SOFTWARE TOOL ARTICLE

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

Sanjeev Sariya1,2, Giuseppe Tosto1-3

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

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 RPackage gateway.

Open Peer Review

Reviewer Status

Invited Reviewers

1

version 1
01 Jul 2021

1. Ettore Mosca1, Institute of Biomedical Technologies, Segrate (Milan), Italy

Any reports and responses or comments on the article can be found at the end of the article.
Corresponding author: Giuseppe Tosto (gt2260@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.

Copyright: © 2021 Sariya S and Tosto G. 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: Sariya S and Tosto G. GARCOM: A user-friendly R package for genetic mutation counts [version 1; peer review: 1 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, yet missing heritability remains extensive. Furthermore, most of the disease-causing variants are rare in nature 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. 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 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 (“Genetic And Regional Count of Mutations”) that provides allelic counts per individual within user-provided genetics/regional boundaries.

**Methods**

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

```r
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 (RRID:SCR_001757) input data, whereas “vcf_counts_SNP_genecoords” accepts VCF input format. After reading in the data, these functions perform operations to count variants within genes/genomic regions for each individuals.

```r
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.
In addition to the function described, GARCOM provides several options for user flexibility. For instance, GARCOM can be run 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).**

<table>
<thead>
<tr>
<th>FID</th>
<th>ID</th>
<th>PAT</th>
<th>MAT</th>
<th>SEX</th>
<th>PHENOTYPE</th>
<th>SNP1_A</th>
<th>SNP2_T</th>
<th>SNP3_G</th>
<th>SNP4_C</th>
<th>SNP5_C</th>
</tr>
</thead>
<tbody>
<tr>
<td>FID1</td>
<td>IID_sample1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>NA</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>FID2</td>
<td>IID_sample2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>NA</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>NA</td>
<td>0</td>
</tr>
<tr>
<td>FID3</td>
<td>IID_sample3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FID4</td>
<td>IID_sample4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FID5</td>
<td>IID_sample5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>GENE</th>
<th>START</th>
<th>END</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENE1</td>
<td>100</td>
<td>180</td>
</tr>
<tr>
<td>GENE2</td>
<td>200</td>
<td>400</td>
</tr>
</tbody>
</table>
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.\textsuperscript{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).

**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.

<table>
<thead>
<tr>
<th>SNP</th>
<th>BP</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNP1</td>
<td>100</td>
</tr>
<tr>
<td>SNP2</td>
<td>101</td>
</tr>
<tr>
<td>SNP3</td>
<td>201</td>
</tr>
</tbody>
</table>

**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.

<table>
<thead>
<tr>
<th>GENE</th>
<th>Individual_ID1</th>
<th>Individual_ID2</th>
<th>Individual_ID3</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENE1</td>
<td>10</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>GENE2</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**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.
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 ≥ 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. 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/rgen_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

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.

---

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