

# Obesity Induced Changes in Gene Expression

This project will focus on analyzing simulated RNAseq data in order to evaluate differential gene expression between control and treatment samples. Your goal is to combine your knowledge of Unix, Python/R, and bioinformatic tools *muscle* and *hmm* to “replicate” a recent study of gene expression in kidney tissues of normal (control) and obese mice (Kuhns & Pluznick 2017). RNAseq involves high-throughput sequencing of cDNA libraries constructed from all expressed RNA in the target tissue. Sequences are filtered and aligned to a transcriptome or genome assembly with established bioinformatic pipelines, including tools such as *Bowtie* or *Tophat*. Expression levels of each transcript in each tissue sample are then quantified/converted to counts using statistical models implemented in programs such as *HTSeq* and differential expression assessed with various programs/packages (e.g. *edgeR* or *DESeq*). In this project, you will approximate this process by building protein models of 6 target transcripts, searching simulated transcriptome data for “hits,” and counting those hits as a proxy for gene expression.

Begin by inspecting your files, you should have six to work with:

1. *uniquetranscripts.fasta*: A fasta file including the sequences of 6 transcripts that showed differential expression between kidney tissues of normal (control) and obese mice in Kuhns & Pluznick 2017.
2. *codonmap.txt*: A tab delimited text file with amino acid abbreviation in the first column and associated three base pair codons in the second.
- 3&4. *control1.fasta* & *control2.fasta*: Transcript sequences produced in a “new *RNAseq* experiment” with kidney tissue from two normal (control) male mice. These sequences have been translated into amino acid sequences.
- 5&6. *obese1.fasta* & *obese2.fasta*: Transcript sequences produced in a “new *RNAseq* experiment” with kidney tissue from two diet-induced obese male mice. These sequences have been translated into amino acid sequences.

**Use BLAST to identify the genes encoded by the 6 differentially expressed transcripts listed in *uniquetranscripts.fasta*.**

Go to <https://blast.ncbi.nlm.nih.gov/Blast.cgi> and select *nucleotide BLAST*, because your data are transcriptome (cDNA) sequences. Upload *uniquetranscripts.fasta* via the *choose file* button shown below.

The screenshot shows the NCBI BLAST Standard Nucleotide BLAST interface. At the top, it says "BLAST® >> blastn suite". Below that, it says "Standard Nucleotide BLAST". There are tabs for "blastn", "blastp", "blastx", "tblastn", and "tblastx", with "blastn" selected. A blue bar says "BLASTn programs search nucleotide databases using a nucleotide". Below this is a section titled "Enter Query Sequence". It has a text input field for "Enter accession number(s), gi(s), or FASTA sequence(s)" with a "Clear" button. To the right is a "Query subrange" section with "From" and "To" input fields. Below the text input field is a section for "Or, upload file" with a "Choose File" button and a "No file chosen" status. Below that is a "Job Title" input field with a placeholder "Enter a descriptive title for your BLAST search". At the bottom left, there is a checkbox labeled "Align two or more sequences".

For this assignment, search the *Nucleotide collection (nr/nt)* Database.

Choose Search Set

Database

☐ Human genomic + transcript
☐ Mouse genomic + transcript
☒ Others (nr etc.):

Organism

Optional

Exclude

Optional

☐ Models (XM/XP)
☐ Uncultured/environmental sample sequences

Limit to

Optional

☐ Sequences from type material

Entrez Query

Optional

[Create custom database](#)

Also, use the default optimization for *Highly similar sequences (megablast)*. And click *BLAST* to initiate the search.

Program Selection

Optimize for

☒ Highly similar sequences (megablast)
☐ More dissimilar sequences (discontiguous megablast)
☐ Somewhat similar sequences (blastn)

Search database **Nucleotide collection (nr/nt)** using **Megablast (Optimize for highly similar sequences)**
☐ Show results in a new window

[Algorithm parameters](#)

When the searches are complete, your browser will redirect to the results page. The drop down menu next to *Results for:* will allow you to toggle between results for the 6 transcripts in your fasta file.

BLAST Results

[Edit and Resubmit](#)
[Save Search Strategies](#)
[Formatting options](#)
[Download](#)

Job title: Transcript\_1 (854 letters)

Results for:

☒ 1:cl|Query\_225146 Transcript\_1(854bp)
☐ 2:cl|Query\_225147 Transcript\_2(1752bp)
☐ 3:cl|Query\_225148 Transcript\_3(3753bp)
☐ 4:cl|Query\_225149 Transcript\_5(1878bp)
☐ 5:cl|Query\_225150 Transcript\_6(2989bp)
☐ 6:cl|Query\_225151 Transcript\_7(1052bp)
☐ 7:cl|Query\_225152 Transcript\_8(3954bp)
☐ 8:cl|Query\_225153 Transcript\_9(2294bp)
☐ 9:cl|Query\_225154 Transcript\_10(2550bp)
☐ 10:cl|Query\_225155 Transcript\_11(1496bp)

RID

Query ID

Description

Molecule type

Query Length

Database Name

Description

Program

Other reports: [Access tree of results](#)

Scroll down to the *Descriptions* heading for your table of hits, which you should **save to your repo**. The best hits, listed first, will have extremely low E-values and high (~100%) Identity. By selecting *All* (top left of the table) the *Download* button becomes available and you can choose the download format from the dropdown menu. **Using Unix commands or R/Python, make a single table that includes the top hit for each transcript.**

Descriptions

Sequences producing significant alignments:

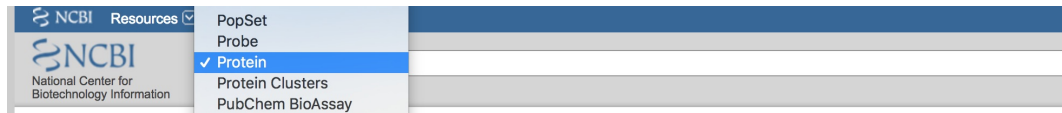
Select: All None Selected: 100

☐ FASTA (complete sequence)
☐ FASTA (aligned sequences)
☐ GenBank (complete sequence)
☐ Hit Table (text)
☒ Hit Table (CSV)
☐ Text
☐ XML
☐ ASN.1

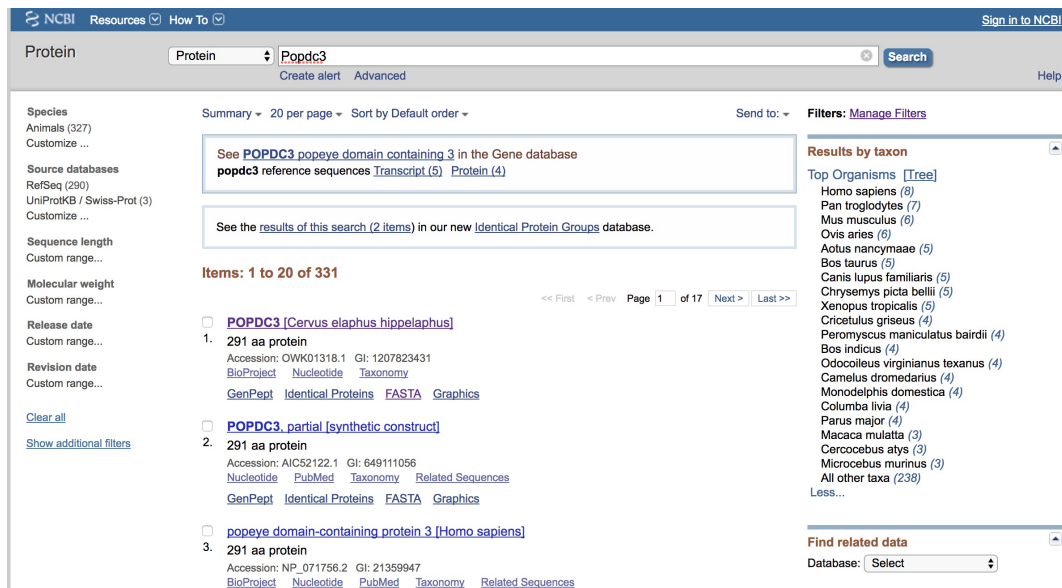
Description	Max score	Total score	Query cover	E value	Ident	Accession
ferase, alpha 2 (Yc2) (Gata2), mRNA	1578	1578	100%	0.0	100%	NM_008182.3
cDNA, RIKEN full-length enriched library, clone:081003006 product:glutathione S-transferase, alpha 2 (Yc2), full insert sequence	1557	1557	100%	0.0	99%	AK002712.1
T41a.mRNA, 3' end	1517	1517	97%	0.0	99%	J03858.1
639 (Gm10839), mRNA	1493	1493	100%	0.0	98%	NM_001122680.1
thione S-transferase, alpha 2 (Yc2) (Gata2), transcript variant X1, mRNA	1478	1582	100%	0.0	100%	XM_006510814.2
ferase, alpha 2 (Yc2), mRNA (cDNA clone MGC:74263 IMAGE:30303519), complete cds	1471	1471	93%	0.0	100%	BC081133.1
ferase, alpha 1 (Yq) (Gata1), mRNA	1450	1450	100%	0.0	97%	NM_008181.3
78 (Gm3776), mRNA	1445	1445	100%	0.0	97%	NM_001243092.1
Mus musculus glutathione S-transferase, alpha 1 (Yq), mRNA (cDNA clone MGC:74264 IMAGE:30303041), complete cds	1404	1404	97%	0.0	97%	BC081134.1
Mus musculus predicted gene, ENSMUSG00000074179, mRNA (cDNA clone MGC:29061 IMAGE:5041881), complete cds	1399	1399	93%	0.0	98%	BC030173.1
Mus musculus glutathione S-transferase, alpha 1 (Yq), mRNA (cDNA clone MGC:184207 IMAGE:40130853), complete cds	1317	1317	90%	0.0	98%	BC132576.1

Search the NCBI protein database for amino acid sequences corresponding to these 6 transcripts.

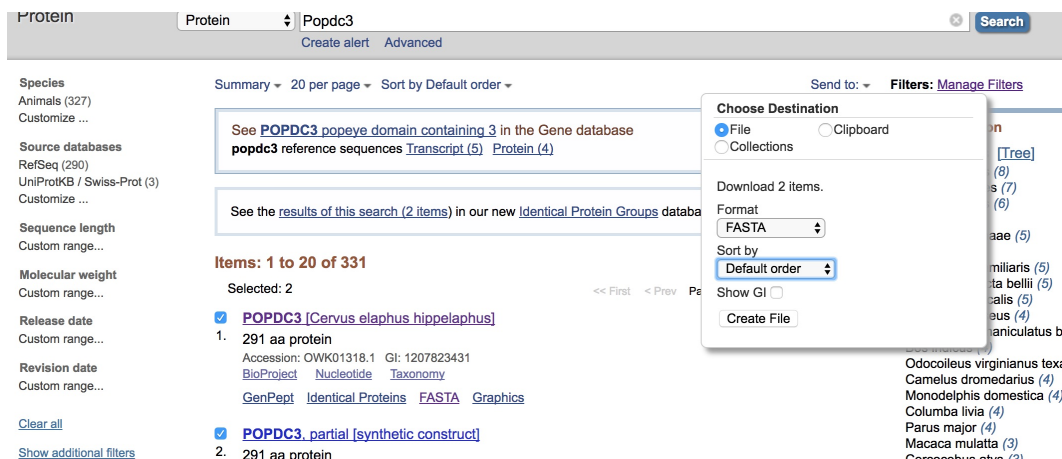
Point your browser to the NCBI homepage <https://www.ncbi.nlm.nih.gov/> and choose the *Protein* database.



Based on the name of the protein listed in the top hit for each transcript in the previous step, search the NCBI protein data base for sequences. By selecting *Animals* under *Species* at the top left of the page, the option to filter by taxon should appear at the top right.



Depending on how conserved a protein is, the protein sequence can vary substantially between distantly related organisms (view taxa as a *TREE* to display relationships). This could change the efficacy of the HMM model you will build below. Choose 10-20 protein sequences from mice (*Mus musculus*) and other closely related organisms (if available) by checking the boxes to the left of the listed matches. Download your selected sequences by clicking on the arrow next to *Send to:*, in the upper right corner, selecting *File* and choosing *FASTA* from the dropdown menu. **You should save one fasta file of protein sequences per identified transcript (6 total).**



## Build a Hidden Markov Model for each of the 6 transcript proteins and search the 4 translated “RNAseq files.”

For each of the 6 transcripts make a muscle alignment from your downloaded protein sequences. Using these alignments, construct 6 HMM protein models using `hmmbuild`. Finally, search all 4 translated “RNAseq files” for each of the 6 HMM protein models using `hmmsearch`. **Use a bash script to loop over the 6 transcript files and 4 “RNAseq files,” executing these commands.**

## Graph the “expression levels” of each protein in each of the “RNAseq files.”

Based on the counts of hmm hits for each transcript (our measure of RNA expression) in each “RNAseq file,” make a **graphical comparison of expression levels** across the 2 normal and 2 obese mice. **Qualitatively compare these results to those reported in Kuhns & Pluznick 2017.**

## Further Exploration

1. For 2 of your 6 transcripts, return to the original *BLAST* search and change the *Optimize for* option. Quantitatively, how do *discontinuous megablast* and *blastn* change your table of *BLAST* hits? This may be easier to explore, if you also restrict the *Database* option to either *Human* or *Mouse*. (You do not need to repeat any other steps here.)
2. For 2 of your 6 transcripts, return to the NCBI protein search and explore the effects of phylogenetic relatedness of your amino acid sequences on the performance of your HMM model. For example, what would happen if you built your HMM protein model using more distantly related mammals (e.g. primates)? Would you still get the same quality hits if your HMM protein model was based on non-mammalian sequences? Pick one of the “RNAseq files” to search in order to test your hypotheses. Compare e-values among HMMs built from differing taxa.