RNASeq Analysis

Martin T. Morgan*

27-28 February 2014

Contents

1 General work flows

A running example: the *pasilla* data set As a running example, we use the *pasilla* data set, derived from [1]. The authors investigate conservation of RNA regulation between *D.˜melanogaster* and mammals. Part of their study used RNAi and RNA-seq to identify exons regulated by Pasilla (*ps*), the *D.˜melanogaster* ortholog of mammalian NOVA1 and NOVA2. Briefly, their experiment compared gene expression as measured by RNAseq in S2-DRSC cells cultured with, or without, a 444 bp dsRNA fragment corresponding to the *ps* mRNA sequence. Their assessment investigated differential exon use, but our worked example will focus on gene-level differences. For several examples we look at a subset of the *ps* data, corresponding to reads obtained from lanes of their RNA-seq experiment, and to the same reads aligned to a *D.˜melanogaster* reference genome. Reads were obtained from GEO and the Short Read Archive (SRA), and were aligned to the *D.˜melanogaster* reference genome *dm3* as described in the *pasilla* experiment data package.

Work flow At a very high level, one can envision a work flow that starts with a challenging biological question (how does *ps* influence gene and transcript regulation?). The biological question is framed in terms of wet-lab pro-

*mtmorgan@fhcrc.org

tocols coupled with an appropriate and all-important experimental design. There are several well-known statistical challenges, common to any experimental data What treatments are going to be applied? How many replicates will there be of each? Is there likely to be sufficient power to answer the biologically relevant question? Reality is also important at this stage, as evidenced in the *pasilla* data where, as we will see, samples were collected using different methods (single versus paired end reads) over a time when there were rapid technological changes. Such reality often introduces confounding factors that require appropriate statistical treatment in subsequent analysis.

The work flow proceeds with data generation, involving both a wet-lab (sample preparation) component and actual sequencing. It is essential to acknowledge the biases and artifacts that are introduced at each of these stages. Sample preparation involves non-trivial amounts of time and effort. Large studies are likely to have batch effects (e.g., because work was done by different lab members, or different batches of reagent). Samples might have been prepared in ways that are likely to influence down-stream analysis, e.g., using a protocol involving PCR and hence introducing opportunities for sample-specific bias. DNA isolation protocols may introduce many artifacts, e.g., non-uniform representation of reads across the length of expressed genes in RNA-seq. The sequencing reaction itself is far from bias-free, with known artifacts of called base frequency, cycle-dependent accuracy and bias, non-uniform coverage, etc. At a minimum, the research needs to be aware of the opportunities for bias that can be introduced during sample preparation and sequencing.

The informatics component of work flows becomes increasing important during and after sequence generation. The sequencer is often treated as a 'black box', producing short reads consisting of 10's to 100's of nucleotides and with associated quality scores. Usually, the chemistry and informatics processing pipeline are sufficiently well documented that one can arrive at an understanding of biases and quality issues that might be involved; such an understanding is likely to be particularly important when embarking on questions or using protocols that are at the fringe of standard practice (where, after all, the excitement is).

The first real data seen by users are *fastq* files. These files are often simple text files consisting of many millions of records, and are described in greater detail earlier in the course. The center performing the sequencing typically vets results for quality, but these quality measures are really about the performance of their machines. It is very important to assess quality with respect to the experiment being undertaken – Are the numbers of reads consistent across samples? Is the GC content and other observable aspects of the reads consistent with expectation? Are there anomalies in the sequence results that reflect primers or other reagents used during sample preparation? Are well-known artifacts of the protocol used evident in the reads in hand?

The next step in many work flows involves alignment of reads to a reference genome. There are many aligners available, including BWA [2], Bowtie / Bowtie2 [3], and GSNAP; merits of these are discussed in the literature. *Bioconductor* packages 'wrapping' these tools are increasingly common (e.g., *Rbowtie*, *gmapR*; *cummeRbund* for parsing output of the *cufflinks* transcript discovery pathway). There are also alignment algorithms implemented in *Bioconductor* (e.g., `matchPDict` in the *Biostrings* package, and the *Rsubread* package); `matchPDict` is particularly useful for flexible alignment of moderately sized subsets of data. Most main-stream aligners produce output in 'SAM' or 'BAM' (binary alignment) format. BAM files are the primary starting point for many analyses, and their manipulation and use in *Bioconductor* was introduced earlier in the course.

2 RNA-seq: case study

2.1 Varieties of RNA-seq

RNA-seq experiments typically ask about differences in transcription of genes or other features across experimental groups. The analysis of designed experiments is statistical, and hence an ideal task for *R*. The overall structure of the analysis, with tens of thousands of features and tens of samples, is reminiscent of microarray analysis; some insights from the microarray domain will apply, at least conceptually, to the analysis of RNA-seq experiments.

The most straight-forward RNA-seq experiments quantify abundance for known gene models. The known models are derived from reference databases, reflecting the accumulated knowledge of the community responsible for the data. The 'knownGenes' track of the UCSC genome browser represents one source of such data. A track like this

describes, for each gene, the transcripts and exons that are expected based on current data. The *GenomicFeatures* package allows ready access to this information by creating a local database out of the track information. This data base of known genes is coupled with high throughput sequence data by counting reads overlapping known genes and modeling the relationship between treatment groups and counts.

A more ambitious approach to RNA-seq attempts to identify novel transcripts. This requires that sequenced reads be assembled into contigs that, presumably, correspond to expressed transcripts that are then located in the genome. Regions identified in this way may correspond to known transcripts, to novel arrangements of known exons (e.g., through alternative splicing), or to completely novel constructs. We will not address the identification of completely novel transcripts here, but will instead focus on the analysis of the designed experiments: do the transcript abundances, novel or otherwise, differ between experimental groups?

2.2 RNA-seq work flows

RNA-seq work flows aim at measuring gene expression through assessment of mRNA abundance. Work flows involve:

1. Experimental design.
2. Wet-lab protocols for mRNA extraction and reverse transcription to cDNA.
3. Sequencing; QA.
4. Alignment of sequenced reads to a reference genome; QA.
5. Summarizing of the number of reads aligning to a region; QA.
6. Normalization of samples to accommodate purely technical differences in preparation.
7. Statistical assessment of differential representation, including specification of an appropriate error model.
8. Interpretation of results in the context of original biological questions; QA.

The inference is that higher levels of gene expression translate to more abundant cDNA, and greater numbers of reads aligned to the reference genome. The enumeration above seems simplistic, but oddly enough one has concerns and commentary on each point.

2.3 Wet-lab protocols, sequencing, and alignment

The important point here is that wet-lab protocols, sequencing reactions, and alignment introduce artifacts that need to be acknowledged and, if possible, accommodated in down-stream analysis. These artifacts and approaches to their remediation are discussed in the following sections.

3 Statistical issues

Important statistical issues are summarized in Table˜1.

3.1 Experimental design

Technical versus biological replication Obviously one should follow best practices for designing experiments appropriate for the data under analysis. A typical experiment will have one or several groups. Because there is uncertainty in each measurement, we require replication. Previous work shows that technical replication (repeating identical wet-lab and sequencing protocols on a single biological sample) introduces variation that is small˜[4] compared to biological replicates (using different samples). Most RNA-seq experiments require biological replication, and seldom include technical replicates.

Table 1: Statistical issues in RNA-seq differential expression.

Analysis stage	Issues
Experimental design	Replication, complexity, feasibility
Batch effects	Known and unknown factors.
Summarize	Counts versus RPKM and other summaries.
Normalize	Robust estimates of library size.
Differential expression	Appropriate error model (Negative Binomial, Poisson, ...); dispersion (under negative binomial) as parameter requiring estimation; 'shrinkage' to balance accuracy of per-gene estimates with precision of experiment-wide estimates.
Testing	Filtering to reduce multiple comparisons & false discovery rate.

Sample size How many biological replicates? It is helpful to think in terms of orders of magnitude – biological treatments with strong and consistent consequences for gene expression will be detected with a handful – 2 or 3 – replicates per treatment. Conversely, statistically subtle effects will not be much revealed by samples of say 5 or 8, but will instead require 10's or 100's of samples. The *RNASeqPower* package provides data-driven guidance on power calculations in RNA-seq experiments; *CSSP* provides ChIP-seq power calculations based on Bayesian estimation for local counting processes.

Complexity How complicated an experimental design? The advice must be to 'keep it simple'. There are many interesting biological questions that one could ask, but experimental designs with more than one or at most two factors, or with multiple levels per factor, will undermine statistical power and complicate analysis. There are exceptions of course, for instance a time course design or an experiment with two or more factors, but these require strong *a priori* motivation and confidence that the design is amenable to analysis even in the face of wet-lab or sequencing catastrophe.

Feasibility of intended statistical analysis What kind of treatment? Two 'lessons learned' from microarray analysis and applicable to RNA-seq inform this question. (a) It is necessary to normalize observations between samples to accommodate purely technical variation in overall patterns of expression. For example, samples provided to the sequencer have different amounts of DNA, resulting in variation in total numbers of sequenced and aligned reads independent of any difference in gene-level differential representation. This implies that the treatment should *affect only a fraction of the genes assayed*, otherwise treatment effects and protocol artifacts are confounded. (b) Between-gene measures of expression differ for reasons unrelated to levels of expression. For instance, standard protocols mean that a long gene is sequenced more often than a short gene, even when the number of mRNA molecules of the two genes are identical. This means that the most productive approach to differential representation will *compare genes across samples*, rather than compare levels of representation of different genes (gene set enrichment analysis and other approaches to between-gene comparison are statistically interesting in part because of the need to overcome between-gene differences arising for purely technical reasons). The combination of lessons (a) and (b) dictate that the treatment should affect only a subset of the genes under study, and that 'interesting' results correspond to treatment groups with differences at the gene level. *A priori* motivation, e.g., about well-defined pathways as targets of differential representation, may trump part (b) of this guideline.

3.2 Batch effects

The reality of executing designed experiments may mean that there are known but unavoidable factors that confound the analysis, but that are not of fundamental biological interest. Perhaps samples are being processed by different groups, or processing is spread over several months to accommodate personnel or sequencer availability. It is essential to avoid confounding such factors with biologically relevant parts of the experiment. Such batch effects are pervasive in high-throughput analysis of diverse data types [5]; addressing batch effects helps to reduce dependence, stabilize error rate estimates, and improve reproducibility.

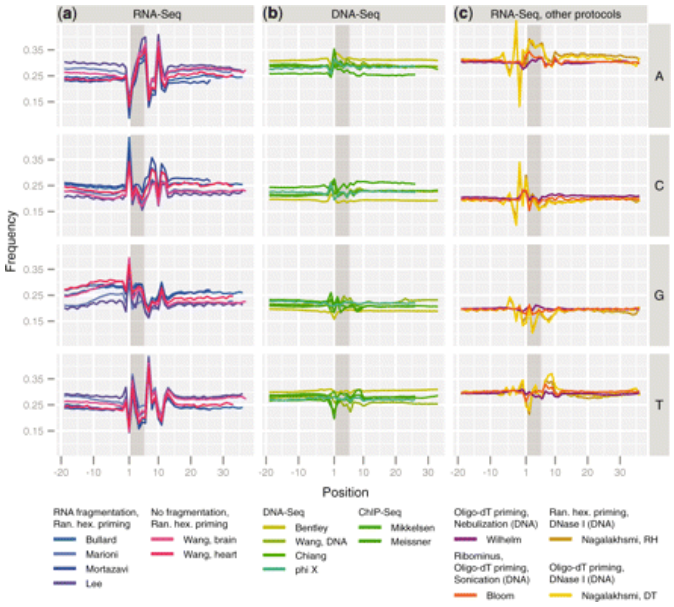

Figure 1: Nucleotide frequency versus position relative to start of alignment, various experiments and protocols; see˜[12].

Having acknowledged a potentially confounding factor, what is to be done? A first reaction might be randomization – arrange for samples to be processed in a random order, for instance, rather than by treatment group – but a better strategy is usually to include a blocking factor, e.g., processed by lab 'A' versus lab 'B' and to ensure that treatments are represented by replicates in each blocking factor. The down-stream analysis can then use replication to statistically accommodate such effects.

An alternative to explicitly modeling batch effects is to identify 'surrogate variables'. Surrogate variables are covariates constructed directly from the data, and can be used in subsequent analysis to adjust for unknown, unmodeled, or latent sources of noise [6, 7, 8]. The *sva* package implements surrogate variable analysis, and can be used with RNA-seq and many other high dimensional data types. *sva* estimates surrogate variables for inclusion in subsequent analysis, or removes known batch effects using ComBat [9].

An interesting approach to addressing batch effects in studies where new samples are accumulated incrementally (e.g., patient assays from physician offices) is to create a 'frozen' correction on a training data set, and perform per-sample correction on new samples as they become available. This is similar to the 'frozen' RMA approach to normalization developed by McCall et al., [10], and is implemented by the `fsva` function in the *sva* package.

3.3 Summarizing

The summary process tallies the number of reads aligning in each region (e.g., gene) of interest. The simplest method is to simply count reads overlapping each region, dividing by the length of the region of interest to accommodate differences in gene length. This is the 'RPKM' (reads per kilobase per million reads) of Mortazavi et al.˜[11]. One problem with this approach is that reads are not sampled uniformly across genes (Figure˜1; [12]), so gene length (the 'PK' part of RPKM) is not a good proxy for expression level.

More fundamentally, each read represents an observation, and contributes to the certainty with which a gene is measured as 'expressed'. A summary measure like RPKM fails to incorporate uncertainty – a particular value of RPKM might result from alignment of one or 100 reads. This contrasts with a simple count of the number of reads in the region of interest. Furthermore, count data has known statistical properties that can be exploited in down-stream statistical analysis. Thus the result of summarization most useful for assessing differential expression is read count.

How to count? For instance, should a read that partly overlaps a 5' UTR or an intron be included in a tally? What about reads that overlap multiple genes? This is a non-trivial question because alignment is only approximate (reflecting sequencing and other biases) and because sample preparation protocols and organism biology (e.g., whether the UTR or fully mature mRNA is sequenced) may dictate particular counting strategies; more elaborate counting strategies might be entertained for paired end reads. Anders enumerates some counting strategies[1]; these are implemented in his *HTSeq* python scripts, in `summarizeOverlaps` in the *GenomicRanges* package, or in functions in the (linux-only) *Rsubread* and *gmapR* packages.

3.4 Normalization

Normalization arises from the need to correct for purely technical differences between samples. The most common symptom of the need for normalization is differences in the total number of aligned reads. The 'M' part of RPKM measure mentioned in the context of summarization is one way of normalizing for total count. This normalization is not appropriate, because the distribution of aligned reads across genes within a sample is not uniform – some regions receive many more alignments than do others – and this distribution may differ between samples.

The overall strategy with normalization is to choose an appropriate baseline, and express sample counts relative to that baseline. There are several approaches to choice of appropriate baseline. One might choose total count for normalization, but this is a poor choice when one or a few regions of interest are very well represented – we are normalizing to the well-represented genes rather than to sequencing depth in each sample. Other straightforward approaches include use of house-keeping genes, or the expression level from a particular quantile of the distribution of gene expression values of each sample˜[13]. One might attempt a robust estimate of sample abundance that is less sensitive to extreme outliers, e.g., the trimmed geometric mean of counts˜[14]. Another approach is TMM˜[15], which measures the trimmed mean of M and A values (M values are the log fold change in the number of reads aligning to a region of interest measured relative to an average or arbitrary sample, A is the average count of a gene; the trimmed mean discards regions of interest that have extreme M or A values and calculates the mean M value of the remainder); the inverse of this mean is used to weight samples. More data-driven approaches exploiting the gene-specific properties include conditional quantile normalization (implemented in the *cqn* package;˜[16]).

Another approach to normalization, increasingly popular as experiment size and data consistency increases, is to perform a data transformation and apply normalization methods developed for analysis of microarrays. Examples of this approach include `varianceStabilizingTransformation` from the *DESeq2* package, and `voom` from the *limma* package; see the corresponding help pages of these functions for details).

3.5 Error model

A Negative Binomial error model is often appropriate for 'smaller' experiments. These models combine Poisson ('shot' noise, i.e., within-sample techincal and sampling variation in read counts) with variation between biological samples. The *edgeR*˜[17] and *DESeq*˜[14] (now *DESeq2*) packages implement these models. Negative bionomial error models involve estimation of dispersion parameters, which are estimated poorly in small samples. *edgeR* and *DESeq2* adopt different data-driven approaches to arrive at more robust dispersion estimates; the packages, relying on different strategies to moderate per-gene estimates with more robust local estimates derived from genes with similar expression values. Other approaches are possible; *DSS*˜[18] estimates are based on γ-Poisson or β-Binomial distributions.

As number of replicates become large, the importance of explicitly modeling biological sampling variance decrease. This encourages use of the Poisson-Tweedie family of distributions to model count data˜[19].

[1] http://www-huber.embl.de/users/anders/HTSeq/doc/count.html

Table 2: Selected *Bioconductor* packages for RNA-seq analysis.

Package	Description
EDASeq	Exploratory analysis and QA; also *qrqc*, *ShortRead*, *DESeq2*.
edgeR, *DESeq2*	Generalized Linear Models using negative binomial error.
BitSeq	Bayesian inference of individual transcript abundances followed by differential expression.
DEXSeq	Exon-level differential representation.
DSS, *vsn*, *cqn*	RNA-seq normalization methodologies. Also `voom` in *limma*.
goseq	Gene set enrichment tailored to RNAseq count data; also *limma*'s `roast` or `camera` after transformation with `voom`.
QuasR	Workflow.
Rsubread	Alignment (Linux only); also *gmapR*; *Biostrings* `matchPDict` for special-purpose alignments.
cummeRbund	Exploration and analysis of Cufflinks results.

3.6 Multiple comparison

1. Increase statistical power and reduce false discovery rate by filtering regions of interest prior to analysis.
2. Motivation (a): just because genes are assayed does not mean, *a priori*, that they represent something requiring a statistical test. (b) Some observations, e.g., zero counts across all samples, cannot possibly be statistically significant, independent of hypothesis under investigation.
3. Approach – detection or 'K over A'-style filter; representation of a minimum of A (normalized) read counts in at least K samples. A usually measured as counts per million. Guidelines for choice of values a little *ad hoc*; see, e.g., the *edgeR* user manual. Variance filter, e.g., IQR (inter-quartile range) provides a robust estimate of variability; can be used to rank and discard least-varying regions.

4 Selected *Bioconductor* software for RNA-seq Analysis

Bioconductor packages play a role in several stages of an RNA-seq analysis (Table˜2; a more comprehensive list is under the RNAseq and HighThroughputSequencing BiocViews terms). The *GenomicRanges* infrastructure can be effectively employed to quantify known exon or transcript abundances. Quantified abundances are in essence a matrix of counts, with rows representing features and columns samples. The *edgeR*˜[15] and *DESeq2*˜[14] packages facilitate analysis of this data in the context of designed experiments, and are appropriate when the questions of interest involve between-sample comparisons of relative abundance. The *DEXSeq* package extends the approach in *edgeR* and *DESeq2* to ask about within-gene, between group differences in exon use, i.e., for a given gene, do groups differ in their exon use?

5 DESeq2 Work Flow Exercises

For this chapter, follow in-course instructions to work through the Parathryoid *DESeq2* analysis.

References

[1] A.˜N. Brooks, L.˜Yang, M.˜O. Duff, K.˜D. Hansen, J.˜W. Park, S.˜Dudoit, S.˜E. Brenner, and B.˜R. Graveley. Conservation of an RNA regulatory map between Drosophila and mammals. *Genome Research*, pages 193–

202, 2011. URL: http://genome.cshlp.org/cgi/doi/10.1101/gr.108662.110, doi:10.1101/gr.108662.110.

[2] H.˜Li and R.˜Durbin. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, 26:589–595, Mar 2010.

[3] B.˜Langmead, C.˜Trapnell, M.˜Pop, and S.˜L. Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.*, 10:R25, 2009.

[4] John˜C Marioni, Christopher˜E Mason, Shrikant˜M Mane, Matthew Stephens, and Yoav Gilad. Rna-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, 18(9):1509–1517, 2008.

[5] J.˜T. Leek, R.˜B. Scharpf, H.˜C. Bravo, D.˜Simcha, B.˜Langmead, W.˜E. Johnson, D.˜Geman, K.˜Baggerly, and R.˜A. Irizarry. Tackling the widespread and critical impact of batch effects in high-throughput data. *Nat. Rev. Genet.*, 11:733–739, Oct 2010.

[6] J.T. Leek and J.D. Storey. Capturing heterogeneity in gene expression studies by 'surrogate variable analysis'. *PLoS Genetics 3:e161*, 2007.

[7] J.T. Leek and J.D. Storey. A general framework for multiple testing dependence. *Proceedings of the National Academy of Sciences 105:18718-18723*, 2008.

[8] J.˜T. Leek. Asymptotic conditional singular value decomposition for high-dimensional genomic data. *Biometrics*, 67:344–352, Jun 2011.

[9] W.E. Johnson, C.˜Li, and A.˜Rabinovic. Adjusting batch effects in microarray data using empirical bayes methods. *Biostatistics*, 8(1):118–127, 2007.

[10] Matthew˜N McCall, Benjamin˜M Bolstad, and Rafael˜A Irizarry. Frozen robust multiarray analysis (frma). *Biostatistics*, 11(2):242–253, 2010.

[11] Ali Mortazavi, Brian˜A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by rna-seq. *Nature methods*, 5(7):621–628, 2008.

[12] Kasper˜D Hansen, Steven˜E Brenner, and Sandrine Dudoit. Biases in illumina transcriptome sequencing caused by random hexamer priming. *Nucleic acids research*, 38(12):e131–e131, 2010.

[13] James˜H Bullard, Elizabeth Purdom, Kasper˜D Hansen, and Sandrine Dudoit. Evaluation of statistical methods for normalization and differential expression in mrna-seq experiments. *BMC bioinformatics*, 11(1):94, 2010.

[14] S.˜Anders and W.˜Huber. Differential expression analysis for sequence count data. *Genome Biol*, 11(10):R106, 2010.

[15] M.˜D. Robinson, D.˜J. McCarthy, and G.˜K. Smyth. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26:139–140, Jan 2010.

[16] Kasper˜D. Hansen, Rafael˜A. Irizarry, and Zhijin Wu. Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics*, 13(2):204–216, 2012.

[17] McCarthy, Davis J., Chen, Yunshun, Smyth, and Gordon K. Differential expression analysis of multifactor rna-seq experiments with respect to biological variation. *Nucleic Acids Research*, 40(10):–9, 2012.

[18] Hao Wu, Chi Wang, and Zhijin Wu. A new shrinkage estimator for dispersion improves differential expression detection in rna-seq data. *Biostatistics*, 2012. doi:10.1093/biostatistics/kxs033.

[19] Mikel Esnaola, Pedro Puig, David Gonzalez, Robert Castelo, and Juan˜R Gonzalez. A flexible count data model to fit the wide diversity of expression profiles arising from extensively replicated rna-seq experiments. *BMC Bioinformatics*, 14:254, 2013.