

flask配套的snakemake项目文档

目录:

1.WGS

2.WGBS

3.CGS

WGS流程配置

- 1.trim
- 2.mapping (genome & spike in)
- 3.deduping (genome)
- 4.cover (genome)
- 5.logging

可选软件

- 1.fastp, trim_galore
- 2.bowtie2, bwa mem
- 3.picard(java), python script, samtools
- 4.qualimap
- 5.python script

输入和输出格式

```
fastp_se:  
fq1="{fq_in_path}/{sample}.fq.gz"  
trim_fq1="{output_dir}/trim/fastp/{sample}_trimmed.fq.gz"  
json="{output_dir}/trim/fastp/{sample}_single.json"  
html="{output_dir}/trim/fastp/{sample}_single.html"
```

```
fastp_pe:  
fq1="{fq_in_path}/{sample}_1.fq.gz"  
fq2="{fq_in_path}/{sample}_2.fq.gz"  
trim_fq1="{output_dir}/trim/fastp/{sample}_1_val_1.fq.gz"  
trim_fq2="{output_dir}/trim/fastp/{sample}_2_val_2.fq.gz"  
json="{output_dir}/trim/fastp/{sample}_pair.json"  
html="{output_dir}/trim/fastp/{sample}_pair.html"
```

trim_galore_se:
fq1="{fq_in_path}/{sample}.fq.gz"
trim_fq="{output_dir}/trim/trim_galore/{sample}_trimmed.fq.gz"

trim_galore_pe:
fq1="{fq_in_path}/{sample}_1.fq.gz"
fq2="{fq_in_path}/{sample}_2.fq.gz"
trim_fq1="{output_dir}/trim/trim_galore/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/trim_galore/{sample}_2_val_2.fq.gz"

bowtie2_se:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_trimmed.fq.gz"
bam="{output_dir}/bam/bowtie2/{sample}_single.bam"

bowtie2_pe:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"
bam="{output_dir}/bam/bowtie2/{sample}_pair.bam"

sp_bowtie2_se:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_trimmed.fq.gz"
bam="{output_dir}/sp_bam/bowtie2/sp_{sample}_single.bam"

sp_bowtie2_pe:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"
bam="{output_dir}/sp_bam/bowtie2/sp_{sample}_pair.bam"

bwa_se:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_trimmed.fq.gz"
bam="{output_dir}/bam/bwa/{sample}_single.bam"

bwa_pe:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"
bam="{output_dir}/bam/bwa/{sample}_pair.bam"

sp_bwa_se:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_trimmed.fq.gz"
bam="{output_dir}/sp_bam/bwa/sp_{sample}_single.bam"

sp_bwa_pe:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"
bam="{output_dir}/sp_bam/bwa/sp_{sample}_pair.bam"

picard:

```
bam="{output_dir}/bam/{align_soft}/{sample}_{mode}.bam"  
de_bam="{output_dir}/de_bam/picard/{sample}_dedup.bam"
```

cover:

```
de_bam="{output_dir}/de_bam/{dup_soft}/{sample}_dedup.bam"  
cover="{output_dir}/cover/{sample}/genome_results.txt"
```

logging:

```
bam="{output_dir}/bam/{align_soft}/{sample}_{mode}.bam"  
dedup_bam="{output_dir}/de_bam/{dup_soft}/{sample}_dedup.bam"  
sp_bam="{output_dir}/sp_bam/{align_soft}/sp_{sample}_{mode}.bam"  
cover="{output_dir}/cover/{sample}/genome_results.txt"  
log="{output_dir}/final_log/{sample}.log"
```

WGBS流程配置

- 1.trim
- 2.mapping (genome & spike in)
- 3.deduping (genome)
- 4.call_modify (genome & spike in)
- 5.cover (genome)
- 6.logging

可选软件

- 1.fastp, trim_galore
- 2.bismark
- 3.picard(java), python script, samtools
- 4.bismark
- 5.qualimap
- 6.python script

输入和输出格式

```
fastp_se:  
fq1="{fq_in_path}/{sample}.fq.gz"  
trim_fq1="{output_dir}/trim/fastp/{sample}_trimmed.fq.gz"  
json="{output_dir}/trim/fastp/{sample}_single.json"  
html="{output_dir}/trim/fastp/{sample}_single.html"
```

```
fastp_pe:  
fq1="{fq_in_path}/{sample}_1.fq.gz"  
fq2="{fq_in_path}/{sample}_2.fq.gz"  
trim_fq1="{output_dir}/trim/fastp/{sample}_1_val_1.fq.gz"  
trim_fq2="{output_dir}/trim/fastp/{sample}_2_val_2.fq.gz"  
json="{output_dir}/trim/fastp/{sample}_pair.json"  
html="{output_dir}/trim/fastp/{sample}_pair.html"
```

```
trim_galore_se:  
fq1="{fq_in_path}/{sample}.fq.gz"  
trim_fq="{output_dir}/trim/trim_galore/{sample}_trimmed.fq.gz"
```

```
trim_galore_pe:  
fq1="{fq_in_path}/{sample}_1.fq.gz"  
fq2="{fq_in_path}/{sample}_2.fq.gz"  
trim_fq1="{output_dir}/trim/trim_galore/{sample}_1_val_1.fq.gz"  
trim_fq2="{output_dir}/trim/trim_galore/{sample}_2_val_2.fq.gz"
```

```
bismark_se:  
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_trimmed.fq.gz"  
bam="{output_dir}/bam/bismark/{sample}_single.bam"  
sort_bam="{output_dir}/bam/bismark/sort_{sample}_single.bam"
```

```
bismark_pe:  
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"  
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"  
bam="{output_dir}/bam/bismark/{sample}_pair.bam"  
sort_bam="{output_dir}/bam/bismark/sort_{sample}_pair.bam"
```

```
sp_bismark_se:  
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_trimmed.fq.gz"  
bam="{output_dir}/sp_bam/bismark/sp_{sample}_single.bam"  
sort_bam="{output_dir}/sp_bam/bismark/sort_sp_{sample}_single.bam"
```

```
sp_bismark_pe:  
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"  
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"  
bam="{output_dir}/sp_bam/bismark/sp_{sample}_pair.bam"  
sort_bam="{output_dir}/sp_bam/bismark/sort_sp_{sample}_pair.bam"
```

picard:

bam="{output_dir}/bam/{align_soft}/sort_{{sample}}_{{mode}}.bam"

de_bam="{output_dir}/de_bam/picard/{{sample}}_dedup.bam"

sort_bam="{output_dir}/de_bam/picard/sort_{{sample}}_dedup.bam"

bismark_call_methylation:

bam="{output_dir}/de_bam/{dup_soft}/{{sample}}_dedup.bam"

methylation_report="{output_dir}/de_bam/bismark/{{sample}}_dedup_splitting_report.txt"

sp_bismark_call_methylation:

bam="{output_dir}/sp_bam/{align_soft}/sp_{{sample}}_{{mode}}.bam"

methylation_report="{output_dir}/sp_bam/bismark/sp_{{sample}}_{{mode}}.CX_report.txt"

cover:

de_bam="{output_dir}/de_bam/{dup_soft}/sort_{{sample}}_dedup.bam"

cover="{output_dir}/cover/{{sample}}/genome_results.txt"

logging:

sort_bam="{output_dir}/bam/{align_soft}/sort_{{sample}}_{{mode}}.bam"

de_sort_bam="{output_dir}/de_bam/{dup_soft}/sort_{{sample}}_dedup.bam"

sp_sort_bam="{output_dir}/sp_bam/{align_soft}/sp_{{sample}}_{{mode}}.bam"

sp_bam="{output_dir}/sp_bam/{align_soft}/sp_{{sample}}_{{mode}}.bam"

mc_report="{output_dir}/de_bam/{call_modify_soft}/{{sample}}_dedup_splitting_report.txt"

mc_sp="{output_dir}/sp_bam/{call_modify_soft}/sp_{{sample}}_{{mode}}.CX_report.txt"

cover="{output_dir}/cover/{{sample}}/genome_results.txt"

log="{output_dir}/final_log/{{sample}}.log"

CGS&CS流程配置

- 1.trim
- 2.hairpin_cut
- 3.mapping (genome & spike in)
- 4.deduping (genome)
- 5.extract_mc_hmc (genome & spike in)
- 6.cover (genome)
- 7.logging

可选软件

- 1.fastp, trim_galore
- 2.new_hairpin_cut
- 3.bowtie2, bwa
- 4.picard(java)
- 5.CGS_extract_modify
- 6.qualimap
- 7.python script

输入和输出格式

fastp_pe:

```
fq1="{fq_in_path}/{sample}_1.fq.gz"
fq2="{fq_in_path}/{sample}_2.fq.gz"
trim_fq1="{output_dir}/trim/fastp/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/fastp/{sample}_2_val_2.fq.gz"
json="{output_dir}/trim/fastp/{sample}_pair.json" html="{output_dir}/trim/fastp/{sample}_pair.html"
```

trim_galore_pe:

```
fq1="{fq_in_path}/{sample}_1.fq.gz"
fq2="{fq_in_path}/{sample}_2.fq.gz"
trim_fq1="{output_dir}/trim/trim_galore/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/trim_galore/{sample}_2_val_2.fq.gz"
```



```
hairpin_cut:
trim_fq1="{output_dir}/trim/{trim_soft}/{sample}_1_val_1.fq.gz"
trim_fq2="{output_dir}/trim/{trim_soft}/{sample}_2_val_2.fq.gz"
cut_fq="{output_dir}/trim/CGS_haipin/{sample}.fq"
cut_fq1="{output_dir}/trim/CGS_haipin/{sample}_cut_f1.fq"
cut_fq2="{output_dir}/trim/CGS_haipin/{sample}_cut_f2.fq"
```

```
bowtie2_se:
cut_fq="{output_dir}/trim/{cut_soft}/{sample}.fq"
bam="{output_dir}/bam/bowtie2/{sample}.bam"
```

```
sp_bowtie2_se:
cut_fq="{output_dir}/trim/{cut_soft}/{sample}.fq"
bam="{output_dir}/sp_bam/bowtie2/sp_{sample}.bam"
```

```
bwa_se:
cut_fq="{output_dir}/trim/{cut_soft}/{sample}.fq"
bam="{output_dir}/bam/bwa/{sample}.bam"
```

```
sp_bwa_se:
cut_fq="{output_dir}/trim/{cut_soft}/{sample}.fq"
bam="{output_dir}/sp_bam/bwa/sp_{sample}.bam"
```

```
picard:
bam="{output_dir}/bam/{align_soft}/{sample}.bam"
de_bam="{output_dir}/de_bam/picard/{sample}_dedup.bam"
```

```
CGS_call_modify:
bam="{output_dir}/bam/{align_soft}/{sample}.bam"
cut_fq1="{output_dir}/trim/{cut_soft}/{sample}_cut_f1.fq"
hmc="{output_dir}/bam/CGS_call_modify/{sample}.bam.all.bed.hmc"
mc="{output_dir}/bam/CGS_call_modify/{sample}.bam.all.bed.mc"
```

```
de_CGS_call_modify:
de_bam="{output_dir}/de_bam/{dup_soft}/{sample}_dedup.bam"
cut_fq1="{output_dir}/trim/{cut_soft}/{sample}_cut_f1.fq"
hmc="{output_dir}/de_bam/CGS_call_modify/{sample}_dedup.bam.all.bed.hmc"
mc="{output_dir}/de_bam/CGS_call_modify/{sample}_dedup.bam.all.bed.mc"
```

```
sp_CGS_call_modify:
sp_bam="{output_dir}/sp_bam/{align_soft}/sp_{sample}.bam"
cut_fq1="{output_dir}/trim/{cut_soft}/{sample}_cut_f1.fq"
hmc="{output_dir}/sp_bam/CGS_call_modify/sp_{sample}.bam.all.bed.hmc"
mc="{output_dir}/sp_bam/CGS_call_modify/sp_{sample}.bam.all.bed.mc"
```

cover:

de_bam="{output_dir}/de_bam/{dup_soft}/{sort_{{sample}}}_dedup.bam"

cover="{output_dir}/cover/{{sample}}/genome_results.txt"

logging:

cut_fq="{output_dir}/trim/{cut_soft}/{sample}.fq"

bam="{output_dir}/bam/{align_soft}/{sample}.bam"

de_bam="{output_dir}/de_bam/{dup_soft}/{sample}_dedup.bam"

sp_bam="{output_dir}/sp_bam/{align_soft}/sp_{{sample}}.bam"

sp_hmc="{output_dir}/sp_bam/{call_modify_soft}/sp_{{sample}}.bam.all.bed.hmc"

sp_mc="{output_dir}/sp_bam/{call_modify_soft}/sp_{{sample}}.bam.all.bed.mc"

cover="{output_dir}/cover/{{sample}}/genome_results.txt"

hmc="{output_dir}/de_bam/{call_modify_soft}/{sample}_dedup.bam.all.bed.hmc"

mc="{output_dir}/de_bam/{call_modify_soft}/{sample}_dedup.bam.all.bed.mc"

log="{output_dir}/final_log/{{sample}}.log"

