SOFTWARE TOOL ARTICLE

Optimized functional annotation of ChIP-seq data [version 1; peer review: 3 approved with reservations]

Bohdan B. Khomtchouk1-3, William C. Koehler4, Derek J. Van Booven5, Claes Wahlestedt6

1Department of Medicine, Stanford University School of Medicine, Stanford, CA, 94305, USA
2VA Palo Alto Health Care System, Palo Alto, CA, 94304, USA
3Department of Biology, Stanford University, Stanford, CA, 94305, USA
4Quiltomics, Palo Alto, CA, 94306, USA
5John P. Hussman Institute for Human Genomics, University of Miami Miller School of Medicine, Miami, FL, 33136, USA
6Center for Therapeutic Innovation and Department of Psychiatry and Behavioral Sciences, University of Miami Miller School of Medicine, Miami, FL, 33136, USA

Abstract

Different ChIP-seq peak callers often produce different output results from the same input. Since different peak callers are known to produce differentially enriched peaks with a large variance in peak length distribution and total peak count, accurately annotating peak lists with their nearest genes can be an arduous process. Functional genomic annotation of histone modification ChIP-seq data can be a particularly challenging task, as chromatin marks that have inherently broad peaks with a diffuse range of signal enrichment (e.g., H3K9me1, H3K27me3) differ significantly from narrow peaks that exhibit a compact and localized enrichment pattern (e.g., H3K4me3, H3K9ac). In addition, varying degrees of tissue-dependent broadness of an epigenetic mark can make it difficult to accurately and reliably link sequencing data to biological function. Thus, there exists an unmet need to develop a software program that can precisely tailor the computational analysis of a ChIP-seq dataset to the specific peak coordinates of the data and its surrounding genomic features.

geneXtendeR optimizes the functional annotation of ChIP-seq peaks by exploring relative differences in annotating ChIP-seq peak sets to variable-length gene bodies. In contrast to prior techniques, geneXtendeR considers peak annotations beyond just the closest gene, allowing users to investigate peak summary statistics for the first-closest gene, second-closest gene, ..., nth-closest gene whilst ranking the output according to biologically relevant events and iteratively comparing the fidelity of peak-to-gene overlap across a user-defined range of upstream and downstream extensions on the original boundaries of each gene’s coordinates. We tested geneXtendeR on 547 human transcription factor ChIP-seq ENCODE datasets and 198 human histone modification ChIP-seq ENCODE datasets, providing the analysis results as case studies. The geneXtendeR R/Bioconductor package (including detailed introductory vignettes) is available under the GPL-3 Open Source license

Open Peer Review

Reviewer Status

Invited Reviewers

<table>
<thead>
<tr>
<th>Invited Reviewers</th>
<th>version 1</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Vincent J. Carey</td>
<td>report</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Ruslan I. Sadreyev</td>
<td>report</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Michael Lawrence</td>
<td>report</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Any reports and responses or comments on the article can be found at the end of the article.
and is freely available to download from Bioconductor at:
https://bioconductor.org/packages/geneXtendeR/

Keywords
ChIP-seq, functional annotation, epigenetics

This article is included in the Bioconductor gateway.

Corresponding author: Bohdan B. Khomtchouk (bohdan@stanford.edu)

Author roles: Khomtchouk BB: Conceptualization, Data Curation, Formal Analysis, Funding Acquisition, Investigation, Methodology, Software, Validation, Visualization; Writing – Original Draft Preparation, Writing – Review & Editing; Koehler WC: Software, Validation; Van Booven DJ: Data Curation, Formal Analysis; Wahlestedt C: Project Administration, Resources, Supervision

Competing interests: No competing interests were disclosed.

Grant information: This work was supported by the American Heart Association (AHA) Postdoctoral Fellowship grant #18POST34030375 (Khomtchouk). This work was also partially supported by the Stanford Training Program in Aging Research grant (NIH/NIA T32-AG0047126) and the Army Research Office (ARO), National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a -- both awarded to BBK from 2014-2018. The content is solely the responsibility of the authors and does not necessarily represent the official views of the American Heart Association, National Institutes of Health, or Department of Defense.

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

Copyright: © 2019 Khomtchouk BB 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: Khomtchouk BB, Koehler WC, Van Booven DJ and Wahlestedt C. Optimized functional annotation of ChIP-seq data [version 1; peer review: 3 approved with reservations] F1000Research 2019, 8:612 (https://doi.org/10.12688/f1000research.18966.1)

First published: 02 May 2019, 8:612 (https://doi.org/10.12688/f1000research.18966.1)
Introduction
The field of epigenetic research studies the process by which heritable changes in gene expression occur without underlying alterations in the DNA sequence. Epigenetics plays a key role in human biology, and dysregulation in epigenetic processes is associated with the pathogenesis of cancer and many other diseases. Epigenetic mechanisms have been demonstrated to be necessary for biological programs that are important for a variety of health and disease outcomes. Understanding the impact of epigenetic architecture on the accessibility of gene promoters and its effect on gene expression patterns is therefore critical for linking chromatin biology to clinical indications. One way to measure such events involves investigating histone modifications, namely post-translational modifications to histones (referred to as chromatin marks) that regulate gene expression by organizing the genome into active regions of euchromatin, where DNA is accessible for transcription, or inactive heterochromatin regions, where DNA is more compact and less accessible for transcription.  

Chromatin marks come in a variety of different shapes and sizes, ranging from the extremely broad to the extremely narrow. This spectrum depends on a number of biological factors ranging from qualitative characteristics such as tissue-type to temporal aspects such as developmental stage. Depending on the peak caller used, computational factors such as the variance observed in peak coordinate positions (peak start, peak end) – both in terms of length distribution of peaks as well as the total number of peaks called – is an issue that persists even when samples are run at identical default parameter values. This variance becomes a factor when annotating peak lists genome-wide with their nearest genes as peaks can be shifted in genomic position (towards 5' or 3' end) or be of different lengths, depending on the peak caller employed. In total, the combined effect of these factors exerts a unique influence over the functional annotation and understanding of genomic variability, which ultimately complicates the study of epigenetic regulation of biological function.

Prior software in the chromatin immunoprecipitation-sequencing (ChIP-seq) functional annotation arena (e.g., ANNOVAR, GREAT, PAVIS, ChIPpeakAnno, ChIPseeker, annotator, HOMER, and BEDTools) has focused exclusively on dissection of peak coordinate positions (peak start, peak end) – both in terms of a range of extension parameters set by the user and a series of visualizations are returned as output to help users hone in on the optimal functional annotation. This is in contrast to most past and present epigenetic analyses, in both ChIP-seq and ATAC-seq studies, that assign gene body definitions (e.g., assigning a default 2 kbp as the cutoff for gene-proximal peaks) ad hoc before mapping the peaks to genomic features. Figure 1 shows why such a practice may be limiting.

From a performance standpoint, the extension algorithm is optimized to handle the computational complexity inherent to performing compute-intensive n-dimensional annotation. This ultimately aids in efficiently capturing cis-regulatory and proximal-promoter element relationships between ChIP-seq peaks and the genes they are (dys-)regulating, as described in further detail in the vignette. All of geneXtendeR’s source code is implemented in the C and R programming languages and shipped within a standalone R/Bioconductor package release that is publicly available for download from either Bioconductor or GitHub. Within its codebase, geneXtendeR leverages the AnnotationDbi, BiocStyle, data.table, dplyr, GO.db, networkD3, RColorBrewer, rtracklayer, SnowballC, testthat, tm, and wordcloud libraries.

Operation
Figure 2 summarizes the key steps of a sample workflow. For an end-to-end example of a comprehensive biological workflow and case-study, please refer to the vignette. An earlier version of this article can be found on bioRxiv (doi: https://doi.org/10.1101/082347).

Results
First, we tested geneXtendeR on all publicly available transcription factor and histone modification ChIP-seq datasets in ENCODE. After downloading and analyzing data from the ENCODE ChIP-seq Experiment Matrix (hg19), our large-scale analysis (Figure 1) indicated that ChIP-seq peaks do not concentrate within any specific upstream extension (e.g., 2000 bp) of their nearest protein-coding genes. This observation that ChIP-seq peaks drop off gradually with genomic distance from genes can often be a noisy process where an adjacent second or third-closest gene may constitute a more viable biological candidate, e.g., during cases of linked genes that are located close to each other. As such, the goal of geneXtendeR is to robustly link differentially enriched peaks with their respective genes, thereby aiding experimental follow-up and validation in designing primers for a set of prospective gene candidates during qPCR.
Figure 1. ENCODE ChIP-seq datasets. Large-scale computational geneXtendeR analysis using hg19 reference genome of 198 histone modification and 547 transcription factor ChIP-seq datasets from ENCODE. To make data directly comparable to each other, the y-axis represents a normalized count of peak clusters (number of peak clusters in a specific interval divided by the total number of peak clusters across all 0-10 kbp intervals for a given chromatin mark or TF), where a peak cluster is defined as a genomic locus harboring at least 5 overlapping peaks. The x-axis, which is segmented into 20 discrete regions ("1" = 0-500 bp interval, "2" = 500-1000 bp interval, ..., "20" = 9500-10000 bp interval), represents a genomic distance (in bp) of the closest protein-coding gene to each respective peak cluster. A steady decline in peak cluster count at further upstream intervals is detected for all (broad and narrow) chromatin marks as well as transcription factors, i.e., peak clusters do not congregate proximally within any specific region of intervals (e.g., 0-2000 bp) of their respective protein-coding genes, as there is a large number of peak clusters that reside further upstream of their nearest gene. For instance, in the 9500-10000 bp interval alone, there are 1043 peak clusters for the H2AFZ chromatin mark, 569 peak clusters for the H3K4me1 chromatin mark, and 716 peak clusters across all transcription factor ChIP-seq datasets. However, there are certainly exceptions like the H3K9me1 chromatin mark, which has only 1 peak cluster in the 7000-7500 bp interval (see the big dip at x-axis=15 in the right-hand panel) and only 7 peak clusters in the 9500-10000 bp interval (see S1_Appendix and S2_Appendix for reproducible code and data).
Figure 2. Sample biological workflow. Sample biological workflow using geneXtendeR in combination with existing statistical software to evaluate the role of ChIP-seq peak significance during functional annotation tasks (see description of hotspotPlot() function in package vignette). It is not uncommon for significant peaks to be located thousands of base pairs away from their nearest genes, suggesting that sequences under these respective peaks may further be extracted and analyzed for the presence of known regulatory elements or repeats (e.g., using software programs like TRANSFAC, MEME/JASPAR, or RepeatMasker) or for investigating potential enhancer effects.

the start of a gene (first exon) suggests that there is no good general guideline cutoff for capturing proximal histone modifications (e.g., prior studies have used 2000 bp) or transcription factor binding peaks. There are still hundreds of peak clusters that reside in proximal promoter regions that are 2000–3000 bp away from their nearest protein-coding genes and in distal regions beyond 3 kbp, making ad-hoc decisions like 2 kbp cutoffs too general to be of broad utility across specific use cases. When applying geneXtendeR to both proximal and distal transcription factor (TF) binding peaks for all cell types, we observed some cell type-dependent and TF-dependent peak aggregation dynamics in intervals ranging from 0 to 10 kbp (Figure 3). Similarly, examining distal peaks in representative plots of different chromatin marks in different cell types indicated that peaks indeed aggregate in a cell type and chromatin mark-dependent manner (Figure 4). S1_Appendix and S2_Appendix provide downloads to the complete compendium of all proximal/distal datasets analyzed from ENCODE.

We then focused our attention on using geneXtendeR to perform an end-to-end analysis of a published histone modification ChIP-seq dataset deposited in the Gene Expression Omnibus under accession number GSE83979. At the peak-calling stage (Figure 2) we ran two different peak callers (SICER and CisGenome) producing two highly variable peak length profiles even at default run parameters (Supplementary Figure 1). Despite the stark difference in peak profiles, geneXtendeR consistently identified the same top two gene candidates, highlighting its utility for robust functional annotation even in the face of extreme peak variability. Details are discussed in the package vignette.

We followed up this computational analysis by performing n-dimensional annotation of the GSE83979 dataset to provide an expanded view of the gene neighborhood around each individual peak – effectively annotating every peak n times (once for the closest gene, once for the second-closest gene, etc.) and grouping the results into a tabular summary format. We show in the vignette how the second-closest gene may be a preferable candidate for experimental follow-up/validation, especially if the first-closest gene is putative/predicted, while the second-closest gene is known to play a role in a similar biological process based on previously published literature.

Discussion

The cell-type and TF/chromatin mark-specific complexity apparent in Figure 3 and Figure 4 motivated the design and implementation of user-friendly functions that can calculate ratios of statistically significant peaks to total peaks in various genomic intervals (see hotspotPlot() documentation in geneXtendeR vignette). Similarly, users can transform peaks into merged peaks (see peaksMerge()). geneXtendeR also allows users to explore gene ontology differences at various extensions (see diffGO())
Figure 3. ENCODE TF analysis. Running geneXtendeR on 547 human transcription factor (TF) ChIP-seq datasets obtained from ENCODE shows that many peaks tend to reside within 500 bp upstream of their respective protein-coding genes yet, depending on the identity of the transcription factor (e.g., EP300) and the specific cell type (e.g., K562), there may be more or less peaks located further upstream and, therefore, a generalized upstream cutoff is not applicable.

as interactive network graphics (see makeNetwork()) or word clouds (see makeWordCloud()). Furthermore, users can investigate mean (average) peak lengths within any genomic interval (see meanPeakLengthPlot()), showing how average peak broadness can change at different upstream extensions, or examine the variance of peak lengths within a specific genomic interval (see peakLengthBoxplot()). It is also possible to examine unique genes and their associated ChIP-seq peaks between any two upstream extension levels (see distinct()). For example, Figure 5 displays all unique genes (and their respective gene ontologies) that are associated with peaks located between 2–3 kbp across the genome. geneXtendeR also allows users to examine the distribution of peak lengths across the entire peak set (see allPeakLengths()), a function that is useful for visualizing the
Running geneXtendeR on 198 human histone modification ChIP-seq distal peak datasets obtained from ENCODE reveals that most distal peaks are not congregating within any specific upstream region of their respective protein-coding genes (here we define "distal" as only those peaks that are more than 2000 bp away from their nearest gene). Additional comprehensive analyses (see S1_Appendix and S2_Appendix) were run for proximal peaks (≤ 2000 bp) as well as the complete set of peaks (proximal + distal) from all 198 histone modification ChIP-seq datasets, and similar patterns were observed.

length distribution of all peaks from a peak caller. These functions (and more) are all explored in detail within the package vignette. After a user has explored the peak coordinates data using these functions to determine the optimal alignment of peaks to a GTF file, the peaks file can be functionally annotated with the annotate() function or one of its counterparts (gene_annotate() or annotate_n()) for n-dimensional annotation.

We have successfully applied geneXtendeR during the analysis of a histone modification ChIP-seq study investigating the
neuroepigenetics of alcohol addiction\textsuperscript{40}, where \textit{geneXtendeR} was used to determine an optimal upstream extension cutoff for H3K9me1 enrichment (a commonly studied broad peak) in rat brain tissue based on line plots of both significant peaks and total peaks. This analysis helped us to identify, functionally annotate, and experimentally validate synaptotagmin 1 (Syt1) as a key mediator in alcohol addiction and dependence\textsuperscript{40}. This analysis is explored in detail in the package vignette. Taken together, \textit{geneXtendeR}'s functions are designed to be used as an integral part of a broader biological workflow (Figure 2).

**Conclusions**

We present an R/Bioconductor package, \textit{geneXtendeR}\textsuperscript{19}, that goes beyond the typical nearest-to-gene analyses commonplace to most standard computational ChIP-seq workflows. \textit{geneXtendeR} offers n-dimensional functional annotation and the ability to investigate the effect of variable-length gene bodies when mapping peaks to genomic features, thereby serving as a next-generation model of peak annotation to nearby features in modern bioinformatics workflows. \textit{geneXtendeR} therefore represents a critical first step towards tailoring the functional annotation of a ChIP-seq peak dataset according to the details of the peak coordinates (chromosome number, peak start position, peak end position) and their surrounding genomic features.

**Data availability**

**Underlying data**

A variety of different publicly available datasets were used to test \textit{geneXtendeR}. From ENCODE, a large-scale computational analysis using the hg19 reference genome was performed on 198 histone modification and 547 transcription factor ChIP-seq datasets. These transcription factor and histone modification ChIP-seq datasets in ENCODE are publicly available.
In addition, geneXtendeR was tested on a histone modification ChIP-seq dataset deposited in the Gene Expression Omnibus under accession number GSE83979.

Extended data
Zenodo: S1 Appendix. geneXtendeR analysis on 547 human TF ChIP-seq ENCODE datasets. https://doi.org/10.5281/zenodo.2646702


Extended data are available under the terms of the Creative Commons Zero “No rights reserved” data waiver (CC0 1.0 Public domain dedication).

Software availability
Software available from: https://bioconductor.org/packages/geneXtendeR/

Source code available from: https://github.com/Bohdan-Khomtchouk/geneXtendeR

Archived source code as at time of publication: https://doi.org/10.5281/zenodo.2646696

License: GNU General Public License-3

Author contributions
BBK conceived the study, designed the algorithms, implemented the code, performed the analyses, and wrote the manuscript. WCK and DJVB assisted with implementation and analysis. CW supervised the study.

Grant information
This work was supported by the American Heart Association (AHA) Postdoctoral Fellowship grant #18POST34030375 (Khomtchouk). This work was also partially supported by the Stanford Training Program in Aging Research grant (NIH/NIA T32-AG0047126) and the Army Research Office (ARO), National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a – both awarded to BBK from 2014–2018. The content is solely the responsibility of the authors and does not necessarily represent the official views of the American Heart Association, National Institutes of Health, or Department of Defense.

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

S1 Fig. SICER vs. CisGenome peak length distribution differences for GSE83979. Violin plot showing the differences in peak length distributions of the same ChIP-seq data (available through the Gene Expression Omnibus database, accession identifier GSE83979) analyzed with two separate peak callers (SICER and CisGenome) – despite significant differences in peak lengths generated by the two callers (i.e., peak variability), geneXtendeR’s gene_annotate() function can still robustly call top gene candidates consistently, as explained in the geneXtendeR package vignette.
References


34. https://genome.ucsc.edu/encode/dataMatrix/encodeChipMatrixHuman.html


Open Peer Review

Current Peer Review Status: ? ? ?

Version 1

Reviewer Report 10 June 2019

https://doi.org/10.5256/f1000research.20791.r48490

© 2019 Lawrence M. 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.

Michael Lawrence
Department of Bioinformatics and Computational Biology, Genentech Inc., South San Francisco, CA, USA

The article presents a new tool for finding the k-nearest neighbouring genes for a set of ChIP-seq peaks or other type of genomic feature. The idea is not particularly novel (it sounds a lot like bedtools closest -k, contrary to what the paper says), but the tool appears to be useful. Most of my concerns are around the organization of the article and how it describes the software.

The Introduction is well written and makes a good case for the tool. It could also incorporate some of the figures and data-driven arguments that come later (or those could go into the Discussion).

It is strange how the Methods section begins with Implementation (what about abstractly describing the method?) but even stranger how the Implementation section includes arguments for why the method is important (e.g., Figure 1).

The Results spends too much time arguing for why finding the k-nearest points is abstractly useful (there being no obvious cut-off). The paper would be strengthened by describing some interesting biological results, such as a meaningful/validated regulatory relationship not discovered by less flexible tools. Maybe these are described in the vignettes but it would be good to highlight them here. The actual examples can stay in the vignettes, but it would be nice to have the salient features described in this section, rather than Discussion. Since “optimized” is emphasized in the title, it would be good to have some details on performance here.

The Discussion should focus on limitations of the tool, potential integration points, and other topics that transcend the tool and method. It seems that the alcohol dataset belongs in Results.

I’m not sure I agree that this is a “critical first step” when there are many tools that find the closest gene; this one just finds the n-closest.

I wonder whether it would have been simpler (if a bit less efficient) to just find all genes within a wide margin of the peaks and then restrict those to the closest ‘k’. The iterative overlap finding, implemented in
C, sounds complicated. I'm also concerned about the package having so many dependencies, including both dplyr and data.table in addition to Bioconductor.

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

**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?**
Yes

**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:** Genomics

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.
marks that are involved in the regulation of gene activity often do not conform to standard cutoffs of distance to the transcription start site or gene body (confirmed in Fig. 1,3,4), and (b) enhancers and other regulatory elements often affect the activity of a gene that is not the closest to this element.

The manuscript provides a general justification of the approach and an overall description of the method itself. However, one part that can definitely be improved is the description of the results that the user can expect and a clear explanation, with examples, of how the user can interpret these results and generate specific biological hypotheses about the involvement of a protein or chromatin mark in the regulation of a specific gene or a group of genes. It seems that reference 40 includes an example of application of this method to a specific experimental dataset, and Fig. 5 is an example of functional annotation of promoters, but a clearly described user case showing the input, output, interpretation, and biological conclusions would be important. Another part that is unclear to me is how the user should interpret the multiple sets of 1st, 2nd, 3rd etc closest gene for a specific peak set. How one should approach selecting the most relevant gene set among these multiple options? An informative example may help clarify this point.

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

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?  
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, epigenetics, epigenomics, chromatin remodeling, chromatin structure

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.
Overall this is a clearly written paper, although I would take issue with the term "Optimized" in the title. What is “optimal functional annotation”? The abstract includes the phrase “precisely tailor the computational analysis of a ChIP-seq dataset to the specific peak coordinates of the data and its surrounding genomic features". This is a complicated objective and can be unpacked in many ways, specifically with respect to "computational analysis".

What the system brings to analysis of ChIP-seq data seems to be tunability and inclusiveness, in the important area of combinatorics of binding events and of histone modifications. ("Inclusiveness" pertains to allowing inspection of the order of proximities.) Can these be related to the term "Optimized" in the title?


Figure 1: Oscillations in a few traces in the left and right panels are probably artifactual. More stable estimates of the relationship between "normalized peak cluster count" and distance from TSS could be obtained using overlapping sliding windows.

I do not find Figure 2 particularly illuminating. The relationship of geneXtendeR and differential expression-oriented packages is not clear and is not described in the caption. It might be more informative to schematize the data structures for peak sets and how they lead to multi-sample hypothesis testing (e.g., with edgeR) or ontology/network inference.

Figure 3 is difficult to parse. Somehow a comparative interpretation is desirable, but all 4 panels look qualitatively similar. The y-axis ranges are different and perhaps log rescaling would be useful. Are the plotted points estimates, and if so, are uncertainty intervals of interest?

Figure 4 is similarly challenging. Are the oscillations seen after the jumps statistically meaningful? The y axis is labelled "differences". This is not explained in the caption.

Figure S1 should employ a logarithmic axis.

If we try example(makeNetwork) and then example(makeWordCloud) in the same session, a rat GTF is downloaded twice. BiocFileCache can be used to simplify user interactions with servers if these are needed.

In fact, the GTF file used by this package is available to Bioconductor users with the AnnotationHub package.

> ah = AnnotationHub::AnnotationHub()
> AnnotationHub::query(ah, c("gtf", "rattus", "84"))
AnnotationHub with 3 records
# snapshotDate(): 2019-05-02
# $dataprovider: Ensembl
# $species: Rattus norvegicus
# $rdataclass: GRanges
# additional mcols(): taxonomyid, genome, description,
# coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
# rdatapath, sourceurl, sourcetype
# retrieve records with, e.g., 'object["AH50914"]'

  title
AH50914 | Rattus_norvegicus.Rnor_6.0.84.abinitio.gtf
AH50915 | Rattus_norvegicus.Rnor_6.0.84.chr.gtf
AH50916 | Rattus_norvegicus.Rnor_6.0.84.gtf

If we use

> z = ah["AH50916"]
downloading 1 resources
retrieving 1 resource
|==========================================================| 100%

loading from cache
‘AH50916 : 57654’
Importing File into R ..

we have a cached version of the required annotation. Any package code requiring this GTF information can use

AnnotationHub::AnnotationHub()["AH50916"]

to get it. Thus:

> AnnotationHub::AnnotationHub()["AH50916"]
snapshotDate(): 2019-05-02
downloading 0 resources
loading from cache
‘AH50916 : 57654’
Importing File into R ..
GRanges object with 750896 ranges and 24 metadata columns:

```
  seqnames ranges strand |   source        type     score
     <Rle>  <IRanges>     <Rle> | <factor>     <factor> <numeric>
[1]  1 396700-409676   + | ensembl    gene     <NA>
[2]  1 396700-409676   + | ensembl  transcript <NA> ...
```

The peaks Input function writes a file "peaks.txt" to the current working directory! This is very poor form and could destroy user data. The function does not even include an option to write the file elsewhere. This content is then regarded as globally accessible to functions like make Network.
In summary the paper describes a number of utilities of potential interest, but essential statistical considerations should be enhanced. Downstream work such as network construction is entirely dependent on a fixed set of peak addresses, but the addresses must be associated with false discovery rates and/or boundary uncertainties. The discussion starts with "mark-specific complexity" apparent in Figures 3 and 4 but it is not clear that "complexity" is the right concept here. Different factors have different effects in different contexts, and distance to nearby gene is one component of context. To the extent that the paper gives users a mechanism for "determining the optimal alignment of peaks to a GTF file", I feel it is the concept of optimality raised here, and not the various functions that support "exploration", that should be detailed clearly in the paper. The optimization process should not be referred to the vignette. Once this optimality concept is stated precisely, the roles of the various functions can be usefully highlighted.

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?
Yes

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: Biostatistics, computational biology, clinical trials, epidemiology, statistical computing.

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