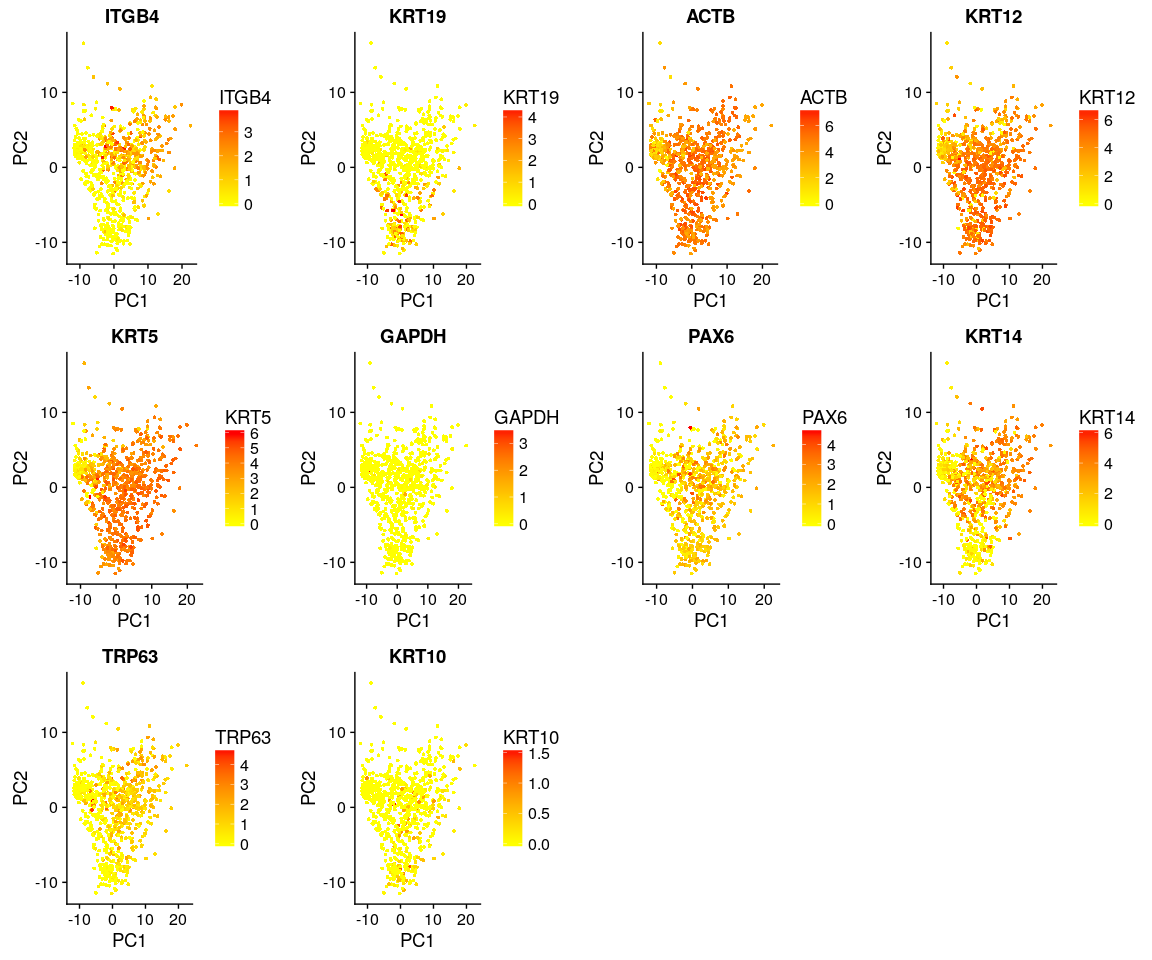
### **PCA** and **tSNE**

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

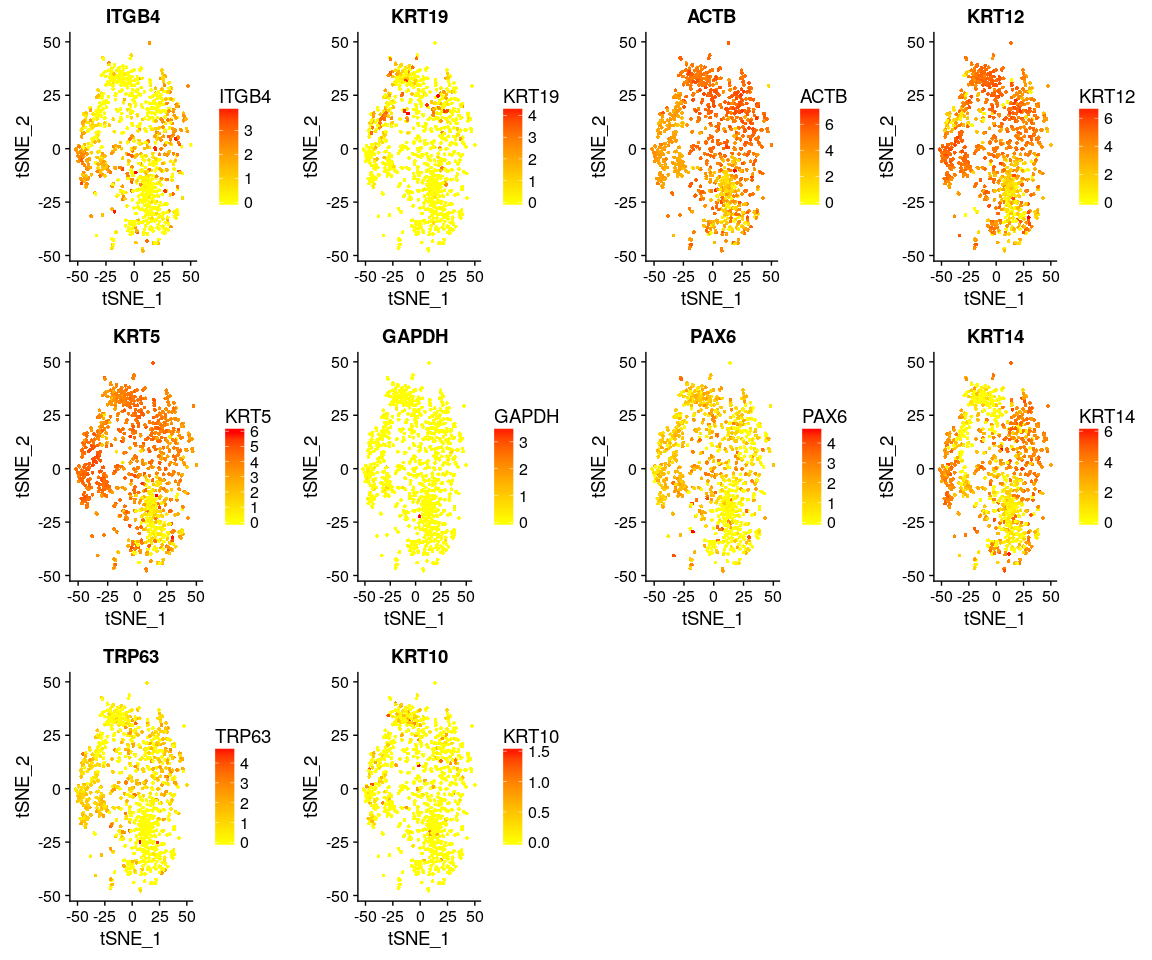
all.pbmc <- PCA.TSNE(object = mouse.all.pbmc, pcs.compute = FALSE, num.pcs = 28)

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

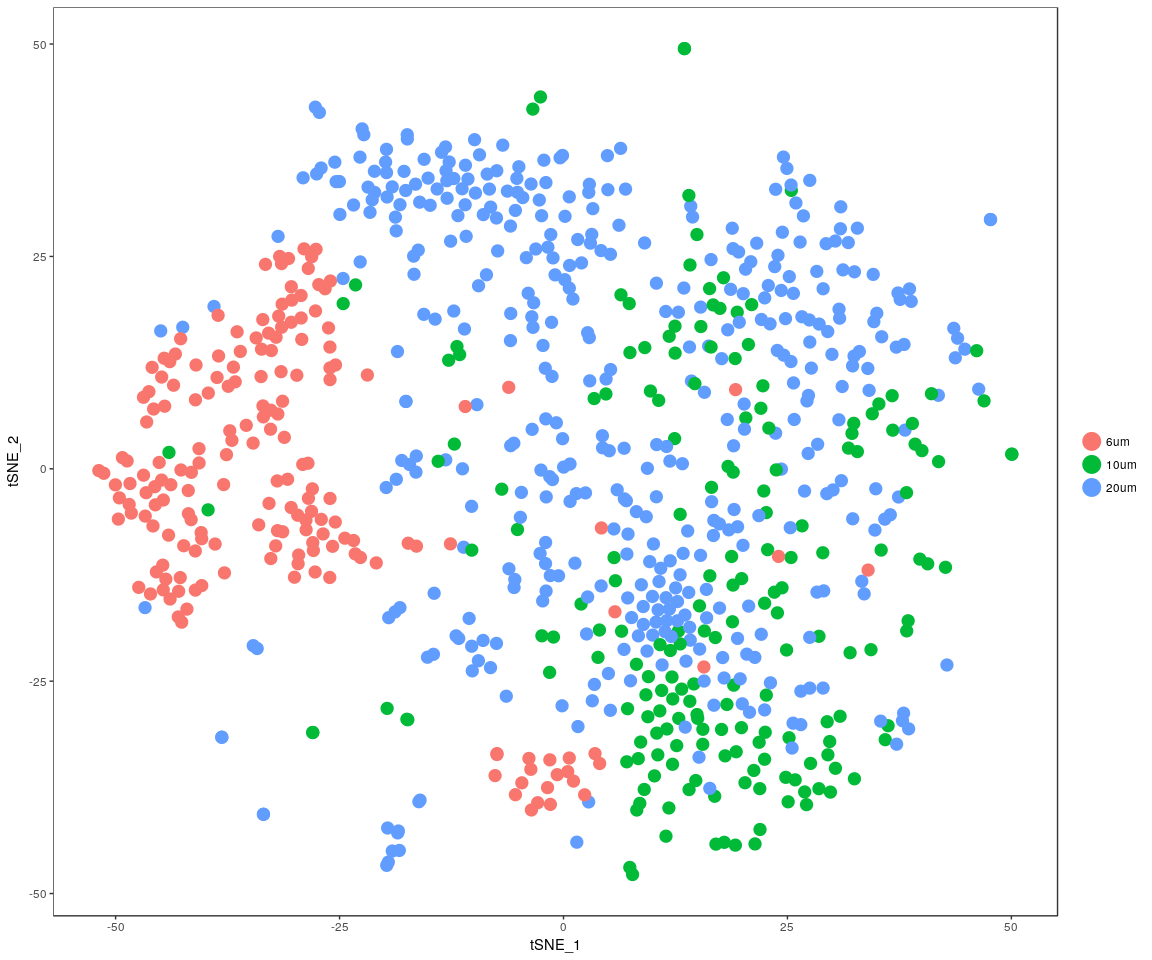
# FeaturePlot(object = all.pbmc,features.plot ='ITGB4',pt.size = 4,no.legend  
# = FALSE) # ITGB4 gene in part dataset  
FeaturePlot(object = all.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(all.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "pca")



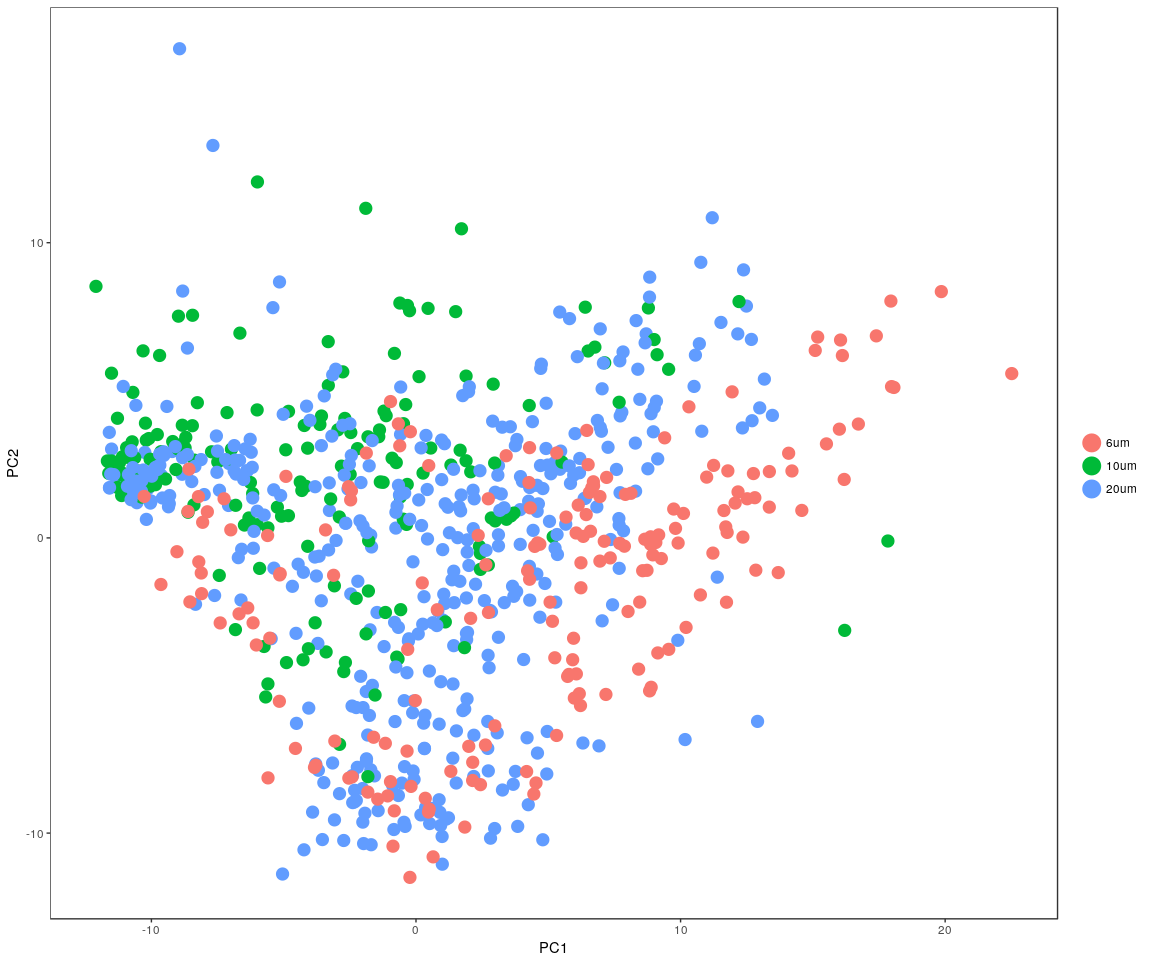
FeaturePlot(object = all.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(all.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "tsne")



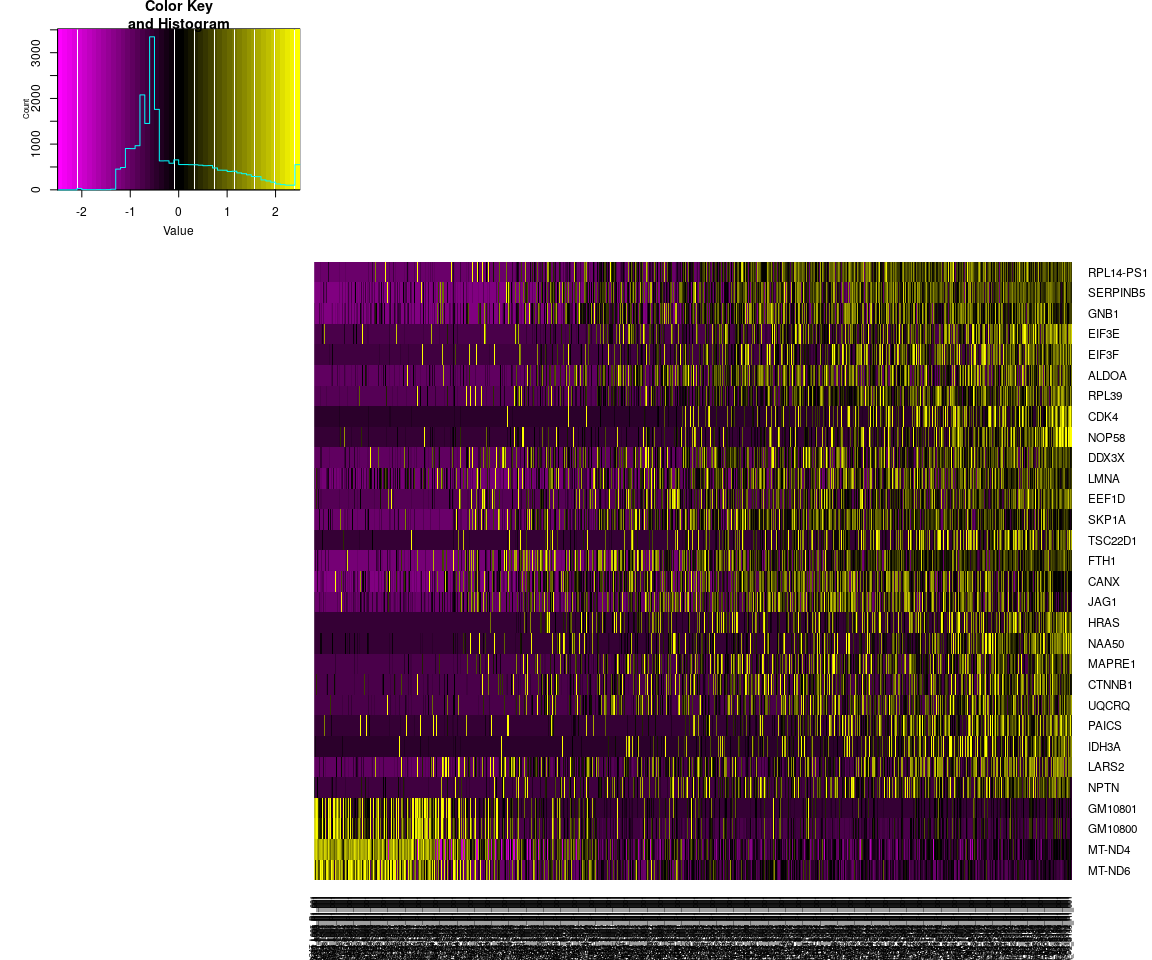
DimPlot(all.pbmc, reduction.use = "tsne", pt.size = 4) # grour by sample



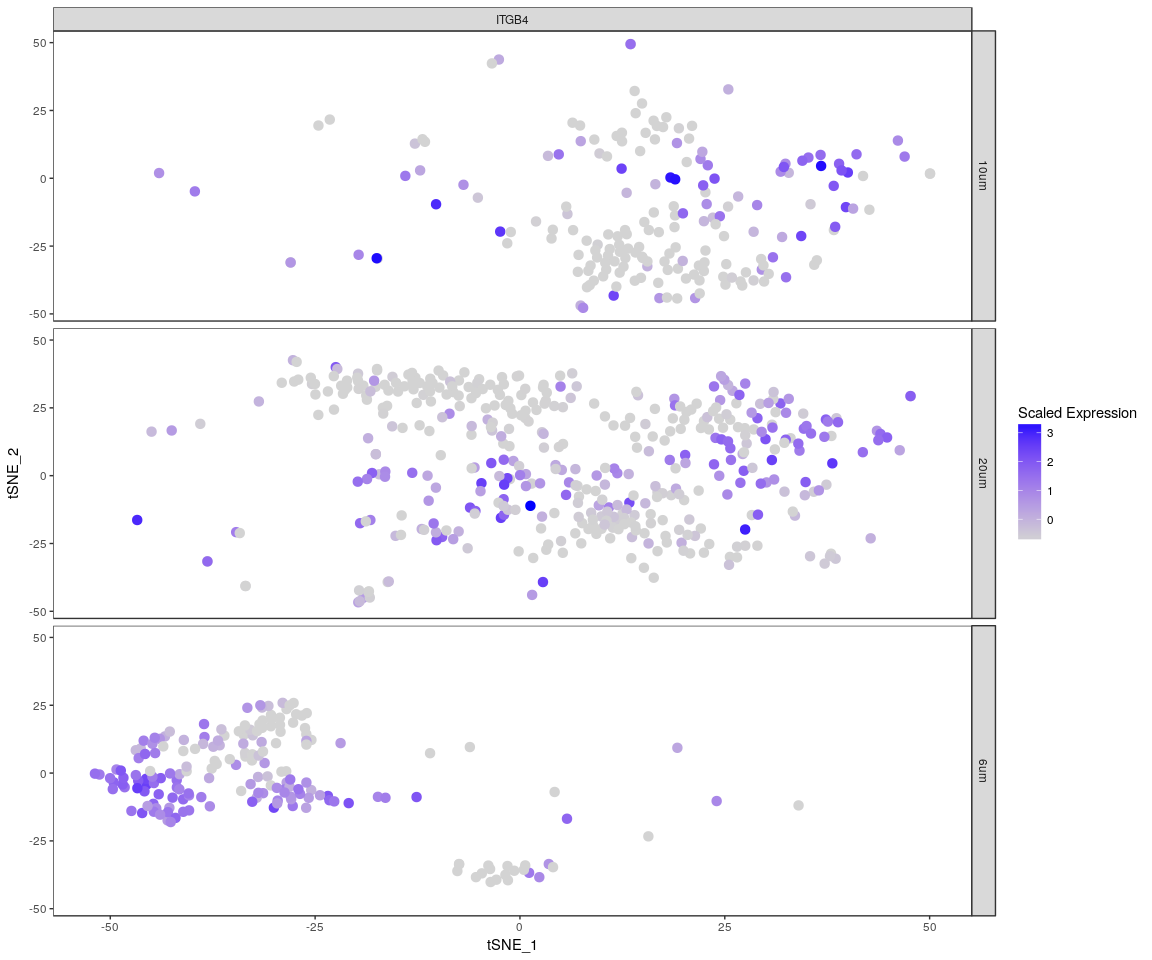
DimPlot(all.pbmc, reduction.use = "pca", pt.size = 4) # grour by sample



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



FeatureHeatmap(all.pbmc, features.plot = "ITGB4", pt.size = 3, plot.horiz = TRUE,   
 cols.use = c("lightgrey", "blue"))

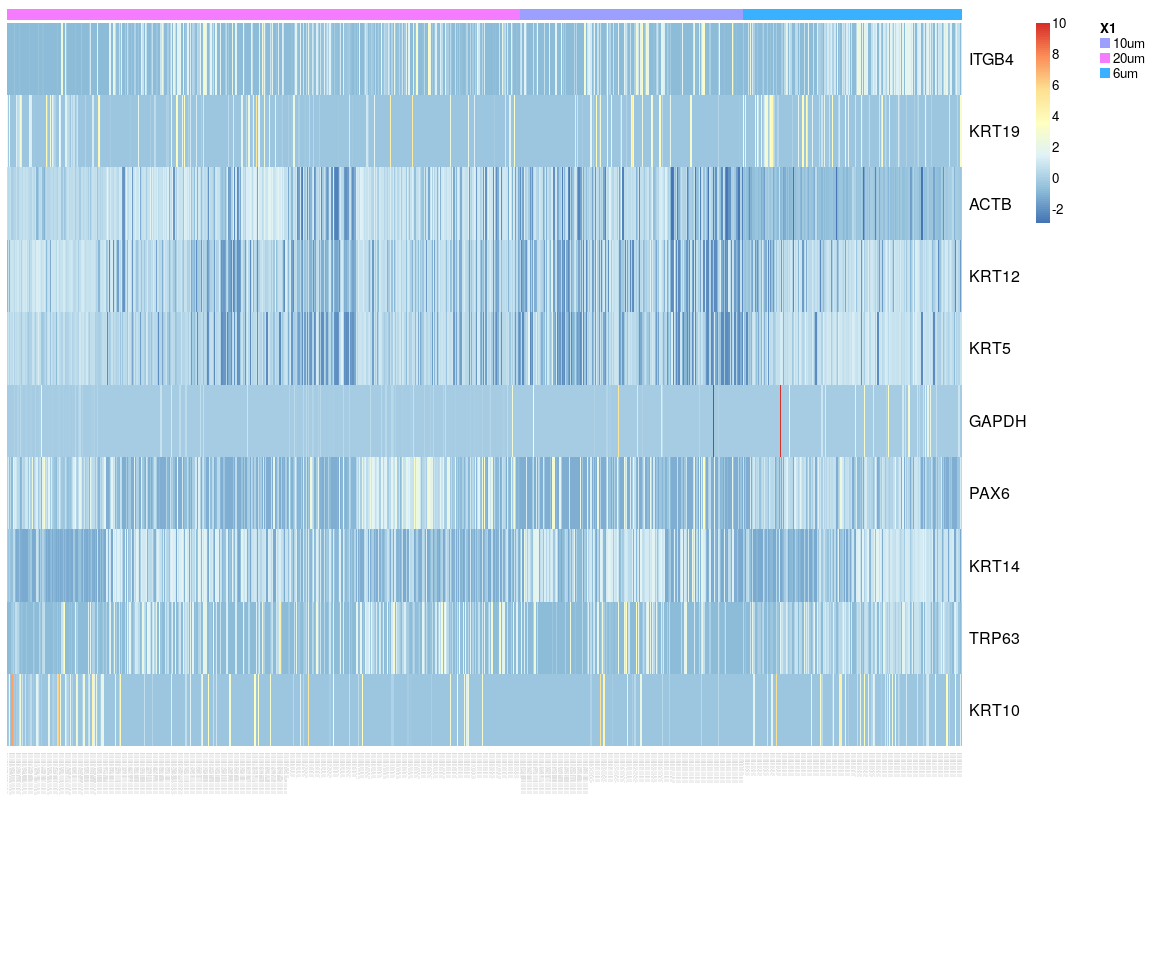


The Faetureplot of ITGB4, KRT19, ACTB, KRT12, KRT5, GAPDH, PAX6, KRT14, TRP63, KRT10based on **PCA,TSNE** shows that,they only has high expression level in few samples,and expresss lowly in most sample.It means that may be these important genes express differently across sample.The plot also tell us the gene **ACTB,KRT12,KRT5,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.

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

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

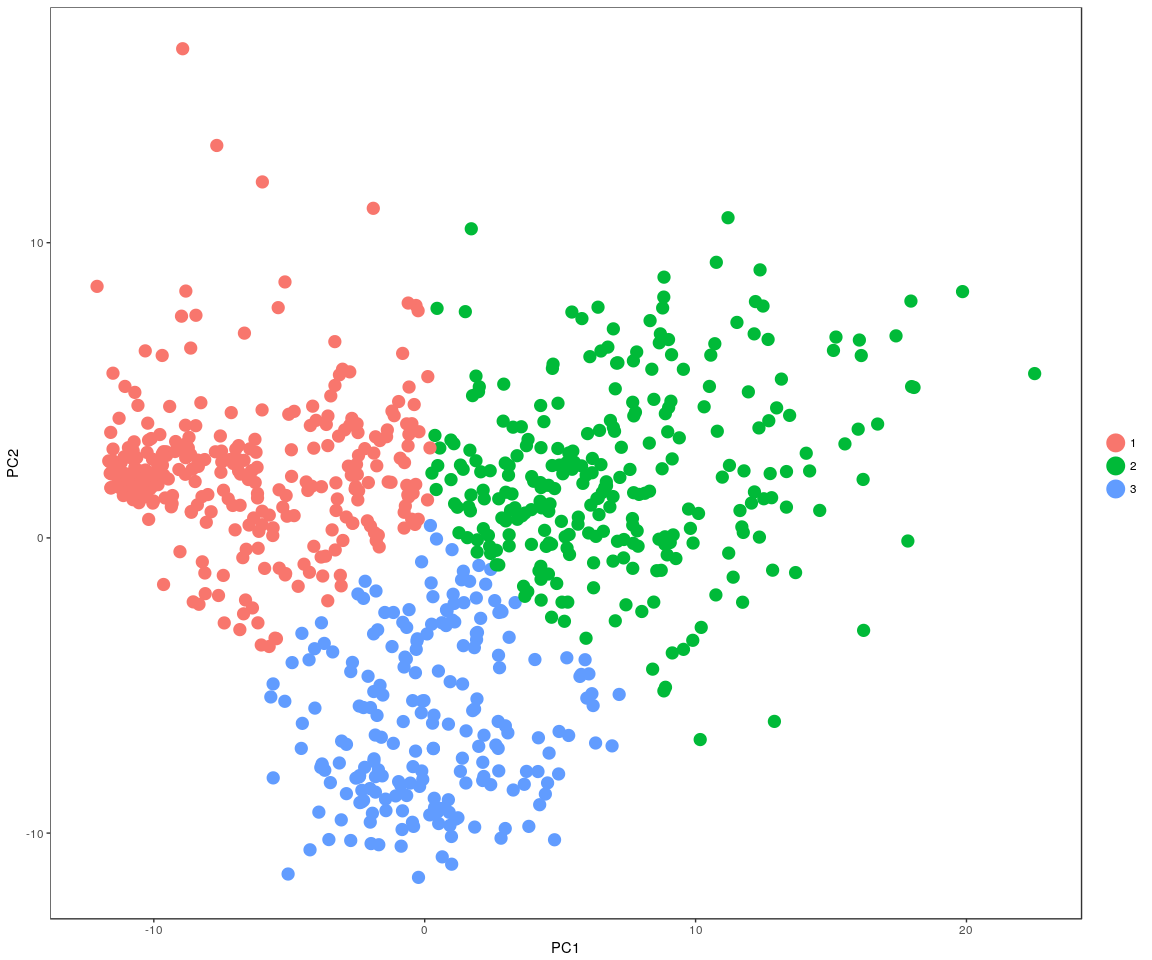
NMF::aheatmap(mouse.heatmap[[2]], Rowv = NA, Colv = NA, annCol = mouse.heatmap[[1]],   
 scale = "none")



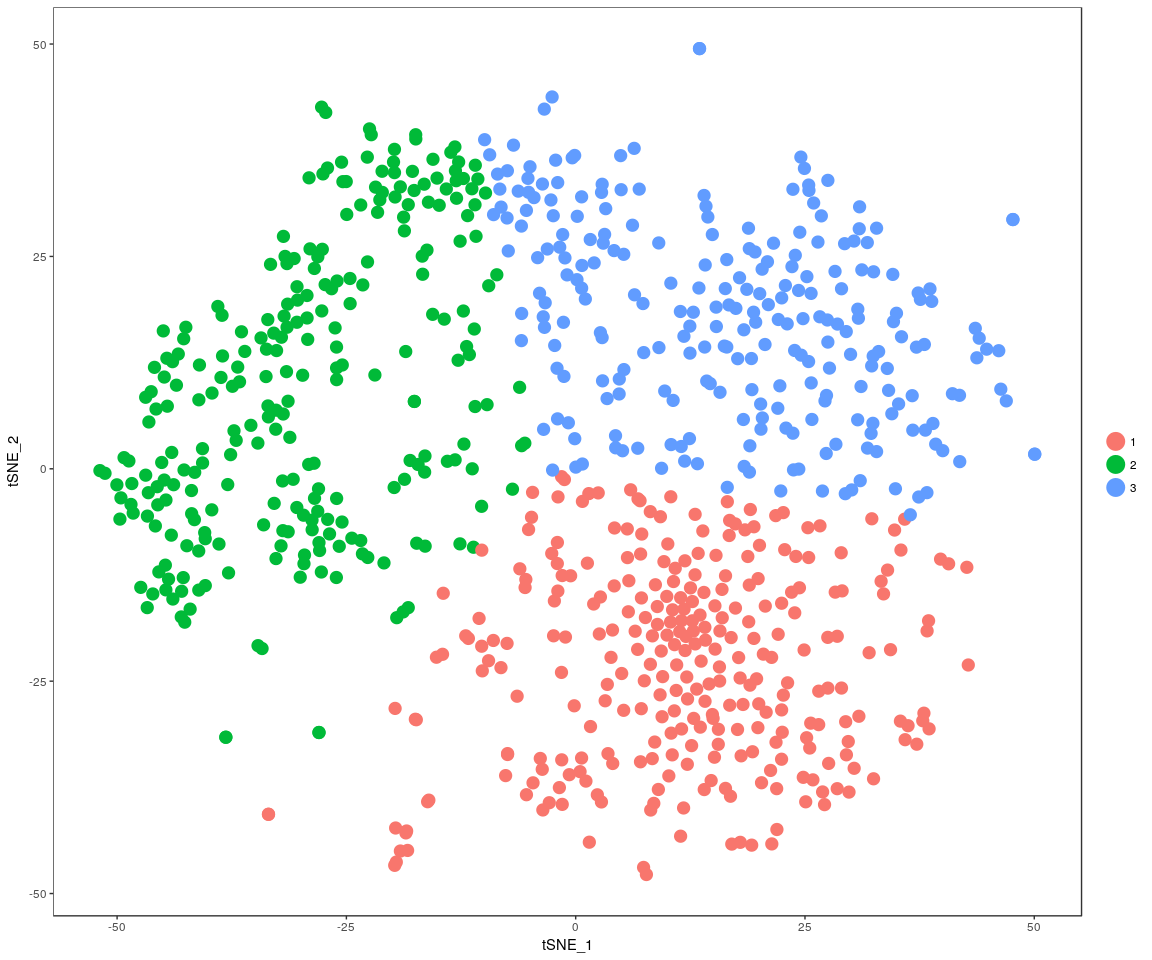
The heatmap of genes ITGB4, KRT19, ACTB, KRT12, KRT5, GAPDH, PAX6, KRT14, TRP63, KRT10 .It tells us that **KRT10,GAPDH,KRT15** expressed differently across sample,expressed more significant.

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

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



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



## Differential expression

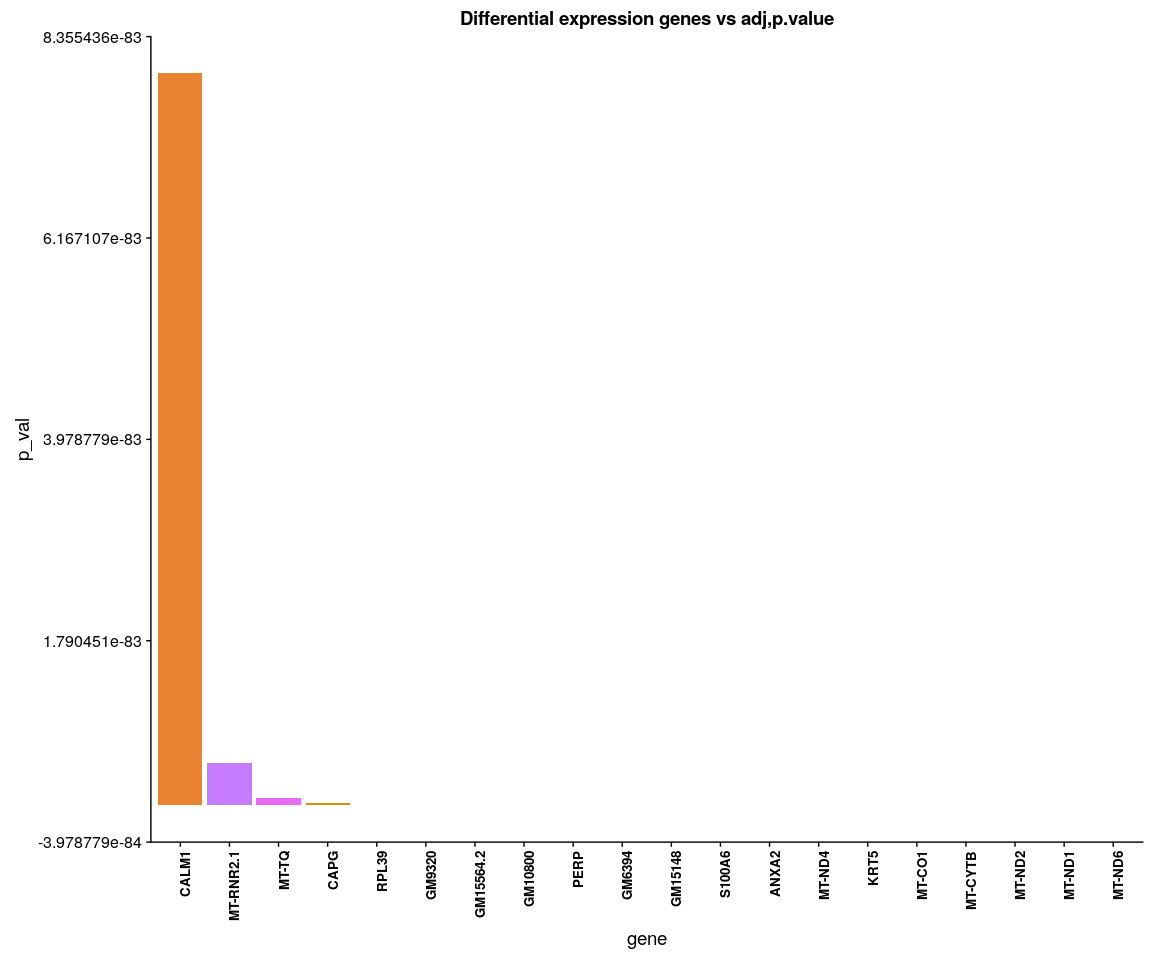
Next,we will have analysis on gene differential expression.Find maker genes across sample.We use the method: \*\*wilcox test\*\*

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

## p\_val avg\_logFC pct.1 pct.2 p\_val\_adj cluster gene  
## MT-ND6 3.462689e-140 2.6677106 0.852 0.883 5.085997e-136 1 MT-ND6  
## MT-ND1 4.551389e-134 2.8392756 0.948 0.996 6.685080e-130 1 MT-ND1  
## MT-ND2 2.696590e-133 2.6605813 0.876 0.977 3.960751e-129 1 MT-ND2  
## MT-CYTB 1.240285e-131 2.9792880 0.966 1.000 1.821731e-127 1 MT-CYTB  
## MT-CO1 3.680520e-125 2.8105358 0.952 0.994 5.405948e-121 1 MT-CO1  
## KRT5 1.906085e-118 -0.8923755 0.869 1.000 2.799658e-114 1 KRT5

We check whether the important genes are still in the marker genes we found from the DESeq analysis. the genes:ITGB4, KRT19, ACTB, KRT12, KRT5, KRT14, TRP63 are still in the marker genes.

### Bar plot of gene’s p.val



Differential expression.

When use the DESeq,it must require the gene count matrix satisify that: **every gene contains at least one zero, cannot compute log geometric means**. So have to take another method to handle data,but I do not know whether it is reasonable.Just try!!!

condition.1 <- unlist(lapply(all.pbmc@cell.names, function(x) return(str\_split(x,   
 "\_")[[1]][2])))  
mouse.xdds <- DESeq\_CT(count.data = all.pbmc@raw.data, condition.1 = condition.1)  
plotDispEsts(mouse.xdds, main = "Per-gene Dispersion")

## Do the DESeq test across all cells with sample group.And get all the significant genes between two groups(p.value < 0.05)

mouse.genes <- DESeq\_result(mouse.xdds, condition = condition.1)

mouse.genes <- as.vector(mouse.genes)  
library(VennDiagram)  
grid.draw(venn.diagram(mouse.genes[1:3], filename = NULL, fill = c("dodgerblue",   
 "goldenrod1", "darkorange1")))