

ColabADMIXTOOLS

Quickstart Guide (2nd Edition)

Written by Florio

Revised January 13th, 2025

Table of Contents

- [Table of Contents](#)
- [What is ADMIXTOOLS?](#)
- [Useful Links and Documentation:](#)
- [Getting Started](#)
 - [\[1\] Install Software](#)
 - [\[2\] Your DNA File Upload](#)
 - [\[3\] Dataset Selection](#)
 - [\[4\] Merging Yourself with the AADR](#)
 - [\[5\] AT2 qpAdm Prep and Running](#)
 - [\[6\] AT1 qpAdm Prep and Running](#)
 - [\[6b\] Data Analysis and Visualization](#)
- [Modeling Yourself with the AADR](#)
 - [\[1\] Install Software](#)
 - [\[2\] Your DNA File Upload](#)
 - [\[3\] Dataset Selection](#)
 - [\[4\] Merging Yourself with the AADR](#)
 - [\[5\] AT2 qpAdm Prep and Running](#)
 - [\[6\] AT1 qpAdm Prep and Running](#)
 - [\[7\] Mounting Drive and Saving Compressed Files](#)
- [Utilities](#)
 - [Extracting a Sample and Converting to 23andMe Format](#)
 - [Extracting Population Groups and Plotting PCA](#)
 - [Computing Fst](#)
 - [Plotting Admixture Graphs](#)
- [General Troubleshooting](#)
- [Acknowledgements](#)

What is ADMIXTOOLS?

ADMIXTOOLS is a powerful software suite designed for population genetics research. It provides tools to analyze genetic data and uncover ancestry, admixture, and population structure patterns in humans and other species. These tools have been widely used in academic studies and are now increasingly accessible to hobbyists interested in exploring ancestry using genetic datasets.

ADMIXTOOLS offers two main versions:

ADMIXTOOLS 1:

- The original version was developed to perform tests such as f-statistics (e.g., f3, f4, D-statistics) and qpAdm modeling. It is written in C and is highly efficient for standard analyses of admixture and phylogeny.

ADMIXTOOLS 2:

- The updated version, written in R, is more user-friendly and includes improved workflows for hobbyists and researchers alike. It integrates modern computational tools, making processing and visualizing genetic data easier.

ADMIXTOOLS is most commonly used to:

- Detect gene flow (admixture) between populations.
- Test hypotheses about relationships among populations.
- Build and refine population phylogenies.
- Analyze ancient DNA data to explore human prehistory.

ADMIXTOOLS is especially suited for those working with genome-wide data, such as SNP arrays or whole-genome sequences. By leveraging its statistical methods, users can uncover fascinating stories about genetic history and population interactions.

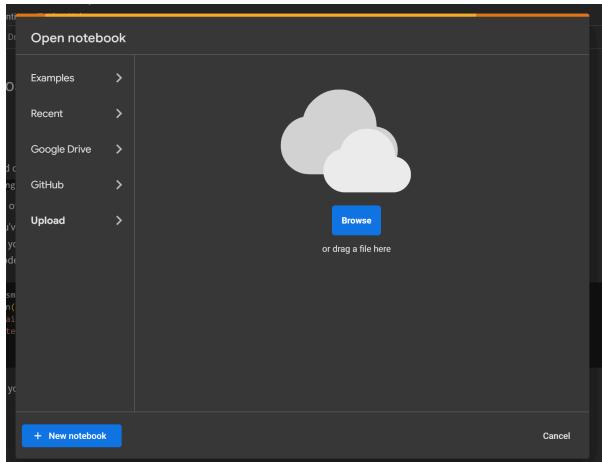
Useful Links and Documentation:

- [Ancient Admixture in Human History | Genetics | Oxford Academic](#)
- [Supplemental Material for Harney et al., 2020](#)
- [Assessing the performance of qpAdm: a statistical tool for studying population admixture | Genetics | Oxford Academic](#)
- [ADMIXTOOLS 2 Tutorial](#)
- [qpAdm best practices and common pitfalls | Indo-European.eu](#)
- [Testing times: disentangling admixture histories in recent and complex demographies using ancient DNA | Genetics | Oxford Academic](#)

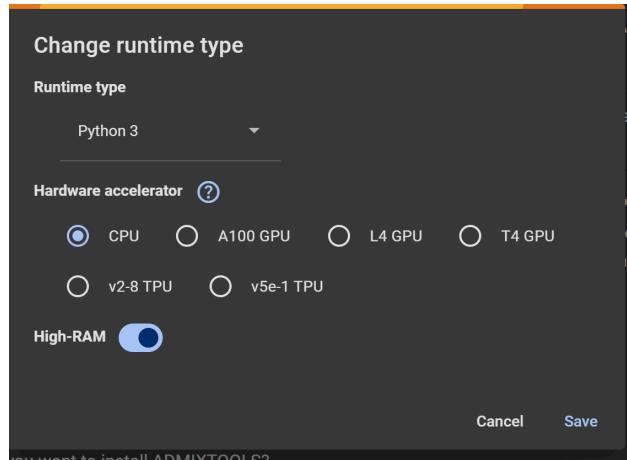
Getting Started

Go to [Colab](#). You will need a Google account to access this tool. Preferably at least 2 GB of Drive space if you want to merge, store your merged files, and return to modeling.

Upload the .ipynb (interactive python notebook) to Google Colab from your device file explorer.

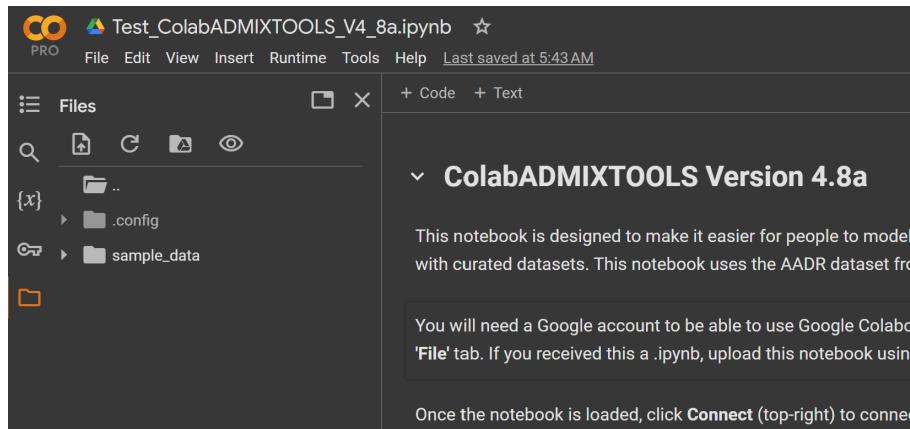


Go to Connect and change the runtime type, make sure you are connected to CPU and Python 3. There is an R environment option but we can bypass this later with a python package called rpy2. A standard free Colab runtime gives you a 12 GB CPU. I have ColabPro so I have access to 25 and 50 GB CPUs.



For this guide, I will be using Version 4.8a of my ColabADMXTOOLS notebook. I cannot guarantee the same performance for older versions that may be circulating.

I like having the Colab File Explorer open on the left at all times to track my files. By default, our Working Directory will be the **/content/** folder. All of our intermediate and final files will be saved there. Colab runs on a Linux environment so it is best to at least understand [Linux filesystem](#) and filepath naming conventions going forward.



[1] Install Software

Time: Estimated 7 minutes and 32 seconds (both AT1 and AT2)

Go ahead and Click on the Play Button for the first Code Cell. First, we will ensure rpy2, PLINK, and R are ready to go in the environment. Then we will install both AT1 and AT2. If for whatever reason you wish not to have one or the other, you may uncheck the box. If you are curious about

the underlying code, feel free to click on Show Code. Double-click outside the code to hide the code.

```

— R CMD build
* checking for file '/tmp/Rtmpv8lELm/remotes3993480e9cf
* preparing 'admixtools':
* checking DESCRIPTION meta-information ... OK
* cleaning src
* checking for LF line-endings in source and make files
* checking for empty or unneeded directories
* building 'admixtools_2.0.8.tar.gz'

ADMXTOOLS2 installation complete!
Setup complete!

WARNING:rpy2.rinterface_lib.callbacks:R[write to console]:
WARNING:rpy2.rinterface_lib.callbacks:R[write to console]: The downloaded source packages
'/tmp/RtmpoxReqQ/downloaded_packages'
*** WARNING:rpy2.rinterface_lib.callbacks:R[write to console]:
WARNING:rpy2.rinterface_lib.callbacks:R[write to console]:

WARNING:rpy2.rinterface_lib.callbacks:R[write to console]: Downloading GitHub repo uqrmai

These packages have more recent versions available.
It is recommended to update all of them.
Which would you like to update?

1: All
2: CRAN packages only
3: None
4: curl (6.0.1 -> 6.1.0) [CRAN]

Enter one or more numbers, or an empty line to skip updates: 3

```

If you get a prompt to update something, you can enter the number 3 and press enter to skip dependency updates for now. Performance should be the same regardless of what option you pick.

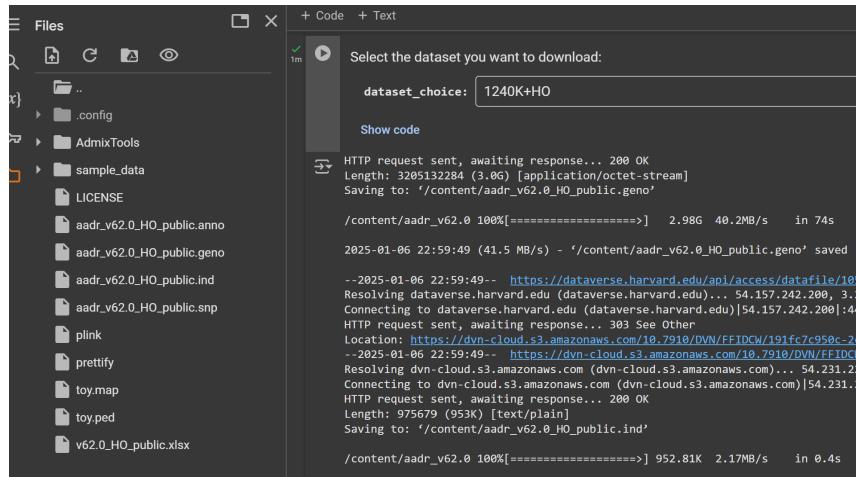
[2] Your DNA File Upload

This will be useful later on for those interested in modeling themselves but for now, let us skip it.

[3] Dataset Selection

Time: 45 seconds for 1240K+HO AADR, 1240K 1 min 18 seconds.

We will assume you know the difference between the two datasets. If you want to know more about the AADR, read [this paper](#). For simplicity, I will download the 1240K+HO as it has less coverage and thus quicker analysis steps.



[4] Merging Yourself with the AADR

Time: Merging with HO about 5-8 minutes. Merging with 1240K about 6-16 minutes.

Merging Overlap Reference: AncestryDNA overlaps with 1240K: about 390K snps. With HO: 130K snps. MyHeritage overlaps with 1240K: about 177K snps. With HO: about 60K snps. 23andMe overlaps with 1240K: about 140K snps. With HO: about 50K snps.

This will be useful later on if you want to model yourself, for now, we will skip this.

[5] AT2 qpAdm Prep and Running

On the left is a .xlsx file corresponding to the HO dataset samples. Double-click or right-click to download it to your local device and open it in Excel or Sheets. For population selection, we will be looking at the **Group ID** column. Feel free to browse through the rest of the samples and information stated there (e.g. locality, country, haplogroups, **Sample ID**, associated paper, coverage, etc). Alternatively, you can double-click on the .ind file and it should open on the right, with the Group ID of the population being the last column.

N1

	A	B	C	D	E	F	G	H	I	J	K	L	M	Group ID
1	Genetic ID (s)	Master ID	Skeletal code	Skeletal elem	Year	data for	Publication	at doi for	public:	Link to the mc Method for	D-Date mean in	Date standar	Full Date One Age at death	
2	00001.AG	Loeschbour	Loeschbour	2 tooth	2014	MathiesonNa	doi:10.1038/ENA-PRIEB82	Direct: Arch	8025	64 6221-5986	c2 34-47 yrs			Luxembourg_Mesolithic.AG
3	Kou01.SG	Kou01	Kou01	2 tooth	2021	ClementeC	doi:10.1016/ENA-PRIEB83	Context: Arch	4250	173 2600-2000	B.C.			Greece_Koufonisi_Cycladic_EBA.SG
4	Kou03.SG	Kou03	Kou03	...	2021	ClementeC	doi:10.1016/ENA-PRIEB83	Context: Arch	4250	173 2600-2000	B.C.			Greece_Koufonisi_Cycladic_EBA.SG
5	Log02.SG	Log02	Log02	...	2021	ClementeC	doi:10.1016/ENA-PRIEB83	Context: Arch	4250	173 2600-2000	B.C.			Greece_Logkas_MBA.SG
6	Log04.SG	Log04	Log04	...	2021	ClementeC	doi:10.1016/ENA-PRIEB83	Context: Arch	4250	173 2600-2000	B.C.			Greece_Logkas_MBA.SG
7	Mik15.SG	Mik15	Mik15	...	2021	ClementeC	doi:10.1016/ENA-PRIEB83	Context: Arch	4250	173 2600-2000	B.C.			Greece_Manika_EBA.SG
8	Pta08.SG	Pta08	Pta08	...	2021	ClementeC	doi:10.1016/ENA-PRIEB83	Context: Arch	4250	173 2600-2000	B.C.			Greece_Crete_Kephala_Petras.SG
9	cay004.SG	cay004	cay004	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Direct: Arch	8590	56 7650-7786	c2...			Turkey_Southeast_Cayonu_PPNS.G
10	cay007.SG	cay007	cay007	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Direct: Arch	9379	119 7650-7786	c2...			Turkey_Southeast_Cayonu_PPNS.G
11	cay008.SG	cay008	cay008	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Direct: Arch	8641	139 10463-45548	c2...			Turkey_Southeast_Cayonu_PPNS.G
12	cay011.SG	cay011	cay011	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Direct: Arch	8369	40 6475-6289	c2...			Turkey_Southeast_Cayonu_PPNS.G
13	cay012.SG	cay012	cay012	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	10100	953 9800-6500	B.C.			Turkey_Southeast_Cayonu_PPNS.G
14	cay013.SG	cay013	cay013	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	8591	177 8893-8327	c2...			Turkey_Southeast_Cayonu_PPNS.G
15	cay014.SG	cay014	cay014	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	8908	257 7470-6481	c2...			Turkey_Southeast_Cayonu_PPNS.G
16	cay015.SG	cay015	cay015	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	10100	953 9800-6500	B.C.			Turkey_Southeast_Cayonu_PPNS.G
17	cay016.SG	cay016	cay016	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	8591	177 8893-8327	c2...			Turkey_Southeast_Cayonu_PPNS.G
18	cay1820.SG	cay1820	cay1820	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	10100	953 9800-6500	B.C.			Turkey_Southeast_Cayonu_PPNS.G
19	cay022.SG	cay022	cay022	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	7303	98 7465-6481	c2...			Turkey_Southeast_Cayonu_PPNS.G
20	cay027.SG	cay027	cay027	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	10100	953 9800-6500	B.C.			Turkey_Southeast_Cayonu_PPNS.G
21	cay033.SG	cay033	cay033	...	2022	AltinpinK	doi:10.1126/ENA-PRIEB83	Context: Arch	10100	953 9800-6500	B.C.			Turkey_Southeast_Cayonu_PPNS.G
22	AV1.AG	AV1	petrous	...	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1360	28 549-640	c2IC...			Hungary_Avar_5_daughter_of_mother_A'
23	AV2.AG	AV2	AV2	petrous	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Direct (WARM)	1348	24 560-645	c2IC...			Italy_North_EarlyMedieval_Langobards
24	CL102.AG	CL102	CL102	petrous	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1345	14 580-630	c2E...			Italy_North_EarlyMedieval_Langobards
25	CL110.AG	CL110	CL110	...	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1345	14 580-630	c2E...			Italy_North_EarlyMedieval_Langobards
26	CL121.AG	CL121	CL121	petrous	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1345	14 580-630	c2E...			Italy_North_EarlyMedieval_Langobards
27	CL145.AG	CL145	CL145	petrous	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1345	14 580-630	c2E...			Italy_North_EarlyMedieval_Langobards
28	CL146.AG	CL146	CL146	petrous	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1345	14 580-630	c2E...			Italy_North_EarlyMedieval_Langobards
29	CL151.AG	CL151	CL151	petrous	2018	AmorinNatur	doi:10.1038/NCBI	sequen:Context: Arch	1345	14 580-630	c2E...			Italy_North_EarlyMedieval_Langobards

For this guide, I will be using my recently downloaded AADR HO dataset, so I will point to the prefix of that dataset, which we can manually type in or right-click on any of its files to copy the file path (either .geno, .ind, .snp file). So the file path should be **/content/aadr_v62.0_HO_public** as it is the prefix without any file extensions.

I have chosen **MXL.DG** as the Target population we wish to make a random model. I will explore modeling them with **IBS.DG,Nahua.DG,BantuSA.DG** as my Left/Source populations. I used: **Mbuti.DG,Israel_Natufian.AG,Israel_PPNB.AG,Russia_MA1_UP.SG,Turkey_Central_Boncuklu_PPN.AG,Turkey_Central_Pinarbasi_Epipaleolithic.AG,Morocco_Iberomaurusian.AG,Serbia_IronGates_Mesolithic.AG,Luxembourg_Mesolithic.DG,Russia_YuzhniyOleniyOstrov_Mesolithic.A** G as my Right/Reference populations. Please read the Documentation for details.

```

1 | FLORIO:Florio M Florio
2 | Ne30_genotyping_noUDG M China_AmurRiver_EarlyN
prefix: "/content/aadr_v62.0.HO_public"
left_poplist: "IBS.DG,Nahua.DG,BantuSA.DG"
right_poplist: "Mbuti.DG,Israel_Natufian.AG,Israel_PPNB.AG,Russia_MA1_UP.SG,Turkey_Central_Boncuklu_PPN.AG,Turkey_Central_Pinarbasi_Epipaleolithic.AG,Morocco_Iberomaurusian.AG,Serbia_IronGates_Mesolithic.AG,Luxembourg_Mesolithic.DG,Russia_YuzhniyOleniyOstrov_Mesolithic.A"
target: "MXL.DG"
weights_file: "weights1.csv"
popdrop_file: "popdrop1.csv"
detach_shinyjs: checked
allsnp: checked
Show code
Run qpAdm (AT2)

```

You may choose to rename the weights and popdrop files if you will be doing repeated runs and choose not to overwrite existing results. Moreover, leave **allsnp** and **detach_shinyjs** checkboxes checked unless you know what they do. For now, click Play on that code cell to save those

parameters, then let us click Play on the next code cell to Run AT2 qpAdm. We should get a bunch of information. At the bottom, there will be a table along with a popdrop table.

```
# A tibble: 3 × 5
  target left      weight      se      z
  <chr>  <chr>     <dbl>    <dbl>    <dbl>
1 MXL.DG IBS.DG    0.430   0.0110   39.1
2 MXL.DG Nahua.DG  0.534   0.0123   43.6
3 MXL.DG BantuSA.DG 0.0364  0.00347  10.5
```

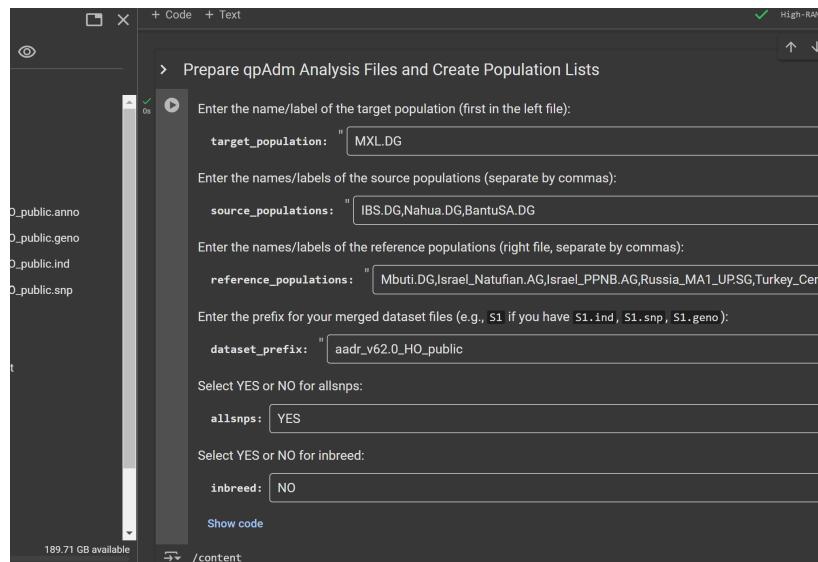
In short, this random model says MXL.DG can be modeled as having 43.0% IBS.DG with a SE of 1.1%, 53.4% Nahua.DG with a SE of 1.2%, and 3.6% BantuSA.DG with a SE of 0.35%. This makes sense considering Mexicans in Los Angeles are primarily of Iberian and Amerindian mixed ancestry with minor Sub-Saharan African admixture (*on average*). Now let's look at the popdrop table under it.

```
# A tibble: 7 × 14
  pat    wt    dof    chisq      p f4rank IBS.DG Nahua.DG BantuSA.DG feasible
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <lgl>
1 000     0     7  25.8  5.53e- 4     2  0.430   0.534   0.0364 TRUE
2 001     1     8 110.   4.00e-20     1  0.380   0.620    NA      TRUE
3 010     1     8 2425.   0          1  0.898   NA       0.102   TRUE
4 100     1     8 478.   3.82e-98     1  NA      1.01     -0.0127 FALSE
5 011     2     9 3982.   0          0  1       NA       NA      TRUE
6 101     2     9 483.   2.15e-98     0  NA      1       NA      TRUE
7 110     2     9 16542.   0          0  NA      NA       1      TRUE
```

The pat column tells you which Left/Source population is being used (with a 0) and what happens if it is not being used (with a 1). This table is great for diagnosing sample selection. As we can see, our initial model is feasible but the p-value of 5.53e-4 is relatively low. For reference, a p-value close to and above 0.05 is enough to reject the assumption that no admixture occurred. This is counter-intuitive, I know. In our scenario, the higher the p-value the better. Please read on Model/Sample Rotation within the Documentation links above for more information on interpreting and troubleshooting a model.

If you get a negative coefficient we can automatically reject the model. Since we have positive coefficients and it is somewhat feasible, we can go ahead and use AT1 to cross-reference our model.

[6] AT1 qpAdm Prep and Running



The parameter format is the same as before. Note pointing to the dataset prefix does not require `/content/` in this step. Leave `allsnps` and `inbreed` params the same for now. Let's click through these code cells. It is good practice to **Compute Fstats** for our population list so that we can remove and swap around Target, Left, and Right sources without having to re-compute their Fstats.

```
so for the given poplist.

> Compute F-Statistics
[13] Show code
pop: Russia_Mal_UP.SG hetrate: 0.000 valid snps: 384991 samples: 1 valid
pop: Turkey_Central_Boncuklu_PPN.AG hetrate: 0.003 valid snps: 506506 samples: 1 valid
pop: Turkey_Central_Pinarbasil_Epipaleolithic.AG hetrate: 0.000 valid snps: 420 samples: 1 valid
pop: Morocco_Iberomauritanian.AG hetrate: 0.075 valid snps: 508559 samples: 1 valid
pop: Serbia_IronGates_Mesolithic.AG hetrate: 0.109 valid snps: 524444 samples: 2 valid
pop: Luxembourg_Mesolithic.DG hetrate: 0.067 valid snps: 525000 samples: 2 valid
pop: Russia_YuzhnyVolenyOstrov_Mesolithic.AG hetrate: 0.099 valid snps: 519500 samples: 2 valid
pop: MXL.DG hetrate: 0.130 valid snps: 515700 samples: 62 valid
pop: IBS.DG hetrate: 0.130 valid snps: 515700 samples: 103 valid
pop: Nahua.DG hetrate: 0.097 valid snps: 515250 samples: 1 valid
pop: BantuSA.DG hetrate: 0.142 valid snps: 524866 samples: 5 valid
lambdaScale: 3.518
statistics multiplied by 1000
fst:
          M   I   I   R   T   T   M   S   L   R   M   I   N   B
          M   0   560  612  614  356  615  353  278  406  306  209  211  282  75
          I   560   0   926  942  608  936  646  527  696  566  475  456  566  582
          R   614  942  1800   0  658 1000  708  584  749  616  529  598  626  558
          T   356  608  658  693   0  653  428  263  314  312  217  183  215  305
          M   615  936 1800 1800 653   0  722  576  742  617  524  583  615  558
          I   353  646  709  735 428  722   0  342  491  378  283  271  370  287
          S   278  527  584  586 263  576  342   0  224  164  115  81  211  218
          L   406  690  749  746 424  742  491  224   0  341  265  235  362  345
          R   388  560  616  601 312  617  378  164  341   0  138  134  215  247
          M   209  475  529  515 217  524  283  115  265  138   0  40  39  147
          I   211  456  506  521 182  584  271  81  235  134  48   0  141  149
          N   282  566  620  588 316  615  378  211  362  215  39  141   0  222
```

Now let's run the AT1 qpAdm. Feel free to change the results output name if you do not want to overwrite previous runs. We should be able to see our weights/coefficients and popdrop table for our model in the results .txt file.

File Edit View Insert Runtime Tools Help All changes saved

After running the f... Double-click to open

[6b] Data

Please paste the path to your qpAdm result file:

```
results3.txt
```

Parse and V... [7] Mount

To avoid the length

```

results3.txt
right1.txt
toy.map
toy.ped

```

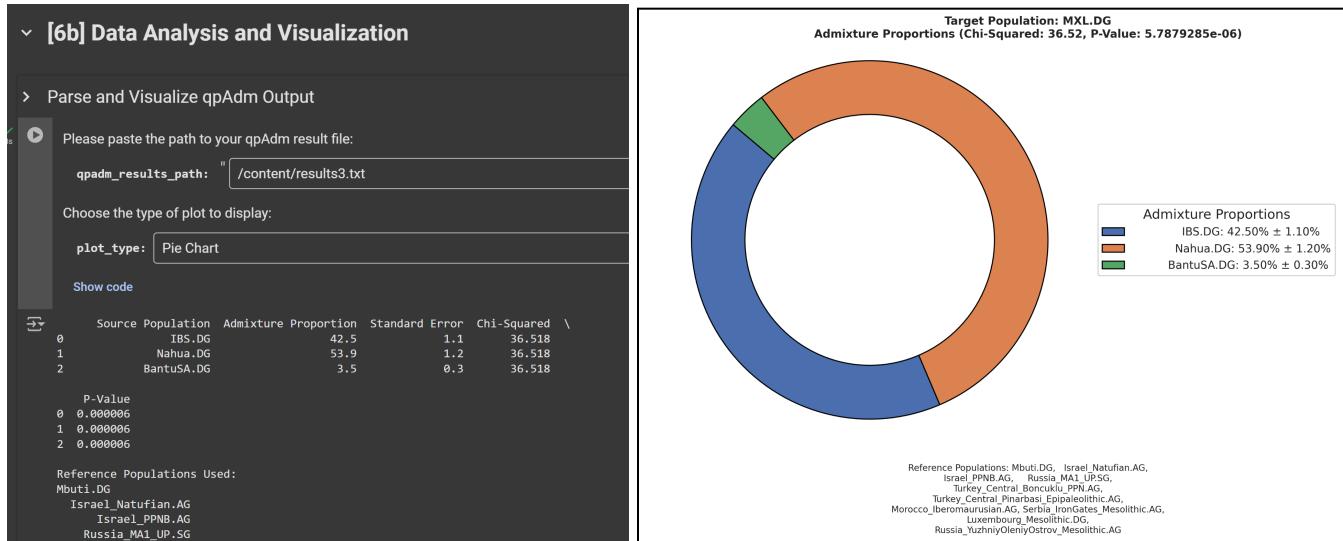
```

93 bootstrap sampling of F-coeffs: 1000
94 zzjmean 0.425 0.539 0.035
95 std. errors: 0.011 0.012 0.003
96
97 error covariance (* 1,000,000)
98 115 -124 9
99 -124 144 -20
100 9 -20 12
101
102
103 sum: MXL.DG 3 0.000006 0.425 0.539 0.035 115 -124
104 12
105
106 fixed pat wt dof chisq tail prob
107 000 0 7 36.518 5.78793e-06 0.425 0.539 0.035
108 001 1 8 118.304 7.42424e-22 0.385 0.615 0.000
109 010 1 8 2127.158 0 0.894 0.000 0.106
110 100 1 8 454.368 0 0.000 1.022 -0.022 infeasible
111 011 2 9 3435.820 0 1.000 0.000 0.000
112 101 2 9 443.366 0 0.000 1.000 0.000
113 110 2 9 15318.577 0 0.000 0.000 1.000
114 best pat: 000 5.78793e-06 -
115 best pat: 001 7.42424e-22 chi(nested): 81.786 p-value for nested model:
116 best pat: 101 7.50875e-90 chi(nested): 325.063 p-value for nested model:
117
118 coeffs: 0.425 0.539 0.035
119
120 ## dscore:: f_4(Base, Fit, Rbase, right2)

```

[6b] Data Analysis and Visualization

It can be troublesome to read through the results output file sometimes. In this section, we can point to our results file path and extract the necessary information we need for visualization (assuming non-negative percentages). For now, let's do a simple Pie Chart.



The weights/coefficients/percentages corroborate our AT2 model but the p-value dropped a bit. Technically it is feasible but we could improve this model with an improved Right list. For the instructive purposes of this guide, this random model will suffice.

Modeling Yourself with the AADR

Now that we understand the workflow of the notebook and the process. We can choose to model ourselves with the AADR. However, the overlap will vary. Generally, AncestryDNA files have the most overlap, followed by MyHeritage and 23andMe. It is possible to impute your raw file using the [DNAGenics imputation service](#) and increase your coverage; however, this statistical imputation and downstream models may not reflect true admixture.

Moreover, if you have tested with multiple kits you can combine them to maximize coverage with the AADR using DNAGenics DNA Kit Studio (see above).

If you have done WGS, then converting to a workable format is possible by using [WGS Extract](#), then converting that vcf to eigenstrat format (see [here](#) to go down that rabbit hole). I have never done WGS so cannot attest to the performance of this workflow.

[1] Install Software

First, let's start at the top of the notebook by double-checking that everything is installed and loaded into the environment correctly. If you are ever disconnected or receive an unexpected error, double-checking **Step 1** should be the first troubleshooting step you should perform after checking for typos and correct file paths.

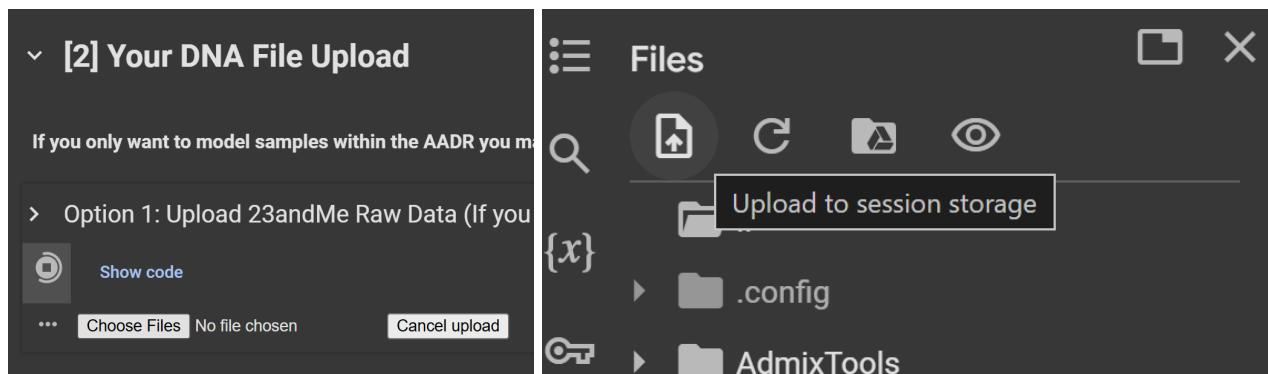
```
▼ [1] Install Software

▶ Select your preferences for the setup process
Do you want to install ADMIXTOOLS?
install_admixtools: 
Do you want to install ADMIXTOOLS2? (This will increase setup time.)
install_admixtools2: 
Show code

R is already installed.
rpy2 is already installed.
The rpy2.ipython extension is already loaded. To reload it, use:
%reload_ext rpy2.ipython
PLINK is already installed.
ADMIXTOOLS is already installed.
Checking for ADMIXTOOLS2 installation...
ADMIXTOOLS2 is already installed.
Setup complete!
```

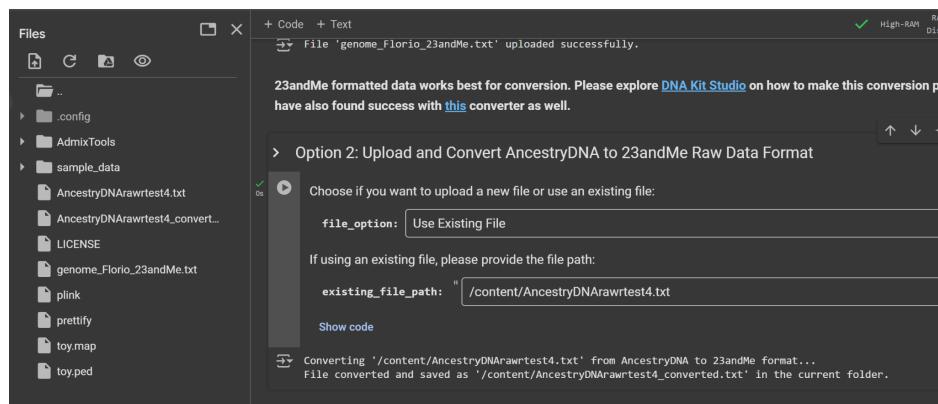
[2] Your DNA File Upload

Now let's get our raw file into the Colab environment. **Option 1** is straightforward for those with raw data already in 23andMe format. We simply Play the code cell, Choose Files, and then select our 23andMe formatted file using our local device file explorer. Alternatively, you can upload to this session manually (for any file from your local device) using the Colab File Explorer upload button.



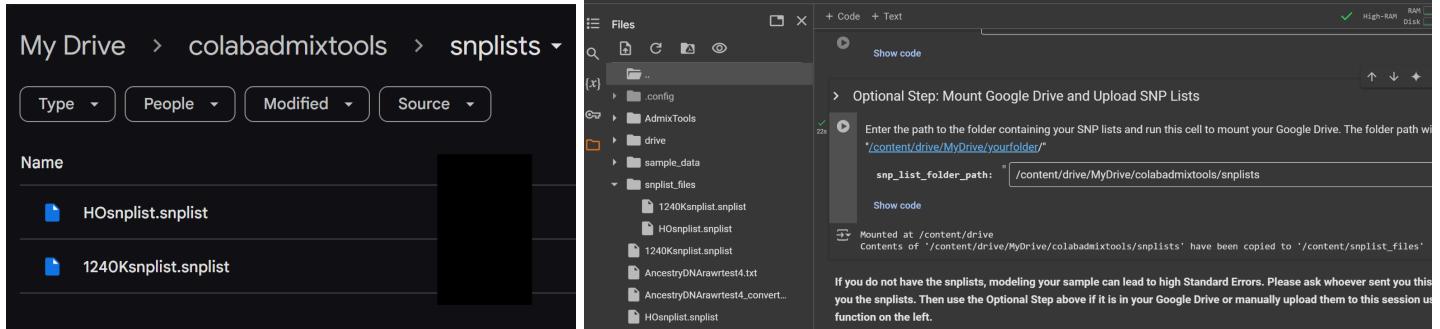
I mention [DNA Kit Studio](#) and [another converter](#) for other formatted files from other companies. Feel free to try those out if none of my converter scripts work for you.

If you have an AncestryDNA formatted file. You can go ahead and perform **Option 2**. In this example, I have manually uploaded an AncestryDNA file and pointed to the file path of the file for conversion. The 23andMe formatted file has `_converted` added to the end.

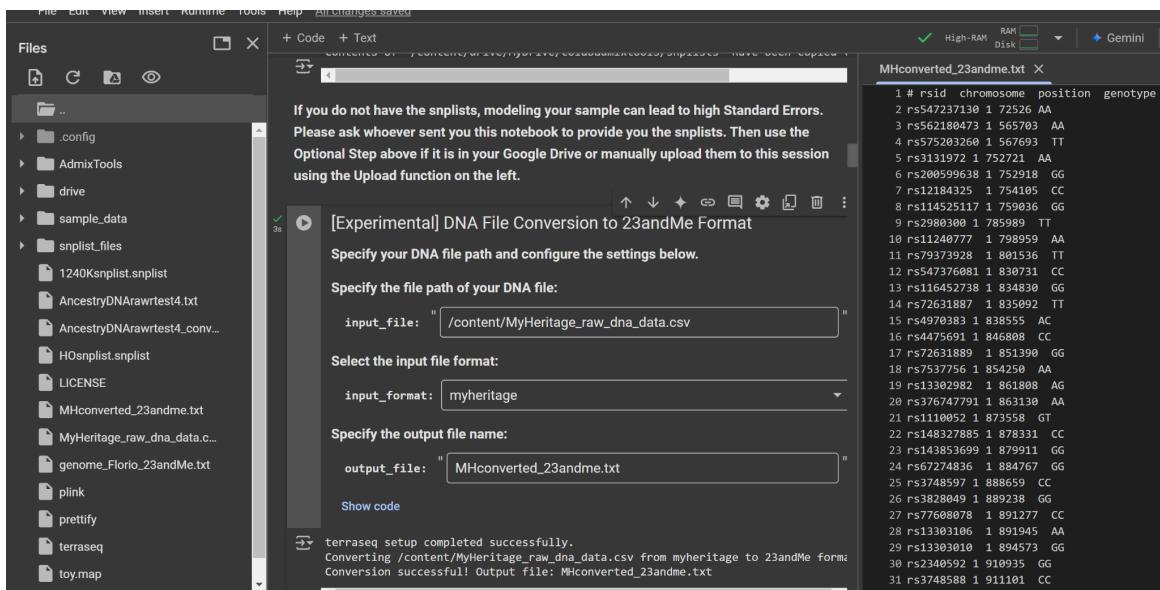


Let's skip **Option 3** for now. It will be useful later. Let's instead focus on uploading the 1240K and HO snplists into our Colab session. You can do so manually from your local device or using the **Optional Step** if they are in your `colabadminstools` folder in your Google Drive. Do so manually if you do not wish to Mount your Drive account, they will go to the `/content/` folder. If using the

script, it will be saved within `/content/` as a new folder called `/snplist_files/`. Either way is fine, it is recommended you trim your raw file to the appropriate snplist for best performance.



For those with MyHeritage or FTDNA formatted files, I have an experimental script that utilizes [terraseq](#) (courtesy of X) that may work for you if DNA Kit Studio does not. However, I cannot guarantee it will always work as expected. In this example, I manually uploaded an MH .csv file to my working directory (`/content/`) and pointed to it. Selected myheritage as the input format and chose **MHconverted_23andme.txt** as my desired output name for it.



Alright, now we have our raw file in 23andMe format.

[3] Dataset Selection

For this example, I will merge with the **1240+HO** dataset (or just **HO** for short). Unfortunately, in a free standard Colab Runtime, we have limited dedicated RAM so [merging with 1240K is only possible with ColabPro HighRAM runtimes at the moment](#). If you trust someone to merge you

with the 1240K on their ColabPro runtime, I recommend removing all personal identifiers in your raw file before sending it over. Normally, I request the person to send me their raw file as a zipped file and I save it to my **colabadminmixtools** folder in my Drive. Then I used **Step 2 Option 3**, point to that zipped file, and uncompress it into the Colab session. From there I perform all merging and use **Step 7** to compress their .geno, .ind, and .snp merged files into a single zipped folder in my Drive. From there I use an anonymous filesharing service like [file.io](#) to share it with them to upload to their Colab sessions and model themselves. Otherwise, you can pay for ColabPro to resolve RAM runtime issues.

```
+ Code + Text
The 1240K has more coverage than the 1240K+HO. However, the 1240K+HO has more modern samples. You can download one after the other if you would like to explore both datasets.

> Download 1240K or HO Datasets (AADR v62)

Select the dataset you want to download:
dataset_choice: 1240K+HO
Show code 1240K

HTTP request sent, = 1240K+HO
Length: 3205132284 (3.66 GB)
Saving to: '/content/aadr_v62.0_HO_public.geno'

</content/aadr_v62.0 100%[=====] 2.98G 36.1MB/s in 84s
2025-01-08 08:03:49 (36.6 MB/s) - '/content/aadr_v62.0_HO_public.geno' saved [3205132284/3205132284]

- 2025-01-08 08:03:49 -- https://dataverse.harvard.edu/api/access/datafile/10527429
Resolving dataverse.harvard.edu (dataverse.harvard.edu)... 3.232.162.235, 3.90.116.199, 54.157.242.200
Connecting to dataverse.harvard.edu (dataverse.harvard.edu)[3.232.162.235]:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://dnv-cloud.s3.amazonaws.com/19_7919/DVN/ETD/CW/191fc7c95c-7r8e27ffef354?response-content-disposition=attachment%
```

[4] Merging Yourself with the AADR

```
+ Code + Text
You can skip to Step 5 if you just want to model within the AADR. For now, only merging with the +HO dataset is possible
✓ because merging with the 1240K dataset requires more RAM with a standard Colab Runtime. If you (or someone else) has ColabPro, they can utilize the High-Ram runtime to get around this issue.

Step 1: Specify Inputs

Filepath to your 23andMe TXT File:
genome_filepath: "/content/genome_Florio_23andMe.txt"

Output Base Prefix: Define a name for intermediate and final files.
output_base: "Florio"

.fam File Parameters: These fields will auto-fill based on output_base, but you can edit sex_code as needed.
sex_code: '1'

Trim Using SNP List?
trim_with_snplist: 
If Trimming, specify:
snplist_path: "/content/HOsnplist.snplist"
```

Time: Merging with HO about 5-8 minutes. Merging with 1240K about 6-16 minutes.

Merging Overlap Reference: AncestryDNA overlaps with 1240K: about 390K snps. With HO: 130K snps. MyHeritage overlaps with 1240K: about 177K snps. With HO: about 60K snps. 23andMe overlaps with 1240K: about 140K snps. With HO: about 50K snps.

Alright, here I have pointed to the filepath for my uploaded 23andMe formatted raw file. Remember to upload the correct snplist to trim to and point to the correct .geno, .snp, and .ind files for the dataset you intend to merge with. Sex code 1 for Male, and 2 for Female, 0 to omit. Leave the **num_threads** parameter alone for now.

Trim Using SNP List?

`trim_with_snplist:`

If Trimming, specify:

`snplist_path: "/content/HOsnplist.snplist"`

Merge Dataset Parameters

Specify the file paths for the dataset you want to merge with:

`geno1_path: "/content/aadr_v62.0_HO_public.geno"`

`snp1_path: "/content/aadr_v62.0_HO_public.snp"`

`ind1_path: "/content/aadr_v62.0_HO_public.ind"`

Number of Threads for Mergeit:

`num_threads: 2`

Show code

```

Files + Code + Text
+ 2m
allele funny: rs7867 G A X X
allele funny: rs4145536 G T X X
allele funny: rs738371 C A X X
allele funny: rs1409993 A G X X
allele funny: rs5922152 T C X X
allele funny: rs1933800 C T X X
allele funny: rs1323223 A G X X
allele funny: rs13303711 C T X X
allele funny: rs35617575 C A X X
allele funny: rs9786855 C T X X
allele funny: rs9341284 G A X X
allele funny: rs9786021 C A X X
allele funny: rs13447370 C T X X
read 1073741824 bytes
read 2147483648 bytes
read 3205126797 bytes
packed geno read OK
end of inpack
packed geno read OK
end of inpack
numsnp input: 584131 53267
packedancestrymap output
numsnp output: 51919 numindivs: 21946

Histogram of checkmatch return codes
kode: 0 197 Allele mismatch
kode: 1 34520 SNP OK (no flip)
kode: 2 17399 SNP OK (flip)
total: 52116

##end of mergeit: 121.592 seconds cpu 490.652 Mbytes in use
Verifying merged files...
Merging completed successfully.

```

We can see the intermediate converted (23andMe to PLINK (PACKEDPED) format) files, the trimmed files (to the HO snplist), and the merged files (in PACKEDANCESTRYMAP format). The overlap of 51K snps is not great but you can improve this by merging with a MyHeritage and/or an AncestryDNA raw file.

```

Files + Code + Text
+ 2m
allele funny: rs143973988 C T X X
allele funny: rs12462449 C A X X
allele funny: rs2450509 A G X X
allele funny: rs7251341 A G X X
allele funny: rs112409210 T C X X
allele funny: rs6040222 A G X X
allele funny: rs685188 T G X X
allele funny: rs1133358 C A X X
allele funny: rs6025861 A G X X
allele funny: rs181376 T G X X
allele funny: rs394484 G A X X
allele funny: rs8129780 C T X X
allele funny: rs2837957 C T X X
allele funny: rs7867 G A X X
allele funny: rs4145536 G T X X
allele funny: rs738371 C A X X
allele funny: rs1409993 A G X X
allele funny: rs5922152 T C X X
allele funny: rs1933800 C T X X
allele funny: rs1323223 A G X X
allele funny: rs1303711 C T X X
allele funny: rs35617575 C A X X
allele funny: rs9786855 C T X X
allele funny: rs9341284 G A X X
allele funny: rs9786021 C A X X
allele funny: rs13447370 C T X X
read 1073741824 bytes
read 2147483648 bytes
read 3205126797 bytes
packed geno read OK
end of inpack
packed geno read OK
end of inpack
numsnp input: 584131 53267
packedancestrymap output
numsnp output: 51919 numindivs: 21946

Florio_merged.ind
21916 MBG010.AG F Germany_Hallstat_IronAge.AG
21917 MBG010.d.AG U Germany_Hallstat_IronAge.AG
21918 MBG011.AG U Germany_Hallstat_IronAge.AG
21919 MBG011.d.AG U Germany_Hallstat_IronAge.AG
21920 MBG012.AG M Germany_Hallstat_IronAge.AG
21921 MBG012.d.AG U Germany_Hallstat_IronAge.AG
21922 MBG013.AG M Germany_Hallstat_IronAge.AG
21923 MBG013.d.AG U Germany_Hallstat_IronAge.AG
21924 MBG014.AG M Germany_Hallstat_IronAge.AG
21925 MBG014.d.AG U Germany_Hallstat_IronAge.AG
21926 MBG015.AG F Germany_Hallstat_IronAge.AG
21927 MBG015.d.AG U Germany_Hallstat_IronAge.AG
21928 MBG016.AG M Germany_Hallstat_IronAge.AG
21929 MBG016.d.AG U Germany_Hallstat_IronAge.AG
21930 MBG017.AG M Germany_Hallstat_IronAge.AG
21931 MBG017.d.AG U Germany_Hallstat_IronAge.AG
21932 SCN001.AG F Germany_Hallstat_IronAge.AG
21933 SCN001.d.AG U Germany_Hallstat_IronAge.AG
21934 UKY001_v62.AG.SG M Russia_EastSiberia_Ustkyakhta_UP.AG.SG
21935 UKY001_v62.AG.M Russia_EastSiberia_Ustkyakhta_UP.AG
21936 UKY001_v62.SG M Russia_EastSiberia_Ustkyakhta_UP.SG
21937 KPT002_v62.AG.SG M Russia_Siberia_Lena_EBA.AG.SG
21938 KPT002_v62.AG.M Russia_Siberia_Lena_EBA.AG
21939 KPT002_v62.SG M Russia_Siberia_Lena_EBA.SG
21940 UKY001_v62_d.AG.SG M Russia_EastSiberia_Ustkyakhta_UP.AG.SG
21941 UKY001_v62_d.AG.M Russia_EastSiberia_Ustkyakhta_UP.AG
21942 UKY001_v62_d.SG M Russia_EastSiberia_Ustkyakhta_UP.SG
21943 KPT002_v62_d.AG.SG M Russia_Siberia_Lena_EBA.AG.SG
21944 KPT002_v62_d.AG.M Russia_Siberia_Lena_EBA.AG
21945 KPT002_v62_d.SG M Russia_Siberia_Lena_EBA.SG
21946 Florio:Florio M ???
```

We should now open our _merged .ind file by double-clicking on it in the File Explorer to the left and then scrolling down where we will find our sample identifier. We can rename the ??? and backspace (leaving one space between your sex identifier) to our desired sample name.

```

allele funny: rs143973988 C T X X
allele funny: rs12462449 C A X X
allele funny: rs2455069 A G X X
allele funny: rs7251341 A G X X
allele funny: rs112409210 T C X X
allele funny: rs6040222 A G X X
allele funny: rs6865188 T G X X
allele funny: rs6025861 A G X X
allele funny: rs113358 C A X X
allele funny: rs181376 T G X X
allele funny: rs394484 G A X X
allele funny: rs8129706 C T X X
allele funny: rs2837957 C T X X
allele funny: rs7867 G A X X
allele funny: rs4145536 G T X X
allele funny: rs738371 C A X X
allele funny: rs1409993 A G X X
allele funny: rs5922152 T C X X
allele funny: rs1933806 C T X X
allele funny: rs1323223 A G X X
allele funny: rs1303711 C T X X
allele funny: rs3561755 C A X X
allele funny: rs9786855 T C X X
allele funny: rs9341284 G A X X
allele funny: rs9786021 C A X X
allele funny: rs13447370 C T X X
read 1073741824 bytes
read 2147483648 bytes
read 3205126797 bytes
packed geno read OK
end of unpack
packed geno read OK
end of unpack
numsnps input: 584131 53267
packedancestrymap output
numsnps output: 51919 numindivs: 21946
21946
21947

```

Once we have done that you can either wait a few seconds for the .ind file name asterisk (meaning it is being edited) to disappear or you can double-click on the tab name to autosave it.

[5] AT2 qpAdm Prep and Running

```

prefix: "/content/Florio_merged"
left_poplist: "IBS.DG,Nahua.DG,BantuSA.DG"
right_poplist: "Mbuti.DG,Israel_Natufian.AG,Israel_PPNB.AG,Russia_MA1_UP.SG,Turkey_Central_Boncuklu_PPN.AG,Turkey_Centr"
target: "Florio"
weights_file: "weights1.csv"
popdrop_file: "popdrop1.csv"
detach_shinyjs: 
allsnps: 

```

Run qpAdm (AT2)

Now we can run a random model to see if it works out. Note that I am pointing to the filepath prefix to my merged dataset now. I am Salvadoran so these Left/Source pops I have entered will suffice but you may want to look through the .xlsx or .anno file to choose appropriate samples for you that will give you non-negative percentages. This time I am trying with this Right/Reference

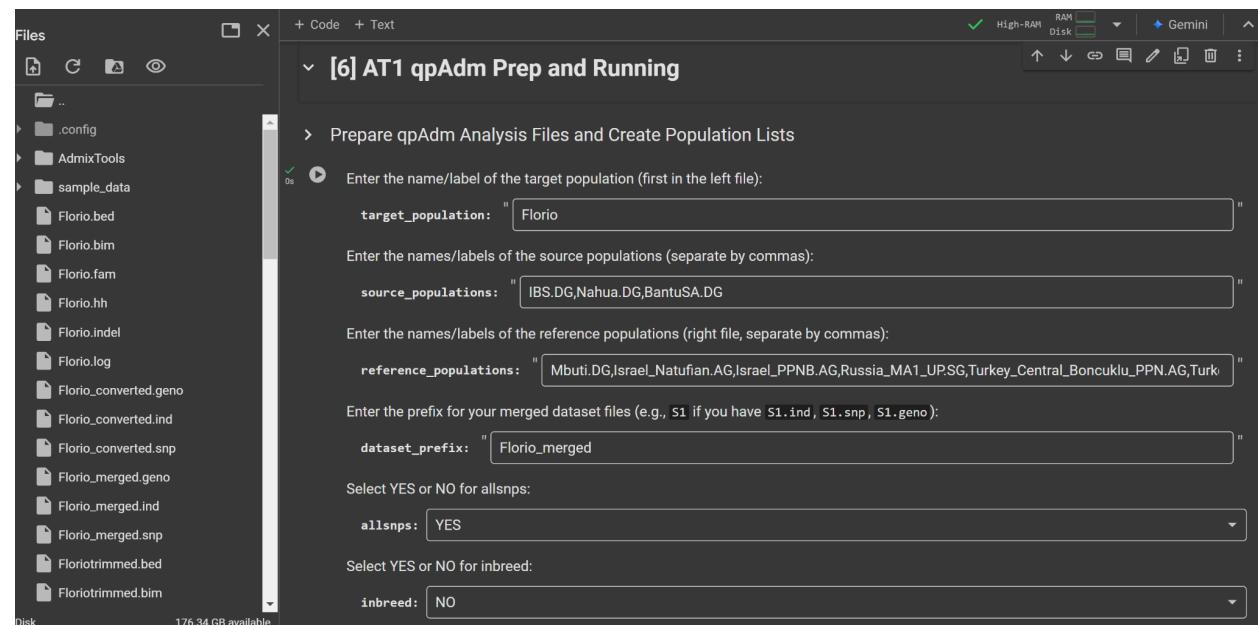
list:

Mbuti.DG,Israel_Natufian.AG,Israel_PPNB.AG,Russia_MA1_UP.SG,Turkey_Central_Boncuklu_PPN.AG,Turkey_Central_Pinarbasi_Epipaleolithic.AG,Morocco_Iberomaurusian.AG,Serbia_IronGates_Mesolithic.AG,Brazil_MG_C_LapaDoSanto_EH_HG_9600BP.AG. Please read the Documentation above regarding Right/Reference population selection.

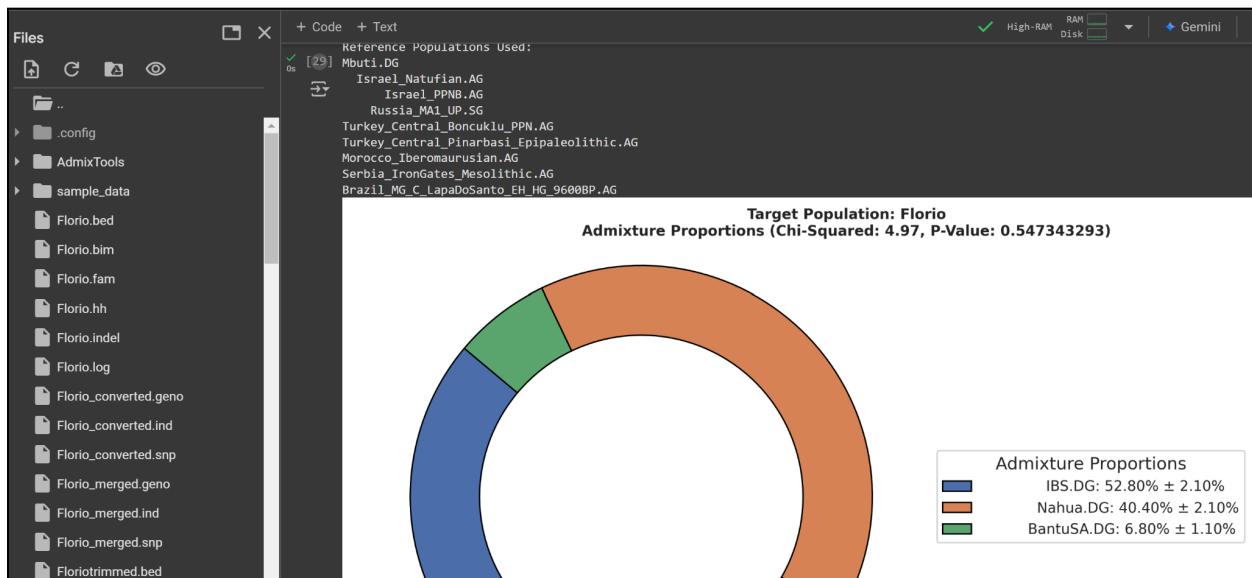
```
# A tibble: 3 × 5
  target    left     weight      se      z
  <chr> <chr>     <dbl>   <dbl> <dbl>
1 Florio IBS.DG    0.532  0.0217 24.5
2 Florio Nahua.DG  0.400  0.0210 19.1
3 Florio BantuSA.DG 0.0677 0.0113  5.98
# A tibble: 7 × 14
  pat      wt    dof   chisq      p f4rank IBS.DG Nahua.DG BantuSA.DG feasible
  <chr> <dbl> <dbl>   <dbl>   <dbl> <dbl> <dbl>   <dbl> <lgl>
1 000      0     6    5.17 5.22e- 1      2  0.532  0.400  0.0677 TRUE
2 001      1     7   67.0 6.08e- 12     1  0.571  0.429  NA     TRUE
3 010      1     7   695. 9.03e-146    1  0.897  NA     0.103  TRUE
4 100      1     7   808. 2.97e-170    1  NA     0.876  0.124  TRUE
5 011      2     8  1524. 4.94e-324    0  1     NA     NA     TRUE
6 101      2     8   943. 3.17e-198    0  NA     1     NA     TRUE
7 110      2     8   6645. 0          0  NA     NA     1     TRUE
```

For reference, I am ~55% European+Western Asian & North African, ~38% Indigenous American, ~6% Sub-Saharan African, and ~3% Unassigned on 23andMe. We could manually rotate and try different Right populations but for this instructive tutorial, this should suffice.

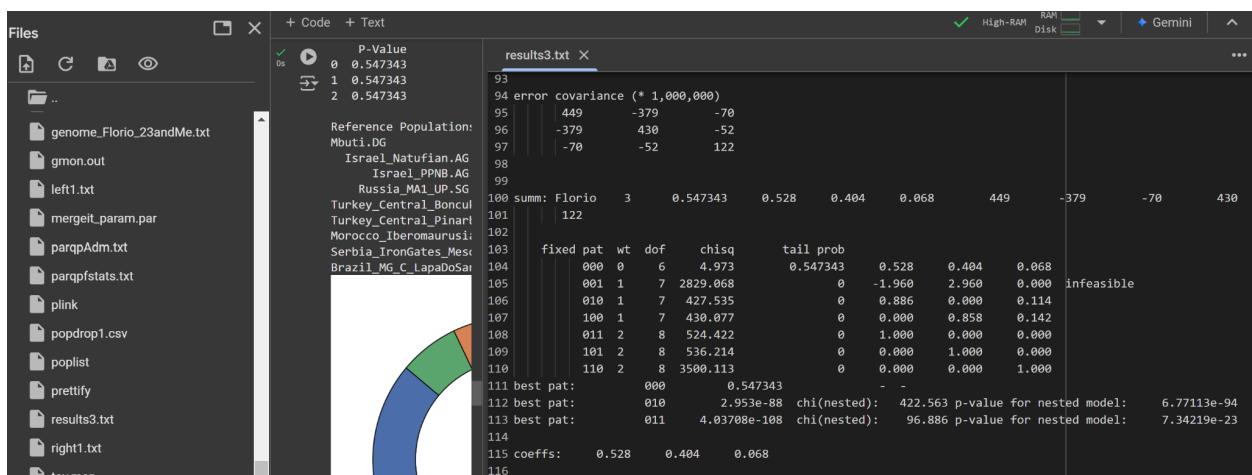
[6] AT1 qpAdm Prep and Running



Now let's cross-reference with AT1 qpAdm and visualize those results. The parameters should be the same, you can leave allsnps and inbreed alone for now unless you know what they do. Moreover, notice that when I point to the dataset prefix in this code cell I am not including /content/ before it.



Everything seems in working order. We can save the image or Copy and Paste it somewhere else by right-clicking on it. Please read the Documentation above regarding model interpretation, especially the Harney et al. Supplementary document.



We can also open up the results output file and scroll to the summary table to get the popdrop table and diagnose if other combinations might be better. Feel free to rename the results file if you do not wish to overwrite it with subsequent runs.

[7] Mounting Drive and Saving Compressed Files

The screenshot shows the Google Colab interface. On the left is the 'Files' sidebar with various files listed, including 'Florio.indel', 'Florio.log', 'Florio_converted.geno', 'Florio_converted.ind', 'Florio_converted.snp', 'Florio_merged.geno', 'Florio_merged.ind', 'Florio_merged.snp', 'Floriotrimmed.bed', 'Floriotrimmed.bim', 'Floriotrimmed.fam', 'Floriotrimmed.log', 'HOsnplist.snpplist', and 'I0001.AG.bed'. The main area contains a code cell with the following content:

```
+ Code + Text
To avoid the lengthy merging process in the future, we can mount our Google Drive and save any intermediate files and results using the code cell below. This will create a folder called colabadmixtools for you. Especially the merged dataset files (i.e. the .ind, .geno, .snp files). If you would like to do analysis on your merged data again in the future, just mount your drive using and specify the path of the compressed folder (Step 1 Option 3). After you mount, your drive should be at /content/drive/MyDrive within Colab. You will still need to perform Step 1 in the future but can then skip to Step 5.

> Mount Google Drive, Compress, and Save Files
Please mount your Google Drive, enter a name for the zipped folder, and then enter the paths of the files you want to save, separated by commas.

zip_folder_name: "Florio_mergedHO"
file_paths: "/content/Florio_merged.geno/content/Florio_merged.ind/content/Florio_merged.snp"

Show code

Mounted at /content/drive
Folder already exists: /content/drive/MyDrive/colabadmixtools
Added '/content/Florio_merged.geno' to the ZIP archive: Florio_mergedHO.zip
Added '/content/Florio_merged.ind' to the ZIP archive: Florio_mergedHO.zip
Added '/content/Florio_merged.snp' to the ZIP archive: Florio_mergedHO.zip
Files have been successfully compressed and saved to '/content/drive/MyDrive/colabadmixtools/Florio_mergedHO.zip'
```

Now that we have merged ourselves, we can Mount our Drive and save our 3 dataset files into a compressed ZIP folder in a folder called **colabadmixtools**, which will be created for you if it does not already exist in your Drive. I have chosen to name it *Florio_mergedHO* as I merged with HO but you can name it however you'd like. You can right-click on the *.geno*, *.ind*, and *.snp* files to copy their filepaths and paste them (separated by commas) into the code cell form field before running.

The screenshot shows the Google Colab interface. On the left is the 'Files' sidebar with files including 'Florio.fam', 'Florio.hh', 'Florio.indel', 'Florio.log', 'Florio_converted.geno', 'Florio_converted.ind', 'Florio_converted.snp', 'Florio_merged.geno', 'Florio_merged.ind', 'Florio_merged.snp', 'Floriotrimmed.bed', and 'Floriotrimmed.bim'. The main area contains a code cell with the following content:

```
+ Code + Text
existing_file_path: "/content/AncestryDNArawrtest4.txt"

Show code

Converting '/content/AncestryDNArawrtest4.txt' from AncestryDNA to 23andMe format...
File converted and saved as '/content/AncestryDNArawrtest4Converted.txt' in the current folder.

> Option 3: Mount Google Drive and Unzip Folder Previous Files (Returning Users)
Enter the path to your zip folder and run this cell to mount your Google Drive. If you previously ran Step 7 of this notebook, the zipped folder path will usually look like "/content/drive/MyDrive/colabadmixtools/examplefolder.zip".

zip_folder_path: "/content/drive/MyDrive/colabadmixtools/Florio_mergedHO.zip"

Show code

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
Contents of '/content/drive/MyDrive/colabadmixtools/Florio_mergedHO.zip' have been extracted to '/content/'

> Optional Step: Mount Google Drive and Upload SNP Lists
```

This way when we close the browser or disconnect for whatever reason, we can open up this notebook again and enter the filepath to our compressed folder which should be in a similar path to the one shown in the image above (replacing *Florio_mergedHO* with whatever you had named it). From there we can do **Step 1** to get the software we need and go straight to modeling.

Utilities

Extracting a Sample and Converting to 23andMe Format

A	B	C	D	E	F	G	H	I	J	K	L	M	N	
I3705.AG	I3705	BAQ3.65 ((3)) petrous		2020 AgranatTamar doi:10.1016/j.ENA:PRJEB37 Direct: IntCal		3344		44 1493-1304 ca..					Jordan_LBA.AG	
I3706.AG	I3706	BAQ3.77 ((4)) petrous		2020 AgranatTamar doi:10.1016/j.ENA:PRJEB37 Direct: IntCal		3304		41 1425-1284 ca..					Jordan_LBA.AG	
I3707.AG	I3707	BAQ3.87 (B3. petrous		2020 AgranatTamar doi:10.1016/j.ENA:PRJEB37 Direct: IntCal		3286		43 1413-1264 ca..					Jordan_LBA.AG	
I3708.AG	I3708	A561	petrous	2018 MathiesonNa doi:10.1038/r.ENA:PRJEB22 Context: Date		6500		548 5500-3600 BC..					Greece_Peloponnese_N.AG	
I3709.AG	I3709	A236 ((theta.1 petrous		2018 MathiesonNa doi:10.1038/r.ENA:PRJEB22 Direct: IntCal		5860		64 4036-3804 ca..					Greece_Peloponnese_N.AG	
I3718.AG	I3718	Mos52 (Deriv/tooth		2018 MathiesonNa doi:10.1038/r.ENA:PRJEB22 Direct: IntCal		7217		43 5359-5212 ca..					Ukraine_N.AG	
I3726.AG	I3726	Bag #574 (Un petrous		2017 SkoglundCell doi:10.1016/j.ENA:PRJEB21 Direct: SHCal		3023		57 1204-937 cal..					Tanzania_Luxmunda_3000BP.AG	
I3727.AG	I3727	M120- 2- L1 B petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Context: Date		1550		231 1-800 CE ..					Taiwan_Hanben_IA.AG	
I3728.AG	I3728	M125-L2 A (H) petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Direct: IntCal		1496		44 401-538 calC..					Taiwan_Hanben_IA.AG	
I3731.AG	I3731	M125- L2 D (I petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Context: Date		1550		231 1-800 CE ..					Taiwan_Hanben_IA.AG	
I3732.AG	I3732	M142 A (HB- : petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Direct: IntCal		1235		41 660-774 calC..					Taiwan_Hanben_IA.AG	
I3734.AG	I3734	M162- L1 (Ba) petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Context: 1d re		1575		43 300-450 CE ..					Taiwan_Hanben_IA.AG	
I3735.AG	I3735	M162- L1 (Ba) petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Direct: IntCal		1532		29 376-532 calC..					Taiwan_Hanben_IA.AG	
I3736.AG	I3736	M173- L1 (Ba) petrous		2021 WangNature2 doi:10.1038/s.ENA:PRJEB42 Context: Date		1550		231 1-800 CE ..					Taiwan_Hanben_IA.AG	
I3757.AG	I3757	LHY142-T tooth		2019 OlaldeScienc doi:10.1126/s.ENA:PRJEB30 Context: Arch		2300		29 400-300 BCE ..					Spain_IA_Celt_o.AG	
I3758.AG	I3758	LHY136 LH tooth, petrous		2019 OlaldeScienc doi:10.1126/s.ENA:PRJEB30 Direct: IntCal		2226		50 373-199 calB adult					Spain_IA_Celt.AG	
I3807.AG	I3807	22 (Individual tooth		2019 OlaldeScienc doi:10.1126/s.ENA:PRJEB30 Context: Arch		400		29 1500-1600 Ct..					Spain_NazariPeriod_Muslim.AG	
I3809.AG	I3809	24 tooth		2019 OlaldeScienc doi:10.1126/s.ENA:PRJEB30 Context: Arch		400		29 1500-1600 Ct..					Spain_NazariPeriod_Muslim.AG	
I3832.AG	I3832	6669 bone (phalan:		2020 AgranatTamar doi:10.1016/j.ENA:PRJEB37 Context: Arch		3300		58 1450-1250 BC..					Israel_MLBA.AG	
I3881.AG	I3881	MACE4; Skelettooth		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Context: Arch		6700		722 6000-3500 Badult					NorthMacedonia_N.AG	
I3890.AG	I3890	1, Tomb 7, Cr;tooth (molar)		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Context: Arch		2700		87 900-600 BCE adult ?					Armenia_KarmirBlur_Urartian.AG	
I3892.AG	I3892	TWO CODES: tooth (molar)		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Direct: IntCal		2749		8 815-781 calB..					Armenia_KarmirBlur_Urartian.AG	
I3911.AG	I3911	DT1; 69-33-1 petrous		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Context: Arch		3450		289 2000-1000 BC..					Iran_DinkhaTepe_BA_IA_1.AG	
I3912.AG	I3912	DT3; 69-33-3 petrous		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Direct: IntCal		3743		54 1881-1693 ca..					Iran_DinkhaTepe_BA_IA_2.AG	
I3913.AG	I3913	DT4; 69-33-4 petrous		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Direct: IntCal		3842		52 2012-1775 ca..					Iran_DinkhaTepe_BA_IA_1.AG	
I3914.AG	I3914	DT377; 66-23 petrous		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Direct: IntCal		3028		43 1192-1008 ca..					Iran_DinkhaTepe_BA_IA_2.AG	
I3915.AG	I3915	DT389; 66-23 petrous		2022 LazaridisAlpa doi:10.1126/s.ENA:PRJEB54 Direct: IntCal		2877		39 1002-841 cal..					Iran_DinkhaTepe_BA_IA_2.AG	

When we download the HO or 1240K datasets in Colab, it is good practice to use Colab File Explorer to download a local copy of the .xlsx sheet locally on our device to search for samples more easily. The first column will be the **Sample ID** (or the individual ID) and a few columns down will be the **Group Label ID** (or the population ID). In this example, I want to extract a single individual sample from the 1240K AADR and convert it to PACKEDPED (PLINK format).

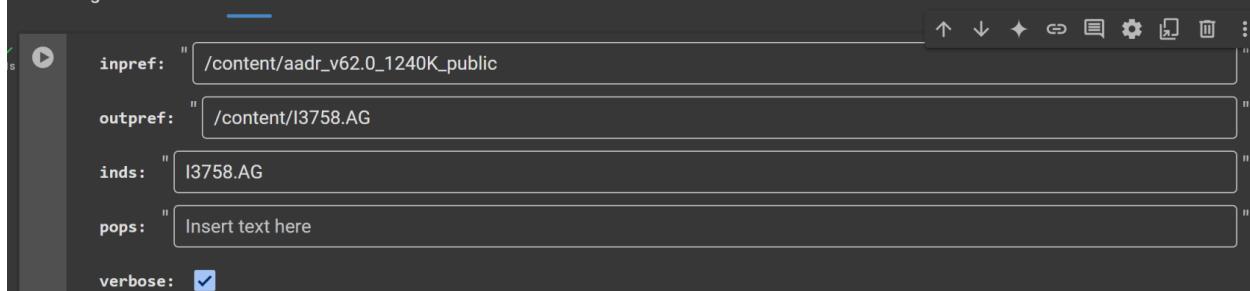
Utilities

EIGENSTRAT/PACKEDANCESTRYMAP to PLINK Conversion (+ Extraction)

Instructions

- **inpref:** Prefix of the input files.
- **outpref:** Desired prefix for the PLINK output files.
- **inds:** (Optional) Comma-separated list of individuals to extract. If provided, `pops` should be left empty.
- **pops:** (Optional) Comma-separated list of populations to extract. If provided, `inds` should be left empty.
- **verbose:** Choose whether to print progress updates.

Ensure that you provide either `inds` or `pops`, but not both if extracting. **WARNING:** Conversion of very large datasets is not possible in the Colab runtime (as of now) - please try converting/extracting the populations (Group Labels/IDs) you deem necessary for further analysis to avoid crashing the runtime.



The screenshot shows a Jupyter Notebook cell with the following code:

```
inpref: "/content/aadr_v62.0_1240K_public"
outpref: "/content/I3758.AG"
inds: "I3758.AG"
pops: "Insert text here"
verbose: 
```

The cell has a play button icon and a toolbar with various icons above it.

We point to the dataset prefix we wish to extract from. We enter the desired prefix/name for these PLINK files (include `/content/` as this is the destination). We enter the individual/sample ID according to the xlsx sheet. We can also extract multiple samples at once, just separate them by a single comma. However, extracting too many at once can lead to a RAM runtime error.

Alternatively, we can also extract group populations entirely (made up of multiple samples). Note that you can only extract individuals or groups at a time, not both at the same time. So for this example, I am extracting one individual sample and leaving `pops` blank.

```

+ Code + Text
+ PLINK to 23andMe Conversion (PLINK)
Instructions
• plink_file: Prefix of the PLINK input files without the extension.
• individual_id: ID of the individual to be converted to 23andMe format.

Ensure that the PLINK input files (.bed, .bim, .fam) are in the specified directory.

plink_file: "/content/I3758.AG"
individual_id: "I3758.AG"

Show code
PLINK is already downloaded. Skipping download step.
<ipython-input-29-e2bb1b22eb20>:24: FutureWarning: The 'delim_whitespace' keyword in pd.read_csv is deprecated and will be removed in a future version.
    fam_df = pd.read_csv(fam_file, delim_whitespace=True, header=None)
PLINK Log File:
PLINK v1.9.0-b.7.7 64-bit (22 Oct 2024)
Options in effect:
--bfile /content/I3758.AG
--keep filter.txt
--out I3758.AG
--recode 23
--snps-only

Hostname: 75b5b538f3a1
Working directory: /content
Start time: Thu Jan 9 20:55:47 2025

```

After playing that code cell we can see that the sample was converted successfully and a .txt file should appear in the Colab File Explorer which we can right-click and Download locally to use.

Extracting Population Groups and Plotting PCA

```

+ Code + Text
+ EIGENSTRAT/PACKEDANCESTRYMAP to PLINK Conversion (+ Extraction)
Instructions
• inpref: Prefix of the input files.
• outpref: Desired prefix for the PLINK output files.
• inds: (Optional) Comma-separated list of individuals to extract. If provided, pops should be left empty.
• pops: (Optional) Comma-separated list of populations to extract. If provided, inds should be left empty.
• verbose: Choose whether to print progress updates.

Ensure that you provide either inds or pops, but not both if extracting. WARNING: Conversion of very large datasets is not possible in the Colab runtime (as of now) - please try converting/extracting the populations (Group Labels/IDs) you deem necessary for further analysis to avoid crashing the runtime.

inpref: "/content/aadr_v62.0_1240K_public"
outpref: "/content/Test3"
inds: "Insert text here"
pops: "IBS.DG,Spanish.DG"
verbose: 

```

We can use the same code cell we used before to extract multiple samples using Group/Population Label IDs using the xlsx sheets. In this example, I will be extracting the 102

IBS.DG samples along with the 2 Spanish.DG samples from the 1240K AADR and performing a PCA on all of them. I will name this PLINK dataset *Test3* to not overwrite my previous PLINK files.

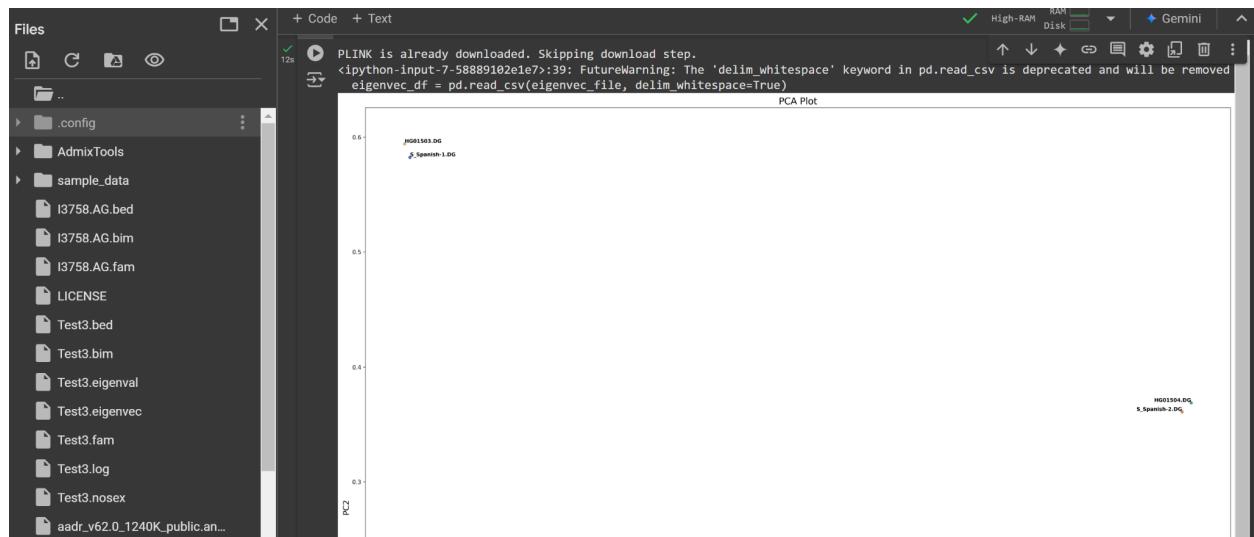
The screenshot shows the Google Colab interface. On the left is the 'Files' sidebar with various sample files like .config, AdmixTools, sample_data, and several AG files. The main area has a code cell titled 'PCA or MDS Plot Creation (PLINK)' with the following configuration:

```

12s [7] plink_file: "/content/Test3"
method: pca
count: 25
figsize_width: 2
figsize_height: 2
    
```

Below the code cell is a note: 'PLINK is already downloaded. Skipping download step.' followed by a warning message about the 'delim_whitespace' keyword. To the right of the code cell is a table titled 'Test3.eigenvec' containing 33 rows of data, each with columns FID, IID, PC1 through PC10, and PC95.

Now we can scroll down to this code cell and point to the filepath prefix for our new dataset. We can choose the PCA method and instruct it to analyze using 25 principal components from the data. We can adjust plot dimensions if the initial graph is difficult to visualize. Moreover, once we run this code cell we will also get an *.eigenvec* file which we can open by double-clicking on it in the Colab File Explorer and then seeing these values for each sample in a separate window.



Sometimes the sample markers overlap heavily, so we can adjust the plot dimensions and re-run the code cell to improve visibility. Alternatively, you can plot yourself using the coordinates.

Computing Fst

The screenshot shows a Jupyter Notebook cell with the title "Compute Fst (AT2)". The cell contains the following code:

```
✓ 0s ⏪ Input Parameters for Fst Computation
prefix: "/content/aadr_v62.0_1240K_public"
pop1_input: "IBS.DG"
pop2_input: "Basque.DG,MXL.DG"
boot: 
adjust_pseudohaploid: 
Show code

⤵ Input parameters set:
Dataset prefix: /content/aadr_v62.0_1240K_public
Population 1: ['IBS.DG']
Population 2: ['Basque.DG', 'MXL.DG']
Bootstrapping: False
Adjust pseudohaploid: True

Run Fst Computation
Show code
... Prefix path: /content/aadr_v62.0_1240K_public
Population 1: IBS.DG
```

The "Input Parameters for Fst Computation" section includes fields for prefix, population 1 (IBS.DG), population 2 (Basque.DG, MXL.DG), bootstrapping (unchecked), and adjust_pseudohaploid (checked). Below this, a summary of the input parameters is shown, followed by a "Run Fst Computation" button and its output.

In this example, we can point to the 1240K AADR dataset and compute Fst (read [this](#) and [this](#) for details). I want to estimate the population differentiation between IBS.DG and two other groups (comma-separated). We can leave **bootstrapping** off for now (sometimes it throws errors). We can leave **adjust_pseudohaploid** on for now. If you are estimating Fst for too many populations it can result in a RAM runtime error, refer to General Troubleshooting below for advice. We can always split it Fst estimates between pops and combine the outputs later if RAM is an issue.

The screenshot shows a Jupyter Notebook interface with several panes. On the left is a 'Files' pane listing various files including 'Test3.bed', 'Test3.bim', 'Test3.eigenval', 'Test3.eigenvec', 'Test3.fam', 'Test3.log', 'Test3.nosex', 'addr_v62.0_1240K_public.an...', 'addr_v62.0_1240K_public.ge...', 'addr_v62.0_1240K_public.ind', 'addr_v62.0_1240K_public.snp', 'fst_results.csv', 'plink', 'prettify', 'toy.map', 'toy.ped', and 'v62.0_1240K_public.xlsx'. The main pane displays code for running FST computation, showing population details and command-line output. A red circle highlights the 'est' column in the resulting CSV table.

pop1	pop2	est	se
IBS.DG	Basque.DG	0.00685776939064478	0.000155248154
IBS.DG	MXL.DG	0.0402012893967518	0.000431396792

Fst is based on allele frequencies so it will be highly sensitive to the amount of coverage. We can open and Download the results of this Fst run using File Explorer.

Plotting Admixture Graphs

The screenshot shows a Jupyter Notebook interface with several panes. On the left is a 'Files' pane listing various files. The main pane displays code for 'Admixture Graphs (AT2)'. It shows input parameters for the Admixture Graph analysis, including 'prefix' set to '/content/addr_v62.0_1240K_public', 'populations_input' set to 'Mbuti.DG,Israel_Natufian.AG,Israel_PPNB.AG,Russia_MA1_UP.SG,Turkey_Central_Boncuklu_PPN.AG,Turkey_C', 'numadmix' set to 2, 'outpop' set to 'Mbuti.DG', and 'stop_gen' set to 25. A red circle highlights the 'prefix' parameter value.

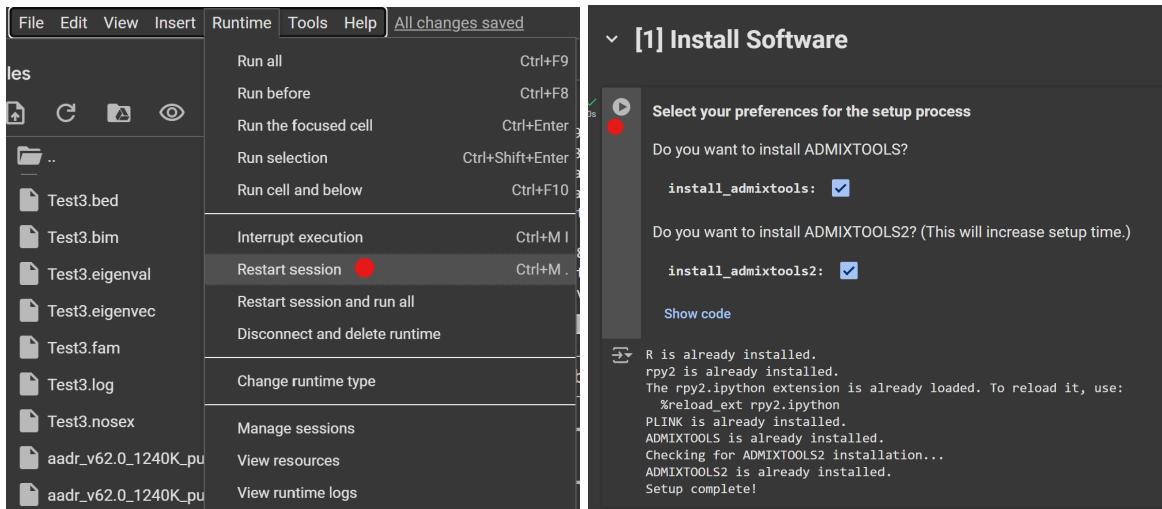
Refer [here](#) for usage. Admixture graphs work great for predicting relationships and contributions of admixture between populations. Personally, I do not use this feature but perhaps someone else would so I have included it in the notebook. In this example, we point to our dataset prefix filepath and I pasted a random Right list to make a random graph for illustrative purposes.

Choose an appropriate (or best guess) of admixture events. Here I chose 2. Choose an appropriate outgroup (here I chose Mbuti.DG). The **stop_gen** parameter you can set higher to simulate graphs and improve the score (however it will take more time).

We run the first code cell to establish the parameters then the next code cell to Run the graph simulations and ranking. The next code cell you can use to adjust the plot. We can also open the .csv file to see contributions and what ghost populations were created. The code cell will create an inverted tree of your best-ranked admixture graph. See Documentation for theory, usage, and interpretation.

General Troubleshooting

Runtime Crash due to RAM Limit:



Sometimes the analysis you are doing causes a runtime error due to limited RAM on a free standard Colab runtime. In this case, a crash can cause some tools to be unloaded, and you can **Restart the session** and perform **Step 1** again to load those important packages.

For most non-runtime related errors, it will usually be due to a typo in the prefix filepath or a sample name. Sometimes unchecking (or checking) additional parameters can fix the issue; read the Documentation to see what scenarios might apply to your particular combination of samples (e.g. using pseudohaploid samples).

Acknowledgements

ARA, Hands, Luka, and X for major contributions to the development of the notebook.

Sam, Ape, regh, qebap, Herman, Builder, rodolfo, and Jorge for testing, suggestions, and sample data.