Error rate of bonito, guppy, rerio data

Heleen

6/12/2020

import data

```
blastrerio <- read.delim("~/lambda_reads/results/blastrerio", header=FALSE, quote="")
blastguppy <- read.delim("~/lambda_reads/results/blastguppy", header=FALSE, quote="")
blastbonito <- read.delim("~/lambda_reads/results/blastbonito", header=FALSE, quote="")
```

change colnames

```
colnames <- c('query', 'hit', 'hitid', 'percentidentity', 'coverage', 'evalue', 'bitscore')

colnames(blastbonito) <- colnames
colnames(blastguppy) <- colnames
colnames(blastrerio) <- colnames
```

get mean percentidentity and mean coverge

```
mrerio <- mean(blastrerio$percentidentity)
mguppy <- mean(blastguppy$percentidentity)
mbonito <- mean(blastbonito$percentidentity)

means <- c(mrerio, mguppy, mbonito)

crerio <- mean(blastrerio$coverage)
cguppy <- mean(blastguppy$coverage)
cbonito <- mean(blastbonito$coverage)

coverage <- c(crerio, cguppy, cbonito)
```

present nicely

```
methods <- c('rerio', 'guppy', 'bonito')

df <- data.frame(methods, means, coverage)

df
```

```
##   methods    means coverage
## 1   rerio 94.35797 98.30572
## 2   guppy 92.93504 98.24572
## 3  bonito 94.02647 98.22170
```

do t tests

```
t.test(blastrerio$percentidentity, blastbonito$percentidentity, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  blastrerio$percentidentity and blastbonito$percentidentity
## t = 13.857, df = 71293, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.2846130 0.3783952
## sample estimates:
## mean of x mean of y
##  94.35797  94.02647
```

```
t.test(blastguppy$percentidentity, blastbonito$percentidentity, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  blastguppy$percentidentity and blastbonito$percentidentity
## t = -45.114, df = 71320, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.138839 -1.044004
## sample estimates:
## mean of x mean of y
##  92.93504  94.02647
```

Make function for finding mode

```
Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]}
```

get mode, median, mean for percent identity

```
cat(paste('Mode guppy:\t', Mode(blastguppy$percentidentity),
          "\nMode bonito:\t", Mode(blastbonito$percentidentity),
          "\nMode rerio:\t", Mode(blastrerio$percentidentity)))
```

```
## Mode guppy:   94.83
## Mode bonito:  96.26
## Mode rerio:   96.89
```

```
cat(paste("\nmedian guppy:\t", median(blastguppy$percentidentity),
          "\nmedian bonito:\t", median(blastbonito$percentidentity),
          "\nmedian rerio:\t", median(blastrerio$percentidentity)))
```

```
##
## median guppy:     93.95
## median bonito:    95.07
## median rerio:     95.4
```

```
cat(paste("\nmean guppy:\t", mean(blastguppy$percentidentity),
          "\nmean bonito:\t", mean(blastbonito$percentidentity),
          "\nmean rerio:\t", mean(blastrerio$percentidentity)))
```

```
##
## mean guppy:   92.935043693276
## mean bonito:  94.0264651722497
## mean rerio:   94.3579692679364
```

add column to dataframe to combine the dataframes

```
blastbonito$basecaller <- rep('bonito', nrow(blastbonito)) #for dataframe 'bonito' create new column wi
blastguppy$basecaller <- rep('guppy', nrow(blastguppy))
blastrerio$basecaller <- rep('rerio', nrow(blastrerio))
```

change percent identity into error rate

```
blastbonito$error <- 100-blastbonito$percentidentity
blastguppy$error <- 100-blastguppy$percentidentity
blastrerio$error <- 100-blastrerio$percentidentity

combineddf <- as.data.frame(rbind(blastrerio, blastguppy, blastbonito))
```

Get summary statitics of error rate and do anova

```
basecallers <- c('guppy', 'bonito', 'rerio')

myErrorStats <- function(){
  sapply(basecallers, function(x){
    cat(paste(
      '\nMode', x, ':\t', Mode(combineddf$error[combineddf$basecaller == x])),
      '\nMean', x, ':\t', mean(combineddf$error[combineddf$basecaller == x]),
      '\nMedian', x, ':\t', median(combineddf$error[combineddf$basecaller == x]),
      '\n')})
}

getSD <- function() {
  lapply(basecallers, function(x){
    cat(paste(x,
              'sd: ',sd(combineddf$error[combineddf$basecaller == x]),
              '\n'))})}


getRow <- function(){
  cat("number of reads in range 1500-4000:\n")
  sapply(basecallers, function(name) {
    cat(paste(name, ':\t',
              nrow(combineddf[combineddf$basecaller == name, ]),
              '\n', sep = ''))})
  cat('\n')}


getSummary <- function()
  sapply(basecallers, function(x)
    {summary(combineddf$error[combineddf$basecaller == x])})


getRow()
```

```
## number of reads in range 1500-4000:
## guppy:   35589
## bonito:  35733
## rerio:   35598
```

```
cat('summary statistics\n')
```

```
## summary statistics
```

```
getSummary()
```

```
##             guppy    bonito     rerio
## Min.     0.000000  0.000000  1.200000
## 1st Qu.  4.880000  3.800000  3.530000
## Median   6.050000  4.930000  4.600000
## Mean     7.064956  5.973535  5.642031
## 3rd Qu.  8.170000  7.020000  6.660000
## Max.    25.230000 26.900000 26.920000
```

```
myErrorStats()
```

```
##
## Mode guppy :  5.17
## Mean guppy :  7.064956
## Median guppy :    6.05
##
## Mode bonito :     3.73999999999999
## Mean bonito :     5.973535
## Median bonito :   4.93
##
## Mode rerio :  3.11
## Mean rerio :  5.642031
## Median rerio :    4.6
```

```
## $guppy
## NULL
##
## $bonito
## NULL
##
## $rerio
## NULL
```

```
print(anova <- aov(error ~ basecaller, data = combineddf))
```

```
## Call:
##    aov(formula = error ~ basecaller, data = combineddf)
##
## Terms:
##                 basecaller Residuals
## Sum of Squares     39466.4 1098122.0
## Deg. of Freedom          2    106917
##
## Residual standard error: 3.204807
## Estimated effects may be unbalanced
```

```
cat('\n')
```

```
summary(anova)
```

```
##                 Df  Sum Sq Mean Sq F value Pr(>F)
## basecaller       2   39466   19733    1921 <2e-16 ***
## Residuals   106917 1098122      10
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
cat('\n')
```

```
TukeyHSD(anova)
```

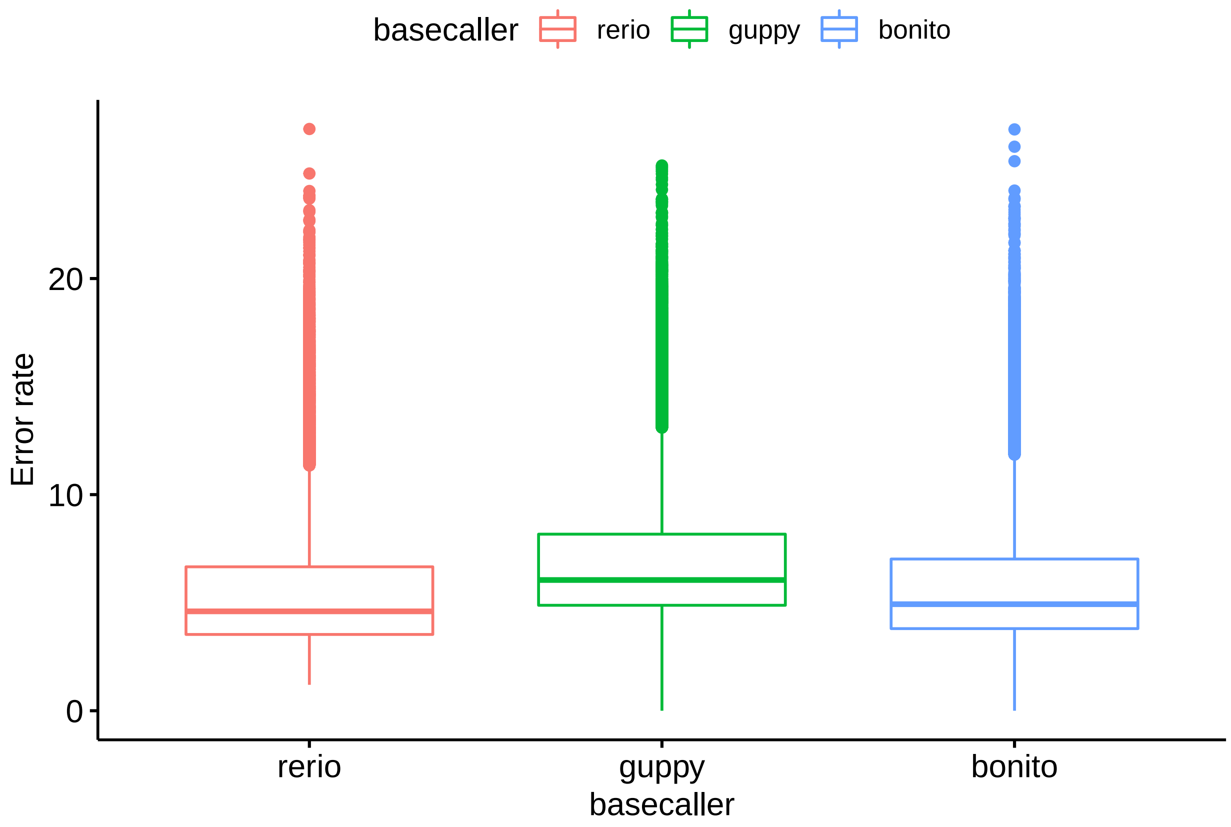
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = error ~ basecaller, data = combineddf)
##
## $basecaller
##                    diff        lwr        upr p adj
## guppy-bonito  1.0914215  1.0351714  1.1476716     0
## rerio-bonito -0.3315041 -0.3877506 -0.2752576     0
## rerio-guppy  -1.4229256 -1.4792289 -1.3666223     0
```

Make boxplot

```
library(ggpubr)
```

```
## Loading required package: ggplot2
```

```r
ggboxplot(combineddf,
          x='basecaller',
          y='error',
          color = 'basecaller',
          ylab = 'Error rate',
          )
```



Make plot

```r
library(ggplot2)
library(plyr)
```

```
##
## Attaching package: 'plyr'

## The following object is masked from 'package:ggpubr':
##
##     mutate
```

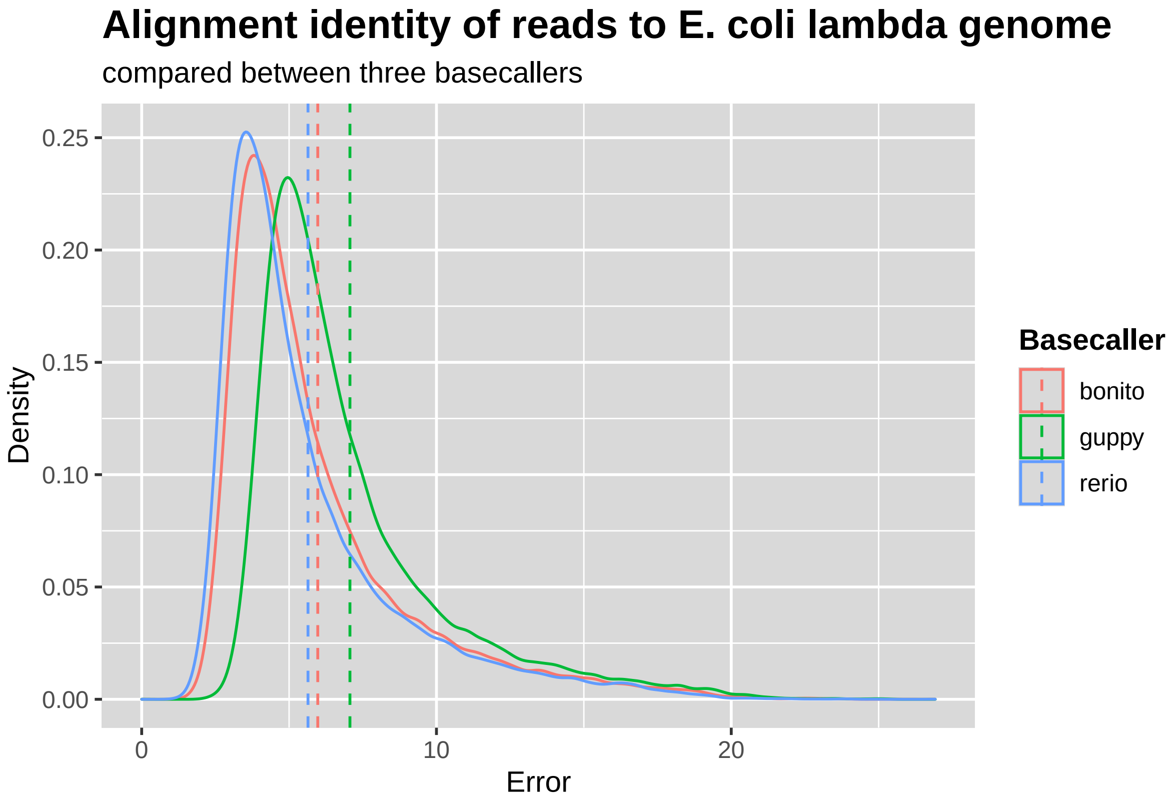
```r
mu <- ddply(combineddf, 'basecaller', summarise, grp.mean=mean(percentidentity))
head(mu)
```

```
##   basecaller grp.mean
## 1     bonito 94.02647
## 2      guppy 92.93504
## 3      rerio 94.35797
```

```r
plot <- ggplot(
  data = combineddf,
  aes(x = percentidentity,
      color=basecaller)) +

  geom_density() +

  geom_vline(
    data=mu,
    aes(xintercept=grp.mean,
        color=basecaller),
    linetype='dashed') +

  labs(
    title = 'Alignment identity of reads to E. coli lambda genome',
    subtitle = 'compared between three basecallers',
    x = 'Percent Idenity',
    y = 'Density') +
  # legend title
  scale_color_discrete(name = 'Basecaller') +
  # edit title and legend title appearance
  theme(plot.title = element_text(size = 15, face = 'bold'),
        legend.title = element_text(face = 'bold'),
        legend.key = element_rect(fill = 'grey85'),
        panel.background = element_rect(fill = 'grey85'),
        plot.background = element_rect(fill = 'white'),
        legend.background = element_rect(fill = 'white'),
        plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), 'cm'))

ggsave('percentplot',
       plot = last_plot(),
       device = 'png',
       width = 20,
       height = 15,
       units = 'cm')

##PLOT 2
mu <- ddply(combineddf, 'basecaller', summarise, grp.mean=mean(error))
head(mu)
```

```
##   basecaller grp.mean
## 1     bonito 5.973535
## 2      guppy 7.064956
## 3      rerio 5.642031
```

```r
plot <- ggplot(
  data = combineddf,
  aes(x = error,
      color=basecaller)) +

  geom_density() +

  geom_vline(
    data=mu,
    aes(xintercept=grp.mean,
        color=basecaller),
    linetype='dashed') +

  labs(
    title = 'Alignment identity of reads to E. coli lambda genome',
    subtitle = 'compared between three basecallers',
    x = 'Error',
    y = 'Density') +
  # legend title
  scale_color_discrete(name = 'Basecaller') +
  # edit title and legend title appearance
  theme(plot.title = element_text(size = 15, face = 'bold'),
        legend.title = element_text(face = 'bold'),
        legend.key = element_rect(fill = 'grey85'),
        panel.background = element_rect(fill = 'grey85'),
        plot.background = element_rect(fill = 'white'),
        legend.background = element_rect(fill = 'white'),
        plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), 'cm'))

#ggsave('errorplot',
#       plot = last_plot(),
#       device = 'png',
#       width = 20,
#       height = 15,
#       units = 'cm')

plot
```


Alignment identity of reads to E. coli lambda genome
compared between three basecallers
Density
0.25
0.20
0.15
0.10
0.05
0.00
0
10
20
Error
Basecaller
bonito
guppy
rerio