Monkey analysis on cell size

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")  
important.genes <- c("ITGB4", "ABCB5", "KRT19", "ACTB", "KRT12", "KRT5", "GAPDH",   
 "KRT3", "PAX6", "WNT7A", "KRT14", "TP63", "KRT10")  
  
table(unlist(lapply(colnames(monkey.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][2]))))

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

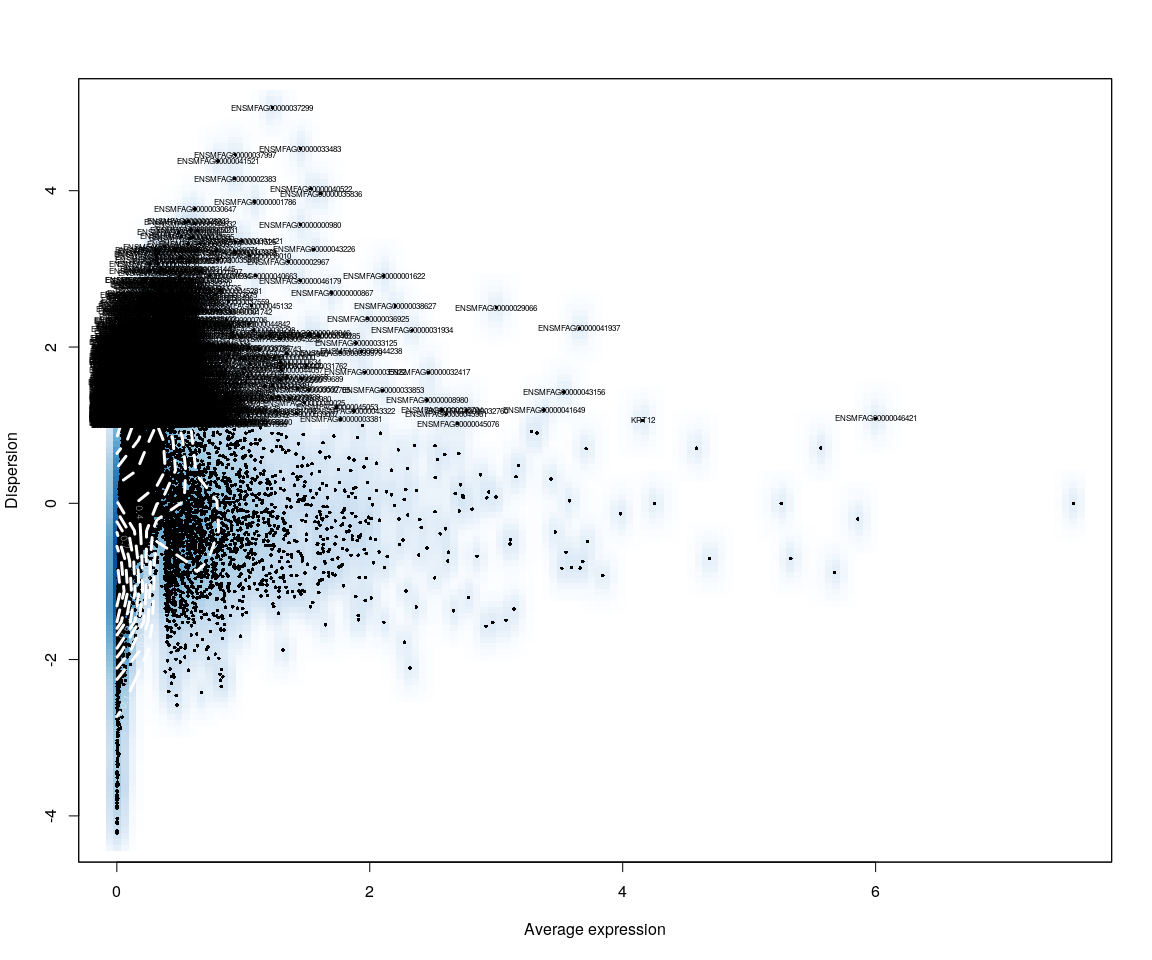
table(unlist(lapply(colnames(monkey.only.pro), function(x) return(str\_split(x,   
 "\_")[[1]][1]))))

##   
## 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)

## Figure Explore

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

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

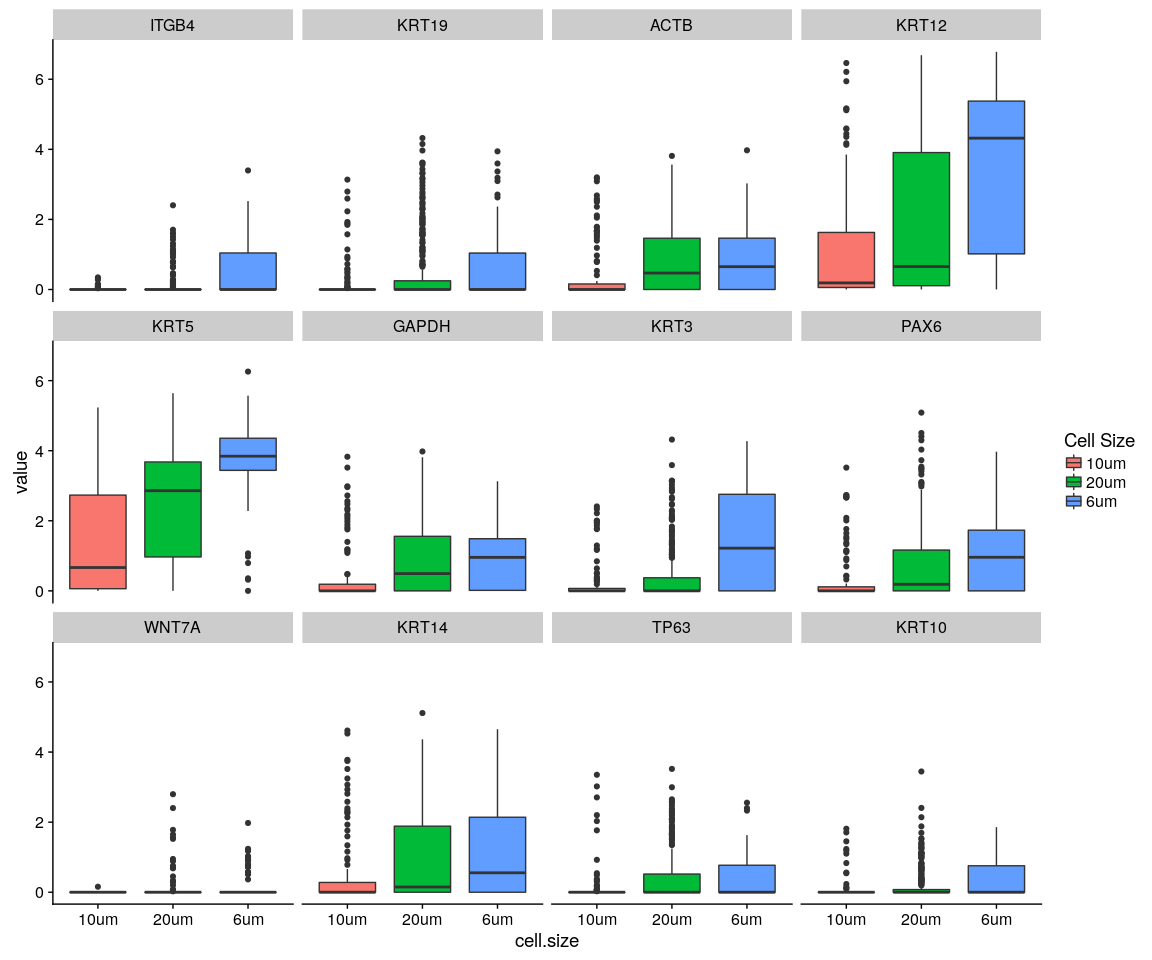
### Figure Explore.1

#### Violin

p <- ggplot(data = monkey.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"))

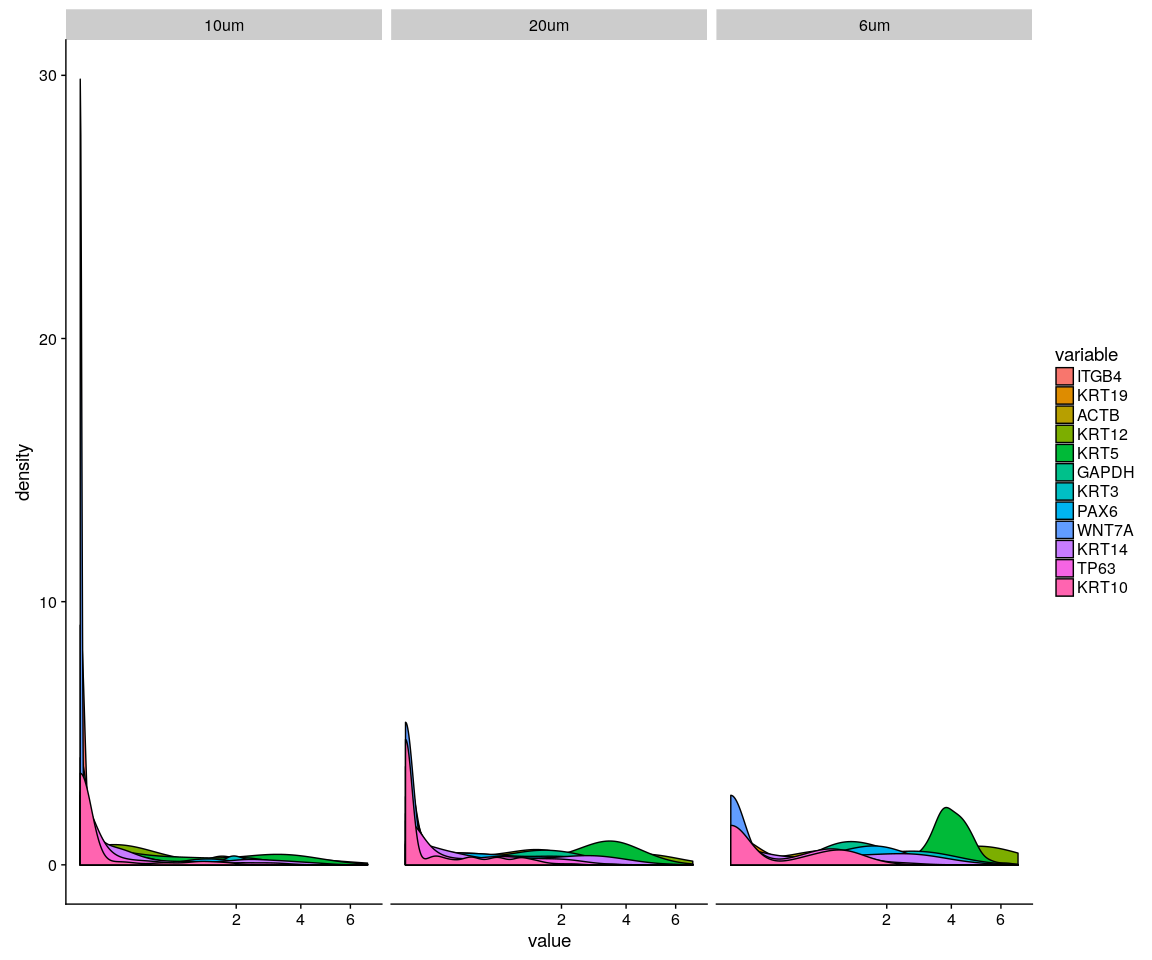
#### Boxplot

p <- ggplot(data = monkey.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)

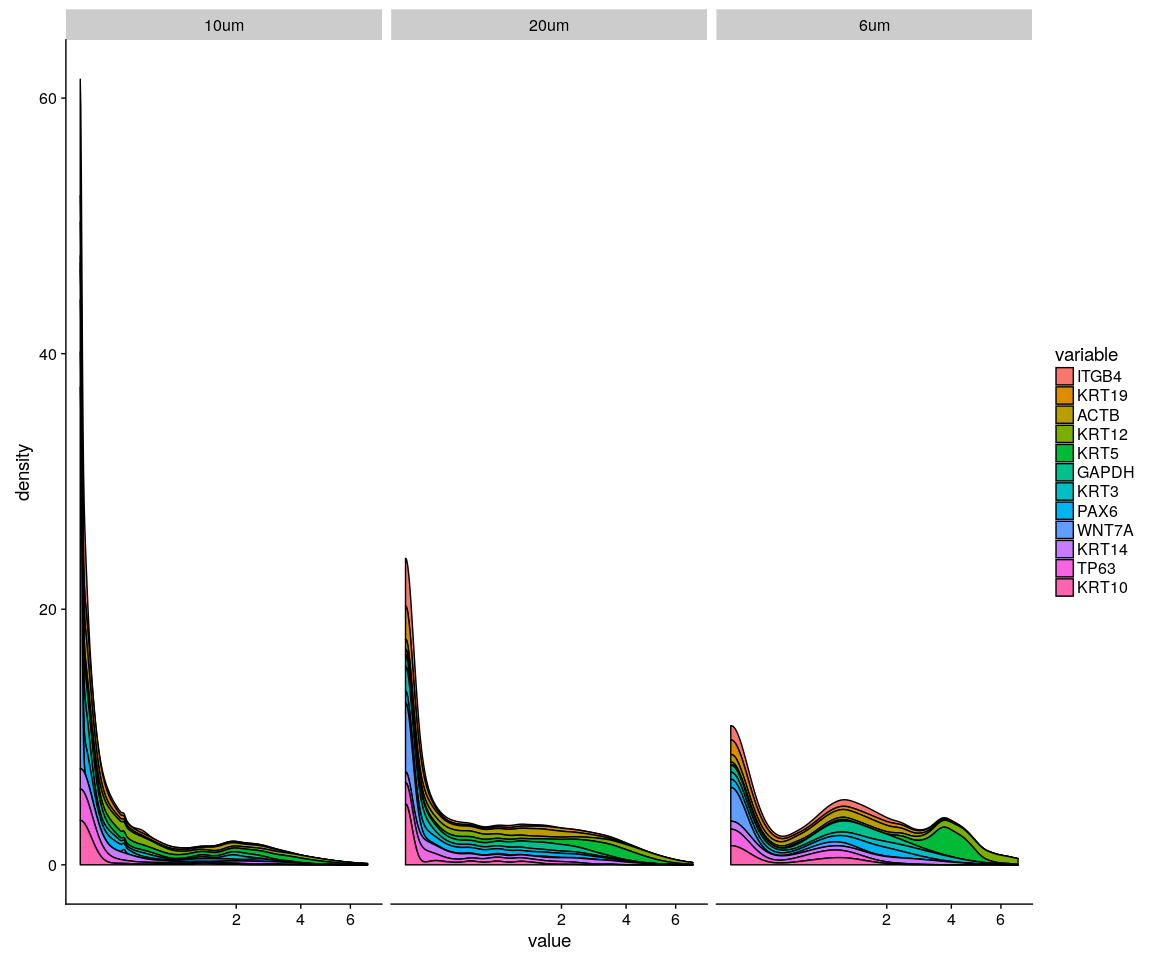


#### Density,histogram

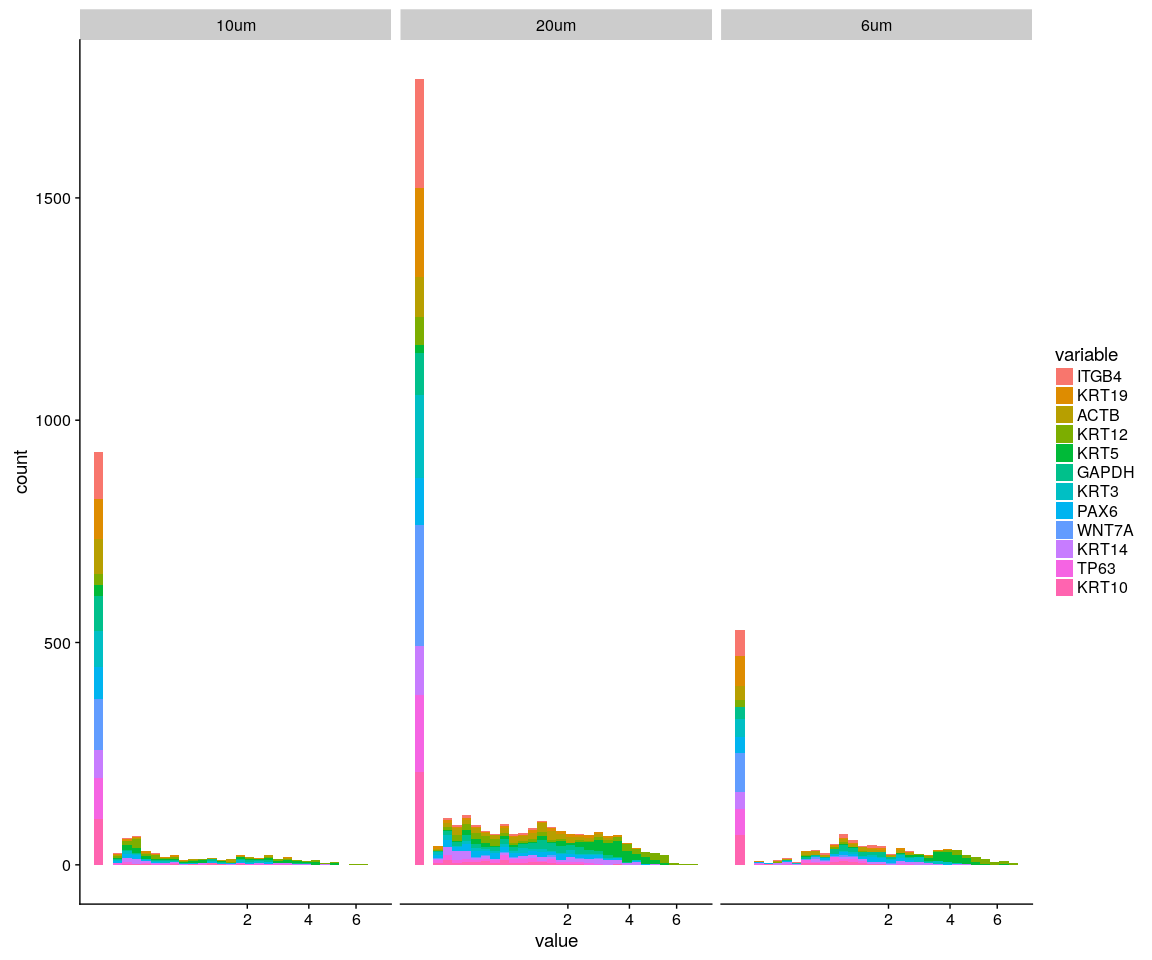
ggplot(data = monkey.imp.lognorm.melt, aes(x = value, fill = variable)) + geom\_density(kernel = "gaussian") +   
 scale\_x\_sqrt() + facet\_wrap(~cell.size)



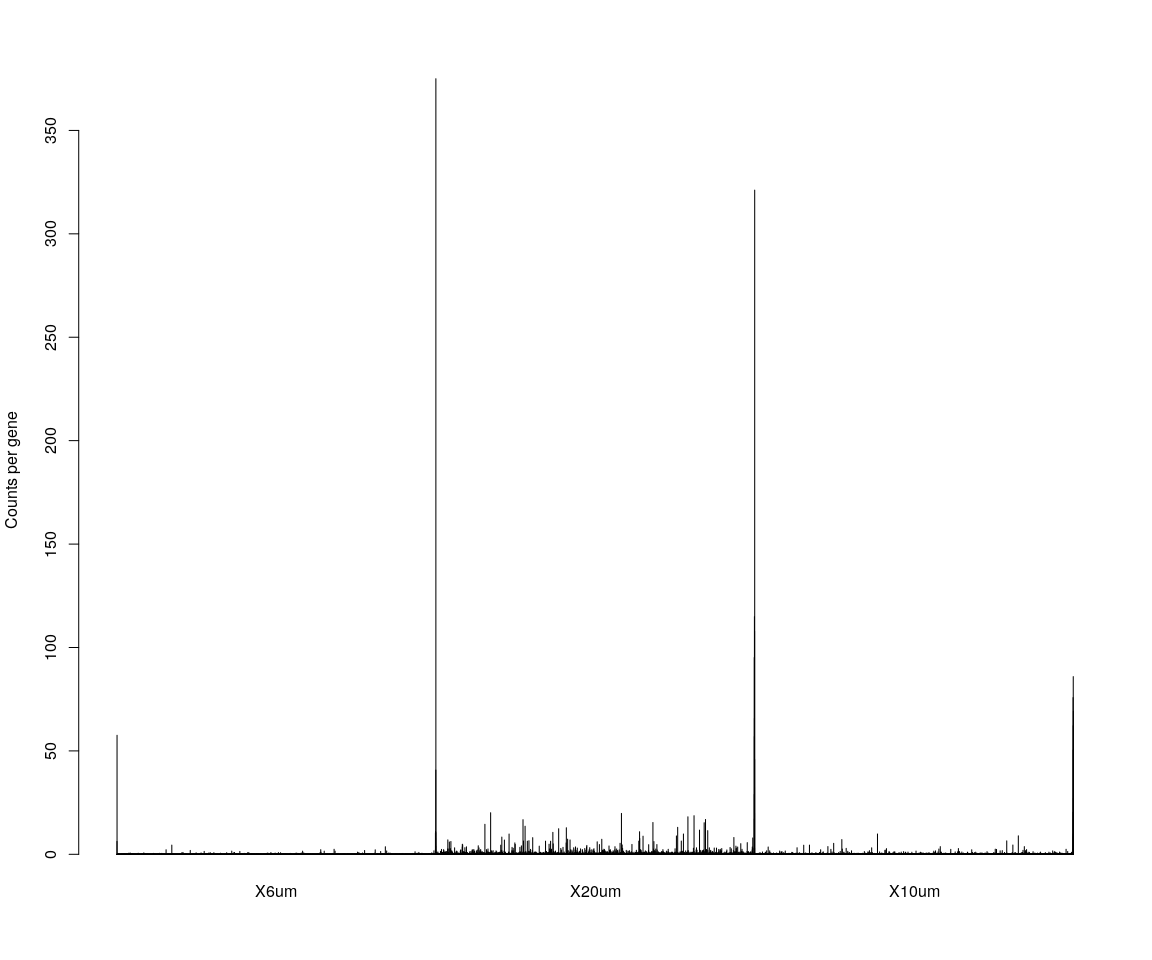
ggplot(data = monkey.imp.lognorm.melt, aes(x = value, fill = variable)) + geom\_density(kernel = "gaussian",   
 position = "stack") + scale\_x\_sqrt() + facet\_wrap(~cell.size)



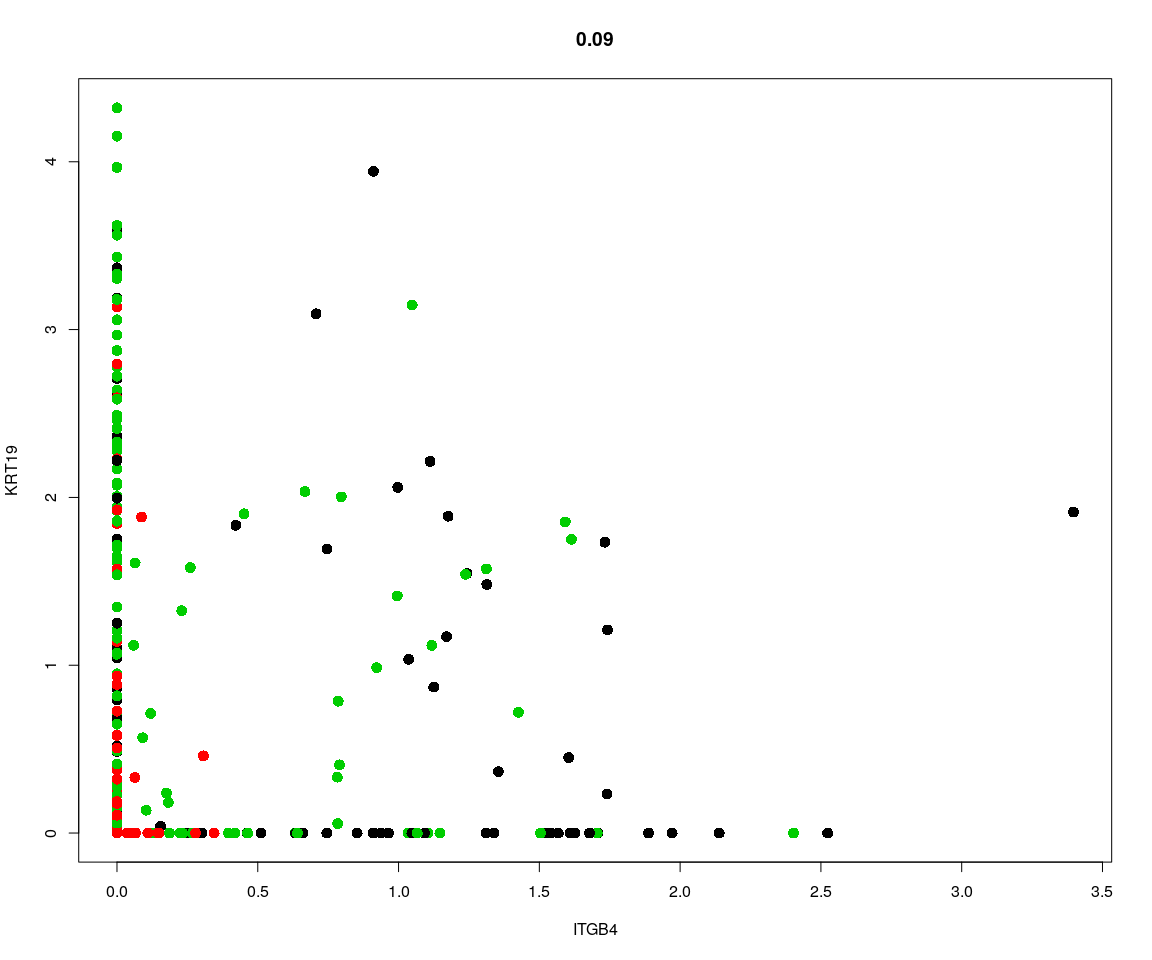
ggplot(data = monkey.imp.lognorm.melt, aes(x = value, fill = variable)) + geom\_histogram() +   
 scale\_x\_sqrt() + facet\_wrap(~cell.size)



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



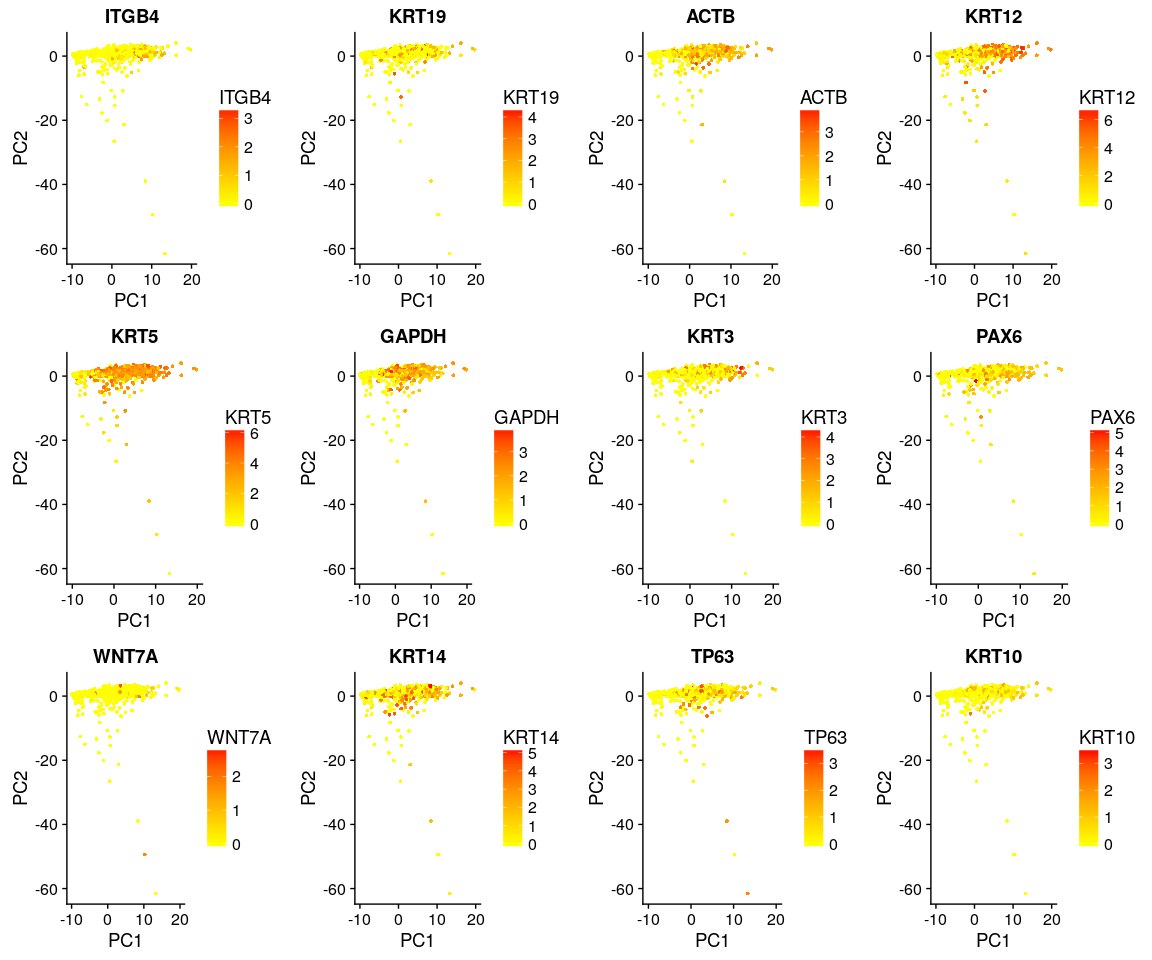
# We are interested in the gene ITGB4  
GenePlot(monkey.all.pbmc, gene1 = "ITGB4", gene2 = important.genes[3])

 ## Dimensionality reduction ### **PCA** and **tSNE** Here,do the dimensionality reduction using the PCA, tSNE method

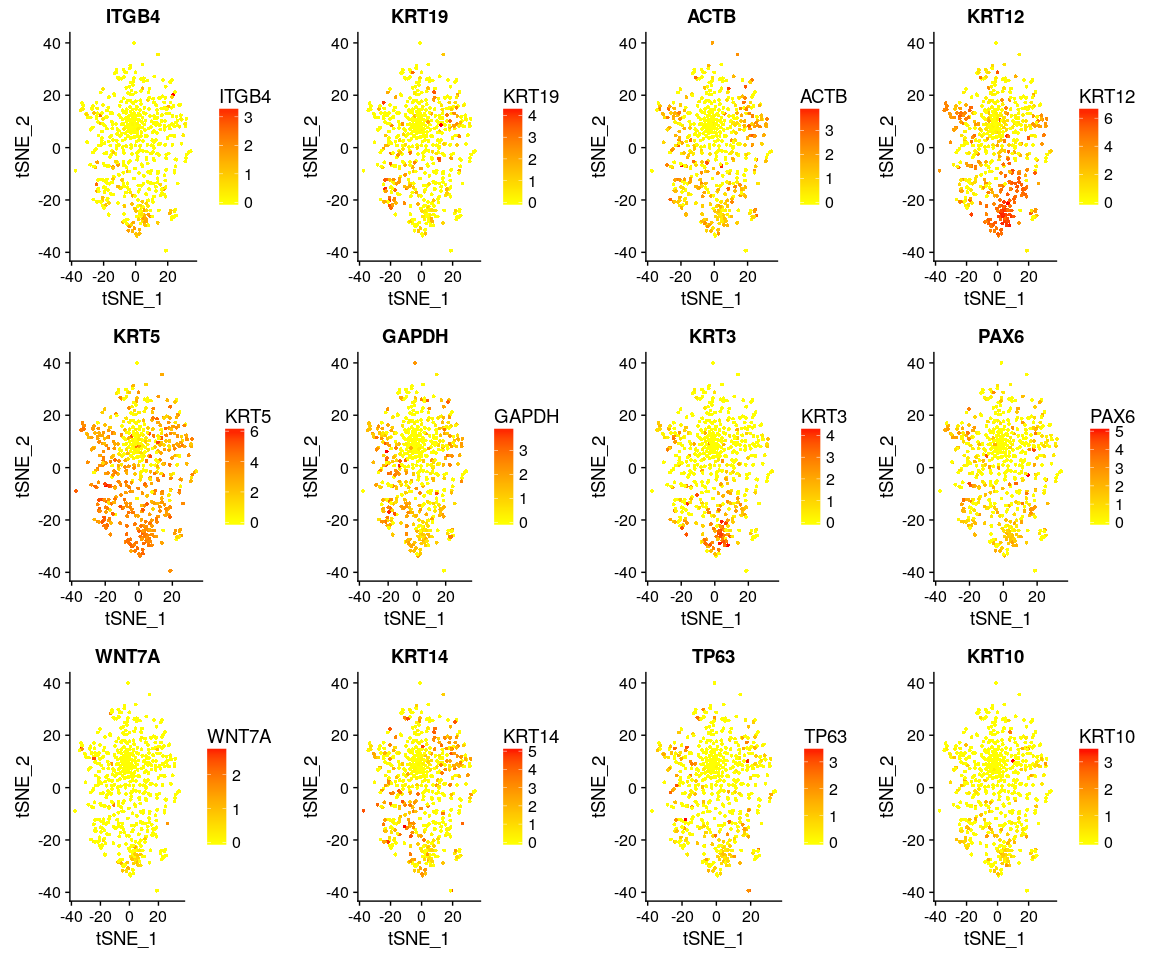
all.pbmc <- PCA.TSNE(object = monkey.all.pbmc, pcs.compute = FALSE, num.pcs = 28)

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

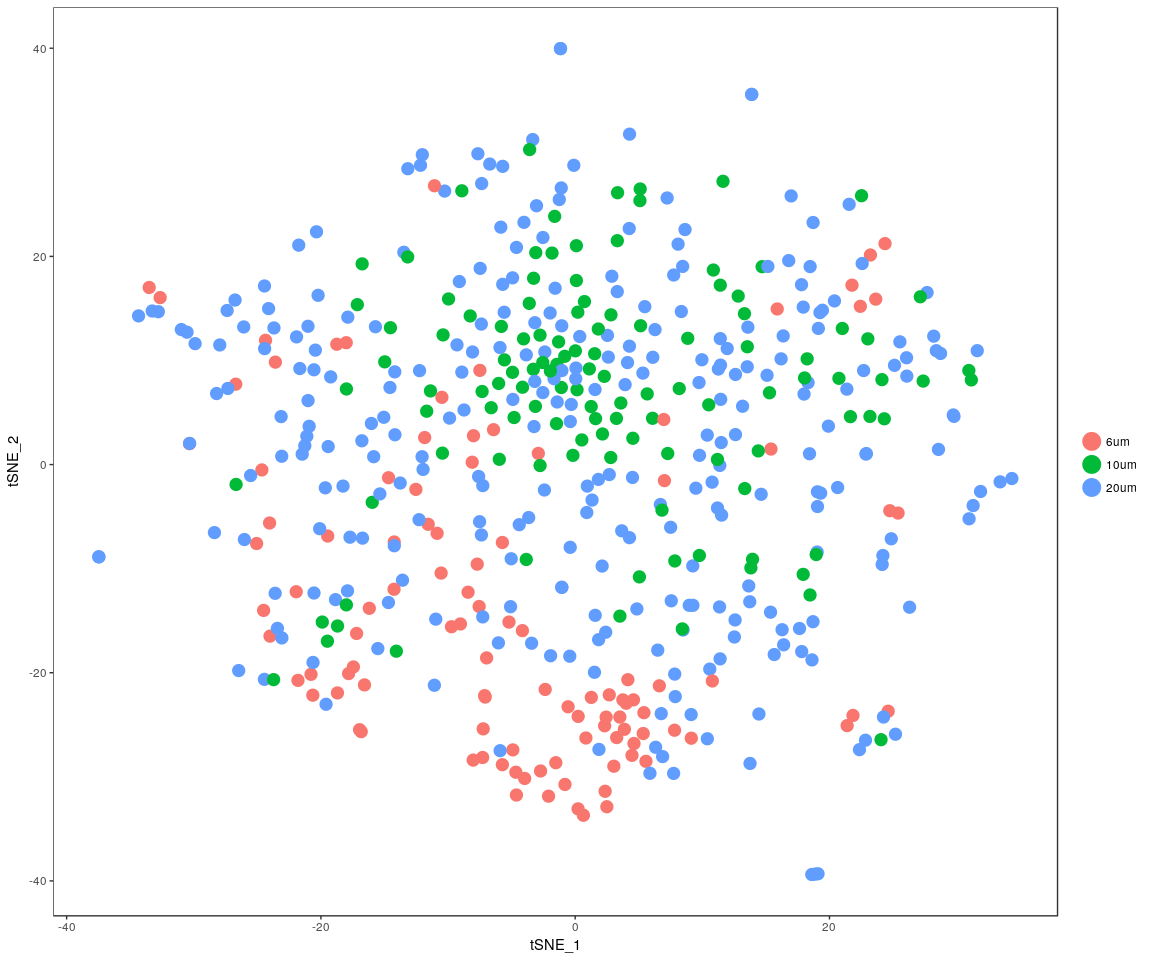
# FeaturePlot(object = all.pbmc,features.plot ='ITGB4',pt.size = 4,no.legend  
# = FALSE) # ITGB4 gene in part dataset  
FeaturePlot(object = all.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(all.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "pca")



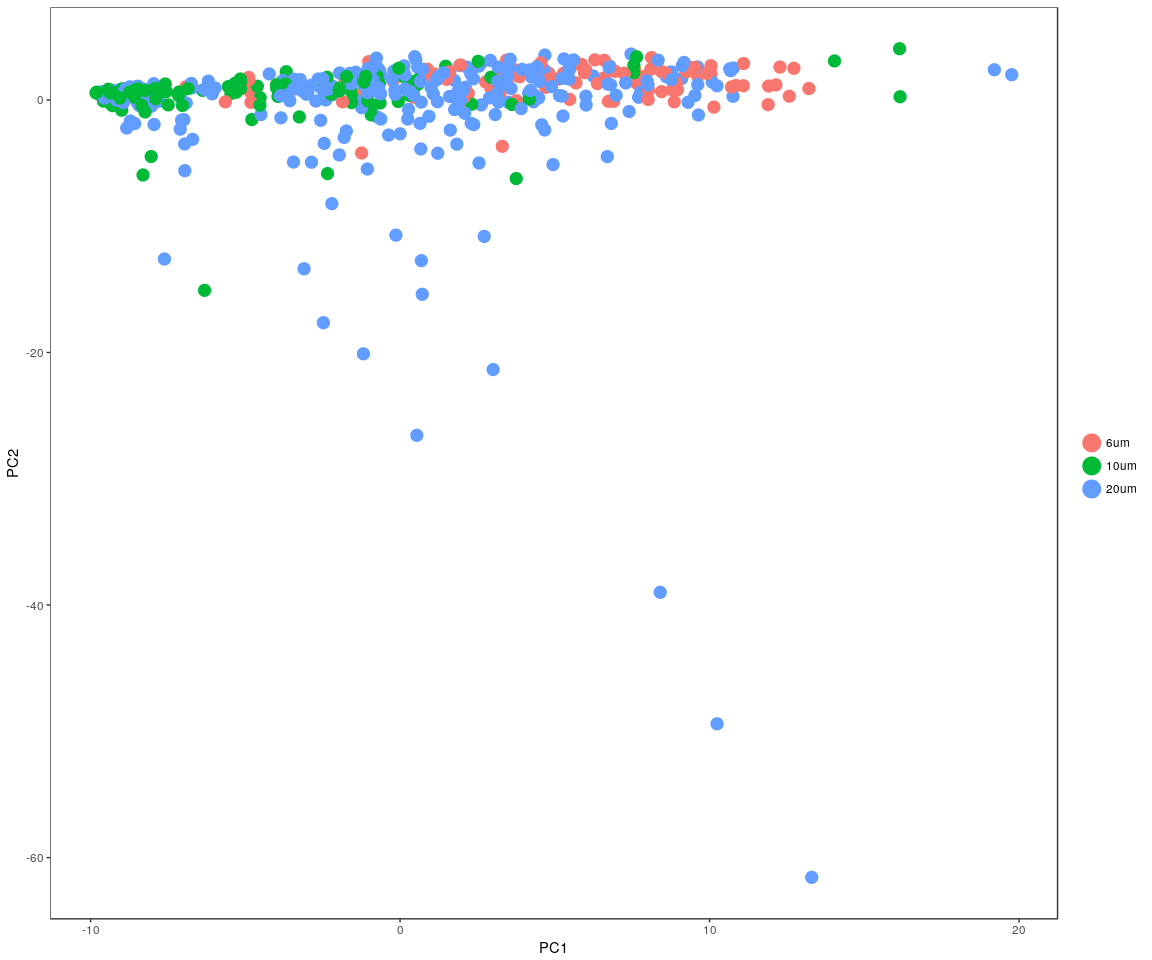
FeaturePlot(object = all.pbmc, features.plot = important.genes[important.genes %in%   
 rownames(all.pbmc@raw.data)], pt.size = 1, no.legend = FALSE, reduction.use = "tsne")



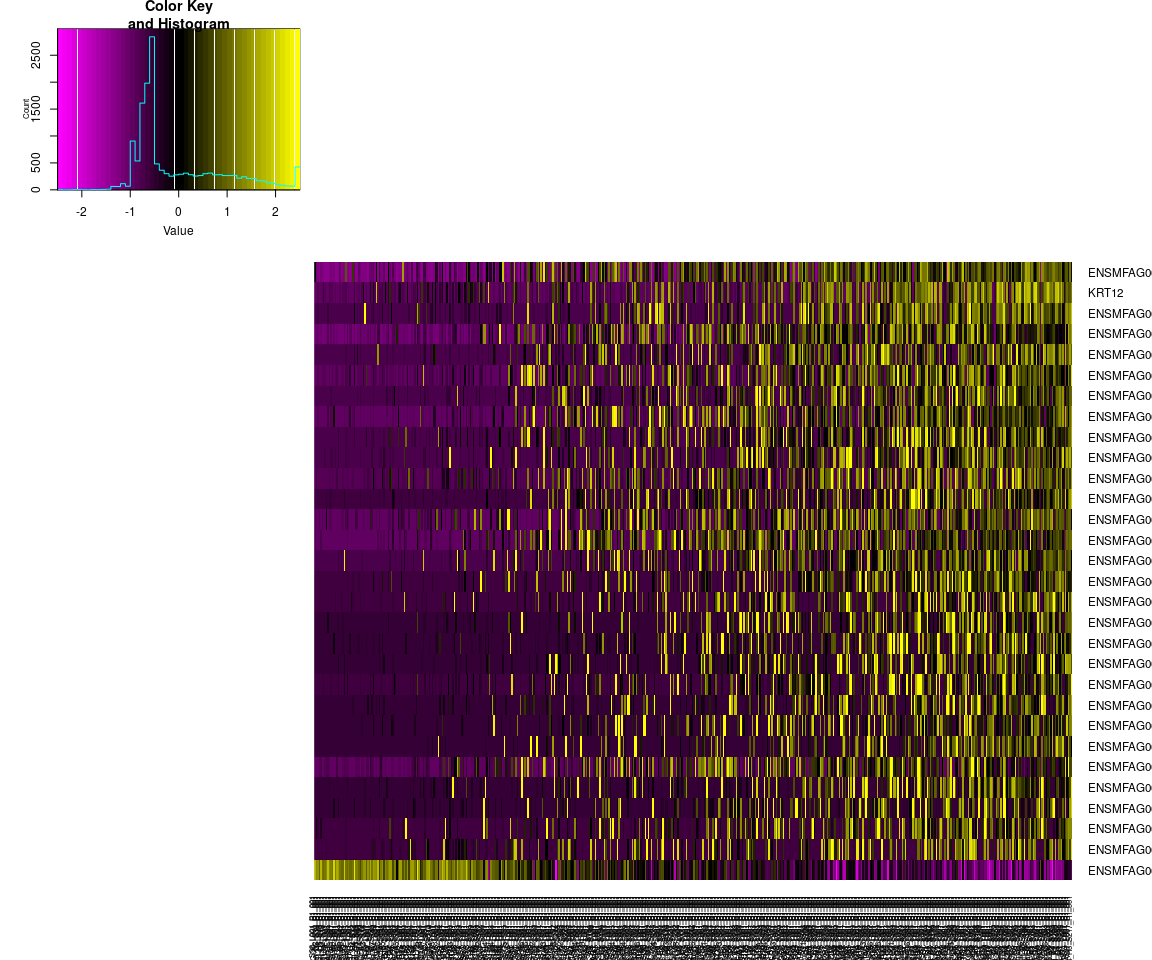
DimPlot(all.pbmc, reduction.use = "tsne", pt.size = 4) # grour by sample



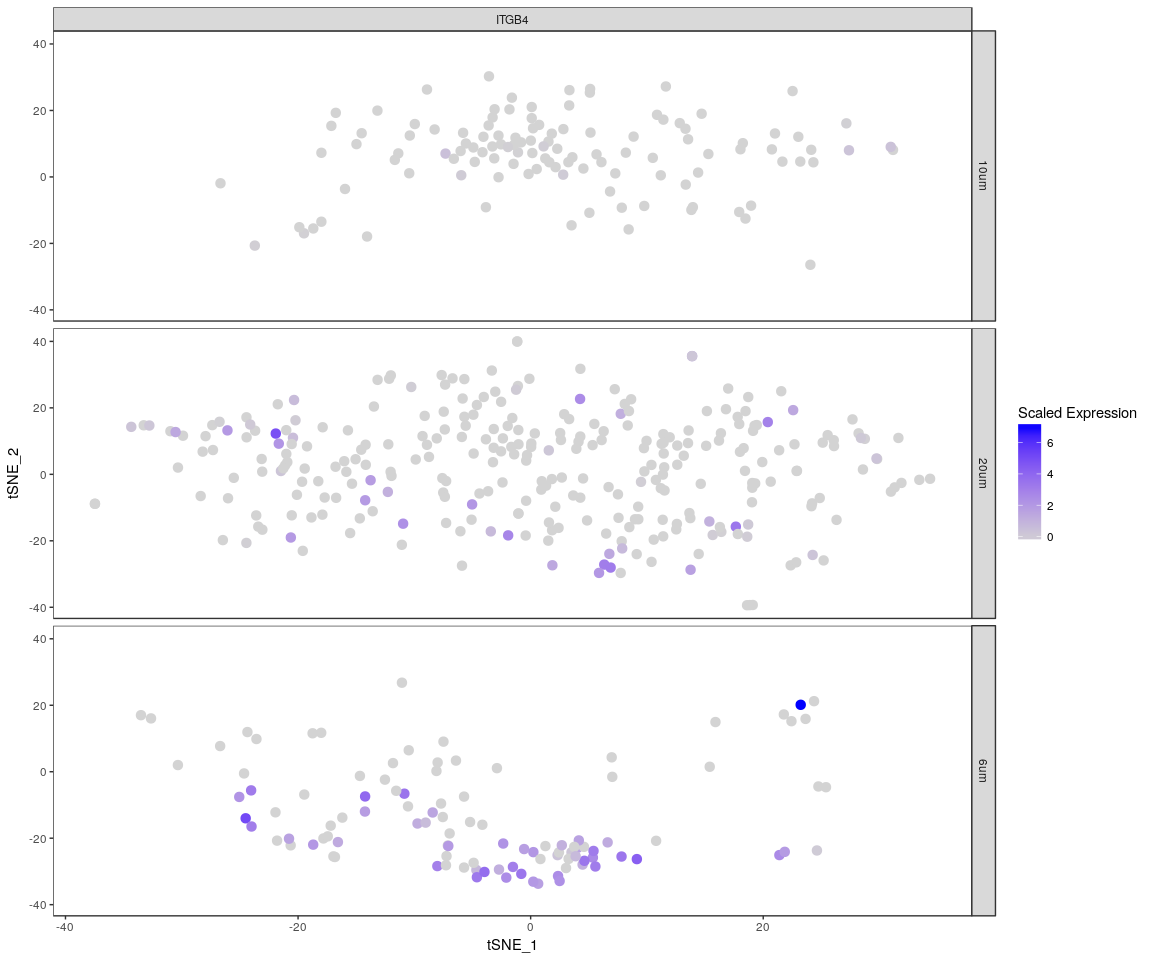
DimPlot(all.pbmc, reduction.use = "pca", pt.size = 4) # grour by sample



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"))

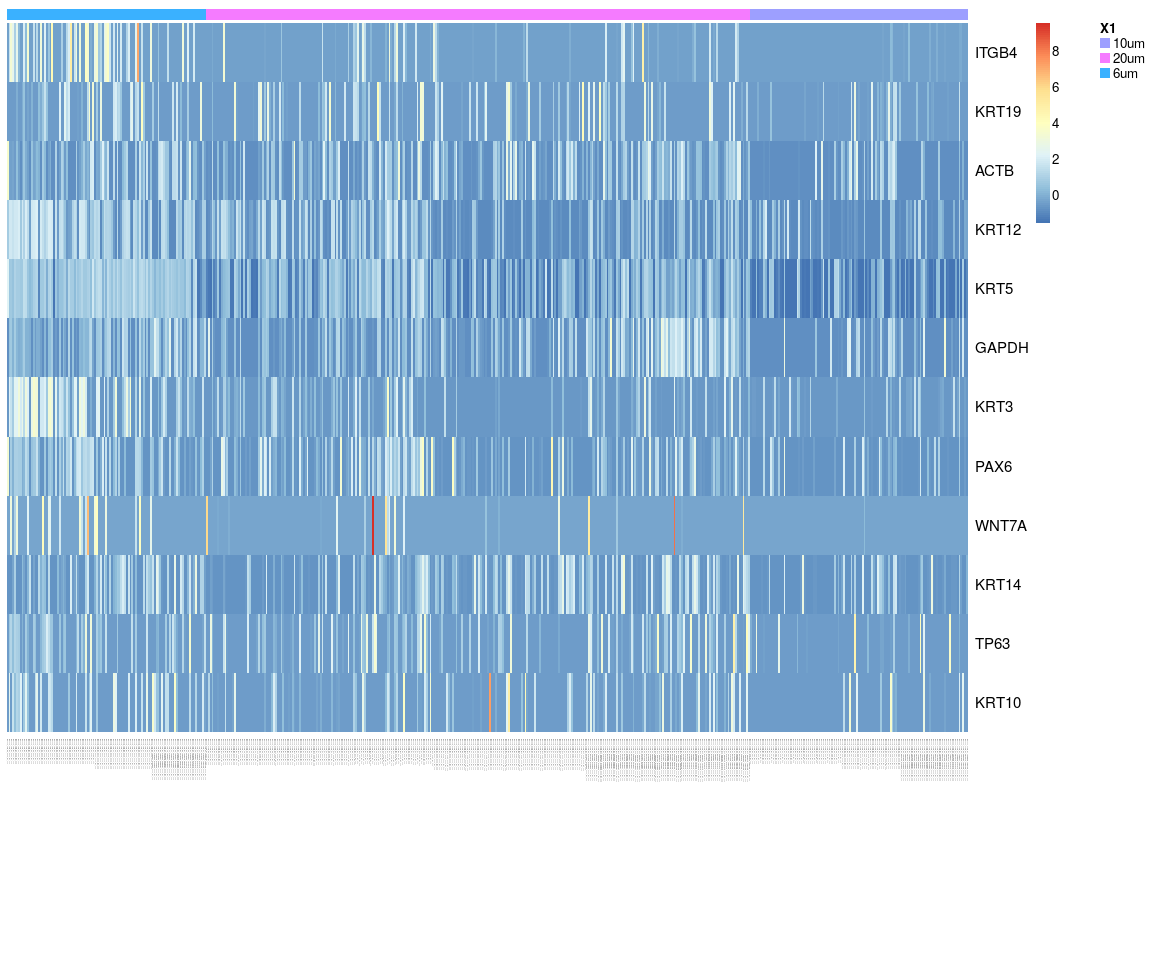


### Heatmap for important genes

monkey.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

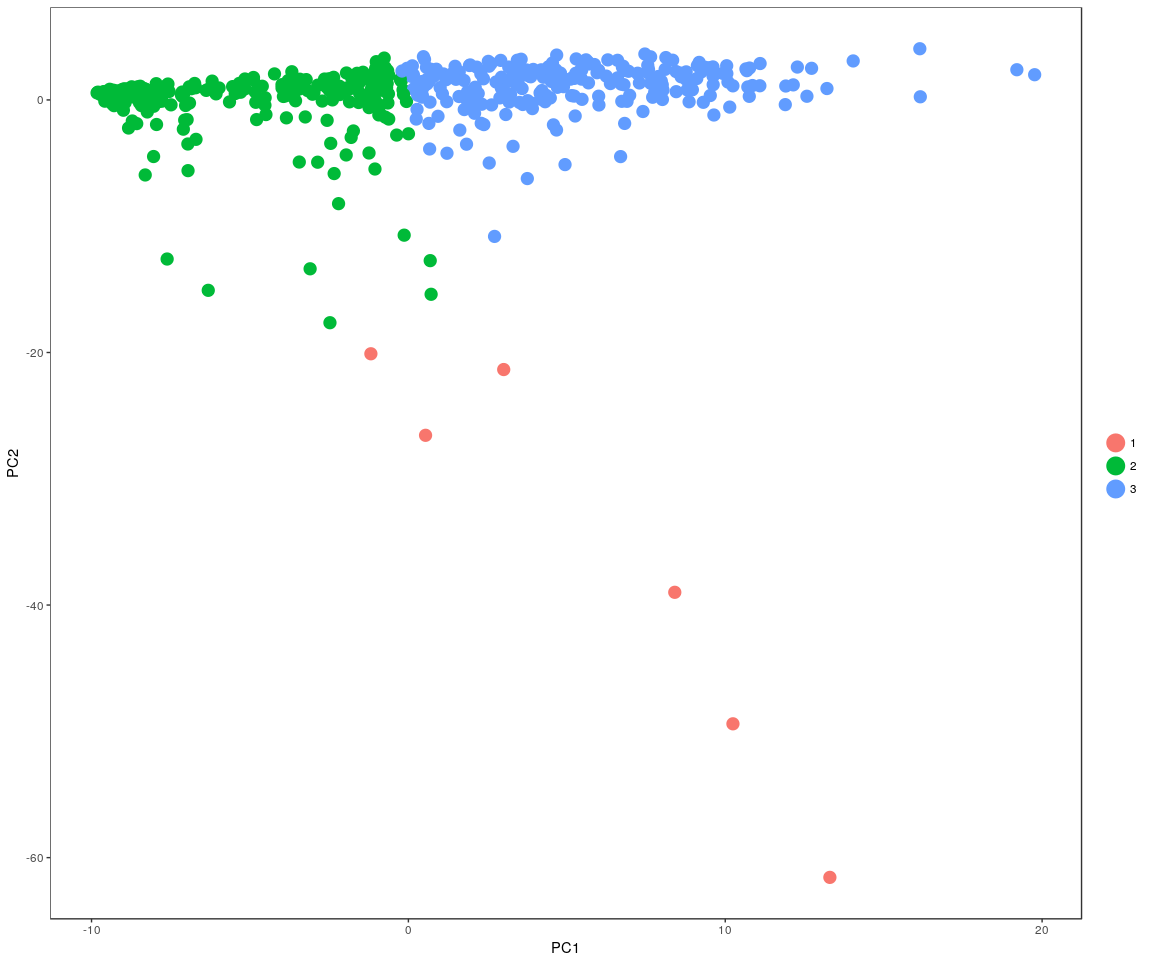
NMF::aheatmap(monkey.heatmap[[2]], Rowv = NA, Colv = NA, annCol = monkey.heatmap[[1]],   
 scale = "none")



The heatmap of genes ITGB4, KRT19, ACTB, KRT12, KRT5, GAPDH, KRT3, PAX6, WNT7A, KRT14, TP63, KRT10 .It tells us that **KRT14,WN&7A,ITGB4** expressed differently across sample,expressed more significant.

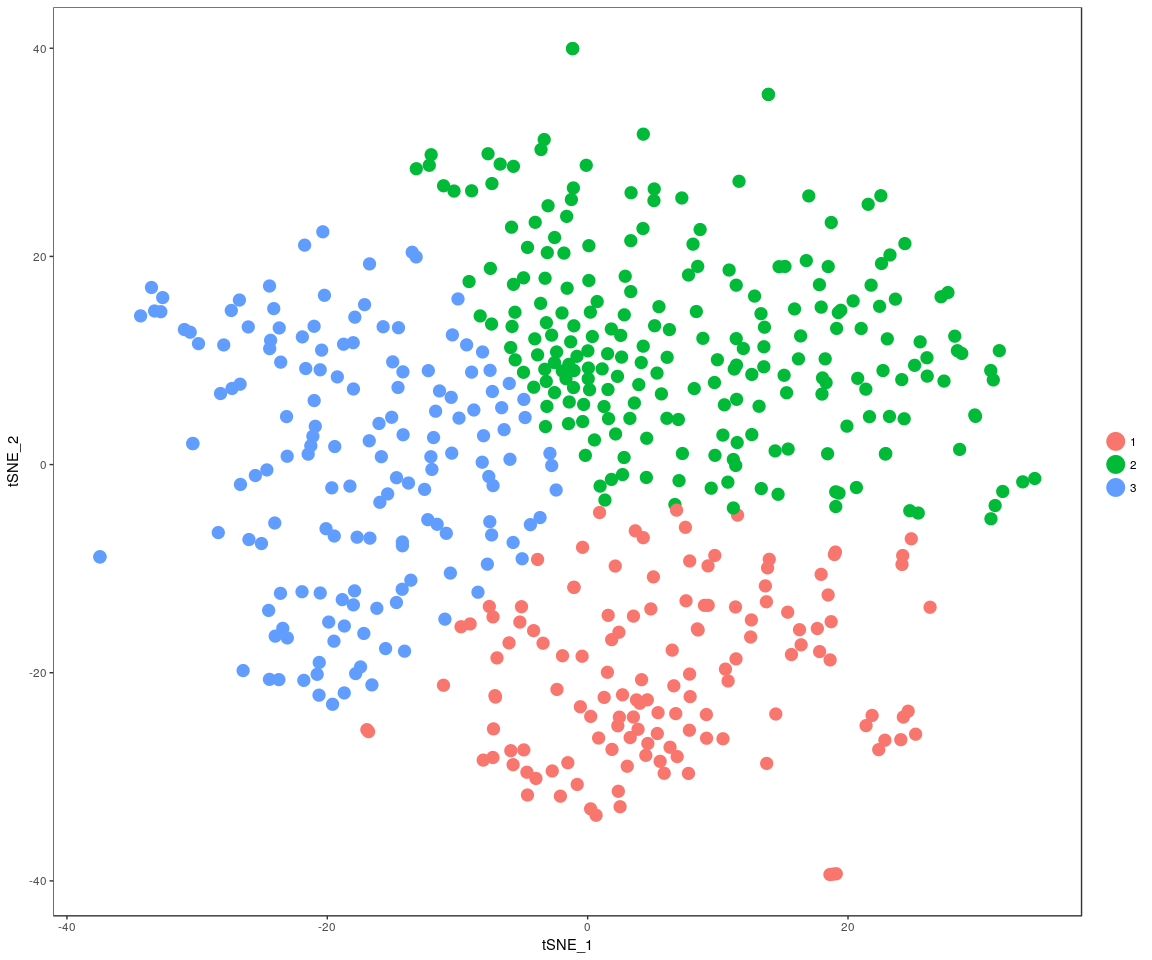
### Next,Spectral k-means clustering on single cells based on PCA

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")



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

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")



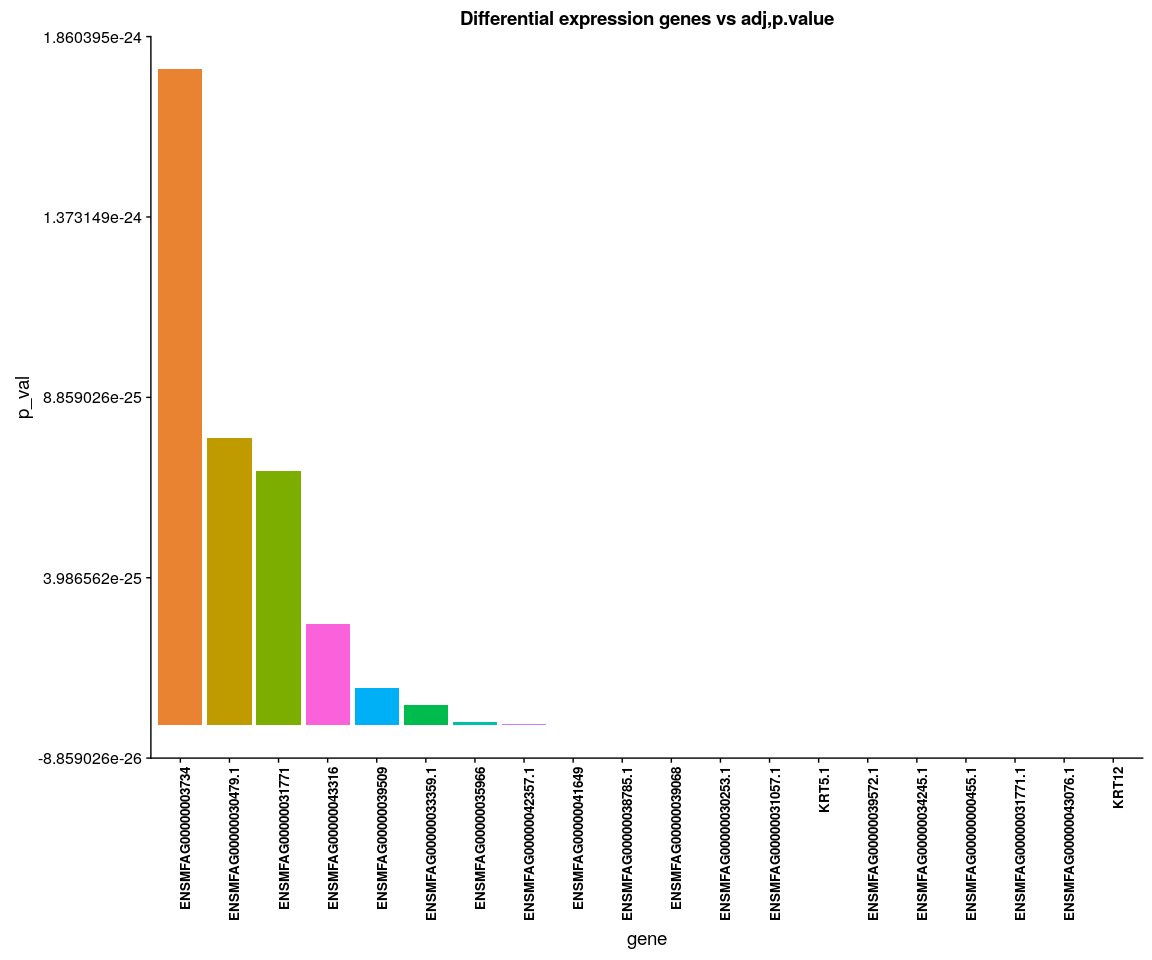
## Differential expression

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  
monkey.markers <- FindAllMarkers(all.pbmc, test.use = "bimod", print.bar = FALSE)  
head(monkey.markers)

## p\_val avg\_logFC pct.1 pct.2 p\_val\_adj cluster  
## KRT12 6.626134e-45 2.180330 0.925 0.759 7.926844e-41 1  
## ENSMFAG00000035966 7.068418e-31 1.403926 0.843 0.398 8.455948e-27 1  
## ENSMFAG00000039509 8.423256e-30 1.171068 0.963 0.838 1.007674e-25 1  
## ENSMFAG00000043316 2.280718e-29 1.119766 0.940 0.796 2.728423e-25 1  
## ENSMFAG00000031771 5.744944e-29 1.098286 0.970 0.733 6.872677e-25 1  
## ENSMFAG00000038785 2.368526e-28 1.308797 0.940 0.665 2.833468e-24 1  
## gene  
## KRT12 KRT12  
## ENSMFAG00000035966 ENSMFAG00000035966  
## ENSMFAG00000039509 ENSMFAG00000039509  
## ENSMFAG00000043316 ENSMFAG00000043316  
## ENSMFAG00000031771 ENSMFAG00000031771  
## ENSMFAG00000038785 ENSMFAG00000038785

### Bar plot of gene’s p.val



### The Kruskal-Wallis Test:Test whether the mean between the cell size or cell sample are equal(more than two sample group,specify for all sample group)

kruskal.test performs a Kruskal-Wallis rank sum test of the null that the location parameters of the distribution of samples are the same in each group

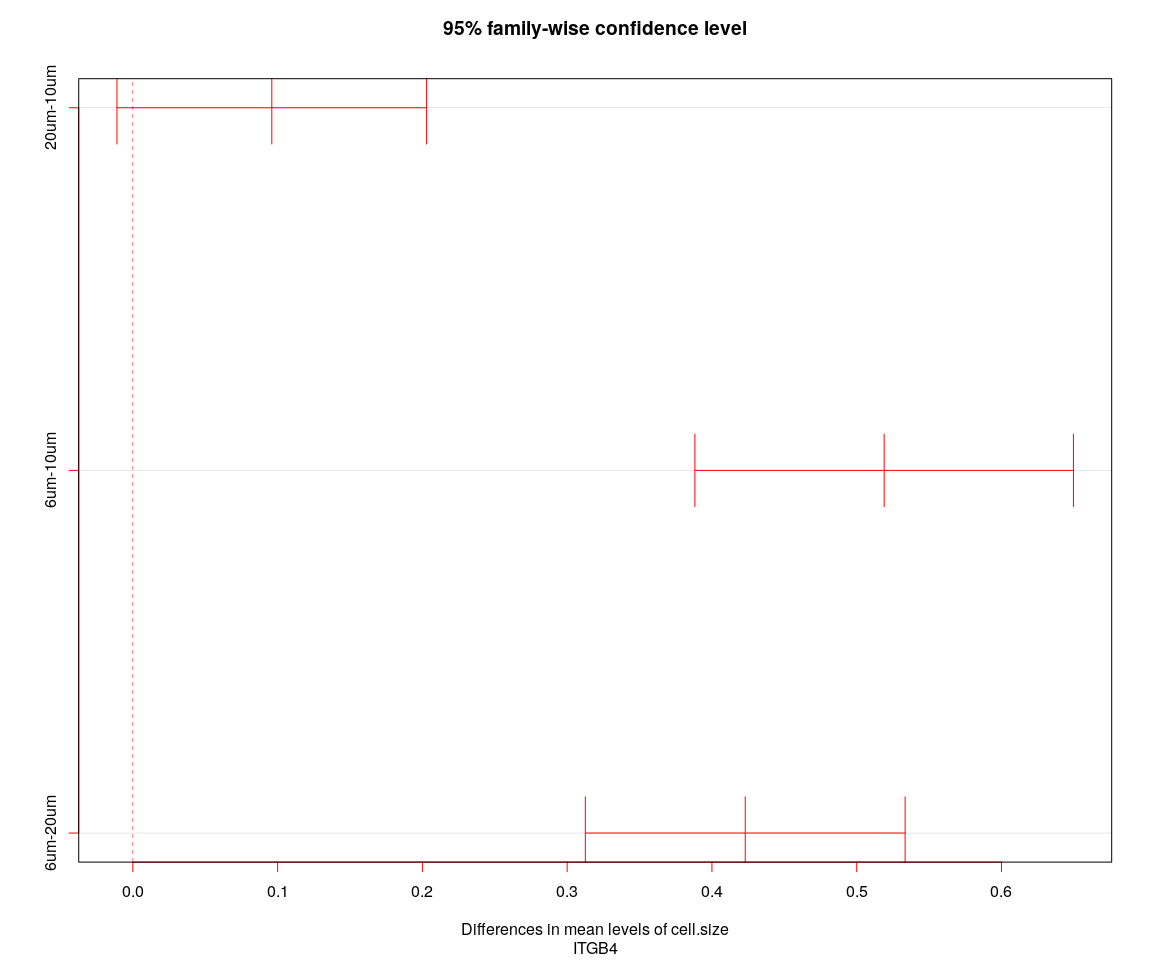
result <- lapply(as.character(unique(monkey.imp.lognorm.melt$variable)), function(x) {  
 KruskalTest.gene(data = monkey.imp.lognorm.melt, gene = x)  
})  
t(data.frame(result))

## chisq p.value  
## ITGB4\_cell.sample 1.423756 6.999758e-01  
## ITGB4\_cell.size 59.642637 1.118835e-13  
## KRT19\_cell.sample 4.770666 1.893812e-01  
## KRT19\_cell.size 7.094489 2.880389e-02  
## ACTB\_cell.sample 9.679937 2.149221e-02  
## ACTB\_cell.size 39.071801 3.278432e-09  
## KRT12\_cell.sample 80.694071 2.178469e-17  
## KRT12\_cell.size 62.301602 2.960596e-14  
## KRT5\_cell.sample 5.724984 1.257845e-01  
## KRT5\_cell.size 118.710388 1.668656e-26  
## GAPDH\_cell.sample 33.307148 2.774281e-07  
## GAPDH\_cell.size 43.081246 4.415972e-10  
## KRT3\_cell.sample 43.963207 1.536588e-09  
## KRT3\_cell.size 57.255298 3.691221e-13  
## PAX6\_cell.sample 24.268454 2.195505e-05  
## PAX6\_cell.size 34.401541 3.386883e-08  
## WNT7A\_cell.sample 7.804619 5.022703e-02  
## WNT7A\_cell.size 21.400517 2.253911e-05  
## KRT14\_cell.sample 33.668052 2.328053e-07  
## KRT14\_cell.size 16.976900 2.058321e-04  
## TP63\_cell.sample 20.718506 1.204405e-04  
## TP63\_cell.size 19.757628 5.124903e-05  
## KRT10\_cell.sample 9.819931 2.016047e-02  
## KRT10\_cell.size 22.676206 1.191034e-05

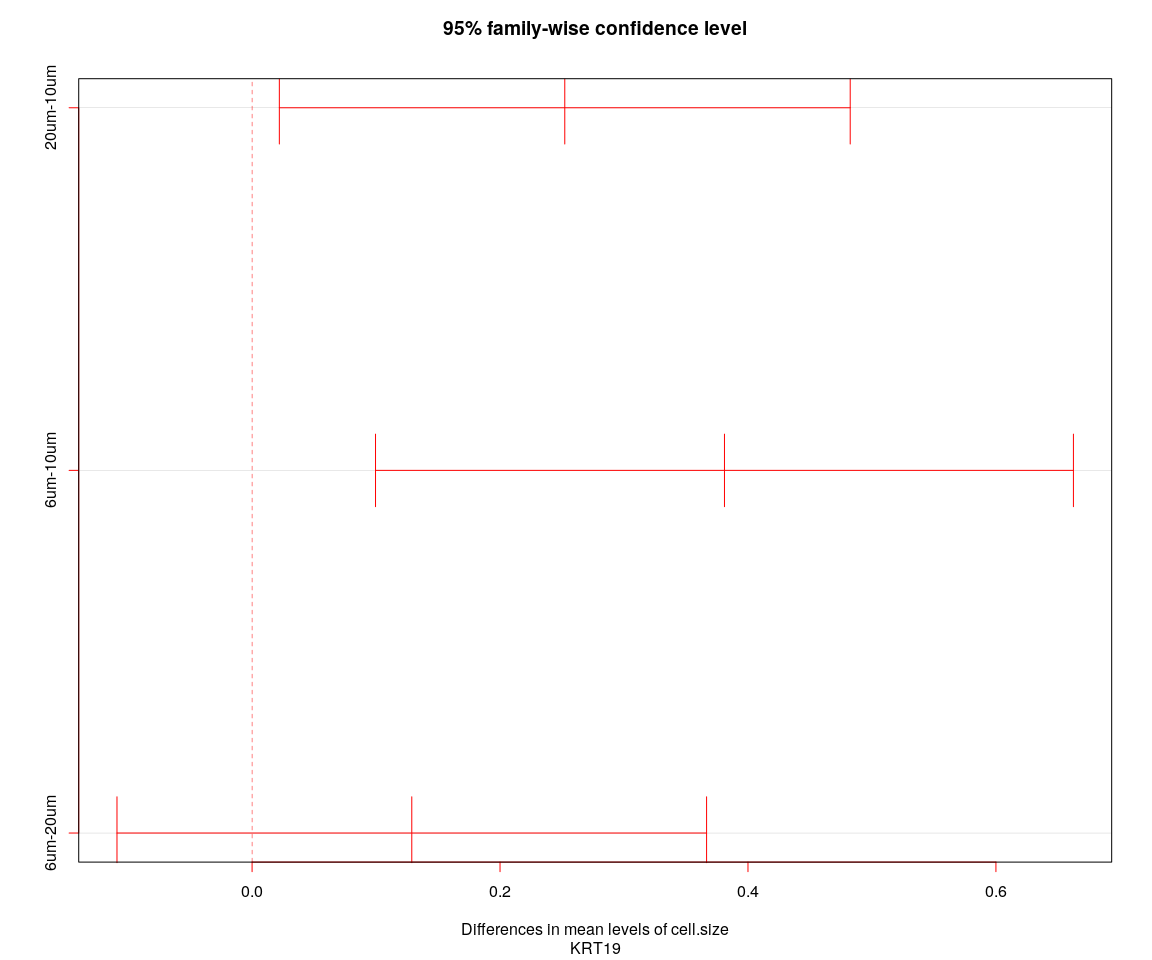
### Anova analsis:test the important genes’mean between cell size,whether is equal(two by two)

#### anavo analysis

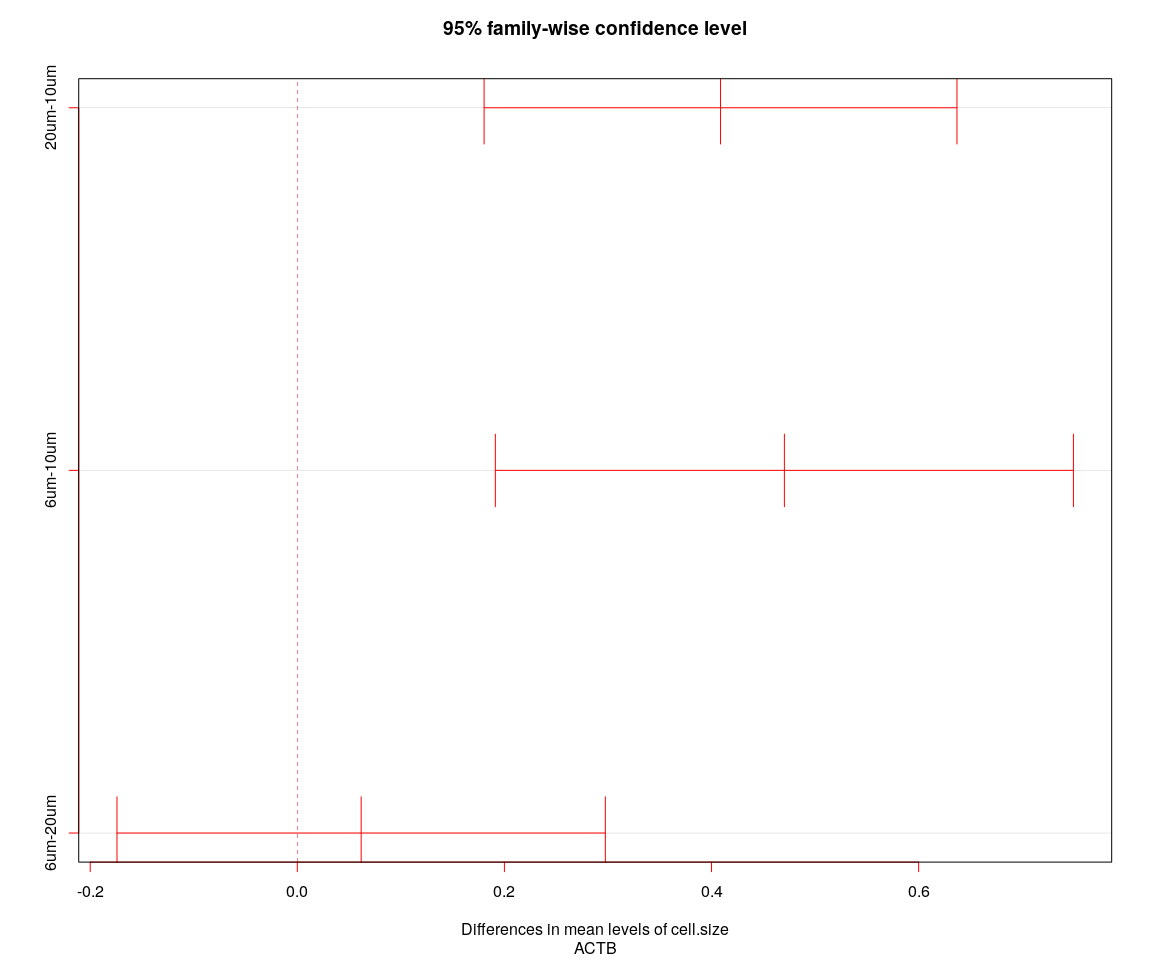
for (gene in unique(monkey.imp.lognorm.melt$variable)) {  
 aov.gene <- Anova.gene(data = monkey.imp.lognorm.melt, gene = gene, tuk.which = "cell.size",   
 inter = FALSE, plot.aov = FALSE)  
}



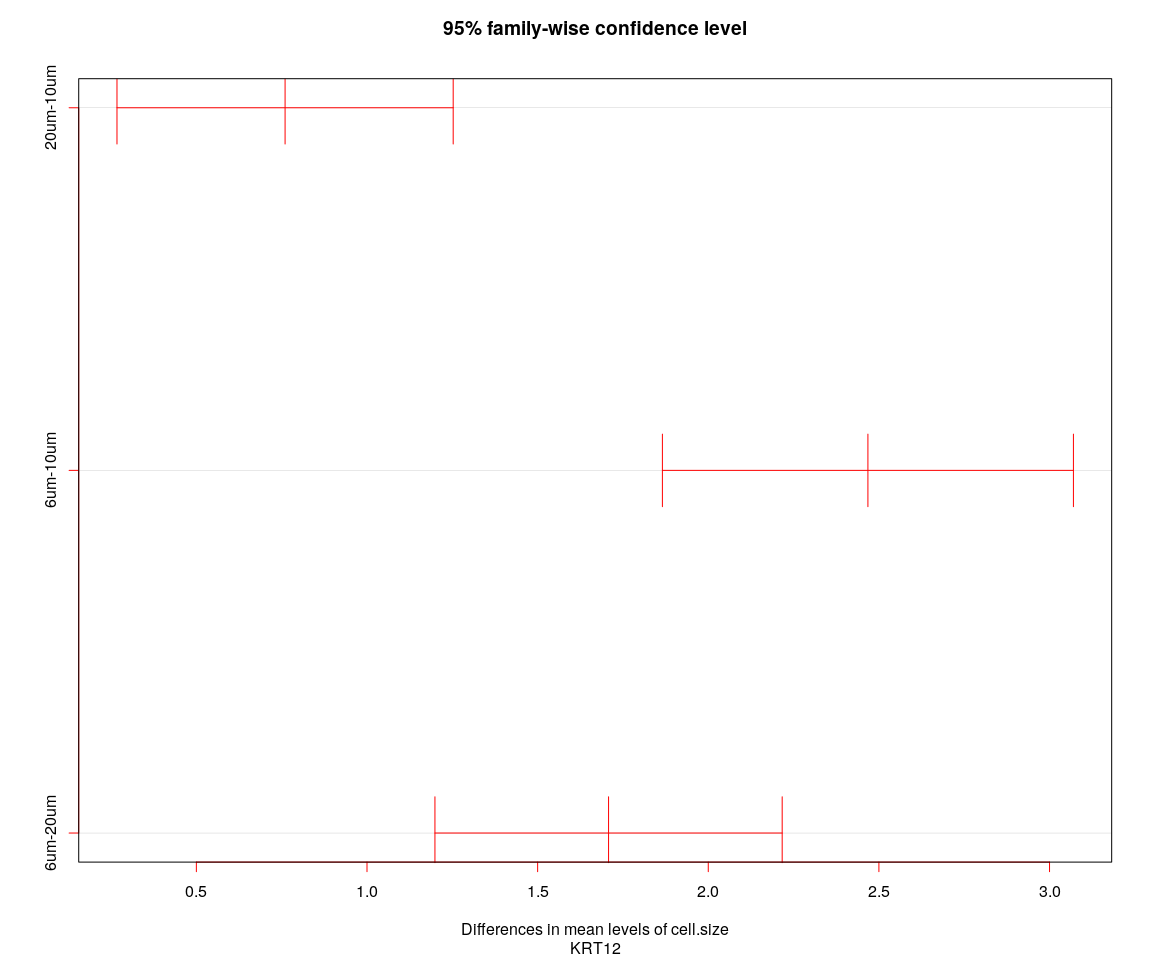
## [1] "----Gene:ITGB4----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.09596351 -0.01097439 0.2029014 0.0889305  
## 6um-10um 0.51896017 0.38822466 0.6496957 0.0000000  
## 6um-20um 0.42299666 0.31254875 0.5334446 0.0000000



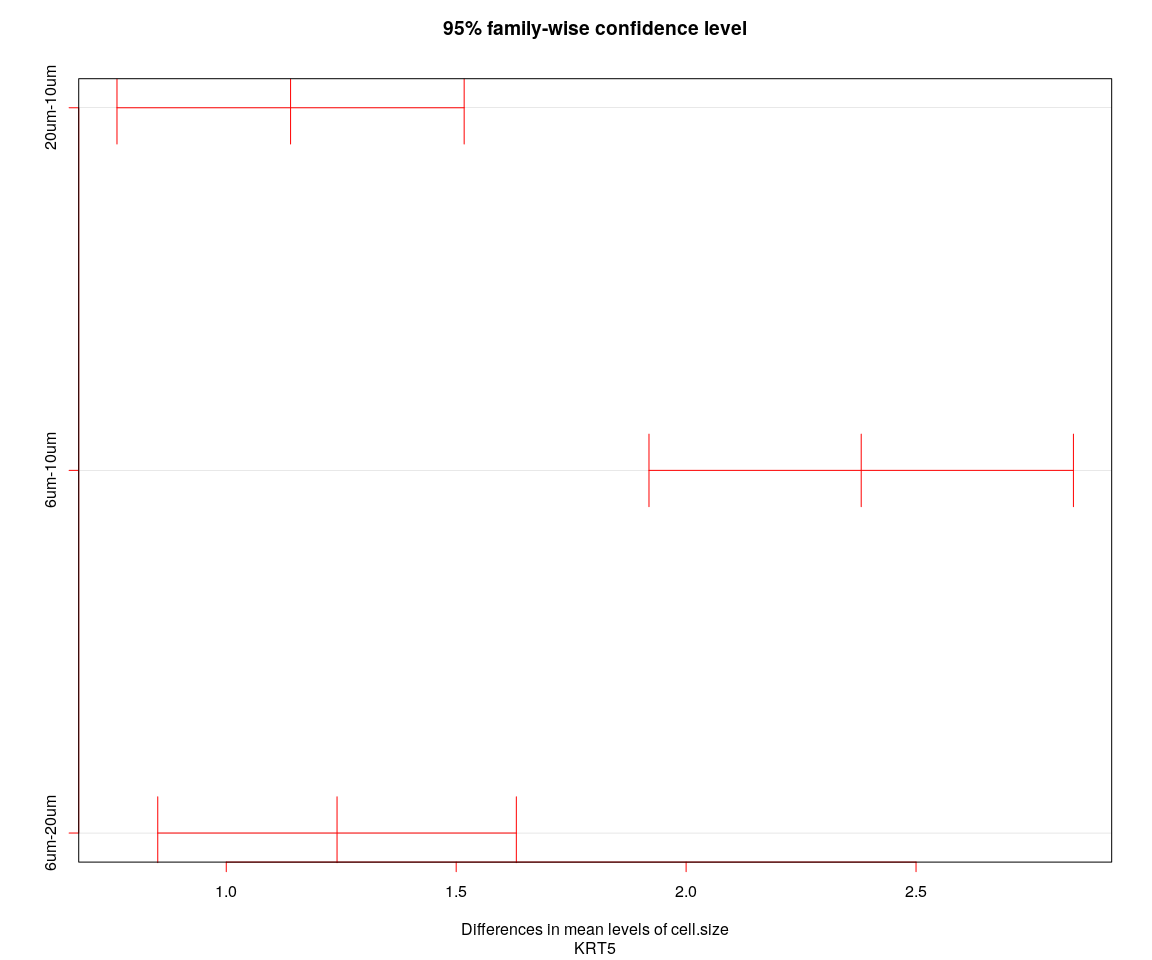
## [1] "----Gene:KRT19----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.2522231 0.02198440 0.4824619 0.0277647  
## 6um-10um 0.3810371 0.09956179 0.6625124 0.0044184  
## 6um-20um 0.1288140 -0.10898187 0.3666098 0.4109353



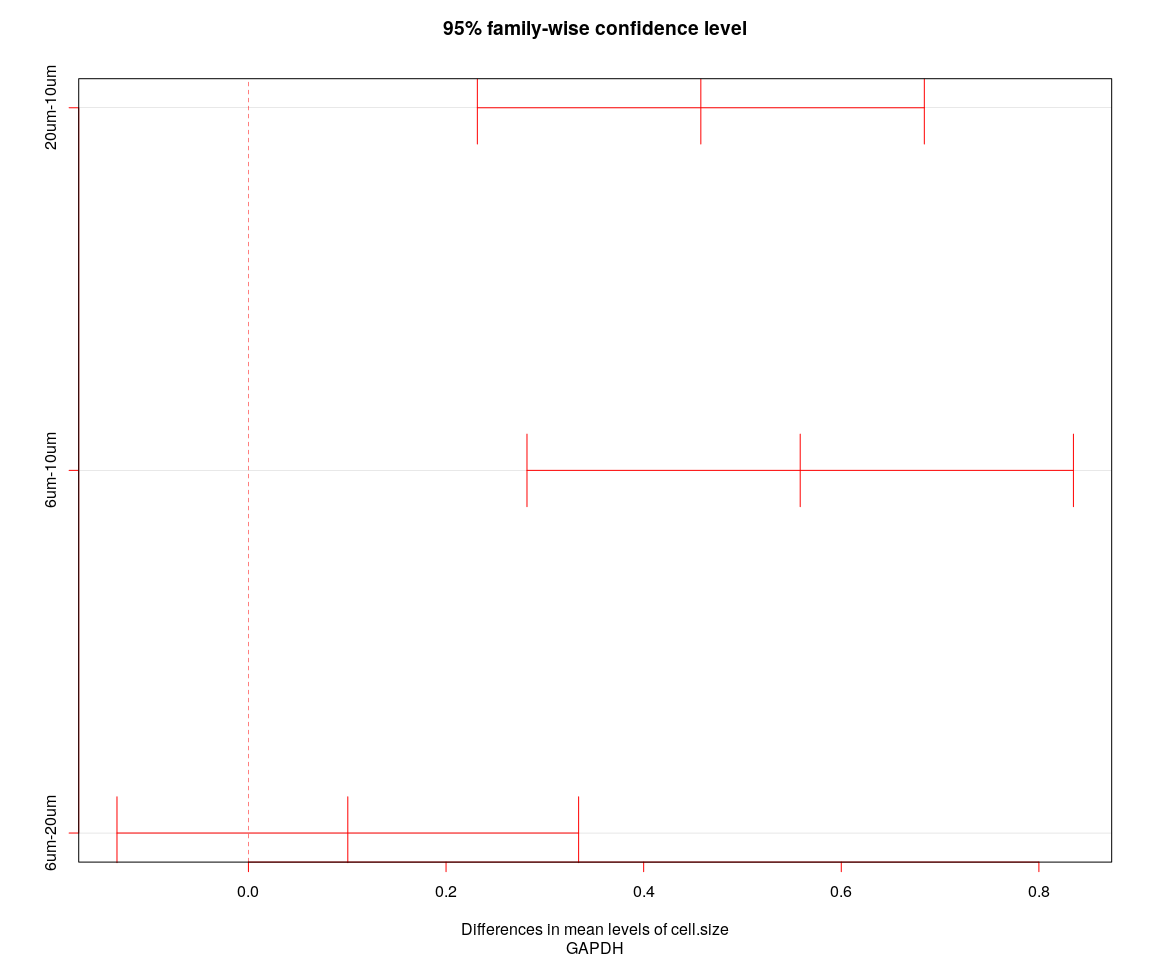
## [1] "----Gene:ACTB----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.40870123 0.1803834 0.6370190 0.0000903  
## 6um-10um 0.47036049 0.1912336 0.7494874 0.0002511  
## 6um-20um 0.06165926 -0.1741526 0.2974711 0.8121925



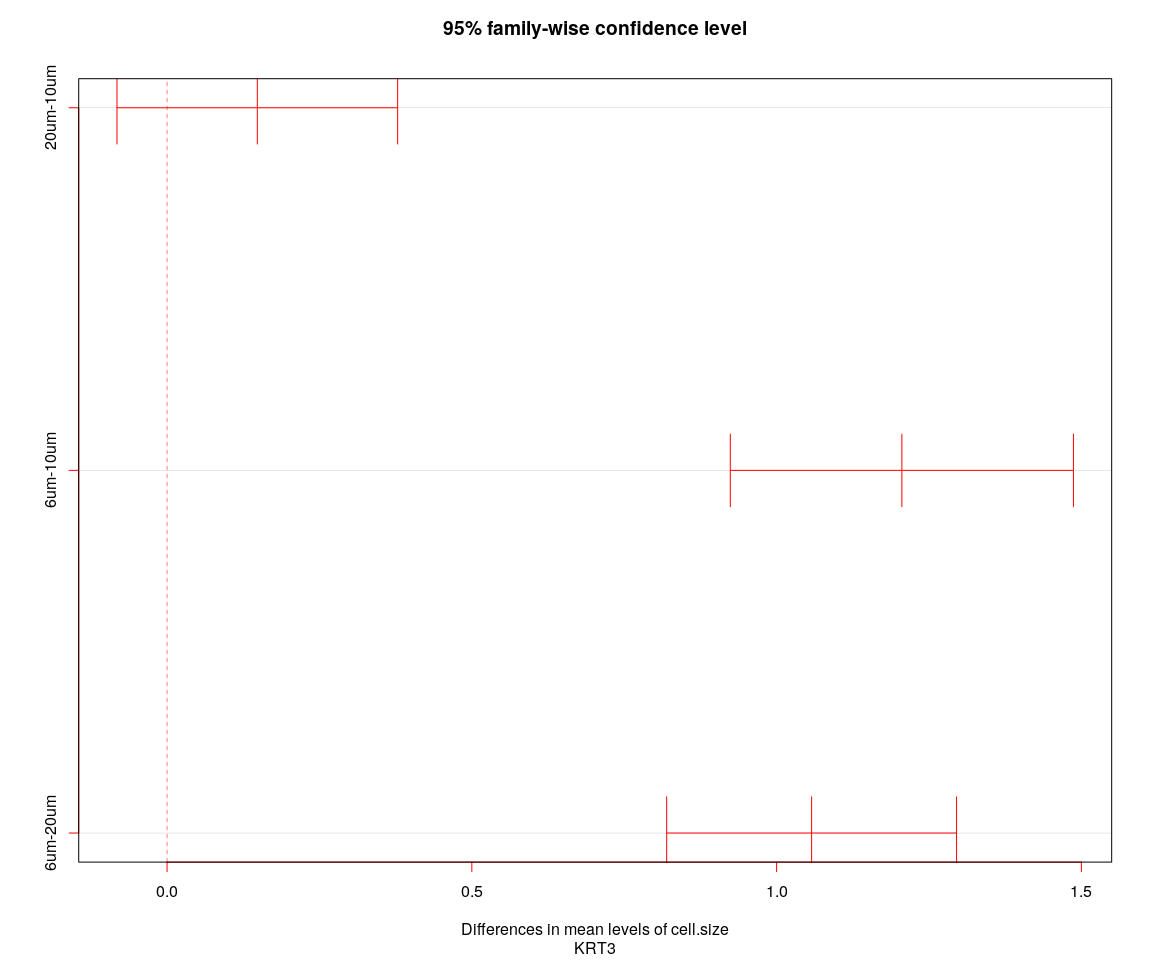
## [1] "----Gene:KRT12----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.7599665 0.2673791 1.252554 0.0009212  
## 6um-10um 2.4677949 1.8655889 3.070001 0.0000000  
## 6um-20um 1.7078285 1.1990730 2.216584 0.0000000



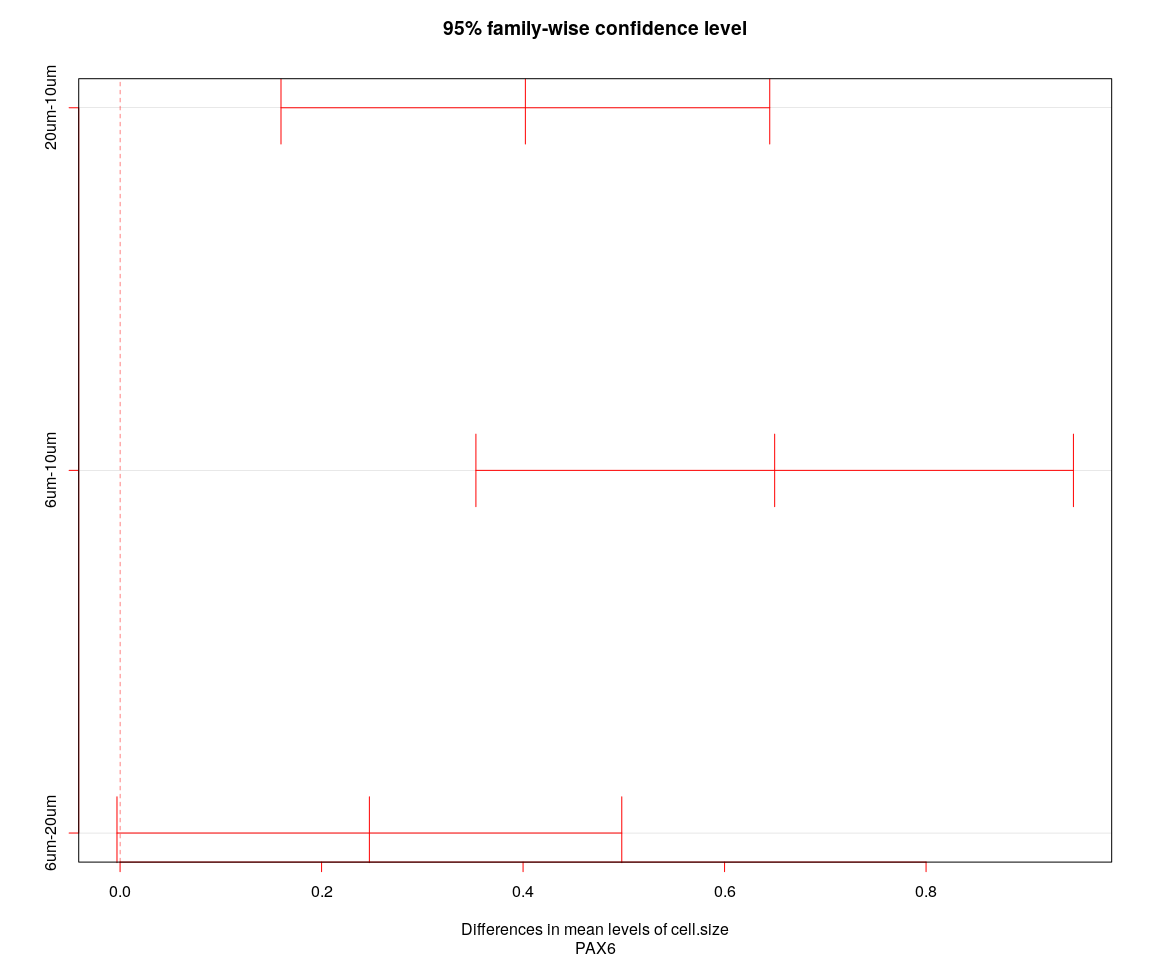
## [1] "----Gene:KRT5----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 1.139845 0.7623508 1.517340 0  
## 6um-10um 2.380757 1.9192563 2.842258 0  
## 6um-20um 1.240912 0.8510268 1.630797 0



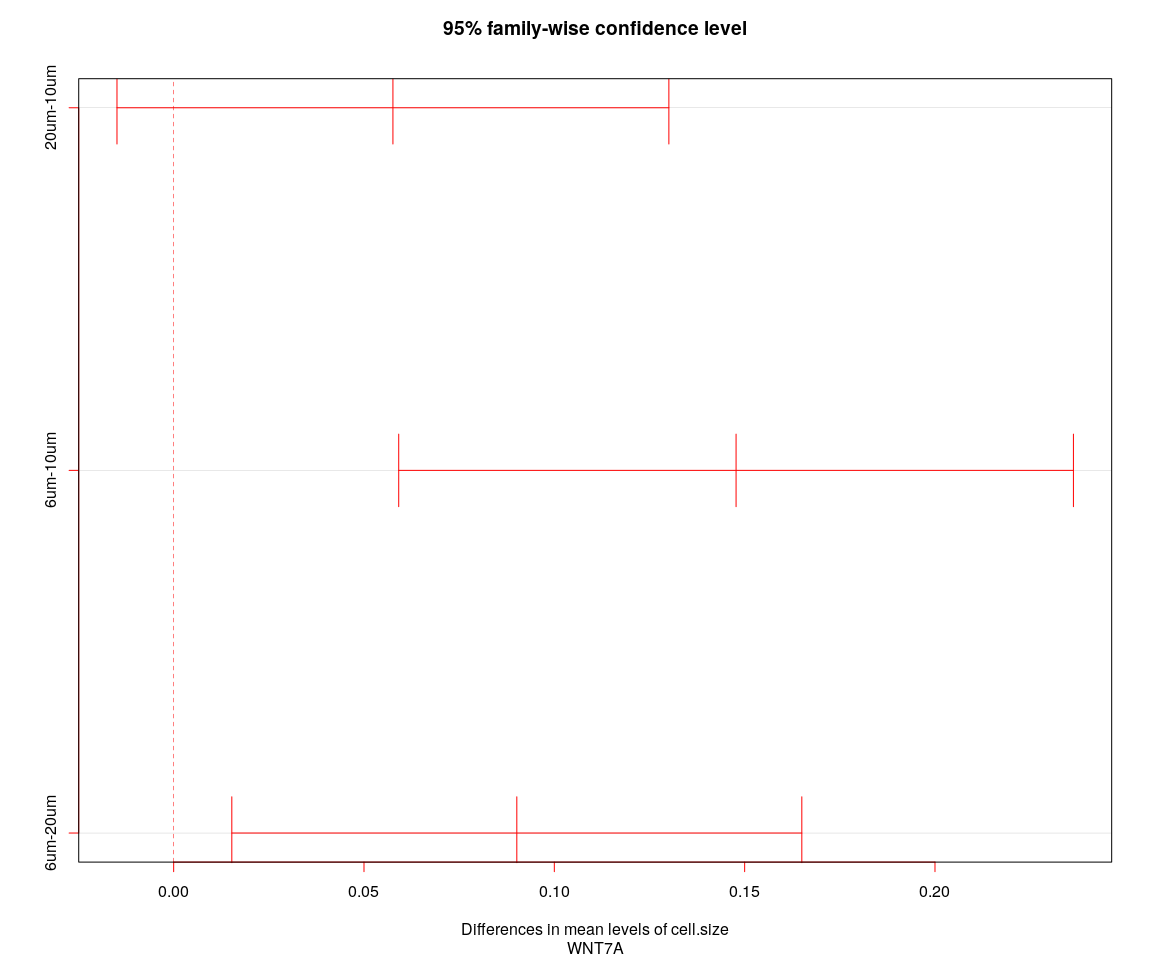
## [1] "----Gene:GAPDH----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.4578612 0.2316786 0.6840438 0.0000076  
## 6um-10um 0.5584298 0.2819133 0.8349463 0.0000080  
## 6um-20um 0.1005686 -0.1330379 0.3341751 0.5696489



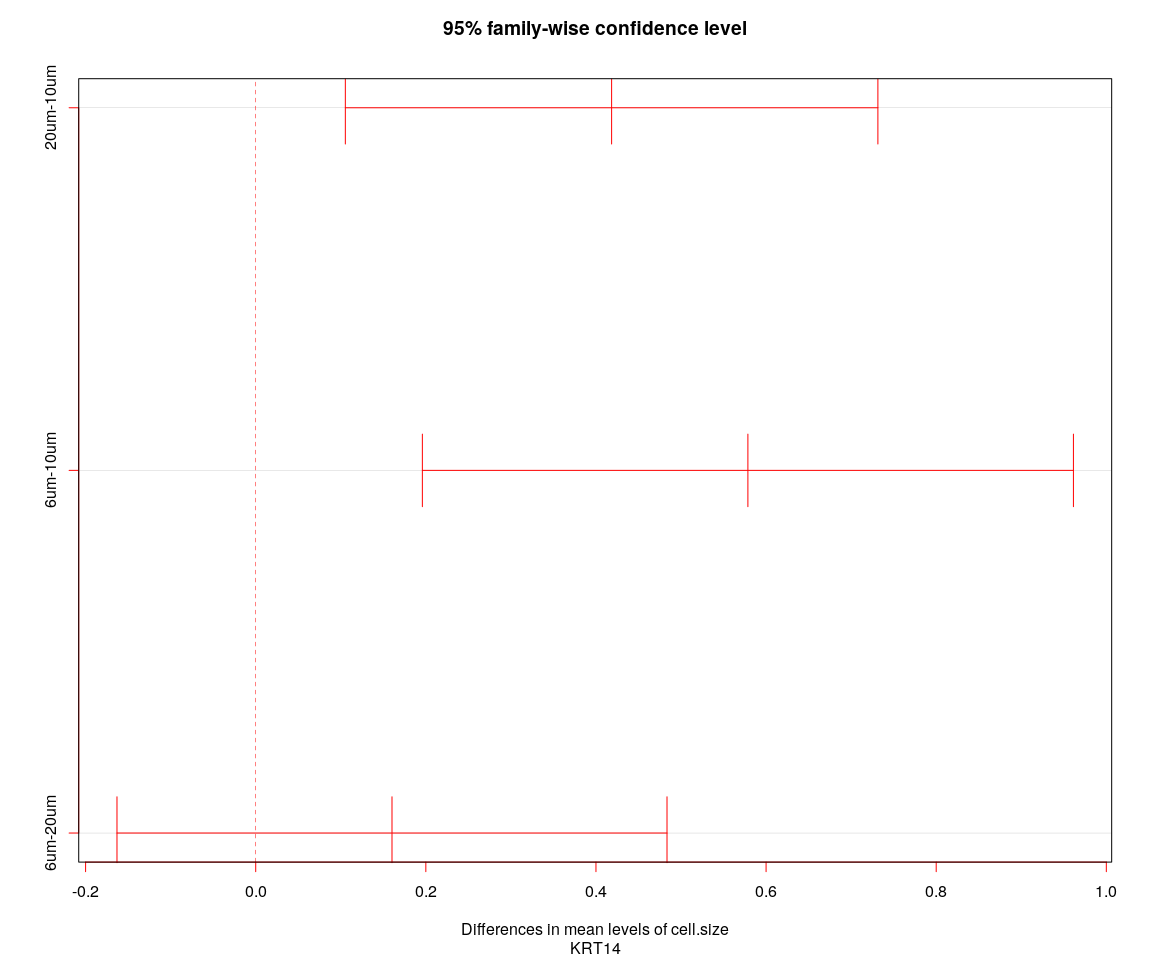
## [1] "----Gene:KRT3----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.1480594 -0.08216785 0.3782867 0.2862556  
## 6um-10um 1.2055059 0.92404463 1.4869672 0.0000000  
## 6um-20um 1.0574465 0.81966248 1.2952305 0.0000000



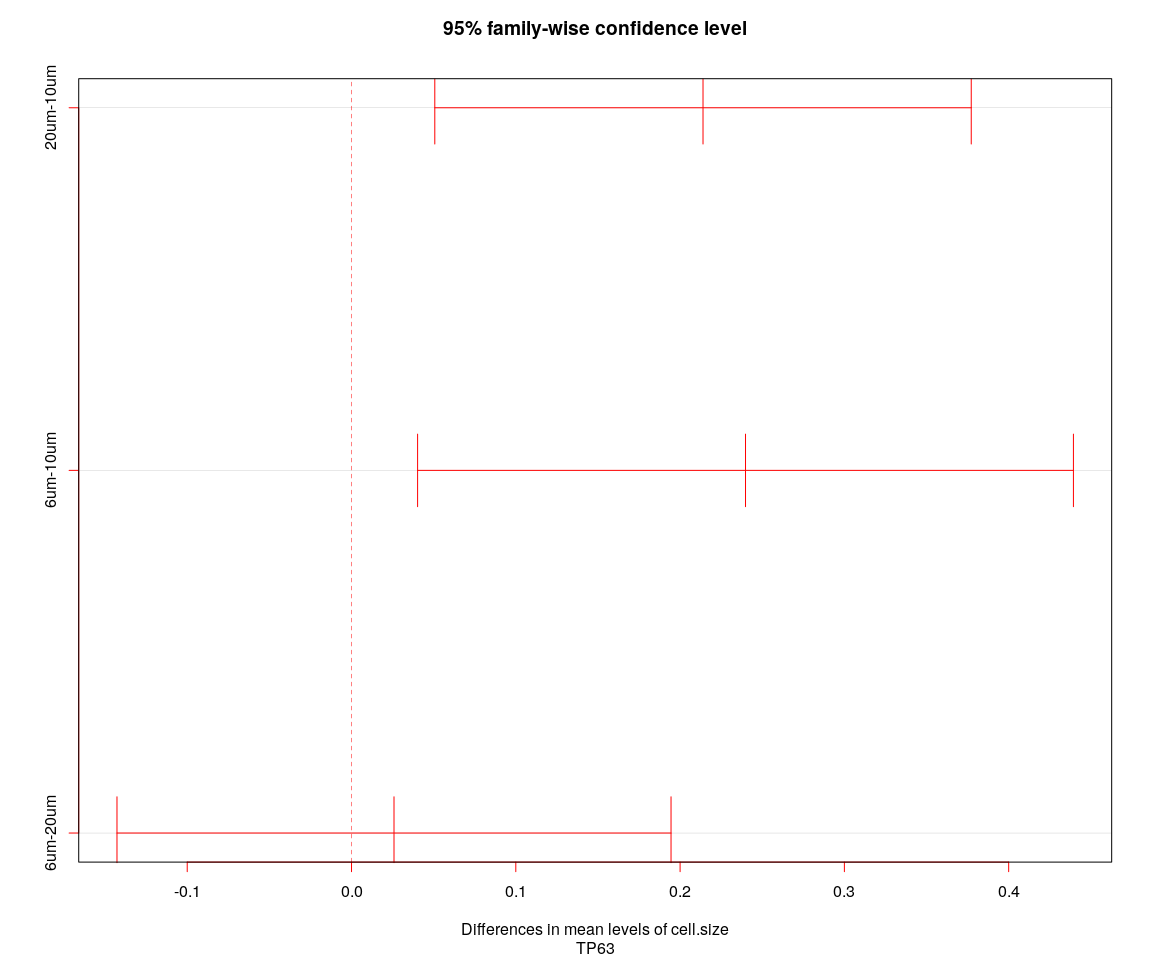
## [1] "----Gene:PAX6----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.4022790 0.159717581 0.6448404 0.0003228  
## 6um-10um 0.6497248 0.353184593 0.9462651 0.0000011  
## 6um-20um 0.2474458 -0.003077163 0.4979688 0.0537337



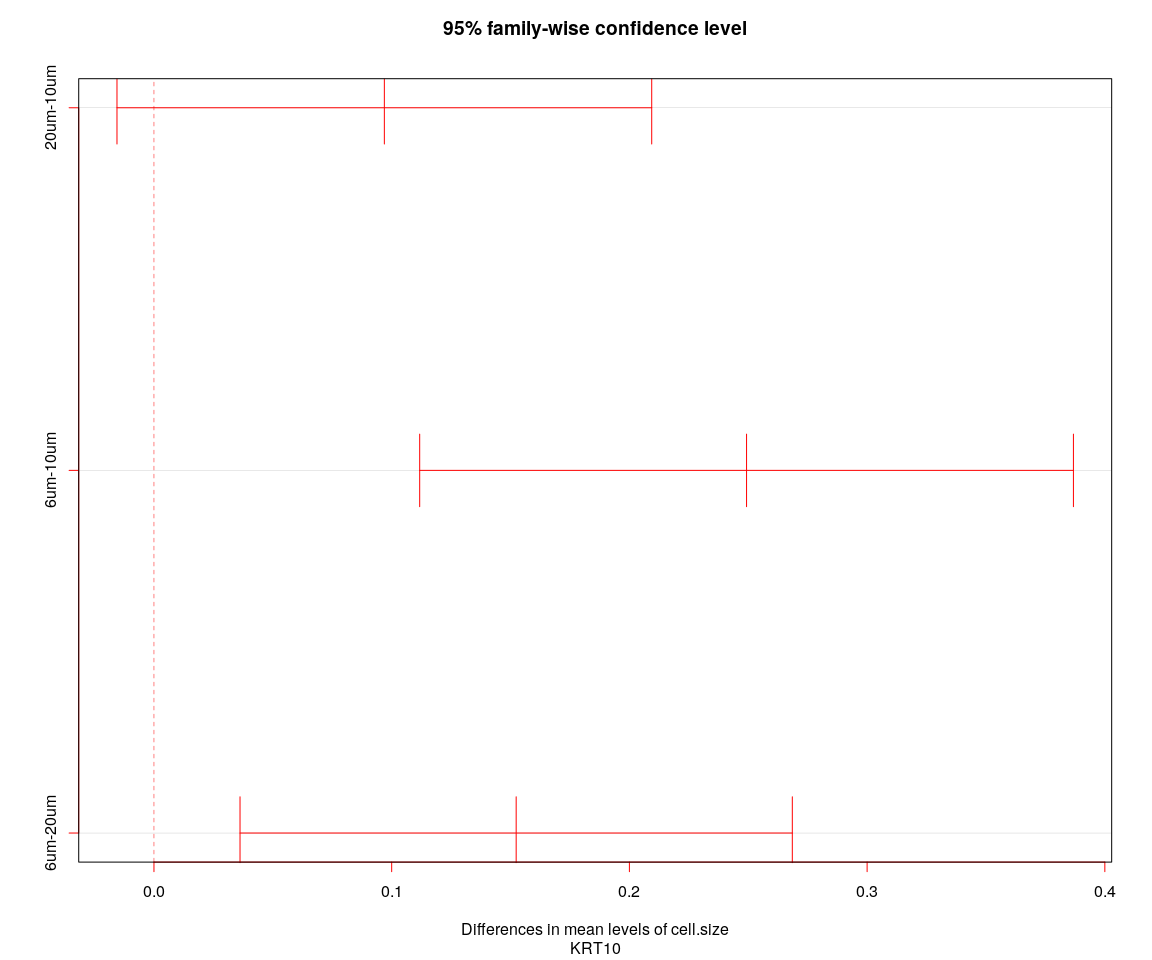
## [1] "----Gene:WNT7A----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.05759956 -0.01489191 0.1300910 0.1491836  
## 6um-10um 0.14774590 0.05912243 0.2363694 0.0002976  
## 6um-20um 0.09014634 0.01527550 0.1650172 0.0133922



## [1] "----Gene:KRT14----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.4184057 0.1053651 0.7314463 0.0050463  
## 6um-10um 0.5786206 0.1959169 0.9613242 0.0012050  
## 6um-20um 0.1602149 -0.1631007 0.4835304 0.4748209



## [1] "----Gene:TP63----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.21395293 0.05069083 0.3772150 0.0061605  
## 6um-10um 0.23981738 0.04022348 0.4394113 0.0136286  
## 6um-20um 0.02586445 -0.14275637 0.1944853 0.9308649



## [1] "----Gene:KRT10----"  
## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = value ~ cell.size + cell.sample, data = genes.data)  
##   
## $cell.size  
## diff lwr upr p adj  
## 20um-10um 0.09691618 -0.01555617 0.2093885 0.1071504  
## 6um-10um 0.24928448 0.11178291 0.3867860 0.0000717  
## 6um-20um 0.15236830 0.03620429 0.2685323 0.0061058