UNIWERSYTET TECHNOLOGICZNO-PRZYRODNICZY
IM. JANA I JĘDRZEJA ŚNIADECKICH W BYDGOSZCZY
ZESZYTY NAUKOWE NR 271
TELEKOMUNIKACJA I ELEKTRONIKA 24 (2020) 5–11

ALGORITHM FOR GENETIC DATA ANALYSIS – COMPARISON OF THE FREQUENCY OF SPECIFIC MUTATIONS IN DIFFERENT POPULATIONS

Anna Marciniak[1,2], Martyna Tarczewska[2], Sylwester Kloska[1]

[1] Faculty of Medicine, Nicolaus Copernicus University in Toruń,
Ludwik Rydygier Collegium Medicum
ul. Jagiellońska 13-15, 85-067 Bydgoszcz, Poland
e-mail: {503015,503013}@stud.umk.edu.pl

[2] Faculty of Telecommunications, Computer Science and Electrical Engineering,
UTP University of Science and Technology,
Al. prof. S. Kaliskiego 7, 85-796 Bydgoszcz, Poland
e-mail:{annmar004, martar003}@utp.edu.pl

Summary: This paper presents a novel algorithm which can be used to analyze genomic data obtained during Next Generation Sequencing (NGS). Due to the interest in the subject among geneticists, it is necessary to develop algorithms and programs which analyze genetic data that will be user-friendly and accessible to people not related to typical bioinformatics. A way of performing comparative analyze, including proper data preprocessing and final data processing is described. Input data for the algorithm are annotated .vcf files. The outcome of presented algorithm is a file with counted percentage of single nucleotide polymorphisms (SNP) in data for every loaded population.

Keywords: Python 3, bioinformatics, .vcf files, genomic data

1. INTRODUCTION

With the rapid growth of genomic data amount achieved thanks to technical improvements in sequencing methods, scientists are drowning in information. Genomic research can provide a lot of useful information. For example, nucleotide sequence analysis can provide information regarding the location and structure of genes, as well as help in identification of regulatory elements. The information contained in the genome of an organism can also be used to predict the function of some genes or even the RNA secondary structure. However, with the diagnostic point of view, the most important information we can obtain from genomic sequence analysis is disease predisposition. Next Generation Sequencing (NGS) provides huge amounts of data [8]. However, these data are "raw". To be used as a source of useful information about the organization of genomes of different species or populations within a given species, they must first be processed. Theoretically, data processing could be done manually, however, it would be time-consuming and difficult task for a researcher. It requires precision, and even with accuracy it is impossible to avoid a mistake. The above-mentioned aspects show how

valuable genomic data can be and therefore, contribute to new discoveries in various disciplines, e.g. personalized genomics and precision medicine. Therefore, it is so important to develop time-effective and accurate bioinformatic methods that enable process automation [1, 4].

Data received straight from the sequencer needs to be processed. Until now, many tools have been created to facilitate data processing for bioinformaticians, so the data meet their needs. However, until now there was no application that would allow direct analysis of .vcf files (Variant Call Format), which is the format commonly used for storing gene sequence variations in bioinformatics. This format was created for the needs of large projects aimed at the thorough study of genomes, including 1000 Genomes Project, however, it is no longer maintained by project executors [5].

Another source of .vcf files is analyzing files in ANNOVAR [9]. This software allows utilization of the newest, updated genomic information. One of the most important functions of ANNOVAR are: 1) single nucleotide polymorphisms (SNP) identification and their contribution to protein sequence – if given SNP has an impact on amino acid order in a protein; 2) genomic regions variants identification, specific regions of genome which can be various protein binding sites; 3) identification of SNPs already listed in databases thanks to knowledge obtained from previously completed projects; 4) prediction of candidate genes which can contribute to occurrence of disease or other features. ANNOVAR is also a part of CoVaCS pipeline [6]. It is automated and precise workflow, which allows genotyping, variant calling and annotation for Next Generation Sequencing data. It is user-friendly, so it can be used by people who are not bioinformatic specialists. It has a wide range of possibilities, which is why it is a very useful tool for processing genomic data.

However, there are some things missing in those applications, which prompted the Authors to create a pipeline to sequencing data analysis with the use of scripts written in Python. At the end, the researcher receives a file with the compared populations.

Python is a programming language which is useful for both newcomers and experienced programmers. Its simplicity makes it quite easy to understand and learn for those, who are new to programming, on the other hand it provides all necessary tools to be useful for those, who have been programming in other languages before. Python has a wide range of programming applications, including data science, in which, together with R, is the dominant language in the field. Its great advantage is the fact that it allows for relatively quick processing of biological data files.

In this paper we will discuss original scripts for the analysis of genetic data (.vcf files) used to compare data (mainly the frequency of different SNPs) for different populations.

2. METHODS

The input data for the algorithm are .vcf files (an example sample file is provided below) (Fig. 1). Each row in a file describes a single SNP. The following file consists of 9 columns: chromosome, position (on the chromosome), ID of the SNP, reference allele, alternative allele, quality of SNP, filters, additional information, and format. To provide best results it is recommended to use already annotated files so SNP name is included. In order to compare the frequency of SNPs in different populations, the input data must contain at least 2 populations. There is no upper population limit; the algorithm can be duplicated any number of times for any number of studied populations.

```
##fileformat=VCFv4.1
##ALT=<ID=NON_REF,Description="Represents any possible alternative allele at this location">
##FILTER=<ID=HARD_TO_VALIDATE,Description="MQ0 >= 4 && ((MQ0 / (1.0 * DP)) > 0.1)">
##FILTER=<ID=LowCoverage,Description="DP < 5 ">
##FILTER=<ID=LowQD,Description="QD < 1.5 ">
##FILTER=<ID=LowQual,Description="Low quality">
##FILTER=<ID=LowQual,Description="QUAL > 30.0 && QUAL < 50.0 ">
##FILTER=<ID=SnpCluster,Description="SNPs found in clusters">
##FILTER=<ID=VeryLowQual,Description="QUAL < 30.0 ">
##FORMAT=<ID=AD,Number=.,Type=Integer,Description="Allelic depths for the ref and alt alleles in the order listed">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Approximate read depth (reads with MQ=255 or with bad mates are filtered)">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=MIN_DP,Number=1,Type=Integer,Description="Minimum DP observed within the GVCF block">
##FORMAT=<ID=PGT,Number=1,Type=String,Description="Physical phasing haplotype information, describing how the alternate alleles are phased in relation to one another">
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="Normalized, Phred-scaled likelihoods for genotypes as defined in the VCF specification">
##FORMAT=<ID=SB,Number=4,Type=Integer,Description="Per-sample component statistics which comprise the Fisher's Exact Test to detect strand bias.">
##INFO=<ID=ABHet,Number=1,Type=Float,Description="Allele Balance for heterozygous calls (ref/(ref+alt))">
##INFO=<ID=ABHom,Number=1,Type=Float,Description="Allele Balance for homozygous calls (A/(A+O)) where A is the allele (ref or alt) and O is anything other">
##INFO=<ID=AC,Number=A,Type=Integer,Description="Allele count in genotypes, for each ALT allele, in the same order as listed">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency, for each ALT allele, in the same order as listed">
##reference=file:///mnt/zgmvol/environment/commons/reference/hg19/hg19.fasta
#CHROM  POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  Sample
chr1    14907   rs79585140  A   G   810.77  HARD_TO_VALIDATE    AC=1;AF=0.500;AN=2;BaseQRankSum=2.424;DB;DP=45  GT:AD:GQ:PL 0/1:14,31:99:839,0,294
chr1    14930   rs75454623  A   G   1292.77 PASS    AC=1;AF=0.500;AN=2;BaseQRankSum=-3.434;DB;DP=51 GT:AD:GQ:PL 0/1:11,40:99:1321,0,178
chr1    14933   rs199856693 G   A   515.77  PASS    AC=1;AF=0.500;AN=2;BaseQRankSum=-2.600;DB;DP=55 GT:AD:GQ:PL 0/1:31,24:99:544,0,884
chr1    14976   rs71252251  G   A   417.77  HARD_TO_VALIDATE    AC=1;AF=0.500;AN=2;BaseQRankSum=0.447;DB;DP=51  GT:AD:GQ:PL 0/1:28,23:99:446,0,679
chr1    15029   rs201045431 G   A   159.77  HARD_TO_VALIDATE    AC=1;AF=0.500;AN=2;BaseQRankSum=-2.718;DB;DP=38 GT:AD:GQ:PL 0/1:27,11:99:188,0,721
chr1    15118   rs71252250  A   G   80.85   HARD_TO_VALIDATE    AC=1;AF=0.500;AN=2;BaseQRankSum=0.198;DB;DP=5   GT:AD:GQ:PL 0/1:1,4:17:109,0,17
chr1    16378   rs148220436 T   C   231.78  PASS    AC=2;AF=1.00;AN=2;DB;DP=9   GT:AD:GQ:PL 1/1:0,9:27:260,27,0
```

Fig. 1. A sample of .vcf file (Source: http://hgdownload.cse.ucsc.edu/goldenpath/hg19/chromosomes/)

2.1. Algorithms

2.1.1. Data preprocessing

The best way of using proposed algorithm is with a proper data preprocessing. The first part of the script is responsible for preparing data for second script. In this script input data are files for every population. In short, this script is responsible for generating a single file for each population as shown in flowchart (Fig. 2).

In order for script to work properly files for each population has to be placed in a single folder, which can be opened and analyzed in this part of a script. At this stage data are cleaned – at first column names and all information after "##" are removed, SNPs without name are removed (SNP marked as ".") as well as mutations with more than one alternative allele (length of this record is higher than 1). In this script also the genotype is decoded (using reference and alternative alleles). The result of this script is single file for each population with columns: sample name, chromosome, position, gene, SNP name, genotype. For purpose of drawing this flowchart we used column names instead of indexes.

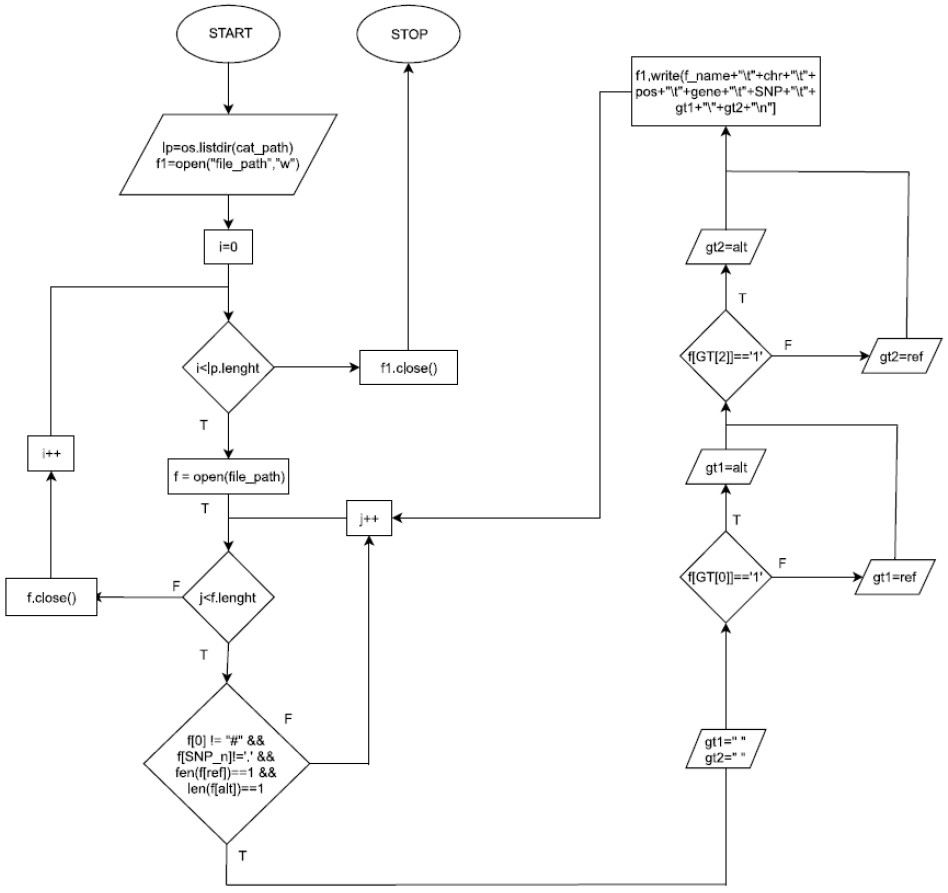

Fig. 2. Flowchart of algorithm of data preprocessing (Source: own study)

2.1.2. Data comparisons analyze algorithm

The main script analyzes every population file produced in data preprocessing step and write into new file only SNPs that occurs given times as shown below (Fig. 3). At first script analyzes SNP column in every population file; it counts how many times given SNP occurs in a population and calculates percentage. If the rate in every population is higher than threshold it is written to the new file. All SNPs below threshold are skipped. A result of this script is a single file with columns: chromosome, position, gene, SNP name, genotype and a column for every analyzed population with percentage.

For purpose of drawing this scheme population sample size was set to 50 and threshold was set to 70%.

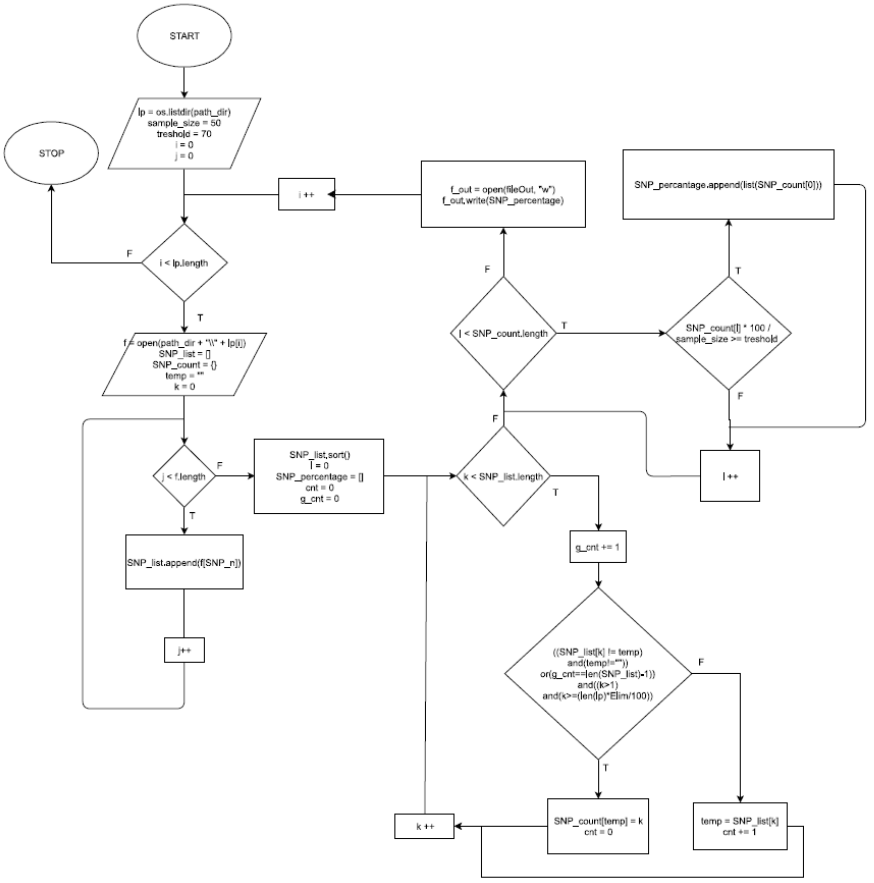

Fig. 3. Flowchart of algorithm of data comparison (Source: own study)

3. RESULTS AND DISCUSSION

By using this algorithm, you can easily perform a comparative analysis between different populations. It requires from the user to enter file paths, population size and threshold values. Depending on the amount of data entered, the calculation time may vary significantly, in case of 2 populations of about 50 samples, the result can be obtained in a few minutes. The resulting file can be opened in programs such as EmEditor, Microsoft Excel or Notepad ++. Further analysis of the data prepared in this way allows for the selection of variants specific for a given population (variants in one population that are absent in another) or for checking the occurrence of variants commonly considered as determining diseases.

There are already plenty of programs and scripts which are intended for various data analysis. However, they usually lack certain features and are therefore not universal. In addition, these programs often require payment for a license to use them, which also reduces their popularity. On the other hand, open-source programs are typically designed to run on Linux systems, which can be a hindrance for many unfamiliar users.

The script described in this article can be a solid work aid for population geneticists. Moreover, it can be used as a base for preparing batch files for other programs and more complexed analysis. An example software is VCFtools. It can be used to filter out specific variants, compare files, summarize variants, validate, and merge files or create intersections and subsets of variants. Thus, it is not suited for comparing large number of different population files. Despite many functionalities, using this program may be difficult due to its preparation mainly in the Perl programming language for Unix operating systems [7].

Another way of analyzing vcf files is using Galaxy website. It can be used to manipulate text files as well as vcf/bcf files. There are many tools for analyzing genomic data on this page. However, the main disadvantage of this solution is the data analysis directly in the browser, which means that in order to analyze the files, you must first upload them to the server [2]

Another example of software for analyzing .vcf files is VCF-Explorer. This software provides an easy-to-use environment. Additionally, users can define various types of queries, which are based on variant and sample genotype level annotations. Thus this software is very useful when analyzing .vcf files it is accommodated for analyzing single file, not entire population at once [3].

As shown above, researchers are constantly working on more effective and user-friendly software for .vcf files analyzes. Although many possibilities of genomic data analysis are still not described, this paper introduces one of the possibilities of analyzing these files.

REFERENCES

[1] Abel H.J., Duncavage E.J., 2013. Detection of structural DNA variation from next generation sequencing data: a review of informatic approaches. Cancer Genet.
[2] Afgan E., Baker D., Batut B., Van Den Beek M., Bouvier D., Čech M., et al., 2018. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Res.
[3] Akgün M., Demirci H., 2017. VCF-Explorer: filtering and analysing whole genome VCF files. Bioinformatics.
[4] Andrews K.R., Luikart G., 2014. Recent novel approaches for population genomics data analysis. Wiley Online Library.
[5] Auton A., Abecasis G.R., Altshuler D.M., Durbin R.M., Bentley D.R., Chakravarti A., et al., 2015. A global reference for human genetic variation. Nature.
[6] Chiara M., Gioiosa S., Chillemi G., D'Antonio M., Flati T., Picardi E., et al., 2018. CoVaCS: a consensus variant calling system. BMC Genomics.
[7] Danecek P., Auton A., Abecasis G., Albers C.A., Banks E., DePristo M.A., et al., 2011. The variant call format and VCFtools. Bioinformatics.
[8] Di Resta C., Ferrari M., 2018. Next generation sequencing: from research area to clinical practice. Ejifcc.
[9] Wang K., Li M., Hakonarson H., 2010. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. Nucleic Acids Res.

ALGORYTM DO ANALIZY DANYCH GENETYCZNYCH – PORÓWNANIE CZĘSTOTLIWOŚCI WYSTĘPOWANIA OKREŚLONYCH MUTACJI WŚRÓD RÓŻNYCH POPULACJI

Streszczenie

W artykule przedstawiono nowatorski algorytm służący do analizy danych genomowych uzyskanych podczas sekwencjonowania nowej generacji (NGS). Ze względu na zainteresowanie tą tematyką wśród genetyków konieczne jest opracowanie przyjaznych dla użytkownika i dostępnych dla osób niezwiązanych z bioinformatyką algorytmów i programów analizujących dane genetyczne. Opisano sposób przeprowadzania analizy porównawczej, w tym wstępne i końcowe przetwarzanie danych. Dane wejściowe algorytmu to pliki formatu .vcf z adnotacjami. Wynikiem przedstawionego algorytmu jest plik zawierający informacje dotyczące częstości występowania polimorfizmów pojedynczego nukleotydu (ang. *single nucleotide polymorphism*, SNP) w badanych populacjach.

Słowa kluczowe: Python 3, bioinformatyka, pliki .vcf, dane genomowe