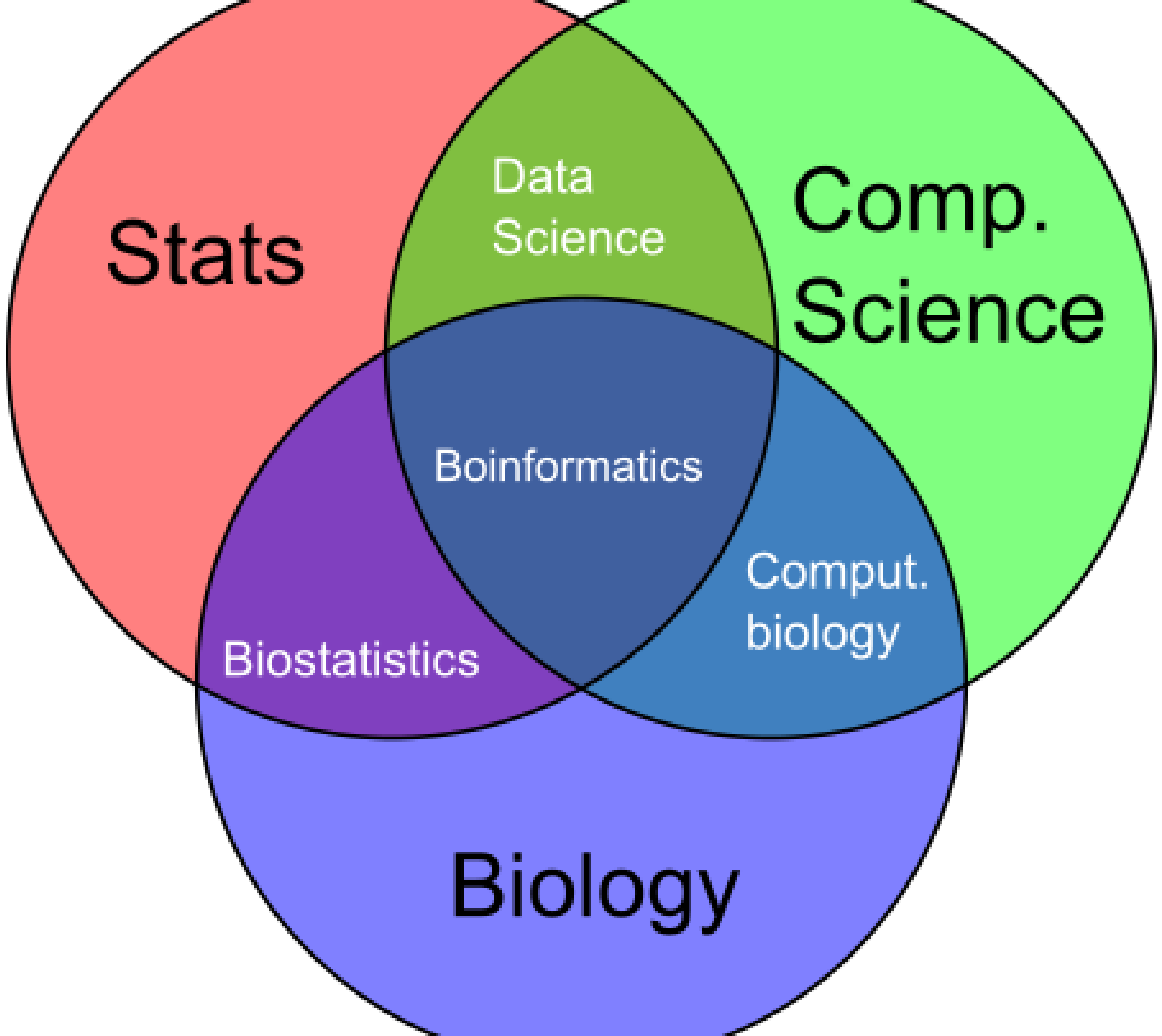
ERNA KINETICS FROM INFB1 INDUCTION

WHO AM I

- Plano, TX
- GitHub
- Undergraduate Molecular Biology '15-'18
- Masters Biotechnology '18-'19
- Joined Functional Genomics group in July '15
- Moved to Dry Lab Summer '17
- Software Developer for Olypsis Technologies since May '18

WHAT I AM LOOKING TO GET OUT OF THIS


Stats
Data Science
Comp. Science
Boinformatics
Biostatistics
Comput. biology
Biology

THINGS I CAN BRING TO THE TABLE

- Wet Lab
- Linux
- Emacs
- Git
- Devops & CI/CD
- Docker
- Snakemake

REPRODUCIBLE RESEARCH

CREATING A WORKFLOW

EMACS

- Evil Keybinding
- Magit
- Dired
- Tramp
- Org-mode
- Email
- RSS

SPACEMACS

- Slow
- Difficult to Customize outside of the box

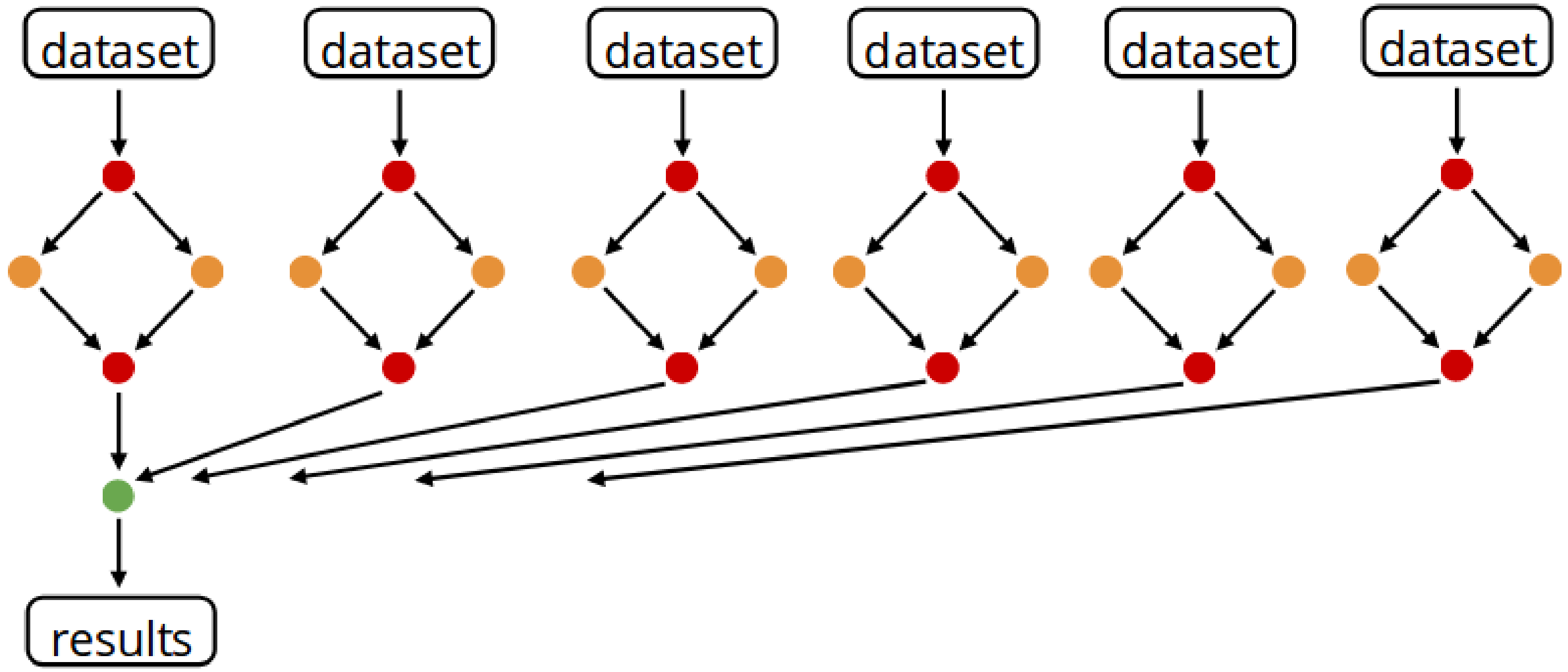
DOOM EMACS

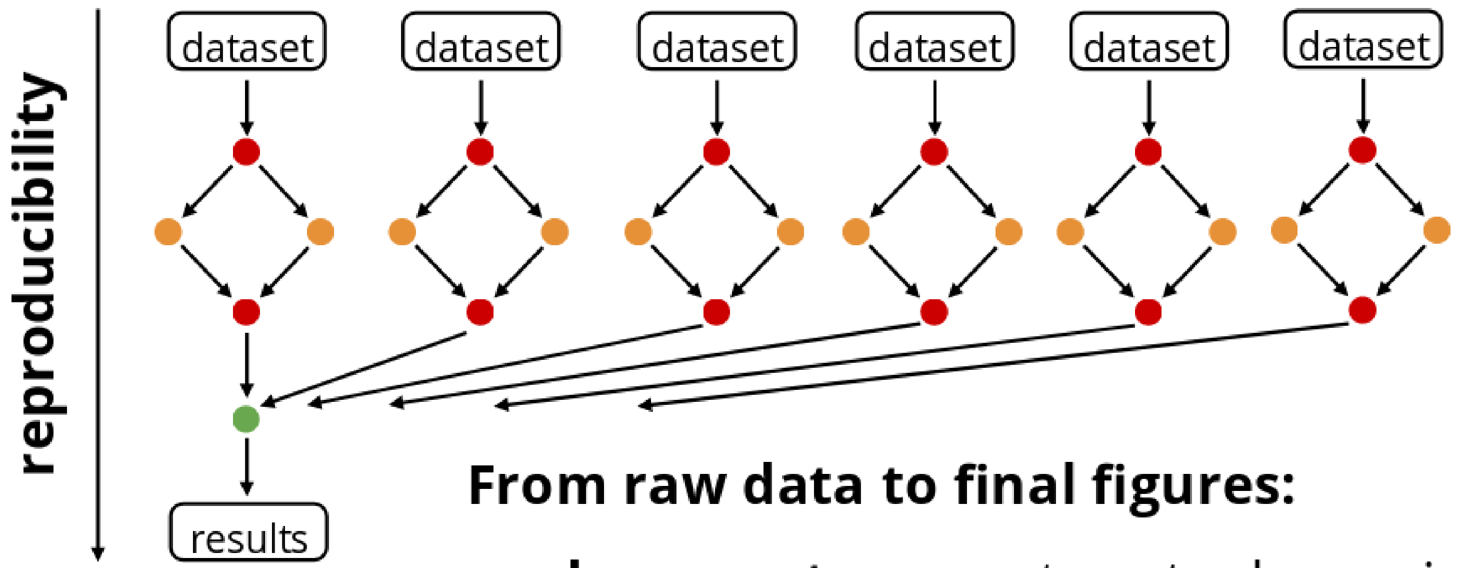
- Optimized
- Community

ORG-MODE

- Slides
- TODOs
- Agenda
- Capture
- Lab-Notebook
- Babel

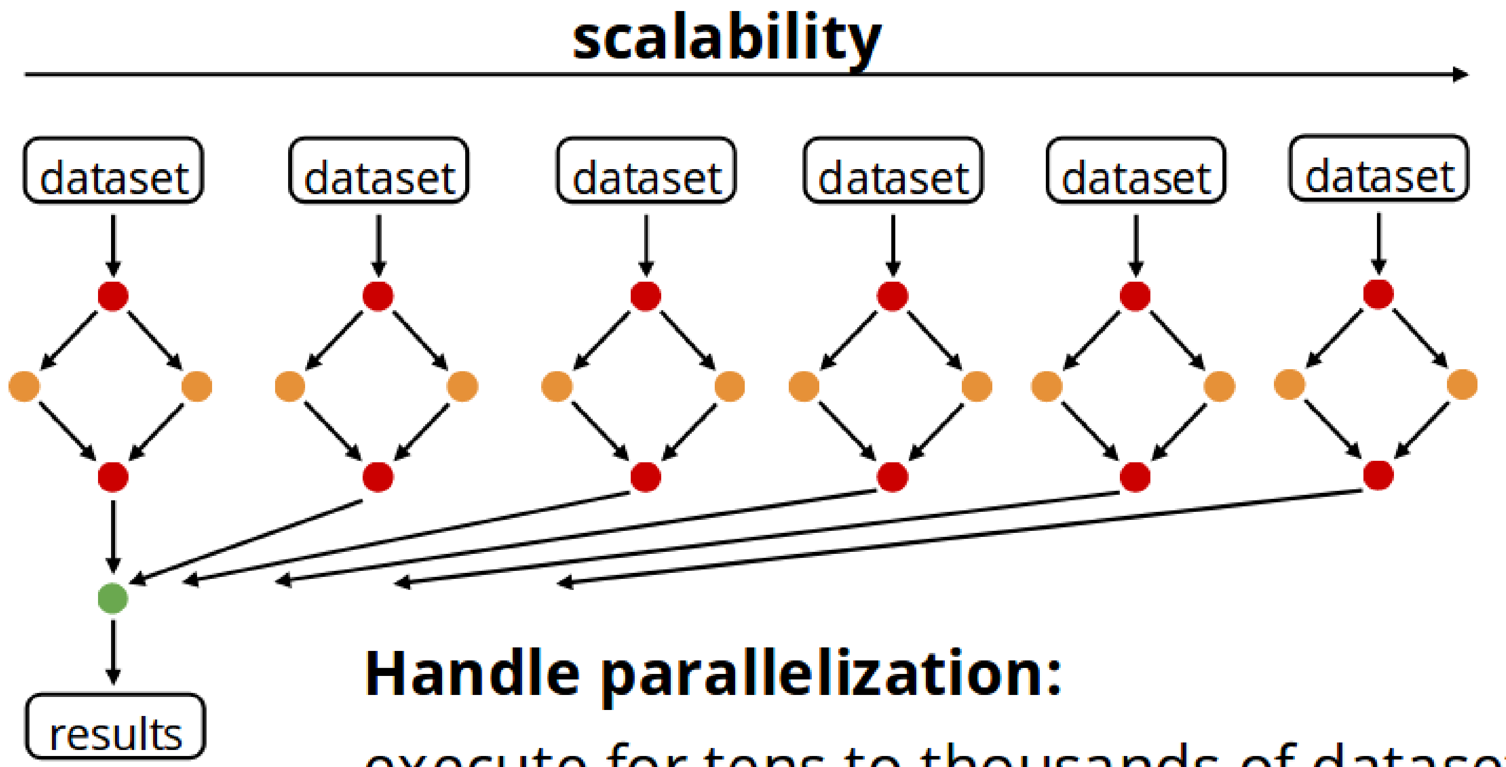
MAKEFILES


dataset
dataset
dataset
dataset
dataset
dataset
results



From raw data to final figures:

- **document** parameters, tools, versions
- **execute** without manual intervention



Handle parallelization:

execute for tens to thousands of datasets

Avoid redundancy:

- when adding datasets
- when resuming from failures

EXAMPLE MAKEFILE

```
all: prot001.txt prot0002.txt prot003.txt

%.txt: ./bin/intensive_operation %.fasta
   $^ > $@

%.fasta:
    curl database.example.com/$* > $@
```

BIOINFORMATICS PIPELINES

TOIL

- Python 2.7
- UCSC
- Complex

NEXTFLOW

- Java
- Barcelona Center for Genomic Regulation
- Push
- Able to script in any language

SNAKEMAKE

- Python 3.5
- Easy to containerize
- Pull
- Quick to use

rule name

```
rule sort:
    input:
        "path/to/dataset.txt"
    output:
        "dataset.sorted.txt"
    shell:
        "sort {input} > {output}"
```

how to create output from input

refer to input and output from shell command

SNAKEMAKE EXAMPLE

```
rule targets:
    input:
        "plots/dataset1.pdf",
        "plots/dataset2.pdf"

rule plot:
    input:
        "raw/{dataset}.csv"
    output:
        "plots/{dataset}.pdf"
    shell:
        "somecommand {input} {output}"
```

SNAKEMAKE R

Requires `rpy2`

```
rule mapRNAseq:
    input:
        file1 = "path/to/data/file1.bam",
        file2 = "path/to/data/file2.bam"
    output:
        "path/to/qlf_table.csv"
    run:
        R("""
        bamfiles <- cbind({input.file1},{input.file2})
        {output} <- featureCounts(bamfiles, annot.inbuilt="hg19", strandspecific=2)
        ...
        ...
        write.csv(qlf_table, file={output}, quote = FALSE)
        """)
```

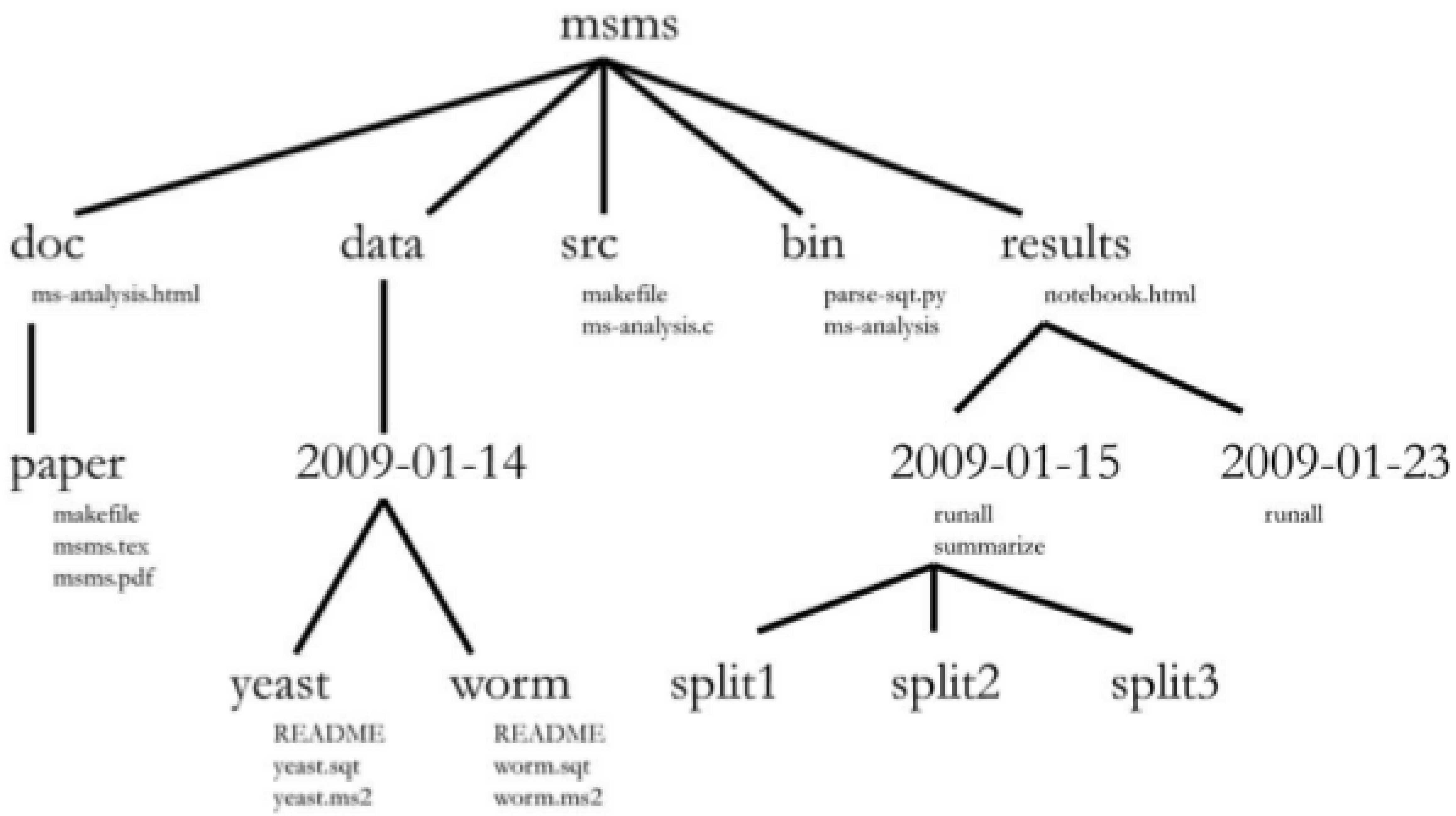
SNAKEMAKE R

```
rule GM19_genes_edgeR:
    input:
        "results/2019-02-28/GM19_gene_counts.rds"
    output:
        "results/2019-03-25/GM19_rawdata_table.csv",
        "results/2019-03-25/GM19_genelen.csv",
        "results/2019-03-25/GM19_normalized.txt"
    conda:
        "../../envs/edgeR.yaml"
    threads: 4
    script:
        "../../scripts/dge.R"
```

TEST DATA

```
snakemake --use-conda --directory .test
```

ORGANIZING BIOINFORMATICS PROJECTS


msms
doc
ms-analysis.html
data
src
makefile
ms-analysis.c
bin
parse-sqt.py
ms-analysis
results
notebook.html
paper
makefile
msms.tex
msms.pdf
2009-01-14
2009-01-15
runall
summarize
2009-01-23
runall
yeast
README
yeast.sqt
yeast.ms2
worm
README
worm.sqt
worm.ms2
split1
split2
split3

HPC CLUSTER

INTRO TO GANYMEDE

- `ganymedeadmins@utdallas.edu`
- centos 7 with OpenHPC
- http://docs.oithpc.utdallas.edu
- Access to `normal`, `debug`, `genomics`, and `GPU1` queue

GETTING ACCESS

- Email `admin@ganymede` for access and to **genomics**
- Takes around a week
- You can log in with `ssh <NETID>@ganymede.utdallas.edu`

SOME BASIC COMMANDS

```
# Info about slurm cluster
sinfo
# List enabled modules
module list
# Available module
module av
# Load anaconda
# Must be done everytime you log in
module load anaconda3
# Cancel all jobs
scancel -u <NETID>
# Check the queue
squeue -p genomics
# Submit a job
sbatch --help
```

INSTALLING CONDA

```
pip install conda
# If that doesn't work
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh

# Make an environment
conda create -n smk -c bioconda snakemake
conda install -c bioconda nextflow
# OR
conda create -n nf -c bioconda nextflow

# Activate your env
conda activate smk
```

RUNNING THINGS

1. Email `admin@ganymede` for access and to **genomics**
2. Setup local cookiecutter
3. Log in `ssh eam150030@ganymede.utdallas.edu`
4. Load Modules
5. Rsync from local computer

```
rsync -avtr /media/enhancer/IMR90/data eam150030@ganymede.utdallas.edu:/home/eam150030/IMR9
```

   1. Activate conda env `conda activate smk`

RUNNING THINGS

Basic script

```
#!/bin/bash
#SBATCH --ntasks=1
#SBATCH --time=00:10:00
#SBATCH --mail-user=eam150030@utdallas.edu
#SBATCH --mail-type=ALL

bowtie2 file.fastq
...
rysnc output back
```

SNAKEMAKE ON SLURM

1. Have `snakemake` installed
2. Make a cluster config file

```
{
    "__default__" :
    {
        "account" : "eam150030",
        "time" : "00:15:00",
        "n" : 1,
        "partition" : "genomics"
    },
    "compute1" :
    {
        "time" : "00:20:00"
    }
}
```

1. Use snakemake to submit jobs

```
snakemake -j 999 --use-conda --cluster-config cluster.json \
    --cluster "sbatch -A {cluster.account} -p {cluster.partition}\
    -n {cluster.n}  -t {cluster.time}"
```

SLURM SCRIPT

```
#!/bin/bash
#SBATCH --ntasks=1
#SBATCH --time=00:10:00
#SBATCH --mail-user=eam150030@utdallas.edu
#SBATCH --mail-type=ALL

conda activate smk

snakemake -j 999 --use-conda --cluster-config cluster.json \
    --cluster "sbatch -A {cluster.account} -p {cluster.partition}\
    -n {cluster.n}  -t {cluster.time}"

rysnc -avtr results myLabComputer
```

NEXTFLOW

1. Have `nextflow` installed
2. Running a toy example

```
nextflow run rnatoy -with-singularity
```

- Ganymede doesn't have docker so you can't use `-with-docker`

1. Executing using slurm (docs)

```
# nextflow.config
process {
  executor = 'slurm'
  queue = 'genomics'
}
```

1. Run pipeline

```
nextflow run tutorial.nf
nextflow -c nextflow.config run rnatoy -with-singularity
```

PYTORCH

```
from __future__ import print_function
import torch
torch.cuda.is_available()
x = torch.rand(5, 3)
print(x)
```

PYTORCH SLURM SCRIPT

```
#!/usr/bin/env bash
#SBATCH -J pytorchtest
#SBATCH -o pytorchtest-%A.out
#SBATCH -e pytorchtest-%A.err
#SBATCH -p GPU1
#SBATCH --gres=gpu:1
#SBATCH -c 1
#SBATCH -t 00:01:00
#SBATCH --mail-user=eam150030@utdallas.edu
#SBATCH --mail-type=ALL

module purge
module load singularity
module load CUDA
# Assuming that the container has been copied to the user's /scratch directory
singularity exec docker://pytorch/pytorch python \
    /home/eam150030/pytorch-demo/pytorch_example.py
```

ERNA PREDICTION PIPELINE FROM GRO-SEQ INFB1 INDUCTION TIMECOURSE

Research

Global transcriptional activity dynamics reveal functional enhancer RNAs

Yoon Jung Kim,[1,2] Peng Xie,[1,2] Lian Cao,[1] Michael Q. Zhang,[1] and Tae Hoon Kim[1]

[1] *Department of Biological Sciences and Center for Systems Biology, University of Texas at Dallas, Richardson, Texas 75080, USA*

Active enhancers of the human genome generate long noncoding transcripts known as enhancer RNAs (eRNAs). How dynamic transcriptional changes of eRNAs are physically and functionally linked with target gene transcription remains unclear. To investigate the dynamic functional relationships among eRNAs and target promoters, we obtained a dense time series of GRO-seq and ChIP-seq data to generate a time-resolved enhancer activity map of a cell undergoing an innate antiviral immune response. Dynamic changes in eRNA and pre-mRNA transcription activities suggest distinct regulatory roles of enhancers. Using a criterion based on proximity and transcriptional inducibility, we identified 123 highly confident pairs of virus-inducible enhancers and their target genes. These enhancers interact with their target promoters transiently and concurrently at the peak of gene activation. Accordingly, their physical disassociation from the promoters is likely involved in post-induction repression. Functional assessments further establish that these eRNAs are necessary for full induction of the target genes and that a complement of inducible eRNAs functions together to achieve full activation. Lastly, we demonstrate the potential for eRNA-targeted transcriptional reprogramming through targeted reduction of eRNAs for a clinically relevant gene, *TNFSF10*, resulting in a selective control of interferon-induced apoptosis.

[Supplemental material is available for this article.]

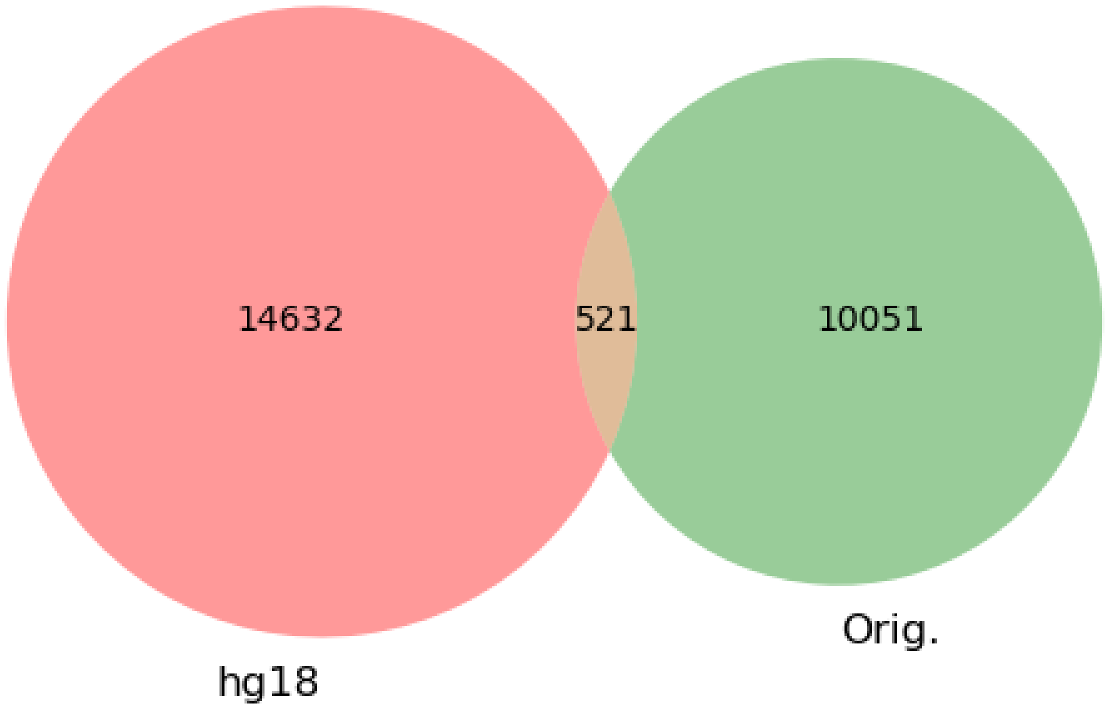
OVERVIEW

- Reproducing GM18
- Predicted IMR90 eRNAs
- Compared IMR90 Predicted Enhancers to GM
- Used Homer scripts to find DE of eRNAs and Genes
- Gene Centric vs. Enhancer Centric

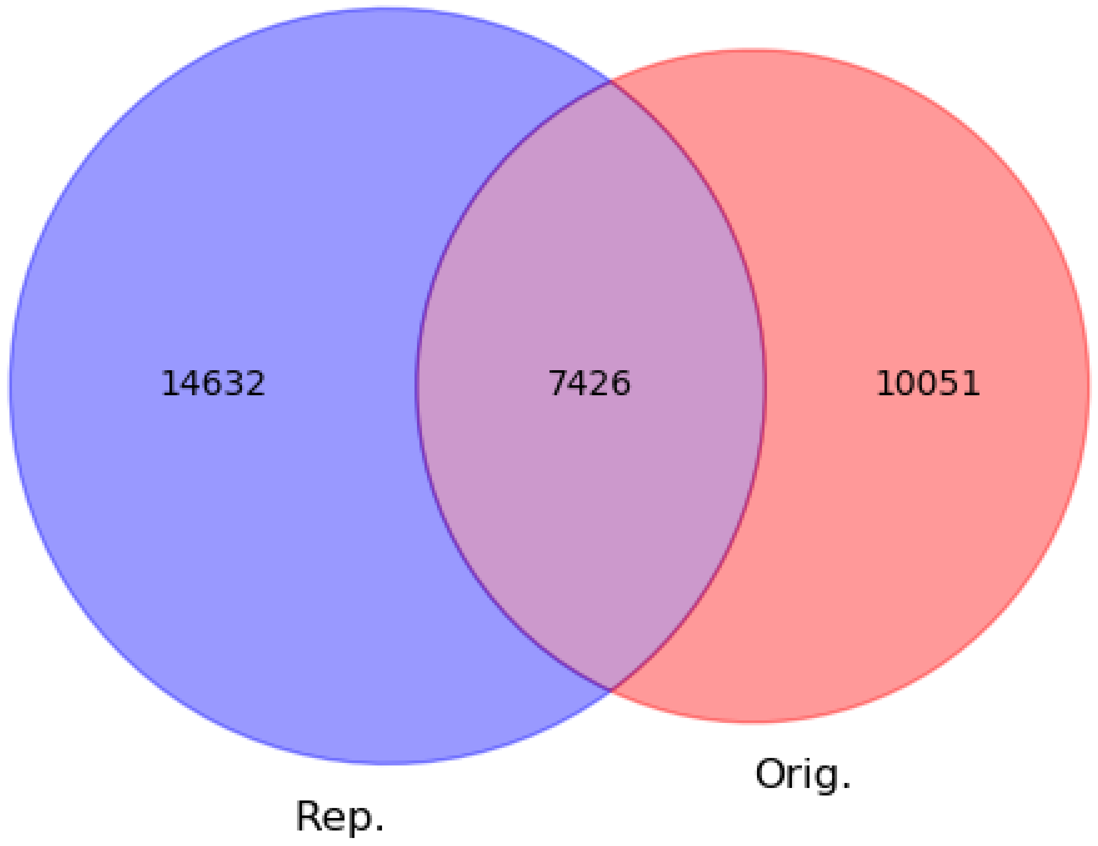
REPRODUCING GM18

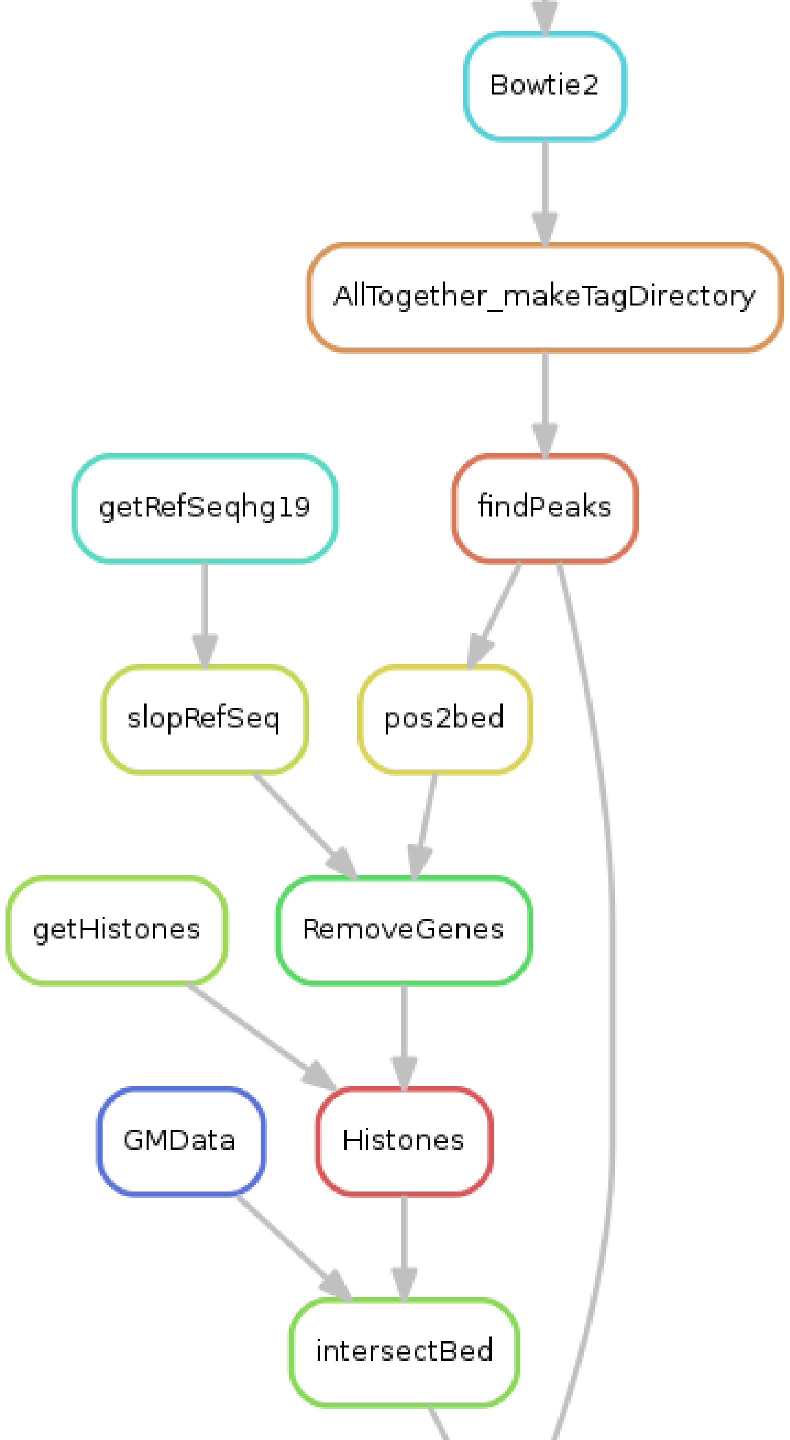
- hg18 vs hg19
- Overpredicting eRNA transcripts
- Past Issue
    - What I thought Peng sent me
    - hg18 -> eRNAs -> Me
    - What actually happened
    - hg18 -> eRNAs -> LiftOver -> hg19 -> Me
- Main issue is homer uniqmap

Without Liftover hg18 vs. GM eRNAs



Reproduction vs GM eRNAs




Bowtie2
AllTogether_makeTagDirectory
getRefSeqhg19
findPeaks
slopRefSeq
pos2bed
getHistones
RemoveGenes
GMData
Histones
intersectBed

ACTUAL PIPELINES

- GM18
- IMR19
- All

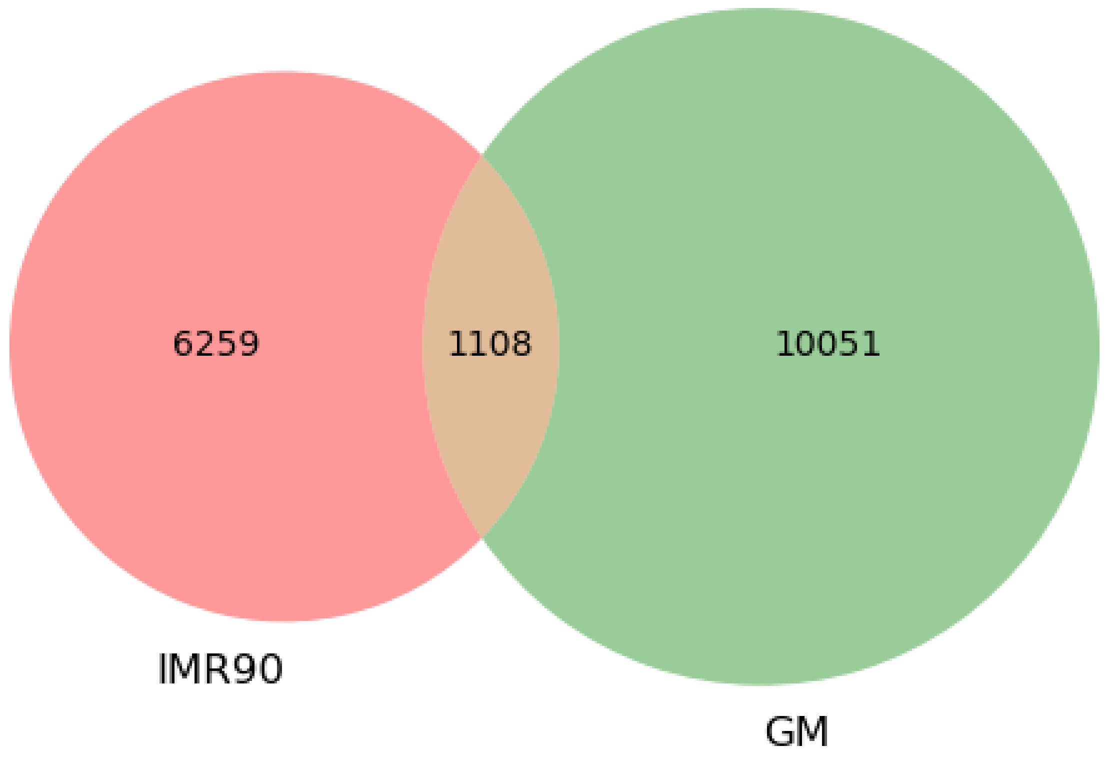
PREDICTED IMR90 ERNAS

Changes from GM18

- hg19
- No liftover

COMPARED IMR90 PREDICTED ENHANCERS TO GM

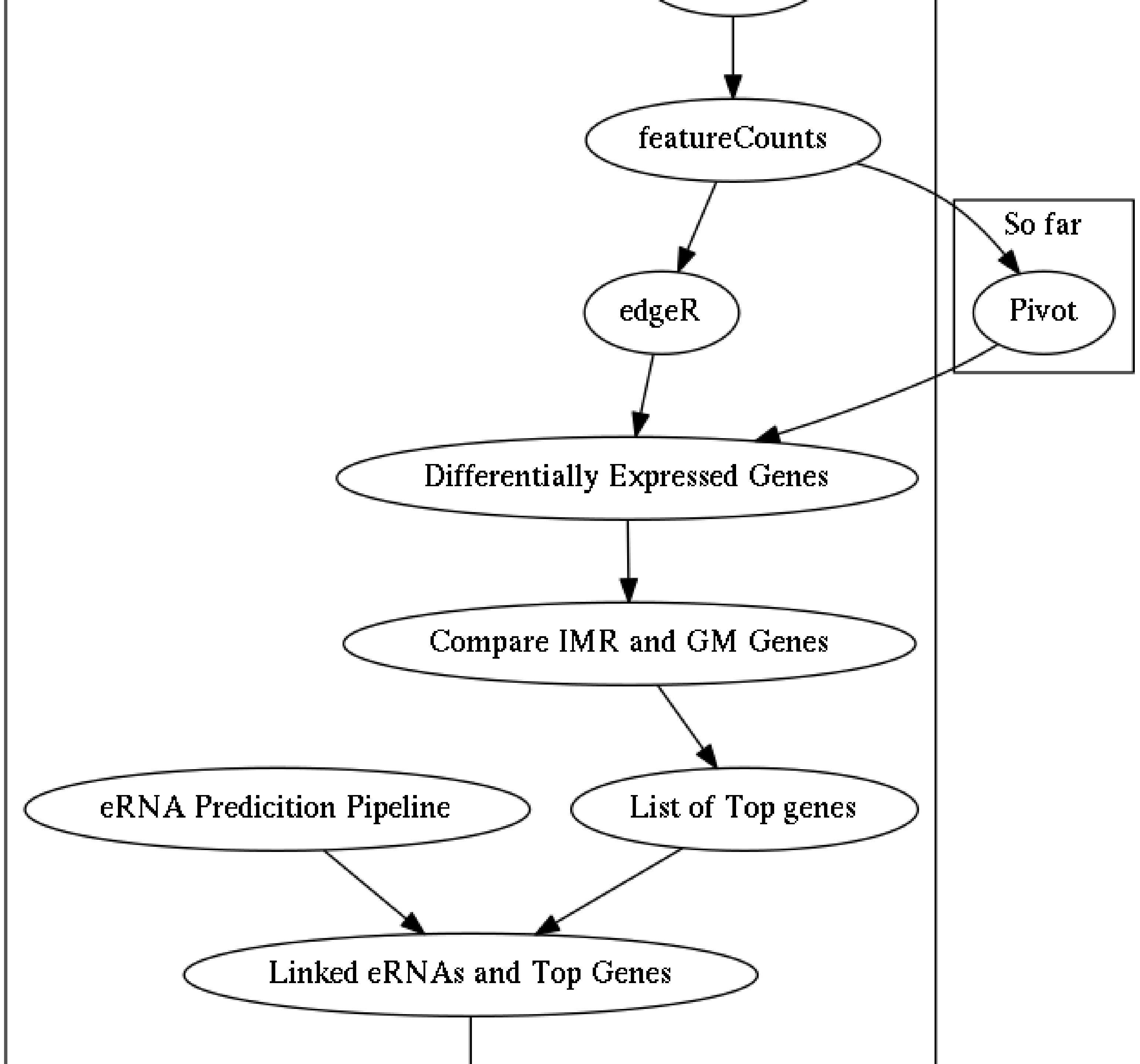
IMR90 vs GM eRNAs



USED HOMER SCRIPTS TO FIND DE OF ERNAS AND GENES

GENE CENTRIC VS. ENHANCER CENTRIC

- Peng's approach
  - Took enhancers that were expressed deferentially
  - Linked them to Genes within 200Kb
- New approach
  - Take genes that are deferentially expressed
  - Link the Enhancers to those genes


featureCounts
So far
edgeR
Pivot
Differentially Expressed Genes
Compare IMR and GM Genes
eRNA Predicition Pipeline
List of Top genes
Linked eRNAs and Top Genes

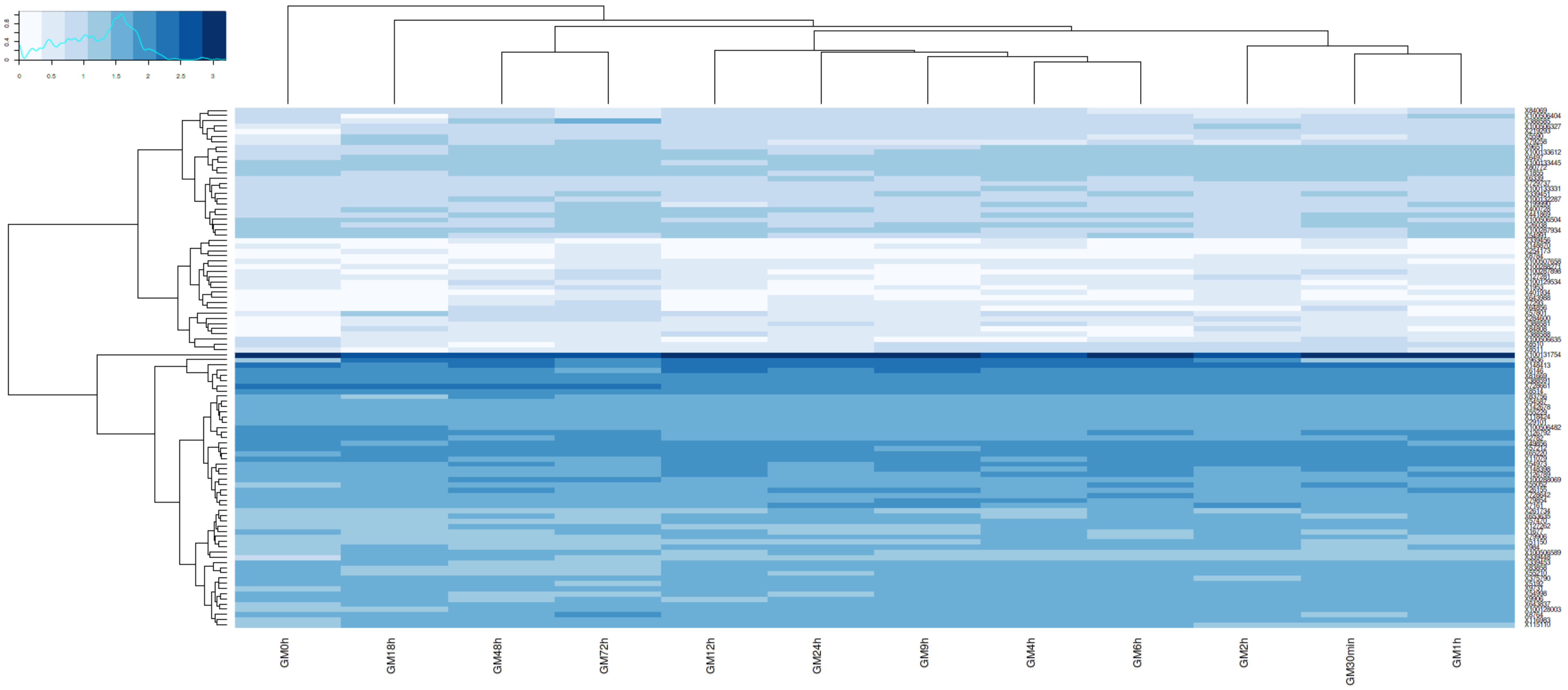
FEATURECOUNTS

- Tried CLI version
- Couldn't get examples to work with edgeR
- Switched to R package

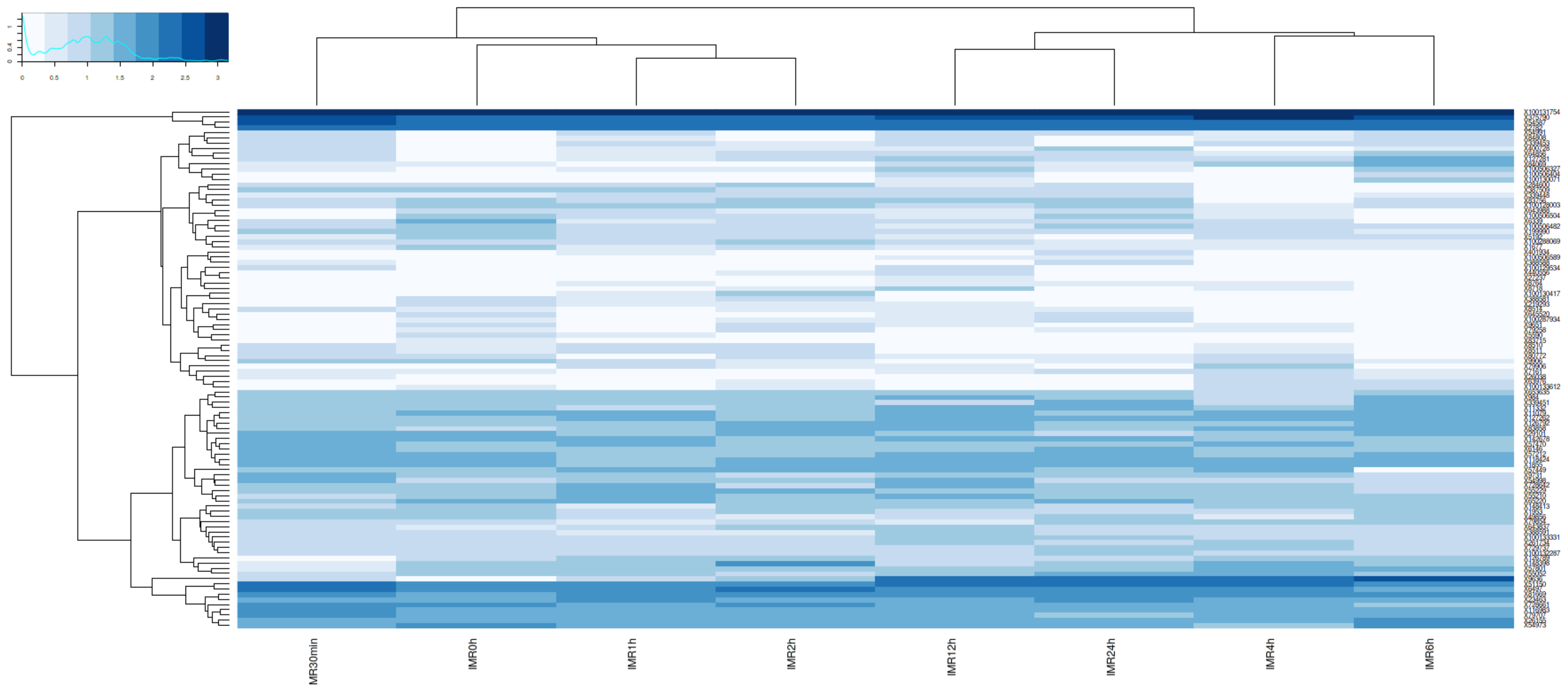
EDGER

- Error with exactTest
- Defaulted to writing raw DGEList to file
- Used in Pivot

RESULTS



0
0.4
0.8
0
0.5
1
1.5
2
2.5
3
GM0h
GM18h
GM48h
GM72h
GM12h
GM24h
GM9h
GM4h
GM6h
GM2h
GM30min
GM1h


0
0.4
1
0
0.5
1
1.5
2
2.5
3
MR30min
IMR0h
IMR1h
IMR2h
IMR12h
IMR24h
IMR4h
IMR6h
X100131754
X375790
X54587
X2192
X51001
X84876
X339453
X400728
X64856
X127281
X84069
X100506327
X100506404
X100130071
X284600
X387579
X339448
X83756
X100128003
X643988
X100506504
X6339
X100506482
X199990
X5192
X100288069
X1677
X401934
X100506589
X388588
X100129534
X440556
X27237
X8764
X8718
X100130417
X388581
X219293
X8514
X645520
X100287934
X9651
X79258
X5590
X63715
X8510
X8511
X80772
X9906
X79906
X7161
X29108
X63976
X100133612
X653635
X984
X339451
X1132
X1029
X127262
X126792
X29101
X142678
X57470
X6146
X57212
X118424
X1855
X57449
X9731
X54998
X728642
X55229
X55210
X65220
X148413
X1953
X100133331
X261734
X100132287
X126789
X148398
X57801
X55052
X51150
X81669
X23463
X728661
X116983
X2907
X26155
X54973

TAKEAWAYS

- Divide by median read values
- Need to go from -2 to 2 to show kinetics

FUTURE

- Kinetics of the eRNAs
- Compare IMR and GM Genes
- Get List of Top Genes
- Link eRNAs and Top genes