---
title: '**Applying eSMC to Reveal the Past Demographic History of Canihua**'
subtitle: '** Module 3502-470 Plant Genetic Resources, Dataset Canihua_3**'
author: "Emilia Koch, 808462"
date: "February 03, 2020"
output:
  html_document: default
  pdf_document: default

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
options(knitr.table.format = "html")
```
<style>
body {
text-align: justify}
</style>

Method

The ecological Sequentially Markovian Coalescent (eSMC) is a novel method, described in the paper "Inference of past demography, dormancy and self-fertilization rates from whole genome sequence data" published in Pols Genetics in April, 2020 (Sellinger et al. 2020b). The implementation of the model is further described at and the required package can be downloaded from the following GitHub repository: https://github.com/TPPSellinger/eSMC (Sellinger 2020a).

The model infers population size and displays changes over past generations. In contrast to most other currently used models, that are based on Sequential Markovian Coalescence theory, it runs on full genome polymorphism data. ESMC runs on individuals. It analyzes two haplotypes at the time, using the sequence of the individual to be investigated and the one of the reference genome. While inferring demographic history, it allows to account for and to estimate self-fertilization rates or rates of seed or egg-banking. The latter one can occur due to low germination rates or seed dormancy (Sellinger et al. 2020b).

Prior to running the analysis, information is needed regarding to: (I) the individuals to run it on, (II) the chromosome number (scaffold number), (III) the recombination and the mutation rate per site per generation, and (IV) whether to account for the recombination, the selfing or the germination rate (Sellinger et al. 2020b).

Many currently used models are based on the assumptions of the Wright-Fisher model. In the presence of self-fertilization and seed-dormancy these assumptions are violated. To account for these ecological and life history traits, eSMC uses two ratios, the molecular ratio of recombination (*r*) over mutation (*μ*) rate $\frac{r}{\mu}$ and the effective ratio of the two parameters $\frac{\rho}{\theta}$. With the assumptions of the Wright-Fisher model, the ratios are equal. However, self-fertilization and seed-banking causes these ratios to differ. Accounting for this differentiation allows to the model reduce bias in its inference (Sellinger et al. 2020b).

However, self-fertilization and seed-dormancy effect these ratios and the effective population size in opposing ways. Self-fertilization is decreasing the effective population size while dormant seeds are increasing it. If used simultaneously, the individual signatures of the two systems cannot get fully disentangled with eSMC. Therefore, only one of the two is accounted for at the same time. (Sellinger et al. 2020b).

The analyzed dataset contains genotypic information of *Chenopodium pallidicaule* commonly called canihua. The vcf file comprises genomic data of 86 plants and nine chromosomes, respectively.

Chenopodium species are predominantly autogenous (Sauer J. D. 1993), and so is canihua (Simmonds N. W. 1965). Therefore the model is used to adjust for self-ferilization while predicting its past population history.

In the following, the major steps to run eSMC are described: (I) software and software packages needed, (II) choosing a representative sample of the population, (III) creating the input files, (IV) running eSMC.

Software and Software Packages

The analysis is conducted in **R** 3.6.3 (R Development Core Team 2019), using the platform of **RStudio** (RStudio Team 2020). **The Packages** needed are: **parallel* (R Core Team 2019)*, **readODS* (Schutten et al. 2018)*, **stringdist* (van der Loo 2014)*, **tidyverse* (Wickham et al. 2019)*, **dplyr* (Wickham et al. 2020a)*, **installr* (Galili 2019)*, **devtools* (Wickham et al. 2020b)*, **eSMC* (Sellinger 2020a)* and **vcfR* (Knaus and Grünwald 2017)*.

```{r eval=FALSE, include=TRUE }
# Adjust the working directory
setwd("~/Term")

# Packages needed to create the input files
library(vcfR)
library(parallel)
library(readODS) #For accession/sample names
library(stringdist)
library(tidyverse)
library(dplyr)
# Additional packages to run eSMC
library("installr")
library(devtools)
library(eSMC)
```

Sampling the Individuals

As an initial step, we choose a representable sample of individuals. Therefore, the results of a Principal component analysis (PCA) are used, which were obtained from Dr. Böndel (Böndel 2020).

```{r Fig.1, echo=FALSE, fig.align='center', fig.cap=" **Figure 1. Principal component analysis (PCA) ** *The PCA is revealing the population structure of Chenopodium pallidicaule plants in the dataset “canihua_3”. Outliers are excluded. Three subpopulations are observed *" }
############
# Display PCA Results
############

# Adjust the working directory to read in the file with the PCA results
TXT<-read.table("C:/Users/koche/Documents/Term/canihua_3/canihua_3_no_5.txt") # read in txt file of with PCA result

# Plot PCA result to get an overview over the population structure
plot(TXT$EV1,TXT$EV2,xlab="PC1: 4.9 % variance",ylab="PC2: 3.4% variance",
   xlim=c(-0.4,0.16), #xlim=c(-0.4,0.45),
   ylim=c(-0.16,0.35), #ylim=c(-0.9,0.35),
    col="black", # this way we basically plot an "empty" graph to which we can then
    # add the groups in different colors
```

```
    pch=1,las=1)
```

Figure 1 revealed that there are three subpopulations. From each subpopulation a random sample of three individuals is chosen, resulting in a sample size of nine individuals.

```{r  eval=FALSE, include=TRUE}
############
# Identify a Sample of Individuals for the eSMC Analysis
############

# Assign individuals of each subgroup into an individual data frame
aa <- TXT[c(1:3)]
group1 <- subset(aa, EV1 > -0.25 & EV1 < -0.15 & EV2 > -0.2 & EV2 < -0.1 , select=c(sample.id,EV1,EV2))
group2 <- subset(aa, EV1 > -0.1  & EV1 < -0.0  & EV2 > -0.2 & EV2 <  0.0 , select=c(sample.id,EV1,EV2))
group3 <- subset(aa, EV1 > -0.0  & EV1 <  0.2  & EV2 > -0.1 & EV2 <  0.1 , select=c(sample.id,EV1,EV2))

# Rrandomly choose 3 individuals from each data frame
id1 <- sample(group1$sample.id, size=3, replace= FALSE, prob = NULL)
id2 <- sample(group2$sample.id, size=3, replace= FALSE, prob = NULL)
id3 <- sample(group3$sample.id, size=3, replace= FALSE, prob = NULL)
id1
id2
id3
```

For the first small subpopulation on the left (Figure 1), the individuals B26, B23 and B24 were sampled (id1). For the second subpopulation in the middle, the individuals B35, B33 and U8 were sampled (id2). In the third and biggest subpopulation on the right, the individuals Ch_pal_05, Ch_pal_39 and Ch_pal_49 were sampled (id3). Both, the sampled individuals and the chromosomes to be analyzed are assigned to a vector.

```{r }
# assign a vector for chosen individuals and chromosomes
good_ind <- c("B26", "B23", "B24", "B35", "B33", "U8", "Ch_pal_05", "Ch_pal_39", "Ch_pal_49")
good_scaffs <- c("lcl|Cp1","lcl|Cp2","lcl|Cp3","lcl|Cp4","lcl|Cp5","lcl|Cp6","lcl|Cp7","lcl|Cp8","lcl|Cp9")
```

Creating Input Files for eSMC

As eSMC runs on individuals, a separate input file is needed for each individual and each chromosome, respectively. A specific format is required for the input files. They come as a txt file with three columns, containing the base of the individual to be analyzed in the first and the reference base in the second column. Their position on the chromosome is noted in the third column.

To create the input files, we first read in the dataset “canihua_3.vcf”.
For each of the sampled individuals and chromosomes, in “good_ind” and “good_scaffs, we pick the information from the fixed and the gt part of the VCF, as described in the code below. Then we extract the single nucleotide polymorphism (SNP) information, and the alternative base, transform the into the desired format and add the respective SNP position. We exclude homozygous sites and sites with less than 100 SNPs. The output is written into a txt file.

The name of the input files contains the method name “eSMC”, the number of the individual, the chromosome number and the name of the individual in the dataset, for example: eSMC1_1_B26.

```{r  eval=FALSE, include=TRUE}
```

```
##########
# Create Input Files for the Chosen Individuals
##########

only_var <- TRUE          # true to only retain heterozygous positions
kickout_less100SNPs <- TRUE # true to not record contigs with < 100 SNPs

file <- read.vcfR("canihua_3.vcf",skip = 0, nrows=-1) # read in the vcf file
sample_names <-colnames(file@gt) # take column names of genotype part

# loop over individuals and chromosomes
for (ind in 1:length(good_ind)){        # ind as index, loop over individuals
  for (scaff in 1:length(good_scaffs)){ # scaff as index, loop over chromosomes

    # pick data from the respective chromosome (scaff) and individual (ind)
    name_pick <- c(1,grep(pattern=good_ind[ind],x = sample_names)) # select individual
    SNP_pick  <- which(file@fix[,"CHROM"]==good_scaffs[scaff])      # choose chromosomes from fixed part of vcf
    vcf_single <- file[SNP_pick,name_pick] # create a list with both

    # merge fixed and gt part into one tidy frame
    haptemp <- vcfR2tidy(vcf_single,single_frame = TRUE)

    # transform into input file format
    # split selected colum, if ther is a "/", and inserts it into a new dataframe
    hapfile <- as.data.frame(strsplit(haptemp$dat$gt_GT_alleles,"/"))
    colnames(hapfile) <- NULL # relabel row names
    hapfile <- as.matrix(hapfile) # turn into a matrix
    hapfile <- rbind(hapfile,haptemp$dat$POS) # add the allel's position

    # remove homozygous sites
    if (only_var){keep1 <- stringdist(hapfile[1,],hapfile[2,],"hamming")
              hapfile <- hapfile[,as.logical(keep1)]}
    # only use output with at least 100 SNP's/  heterozygous positions
    if (kickout_less100SNPs){if (ncol(hapfile)<100){next}}

    # write data into txt files
    if (!only_var){ write.table(hapfile,file=paste0("eSMCa",ind, "_",scaff,".txt"),
                quote = FALSE,row.names = FALSE,col.names = FALSE)}
      else      { write.table(hapfile,file=paste0("eSMC",ind, "_",scaff,"_",good_ind[ind],".txt"),
                quote = FALSE,row.names = FALSE,col.names = FALSE)
              }
  }}
```

The input files are saved in the working directory.

Perform eSMC

Several parameters are adjusted before running eSMC. To start, the mutation rate (mu) has been set to $6.95*10^{-9}$ mutations peer site per generation (Weng et al. 2019) and the recombination rate (r) has been set to $3.6*10^{-8}$ per site per generation (Salomé et al. 2012). The values are taken from *Arabidopsis thaliana*, because rates for species which are more closely related to canihua could not be found. Canihua is a predominantly self-fertilizing crop (Simmonds N. W. 1965), therefore eSMC is used to account for self-fertilization. Hence, the parameters to estimate the effective recombination rate (ER) and germination rate (SB) are set to FALSE, while the estimation of the selfing rate (SF) is set to TRUE. Default settings for the boundaries are kept, as they cover a very broad and biologically relevant range.

```{r}
##############
# Set Parameters for eSMC
##############

mu=6.95*10^(-9) # mutation rate per position per generation
r= 3.6*10^(-8)  # recombination rate per position per generation
rho=r/mu        # ratio recombination/mutation

# set one to true (rest to false)
ER=F # false to estimate recombination rate
SF=T # true to estimate selfing rate
SB=F # false to estimate germination rate

# set boundaries
BoxB=c(0.05,1) #  min and max value of germination rate
Boxs=c(0,0.99) #  min and max value of selfing rate
```

In the next step, we read in the input files. Only the files with the individuals and chromosomes on which the analysis is run on are kept in the working directory. For this analysis, files of all nine sampled individuals with chromosome one to three are kept.

In input files, the individuals were assigned numbers. The number of the analyzed individual is adjusted for the “individual_name” and the “result”. Here, the analysis will start with the first sampled individual. The results were put into lists and stored. Their plots are display in the results section.

```{r  eval=FALSE, include=TRUE}
##############
# Load Input Files
##############

#First adjust number of the individual in "individual_name" the resultnumber
individual_name <- paste("eSMC",1,sep="")  # create the individual_name, use pattern of input file name
single_ind <- list.files(pattern = individual_name,full.names = TRUE) # load the files with this pattern

test_data <- vector("list",length(single_ind)) # create a vector, its lenth equals nr of  txt files

for (j in 1:length(single_ind)){
   input_name <- single_ind[j]  # []=inex, takes the value in this position
   test_data[[j]] <- as.matrix(read.table(input_name))   # [[]] line of the index
   }

NC <- length(test_data) # create a vector with length of number of analyzed chromosomes

##############
# Run eSMC
##############

result1= eSMC(n=30,rho=rho,test_data,BoxB=BoxB,Boxs=Boxs,SB=SB,SF=SF,Rho=ER,Check=F,NC=NC)

##############
# Save the Results and Combine Them into Lists
##############

# save the respective result
save(result1,file="eSMC_r1_B26.RData") # filename includes the number and the name of the individual
#save(result2,file="eSMC_r2_B23.RData")
```

```
#save(result3,file="eSMC_r3_B24.RData")
#save(result4,file="eSMC_r4_B35.RData")
#save(result5,file="eSMC_r5_B33.RData")
#save(result6,file="eSMC_r6_U8.RData")
#save(result7,file="eSMC_r7_Ch_pal_05.RData")
#save(result8,file="eSMC_r8_Ch_pal_39.RData")
#save(result9,file="eSMC_r9_Ch_pal_49.RData")

# create result lists of all individuals and of each subpopulation
result_list =list(result1,result2,result3,result4,result5,result6,result7,result8,result9)
pop1_list =list(result1,result2,result3)
pop2_list =list(result4,result5,result6)
pop3_list =list(result7,result8,result9)

# save the result lists
save(result_list,file="result_list.RData")
save(pop1_list,file="pop1_list.RData")
save(pop2_list,file="pop2_list.RData")
save(pop3_list,file="pop3_list.RData")
```

Results

The eSMC result estimates the past demographic history of the canihua population, while accounting for self-fertilization. Figure 2 displays the population size (Xi) depending on the past generations. Each line represents an estimation based on one individual and three chromosomes. For the nine sampled haplotypes the graph infers a population history from around 30 to 3000 generations into the past. The population size decreases until around 150 generations ago. Up this time, the haplotypes appear to have the same population history, with low variability between the individuals. Starting from around 70 generations ago, two opposing tendencies are displayed, the shift to an increasing population size for three individuals and a continued decline for the rest of them.

```{r Fig.2, echo=FALSE, fig.align='center',  fig.cap=" **Figure 2. Estimated demographic history of *Chenopodium pallidicaule*.** *The graph displays nine sampled individuals of a Chenopodium pallidicaule population, using eSMC to account for selfing. Chromosome one to three are used for the analysis for each individual respectively. The Mutation rate is set to 6.95 10^-9^ per generation per site and the recombination rate is set to 3.6 10^-8^.* " }

library(eSMC)
load("~/Term/result_list.RData")

Plot_esmc_results(result_list,mu,WP=F,LIST=T,x=c(10^1.2,10^3.6),y=c(2,4)) # plot all results
```

Figure 3 shows the eSMC results for each of the three subpopulations. In the first subpopulation (left) the population size is consistently decreasing. Whereas, the second subpopulation (middle) and the third subpopulation (right) show the very recent split into the two opposing directions.
For these two subpopulations, the population history of the individuals B35, Ch_pal_49 and Ch_pal_39 shows a continuously decreasing trend, while population size for the populations of the individuals U8, B33 and Ch_pal_05 starts to increase again.

```{r Fig.3, fig.align='center',  fig.cap=" **Figure 3. Demographic history of the three sampled individuals for each subpopulation of *Chenopodium pallidicaule***. *The demographic history is inferred using eSMC and accounting for self-
```

```
fertilization. Chromosome one to three are used for the analysis for each individual respectively. The Mutation rate is set to 6.95 10^-9^ per generation per site and the recombination rate is set to 3.6 10^-8^. The first subpopulation (left) displays the individuals B26, B23 and B24, reading the from top red line to bottom one. The second subpopulation (middle) shows U8, B33 and B35. The third subpopulation (right) shows Ch_pal_05, Ch_pal_49 and Ch_pal_39. * "}

load("~/Term/pop1_list.RData")
load("~/Term/pop2_list.RData")
load("~/Term/pop3_list.RData")

par(mfrow=c(1,3))
Plot_esmc_results(pop1_list,mu,WP=F,LIST=T,x=c(10^1.2,10^3.6),y=c(2,4))
Plot_esmc_results(pop2_list,mu,WP=F,LIST=T,x=c(10^1.2,10^3.6),y=c(2,4))
Plot_esmc_results(pop3_list,mu,WP=F,LIST=T,x=c(10^1.2,10^3.6),y=c(2,4))
```

The algorithm inferred the self-fertilization rate (σ) of zero and germination rate (β) of 1 for all sampled individuals, which implies that there is no self-fertilization and no long-term seed-banking.

With three individuals three chromosomes per individual, 2.6 Mb of data was analyzed. The average running time was 90 minutes.

Discussion

ESMC only needs one or two individuals and scaffolds of minimally 1 MB length to accurately predict population history. These basic prerequisites were met by our dataset. As the quality of the result depends more on the accuracy of the sequence rather than on the amount, we cannot definitely say how good our predictions are (Sellinger et al. 2020b).

Due to the continuous decline in population size, it is assumed that the population goes through a prolonged bottleneck. Starting from around 70 generations ago, some individuals of the second and the third subpopulation diverging trends for their population size. This divergence could origin from beneficial mutations, which could have occurred simultaneously in both subgroups. This information on the demographic history of a population can be used as base for further models and to increase selection gain. However, recombination events might be displayed with a certain delay and sudden or recent demographic events would eventually not be displayed accurately (Sellinger et al. 2020b). It is important to note, that the same number of individuals was sampled of each sub-population, independent of its size. Including a certain percentage of each population could reveal clearer tendencies. Also increasing the number of hidden states and the number of included chromosomes could help to increase the precision of the result. However, this would also increase the computation time.

Because mutation and recombination rate are held constant over the time-period, only an average effect of real values of the selfing rates can be estimated within the time window. However, the estimated self-fertilization of 0 constitutes a contrast to literature, that describes canihua as a mainly self-fertilizing crop (Simmonds N. W. 1965; Sauer J. D. 1993). That the model accounts only for homologous recombination events, but not for, for example chromosomatic mutations, could be one reason for an underestimated self-fertilization rate (Sellinger et al. 2020b). One student had a self-fertilization rate of 0,99 which would make more sense, considering canihua's biological background (Damm, A. 2020).

Similar to the PCA, the ALStructure analysis also suggested three subpopulations (Herbold, T. 2020). Results of other students, who also use eSMC showed the same declining tendency for their population size (Damm et al. 2020). However, their time frame was different. One reason for this could be the different priors for the mutation and recombination rate (citation). Therefore, mutation and recombination rates of canihua would be needed to estimate the real time frame. These student's data also showed a difference in more recent generations, where the population size either continued to decrease or split up with a different pattern (Damm et al. 2020). These divergences could arise due to different samples or different datasets.

- Coalescence assumes neutral evolution – if some selection – con founding effect
- Tahima's D – negative - negative population growth responsible, not selection
- ??? didn't understand that yet

References

1. Böndel, K. (2020): Population Structure Result, University of Hohenheim, Stuttgart, Germany
2. Damm, A.; Herrmann, M., A. (2020): Exchange of eSMC Results, University of Hohenheim, Stuttgart, Germany
3. Galili, T. (2019): installr: Using R to Install Stuff on Windows OS (Such As: R, 'Rtools', 'RStudio', 'Git', and More!). Version 0.22.0. Available online at https://CRAN.R-project.org/package=installr.
4. Herbold, T. (2020): ALStructure Result, University of Hohenheim, Stuttgart, Germany
5. Knaus, B. J.; Grünwald, N. J. (2017): VCFR: a package to manipulate and visualize variant call format data in R. In Molecular ecology resources 17 (1), pp. 44–53. DOI: 10.1111/1755-0998.12549.
6. R Core Team (2019): R: A language and environment for statistical computing. Version 3.6.2: R Foundation for Statistical Computing, Vienna, Austria. Available online at https://www.R-project.org/.
7. R Development Core Team (2019): R: A language and environment for statistical computing: R Foundation for Statistical Computing, Vienna, Austria. Available online at http://www.R-project.org.
8. RStudio Team (2020): RStudio: Integrated Development Environment for R: RStudio, PBC, Boston, MA. Available online at http://www.rstudio.com/.
9. Salomé, P. A.; Bomblies, K.; Fitz, J.; Laitinen, R. A. E.; Warthmann, N.; Yant, L.; Weigel, D. (2012): The recombination landscape in Arabidopsis thaliana F2 populations. In Heredity 108 (4), pp. 447–455. DOI: 10.1038/hdy.2011.95.
10. Sauer J. D. (1993): Historical Geography of Crop Plants. A Select Roster: CRC Press.
11. Schutten, G.; Chan, C.; Leeper T. J. (2018): readODS: Read and Write ODS Files. Version 1.6.7. Available online at https://CRAN.R-project.org/package=readODS.
12. Sellinger, T. (2020a): eSMC: Ecological Sequentially Markovian Coalescent. Version 1.0.0. Available online at https://github.com/TPPSellinger/eSMC.
13. Sellinger, T. P.; Abu, A. D.; Moest, M.; Tellier, A. (2020b): Inference of past demography, dormancy and self-fertilization rates from whole genome sequence data. In PLoS genetics 16 (4), e1008698. DOI: 10.1371/journal.pgen.1008698.
14. Simmonds N. W. (1965): The Grain Chenopods of the Tropical American Highlands. In Economic botany 19 (3), pp. 223–235.

15. van der Loo, M. P. J. (2014): The stringdist package for approximate string matching. In The R Journal 6 (1), pp. 111–122. Available online at https://CRAN.R-project.org/package=stringdist.
16. Weng, M. L.; Becker, C.; Hildebrandt, J.; Neumann, M.; Rutter, M.; Shaw, R. et al. (2019): Fine-Grained Analysis of Spontaneous Mutation Spectrum and Frequency in Arabidopsis thaliana. In Genetics 211 (2), pp. 703–714. DOI: 10.1534/genetics.118.301721.
17. Wickham, H.; Averick, M.; Bryan, J.; Chang, W.; McGowan, L.; François, R. et al. (2019): Welcome to the Tidyverse. In Journal of Open Source Software 4 (43), p. 1686. DOI: 10.21105/joss.01686.
18. Wickham, H.; François, R.; Henry, L.; Müller, K. (2020a): dplyr: A Grammar of Data Manipulation. Version 1.0.0. Available online at https://CRAN.R-project.org/package=dplyr.
19. Wickham, H.; Hester, J.; Chang, W. (2020b): devtools: Tools to Make Developing R Packages Easier. Version 2.3.0. Available online at https://CRAN.R-project.org/package=devtools.