Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Computational Biomedicine I
Fall Semester 2020

Project 2: Identification of disease variants

Assigned on: **03.11.2019** Due by: **08.12.2019 11:59pm**

Overview

With this exercise sheet, we will present the second practical project of the class. Topic of this second project is the determination of disease variants from a given control and disease cohort. The goal is to build a small tool that is capable of identifying variants specific to the disease cohort and interpret them. We will provide two variant files containing the genome information of a healthy control cohort as well as the genome cohort of a disease cohort.

In particular, in this exercise you will:

- decide on and implement a variant interpretation strategy
- implement a routine to read genome information from a standardized file format
- process genome information from existing databases
- define and apply evaluation metrics to judge the quality of your tool

We have split the work into several packages. There is no need that you exactly follow these packages. The split is rather a suggestion and thought as guidance on which steps we deem important.

Work Package 1.1 – Input Data

We have provided two vcf files summarizing the two populations you will work with. The complete specification of the format can be found here:

▷ http://www.internationalgenome.org/wiki/Analysis/vcf4.0/

The healthy population is a random subset of the 1000's genome dataset (https://www.internationalgenome.org/data). For patient privacy reasons we can not utilize real patient data for this project, however we have derived an artifical disease cohort from a subset of the 1000's genome dataset participant. We have embedded disease variants as well as some background variation for this purpose.

All input data is available for download from:

▷ http://public.bmi.inf.ethz.ch/teaching/cbm_2020/cbm_2020_project2/

Please download all data and familiarize yourself with the provided data formats. In case of question, please use the exercise sessions, moodle, or e-mail the TAs. All data is text-based and should be human-readable.

You will need to write a tool to write a reader for this input data based on the vcf-file specification (see URL mentioned earlier). The output data should also follow a vcf format with an appropriate extension to the format (essentially extend the informatino column) to indicate the result of your variant interpretation.

Depending on your chosen strategy you may want to plug into existing databases for known disease variants. You will learn about some of them in class. Generally you are free to choose any publicly available database. A good starting point is clinvar (https://www.ncbi.nlm.nih.gov/clinvar/). If you are uncertain about the credibility of a specific database, feel free to get in touch.

Work Package 1.2 – Variant interpretation strategy

There are generally two major strategies you can choose from and we encourage you to be creative here. In fact, creativity will be an evaluation criteria. One general strategy can be to try to identify the systematic differences between the two populations and connect to an existing database server to identify and interpret all disease variants. Alternatively, you can use an existing database to learn a classifier that is able to pick out and interpret the disease variants directly. Depending on your background, one solution may appeal more to you over the other. Just be aware that we expect an end-to-end script independent of the solution that you are choosing. In other words, if you choose to do database look-ups your script will need to take care of this queries automatically. Our expectations of performance are adjusted for the approach you take and overall performance is only a minor criteria in the overall grading scheme.

Work Package 1.3 – Evaluation

In order to properly evaluate your approach you will need to come up with a reasonable set of evaluation criteria. We encourage you to find inspiration by comparable approaches in the literature. Generally, ML approaches will feature some form of cross-validated approaches while database look-ups may want to look at the detection accuracies, statistical power and other related measures.

Submission

This is an open ended project, thus in general all libraries that do not side-step the work-packages are allowed. In other words, you can not use a vcf library to read the vcf files. However, you are allowed to use machine learning libraries, statistical libraries or libraries facilitating online database access. Also existing libraries for evaluation of ml approaches etc. can be used. We prefer python solutions, but if you have a strong opinion on a different language please get in touch with your TA team.

For development and versioning of your code, we provide each group with a git repository using the department's GitLab instance. The same git repository will also be used for your project submission. Once the project is due, we will take the code on the `master` branch of your group's repository. Please provide a conda environment file and a README. A Makefile would also be nice in case any of your code requires compiling. Please also check-in all benchmark plots you may have done into your repository with a short (max 1 page) summary description of your evaluation solution, evaluation results and conclusion.

Similar to project 1, we will provide a link for submission of a short 5 minute presentation for every group member outlining the solution and their role in the project.

Evaluation

The following list is a non-exhaustive sample of grading criteria:

- Providing a runnable tool that solves the given task (Minimum criteria to pass!)
- Documentation of solution.
- Creativity of solution.
- Proper choice of evaluation criteria.
- Short presentation.