# Homework Exercise 12

1)

A)

According to ExAC, how do allele frequencies vary across synonymous, missense, nonsense and frameshift variants in protein-coding genes? Plot the 4 distributions and determine significance.

Ideally, this kind of analysis should be conducted on the entire ExAC dataset. However, the full VCF file of ExAC (ExAC.r1.sites.vep.vcf.gz, available at ftp://ftp.broadinstitute.org/pub/ExAC_release/release1/) is quite big (~4.6 GB, after compression), so instead you can work on the subset of variants in chromosome 22, which is available in the ~97 MB VCF file **ExAC.r1.sites.chr22.vep.vcf.gz** at the Moodle.

Just for reference, this file was generated from the original file using *vcftools*, by executing the following command: (in Linux)

*vcftools --gzvcf ExAC.r1.sites.vep.vcf.gz --chr 22 --recode --recode-INFO-all --stdout | gzip -c > ExAC.r1.sites.chr22.vep.vcf.gz*

**Tip**: All the information you need should be available within the VCF file (the INFO fields in ExAC are quite rich).

B)

Could the different allele frequencies of synonymous and missense SNPs be explained by GC content? Explain why it might be a confounder.

Test this confounding hypothesis by a multivariate analysis with the GC content of the reference allele as a covariate. For simplicity, you may focus on synonymous variants (i.e. comparing synonymous variants to the other three types).

C)

[**Optional**]

Repeat the analysis in (A) across different human populations (e.g. African, European, etc.).

D)

Do you believe your results reflect true phenomena of human population genetics, or are they artifacts caused by technical issues? Discuss various possible explanations.

# Bonus Questions

1)

What is multiple inheritance? How can it cause troubles, and why do some programming languages forbid it? What is Python's attitude to this issue?

Demonstrate how it works in Python by writing a class for SNVs (Single Nucleotide Variations) and another class for splice variants (i.e. genetic variations affecting splicing, e.g. by hitting exon junctions). Each of the two classes should implement an __str__ function. Demonstrate what happens when you have a third class inheriting from the two classes (i.e. a class representing an SNV with splicing consequences).

2)

So far in our course we considered two types of scripts - Jupyter notebooks, to be run sequentially and interactively by a human user specifically within the Jupyter environment, and Python modules, to be run from within other Python scripts. There are other ways of making our code accessible for use by others.

1. Compare the solution of creating a Command Line Interface (CLI) for each of your programs, as opposed to a Graphical User Interface (GUI) or just distributing the code directly. Describe some advantages of each approach.

Read the introduction of the profiling bonus question from the previous HW. That script was improved, and now runs very quickly, and even has a new feature - the user can input arbitrary kmers, of any size. This proved to be a hit. Even researchers from outside of the lab want to use the script. Your task is to create a CLI for that script (download it from the 'argparse question handout' folder).

2. Use the Python argparse module to create the CLI for the script. The following things should be parameterized:
    1. The input file (**mandatory**).
    2. The output file (**mandatory**).
    3. The kmers to search for (comma-separated, **optional**). If kmers are not specified, use the default behavior already present in the script of looking for all 4-mers.

    Make sure that it is well documented to the user (without them opening the code - i.e., by running the script with the -h/--help flag), and that it will only run if the input is valid (within the reasonable constraints of what argparse can check by itself - you shouldn't write your own error checking code for this assignment).
3. It turns out your hard work pays dividends! Demonstrate the Gooey tool, and make it work with the script you prepared in step 2.

3)

What is Fisher’s method (also known as Fisher's combined probability test)? Use it to obtain the overall significance of the association between allele frequencies to variant types after accounting for human populations (as in Q1C).

Why was it important to split the analysis per human population if we then merged the p-values?