PISCES: a package for quantitation and QC of big mRNA-seq datasets

Matt Shirley (@mdshw5/twitter/github)
Investigator – Novartis Institute for Biomedical Research
February 16, 2017
Why a new pipeline?

1. New tools are faster
2. Tooling around new tools is lacking
   – Expression QC
   – Genotyping/fingerprinting
3. Automation for reprocessing large datasets
4. Reproducibility
What is PISCES?

PISCES is a package that eases the burden of processing large numbers of mRNA-seq libraries, and subsequently reducing errors in parameter selection and QC validation and consisting of three analysis modules:

1. **Single sample analysis of individual mRNA-seq libraries**
   - species detection, SNP fingerprinting, library geometry detection, and quantitation using salmon

2. **Multiple sample aggregation of analysis results**
   - summarization, TMM normalization, and differential expression analysis of multiple libraries to produce data formats ready for visualization and further analysis

3. **Multiple sample aggregation of quality control (QC) results**
   - visualization of mRNA-seq library QC metrics
PISCES implementation details

1. PISCES is implemented as a python3 package
   - bundled with all necessary dependencies to enable reproducible analysis and easy deployment

2. Configuration files are specified to:
   - build transcriptome indices
   - supply sample metadata
   - define contrasts for differential expression analysis using DEseq2
   - define default program parameters

3. Development versions will be available on Bitbucket, with python packages installable using pip.
PISCES stats at Novartis Oncology (December 2016)

1. **2,894** RNA-seq samples processed
   - ~30 CPU years for our previous cufflinks-based pipeline
   - ~2/3 CPU years for PISCES
   - We can reprocess TCGA, GTEx... When we need

2. **9,475** lines of code
   - 8757 python
   - 718 R

3. **Six “stable” releases**
   
   2015-12-08
   
   ![Matthew Shirley](image) acab3c5 initial commit of SVN version
PISCES workflow

- *pisces index* ➤ Once
- *pisces run* ➤ Once each sample
- *pisces qc* ➤ Once each experiment
- *pisces summarize* ➤ Once each experiment
PISCES “index”

1. Creates transcriptome FASTA from input GTFs and genomic FASTAs
   – Optionally masks sequence – ATCccccGTA → ATCNNNNNGTA
   – Add as many as you need: e.g. mouse/human xenograft

2. Incorporates “extra” user-defined FASTA files
   – e.g. viral sequences, repetitive elements

3. Generates salmon and bowtie2 index files
   – Bowtie2 indices are only used for QC metrics
Reproducible index builds

},
"xeno": {
    "gencode": {
        "gtfs": ["/da/onc/harmonization/pisces/annotations/gencode_v25/gencode.v25.annotation.gtf",
                 "/da/onc/harmonization/pisces/annotations/gencode_vM10/gencode.vM10.annotation.gtf"],
        "fastas": ["/db/nibrgenome/NG00009.0/fasta/hg38.fa", "/db/nibrgenome/NG00009.0/fasta/mm10.fa"],
        "extra_fastas": [],
        "index": "/da/onc/harmonization/pisces/indices/gencode_v25_vM10",
        "options": {}
    },
    "gencode_plus": {
        "gtfs": ["/da/onc/harmonization/pisces/annotations/gencode_v25/gencode.v25.annotation.gtf",
                 "/da/onc/harmonization/pisces/annotations/gencode_vM10/gencode.vM10.annotation.gtf"],
        "fastas": ["/db/nibrgenome/NG00009.0/fasta/hg38.fa", "/db/nibrgenome/NG00009.0/fasta/mm10.fa"],
        "extra_fastas": ["/home/merkijal/annotations/dfam/DFam.named.fa", "/home/skekepe1/viper/db/160205_virus_nucl.fa"],
        "index": "/da/onc/harmonization/pisces/indices/gencode_v25_vM10_plus",
        "options": {}
    },
    "gencode_plus_masked": {
        "gtfs": ["/da/onc/harmonization/pisces/annotations/gencode_v25/gencode.v25.annotation.gtf",
                 "/da/onc/harmonization/pisces/annotations/gencode_vM10/gencode.vM10.annotation.gtf"],
        "fastas": ["/db/nibrgenome/NG00009.0/fasta/hg38.fa", "/db/nibrgenome/NG00009.0/fasta/mm10.fa"],
        "extra_fastas": ["/home/merkijal/annotations/dfam/DFam.named.fa", "/home/skekepe1/viper/db/160205_virus_nucl.fa"],
        "index": "/da/onc/harmonization/pisces/indices/gencode_v25_vM10_plus_masked",
        "options": {"masked": true}
Reproducible index builds

```json
{
    "human": {
        "gencode": {
            "gtfs": ["/da/onc/harmonization/pisces/annotations/gencode_v25/gencode.v25.annotation.gtf"],
            "fastas": ["/db/nirbgenome/NG00009.0/fasta/hg38.fa"],
            "extra_fastas": [],
            "index": "/da/onc/harmonization/pisces/indices/gencode_v25",
            "options": {}
        },
        "gencode_plus": {
            "gtfs": ["/da/onc/harmonization/pisces/annotations/gencode_v25/gencode.v25.annotation.gtf"],
            "fastas": ["/db/nirbgenome/NG00009.0/fasta/hg38.fa"],
            "extra_fastas": ["/home/merkija1/annotations/dfam/dfam.named.fa", "/home/skewepel/viper/db/160205_virus_nucl.fa"],
            "index": "/da/onc/harmonization/pisces/indices/gencode_v25_plus",
            "options": {}
        },
        "gencode_plus_masked": {
            "gtfs": ["/da/onc/harmonization/pisces/annotations/gencode_v25/gencode.v25.annotation.gtf"],
            "fastas": ["/db/nirbgenome/NG00009.0/fasta/hg38.fa"],
            "extra_fastas": ["/home/merkija1/annotations/dfam/dfam.named.fa", "/home/skewepel/viper/db/160205_virus_nucl.fa"],
            "index": "/da/onc/harmonization/pisces/indices/gencode_v25_plus_masked",
            "options": {"masked": true}
        }
    },
    "mouse": {
        "genomes": ["mm10", "mm9"],
        "options": {}
    }
}
```
PISCES workflow

- pisces index ➤ Once
- pisces run ➤ Once each sample
- pisces qc ➤ Once each experiment
- pisces summarize ➤ Once each experiment
PISCES “run”

• Minimal examples
  – pisces run -fq1 r1_1.fq.gz r1_2.fq -fq2 r2_1.fq ...
  – pisces run -fq1 r1.fq.gz
  – pisces run … --sample-type xeno --salmon-indices gencode
  – pisces run … --threads 8 --name patient_10_liver
  – pisces run … --config user-config.json
  – All parameters have defaults, or are inferred from the FASTQ files
PISCES “run”

(v0.6) -bash-4.1$ pisces run -h
usage: pisces run -fq1 [FQ1 [FQ1 ...]] [-fq2 [FQ2 [FQ2 ...]]] [-n NAME]
[-o OUT] [-p THREADS] [-t {human,mouse,xeno}]
[-l {IU,ISF,ISR}] [--scratch-dir SCRATCH_DIR] [--overwrite]
[--salmon-indices [SALMON_INDICES [SALMON_INDICES ...]]]
[--no-vcf] [-c CONFIG_FILE] [-h]

required arguments:
-fq1 [FQ1 [FQ1 ...]]  space-separated list of gzipped FASTQ read 1 files

optional arguments:
-fq2 [FQ2 [FQ2 ...]]  space-separated list of gzipped FASTQ read 2 files
-n NAME, --name NAME  sample name used in output files. default=auto
-o OUT, --out OUT  path to output directory. default=/path/to/$FQ1/PISCES
-p THREADS, --threads THREADS  total number of CPU threads to use default=1
-t {human,mouse,xeno}, --sample-type {human,mouse,xeno}  species of the sample library default=auto
-l {IU,ISF,ISR}, --libtype {IU,ISF,ISR}  library geometry for Salmon (http://salmon.readthedocs.org/en/latest/salmon.html#what-s-this-libtype) default=auto
--scratch-dir SCRATCH_DIR  path to scratch directory default=/scratch
--overwrite  overwrite existing files
--salmon-indices [SALMON_INDICES [SALMON_INDICES ...]]  salmon index names (defined in --config-file) default=['gencode_plus']
--no-alignment-qc  do not generate picard qc metrics
--make-bam  make a BAM file for visualization
--no-salmon  do not run salmon
--no-fastqp  do not generate read-level qc metrics
--no-vcf  do not generate vcf file
-c CONFIG_FILE, --config-file CONFIG_FILE  default=/usr/prog/onc/seqtools/pisces/v0.6/src/novartis-pisces/pisces/config.json

-h, --help
PISCES “run”
**PISCES “run” outputs**

<table>
<thead>
<tr>
<th>(v0.6) –bash-4.1$ ls</th>
</tr>
</thead>
<tbody>
<tr>
<td>BA-83-ZT03_1_fastqp.txt</td>
</tr>
<tr>
<td>BA-83-ZT03_1_fastqp.zip</td>
</tr>
<tr>
<td>BA-83-ZT03_2_fastqp.txt</td>
</tr>
<tr>
<td>BA-83-ZT03_2_fastqp.zip</td>
</tr>
</tbody>
</table>

- **--name “BA-83-ZT03”**

- **fastqp** Python clone of FastQC
  - [https://github.com/mdshw5/fastqp](https://github.com/mdshw5/fastqp)

- **fastq_fingerprint.vcf**: genotypes derived from kmer counts

- **pct_human_mouse**: estimate of mouse/human percent derived from beta-actin kmers

- **qcANALYSIS**: picard metrics from *100,000 downsampled alignments* using bowtie2

- Salmon directory contains one or more salmon quant.sf files corresponding to –salmon-indices defined in --config
PISCES workflow

- `pisces index` ➤ `Once`
- `pisces run` ➤ `Once each sample`
- `pisces qc` ➤ `Once each experiment`
- `pisces summarize` ➤ `Once each experiment`
PISCES “qc”

• Minimal examples
  – pisces qc --tab out.table --tall out.tall [dir1 [dir2...]]
  – pisces qc --metadata samples.csv
  – pisces qc --fingerprint [dir1 [dir2...]]

• --tab output gathers statistics in a wide table
• --tall output is a tidy table used for visualization
• --fingerprint produces a table of sample identities and pairwise probabilities
  – Use this to find sample swaps
PISCES “qc” Spotfire vis
PISCES “qc” Spotfire vis
PISCES “summarize”

• Minimal examples
  – pisces summarize [dir1 [dir2...]]
  – pisces summarize –metadata sample.csv
  – pisces summarize –metadata sample.csv –group-by cell_line –norm-by treatment –control-factor DMSO
  – pisces summarize –metadata sample.csv –deseq-contrasts contrasts.yaml –patsy ~treatment+cell_line

• Output files are prefixed by –name
PISCES “summarize”

• Minimal examples
  – `pisces summarize [dir1 [dir2...]]`
  – `pisces summarize -metadata sample.csv`
  – `pisces summarize -metadata sample.csv -group-by cell_line -norm-by treatment -control-factor DMSO`
  – `pisces summarize -metadata sample.csv -deseq-contrasts contrasts.yaml -patsy ~treatment+cell_line`

• Output files are prefixed by `–name`
PISCES “summarize”

• Output tables are genes/isoforms x samples (rows x columns)

• Read salmon files using tximport in DESeq2 package

• Annotation for gene-level summaries provided by https://github.com/stephenturner/annotables

• “Tidy” deseq table is 5 column: contrast, log2fc, log10p, basemean, stderr

• **Normalization**: TPM > remove mito/ribo genes > calculate TMM scaling on protein coding genes > TMM scale *all genes*
PISCES “summarize”

• Minimal examples
  – pisces summarize -metadata sample.csv -group-by cell_line -norm-by treatment -control-factor DMSO

metadata.csv:

SampleID,UUID,CellLine,Treatment,Time,Directory,Groups
A3_DMSO_6hr_R1,CA-96-XXXX,A375,DMSO,6hr,../CA-96IY67/PISCES,A3_DMSO_6hr
A3_DMSO_6hr_R2,YA-97-XXXX,A375,DMSO,6hr,../YA-97-IB67/PISCES,A3_DMSO_6hr
A3_DMSO_24hr_R1,WA-95-XXXX,A375,DMSO,24hr,../WA-95-XA65/PISCES,A3_DMSO_24hr
A3_DMSO_24hr_R2,SA-95-XXXX,A375,DMSO,24hr,../SA-95-XE65/PISCES,A3_DMSO_24hr
PISCES “summarize”

- pisces summarize -metadata sample.csv -deseq-contrasts contrasts.yaml -patsy ~Treatment~CellLine

contrasts.yaml

Treatment:
- [DrugA_1uM_6h, DMSO_0uM_6h]
- [DrugA_5uM_6h, DMSO_0uM_6h]
- [DrugA_1uM_16h, DMSO_0uM_16h]
- [DrugA_5uM_16h, DMSO_0uM_16h]
- [DrugB_1uM_6h, DMSO_0uM_6h]
- [DrugB_5uM_6h, DMSO_0uM_6h]
- [DrugB_1uM_16h, DMSO_0uM_16h]
- [DrugB_5uM_16h, DMSO_0uM_16h]
- [shRNA1_0uM_48h, Dox_0uM_48h]
- [shRNA2_0uM_48h, Dox_0uM_48h]
- [shRNA1_0uM_72h, Dox_0uM_72h]
- [shRNA2_0uM_72h, Dox_0uM_72h]
PISCES “summarize”

(v0.6) -bash-4.1$ pisces summarize -h
PISCES summary expression matrix and differential expression

Usage: summarize [options] [--exclude-genes=GENE]... [DIR> <DIR>...]

Options:
- n NAME, --name NAME
- q IDX, --salmon-quant SALMON_INDEX
- m META, --metadata METADATA_DIR
- r VAR, --group-by VAR
- b VAR, --norm-by VAR
- c FACTOR, --control-factor FACTOR
- d PATSY, --deseq-formula PATSY

a)
- i YAML, --deseq-contrasts YAML
- s BIOTYPE, --scale-tpm BIOTYPE
- e TPM, --median-expression TPM_CUTOFF
- t FILE, --spotfire-template FILE
- x GENE, --exclude-genes GENE
--exclude-ribosomal
--isoforms
--debug

Arguments:
<DIR> Directories containing `pisces run` analysis results

Output file base name [default: expression_matrix]
PISCES Salmon run to aggregate [default: gencode_plus]
CSV file describing contrast variables and sample names
Column name describing variable to group samples for normalization
Column name of the main variable used for within-group comparisons
Name of factor in '--norm-by' column used for within-group normalization
`patsy` notation to be passed to DESeq2 e.g: ~ treatment

YAML annotation of the contrasts of interest (see example)
TMM normalize using genes belonging to this ENSEMBL `biotype`
Exclude genes from TMM normalization that have expression
File path at which to create Spotfire template DXP
List of genes to exclude from TMM normalization
Exclude genes starting with RPS or RPL from TMM scaling
Output transcript isoform level matrices
Print debugging information
PISCES “summarize”

```bash
(v0.6) -bash-4.1$ ls *txt
expression_matrix.human.counts.txt
expression_matrix.human.deseq.tidy.txt
expression_matrix.human.deseq.txt
expression_matrix.human.log2fc.TMM-scaled.txt
expression_matrix.human.log2fc.txt
expression_matrix.human.raw.TMM-scaled.txt
expression_matrix.human.raw.txt
```

```
median_length  entrez  symbol  chr  start  end  strand  biotype  description
1933.075778584  7105  TSPAN6  X  100627109  100639991  -1  protein_coding  tetraspanin 6 [Source:NCBI Gene; Source:UCSC RefSeq]
825.33540625  64102  TNMD  X  100584082  100599885  1  protein_coding  tenomodulin [Source:HGNC Symbol; Source:UCSC RefSeq]
899.972561340993  8813  DPM1  20  50934867  50958555  -1  protein_coding  dolichyl-phosphate mannosyltransferase 1 [Source:NCBI Gene; Source:UCSC RefSeq]
3774.68332494071  57147  SCYL3  1  169849631  169894267  -1  protein_coding  SCY1-like, kinase-like 1 [Source:NCBI Gene; Source:UCSC RefSeq]
2659.59198414737  55732  C1orf112  1  169662007  169854080  1  protein_coding  chromosome 1 open reading frame 112 [Source:NCBI Gene; Source:UCSC RefSeq]
1839.80820111712  2268  FGR  1  27612064  27635277  -1  protein_coding  FGR proto-oncogene, S [Source:NCBI Gene; Source:UCSC RefSeq]
3234.26381228412  3075  CFH  1  196651878  196747504  1  protein_coding  complement factor H [Source:NCBI Gene; Source:UCSC RefSeq]
```
PISCES “summarize” Spotfire vis
Near term development goals

1. Normalization efforts
   - Best practices for TMM normalization
   - Investigate *shoal* for improving abundance estimates during *pisces summarize*
     - [https://github.com/COMBINE-lab/shoal](https://github.com/COMBINE-lab/shoal)

2. Automated re-identification of samples against a multi-sample VCF

3. Determine best practice for sequence masking

4. Open source visualizations

5. Publication
Takeaways

1. PISCES was developed to solve real-world issues:
   - Large number of datasets
   - Realize gains in efficiency using new “alignment-free” tools
   - Quick, routine QC of each sample, with fingerprinting identity
   - Identify sample/species swaps
   - Integrated tools to produce analysis or visualization-ready tables
   - Packaging of tool dependencies
   - Reproducibility of results
   - Standardization of RNAseq analysis within NIBR

2. PISCES builds on (mostly) open-source tools

3. I’ll be publishing the framework as a preprint Q1 2017
Acknowledgements

- NIBR
  - Josh Korn
  - Vivek Krishnamurthy Radhakrishna
  - Peter Skewes-Cox
  - Jason Merkin

- Stony Brook University
  - Rob Patro (salmon)
Thank you