SOFTWARE TOOL ARTICLE

GARCOM: A user-friendly R package for genetic mutation counts [version 1; peer review: 2 approved with reservations]

Sanjeev Sariya [1,2], Giuseppe Tosto [1-3]

[1]The Gertrude H. Sergievsky Center College of Physicians and Surgeons, Columbia University Medical Center, New York, NY, 10032, USA
[2]Taub Institute for Research on Alzheimer's Disease and the Aging Brain, Columbia University Medical Center, New York, NY, 10032, USA
[3]Department of Neurology College of Physicians and Surgeons, Columbia University Medical Center, New York, NY, 10033, USA

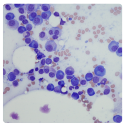
v1 **First published:** 01 Jul 2021, **10**:524
https://doi.org/10.12688/f1000research.53858.1
Latest published: 01 Jul 2021, **10**:524
https://doi.org/10.12688/f1000research.53858.1

Abstract

Next-generation sequencing (NGS) has enabled analysis of rare and uncommon variants in large study cohorts. A common strategy to overcome these low frequencies and/or small effect sizes relies on collapsing strategies, i.e. to bin variants within genes/regions. Several tools are now available for advanced statistical analyses however, tools to perform basic tasks such as obtaining allelic counts within defined genetics boundaries are unavailable or require complex coding. GARCOM library, an open-source freely available package in R language, returns a matrix with allelic counts within defined genetic boundaries. GARCOM accepts input data in PLINK or VCF formats, with additional options to subset data for refined analyses.

Keywords

mutation, plink, allele, genetics, VCF



This article is included in the Cell & Molecular Biology gateway.

This article is included in the RPackage gateway.

Open Peer Review

Approval Status ? ?

	1	2
version 1 01 Jul 2021	? view	? view

1. **Ettore Mosca**, Institute of Biomedical Technologies, Segrate (Milan), Italy

2. **Stephen M. Pederson**, University of Adelaide, Adelaide, Australia

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

Corresponding author: Giuseppe Tosto (gt2260@cumc.columbia.edu)

Author roles: Sariya S: Conceptualization, Formal Analysis, Methodology, Software, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Tosto G**: Conceptualization, Methodology, Project Administration, Resources, Software, Supervision, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This research was funded by National Institute of Aging R56AG069118, R56AG066889 (PI: Giuseppe Tosto). The funders had no role in study design, analysis, data collection, decision to publish or/and manuscript preparation.
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

How to cite this article: Sariya S and Tosto G. **GARCOM: A user-friendly R package for genetic mutation counts [version 1; peer review: 2 approved with reservations]** F1000Research 2021, **10**:524 https://doi.org/10.12688/f1000research.53858.1

First published: 01 Jul 2021, **10**:524 https://doi.org/10.12688/f1000research.53858.1

Introduction

Genome-wide association studies (GWAS) have led to the identification of several genomic common variants associated with complex diseases,[1] yet missing heritability remains extensive. Furthermore, most of the disease-causing variants are rare in nature[2] where common variants serve as a proxy. Rapid decline in sequencing costs have enabled in-depth analysis of rare variants (RVs; minor allele frequency < 1%) through Whole-Genome sequencing (WGS) and Whole-Exome Sequencing (WES). Furthermore, large-scale reference panels have allowed for imputation of RVs.[3-5] Power to identify statistically significant RVs decreases as the minor allele frequency decreases: therefore, an ideal method to overcome this limitation is to group RV at the gene/region level, usually via a collapsing test.

Despite the availability of sophisticated tools for annotation, quality-control and association analyses, tools to perform basic tasks, for instance, obtaining allelic count within defined genetic boundaries (genes and/or regions) are lacking, to our knowledge. R libraries such as BEDMatrix and bigsnpr[6] provide allelic counts for each SNP per individual but algorithms to extract information within genetic boundaries in a collapsed fashion are unavailable.

Here we introduce a user-friendly R package, GARCOM ("**G**enetic **A**nd **R**egional **C**ount **o**f **M**utations") that provides allelic counts per individual within user-provided genetics/regional boundaries.

Methods

GARCOM is written and developed in open-source R[7] statistical and programming language. GARCOM imports *data.table*,[8] *vcfR*,[9] *bigstatr*, *bigsnpr* and *stats* libraries for internal data transformation and processing. A stable version is released and publicly available on the CRAN repository.

install.packages("garcom")

Operation

GARCOM was developed on R (≥4.0) (RRID:SCR_017299) with other dependencies and minimum versions as: data.table (≥1.12.8), vcfR (≥1.12.0), bigsnpr (≥1.4.11). Full documentation of dependencies and installation is available at GARCOM github repository. There is no minimum memory (RAM) requirement as far as we know, but that may vary according to the nature and size of input genetics data. GARCOM was developed on Unix platform but can also be used on other platforms (e.g. Windows, Ubuntu).

Implementation

GARCOM operates through two main functions: "*gene_pos_counts*" accepts PLINK[10] (RRID:SCR_001757) input data, whereas "*vcf_counts_SNP_genecoords*" accepts VCF[11] input format. After reading in the data, these functions perform operations to count variants within genes/genomic regions for each individuals.

output <- gene_pos_counts(recoded_genetic_data, gene_boundaries, snp_locations)

output <- vcf_counts_SNP_genecoords(recoded_genetic_data, gene_boundaries, snp_locations)

where, "*output*" is the object generated by our library after a successful run of function; "*recoded_genetic_data*" is the main input file in PLINK or VCF formats; "*gene_boundaries*", and "*snp_locations*" are additional input files for gene and SNP information, respectively.

Typical workflow is shown in Figure 1. In brief, *the "gene_pos_counts*" function will process genetic input data ("*recoded_genetic_data"*) generated from the PLINK software through the --recode A option. Data are read in standard matrix format using the *data.table* R library. For VCF files, the "*vcf_counts_SNP_genecoords"* function reads the VCF input file employing the *extract.gt* function from the *vcfR* library. The genotype values are read within the "GT" field.

In addition to the --recode A genetic input, GARCOM needs genetic boundaries information and SNP information as shown in Table 2 and Table 3, respectively.

Output produced by GARCOM is a matrix, with *M* rows and *N* columns, where *M* represents the genes/genomic regions with at least one allele count and *N* represents the individuals. Genes with zero allelic counts across all individuals are excluded from the final output. Missing values are counted as zero in final output. When no allelic counts are present in the user-defined genes, NULL value is returned. GARCOM allows missing values (NA) in input data.

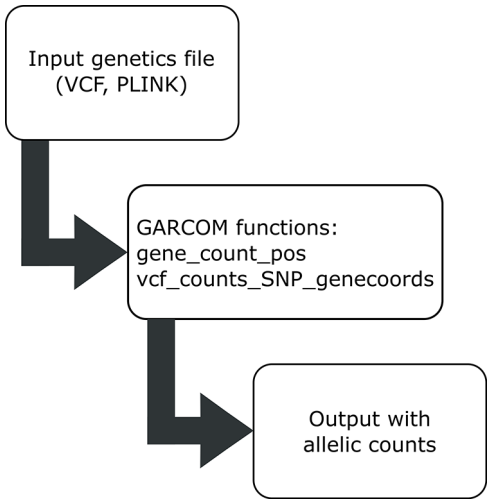


Figure 1. Workflow for standard GARCOM functions. GARCOM provides functions to read output from PLINK, VCF format. The final output is a matrix with rows as gene names and columns containing individuals' IDs.

In addition to the function described, GARCOM provides several options for user flexibility. For instance, GARCOM can be ran restricting analyses to 1) a list of genes, or 2) filter SNPs and extract individuals of interest. For instance, users can provide list of individuals using the "*keep_indiv"* parameter; similarly, genes can be filtered in by using the "*filter_gene*" parameter.

output <- gene_pos_counts(recoded_genetic_data, gene_boundaries, snp_locations, *keep_indiv=mylist.txt*)

output <- gene_pos_counts(recoded_genetic_data, gene_boundaries, snp_locations, *filter_gene=mysetofgenes.txt*)

Use cases

The input PLINK file has a matrix structure of *N* rows with *M* columns, where *N* rows represent individuals (one for each ID). The first six columns are family ID, individual ID, paternal ID, maternal ID, sex, and phenotype (standard output from PLINK (Table 1)). Following columns consist of the variants included in the analyses.

The input VCF file follows the standard VCF formats (please refer to the vcfR library documentation).

Toy data (gene and SNP coordinates) are shared within the package as "*genecoord*" and "*snppos*", respectively.

Table 1. Sample rows and columns for input genetics data recoded from PLINK software (--recode A).

FID	IID	PAT	MAT	SEX	PHENOTYPE	SNP1_A	SNP2_T	SNP3_G	SNP4_C	SNP5_C
FID1	IID_sample1	0	0	1	NA	1	1	0	NA	NA
FID2	IID_sample2	0	0	1	NA	0	1	0	NA	0
FID3	IID_sample3	0	0	1	1	0	0	1	0	0
FID4	IID_sample4	0	0	1	1	0	0	1	0	0
FID5	IID_sample5	0	0	1	1	0	0	1	0	0

Table 2. Sample data for genetics boundaries. Data must contain GENE, START and END column names.

GENE	START	END
GENE1	100	180
GENE2	200	400

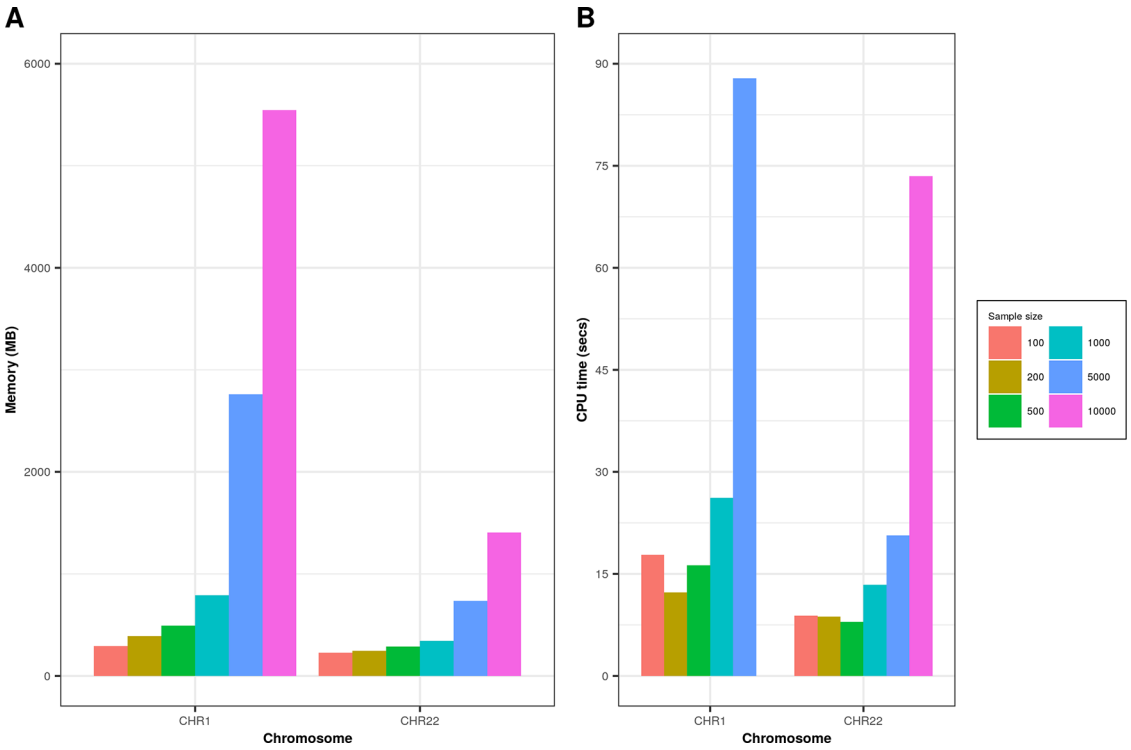


Figure 2. Comparison of memory (in MB) and CPU time (in seconds) for CHR1 and CHR22 on different sample sizes. Graph A represents memory consumption; graph B shows processing time in seconds for various sample sizes on CHR1 and CHR22.

Table 3. Sample data for SNP information where SNP and BP column names are must in input data, where SNP is single nucleotide polymorphism or variant and BP is base pair location.

SNP	BP
SNP1	100
SNP2	101
SNP3	201

Table 4. Sample output, where GENE column consists of gene names with corresponding individuals. Individual_ID1, Individual_ID2 and Individual_ID3 are sample individual IDs, where values represent allelic counts within gene for individual.

GENE	Individual_ID1	Individual_ID2	Individual_ID3
GENE1	10	2	1
GENE2	2	1	0

Simulation

We performed simulation on real data for CHR1 (# of variants = 23,456) and CHR22 (# of variants = 4,814) on randomly sampled individuals (N = 100, 200, 500, 1000, 5000, 10,000) extracted from whole-exome sequencing dataset as described in the study by Tosto et al.[12] Genetics data were recoded using PLINK --recode A flag. On both chromosomes we found increased memory consumption and time (Figure 2) as we increased the number of individuals processed. Memory consumption for CHR22 was significantly lower than CHR1 due to a smaller number of variants and genomic boundaries. Simulations were performed with 16GB memory (RAM) requested on computing cluster node.

All simulations were conducted on R (v4.0), data.table (v1.13.6) with default 16 threads, GARCOM (v1.40), bigsnpr (v1.6.1).

Discussion

GARCOM is easy to use where basic knowledge of R programming language is helpful but not desired. GARCOM is designed by harnessing existing libraries, such as *data.table,* that allow for efficient handling of large data. GARCOM data processing is independent of the reference genome build. GARCOM can be used on several platforms (e.g., Unix, Windows). GARCOM comes with certain limitations: genomic boundaries and variants' location need to be specified, as mentioned in the package documentation. In case of large-sized studies, for example UK biobank ($N \geq 200K$), processing data per chromosome is highly recommended due to memory limitations. Lastly, GARCOM depends on public and freely available R packages.

Future

VCF format can accommodate locus annotation performed by software such as ANNOVAR.[13] To this end, GARCOM plans to accommodate annotation filters in addition to the existing ones. One challenge associated with annotated VCF is the resulting large file size; we will try to add this functionality, keeping RAM limitations and processing time in mind. We plan to add features to handle bgen format which stores large amount of genetics data, with appropriate R library (http://www.well.ox.ac.uk/~gav/resources/rbgen_v1.1.5.tgz).

Data and software availability

Sample data associated with the package where applicable are provided within the library with proper documentation. No additional source data are required. We distribute the package under the MIT license. GARCOM can be downloaded from CRAN and GitHub from https://cran.r-project.org/web/packages/GARCOM/index.html and https://github.com/sariya/GARCOM respectively.

Reporting guidelines: Bugs and suggestions are welcome at the GitHub repository.

Author Contribution: SS, GT

Ethical Statement: Informed consent was obtained from all participants. For the whole-exome sequencing, the study protocol was approved by the Institutional Review Board (IRB) of Columbia university Medical Center (CUMC) (Approval number: AAAP0477). The study was conducted according to the principles expressed in the Declaration of Helsinki.

References

1. Frayling TM: **Genome-wide association studies: the good, the bad and the ugly.** *Clin Med (Lond).* 2014; **14**(4): 428–31.
PubMed Abstract | Publisher Full Text | Free Full Text
2. Gibson G: **Rare and common variants: twenty arguments.** *Nat Rev Genet.* 2012; **13**(2): 135–45.
PubMed Abstract | Publisher Full Text | Free Full Text
3. Sariya S, *et al.*: **Rare Variants Imputation in Admixed Populations: Comparison Across Reference Panels and Bioinformatics Tools.** *Front Genet.* 2019; **10**: 239.
PubMed Abstract | Publisher Full Text | Free Full Text
4. Vergara C, *et al.*: **Genotype imputation performance of three reference panels using African ancestry individuals.** *Hum Genet.* 2018; **137**(4): 281–92.
PubMed Abstract | Publisher Full Text | Free Full Text
5. Chou W-C, *et al.*: **A combined reference panel from the 1000 Genomes and UK10K projects improved rare variant imputation in European and Chinese samples.** *Sci Rep.* 2016; **6**(1): 39313.
PubMed Abstract | Publisher Full Text | Free Full Text
6. Prive F, *et al.*: **Efficient analysis of large-scale genome-wide data with two R packages: bigstatsr and bigsnpr.** *Bioinformatics.* 2018; **34**(16): 2781–87.
PubMed Abstract | Publisher Full Text | Free Full Text
7. Team, R.C. and R.F.f.S. Computing: *R: A Language and Environment for Statistical Computing*. Austria, Vienna; 2020.
8. Dowle M, Srinivasan A: *data.table: Extension of ?data.frame?* 2019.
9. Knaus BJ, Grunwald NJ: **vcfr: a package to manipulate and visualize variant call format data in R.** *Mol Ecol Resour.* 2017; **17**(1): 44–53.
PubMed Abstract | Publisher Full Text
10. Purcell S, *et al.*: **PLINK: a tool set for whole-genome association and population-based linkage analyses.** *Am J Hum Genet.* 2007; **81**(3): 559–75.
PubMed Abstract | Publisher Full Text | Free Full Text
11. Danecek P, *et al.*: **The variant call format and VCFtools.** *Bioinformatics.* 2011; **27**(15): 2156–8.
PubMed Abstract | Publisher Full Text | Free Full Text
12. Tosto G, *et al.*: **Association of Variants in PINX1 and TREM2 With Late-Onset Alzheimer Disease.** *JAMA Neurol.* 2019.
PubMed Abstract | Publisher Full Text | Free Full Text
13. Wang K, Li M, Hakonarson H: **ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data.** *Nucleic Acids Res.* 2010; **38**(16): e164.
PubMed Abstract | Publisher Full Text | Free Full Text
14. Band G, Marchini J: **BGEN: a binary file format for imputed genotype and haplotype data.** *bioRxiv.* 2018: p. 308296.
15. Mbatchou J, *et al.*: **Computationally efficient whole genome regression for quantitative and binary traits.** *bioRxiv.* 2020: p. 2020.06.19.162354.
16. Privé F, Arbel J, Vilhjálmsson BJ: **LDpred2: better, faster, stronger.** *bioRxiv.* 2020: p. 2020.04.28.066720.

Open Peer Review

Current Peer Review Status: ? ?

Version 1

Reviewer Report 29 November 2021

https://doi.org/10.5256/f1000research.57282.r98677

?

Stephen M. Pederson
Dame Roma Mitchell Cancer Research Laboratories, Adelaide Medical School, University of Adelaide, Adelaide, SA, Australia

The authors have provided four primary functions for summarising SNP counts using regions or genes as mapping architecture. The *data.table* package is known for its speed and this may lend a significant performance advantage to these functions, particularly when dealing with large numbers of samples or SNPs.

However, as it stands, this submission appears to fall short in a few key areas. Firstly, there is a typo in the installation instructions: **install.packages("garcom")** will cause an error. The correct instructions should be **install.packages("GARCOM")**.

The authors seem unaware of significant pre-existing infrastructure for range-based and SNP-based operations. The Bioconductor community has a wide-ranging suite of packages for robust mapping between genes, ranges, and SNPs, and no performance comparison, nor acknowledgment of these packages has been made. I suspect that using *data.table* may lend significant performance advantages, but the robustness of mapping between ranges provided by *GenomicRanges* is advantageous when performing rigorous research. For example, the lack of insistence on a CHR column for the *dt_gene* argument of *gene_pos_counts()* raises the distinct possibility of erroneous mapping between SNPs and genes/regions, and this should be addressed for robustness.

The authors also seem unaware of other packages which deal very efficiently with PLINK and VCF data, such as *seqArray* and *snpRelate.* In particular, these packages utilise the *gdsfmt* infrastructure, which is extremely efficient at working with large numbers of SNPs and samples. How does GARCOM compare by using the *data.table* backend? The provision of a new, lesser performing package will serve minimal purpose and this becomes an important question.

Whilst stating that the argument *dt_gene* follows PLINK format, an example of importing PLINK data would be most helpful.

A simple example vcf should also be included in the package so that examples for the two vcf-based functions can be run by any potential users. An example with 10 SNPs and 10 individuals, as per the examples for *gene_annot_counts* and *gene_pos_counts,* would fall within size limits for an R package. The fact that this has not been done also highlights the fact that the two vcf-based functions have not been included in the *test_that* suite of tests, rendering them open to future or even current bugs.

It is clear that the authors have found this package to be very useful for their work and their enthusiasm for making it available is commendable. However, I fear that the documentation is not adequate enough to invite new users. A *pkgdown* vignette appears to have been generated on the github repository but I could not find this vignette visible available anywhere and am unaware of its contents beyond skimming the visible HTML code. Please enable viewing of this via github.io pages which will take a matter of seconds. Making this visible (see https://sa-lee.github.io/plyranges for an excellent example), should enable new users to better comprehend the motivation and use of these functions. This would also enable the authors to include test coverage and other informative metrics.

I also note that the default behaviour of *gene_pos_counts()* is to drop genes with zero counts across all individuals, instead of returning all input genes. This is not stated anywhere in the documentation and should be, or alternatively, an argument allowing this behaviour to be modified should be added. Zero counts is not a non-result. It is easy to imagine scenarios where this is indeed important information.

The example data provided is also a little confusing. SNP1 in the object *snpgene* is mapped to both GENE1 and GENE4, and this relationship is not maintained in the *snppos* object. Is this meant to be SNP10 mapping to GENE4, or are the authors trying to highlight that multiple SNP to gene mappings are handled well by this package? If the latter, this may be a strong feature of the package and more detail should be provided for the handling of multi-mapping in the documentation. If, however, it is an oversight, this should be corrected so as not to confuse any future users.

As to the F1000 submission, I would strongly encourage the inclusion of executed code where possible, with results from actual data objects so readers can easily see what each function is doing. As it stands, users will have to install the package locally and run the examples to really understand what the package is offering. Given the lack of example vcf or parsing of PLINK objects, this may prove an insurmountable hurdle for some. If clear examples are provided, I suspect this will also increase the uptake of the package by other users, which is an important consideration that I'm sure the authors wish to see.

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

Partly

Is the description of the software tool technically sound?

Partly

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

No

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: Bioinformatician, R Package Developer

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.

Reviewer Report 18 October 2021

https://doi.org/10.5256/f1000research.57282.r95650

?

Ettore Mosca
National Research Council, Institute of Biomedical Technologies, Segrate (Milan), Italy

The authors present GARCOM, a tool for quantifying allelic counts within defined genetic boundaries. Overall, the article is well-written; the software is available in CRAN and GitHub, and it is well-documented. At the same time, the article is quite short.

I recommend adding some details to clarify how the proposed counting process works (e.g. it does not appear to keep track of the different haplotypes in different individuals) and its usefulness for genetic studies. This would clarify in which scenario this tool can be used and which kind of research questions it helps to address.

In the proposed examples, genetic coordinates are given without chromosomes. This suggests that the tool handles one chromosome at a time. Even if this is the case, it might be a dangerous approach, because it could increase the probability of mixing genetic coordinates from different chromosomes while the user prepares the input for GARCOM.

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

Is the description of the software tool technically sound?
Partly

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

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

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

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bioinformatics, NGS data analysis, network analysis, pathway analysis, metabolomics data analysis.

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.