1. Quantify some transcripts using Kallisto
2. Download a gzipped tarball containing the reads from a yeast RNAseq experiment: . <https://www.dropbox.com/s/bfxsa324qrbg0st/yeast_replicates.tar.gz> Figure out how to extract the files from the .tar.gz file using the command line.
3. This is actually a very small amount of data from a huge dataset that was generated to test the impact of biological variation on RNAseq studies (<http://rnajournal.cshlp.org/content/22/6/839.full> ). The data consists of a whole lot of biological replicates from a wildtype yeast strain and a strain that has the gene SNF2 deleted. The file sample\_metadata.txt that was in the tarball tells you whether each sample is wildtype (WT) or SNF2 delete (SNF2). Figure out which of your samples are which.
4. Download Kallisto (<https://pachterlab.github.io/kallisto/> ).
5. Kallisto needs to have an indexed transcriptome. Luckily they provide you a yeast transcriptome on their website as a gzipped fasta. Find it and download it.
6. Index the transcriptome, using kallisto index,

kallisto index -i transcripts.idx transcripts.fasta.gz

1. Quantify expression for one of your samples using kallisto quant.

kallisto quant -i transcripts.idx -o --single -l 180 -s 20 reads\_1.fastq.gz

1. Take a look at the abundance.tsv file, which has the quantification for every gene.
2. Write a Python script using system calls to quantify ALL the samples.
3. Read data into R and test for differential expression using DESeq2
4. Install two bioconductor packages, DESeq2 and tximport (this is not done using install.packages, it’s done by searching for them on google and following the directions).
5. Use tximport to read in the Kallisto files. Remember: it needs a named list of files!
6. Generate your condition matrix. You’ll probably have to look at sample\_metadata.txt to figure out which sample is which condition
7. Run DESeq2!
8. Summarize data in various ways
9. Look at the DESeq2 documentation to figure out how to make an MAplot, and a DispersionPlot. Make them! What do these tell you?
10. DESeq2 automatically computes the false discovery rate for you. You can access it by the $padj attribute of the results object you get out of DESeq2. You can also get the nominal pvalue from the $pvalue attribute. How many genes are significant with a nominal p-value of 5% vs. a false discovery rate of 5%? You might find it helpful that you sum(c(TRUE, TRUE, FALSE, TRUE)) gives you 3, and c(.01,.06,.1,.02) < .05 gives you c(TRUE, FALSE, FALSE, TRUE). You might also be interested in the na.rm argument of the function sum (try ?sum to see what it means).