SEPT 19TH 2019

Ontario Institute for Cancer Research

Nanopolish tutorial

Background

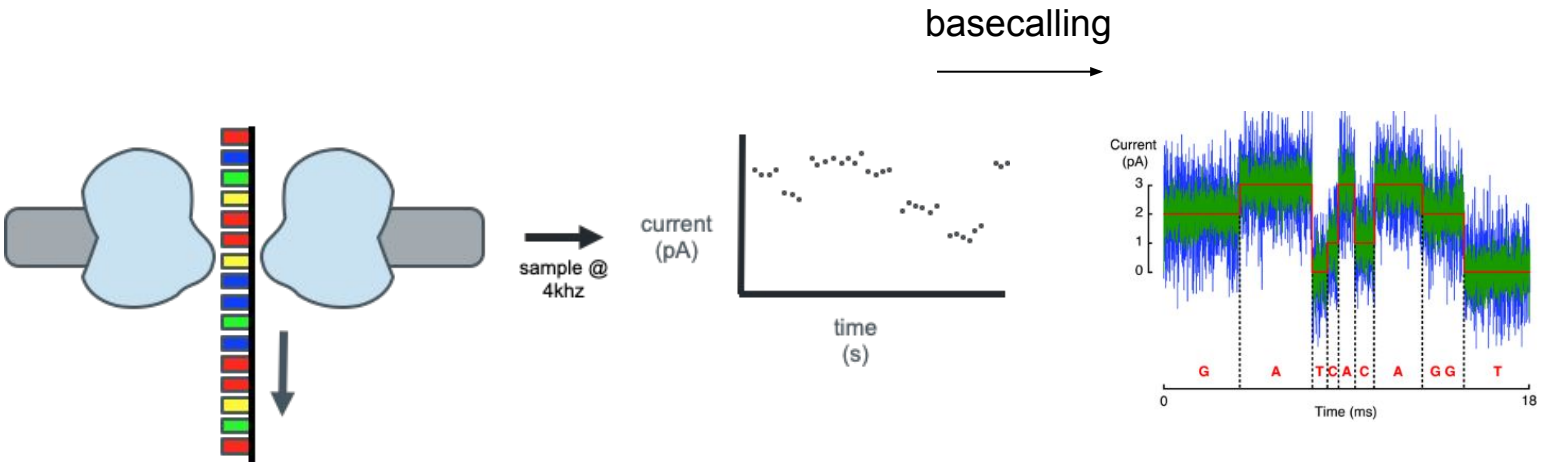
- Nanopolish is a tool for processing Oxford Nanopore signal-level data
- Originally developed for improving genome assembly
- Developed by Simpson lab at the Ontario Institute for Cancer Research (OICR)

Publications

- Loman, Nicholas J., Joshua Quick, and Jared T. Simpson. "*A complete bacterial genome assembled de novo using only nanopore sequencing data.*" Nature methods.
- Quick, Joshua, et al. "*Real-time, portable genome sequencing for Ebola surveillance.*" Nature.
- Simpson, Jared T., et al. "*Detecting DNA cytosine methylation using nanopore sequencing.*" Nature methods.

Overview

Determining sequence of DNA fragment by measuring differences in ionic signal in nanopore



Agenda

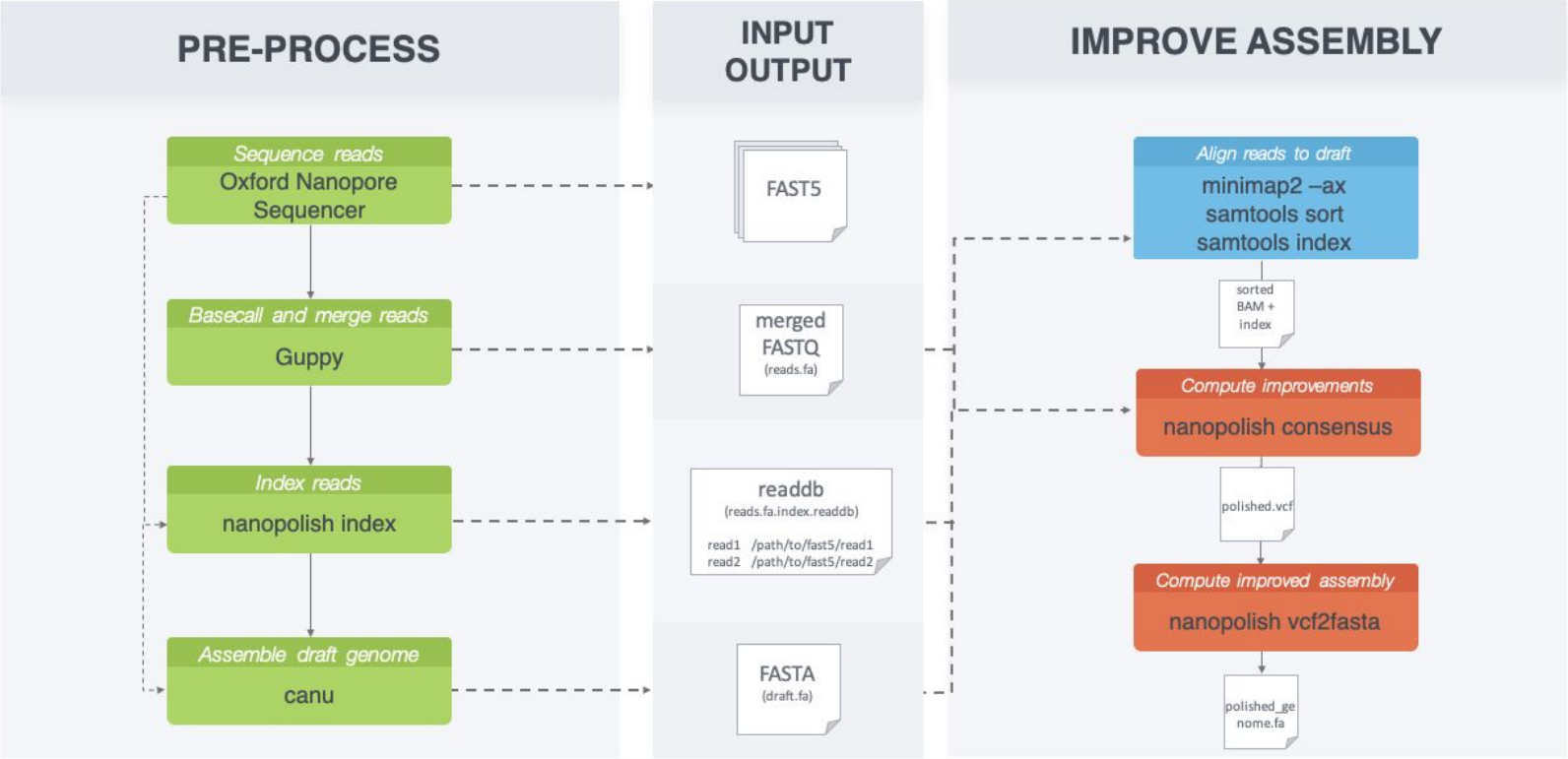
- Using nanopolish to polish an assembly
- Using nanopolish to call methylation
- Using nanopolish to estimate poly(A) tail length

Tutorial 1: Using nanopolish to polish an assembly

Motivation

- Availability of long reads is important in genome assembly as they can span repetitive elements
- Reduced accuracy is limiting
- Leveraging signal-level information helps

Overview


PRE-PROCESS
INPUT OUTPUT
IMPROVE ASSEMBLY
Sequence reads
Oxford Nanopore Sequencer
FAST5
Basecall and merge reads
Guppy
merged FASTQ
(reads.fa)
Index reads
nanopolish index
readdb
(reads.fa.index.readdb)
read1 /path/to/fast5/read1
read2 /path/to/fast5/read2
Assemble draft genome
canu
FASTA
(draft.fa)
Align reads to draft
minimap2 –ax
samtools sort
samtools index
sorted BAM + index
Compute improvements
nanopolish consensus
polished.vcf
Compute improved assembly
nanopolish vcf2fasta
polished_genome.fa

Dataset

- Sample : *E. coli* str. K-12 substr. MG1655
- Instrument : MinION sequencing R9.4 chemistry
- Basecalle

TIP
seven zeros

- Region: “tig00000001:200,000-202,000”

Set up

```
# get to tutorial directory
cd tutorial1

# look at contents
ls
```

```
draft.fa  fast5_files  reads.fasta  ref.fa  run1.sh
```

Step 0

```
# point nanopolish to proper signal-level data
nanopolish index -d fast5_files/ reads.fasta

# check indexing file readdb
head reads.fasta.index.readdb
```

```
0a238451-b9ed-446d-a152-badd074006c4  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch281_read4019_strand.fast5
0d624d4b-671f-40b8-9798-84f2ccc4d7fc  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch391_read2287_strand.fast5
0e56d03e-268c-42cc-a0f3-90d688115d07  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch433_read1144_strand.fast5
0eb7ac67-e215-4aa1-958a-7cd320d229a4  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch43_read3275_strand.fast5
10df3a9d-60a4-4cb5-ab3d-46483be7bb59  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch200_read6055_strand.fast5
1896c369-b7d0-4e98-aa23-830f7a9f001f  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch64_read6087_strand.fast5
1c935ec4-4503-44ea-be57-7201fc03eeb6  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch57_read1147_strand.fast5
1d5ccec9-296a-4afd-ad74-010ebd3732ec  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch391_read3493_strand.fast5
237f1f8e-b267-437b-b8d0-464a8f838fc0  fast5_files/odw_genlab4209_20161213_FN_MN16303_sequencing_run_sample_id_32395_ch302_read1093_strand.fast5
```

Step 0

```
# map reads to the draft genome
minimap2 -ax map-ont draft.fa reads.fasta | samtools sort -o
reads.sorted.bam -T reads.tmp

# index the bam file
samtools index reads.sorted.bam
```

Step 1

```
# identify changes needed to polish genome
nanopolish variants --consensus -o polished.vcf \
     -w "tig00000001:200,000-202,000" \
     -r reads.fasta \
     -b reads.sorted.bam \
     -g draft.fa
```

Step 1

```
# look at output

cat polished.vcf
```

```
##fileformat=VCFv4.2
##nanopolish_window=tig00000001:200000-202000
##INFO=<ID=TotalReads,Number=1,Type=Integer,Description="The number of event-space reads used to call the variant">
##INFO=<ID=SupportFraction,Number=1,Type=Float,Description="The fraction of event-space reads that support the variant">
##INFO=<ID=BaseCalledReadsWithVariant,Number=1,Type=Integer,Description="The number of base-space reads that support the variant">
##INFO=<ID=BaseCalledFraction,Number=1,Type=Float,Description="The fraction of base-space reads that support the variant">
##INFO=<ID=AlleleCount,Number=1,Type=Integer,Description="The inferred number of copies of the allele">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
#CHROM  POS     ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  sample
tig00000001     200061  .       T       TA      22.4    PASS    TotalReads=67;AlleleCount=1;SupportFraction=0.583757    GT      1
tig00000001     200180  .       C       CA      30.6    PASS    TotalReads=66;AlleleCount=1;SupportFraction=0.596279    GT      1
tig00000001     200484  .       G       GA      25.3    PASS    TotalReads=63;AlleleCount=1;SupportFraction=0.569916    GT      1
tig00000001     200672  .       T       TA      94.1    PASS    TotalReads=65;AlleleCount=1;SupportFraction=0.431464    GT      1
tig00000001     200776  .       C       CA      82.3    PASS    TotalReads=67;AlleleCount=1;SupportFraction=0.362655    GT      1
tig00000001     200796  .       T       TAA     117.1   PASS    TotalReads=66;AlleleCount=1;SupportFraction=0.268565    GT      1
tig00000001     201007  .       A       AG      31.0    PASS    TotalReads=65;AlleleCount=1;SupportFraction=0.604821    GT      1
tig00000001     201216  .       A       AT      85.8    PASS    TotalReads=67;AlleleCount=1;SupportFraction=0.457699    GT      1
tig00000001     201273  .       G       GT      25.2    PASS    TotalReads=66;AlleleCount=1;SupportFraction=0.588877    GT      1
tig00000001     201554  .       G       GC      49.9    PASS    TotalReads=76;AlleleCount=1;SupportFraction=0.632029    GT      1
tig00000001     201588  .       C       CG      125.4   PASS    TotalReads=75;AlleleCount=1;SupportFraction=0.387616    GT      1
tig00000001     201712  .       C       CA      21.1    PASS    TotalReads=74;AlleleCount=1;SupportFraction=0.57415     GT      1
```

Step 2

```
# make changes to genomic sequence
nanopolish vcf2fasta --skip-checks \
    -g draft.fa \
    polished.vcf > polished_genome.fa
```

Evaluate

```
# subset polished genome
samtools faidx polished_genome.fa
samtools faidx polished_genome.fa "tig00000001:200,000-202,000"
> polished.subset.fa
```

TIP
seven zeros

Evaluate

```
# subset draft genome
samtools faidx draft.fa
samtools faidx draft.fa "tig00000001:200,000-202,000" >
draft.subset.fa
```

TIP
seven zeros

Evaluate

```
# measure similarity to reference
dnadiff --prefix draft.dnadiff ref.fa draft.subset.fa
dnadiff --prefix polished.dnadiff ref.fa polished.subset.fa
```

Evaluate

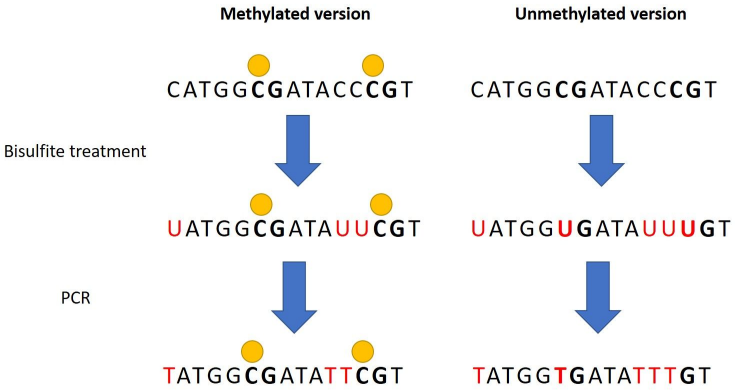
```
# comparison between draft genome and reference
# and compare between polished genome and reference
grep -A4 "1-to-1" *.dnadiff.report | grep AvgIden
```

```
Generating report file
draft.dnadiff.report-AvgIdentity                    99.38                99.38
polished.dnadiff.report-AvgIdentity                    99.93                99.93
```

Tutorial 2: Using nanopolish to call methylation

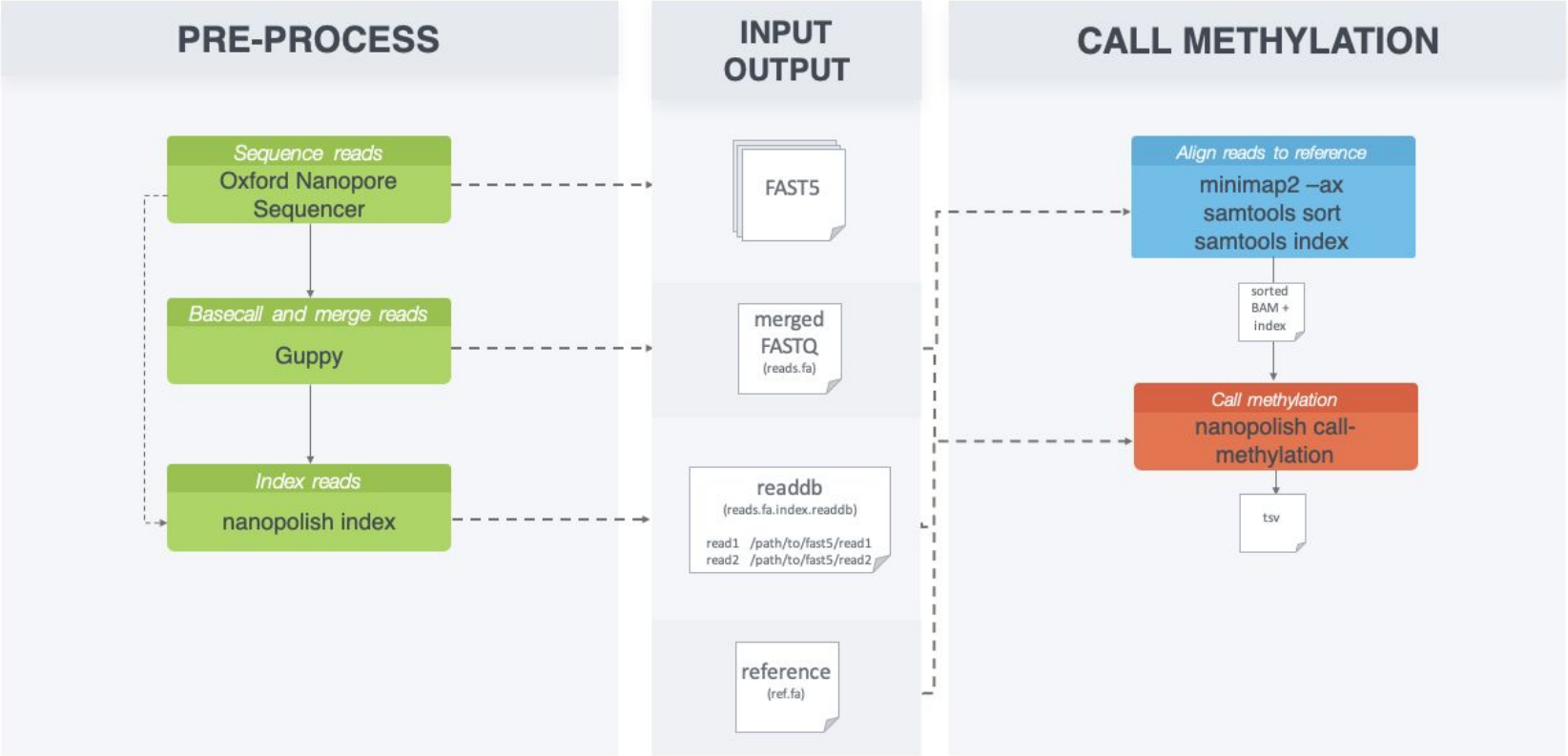
Motivation

- Bisulfite treatment is used to determine short patterns of methylation, however resolution is limited
- Alternatively, ONT supports detection of methylation sites directly and we can gain longer range information



Simpson, Jared T., et al. "*Detecting DNA cytosine methylation using nanopore sequencing.*" Nature Methods (2017).

Overview


PRE-PROCESS
INPUT OUTPUT
CALL METHYLATION
Sequence reads
Oxford Nanopore Sequencer
Basecall and merge reads
Guppy
Index reads
nanopolish index
FAST5
merged FASTQ
(reads.fa)
readdb
(reads.fa.index.readdb)
read1 /path/to/fast5/read1
read2 /path/to/fast5/read2
reference
(ref.fa)
Align reads to reference
minimap2 –ax
samtools sort
samtools index
sorted BAM + index
Call methylation
nanopolish call-methylation
tsv

Dataset

- Sample : Human cell line (NA12878)
- Basecaller : Guppy
- Region: chr20:9,000,000-10,000,000

Set up

```
# change to home directory
cd ..

# get to tutorial directory
cd tutorial2

# look at contents of directory
ls
```

```
bisulfite.ENCFF835NTC.example.tsv   compare_methylation.py  plot.R       reference.fasta
calculate_methylation_frequency.py  fast5_files             reads.fastq  run2.sh
```

Step 0

```
# point nanopolish to proper signal-level data
nanopolish index -d fast5_files/ reads.fastq

# map the reads to the reference, sort and index
minimap2 -ax map-ont reference.fasta reads.fastq | samtools
sort -T tmp -o output.sorted.bam

# index the bam file
samtools index output.sorted.bam
```

Step 1

```
# call methylation
nanopolish call-methylation -r reads.fastq \
    -b output.sorted.bam \
    -g reference.fasta \
    -w "chr20:9,000,000-10,000,000" > methylation_calls.tsv
# look at output
column -t methylation_calls.tsv | head
```

chromosome	strand	start	end	read_name	log_lik_ratio	log_lik_methylated	log_lik_unmethylated	num_calling_strands	num_motifs	sequence
chr20	+	9000033	9000033	877af79c-fa10-40fa-8919-e657eae6a68f	3.35	-86.28	-89.63	1	1	CACCCCGAGAA
chr20	+	9000046	9000046	877af79c-fa10-40fa-8919-e657eae6a68f	-2.97	-93.46	-90.49	1	1	TCAGGCGACCA
chr20	+	9000196	9000196	877af79c-fa10-40fa-8919-e657eae6a68f	-10.40	-103.85	-93.45	1	1	CTGGGCGAATG
chr20	+	9000297	9000305	877af79c-fa10-40fa-8919-e657eae6a68f	6.25	-126.47	-132.72	1	2	TCAAGCGAGCATGCGTACA
chr20	+	9000329	9000329	877af79c-fa10-40fa-8919-e657eae6a68f	6.48	-105.88	-112.35	1	1	CACTGCGGATG
chr20	+	9000386	9000386	877af79c-fa10-40fa-8919-e657eae6a68f	-2.77	-87.98	-85.20	1	1	CCCCACGGAAG
chr20	+	9000419	9000419	877af79c-fa10-40fa-8919-e657eae6a68f	-0.02	-88.67	-88.65	1	1	GACTCCGGAAA
chr20	+	9000683	9000683	877af79c-fa10-40fa-8919-e657eae6a68f	-6.80	-125.51	-118.71	1	1	GGATTCGCTGC
chr20	+	9000758	9000758	877af79c-fa10-40fa-8919-e657eae6a68f	-1.37	-82.30	-80.93	1	1	AATACCGAACT

Step 2

```
# call methylation frequency per genomic position
python calculate_methylation_frequency.py
methylation_calls.tsv > methylation_frequency.tsv

# look at output
column -t methylation_frequency.tsv | head
```

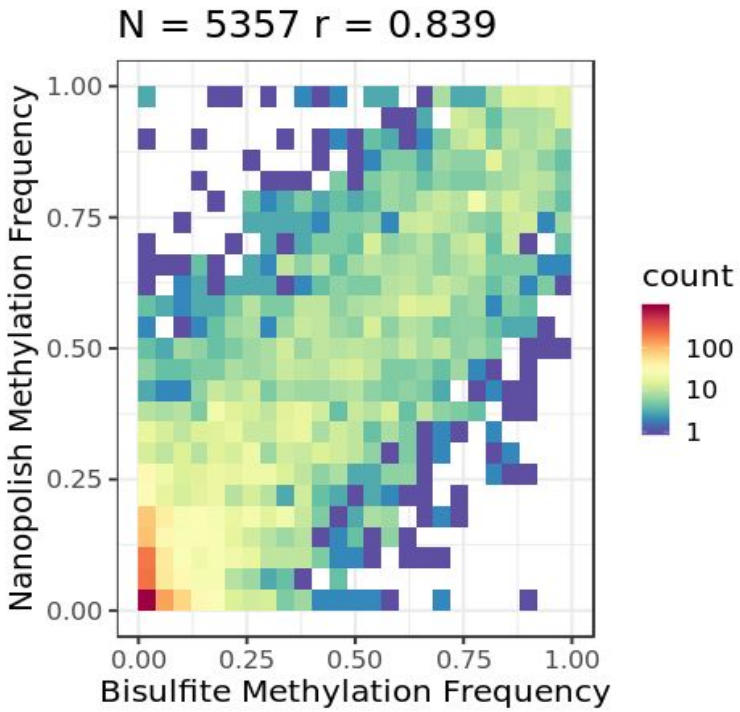
chromosome	start	end	num_motifs_in_group	called_sites	called_sites_methylated	methylated_frequency	group_sequence
chr20	9000033	9000033	1	9	5	0.556	CACCCCGAGAA
chr20	9000046	9000046	1	11	7	0.636	TCAGGCGACCA
chr20	9000196	9000196	1	14	1	0.071	CTGGGCGAATG
chr20	9000297	9000305	2	36	26	0.722	TCAAGCGAGCATGCGTACA
chr20	9000329	9000329	1	10	7	0.700	CACTGCGGATG
chr20	9000386	9000386	1	4	3	0.750	CCCCACGGAAG
chr20	9000419	9000419	1	6	2	0.333	GACTCCGGAAA
chr20	9000683	9000683	1	10	1	0.100	GGATTCGCTGC
chr20	9000758	9000758	1	9	8	0.889	AATACCGAACT

Evaluate

```
# compare to bisulfite calls
python compare_methylation.py \
    bisulfite.ENCFF835NTC.example.tsv \
    methylation_frequency.tsv > bisulfite_vs_nanopolish.tsv

Rscript plot.R
```

Evaluate


N = 5357 r = 0.839
Nanopolish Methylation Frequency
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
Bisulfite Methylation Frequency
count
100
10
1

Tutorial 3: Using nanopolish to estimate poly(A) tail length

Motivation

- Polyadenylation of RNA 3′ ends regulates RNA stability and translation efficiency by modulating RNA-protein binding and RNA structure
- Nanopore poly(A) tail length estimation advantages:
  - each RNA strand is read directly so errors due to priming of internal poly(A) segments cannot occur,
  - entire length of transcript can be read,
  - and no additional preparation steps

Workman, Rachael E., et al. "*Nanopore native RNA sequencing of a human poly (A) transcriptome.*" *BioRxiv* (2018).

Overview

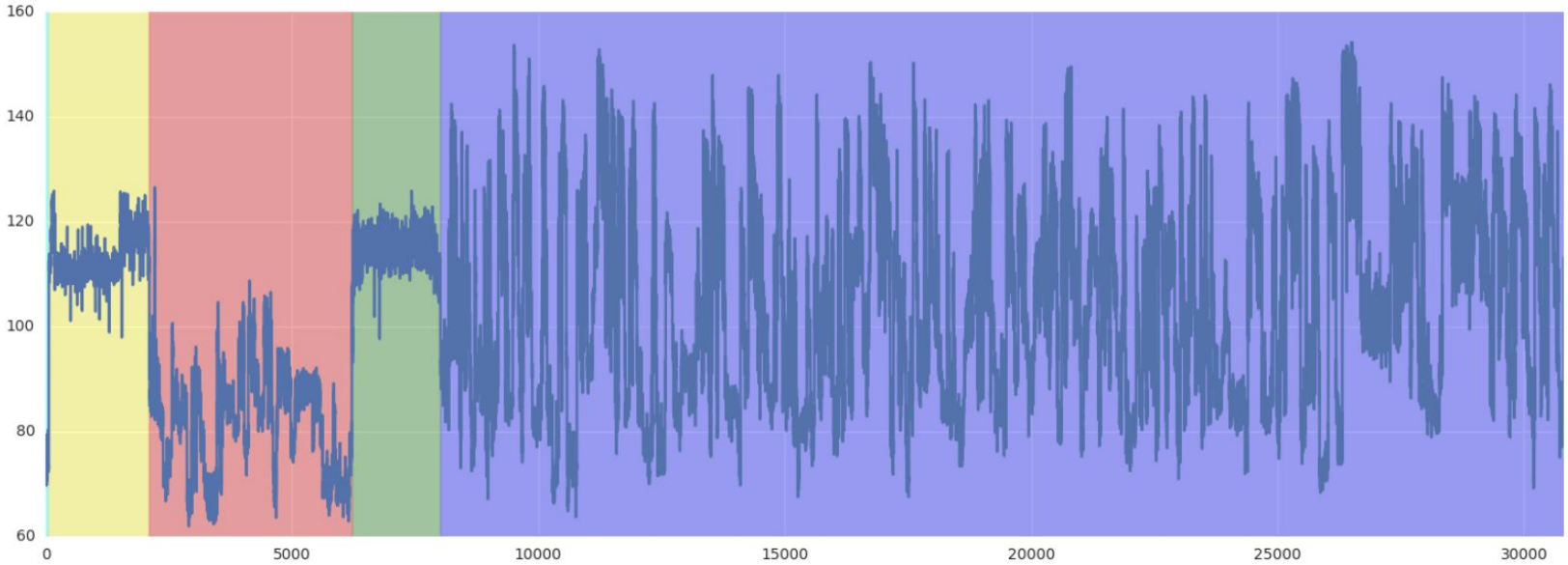
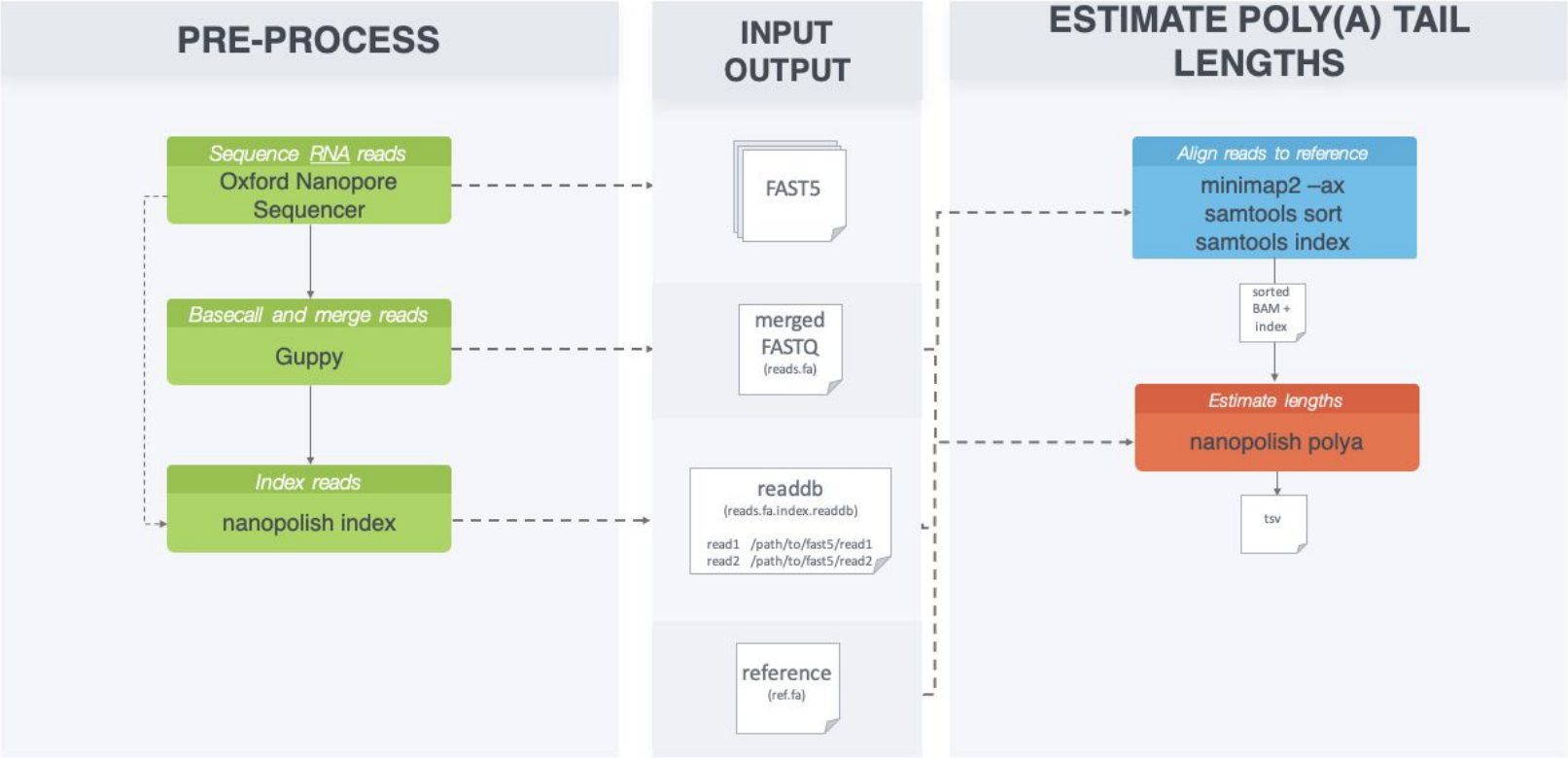


Figure. An example of a squiggle segmentation generated by the hidden markov model. Distinct regions, from left to right: start (cyan), leader (yellow), adapter (red), poly(A) tail (green), and transcript (purple). Two samples flagged as “cliffs” can be observed in the poly(A) tail.

Workman, Rachael E., et al. "*Nanopore native RNA sequencing of a human poly (A) transcriptome.*" *BioRxiv* (2018).

Overview


PRE-PROCESS
INPUT OUTPUT
ESTIMATE POLY(A) TAIL LENGTHS
Sequence RNA reads
Oxford Nanopore Sequencer
Basecall and merge reads
Guppy
Index reads
nanopolish index
FAST5
merged FASTQ
(reads.fa)
readdb
(reads.fa.index.readdb)
read1 /path/to/fast5/read1
read2 /path/to/fast5/read2
reference
(ref.fa)
Align reads to reference
minimap2 –ax
samtools sort
samtools index
sorted BAM + index
Estimate lengths
nanopolish polya
tsv

Dataset

- Sample : synthetic *S. cerevisiae*
- Note: controlled poly(A) tail length of 30 nucleotides

Set up

```
# change to home directory
cd ..

# get to tutorial directory
cd tutorial3

# look at contents
ls
```

```
30xpolyA.fastq          enolase_reference.fas.fai  plot_estimates.py
enolase_reference.fas   fast5                      run3.sh
```

Step 0

```
# point nanopolish to proper signal-level data
nanopolish index -d fast5/pass 30xpolyA.fastq

# map to the reference
minimap2 -ax map-ont enolase_reference.fas 30xpolyA.fastq
| samtools sort -o 30xpolyA.sorted.bam -T reads.tmp

# index the bam file
samtools index 30xpolyA.sorted.bam
```

Step 1

```
# call polya estimator
nanopolish polya -r 30xpolyA.fastq -t 8\
    -b 30xpolyA.sorted.bam \
    -g enolase_reference.fas | head -300 > polya_results.tsv
# check output
column -t polya_results.tsv | head
```

readname	contig	position	leader_start	adapter_start	polya_start	transcript_start	read_rate	polya_length	qc_tag
3f34d4b5-6016-4ee6-ac36-99831b68feb6	YHR174W	0	156.0	2832.0	6406.0	7330.0	130.96	35.13	PASS
7cdc452d-e244-4508-aaaf-e5f08de21bcf	YHR174W	0	19.0	1223.0	5811.0	7332.0	136.91	64.09	PASS
107584f0-2e71-4373-8ca2-35798c7294d9	YHR174W	0	48.0	5362.0	9468.0	10336.0	111.56	27.11	PASS
b46925e2-c8e5-4524-9cb9-5dd959c2ff11	YHR174W	0	26.0	6515.0	9468.0	10345.0	120.48	30.04	PASS
55ddd499-19f2-4367-aab7-363cce7a3798	YHR174W	0	53.0	2086.0	6216.0	7013.0	125.50	28.17	PASS
8c1ef536-3975-45a4-ae75-4c1080379c5b	YHR174W	0	160.0	3662.0	8840.0	9761.0	103.86	26.72	PASS
90417387-308e-4c45-9b22-969141411121	YHR174W	0	92.0	3989.0	11062.0	12142.0	115.85	36.50	PASS
2db51e94-0445-499b-a4d0-a10d41330798	YHR174W	0	716.0	2759.0	5962.0	6614.0	130.96	23.30	SUFFCLIP
39705c2f-a6bb-45f2-a22a-831971c1c4b3	YHR174W	0	115.0	1806.0	6747.0	7513.0	125.50	26.87	PASS

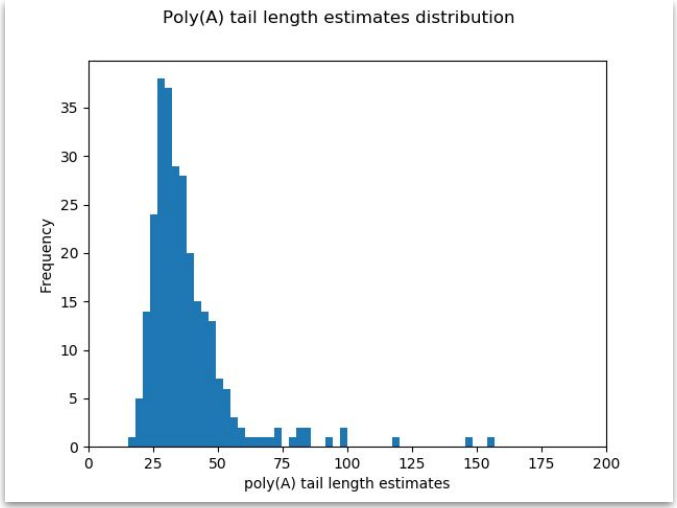
Step 2

```
# filter based on qc_tag
grep 'PASS' polya_results.tsv > polya_results.pass_only.tsv
```

Evaluate

```
# plot distribution of estimates
python plot_estimates.py polya_results.pass_only.tsv
```

```
Mean: 39.90135036496349. Median: 33.7699999999999996, num: 274
```



Funding provided by the
Government of Ontario.