ABCME: Summary statistics selection for ABC inference in R

Matt Nunes* and David Balding†

*Lancaster University †UCL Genetics Institute

Outline

- Motivation: why the `ABCME` package?
- Description of the package
- Example of informative summary selection algorithms
- Implementation and examples of the `ABCME` package
- Summary

ABC in a nutshell

- For complex or high-dimensional data, inference about a parameter θ is often performed via Approximate Bayesian Computation (ABC), since likelihoods are often intractable/difficult to compute.

- In ABC, datasets X_i are simulated under a model, $M(\theta)$ and then **summary statistics** of the simulated data are used to compare to the (summaries of the) observed dataset X for inference.

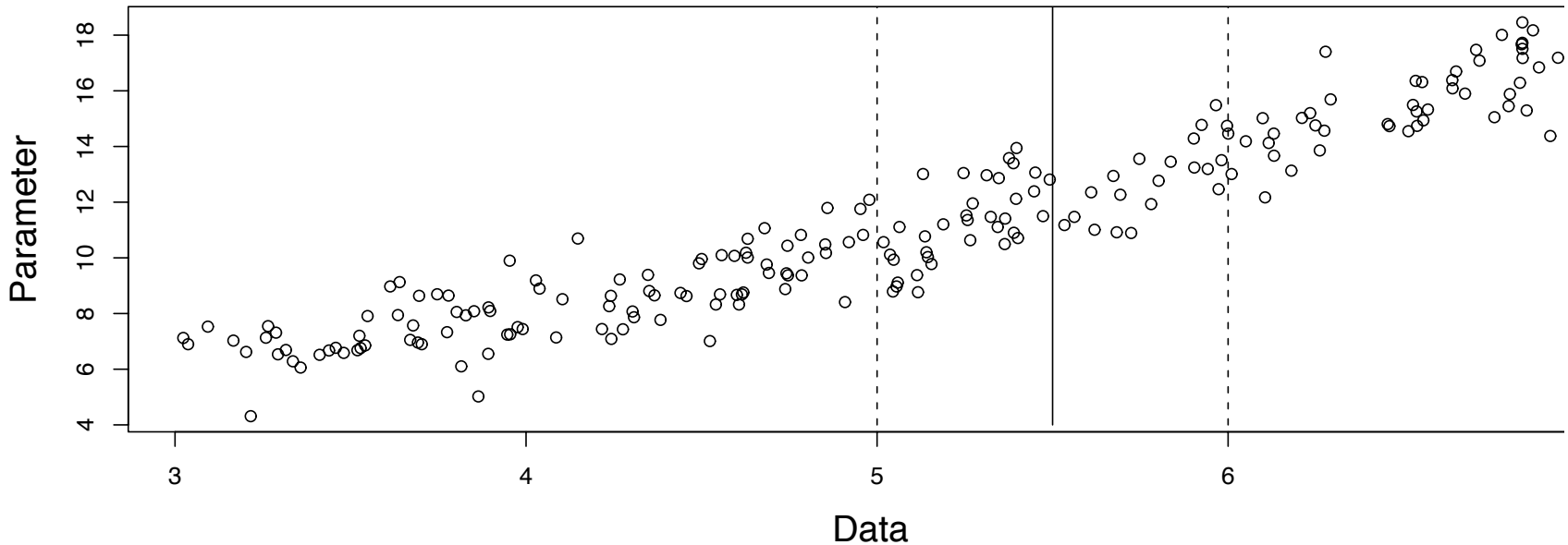
ABC in a nutshell

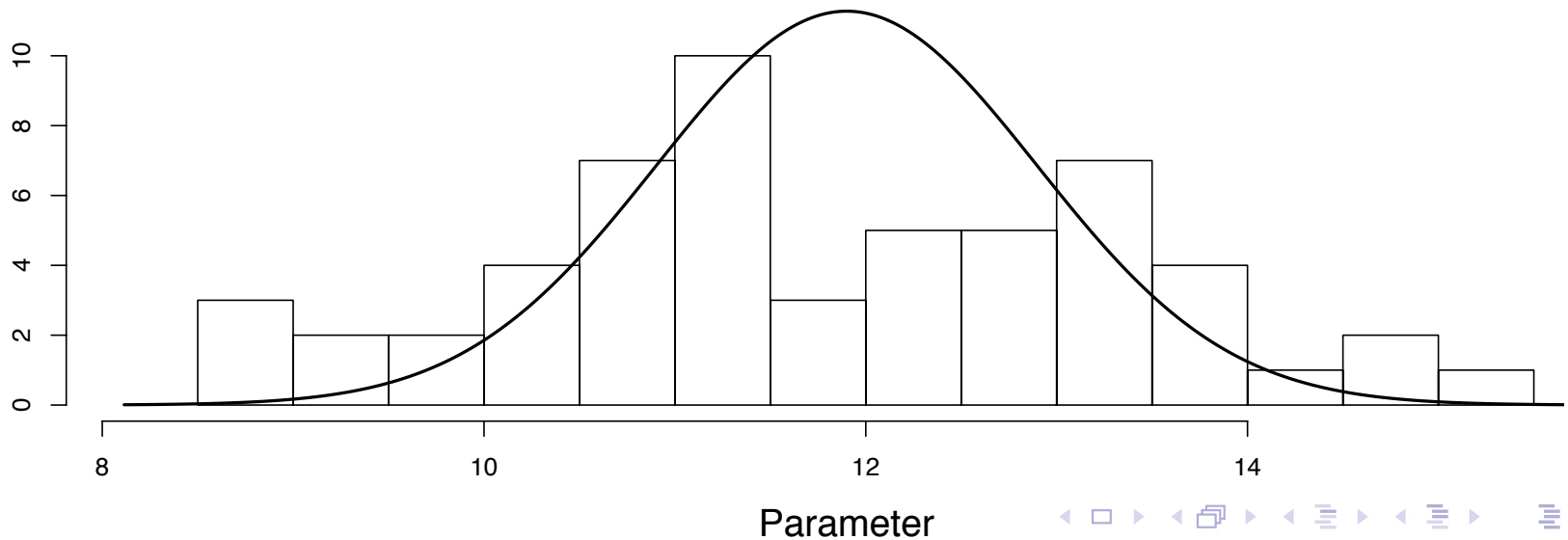
- For complex or high-dimensional data, inference about a parameter θ is often performed via Approximate Bayesian Computation (ABC), since likelihoods are often intractable/difficult to compute.

- In ABC, datasets X_i are simulated under a model, $M(\theta)$ and then **summary statistics** of the simulated data are used to compare to the (summaries of the) observed dataset X for inference.

  e.g, for DNA sequences, the number of *polymorphic sites* is useful for global parameters such as the mutation rate.

Rejection-ABC





Motivation: which statistics are best?

- In some scientific areas (e.g. population genetics), there are many well-established summary statistics that investigators know and love, often selected by investigators on the basis of intuition and established practice in the field.

Motivation: which statistics are best?

- In some scientific areas (e.g. population genetics), there are many well-established summary statistics that investigators know and love, often selected by investigators on the basis of intuition and established practice in the field.
- However, the most informative summary statistics are difficult to judge, and depend on the application and observed dataset under investigation.

Motivation: which statistics are best?

- In some scientific areas (e.g. population genetics), there are many well-established summary statistics that investigators know and love, often selected by investigators on the basis of intuition and established practice in the field.
- However, the most informative summary statistics are difficult to judge, and depend on the application and observed dataset under investigation.

$\rightsquigarrow$ Given a set of summary statistics, the `ABCME` package offers practitioners automatic algorithms for finding the best subset of statistics.

What the `ABCME` package does

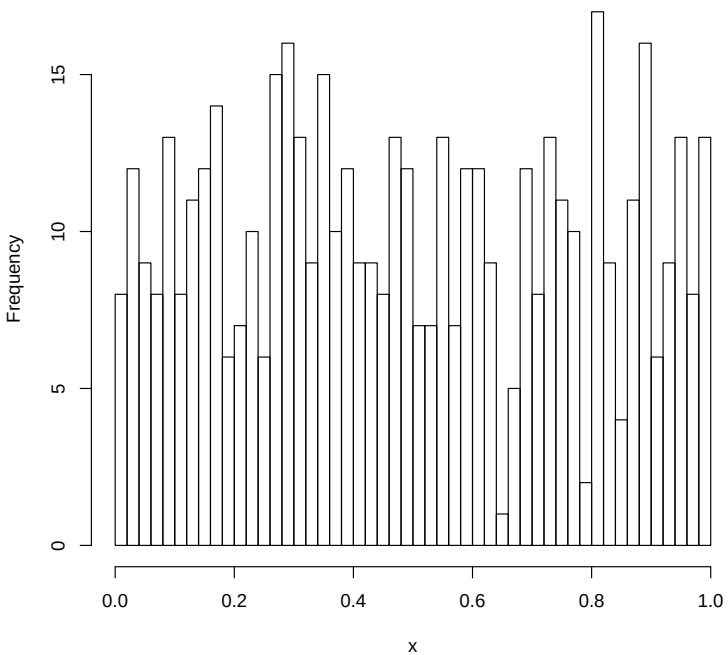
- The `ABCME` package implementing procedures to select summary statistics for ABC inference, namely
  - Approximate sufficiency ratio (Joyce and Marjoram, 2008)
  - selection via PLS regression (Wegmann et al., 2009)
  - summary selection using a minimum entropy criterion (Nunes and Balding, 2010)
  - two-stage approximate error summary selection (Nunes and Balding, 2010)

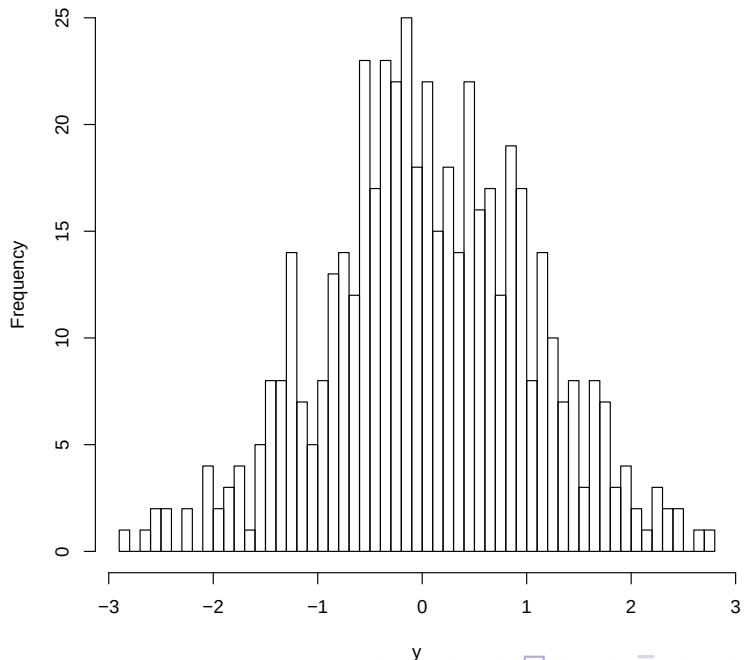
- Code allows for greedy searches and reduction of the space of summary statistics.
- Designed to be as **modular** as possible for flexibility.
- Suggests: `abc` (Csillery et al., 2011)
  `pls` (Mevik and Wehrens, 2007)

Choosing summaries via minimum entropy

ME algorithm: minimises sample-based entropy of posterior samples as a heuristic for selecting summary statistics.

Motivation: Entropy measures concentration of the posterior sample ($\leftrightarrow$ spread, peakedness of distribution).





Choosing summaries via minimum entropy

ME algorithm: minimises sample-based entropy of posterior samples as a heuristic for selecting summary statistics.

Motivation: Entropy measures concentration of the posterior sample ($\leftrightarrow$ spread, peakedness of distribution).

For every subset of summary statistics $S \in \mathcal{P}(\Omega)$:

1. Perform ABC inference to obtain posterior sample (e.g. rejection-ABC using the `abc` function in the `abc` package).
2. Compute sample entropy of resulting posterior. The `ABCME` function `nn.ent` can be used to compute "nearest neighbour entropy" (Singh et al., 2003).

- Choose the subset S with the minimum entropy.

Two-stage procedure

Stage 1: Find an initial good candidate $S \subset \Omega$ via rejection-ABC (e.g. using ME algorithm).

Stage 2: Using S identified in Stage 1, take k *simulated* datasets nearest S_0. For each $S \subset \Omega$, perform rejection-ABC and compute the error of $\Pi(\theta|S)$ for each of the k datasets, then select the subset of Ω that minimises mean square error.

Two-stage procedure

Stage 1: Find an initial good candidate $S \subset \Omega$ via rejection-ABC (e.g. using ME algorithm).

Stage 2: Using S identified in Stage 1, take k *simulated* datasets nearest S_0. For each $S \subset \Omega$, perform rejection-ABC and compute the error of $\Pi(\theta|S)$ for each of the k datasets, then select the subset of Ω that minimises mean square error.

- The two-stage algorithm is only dependent on the subset chosen in Stage 1.
- Any preferred measure of accuracy can be used in place of MSE.

ME algorithm implementation: `selectsumm()`

Function call:

```
selectsumm(theta,stats,data,abcmethod,crit,...)
```

Arguments:

`theta`: parameter values used to generate data from a model

`stats`: summary statistics computed from simulated datasets corresponding to `theta`

`data`: the summary statistics for the observed dataset

`abcmethod`: a function to perform an ABC algorithm (e.g. `abc`)

`crit`: an entropy function to minimize using the approximate posterior samples (e.g `nn.ent`)

Two stage algorithm implementation: `stage2()`

Function call:

```
stage2(theta,stats,data,stage1,abcmethod,crit,dsets,...)
```

Two stage algorithm implementation: `stage2()`

Function call:

```
stage2(theta,stats,data,stage1,abcmethod,crit,dsets,...)
```

Same arguments as `selectsumm` except:

`stage1`: index of the best summary subset from "stage 1"

`crit`: an error criterion to minimize using simulated datasets (e.g the `ABCME` function `mse`)

`dsets`: number of closest simulated datasets to minimize `crit`

Two stage algorithm implementation: `stage2()`

Function call:

```
stage2(theta,stats,data,stage1,abcmethod,crit,dsets,...)
```

Same arguments as `selectsumm` except:

`stage1`: index of the best summary subset from "stage 1"

`crit`: an error criterion to minimize using simulated datasets (e.g the `ABCME` function `mse`)

`dsets`: number of closest simulated datasets to minimize `crit`

There are other optional arguments to both algorithms including: `limit`, `do.only`, `final.dens`...

Example: `coal` Data

- Datasets consisted of sets of 50 haplotypes (DNA sequences) simulated under the coalescent model, summarized by 6 statistics, e.g. the number of segregating sites, the number of distinct haplotypes etc.

Example: `coal` Data

- Datasets consisted of sets of 50 haplotypes (DNA sequences) simulated under the coalescent model, summarized by 6 statistics, e.g. the number of segregating sites, the number of distinct haplotypes etc.

- The goal is inference about the scaled mutation and recombination parameters, θ and ρ, which were simulated under uniform priors for input into the model.

Example: `coal` Data

- Datasets consisted of sets of 50 haplotypes (DNA sequences) simulated under the coalescent model, summarized by 6 statistics, e.g. the number of segregating sites, the number of distinct haplotypes etc.

- The goal is inference about the scaled mutation and recombination parameters, θ and ρ, which were simulated under uniform priors for input into the model.

- `coal` consists of two columns of parameter values (θ, ρ), followed by the 6 columns of statistics (computed from 10^6 datasets simulated under the model).

Example (code snippet I)

```
# load coalescent data:
> data(coal)
> data(coalobs)

# use entropy to find subset guess (bivariate inference):

> stage1<-selectsumm(coal[,1:2],coal[,3:8],coalobs,
+ abcmethod=abc,crit=nn.ent)

doing statistics:  1 ( 1 / 63 )
doing statistics:  2 ( 2 / 63 )
doing statistics:  3 ( 3 / 63 )
doing statistics:  4 ( 4 / 63 )
doing statistics:  5 ( 5 / 63 )
doing statistics:  6 ( 6 / 63 )
doing statistics:  1 2 ( 7 / 63 )
...
> stage1$best
C1 C4

1 4
```

Example (code snippet II)

Now perform `stage2` using `stage1$best` and `mse` as the error function, with some optional arguments:

- `final.dens=TRUE` to retrieve approximate posterior sample
- `limit=3` to limit the search to subsets of less than 4 statistics

```
> st2<-stage2(coal[,1:2],coal[,3:9],coalobs,stage1=c(1,4),
+abcmethod=abc,crit=mse,dsets=100,final.dens=T,limit=3)

best guess subset is:  9
close datasets:  100
dataset...  1
dataset...  2
...
getting final posterior sample...done.
```

- Can then estimate and plot the density using e.g. `kde2d` + `filled.contour/persp` etc

Concluding remarks

- The `ABCME` package provides an implementation for recent ABC summary statistics selection methods.
- The package could be easily extended to use other ABC methods (e.g. MCMC-ABC, SMC-ABC).
- Other error/entropy criteria can also be implemented within the main routines of the package.
- Since the separate subset calculations are independent, the routines could benefit from parallelization (using e.g. `snow`).

More information about *ABCME* can be found at:

`http://www.maths.lancs.ac.uk/~nunes/ABCME.html`

Csillery, K., M. Blum, and O. Francois (2011). *abc*: Tools for approximate bayesian computation (abc). R package version 1.3.

Joyce, P. and P. Marjoram (2008). Approximately sufficient statistics and Bayesian computation. *Stat. Appl. Gen. Mol. Biol. 7*(1), 1–16.

Mevik, B.-H. and R. Wehrens (2007). The PLS package: principal component and partial least squares regression in R. *J. Stat. Soft. 18*(2), 1–24.

Nunes, M. A. and D. J. Balding (2010). On optimal selection of summary statistics for Approximate Bayesian Computation. *Stat. Appl. Genet. Mol. Biol. 9*(1).

Singh, H., V. Misra, N.and Hnizdo, A. Fedorowicz, and E. Demchuk (2003). Nearest neighbor estimates of entropy. *Am. J. Math. Man. Sci. 23*, 301–321.

Wegmann, D., C. Leuenberger, and L. Excoffier (2009). Efficient approximate Bayesian computation coupled with Markov chain Monte Carlo without likelihood. *Genetics 182*(4), 1207–1218.

Rejection-ABC & stage 2