Universiteit Leiden

Bachelor Project

Exploring the feasibility of pharmacogenomic data analysis

Rutger Homma

s1792261

r.j.homma@umail.leidenuniv.nl

supervised by
Katy Wolstencroft

18^{th} of August, 2021

Abstract

Due to development in genomic data collection, the cost of sequencing a genome is decreasing every year [1], this means that efficient ways of analysing this data are more valuable than ever. In this project, the feasibility of pharmacogenomic data analysis is explored. This is done by using data from 67 samples collected for the Canadian Personal Genome Project. These contain valuable information regarding genomic variants of the individuals. These files are analysed according to a workflow designed and described in this project, this is done to annotate these variants and link them with regard to known gene-drug interactions, and thus assist in the creating of personalised medicine for individuals. This information can be used by clinicians in an healthcare organisation to assist in medical decision making. Furthermore, a clean way of presenting the results from this workflow is designed, as well as the possibility for further statistical analysis of variants found using this workflow.

Contents

1 Introduction & background

In this project the feasibility of pharmacogenomic data analysis is explored. This is done by looking at the data available, and ways to analyse it. Due to cost of genomic sequence analysis getting lower every year [1], there is an abundance of genomic data. This data can provide valuable information that can be used in healthcare organisations. Due to the fact that this genomic data will be getting more abundant, we have to prepare for a future where every individual's genome is sequenced. And we have to prepare for large scale pharmacogenomic data analysis, so that we can use this data and not let it go unused. This project also aims to analyse some of this data that is available online. What this project does not take into account is the ethical side of this discussion, should we want that all of our genomic data is available for clinicians and hospitals. Also the security pitfalls, how to store and encrypt the data, are not answered. This project only asks the question; what is the feasibility of pharmacogenomic data analysis, something we can expect to do in the near future. The data used in this project is all found in already existing online databases. For the clinical application of this project, it often would be the case that the end user is also the supplier of the data, being the health care organisation.

1.1 Genomic Variant Calling

The human genome consists of various nucleotides. An individual has two copies of each chromosome, thus two copies of each gene. These two genes are called an allele. It is believed that the human genome contains around 25.000 genes [3]. Genes form the basis of inheritance and regulate protein production which is of vital importance. Between individuals these nucleotides which are building blocks of genes, may vary at a given position. This

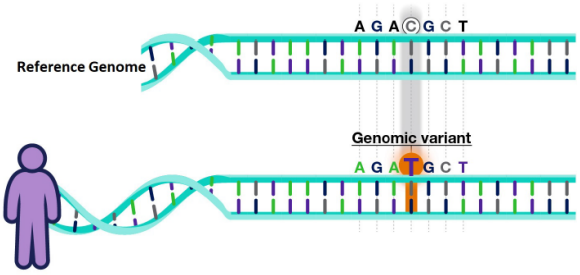

Fig. 1: Comparing an individuals genome to a reference genome [2]

variation is called a Single Nucleotide Variant (SNV). These variations are called against a reference genome, see Figure 1. This reference genome is a collection and does not represent an individual but rather an organism as a whole [4]. This means that every individual has a unique genome, which differs from the reference genome. These genomic variants may have a wide scale of effects. Most variants have no consequence at all, but some may be indicative of susceptibility to disease or insensitivity or over-sensitivity to certain drug treatments. This is because certain genes in the human body code for enzymes that play a role in the metabolism of drugs [5]. Due to the aforementioned variations that can occur in the human genome, it is possible for certain individuals with a certain variant, to be prone to disease or drug treatment. Exploring variations in genes and their effect on drug response is called pharmacogenomics. So pharmacogenomics can assist in creating personalised medicine for individuals.

1.2 Variant Call Format (VCF)

Variations of an individual are stored in a `.vcf` file [6]. This file-type is designed to contain information such as the reference genome the variations are called against, the positions of the variants and the nucleotide in the reference genome and the nucleotide of the individual for a specific position. So a `.vcf` file only contains variants of an individual, not the whole genome. But with help of the reference genome, the variants are the only information you need to construct the whole genome of that specific individual.

```
##fileformat=VCFv4.2
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS     ID        REF    ALT     QUAL FILTER INFO                              FORMAT      NA00001        NA00002        NA00003
20     14370   rs6054257 G      A       29   PASS   NS=3;DP=14;AF=0.5;DB;H2           GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.,.
20     17330   .         T      A       3    q10    NS=3;DP=11;AF=0.017               GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3   0/0:41:3
20     1110696 rs6040355 A      G,T     67   PASS   NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2   2/2:35:4
20     1230237 .         T      .       47   PASS   NS=3;DP=13;AA=T                   GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20     1234567 microsat1 GTC    G,GTCT  50   PASS   NS=3;DP=9;AA=G                    GT:GQ:DP    0/1:35:4       0/2:17:2       1/1:40:3
```

Fig. 2: Example vcf file [7]

Important to note is that the example vcf file shown in Figure 2 is showing entries for chromosome 20 only. A vcf file often contains variations of a specific

chromosome, or variations of the whole genome.

Other fixed fields are:

- POS - position: The reference position.
- ID - identifier: Identifier reference to the dbSNP database [8]
- REF - reference: Reference base. Nucleotide string, in case of insertion or deletion, REF and ALT include nucleotide string before the event.
- ALT - alternate: Alternate, non-reference bases.
- QUAL - quality: Quality score for assertion made in ALT.
- FILTER - filter status: PASS means the position passed all filters. Otherwise failed filters are shown, e.g. "q10" meaning that the QUAL score is below 10.
- INFO - additional information: Additional information can be shown.

The file format was first developed for the 1000 genomes project [9], and has been the community accepted standard for representing variation data ever since. This `.vcf` format is the backbone of the workflow explored in this project, because it is a relative easy and compact way of storing a lot of valuable information. Also the well-defined formatting and information contained by a `.vcf` file makes analysing it a well explored area.

1.3 Cytochrome P450

Cytochrome P450 (CYP) is an enzyme family that contains enzymes who play a role in drug metabolisms in individuals. This means that genes encoding these enzymes are of vital importance to pharmacogenomics. As a result, variations in certain genes (mainly CYP2C9, CYP2C19, CYP2D6, and CYP3A5, see Figure 3) may have an effect on drug uptake, and thus greatly alter the response to certain drug treatments [5].

Enzyme/ Gene	Chromosome	Significance of Genetic Variability	Examples of Substrates
CYP1A2	15q24.1	±	Agomelatine, caffeine, clozapine, lidocaine, melatonin, tacrine, theophylline, tizanidine
CYP2A6	19q13.2	+	Cotinine, coumarin, letrozole, nicotine, tegafur
CYP2B6	19q13.2	+	Bupropion, efavirenz, cyclophosphamide
CYP2C8	10q23.33	+	Amodiaquine, cerivastatin, dasabuvir, imatinib, loperamide, montelukast, pioglitazone, paclitaxel, repaglinide, rosiglitazone
CYP2C9	10q23.33	++	Acenocoumarol, diclofenac, fluvastatin, glimepiride, glipizide, ibuprofen, losartan, **phenytoin, S-warfarin**
CYP2C19	10q23.33	++	**Amitriptyline** (**and other tertiary amine tricyclic antidepressants**), **clopidogrel**, (**es**)**citalopram**, omeprazole, pantoprazole, **sertraline, voriconazole**
CYP2D6	22q13.2	++	**Amitriptyline** (**and other tricyclic antidepressants**), aripiprazole, atomoxetine, **codeine**, dextromethorphan, **fluvoxamine**, metoprolol, mianserine, MDMA, **ondansetron, paroxetine**, propafenone, risperidone, **tamoxifen**, thioridazine, timolol, tramadol, **tropisetron**, venlafaxine
CYP2E1	10q26.3	±	Ethanol, halothane, paracetamol
CYP2J2	1p32.1	±	Amiodarone, arachidonic acid, astemizole, cyclosporine
CYP3A4	7q22.1	±	Alfentanil, alprazolam, atorvastatin, budesonide, buspirone, cyclosporine, dexamethasone, erythromycin, felodipine, gefitinib, ibrutinib, lovastatin, midazolam, nifedipine, quetiapine, saquinavir, sildenafil, simvastatin, tacrolimus, testosterone, triazolam, verapamil, vincristine
CYP3A5	7q22.1	++	Saquinavir, **tacrolimus**, verapamil, substrates overlap with CYP3A4

±, Minor or no clinical significance.
+, Moderate clinical significance.
++, Major clinical significance.
Substrates with an associated Clinical Pharmacogenetics Implementation (CPIC) guideline are given in *bold*.

Fig. 3: CYP genes and their drug interactions [5]

CYP2D6 plays a role in the metabolism of 25% of all drugs in clinical use. This means that an alteration of this gene and a different metabolism than expected can have an effect on one quarter of the drugs used. Differences in the metabolism of pharmacogenomic genes are expressed by using an activity score, this activity score determines which type of metabolizer an individual is for this gene. See Table 1 for the scores and their phenotype. This score is based on genetic variations in alleles and their phenotypic consequences. Variations can be homozygous meaning that the same variation occurs in both of the alleles, or hetrozygous, when variations occur only in one of the alleles. Specific variations have a specific acitivity score, these are all described by PharmGKB in their gene-specific information table [10].

Thus, this project aims to analyse these variations to try and present valuable information about metaboliser phenotypes of a specific individual. This information then can be used to construct a recommendation of personalised medicine dosage. 17 of the 35 genes described by CPIC guidelines [11] are genes related to the Cytochrome P450 enzyme family. Guidelines constructed by CPIC aim to assist in specifying dosages of certain drugs with regards to the phenotype of individuals. Meaning that your phenotype for a certain gene

influences the recommended dosage, of usage as a whole, of a certain drug. For instance, the CPIC guideline regarding the dosage of Tricyclic Antidepressants [12] gives an overview of which dosage is recommended for individuals with a specific CYP2D6 phenotype.

1.4 Large scale genomic data analysis

Due to the fact that genomic data availability is rapidly expanding, a lot of tools and workflows have been designed to analyse this data. One of the most well known projects is the 1000 genomes project [9], which aimed to map the genetic variation of individuals across the world. Other projects developed programs to assist in this analysis, most of these programs aim to visualise or perform statistical studies on large data-sets. The tool used in this project, PharmCAT aims to annotate genomic variants. Projects like glow [13] aim to facilitate large scale genomic data analysis based on cloud computing. Or VIVA [14], which can be used to visualise `.vcf` files. Projects like PGen [15] have shown similar objectives, only focused on plant based genomics. And also, papers have been written to give an overview on managing large scale human genome data [16].

Other studies have taken a look at the flip side of the genomic data explosion and describe a way to make sure that, while sharing large amounts of personal genomic data, privacy is still being mandated [17].

1.4.1 Pharmacogenomics tools

There are a various number of pharmacogenomic tools available online. Most of these tools assist in annotating variants such as ANNOVAR [18] or PharmCAT [19]. Annotation of variants is the process of linking input variants to certain guideline so that a consequence of the specific variation may be noted. PharmCAT 0.8.0 also comes with a `VCF_preprocessing.py`-tool, which can prepare `.vcf` files to be analysed with PharmCAT. Other interesting resources may include PharmGKB [10], an online resource which curates pharmacogenetic knowledge. And, CPIC [20] which manages and curates gene/drug pathways. PharmCAT is linked to CPIC, so that information about gene-drug interactions is retrieved from the CPIC guidelines. Currently 442 gene-drug interactions are being, of have been researched by CPIC. With 218 interactions having enough evidence for a prescribing action.

1.5 Summary

Thus, in short, pharmacogenomic data is analysed in this project to help constructing a personalised medicine recommendation for individuals. This will be done by designing a workflow to analyse pharmacogenomic data, and constructing a program to present the results of this workflow in such a way that recommendations of certain dosages of drugs are clear to clinicans.

2 Materials & Methods

2.1 Materials

2.1.1 PharmCAT

To analyse the data, a program had to be found which took this data as an input and produced a suitable output to present in a clear way. Due to the fact that the focus of this research is more aimed at exploring the whole pipeline of data analysis, it was necessary to use already existing software. The most developed pharmacogenomic tool available was PharmCAT, this was the reason to investigate PharmCAT further.
PharmCAT [19] is a clinical annotation tool. Variants of an individual stored in a `.vcf` file, are linked to genes with known drug and/or treatment interactions. These interactions are taken from the Clinical Pharmacogenetics Implementation Consortium (CPIC) [11] guideline. It should be noted that PharmCAT only takes the interactions defined by CPIC into account. When new interactions are discovered, and they are not (yet) added to the CPIC guideline, then PharmCAT will not take them into account. This is most important with respect to the robustness of the pipeline.

2.1.2 Example input and output

The PharmCAT output consists of an `report.html` and (if specified in the command line) an `report.json`. The `report.html` output details all the possible variations of interest, and all the variations found for the specific sample. Often this much information is not needed and a clinician is only interested in the actual consequences of the variations of an individual. Therefore, an additional `output_table.html` is generated, displaying this information. Input can be both untrimmed, meaning all the variations of an individuals genome or untrimmed, meaning that the input only contains pharmacogenomic regions of interest that PharmCAT takes into consideration, as long as the reference genome is correct. Trimmed input is often smaller in size, and analysing will take less time than untrimmed input. Please consult the GitHub repository for the collection of data used in this project. [21]

2.1.3 Versions of PharmCAT

At the start of this research, the then-latest version of PharmCAT was used, version 0.7.1. During this project a newer version of PharmCAT was released, version 0.8.0. In the later stages this version was used. Version 0.8.0 was an improvement over 0.7.1 in several ways. Most importantly, the vcf-preprocessing tool.

2.1.4 Data used

VCF data from three different sources has been used for this project. See them listed below.

- PGP-Canada Data [22]: genomes of 67 individuals from the personal genome project Canada. Lifted over to correct reference genome. These are the files that were used in this project in the end. Output generated using these input files can be found on the GitHub page [21] and the results section. No other data from the previous two sources will be presented in this project, or on the GitHub page.
- PGP-UK Data [23]: 11 samples lifted over to the correct reference genome by Nienke Biesot. To get back on track, Nienke Biesot was kind to provide 11 `.vcf` files she herself had prepared for her project. These files were lifted over to the correct reference genome, and suitable results were generated. Because the data-set of 11 was quite small, an alternative source had to be found.
- EVA/EBI Data [24]: Data from EVA/EBI was explored but turned out not to be suitable for PharmCAT to analyse.

Also, these files are used in one or more steps of the workflow.

- `PharmCAT 0.8.0` [25]
- `hg19tohg38.over.chain` [26] liftover chain needed to lift the Personal Genome Project Canada files over from reference genome hg19 to reference genome hg38. PharmCAT requires input files to be aligned to reference genome hg38.
- `hg38.fa` [27] `.fasta` file of the whole reference genome hg38, needed to lift the personal genome project Canada files over to the correct reference genome so that it can be used as input for PharmCAT.

All code is written in Python 3.8.10 [28] and BASH [29]. Packages used within Python are:

- JSON encoder and decoder [30]
- Miscellaneous operating system interfaces [31]
- System-specific parameters and functions [32]
- Pandas [33]
- Beautiful Soup [34]
- Codecs [35]
- Matplotlib [36]
- Numpy [37]

For a collection of the code and all the output files, please see the GitHub page. [21]

2.1.5 VCF preprocessing

This tool is provided by the developers of PharmCAT and can be used to trim vcf-files to only contain pharmacogenomic variant positions that PharmCAT takes into consideration, and normalise vcf files. The positions PharmCAT takes into account are the positions annotated by CPIC, they can be found in `pharmcat_positions.0.8.0.vcf` on the GitHub releases page of PharmCAT [25]. For the PGP-UK files prepared by Nienke Biesot, pre-processing was necessary because of a prefix 'chr' missing in the CHROM field of the `.vcf` file (this is also the case for the example `.vcf` file as can be seen in Figure 2). Running the preprocessing tool added this prefix as well as trimming the data set.

The vcf pre-processing tool trimmed this data and provided some necessary

syntax for PharmCAT to be able to process these results into output `.html` files.

This was the case for the PGP-UK data, preprocessed by Nienke Biesot. Due to the formatting of the `.vcf` file, PharmCAT could only analyse the files after they were preprocessed with the tool provided by PharmCAT. The PGP-Canada files were (with the exception of the reference genome) in the correct format to be analysed by PharmCAT. This means that the preprocessing tool is not strictly necessary, it will only delete variant positions which PharmCAT does not take into consideration. Trimming these files will just speed up the PharmCAT analytics process. The time saved by trimming these files will be discussed in the technical feasibility section of the results.

2.1.6 ENA / EVA

Also needed is suitable data to analyse. Because PharmCAT is used, this data had to meet certain requirements. The first and most important requirement was that this data had to be in .vcf format. As talked about, this is the format used for (almost) all variant data. Another requirement was the reference genome. This had to be GRCh38.13 [4]. This last requirement is the main reason why not all data can be used. As will be discussed, it is possible to perform a 'lift-over' to change reference genome of VCF data. But this process can be tedious, so for easy of use, only data with the correct reference genome is searched for.

There are a couple of databases containing variation data. The European Variation Archive (EVA) is a database maintained by the European Bioinformatics Institute (EBI) [38]. The EVA is a database containing variation data of a wide scale of studies. For this project, the short genetic variants of the homo sapiens are used. Specifically whole genomic sequencing data. When using these filters, dozens of studies (and their data) are presented. Sadly it was not possible to filter on reference genome, so a lot of these studies where not considered.

2.2 Methods

2.2.1 Introduction

As can be seen in Figure 4, we start the process with a database containing raw VCF data. In this case the Canadian Personal Genome Project [22]. If this data is in the incorrect format, which is the case for this data, a liftover has to be performed. This is done using a shell script which performs liftover on all the `.vcf` files. See `lift_rh.sh`. This liftover-data then can be trimmed using the preprocessing tool provided by PharmCAT. In this case, the syntax of the input files are correct, so trimming is not necessary. The output of this preprocessing tool can be analysed through PharmCAT, which then produces two output-files of interest. A `.json` and `.html` file. This `.html` file can be used as raw output. But due to the abundance of information in this report, a separate program (`scores.py`) is written to process this report into a cleaner overview. Needed for this action is the `.html` and `.json` file, as well as an gene-specific information table from PharmGKB [39] to clean-up the CYP2D6 results. The output of this python program is a simple and easy to digest table containing all the relevant information regarding phenotypes for an individual as a result of variations in their genome. Also generated are two `.txt` files, containing all the CYP2D6 activity scores for individuals with variations in this gene, and all the CYP2D6 variations found in individuals. These two `.txt` files can be used as input in another small python script, `graph.py`. This will produce a histogram showing the distribution of activity scores, and will display some variation information in the terminal, such as occurrences of star allele variations.

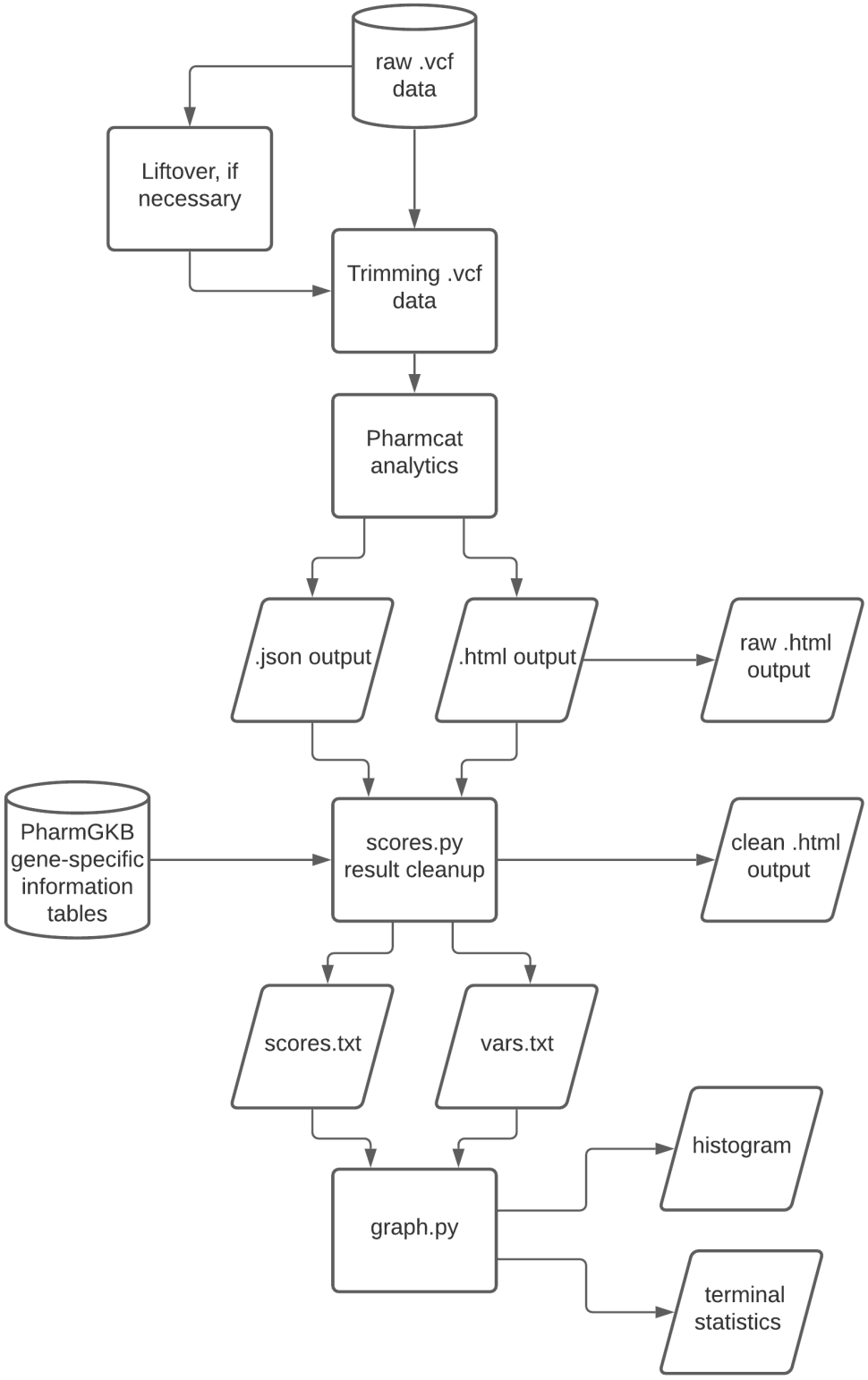

Fig. 4: Workflow diagram detailing the process with data-flows

2.2.2 Trimming

During this project, it became clear that most of the `.vcf` files that were analysed contained much more information than PharmCAT actually takes into consideration. This means that a lot of the initial `.vcf` file content is not used in any way, thus is redundant. It is important to note that when the CPIC guidelines are updated, certain positions that were considered redundant in the past, may be interesting now. This means that it is important to still store the most comprehensive `.vcf` file, with all the positions regardless of valuable information. However, it may be time saving to trim the files before analysing them using PharmCAT. This because a smaller file results in quicker and lighter analytic process. At first a trimming script was written in python, this resulted in longer time trimming than the time saved when analysing a trimmed file in comparison to the original file. In PharmCAT version 0.8.0 a vcf-preprocessing tool had been made available. This tool, among many other things, will filter the original `.vcf` file to an output containing only relevant positions. This tool was used to generate small and quick-to-analyse `.vcf` files.

2.2.3 Liftover

When using `.vcf` files that are aligned to a reference genome other than hg38, the reference genome PharmCAT requires, a liftover can be performed. This means that all the variant positions in the `.vcf` file will be translated from one reference genome to another. Tools used that are available for this action is CrossMap [40]. This requires a chain file (specific for your current reference genome and target reference genome), used in the project is the `hg19tohg38` chain file [26]. The fasta file from the target reference genome is also required, in this case the fasta file from hg38 [27]. A shell-script is written to automate this process, see the GitHub page [21].

2.2.4 Generating results

As mentioned before, the `.vcf` files were prepared using the preprocessing tool made available by PharmCAT. The results of this preprocessing are input-ready `.vcf` files for PharmCAT. A shell-script is written to easily and quickly analyse all the files in a specific folder. All the `.vcf` files are analysed using one command. The `.html` files are collected as output. These `.html` files contain information regarding the pharmacogenomic consequences of the variants of an

individual. However, these output `.html` files are fairly cluttered with information, so additional processing is needed for a clear overview of results.

2.2.5 Processing results

PharmCAT produced `.json` and `.html` files. These are both used in producing an additional result-table. To produce this table, a python program is written, see `scores.py`. This program uses an allele reference table provided by PharmGKB [39] to simplify the result table given in the output `.html` file. The current version of `score.py` will translate the given CYP2D6 variations into the corresponding activity score. Currently only CYP2D6 is supported, but by providing a different allele reference table and changing the variation input of `scores.py`, any gene with variations currently listed in the `report.html` can be translated into activity scores. The focus for this project is to translate the CYP2D6 variations because individuals often have multiple variations in these genes, with as a result that the phenotype is not clear from the standard `report.html`. `scores.py` calculates all the activity scores for variations that a certain individual has. Then it presents the highest calculated activity score for an individual as the final activity score, and connects this to the metabolizer table (see Table 1)

Enzyme function	Activity score
Poor metabolizer (PM)	0
Intermediate metabolizer (IM)	0.5
Extensive metabolizer (EM)	$1.0 - 2.0$
Ultra-rapid metabolizer (UM)	>2.0

Tab. 1: Enzyme function and activity scores for CYP2D6 [41]

Also, multiple variations resulting in the same phenotype are combined into one phenotype. See image 6.

Also written is a small script, `graph.py` which is able to plot all the activity scores (provided an individual has any variation for a given gene). In this case all the scores of individuals with an activity score for CYP2D6 are plotted. See Figure 5 for a graph. This script will also display some small statistics in the terminal, using `scores.txt` and `var.txt`, two text files which contain all the activity scores for samples with CYP2D6 variations and all the specific CYP2D6

variations respectively. Besides a histogram, `graph.py` will print in the terminal the percentages of activity scores (see Table 2) and a descending list of total variation occurrences (see Table 3)

3 Results

For this project, I analysed 67 genomes from the Personal Genome Canada Project. All these 67 files were analysed using the workflow that was developed in this project, see image 4. Below, in Figure 5 you can see the distribution of activity scores for variations in the CYP2D6 gene. Note that this graph displays the activity score all the samples. 29 samples had a variation in the CYP2D6 gene that can alter the activity score, if an individual has no known pharmacogenomic variations, their activity score is set at 1.0 (fully functional). Table 2 shows the percentages of samples that have a specific activity score. This table shows that almost 80% of the individuals are extensive metabolisers. With around 3% being poor or ultra-rapid metabolisers. Table 3 displays the total occurrences of a certain star allele in the 29 samples with variations in the CYP2D6 gene. A total of 301 variations were found, and all star alleles which occurred 4 times or more in total are shown.

Also shown is a comparison of the raw and trimmed `.html` output files. Figure 6 and Figure 7 show a snippet of both output files. Respectively the results of combining multiple phenotypes into one, and calculating the phenotype of CYP2D6 variations is shown. This is done so that the trimmed output file is less cluttered and it is easier for clinicians to spot anomalies within an individuals pharmacogenomic phenotype, so that the consequences for personalised medicine are easier to see.

All these results are generated with `scores.py` and `graph.py` as can be seen in the workflow diagram in Figure 4.

3.1 Technical Feasibility

Each genome of each sample from the Personal Genome Project Canada can be downloaded in the `.gz`-zipped format. For the purpose of comparing sizes and speed, two samples are compared. The genome of individual 1 and the genome of individual 84. These zipped files range from sizes 100MB for individual 1 to 200MB for individual 84. Unzipped, these pharmacogenomic variants range from sizes 460MB for individual 1 to 1.3GB for individual 84. Processing these files with the tool provided by PharmCAT can reduce the size of these files to 20MB and 55MB respectively. This takes around 10 seconds per file, assuming

the necessary files are provided. When analysing the files in PharmCAT, the preprocessed files from individual 1 and 84 take 16 and 19 seconds respectively. Analysing the raw files, it takes 3 minutes and 29 seconds and 4 minutes and 29 seconds respectively. So using the PharmCAT preprocessing tool, around 3 to 4 minutes can be saved per sample analysed.

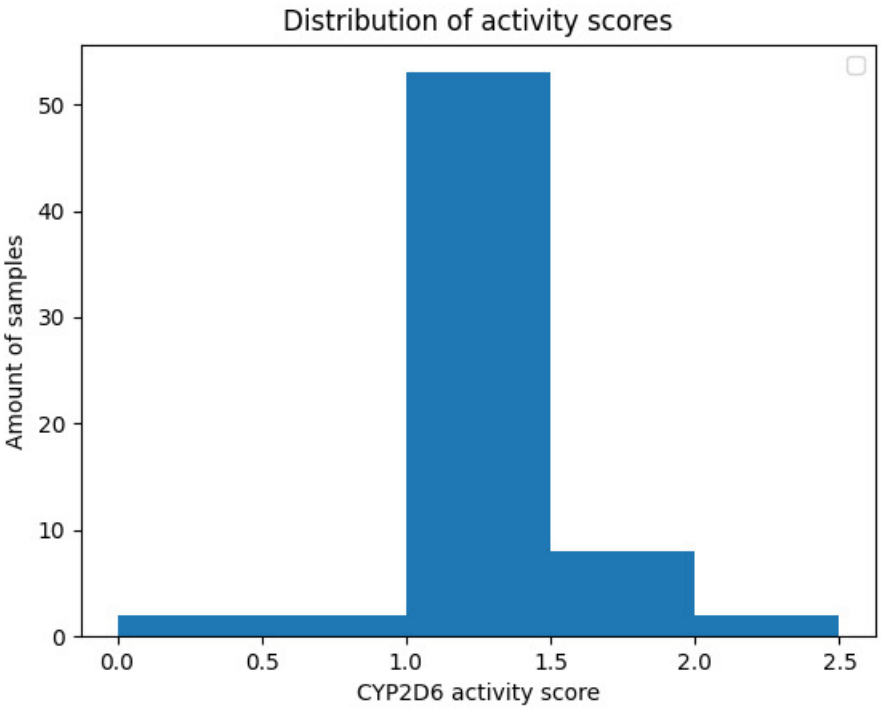

Fig. 5: Histogram showing the distribution of activity scores for CYP2D6 among all individuals in this data-set. Individuals with no known pharmacogenomic variants for CYP2D6 have their activity score set at 1.0 (fully functional)

Activity Score	Percent of total (N=67)
0.0	3%
0.5	3%
1.0	79.1%
1.5	11.9%
2.0	3%

Tab. 2: Distribution of CYP2D6 activity scores in percent against the total amount of samples

Variation (star allele)	Occurrence (Total variations = 301)
*1/*4	10
*1/*132	6
*1/*69	6
1//119	4
*1/*123	4
1*/*138	4
*1/*32	4
*1/*41	4
*1/*91	4
*4/*132	4
*4/*69	4

Tab. 3: CYP2D6 Star Alleles and their occurrences in the 29 individuals with known pharmacogenomic variations in the CYP2D6 gene

Genotype Summary

Genotypes called: 6 / 16

Drugs	Gene	Genotype	Allele Functionality[a]	Phenotype[a]	Missing Variant Input[b]
desflurane enflurane halothane isoflurane methoxyflurane sevoflurane succinylcholine	CACNA1S	not called	N/A	N/A	Yes
ivacaftor	CFTR	not called	N/A	N/A	Yes
efavirenz	CYP2B6	*1/*6 *1/*7 *1/*9 *1/*13 *1/*19 *1/*20 *1/*26 *1/*34 *1/*36 *1/*37 *1/*38	One normal function allele and one decreased function allele One normal function allele and one decreased function allele One normal function allele and one decreased function allele One normal function allele and one no function allele One normal function allele and one decreased function allele One normal function allele and one decreased function allele One normal function allele and one decreased function allele One normal function allele and one decreased function allele One normal function allele and one decreased function allele One normal function allele and one no function allele One normal function allele and one no function allele	Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer	Yes

(a) Sample 36 raw output

Genotype Summary

Genotypes called: 6 / 16

Drugs	Gene	Phenotype[a]	Missing Variant Input[b]
desflurane enflurane halothane isoflurane methoxyflurane sevoflurane succinylcholine	CACNA1S	N/A	Yes
ivacaftor	CFTR	N/A	Yes
efavirenz	CYP2B6	Intermediate Metabolizer	Yes

(b) Sample 36 trimmed output

Fig. 6: Sample number 36, with (a) being the raw PharmCAT output, and (b) being the `score.py` output.

amitriptyline atomoxetine clomipramine codeine desipramine doxepin fluvoxamine hydrocodone imipramine nortriptyline ondansetron paroxetine tamoxifen tramadol trimipramine tropisetron	CYP2D6†	*1/*87 *1/*94 *1/*95 *1/*99 *1/*100 *1/*101 *1/*114 *1/*132 *1/*142 *4/*4 *4/*10 *4/*36 *4/*37 *4/*47 *4/*49 *4/*52 *4/*54 *4/*56 *4/*57 *4/*64 *4/*65 *4/*69 *4/*72 *4/*87 *4/*94 *4/*95 *4/*99 *4/*100 *4/*101 *4/*114 *4/*132 *4/*142 *10/*56 *36/*56 *37/*56	One normal function allele and one uncertain function allele One normal function allele and one uncertain function allele One normal function allele and one uncertain function allele One normal function allele and one no function allele One normal function allele and one no function allele One normal function allele and one no function allele One normal function allele and one no function allele One normal function allele and one unknown function allele N/A Two no function alleles One no function allele and one decreased function allele Two no function alleles One no function allele and one uncertain function allele Two no function alleles One no function allele and one decreased function allele One no function allele and one uncertain function allele One no function allele and one decreased function allele Two no function alleles Two no function alleles One no function allele and one uncertain function allele One no function allele and one uncertain function allele Two no function alleles One no function allele and one uncertain function allele One no function allele and one uncertain function allele One no function allele and one uncertain function allele One no function allele and one uncertain function allele Two no function alleles Two no function alleles Two no function alleles Two no function alleles One no function allele and one unknown function allele N/A One decreased function allele and one no function allele Two no function alleles One uncertain function allele and one no function allele	Indeterminate Indeterminate Indeterminate Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Intermediate Metabolizer Indeterminate N/A Poor Metabolizer Intermediate Metabolizer Poor Metabolizer Indeterminate Poor Metabolizer Intermediate Metabolizer Indeterminate Intermediate Metabolizer Poor Metabolizer Poor Metabolizer Indeterminate Indeterminate Poor Metabolizer Indeterminate Indeterminate Indeterminate Indeterminate Poor Metabolizer Poor Metabolizer Poor Metabolizer Poor Metabolizer Indeterminate N/A Intermediate Metabolizer Poor Metabolizer Indeterminate	N/A

(a) Sample 36 - CYP2D6 raw output

amitriptyline atomoxetine clomipramine codeine desipramine doxepin fluvoxamine hydrocodone imipramine nortriptyline ondansetron paroxetine tamoxifen tramadol trimipramine tropisetron	CYP2D6†	Extensive metabolizer (Activity score: 1.5)	N/A

(b) Sample 36 - CYP2D6 trimmed output

Fig. 7: Sample number 36, with (a) being the raw PharmCAT output, and (b) being the `score.py` output. Note that in image (a) the list is longer than displayed.

4 Conclusion & Discussion

This project aimed to explore the feasibility of pharmacogenomic data analysis for personalised drug prescription. The project succeeded in analysing this data set and thus one can say that large scale genomic data analysis is feasible within the area that this project covered. Around 9 % of the individuals analysed in this project had a different activity score than a 'normal' metaboliser, meaning that they could benefit from applications in personalised medicine for specific drug doses. However, as shown in this project, it's not easy to streamline the process of analysing genomic variation data. There are tools available online, such as PharmCAT, which can be used to be of assistance in this process. However, when using these tools in a setting where you are both the provider and user of data, these tools can be a valuable asset in the analytics pipeline. In this case it is important that the data that is generated is in precisely the correct format. Following the constructed workflow provided in this project. It is possible to analyse large amount of data, provided that time and store is sufficient. For smaller data sets it is unnecessary to trim this data, but for larger data sets the time saved will become significant. It is also important to note that the size of files plays a role when dealing with large data sets. For the Canadian Personal Genome Project, each unzipped file was around 400 megabytes large. One can imagine the amount of storage needed when dealing with data sets containing 1000 files or more.

As can be seen in the results section, there are ways to visualise results generated using the `scores.py` program, so that a quick overview can be made. This may be useful in later stages when performing at population wide variation studies.

4.1 Discussion

It is important to note that due to the fact that PharmCAT is used in this project. Only variants that PharmCAT takes into account are annotated. This means that if a source other than CPIC (the guideline PharmCAT uses) decides to include a variation into their guideline, PharmCAT will not consider this, unless CPIC annotates this.

The pitfalls in this project emphasise the necessity for a standardised data format. Currently this has been realised by using the `.vcf` file format. But it is important to make sure that data that will be produced in the future can

and will be freely exchanged between sources, for instance health care organisations. When different organisations use different data-standards, conducting population wide variation analysis will become more difficult than necessary.

4.2 Future Research

This project gives an introductory overview into large scale pharmacogenomic data analysis. There are plenty of areas worth expanding research on. For instance, it would be interesting to take a look at bigger data sets from a population, and compare the results for specific genes to other populations. Or specific dosage recommendations could be added to the output table PharmCAT provides. Also more in-depth statistical analysis of specific variation occurrences could be looked into.

5 Acknowledgements

First, I would like to thank Katy Wolstencroft for her supervision during this bachelor project, she has been of great help to me while working on, and writing this project. Also a special thank you to Nienke Biesot, for kindly providing me with data from the personal genome project UK she herself prepared for her own bachelor project.

6 References

References

[1] Kris A. Wetterstrand, M.S. The cost of sequencing a human genome, 2020. [Online; accessed 20-July-2021], available at `https://www.genome.gov/about-genomics/fact-sheets/Sequencing-Human-Genome-cost`.

[2] `Genome.gov`. Polygenic risk scores, 2020. [Online; accessed 01-August-2021, Image adapted from `https://www.genome.gov/Health/Genomics-and-Medicine/Polygenic-risk-scores`].

[3] Salzberg S. L. Open questions: How many genes do we have? *BMC biology*, 16:94, 2018.

[4] Bethesda (MD): National Library of Medicine (US) and National Center for Biotechnology Information. Genome reference consortium human build 38, patch release 13, 2013. [available at `https://www.ncbi.nlm.nih.gov/assembly/GCF_000001405.39`].

[5] Aleksi Tornio and Janne T. Backman. Chapter one - cytochrome p450 in pharmacogenetics: An update. In Kim Brøsen and Per Damkier, editors, *Pharmacogenetics*, volume 83 of *Advances in Pharmacology*, pages 3–32. Academic Press, 2018.

[6] Danecek P, Auton A, Abecasis G, Albers CA, DePristo MA Banks E, Handsaker RE, Lunter G, Marth GT, Sherry ST, McVean G, and Durbin R; 1000 Genomes Project Analysis Group. The variant call format and vcftools. *Bioinformatics*, 27:2156–2158, 2011.

[7] samtools organization. The variant call format (vcf) version 4.2 specification, 2021. [Online; accessed 15-June-2021], available at `https://samtools.github.io/hts-specs/VCFv4.2.pdf`.

[8] Sherry ST, Ward MH, Kholodov M, and et al. dbsnp: the ncbi database of genetic variation. *Nucleic Acids Research*, 29:308–311, 2001.

[9] The 1000 Genomes Project Consortium. A global reference for human genetic variation. *Nature*, 526:68–74, 2015.

[10] Michelle Whirl-Carrillo, Rachel Huddart, Li Gong, Katrin Sangkuhl, Caroline F. Thorn, Ryan Whaley, and Teri E. Klein. An evidence-based

framework for evaluating pharmacogenomics knowledge for personalized medicine. *Clinical Pharmacology & Therapeutics*, n/a:n/a, 2021.

[11] Clinical Pharmacogenetics Implementation Consortium (CPIC). Guidelines, 2021. [Online; accessed 25-June-2021, available at `https://cpicpgx.org/guidelines/`].

[12] Hicks JK, Sangkuhl K, and et al. Swen JJ. Clinical pharmacogenetics implementation consortium guideline (cpic) for cyp2d6 and cyp2c19 genotypes and dosing of tricyclic antidepressants: 2016 update. *Clinical Pharmacology & Therapeutics*, 102:37–44, 2017.

[13] Kiavash Kianfar . Glow 0.3.0 introduces new large-scale genomic analysis features, 2020. [Online; accessed 01-August-2021, available at `https://databricks.com/blog/2020/04/23/glow-0-3-0-introduces-new-large-scale-genomic-analysis-features.html`].

[14] G.A. Tollefson, J. Schuster, and F. et al. Gelin. Viva (visualization of variants): A vcf file visualization tool. *Scientific Reports*, 9:2045–2322, 2019.

[15] Y. Liu, S.M. Khan, and J. et al Wang. Pgen: large-scale genomic variations analysis workflow and browser in soykb. *BMC Bioinformatics*, 17:177 – 186, 2016.

[16] T. Tanjo, Y. Kawai, and K. et al. Tokunaga. Practical guide for managing large-scale human genome data in research. *Journal of Human Genetics*, 66:39–52, 2021.

[17] Z. Wan, Y. Vorobeychik, W. Xia, E. W. Clayton, and B Kantarcioglu, M. & Malin. Expanding access to large-scale genomic data while promoting privacy: A game theoretic approach. *American journal of human genetics*, 100:316–322, 2017.

[18] K. Yang, H. & Wang. Genomic variant annotation and prioritization with annovar and wannovar. *Nature protocols*, 10:1556–1566, 2015.

[19] M. D. Klein, T. E. & Ritchie. Pharmcat: A pharmacogenomics clinical annotation tool. *Clinical pharmacology and therapeutics*, 104:19–22, 2018.

[20] Relling MV and Klein TE. Pharmgkb:cpic: Clinical pharmacogenetics implementation consortium of the pharmacogenomics research network. *Clinical Pharmacology & Therapeutics*, 89:464–467, 2011.

[21] Homma R.J. Bachelor project, 2021. [available at `https://github.com/rutger1501/bachelorproject`].

[22] Miriam S. Reuter, Susan Walker, Bhooma Thiruvahindrapuram, Joe Whitney, Iris Cohn, Neal Sondheimer, Ryan K.C. Yuen, Brett Trost, Tara A. Paton, Sergio L. Pereira, Jo-Anne Herbrick, Richard F. Wintle, Daniele Merico, Jennifer Howe, Jeffrey R. MacDonald, Chao Lu, Thomas Nalpathamkalam, Wilson W.L. Sung, Zhuozhi Wang, Rohan V. Patel, Giovanna Pellecchia, John Wei, Lisa J. Strug, Sherilyn Bell, Barbara Kellam, Melanie M. Mahtani, Anne S. Bassett, Yvonne Bombard, Rosanna Weksberg, Cheryl Shuman, Ronald D. Cohn, Dimitri J. Stavropoulos, Sarah Bowdin, Matthew R. Hildebrandt, Wei Wei, Asli Romm, Peter Pasceri, James Ellis, Peter Ray, M. Stephen Meyn, Nasim Monfared, S. Mohsen Hosseini, Ann M. Joseph-George, Fred W. Keeley, Ryan A. Cook, Marc Fiume, Hin C. Lee, Christian R. Marshall, Jill Davies, Allison Hazell, Janet A. Buchanan, Michael J. Szego, and Stephen W. Scherer. The personal genome project canada: findings from whole genome sequences of the inaugural 56 participants. *CMAJ*, 190(5):E126–E136, 2018.

[23] Chervova O, Conde L, Guerra-Assunção JA, Moghul I, Webster AP, Berner A, Larose Cadieux E, Tian Y, Voloshin V, Jesus TF, Hamoudi R, Herrero J, and Beck S. The personal genome project-uk, an open access resource of human multi-omics data. *Scientific Data*, 6:257–266, 2019.

[24] Lowy-Gallego E, Fairley S, and et al. Zheng-Bradley X. Variant calling on the grch38 assembly with the data from phase three of the 1000 genomes project. *Wellcome open research*, 4:50, 2019. [database available at `https://www.ebi.ac.uk/eva/?eva-study=PRJEB31735`].

[25] Pharmcat, 2021. [available at `https://github.com/PharmGKB/PharmCAT`].

[26] Robert Kuhn, David Haussler, and W Kent. The ucsc genome browser and associated tools. *Briefings in bioinformatics*, 14, 08 2012. [hg19 to hg38 liftover chain available at `https://hgdownload.soe.ucsc.edu/gbdb/hg19/liftOver/`].

[27] Robert Kuhn, David Haussler, and W Kent. The ucsc genome browser and associated tools. *Briefings in bioinformatics*, 14, 08 2012. [hg38 fasta file available at `http://hgdownload.cse.ucsc.edu/goldenpath/hg38/bigZips/`].

[28] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[29] P GNU. Free software foundation. bash (3.2. 48)[unix shell program], 2007.

[30] Python Software Foundation. Json encoder and decoder, 2021. [Online; accessed 11-August-2021, available at `https://docs.python.org/3/library/json.htmll`].

[31] Python Software Foundation. Miscellaneous operating system interfaces, 2021. [Online; accessed 11-August-2021, available at `https://docs.python.org/3/library/os.html`].

[32] Python Software Foundation. System-specific parameters and functionss, 2021. [Online; accessed 11-August-2021, available at `https://docs.python.org/3/library/sys.html`].

[33] Wes McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. Austin, TX, 2010.

[34] Leonard Richardson. Beautiful soup documentation. *April*, 2007.

[35] Python Software Foundation. Codec registry and base classes, 2021. [Online; accessed 11-August-2021, available at `https://docs.python.org/3/library/codecs.html`].

[36] John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90–95, 2007.

[37] Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585:357–362, 2020.

[38] Madeira F, Park YM, Lee J, Buso N, Gur T, Madhusoodanan N, Basutkar P, Tivey ARN, Potter SC, Finn RD, and Lopez R. The embl-ebi search and sequence analysis tools apis in 2019. *Nucleic Acids Research*, 47:636–641, 2019.

[39] PharmGKB and CPIC. Gene-specific information tables for cyp2d6, 2021. [available at `https://www.pharmgkb.org/page/cyp2d6RefMaterials`].

[40] Zhao H, Sun Z, Wang J, Huang H, Kocher JP, and Wang L. Crossmap: a versatile tool for coordinate conversion between genome assemblies. *Bioinformatics*, 30:1006–1007, 2014.

[41] S. Ben, R.M. Cooper-DeHoff, and H.K. et al. Flaten. Multiplex snapshot—a new simple and efficient cyp2d6 and adrb1 genotyping method. *Human Genomics*, 10:1–9, 2016.