SOFTWARE TOOL ARTICLE

SNPector: SNP inspection tool for diagnosing gene pathogenicity and drug response in a naked sequence

[version 1; peer review: 1 approved with reservations]

Peter T. Habib ¹, Alsamman M. Alsamman ², Sameh E. Hassanein ³, Ghada A. Shereif⁴, Aladdin Hamwiah¹

¹Department of Biodiversity and Crop Improvement, International Center for Agriculture Research in the Dry Areas (ICARDA), Giza, Egypt

²Department of Genome Mapping, Molecular Genetics and Genome Mapping Laboratory, Agricultural Genetic Engineering Research Institute (AGERI), Giz, Egypt

³Department of Bioinformatics & Computer Networks, AGERI, Agricultural Research Center (ARC), Giza, Egypt

⁴Faculty of Pharmacy, October 6 University, Giza, Egypt

V1 First published: 20 Dec 2019, 8:2133
<https://doi.org/10.12688/f1000research.21556.1>
 Latest published: 20 Feb 2020, 8:2133
<https://doi.org/10.12688/f1000research.21556.2>

Abstract

Due to the ability to diagnose diseases early and evaluate the effectiveness of medicinal drugs, single nucleotide polymorphism (SNP) identification receives significant interest. Detection and diagnosis of genetic variation through skill-less computational tools would help researchers reducing the severity of such health complications and improving well-tailored therapies using discovered and previously known information. We introduce SNPector, which is a standalone SNP inspection software, which can be used to diagnose gene pathogenicity and drug reaction in naked genomic sequences. It identifies and extracts gene-related SNPs, and reports their genomic position, associated phenotype disorder, associated diseases, linkage disequilibrium, in addition to various drug reaction information. SNPector detects and verifies the existence of an SNP in a given DNA sequence based on different clinically relevant SNP databases, such as NCBI ClinVar, AWESOME, and PharmGKB, and generates highly informative visualizations of the recovered information.

Keywords

SNP, Disease, Python, Bioinformatics

This article is included in the **Python** collection.

Open Peer Review

Approval Status

	1	2	3
version 2			
(revision)			
20 Feb 2020	view	view	view
version 1			
20 Dec 2019	view		

1. **Fakher Rahim** , Ahvaz Jundishapur University of Medical Sciences, Ahvaz, Iran
2. **Ka-Chun Wong**, City University of Hong Kong, Kowloon Tong, Hong Kong, China
3. **Tim Kacprowski** , Peter L. Reichertz Institute for Medical Informatics of TU Braunschweig and Hannover Medical School, Braunschweig, Germany

Any reports and responses or comments on the article can be found at the end of the article.

Corresponding authors: Peter T. Habib (p.habib911@gmail.com), Aladdin Hamwiah (a.hamwiah@cgiar.org)

Author roles: **Habib PT:** Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Project Administration, Resources, Software, Supervision, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Alsamman AM:** Investigation, Validation, Writing – Original Draft Preparation, Writing – Review & Editing; **Hassanein SE:** Supervision; **Shereif GA:** Validation, Writing – Review & Editing; **Hamwiah A:** Funding Acquisition, Supervision

Competing interests: No competing interests were disclosed.

Grant information: The author(s) declared that no grants were involved in supporting this work.

Copyright: © 2019 Habib PT *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Habib PT, Alsamman AM, Hassanein SE *et al.* **SNPector: SNP inspection tool for diagnosing gene pathogenicity and drug response in a naked sequence [version 1; peer review: 1 approved with reservations]** F1000Research 2019, 8:2133 <https://doi.org/10.12688/f1000research.21556.1>

First published: 20 Dec 2019, 8:2133 <https://doi.org/10.12688/f1000research.21556.1>

Introduction

In recent years, the number of cases of genetically originated diseases has increased, alarming the world and sparking interest in the development of precision medicine using molecular biomarkers. Single nucleotide polymorphisms (SNPs), the most common genetic difference among individuals, occurs in the human genome. These randomized modifications in DNA bases cause alterations in protein sequence residues of amino acids, thus altering their functions, which leads to different disease conditions in individuals¹. Several of these SNPs have been identified as disease-related genetic markers that have been used to recognize genes responsible for particular diseases in humans².

Distinguishing the evidence and the interpretation of a rich range of markers will be necessary to relate major alterations in the SNPs and to discover their connection in the progression of disease. Clarification of the phenotypic-associative mechanisms for these variations is therefore vital for comprehending the sub-atomic subtleties of disease origin and for developing novel therapeutic methods^{3,4}.

Although SNPs may exist in various areas of the gene, such as promoters, introns, 5'-and 3' UTRs, to date, most research has focused on disease-associated SNPs in coding regions or exons, especially non-synonymous SNPs, which may alter the biochemical ability of encoded proteins. In turn, altering gene promoters impact gene expression by changing transcription, binding transcription factor, methylation of DNA and modifications of histones. As a consequence, changes in gene expression, their impact on disease susceptibility, and drug responses can differ depending on the location of the SNP⁵⁻⁷.

With the expansion of genetic variants, different software could be used to generate new knowledge to support disease diagnosis and drug response studies and to develop new biomarkers for disease identification and drug customization. In this regard, a number of software applications have been developed in the last few years to classify, prioritize and evaluate the impact of genomic variants. For example, the **Ensembl Variant Effect Predictor** offers access to a large range of genomic annotations, with a variety of frameworks that answer different needs, with easy setup and evaluation methods⁸. Similarly, **SnpEff** categorizes the results of genome sequence variations, annotates variants according to their genomic location and estimates the coding effects. Depending on genome annotation, it is possible to predict coding effects such as non-synonymous or synonymous substitution of amino acids, stop codon gains or losses, start codon gains or losses, or frame changes⁹.

On the other hand, another tool, **PolyPhen-2**, assesses the potential impact of the genetic substitution of amino acids on the basis of physical, evolutionary comparative factors and model structural changes. Based on these profiles, the probability of a missense mutation becoming dangerous is measured on the basis of a combination of all these properties¹⁰. Similarly, **SIFT** calculates whether the substitution of amino acids affects protein activity, based on the homology of sequences and the physical properties of amino acids. It may be used for non-synonymous polymorphisms and laboratory-induced missense mutations that naturally occur, to effectively classify the effects of SNPs as well as other types, including multiple nucleotide polymorphisms¹¹.

Moreover, **Phyre2** is a web-based suite of tools for predicting and analysing protein structure, function and mutations. It has sophisticated remote homology identification methods to build 3D models, anticipate ligand binding sites, and evaluate the effect of amino acid variants, e.g. non-synonymous SNPs¹². **Missense 3D** uses the user-provided UniProt ID of the query protein, wild-type residue and substitution and other information to generate PDB residue mapping and predict the substitution effect on the 3D protein structure¹³.

To conclude the effect and possible phenotype of SNP, these software and web applications require minimum information such as SNP genomic position, SNP ID, allele form, and/or gene name. Acquiring this information requires using different computational tools, extensive time and some analysis skills. Most of the time, only gene sequences are available in which the SNPs are hidden without any additional information.

We therefore introduce **SNPector**, a standalone SNP inspection software that can be used to diagnose gene pathogenicity and drug reaction in naked genomic sequences. **SNPector** identifies and extracts gene-related SNPs, and reports their genomic position, associated phenotype disorder, associated diseases, linkage disequilibrium, in addition to various drug reaction information. It detects and verifies the existence of an SNP in a given DNA

sequence based on different clinically relevant SNP databases, such as NCBI ClinVar¹⁴, AWESOME¹⁵, and PharmGKB¹⁶. Lastly, it connects identified SNPs, related diseases and drugs, and produces numerous visualization figures to explain these relationships with the support of different Python modules.

Methods

Functions

The SNPector Python tool uses many packages to inspect the existence of SNPs in a given sequence. Moreover, SNPector provides users with detailed visualization figures, highlighting other SNPs with similar mutation effects on protein phosphorylation, ubiquitination, methylation, or sumoylation sites, and predicts substrates of N-acetyltransferase.

Additionally, SNPector provides the ability to visualize obtained information about the linkage disequilibrium of detected SNPs using various Python packages, such as Matplotlib¹⁷, generating a number of figures that summarize vast amounts of previously published data indicating SNPs allelic segregation, association, minor allele frequency. Figure 3 shows an example of illustrations that can be generated through SNPector.

Operation

SNPector requires at least Python 3.5, 16GB RAM, i7 cores, and 8 MB.

Workflow

SNPector was written using Python3 programming language as a standalone package and can be run on different operating systems platforms supported with Python 3.x compilers. To achieve user-friendly usage, the SNPector only accepts input from FASTA sequence (Figure 1) and can be operated from a console through simple command line (Figure 2).

SNPector uses different SNP record information collected from NCBI ClinVar (159,184 records), AWESOME (1,080,551 records), and PharmGKB¹⁸ (3,932 records). Ldlink is an online tool that can be used to assess linkage imbalance (LD) throughout ancestral populations and is a popular method of exploring population-specific genetic framework and functionally navigation disease susceptibility areas¹⁹. In SNPector, an Application

```
>EGFR
TAACACTTGGCGCTACCTGGACGACAGCCAATCTAGTTGGAATGAGAGTTAGAGGCCATAGTCTGTCTCAGC
TGGGAAAGCAGCTTTTATTCCAAGGTGTGCCAACCGAAAGGCCACATGTTATTGTCACAACCTGGTACCT
ACATCAGTGCTGACATCTTTAAGAACCTTAGAATTGGGAAATCAGTTTAGCCCTATCTGCATGTGTAGCC
GACAACCACACAATTGTTCCAACCTTGAGGTTGCATTAGAGCAACCTCATTTCCTCCCATCTCTGAGGA
AAAGCAGACCAGAGACGCTGGGTCAATCCAGAGTTATGGTTGGAAAAATGATGGAATAATTCTGCCCTG
GTGATAGGAGAGAGGGACTCCATCTTGTCAACTGTCATGGTTCCCATGTGAAAGCTATCATTATCACTGA
AATTGAATGAGAACACAGAAGGGAAGAACAGGGAATCCCCACAGAGTTAAAGAGGATGTGAAGATTGCT
TCATGTTTAAATGTTTGTGTAAGTGCTTTGGGTTGGTTATGTGCTGTCTGAACATGTGCTCATTTCATGG
CTCATTGAGAGGGCAGACAGTCCAATGATACTCTTTAGAATCATTCCCATGGGGAAGGAACAAAGAAGCC
TGTAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGTAAAGT
TGCTCTACCACTGGTTACTGTGATTGCAGAAAGTGCCTTTGAGATGAGGAAGAAGTGGGCAAGGCTCA
ATCCAACATCCAAAGCCAGAGGCCATATTTCTTCACTCTTAAGATAATTTGGGTTCAAATTATAGTCCCT
TTACACACTCTCTGCCTCAAAAGGCCCAAGACTCTCTTTGTTATGCTTGCCTAAACATGCCTTTCAAAG
AACTAGTTCTGTAAATACAACCTTTATTATAAACCTCTCCTTTGCTTTTAAAAATGGATCACCACGTCCAT
TTCTATGGTCCAACCTTTGTCCCTTAATTTAAAAATTTTTCTTGGATTAAGTTTGATGCCTTGAACATTA
GGAATCAAGCATACAAGATTGTATGCTGGTGGTGGGGAAGTAACTGTGCCTCCGCCTGTGCTGGGTGG
ATCAACATGGAGTGTGGACGAGCATAGGGATGTGTGGGTTTCTACTAGCTGAGAGTGTGTTTTAAATGTT
GTATTTTGATGTTTGTATTTTCTGAATATTCTACAGTTAGACCTTTGATTTATTCTTTGATGCATTCA
TTGAATAATATTTTAACTCCAGCCAGTTAGGTTTTTAATTTACACTTTTGTCCCTGATTTTAGGTGTA
GTGTTGTGTACACTACTGCCAGTGATGTTATGTTTGAACATTCATTGCACGCACAACAATGTGACT
```

Figure 1. Example input naked FASTA sequence.

```

/SNPector$ python3 scan_dna.py -blaston -modesearch -circoson -networkon -download -vis GivenSequence.fasta

```

A B C D E F G H I

Figure 2. SNPector command line parameters.

Figure 3. Collective figure to show all Illustrations provided by SNPector. (A) Circos illustration where other SNPs that have same proprieties are located. (B) Lollipop plot shows values by vertical columns (C) Counter Plot between two values creating a different coloured shade in which more contrast means higher value. (D) Numerical schematic showing the distribution between four values by plotting and scaling colour contrast according to other to values. (E) Heat map between SNP linkage disequilibrium matrix to show how two SNPs are linked. (F) Marginal plot combining column graph and plot, both showing the relationship between two values. (G) Dendrogram with heat-map showing how all SNP are linked to each other. (H) Histogram with box plot to compare visually between two values. (I) Plot illustrating the regression fit of two plotted value. (J) 3D plot of three values. (K) Annotated heat-map showing the plotted value.

Program Interface (API) has been programmed to download an LDhap file containing linkage disequilibrium statistics and potentially functional variants for a query variant resulting from the inputted FASTA sequence.

SNPector starts by running BLAST²⁰ software locally to find out the genomic location of a given DNA sequence on human genome. If successful, it retrieves the SNP records located within the query genomic range using NCBI ClinVar. According to retrieved records from the database, the detected SNPs in user-provided queries are marked as wild or mutated. Additionally, more information regarding detecting SNPs records will be retrieved from different implemented databases. This information will be used to generate different illustration figures.

If the process is successfully finished, SNPector will generate four different files: (A) Text file containing the output BLAST result, where the genomic location of the user-defined sequences is predicted; (B) tab delimited file containing SNPs retrieved by NCBI ClinVar located in the same regions; (C) two files regarding specific SNPs information retrieved from AWESOME and PharmGKB databases; (D) different figures depicting SNPs with a similar mutation effect to the detected SNPs located on other genomic regions, SNP linkage disequilibrium, the relationship between SNP, drug, and phenotype (Figure 3).

To achieve maximum user-friendly usage, SNPector can be run and controlled by command line. SNPector command line structure (Figure 2) is as follows: A) Python3 compiler; B) scan_dna.py: program main script that contains all functions; C) -blaston / -blastoff: in order to initiate BLAST process to provide sequence alignment against the genome to locate where the sequence is situated, if the -blastoff is chosen it will use previous BLAST results; D) -modescan: to scan the given sequence and find out whether SNP occurs or exists in sequence or not, and -modesearch: to extract all SNPs occur in this range of sequence regardless they exist or not; E) -circoson: draws a Circos figure to illustrate where SNPs with same properties/effect are located; F) -networkon: in order to link between SNPs, diseases and drugs and produces network HTML file; G) -download: activates the API to download data for identified SNPs from LDlink database; H) -vis: produces different figures and plots; I) GivenSequence.fasta: The user-provided sequence in FASTA file format. Any of the previous parameters can be deactivated when replaced with -off.

Use case

In this section we provide an example on how to use SNPector to extract SNPs from a naked given sequence without a reference sequence and how these extracted SNPs are linked to disease development and how they affect drug response. We show how to define the arguments of the SNPector function, interpret the results, and make visualizations.

Data

We use part of an EGFR gene sequence downloaded from NCBI nucleotide database in FASTA format as shown in *Extended data: File 1*²¹. The EGFR gene FASTA sequence submitted to NCBI contains SNPs that have a clinical effect involved in disease development, such as breast cancer. SNPector uses: (i) NCBI ClinVar database that describes SNP chromosome, position, ID, reference nucleotide, alternative nucleotide, quality, filter, and information to compare and detect SNPs in EGFR sequence that has clinical complications; (ii) PharmGKB database to investigate the SNP effect on disease development and drug response; (iii) AWESOME database to explore SNP effect on phosphorylation, ubiquitination, methylation, and sumoylation sites; and (iv) LDlink API database of SNP linkage disequilibrium to find out how detected SNPs are linked to other SNPs.

Loading SNPector libraries

SNPector uses different libraries to import, read, and read data and results. `os` library is used to run BLAST bash script:

```
import os
```

`time` is used to calculate the time that program.

```
import time
```

`re` refers to regular expression. This library sorts and splits input data with function `re.split()`.

```
import re
```

`itemgetter` module is used to sort BLAST data according to identity, mismatch, and p-value.

```
from operator import itemgetter
```

From `sys` library we used `sys.argv[]` to convert script to command line, which can be run and controlled from the terminal.

```
import sys
```

Then we import the Scripts package to visualize the data as follows:

```
from Scripts.Circos import DrawCircos
from Scripts.Network import DrawNetwork
from Scripts.Run_BLAST import RunBLAST
from Scripts.Extraction import ExtractSNP
from Scripts.APIcommands import APIcommands
from Scripts.Visualizations import visualization
from Scripts.DataVisualization.DownloadWithAPI.LDmatrix import LDmatrix
from Scripts.DataVisualization.DownloadWithAPI.LDhap import LDhap
from Scripts.DataVisualization.DownloadWithAPI.LDproxy import LDproxy
from Scripts.DataVisualization.CompleteScripts.Ready.ContourPlotWithSeaborn import
CounterPlot
from Scripts.DataVisualization.CompleteScripts.Ready.CustomLinearRegressionFitSeaborn
import LinearReg
from Scripts.DataVisualization.CompleteScripts.Ready.CustomLollipopPlot import Lollipop
from Scripts.DataVisualization.CompleteScripts.Ready.
DendrogramWithHeatmapAndColouredLeaves import DendoWithHeatMap
from Scripts.DataVisualization.CompleteScripts.Ready.DensityPlotWithMatplotlib import
DenistyPlot
from Scripts.DataVisualization.CompleteScripts.Ready.HistogramWithBoxPlot import
HistWithBoxPlot
from Scripts.DataVisualization.CompleteScripts.Ready.MarginalPlotWithSeaborn import
MarginalPlot
from Scripts.DataVisualization.CompleteScripts.Ready.ThreeDscatterplot import
ThreeDimPlot
from Scripts.DataVisualization.CompleteScripts.Ready.UseNormalizationOnSeabornHeatmap
import SeabornHeatMap
from Scripts.DataVisualization.CompleteScripts.Ready.AnnotatedHeatMap import
AnnoHeatMap
from Scripts.DataVisualization.CompleteScripts.Ready.NumericalSemantics import NumSchem
from Scripts.DataVisualization.CompleteScripts.Ready.ThreeDscatterplot import
ThreeDimPlot
from Scripts.DataVisualization.CompleteScripts.Ready.VolcanoLD import VolLD
```

SNPector variables

To sort data between the given sequence, Clinvar, AWESOME, BLAST, and PharmGKB, we implement the SNPector variables. This inherits the built-in function `open()` and nine variables are created as follows:

1. PharmGKB: data frame describes variant ID, gene name, type of effect, level of evidence, chemicals used to treat the phenotype, and phenotypes;
2. BLAST_RESULT: data frame lists BLAST output results of alignment of the given sequence against the human genome;
3. AwesomeDB: data frame lists SNPs chromosome, location, and properties, such as phosphorylation, ubiquitination, methylation, and sumoylation sites;

4. NCBIclinVar: data frame of SNPs that has clinical impact and involvement in disease;
5. SNPInDetails: data frame that lists the detected SNPs that SNPector found in the given FASTA sequence;
6. SNPInDetailsPharmGKB: data frame that lists detected SNPs and its impact on disease development and drug response;
7. SNPInDetailsAwesome: data frame that lists the properties of detected SNPs;
8. BLASTfile: function to open and read BLAST output results;
9. SeqFile: function to read the input file containing the sequence.

Each imported dataset can be found in *Extended data*²¹.

SNPector building blocks

RunBLAST() takes the file path of FASTA sequence and starts to align the sequence against the human genome and writes the results to BLAST_RESULT.txt (*Extended data: File 2*²¹).

```
os.system('./Scripts/blastn -query GivenSequence.fasta -db ./Data/Hum_Genom38 -outfmt 6 -out ./RESULTS/BLAST_RESULT.txt')
```

ExtractSNP() reads BLAST_RESULT.txt and sort its with *itemgetter()* according to the identity, length and p-value, then stores the start and end input given sequence (the query) and subject to use later in the extraction step. It also reads the input FASTA sequence file and stores the sequence variable to use in the comparing extraction step. SNPector provides two inspection modes that can be determined from the terminal, Search and Scan. If mode was “-modesearch”, then SNPector begins to extract all SNPs within the query start and end regardless of their existence in the query. In the mode “-modescan”, SNPector will extract only SNPs that exist in the query

SNPector begins to obtain the alternative nucleotides of SNP through the input sequence and obtains the nucleotides that range from SNP position in ClinVar minus the end position of the subject, to the SNP position in ClinVar minus the end position of the subject plus alternative SNP length to ensure the capture of SNPs from the given sequence and also to detect variants with length more than one nucleotide, finally storing it in the “*query_nuc_alt*” variable.

```
query_nuc_alt = sequence[int(snp_pos) - int(subject_end):int(snp_pos) - int(subject_end) + len(snp_alt)]
```

After the process of extraction the result saved to: “FromAwesom.tsv” file (*Extended data: File 3*²¹), in which SNPector list all other SNPs that have the same effect in different sites in proteins; “FromNCBI.tsv” (*Extended data: File 4*²¹), which is list of the SNPs that SNPector detects in a given sequence and retrieves from NCBI ClinVar Dataset; “FromPharmGKB.tsv” (*Extended data: File 5*²¹), which lista the effect of SNPs in disease development and drug response.

APIcommands() imports SNP IDs from “FromNCBI.tsv” and uses **Ldlink API** to download “LDhap.csv” file (*Extended data: File 6*²¹), which describes the allele frequency of extracted SNPs, “LDmatrix.csv” file (*Extended data: File 7*²¹), which shows how far detected SNPs are linked to other SNPs, and a file titles with the SNP id (e.g. rs516316.csv) (*Extended data: File 8*²¹), which includes additional information, such as minor allele frequency, linkage disequilibrium and distance of other SNPs linked to the detected SNP.

```
LDmatrix('./RESULTS/FromNCBI.tsv')
LDhap('./RESULTS/FromNCBI.tsv')
LDproxy('./RESULTS/FromNCBI.tsv')
```


DrawCircos() uses SNP properties from “FromAwesom.tsv” file (*Extended data: File 3²¹*) and searches for other SNPs that have the same properties. SNPector then imports pycircos package to draw SNP location on Circos (*Figure 3A*).

```
import pycircos
```

DrawNetwork() draw network using “FromPharmGKB.tsv” (*Extended data: File 5²¹*) to get the gene name (e.g. EGFR) and by gene name get all SNP that occur in this gene. Using SNP IDs, SNPector obtains disease names caused by these SNPs, and with the disease name SNPector can extract drugs used in treatment for this disease. Finally, with the drug name SNPector can obtain the clinical annotation of the drug. SNPector uses webweb package to draw the network and export it to .html file (*Extended data: File 10*).

```
from webweb import web
edge_list = Network
Web(edge_list).save("./RESULTS/%sVarPhenoDrugNetwork.html" % GeneName)
```

Visualization() uses data downloaded in the “LDmatrix.csv” file (*Extended data: File 7²¹*), and the SNP ID file (e.g. rs516316.csv) (*Extended data: File 8²¹*) to draw other figures (*Figures 3B–K*).

Discussion

SNPector can collect and retrieve information from the user-provided DNA sequence in the simplest way possible. By integrating different databases into SNPector, it is possible to detect the fluctuations in the abundance of SNPs in query through comparison with known variants of human genome. Such steps are accompanied by the use of online and verified sources to gather previously published details regarding target genomic regions and to generate highly informative visualizations of the recovered information.

Many tools, however, provide SNPs annotation, but they are still limited to the information provided (*Table 1*). SNPector, on the other hand, provides a new technique that extracts SNP from a naked sequence with no prior information. In addition, another benefit of SNPector is to annotate the discovered SNPs from information retrieved from various known databases.

Conclusion

One of the currently growing medical research paradigms is the diagnosis of genetic virulence that accumulates in our genome causing catastrophic health problems. Detection and diagnosis of genetic variation through skill-less computational tools would help researchers reducing the severity of such health complications and improving well-tailored therapies using discovered and previously known information.

SNPector provides and detects all available information about the disease-related SNPs in the given query with minimum user-provided information. It connects between different available information and produces various illustrations depicting SNP related diseases and treatment network, linked disequilibrium, minor allele frequency, similar SNPs with the same mutation effect and other information.

Table 1. Comparison between SNPector and published SNP annotation tools.

Software	SNPector	Ensembl VEP	PolyPhen-2	Missense 3D	SIFT	Snpeff	Phyre2
SNP detection from sequence	Yes	No	No	No	No	No	No
Disease and drug annotation	Yes	No	No	No	No	No	No

Software availability

Source code available from GitHub: <https://github.com/peterhabib/SNPector>

Archived source code as at time of publication: <http://doi.org/10.5281/zenodo.3558393>²².

License: MIT

Data availability

Underlying data

Homo sapiens chromosome 7, GRCh38.p13 Primary Assembly, Accession number NC_000007.14: https://www.ncbi.nlm.nih.gov/nuccore/NC_000007.14?report=fasta&from=55019017&to=55211628

Extended data

Zenodo: SNPector Supplementary Data, <http://doi.org/10.5281/zenodo.3569790>²¹.

This project contains the following extended data:

- Supplementary Files 1–10: output files from SNPector for the FASTA sequence use case (NC_000007.14).

Data are available under the terms of the [Creative Commons Attribution 4.0 International license](#) (CC-BY 4.0).

Acknowledgements

The authors are deeply grateful to Omar S. Abdel-Gaffar, teaching assistant at the College of Biotechnology, Misr University for Science and Technology.

A previous version of this article is available: <https://doi.org/10.1101/834580>.

References

- Chaudhary R, Singh B, Kumar M, *et al.*: **Role of single nucleotide polymorphisms in pharmacogenomics and their association with human diseases.** *Drug Metab Rev.* 2015; **47**(3): 281–90.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Kong J, Zhu J, Keyser UF: **Single molecule based SNP detection using designed DNA carriers and solid-state nanopores.** *Chem Commun (Camb).* 2016; **53**(2): 436–9.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Welter D, MacArthur J, Morales J, *et al.*: **The NHGRI GWAS Catalog, a curated resource of SNP-trait associations.** *Nucleic Acids Res.* 2014; **42**(Database issue): D1001–D1006.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Stranger BE, Stahl EA, Raj T: **Progress and promise of genome-wide association studies for human complex trait genetics.** *Genetics.* 2011; **187**(2): 367–83.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Schirmer MA, Lüske CM, Roppel S, *et al.*: **Relevance of Sp binding site polymorphism in WWOX for treatment outcome in pancreatic cancer.** *J Natl Cancer Inst.* 2016; **108**(5).
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Fan H, Liu D, Qiu X, *et al.*: **A functional polymorphism in the DNA methyltransferase-3A promoter modifies the susceptibility in gastric cancer but not in esophageal carcinoma.** *BMC Med.* 2010; **8**(1): 12.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Rintisch C, Heinig M, Bauerfeind A, *et al.*: **Natural variation of histone modification and its impact on gene expression in the rat genome.** *Genome Res.* 2014; **24**(6): 942–53.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- McLaren W, Gil L, Hunt SE, *et al.*: **The ensembl variant effect predictor.** *Genome Biol.* 2016; **17**(1): 122.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Cingolani P, Platts A, Wang le L, *et al.*: **A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of *Drosophila melanogaster* strain *w1118; iso-2; iso-3.*** *Fly (Austin).* 2012; **6**(2): 80–92.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Adzhubei I, Jordan DM, Sunyaev SR: **Predicting functional effect of human missense mutations using PolyPhen-2.** *Curr Protoc Hum Genet.* 2013; **76**(1): 7–20.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Ng PC, Henikoff S: **SIFT: Predicting amino acid changes that affect protein function.** *Nucleic Acids Res.* 2003; **31**(13): 3812–4.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Kelley LA, Mezulis S, Yates CM, *et al.*: **The Phyre2 web portal for protein modeling, prediction and analysis.** *Nat Protoc.* 2015; **10**(6): 845–58.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Ittisoponpisan S, Islam SA, Khanna T, *et al.*: **Can Predicted Protein 3D Structures Provide Reliable Insights into whether Missense Variants Are Disease Associated?** *J Mol Biol.* 2019; **431**(11): 2197–212.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Landrum MJ, Lee JM, Benson M, *et al.*: **ClinVar: public archive of interpretations of clinically relevant variants.** *Nucleic Acids Res.* 2016; **44**(D1): D862–D868.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Yang Y, Peng X, Ying P, *et al.*: **AWESOME: a database of SNPs that affect protein post-translational modifications.** *Nucleic Acids Res.* 2019; **47**(D1): D874–D880.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Thorn CF, Klein TE, Altman RB: **PharmGKB: the pharmacogenomics knowledge base.** *Methods Mol Biol.* In: *Pharmacogenomics*. Springer. 2013; **1015**: 311–20.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Habib PT, Alsamman AM, Hamwiah A: **BioAnalyzer: Bioinformatic Software of Routinely Used Tools for Analysis of Genomic Data.** *Biotechnology.* 2019; **10**(3): 33–41.
[Publisher Full Text](#)
- Hewett M, Oliver DE, Rubin DL, *et al.*: **PharmGKB: the**

- pharmacogenetics knowledge base. *Nucleic Acids Res.* 2002; 30(1): 163–5.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
19. Machiela MJ, Chanock SJ: **LDlink: a web-based application for exploring population-specific haplotype structure and linking correlated alleles of possible functional variants.** *Bioinformatics.* 2015; 31(21): 3555–7.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
20. Altschul SF, Madden TL, Schäffer AA, *et al.*: **Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.** *Nucleic Acids Res.* 1997; 25(17): 3389–402.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
21. Peter: **SNPector Supplementary Data [Data set].** *Zenodo.* 2019.
<http://www.doi.org/10.5281/zenodo.3569790>
22. peterhabib: **peterhabib/SNPector: SNPector: SNP inspection tool for diagnosing gene pathogenicity and drug response in a naked sequence (Version v1.0.0).** *Zenodo.* 2019.
<http://www.doi.org/10.5281/zenodo.3558393>

Open Peer Review

Current Peer Review Status: ?

Version 1

Reviewer Report 08 January 2020

<https://doi.org/10.5256/f1000research.23753.r58286>

© 2020 Rahim F. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Fakher Rahim

Research Center of Thalassemia and Hemoglobinopathy, Health Research Institute, Clinical Research Development Unit, Golestan Hospital, Ahvaz Jundishapur University of Medical Sciences, Ahvaz, Iran

1. The rationale for developing the new software tool was not clearly explained. Given lots of closed or similar tools such as SCOPA¹, SIFT, PolyPhen-2, and dbSNP, etc., I was expect more adding value of the present study to describe. One important concern associate with such databases is that these tools should get updated regularly with regards to recent GWAS etc. Ideally, these databases are manually curated.
2. I have concern about the format of output and interpretation of the results. I think the author should tested this tool even in very small dataset and report that.
3. One major point is that introducing a new tools need some strong evidences to compete with available and known tools. So the authors should compare the validity and SWOT aspects of this tools with the present tools. So, claiming that "SNPector provides and detects all available information" without a good and clear comparison is only lead to adding a tool to the previous sets of tools.

References

1. Mägi R, Suleimanov YV, Clarke GM, Kaakinen M, et al.: SCOPA and META-SCOPA: software for the analysis and aggregation of genome-wide association studies of multiple correlated phenotypes. *BMC Bioinformatics*. 2017; **18** (1): 25 [PubMed Abstract](#) | [Publisher Full Text](#)

Is the rationale for developing the new software tool clearly explained?

No

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

No

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

No

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bioinformatics and clinical epidemiology

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 09 Jan 2020

Peter Habib, International Center for Agriculture Research in the Dry Areas (ICARDA), Giza, Egypt

First of all, we thank the reviewer for his insightful comments/suggestions that have improved the quality of the manuscript.

1. We agree with the reviewer that the rationale for developing the new software tool was not clearly explained so, we edited the introduction and abstract to clarify the goal of SNPector. our idea was to extract the SNPs from a given sequence in FASTA format without the need to do several steps to extract those SNPs and visiting different databases to get the information of detected SNPs. with previous tools, you have to extract the SNPs by aligning against the genome, exporting those detected SNPs in the proper format which is compatible with other tools you need to study SNP effect, migrate the results from database to another to collect all allele frequency and linkage disequilibrium of each SNP, and take the matrices of linkage disequilibrium and excel sheets of allelic frequency and call different scripts and software to visualize the results. in SNPector, with only one command line SNPs extracted, Retrieve information related to drug response and disease development, Collect information of SNP structural effect of different protein critical sites (e.g. phosphorylation sites), downloading linkage disequilibrium and allelic frequency of detected SNP and other linked SNPs on the same chromosome, sorting the data in excel sheets, and finally visualize the downloaded data to be more understandable.

2. We already tested SNPector and the example provided in the paper is itself the testing. we downloaded the part of the EGFR gene, run SNPector, and the results were provided in the paper.

3. We agree with the reviewer so, we edited the discussion section and include comparing examples.

Competing Interests: No competing interests were disclosed.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research