Practical 3: Bayesian estimation of speciation times

In this practical, we are interested in estimating the divergence time, T, between polar bears and brown bears using genomics data.

We are going to be using Approximate Bayesian Computation (ABC) methods to inference such time. To help us with that, we would want to calculate summary statistics.

First, 10^4 simulations are performed by drawing from a prior distribution of T, and recording the drawn values and the corresponding summary statistics generated by that value of T.

The prior distribution is chosen to be uniform with lower and upper bound values of 200k and 700k, respectively.

After 10^4 simulations, we perform correlation plots in order to study the importance of the summary statistics. From Table 1 and Figure 4, we can deduce that the F_{st} summary statistic is the most important one, and therefore will be the one used to perform the ABC algorithm.

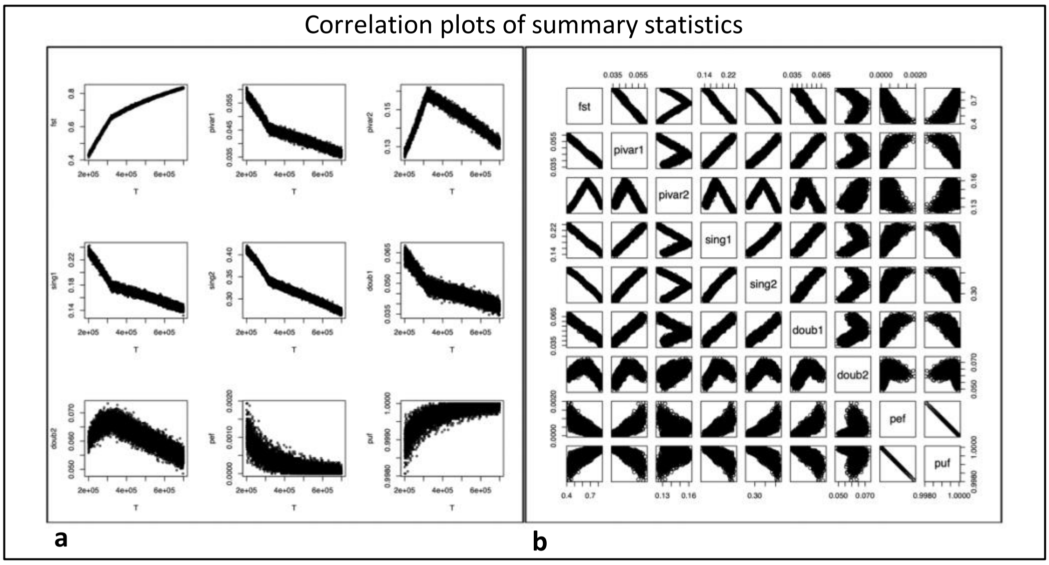

Figure 4: Correlation plots between (a) summary statistics, and between (b) summary statistics and T

Table 1: Correlation between summary statistics

	fst	pivar1	pivar2	sing1	sing2	doub1	doub2	pef	puf
fst	1	-0.990	0.005	-0.989	-0.991	-0.964	-0.592	-0.872	0.872
pivar1	-0.990	1	-0.035	0.983	0.978	0.962	0.565	0.865	-0.865
pivar2	0.005	-0.035	1	-0.044	0.055	-0.041	0.673	-0.120	0.120
sing1	-0.989	0.983	-0.044	1	0.979	0.954	0.558	0.863	-0.863
sing2	-0.991	0.978	0.055	0.979	1	0.953	0.622	0.853	-0.853
doub1	-0.964	0.962	-0.041	0.954	0.953	1	0.547	0.843	-0.843
doub2	-0.592	0.565	0.673	0.558	0.622	0.547	1	0.429	-0.429
pef	-0.872	0.865	-0.120	0.863	0.853	0.843	0.429	1	-1
puf	0.872	-0.865	0.120	-0.863	-0.853	-0.843	-0.429	-1	1

One last step before calculating the posterior distribution is scaling the observed and simulated summary statistics with mean = 0 and standard deviation = 1. Now, we can compute the posterior distribution using the abc rejection algorithm. Figure 5 below shows the result of the histogram of the accepted values of T.

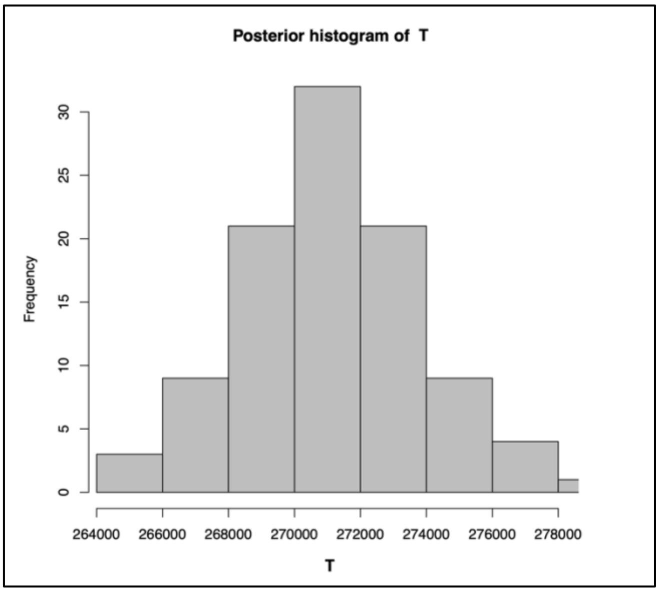

Figure 5: Posterior probability histogram of the divergence time, T.

Some useful statistical values of this distribution:

- Mean = 271068.3
- Median = 271176.1
- 95% credible interval = [265536.2, 276886.4]
- Notable quantiles:
  - Q1 (25%) = 269069.3
  - Q2 (50%) = 271021.5
  - Q3 (75%) = 273008.4

Code for the simulation, correlation plots, and the calculation for the posterior distribution ca be found in **Appendix A**.

Practical 3

Code for the simulation:

```
# number of chromosomes for each species
nChroms.polar = nrow(polar.brown.sfs)-1
nChroms.brown = ncol(polar.brown.sfs)-1

# number of analysed sites
nrSites = sum(polar.brown.sfs, na.rm = T)

# calculating summary statistics
obsSummaryStats = calcSummaryStats(polar.brown.sfs)

# SIMULATIONS
# number of simulations
nrSimul = 1e4

# define a uniform prior distribution to our parameter of interest
# (T - divergence time)
T = runif(nrSimul, 2e5, 7e5)

# preallocate dataframe to store summary statistics values
simulatedSummaryStats = matrix(NA, nrow = nrSimul, ncol =
length(obsSummaryStats))
colnames(simulatedSummaryStats) = c("fst", "pivar1", "pivar2",
"sing1", "sing2", "doub1", "doub2", "pef", "puf")

# set path to the "ms" software
msDir = "../Notebooks/Bayesian/Software/msdir/ms"

# set the name of the output text file
fout = "ms.txt"

# run simulation
for(i in 1:nrSimul){
  # simulate data
  simulate(T = T[i], M = 0, nr_snps = nrSites, ms_dir = msDir,
           fout = fout)

  # calculate summary statistics
  simulatedSFS = fromMStoSFS(fout, nrSites, nChroms.polar,
                             nChroms.brown)
  simulatedSummaryStats[i, 1:length(obsSummaryStats)] =
  calcSummaryStats(simulatedSFS)
}
```

Code for the correlation plots and the table:

```
# CHOOSING IMPORTANT SUMMARY STATISTICS
pairs(simulatedSummaryStats)

cor(simulatedSummaryStats)

par(mfrow=c(3,3))
for(i in 1:length(colnames(simulatedSummaryStats))){
  plot(T, simulatedSummaryStats [,i], cex = 0.5,
       ylab = colnames(simulatedSummaryStats)[i])
}
```

Code for computing the posterior distribution:

```
# POSTERIOR PROBABILITY DISTRIBUTION

sumStats = rbind(obsSummaryStats, simulatedSummaryStats)
scaledStats = scale(sumStats)

library(abc)
post = abc(target = scaledStats[1,1], param = T,
           sumstat = scaledStats[2:(nrow(scaledStats)-1),1],
           tol = 0.01, method = "rejection")
hist(post)
```