# Zebrafish-Specific 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>

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** | Log2 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** | Log2 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** | Log2 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** | Log2 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 as follow:

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

1. 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

1. 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

1. 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' ' '`

1. 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

1. 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

1. 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

1. 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