[Re] Local alignment statistics - Compile Table 1 from Raw data

Nathan Brouwer

11/18/2019

Analysis of all data

Preliminaries

```
#install.packages("extRemes")
library(extRemes)
```

```
## Loading required package: Lmoments
```

```
## Loading required package: distillery
```

```
##
## Attaching package: 'extRemes'
```

```
## The following objects are masked from 'package:stats':
##
##     qqnorm, qqplot
```

Load saved data

```
load("~/1_R/git/blaststats/full_experiment.RData")
```

Output from simulations

All data objects produced by each for loop

random.scores.191 random.scores.245 random.scores.314 random.scores.403 random.scores.518 random.scores.665 random.scores.854 random.scores.1097 random.scores.1408 random.scores.1808 random.scores.2322 random.scores.2981

Get mean length of alignments

```
l.191 <- mean(random.scores.191$length.i)
l.245 <- mean(random.scores.245$length.i)
l.314 <- mean(random.scores.314$length.i)
l.403 <- mean(random.scores.403$length.i)
l.518 <- mean(random.scores.518$length.i)
l.665 <- mean(random.scores.665$length.i)
l.854 <- mean(random.scores.854$length.i)
l.1097 <- mean(random.scores.1097$length.i)
l.1408 <- mean(random.scores.1408$length.i)
l.1808 <- mean(random.scores.1808$length.i)
```

```
l.2322 <- mean(random.scores.2322$length.i)
l.2981 <- mean(random.scores.2981$length.i)
```

Fit Gumbel extreme value distribution model to each set of output

For each subexperiment (Each row of the table) a Gumbel EVD model is fit to the data. This provides an estimate of mu and lambda.

```
fit.gumbel.191 <- fevd(random.scores.191$score.i,
                    type = "Gumbel",
                    method = "MLE")


fit.gumbel.245 <- fevd(random.scores.245$score.i,
                    type = "Gumbel",
                    method = "MLE")


fit.gumbel.314 <- fevd(random.scores.314$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.403 <- fevd(random.scores.403$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.518 <- fevd(random.scores.518$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.665 <- fevd(random.scores.665$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.854 <- fevd(random.scores.854$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.1097 <- fevd(random.scores.1097$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.1408 <- fevd(random.scores.1408$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.1808 <- fevd(random.scores.1808$score.i,
                    type = "Gumbel",
                    method = "MLE")

fit.gumbel.2322 <- fevd(random.scores.2322$score.i,
                    type = "Gumbel",
                    method = "MLE")
```

```
fit.gumbel.2981 <- fevd(random.scores.2981$score.i,
                   type = "Gumbel",
                   method = "MLE")
```

Get summary out from each model

THe "location" parameter is what the modeling function calls "mu". The "Scale parameter" relates to lambda; lambda = 1/scale. These can both be extracted from the output of the model.

All of the output from all of the models

fit.gumbel.191 fit.gumbel.245 fit.gumbel.314 fit.gumbel.403 fit.gumbel.518 fit.gumbel.665 fit.gumbel.854 fit.gumbel.1097 fit.gumbel.1408 fit.gumbel.1808 fit.gumbel.2322 fit.gumbel.2981

```
# m = n = 191
## run summary on model
summary.gumbel.191 <- summary(fit.gumbel.191)
```

```
##
## fevd(x = random.scores.191$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  27619.06
##
##
##  Estimated parameters:
## location    scale
## 26.12708  3.25730
##
##  Standard Error Estimates:
##   location      scale
## 0.03427803 0.02548092
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0011749833 0.0002720403
## scale    0.0002720403 0.0006492772
##
##  AIC = 55242.12
##
##  BIC = 55256.54
```

```
## extract location paramter
loc.param.191 <- summary.gumbel.191$par[1]

## extract and reformat scale paramter
scale.param.191  <- 1/summary.gumbel.191$par[2]

summary.gumbel.245 <- summary(fit.gumbel.245)
```

```
##
## fevd(x = random.scores.245$score.i, type = "Gumbel", method = "MLE")
##
```

```
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  27790.29
##
##
##  Estimated parameters:
##  location     scale
## 27.983646  3.327913
##
##  Standard Error Estimates:
##   location      scale
## 0.03504067 0.02591524
##
##  Estimated parameter covariance matrix.
##             location        scale
## location 0.001227849 0.0002842990
## scale    0.000284299 0.0006715996
##
##  AIC = 55584.58
##
##  BIC = 55599
```

```
loc.param.245 <- summary.gumbel.245$par[1]
scale.param.245  <- 1/summary.gumbel.245$par[2]

summary.gumbel.314 <- summary(fit.gumbel.314)
```

```
##
## fevd(x = random.scores.314$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  27970.23
##
##
##  Estimated parameters:
##  location     scale
## 29.672855  3.383495
##
##  Standard Error Estimates:
##   location      scale
## 0.03562192 0.02640708
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0012689211 0.0002941956
## scale    0.0002941956 0.0006973337
##
##  AIC = 55944.46
##
##  BIC = 55958.88
```

```r
loc.param.314 <- summary.gumbel.314$par[1]
scale.param.314  <- 1/summary.gumbel.314$par[2]

summary.gumbel.403 <- summary(fit.gumbel.403)
```

```
##
## fevd(x = random.scores.403$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28096.64
##
##
##  Estimated parameters:
##  location     scale
## 31.541461  3.436437
##
##  Standard Error Estimates:
##   location      scale
## 0.03619412 0.02674956
##
##  Estimated parameter covariance matrix.
##             location        scale
## location 0.001310014 0.0003039400
## scale    0.000303940 0.0007155388
##
##  AIC = 56197.28
##
##  BIC = 56211.7
```

```r
loc.param.403 <- summary.gumbel.403$par[1]
scale.param.403  <- 1/summary.gumbel.403$par[2]

summary.gumbel.518 <- summary(fit.gumbel.518)
```

```
##
## fevd(x = random.scores.518$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28189.41
##
##
##  Estimated parameters:
## location    scale
## 33.35689  3.45273
##
##  Standard Error Estimates:
##   location      scale
## 0.03634377 0.02700014
##
##  Estimated parameter covariance matrix.
```

```
##              location        scale
## location 0.0013208693 0.0003063471
## scale    0.0003063471 0.0007290075
##
##  AIC = 56382.81
##
##  BIC = 56397.23
```

```
loc.param.518 <- summary.gumbel.518$par[1]
scale.param.518  <- 1/summary.gumbel.518$par[2]

summary.gumbel.665 <- summary(fit.gumbel.665)
```

```
##
## fevd(x = random.scores.665$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28249.17
##
##
##  Estimated parameters:
##  location     scale
## 35.156877  3.479327
##
##  Standard Error Estimates:
##   location      scale
## 0.03662701 0.02712364
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0013415379 0.0003104033
## scale    0.0003104033 0.0007356919
##
##  AIC = 56502.33
##
##  BIC = 56516.75
```

```
loc.param.665 <- summary.gumbel.665$par[1]
scale.param.665  <- 1/summary.gumbel.665$par[2]

summary.gumbel.854 <- summary(fit.gumbel.854)
```

```
##
## fevd(x = random.scores.854$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28299.64
##
##
##  Estimated parameters:
##  location     scale
```

```
## 36.931682  3.495234
##
##  Standard Error Estimates:
##   location      scale
## 0.03679377 0.02727380
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0013537816 0.0003134883
## scale    0.0003134883 0.0007438600
##
##  AIC = 56603.28
##
##  BIC = 56617.71
```

```
loc.param.854 <- summary.gumbel.854$par[1]
scale.param.854  <- 1/summary.gumbel.854$par[2]

summary.gumbel.1097 <- summary(fit.gumbel.1097)
```

```
##
## fevd(x = random.scores.1097$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28304.86
##
##
##  Estimated parameters:
##  location     scale
## 38.840037  3.503233
##
##  Standard Error Estimates:
##   location      scale
## 0.03688696 0.02729005
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0013606478 0.0003151775
## scale    0.0003151775 0.0007447466
##
##  AIC = 56613.71
##
##  BIC = 56628.14
```

```
loc.param.1097  <- summary.gumbel.1097$par[1]
scale.param.1097  <- 1/summary.gumbel.1097$par[2]

summary.gumbel.1408 <- summary(fit.gumbel.1408)
```

```
##
## fevd(x = random.scores.1408$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
```

```
##
##
##  Negative Log-Likelihood Value:  28370.88
##
##
##  Estimated parameters:
##  location     scale
## 40.585266  3.526944
##
##  Standard Error Estimates:
##   location      scale
## 0.03714250 0.02750978
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0013795652 0.0003203816
## scale    0.0003203816 0.0007567878
##
##  AIC = 56745.76
##
##  BIC = 56760.18
```

```
loc.param.1408 <- summary.gumbel.1408$par[1]
scale.param.1408  <- 1/summary.gumbel.1408$par[2]

summary.gumbel.1808 <- summary(fit.gumbel.1808)
```

```
##
## fevd(x = random.scores.1808$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28398.71
##
##
##  Estimated parameters:
##  location     scale
## 42.378338  3.533792
##
##  Standard Error Estimates:
##   location      scale
## 0.03721067 0.02758840
##
##  Estimated parameter covariance matrix.
##              location       scale
## location 0.001384634 0.000321574
## scale    0.000321574 0.000761120
##
##  AIC = 56801.41
##
##  BIC = 56815.83
```

```
loc.param.1808 <- summary.gumbel.1808$par[1]
scale.param.1808  <- 1/summary.gumbel.1808$par[2]
```

```
summary.gumbel.2322 <- summary(fit.gumbel.2322)
```

```
##
## fevd(x = random.scores.2322$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28550.33
##
##
##  Estimated parameters:
##  location     scale
## 44.148781  3.596093
##
##  Standard Error Estimates:
##   location      scale
## 0.03787410 0.02797747
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0014344472 0.0003325191
## scale    0.0003325191 0.0007827387
##
##  AIC = 57104.66
##
##  BIC = 57119.08
```

```
loc.param.2322 <- summary.gumbel.2322$par[1]
scale.param.2322  <- 1/summary.gumbel.2322$par[2]

summary.gumbel.2981<- summary(fit.gumbel.2981)
```

```
##
## fevd(x = random.scores.2981$score.i, type = "Gumbel", method = "MLE")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  28498.35
##
##
##  Estimated parameters:
##  location     scale
## 45.997669  3.581599
##
##  Standard Error Estimates:
##   location      scale
## 0.03772511 0.02781623
##
##  Estimated parameter covariance matrix.
##              location        scale
## location 0.0014231842 0.0003295947
## scale    0.0003295947 0.0007737428
```

```
##
##  AIC = 57000.71
##
##  BIC = 57015.13
```

```
loc.param.2981 <- summary.gumbel.2981$par[1]
scale.param.2981  <- 1/summary.gumbel.2981$par[2]
```

Compile table from simulations

```
# Sequences lengths used in the simulation
mn <- c(191,245,314,403,518,
        665,
        854,
        1097,
        1408,1808,2322,
        2981
        )

# mean lengths of ALIGNMETNS
lenght.l <- c(l.191 ,l.245 ,l.314 ,l.403 ,l.518 ,
              l.665,
              l.854 ,l.1097 ,
              l.1408 ,l.1808 ,l.2322 ,
              l.2981
              )

# mu (location) parameters from all of the models fit above
location.u <- c(loc.param.191,
                loc.param.245,
                loc.param.314,
                loc.param.403,
                loc.param.518,
                loc.param.665,
                loc.param.854,
                loc.param.1097,
                loc.param.1408,
                loc.param.1808,
                loc.param.2322,
                loc.param.2981
                )

# scale parameters (lambda) from all of the models
scale.lambda <- c(scale.param.191,
                  scale.param.245,
                  scale.param.314,
                  scale.param.403,
                  scale.param.518,
                  scale.param.665,
                  scale.param.854,
                  scale.param.1097,
                  scale.param.1408,
                  scale.param.1808,
```

```
                  scale.param.2322,
                  scale.param.2981
                  )


# comile table
table1.NLB <- data.frame(mn = mn,
                   l = lenght.l,
           ln.mn = log(mn*mn),
           mu = location.u,
           lambda =scale.lambda
           )
```

Calculate K

Given the equations in Altschul and Gish we can calculate K directly from our estimates of lambda and mu

```
table1.NLB$K <-  exp((location.u*scale.lambda) - log(mn*mn))
```

Look at table

```
head(table1.NLB)
```

```
##    mn       l    ln.mn       mu    lambda          K
## 1 191 22.0003 10.50455 26.12708 0.3070028 0.08345393
## 2 245 24.6035 11.00252 27.98365 0.3004887 0.07473939
## 3 314 27.3328 11.49879 29.67285 0.2955524 0.06529082
## 4 403 30.1876 11.99787 31.54146 0.2909991 0.05964555
## 5 518 33.2113 12.49995 33.35689 0.2896259 0.05848812
## 6 665 36.3463 12.99957 35.15688 0.2874119 0.05529516
```

Save table

```
write.csv(table1.NLB, file = "table1NLB.csv")
```

Round off columns

```
# table1.NLB$ln.mn  <- round(table1.NLB$ln.mn, 3)
# table1.NLB$lenght.l     <- round(table1.NLB$l, 2)
# table1.NLB$mu     <- round(table1.NLB$mu, 2)
# table1.NLB$lambda <- round(table1.NLB$lambda, 3)
# table1.NLB$K      <- round(table1.NLB$K, 3)
```