A Computational Pipeline for Cross-Species Analysis of RNA-seq Data Using R and Bioconductor

Peter R. LoVerso and Feng Cui

Thomas H. Gosnell School of Life Sciences, Rochester Institute of Technology, One Lomb Memorial Drive, Rochester, NY, USA.

ABSTRACT: RNA sequencing (RNA-seq) has revolutionized transcriptome analysis through profiling the expression of thousands of genes at the same time. Systematic analysis of orthologous transcripts across species is critical for understanding the evolution of gene expression and uncovering important information in animal models of human diseases. Several computational methods have been published for analyzing gene expression between species, but they often lack crucial details and therefore cannot serve as a practical guide. Here, we present the first step-by-step protocol for cross-species RNA-seq analysis with a concise workflow that is largely based on the free open-source R language and Bioconductor packages. This protocol covers the entire process from short-read mapping, gene expression quantification, differential expression analysis to pathway enrichment. Many useful utilities for data visualization are included. This complete and easy-to-follow protocol provides hands-on guidance for users who are new to cross-species gene expression analysis.

KEYWORDS: RNA-seq, computational pipeline, cross-species

CITATION: LoVerso and Cui. A Computational Pipeline for Cross-Species Analysis of RNA-seq Data Using R and Bioconductor. *Bioinformatics and Biology Insights* 2015:9 165–174 doi: 10.4137/BBI.S30884.

TYPE: Original Research

RECEIVED: June 30, 2015. **RESUBMITTED:** October 21, 2015. **ACCEPTED FOR PUBLICATION:** October 24, 2015.

ACADEMIC EDITOR: J. T. Efird, Associate Editor

PEER REVIEW: Thirteen peer reviewers contributed to the peer review report. Reviewers' reports totaled 1912 words, excluding any confidential comments to the academic editor.

FUNDING: This work was supported by the start-up fund, Faculty of Development (FEAD) fund, and Dean's Research Initiation Grant fund from the Rochester Institute of Technology and a grant from the National Institutes of Health (R15GM116102 to FC). The authors confirm that the funder had no influence over the study design, content of the article, or selection of this journal

COMPETING INTERESTS: Authors disclose no potential conflicts of interest.

CORRESPONDENCE: fxcsbi@rit.edu

Introduction

The transcriptome representing the entire repertoire of gene transcripts in a cell bridges the gap between genetic information encoded in DNA and phenotypes. A quantitative measurement of the transcriptome provides a snapshot of the content and dynamics of RNA species under a certain cellular condition. Traditional tools for gene expression profiling include Northern blot, reverse-transcription polymerase chain reaction, expressed sequence tags, and serial analysis of gene expression. The advent of microarray[1,2] and, more recently, RNA sequencing[3,4] (RNA-seq) allows fast, cost-effective, and comprehensive measurement of messenger RNA abundance for thousands of genes simultaneously. Many studies that compare the two technologies in the same system have found that RNA-seq has increased sensitivity for the identification of differentially expressed genes, compared to microarray measurements.[5–10]

Although microarray or RNA-seq experiments are often used to probe changes in gene expression within a species,[11–17] understanding the differences in gene expression between species has a number of important applications in the fields of biology and medicine, including (1) evolution in gene expression[18–20]; (2) animal models of human diseases such as cancers,[21,22] Alzheimer's disease,[23] Huntington's disease,[24] diabetes,[25] and hypertension[26]; (3) developmental biology[27]; (4) aging[28–30]; (5) toxicology[31]; and (6) biomarkers.[32,33] As such, several computational methods have been proposed and developed to analyze interspecies gene expression data.[8,34–36]

However, a detailed, step-by-step protocol is not available for cross-species RNA-seq data analysis, which hampers the full utilization of gene expression data in public repositories such as Gene Expression Omnibus[37] and ArrayExpress.[38] Here, we address this need with a protocol based on published computational pipelines[39–41] and relevant Bioconductor packages. The steps in this protocol are detailed in "Description of the protocol" section, which include (1) short-read alignment to a genome, (2) quantification of gene expression based on a given annotation, (3) lifting of annotations between species to their best orthologs, (4) differential expression analysis between multiple species or between multiple samples of one species, and (5) pathway enrichment and analysis of differentially expressed genes.

Description of the Protocol

RNA-seq analysis typically begins with the sequencing of many individual complementary DNA reads, which are usually no more than several hundred base pairs long. Quality control software assesses the quality of each base pair of a sequenced read and returns a file in the FASTQ format with both DNA sequence and a quality score for each nucleotide. A scientist then examines this file and may use tools to improve the average quality of the data by truncating the read fragments with low-quality scores. Other operations are often performed in this step as well, such as the demultiplexing of barcoded samples. At the end of the quality control process, a file with the high-quality reads is the starting point for this protocol (Fig. 1, Step 1).

The reads are then aligned to the genome of the organism using *SHRiMP*[42] (Fig. 1, Step 2). This part of the protocol is dependent upon input data and is not significantly distinct from many other methods of sequencing. *SHRiMP* can easily be exchanged for other aligners of a user's preference (eg, *Tophat*[43] and *GSNAP*[44]). The output of many sequencing programs is in the sequence alignment/map (SAM) format[45] and is then converted to a binary format (BAM) for improved performance and storage efficiency. Performance can be further improved by sorting and indexing the file (Fig. 1, Step 3).

The next step is to quantify gene expression across species based on a gene annotation (Fig. 1, Step 4). This part of the protocol is quite specific to comparisons between species and is sensitive to errors. A single reference species is identified, in this protocol the mm10 annotation,[46,47] and the annotation file is downloaded in the GFF format. Constitutive exons, which are exons that are always included in the final gene product, are identified in this annotation using MISO.[48] Other parts of the annotation that are not constitutive exons are discarded. Pairwise genome alignments of the chosen reference annotation to each query species are downloaded in the AXT format.[49–51] All exons in the reference annotation that have complete orthologous regions in all query species genomes are lifted to their respective orthologous position in each query species, while maintaining the gene IDs of the reference species. The resulting annotations are then converted from the GFF format to the GTF format using the *gffread* utility from the Cufflinks package.[43] This step is discussed in more detail in "Generation of cross-species genome annotations" section as it is not covered in any other published protocols.

The annotations are then used to count the number of reads that align to each exon from the indexed alignment file, which is used to calculate expression on a per-gene level. For comparison between species, this pipeline uses a count-based method rather than an FPKM-based method for quantifying expression, as it is easier to integrate this information into downstream expression analysis tools. The reason for this is twofold. First, many tools that compute differential expression (eg, *cuffdiff*) require that one annotation can be given for all input alignment files, which does not function when comparing between species, with a different annotation for each species. Second, many FPKM measurements take into account the expression of genomic locations that are not included in the annotation. Expressed reads aligned outside the annotation is used to normalize the expression levels of genes within the annotation. This is desirable for many analyses, as it allows to see if one gene is expressed or not expressed compared to other genes. However, it is not desirable in our case, as genomic locations outside of the annotation are not considered homologous. Including them to measure FPKM would render the data incomparable between species. Instead, gene expression within a sample is normalized against total expression within the annotation for that sample. To this end, the annotations are pared down to only those constitutive exons orthologously present in all queried species. Differential expression analysis should focus on those exons and genes that can be measured in all samples.

Mapped short reads are counted for each sample against the respective annotation using Rsubread,[52] which returns the count information in a list (Fig. 1, Step 5). This list can then be read into edgeR[53] that is able to perform a number of statistical tests upon the count data (Fig. 1, Step 6). Of primary importance, differential expression is computed for each gene between each sample using a negative binomial distribution.[54,55] The list of differentially expressed genes may then be subset by magnitude and reported directly, as well as lend itself to further downstream analysis. In particular, our method covers the use of *SPIA*,[56] *GAGE*,[57] and *pathview*,[58] which are analysis packages from Bioconductor[59] (Fig. 1, Steps 7 and 8). *GAGE*, which stands for Generally Applicable Gene-set Enrichment, examines all differential expression between two samples and determines which annotated cellular pathways are significantly different between the two samples, based on a given set of pathway annotations. This protocol utilizes pathways from the Kyoto Encyclopedia of Genes and Genomes (KEGG).[60] This is done using a standard gene set enrichment, where DEGs are ranked by log fold change. Then based on the ranks and numbers of pathways, certain pathways are determined to be significant. *SPIA* performs similarly but has the added feature of assessing the topology of the pathway. For example, if in a particular comparison, the first sample compared has high expression in genes promoting a certain pathway and the second sample has increased expression in genes repressing that pathway, it rates that pathway as more significantly different than if the DEGs are randomly distributed through the pathway. However, this can also backfire if genes that are involved in both activation and repression are both upregulated in one sample, possibly reducing the likelihood of discovering new pathways. For this reason, both *GAGE* and *SPIA* are used in this protocol. Once significantly different pathways are determined, a tool named *pathview* is used to give gene names, pathway names, and expression levels, and it queries the KEGG servers for the pathway diagrams, annotating and coloring them in accordance with expression levels. Pathway-level expression analysis is the final goal of this protocol, enabling researchers to view and explore the differences between two samples of different species, in terms of one reference species' pathways, tying the difference in gene expression into the true biological differences and allowing for a much more human-readable set of results.

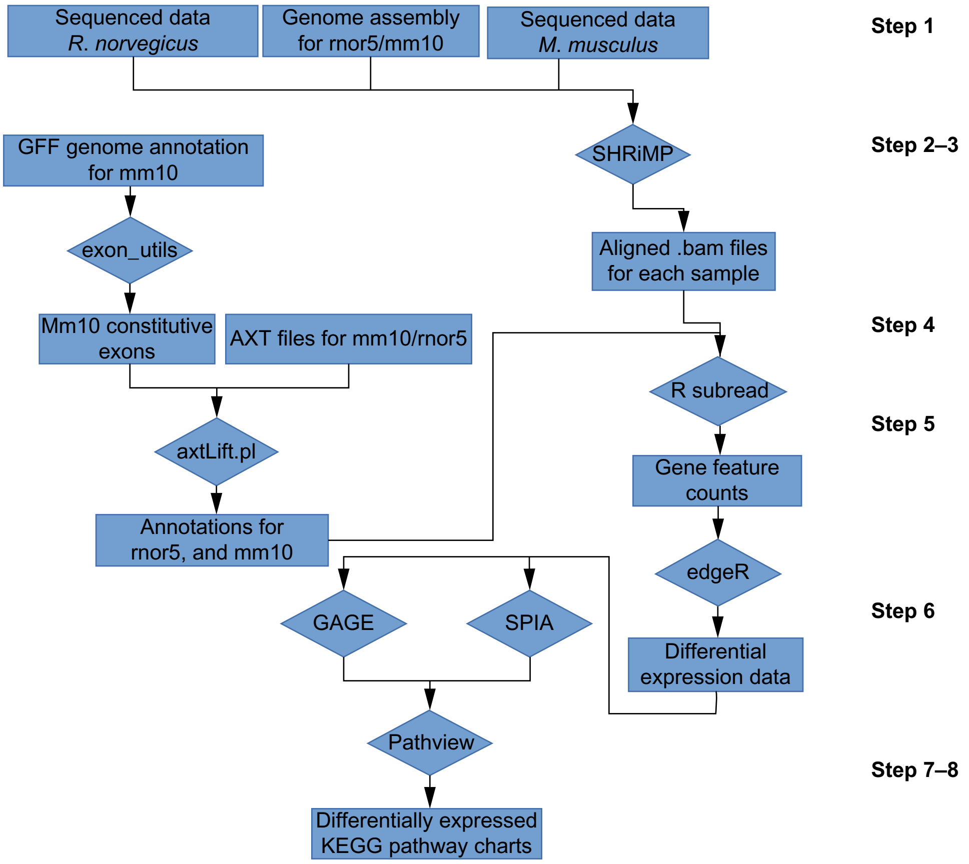

Figure 1. Computational pipeline for cross-species expression analysis using RNA-seq. The pipeline is divided into eight steps. Description of each step and corresponding commands are included in the text. The custom scripts for RNA-seq analysis and data visualization can be downloaded from https://github.com/ploverso. Other software or R packages listed are available for downloading (details are given in "Hardware and software" section).

In addition to the above-mentioned pathway analysis charts, a number of scripts are written that leverage various other R packages to display various interesting aspects of the data using heatmaps, Venn diagrams, and other charts.

Generation of Cross-Species Genome Annotations

The key to being able to compare RNA-seq data between different species is the generation of a cross-species genome annotation. To this end, one species is selected as a "reference" species against which any other query species is compared. In this protocol, the mouse genome and annotation are selected as the reference. The goal is to ultimately compare all data from all relevant samples to one another in terms of genes and pathways in the reference species. The best way to do this is to use orthologous genome regions. Because many annotations may be variably complete or have similarly named genes that have different functions, comparisons at a base pair level are used to determine which regions in the query species' genomes are to match up to each region in the reference annotation.

One commonly used tool for the translation of genomic coordinates from one annotation version to another, or indeed one species to another, is the University of California Santa Cruz (UCSC)'s *liftOver* utility. However, the chain/net files used by *liftOver* are ill suited to comparisons between species as small changes to the parameters can cause huge changes in the output when the two species have a large evolutionary distance. To adjust for this, the UCSC conservation track, which is the best alignment between two genomes, is used instead. This track makes the comparison more robust – not only are the conservation tracks partially based on ontology, but location conversion between the two genomes are also symmetrical. Symmetrical location conversion means that if a region in the mouse genome is converted to the rat genome using the conservation track, the resulting region in the rat converts back to the exact original region in the mouse. This is not always the case when using *liftOver*, due to the asymmetrical nature of dynamic masking in *Blastz*. The symmetry of the conservation track allows for a much more robust comparison between species.

Furthermore, in our protocol, the reference annotation is filtered such that only constitutive exons, that is, the exons in a gene that are always incorporated into the final gene product, are included. Although alternatively splicing is biologically important, comparing all exons in a gene between species is less meaningful as exons may differ in size and number. As such, exons that may be spliced out of the primary transcript, that is, cassette exons, are removed from consideration since they serve as a source of variation between samples.

Comparisons with Other Methods

To the best of the knowledge of the authors, this is the first published protocol providing a generally applicable set of instructions for the comparison of RNA-seq data between different species. This protocol is partially based on a method published by Liu et al.[8] for comparing RNA-seq data between closely related species, while adapting other parts of this protocol on previously published protocols and software[39–41] that are already commonly used throughout the field. The goal of this protocol is to use off-the-shelf software when possible for the analysis of the data, while writing and publishing new code for ease of comparisons between species. There are often many options available for most steps in the protocol, which work with similar input and output files; the user may alter some steps to use one of the many other tools available, for example, *Tophat* rather than *SHRiMP*, or *DEseq* instead of *edgeR*, as drop-in replacements.

Several other groups have published tools for cross-species gene expression analysis.[34–36] For instance, Kuhn et al.[34] used an approach similar to ours (gene ontology), but rather than examine either species at a base pair level, they have built tools to query Homologene for gene IDs and return orthologous mappings in another species. This tool has been wrapped and published in a Bioconductor package "annotationTools." One weakness of their method is that it does not contain any function to judge the relatedness between the genes. It simply returns the gene IDs, relying on Homologene to do the heavy lifting.

Zhu et al.[36] used a method that is also similar to ours. That is, they have built cross-species annotations using liftOver, which poses some problems discussed in "Generation of cross-species genome annotations" section. They then used BLAT to filter out their orthologous exons, while we used AXT files instead.

Kristiansson et al.[35] defined and implemented a method for cross-species gene expression analysis as well, although they did not provide a detailed step-by-step protocol for upstream preparation and downstream analyses. While the general idea of this method is the same as ours – comparison of expression based on ontology – the implementation is very different. They take into account the homology structure between compared species and compare the expression data from genes that have any number of orthologs and paralogs. A simulation study has shown that this method has increased statistical power compared to other methods. We may construct a separate protocol later based on their method for the analysis of cross-species gene expression data.

Confounding Effects

Several confounding effects may be introduced in the cross-species analysis of RNA-seq data. First, as this protocol seeks to compare RNA-seq data between different species using one annotation, differences in the procurement and treatment of cells can introduce variations into the data, as well as the relatedness of the species. Second, the quantification of orthologous genes across species can only be an approximation based on alignment scores, and in distantly related species, this may introduce confounding of the data. Third, if the protocol is used to compare data from more than one study, differences in cells or data treatments between studies may also introduce variations. Fourth, experimental and computational tools used for the analysis may introduce variations. It is advised that a user should keep track of all software versions and cell treatment protocols used. Any differences in the protocols may be used to determine whether differences found in the downstream analysis are the result of these factors or true biological expression variations. Fifth, comparing one sample from one species to one sample from another species may have bias introduced in the process of the cross-species annotation. In order to control for this, it is recommended that multiple cell types in a species be examined. Thus, variations in gene expression in individual samples can be controlled against the average expression of all samples in the same species.

Hardware and Software

The computing resources necessary for this protocol are heavy, particularly for the alignment of sequencing reads to the reference genome. While there are ways to reduce the memory footprint of the alignment if necessary, it is recommended to use a computer with at least eight cores and 64 GB of RAM, as well as at least 500 GB of hard drive space. Additional resources allow multiple samples to be run simultaneously, significantly speeding up the analysis.

This protocol is constructed on and for a GNU/Linux operating system, and commands are given assuming that the user is using a POSIX-compliant operating system with access to a shell such as bash. While it is possible to run the protocol under Microsoft Windows, several additional steps would be necessary for the proper execution of various programs, which is outside the scope of this protocol. The author recommends one of the Debian or Red Hat distro's for this protocol.

The SHRiMP alignment software may be downloaded from http://compbio.cs.toronto.edu/shrimp/.

To work with the alignment files, SAMtools is used for conversion, sorting, and indexing of the files. It may be installed from your distro's software repository or downloaded from http://samtools.sourceforge.net/.

Determination of constitutive exons leverages MISO, which can be downloaded from https://miso.readthedocs.org/en/fastmiso/index.html.

Various scripts and utilities for working with axt files, as well as various downstream analyses, were written by the author and may be downloaded from a git repo created for this protocol at https://github.com/ploverso.

The *gffread* utility, as well as other programs for downstream analysis, is part of the Cufflinks package and may be installed from your distro's software repository or downloaded from https://cole-trapnell-lab.github.io/cufflinks/.

The R statistical computing environment may be downloaded from http://cran.rstudio.com/.

Bioconductor and several of its packages (*Rsubread*, *edgeR*, *gage*, *gageData*, *pathview*, *Org.mm.eg.db*, and others specified below) as well as all dependencies may be installed using the Bioconductor package installer.

The *gplots* package may be installed using R's built-in package installer.

Following is the output of R's sessionInfo() command, which will show the versions of all packages used:

```
> sessionInfo()
R version 3.1.1 (2014–07–10)
Platform: x86_64-pc-linux-gnu (64-bit)
locale:
[1] LC_CTYPE=en_US.utf8    LC_NUMERIC=C
[3] LC_TIME=en_US.utf8    LC_COLLATE=en_US.utf8
[5] LC_MONETARY=en_US.utf8    LC_MESSAGES=en_US.utf8
[7] LC_PAPER=en_US.utf8    LC_NAME=C
[9] LC_ADDRESS=C    LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.utf8    LC_IDENTIFICATION=C
attached base packages:
[1] grid splines parallel stats4 stats graphics grDevices
[8] utils datasets methods base
other attached packages:
[1] org.Rn.eg.db_3.0.0 Rsubread_1.16.1 BiocInstaller_1.16.4
[4] VennDiagram_1.6.9 RColorBrewer_1.1–2 gplots_2.16.0
[7] gageData_2.3.1 gage_2.16.0 pathview_1.6.0
[10] org.Hs.eg.db_3.0.0 KEGGgraph_1.24.0 graph_1.44.1
[13] XML_3.98–1.1 edgeR_3.8.6 limma_3.22.7
[16] org.Mm.eg.db_3.0.0 RSQLite_1.0.0 DBI_0.3.1
[19] AnnotationDbi_1.28.2 GenomeInfoDb_1.2.4 IRanges_2.0.1
[22] S4Vectors_0.4.0 Biobase_2.26.0 BiocGenerics_0.12.1
[25] biomaRt_2.22.0
loaded via a namespace (and not attached):
[1] Biostrings_2.34.1 bitops_1.0–6 caTools_1.17.1 gdata_2.13.3
[5] gtools_3.4.2 httr_0.6.1 KEGGREST_1.6.4 KernSmooth_2.23–14
[9] png_0.1–7 RCurl_1.95–4.5 Rgraphviz_2.10.0 stringr_0.6.2
[13] tools_3.1.1 XVector_0.6.0 zlibbioc_1.12.0
```

Alternative Aligners to the Protocol

This protocol is modular, meaning that users can use aligners other than *SHRiMP*, such as *Tophat* or *GSNAP* to count features. With the sorted, indexed BAM files, users can proceed at Step 4 of the protocol (see later).

Input Data

Theoretically, this protocol is suitable for RNA-seq data from any species generated from a commercial NGS platform (Illumina, SOLiD, or Ion Torrent), provided that the quality control on the reads are performed according to the manufacturer's instructions. For the purpose of illustration, we use RNA-seq data from two different species, *Rattus norvegicus* and *Mus musculus*, to describe the protocol. The rat data are single-ended sequencing data, while the mouse data are paired-end sequencing data. All input files are based on output from Illumina sequencing machines and in the FASTQ format. Detailed analysis of the rat and mouse samples using this protocol was published in a separate paper.[61]

Protocol

Preparation of reference genome. The reference genomes for rat and mouse (rnor5 and mm10, respectively) are downloaded in compressed FASTA format from llumina's igenomes FTP server: ftp://igenome:G3nom3s4u@ussd-ftp.illumina.com/. Furthermore, the comprehensive gene annotation for mm10 is downloaded in the GFF format from GENCODE: http://www.gencodegenes.org/mouse_releases/3.html. Once the genome FASTA files are downloaded and extracted for each species, they are preprocessed with *SHRiMP*, which greatly decreases the time required to align the RNA-seq samples to that genome. The general command used to preprocess the genomes run from the location of each genome.fa file is

```
$~/SHRiMP_2_2_3/bin/gmapper-ls -S <assemblyName> -N 8 genome.fa
```

This command indexes the genome and saves the indexes and projections of the genome to files that are loaded for each sample to align against. If this preprocessing step is not done, it will be performed prior to mapping for each sample, causing the alignment step per sample to take many hours longer. The index files will be several times larger than the original genome.fa files and should be placed in a location convenient to the RNA-seq samples.

Alignment of RNA-seq reads to indexed genome. Each sample should be aligned to its respective genome, which is specified with the assembly name given when performing the indexing. This step is the most computationally intensive and should be performed on a computer with at least 60 GB of RAM. This step can be parallelized easily, and allotting more CPU cores to the alignment allows it to run significantly faster. We suggest writing a simple bash script to run the alignments, to reduce the amount of oversight necessary for large numbers of samples. The general command used to align the rat data is

```
$ ~/SHRiMP_2_2_3/bin/gmapper-ls -Q --qv-offset 33 -L./<indexLocation>/<assemblyName> -N 8 --all-contigs $infile > $outfile
```

This command outputs the alignment to a specified output file in the SAM format. It is only valid for the rat data, which are single-ended RNA-seq data. The mouse data are paired-end data, with each sample name suffixed with a _1 or _2 in the FASTQ format. The command used for the mouse data is

```
$ ~/SHRiMP_2_2_3/bin/gmapper-ls -Q --qv-offset 33 -L./<indexLocation>/mm10 -N 8 --all-contigs -p opp-in -1 $infile1–2 $infile2 > $outfile
```

The -p option specifies how *SHRiMP* attempts to align the paired-end data. Mapping statistics for all the samples is shown in an output table.

Conversion, sorting, and indexing of SAM files. The SAM output of *SHRiMP* is a plain-text file. While somewhat human-readable, it is uncompressed, and each SAM file may be many tens of GB. The file size slows access to the file for analysis as it requires inflating storage space on a hard drive. To solve these problems, the SAM files are converted to the compressed BAM format, then sorted and indexed. The general commands used for this are as follows:

```
$ samtools view -bS $inSAM > $tempBAM
$ samtools sort $tempBAM $outPrefix && rm $tempBAM
$ samtools index $outBAM
```

BAM files that are sorted and indexed have greatly enhanced access, which speeds up downstream analysis. Additionally, BAM files tend to be only a small fraction of the size of the SAM files, freeing up disk space and reducing RAM requirements for downstream analysis programs that need to load the entire BAM files into memory.

Generation of cross-species annotations. Starting with the GFF file downloaded from GENCODE in a previous step, the constitutive exons must be identified. This is done with the *exon_utils* program, part of MISO. The command is

```
$ exon_utils --get-const-exons mm10.gff --min-exon-size 100 --output-dir exons/
```

This command extracts all constitutive exons (ie, all exons are always incorporated into the final gene product) that are greater than 100 bp, an arbitrary cutoff value, into a GFF file in a specified folder. This file is then broken down into individual chromosomes using a PERL script from the github repo specified above, gffToChrs.pl.

```
$ perl gffToChrs.pl mm10.const_exons.gff./gffChrs/
```

Next, the axt files are downloaded, one axt file per chromosome. The wget utility may be helpful in doing so. The axt files are provided for many species by UCSC at: http://hgdownload.cse.ucsc.edu/goldenPath/mm10/vsRn5/axtNet/. Then, the *axtLift.pl* script is used to convert each chromosome of the reference annotation to the coordinates of the query annotation. Exons that do not match 100% are discarded, for example, if the exon hangs off one end or the other of the aligned region. Alignments with gaps are supported. Then, the individual chromosome files are mapped to the new genomes. The axtLift.pl script does this, and it must be run for each chromosome. To run this script, the command is

```
$ perl axtLift.pl./finalChrs/chr1.gff./mm10TOrn5/chr1.mm10.rn5.net.axt./rn5/
```

It should be noted that the output folder must be created prior to running the script. Furthermore, any files in the folders should be deleted if this step needs to be rerun as the script appends rather than overwrites files. The script only supports a single input GFF and AXT file, so a bash script may be useful to run all chromosomes for the reference species. It is not recommended to run the chromosomes in parallel as each input chromosome may map to any of the output chromosome files.

Once this is completed, the GFF files should be sorted and combined into a single annotation file. This is again easy to do with R. For example:

```
$ R
> final <- data.frame(matrix(nrow=0, ncol=9))
> for(gffFl in dir("rnor5")){
gffData <- read.table(paste0("./rnor5/", gffFl), header=F, sep="\t", as.is=T)
gffData <- gffData[order(gffData[,4]),]
final <- rbind(final, gffData)
}
> write.table(final, file="rnor5_final.gff", row.names=F, col.names=F, quote=F, sep="\t")
```

Important: The final annotation for each species MUST be in the same chromosomal order as the genome.fa file for that species, otherwise the final GFF file will be sorted improperly and many gene quantification tools will fail to work. Additionally, chromosomes must be in the same format as in genome.fa for that species (eg, chr10 vs 10).

Finally, the GFF files should be converted to GTF format. GTF is essentially a simplified, more specific form of the GFF format. The Cufflinks package comes with a utility for performing this conversion, *gffread*.

```
$ gffread mm10_final.gff -T -o mm10.gtf
$ gffread rn5_final.gff -T -o rnor5.gtf
```

These GTF files are then used together with the BAM files generated previously to quantify the expression at a per-gene level.

Counting of gene features. The next step is to quantify the expression for each sample, which is done using the Bioconductor package *Rsubread*. The package takes the BAM files and the GTF annotation, as well as some other parameters describing the data, and produces a count table of each gene ID. As the count data are returned as a variable to the R environment rather than written to a file, and as *Rsubread* outputs information such as the number and percent of successfully counted reads, it may be advisable to use a script for counting and to redirect terminal output to a file. The R commands used for counting all the data in this experiment, as well as for saving the R environment with all count data for later analysis, are shown in the Supplementary Code Snippets (#1).

This last command saves all the data, from all the samples, to a file that can be reloaded by R at any time. This is useful when analyses are done on a computer that a researcher may not have access all the time or when further analysis may be desired at a later time. Unless saved, the data are deleted when the R environment is closed.

The data are then prepared for loading into *edgeR*. The input format for *edgeR* is a matrix with counts for each sample for each gene ID. It may simplify the analyses to extract these to their own text files using R (Supplementary Code Snippets #2).

These commands do additional sorting and filtering to ensure that all genes for all samples are in the correct order, and there are no gene IDs that do not exist for all samples.

Differential expression analysis. Once the files with the counts have been prepared, they can be analyzed using *edgeR* for differential expression. *EdgeR* is chosen in particular for the differential expression analysis for several useful features in the following. First, it supports input in the form of a single matrix of counts with gene IDs as the names for each row, allowing easy integration of the counted data from the previous step. Second, it has superb support for complex comparisons and experimental design. It is trivial to compare individual samples, or specify groups for comparison, or even to make two comparisons, and compare the results of that comparison against one another, allowing for essentially any dimension of analysis desired.

To load the count data into *edgeR* and then build the labeled experiment design, estimate the count dispersions, and build a fitted model, the following commands are used that are defined in an R script in the github repo mentioned earlier. Full code for the downstream analysis is available upon request. Due to the length of the code, complete commands have been omitted from this document.

The generated experimental design matrix (specifying which samples/replicates in the "counts.txt" file to include under which labels) appears as follows:

```
> design
mmast mmneu mmopc rnast rnneu rnopc
1  0 0 0 1 0 0
2  0 0 0 1 0 0
3  0 0 0 0 1 0
4  0 0 0 0 1 0
5  0 0 0 0 0 1
6  0 0 0 0 0 1
7  0 0 1 0 0 0
8  0 0 1 0 0 0
9  0 1 0 0 0 0
10 0 1 0 0 0 0
11 1 0 0 0 0 0
12 1 0 0 0 0 0
attr(,"assign")
[1] 1 1 1 1 1 1
attr(,"contrasts")
attr(,"contrasts")$Group
[1] "contr.treatment
```

While not strictly necessary, a design matrix is exceptionally useful as it allows comparisons to be made with the *makeContrasts()* function in *edgeR*, specifying groups to compare by their group name, rather than manually entering number values representing which columns to include at certain weights. The following commands are used to define various comparisons that would be made among the data, each of which would have its own differential gene expression analysis (Supplementary Code Snippets #3).

These comparisons allow for the determination of differentially expressed genes both generally across all cell types as well as on a per-cell-type basis. Comparisons of cellular differences between individual species as well as comparisons between cell types are also defined. Not all the above comparisons wound up having their data used for the final reporting of results. However, the presence of many of these comparisons allowed for additional error checking of the data. Additionally, as analyses are done, certain comparisons that had not been of interest before may turn to be of interest. Overall, it is useful to have all the data preprocessed and ready to go if it should be decided that any subset of it may be needed, rather than adding onto already existing data structures after the fact.

The next step is to use *glmLRT()* to find all differentially expressed genes, which is done by looping through the data frame of comparisons. The result of each comparison is saved in a list for later use.

The table of fold changes and false discovery rate (FDR)-corrected *P*-values is used by *GAGE* and *SPIA* to perform KEGG pathway enrichment. Note that native *GAGE* and *SPIA* pathway enrichment analyses consider only the fold change values and do not consider the number of samples that went into the fold change comparison. This causes the *q*-value outputs from *GAGE* to be falsely inflated. To account for this, the *edgeR* differential expression table is filtered to only include genes with a *P*-value of < 0.05 and a FDR of < 0.01. Only these significantly expressed genes with a low FDR are used in the calculation of enriched pathways. All KEGG pathway IDs found with a *P*-value < 0.05 are then returned and also saved to a list for further use.

In cross-species analyses, simple comparisons – for example, comparing the rat astrocytes directly against the mouse astrocytes – may present skewed data, as artifacts introduced by the cross-species annotation are not controlled in this comparison. To tackle this issue, in the *ast.rnVSmm* and other comparisons specified above, individual cell type comparisons are controlled against the average expression of all cell types in that species. This means that in the resulting comparison, only the genes that are differently expressed in one cell type between two species are reported. In other words, the expression for each cell type is compared to the overall profile for that species, and the results of those comparisons are compared across species. In this way, when comparing a cell type across species, only those genes that are significantly different from the average gene expression of that species are compared across species.

Visualization of data. Using the gene expression and pathway data contained in the lists described earlier, the data are graphed and visualized. First, *pathview* is used to visualize differentially enriched pathways, downloading the pathway map from KEGG, and then color coding each gene with the fold change differences. The general gene expression profiles are visualized using R scripts written for this purpose. The gene expression values for each gene are quantified from the gene read counts with $\log_2$(counts per million(CPM) + 1). Several charts are generated from this. Second, a heatmap showing the 200 genes with the highest expression in any sample are presented, sorted by similarity of expression across all samples. The $\log_2$(CPM + 1) table is then regenerated, using mean counts per species cell type, as divided in the experimental design earlier. Third, a heatmap is drawn with the top 200 expressed genes of any cell type, sorted by similarity of expression across all cell types. Finally, a heatmap is drawn with the top 25 expressed genes for each of the six cell types in turn, sorted top to bottom by expression level in that cell type. These heatmaps illustrate the similarities and differences among the cell types examined, allowing for easy visual identification of potential problems in the data that may not have been clear earlier in the protocol. For instance, if two replicates of the same cell type present extremely different expression profiles, the heatmaps would send a warning signal about the reproducibility of the data. These heatmaps are drawn using the *gplots* R package. In the heatmaps, the gene symbol names are displayed next to each row. The cross-species annotation uses exclusively Ensembl gene IDs. The *biomaRt* Bioconductor package is used to translate these IDs to gene symbols.

Furthermore, Venn diagrams are generated using the *VennDiagram* R package to visualize which genes among the lists of DEGs are in common or different between various comparisons. These Venn diagrams have the potential to grant further biological insight.

Pathway analysis and visualization. While lists of differentially expressed genes are a detailed and robust way to represent differences in expression between two samples, they are not a very friendly format for humans to understand. A table of thousands of gene ID tags, each with individual expression values, is not easy to read and to visually extrapolate to biological significance. To this end, *GAGE* and *SPIA* are used in this protocol to analyze the previously generated gene lists and determine the differentially expressed pathways. Unfortunately, neither of these software packages has the capacity to properly deal with unequal input samples (eg, comparing a group of two samples from one species against a group of three samples from another species). For this reason, the FDR values reported by either of these software packages may not be accurate. To deal with this, the FDR values (q-value) for individual genes provided by *edgeR* are used, and the pathway output of *SPIA* and *GAGE* is filtered by P-value (<0.05). When these pathways are passed into *pathview* for visualization, only the genes with a q-value less than 0.05 are provided. So in the final pathway charts, only significant genes are used. It should be noted that the *SPIA* package provided by Bioconductor has many of its significance values hardcoded and lacks handling for some data aberrations such as NA values in R. We downloaded and modified the source code of this package to suit the purposes of this experiment. As *SPIA* is licensed under the GPL, this modified source code should be made available to the general public. It has been hosted in the github repo mentioned earlier. Scripts are written to perform both *GAGE* and *SPIA* pathway enrichments on all the comparisons mentioned earlier. As both of these packages take Entrez IDs as input and the genes are listed by Ensembl ID, a script is written to leverage Bioconductor's *org.Mm.eg.db* package to convert the IDs.

Once lists of significantly enriched pathways have been generated, the KEGG IDs of enriched pathways for each comparison are fed into *pathview*, which queries each pathway ID against the KEGG database and downloads the PNG and XML files for that pathway map. It also takes the list of DEGs and logarithms of fold change values for that comparison and color codes genes on the pathway map to illustrate which parts of that pathway are differently expressed and in what direction. These images are then saved to files for manual examination. Furthermore, as each list of differentially expressed pathways is produced, tables of all pathway names and significance values are saved for future reporting.

Discussion

Here we describe, to the best of our knowledge, the first comprehensive protocol for the comparison of RNA-seq data between species. The novelty of this should be emphasized. While other methods have been developed for interspecies comparisons, a comprehensive protocol that walks users through the analysis process does not exist. By thoroughly explaining and documenting each step, it is easy for a person without prior experience to comprehend and follow this protocol rather than attempt to run or even develop it on his/her own.

Compared to other previously published methods, this protocol has a number of strengths. For example, the generation of a cross-species annotation enables the use of other commonly used downstream analysis tools. This means that the cross-species annotation can be easily integrated into existing pipelines for automation. The downstream analysis methods used in this protocol are essentially identical to those used in within-species RNA-seq data analysis. Most common downstream analysis tools will be able to natively support the sorts of comparisons necessary for comparing between species with minimal effort or changes to existing workflows.

Furthermore, the use of UCSC axt files means that huge numbers of comparisons are possible. Any species can theoretically be compared to any other species so long as an alignment exists between their genomes, although for distant species the quality of the comparison may be questionable. The axt files can be used as a drop-in to create annotations for as many species as is desired. Additionally, they provide a standardized format, which again proves useful in automation.

Author Contributions

Conceived and designed the experiments: FC. Analyzed the data: PRL. Wrote the first draft of the manuscript: PRL, FC. Contributed to the writing of the manuscript: PRL, FC. Agreed with manuscript results and conclusions: PRL, FC. Jointly developed the structure and arguments for the paper: PRL, FC. Made critical revisions and approved the final version: PRL, FC. Both the authors reviewed and approved the final manuscript.

Acknowledgment

An earlier version of this research was presented by Peter R. LoVerso in fulfillment of the requirements of his M.S. in Bioinformatics at Rochester Institute of Technology.

Supplementary Material

Supplementary Code Snippets #1.

Supplementary Code Snippets #2.

Supplementary Code Snippets #3.

REFERENCES

1. Schena M, Shalon D, Davis RW, Brown PO. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*. 1995;270(5235):467–70.
2. Lashkari DA, DeRisi JL, McCusker JH, et al. Yeast microarrays for genome wide parallel genetic and gene expression analysis. *Proc Natl Acad Sci U S A*. 1997;94(24):13057–62.
3. Mortazavi A, William BA, McCue K, Schaeffer L, Wold B. Mapping and quantifying mammalian transcriptomes by RNA-seq. *Nat Methods*. 2008;5:621–8.
4. Wang Z, Gerstein M, Snyder M. RNA-seq: a revolutionary tool for transcriptomics. *Nat Rev Genet*. 2009;10:57–63.
5. Marioni JC, Mason CE, Mane SM, Stephens M, Gilad Y. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res*. 2008;18:1509–17.
6. Bradford JR, Hey Y, Yates T, Li Y, Pepper SD, Miller CJ. A comparison of massively parallel nucleotide sequencing with oligonucleotide microarrays for global transcription profiling. *BMC Genomics*. 2010;11:282.
7. Agarwal A, Koppstein D, Rozowsky J, et al. Comparison and calibration of transcriptome data from RNA-seq and tiling array. *BMC Genomics*. 2010;11:383.
8. Liu S, Lin L, Jiang P, Wang D, Xing Y. A comparison of RNA-seq and high-density exon array for detecting differential gene expression between closely related species. *Nucleic Acids Res*. 2010;39:578–88.
9. Bloom JS, Khan Z, Kruglyak L, Singh M, Caudy AA. Measuring differential gene expression of short read sequencing: quantitative comparison to 2-channel gene expression microarray. *BMC Genomics*. 2009;10:221.
10. Malone JH, Oliver B. Microarray, deep sequencing and the true measure of the transcriptome. *BMC Biol*. 2011;9:34.
11. Cheok MH, Yang W, Pui CH, et al. Treatment-specific changes in gene expression discriminate *in vivo* drug response in human leukemia cells. *Nat Genet*. 2003;34:85–90.
12. Arbeitman MN, Furlong EE, Imam F, et al. Gene expression during the life cycle of *Drosophila melanogaster*. *Science*. 2002;297:2270–5.
13. Spellman PT, Sherlock G, Zhang MQ, et al. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol Biol Cell*. 1998;9:3273–97.
14. Kai T, Williams D, Spradling AC. The expression profile of purified *Drosophila* germline stem cells. *Dev Biol*. 2005;283:486–502.
15. Chan ET, Quon GT, Chua G, et al. Conservation of core gene expression in vertebrate tissues. *J Biol*. 2009;8:33.
16. Alizadeh AA, Eisen MB, Davis RE, et al. Distinct types of disuse large B-cell lymphoma identified by gene expression profiling. *Nature*. 2000;403:503–11.
17. Golub TR, Slonim DK, Tamayo P, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*. 1999;286:531–537.
18. Khaitovich P, Enard W, Lachmann M, Paabo S. Evolution of primate gene expression. *Nat Rev Genet*. 2006;7:693–702.
19. Gilad Y, Borevitz J. Using DNA microarray to study natural variation. *Curr Opin Genet Dev*. 2006;16:553–8.
20. Preuss TM, Caceres M, Oldham MC, Geschwind DH. Human brain evolution: insights from microarrays. *Nat Rev Genet*. 2004;5:850–860.
21. Segal E, Friedman N, Kaminski N, Regev A, Koller D. From signatures to models: understanding cancer using microarrays. *Nat Genet*. 2005;37:S38-S45.
22. Sweet-Cordero A, Mukherjee S, Subramanian A, et al. An oncogenic KRAS2 expression signature identified by cross-species gene-expression analysis. *Nat Genet*. 2005;37:48–55.
23. Miller JA, Horvath S, Geschwind DH. Divergence of human and mouse brain transcriptome highlights Alzheimer disease pathway. *Proc Natl Acad Sci U S A*. 2010;107:220–9.
24. Kuhn A, Goldstein DR, Hodges A, et al. Mutant huntingtin's effects on striatal gene expression in mice recapitulate changes observed in human Huntington's disease brain and do not differ with mutant huntingtin length or wild-type huntingtin dosage. *Hum Mol Genet*. 2007;16:1845–61.
25. Rasche A, Al-Hasani H, Herwig R. Meta-analysis approach identifies candidate genes and associated molecular networks for type-2 diabetes mellitus. *BMC Genomics*. 2008;9:310.
26. Marques FZ, Campain AE, Yang YHJ, Morris BJ. Meta-analysis of genome-wide gene expression differences in onset and maintenance phases of genetic hypertension. *Hypertension*. 2010;56:319–24.
27. Ginis I, Luo Y, Miura T, et al. Differences between human and mouse embryonic stem cells. *Dev Biol*. 2004;269:360–80.
28. McCarroll SA, Murphy CT, Zou S, et al. Comparing genomic expression patterns across species identified shared transcriptional profile in aging. *Nat Genet*. 2004;36:197–204.
29. Pan F, Chiu CH, Pulapura S, et al. Gene Aging Nexus: a web database and data mining platform for microarray data on aging. *Nucleic Acids Res*. 2007;35:D756–9.
30. De Magalhaes JP, Curado J, Church GM. Meta-analysis of age-related gene expression profiles identifies common signature of aging. *Bioinformatics*. 2009;25:875–81.
31. Okyere J, Oppon E, Dzidzienyo D, Sharma L, Ball G. Cross-species gene expression analysis of species specific differences in the preclinical assessment of pharmaceutical compounds. *PLoS One*. 2014;9:e96853.
32. Gunnarsson L, Kristiansson E, Forlin L, Nerman O, Larsson DGJ. Sensitive and robust gene expression changes in fish exposed to estrogen – a microarray approach. *BMC Genomics*. 2007;8:149.
33. Ung CY, Lam SH, Hlaing MM, et al. Mercury-induced hepatotoxicity in zebrafish: in vivo mechanistic insights from transcriptome analysis, phenotype anchoring and targeted gene expression validation. *BMC Genomics*. 2010;11:212.
34. Kuhn A, Luthi-Carter R, Delorenzi M. Cross-species and cross-platform gene expression studies with the bioconductor-compliant R package 'annotationTools'. *BMC Bioinformatics*. 2008;9:26.
35. Kristiansson E, Österlund T, Gunnarsson L, Arne G, Larsson DG, Nerman O. A novel method for cross-species gene expression analysis. *BMC Bioinformatics*. 2013;14:70.
36. Zhu Y, Li M, Sousa AMM, Sestan N. XSAnno: a framework for building ortholog models in cross-species transcriptome comparison. *BMC Genomics*. 2014;15:343.
37. Barrett T, Troup DB, Wilhite SE, et al. NCBI GEO: archive for functional genomics data sets – 10 years on. *Nucleic Acids Res*. 2011;39:D1005–10.
38. Parkinson H, Sarkans U, Kolesnikov N, et al. ArrayExpress updates – an archive of microarray and high-throughput sequencing-based functional genomics experiments. *Nucleic Acids Res*. 2011;39:D1002–4.
39. Anders S, McCarthy DJ, Chen Y, et al. Count-based differential expression analysis of RNA sequencing data using R and Bioconductor. *Nat Protoc*. 2013;8(9):1765–86.
40. Trapnell C, Roberts A, Goff L, et al. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat Protoc*. 2012;7:562–78.
41. Garber M, Grabherr MG, Guttman M, Trapnell C. Computational methods for transcriptome annotation and quantification using RNA-seq. *Nat Methods*. 2011;8(6):469–77.
42. Rumble SM, Lacroute P, Dalca AV, Fiume M, Sidow A, Brudno M. SHRiMP: accurate mapping of short color-space reads. *PLoS Comput Biol*. 2009;5(5):e10003386.
43. Trapnell C, Williams BA, Pertea G, et al. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotechnol*. 2010;28(5):511–5.
44. Wu TD, Nacu S. Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*. 2010;26(7):873–81.
45. Li H, Handsaker B, Wysoker A, et al; 1000 Genome Project Data Processing Subgroup. The sequence alignment/map format and SAMtools. *Bioinformatics*. 2009;25(16):2078–9.
46. Harrow J, Denoeud F, Frankish A, et al. GENCODE: producing a reference annotation for ENCODE. *Genome Biol*. 2006;7(suppl 1):S4.1–4.9.
47. Meyer LR, Zweig AS, Hinrichs AS, et al. The UCSC genome browser database: extensions and updates 2013. *Nucleic Acids Res*. 2013;41(Database issue):D64–9.

48. Katz Y, Wang ET, Airoldi EM, Burge CB. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nat Methods*. 2010;7(12):1009–15.

49. Chiaromonte F, Yap VB, Miller W. Scoring pairwise genomic sequence alignments. *Pac Symp Biocomput*. 2002;7:115–26.

50. Kent WJ, Baertsch R, Hinrichs A, Miller W, Haussler D. Evolution's cauldron: duplication, deletion, and rearrangement in the mouse and human genomes. *Proc Natl Acad Sci U S A*. 2003;100(20):11484–9.

51. Schwartz S, Kent WJ, Smit A, et al. Human – mouse alignments with BLASTZ. *Genome Res*. 2003;13(1):103–7.

52. Liao Y, Smyth GK, Shi W. The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote. *Nucleic Acids Res*. 2013;41(10):1–17.

53. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*. 2009;26(1):139–40.

54. McCarthy DJ, Chen Y, Smyth GK. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Res*. 2012;40(10):4288–97.

55. Robinson MD, Smyth GK. Small-sample estimation of negative binomial dispersion, with applications to SAGE data. *Biostatistics*. 2008;9(2):321–32.

56. Tarca AL, Draghici S, Khatri P, et al. A novel signaling pathway impact analysis. *Bioinformatics*. 2008;25(1):75–82.

57. Luo W, Friedman MS, Shedden K, Hankenson KD, Woolf PJ. GAGE: generally applicable gene set enrichment for pathway analysis. *BMC Bioinformatics*. 2009;10:161.

58. Luo W, Brouwer C. Pathview: an R/bioconductor package for pathway-based data integration and visualization. *Bioinformatics*. 2013;29(14):1830–1.

59. Gentleman RC, Carey VJ, Bates DM, et al. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol*. 2004;5(10):R80.

60. Kanehisa M, Goto S. Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res*. 2000;28(1):27–30.

61. LoVerso PR, Wachter CM, Cui F. Cross-species transcriptomic comparison of *in vitro* and *in vivo* mammalian neural cells. *Bioinfo Biol Insights*. 2015;9:153–64.