Monkey deep analysis:distribution test

yejg

2018/2/1

### 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

monkey.only.pro <- Load\_data(data\_dir = "../data/monkey.txt")  
rownames(monkey.only.pro) <- unlist(lapply(rownames(monkey.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(monkey.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][2]))))

##   
## 10um 20um 6um   
## 124 344 126

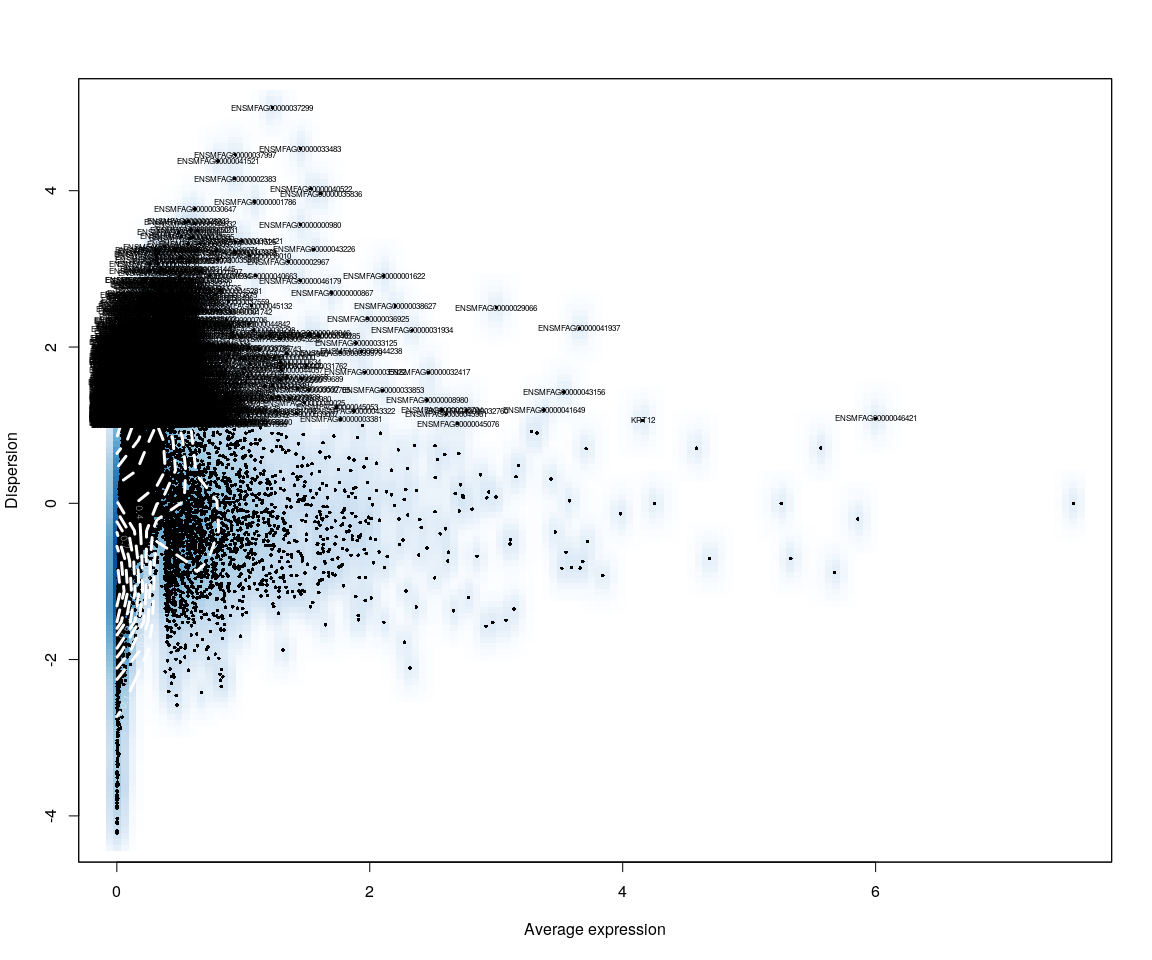
table(unlist(lapply(colnames(monkey.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][1]))))

##   
## mkc001 mkc003 mkc004 mkc005   
## 156 155 169 114

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

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



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

## all.sample.size  
## all.sample.group 10um 20um 6um  
## mkc001 36 88 29  
## mkc003 32 82 31  
## mkc004 49 44 22  
## mkc005 0 78 25

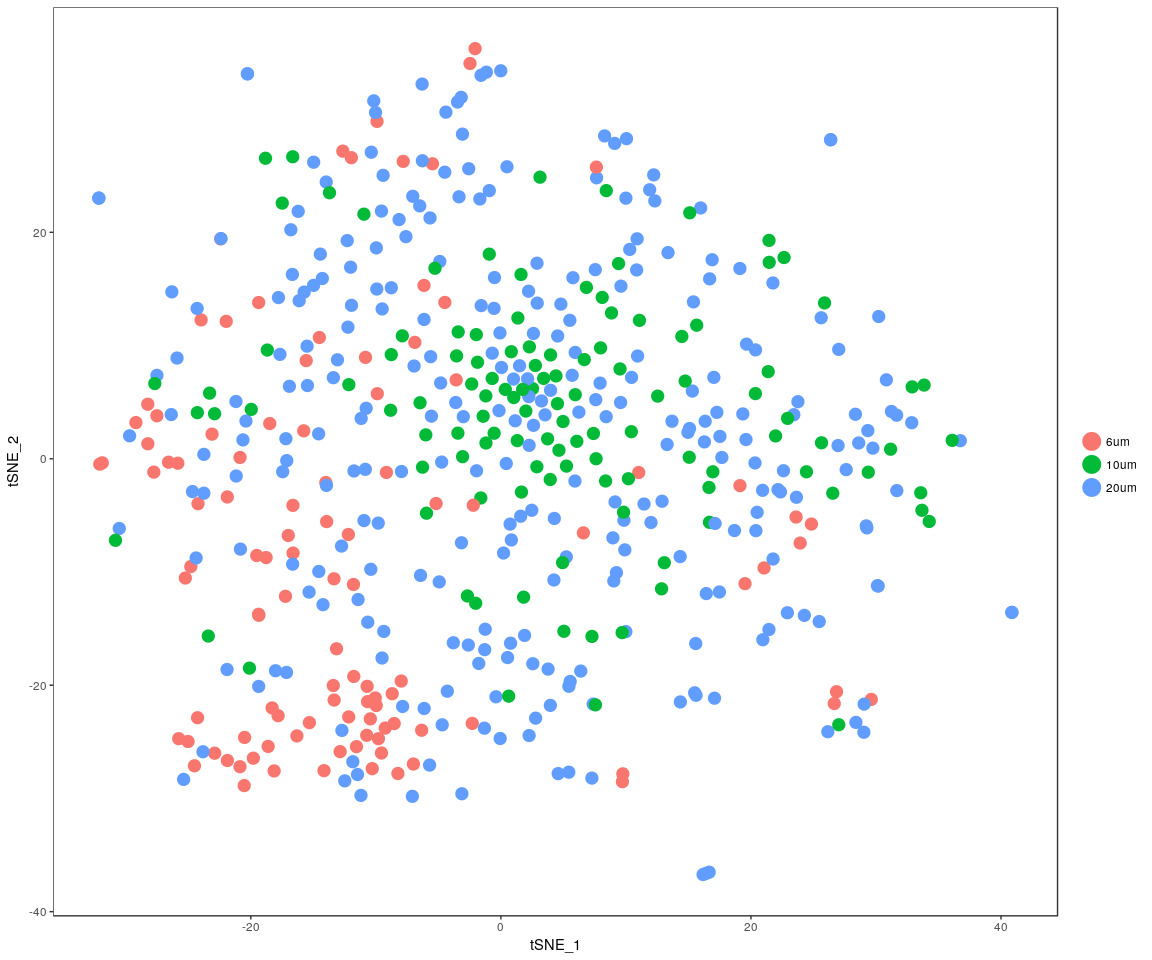
## Dimensionality reduction

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

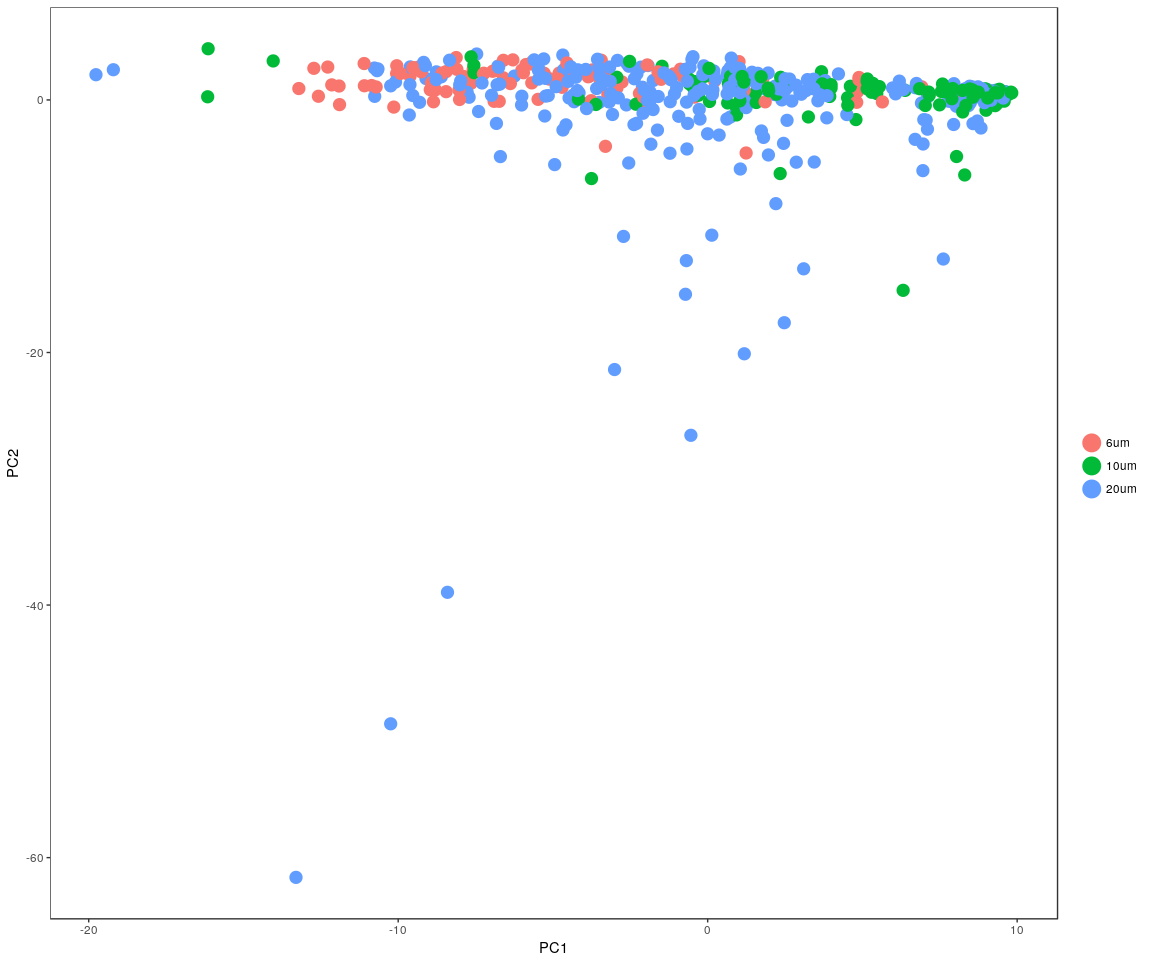
Here,do the dimensionality reduction using the PCA, tSNE method

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

DimPlot(all.pbmc, reduction.use = "tsne", pt.size = 4)



DimPlot(all.pbmc, reduction.use = "pca", pt.size = 4)

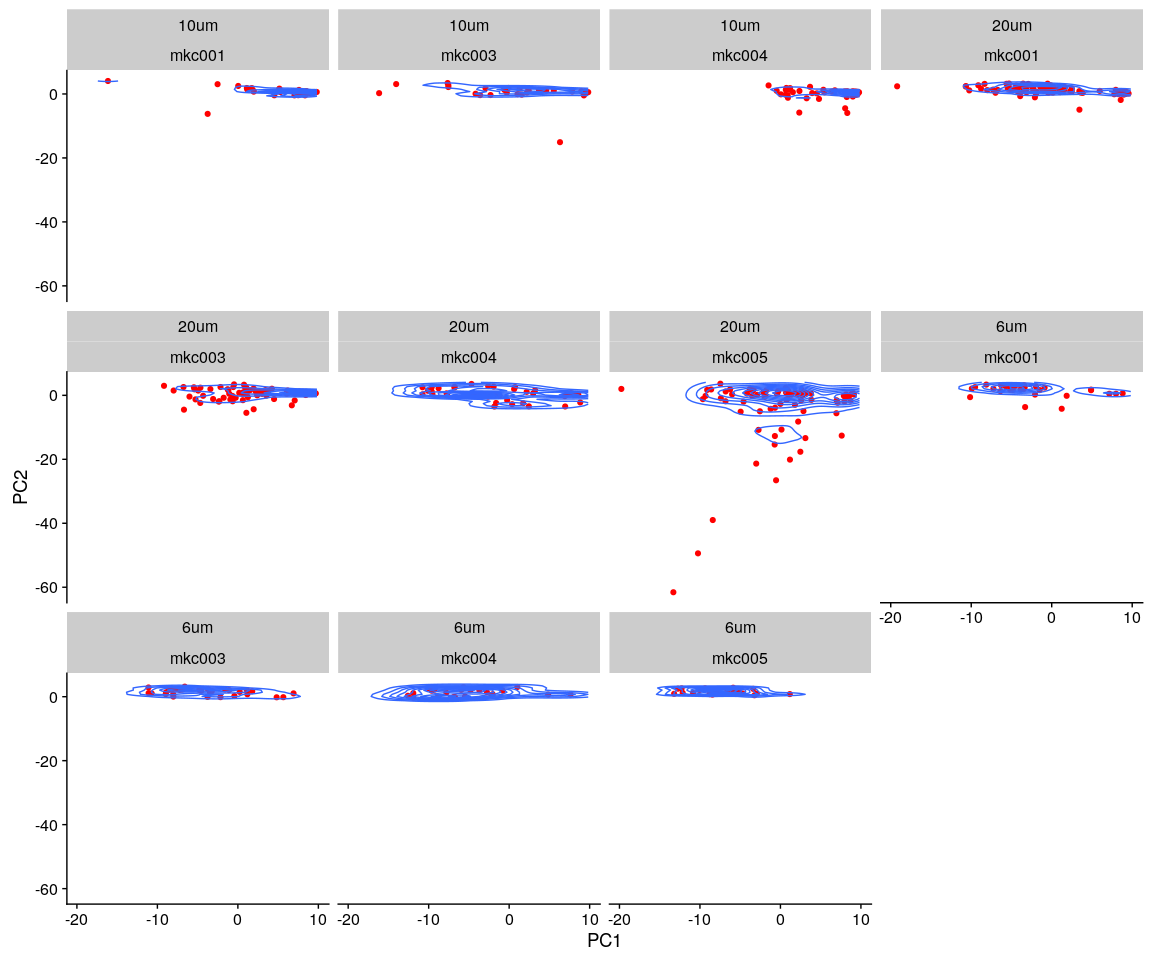


### In the sample cell size,compare the sample distribution

pca <- data.frame(all.pbmc@dr$pca@cell.embeddings)  
tsne <- data.frame(all.pbmc@dr$tsne@cell.embeddings)  
pca$cell.size <- all.sample.size  
pca$cell.group <- all.sample.group  
tsne$cell.size <- all.sample.size  
tsne$cell.group <- all.sample.group

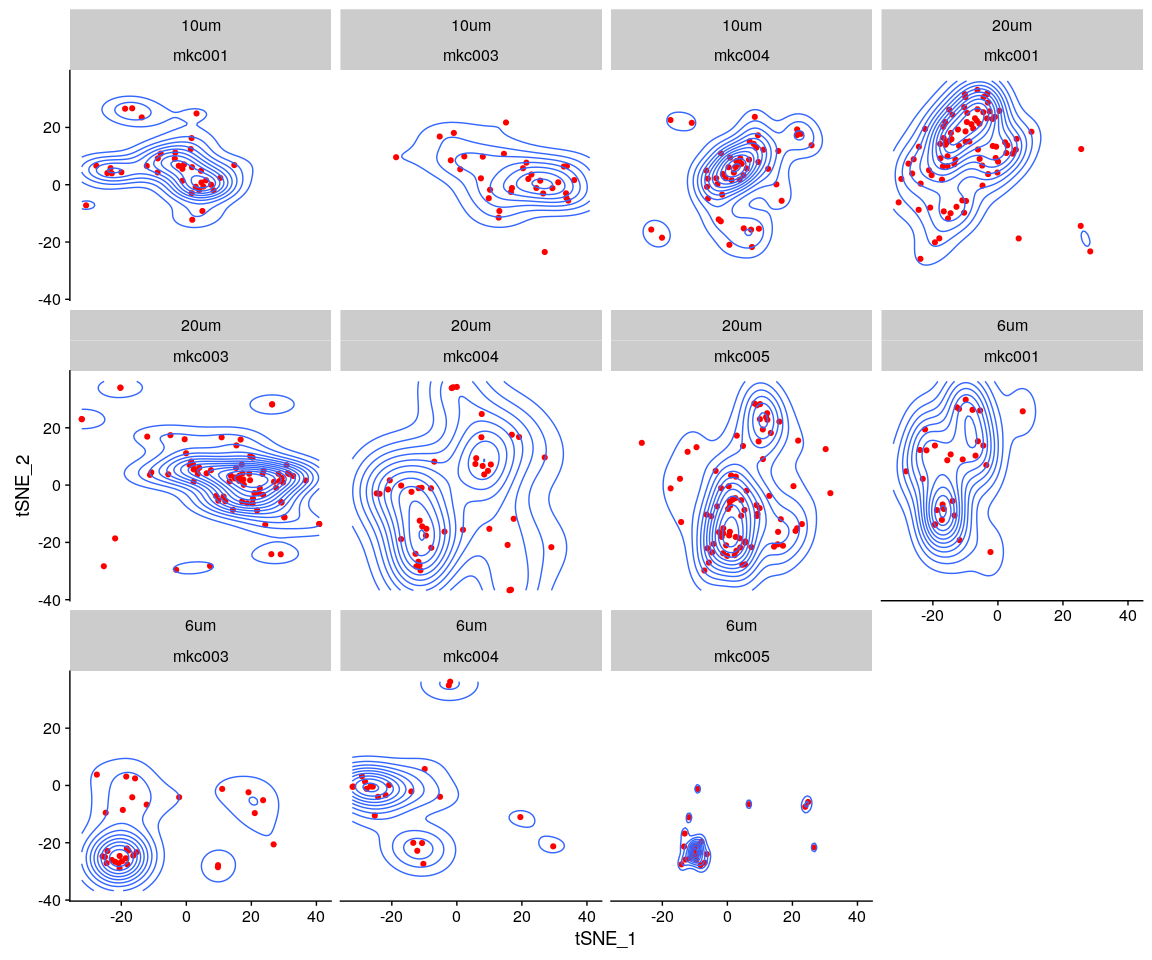
#### PCA method

ggplot(data = pca) + geom\_point(aes(x = PC1, y = PC2), color = "red") + geom\_density2d(aes(x = PC1,   
 y = PC2), contour = TRUE) + facet\_wrap(~cell.size + cell.group)



#### tSNE method

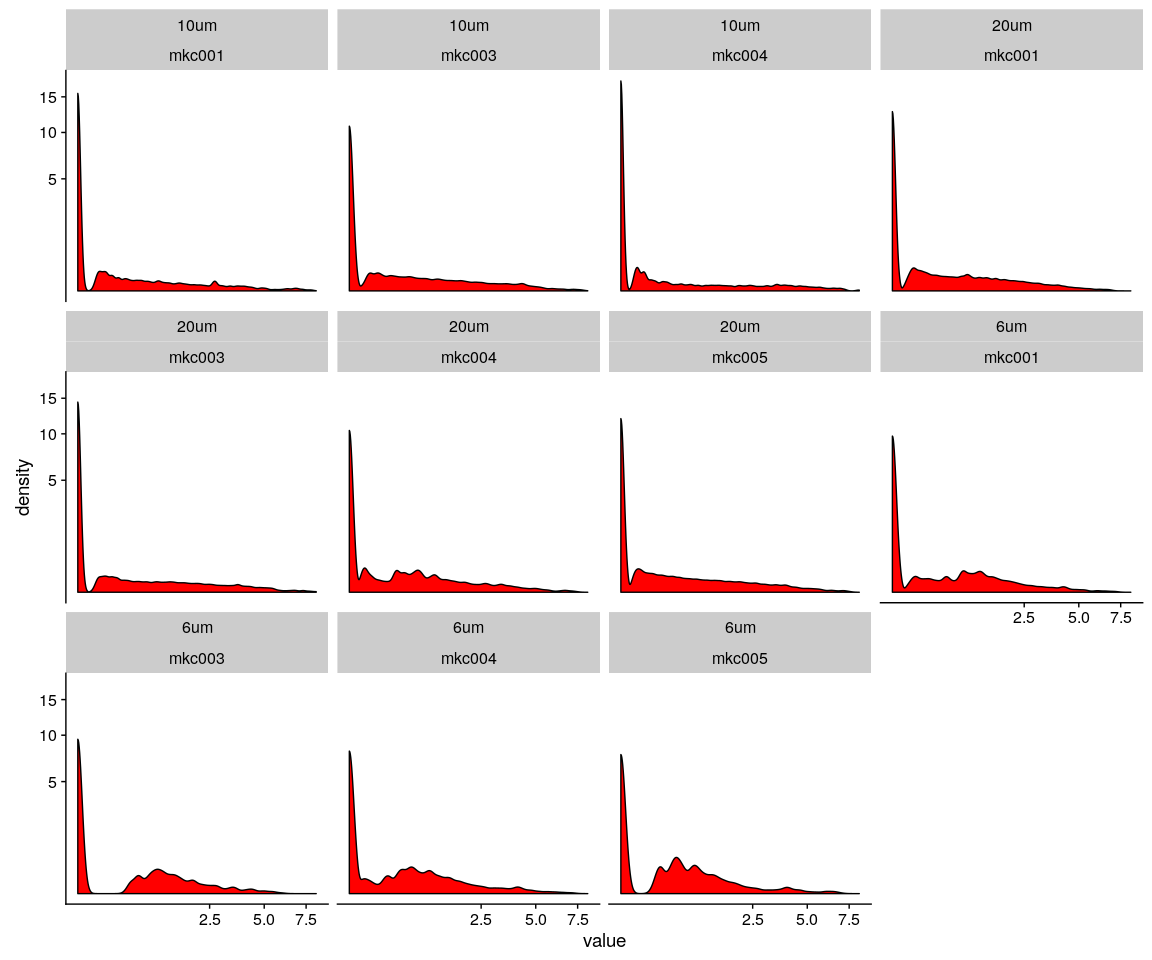
ggplot(data = tsne) + geom\_point(aes(x = tSNE\_1, y = tSNE\_2), color = "red") +   
 geom\_density2d(aes(x = tSNE\_1, y = tSNE\_2), contour = TRUE) + facet\_wrap(~cell.size +   
 cell.group)



#### density method

monkey.lognorm <- data.frame(FetchData(all.pbmc, vars.all = all.pbmc@var.genes))  
monkey.lognorm$cell.size <- all.sample.size  
monkey.lognorm$cell.group <- all.sample.group  
monkey.lognorm.melt <- melt(monkey.lognorm)

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



## Distribution test

#### the Kolmogorov–Smirnov test (K–S test or KS test) is a nonparametric test of the equality of continuous, one-dimensional probability distributions that can be used to compare a sample with a reference probability distribution (one-sample K–S test), or to compare two samples (two-sample K–S test)

##### The null distribution of this statistic is calculated under the null hypothesis that the sample is drawn from the reference distribution (in the one-sample case) or that the samples are drawn from the same distribution (in the two-sample case). In each case, the distributions considered under the null hypothesis are continuous distributions but are otherwise unrestricted

size.group.table <- as.data.frame(as.matrix(table(all.sample.group, all.sample.size)))  
size.group.table <- size.group.table[size.group.table$Freq > 0, ] # only test cell size,group both not null

### 6um:under the cell size 6um test

KS.test.comb(all.pbmc, genes = important.genes, cell\_size = "6um")

## 6um\_p.value  
## mkc001\_mkc003 1.250517e-02  
## mkc001\_mkc004 1.730060e-02  
## mkc001\_mkc005 8.006100e-07  
## mkc003\_mkc004 1.154220e-01  
## mkc003\_mkc005 2.158024e-03  
## mkc004\_mkc005 6.247041e-02

### 10um:under the cell size 10um test

KS.test.comb(all.pbmc, genes = important.genes, cell\_size = "10um")

## 10um\_p.value  
## mkc001\_mkc003 0.0014603018  
## mkc001\_mkc004 0.0337094959  
## mkc003\_mkc004 0.0000003298

### 20um:under the cell size 20um test

KS.test.comb(all.pbmc, genes = important.genes, cell\_size = "20um")

## 20um\_p.value  
## mkc001\_mkc003 4.666223e-11  
## mkc001\_mkc004 3.503614e-03  
## mkc001\_mkc005 1.127960e-04  
## mkc003\_mkc004 1.110223e-15  
## mkc003\_mkc005 1.885279e-03  
## mkc004\_mkc005 1.254279e-07