NGS Homework Week5

XiaoyanWen

2/23/2022

Next Generation Sequence Analysis Homework Week 5

Week 5 serves as an introduction to population genomic analysis with NGS data. A key point from the pre-recorded videos for Week 5 is that low coverage re-sequencing data have high genotyping error rates

In this exercise, you will learn:

- how to assess genotype qualities in the Variant Call Format (VCF)
- the difference between analysis with genotypes versus genotype likelihoods
- the importance of genotype likelihooods in low coverage sequencing analysis
- how to calculate genotype likelihoods for population genomic analysis
- how to use genotype likelihoods to estimate genetic ancestry in low coverage data

How high are genotyping error rates and why is this a problem for population genomic inference? The sequencing error rate in Illumina platforms is quite low (<.1%), but calling genotypes (i.e. whether an individual is heterozygous or homozygous at a SNP site) can have error rates exceeding 1-2%.

A primary reason for a high genotyping error rate in low coverage data is binomial sampling. If an individual is heterozygous at a genomic position, at 1X coverage we can't tell if the individual is heterozygous. At 2X, we in theory could sequence reads representing both alleles, but we also might, by chance, sequence the same allele twice. As you can see, the deeper we sequence, the higher the probability that we will sequence both alleles in a heterozygote. It should be obvious that low coverage data has higher genotyping error rates because of this problem.

When sequencing coverage is less than 15X, downstream inferences made from genotype calls can be biased. This is frequently due to bias in the estimation of allele frequencies which many population genomic inferences rely upon. In such cases, it is common to use *genotype likelihoods* (rather than genotype calls themselves) which allows tools to incorporate uncertainty in the genotype calls into parameter (e.g., allele frequency) estimates. This approach improves estimation of parameters relative to the alternate approach of calling genotypes in a set of samples and counting alleles to obtain the allele frequencies. I provide a number of specific examples of this in the pre-recorded videos for Week 5.

In task 1 you will take a look at a typical VCF record and review how genotypes, genotype qualities, and genotype likelihoods are represented in VCF format. In Task 2-4 you will conduct analysis of genetic ancestry using genotype likelihoods with the softwares **ANGSD** and **NGSadmix**.

About the data

This week we will work with the VCF produced by the Week 2-4 re-sequencing workflow (Task 1) and additional data from the NGSadmix tutorial below. Both use raw short read re-sequencing human data from the 1000 Genomes Project. The NGSadmix data is referenced here:
http://www.popgen.dk/software/index.php/NgsAdmixTutorial

Task 1: Genotype qualities and Phred-scaled genotype likelihoods (PLs) in VCF

In previous assignments, you generated a snp and genotype callset using the GATK HaplotypeCaller/GenotypeGVCFs workflow. This workflow called genotypes by choosing the genotype with the highest genotype quality. The genotype quality (GQ) represents the PHRED-scaled probability that the genotype assignment (GT) was called in error. A high GQ indicates a high confidence call and a low probability of an error. (e.g., 30 = 1/1000 probability of an error). In deep coverage applications, it is common to use genotypes in downstream analysis because genotype qualities are typically high and genotyping error rates are relatively low.

Lets review one VCF record from the week 2-4 workflow. Log in to Greene and extract a single VCF record for the SNP at Chromosome 1 position 284,891.

```
srun --time=4:00:00 --mem=4GB --pty /bin/bash
gunzip -c /scratch/work/courses/BI7653/hw5.2022/task1/cohort.chr1to5.biallelic_snpsonly.v
cf.gz | grep -P '^1\t284891' # you could also use zcat instead of gunzip -c
```

The following questions concern the VCF format, so you may wish to review the GATK introduction to VCF format and the VCF specification for details:

GATK introduction to VCF https://gatk.broadinstitute.org/hc/en-us/articles/360035531692

VCF specification: https://samtools.github.io/hts-specs/VCFv4.3.pdf

Q1.1 Copy the variant record output from the grep command above and answer the following

3points

:

Q1.1a What is the variant quality for this SNP and what is the probability that it it is a false positive (i.e., called in error)?

hint: https://gatk.broadinstitute.org/hc/en-us/articles/360035531872-Phred-scaled-quality-scores

```
Answer: 10^(-38.31/10)=0.015%

The probability that it is a false positive is 0.015%.
```

Q1.1b How many heterozygous genotypes were called at this site? Answer the question and copy the sample field data (beginning with the GT field and ending with the PL field) for these genotypes.

```
Answer: two heterozygous genotypes were called at this site.
```

```
=1.24;SOR=1.112 GT:AD:DP:GQ:PL  ./.:0,0:0:.:0,0,0       0/0:2,0:2:6:0,6,82      0/0:2,0:2:6:0,6,70      0/0:2,0:2:6:0,6,76
./.:0,0:0:.:0,0,0       0/0:1,0:1:3:0,3,33      0/0:6,0:6:18:0,18,237   0/0:2,0:2:6:0,6,77      0/0:6,0:6:18:0,18,254   ./
.:0,0:0:.:0,0,0 0/0:6,0:6:18:0,18,270   ./.:2,0:2:.:0,0,1       0/0:3,0:3:9:0,9,112     0/0:4,0:4:12:0,12,159   0/1:6,2:8:
40:40,0,169     0/0:4,0:4:12:0,12,133   0/0:5,0:5:0:0,0,133     0/0:4,0:4:12:0,12,109   0/0:1,0:1:3:0,3,39      0/1:1,2:3:
18:42,0,18      0/0:2,0:2:6:0,6,63      0/0:2,0:2:6:0,6,47      ./.:0,0:0:.:0,0,0       ./.:0,0:0:.:0,0,0       0/0:5,0:5:
0:0,0,129       0/0:2,0:2:6:0,6,56      0/0:3,0:3:9:0,9,89      0/0:6,0:6:18:0,18,214   0/0:1,0:1:3:0,3,30
```

```
Answer: for the 1st heterozygous sample: 6 read for reference, 2 for alternate alleles

        for the 2nd heterozygous sample: 1 read for reference, 2 for alternate alleles
```

Q1.1d What are the nucleotide genotypes (in terms of A, T, G, C) of the 0/0 samples? The 0/1 genotypes? The 1/1 genotypes?

```
1       284891  .       C       T       38.31   .       AC=2;AF=0.043;AN=46;BaseQRankSum=0.967;ClippingRankSum=0.00;DP=82;
ExcessHet=3.1175;FS=0.000;InbreedingCoeff=-0.1403;MLEAC=2;MLEAF=0.043;MQ=25.60;MQRankSum=-1.800e-01;QD=3.48;ReadPosRankSum
=1.24;SOR=1.112 GT:AD:DP:GQ:PL   ./.:0,0:0:.:0,0,0       0/0:2,0:2:6:0,6,82      0/0:2,0:2:6:0,6,70      0/0:2,0:2:6:0,6,76
```

https://software.broadinstitute.org/gatk/documentation/article?id=11075

Answer: 0 for the REF allele, 1 for the first ALT allele, 2 for the second ALT allele

For this site, ref=C, alt=T → 0/0 = C/C 0/1 = C/T 1/1 = T/T

Q1.1e What is the GQ of the heterozygote genotypes you reported above with the lowest probability that the genotype is called an error?

Answer: 1st heterozygote genotype GQ=40 has the lowest probability that is called an error;

2nd heterozygote genotype GQ=18 has a higher p for error.

Q1.1f. A common observation is that the GQs of heterozygote genotypes are higher than GQs of homozygotes (you can convince yourself of this by looking at the above VCF record or other records in the VCF file). Provide an intuitive explanation why this might be.

Answer: heterozygous genotype has a higher relative fitness than homozygous.

Task 2: Estimation of genotype likelihoods from BAM

In cases where genotypes can be reliably inferred, we can use the genotypes in downstream analysis. For example, high quality genotypes could be used in a diagnostic test to determine if an individual is heterozygous or homozygous at particular disease-causing position in the genome or they could be used in applications like Genomewide Association Mapping (GWAS) of trait variation. You might be thinking, obviously, if we are conducting genome re-sequencing, isn't inference of the genotypes the whole point?

The answer is yes and no. With low coverage data it is risky to rely on genotype calls to draw inferences because the error rate may be unacceptably high. This doesn't, however, mean low coverage data are useless for many applications. In fact, many key population genomic parameters can be estimated from low coverage data such as the Site Frequency Spectrum (SFS)(i.e., a histogram of allele frequencies across the entire genome estimated from population re-sequencing data), inbreeding coefficients, nucleotide diversity, or genetic ancestry (e.g., as in 23andMe-type inferences). In low coverage situations, instead of using an inferred genotype call (which could easily be wrong), we should use genotype likelihoods (GLs), a set of probabilities for each genotype (e.g., homozygous REF, homozygous ALT, heterozygous) conditioned on the observed read data for a sample at the site in question. The genotype likelihoods explicitly account for the uncertainty in the genotypes and therefore are preferred in population genomic analysis of low coverage data. Methods that use genotype likelihoods are sometimes called "NGS methods" and are well-suited to analysis of low coverage datasets, although there applicability is limited to drawing inferences about populations, although they have although they have also been applied in Genomewide Association Studies (GWAS).

The first step to implementing these methods is to estimate genotype likelihoods from the data prior to conducting downstream analysis. The most common approach is to estimate them directly from the short read alignments (BAMs). The software ANGSD is a popular tool for this.

Here you will use **ANGSD** to generate a table of genotype likelihoods at sites that are likely to be polymorphic in a set of 30 small BAM files from the 1000 Genomes Project.

The BAMs and their index files are located here:

```
`/scratch/work/courses/BI7653/hw5.2022/task2/smallerbams`
```

Now create a list of BAM file paths.

```
    cd $SCRATCH
    mkdir ngs.week5
    cd ngs.week5
    mkdir task2
    cd task2
    find /scratch/work/courses/BI7653/hw5.2022/task2/smallerbams -name \*bam > bamfiles_f
or_GLs.txt
    # note * is a wild card, but in this context must be preceded by "\"
```

Review `bamfiles_for_GLs.txt` and confirm it has a list of full paths from root (i.e., "/" at the beginning of the path)

Now you will create a slurm script to generate a matrix of genotype likelihoods in Beagle format for all 30 samples at all polymorphic sites in the BAM. Review the ANGSD genotype likelihood page here:

http://www.popgen.dk/angsd/index.php/Genotype_Likelihoods

Now include the following in your job submission script:

1. Load the most recent ANGSD module
2. Add the command below to your script
3. Include the bam list file created above as the argument to the -bam option Consider also increasing the `#SBATCH --cpus-per-task` directive to run on multiple threads (recommended is 4 cpus).
4. Note in command below that we use the SLURM_CPUS_PER_TASK environmental variable in the command line. This will substitute the number of cpus you requested with `--cpus-per-task` into your command.

```
angsd -bam <bam file list> -P ${SLURM_CPUS_PER_TASK} -GL 2 -doMajorMinor 1 -doMaf 1 -SNP_
pval 2e-6 -minMapQ 30 -minQ 20 -minInd 25 -minMaf 0.05 -doGlf 2 -out GLs.gz
```

When you are ready, submit your job script. Monitor your job status with `squeue` and proceed when it is done.

Q2.1. What genotype likelihood model does the above script use to infer GLs? Your answer should be a model not a number. See above ANGSD genotype likelihoods website.

$1 point$

```
Answer:GATK model
```

Q2.2. How many genotype likelihoods are there per sample per site in the output file GLs.gz?

$1 point$

```
Answer: there are 3 genotypes likelihoods per sample per site:
major/major genotype, major/minor genotype, and minor/minor genotype.
```

Hopefully it is clear that these genotype likelihoods provide information about the uncertainty of the genotypes and that this uncertainty can now be used by downstream population genomic applications to better draw inferences from the data.

Task 3: Analysis of population structure with genotype likelihoods I

A common task in population genomics is to infer the genetic ancestry of individuals in a population sample such as is done by the company 23andMe. This is performed by clustering the samples into groups and simultaneously inferring the proportion of each sample genome that is derived from different ancestral populations. This proportion is sometimes called the "ancestry fraction" or "admixture proportion".

There are many software that can perform such model-based unsupervised clustering including STRUCTURE, fastSTRUCTURE, and ADMIXTURE. These approaches are "model-based" in the sense that they identify groups of individuals that are at Hardy-Weinberg and linkage equilibrium (and therefore represent a population of randomly mating individuals). These tools rely on genotype calls (not genotype likelihoods) such as are found in your VCF produced in Week 4. However, with low coverage data, errors in genotype calls bias the clustering and ancestry estimation.

With low coverage data, a better approach is to use software that use genotype likelihoods. Here we will illustrate the use of genotype likelihoods in ancestry estimation using a software called **NGSadmix**, a so-called "NGS method".

To begin, we will use a small dataset from this NGSadmix tutorial:

http://www.popgen.dk/software/index.php/NgsAdmixTutorial

Log in to Greene and perform the following:

```
srun --time=4:00:00 --mem=8GB --pty /bin/bash
cd $SCRATCH/ngs.week5
mkdir task3
cd task3
```

Genotype likelihoods can be calculated from the PLs in a VCF (see Task 1 and http://www.popgen.dk/software/index.php/NgsAdmix), but more commonly they are generated by a dedicated tool (e.g., ANGSD) directly from the BAM alignments.

For this task we will use a set of pre-computed genotype likelihoods (GLs). Download the pre-computed GLs and a population info file which provides information about the sample origins:

```
wget popgen.dk/software/download/NGSadmix/data/Demo1input.gz
wget popgen.dk/software/download/NGSadmix/data/Demo1pop.info
```

Have a look at the genotype likelihoods file:

```
gunzip -c Demo1input.gz | head -n 10 | cut -f 1-10 | column -t
```

You should again see multiple genotype likelihoods per sample per SNP.

Now we will run NGSadmix. Create a job submission script (2 hours and 8 GB memory is more than enough) that loads the most recent module of NGSadmix and executes the following command to run the

software. You may wish to start with the slurm template script from week 1:
`/scratch/work/courses/BI7653/hw1.2022/slurm_template.sh`

The following command will run NGSadmix with K (=number of clusters) of 3 and a minimum minor allele frequency set to 0.05:

```
    NGSadmix -likes Demo1input.gz -K 3 -minMaf 0.05 -seed 1 -o Demo1NGSadmix
```

Execute your script with sbatch when you are ready and monitor your job with `squeue` although this should finish quickly.

NGSadmix produces 4 files. Review the .log file to determine the log likelihood of the estimates ("best like")

```
    less Demo1NGSadmix.log # q to exit
```

The gzipped .fopt file contains an estimate of the allele frequency in each of the K=3 assumed ancestral populations at each snp locus. Have a look at the allele frequency estimates for the first 5 snps:

```
    gunzip -c Demo1NGSadmix.fopt.gz | head -n 5
```

Review the .qopt file. This contains an estimate of the individual's ancestry proportion from each of the assumed K=3 ancestral populations (rows are individuals, columns are populations).

Finally, DemoNGSadmix.filter contains sites that were filtered from the analysis.

You will download the .info and .qopt file to your personal computer and create an ancestry diagram (similar to the diagram presented in the pre-recorded video for ancestries of South American individuals) using ggplot2 in R. Before downloading we need to clean up the .qopt file. Review the file and display hidden characters with `-et` options to `cat`:

```
    cat -et Demo1pop.info | head -n 5  # $ symbol represents the end of line character \n
    cat -et Demo1NGSadmix.qopt | head -n 5
```

Note that Demo1NGSadmix.qopt has a trailing white space which will affect how the plotting code below reads the data. You can strip the extra whitespace as follows:

```
    perl -pe 's/\s$//' Demo1NGSadmix.qopt > Demo1NGSadmix_nowhite.qopt
```

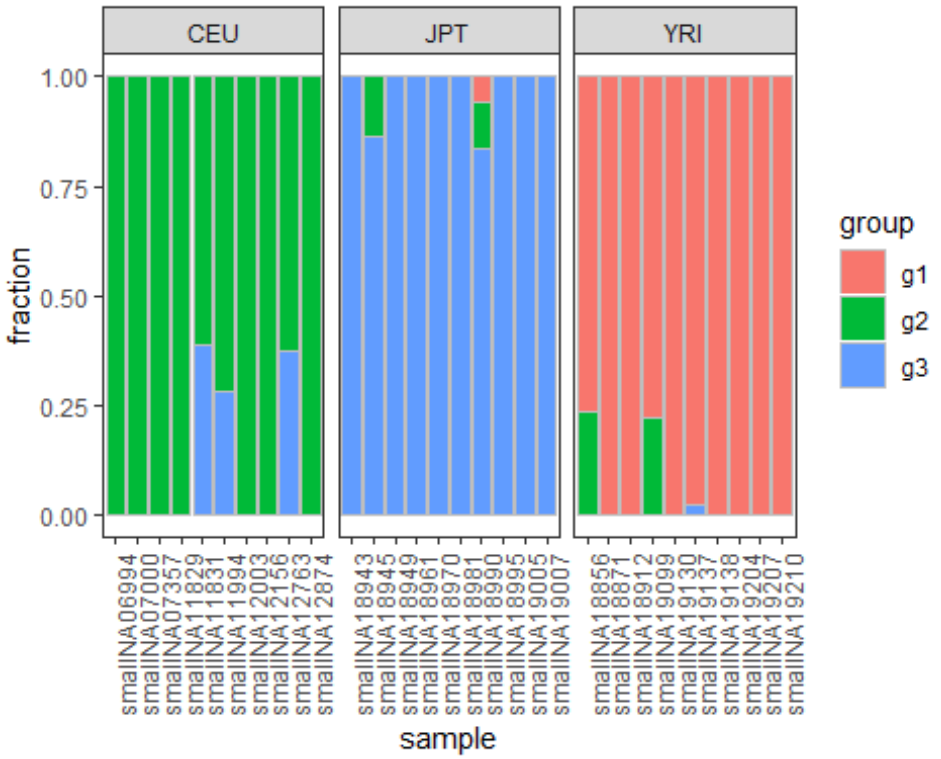
Review `Demo1NGSadmix_nowhite.qopt` to confirm there is no longer a whitespace before the end of line character.

Download the .info and .qopt files to your personal computer. Now open RStudio and make sure tidyverse is installed. If not enter `install.packages('tidyverse')` before proceeding. Now execute the following:

```
    library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
    pop.tbl_df <- read_delim(file = "Demo1pop.info",delim = " ", col_names = FALSE)

## Rows: 30 Columns: 2

## -- Column specification --------------------------------------------------------
## Delimiter: " "
## chr (2): X1, X2

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

    qopt.tbl_df <- read_delim(file = "Demo1NGSadmix_nowhite.qopt",delim = " ", col_names
= F)

## Rows: 30 Columns: 3

## -- Column specification --------------------------------------------------------
## Delimiter: " "
## dbl (3): X1, X2, X3

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

    qopt.tbl_df <- bind_cols(pop.tbl_df,qopt.tbl_df)

## New names:
## * X1 -> X1...1
## * X2 -> X2...2
## * X1 -> X1...3
## * X2 -> X2...4

    names(qopt.tbl_df) <- c("pop","sample","g1","g2","g3")

    qopt.tbl_df.long <- qopt.tbl_df %>%
                            pivot_longer(cols = g1:g3, names_to = 'group', values_to = 'fra
ction')

    # Plot with ggplot
    # Here we pipe pivoted ("long") output to ggplot
    qopt.tbl_df.long %>%
      ggplot(aes(x=sample,y=fraction,fill=group)) + geom_col(color = "gray", size = 0.1)
+
        facet_grid(~ pop, scales = "free", space = "free") +
        theme_bw() +
        theme(panel.grid.major=element_blank(),
            panel.grid.minor=element_blank(),
            axis.text.x = element_text(angle = 90))
```



If you cannot embed above code in an RMarkdown file, you could write the plot to a .pdf file:

```
    pdf("week5_task3.pdf",w=5,h=3.5)
    <ggplot code here>
    dev.off()
```

```
Answer: each individual on x-axis,

        fractions of the 3 ancestry genotypes were plotted on y-axis.
```

Q3.2a Which individuals in the CEU ("Utah residents (CEPH) with Northern and Western European ancestry") appear to have a second source of ancestry at K=3? Wher

$$1 point$$

```
Answer:3 individuals: smallNA11831, smallNA11994, smallNA12763

        appear to have a second source of ancestry at K=3.
```

Q3.2b Based on the populations in this analysis, which population is this additional source of ancestry shared with?

```
Answer: JPT is this additional source of ancestry.
```

Task 4: Analysis of population structure with genotype likelihoods II

Now repeat the analysis from Task 3 with a larger sample that includes the following populations from the 1000 genomes project

- ASW = Americans of African Ancestry in SW USA
- CEU = Utah residents (CEPH) with Northern and Western European ancestry
- CHB = Han Chinese in Beijing
- MXL = Mexican Ancestry from Los Angeles USA
- YRI = Yoruba in Ibadan, Nigeria

You can download the pre-calculated genotype likelihood and population information files to Greene:

```
wget popgen.dk/software/download/NGSadmix/data/Demo2input.gz
wget popgen.dk/software/download/NGSadmix/data/Demo2pop.info
```

Create a job submission script that will execute ngsadmix for K=3, K=4 and K=5.

Q4.1 Copy the contents of your script into your assignment document *1point*

```
#!/bin/bash
#
#SBATCH --nodes=1
#SBATCH --tasks-per-node=1
#SBATCH --cpus-per-task=1
#SBATCH --time=4:00:00
#SBATCH --mem=8GB
#SBATCH --job-name=slurm_wk5task4
#SBATCH --mail-type=FAIL
#SBATCH --mail-user=xw2470@nyu.edu

module purge

echo script begin: $(date)

module load ngsadmix/intel/20210224

NGSadmix -likes Demo2input.gz \
-K 3 \
-minMaf 0.05 \
-seed 1 \
-o Demo2K3NGSadmix

NGSadmix -likes Demo2input.gz \
-K 4 \
-minMaf 0.05 \
-seed 1 \
-o Demo2K4NGSadmix

NGSadmix -likes Demo2input.gz \
-K 5 \
-minMaf 0.05 \
-seed 1 \
-o Demo2K5NGSadmix

echo script completed: $(date)
```

Q4.2 Plot the results separately for K=3, K=4 and K=5 with ggplot using the example from Task 3 to assist you. Include your R code and plot in your answer.

1point

1. Be sure to strip the trailing white space from each ".qopt" file
2. Add column name(s) for K=4 and K=5 when renaming the columns of the qopt.tbl_df data frame in R using example code above as a guide.
3. Modify the pivot_longer command to include the column for K=4 and K=5.

You will note that a single population will take on different colors in different K values. This is simply because the clustering algorithm randomly assigns ancestries to K different populations in each run of the algorithm. Often, we manually align the colors across K values when plotting to make interpretation easier, but you do not need to do that here.

```
    perl -pe 's/\s$//' Demo2K3NGSadmix.qopt > Demo2K3NGSadmix_nowhite.qopt
    perl -pe 's/\s$//' Demo2K4NGSadmix.qopt > Demo2K4NGSadmix_nowhite.qopt
    perl -pe 's/\s$//' Demo2K5NGSadmix.qopt > Demo2K5NGSadmix_nowhite.qopt
```

check pop.info and confirm the deletions in .qopt files

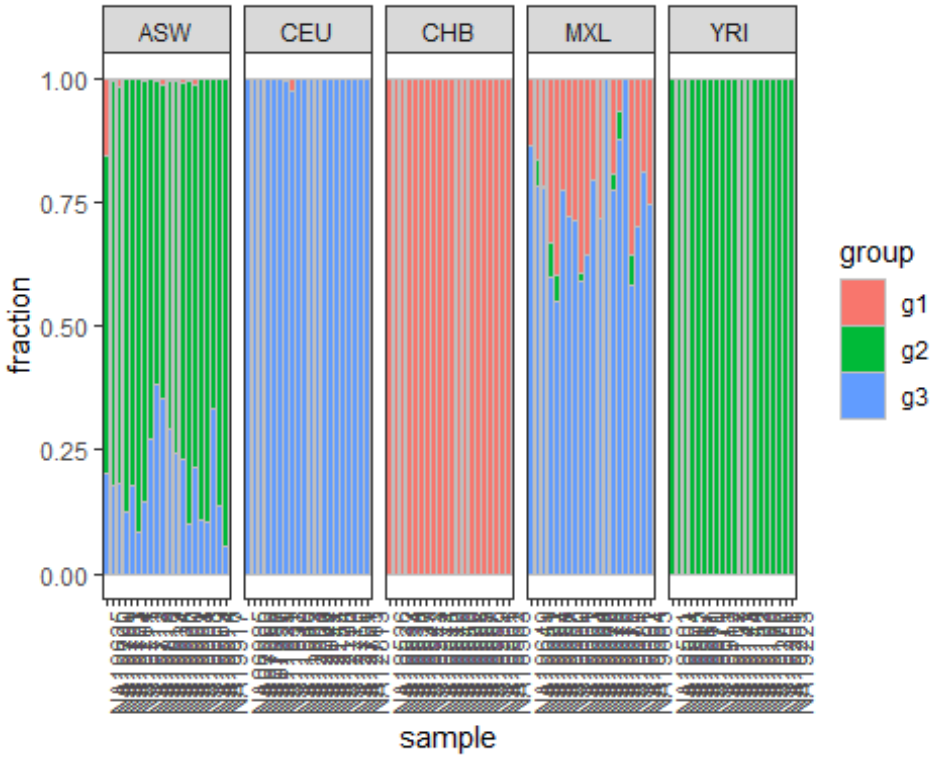
```
    cat -et Demo2pop.info | head -n 5 # $ symbol represents the end of line character \n
    cat -et Demo2K3NGSadmix_nowhite.qopt | head -n 5
    cat -et Demo2K4NGSadmix_nowhite.qopt | head -n 5
    cat -et Demo2K5NGSadmix_nowhite.qopt | head -n 5
```

Note that Demo1NGSadmix.qopt has a trailing white space which will affect how the plotting code below reads the data. You can strip the extra whitespace as follows:

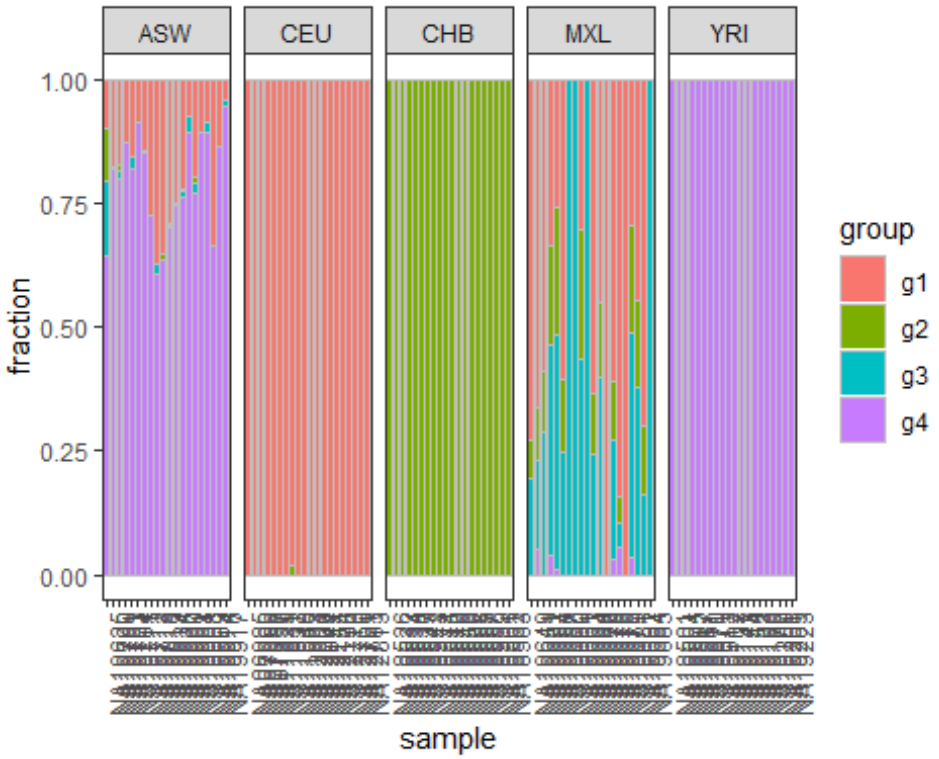
```
# Read-in pop.info file

pop_Demo2.tbl_df <- read_delim(file = "Demo2pop.info",delim = " ", col_names = FALSE)

## Rows: 100 Columns: 2

## -- Column specification --------------------------------------------------------
## Delimiter: " "
## chr (2): X1, X2

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

# read qopt files to dataframe files
qopt_Demo2K3.tbl_df <- read_delim(file = "Demo2K3NGSadmix_nowhite.qopt",delim = " ", col_
names = F)

## Rows: 100 Columns: 3

## -- Column specification --------------------------------------------------------
## Delimiter: " "
## dbl (3): X1, X2, X3
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

qopt_Demo2K3.tbl_df <- bind_cols(pop_Demo2.tbl_df,qopt_Demo2K3.tbl_df)

## New names:
## * X1 -> X1...1
## * X2 -> X2...2
## * X1 -> X1...3
## * X2 -> X2...4

names(qopt_Demo2K3.tbl_df) <- c("pop","sample","g1","g2","g3")

qopt_Demo2K4.tbl_df <- read_delim(file = "Demo2K4NGSadmix_nowhite.qopt",delim = " ", col_
names = F)

## Rows: 100 Columns: 4

## -- Column specification --------------------------------------------------------
## Delimiter: " "
## dbl (4): X1, X2, X3, X4

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

qopt_Demo2K4.tbl_df <- bind_cols(pop_Demo2.tbl_df,qopt_Demo2K4.tbl_df)

## New names:
## * X1 -> X1...1
## * X2 -> X2...2
## * X1 -> X1...3
## * X2 -> X2...4

names(qopt_Demo2K4.tbl_df) <- c("pop","sample","g1","g2","g3", "g4")

qopt_Demo2K5.tbl_df <- read_delim(file = "Demo2K5NGSadmix_nowhite.qopt",delim = " ", col_
names = F)

## Rows: 100 Columns: 5

## -- Column specification --------------------------------------------------------
## Delimiter: " "
## dbl (5): X1, X2, X3, X4, X5

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

qopt_Demo2K5.tbl_df <- bind_cols(pop_Demo2.tbl_df,qopt_Demo2K5.tbl_df)

## New names:
## * X1 -> X1...1
## * X2 -> X2...2
## * X1 -> X1...3
## * X2 -> X2...4
```

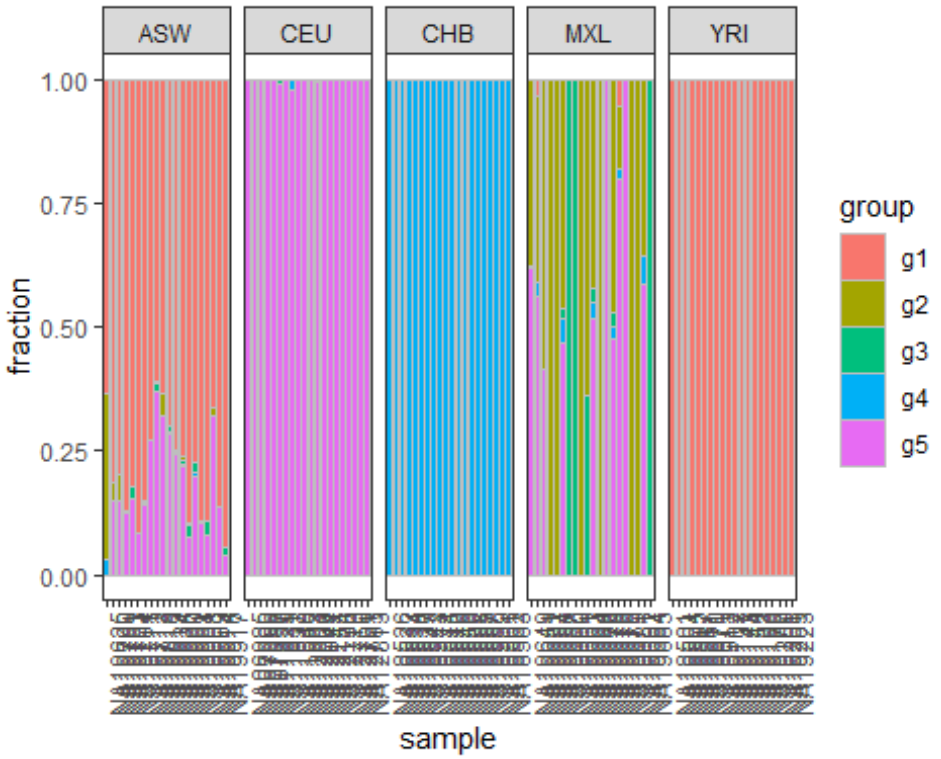
```
names(qopt_Demo2K5.tbl_df) <- c("pop","sample","g1","g2","g3", "g4", "g5")

# plotting Demo2 k=3
qopt_Demo2K3.tbl_df %>% pivot_longer(cols = g1:g3, names_to = 'group', values_to = 'fract
ion') %>%
  ggplot(aes(x=sample,y=fraction,fill=group))+
  geom_col(color = "gray", size = 0.1)+
  facet_grid(~ pop, scales = "free", space = "free")+
  theme_bw()+
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_text(angle = 90))
```




```
# plotting Demo2 k=4
qopt_Demo2K4.tbl_df %>% pivot_longer(cols = g1:g4, names_to = 'group', values_to = 'fract
ion') %>%
  ggplot(aes(x=sample,y=fraction,fill=group))+
  geom_col(color = "gray", size = 0.1)+
  facet_grid(~ pop, scales = "free", space = "free")+
  theme_bw()+
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_text(angle = 90))
```




```
# plotting Demo2 k=5
qopt_Demo2K5.tbl_df %>% pivot_longer(cols = g1:g5, names_to = 'group', values_to = 'fract
ion') %>%
  ggplot(aes(x=sample,y=fraction,fill=group))+
  geom_col(color = "gray", size = 0.1)+
  facet_grid(~ pop, scales = "free", space = "free")+
  theme_bw()+
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_text(angle = 90))
```



Q4.3 Review your outputs from Q4.2 and answer the following

$1 point$

:

Q4.3a Which populations appear to be "pure/unadmixed" at all K values? Select all that apply:

1. ASW
2. CEU
4. MXL
5. YRI

```
Answer: 2, 3, 5
```

Q4.3b Which populations appear as admixed (i.e. has mixed ancestry) at all values of K? Select all that apply.

1. ASW
2. CEU
3. CHB
4. MXL
5. YRI

```
Answer: 1, 4
```

Q4.3c Strict interpretation of ancestry diagrams often depends on identifying an appropriate K (similar to K-means clustering). This is a controversial topic. For assistance with this question and how generally to avoid mistakes when interpreting this type of analysis, see:

Lawson et al. 2018. A tutorial on how not to over-interpret STRUCTURE and ADMIXTURE bar plots. Nat. Comm. 9: 3258.

However, we can often gain insight into ancestry by also considering the totality of the analysis across different K values. Which population do you think is most likely to have internal population structure (e.g., have unrelated individuals with unique sources of ancestry?). Select the single best answer.

1. ASW
2. CEU
3. CHB
4. MXL
5. YRI

```
Answer: 4
```

You are finished, please review the Completing your assignment section above before submitting your report.