UNIVERSITY OF CALIFORNIA
RIVERSIDE

Reproducible Analysis of Complex Data & Prediction of Bioactive Natural Compounds

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Genetics, Genomics & Bioinformatics

by

Le Zhang

June 2023

Dissertation Committee:

Dr. Thomas Girke, Chairperson
Dr. Zhenyu Jia
Dr. Wenxiu Ma

The Dissertation of Le Zhang is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I sincerely thank my main advisor Dr. Thomas Girke for the support, guidance, and mentorship throughout my time at UC Riverside. I also thank other members of my committee, Dr. Zhenyu Jia and Dr. Wenxiu Ma. Their support, expertise, and constructive feedback have helped me to become a better researcher and have significantly contributed to the success of my projects.

I am grateful for the support and assistance that I have received from my dear friends, colleagues, and fellow graduate students at UC Riverside. I am particularly grateful to my lab colleagues and collaborators, Dr. Daniela Cassol, Jianhai Zhang, Lei Yu, Dr. Brendan Gongol, and Dr. Yuzhu Duan, whose contribution to my research work has been nothing less than exceptional.

I am eternally grateful to my family, Jun Zhang, Yunju Xiao, and Jingyan Guo, for their unconditional love and unwavering care. It is difficult to put into words just how much they mean to me, and for that reason, I want to express my deepest thankfulness to them.

ABSTRACT OF THE DISSERTATION

Reproducible Analysis of Complex Data & Prediction of Bioactive Natural Compounds

by

Le Zhang

Doctor of Philosophy, Graduate Program in Genetics, Genomics & Bioinformatics
University of California, Riverside, June 2023
Dr. Thomas Girke, Chairperson

This dissertation presents three projects that are described in individual chapters. The first two projects are about software development for reproducible research involving big data, and the third project is about large-scale discovery of drug-like natural compounds. The following briefly summarizes each chapter.

Chapter 1 introduces the general research area of each project. These introductions focus on the status of the field, current challenges and future needs. At the end of each project introduction, an overview is given about the specific solutions proposed and implemented by the research of this dissertation.

Chapter 2 describes the design and development of *systemPipeR*. This generic software environment provides flexible utilities for designing, building and running automated end-to-end analysis workflows for a wide range of research applications. Important functionalities include a uniform workflow interface across different data analysis applications, automated report generation, and support for running both R and command-line software, such as big data analysis tools, on personal computers, HPC clusters, and cloud systems.

The third chapter introduces *systemPipeShiny*. This web system extends the *systemPipeR* workflow environment with a versatile graphical user interface provided by a Shiny App. It allows non-R users to run many of *systemPipeR*'s workflow design, control, and visualization functionalities interactively without requiring command-line knowledge. It also integrates highly interactive visualization functionalities and a graphics workbench.

Chapter 4 is a large-scale discovery project of bioactive natural compounds. First, the known structures and annotations of about 0.5 million natural compounds and about 4,000 known drugs were organized in a rational database. Second, the assembled database was used for virtual screening, where a series of supervised and non-supervised machine learning methods was applied to predict drug-like natural product candidates. This also identified drugs that have natural compound alternatives with identical structures and *vice versa*. Third, known drug-target annotations were used to systematically identify which molecular mechanisms and disease processes are perturbable by bioactive natural compounds. This study identified a large panel of new bioactive natural compounds with interesting applications in human health, including healthy aging and anticancer.

Contents

List of Figures **x**

List of Tables **xvi**

1 Introduction **1**
1.1 Data Analysis Workflows . . . 1
1.2 Workflow Environments . . . 2
1.2.1 Feature Requirements of Analysis Workflow Environments . . . 3
1.2.2 *systemPipe* Workflow Tools . . . 5
1.2.3 Comparison of Workflow Environments . . . 6
1.3 Command-Line Interface . . . 6
1.4 Other Interface Options . . . 7
1.5 Features of GUI . . . 8
1.6 Computational Drug Discovery . . . 9
1.6.1 Represent Compounds Computationally . . . 9
1.6.2 Common Methods to Find Compound Similarity . . . 11
1.7 Natural Products . . . 14
1.7.1 Opportunities Provided by BNPs . . . 15
1.7.2 Major Natural Product Databases . . . 16
1.7.3 Challenges of Analyzing BNPs . . . 16
1.7.4 Objectives . . . 17
1.8 Organization of Dissertation . . . 18

2 systemPipeR: Workflow and Visualization Toolkit **19**
2.1 Abstract . . . 19
2.2 Introduction . . . 20
2.3 Methods . . . 23
2.3.1 Overview . . . 23
2.3.2 Workflow Design Structure Using *SYSargsList* . . . 24
2.3.3 Command-Line Software Support . . . 26
2.3.4 Workflow Modularity . . . 28
2.3.5 Annotation System . . . 28

2.3.6 Reports Generation . . . 29
2.3.7 Command-Line Software Support . . . 29
2.3.8 Visualization . . . 30
2.4 Discussion . . . 30
2.5 Conclusion . . . 32

3 systemPipeShiny: Interactive Framework for Workflow Management and Data Visualization 33
3.1 Abstract . . . 33
3.2 Introduction . . . 34
3.3 Materials and Methods . . . 37
3.3.1 Overview . . . 37
3.3.2 Workflow Module . . . 38
3.3.3 Data Visualization Module . . . 41
3.3.4 Image Processing Module (Canvas) . . . 43
3.3.5 Advanced Features . . . 43
3.4 Results and Discussion . . . 46
3.4.1 Novel Features . . . 46
3.4.2 Reproducibility . . . 47
3.4.3 Use Case Examples . . . 48
3.5 Conclusions . . . 52
3.6 Additional Figures . . . 53

4 Large-Scale Prediction of Natural Product Alternatives for Known Drugs 60
4.1 Abstract . . . 60
4.2 Introdcution . . . 61
4.3 Methods . . . 65
4.3.1 Data Collection . . . 65
4.3.2 Compound Cleaning, Annotation, and Calculation . . . 65
4.3.3 Similarity Computation . . . 67
4.3.4 BNP ML Models . . . 68
4.3.5 BNP Validation/Use Case Set . . . 69
4.3.6 Unsupervised Search . . . 70
4.3.7 Downstream Annotations . . . 71
4.3.8 Functional Enrichment Analysis . . . 71
4.3.9 Substructure Search & ML Models . . . 72
4.3.10 Statistical Comparisons . . . 72
4.3.11 Other Packages Used . . . 73
4.3.12 Reproduce and Customize the Results . . . 73
4.4 Results . . . 75
4.4.1 Databases Cleaning and Overview . . . 75
4.4.2 Estimate Raw BNP Number . . . 76
4.4.3 Search for Drug Alternative BNPs . . . 78
4.4.4 Combine Search Results . . . 80
4.4.5 Functional Annotation of BNPs . . . 82

4.4.6 BNP Set Validation . . . 84
4.4.7 Functional Enrichment . . . 85
4.4.8 Explore BSDs and BUDs . . . 87
4.4.9 Predict BUDs . . . 90
4.4.10 Safety of BNPs . . . 94
4.4.11 Use Case of BNPs . . . 95
4.5 Discussions . . . 99
4.5.1 Extend the Use Cases . . . 99
4.5.2 Limitations . . . 101
4.5.3 Other Applications . . . 101
4.6 Conclusions . . . 102
4.7 Tables . . . 103
4.8 Additional Figures . . . 106

5 Conclusions and Future Work **115**
5.1 Conclusions . . . 115
5.2 Future Work . . . 116

Bibliography **117**

List of Figures

1.1 Relationship between workflow environment, workflow, and data process step 3
1.2 Example of substructure-based FP conversion. The blues circles indicate a substructure that can be transformed into FP features. Each row of the table indicates the FP of each compound. 12

2.1 Overview of *systemPipeR* workflows management instances. A) A typical analysis workflow requires multiple single software (red), as well the description of the input data (green), and the expected outfiles and reports analysis (purple). B) *systemPipeR* provides multiple utilities to design and build a workflow, allowing multi-instance, integration of R code and command-line software, a simple and efficient annotation system, that allows automatic control of the input and output data, and multiple resources to manage the entire workflow. c) The execution of the analysis is independent of the design and build, enabling portability and more meaningful code sharing. *systemPipeR* provides options to execute a single step or multi-steps in a compute session environment, enabling scalability, resources to re-run one or multi-steps, checkpoints, and generation of execution reports that can track parameters and versions, providing transparency and data provenance. 24
2.2 Example of "Hello World" message using CWL syntax and demonstrating the connectivity with *systemPipeR*. (A) This file describes the command-line tool, here using 'echo' command. (B) This file describes all the parameters variables connected with the tool specification file. Here the reference value of the input parameter can be specific or can be filled dynamically, adding a variable that connects with the targets files from *systemPipeR*. (C) *SYSargsList* function provides the 'inputvars' arguments to build the connectivity between the CWL description of parameters and the targets files. The argument requires a named vector where each vector element is required to be defined in the CWL description of parameters file (B), and the names of the elements are needed to match the column names defined in the targets file (D). 27

3.1 SPS Design Overview. (A). Module categories in *SPS*, including workflow management, data visualization and image processing. (B) Customization features allowing users to personalize content, appearance, and other components. (C) Extensions to Shiny to improve user experience and software management. (D) Functionalities to export data. (E) Add-on packages published on CRAN providing visualization tools and supporting future *SPS* development. 37
3.2 Workflow Module. It comprises six steps for configuring, running, and monitoring workflows. Steps 1-2 are for choosing workflows, and configuring samples and metadata. Step 3 provides access to an interactive viewing and editing tool for workflows where topologies can be changed by intuitive drag-and-drop actions. Step 4 is optional, allowing expert users to work with CWL configuration files directly. Step 5 is for initializing workflow runs or porting them to other systems. Step 6 provides an array of workflow monitoring functionalities. Solid and dashed lines indicate mandatory and optional actions, respectively, as well as connectivity to other modules. 39
3.3 Visualization Module of *SPS*. (A) Display of tabular data. (B) Plot window with RNA-seq data as an example. (C) Quick *ggplot* interface. Step-wise illustration of a common use case: (1) Workflow results are uploaded for visualization. If needed, common data pre-processing and filtering routines can be applied. (2) Upon confirming the dataset, different plotting options are available. (3) Most plots are interactive, *e.g.* mousing hovering displays values. (4) Numerous customization options are provided, such as styling of graphical components. (5) The code for generating a plot can be accessed and saved for reproducibility. (6) Any plot instance can be sent to the Canvas workbench or saved to a file. 41
3.4 Canvas Module of *SPS*. The Canvas Module has two main parts: (1) the workbench and (2) the image browser. (3) After dragging one or more plot thumbnails to the Canvas workbench, users can rearrange and edit them as needed. (4) Image annotations can be added via a text panel. (5) Images can also be imported from local files. (6) Several menu options are available to change general settings, such as export options, help, and keyboard shortcuts. (7) Extended image editing options are provided, such as layer opacity. . . 42
3.5 SPS Usage Paradigm. (A) Data analysis steps in SPS. A typical data analysis starts with the Workflow Module where users can design, edit and run workflows. When using a pre-configured workflow, the design and editing steps are optional. The Visualization and Image Processing Modules can be used to plot the results and to assemble composite images, respectively. If needed, users can revise the results by repeating the previous steps. Source code, image and tabular data are downloadable as individual files or batch archives. (B) Most functionalities can be accessed by selecting the corresponding icons on SPS's welcome page. 53

3.6 RNA-Seq Utilities. (1) When starting from a read count table, the corresponding tabular file and a sample annotation (targets) information need to be provided as input. (2) After validating the input, the raw read counts are normalized, and (3) differentially expressed genes (DEGs) identified. (4) The results can be visualized with different plotting routines. Subsequently, multiple plots can be post-processed, and a detailed analysis report generated. Dashed lines indicate optional actions. 54
3.7 Quick ggplot Utility. (1) Uploaded tabular data can be displayed, filtered, and validated. (2) Table columns to include in a plot can be selected under the first two rows. (3) The plot type can be chosen from a large collection. (4) Plotting styles can be revised at the bottom, including legends, shapes, symbols, font size, line width, and color palettes. The "To canvas" button on the top sends the current plot to SPS' Canvas module. 55
3.8 Administration & Security Management. (A) Activated analytics features can be inspected on SPS' administrator page. (B) If needed, user authentication can be activated by the administrator of an SPS deployment. When turned on, users are required to provide their access credentials (username and password) to access a protected SPS instance. (D) Accessing these features requires administrative permissions. 56
3.9 Flowchart of SPS deployment. During an SPS deployment, a hidden environment is created that stores configuration information, data, and other components. Default or custom configuration options are validated. After this, the process split into UI- and server-specific steps. Next, the different components are assembled into a ready-to-be-launched Shiny instance. . . 57
3.10 SPS Architecture. The connectivity of the SPS components is illustrated in a network diagram. Components that always need to be loaded are indicated by solid lines. Dashed lines indicate optional components that can be disabled. 58
3.11 Run custom workflows. (1) The existing option on the workflow setup page allows users to load custom workflows. (2) Path of workflow templates and targets are required for custom workflows. 59

4.1 Overall Design of the Study. This study can be divided into three major sections. First, NPs and approved drugs were collected from different sources. Annotations were added and physicochemical properties were calculated. Second, drug alternative BNPs were identified based on the intersection of structural similarity, physicochemical similarity, and drug likelihood results. Third, identified BNPs were annotated with additional information, such as source food, source organism, toxicity, pathways, diseases, and more. 74
4.2 Statistics of compounds. (A) Intersection table of NPs, drugs, and their source databases. On each tile, the top number indicates the number of compounds shared (overlapping) between two data sources, and the bottom number indicates the shared identical compound percentage between two data sources. (B) Intake method statistics of approved drugs. (C) NP source organism statistics (N.A. values removed). (D) Top 20 NP source food statistics. 76

4.3 History relationship between approved drugs and NPs. (A) Approved drug numbers from 1940 to 2020. The total number of drugs approved yearly is marked in green columns, and the green line represents the trend. Drugs that were identical to their NP form (as-NP) were marked in red dots, and the red line represents the trend. Drugs that were highly similar to NPs (high-similarity-to-NP) are marked in pink dots, and the pink line represents the trend. Withdrawn drugs each year are represented as negative numbers, and the brown line represents the trend. (B) The relative percentage of different types of drugs. The as-NP drug is represented in red, and the approximate area is estimated under the red line. The high-similarity-to-NP drugs are represented in pink. Low-similarity-to-NP drugs are represented in gray color. The area of as-NP and high-similarity-to-NP drugs together is estimated under the blue line. 77

4.4 Drug alternative NP search design and results. (A) Detailed design of ML search for drug alternative NPs. (1) Structural FPs and PPs, were calculated separately and later used as two types of inputs for the ML models. Invalid compounds were removed in the reprocessing stage. (2) A relaxed filter was performed to reduce the FP and PP inputs sample size to around 40,000 before applying ML methods. (3) STS distance matrix and PES distance matrix each served as input for Hclust and Kmeans models separately. PP values were also used to train a drug-likelihood assessment RF model. (4) Results from all models were filtered in the intersection table. (B) The table displays one ML method (X-axis) intersects with all other methods (Y-axis). Notation of axis labels: Hierarchical clustering (HC). Kmeans (KM). results of STS models (STS HC, STS KM), results of PES models (PES HC, PES KM), the union of PES/PES Hclust models (uHC), the union of PES/PES Kmeans models (uKM), and the PP-based drug likelihood RF model (RF). More complicated intersection notations are: intersection of uHC and uKM (uHCiuKM), and the most strict set, the intersection of STS HC/Kmeans models intersecting with the intersection of all PES HC/Kmeans models (iSTSiiPES). The three numbers on each tile are: first row, the number of BNPs candidates (starting with "B"); second row, the number of BSDs (starting with "D"); third row, the overlapping percentage between two methods. The color code on each tile is based on stringency value reference LBR and UBR. The color is red if the number is below LBR, blue if the number is above UBR, and green if the number is in between. The final set for downstream analysis is marked in the black box. 79

4.5 Validation of BNPs. (A) BNP-drug pairs were matched to validate if they have the same gene targets, MOA, MOA type, MeSH indications, and pathways. The numbers on each bar are: the percentage of matching, the number of matching vs. the total number of BNP-drug pairs found with available information. (B) DrugAge-associated BNP compound names, DrugAge compound names, and the same number of random NP names were searched separately on PubMed for publication records. The search keywords were "longevity" + compound names. The number of publication found under each category is displayed on a boxplot (C) Similar PubMed search was performed on anticancer drugs and their associated BNPs, with the first keyword changing to "anticancer". Notation for p-value: *** ≤ 0.01. 83
4.6 BSDs and BUDs. (A) Reactome enriched pathways overview firework plot of substitutable and unsubstitutable drugs. (B) Top 20 Reactome enriched pathways of substitutable (green) and unsubstitutable drugs (yellow). The order is sorted by adjusted p-value in each category separately. The pathway ratio was calculated by using the number of genes enriched divided by the total number of genes in the pathway. (C) (D) Approved year difference with substitutable and unsubstitutable drugs. 86
4.7 Characteristics between BSDs, BUDS, and NPs. (A) Top 10 frequency different substructures in BUDs *vs.* NPs. (B) Number of nitrogen atoms in different compound groups. (C) tSNE plot of different compound groups. . 88
4.8 Model benchmarks for BUD prediction. First row (A-B), binary classification models. Second row (C), trinary classification models. (A) ROC curve of different binary models. The color code is displayed in (B) as row names. (B) Metrics of different binary models, sorted by accuracy (C) Metrics of different trinary models, sorted by accuracy. 91
4.9 BNP toxicity and drug interactions. (A) Violin plot of mice LD50 toxicity predictions calculated by TEST for different compound groups. The larger the value, the more toxic it is. (B) Network plot example of drug interactions. The top 5 drugs with the most BNP partners are selected in this example. Each blue dot connection to the red circle indicates a BNP molecule association. The size of the red circles indicates the number of drug interactions with other drugs (not shown on the plot). The gray lines indicate drug interactions between these 5 drugs. p-value indication: ***, < 0.001. 93
4.10 Source information of BNPs. (A) Top 10 food sources of all BNPs. (B) Top 10 food sources of longevity-associated drug BNPs. (C) Top 5 food sources of anticancer drug BNPs. (D) Top 10 organism sources of all BNPs. (B) Top 10 organism sources of longevity-associated drug BNPs. (A-E) were ranked by relative abundance. (F) The number of dietary supplement products found for different categories of molecules. The two numbers on each bar are: the raw numbers of compounds and the percentage of compounds under each category found to have dietary supplement products. 96

4.11 AA content in NPs. The AA percentage in NPs is compared to the percentage of protein compounds in ChEMBL database. The solid line indicates the 1% quantile of the reference content of proteins in ChEMBL database. 106
4.12 Optimization of K for Hclust and Kmeans. The green lines indicate the number of clusters obtained when K is increased. The red lines indicate when K is increased, the trend of the number of clusters containing at least one drug and one NP (n). The optimal K is obtained when the max n is reached for all 4 clusterings. 107
4.13 Performance of stack model and RF model for predicting drug likelihood. (A) ROC curve of models. (B) Error rate of models predicting on independent test compound set. This set contains NPs and drugs at clinical phases 1-4. 107
4.14 Results in a Database. All results from this research are collected into an SQL database. Tables generated from this research are located in the Primary Tables and Other Internal Tables columns. The primary key of each table is in bold. Connections between tables are marked in solid arrows. Connections to outside sources are marked in dashed lines. 108
4.15 Determination of the threshold of RF model. (A) The cross-validation error rate on predicting drugs, NPs, and both on average. (B) The independent test dataset error rate. The solid line indicates the optimal threshold for cross-validation. The dashed line indicates the threshold selected in this study for downstream analysis. 109
4.16 Top 10 MOA terms of BSDs (A), BUDs (B), and BUD exclusive terms (C). The blue terms in (A) are top 10 BUD terms that do not exist in top BSD terms. 110
4.17 Nine typical examples of BUD structures 111
4.18 N/C ratio between NPs, BSDs (substitutable), and BUDs (substitutable). . 112
4.19 Top 10 frequency different substructures. (A) BUDs *vs.* BSDs. (A) BSDs *vs.* NPs. Both plots are sorted by the substructure frequency difference from left to right (blue column). 113
4.20 Annotation rate of NPs. (A) Source annotate rate of NPs. (B) Functional annotation rate of NPs. 114

List of Tables

1.1 Comparison of design principles of different workflow management tools . . 6
1.2 Example of a PP matrix 13

2.1 Selected functions categories. The table lists a subset of over 50 methods and functions defined by *systemPipeR*. Usage instructions are provided in the corresponding help pages and vignettes of the package. 31

3.1 Important Configuration Options of SPS. The table lists the most important configuration options of SPS installs. They control the general behavior and visual design of a deployment. 44

4.1 All compound metadata. This table includes some logical columns. "1" means "Yes" or "Present"; "0" means "No" or "Unpresent". 103
4.2 All compound PPs. Values are calculated from JDK. SMILES invalid compounds were removed. PP names are listed in abbreviation. Refer Table. 4.3 for the full names. 103
4.3 Full PP descriptor names for columns in Table. 4.2. 103
4.4 CTS mapping information. This table includes possible compound ID mappings from this study to other databases. 103
4.5 All compound KEGG mapping. Compound ID mappings from this study to KEGG. 103
4.6 BNP metadata. General information of BNPs selected in this study, including MOA, targets, disease, source, *etc.* 104
4.7 BNP metadata use drug ID as primary key. This table is similar to Table. 4.6, but instead of using BNP as the key, this table is summarized by drug IDs. 104
4.8 DrugAge BNP PubMed validation. Number of publications search results using DrugAge drugs and DrugAge-associated BNPs. 104
4.9 Anticancer BNP PubMed validation. Number of publications search results using anticancer drugs and anticancer-associated BNPs. 104
4.10 BSD BUD pathway enrichment. Raw PFEA results for both BSDs and BUDs. 104

4.11 Cleaned Global Chem SMARTS for common compounds. This table contains common chemical structures as SMARTS, which are summarized from various studies by Global Chem. Empty strings removed. 104
4.12 Raw substructure search results. Substructure queries made in rdkit. If a compound has a substructure, it records as "1", otherwise "0". 104
4.13 Substructure frequency difference of NP BUD BSD. This table has substructure counting and substructure frequencies for NPs, BSDs, BUDs. It also has substructure frequencies differences between groups, and the table is sorted based on substructure frequencies between BUDs and NPs. 104
4.14 Kruskal Wallis Test of the importance of features. This table records Kruskal Wallis Test results using all other compound features to test against the type classification column (BUD, BSD, NP). 104
4.15 Best substructure binary stack model composition. Individual models and their importance in the best-performing BUD-NP predicting stack model. . 104
4.16 LD_{50} Toxicity predictions by TEST for all compounds. 104
4.17 Drug interactions summarized from Drug Bank queries. 105
4.18 Drug combination adverse effects cleaned from pAERS database. 105
4.19 Raw compound food source mapping information for all compounds, cleaned from FooDB. 105
4.20 Food source enrichment for all BNPs, longevity BNPs, and anticancer BNPs. 105
4.21 Source organism genus enrichment for all BNPs, longevity BNPs, and anticancer BNPs. 105
4.22 Food supplement product information. This table contains food supplement product information for all compounds, fetched from DSLD database. . . . 105

Chapter 1

Introduction

1.1 Data Analysis Workflows

Data analysis workflows include several computing steps that are chained together to process the input data in an orderly fashion. Usually, the output of a previous workflow step becomes the input for the next step. Several data pre- and post-processing steps, as well as statistical and visualization routines are used in many steps. A typical data analysis workflow can be summarized by the following general steps (Scott Long, 2009):

1. Assembly and cleaning of data
2. Organization in efficient data structures
3. One or more data processing steps
4. Statistical tests
5. Data visualization
6. Report generation

Data analysis workflows in biology and biomedical sciences often process assay data collected from high-throughput instruments such as next-generation sequencers, mass

spectroscopy, cell sorters or plate readers. The initial steps in an analysis workflow involve pre-processing and quality control of the raw data. Next, quantitative and semi-quantitative analysis steps are performed that determine abundance differences or profiling of biomolecules (*e.g.* transcripts, proteins, metabolites). Alternatively, presence or absence of biological features can be identified in discovery approaches involving genetic variants, structural changes or other features. Omics studies often aim to assemble complete or partial genomes, transcriptomes or proteomes. Downstream of these assay-specific analysis steps, various annotation and enrichment steps are applied that incorporate information from existing biological databases. To interpret the results, visualization techniques are used for different workflow steps. Widely used scientific visualization routines include scatter, bar and box plots, Venn diagrams, volcano plots, heatmaps or genome graphics. Finally, the results are assembled in scientific reports.

1.2 Workflow Environments

Data analysis workflow environments refer to integrated systems or frameworks that facilitate the seamless execution of data analysis tasks and provide a structured approach to handling and processing data (Oinn et al., 2006). These environments typically serve as control systems of a combination of tools, libraries, and frameworks that are used in various stages of the data analysis process. Namely, a single workflow is the management of different data process steps, but a workflow environment is a control system of different workflows. The relationship between the workflow environment, workflow, and data

process step can be understood in the following diagram (Fig. 1.1). Therefore, workflow environments are high-level management tools built on top of data analysis workflows.

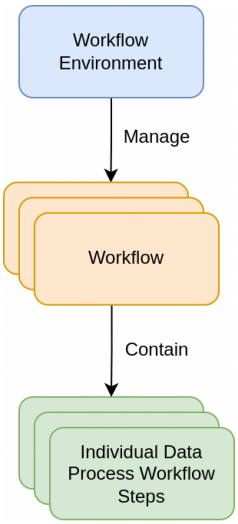

Figure 1.1: Relationship between workflow environment, workflow, and data process step

Data analysis workflow environments are essential because they provide a structured and organized approach to data analysis. They ensure reproducibility, facilitate collaboration, improve efficiency, and address scalability challenges. Researchers can define a logical and reproducible sequence of tasks through workflow environments, promoting transparency and enhancing the credibility of scientific findings. Workflow environments also foster collaboration and knowledge sharing among researchers, enabling simultaneous work and sharing of workflows, code, and data.

1.2.1 Feature Requirements of Analysis Workflow Environments

Typical bioinformatic workflow environments need to satisfy the following features (Wratten et al., 2021).

(1) *Automation.* Automating data analysis workflows is crucial for efficiency, consistency, and collaboration. It saves time and effort by standardizing analysis processes and automating repetitive tasks. This ensures that operations are consistently applied, enhancing reproducibility and promoting transparency. Automation also enables scalability through parallel processing and distributed storage for large datasets. Additionally, it reduces the risk of human error and fosters collaboration and replication by allowing the sharing of automated workflows, facilitating knowledge exchange and cross-validation of results.

(2) *Resource management.* To gain meaningful insights from a variety of sources, it is crucial to analyze different types of data from different data structures in a data analysis workflow. Researchers can capture various dimensions of information by including numerical, categorical, textual, or temporal data. Different data structures, such as tabular data, hierarchical data, graphs, or unstructured formats, provide unique representations of information and require specific analysis techniques. By incorporating these diverse data types and structures in the workflow environment, researchers can use appropriate analytical methods, tools, and algorithms to uncover patterns, trends, correlations, and anomalies. Analyzing different types of data within a data analysis workflow environment enhances the depth and breadth of insights, enabling researchers to address complex research questions effectively.

(3) *Scalability.* When it comes to analyzing data, scalability is key. It allows for the handling of large datasets and resource-intensive analyses. As data continues to grow at an exponential rate, researchers require scalable solutions that can efficiently process and analyze massive amounts of information. Scalability ensures that the workflow adapts to increasing data sizes, computational complexity, and user demands, all while maintaining performance

and accuracy. Utilizing distributed computing, parallel processing, and effective resource allocation, a scalable workflow environment can tackle the challenges posed by big data. This empowers researchers to explore complex patterns, derive meaningful insights, and make data-driven decisions on a larger scale.

(4) *Reproducibility.* Ensuring reproducibility is crucial in data analysis workflow environments, especially in scientific research, as it establishes the credibility and dependability of research findings. Reproducible results allow for independent verification and validation of analyses, which strengthens the scientific method and maintains the integrity of research outcomes. By providing a transparent and documented record of data, methodologies, and analysis workflows, reproducibility allows other researchers to replicate and build upon previous work, encouraging collaboration and cumulative scientific progress. Reproducibility also simplifies peer review, allowing experts to evaluate and scrutinize methods and results.

1.2.2 *systemPipe* Workflow Tools

To address the need for workflow environments that satisfy the requirements listed above, a set bioinformatic workflow environment software, *systemPipe* workflow tools, is implemented in this dissertation. *systemPipe* workflow tools provide flexible utilities for designing, building, and running automated end-to-end analysis workflows for a wide range of research applications. The biggest feature of *systemPipe* tools is its integration of command-line software with the R/Bioconductor ecosystem in a centralized control system. Other key features include but are not limited to the reusable/extendable workflow design, reproducible analysis with parallelization support, and a highly customizable graphical user interface. Details of this tool are further discussed in Chapter 2 and Chapter 3.

1.2.3 Comparison of Workflow Environments

Besides the *systemPipe* workflow tools, other similar works are briefly introduced here (Table. 1.1). Some famous examples include, *Galaxy*, an web-based open-source platform for analyzing and visualizing large amounts of biological data, *Snakemake*, a Python-based open-source platform for creating and executing reproducible bioinformatics workflows, *Nextflow*, a platform for creating and executing scalable and reproducible bioinformatics workflows, and *Common Workflow Language (CWL)*, an open-source language framework for creating and executing workflows in a portable and scalable manner.

Table 1.1: Comparison of design principles of different workflow management tools

Workflow Environment	**Design Principles**
SystemPipe	Integration of code, data, and analysis documentation. Emphasis on reproducibility and integration to R ecosystem.
Snakemake	Simplicity, scalability, and reproducibility through explicit rules and dependencies.
Nextflow	Scalability, portability, and reproducibility with support for diverse computing environments.
Galaxy	Ease of use, emphasis on its graphical interface, and comprehensive capturing of analysis processes.
CWL	Portability, standardization, and reproducibility through a platform-agnostic syntax.

1.3 Command-Line Interface

Most workflow management environments are developed with the command-line interface (CLI). The CLI is a text-based interface that allows users to interact with a computer program or operating system by typing commands into a terminal or command prompt. It provides a way to execute commands, run programs, and perform various

tasks using a series of text-based instructions. To initiate specific actions or operations, one typically types commands with specific syntax and parameters. CLI is widely used in software development, system administration, and data analysis workflows, as it provides a powerful and flexible means of interacting with computer systems and applications through a textual interface.

1.4 Other Interface Options

While CLI provides a systematic way for users to automate the workflow, however, operating such tools requires some terminal operation knowledge. Not all researchers have coding experience. On the other hand, software with a graphic user interface (GUI) is usually easier to use. GUIs also provide graphical representations of data and results, making it easier to understand and interpret results. In addition, setting or changing parameters on a GUI can be as simple as a few mouse clicks, which makes it easier to control the behavior of the tool. Last but not least, today's GUIs are usually web-based applications, which are convenient for multiple users to collaborate on analysis, as users can simplify communication and sharing of results through the internet.

Therefore, designing a GUI for the workflow manager is necessary to benefit a broader range of researchers. For the workflow managers mentioned above (Table 1.1), only Galaxy (Jalili et al., 2020) has native GUI support (GUI developed by the same group that developed the command-line interface), but does not have native command-line support. For other platforms, the GUI comes from a third-party developer, such as DolphinNext for Nextflow (Yukselen et al., 2019), and Sequanix for Snakemake (Desvillechabrol et al., 2018).

1.5 Features of GUI

Well-designed GUIs are essential for non-expert users. They improve user experience, increase productivity, and knowledge extraction from data (Fleming, 1998; Bødker, 2021; Tidwell, 2010). To achieve this, they should include the following features. 1) Intuitive design: A good GUI should have an intuitive and straightforward design that makes it easy for users to navigate and find what they need. This includes using clear and consistent visual elements and using common design patterns and conventions. 2) Ease of use: The GUI should be easy to use and require minimal training. Users should be able to complete tasks quickly and efficiently without needing to consult manual or online resources. For example, To enhance user experience, it is advisable to incorporate frequently used widgets and styles from popular websites into the new GUI. This will enable users to navigate and adapt to the new interface easily. 3) Feedback: The GUI should provide users with clear and meaningful feedback on their actions. This includes visual alerts, error messages, and confirmation pop-ups that help users understand the outcome of their actions. For example, if some processes take a long time to complete, suitable messages and progress bars should inform the user about what to expect. When exceptions happen, users should be informed of the error message, *e.g.* in pop-up windows or operation logs. 4)Accessibility: Designers should also consider the user's physical abilities or disabilities. This includes ensuring that the interface is usable with keyboard-only navigation, providing alternative text for images, and using high-contrast color schemes. 5) Reproducibility: GUIs for data-centric applications often run heavy data analyses in the background. These applications require multiple steps and may require long computing time. In order for users to resume an analysis session

later, repeat certain processes, or transfer results to another platform. It is important that the GUI supports these reproducible data process features.

1.6 Computational Drug Discovery

Computer-assisted drug discovery (CADD) is one of the essential methods for drug discovery today. By using the computing power from massive high-performance computing (HPC) clusters, it has become an effective way to accelerate and economize drug filtering and discovery (Ou-Yang et al., 2012).

Common use cases of CADD include (Sliwoski et al., 2014; Ou-Yang et al., 2012): (1) Filter large compound databases to find smaller sets that can be further tested experimentally; (2) Guide the optimization of compounds by calculating properties such as affinity, metabolism, and kinetics; (3) Designing new compounds; (4) Identification of targets. With trained models and statistical methods, the protein binding sets are predictable.

1.6.1 Represent Compounds Computationally

Simplified Molecular Input Line System (SMILES, Wiswesser, 1985) is a line notation representation of molecular structure in the form of a string of characters. It is used to encode the molecular structure of a chemical compound in a machine-readable format.

SMILES provides a way to represent the molecular structure of a compound, including its atoms, bonds, and stereochemistry. The representation is based on a set of simple rules and uses ASCII characters to encode the molecular structure, and it is computer-readable. The SMILES string for a molecule can be easily translated into a graphical

representation of its structure, making it a useful tool for chemical informatics and computational chemistry. However, the way to convert compounds to SMILES varies depending on the algorithm and version used. For example, some versions ignore the stereochemistry while others do not. It results in different representations of the same compound in different databases, which makes it difficult to search across sources.

Later, The International Chemical Identifier (InChI) was introduced (Heller et al., 2015). InChI describes molecules in terms of the compound layer structure in computer-readable format– atoms and connected bonds, charge, isotopes, and stereochemistry information are described layer by layer. Unlike other formats, InChI provides a unique and consistent identifier for a chemical compound, regardless of its source or representation, and can be used for chemical database searching and data exchange.

When a compound is searched in the database, one wants to input as little information as possible to identify the compound without ambiguity. SMILES can be short, but it is not unique. InChI provides a unique identifier, but as the complexity of the compound increases, the length of InChI grows rapidly. For example, the InChI for benzene is "InChI=1S/C6H6/c1-2-4-6-5-3-1/h1-6H", with 34 characters, but with one carboxylic group attached to the ring (benzoic acid), the InChI becomes "InChI=1S/C7H6O2/c8-7(9)6-4-2-1-3-5-6/h1-5H,(H,8,9)", with 50 characters. One can expect the length of large molecules can be problematic as search keywords. For example, the InChI for insulin has 2655 characters which is unusable for a database as the index.

Commonly, the hashed version of InChI is used as a database index. It is also called the InChIKey. It has a fixed length of 27 characters, no matter how complex

the molecule is. For instance, the InChIKeys for benzene, benzoic acid, and insulin are: "UHOVQNZJYSORNB-UHFFFAOYSA-N", "WPYMKLBDIGXBTP-UHFFFAOYSA-N", "PBGKTOXHQIOBKM-FHFVDXKLSA-N" respectively. This format is human-unreadable but friendly to computers due to the fixed length. Unlike InChI, the hashed InChIKey is not unique, and it has about a 0.014% chance of collision (Pletnev et al. 2012). Overall, the chance is low, and it is generally safe to use InChIkey as the search keyword. Some occasional collisions can be manually filtered.

1.6.2 Common Methods to Find Compound Similarity

Compound similarity measures how closely related two chemical compounds are in terms of their molecular structure and properties. Before any similarity calculation is applied, compounds need to be standardized to computer-readable formats, such as numeric numbers. Two major standardization methods that are highly related to this thesis are discussed below.

The first method is molecular fingerprints (FPs). FP is a binary representation (0, and 1) of the molecular structure of a chemical compound. Each bit represents a specific feature of the molecule. For example, FP can be used to describe if a certain substructure is present in a compound (Fig. 1.2). If it does, the number of that bit is recorded as 1, otherwise 0. The choice of features to include in a molecular fingerprint will depend on the specific requirements of the research project and the nature of the data being analyzed. For example, the substructure key-based fingerprints are used in Molecular ACCess Systems keys fingerprint (MACCS) and PubChem (PubChemFP), whereas topological path-based fingerprints are used AtomPairs2DFingerprint (APFP) (Seo et al., 2020).

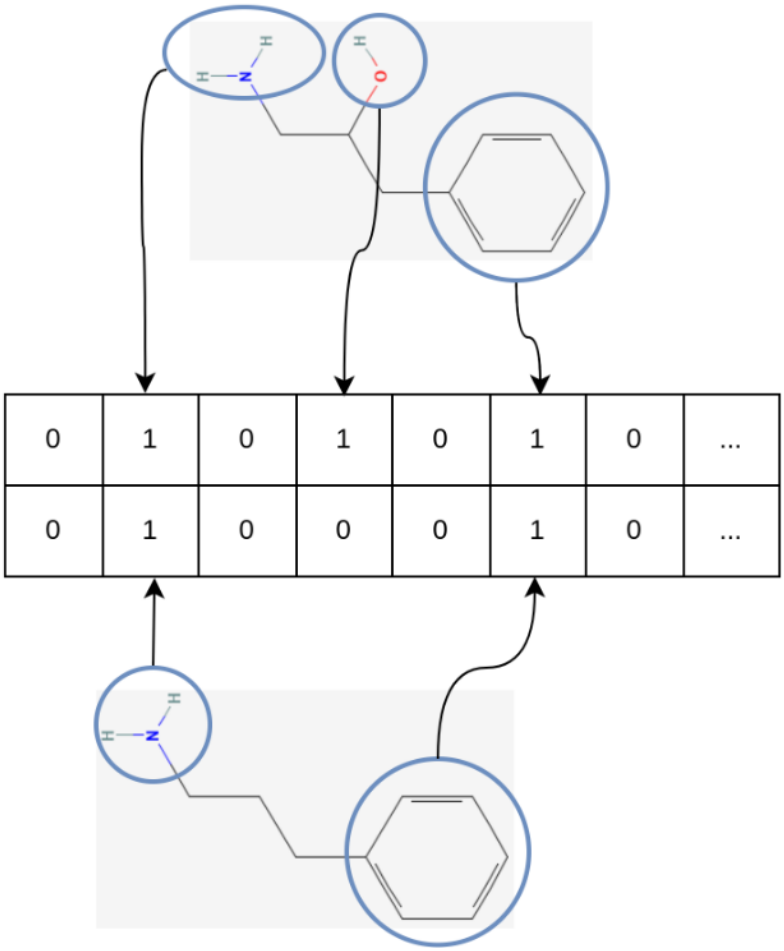

Figure 1.2: Example of substructure-based FP conversion. The blues circles indicate a substructure that can be transformed into FP features. Each row of the table indicates the FP of each compound.

To calculate the similarity of FPs, Tanimoto Similarity (TS) is often used (Bajusz et al., 2015). It describes the ratio of the number of features common to both molecules to the total number of features.

The calculation can be done as follow:

$$T_{A,B} = \frac{A \cap B}{A + B - A \cap B} \tag{1.1}$$

Where $T_{A,B}$ is the TS between compound A and B, A and B are the number of features in each compound respectively.

The second method is physiochemical properties (PPs). PPs are numerical values that describe the molecular properties of a compound, such as its molecular weight, number of hydrogen bond donors, and polar surface area (Table. 1.2). The similarity calculation can be done on the matrix-like data structure, where compounds are represented by rows and PPs are presented as columns.

Table 1.2: Example of a PP matrix

ID	Molecular Weight	Polarity Index	Surface Area	Property XXX
cmp1	150	5	202	...
cmp2	233	−2	345	...
cmp...	...	...	...	...

The simplest way to obtain similarities of PPs between two compounds is by calculating the Euclidean Similarity (ES) (Liberti et al., 2014).

$$ES_{A,B} = \frac{1}{1 + \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2}} \tag{1.2}$$

Where $ES_{A,B}$ is the ES between compound A and B, $\sqrt{\sum_{i=1}^{n}(A_i - B_i)^2}$ is distance between A and B given n properties.

1.7 Natural Products

Natural products (NPs) are substances produced by living organisms, including plants, animals, fungi, and bacteria. NPs have a wide range of chemical structures, from simple compounds like sugars and amino acids to complex compounds like alkaloids and polyketides (Harvey, 2008). In addition, NPs have been used for centuries as traditional medicines (Che et al., 2016), and they continue to be important sources for developing new drugs for treating a variety of diseases. NPs are highly related to the food and agriculture industries. For example, spices, essential oils, flavors, and pigments extracted from plants can be used as food additives (González-Manzano and Dueñas, 2021). Pesticides and herbicides derived from NPs are often considered safer and more environmentally friendly than synthetic alternatives (Yan et al., 2018).

When a subset of NPs exhibit specific biological activities or pharmacological effects, they become bioactive natural products (BNPs). These compounds can interact with various biological targets in the human body, such as proteins, enzymes, receptors, or DNA, and modify their function. Similar to NPs, BNPs have a broad range of therapeutic properties (Colegate and Molyneux, 2007).

1.7.1 Opportunities Provided by BNPs

First, nature has carefully chosen BNPs through evolutionary processes to interact with biological targets and modify particular pathways or processes. Consequently, they usually have strong and specific activity against their intended targets, making BNPs desirable starting points for drug discovery initiatives (Berdy and Kertesz, 1989). BNPs also offer the potential for identifying new therapeutic targets and mechanism of action (MOA), thereby expanding the options for drug development and improving treatment strategies (Li and Lou, 2018). In fact, BNPs are a major source for today's drug selection. BNPs and BNP derivatives made up most of the drugs approved by the U.S. Food and Drug Administration (FDA) before 2000 (Sneader 1997).

Next, BNPs can be used for healthy diet guidance, including their incorporation as dietary supplements. Valuable essential nutrients, including vitamins, minerals, and other bioactive compounds, are enriched in BNP-containing sources. Consuming diets or food supplements enriched with BNPs instead of generic NPs, we can target specific health concerns and optimize the nutritional value of our diets for certain health concerns or medical condition treatments more easily (Weymouth-Wilson, 1997).

Moreover, BNPs are often safer options for long-term use in disease prevention strategies compared to many synthetic compounds. They have evolved in natural systems and tend to have lower toxicity levels (Qian et al., 2010), which makes them well-tolerated by the human body and reduces the risk of adverse effects. This property makes BNPs the great generally favorable for chronic disease prevention, such as neurological diseases or cardiovascular diseases.

1.7.2 Major Natural Product Databases

There are many NP databases available. Some of them are open-access, and some of them are commercially available. Famous open-access NP databases are Super Natural II (Banerjee et al., 2014), ChEBI (Wohlgemuth et al., 2010), and 3DMET (Maeda and Kondo, 2013). Examples of commercial databases are Chemical Abstracts Service (Gabrielson, 2018), Berdy's Bioactive Natural Products Database (Berdy and Kertesz, 1989), and Reaxys (Reaxys, 2023). In recent years some aggregation databases (super collections of other databases together) have been created, such as COCONUT (Sorokina and Steinbeck, 2020) and Natural Products Atlas 2.0 (van Santen et al., 2022).

1.7.3 Challenges of Analyzing BNPs

Despite the many opportunities and resources available for NPs and BNPs, this field still has challenges. One of the biggest challenges is the low annotation rate. While many NP databases have well-characterized NPs' chemical and structural properties, few describe the bioactivities of BNPs, including protein targets, MOAs, pathways, and more (Lachance et al., 2012). Identifying specific biological activities requires extensive experimental validation and the integration of data from a variety of sources.

One of the good approaches to utilizing low-annotated NPs in drug discovery is to search for NPs that are similar to existing drugs. Some researchers have already conducted studies on this. For example, some researchers compared natural products with existing drugs i*in vitro* to identify potential antibacterial drugs and anti-mitotic drugs (Ayon, 2023;

Mazzio et al., 2014), while others screened for anticancer drugs from NPs in *in silico* (van Leeuwen et al., 2022).

However, a major drawback of these studies is that they often focus on specific areas, leaving many potential therapeutic areas unexplored. This limited scope makes it difficult to analyze and identify BNPs that could be drug-linked candidates comprehensively. Additionally, the absence of a centralized and comprehensive database hinders researchers from accessing relevant information, comparing results, and identifying potential leads for drug development. To address this issue, there is a need for collaborative efforts to establish large-scale screening initiatives and comprehensive databases that encompass a wider range of therapeutic areas, facilitating the discovery of novel drug alternatives from natural products. Such initiatives would provide valuable insights and accelerate the identification of promising candidates for various diseases and conditions. Other insights into challenges are discussed in Chapter 4.

1.7.4 Objectives

Considering the opportunities and challenges NP presents, Chapter 4 presents a comprehensive analysis of NPs through machine learning-based approaches. The aim is to identify potential drug-like BNP candidates. The specific objectives are outlined below.

1. Conduct computational screening of large-scale drug alternative BNP candidates.
2. Utilize more robust and reasonable methods, such as machine-learning-based algorithms, to screen potential BNP candidates instead of using a fixed threshold.

3. Investigate the reasons why some of the drugs can be found with drug-like BNP candidates while some others cannot. Explore the functional and structural differences between these compounds.
4. Establish connections between drugs and their corresponding BNP candidates to elucidate the unknown bioactivities of BNPs, including their biological functions, pathways, and protein targets.
5. Demonstrate the value of the annotated BNP candidate set through various biomedical and drug discovery applications. Examples include providing dietary recommendations for longevity-promoting diets and identifying potential BNP-containing organisms for screening alternative anticancer drugs.

1.8 Organization of Dissertation

The following chapters present the research projects of this dissertation. The first project is about the development of a generic data analysis workflow environment for reproducible research. The second project extends the workflow environment with an interactive GUI. The third project is a large-scale discovery study of drug-like natural compounds.

Chapter 2

systemPipeR: Workflow and Visualization Toolkit

2.1 Abstract

The *systemPipeR* software, developed by this project, establishes a versatile environment for designing, building and running end-to-end analysis workflows on local machines, HPC clusters, and cloud systems while generating at the same time publication quality analysis reports. Pre-configured workflow templates are provided for a wide range of omics applications. The package offers sufficient flexibility to choose for each step in a workflow the optimal R or command-line software, customize workflows and design entirely new ones while taking advantage of community S4 classes of the Bioconductor ecosystem. Thus, it serves as a federation hub for integrating both R and command-line based software, that makes efficient use of existing software without limiting users to tools from a single

community project. It has been designed to improve the reproducibility of large-scale data analysis projects, while substantially reducing the time it takes to analyze complex omics data sets. Recently developed improvements include the following enhancement. First, the Common Workflow Language (CWL) has been integrated for defining and executing workflows. Second, a graphical workflow management interface has been added allowing users to visualize workflow designs in different graphical layouts, and to monitor their run status while tracking all metadata associated with a project. Third, detailed scientific and technical reports are now automatically generated during workflow runs. Fourth, a workflow control class (S4) has been added allowing users to execute single or any number of complex workflow steps with a single command. Fifth, a Shiny interface has been created to support interactive graphics in workflow reports.

2.2 Introduction

Rapid technological advances have led to large-scale and high-throughput data generation, transforming biological science research. These phenomena associated with the heterogeneous and complex nature of biological data require the development of an extensive ecosystem of specialized applications (Goodwin et al. 2016). Frequently, multiple single applications are required to be combined in multi-steps analysis to extract meaningful knowledge from the data. The integration of multi-steps, combining various applications with a specific set of configurations, is usually called computational workflows or pipelines. Typically, each step requires one or more command-line tools, a set of specific inputs, and

outfiles definition, which can be used in the downstream analysis stage, besides particular configurations settings and/or computing resources for each step.

Low-level details in computational biology, such as raw data, algorithms, scripts, and parameters used, associated with the description of the procedures, are essential to inspect any research's findings. Reproducibility, which allows finding the same results given the same initial data set and parameters configurations (Wratten et al. 2021), is crucial to improving research reliability and promoting scientific advancement. For example, sharing the methodology description associated with raw data is not sufficient to make the reproducible research (Hothorn and Leisch, 2011) because a minor modification in the software or reference annotation version, or a different set of parameters can alter the results significantly. Therefore, the computational protocols should be easy to access, not demand high programming skills, and in a readable format, improving access to the intermediate and final results for any researcher. To promote reproducible research is necessary to deliver the software environment, computing code, and data with the publication to allow others to recreate the study and achieve the same conclusions (Hicks and Peng, 2019).

To address the requirement of design and creation, data management (input and outfiles), and environment execution of the multi-step, many workflow management systems (WMS) have been developed. For example, applications like Taverna (Wolstencroft et al., 2013), Pegasus (Deelman et al., 2015), and Galaxy (Afgan et al., 2018) offer integration of web services and local resources to create, manage, and execution of workflows. In addition, domain-specific languages like Nextflow (Di Tommaso et al., 2017) (written in Groovy - a scripting language for the Java Virtual Machine), Snakemake (Köster and Rahmann, 2012

(Python), Workflow Description Language (WDL) (www.openwdl.org/), Common Workflow Language (CWL) Amstutz et al. 2016) (YAML and/or JSON) aim to simplify the creating of complex and common analysis pipelines and present specific strengths and benefits for particulars goal or individual preferences (Reiter et al., 2021). Each system differs in the implementation, target audience, and learning curve but ultimately attempts to simplify standard bioinformatics analyses.

While many workflow management resources exist for various data analysis programming languages, only insufficient general-purpose workflow solutions are currently available for the popular R programming language. R and the affiliated Bioconductor environment provide many widely used tools with a large user base in this area (Huber et al., 2015*a*). Thus, a workflow framework for generic applications from within R will benefit experimental and computational scientists using R for omics data analysis.

Previously, an early version of *systemPipeR* was presented for building and running workflows focused on NGS applications with support for integrating a wide array of command-line and R/Bioconductor software. Previous versions implemented an ad-hoc solution for the command-line software by adding the arguments settings in a tabular *param* file (H Backman and Girke, 2016*a*). This implementation had several limitations that were addressed in the new development, for example: (i) using a domain-specific language for describing the command-line tools/software; (ii) workflow execution monitoring; (iii) a web interface to support non-expert users who are not familiar with data analysis programming environments like R. This chapter presents the new infrastructure of *systemPipeR*, high-

lighting the new developments for modular design, building, execution of the workflows, and features of reusability and reproducibility of data analysis.

2.3 Methods

2.3.1 Overview

The software *systemPipeR* is implemented as an open-source Bioconductor (Gentleman et al., 2004; Huber et al., 2015*a*) package, using R's programming language (R Core Team, 2021). Its integration with the Bioconductor project promotes the re-usability of genomics software components efficiently using many of the existing packages that are well tested and widely used by the community. systemPipeR new workflow management control allows integrating individual command-line tools and R-based code in a unique instance, connecting the data analysis steps, and coordinating the input and expected outfiles for each step in a robust and efficient manner (Fig. 2.1A-B).

The associate data package provides direct access to the pre-configurated workflow and demo data sets to test all the available features quickly. All the packages and templates are available for all the common operating systems, followed by complete documentation and template examples. Third-party command-line software can be used in any workflow analysis, and therefore, if the software is operating system-specific, the workflow execution may be limited.

To support long-term reproducibility of analysis outcomes, *systemPipeR* is also distributed as Docker and Singularity image. Docker and Singularity containers provide an efficient solution for packaging complex software together with all its system dependencies

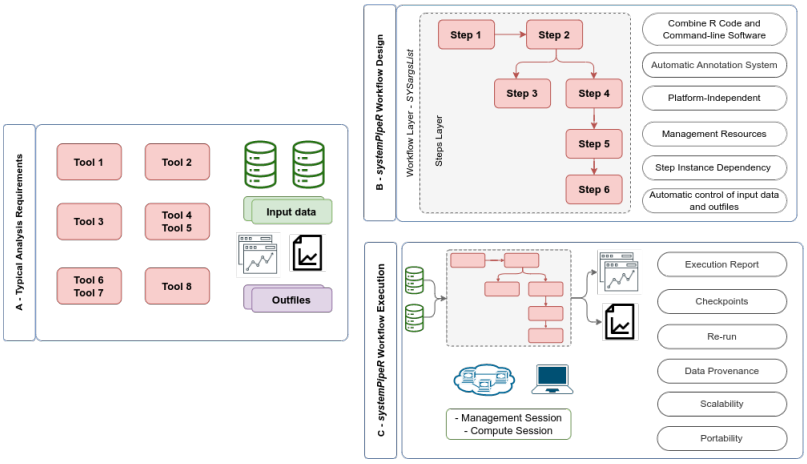

Figure 2.1: Overview of *systemPipeR* workflows management instances. A) A typical analysis workflow requires multiple single software (red), as well the description of the input data (green), and the expected outfiles and reports analysis (purple). B) *systemPipeR* provides multiple utilities to design and build a workflow, allowing multi-instance, integration of R code and command-line software, a simple and efficient annotation system, that allows automatic control of the input and output data, and multiple resources to manage the entire workflow. c) The execution of the analysis is independent of the design and build, enabling portability and more meaningful code sharing. *systemPipeR* provides options to execute a single step or multi-steps in a compute session environment, enabling scalability, resources to re-run one or multi-steps, checkpoints, and generation of execution reports that can track parameters and versions, providing transparency and data provenance.

to ensure it will run the same in the future across different environments, including various operating systems and cloud-based solutions.

2.3.2 Workflow Design Structure Using *SYSargsList*

SYSargsList enables easy integration with the Bioconductor ecosystem, which provides access to a large and diverse number of analysis tools. Furthermore, the S4 class allows interoperability between many different ecosystem tools, presenting several advantages. First, the usage of generic workflow environment objects and class advances reproducibility for the users, allowing the integration with the existing R/Bioconductor ecosystem and

external (third party software/tools). Also, it improves maintainability since it is a formal object-oriented system with a strict structure, avoiding code duplication and promoting reusability.

SYSargsList is an S4 class list-like container where each instance stores all the input/output paths and parameter components required for a particular data analysis step. *SYSargsList* instances are generated by two-parameter files for command-line steps or R code for R-base steps. The connectivity among all workflow steps is achieved by the *SYSargsList* workflow control class (Fig. 2.1).

When running pre-configured workflows, the only input the user needs to provide is the initial targets file containing the paths to the input files (which can be optional) along with unique sample labels. Subsequent targets instances are created automatically. The CWL description files provide the parameters required for running command-line software.

SYSargsList class provides a sophisticated container for workflow infrastructure to design, build, run, monitor, and share workflow analysis. The flexibility of *systemPipeR*'s new interface workflow control class is the driving factor behind the use of as many steps necessary for the analysis, as well as the connection between command-line- or R-based software. The connectivity among all workflow steps is achieved by the SYSargsList workflow control class. This S4 class is a list-like container where each instance stores all the input/output paths and parameter components required for a particular data analysis step. SYSargsList instances are generated using as data input targets or yaml files as well as two cwl parameter files. When running pre-configured workflows, the only input the user needs

to provide is the initial target file containing the paths to the input files along with unique sample labels. Subsequent target instances are created automatically (Fig. 2.2).

A specific analysis in the workflow may involve many tools to achieve one final result. Therefore, those tools need to be grouped in sub-steps or a nested workflow. *systemPipeR* facilitates the designer of those sub-steps, promoting reusability and reproducibility of workflows (Fig. 2.2A).

systemPipeR design layer is an interoperable software library and is organized in layers. The first layer (step- layer gray) corresponds to the original tool or R code-based step. It provides the module with a well-defined interface for input, output, configuration, and provenance. It performs internally the necessary format conversions at input and output and storage all the code to be executed. The second one, the workflow layer, provides s a uniform and stable interface, allowing to plug the components into interoperable workflows. This layer controls the integration between the steps, as the interdependencies and status of each step. The configuration allows integration with HPC systems. Also, it enables the location of each final outputs and the reports (Fig. 2.1C).

2.3.3 Command-Line Software Support

systemPipeR adopted the widely used community standard Common Workflow Language (CWL) (Amstutz et al., 2016) for describing command-line tools and workflows in a declarative, generic, and reproducible manner. CWL specifications are text-based files and can be structured using YAML (https://yaml.org/) syntax. Therefore, the description files can be easily accessed and are readable. The significant advantage of adopting CWL as a standard description of command-line tools within *systemPipeR* is the flexibility of

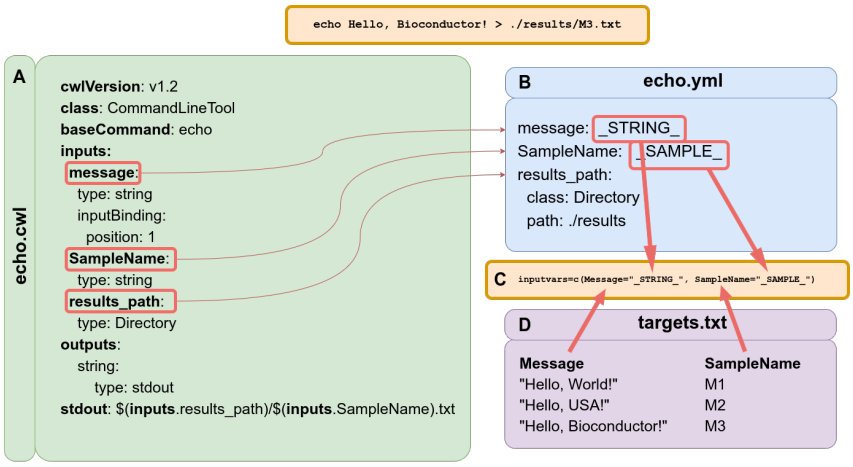

Figure 2.2: Example of "Hello World" message using CWL syntax and demonstrating the connectivity with *systemPipeR*. (A) This file describes the command-line tool, here using 'echo' command. (B) This file describes all the parameters variables connected with the tool specification file. Here the reference value of the input parameter can be specific or can be filled dynamically, adding a variable that connects with the targets files from *systemPipeR*. (C) *SYSargsList* function provides the 'inputvars' arguments to build the connectivity between the CWL description of parameters and the targets files. The argument requires a named vector where each vector element is required to be defined in the CWL description of parameters file (B), and the names of the elements are needed to match the column names defined in the targets file (D).

workflow reusability for different computing environments and workflow frameworks in the community, improving reproducibility, portability, and shareability between collaborators and community.

Following the CWL Command Line Tool Description Specification, the basic elements of the CWL tool description are defined in two files. Figure 2.2A-B illustrate the "hello world" example. The main file contains all the information necessary to build the command-line that will be executed, specifying the input, expected output files, and arguments for the command-line (Figure 2.2A). The second file is optional yet provides flexibility

to assign values to parameters required to make the input or output objects when building the command-line.

2.3.4 Workflow Modularity

systemPipeR presets a modular system, enabling reusability of the entire workflow for a different data set, and also the reusability of some steps a modular design might be useful; for example, some command- line tools could be reused in multiple workflows.

2.3.5 Annotation System

Metadata are the descriptions of sampling sites and habitats that provide the context for sequence information. Metadata is of great importance for metagenomic sequence data for two reasons. First, only by fully describing the samples from which metagenomics sequences have been obtained can one have any possibility of replicating a study. Samples from environmental or biological sources can never be fully replicated, but it is important that samples be sufficiently well described for an independent researcher to have the possibility of re-sampling. Second, metadata is essential for the analysis of metagenomic sequence data. Metagenomic sequence data that lack an environmental context have no value. Also, *systemPipeR* makes use of the Bioconductor ecosystem, integrating the target files with *SummarizedExperiment* (Morgan et al. 2021). This integration allows efficient and fast conversion to a *SummarizedExperiment* object.

2.3.6 Reports Generation

systemPipeR generates automated analysis reports with *Rmarkdown*. These modern reporting environments integrate R code with LaTeX or Markdown. During the evaluation of the R code, reports are dynamically generated in PDF or HTML format. A caching system allows to re-execute selected workflow reporting steps without repeating unnecessary components. This way one can generate reports that resemble a research paper where user-generated text is combined with analysis results. This includes support for citations, auto-generated bibliographies, code chunks with syntax highlighting, and inline evaluation of variables to update text content. Data components in a report such as tables and figures are updated automatically when rebuilding the document and/or rerunning workflows partially or entire.

2.3.7 Command-Line Software Support

systemPipeR adopted the widely used community standard Common Workflow Language (CWL) (Amstutz et al. 2016) for describing parameters analysis workflows in a generic and reproducible manner. Using this community standard in systemPipeR has many advantages. For instance, the integration of CWL allows running systemPipeR workflows from a single specification instance either entirely from within R, from various command-line wrappers (e.g., cwl-runner) or from other languages (e.g., Bash or Python). systemPipeR includes support for both command-line and R/Bioconductor software as well as resources for containerization, parallel evaluations on computer clusters along with the automated generation of interactive analysis reports. *systemPipeR* also provides several convenience

functions that are useful for designing and debugging workflows, such as a command-line rendering function to retrieve the exact command-line strings for each data set and processing step prior to running a command-line.

2.3.8 Visualization

systemPipeShiny package has been added that allows users to design workflows in an interactive graphical user interface (GUI). This allows non-R users, such as experimentalists, to run many systemPipeR's workflow designs, control, and visualization functionalities interactively without requiring knowledge of R. Most importantly, *systemPipeShiny* has been designed as a general purpose framework for interacting with other R packages in an intuitive manner. Like most Shiny Apps, *systemPipeShiny* can be used on both local computers as well as centralized server-based deployments that can be accessed remotely as a public web service for using *systemPipeR*'s functionalities with community and/or private data. The framework can integrate many core packages from the R/Bioconductor ecosystem.

2.4 Discussion

The flexibility of *systemPipeR*'s new interface workflow control class is the driving factor behind the use of as many steps necessary for the analysis, as well as the connection between command-line- or R-based software. The core approach for any workflow management is organization and effective management of the data, tools, and configuration settings. *systemPipeR* controls the workflow through an S4 class called *SYSargsList* (Fig.2.1).

Table 2.1: Selected functions categories. The table lists a subset of over 50 methods and functions defined by *systemPipeR*. Usage instructions are provided in the corresponding help pages and vignettes of the package.

Utilities	Description
Build	Functions to generate templates, build the directory structure and build SYSargsList workflow control module instances
Execution	Executes command-line software and R-code specified in SYSargsList instances
Visualization	Plot visual workflow designs and topologies with different graphical layouts
Report	Render Scientific and Technical report based on RMarkdown
Print Instances	Several functions to return the overview of the workflow control instances, print specific aspects of the workflow components, like the workflow steps computational status
Replacement Methods	Replacement methods of the workflow control instances, replacing specific aspects of the workflow components and parameters
Export	Export a workflow to bash script or RMarkdown file

systemPipeR package offers a series of features to design, build, run, monitor, and share workflow analysis within R programming and command-line environments. In addition, the *systemPipeR* package provides the computational infrastructure to standardize and streamline data analyses. Key features of *systemPipeR* include: (i) the workflow management system provided by the SYSargsList class, an S4 class; (ii) integration with R code-based and command-line software, using a community standard to describe the workflows; (iii) annotation system, with tracking provenance between the multi-steps analysis and the downstream metadata is automatically controlled by the workflow container; (iv) extensive visualization capabilities, and integration with shiny apps; and (v) generation of scientific and technical reports.

Omics data analysis has become more complex, as a result of advantages in sequencing technologies, imposing challenges for all the professionals required to analyze the data. Or that need quick and efficient results to decide the next step in the lab, quick as-

sessment of the data, of the research outcome. These scenarios impose a challenge because the user is required to connect multiple steps, with a large number of samples, replicates, and in some cases integrate multiple types of data.

The FAIR principles (Wilkinson et al. 2016) establish a guide to enhance the reusability and support easy access to the data, algorithms, and workflows. *systemPipeR* follows the guidelines providing the distribution of the software through the Bioconductor ecosystem, with the source code and documentation available in GitHub and with a central landing website. The software is open and broadly accessible from various sources, including Docker and Singularity containers.

2.5 Conclusion

systemPipeR provides a solution for developing and running dynamic and complex workflow analysis through a flexible framework interface suitable for experimentalist researchers. The conceptual continuity between the features provided by the *systemPipeR* brings different aspects of a single long-term goal, data analysis reproducibility, combined with easy integration of relevant software analysis from diverse languages and execution in different sources, leveraging it towards the extraction of knowledge of complex biological data.

Chapter 3

systemPipeShiny: Interactive Framework for Workflow Management and Data Visualization

3.1 Abstract

Data analysis has become extremely complex in life sciences, making it increasingly challenging for non-experts to analyze their data in a competent and time-efficient way while maintaining sufficient reproducibility. Effective visualization for identifying patterns in large data sets is another challenge. To address these needs, I introduce systemPipeShiny (*SPS*)

as an easy to use and extendable environment for intuitive large-scale data analysis and visual knowledge discovery.

SPS enables non-expert users to design, customize and execute data analysis workflows with intuitive to use drag-and-drop operations. It integrates extensive visualization functionalities where users can interactively arrange composite figures on a graphics workbench. To maximize reproducibility, one can access and record the code used during an analysis, and auto-generate publication quality reports. I have implemented *SPS* as a Shiny app users can run on local systems, cloud services and web servers. To simplify future workflow development, I have developed several modularized Shiny extensions for *SPS*.

3.2 Introduction

The rise of large-scale profiling technologies in life and biomedical sciences has resulted in the development of a wide range of software tools for analyzing the acquired big data, including omics data (Lightbody et al., 2019). Efficient analysis of these data requires assembling a series of software tools into a data analysis workflow comprising various data processing steps that are executed in a coordinated manner. Assembling and running data analysis workflows is essential for today's high throughput data analysis (Stoudt et al., 2021; Fjukstad and Bongo, 2017) to assure results can be auto-generated fast and are reproducible. Several workflow management environments are available in most programming languages commonly used for data analysis (Wratten et al., 2021). Many of them make use of language agnostic workflow specification standards, such as the Common Workflow Language CWL (Crusoe et al., 2021) and the Workflow Description Language WDL (Voss et al., 2017), that

formalize rules for defining workflows. With the resulting higher level of abstraction, one can separate workflow specification from execution, which boosts workflow portability. Most workflow management environments use command-line interfaces (CLI) that require some level of scripting experience. This creates a barrier for non-experts to use these workflows. Graphical user interfaces (GUIs) eliminate this barrier for non-expert users. Examples of workflow systems with a GUI include Rabix, Galaxy, and DolphinNext (Kaushik et al., 2017; Jalili et al., 2020; Yukselen et al., 2019). While these tools are easy to use, they often lack sufficient flexibility in customizing analysis steps and visualizing results.

In life and biomedical data science areas, the most widely used programming languages are R, Python, Bash and Java (Russell et al., 2018). Several workflow management tools implemented in these languages use CWL or WDL to specify workflows. I have developed the workflow environment systemPipeR (SPR; H Backman and Girke, 2016*b*) that is implemented in R and uses CWL for specifying workflows. As a generic and highly flexible workflow environment, it addresses the analysis needs of life scientists working with large-scale data. Initially, *SPR* focused mainly on the analysis of next-generation sequencing (NGS) data. In recent upgrades, I have extended its usability to a powerful data analysis workflow and report generation environment suitable for nearly any data type, including small- and large-scale omics data. Although the new version of *SPR* is versatile for analyzing data from a broad range of scientific areas, workflow management in *SPR* requires basic command-line and R knowledge. Thus, the addition of a GUI will simplify its usage for users with limited or no programming experience.

Several R packages provide GUIs implemented within a web framework, including Shiny (Chang et al., 2021), Dash (dash.plotly.com), and more recently, Voila (voila.readthedocs.io). The most widely used web framework for R is Shiny (Jia et al., 2021). Examples of R applications, including a Shiny-based GUI are START, DEBrowser and POMAShiny (Nelson et al., 2017; Kucukural et al., 2019; Castellano-Escuder et al., 2021). Although powerful, native Shiny has various apparent drawbacks. First, Shiny does not easily scale to larger applications. As a result, they can quickly become complex and difficult to maintain. Second, for sensitive data, Shiny lacks native functionalities for administering authentication and security. Third, standard Shiny apps hide the analysis source code from users, hampering the required transparency to understand the details of an analysis and reproduce the results. Fourth, modifications to the source code of an application, no matter how trivial they are, require redeployment of the entire Shiny app, which can be disruptive and time-consuming.

To address the need of extending *SPR* with a versatile GUI and overcome shortcomings observed in other Shiny-based applications, I have developed the systemPipeShiny (*SPS*) framework. *SPS* is a user-friendly web application for managing data analysis workflows, visualizing downstream results, and processing figures to generate publication quality graphs within the same application. *SPS* is a one-stop solution for workflow design and execution, and visualization (Fig. 3.1). I accomplished this by developing a series of Shiny extensions that enhance both user experience and maintainability of *SPS*.

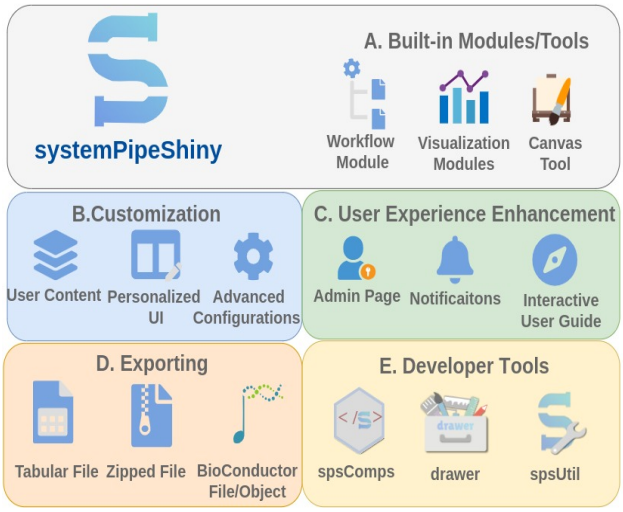

Figure 3.1: SPS Design Overview. (A). Module categories in *SPS*, including workflow management, data visualization and image processing. (B) Customization features allowing users to personalize content, appearance, and other components. (C) Extensions to Shiny to improve user experience and software management. (D) Functionalities to export data. (E) Add-on packages published on CRAN providing visualization tools and supporting future *SPS* development.

3.3 Materials and Methods

3.3.1 Overview

I implemented *SPS* mainly in R with JavaScript and SQL extensions (Fig. 3.10). The software is available as a free open-source package on Bioconductor (Huber et al., 2015*b*). Throughout the environment, I have applied modular design principles. This comprises *Functional Modules* accessible via the user interface (UI) and the use of the *Shiny Module* system on the backend. This modularity makes the software highly customizable and extendable. The following describes the three main *Functional Modules* of *SPS* including workflow, data visualization, and image processing modules. Users can access the individual *Functional Modules* via dedicated dashboard tabs (Fig. 3.5B). For clarity, I

structured the content of the individual dashboard tabs consistently, including expandable help sections, images and links to other resources, followed by an interactive step-by-step guide.

3.3.2 Workflow Module

The *Workflow Module* extends *SPR*'s command-line interface with a GUI. It enables users to design, run and visualize data analysis workflows interactively. Step-by-step instructions guide users through six configuration steps where each has its own sub-page (3.2). Throughout the process, the environment displays the analysis options and execution status of each step with illustrations and color codes, respectively. The following outlines each step.

1. Setup. Users have three options to initialize workflows. First, they can select pre-configured NGS workflow templates, including variant detection (Var-Seq), transcriptional profiling (RNA-Seq), ribosome profiling (Ribo-Seq), and chromatin immunoprecipitation (ChIP-Seq). Second, they can upload custom workflow templates. Third, they can generate workflows from scratch using *SPS*. After making a selection, *SPS* creates a workflow project environment, where it stores all subsequent data and result files.

2. Metadata. Users can specify information about samples, experimental design and file locations in a *targets* table (Fig. 3.2.2) that corresponds to the sample and metadata slots of Bioconductor's widely used *SummarizedExperiments* object. *SPS* allows to upload the *targets* table from a file, or to create it interactively in an online table editor. *SPS* validates the format of the targets table and points to potential errors.

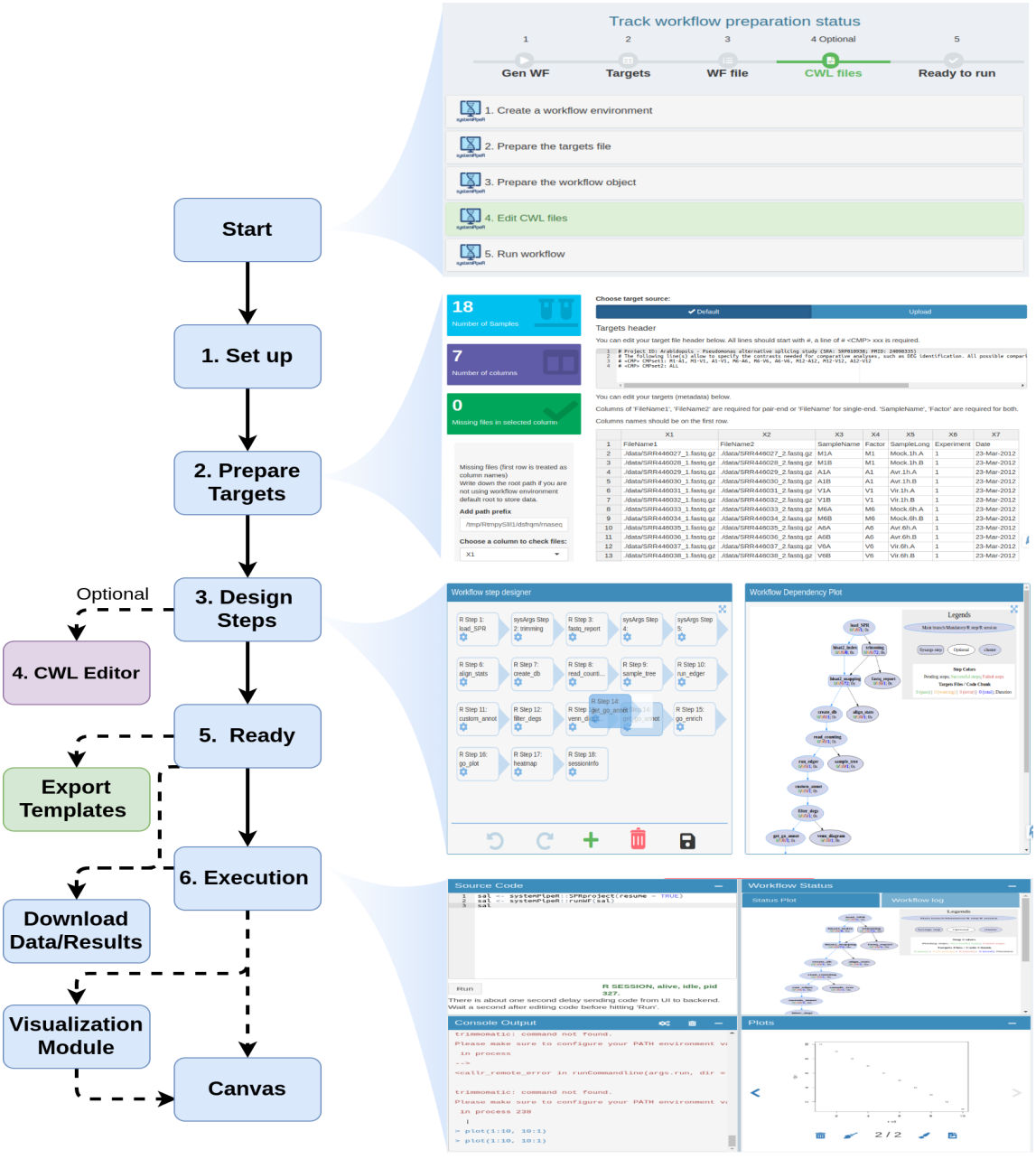

Figure 3.2: Workflow Module. It comprises six steps for configuring, running, and monitoring workflows. Steps 1-2 are for choosing workflows, and configuring samples and metadata. Step 3 provides access to an interactive viewing and editing tool for workflows where topologies can be changed by intuitive drag-and-drop actions. Step 4 is optional, allowing expert users to work with CWL configuration files directly. Step 5 is for initializing workflow runs or porting them to other systems. Step 6 provides an array of workflow monitoring functionalities. Solid and dashed lines indicate mandatory and optional actions, respectively, as well as connectivity to other modules.

3. Topology. A workflow design tool is available for viewing and modifying workflows interactively (Fig. 3.2.3). This functionality allows users to add, change and delete steps, or change their order. New steps can be defined by completing a configuration form. New workflow steps can contain pure R code or steps invoking command-line software. For pure R steps, *SPS* provides a code editor, and for command-line steps users can choose from a collection of pre-configured CWL templates. Interactive modifications to existing workflow steps are also supported. Common modifications include changes to names of steps or command-line tools, dependencies, or sample information. To test compatibility, *SPS* offers a dry-run option where problems are indicated visually. All topological changes to a workflow are displayed on the same page in a high-level workflow graph in real-time.

4. CWL. This configuration step is for expert users to modify or create new CWL templates and test their functionality (Fig. 3.2.4). A CWL template comprises metadata, input and configuration files. The page has an upload and edit utility for these files. A dry-run option is available to test and render the corresponding software's command-line calls.

5. Initialize Run or Download. At this step, users can initialize workflow runs directly from *SPS*'s UI (Fig. 3.2.5). Alternatively, one can download fully pre-configured workflows to run them later, or port them to other systems, such as local computers, high-performance clusters, or cloud services (Fig. 1D), and then run them there.

6. Workflow Monitoring. When *SPS* executes a workflow (Fig. 3.2.6), the UI monitors the run by displaying detailed information including R and command-line code, locations of output files such as tables and graphics. In addition, *SPS* automatically collects run logs and updates them in short intervals.

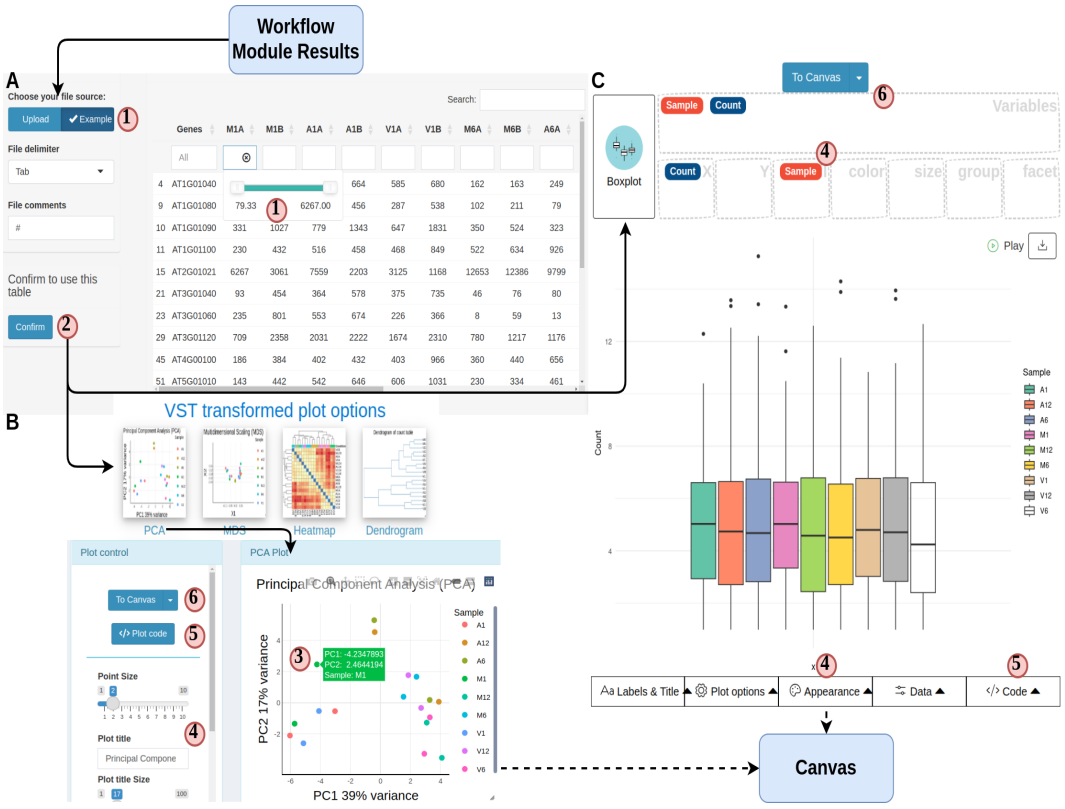

Figure 3.3: Visualization Module of *SPS*. (A) Display of tabular data. (B) Plot window with RNA-seq data as an example. (C) Quick *ggplot* interface. Step-wise illustration of a common use case: (1) Workflow results are uploaded for visualization. If needed, common data pre-processing and filtering routines can be applied. (2) Upon confirming the dataset, different plotting options are available. (3) Most plots are interactive, *e.g.* mousing hovering displays values. (4) Numerous customization options are provided, such as styling of graphical components. (5) The code for generating a plot can be accessed and saved for reproducibility. (6) Any plot instance can be sent to the Canvas workbench or saved to a file.

3.3.3 Data Visualization Module

The *Data Visualization Module* is composed of general plotting utilities. As under the *Workflow Module*, this module guides users through three major steps. First, a data preparation step uploads, displays and filters the data (Fig. 3.3A). Second, users can visualize their data with multiple types of plots. Examples are given in Fig. 3.6.3-4 and

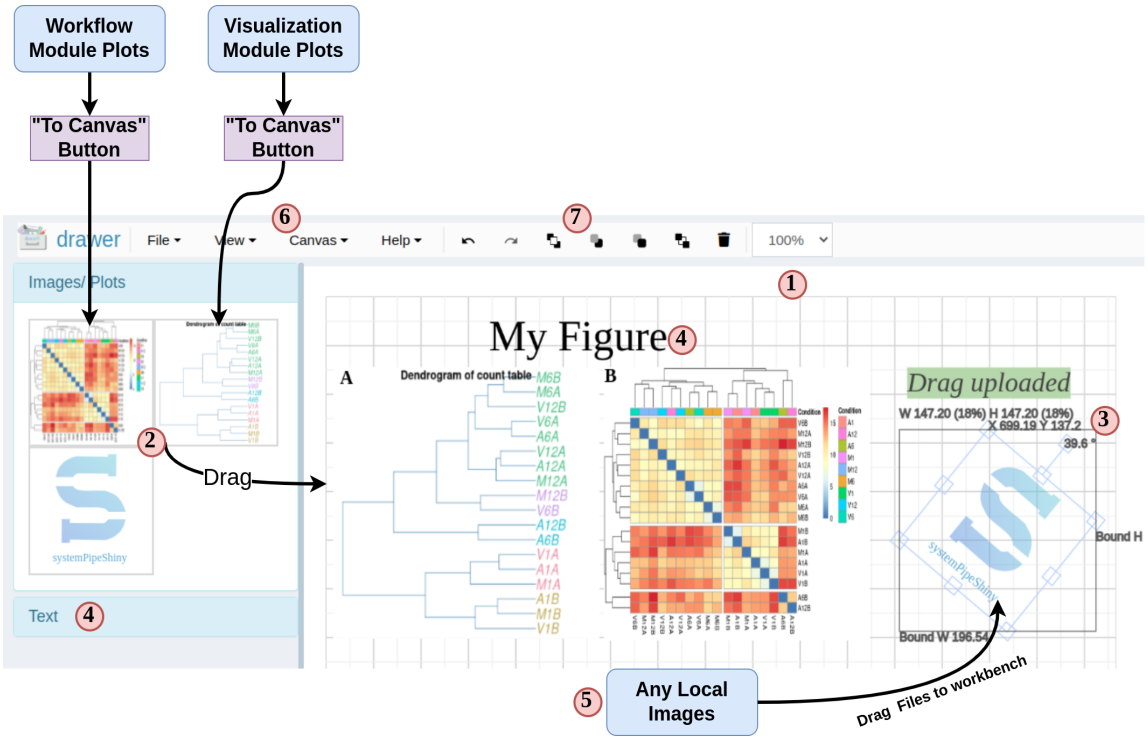

Figure 3.4: Canvas Module of *SPS*. The Canvas Module has two main parts: (1) the workbench and (2) the image browser. (3) After dragging one or more plot thumbnails to the Canvas workbench, users can rearrange and edit them as needed. (4) Image annotations can be added via a text panel. (5) Images can also be imported from local files. (6) Several menu options are available to change general settings, such as export options, help, and keyboard shortcuts. (7) Extended image editing options are provided, such as layer opacity.

Fig. 3.7.3. Third, the resulting plots can be fully customized in an interactive refinement window (Fig. 3.3B-C). If needed, users can choose from a wide range of data transformation utilities (*e.g.* normalization) to pre-process their data prior to plotting properly. Several data validation steps are included to identify a suitable data transformation solution and avoid the usage of incompatible or incorrectly formatted data. These data validations are discussed in more detail under the 'Use Case' section.

3.3.4 Image Processing Module (Canvas)

The image processing module, or Canvas, is an interactive workbench for arranging and post-processing one or many plots generated by a workflow in multi-component plots (Fig. 3.4). There are numerous steps in *SPS* where users can send their plots to the Canvas (Fig. 3.3B.6 and 3.3C.6). The resulting snapshots are visible in an image browser from where they can be dragged to the Canvas workbench. After this, users can organize their plots in a single view and edit them. Supported image processing operations include resizing, rotations, and various other editing routines (Fig 3.4.3). Keyboard shortcuts and grid lines are supported to accelerate these operations and to make them more precise, respectively (Fig 3.4.3 and 3.4.6). Custom images can also be imported and edited via an upload functionality (Fig. 3.4.5). Supported image formats include but are not limited to *png*, *jpg* and *svg*.

3.3.5 Advanced Features

I have developed several administrative features for controlling the general behavior of the environment. Developing these functionalities by this project was necessary because they are not readily available in Shiny. The modular design of *SPS* makes developing such custom features relatively straightforward (Table. 3.1).

Table 3.1: Important Configuration Options of SPS. The table lists the most important configuration options of SPS installs. They control the general behavior and visual design of a deployment.

Option	Description	Default Value	Other Value
mode	running mode	"local"	"server"
title	app title	"systemPipeShiny"	any string
title_logo	app logo to display on browser tab	"img/sps_small.png"	any path
warning_toast	show security warnings?	TRUE	FALSE
login_screen	add login screen?	TRUE	FALSE
login_theme	login screen theme	"random"	see manuals
use_crayon	colorful console message?	TRUE	FALSE
verbose	more details for SPS functions?	FALSE	TRUE
admin_page	enable admin page?	FALSE	TRUE
admin_url	admin_page query URL	"admin",	any string
warning_toast	for internal tests only	TRUE	FALSE
module_wf	load workflow module?	TRUE	FALSE
module_rnaseq	load RNAseq module?	TRUE	FALSE
module_ggplot	load quick ggplot module?	TRUE	FALSE
tab_welcome	load welcome tab?	TRUE	FALSE
tab_vs_main	load custom visualization main tab?	TRUE	FALSE
tab_canvas	load Canvas tab?	TRUE	FALSE
tab_about	load about tab?	TRUE	FALSE
note_url	SPS notification remote URL	see manuals	any URL
is_demo	useful if deploy the app as a demo	FALSE	TRUE
welcome_guide	enable the welcome guide?	TRUE	FALSE
app_path	hidden, automatically added	N.A.	N.A.

Install and Validation

During the installation of *SPS*, several automated validation steps examine the integrity of the deployment (Fig. 3.9). This includes the availability of dependencies, proper parameter settings, and functionality of the application database. Problems are

saved via a dual-logging mechanism and reported to the user along with instructions on how to correct them, or reset to the default settings. Users can modulate the verbosity of the install messages. Traditional Shiny applications lack these advanced deployment features.

Administration

Server-based deployments of Shiny applications allow many users to access the same web instance. Standard deployments of the free Shiny software lack built-in authentication support. The commercial version supports this feature, but is cost prohibitive for academic users. Advanced security features for protecting sensitive data are also not available in the free version. I have developed these essential features in *SPS*. After activating them in an administrator console (Fig. 3.8), users are required to log in prior to using a protected web instance of *SPS* (Fig 3.8.C-D). The usernames and passwords are encrypted, and stored in a secure database. An analytics tool (Fig 3.8.A) allows administrators to access and plot basic statistics about the deployment, such as usage of CPU, RAM, and storage.

Additional Developer Tools

When the content of a Shiny application changes, a running web instance needs to be restarted in order to make those changes life for users. To assure timely deployment of ongoing development or broadcasting documentation changes, a human-readable *YAML* file can be used. Any modification made to this file immediately takes effect when re-visiting *SPS* without redeployment on the server-side.

For future development, the environment is structured into four packages, including the *SPS* core package, and three development packages hosted on CRAN: *spsComps*, *drawer* and *spsUtil*. *spsComps* is a collection of UI components and server enhancement utilities for Shiny development. The *drawer* package is the development platform for *SPS*'s Canvas module. It is a front-end image editing tool that can be used, apart from Shiny, in static webpage rendering environments, such as *Rmarkdown*, *Bookdown*, or *Blogdown*. General R utility functions are provided by *spsUtil*.

3.4 Results and Discussion

3.4.1 Novel Features

SPS provides a user-friendly GUI for designing and running data analysis workflows integrated with rich visualization and reporting functionalities (Fig. 3.1A). Users can control all analysis steps with intuitive mouse actions without requiring knowledge of command-line syntax (Fig. 3.5B). An important design feature is a tight integration with *systemPipeR* as a generic workflow environment. This architecture enhances *SPS*'s usability for a wide range of data analysis areas. Users can choose from pre-configured workflows, edit them, upload or design new ones. Most other web environments in this area are typically limited to processing specific data with predefined methods, and are not easily extendable to other data types or analysis tools. They lack many important functionalities that are essential for processing complex data by providing efficient functionalities for workflow automation, reporting, visualization and re-usability. In *SPS*, automation saves time and enhances reproducibility by running complex multi-step analyses from start to finish with-

out human intervention. Reporting functionalities generate publication-quality documents containing detailed and reproducible analysis results. Visualization is important for interpretation to translate analysis results into novel findings. Re-usability provides flexibility to utilize the environment for many different data analysis tasks. While addressing these user-facing requirements, *SPS* also provides a series of important features for developers. This includes robust APIs for most modules, add-on packages for developers, and debugging functionalities. Customization of *SPS* deployments is possible via an administrative console. Combined, these advanced features make the environment highly expandable and maintainable.

3.4.2 Reproducibility

Reproducible data analysis is critical for all scientific areas. I have implemented in *SPS* a series of mechanisms to address this need. The most important data reproducibility features are outlined here. First, the usage of a workflow environment, such as *systemPipeR* combined with CWL, supports standardization and automation of complex data analysis routines. Second, *SPS*'s workflow module allows to download fully configured workflows as an *SPR* data analysis project (see 5 in Fig. 3.2.5). It contains all relevant analysis code, and input and output data needed to reproduce a data analysis workflow. If needed, users can exclude protected data by restricting the download to analysis code. Third, when generating plots interactively, *SPS* updates the corresponding R code in the background in real-time, and users can reveal and download the code upon request. Fourth, similar export and import options exist for composite plots generated with the Canvas module. Here users can export composite plots to files in different image formats (Fig. 3.4.6), and if needed reimport

them back into the main workbench (Fig. 3.4.5). Fifth, exchange routines of complex data sets with other software are supported by using in *SPS* widely used data structures of the Bioconductor ecosystem, such as the *Summarized Experiment* class (Morgan et al., 2021).

3.4.3 Use Case Examples

Automated Workflow Runs

A typical analysis project in *SPS* starts with initializing a workflow project and validating its input data and parameters. These steps are completed interactively under Steps 1-5 of the *Workflow Module* (Fig. 3.2). Next, an initialized and validated workflow can be run from start to finish fully automatically. After successful completion, the results can be accessed in a project's result directory or downloaded from an online *SPS* instance including a detailed workflow report, as well as raw and processed result and image files. The *Visualization* and *Canvas Modules* can be used to visualize the results interactively (see below). Warnings and errors are reported on the individual workflow pages and written to downloadable log files. If an error occurred, users can restart the workflow at the affected step after addressing the reported problem without repeating successfully completed steps. A major advantage of running workflows in a Shiny app is the reactivity of the environment. With this analysis results are not just static snapshots of predefined parameter combinations. Rather they remain modifiable where users can continue to adjust parameters as needed and update the corresponding tables and plots in real-time without rerunning the upstream analysis steps or an entire workflow.

Step-Wise Workflow Runs

If preferred users can run each step of a workflow one by one. Online dialog instructions provided under each step guide users through this process. The initialization process of a workflow project is the same as under the automated option above. This step-by-step mode provides the most control when running workflows for the first time or optimizing parameters.

Visualizing Workflow Results

The *Visualization Module* allows users to plot intermediate or final results with a wide range data visualization routines. Some are mainly useful for certain assay technologies, while others are applicable to a wide range of data types. For instance, quantitative assay data from various omics experiments, can be visualized by principal components analysis (PCA), t-distributed stochastic neighbor embedding (t-SNE), heatmaps, volcano plots, and more (Fig. 3.6.3-4). Both the underlying data and the resulting plots can be downloaded in widely used community data structures and file formats. Moreover, the code for data processing and generating the plots can be inspected online. If additional plotting and data visualization utilities are needed, users can take advantage of the *quick ggplot* functionality included in *SPS*. It adapts *ggplot2*, one of the most popular graphic systems in R (Wickham, 2016), that helps users to generate widely applicable plot types. For example, this includes generic scatter plots, bar plots, pie charts, and many more (Fig. 3.7). The input for *quick ggplot* is also generic. Most widely used tabular file types are supported, such as *csv* and *tsv*. An intuitive drag-and-drop interface simplifies common mapping routines

of data, symbols, colors, font and line types (Fig. 3.7.2). The plotting type is automatically chosen for the user based on the input data provided, or users can manually choose one (Fig. 3.7.3). Lastly, users can download the plots in different image formats as well as the underlying code required to fully reproduce the plots within or outside of *SPS* (Fig. 3.7.4).

Composite Plots

After generating individual plots with the visualization module, a common need is to arrange multiple plots in a single view. This can be important for comparing similar plots or assembling a series of plots to composite images for slide presentations or publication purposes. The Canvas Module is designed to accomplish this in *SPS* in an efficient and user-friendly manner (Fig. 3.4). If any of the individual plots requires revision, one can update them as needed by revisiting the corresponding workflow step (Fig. 3.5A).

Domain Specific Example: RNA-Seq Data Analysis

This section illustrates the analysis of domain-specific data with *SPS*, where RNA-Seq has been chosen as a workflow example. RNA-Seq is an NGS-based omics technology for profiling mRNA abundance levels in cells, tissues, or entire organisms on a genome-wide level. It is widely used in many areas of biology, genetics, and biomedical sciences (Barrett et al., 2013). Due to the complexity and inherent noise of RNA-Seq data, their analysis can be challenging for non-experts. In *SPS*, I have made the analysis of RNA-Seq data straightforward. Only the most important workflow steps are highlighted in this use case example. A typical RNA-seq analysis workflow includes multiple QC and analysis steps. This includes the alignment of NGS sequences (reads), read counting, analysis of

differentially expressed genes (DEGs), and functional enrichment analysis (FEA). The first two steps involve relatively large data files of NGS reads and alignments, whereas the resulting read count table is much smaller in size, often by more than two magnitudes. If a read count table is available, then users can use it directly and skip the alignment and read counting steps. The alignments can be generated with an RNA-Seq aligner specified by the user in the workflow and installed on the system running *SPS*. In this step the user has to provide the sample names and file paths of the RNA-Seq data, the corresponding reference genome sequence and annotation. Subsequently, the reads overlapping with gene regions are quantified with a read counter. After normalizing the RNA-Seq read counts, DEGs can be identified with statistical methods such as *DESeq2* or *edgeR*. To functionally interpret the results, the DEG lists are used for FEA. When starting from a pre-generated read count table, *SPS* automatically validates the required metadata and raw counts and then continues with the subsequent steps including normalization and DEG analysis (Fig. 3.6.1-2). After completing the basic steps of an RNA-Seq analysis, users can visualize the results with a suite of plotting tools useful for RNA-Seq analysis (Fig. 3.6) including PCA, t-SNE, volcano plots of DEG data, heatmaps of hierarchical clustering results, and more (Fig. 3.6.3-4). All result files including tables and plots can be exported to the project directory along with the corresponding analysis code.

Working With Other Community or Custom Workflows

Users can work with custom data analysis workflows designed by themselves or downloaded from community repositories. For example, if one has used has *SPR/SPS* data analysis projects before, workflow templates are included. These templates can be shared

with collaborators and the community. Following some simple instructions in the Workflow module, one can easily load custom workflows back to *SPS* (Fig. 3.2.1, Fig. 3.11). For experienced users, setting up an "Empty" workflow allows them to fully customize the analysis workflow from scratch (Fig. 3.11).

3.5 Conclusions

SPS allows to design and execute data analysis workflows within an intuitive-to-use GUI. As an extension of the generic SPR workflow management tool, SPS is highly extensible and flexible for analyzing a wide range of data types with existing or novel methods. Reproducibility of the data analysis results is enhanced by the usage of automated workflow concepts that are based on community standards combined with extensive reporting functionalities. Rich data visualization functionalities are integrated to help users interpret results and extract knowledge.

3.6 Additional Figures

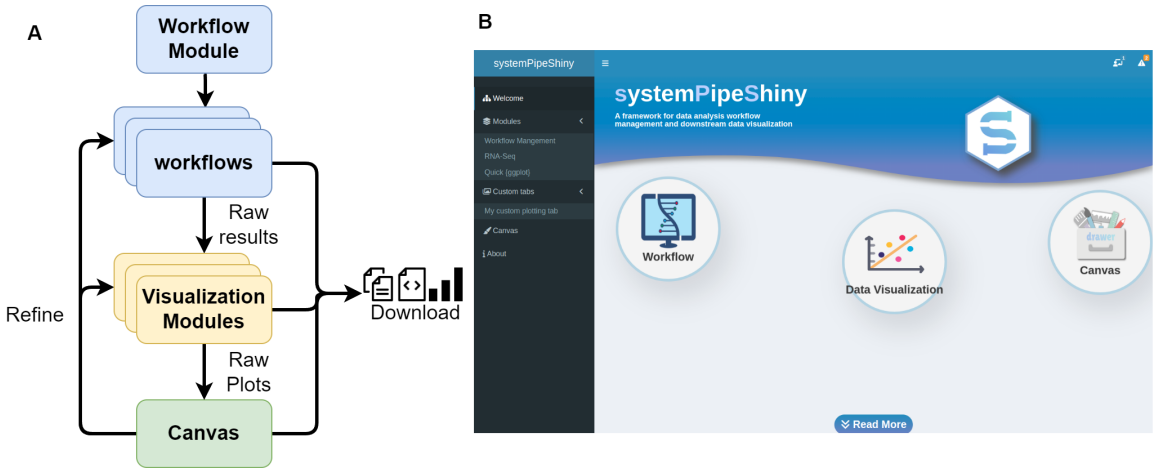

Figure 3.5: SPS Usage Paradigm. (A) Data analysis steps in SPS. A typical data analysis starts with the Workflow Module where users can design, edit and run workflows. When using a pre-configured workflow, the design and editing steps are optional. The Visualization and Image Processing Modules can be used to plot the results and to assemble composite images, respectively. If needed, users can revise the results by repeating the previous steps. Source code, image and tabular data are downloadable as individual files or batch archives. (B) Most functionalities can be accessed by selecting the corresponding icons on SPS's welcome page.

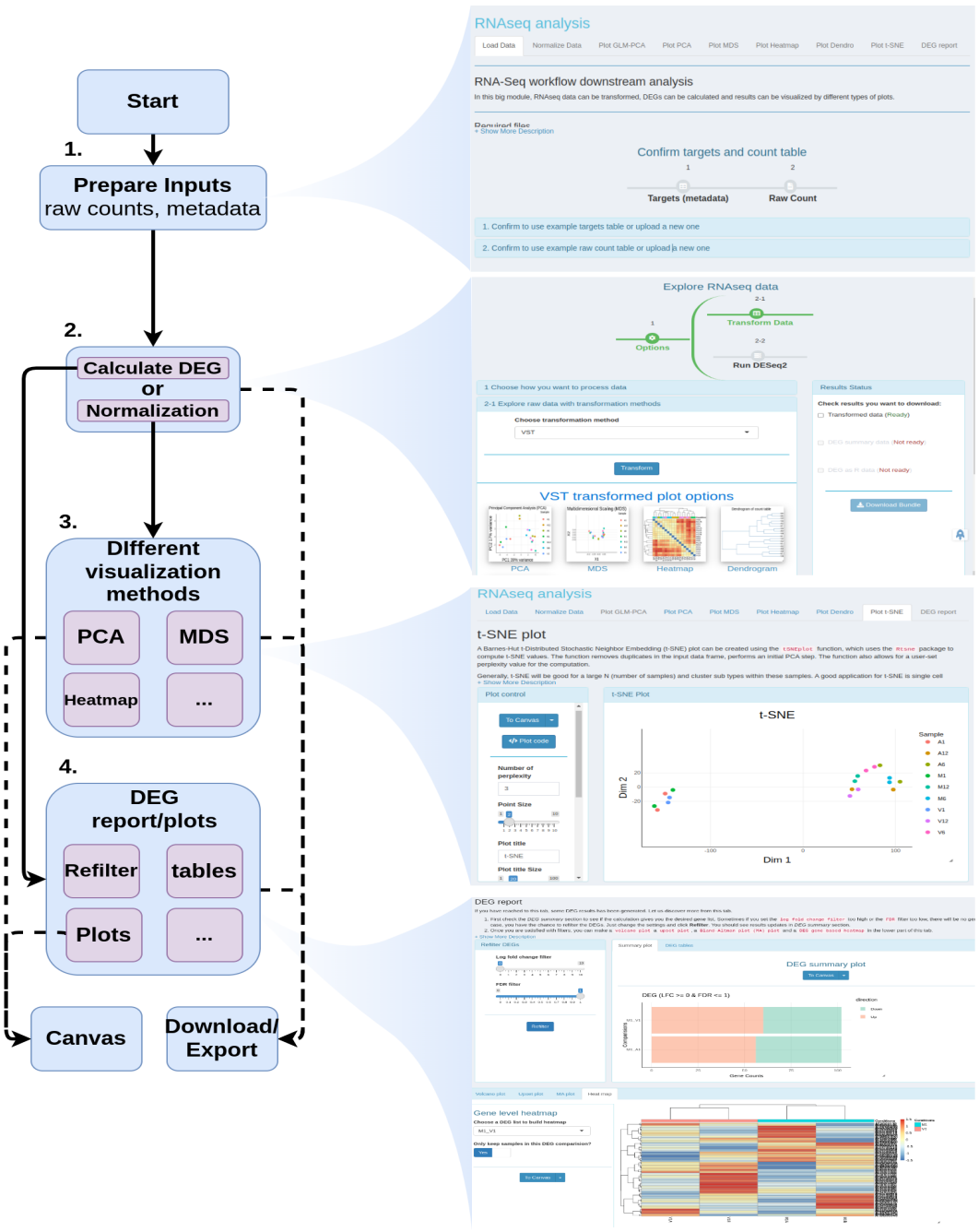

Figure 3.6: RNA-Seq Utilities. (1) When starting from a read count table, the corresponding tabular file and a sample annotation (targets) information need to be provided as input. (2) After validating the input, the raw read counts are normalized, and (3) differentially expressed genes (DEGs) identified. (4) The results can be visualized with different plotting routines. Subsequently, multiple plots can be post-processed, and a detailed analysis report generated. Dashed lines indicate optional actions.

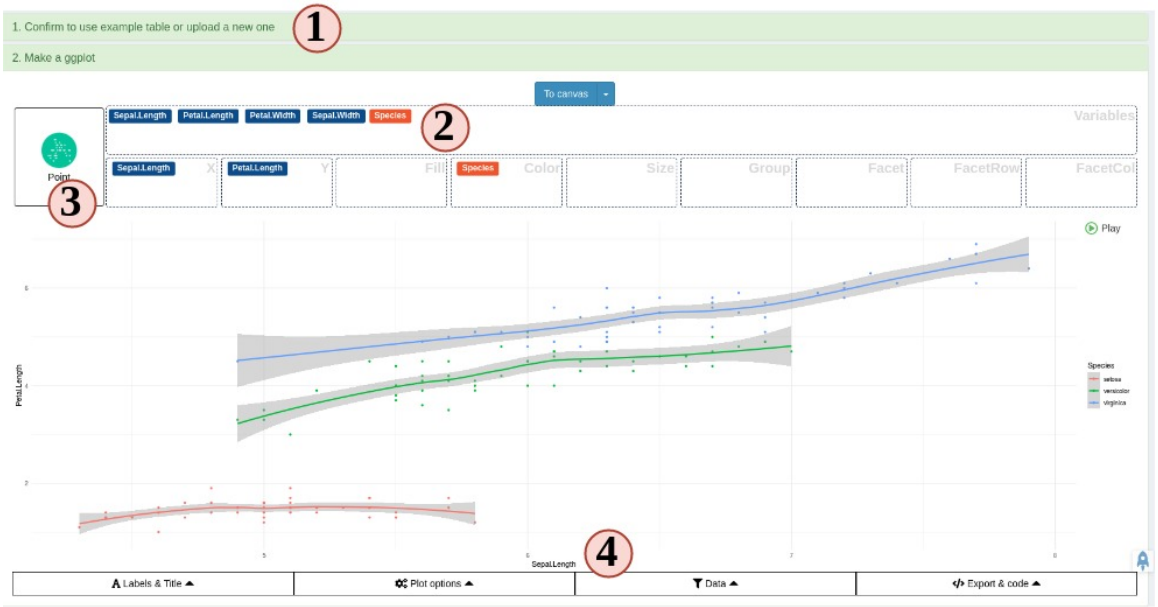

Figure 3.7: Quick ggplot Utility. (1) Uploaded tabular data can be displayed, filtered, and validated. (2) Table columns to include in a plot can be selected under the first two rows. (3) The plot type can be chosen from a large collection. (4) Plotting styles can be revised at the bottom, including legends, shapes, symbols, font size, line width, and color palettes. The "To canvas" button on the top sends the current plot to SPS' Canvas module.

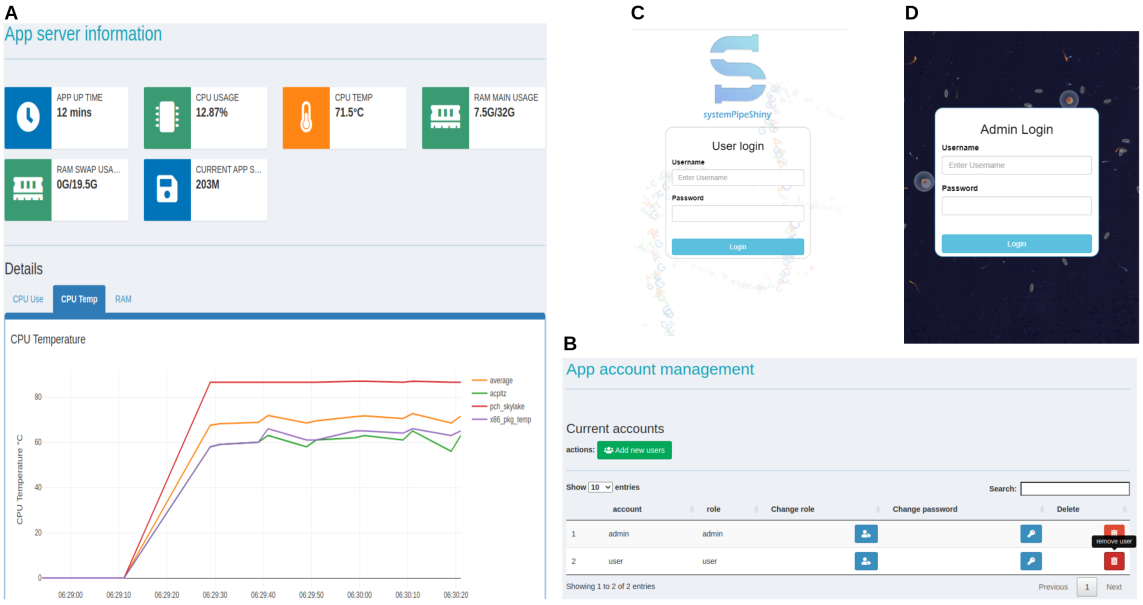

Figure 3.8: Administration & Security Management. (A) Activated analytics features can be inspected on SPS' administrator page. (B) If needed, user authentication can be activated by the administrator of an SPS deployment. When turned on, users are required to provide their access credentials (username and password) to access a protected SPS instance. (D) Accessing these features requires administrative permissions.

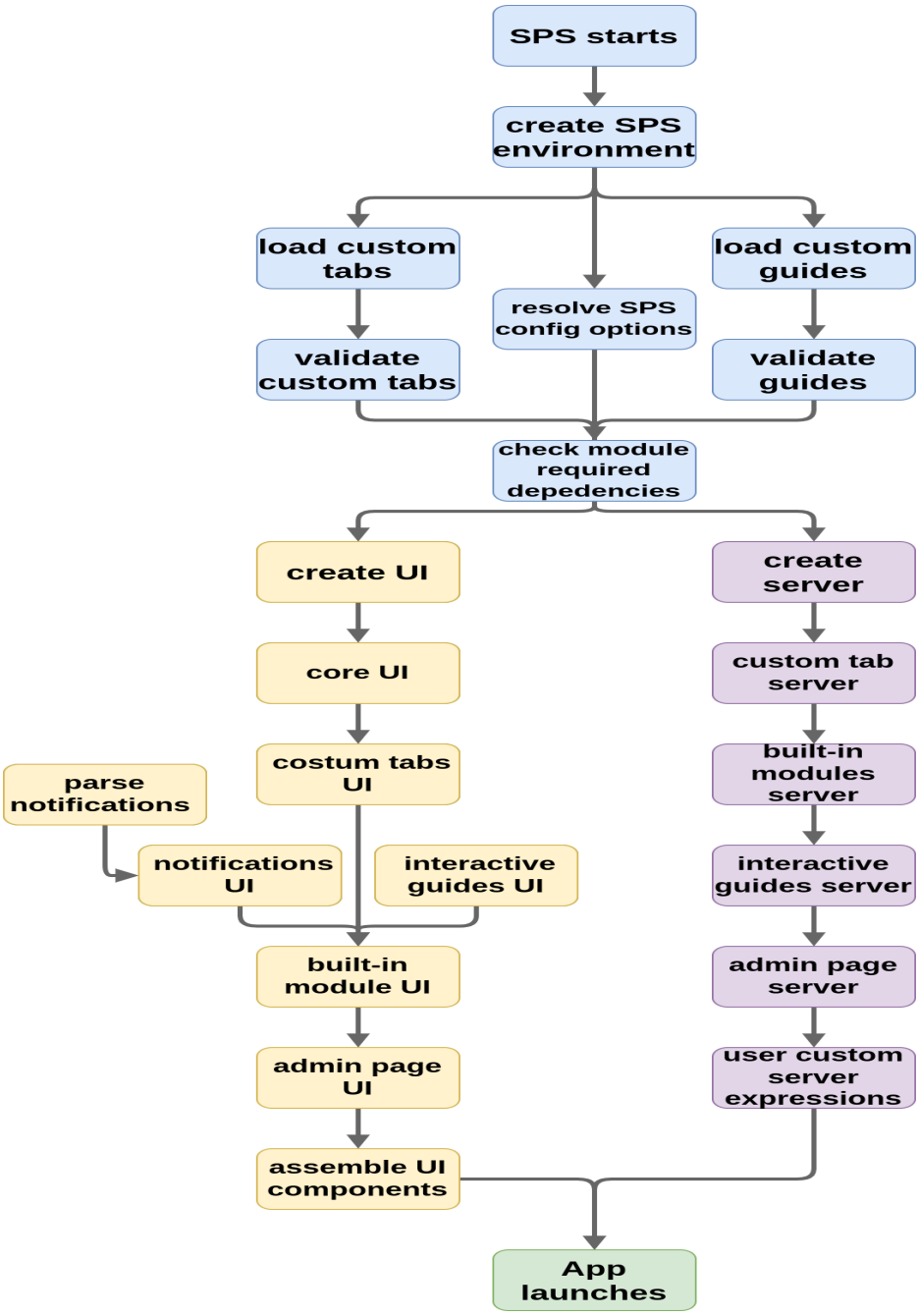

Figure 3.9: Flowchart of SPS deployment. During an SPS deployment, a hidden environment is created that stores configuration information, data, and other components. Default or custom configuration options are validated. After this, the process split into UI- and server-specific steps. Next, the different components are assembled into a ready-to-be-launched Shiny instance.

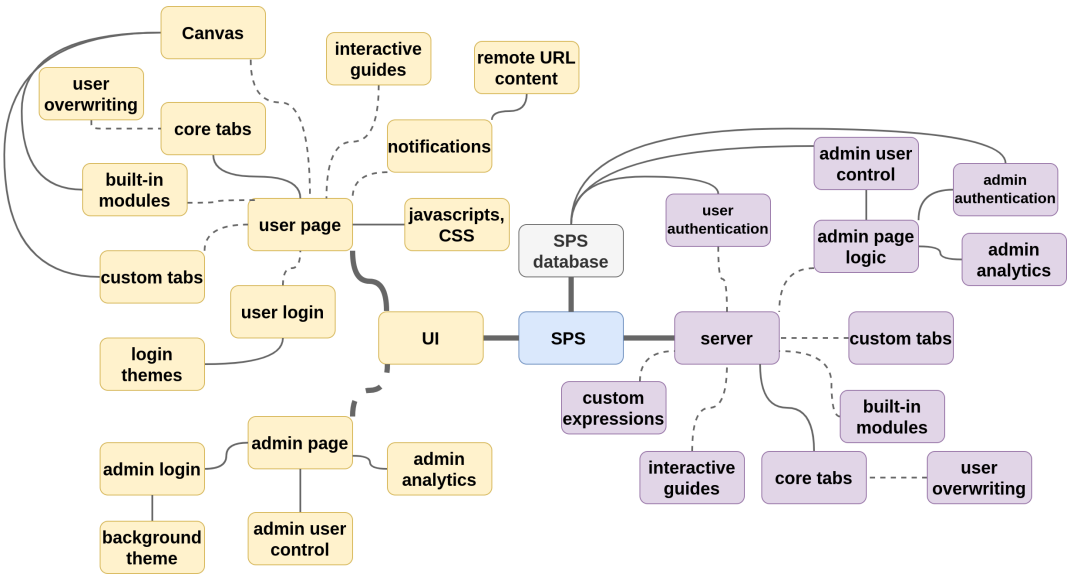

Figure 3.10: SPS Architecture. The connectivity of the SPS components is illustrated in a network diagram. Components that always need to be loaded are indicated by solid lines. Dashed lines indicate optional components that can be disabled.

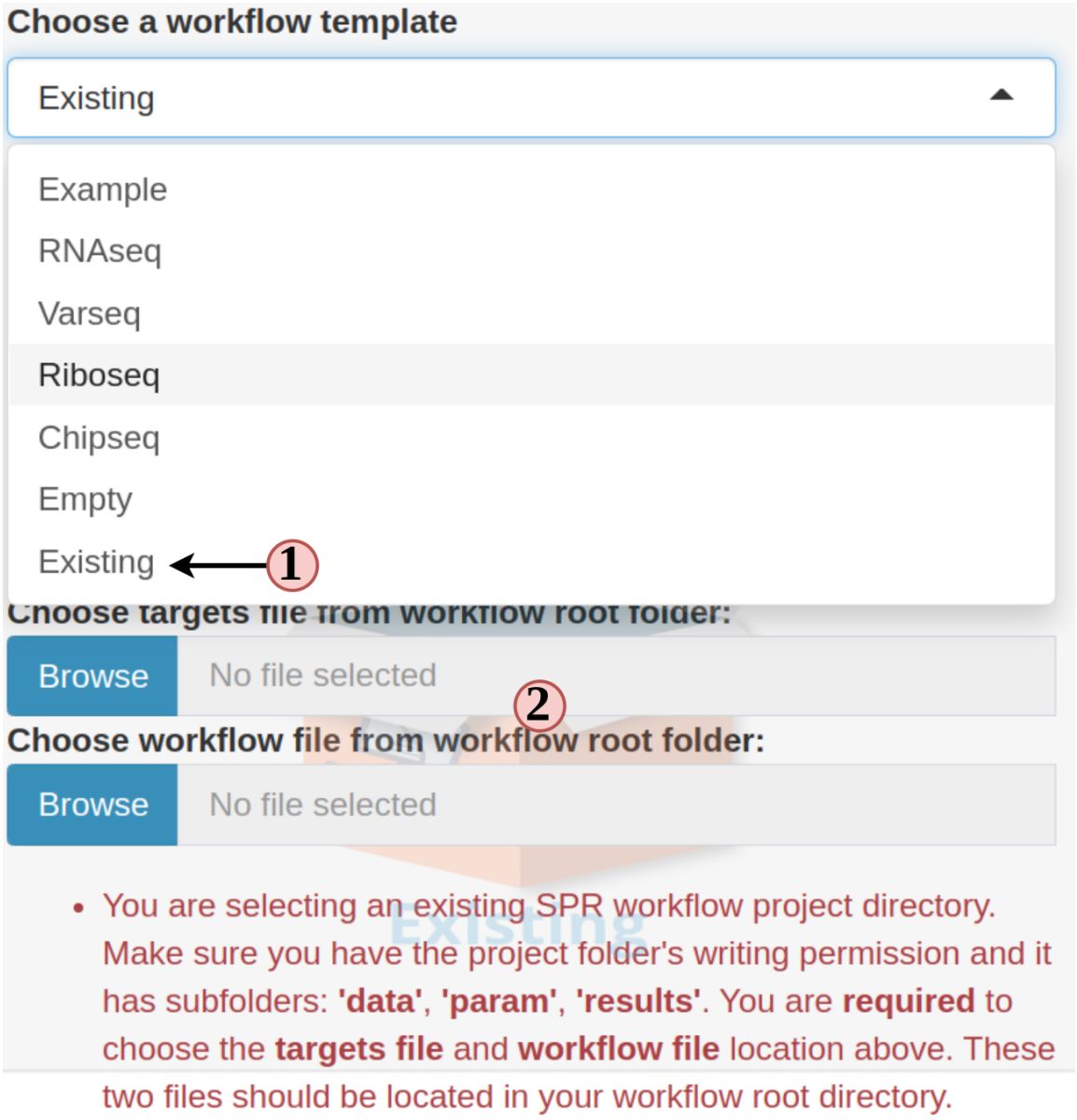

Figure 3.11: Run custom workflows. (1) The existing option on the workflow setup page allows users to load custom workflows. (2) Path of workflow templates and targets are required for custom workflows.

Chapter 4

Large-Scale Prediction of Natural Product Alternatives for Known Drugs

4.1 Abstract

Natural products (NPs) have been recognized for their potential to elicit biological activities upon consumption. This has led to their extensive use in medicinal therapy development, new drug exploration, and dietary supplement production. Despite their widespread applications, the majority of NPs still require further research to understand their properties and effects fully. To bridge this gap, mystudy utilized both supervised and unsupervised machine learning techniques to establish an association linkage between approved drugs and large-scale NPs. My screening process analyzed 433,420 NPs against

3,590 approved drugs and identified 8,480 high-confidence bioactive NP (BNP) alternatives across 2,228 drugs. These drugs are classified as BNP-substitutable drugs (BSDs). For the remaining drugs that failed to detect BNPs, I classify them as BNP-unsubstitutable drugs (BUDs). I explored the chemical and structural characteristics of BUDs and found out amine groups played an important role in distinguishing BUDs, BSDs, and NPs. It then led to the construction of high-accurate prediction models to classify the compound type. Further annotations of the BNPs with functional, safety, source, and dietary supplement product information were also carried out. To demonstrate the use cases of my selected BNPs, I focused on their application in the longevity and anticancer categories. Through alternative sources, I illustrated how my findings could be helpful in identifying potential longevity-promoting and anticancer organisms, diets, and dietary supplements. I firmly believe that the information provided in this research can be incredibly beneficial in terms of providing valuable health recommendations for longevity and aiding in the discovery of vital anticancer drugs. Overall, my study yields valuable information on the potential for NPs to elicit biological effects, paving the way for their application in medicinal therapy development and new drug discovery. The annotated information derived from this study stands as a valuable resource for future research in this area.

4.2 Introdcution

Natural products (NPs) are compounds produced by living organisms. When some of NPs exhibit specific biological activities, they are classified as bioactive NPs (BNPs). There are various applications of BNPs (Sorokina and Steinbeck 2020). One direct applica-

tion is to apply NPs as medicine through topical or oral use. Another primary application of NPs is new drug discovery. NPs or NP derivatives made up more than 80% of all drugs approved by the U.S. Food and Drug Administration (FDA) before 2000 (Sneader 1997) and currently make up more than 50% of the approved drugs on the market (Kingston 2011). However, using NPs as putative therapies is not limited to drug screening. NPs may have biological effects if consumed from dietary sources and researching the value of daily consumption of NPs in diet is of great interest (Atanasov et al. 2015).

Using NPs for medicinal purposes has a long history, starting with Traditional Chinese Medicine (TCM) and Indian Ayurveda (Patwardhan et al. 2005). Even though there is limited scientific evidence supporting these practices, there is a large population of people who use them as dietary guides to decipher the sources of NPs in order to obtain a specific biological effect or treat various ailments(Normile 2003). For example, NPs in garlic reduce the occurrence of cancer (Lau et al. 1990; Almatroodi et al. 2019; Dorant et al. 1993) while green tea not only decreases cancer prevalence but also (Yu et al. 1995; Zhong et al. 2001) prevents a variety of health problems including stroke, promotes oral health and reduces inflammation (Narotzki et al. 2012; Kokubo et al. 2013; Chan et al. 1997). NPs discovered in many of these bioactive organisms can be purified for their therapeutic benefit. Camptothecin, artemisinin, and ephedrine, for example, are naturally occurring in *Camptotheca acuminata*, *Artemisia annua*, and *Ephedra sinica* respectively and can be applied in treating anticancer, malaria, and hypotension (Efferth et al. 2007).

In addition, instead of directly taking NPs as medicines, NPs have come into sight with another alternative form of dietary supplements in recent decades. A study in 2012

suggests that more than half of the U.S. population uses some dietary supplement (Marik and Flemmer 2012). The trend of consuming dietary supplements is growing. The global dietary supplement market is expected to nearly double from 2016 to 2022 (Lam et al. 2022), whereas the global GDP only grew 12% from 2016-2021 (The World Bank 2022).

There are both advantages and disadvantages to purifying NPs for dietary supplementation. While the isolation process removes compounds that may be important key ingredients to promote overall health, purification may enhance the efficacy of the compounds by increasing the concentration that can be consumed in a single dose (Ronis et al. 2018). Isolation also enables the study of the effects of specific NPs on the prevention and treatment of various diseases (Harvey 2008; Kingston 2011; Shen 2015) (Atanasov et al. 2015). For example, Galanthamine, berberine, and mangiferin supplementation may be beneficial for treating Alzheimer's and Parkinson's disease (Bui and Nguyen 2017; Marco and Carreiras 2006; Ji and Shen 2011; Feng et al. 2019). Further, extraction of multiple NPs from Chinese medicinal plants is used to treat and prevent osteoporosis, a long-term progressive bone loss disorder (Che et al. 2016).

Considering the existing value of NP drugs and the urgent need for NPs in the food supplement market, identifying drug alternative NPs could be a good direction. However, even though numerous studies have endeavored to research NPs, such as evaluating the value of NPs (Harvey 2008; Kingston 2011; Shen 2015), collecting available NP resources as databases (Zeng et al. 2019; Sorokina and Steinbeck 2020; Gaulton et al. 2017; Banerjee et al. 2014; The Metabolomics Innovation Centre n.d.), mapping one or limited BNPs as alternatives of drug alternatives (Corrêa et al. 2020; Gonçalves and Romeiro 2019),

few of these studies conducted a systematic analysis of potential drug alternatives from widely available NP databases. Yet the connection between approved drugs and their NP alternatives has not been established.

In order to elucidate the effects of specific NPs on human health, a broad categorization of NPs is needed. First, approved drugs obtained from ChEMBL and DrugBank (Gaulton et al. 2017; Wishart et al. 2018) were combined with NPs from CMAUP, FooDB, and COCONUT into a comprehensive database (Zeng et al. 2019; The Metabolomics Innovation Centre n.d.; Sorokina and Steinbeck 2020). To search for drug alternatives, some NPs that are also approved drugs (approved NPs, ANPs), can be directly identified. For the reminding drugs, machine-learning-based similarity searching using approved drugs as queries were performed to identify NP alternatives. The search identified 8,480 drug-similar NPs that were associated with 2,228 approved drugs and were classified as BNPs. Among the BNPs, 7,422 are newly discovered BNPs, and 1,580 are BNPs that have been approved as drugs (BNP drugs, or BNPDs). Throughout my approach, drugs that had BNP(s) successfully identified, were classified as BNP-substitutable drugs (BSDs), otherwise, they were BNP-unsubstitutable drugs (BUDs). Chemical properties and substructures were explored to explain the difference between BUDs, BSDs, and NPs. Later, BNPs were annotated with the mechanism of action (MOA) and Medical Subject Headings (MeSH) information in order to determine which general pathways and diseases they may affect. A diverse variety of organisms, food sources, and dietary supplement products containing these BNPs were also identified. Finally, information about adverse side effects, drug interactions, and toxicity, was compiled using existing data and/or predictions. Ultimately, this research systemati-

cally explores the NP similarity to approved drugs to provide information on potential drug NP alternatives to guide dietary supplement development and NP research.

4.3 Methods

4.3.1 Data Collection

Most NPs were collected from the COCONUT database (March 2021 version), an aggregation collection of previously well-known NP databases, including SuperNatural II, CMAUP, FooDB, ChEMBL Natural Product, and 49 other sources (Banerjee et al. 2014; Zeng et al. 2019; The Metabolomics Innovation Centre n.d.; Gaulton et al. 2017). However, some major sources included in COCONUT have been updated after this version. To obtain the latest NP information relevant to human health and diet, I combined the latest CMAUP (downloaded December 2021) and FooDB (April 2020 version) together.

Approved drugs' molecular information and data were downloaded from ChEMBL v29 and DrugBank v2022. Approved drugs that had withdrawn flags were excluded.

4.3.2 Compound Cleaning, Annotation, and Calculation

Compounds of NPs from different sources and approved drugs were combined together by InChIKey as the unique identifier. Other molecular information, like SMILE, IUPAC name, common names, and other additional information, were annotated by extracting the information from the original database or by querying ChEMBL and PubChem (Gaulton et al. 2017; Kim et al. 2021). Additional mapping information (identifier conver-

sions between databases) of each molecule to other databases was added by querying the Chemical Translation Service (CTS) API (Wohlgemuth et al. 2010).

Molecular structure files (SDF) were directly downloaded from ChEMBL or PubChem APIs by using InChIKeys as the identifier. Structure API-query-failed molecules were later computed by OpenBabel v3.1.1 in R's (v4.1.0) wrapper package ChemmineOB v1.32.0 using InChI strings (O'Boyle et al. 2011; Kevin Horan 2022). In OpenBabel calculation, molecules with ambiguous structures that resulted in errors or warnings were removed from all downstream structure-related computations. Also, at least four atoms (including hydrogen) were required to proceed. For example, molecules with too simple structures, like water, a single heavy metal atom, and simple salts, were excluded. The structure of all passed molecules was then loaded in ChemmineR v3.46.0 to calculate atom pair (AP) information (Carhart et al. 1985). APs were further standardized into fingerprints (FPs) in 1024bit. AP and FP validations were performed with ChemmineR. All invalid compounds were excluded from the downstream clustering analyses.

Forty-seven different physicochemical properties (PPs) for each molecule were calculated by CDK v2.3 in R's wrapper package rcdk v3.6.0 by providing SMILEs as input (Willighagen et al. 2017; Guha 2007). All downstream PP-based analyses excluded molecules with invalid SMILEs or failed CDK computation.

During my cleaning, I found some NPs had high amino acid (AA) content in substructures. NPs with large AA substructures are easier to be degraded in digestive tracts (Ao and Li 2013) and lead to loss of bioactivity upon intake. This research is biased toward drug alternative NPs that can be found in edible/dietary sources, so high AA fraction

NPs are not ideal for this purpose. To remove these high AA NPs, AA weight was first calculated in CDK, and then the AA fraction weight (AFW) was calculated by dividing AA weight by the total molecular weight. To determine a reasonable threshold for AFW, I extracted all molecules that were labeled with "Protein" from ChEMBL, calculated AFW for them, and used AFW of these protein molecules as the reference. Here I set the low-protein confidence level to be ≥ 0.99, which requires the AFW to be less than 1% quantile of the reference. Then the AFW cutoff was determined as 0.01055. NPs with AFW higher than this number were considered high AA-containing and removed from all downstream analyses (Fig. 4.11).

Also, when I examined molecules from FooDB, there were too many structural high-similar NPs, for example, long-chain lipid molecules that only have one or a few carbons different at the tail or molecules that only have stereochemical differences. In FooDB, all structurally identical molecule groups were identified to reduce the redundancy of molecules. Only the first molecule in each group was retained; other molecules were treated as duplicates and removed from the downstream analysis.

4.3.3 Similarity Computation

Two types of compound similarities were calculated, structural Tanimoto similarity (STS), which was based on FPs, and property Euclidian similarity (PES) which was based on PP values. The similarity values for STS and PES were calculated in ChemmineR and rdist v0.05 packages in R, respectively.

4.3.4 BNP ML Models

During the BNP searching stage, hieratical clustering (Hclust) models were built using STS and PES as input separately by fastcluster package v1.2.3 (Müllner 2013). k-means (Kmeans) models were trained using STS and PES as input separately by ClusterR package v1.2.9 (Mouselimis n.d.). The random forest (RF) model and auto-trained stack model were both trained and validated in the H2O framework with the h2o (v3.36) R package (Ledell and Poirier n.d.). 5-fold cross-validation was performed.

Before applying ML methods, 429,602 FP-based molecules passed my initial cleaning. It was computationally too challenging to compute a 400k x 400k distance matrix. I applied a prefiltering step here. In this step, a loose STS threshold at 0.75 was performed to filter out NP molecules with a low chance of being clustered with drugs. This step resulted in 32,477 NPs. Adding up the 3590 drugs, a much smaller distance matrix of 36,986 x 36,986 was generated for clustering.

The same loose threshold could not be applied to PES. Even if I set the threshold to 0.99, there were still an excessive amount of NPs passed the filter. Instead, I sorted NPs by the distance to each drug and took the top 20 closest NPs for each drug. In this way, I were able to obtain a similar-sized candidate set for PES (39,395) compared to STS (36,986), which made the clustering results more comparable later.

To train the RF model, compounds with valid PPs were categorized into two classes: approved drugs and NPs. If an NP was also an approved drug, it was classified as an approved drug. The goal was to develop a binary classification RF model to predict the possibility of incoming new compounds being approved drugs. Initially, 3089 drugs with

PPs passed the initial cleaning stage, whereas the number of NPs was over 400,000. This resulted in an extremely unbalanced training set. To address this issue, I randomly selected 3,089 NPs, similar in number to the drugs, as the training data. As a result, the training set consisted of 6,178 samples, with an equal number of drugs and NPs. Accuracy was the performance metric used for selection, and the number of trees (K) in the RF model was the optimization parameter.

4.3.5 BNP Validation/Use Case Set

Two drug lists were used to subset the entire approved drug list and the corresponding subset of BNPs to validate and demonstrate the use cases of BNPs. The first set was the DrugAge longevity-associated drug list (Barardo et al. 2017). Among all drugs in DrugAge, only those approved and positively contributing to lifespan were selected. The second set was a list of approved anticancer drugs (Sun et al. 2017).

The DrugAge list had 149 longevity drugs overlapping with my approved drug list, and these drugs paired with 1,134 BNPs. I then searched these 149 drugs, these drugs linked 1,134 BNPs, and another randomly selected 1,000 NPs (negative control) on PubMed API with the keyword pattern "longevity + {compound name}" to collect the number of scientific publications related to the compound in each group. A similar approach was used on anticancer drugs. 125 anticancer drugs, these drugs linked 384 BNPs, and 1,000 random NPs were searched with the pattern "anticancer + {compound name}".

4.3.6 Unsupervised Search

Unsupervised clustering methods, Hclust and Kmeans, were separately applied on prefiltered STS and PES candidates (Fig. 4.4A.3, Additional File. 3-4). Rather than using commonly used metrics like minimum variance or distance between clusters, my goal in this stage was to obtain as many candidate clusters as possible, even if it increased the false discovery rate (FDR). The FDR would later be controlled at the ML method intersection stage (Fig. 4.4A.4). Therefore, the clustering optimization criteria were set to find the maximum number of clusters that contains at least one drug and one NP. The optimal number of clusters was K=4520 for STS and K=12350 for PES using Hclust (Fig. 4.12.A-B) and K=6900 for STS and K=8700 for PES using Kmeans (Fig. 4.12.C-D).

Supervised Search

The optimal RF model was obtained with the smallest possible K (trees) at 650 when the maximum accuracy was stabilized to prevent overfitting. I were able to train an RF model with an accuracy of 0.929. To further evaluate the performance of this RF model, I selected an independent validation set consisting of 2963 drugs from Phase 1 to Phase 4 that were either not approved or approved but later withdrawn from ChEMBL, along with an equal number of random NPs not used in the training data. Furthermore, I compared this manually optimized RF model with the best auto-optimized stack model (Fig. 4.13). The RF model had a slightly lower performance (0.927) than the stack model (0.942, Additional File. 4) in cross-validation (Fig. 4.13A). In the independent set accuracy test, my RF model also had much higher accuracy than the stack model (0.899 vs. 0.834). As anticipated, the

binary RF model worked well in distinguishing NPs and Phase 4 approved drugs, as it had a higher error rate on Phase 1-3 drugs but a low error rate on NPs and Phase 4 drugs (Fig. 4.13B).

4.3.7 Downstream Annotations

Protein (gene) targets, MOA, MOA type, and MeSH terms annotations of drugs were obtained from ChEMBL v28. Reactome pathway information was obtained from Reactome v2022 (Gillespie et al. 2022). Food source information of NPs was obtained from FooDB v2020. Organism source information of NPs was obtained from CMAUP v2021 and COCONUT v2021. Drug interaction annotations were obtained from DrugBank v2022. Drug adverse effects information was cleaned from the pAERS drug adverse event database (Poleksic and Xie 2019).

Toxicity predictions for all compounds were done in Toxicity Estimation Software Tool (TEST) v5.1 (Martin 2016). The LD_{50} was calculated with parameters "-m consensus -endpoint LC50".

4.3.8 Functional Enrichment Analysis

Reactome pathway functional enrichment analysis (PFEA) was performed with ReactomePA package v1.42.0, and signatureSearch package v1.12.0 in R (Yu and He 2016; Duan et al. 2020). p-value was adjusted with Benjamini-Hochberg (BH) method. The adjusted p-value cutoff was set to 0.05. To prevent enriched terms from falling into too small or too large pathways, parameters "minGSSize = 10, maxGSSize = 105" were used.

4.3.9 Substructure Search & ML Models

Common compound substructures were collected from Global-Chem (Sharif et al. 2022). Substructures with empty or unmeaningful SMARTS were removed from the list. In total 2,945 SMARTS strings were used to query whether the compound has certain substructures. The query was performed in rdkit v2020.3 (Landrum et al. 2020) in Python 3.6.1, and the results were stored as binary numbers as 1 (present), and 0 (not present).

Substructure results, PPs, and the number of common organic compound elements information were used as features to build 2 types of classification models in the H2O framework. The first type was binary prediction models. All 1,335 unsubstitutable drugs and the same amount of 1,335 randomly selected NPs were used in training. The prediction outcome was the classification of NP or unsubstitutable drug. The second type was trinary prediction models. All 1,335 unsubstitutable drugs, all 2,219 substitutable drugs, and 3,554 randomly selected NPs were used in training. The prediction outcome was the classification of NP or substitutable drug, or unsubstitutable drug. 5-fold cross-validation was performed in both types. Hyperparameters were auto-tuned by the framework. The best model was selected with the best cross-validation accuracy.

4.3.10 Statistical Comparisons

Statistical comparisons between two groups were first performed on a Shapiro-Wilk's test to validate the normality (p-value > 0.05). If the normality was verified, a Student's t-test was performed. If not, the Mann-Whitney U Test was performed. For variables containing with more than two categorical classes in each variable, Kruskal-Wallis

test was used to test the difference. For all tests, the p-value cutoff was set to ≤ 0.05 to be considered as statistically different.

4.3.11 Other Packages Used

Other major packages used in this study include, tidyverse v1.3.2 (Wickham et al. 2019), ggpubr v0.5.0 (Kassambara 2022), ggsignif v0.6.4 (Constantin and Patil 2021), and ggbreak v0.1.1 (Xu et al. 2021).

4.3.12 Reproduce and Customize the Results

The methods and filters utilized in this study were based on the principles of computational rationality. However, it should be acknowledged that these filters may not always be applicable in practice. To account for the various need, the filters' stringency needs to be adjusted. In recognition of this, I have open-sourced the project and provided instructions for others to reproduce the research. This allows users to customize different intersection sets from Fig. 4.4B (i.e., picking a different set of BNPs) according to their specific stringency requirements.

As demonstrated through the above examples, systematically subsetting the BNPs is crucial. To facilitate this, I have organized all the results into an SQLite database which includes 16 tables (Fig. 4.14, Additional File. 7). This database consists of complete molecule information, BNP details, and a comprehensive collection of annotations employed throughout the study.

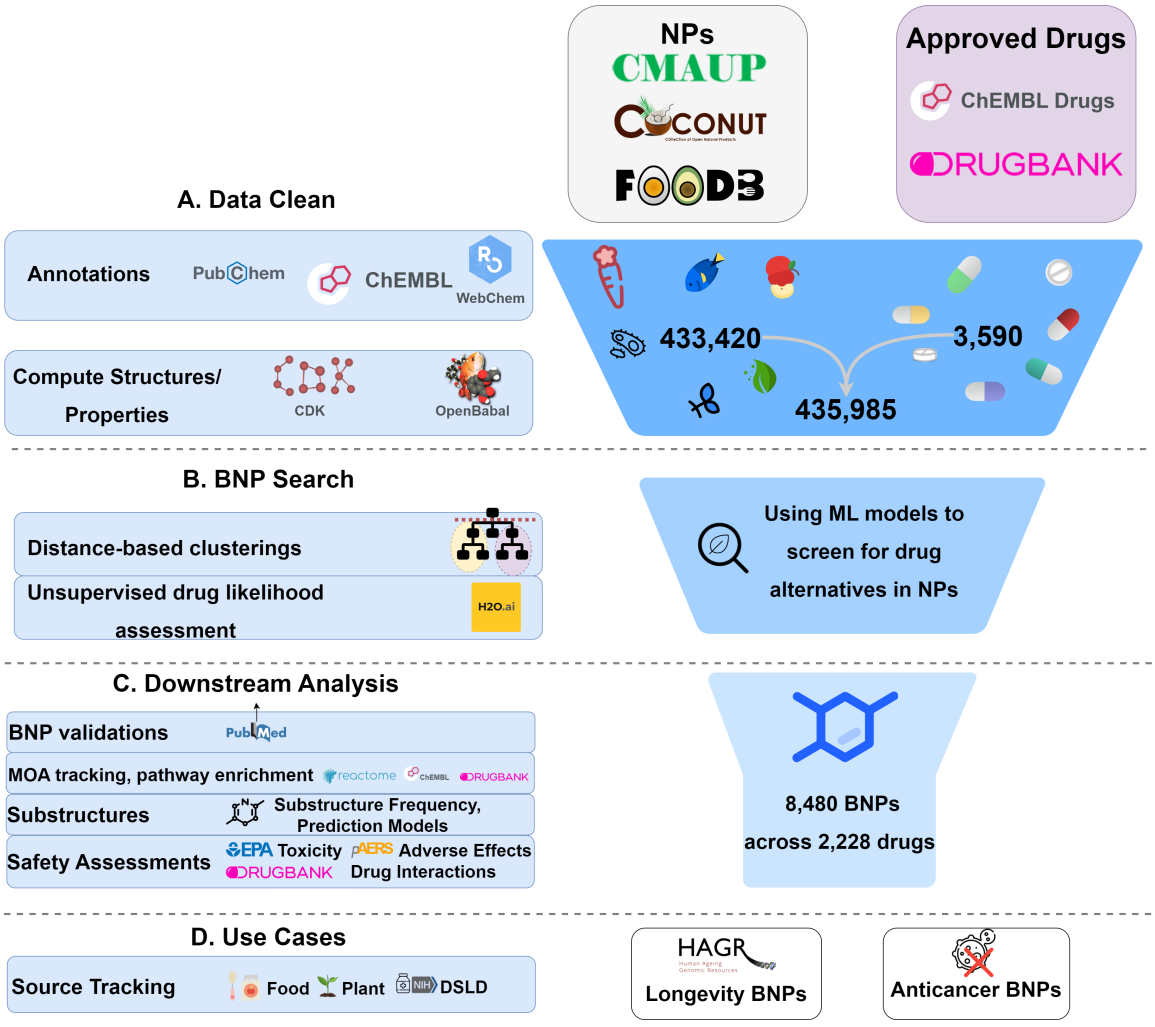

Figure 4.1: Overall Design of the Study. This study can be divided into three major sections. First, NPs and approved drugs were collected from different sources. Annotations were added and physicochemical properties were calculated. Second, drug alternative BNPs were identified based on the intersection of structural similarity, physicochemical similarity, and drug likelihood results. Third, identified BNPs were annotated with additional information, such as source food, source organism, toxicity, pathways, diseases, and more.

4.4 Results

4.4.1 Databases Cleaning and Overview

This study created a comprehensive collection of compounds obtained from approved drugs and natural products (NPs) by combining and cleaning data from ChEMBL, DrugBank, CMAUP, FooDB, PubChem, and COCONUT databases (Gaulton et al. 2017; Wishart et al. 2018; Zeng et al. 2019; The Metabolomics Innovation Centre n.d.; Sorokina and Steinbeck 2020) for a total of 435,985 compounds. Additional molecular information, compound annotations, and PPs were annotated or computed to form this comprehensive dataset (Table. 4.1-4.5, Additional File. 1 & Fig. 4.1A).

Assembling this comprehensive molecular database enabled the utilization of various machine learning approaches, including distance-based clustering and unsupervised drug likelihood assessments, to identify naturally occurring small molecules that have similar therapeutic effects as their structurally similar approved drugs (Fig. 4.1B). Putatively identified BNPs are assembled into a comprehensive database that can be used for downstream analysis, including computational validation, substructure exploration, pathway enrichment, safety assessments, and (Fig. 4.1C). Later, the use cases of BNPs, including dietary recommendations, organism source tracking and food supplement availability, were demonstrated with longevity/anticancer drug-associated BNPs (Fig. 4.1D).

Although a low percentage of NPs had been previously annotated in public chemical databases such as PubChem (44.8%) and ChEMBL (7.4%), a large number of undiscovered NPs with great research potential remained (Fig. 4.2A). While over half of the drugs

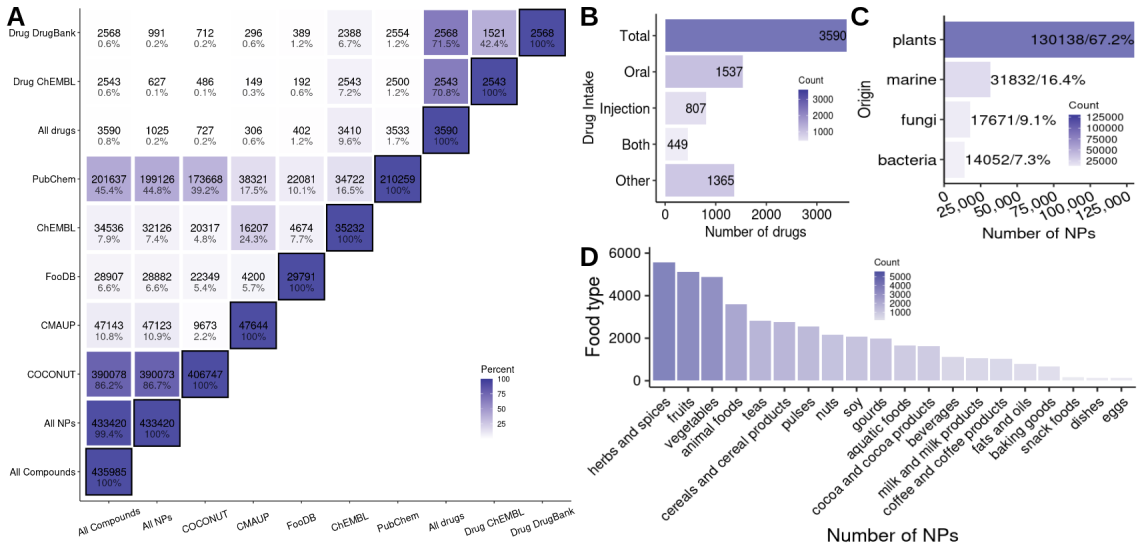

Figure 4.2: Statistics of compounds. (A) Intersection table of NPs, drugs, and their source databases. On each tile, the top number indicates the number of compounds shared (overlapping) between two data sources, and the bottom number indicates the shared identical compound percentage between two data sources. (B) Intake method statistics of approved drugs. (C) NP source organism statistics (N.A. values removed). (D) Top 20 NP source food statistics.

(1,895, 53%) in the collection are predominantly administered orally or by injection (Fig. 4.2B), plants were the largest source of NPs, with only a small portion derived from marine, fungi, bacteria, and other sources (Fig. 4.2C). Similarly, the top-ranking food types in the collection are primarily plant-based, with herbs, fruits, and vegetables ranking the highest (Fig. 4.2D).

4.4.2 Estimate Raw BNP Number

At the molecular level, the unique combination and arrangement of functional groups on a compound create a structure-function relationship that dictates its molecular targets, which determines how it affects cellular function. Identifying NP alternatives to

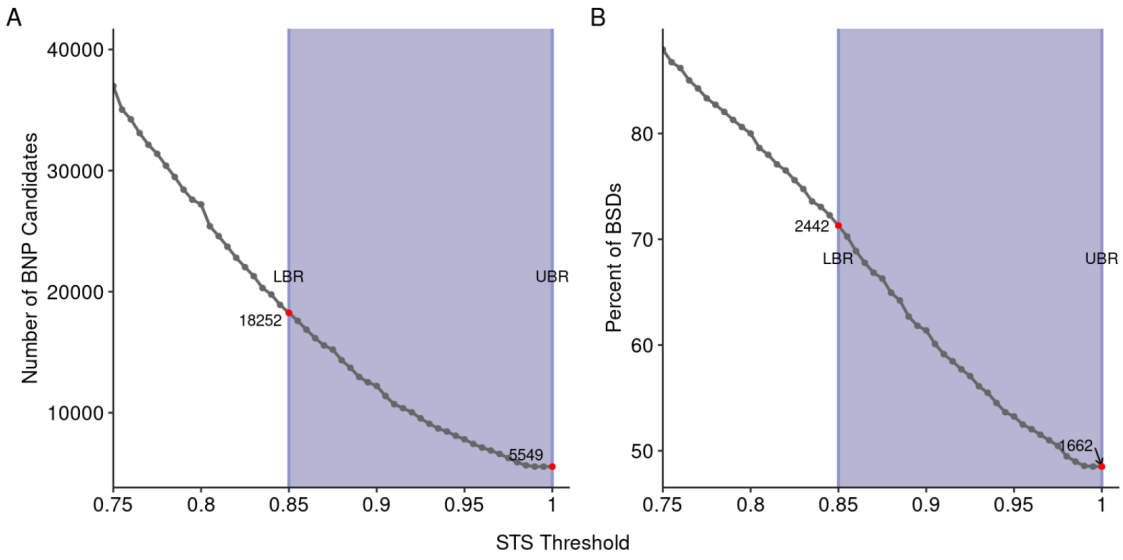

Figure 4.3: History relationship between approved drugs and NPs. (A) Approved drug numbers from 1940 to 2020. The total number of drugs approved yearly is marked in green columns, and the green line represents the trend. Drugs that were identical to their NP form (as-NP) were marked in red dots, and the red line represents the trend. Drugs that were highly similar to NPs (high-similarity-to-NP) are marked in pink dots, and the pink line represents the trend. Withdrawn drugs each year are represented as negative numbers, and the brown line represents the trend. (B) The relative percentage of different types of drugs. The as-NP drug is represented in red, and the approximate area is estimated under the red line. The high-similarity-to-NP drugs are represented in pink. Low-similarity-to-NP drugs are represented in gray color. The area of as-NP and high-similarity-to-NP drugs together is estimated under the blue line.

approved drugs requires a thorough understanding of the associations between drugs and NPs. To compare the similarity between two compounds, in other studies, STS was often used. As a common practice, an STS cutoff between 0.75-0.85 is usually applied ((Cramer et al. 2002; Martin et al. 2002). However, it is hard to justify the reasons behind the choice. Instead of using a fixed STS threshold, a more robust and unbiased process is required. Meanwhile, estimating the number of BNP candidates generated from a fixed STS cutoff could provide an important reference on the stringency level of my proposed new methods.

Therefore, I calculated the number of BNPs and number of BSDs using different STS, ranging from 0.75-1 (Fig. 4.3), resulting in 36,986-5,549 BNP candidates for 3,013-1,662 drugs. At STS=0.75, more than 90% of drugs are found with BNPs, and clearly, this threshold is too relaxed. Considering the size of NPs gathered in this study, an elevated threshold should be used. Hence, I slightly increased the estimation threshold range to 0.8-1.0 (blue regions in Fig. 4.3). This rangeis then used as my reference for stringency examinations of more complex machine learning approaches. At STS = 0.8, 27,208 BNP candidates were found, and this was used as the lower bound reference (LBR). At STS = 1.0, 5,549 BNP candidates were found, and this was used as the lower bound reference (LBR). A method with appropriate stringency should stay between LBR and UBR.

4.4.3 Search for Drug Alternative BNPs

An ML approach was conducted in this study to address this. To filter BNP candidates, I used three ML methods - including two unsupervised (Hclust and Kmeans) and one supervised (RF), that are commonly used in drug discovery (Lo et al. 2018; Vatansever

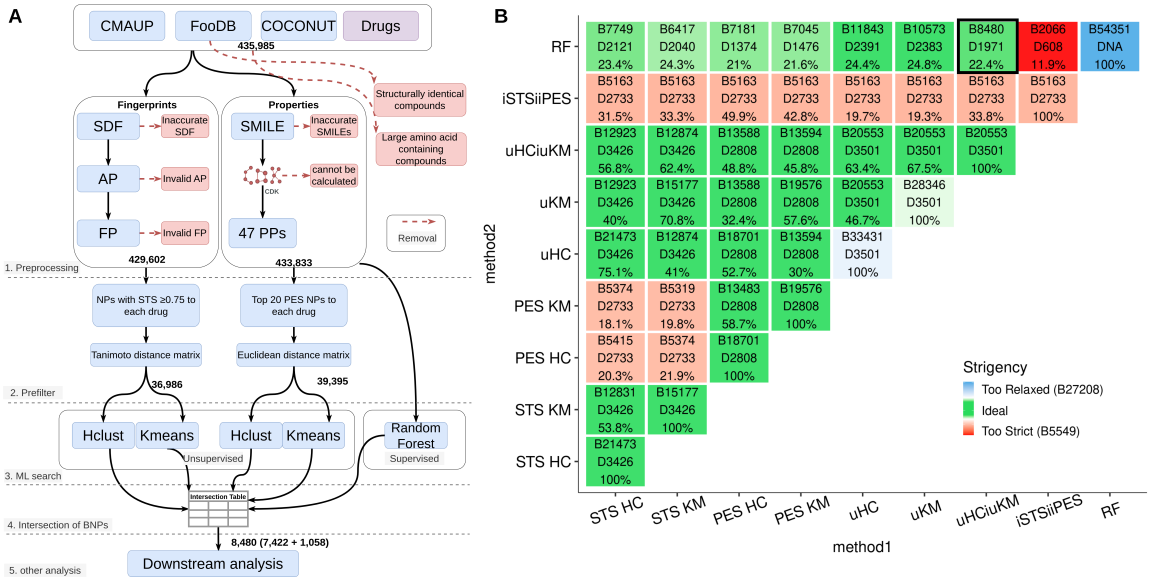

Figure 4.4: Drug alternative NP search design and results. (A) Detailed design of ML search for drug alternative NPs. (1) Structural FPs and PPs, were calculated separately and later used as two types of inputs for the ML models. Invalid compounds were removed in the reprocessing stage. (2) A relaxed filter was performed to reduce the FP and PP inputs sample size to around 40,000 before applying ML methods. (3) STS distance matrix and PES distance matrix each served as input for Hclust and Kmeans models separately. PP values were also used to train a drug-likelihood assessment RF model. (4) Results from all models were filtered in the intersection table. (B) The table displays one ML method (X-axis) intersects with all other methods (Y-axis). Notation of axis labels: Hierarchical clustering (HC). Kmeans (KM). results of STS models (STS HC, STS KM), results of PES models (PES HC, PES KM), the union of PES/PES Hclust models (uHC), the union of PES/PES Kmeans models (uKM), and the PP-based drug likelihood RF model (RF). More complicated intersection notations are: intersection of uHC and uKM (uHCiuKM), and the most strict set, the intersection of STS HC/Kmeans models intersecting with the intersection of all PES HC/Kmeans models (iSTSiiPES). The three numbers on each tile are: first row, the number of BNPs candidates (starting with "B"); second row, the number of BSDs (starting with "D"); third row, the overlapping percentage between two methods. The color code on each tile is based on stringency value reference LBR and UBR. The color is red if the number is below LBR, blue if the number is above UBR, and green if the number is in between. The final set for downstream analysis is marked in the black box.

et al. 2021; Zhang et al. 2017; Lavecchia 2015). Unsupervised models were clustered on structural- and PP-based similarities (STS and PES), where high structural/PP similar candidates were selected without manually setting up a fixed threshold. Meanwhile, the supervised RF model was trained on raw PP values as a classifier to predict the likelihood of a given compound being approved drug. Together, incorporate a diverse set of chemical characteristics across the three ML models (Fig. 4.4A). This enhanced the ability to identify compounds that had not only high structural/physicochemical similarity to at least one drug but also the likelihood of obtaining approval as a pharmaceutical agent.

However, I did not directly use the RF model to screen the remaining NPs for drug alternative candidates. It would allow only a tiny fraction of NPs to pass the filter as candidates due to its high accuracy. The max-accuracy threshold was too stringent. Instead, I aimed to allow some "false positives" NP candidates to be marked as drugs by loosening the threshold to increase the sensitivity. The specificity was controlled later in the intersection step. The optimal accuracy probability threshold value was 0.51 in the training dataset (Fig. 4.15A), but I chose 0.65 as the threshold for the RF model (Fig. 4.15B). This value was within a balanced area of equalized error rate for both NPs, approved drugs, and overall performance in the independent validation dataset. Using the RF model with a 0.65 threshold yielded 53,293 BNP candidates.

4.4.4 Combine Search Results

All ML search results were summarized in an intersection table (Fig. 4.4B). In order to assess the stringency of my machine learning (ML) model, I compared the results it yielded with hard STS cutoffs. The most stringent hard cutoff, which I refer to as the

LBR, was obtained in a previous section by setting the STS $\geq$ 0.99. In contrast, the upper bound reference (UBR) is typically a number between 0.75-0.85 in other studies ((Cramer et al. 2002; Martin et al. 2002). However, due to the use of a massive number of NPs in my research, even setting the UBR at 0.85 would still generate an excessive amount of drug-bound NPs. Therefore, I increased the UBR to 0.9, yielding 35,953 BNP candidates for 2,852 drugs (Fig. 4.4B). With the LBR and UBR values determined, I could then select my final BNPs such that it fell between these two thresholds. If a set fell outside this range, I considered it too loose or too tight in stringency.

The ML searching stage (Fig. 4.4A.3) was designed to be permissive, with the inclusion of considerable false positives to increase sensitivity. In the intersection stage (Fig. 4.4A.4), specificity was carefully controlled by intersecting different ML methods. Ultimately, my final selection consisted of the union of all Hclust results intersecting with the union of all Kmeans results and then intersecting with RF (Fig. 4.4B black box). This process identified 8,480 BNPs (7,422 NPs and 1,058 NP-as-drug compounds) paired with 2,228 drugs (Table. 4.6-4.7). The rationales are:

1. It had suitable stringency, falling between LBR and UBR.

2. Taking the union inside each of Hclust/Kmeans considered structural- and PP-based similarity.

3. Intersecting of Hclust and Kmeans results generated robust consensus clusters between clustering methods and lowered the FDR.

4. Intersecting with RF results further narrowed the candidates' scope to those with a high predicted potential to be later approved.

4.4.5 Functional Annotation of BNPs

In contrast to approved drugs, which have numerous studies and functional annotations as evidence, the molecular mechanisms of most natural products remain unclear and unannotated. This poses a significant challenge for researchers, as experimental validation for a large number of compounds with unknown target information in this study is unrealistic. Therefore, an alternative approach was necessary. One strategy for identifying unknown targets of compounds is to predict their target sites. Many models have been built using structural and PP similarities, such as SuperPred and LigTMap (Gallo et al. 2022; Shaikh et al. 2021), to achieve this goal. However, these methods have been shown to have unstable accuracy between different datasets and questionable consistency between methods (Mayr et al. 2018). In this study, my ML search had already utilized structural and PP similarity information. Instead of applying another layer of prediction to complicate the problem, and based on the evidence outlined above, I assumed that the functional targets of drug alternative BNPs were the same as the paired drugs due to their high similarity. This approach allowed us to effectively address the challenge posed by the large number of compounds with unknown target information.

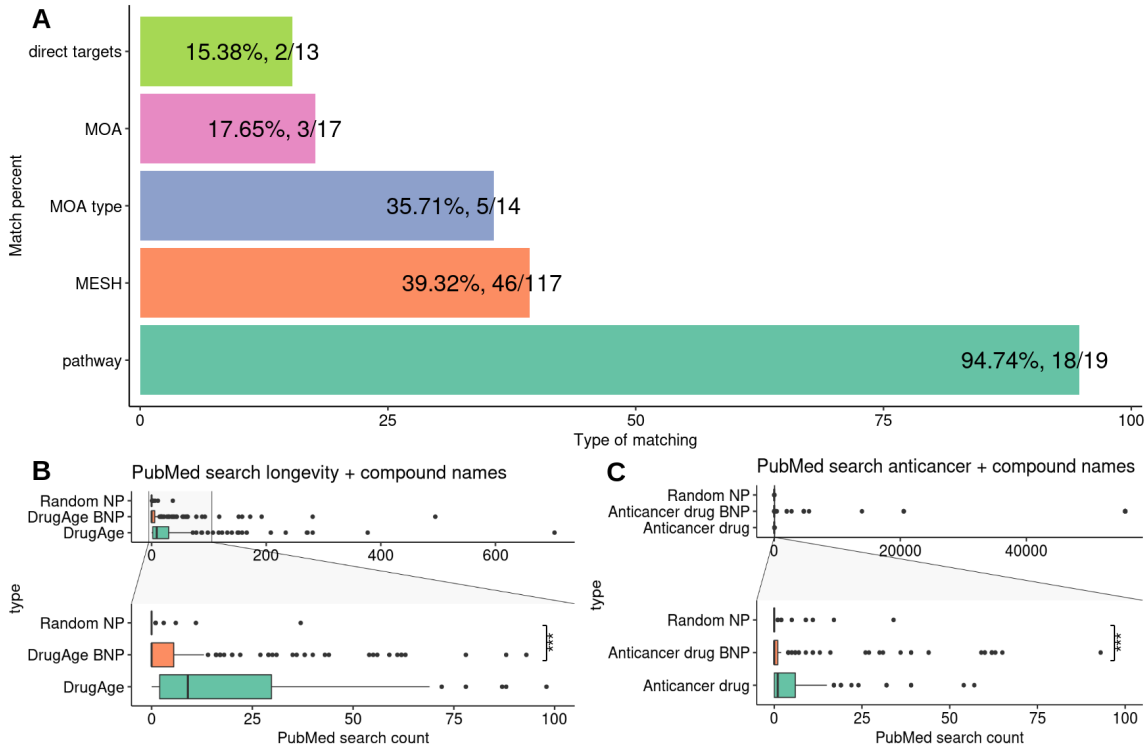

Figure 4.5: Validation of BNPs. (A) BNP-drug pairs were matched to validate if they have the same gene targets, MOA, MOA type, MeSH indications, and pathways. The numbers on each bar are: the percentage of matching, the number of matching vs. the total number of BNP-drug pairs found with available information. (B) DrugAge-associated BNP compound names, DrugAge compound names, and the same number of random NP names were searched separately on PubMed for publication records. The search keywords were "longevity" + compound names. The number of publication found under each category is displayed on a boxplot (C) Similar PubMed search was performed on anticancer drugs and their associated BNPs, with the first keyword changing to "anticancer". Notation for p-value: *** ≤ 0.01.

4.4.6 BNP Set Validation

I performed two validations to test if the assumption on inheriting functional annotations from drugs to BNPs were valid. If there is molecular evidence to support the possibility that a BNP may have a similar effect as its approved analog, and if there is any biologically meaningful information that indicates if the BNP set may have similar therapeutic efficacy as the original drug.

The first validation involved comparing BNP targets, MOA, MOA type, MeSH terms, and pathways to their paired drug. Since drugs with similar MOAs, targets, MeSH terms, and target pathways elicit similar physiological effects, the similarity between a BNP and its approved analog for these criteria is an indication that the two compounds may elicit the same phenotypic outcome. Ordered from the target level to a broader pathway level, the matching ratio elevated from 15.38% to 94.74% (Fig. 4.5A). This suggests that although BNPs might not share the same molecular target as their paired drug, they might work on proteins within the same pathways.

In addition to determining the potential biological similarity between paired BNPs and drugs, a second validation method involving identifying the number of literature reports on both drugs and BNPs within a specific area was performed. Theoretically, both BNPs and their paired drugs should have substantially more scientific reports in the selected research area than randomly selected compounds. In this study, I utilized the DrugAge (longevity) drug list and anticancer drug list for this purpose. As anticipated, DrugAge and anticancer drugs had the highest number of scientific publications on average in their respective areas (Fig 5.B-C, Table. 4.8-4.9). DrugAge BNPs and anticancer BNPs did not

have as many publications as the original drugs, but they still had significantly more literature reports (Mann-Whitney U test p-value ≤ 0.05) than random NPs (negative control). This finding suggested that my ML search strategies indeed identified some functionally similar and meaningful compounds to the original drugs in specific research areas. Thus, two validations together have approved my assumption that inheriting functional annotations from drugs to BNPs is valid.

4.4.7 Functional Enrichment

While annotating the BNPs, I noticed that some drugs have BNPs but some do not. To understand the different characteristics, such as the targets, pathways, diseases, between different drugs. They were divided into two categories. Drugs with at least one BNP alternative were classified as BNP-substitutable drugs (BSDs), while drugs without any BNP alternatives were classified as BNP-unsubstitutable drugs (BUDs). PFEA was then applied to both BSDs and BUDs. Both categories were enriched with some common pathways, but unique enriched pathways were also identified for each category (Fig. 4.6A). Next, pathways were sorted by adjusted p-value and the top-ranking pathways from both categories. BSDs were enriched in broad metabolism pathways, such as potassium channel and various degradation pathways (Fig. 4.6B, Table. 4.7). Classic cancer pathways, including p53 and S phase-expressed protein 1 (GTSE1), were also among the top enriched pathways. Similar to BSDs, some other cancer target pathways, such as General Control Nonderepressible 2 (GCN2) (Gold and Masson 2022), were enriched for BUDs. However,

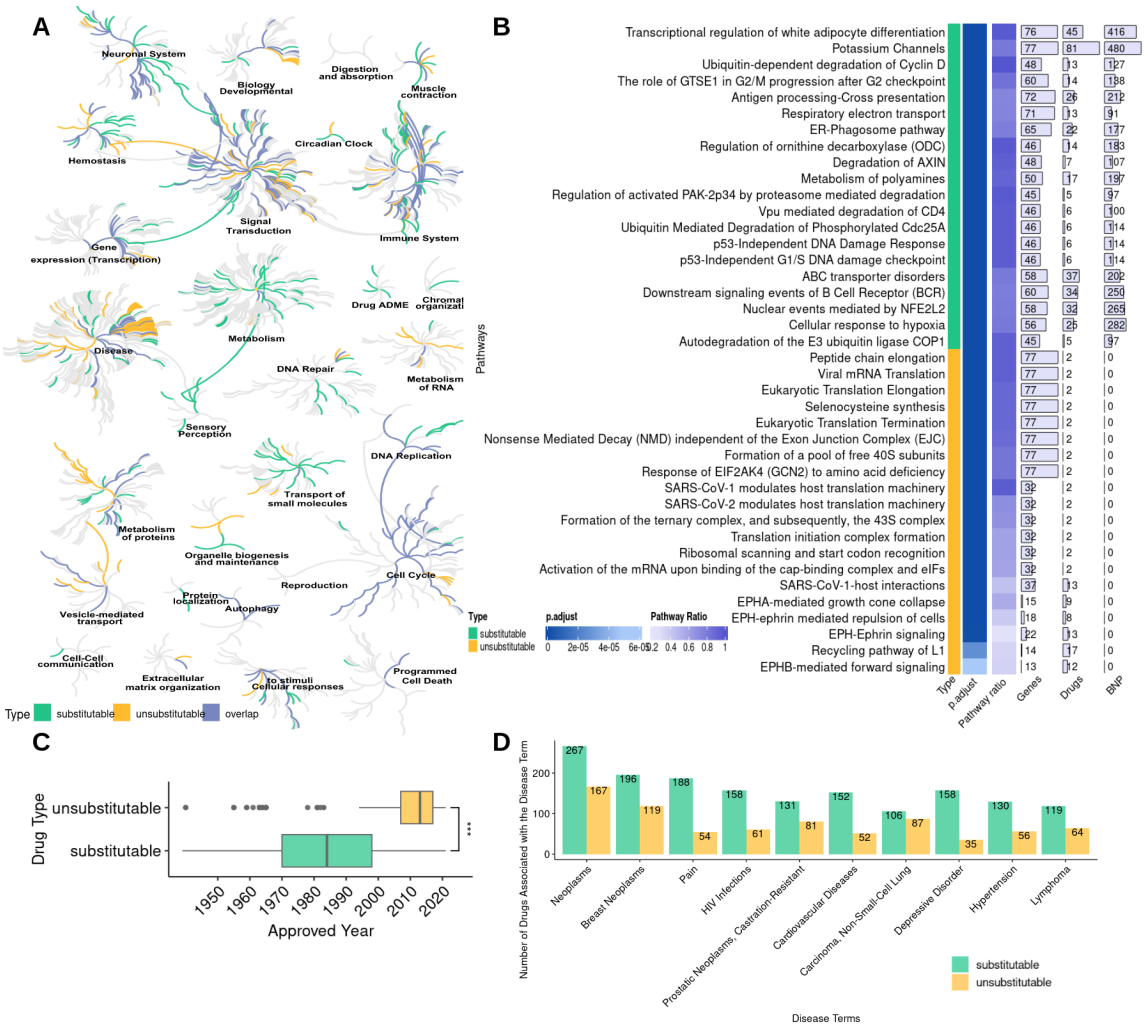

Figure 4.6: BSDs and BUDs. (A) Reactome enriched pathways overview firework plot of substitutable and unsubstitutable drugs. (B) Top 20 Reactome enriched pathways of substitutable (green) and unsubstitutable drugs (yellow). The order is sorted by adjusted p-value in each category separately. The pathway ratio was calculated by using the number of genes enriched divided by the total number of genes in the pathway. (C) (D) Approved year difference with substitutable and unsubstitutable drugs.

pathways exclusively enriched for BUDs are more related to antivirus, protein synthesis, and neuron signaling.

Both BSDs and BUDs could be enriched on a range of pathways, indicating a potential for treating various diseases (Fig. 4.6D, Table. 4.10). The little deviation between the two categories became apparent when examining the disease terms. However, a more interesting finding was discovered when comparing the dates of approval for BSDs versus BUDs. Specifically, BUDs have significantly more recent approved dates than BSDs (2010s vs. 1985s) (Mann-Whitney U test p-value ≤ 0.05) (Fig. 4.6C). Possible reasons could be the plateau effect on using existing NPs as drug candidates (Fig. 4.2) and the advancement in synthetic chemistry has allowed us to design drugs on previously impossible targets.

4.4.8 Explore BSDs and BUDs

In order to gain a better understanding of why certain drugs had no BNP partners, I first compared MOAs of both BSDs and BUDs (Fig. 4.16). The majority of top-ranking BUD-exclusive MOAs were kinase inhibitors (Fig. 4.16C), with a major pharmaceutical application being anti-cancer treatment (Bhullar et al. 2018). For example, all drugs within the top-ranking BUD-exclusive term "receptor protein-tyrosine kinase erbB-2 inhibitor" were anticancer drugs (Fig. 4.17, first two rows), and other common BUD-exclusive drugs typically had suffixes such as "-tant," "-pant," and "-sib," and were also predominantly anti-cancer drugs. (Fig. 4.17, last row). A clear pattern is that these drugs were found to be enriched with nitrogen atoms. Therefore, one possible hypothesis was the number of nitrogen atoms made the difference between BSDs and BUDs.

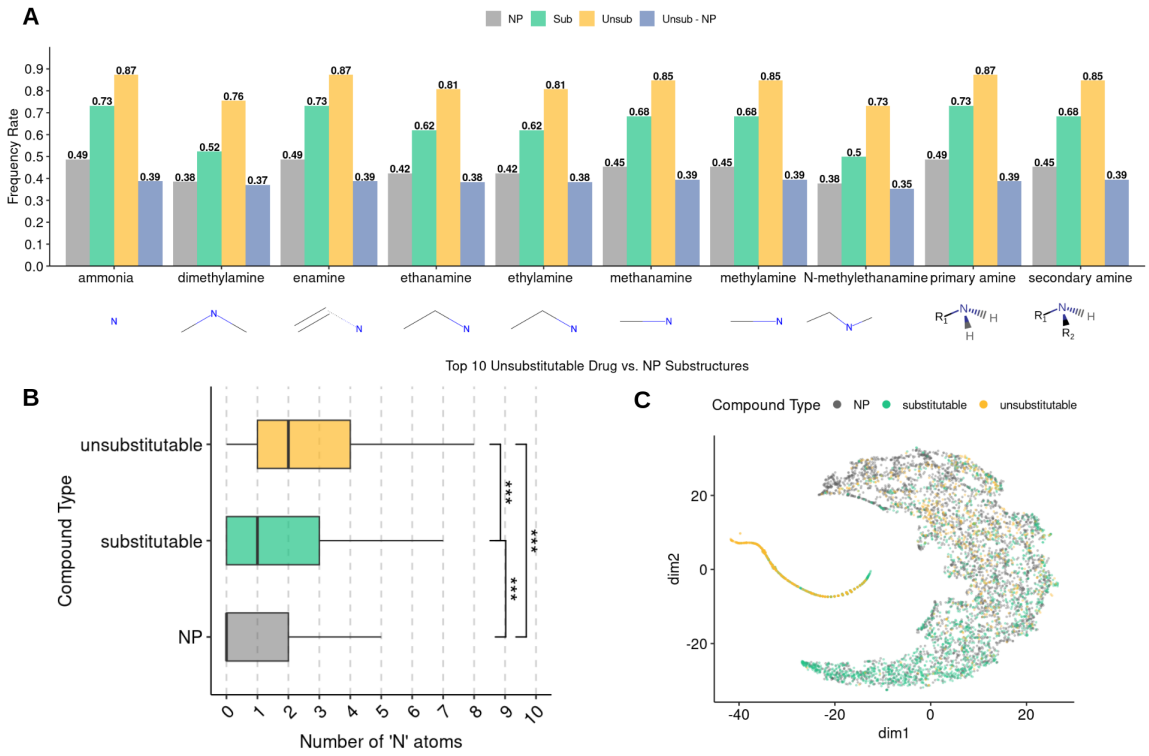

Figure 4.7: Characteristics between BSDs, BUDS, and NPs. (A) Top 10 frequency different substructures in BUDs *vs.* NPs. (B) Number of nitrogen atoms in different compound groups. (C) tSNE plot of different compound groups.

I then verified this hypothesis by comparing the nitrogen atoms to carbon atoms ratio (N/C) in BSDs, BUDs, and NPs (Fig. 4.18). I observed an increasing trend of N/C ratio moving from NPs to BSDs and finally to BUDs, and all pairwise comparisons were found to be statistically significant (Mann Whitney U Test, p-value ≤ 0.05). However, I could not conclusively determine whether nitrogen atoms were the major reason for the observed difference. Atoms do not contribute to the molecule's bioactivity alone. Interactions between atoms, and the formation of substructures (functional groups) could provide a better explanation for differences between NP, BSD, and BUD.

To discover the substructure difference in these three groups, I downloaded common substructure SMARTS strings from GlobalChem (Table. 4.11), and used rdkit to match if BSDs, BUDs, and NPs contained these common substructures. The list was then sorted by the frequency difference between BUDs and NPs (Fig. 4.7A, Table. 4.12-4.13). The results confirmed my hypothesis. All top 10 substructures were amines and derivatives of amine (nitrogen-containing functional groups). Likewise, the top 10 substructure frequency differences between BUDs *vs.* BSDs and and BSDs *vs.* NPs were smaller, but the majority of top-ranked substructures were still amines. (Fig. 4.19).

Next, it is critical to know how amine groups impact the bioactivity of drugs. One easy way to solve the problem I used was running Kruskal–Wallis tests of all other compound features against the compound classification (BSD, BUD, or NP). Other features included PPs, the number of common elements found in organic matters (C, H, O, N, S, P), and whether the compound had certain substructures. This would give us an approximation of how important other features contribute to compound classification. The result was sorted

by adjusted p-value (BH corrected), and I found that the most significant feature was the property Petitjean Shape (Table. 4.14, "Petitjean number of a molecule", "Petitjean topo shape index"). This property describes the vertex eccentricity (Petitjean 1992). As implied from other studies, the vertex eccentricity provides an important measurement of the compound's size, and shape. It is an essential property to predict bioactivity (Sharma et al. 1997; Gupta et al. 2002). As above, I have shown that BUDs had a larger frequency of containing amine groups, leading to a larger Petitjean number and bigger size than BSDs and NPs. It could be the bigger size had created more reactive sites, which makes BUDs more likely to participate in chemical reactions. Unfortunately, this could not be easily verified in my study without experimental validations. Nevertheless, other research has suggested replacing the carbon atoms with nitrogen atoms in the structure would dramatically increase the pharmacological parameters up to 1,000 fold (Pennington and Moustakas 2017).

4.4.9 Predict BUDs

Through my investigations, I have found compelling evidence indicating that nitrogen, specifically amines, played a significant role in distinguishing the categories of NP, BSD, and BUDs. Furthermore, various key features, such as substructures and PPs, played a crucial part in creating recognizable patterns that set these categories apart. These signals gave us the confidence to believe that creating a prediction model to classify these three groups of compounds was a promising endeavor. Subsequently, I utilized t-Distributed Stochastic Neighbor (tSNE) to gain a comprehensive overview of the pattern clearance between the groups (Fig. 4.7C). My analysis revealed that BUDs were concentrated mainly

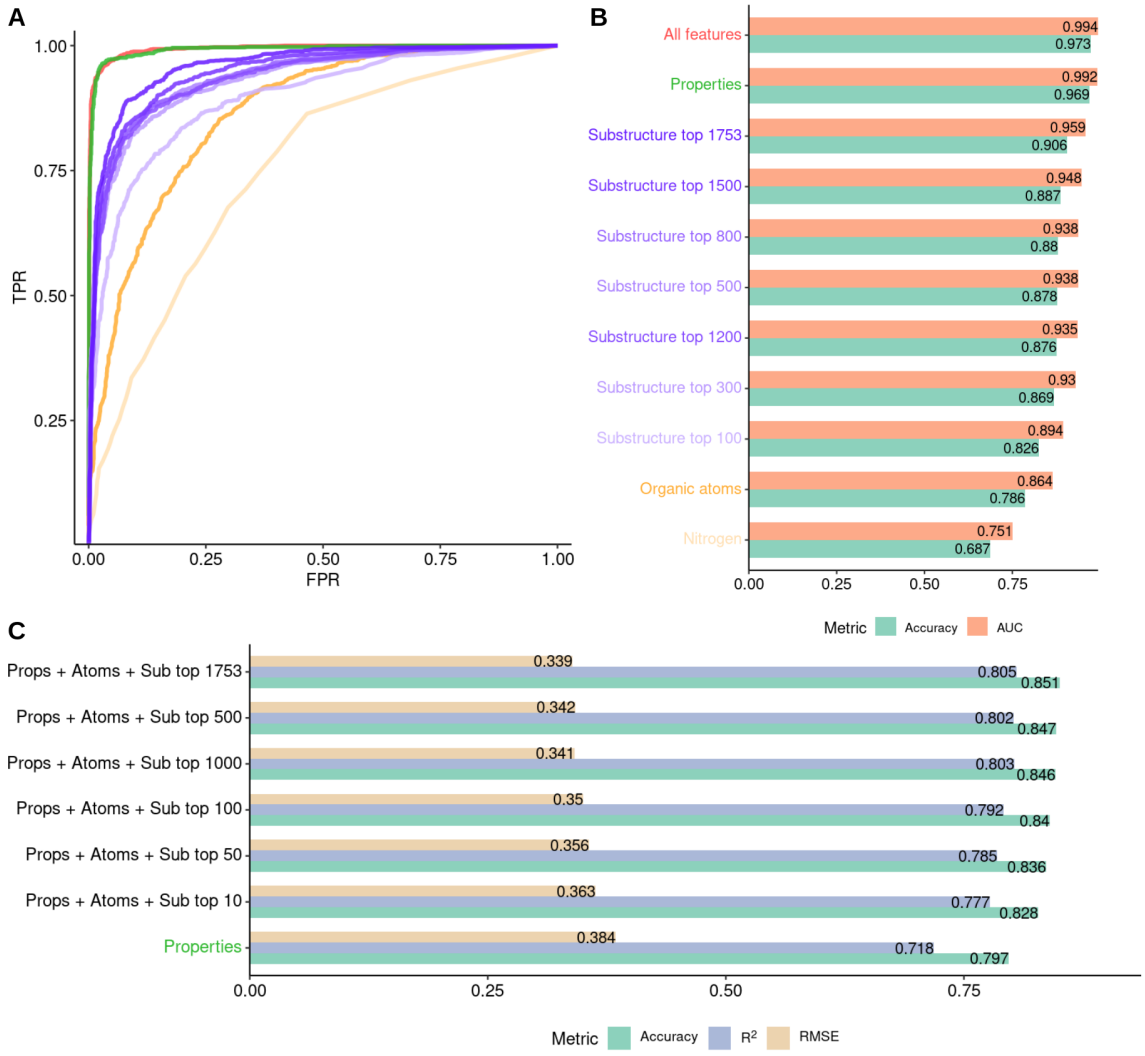

Figure 4.8: Model benchmarks for BUD prediction. First row (A-B), binary classification models. Second row (C), trinary classification models. (A) ROC curve of different binary models. The color code is displayed in (B) as row names. (B) Metrics of different binary models, sorted by accuracy (C) Metrics of different trinary models, sorted by accuracy.

in the isolated long tail, while the majority of BSDs held positions at the bottom of the right shape. The higher concentration of NPs were situated at the top of the right shape. However, in the middle of the right shape, compounds from all three groups were clumped together. It is crucial to highlight that the tSNE plot was used to reduce the dimensions to two for visualization purposes. It was expected that with more dimensions (features) included, the boundary between the groups would become more distinguishable and evident in the actual model.

Similar to the unsupervised model I built for drug likelihood classification, here, I used the same H2O framework to auto-train models with various learning methods, including but not limited to GLM, XGboost, RF, and other methods (Table. 4.15). Only the best-performing stack model in terms of cross-validation accuracy was reported.

Initially, I trained my model to differentiate between NPs and BUDs by considering over 1800 features, including PPs, organic common elements, and substructures (Fig. 4.8A-B). My intuition was correct as this approach yielded a high accuracy of 0.973 (Additional File. 5), but the process was time-consuming and difficult for future researchers to implement. Therefore, I aimed to develop a smaller model with fewer features that could still deliver comparable accuracy. Through several other models, I discovered that a single nitrogen count feature could give an accuracy level of 0.7, and by expanding the atom count to include other common organic elements, the accuracy rose to 0.8. This suggests that by knowing the chemical composition of a compound, it is possible to determine whether it is an NP or BUD, indicating whether it occurs naturally or synthetically. I also developed a

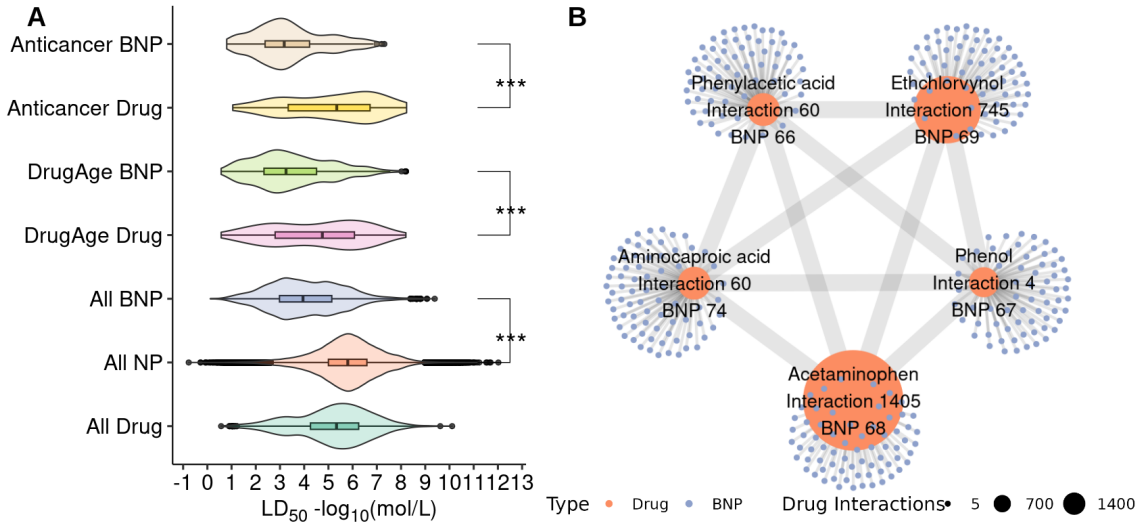

Figure 4.9: BNP toxicity and drug interactions. (A) Violin plot of mice LD50 toxicity predictions calculated by TEST for different compound groups. The larger the value, the more toxic it is. (B) Network plot example of drug interactions. The top 5 drugs with the most BNP partners are selected in this example. Each blue dot connection to the red circle indicates a BNP molecule association. The size of the red circles indicates the number of drug interactions with other drugs (not shown on the plot). The gray lines indicate drug interactions between these 5 drugs. p-value indication: ***, < 0.001.

properties model with 47 PP features, which achieved an accuracy level of 0.969, similar to the all-features model with over 1800 features.

Subsequently, I tested my model on trinary classifications (NP, BSD, and BUD) and found that their accuracy levels did not match those of the binary models. The highest accuracy achieved among these models was 0.85 in the all-features model (Prop + Atoms + Sub top 1753) (Fig. 4.8C, Additional File. 6). Nevertheless, the properties model outperformed the others by achieving 0.8 accuracy with only 47 features, making it the most effective and efficient model among these trinary classifications.

4.4.10 Safety of BNPs

Coming back to BNPs, the safety of compounds is a crucial concern in drug discovery, encompassing toxicity, drug interactions, and adverse effects. In this study, I utilized the TEST software to estimate the LD_{50} toxicity of all molecules (Fig. 4.9A, Table. 4.16), and my selected BNP subset exhibited significantly lower toxicity compared to random NPs. Furthermore, their DrugAge/Anticancer associated BNPs showed significantly lower toxicity than the original drugs in these lists respectively. This highlights the potential of BNPs in future drug discovery, especially in the search for alternatives to highly toxic anticancer drugs (Remesh 2012).

While data on drug interactions and adverse effects are usually generated from animal and cell culture studies, as well as patient reports, such research is costly and often unavailable for BNPs. Nonetheless, since BNPs were selected based on their structural and property similarity to drugs, and their biological functions have been similarly approved, it is reasonable to assume that they would have similar drug interactions and adverse effects. Thus, I extended drug interaction linkages between drugs and BNPs based on information from DrugBank, and adverse effect information from pAERS (Table. 4.17-4.18). An example of interaction linkage can be illustrated in Fig. 4.9B. Although it is difficult to assess the safety of drug interactions and adverse effects in a generalized manner, as they often depend on the treated symptoms, such information could be useful on a case-by-case discussion.

One cannot be readily inferred from data obtained via pAERS whether a molecule's adverse effect safety is comparable to that of another molecule. This is due to the fact that

the number of adverse effect records is skewed towards drugs well-studied and have long history. For example, three of the most commonly reported drugs with the highest side effect records in the database were Aspirin, Acetaminophen and Folic Acid (Table. 4.18). The database does not provide usage statistics, which makes it impossible to normalize the counts. Despite this limitation, the adverse effect annotations are valuable as explanatory safety guidelines for drugs/BNPs.

4.4.11 Use Case of BNPs

With annotations of BNPs, some more practical questions could be addressed, for instance, the source organisms, food, and dietary supplements that contain these BNPs. Again, as illustrated above, the BNP set generated in this research can be used as drug alternatives for a variety of symptoms and diseases. There's no single panacea. The same applies to BNPs. Therefore, one would always need to further subset the BNPs to a smaller set that fits certain conditions. Here, I continued to use the longevity drug list and anti-cancer drug list as examples to demonstrate the source tracking of BNPs.

With annotations of BNPs, some more practical questions could be addressed, for instance, the source organisms, food, and dietary supplements that contain these BNPs. Again, as illustrated above, the BNP set generated in this research can be used as drug alternatives for a variety of symptoms and diseases. There's no single panacea. The same applies to BNPs. Therefore, one would always need to further subset the BNPs to a smaller set that fits certain conditions. Here, I continued to use the longevity drug list and anti-cancer drug list as examples to demonstrate the source tracking of BNPs.

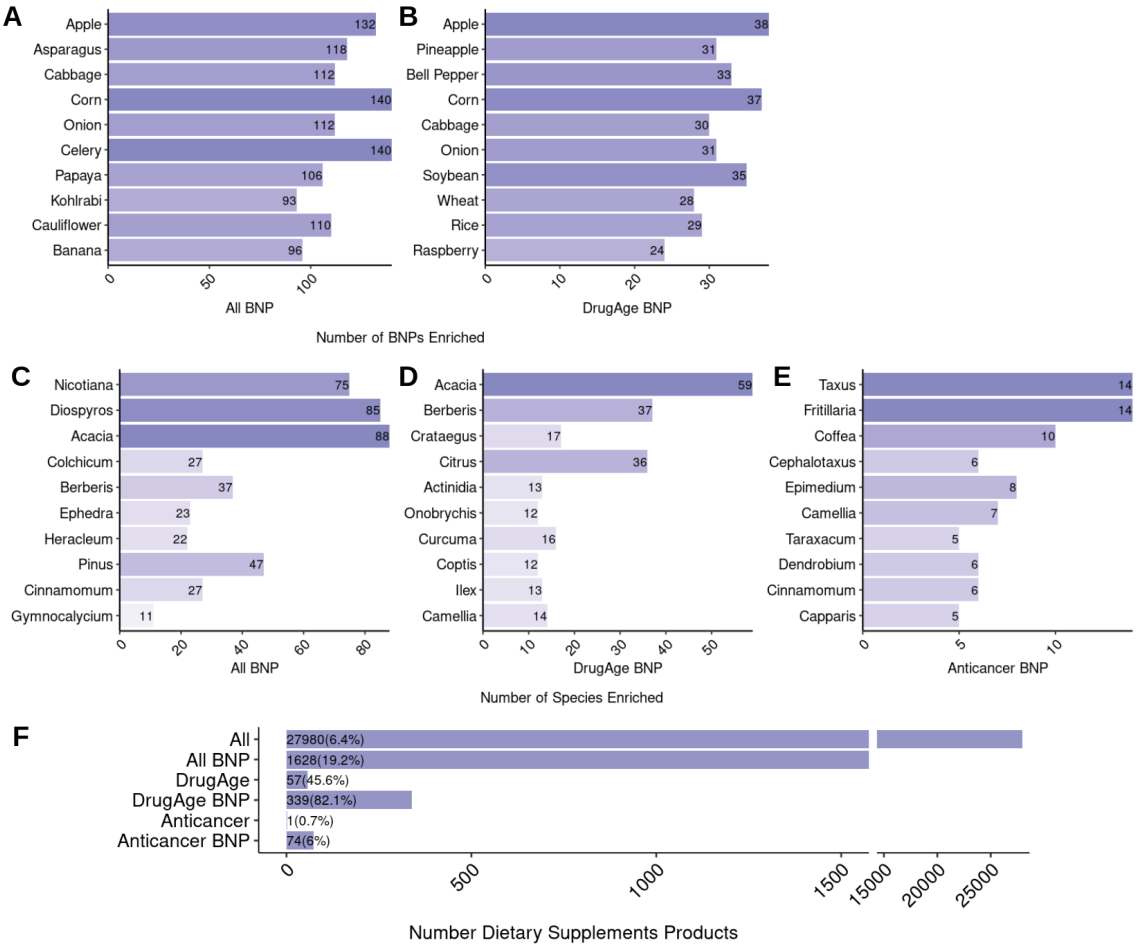

Figure 4.10: Source information of BNPs. (A) Top 10 food sources of all BNPs. (B) Top 10 food sources of longevity-associated drug BNPs. (C) Top 5 food sources of anticancer drug BNPs. (D) Top 10 organism sources of all BNPs. (B) Top 10 organism sources of longevity-associated drug BNPs. (A-E) were ranked by relative abundance. (F) The number of dietary supplement products found for different categories of molecules. The two numbers on each bar are: the raw numbers of compounds and the percentage of compounds under each category found to have dietary supplement products.

The original databases containing information about organisms and their food sources were skewed toward those organisms that have received more research attention, such as Arabidopsis (Table. 4.19). To rectify this bias and ensure a fairer assessment, I conducted hyper-geometric enrichment on BNP source food terms and organism terms at the genus level for DrugAge BNPs and anticancer BNPs respectively. The results were then arranged based on the adjusted p-value (Fig. 4.10A-E, Table. 4.20). With the incorporation of BNPs, a variety of practical questions pertaining to source organisms, food, and dietary supplements that contain these compounds can be addressed. As mentioned earlier, the BNP set derived from this research presents promising opportunities for the development of drug alternatives targeting multiple symptoms and diseases. However, there is no single panacea for all ailments, and the same applies to BNPs. Thus, it becomes necessary to further categorize and subdivide the BNPs into a smaller set that specifically addresses certain conditions. Source tracking of these smaller sets allows us to explore novel avenues of research for the betterment of human health. To this end, here, I demonstrate the source tracking of BNPs using the longevity drug list and anticancer drug list as examples. By identifying the plethora of compounds in each category, I could better understand the biological mechanisms of these drugs and the potential benefits of BNPs as drug alternatives.

The food source tracking indicated that general BNP-containing foods were largely comprised of vegetables and fruits, with 7 out of 10 top enriched foods being vegetables and the remaining 3 being fruits (Fig. 4.10A). However, when focusing on BNP-containing foods related to longevity, the top 10 sources were more evenly distributed between vegetables, grains, and fruits (4, 3, 3, Fig. 4.10B). According to the USDA dietary guidelines, a healthy

diet should comprise grains, vegetables, fruits, proteins, and lipids (USDA n.d.). Such a balanced diet has been proven to promote longevity (Lim 2018), particularly regarding fruits and vegetables (Åhlberg 2021; Dossey 2002). While I were unable to identify any significant enrichment of anticancer BNP in food sources, this outcome was not unexpected. Anticancer drugs often have severe effects on dividing cells resulting in higher toxicity levels, and therefore, the BNP forms of these drugs are typically not found in food.

As for the organisms that contain BNPs (Table. 4.21), as expected, most of them were plants (Fig. 4.10C). Furthermore, when considering only longevity-associated BNPs (Fig. 4.10D), the trend remained the same with the majority of the organisms. Interestingly, all the top 10 source organisms identified were commonly known TCMs: Acacia (wattles), Berberis (barberry), Crataegus (hawthorn), Citrus, Actinidia (kiwifruit), Curcuma (turmeric), Coptis (Huang Lian), Onobrychis (sainfoin), Camellia, and Ilex (Dong Qing). Previous studies have emphasized the positive effects of TCMs on longevity and their potential role in enhancing life expectancy (Liu et al. 2017; Wang et al. 2022). Moreover, extracts of active ingredients from some of these top-listed genera, such as Acacia (Abduljawad 2020), Berberis (Kalmarzi et al. 2019), and Citrus (Fernández-Bedmar et al. 2011) have been reported to promote longevity. With regards to the source organisms for anticancer BNPs (Fig. 4.10E), some well-known anticancer genera such as Taxus (Malik et al. 2011), Fritillaria (Al-Snafi 2019) and Camellia (Esghaei et al. 2018) were among the top-enriched ones.

Lastly, I monitored the details of dietary supplement products containing longevity drugs, longevity-associated BNPs, anticancer drugs, anticancer-associated BNPs, and BNPs

in general using DSLD (Fig. 4.10F, Table. 4.22). Understandably, that is uncommon to directly add approved drugs, especially anticancer drugs as ingredients in dietary supplements. However, using drug BNPs as search keywords, the discovery rate of longevity-associated dietary supplement products increased from 45.6% to 74.7%, and from 0.7% to 6.4% for anticancer-associated products. Therefore, my BNPs can serve as a link between drugs and food supplements, enabling potential food supplement products to be annotated with drugs that were originally lacking food supplement information.

Overall, my source tracking on food, organism, and dietary supplement not only generated some sets of meaningful sets that can be used for future health and dietary recommendations, but also, on the other hand, has proven that my initial selection of the BNPs was reasonable.

4.5 Discussions

4.5.1 Extend the Use Cases

My approach has provided a novel way to link drugs with NPs. These drugs can treat diverse symptoms/diseases which cover a wide range of research areas. One would need to have a list of drugs or diseases in mind to subset BNPs into smaller research topics. I used longevity and anticancer lists as examples, but this can be extended to other high-demand topics, like neurological degenerative drugs, or cardiovascular drugs, *.etc.* In the case where there are only a few drugs/diseases of interest. Analyzing data in greater depth is required, which is also supported by my results. The BNP table could be used to query specific BNPs, source foods, organisms, and dietary supplements.

The approach I have developed offers a unique method for linking drugs with BNPs. This innovation has tremendous potential in treating an array of diseases and symptoms across multiple research areas. To begin, researchers may create a list of drugs or diseases in mind and use the resulting information to subdivide the BNPs into smaller research topics. I have provided examples using longevity and anticancer lists, but this technique could be adapted to other high-demand topics, such as drugs to treat neurological degeneration and cardiovascular disease. In some other cases, where the list of drugs of interest is small, deeper data analysis is required, but still, such requirements can be supported with my research findings.

For instance, exploring specific drugs from the DrugAge (longevity) list can provide greater insights into their health benefits and alternative options. Here, I use two popular diabetic drugs, metformin and acarbose, as an example. They have BNP annotations that have been tracked from my results (Table. 4.13). The BNP-containing alternatives to these drugs have been found in many food items commonly consumed in our daily diet, such as pepper, berry, onion, and others. While some of these food ingredients might contain sugar, it is reassuring to know that my BNP table provides 1,350 acarbose and 15 metformin alternative BNP-containing dietary supplements choices for those who are concerned (Table. 4.13). Similarly, another longevity drug curcumin, known for its numerous health benefits, has also been found in various food sources and dietary supplement products (Table. 4.13). Previous studies have highlighted that the curcumin dosage for maximum benefits could be as low as 80mg/day (Cox et al. 2015), equivalent to about 2-3g of turmeric powder. In fact, turmeric, commonly used in Indian cuisine, has the highest recorded content of curcumin,

with dry powder containing up to 3.14% (Tayyem et al. 2006). Thus, one can look up my BNP source food tracking table and look for curcumin-containing food, such as different types of curries. For groups such as Indians who regularly consume these types of spices, they may be aware that additional curcumin drugs or supplements may not be necessary.

4.5.2 Limitations

One of the limitations of this study is the lack of knowledge concerning NPs, which has resulted in a low NP annotation rate (Fig. 4.20A). This study sourced compounds from various NP databases, so the source organisms for most NPs were identified (75%). There is a dearth of knowledge on other annotations such as MOA, targets, adverse reports, and interactions with other compounds, with levels close to 0%. While the study has enabled drug-BNP linkage, the annotation levels remain low, as highlighted in (Fig. 4.20B). One alternative method to annotate NPs is using other software to generate direct predictions rather than relying on drug-BNP linkage. However, selecting an accurate and stable model is one of the most significant challenges. As additional insights continue to emerge about the characteristics of NPs, I believe that further promising findings will materialize, further fueling the research on drug-BNP predictions.

4.5.3 Other Applications

The potential applications of my research findings are far-ranging and exciting. One notable example is their use in medicinal research, where NPs are often paired with metabolomes. The discovery of BNPs could help identify metabolome changes that may have potential benefits and significant implications. On the drug discovery side, monitored

analysis of the metabolites present in BNPs could contribute to the development of new drugs and therapies. Further down the road, a thorough examination of the metabolic pathways involved in the biosynthesis and degradation of natural products using metabolome analysis on BNPs can inform the design of new biotechnological processes.

Moreover, my results can be employed as a valuable tool for maintaining a healthy diet. By tracking symptoms and drugs, one can identify corresponding BNPs, food, and food supplements. Conversely, by providing a list of foods, one could track the included BNPs and trace them back to their associated diseases or drugs. This method enables the evaluation of one's diet for the presence of required BNPs, ensuring their intake for particular medical purposes.

4.6 Conclusions

Through my work, I have achieved significant progress in identifying a set of BNPs using advanced unsupervised and supervised methods. By focusing on structural and PP similarities, I have established a large-scale connection between approved drugs and known NPs, providing a valuable resource for researchers in various fields. For BUDs that failed to link to any NP, I analyzed the reasons behind the chemical substructures, built prediction models that can help researchers distinguish between NPs, BSDs, and BUDs. Thus, my work not only expands the repertoire of known BNPs but also enables more accurate screening and selection of potentially useful compounds. Furthermore, I have annotated my BNP set with additional information on their functions, safety, and sources. By tracing a few examples of BNPs to specific organisms, foods, and supplements, I have demonstrated the

power of source tracking and its potential applications in various fields. My annotations also offer new insights into the properties and potential uses of individual compounds, enhancing their value as resources for drug discovery, metabolic profiling, and healthy nutrition. Overall, my work represents a significant step forward in the identification and annotation of BNPs, with many practical applications and potential benefits in diverse domains. By providing a valuable resource for scientists and researchers, I hope to facilitate further progress in the field and contribute to the development of new and effective bioactive molecules.

4.7 Tables

Most tables and files in this study are too big to fit in a text document. For tables, only the captions are listed in the dissertation. I have uploaded all the files to Synapse at https://www.synapse.org/bioNPs. The content for tables and files is provided in the online supplement, and please view them at Synapse.

Table 4.1: All compound metadata. This table includes some logical columns. "1" means "Yes" or "Present"; "0" means "No" or "Unpresent".

Table 4.2: All compound PPs. Values are calculated from JDK. SMILES invalid compounds were removed. PP names are listed in abbreviation. Refer Table. 4.3 for the full names.

Table 4.3: Full PP descriptor names for columns in Table. 4.2.

Table 4.4: CTS mapping information. This table includes possible compound ID mappings from this study to other databases.

Table 4.5: All compound KEGG mapping. Compound ID mappings from this study to KEGG.

Table 4.6: BNP metadata. General information of BNPs selected in this study, including MOA, targets, disease, source, *etc.*

Table 4.7: BNP metadata use drug ID as primary key. This table is similar to Table. 4.6, but instead of using BNP as the key, this table is summarized by drug IDs.

Table 4.8: DrugAge BNP PubMed validation. Number of publications search results using DrugAge drugs and DrugAge-associated BNPs.

Table 4.9: Anticancer BNP PubMed validation. Number of publications search results using anticancer drugs and anticancer-associated BNPs.

Table 4.10: BSD BUD pathway enrichment. Raw PFEA results for both BSDs and BUDs.

Table 4.11: Cleaned Global Chem SMARTS for common compounds. This table contains common chemical structures as SMARTS, which are summarized from various studies by Global Chem. Empty strings removed.

Table 4.12: Raw substructure search results. Substructure queries made in rdkit. If a compound has a substructure, it records as "1", otherwise "0".

Table 4.13: Substructure frequency difference of NP BUD BSD. This table has substructure counting and substructure frequencies for NPs, BSDs, BUDs. It also has substructure frequencies differences between groups, and the table is sorted based on substructure frequencies between BUDs and NPs.

Table 4.14: Kruskal Wallis Test of the importance of features. This table records Kruskal Wallis Test results using all other compound features to test against the type classification column (BUD, BSD, NP).

Table 4.15: Best substructure binary stack model composition. Individual models and their importance in the best-performing BUD-NP predicting stack model.

Table 4.16: LD_{50} Toxicity predictions by TEST for all compounds.

Table 4.17: Drug interactions summarized from Drug Bank queries.

Table 4.18: Drug combination adverse effects cleaned from pAERS database.

Table 4.19: Raw compound food source mapping information for all compounds, cleaned from FooDB.

Table 4.20: Food source enrichment for all BNPs, longevity BNPs, and anticancer BNPs.

Table 4.21: Source organism genus enrichment for all BNPs, longevity BNPs, and anticancer BNPs.

Table 4.22: Food supplement product information. This table contains food supplement product information for all compounds, fetched from DSLD database.

4.8 Additional Figures

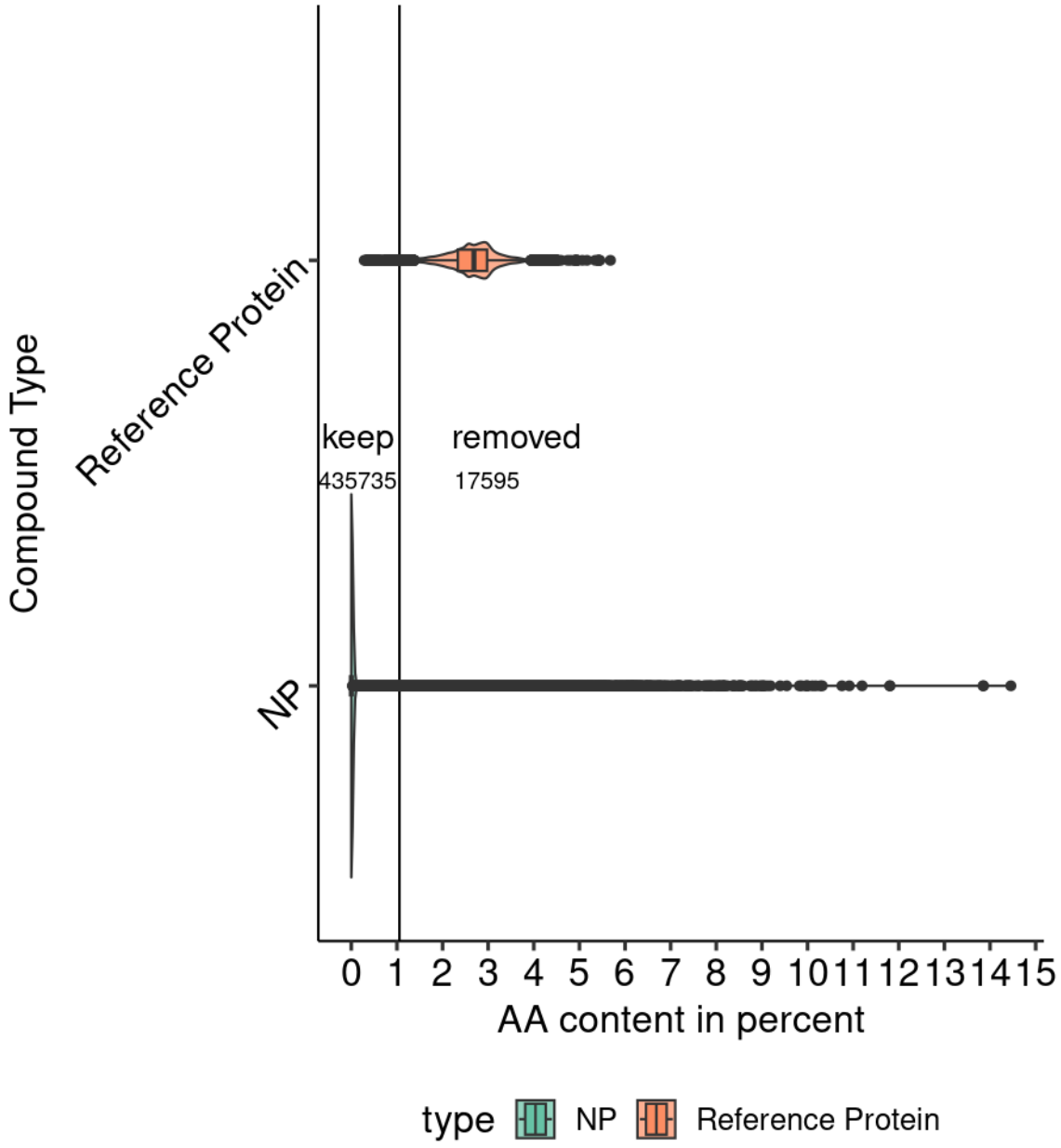

Figure 4.11: AA content in NPs. The AA percentage in NPs is compared to the percentage of protein compounds in ChEMBL database. The solid line indicates the 1% quantile of the reference content of proteins in ChEMBL database.

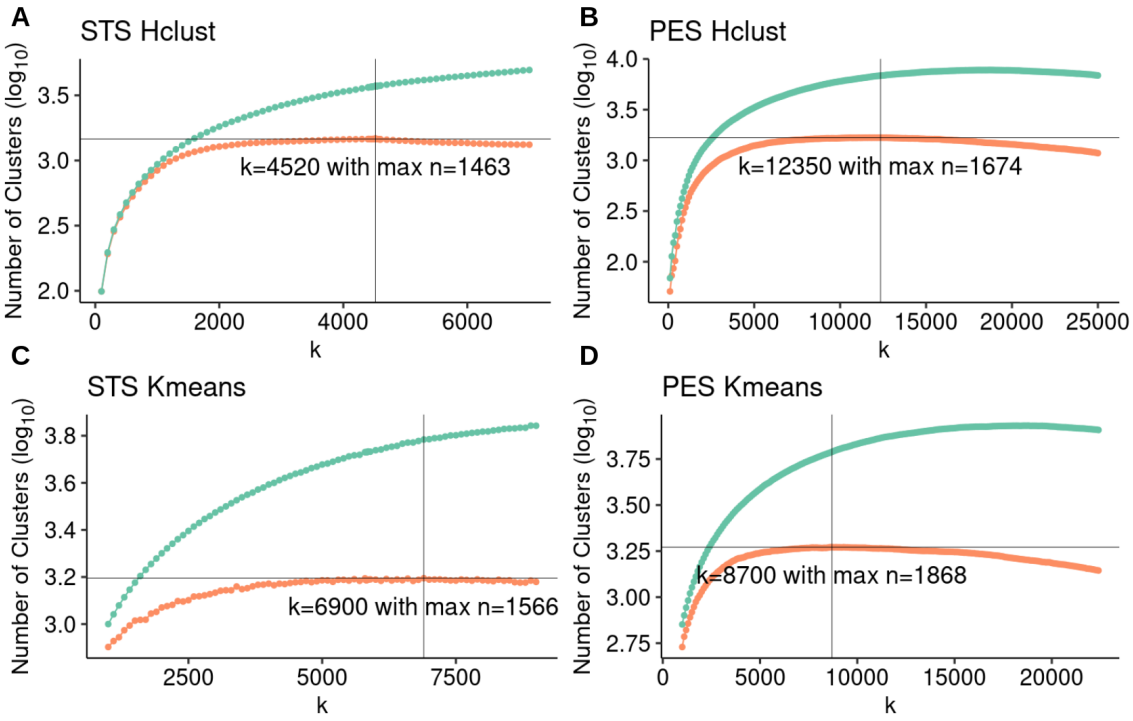

Figure 4.12: Optimization of K for Hclust and Kmeans. The green lines indicate the number of clusters obtained when K is increased. The red lines indicate when K is increased, the trend of the number of clusters containing at least one drug and one NP (n). The optimal K is obtained when the max n is reached for all 4 clusterings.

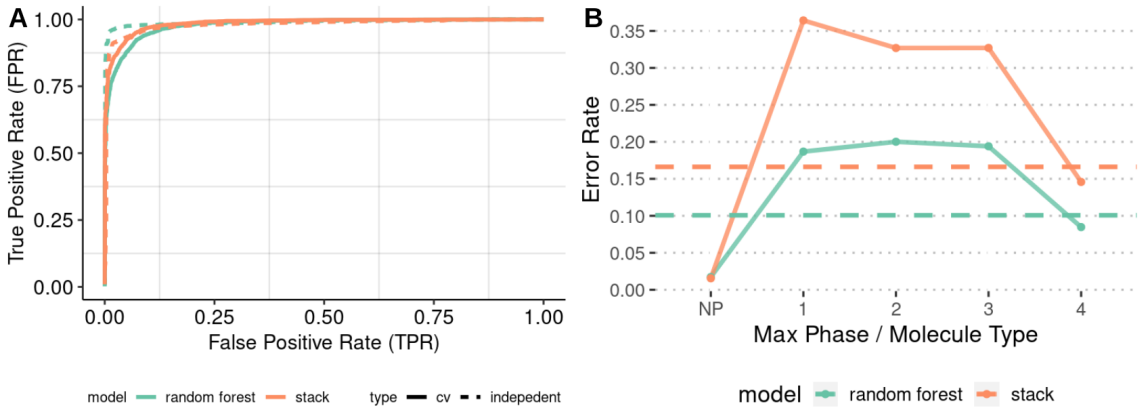

Figure 4.13: Performance of stack model and RF model for predicting drug likelihood. (A) ROC curve of models. (B) Error rate of models predicting on independent test compound set. This set contains NPs and drugs at clinical phases 1-4.

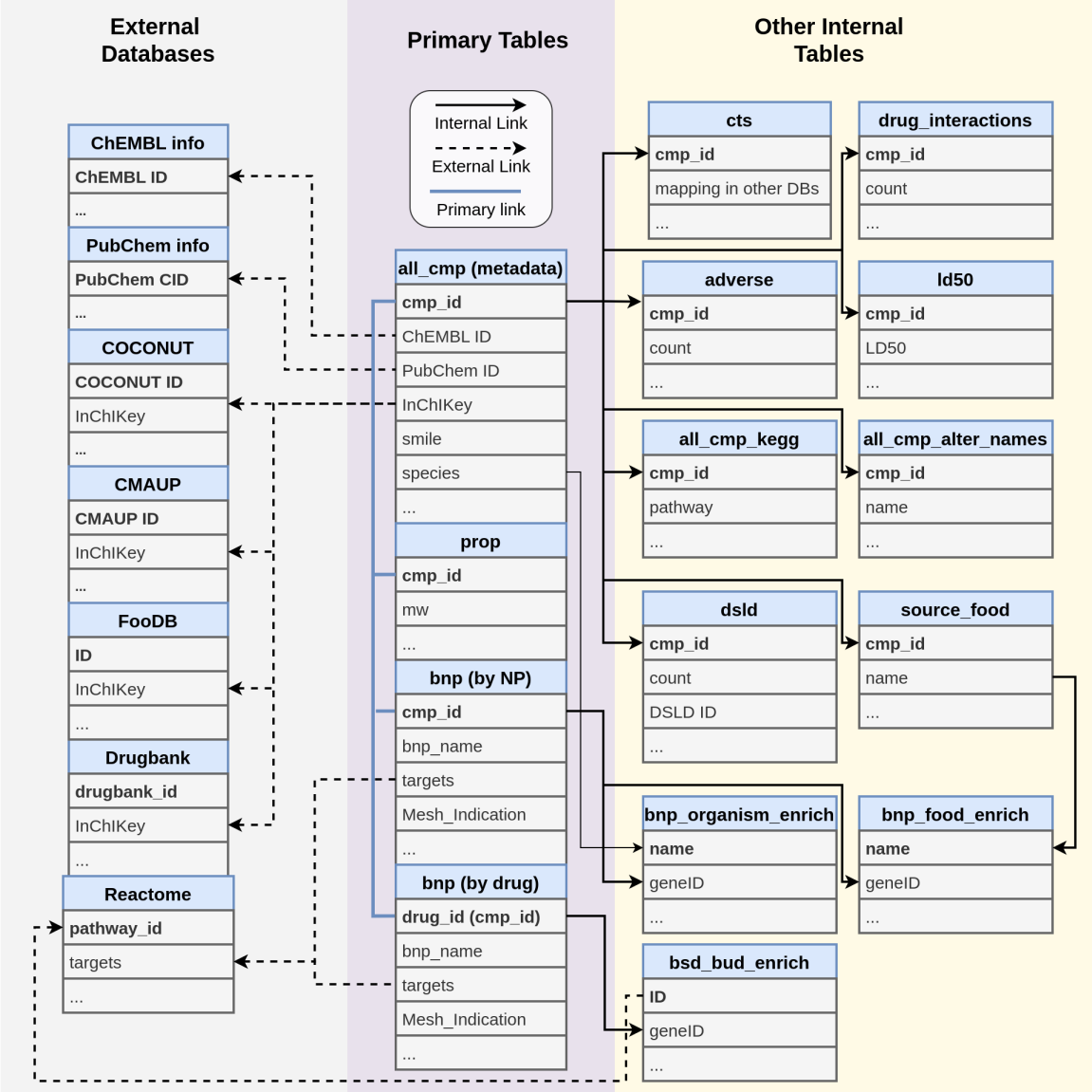

Figure 4.14: Results in a Database. All results from this research are collected into an SQL database. Tables generated from this research are located in the Primary Tables and Other Internal Tables columns. The primary key of each table is in bold. Connections between tables are marked in solid arrows. Connections to outside sources are marked in dashed lines.

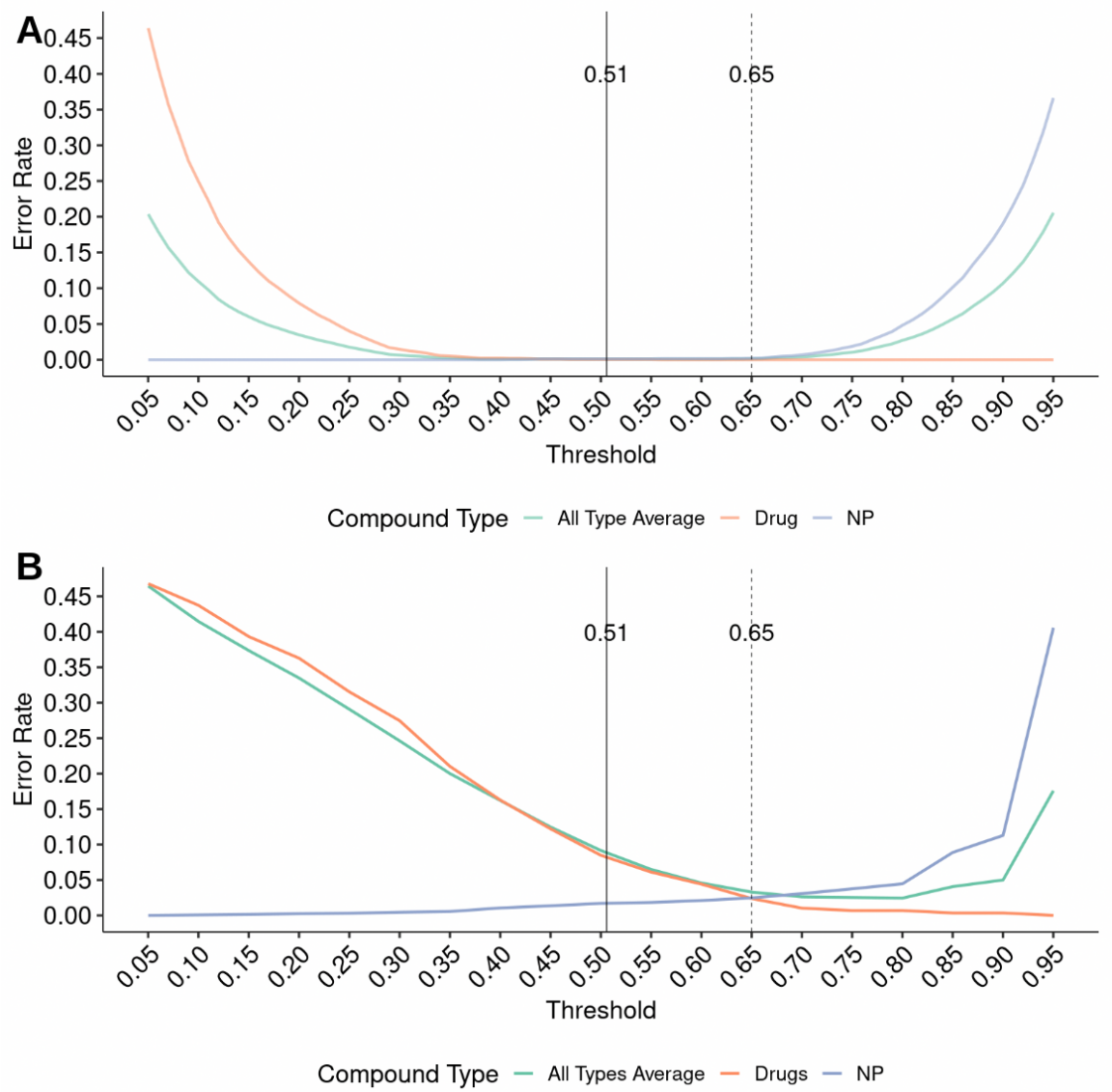

Figure 4.15: Determination of the threshold of RF model. (A) The cross-validation error rate on predicting drugs, NPs, and both on average. (B) The independent test dataset error rate. The solid line indicates the optimal threshold for cross-validation. The dashed line indicates the threshold selected in this study for downstream analysis.

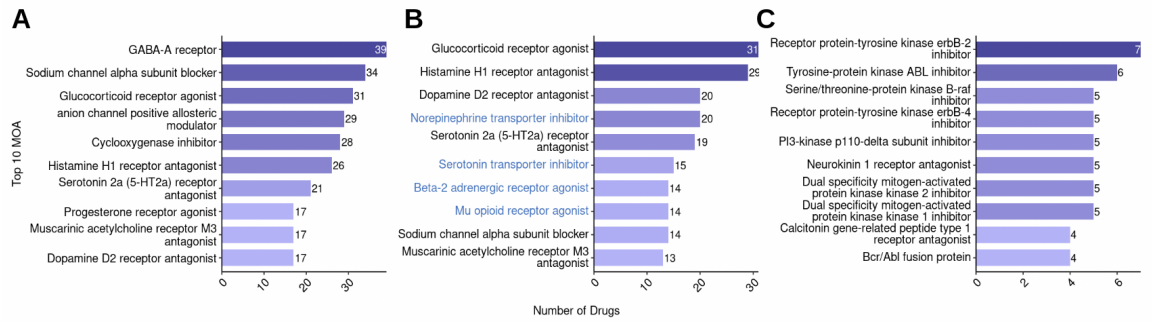

Figure 4.16: Top 10 MOA terms of BSDs (A), BUDs (B), and BUD exclusive terms (C). The blue terms in (A) are top 10 BUD terms that do not exist in top BSD terms.

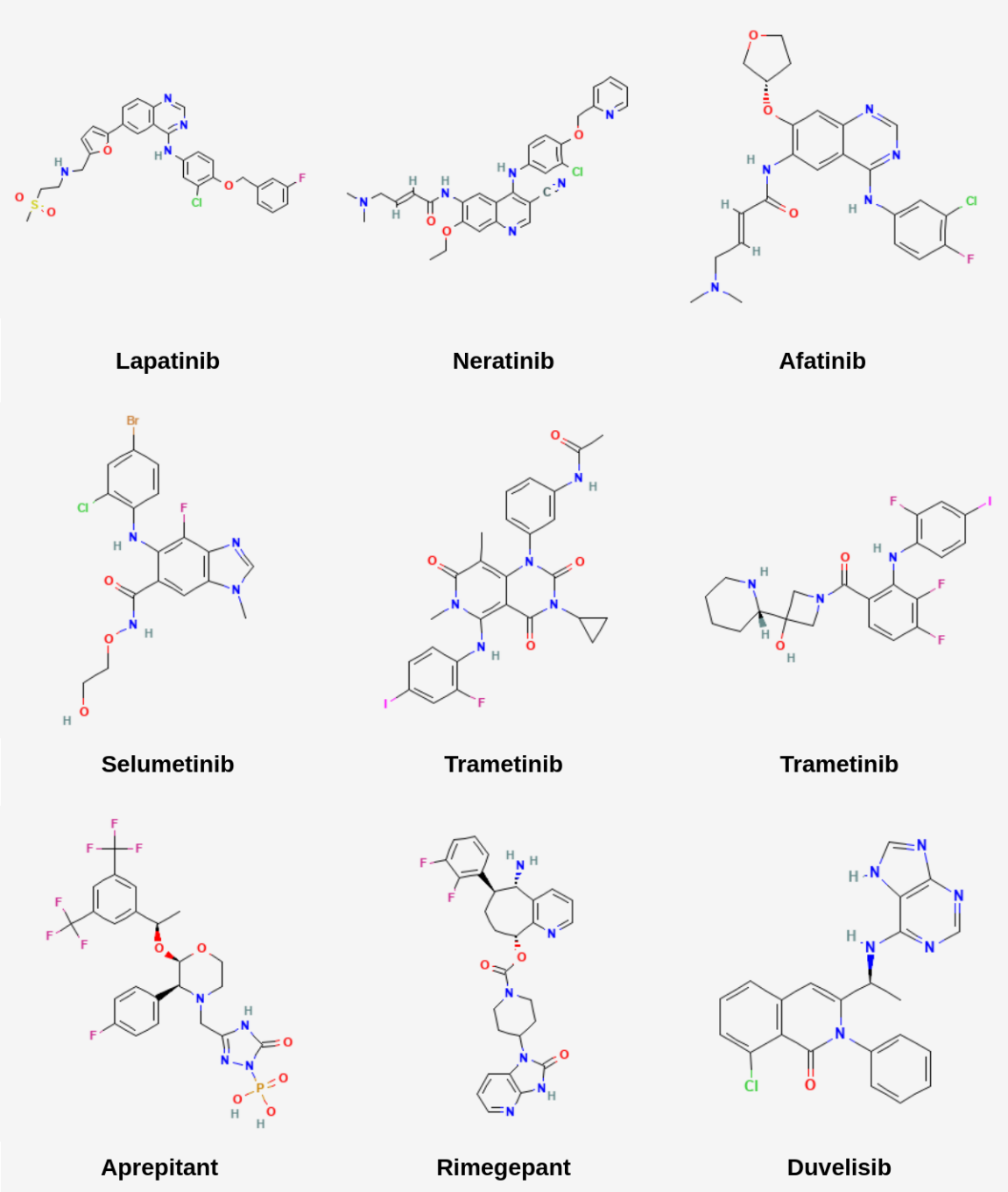

Figure 4.17: Nine typical examples of BUD structures

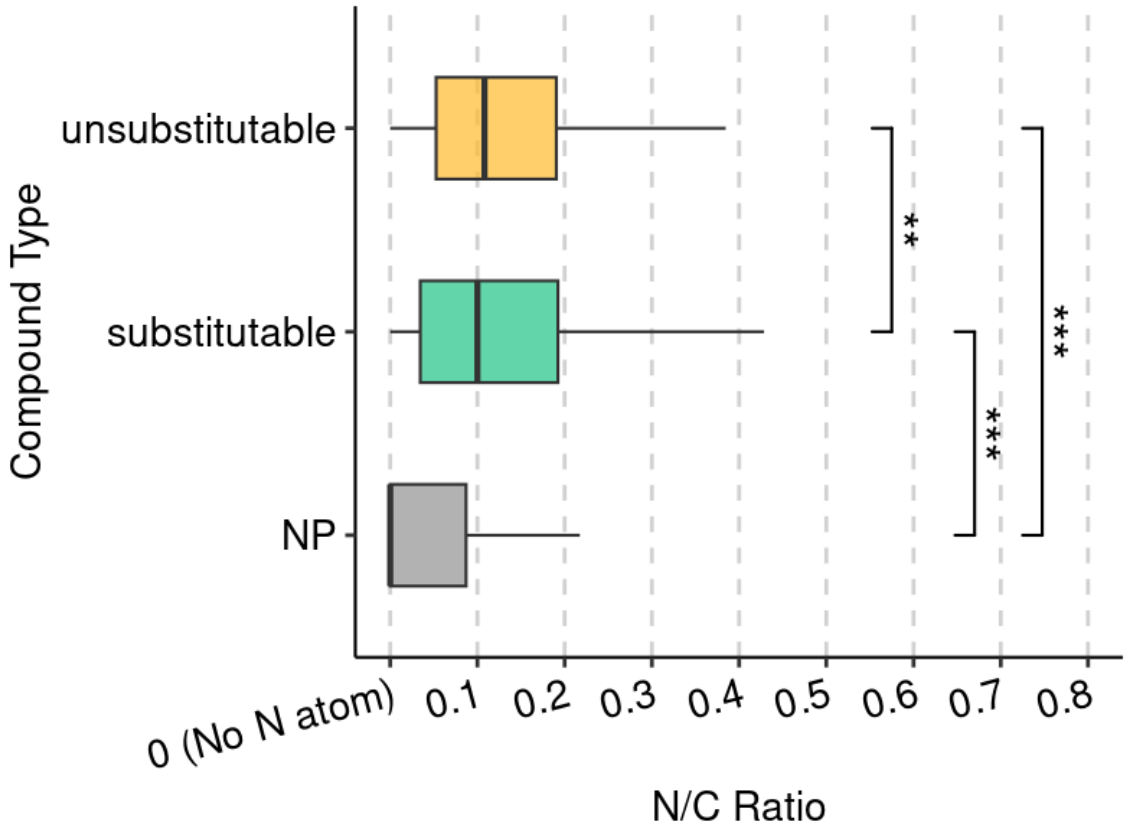

Figure 4.18: N/C ratio between NPs, BSDs (substitutable), and BUDs (substitutable).

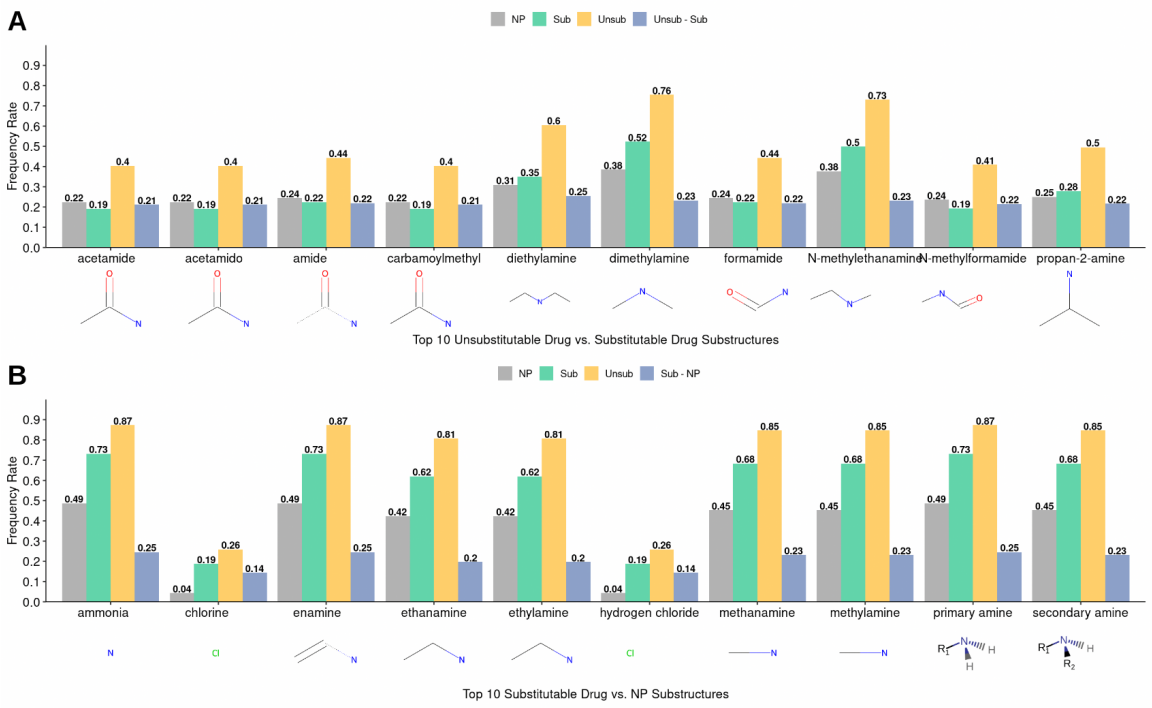

Figure 4.19: Top 10 frequency different substructures. (A) BUDs *vs.* BSDs. (A) BSDs *vs.* NPs. Both plots are sorted by the substructure frequency difference from left to right (blue column).

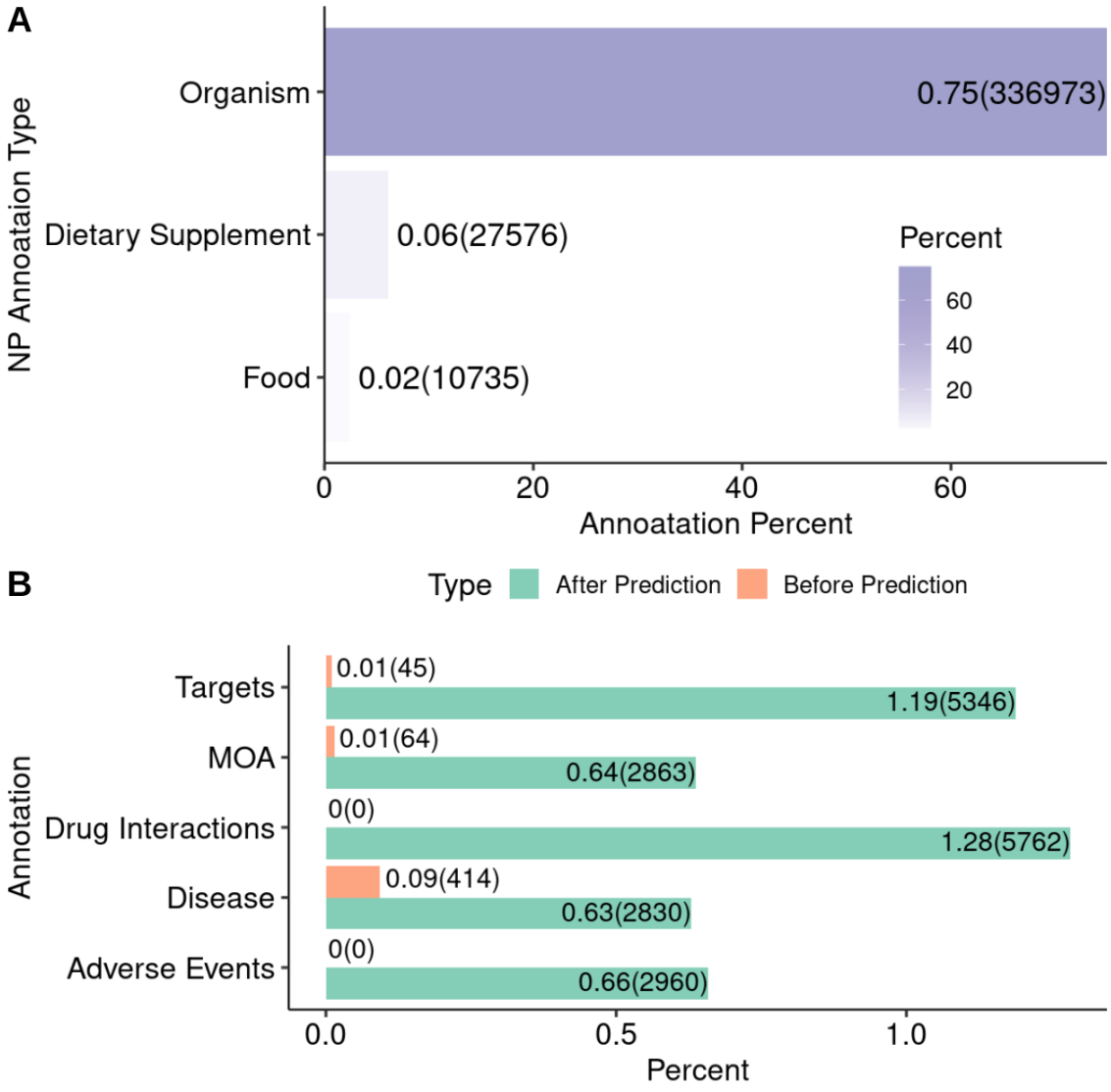

Figure 4.20: Annotation rate of NPs. (A) Source annotate rate of NPs. (B) Functional annotation rate of NPs.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this dissertation, the systemPipe workflow toolkits, including SPR, SPS, and their supporting packages, are developed. The object-oriented methods in SPR provide a new way to design, execute, visualize, and report end-to-end workflows systemically. The functionalities of SPR create a robust environment for automated, reproducible, and scalable workflow analysis. Combining the GUI from SPS, a broad spectrum of researchers, even those with no programming experience, can easily handle data analysis workflows. It is reasonable to believe that systemPipe tools will gain ground in future biological/bioinformatic research.

The large-scale prediction of drug alternative NPs is established in Chapter 4. It allows me to obtain a list of drug alternatives from natural sources. Meanwhile, meaningful annotations of NPs can be associated through the linkage, such as targets, pathways, and diseases. All results are packaged into a rational database for future researchers to query.

It is anticipated that this prediction can serve as guidance for future biological experiments and drug discovery. The results can also be used as a healthy diet tool to query and evaluate the bio-functionalities of given diets.

5.2 Future Work

As the advancements in sequencing, spectrometers, and other related technologies in the biological/biomedical field, more data workflows will be developed. These workflow templates will certainly be demanded in systemPipe toolkits. One direction is to provide additional templates. A better choice is to continue developing the workflow template creation functionalities, thereby providing a user-friendly interface to support new custom templates.

Adding a GUI can be a good future direction for the drug discovery part, even though I have summarized results in a concise format. Similar to the story of SPR and SPS, one cannot require all users to be familiar with the command-line interface. In this case, it is the knowledge to query an SQLite database. Similar databases that have been mentioned in this dissertation, like PubChem, ChEMBL, and COCONUT, all have a web interface. Developing a web-based GUI, therefore, will be a critical next step.

Bibliography

Abduljawad, E. A. 2020, 'Review of some evidenced medicinal activities of acacia nilotica', `https://archivepp.com/storage/models/article/fMGOI34ZdE86zDWf0Ks2o0ZL1DWlV2P6InRhIRxlk3uSfaBlH74mYQkVGFAj/review-of-some-evidenced-medicinal-activities-of-acacia-nilotica.pdf`. Accessed: 2023-4-11.

Afgan, E., Baker, D., Batut, B., van den Beek, M., Bouvier, D., Cech, M., Chilton, J., Clements, D., Coraor, N., Grüning, B. A., Guerler, A., Hillman-Jackson, J., Hiltemann, S., Jalili, V., Rasche, H., Soranzo, N., Goecks, J., Taylor, J., Nekrutenko, A. and Blankenberg, D. 2018, 'The galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update', *Nucleic Acids Res.* **46**(W1), W537–W544.
URL: *http://dx.doi.org/10.1093/nar/gky379*

Åhlberg, M. K. 2021, 'A profound explanation of why eating green (wild) edible plants promote health and longevity', *Food Front.* **2**(3), 240–267.

Al-Snafi, A. E. 2019, 'Fritillaria imperialis-a review', *IOSR Journal of pharmacy* **9**(3), 47–51.

Almatroodi, S. A., Alsahli, M. A., Almatroudi, A. and Rahmani, A. H. 2019, 'Garlic and its active compounds: A potential candidate in the prevention of cancer by modulating various cell signalling pathways', *Anticancer Agents Med. Chem.* **19**(11), 1314–1324.

Amstutz, P., Crusoe, M. R., Tijanić, N., Chapman, B., Chilton, J., Heuer, M., Kartashov, A., Leehr, D., Ménager, H., Nedeljkovich, M., Scales, M., Soiland-Reyes, S. and Stojanovic, L. 2016, 'Common workflow language, v1.0'.
URL: *https://escholarship.org/uc/item/25z538jj*

Ao, J. and Li, B. 2013, 'Stability and antioxidative activities of casein peptide fractions during simulated gastrointestinal digestion in vitro: Charge properties of peptides affect digestive stability', *Food Res. Int.* **52**(1), 334–341.

Atanasov, A. G., Waltenberger, B., Pferschy-Wenzig, E.-M., Linder, T., Wawrosch, C., Uhrin, P., Temml, V., Wang, L., Schwaiger, S., Heiss, E. H., Rollinger, J. M., Schuster, D., Breuss, J. M., Bochkov, V., Mihovilovic, M. D., Kopp, B., Bauer, R., Dirsch, V. M. and Stuppner, H. 2015, 'Discovery and resupply of pharmacologically active plant-derived natural products: A review', *Biotechnol. Adv.* **33**(8), 1582–1614.

Ayon, N. J. 2023, 'High-Throughput screening of natural product and synthetic molecule libraries for antibacterial drug discovery', *Metabolites* **13**(5).

Bajusz, D., Rácz, A. and Héberger, K. 2015, 'Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?', *J. Cheminform.* **7**, 20.

Balamurugan, R., Dekker, F. J. and Waldmann, H. 2005, 'Design of compound libraries based on natural product scaffolds and protein structure similarity clustering (PSSC)', *Mol. Biosyst.* **1**(1), 36–45.

Banerjee, P., Erehman, J., Gohlke, B.-O., Wilhelm, T., Preissner, R. and Dunkel, M. 2014, 'Super natural II—a database of natural products', *Nucleic Acids Res.* **43**(D1), D935–D939.

Barardo, D., Thornton, D., Thoppil, H., Walsh, M., Sharifi, S., Ferreira, S., Anžič, A., Fernandes, M., Monteiro, P., Grum, T., Cordeiro, R., De-Souza, E. A., Budovsky, A., Araujo, N., Gruber, J., Petrascheck, M., Fraifeld, V. E., Zhavoronkov, A., Moskalev, A. and de Magalhães, J. P. 2017, 'The DrugAge database of aging-related drugs', *Aging Cell* **16**(3), 594–597.

Barrett, T., Wilhite, S. E., Ledoux, P., Evangelista, C., Kim, I. F., Tomashevsky, M., Marshall, K. A., Phillippy, K. H., Sherman, P. M., Holko, M., Yefanov, A., Lee, H., Zhang, N., Robertson, C. L., Serova, N., Davis, S. and Soboleva, A. 2013, 'NCBI GEO: archive for functional genomics data sets–update', *Nucleic Acids Res.* **41**(Database issue), D991–5.

Bauer, M., Klau, G. W. and Reinert, K. 2007, 'Accurate multiple sequence-structure alignment of RNA sequences using combinatorial optimization.', *BMC Bioinformatics* **8**, 271. **URL:** *http://dx.doi.org/10.1186/1471-2105-8-271*

Berdy, J. and Kertesz, M. 1989, Bioactive natural products database: an aid for natural products identification, *in* 'Chemical Information', Springer Berlin Heidelberg, pp. 237–251.

Bhullar, K. S., Lagarón, N. O., McGowan, E. M., Parmar, I., Jha, A., Hubbard, B. P. and Rupasinghe, H. P. V. 2018, 'Kinase-targeted cancer therapies: progress, challenges and future directions', *Mol. Cancer* **17**(1), 48.

Bødker, S. 2021, *Through the interface: A human activity approach to user interface design*, CRC Press.

Bui, T. T. and Nguyen, T. H. 2017, 'Natural product for the treatment of alzheimer's disease', *J. Basic Clin. Physiol. Pharmacol.* **28**(5), 413–423.

Carhart, R. E., Smith, D. H. and Venkataraghavan, R. 1985, 'Atom pairs as molecular features in structure-activity studies: definition and applications', *J. Chem. Inf. Comput. Sci.* **25**(2), 64–73.

Castellano-Escuder, P., González-Domínguez, R., Carmona-Pontaque, F., Andrés-Lacueva,

C. and Sánchez-Pla, A. 2021, 'POMAShiny: A user-friendly web-based workflow for metabolomics and proteomics data analysis', *PLoS Comput. Biol.* **17**(7), e1009148.

Chan, M. M., Fong, D., Ho, C. T. and Huang, H. I. 1997, 'Inhibition of inducible nitric oxide synthase gene expression and enzyme activity by epigallocatechin gallate, a natural product from green tea', *Biochem. Pharmacol.* **54**(12), 1281–1286.

Chang, W., Cheng, J., Allaire, J. J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A. and Borges, B. 2021, 'shiny: Web application framework for R'.

Che, C.-T., Wong, M. S. and Lam, C. W. K. 2016, 'Natural products from chinese medicines with potential benefits to bone health', *Molecules* **21**(3), 239.

Colegate, S. M. and Molyneux, R. J. 2007, *Bioactive Natural Products: Detection, Isolation, and Structural Determination, Second Edition*, CRC Press.

Constantin, A.-E. and Patil, I. 2021, 'ggsignif: R package for displaying significance brackets for 'ggplot2''.

Cooper, R., Morré, D. J. and Morré, D. M. 2005, 'Medicinal benefits of green tea: part II. review of anticancer properties', *J. Altern. Complement. Med.* **11**(4), 639–652.

Corrêa, R. C. G., Heleno, S. A., Alves, M. J. and Ferreira, I. C. F. R. 2020, 'Bacterial resistance: Antibiotics of last generation used in clinical practice and the arise of natural products as new therapeutic alternatives', *Curr. Pharm. Des.* **26**(8), 815–837.

Cox, K. H. M., Pipingas, A. and Scholey, A. B. 2015, 'Investigation of the effects of solid lipid curcumin on cognition and mood in a healthy older population', *J. Psychopharmacol.* **29**(5), 642–651.

Cramer, R. D., Jilek, R. J. and Andrews, K. M. 2002, 'Dbtop: topomer similarity searching of conventional structure databases', *J. Mol. Graph. Model.* **20**(6), 447–462.

Crusoe, M. R., Abeln, S., Iosup, A., Amstutz, P., Chilton, J., Tijanić, N., Ménager, H., Soiland-Reyes, S., Gavrilovic, B. and Goble, C. 2021, 'Methods included: Standardizing computational reuse and portability with the common workflow language'.

Deelman, E., Vahi, K., Juve, G., Rynge, M., Callaghan, S., Maechling, P. J., Mayani, R., Chen, W., Ferreira da Silva, R., Livny, M. and Wenger, K. 2015, 'Pegasus, a workflow management system for science automation', *Future Gener. Comput. Syst.* **46**, 17–35.
URL: *https://www.sciencedirect.com/science/article/pii/S0167739X14002015*

Desvillechabrol, D., Legendre, R., Rioualen, C., Bouchier, C., van Helden, J., Kennedy, S. and Cokelaer, T. 2018, 'Sequanix: a dynamic graphical interface for snakemake workflows', *Bioinformatics* **34**(11), 1934–1936.

Di Tommaso, P., Chatzou, M., Floden, E. W., Barja, P. P., Palumbo, E. and Notredame, C. 2017, 'Nextflow enables reproducible computational workflows', *Nat. Biotechnol.*

35(4), 316–319.
URL: *http://dx.doi.org/10.1038/nbt.3820*

Dietary Supplement Health and Education Act 1994.

Dorant, E., van den Brandt, P. A., Goldbohm, R. A., Hermus, R. J. and Sturmans, F. 1993, 'Garlic and its significance for the prevention of cancer in humans: a critical view', *Br. J. Cancer* **67**(3), 424–429.

Dossey, L. 2002, 'Longevity', *Altern. Ther. Health Med.* **8**(3), 12–16, 125–34.

Duan, Y., Evans, D. S., Miller, R. A., Schork, N. J., Cummings, S. R. and Girke, T. 2020, 'signaturesearch: environment for gene expression signature searching and functional interpretation', *Nucleic Acids Res.* **48**(21), e124.

Efferth, T., Fu, Y.-J., Zu, Y.-G., Schwarz, G., Konkimalla, V. S. B. and Wink, M. 2007, 'Molecular target-guided tumor therapy with natural products derived from traditional chinese medicine', *Curr. Med. Chem.* **14**(19), 2024–2032.

Esghaei, M., Ghaffari, H., Rahimi Esboei, B., Ebrahimi Tapeh, Z., Bokharaei Salim, F. and Motevalian, M. 2018, 'Evaluation of anticancer activity of camellia sinensis in the caco-2 colorectal cancer cell line', *Asian Pac. J. Cancer Prev.* **19**(6), 1697–1701.

Everitt, A. V., Heilbronn, L. K. and Le Couteur, D. G. 2010, Food intake, life style, aging and human longevity, *in* A. V. Everitt, S. I. S. Rattan, D. G. le Couteur and R. de Cabo, eds, 'Calorie Restriction, Aging and Longevity', Springer Netherlands, Dordrecht, pp. 15–41.

Feng, S.-T., Wang, Z.-Z., Yuan, Y.-H., Sun, H.-M., Chen, N.-H. and Zhang, Y. 2019, 'Mangiferin: A multipotent natural product preventing neurodegeneration in alzheimer's and parkinson's disease models', *Pharmacol. Res.* **146**, 104336.

Fernández-Bedmar, Z., Anter, J., de La Cruz-Ares, S., Muñoz-Serrano, A., Alonso-Moraga, A. and Pérez-Guisado, J. 2011, 'Role of citrus juices and distinctive components in the modulation of degenerative processes: genotoxicity, antigenotoxicity, cytotoxicity, and longevity in drosophila', *J. Toxicol. Environ. Health A* **74**(15-16), 1052–1066.

Fjukstad, B. and Bongo, L. A. 2017, 'A review of scalable bioinformatics pipelines', *Data Science and Engineering* **2**(3), 245–251.

Fleming, J. 1998, 'Web navigation: Designing the user experience', `http://jepelet.free.fr/studies/MBA/design/s4/lectures/Web%20Navigation%20Designing%20the%20User%20Experience.pdf`. Accessed: 2023-2-1.

Friedrich, J., Dandekar, T., Wolf, M. and Müller, T. 2005, 'ProfDist: a tool for the construction of large phylogenetic trees based on profile distances.', *Bionformatics* **21**(9), 2108–2109.
URL: *http://dx.doi.org/10.1093/bioinformatics/bti289*

Gabrielson, S. W. 2018, 'SciFinder', *J. Med. Libr. Assoc.* **106**(4).

Gallo, K., Goede, A., Preissner, R. and Gohlke, B.-O. 2022, 'SuperPred 3.0: drug classification and target prediction-a machine learning approach', *Nucleic Acids Res.* **50**(W1), W726–31.

Gascuel, O. 1997, 'BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data.', *Mol Biol Evol* **14**(7), 685–695.

Gaulton, A., Hersey, A., Nowotka, M., Bento, A. P., Chambers, J., Mendez, D., Mutowo, P., Atkinson, F., Bellis, L. J., Cibrián-Uhalte, E., Davies, M., Dedman, N., Karlsson, A., Magariños, M. P., Overington, J. P., Papadatos, G., Smit, I. and Leach, A. R. 2017, 'The ChEMBL database in 2017', *Nucleic Acids Res.* **45**(D1), D945–D954.

Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Leisch, F., Li, C., Maechler, M., Rossini, A. J., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J. Y. H. and Zhang, J. 2004, 'Bioconductor: open software development for computational biology and bioinformatics', *Genome Biol.* **5**(10), R80.
URL: *http://dx.doi.org/10.1186/gb-2004-5-10-r80*

Gerlach, D., Wolf, M., Dandekar, T., Müller, T., Pokorny, A. and Rahmann, S. 2007, 'Deep metazoan phylogeny.', *In Silico Biol* **7**(2), 151–154.

Gillespie, M., Jassal, B., Stephan, R., Milacic, M., Rothfels, K., Senff-Ribeiro, A., Griss, J., Sevilla, C., Matthews, L., Gong, C., Deng, C., Varusai, T., Ragueneau, E., Haider, Y., May, B., Shamovsky, V., Weiser, J., Brunson, T., Sanati, N., Beckman, L., Shao, X., Fabregat, A., Sidiropoulos, K., Murillo, J., Viteri, G., Cook, J., Shorser, S., Bader, G., Demir, E., Sander, C., Haw, R., Wu, G., Stein, L., Hermjakob, H. and D'Eustachio, P. 2022, 'The reactome pathway knowledgebase 2022', *Nucleic Acids Res.* **50**(D1), D687–D692.

Gold, L. T. and Masson, G. R. 2022, 'GCN2: roles in tumour development and progression', *Biochem. Soc. Trans.* **50**(2), 737–745.

Gonçalves, P. B. and Romeiro, N. C. 2019, 'Multi-target natural products as alternatives against oxidative stress in chronic obstructive pulmonary disease (COPD)', *Eur. J. Med. Chem.* **163**, 911–931.

González-Manzano, S. and Dueñas, M. 2021, 'Applications of natural products in food', *Foods* **10**(2).

Goodwin, S., McPherson, J. D. and McCombie, W. R. 2016, 'Coming of age: ten years of next-generation sequencing technologies', *Nat. Rev. Genet.* **17**(6), 333–351.
URL: *http://dx.doi.org/10.1038/nrg.2016.49*

Grajales, A., Aguilar, C. and Sánchez, J. A. 2007, 'Phylogenetic reconstruction using sec-

ondary structures of internal transcribed spacer 2 (its2, rdna): finding the molecular and morphological gap in caribbean gorgonian corals.', *BMC Evol Biol* **7**, 90.
URL: *http://dx.doi.org/10.1186/1471-2148-7-90*

Guha, R. 2007, 'Chemical informatics functionality in R', *J. Stat. Softw.* **18**, 1–16.

Gupta, S., Singh, M. and Madan, A. K. 2002, 'Eccentric distance sum: A novel graph invariant for predicting biological and physical properties', *J. Math. Anal. Appl.* **275**(1), 386–401.

H Backman, T. W. and Girke, T. 2016*a*, 'systemPipeR: NGS workflow and report generation environment', *BMC Bioinformatics* **17**, 388.
URL: *http://dx.doi.org/10.1186/s12859-016-1241-0*

H Backman, T. W. and Girke, T. 2016*b*, 'systemPipeR: NGS workflow and report generation environment', *BMC Bioinformatics* **17**, 388.

Harvey, A. L. 2008, 'Natural products in drug discovery', *Drug Discov. Today* **13**(19-20), 894–901.

Heller, S. R., McNaught, A., Pletnev, I., Stein, S. and Tchekhovskoi, D. 2015, 'InChI, the IUPAC international chemical identifier', *J. Cheminform.* **7**, 23.

Hicks, S. C. and Peng, R. D. 2019, 'Evaluating the success of a data analysis'.
URL: *http://arxiv.org/abs/1904.11907*

Hochsmann, M., Hochsmann, M., Voss, B. and Giegerich, R. 2004, 'Pure multiple RNA secondary structure alignments: a progressive profile approach', *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **1**(1), 53–62.

Hothorn, T. and Leisch, F. 2011, 'Case studies in reproducibility', *Brief. Bioinform.* **12**(3), 288–300.
URL: *http://dx.doi.org/10.1093/bib/bbq084*

Huber, W., Carey, V. J., Gentleman, R., Anders, S., Carlson, M., Carvalho, B. S., Bravo, H. C., Davis, S., Gatto, L., Girke, T., Gottardo, R., Hahne, F., Hansen, K. D., Irizarry, R. A., Lawrence, M., Love, M. I., MacDonald, J., Obenchain, V., Oleś, A. K., Pagès, H., Reyes, A., Shannon, P., Smyth, G. K., Tenenbaum, D., Waldron, L. and Morgan, M. 2015*a*, 'Orchestrating high-throughput genomic analysis with bioconductor', *Nat. Methods* **12**(2), 115–121.
URL: *http://dx.doi.org/10.1038/nmeth.3252*

Huber, W., Carey, V. J., Gentleman, R., Anders, S., Carlson, M., Carvalho, B. S., Bravo, H. C., Davis, S., Gatto, L., Girke, T., Gottardo, R., Hahne, F., Hansen, K. D., Irizarry, R. A., Lawrence, M., Love, M. I., MacDonald, J., Obenchain, V., Oleś, A. K., Pagès, H., Reyes, A., Shannon, P., Smyth, G. K., Tenenbaum, D., Waldron, L. and Morgan, M. 2015*b*, 'Orchestrating high-throughput genomic analysis with bioconductor', *Nat. Meth-*

ods **12**(2), 115–121.

Hudelot, C., Gowri-Shankar, V., Jow, H., Rattray, M. and Higgs, P. G. 2003, 'RNA-based phylogenetic methods: application to mammalian mitochondrial RNA sequences.', *Mol Phylogenet Evol* **28**(2), 241–252.

Inc., P. T. 2015, 'Collaborative data science'.
URL: *https://plot.ly*

Jalili, V., Afgan, E., Gu, Q., Clements, D., Blankenberg, D., Goecks, J., Taylor, J. and Nekrutenko, A. 2020, 'The galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2020 update', *Nucleic Acids Res.* **48**(W1), W395–W402.

Jamet, E. and Fernandez, J. 2016, 'Enhancing interactive tutorial effectiveness through visual cueing', *Educ. Technol. Res. Dev.* **64**(4), 631–641.

Jeong, W. Y., Jin, J. S., Cho, Y. A., Lee, J. H., Park, S., Jeong, S. W., Kim, Y.-H., Lim, C.-S., Abd El-Aty, A. M., Kim, G.-S., Lee, S. J., Shim, J.-H. and Shin, S. C. 2011, 'Determination of polyphenols in three capsicum annuum l. (bell pepper) varieties using high-performance liquid chromatography-tandem mass spectrometry: their contribution to overall antioxidant and anticancer activity', *J. Sep. Sci.* **34**(21), 2967–2974.

Ji, H.-F. and Shen, L. 2011, 'Berberine: a potential multipotent natural product to combat alzheimer's disease', *Molecules* **16**(8), 6732–6740.

Jia, L., Yao, W., Jiang, Y., Li, Y., Wang, Z., Li, H., Huang, F., Li, J., Chen, T. and Zhang, H. 2021, 'Development of interactive biological web applications with R/Shiny', *Brief. Bioinform.* .

Jow, H., Hudelot, C., Rattray, M. and Higgs, P. G. 2002, 'Bayesian phylogenetics using an rna substitution model applied to early mammalian evolution.', *Mol Biol Evol* **19**(9), 1591–1601.

Jukes, T. and Cantor, C. 1969, Evolution of protein molecules, *in* H. Munro, ed., 'Mammalian Protein Metabolism', Academic Press, New York, USA, pp. 21–132.

Kalmarzi, R. N., Naleini, S. N., Ashtary-Larky, D., Peluso, I., Jouybari, L., Rafi, A., Ghorat, F., Heidari, N., Sharifian, F., Mardaneh, J., Aiello, P., Helbi, S. and Kooti, W. 2019, 'Anti-Inflammatory and immunomodulatory effects of barberry (berberis vulgaris) and its main compounds', *Oxid. Med. Cell. Longev.* **2019**, 6183965.

Kassambara, A. 2022, 'ggpubr: 'ggplot2' based publication ready plots'.

Kaushik, G., Ivkovic, S., Simonovic, J., Tijanic, N., Davis-Dusenbery, B. and Kural, D. 2017, 'RABIX: AN OPEN-SOURCE WORKFLOW EXECUTOR SUPPORTING RECOMPUTABILITY AND INTEROPERABILITY OF WORKFLOW DESCRIPTIONS', *Pac. Symp. Biocomput.* **22**, 154–165.

Kevin Horan, T. G. 2022, 'ChemmineOB: OpenBabel wrapper package for R'.

Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B. A., Thiessen, P. A., Yu, B., Zaslavsky, L., Zhang, J. and Bolton, E. E. 2021, 'PubChem in 2021: new data content and improved web interfaces', *Nucleic Acids Res.* **49**(D1), D1388–D1395.

Kingston, D. G. I. 2011, 'Modern natural products drug discovery and its relevance to biodiversity conservation', *J. Nat. Prod.* **74**(3), 496–511.

Koch, M. A. and Waldmann, H. 2005, 'Protein structure similarity clustering and natural product structure as guiding principles in drug discovery', *Drug Discov. Today* **10**(7), 471–483.

Kokubo, Y., Iso, H., Saito, I., Yamagishi, K., Yatsuya, H., Ishihara, J., Inoue, M. and Tsugane, S. 2013, 'The impact of green tea and coffee consumption on the reduced risk of stroke incidence in japanese population: the japan public health center-based study cohort', *Stroke* **44**(5), 1369–1374.

Köster, J. and Rahmann, S. 2012, 'Snakemake–a scalable bioinformatics workflow engine', *Bioinformatics* **28**(19), 2520–2522.
URL: *http://dx.doi.org/10.1093/bioinformatics/bts480*

Kucukural, A., Yukselen, O., Ozata, D. M., Moore, M. J. and Garber, M. 2019, 'DEBrowser: interactive differential expression analysis and visualization tool for count data', *BMC Genomics* **20**(1), 6.

Lachance, H., Wetzel, S., Kumar, K. and Waldmann, H. 2012, 'Charting, navigating, and populating natural product chemical space for drug discovery', *J. Med. Chem.* **55**(13), 5989–6001.

Lam, M., Khoshkhat, P., Chamani, M., Shahsavari, S., Dorkoosh, F. A., Rajabi, A., Maniruzzaman, M. and Nokhodchi, A. 2022, 'In-depth multidisciplinary review of the usage, manufacturing, regulations & market of dietary supplements', *J. Drug Deliv. Sci. Technol.* **67**, 102985.

Landrum, G., Tosco, P., Kelley, B., sriniker, gedeck, NadineSchneider, Vianello, R., Ric, Dalke, A., Cole, B., AlexanderSavelyev, Swain, M., Turk, S., Dan, N., Vaucher, A., Kawashima, E., Wójcikowski, M., Probst, D., Godin, G., Cosgrove, D., Pahl, A., JP, Berenger, F., strets, JLVarjo, O'Boyle, N., Fuller, P., Jensen, J. H., Sforna, G. and DoliathGavid 2020, 'rdkit/rdkit: 2020_03_1 (q1 2020) release'.

Lau, B. H. S., Tadi, P. P. and Tosk, J. M. 1990, 'Allium sativum (garlic) and cancer prevention', *Nutr. Res.* **10**(8), 937–948.

Lavecchia, A. 2015, 'Machine-learning approaches in drug discovery: methods and applications', *Drug Discov. Today* **20**(3), 318–331.

Ledell, E. and Poirier, S. n.d., 'H2O AutoML: Scalable automatic machine learning'. Accessed: 2022-12-19.

Lew, P., Olsina, L. and Zhang, L. 2010, Quality, quality in use, actual usability and user experience as key drivers for web application evaluation, *in* 'Web Engineering', Springer Berlin Heidelberg, pp. 218–232.

Li, G. and Lou, H.-X. 2018, 'Strategies to diversify natural products for drug discovery', *Med. Res. Rev.* **38**(4), 1255–1294.

Liberti, L., Lavor, C., Maculan, N. and Mucherino, A. 2014, 'Euclidean distance geometry and applications', *SIAM Rev.* **56**(1), 3–69.

Lightbody, G., Haberland, V., Browne, F., Taggart, L., Zheng, H., Parkes, E. and Blayney, J. K. 2019, 'Review of applications of high-throughput sequencing in personalized medicine: barriers and facilitators of future progress in research and clinical application', *Brief. Bioinform.* **20**(5), 1795–1811.

Lim, S. 2018, 'Eating a balanced diet: A healthy life through a balanced diet in the age of longevity', *J Obes Metab Syndr* **27**(1), 39–45.

Liu, J., Peng, L., Huang, W., Li, Z., Pan, J., Sang, L., Lu, S., Zhang, J., Li, W. and Luo, Y. 2017, 'Balancing between aging and cancer: Molecular genetics meets traditional chinese medicine', *J. Cell. Biochem.* **118**(9), 2581–2586.

Lo, Y.-C., Rensi, S. E., Torng, W. and Altman, R. B. 2018, 'Machine learning in chemoinformatics and drug discovery', *Drug Discov. Today* **23**(8), 1538–1546.

Lo, Y.-H., Chen, Y.-J., Chung, T.-Y., Lin, N.-H., Chen, W.-Y., Chen, C.-Y., Lee, M.-R., Chou, C.-C. and Tzen, J. T. C. 2015, 'Emoghrelin, a unique emodin derivative in heshouwu, stimulates growth hormone secretion via activation of the ghrelin receptor', *J. Ethnopharmacol.* **159**, 1–8.

Love, M. I., Huber, W. and Anders, S. 2014, 'Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2', *Genome Biol.* **15**(12), 550.

Maeda, M. H. and Kondo, K. 2013, 'Three-dimensional structure database of natural metabolites (3DMET): a novel database of curated 3D structures', *J. Chem. Inf. Model.* **53**(3), 527–533.

Malik, S., Cusidó, R. M., Mirjalili, M. H., Moyano, E., Palazón, J. and Bonfill, M. 2011, 'Production of the anticancer drug taxol in taxus baccata suspension cultures: A review', *Process Biochem.* **46**(1), 23–34.

Marco, L. and Carreiras, M. C. 2006, 'Galanthamine, a natural product for the treatment of alzheimer's disease'.

Marik, P. E. and Flemmer, M. 2012, 'Do dietary supplements have beneficial health ef-

fects in industrialized nations: what is the evidence?', *JPEN J. Parenter. Enteral Nutr.* **36**(2), 159–168.

Martin, T. 2016, 'Toxicity estimation software tool (TEST) U.S. environmental protection agency'.

Martin, Y. C., Kofron, J. L. and Traphagen, L. M. 2002, 'Do structurally similar molecules have similar biological activity?', *J. Med. Chem.* **45**(19), 4350–4358.

Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J. K., Ceulemans, H., Clevert, D.-A. and Hochreiter, S. 2018, 'Large-scale comparison of machine learning methods for drug target prediction on ChEMBL', *Chem. Sci.* **9**(24), 5441–5451.

Mazzio, E., Badisa, R., Mack, N., Deiab, S. and Soliman, K. F. A. 2014, 'High throughput screening of natural products for anti-mitotic effects in MDA-MB-231 human breast carcinoma cells', *Phytother. Res.* **28**(6), 856–867.

Meyer, F. and Perrier, V. 2021, 'esquisse: Explore and visualize your data interactively'.

Morgan, M., Obenchain, V., Hester, J. and Pagès, H. 2021, 'SummarizedExperiment: SummarizedExperiment container'.

Mouselimis, L. n.d., 'ClusterR: Gaussian mixture models, K-Means, Mini-Batch-Kmeans, K-Medoids and affinity propagation clustering; 2019. R package version 1.2. 1'.

Müllner, D. 2013, 'fastcluster: Fast hierarchical, agglomerative clustering routines for R and Python'.

Müller, T., Rahmann, S., Dandekar, T. and Wolf, M. 2004, 'Accurate and robust phylogeny estimation based on profile distances: a study of the Chlorophyceae (Chlorophyta).', *BMC Evol Biol* **4**, 20.
URL: *http://dx.doi.org/10.1186/1471-2148-4-20*

Müller, T., Spang, R. and Vingron, M. 2002, 'Estimating amino acid substitution models: a comparison of Dayhoff's estimator, the resolvent approach and a maximum likelihood method.', *Mol Biol Evol* **19**(1), 8–13.

Müller, T. and Vingron, M. 2000, 'Modeling amino acid replacement.', *J Comput Biol* **7**(6), 761–776.
URL: *http://dx.doi.org/10.1089/10665270050514918*

Narotzki, B., Reznick, A. Z., Aizenbud, D. and Levy, Y. 2012, 'Green tea: a promising natural product in oral health', *Arch. Oral Biol.* **57**(5), 429–435.

Nelson, J. W., Sklenar, J., Barnes, A. P. and Minnier, J. 2017, 'The START app: a web-based RNAseq analysis and visualization resource', *Bioinformatics* **33**(3), 447–449.

Niu, H., Fan, J. W., Wang, G. P., Wang, J., Chu, Y. B., Yang, Q. F., Wang, L. B. and

Tian, B. 2017, 'Anti-tumor effect of polysaccharides isolated from taraxacum mongolicum Hand-Mazz on MCF-7 human breast cancer cells', *Tropical Journal of Pharmaceutical Research* **16**(1), 83–89.

Normile, D. 2003, 'The new face of traditional chinese medicine', *Science* **299**(5604), 188–190.

Nouroz, F., Mehboob, M., Noreen, S., Zaidi, F. and Mobin, T. 2015, 'A review on anticancer activities of garlic (allium sativum l.)', *Middle East J Sci Res* **23**(6), 1145–1151.

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T. and Hutchison, G. R. 2011, 'Open babel: An open chemical toolbox', *J. Cheminform.* **3**, 33.

Oinn, T., Greenwood, M., Addis, M., Alpdemir, M. N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M. R., Senger, M., Stevens, R., Wipat, A. and Wroe, C. 2006, 'Taverna: lessons in creating a workflow environment for the life sciences', *Concurr. Comput.* **18**(10), 1067–1100.

Ou-Yang, S.-S., Lu, J.-Y., Kong, X.-Q., Liang, Z.-J., Luo, C. and Jiang, H. 2012, 'Computational drug discovery', *Acta Pharmacol. Sin.* **33**(9), 1131–1140.

Patwardhan, B., Warude, D., Pushpangadan, P. and Bhatt, N. 2005, 'Ayurveda and traditional chinese medicine: a comparative overview', *Evid. Based. Complement. Alternat. Med.* **2**(4), 465–473.

Pennington, L. D. and Moustakas, D. T. 2017, 'The necessary nitrogen atom: A versatile High-Impact design element for multiparameter optimization', *J. Med. Chem.* **60**(9), 3552–3579.

Petitjean, M. 1992, 'Applications of the radius-diameter diagram to the classification of topological and geometrical shapes of chemical compounds', *J. Chem. Inf. Comput. Sci.* **32**(4), 331–337.

Pletnev, I., Erin, A., McNaught, A., Blinov, K., Tchekhovskoi, D. and Heller, S. 2012, 'InChIKey collision resistance: an experimental testing', *J. Cheminform.* **4**(1), 39.

Podyacheva, E., Afanasyev, O. I., Vasilyev, D. V. and Chusov, D. 2022, 'Borrowing hydrogen amination reactions: A complex analysis of trends and correlations of the various reaction parameters', *ACS Catal.* **12**(12), 7142–7198.

Poleksic, A. and Xie, L. 2019, 'Database of adverse events associated with drugs and drug combinations', *Sci. Rep.* **9**(1), 20025.

Qian, P.-Y., Xu, Y. and Fusetani, N. 2010, 'Natural products as antifouling compounds: recent progress and future perspectives', *Biofouling* **26**(2), 223–234.

QuantStack 2019, 'And voilà! jupyter blog', `https://blog.jupyter.org/and-voil%C3%A0-f6a2c08a4a93`.

R Core Team 2021, 'R: A language and environment for statistical computing'.
URL: *https://www.R-project.org/*

Rahmann, S., Muller, T., Dandekar, T. and Wolf, M. 2006, Efficient and robust analysis of large phylogenetic datasets, *in* H.-H. Hsu, ed., 'Advanced Data Mining Technologies in Bioinformatics', Idea Group, Inc., Hershey, PA, USA, pp. 104–117.

Reaxys 2023, 'Reaxys', `https://www.reaxys.com/#/search/quick`. Accessed: 2023-2-2.

Reiter, T., Brooks, P. T., Irber, L., Joslin, S. E. K., Reid, C. M., Scott, C., Brown, C. T. and Pierce-Ward, N. T. 2021, 'Streamlining data-intensive biology with workflow systems', *Gigascience* **10**(1).
URL: *http://dx.doi.org/10.1093/gigascience/giaa140*

Remesh, A. 2012, 'Toxicities of anticancer drugs and its management', *Int. J. Basic Clin. Pharmacol.* **1**(1), 2.

Ronis, M. J. J., Pedersen, K. B. and Watt, J. 2018, 'Adverse effects of nutraceuticals and dietary supplements', *Annu. Rev. Pharmacol. Toxicol.* **58**, 583–601.

Russell, P. H., Johnson, R. L., Ananthan, S., Harnke, B. and Carlson, N. E. 2018, 'A large-scale analysis of bioinformatics code on GitHub', *PLoS One* **13**(10), e0205898.

Saitou, N. and Nei, M. 1987, 'The neighbor-joining method: a new method for reconstructing phylogenetic trees.', *Mol Biol Evol* **4**(4), 406–425.

Sarfraz, I., Asif, M. and Campbell, J. D. 2021, 'ExperimentSubset: An R package to manage subsets of bioconductor experiment objects', *Bioinformatics* .
URL: *http://dx.doi.org/10.1093/bioinformatics/btab179*

Schultz, J., Maisel, S., Gerlach, D., Müller, T. and Wolf, M. 2005, 'A common core of secondary structure of the internal transcribed spacer 2 (ITS2) throughout the Eukaryota.', *RNA* **11**(4), 361–364.
URL: *http://dx.doi.org/10.1261/rna.7204505*

Schultz, J., Müller, T., Achtziger, M., Seibel, P. N., Dandekar, T. and Wolf, M. 2006, 'The internal transcribed spacer 2 database–a web server for (not only) low level phylogenetic analyses.', *Nucleic Acids Res* **34**(Web Server issue), W704–W707.
URL: *http://dx.doi.org/10.1093/nar/gkl129*

Schöniger, M. and von Haeseler, A. 1994, 'A stochastic model for the evolution of autocorrelated DNA sequences.', *Mol Phylogenet Evol* **3**(3), 240–247.
URL: *http://dx.doi.org/10.1006/mpev.1994.1026*

Scott Long, J. 2009, 'The workflow of data analysis using stata', `https://www.stata-press.com/books/preview/wdaus-preview.pdf`. Accessed: 2023-1-28.

Seibel, P. N., Müller, T., Dandekar, T., Schultz, J. and Wolf, M. 2006, '4SALE–a tool

for synchronous RNA sequence and secondary structure alignment and editing.', *BMC Bioinformatics* **7**, 498.
URL: *http://dx.doi.org/10.1186/1471-2105-7-498*

Selig, C., Wolf, M., Müller, T., Dandekar, T. and Schultz, J. 2008, 'The ITS2 Database II: homology modelling RNA structure for molecular systematics.', *Nucleic Acids Res* **36**(Database issue), D377–D380.
URL: *http://dx.doi.org/10.1093/nar/gkm827*

Seo, M., Shin, H. K., Myung, Y., Hwang, S. and No, K. T. 2020, 'Development of natural compound molecular fingerprint (NC-MFP) with the dictionary of natural products (DNP) for natural product-based drug development', *J. Cheminform.* **12**(1), 6.

Shaikh, F., Tai, H. K., Desai, N. and Siu, S. W. I. 2021, 'LigTMap: ligand and structure-based target identification and activity prediction for small molecular compounds', *J. Cheminform.* **13**(1), 44.

Sharif, S., Liu, R., Orr, A. A., Khavrutskii, D., Jo, S., Lier, B., Croitoru, A., Burke, C., Frank, A., Weiner, J., McClean, N., Romany, A., Zhao, M., Serizawa, T., Deacon, J., Jones, I., Zhan, S., Kumar, A., Woster, M., Pinette-Dorin, R., Chow, E. Y., Schulhoff, S. and MacKerell, Jr, A. D. 2022, 'Global-Chem: A chemical knowledge graph of common small molecules and their IUPAC/SMILES/SMARTS for selection of compounds relevant to diverse chemical communities'.

Sharma, V., Goswami, R. and Madan, A. K. 1997, 'Eccentric connectivity index: A novel highly discriminating topological descriptor for {Structure-Property} and {Structure-Activity} studies', *J. Chem. Inf. Comput. Sci.* **37**(2), 273–282.

Shen, B. 2015, 'A new golden age of natural products drug discovery', *Cell* **163**(6), 1297–1300.

Shin, S., Son, D., Kim, M., Lee, S., Roh, K.-B., Ryu, D., Lee, J., Jung, E. and Park, D. 2015, 'Ameliorating effect of akebia quinata fruit extracts on skin aging induced by advanced glycation end products', *Nutrients* **7**(11), 9337–9352.

Siebert, S. and Backofen, R. 2005, 'MARNA: multiple alignment and consensus structure prediction of RNAs based on sequence structure comparisons.', *Bioinformatics* **21**(16), 3352–3359.
URL: *http://dx.doi.org/10.1093/bioinformatics/bti550*

Sliwoski, G., Kothiwale, S., Meiler, J. and Lowe, Jr, E. W. 2014, 'Computational methods in drug discovery', *Pharmacol. Rev.* **66**(1), 334–395.

Smith, A. D., Lui, T. W. H. and Tillier, E. R. M. 2004, 'Empirical models for substitution in ribosomal RNA.', *Mol Biol Evol* **21**(3), 419–427.
URL: *http://dx.doi.org/10.1093/molbev/msh029*

Sneader, W. 1997, 'Drug prototypes and their exploitation', *Eur. J. Med. Chem.* **1**(32), 91.

Sorokina, M. and Steinbeck, C. 2020, 'Review on natural products databases: where to find data in 2020', *J. Cheminform.* **12**(1), 20.

Stec, D. E. and Hinds, Jr, T. D. 2020, 'Natural product heme oxygenase inducers as treatment for nonalcoholic fatty liver disease', *Int. J. Mol. Sci.* **21**(24).

Stoudt, S., Vásquez, V. N. and Martinez, C. C. 2021, 'Principles for data analysis workflows', *PLoS Comput. Biol.* **17**(3), e1008770.

Sun, J., Wei, Q., Zhou, Y., Wang, J., Liu, Q. and Xu, H. 2017, 'A systematic analysis of FDA-approved anticancer drugs', *BMC Syst. Biol.* **11**(Suppl 5), 87.

Tayyem, R. F., Heath, D. D., Al-Delaimy, W. K. and Rock, C. L. 2006, 'Curcumin content of turmeric and curry powders', *Nutr. Cancer* **55**(2), 126–131.

The Metabolomics Innovation Centre n.d., 'FooDB'.

The World Bank 2022, 'GDP growth 2016-2021'.

Tidwell, J. 2010, *Designing Interfaces: Patterns for Effective Interaction Design*, "O'Reilly Media, Inc.".

Trolltech 2008, 'http://trolltech.com/products/qt/'.
URL: *http://trolltech.com/products/qt/*

USDA n.d., 'Dietary guidance', `https://www.nal.usda.gov/human-nutrition-and-food-safety/dietary-guidance`. Accessed: 2023-1-18.

van Leeuwen, J. E., Ba-Alawi, W., Branchard, E., Cruickshank, J., Schormann, W., Longo, J., Silvester, J., Gross, P. L., Andrews, D. W., Cescon, D. W., Haibe-Kains, B., Penn, L. Z. and Gendoo, D. M. A. 2022, 'Computational pharmacogenomic screen identifies drugs that potentiate the anti-breast cancer activity of statins', *Nat. Commun.* **13**(1), 6323.

van Santen, J. A., Poynton, E. F., Iskakova, D., McMann, E., Alsup, T. A., Clark, T. N., Fergusson, C. H., Fewer, D. P., Hughes, A. H., McCadden, C. A., Parra, J., Soldatou, S., Rudolf, J. D., Janssen, E. M.-L., Duncan, K. R. and Linington, R. G. 2022, 'The natural products atlas 2.0: a database of microbially-derived natural products', *Nucleic Acids Res.* **50**(D1), D1317–D1323.

Vanormelingen, P., Hegewald, E., Braband, A., Kitschke, M., Friedl, T., Sabbe, K. and Vyverman, W. 2007, 'The systematics of a small spineless Desmodesmus species, D-costato-granulatus (Sphaeropleales, Chlorophyceae), based on ITS2 rDNA sequence analyses and cell wall morphology', *Journal of Phyclology* **43**(2), 378–396.

Vatansever, S., Schlessinger, A., Wacker, D., Kaniskan, H. Ü., Jin, J., Zhou, M.-M. and

Zhang, B. 2021, 'Artificial intelligence and machine learning-aided drug discovery in central nervous system diseases: State-of-the-arts and future directions', *Med. Res. Rev.* **41**(3), 1427–1473.

Voss, K., der Auwera, G. A. V. and Gentry, J. 2017, 'Full-stack genomics pipelining with gatk4 + wdl + cromwell', *F1000Research* **6**.

Wang, Q., Hu, Y., Jiang, L., Guo, B. and Luo, Y. 2022, 'Molecular basis of longevity sustaining characteristics of chinese medicine herbs', *Pharmacological Research - Modern Chinese Medicine* **2**, 100037.

Weymouth-Wilson, A. C. 1997, 'The role of carbohydrates in biologically active natural products', *Nat. Prod. Rep.* **14**(2), 99–110.

Wheeler, D. L., Chappey, C., Lash, A. E., Leipe, D. D., Madden, T. L., Schuler, G. D., Tatusova, T. A. and Rapp, B. A. 2000, 'Database resources of the National Center for Biotechnology Information.', *Nucleic Acids Res* **28**(1), 10–14.

Wickham, H. 2016, 'ggplot2: Elegant graphics for data analysis'.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K. and Yutani, H. 2019, 'Welcome to the tidyverse'.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., Gonzalez-Beltran, A., Gray, A. J. G., Groth, P., Goble, C., Grethe, J. S., Heringa, J., 't Hoen, P. A. C., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S. J., Martone, M. E., Mons, A., Packer, A. L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S.-A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M. A., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J. and Mons, B. 2016, 'The FAIR guiding principles for scientific data management and stewardship', *Sci Data* **3**, 160018.
URL: *http://dx.doi.org/10.1038/sdata.2016.18*

Willighagen, E. L., Mayfield, J. W., Alvarsson, J., Berg, A., Carlsson, L., Jeliazkova, N., Kuhn, S., Pluskal, T., Rojas-Chertó, M., Spjuth, O., Torrance, G., Evelo, C. T., Guha, R. and Steinbeck, C. 2017, 'The chemistry development kit (CDK) v2.0: atom typing, depiction, molecular formulas, and substructure searching', *J. Cheminform.* **9**(1), 33.

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., Assempour, N., Iynkkaran, I., Liu, Y., Maciejewski, A., Gale, N., Wilson, A., Chin, L., Cummings, R., Le, D., Pon, A., Knox, C. and Wilson, M. 2018, 'DrugBank 5.0: a major update to the DrugBank database for 2018', *Nucleic Acids Res.* **46**(D1), D1074–D1082.

Wiswesser, W. J. 1985, 'Historic development of chemical notations', *J. Chem. Inf. Comput. Sci.* **25**(3), 258–263.

Wiszniak, S. and Schwarz, Q. 2021, 'Exploring the intracrine functions of VEGF-A', *Biomolecules* **11**(1).

Wohlgemuth, G., Haldiya, P. K., Willighagen, E., Kind, T. and Fiehn, O. 2010, 'The chemical translation service–a web-based tool to improve standardization of metabolomic reports', *Bioinformatics* **26**(20), 2647–2648.

Wolf, M., Achtziger, M., Schultz, J., Dandekar, T. and Müller, T. 2005, 'Homology modeling revealed more than 20,000 rRNA internal transcribed spacer 2 (ITS2) secondary structures.', *RNA* **11**(11), 1616–1623.
URL: *http://dx.doi.org/10.1261/rna.2144205*

Wolstencroft, K., Haines, R., Fellows, D., Williams, A., Withers, D., Owen, S., Soiland-Reyes, S., Dunlop, I., Nenadic, A., Fisher, P., Bhagat, J., Belhajjame, K., Bacall, F., Hardisty, A., Nieva de la Hidalga, A., Balcazar Vargas, M. P., Sufi, S. and Goble, C. 2013, 'The taverna workflow suite: designing and executing workflows of web services on the desktop, web or in the cloud', *Nucleic Acids Res.* **41**(Web Server issue), W557–61.
URL: *http://dx.doi.org/10.1093/nar/gkt328*

Wratten, L., Wilm, A. and Göke, J. 2021, 'Reproducible, scalable, and shareable analysis pipelines with bioinformatics workflow managers', *Nat. Methods* **18**(10), 1161–1168.

Xu, S., Chen, M., Feng, T., Zhan, L., Zhou, L. and Yu, G. 2021, 'Use ggbreak to effectively utilize plotting space to deal with large datasets and outliers', *Front. Genet.* **12**, 774846.

Yan, Y., Liu, Q., Jacobsen, S. E. and Tang, Y. 2018, 'The impact and prospect of natural product discovery in agriculture: New technologies to explore the diversity of secondary metabolites in plants and microorganisms for applications in agriculture', *EMBO Rep.* **19**(11).

Yu, G. and He, Q.-Y. 2016, 'ReactomePA: an R/Bioconductor package for reactome pathway analysis and visualization', *Mol. Biosyst.* **12**(2), 477–479.

Yu, G. P., Hsieh, C. C., Wang, L. Y., Yu, S. Z., Li, X. L. and Jin, T. H. 1995, 'Green-tea consumption and risk of stomach cancer: a population-based case-control study in shanghai, china', *Cancer Causes Control* **6**(6), 532–538.

Yukselen, O., Turkyilmaz, O., Ozturk, A. R., Garber, M. and Kucukural, A. 2019, DolphinNext: A graphical user interface for creating nextflow pipelines.

Zeng, X., Zhang, P., Wang, Y., Qin, C., Chen, S., He, W., Tao, L., Tan, Y., Gao, D., Wang, B., Chen, Z., Chen, W., Jiang, Y. Y. and Chen, Y. Z. 2019, 'CMAUP: a database of collective molecular activities of useful plants', *Nucleic Acids Res.* **47**(D1), D1118–D1127.

Zhang, L., Tan, J., Han, D. and Zhu, H. 2017, 'From machine learning to deep learning: progress in machine intelligence for rational drug discovery', *Drug Discov. Today* **22**(11), 1680–1685.

Zhang, Z., Bajic, V. B., Yu, J., Cheung, K.-H. and Townsend, J. P. 2011, 'Data integration in bioinformatics: current efforts and challenges', *Bioinformatics-Trends and Methodologies* pp. 41–56.

Zhong, L., Goldberg, M. S., Gao, Y. T., Hanley, J. A., Parent, M. E. and Jin, F. 2001, 'A population-based case-control study of lung cancer and green tea consumption among women living in shanghai, china', *Epidemiology* **12**(6), 695–700.

Zhou, D. H. 1982, 'Preventive geriatrics: an overview from traditional chinese medicine', *Am. J. Chin. Med.* **10**(1-4), 32–39.