Using configuration YAML files as a one-stop for all custom parameter needs

BYOB 06-08-2022

Caroline Esnault
NICHD Bioinformatics and Scientific Programming Core

What should *not* be done

Snakefile-hc

```
import sys
import os









# define targets to create
final_targets = ['data/featurecounts.tsv']

# uncomment line below to run differential expression
final_targets.append('downstream/rnaseq.html')

rule targets:
    """
    Final targets to create
    """
    input: final_targets


# featurecounts
rule featurecounts:
    """
    Count reads in annotations with featureCounts from the subread package
    """
    input:
        annotation='../raw-data/ecoli.gtf',
        bam='../raw-data/example.bam'
    params:
        strand='s2'     # change here to match library strandedness
    output:
        counts='data/featurecounts.txt'
    log:
        'logs/featurecounts.txt.log'
    shell:
        'featureCounts '
        '-{params.strand} -t exon -g derived_id ' # change to specify columns
        '-p '
        '-a {input.annotation} '
        '-o {output.counts} '
        '{input.bam} '
        '&> {log}'

```

comment out / uncomment depending on needs

hard-coded paths

hardcoded parameters to modify

test.pl

```perl
#!/usr/bin/perl -w

## Script to truncate the last 50bp of the sequences from sequence file
## (MiSeq for exemple).
## The goal is to have the same structure of sequences between HiSeq
##(101bp run) and MiSeq (151bp run)


my $n = 1;
my $seqline;
my $lenght = 73;       # can modify here the lenght of sequence to keep

# ask for the sequence file
print "fastq file:";
chomp($fastqfile = <STDIN>);

$output = $fastqfile;
$output =~ s/.txt/_truncated73.txt/;
$output =~ s/.fastq/_truncated73.txt/;

open OUTPUT, ">$output" or die "can't open '$output': $!";

open (FASTQFILE, $fastqfile);
    print "Screening the sequences\n";
    while (<FASTQFILE>) {
        $seqline = $_;
        chomp $seqline;
        # truncate at the first 101bp if it is a sequence lines (1 / 4)
        if ($n == 2) {
            $seqline = substr ($seqline, 0, $lenght);···
        }
        if ($n == 4) {
            $n = 0;
        }
        print OUTPUT "$seqline\n";
        $n += 1;
    }

close(FASTQFILE);


close OUTPUT;
```

Real-life example of what not to do

Goals

- move all parameters and variables OUTSIDE of the code
- no hardcoding in script
- main version of code fits all experiments
- easy 1-stop way to find which parameters were used

YAML language

definition:

YAML (rhymes with “camel”) is a data serialization language designed to be human-friendly and work well with modern programming languages for common everyday tasks.

```
Y : Yet                Y : YAML
A : Another            A : Ain’t
M : Markup             M : Markup
L : Language           L : Language
```
(source: redhat.com)

recursive acronym
(source yaml.org)

timeline:

- **2001**: 1st YAML framework written in Perl
- **2003**: Ruby 1st language to ship a YAML framework as part of its core language
- **2004**: YAML 1.0 specification published by Clark Evans, Oren Ben-Kiki, and Ingy döt Net
- **2005**: YAML 1.1
- **2006**: Kyrylo Simonov produced PyYAML and LibYAML
- **2009**: YAML 1.2
- **2020**: new YAML language design team began meeting regularly to discuss improvements
- **2021**: YAML 1.2.2

https://yaml.org/spec/1.2.2/

config.yaml

```yaml
# YAML
---

# comment
key: some_text

some_sequence: ['item1', 'item2']

another_sequence:
  - 'item1'
  - 'item2'

some_mapping: {'control': 'blue', 'mutant': 'green'}

another_mapping:
  'control': 'blue'
  'mutant': 'green'

mapping_of_mappings:
  mapping_of_sequence:
    - level1
    - leval2
    - level3
  nested_mapping_of_mappings:
    'red'
    'orange'
    'yellow'

plain:·
  This unquoted scalar
  spans many lines.

quoted:
  "So does this
  quoted scalar.\n"

```

YAML structure

for comments

three dashes (“---”) to separate directives
from document content

scalar

sequence (ordered)

mapping (not ordered)

single or double quote, or even no quote

structure is determined by *indentation* (not TAB!)

scalars (quoted or unquoted) can span many lines

https://yaml.org/spec/1.2.2/

Parsers exist in many languages

- perl

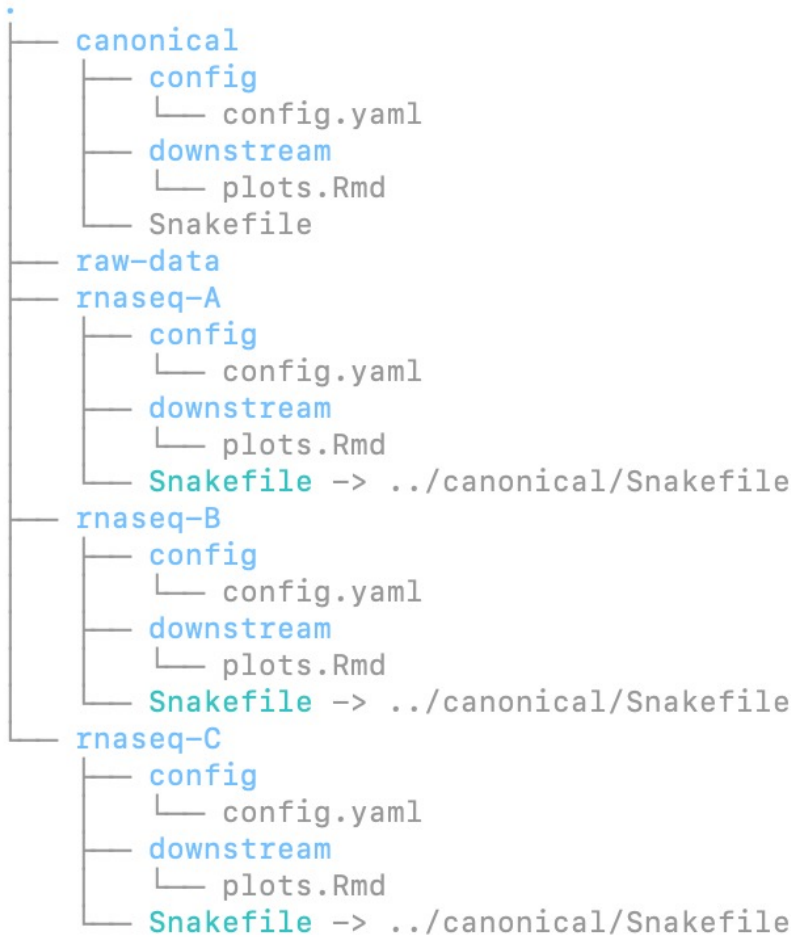
```perl
use strict;
use warnings;
use YAML::XS 'LoadFile';
use Data::Dumper;

my $config = LoadFile('config.yaml');

print Dumper($config);
```

- Java
- Javascript
- C/C++
- Rust
- Shell
- Python
- R
- ...
- anything? structure makes it possible to parse relatively easily

https://yaml.org/



Use unique script in complex project structure

main copy in canonical

change 1 copy to change all

can use the same config.yaml for python and downstream Rmd

Loading YAML in python

```
1  import sys
2  import os
3  import yaml
4
5
6  # load config
7  configfn= 'config/config.yaml'
8  config = yaml.safe_load(open(configfn))
9
10
```

Snakefile-hc

```
import sys
import os









# define targets to create
final_targets = ['data/featurecounts.tsv']

# uncomment line below to run differential expression
final_targets.append('downstream/rnaseq.html')

rule targets:
    """
    Final targets to create
    """
    input: final_targets


# featurecounts
rule featurecounts:
    """
    Count reads in annotations with featureCounts from the subread package
    """
    input:
        annotation='../raw-data/ecoli.gtf',
        bam='../raw-data/example.bam'
    params:
        strand='s2'     # change here to match library strandedness
    output:
        counts='data/featurecounts.txt'
    log:
        'logs/featurecounts.txt.log'
    shell:
        'featureCounts '
        '-{params.strand} -t exon -g derived_id ' # change to specify columns
        '-p '
        '-a {input.annotation} '
        '-o {output.counts} '
        '{input.bam} '
        '&> {log}'

```

config.yaml

```
# rnaseq-A YAML
---

# Snakefile parameters
# --------------------

# annotation
gtf: ../raw-data/ecoli.gtf

# BAM file
bam: ../raw-data/example.bam

# Extra featureCounts parameters
extra: '-s2 -t exon -g derived_id'

# output directory
outdir: data

# True of False to run differential expression
differential_expression: True


# downstream DE Rmd parameters
# ----------------------------

# title
title: 'Differential expression RNAseq A'

# sampletable
sampletable: '../config/sampletable.tsv'

# featurecounts file
featurecounts: '../data/featurecounts.txt'

# reference level for dds object
level.column: 'group'
level.ref: 'log'

# contrast(s): indicate for each contrast the ctrst.group,
# cntrst.test, ctrst.control
contrasts:
  mutant_vs_ctrl:
    ctrst.group: 'group'
    ctrst.test: 'exp'
    ctrst.control: 'log'

# genes of interest to plot, comment out if none
genes.to.plot:
  - 'G0-9381'
  - 'EG10747'

```

Snakefile bash

```
import sys
import os
import yaml


# load config
configfn= 'config/config.yaml'
config = yaml.safe_load(open(configfn))


# define targets to create
final_targets = [config['outdir'] + '/featurecounts.tsv']

if config['differential_expression']:
    final_targets.append('downstream/rnaseq.html')

rule targets:
    """
    Final targets to create
    """
    input: final_targets


# featurecounts
rule featurecounts:
    """
    Count reads in annotations with featureCounts from the subread package
    """
    input:
        annotation=config['gtf'],
        bam=config['bam']
    params:
        extra=config['extra']
    output:
        counts=config['outdir'] + '/featurecounts.txt'
    log:
        'logs/featurecounts.txt.log'
    shell:
        'featureCounts '
        '{params.extra} '
        '-p '
        '-a {input.annotation} '
        '-o {output.counts} '
        '{input.bam} '
        '&> {log}'

```

config.yaml

```
# rnaseq-A YAML
---

# Snakefile parameters
# --------------------

# annotation
gtf: ../raw-data/ecoli.gtf

# BAM file
bam: ../raw-data/example.bam

# Extra featureCounts parameters
extra: '-s2 -t exon -g derived_id'

# output directory
outdir: data

# True of False to run differential expression
differential_expression: True


# downstream DE Rmd parameters
# ----------------------------

# title
title: 'Differential expression RNAseq A'

# sampletable
sampletable: '../config/sampletable.tsv'

# featurecounts file
featurecounts: '../data/featurecounts.txt'

# reference level for dds object
level.column: 'group'
level.ref: 'log'

# contrast(s): indicate for each contrast the ctrst.group,
# cntrst.test, ctrst.control
contrasts:
  mutant_vs_ctrl:
    ctrst.group: 'group'
    ctrst.test: 'exp'
    ctrst.control: 'log'

# genes of interest to plot, comment out if none
genes.to.plot:
  - 'G0-9381'
  - 'EG10747'

```

Snakefile bash

```
import sys
import os
import yaml


# load config
configfn= 'config/config.yaml'
config = yaml.safe_load(open(configfn))


# define targets to create
final_targets = [config['outdir'] + '/featurecounts.tsv']

if config['differential_expression']:
    final_targets.append('downstream/rnaseq.html')

rule targets:
    """
    Final targets to create
    """
    input: final_targets


# featurecounts
rule featurecounts:
    """
    Count reads in annotations with featureCounts from the subread package
    """
    input:
        annotation=config['gtf'],
        bam=config['bam']
    params:
        extra=config['extra']
    output:
        counts=config['outdir'] + '/featurecounts.txt'
    log:
        'logs/featurecounts.txt.log'
    shell:
        'featureCounts '
        '{params.extra} '
        '-p '
        '-a {input.annotation} '
        '-o {output.counts} '
        '{input.bam} '
        '&> {log}'

```

config.yaml

```
# rnaseq-A YAML
---

# Snakefile parameters
# --------------------

# annotation
gtf: ../raw-data/ecoli.gtf

# BAM file
bam: ../raw-data/example.bam

# Extra featureCounts parameters
extra: '-s2 -t exon -g derived_id'

# output directory
outdir: data

# True of False to run differential expression
differential_expression: True


# downstream DE Rmd parameters
# ----------------------------

# title
title: 'Differential expression RNAseq A'

# sampletable
sample
# feat
featur

# refe
level.
level.

# cont
# cntr
contrasts:
  mutant_vs_ctrl:
    ctrst.group: 'group'
    ctrst.test: 'exp'
    ctrst.control: 'log'

# genes of interest to plot, comment out if none
genes.to.plot:
  - 'G0-9381'
  - 'EG10747'

```

```
In [21]: config
Out[21]:
{'gtf': '../raw-data/ecoli.gtf',
 'bam': '../raw-data/example.bam',
 'extra': '-s2 -t exon -g derived_id',
 'outdir': 'data',
 'differential_expression': True}
```

Loading YAML in R

```
34
35 library(yaml)
36 cfgfn <- '../config/congif.yaml'
37 cfg <- read_yaml(cfgfn)
38
39
```

plots.Rmd bash

```
---
output:
  html_document:
    code_folding: hide
    toc: true
    toc_float: true
    toc_depth: 3
---


```{r}
knitr::opts_chunk$set(message=FALSE, warning=FALSE)
```

```{r libraries}
library(ggplot2)
library(tidyr)
library(dplyr)
library(DESeq2)
library(yaml)
```

```{r configs}
# read config yaml
cfg <- read_yaml('../config/config.yaml')
```

`r cfg[['title']]`

```{r coldata_setup, cache=TRUE}
# Set up all of the metadata for the samples and experimental design.
colData <- read.table(cfg[['sampletable']], sep='\t', header=TRUE, stringsAsFactors=FALSE)

# relevel factor
level.col <- cfg[['level.column']]
level.ref <- cfg[['level.ref']]
colData[[level.col]] <- as.factor(colData[[level.col]])
colData[[level.col]] <- relevel(colData[[level.col]], ref=level.ref)
rownames(colData) <- colData[,1]

knitr::kable(colData)
```

DESeq Data Set

```{r dds}
# count matrix from featureCounts
cts <- read.table(cfg[['featurecounts']], sep='\t', header=TRUE, row.names='Geneid')
cts <- cts[colData[['samplename']]]

# dds
dds <- DESeqDataSetFromMatrix(countData = cts,
                              colData = colData,
                              design = formula(paste("~",level.col)))
dds <- DESeq(dds)
print(dds)
```

Contrasts results
```

config.yaml

```
# rnaseq-A YAML
---

# Snakefile parameters
# --------------------

# annotation
gtf: ../raw-data/ecoli.gtf

# BAM file
bam: ../raw-data/example.bam

# Extra featureCounts parameters
extra: '-s2 -t exon -g derived_id'

# output directory
outdir: data

# True of False to run differential expression
differential_expression: True


# downstream DE Rmd parameters
# ----------------------------

# title
title: 'Differential expression RNAseq A'

# sampletable
sampletable: '../config/sampletable.tsv'

# featurecounts file
featurecounts: '../data/featurecounts.txt'

# reference level for dds object
level.column: 'group'
level.ref: 'log'

# contrast(s): indicate for each contrast the ctrst.group,
# cntrst.test, ctrst.control
contrasts:
  mutant_vs_ctrl:
    ctrst.group: 'group'
    ctrst.test: 'exp'
    ctrst.control: 'log'

# genes of interest to plot, comment out if none
genes.to.plot:
  - 'G0-9381'
  - 'EG10747'

```

plots.Rmd bash

```
---
output:
  html_document:
    code_folding: hide
    toc: true
    toc_float: true
    toc_depth: 3
---


```{r}
knitr::opts_chunk$set(message=FALSE, warning=FALSE)
```

```{r libraries}
library(ggplot2)
library(tidyr)
library(dplyr)
library(DESeq2)
library(yaml)
```

```{r configs}
# read config yaml
cfg <- read_yaml('../config/config.yaml')
```

`r cfg[['title']]`

```{r coldata_setup, cache=TRUE}
# Set up all of the metadata for the samples and experimental design.
colData <- read.table(cfg[['sampletable']], sep='\t', header=TRUE, stringsAsFactors=FALSE)

# relevel factor
level.col <- cfg[['level.column']]
level.ref <- cfg[['level.ref']]
colData[[level.col]] <- as.factor(colData[[level.col]])
colData[[level.col]] <- relevel(colData[[level.col]], ref=level.ref)
rownames(colData) <- colData[,1]

knitr::kable(colData)
```

DESeq Data Set

```{r dds}
# count matrix from featureCounts
cts <- read.table(cfg[['featurecounts']], sep='\t', header=TRUE, row.names='Geneid')
cts <- cts[colData[['samplename']]]

# dds
dds <- DESeqDataSetFromMatrix(countData = cts,
                              colData = colData,
                              design = formula(paste("~",level.col)))
dds <- DESeq(dds)
print(dds)
```

Contrasts results
```

config.yaml

```
# rnaseq-A YAML
---

# Snakefile parameters
# --------------------


# annotation
gtf: ../raw-data/ecoli.gtf

# BAM file
bam: ../raw-data/example.bam

# Extra featureCounts parameters
extra: '-s2 -t exon -g derived_id'

# output directory
outdir: data

# True of False to run differential expression
differential_expression: True


# downstream DE Rmd parameters
# ----------------------------

# title
title: 'Differential expression RNAseq A'

# sampletable
sampletable: '../config/sampletable.tsv'

# featurecounts file
featurecounts: '../data/featurecounts.txt'

# reference level for dds object
level.column: 'group'
level.ref: 'log'

# contrast(s): indicate for each contrast the c
# cntrst.test, ctrst.control
contrasts:
  mutant_vs_ctrl:
    ctrst.group: 'group'
    ctrst.test: 'exp'
    ctrst.control: 'log'

# genes of interest to plot, comment out if none
genes.to.plot:
  - 'G0-9381'
  - 'EG10747'

```

```
> cfg
$gtf
[1] "../raw-data/ecoli.gtf"

$bam
[1] "../raw-data/example.bam"

$extra
[1] "-s2 -t exon -g derived_id"

$outdir
[1] "data"

$differential_expression
[1] TRUE

$title
[1] "Differential expression RNAseq A"

$sampletable
[1] "../config/sampletable.tsv"

$featurecounts
[1] "../data/featurecounts.txt"

$level.column
[1] "group"

$level.ref
[1] "log"

$contrasts
$contrasts$mutant_vs_ctrl
$contrasts$mutant_vs_ctrl$ctrst.group
[1] "group"

$contrasts$mutant_vs_ctrl$ctrst.test
[1] "exp"

$contrasts$mutant_vs_ctrl$ctrst.control
[1] "log"
```

Loop through contrasts

plots.Rmd

```
# Contrasts results

```{r contrasts}
# results
res <- list()
for (ctrst in names(cfg[['contrasts']])) {
    res[[ctrst]] <- lfcShrink(dds,
                              contrast=c(cfg[['contrasts']][[ctrst]][['ctrst.group']],
                                         cfg[['contrasts']][[ctrst]][['ctrst.test']],
                                         cfg[['contrasts']][[ctrst]][['ctrst.control']]),
                              type='normal')
    cat(paste0('\n\n## ', ctrst, '\n\n'))
    print(res[[ctrst]])
}
```
```

config.yaml

```
# contrast(s): indicate for each contrast the ctrst.group,
# cntrst.test, ctrst.control
contrasts:
  mutant_vs_ctrl:
    ctrst.group: 'group'
    ctrst.test: 'exp'
    ctrst.control: 'log'
```

Define conditional rule / chunk

plots.Rmd

```
# Conditional plotting

```{r}
if ('genes.to.plot' %in% names(cfg)) {
    eval_plot <- TRUE
} else {
    eval_plot <- FALSE
    cat('\n\nNot run\n\n')
}
```

```{r plots, eval=eval_plot}
for (gene in cfg[['genes.to.plot']]) {
    plotCounts(dds, gene=gene, intgroup=cfg[['contrasts']][[ctrst]][['ctrst.group']])
}
```
```

config.yaml

```
# genes of interest to plot, comment out if none
genes.to.plot:
  - 'G0-9381'
  - 'EG10747'

```

Snakefile

```

if config['differential_expression']:
    final_targets.append('downstream/rnaseq.html')

```

```
# conditional rule: downstream DESeq2 analysis
if config['differential_expression']:

    rule rnaseq_rmarkdown:
        """
        Run and render the RMarkdown file that performs differential expression
        """
        input:
            featurecounts=rules.featurecounts.output,
            rmd='downstream/rnaseq.Rmd',
            sampletable=config['sampletable']
        output:
            'downstream/rnaseq.html'
        log:
            'downstream/rnaseq.log'
        shell:
            'Rscript -e '
            '''"rmarkdown::render('{input.rmd}')" '''
            '> {log} 2>&1'
```

config.yaml

```

# True of False to run differential expression
differential_expression: True


```

```
commit daf9178f66a691f54901ba940d268fd8e5660078
Author: Caroline Esnault <caroline.esnault@nih.gov>
Date:   Wed Jun 8 21:35:03 2022 -0400

    rename contrast

diff --git a/rnaseq-A/config/config.yaml b/rnaseq-A/config/config.yaml
index 669d7b8..b636fbd 100644
--- a/rnaseq-A/config/config.yaml
+++ b/rnaseq-A/config/config.yaml
@@ -34,12 +34,12 @@ featurecounts: '../data/featurecounts.txt'

 # reference level for dds object
 level.column: 'group'
-level.ref: 'exp'
+level.ref: 'log'

 # contrast(s): indicate for each contrast the ctrst.group,
 # cntrst.test, ctrst.control
 contrasts:
-  mutant_vs_ctrl:
+  exp_vs_log:
     ctrst.group: 'group'
     ctrst.test: 'exp'
     ctrst.control: 'log'

commit e19d6c55e3ea915f7cc2f1070b1156adf3898baf
Author: Caroline Esnault <caroline.esnault@nih.gov>
Date:   Wed Jun 8 21:33:37 2022 -0400

    change level

diff --git a/rnaseq-A/config/config.yaml b/rnaseq-A/config/config.yaml
index c1aa100..669d7b8 100644
--- a/rnaseq-A/config/config.yaml
+++ b/rnaseq-A/config/config.yaml
@@ -34,7 +34,7 @@ featurecounts: '../data/featurecounts.txt'

 # reference level for dds object
 level.column: 'group'
-level.ref: 'log'
+level.ref: 'exp'

 # contrast(s): indicate for each contrast the ctrst.group,
 # cntrst.test, ctrst.control
```

Get history of parameters when using version control

```
git log -p -- rnaseq-A/config/config.yaml
```

Thanks for listening!