Eclipse: Alignment of Two or More Nontargeted LC-MS Metabolomics Datasets using Directed Subalignments

Daniel S. Hitchcock, Jesse N. Krejci, Courtney A. Dennis, Sarah T. Jeanfavre, Julian R. Avila-Pacheco, Clary B. Clish

ABSTRACT

Nontargeted LC-MS metabolomics datasets contain a wealth of information but present many challenges during analysis and processing. Often, more than two independently processed datasets must be aligned, but no software natively allows for this. To align two or more processed nontargeted datasets, we have created an open-source Python package called Eclipse. Eclipse uses a novel subalignment approach to model the whole alignment and has built-in graph aggregation options for reporting tabular data. Each subalignment independently transforms and scales feature descriptors (retention time, mass-to-charge ratio, average feature intensity) and scores feature matches in a data driven approach. Subalignments run independently, thus could be run in parallel or over time to construct large networks. Eclipse is fast (two datasets in 7 seconds, nine datasets in 39 seconds), workflow-agnostic, and customizable even for use outside of LC-MS datasets should a need arise. Eclipse is open source and available as part of our broader processing tools BMXP (https://github.com/broadinstitute/bmxp). Eclipse can be installed via the pip command “pip install bmxp”.

INTRODUCTION

Nontargeted liquid-chromatography tandem mass-spectrometry (LC-MS) is a powerful methodology for inspecting the metabolic state of a biological specimen (Clish 2015). In a routine processing workflow, feature extraction software converts raw instrument files to tabular datasets, and features are identified, integrated, and labeled by their chromatographic retention time (RT) and mass-to-charge ratio (*m/z*) (Smith *et al.* 2006; Pluskal *et al.* 2010). Thousands of features are typically detected in nontargeted datasets and though many of these features may represent redundant ions or chemical contaminants (Mahieu and Patti 2017), many of the signals arise from yet unannotated compounds of biological significance (Chen *et al.* 2022; Tahir *et al.* 2022; Vatanen *et al.* 2022). A challenge in analyzing

nontargeted metabolomics is the concatenation of unknown features among datasets that have been acquired and processed separately, a process that can be referred to as alignment (Smith, Ventura and Prince 2015). Though software packages and algorithms have been created to accomplish post-extraction feature alignment for a variety of purposes (Brunius, Shi and Landberg 2016; Koch *et al.* 2016; Mak *et al.* 2020; Habra *et al.* 2021; Climaco Pinto *et al.* 2022), our workflow requires functionality not offered by these tools. Specifically, our software must: 1) not require raw files and not be limited by the number of samples, 2) not produce ambiguous or multiple feature matches, 3) be written in a cloud-compatible programming language, and 4) align greater than two datasets, with the results not being influenced on insertion order.

To this end, we developed Eclipse (https://github.com/broadinstitute/bmxp). Eclipse uses a novel alignment strategy, running directed subalignments between some or all datasets before aggregating the results in a graph. The individual subalignments and graph can be exported or converted to tabular data using one of two built-in approaches. Eclipse is fast, accurate, and flexible with respect to the experiment. We demonstrate this by 1) aligning four ~13,000 feature datasets, running for 30 seconds and accurately matching 177 of 181 annotated features, 2) aligning five datasets of various tissue types to identify overlapping features, and 3) aligning all nine datasets, running between 39 and 125 seconds depending on mode.

ECLIPSE OVERVIEW

Eclipse aligns features among two or more datasets by comparing the features' descriptor values. By default the descriptors are RT, *m/z* and the average intensity. All descriptor comparisons and calculations are performed using transformed values, and residuals are normalized by their residual standard errors (RSE) to ensure accurate modeling and equal weighting among descriptors. By default, RT is treated with a simple addition or subtraction (*linear*), m/z by PPM (*ppm*), and intensity by log-scaling (*log*). Custom descriptors and transformation modes can be specified, potentially allowing for use outside of LC-MS.

Eclipse natively aligns greater than two datasets by running a whole alignment as independent

subalignments, where a Source->Target pair of datasets (e.g., DS1->DS2, DS2->DS1, etc.) is independently scaled and matched (**Figure 1**). The subalignment match results are then used to construct a whole-alignment feature graph. Subalignments are directed, meaning DS1->DS2 and DS2->DS1 are independent and distinct. By default, Eclipse runs All-by-All, requiring *n*(n-1)* subalignments. Optionally, a faster Ref-by-All strategy can be used, requiring only *2*(n-1)* subalignments. Since subalignments run independently of each other and the whole alignment, they may be run in parallel, or even piecemeal over time and deposited to a continually growing graph. The use of subalignments and a feature graph is the core of the Eclipse workflow; other individual steps (scaling, matching, deconvolution) can be replaced as better suited algorithms are identified.

Subalignment Determination of Scalers and RSEs

Prior to feature matching, the descriptors must be scaled and RSEs approximated. This is accomplished via a survey alignment of reduced Source and Target datasets. Reduced datasets are created by removing all features that fall within a specified range of any neighbors (default RT +/- 0.5 min, *m/z* +/- 15 ppm, intensity +/- 2 orders of magnitude), leaving a set of features we refer to as “anchors”. Source->Target anchor matches are identified by querying the Source features to the Target dataset, using the same ranges. The residuals of the matches are transformed and modeled by a LOWESS smoothing curve, which becomes the scalers. Finally, the RSE (modeling the expected inter-dataset noise) for each descriptor is calculated by applying the scalers to the residuals and calculating the standard deviation.

Next, the subalignment matching results from whole Source and Target datasets are determined. The Source dataset is scaled, and the best Target dataset matches are identified by finding features that fall within +/- 6 RSEs of the RT, *m/z*, and average intensity of the Source dataset. Candidate matches are ranked according to a penalty (Supplemental Equation 1) and the best match is recorded into the subalignment result table.

Feature Aggregation and Deconvolution

The results from each subalignment are combined to form a graph, with features as nodes and

individual subalignment matches as directed edges. To generate a tabular dataset of aligned features, one of two built-in deconvolution approaches is used to convert the graph into a tabular dataset: a strict mode and a dataset-centric mode. Both modes begin by recording bidirectional matches (i.e., two opposite subalignment matches) and removing remaining unidirectional matches. Strict mode only reports maximal cliques which contain a member from all datasets, discarding partial matches and preventing multiple match scenarios from occurring. Dataset-centric mode reports all maximal cliques that contain a feature from one or more specified datasets, potentially resulting in multiple rows per feature.

METHODS

Processed datasets, denoted as DS1 through DS9, were acquired on LC-MS instruments comprised of Shimadzu Nexera X2 U-HPLCs coupled to Thermo Exactive series orbitrap mass spectrometers. DS1-4 were created from pooled reference samples in a multi-batch. DS5-9 were derived from datasets of different rodent tissues. All datasets were acquired using the same HILIC-Pos method (Mascanfroni *et al.* 2015). Feature extraction was performed using Progenesis QI. Dataset information is summarized in Supplementary Table 1. All data used for analysis is available in the Supplementary Information.

All alignments were performed on an AMD Ryzen 5 3500x Windows 11 PC, running Python 3.8 and BMXP version 0.0.14. The benchmark times did not include file I/O, only the time taken to perform subalignments and feature aggregation/deconvolution. All Eclipse settings were left as default unless otherwise specified. The code used to run the alignment demonstrations can be found in Supplementary Code 1-5.

RESULTS

Same-Matrix Datasets

Our primary use and motivation for building Eclipse is to combine multiple datasets as part of a robust processing pipeline. To demonstrate the accuracy of Eclipse, two annotated datasets from a multi-batch study (DS1 and DS2) with 181 overlapping annotations were aligned, finishing in 7 seconds

(Supplementary Code 1). 6840 features were aligned between the two datasets, or 51% of the smaller dataset (DS1). 181 of 181 annotations (100%) were accurately matched. The DS1->DS2 RT scaling results are presented in Supplemental Figure 1. The individual subalignment scaling and matching results for RT, *m/z*, and intensity (including the similar but opposite DS2->DS1 results), are shown in Supplementary Figures 1-4.

To demonstrate Eclipse's ability to align more than two datasets, four human plasma datasets (DS1, DS2, DS3, DS4) were aligned, running in 30 seconds (Supplementary Code 2). Eclipse identified 3858 features (29% of the smallest dataset, DS1) and correctly matched 177 of 181 (98%) annotated features.

Five Disparate-Matrix Datasets

A secondary use of Eclipse is to identify equivalent features among biospecimens of different origins, such as different tissue types or different types of biological fluids. To demonstrate, rat plasma features (DS5) were aligned to features detected in rat gastrocnemius (DS6), rat liver (DS7), rat heart (DS8); and rat white adipose (DS9). An All-by-All alignment was performed, finishing in 29 seconds (Supplementary Code 3). Dataset-centric mode was used for feature deconvolution, setting plasma (DS5) as the reference for feature inclusion in the tabular results. Of the 12959 features in DS5, 1442 were found in all datasets, 4337 had partial matches (i.e. to one or more datasets), and 7180 did not have a match to any dataset. Out of the 140 annotated features which were present in all datasets, 129 were fully matched from plasma to all other datasets, 8 were partially matched, and 3 did not show any matches. We also tested Ref-by-All mode, rerunning the experiment but only performing subalignments which involved the reference dataset, DS5 (Supplementary Code 4). This required eight subalignments and finished in 18 seconds. Of the 12959 DS5 features, 1600 had matches to all other datasets, 3681 had partial matches, and like the results from the All-by-All alignment, 7180 had no matches. Similarly, 130 annotated plasma features were fully matched, 7 were partially matched, and 3 were not found.

Nine Datasets Benchmark

Finally, for benchmarking purposes, all nine datasets (human and rat) were run in both All-by-All

and Ref-by-All modes, requiring 72 and 16 subalignments (Supplementary Code 5). With intensity enabled, Eclipse ran for 125 and 39 seconds, and with intensity disabled, 97 and 39 seconds.

DISCUSSION

Alignment Demonstration Results

Eclipse demonstrates its accuracy and performance when combining processed datasets. In the two-dataset example, all 181 annotations were correctly aligned in a process which took under 7 seconds. In the four-dataset example, Eclipse ran for 30 seconds, successfully aligning 177 of 181 annotated features between all four datasets. Additionally, these results required no post-alignment manual intervention, such as resolving multiple matches. For our second case, we interrogated a reference dataset (DS5) to four others, in both All-by-All and Ref-by-All modes. All-by-All is useful when the relationship between non-reference datasets is of interest. Otherwise, Ref-by-All is sufficient and will run faster. This performance difference is highlighted in our final demonstration, aligning all nine datasets in times ranging from 39 and 125 seconds. This experiment also makes use of dataset-centric mode, which reveals partial matches at the expense of generating multiple hits. While this tradeoff is unacceptable for automated processing, it has value in exploratory experiments such as this.

Comparison to other tools

Currently no software offers the ability to align more than two datasets. In comparing to two-dataset programs, the most tools similar are an unpublished Eclipse precursor which has been used to assemble datasets (Tahir *et al.* 2022), metabCombiner (Habra *et al.* 2021), and M2S (Climaco Pinto *et al.* 2022). The tool used in Tahir *et al.* uses a nearly identical scaling and scoring approach compared to Eclipse, but lacked the graph generation abilities which enable $n>2$ dataset alignments. metabCombiner differs from Eclipse in that it uses an iterative GAM fit with outlier removal to generate the scaling factors, and different approaches for residual transformation and identifying matches. M2S uses very similar scoring and feature transformation approaches but differs in how matching is performed.

Limitations and Workarounds

Like other post-extraction alignment tools, Eclipse will only correctly align LC-MS features if the elution order is acceptably preserved and peak apexes are assigned properly. For problematic LC-MS features, we recommend keeping a list for targeted extraction and alignment. The most common problem we encounter is the failure to determine acceptable scalers and RSEs, which might arise if datasets are sparse or there are large deviations in descriptors. Workarounds include increasing the anchor or survey-alignment window, modifying the default LOWESS parameters, scaling the descriptors outside of Eclipse, or providing custom scalers.

Conclusion

In conclusion, we are excited to present Eclipse to the community. Eclipse handles critical steps in our processing pipeline, as such we intend to support it indefinitely, as long as alignment remains relevant to our workflow and research. Eclipse is open source, and we welcome feedback and new feature requests from the metabolomics community. The code, instructions, and examples can be found as part of a larger processing toolset used by the Broad MXP platform at https://github.com/broadinstitute/bmxp.

REFERENCES

Brunius C, Shi L, Landberg R. Large-scale untargeted LC-MS metabolomics data correction using between-batch feature alignment and cluster-based within-batch signal intensity drift correction. *Metabolomics* 2016;**12**:173.

Chen Z-Z, Pacheco JA, Gao Y *et al.* Nontargeted and Targeted Metabolomic Profiling Reveals Novel Metabolite Biomarkers of Incident Diabetes in African Americans. *Diabetes* 2022;**71**:2426–37.

Climaco Pinto R, Karaman I, Lewis MR *et al.* Finding Correspondence between Metabolomic Features in Untargeted Liquid Chromatography–Mass Spectrometry Metabolomics Datasets. *Anal Chem* 2022;**94**:5493–503.

Clish CB. Metabolomics: an emerging but powerful tool for precision medicine. *Cold Spring Harb Mol Case Stud* 2015;**1**:a000588–a000588.

Habra H, Kachman M, Bullock K *et al.* metabCombiner: Paired Untargeted LC-HRMS Metabolomics Feature Matching and Concatenation of Disparately Acquired Data Sets. *Anal Chem* 2021;**93**:5028–36.

Koch S, Bueschl C, Doppler M *et al.* MetMatch: A Semi-Automated Software Tool for the Comparison and Alignment of LC-HRMS Data from Different Metabolomics Experiments. *Metabolites* 2016;**6**, DOI: 10.3390/metabo6040039.

Mahieu NG, Patti GJ. Systems-Level Annotation of a Metabolomics Data Set Reduces 25 000 Features to Fewer than 1000 Unique Metabolites. *Anal Chem* 2017;**89**:10397–406.

Mak TD, Goudarzi M, Laiakis EC *et al.* Disparate Metabolomics Data Reassembler: A Novel Algorithm for Agglomerating Incongruent LC-MS Metabolomics Datasets. *Anal Chem* 2020;**92**:5231–9.

Mascanfroni ID, Takenaka MC, Yeste A *et al.* Metabolic control of type 1 regulatory T cell differentiation by AHR and HIF1-α. *Nat Med* 2015;**21**:638–46.

Pluskal T, Castillo S, Villar-Briones A *et al.* MZmine 2: Modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinformatics* 2010;**11**:395.

Smith CA, Want EJ, O'Maille G *et al.* XCMS: Processing Mass Spectrometry Data for Metabolite Profiling Using Nonlinear Peak Alignment, Matching, and Identification. *Anal Chem* 2006;**78**:779–87.

Smith R, Ventura D, Prince JT. LC-MS alignment in theory and practice: a comprehensive algorithmic review. *Brief Bioinform* 2015;**16**:104–17.

Tahir UA, Katz DH, Avila-Pachecho J *et al.* Whole Genome Association Study of the Plasma Metabolome Identifies Metabolites Linked to Cardiometabolic Disease in Black Individuals. *Nature Communications* 2022;**13**:4923.

Vatanen T, Jabbar KS, Ruohtula T *et al.* Mobile genetic elements from the maternal microbiome shape infant gut microbial assembly and metabolism. *Cell* 2022;**185**:4921-4936.e15.

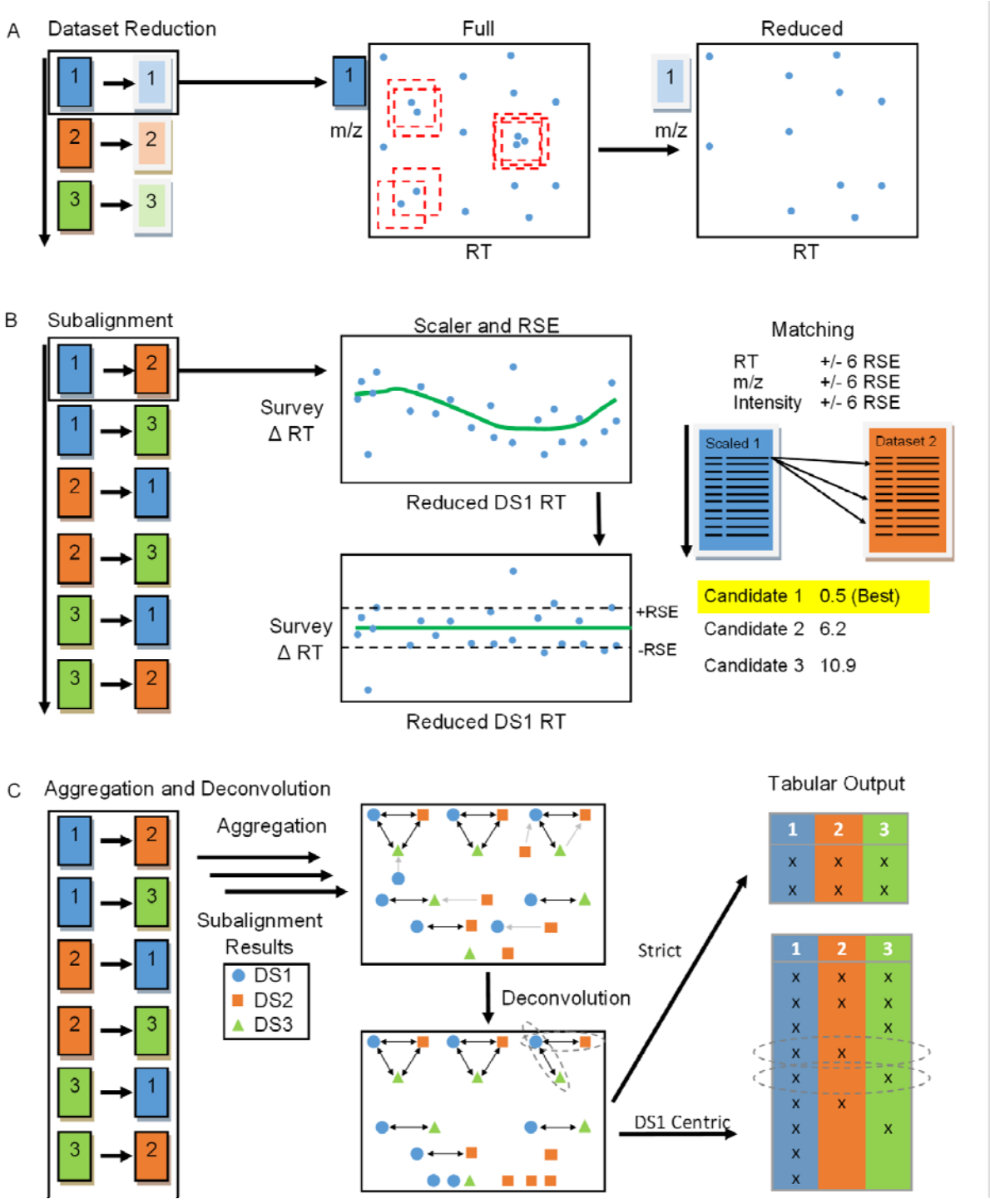
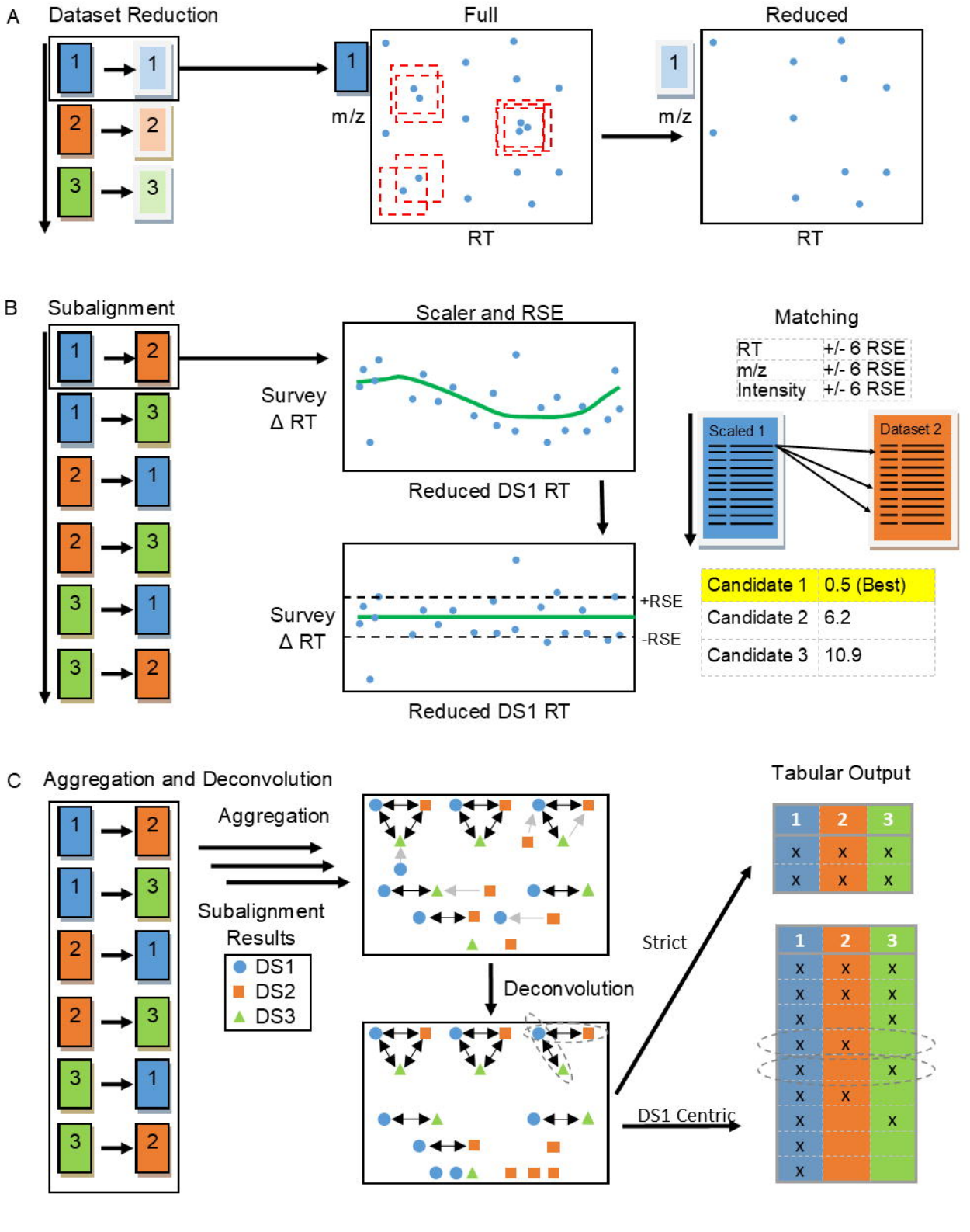


Figure 1. Eclipse workflow with a three-dataset example. A) Reduced datasets are produced from each dataset by removing all features that fall within a specified descriptor distance of another B) Subalignments begin. Reduced datasets are used in survey alignments to calculate scalers and residual RSEs for each descriptor. The full Source dataset is scaled and queried to the Target (+/- 6 RSEs) to candidate matches, which are ranked. C) The best match results from each subalignment are loaded into a graph as directed edges. The graph is reduced to only bidirectional edges (black) and one-way matches (gray) are removed. The graph can be deconvoluted based on two built-in approaches. The strict mode keeps only feature subgroups in cliques containing all datasets. The dataset centric mode (DS1 Centric)

records all feature subgroups which contain a member of a dataset, in this case DS1, potentially creating multiple matches.


A
Dataset Reduction
Full
Reduced
m/z
RT
B
Subalignment
Scaler and RSE
Survey
Δ RT
Reduced DS1 RT
Matching
RT +/- 6 RSE
m/z +/- 6 RSE
Intensity +/- 6 RSE
Scaled 1
Dataset 2
+RSE
-RSE
Candidate 1 0.5 (Best)
Candidate 2 6.2
Candidate 3 10.9
C
Aggregation and Deconvolution
Aggregation
Subalignment Results
DS1
DS2
DS3
Deconvolution
Strict
DS1 Centric
Tabular Output