Dynamic Statistical Comparisons

Gao Wang Matthew Stephens

gaow@uchicago.edu mstephens@uchicago.edu

University of Chicago http://github.com/stephenslab

DSC aids in reproducible research

Motivation

- Statistical comparisions typically performed is suboptimal in many ways
- DSC is an attempt to make statistical comparisions extensible & reproducible

Goal

- A *platform* to make it simple, even fun to perform DSCs in research [1]
- A *repository of DSCs* for projects in & outside Stephens Lab

DSCR (a.k.a. DSC1)

- The pilot phase of DSC design implemented in the R language
- Readily usable [2]
- Lacks flexibility for complex and capacity for large scale problems

DSC2: a multilingual system

- Seamless integration of routines written in R, Matlab and Python
- Flexible assembly of multiple statistical procedures at will
- Designed with modern workflow management system standards
- Focused on experience of simplicity
- Works in three major computational environments

Tell DSC how benchmark data is generated

```
simulate:
  exec: datamaker.R
  seed: R(1:50)
  params:
    tissue: Adipose-Subcutaneous,
            (Adipose-Subcutaneous, Lung)
    Nsamp: 2, 10, 50
    Ngene: 10000
    breaksample: FALSE, TRUE
    .alias: args = Pack()
  return: data, meta = R(data$meta)
```

Tell DSC we can generate data differently based on what we've done

```
simulate_normalized(simulate):
  params:
    voom.normalize: TRUE
```

Tell DSC to consider these counfounder control methods

```
correction:
  exec: SVA.R,
        RUV.R,
        myrna.R
  params:
    data: $data
    .alias: args = Pack()
  return: data
```

These are statistical methods tackling the same problem

... and these normalization methods for RNA-Seq data

```
transform:
  exec: voom.R, quasibinom.R,
        edgeRglm.R, DESeq2glm.R
  params:
    data: $data
    .alias: args = Pack()
  return: data
```

... and these differential expression (DE) analysis

```
test:
  exec: edgeR.R, DESeq2.R, ash.R,
        jointash.R, limma.R
  params:
    data: $data
    .alias: args = Pack()
    exec[1,2]:
      glm: TRUE, FALSE
    exec[5]:
      robust: TRUE, FALSE
  return: output
```

A method may have different flavors (parameters)

Give DSC a metric to evaluate DE methods

```
score:
  exec: score.R
  params:
    data: $meta
    work_dir: $output
  return: result
```

Finally, construct DSC pipelines by connecting various computational routines

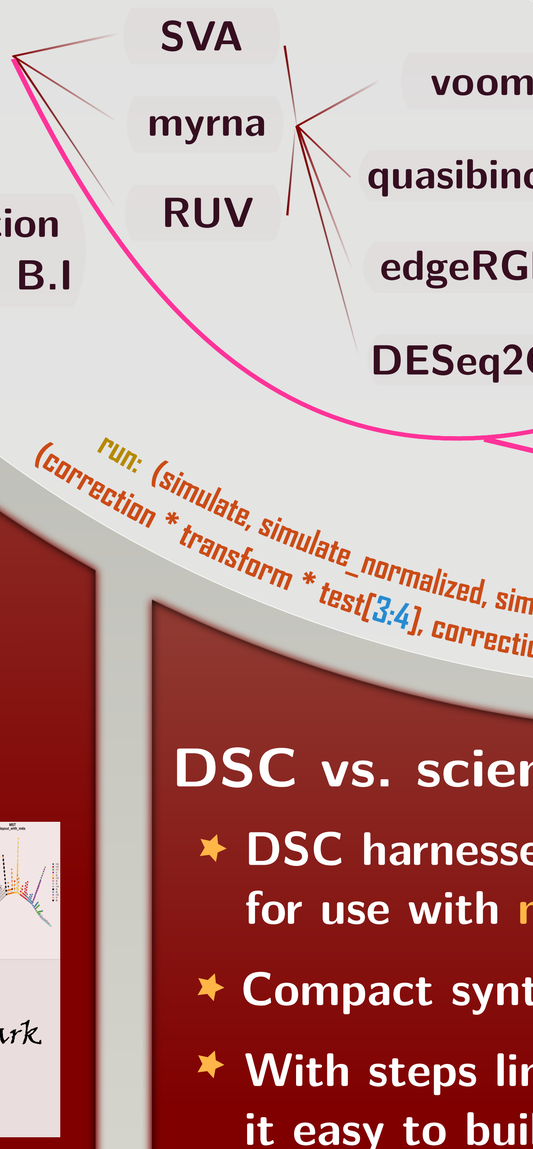
```
run: (simulate, simulate_normalized, simulate_partial_null, simulate_thinning) *
     (correction * transform * test[3:4], correction * transform[1] * test[5], test[1:2]) * score
```

Illustration

Comparison of methods for differential gene expression analysis of RNA sequencing data

Command interface

```
> dsc settings.dsc -j8 --host midway.rcc.uchicago.edu
INFO: DSC script exported to settings.html ...
INFO: Constructing DSC from settings.dsc ...
INFO: Running DSC jobs ...
simulate_1+correction_1+transform_1+test_3+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_1+test_4+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_2+test_3+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_2+test_4+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_3+test_3+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_3+test_4+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_4+test_3+score_1: 100% [==========] 5 20.8/s
simulate_1+correction_1+transform_4+test_4+score_1: 100% [==========] 5 20.8/s
... ...
simulate_1+test_1: 100% [==========] 2 20.8/s
simulate_1+test_2: 100% [==========] 2 20.8/s
... ...
DSC: 100% [==========] 116 6.1/s
INFO: Building output database rna_seq.rds ...
INFO: DSC complete!
INFO: Elapsed time 254.829 seconds.
```



DSC vs. scientific workflow systems

- DSC harnesses workflow systems [3] yet is tailored for use with methods development in data science
- Compact syntax to configure families of methods
- With steps linked by *variables* not *files* DSC makes it easy to build benchmarks from existing scripts

Reference & Links

[1] DSC2 URL / QR code (scan above)
github.com/stephenslab/dsc2

[2] DSCR URL
github.com/stephens999/dscr

[3] DSC2 uses pysos library for workflow management and redis cluster for job distribution
github.com/bopeng/sos
redis.io/topics/cluster-tutorial

[4] The RNA-Seq illustration is adapted from a DSCR implemention by Mengyin Lu
github.com/mengyin/dscr-gtex-total