Journal of Visualized Experiments

A bioinformatics pipeline for investigating molecular evolution and gene expression using RNA-seq

--Manuscript Draft--

Article Type:	Invited Methods Article - JoVE Produced Video
Manuscript Number:	JoVE61633R1
Full Title:	A bioinformatics pipeline for investigating molecular evolution and gene expression using RNA-seq
Corresponding Author:	Aide Macias-Muñoz University of California Irvine Irvine, CA UNITED STATES
Corresponding Author's Institution:	University of California Irvine
Corresponding Author E-Mail:	amaciasm@uci.edu
Order of Authors:	Aide Macias-Muñoz
	Ali Mortazavi
Additional Information:	
Question	**Response**
Please indicate whether this article will be Standard Access or Open Access.	Standard Access (US$2,400)
Please indicate the **city, state/province, and country** where this article will be **filmed**. Please do not use abbreviations.	Irvine, CA or Peoria, AZ
Please confirm that you have read and agree to the terms and conditions of the author license agreement that applies below:	I agree to the Author License Agreement
Please specify the section of the submitted manuscript.	Biology
Please provide any comments to the journal here.	

TITLE:

A bioinformatics pipeline for investigating molecular evolution and gene expression using RNA-seq

AUTHORS AND AFFILIATIONS:

Aide Macias-Muñoz[1*], Ali Mortazavi[1,*]

[1] Department of Developmental and Cell Biology, University of California, Irvine, CA, U.S.A.

*Co-corresponding authors:
Ali Mortazavi, ali.mortazavi@uci.edu
Aide Macias-Muñoz, amaciasm@uci.edu

KEYWORDS:

bioinformatics, gene expansions, BLAST, transcriptome, genome, MEGA

SUMMARY:

The purpose of this protocol is to investigate the evolution and expression of candidate genes using RNA sequencing data.

ABSTRACT:

Distilling and reporting large datasets, such as whole genome or transcriptome data, is often a daunting task. One way to break down results is to focus on one or more gene families that are significant to the organism and study. In this protocol, we outline bioinformatic steps to generate a phylogeny and to quantify the expression of genes of interest. Phylogenetic trees can give insight into how genes are evolving within and between species as well as reveal orthology. These results can be enhanced using RNA-seq data to compare the expression of these genes in different individuals or tissues. Studies of molecular evolution and expression can reveal modes of evolution and conservation of gene function between species. The characterization of a gene family can serve as a springboard for future studies and can highlight an important gene family in a new genome or transcriptome paper.

INTRODUCTION:

Advances in sequencing technologies have facilitated the sequencing of genomes and transcriptomes of non-model organisms. In addition to the increased feasibility of sequencing DNA and RNA from many organisms, an abundance of data is publicly available to study genes of interest. The purpose of this protocol is to provide bioinformatic steps to investigate the molecular evolution and expression of genes that may play an important role in the organism of interest.

Investigating the evolution of a gene or gene family can provide insight into the evolution of biological systems. Members of a gene family are typically determined by identifying conserved motifs or homologous gene sequences. Gene family evolution was previously investigated using genomes from distantly related model organisms[1]. A limitation to this approach is that it is not

clear how these gene families evolve in closely related species and the role of different environmental selective pressures. In this protocol, we include a search for homologs in closely related species. By generating a phylogeny at a phylum level, we can note trends in gene family evolution such as that of conserved genes or lineage-specific duplications. At this level, we can also investigate whether genes are orthologs or paralogs. While many homologs likely function similarly to each other, that is not necessarily the case[2]. Incorporating phylogenetic trees in these studies is important to resolve whether these homologous genes are orthologs or not. In eukaryotes, many orthologs retain similar functions within the cell as evidenced by the ability of mammalian proteins to restore the function of yeast orthologs[3]. However, there are instances where a non-orthologous gene carries out a characterized function[4].

Phylogenetic trees begin to delineate relationships between genes and species, yet function cannot be assigned solely based on genetic relationships. Gene expression studies combined with functional annotations and enrichments provide stronger support for gene function. Cases where gene expression can be quantified and compared across individuals or tissue types can be more telling of potential function. The following protocol follows methods used in investigating opsin genes in *Hydra vulgaris*[7] but they can be applied to any species and any gene family. The results of such studies provide a foundation for further investigation into gene function and gene networks in non-model organisms. As an example, the investigation of the phylogeny of opsins, which are proteins that initiate the phototransduction cascade, gives context to the evolution of eyes and light detection[8–11]. In this case, non-model organisms especially basal animal species such as cnidarians or ctenophores can elucidate conservation or changes in the phototransduction cascade and vision across clades[12–14]. Similarly, determining the phylogeny, expression, and networks of other gene families will inform us about the molecular mechanisms underlying adaptations.

PROTOCOL:

This protocol follows UC Irvine animal care guidelines.

1. RNA-seq library preparation

1.1. Isolate RNA using the following methods.

1.1.1. Collect samples. If RNA is to be extracted at a later time, flash freeze the sample or place in RNA storage solution[15] (**Table of Materials**).

1.1.2. Euthanize and dissect the organism to separate tissues of interest.

1.1.3. Extract total RNA using an extraction kit and purify the RNA using an RNA purification kit (**Table of Materials**)

NOTE: There are protocols and kits that may work better for different species and tissue types[16,17]. We have extracted RNA from different body tissues of a butterfly[18] and a gelatinous *Hydra*[19] (see discussion).

1.1.4. Measure the concentration and quality of the RNA of each sample (**Table of Materials**). Use samples with RNA integrity numbers (RIN) higher than 8, ideally closer to 9[20] to construct cDNA libraries.

1.2. Construct cDNA library and sequence as follows.

1.2.1. Build cDNA libraries according to the library prep instruction manual (see discussion).

1.2.2. Determine cDNA concentration and quality (**Table of Materials**).

1.2.3. Multiplex the libraries and sequence them.

2. Access a computer cluster

NOTE: RNA-seq analysis requires manipulation of large files and is best done on a computer cluster (**Table of Materials**).

2.1. Login to the computer cluster account using the command **ssh username@clusterlocation** on a terminal (Mac) or PuTTY (Windows) application window.

3. Obtain RNA-seq reads

3.1. Obtain RNA-seq reads from the sequencing facility or, for data generated in a publication, from the data repository where it was deposited (3.2 or 3.3).

3.2. To download data from repositories such as ArrayExpress do the following:

3.2.1. Search the site using the accession number.

3.2.2. Find the link to download the data, then left-click and select **Copy Link**.

3.2.3. On the terminal window, type **wget** and select **Paste link** to copy the data to the directory for analysis.

3.3. To download NCBI Short Read Archive (SRA) data follow these alternative steps:

3.3.1. On the terminal download SRA Toolkit v. 2.8.1 using **wget**.

NOTE: Downloading and installing programs to the computer cluster may require root access, contact your computer cluster administrator if installation fails.

3.3.2. Finish installing the program by typing **tar -xvf $TARGZFILE**.

3.3.3. Search NCBI for the SRA accession number for the samples you want to download, it should have the format SRRXXXXXX.

3.3.4. Obtain the RNA-seq data by typing **[sratoolkit location]/bin/prefetch SRRXXXXXX** to the terminal window.

3.3.5. For paired end files type **[sratoolkit location]/bin/fastq-dump --split-files SRRXXXXXX** to get two fastq files (SRRXXXXXX_1.FASTQ and SRRXXXXXX_2.FASTQ).

NOTE: To do a Trinity de novo assembly use the command **[sratoolkit location]/bin/fastq-dump --defline-seq '@$sn[_$rn]/$ri' --split-files SRRXXXXXX**

4. Trim adapters and low-quality reads (optional)

4.1. Install or load Trimmomatic[21] v. 0.35 in the computing cluster.

4.2. In the directory where the RNA-seq data files are located, type a command that includes the location of the trimmomatic jar file, the input FASTQ files, output FASTQ files, and optional parameters such as read length and quality.

NOTE: The command will vary by the raw and desired quality and length of the reads. For Illumina 43 bp reads with Nextera primers, we used: **java -jar /data/apps/trimmomatic/0.35/trimmomatic-0.35.jar PE $READ1.FASTQ $READ2.FASTQ paired_READ1.FASTQ unpaired_READ1.FASTQ paired_READ2.FASTQ unpaired_READ2.FASTQ ILLUMINACLIP:adapters.fa:2:30:10 LEADING:20 TRAILING:20 SLIDINGWINDOW:4:17 MINLEN:30.**

5. Obtain reference assembly

5.1. Search google, EnsemblGenomes, and NCBI Genomes and Nucleotide TSA (Transcriptome Shotgun Assembly) for a reference genome or assembled transcriptome for the species of interest (**Figure 1**).

NOTE: If a reference genome or transcriptome are not available or low-quality, proceed to STEP 6 to generate a de novo assembly.

5.2. If a reference genome or assembled transcriptome exists, download it as a fasta file to where the analysis will be performed following the steps below.

5.2.1. Find the link to download the genome, left-click and **Copy Link**.

5.2.2. On the terminal window type **wget** and paste the link address. If available, also copy the GTF file and protein FASTA file for the reference genome.

6. Generate a de novo assembly (Alternative to Step 5)

6.1. Combine the RNA-seq READ1 and READ2 fastq files for all samples by typing **cat *READ1.FASTQ > $all_READ1.FASTQ** and **cat *READ2.FASTQ > all_READ2.FASTQ** on the terminal window.

6.2. Install or load Trinity[22] v.2.8.5 on the computing cluster.

6.3. Generate and assembly by typing on the terminal: **Trinity --seqType fq --max_memory 20G --left $all_READ1.FASTQ --right $all_READ2.FASTQ.**

7. Map reads to the genome (7.1) or de novo transcriptome (7.2)

7.1. Map reads to the reference genome using STAR[23] v. 2.6.0c and RSEM[24] v. 1.3.0.

7.1.1. Install or load STAR v. 2.6.0c. and RSEM v. 1.3.0 to the computing cluster.

7.1.2. Index the genome by typing **rsem-prepare-reference --gtf $GENOME.GTF --star -p 16 $GENOME.FASTA $OUTPUT.**

7.1.3. Map reads and calculate expression for each sample by typing **rsem-calculate-expression -p 16 --star --paired-end $READ1.FASTQ $READ2.FASTQ $INDEX $OUTPUT.**

7.1.4. Rename the results file to something descriptive using **mv RSEM.genes.results $sample.genes.results.**

7.1.5. Generate a matrix of all counts by typing **rsem-generate-data-matrix *[genes/isoforms.results] > $OUTPUT**.

7.2. Map RNA-seq to the Trinity de novo assembly using RSEM and bowtie.

7.2.1. Install or load Trinity[22] v.2.8.5, Bowtie[25] v. 1.0.0, and RSEM v. 1.3.0.

7.2.2. Map reads and calculate expression for each sample by typing **[trinity_location]/align_and_estimate_abundance.pl --prep-reference --transcripts $TRINITY.FASTA --seqType fq --left $READ1.FASTQ --right $READ2.FASTQ --est_method RSEM --aln_method bowtie --trinity_mode --output_dir $OUTPUT.**

7.2.3. Rename the results file to something descriptive using **mv RSEM.genes.results $sample.genes.results.**

7.2.4. Generate a matrix of all counts by typing **[trinity_location]/abundance_estimates_to_matrix.pl --est_method RSEM *[genes/isoforms].results**

8. Identify genes of interest

NOTE: The following steps can be done with nucleotide or protein FASTA files but work best and are more straightforward with protein sequences. BLAST searches using protein to protein is more likely to give results when searching between different species.

8.1. For a reference genome, use the protein FASTA file from STEP 5.2.2 or see Supplemental Materials to generate a custom gene feature GTF.

8.2. For a de novo transcriptome, generate a protein FASTA using TransDecoder.

8.2.1. Install or load TransDecoder v. 5.5.0 on the computer cluser.

8.2.2. Find the longest open reading frame and predicted peptide sequence by typing **[Transdecoder location]/TransDecoder.LongOrfs -t $TRINITY.FASTA.**

8.3. Search NCBI Genbank for homologs in closely related species.

8.3.1. Open an internet browser window and go to https://www.ncbi.nlm.nih.gov/genbank/.

8.3.2. On the search bar type the name of the gene of interest and the name of closely related species which have been sequenced or genus or phylum. On the left of the search bar select protein then click search.

8.3.3. Extract sequences by clicking **Send to** and then select **File.** Under Format, select FASTA then click **Create File.**

8.3.4. Move FASTA file of homologs to the computer cluster by typing **scp $FASTA username@clusterlocation:/$DIR** on a local terminal window or use FileZilla to transfer files to and from computer and cluster.

8.4. Search for candidate genes using BLAST+[26].

8.4.1. Install or load BLAST+ v. 2.8.1 on the computer cluster.

8.4.2. On the computer cluster, make a BLAST database from the genome or transcriptome translated protein FASTA by typing **[BLAST+ location]/makeblastdb -in $PEP.FASTA -dbtype prot -out $OUTPUT**

8.4.3. BLAST the homologous gene sequences from NCBI to the database of the species of interest by typing **[BLAST+ location]/blastp -db $DATABASE -query $FASTA -evalue 1e-10 -outfmt 6 -max_target_seqs 1 -out $OUTPUT**.

8.4.4. View the output file using the command **more**. Copy unique gene IDs from the species of interest to a new text file.

8.4.5. Extract the sequences of candidate genes by typing **perl -ne 'if(/^>(\S+)/){$c=$i{$1}}$c?print:chomp;$i{$_}=1 if @ARGV' $gene_id.txt $PEP.FASTA > $OUTPUT.**

8.5. Confirm gene annotation using reciprocal BLAST.

8.5.1. On the internet browser go to https://blast.ncbi.nlm.nih.gov/Blast.cgi.

8.5.2. Select blastx, then paste the candidate sequences, select the Non-redundant protein sequence database and click **BLAST**.

8.6. Identify additional genes by annotating all genes in the genome or transcriptome with gene ontology (GO) terms (see discussion).

8.6.1. Transfer the protein FASTA to the local computer.

8.6.2. Download and install Blast2GO[27–29] v. 5.2 to the local computer.

8.6.3. Open **Blast2GO**, click **File**, go to **Load**, go to **Load Sequences**, click **Load Fasta File (fasta)**. Select the FASTA file and click **Load**.

8.6.4. Click on **Blast,** choose **NCBI Blast**, and click **Next**. Edit parameters or click **Next**, edit parameters and click **Run** to find the most similar gene description.

8.6.5. Click mapping then click **Run** to search Gene Ontology annotations for similar proteins.

8.6.6. Next click **interpro**, select **EMBL-EBI InterPro**, and click **Next**. Edit parameters or click **Next**, and click **Run** to search for signatures of known gene families and domains.

8.6.7. Export the annotations by clicking **File**, select **Export**, click **Export Table**. Click **Browse**, name the file, click **Save**, click **Export**.

8.6.8. Search the annotation table for GO terms of interest to identify additional candidate genes. Extract the sequences from the FASTA file (STEP 8.4.5)

9. Phylogenetic trees

9.1. Download and install MEGA[30] v. 7.0.26 to your local computer.

9.2. Open MEGA, click on **Align**, click **Edit/Build Alignment**, select **Create a new alignment** click **OK**, select **Protein**.

9.3. When the alignment window opens, click on **Edit**, click **Insert sequences from file** and select the FASTA with protein sequences of candidate genes and probable homologs.

9.4. Select all sequences. Find the arm symbol and hover over it. It should say Align sequences using MUSCLE[31] algorithm. Click on the arm symbol and then click **Align Protein** to align the sequences. Edit parameters or click **OK** to align using default parameters.

9.5. Visually inspect and make any manual changes then Save and close the alignment window.

9.6. In the main MEGA window, click on **Models**, click **Find Best DNA/Protein models (ML)**, select the alignment file and select corresponding parameters such as: **Analysis: Model Selection (ML), Tree to use: Automatic (neighbor-joining tree), Statistical Method: Maximum Likelihood, Substitution Type: Amino Acid, Gap/missing data treatment: Use all sites, Branch site filter: None.**

9.7. Once the best model for the data is determined, go to the main MEGA window. Click **Phylogeny** and click **Contruct/Test Maximum Likelihood Tree** and then select the alignment, if necessary. Select the appropriate parameters for the tree: **Statistical method: Maximum Likelihood, Test of Phylogeny: Bootstrap method with 100 replicates, substitution type: amino acid, model: LG with Freqs. (+F), rates among sites: gamma distributed (G) with 5 discrete gamma categories, gap/missing data treatment: use all sites, ML heuristic method: Nearest-Neighbor-Interchange (NNI).**

10. Visualize gene expression using TPM

10.1. For Trinity, on the computer cluster go to the directory where **abundance_estimates_to_matrix.pl** was run and one of the outputs should be matrix.TPM.not_cross_norm. Transfer this file to your local computer.

NOTE: See Supplemental Materials for cross sample normalization.

10.2. For TPMs from a genome analysis follow the steps below.

10.2.1. On the computer cluster, go to the RSEM installation location. Copy rsem-generate-data-matrix by typing **scp rsem-generate-data-matrix rsem-generate-TPM-matrix**. Use **nano** to edit the new file and change “my $offsite = 4” from 4 to 5 for TPM, it should now read “my $offsite = 5”.

10.2. Go to the directory where the RSEM output files .genes.results are and now use **rsem-generate-TPM-matrix *[genes/isoforms.results] > $OUTPUT** to generate a TPM matrix. Transfer results to a local computer.

10.3. Visualize the results in ggplot2.

10.3.1. Download R v. 4.0.0 and RStudio v. 1.2.1335 to a local computer.

10.3.1. Open RStudio on the right of the screen go to the **Packages** tab and click **Install**. Type **ggplot2** and click **install**.

10.3.2. On the R script window read in the TPM table by typing **data<-read.table("$tpm.txt",header = T)**

10.3.3. For bar graphs similar to **Figure 5** type something similar to:

```
p<- ggplot() + geom_bar(aes(y=TPM, x=Symbol, fill=Tissue), data=data, stat="identity")
fill<-c("#d7191c","#fdae61", "#ffffbf", "#abd9e9", "#2c7bb6")
p<-p+scale_fill_manual(values=fill)
p + theme(axis.text.x = element_text(angle = 90))
```

REPRESENTATIVE RESULTS:

The methods above are summarized in **Figure 1** and were applied to a data set of *Hydra vulgaris* tissues. *H. vulgaris* is a fresh-water invertebrate that belongs to the phylum *Cnidaria* which also includes corals, jellyfish, and sea anemones. *H. vulgaris* can reproduce asexually by budding and they can regenerate their head and foot when bisected. In this study, we aimed to investigate the evolution and expression of opsin genes in *Hydra*[7]. While *Hydra* lack eyes, they exhibit light-dependent behavior[32]. Opsin genes encode proteins that are important in vision to detect different wavelengths of light and begin the phototransduction cascade. Investigating the molecular evolution and expression of this gene family in a basal species can provide insight into the evolution of eyes and light detection in animals.

We generated a guided assembly using the Hydra 2.0[33] reference genome and publicly available RNA-seq data (GEO accession GSE127279) **Figure 1**. This step took approximately 3 days. Although we did not generate a de novo transcriptome in this case, a Trinity assembly can take up to 1 week to generate and each library can take a few hours for read mapping depending on the mapper. The merged *Hydra* assembly (~50,000 transcripts) was annotated using Blast2GO which took about 1-week **Figure 1**. Sequences for opsin-related genes were extracted into a fasta file. Sequences for opsin genes from other species were also extracted from NCBI GenBank. We used opsins from cnidarians *Podocoryna carnea*, *Cladonema radiatum*, *Tripedelia cystophora*, and *Nematostella vectensis*, and we also included outgroups *Mnemiopsis leidyi Drosophila melanogaster* and *Homo sapiens*. Opsin genes were aligned in MEGA7 **Figure 2**. By viewing the alignment, we were able to identify *Hydra* opsins that were missing a conserved lysine amino acid necessary to bind a light sensitive molecule. After visual inspection, we determined the best model by doing a model selection analysis. We generated a maximum-likelihood tree using the

model LG + G + F with bootstrap value of 100 **Figure 3**. For 149 opsin genes, the tree was finished in approximately 3 days. The phylogeny suggests opsin genes are evolving by lineage-specific duplications in cnidarians and potentially by tandem duplication in *H. vulgaris*[7].

We performed a differential expression analysis in edgeR and looked at absolute expression of opsin genes. We hypothesized that one or more opsins would be upregulated in the head (hypostome) and performed pair-wise comparisons of hypostome versus the body column, budding zone, foot and tentacles. As an example of a pair-wise comparison, 1,774 transcripts were differentially expressed between the hypostome and body column. We determined the genes that were upregulated across multiple comparisons and did a functional enrichment in Blast2GO **Table 1**. Grouping of G-protein coupled receptor activity included opsin genes. Finally, we looked at the absolute expression of opsin genes in different tissues, during budding and during regeneration by plotting their TPM values using ggplot **Figure 4**. Using the methods outlined here, we identified 2 opsin genes that did not group with the other opsins in the phylogeny, found one opsin that was expressed almost 200 times more than others, and we found a few opsin genes co-expressed with phototransduction genes that may be used for light detection.

FIGURES AND TABLES:

Figure 1. Workflow schematic. Programs used to analyze data on the computer cluster are in blue, in magenta are those that we used on a local computer and in orange is a web-based program. (1) Trim RNA-seq reads using trimmomatic v. 0.35. If a genome is available but gene models are missing, generate a guided assembly using STAR v. 2.6.0c and StringTie v. 1.3.4d. (Optional see Supplemental Materials) (2) Without a reference genome, use trimmed reads to make a de novo assembly using Trinity v 2.8.5. (3) To quantify gene expression using a reference genome, map reads using STAR and quantify using RSEM v. 1.3.1. Extract TPMs using RSEM and visualize them in RStudio. (4) Bowtie and RSEM can be used to map and quantify reads mapped to a trinity transcriptome. A Trinity script can be used to generate a TPM matrix to visualize counts in RStudio. (5) Use web-based NCBI BLAST and command-line BLAST+ to search for homologous sequences and confirm using reciprocal BLAST. Annotate genes further using Blast2GO. Use MEGA to align genes and generate a phylogenetic tree using the best fit model.

Figure 2. Example of aligned genes. Snapshot shows a portion of *Hydra* opsin genes aligned using MUSCLE. The arrow indicates the location of a retinal-binding conserved lysine.

Figure 3. Cnidarian opsin phylogenetic tree. Maximum-likelihood tree generated in MEGA7 using opsin sequences from *Hydra vulgaris*, *Podocoryna carnea*, *Cladonema radiatum*, *Tripedelia cystophora*, *Nematostella vectensis, Mnemiopsis leidyi, Trichoplax adhaerens*, *Drosophila melanogaster* and *Homo sapiens*.

Figure 4. Expression of Opsin genes in *Hydra vulgaris*. (A) Expression in transcripts per million (TPM) of *Hydra vulgaris* opsin genes in the body column, budding zone, foot, hypostome and

tentacles. (B) Expression of opsin genes during different stages of *Hydra* budding. (C) Expression of opsin genes of the *Hydra* hypostome during different time points of regeneration.

Table 1. Functional enrichment of genes upregulated in the hypostome

DISCUSSION:

The purpose of this protocol is to provide an outline of the steps for characterizing a gene family using RNA-seq data. These methods have been proven to work for a variety of species and datasets[4,34,35]. The pipeline established here has been simplified and should be easy enough to be followed by a novice in bioinformatics. The significance of the protocol is that it outlines all the steps and necessary programs to complete a publishable analysis. A crucial step in the protocol is having properly assembled full length transcripts this comes from high quality genomes or transcriptomes. To obtain proper transcripts, one needs high quality RNA and/or DNA and good annotations discussed below.

For RNA-seq library preparation, we include list kits that worked for small body parts of *Hydra*[19] and butterflies[18] (**Table of Materials**). We note that for low input RNA we used a modified protocol approach[36]. Methods for RNA extraction have been compared in multiple sample types including yeast cells[17], neuroblastoma[37], plants[38], and insect larvae[16] to name a few. We recommend the reader acquire a protocol that works for their species of interest, if any exist, or troubleshoot using commonly commercially available kits to start. For proper gene quantification, we recommend treating the RNA sample with DNase. The presence of DNA will affect proper gene quantification. We also recommend using a cDNA library prep kit that includes a polyA tail selection to select for mature mRNA. While rRNA depletion results in more read depth, the percentage of exon coverage is much lower than the exon coverage of RNA using polyA+ selection[39]. Finally, when possible it is best to use paired-end and stranded[40,41]. In the protocol above the read mapping commands will have to be modified when using single end reads.

As mentioned above it is important to be able to identify genes of interest and also to differentiate between recent gene duplications, alternative splicing, and haplotypes in sequencing. In some instances, having a reference genome can help by determining where genes and exons are located relative to each other. One thing to note is that if a transcriptome is obtained from a public database and is not high quality, it may be best to generate using Trinity[42] and combining RNA-seq libraries from tissues of interest. Likewise, if a reference genome does not have good gene models, RNA-seq libraries can be used to generate new GTFs using StringTie[43]. In addition, in cases where genes are incomplete and there is access to a genome, genes can be manually edited using homolog sequences then aligned to the genome using tblastn. The BLAST output can be used to determine the actual sequence, which may be different from the correction done using homologs. If there is no match, leave the sequence as was originally. When checking output pay attention to the genome coordinates to make sure the missing exon is indeed part of the gene.

Although we focus on software and programs that we used, modifications to this protocol exist due to many programs available which might work better for different datasets. As an example, we show commands for mapping reads to the transcriptome using bowtie and RSEM, but Trinity now has the option for much faster aligners such as kallisto[44] and salmon[45]. Similarly, we describe annotations using Blast2GO (now OmicsBox) but there are other mapper tools that can be found free and online. Some that we have tried include: GO FEAT[46], eggNOG-mapper[47, 48], and a very fast aligner PANNZER2[49]. To use these web-based annotation tools simply upload the peptide FASTA and submit. Standalone versions of PANNZER and eggNOG-mapper are also available to be downloaded to the computer cluster. Another modification is that we used MEGA and R on a local computer and used the online NCBI BLAST tool to do reciprocal BLASTs however both of these programs can be used on the computer cluster by downloading the necessary programs and databases. Likewise, aligners kallisto and salmon can be used on a local computer as long as a user has enough RAM and storage. However, FASTQ and FASTA files tend to be very large and we highly recommend using a computer cluster for ease and speed. In addition, while we provide instructions and links to download programs from their developers many of them can be installed from bioconda: https://anaconda.org/bioconda.

A common problem faced when doing bioinformatic analyses is shell scripts failing. This can be due to a variety of reasons. If an error file is created, these error file should be checked before troubleshooting. A few common reasons for an error are typos, missing key parameters, and compatibility issues between software versions. In this protocol, we include parameters for the data, but software manuals can provide more detailed guidelines for individual parameters. In general, it is best to use the most up to date versions of software and to consult the manual corresponding to that version.

Enhancements to this protocol include doing a transcriptome-wide differential expression analysis and functional enrichment analysis. We recommend edgeR[50] for differential expression analysis a package available in Bioconductor. For functional enrichment analysis, we have used Blast2GO[29] and web-based DAVID[51, 52]. We also recommend further editing the phylogeny by extracting it as a newick file and using web-based iTOL[53]. Furthermore, while this protocol will investigate the molecular evolution and expression patterns of genes, additional experiments can be used to validate gene or protein locations and functions. mRNA expression can be confirmed by RT-qPCR or in situ hybridization. Proteins can be localized using immunohistochemistry. Depending on the species, knockout experiments can be used to confirm gene function. This protocol can be used for a variety of objectives including, as shown above, to explore a gene family typically associated with photoreception in a basal species[7]. Another application of these methods is to identify changes in a conserved pathway under different selective pressures. As an example, these methods were used to discover variation in the expression of vision transient receptor potential channels between diurnal butterflies and nocturnal moths[34].

ACKNOWLEDGMENTS:

We thank Adriana Briscoe, Gil Smith, Rabi Murad and Aline G. Rangel for advice and guidance in incorporating some of these steps into our workflow. We are also grateful to Katherine Williams,

Elisabeth Rebboah, and Natasha Picciani for comments on the manuscript. This work was supported in part by a George E. Hewitt Foundation for Medical research fellowship to A.M.M.

DISCLOSURES:

The authors have nothing to disclose.

REFERENCES:

1. Lespinet, O., Wolf, Y.I., Koonin, E. V., Aravind, L. The role of lineage-specific gene family expansion in the evolution of eukaryotes. *Genome Research*. **12** (7), 1048–1059 (2002).
2. Gabaldón, T., Koonin, E. V. Functional and evolutionary implications of gene orthology. *Nature Reviews Genetics*. **14** (5), 360–366 (2013).
3. Dolinski, K., Botstein, D. *Orthology and Functional Conservation in Eukaryotes*. *Annual Review of Genetics*. **41** (1) (2007).
4. Macias-Muñoz, A., McCulloch, K.J., Briscoe, A.D. Copy number variation and expression analysis reveals a non-orthologous pinta gene family member involved in butterfly vision. *Genome Biology and Evolution*. **9** (12), 3398–3412 (2017).
5. Cannon, S.B., Mitra, A., Baumgarten, A., Young, N.D., May, G. The roles of segmental and tandem gene duplication in the evolution of large gene families in Arabidopsis thaliana. *BMC plant biology*. **4**, 10 (2004).
6. Eastman, S.D., Chen, T.H.P., Falk, M.M., Mendelson, T.C., Iovine, M.K. Phylogenetic analysis of three complete gap junction gene families reveals lineage-specific duplications and highly supported gene classes. *Genomics*. **87** (2), 265–274 (2006).
7. Macias-Munõz, A., Murad, R., Mortazavi, A. Molecular evolution and expression of opsin genes in Hydra vulgaris. *BMC Genomics*. **20** (1), 1–19 (2019).
8. Hisatomi, O., Tokunaga, F. Molecular evolution of proteins involved in vertebrate phototransduction. *Comparative Biochemistry and Physiology - B Biochemistry and Molecular Biology*. **133** (4), 509–522 (2002).
9. Arendt, D. Evolution of eyes and photoreceptor cell types. *International Journal of Developmental Biology*. **47**, 563–571 (2003).
10. Shichida, Y., Matsuyama, T. Evolution of opsins and phototransduction. *Philosophical Transactions of the Royal Society B: Biological Sciences*. **364** (1531), 2881–2895 (2009).
11. Porter, M.L. et al. Shedding new light on opsin evolution. *Proceedings of the Royal Society B: Biological Sciences*. **279** (1726), 3–14 (2012).
12. Plachetzki, D.C., Degnan, B.M., Oakley, T.H. The origins of novel protein interactions during animal opsin evolution. *PLoS ONE*. **2** (10), e1054 (2007).
13. Ramirez, M.D. et al. The last common ancestor of most bilaterian animals possessed at least nine opsins. *Genome Biology and Evolution*. **8** (12), 3640–3652 (2016).
14. Schnitzler, C.E. et al. Genomic organization, evolution, and expression of photoprotein and opsin genes in *Mnemiopsis leidyi*: a new view of ctenophore photocytes. *BMC Biology*. **10**, 107 (2012).
15. Pedersen, K.B., Williams, A., Watt, J., Ronis, M.J. Improved method for isolating high-quality RNA from mouse bone with RNAlater at room temperature. *Bone Reports*. **11** (January), 100211 (2019).
16. Ridgeway, J.A., Timm, A.E., Fallon, A. Comparison of RNA isolation methods from insect

larvae. *Journal of Insect Science*. **14** (1), 4–8 (2014).
17. Scholes, A.N., Lewis, J.A. Comparison of RNA isolation methods on RNA-Seq: Implications for differential expression and meta-Analyses. *BMC Genomics*. **21** (1), 1–9 (2020).
18. Briscoe, A.D. et al. Female behaviour drives expression and evolution of gustatory receptors in butterflies. *PLoS genetics*. **9** (7), e1003620 (2013).
19. Murad, R., Macias-Muñoz, A., Wong, A., Ma, X., Mortazavi, A. Integrative analysis of *Hydra* head regeneration reveals activation of distal enhancer-like elements. *bioRxiv*. 544049 (2019).
20. Gallego Romero, I., Pai, A.A., Tung, J., Gilad, Y. Impact of RNA degradation on measurements of gene expression. *BMC Biology*. **12**, 42 (2014).
21. Bolger, A.M., Lohse, M., Usadel, B. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*. **30** (15), 2114–2120 (2014).
22. Trinity, I., RNA-Seq De novo Assembly Using Trinity. 1–7 (2014).
23. Dobin, A. et al. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*. **29**, 15–21 (2013).
24. Li, B., Dewey, C.N. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*. **12**, 323 (2011).
25. Langmead, B., Trapnell, C., Pop, M., Salzberg, S.L. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biology*. **10**, R25 (2009).
26. Camacho, C. et al. BLAST+: architecture and applications. *BMC Bioinformatics*. **10**, 421 (2009).
27. Conesa, A., Götz, S. Blast2GO: A comprehensive suite for functional analysis in plant genomics. *International Journal of Plant Genomics*. **619832** (2008).
28. Conesa, A. et al. Blast2GO: A universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics*. **21** (18), 3674–3676 (2005).
29. Götz, S. et al. High-throughput functional annotation and data mining with the Blast2GO suite. *Nucleic Acids Research*. **36** (10), 3420–3435 (2008).
30. Kumar, S., Stecher, G., Tamura, K. MEGA7: Molecular Evolutionary Genetics Analysis version 7.0 for bigger datasets. *Molecular biology and evolution*. **33** (7), 1870–1874 (2016).
31. Edgar, R.C. MUSCLE: Multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*. **32** (5), 1792–1797 (2004).
32. Taddei-Ferretti, C., Musio, C., Santillo, S., Cotugno, A. The photobiology of *Hydra*'s periodic activity. *Hydrobiologia*. **530/531**, 129–134 (2004).
33. Chapman, J.A. et al. The dynamic genome of *Hydra*. *Nature*. **464** (7288), 592–596 (2010).
34. Macias-Muñoz, A., Rangel Olguin, A.G., Briscoe, A.D. Evolution of phototransduction genes in Lepidoptera. *Genome Biology and Evolution*. **11** (8), 2107–2124 (2019).
35. Macias-Munõz, A., Murad, R., Mortazavi, A. Molecular evolution and expression of opsin genes in Hydra vulgaris. *BMC Genomics*. **20** (1) (2019).
36. Picelli, S. et al. Full-length RNA-seq from single cells using Smart-seq2. *Nature Protocols*. **9** (1), 171–181 (2014).
37. Tavares, L., Alves, P.M., Ferreira, R.B., Santos, C.N. Comparison of different methods for DNA-free RNA isolation from SK-N-MC neuroblastoma. *BMC research notes*. **4**, 3 (2011).
38. Johnson, M.T.J. et al. Evaluating Methods for Isolating Total RNA and Predicting the Success of Sequencing Phylogenetically Diverse Plant Transcriptomes. *PLoS ONE*. **7** (11)

(2012).
39. Zhao, S., Zhang, Y., Gamini, R., Zhang, B., Von Schack, D. Evaluation of two main RNA-seq approaches for gene quantification in clinical RNA sequencing: PolyA+ selection versus rRNA depletion. *Scientific Reports*. **8** (1), 1–12 (2018).
40. Zhao, S. et al. Comparison of stranded and non-stranded RNA-seq transcriptome profiling and investigation of gene overlap. *BMC Genomics*. **16** (1), 1–14 (2015).
41. Corley, S.M., MacKenzie, K.L., Beverdam, A., Roddam, L.F., Wilkins, M.R. Differentially expressed genes from RNA-Seq and functional enrichment results are affected by the choice of single-end versus paired-end reads and stranded versus non-stranded protocols. *BMC Genomics*. **18** (1), 1–13 (2017).
42. Haas, B.J. et al. De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis. *Nature Protocols*. **8** (8), 1494–1512 (2013).
43. Pertea, M. et al. StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nature biotechnology*. **33** (3), 290–295 (2015).
44. Bray, N.L., Pimentel, H., Melsted, P., Pachter, L. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology*. **34** (5), 525–527 (2016).
45. Patro, R., Duggal, G., Love, M.I., Irizarry, R.A., Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nature Methods*. **14** (4), 417–419 (2017).
46. Araujo, F.A., Barh, D., Silva, A., Guimarães, L., Thiago, R. OPEN GO FEAT : a rapid web-based functional annotation tool for genomic and transcriptomic data. 8–11 (2018).
47. Huerta-Cepas, J. et al. Fast genome-wide functional annotation through orthology assignment by eggNOG-mapper. *Molecular Biology and Evolution*. **34** (8), 2115–2122 (2017).
48. Huerta-Cepas, J. et al. EggNOG 5.0: A hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. *Nucleic Acids Research*. **47** (D1), D309–D314 (2019).
49. Törönen, P., Medlar, A., Holm, L. PANNZER2: A rapid functional annotation web server. *Nucleic Acids Research*. **46** (W1), W84–W88 (2018).
50. Robinson, M., Mccarthy, D., Chen, Y., Smyth, G.K. edgeR : differential expression analysis of digital gene expression data User ' s Guide. (March) (2013).
51. Huang, D.W., Sherman, B.T., Lempicki, R.A. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature Protocols*. **4** (1), 44–57 (2009).
52. Huang, D.W., Sherman, B.T., Lempicki, R.A. Bioinformatics enrichment tools: Paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Research*. **37** (1), 1–13 (2009).
53. Letunic, I., Bork, P. Interactive tree of life (iTOL) v3: an online tool for the display and annotation of phylogenetic and other trees. *Nucleic acids research*. **44** (W1), W242–W245 (2016).

Figure 1

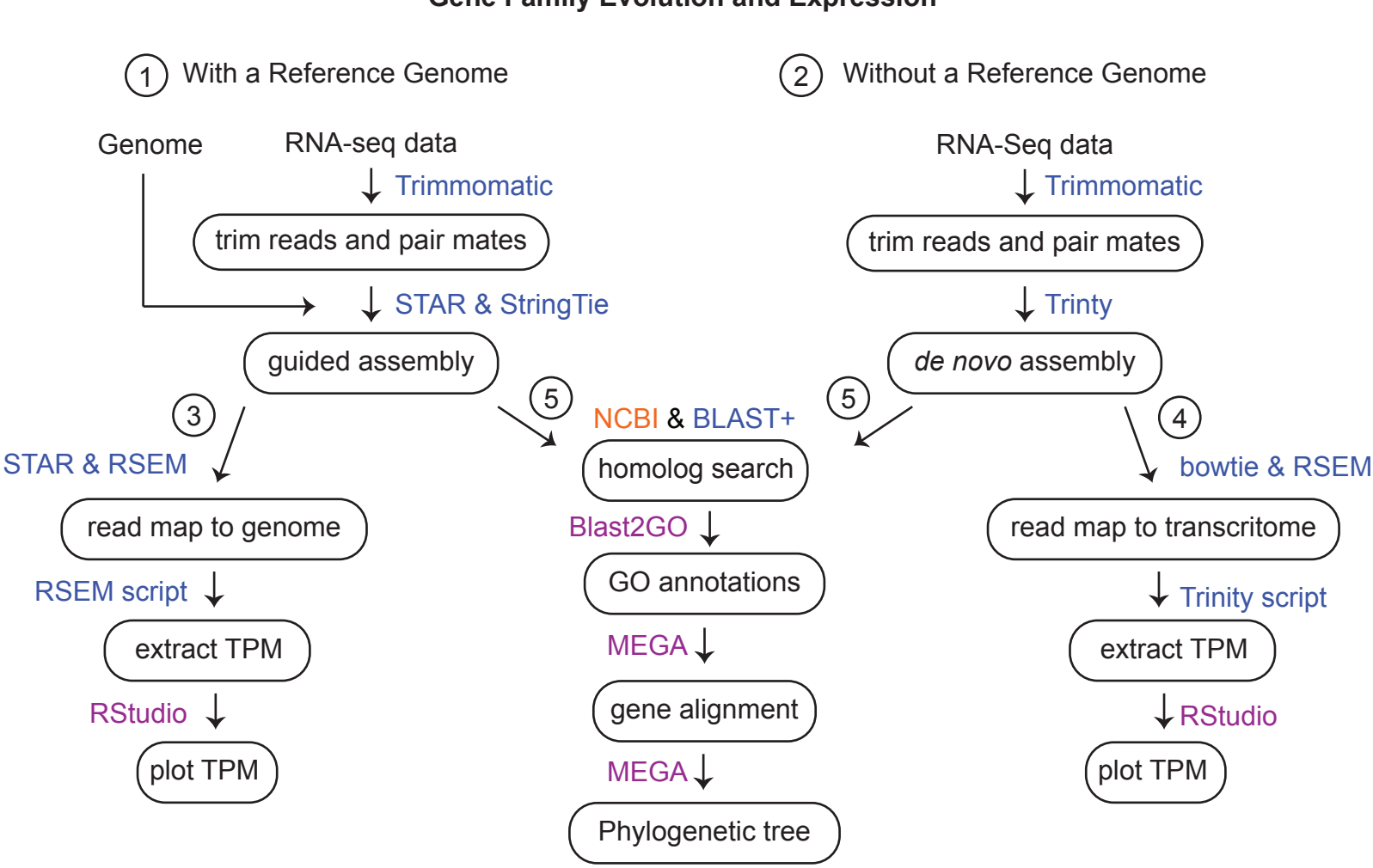

Figure 2

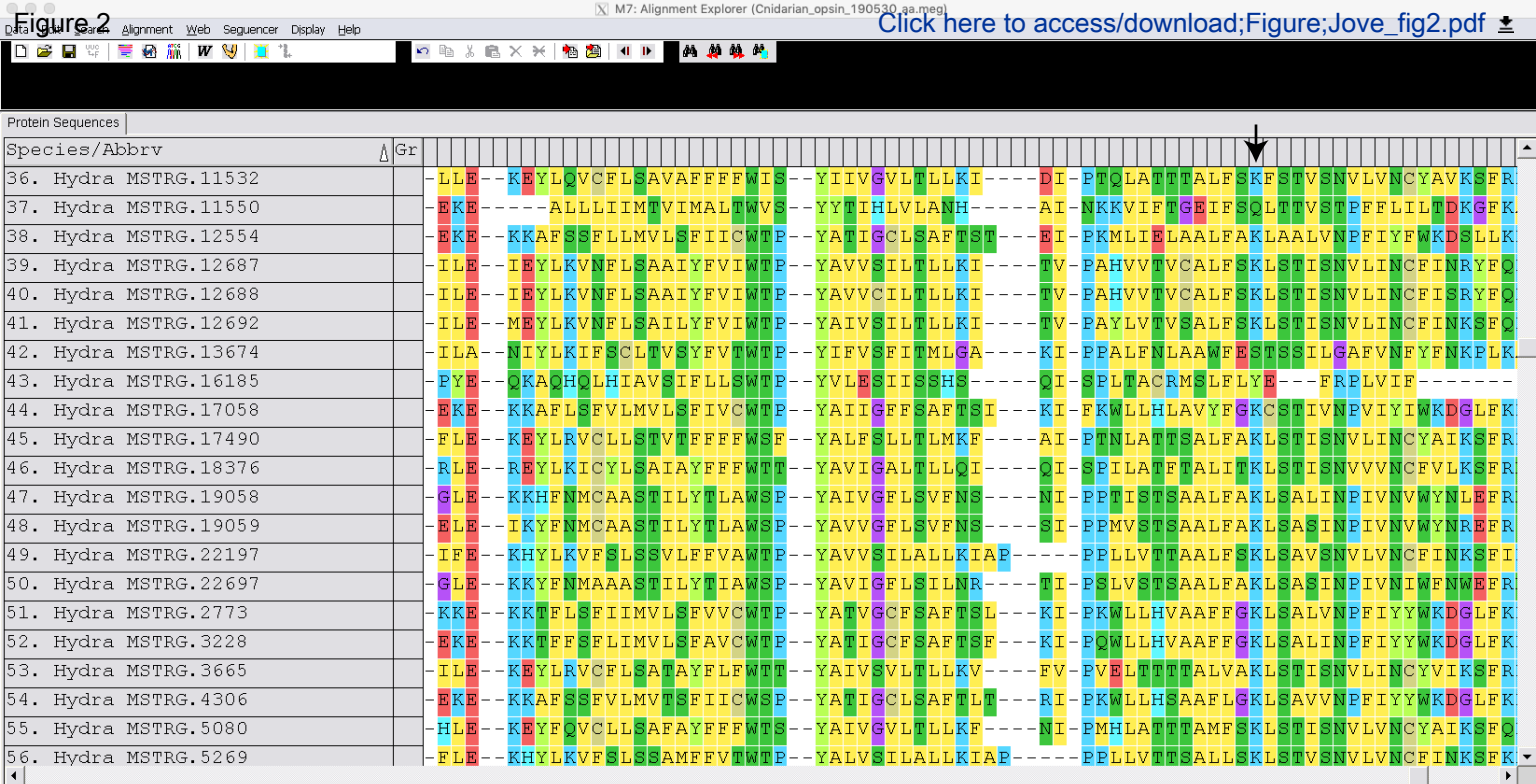

Figure 3

Click here to access/download;Figure;Jove_fig3.pdf

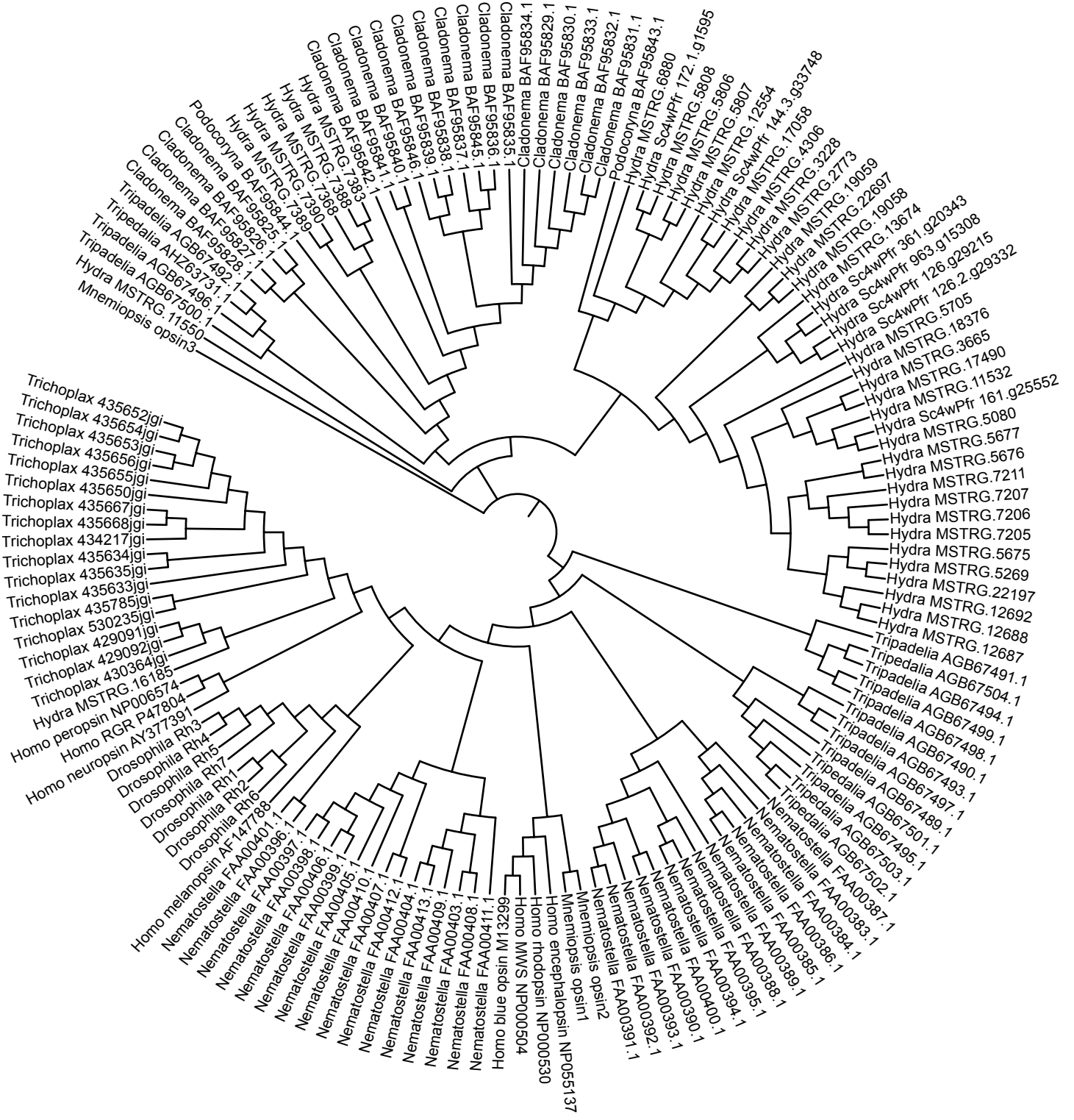

Figure 3

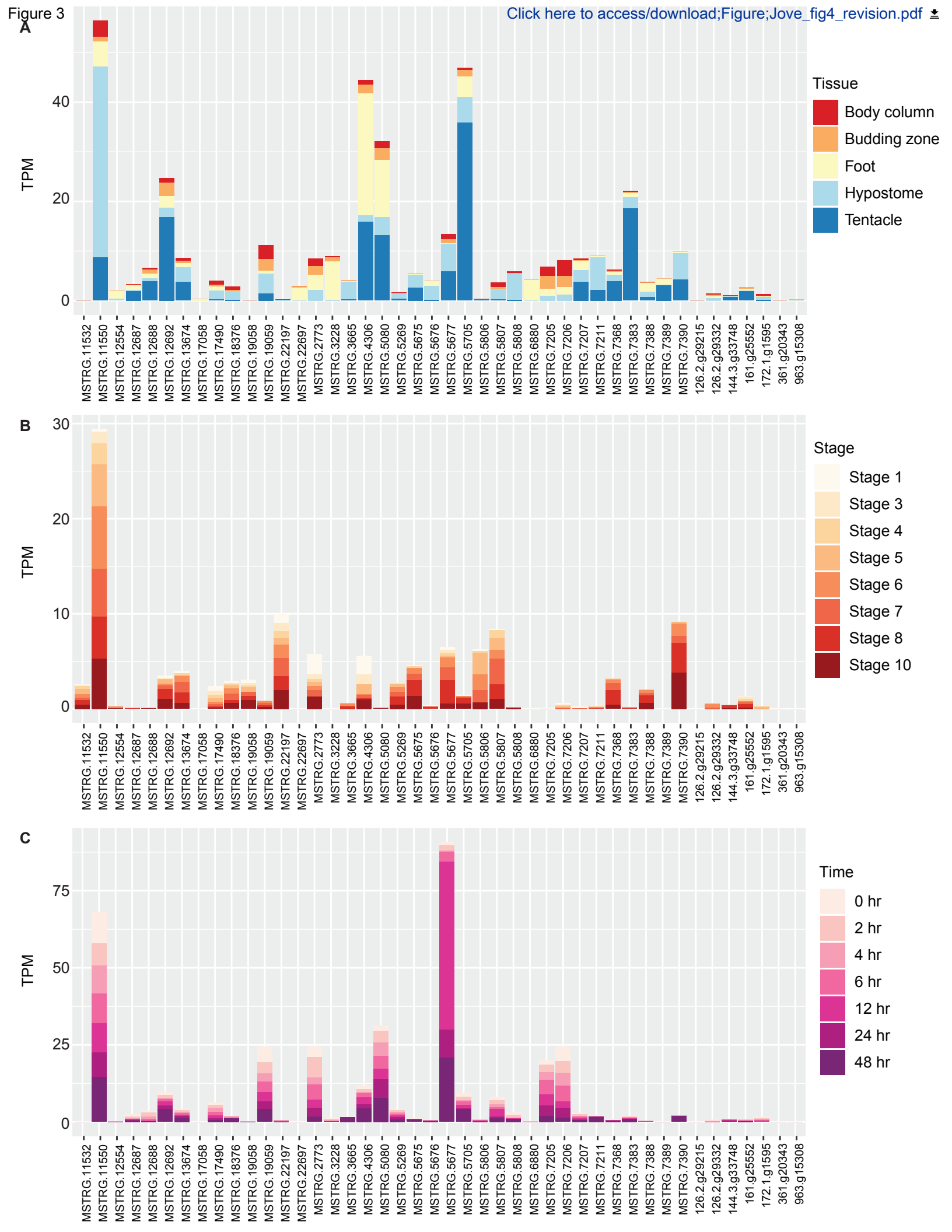

Table of Materials

Click here to access/download;Table of Materials;JoVE_Materials_AMM_revision.xls

Name of Material/ Equipment	Company	Catalog Number	Comments/Description	Link to latest release
Bioanalyzer-DNA kit	Agilent	5067-4626	wet lab materials	
Bioanalyzer-RNA kit	Agilent	5067-1513	wet lab materials	
BLAST+ v. 2.8.1			On computer cluster*	https://ftp.ncbi.nlm.nih.gov/blast
Blast2GO (on your PC)			On local computer	https://www.blast2go.com/b2g-r
boost v. 1.57.0			On computer cluster	
Bowtie v. 1.0.0			On computer cluster	https://sourceforge.net/projects/
Computing cluster (highly recommended)			NOTE: Analyses of genomic data are best done on a high-p	
Cufflinks v. 2.2.1			On computer cluster	
edgeR v. 3.26.8 (in R)			In Rstudio	https://bioconductor.org/packag
gcc v. 6.4.0			On computer cluster	
Java v. 11.0.2			On computer cluster	
MEGA7 (on your PC)			On local computer	https://www.megasoftware.ne
MEGAX v. 0.1			On local computer	https://www.megasoftware.ne
NucleoSpin RNA II kit	Macherey-Nagel	740955.5	wet lab materials	
perl 5.30.3			On computer cluster	
python			On computer cluster	
Qubit 2.0 Fluorometer	ThermoFisher	Q32866	wet lab materials	
R v.4.0.0			On computer cluster	https://cran.r-project.org/src/
RNAlater	ThermoFisher	AM7021	wet lab materials	
RNeasy kit	Qiagen	74104	wet lab materials	
RSEM v. 1.3.0			Computer software	https://deweylab.github.io/RS
RStudio v. 1.2.1335			On local computer	https://rstudio.com/products/
Samtools v. 1.3			Computer software	
SRA Toolkit v. 2.8.1			On computer cluster	https://github.com/ncbi/sra-to
STAR v. 2.6.0c			On computer cluster	https://github.com/alexdobin/
StringTie v. 1.3.4d			On computer cluster	https://ccb.jhu.edu/software/s
Transdecoder v. 5.5.0			On computer cluster	https://github.com/TransDeco
Trimmomatic v. 0.35			On computer cluster	http://www.usadellab.org/cms
Trinity v.2.8.5			On computer cluster	https://github.com/trinityrnas
TRIzol	ThermoFisher	15596018	wet lab materials	
TruSeq RNA Library Prep Kit v2	Illumina	RS-122-2001	wet lab materials	

TURBO DNA-free Kit	ThermoFisher	AM1907	wet lab materials

*Downloads and installation on the computer cluster may require root access. Contact your network administrator.

t/executables/blast+/LATEST/
egister-basic

/bowtie-bio/files/bowtie/1.3.0/
performance computing cluster because files are very large.

es/release/bioc/html/edgeR.html

et
et

base/R-4/

EM/
rstudio/download/#download

ools/wiki/01.-Downloading-SRA-Toolkit
/STAR
stringtie/
der/TransDecoder/releases
s/?page=trimmomatic
eq/trinityrnaseq/releases

UNIVERSITY OF CALIFORNIA, IRVINE

BERKELEY • DAVIS • IRVINE • LOS ANGELES • MERCED • RIVERSIDE • SAN DIEGO • SAN FRANCISCO SANTA BARBARA • SANTA CRUZ

Developmental and Cell Biology
School of Biological Sciences

Ali Mortazavi, Ph.D
Professor

2218 Biological Sciences 3
Irvine, CA 92697-2300
Phone: (949) 824-6762
Fax: (949) 824-4709
E-mail: ali.mortazavi@uci.edu

September 7, 2020

Journal of Visualized Experiments
Biology Section
Cambridge, MA

Dear Editors,

Please consider our revised manuscript entitled " Bioinformatics pipeline for investigating molecular evolution and gene expression using RNA-seq " for possible publication as a video protocol in JoVE. In this manuscript we outline bioinformatic methods for investigating the molecular evolution and expression of candidate genes. We made most of the revisions recommended by the editor and reviewers and believe our manuscript is much improved and easy to follow. We now provide more details per step and the protocol reads more like instructions to the reader.

Our methods will be of interest to readers investigating genomics of non-model organisms. We are confident that someone with minimal bioinformatics experience could follow this protocol to investigate desired genes in their favorite species.

Thank you for considering this manuscript. We look forward to hearing from you.

Sincerely,

Ali Mortazavi, Ph.D.
Professor of Developmental and Cell Biology
University of California, Irvine

Editorial comments:
NOTE: Please read this entire email before making edits to your manuscript. Please include a line-by-line response to each of the editorial and reviewer comments in the form of a letter along with the resubmission.

• Please take this opportunity to thoroughly proofread the manuscript to ensure that there are no spelling or grammatical errors.
We are grateful to the editor and the reviewers for thorough suggestions on improving the manuscript and making our methods more accessible. We have reviewed the manuscript to correct any spelling or grammatical errors.
• The title is too broad. Please focus it on the protocol.
We changed the title to: Bioinformatics pipeline for investigating molecular evolution and gene expression using RNA-seq
• Protocol Language: Please ensure that all text in the protocol section is written in the imperative voice/tense as if you are telling someone how to do the technique (i.e. "Do this", "Measure that" etc.) Any text that cannot be written in the imperative tense may be added as a "Note", however, notes should be used sparingly and actions should be described in the imperative tense wherever possible.
We made these changes throughout the manuscript and hope that it now reads as instructions with enough detail.
1) Line 90-109: Remove the list. List items in the table of materials.
List removed.
2) 3.1.1., 3.1.2: obtain from where?
We list a few online sites where readers can find genomes or assemblies: ensemblegenomes, NCBI genomes and Nucleotide (TSA).
3) 3.2.1: which database?
NCBI
4) 3.2.2: how is this done?
We fixed this by including detailed instructions.
5) 5.1.1: A reference is not sufficient if you wish to film this.
We give detailed instructions in the manuscript now.

• Protocol Detail: Please note that your protocol will be used to generate the script for the video, and must contain everything that you would like shown in the video. Please add more specific details (e.g. button clicks for software actions, numerical values for settings, etc) to your protocol steps. There should be enough detail in each step to supplement the actions seen in the video so that viewers can easily replicate the protocol.
• Please ensure that all specific details (e.g. button clicks for software actions, numerical values for settings, etc) have been added to your protocol steps. There should be enough detail in each step to supplement the actions seen in the video so that viewers can easily replicate the protocol.
We added all of the necessary details to each step.

1) Please include an ethics statement before your numbered protocol steps indicating that the protocol follows the animal care guidelines of your institution.
Done
2) 1.1.: Cite references for dissection and tissue extraction. Provide examples of organisms and mention those you have tested.
Done
3) 1.1.2: How is the tissue processed before RNA extraction? Cite a reference.
Flash frozen or place in RNAlater, a reference was also added.
4) 1.1.3, 1.2.2, 1.2.3 : Needs references.
Done

• Protocol Numbering:
1) Please adjust the numbering of your protocol section to follow JoVE's instructions for authors, 1. should be followed by 1.1. and then 1.1.1. if necessary and all steps should be lined up at the left margin with no indentations.
2) Add a one-line space between each protocol step.
We ensured that everything was numbered correctly.

• Protocol Highlight: After you have made all of the recommended changes to your protocol (listed above), please re-evaluate the length of your protocol section. There is a 10-page limit for the protocol text, and a 3- page limit for filmable content. If your protocol is longer than 3 pages, please highlight ~2.5 pages or less of text (which includes headings and spaces) in yellow, to identify which steps should be visualized to tell the most cohesive story of your protocol steps.

Done

1) The highlighting must include all relevant details that are required to perform the step. For example, if step 2.5 is highlighted for filming and the details of how to perform the step are given in steps 2.5.1 and 2.5.2, then the sub-steps where the details are provided must be included in the highlighting.
2) The highlighted steps should form a cohesive narrative, that is, there must be a logical flow from one highlighted step to the next.
3) Please highlight complete sentences (not parts of sentences). Include sub-headings and spaces when calculating the final highlighted length.
4) Notes cannot be filmed and should be excluded from highlighting.

• Discussion: JoVE articles are focused on the methods and the protocol, thus the discussion should be similarly focused. Please ensure that the discussion covers the following in detail and in paragraph form (3-6 paragraphs): 1) modifications and troubleshooting, 2) limitations of the technique, 3) significance with respect to existing methods, 4) future applications and 5) critical steps within the protocol.

We rewrote the discussion to cover all of the points listed above.

• Figures:
1) Please remove the embedded figures and tables from the manuscript. Figure legends, however, should remain within the manuscript text, directly below the Representative Results text. Upload all tables as excel files.

We deleted all figures and tables from the manuscript.

• Commercial Language: JoVE is unable to publish manuscripts containing commercial sounding language, including trademark or registered trademark symbols (TM/R) and the mention of company brand names before an instrument or reagent. Examples of commercial sounding language in your manuscript are TRIzol, NucleoSpin, Qubit, Bioanalyzer, e TruSeq v2, etc.
1) Please use MS Word's find function (Ctrl+F), to locate and replace all commercial sounding language in your manuscript with generic names that are not company-specific. All commercial products should be sufficiently referenced in the table of materials/reagents. You may use the generic term followed by "(see table of materials)" to draw the readers' attention to specific commercial names.

We removed the commercial names and reference the table of materials.

2) Please remove the registered trademark symbols TM/R from the table of reagents/materials.

Done

• Table of Materials: Sort the list alphabetically.

Done

• If your figures and tables are original and not published previously or you have already obtained figure permissions, please ignore this comment. If you are re-using figures from a previous publication, you must obtain explicit permission to re-use the figure from the previous publisher (this can be in the form of a letter from an editor or a link to the editorial policies that allows you to re-publish the figure). Please upload the text of the re-print permission (may be copied and pasted from an email/website) as a Word document to the Editorial Manager site in the "Supplemental files (as requested by JoVE)" section. Please also cite the figure appropriately in the figure legend, i.e. "This figure has been modified from [citation]."

The figures in this manuscript have not been previously published.

_____

Reviewers' comments:

Reviewer #1:

Manuscript Summary:

In the manuscript "Investigating Gene Family Evolution and Expression" the authors outline the different steps to identify and analyze a specific gene family group within a non-model organism. In today's data driven bioinformatic research, it is important to know the different analysis and tools that can be applied to a wide range of datasets (genomes and gene expression). The authors have combined in a well step by step pipeline all the necessary analysis, with the latest bioinformatic software, from gene expression RNA-seq data, identification of a specific gene family (opsins) to their evolutionary relationship.

The manuscript use technical descriptors well formulated. Authors provide good examples of code that can be used for every program mentioned. The text is accompanied with figures and tables well formatted that help the reader with the different analysis performed.

Major Concerns:

The authors show an approach for identifying genes of interest, using a combination of blast and GO annotation analysis (both on Blast2GO software). Unfortunately, I see big drawbacks within the step: If the main goal is identification of a specific gene family (opsins) within a new species, my first take would be an homology search (blast) using the specific gene family from closer species (orthologs sequences) as query. And not only that; it should be followed by a reciprocal blast search in order to minimize potential false positive in your final gene pool. Then, it can be used for proper alignment and phylogeny analysis. I understand that the authors don't stop at that, using the outcome from the entire transcriptome annotation analysis for further inputs (differential expression, functional enrichment). But that initial set up (reciprocal blast) would be very beneficial, not only in computational time, where you can get an fair and quick estimate of potential targets in you new-to-you genome/transcriptome (without having to annotate the entire genome), and very effective in orthologous sequence search. I understand the authors claim a similar step at paragraph 4.2, which is after the GO annotations step. But my opinion is that getting genes of interest by merely using GO annotations and key words in their description, may lead to inaccuracies, which will depend on the family of proteins you are working with.

We agree with the reviewer and edited our methods for identifying genes of interest (now section 8). We now provide detailed instructions on how to extract homologs, BLAST to the genome or transcriptome, and do a reciprocal BLAST.

Since we would like to have a functional annotation in your genome, for future differential expression and functional enrichment analysis, the other big drawback I found is the software mentioned at this step (Blas2GO). Although the authors have come in hand with free and open-source software along the manuscript, they get tempted by the dark side at this step. Unless it is running the Pro account license, which will cost money or maybe your institution has one, the basic account can take a lot of time in order to get a whole genome analysis done (I have been there). The authors already mention an alternative, Trinotate, which goes handy if you are using Trinity scripts in your analysis (which are in the manuscript). Or maybe it can be mentioned an intermediate approach; since blast is the real limiting factor in Blast2GO basic, why don't combined an standard blast within the high capacity cluster (which most of the institutions have) and use the output for GO annotation with Blast2GO. I remember doing something similar, but it was on previous versions, not sure how it would work know.

Yes, the basic version of Blast2GO can be very slow. In the discussion we now mention other alternatives that we tested. One of which worked very quickly (web-based PANZZER2).

Minor Concerns:

Although I think descriptors are good enough along the text, it's also true that I am quite familiar with most of the pipeline described in the protocol: programs used and specific parameters. But I think it would be more powerful for the potential reader a brief description of whats being done in each step. I am not talking about the parameters used (for

that you can read the manuals), but further descrptions in some steps would be great for clarification on what is being done. For example, Line 158 and Line 164. Something like "This file type need to be generated in order to run the next step which will involve this and this to get...". In Line 278 "RSEM quantification" can have some explanatory text.
We had added some detail to the text about input and output but removed them because it was extra detail that did not read like a list of instructions.

During the text sometimes I am confused whether it is referring to genome assembly (reference genome), gene models, transcriptome assembly (guided, de novo), etc. Like I said in the paragraph above, descriptors should be more elaborated. Eg. L145 "genome gtf file" what is in there? L170 "StringTie assembly" that's a trascript guide assembly, right?
We provide more detail now in the supplemental results. We did not include them in the main text for the reasons stated above.

In point 5. Check and verify gene sequence (Line 228), the authors propose a visual inspection for missing exons and blasting. An alternative could be to use the candidate genes and use public databases like Pfam and Conserved Domain Database (CDD) to look for those truncations in their sequence.
We appreciate the reviewer's suggestions and insight. We tired Pfam and were able to view a matching protein sequence that can be used to blast the genome and recover the true sequence. We now recommend this in the supplement.

When a study points out a phylogenetic analysis, there should be accompanied with the alignment (fasta format) of the sequences studied. In this case, it would be a great opportunity to interact with the pipeline outlined and visualized the results.
We now refer to figure 2 in this section so that the reader can see what an alignment of a FASTA file looks like.

Reviewer #2:
Manuscript Summary:
The manuscript described different computational protocols that are essential to perform a genome-wide study of gene families.

Major Concerns:
Major concern is its Novelty.
While the methods are not new, we believe they are a good resource for biologist inexperienced in bioinformatics and seeking to investigate a handful of genes. Our protocol outlines commonly used software that can give insight into evolution and function with easily attainable data.

Minor Concerns:
Nil

Reviewer #3:
Manuscript Summary:
The manuscript details a protocol to study the evolution in sequence and expression of a protein family of interest. The manuscript reads well, and describes the protocol with sufficient detail. Although other choices of software and tools are possible, the one chosen by the authors is reasonable and justified.

Major Concerns:
- Nowadays there are better options to share a computational protocol other than listing the programs to install and the commands. I recommend the authors to consider Docker or any other container option where they could share the very same environment used in their analysis, this will ensure reproducibility and multiply the impact of their protocol.
In order to make our pipeline easier to follow, we included additional supplementary figures, scripts, and R code for edgeR.

- The search for "homologs" in Genebank can be more tricky than described, given the common presence of duplicated sequences and misannotations, I would recommend also searching for homologs of the genes of interest in pre-computed databases such as EggNOG, MetaPhOrs or OMA. Just leaving NCBI search for the few specialized species that may not yet be in those databases.
We agree with the reviewer and for that reason we recommend doing a more thorough annotation of the genes. In the manuscript we give instructions for Blast2GO which searches across different databases but we also list a few alternatives that we tested in the discussion such as the above mentioned eggNOG.

Minor Concerns:
- Some versions are not indicated, for instance for python or R.
We added versions to all programs and we also provided links for download in the table of materials.

Reviewer #4:
Summary
In this manuscript, Macias-Muñoz and Mortazavi outline a computational approach to initial characterization of a gene family by using long RNA-sequencing data. Authors then show how this approach can be used to characterize opsin genes in Hydra vulgaris. Manuscript is written in a clear fashion, although re-arranging and slight expansion of the Introduction section, as well as fixing some formatting issues might be necessary. Below, I outline how, in my opinion, the manuscript can be improved. I do not have major concerns; all my suggestions are for minor edits that do not regard the authors' approach. Because the goal of JoVE is to provide enough information for the researchers that are learning a new technique, I also included some more specific suggestions about which details could be added to the Protocol section. I believe that such details would help readers and viewers reproduce the reported representative results.

General suggestions
1. In my opinion, the Introduction section needs re-structuring and a bit of expanding. I suggest that it is done as follows. I would start with a summary about the main current approaches to study evolution of gene families, and then - of how gene expression can be used to help such studies. I suggest that authors provide examples of how gene expression was successfully used to characterize gene families not only in non-model, but also - in model organisms. I would continue with listing the most important challenges to the field - theoretical considerations and practical difficulties. I would suggest finishing the Introduction with a very brief, one or two sentence summary of the cited work on Hydra vulgaris (ref. 13), and with the description of the pipeline reported in this work and how this pipeline addresses previously mentioned challenges.
We rewrote the introduction to address these points.

2. Authors provide a protocol for a very particular type of library. I think it would be helpful to provide at least some indications, how different library preparation protocols (e.g., single-end vs. paired-end, with or without rRNA depletion, preserving or not the strand information, libraries with different insert size) would affect the analysis of the data (i.e., what command line parameters users would need to pay attention to).
We address these points in the discussion.

3. I would suggest that authors provide time estimates for each of the steps. Because the time spent for each step largely depends on the hardware used and on the size of datasets, estimates can be given for the system that authors used to produce their Representative Results.
We added time estimates to the representative results as well as we could recall.

4. I believe that in the Representative Results section, it would be appropriate not only list what was done, but also provide readers with at least one paragraph with interpretation of the data obtained and conclusions derived from them.
We now include a concluding sentence of the main results from that project.

5. Please consider reporting command lines with variables instead of specific file names and paths, as well as instead of specific parameters that would change based on, for example, size of the insert (in STAR genome index preparation, --sjdbOverhang 42). For example genome_file.fa would become $GENOME_NAME, and index_name would become

$GENOME_INDEX, etc. All used variables can be listed in the beginning of the protocol along with example values. I think this would make it a little easier and convenient for users to adjust the pipeline for their particular needs.
We changed the command to include variables as is often seen in instruction manuals.

6. Throughout the manuscript, where it is not a part of the file name extension, I recommend that capital letters are used for referencing GTF, BAM, FASTA and FATSQ files. Also, please make sure that all files are referenced in the notes to commands consistently (e.g., READ1.fastq file is currently referred to in the text as "READ1", "read1" and "Read1" fastq file). The same suggestion applies to the names of the tools used (e.g., "Blast2GO" and "BLAST2GO").
Thank you for pointing that out. We reviewed the manuscript to make sure everything was consistent.

Specific suggestions by line / by section
Note that on lines 27-28, "publicly available" is used twice - please re-phrase.
Done
1.1.: Please add a reference to the table of materials.
Done
1.1.2.: Authors mention that there could be kits that work better for different species or tissue types. If possible, please share any information in that regard or provide relevant references to the published data for interested readers. Also, indicate how presence or absence of DNase treatment step could affect the downstream results.
Done
1.1.3.: Please indicate the target range of yields and of the Bioanalyzer RNA Integrity Numbers for the obtained total RNA.
> 8 but ideally around 9.
1.2.1.: I think it would be a good idea to list at least one alternative kit for normal and for small input library preparation, and to note that depending on the kit used for library preparation, adapter trimming procedure in the analysis stage can be affected. I also suggest that authors briefly comment on whether rRNA depletion is required, as well as on whether building a strand-specific library is important.
We address this in the discussion.

1.2.3.: Comment on how critical for the task it is or it is not to build a paired-end library, as opposed to a single-end library. Also, make a note on how using single-end RNAseq library would affect the usage of the tools, as the example commands are given for the paired-end libraries only. What is the minimum depth of sequencing that authors recommend to obtain?
We address this in the discussion.

2.: For computing cluster, mention the minimum hardware setup that is required for successful execution of the protocol. Make a note on the operating system requirements. For other tools, please add an indication whether or not root access is required for the installation of the program - this can be a helpful piece of information when the users of the protocol do not have root access to the computing cluster. Also, do authors think that it would be helpful to provide links to the main pages of the projects where download and installation instructions can be found?
We added links to where programs can be downloaded in the Table of Materials. We also mention which program installations may require root access.

3.1.1., 3.1.2.: It would be useful for the readers, especially those who are new to the field, to have a list of databases, where up-to-date reference genome sequences (at least - for the most commonly used organisms) can be obtained.
We listed EnsemblGenomes and a google search.

3.2.1.: Add that the commands used here are a part of SRA Toolkit. Does the command produce the output named as READ1.fastq and READ2.fastq or is it in the format SRRXXXXXX_1.fastq and SRRXXXXXX_2.fastq? In the latter case, for compatibility with the next commands, it might make sense to change the indicated file names accordingly.
Done

3.2.2., 3.3.2., 3.4.2.: Note that the output of the upstream command (3.2.1.) is uncompressed fastq files, and the input for Trimmomatic in the command line provided is compressed fastq files. Accordingly, the output of the command is

compressed fastq files, and the indicated input for mapping reads with STAR is uncompressed fastq files. On line 136, please replace "fastqs" with "fastq files", and the use of "mated" and "paired" in "mated or paired" seems redundant.
We made the recommended changes and replaced mater with paired. In our edits we no longer included zipped files to make it easier for the reader to follow.

3.3.1.: Authors do not mention where the GTF file is obtained from, and what to do if only genome fasta file is available, but not the GTF annotation.
We address this in the discussion and supplemental material.

3.3.2. I suggest that Input is denoted as "a pair of mated FASTQ files per sample and location of STAR genome index", and Output - as "one sorted BAM file per sample".
We changed this (now in the supplemental).

3.3.3.: Is the genome.gtf file used for StringTie the same as gtf_file.gtf used in the step 3.3.1.?

3.3.4.: I suggest that on line 167, "StringTie gtf files for each sample" is better replaced with "gtf files generated by StringTie for all samples".
We made this change.

3.4.1.: Please separate cat command by placing it on a new line. Authors recommend to include the smallest possible number of samples per experiment - which ones would they recommend choosing - the most deeply sequenced, moderately deeply sequenced or the least deeply sequenced?
We removed this recommendation as it was confusing.
3.4.2.: Is "out" the name of the default output directory for this command line? The sentence on line 186 is phrased in somewhat confusing way, please re-phrase.
We now include $OUT to let the reader know it is variable what they decide to name their output.

4.1.: Indicate which version of Blast2GO was used, and what Trinotate version authors tested as an alternative.
Done.

4.1.1.: Where are GO annotations obtained from? Do they come together with Blast2GO or they should be downloaded separately? In the latter case, please make sure that video and manuscript feature instructions about how to download the annotations.
This is now in the manuscript.

4.2.1.: "genome or transcriptome fasta" can be phrased as "genome or transcriptome FASTA file represented by genome.fa file (or replace with a variable, as in General Suggestion #5)"
Line 225, "ids" should be "IDs"
We replaces all instances of fasta to FASTA and ids to IDs.

5.1.2.: It is not clear, what manually corrected index is - is it represented by pseudogene.fa file? I would use a different name (or a variable) for that file to not confuse readers. Also, I think it would be helpful to provide an example for this step on a figure, or even include it into the video.
We include a supplemental figure now and more details in the supplement.

6.2., 6.3.: In the Discussion section, please comment on the parameters selected and what other parameter alternatives can be used at these steps.
We mention this in supplemental materials.

7.4.: What do authors mean by "To visualize individual genes of interest, calculate and normalize TPM from counts matrix"? Please re-phrase or elaborate.
Do authors suggest using normalized read counts in TPM or raw counts as an input for EdgeR?

Line 373: Authors did not indicate in the protocol, how the graph was made with ggplot.
We include this at the end of the protocol.

Figure 4: Labels on the left of are illegible. What is the purpose for presenting Figure 4? I do not think that heat map adds any information to the presented results, and can be removed.
We removed figure 4.

Figure 5: In the legend, please indicate which datasets were used as a source for each panel.
Lines 451-452: "qPCR, RT-PCR" can be replaced with just "RT-qPCR" to avoid redundancy in the text.
Done.

Custom genome GTF using StringTie[1]

If a genome is missing genes of interest, a transcript guided assembly can be generated. Build gene models using STAR and StringTie as outlined below:

1. Index genome using STAR[2] v. 2.6.0c bby typing: **STAR --runThreadN 16 --runMode genomeGenerate --genomeDir index_name --sjdbGTFfile $GENOME.GTF --sjdbOverhang $42 --genomeFastaFiles $GENOME.FASTA**
Input a genome fasta file containing nucleotide sequences and genome gtf file containing coordinates for exons and coding sequences.

2. Map reads to genome using STAR by typing **STAR --runThreadN 16 --genomeDir $index_name --outFileNamePrefix $SAMPLE --outSAMtype BAM SortedByCoordinate --sjdbScore 1 --readFilesIn $paired_READ1.FASTQ paired_READ2.FASTQ**
Input a pair of mated FASTQ files per sample and location of STAR genome index. The output should be one sorted BAM file per sample.

3. Generate a StringTie[1] gtf file using: **stringtie sample_id.sortedByCoord.out.bam -G $GENOME.GTF -o $SAMPLE.GTF -p 12 -A stringtie_id.abundance.txt.**
Input: bam file (output from STAR containing mapped reads) and the genome gtf.
Output: gtf and abundance table for each sample.

4. Merge gtf files using StringTie.
stringtie <$SAMPLE.GTF(s)> --merge -G $GENOME.GTF -o $OUT.GTF -A merged_abundance.txt.
Input: genome gtf and gtf files generated by StringTie for all samples.
Output: merged gtf of the genome and all sample gtfs.

5. Extract fasta sequence for the merged StringTie assembly using cufflinks[3] v. 2.2.1.
gffread -w $OUT.FASTA -g $GENOME.FASTA $MERGE.GTF.
Input: merged gtf and genome fasta file.
Output: fasta file with StringTie denoted transcripts.

To obtains a GTF from a transcriptome or genome GFF3 file use cufflinks[3] v. 2.2.1.
gffread $IN.GFF3 -T -o $OUT.GTF
For a transcriptome, a GFF3 can be produced by running transdecoder as explained in the main text.

Recovering missing gene fragments from a genome

In some genomes the genes of interest may not be well annotated in terms of exon start and stop and may be missing gene beginning, middle or end. In such instances, the complete sequence can be recovered by searching the genome using homologous sequences.

1. Open MEGA and import the sequences of interest.
2. Align using MUSCLE as listed in the main text.
3. Visually inspect sequences for large missing fragments.

4. Manually complete any missing parts of the gene using orthologs or paralogs. An alternative is to search Pfam and view the alignment of the matching sequence.
5. Confirm the correct complete gene sequence by using tblastn against the reference genome.

⇒ **tblastn -db $GENOME_BLAST_DATABASE -query $MANUALLY_COMPLETE_SEQ -evalue 1e-10 -out $ALIGNMENT.OUT**

Input: genome database name and manually corrected index.
Output: alignment showing matches and mismatches for the manual correction.
NOTE: Check the BLAST output (FIG S3) to determine the actual sequence, which may be different from the correction done using orthologs or paralogs. If there is no match, leave the sequence as was originally. When checking output pay attention to the genome coordinates to make sure the missing exon is indeed part of the gene.

Cross sample normalization and differential expression analysis

TPMs can be normalized across samples using R v. 4.0.0 and a package called NOISeq[4, 5].
https://www.bioconductor.org/packages/release/bioc/html/NOISeq.html

Comparisons across samples to identify differentially expressed genes can be done in R using edgeR[6]. A sample R script is below comparing Hydra hypostome (hypo) to the body column (bc).

```
Hydra_reads<-read.table("Hydra_tpm.txt",header = T,row.names = 1)
    head(Hydra_reads)
    Hydra_reads<-round(Hydra_reads)
    keep<-rowSums(cpm(Hydra_reads)>1) >=2
    Hydra_reads<-Hydra_reads[keep,]
    head(Hydra_reads)
    tissue<-factor(c("hypo","hypo","bc", "bc"))
    data.frame(sample=colnames(Hydra_reads),tissue)
    design<-model.matrix(~tissue)
    rownames(design) <- colnames(Hydra_reads)
    list<-DGEList(counts=Hydra_reads,group=tissue)
    list<-calcNormFactors(list)
    plotMDS(list)
    list<-estimateGLMCommonDisp(list,design)
    list<-estimateGLMTagwiseDisp(list,design)
    fit<-glmFit(list,design)
    LRT<-glmLRT(fit,coef=2)
    DEgenes<-LRT$table
    head(DEgenes)
    FDR <- p.adjust(LRT$table$PValue, method="BH")
    DE_hydra_tissue<-cbind(DEgenes,FDR=FDR)
```

Figure S1.

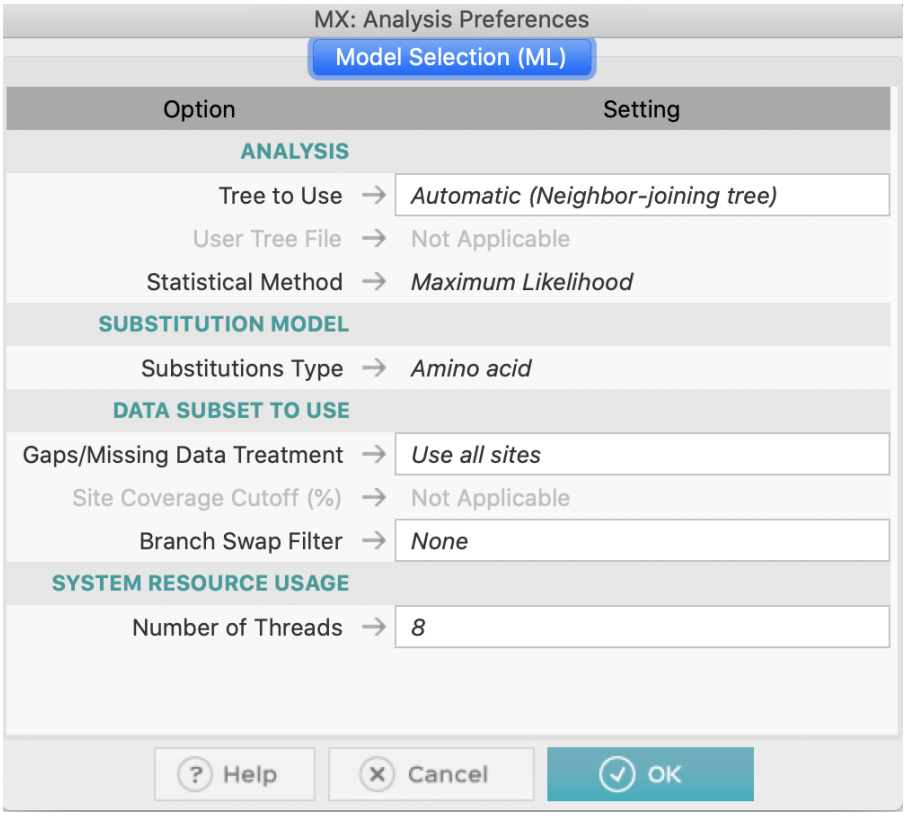

Figure S1. MEGAX Model Selection. The figure shows the interface for the MEGAX model selection tool. The figure shows the parameters and that we chose but Tree to Use, Gaps/Missing data treatment, Branch Swap Filter and Number of Threads can be changed.

Figure S2.

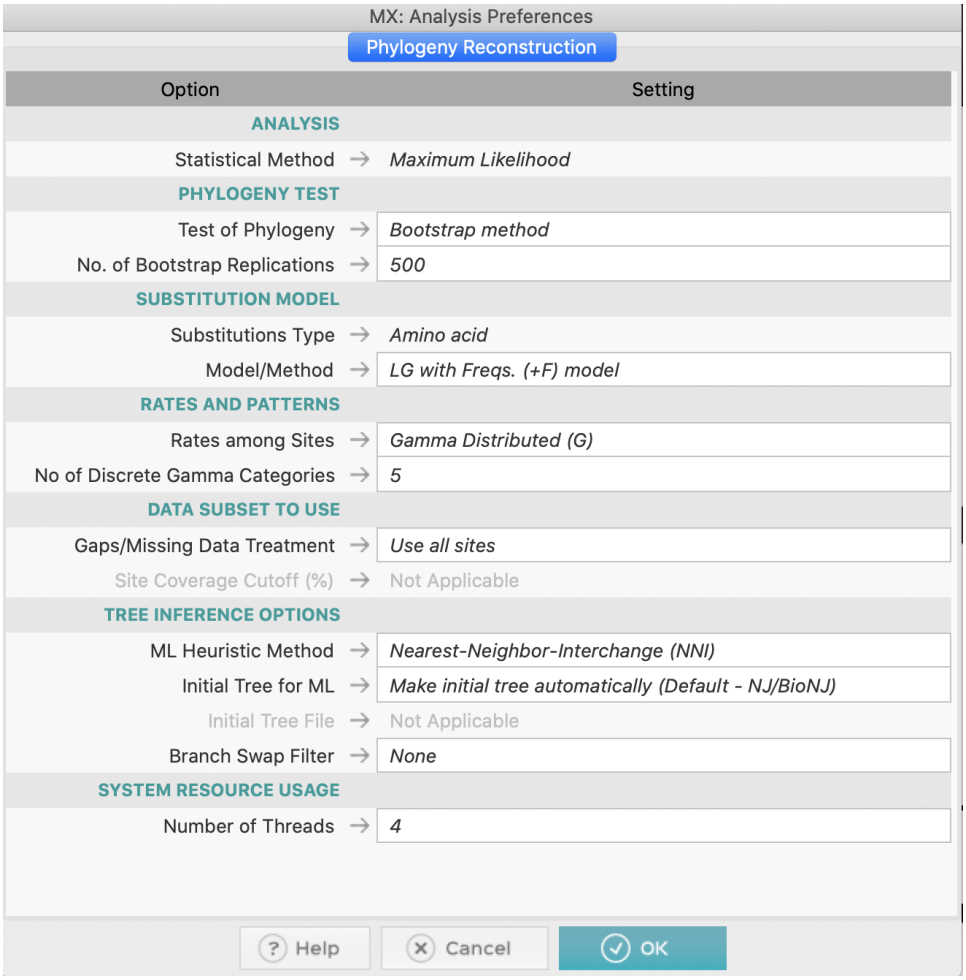

Figure S2. MEGAX Phylogeny Reconstruction. The figure shows the interface for the MEGAX phylogeny reconstruction tool. This figure has the parameters that we chose using the output of the previous step to set the substitution model and the rates among site.

Figure S3.

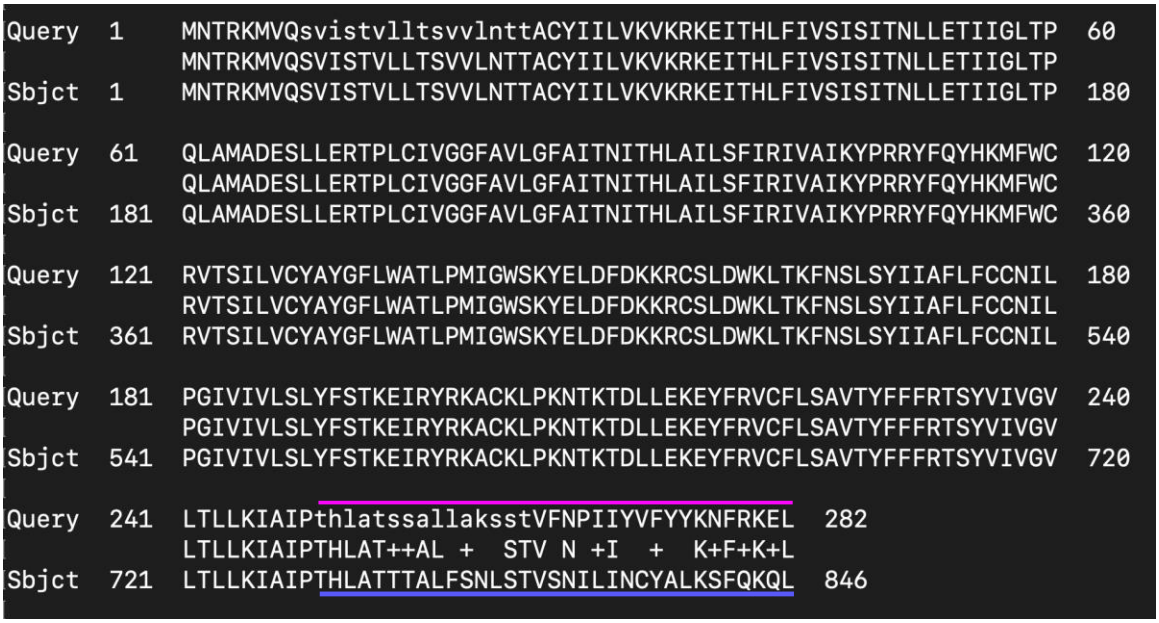

Figure S3. Partial gene recovery. Below the magenta line is the sequence from a closely related species that was manually added in MEGA. Above the blue line is the true sequence recovered from the genome. The in complete sequence can be corrected and used to generate a phylogeny and can also be used to repeat read-mapping if necessary.

References:

1. Pertea, M., Pertea, G.M., Antonescu, C.M., Chang, T.-C., Mendell, J.T., Salzberg, S.L. StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nature biotechnology*. **33** (3), 290–295, doi: 10.1038/nbt.3122 (2015).

2. Dobin, A. *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*. **29**, 15–21, doi: 10.1093/bioinformatics/bts635 (2013).

3. Trapnell, C. *et al.* Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology*. **28** (5), 511–515, doi: 10.1038/nbt.1621 (2010).

4. Tarazona, S., García-Alcalde, F., Dopazo, J., Alberto, F., Conesa, A. Differential expression in RNA-seq: A matter of depth. *Genome Research*. 2213–2223, doi: 10.1101/gr.124321.111.Freely (2011).

5. Tarazona, S. *et al.* Data quality aware analysis of differential expression in RNA-seq with NOISeq R/Bioc package. *Nucleic Acids Research*. **43** (21), doi: 10.1093/nar/gkv711 (2015).

6. Robinson, M.D., Mccarthy, D.J., Smyth, G.K. edgeR : a Bioconductor package for differential expression analysis of digital gene expression data. **26** (1), 139–140, doi: 10.1093/bioinformatics/btp616 (2010).