SOFTWARE TOOL ARTICLE

Genomic variant annotation workflow for clinical applications [version 1; referees: 2 approved with reservations]

Thomas Thurnherr[1,2], Franziska Singer[2,3], Daniel J. Stekhoven[2,3], Niko Beerenwinkel[1,2]

[1]Department of Biosystems Science and Engineering, ETH Zurich, Basel, Switzerland
[2]SIB Swiss Institute of Bioinformatics, Basel, Switzerland
[3]NEXUS Personalized Health Technologies, ETH Zurich, Zurich, Switzerland

v1 **First published:** 12 Aug 2016, **5**:1963 (doi: 10.12688/f1000research.9357.1)
Latest published: 12 Aug 2016, **5**:1963 (doi: 10.12688/f1000research.9357.1)

Abstract

Annotation and interpretation of DNA aberrations identified through next-generation sequencing is becoming an increasingly important task. Even more so in the context of data analysis pipelines for medical applications, where genomic aberrations are associated with phenotypic and clinical features. Here we describe a workflow to identify potential gene targets in aberrated genes or pathways and their corresponding drugs. To this end, we provide the R/Bioconductor package rDGIdb, an R wrapper to query the drug-gene interaction database (DGIdb). DGIdb accumulates drug-gene interaction data from 15 different source databases and allows filtering on different levels. The rDGIdb package makes these resources and tools available to R users. Moreover, DGIdb queries can be automated through incorporation of the rDGIdb package into NGS sequencing pipelines.

This article is included in the Bioconductor channel.

This article is included in the RPackage channel.

Open Peer Review

Referee Status: ? ?

	Invited Referees 1	2
version 1 published 12 Aug 2016	? report	? report

1 **Christopher Southan**, University of Edinburgh UK

2 **Ankush Sharma**, National Research Council Italy, **Md. Sahidul Islam**, University of Rajshahi Bangladesh

Discuss this article

Comments (0)

Corresponding author: Niko Beerenwinkel (niko.beerenwinkel@bsse.ethz.ch)

How to cite this article: Thurnherr T, Singer F, Stekhoven DJ and Beerenwinkel N. **Genomic variant annotation workflow for clinical applications [version 1; referees: 2 approved with reservations]** *F1000Research* 2016, **5**:1963 (doi: 10.12688/f1000research.9357.1)

Grant information: This work was supported by EU Horizon 2020 PHC grant No. 633974 (SOUND – Statistical multi-Omics UNDerstanding of Patient Samples).
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: No competing interests were disclosed.

First published: 12 Aug 2016, **5**:1963 (doi: 10.12688/f1000research.9357.1)

Introduction

In recent years, next-generation sequencing (NGS) pipelines have been established and employed extensively in research settings. These efforts have helped tremendously to improve our understanding of genetic malignancies such as cancer. More recently, joint efforts of research groups and clinics aim to further enhance our knowledge of these malignancies for better diagnostic and treatment options. For example, the Cancer Genome Atlas (TCGA)[1] Consortium has sequenced several thousand samples of more than 20 different cancer types. One of the aims of this project is to better characterize different cancer types, for example through identification of distinct molecular sub-types.

There are also substantial efforts to move NGS technologies and pipelines into molecular diagnostics, for example, for the characterization of somatic variants of individual tumor samples through targeted panel sequencing. Targeted panel sequencing covers a specific set of genes or locations, typically between 50 and a few hundred. Panels focus on frequently mutated or otherwise altered genes or genomic locations. Currently, several generic cancer panels and panels for specific cancer types are available[2,3]. Based on the panel characterization, targeted therapies for the specific genetic aberrations can be applied.

The number of targeted therapies for cancer available today is still relatively small and their approval is typically limited to one or several cancer sub-types[4]. However, as the therapeutic options increase, more patients can benefit from these targeted therapies. As a consequence, several clinics or institutes developed and implemented molecular diagnostic approaches based on whole-exome and/or whole-genome sequencing[5–8]. Unlike targeted panels, whole-exome or whole-genome sequencing is not limited to a set of pre-selected genes, but allows for the detection of somatic aberrations across all protein coding sequences or the entire genome, respectively.

An exome- or genome-wide approach provides great advantage over targeted gene panels. They allow for a more comprehensive picture of the mutational landscape of a specific tumor. In addition, with more such data available and a better understanding of gene-gene and drug-gene interactions, prediction of drug efficacy as well as adverse drug reactions may become feasible. However, workflows based on whole-exome or whole-genome sequencing require clinical interpretation of the identified genetic variants. The result of an NGS pipeline is generally a list of genes harboring somatic variants or other genomic aberrations. To identify clinically actionable targets, these genomic aberrations need to be associated with drugs specifically targeting them.

Here we suggest a workflow to automate the identification of potential drug targets from a list of genomic aberrations, represented by a list of genes harboring them. For these genes, we mine drug-gene interactions using the drug-gene interaction database (DGIdb)[9]. DGIdb integrates drug-gene interactions from 15 different source databases. We provide the R/Bioconductor package `rDGIdb` (http://bioconductor.org/packages/rDGIdb/), which allows to efficiently integrate drug-gene annotation with NGS pipelines. `rDGIdb` can query DGIdb and filter results on different levels, i.e., source databases, interaction types, and gene categories. Through the `rDGIdb` package, drug-gene interaction mining can be automated and incorporated easily into NGS pipelines. Moreover, the `rDGIdb` package also provides functionality to visualize results.

Somatic mutation calling

Somatic variants or other genomic aberrations are identified from raw sequencing data and filtered using a standard NGS pipeline. The number of somatic variants might vary substantially, depending on the sequencing approach used and the levels or stringency of filtering employed. Next, somatic mutations are annotated with gene names, for which interacting drugs can then be queried through `rDGIdb`.

Identification of targetable aberrations

Provided a list of genes with genomic aberrations, we identify aberrations targetable with a drug or compound. The R/Bioconductor package `rDGIdb` provides functionality to query drug-gene interactions provided by DGIdb and to apply filtering on different levels.

R session setup

The package can be installed from an open R session. Instructions are provided on the `rDGIdb` Bioconductor page (http://bioconductor.org/packages/rDGIdb/). After installation of the package and all its dependencies, `rDGIdb` needs to be attached and a gene vector prepared. Gene names can be loaded from a text file or manually entered. The code below illustrates how to load gene names from a text file called `aberrated-genes.txt`, assuming the text file lists one gene symbol per line.

```
library("rDGIdb")
genes <- read.table("aberrated-genes.txt",
sep = "\t", header = FALSE, stringsAsFactors
= FALSE)
genes <- genes[,1]
```

Query drug-gene interactions

To query DGIdb, the `rDGIdb` package provides a simple query function, `queryDGIdb`. The function takes a vector of official gene symbols for which drug-gene interactions are to be queried. This is the only required argument to the query function, all other arguments are optional.

```
genes <- c("DDR2")
queryResult <- queryDGIdb(genes)
```

The function returns the query result as an object of type `rDGIdbResult`. The result is accessible through S4 methods. These methods format the result according to the result tabs provided on the DGIdb web interface. More specifically, the package provides four methods that return result data resembling the format provided through the DGIdb web interface, namely "Results Summary", "Detailed Results", "By Gene", and "Search Term Summary".

```
resultSummary(queryResult) # Summary table of the results
detailedResults(queryResult) # Detailed result table listing source and interaction type
byGene(queryResult) # Gene summary
searchTermSummary(queryResult) # Genes successfully mapped
```

An example output of `resultSummary` for the *DDR2* gene is shown in Table 1. The data can either be further processed using R or saved to a text file for analysis with other software tools.

Filter drug-gene interactions

Depending on the application, it may be desirable to filter for specific drug-gene interactions. The `rDGIdb` package allows filtering on the level of (1) source database, (2) gene category, (3) interaction type, and (4) other criteria, applied directly to the query result.

Filter by source database

DGIdb accumulates drug-gene interactions from 15 different source databases. These are summarized in Table 2. Depending on the application for which drug-gene interactions are queried, one or several source databases might be more relevant. The specific database or a group of databases to be queried is specified through the `sourceDatabases` argument. `rDGIdb` will only return hits listed in respective source databases. For example, the query below returns drug-gene interactions from databases: `MyCancerGenome` and `MyCancerGenomeClinicalTrials` only.

```
genes <- c("KRAS", "BRAF")
databases <- c("MyCancerGenome", "MyCancerGenomeClinicalTrials")
filter1 <- queryDGIdb(genes, sourceDatabases = databases)
```

The package provides a helper function that prints a list of all available source databases.

```
sourceDatabases()
```

Table 1. rDGIdb result summary of *DDR2* drug interactions. The number in the table indicates if a drug-gene interaction was found in a source database, where 1 means yes and 0 means no. Drug-gene interactions are sorted by their score, which is the total number of source databases listing the interaction.

Gene	Drug	Drug-Bank	MyCancer-Genome-ClinicalTrial	GuideTo-Pharmacology-Interactions	CIViC	DoCM	Score
DDR2	DASATINIB	0	1	0	1	1	3
DDR2	ERLOTINIB	0	0	0	1	1	2
DDR2	REGORAFENIB	1	1	0	0	0	2
DDR2	SORAFENIB	0	0	1	0	0	1

Table 2. Sources from which drug-gene interactions are accumulated in DGIdb.

Source	Link	Reference
CancerCommons	https://www.cancercommons.org	10
ChEMBL	https://www.ebi.ac.uk/chembl	11
CIViC	https://civic.genome.wustl.edu	12
ClearityFoundationBiomarkers	http://www.clearityfoundation.org	13
ClearityFoundationClinicalTrial	http://www.clearityfoundation.org/clinical-trials	13
DoCM	http://docm.genome.wustl.edu	14
DrugBank	http://www.drugbank.ca	15
GuideToPharmacologyInteractions	http://www.guidetopharmacology.org	16
MyCancerGenome	https://www.mycancergenome.org	4
MyCancerGenomeClinicalTrial	https://www.mycancergenome.org/clinicaltrials	4
PharmGKB	https://www.pharmgkb.org/	17
TALC	–	18
TEND	–	19
TdgClinicalTrial	–	20
TTD	http://bidd.nus.edu.sg/group/cjttd	21

Filter by gene category

Similarly, we can filter for specific gene categories. With the gene categories filter, drug interactions for genes with a specific category label can be queried. Examples of gene categories are `clinically actionable`, `kinase`, or `tumor suppressor`. The optional `geneCategories` argument can be used to filter by gene categories.

```
categories <- c("clinically
actionable","kinase", "tumor suppressor")
filter2 <- queryDGIdb(genes, geneCategories =
categories)
```

There are 41 different gene categories available. The following command lists all available gene categories.

```
geneCategories()
```

Filter by interaction type

Finally, the package provides filtering by interaction type. An interaction type is a label for the type of drug-gene interaction. 33 different interaction types are available and examples are: `activator`, `inhibitor`, `cofactor`, or `modulator`. The code below illustrates how to filter for specific interaction types.

```
interactions <- c("activator","inhibitor")
filter3 <- queryDGIdb(genes, interactionTypes =
interactions)
```

To print a list of all available interaction types, one can use the following method:

```
interactionTypes()
```

Manual filtering

Depending on the requirement of a specific application, additional filtering might be applied directly on the query results. For example, to increase confidence of results, drug-gene interactions might be filtered by setting a minimum cutoff on the score. As a result, only drug-gene interactions supported by a minimum number of source databases will be reported. Different score cutoffs may be employed, depending on whether the aim is to query interactions with support from multiple source databases or to include as many drug-gene interactions as there are available in the source databases. The example below illustrates how to filter out drug-gene interactions with only a single supporting source database from the result summary table.

```
subset(resultSummary(filter2), Score > 1)
```

Limitations of filtering

Although `rDGIdb` returns information on the type of interacting drug (such as `inhibitor`), to assist the follow-up interpretation of drug-gene interactions, querying and filtering through `rDGIdb` has limitations. For example, it is not possible to filter for specific drug-mutation interactions. That is, mutations in different locations of the same gene might have different biological effects in a cell or tumor. However, as querying is done on a gene level, mutations can not be distinguished. Additional expert knowledge or other approaches will have to be employed to exclude non-relevant drug-gene interactions from the query results.

Plotting of results

The package allows basic plotting of the results. Specifically, the number of interactions by source database can be visualized. An example plot is provided in Figure 1. This plot indicates which source databases report specifically large or small number of drug-gene interactions.

```
plotInteractionsBySource(filter2)
```

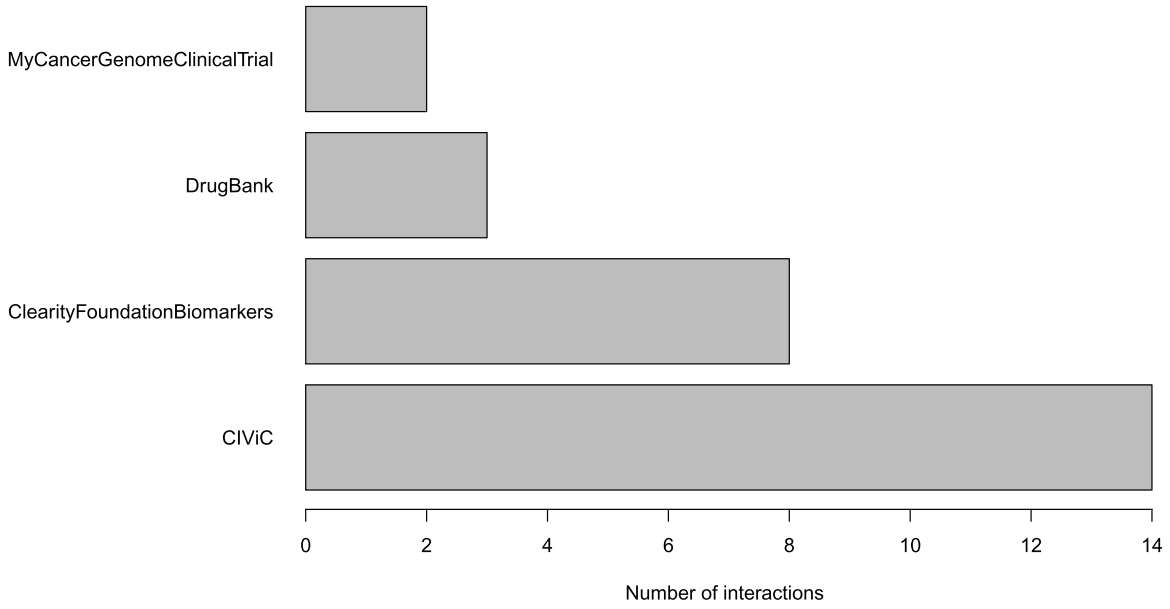

Figure 1. Example of the number of interactions by source for the KRAS gene.

Summary

We have described a workflow to identify potentially actionable genomic aberrations. More specifically, we have introduced the R/Bioconductor package `rDGIdb`, which provides an interface to query DGIdb using R. Given a list of genes with genomic aberrations, `rDGIdb` queries drug-gene interactions. The package allows filtering on different levels and visualization of the results. The `rDGIdb` package further includes detailed documentation and a vignette, which provides a step-by-step description of the workflow.

Package content and dependencies

`rDGIdb` depends on `jsonlite` and `httr`, which are available in R version 3.3.1 or higher. Briefly, `rDGIdb` queries the API provided by DGIdb (http://dgidb.genome.wustl.edu/api) using the `POST` function implemented in `httr`. Drug-gene interactions are returned by DGIdb in JSON format. Next, the data is deserialized into an R list object using the `jsonlite` package. Finally, the list is parsed and stored as an object of type `rDGIdbResult`. In order for `rDGIdb` to work, `jsonlite`, `httr`, and their dependencies need to be installed. A complete `sessionInfo()` output is provided below, which includes minimal version numbers of all dependencies.

- R version 3.3.1 (2016-06-21), `x86_64-apple-darwin13.4.0`
- Locale: `en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8`
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: rDGIdb 0.99.4
- Loaded via a namespace (and not attached): httr 1.1.0, jsonlite 1.0, R6 2.1.2, tools 3.3.1

Software availability

Software available from: http://bioconductor.org/packages/rDGIdb/

Latest source code: https://github.com/Bioconductor-mirror/rDGIdb

Archived source code as at time of publication: http://dx.doi.org/10.5281/zenodo.59253[22]

License: MIT license

Author contributions

TT and FS designed the query framework, tested the package, and wrote the manuscript. TT implemented the package. NB and DS supervised the work. All authors read and approved the manuscript.

Competing interests

No competing interests were disclosed.

Grant information

This work was supported by EU Horizon 2020 PHC grant No. 633974 (SOUND – Statistical multi-Omics UNDerstanding of Patient Samples).

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgments

The authors acknowledge Anja Irmisch and Mitchell Levesque from the University Hospital Zurich (USZ) for their valuable feedback on filtering and interpretation of drug-gene interactions.

References

1. Cancer Genome Atlas Research Network: **Comprehensive genomic characterization defines human glioblastoma genes and core pathways.** *Nature.* 2008; **455**(7216): 1061–1068.
PubMed Abstract | Publisher Full Text | Free Full Text
2. Khodakov D, Wang C, Zhang DY: **Diagnostics based on nucleic acid sequence variant profiling: PCR, hybridization, and NGS approaches.** *Adv Drug Deliver Rev.* 2016; pii: S0169-409X(16)30104-1.
PubMed Abstract | Publisher Full Text
3. Easton DF, Pharoah PD, Antoniou AC, *et al.*: **Gene-panel sequencing and the prediction of breast-cancer risk.** *N Engl J Med.* 2015; **372**(23): 2243–2257.
PubMed Abstract | Publisher Full Text | Free Full Text
4. Levy MA, Lovly CM, Pao W: **Translating genomic information into clinical medicine: Lung cancer as a paradigm.** *Genome Res.* 2012; **22**(11): 2101–2108.
PubMed Abstract | Publisher Full Text | Free Full Text
5. **Clinical translation: NCT promotes swift translation of innovative high-throughput diagnostics into clinical practice.** Accessed: 2016-06-22.
Reference Source
6. **The Caryl and Israel Englander Institute for Precision Medicine at Weill Cornell Medical College.** Accessed: 2016-06-22.
Reference Source
7. **MD Anderson Cancer Center.** Accessed: 2016-06-22.
Reference Source
8. **Personalized medicine at the Mayo Clinic.** Accessed: 2016-06-22.
Reference Source
9. Wagner AH, Coffman AC, Ainscough BJ, *et al.*: **DGIdb 2.0: mining clinically relevant drug-gene interactions.** *Nucleic Acids Res.* 2016; **44**(D1): D1036–D1044.
PubMed Abstract | Publisher Full Text | Free Full Text
10. Shrager J, Tenenbaum JM, Travers M: **Cancer Commons: Biomedicine in the internet age.** In *Ekins/- Collaborative Computational Technologies for Biomedical Research* Wiley-Blackwell; 2011; 161–177.
Publisher Full Text
11. Bento AP, Gaulton A, Hersey A, *et al.*: **The ChEMBL bioactivity database: an update.** *Nucleic Acids Res.* 2014; **42**(Database issue): D1083–D1090.
PubMed Abstract | Publisher Full Text | Free Full Text
12. **CIViC: Clinical Interpretations of Variants in Cancer.** Accessed: 2016-06-07.
Reference Source
13. The Clearity Foundation**.** Accessed: 2016-06-07.
Reference Source
14. **DoCM: Database of Curated Mutations.** Accessed: 2016-06-07.
Reference Source
15. Law V, Knox C, Djoumbou Y, *et al.*: **DrugBank 4.0: shedding new light on drug metabolism.** *Nucleic Acids Res.* 2014; **42**(Database issue): D1091–D1097.
PubMed Abstract | Publisher Full Text | Free Full Text
16. Pawson AJ, Sharman JL, Benson HE, *et al.*: **The IUPHAR/BPS Guide to PHARMACOLOGY: an expert-driven knowledgebase of drug targets and their ligands.** *Nucleic Acids Res.* 2014; **42**(Database issue): D1098–D1106.
PubMed Abstract | Publisher Full Text | Free Full Text
17. Whirl-Carrillo M, McDonagh EM, Hebert JM, *et al.*: **Pharmacogenomics Knowledge for Personalized Medicine.** *Clin Pharmacol Ther.* 2012; **92**(4): 414–417.
PubMed Abstract | Publisher Full Text | Free Full Text

18. Somaiah N, Simon NG, Simon GR: **A tabulated summary of targeted and biologic therapies for non-small-cell lung cancer.** *J Thorac Oncol.* 2012; **7**(16 Suppl 5): S342–S368.
PubMed Abstract | Publisher Full Text

19. Rask-Andersen M, Almén MS, Schiöth HB: **Trends in the exploitation of novel drug targets.** *Nat Rev Drug Discov.* 2011; **10**(8): 579–590.
PubMed Abstract | Publisher Full Text

20. Rask-Andersen M, Masuram S, Schiöth HB: **The druggable genome: Evaluation of drug targets in clinical trials suggests major shifts in molecular class and indication.** *Annu Rev Pharmacol Toxicol.* 2014; **54**(1): 9–26.
PubMed Abstract | Publisher Full Text

21. Zhu F, Han B, Kumar P, *et al.*: **Update of TTD: Therapeutic Target Database.** *Nucleic Acids Res.* 2009; **38**(Database issue): D787–D791.
PubMed Abstract | Publisher Full Text | Free Full Text

22. Thurnherr T: **rDGIdb: First release.** *Zenodo.* 2016.
Data Source

Open Peer Review

Current Referee Status: ? ?

Version 1

Referee Report 13 September 2016

doi:10.5256/f1000research.10075.r15658

? **Ankush Sharma**[1], **Md. Sahidul Islam**[2]

[1] Institute of Clinical Physiology, National Research Council, Siena, Italy

[2] Department of Statistics, University of Rajshahi, Rajshahi, Bangladesh

This R Package "rDGidb" is of immense usability for genomics and proteomics research community for integrating drug interactions with variations obtained from NGS data and researchers studying complex multi target drug-gene/protein interactions. The research article is written clearly and well organized except for results section which has a room for improvement.
The minor concerns are outlined as follows:-

1. We recommend authors to demonstrate results shown in Table 1 as a pictorial representation such as drug-gene interaction network to increase readability.

2. We suggest inclusion of the information related to Source Trust Level.

3. It would be nice to include query option using **r**eference **S**NP ID number ("**rs**" ID) or by chromosomal position of genomic aberrations obtained from Next Generation Sequencing pipeline to directly identify drugs associated with these clinically actionable variations.

We encountered a problem in installation of package "rDGidb"in R (version 3.3.1, release date 2016-06-21) with a warning message i.e. Package 'rDGIdb' is not available (for R version3.3.1).

We recommend authors to make "rDGidb" working and if this warning message is platform dependent, then please provide detailed documentation on software's or any updates needed in existing packages before installation of package.

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

Competing Interests: No competing interests were disclosed.

Author Response 10 Oct 2016

Thomas Thurnherr,

We would like to thank Ankush Sharma and Sahidul Islam for their time and effort to review our manuscript. The concerns are addressed below:

1. As suggested, we added a figure to the manuscript that illustrates drug-gene interactions of *DDR2.*

2. “Source trust level” is a parameter available through the DGIdb API, but not through the DGIdb web interface. The parameter can either be set to “expert curated” or “non-curated”. We did not include it as a parameter in rDGIdb for mainly two reasons: 1) We aim to reflect the web interface as consistently as possible and “Source trust level” is not available for DGIdb website queries; and 2) It is not clear which resources or drug-gene interactions are considered “expert curated” and which are not.

3. Thank you for the suggestion. Other packages implement variant call format (VCF) file import and annotation functionality. We recommend to use those. However, we added a paragraph to the manuscript (section “R session setup”) on that topic. Moreover, in the package vignette, we show how to employ the workflow with a VCF file as input. With this, we now provide a complete annotation workflow, from variants in VCF format to drug-gene interactions. Finally, we would like to point out that DGIdb queries are currently only possible on a gene level, but not on a variant level. Therefore, the association of a mutation in a specific position of the genome with a drug requires manual curation of the results obtained through rDGIdb.

Finally, the package is not yet available in the current Bioconductor release branch (version 3.3). This is likely the reason why you encountered an error while installing the package. The release is scheduled for October 2016 (version 3.4). The release will make the package available through the standard installation procedure. Installation instructions for packages in the development branch are provided on the Bioconductor website (https://www.bioconductor.org/developers/).

Competing Interests: No competing interests were disclosed.

Referee Report 18 August 2016

doi:10.5256/f1000research.10075.r15657

? **Christopher Southan**
IUPHAR/BPS Guide to PHARMACOLOGY, Center for Integrative Physiology, University of Edinburgh, Edinburgh, UK

This describes an R-based tool to query the drug-gene interactions in DGIdb. The paper is well written and the tool clearly has some utility. However, my reservations are outlined below.

1. As the application of NGS to cancer samples accelerates the resultant explosion of somatic variants threatens to swamp user’s ability to select them to input to this tool. What filters can be put in place to reduce huge aberration lists associated with passenger (i.e. probably non-causative and spurious) rather than driver mutations?

2. Given the latest Nature publication on the analysis of protein-coding genetic variation in 60,706 humans now available in the ExAC resource I suggest the utility emphasis for looking at germ-line vs somatic target aberrations should be re-balanced.

3. According to their website, DGIdb (v2.22 - sha1 aa9170e) was last updated 2016-02-21 and not all primary sources loaded were the latest versions even then. For example DrugBank is now up to

5.0 and GtoPdb is up to 2016.3 and it is not clear if it has only ChEMBL 20 rather than 21. Unless DGIdb can be prevailed upon to update more frequently and provide the release statistics of content, the utility of this tool is constrained because users cannot trust the results to be up to date.

4. The main goal of this tool for the identification of targetable aberrations will be confounded by the conflation of loss vs gain of function on both the target and drug sides. As we know, genetic aberrations are predominantly LOF but most drugs also negatively modulate their targets. This should be discussed and perhaps even made filterable in some way?

5. Why does Table 1 show such an apparently inconsistent mosaic of results? Reasons for discordance between the individual sources need to be explained.

6. Given this tool was developed by SIB would it be possible to add in Swiss-Var as an independent source via Swiss-Prot or NeXtProt?

I have read this submission. I 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.

Competing Interests: No competing interests were disclosed.

Author Response 10 Oct 2016

Thomas Thurnherr,

We would like to thank Christoper Southan for his thoughtful comments. Please find our response below:

1. There are several strategies that can be put in place to reduce the number of somatic aberrations to those which are likely relevant/functional. These include, but are not limited to: 1) Identification of recurrent mutations; 2) Prediction of the functional impact of a mutation; 3) Identification of recurrent combinations of mutations; and 4) Experimental validation. At this point, we would like to remark that the aim of rDGIdb is to annotate variants with potential drug-gene interactions and not to filter somatic variants. Other methods take care of filtering variants. Finally, rDGIdb is not limited in the number of genes to query. We query drug-gene interactions for several thousand genes at the time.

2. We agree that germline mutations play a role in a variety of diseases, specifically in cancer. Although rDGIdb is not limited to somatic mutations, we aim to identify potential targeted therapies. That is, drugs that specifically target malignant cells. Therefore, we think that considering somatic mutations rather than germline mutations is justified in this case.

3. We agree with the reviewer that DGIdb does not currently use the latest versions of all the resources it integrates. As a consequence, drug-gene interactions queried through rDGIdb might not agree with results from the most up-to-date resources. Results queried through rDGIdb are based on results from DGIdb and the resources it integrates. As a consequence, we have no control over how frequently resources are updated by DGIdb. However, we added a function to the package that prints the versions of all resource integrated by DGIdb. This helps the user to decide if the version available in rDGIdb/DGIdb is sufficient for the

intended purpose. The function is documented in the updated manuscript and in the package vignette.

4. The type of a drug-gene interaction can be filtered through an optional argument (interactionType) to the main query function. Possible values include suppressor, inhibitor, or activator. These limit reported drug-gene interactions to the interaction type of interest. Moreover, rDGIdb allows to query for specific gene categories, for instance tumor suppressor. These information/filters may help the user interpret the results provided by rDGIdb. Finally, we would like to point out that the interpretation of mutations in regard of their suitability as targetable mutations is beyond the scope of rDGIdb. All described filters and additional information can only assist the user in the interpretation of a specific mutation or interaction. The applicability of a certain therapy depends on a number of factors: cancer type, treatment history, and many others.

5. We selected an example that is brief enough to be presented as a table in the manuscript. *DDR2* seemed reasonable, with drug interactions in five different resources. The diversity of the drug-gene interactions in Table 1 can be explained by the diversity of these resources. For example, DrugBank lists experimental and approved drugs in any disease. In contrast, MyCancerGenomeClinicalTrials and CIViC list drugs in cancer only, which have either been approved by the authorities or are currently investigated through a clinical study. Finally, in the manuscript we mention that the most appropriate resources to be queried might depend on the application. We further explain how to filter for specific resources.

6. To our knowledge, SwissVar does not catalogue drug-gene interactions, but provides information on variants and their disease relations. At the moment, the main focus of the package is to report drug-gene interactions. However, we agree that SwissVar provides useful additional information on the genes queried through rDGIdb. Therefore, we consider an extension of the scope in regard to disease associations for a future release of the package.

Competing Interests: No competing interests were disclosed.