Relative Abundance Normalization Method Comparison

Nate Olson

2017-11-29

Comparison of relative abundance error rate for different normalization methods

```
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr

## Conflicts with tidy packages ----------------------------------------------

## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
library(ggridges)
```

```
norm_count_df <- readRDS("~/Desktop/to_file/mgtst_RDS/norm_count_df.RDS")
```

Mean variance relationship by normalization method - not sure if the difference is due to scaling or normalization method.

```
subsample_norm <- norm_count_df %>%
    # filter(mean_count != 0, biosample_id != "NTC") %>%
    # filter(mean_count != 0, var_count > 1e-10,
           # biosample_id == "E01JH0004") %>%
    filter(mean_count != 0, var_count > 1e-10, t_fctr == 0)

ggplot(subsample_norm) +
    geom_point(aes(x = mean_count, y = var_count, fill = biosample_id),
               shape = 21, alpha = 0.25) +
    geom_abline(aes(intercept = 0, slope = 1 )) +
    facet_wrap(~norm_method) +
    theme_bw() + scale_y_log10() + scale_x_log10() +
    labs(x = "Mean", y = "Variance") +
    coord_equal()
```

Calculating Error Rate

```
pa_summary_anno_df <- readRDS("~/Desktop/to_file/mgtst_RDS/pa_summary_anno_df.RDS")
theta_est <- readRDS("~/Desktop/to_file/mgtst_RDS/bootstrap_theta_estimates.rds")

pre_post_prop <- norm_count_df %>%
      ungroup() %>%
      filter(t_fctr %in% c(0,20)) %>%
      mutate(end_point = if_else(t_fctr == 0 , "post", "pre")) %>%
      select(-t_fctr, -var_count) %>%
      ## setting values to 0 when one or more of the PCR replicates are 0 for titration end-points
      spread(end_point,mean_count, fill = 0)
```

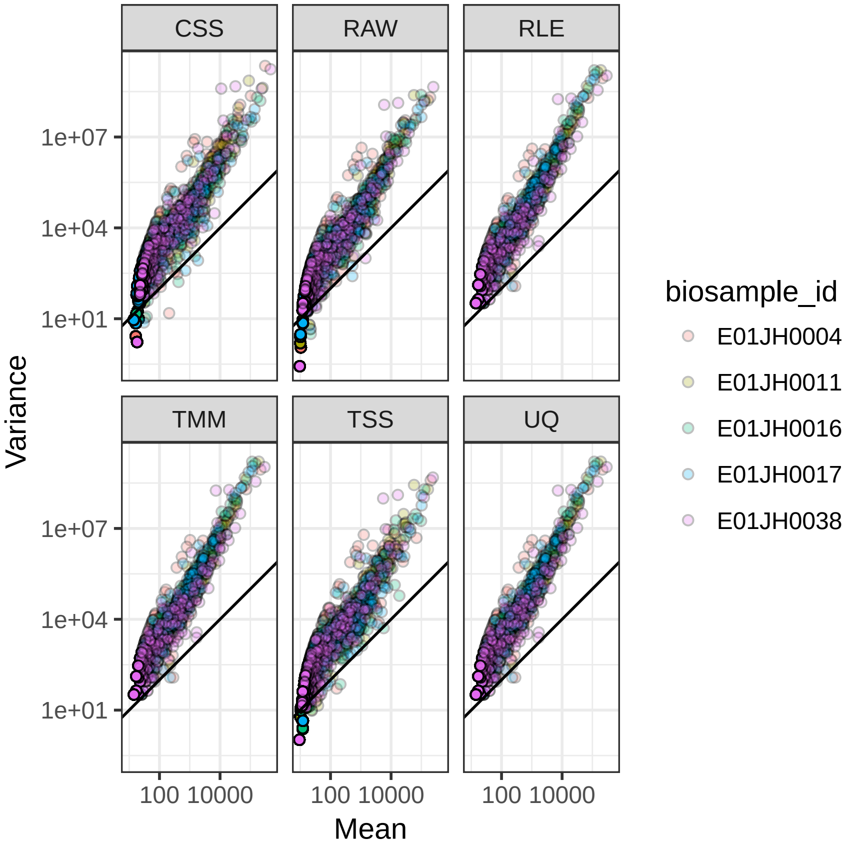


Figure 1: Comparison of relative abundance mean and variance relationship for PCR replicates across normalization methods. RLE - relative log expression, TMM - weighted trim mean of M-values, RAW - unnormalized, CSS - cumulative sum scaling, TSS - total sum scaling, UQ - upperquartile.

```r
prop_inferred <- theta_est %>%
      filter(pipe == "unclustered") %>%
      ungroup() %>%
      mutate(t_fctr = factor(t_fctr, levels = c(0:5, 10, 15, 20))) %>%
      select(biosample_id, theta_hat_mean, t_fctr) %>%
      right_join(norm_count_df) %>%
    right_join(pre_post_prop) %>%
      filter(t_fctr %in% c(1:5,10,15)) %>%
      ## Using inferred theta estimates to calculate expected values
      mutate(inferred_prop = post * theta_hat_mean + pre * (1 - theta_hat_mean))
```

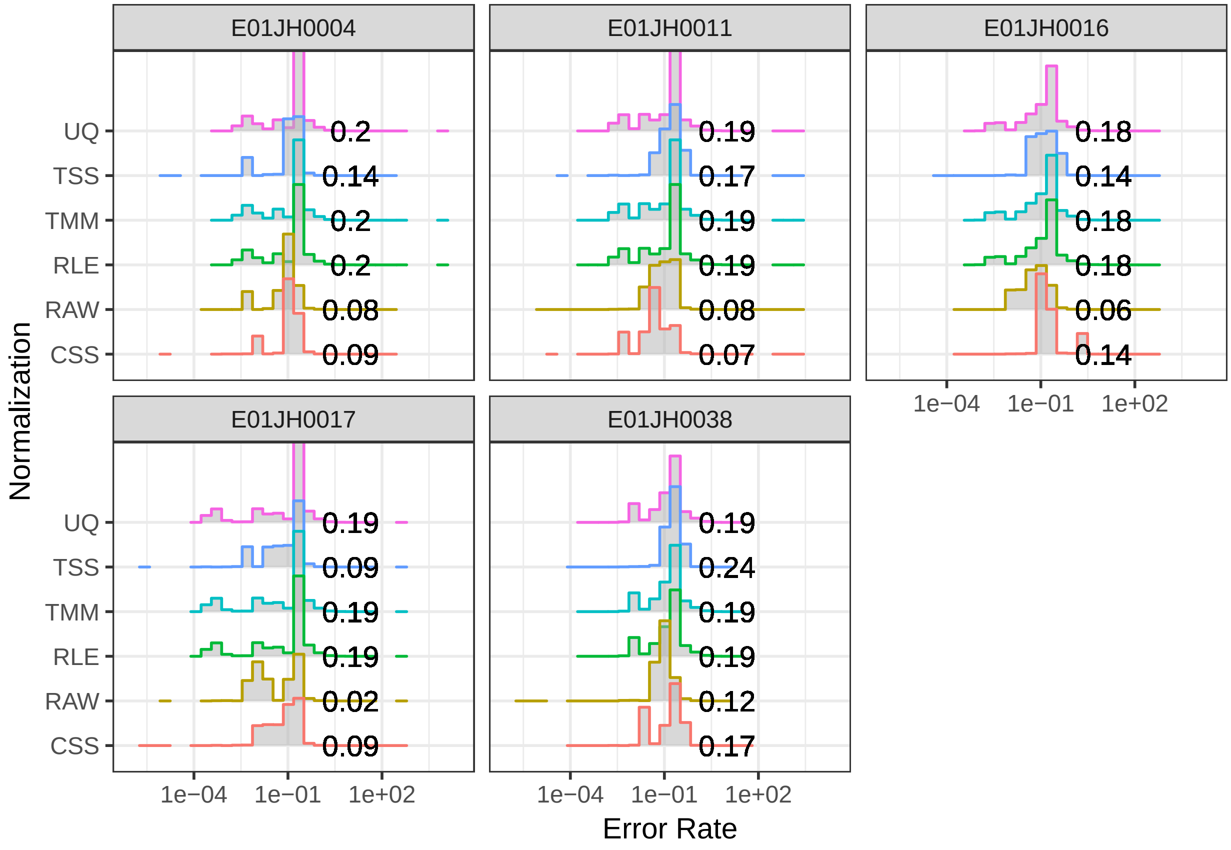
```
## Joining, by = c("biosample_id", "t_fctr")
```

```
## Warning: Column `t_fctr` joining factors with different levels, coercing to
## character vector
```

```
## Joining, by = c("biosample_id", "norm_method", "feature_id", "pipe")
```

```r
## Excluding mix and unmix specific features
## Only including features observed in all or none of the four pre- post- PCR replicates
## Features with relative abundance estimates less than 1e-7, these are features that we would not expe
pa_filter <- pa_summary_anno_df %>%
      filter(pa_specific == "unspecific") %>%
      select(biosample_id, pipe, feature_id, full_pre, T00, T20, pa_mixed) %>%
      filter(T00 %in% c(0,4), T20 %in% c(04))

# prop_inferred <- prop_inferred %>%
#       right_join(pa_filter) %>%
#       filter(nb_prop > 1e-7)


#### Error Rate Calculations
rel_abu_error <- prop_inferred %>%
      mutate(t_fctr = factor(t_fctr, levels = c(1:5, 10, 15))) %>%
      mutate(inferred_error = abs(mean_count - inferred_prop),
             inferred_error_rate = inferred_error/inferred_prop)
```

```r
rel_abu_ridge_df <- rel_abu_error %>%
    mutate(inferred_error_rate = if_else(inferred_error_rate < 1e-10,
                                         0, inferred_error_rate)) %>%
    filter(inferred_error_rate != 0 & mean_count > 1e-10)

rel_abu_med <- rel_abu_ridge_df %>%
      group_by(biosample_id, norm_method) %>%
      mutate(med_error = median( inferred_error_rate))

rel_abu_ridge_df %>%
      ggplot() +
      geom_density_ridges(aes(x =  inferred_error_rate, y = norm_method, color = norm_method),
                          alpha = 0.5, stat = "binline", bins = 30, draw_baseline = FALSE)  +
      geom_text(data = rel_abu_med,
                aes(x = 10, y = norm_method, label = round(med_error,2))) +
      facet_wrap(~biosample_id) + theme_bw() +
    scale_x_log10() +
      labs(x = "Error Rate", y = "Normalization", color = "Normalization") +
    theme(legend.position = "none")
```


E01JH0004
UQ
TSS
TMM
RLE
RAW
CSS
E01JH0011
E01JH0016
1e−04
1e−01
1e+02
E01JH0017
UQ
TSS
TMM
RLE
RAW
CSS
1e−04
1e−01
1e+02
E01JH0038
1e−04
1e−01
1e+02
Normalization
Error Rate