SOFTWARE TOOL ARTICLE

Increasing workflow development speed and reproducibility with Vectools [version 1; referees: awaiting peer review]

Tyler Weirick¹,², Raphael Müller²,³, Shizuka Uchida¹,²,⁴

¹Cardiovascular Innovation Institute, University of Louisville, Louisville, KY, 40202, USA
²Institute of Cardiovascular Regeneration, Goethe University Frankfurt, Frankfurt am Main, Hessen, 60590, Germany
³Institute for Bioinformatics and Systems Biology, Justus Liebig University Giessen, Giessen, Hessen, 35392, Germany
⁴Institute of Molecular Cardiology, University of Louisville, Louisville, KY, 40202, USA

Abstract
Despite advances in bioinformatics, custom scripts remain a source of difficulty, slowing workflow development and hampering reproducibility. Here, we introduce Vectools, a command-line tool-suite to reduce reliance on custom scripts and improve reproducibility by offering a wide range of common easy-to-use functions for table and vector manipulation. Vectools also offers a number of vector related functions to speed up workflow development, such as simple machine learning and common statistics functions.

Keywords
bioinformatics, reproducibility, workflow, vector, matrix, spreadsheet

Corresponding author: Shizuka Uchida (heart.lncrna@gmail.com)

Author roles: Weirick T: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Validation, Writing – Original Draft Preparation, Writing – Review & Editing; Müller R: Software, Writing – Review & Editing; Uchida S: Funding Acquisition, Project Administration, Resources, Supervision, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This study was supported by the start-up funding from the Mansbach Family, the Gheens Foundation and other generous supporters at the University of Louisville; University of Louisville 21st Century University Initiative on Big Data in Medicine (Z1762); and the Deutsche Forschungsgemeinschaft (SFB834 Z4).

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

Copyright: © 2018 Weirick T et al. This is an open access article distributed under the terms of the Creative Commons Attribution Licence, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Weirick T, Müller R and Uchida S. Increasing workflow development speed and reproducibility with Vectools [version 1; referees: awaiting peer review] F1000Research 2018, 7:1499 (doi: 10.12688/f1000research.16301.1)

Introduction

Although the importance of computational analyses in biological research is increasingly appreciated, many analyses are time consuming to implement and remain complicated, as well as being difficult to reproduce\(^1\). Workflow-managers [e.g., Snakemake\(^2\)] have greatly simplified many aspects needed for reproducibility. However, custom scripts (i.e., software not intended for use by a wider audience) remain a problem\(^3\). Custom scripts are often needed to further process data generated by high-use programs (i.e., programs intended for a wide user base). At the most basic level, analysis pipelines requiring custom scripts simply take more time to implement as additional code needs to be written. However, writing custom scripts also increases the chance of software bugs, which is concerning as even small bugs have led to retractions, such as mislabeling metadata\(^4\) or a sign change\(^5\). Furthermore, analyses using custom scripts also hamper reproducibility as the scripts may be publicly unavailable, lack documentation, or do not work on certain operation systems. To reduce the impact of these problems, we introduce Vectools\(^6\), a command-line tool for working with vectors, matrices, and tables. Vectools reduces the need for custom scripts by offering an easy-to-use command-line tool with a wide range of functionalities for manipulating tables, one of the most commonly used formats in bioinformatics. Further, Vectools incorporates a number of other useful vector-related functions, such as statistics and machine learning. Altogether, Vectools helps to speed up workflow development and improves reproducibility by offering a wide range of useful functions.

Methods

Installation

Vectools can be run via command-line by simply typing “vectools”, which will print the main help menu. Vectools contains over 45 operations organized by headings. These are

<table>
<thead>
<tr>
<th>Analysis</th>
<th>Descriptors</th>
<th>Manipulation</th>
<th>Math</th>
<th>Normalization</th>
<th>Supervised Learning</th>
<th>Unsupervised Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To run an operation, simply type “vectools” followed by the operation name. If the “—help” argument is added after an operation name, a help menu with usage instructions and examples will be printed.

Operation

A standard laptop computer with a recent version of Python3 will handle most applications.

Use cases

When manipulating data in tables, Core Utilities (Coreutils) programs (e.g., awk, grep, sed, and join) can be used instead of custom scripts. Using Coreutils helps to solve problems with availability as they are common to Unix-based systems. Here, we compared the usage of Vectools to Coreutils. Methods and output can be found in the archived data\(^7\). One downside of Coreutils programs is that they can be complex and difficult to understand. For example, joining multiple tables requires a Bash script using Coreutils-join, whereas this can be done with a single line with Vectools (Figure 1A). Furthermore, while common in Unix systems, the behavior of Coreutils programs may differ depending on the operating system. These differences can potentially cause errors or unexpected behavior, such as aggregating Gene Ontology (GO) terms by gene accession numbers with sed (Figure 1B). Instead of aggregating values on MacOS or other Berkeley Software Distribution (BSD) Unix systems, the Coreutils function prints the original input data. These errors can be caused by multiple reasons, such as BSD-sed not interpreting ANSI-C escape sequences (e.g., \n for newline, \t for tab) and differences in how regular expressions are evaluated. These problems can be overcome with Vectools.
with only one line of command. Vectools offers many functions that are currently unavailable in Coreutils, such as basic machine learning. Here, we show a simple example of using a support-vector machine to find potential novel carbonic anhydrases independent of sequence homology (Figure 1C). Carbonic anhydrases were chosen as they have multiple distinct classes, which arose via convergent evolution. Vectools significantly simplifies a number of steps needed for this task. For example, the “svmtrain” operation handles hyper-parameter tuning via grid search, k-fold testing, and independent set testing. This significantly simplifies implementing machine learning in analysis pipelines.

Discussion
Here, we show that Vectools reduces the need for custom scripts and is simpler to use than Coreutils. While Coreutils is faster and uses less memory, this is a minor issue given the increasing power and decreasing cost of computational resources. Vectools also offers various other functionalities, such as allowing easy incorporation of machine learning into analysis pipelines. Furthermore, Vectools helps to increase reproducibility by making analysis pipelines easier to share and reducing bugs. Users may also be interested in comparison with R. While certainly suited to the same tasks: 1) integrating R into a pipeline requires custom scripts; and 2) the use-cases for R and Vectools are different. R offers a large variety of functions at the cost of package dependency issues. Conversely, Vectools emphasizes ease-of-use by hosting a curated list of common functions. Thus, one common use-case of Vectools when combined with a workflow-manager is to replace work done in spreadsheets. This use-case offers a number of benefits. For example, it is in line with a recent technology feature in Nature, which argues that the concept of reproducibility extends to creating easy-to-update analysis pipelines. With Vectools, these easy-to-update pipelines will also be easy to share, making valuable tool for bioinformatics research.

Data availability
All data used in the paper are archived in Zenodo.

Software availability
Source code available from: https://vectools.bitbucket.io/


License: The software, and data and analysis pipelines are available under a Creative Commons Attribution 4.0 International (CC BY 4.0) license.

Grant information
This study was supported by the start-up funding from the Mansbach Family, the Gheens Foundation and other generous supporters at the University of Louisville; University of Louisville 21st Century University Initiative on Big Data in Medicine (Z1762); and the Deutsche Forschungsgemeinschaft (SFB834 Z4).

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

Supplementary material
Supplementary File 1. A list of operations offered by Vectools with short descriptions of their functions.

Click here to access the data

References

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