Workman et al. RNA Data Analysis

Adnan M. Niazi & Maximillian Krause

5/20/2019

This document contains all the analyses done on RNA data that was generated for Workman et al paper. Knit this R markdown file after you have successfully run `drake::r_make()`.

Load the required libraries first:

```
pacman::p_load(pander, drake, knitr, ggpubr, here, tidyverse)
```

Now load the data:

```
loadd(rna_wo_data)
```

Here is a description of columns of `dna_kr_data`:

```
pander(col_names_df)
```

Columns	Description
dataset	Specifies the name of the conditions in Workman et al. data (N.B.: 60x and 60xb were combined into a single condition called 60x)
read_id	Read ID
tail_start_tf	tailfindr estimate of tail start
tail_end_tf	tailfindr estimate of tail end
samples_per_nt_tf	tailfindr estimation of read_specific transloation rate in units of samples per nucleotide
tail_length_tf	tailfindr tail length estimate
tail_start_np	Nanopolish tail start estimate
tail_end_np	Nanopolish tail end estimate (originally it is transcript_start in Nanopolish output)
read_rate_np	Nanopolish read rate
tail_length_np	Nanopolish estimation of tail length
qc_tag_np	Nanopolish QC Tag
samples_per_nt_np	Nanopolish estimation of read-specific translocation rate in units of samples per nucleotide calculated using the formula: 3012/read_rate
barcode	The expected tail length

Data summary

```
# define the function for computing standard error
std_err <- function(x) sd(x, na.rm = TRUE)/sqrt(length(x))

# summarize the data and display a table
summary_data <- rna_wo_data %>%
  group_by(dataset) %>%
  summarise(read_count = n(),
```

```
            mean_tf = mean(tail_length_tf, na.rm = TRUE),
            mean_np = mean(tail_length_np, na.rm = TRUE),

            median_tf = median(tail_length_tf, na.rm = TRUE),
            median_np = median(tail_length_np, na.rm = TRUE),

            std_dev_tf = sd(tail_length_tf, na.rm = TRUE),
            std_dev_np = sd(tail_length_np, na.rm = TRUE),

            std_err_tf = std_err(tail_length_tf),
            std_err_np = std_err(tail_length_np)
            )
summary_data %<>% mutate(cof_var_tf = std_dev_tf/mean_tf,
                         cof_var_np = std_dev_np/mean_np)
pander(summary_data)
```

Table 2: Table continues below

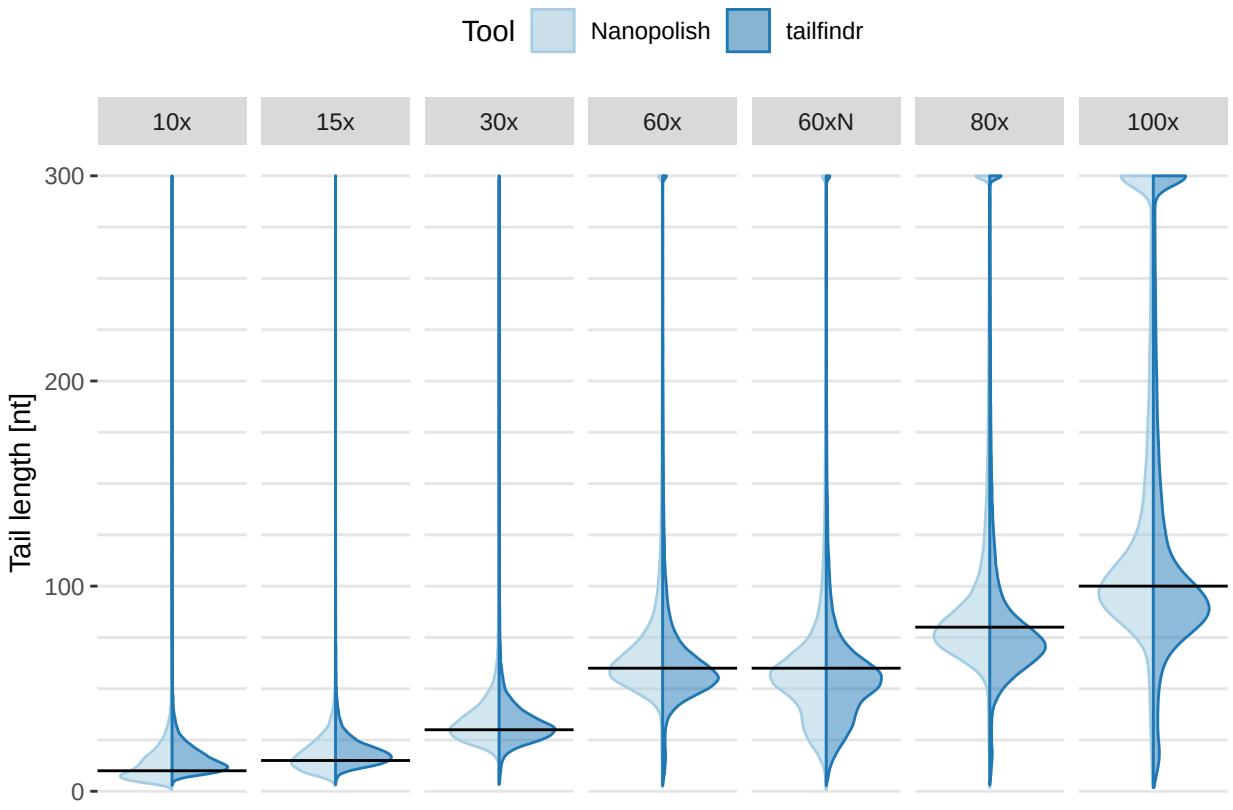
dataset	read_count	mean_tf	mean_np	median_tf	median_np	std_dev_tf
10x	27790	18.08	15.09	14.86	12.04	17.49
15x	23620	21.72	19.99	18.5	17.08	19.53
30x	17867	35.46	37.15	31.63	32.71	22.68
60x	100720	68.02	74.33	58.58	63.08	49.93
60xN	98221	60.57	63.97	53.84	56.03	50.8
80x	199121	89.65	102.5	73.92	81.83	70.46
100x	62563	152.3	173.3	97.94	108.1	157.2

std_dev_np	std_err_tf	std_err_np	cof_var_tf	cof_var_np
14.34	0.1049	0.08601	0.9675	0.9501
16.49	0.127	0.1073	0.899	0.8248
24.88	0.1697	0.1861	0.6397	0.6696
52.9	0.1573	0.1667	0.734	0.7116
54.87	0.1621	0.1751	0.8387	0.8578
80.42	0.1579	0.1802	0.786	0.7843
174.7	0.6285	0.6986	1.033	1.008

Comparing Nanopolish vs. tailfindr tail length estimates

```
# make long data
long_data_tf_np_tail_length <- rna_wo_data %>%
  select(dataset, barcode, tail_length_tf, tail_length_np) %>%
  gather(key = 'tool', value = 'tail_length', tail_length_tf, tail_length_np) %>%
  mutate(tail_length = ifelse(tail_length >= 300, 300, tail_length))

p <- ggplot(long_data_tf_np_tail_length, aes(x = dataset, y = tail_length,
                                             color = tool, fill = tool)) +
  geom_two_sided_flat_violin(position = position_nudge(x = 0, y = 0), alpha = .5) +
  facet_grid(~dataset, scales = 'free') +
  geom_hline(aes(yintercept = as.numeric(as.character(barcode))))
```


Tool
Nanopolish
tailfindr
10x
15x
30x
60x
60xN
80x
100x
Tail length [nt]
300
200
100
0