Zebrafish Dataset

Before using the dataset, copy "deseq2-results.tsv" from "penelopeprime" to your home directory. Alternatively, you can download the file from:
https://funcgen2019.buschlab.org/downloads/deseq2-results.tsv

You should also copy a file called "samples.tsv" from "penelopeprime" to your home directory. This was used by DESeq2 (see below) and lists all the samples along with their corresponding DESeq2 conditions. You can also download the file from:
https://funcgen2019.buschlab.org/downloads/samples.tsv

This dataset comes from a current collaboration and it's not yet published so please **DO NOT** share the data outside of this course.

The dataset consists of four comparisons, each of which is between 12 homozygous zebrafish embryos and 12 of their heterozygous and wild-type siblings. The four comparisons vary according to the age of the embryos (either 3, 5 or 7 dpf) and as to whether or not they have been infected with *Mycobacterium marinum*. The mutation was identified in a genetic screen for loci affecting infection susceptibility.

Each of the 96 samples (4 x 12 vs 12) has a name like "inf_5dpf_hom_repX", where "inf" indicates the sample was infected (as opposed to "uninf"), "5dpf" indicates the embryo is 5 days post fertilisation, "hom" indicates the embryo is homozygous for the mutation (as opposed to "het" or "wt") and X is a number indicating the replicate.

The column headings are:

1	**GeneID**	Ensembl ID
2	**inf_5dpf_hom_vs_sib_pval**	p-value for homozygote vs sibling comparison in infected 5 dpf embryos
3	**inf_5dpf_hom_vs_sib_adjp**	Adjusted p-value for homozygote vs sibling comparison in infected 5 dpf embryos
4	**inf_5dpf_hom_vs_sib_log2fc**	Log_2 fold change for homozygote vs sibling comparison in infected 5 dpf embryos
5	**uninf_3dpf_hom_vs_sib_pval**	p-value for homozygote vs sibling comparison in uninfected 3 dpf embryos
6	**uninf_3dpf_hom_vs_sib_adjp**	Adjusted p-value for homozygote vs sibling comparison in uninfected 3 dpf embryos
7	**uninf_3dpf_hom_vs_sib_log2fc**	Log_2 fold change for homozygote vs sibling comparison in uninfected 3 dpf embryos
8	**uninf_5dpf_hom_vs_sib_pval**	p-value for homozygote vs sibling comparison in uninfected 5 dpf embryos
9	**uninf_5dpf_hom_vs_sib_adjp**	Adjusted p-value for homozygote vs sibling comparison in uninfected 5 dpf embryos
10	**uninf_5dpf_hom_vs_sib_log2fc**	Log_2 fold change for homozygote vs sibling comparison in uninfected 5 dpf embryos
11	**uninf_7dpf_hom_vs_sib_pval**	p-value for homozygote vs sibling comparison in uninfected 7 dpf embryos
12	**uninf_7dpf_hom_vs_sib_adjp**	Adjusted p-value for homozygote vs sibling comparison in uninfected 7 dpf embryos
13	**uninf_7dpf_hom_vs_sib_log2fc**	Log_2 fold change for homozygote vs sibling comparison in uninfected 7 dpf embryos
14	**Chr**	Chromosome (or scaffold) name
15	**Start**	Gene start (in bp)
16	**End**	Gene end (in bp)

17	**Strand**	Gene strand (1 or -1)
18	**Biotype**	Gene biotype (e.g. protein coding or lincRNA)
19	**Name**	Gene name
20	**Description**	Gene description
21	**inf_5dpf_wt_rep1_count**	Counts for 1st inf_5dpf_wt replicate
22	**inf_5dpf_wt_rep2_count**	Counts for 2nd inf_5dpf_wt replicate
…	**…**	…
116	**uninf_7dpf_hom_rep12_count**	Counts for 12th uninf_7dpf_hom replicate
117	**inf_5dpf_wt_rep1_normalised_count**	Normalised counts for 1st inf_5dpf_wt replicate
118	**inf_5dpf_wt_rep2_normalised_count**	Normalised counts for 2nd inf_5dpf_wt replicate
…	**…**	…
212	**uninf_7dpf_hom_rep12_normalised_count**	Normalised counts for 12th uninf_7dpf_hom replicate

For reference (and only for reference – none of this is necessary for this course), this dataset was generated using STAR and DESeq2 as follows:

1. The zebrafish GRCz11 genome and Ensembl 98 transcriptome were downloaded and unzipped using:

```
wget ftp://ftp.ensembl.org/pub/release-
98/fasta/danio_rerio/dna/Danio_rerio.GRCz11.dna_sm.primary_assembly.fa.gz
wget ftp://ftp.ensembl.org/pub/release-98/gtf/danio_rerio/Danio_rerio.GRCz11.98.gtf.gz
gunzip Danio_rerio.GRCz11.dna_sm.primary_assembly.fa.gz
gunzip Danio_rerio.GRCz11.98.gtf.gz
```

2. The genome was indexed using STAR:

```
mkdir grcz11 genome-generate
STAR \
--outFileNamePrefix genome-generate/ \
--runThreadN 4 \
--runMode genomeGenerate \
--genomeDir grcz11 \
--genomeFastaFiles Danio_rerio.GRCz11.dna_sm.primary_assembly.fa \
--sjdbGTFfile Danio_rerio.GRCz11.98.gtf \
--sjdbOverhang 74
```

3. For each sample ($sample below) a pair of FASTQ files were aligned to the genome using STAR:

```
mkdir -p star1/$sample
STAR \
--runThreadN 1 \
--genomeDir grcz11 \
--readFilesIn fastq/$sample.1.fastq.gz fastq/$sample.2.fastq.gz \
--readFilesCommand zcat \
--outFileNamePrefix star1/$sample/ \
--quantMode GeneCounts \
--outSAMtype BAM SortedByCoordinate
done
```

4. Each pair of FASTQ files were aligned to the genome for a second round using STAR:

```
mkdir -p star2/$sample
STAR \
--runThreadN 1 \
--genomeDir grcz11 \
--readFilesIn fastq/$sample.1.fastq.gz fastq/$sample.2.fastq.gz \
--readFilesCommand zcat \
--outFileNamePrefix star2/$sample/ \
--quantMode GeneCounts \
--outSAMtype BAM SortedByCoordinate \
--sjdbFileChrStartEnd `find star1 | grep SJ.out.tab$ | sort | tr '\n' ' '`
```

5. DESeq2 input files were made using:

```
wget https://raw.githubusercontent.com/iansealy/bio-misc/master/make_deseq_from_star.pl
cat /dev/null > counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 5dpf | grep -v uninf | grep wt >>
counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 5dpf | grep -v uninf | grep -v wt >>
counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 3dpf | grep wt >> counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 3dpf | grep -v wt >> counts
files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 5dpf | grep uninf | grep wt >>
counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 5dpf | grep uninf | grep -v wt >>
counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 7dpf | grep wt >> counts-files.txt
find star2 | grep ReadsPerGene.out.tab | sort -V | grep 7dpf | grep -v wt >> counts-files.txt
perl make_deseq_from_star.pl --count_files counts-files.txt
rm counts-files.txt
mkdir deseq2
mv samples.txt counts.txt deseq2
```

6. DESeq2 was run using:

```
wget https://raw.githubusercontent.com/iansealy/bio-misc/master/run_deseq2_rnaseq.pl
perl run_deseq2_rnaseq.pl \
--comparisons \
inf_5dpf_hom:inf_5dpf_wt,inf_5dpf_het=inf_5dpf_sib \
uninf_3dpf_hom:uninf_3dpf_wt,uninf_3dpf_het=uninf_3dpf_sib \
uninf_5dpf_hom:uninf_5dpf_wt,uninf_5dpf_het=uninf_5dpf_sib \
uninf_7dpf_hom:uninf_7dpf_wt,uninf_7dpf_het=uninf_7dpf_sib \
--remove_other_conditions
```

7. A file containing all Ensembl 98 zebrafish genes in TSV format was downloaded from BioMart and includes:
    - Gene stable ID
    - Chromosome/scaffold name
    - Gene start (bp)
    - Gene end (bp)
    - Strand
    - Gene type
    - Gene name
    - Gene description

   The file was called `annotation.txt` and empty fields were changed to "-" using:

```
perl -spi -e 's/\t\t/\t-\t/g' annotation.txt
perl -spi -e 's/\t$/\t-/g' annotation.txt
```

8. Results were merged into one file using:

```
echo -ne "GeneID\t" > deseq2_results.tsv
echo -ne "inf_5dpf_hom_vs_sib_pval\tinf_5dpf_hom_vs_sib_adjp\tinf_5dpf_hom_vs_sib_log2fc\t"
>> deseq2_results.tsv
echo -ne
"uninf_3dpf_hom_vs_sib_pval\tuninf_3dpf_hom_vs_sib_adjp\tuninf_3dpf_hom_vs_sib_log2fc\t" >>
deseq2_results.tsv
echo -ne
"uninf_5dpf_hom_vs_sib_pval\tuninf_5dpf_hom_vs_sib_adjp\tuninf_5dpf_hom_vs_sib_log2fc\t" >>
deseq2_results.tsv
echo -ne
"uninf_7dpf_hom_vs_sib_pval\tuninf_7dpf_hom_vs_sib_adjp\tuninf_7dpf_hom_vs_sib_log2fc\t" >>
deseq2_results.tsv
echo -ne "Chr\tStart\tEnd\tStrand\tBiotype\tName\tDescription" >> deseq2_results.tsv
for sample in `head -1 deseq2/counts.txt`
do
echo -ne "\t${sample}_count" >> deseq2_results.tsv
done
for sample in `head -1 deseq2/counts.txt`
do
echo -ne "\t${sample}_normalised_count" >> deseq2_results.tsv
done
echo >> deseq2_results.tsv
join -j1 -t $'\t' <(sort deseq2/inf_5dpf_hom_vs_inf_5dpf_sib/output.txt) \
<(sort deseq2/uninf_3dpf_hom_vs_uninf_3dpf_sib/output.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_5dpf_hom_vs_uninf_5dpf_sib/output.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_7dpf_hom_vs_uninf_7dpf_sib/output.txt) \
| join -j1 -t $'\t' - <(sort annotation.txt) \
| join -j1 -t $'\t' - <(sort deseq2/inf_5dpf_hom_vs_inf_5dpf_sib/counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_3dpf_hom_vs_uninf_3dpf_sib/counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_5dpf_hom_vs_uninf_5dpf_sib/counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_7dpf_hom_vs_uninf_7dpf_sib/counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/inf_5dpf_hom_vs_inf_5dpf_sib/normalised-counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_3dpf_hom_vs_uninf_3dpf_sib/normalised-counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_5dpf_hom_vs_uninf_5dpf_sib/normalised-counts.txt) \
| join -j1 -t $'\t' - <(sort deseq2/uninf_7dpf_hom_vs_uninf_7dpf_sib/normalised-counts.txt) \
>> deseq2_results.tsv
```