**Methods**

*Strains, RNA extraction, and sequencing*

The strains used were obtained from previous mutation accumulation experiments, one from a hybrid of NCYC 3631, a Mat alpha derivative of YPS 606 (oak strain, PA, USA), and NCYC 3596, a Mat a derivative of DBPVG1106 (lici fruit, Indonesia, wine clade) (unpublished data). This hybrid strain was highly heterozygous, with a heterozygous site every ~250 bp (unpublished data). The other strain used was a lab strain from a different mutation accumulation experiment and had the genotype *ade2, lys2-801, his3-∆D200, leu2-3.112, ura 3-52,* and was homozygous at all loci except the mating type locus (Zhu *et al.* 2014). Both strains from the separate mutation accumulation experiments were diploid and put through a bottleneck every ~20 generations for ~2000 generations.

30 lines from both experiments (12 aneuploid hybrid ancestor, 8 euploid hybrid ancestor, 6 aneuploid lab strain, 2 euploid lab strain), 3 replicates of each, were inoculated in 3ml liquid YPD (yeast extract 2%, peptone 2%, glucose 2%), and incubated on a rotator at 30° C for 24 hours. RNA was extracted using the MasterPure Yeast RNA Purification Kit (Epicentre). Integrity, concentration, and quality of RNA samples were assessed using the Qubit. Libraries were prepared using the Illumina Stranded RNAseq Kit. They were sequenced at the Georgia Genomics Facility on the Illumina NextSeq (75 cycles) SE75 High Output flow cell. Samples were multiplexed and split across two runs.

For the previously sequenced RNA, we selected 45 colonies (3 replicates each of 15 lines), inoculated each into 3 ml liquid YPD medium, and incubated on a rotator at 30 for 24 hours. After 24 hours, mRNA was extracted using the MasterPure Yeast RNA Purification kit (Epicentre). RNA libraries were constructed using the Illumina Tru-seq mRNA Stranded Kit, amplified using 13 cycles of PCR and sequenced on an Illumina HiSeq 2500.

Because these datasets were extracted by different individuals, libraries were made with different kits, and the samples were run on different sequencing machines, analyses were carried out separately, although identically.

*RNAseq data analysis*

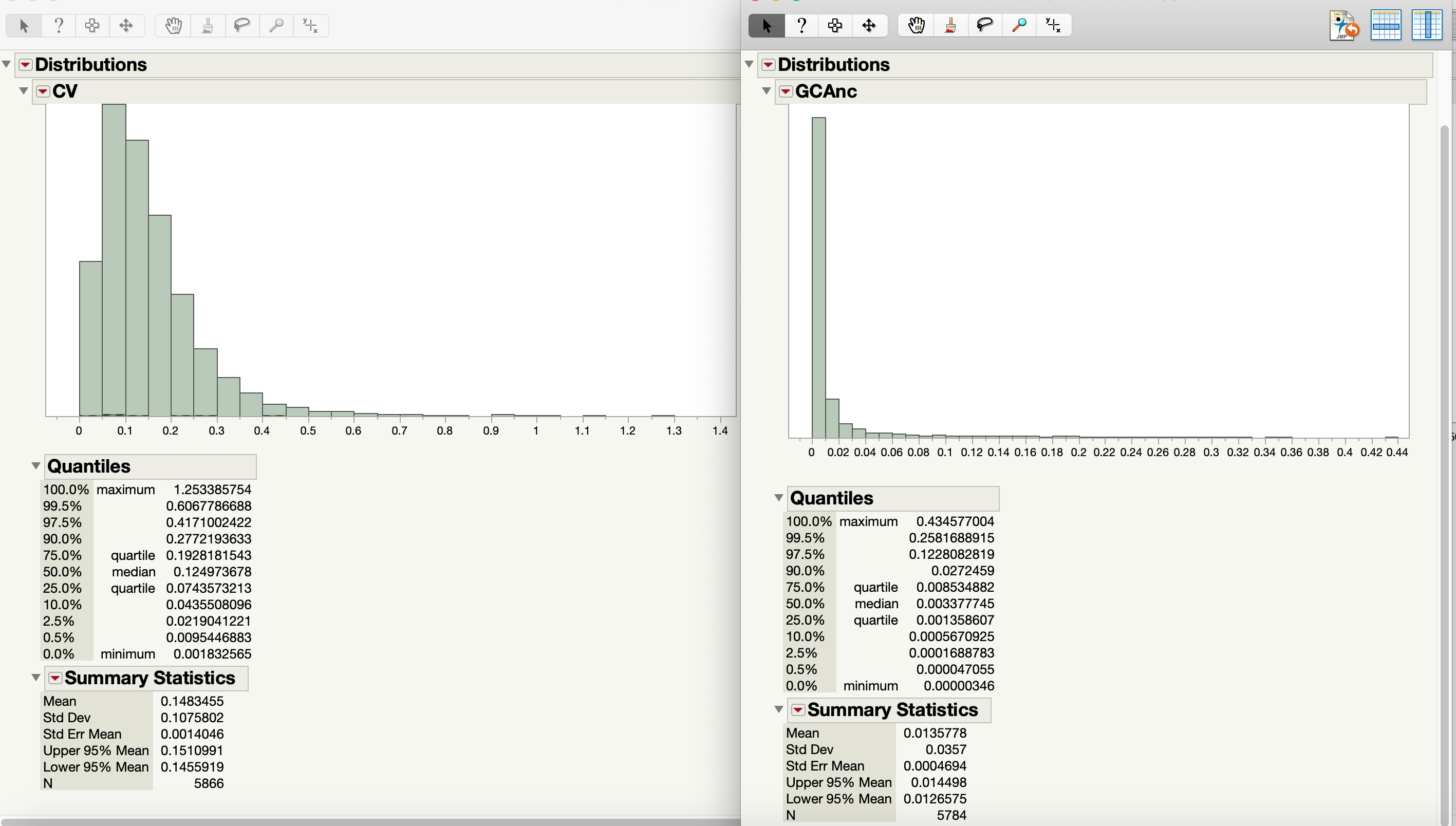
Raw data was obtained from the Georgia Genomics Facility, with adapters removed. Quality control was done using FastQC version 1.8.0\_20 using default parameters (available at <https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>). Low-quality bases were trimmed using Trimgalore v. 0.4.4 using -phred 33, -q 20 (available at https://www.bioinformatics.babraham.ac.uk/projects/trim\_galore/). RNA samples were aligned to the reference genome and annotated transcripts (obtained from <https://support.illumina.com/sequencing/sequencing_software/igenome.html>) using Tophat v. 2.1.1 with -i 10 -I 10000 (Trapnell *et al.* 2012b). Cufflinks v. 2.2.1 was used to assemble the transcriptomes using default parameters (Trapnell *et al.* 2012a). Normalization was done using Cuffnorm v. 2.2.1 with default parameters, differential expression was found using Cuffdiff v. 2.2.1 with defult parameters, and read counts were found using HTseq v. 0.6.1pl (Python v. 2.7.8) (Anders *et al.* 2015). Samtools (version 1.3.1) was used to convert .sam files into .bam files and sort the resulting .bam files (Li *et al.* 2009). Scripts can be found at <https://github.com/hollygene/Dosage_Compensation/blob/master/assembly_script.sh>.

Genes were filtered to eliminate those genes with high variance. edgeR (CITE) was used to calculate counts per million, then filtered based on high variance. The ancestral replicates were used to find the highly variable genes. It was found that the majority of the genes with high variance also had a cpm < 1, so those genes were removed from the following analyses. The resulting list of genes were used in all subsequent analyses. Approximately 1300 (~1.8%) genes were removed from the analysis.

Cuffnorm was used to find FPKM (fragments per kilobase per million reads). A homebrew bash script was used to join the FPKM values for each line with the gene attributes file, turn the file into a .csv, remove mitochondrial sequences, and change the chromosome names from Roman numerals to numbers (script can be found at <https://github.com/hollygene/Dosage_Compensation/blob/master/DC_workflow_April2017.sh>). This data file was then read into the R statistical software (Team 2013). Average FPKM between the replicates for each line at each chromosome was found, followed by the average FPKM ratio (average FPKM of descendent line/average FPKM of ancestral line). Boxplots and ANOVAs were done in R. Comparisons made between aneuploid lines and the expected log2 ratio of the gene expression of trisomic, monosomic, or tetrasomic chromosomes were made using a one-sample t-test with mu = log2(1.5) for trisomic, mu = log2(0.5) for monosomic, and mu = log2(2) for tetrasomic. Two separate, but identical, analyses were done for each of the datasets. R scripts are located at <https://github.com/hollygene/Dosage_Compensation/blob/master/DC_workflow_old_MA.Rmd> and <https://github.com/hollygene/Dosage_Compensation/blob/master/DC_workflow.Rmd>.

Individual genes were analyzed for differential gene expression using the edgeR, voom, and limma workflow from (Law *et al.* 2016), with modifications to fit this dataset. The eBayes fitted model was used, and differentially expressed genes were found by comparing each line to its corresponding ancestor.

\*This is as far as I’ve gotten, I’ll add more when I’m done with the individual gene analysis stuff!



Left: Coefficient of variance in GC ancestor BEFORE filtering out genes that are less than 1 in all ancestor replicates

Right: Coefficient of variation in GC ancestor AFTER filtering out genes that are less than 1 in all ancestor replicates

Analyses done/methods used:

Individual Genes

* Differentially expressed genes in aneuploids – on aneuploid chromosome
  + And shared between different aneuploids
  + Have csvs and Venn diagrams
  + GC
* Non-differentially expressed genes in aneuploids – on aneuploid chromosome
  + Csvs and Venn diagrams
  + GC
* Differentially expressed genes in euploids
  + CSVs and Venn diagrams
  + And shared between different euploids
  + GC

DE Genes

* # common DE genes – All samples
* DE genes – all lines:
  + Aneuploids
    - Shared DE genes between samples aneuploid for same chromosome
    - # common DE genes all aneuploids (CSV)
  + Euploids
    - Shared DE genes (all euploids) CSV and Venn
    - # shared DE genes
    - DE genes for each sample (.txt)
* GC

Non-DE Genes

* % nonDE – all lines
* Aneuploids:
  + Non DE genes per sample
  + # non DE genes per sample (CSV)
  + % non DE genes per sample (CSV)
  + Common nonDE genes on same chromosome (CSV and Venn)
* GC

GO Analysis

* Common DE genes Euploids
* GC

Dosage-sensitive genes/DE/NonDEGenes

* nonDE, dosage-sensitive genes on aneuploid chromosome
* GC

Whole-Chromosome Analysis

* Graph of chromosome by log2 fold change (1 per sample)
  + GC
  + MA
* Graph of FDR by ancestor FPKM (1 per sample)
  + GC
  + MA
* Graph (boxplot) of line by log2 ratio (1 per chrom)
  + GC
  + MA
* Graph (scatterplot) of total # genes by # DE genes (1 per sample)
  + GC
  + MA
* Graph of line by log2 ratio (1 per chrom)
  + GC
  + MA
* Graph of Tukey test (each line compared to each other) (1 per chrom)
  + GC
  + MA
* Graph of chrom by log2fold change (1 per sample)
  + GC
  + MA
* edgeR pairwise comparison (each sample by ancestor) .csv
  + GC
  + MA

Individual Genes /Dropbox/McQueary/Dosage-Compensation/Indiv\_Genes

Go Analyses

* nonDE genes on aneuploid chromosomes (Chr I and V)
  + GC
* PCAExplorer
  + GC
* DE genes:
  + Common DE genes in euploids (# and Venn)
    - GC
  + Common DE genes in aneuploids
    - GC
    - Same chrom
    - All aneuploids
    - Per aneuploid on aneuploid chrom
  + Common DE genes in all lines
    - GC
* Non DE genes:
  + % nonDE genes in all lines
    - GC
  + Non DE genes on aneuploid chrom
    - GC
    - Er line
    - Per aneuploid chrom (shared between lines)
    - Percentage per line
* Cuffdiff
  + All cuffdiff output files
    - GC: 7, 31, 21, 11
  + List of commonly sig DE genes
    - GC
  + .diff files
    - GC
    - MA
* Old R Files
  + MA Plots
    - GC
    - MA
  + Ratio of p-values to mean normalized count
    - GC
    - MA
* HTSeq
  + Exons\_unstranded
    - GC
    - MA new
  + Glimma plots (MDS plot)
    - GC
* Variance
  + MA plots
    - GC
  + P-value histograms
    - GC
  + Mean-variance relationship
    - GC
  + Distribution of variance
    - GC
  + Boxplots per chrom
    - GC
    - MA
  + P value versus coefficient of variation
    - GC
* edgeR exactTests (sample vs ancestor)
  + GC
  + MA
* MD plots
  + GC
  + MA new
* List of low expressed genes
  + GC
  + MA
* List of high expressed genes
  + GC
  + MA

Update 2/5/2019

* Getting rid of counts per million <1 in the ancestor replicates
  + make new data frame with only the ancestor replicates
  + use rowSums(anc>1)==3
    - How many rows are greater than 1 in all 3 replicates
  + GC: 5799 genes (lost 1331 genes)
  + MA New: 5767 (lost 1363)
  + MA Old: 5798 (lost 1332)
* If you then get rid of average cpm in ancestor <1, go to 5784 genes

2/6/2019

* Filtering in DESeq2 using similar (same) criteria as in edgeR
  + Only keep genes where there is more than 10 COUNTS in the ancestor (across all replicates)
  + 5804 genes in GC (1322 excluded)

2/8/2019

* DESeq2 reanalysis
* Loading in ALL datasets – MA new, MA old, GC
* Filtering genes as above:
  + Table(rowSums(ancData>10)==3)
    - For each gene, find how many replicates of ancestor counts are greater than 10
    - Only keep those genes that have counts > 10 in all 3 ancestor replicates
* Moving onto results with new (shortened # of genes) dataset

|  |  |  |  |
| --- | --- | --- | --- |
| Line | LFC > 0 | LFC < 0 | Ploidy |
| 2 | 1144 | 1029 | Euploid |
| 7 | 0 | 1 | 3n for Chr I |
| 18 | 323 | 183 | 3n ChrI and ChrXII |
| 11 | 337 | 108 | 3n chr XV, 1n chr I |
| 1 | 225 | 40 | Euploid |
| 3 | 1142 | 1010 | Euploid |
| 4 | 501 | 143 | 3n chrV |
| 49 | 88 | 33 | 3n chrV |
| 59 | 227 | 26 | 3n Chr VII |
| 61 | 307 | 34 | 3n chr VII |
| 8 | 1496 | 904 | 3n Chr XVI |
| 76 | 297 | 17 | 3n chr IX, partial chr XIV |
| 77 | 212 | 39 | 3n chr XII |
| 5 | 1480 | 1520 | euploid |

2/11/2019

* Today’s goals:
  + Find common DE genes in euploid lines
    - Run results() on euploid lines only first
    - Run my commonDEGenes function on those
  + Remove those common DE genes from dataset that will be used for finding DE genes in euploids
  + Find DE genes in aneuploid lines using shortened dataset
  + Find out what is the average read depth in lines that are aneuploid but have low #s of DE genes
    - For those lines, if you only include the aneuploid chromosome(s) in the test, what happens?

2/13/2019

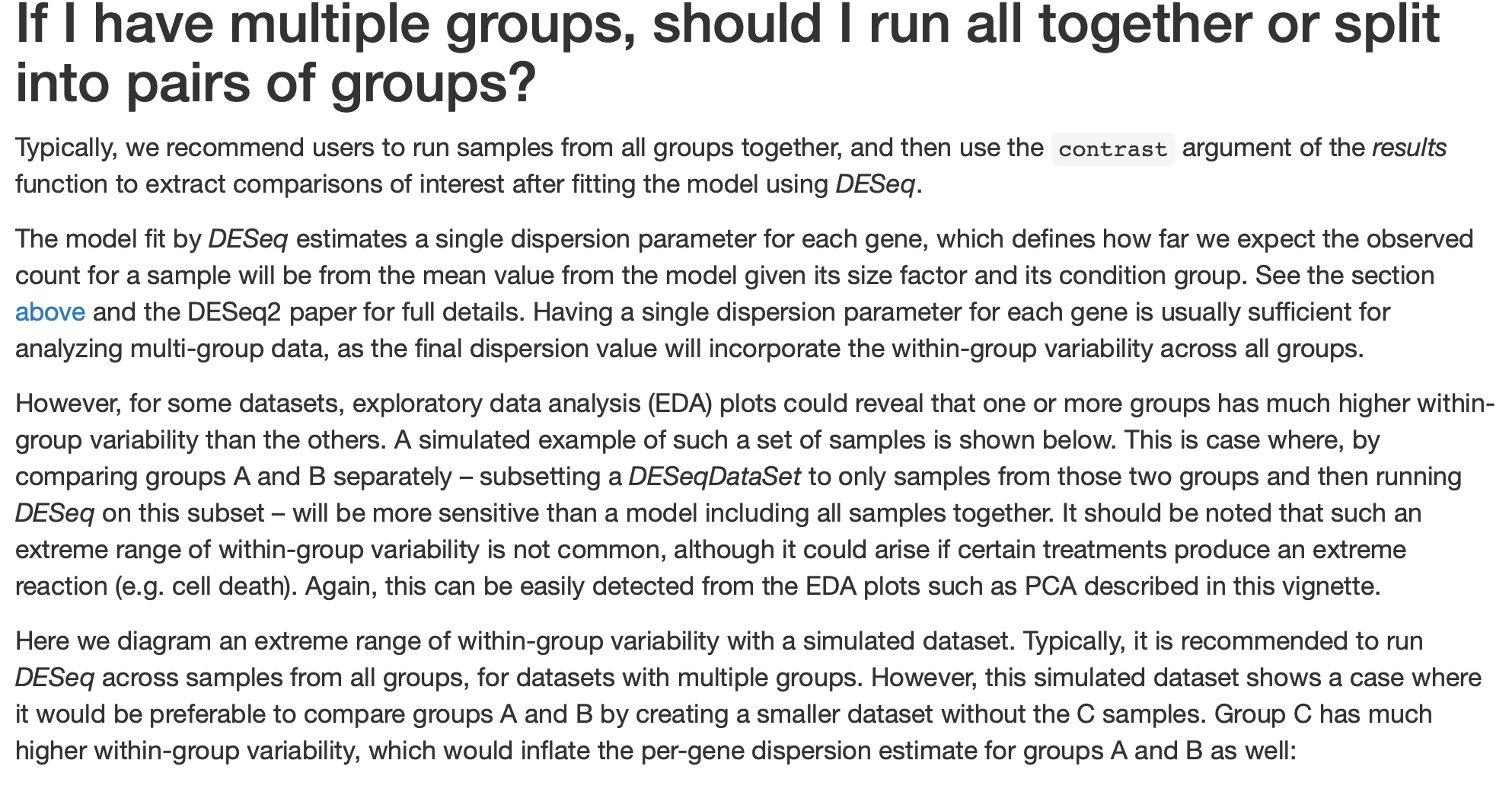
* Looked at FASTQC results for every line
  + Looked fine, no indication that lines with low DE genes were of poorer quality than lines with high DE genes

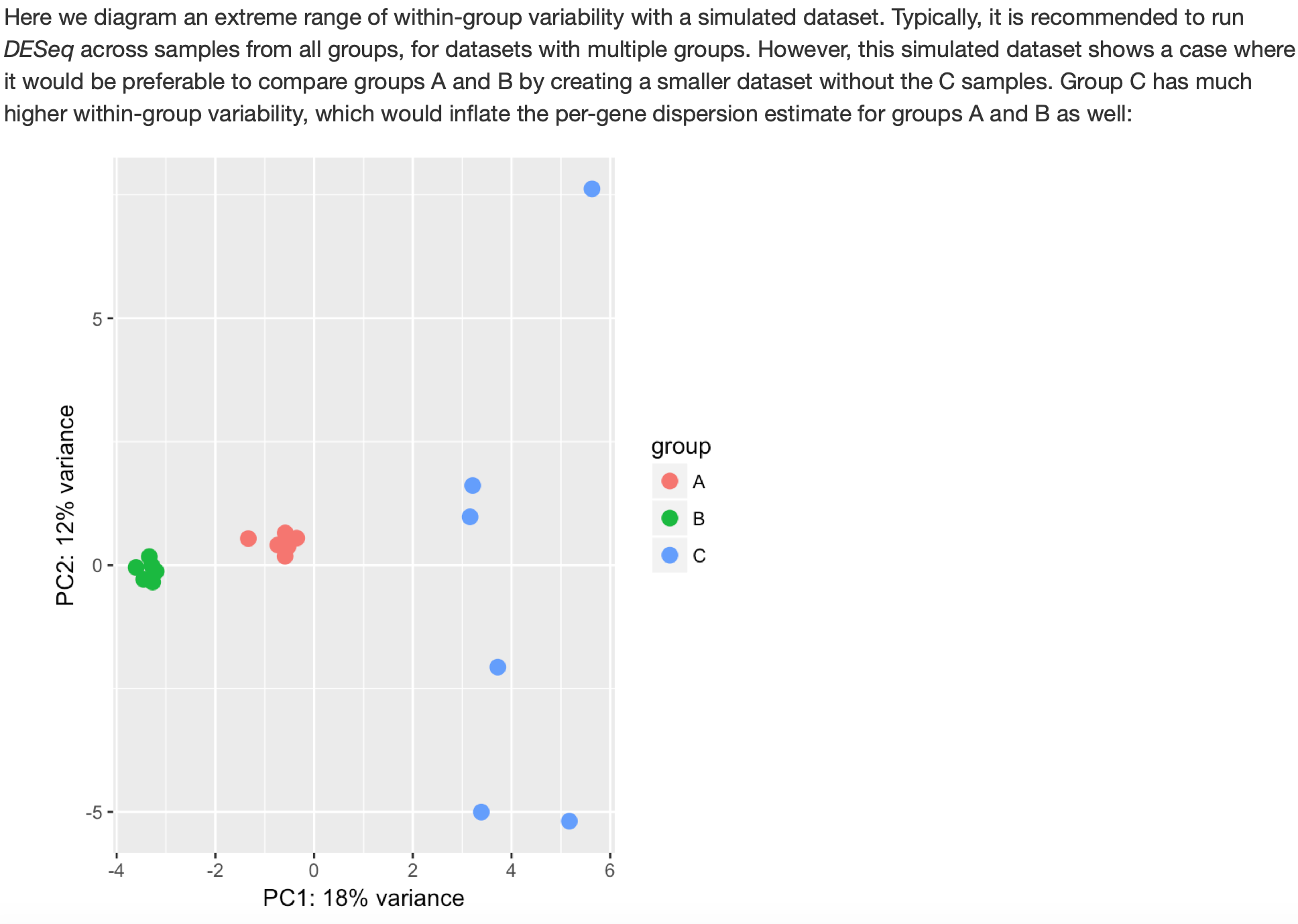
2/14/2019

* Removed those genes that had high variance and low counts from original count data
* Results without said genes:

|  |  |  |  |
| --- | --- | --- | --- |
| Line | LFC >0 | LFC < 0 | Ploidy |
| 1 | 5 | 0 | Euploid |
| 2 | 372 | 332 | Euploid |
| 3 | 525 | 359 | Euploid |
| 5 | 963 | 956 | Euploid |
| 9 | 24 | 7 | Euploid |
| 4 | 56 | 6 | Aneuploid |
| 7 | 0 | 1 | Aneuploid |
| 8 | 884 | 619 | Aneuploid |
| 11 | 157 | 76 | Aneuploid |
| 18 | 66 | 9 | Aneuploid |
| 49 | 52 | 13 | Aneuploid |
| 59 | 33 | 1 | Aneuploid |
| 61 | 125 | 2 | aneuploid |
| 76 | 172 | 17 | Aneuploid |
| 77 | 113 | 27 | Aneuploid |

2/15/2019

* Goal: plot mean of ancestor by mean of lines
  + MA plot?
  + Plot counts
* Question: if I have multiple groups, should I run together or split into pairs of groups?
  + Exploratory data analysis plots could indicate that one or more groups has much higher within-group variability than others
    - Can be easily detected from EDA plots such as PCA
    - 
* Next: run PCA on samples
  + Vst (variance stabilizing) and rlog (regularized logarithm) transformations



* Do analyses individually and see if results are any different

GC Lines, After Running them all Individually

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Line | LFC >0  p<0.1 | LFC >0  p<0.05 | LFC <0  p<0.1 | LFC <0  p<0.05 |
| 1 | 1138 | 927 | 1280 | 1105 |
| 2 | 775 | 612 | 774 | 588 |
| 3 | 382 | 298 | 317 | 247 |
| 4 | 569 | 490 | 574 | 469 |
| 7 | 254 | 178 | 211 | 148 |
| 8 | 1159 | 986 | 1223 | 1027 |
| 9 | 30 | 9 | 116 | 63 |
| 11 | 26 | 10 | 40 | 23 |
| 18 | 601 | 445 | 662 | 507 |
| 49 | 44 | 28 | 28 | 14 |
| 59 | 492 | 361 | 513 | 368 |
| 61 | 428 | 314 | 357 | 270 |
| 76 | 294 | 224 | 87 | 59 |
| 77 | 367 | 256 | 334 | 232 |

2/16/2019

* Running DESeq2 on MA old and MA new individually and comparing to initial results of all of them together

MA Old

Before: all lines ran together

After: all lines ran separate

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sample | LFC >0  Before | LFC > 0  After | LFC <0  Before | LFC <0  After |
| 1 | 152 | 43 | 60 | 12 |
| 2 | 58 | 131 | 51 | 173 |
| 3 | 146 | 216 | 196 | 389 |
| 4 | 751 | 497 | 679 | 427 |
| 5 | 331 | 250 | 210 | 143 |
| 6 | 5 | 8 | 3 | 15 |
| 7 | 64 | 95 | 34 | 77 |
| 8 | 2 | 40 | 3 | 77 |
| 9 |  | 610 |  | 487 |
| 11 | 4 | 58 | 2 | 31 |
| 15 | 106 | 203 | 21 | 56 |
| 28 | 199 | 527 | 179 | 496 |
| 88 | 297 | 523 | 301 | 610 |
| 108 | 1557 | 1608 | 1709 | 1623 |
| 119 | 147 | 75 | 35 | 25 |

MA New

Before: all lines ran together

After: all lines ran separate

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sample | LFC > 0  Before | LFC >0  After | LFC <0  Before | LFC < 0  After |
| 29 | 147 | 220 | 133 | 255 |
| 50 | 397 | 925 | 164 | 674 |
| 112 | 65 | 13 | 4 | 4 |
| 115 | 0 | 255 | 0 | 134 |
| 117 | 92 | 622 | 6 | 453 |
| 123 | 48 | 513 | 7 | 508 |
| 141 | 14 | 11 | 1 | 1 |
| 152 | 2 | 5 | 0 | 2 |

Update 2/21/2019

* Found in the FAQ for DESeq2 that some samples in your dataset might have much higher variance than others, so it is probably best to analyze those separately from the rest of the dataset
* I did a variance-stabilizing transformation (vst) on my raw count data and then performed a principle components analysis (PCA)

GC Lines by Line



GC Lines by Sequencing Run



MA Old Lines By Line



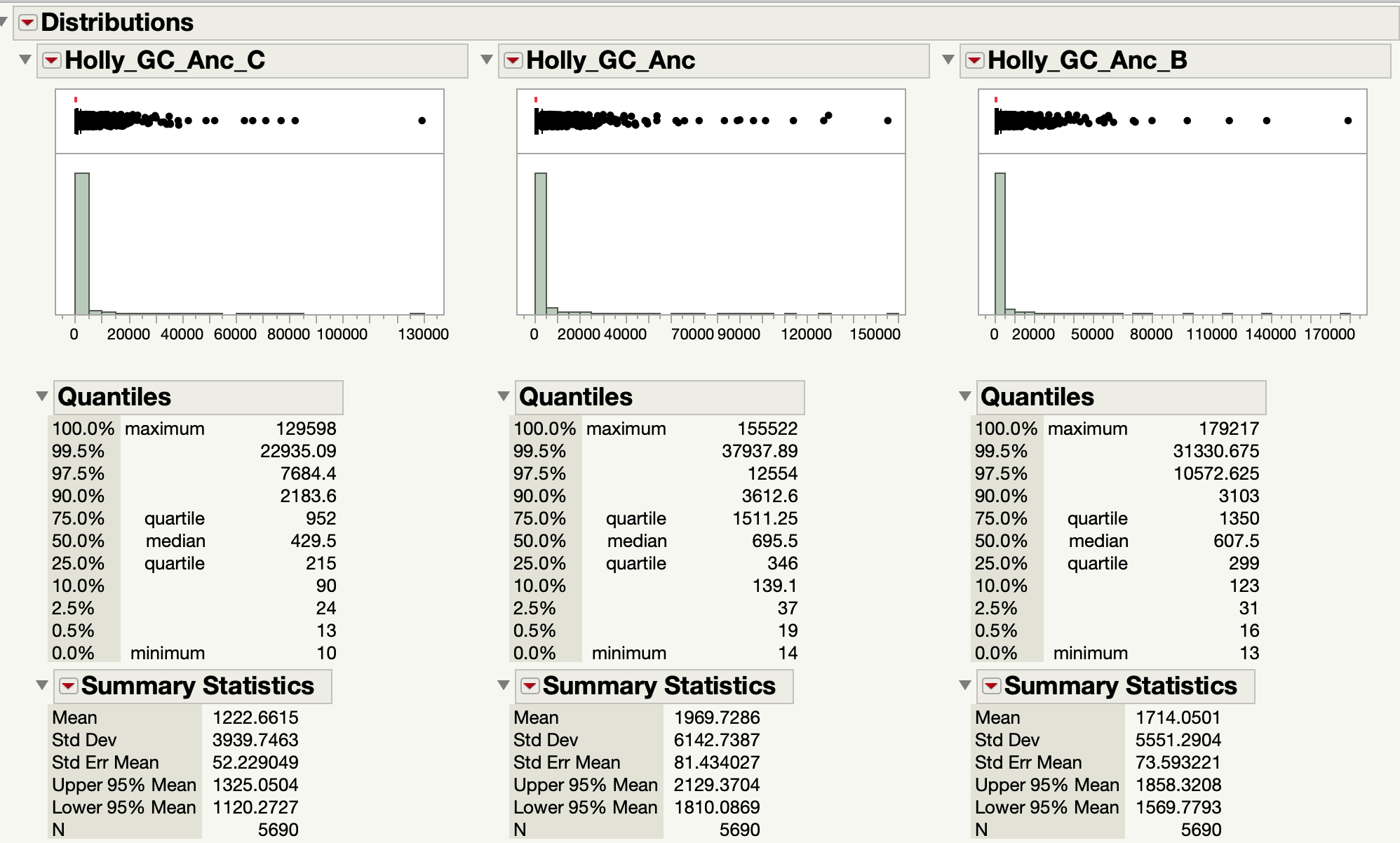
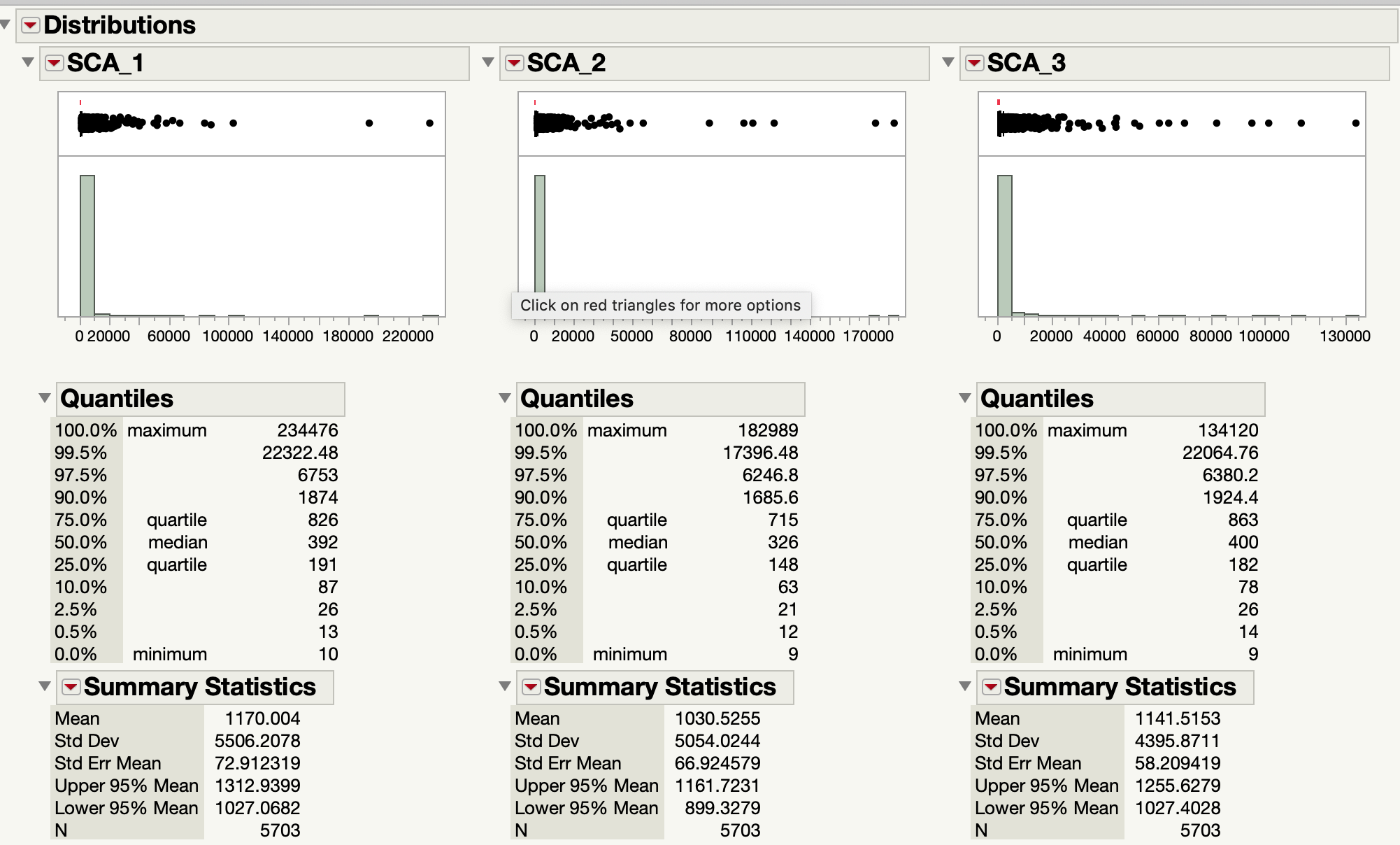
MA New Lines by Line



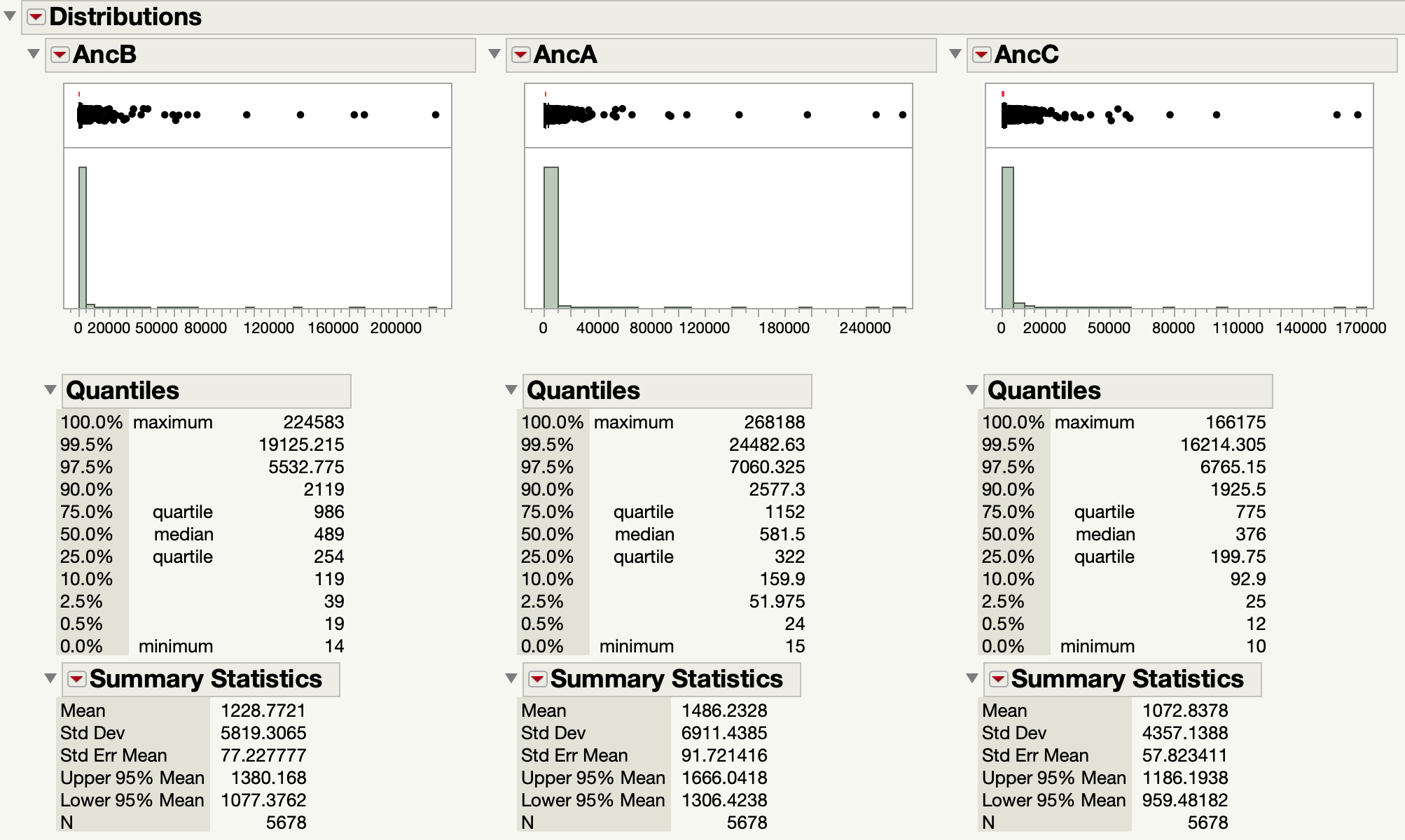
MA New Lines by Sequencing Run



2/21/2019

* Accidentally used UNFILTERED data for the above PCAs
* Went in and made sure that my FILTERED data was in the right place and CLEARLY LABELED
* Then loaded this data into R, and exported it back out of R to make sure it is the correct one
* Put it in a distribution in JMP
* GC Ancestor replicate count distributions
* MA Old Ancestor replicate count distributions
* 

MA New Ancestor Replicates Count Distributions



4/2019:

* Made distribution of normalized read counts (normalized using DESeq2) for all samples for cis (on aneuploid chromosome) and trans (not on aneuploid chromosome) genes as well as for the euploid samples (all genes)
* Performed a test to parse out significant genes:
  + Test 1: expression = expected dose
    - 1 x genes = -1 log2fold change
    - 2x genes = 0 log2fold change
    - 3x genes = 0.585 log2fold change
    - 4x genes = 1 log2fold change
    - p > 0.1: no DC
    - p < 0.1: Further tests needed:
  + Test 2: expression = expected dose of DISOMIC genes
    - Log2fold change =0
    - p < 0.1: partial/over/anti compensated
    - p > 0.1: fully compensated
* Compensation types:
  + No Dosage compensation: test against expected gene expression came back nonsignificant
  + Partial dosage compensation: test against expected gene expression came back significant, but test against expected dose of disomic genes came back significant AND the gene expression was skewed towards the disomic dose
  + Over dosage compensation: test against expected gene expression came back significant, but test against expected dose of disomic genes came back significant AND the gene expression was less than (or greater than for monosomic genes) expected for disomic genes
  + Anti dosage compensation: test against expected gene expression came back significant, but test against expected dose of disomic genes came back significant AND the gene expression was skewed in the opposite direction of dose expected of disomic genes
  + Full dosage compensation: test against expected gene expression came back significant, but test against expected dose of disomic genes came back nonsignificant
* Performing GO analysis on these genes ^
* Found gene expression values for ESR genes in all samples
* Found histone gene expression values compared to the ancestor in the samples that have aneuploid chromosomes with histone genes on them

Anders, S., P. T. Pyl and W. Huber, 2015 HTSeq—a Python framework to work with high-throughput sequencing data. Bioinformatics 31**:** 166-169.

Law, C. W., M. Alhamdoosh, S. Su, G. K. Smyth and M. E. Ritchie, 2016 RNA-seq analysis is easy as 1-2-3 with limma, Glimma and edgeR. F1000Res 5**:** 1408.

Li, H., B. Handsaker, A. Wysoker, T. Fennell, J. Ruan *et al.*, 2009 The sequence alignment/map format and SAMtools. Bioinformatics 25**:** 2078-2079.

Team, R. C., 2013 R: A language and environment for statistical computing.

Trapnell, C., A. Roberts, L. Goff, G. Pertea, D. Kim *et al.*, 2012a Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. Nature protocols 7**:** 562.

Zhu, Y. O., M. L. Siegal, D. W. Hall and D. A. Petrov, 2014 Precise estimates of mutation rate and spectrum in yeast. Proceedings of the National Academy of Sciences 111**:** E2310-E2318.