Analysis of eSMC Data for a Single Chromosome

Lukas Metzger

2024-12-03

Contents

Introduction

This analysis processes eSMC (ecological Sequentially Markovian Coalescent) data for a single chromosome using the `eSMC2` package in R. The goal is to estimate the effective population size and selfing rate over time.

Load Libraries

```
knitr::opts_chunk$set(echo = TRUE)
library(eSMC2)
```

Set Parameters

Here we can adjust the input for eSMC

```
# Parameters
M <- 32          # Number of haplotypes
NC <- 1          # Number of cores
mu <- 7e-9       # Mutation rate per site per generation
r <- 3.5e-8      # Recombination rate per site per generation
chrom_name <- 1  # Name of the chromosome
```

Load Data

```r
# File paths
data_path <- "/Users/luki/Downloads/SMC_files/"
input_file <- "arabis_allo_all_ref_nemo_woRUS_chr1_equalsize_region.txt"

# Load data
Os_input <- Get_real_data(
  path = data_path,
  M = M,
  input_file,
  delim = "\t"
)
```

```
## Rows: 33200 Columns: 4
## -- Column specification -----------------------------------------------------
## Delimiter: "\t"
## chr (1): X4
## dbl (3): X1, X2, X3
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
print(Os_input[1:34, 1:7])
```

```
##         [,1]    [,2]    [,3]    [,4]    [,5]    [,6]    [,7]
##  [1,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [2,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [3,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [4,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [5,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [6,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [7,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [8,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
##  [9,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
## [10,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
## [11,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
## [12,] "A"     "T"     "C"     "C"     "A"     "T"     "A"
## [13,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [14,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [15,] "C"     "C"     "T"     "T"     "G"     "A"     "G"
## [16,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [17,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [18,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [19,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [20,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [21,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [22,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [23,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [24,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [25,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [26,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
## [27,] "C"     "C"     "T"     "T"     "G"     "A"     "G"
## [28,] "C"     "T"     "T"     "T"     "G"     "A"     "G"
```

```
## [29,] "C"       "C"       "T"       "T"       "G"       "A"       "G"
## [30,] "C"       "T"       "T"       "T"       "G"       "A"       "G"
## [31,] "C"       "C"       "T"       "T"       "G"       "A"       "G"
## [32,] "C"       "T"       "T"       "T"       "G"       "A"       "G"
## [33,] "6390663" "18"      "250"     "49"      "16"      "57"      "53"
## [34,] "6500180" "6500198" "6500450" "6500499" "6500515" "6500572" "6500625"
```

```r
# Remove specific columns from the data
Os_trim <- Os_input[-c(1,3,5,7,9,10,11,12,13,15,16,17,19,21,22,23,25,26,27,28,29,30,31,32), ]

dim(Os_trim)
```

```
## [1]    10 33200
```

```r
print(Os_trim[1:10, 1:7])
```

```
##        [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
##  [1,] "A"       "T"       "C"       "C"       "A"       "T"       "A"
##  [2,] "A"       "T"       "C"       "C"       "A"       "T"       "A"
##  [3,] "A"       "T"       "C"       "C"       "A"       "T"       "A"
##  [4,] "A"       "T"       "C"       "C"       "A"       "T"       "A"
##  [5,] "C"       "T"       "T"       "T"       "G"       "A"       "G"
##  [6,] "C"       "T"       "T"       "T"       "G"       "A"       "G"
##  [7,] "C"       "T"       "T"       "T"       "G"       "A"       "G"
##  [8,] "C"       "T"       "T"       "T"       "G"       "A"       "G"
##  [9,] "6390663" "18"      "250"     "49"      "16"      "57"      "53"
## [10,] "6500180" "6500198" "6500450" "6500499" "6500515" "6500572" "6500625"
```

Run teSMC

```r
results <- teSMC(
  n = 40,
  rho = r / mu,
  Os_trim,
  model = "One transition",
  estimate = "SF",
  Constant_Pop = FALSE,
  BoxP = c(3, 3),
  Boxs = list(list(c(0.5, 0.99), c(0, 0.5))),
  NC = NC,
  Big_Window = TRUE
)

load("/Users/luki/Downloads/SMC_files/all_allo_equalsize_MSMC_window_scaling_chr1.RData")
results <- results_MSMCwindow_equalsize_AllAllo_chr1
str(results)
```

```
## List of 9
##  $ LH   : num -2150947
##  $ Tc   : num [1:40] 0 0.00176 0.00236 0.00315 0.00422 ...
##  $ Xi   : num [1:40] 0.414 0.414 0.388 0.388 0.338 ...
##  $ mu   : num 0.00246
##  $ L    : num 11496837
##  $ beta : num 1
```

```
##  $ sigma: num [1:40] 0.862 0.862 0.862 0.862 0.862 ...
##  $ rho  : num 0.0195
##  $ N    : num [1:40, 1:40] 2.87e+05 7.30e-03 1.04e-02 1.33e-02 1.79e-02 ...
```

Extract Ke Components from teSMC output

```r
# Extract time points
time_points <- results$Tc

# Extract selfing rate estimates
selfing_rates <- results$sigma

# Extract population size estimates
population_sizes <- results$Xi

# Calculate effective population size (Ne)
Ne <- mean(results$mu / mu)

# Time in generations ago
time_generations <- time_points * Ne

# Effective population size over time
Ne_t <- population_sizes * 0.5 * Ne
```

Plot effective population size and selfing rate over time
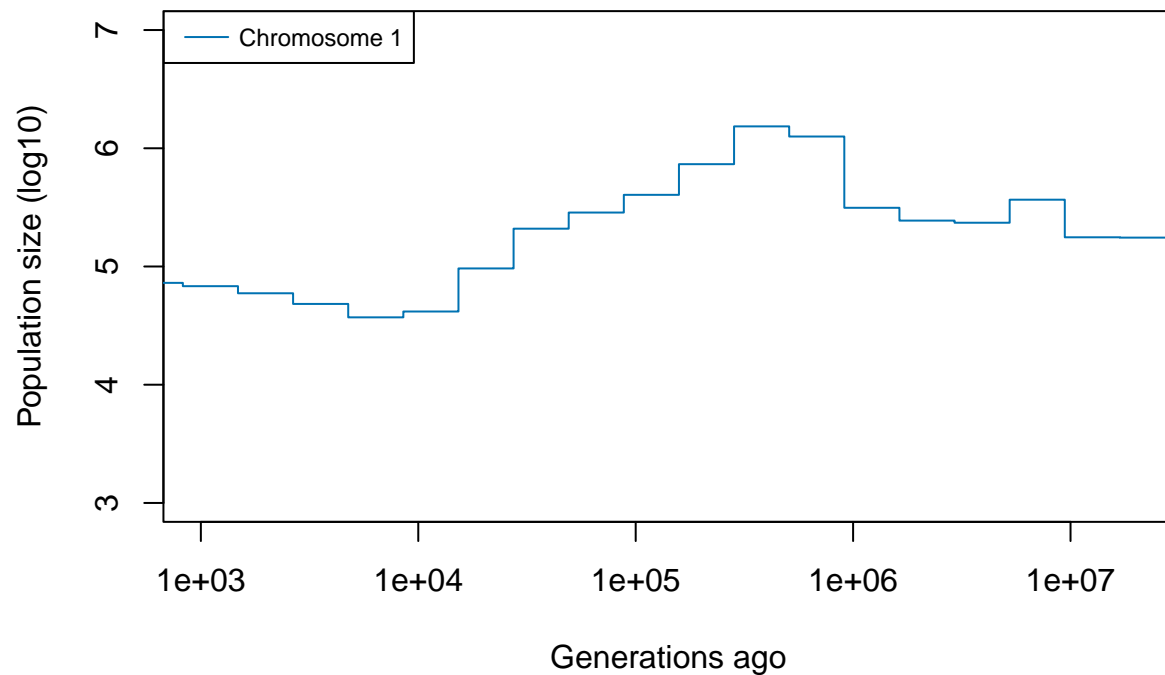
```r
# Define the plot area
plot(
  x = c(1000, 2 * 10^7),    # x-axis range
  y = c(1, 1),              # Placeholder y-values
  log = "x",                # Logarithmic x-axis
  ylim = c(3, 7),           # y-axis range
  type = "n",               # No plotting yet
  xlab = "Generations ago",
  ylab = "Population size (log10)",
  main = "Population Size Over Time for Chromosome 1"
)

# Add step lines to the plot
lines(
  x = (time_generations),
  y = log10(Ne_t),
  type = "s",
  col = "#0072B2"  # Example color
)

# Add a legend
legend(
  "topleft",
  legend = "Chromosome 1",
```

```
  col = "#0072B2",
  lty = 1,
  cex = 0.75,
  x.intersp = 0.5,
  y.intersp = 0.8
)
```

Population Size Over Time for Chromosome 1


```
# Define the plot area for selfing rate
plot(
  x = c(1000, 2 * 10^7),    # x-axis range
  y = c(0, 1),              # y-axis range for selfing rate (adjust if needed)
  log = "x",                # Logarithmic x-axis
  type = "n",               # No plotting yet
  xlab = "Generations ago",
  ylab = "Selfing rate",
  main = "Selfing Rate Over Time for Chromosome 1"
)

# Add step lines for selfing rates
lines(
  x = time_generations,
  y = selfing_rates,
  type = "s",
  col = "#E69F00"  # Example color for selfing rate
)

# Add a legend for selfing rates
legend(
  "topright",
```

```
legend = "Selfing Rate for Chromosome 1",
col = "#E69F00",
lty = 1,
cex = 0.75,
x.intersp = 0.5,
y.intersp = 0.8
)
```

Selfing Rate Over Time for Chromosome 1