Accurate Local Ancestry Proportion Estimation from Genetic Summary Data

Hayley R. Stoneman[1]

Abstract

Local ancestry estimation from summary level data can be useful for mapping diseases in populations to genetic loci, as well as inferring natural selection. Here, we investigate the application of Summix, a method for ancestry proportion estimation from summary data, for local ancestry deconvolution. In order to use a method for discovery and research purposes, we must first determine whether it is accurate and precise. Here we show that Summix is able to generate unbiased estimates of local ancestry proportions with very small standard error, supporting its use for local ancestry deconvolution using summary level data. We then apply the method to chromosome 2 in the African/African American group in gnomAD to detect possible regions of selection.

Introduction

Local ancestry can be defined as the genetic ancestry at a specific chromosomal location. Historically, methods for local ancestry deconvolution have required access to individual level data, which are not always available due to privacy concerns. Instead, summary level data (allele frequencies for a population) are made publicly available. Local ancestry can be used to detect regions of selection and to map genomic regions to diseases. For these reasons, there is interest in applying Summix[1], a method for global ancestry proportion estimation using genetic summary data, for local ancestry. Since local ancestry will require applying Summix to a subset of the data (pertaining to the chromosomal localtion of interest), this will inherently involve applying the method to less data, which could lead to inaccurate results. Here we need to understand whether Summix can still be used to accurately and precisely estimate local ancestry proportions. After determining if Summix can indeed accurately assess local ancestry, it will be applied to real population data to detect possible regions of selection.

Materials and Methods

In order to determine whether Summix can be used for accurate and precise local ancestry estimation, the true local ancestry proportions and structure need to be known. As this is not easily obtainable, simulations are used to generate a dataset with known proportions. These simulations rely on the Hardy-Weinberg equation $p^2 + 2pq + q^2 = 1$ in which p = allele frequency (AF) of the reference allele and q = AF of the alternate allele. Likewise p^2, $2pq$, and q^2 are the genotype frequencies. In summary statistics, the reference AF for the population is reported, so this number can be used to generate probabilities for each of the three genotypes, which can then be sampled using a multinomial distribution, creating a simulated dataset with known proportions.

The gnomAD v3.1 dataset is used for the African/African American group on chromosome 19[2]. The African American group is known to be an admixed group with approximately 85% African (AFR) and 15% European (EUR) ancestry. The reference data is from the 1000 Genomes Project[3] and Human Genome Diversity Project (HGDP)[4]. gnomAD is a global resource that aggregates and harmonizes exome and genome sequencing data from many large-scale projects to make the summary data publically available for researchers. The latest version (v3.1.2) contains 76,156 total genomes.

The local ancestry method will be applied using a basic sliding windows algorithm, in which the user defines the window size in number of variants, and the window overlap size, also in number of variants. At each window, Summix will be applied to just the SNPs within the window, with overlapping SNPs assigned to the latter window.

To detect regions of selection, a variation of the sliding windows algorithm will be used, called the fast-catchup algorithm. In this method Summix will be applied to the SNPs within the window. Then, without moving the start pointer, the stop pointer advances forward. With each advance Summix is reapplied, until a threshold of difference is discovered. This defines one block of local ancestry, and the start pointer is advanced to the point of the stop pointer and the algorithm repeats. This method is better suited to detect regions of change, without as large of a risk of averaging out any differences.

All analyses will be performed using R version 4.1.0 using RStudio version 2021.9.1 on a windows PC with 12Gb RAM, 4 cores and 8 processors.

[1] Human Medical Genetics and Genomics Program, Hendricks Lab, University of Colorado Anschutz Medical Campus, Aurora, CO

Analysis Plan

Accurate and precise local ancestry estimation

To create the simulated AF, first an AFR and EUR simulated population will be created, each of size 1000. Then a weighted average of the AFs by desired ancestry proportions will be taken to get the simulated observed AF. 100 simulations will be run with 85% AFR and 15% EUR. A window size of 5000 SNPs with a 100 SNP overlap will be used.

For each block in each simulation replicate the accuracy and precision will be assessed. Accuracy will be defined as the bias, or the difference between the simulated ancestry proportion and the estimated ancestry proportion. Precision will be assessed by examining the standard error of the ancestry proportion estimate. Standard error will be assessed using bootstrapping, in which the SNPs within the window will be bootstrapped, then Summix will be reapplied. The standard deviation of the output ancestry proportion estimate will then be calculated. The bias and standard error will be averaged across all blocks in a replicate.

Real data application

Once it is confirmed that Summix can produce unbiased and minimally variable local ancestry estimates, it can be used to detect potential regions of selection. We will use the chromosome 2 data from gnomAD v3.1.2 for the African/African American group. A prior study found two potential regions of selection on chromosome 2, one with excess AFR ancestry and one with excess EUR ancestry[5]. P-values will be reported as raw p-values, but significant blocks will be tested against a Bonferroni-corrected threshold.

Results

Accurate and precise local ancestry estimation

Table 1: Summary of data used in this project

	AFR genomes	EUR genomes	Ref
gnomAD	20744	34029	2
1000G	840	621	3
HGDP	51	149	4

Table 2: Accuracy and Precision of Estimates

Bias	Standard Error
-1.13e-05	0.000827

A summary of the data used in this project are in Table 1. One hundred simulation replicates were run with 85% AFR and 15% EUR to assess the accuracy and precision of using Summix for local ancestry estimation. To calculate the standard error, 100 bootstrap replicates were used for each window. It was found that the average bias of the local ancestry estimate across all 98 local ancestry blocks across the 100 simulations was -0.0000113 as shown in Table 2.

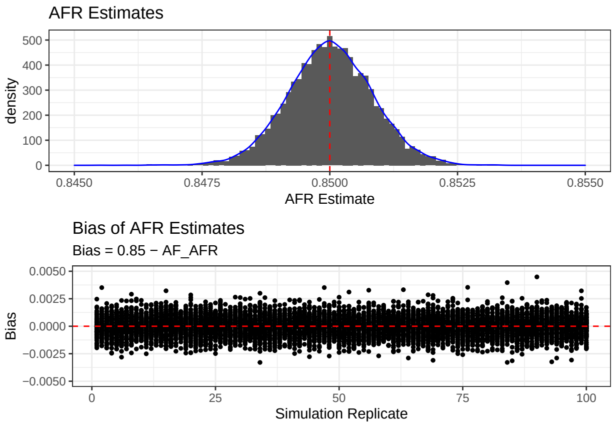

Figure 1: Accuracy of Local Ancestry estimates

The ancestry proportion estimate centered around the truth of 85% AFR (Figure 1). The average standard error of that local ancestry proportion estimation was 0.000827. The standard error was consistent across the chromosome (Figure 2).

Real data application

Summix was applied to detect regions of selection on chromosome 2 in the African/African American group in the gnomAD v3.1 data. Significant local ancestry blocks (with Bonferroni correction) are shown in Table 3. One significantly different block was found with excess African ancestry, suggesting selection at this region.

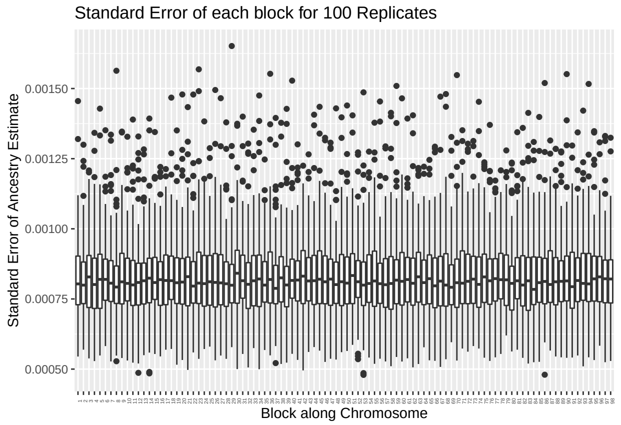

Figure 2: Precision of Local Ancestry estimates

Table 3: Significant Selection Signals from Chr 2

Start	End	AFR	EUR	P-val
89226026	91529992	0.875531	0.124469	9e-07

Discussion

Here we examine the feasibility of using Summix, a method for global ancestry proportion estimation from genetic summary data, to estimate local ancestry proportions. Local ancestry is the population structure at a particular chromosomal location. Therefore, using Summix for local ancestry deconvolution will require applying the method for a subset of the data (SNPs), rather than the entire genome. In order to determine whether Summix can be used for local ancestry estimation, the accuracy and precision of the estimates needed to be assessed. This was done using 100 simulations, in which genotypes of 85% AFR and 15% EUR were simulated, and the allele frequencies used as input for Summix.

From these results, we found that the estimated ancestry proportions for AFR centered around 85% (Figure 1). This shows that the estimates are unbiased, which is validation for the use of this method. Additionally, the average bias (true proportion - estimated proportion) was -0.000113, and ranged from -0.00329 to 0.00449. Using these 100 simulations, we find that the ancestry proportion estimates are off by no more than 0.5%, which is not likely a biologically meaningful difference, and therefore can be considered highly accurate.

Likewise, we determined the variability around these ancestry proportion estimates by calculating the standard error. The standard error was calculated by taking the standard deviation from the bootstrap sampling distribution. We found that the average standard error was 0.000827, and this was consistent across the chromosome. This suggests that the ancestry proportion estimates are highly precise, and that there are no regions of the chromosome for which this method should not be applied.

After determining that Summix was a valid method for local ancestry estimation from genetic summary data, we applied the method to chromosome 2 of the African/African American gnomAD group to detect potential regions of selection. We discovered one Bonferroni-corrected significant block from approximately 89.2Mb to 91.5Mb. This is a new region of selection from the prior study. This region contains many imunoglobulin markers, which are involved in the immune response.

There are some limitations to this study. First, only 100 simulation replicates and 100 bootstrap replicates were used. For the scope of this assignment, 100 replicates was used to keep the run-time of the simulations and bootstrapping under 24 hours. In the future, 1000 or 10,000 replicates would be useful to ensure more accurate assessments of bias and standard error. Additionally, this was only performed on a single chromosome (19), selected because it is a shorter chromosome and therefore has less data and shorter run-time. In the future this study should be repeated for all chromosomes to ensure portability across the genome. Finally, this was done for an admixed group with only two ancestry populations. In the future an admixed group of more than two ancestral populations should also be assessed to see if there are more nuances when there is more mixing. Additionally, we only applied the selection detection method to a single chromosome in a single group. Future studies should continue to validate this method and attempt to repeat previously discovered regions of selection in other admixed populations.

Overall, this study found that Summix can be used for accurate and precise local ancestry estimation. It also showed that this method can detect potential regions of selection by identifying local ancestry blocks with excess ancestry. This opens the possibility of using summary level genetic data for disease mapping and selection detection through local ancestry methods.

References

1. Arriaga-MaccKenzie, I.S., Matesi, G., Chen, S., Ronco, A., Marker, K.M., Hall, J.R., Scherenberg, R., Khajeh-Sharafabadi, M., Wu, Y., Gignoux, C.R., Null, M. Hendricks, A.E. Summix: A method for detecting and adjusting for population structure in genetic summary data. AJHG 108 (7), 1270-1282 (2021). https://doi.org/10.1016/j.ajhg.2021.05.016

2. Karczewski, K.J., Francioli, L.C., Tiao, G. et al. The mutational constraint spectrum quantified from variation in 141,456 humans. Nature 581, 434–443 (2020). https://doi.org/10.1038/s41586-020-2308-7

3. The 1000 Genomes Project Consortium., Corresponding authors., Auton, A. et al. A global reference for human genetic variation. Nature 526, 68–74 (2015). https://doi.org/10.1038/nature15393

4. Bergstrom, A., McCarthy, S.A., Hui, R., Almarri, M.A., Ayub, Q., Danecek, P., Chen, Y., Felkel, S., Hallast, P., Kamm, J., Blanche, H., Deleuze, J.F., Cann, H., Mallick, S., Reich, D., Sandhu, M.S., Skoglund, P., Scally, A., Xue, Y., Durbin, R., Tyler-Smith, C. Insights into human genetic variation and population history from 929 diverse genomes. Science 367 (6484): eaay5012. https://doi.org/10.1126/science.aay5012

5. Jin, W., Xu, S., Wang, H., Yu, Y., Shen, Y., Wu, B., & Jin, L. (2012). Genome-wide detection of natural selection in African Americans pre- and post-admixture. Genome research, 22(3), 519–527. https://doi.org/10.1101/gr.124784.111

Code Appendix

```r
knitr::opts_chunk$set(echo = TRUE,
                      warning = FALSE,
                      cache = TRUE,
                      cache.lazy = FALSE)

library(kableExtra) # load for tables

# load packages needed for LA method
library(nloptr)
library(devtools)
library(tidyverse)
library(Summix)
library(scales)
library(DescTools)
library(magrittr)
library(cowplot)
library(progress)
setwd("C:/Users/HayBa/Documents/Graduate School/Year 2/Fall/BIOS6611/")

# then read in the needed data
chr19_v3 <- read_csv("C:/Users/HayBa/Documents/Graduate School/Local Ancestry/Chr19_data_forSimulations.
chr19_v3$...1 <- NULL
chr19_v3$CHR <- 19

doSimulationsFull <- function(numSimulations, chrData, N_afr, N_eur, pi_afr, pi_afr_block = NA) {
  simulations <- vector(mode = "list", length = numSimulations)
  pb <- progress_bar$new(total = numSimulations)

  for(rep in 1:numSimulations) {
    af_afr1 <- chrData$AF_AFR
    af_eur1 <- chrData$AF_EUR

    af_sim_afr <- rep(NA, length(af_afr1))
    af_sim_eur <- rep(NA, length(af_afr1))
    for(s in 1:length(af_afr1)) {
      afr_sim <- rmultinom(1, N_afr, c(af_afr1[s]^2,
                                       2*af_afr1[s]*(1-af_afr1[s]),
                                       (1-af_afr1[s])^2))
      eur_sim <- rmultinom(1, N_eur, c(af_eur1[s]^2,
                                       2*af_eur1[s]*(1-af_eur1[s]),
                                       (1-af_eur1[s])^2))
      af_sim_afr[s] <- ((afr_sim[1]*2) +
                          (afr_sim)[2])/(2*N_afr)
      af_sim_eur[s] <- ((eur_sim[1]*2) + eur_sim[2])/(2*N_eur)
    }
    af_sim <- af_sim_afr*pi_afr + af_sim_eur*(1-pi_afr)

    sim1 <- chrData
    sim1$SIM <- af_sim
    if(!is.logical(pi_afr_block)) {
      indices <- which(sim1$POS > 10000000 & sim1$POS < 10300000)
      sim1[indices,]$SIM <- af_sim_afr[indices]*pi_afr_block + af_sim_eur[indices]*(1-pi_afr_block)
    }
    sim1_data <- sim1 %>% select(CHR, A1, A2, RSID, POS, SIM, AF_AFR,
                                 AF_IAM, AF_EAS,
                                 AF_EUR, AF_SAS)
    sim1_data$CHR <-19

    simulations[[rep]] <- sim1_data
    pb$tick()
  }
  return(simulations)
}
# first set the seed
set.seed(6611)

# create the simulated data sets
simData <- doSimulationsFull(numSimulations = 100,
                             chrData = chr19_v3,
                             N_afr = 1000,
                             N_eur = 1000,
                             pi_afr = 0.85)

# apply the local ancestry function, which also does the bootstrapping
# will be running these in parallel using 8 cores
library(parallel)
cl <- makeCluster(8, type = "PSOCK")
# source code for the local ancestry method is being loaded via source()
# code is included as a separate file
# code will be uploaded to github with publication in the future
clusterEvalQ(cl, source("C:/Users/HayBa/Documents/Graduate School/Local Ancestry/LaSourceCode.R"))

simRes <- parLapply(cl, simData, function(x)
  summix_LA_window(data = x,
                   chromosome = 19,
                   reference = c("AF_AFR", "AF_EUR"),
                   observed = "SIM",
                   type = "variants",
                   windowSize = 5000,
                   windowOverlap = 100))
stopCluster(cl)
# calculate accuracy (bias) = simulated - estimated
# put it into a matrix where each row is a rep and each column
# is a block
bias <- matrix(unlist(lapply(simRes, function(x)
  0.85 - x$results$AF_AFR)), nrow = 100, ncol = 98)

# get the standard error results and put them in same format
se <- matrix(unlist(lapply(simRes, function(x)
  x$localSE$AF_AFR)), nrow = 100, ncol = 98)
source("C:/Users/HayBa/Documents/Graduate School/Local Ancestry/LaSourceCode.R")
# first load in the data
chr2 <- read_csv("C:/Users/HayBa/Documents/Graduate School/Local Ancestry/chr2_subset_forFinalProject.cs
chr2$CHR <- 2

# apply the selection detection method
chr2_res <- summix_LA_fastCatch(data = chr2,
                                chromosome = 2,
                                reference = c("AF_AFR", "AF_EUR"),
                                observed = "AFafr",
                                type = "variants",
                                minVariants = 3500,
                                maxVariants = 4000,
                                maxStepSize = 1000,
                                diffMethod = "percent",
                                diffThreshold = 0.015)

# Make table with power and T1E results
tab_res <- matrix(c(20744, 840, 51, 34029, 621, 149, 2, 3, 4), nrow=3, ncol=3)
rownames(tab_res) <- c('gnomAD', '1000G', 'HGDP')

# Create table of results
kbl(tab_res,
    caption = 'Summary of data used in this project',
    booktabs=T, align='cc',
    col.names = c('AFR genomes','EUR genomes', 'Ref'))  %>%
    collapse_rows(columns=1, latex_hline='major', valign='middle') %>%
    kable_styling(latex_options = "HOLD_position")
# Make table with power and T1E results
tab_res2 <- matrix(c(mean(bias), mean(se)), nrow=1, ncol=2)

# Create table of results
kbl(tab_res2,
    caption = 'Accuracy and Precision of Estimates',
    booktabs=T, align='cc',
    col.names = c('Bias','Standard Error'))  %>%
    collapse_rows(columns=1, latex_hline='major', valign='middle') %>%
    kable_styling(latex_options = "HOLD_position")

# first need to put bias data into a dataframe
bias_df <- data.frame(bias = as.vector(bias),
                      rep = rep(1:100, 98))
p1 <- ggplot(data = bias_df, aes(x = rep, y = bias)) +
  geom_point(position = position_jitter(h = 0.00001, w=0.00001),
             size = 1) +
  theme_bw() + ylim(c(-0.005, 0.005)) +
  labs(title = "Bias of AFR Estimates",
       subtitle = "Bias = 0.85 - AF_AFR") +
  xlab("Simulation Replicate") +
  ylab("Bias") +
  geom_hline(yintercept = 0, col = "red", linetype = "dashed")

# want a df of just the AF_AFR estimates
afr_df <- data.frame(AFR = unlist(lapply(simRes, function(x)
  x$results$AF_AFR)))
p2 <- ggplot(data = afr_df, aes(x = AFR)) +
  geom_histogram(bins = 100) +
  geom_density(col = "blue") +
  theme_bw() + xlab("AFR Estimate") +
  ggtitle("AFR Estimates") +
  geom_vline(xintercept = 0.85, col = "red",
             linetype = "dashed") +
  xlim(c(0.845, 0.855))

plot_grid(plotlist = list(p2, p1), ncol = 1)

# first put it into dataframe for ggplot
se_df <- data.frame(se = as.vector(t(se)),
                      block = rep(1:98, 100))

p3 <- ggplot(se_df, aes(x = as.factor(block), y = se)) +
  geom_boxplot() +
  ggtitle("Standard Error of each block for 100 Replicates") +
  xlab("Block along Chromosome") +
  ylab("Standard Error of Ancestry Estimate") +
  theme(axis.text.x = element_text(angle = 90, size = 3.5))
p3
# look at the results that are sig different by bonferroni corrected p-value
real_res <- chr2_res$results %>%
  filter(p.AF_AFR.AllBootSE < (0.05/nrow(chr2_res$results))) %>%
  select(Start_Pos, End_Pos, AF_AFR, AF_EUR,
         p.AF_AFR.AllBootSE)

rownames(real_res) <- c("1")

kbl(real_res,
    caption = 'Significant Selection Signals from Chr 2',
    booktabs=T, align='cc',
    col.names = c("Start",
                            "End", "AFR", "EUR",
                            "P-val"))  %>%
    collapse_rows(columns=1, latex_hline='major', valign='middle') %>%
    kable_styling(latex_options = "HOLD_position",
                  full_width = F)
```