The entire code repository can be found at https://github.com/dgsaf/hpc-assignment-4.

Contents

List of Figures

List of Tables

1 Interpretation

The `count` process is presented in Listing 4.

```
48 process count {
49   input:
50   path(files) from files_ch.collect()
51
52   output:
53   file("results.csv") into counted_ch
54
55   container = ""
56
57   shell:
58   '''
59   echo "seed,ncores,nsrc" > results.csv
60   files=($(ls table*.csv))
61   for f in ${files[@]}; do
62     seed_cores=($(echo ${f} | tr '_.' ' ' | awk '{print $2 " " $3}'))
63     seed=${seed_cores[0]}
64     cores=${seed_cores[1]}
65     nsrc=$(echo "$(cat ${f} | wc -l) - 1" | bc -l)
66     echo "${seed},${cores},${nsrc}" >> results.csv
67   done
68   '''
69 }
```

Listing 1: The process `process count` in `nextflow/main.nf`.

i. `echo "seed,ncores,nsrc" > results.csv`

ii. `echo "${seed},${cores},${nsrc}" >> results.csv`

iii. `cat ${f} | wc -l`

iv. `$(<command>)` and `($(<command>))`

v. `$(ls table*.csv)`

vi. `echo ${f}`

vii. `tr '_.' ' '`

viii. `awk '{print $2 " " $3}'`

ix. `cat ${f}`

x. `wc -l`

xi. `echo "$(cat ${f} | wc -l)-1" | bc -l`

xii. remove this `cat ${f}`

2 Development

```
72 counted_ch.into{counted_for_ch; counted_xargs_ch}
```

Listing 2: The channel `counted_ch` is duplicated, with one for each of `process plot_for`, and `process plot_xargs`, in `nextflow/main.nf`.

2.1 Bash For Loop

```
75 process plot_for {
76   input:
77   path(table) from counted_for_ch
78
79   output:
80   file("*.png") into final_for_ch
81
82   cpus 4
83
84   shell:
85   '''
86   ncores_set=$(awk -F, '{if (NR != 1) {print $2}}' !{table} | sort | uniq)
87
88   for ncores in $ncores_set ; do
89     python !{projectDir}/plot_completeness.py \
90     --infile !{table} --outfile plot_for_${ncores}.png --cores $ncores
91   done
92   '''
93 }
```

Listing 3: The process `process plot_for` in `nextflow/main.nf`.

2.2 xargs Command

```
96 process plot_xargs {
97   input:
98   path(table) from counted_xargs_ch
99
100  output:
101  file("*.png") into final_xargs_ch
102
103  cpus 4
104
105  shell:
106  '''
107  ncores_set=$(awk -F, '{if (NR != 1) {print $2}}' !{table} | sort | uniq)
108
109  printf '%s ' $ncores_set | xargs -n1 -P4 -I ncores -d ' ' \
110  python !{projectDir}/plot_completeness.py \
111  --infile !{table} --outfile plot_xargs_ncores.png --cores ncores
112  '''
113 }
```

Listing 4: The process `process plot_for` in `nextflow/main.nf`.

3 Execution

3.1 SNR Plot

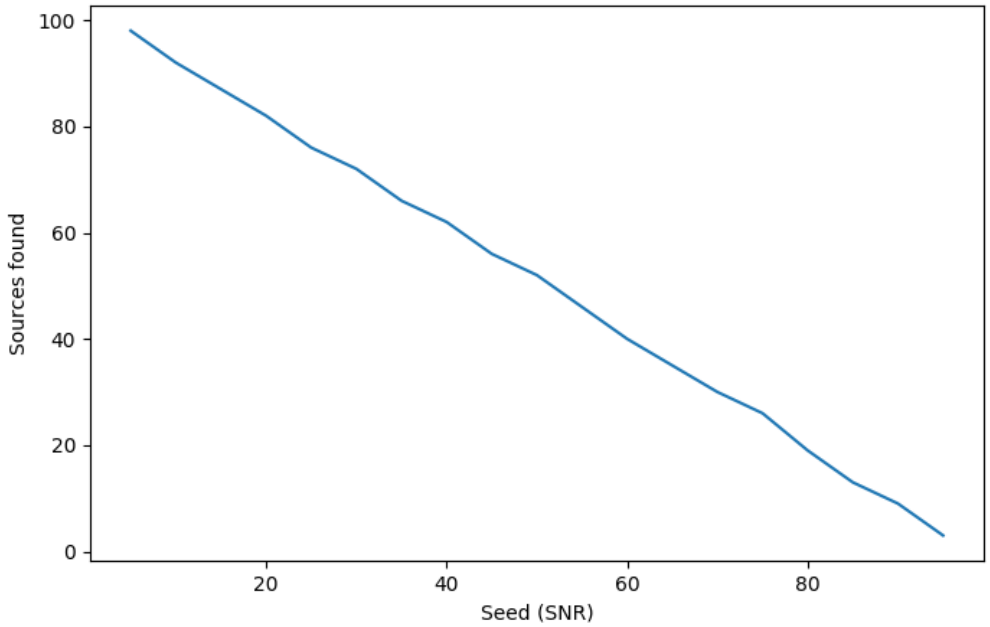

Figure 1: The Signal-to-Noise Ratio (SNR) is presented for a range of seed SNR values. This figure was produced by `process plot_for`*, for the case of* `cores = 1`*. No difference was observed between this plot and any of the other plots produced for any value of* `cores`*, nor whether if* `process plot_for` *or if* `process plot_xargs` *were used.*

3.2 Workflow DAG

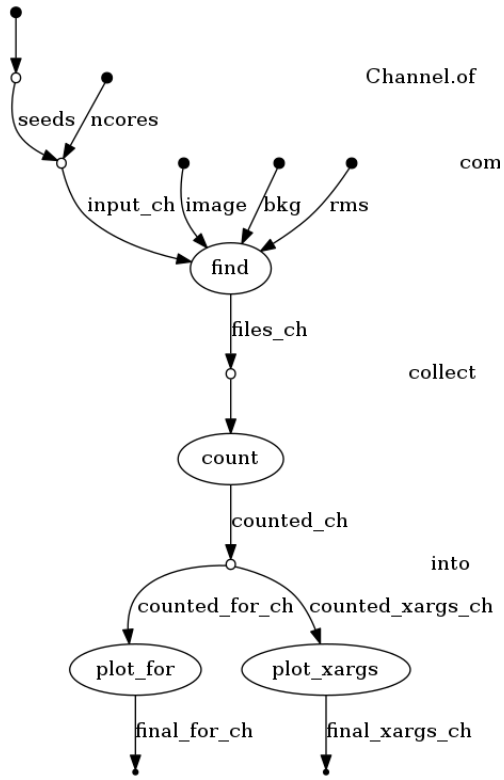

Figure 2: The Directed Acyclic Graph (DAG) of the workflow is presented. Note that the `combine` *operator (below* `Channel.of`*) appears to have been cropped out by the tool producing the DAG.*

4 Analysis