Intervene: a tool for intersection and visualization of multiple gene or genomic region sets

Aziz Khan[1,*] and Anthony Mathelier[1,2,*]

[1]Centre for Molecular Medicine Norway (NCMM), Nordic EMBL Partnership, University of Oslo, 0349 Oslo, Norway, [2]Department of Cancer Genetics, Institute for Cancer Research, Oslo University Hospital Radiumhospitalet, 0372 Oslo, Norway

*To whom correspondence should be addressed.

Abstract

Summary: A common task for scientists relies on comparing lists of genes or genomic regions derived from high-throughput sequencing experiments. While several tools exist to intersect and visualize sets of genes, similar tools dedicated to the visualization of genomic regions sets are currently limited. To fill this gap, we have developed Intervene, which provides an easy and automated interface for effective intersection and visualization of genomic region sets, thus facilitating their analysis and interpretation. Intervene contains three modules: *venn* to generate Venn diagrams of up-to 6 sets, *upset* to generate UpSet plots of more than 3 sets, and *pairwise* to compute and visualize intersections of genomic sets as clustered heatmap.

Availability and Implementation: Intervene is implemented in Python and R and is freely available at https://bitbucket.org/CBGR/intervene with a Shiny App at https://asntech.shinyapps.io/intervene

Contact: aziz.khan@ncmm.uio.no, anthony.mathelier@ncmm.uio.no

1 Introduction

Most of next-generation sequencing based high-throughput assays provide genomic region sets, which represent genomic locations for specific features such as transcription factor – DNA interactions, transcription start sites, histone modifications, or DNase hypersensitivity sites. A common task is to find similarities, differences, and enrichments between genomic region sets coming from different samples, experimental conditions, or cell and tissue types.

Several tools exist to perform genomic region set intersections, such as BEDTools (Quinlan and Hall, 2010), BEDOPS (Neph et al., 2012) and pybedtools (Dale et al., 2011) but tools for effective visualization of such intersections are limited (Zhu *et al.*, 2010; Dale *et al.*, 2011).

A common approach to represent intersection or overlap between different data sets, such as gene lists, is by using Venn diagrams. However, if the number of sets exceeds four, the Venn diagrams become complex and difficult to interpret. As an alternative approach, UpSet plots were introduced to depict the intersection of more than three sets (Lex *et al.*, 2014). However, if the number of sets exceeds ten, UpSet plots also become an ineffective way of illustrating set intersections. To visualize more than ten sets, one can represent pairwise intersections using a clustered heat map.

We developed Intervene, an automated tool to compute intersections of genomic region sets or gene lists (or any list of names) and visualize them as Venn diagrams, UpSet plots, or clustered heatmaps.

2 Intervene implementation

Intervene is implemented in Python and R and comes with a command line interface. Intervenes uses pybedtools (Dale *et al.*, 2011) to perform the intersection of genomic region sets and Matplotlib (Hunter, 2007), UpSetR (Lex *et al.*, 2014), and Corrplot (Wei and Simko, 2016) for visualization. Intervene requires genomic regions in BED, GFF, or GTF format, or lists of genes/names as input. It outputs publication quality figures, intersection matrices, and R scripts to further enable and facilitate plot customization.

3 Intervene modules

Intervene consists of three modules to compute and visualize the intersections of genomic region sets or lists, which are accessible through the subcommands *venn*, *upset*, and *pairwise*.

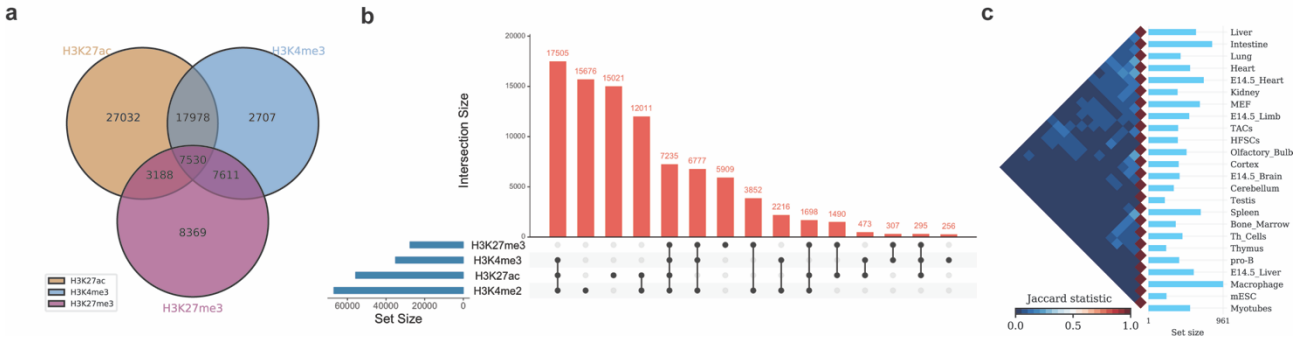

Fig. 1. Example of Intervene plots. *(**a**) A 3-way Venn diagram of ChIP-seq peaks of histone modifications (H3K27ac, H3Kme3 and H3K27me3) in hESC from ENCODE (Dunham et al., 2012) (**b**) UpSet plot of the intersection of four histone modification peaks in hESC (**c**) A heatmap of pairwise intersections terms of Jaccard statistics of super-enhancers in 24 mouse cell and tissue types from dbSUPER(Khan and Zhang, 2016).*

Intervene provides flexibility to the user to choose figure colors, label text, size, resolution, and type to make them publication standard. To read the help of any module, the user can type *intervene <subcommand> [venn, upset, pairwise] -h* on the command line. A detailed documentation is provided as Supplementary Material and is available at http://intervene.readthedocs.io/.

3.1 Venn diagrams

Venn diagrams are the classical approach to show intersections of sets. There are several web-based applications, tools, and R packages available to visualize intersections of up-to 6 list sets. However, a very limited number of tools are available to visualize in up-to 5-way classical Venn diagrams of genomic region intersections (Zhu *et al.*, 2010; Dale *et al.*, 2011). Here, we are providing up-to 6-way classical, Chow-Ruskey and Edwards' Euler/Venn diagrams to visualize the intersections of genomic regions or list sets. As an example, one might be interested to calculate the number of overlapping ChIP-seq peaks between different type of histone modification marks (H3K27ac, H3K4me3, and H3K27me3) (**Fig. 1a,** generated with the command *intervene venn --test*).

3.2 UpSet plots

When the number of sets exceeds four, Venn diagrams become difficult to read and interpret. An alternative and more effective approach is using UpSet plots to visualize the intersections. An R package and an interactive web-based tool are available at http://vcg.github.io/upset to visualize multiple list sets. However, there is no tool available to draw the UpSet plots for genomic region set intersections. Intervene's *upset* subcommand can be used to visualize the intersection of multiple genomic region sets using UpSet plots. As an example, we show the same intersections of ChIP-seq peaks as in Fig. 1a but for 4 sets using an UpSet plot, and ranked the interactions by frequency (**Fig. 1b,** generated with the command *intervene upset --test).* This plot is easier to understand than the 4-way Venn diagram (Supplementary Material).

One advantage of the UpSet plot is its capacity to rank the intersections and alternatively hide combinations with zero intersections, which is not possible using Venn diagrams.

3.3 Pairwise intersection heat maps

If the number of sets increases even more, visualizing all possible intersections becomes unfeasible by using Venn diagrams or UpSet plots. A possibility is to compute pairwise intersections and plot intersection ratios as a clustered heat map. Intervene's *pairwise* module provides several traditional and statistical approaches (Favorov *et al.*, 2012) to assess intersections, including number of overlaps, fraction of overlap, Jaccard statistics, Fisher's exact test, and distribution of relative distances. The user can choose from different styles of heat maps and clustering approaches. For example, one might be interested to calculate the pairwise intersection in terms of Jaccard statistics of the super-enhancers in several cell-types (**Fig. 1c,** generated using the command *intervene pairwise --test*).

4 Intervene Shiny App

Intervene also comes with a Shiny App to further explore and filter the results in an interactive way. Furthermore, Intervene's command line interface also gives an option to produce results as text files, which can be easily imported to the Shiny App for interactive visualization and customization of plots. The Shiny App is freely available at https://asntech.shinyapps.io/intervene.

Acknowledgements

We thank Marius Gheorghe for his useful suggestions and testing the tool.

Funding

This work has been supported by the Norwegian Research Council, Helse Sør-Øst, and the University of Oslo through the Centre for Molecular Medicine Norway (NCMM), which is part of the Nordic European Molecular Biology Laboratory partnership for Molecular Medicine.

Conflict of Interest: none declared.

References

Dale,R.K. *et al.* (2011) Pybedtools: A flexible Python library for manipulating genomic datasets and annotations. *Bioinformatics*, **27**, 3423–3424.

Dunham,I. *et al.* (2012) An integrated encyclopedia of DNA elements in the human genome. *Nature*, **489**, 57–74.

Favorov,A. *et al.* (2012) Exploring massive, genome scale datasets with the genometricorr package. *PLoS Comput. Biol.*, **8**.

Hunter,J.D. (2007) Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.*, **9**, 99–104.

Khan,A. and Zhang,X. (2016) dbSUPER: a database of super-enhancers in mouse and human genome. *Nucleic Acids Res.*, **44**.

Lex,A. *et al.* (2014) UpSet: Visualization of intersecting sets. *IEEE Trans. Vis. Comput. Graph.*, **20**, 1983–1992.

Neph,S. *et al.* (2012) BEDOPS: High-performance genomic feature operations. *Bioinformatics*, **28**, 1919–1920.

Quinlan,A.R. and Hall,I.M. (2010) BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, **26**, 841–2.

Wei,T. and Simko,V. (2016) corrplot: Visualization of a Correlation Matrix. R package version 0.77.

Zhu,L.J. *et al.* (2010) ChIPpeakAnno: a Bioconductor package to annotate ChIP-seq and ChIP-chip data. *BMC Bioinformatics*, **11**, 237.