Published online DD MM YYYY

Nucleic Acids Research, YYYY, Vol. xx, No. xx ***1–7***
doi:10.1093/nar/gkn000

Visualizing (differential) expression patterns with fuzzy concepts as FlowSets

Felix Offensperger,[1,*], Markus Joppich[1,+,*], and Ralf Zimmer[1,+]

[1] LFE Bioinformatics, Institute for Informatics, Amalienstr. 17, 80333 München, Germany
To whom correspondence should be addressed: {offensperger|joppich|zimmer}@bio.ifi.lmu.de
[+,*] equal contribution

Received YYYY-MM-DD; Revised YYYY-MM-DD; Accepted YYYY-MM-DD

ABSTRACT

With the broad accessibility of (single cell) RNA sequencing, the analysis of large, complex and highly differential expression data sets becomes increasingly important. Commonly, differential gene expression along relevant dimensions with several conditions is performed to identify and understand the disease-causing differences, eventually in combination with gene set enrichment analysis. However, identifying the underlying processes from such scattered analyses might be bothersome.

While there are concepts for analysing temporal processes, these often interpret real or pseudo-time, which might not match discrete and categorical conditions actually measured in experiments. Moreover, many methods work on (average) gene expression, which does not resemble the whole variability within a (sc)RNA-seq experiments, as the zero-inflated gene expression of single cells, or variability of replicates, is not reflected accurately.

Here, we present FlowSets, a framework for the analysis of discrete-series expression data using fuzzy concepts. FlowSets allows a flexible analysis of multi-dimensional data sets and the identification of patterns (flows) of gene expressions or fold-changes between several conditions at one glance. In order to facilitate an easily interpretable overview of such patterns, expression values or fold changes are systematically discretized into linguistic variables (fuzzy values). FlowSets then visualizes the genes' change between the states of a series of conditions. Moreover, flows can be analysed for enrichment according to given gene sets and, conversely, interesting sets of genes can be visualized as flows. Finally, findings can be contrasted with the overall set of flows.

GRAPHICAL ABSTRACT

INTRODUCTION

High-throughput (sequencing) data are becoming increasingly popular. For instance, the number of publications listed in PubMed for scRNA-seq has been almost doubled from 2020 to 2021 and is yet to soar to a new record in 2023. With broader accessibility to scRNA-seq, and ever easier to perform wet-lab protocols, increasingly complex comparisons are performed, requiring the analysis of (un)ordered series data. At the same time, bulk RNA-seq has become a default tool for measuring gene expression. Large collections of public data sets are available, making the acquisition of large studies with many replicates relatively easy, but the analysis quite complex. For such complex data sets relying solely on explorative analyses, such as dimensionality reduction or differential gene expression analysis, is not enough anymore. With (un)ordered series data, comprising of different measurement points, conditional states, or replicates, the discovery and visualization of relevant genes and gene sets in an easily comprehensible way should be possible.

Yet, tools specifically designed to analyse series data exist. Tempora (15), for instance, can be used to derive trajectories through clusters and (discrete) time points by inferring time for the given sets of cells, which is subsequently used to calculate time-dependent pathways. This approach, however, requires ordered measurements in order to represent time. Process relevant genes can be defined as commonly regulated groups of genes (7). Many scRNA-seq analyses apply the concept of pseudo-time analysis and trajectory inference [3, 4]. With these analyses, similarly regulated genes, according to the inferred pseudo-time, can be identified, however, these methods cannot easily be applied to time-independent data. Kazer et al. propose the use of WGCNA (8) with subsequent post-processing to identify series-dependent modules. This method has already been applied to derive sets of similarly regulated genes over a series of (not necessarily ordered) measurements (7, 10). However, this method is commonly used only on a subset of all genes and is fine-tuned, as the parameter choices heavily influence which modules are detected. Choosing good parameters requires a deep understanding of the method, which hampers simple targeted analysis by a broad user base.

With our FlowSets framework we provide a new method for fuzzified data analysis of (un)ordered measurements of multiple measurement modalities. The FlowSets methods

allows a time- and order-independent analysis in a targeted way, focusing either on specific genes or expression patterns of interest. In addition, FlowSets can also be used on differential analysis results, thereby making a series of differential results comparable and comprehensible (double-differential view on the data). FlowSets makes use of fuzzy classes which encode the gene expression for a predefined set of cells (e.g., a cluster). This not only enables to represent the specific distribution of gene expression data, but also makes expression values easily interpretable as linguistic variables (16) by using fuzzy concepts characterizing gene expression as low or high. By representing gene expression via membership functions for each linguistic variable, thresholding is circumvented, as genes can be within multiple fuzzy concepts with specific fractions (memberships). This is particularly helpful for gene set over-representation analysis, as Boolean thresholding is avoided. The FlowSets framework, incorporating the above described concepts and providing visualizations, is available on GitHub (https://github.com/mjoppich/FlowSets).

METHODS

Nomenclature

In the description of the following methods and results, the state refers to single measurements (e.g. at a given time point). Likewise, the linguistic classes, are referred to expression classes or classes, only.

Collecting input for FlowSet

The input for FlowSets is a data frame in long format, which contains one line per measured feature in each state. In the case of absolute expression values, this can be the mean feature expression enhanced by the standard deviation and fraction of expressing members (for scRNA-seq data, for instance, mean gene expression within a cluster, standard deviation of gene expression within a cluster and percent expressing cells within cluster). For the double differential approach, differential comparisons along the series variable are performed between the members of the states to be compared. The recorded fold changes are then again stored in a long data frame and may be enhanced by additional information on the fold change distribution (e.g. percent expressing, standard deviation, etc.).

The code for deriving the input data from Seurat (4) objects is available from the project GitHub.

Transforming gene expression to fuzzy concepts

Fuzzy sets (16) are a useful discretization when an object can not precisely be mapped to a specific class, like gene expression is more like a distribution, spanning multiple classes, than a crisp value falling into just one class. The continuous values of features (e.g. gene expression) is divided into user-defined memberships (so called linguistic classes) (e.g. high or low expression) according to the feature's signal distribution and the membership function. The meaning of the linguistic class, its size and membership function is problem-specific. Often an equal distribution between minimum and maximum of the expression values is an initially good solution, but custom membership functions are also possible. In FlowSets, a gaussian, triangular and crisp membership function is provided, but the framework is also easily extensible.

For example, considering differential gene expression data, genes with a log fold change above 1 could be designated as up-regulated, and should also be discriminated from fold changes greater than 2, which represent strongly up-regulated genes. Genes with absolutely smaller fold changes or insignificant p-value could be regarded as neutral or no regulation. These thresholds can easily be adapted depending on the distribution of fold changes within the experiment.

Formally, for every feature f with signal x (which can also be a distribution of signals) the user defines a membership function $F(x)=m$. A Fuzzy Concept then is a set of all desired membership functions $FC(x)=\{F_1(x),\ldots,F_n(x)\}$. In brief, this fuzzy concept FC describes the membership of every feature to every membership: $FC(f)=\{m_1,\ldots,m_n\}$. This membership thus reflects the fraction (or: percentage) to which a feature is contained within a class. It thus must be ensured that for every feature f the sum $\sum_i m_i$ equals 1. The fuzzy concepts and memberships are calculated using the skfuzzy python package.

Calculating flows over all series states and membership classes

Flows are representatives of changes in the features (e.g. genes) over different states. In the crisp world, if a feature is changing class between state 1 to state 2, it is contained in the respective flow with membership 1. In fuzzy theorem, the paths are equally distributed for the classes and can be calculated by vector multiplication, pairwise combining the states in a linear fashion:

$$\begin{aligned} F(s_1,s_2) = & \quad \{m_1,\ldots,m_n\}_{s1} \times \{m_1,\ldots,m_n\}_{s2} \\ = & \quad \{m_{1,s1}\cdot m_{1,s2},\ldots,m_{n,s1}\cdot m_{n,s2}\} \end{aligned}$$

The complexity is increasing by the factor of the classes resulting in $|classes|^{|states|}$ flows, each having a membership for every feature. Due to this combinatorial problem, a gene's membership in each flow is not pre-calculated. For each flow, the calculation of its size is performed by summing up the individual genes' memberships to this flow. Given n genes, this is possible in $\mathcal{O}(n)$, while the calculation of a single gene's membership is in $\mathcal{O}(1)$. Nonetheless, for additional states, the number of flows increases exponentially. In order to tackle this problem, we apply parallel calculation of the individual flow memberships in order to keep wall clock time low. For a data set consisting of two classes and four states, the pre-processing (creation of the data frame) takes less than a minute, the calculation of all flows takes less than 10 seconds, and the pathway analysis for each flow takes about 2 minutes.

Identifying flows by specific patterns

A FlowFinder routine has been designed to find individual paths that follow the same pattern. This can be expressed either by the difference between two states, where the paths must stay in the same class, or change in a certain direction to analyze trends. The grammar used to describe these patterns

Table 1. FlowFinder grammar for selecting flows of interest. Between each state, the class change can be specified using the symbols below. At each state the user can also specify minimum or maximal class per state.

Symbol	Meaning
$?$	Any transition
$=$	Stays the same class
$\sim$	Remains approx. at class ($\pm$ 1)
$>$	Decreases class
$\gg$	Decreases class >1
$<$	Increases class
$\ll$	Increases class >1

is shown in Table 1. Furthermore, one can also select the minimum or maximum class in individual states, restricting flows both by their class at a given state and their relative path, in order to obtain desired flows. For example, in the marker gene search, one would search for genes that are only in one state in a high expressed class and are not expressed in all other states.

Analysis of resulting flows

The simplest way to assess the content of a single flow or a group of flows is to generate a ranked gene list of the membership of each feature to the specific flow. In order to achieve this, the membership F_f of a feature in a set of flows f must be calculated, which is defined as $F_f = \sum_{i=1}^{n \epsilon f} m_i$. By aggregating the memberships of features over flows it is not only possible to generate ranked gene lists of the top genes contained in a flow, but it also opens the door for further down-stream methods.

A more exploratory way of looking at all calculated flows is to assess the membership of gene set within a flow: fuzzy gene-set over-representation analysis. Similar to the crisp world, gene-set over-representation analysis here is seen as an over-representation of a specific set of features within the data. Thus, for each pathway, we sum up the membership of the pathway's genes within a specific flow (or a group of flows). We then calculate the pathway coverage, the fraction of the summed memberships and the number of measured pathway genes, which has a zero-inflated normal distribution. Since it was noted, that smaller pathways are more prone to higher pathway coverages, all pathways were assigned to bins depending on the number of contained genes. By default, the bin boundaries are 2, 10, 50 and 100. Within these bins, the pathway coverages are z-scaled, and for the positive z-values we subsequently derive p-values from the survival function of the underlying normal distribution. The resulting p-values are corrected for multiple testing (Benjamini-Hochberg) and can be accessed per flow or set of flows. As input for gene sets any gmt-formatted definition can be used, such as Reactome pathways (6), Gene Ontology (1) or MSigDB pathways (14). In general, the FlowSets framework accepts any gene-set in gmt format, and custom gene sets can be added from python directly.

WGCNA method

We employ the same WGCNA(8) method for comparison that was also employed by Pekayvaz et al. (10), and thus follow the WGCNA-based method initially described by Kazer et al. (7). The analysis is performed on 10 PCs, with a softPower set to 10 and takes 100 genes per principal component for analysis.

Simulated data

We simulate scRNA-seq counts using SPARSim (3) and the 10X PBMC preset. SPARSim was chosen because it ranks among the best general purpose simulation tools, particularly allowing the simulation of multiple conditions as well as a user defined amount of differential expression (5). We simulate 4 different conditions/states with each 5419 cells, calling them "wildtype" and "knockout" 1 to 3. The default sparsity of 95% is undesired, hence the library size was taken tenfold. Over the four different conditions, three different patterns of (differential) gene expression should be simulated (Table 4). For each pattern, ten non-overlapping GO (1) gene sets were chosen randomly, and the there-in contained genes were simulated according to the three pre-defined trends over the four states. The different trends were applied to each gene from the selected random gene sets. Hence, some selected genes show very small to no gene expression, if their original expression was very low, too. Up-regulating such genes in the trends then does not necessarily create a detectable signal, we limited the 'true' set of altered genes so having an intensity of at least 0.2 in one of the conditions.

Use-case data

The data for the use-case was obtained from a SARS-CoV-2 related study in which pneumonic (symptomatic) and non-pneumonic (non-symptomatic) patients were characterized using scRNA-seq analysis (10). In this use-case we are only interested in the pneumonic or non-pneumonic ones and hence ignore the control group for the analysis in this manuscript.

Availability

The source code of FlowSets and the conducted analysis are available online at https://github.com/mjoppich/FlowSets.

RESULTS AND DISCUSSION

The FlowSets method allows fuzzy-logic-based discrete-series analysis of (differential) expression data (e.g. bulk or single cell- RNA-seq or proteomics). Its full potential can be exploited on scRNA-seq datasets, where gene expression is to be compared across a discrete series of events, because the fuzzification allows the representation of zero-inflated gene expression distributions. We thus showcase the FlowSets method on two scRNA-seq datasets. First, using simulated data, we evaluate our method and compare it with a WGCNA-based method. Finally, we apply our method to a real-word scRNA-seq dataset.

Evaluation on Simulated data

On the simulated data we want to find out how well our FlowSets and the WGCNA methods can (a) identify the simulated patterns, and (b) how well the regulated genes and gene sets can be identified. FlowSets was called in its three versions of fuzzy concepts (crisp, gaussian and triangular). For all fuzzy concepts the same centers and widths were

Table 2. Results of the simulated data shown on gene-level. N_genes is the number of altered genes. Afterwards are shown the number of relevant results and found matches of the different parameters and tools.

	N_genes	N_trues	N_crisp	N_crisp_found	N_gaussian	N_gaussian_found	N_triangular	N_triangular_found	N_WGCNA	N_WGCNA_found
Pattern 1	39	23	10	10	23	19	23	18	6	6
Pattern 2	47	30	30	30	30	27	30	26	17	15
Pattern 3	34	16	0	0	16	12	16	4	0	0

Table 3. Results of the simulated data shown at pathway-level

	N_pathways	N_crisp	N_crisp_found	N_gaussian	N_gaussian_found	N_triangular	N_triangular_found	N_WGCNA	N_WGCNA_found
Pattern 1	10	0	0	68	7	69	7	NA	NA
Pattern 2	10	6	3	165	10	172	10	NA	NA
Pattern 3	10	0	0	31	10	72	10	NA	NA

used, and for the crisp version, only the mean expression of all expressing cells was used to determine the gene's class. The size of the fuzzy sets was chosen such that all values are well representable, there are not too many nor too few classes and that a fold-change of 2 could be identified. Here, for each method seven classes, reaching from log(UMI) count 0.1 to 2.5, were created. In order to perform the analysis with FlowSets, the target flows were identified using the `flow_finder`, reflecting the simulated fold changes and the anticipated change of at least two classes (Table 4).

For the three patterns, we use the `flow_finder` to find the respective target flows. We then check how many genes are contained within these flows, and chose the genes with the highest membership and the same amount as the altered set. The intersection of the such identified genes and the set of altered genes in the simulated data set then is used to assess the performance of both the FlowSets and WGCNA method (Table 2 & Figure 1). Due to the decision to use realistically simulated data and to regulate random gene sets, it is possible that not all genes of a regulated gene set are expressed. As a result of this, it is possible that not all genes are identifiable as regulated. We therefore curated the list of changed genes such that only genes, for which a change was quantifiable, are contained. Using recall, precision, the f_1-score and the jaccard-index we assess the performance of the different methods.

First we want to compare the performance of the three different fuzzification concepts use-able from within FlowSets, which have all different properties regarding the translation to fuzzy values. The triangular function is more balanced and usually has larger overlaps with its neighbours. The gaussian function has only small border areas with shared memberships. Finally, the crisp function has no shared memberships, assigning a single class to each feature. It can thus be assumed that the triangular fuzzy concept is the most fuzzy or distributed concept, followed by the gaussian version and the crisp version, which shows no real fuzzification any more, because a feature is fully assigned to a single flow and therefore contained completely in this specific flow.

In the evaluation this can also be seen. For the first pattern, the crisp version only identifies 10 genes within the target flows, of which 10 are among the regulated ones. In contrast, the gaussian and triangular versions are set to the number of altered genes, which they can almost completely identify. Similar observations can also be made for the other patterns. In conclusion, gaussian and triangular mode retrieve approximately the same amount of genes with a good performance and seem overall balanced. It can be noted that the crisp version retrieves the least regulated genes, while highly varying in the examples, emphasising the use of fuzzy concepts. While crisp cut-offs show a clear classification of genes to a flow and therefore a more simplistic downstream analysis, fuzzy memberships show a higher performance in this evaluation. In contrast, with WGCNA it is not possible to retrieve as many genes. Notably, for pattern 3 no genes are found by WGCNA, performing similar to the crisp version of FlowSets. While it can not be outruled that using different parameters for WGCNA might retrieve more genes and pathways, WGCNA remains an explorative method which focuses on identifying the largest differences. Therefore, it does not allow a detailed look into specific patterns, as it is possible with FlowSets. This probably explains also, why for pattern 3 no modules and genes are identified.

Going one step further, with FlowSets it is possible to examine whether the changed pathways can be inferred from the target flows (Table 3). Again, checking the different fuzzification concepts, the prior observations can be confirmed. Particularly the difference between the crisp and the fuzzified version are emphasized. Since WGCNA does not allow for gene set enrichment directly, we applied the hypergeometric test on the identified genes and the Gene Ontology gene sets. Given the fewer identified genes, it was not possible to infer many regulated gene sets. This highlights the applicability of FlowSets for describing gene expression in terms of fuzzy concepts, from which regulated genes can be described in terms of patterns to accurately identify regulated gene sets within these patterns.

The difference between the triangular and gaussian membership function is no longer so strongly reflected in the results. Although the ranking of the genes is different, the genes found in the top 50 are almost identical. They also differ slightly in the identified enriched pathways, but with negligible differences. The comparison with the crisp analysis shows little variations, although the genes are now completely included, e.g. with a membership of 1, most of the modified genes are still in the list. The pathway analysis (which was mainly optimized for the membership assignment) gives only very few significant results, but these then agree with the found pathways of the other fuzzy concepts.

Here, the power of the memberships for each feature are clearly visible as they are used as weight of importance from

the other fuzzy concepts, giving them a leg-up with a ranked list.

Use-case

With the use-case we want to highlight the applicability of FlowSets to biomedical data. Using an existing real world data set we want to reproduce the therein presented results.

The use-case data set consists of human peripheral blood mononuclear cells (PBMCs) of patients with pneumonic (symptomatic) and non-pneumonic (non-symptomatic) COVID-19, analysed at three distinct timepoints (10). This data set is thus well suitable for a FlowSets analysis, because not only the non-homogeneous nature of scRNA-seq expression among cells can be exploited for fuzzification, but also because it allows for 1) an analysis of gene expression data along the distinct timepoints for each disease state and 2) a double-differential comparison along the time point series and between the two disease states. In this study we focus, like in the original publication, on monocytes. Hence this use-case operates only on (differential) gene expression from monocyte clusters.

The fuzzy nature of FlowSets applies exceptionally well to scRNA-seq data because it suits the zero-inflated distribution of the single-cell count data (cells expressing and cells not expressing a gene within one cluster). The fuzzy implementation represents a distribution of gene expression values, and thus not only allows us to see changes in the intensity of gene expression, but also in the amount of expressing cells, opening a completely new view of the complex nature of gene expression in scRNA-seq data.

FlowSets provides instructions for creating gene expression output directly from Seurat (4) objects (see GitHub repository), resulting in a file with one mean expression value (log-space) per gene, and corresponding standard deviation and percent expressing cells. Given these additional information about the distribution of the gene expression values, the fuzzification can be performed by sampling from the such defined distribution (here: 1000 samples). These are then fuzzified and averaged for each sample, and weighted by the fraction of cells expressing the particular gene. This way, scRNA-seq gene expression is accurately represented in a fuzzified form. This output is read in by FlowSets, and for each gene, at each state, the average expression is transformed into a membership for each designated class of the fuzzification. In this analysis, we choose six distinct classes ("NO", "LOW", "LOWMED", "MED", "MEDHIGH" and "HIGH") as triangular membership functions with their peak/center at 0.4, 0.8, 1.2, 1.6, 2 and 2.4.

When analyzing the data, we first want to get an overview of gene expression patterns over the full data set (Figure 2a). In the general FlowSets overview, it can be seen that a majority of genes are not expressed and do not change over the series (NO class). Only a few genes are in the medium or high expression range, which is depicted by only thin flows/lines. Nonetheless, we are interested in specific expression patterns which are indicative of the studied SARS-CoV-2 infection. These are genes, which are initially highly expressed in order to tackle the infection, and then return to a lower class of expression (Figure 2b). Flows following such a pattern can be selected using the `flow_finder` routine. The FlowSets framework provides means to analyse these selected (or all) flows also on a pathway or gene set basis. For each flow, it is checked whether a specific gene set (e.g. from a collection like Reactome (6) or Gene Ontology (1)) is over-represented. Here, we performed such analysis on the target flows using Gene Ontology biological process gene sets and the two ISG gene signatures from the original publication (10). The gene set over-representation results were filtered for significance (here: adjusted p-value <0.05) and gene set sizes (here: size >5 genes) and visualized using inbuilt functionality (Figure 3). The filtering of small gene sets is performed because very small pathways are not deemed of interest. From the original publication it is known that the interferon stimulating genes play a major role in early stages of SARS-CoV-2 infection, and thus it is re-assuring that these gene sets are also found significantly enriched in the target flows. Moreover, we can see that the short ISG gene set is more abundant in the non-syptomatic case (pathway coverage 4.473 vs 3.580). It is interesting to see several metabolic processes listed within the top 20 over-represented gene sets in the non-symptomatic data. This suggests, that symptomatic patients lack a mitochondrial up-regulation, which is in-line with existing literature (13).

With the FlowSets framework it is also possible to have a close look at the memberships of specific genes over all states and classes (Figure 4a). For instance, the genes IFI27, IFI44L, IFI6, IFITM3, ISG15 and LY6E show a high expression level at TP1 and then a decreasing expression at states TP2 and TP3. With this functionality it is possible to check the behaviour of specific genes. But FlowSets also has the capability to look into specific flows or sets of flows. For the selected target flows it is possible to look at which genes have the highest memberships within these flows (Figure 4b). Unsurprisingly, many ISG-genes are among the top genes representing the target flows. On the other hand, it becomes also apparent that the membership within the target flows drops fast to below 25%, indicating that these genes have the majority of their membership in other flows.

Next, we are interested in identifying key differences between symptomatic and non-symptomatic patients. Again, we concentrate on the monocytes within the data set, because these are known to be key indicators for disease outcome (9, 12). For this, for each state within the series, a differential comparison between both disease-state groups was performed using Seurat's FindMarkers function. The thereby returned single differential expression results were concatenated to a discrete series, and the log2 fold changes are used as input for FlowSets. The linguistic classes are defined such that a gene is clearly up-regulated in one group (`SYMPT` or `ASYMPT`), most abundant in one group (`sympt` or `asympt`) or `nodiff` if either the fold change is too small, or there was no significant fold change. The widths and centers of the fuzzy concepts are automatically derived by FlowSets. Again, we first checked the overall patterns (Figure 5). A larger set of genes, which is more prevalent `sympt` at TP1 can be noted, while no such a flow is visible in `asympt`. This warrants a closer look at flows which have a disease-associated pattern (e.g. changing from one extreme at TP1 to the other, or no regulation, at TP3; Figure 7a,b). For both target flow sets, gene set enrichment on the combined flow was performed. Again, results were

filtered for significance, and gene set size and finally sorted by adjusted p-value (Figure 6).

In the symptomatic cases (Figure 7a) it is interesting to see an up-regulation of apoptotic process as well as a viral entry into host cell. Albeit not significant anymore, among the top 20 gene sets in the symptomatic case many apoptose- and cell death related gene sets are identified. Again, this is expected, as programmed cell death is a hallmark of severe COVID-19 infection (2). Likewise, there are many gene sets associated with T cell activation found for genes up-regulated in the non-symptomatic case (Figure 7b). It is also possible to confirm the finding, that ISG-related gene signatures are over-represented in non-symptomatic patients. As a consequence of this, anti-viral processes like *clearance of foreign intracellular DNA*, or processes inhibiting cell growth like *negative regulation of meiotic cell cycle* are found in the non-symptomatic case. A high level of cell cycle activity has been associated with severe COVID-19 (11).

These results show that FlowSets is able to easily extract relevant processes and activities from gene expression or differential gene expression data. It was possible to recapitulate the results from the original publication, and to identify key processes like cell cycle activity only uncovered recently.

CONCLUSION

Here, we present the FlowSets framework, which allows flow-based analysis of (discrete) series data, on both expression and differential values using fuzzy concepts. While we demonstrate the use of our framework only on scRNA-seq data, the framework can be applied to a whole range of data, from RNA-seq, over spatial technologies to proteomics. The fuzzy representation of the expression variables is of most help when the expression can not be quantified exactly, e.g. in scRNA-seq or spatial proteomics/transcriptomics data, where zero-inflated gene expression distributions are prevalent. With the help of discrete expression classes, the complexity in interpreting gene expression values is reduced, also by leveraging expression to linguistic classes.

We demonstrate that FlowSets can identify more regulated genes in the simulated data than WGCNA can. Our data also show that crisp gene assignments are able to reproduce most of the simulated regulations, but fail to identify the regulated gene sets. These can successfully be identified using fuzzified memberships, independently of using triangular or gaussian membership functions. Opening the door to all kinds of weighted or ranked downstream analyses.

With the performed analysis we could reproduce known facts about COVID-19, and reproduce findings from the selected use-case publication. Most importantly, it was easy to query for disease-related processes depending on the expression pattern within the discrete series. Given a new data set and a new scientific question, the FlowSets framework can be of great help to discover relevant flows, check the expression patterns of specific genes and identify relevant gene sets, and by these active processes within the matter of study. With FlowSets it is easy to get a general overview of relevant processes (e.g. by pathway enrichment) or to look into pathways following a specific pattern through the series states. Also, a targeted analysis is easily possible by highlighting genes or gene-sets within the flows between states and linguistic classes respectively. In contrast to explorative methods, with FlowSets it is possible to have both a gene centric view on the data, as well as a pattern centric view. The analysis with FlowSets can be performed for any target gene or pattern, independent of it being identified as specifically interesting by the method.

Table 4. Simulated Expression classes in the simulated scRNA-seq data. The table shows the factors the intensities of the chosen pathways genes were multiplied.

	Expression Class				
Pattern	State 1	State 2	State 3	State 4	flow_finder pattern
1	1	1	4	4	$[=,\gg,=]$
2	1	8	8	1	$[\gg,=,\ll]$
3	2	2	1	1	$[=,\ll,=]$

In summary, with the FlowSets framework an easy and comprehensible analysis of complex series data is possible. FlowSets can be used for any expression data, and even on differential expression analysis results. The user can select expression patterns of interest, and, therein, FlowSets is able to identify even subtle differences in expression. At all stages of an analysis, the interpretation of the data is supported by meaningful visualizations. As such, FlowSets is a versatile tool for analyses of many different modalities, even beyond the gene expression community. The FlowSets framework is available on GitHub (https://github.com/mjoppich/FlowSets).

AUTHOR CONTRIBUTION

Felix Offensperger: Conceptualization, Methodology, Software, Visualization, Investigation, Writing - Original Draft; Markus Joppich: Conceptualization, Methodology, Software, Visualization, Investigation, Writing - Original Draft, Supervision; Ralf Zimmer: Conceptualization, Writing - Review & Editing, Supervision

REFERENCES

1. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M., Sherlock, G.: Gene Ontology: tool for the unification of biology. Nat Genet **25**(1), 25–29 (May 2000). https://doi.org/10.1038/75556, http://www.nature.com/doifinder/10.1038/75556, publisher: Nature Publishing Group
2. Bader, S.M., Cooney, J.P., Pellegrini, M., Doerflinger, M.: Programmed cell death: the pathways to severe COVID-19? Biochemical Journal **479**(5), 609–628 (Mar 2022). https://doi.org/10.1042/BCJ20210602, https://doi.org/10.1042/BCJ20210602
3. Baruzzo, G., Patuzzi, I., Di Camillo, B.: SPARSim single cell: a count data simulator for scRNA-seq data. Bioinformatics **36**(5), 1468–1475 (Mar 2020). https://doi.org/10.1093/bioinformatics/btz752, https://doi.org/10.1093/bioinformatics/btz752
4. Butler, A., Hoffman, P., Smibert, P., Papalexi, E., Satija, R.: Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nature Biotechnology **36**(5), 411–420 (May 2018). https://doi.org/10.1038/nbt.4096, http://www.nature.com/articles/nbt.4096, publisher: Nature Publishing Group
5. Cao, Y., Yang, P., Yang, J.Y.H.: A benchmark study of simulation methods for single-cell RNA sequencing data. Nature Communications

12(1), 6911 (Nov 2021). https://doi.org/10.1038/s41467-021-27130-w, https://www.nature.com/articles/s41467-021-27130-w, number: 1 Publisher: Nature Publishing Group

6. Jassal, B., Matthews, L., Viteri, G., Gong, C., Lorente, P., Fabregat, A., Sidiropoulos, K., Cook, J., Gillespie, M., Haw, R., Loney, F., May, B., Milacic, M., Rothfels, K., Sevilla, C., Shamovsky, V., Shorser, S., Varusai, T., Weiser, J., Wu, G., Stein, L., Hermjakob, H., D'Eustachio, P.: The reactome pathway knowledgebase. Nucleic Acids Research **48**(D1), D498–D503 (Jan 2020). https://doi.org/10.1093/nar/gkz1031, https://orcid.org, publisher: Oxford University Press
7. Kazer, S.W., Aicher, T.P., Muema, D.M., Carroll, S.L., Ordovas-Montanes, J., Miao, V.N., Tu, A.A., Ziegler, C.G.K., Nyquist, S.K., Wong, E.B., Ismail, N., Dong, M., Moodley, A., Berger, B., Love, J.C., Dong, K.L., Leslie, A., Ndhlovu, Z.M., Ndung'u, T., Walker, B.D., Shalek, A.K.: Integrated single-cell analysis of multicellular immune dynamics during hyperacute HIV-1 infection. Nature Medicine **26**(4), 511–518 (Apr 2020). https://doi.org/10.1038/s41591-020-0799-2, https://www.nature.com/articles/s41591-020-0799-2, number: 4 Publisher: Nature Publishing Group
8. Langfelder, P., Horvath, S.: WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics **9**(1), 559 (Dec 2008). https://doi.org/10.1186/1471-2105-9-559, https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-9-559
9. Merad, M., Martin, J.C.: Pathological inflammation in patients with COVID-19: a key role for monocytes and macrophages. Nature Reviews Immunology **20**(6), 355–362 (Jun 2020). https://doi.org/10.1038/s41577-020-0331-4, https://www.nature.com/articles/s41577-020-0331-4, number: 6 Publisher: Nature Publishing Group
10. Pekayvaz, K., Leunig, A., Kaiser, R., Joppich, M., Brambs, S., Janjic, A., Popp, O., Nixdorf, D., Fumagalli, V., Schmidt, N., Polewka, V., Anjum, A., Knottenberg, V., Eivers, L., Wange, L.E., Gold, C., Kirchner, M., Muenchhoff, M., Hellmuth, J.C., Scherer, C., Rubio-Acero, R., Eser, T., Deák, F., Puchinger, K., Kuhl, N., Linder, A., Saar, K., Tomas, L., Schulz, C., Wieser, A., Enard, W., Kroidl, I., Geldmacher, C., von Bergwelt-Baildon, M., Keppler, O.T., Munschauer, M., Iannacone, M., Zimmer, R., Mertins, P., Hubner, N., Hoelscher, M., Massberg, S., Stark, K., Nicolai, L.: Protective immune trajectories in early viral containment of non-pneumonic SARS-CoV-2 infection. Nature communications **13**(1), 1018 (Feb 2022). https://doi.org/10.1038/s41467-022-28508-0, https://www.nature.com/articles/s41467-022-28508-0, publisher: Nature Publishing Group
11. Prado, C.A.d.S., Fonseca, D.L.M., Singh, Y., Filgueiras, I.S., Baiocchi, G.C., Plaça, D.R., Marques, A.H.C., Dantas-Komatsu, R.C.S., Usuda, J.N., Freire, P.P., Salgado, R.C., Napoleao, S.M.d.S., Ramos, R.N., Rocha, V., Zhou, G., Catar, R., Moll, G., Camara, N.O.S., de Miranda, G.C., Calich, V.L.G., Giil, L.M., Mishra, N., Tran, F., Luchessi, A.D., Nakaya, H.I., Ochs, H.D., Jurisica, I., Schimke, L.F., Cabral-Marques, O.: Integrative systems immunology uncovers molecular networks of the cell cycle that stratify COVID-19 severity. Journal of Medical Virology **95**(2), e28450 (2023). https://doi.org/10.1002/jmv.28450, https://onlinelibrary.wiley.com/doi/abs/10.1002/jmv.28450, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jmv.28450
12. Schulte-Schrepping, J., Reusch, N., Paclik, D., Baßler, K., Schlickeiser, S., Zhang, B., Krämer, B., Krammer, T., Brumhard, S., Bonaguro, L., De Domenico, E., Wendisch, D., Grasshoff, M., Kapellos, T.S., Beckstette, M., Pecht, T., Saglam, A., Dietrich, O., Mei, H.E., Schulz, A.R., Conrad, C., Kunkel, D., Vafadarnejad, E., Xu, C.J., Horne, A., Herbert, M., Drews, A., Thibeault, C., Pfeiffer, M., Hippenstiel, S., Hocke, A., Müller-Redetzky, H., Heim, K.M., Machleidt, F., Uhrig, A., Bosquillon de Jarcy, L., Jürgens, L., Stegemann, M., Glösenkamp, C.R., Volk, H.D., Goffinet, C., Landthaler, M., Wyler, E., Georg, P., Schneider, M., Dang-Heine, C., Neuwinger, N., Kappert, K., Tauber, R., Corman, V., Raabe, J., Kaiser, K.M., Vinh, M.T., Rieke, G., Meisel, C., Ulas, T., Becker, M., Geffers, R., Witzenrath, M., Drosten, C., Suttorp, N., von Kalle, C., Kurth, F., Händler, K., Schultze, J.L., Aschenbrenner, A.C., Li, Y., Nattermann, J., Sawitzki, B., Saliba, A.E., Sander, L.E., Angelov, A., Bals, R., Bartholomäus, A., Becker, A., Bezdan, D., Bonifacio, E., Bork, P., Clavel, T., Colome-Tatche, M., Diefenbach, A., Dilthey, A., Fischer, N., Förstner, K., Frick, J.S., Gagneur, J., Goesmann, A., Hain, T., Hummel, M., Janssen, S., Kalinowski, J., Kallies, R., Kehr, B., Keller, A., Kim-Hellmuth, S., Klein, C., Kohlbacher, O., Korbel, J.O., Kurth, I., Landthaler, M., Li, Y., Ludwig, K., Makarewicz, O., Marz, M., McHardy, A., Mertes, C., Nöthen, M., Nürnberg, P., Ohler, U., Ossowski, S., Overmann, J., Peter, S., Pfeffer, K., Poetsch, A.R., Pühler, A., Rajewsky, N., Ralser, M., Rieß, O., Ripke, S., Nunes da Rocha, U., Rosenstiel, P., Saliba, A.E., Sander, L.E., Sawitzki, B., Schiffer, P., Schulte, E.C., Schultze, J.L., Sczyrba, A., Stegle, O., Stoye, J., Theis, F., Vehreschild, J., Vogel, J., von Kleist, M., Walker, A., Walter, J., Wieczorek, D., Ziebuhr, J.: Severe COVID-19 Is Marked by a Dysregulated Myeloid Cell Compartment. Cell **182**(6), 1419–1440.e23 (Sep 2020). https://doi.org/10.1016/j.cell.2020.08.001, https://linkinghub.elsevier.com/retrieve/pii/S0092867420309922
13. Srinivasan, K., Pandey, A.K., Livingston, A., Venkatesh, S.: Roles of host mitochondria in the development of COVID-19 pathology: Could mitochondria be a potential therapeutic target? Molecular Biomedicine **2**(1), 38 (Nov 2021). https://doi.org/10.1186/s43556-021-00060-1, https://doi.org/10.1186/s43556-021-00060-1
14. Subramanian, A., Tamayo, P., Mootha, V.K., Mukherjee, S., Ebert, B.L., Gillette, M.A., Paulovich, A., Pomeroy, S.L., Golub, T.R., Lander, E.S., Mesirov, J.P.: Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. Proceedings of the National Academy of Sciences **102**(43), 15545–15550 (2005). https://doi.org/10.1073/pnas.0506580102, https://www.pnas.org/doi/abs/10.1073/pnas.0506580102
15. Tran, T.N., Bader, G.D.: Tempora: Cell trajectory inference using time-series single-cell RNA sequencing data. PLOS Computational Biology **16**(9), e1008205 (Sep 2020). https://doi.org/10.1371/JOURNAL.PCBI.1008205, https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1008205, publisher: Public Library of Science
16. Zadeh, L.A.: Fuzzy sets. Information and Control **8**(3), 338–353 (Jun 1965). https://doi.org/10.1016/S0019-9958(65)90241-X, publisher: Academic Press

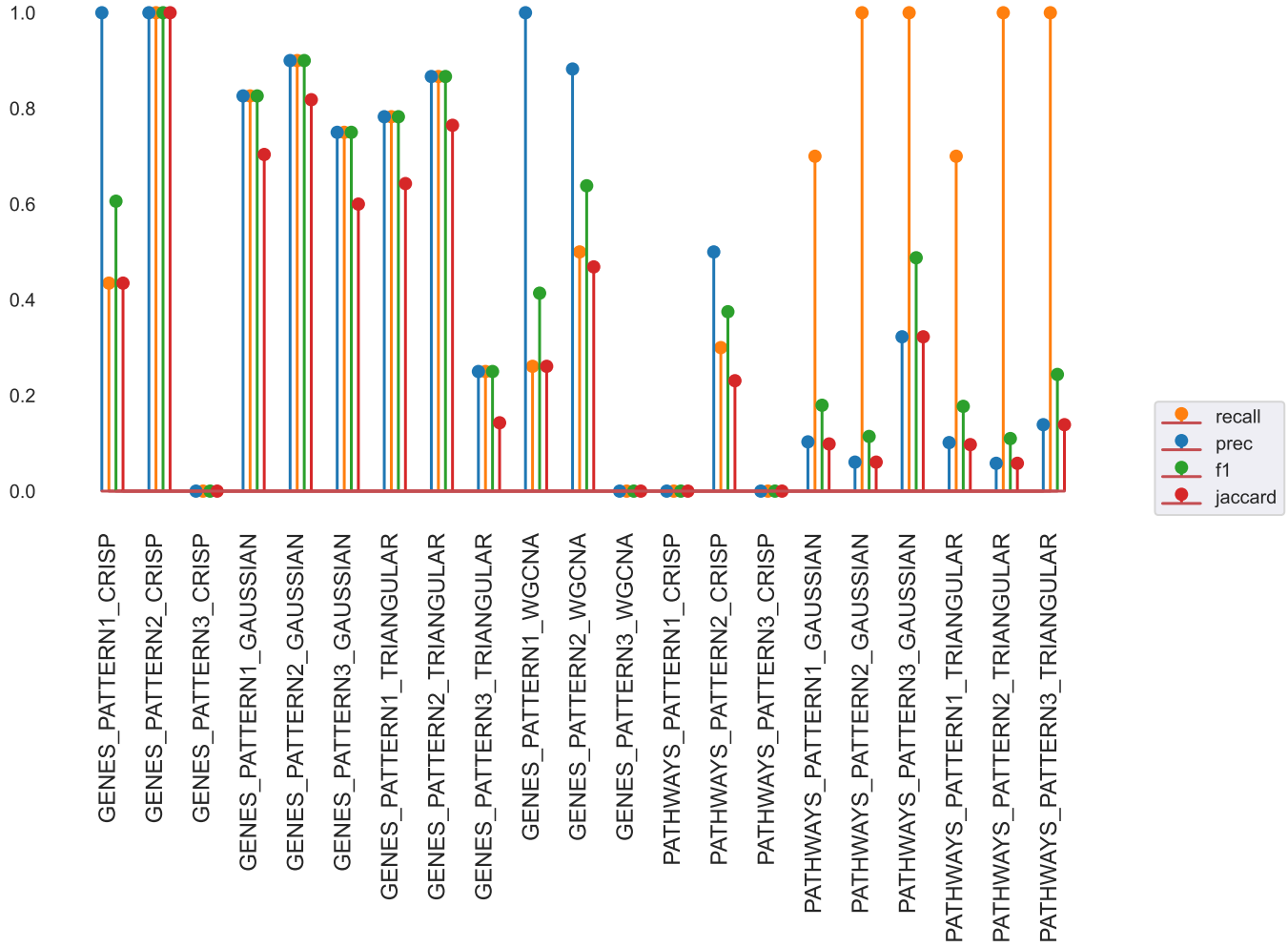
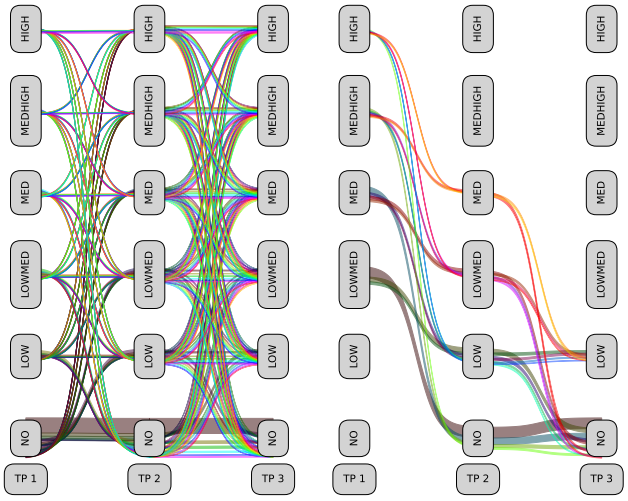
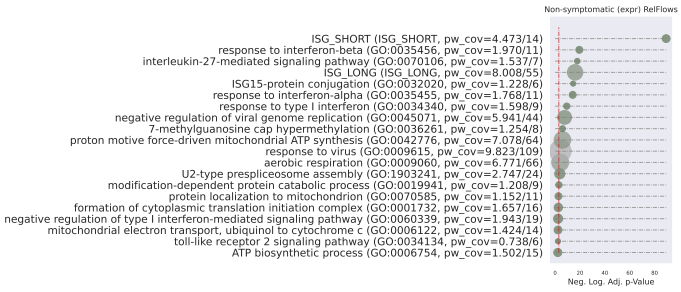


Figure 1. Evaluation of performance of FlowSets on the benchmark on gene-level (left) and pathway-level (right). On gene-level it is possible to compare the results to WGCNA directly. For each evaluation we compare recall, precision, the f_1-score and the jaccard-index. The fuzzy versions of FlowSets show continuously a strong balanced performance on gene-level and an enormous strength in recall on pathway-level.

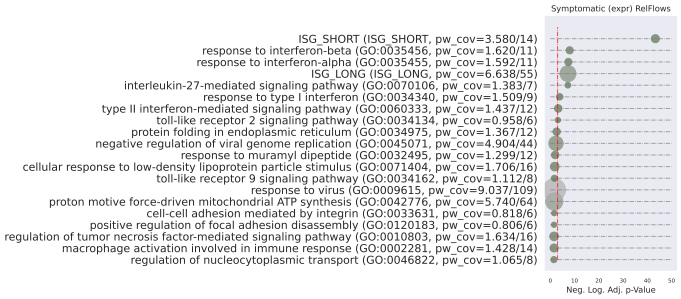


(a) FlowSets Overview (b) Target Flows

Figure 2. (a) FlowSets plot for the (non-symptomatic) scRNA-seq data set showing the course of 25.767 genes. (b) The subset of the FlowSets plot showing only such flows which indicate a decreasing gene activity over the course of the distinct time points 1-3. Such flows can easily be found using the `flow_finder` routine.

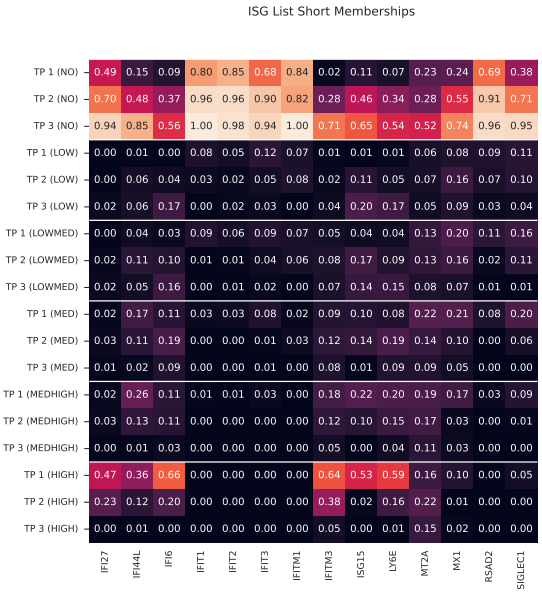


(a) Target Flows Enrichment (non-symptomatic)

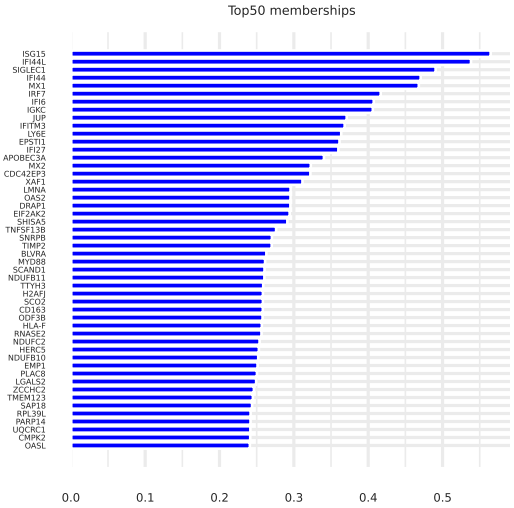


(b) Target Flows Enrichment (Symptomatic)

Figure 3. Result of the gene set over-representation analysis performed on the combined target flows for monocytes from (a) non-symptomatic patients and (b) symptomatic patients. The gene sets were filtered for size (>5 genes), significance (adjusted p-value <0.05) and sorted by adjusted p-value. It can be seen that the ISG_SHORT gene set is less enriched (pw_cov smaller) in the symptomatic case, hinting at a lower ISG activation in those patients.



(a) FlowSets Memberships for ISG_short genes in non-symptomatic sample



(b) Gene memberships for target flows in non-symptomatic sample

Figure 4. (a) FlowSets membership plot for the ISG_short gene sets in non-symptomatic monocytes, (b) Gene membership in selected target flows of the non-symptomatic monocytes

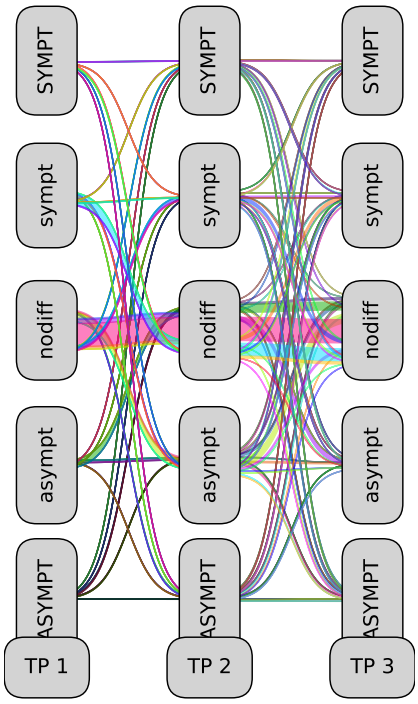
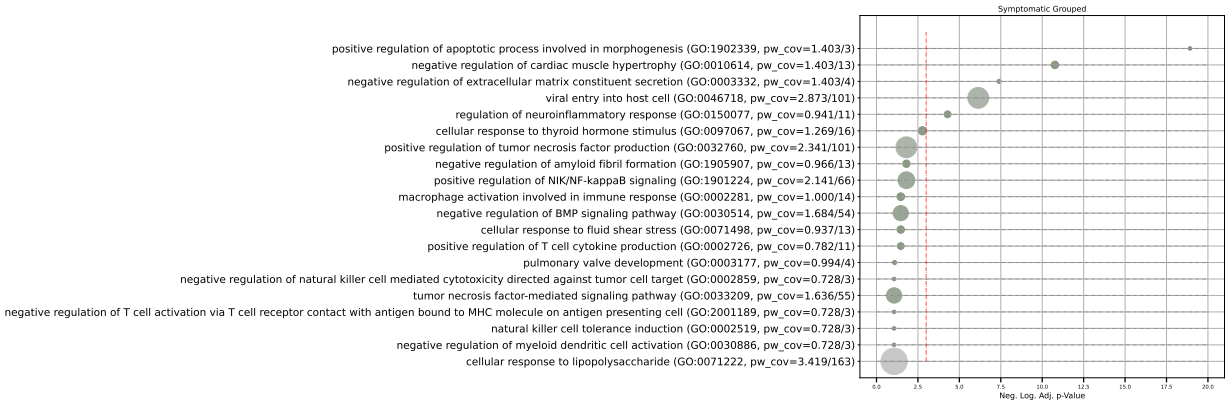
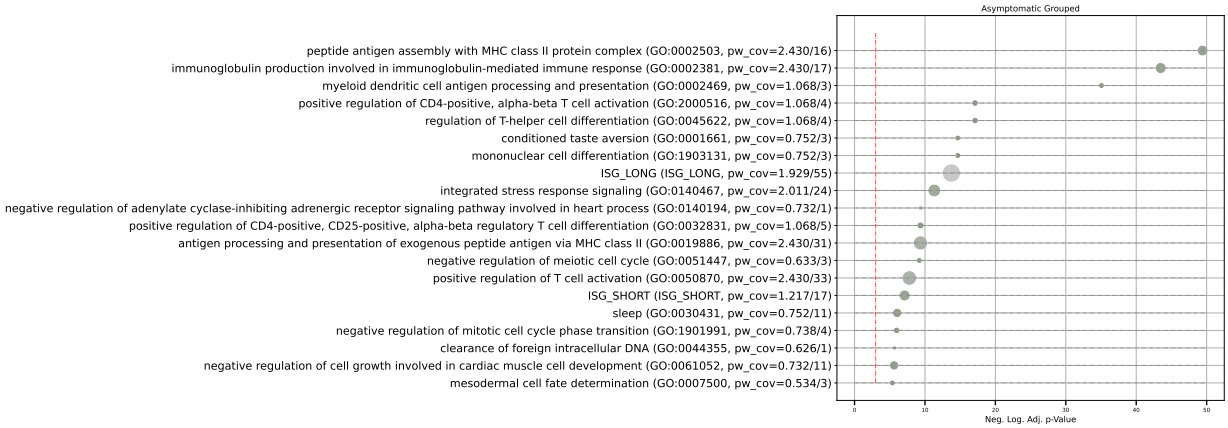


Figure 5. FlowSets plot for the scRNA-seq data set visualizing the differential gene expression analysis results between the symptomatic and non-symptomatic group. A large set of up-regulated genes can be noted in the sympt class at TP1, while such a flow is not visible in the asympt case.

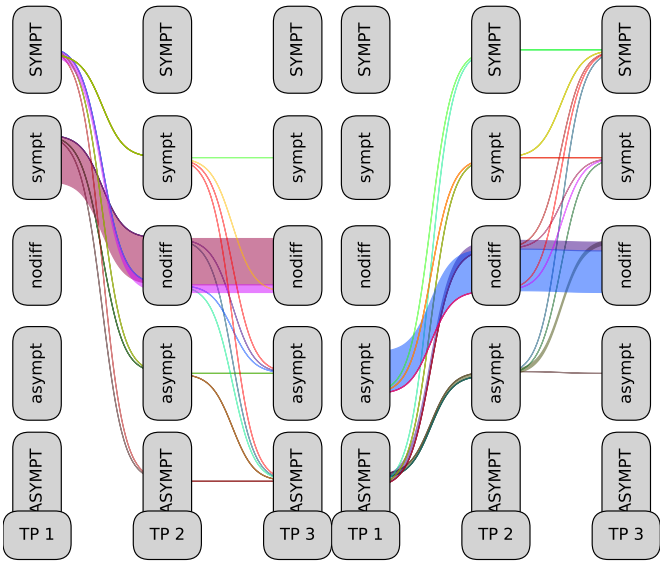


(a) Flow enrichment for all symptomatic target flows



(b) Flow enrichment for all non-symptomatic target flows

Figure 6. FlowSets enrichment analysis on the combined target flows for the (a) symptomatic and (b) non-symptomatic case on Gene Ontology (biological process) gene sets.



(a) Target Flows for Symptomatic case(b) Target Flows for Non-symptomatic case

Figure 7. Target flows indicative of (a) symptomatic or (b) non-symptomatic disease progression. The flows are selected such that the expression classes decrease over all states of the series, with a minimum expression class at the `sympt` level, or `asympt`, respectively. Flows which indicate no change throughout disease progression are left out.