Population Genomics: background and tools

Haplotype-based methods for demography

Garrett Hellenthal, University College London

25/05/2017

For this practical, we will be applying the statistical software `GLOBETROTTER` and `PSMC` to simulated individuals in order to infer demographic history. In particlar we will infer admixture events and past population sizes at various time points.

1 Inferring admixture: GLOBETROTTER

Here we will use the same dataset as in the "Clustering Algorithms" practical. I.e. we will use a subset of dataset explored in [1], again working only with chromosome 22 (6,812 SNPs) and the following populations:

Population	**Country**	**Region**	**number of individuals**
Balochi	Pakistan	Central South Asia	21
BantuKenya	Kenya	Africa	11
BantuSouthAfrica	South Africa	Africa	8
Burusho	Pakistan	Central South Asia	25
English	Britain	Europe	6
HanNchina	China	East Asia	10
Kalash	Pakistan	Central South Asia	23
Makrani	Pakistan	Central South Asia	22
Mandenka	Senegal	Africa	22
MbutiPygmy	Congo	Africa	13
Mongola	Mongolia	East Asia	10
NorthItalian	Italy	Europe	12
Orcadian	Britain	Europe	15
Pathan	Pakistan	Central South Asia	22
Sardinian	Italy	Europe	28
Tuscan	Italy	Europe	8
Total			256

The aim here is to see how well `GLOBETROTTER` can reconstruct an admixture event in the simulated "population" described in the "Clustering Algorithms" practical. This simulated group consists of 20 individuals descending from an admixture event occurring 30 generations ago, where 80% of the DNA was contributed from present-day Brahui individuals (from Pakistan, Central South Asia) and the remaining 20% from present-day Yoruba individuals (from Nigeria, Africa). To identify this admixture event, we will use the 16 populations above as surrogates to the admixing sources.

First we will apply `ChromoPainterv2` [3] to these data, but in a slightly different way than before in order to detect admixture. (This is described in detail in Section 8.2 of the `ChromoPainterv2` user manual, but we have already done many of these steps

in the earlier practical.) Navigate to your folder containing ChromoPainterv2 and the CHROMOPAINTER input files you used in the "Clustering Algorithms" practical, and type:

```
./ChromoPainterv2 -g example/BrahuiYorubaSimulationChrom22.haplotypes
-r example/BrahuiYorubaSimulationChrom22.recomrates
-t example/BrahuiYorubaSimulation.idfile.txt
-f BrahuiYorubaSimulation.poplistReduced.txt 0 0
-o example/BrahuiYorubaSimulationAdmixtureChrom22
```

This is very similar to the previous command used in "Clustering Algorithms", but changes the output name (specified by "-o") and, importantly, removes the "-a 0 0". Thus this uses the file BrahuiYorubaSimulation.poplistReduced.txt to determine which populations to paint and which to use as donors for this painting. Looking at BrahuiYorubaSimulation.poplistReduced.txt, we see that the population BrahuiYorubaSimulation is the only recipient population (R), while other populations are specified as donors (D). This means each BrahuiYorubaSimulation individual will be painted using all individuals from all other listed populations as donors. This is the same as yesterday's painting, except we do NOT allow the BrahuiYorubaSimulation individuals to be painted using themselves as donors. This is because we need to identify specific DNA segments inherited from admixing sources in order to generate the "coancestry curves" used to date admixture. When doing so, segments that "best match" to (i.e. are painted by) other BrahuiYorubaSimulation individuals would simply be discarded, because GLOBETROTTER does not allow a population to be an admixture source for itself, and so we would throw away information.

Next we will run GLOBETROTTER to infer admixture, using the painting samples output from ChromoPainterv2 (once it has finished!). Unzip and extract GLOBETROTTER:

```
gunzip GLOBETROTTER.tar.gz
tar -xvf GLOBETROTTER.tar
```

Next compile with:

```
R CMD SHLIB -o GLOBETROTTERCompanion.so GLOBETROTTERCompanion.c -lz
```

To run GLOBETROTTER, you have to specify three files:

(I) The parameter input file, which describes all settings, including which population to detect admixture in and which populations to use as ancestry surrogates. The file we will use here is BrahuiYorubaSimulationAdmixture.paramfile.txt.

(II) The painting samples file, which points to the ChromoPainterv2 output file(s) containing the painting samples of the putatively admixed population. The file we will use here is BrahuiYorubaSimulationAdmixture.samplesfile.txt.

(III) The recombination rate file, which points to the recombination rate file(s) used when running ChromoPainterv2. The file we will use here is BrahuiYorubaSimulationAdmixture.recomfile.txt.

File (II) will point to the `example/BrahuiYorubaSimulationAdmixtureChrom22.samples.out` file we just made using `ChromoPainterv2`, which contains 10 sampled paintings for each haplotype of each individual from our simulated admixture population. File (III) will point to the `example/BrahuiYorubaSimulationChrom22.recomrates` we used when generating these paintings.

For file (I), we have specified the input file `BrahuiYorubaSimulation.idfile.txt` we used when running `ChromoPainterv2` above, as well as the output filenames (`save.file.XX`). We have also specified the painting we made during the "Clustering Algorithms" practical, which painted the target population and each of the surrogate populations using the same set of donors. (In this case the donor populations – `copyvector.popnames` – are the same as the surrogate populations – `surrogate.popnames`.) This information is used in the linear model we discussed in the lecture. The other parameters specify ways to run `GLOBETROTTER`. Most of these will likely not be changed, except for the first 3, which specify whether to infer dates and admixture proportions (`prop.ind`) and/or bootstrap re-sample to determine uncertainty in date estimation (`bootstrap.date.ind`), and/or whether to standardize estimates by a "NULL" individual to help eliminate spurious signals of admixture (`null.ind`). (As long as you are detecting admixture in $\geq$3 individuals, I highly recommend doing the latter, as we will do below.)

To run `GLOBETROTTER` under these settings type:

```
R < GLOBETROTTER.R BrahuiYorubaSimulationAdmixture.paramfile.txt
BrahuiYorubaSimulationAdmixture.samplesfile.txt
BrahuiYorubaSimulationAdmixture.recomfile.txt --no-save > output.out
```

It will take a couple minutes to run. You can follow progress by typing:

```
pic output.out
```

Once finished, the following output files will be produced, each in the `example/` directory:

(I) `BrahuiYorubaSimulationAdmixed.globetrotter.main.txt` – this gives the results, including `GLOBETROTTER`'s "best-guess" conclusion regarding admixture and the inferred dates and proportions

(II) `BrahuiYorubaSimulationAdmixed.globetrotter.main.pdf` – this gives you "coancestry curves" for every combination of surrogate populations that are inferred to have contributed >0.1% ancestry to the target population

(III) `BrahuiYorubaSimulationAdmixed.globetrotter.main_curves.txt` – this gives all of the raw data used to produce the curves in (II), in case you want to make your own plots

We will now run `GLOBETROTTER` again, but incorporating a "NULL" individual that attempts to account for any signal in LD decay that is *not* explained by genuine admixture. To do so, in `BrahuiYorubaSimulationAdmixture.paramfile.txt` change `null.ind` to "1" and change `save.file.main` to "example/BrahuiYorubaSimulationAdmixed.globetrotter.mainNULL". Then run `GLOBETROTTER`

again, exactly as above. This will make each of the output files as above, but with a `xx.mainNULLxx` in the filename.

Now answer the following questions:

1. From the `GLOBETROTTER` user manual, what do the different measures in `BrahuiYorubaSimulationAdmixed.globetrotter.main.txt` tell you?
2. What is `GLOBTROTTER`'s conclusion about admixture in this application?
3. How do you interpret the coancestry curves in `BrahuiYorubaSimulationAdmixed.globetrotter.main.pdf`? Do the results from `BrahuiYorubaSimulationAdmixed.globetrotter.main.txt` make sense in light of these coancestry curves?
4. Do results change when incorporating the "NULL" individual?

If more time, change `bootstrap.date.ind` to "1", which will provide 20 bootstrap resample estimates of the date. Then try changing the surrogates (i.e. remove some populations listed in `surrogate.popnames`). How do results change?

2 Inferring population size changes over time: PSMC

Now we will use `PSMC` [4] to infer the demographic history of single samples of individuals. Here we will simulate data using the program `ms` [5], following the instructions at http://willyrv.github.io/tutorials/bioinformatics/ms-psmc.html by Willy Rodriguez. **To be able to run these programs, you have to add `-Y` when using `ssh` to log into the system, i.e.:**

```
ssh -Y username@elixir-it-trein.recas.ba.infn.it
```

First unpack both of these programs:

```
gunzip psmc-0.6.5.tar.gz
tar -xvf psmc-0.6.5.tar
gunzip ms.tar.gz
tar -xvf ms.tar
```

To compile `psmc`, change directory to `psmc-0.6.5/` and then compile with:

```
make
```

Then change directory again to `utils/` and compile with:

```
make
```

Then leave this `psmc-0.6.5/` directory and change to the `msdir/` directory to compile `ms`:

```
gcc -O3 -o ms ms.c streec.c rand1.c -lm
```

Navigate out of `msdir/`. We will first use `ms` to simulate some data:

```
msdir/./ms 2 100 -t 30000 -r 6000 30000000 -eN 0.01 0.1 -eN 0.06 1 -eN
0.2 0.5 -eN 1 1 -eN 2 2 -p 8 > sim1.ms
```

This specifies that you will run 100 simulations, each consisting of 2 haplotypes (i.e. one individual) simulated over a 30Mb region with uniform mutation and recombination rates of 1.0/Mb and 0.2/Mb, respectively. The "`-eN t x`" parameters specify the population demography of this population, with `t` the time (in units of *4N0* generations) at which the population's size becomes `x` times that of the present-day population size. Here time goes from present to past, so that "`-eN 0.01 0.1`" means that the population shrinks to 10% of its present-day size at `0.01x4xN0` generations ago (e.g. 400 generations ago if *N0*=10K), continuing at this size until it hits the next time point specified by `-eN` (which is `0.06` in this case).

Next we will convert this `ms` output to `psmc` input, using the program `./ms2psmcfa.py`:

```
./ms2psmcfa.py ./sim1.ms > sim1.psmcfa
```

Now run `psmc`:

```
psmc-0.6.5/./psmc -N25 -t15 -r5 -p 4+25*2+4+6 -o
example/dem_history_sim1.psmc ./sim1.psmcfa
```

Here `-N` is the maximum number of MCMC iterations, `-t` is the maximum coalescent time, `-r` is the initial ratio of mutation to recombination, and `-o` is the output file. `-p` specifies that there will be 28 parameters over 64 time intervals, with the first parameter spanning 4 of these time intervals, the next 25 parameters each spanning 2 time intervals, the 27th parameter spanning 4 time intervals and the 28th spanning 6 time intervals. (This is the recommended command line for modern humans.)

This takes a long time to run (≈3 hours), so I have put the output `dem_history_sim1.psmc` into your main directory. We can use the program `./plot_results.py` to plot this output:

```
./plot_results.py
```

This will make the file `PSMCExampleFigure.pdf` containing the figure.

Now answer the following questions:

1. How do you interpret the plot?
2. How accurate is the inference?
3. When does the inference go wrong?

References

[1] G. Hellenthal, G.B.J. Busby, G. Band, J.F. Wilson, C. Capelli, D. Falush, and S. Myers. A genetic atlas of human admixture history. *Science*, 343:747–751, 2014.

[2] D.H. Alexander, J. Novembre, and K. Lange. Fast model-based estimation of ancestry in unrelated individuals. *Genome Research*, 19:1655–1664, 2009.

[3] D.J. Lawson, G. Hellenthal, S. Myers, and D. Falush. Inference of population structure using dense haplotype data. *PLoS Genet*, 8(1):e1002453, 2012.

[4] H. Li and R. Durbin. Inference of human population history from individual whole-genome sequences. *Nature*, 475:493–496, 2011.

[5] R.R. Hudson. Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics*, 18(2):337–8, 2002.