RNA-seq analysis: SLBP (EDA)

Sudarshan Chari

This R Markdown document contains a walk-through for performing exploratory data analysis of pooled (single-end) RNA-seq on single *D.melanogaster* histone overexpression mutant embryos caused by a knock-down of *Slbp* (Stem loop binding protein) gene and corresponding wild-type/ control. Raw transcript count data obtained by mapping to a reference transcriptome using Salmon (slurm/bash scripts calling the relevant functions to preprocess and map are in another repository).

This document includes
- Importing and Normalizing the RNA-seq count data via DESeq2 and rlog transformation
- Visualizing sample similarity based on Euclidean distance and Principal components analysis (PCA)
- custom plotting/ visualization in base R,ggplot2 and pheatmap

"Desktop/RNAseq_output/" has been used as a local directory for this project, but that can be changed to any desired local or cloud based directory.

```
# Libraries for RNA-seq count data analysis
library(tximport)
library(DESeq2)

# Libraries for data handling and visualization
library(ggplot2)
library(dplyr)
library(pheatmap)
library(RColorBrewer)
library(genefilter)
library(dendsort)
library(reshape2)
library(tidyr)

#Libraries for database
library(TxDb.Dmelanogaster.UCSC.dm6.ensGene)
library(org.Dm.eg.db)
```

Loading Gene annotation datasets transcript to geneID to gene name mappings. This can also be obtained from an organism specific database (in this case http://flybase.org/) or can be imported from org.db (in this case org,Dm.eg.db obtained from Bioconductor)

```
setwd("~/Desktop/RNAseq_output/Abo_Wt_StagedNC_flybase_transcriptome/")
ttg.slbp <- read.table("scripts_samples_ttg/flybase_transcript_to_gene.txt", h=T)
head(ttg.slbp)
```

```
##          txId                 Gene
## 1 FBtr0081624 dmel_7SLRNA:CR32864
## 2 FBtr0100521              dmel_a
## 3 FBtr0071764              dmel_a
## 4 FBtr0342981              dmel_a
## 5 FBtr0071763              dmel_a
## 6 FBtr0083388          dmel_abd-A
```

```
fbgn.gene.conv <- read.table("scripts_samples_ttg/flybase_fbgn_to_gene.tsv", h=T)
head(fbgn.gene.conv)
```

```
##        geneId         Gene
## 1 FBgn0262029       dmel_d
## 2 FBgn0052532 dmel_CG32532
## 3 FBgn0023536  dmel_CG3156
## 4 FBgn0029718  dmel_mRpL30
## 5 FBgn0031101  dmel_CG1631
## 6 FBgn0030952 dmel_CG12609
```

Exploratory Analysis

```
setwd("~/Desktop/RNAseq_output/Abo_Wt_StagedNC_flybase_transcriptome/data/SLBP/main/")
# Reading in the sample information file
sample.slbp.all <- read.csv("salmon_data_slbp_main_pilot_inc/samples_slbp_wt_main_pilot_inc.csv",h=T)
str(sample.slbp.all)
```

```
## 'data.frame':    28 obs. of  7 variables:
##  $ Sample       : Factor w/ 28 levels "SLBP_NC0_Pre9_R1_B2",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ Genotype     : Factor w/ 2 levels "SLBP","WT": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Cycle        : Factor w/ 5 levels "NC0_Pre9","NC13",..: 1 1 1 1 1 2 2 2 2 2 ...
##  $ Replicate    : Factor w/ 6 levels "R1","R2","R3",..: 1 2 3 4 5 1 1 2 2 3 ...
##  $ Gen_Rep      : Factor w/ 25 levels "SLBP_NC0_Pre9_R1",..: 1 2 3 4 5 6 6 7 7 8 ...
##  $ Gen_Rep_Batch: Factor w/ 28 levels "SLBP_NC0_Pre9_R1_B2",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ Gen_Cyc      : Factor w/ 5 levels "SLBP_NC0_Pre9",..: 1 1 1 1 1 2 2 2 2 2 ...
```

```
head(sample.slbp.all)
```

```
##                Sample Genotype    Cycle Replicate          Gen_Rep
## 1 SLBP_NC0_Pre9_R1_B2     SLBP NC0_Pre9        R1 SLBP_NC0_Pre9_R1
## 2 SLBP_NC0_Pre9_R2_B2     SLBP NC0_Pre9        R2 SLBP_NC0_Pre9_R2
## 3 SLBP_NC0_Pre9_R3_B2     SLBP NC0_Pre9        R3 SLBP_NC0_Pre9_R3
## 4 SLBP_NC0_Pre9_R4_B2     SLBP NC0_Pre9        R4 SLBP_NC0_Pre9_R4
## 5 SLBP_NC0_Pre9_R5_B2     SLBP NC0_Pre9        R5 SLBP_NC0_Pre9_R5
## 6     SLBP_NC13_R1_B1     SLBP     NC13        R1     SLBP_NC13_R1
##         Gen_Rep_Batch       Gen_Cyc
## 1 SLBP_NC0_Pre9_R1_B2 SLBP_NC0_Pre9
## 2 SLBP_NC0_Pre9_R2_B2 SLBP_NC0_Pre9
## 3 SLBP_NC0_Pre9_R3_B2 SLBP_NC0_Pre9
## 4 SLBP_NC0_Pre9_R4_B2 SLBP_NC0_Pre9
## 5 SLBP_NC0_Pre9_R5_B2 SLBP_NC0_Pre9
## 6     SLBP_NC13_R1_B1     SLBP_NC13
```

```
levels(sample.slbp.all$Gen_Cyc)
```

```
## [1] "SLBP_NC0_Pre9" "SLBP_NC13"     "WT_NC0_Pre9"   "WT_NC13"
## [5] "WT_NC14"
```

```
sample.slbp.all$Gen_Cyc <- relevel(sample.slbp.all$Gen_Cyc, ref="WT_NCO_Pre9")
# Ordering the dataset such that WT_NCO_Pre9
# i.e. the most initial developmental stages is the reference for all comparisons

# Reading in the raw count matrix files and coverting it into a DESeq object
files.slbp.all <- file.path("salmon_data_slbp_main_pilot_inc",
                            sample.slbp.all$Sample, "quant.sf")
names(files.slbp.all) <- paste0("Sample", 1:28)
all(file.exists(files.slbp.all))
```

```
## [1] TRUE
```

```
txi.slbp.all <- tximport(files.slbp.all, type = "salmon", tx2gene = ttg.slbp)
```

```
## reading in files with read_tsv
```

```
## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
## summarizing abundance
## summarizing counts
## summarizing length
```

```
names(txi.slbp.all)
```

```
## [1] "abundance"           "counts"              "length"
## [4] "countsFromAbundance"
```

```
rownames(sample.slbp.all) <- colnames(txi.slbp.all$counts)
ddsTxi.slbp.all <- DESeqDataSetFromTximport(txi.slbp.all,
                                            colData = sample.slbp.all,
                                            design = ~ Gen_Cyc)
```

```
## using counts and average transcript lengths from tximport
```

```
# The design here is Gen_Cyc == Genotype*Cycle == Genotype + Cycle + Genotype:Cycle
# main and interaction effects of Genotype and Cell cycle
```

There are a couple of other manipulations that can be performed before performing the actual analysis
- The same libraries were sequenced on multiple lanes. These form technical replicates and can be combined into the appropriate biological replicate by summing the counts, using the `collapseReplicates()` function.
- The genes that have not received any reads or say below a threshold level of counts can be eliminated before analysis.

```
# Collapsing the technical replicates
ddsColl.slbp.all <-collapseReplicates(ddsTxi.slbp.all,
                                      ddsTxi.slbp.all$Gen_Rep,
                                      ddsTxi.slbp.all$Gen_Rep_Batch)
colData(ddsColl.slbp.all)
```

```
## DataFrame with 25 rows and 8 columns
##                              Sample Genotype    Cycle Replicate
##                            <factor> <factor> <factor>  <factor>
## SLBP_NC0_Pre9_R1 SLBP_NC0_Pre9_R1_B2     SLBP NC0_Pre9        R1
## SLBP_NC0_Pre9_R2 SLBP_NC0_Pre9_R2_B2     SLBP NC0_Pre9        R2
## SLBP_NC0_Pre9_R3 SLBP_NC0_Pre9_R3_B2     SLBP NC0_Pre9        R3
## SLBP_NC0_Pre9_R4 SLBP_NC0_Pre9_R4_B2     SLBP NC0_Pre9        R4
## SLBP_NC0_Pre9_R5 SLBP_NC0_Pre9_R5_B2     SLBP NC0_Pre9        R5
## ...                                ...      ...      ...       ...
## WT_NC14_R1             WT_NC14_R1_B2       WT     NC14        R1
## WT_NC14_R2             WT_NC14_R2_B2       WT     NC14        R2
## WT_NC14_R3             WT_NC14_R3_B2       WT     NC14        R3
## WT_NC14_R4             WT_NC14_R4_B2       WT     NC14        R4
## WT_NC14_R5             WT_NC14_R5_B2       WT     NC15        R5
##                           Gen_Rep       Gen_Rep_Batch       Gen_Cyc
##                          <factor>            <factor>      <factor>
## SLBP_NC0_Pre9_R1 SLBP_NC0_Pre9_R1 SLBP_NC0_Pre9_R1_B2 SLBP_NC0_Pre9
## SLBP_NC0_Pre9_R2 SLBP_NC0_Pre9_R2 SLBP_NC0_Pre9_R2_B2 SLBP_NC0_Pre9
## SLBP_NC0_Pre9_R3 SLBP_NC0_Pre9_R3 SLBP_NC0_Pre9_R3_B2 SLBP_NC0_Pre9
## SLBP_NC0_Pre9_R4 SLBP_NC0_Pre9_R4 SLBP_NC0_Pre9_R4_B2 SLBP_NC0_Pre9
## SLBP_NC0_Pre9_R5 SLBP_NC0_Pre9_R5 SLBP_NC0_Pre9_R5_B2 SLBP_NC0_Pre9
## ...                           ...                 ...           ...
## WT_NC14_R1             WT_NC14_R1       WT_NC14_R1_B2       WT_NC14
## WT_NC14_R2             WT_NC14_R2       WT_NC14_R2_B2       WT_NC14
## WT_NC14_R3             WT_NC14_R3       WT_NC14_R3_B2       WT_NC14
## WT_NC14_R4             WT_NC14_R4       WT_NC14_R4_B2       WT_NC14
## WT_NC14_R5             WT_NC14_R5       WT_NC14_R5_B2       WT_NC14
##                         runsCollapsed
##                           <character>
## SLBP_NC0_Pre9_R1 SLBP_NC0_Pre9_R1_B2
## SLBP_NC0_Pre9_R2 SLBP_NC0_Pre9_R2_B2
## SLBP_NC0_Pre9_R3 SLBP_NC0_Pre9_R3_B2
## SLBP_NC0_Pre9_R4 SLBP_NC0_Pre9_R4_B2
## SLBP_NC0_Pre9_R5 SLBP_NC0_Pre9_R5_B2
## ...                              ...
## WT_NC14_R1             WT_NC14_R1_B2
## WT_NC14_R2             WT_NC14_R2_B2
## WT_NC14_R3             WT_NC14_R3_B2
## WT_NC14_R4             WT_NC14_R4_B2
## WT_NC14_R5             WT_NC14_R5_B2
```

```
# Retaining a gene if it has over 1 reads in 2 or more samples
# This is a very permissive threshold to eliminate any unintentional filtration bias
ddsColl.slbp.all
```

```
## class: DESeqDataSet
## dim: 13728 25
## metadata(1): version
## assays(2): counts avgTxLength
## rownames(13728): dmel_1-Dec dmel_1-Sep ... dmel_zye dmel_Zyx
## rowData names(0):
## colnames(25): SLBP_NC0_Pre9_R1 SLBP_NC0_Pre9_R2 ... WT_NC14_R4
##   WT_NC14_R5
## colData names(8): Sample Genotype ... Gen_Cyc runsCollapsed
```

```
keep.slbp.all <- rowSums(counts(ddsColl.slbp.all) >1) >= 2
ddsColl.keep.slbp.all <- ddsColl.slbp.all[keep.slbp.all,]
ddsColl.keep.slbp.all # 10860 genes
```

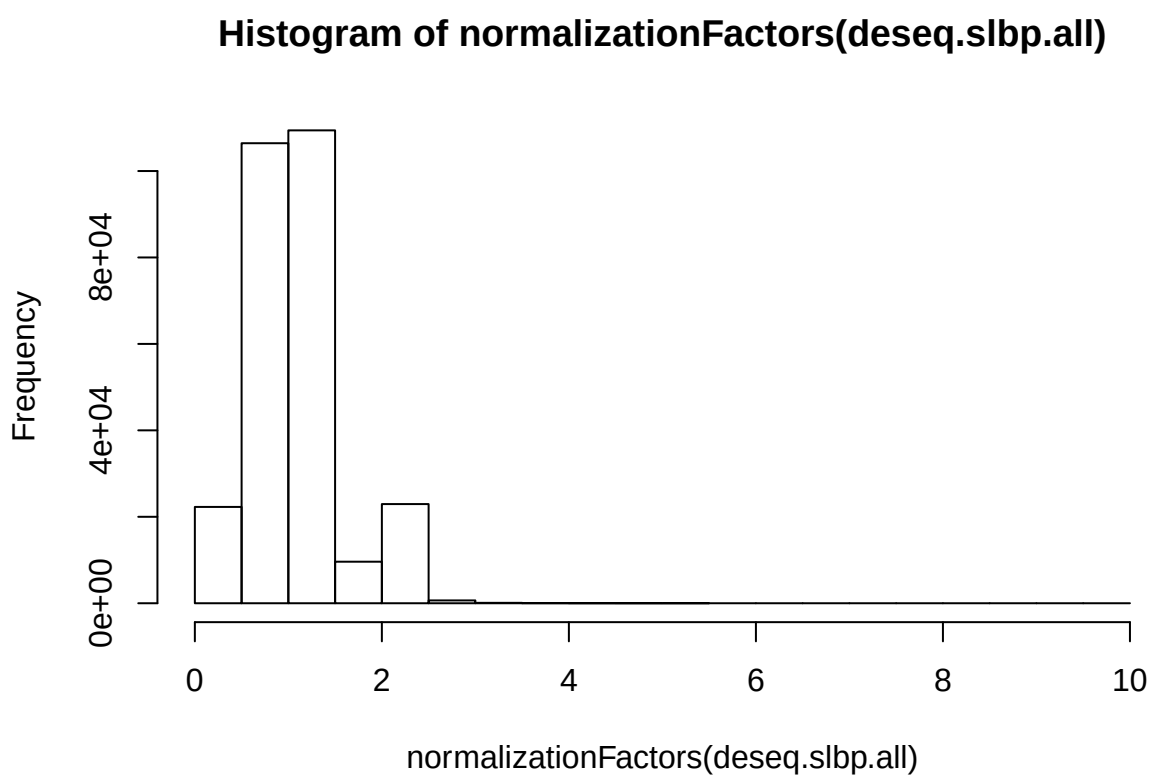
```
## class: DESeqDataSet
## dim: 10860 25
## metadata(1): version
## assays(2): counts avgTxLength
## rownames(10860): dmel_1-Dec dmel_1-Sep ... dmel_zye dmel_Zyx
## rowData names(0):
## colnames(25): SLBP_NC0_Pre9_R1 SLBP_NC0_Pre9_R2 ... WT_NC14_R4
##   WT_NC14_R5
## colData names(8): Sample Genotype ... Gen_Cyc runsCollapsed
```

Perform the DESeq2 normalization followed by
- regularized log transformation
- similarity between samples based on Euclidean distance
- principal components analysis (PCA)

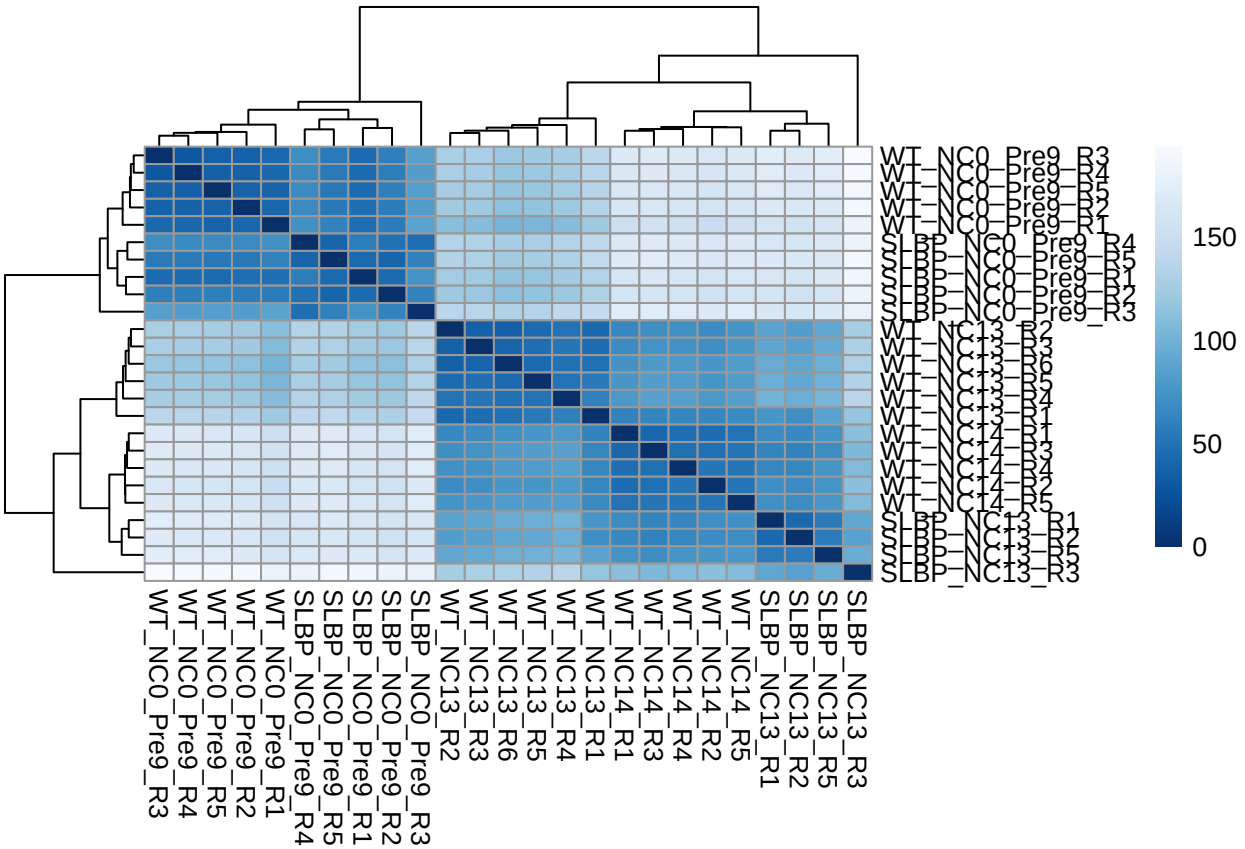
```
deseq.slbp.all <- DESeq(ddsColl.keep.slbp.all)
```

```
## estimating size factors

## using 'avgTxLength' from assays(dds), correcting for library size

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

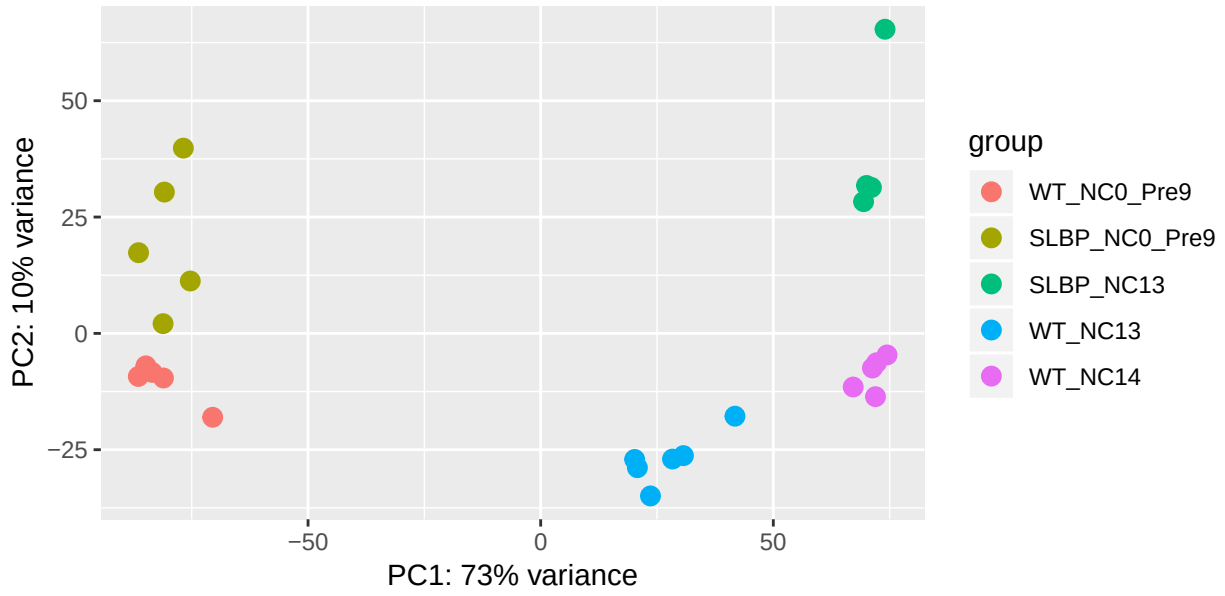
```
hist(normalizationFactors(deseq.slbp.all))
```




```
# histogram of normalization factors typically distributed with a mean/ median ~1.0
rld.slbp.all <- rlog(deseq.slbp.all, blind=T)

sampleDists.slbp.all <- dist(t(assay(rld.slbp.all)))
hm.mat.slbp.all <- as.matrix( sampleDists.slbp.all)
colors <- colorRampPalette( rev(brewer.pal(9, "Blues")) )(255)

# heirarchical clustering of the sample distances
hm.col.clust.slbp.all <- hclust(dist(t(hm.mat.slbp.all)))
sort.hclust <- function(...) as.hclust(dendsort(as.dendrogram(...)))
hm.col.clust.slbp.all <- sort.hclust(hclust(dist(t(hm.mat.slbp.all))))
hm.row.clust.slbp.all <- sort.hclust(hclust(dist(hm.mat.slbp.all)))

#visualizing the clustered sample distances
pheatmap(
  mat               = hm.mat.slbp.all,
  color                  = colors,
  cluster_cols      = hm.col.clust.slbp.all,
  cluster_rows      = hm.row.clust.slbp.all,
  show_rownames     = T
)
```




```
#plotting PCA
plotPCA(rld.slbp.all, intgroup=c("Gen_Cyc"), ntop=5000)
```



The Euclidean distance based similarity of expression data for WT and Slbp embryos at different time-points demonstrate that replicates are more similar to each other than they are to other genotypes and/ or timepoints.

Principal Components Analysis of Slbp and WT expression data also shows that global transcriptomic profile is shifted in Slbp NC13 as compared WT NC13 and is more similar to WT NC14 indicating that the onset of transcription is advanced in Slbp embryos.

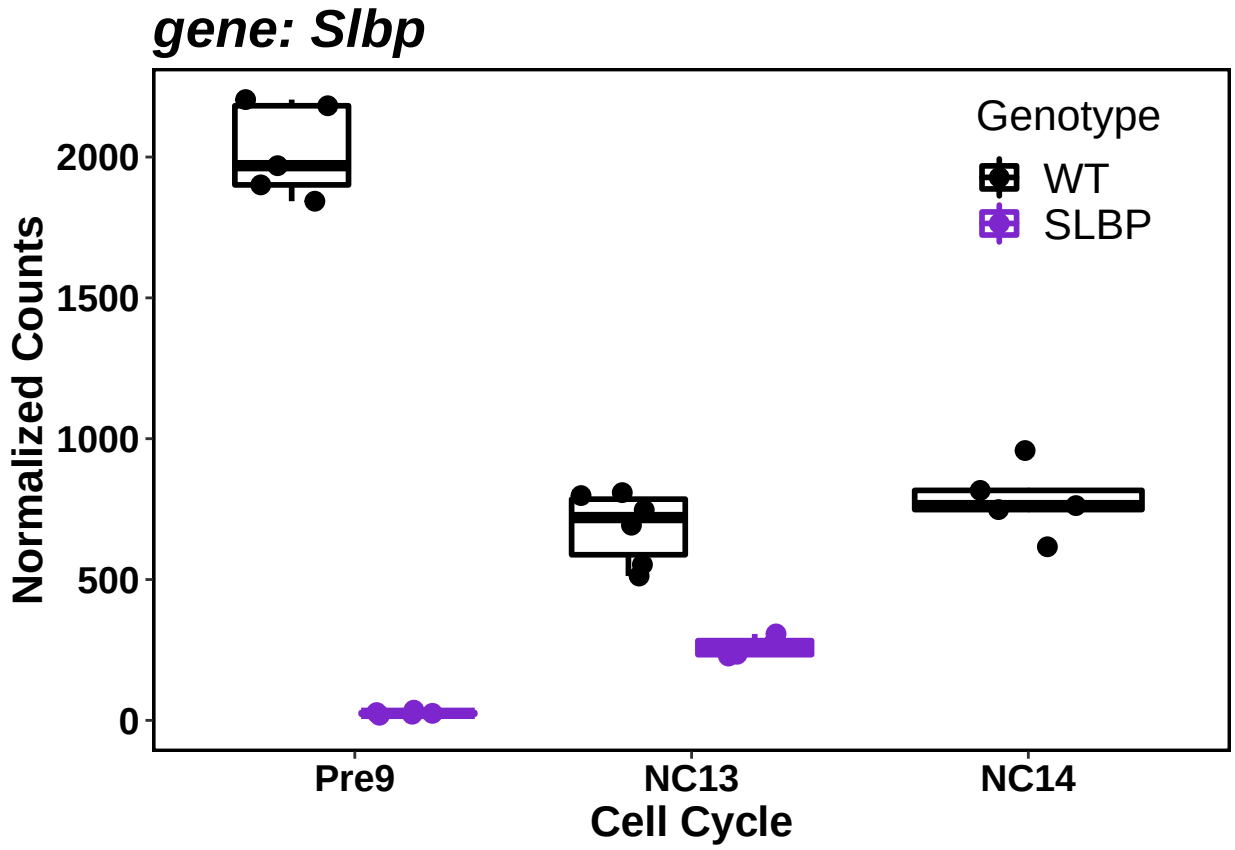
Plotting specific gene counts across development

```
# Extract Normalized Counts
counts.slbp.all <- as.data.frame(counts(deseq.slbp.all, normalized=T))
counts.slbp.all <- cbind(Gene = rownames(counts.slbp.all), counts.slbp.all)
counts.slbp.all.fbgn <- merge (fbgn.gene.conv, counts.slbp.all, by="Gene")

# Reorder such that WT_Pre9 is the 1st column after Gene Id
counts.slbp.all.ord <- counts.slbp.all.fbgn[c(1,2,12,13,14,15,16,3,4,5,6,7,17,18,19,20,21,22,8,9,10,11,1

# Change colnames for simplification
# Tip if you require the only entire string to match then use "^NC0_Pre9$"
names(counts.slbp.all.ord) <- gsub("NC0_Pre9","Pre9",names(counts.slbp.all.ord))

# convert it to long form for ease of plotting using tidy (or melt() in reshape2)
counts.long.slbp <- gather (counts.slbp.all.ord, Sample_Id, Normalized_Count,-Gene,-geneId)

# split the Sample_Id column into appropriate columns further
```

```
counts.long.slbp$Sample_Id2 <- counts.long.slbp$Sample_Id # duplicating
counts.long.slbp <- counts.long.slbp[c(1,2,3,5,4)] # reorder
counts.slbp.plt <- separate (counts.long.slbp, Sample_Id2,
                             into = c("Genotype","Cell_Cycle","Replicate"),
                             sep="_")

#### Plotting using ggplot2 ###
str(counts.slbp.plt) # shows the newly created columns are characters
```

```
## 'data.frame':    271500 obs. of  7 variables:
##  $ Gene           : Factor w/ 17753 levels "dmel_1-Dec","dmel_1-Sep",..: 1 2 3 4 5 6 12 13 14 27 ..
##  $ geneId         : Factor w/ 17753 levels "FBgn0000003",..: 138 1252 1058 1866 919 1059 768 1541 13
##  $ Sample_Id      : chr  "WT_Pre9_R1" "WT_Pre9_R1" "WT_Pre9_R1" "WT_Pre9_R1" ...
##  $ Genotype       : chr  "WT" "WT" "WT" "WT" ...
##  $ Cell_Cycle     : chr  "Pre9" "Pre9" "Pre9" "Pre9" ...
##  $ Replicate      : chr  "R1" "R1" "R1" "R1" ...
##  $ Normalized_Count: num  0 3593 1988 23419 7521 ...
```

```
#Convert them to factor since subsetting, releveling, equality testing etc become easier
counts.slbp.plt[,c(3:6)] <- lapply(counts.slbp.plt[,c(3:6)], factor)
counts.slbp.plt$Genotype <- relevel(counts.slbp.plt$Genotype, ref="WT")
counts.slbp.plt$Cell_Cycle <- relevel(counts.slbp.plt$Cell_Cycle, ref="Pre9")
counts.slbp.plt[is.na(counts.slbp.plt)] <- 0

# Boxplot w/ individual data points
gn.all.slbp <- subset(counts.slbp.plt, Gene=="dmel_Slbp") # gene name within quotes as appears in the f
ggplot(gn.all.slbp,
       aes(x = Cell_Cycle, y = Normalized_Count, color=Genotype)) +
  #provides x, y axis and colors
       geom_boxplot(aes(group = interaction(Cell_Cycle,Genotype)),lwd=1,outlier.shape=NA) +
  # instructs to do boxplots by Cycle & Genotype
       geom_point(aes(group = interaction(Cell_Cycle,Genotype)), size=3,
                  position=position_jitterdodge(0.6)) +
       scale_y_continuous(limits=c(0,max(gn.all.slbp$Normalized_Count))) +
  # Automatically plots y axis from 0 to the maximum Count value for a given gene
       labs(title="gene: Slbp", x="Cell Cycle", y="Normalized Counts") +
  # title for the graph i.e. the gene name within quotes
       scale_colour_manual(values=c("black","purple3"),
                           name="Genotype",
                           breaks=c("WT","SLBP")) +
  theme_bw() +
  theme(plot.title = element_text (color="black", size=20, face="bold.italic"),
              axis.text.x = element_text (color="black", size=14, face="bold"),
              axis.text.y = element_text (color="black", size=14, face="bold"),
              axis.title.x = element_text (color="black", size=16, face="bold"),
              axis.title.y = element_text (color="black", size=16, face="bold"),
              legend.position=c(0.85,0.85),
              legend.title=element_text(color="black", size=16),
              legend.text = element_text(size = 16),
              panel.grid.major = element_blank(),
              panel.grid.minor = element_blank(),
              panel.border = element_rect(colour = "black", fill=NA, size=1))
```

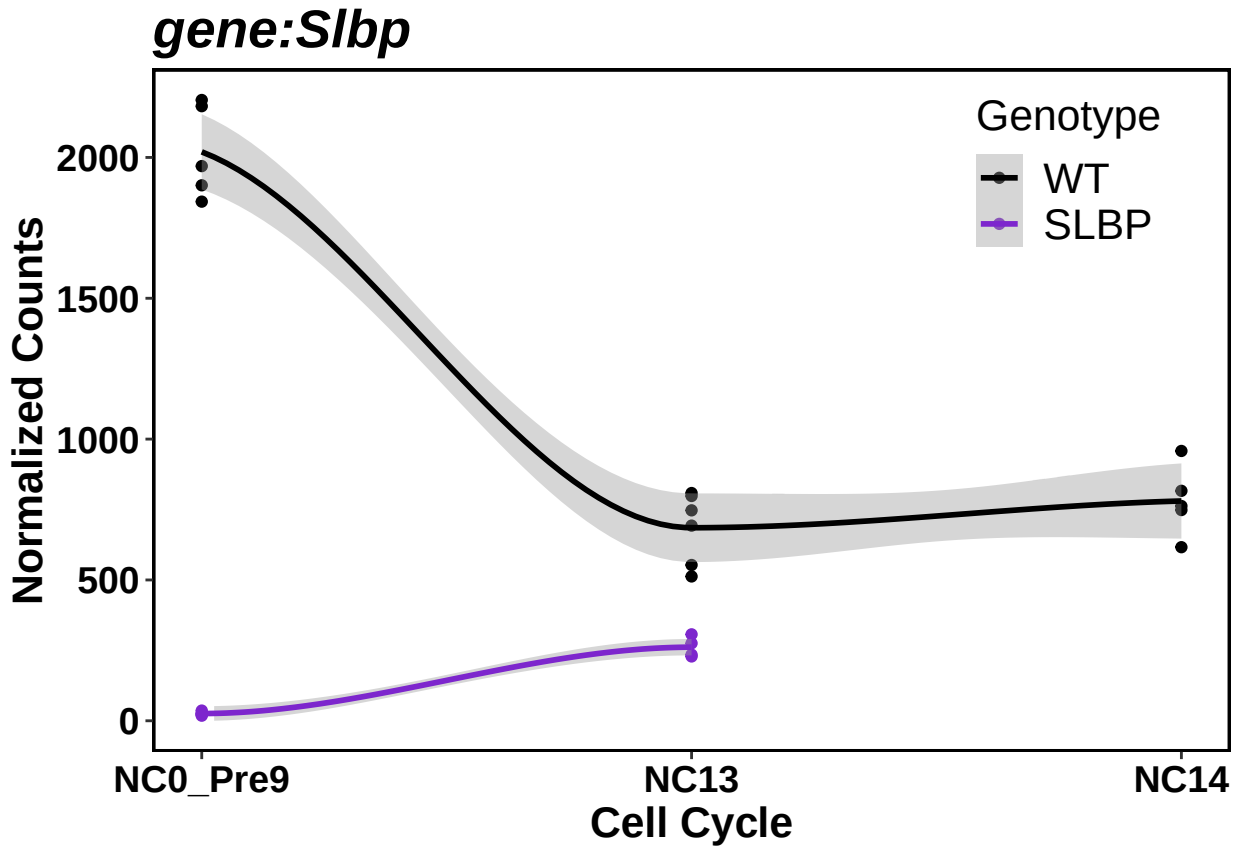


The gene that is supposed to be knocked down in the mutant genonotype *Slbp* has a lower expression in the mutant as compared to the wild-type.

This graph can also be plotted as a set of points with a spline function with SE.

```
gn.all.slbp <- subset(counts.slbp.plt, Gene=="dmel_Slbp")

ggplot(gn.all.slbp,
  aes(x = as.numeric(Cell_Cycle), y = Normalized_Count, color=Genotype)) +
  geom_point(aes(group = interaction(Cell_Cycle,Genotype))) +
  geom_smooth(se=T,method="loess") +
  scale_y_continuous(limits=c(0,max(gn.all.slbp$Normalized_Count))) +
  # Automaticall.slbpy plots y axis from 0 to the maximum Count value for a given gene
  scale_x_continuous(breaks=c(1,2,3),labels=c("NC0_Pre9","NC13","NC14")) +
  labs(title="gene:Slbp", x="Cell Cycle", y="Normalized Counts") +
  # title for the graph i.e. the gene name within quotes
  scale_colour_manual(values=c("black","purple3"),
                      name="Genotype",
                      breaks=c("WT","SLBP")) +
  theme_bw() +
  theme(plot.title = element_text (color="black", size=20, face="bold.italic"),
              axis.text.x = element_text (color="black", size=14, face="bold"),
              axis.text.y = element_text (color="black", size=14, face="bold"),
              axis.title.x = element_text (color="black", size=16, face="bold"),
              axis.title.y = element_text (color="black", size=16, face="bold"),
              legend.position=c(0.85,0.85),
              legend.title=element_text(color="black", size=16),
```

```
        legend.text = element_text(size = 16),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_rect(colour = "black", fill=NA, size=1))
```



We can similarly plot any particular gene of interest.