

# Linnarsson Data Analysis

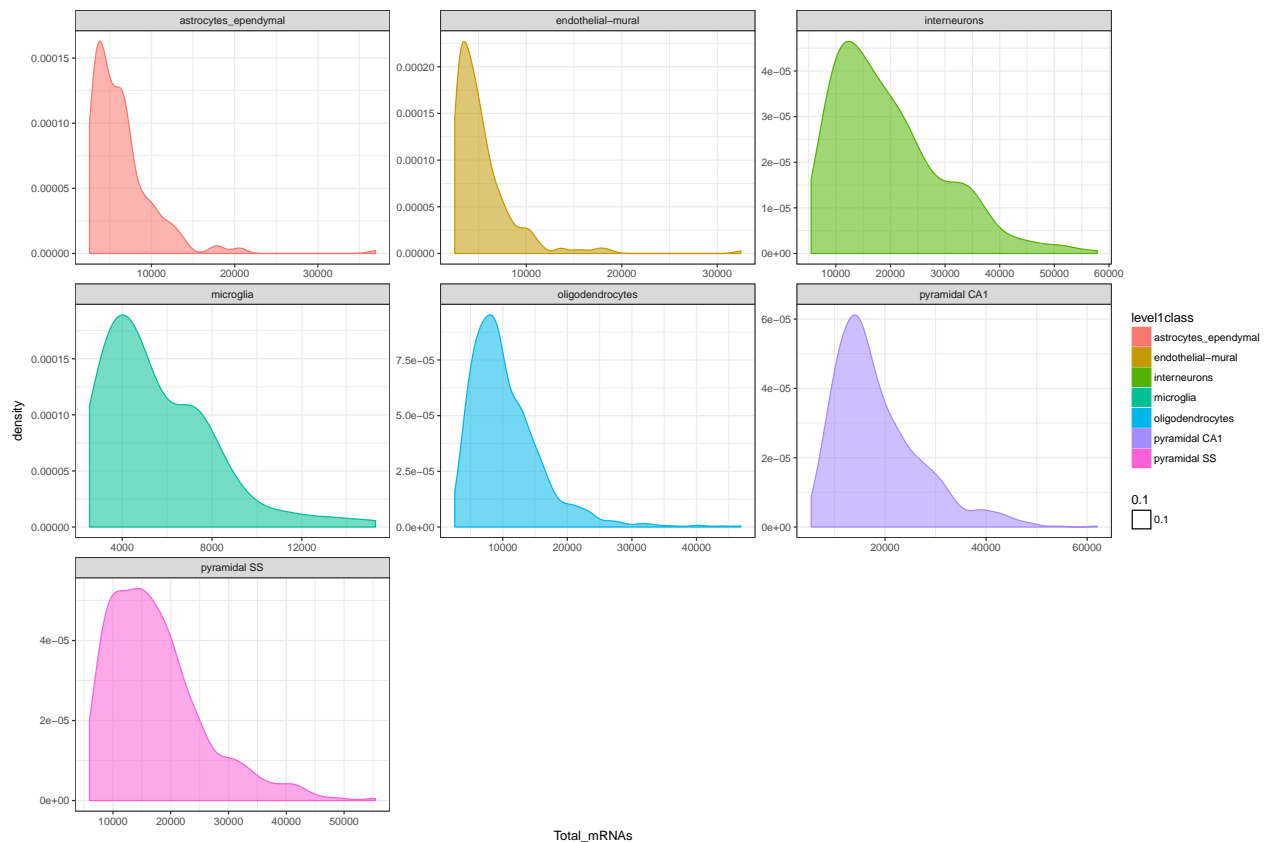
Jonathan Augustin

1/9/2017

Generating the CellDataSet object

```
# QC of data after
pData(dat)$Total_mRNAs <- Matrix::colSums(exprs(dat))

qplot(Total_mRNAs, data = pData(dat), color = level1class, fill = level1class,
      geom = "density", alpha = 0.1) + facet_wrap("level1class", scales = "free") +
      theme_bw()
```



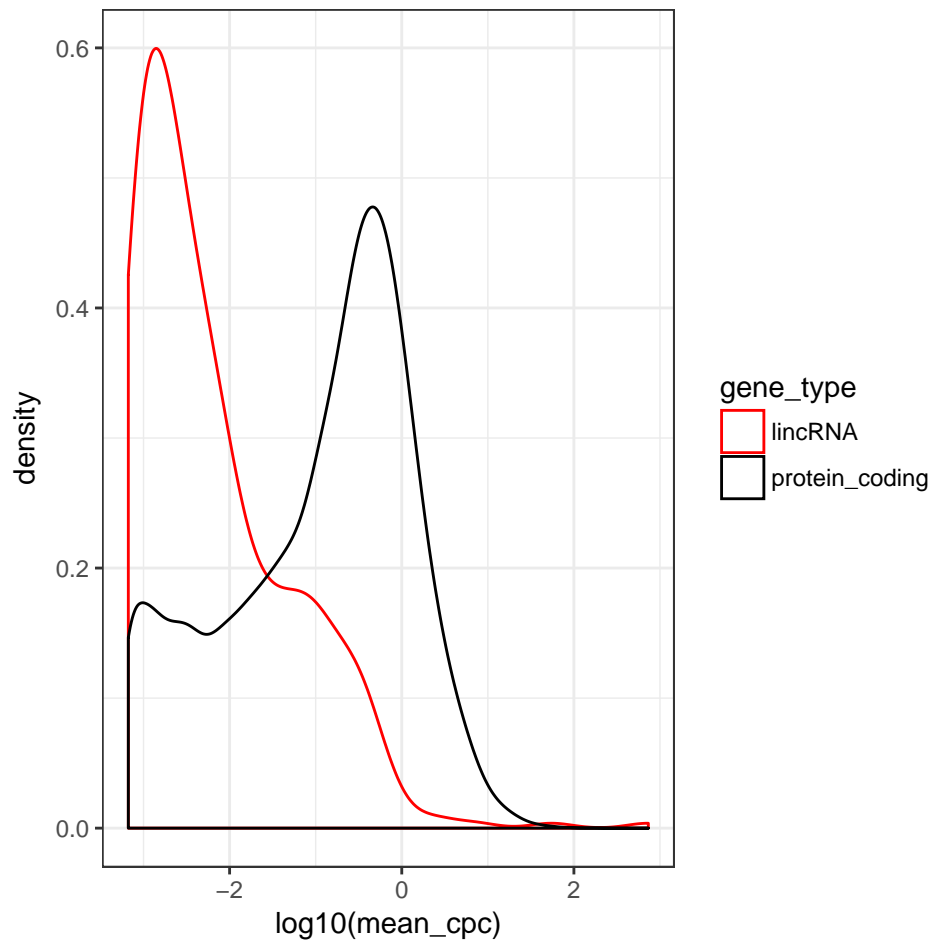
```
# Calculate the mean copies per cell among all classes (Bulk) and draw the
# density plot for lincRNAs vs protein coding
dat.means <- detectGenes(dat, min_expr = 0.001)
dat.means <- dat.means[fData(dat.means)$num_cells_expressed >= 1, pData(dat.means)$num_genes_expressed >= 250]
fData(dat.means)$mean_cpc <- apply(exprs(dat.means), 1, mean)

tmp <- data.frame(gene_short_name = fData(dat.means)$gene_short_name, gene_type = fData(dat.means)$transcript_type,
                 mean_cpc = fData(dat.means)$mean_cpc)

dat_means <- subset(tmp, gene_type %in% c("protein_coding", "lincRNA"))
```

```
density.plot <- ggplot(dat_means) + geom_density(aes(x = log10(mean_cpc), color = gene_type)) +
  scale_color_manual(values = c("red", "black")) + theme_bw()
```

density.plot



```
# List the lincRNAs that are expressed with a mean_cpc greater than 1
dat_lincRNA_sort <- subset(dat_means, gene_type %in% "lincRNA")
dat_mRNA_sort <- subset(dat_means, gene_type %in% "protein_coding")
```

```
# length(dat_lincRNA_sort$gene_short_name)
print("Number of lncRNAs = 441")
```

```
## [1] "Number of lncRNAs = 441"
```

```
# length(dat_mRNA_sort$gene_short_name)
print("Number of mRNAs = 17091")
```

```
## [1] "Number of mRNAs = 17091"
```

Separate the “dat” CellDataSet by “Cluster” and calculate mean expression of genes

```
# Separate the 'dat' CellDataSet by 'Cluster' and calculate mean expression
# of genes
```

```
Cluster.split <- lapply(unique(pData(dat)$group_num), function(x) {
  dat[, pData(dat)$group_num == x]
})
```

```

Cluster.split <- lapply(c(1:length(Cluster.split)), function(x) {
  detectGenes(Cluster.split[[x]], min_expr = 0.01)
})

Cluster.split <- lapply(c(1:length(Cluster.split)), function(i) {
  x <- Cluster.split[[i]]
  x[fData(x)$num_cells_expressed > 1]
})

Cluster.split <- lapply(Cluster.split, function(x) {
  mean_cpc <- apply(exprs(x), 1, mean)
  fData(x)$mean_cpc <- mean_cpc
  return(x)
})

tmp <- data.frame()
group_means <- lapply(c(1:length(Cluster.split)), function(i) {
  x <- Cluster.split[[i]]
  res <- data.frame(gene_short_name = fData(x)$gene_short_name, gene_type = fData(x)$transcript_type,
    mean_cpc = fData(x)$mean_cpc, group_num = unique(pData(x)$group_num))
  tmp <- rbind(tmp, res)
})

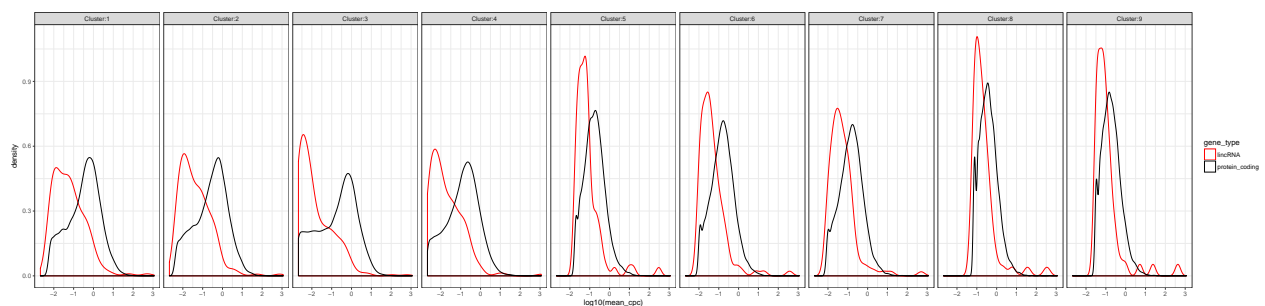
tmp <- plyr::ldply(group_means, data.frame)

group_means <- subset(tmp, gene_type %in% c("protein_coding", "lincRNA"))

density.plot_Cluster <- ggplot(group_means) + geom_density(aes(x = log10(mean_cpc),
  color = gene_type)) + facet_grid(. ~ group_num, labeller = labeller(group_num = function(x) {
    paste("Cluster", x, sep = ":")
  }))) + scale_color_manual(values = c("red", "black")) + theme_bw()

density.plot_Cluster

```



Separate the “dat” CellDataSet by level2class and calculate mean expression of genes

```

# Separate the 'dat' CellDataSet by level2class and calculate mean
# expression of genes
level2.split <- lapply(unique(pData(dat)[pData(dat)$level1class == "interneurons",
  ]$level2class), function(x) {
  dat[, pData(dat)$level2class == x]
})

level2.split <- lapply(c(1:length(level2.split)), function(x) {

```

```

    detectGenes(level2.split[[x]], min_expr = 0.01)
  })

level1.split <- lapply(c(1:length(level2.split)), function(i) {
  x <- level2.split[[i]]
  x[fData(x)$num_cells_expressed > 1]
})

level2.split <- lapply(level2.split, function(x) {
  mean_cpc <- apply(exprs(x), 1, mean)
  fData(x)$mean_cpc <- mean_cpc
  return(x)
})

tmp <- data.frame()

group_means_level2class <- lapply(c(1:length(level2.split)), function(i) {
  x <- level2.split[[i]]
  res <- data.frame(gene_short_name = fData(x)$gene_short_name, gene_type = fData(x)$transcript_type,
    mean_cpc = fData(x)$mean_cpc, level2class = unique(pData(x)$level2class))
  tmp <- rbind(tmp, res)
})

tmp <- plyr::ldply(group_means_level2class, data.frame)

group_means_level2class <- subset(tmp, gene_type %in% c("protein_coding", "lincRNA"))

density.plot_level2class <- ggplot(group_means_level2class) + geom_density(aes(x = log10(mean_cpc),
  color = gene_type)) + facet_wrap(~level2class, nrow = 2) + scale_color_manual(values = c("red",
  "black")) + theme_bw()

density.plot_level2class

```

## Warning: Removed 107012 rows containing non-finite values (stat\_density).

