Deep Analysis on mouse

yejg

2018/1/30

### 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

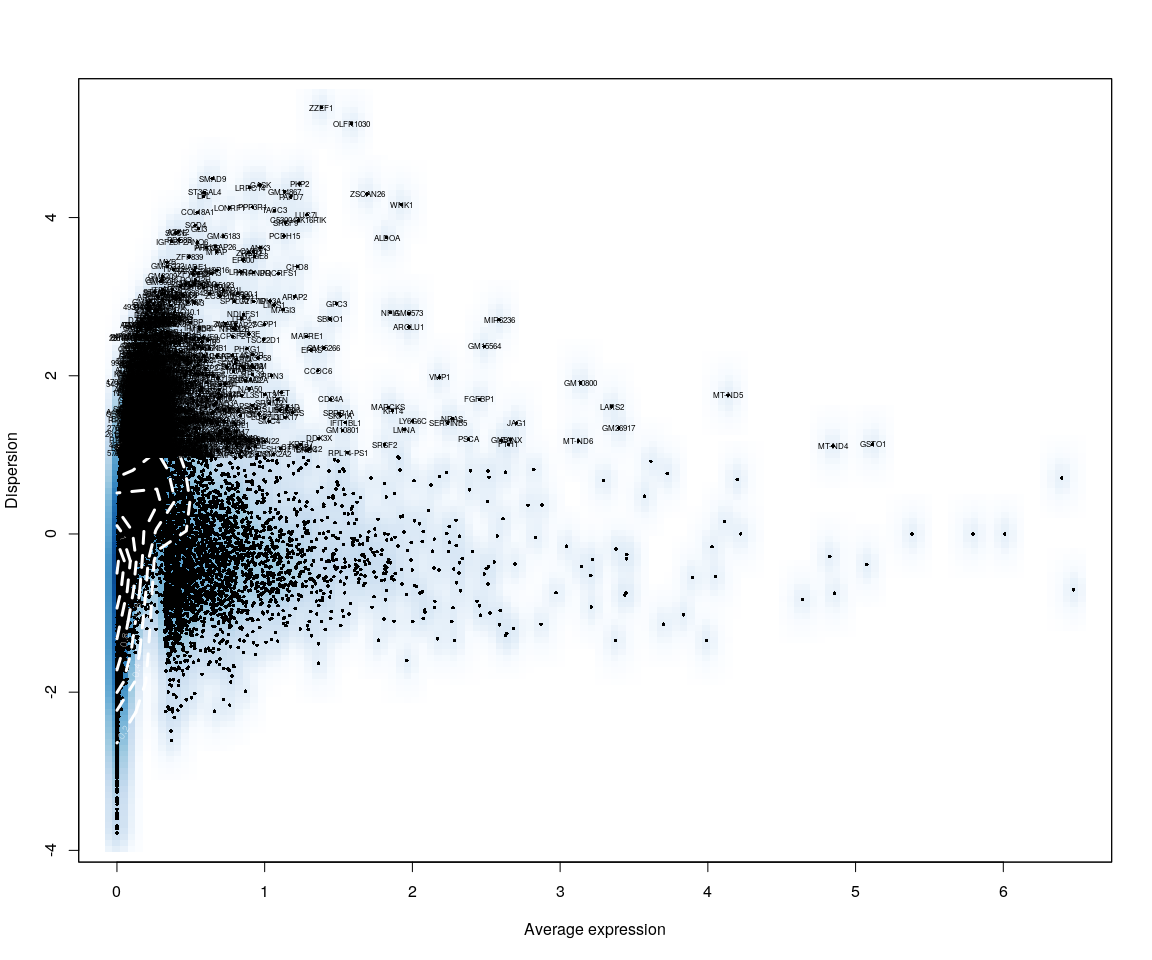
table(unlist(lapply(colnames(mouse.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][1]))))

##   
## 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)  
table(all.sample.group, all.sample.size)

## all.sample.size  
## all.sample.group 10um 20um 6um  
## mc002 0 83 0  
## mc004 0 54 0  
## mc005 57 0 0  
## mc006 0 64 0  
## mc007 70 0 0  
## mc008 0 34 0  
## mc009 60 0 0  
## mc010 0 57 0  
## mc011 0 84 0  
## mc012 0 55 0  
## mc013 0 0 31  
## mc014 0 0 31  
## mc015 0 0 31  
## mc016 0 0 31  
## mc017 0 0 30  
## mc018 0 0 30

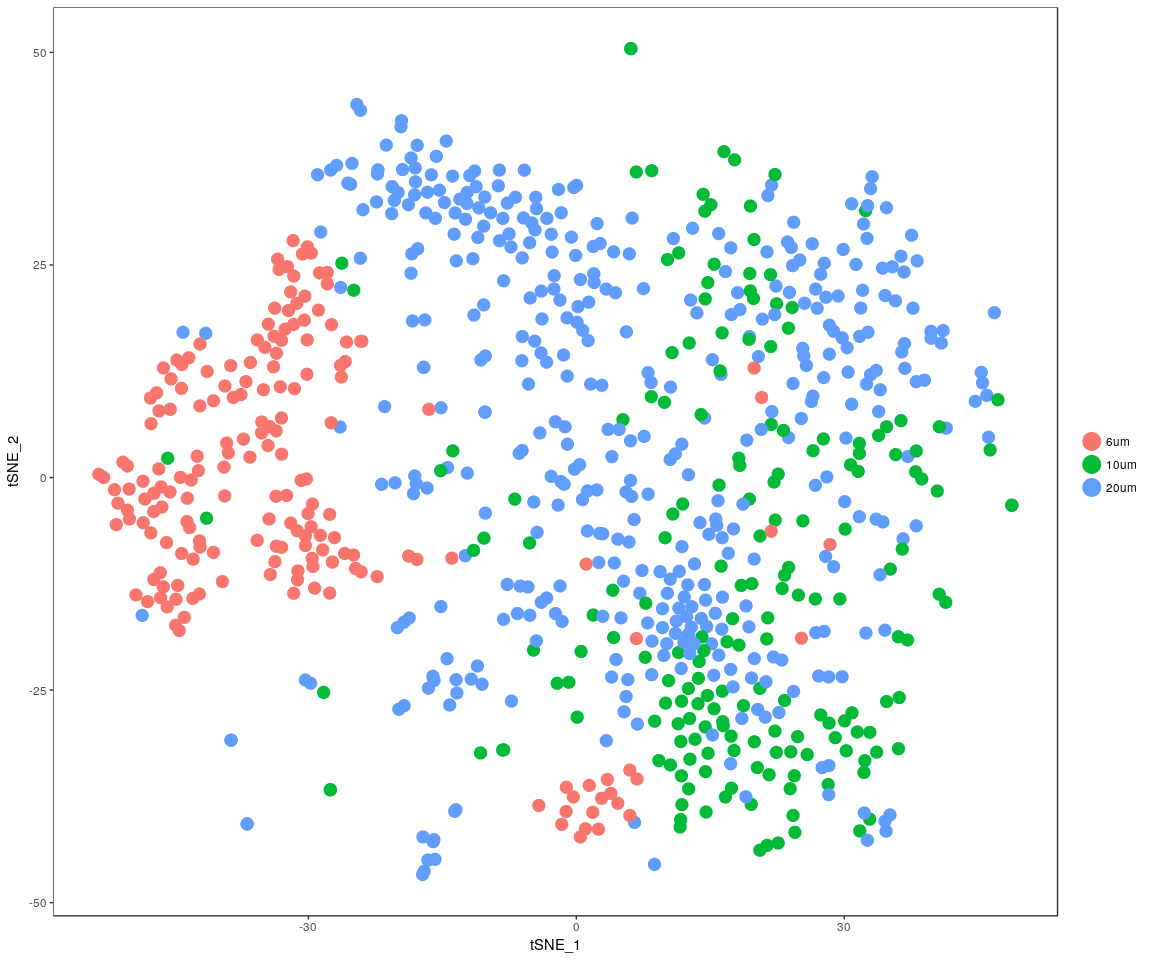
## Dimensionality reduction

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

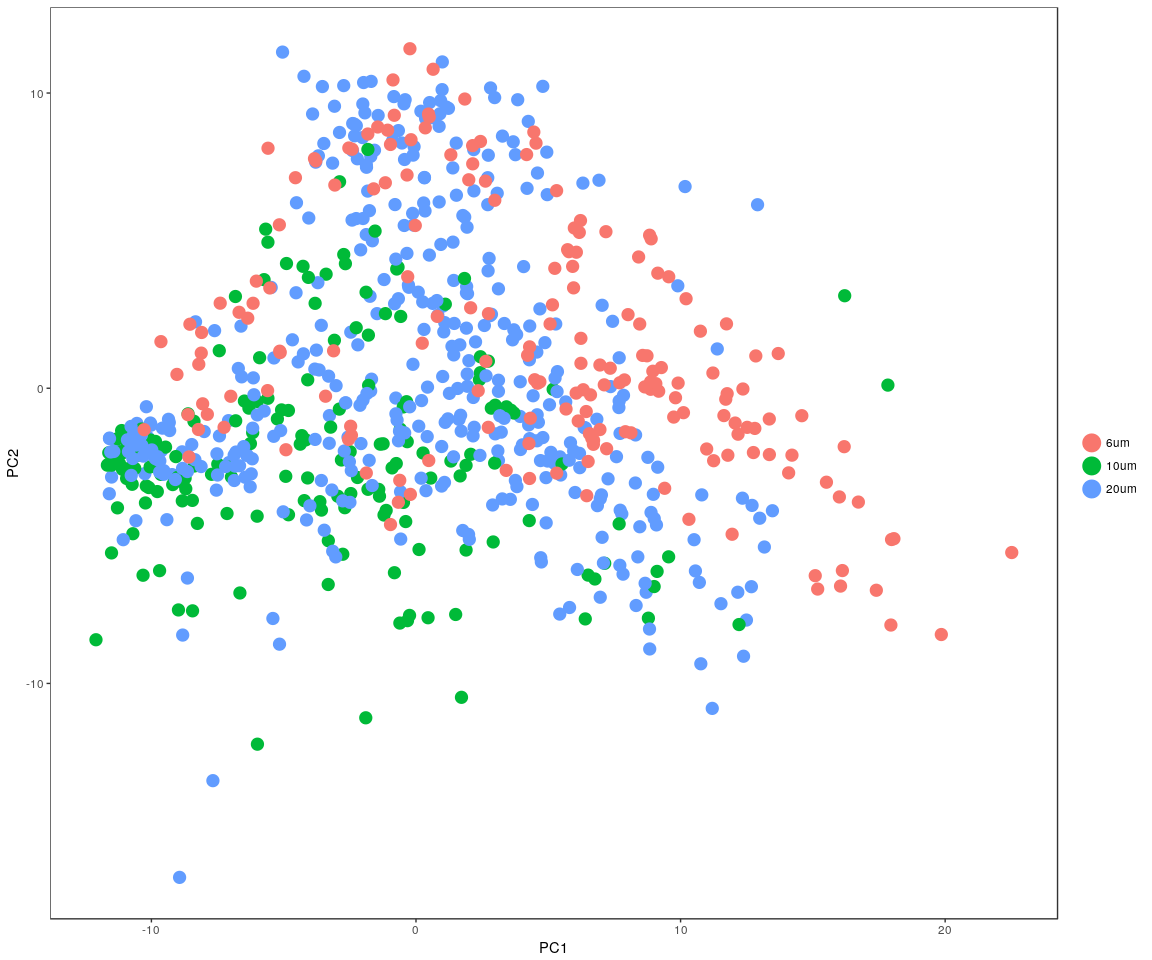
Here,do the dimensionality reduction using the PCA, tSNE method

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

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



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

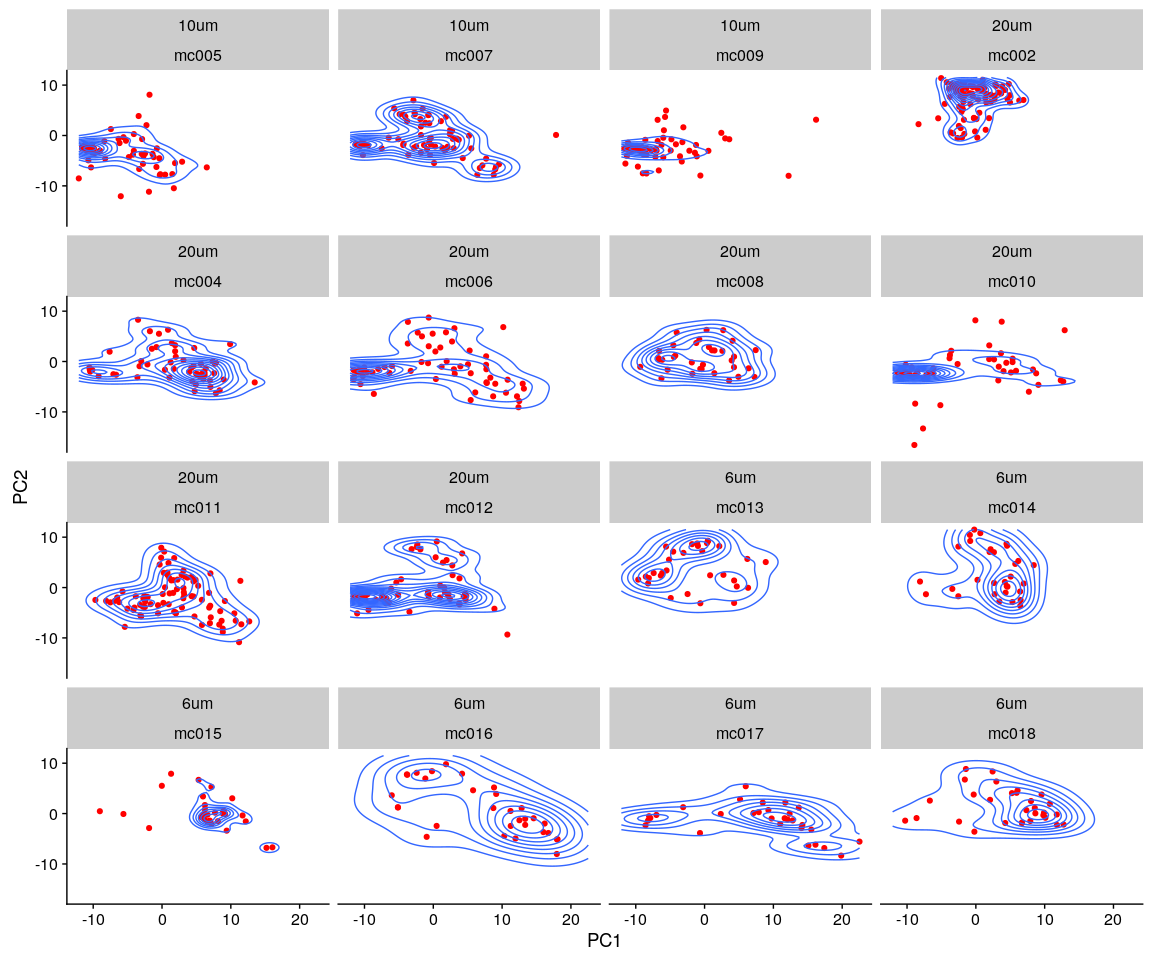


### In the sample cell size,compare the sample distribution

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

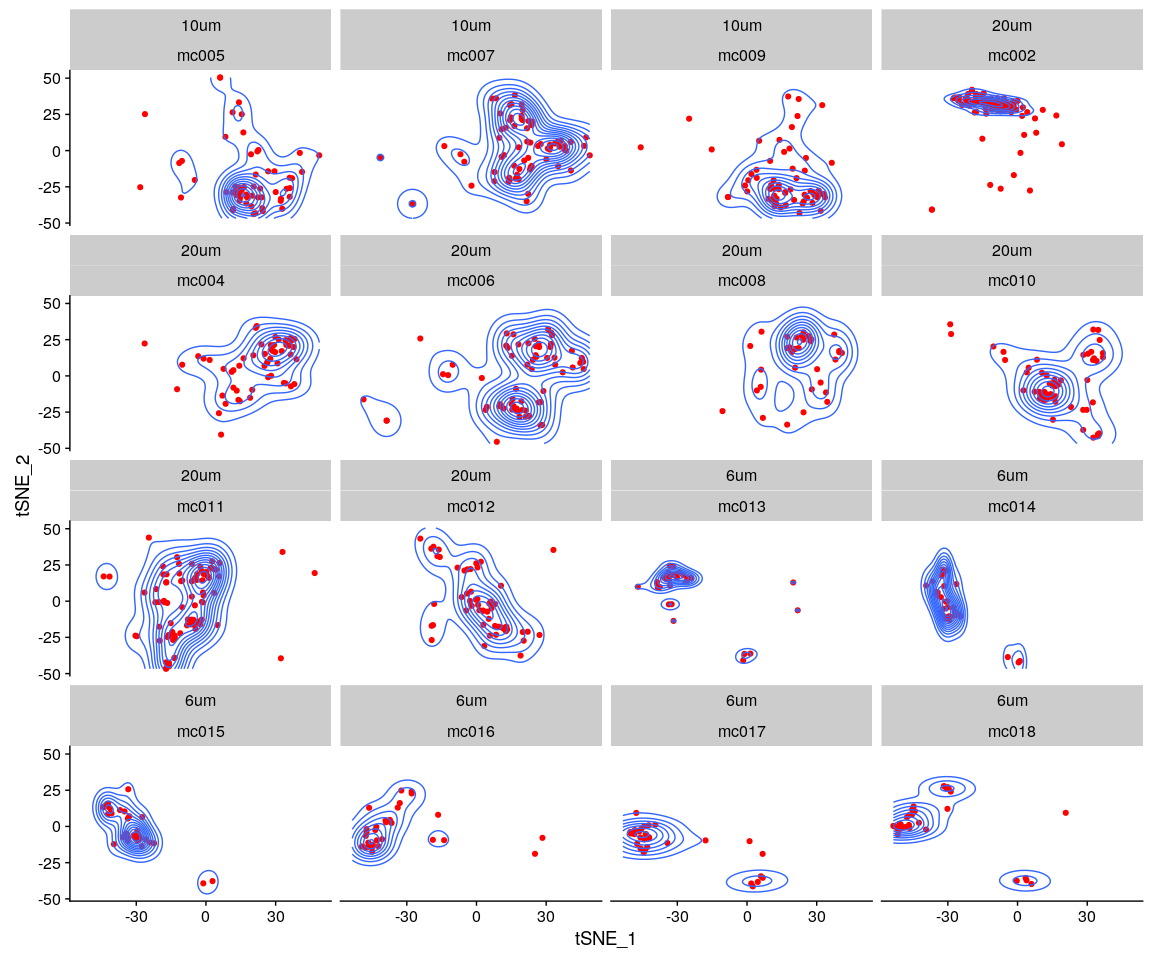
#### PCA method

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

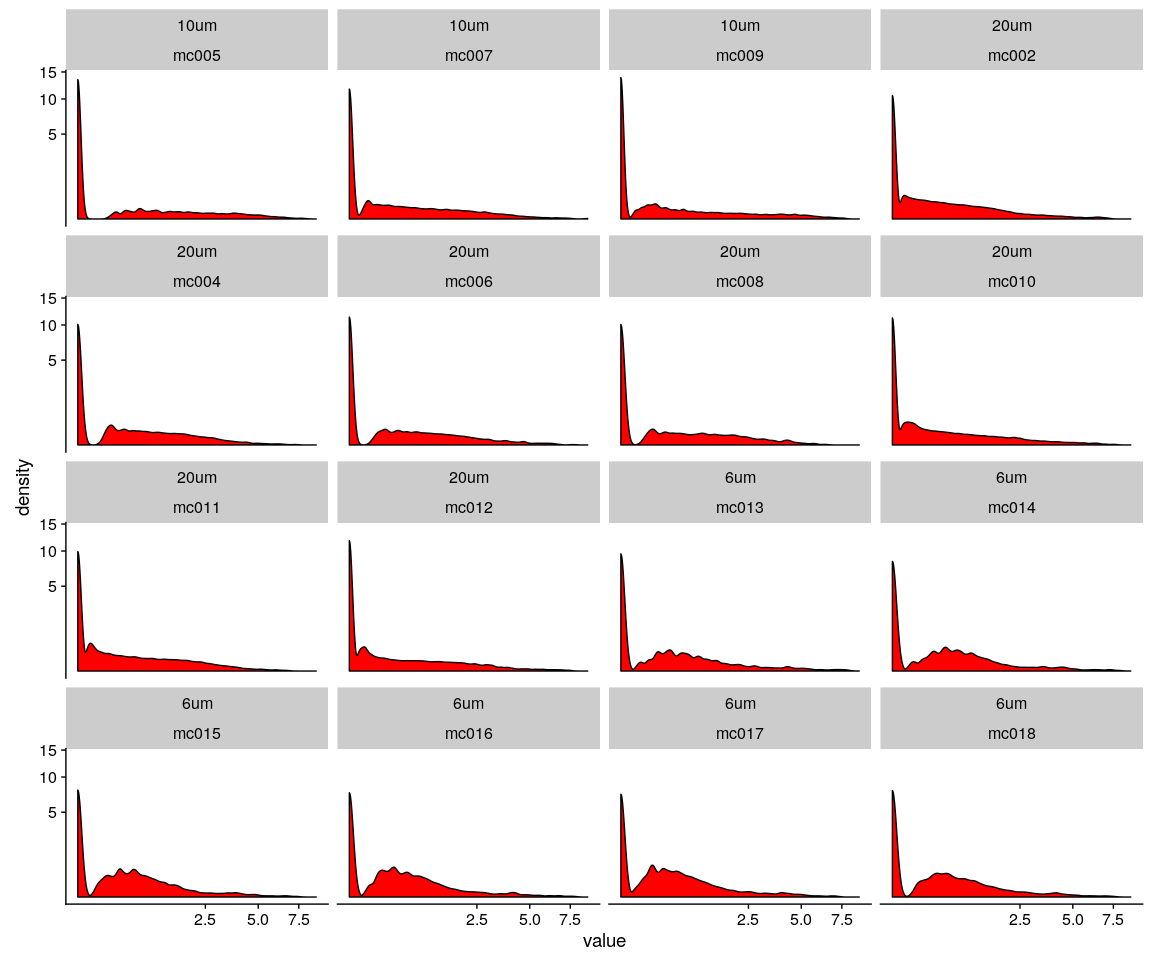
ggplot(data = tsne) + geom\_point(aes(x = tSNE\_1, y = tSNE\_2), color = "red") +   
 geom\_density2d(aes(x = tSNE\_1, y = tSNE\_2), contour = TRUE) + facet\_wrap(~cell.size +   
 cell.group)



#### density method

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

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



#### 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  
## mc013\_mc014 2.657110e-01  
## mc013\_mc015 3.257327e-04  
## mc013\_mc016 5.614921e-05  
## mc013\_mc017 2.889420e-05  
## mc013\_mc018 3.031103e-03  
## mc014\_mc015 3.057717e-02  
## mc014\_mc016 1.896959e-02  
## mc014\_mc017 3.951401e-03  
## mc014\_mc018 1.079105e-01  
## mc015\_mc016 5.961891e-02  
## mc015\_mc017 1.751558e-01  
## mc015\_mc018 2.200257e-01  
## mc016\_mc017 1.856361e-01  
## mc016\_mc018 2.325075e-01  
## mc017\_mc018 4.241744e-02

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

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

## 10um\_p.value  
## mc005\_mc007 3.413408e-09  
## mc005\_mc009 1.396489e-03  
## mc007\_mc009 1.820100e-12

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

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

## 20um\_p.value  
## mc002\_mc004 1.139726e-03  
## mc002\_mc006 6.628756e-05  
## mc002\_mc008 3.995832e-04  
## mc002\_mc010 7.058044e-06  
## mc002\_mc011 2.355446e-02  
## mc002\_mc012 3.497748e-02  
## mc004\_mc006 3.351270e-01  
## mc004\_mc008 4.375959e-01  
## mc004\_mc010 3.818680e-06  
## mc004\_mc011 7.226579e-05  
## mc004\_mc012 4.948285e-02  
## mc006\_mc008 1.271490e-01  
## mc006\_mc010 1.625265e-05  
## mc006\_mc011 2.232755e-06  
## mc006\_mc012 2.644405e-02  
## mc008\_mc010 2.423119e-06  
## mc008\_mc011 3.328080e-05  
## mc008\_mc012 8.358292e-03  
## mc010\_mc011 1.395645e-07  
## mc010\_mc012 3.479084e-02  
## mc011\_mc012 5.070303e-03