Module 3.1: Learn - Biology

Overview

In the previous modules, you have learned about how to use RNAseq to measure gene expression and prioritize candidate genes that explain sex differences in the placenta. In this module, we will learn important steps in how to process gene count data in order to determine those differentially expressed genes and ways that we can visualize those results. This section includes several awesome videos that visually describe important algorithms used in data processing and visualization of gene expression data; we will do our best to present the information in multiple ways to help you understand the techniques you are using.

In this section of Module 3 you will learn:

1. Why we conduct transformation and normalization of RNAseq data
2. How linear modeling can be used to identify differentially expressed genes
3. How to visualize sample groupings by gene expression profiles
4. Aspects of gene expression that we can use to describe the effects of trimming on data

Normalization

Starting with count data (the number of sequencing reads that map to genes, as defined by their coordinates in the genome), it is an important step to normalize the data properly to make accurate and precise comparisons of gene expression between samples.

It is understood that for RNAseq experiments, the counts of mapped reads for each gene is proportional to the expression of that gene, which is defined as the transcription of mRNA from the DNA of each gene in the chromosomes in the cells or tissues we are assaying, as well as other technical factors in the processing of the sample. While we try our best during sample collection and sequencing to process samples cleanly and reproducibly, there will always be some level of technical variability in sequencing data. We want to process

our data to minimize the effect of technical variation in the samples so that we can better distinguish the effect of are interested in.

Count Normalization

The first type of normalization that is done is a transformation of raw counts into a unit that accounts for differenc number of sequenced reads in a sample).

The main factors we consider during count transformation/normalization are (figures below from this guide: **DGE (https://hbctraining.github.io/DGE_workshop/lessons/02_DGE_count_normalization.html)**):

1. Sequencing depth: the number of reads sequenced that align to genes
    - We can talk about the sequencing depth for specific genes (for example, sequencing depth across all exo specific chromosome)
    - We can talk about overall sequencing depth (for example, 60 million reads total from a particu

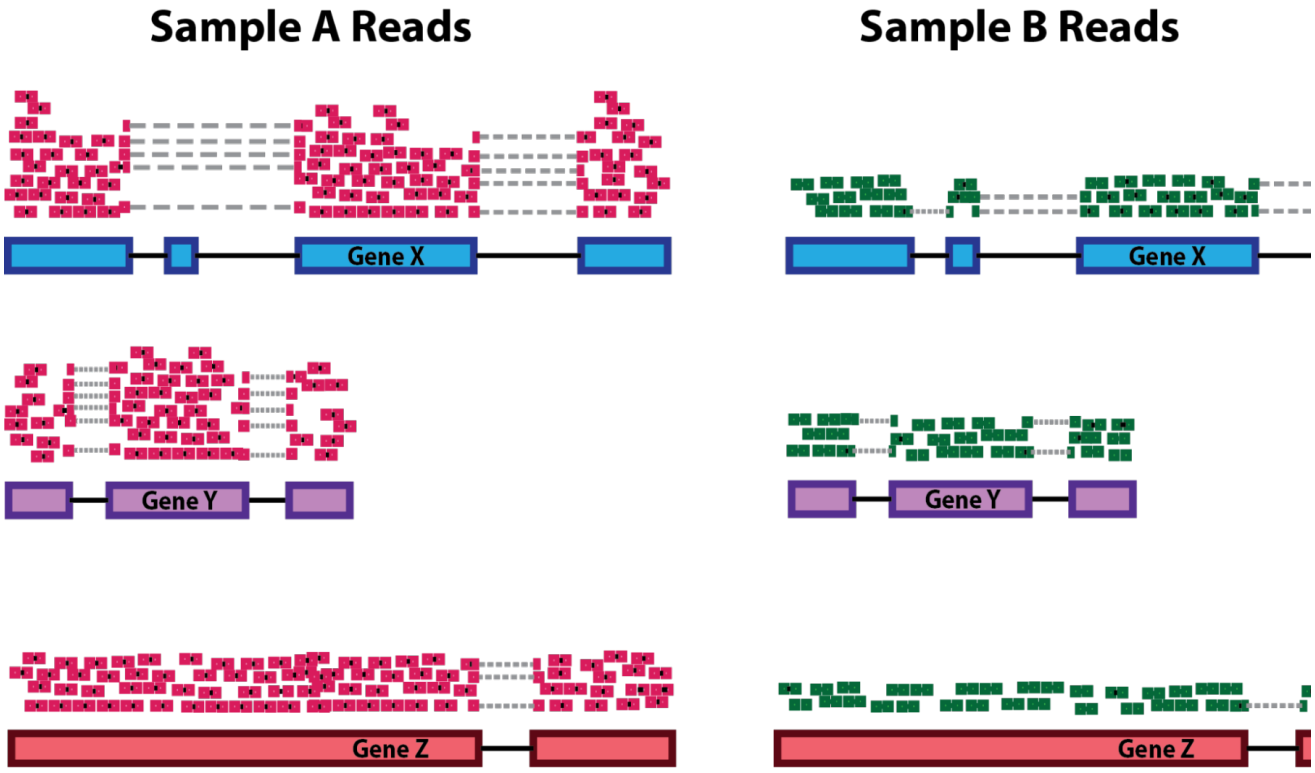


Figure. Aligning reads to genes.
The pink and gene rectangles represent reads aligned to exons of a gene and the dashed lines are reads aligned to intronic sequence. Each gene in Sample A appears to have doubled in expression, but that is only because there was double overall sequencing depth for Sample A than in Sample B, which you can see because all of the genes in Sample A have twice the counts as in Sample B.

2. Gene length: longer genes will have more reads that map simply because there is more sequences to read fr have higher expression
    - Accounting for this is important when you are comparing counts of different genes within the same sample
    - This is not considered when you are comparing the expression of a specific gene across samples becau length remains constant in each sample

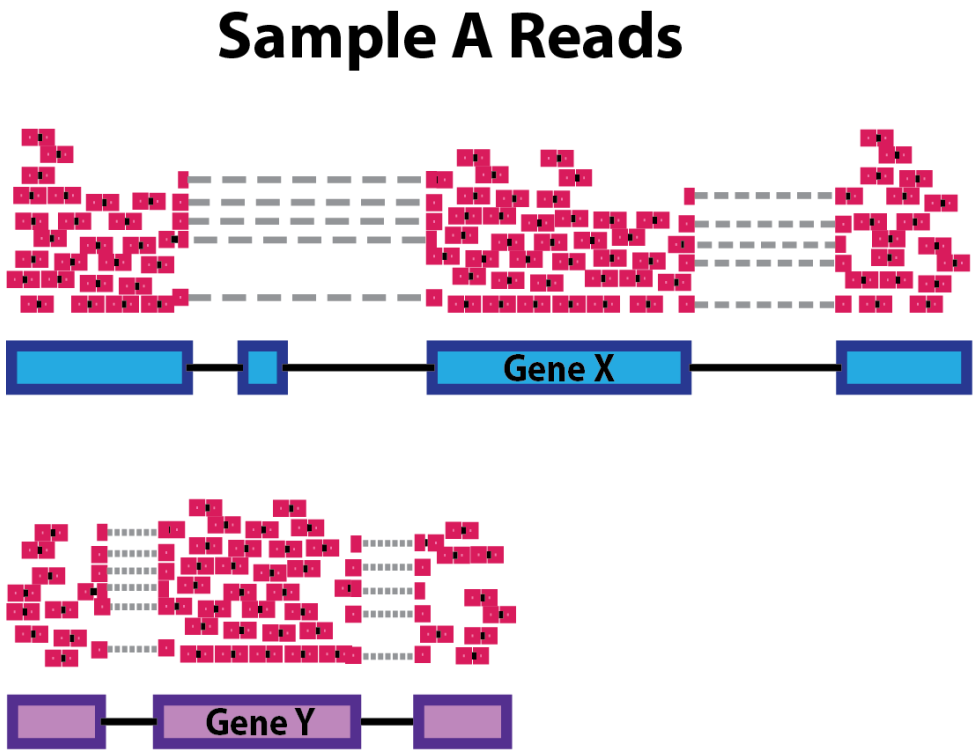


Figure. Sample A reads aligning to the sex chromosomes.
Gene X and Gene Y have similar levels of expression but more

reads align to Gene X just because it is longer.

3. RNA composition: High expression of specific gene(s) in specific samples or the presence of contamination c
   up your normalization methods
   - This one can be harder to detect and is more important when comparing gene expression wit

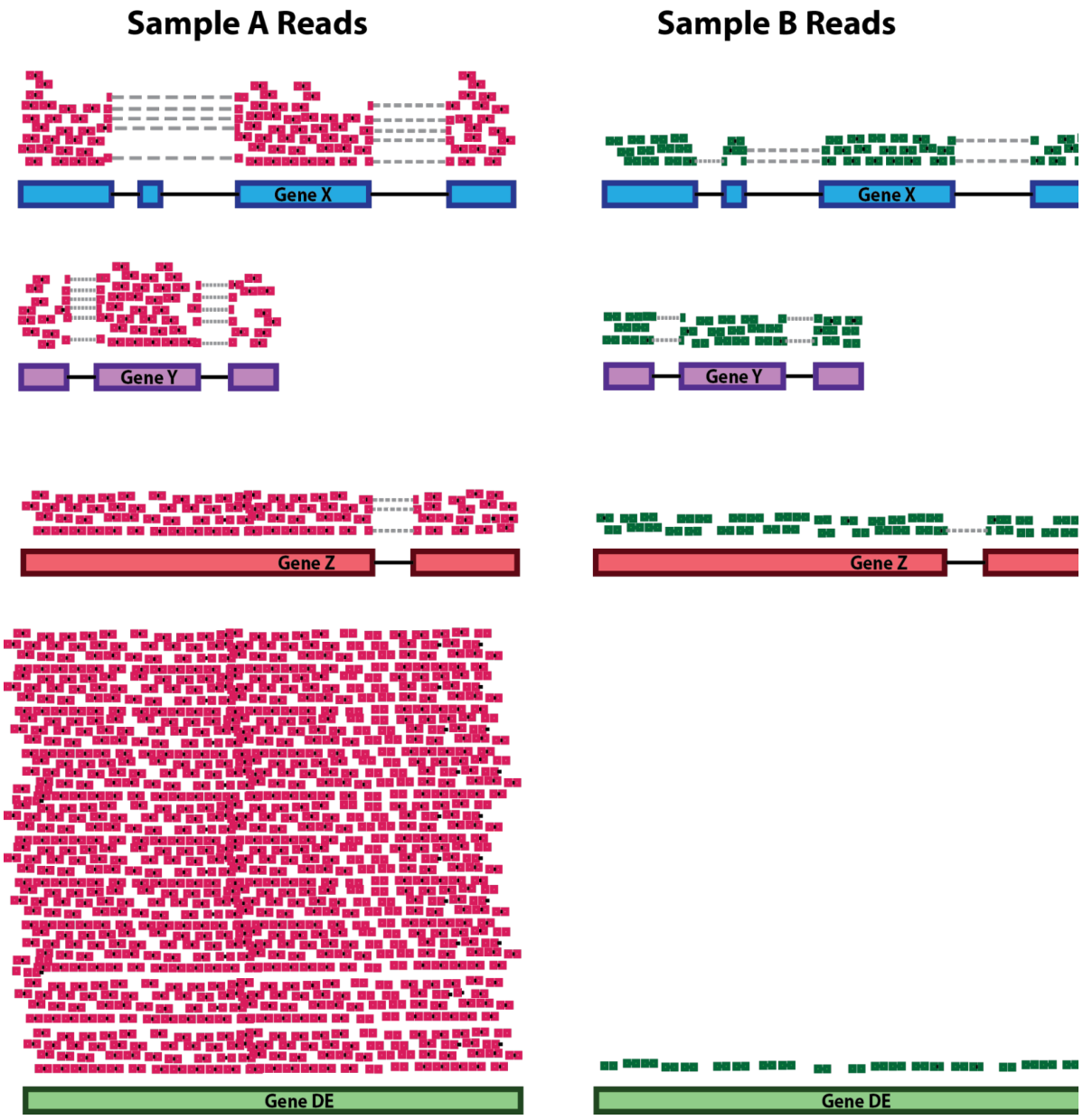


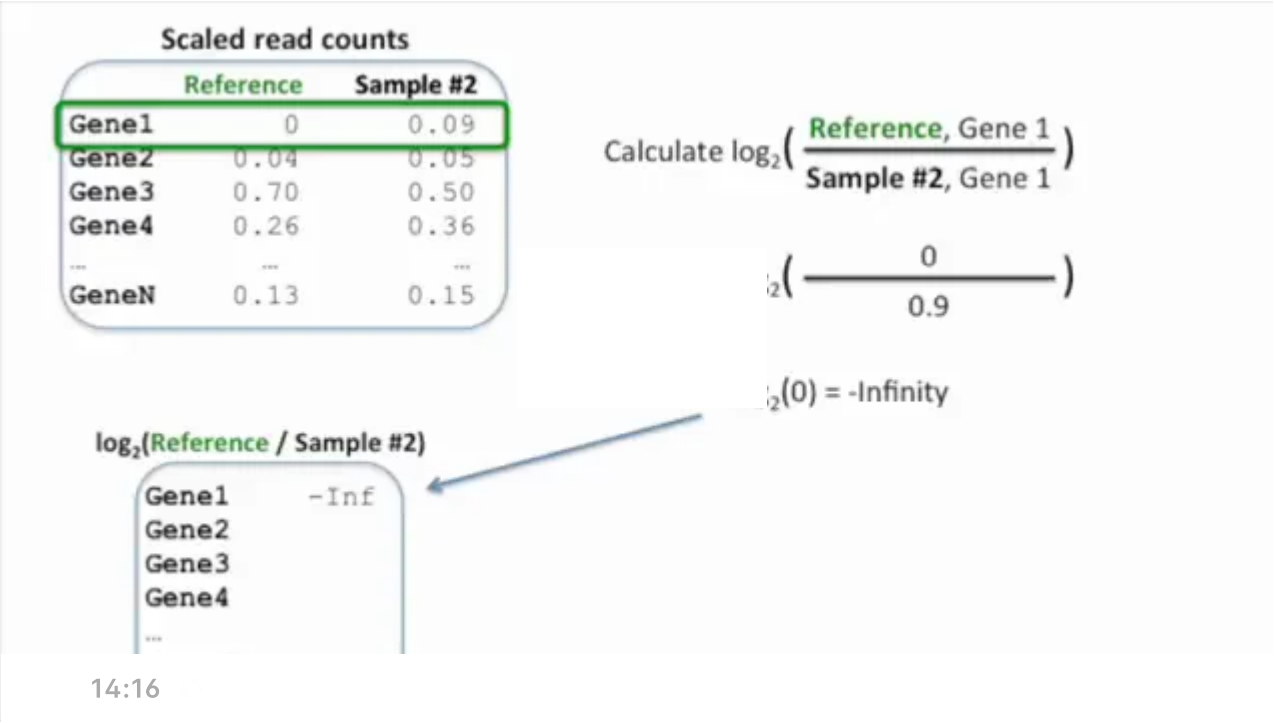
Figure: Differentially expressed gene counts.
Differentially expressed Gene DE accounts for most of the
counts in the sample, so dividing all the other genes in the sample
using the total number of counts would make it appear that all the
other genes are expressed at lower levels.

Since we are focused on looking at differences between samples, our pipeline converts the raw data to counts pe
dividing the counts by the total number of read sequences in the sample.

Sample Normalization

After converting to CPM, the next type of normalization is at a sample level to account for external factors that ar
studying in our experiment. We try to normalize for fluctuations in the samples that affect the samples independe
differences in overall sequencing or read depth. Normalization is required to ensure that the range of expression
all samples in a data set so that the differences we observe between samples are reflective of biological variatior

Normalization of our data will be done with the edgeR package in R. While the chunk of R code for this step is o
video explains all the stuff happens under the hood:



Video. Library Normalization using edgeR.

This video describes how edgeR normalization method to calculate scaling factors.

View transcript. (https://canvas.asu.edu/courses/122165/files/54792258?wrap=1) **(https://canvas.asu.edu/courses/122165/files/54792258/download?download_frd=1)**

edgeR normalization method to calculate scaling factors:

1. Remove genes with low expression
2. Uses scaled read counts (CPM)
3. Picks a reference sample to normalize to that has average expression of most genes
4. Filter genes with biased gene expression and genes with very high or low expression
5. Calculate weighted average of ratios between the reference and the sample being normalized for unbiased, a genes
6. Converts weighted averages to whole numbers
7. Center them around 1 to get a final scaling factor for each sample

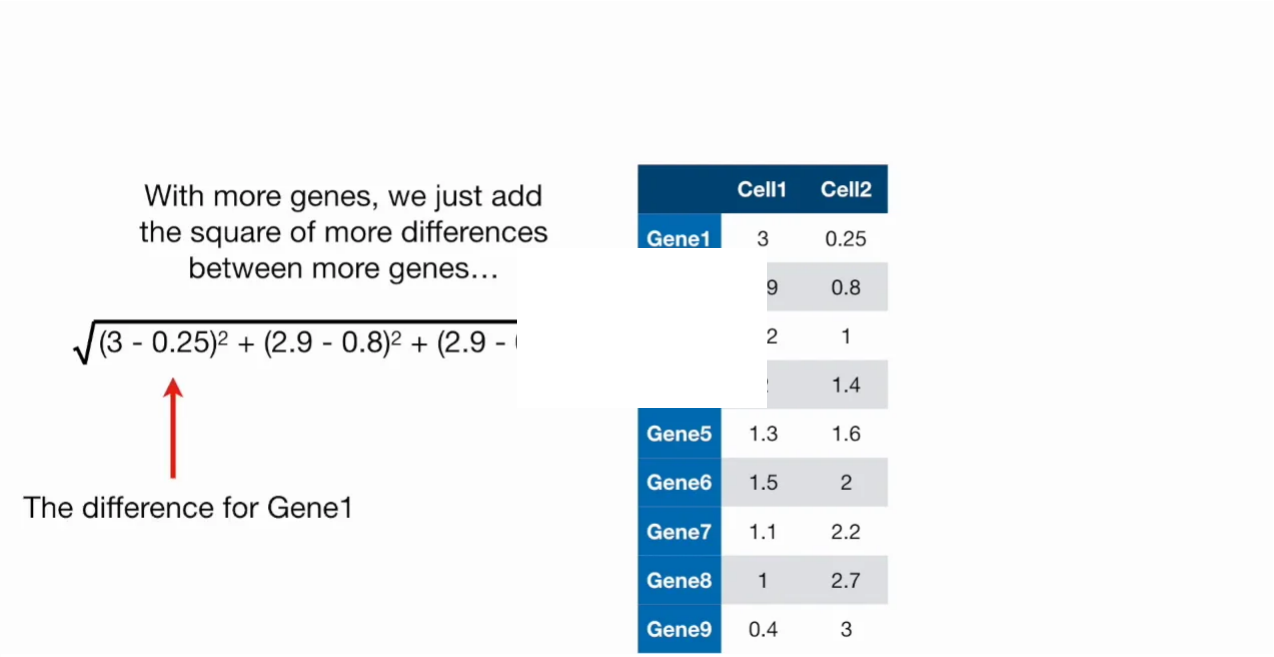
Grouping samples by gene expression profile

Once the data is properly normalized, researchers typically start exploring their data by getting a big picture view are related to each other in terms of their overall gene expression profiles. This can help you to determine wheth are interested in studying have a large or small effect on gene expression as a whole. It can also help you to see might have to specifically address using data processing before continuing with other analyses. For example, if y samples and see a batch effect (grouping of samples run on the same day), you might need to apply a batch corr doing differential gene expression analysis.

Some example batch effects include samples run on the sequencer on one day might be more similar than samp day. Similarly, specific technicians, specific instruments, specific manufacturing batches of assay materials, and influence the distribution of expression values in specific samples.

While there are many ways to visualize relationships between samples in a study, we are going to describe one t pipeline called multi-dimensional scaling (MDS). This plot shows similarity or differences between samples in an (without factoring in any known phenotype information of the samples, using only the gene expression data). Thi represents the differences between expression values as a distance and plots the samples according to average pairs of samples.

This video describes how MDS (and related technique Principal Component Analysis (PCA) creates a 2D plot to between samples' gene expression profiles:



Video. MDS and PCoA clearly explained.

This video explains multidimensional scaling (MDS) and principal coordinate analysis (PCo/

View Transcript. (https://canvas.asu.edu/courses/122165/files/54792256?wrap=1) ↓

(https://canvas.asu.edu/courses/122165/files/54792256/download?download_frd=1)

The video compares different “cells” but our gene expression measurements in our case is from tissues, which yc average gene expression in all the cells in the tissue sample collected. It describes how we can calculate distanc and summarize results from many genes to plot the similarity between samples.

The figure panel below (included in the placenta sex differences paper) shows MDS plots for gene expression pr term placentas in our study labeled by genetic sex. The left figure uses the expression of all genes while the right top 100 most varying genes (standard deviation divided by the mean expression across all samples). You can se between the male and female samples. The results for MDS will include all the principal components sorted in de variation captured; the vast majority of the variation in gene expression data sets will be captured within the first t differential expression pipeline code you will be running this week, you will see the percentage of total variation ir component printed in the axis labels.

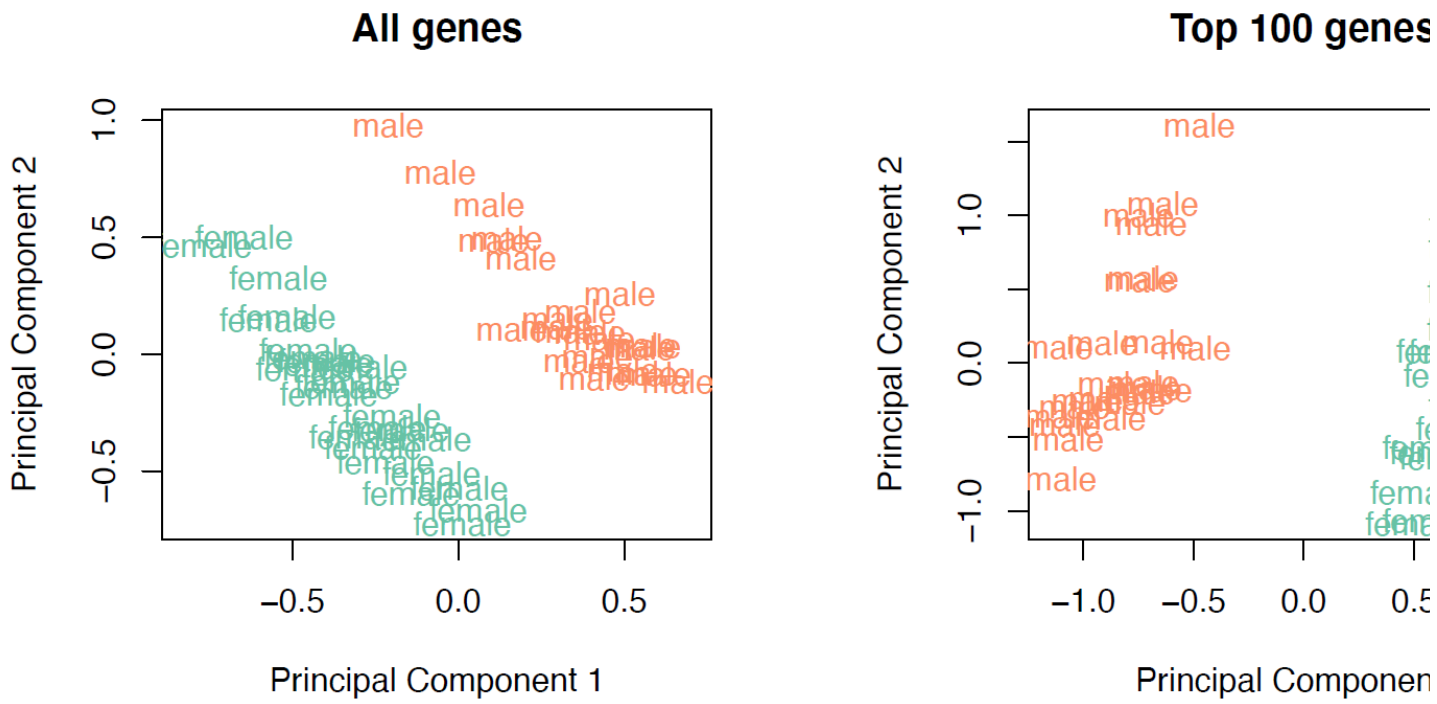


Figure. MDS Plots clustered by sexes.

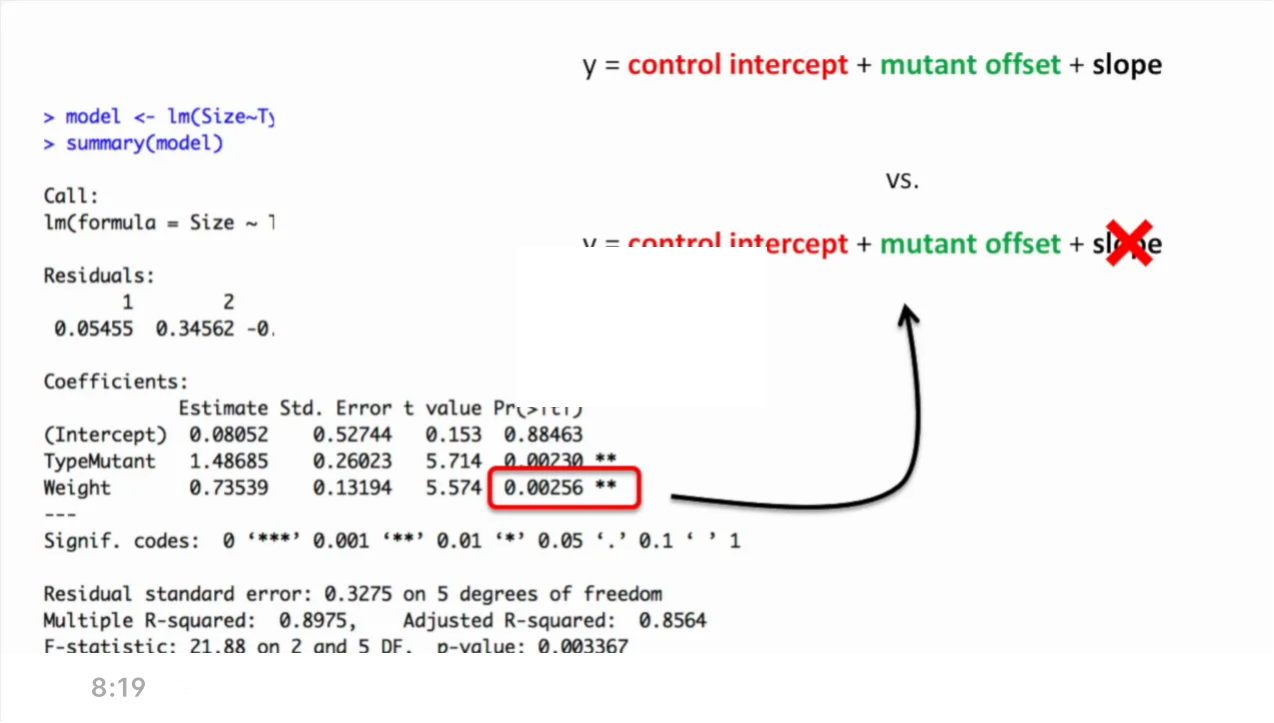
MDS plots of all full term placenta samples from placenta sex differences manuscript.

When analyzing MDS plots, it's good to note which principal components your samples separate on. The first pri summarizes results of genes with the most common expression pattern, the second principal component summa next most common expression pattern when you factor out the variation captured in the first principal component expression of all genes (left), you can see that the samples separate using both principal components 1 and 2 an most varying genes (right) shows separation on the first principle component but not on the second. The reaso that the more you can see separation in the first principal component, the stronger your signal is. If you don't see samples within the first few principal components, you very likely will not detect any differentially expressed gene features you are visualizing (sex in our case). In this way, MDS plots are used as a quick first pass to see if vario samples can be associated with specific gene expression signatures.

Linear modeling methods for differential expression analysis

In our pipeline, we create a design matrix (or model matrix) to hold information about our data and the features of interested in studying (genetic sex). To declare our comparison of interest, XX female versus XY male, we set up (`FemalevsMale = female - male`). These inputs can be analyzed with linear modeling, the process of describing a response variable as a function of one or more predictor variables. We use linear modeling with the R package of the coefficient sex to the gene expression in our two groups, XX female placentas and XY male placentas.

To explain the basics of linear modeling, this video uses a simple example to show how to create a design matrix groups of mice have a different weight using the basic linear modeling function lm available in R in the default in same tools to set up a design matrix to describe gene expression data of placenta samples of diffe



Video. Design matrix for linear modeling.

This video outlines how to construct a design matrix to show differences between controls and mutations.

View Transcript. (https://canvas.asu.edu/courses/122165/files/54792254?wrap=1) ↓

(https://canvas.asu.edu/courses/122165/files/54792254/download?download_frd=1)

Voom transformation

Before fitting our gene expression data to our variable of interest (genetic sex), our pipeline applies a data transfo
the observation that variation of data is not independent of the mean expression of a gene. Genes with lower me
have higher overall variance. The `voom` function in the `limma` package applies a weighting to the expression da
of higher noise in lower expression data so that we can get more accurate results when finding differentially expr
lower expression.

The figure on the left shows gene expression after converting to log(CPM). Expression on the lower end (left of x
higher variation (standard deviation across samples). The figure on the right shows gene expression after mean-
transformation using the voom function. Now, the mean of the variance is the same whether the expression is lo
or high (right side of x-axis).

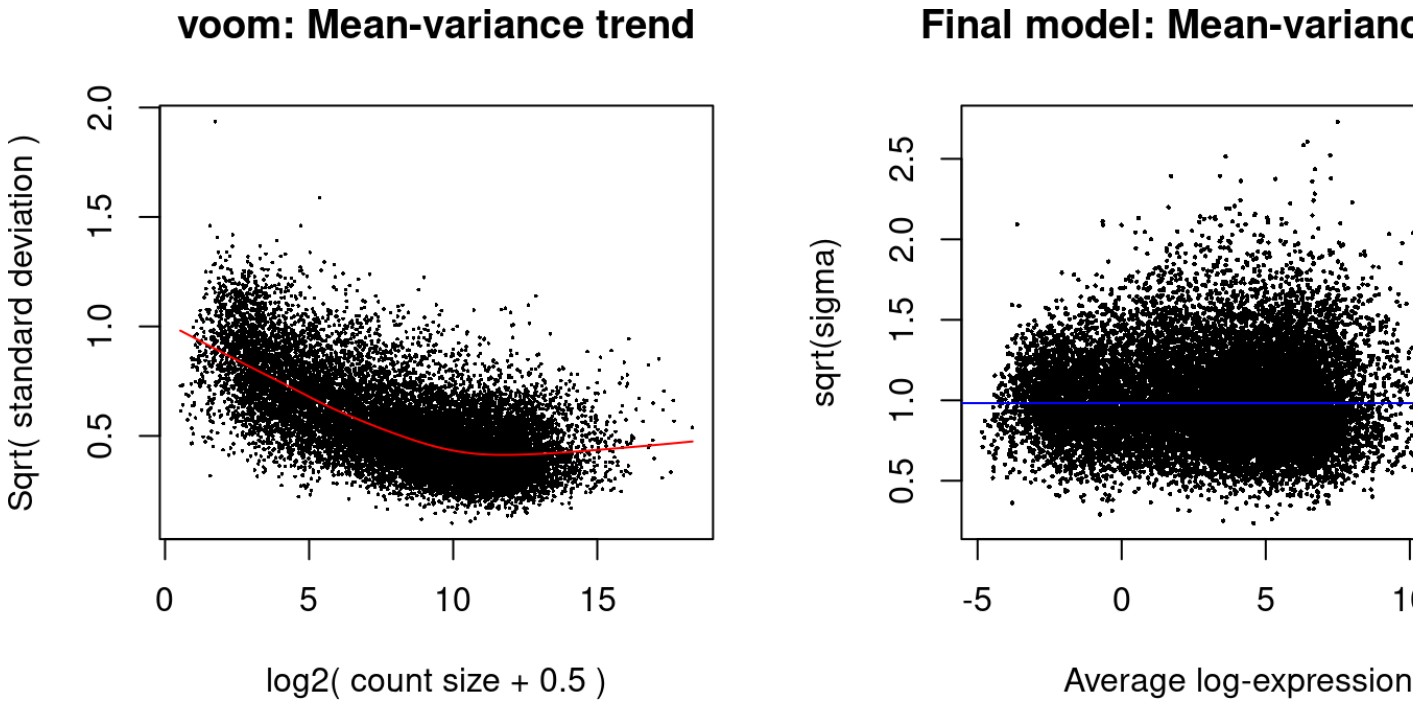


Figure. Voom mean-varience trend.
The quality weights put extra value on data points depending on variance relationship

Fitting the model to our comparisons between female and mal

Our pipeline uses the R package limma which was designed to use linear modeling techniques specifically for ge
using assays like RNAseq. In our code, we will set up the design matrix and use the `makeContrasts` function to
analysis between XX females and XY males.

This video explains the functions of the `limma` package and how to use specific modeling methods to calculate s
to determine differentially expressed genes using a pairwise analysis:

Specifying a linear model in R

5:30

Video. Differential expression analysis with limma in R.
This video will show you how to conduct differential expression analysis with the limma package in R.
View Transcript. (https://canvas.asu.edu/courses/122165/files/54792253?wrap=1)
(https://canvas.asu.edu/courses/122165/files/54792253/download?download_frd=1)

The output of the limma functions is a list of genes that are upregulated in one group (female), downregulated in equivalently upregulated in the second group (male), and not different between the groups. You can look at the v see how different their expression is compared between the two groups you are studying. In the next module, we statistics in more detail. In our case, we will apply a voom transformation before fitting our model so that we get l with lower expression.

Results for differential expression results

After running the differential expression pipeline, you will get a table that reports several important metrics to tell y expression of each gene is in females versus males.

Average expression and fold change

The fold change measures the difference in expression level between the two groups. A gene that truly shows se placentas gene expression will have a small p-value and high fold change.

Fold change is the ratio between the mean expression of a gene in two groups (mean expression in females / me males). Fold changes can be signed by assigning one group as negative and always dividing the larger mean by For example, if the mean expression of GeneA in females was 10, in males was 5, you would get a fold change o 10 / 5, or 2. If the mean expression of GeneB was 5 in females, and 15 in males, we could calculate the fold cha 0.33, or you could assign a negative and take the larger value of 15 divided by the smaller value of 5, giving you Fold changes can also be log-transformed to allow low fold change genes to be more easily displayed with high f

P-values and multiple hypothesis correction

One very important measure of differential gene expression is a p-value, which is a measure of how likely it is tha of the genes in two groups indicates that the two groups are distinct. This video explains what a p-value means:

Video. Statistical significance and p-value.
This video discusses hypothesis testing, how to determine if results are statistically significant, and how p-values are used in research.
View Transcript. (https://canvas.asu.edu/courses/122165/files/54792232?wrap=1)
(https://canvas.asu.edu/courses/122165/files/54792232/download?download_frd=1)

In our analysis, the null hypothesis is that the gene has effectively the same level of expression in the female anc
when we see a p-value below 0.05 that states that the difference in the expression level of a gene gives us enou
that null hypothesis using a Student's t-test (using a t-statistic), we will say that a gene is differentially expressed.

When thinking about statistical significance, we must also consider how many tests are being conducted. Given
differential expression analysis we are doing thousands of tests at a time (for thousands of genes), we have to ac
portion of tests are going to come up positive by random chance since the more tests we do, the more likely it is
value < 0.05 for one of the genes. Adjusting for this is called multiple hypothesis correction.

This video helps to explain this problem by using a simple example of rolling a dice:

Video. The Multiple Comparisons Problem.
This video explains multiple hypothesis correction and adjusting for random chance.
View Transcripts. (https://canvas.asu.edu/courses/122165/files/54792252?wrap=1) **(https://canvas.asu.edu/courses/122165/files/54792252/download?download_frd=1)**

Our code applies a Benjamini Hochberg adjustment in the decideTests function when reporting differential expres
chose an adjusted p-value < 0.05 as our threshold for differential expression.

Visualize the full set of differentially expressed genes

A volcano plot is a scatter plot of all the genes by their p-value and fold change with thresholding that allows you
genes are characteristic of females versus males.

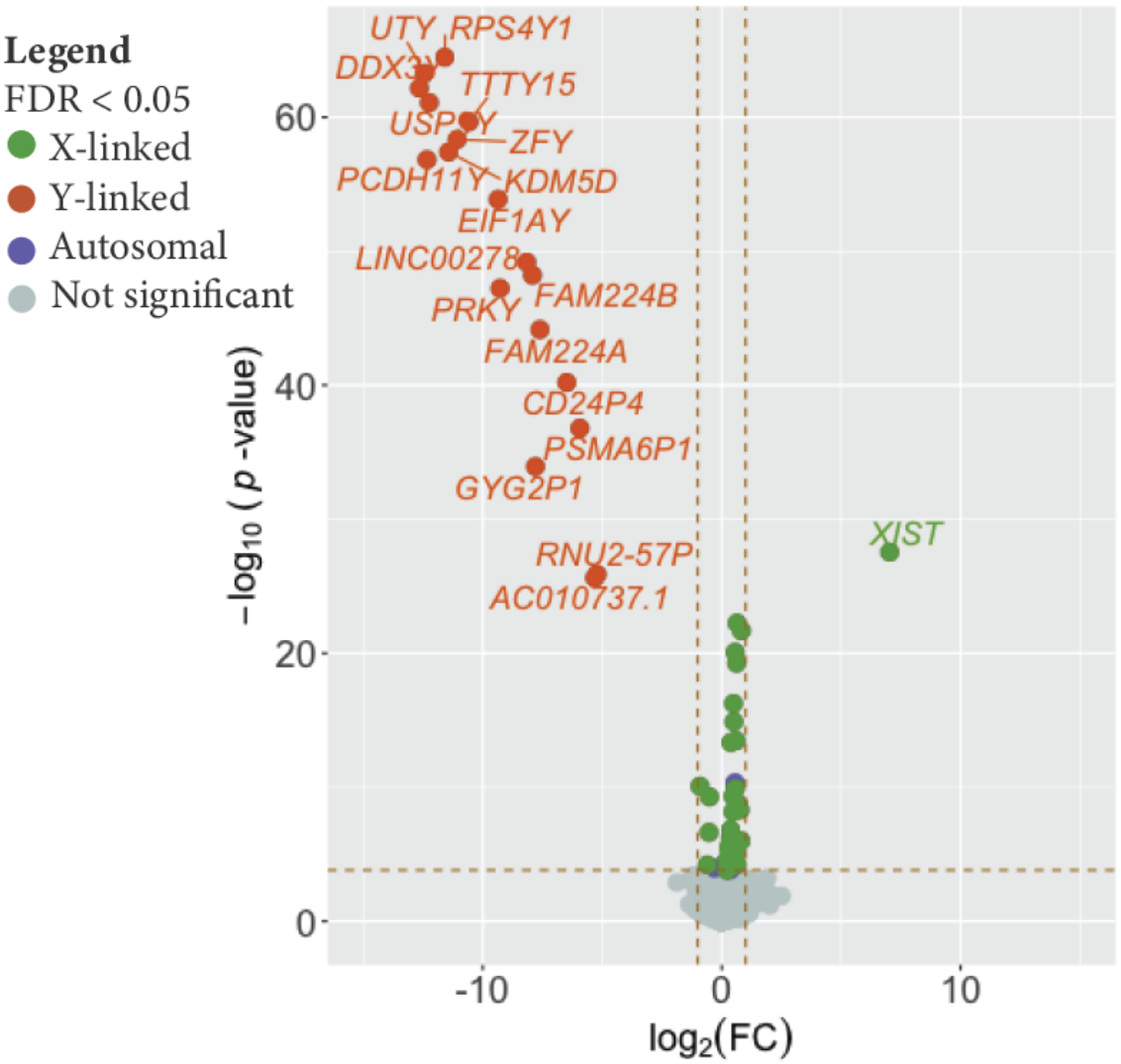


Figure. Volcano Plots.

Volcano plot of sex differences in placenta gene expression from the original manuscript.

To make p-values easier to display, the volcano plot takes the negative log10, pulling “good p-values” away from values are shown along the y-axis of a volcano plot, showing genes like UTY with a negative log10 p-value of ab that the chances of gene expression of UTY in females and males representing one group is 10^{-60}, so it is much, UTY expression is distinctly different in females than in males.

Log-transformed fold changes are on the x-axis of a volcano plot. You can see the gene XIST as having a log2 f
females to males, meaning the fold change is 2^8 or 256 times higher in females than in males. Not surprisingly, t
that passes the threshold for significance we set. This also makes sense given that this gene is involved in X chr
which only happens in XX individuals (females). On the other side, we see many genes that have high negative
indicating higher expression in males, that also have log p-values that pass the threshold for significance. By usir
pull out and label differentially expressed genes with large differences in expression level in a volcano plot.

Visualize gene expression differences for specific genes

Once we get our list of differentially expressed genes, we often want to visualize the expression differences for in
find interesting. These genes can be selected based on the data, such as the gene with the highest difference in
gene expression. These genes can also be selected based on what is known about the biology of the samples y
as genes that mark specific cellular processes in the placenta for which we see differential expression by sex usi
many plots that are used to compare values in sample groups; we have detailed some popular ones we have use
Typically, several plots are placed side by side to compare the distributions of data points among several groups.

Box plot

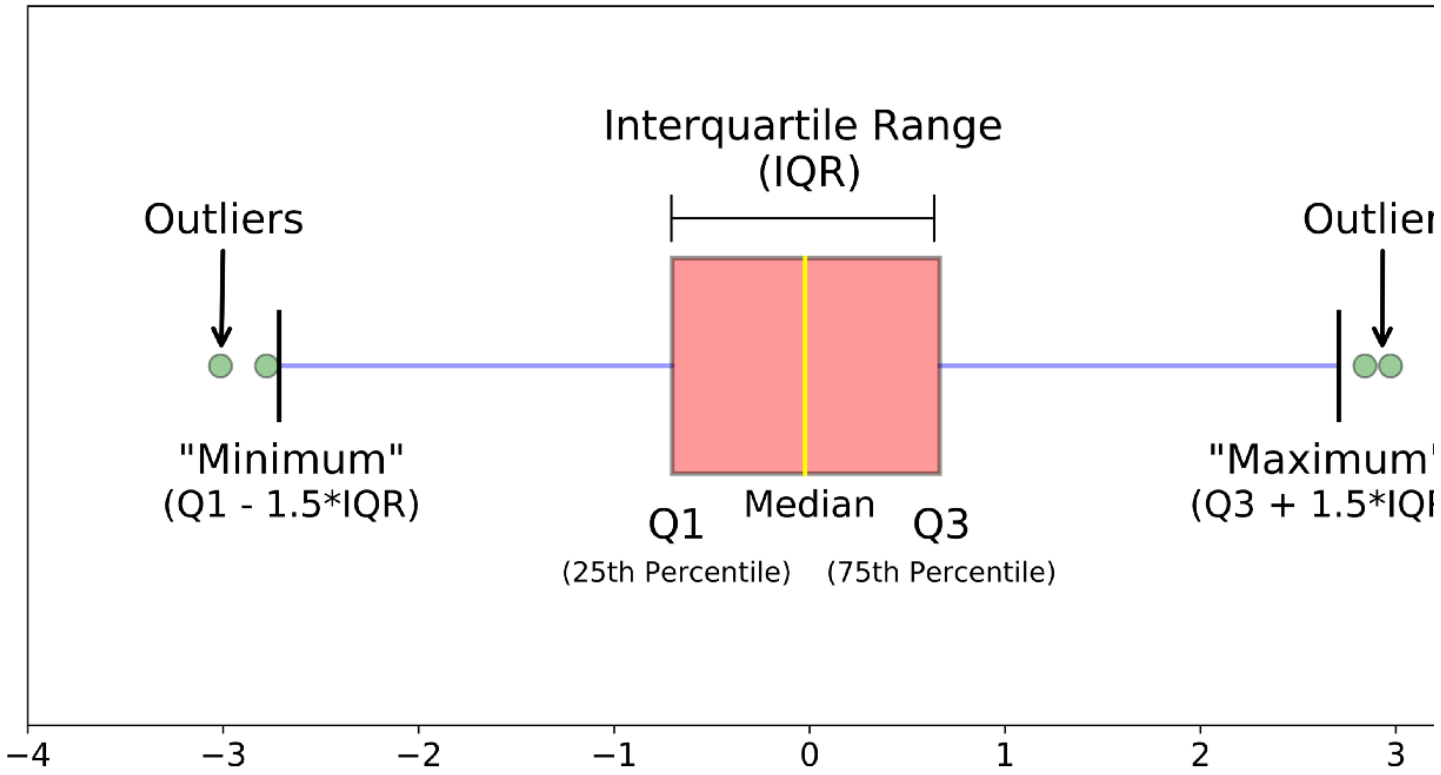


Figure: Parts of box plot.

From left to right: outliers, minimum, Q1, median, Q3, maximum, outliers.

[Source: Towards Data Science, Understanding Boxplots]

A box plot displays a 5 number summary of the distribution of a specific measurement across a set of samples. A median of the values. A box is drawn between the 25th percentile (Q1) and the 75th percentile (Q3). Lines calle to show the distance between the minimum and maximum values of the range. In some cases, whiskers are dra minimum and maximum, while in other cases, outliers are detected and the minimum and maximum are determir those outliers. This type of visualization presents the distribution of the values without showing the specific obse sample. This visualization is often used when sample numbers are very large and values are evenly dispersed th between the minimum and maximum of the distribution.

Violin Plot

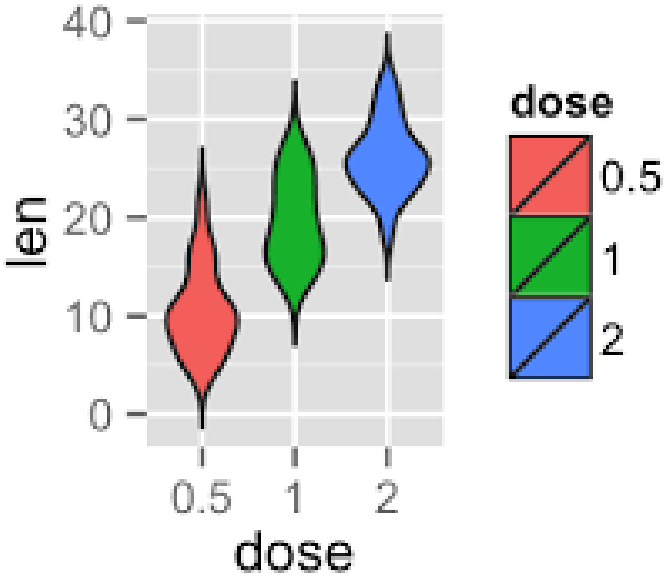


Figure. Violin plots.

Violin plots showing the length of teeth after a dose of drug treatment

[Source: **STHDA, ggplot2 violin plot :Quick start guide - R software and data visualizatio**
(http://www.sthda.com/english/wiki/ggplot2-violin-plot-quick-start-guide-r-software-and-data-visualiz

Violin plots are similar to box plots in that they show the distribution, but the width of the plot is smoothed by a ke
which shows the range at which there are many samples data points as wider. The example above shows only t
plots can also be marked with the median and percentiles.

Jitter Plot

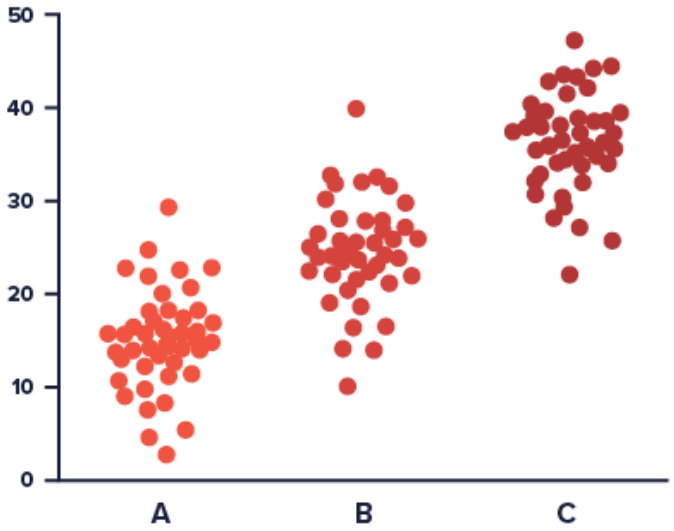


Figure. Jitter plots.

Example of a jitter plot visualizing distribution and showing individual values.

[Source: **datavizproject, Jitter Plo (https://datavizproject.com/data-type/jitter-plot/) t**]

A jitter plot is used to visualize the distribution and show the individual values, instead of just the summary of the values are plotted as dots along one axis and the dots are then shifted randomly along the other axis, which has data-wise, allowing the dots not to overlap.

Combination Plots

Depending on what you want to convey with your visualizations, different kinds of plots are often combined to sh of the distribution of values within a group. You can mix and match easily using the `ggplot` function in R as you

section of this module.

Violin box plot

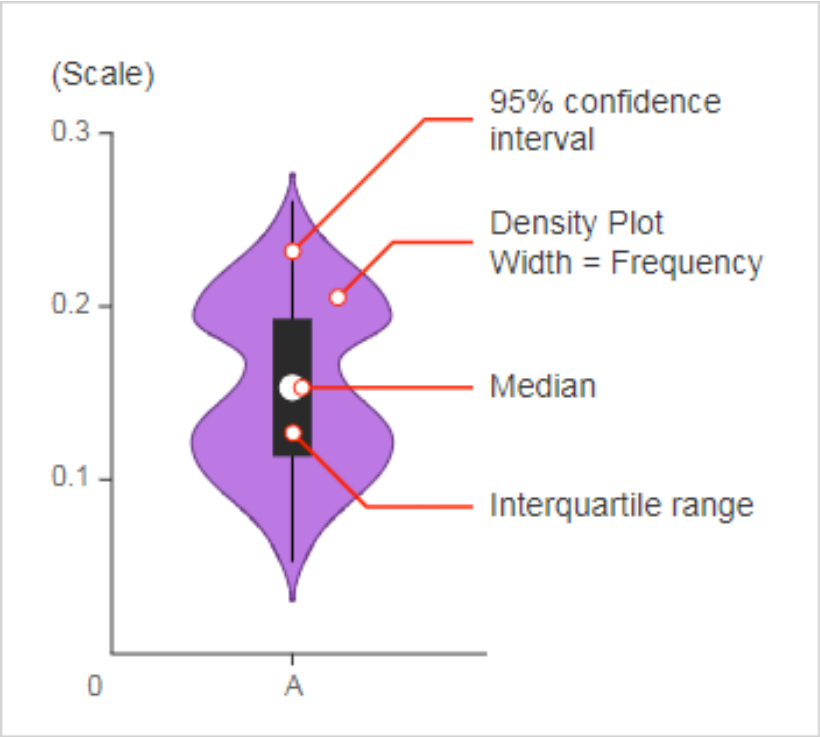


Figure: Parts of a violin plot.

From the top to bottom: 95% confidence interval, frequency, median, interquartile range.

[Source: **Infinity Insight, Violin Plots: What They Are and Why You Should Care (https://datavizcatalogue.com/methods/violin_plot.html)**]

Violin jitter box plot

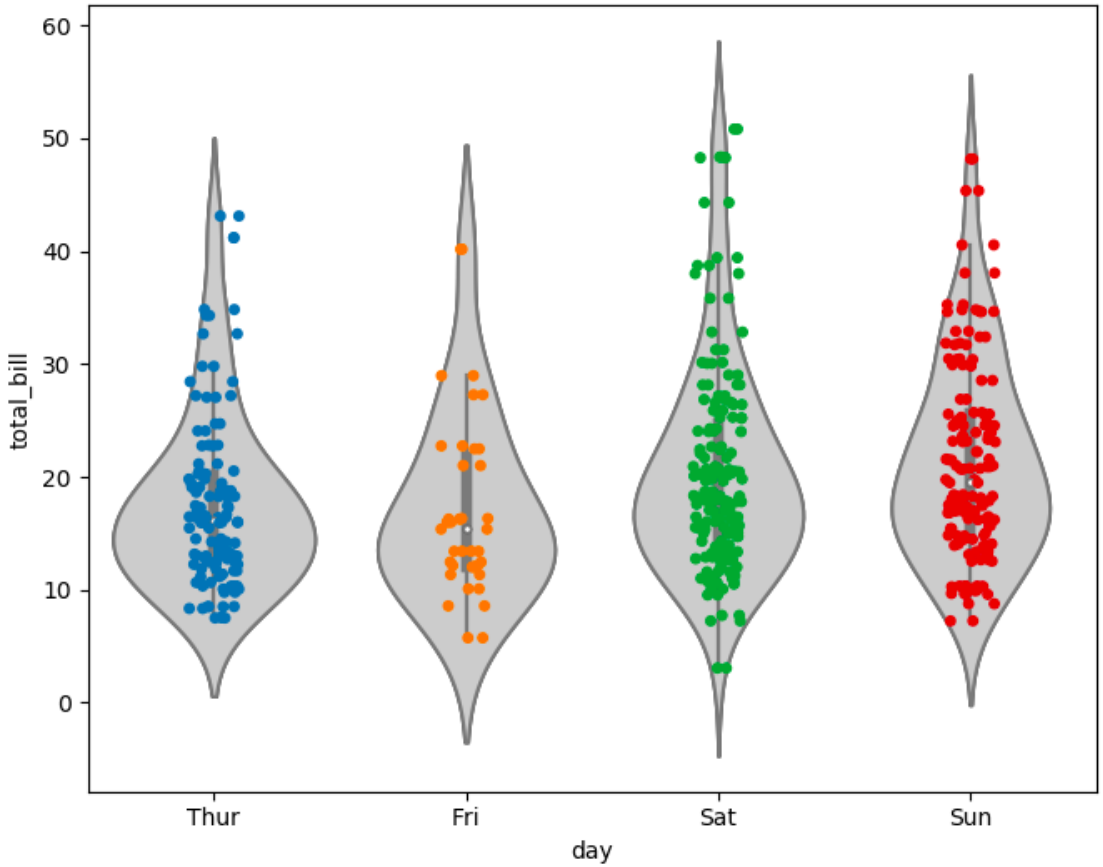


Figure. Violin plot with box plots and jitter plot added in.

This plot has the distribution of the jitter plot with the density of the violin plot.

[Source: **Stack Overflow** (https://stackoverflow.com/questions/55797760/seaborn-stripplot-with-violin-plot-ba

Expected effects of trimming

Now that we have learned about differential gene expression analysis, we can talk about what we might expect t project which aims to study the effect of trimming on this process. To remind you, our experimental design has g processed 10 data sets which will give us gene counts to pass into the differential expression pipeline:

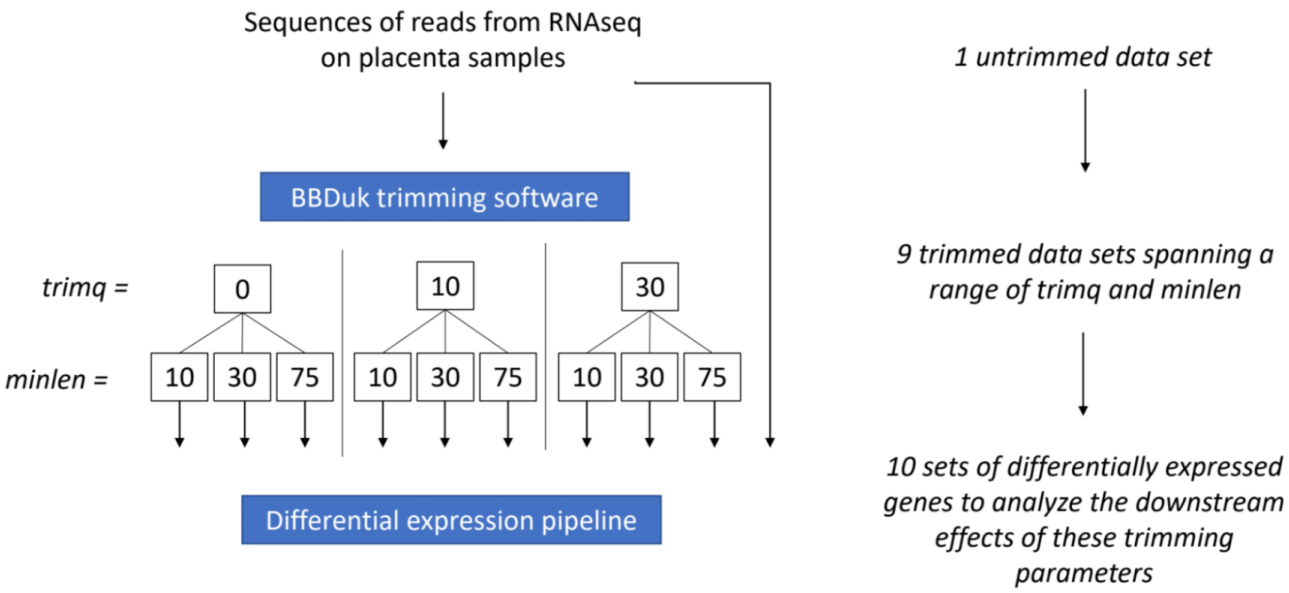


Figure. Experimental design of trimming parameters.
We will be using one (1) untrimmed dataset and nine (9) trimmed datasets with a range of parameters to generate ten (10) sets of differentially expressed genes to analyze.

This week we will all do differential expression analysis on the untrimmed data set. This will allow you to see the ask your instructors and classmates for help if anything is confusing. The code has all the parts of the analysis y this section as well as places that you can run analysis for specific genes of interest as you explore the results yc you to build on the code we have provided.

In the next module, you will run a specific trimmed data set through the differential expression pipeline. What diffe How do we measure those differences? Here are some ideas to get you started, but we would like for you to thinl and ways to answer them as well:

1. Do we see more or less differentially expressed genes after trimming?
2. For the differentially expressed genes observed:

    - Do we see more or less difference in average expression in males (fold change)? Females?
    - Do we see a change in the number of upregulated or downregulated genes?
3. Which genes are common to both untrimmed and trimmed conditions? Which are unique?
4. Can we figure out something shared among the genes that are specific to trimmed?
    - Do they all have relatively low expression? Relatively high expression?
    - Do they all have a similar function in the cell?
5. Does trimming have the same effect on genes upregulated in females and genes upregulated in males?

To be continued… Put on those thinking caps…

To be clear, it might be that you don't see any change or maybe you will see lots of changes. In research, our job to the question we are asking, no matter what that answer is. If we do see differences with trimming, we can ana differences. If we do not see any changes with trimming, we can think about why that is. Is it because the data v begin with? Maybe trimming with more strict parameters is necessary to see an effect? We will start by comparin with the results from one specific trimming parameter data set; then you will be comparing results from all the trin collaboratively using results from your classmates. By the end, we should be able to analyze the behaviors over trimming parameters we varied (trimq and minlen).

Module 3.1 Additional Resources

- Workshops on RNAseq normalization methods
    - **Introduction to DGE (https://hbctraining.github.io/DGE_workshop/lessons/02_DGE_count_normalizatior**
    - **Tutorial on RNASeq Normalization and Differential Expression (http://computationalgenomics.bioinformatics.ucla.edu/portfolio/jo-hardin-tutorial-on-rnaseq-normalization-a expression/)**
- **Bioconductor limma workflow (https://bioconductor.org/packages/release/workflows/vignettes/RNAseq123/inst/doc/limmaWorkflow.html)**