Computational Toxinology

Joseph D. Romano

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
under the Executive Committee
of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2019

ABSTRACT

Computational Toxinology

Joseph D. Romano

Venoms are complex mixtures of biological macromolecules and other compounds that are used for predatory and defensive purposes by hundreds of thousands of known species worldwide. Throughout human history, venoms and venom components have been used to treat a vast array of illnesses, causing them to be of great clinical, economic, and academic interest to the drug discovery and toxinology communities. In spite of major computational advances that facilitate data-driven drug discovery, most therapeutic venom effects are still discovered via tedious trial-and-error, or simply by accident. In this dissertation, I describe a body of work that aims to establish a new subdiscipline of translational bioinformatics, which I name "computational toxinology".

To accomplish this goal, I present three integrated components that span a wide range of informatics techniques: (1) VenomKB, (2) `VenomSeq`, and (3) VenomKB's Semantic API. To provide a platform for structuring, representing, retrieving, and integrating venom data relevant to drug discovery, VenomKB provides a database-backed web application and knowledge base for computational toxinology. VenomKB is structured according to a fully-featured ontology of venoms, and provides data aggregated from many popular web resources. `VenomSeq` is a biotechnology workflow that is designed to generate new high-throughput sequencing data for incorporation into VenomKB. Specifically, we expose human cells to controlled doses of crude venoms, conduct

RNA-Sequencing, and build profiles of differential gene expression, which we then compare to publicly-available differential expression data for known diseases and drugs with known effects, and use those comparisons to hypothesize ways that the venoms could act in a therapeutic manner, as well. These data are then integrated into VenomKB, where they can be effectively retrieved and evaluated using existing data and known therapeutic associations. VenomKB's Semantic API further develops this functionality by providing an intelligent, powerful, and user-friendly interface for querying the complex underlying data in VenomKB in a way that reflects the intuitive, human-understandable meaning of those data. The Semantic API is designed to cater to the needs of advanced users as well as laypersons and bench scientists without previous expertise in computational biology and semantic data analysis.

In each chapter of the dissertation, I describe how we evaluated these 3 components through various approaches. We demonstrate the utility of VenomKB and the Semantic API by testing a number of practical use-cases for each, designed to highlight their ability to rediscover existing knowledge as well as suggesting potential areas for future exploration. We use statistics and data science techniques to evaluate `VenomSeq` on 25 diverse species of venomous animals, and propose biologically feasible explanations for significant findings. In evaluating the Semantic API, I show how observations on `VenomSeq` data can be interpreted and placed into the context of past research by members of the larger toxinology community.

Computational toxinology is a toolbox designed to be used by multiple stakeholders (toxinologists, computational biologists, and systems pharmacologists, among others) to improve the return rate of clinically-significant findings from manual experimentation. It aims to achieve this goal by enabling access to data, providing means for easy validation of results, and suggesting specific hypotheses that are preliminarily supported by rigorous inferential statistics. All components of the research I describe are open-access and publicly available, to improve reproducibility and encourage widespread adoption.

Contents

List of Figures

List of Tables

Acknowledgements

The road to a PhD is a long one, and is filled with countless examples of colleagues, family, and friends providing all manners of support. While it's not possible to acknowledge each and every one of them, I'd like to take the opportunity to thank some of the folks who have been there to help along the way.

First, the Tatonetti Lab is like a second family, always ready to share in the ups and the downs of graduate student life, to attend conferences with, and to practice lectures and talks with. To Rami Vanguri, Alexandre Yahi, Anna Basile, Phyllis Thangaraj, Theresa Koleck, Nick Giangreco, and Jenna Kefeli, it would be impossible to make it this far without the support of colleagues like you. And to the departed members of the lab, especially Tal Lorberbaum, Yun Hao, Fernanda Polubriaginof, Mary Regina Boland, Ola Jacunski, Robert Moskovitch, and Jeremy Chang, each of you have left large shoes to fill and have set the bar very high for my contemporaries and myself. And, of course, to Dr. Tatonetti, who has graciously supported my research and crazy ideas since joining the lab in 2014.

To my incredible dissertation committee, thank you for devoting so much of your time and energy to this process, and for enduring my frantic—and usually late-night—emails to schedule meetings across multiple time zones. I'm thrilled to have been able to assemble such a diverse panel of my most respected academic mentors—faculty members whom I know will challenge my work in the best ways possible.

No major body of research can be accomplished individually, and many collaborators have contributed to the venom studies that are described in this dissertation. For VenomKB, Victor Nwankwo provided a great deal of help constructing the web infrastructure. For

VenomSeq, the ‘wet-lab’ skills of Ron Realubit, Hai Li, and Chuck Karan saved many aliquots of cells and samples of venoms from the ignorance of a computational biologist (me), and their hard work resulted in some truly phenomenal data for my analyses. Marine Saint-Mézard provided crucial assistance in assembling and testing both the inner components and the graphical user interface for the Semantic API. Andrea Califano—one of the original creators of the PLATE-Seq technology used in VenomSeq—graciously provided the dataset of IMR-32 cells perturbed with 37 known drugs, which we used to evaluate whether VenomSeq data could be used reliably with the Connectivity Map reference dataset to discover putative therapeutic associations.

And finally, my deepest gratitude to the support given by my family and friends beyond the boundaries of Columbia and my academic circle. My parents, John and Kathy, and my sister, Tess, you bore with me for 18 years at home plus 10 years (and counting) beyond that, but more importantly you taught me that I really can attain my hopes and dreams. Sarah, the love of my life, you have been a constant source of stability and comfort in the best and the worst of times. I wouldn’t trade our time in NYC finishing up our studies together for the world.

To my mother, who has taught me the true meaning of strength in the face of adversity.

Preface

It is a common convention for dissertations to be written in the first-person, but given the amount of work performed by my various coauthors, I prefer to use the pronoun “we” rather than “I” when describing the research herein (unless referring to the dissertation itself). For ease-of-use, all cross-references (figure and table numbers, chapter/section references, and citation labels) and URLs are hyperlinks in the PDF version of this document.

Matrices are represented by upper-case bold letters (e.g., $\mathbf{X}$), while vectors are represented by lower-case bold letters (e.g., $\mathbf{x}$). Unless otherwise specified, matrix indices are denoted using “C-style” notation (row-major ordering). For example, both $\mathbf{R}_{ij}$ and $\texttt{R}[i,j]$ refer to the cell in the ith row and jth column of matrix $\mathbf{R}$. Occasionally I use “typewriter text” (monospace) fonts for specific variable names and other stylized proper nouns, such as `VenomSeq`. I use capitalized typewriter text to refer to computational subroutines, such as $\texttt{RANK}(\mathbf{V}_{ij})$. All figures and images are reproduced with permission from the copyright holders. Any figures or photographs borrowed from external authors are cited appropriately, unless they are in the public domain. All other conventions follow the guidelines set forth by the Graduate School of Arts and Sciences of Columbia University.

To readers looking for an abridged (i.e., “tl;dr”) version of this dissertation, I recommend focusing on §1.1 and Chapter 5.

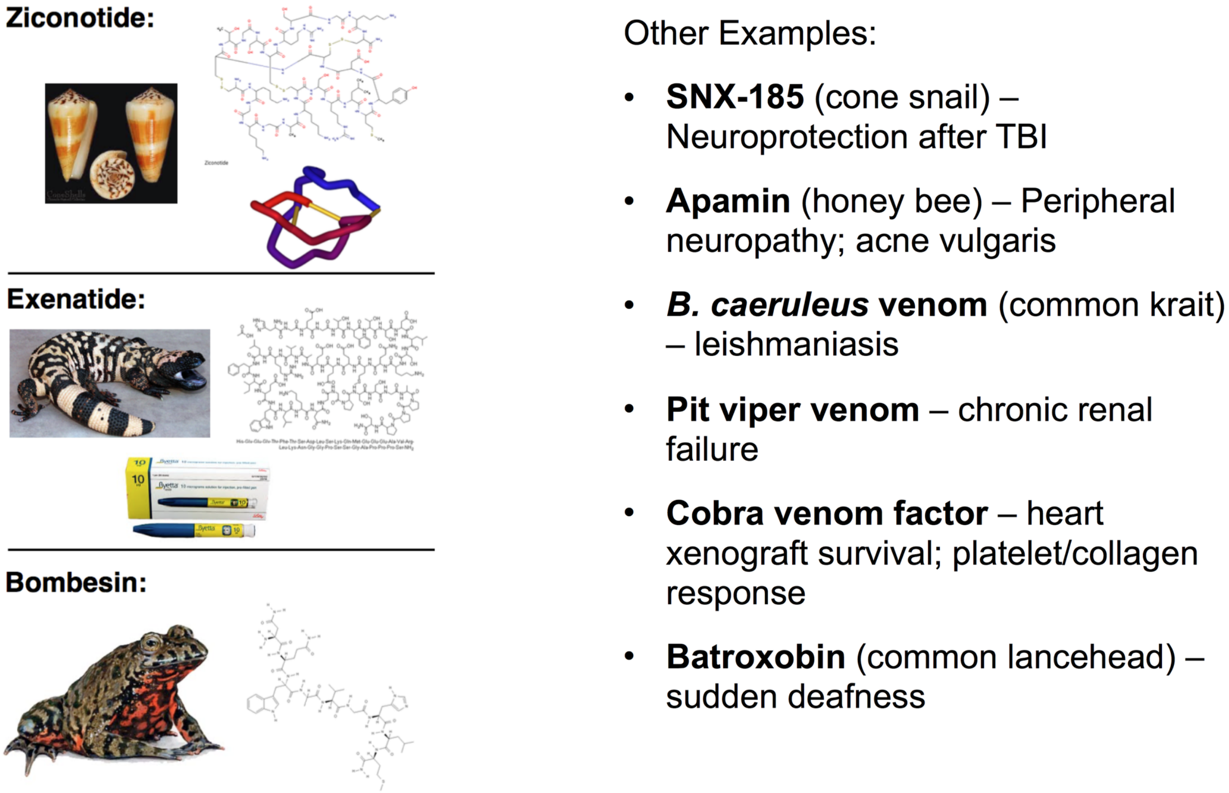
Chapter 1.

Background and Introduction

Venoms are complex mixtures of organic molecules and inorganic cofactors that animals use for defensive or offensive purposes (or, in some cases, both). The number of extant venomous species is vast, with current counts of known species exceeding 46,000 venomous spiders [195], 2,000 venomous fish [229], 600 venomous snakes [214], among many others. Current estimates suggest that these only comprise a small fraction of the actual number of venomous species in nature.

Since at least the dawn of recorded history, humans have used animal venoms and other natural toxins for therapeutic purposes, including the treatment of infectious diseases, chronic conditions, and trauma [141]. Therapeutic applications of venoms have continued to be used in modern medicine to great scientific and commercial success, with molecular characterization of venom effects on the human body being one of the most popular and well-funded areas of toxinology. Their highly targeted nature and bioavailability both suggest that venom-derived compounds are ideal for drug discovery.

In spite of these facts, the venom-based drug discovery industry faces many scientific obstacles compared to other classes of drug candidates. Venoms present unique challenges to researchers, including difficulty of compound isolation/characterization/synthesis, powerful toxic effects, and difficulties in working with the large macromolecules that comprise most active venom components. Furthermore, it is often challenging to work with venomous species, due both to risks in handling as well as complex issues related to animal conservation and trade laws. Although the study of venoms has made extensive use of emerging next-generation sequencing technologies, proteomics, and related computational approaches to studying venom composition and molecular function [35], other informatics and data science methods have been substantially underutilized.



Other Examples:

- **SNX-185** (cone snail) – Neuroprotection after TBI
- **Apamin** (honey bee) – Peripheral neuropathy; acne vulgaris
- ***B. caeruleus* venom** (common krait) – leishmaniasis
- **Pit viper venom** – chronic renal failure
- **Cobra venom factor** – heart xenograft survival; platelet/collagen response
- **Batroxobin** (common lancehead) – sudden deafness

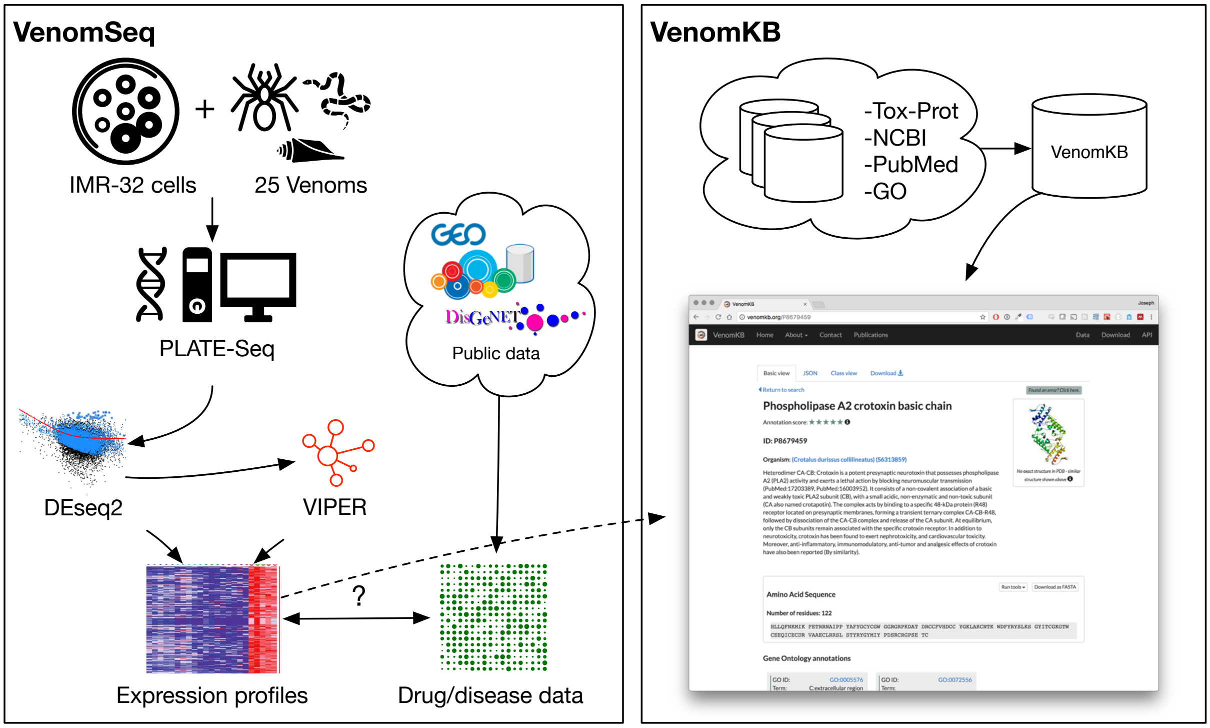
Figure 1.1.: Some examples of venoms and venom components with known therapeutic effects. Ziconotide (marketed as Prialt) is one of the most potent treatments for chronic pain. Exenatide (marketed as Byetta) is a successful drug for treating type-2 diabetes. Bombesin has demonstrated the ability to treat various gastrointestinal illnesses.

In this dissertation, I describe the development, integration, and application of two related resources that use cutting-edge translational bioinformatics to identify drug leads from venoms. VenomKB is an open-source knowledge repository and standardized data representation framework for venoms, venom components, and their effects (both molecular and systemic) on the human body. It provides a number of advanced tools for programmatic interaction with large quantities of standardized venom data, alongside a responsive, modern web interface for graphical browsing of the knowledge base contents. `VenomSeq` is a next-generation sequencing workflow that characterizes the effects that venoms have on gene expression in human cells, and uses these data as signals for inferring potential therapeutic effects of those venoms. VenomKB and `VenomSeq`—which are tightly integrated—provide considerable advantages over existing venom data analysis tools, especially for drug discovery.

1.1. Dissertation Overview

1.1.1. Specific aims

The goals of this dissertation can be broadly generalized into 3 *specific aims*. **Aim 1** concerns the development of VenomKB, **Aim 2** focuses on `VenomSeq` and the process of generating new data using the `VenomSeq` approach, and **Aim 3** involves the integration of these two components to perform discovery and validation (including the conceptualization and implementation of VenomKB's Semantic API, which helps to realize this goal).



Phospholipase A2 crotoxin basic chain

Annotation score: ★★★★★

ID: P8679459

Organism: (Crotalus durissus collilineatus) (S6313859)

Heterodimer CA-CB: Crotoxin is a potent presynaptic neurotoxin that possesses phospholipase A2 (PLA2) activity and exerts a lethal action by blocking neuromuscular transmission (PubMed:17203389, PubMed:16003952). It consists of a non-covalent association of a basic and weakly toxic PLA2 subunit (CB), with a small acidic, non-enzymatic and non-toxic subunit (CA also named crotapotin). The complex acts by binding to a specific 48-kDa protein (R48) receptor located on presynaptic membranes, forming a transient ternary complex CA-CB-R48, followed by dissociation of the CA-CB complex and release of the CA subunit. At equilibrium, only the CB subunits remain associated with the specific crotoxin receptor. In addition to neurotoxicity, crotoxin has been found to exert nephrotoxicity, and cardiovascular toxicity. Moreover, anti-inflammatory, immunomodulatory, anti-tumor and analgesic effects of crotoxin have also been reported (By similarity).

Amino Acid Sequence

Number of residues: 122

```
HLLQFNKMIK FETRRNAIPP YAFYGCYCGW GGRGRPKDAT DRCCFVHDCC YGKLAKCNTK WDFYRYSLKS GYITCGKGTW
CEEQICECDR VAAECLRRSL STYRYGYMIY PDSRCRGPSE TC
```

Gene Ontology annotations

GO ID:	GO:0005576	GO ID:	GO:0072556
Term:	C:extracellular region	Term:	

Figure 1.2.: Schematic overview of VenomKB and VenomSeq.

AIM 1: Create a centralized knowledge representation and data repository for computational toxinology

Aim 1 essentially consists of three components:

1. VenomKB v1.0 [205]
2. Venom Ontology [206]
3. VenomKB v2.0 [204]

These three parts comprise a unified informatics framework hosted at `venomkb.org` that provides structure and a formal representation for both new and previously existing venom data, which are required to answer the major questions I pose in this dissertation.

In VenomKB v1.0, we used literature mining techniques to retrieve MEDLINE articles describing therapeutic uses of venoms, and extracted specific mentions and biomedical predications corresponding to those uses, building a public database comprised of those results. Since the semantic landscape of venoms is poorly characterized, this provides a starting point for the development of more rigorous representations of venoms and venom data. We then built an ontology of venoms using data from the ToxProt annotation project of UniProtKB/Swiss-Prot [113], and populated the ontology to allow for formal structuring of venom knowledge. In VenomKB v2.0 we used the Venom Ontology to restructure the data from v1.0, ToxProt, and other public resources pertaining to various aspects of venoms and therapeutic applications of those venoms, and present them in a modern interface along with several analysis tools and APIs that make the data more accessible, both for humans and for computer programs.

AIM 2: Design a biotechnology platform for high-throughput screening of therapeutic venom effects

To accomplish **Aim 2**, we developed the biotechnology aspect of this dissertation—namely, the data generation pipeline I have named `VenomSeq` (which I stylize in a separate font to indicate its ability to stand alone and be reused as a contiguous workflow). `VenomSeq` is a transcriptomic computational screening approach for drug discovery that consists of exposing human cells in culture to dilute concentrations of venoms. After exposure, we carried out the PLATE-Seq protocol to obtain per-sample counts for nearly 20,000 human genes, yielding gene expression profiles under venom perturbation for differential expression analysis. In order to assess the efficiency of `VenomSeq` we compared the protocol to traditional RNA-Seq, focusing particularly on cost, time expenses, and resolution of the final data.

To test `VenomSeq` in a practical setting, we obtained venom samples from 25 diverse venomous species and carried out the `VenomSeq` protocol as described. We then used differential expression analysis software to identify which genes are significantly up- and down-regulated upon exposure to specific venoms. By comparison to public databases of gene expression, we compared these expression signatures to those for existing drugs and known diseases using various similarity metrics that allowed us to pose new hypotheses about the potentially therapeutic effects these venoms exert on human cells.

AIM 3: Integrate VenomSeq and VenomKB using semantic data analysis to improve detection and validation of therapeutic venom effects

Aim 3 seeks to unify the previous two aims by means of semantic data integration and harmonization to enable capabilities that are otherwise unattainable. In other words, **Aim 3** is a case study in using classical informatics techniques to improve the application of emerging next-generation sequencing technologies in the domain of natural product drug discovery.

After analyzing the data generated by `VenomSeq`, we incorporated both the raw data and the analysis results into the database schema of VenomKB. Under this expanded structure, gene expression data are considered instances of the ontology class `MolecularEffect`, and the analysis of the expression profiles effectively annotates them to instances of the ontology class `SystemicEffect`. We demonstrate that ontological inference can be used to link venomous species directly to clinically meaningful `SystemicEffect`s. To demonstrate the usability of this logical model, I also describe browsing and retrieval functionalities built into the VenomKB web application.

Beyond these routine data integrations, I conceptualized and constructed a new service that interfaces with VenomKB, the Venom Ontology, and a server-side graph database to deliver responses to *semantic queries* submitted by users of VenomKB. This system—which I have named the Semantic API—is a novel application of bioontologies and software engineering to extract relevant and highly domain-specific knowledge from complex data schemas. A semantic query consists of an intuitive description of the type of data the user is trying to obtain, which the Semantic API interprets and executes on the graph database. The Semantic API is a tool for both novices and veterans of manipulating complex data, serving multiple roles, including standardization, reduction of errors, saving time, reducing the complexity of querying the database, and even performing novel discovery. We demonstrate exactly how the Semantic API is beneficial both by validating known therapeutic associations of venoms as well as by exploring the plausibility of novel associations discovered by use of `VenomSeq`. The Semantic API is a general tool for intelligent information retrieval and knowledge discovery that can be ported to other domains.

The overarching goal of **Aim 3** is to demonstrate true translational research in the context of drug discovery from venoms: VenomKB is the application of traditional informatics and knowledge engineering techniques to a new problem, while `VenomSeq` provides new capabilities for generating and analyzing molecular data. However, it is only by using both of them in conjunction that we can translate observations on molecular data to findings that can improve human health.

1.1.2. Knowledge gaps

This dissertation is motivated by an overall need to incorporate certain classes of modern informatics techniques into toxinology and venom-based drug discovery research. Toxinologists have enthusiastically embraced next-generation sequencing for understanding venom composition at the levels of genomes, transcriptomes, and proteomes, and there is no lack of new research in these areas. However, the number of venomous species is vast, and new solutions are needed to apply venomics efficiently and economically to a larger fraction of these species. The technical challenges presented by this goal feed into the second major knowledge gap facing computational toxinology: namely, the need for standardization and centralization of venom-related data. Venom data are largely fragmented across a multitude of general-purpose biomedical databases that cannot adequately leverage the special characteristics of venoms that make them particularly attractive for drug discovery. A limited number of specialized venom databases do exist [94,115,192], but their scopes are too narrow to effectively convey information and knowledge about *all* venoms[1]. Also, since these existing databases are manually assembled and annotated, the data they contain are generally of high quality, but the *amounts* of data they contain are limited by available manpower and other human factors.

This dissertation also tackles other knowledge gaps beyond those in toxinology. Although perturbational differential expression analysis has been widely adopted by biomedical data scientists over the past decade, almost all molecular perturbations are conducted using small molecule drugs or drug candidates. Many compounds—including natural toxins—perturb cells in highly complex ways, especially when those compounds contain numerous types of molecules (e.g., venom peptides) [84,122,202]. The patterns in gene expression that result from these perturbations are almost completely uncharacterized. With `VenomSeq`,

[1] It should be mentioned that they can be very well-suited for representing data related to constrained groups of venomous species. Two excellent examples are Conoserver and ArachnoServer (described later in the dissertation, in §2.3.2).

we provide a set of these kind of data for analysis (using 25 crude venoms applied to one human cell line), and begin to explore the behavior of these perturbations using existing algorithms and by comparison to publicly accessible databases of small-molecule and disease perturbation.

Bioontologies are resources that have been touted for their ability to structure and formally represent virtually all types of biomedical data [28]. However, in recent years, interest in ontology-enabled biomedical data science has arguably waned, with attention instead being driven towards probabilistic (i.e., "knowledge-free") methods [251], meaning that impactful studies unifying bioontology research with cutting-edge computational methods have either failed to come to fruition or have simply been eclipsed by the surge in fields like machine learning and deep learning.

1.1.3. Significance and contributions

Contributions to toxinology

VenomKB and `VenomSeq` are intended to be major resources for the larger toxinology community. VenomKB has already been discussed at major conferences on venoms and covered in the popular media (see, for example, [174]), and we have begun to work in collaboration with highly established venom researchers to use `VenomSeq` in new contexts. Perturbational differential expression data is not a new concept, but we are the first to generate, analyze, and disseminate these data using crude venoms as perturbagens. Furthermore, we use these data to pose several novel hypotheses for therapeutic uses of venoms, and then validate those using real biological knowledge that is already contained in VenomKB. We also hope that the Semantic API proves popular and effective for bench and field biologists who do not have the expertise to construct complex data queries and aggregations.

Contributions to informatics

Together, these studies comprise an example of 'full-stack informatics', in that they range from the fundamentals of information representation to next-generation sequencing and systems pharmacology. Many fields of biomedicine have utilized the tools provided by the informatics community to great success, but toxinology has currently only done so in a few specific areas (see §1.1.2). The set of tools and techniques we develop in these studies seek to change that, by structuring existing venom knowledge and providing a foundation for data generation and analysis in the future.

Contributions to computer science

The Semantic API, in spite of its use in a highly biological context, is at its core computer science and software engineering. By abstracting ontologies as directed graphs, we use common and fundamental algorithms—such as filtering, sorting, and shortest path finding—and combine them using the complex domain knowledge asserted in the Venom Ontology to result in an intelligent software tool for information retrieval and knowledge discovery. Most modern tools for similar tasks rely on expensive computation (such as training deep learning models on distributed computing platforms) [55], constantly have to grapple with the bias–variance tradeoff (mitigated through the use of appropriately built ontologies) [82], and struggle to cope with relatively narrow domains (e.g., venom components and their effects on the human body) [161]. Although the individual components of the semantic query algorithm are mechanistically simple and well-established, they still make use of cutting-edge paradigms in software engineering, including graph databases and asynchronous execution.

1.1.4. Limitations

Like any body of scientific research, there are limitations to the studies described in this dissertation.

Limitations in Aim 1

One of the frequent criticisms regarding resources like VenomKB is that defining new standards, databases, and knowledge representations rather than improving existing ones perpetuates the issue of having too many competing resources that are not intrinsically compatiblee. Unfortunately, there is no easy solution to this problem. General resources used to create VenomKB (such as ToxProt, MEDLINE/MeSH, and others) cannot adequately address the characteristics unique to venoms, and other more specific databases (such as Conoserver and ArachnoServer) are too specific to yield translational discoveries that apply to all venoms—VenomKB hopes to reconcile these issues, but it is certain to inherit them to some degree, as well.

Limitations in Aim 2

`VenomSeq` comes with its own set of limitations, as well. In order to make the study tractable, it was necessary to place technical limitations on its methods that limit the generalizability of the study's results. For example, in order to mitigate between-sample variability and to

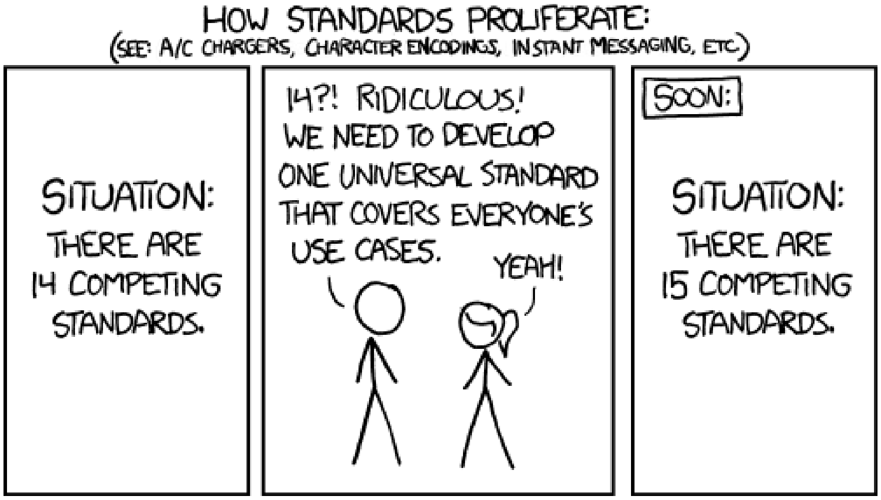

HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

Figure 1.3.: New standards (such as VenomKB, Venom Ontology, and `VenomSeq`) should be introduced cautiously. Used with permission from `https://xkcd.com/927/`.

capitalize on the availability of existing data, I made the decision to develop the protocol using the IMR-32 human neuroblastoma cell line [200]. Venoms and their components interact with different cell lines in different and unexpected ways, so the entire protocol needs to be eventually repeated for many human cell lines in order to obtain a more complete understanding of the effects of venoms on human gene transcription. A number of other issues are related to factors that are challenging (or even impossible, given present technical limitations) to control:

- The 25 venoms were selected *ad hoc* based on availability and cost, and therefore may inadequately represent certain groups of venomous taxa.
- It is particularly difficult to determine the effects of *individual components* of the 25 venoms, although we are working with collaborators to address this by applying `VenomSeq` to purified venom peptides in the near future.
- PLATE-Seq—although inexpensive and well-validated—still suffers from a lack of resolution compared to traditional RNA-seq; it is unclear precisely how this loss of reso-

lution impacts our findings.

- Determining an adequate dosage for venoms is challenging. To remain systematic, we opted to set dosages using observed GI_{20} concentrations to mitigate cytotoxic effects while ensuring the venom is still effecting the biochemistry of the cell.

Limitations in Aim 3

The Semantic API is arguably the least proven of the tools we have developed for computational toxinology. One of the major limiting factors in promoting the Semantic API as a highly effective and general purpose tool is that we have not had the ability to perform extensive user testing. In other words, although we designed the Semantic API to be friendly for computational novices as well as data science experts, we have not yet established whether each of these groups can effectively use it to solve real problems that they encounter in their research. Therefore, user testing will be essential moving forward. We also have yet to evaluate time complexity of the algorithms and how well they scale with increasing amounts of data. Since the Semantic API is really meant to be a general tool, we plan to continue to develop and refine it beyond the boundaries of this dissertation and its applications to computational toxinology.

1.2. Toxins, toxinology, and natural products

Toxinology—the study of venoms, poisons, and other naturally occurring toxins—is a scientific field that incorporates diverse aspects of venom research, including venom composition, the phylogeny of venoms and the species that produce them, the ecological functions of venom, and the interactions between humans and venoms (including the use of venoms to treat disease) [257]. Toxinology should not be confused with *toxicology*, which is the scientific study of adverse effects that chemicals (natural or otherwise) have on living organisms, especially humans. Toxinology and toxicology can, of course, overlap, since toxins by definition cause adverse effects. The different classes of toxins present in nature comprise subgroups of a larger class of chemical compounds known as *natural products* (NPs). Since other classes of NPs also form excellent chemical libraries for mining therapeutic effects, we will next explore the techniques available for drug discovery from NPs, which will help to provide perspective on methods available in searching for therapeutic effects of venoms in particular, as well as additional challenges and opportunities we face that have not yet been covered in this chapter.

1.3. Informatics methods in natural product drug discovery

Drug discovery is the process by which new pharmaceutical drugs are identified, and along with drug development (validating, testing, and marketing a new drug), it comprises one of the most substantial activities in pharmaceutical science. A 2018 analysis showed that roughly 20% of the US National Institutes of Health (NIH) budget for the years 2010–2016 funded the discovery and development of 210 new molecular entities [44]. Since the advent of modern medical science, most systematic drug discovery has focused on small molecule candidates—for example, over 86% of the drugs (both approved and experimental) in the DrugBank database are comprised of small molecules [261]. This is due to many reasons, including relative ease of synthesis, generally high chemical stability, and more straightforward characterization of reactivity [63]. The pervasiveness of small molecules in drug discovery is

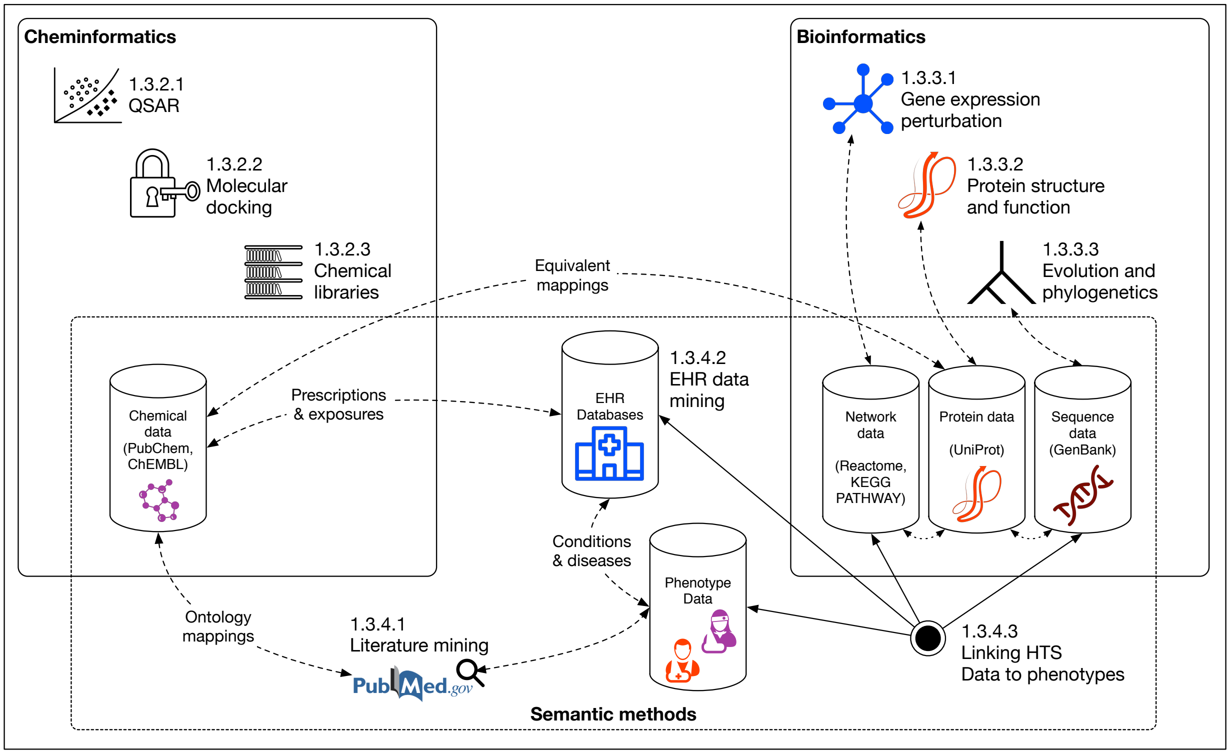

Cheminformatics
1.3.2.1 QSAR
1.3.2.2 Molecular docking
1.3.2.3 Chemical libraries
Bioinformatics
1.3.3.1 Gene expression perturbation
1.3.3.2 Protein structure and function
1.3.3.3 Evolution and phylogenetics
Equivalent mappings
Prescriptions & exposures
Chemical data (PubChem, ChEMBL)
EHR Databases
1.3.4.2 EHR data mining
Network data (Reactome, KEGG PATHWAY)
Protein data (UniProt)
Sequence data (GenBank)
Conditions & diseases
Phenotype Data
Ontology mappings
1.3.4.1 Literature mining
PubMed.gov
1.3.4.3 Linking HTS Data to phenotypes
Semantic methods

Figure 1.4.: Informatics methods for natural product drug discovery covered in this review. Numbers preceding methods correspond to section/subsection numbers in the manuscript describing the method. Dashed lines indicate inferred links between various data resources.

Table 1.1.: Summary of popular computational drug discovery methods described in this review and their applicability to NP drug discovery, stratified by the major branches of informatics discussed in this review.

Informatics branch	Method	Use with NPs
Cheminformatics	QSAR analysis (§1.3.2)	Multiple
	Molecular docking (§1.3.2)	Multiple
	Computational library design (§1.3.2)	Multiple
Bioinformatics	Gene expression perturbation (§1.3.3)	Little to none
	Protein modeling (§1.3.3)	Multiple
	Phylogenetic approaches (§1.3.3)	Multiple
Semantic methods	Literature mining (§1.3.4)	Limited
	EHR mining (§1.3.4)	None
	Linking HTS data to effects (§1.3.4)	Little to none

even reflected in Lipinski's "rule of five," which defines a set of common best-practice guidelines for filtering potential orally-active drug candidates: "Good" compounds should have a molecular mass less than 500, no more than 5 hydrogen bond donors, and no more than 10 hydrogen bond acceptors, among other principles [144]. In recent decades, the ubiquity of computers and computational methods in science has extended to drug discovery [227]. Cheminformatics, for example, is the application of computer science to understanding and characterizing molecular attributes and chemical behavior of specific compounds. These methods have generated massive libraries of small molecules to screen against specific therapeutic processes [27]. Once candidates are identified, other cheminformatics methods can be used to generate libraries of compounds structurally and chemically similar to the identified 'hits,' in order to optimize stability, toxicity, and kinetics. Complementarily, bioinformatics techniques can be used to discover how candidate drugs cause therapeutic activity within the human body, which can include predicting interactions between drugs and proteins, analysis of impact on biological pathways and functions, and elucidating genomic variants that can alter drug response [63].

Despite these technological advances in drug discovery, the approval of new therapeutic drugs has slowed considerably in recent years. For example, between 1996 and 2007, the number of new molecular entities approved by the US FDA has fallen from 53 to 17 per year—the same rate as over 50 years ago [73, 175]. This seems to be due to many factors, including the following:

1. The “lowest hanging fruits” in terms of small molecule drug candidates have been extensively investigated, and computational challenges hinder extension of traditional methods to more complex structures. Researchers refer to “rediscovering the sweet spot” in the discovery process [30], and have devoted a great amount of effort to producing new, targeted screening libraries that leverage anticipated characteristics of lead compounds [41, 256].
2. Many remaining diseases of top clinical priority have highly complex etiologies, and are accordingly difficult to associate with potential drug targets [199].
3. Model organisms may not provide adequate templates for testing treatments of more complex diseases, due to inter-species variations that are crucial to therapeutic action [68, 106].

A natural way to address the first two challenges is to focus on new classes of potential drugs outside of small molecules. *Natural products* (NPs) may serve this need by returning to the sources of therapeutic compounds that have treated illness for thousands of years [59]. Although rigorous pharmaceutical science is young in comparison to the historical use of NP drugs, many cutting-edge advances have emerged with the promise of ‘modernizing’ this field [90]. Along with a renewed interest for NP drugs within the biomedical research community, this has already resulted in substantial developments in the pharmaceutical industry—a comprehensive enumeration by Newman and Cragg shows that 41% (646/1562) of all new drug approvals between 1981 and 2014 are NPs or derived from NPs [180]. Several recent reviews provide excellent summaries of NP drugs and the broad spectrum of techniques that have been used both for their identification and characterization [119,203], particularly from the perspective of bench research techniques and state-of-the-art developments in biotechnology. Considering the aforementioned trends in new computational methods and advances in classical informatics for translational applications of these methods, these reviews can be complemented by a dedicated discussion restricted to *in silico* approaches for NP drug discovery.

Another trend in drug discovery enabled by informatics and computational methods is an increasing shift towards a *data driven* drug discovery [147, 240]. Traditionally, drug discovery has been performed as follows: basic scientists first find a target structure in the human body related to a disease or illness, followed by screening for "lead" compounds that show affinity for the target. Subsequently, the list of candidates is narrowed down (using some of the methods described in this review) to find the most promising leads, which then go through the development process to assess safety and efficacy in model organisms and, eventually, humans. A detailed description of these steps can be found in other reviews [105]. Failure at any stage in this workflow can—and usually does—necessitate starting over from the beginning, contributing to the estimated cost of 2.6 billion USD to bring a new drug to market [10]. Data-driven drug discovery turns ths process on its head, by using data mining on large data repositories of candidate compounds and disease knowledge to generate novel therapeutic hypotheses systematically rather than hoping for a single therapeutic hypothesis to deliver actionable results. Aside from avoiding systematic biases present in the hypothesis-driven model, this additionally helps to improve the return rate on subsequent manual experimentation and validation of lead compounds, ultimately lowering costs and increasing productivity [111]. Data-driven drug discovery leverages new data types that were previously inaccessible, and relies heavily upon computers and informatics techniques to produce increasingly accurate results [34].

In this review, we first discuss various major classes of natural products based both on source organism and their biological functions. In addition, we provide examples of specific members of those classes with demonstrated therapeutic potential. We then explore several major disciplines based upon informatics and computational methods—cheminformatics, bioinformatics, and semantic (or 'knowledge-based') informatics—and their associated methods that can be used specifically for NP drug discovery. These methods are summarized graphically in **Figure 1.4**. Finally, we conclude with a recap of the major gaps currently facing the field of computational NP drug discovery, and suggest actions for the future that could help to resolve these problems.

1.3.1. Classes of therapeutic natural products

There is no definitive consensus on what groups of substances comprise "natural products", with some authors restricting them to small molecule secondary metabolites [179], and others more broadly stating that an NP is any chemical substance produced by a living organism [178]. For the purpose of this review, we adopt the latter of these two definitions: that natural products include all classes of chemical substances that are produced or recruited by living organisms, and have the ability to be isolated and reused by humans. This definition includes an incredibly diverse range of compound types; therefore, it is crucial to understand the different subgroups of NPs, along with their characteristics. These classes of NPs frequently overlap and have vaguely defined boundaries, but they are nevertheless useful for understanding the methods that can be applied to them.

Phytochemicals

Phytochemicals—chemicals synthesized by plants—encompass an broad range of NPs, including members of many of the other classes described later in this section. Phytochemicals can be toxic, they can provide important dietary nutrients (such as amino acids, antioxidiants, and dietary fiber), or they can be inert in humans. For most research purposes, however, phytochemicals are limited to primary and secondary metabolites in plants, which can be generally divided into phenolic acids, stilbenes, and flavonoids (which, themselves, can be further subdivided into more specific subclasses), all of which are small molecules (rather than macromolecules, which tend to be prevalent in many of the other classes we discuss) [88]. These chemicals have been the source of many traditional and modern medicines, famous examples of which include the analgesic acetylsalicylic acid (aspirin), the heart medication digoxin, and the chemotherapy drug paclitaxel [173].

Fungal metabolites

Fungal metabolites serve a relatively similar role to plant metabolites, so much so that they share some of the same subclasses (perhaps most notably the flavonoid compounds). Like plant metabolites, fungal metabolites can treat a wide variety of diseases and conditions, but they are perhaps most famous as a source of many successful antibiotics. Other areas of successful application include antimalarials (antiamoebin), immunosuppressants (ciclosporin), statins (mevastatin, lovastatin), and more [244].

Toxins

Toxins are substances that can potentially harm or kill. They include poisons and venoms, and are (by definition) produced by living organisms. Poisons are toxins that cause harmful effects when swallowed, inhaled, or absorbed through the surface of the skin, while venoms are toxins that cause harm when actively injected via a sting or a bite.

Poisons are produced by members of many major clades of organisms, including plants, fungi, bacteria, and most groups of animals. Natural poisons are usually used for defensive purposes, although some species have adapted them for more complex roles [123]. They can include members of all classes of molecules, and although many tend to consist of relatively small molecular structures, macromolecules such as proteins, large cabohydrates, and lipids can be poisonous as well. NP poisons include many chemotherapy drugs, particularly when their toxic effects act more selectively on cancer cells than healthy cells. Some examples include paclitaxel (from *Taxus brevifolia*) and vinblastine (from *Catharanthus roseus*) [244].

Venoms are complex mixtures of chemicals produced by animals for either defensive or offensive purposes (or, sometimes, both in the same species). An individual species' venom can include hundreds of unique chemical compounds, many of which are proteins that act on specific molecular targets. Venoms are highly evolutionarily optimized to fit organisms' biological niches [52], but due to interspecies homology, the effects of individual venom components have led to numerous therapeutic applications, including FDA-approved treatments for hypertension, diabetes, neuropathic pain, and more [141]. Like poisons, venoms have also demonstrated potent anti-cancer effects, and their high target specificity has made them of particular interest for applications of precision medicine, particularly for rare or aggressive cancer types [206, 267].

Antibodies

Components of the immune system—particularly antibodies—have long been attractive for drug discovery and design. Their primary function is recognition and inactivation of pathogens, including bacteria and viruses, but biotechnologists have repurposed them for many ‘unintended’ uses, including the targeted treatment of various diseases. One approach, known as immunotherapy, involves the design and application of monoclonal antibodies that bind specifically to certain cells or proteins related to the disease of interest. Naturally, these are often autoimmune diseases, such as rheumatoid arthritis [222] and allergies [114], but they have also been applied to diverse diseases such as viral infections [140] and multiple sclerosis [100]. Recently, substantial attention has been given to immunotherapy treatments for cancer, exemplified by the 2018 Nobel Prize in Medicine being awarded for research in this area [108, 136, 209]. The second approach involves using antibodies as delivery agents for therapeutic compounds, which is also being explored extensively for cancer, due to its capacity to mitigate off-target effects [11]. Interestingly, this delivery method has attracted specific attention for the delivery of chemotherapeutics that are, themselves, NPs [157].

It should be noted that—in spite of the substantial accomplishments described above—antibodies have failed to deliver on several therapeutic applications that originally held promise, often for characteristics that are inherent to antibodies in general. One example involves the treatment of Alzheimer Disease (AD) using monoclonal antibodies. Antibody-based treatments for AD performed strongly in mouse models [13] and in early-phase clinical trials [99], but in phase-2 trials and beyond, they have failed to deliver [241]. Multiple theories have been posed, but the two leading hypotheses for failure have been that (1) antibodies are limited in their ability to cross the blood-brain barrier, and (2) certain degenerative diseases require early treatment for antibodies to be effective, far before patients begin to show symptoms [231]. Other failures in antibody therapy are related to the activity of antibodies themselves—drugs like theralizumab (designed to treat leukemia and rheumatoid arthritis) failed in human trials due to inciting a life-threatening 'cytokine storm' in all healthy volunteers [67]. Nonetheless, much research on new antibody therapies is being conducted to treat the same diseases associated with these early failures [223].

NPs with limited therapeutic potential

The classes of NPs described above cover substantial breadth. However, to provide a more complete image of drug discovery in terms of NPs, it is also important to consider classes with only limited—or at least presently unknown—therapeutic potential. For the purposes of this review, we focus on whether a compound is reactive enough in living systems to potentially perturb that system. If it is, then there exists an opportunity to exploit the perturbations for potentially therapeutic outcomes. The largest group of NPs that falls short in this regard is those with purely structural purposes, including materials like wood, biopolymers, and excretions like spider silk, which suggests that the drug discovery methods discussed in subsequent sections of this review are unlikely to generate many new lead compounds.

Nonetheless, biology is rife with exceptions to every rule, and even these groups of NPs have occasionally yielded compounds with therapeutic use. Wood creosote has been used for centuries as a treatment for diarrhea, and is currently marketed in Japan under the trade name Seirogan [98]. Biopolymers have not resulted in drugs themselves, but have been used many times to successfully deliver drugs within living systems [183]. Even spider silk has shown potential in drug delivery [232], and has been bioengineered to have antibiotic properties [91]. For this reason, we hesitate to say that any class of NPs has no therapeutic potential. In a practical sense, these observations are most useful in a cost-benefit analysis scenario, when it is necessary to balance research budget with scientific risk, highlighted by Dickson and Gagnon as one of the major factors influencing the total output of the pharmaceutical industry [60].

1.3.2. Cheminformatics methods

Cheminformatics methods can generally be classified according to the types of characteristics they exploit: either direct measures of chemical activity (e.g., chemical constants, reactive groups, or ADME measurements), or indirect measures (e.g., structural motifs, compound class membership, or other higher-order observations). These techniques can be further subdivided; for example, structural comparisons can be applied either before or after promising chemical activity is known (which we refer to here as *prospective* and *retrospective* structure mining, respectively). Prospective structure mining is conducted in a supervised manner, where known chemical activity of well-characterized compounds is compared to the structures of query compounds to predict the therapeutic potential of the queries. Retrospective structure mining, on the other hand, is more analogous to unsupervised learning techniques, where other screening techniques first identify a compound of interest (referred to as a “hit”), and then seek to expand the number of candidate compounds by searching for structures that are similar to the hit compound.

Many traditional cheminformatics methods are challenging to adapt to certain classes of NPs, particularly when the NPs consist of large chemical structures (like venoms, antibodies, or other protein-based NP drug candidates). For example, generating combinatorial libraries of large polypeptides is currently intractable, due to the massive search space. However, additional characteristics that are unique to these classes of NPs enable either simplifying assumptions to be made or the invention of entirely new approaches for predicting bioactivity [104]. Here, we divide cheminformatics into 3 major categories of methods that have been used to success with NPs, providing discussion of the caveats that must be considered for NPs in particular.

Natural product QSAR analysis

Quantitative Structure Activity Relationship (QSAR) analysis is a widely used—if often ambiguously defined—technique in cheminformatics for predicting a response variable given a set of structural, chemical, and or physical input variables (known as *molecular descriptors*). Generally, the goal is to learn a function of the form

$$\hat{y} = f(\mathbf{x}) + \epsilon$$

where $\mathbf{x} = (x_1, \ldots, x_N)$ is the vector of N input variables, $\hat{y}$ is the estimated response (continuous in the case of regression, and integer-valued in the case of classification), and ϵ is an error term. f can be any appropriate model; common choices include logistic regression, support vector machines, random forest, artificial neural networks, and others. Recently, deep learning has shown to be particularly effective for predicting a wide variety of responses, including solubility, probe-likeness, and others [126]. A number of free and commercial software implementations of QSAR are available for a variety of use cases [19, 246], and approaches for adapting generic statistical and machine learning models for QSAR are readily available [135].

QSAR has been applied fairly widely to different classes of NPs, where specific classes tend to dictate the chosen molecular descriptors. Typical choices for non-NP applications include symbolic (1- or 2-D) descriptors, 3-D spatial organization, higher-order (e.g., time-dependent or ligand-bound) conformational characteristics [196], experimental measurements (partition coefficient, polarizability, refractivity, etc), and many others. For a detailed review of these and similar descriptors, see [42]. Additional characteristics that can be used for small-molecule NPs include categorical (‘one-hot’) variables indicating class membership (e.g., alkaloid, terpenoid), species of origin (or more general taxonomic clades), and other biological features. Macromolecular NPs are substantially more restricted in terms of the types of descriptors that can be used effectively. Generally, 3-D conformational descriptors and binding data function best for these NPs, and yield good results [58, 171]. QSAR has performed adequately for predicting binding affinity of antibodies to proteins—Mandrika et al. describe a model consisting of 26 physicochemical descriptors (covering hydrophobicity, polarity, electronegativity, etc.) at each amino acid position in a library consisting of single chain monoclonal antibodies [156]. While this model has not yet been applied to NP drug discovery, it seems to be a feasible way forward.

Molecular docking and dynamics

QSAR is a useful statistical method for predicting potentially therapeutic interactions, but it is often desirable to directly model the chemical or physical interaction that is being investigated. *Molecular docking* is an approach that seeks to predict if and how two compounds (usually a *target* and a *ligand*) physically interact. This is usually performed in two steps:

(1) searching for potential conformational fits, and (2) scoring those fits. Molecular dynamics is a particular simulation technique that can be applied to docking, and is popular in drug development. From a high level, molecular dynamics performs a computational simulation of the atoms and molecules (often including solvents) present in a putative reaction, and allows the molecules to interact for a period of time. The technical details and algorithms for docking and dynamics are well summarized elsewhere [118, 189]—we will instead focus on broad caveats, issues, and innovations in applying these to NPs.

The class of NP compound tends to dictate the role (target vs. ligand) that the compound plays in docking simulations. Typically, small molecule NPs and relatively short polypeptides (e.g., peptide toxins and venom components) act as ligands, while larger proteins and protein complexes act as targets (although exceptions are common). This distinction is important, especially when the goal is screening many candidate compounds: usually, the target is held fixed, while the ligand can be drawn from libraries of many compounds. Therefore, it is computationally feasible to perform docking of many small molecule compounds when a specific molecular target is already known [121, 137, 151]. Conversely, if a macromolecular NP is suspected of interacting with endogenous small-molecule metabolites (e.g., in human cancer cells), docking simulations can be used to mine *which* metabolites could bind to the NP [194]. If both a target and a ligand are already predicted by other means (e.g., QSAR or other methods described in this review), docking is commonly used as a secondary validation method. In spite of their large molecular weight, antibodies are relatively easy to screen in large numbers via docking, due to their specific structural and binding constraints that can substantially reduce computational complexity of simulations [1, 252].

Molecular dynamics is an important technique for characterizing physical interactions of putative drugs with their targets, but due to computational challenges it cannot be used with current technologies in a data-driven manner to screen very large numbers of NPs against similarly large numbers of potential targets simultaneously [217]. However, it has proven incredibly valuable in uncovering specific therapeutic mechanisms of NPs (venom proteins in particular). An early and influential example of this came in 1995, when Albrand et al. combined molecular dynamics with NMR to explain how Toxin FS2 (from Black Mamba venom) blocks L-type calcium channels, causing potent cardiotoxic effects [3]. Additionally, there are noteworthy success stories that have emerged from screening relatively small NP databases against specific drug targets: The compound ellagic acid—which has shown both antiproliferative and antioxidant properties—was identified by Moro et al. by screening a proprietary database of 2,000 NPs against the oncoprotein casein kinase 2 [50]. Similarly, Fu et al. identified Jadomycin B—another molecule with anticancer effects—by screening 15,000 microbial small molecule metabolites against the oncoprotein Aurora-B kinase [78]. These examples illustrate the feasability of molecular dynamic studies for discovering new therapeutic NPs, and suggest that overcoming associated computational challenges will enable their widespread application in diverse and data-driven contexts.

Computational mutagenesis and library construction

One of the most common techniques for identifying drug candidates is to generate massive libraries of compounds that can be screened in parallel, with the understanding that only a very small fraction will result in 'hits' (potential therapeutic activity). There are many ways such libraries are generated, many of which fall under the umbrella term of *combinatorial chemistry* (i.e., enumerating chemical structures using combinatorics) [242]. NPs provide some advantages over traditional (non-NP) classes of candidate compounds, namely that such 'libraries' already exist in nature. General purpose online databases of chemical compounds (such as PubChem and ChEMBL) [81, 142] contain many NPs that are annotated by compound class, while other, more specific databases (such as ArachnoServer, VenomKB, and the Dictionary of Marine Natural Products) provide even more granular annotations for aggregating NP libraries with various characteristics of interest [192, 204].

Computational mutagenesis is a related class of techniques that has shown efficacy in certain classes of NPs. This method involves specifying a template (e.g., a certain antibody with putative therapeutic activity that requires optimization), and then sequentially mutating locations in the template's structure to generate a library of candidate compounds. These libraries can then be screened *in silico* (e.g., using molecular docking simulations as described in §1.3.2) to find structures that can be engineered in the lab. Antibodies, in particular, are particularly well-suited to computational mutagenesis, by modifying amino acids in binding regions [226, 262]. The feasability of mutagenesis techniques in the context of NP drug discovery was demonstrated by Chen et al., who generated a library of analogues of the 7-residue NP peptide HUN-7293 to optimize its inhibitory effects on cell-adhesion [40].

It should be noted that one of the advantages of working with NPs is the potential of avoiding library screening entirely, under the assumption that nature has optimized it for biological activity. This point is expanded on in §1.3.3.

1.3.3. Bioinformatics methods

Bioinformatics methods for drug discovery include anything related to the *biological* function of potentital drug candidates, including sequence-based characteristics, interactions with body structures (metabolites, proteins, cells, tissues, etc.), pathway perturbations, and toxicity, among others. Multi-omics and high-throughput sequencing are also major areas within bioinformatics. Most subdisciplines of bioinformatics can be applied in some way to the drug discovery process [244,261].

In the case of NPs, researchers are able to make use of an entire range of techniques related to the organisms that produce the compounds. In particular, phylogenetics and evolution provide many routes for various drug discovery activities. Closely related organisms often produce similar proteins and metabolites, so when one natural compound with promising activity has an unsuitable therapeutic index for human use, libraries of similar compounds can be easily constructed by searching in organisms within the same genus. However, these techniques must be applied with caution: members of some groups of natural compounds (such as venom proteins) are heavily optimized to fit a very particular biological niche, so even members of the same species may have entirely unique metabolic profiles with respect to compounds of interest. One prominent example of this was found in the rattlesnake species *Crotalus oreganus helleri*, where members of the species living on different sides of a mountain range produced entirely separate venom profiles [237].

Gene expression perturbation

The rise of multi-omics approaches to uncovering mechanisms of disease has led to multitudes of ways to assess the effect that putative drugs have on cells. In particular, gene expression perturbation—quantified using RNA-sequencing and transcriptomics—has led to a number of innovative breakthroughs in drug discovery for diseases associated with gene disregulation, including cancers and various other diseases with complex genetic etiologies [225,234]. Along with environmental exposures, structural abnormalities, and other influencing factors, these diseases often can be attributed in part to abnormalities in gene expression, including the systems-level effects of expression perturbation in the larger context of cell signaling and metabolic networks [47, 182]. More accurately, differential expression can be treated as a phenotypic signal that arises from underlying disease etiology. Accordingly, drugs and drug candidates that effectively *invert* such deleterious effects are potential therapies for these diseases.

This technique is particularly well-adapted for use in NP drug discovery, as vast numbers of compounds from all classes of NPs are specifically optimized to have roles in cell signaling or metabolic networks, and are already known to be relatively biologically stable [141]. Compounds used in Traditional Chinese Medicine (TCM) have been particularly well utilized in this area. In a 2014 study, researchers uncovered likely mechanisms by which the TCM compound berberine exhibits anti-cancer activity, using publicly-available expression data for berberine-perturbed human cells taken from the Connectivity Map (CMap) project [138]. Another important recent example by Lv et al. provides differential gene expression profiles in response to 102 different TCM compounds, presented as a framework from which to base future systematic research on the activities of TCMs [149].

A separate but related approach involves analysis of differential expression in the organisms *producing* the NPs (rather than the organisms that NPs act upon). An investigation by Amos et al. discovered previously unknown NPs—as well as putative mechanisms describing their functionality—by comparing transcriptome profiles of different bacterial species in the genus *Salinispora* [5], underscoring the diversity of emerging multi-omics techniques that can be employed within NP drug discovery.

Modeling protein structure and function

Although the size and complexity of proteins is often prohibitive to structure-based analyses designed for small molecules, other drug discovery approaches leverage the unique characteristics of proteins and other macromolecules to perform discovery in ways that are otherwise impossible. Since many classes of NPs are comprised of proteins, these techniques can often be adapted to NP drug discovery with relative ease.

Some methods use supervised machine learning algorithms trained on protein structures (and motifs) with known activity to predict activity in previously uncharacterized proteins; this is essentially traditional QSAR designed to work on proteins. The FEATURE framework [85] does this using 3-dimensional spatial orientation of atoms to predict activity at numerous "microenvironments" within a larger macromoleucle, and is therefore generalizable to diverse proteins with conserved functional activity. Other research teams have designed similar frameworks based on other machine learning models, including deep learning models like convolutional neural networks [243, 245]. For further details on learning protein function from structure, we refer the reader to [190].

Still other protein functional modeling approaches rely on input variables that behave like "abstractions" of raw molecular characteristics, including amino acid or DNA structure (along with sequence alignment algorithms) [250], ontology annotations (see §1.3.4 for more details) [177], and biomarker response [75].

Using evolution to discover drug candidates

The fact that NPs are derived from living organisms implies that they either serve a specific purpose in the context of that organism, or they are a byproduct of an important process [233]. Therefore, we can use evolution and taxonomy as tools for both discovering new compounds and their effects, as well as for generating libraries of similar natural products [160].

The simplest—and most common—use of phylogenetics in natural product drug discovery revolves around the axiom that *closely related species produce similar NPs.* This can be used to predict the structures of NPs, given structures for similar NPs in related species are already known [273]. Following a pattern akin to QSAR modeling (described in §1.3.2), phylogenetics can also be repurposed to predict other characteristics of closely related NPs, including molecule classes, toxicity, stability, and others, where instead of using molecular descriptors as observed features of the NP, you instead use evolutionary characteristics to build a predictive model. A noteworthy example is given by Malhotra et al., who used discriminant function analysis (DFA) to classify and predict functions of over 250 phospholipase A_2 proteins from viperid snakes, where aligned amino acid sequences alone were used to construct the input features for the DFA model [154].

Other uses of evolution in drug discovery employ phylogenomics to discover associations across more distantly related species (e.g., between humans and microbes). This includes efforts to catalog the entire breadth of various classes of natural products to create comprehensive NP class libraries (see §1.3.2 for more details) [208]. In 2016, Rudolf et al. showed that comparative genomics in diverse microbial species could identify 87 distinct gene clusters across 78 bacterial species corresponding to a class of putative NP anticancer drugs known as enediynes [213]. By finding instances of NP coevolution in distantly related species, studies have uncovered compounds that play keystone roles in metabolic processes, leading to therapeutic solutions in analogous processes in humans. A noteworthy and sophisticated example is shown in the CSMNA method [271], which is based on the hypothesis that similarities between human and plant metabolic networks can be used to guide phytochemical drug discovery. The authors validate their drug discovery algorithm by showing that similarities between the plant Halliwell-Asada (HA) cycle and the human Nrf2-ARE pathway underlie antioxidant activity of HA cycle molecules on proteins in the Nrf2-ARE pathway.

Some caveats need to be kept in mind when using evolutionary approaches. Certain classes of NPs are under evolutionary pressures that complicate phylogenetic analysis. Venom proteins, in particular, can be highly divergent even among species within the same genus [36], a phenomenon attributed to the high metabolic cost of venom production, and the highly targeted nature of many venom proteins to specific prey species.

1.3.4. Semantic (knowledge-based) methods

Cheminformatics and bioinformatics are two of the major disciplines within biomedical informatics, and comprise two of the primary fields involved in translational research and drug discovery. We now turn our focus to a set of methods that emerged from semiotics, linguistics, and library science, but have been adapted to serve broad functions in computer science and artificial intelligence—known as *knowledge-based* or *semantic* (i.e., relating to human-interpretable meaning) methods. In general, these are methods involving the application of various knowledge representations, such as ontologies and structured terminologies. Some activities within this group include rule-based natural language processing, certain types of clinical data mining, knowledge extraction, semantic data normalization, and others. Especially in the context of drug discovery, knowledge-based methods are frequently applied in coordination with bioinformatics and/or cheminformatics methods, and serve as one of the main approaches to combining and unifying findings and intermediate results spread across separate research activities.

Perhaps the most well-utilized resource in knowledge-based approaches to drug discovery is the Gene Ontology [8], which classifies conceptual biological entities into 3 groups: molecular functions, cellular components, and biological processes (each of which is important in various stages of the drug discovery process). Researchers have created multitudes of data resources to assist in drug discovery, and many of these are mapped to the Gene Ontology to assist with *in silico* aggregation and preliminary validation of putative hypotheses. Some of these linked resources include DrugBank [261], UniProtKB/Swiss-Prot (and associated annotation programs like ToxProt) [113], and ChEMBL [81], all of which catalog compounds that may confer some therapeutic effect.

Still other tools have been created to map unstructured data relevant to drug discovery (such as journal article abstracts in PubMed) to more structured representations. MetaMap, SemRep, and Semantic Medline from the National Library of Medicine, as well as the NCBO Annotator from the National Center for Biomedical Ontology identify ontology and terminology terms within free text (usually pulled from journal articles) at various levels of abstraction. These tools have been used to successfully perform ontological inference across multiple levels of evidence for many discovery tasks, including drug discovery. For further details, we refer the reader to the original paper describing Swanson's Fish Oil-Raynaud's Syndrome hypothesis [238], which explains how structured knowledge and graph algorithms can be used to discover informative associations fragmented across otherwise unrelated publications [38].

Other levels of knowledge representation (e.g., not formally controlled at the concept level) also have important roles in drug discovery; tools like OMIM can be used to map newly discovered drug-gene associations to diseases that are modulated by that gene or set of genes. For comprehensive listings of the various ontologies, knowledge representations, and similar tools with proven roles in drug discovery, we refer the reader to a number of existing reviews [79, 150, 244].

While the number of ontologies and similar resources relevant to drug discovery are vast, advanced applications of these resources are relatively scarce. This trend is even more striking in regards to NP drug discovery. As of now, most therapeutic associations between NPs and disease are discovered serendipitously rather than through systematic, rigorous applications, although earlier sections of this review describe notable exceptions to this trend. In light of the fact that advanced use of semantic methods is rare in NP drug discovery, we will additionally consider applications of ontologies and terminologies used for drug discovery that *could* be applied to NPs, based on current knowledge.

Literature mining

Literature mining—the process of using text mining on scientific literature databases—is one of the most common usages of semantic biomedical knowledge resources. The MEDLINE/PubMed database contains over 26 million biomedical text citations, many thousands of which contain knowledge related to NPs, and possibly describing characteristics of those NPs that provide direct or indirect evidence of therapeutic activity. There are generally two ways to automatically extract such knowledge from biomedical publications: (1) Using existing ontology/terminology annotations, or (2) using natural language processing (NLP) techniques that discover such annotations.

Medical Subject Headings (MeSH) are one terminology resource designed to structure the content of PubMed articles, and are applied manually by expert annotators at the US National Library of Medicine (NLM) to new articles shortly after indexing in PubMed [145]. MeSH terms cover a diverse range of biomedical concepts, arranged in a hierarchical fashion, and cover various classes of NPs. MeSH can be used to aggregate PubMed articles describing certain types of NPs, and can be refined using additional terms (e.g., "`Drug Discovery`") or qualifiers (e.g., "`/therapeutic use`"). MeSH terms can link journal entities to structured external databases by either using cross-mappings (including via the NLM's Unified Medical Language System (UMLS)) or annotations in external databases directly to MeSH terms [211]. MeSH terms have been used to summarize components of plant genomes [18], demonstrating potential paths forward in discovering novel NPs (rather than using the terms to gather knowledge about known NPs).

A limited number of databases provide access to curated sets of articles describing NPs. VenomKB provides articles annotated to venom components as well as literature predictions describing the putative therapeutic effects of those components and mappings to other external databases [205] (we will examine VenomKB in depth in **Chapter 2**). Similarly, the NPASS database presents chemical characteristics of a broader range of NPs and provides references to PubMed entries describing manually-curated biological activity measurements in a range of organisms (including humans) [270]. Other databases, including MarinLit and NAPRALERT, provide commercial and paid access to curated NP literature data.

Electronic health record mining

Similarly to literature mining, we can apply knowledge retrieval techniques to observational data sources. As far as drug discovery is concerned, observational data provides a method for assessing the effects compounds have on humans in the absence of rigorously controlled clinical research studies. This style of data analysis offers several major advantages over clinical trials, including avoidance of exposing new patients to potentially harmful treatments, and mitigating certain types of bias associated with eligibility and patient selection. Observational data can often produce larger cohorts than clinical trials. Various sources of observational data can be utilized for drug discovery, but here we will focus on electronic health records (EHRs), due to their prevalence and proven utility for many translational research tasks. Although privacy concerns, data fragmentation, and standardization have traditionally hampered access to EHR data—particularly for research teams without clinical expertise or affiliation with a large academic medical center—rapidly growing efforts such as Observational Health Data Sciences and Informatics (OHDSI) [103] and the Electronic Medical Records and Genomics (eMERGE) network [166] are breaking these barriers in ways that will increase access to data covering the breadth of the translational spectrum.

EHR data are complex, multimodal, and subject to many unique biases and ethical/legal constraints [255]. In addition to free text (recorded by health care providers), a number of structured data types are also present (including claims data, medication orders, laboratory measurements, patient demographics, and others). As of now, no major applications of EHR data mining to NP drug discovery have been reported, but a number of related areas provide hints as to its feasibility. A review by Yao et al. highlights 3 specific ways that EHRs can aid drug discovery: (1) Finding relationships between diseases for the purposes of drug repurposing, (2) evaluating the usage patterns and safety of drugs and/or drug candidates, and (3) discovering phenotype–genotype associations that can lead to the discovery of new drug targets for specific diseases [268]. Relevant caveats of each of these can be discussed from the perspective of NP drug discovery, including specific advantages and disadvantages that NPs provide when compared to non-NP drugs and drug candidates.

Drug repositioning involves taking an existing drug and using it to treat a different disease than what it is currently intended for [7]. EHRs have been used for a number of drug repositioning approaches. The most common repositioning strategy involves discovering similarities between diseases, and then using those similarities to imply new treatments. This is based on the assumption that diseases with similar etiologies will produce similar signals in the EHR, and that similar etiologies may imply similar treatments. An important example by Rzhetsky et al. showed unexpected similarity between bipolar disorder and breast cancer [215]. Recently, it has been demonstrated that the breast cancer drug tamoxifen may be useful for treating the symptoms of bipolar disorder [130].

EHR data can also be used to assess the safety of drugs (or putative drugs), by determining whether exposure to the drug increases risk of adverse effects [221, 240]. This is easiest for approved drugs that have coded representations in the EHR software (e.g., those with ATC codes or similar—experimental and unapproved drugs generally do not have a structured representation in EHR databases), but natural language processing can identify experimental and putative drugs with reasonable efficacy [25]. This suggests that NP drug-candidate safety surveillance could be performed on free-text notes in the EHR, especially when treated as environmental exposures rather than physician-prescribed interventions. The feasibility of this approach was demonstrated by Zhang et al., who showed that herbal and natural supplements (which are usually considered NPs) could be identified in medication lists using natural language processing, and quantified the gap between structured drug representations and these compounds [272]. Two of the main gaps in need of resolution to realize this goal include specifying a standardized nomenclature for NPs [57], and identifying where (geographically) hospital patients may be exposed to the NPs being investigated.

Discovering new drug targets is not strictly the same thing as drug discovery, but it does provide an essential starting point for identifying new drug leads. Recent decades have seen a steady decline in the discovery of new targets, and previous reviews on the topic have called for new and innovative strategies to address this issue [143, 230]. Using EHR data and clinical biobanks to conduct Genome Wide Association Studies (GWASs) and Phenome Wide Association Studies (PheWASs) are touted as solutions [268], by providing associative links between diseases and specific genetic loci, which can then be used as targets for new precision drug therapies [167, 258]. NPs, in particular, come into play when considering their unique abilities to target certain genes and gene products that are poorly targeted by small molecules. Both monoclonal antibodies and protein-based therapeutics are known for their ability to target individual cell types, especially useful in cancers with specific genetic signatures [2,49]. GWAS and PheWAS are relatively new compared to the drug discovery and development timeline, but we will likely see many NP drugs emerging from clinical trials that used EHR- and biobank-enabled analyses for target discovery in the coming decades [244].

Linking HTS data to putative disease treatments

Until now, we have discussed ways that ontologies and terminologies can be used to *retrieve* and *structure* knowledge, but another important role semantic techniques play in biomedicine is *integrating* disparate data sources in ways that otherwise require massive amounts of manual interpretation and annotation to apply at scale. This is important for many reasons, including experimental validation, increasing statistical power and inferential capacity, and even to discover new knowledge entirely. A particular application that has experienced rapid growth and major methodological advancements in drug discovery is linking new types of high-throughput sequencing (HTS) data to clinically-meaningful associations. Previously mentioned techniques such as gene expression perturbation (§1.3.3) yield results consisting of signals that have biological meaning, but no explicit connection to clinical phenotypes. Important early examples of data-driven drug discovery from gene expression formed therapeutic associations between cimetidine and lung adenocarcinoma [225], as well as topiramate and inflammatory bowel disease [64], but these examples required manual curation of many phenotype-linked expression profiles from which discovery could be performed. Knowledge representations provide a method for making these connections automatically, when correctly leveraged.

Successful knowledge integration of this type requires links to be formed between (a.) sets of genes (or, more specifically, groups of probe sets) and metabolic pathways, as well as (b.) links between pathways and phenotypes. A number of well-established and richly annotated gene-pathway databases (including Reactome and KEGG) [70, 117] already exist, and are used widely by the biomedical research community. Resources linking pathways to phenotypes are considerably less prevalent (and less complete), due largely to a limitation of available, relevant data, but ongoing efforts in the translational bioinformatics community are changing this. Integrating differences in gene expression and phenotypic response at the cell- and tissue-level with pathway data has shown particular promise in this area [86, 87].

A recent review by Oellrich et al. outlines emerging and established tools for computational phenotyping [187].

Similar studies are, however, nearly absent from the realm of NP drug discovery. The unique characteristics of different NP classes (especially those described earlier in this review) can facilitate the phenotyping process. Metabolomics data provides clues as to NPs' original functions in their source organisms, which can often be extended to their effects when applied to humans [264, 266, 271]. Phylogenomics can highlight similarities between the genetic epidemiologies of complex diseases in humans versus model organisms, possibly suggesting species from which to mine compounds that can treat these diseases [207]. Even the predator/prey adaptations of NP-producing species can suggest the biological function of NPs [54, 169]; the discovery that the cone snail *Conus geographus* hunts fish by releasing insulin into the surrounding water (resulting in rapid hypoglycemic shock in the prey) led to the identification of a powerful insulin-receptor-binding motif that has shown considerable promise for future treatments of diabetes [168]. Some recent studies focusing on discovery from TCM data show promise: Cui et al., for example, created a TCM chemical structure database that they screened against acetylcholinesterase (ACE) inhibitors, both via docking simulations with the known structure of ACE, as well as similarity to existing ACE inhibitors retrieved from BindingDB [51]. Conceivably, ontology resources could be used to adapt these methods into an automated approach for screening many drug classes with little to no manual curation.

Linking HTS data to disease phenotypes is only one application of semantic knowledge resources that could be a boon for NP drug discovery. There are many other conceivable uses for linking evidence between clinical datasets, drug terminologies, literature-mined associations, and organismal biodiversity data, any of which could lead to potentially valuable discoveries and improved evidence for unproven hypotheses.

1.3.5. Gaps and opportunities in NP drug discovery

Computers have revolutionized the way medicine and biomedical research are conducted, and the same applies to drug discovery. In doing so, it is critical to consider all of the ways in which computers can assist the discovery process in order to maximize the return on research efforts. In terms of *natural product* drug discovery, this review reveals that while some branches of informatics are being utilized extensively, other methods have not been fully explored. By summarizing 9 representative groups of informatics methods (see **Figure 1.4** and **Table 1.1**), we highlight these disparities and, by extension, areas of opportunity for future research.

Pharmacologists and the pharmaceutical industry have championed the use of advanced cheminformatics techniques in concert with cutting-edge biotechnology innovations. Although NP drug discovery has always been a hallmark activity in pharmacology, pharmaceutical researchers have only applied these cheminformatic techniques to NPs rather recently. Both QSAR (§1.3.2) and docking simulations (§1.3.2) are standard practice for studying the therapeutic potential and mechanisms of NPs. There is also a fair number of NP library studies (§1.3.2) that have been used to success—especially when focused on antibodies [101]—leading to the discovery of drugs such as adalimumab [109], ecallantide [163], and others [184]. As computing power improves, it is likely that we will see similar attention be paid to more challenging NP classes, such as venom peptides and other macromolecular compounds.

Bioinformatics demonstrates a similar trend, albeit somewhat earlier in its development (with regards to NP drug discovery) than cheminformatics. The bioinformatics methods covered in this review are intriguing in that each is a technique originally intended for uses other than drug discovery. Differential gene expression analysis (§1.3.3) was originally used to explore differences between cell lines and disease states rather than the effects of drug perturbation, although the conceptual jump in applying expression analysis to drug discovery is arguably an obvious one. However, due to this technique's relatively recent emergence, few examples using NPs (as opposed to non-NP small molecule candidates) currently exist in the literature, none of which are truly data-driven (i.e., agnostic to both specific diseases and specific NP drug candidates). Nonetheless, analyses targeted towards specific diseases compared against the Connectivity Map dataset have resulted in two substantial discoveries based on plant metabolites: Celastrol as a treatment for acute myeloid leukemia [92], and gedunin as a treatment for prostate cancer [97]. Therefore, the preliminary groundwork for truly data-driven drug discovery for NPs via perturbational differential expression analysis has already been established. For further examples of the successes of the Connectivity Map approach to data-driven drug discovery overall, we direct readers to a previous review by Musa et al. [176]. Phylogenetics (§1.3.3)—one of the earlier uses for computers in biology—has become known for its diverse areas of application, including drug discovery. Since NPs come from organisms that can be studied in a phylogenetic context, bioinformaticians have realized just how valuable of a tool this can be for NP drug discovery, and a number of completed and ongoing research initiatives capitalize on this.

Semantic methods have been used much less frequently for drug discovery than the other branches of informatics, and even less so for NPs. Only a few sparse examples of literature mining applications (§1.3.4) exist for NP drug discovery. A few studies show that ontologies and similar methods that link experimental evidence to HTS data and structured knowledge representations (§1.3.4) could easily be adapted to perform preliminary validation for expensive and time-consuming manual experimentation to prove therapeutic activity in NPs, but the actual use of these methods for this purpose is also virtually nonexistent. EHRs and other clinical data resources are in a similar situation—as far as we can tell, there are currently no published examples of clinical data mining (§1.3.4) being used to discover therapeutic associations from NPs.

1.3.6. Data needs for NP drug discovery

Throughout this review, we have touched upon computational and informatics methods with varying data needs, and have naturally mentioned several data resources that are dedicated to (or have strong relevance to) NP drug discovery. Just as certain discovery methods are enabled by characteristics specific to NPs, certain data types and dimensions are as well. This includes taxonomic/evolutionary data [48, 132], primary (i.e., "intended") targets and functions of NPs in nature [22], the crude composition of NPs (often leading to synergistic effects, analogous to drug combination therapies) [29, 39], and others specific to particular classes of NPs. A more comprehensive description of NP databases is presented in a review by Xie et al. [265], but here we will cover some of them in brief as they pertaining to specific data needs.

The diversity and complexity of data types relevant for NP drug discovery research poses challenges in storing, representing, and exchanging these data. An immediate consequence is that many NP databases are limited to a narrow range of closely related NPs, which results in data fragmentation for the sake of completeness [259]. ConoServer [116] and ArachnoServer [192] are two NP databases with rich and highly descriptive data, but each only applies to toxins produced by a single clade of species. One partial solution to this problem is to form dedicated efforts *within* larger, more general purpose databases that are dedicated to improving the representation of NPs, which is the approach taken by the Tox-Prot manual annotation program within UniProtKB/Swiss-Prot [113]. However, this does not completely resolve the greater issue of being able to leverage all important data types that are unique to certain classes of NPs. One other advantage that larger database efforts have over smaller, specialized NP databases is the presence of APIs and other tools that enable computational access. Many of the specialized databases do offer the ability to download data in bulk, but these can be incomplete and out-of-date. Furthermore, APIs can assist in making databases interoperable—an integrated network of specialized and well-annotated databases that can exchange semantic knowledge solves the issue of adequately representing granular characteristics while providing many of the benefits of larger data repositories.

Fragmentation of NP databases has also led to issues in maintaining those databases in the event of funding inconsistencies and institutional career changes—an issue that is at least partially safeguarded against when data resources are maintained by larger teams with more robust operating budgets. Three examples of now-defunct NP databases are the Traditional Chinese Medicine Systems Pharmacology (TCMSP) database [210], the Animal Toxin Database (ATDB) [93], and the SuperNatural database [65]. Smaller NP databases can also suffer from issues like having unwieldy and non-descriptive URLs, such as that for the Tea Metabolome Database (found at `http://pcsb.ahau.edu.cn:8080/TCDB/f`) [269]. Furthermore, if ownership of such a database changes, or if the principle investigator moves to a new institution, the URL would likely break, creating issues in finding the database when reading the manuscript that describes it—a phenomenon sometimes referred to as “link rot” [164].

Taking into account these and related issues, a wealth of opportunity is available for informatics researchers and data scientists to improve the quality, quantity, and interconnectedness of NP databases and knowledge representations. In the following section, we will reiterate these and other areas of importance for the near future, as elucidated over the course of this review.

1.4. Semantic knowledge resources

1.4.1. Knowledge representations

Computers are, fundamentally, devices that read series of arithmentic and/or logical instructions and then carry those instructions out. By combining and layering these instructions in increasingly complex ways, computer programmers ascribe meaning to the operations in ways that correspond to human knowledge [260]. Knowledge—the *meaningful* understanding of something or someone, acquired through perceiving or learning—implies forming logical connections between concepts and the real-world objects and/or phenomena that those concepts represent. The computers of today cannot truly obtain knowledge in the strict understanding of the word, but they can be taught the associations between real-world concepts that define their meaning, and in doing so, can achieve a convincing approximation of knowledge about concepts. *Semantics* (specifically, formal semantics) is the branch of logic concerned with defining the meaning of concepts through logical axioms and relationships with other concepts, and it provides the toolkit needed to represent knowledge in a way that can be serialized and reasoned over by computer systems.

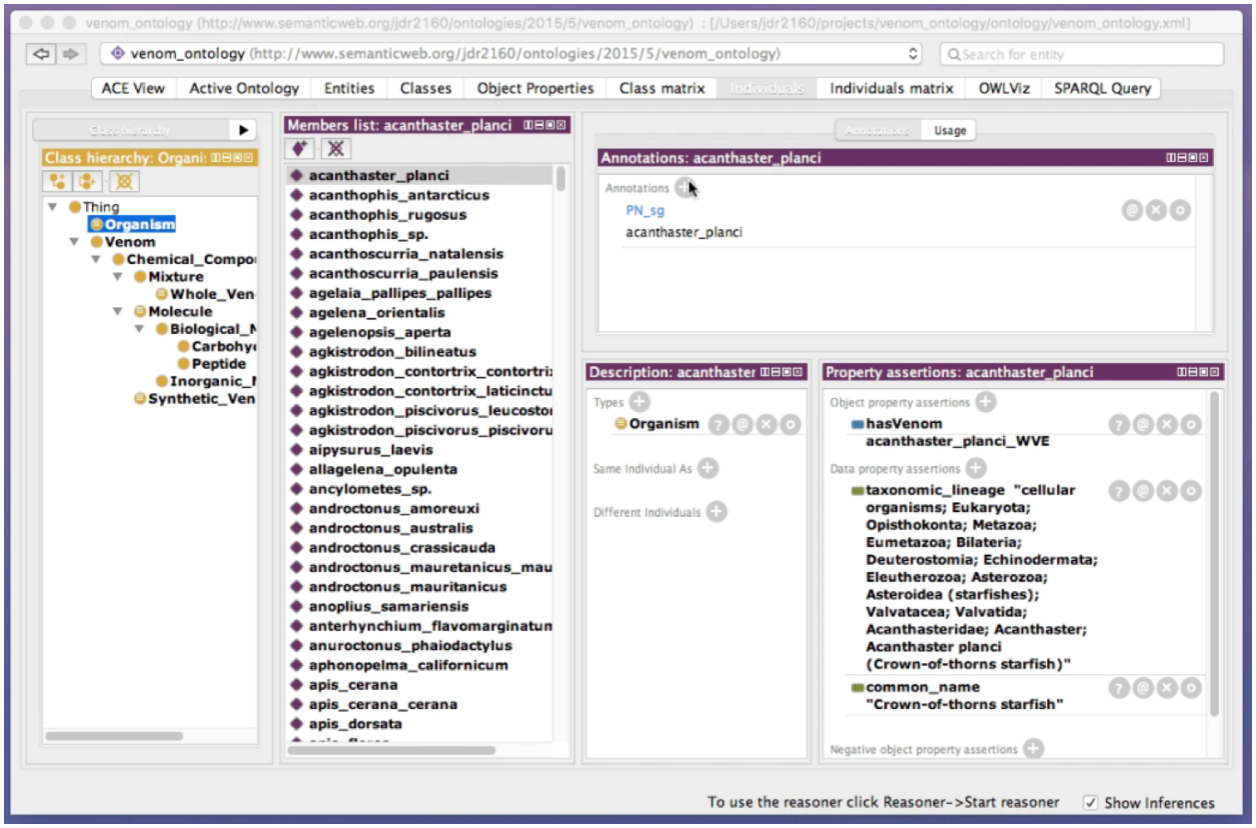
There are different types of semantic knowledge representations that can be used to represent biomedical data and/or knowledge, each of which has advantages and disadvantages in different scenarios. These vary in their level of *expressiveness* as well as *formality*. More expressive representations of knowledge can be thought of as conveying more meaning. Formal knowledge representations are those that strictly follow a well-defined system of thought based on logical axioms that link a finite set of *symbols*, such as unique identifiers or names.

The least formal type of knowledge representation is a *database*, which contains data records that follow a consistent, ordered structure, and cannot be used for inference. *Structured terminologies* provide additional expressivity, by arranging entities in a configuration that specifies subsumption relationships between those entities (also known as `IS_A` relationships.) A *semantic network* is similar to a structured terminology in that links define their meaning, but the links are not necessarily directed, and they typically assume additional meanings instead of only subsumption (e.g., 'John `HAS_DIAGNOSIS` type-2 diabetes'). An *ontology* is the most expressive form of knowledge representation. Although the definition of an ontology somewhat varies, they generally consist of a semantic network that defines entities and their relations, along with additional formally defined data annotations attached to those entities. Additionally, they usually contain both hierarchical and nonhierarchical relationships. Ontologies closely resemble class hierarchies from object-oriented programming languages, and can contain many similar features. The most common variation of ontology in practical applications today is known as an OWL (Web Ontology Language) ontology. In OWL ontologies, *classes* group together similar entities that have the same set of attributes and the same relations to entities of other classes, and specific instances of these entities are known as *individuals*. The links between entities are called *object properties*, while annotations to direct pieces of data are called *data properties*. OWL ontologies comprise a major feature of the so-called 'semantic web', and they can be developed and interacted with using software such as Protégé.

1.4.2. Constructing and using ontologies

In practical terms, the process of building an ontology usually consists of first specifying the *ontological commitment*, which is a definition of the extent of the ontology's intended capabilities, in terms of the domain of knowledge it covers. Once this is determined, the builder then defines the class hierarchy, adds nonhierarchical relationships, specifies data annotations and their correct data types, and then fills in individuals for each class. Finally, ontology reasoning software should be used to ensure that the individuals are defined correctly and have all of the required object and data properties as required by the specifications laid out in the class hierarchy.

Rich semantic knowledge representations like OWL ontologies allow for many useful types of inference to be performed on data. Since they define links between real-world entities in a way that computers can process, they can be used to automatically answer complex questions that convey both meaning and new knowledge. Aside from the reaserch I discuss in **Chapters 2 and 4**, there is no such suitable representation for venom data. The only existing ontologies that contain any terms regarding venoms (such as the Gene Ontology and



venom_ontology (http://www.semanticweb.org/jdr2160/ontologies/2015/5/venom_ontology) : [/Users/jdr2160/projects/venom_ontology/ontology/venom_ontology.xml]

venom_ontology (http://www.semanticweb.org/jdr2160/ontologies/2015/5/venom_ontology)

Thing
Organism
Venom
Chemical_Compo
Mixture
Whole_Ven
Molecule
Biological_N
Carbohy
Peptide
Inorganic_I
Synthetic_Ven

acanthaster_planci
acanthophis_antarcticus
acanthophis_rugosus
acanthophis_sp.
acanthoscurria_natalensis
acanthoscurria_paulensis
agelaia_pallipes_pallipes
agelena_orientalis
agelenopsis_aperta
agkistrodon_bilineatus
agkistrodon_contortrix_contortri
agkistrodon_contortrix_laticinctu
agkistrodon_piscivorus_leucostoi
agkistrodon_piscivorus_piscivoru
aipysurus_laevis
allagelena_opulenta
ancylometes_sp.
androctonus_amoreuxi
androctonus_australis
androctonus_crassicauda
androctonus_mauretanicus_mau
androctonus_mauritanicus
anoplius_samariensis
anterhynchium_flavomarginatun
anuroctonus_phaiodactylus
aphonopelma_californicum
apis_cerana
apis_cerana_cerana
apis_dorsata

PN_sg
acanthaster_planci

Organism

hasVenom
acanthaster_planci_WVE

taxonomic_lineage "cellular organisms; Eukaryota; Opisthokonta; Metazoa; Eumetazoa; Bilateria; Deuterostomia; Echinodermata; Eleutherozoa; Asterozoa; Asteroidea (starfishes); Valvatacea; Valvatida; Acanthasteridae; Acanthaster; Acanthaster planci (Crown-of-thorns starfish)"

common_name "Crown-of-thorns starfish"

Figure 1.5.: Screenshot of an OWL ontology (specifically the Venom Ontology—see §2.2.3) in the Protégé ontology editor. The ontology's class hierarchy is in the leftmost panel, with the class `Organism` selected. Members of that class are listed in the second panel from the left. The remaining panels show annotations and other properties `acanthaster_planci`, an individual that is a member of the class `Organism`.

SNOMED-CT) have too broad ontological commitment, and therefore have limited coverage and value. Scenarios such as that described by Don Swanson in 1986 (in which literature mining illuminated a link between fish oil supplements and Raynaud's syndrome) could be commonplace in venom data, and using ontologies to integrate new venom data with existing data could provide many hypotheses for drug discovery purposes.

1.5. Data-driven discovery from gene expression data

1.5.1. Data-driven science

The traditional paradigm for scientific investigation involves posing a hypothesis, and then attempting to disprove that hypothesis. Well-established scientific theories are based on failing to disprove similar hypotheses many times, under varying experimental conditions. Usually, hypotheses are made using some amount of prior supporting knowledge. For example, the observation that *Conus geographus* (geography cone snail) venom suppresses the sensory circuitry of prey fish in a way that mimics hypoglycemic shock led to the (now well-established) hypothesis that this venom contains specialized insulins that could be used for therapeutic applications in humans [168, 216]. This approach to scientific discovery provides many substantial advantages to the scientific process, largely by reducing certain types of bias and by allowing a methodical approach to building important research on many layers of supporting prior evidence, theoretically increasing both the precision and the recall of new discoveries.

One of the most substantial disadvantages to hypothesis-driven science is that it requires researchers to conceive of scientific phenomena (or at least the outcomes of those phenomena), which limits scientific discovery primarily to concepts and applications that are at least peripherally being focused on by existing investigations. The main exception to this is serendipitous discovery, where some new phenomenon is initially observed or hinted at as a matter of coincidence. A widely known example of this is the discovery of the antibiotic properties of *Penicillium* mould, as reported by Alexander Fleming in 1929, when he noticed that cultures of staphylococcal bacteria cleared in the area surrounding mould spores that

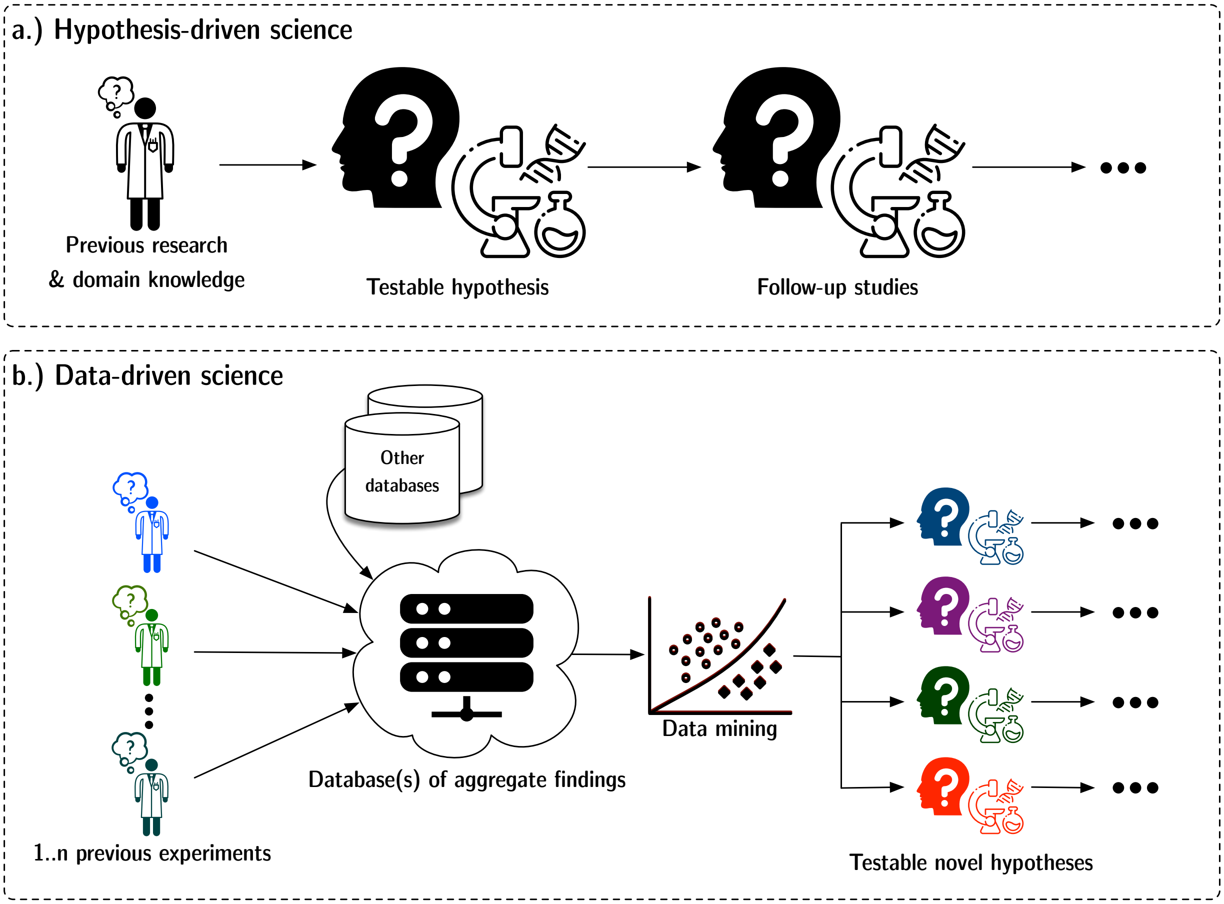


Figure 1.6.: Hypothesis-driven vs. data-driven science. a.) Hypothesis-driven science follows a linear path, where previous studies build a basis for formulating new hypotheses. b.) In data-driven science, data are collected and then mined for statistically significant patterns, which can then be used as the foundation for hypothesis-driven lines of research.

accidentally contaminated the plate [74]. Although this initial discovery was made by matter of accident, it led to a careful and highly impactful line of research—supported by many subsequent hypothesis-driven investigations—that would ignite the modern era of antibiotic treatment, saving countless lives and becoming one of the major pillars supporting today's pharmaceutical industry [239].

A compromise between hypothesis-driven science and exploratory analyses that can uncover entirely novel discoveries is a relatively new approach known as *data-driven science*, which is summarized in **Figure 1.6**. In data-driven science, rather than posing a single hypothesis and then seeking to disprove that hypothesis, investigators instead collect large sets of data and use statistical techniques to highlight patterns in the data that may correspond to nonrandom scientific phenomena. Subsequently, researchers can then gather these patterns and convert each of them into traditional hypotheses that can be explored individually, using the well-established hypothesis-driven paradigm.

Data-driven science shows up in two parts of this dissertation: (1) In `VenomSeq`, we use data-driven methods to find new associations between venoms and drug classes / disease, and (2) VenomKB is meant to be a tool that facilitates data-driven approaches to discovery from venom data, both by providing that data in a structured form where venom-related entities can be compared to one another easily, and by presenting tools to the users that help provide supporting evidence to accompany putative discoveries that are uncovered by data driven methods (either using data from VenomKB/`VenomSeq` or from elsewhere).

1.5.2. Differential expression analysis

The human genome contains some 20,000 protein-coding genes, in addition to a vast number of regulatory elements and other stretches of DNA with unknown functions. While each cell in the human body contains the same genes (with some exceptions), differences in cell types (functional and morphological) are largely the result of those genes being expressed in different amounts—and in different combinations—based on the characteristics and the needs of the cell. Beyond differences in cell and tissue type, gene expression can be altered at the level of *individual* cells due to specific chemical and physical influences, such as injury, disease (e.g., viruses that use cells for reproduction), hormone signaling, and perturbation by exogenous chemicals, including toxins, dietary nutrients, and pharmaceutical drugs. When these types of perturbations initiate a specific, targeted mode of action (rather than global, nonspecific cellular processes, such as DNA damage resulting from exposure to ionizing radiation), the effects on differential expression tend to be reproducible and correlated with the underlying changes in the cell.

From a broader perspective, differential expression analysis is generally used for the following purposes:

1. Determining differences in gene expression between cell lines or tissue types.
2. Determining differences in gene expression between healthy and diseased cells.
3. Determining differences in gene expression between perturbed (e.g., by drugs or genetic alterations) and non-perturbed cells.

The last of these (perturbed vs. non-perturbed cells) can be further divided into applications for learning the mechanisms of certain perturbagens, as well as discovering new drugs or drug candidates that have a ‘therapeutic-like’ effect on expression.

Aside: Platforms for gene expression analysis

Changes in expression can be measured (technically, estimatated, since we do not directly count every transcript in a cell) via a number of different approaches. *DNA microarrays* are one of the most widely used of these, due to their ease of use and relatively low cost [124]. DNA microarrays are glass slides with a large number of short nucleotide oligomers (small fragments of DNA) arranged in a grid. In microarray analysis, mRNA (i.e., actively transcribed genes) extracted from a population of cells are reverse transcribed into cDNA which is then fragmented and labeled with fluorescent tags. These labeled fragments are passed over the surface of the microarray, and the fragments bind to their complementary sequences on the array. After binding, the intensity of each probe's fluorescence is proportional to the level of expression of the corresponding mRNA sequence in the cell population of interest. While microarray experiments have proven to be very successful for many purposes (especially when performing high-throughput screening of many cell populations simultaneously), it suffers from a number of limitations, most notably that quantifying fluorescence instead of counting the number of transcripts directly results in an additional contributing factor to estimation error [43].

A more expensive technique with higher resolution is *RNA sequencing* (RNA-Seq), which involves collecting and reverse-transcribing mRNA in a sample (like with microarrays), but then performing sequencing on the cDNA fragments to obtain individual reads corresponding to a particular isolated mRNA fragment [188]. Computational techniques (some of which are described in §3.4.4) are then used to determine the number of reads for each gene being measured, yielding a random sample of the actual transcripts, rather than a fluorescence value that is merely proportional to this number. The processed results of

RNA-Seq are *counts* for each gene in the human genome.

A third and arguably less popular approach uses an experimental procedure known as quantitative polymerase chain reaction (*qPCR*). qPCR is an older technique, and is the most low-throughput of the three I have discussed. In qPCR, mRNA is amplified using polymerase chain reaction, and then separated into individual transcripts via electrophoresis. The bands are then isolated, and then by allowing hybridization with probes and detection via autoradiography, the researcher is able to determine the identity and the relative abundance of the transcript in that band [249]. Obviously, this method is extremely laborious and unsuitable for analysis of many different transcripts (e.g., entire transcriptomes), since each transcript is handled in a distinct experiment.

PLATE-Seq

As I suggested in the previous section, one of the primary barriers to wide adoption of RNA-Seq for exploratory analysis is cost. In recent years, a number of innovative biotechnology and informatics approaches have been designed to address this issue. One of these—a protocol named PLATE-Seq—was developed as a joint effort in the Califano and Sims labs at Columbia University [32]. Whereas traditional RNA-Seq involves isolating transcripts and performing individual sequencing runs on each sample, PLATE-Seq instead pools many samples together in a single sequencing run, after using barcoded primers unique to each sample in the reverse transcription process. After performing sequencing on the single pooled sample, the reads are then matched back to their original sample based on the identity of the barcoded adapter at the end of the sequence. PLATE-Seq is the specific sequencing technology we use with `VenomSeq`, which is described in **Chapter 3**.

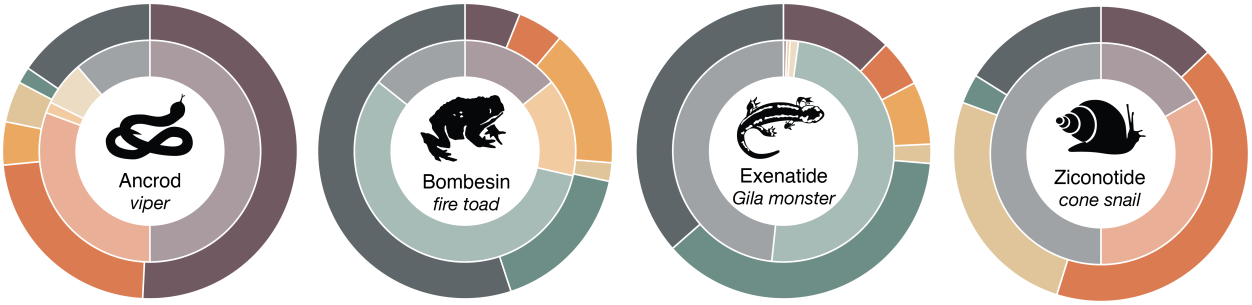
The PLATE-Seq approach is primarily limited by a decrease in resolution (quantified by the average depth of any given transcript) over traditional RNA-Seq, resulting from the fact that any single sample (pooled or otherwise) has a technical limit on the number of total reads in that sample that can be sequenced[2]. Therefore, it is more challenging to assure sufficient coverage over all areas of the genome, particularly at high enough frequencies to ensure detection of sequencing errors. However, recent analyses have demonstrated that the resolution of PLATE-Seq is plenty sufficient for all but the most stringent clinical applications, where the goal is to obtain high confidence estimates for a small number of transcripts (e.g., discovering specific mutations occurring in a certain gene). In applications that aim to identify large-scale trends (such as disregulation of large sets of functionally related genes), this limitation is relatively inconsequential.

Additionally, the authors of PLATE-Seq have been involved in developing additional tools to augment the results in ways that compensate for the decreased number of reads. One of the most importance of these is an algorithm known as *VIPER* (Virtual Inference of Protein activity by Enriched Regulon Analysis) [4]. The VIPER algorithm accepts gene expression data (such as those returned by PLATE-Seq) and a previously determined network of gene regulation specific to the cell line used for producing the PLATE-Seq data. Using network inference, the expression levels of master regulators for that cell line can be estimated within known bounds, 'filling in the gaps' left by the lower resolution data.

[2] This limit varies based on the sequencing platform used, but is present in all currently available platforms.

Chapter 2.

An informatics infrastructure for computational toxinology



2.1. VenomKB v1

2.1.1. Introduction

We begin by introducing VenomKB—an online knowledge base that is designed to facilitate the emergence of computational techniques to investigate therapeutic uses for venom compounds. As of its first release (v1.0), VenomKB consisted of three database tables. The first is a manually curated list of putative and active (i.e., in clinical use) venom therapies. The second and third detail the outputs of two different algorithms (VExtractor and SemanticVExtractor) that were used to automatically extract (by natural language processing or knowledge discovery techniques) putative venom therapies in a corpus of abstracts from the

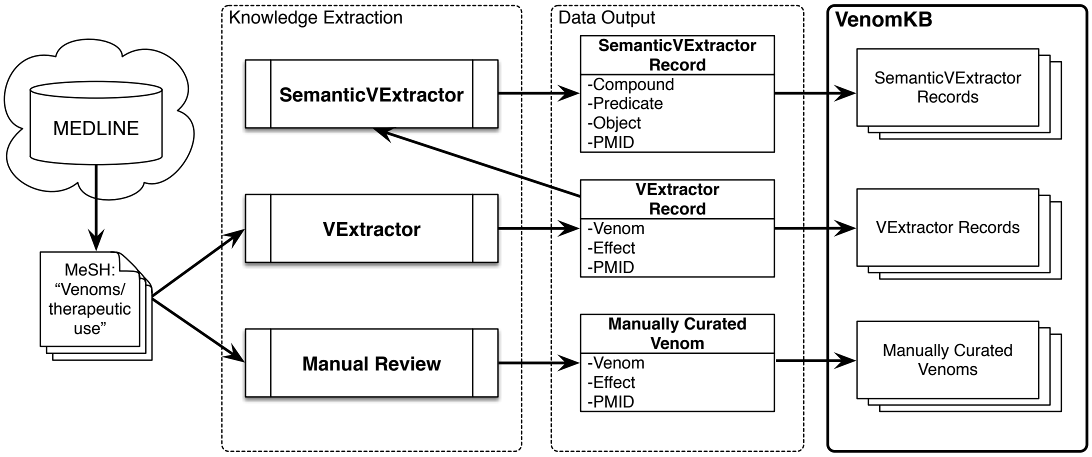
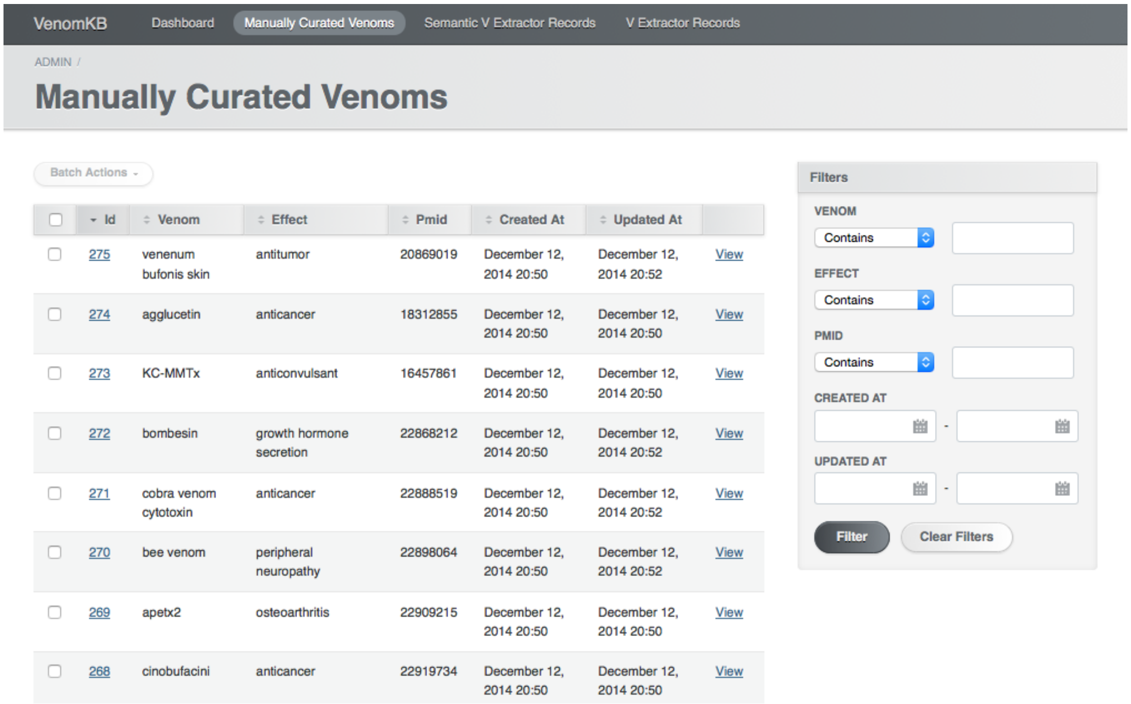


Figure 2.1.: MeSH was used to identify a core set of relevant articles, which were then passed to three methods of knowledge extraction (manual review, the VExtractor algorithm, and the SemanticVExtractor algorithm). The outputs of these three methods were then collected and assembled as VenomKB.

scientific literature. A schematic outlining the processes of data collection and curation in building v1.0 is shown in **Figure 2.1**. VenomKB is an open-source and publicly accessible resource for researchers and other individuals interested in venom therapeutics and may be accessed at the project's official website (`http://www.venomkb.org`). The website contains a tabular interface for searching, sorting, and viewing the different records in each database table, and data records of interest may be selectively downloaded in CSV, XML, and JSON formats, as desired. Additionally, a "frozen" copy of the data as it exists at the time of v1.0's publication can be found on FigShare (see §2.1.6 for individual data citations). At this time, the knowledge base contained 42,723 unique records.

The original goals of VenomKB were twofold: (1) to make it easier to discover proposed or suggested venom therapies for a disease of interest (or vice-versa), and (2) to facilitate the identification of studies on established venom therapies in order to guide the study of newly



Manually Curated Venoms

□	Id	Venom	Effect	Pmid	Created At	Updated At	
□	275	venenum bufonis skin	antitumor	20869019	December 12, 2014 20:50	December 12, 2014 20:52	View
□	274	agglucetin	anticancer	18312855	December 12, 2014 20:50	December 12, 2014 20:50	View
□	273	KC-MMTx	anticonvulsant	16457861	December 12, 2014 20:50	December 12, 2014 20:50	View
□	272	bombesin	growth hormone secretion	22868212	December 12, 2014 20:50	December 12, 2014 20:52	View
□	271	cobra venom cytotoxin	anticancer	22888519	December 12, 2014 20:50	December 12, 2014 20:52	View
□	270	bee venom	peripheral neuropathy	22898064	December 12, 2014 20:50	December 12, 2014 20:52	View
□	269	apetx2	osteoarthritis	22909215	December 12, 2014 20:50	December 12, 2014 20:50	View
□	268	cinobufacini	anticancer	22919734	December 12, 2014 20:50	December 12, 2014 20:50	View

VENOM: Contains
EFFECT: Contains
PMID: Contains
CREATED AT: -
UPDATED AT: -

Figure 2.2.: Image shows the first 8 records of the 'Manually Curated Venoms' table. The top bar has links to the knowledge base home page and each of the three current database tables. Search filters are in the frame entitled 'Filters' on the right side of the interface. Download links (for CSV, XML, and JSON format) and pagination functionality are located at the bottom of the page, out of range of the screenshot.

discovered or newly classified venoms. Since the discipline of computational approaches for discovering venom therapies is emergent, we expected VenomKB to grow rapidly in the future. We encourage interested users to monitor additions and changes to the knowledge base by viewing the website's home page (`http://www.venomkb.org`), which will be updated in the event of all major developments within both the knowledge base and the field of study as a whole.

2.1.2. Methods

The MEDLINE biomedical literature repository contained 22,376,811 searchable titles and abstracts as of its 2014 release (when v1.0 of VenomKB was first assembled). We used this resource to extract all venom therapies, established or hypothetical. We found 5,117 relevant articles using the Medical Subject Heading (MeSH) term `Venoms/therapeutic use`. We saved the abstracts in MEDLINE format—a text-based format that includes the article titles, abstracts, and important metadata records for each article. We then applied three separate methods for extracting data regarding putative venom therapies on these data—the first of these was manual review and curation of journal articles and the second and third were computational algorithms that automatically extracted relevant knowledge from the pre-filtered set of MEDLINE articles.

To manually curate the abstracts, we first randomized the order of the 5,117 journal articles (to avoid bias by only selecting articles from a short span of time) and selected the first 275 records that describe putative venom therapies. We skipped articles that were incorrectly tagged with the MeSH term `Venoms/therapeutic use` or where the proposed venom therapy was unclear or subjectively deemed insignificant. We also ignored articles that described venom immunotherapy—a technique that involves reducing sensitivity to venoms (typically bee venoms) by administering small dosages of the venom over time in order to desensitize the immune response [6]. For abstracts that we determined contained valid putative venom therapies we recorded the data in the following format:

$$\langle \textit{venom} \rangle \mid \langle \textit{physiologic effect} \rangle \mid \langle \textit{PubMed ID (PMID)} \rangle$$

In the first of the two automatic knowledge extraction methods, called VExtractor (see **Appendix D**; filename `vextractor.py`), we used the NCBO BioPortal Annotator API to extract ontology terms from the text of the title and abstract for each of the 5,117 entries. We then filtered the annotations by chosen ontologies and Unified Medical Language System (UMLS) semantic types that have a high likelihood of selecting venom compounds and physiological effects of the compounds on the human body. The ontologies and semantic types we used are documented in full in the repository listed in **Appendix D** (filename `vextractor_ontologies_and_semtypes.txt`). We selected these ontologies and semantic types based on knowledge of ontology contents and a trial-and-error process of manually altering the filtering strategy and observing if the algorithm was able to precisely identify venom compound names and physiological effects as was determined for five of the manually reviewed articles, selected randomly. Finally, VExtractor then sorts these terms into the appropriate data structures (see below) and returns them as output. The NCBO annotator can identify more than one venom compound and/or physiologic effect, so all were recorded for loading into the knowledge base. This application was run for each of the 5,117 originally identified journal articles as input—a task facilitated by the ability of VExtractor to accept a list of PMIDs and run the script for each one, returning the results as two comma separated value (CSV) text files, in the following format:

$$\langle \mathit{PMID} \rangle \mid \langle \mathit{venom\ compound} \rangle$$

$$\langle \mathit{PMID} \rangle \mid \langle \mathit{physiologic\ effect} \rangle$$

In this format, the results could be interpreted as a list (0 or more) of both potential venom compounds and effects. The outcome is greater flexibility in interpretation of the knowledge contained in a given article, at the expense of potentially losing resolution if multiple venom compounds—each with unique effects on the human body—are discussed in the same journal article. Finally, we combined these two lists using a Ruby script (see **Appendix D**; filename `make_vextractor_table.rb`) into a single list with records in the format:

$$\langle \textit{venom compound} \rangle \mid \langle \textit{physiologic effect} \rangle \mid \langle \textit{PMID} \rangle$$

We performed the second automated knowledge extraction method (a workflow we named 'SemanticVExtractor') using a program named 'SMDB_Search' (see **Appendix D**; directory `smdb_search/`). SMDB_Search is a utility that connects to a local copy of the new Semantic MEDLINE (SemMedDB) resource—a database that enables searching MEDLINE by semantic concept rather than a traditional search query—and extracts semantic predicates for either a given list of PMIDs or a certain UMLS or Gene Ontology (GO) term. For the purposes of this study, we used all of the PMIDs that returned valid records in the VExtractor procedure (i.e., all for which at least one possible venom compound and at least one effect) as input for SMDB_Search. The output of SMDB_Search was a list of Java Script Object Notation (JSON) formatted data structures. It should be noted that instead of recording the output values as 'potential venoms' and 'effects' (as was done for the previous two knowledge extraction methods), we recorded them as 'subject' and 'object', since venoms and effects can occur in either order (e.g., `venom_x treats condition_y` versus `condition_y treated_by venom_x`)—see §2.1.3 for further explanation. In addition to the subjects and objects, we recorded the predicate phrase and the UMLS semantic types for each the subject and the object, to allow for more detailed analysis of data results in future

Table 2.1.: UMLS Semantic Types for filtering SemanticVExtractor output.

UMLS Semantic Type	**4-letter abbreviation**
Amino Acid, Peptide, or Protein	aapp
Amino Acid Sequence	amas
Biologically Active Substance	bacs
Body Substance	bdsu
Chemical	chem
Chemical Viewed Functionally	chvf
Chemical Viewed Structurally	chvs
Clinical Drug	clnd
Eicosanoid	eico
Enzyme	enzy
Hazardous or Poisonous Substance	hops
Hormone	horm
Immunologic Factor	imft
Nucleic Acid, Nucleoside, or Nucleotide	nnon
Neuroreactive Substance or Biogenic Amine	nsba
Organophosphorus Compound	opco
Pharmacologic Substance	phsu
Substance	sbst

additions to VenomKB. Finally, we used these semantic types to filter the output values of SemanticVExtractor—only data records with subject semantic types listed in **Table 2.1** were retained, because those semantic types are the ones that logically may be assigned to venom compounds. Like with the VExtractor method, we designed SemanticVExtractor with the ability to return 0 or more data records for each journal article. In order to remove a large number of the false positives identified by the two automated methods, we performed a manual review of the database contents, removing obviously erroneous entries. This included compounds that are not related to venom compounds (e.g., 'insulin treats type-2 diabetes mellitus', and uninformative/nonsensical entries (e.g., 'medications treat patients' or 'venoms are venoms').

2.1.3. Description of data in VenomKB v1.0

The results of the data collection methods described above are all available on the knowledge base website (`http://www.venomkb.org/`), as well as in a public FigShare repository (see individual data citations below) for the purposes of data permanence and reproducibility of data integrity analyses that we have performed (see §2.1.4). While the data records on FigShare are static, the content of the knowledge base itself will change over time as data records are validated/invalidated and as new knowledge extraction methods are developed for the emerging field of computationally-predicted venom therapies. To create the data files on FigShare, we exported the complete contents of the three relevant PostgreSQL tables as CSV-formatted files, where the first line of the file consists of the headers describing each data field, and each line thereafter represents a single data record. All of the tables include the following records: `id` (a unique numerical identifier), `pmid` (the PubMed identifier for an article supporting the data record), `created_at` (the date and time at which the record was added to the database), and `updated_at` (the date and time at which the record was most recently modified, which is identical to the contents of `created_at` in many cases). Each of the three individual files is described below, with the addition of the other fields that are unique to each table.

The manually vetted putative venom therapies ('Manually Curated Venoms') are stored in a file named `manual_venoms.csv` (see §2.1.6; Data Citation 1). A sample of the first three records is shown in **Table 2.2**. This table contains two unique fields: `venom` and `effect`. `venom` is the name of the venom compound. These names may or may not be a trade name,

Table 2.2.: Sample data from 'Manually Reviewed Venoms' table.

ID	venom	effect	PMID
1	bombesin	gastric secretion	11996
2	ancrod	claudication	66429
3	ancrod	deep vein thrombosis	80632
...	...	...	...
n	*[venom n]*	*[effect n]*	*[PMID n]*

a compound name, or some other name, but they reflect the name used in the associated journal article. It should be noted that this is not an arbitrary design decision—since there is no standardized naming format or classification system for venom components (e.g., the compound EMD 121974—a modified snake venom protein—is almost ubiquitously referred to by the trade name Cilengitide), the most methodical approach is simply to preserve the name(s) given by the author of the journal article. `effect` is the primary purported physiologic, molecular, or phenotypic effect or target of the venom. However, this is not explicitly qualified—for example, a venom compound that is reported as `effect` being 'Parkinson's disease' likely intends to mean that the venom treats Parkinson's disease, not that it causes the disease. Although this introduces some ambiguity into the database, it was a design choice made to facilitate easy searching for diseases and molecular targets via the web interface. The data for the first automated knowledge extraction algorithm—which utilizes the NCBO Annotator API; named VExtractor—is contained in the file named `vextractor.csv` (see §2.1.6; Data Citation 2). A sample of the first three records is shown in **Table 2.3**. Aside from the common fields mentioned above, these data records have two additional fields: `venom` and `effect`. Like with the other methods, `venom` describes the venom or venom component being discussed. Since these terms were automatically extracted and standardized to

Table 2.3.: Sample data from ‘VExtractor’ table.

ID	venom	effect	PMID
1	ceruletide	tachyphylaxis	11996
2	ceruletide	gastric secretion	11996
3	ceruletide	pancreatitis	2717605
...	...	...	...
n	*[venom n]*	*[effect n]*	*[PMID n]*

the terms contained in the UMLS, the naming scheme is more consistent than in the Manually Curated Venoms table. `effect` is similar to the equivalent field in Manually Curated Venoms, but it is more commonly a disease or an observable physiological effect rather than a molecular mode of action or molecular target. Likewise, although the effect is often listed as a disease name, it should usually be interpreted as treating that disease rather than causing it. This should, however, be done with regard to context: venom compounds in many situations may in fact be the cause of particular generalizable diseases (e.g., pancreatitis as a result of *Tityus trinitatis* scorpion envenomation [14]). For this reason, we urge users to refer to the supplied PubMed IDs when looking at individual VenomKB records. The label for the column was not chosen to be `treats` because the field does not always describe a treatment. If there is any ambiguity in a data record of interest, it is strongly recommended to view the cited PubMed article to determine the exact context of the therapeutic effect of the venom. Formatted output from the second automated method—using the SMDB_Search utility; named SemanticVExtractor—is contained in a file named `semantic_vextractor.csv` (see §2.1.6; Data Citation 3). A sample of the first three records is shown in **Table 2.4**. Unlike the prior database tables, this one contains three fields of interest: `compound`, `predicate`, and `object`. These three fields describe the three components of a predication stored in

Table 2.4.: Sample data from 'SemanticVExtractor' database table.

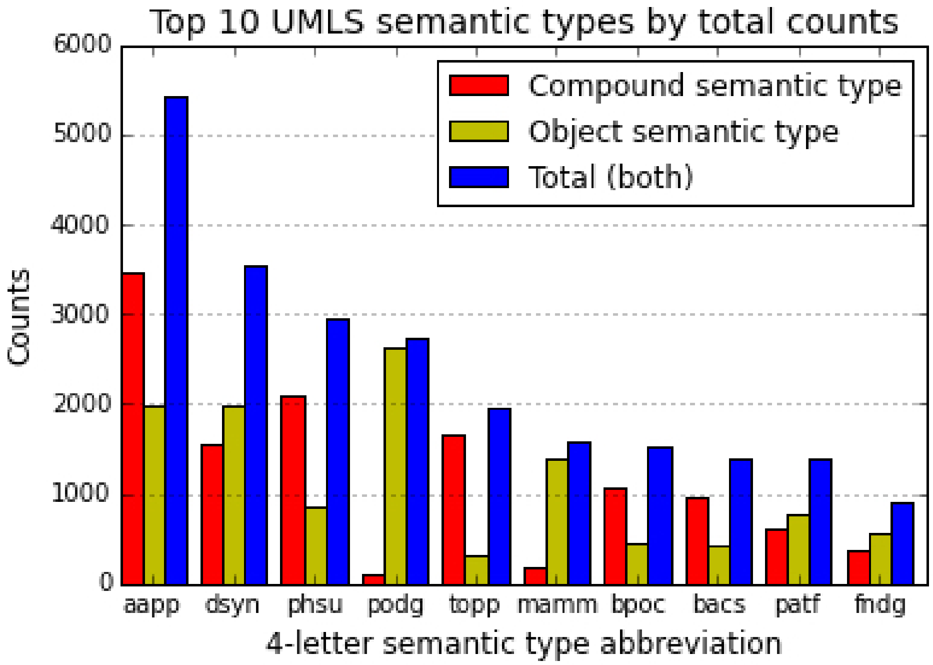
ID	compound	predicate	object	pmid
1	bombesin	isa	tetradecapeptide	11996
2	bombesin	augments	gastric	11996
3	caerulein	affects	acidification	11996
...	...	...	...	
n	*[compound n]*	*[predicate n]*	*[object n]*	*[pmid n]*

the SemMedDB database—a subject, a predicate, and an object. A predication describes a relationship between two entities (the subject and the object), and its predicate defines the type of relationship. The order ⟨*subject*⟩ | ⟨*predicate*⟩ | ⟨*object*⟩ has the advantage of being similar to the structure of an English language sentence, so the semantic concept underlying the predication can be easily read by a human. For example, if the predication is `caerulein | augments | pancreatic juice secretion`, it is easily understood as equivalent to the phrase, 'The (venom-derived) compound named caerulein augments the secretion of pancreatic juice.' In this context, the subject of the predication is always a chemical compound, so the `subject` field of SemanticVExtractor output was renamed to `compound` upon loading into the knowledge base. However, the venom component being referred to is not always the subject—it could also be the object of the predication. For instance, one of the predications in this table could be `compound 'X' | inhibited_by | bombesin`'. This predication describes the effect of bombesin on compound *X*, yet bombesin is the object of the predication. In this table, `compounds` and `objects` are always either UMLS terms or GO terms, and 'predicates' are the predicates that are contained within the SemMedDB database (specifically contained in the `PREDICATION` table of the database). A parallel bar chart of the 10 most frequent semantic types in Semantic VExtractor is shown in **Figure 2.3**. To improve the utility of VenomKB beyond that of a purely static knowledge resource, individual web pages describing the data records contain links to their cited PubMed articles, as well as links to search queries for compounds and other terms on a number of external databases/ontologies. Since there is no structured terminology or naming scheme for venoms and/or venom derived compounds, we cannot guarantee that all records in VenomKB will return useful search results—this is something that we intend to improve upon in the future by creating a hierarchical terminology of venoms that can be used to standardize the contents of VenomKB, and generate cross-mappings to other knowledge resources regardless of synonym variation.

2.1.4. Technical validation of VenomKB v1

The manually reviewed and curated list of putative venom therapies was considered the 'gold standard' against which the two automated methods of knowledge extraction were validated. We validated the ability of the two automated methods to identify venom compounds and their purported effects on the human body. It was assumed that the precision of the two automated methods would be low, since there is no UMLS semantic type or other unique identifier with which venom compounds are annotated in a consistent manner in the scientific literature. As a result, many of the identified compounds are not venoms at all, but belong to the same UMLS semantic types as venoms and venom components. However, we designed the two algorithms to be highly sensitive. In essence, we expected to see a high occurrence of false positives but a substantially lower occurrence of false negatives.

In order to determine percent recall of the two algorithms, we selected 100 records at random from the table of manually reviewed venom therapies. For each of those selected



Key to abbreviations:
aapp: Amino Acid, Peptide, or Protein
dsyn: Disease or Syndrome
phsu: Pharmacologic Substance
podg: Patient or Disabled Group
topp: Therapeutic or Preventive Procedure
mamm: Mammal
bpoc: Body Part, Organ, or Organ Component
bacs: Biologically Active Substance
patf: Pathologic Function
fndg: Finding

Figure 2.3.: The top 10 most frequent UMLS semantic types represented in the SemanticVExtractor data output, plotted by total counts.

data records, we then manually recorded whether the same venom compound and effect were identified by each of the two algorithms for the MEDLINE article associated with the respective PMID. For this measurement, VExtractor exhibited a 76% recall with respect to the gold standard, and SemanticVExtractor exhibited a 67% recall. Additionally, we recorded whether the ⟨*venom*⟩ | ⟨*effect*⟩ pair was found in any record, regardless of PMID. For this second measurement (where the 'PMID' field was disregarded), VExtractor had a recall of 89% (a change of +13%) and SemanticVExtractor had a recall of 84% (a change of +17%). These data support the conclusion that the two algorithms have a relatively high degree of sensitivity for correctly extracting venoms and their purported effects, and the false negatives (⟨*venom*⟩ | ⟨*effect*⟩ pairs not identified by one of the two algorithms) are substantially offset by the ability of the algorithms to identify equivalent ⟨*venom*⟩ | ⟨*effect*⟩ pairs elsewhere in the scientific literature. We calculated the specificity of each of the two algorithms by selecting 100 random records and determining whether those records describe a venom or a venom compound, and also whether they describe a physiologic target or effect of that venom compound. Prior to pruning obvious false positives from each of the two database tables, VExtractor demonstrated a precision rate of 66%, and SemanticVExtractor demonstrated a precision rate of 52%. After pruning false positives, we resampled the two database tables and recomputed precision. Each of the two values improved substantially: VExtractor demonstrated a new precision rate of 82% (a change of +16%), and Semantic VExtractor demonstrated a precision rate of 80% (a change of +28%), empirically demonstrating the value of manually filtering 'bad values'. These values (both recall and precision) are shown in **Table 2.5**, and the specific data records used to conduct the validation are available on FigShare (see §2.1.6; Data Citation 4). Although these rates for precision are relatively high

Table 2.5.: Percent recall rates of the 'VExtractor' and 'SemanticVExtractor' algorithms as compared to the gold standard (manually reviewed venoms), using 100 randomly selected records from the 'Manually Reviewed Venoms' table. Reported precision was computed after manually pruning obvious false positives.

Algorithm name	**Num. records**	**Recall (PMID-sensitive)**	**Recall (PMIDs disregarded)**	**Precision**
VExtractor	35,240	76%	89%	82%
SemanticVExtractor	7,208	66%	84%	80%

for a novel knowledge discovery pipeline, they do raise the question of how to minimize false positives in a maintainable fashion, rather than via manual review and culling of erroneous records within very large database tables. As mentioned below (in §2.1.5), VenomKB allows for users to 'flag' individual records for removal. This method of 'crowd sourcing' the removal of erroneous records will continue to improve in its robustness as VenomKB gains content and new users.

False negatives are another important concept to consider. Our method for identifying relevant PubMed articles involved searching for the MeSH term `Venoms/therapeutic use`, but since MeSH terms are manually curated annotations, there is no way to ensure full coverage of relevant articles. Furthermore, a lack of structured terminological resources for studying venoms and venom components makes more complex methods of knowledge retrieval (e.g., using alternative machine learning techniques that incorporate semantic knowledge of venom compounds) nearly impossible. To this end, we are planning a follow-up study that involves the creation of an ontology for venoms and their contained compounds, as well as synthetic derivatives that are already used therapeutically. After creating the ontology, we should be able to devise novel methods for identifying false negatives—those records erroneously omitted from the database due to a lack of complete MeSH term annotation.

2.1.5. VenomKB v1 usage notes

Any internet enabled device using a modern web browser should be able to access the knowledge base and download data from both the knowledge base itself (at `http://www.venomkb.org`) and from the FigShare repository (See Data Citations in §2.1.6). No user account is necessary to access or download the data records, but a number of community editing and contribution features do require users to create a private profile. Users have the ability to add or edit records on the manually-curated portion of the knowledge base. Deletion privileges are not publicly available, in order to prevent abuse. However, if a user feels that a particular record was included erroneously, there is a button on the data record's page that allows the user to 'flag' the record for review by site administrators. Once flagged, administrators are notified, after which they decide whether to remove the item or not. Users can see on the index page whether individual items have been flagged or not. Furthermore, users may contribute to a 'comment' thread on each data record, given that they have logged in to an account. Comment threads are visible on the pages for each individual database record. Major changes to the knowledge base are announced on the knowledge base website when they occur. As mentioned previously, data records may be selected and downloaded in one of three software formats: CSV, XML, and JSON. Although these may be manipulated and analyzed by most modern programming languages and data analysis software packages, we performed technical validation of the data sets using the standard libraries of the Python and Ruby programming languages. Intermediary data files (prior to loading into a relational database) were structured as to make them 'self documenting' (i.e., key-value pairs include descriptive key labels). A GitHub repository with all of the scripts used to analyze and process the data is linked to in **Appendix D**. Within the knowledge base, numerical identifiers for individual records were assigned arbitrarily based upon the order in which they were added to the database.

As mentioned previously, VenomKB will change and grow as new records are added. In particular, we plan to expand the 'manually curated venoms' database by identifying additional relevant MeSH terms that may also refer to therapeutic uses of venoms (aside from `Venoms/therapeutic use`). Furthermore, we plan to closely monitor new studies regarding novel venom therapies, adding them to the knowledge base as we come across them.

Since VenomKB is intended to grow into a collaborative, public resource on computational analysis and prediction of putative venom therapies, we encourage suggestions and comments regarding new additions and revisions. Up-to-date contact information can be found from the homepage of the knowledge base website, or alternatively, readers can contact the corresponding author for this study as listed below.

A final note to users regards data records that appear to be irrelevant at first glance, yet actually do describe a property of a venom compound being used for therapeutic purposes. For example, consider entries in the VExtractor database for PMID 22098810. The database contains 4 entries for this PMID, all referring to a venom compound named `hypoglycemic agent`, which treats `obesity`. Upon inspecting the journal article referenced by this PMID, it can be seen that the hypoglycemic agent in question is actually the venom compound `exenatide`, which does treat both type-2 diabetes mellitus and obesity [153]. As mentioned previously, we plan to build a structured terminology for venom compounds that can be used to resolve relatively uninformative descriptors (such as `hypoglycemic agent`) into their actual specific venom compound names, but since such a resource currently does not exist, we suggest that users follow links to the PubMed pages to validate the compound(s) themselves.

2.1.6. VenomKB v1 Data Citations

The following citations point to persistent copies of the data as referred to in the preceding text:

1. Romano, J.D., & Tatonetti, N.P. *Figshare* [`http://dx.doi.org/10.6084/m9.figshare.1287000`] (2015).
2. Romano, J.D., & Tatonetti, N.P. *Figshare* [`http://dx.doi.org/10.6084/m9.figshare.1286999`] (2015).
3. Romano, J.D., & Tatonetti, N.P. *Figshare* [`http://dx.doi.org/10.6084/m9.figshare.1287001`] (2015).
4. Romano, J.D., & Tatonetti, N.P. *Figshare* [`http://dx.doi.org/10.6084/m9.figshare.1289881`] (2015).

2.2. Venom Ontology

2.2.1. Introduction

Perhaps the most fundamental issue standing in the way of modern translational research for venom-based drug discovery is the almost complete lack of an informatics infrastructure uniting our existing knowledge on venoms. In this study, we present a novel ontology of venoms and related concepts that addresses this problem systematically. Biomedical ontologies allow for consistent and unambiguous naming of entities (in this cases, venoms, venom components, and the species from which they are sourced) and how they are interconnected. We also present a number of initial investigations regarding venom biodiversity across the tree of life, and explore how they can inform the discovery and refinement of novel therapeutic uses for venom compounds.

2.2.2. Methods

Building the Venom Ontology

We used Protégé (ver. 4.2) [186] to create the class structure of the Venom Ontology using domain knowledge: By our definition, every venomous species has exactly one venom, and every venom has one or more molecular components that can be classified by the class of molecule they are (e.g., peptide, carbohydrate, inorganic cofactor). Recent reports suggest that *Conus geographicus* modifies its venom based on whether it is used defensively or offensively [66], but for the purposes of this ontology they can be grouped together as a single venom. If a venom component is a peptide, it has a canonical amino acid sequence. Each of the entities may have one or more other pieces of metadata, including links to other ontologies and structured terminologies. After defining the class structure of the ontology, we populated the ontology with individuals (specific instances of the ontology's classes) sourced from UniProtKB/Swiss-Prot's Tox-Prot database [113]. This database is a manually curated list of venom peptides containing numerous annotation tags including species of origin, amino acid sequences, full taxonomic lineage, and automated cross-mappings to other online resources. However, the structure of Tox-Prot does not support semantic reasoning. Due to the large number of individual records in the Tox-Prot database (6,092 at the time of creating the ontology), we added the contained information programmatically by first exporting the ontology from Protégé to an RDF-formatted XML file [17], and then using Apache's Jena framework [165] to parse the venom records and insert relevant data into the appropriate spot within the ontology's class hierarchy.

Exploratory analysis of venom ontology data records

To demonstrate some potential applications of Venom Ontology, we performed three exploratory analyses of its contained data. The first of these involved assessing the similarity of amino acid sequences for venom peptides produced by species of the same genera. To accomplish this, we grouped species (stored as "Organisms" in the ontology) by genus, along with their derived peptide compounds. We then selected 2 genera that are well represented in the data set, and built "sequence similarity networks" for each of them. In selecting these genera, we looked for ones that are prolific enough within the ontology to generate informative (non-trivial) networks, yet not so prolific as to be unwieldy in terms of visualization or computation. In practice, we looked for two genera with approximately 20 species in the ontology. For each genus selected, we used BLASTp [37] to align all pairs of peptides within the genus. We constructed the networks using peptide sequences as nodes, and the alignments between them as edges. We transformed the BLAST scores (which represent the percent coverage of the alignments; denoted S)for alignments using the following equation:

$$S' = \frac{1}{e^S}$$

which allows us to define a "distance" between two peptide sequences (i.e., smaller values of S' indicate higher similarity), used as edge weights in the final networks. S' is a value in the interval $(0; 1]$, and is generally very small (e.g., $< 1 * 10^{-15}$). Finally, we filtered edges by setting a maximum expect value ("e-value"—a normalized p-value defining confidence that the alignment is non-random) threshold of $1 * 10^{-50}$. Alignments that fell below this maximum cutoff almost certainly signify evolutionarily related sequences, and are therefore informative for the purposes of constructing these networks. For visualization purposes, we rendered the networks in Cytoscape [224] using the prefuse force-directed layout [96], and colored nodes (individual peptides) by the species from which those peptides were sourced.

Our second analysis was a basic exploration of the distribution of both species and individual peptides in the ontology across the tree of life. We defined common groupings of animals (cnidarians, molluscs, insects, arachnids, fish, amphibians, reptiles, birds, and mammals) that may contain venomous species. From these large classes, we used NCBI's Taxonomy database [72] to determine the highest-level taxa common to all members of those groups (grouping multiple taxa for paraphyletic groups, such as "fish"). For each of these taxa, we searched for their frequency of occurrence in the set of all species present in the database. We also enumerated the number of total sequences in the database for the groups listed.

The third and final analysis consisted of observing the complexity of venoms within the ontology. In this context, we simplistically define complexity as the number of distinct peptide components within the venom (e.g., a venom containing 20 peptide components is more complex than a venom containing only 10). We investigated the distributions of venom complexity for each of the taxonomic groups mentioned in the previous paragraph, making note of features such as mean number of peptide components per venom, standard deviation, and skewness (i.e., lack of symmetry, computed as the estimated third standardised moment $E\left[\underline{\mathrm{x}}^3\right]$).It should be noted that the results of these analyses are subject to systematic biases depending on how well the data in Tox-Prot is representative of the totality of venoms that exist in nature (refer to §2.2.4 for further discussion).

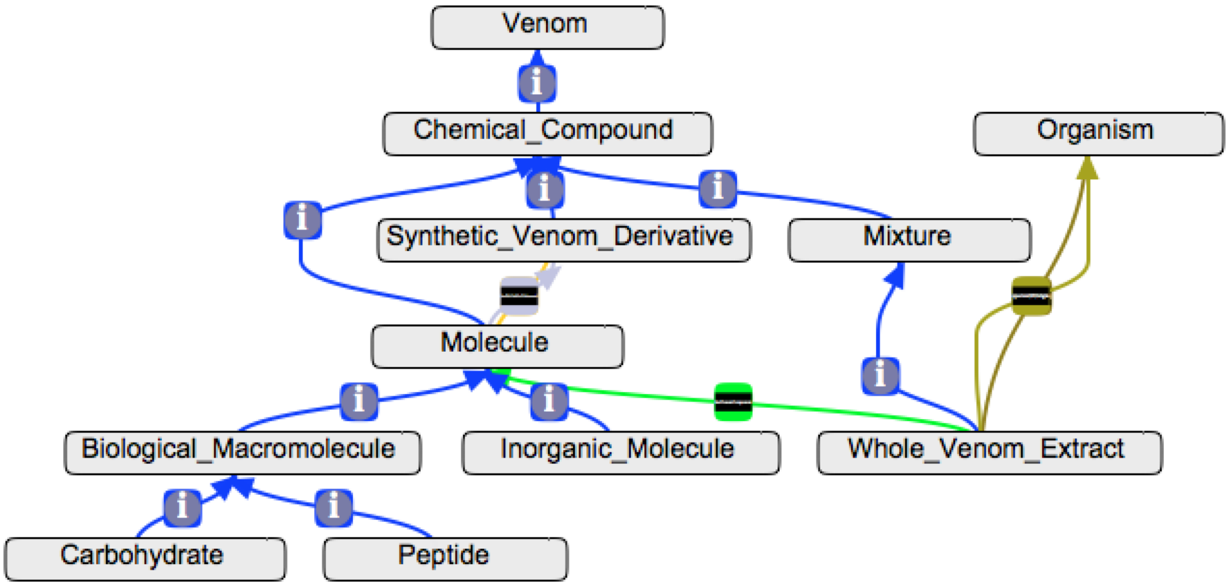


Figure 2.4.: A schematic diagram of classes and class relationships in the Venom Ontology. Blue arrows denote `is_a` relationships, while other colors denote object property relationships. Diagram automatically generated by the Protégé plugin "OBO Graph View".

2.2.3. Results

All code and data files used in this study are available for public use on GitHub at (`http://github.com/JDRomano2/venom_ontology_code`). The ontology is available online, hosted both on BioPortal (`http://bioportal.bioontology.org/ontologies/CU-VO`) and on the project's homepage, at `http://venomkb.tatonettilab.org/ontology`. A visualization of the ontology's class hierarchy and object property associations is shown in **Figure 2.4**.

Venom ontology

Venom Ontology presently contains 614 known venomous species, and 6,092 curated peptides, each of which has a known amino acid sequence. There are correspondingly 614 "whole venom extract" entities, arising from the following axiom:

$$\text{Organism} \supseteq \geq 1 \text{has Venom.WholeVenomExtract} \cap \leq 1 \text{has Venom.WholeVenomExtract}$$

which states that every organism has exactly one whole venom extract. Due to our data source being peptide-centric, each whole venom extract (and, correspondingly, each organism) currently included in the ontology has at least one peptide, although this is not defined as necessary (i.e., the ontology allows for whole venom extracts to contain zero or more peptides). We added a small number of synthetic venom compounds (all clinically approved drugs) to the ontology by manually entering them as individuals for the "Synthetic_Venom_Derivative" class. This is a tractable approach presently, but as venom-derived therapeutic agents continue to be discovered and are coerced into a structured format, an automated means for adding them will become necessary—this point is elaborated on below, in §2.2.4. Venom Ontology was validated using the FaCT++ reasoning engine [248].

Analysis of the ontology's contained data

Our analysis of venom peptide sequence similarity for a number of well-represented genera highlights some noteworthy features of venoms that have significant implications for drug discovery. In **Figure 2.5**, we show two sequence similarity networks—one for genus *Loxosceles* (widow spiders) and one for *Bungarus* (kraits—a genus of venomous snakes)—yet our methods could be applied to any other taxonomic group that is present in the ontology. Since we only kept alignments with strong statistical support (low e-value—see §2.2.2 for details), the graphs are not fully connected. Small connected components (e.g., the "islands" seen

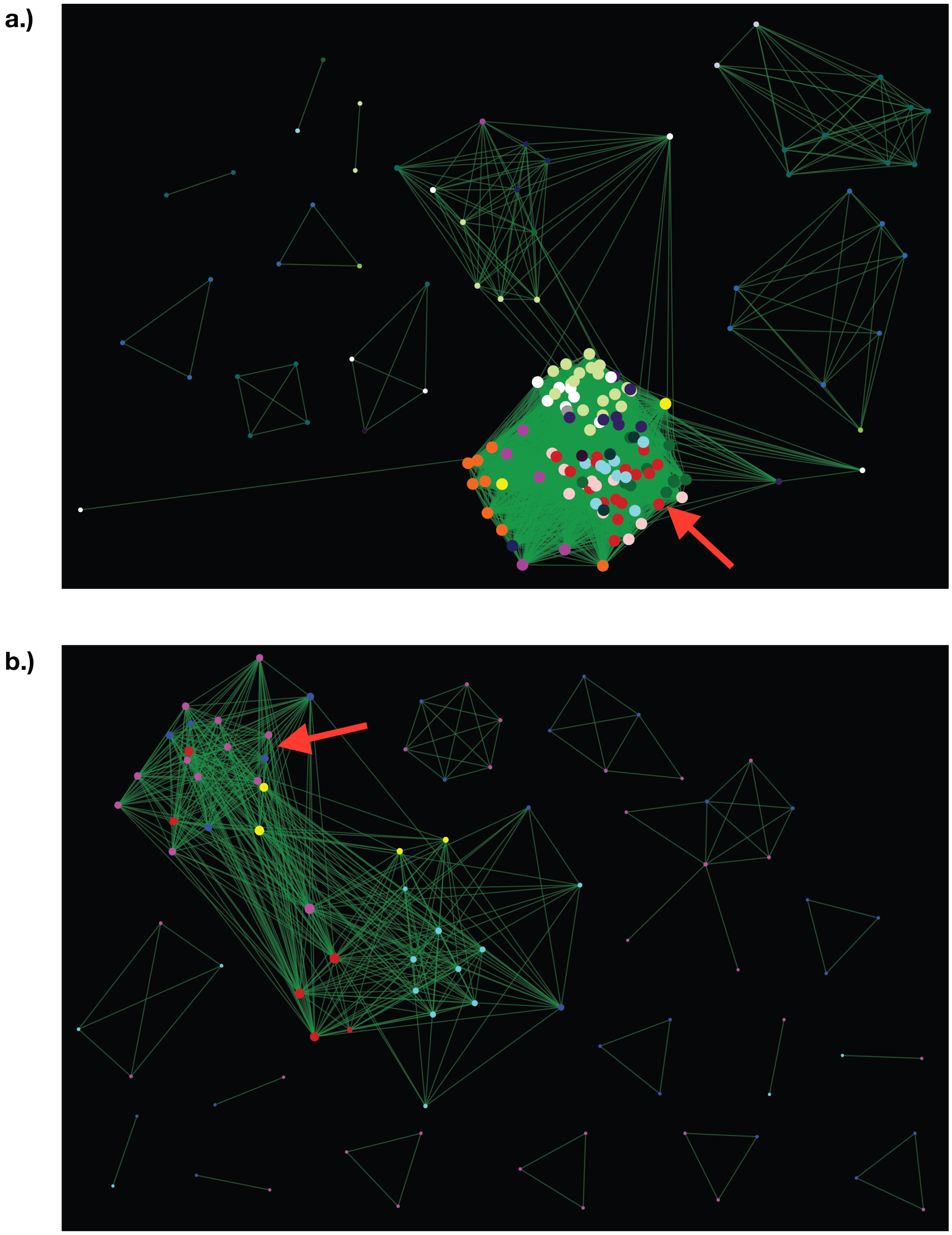


Figure 2.5.: Two sequence similarity networks for venom peptides within the same genus. a.) shows peptides from species in genus Loxosceles, and b.) shows peptides from species in genus Bungarus. Relative node size is based on the degree of the node, and length of the edges is based on the inverse BLASTp score (see eq. (1)). Nodes of the same color are peptides from the same species of animal. Red arrows indicate "clusters" with high species diversity (i.e., similar peptides found in a number of closely related species).

around the periphery of the networks) as well as clusters within larger connected components can be interpreted as groups of peptides that are likely to be closely related on a structural level. Although we originally expected sequences from a given species to segregate together, there are clusters in each of the networks that contain a diverse mixture of sequences from numerous species (denoted in the **Figure 2.5** by red arrows). The smaller connected components tend to be more homogeneous in terms of their species composition (e.g., they have higher cluster purity). Subjectively, it is also noteworthy that the networks do not display the properties of "scale-free" networks (characterized primarily by few nodes of very high degree, and many nodes of very low degree), which are arguably the most prevalent family of networks that arise from biological phenomena [12]. While speculation as to why this occurs is beyond the scope of this exploratory analysis, it would be an interesting topic to pursue in a follow-up study.

The distribution of species and sequences by higher taxonomic groupings is shown in **Table 2.6**. Both "fish" and "reptiles" are common names that consist of multiple clades (i.e., they are paraphyletic). It should be noted that 5 species, containing a total of 1,348 sequences, are not classified within any of these groups. While this only makes up 0.81% of the total number of species, it contains 22.13% of the total number of sequences found in the ontology. This seems to be the result of numerous sequences that have poorly formed or absent "taxonomic lineage" annotations in Tox-Prot (meaning that some of the 'orphaned'/unclassified sequences likely come from already classified species that are included in the larger taxonomic groups). After looking at properties of venoms exposed by

[1] Skewness is the estimated third standardized moment of the empirical distribution, $\mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right]$. Higher skewness indicates greater lack of symmetry about the mean.

Table 2.6.: Distribution of species and sequences in the Venom Ontology across common taxonomic groups. Some groups with no species or sequences are included for completeness.

Common Name	Taxonomic group	# species in ontology	% total species	# sequences in ontology	% total sequences
Cnidarians	Cnidaria	0	0.00	0	0.00
Molluscs	Mollusca	97	15.80	1089	17.88
Insects	Insecta	79	12.87	245	4.02
Arachnids	Arachnida	183	29.80	1089	17.88
Fish	Actinopterygii	4	0.65	12	0.20
	Coelacanthimorpha	0	0.00	0	0.00
	Chondrichthyes	1	0.16	2	0.03
	Cyclostomata	0	0.00	0	0.00
	Dipnoi	0	0.00	0	0.00
Amphibians	Amphibia	2	0.33	3	0.05
Reptiles	Archelosauria	0	0.00	0	0.00
	Squamata	242	39.41	2298	37.72
Birds	Aves	0	0.00	0	0.00
Mammals	Mammalia	1	0.16	6	0.10
Other/unclassified		0	0.00	0	0.00
Total		614		6092	

Table 2.7.: Distribution of venom complexity across the tree of life, by common taxonomic groups. A venom's complexity is defined as the number of known distinct peptide components it contains.

Common name	**Minimum**	**Median**	**Mean**	**Maximum**	**Skewness**[1]
Molluscs	1	4	11.230	118	3.638
Insects	1	2	3.101	15	2.211
Arachnids	1	4	13.020	293	6.576
Fish	1	2	2.800	6	1.517
Amphibians	1	1.5	1.500	2	n/a
Reptiles	1	4	9.496	64	2.271
Mammals	6	6	6.000	6	n/a
All species	1	4	9.922	293	7.987

Table 2.8.: Mann-Whitney *U* test results for all pairs of venom complexity distributions. A *p*-value of less than 0.05 signifies that two distributions are statistically different.

	Arachnids	Fish	Insects	Mammals	Molluscs	Reptiles
Amphibians	0.117	0.417	0.439	0.667	0.167	0.126
Arachnids		0.155	**1.85e−7**	0.842	0.909	0.725
Fish			0.732	0.366	0.216	0.170
Insects				0.194	**2.74e−5**	**2.20e−7**
Mammals					0.858	0.813
Molluscs						0.878

the ontology at the genus level, we investigated the distribution more generally across the tree of life. Distributions of venom complexity are shown in **Table 2.7**. In this portion of the data analysis, we only show the common taxonomic groups from **Table 2.6** that have at least 1 venom and 1 peptide. The final row of the table shows the distribution across all species present in the ontology. Additionally, **Figure 2.6** shows a graphical representation of these distributions, drawn as violin plots with a logarithmic scale on the vertical axis.

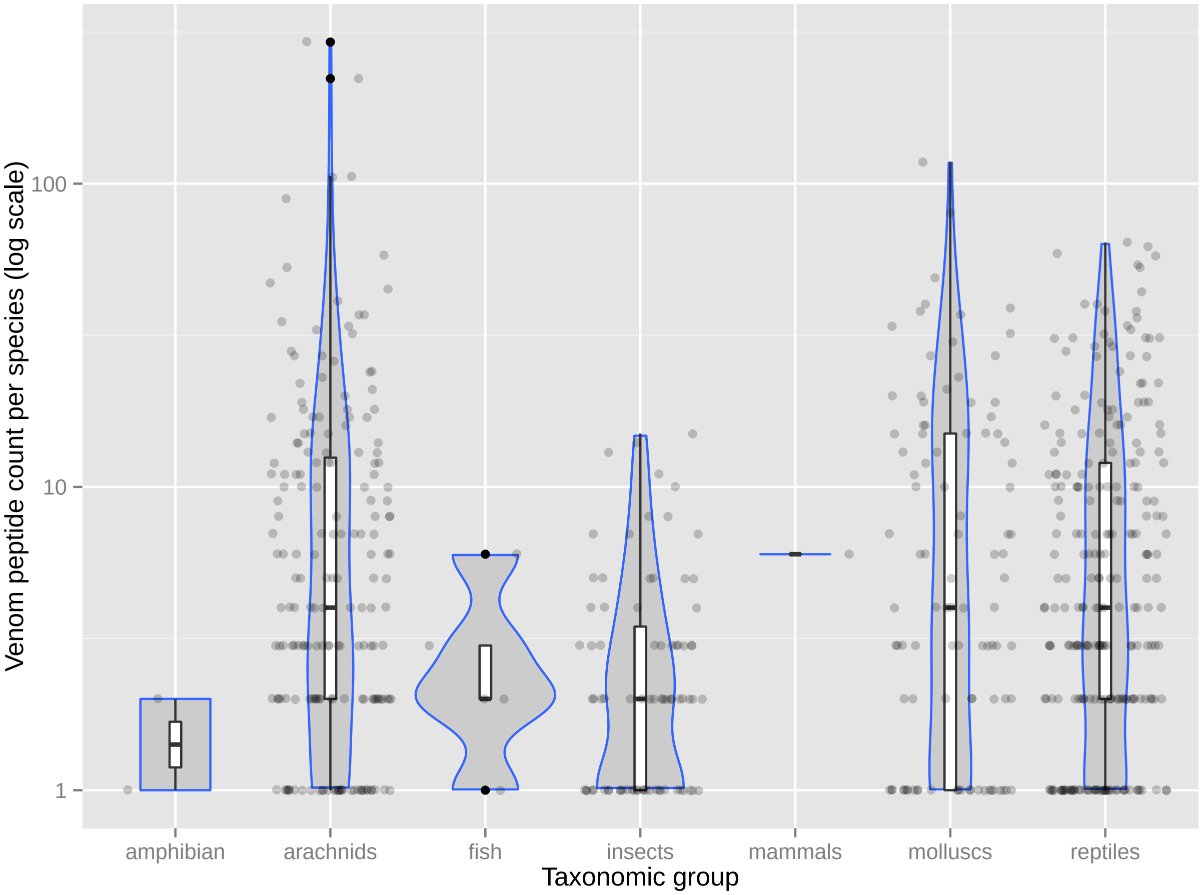


Figure 2.6.: Violin plots showing distributions of venom complexity in 7 common taxonomic groups. Numeric summary statistics are listed in **Table 2.7** for each of the groups shown. Complexity is measured as the number of venom peptides in Venom Ontology for a single species—the vertical axis is the complexity measure for a given species, and the widths of individual plots correspond to the density of the distribution at that complexity measure. Individual species are shown as transparent dots—they are spread horizontally ("jittered") to better visualize dense groups of data points.

2.2.4. Discussion

Some ontology classes possess no individuals, yet are still informative

The Venom Ontology contains several terminal classes that do not have any members ("individuals"), classes `Carbohydrate`, and `Inorganic_Molecule`. The rationales for their inclusion are threefold: (1) The ontology is meant to convey computable semantic knowledge of venoms, and with the current structure ontology reasoning software is able to understand that venoms may contain a number of different components, of which only some may be peptides. (2) Since future revisions to the ontology may incorporate new data sources, we hope to be able to populate these classes with informative instances in a future release. (3) We hope to be able to generate members for these classes using machine learning methods that don't require a curated dataset of venom components (such as "ontology learning from text") [263]. Another class—"Synthetic Venom Derivative"—seems to be specific enough to allow for manual population using domain knowledge of existing synthetic versions of venoms used as pharmaceuticals. However, existing synthetic venom derivatives are more numerous than it would initially seem. For example, a number of conantokins (a specific sub-class of conotoxins—sourced from snails in the genus *Conus*) have been modified and produced synthetically, yet none have received approval for clinical use [45, 202]. For this reason, a potential follow-up to this study would be a comprehensive survey of synthetic derivatives of venom peptides.

Grouping venom peptides by genus reveals clusters of similar venoms across species

As briefly alluded to in §2.2.3, the networks in **Figure 2.5** show clusters of venom peptides that contain members from a number of closely related species. This suggests a novel approach for discovering libraries of therapeutic venom-derived peptides with a similar therapeutic effect. During drug development, having a large number of drug candidates available improves the likelihood of finding a molecule that simultaneously has the greatest therapeutic effect while minimizing toxic effects (a notoriously challenging obstacle in repurposing venoms for clinical use). This proposed approach provides a data-driven framework for discovering venom-derived therapeutic agents, which is an improvement over traditional methods that are almost entirely based on serendipitous discovery or borrowed from ancient traditional medicine [141].

Non-reptile venomous species are underrepresented in existing data

Recent analyses of venom biodiversity reveal surprising patterns, including that the prevalence of venomous fish is far higher than in any other major taxonomic group, including reptiles [229]. **Table 2.7**, however, shows a strong bias towards venomous reptiles in available data (fish peptides make up only 0.23% of venom sequences in the Tox-Prot dataset, while reptilian peptides make up 37.72%). Other discrepancies are also apparent: for example, only one venomous mammal is included in the database: *Ornithorhynchus anatinus* (duck-billed platypus). While it is uncommon for mammals to be venomous, reviews on the subject have identified numerous others aside from *O. anatinus*, including multiple shrews, bats, and certain species of loris (taxonomic family Lorinae).

By knowing about these discrepancies, we can prioritize future venom research to include presently underrepresented categories of animals, which should in-turn increase the likelihood of discovering novel compounds that have diverse therapeutic effects.

Apparent complexity of venoms varies across the tree of life

Venoms usually consist of a complex mixture of organic and inorganic molecules, each of which has a particular effect. If we define "complexity" as the number of distinct peptide components in a venom, our results show that venom complexity is highly variable across the tree of life. In **Table 2.7** we list summary statistics for venom complexity distribution across 7 common taxonomic groupings. These data are additionally visualized in **Figure 2.6** as a violin plot. The plot, shown with number of peptides per venom on a logarithmic scale, highlights that there are many outliers in the dataset—species with extremely complex venoms compared to the mean of 9.922 peptides per venom. Furthermore, each of the taxonomic groups has its own unique distribution. Although the sizes of some groups in the ontology are too small to result in viable statistical inferences (e.g., mammals and amphibians), variable distributions of venom complexity suggest that complexity is regulated in some manner that is conserved by evolution—otherwise, all of the distributions would converge. In particular, insects seem to have venoms that are relatively simple compared to arachnids, molluscs, and reptiles. Interestingly, arachnids have the largest number of outlier species that have extremely complex venoms. Reptiles, by far the most well-represented group in the dataset, have notably fewer highly complex outliers than either molluscs or arachnids. As an example of a quantitative approach to comparing these distributions, **Table 2.8** shows the p-values of the Mann-Whitney U test applied pairwise to all of the distributions shown in **Figure 2.6**. These observations may be an artifact of data completeness (see §2.2.4), but if not, they can help to guide research towards more rich libraries of venoms that may include important therapeutic compounds.

Using venom ontology in conjunction with VenomKB to support drug discovery

In §2.1, we described VenomKB v1.0—a knowledge base cataloguing putative therapeutic uses of venoms and venom-derived compounds, constructed via manual literature review and automated knowledge discovery techniques applied to MEDLINE [205]. Linking these two separate data resources may optimize the process of computational drug discovery by implying a polyhierarchical structure on many of VenomKB's data records (specifically, ones that map to instances in Venom Ontology). For example, if a record in VenomKB describes the therapeutic effect of a compound produced by species X, we may be able to find highly similar (and possibly more efficacious) molecules by using Venom Ontology to identify venom peptides from species that are in the same genus as species X. In the future, we intend to add a component to the ontology that resolves venom names with their synonyms, which could allow us to identify venoms with multiple therapeutic effects, as well as increase confidence in therapies when multiple studies corroborate the same effect. We plan to fully integrate these two resources, so that VenomKB can be browsed by navigating the hierarchical structure of Venom Ontology, and vice versa.

Limitations—structured data on venoms are largely incomplete

It is important to remember that these studies necessarily omit data on venoms from many clades of venomous animals. Since (a.) venom data are sparse even for most known venomous species, and (b.) we only have discovered a small handful of the vast number of venomous species believed to exist (and have actually studied even a smaller number), we treat the Tox-Prot dataset as a “best approximation” of venom diversity based on available data. This obviously introduces various sources of systematic bias into the inferences that are made from the ontology’s contained data.

In §2.2.2, we mention this limitation in regards to our definition of venom complexity (the relative number of peptide components contained within a venom). We analyze our data under the assumption that the Tox-Prot data set does not prioritize certain species for “completeness”—in other words, that the ratio of the actual number of peptides to the number that are in the data set remains consistent for all species. However, this may not be the case. The available data for some species may be substantially more complete than for others. Also, it may be more challenging to run proteomic analyses on some species than others. Each of these factors would affect the consistency of completeness across the dataset. A future goal that could help eliminate these potential sources of bias would be only to populate the ontology with complete proteomic surveys of species’ venoms.

We intend for the Venom Ontology to be one of the first major steps towards systematically and consistently coercing newly discovered venoms and venom components into a standardized format. The ontology’s structure suggests numerous ways to define a consistent vocabulary for these semantic concepts.

2.3. VenomKB v2

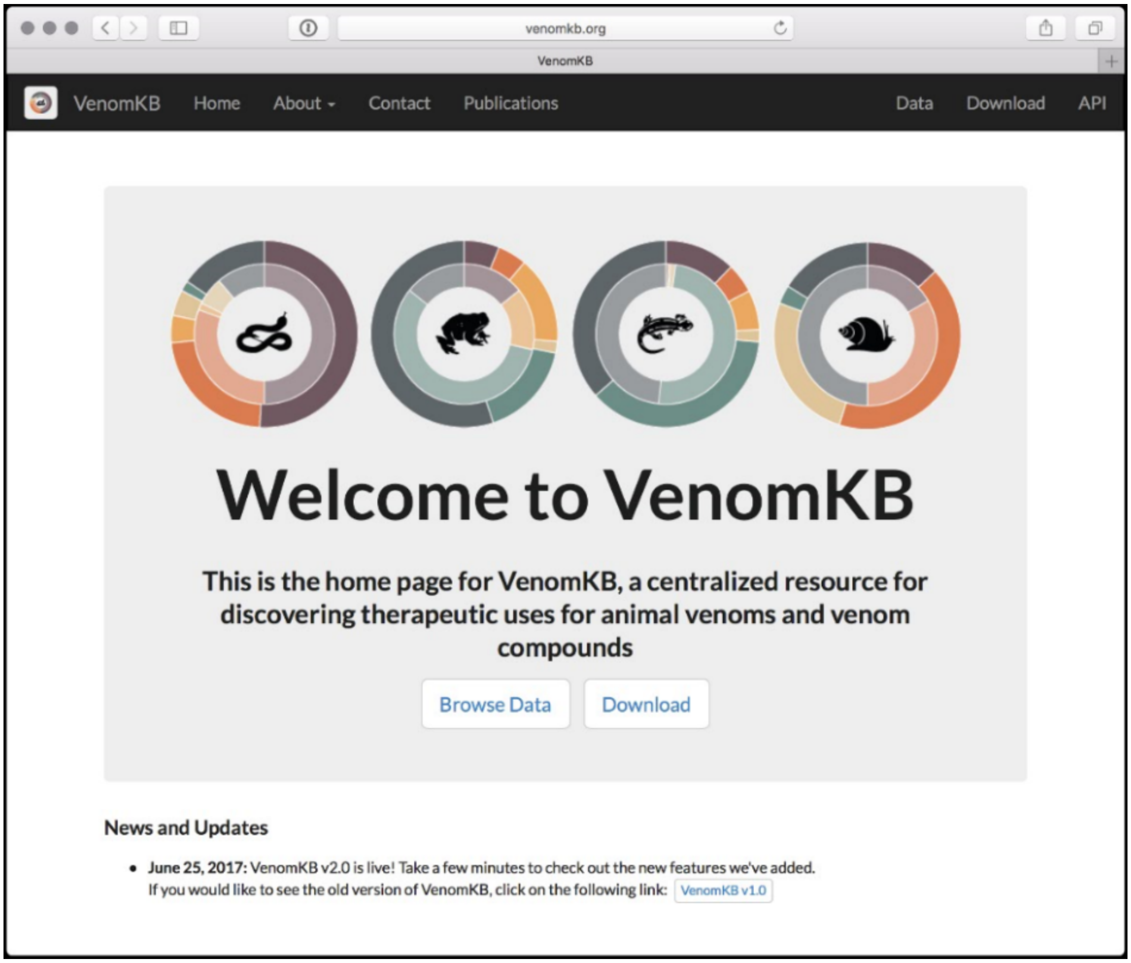
We then completely rewrote VenomKB to take advantage of the semantic structure provided by the Venom Ontology: In its revised form, VenomKB v2.0 is a resource for aggregating and representing venom knowledge including molecular characteristics, biodiversity data, manually- and automatically-identified literature data, and a standardized ontological representation for these different data types. VenomKB v2.0 is a complete rewrite of a previous toxinology resource aimed specifically at literature data [205], the contents of which are included in v2.0 in a more controlled and robust format. VenomKB is built with a modern and intuitive interface along with a REST API to make all data elements programmatically available. This knowledge base is the most complete public resource for computational toxinology research to-date, and it stands to become a major resource for toxinologists, informaticians, molecular biologists, and educators interested in venoms and/or their components.

2.3.1. Results

VenomKB can be accessed online at `http://venomkb.org/`. The original version of the knowledge base is still available for use, and can be accessed via a link on the home page of the URL above.

Size and structure of VenomKB

VenomKB currently catalogues 6,236 venom proteins from 632 venomous species of animals. VenomKB also contains 5 genomes from venomous animals, which—at the time of writing—is the entirety of publicly available venomous animal genomes known to the authors. 5 FDA-approved venom-derived drugs are included, as well as the targets that those drugs (and the venom peptides from which they are derived) are known to act upon. The major data types in the knowledge base are summarized in **Table 2.9**. **Figure 2.9** shows counts of the various data types contained in VenomKB.



Welcome to VenomKB

This is the home page for VenomKB, a centralized resource for discovering therapeutic uses for animal venoms and venom compounds

News and Updates

- **June 25, 2017:** VenomKB v2.0 is live! Take a few minutes to check out the new features we've added.
  If you would like to see the old version of VenomKB, click on the following link: VenomKB v1.0

Figure 2.7.: Image of the home page for VenomKB (v2.0). Users can access data and informational pages via the navigation bar or in the main body of the website. A "News and Updates" section provides useful information and changes made to the website.

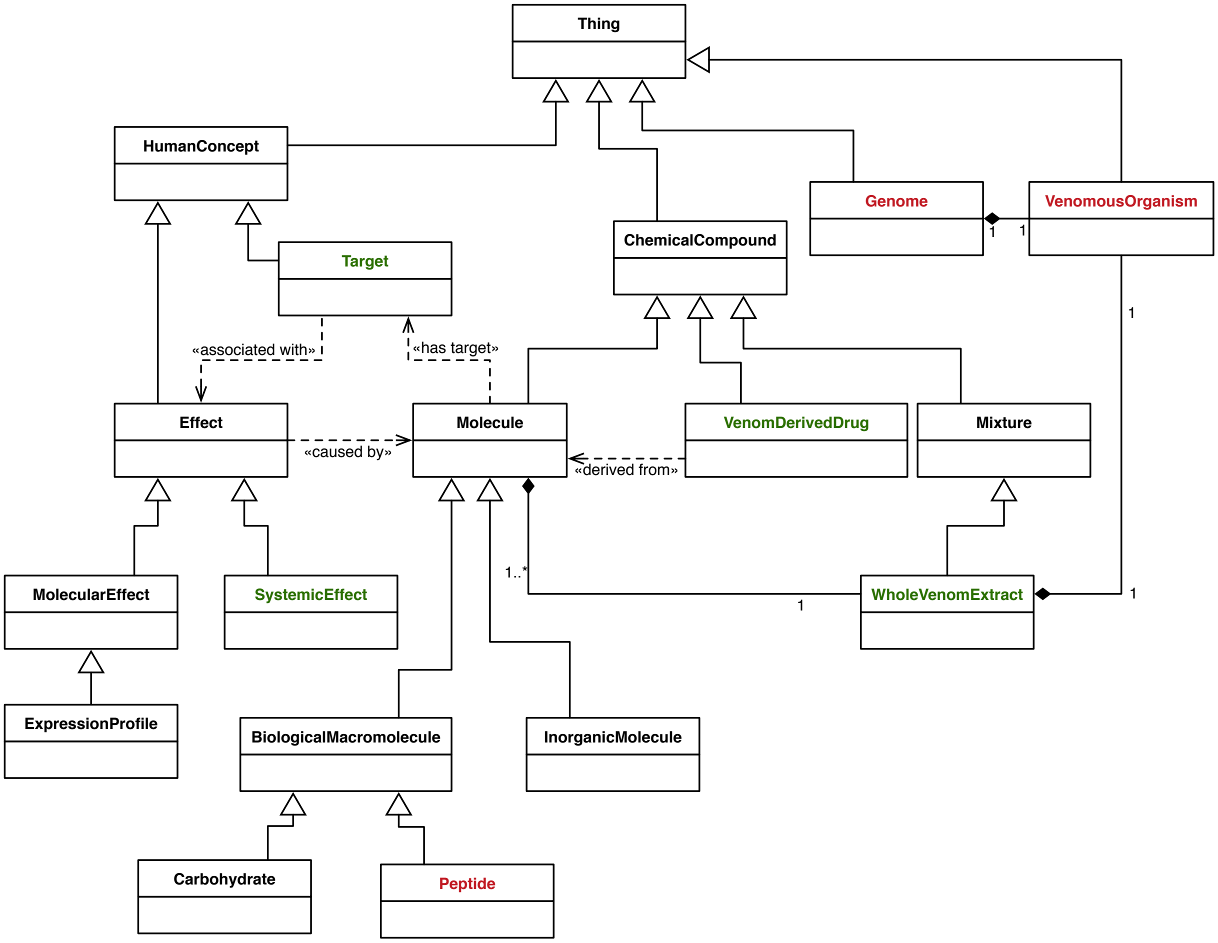


Figure 2.8.: UML class diagram showing the class hierarchy of Venom Ontology when used in VenomKB v2.0. Note the addition of several new ontology classes to accomodate new features not supported by the original version of the Venom Ontology, including `Effect` (and its descendents) and `Genome`. Red class names indicate classes that have dedicated data pages in the VenomKB v2.0 web application, and green class names indicate classes that are rendered as subcomponents on data pages for other classes.

Table 2.9.: VenomKB size and data types

Data type	Number of Records	VenomKB ID Prefix
Proteins	6,236	P
Species	632	S
Genomes	5	G
Disease/condition annotations	1,065	E
Approved venom-derived drugs	5	D
Literature predications	14,710	–
Gene Ontology annotations	18,677	–

Each of the previously described data types is structured according to the Venom Ontology [206], which provides a formal description of the different types of data related to venoms, along with the types of relationships that exist between them. Every data record in VenomKB is assigned a unique, permanent identifier that consists of one alphabetical character followed by seven digits. The first character indicates the data type (see **Table 2.9**), and the seven digits are randomly assigned.

We sourced all non-inferred data in VenomKB from other publicly available resources. A large number of the protein data were adapted from UniProtKB/Swiss-Prot's Tox-Prot annotation system [113], which is a major effort to identify and manually curate animal toxin peptides (including venom components) in UniProtKB. The number of proteins (6,236) currently in VenomKB is equal to the number of venom components in Tox-Prot at the time of constructing the database.

VenomKB also contains 39,179 literature annotations that describe a venom or a venom component treating a disease or health condition, which we transferred from VenomKB v1.0. Of these, 275 were manually curated, and 33,284 are normalized semantic predications extracted from the Semantic MEDLINE database using a knowledge discovery approach, which is described in a previous study [205]. We automatically mapped 14,710 of these predications to both species and individual proteins using ontological inference; these predications are shown in both the Species and Protein data pages, as well as the raw JSON representations of these data types.

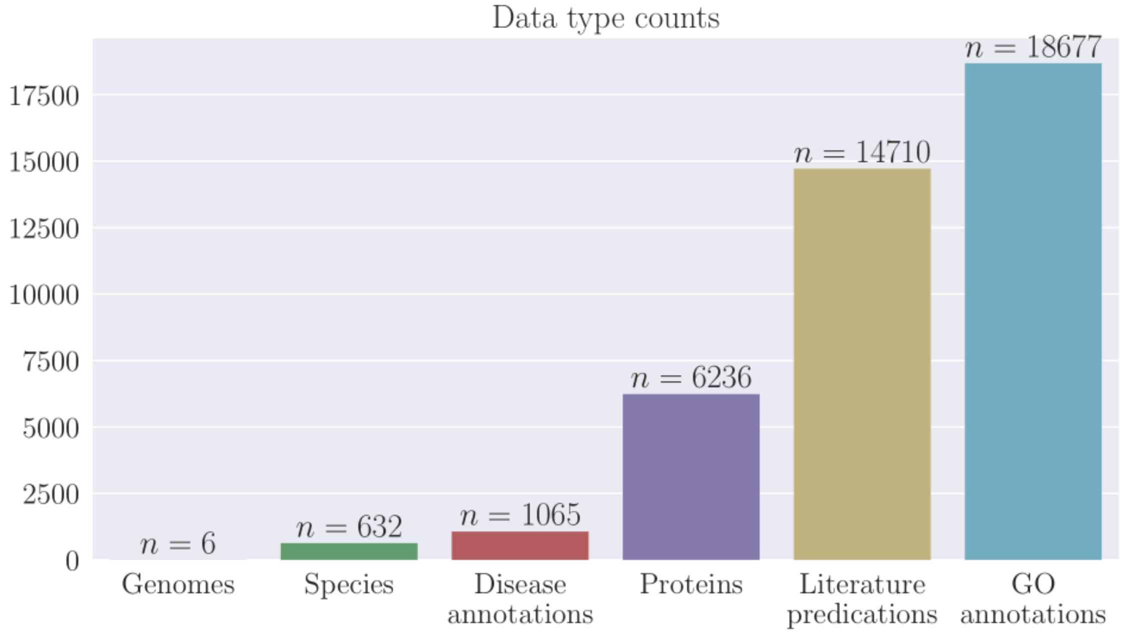


Figure 2.9.: Barplot of counts of data types in VenomKB v2.0. Genomes, Species, and Proteins are 'primary' data types represented as instances of Venom Ontology classes; Disease annotations, Literature predications, and GO annotations are 'secondary' data types that are represented as properties of primary data types.

Web application description

The home page for VenomKB is shown in **Figure 2.7**. From the home page, users can access most components of the web application, as well as a link to the VenomKB v1.0 application, for backwards compatibility. The main interface for exploring data is located at `http://venomkb.org/data`, or from links on the home page. The interface is shown in **Figure 2.12**, for reference. Users can filter data records in several ways, including by name, data type (e.g., proteins, species, or genomes), annotation score (1 to 5 stars, explained below), or by disease/condition annotations. Data types not included in this interface (such as literature predications and other annotations) are embedded within the structure of their corresponding documents. The search interface allows sorting by column. When users find a data record of interest, they can view it by clicking on its corresponding VenomKB ID (VKBID), or by navigating to '`http://venomkb.org/{VKBID}`'. An image of a protein detail page is shown in **Figure 2.13**. The detail page for individual data records presents information that is not available in the data search interface (e.g., for proteins, this includes

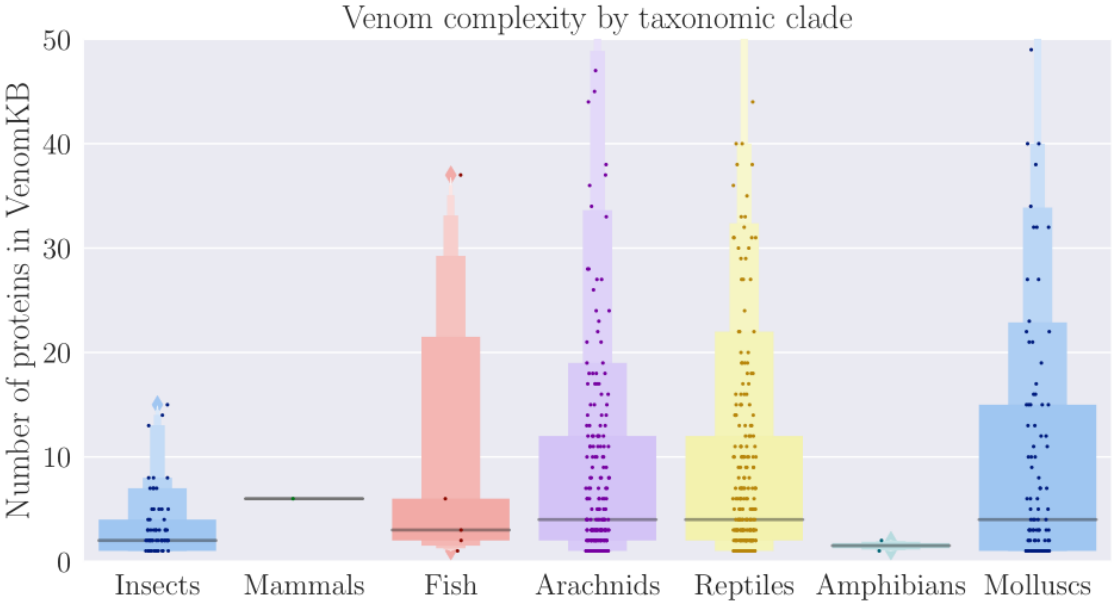
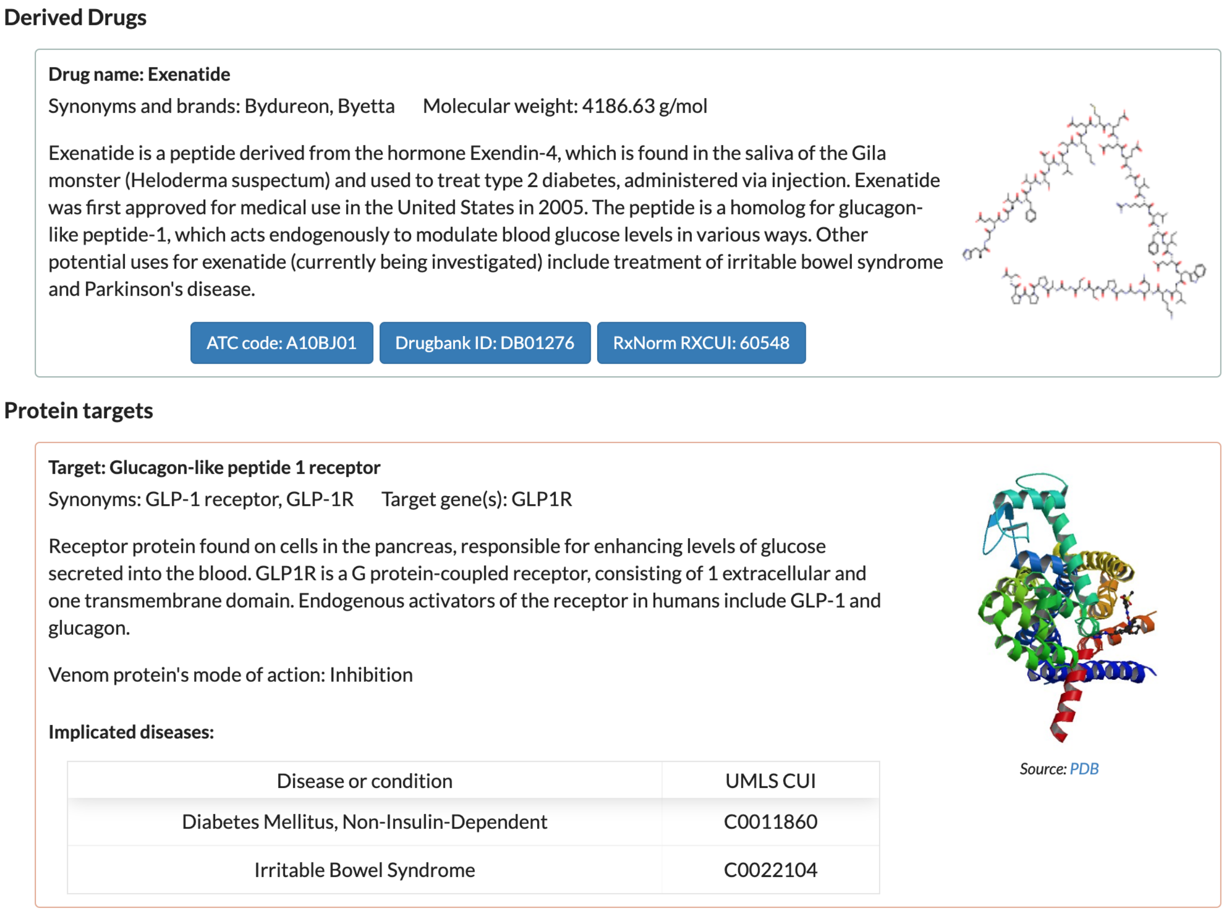


Figure 2.10.: Venom complexity by major taxonomic groups. 'Complexity' is defined as the number of proteins present in VenomKB for a specific venomous species. The relatively low complexity of insect venoms compared to arachnids, reptiles, and molluscs could be informative for the purposes of drug discovery.

amino acid sequence information, Gene Ontology annotations, literature predications, related articles from PubMed, a link to the species the venom is from, and others). Since literature predications are highly redundant and often uninformative for most uses, the web interface collapses duplicate predications and highlights those likely to be clinically relevant (based on the UMLS semantic types of the subject and object concepts). Additionally, tabs at the top of the data detail page allow the user to view the record in JSON (JavaScript Object Notation) format or download the record as a JSON text file. Users can run BLAST on the amino acid sequences for protein data records, and we plan to add other external analysis tools in the near future. Whenever possible, species pages provide a complete taxonomic lineage for the venomous species being described (the major exception to this is for some species of scorpion, which are interestingly underrepresented in ITIS—the public database we used to source taxonomies). Where appropriate, an image is displayed showing the current data element. At the bottom of each page is a list of external identifiers corresponding to the element currently being viewed. If users find an error in any given data element, a button allows them to report the issue to the website's administrator.

Since VenomKB focuses largely on characteristics of venoms related to drug discovery, pages corresponding to venom proteins that have led to the development of approved drugs also contain information about those drugs and the endogenous human structures that these proteins (and, by extension, their derived drugs) target. An example of these website components is shown in **Figure 2.11**.



Derived Drugs

Drug name: Exenatide

Synonyms and brands: Bydureon, Byetta Molecular weight: 4186.63 g/mol

Exenatide is a peptide derived from the hormone Exendin-4, which is found in the saliva of the Gila monster (Heloderma suspectum) and used to treat type 2 diabetes, administered via injection. Exenatide was first approved for medical use in the United States in 2005. The peptide is a homolog for glucagon-like peptide-1, which acts endogenously to modulate blood glucose levels in various ways. Other potential uses for exenatide (currently being investigated) include treatment of irritable bowel syndrome and Parkinson's disease.

ATC code: A10BJ01 Drugbank ID: DB01276 RxNorm RXCUI: 60548

Protein targets

Target: Glucagon-like peptide 1 receptor

Synonyms: GLP-1 receptor, GLP-1R Target gene(s): GLP1R

Receptor protein found on cells in the pancreas, responsible for enhancing levels of glucose secreted into the blood. GLP1R is a G protein-coupled receptor, consisting of 1 extracellular and one transmembrane domain. Endogenous activators of the receptor in humans include GLP-1 and glucagon.

Venom protein's mode of action: Inhibition

Implicated diseases:

Disease or condition	UMLS CUI
Diabetes Mellitus, Non-Insulin-Dependent	C0011860
Irritable Bowel Syndrome	C0022104

Source: PDB

Figure 2.11.: Example of a venom-derived drug and a venom protein target, taken from the page for Exendin-4 (VenomKB ID: P5730495). Similar components are included in VenomKB for all venom proteins that have led to the development of an approved drug, and more will be added in the future, as experimental drugs reach the market.

Table 2.10.: Version differences; VenomKB v1.0 vs. v2.0

	VenomKB v1.0	**VenomKB v2.0**
Web framework	Ruby on Rails	Node.js + React + Express
Database back-end	PostgreSQL	MongoDB
Database structure	3 unstructured SQL tables	Structured documents mapped to Venom Ontology
API	None	REST API, implemented in Mongoose.js
Legacy support	n/a	VenomKB v1.0 rows mapped to v2.0 documents

VenomKB augments existing knowledge using ontological inference

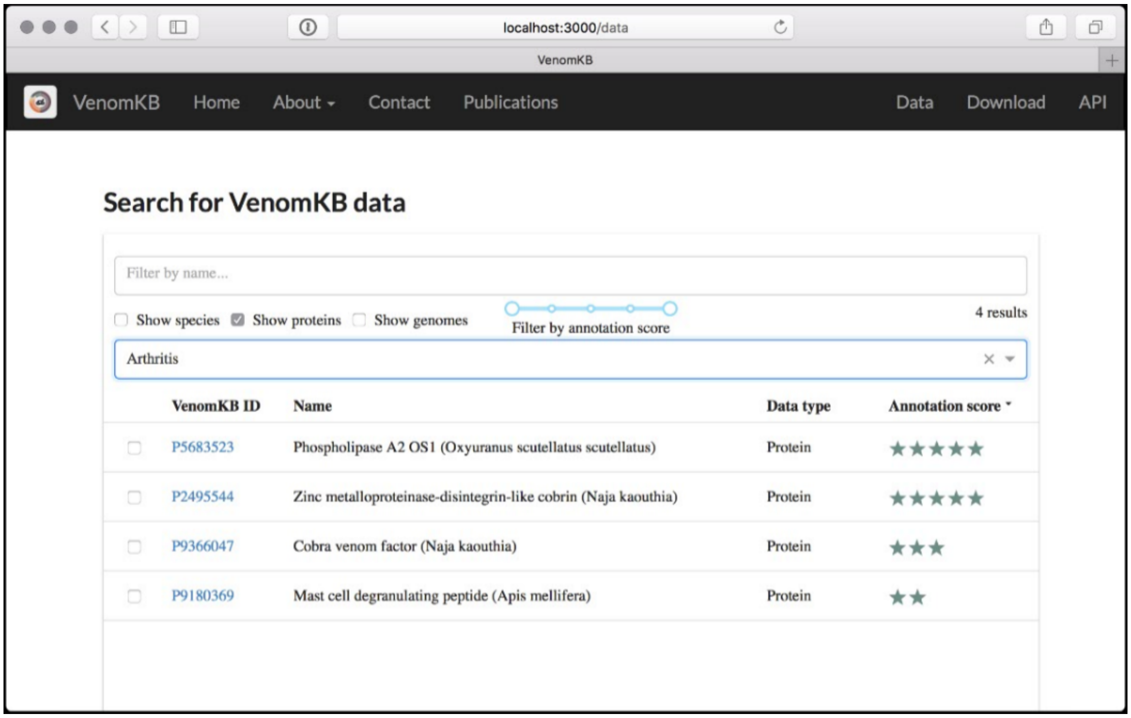
There are generally two types of ontological inference in VenomKB, both of which are dependent on the structure of the Venom Ontology: 1.) Inferred data types and 2.) inferred data associations. Currently, the only inferred data type in VenomKB is "Systemic Effects", which are diseases and conditions that are either associated with or resulting from the administration of a venom or venom component to the human body. Another inferred data type that we plan to add in the future is "Molecular Effects", which are the specific effects that venoms and their components have on biomolecular structures at the cellular or sub-cellular level in the human body. By using the structure of the Venom Ontology, we can use class assertions to infer and validate the molecular effects associated with diseases and conditions, and potentially discover new disease/condition associations for venoms and venom components.

In **Figure 2.10** we illustrate a specific example of the type of observation that can be made using a combination of VenomKB's data and ontological inference. Here, we define venom complexity as the number of unique protein components in a species' venom. By grouping species in VenomKB using the available taxonomic hierarchy and then counting the number of linked protein records for those species, we can plot distributions of venom complexity for major taxonomic clades, such as reptiles, insects, molluscs, and others. In addition to highlighting the relative lack of mammals, fish, and amphibians in VenomKB (and, by extension, other databases containing venom data), these distributions highlight that insect venoms seem to be of lower complexity than arachnid, reptile, and mollusc venoms. This observation may be useful for the purposes of drug discovery—for example, it could suggest that components of insect venom tend to be less specific in their molecular targets, perhaps so they have activity in a wider range of species (which is well-supported in the literature of evolutionary toxinology) [125, 219].

Heuristic annotation scores provide a relative measure of data quality

A major aspect of creating publicly available databases for science is to provide methods for assessing the quality of the data. Data quality can be assessed using two general approaches: task-based assessment, and by performing intrinsic tests on the data records. Intrinsic assessments of data quality are challenging, especially when designing inferred data types that lack a baseline reference. One noteworthy example of addressing this issue is in the UniProt database, where data elements are given scores that indicate completeness and confidence in the assertions made by that element. However, few structured databases outline an objective approach to assigning quality scores.

We defined heuristic annotation scores for each data record in VenomKB, which are designed to provide a means for comparing the quality of VenomKB data entities relative to



Search for VenomKB data

Filter by name...

□ Show species ☑ Show proteins □ Show genomes

Filter by annotation score

4 results

Arthritis

	VenomKB ID	Name	Data type	Annotation score
□	P5683523	Phospholipase A2 OS1 (Oxyuranus scutellatus scutellatus)	Protein	★★★★★
□	P2495544	Zinc metalloproteinase-disintegrin-like cobrin (Naja kaouthia)	Protein	★★★★★
□	P9366047	Cobra venom factor (Naja kaouthia)	Protein	★★★
□	P9180369	Mast cell degranulating peptide (Apis mellifera)	Protein	★★

Figure 2.12.: Interface for graphically searching and browsing data in VenomKB. Users can search by string, data type, data quality score, and by disease/condition annotation. The query results page allows sorting by various fields. To access a particular data record, click on the VenomKB ID corresponding to the entry of interest.

all other entities of the same data type. These scores are represented as integers in the range [1 . . 5], inclusive, and are displayed as 'star' icons on the data browse and data detail pages in the web application. To ensure that these quality measures are well-distributed within each data type, we balanced the number of elements attaining each of the five possible scores. The procedure we used to create these annotation scores is described in Experimental Procedures.

Data availability and programmatic access to VenomKB

All data and code related to VenomKB are freely and publicly available online. A version-controlled Git repository for a.) generating the database back-end and b.) the VenomKB web application itself can be accessed at `http://github.com/jdromano2/venomkb`. The code used to generate the database is written in the Python programming language, and it uses the PyMongo library to populate a MongoDB database instance with the generated data. The web application is written in JavaScript (using the React library to design the user interface and Redux to represent the internal state of the data model), and communicates with the MongoDB back-end via a REST API (Application Programming Interface) that is also accessible for programmatic access by end-users. The API functionality is documented on VenomKB's website at `http://venomkb.org/about/api`.

Table 2.11.: Data sources used in VenomKB v2.0.

Data source	Used for
ToxProt	Most molecular data in "Protein" records
NCBI Taxonomy	Species nomenclature data
ITIS	Species taxonomic hierarchies
Protein Databank (PDB)	Protein images
Wikimedia Commons	Species images
MEDLINE / SemMedDB	Structured semantic predications
VenomKB v1.0	Raw semantic predications
Gene Ontology	GO protein annotations

2.3.2. Discussion

Advantages of VenomKB over existing venom databases

To our knowledge, VenomKB is one of five public databases focused on venoms and their components. In designing VenomKB, we aimed to improve on a number of characteristics that make these databases unsuitable for many tasks. The other four databases are UniProtKB/Swiss-Prot's Tox-Prot dataset [113], ConoServer [115, 116] and ArachnoServer [192] databases, and the Animal Toxin DataBase (ATDB) [94]. ConoServer and ArachnoServer are each focused on specific clades of venomous animals (cone snails and arachnids, respectively). Tox-Prot is a relatively small component of the much larger UniProtKB, and therefore does not have the ability to support many of the characteristics unique to venoms. ATDB seems to no longer be available for public use, at the time of writing.

VenomKB seeks to address each of these shortcomings. Of particularly critical importance is VenomKB's inclusion of several datatypes that are present in none of the alternative venom databases. This includes inferred disease/condition associations, explicit representations of the animal species from which the proteins are derived, publicly available genome data, and the semantic predications extracted from previous scientific publications. As described by [77], types of data like these are critical to the drug discovery process. For example, if a protein has a known therapeutic effect but is too toxic to administer to humans, similar species may synthesize less toxic alternatives.

VenomKB is not limited to certain clades of venomous species. In addition to improving the coverage of the data, this also allows users to compare characteristics of venoms that have similar properties despite coming from unrelated species. However, it does limit its focus to venoms and concepts related to venoms, which allowed us to structure the knowledge base around the Venom Ontology and exploit the unique semantic features of venoms in a way to make inferences that would otherwise be challenging. We specifically host VenomKB on its own domain (`venomkb.org`) instead of on an institutional website: Since institutional websites and affiliations tend to change, having a dedicated domain name improves the site's sustainability model.

Extending the VenomKB technique beyond venoms

Although VenomKB was designed specifically to manage venoms and venom component data, it is reasonable to assume that our techniques could be extended to other similar domains of interest. Plant metabolites, in particular, provide an interesting target, especially given that they already comprise a major source of approved therapeutics worldwide [56, 181]. The process of translating the structure of VenomKB to another domain would essentially involve three steps: (1) redefining the ontology on which the knowledge base is built (e.g., creating an appropriate plant metabolite ontology), (2) finding the appropriate data sources for populating the knowledge base, and (3) making inferences to define new data types where possible.

VenomKB as a model for open access of scientific data

As mentioned in the previous section, all code and data related to VenomKB are freely accessible to the public. These resources are maintained under the open-source GNU General Public License v3 [247], which permits use, reuse, and modification under limited terms. A copy of this license is distributed as part of the source code repository.

Limitations

VenomKB is limited by a general lack of availability of venom data. Given that scientists believe there may be millions of venomous species on the planet [229], the 632 species represented in VenomKB comprise only a miniscule fraction of the total. This disparity is even more apparent when viewed from the perspective of whole-genome sequencing data: VenomKB only contains 5 species' whole genomes (which, as stated before, is the entirety of publicly available genomes from venomous species, at the time of writing). This issue is exacerbated further by the fact that it is often challenging to tell whether a species is venomous or not—for example, it was only discovered in 2009 that the common octopus (*Octopus vulgaris*) is venomous, since the octopus is neither aggressive, nor is the venom appreciably toxic to humans [212].

Although VenomKB contains novel data in the form of literature predications and automatically inferred disease/condition associations (as well as the ontological relationships between datatypes), much of the knowledge base is aggregated from previously compiled data sources, such as UniProtKB, NCBI, and others. However, in the near future, VenomKB will soon include novel experimental data in the form of human gene expression profiles that capture transcriptional responses to being exposed to specific animal venoms.

Future additions to VenomKB

VenomKB is—and likely will remain—a work in progress. Our goal is to provide a *comprehensive* knowledge resource for computational toxinology, but due to both the breadth of venom data types (experimental, clinical, molecular, etc.), and the rapid generation of new venom data, it is unlikely that any venom data resource will ever be truly comprehensive.

To address this challenge, a crucial aspect of VenomKB is a map of current and planned features that grows with and adapts to the evolving needs of the toxinology and drug discovery communities. This feature map is available to view at `http://venomkb.org/about/features/`. Aside from the novel gene expression profile data that was mentioned previously, important additions in the near future include the following:

- Important pharmacokinetic and biochemical measures (when known), such as IC_{50}, K_i, and molecular mass
- Additional gene-level data, including nucleotide sequences, protein isoforms, and gene families
- Annotations to clinical trials exploring particular venom compounds
- Metrics related to whole genomes, such as total size and sequencing methods used
- Species-level data related to natural uses of venoms, such as predation/defense, venom delivery, and target species

Furthermore, we strongly encourage input from researchers who could benefit from additional features. Contact methods for the authors are provided on the VenomKB website, at `http://venomkb.org/contact`.

2.3.3. Methods

The original version of VenomKB was written using the Ruby on Rails web framework for the Ruby programming language, but for v2.0 we rewrote the entire web application in JavaScript, using the React.js library to implement the interactive user-interface, and the Mongoose library to construct the data model for the REST API. The differences between v1.0 and v2.0 are summarized in **Table 2.10**. We maintain the database back-end for

VenomKB on a MongoDB server that is separate from the web application for security and performance.

We constructed the database using an iterative approach, starting with data aggregated from existing databases and then transitioning to the addition of inferred and novel data types. To serve as a starting point for building the database, we treat the ToxProt venom protein annotation program as a gold-standard, being arguably the most complete existing venom database that is not constrained to a certain set of taxa. First, we retrieved all venom peptides in the ToxProt database and extracted core attributes relevant to VenomKB (such as amino acid sequences and cross-references to other databases). We then retrieved taxonomy data for all species with at least one peptide, and used both the NCBI Taxonomy database [72] and the Integrated Taxonomic Information System (ITIS) to build taxonomic lineages and to retrieve other species-level data, such as common names, synonyms, and external identifiers.

To link literature annotations and predication data from VenomKB v1.0 to the new knowledge base, we used expert-identified literature references provided by the ToxProt program. For each PubMed identifier in ToxProt, we retrieved corresponding VenomKB v1.0 predications, and linked them to both their respective protein and species data records. Since many literature annotations are duplicated both within a single document and between multiple documents, we merged duplicate records.

In VenomKB, we represented data provenance using the PROV-DM data model standard [170]. Data provenance is a representation of the sources of each data type in VenomKB along with the methods employed to manipulate and restructure data. Beyond accountability and reproducibility, provenance allows for data quality assessment [89]—data aggregated, created, or validated by more rigorous methods generally are deemed to be of better quality than otherwise. The provenance model for VenomKB can be downloaded from the website, at `http://venomkb.org/download`.

Generating balanced heuristic annotation scores

In Results we explain the use of heuristic annotation scores to provide a method for comparing data quality and completeness relative to VenomKB's other data elements of the same type. We accomplished this task by first assigning raw (unscaled) scores to each instance of each data type based on presence and absence of certain elements. For example, the raw score of a protein was increased by 0.05 for each literature predication, and decreased by 0.2 if it had no literature predications. A species' raw score was increased by 3.2 if a complete taxonomic lineage was present, and decreased by 1.0 if no image of that species was available. The complete details for assigning raw scores is outlined in the VenomKB code repository—the following Python code sample (from `generate_annotation_scores.py`) shows the scoring function for a `Protein`:

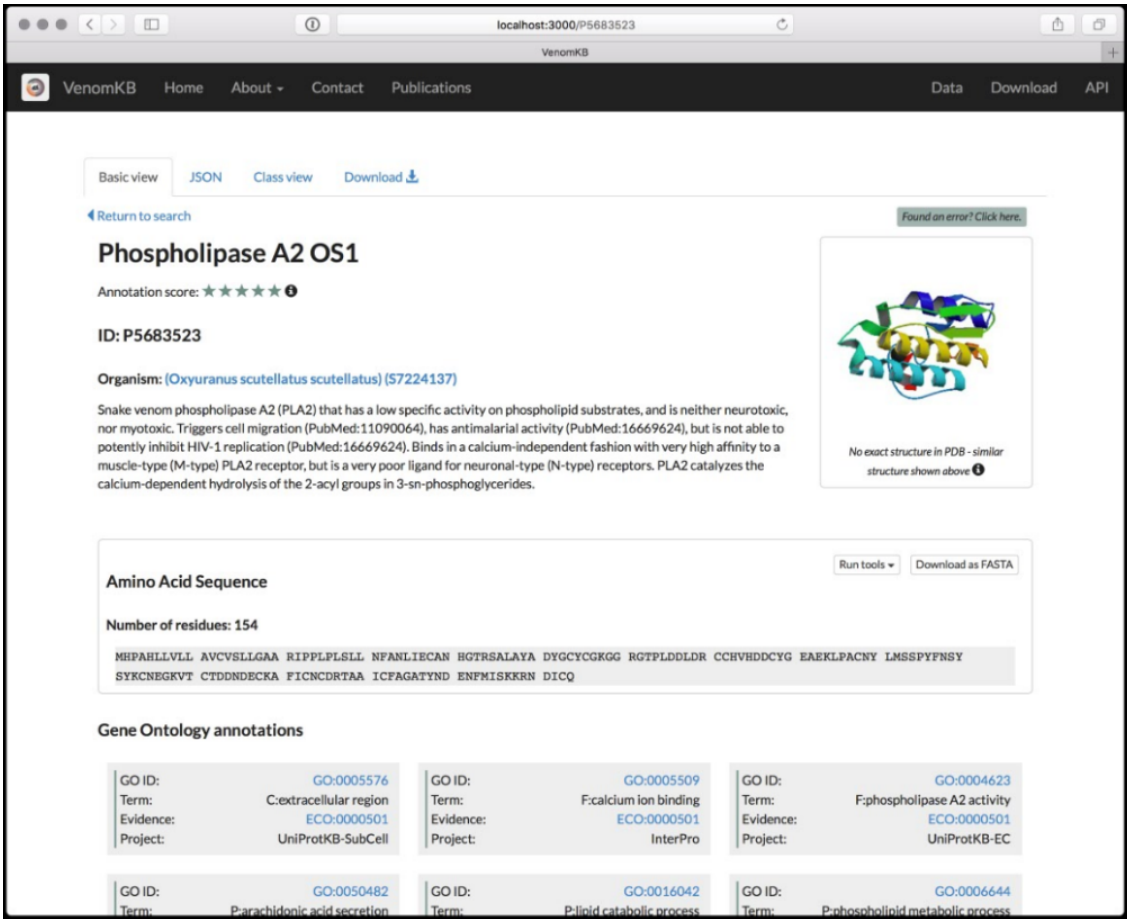
```
def score_protein(p):
    """Score a protein.
    Keyword arguments:
    p -- The protein, as an encoded JSON document.

    Returns:
    Floating point value >= -4.0"""
    score = 0.
    if ’pdb_structure_known’ in p.keys():
        if p[’pdb_structure_known’] == True:
            score += 3
    if ’pdb_image_url’ in p.keys():
        if p[’pdb_image_url’] != "":
            score += 1.
        if ’None’ in p[’pdb_image_url’]:
            score -= 4
    if ’description’ in p.keys():
        score += 1.
    score += (len(p[’out_links’]) * 0.1)
    if ’literature_predications’ in p.keys():
        score += 0.2
        score += (len(p[’literature_predications’]) * 0.05)
    return score
```

After computing raw scores, we then adjusted the scores for each data type to a discrete uniform distribution on the range $[1\,..\,5]$ using the following transformation:

$$\mathbf{x}' = \left\lfloor \left(\frac{\mathbf{x}_{(i)}}{|\mathbf{x}|} * (5-1) \right) + 1 \right\rfloor$$

where $|\mathbf{x}|$ denotes the number of elements of data type $\mathbf{x}$, $\mathbf{x}_{(i)}$ is the vector of order statistics for the raw scores of data type $\mathbf{x}$, and $\mathbf{x}'$ is the vector of transformed scores. This procedure produces five evenly sized bins from 1 to 5 for each data type in VenomKB.



Phospholipase A2 OS1

Annotation score: ★★★★★

ID: P5683523

Organism: (Oxyuranus scutellatus scutellatus) (S7224137)

Snake venom phospholipase A2 (PLA2) that has a low specific activity on phospholipid substrates, and is neither neurotoxic, nor myotoxic. Triggers cell migration (PubMed:11090064), has antimalarial activity (PubMed:16669624), but is not able to potently inhibit HIV-1 replication (PubMed:16669624). Binds in a calcium-independent fashion with very high affinity to a muscle-type (M-type) PLA2 receptor, but is a very poor ligand for neuronal-type (N-type) receptors. PLA2 catalyzes the calcium-dependent hydrolysis of the 2-acyl groups in 3-sn-phosphoglycerides.

No exact structure in PDB - similar structure shown above

Amino Acid Sequence

Number of residues: 154

```
MHPAHLLVLL AVCVSLLGAA RIPPLPLSLL NFANLIECAN HGTRSALAYA DYGCYCGKGG RGTPLDDLDR CCHVHDDCYG EAEKLPACNY LMSSPYFNSY
SYKCNEGKVT CTDDNDECKA FICNCDRTAA ICFAGATYND ENFMISKKRN DICQ
```

Gene Ontology annotations

GO ID:	GO:0005576	GO ID:	GO:0005509	GO ID:	GO:0004623
Term:	C:extracellular region	Term:	F:calcium ion binding	Term:	F:phospholipase A2 activity
Evidence:	ECO:0000501	Evidence:	ECO:0000501	Evidence:	ECO:0000501
Project:	UniProtKB-SubCell	Project:	InterPro	Project:	UniProtKB-EC
GO ID:	GO:0050482	GO ID:	GO:0016042	GO ID:	GO:0006644
Term:	P:arachidonic acid secretion	Term:	P:lipid catabolic process	Term:	P:phospholipid metabolic process

Figure 2.13.: Example of a page containing a single venom protein. Userschoose the way that they view data using the tabs at the top of the interface. The user is presented with an image of the protein, descriptive information, a link to the species from which the protein was discovered, amino acid data (with links to external tools such as BLAST), and gene ontology annotations. Other fields are out of view, including literature predications, links to external databases, and related publications from MEDLINE.

Chapter 3.

A transcriptomic approach for generating therapeutic effect data from venoms

3.1. Introduction (`VenomSeq`)

In Chapter 1, we introduced `VenomSeq` as a new platform for creating new next-generation sequencing data from venoms that can be used to discover therapeutic associations. Briefly, `VenomSeq` involves exposing human cells to dilute venoms, and then generating differential expression profiles for each venom, comprised of the significantly up- and down-regulated genes in cells perturbed by the venom. We then compare the differential expression profiles to data from public compendia of perturbational gene expression data and gene regulatory data corresponding to disease states. `VenomSeq` works in the absence of any predefined hypotheses, instead allowing the data to suggest hypotheses that can then be explored comprehensively using rigorous traditional approaches.

3.1.1. Enrichment analysis

A major challenge in working with large scale -omics datasets lies in finding parametric representations for higher-order biological phenomena that allow us to assess their statistical significance in specific experimental contexts. For example, genome wide association studies

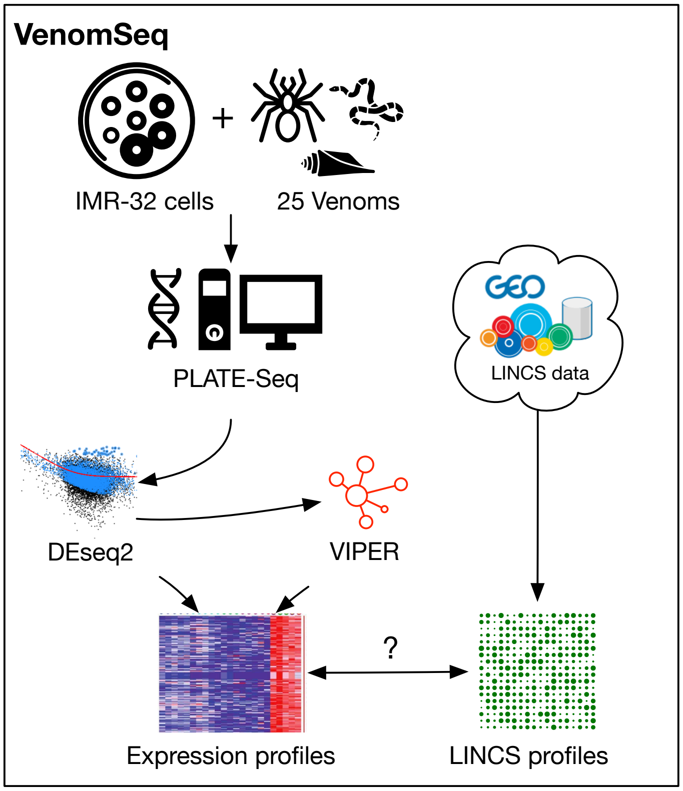


Figure 3.1.: Graphical abstract outlining the `VenomSeq` workflow.

(GWAS) often assess the significance of individual single nucleotide polymorphisms (SNPs) with statistical tests that assume normality in the distribution of the target trait, in spite of this usually being unrealistic [33]. When performing transcriptomic research, the relationships between expression levels of individual genes and varying states of cellular perturbation are even *more* complex, being dependent on vast numbers of overlapping regulatory processes and signaling cascades [139], as well as naturally-occurring randomness [120]. To circumvent this issue, biostatisticians instead turn to nonparametric hypothesis tests, which make no assumptions about the underlying distribution of the data. These usually rely on statistics (which are just functions of a sample and therefore nonparametric) to derive a measure of significance. For example, the Mann-Whitney U test is a nonparametric test that rely on a statistic U that is derived from the sum of ranks within a group of interest.

One of the most valuable classes of nonparametric statistical tests for transcriptomic and gene expression analyses is known as *enrichment analysis*, popularized in 2004 in a method named gene-set enrichment analysis (GSEA) [235]. Briefly, the goal is to determine whether a subset S of members in a larger set G (e.g., genes involved in a constrained metabolic pathway within the set of all genes in the human transcriptome) tend to occur closer to the front or the back of a list that orders the members of the set with respect to some value of interest (e.g., relative expression level). Enrichment analysis deserves special consideration in this dissertation, since it shows up in no fewer than three of the algorithms described in this chapter (connectivity score computation, msVIPER, and enrichment of phenotypes in DisGeNET—all described in §3.4.6).

The key statistic in enrichment analysis is known as an *enrichment statistic* or *enrichment score* (ES), which is itself derived from a nonparametric test named the Kolmogorov–

Smirnov test (or KS test). The procedure for finding the ES is as follows:

1. Construct a vector of ranks L with respect to the measure of interest (e.g., relative expression) over all members of G.
2. Traverse the elements of L from the front to the back.
3. Maintain a running score, where at each element $l \in L$ you either add a quantity (if $l \in S$) or subtract a quantity (if $l \notin S$)[1].
4. Set ES to the value of the running score with the greatest magnitude.

ES can be positive or negative—a positive ES indicates a shift towards the front of the ranked list (i.e., 'enriched'), while a negative ES indicates a shift towards the back of the ranked list (i.e., 'depleted'). Statistical significance of the ES can be assessed in one of two ways: (1) Via comparison to a critical value determined analytically using the Kolmogorov–Smirnov distribution (related to Brownian motion and based on the idea that ES should behave like a "random walk" under the null hypothesis), or (2) by comparing ES empirically to a null model generated by randomly permuting either the samples or the features of the dataset a large number of times. In practice, most studies use the permutation approach rather than the analytical approach.

While permutation tests such as these are criticized for reduced statistical power over analytic alternatives [185], the approach does provide one crucial advantage in the context of `VenomSeq`: since we use a chain of algorithms where the input to one is the set of statistically significant elements identified by the previous (creating a kind of "stacking" of statistical models), finding a closed form of an analytical solution would be substantially more complex

[1] The quantity to add or subtract changes based on the specific task. It can range from a fixed quantity to a quantity whose magnitude is different for each $l \in L$.

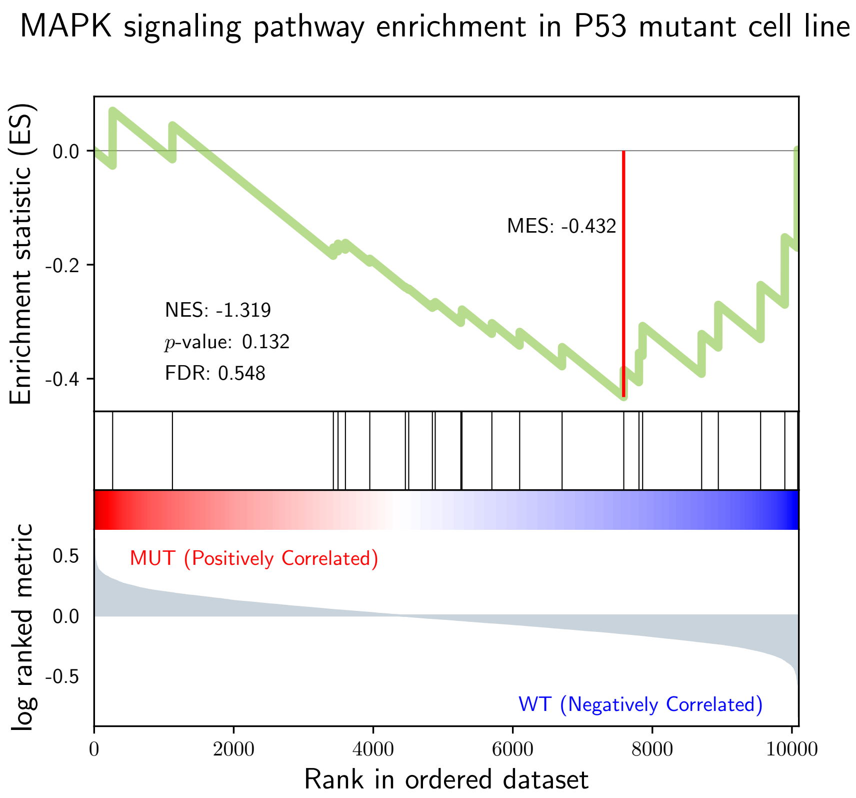


Figure 3.2.: Example of of enrichment analysis using GSEA, with a null model consisting of 1000 iterations of phenotype permutation. The analysis indicates depletion of MAPK signaling pathway gene expression in P53-mutant cells, but does not pass the significance threshold. Notice that ES increases by a magnitude determined by rank correlation of the gene with the phenotype.

and error-prone than simply performing permutation tests at each stage in the process[2].

3.2. Results (VenomSeq)

[2] Although we do still have to be careful of certain mechanistic factors, such as whether to permute *genes* or *phenotypes* [110]

Table 3.1.: Statistics for *S. maurus* growth inhibition data.

***S. maurus* venom vs. IMR-32**		
GI_{20}(µg µl^{-1})		0.0926
R^2		0.991
Hill slope	Bottom	−2.096
	Top	92.572
	log GI_{50}	−0.640
	Slope (h)	−1.928

3.2.1. Venom dosages

In order to optimize the exposure concentrations of each venom, we performed growth inhibition assays on human cells exposed to varying concentrations of the venoms. This is necessary to minimize the impact of toxicity while ensuring the venom is in high enough concentration to exert an effect on the human cells. Since each venom is comprised of many (largely unknown) molecular components, we performed the assays on samples of venom measured in mass per volume, rather than compound concentration (molarity). We used GI_{20}—the concentration of a venom at which it inhibits growth of the human cells by 20%—as the effective treatment dose in all subsequent experiments.

The experimental GI_{20} values and complete dose-response data for each of the 25 venoms are provided in **Appendix A** (**Table A.1**), a sample of which is reproduced (for *S. maurus*) in **Table 3.1**. The resulting growth inhibition curves for all venoms are shown in **Figure 3.3**. Venoms from *L. colubrina*, *D. polylepis*, *S. verrucosa*, *S. horrida*, *C. marmoreus*, *O. macropus*, and *P. volitans* did not demonstrate substantial growth inhibition at any tested concentration, so for those venoms we instead performed sequencing at 1.0 µg µl^{-1}, which is the highest concentration used in the growth inhibition curves.

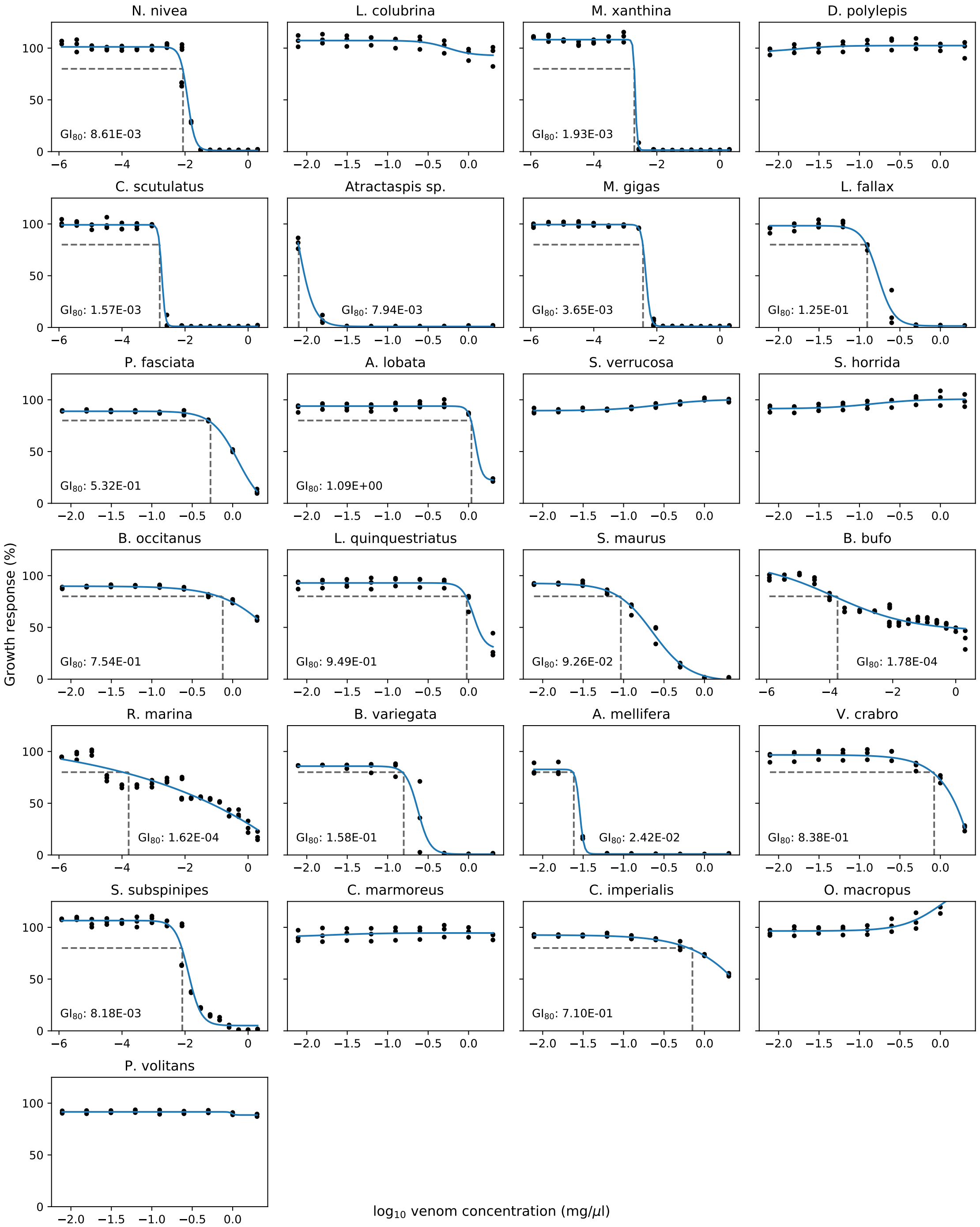


Figure 3.3.: Growth inhibition plots for each of the 25 venoms. GI$_{80}$ values are provided, unless growth inhibition was not observed (in which case sequencing was instead performed at $2\,\mathrm{mg\,\mu l^{-1}}$).

Table 3.2.: Experimental conditions for RNA-Seq.

Venoms	25 species
Cell line	IMR-32 (Human neuroblastoma)
Dosage	GI_{20} for each venom
Time points	6/24/36 hours post-treatment
Replicates	3 per time point per venom
Controls	12 water controls, 9 untreated
Solvent	Water

3.2.2. mRNA sequencing of venom-perturbed human cells

After determining appropriate dose concentrations for each venom, we performed RNA-Seq on human IMR-32 cells exposed to the individual venoms. **Table 3.2** summarizes the experimental conditions used for sequencing. After transforming the raw sequencing reads to gene counts (see §3.4.4), we compiled the results into a matrix, where rows represent genes, columns represent samples, and cells represent counts of a gene in a sample. For detailed quality control data, refer to **Appendix A**, which includes links to related files. The raw (i.e., FASTQ files produced by the sequencer) and processed (i.e., gene counts per sample) data files are available for download and reuse on NCBI's Gene Expression Omnibus database; accession GSE126575.

3.2.3. Differential expression signatures of venom-perturbed human cells

We constructed differential expression signatures for each of the 25 venoms as described in §3.4.5, where each signature consists of a list (length ≥ 0) of significantly upregulated genes, and a list (length ≥ 0) of significantly downregulated genes. The specific expression

Table 3.3.: Partial differential expression signature for *O. macropus*. Most of the significantly differentially expressed genes (35 of 41 total) are omitted for brevity.

Gene	Base mean	$\log_2$-FC	Wald statistic	p-adj
SPRY4	37.38	-2.27534	-3.3084	0.0991
REPIN1	38.30	-0.95256	-4.3326	0.0061
DUSP14	33.88	-0.91311	-3.3327	0.0991
⋮	⋮	⋮	⋮	⋮
BRD3	130.81	1.37645	4.115	.0096
RSRC1	63.48	1.38140	4.2042	0.0091
BAZ1B	120.05	1.69463	5.0846	0.0003

signatures are available on FigShare at `https://doi.org/10.6084/m9.figshare.7609160`. An excerpt from the expression signature for *O. macropus* is shown in **Table 3.3**. The total number of differentially expressed genes for each venom ranges from 2 genes (*Laticauda colubrina* and *Dendroaspis polylepis polylepis*) to 1494 genes (*Synanceia verrucosa*). Note that these signatures are specific to IMR-32 cells—we expect that the same procedure applied to other cell lines would yield substantially different expression signatures.

Gene-wise statistical significance is a function of both $\log_2$ fold change and the number of observed counts. This relationship is illustrated in **Figure 3.4**, which is derived from the same data shown in **Table 3.3** (for *O. macropus*).

3.2.4. Associations between venoms and existing drugs

Using publicly-available differential expression profiles for existing drugs—many with known effects and/or disease associations—we were able to identify statistically significant associations between venoms and classes of drugs. These associations are based on the methods designed by the Connectivity Map (CMap) team [131], and utilize their perturbational differ-

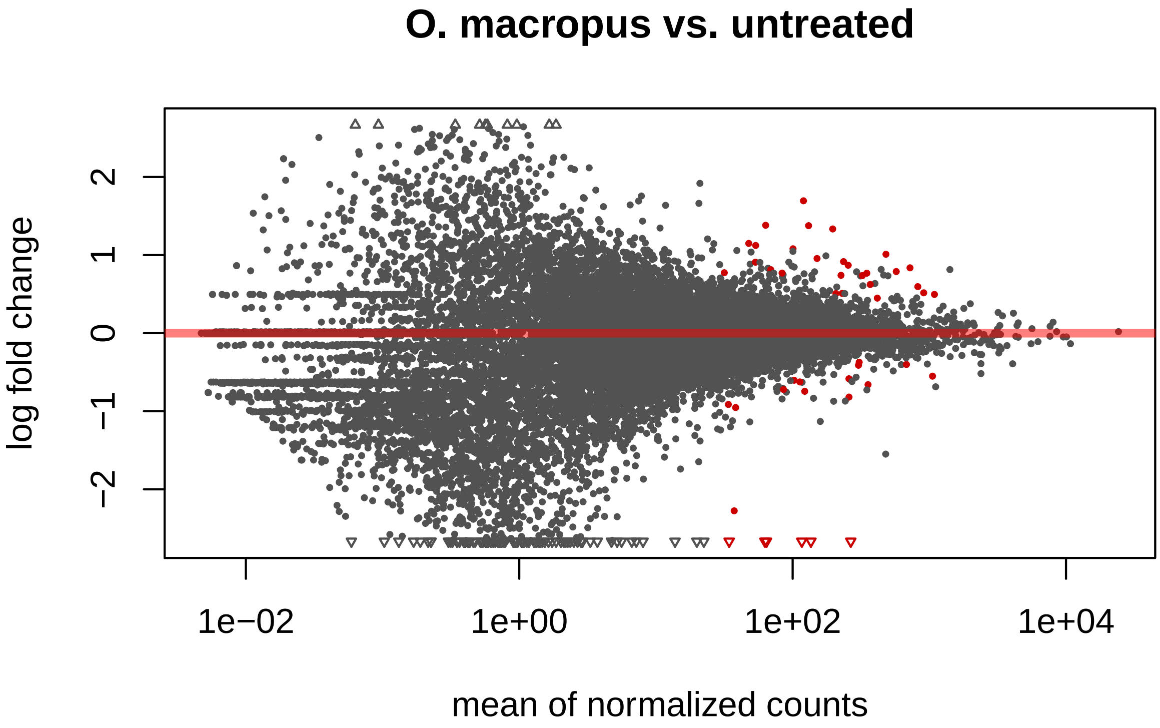


Figure 3.4.: MA plot showing genewise relationship between $\log_2$ fold change and mean of normalized counts in samples corresponding to *O. macropus* venom. Each point represents one gene. Points in red indicate statistically significant genes with regard to differential expression.

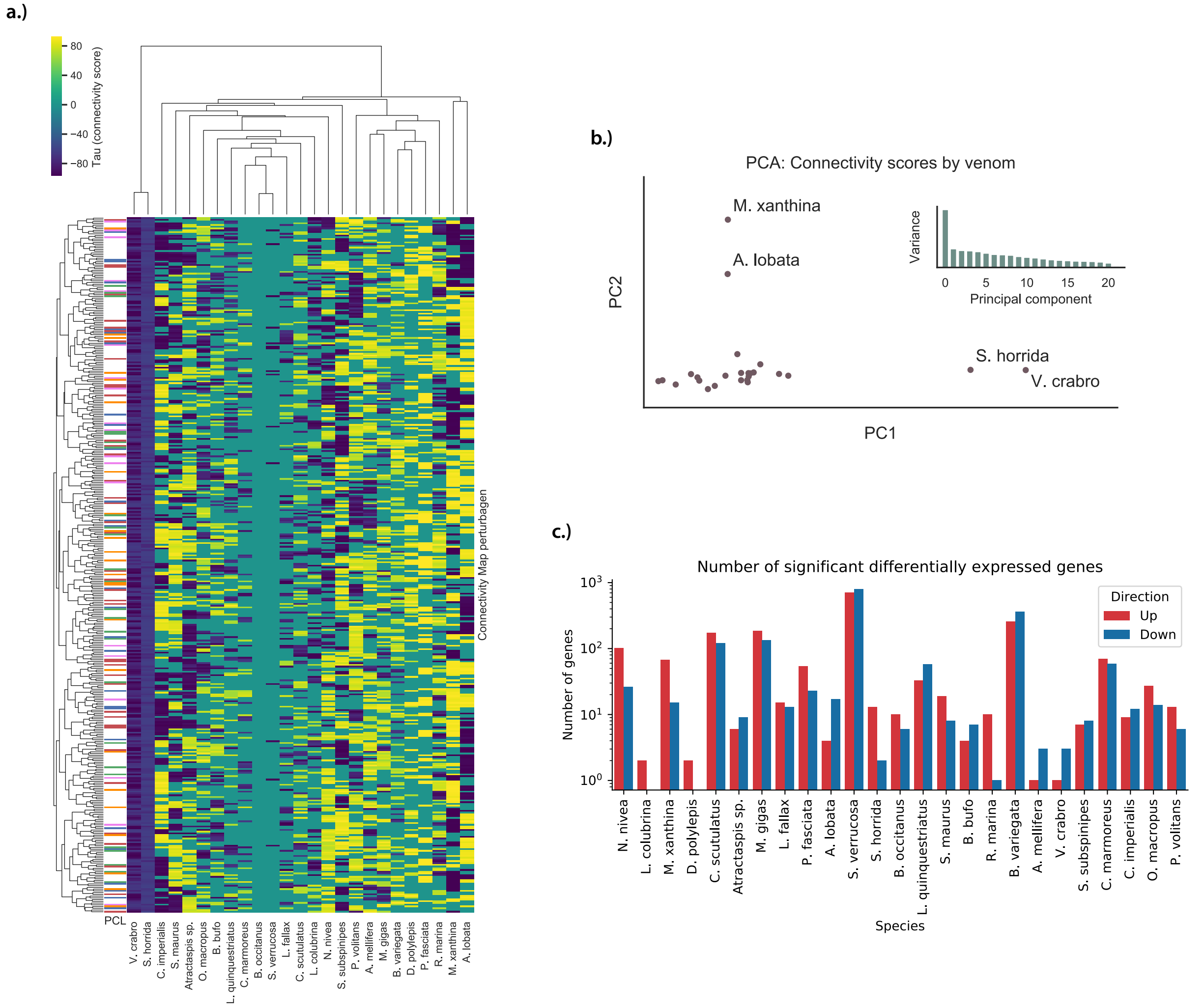


Figure 3.5.: Connectivity analysis results. **a.)** Heatmap of τ-scores between the 25 venom perturbations and the 500 Connectivity Map signatures with the highest variance across all venoms. A distinct hierarchical clustering pattern is evident across the venom perturbations, although it does not conform to any obvious grouping pattern of the venoms. **b.)** Principle component analysis of the 25 venom perturbations, where features are all τ-scores between the venom and signatures from the Connectivity Map reference database. 4 distinct outliers are labeled—these venoms correspond to outliers in the heatmap. Also shown are the ratios of variance explained by each of the first 21 principle components—after the first principle component, the distribution is characterized by a long tail, suggesting that much of the variance is spread across many dimensions, underscoring the complexity of the connectivity score data. **c.)** Barplot showing the number of significant differentially expressed genes for IMR-32 cells exposed to each of the 25 venoms.

ential expression data as the "gold standard" against which to evaluate the venom expression data. In short, this approach uses a Kolmogorov-Smirnov–like signed enrichment statistic to compare a query signature (i.e., venoms) to all signatures in a reference database (i.e., known drugs), normalizing for cell lines and other confounding variables, and finally aggregating scores of 'like' signatures (i.e., drug MoAs) using a maximum-quantile procedure. Complete details of these methods are provided in §3.4.6.

Different venoms yield different profiles of connectivity scores based on the genes present in their differential expression signatures. For example, all connectivity scores between *B. occitanus* and CMap perturbagens are zero, and all connectivity scores between *S. horrida* and CMap perturbagens are negative, which suggest that these venoms either behave like no known perturbagen classes, or that the venoms have no therapeutic activity on IMR-32 cells. Kernel density plots of the connectivity scores for each venom are shown in **Figure 3.6**. In **Figure 3.5**, we show several visualizations of the connectivity analysis results that highlight characteristics of the data. Interestingly, when hierarchical clustering is performed on the connectivity scores by venom perturbation, the venom perturbations form robust clustering patterns that persist across multiple non-overlapping subsets of the connectivity data. This suggests that the clustering corresponds to meaningful characteristics of the venom perturbations in comparison to known drugs, although these characteristics are not readily apparent (i.e., the clustering does not reproduce taxonomy, or other obvious traits of the venoms).

The associations we identified are shown in **Table 3.4**. As we anticipated, only some venoms show strong associations to any classes of drugs. Interestingly, only one venom (*S. subspinipes dehaani*) was linked to an ion channel inhibition MoA—venoms, in general,

Table 3.4.: Venom–drug class associations.

Venom	Drug class (MoA)
Synanceia horrida	ATPase inhibitor CDK inhibitor DNA synthesis inhibitor
Scolopendra subspinipes dehaani	T-type Ca^{2+} channel inhibitor
Pterois volitans	Topoisomerase inhibitor
Argiope lobata	ATPase inhibitor PI3K inhibitor PPARγ agonist
Scorpio maurus	FGFR inhibitor
Rhinella marina	HIV protease inhibitor

tend to have powerful ion channel blocking or activating effects. However, this may be due to a preponderance of non-ion channel MoAs in the CMap data rather than an actual lack of ability to identify ion channel activity.

Many of these MoAs comprise either well-established or emerging classes of cancer drugs. Some that have been used extensively as chemotherapeutic agents include CDK inhibitors (palbociclib, ribociclib, and abemaciclib), topoisomerase inhibitors (doxorubicin, teniposide, and irinotecan, among others), and DNA synthesis inhibitors (mitomycin C, fludarabine, and floxuridine). Meanwhile, PI3K inhibitors and FGFR inhibitors are classes of "emerging" chemotherapy drugs, each recently leading to many high-impact research studies and early-stage clinical trials.

The other classes are indicated for a diverse range of diseases, including circulatory and mental conditions (calcium channel blockers), and cardiac abnormalities (ATPase inhibitors). PPAR receptor agonists have been used to treat diabetes, hyperlipidemia, pulmonary inflam-

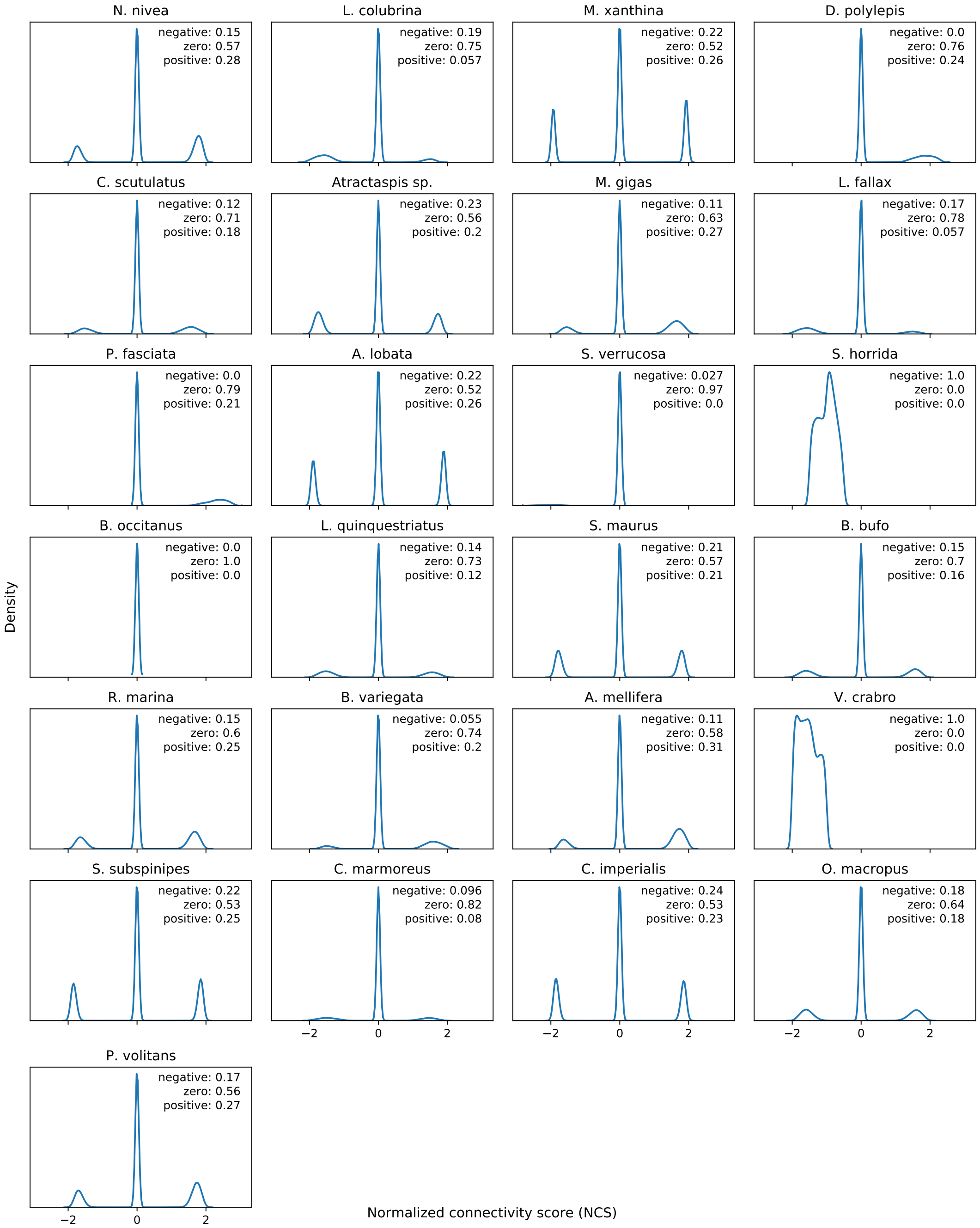


Figure 3.6.: Kernel density plots of normalized connectivity scores (NCSs) for each of the 25 venoms. Note the tendency to introduce sparsity by setting NCS to zero if the quantities a and b have opposite signs (see §3.4.6). Text labels indicate proportion of NCSs for a single venom that are negative, zero, or positive. Each plot is based on 473,647 NCSs (all differential expression profiles in GSE92742 [234]).

mation, and cholesterol disorders.

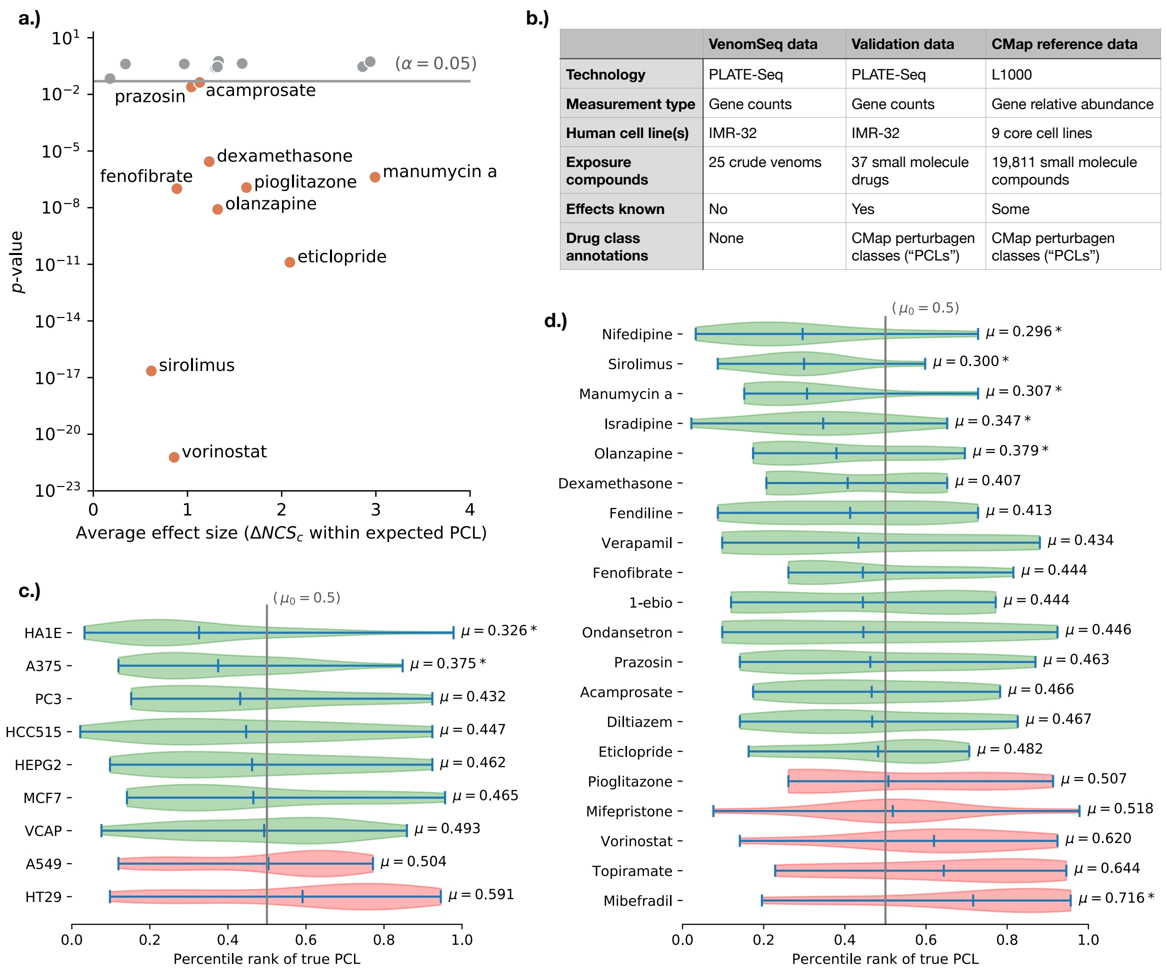
We are in the process of validating several of the associations listed in **Table 3.4** using targeted, cell-based assays, the results of which will be documented in subsequent publications.

VenomSeq technical validation

Following the procedures described in §3.4.7, we used a secondary PLATE-Seq dataset of 37 existing drugs (with known effects) tested on IMR-32 cells to assess whether the sequencing technology (PLATE-Seq) and cell line (IMR-32) employed by VenomSeq are compatible with connectivity analysis and the CMap reference dataset. In this dataset, we were able to map 20 of the 37 drugs to a single existing CMap perturbational class (PCL). The drugs, their modes of action, and the PCLs of which they are members are listed in **Table 3.5**.

VenomSeq technical validation: Recovering connectivity by integrating cell lines When we aggregated all connectivity scores between a known drug and members of the same PCL in the CMap dataset, irrespective of cell line, the connectivity scores are significantly greater than those in a null model in 12 out of 20 instances, which indicates that drugs within the same functional class tend to have more similarities in the query and reference datasets than if the compounds are chosen at random. In all 20 cases, the average effect size[3] was positive, regardless of statistical significance. These—and their corresponding measures of

[3] Effect size is defined as the average difference between connectivities within the expected PCL and the null model of random connectivities for the same query



	VenomSeq data	Validation data	CMap reference data
Technology	PLATE-Seq	PLATE-Seq	L1000
Measurement type	Gene counts	Gene counts	Gene relative abundance
Human cell line(s)	IMR-32	IMR-32	9 core cell lines
Exposure compounds	25 crude venoms	37 small molecule drugs	19,811 small molecule compounds
Effects known	No	Yes	Some
Drug class annotations	None	CMap perturbagen classes (“PCLs”)	CMap perturbagen classes (“PCLs”)

Figure 3.7.: Results of applying the VenomSeq sequencing and connectivity analysis workflow to 37 existing drugs with known effects, to validate the compatibility of PLATE-Seq and IMR-32 cells with the connectivity analysis algorithm and dataset. **a.)** Scatter plot showing validation drugs that are members of a CMap PCL and the mean differences between within-PCL connectivity scores and a null distribution of random connectivity scores for the same drug (**Table 3.6**). Verticle axis shows the p-value of a Student's t-test comparing the within-PCL and null connectivity score distributions (corrected for multiple testing). Statistically significant drugs are labeled by name. **b.)** Summary of the validation strategy, showing that the validation dataset bridges certain gaps between the VenomSeq data and the CMap reference data. **c.)** Distributions of rank percentiles of expected ("true") PCLs within the list of all PCLs ordered by average connectivity score (**Table 3.7**), aggregated by CMap dataset cell lines, and **d.)** validation drugs. Green distributions indicate a shift towards the front of the rank ordered list, indicating stronger compatibility with the PLATE-Seq/IMR-32 query data, based on expected connections, and "*" indicates statistically significant shifts.

Table 3.5.: Drugs used to validate PLATE-Seq and the IMR-32 cell line for connectivity analysis. Not all compounds of a given mechanism of action will necessarily map to that mechanism's associated PCL—PCLs consist of compounds that are members of the same functional class and also have high transcriptional impact.

Drug	Mechanism of Action	CMap perturbagen class (PCL)
Mibefradil	T-type Ca^{2+} channel inhibitor	CP_T_TYPE_CALCIUM_CHANNEL_BLOCKER
Isradipine	L-type Ca^{2+} channel inhibitor	CP_CALCIUM_CHANNEL_BLOCKER
Nifedipine	L-type Ca^{2+} channel inhibitor	CP_CALCIUM_CHANNEL_BLOCKER
Diltiazem	Ca^{2+} channel inhibitor	CP_CALCIUM_CHANNEL_BLOCKER
Verapamil	Ca^{2+} channel inhibitor	CP_CALCIUM_CHANNEL_BLOCKER
Fendiline	Ca^{2+} channel inhibitor	CP_CALCIUM_CHANNEL_BLOCKER
Topiramate	Na^{+} and Ca^{2+} channel modulator	CP_SODIUM_CHANNEL_BLOCKER
Ionomycin	Ca^{2+} channel signal inducer	
1-EBIO	Ca^{2+}-gated K^{+} channel activator	CP_POTASSIUM_CHANNEL_ACTIVATOR
Forskolin	Adenylyl cyclase activator	
Pregabalin	Increases GABA biosynthesis	
Gabapentin	Increases GABA biosynthesis	
Baclofen	GABA$_B$-receptor agonist	
Memantine	Glu-receptor inhibitor	
Acamprostate	Glu-receptor inhibitor	CP_GABA_RECEPTOR_ANTAGONIST
MTEP	Glu-receptor inhibitor	
Ivermectin	Glu-gated Cl^{-} channel inhibitor	
Carbenoxolone	Glucocorticoid metabolism inhibitor	
Mifepristone	Glucocorticoid receptor inhibitor	CP_PROGESTERONE_RECEPTOR_ANTAGONIST
Dexamethasone	Glucocorticoid receptor agonist	CP_GLUCOCORTICOID_RECEPTOR_AGONIST
Aldosterone	Mineralocorticoid receptor agonist	
Spironolactone	Mineralocorticoid receptor inhibitor	
Olanzapine	Dopamine receptor inhibitor	CP_DOPAMINE_RECEPTOR_ANTAGONIST
Eticlopride	Dopamine receptor inhibitor	CP_DOPAMINE_RECEPTOR_ANTAGONIST
Ondansetron	5-HT$_3$ serotonin receptor inhibitor	CP_SEROTONIN_RECEPTOR_AGONIST
Naltrexone	Opioid receptor inhibitor	
Disulfiram	Acetaldehyde dehydrogenase inhibitor	
Cerlitinib	ALK inhibitor	
Crizotinib	ALK inhibitor	
Sirolimus	mTOR inhibitor	CP_MTOR_INHIBITOR
Manumycin a	Farnesyltransferase inhibitor	CP_NFKB_PATHWAY_INHIBITOR
Vorinostat	HDAC (I/II/IV) inhibitor	CP_HDAC_INHIBITOR
Prazosin	Adrenergic receptor inhibitor	CP_BETA_ADRENERGIC_RECEPTOR_AGONIST
Rolipram	Phosphodiesterase-4 inhibitor	
Minocycline	NOS inhibitor	
Pioglitazone	PPARγ/α inhibitor	CP_PPAR_RECEPTOR_AGONIST
Fenofibrate	PPARα agonist	CP_PPAR_RECEPTOR_AGONIST

significance—are shown in **Figure 3.7** and **Table 3.6**. Overall, these data are congruent with those made by the Connectivity Map team in [234]—namely, that expected connections between query drugs and reference compounds can be recovered for some PCLs, but not for others. Importantly, in both our observations and the observations in [234], PCLs related to highly conserved core cellular functions perform better under this approach.

VenomSeq technical validation: Impact of reference cell lines and query drugs on expected PCL percentile ranks Since IMR-32 cells are not present in the CMap reference dataset, we were particularly interested in seeing which cell lines present in the reference dataset (if any) performed better than others at the task of recovering expected connections. Using the PCL ranking strategy described in §3.4.7, 7 of the 9 core cell lines show at least a moderate tendancy to place the true PCL towards the front of the ranked list of all PCLs, indicating that at least some of the ability to recover expected connections is retained when looking at those 7 cell lines individually. PCL rankings stratified by drug (rather than cell line) show a similar pattern—15 of 20 PCL-annotated drugs tend to have the expected PCL ranked towards the front of the list ("enrichment"), while 5 tend to have the expected PCL show up towards the back of the list ("depletion"). Of these 20, the only It should be noted that—due to the rather small number of profiles in the reference dataset that are annotated to PCLs—these two analyses were limited in terms of statistical power, and deserve a follow up analysis in the future, when more PCLs and members of those PCLs are present in the reference database.

Table 3.6.: Enrichment of strong connections in expected PCL annotations . *p*-values correspond to independent, two-sample Student's *t*-tests between "within-PCL" connectivities and a null model of randomly sampled compound connectivities (see text) for the same query drug, and are corrected for multiple testing using the Benjamini-Hochberg procedure. Effect size is the difference of means between those two groups, such that larger effect sizes correspond to higher expected connectivity scores between the query drug and members of its same drug class. Note that effect sizes are relatively small in most cases—this is due in part to the sparsity of connectivity scores.

Drug	PCL	*p*-value	Effect size
Topiramate	`CP_SODIUM_CHANNEL_BLOCKER`	**1.018e-31**	13.168
Vorinostat	`CP_HDAC_INHIBITOR`	**5.952e-22**	1.717
Sirolimus	`CP_MTOR_INHIBITOR`	**2.240e-17**	1.232
Eticlopride	`CP_DOPAMINE_RECEPTOR_ANTAGONIST`	**1.278e-11**	4.175
Olanzapine	`CP_DOPAMINE_RECEPTOR_ANTAGONIST`	**8.117e-09**	2.640
Fenofibrate	`CP_PPAR_RECEPTOR_AGONIST`	**1.012e-07**	1.775
Pioglitazone	`CP_PPAR_RECEPTOR_AGONIST`	**1.158e-07**	3.252
Manumycin a	`CP_NFKB_PATHWAY_INHIBITOR`	**4.124e-07**	5.983
Dexamethasone	`CP_GLUCOCORTICOID_RECEPTOR_AGONIST`	**2.741e-06**	2.462
Prazosin	`CP_BETA_ADRENERGIC_RECEPTOR_AGONIST`	**2.476e-02**	2.083
Acamprosate	`CP_GABA_RECEPTOR_ANTAGONIST`	**4.290e-02**	2.260
Mibefradil	`CP_T_TYPE_CALCIUM_CHANNEL_BLOCKER`	6.871e-02	0.355
1-EBIO	`CP_POTASSIUM_CHANNEL_ACTIVATOR`	2.573e-01	2.597
Fendiline	`CP_CALCIUM_CHANNEL_BLOCKER`	2.854e-01	2.636
Diltiazem	`CP_CALCIUM_CHANNEL_BLOCKER`	2.929e-01	5.719
Isradipine	`CP_CALCIUM_CHANNEL_BLOCKER`	4.062e-01	0.683
Nifedipine	`CP_CALCIUM_CHANNEL_BLOCKER`	4.100e-01	1.932
Mifepristone	`CP_PROGESTERONE_RECEPTOR_ANTAGONIST`	4.309e-01	3.160
Verapamil	`CP_CALCIUM_CHANNEL_BLOCKER`	5.404e-01	5.880
Ondansetron	`CP_SEROTONIN_RECEPTOR_AGONIST`	5.710e-01	2.659

Table 3.7.: Correct PCL ranks aggregated by cell line. Mean rank percentile is the mean rank of the correct ("true") PCL, aggregated over all query drugs and divided by the total number of PCLs (92), reported by cell line.

CMap cell line	Mean rank percentile	FDR-corrected p-value
HA1E	0.326087	0.001663
A375	0.375000	0.004926
PC3	0.431522	0.109226
HCC515	0.446739	0.193877
HEPG2	0.461957	0.258068
MCF7	0.465217	0.279325
VCAP	0.492935	0.443995
A549	0.503804	0.468387
HT29	0.075445	0.591304

3.2.5. Associations between venoms and disease regulatory networks

Direct observations of expressed genes (via mRNA counts) provide an incomplete image of the regulatory mechanisms present in a cell. To complement the CMap approach that focuses on perturbations at the *gene* level, we designed a parallel approach that uses cell regulatory network data to investigate perturbations at the *regulatory module* (e.g., pathways and metabolic networks) level; an approach we refer to as *master regulator analysis*. In master regulator analysis, the ARACNe algorithm [162] is used to obtain regulatory network data for our cell line of interest (in this case, IMR-32), consisting a list of *regulons*—overlapping sets of proteins whose expression is governed by a master regulator (e.g., a transcription factor). The msVIPER algorithm [4] is then used to determine the activity of each regulon by computing enrichment scores from observed expression levels of the genes/proteins contained in that regulon (here, using the RNA-Seq results described in §3.2.2).

We matched the significantly up- and down-regulated master regulators for each venom to diseases using high-confidence TF-disease associations in DisGeNET [193]—a publicly available database of associations between diseases and gene network component. This approach is based on the idea that diseases caused by disregulation of metabolic and signaling networks can be treated by administering drugs that "reverse" the cause (i.e., abnormal master regulator activity) of disregulation. Since we are interested in discovering associations with multiple corroborating pieces of evidence, we specifically filtered for diseases where *two or more* linked TFs are disregulated when perturbed by the venom. The complete list of associations are provided on figshare at `https://doi.org/10.6084/m9.figshare.7609793`; here, we describe a handful of interesting observations.

The most prevalent class of illness (comprising 19.7% of all associations across all venoms) is `DISEASES OF THE NERVOUS SYSTEM AND SENSE ORGANS`. This is not surprising, considering many of the 25 venoms have neurotoxic effects, and IMR-32 is a cell line derived from neuroblast cells. One source of bias in these results is that similar diseases tend to be associated with the same regulatory mechanisms [236]. For example, associations between a venom and schizophrenia will often be co-reported with associations to other mental conditions, such as bipolar disorder and alcoholism.

3.3. Discussion (VenomSeq)

3.3.1. Venoms versus small-molecule drugs

In the connectivity analysis portion of **VenomSeq**, we demonstrated that these techniques have the ability to identify novel venom–drug class associations, and corroborate known venom activity. One distinct advantage of performing queries against the CMap reference dataset is their inclusion of manually-curated PCLs, which allow for normalization of data gathered from multiple perturbagens and multiple cell lines, aggregated at a class level that corresponds approximately with drug mode of action. For this reason, hypotheses generated by the connectivity analysis portion of `VenomSeq` are often testable at the protein level.

One important caveat is that venom components have a tendency to interact with cell surface receptors (e.g., ion channels or GPCRs), inciting various signaling cascades and therefore acting indirectly on downstream therapeutic targets. While this is certainly the case for many drugs as well (GPCRs are considered the most heavily investigated class of drug targets [102]), small molecules often can be designed to enter the cell and interact directly with the downstream therapeutic target. This has important implications regarding assay selection for *in vitro* validation of associations learned through the connectivity analysis. For example, if the MoA of interest is inhibition of an intracellular protein (e.g., topoisomerase), a cell-based assay should be considered when testing venom hypotheses, since the venom likely is not interacting directly with the topoisomerase (and, therefore, the effect would not occur in non-cell based assays).

3.3.2. Venoms versus human diseases

The master regulator analysis portion of `VenomSeq` discovers associations between venoms and the diseases they may be able to treat, rather than to drugs. This could be especially useful for discovering treatments to diseases with no or few existing indicated drugs (or drugs that are not present in public differential expression databases). Additionally, since the master regulator approach is sensitive to complex metabolic network relationships, it is (theoretically) more sensitive to patterns, as well as more suited to diseases with complex genetic etiologies that are not explainable by observed gene counts alone.

Currently, the primary drawback to the master regulator approach is that criteria for statistical significance are not well established. Therefore, it is challenging to determine which venom-disease associations are most likely to reflect actual therapeutic efficacy. As a temporary alternative, we used several heuristics to ensure there are multiple corroborating sources of evidence for the reported associations.

As discussed previously, the connectivity analysis produces hypotheses that are relatively straightforward to validate experimentally, using affordable, widely available assay kits and reagents. Since the master regulator workflow gives hypotheses at the disease level (where the underlying molecular etiologies can be unknown), validation instead needs to be performed at the *phenotype* level, either using animal models of disease, or carefully engineered, cell-based phenotypic assays that measure response at multiple points in disease-related metabolic pathways (e.g., DiscoverX's BioMAP® platform [21]).

3.3.3. Biologically plausible therapeutic hypotheses

`VenomSeq` contains multiple types of data analysis for two reasons: (1) It allows us to cover diseases with a wider array of molecular etiologies, and (2) it provides a means for obtaining multiple pieces of corroborating evidence for a given hypothesis. If a link between a venom and a drug/disease is suggested by both connectivity analysis and master regulator analysis, and there is additional literature evidence that lends biological or clinical plausibility, this increases our confidence that the suggested therapeutic effect is "real".

Argiope lobata venom versus cardiopulmonary and psychiatric diseases

A. lobata is a species of spider in the same genus as the common garden spider. The species is relatively understudied, largely due to its lack of interaction with humans, in spite of being distributed across Africa and much of Europe and Asia. The venom from species of *Argiope* spiders contain toxins known as *argiotoxins* [198], which are harmless to humans, in spite of having inhibitory effects on AMPA, NMDA, kainite, and nicotine acetylcholine receptors, which have been implicated in neurodegenerative and cardiac diseases. `VenomSeq` provides supporting evidence for therapeutic activity in each of these classes.

Connectivity analysis links *A. lobata* venom to ATPase inhibitor drugs (see **Figure 3.8**), which include digoxin, ouabain, cymarin, and other cardiac glycosides, and are used to treat a variety of heart conditions. Another venom-derived compound—bufalin (from the venom of toads in the genus *Bufo*) [133]—is considered an ATPase inhibitor, and has demonstrated powerful cardiotonic effects. Connectivity analysis also links the venom to PPAR agonist drugs, which are used to treat cholesterol disorders, metabolic syndrome, and pulmonary inflammation. Interestingly, PPARγ activation results in cellular protection from NMDA toxicity. Given the known inhibitory effect of argiotoxins on NMDA receptors [172], this is striking and biologically plausible evidence for toxin synergism, where two or more venom components target multiple cellular structures with related functions in order to incite a more powerful response [134].

Master regulator analysis supports these findings, as well. We found that *A. lobata* venom is associated with a number of circulatory diseases, including hypertension, heart failure, cardiomegaly, myocardial ischemia, and others. Additionally, it reveals strong associations with an array of mental conditions, such as schizophrenia, bipolar disorder, and

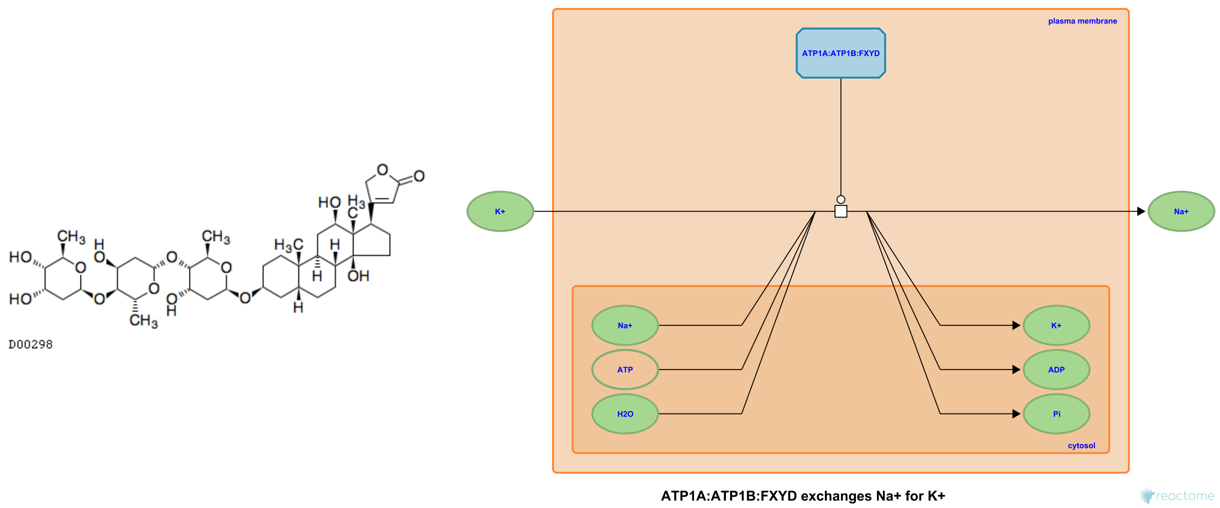


Figure 3.8.: Structure of digoxin (left), a cardiac glycoside that inhibits the function of the Na+/K+ ATPase (ATP1A; right) in the myocardium, which causes a decrease in heart rate [129]. *A. lobata* venom has similar differential expression effects to those of digoxin and other ATPase inhibitor drugs, based on connectivity analysis. Diagram from Reactome [70].

psychosis. These associations are supported by recent research into argiotoxins (and other polyamine toxins), showing that their affinity for iGlu receptors can be exploited to treat both psychiatric diseases and Alzheimer disease [198].

Scorpio maurus venom for cancer treatment via FGFR inhibition

S. maurus—the Israeli gold scorpion—is a species native to North Africa and the Middle East. Its venom is not harmful to humans, but it is known to contain a specific toxin, named maurotoxin, which blocks a number of types of voltage-gated potassium channels—an activity that is under investigation for treatment of gastrointestinal motility disorders [24].

Our connectivity analysis suggests an additional association with FGFR inhibitor drugs. FGFR inhibitors are an emerging class of drugs with promising anticancer activity, and much research focused on them aims to understand and counteract their adverse effects

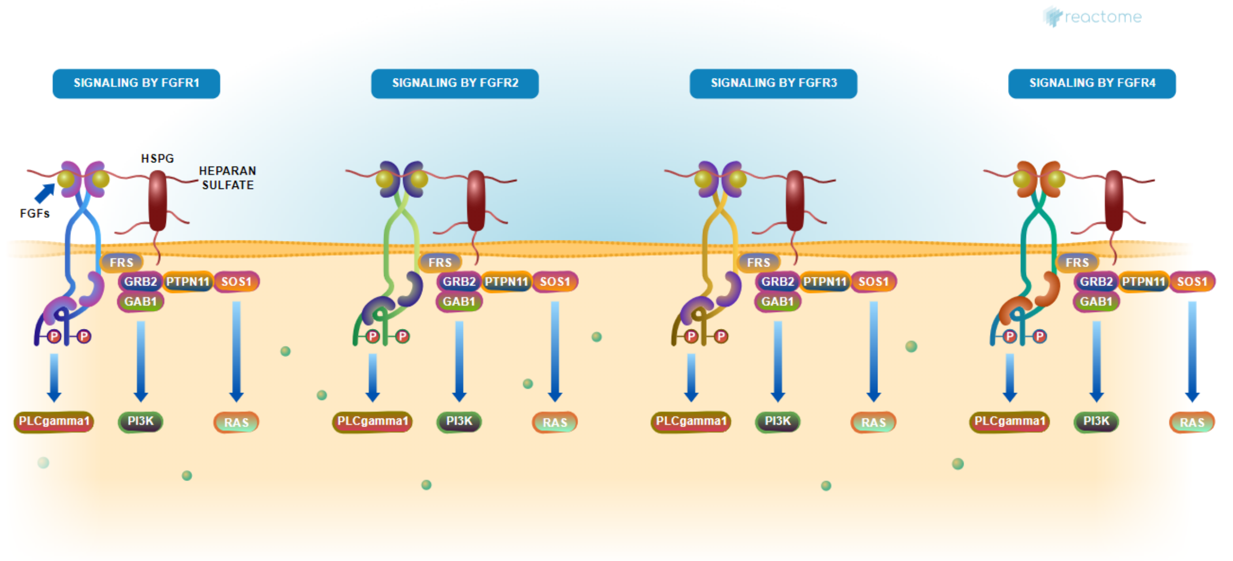


Figure 3.9.: Diagram of FGFR signaling pathways. FGFR inhibitors target 1 of the 4 types of FGFR complexes, abnormal activity of which are involved in angiogenesis. VenomSeq suggests therapeutic similarity between *S. maurus* venom and existing FGFR inhibitor drugs. Pathway diagram from Reactome [53].

(see **Figure 3.9**). Although there is no prior mention of FGFR-related activity from this or related species of scorpions, descriptions of unexpected side effects of *S. maurus* venom on mice provides evidence that such activity could be true. In particular, the venom has been shown to have biphasic effects on blood pressure: when injected, it causes rapid hypotension, followed by an extended period of hypertension. The fast hypotension is known to be caused by a phospholipase A_2 in the venom, but no known components elicit hypertension when administered in purified form [69]. The observed FGFR inhibitor-like effects on gene expression suggest that an unknown component (or group of components) may cause the hypertensive effect via FGFR inhibition. We are currently performing experimental validation of this link, and will report results in future revisions of this manuscript.

3.3.4. Accessing and querying VenomSeq data

`VenomSeq` is designed as a general and extensible platform for drug discovery, and we encourage secondary use of both the technology as well as the data produced using the 25 venoms tested on IMR-32 cells described in this manuscript. We maintain the data in two publicly-accessible locations: (1.) a "frozen" copy of the data, as it exists at the time of writing (on figshare, at `https://doi.org/10.6084/m9.figshare.7611662`), and (2.) a copy hosted on `venomkb.org`, available both graphically and programmatically, and designed to be expanded as new data and features are added to VenomKB.

3.3.5. `VenomSeq` data analysis software

To encourage reuse and reproducibility, we provide an open-source Python package containing all of the data structures and algorithms used in the data analysis portion of `VenomSeq`. The software can be downloaded from its source code repository on GitHub at `https://github.com/jdromano2/venomseq`, or from the Python Package Index at `https://pypi.org/project/venomseq`. The package contains documentation and example code for reproducing the results and figures from this chapter. Several auxiliary datasets (such as the Connectivity Map expression profiles) must be downloaded from their original sources in order to reproduce certain segments of the pipeline, but these are documented where applicable.

3.3.6. Transitioning from venoms to venom components

`VenomSeq` is a technology for discovering early evidence that a *venom* has a certain therapeutic effect. However, most successful approved drugs derived from venoms make use of the activity of a single component within that venom, rather than the entire (crude) venom. As previously mentioned, venoms can be comprised of hundreds of unique components, each with a unique function and molecular target. We are in the early stages (in collaboration with the Holford lab at CUNY–Hunter College) of applying `VenomSeq` individually to purified samples of each of the peptides from the venom of a snail in the family Terebridae. The goal of this project will be twofold: (1) To demonstrate the use of `VenomSeq` to screen individual venom components rather than crude venoms, and (2) to determine *which* of these venom components actually exerts transcriptomic effects on human cells. Each of these questions provides opportunities to understand better how specific venoms can cause therapeutic changes in human cells.

Even though most existing venom-derived drugs consist of a single component, crude venoms in nature use the synergistic effects of multiple components to cause specific phenotypic effects [134]. Therefore, testing each venom component individually using the `VenomSeq` workflow might fail to capture all of the clinically beneficial activities demonstrated by the crude venom. A brute-force solution is to perform `VenomSeq` on all combinations of the isolated venom components, but doing so requires a massive number of experiments ($2^n - 1$, where n is the number of components in the venom). Therefore, it will be necessary to establish a protocol for prioritizing combinations of venom components. One potential solution is to fractionate the venom (i.e., using gel filtration) and perform `VenomSeq` on combinations of the fractions, but this will need to be tested. Alternatively, integrative systems biology techniques could be used to predict which components act synergistically, via similarity to structures with well-established activities.

3.3.7. Applying VenomSeq to other natural product classes

`VenomSeq` was, obviously, designed for the purpose of discovering therapeutic activities from venoms, but it could be feasibly extended to other types of natural products, including plant and bacterial metabolites, and immunologic components. Venoms provide a number of advantages and simplifying assumptions that were useful in designing the technology, but once `VenomSeq` becomes more proven it should be possible to relax these assumptions with some minor modifications to experimental protocol and data analysis. We foresee a few of these as the following:

- Venoms' targeted nature makes it easy to assume they will have some effect in animals;

other natural products may be inert.

- Venom components are intentionally delivered as a mixture; other natural product mixtures might only be easy to collect as a mixture, in spite of unrelated biological activities.
- Venoms are usually soluble in water, while other natural products often are not.
- Non-venom toxins may have less-targeted MoAs, disrupting biological systems indiscriminantly (e.g., by interrupting cell membranes regardless of cell type).
- The kinetics of non-venom natural products may be more subtle than venoms, which tend to have powerful binding and catalytic protperties.

3.3.8. Interpreting connectivity analysis validation results

In §3.2.4, we described the results of the connectivity analysis procedure applied to PLATE-Seq expression data from IMR-32 cells treated with 37 existing drugs that have known effects, many of which are members of Connectivity Map perturbagen classes (PCLs). Since `VenomSeq` uses an expression analysis technology that is different from the Connectivity Map's L1000 platform, as well as a cell line that is not present in the Connectivity Map reference dataset, this is crucial for establishing that one can discover meaningful associations between crude venoms and profiles in the reference data within the `VenomSeq` framework.

Overall, the findings of our analysis are congruent with those made by the Connectivity Map team in [234]. Specifically, PCLs that affect highly conserved, core cellular functions (such as HDAC inhibitors, mTOR inhibitors, and PPAR receptors) tend to form strong connectivities with members of the same class regardless of cell line. Therefore, associations discovered between crude venoms and these drug classes are likely "true associations", even when using IMR-32 cells in the analysis. Furthermore, by virtue of leveraging data corresponding to drugs with known effects, but using a new cell line and different assay technology, we have made the following novel findings:

- Although IMR-32 is not present in the reference dataset, similarities between IMR-32 and cell lines that *are* present in the reference data can be leveraged to select reference expression profiles that are more likely to reproduce true associations. For example, HA1E and A375 cells produce expression profiles that form reasonably strong connectivities between IMR-32 query signatures and members of the same drug classes.
- More cell lines need to be included in the Connectivity Map data in order to better understand correlation structures in cell-specific expression, as well as to better capture therapeutic associations that are specific to cell types underrepresented in current datasets.
- Similarly, continued effort should be devoted to adding new PCL annotations. Currently, only 12.3% of compound signatures in the reference dataset are annotated to at least one PCL, and some PCLs contain only a few signatures. A more rigorous definition of what specifically comprises a PCL would allow secondary research groups to contribute to this effort, ultimately improving the utility of the CMap data and increasing the sensitivity of the algorithms used to discover new putative therapeutic associations.

In spite of the large degree of corroborating evidence these results provide (e.g., every drug in our validation set produced a positive average effect on within-PCL connectivities versus corresponding null distributions), we cannot confidently predict that the associations discovered for crude venoms are true associations, rather than simply data artifacts. Although our confidence in the novel associations would be greatly improved by more PCL annotations to allow our analyses to attain greater statistical power, the ultimate test is to perform *in vitro* and (eventually) *in vivo* tests for these predicted therapeutic mechanisms of action. Aside from larger quantities of reference data against which to run the validation analyses, we also hope to employ other data science techniques involving network analysis and more advanced applications of master regulator analysis (see, e.g., §3.2.5) to further understand the dynamic interactions between cell types, gene expression, and perturbational signals that underly therapeutic processes.

3.4. Methods (VenomSeq)

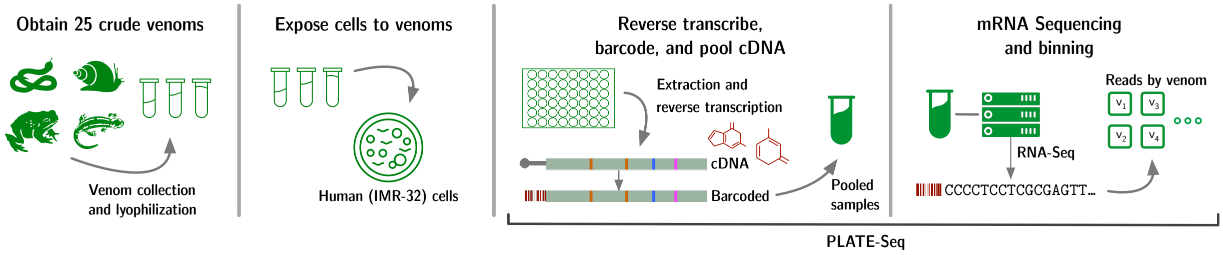


Figure 3.10.: RNA-Seq strategy for VenomSeq. Crude venoms are extracted and lyophilized. IMR-32 cells in culture are then treated with predetermined dosages of reconstituted venoms, and the PLATE-Seq method [32] is used to isolate, sequence, and count reads corresonding to cellular mRNA.

3.4.1. Reagents and materials

We performed growth inhibition assays and perturbation experiments using IMR-32 cells [200]—an adherent, metastatic neuroblastoma cell line used in previous applications of PLATE-Seq and VIPER—grown in Eagle's Minimum Essential Medium (EMEM) supplemented with fetal bovine serum. All venoms were provided in lyophilized form and stored at −20 °C. Since venoms naturally exist in aqueous solution, we reconstituted them in ddH$_2$O at ambient temperature.

3.4.2. Obtaining 25 venoms

VenomSeq is designed to apply to all venomous species across all taxonomic clades. Accordingly, we validated the workflow using 25 venoms sampled from a diverse range of species distributed across the tree of life. We selected the 25 species based on availability and compliance with international law, and sought to balance maximal cladistic diversity with minimal expected cytotoxicity (e.g., snakes in the genus *Bitis* are known for inducing tissue death and necrosis, and are therefore challenging to use for drug discovery applications [197]). We purchased the 25 venoms from Alpha Biotoxine in lyophilized form, and obtained prior approval from the US Centers for Disease Control (CDC) through the Federal Select Agent Program [83] for importing venoms containing α-conotoxins. The 25 venoms we selected are shown in **Table 3.8**. Note that we assigned a numeric identifier to each venom for convenience—these numbers show up numerous places in the data for VenomSeq. We also have included a rooted cladogram of the 25 species in **Figure 3.11**.

3.4.3. Growth inhibition assays

A major challenge in generating differential gene expression data for discovery purposes is finding appropriate dosages for the compounds being tested. This is done to ensure the compound is in sufficient concentration to be exerting an observable effect on the cells, while also mitigating processes that result from toxicity (e.g., apoptosis). In practice, determining an appropriate dosage concentration usually makes use of previous experimental evidence and/or biochemical constants, but since these are generally not available for crude venoms, we instead determined dosages based on growth inhibition.

We prepared 2-fold serial dilutions of each venom, using a starting concentration of

Table 3.8.: 25 venoms used to validate the `VenomSeq` workflow. Numbers in the right column are used as placeholder names for the venoms in data files.

Species name	Common name	Venom number
Naja nivea	Cape cobra	1
Laticauda colubrina	Banded sea krait	2
Montivipera xanthina	Ottoman viper	3
Dendroaspis polylepis polylepis	Black mamba	4
Crotalus scutulatus scutulatus	Mojave rattlesnake	5
Atractaspis sp.	Burrowing asp	6
Macrothele gigas	Japanese funnel web spider	7
Linothele fallax	Tiger spider	8
Poecilotheria fasciata	Sri Lanka ornamental spider	9
Argiope lobata	—	10
Synanceia verrucosa	Reef stonefish	11
Synanceia horrida	Estuarine stonefish	12
Buthus occitanus	Common yellow scorpion	13
Leiurus quinquestriatus	Deathstalker	14
Scorpio maurus	Large-clawed scorpion	15
Bufo bufo	Common toad	16
Rhinella marina	Cane toad	17
Bombina variegata	Yellow-bellied toad	18
Apis mellifera	Western honey bee	19
Vespa crabro	European hornet	20
Scolopendra subspinipes dehaani	Vietnamese centipede	21
Conus marmoreus	Marbled cone snail	22
Conus imperialis	Imperial cone snail	23
Octopus macropus	Atlantic white-spotted octopus	24
Pterois volitans	Red lionfish	25

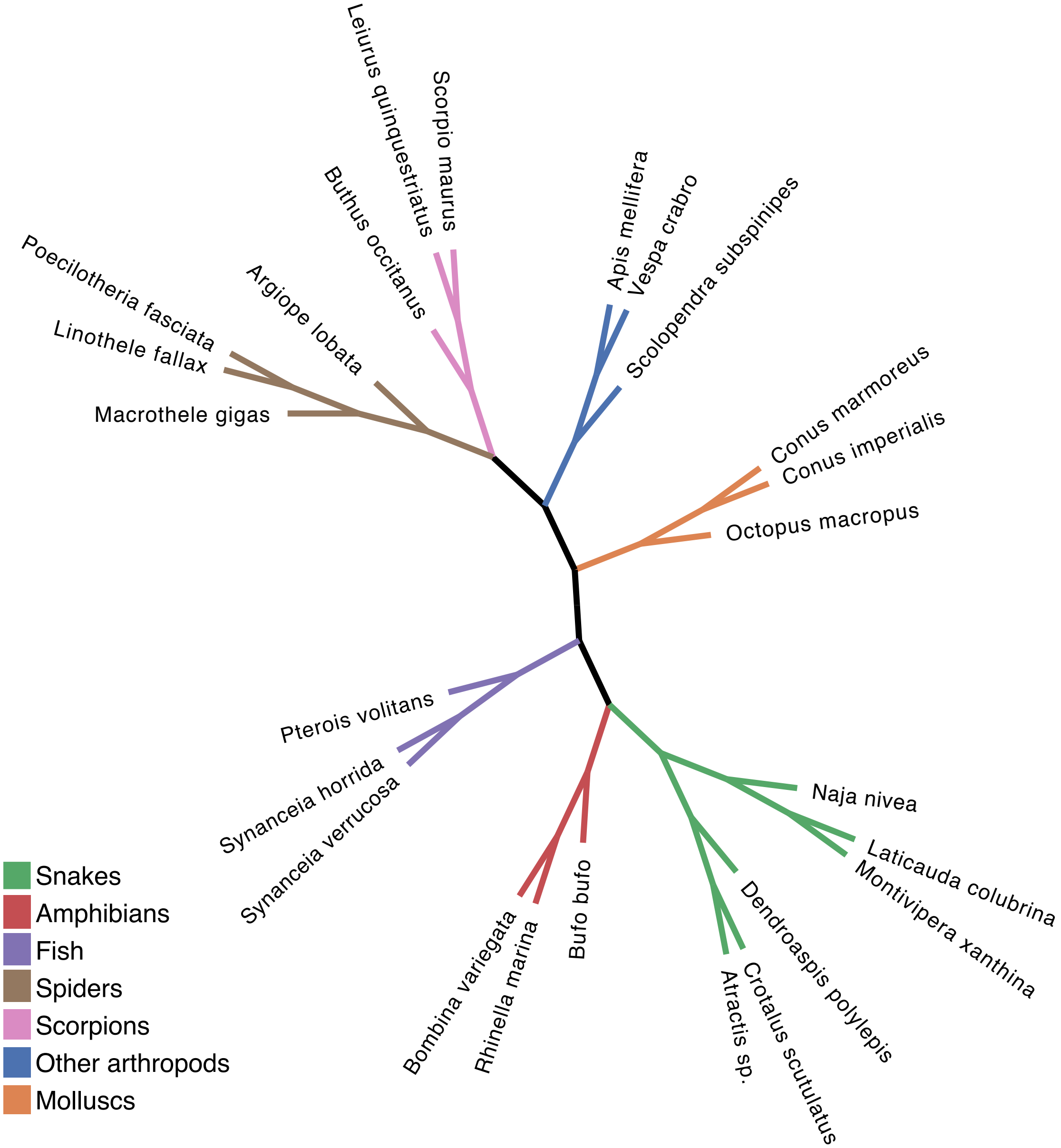


Figure 3.11.: Rooted cladogram showing the 25 species used in `VenomSeq`. Clades corresponding to major taxonomic groups are labeled as indicated.

$2.0\,\mathrm{mg}\,\mathrm{\mu l}^{-1}$. We seeded 96-well plates with IMR-32 cells and exposed them to the serial dilutions of the venoms after 24 hours of incubation. 48 hours after exposure, we quantified growth inhibition of the IMR-32 cells via cell viability luminesence assays.

For each venom, we fit these data to the Hill equation:

$$y = \text{Bottom} + \frac{(\text{Top} - \text{Bottom})}{1 + 10^{(\log \mathrm{GI}_{50} - x) \times h}}$$

where x is venom concentration, y is response (i.e., percent growth compared to untreated cells), Top and Bottom are the maximum and minimum values of y, respectively, and h is a constant that controls the shape of the sigmoidal curve. We used the resulting GI_{20} values (i.e., the value of x such that $y = 100\% - 20\% = 80\%$) as the venom exposure concentrations for the following sequencing experiments. Since some of the curves had very steep slopes (indicating rapid loss of total cell viability after miniscule changes in venom concentration), we confirmed the accuracy of the GI_{20} concentrations via secondary viability assays using the exact GI_{20} values extrapolated from the growth inhibition curves.

3.4.4. mRNA Sequencing

We prepared samples of human IMR-32 cells in 96-well cell culture plates, allowing for 3 replicates at each of 3 time points (6, 24, and 36 hours post-treatment) for each of the 25 venoms. The layout of the samples across 2 96-well plates is available in **Appendix A**. We reconstituted the crude venoms in water, and treated the samples with corresponding venoms at the previously determined GI_{20} values. We additionally prepared 12 control samples treated with water only, and 9 control samples that were untreated. Following total mRNA extraction, we carried out the PLATE-Seq protocol [32] to obtain gene counts for each sample. All sequencing was performed on the Illumina HiSeq platform. We used STAR [61] to (1) map the demultiplexed reads to the human genome (build GRCh38 [220]) and (2) count the reads mapping to known genes. For detailed quality control data for the sequencing experiments, refer to **Appendix A**.

3.4.5. Constructing expression signatures

We constructed differential gene expression signatures using the DESeq2 [146] library for the R programming language. DESeq2 fits observed counts for each gene to a negative binomial distribution with mean μ_{ij} and dispersion (variance) α_i, which we find to be a more robust model than traditional approaches based on the Poisson distribution (i.e., by allowing for unequal means and dispersions). In practice, users can substitute any method for determining significantly up- and down-regulated genes from count data. We filtered for genes with an FDR-corrected p-value < 0.05, and recorded their respective mean $\log_2$-fold change values, noting whether expression increased (up-regulated) or decreased (down-regulated).

3.4.6. Comparing venoms to known drugs and diseases

Comparing to known drugs using connectivity analysis

We retrieved the most recently published Connectivity Map dataset from the Clue.io Data Library (GSE92742), which contains 473,647 perturbational signatures, each consisting of robust Z-scores ("level 5 data" in the nomenclature of the Connectivity Map project) for

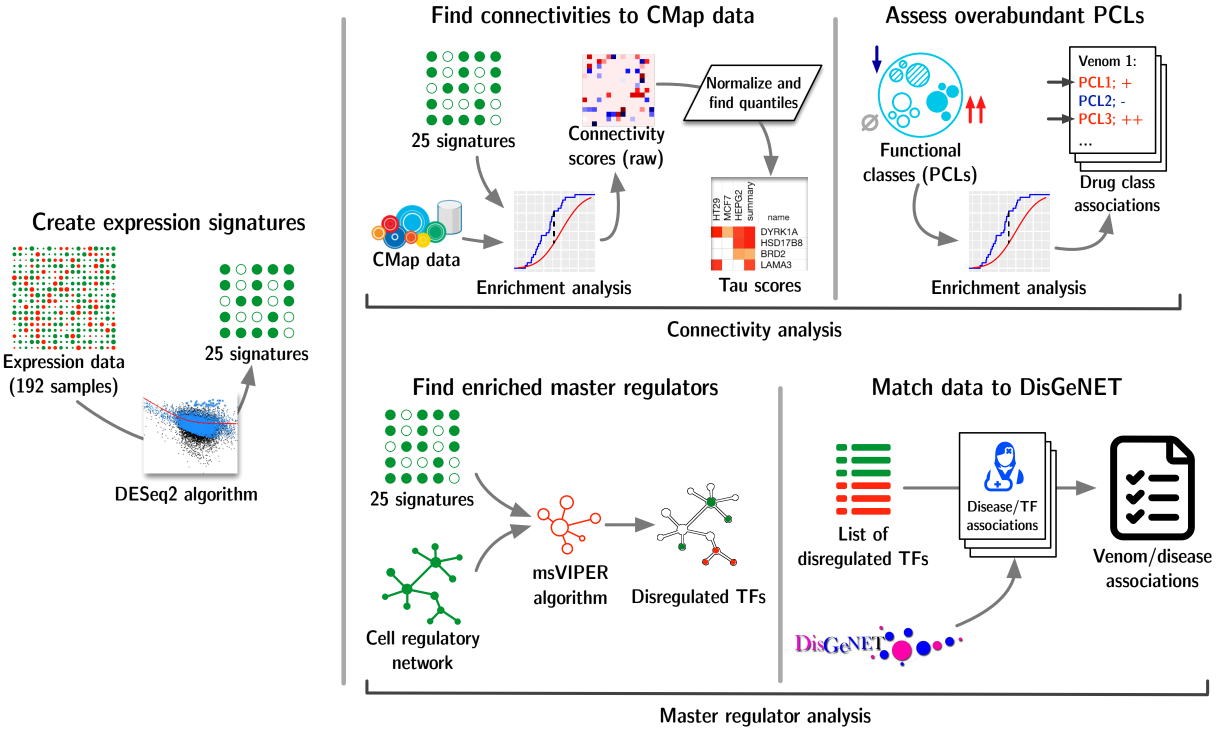


Figure 3.12.: Strategy for discovering new associations from `VenomSeq` data. After obtaining processed gene counts per sample, we generated differential expression signatures for each venom, and then used the signatures in two parallel analyses: connectivity analysis, and master regulator analysis.

12,328 genes, along with relevant metadata [234], including cell line annotations, dosages, time elapsed post-exposure, replicate numbers, and membership in manually-curated drug classes. We then used the procedure described by the Connectivity Map team [131] to generate connectivity scores between each of the `VenomSeq` gene expression signatures and each of the reference signatures in the Connectivity Map database. This procedure, adapted for `VenomSeq`, is summarized below.

Let a query q_i be the two lists of up- and down-regulated genes corresponding to the differential expression signature for venom i, and $\mathbf{r}_j \in \mathbf{R}$ be a vector of gene-wise Z-scores in reference signature j. We first generate a *Weighted Connectivity Score (*WCS*)* (or *Raw Connectivity Score*) between q_i and $\mathbf{r}_j$:

$$w_{qr} = \begin{cases} (ES^{q,r}_{\text{up}} - ES^{q,r}_{\text{down}})/2 & \text{if } \operatorname{sgn}(ES^{q,r}_{\text{up}}) \neq \operatorname{sgn}(ES^{q,r}_{\text{down}}) \\ 0 & \text{otherwise} \end{cases}$$

where sgn denotes the sign function $\frac{d}{dx}|x|$, and $ES^{qr}_{\cdot}$ is the signed enrichment score for either the up- or down-regulated genes in the signature, calculated separately (see below for details).

Although we validated `VenomSeq` on only a single human cell line, the reference database provided by the Connectivity Map provides expression profiles on 9 core cell lines, across multiple classes of perturbagens. Therefore, we compute normalized versions of WCS called Normalized Connectivity Scores (NCSs):

$$NCS_{q,r} = \begin{cases} w_{q,r}/\mu^{+}_{c,t} & \text{if } \operatorname{sgn}(w_{q,r}) > 0 \\ w_{q,r}/\mu^{-}_{c,t} & \text{otherwise} \end{cases}$$

where $\mu_{c,t}^{+}$ and $\mu_{c,t}^{-}$ are the means of all positive or negative WCSs (respectively) for the given cell line and perturbagen type.

The final step in computing connectivity scores between a venom q and a reference r is to convert $NCS_{q,r}$ into a value named τ, which represents the signed quantile score in the context of all positive or negative NCSs:

$$\tau_{q,r} = \operatorname{sgn}(NCS_{q,r}) \frac{100}{N} \sum_{i=1}^{N} [|NCS_{i,r}| < |NCS_{i,r}|]$$

where N is the number of all expression signatures in the reference database and $|NCS|$ is the absolute magnitude of an NCS.

Enrichment Score computation For a venom q and reference signature r, the enrichment score $ES_{\cdot}^{qr}$ is a signed Kolmogorov–Smirnov-like statistic indicating whether the subset of up- or down-regulated genes in q tend to occur towards the beginning or the end of a list of all genes ranked by expression level in r. We follow a procedure similar to that described by Lamb *et al.* in [131]. Specifically, we compute the following two values:

$$a = \max_{j=1}^{t} \left[\frac{j}{t} - \frac{\mathbf{V}_{qr}(j)}{n} \right]$$

$$b = \max_{j=1}^{t} \left[\frac{\mathbf{V}_{qr}(j)}{n} - \frac{(j-1)}{t} \right]$$

where $\mathbf{V}_{qr}$ is the vector of nonnegative integers that gives the indexes of the genes in q within the list of all genes ordered corresponding to their assumed values in r, t is the number of genes in q, and n is the number of genes reported in the reference database (in practice, $t \ll n$). We then set ES as follows:

$$ES^{qr}_{\cdot} = \begin{cases} a & \text{if } a > b \\ -b & \text{if } a < b \end{cases}$$

Since each query q consists of two lists—one of up-regulated and one of down-regulated genes—we compute both ES^{qr}_{up} and ES^{qr}_{down}, respectively, and use these two values to compute w_{qr}, as described above. For a more detailed, formalized description of the connectivity analysis algorithm, refer to **Appendix C.1**.

Comparing to known diseases using master regulator analysis

We discovered associations between the venom expression profiles and known diseases (coded as UMLS concept IDs) as the result of two sequential steps: (1) algorithmic determination of substantially perturbed cell regulatory modules (called *regulons*), and (2) mapping master regulators to diseases using high-confidence associations distributed in the DisGeNET database. These took as input the same differential expression data used in the connectivity analysis. IMR-32 regulon data (in the form of an adjacency matrix, where nodes are genes and edges are measures of mutual information with respect to their coexpression) were provided by the authors of the ARACNe algorithm.

In order to identify perturbed regulons, we first performed a 2-tailed Student's t-test between the genes' expression in the 'test' set (samples perturbed by venoms) and the 'reference' set (control samples). To make the final expression signatures, we then converted the results of the t-tests to Z-scores, to make them consistent with the models used by downstream algorithms. We generated null scores by performing the same test on the expression data with permuted sample labels, to account for correlation structures between genes. Once we had computed Z-scores, we ran the msVIPER algorithm, which derives enrichment statistics for each regulon based on the expression levels of the genes contained in the regulon. The result of msVIPER is a table of regulons (labeled by their master regulator), with enrichment scores, p-values, and FDR-corrected adjusted p-values.

We then compared the significantly upregulated regulons to the manually curated subset of TF–disease associations from DisGeNET. To do so, we mapped the statistically significant master regulator TFs for each venom to TFs reported in DisGeNET, and then mapped those TFs to their associated diseases. To help with filtering venom–disease associations with low evidence, we only retained diseases where *at least two* of the regulons that were significantly disregulated by the venom are associated with the same disease. Accordingly, we considered diseases with the highest number of significantly disregulated master regulators to comprise the associations with the greatest amount of evidence.

Similarly to how we mapped drugs to drug classes, we mapped diseases to disease categories. To do so, we identified the set of ICD-9 codes for each disease, based on the diseases' entries in the UMLS (UMLS CUIs were provided by DisGeNET). We then identified the disease category as the top-level ICD-9 'chapter' corresponding to that ICD-9 code (e.g., `NEOPLASMS`, `MENTAL DISORDERS`, `DISEASES OF THE RESPIRATORY SYSTEM`, etc.). In rare instances where a disease or condition was present in two locations (e.g., 'hypertension' is found in 2 chapters: `DISEASES OF THE CIRCULATORY SYSTEM` (401), and `INJURY AND POISONING` (997.91)), we opted for the more specific of the two (e.g., avoiding entries containing "not elsewhere classified").

3.4.7. Assessing sequencing technology and cell type compatibility

Since `VenomSeq` uses a sequencing technology (PLATE-Seq) and a cell line (IMR-32) that have not been used previously with the connectivity analysis approach, we evaluated their compatibility using a secondary dataset consisting of IMR-32 cells perturbed with 37 drugs and sequenced using PLATE-Seq. Since these drugs have known effects—and since many are present in the L1000 reference dataset—we sought to determine the extent to which connectivity analysis captures functional similarities between these drug data and the L1000 reference profiles. The 37 drugs are listed in **Table 3.5**. For the purposes of this discussion, a "query signature" is an expression signature corresponding to one of the 37 drugs in the validation dataset, and a "reference profile" is an L1000 expression profile from the dataset (GSE92742) published by the Connectivity Map team and used in the crude venom connectivity analysis.

Using these data (consisting of gene count matrices with several technical replicates per drug), we constructed differential expression signatures and performed the connectivity analysis algorithm in the same manner as we had for IMR-32 cells exposed to the 25 crude venoms. We annotated each of the 37 drugs (where possible) with perturbagen classes (PCLs) defined by the Connectivity Map team, which allowed us to identify L1000 expression profiles that come from the same drug classes as the drugs in our validation dataset. We then evaluated connectivity scores among members of the same PCL from two perspectives: (1) By aggregating all τ scores for reference signatures corresponding to a given compound, integrating evidence from all cell lines, and (2) by aggregating τ scores within individual cell lines, allowing us to assess the degrees to which specific cell lines are compatible with IMR-32/PLATE-Seq query signatures.

For the first of these two approaches, we collected all values of τ connecting query signatures in a PCL to reference profiles in the same PCL, and constructed null models by retrieving τ scores between the same query signature and all reference profiles that are members of any PCL. We defined the "effect size" of each PCL annotation as the difference of the mean of the scores within the true PCL and the mean of the scores in the null model. Additionally, we determined statistical significance using independent two-sample Student's t-tests. To correct for multiple testing, we adjusted p-values using the Benjamini-Hochberg procedure ($\alpha = 0.05$).

For the second approach—in which we evaluated each of the 9 core L1000 cell lines separately for each query signature—we retrieved τ scores between query signatures and each of the 92 PCLs in the reference dataset. Then, for each of the 9 cell lines and each of the query signatures annotated to a PCL, we constructed ordered lists of all PCLs ranked by their mean τ score in descending order (highest to lowest connectivity). In each of those lists, we determined the rank corresponding to the expected ("true") PCL—which we call the *rank percentiles*—and aggregated these ranks separately by (a) the drug corresponding to the query signature and (b) cell line of the reference profile. These two strategies allow us to separately assess the effects of *drugs* and *cell lines* on the behavior of connectivity scores. Under the null hypothesis that there is no selective preference for the true PCL in the connectivity data, the mean rank percentiles would follow a continuous uniform distribution in the range $[0, 1]$. Alternatively, if there is a selective preference for the expected PCL in the connectivity data, this rank will tend to occur towards the front of the list of ranks (and vice-versa).

Chapter 4.

Integrating and delivering venom knowledge using semantic data analysis

4.1. Structuring and representing `VenomSeq` data in VenomKB

`VenomSeq`—like each of the other components of this dissertation—was designed to be an open-access, publicly available resource that others can use to adapt to their own research needs. To remove ambiguity and encourage reproducibility, we have designed a data schema for representing the findings of `VenomSeq` experiments. This schema conforms to the JSON Schema standard, version `draft-06`[1]. For reference, the complete schema for `VenomSeq` data is listed in **Appendix B**.

As a result, the entire data contents of VenomKB can be serialized into a simple, intuitive JSON format. The raw data generated by PLATE-Seq (i.e., gene counts by sample) are not stored/distributed on VenomKB, but the schema includes links to these data on NCBI's Gene Expression Omnibus. We anticipate that users will prefer to interact with processed data files, particularly the expression signatures indicating which genes are differentially expressed on perturbation to each of the 25 venoms.

[1] available at `https://tools.ietf.org/html/draft-wright-json-schema-01`

The logical entity corresponding to a `VenomSeq` data object is a venom and the effects that venom exerts on a human cell; therefore, a single `VenomSeq` data object corresponds to all PLATE-Seq samples (and analyses derived from those samples) reporting perturbation by a single venom. In other words, a `VenomSeq` data object collapses replicates, time points, and human cell lines (of which we have only used 1, so far), but not venoms. The control samples (untreated and 'water only') are available to explore in the raw data files, and described in the metadata for the JSON data record. The top-level attributes of a `VenomSeq` JSON object are as follows:

`experiment-description`: String describing the context of the experimentation.

`investigators`: List of contributing author names and emails.

`release-date`: When the data were initially made available for public access.

`sequencing-platform`: String describing which platform was used to produce the data (e.g., "`Illumina HiSeq X`").

`cell-type`: Structured description of the human cell type(s) used.

`venom`: Structured description of the venom used.

`data`: Differential expression profile for the venom.

4.2. The VenomKB Semantic API

Ontologies are incredibly powerful tools that can be adapted to a wide variety of computational needs, including automation of tasks, standardization of data, and structuring artificial intelligence applications. In the context of VenomKB and `VenomSeq`, we are interested in using the Venom Ontology for the following tasks:

1. Providing a standardized way to interpret and structure venom data.
2. Enabling reproducibility of `VenomSeq` experiments, to encourage reuse and adaptation (e.g., as described in §4.1).
3. Retrieving data with a highly complex underlying structure when given *meaningful* queries that correspond to specific research questions.

Of these tasks, we have already described and illustrated uses for the first 2. The 3rd, on the other hand, is interesting and different for a number of reasons. First, it describes a use for bioontologies that is of interest to both the knowledge engineering and information retrieval communities. Additionally, it provides an opportunity to improve accessibility to the knowledge contained in VenomKB (and the knowledge generated using `VenomSeq`) for the broader biomedical research community, by removing the need for users to interact with the complex underlying data structures that comprise the knowledge base. Most users would benefit greatly from the implementation of software tools that perform these data manipulations automatically, and in a way that adapts to the changing data needs of the user.

Here, we introduce a new concept which we refer to as the *Semantic API*. Most existing APIs are capable of simple tasks (e.g., fetching data using a unique identifier) as well as more complex tasks (e.g., searching a database using queries containing filters, pattern matching, and secondary data manipulation subroutines), but they almost always are constrained to operations that have well-defined behavior, which makes the API easier to validate and integrate into automated workflows, but comes at the cost of limited flexibility and generalizability. Furthermore, as traditional APIs grow to allow increasingly complex queries they have a tendency to impose more rigorous demands on the user, requiring knowledge of intricate data models. This is prohibitive to many classes of users, particularly those with limited computational expertise.

The Semantic API is an auxiliary server-side application bundle that interfaces with VenomKB's existing REST API. The Semantic API's structure consists of a graph representation of the Venom Ontology (stored in a Neo4j graph database [254]) alongside a JavaScript application that carries out translation between the user and the graph database. We will take a detailed look at the individual components that make up the Semantic API in §4.2.3, but first will describe the specific advantages that make it an important step forward for information retrieval and artificial intelligence.

4.2.1. Related work and advantages over existing tools

The concept of a semantic API draws upon several decades of research conducted in the fields of biomedical semantics and ontology design. The utility of ontologies and structured terminologies for representing, integrating, and delivering knowledge for both human and computer consumption is already well-established [26, 28, 155]. Unsurprisingly, this has contributed to major advances in related areas like natural language processing (NLP), where human-provided natural language (from sources such as news articles, web search engine queries, and free-text medical notes) is parsed and translated into a structured, formal representation that can be reused by computers [31, 159]. "Rule-based" NLP is a technique that uses a predefined grammar and structured knowledge resources (e.g., lexicons, terminologies, or ontologies) to parse user-submitted text[2]. Rule-based NLP has been especially effective in domains with distinctive vocabularies and styles for sentence composition, in-

[2] The other major approach to NLP is *probabilistic* NLP, which instead interprets text using mathematical models, typically via machine learning.

cluding most areas of biomedicine. Some examples of rule-based NLP tools for biomedicine include MedLEE/BioMedLEE [76, 148], which uses context-free grammars and domain lexicons for pattern-matching on biomedical text, and Apache cTAKES [218], which reuses the lexical analysis tools contained in the National Library of Medicine's UMLS to parse text.

Previous work has also been conducted on designing programming interfaces that can communicate with ontologies. The OWL API, for example, is a programmatic interface for performing high-level access and manipulation capabilities to OWL ontologies, designed mainly as a resource for implementing semantic web resources [16]. Also developed for the semantic web, the Jena toolkit [165]—which we used to populate individuals in the Venom Ontology—is a Java API for manipulating files in the RDF format, which is one of several file formats compatible with the OWL standard [165]. Arguably the most similar body of work preceding our semantic API is described by Koutsomitropoulos *et al* [127], where they designed their own semantic API with a semantic querying interface to solve similar problems to ours—namely, that querying semantic resources requires a deep understanding of data models, querying protocols, and other enigmatic peculiarities of knowledge engineering. Their proposed tool is an interface that allows users to design an entailment-based query for Semantic Web resources, and then translates the query to an equivalent SPARQL query (which is directly compatible with most Semantic Web applications). Unfortunately, we find that their new semantic query syntax still remains esoteric and out-of-reach for most potential users. Furthermore, the utility is designed primarily as a proof-of-concept rather than a tool to support ongoing research in a certain domain (such as our use of a semantic API for toxinology).

Our Semantic API builds on these projects' underlying goals in various ways. As far as we can tell, our semantic API is one of only a few attempts to create an API that not only interfaces with ontologies, but uses the ontologies to interpret the *implied meaning* of user queries, filling in gaps in the information content of the query using the assertions of the ontology itself, and the *only* attempt that truly attempts to accomodate non-expert users. It also represents a departure from the recent trend (probably related to interest in the semantic web) of designing ontology-based tools with increasingly general domains of ontological commitment [23, 71, 95]. Although the Semantic API can be generalized to virtually any domain of knowledge, its power is largely derived from focusing on a single specific domain (in our case, toxinology), which substantially augments its inferential capabilities. A final distinction of note is how it is intended to be used—most existing tools that interface with ontologies are highly infrastructural, and are not intended for direct use by consumers. The Semantic API is first and foremost a utility meant to be queried directly (either by writing queries by hand or through a graphical form-based interface), although it is also completely capable of consuming automatically-generated HTTP queries submitted by other web services.

4.2.2. A theoretical framework for the semantic API

Before we can describe how this Semantic API can be implemented computationally, we need to define the theoretical task being performed. An OWL ontology can be abstracted as two unweighted *directed graphs* (digraphs) containing the ontology's class hierarchy and the individuals that comprise the ontology—which we name C and I—respectively. C consists of the vertex set V_C (ontology classes) and the edge set E_C (relationships between classes), and I consists of the vertex set V_I (individuals) and the edge set E_I (relationships between individuals). We know these two graphs have the following properties:

- Each individual in V_I is an instance of a single class in V_O (inherited class membership can be ignored for now).
- If a pair of individuals $\{v_{I_1}, v_{I_2}\} \subseteq V_I$ are joined by a relationship $e_{I_1} \subseteq E_I$, an identical relationship $e_{C_1} \subseteq E_C$ must join the classes $\{v_{C_1}, v_{C_2}\} \subseteq V_C$ of which those 2 individuals are members.
- C must be at least weakly connected[3], but I can be disconnected. Accordingly, the larger graph implied by joining all individuals in I to their respective classes in C is also at least weakly connected.

Additionally, each edge and vertex in I can have an arbitrary number of metadata tags, known in the ontology engineering community as *data properties*. Although a complete ontology specifies many other components (such as domains and ranges, relationship types, and additional rules and restrictions), we can ignore these for the time being.

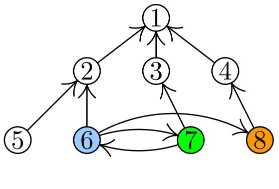
The main task of the Semantic API is to find each of the smallest spanning trees[4] $(s_1, \ldots, s_n) = S(J)$ of a subgraph $J \subseteq I$ that satisfy the following:

- J contains at least one individual from each of a set of ontology classes $G \subseteq V_C$ that are determined to be 'relevant' classes based on the semantic content of a user's query to the API, and no individuals from 'irrelevant' ontology classes. The way we determine which classes are relevant is described in §4.2.3.
- The tree satisfies each of an optional set of filters (which we call *constraints*) provided in the query.

For example, suppose a simplified ontology is represented by the graphs

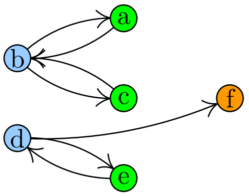
[3] I.e., if you replace all directed edges with undirected edges, the graph will be connected.

[4] A *spanning tree* of a graph G is any subgraph T that covers all vertices of G and contains no cycles. A *smallest spanning tree* is the spanning tree(s) containing the fewest possible number of edges.



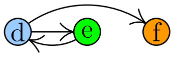
C

and



I

and that colored nodes indicate class membership. A user submits a query that specifies classes 7 and 8 as relevant ($G = \{7, 8\}$). The only subgraph J that satisfies the first of these two criteria is



J .

If the members of J satisfy all of the metadata constraints in the query, J is considered a match, and is returned as input to the aggregation subroutines of the semantic API for further manipulation, as needed.

4.2.3. Semantic API implementation

A simplified schematic of VenomKB's semantic API is shown in **Figure 4.1** (for a more detailed version, refer instead to **Figure 4.2**). There are two components to the semantic api: (1) the query engine, and (2) the graph database. When a user submits a JSON-formatted `POST` request to VenomKB's API containing the base URL `semantic/`, the request's body is passed to a script named `Query.js`. This script interprets the user query and translates it into the Cypher graph database query language. This Cypher query is then executed on the Neo4j graph database instance, and the results are returned to `Query.js`. The script then executes any remaining aggregations (such as sorting, counting, formatting, and other

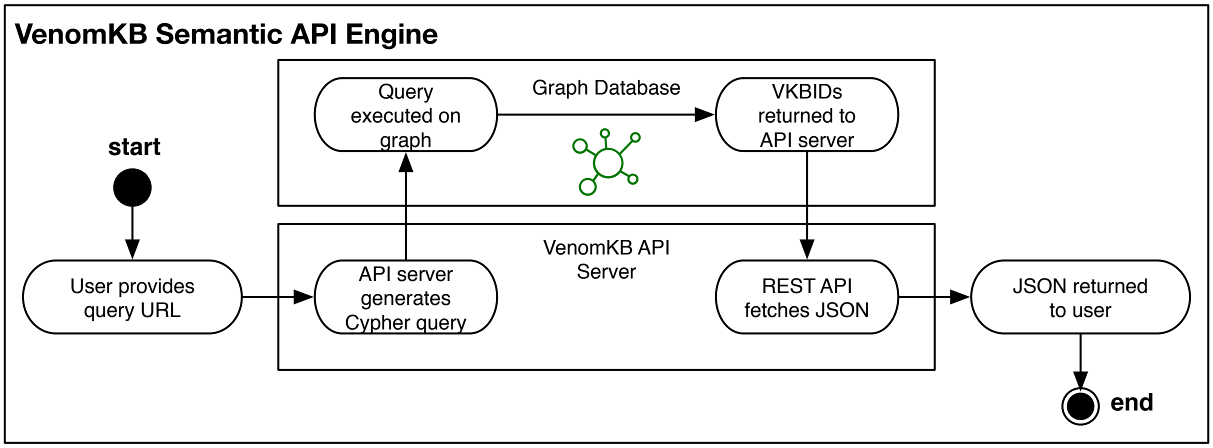


Figure 4.1.: Simplified schematic of VenomKB's semantic API. Note that the 'standard' (REST) API accepts a semantic query, but much of the computation is performed on a separate graph database derived from the Venom Ontology. After running the graph query, the results are translated into a JSON response and returned to the user. For a more detailed view of the algorithm, see **Figure 4.2**.

fundamental operations not performed by the graph database server). We will now consider each of these operations in detail.

The semantic query format

Each call to the Semantic API is called a *semantic query*. A semantic query is structured as a JSON object and included in the body of an HTTP POST request sent to the Semantic API's URL endpoint (for VenomKB's Semantic API, this is `http://www.venomkb.org/api/semantic/`). Semantic queries are structured according to the following template:

```
{
    "select": "[ONTOLOGY CLASS]",
    "declare": {
        "[ONTOLOGY CLASS]": [
            {
                "attribute": "[ATTRIBUTE]",
                "operator": "[OPERATOR]",
                "value": "[VALUE]"
            },
            ...
        ],
        ...
    },
    "aggregate": {
        "[AGGREGATION]": {
            ...
        },
        ...
    }
}
```

Summarized, this consists of up to 3 fields: `select` (required), `declare` (optional), and `aggregate` (also optional). In the template, strings in square brackets indicate variable fields that are filled in by the user, and ellipses ("...") indicate where multiple fields can be chained together (e.g., the user can `declare` constraints for 0 or more ontology classes, and each of those classes can have 1 or more constraints). The details for each of these three fields are explained in more detail below.

Extracting the result type ("`select`")

The semantic API is designed to be far more flexible than most traditional APIs. Many different queries can be posed to obtain the same conceptual result, so it is important that the API's logic can reflect this. Accordingly, the application needs to execute a number of normalization procedures to interpret the question being represented by a specific query. The first of these tasks is to identify the ontology class(es) of the desired result. For example, if the user submits a query to identify the species that produce venom proteins with the word "phospholipase" in their name, the desired ontology class is `Species`. This is the one portion of the query that is unambiguously provided by the user, so the semantic API simply needs to validate the query field named `select`, labeled as such in order to reflect its similarity to the more familiar `SELECT` clauses used in all major variants of SQL.

Collecting constraints (“`declare`”)

The software then reads the `declare` block to identify filters that reduce the complete graph representation of the API down to a subgraph consisting of the data relevant to the query. We call these *constraints*, because they can be thought of as constraints that are applied to the subgraph(s) matching a user query. Constraints take the following format:

$$\langle \textit{class} \rangle \mid \langle \textit{attribute} \rangle \mid \langle \textit{operator} \rangle \mid \langle \textit{value} \rangle$$

where ⟨*class*⟩ is an ontology class, ⟨*attribute*⟩ is an attribute defined for that class in the VenomKB data schema, ⟨*operator*⟩ is any valid comparison operator (e.g., `=`, `>=`, `CONTAINS`...), and ⟨*value*⟩ is a string or number.

Finding relevant ontology classes

For a certain query, the relevant ontology classes are simply the set of classes that are mentioned at least once in either the `select` or `declare` blocks of the JSON query. In other words, they are all classes that either (a.) contain individuals of the type desired in the response to the query, or (b.) contain individuals that are tested against the constraints (filters) in the query.

Finding a shortest path on the ontology class structure

At this point, the semantic query engine begins the process of interpreting the semantic content of the user's query. The first task in doing so is to find the smallest subgraph C' that contains relevant ontology classes (i.e., as identified in §4.2.3) within the graph C containing *all* ontology classes. The query we will be passing to the graph database server needs to unambiguously specify a pattern to match against the members of the database in a format representing a chain of natural language predications, and we therefore need to know the chain of edges and vertices that links the relevant ontology classes. Finding the smallest subgraph is not strictly necessary, but it helps to ensure efficiency and improve early error detection.

Finding the smallest spanning subtree is, unfortunately, a nontrivial task—the most cutting-edge algorithms run in $\mathcal{O}(m \log n)$ time [191], and for some graph types it is an NP-hard problem [112]. Furthermore, combinatorial algorithms need to be used judiciously for production-quality applications, where a minor drop in algorithmic efficiency is compounded both by the number of user requests as well as the size of the underlying database being searched (both of which we hope will expand steadily as VenomKB develops a larger user base).

Generating a Cypher query

We now have all of the components we need to construct a Cypher query to pass to the Neo4j database server. Each request we construct consists of at least two mandatory clauses (delimited by whitespace or a newline character): `MATCH` and `RETURN`. The `MATCH` clause consists of a symbolic representation of the shortest path identified in the previous step:

```
MATCH (c1:Class1)-[:REL_1]->...-[:REL_N-1]->(cn:ClassN)
```

(where the appropriate class and relationship names are used instead of placeholders). When only one ontology class is relevant (e.g., the query is a simple search on a single datatype), this reduces to `MATCH (c1:Class1)`. Note that we assign variable names (e.g., the `c1` in `(c1:Class1)`)—these variables are used in subsequent clauses to refer to nodes that match the pattern specified in `MATCH`.

If the user has provided constraints, the next clause is `WHERE`. Since constraints are optional, if they are not present, the cypher query contains no `WHERE` clause. Otherwise, the semantic query engine iterates over the array of constraints, converting them into the format

`WHERE c1.`⟨*attribute 1*⟩ ⟨ *operator*⟩ ⟨*value*⟩ `AND c2.`⟨*attribute 2*⟩ . . .

and so on, depending on how many constraints are given.

Finally, we build the `RETURN` clause. This can simply be a variable name that corresponds to the ontology class in the `return` block of the original query:

```
RETURN c1
```

Certain aggregations will also be applied here when the graph database is especially efficient. These aggregations are `sort` and `unique`, each of which has an equivalent reserved keyword built into the Cypher language (`ORDER BY` and `DISTINCT`, respectively).

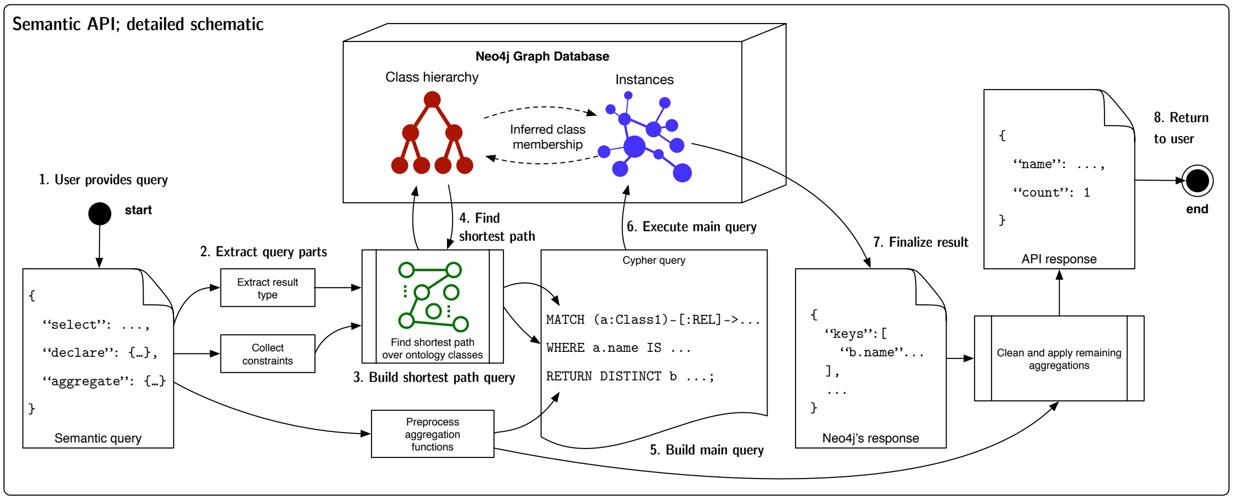
Executing the query on the Neo4j graph database

The semantic query engine—which is written in JavaScript and runs on the Node.js runtime environment, like the rest of VenomKB—communicates with Neo4j using the Bolt driver, which is maintained and provided by the authors of Neo4j. Once the query engine has constructed a string containing a properly formatted Cypher query, it initiates the query as a new transaction with the Bolt driver, and uses the "async/await" pattern to receive the graph database's response (allowing concurrent execution of multiple simultaneous queries).

Aggregating results ("`aggregate`")

Aggregations are usually defined as functions that accept a list of objects (e.g., rows in a database, or nodes in a graph) and produce a single summary value describing that list. For the semantic API, we adopt a slightly more permissive defintion: A semantic API aggregation is any function that can be used to manipulate the subgraph returned by Neo4j. This includes single summary values (like in the more strict definition) as well as transformed versions of the entire list. As mentioned previously, aggregations can be applied either on the results of the graph database query, or can be embedded into the graph database query itself. This choice is predetermined for each aggregation function. Aside from the aggregations described above (see *Generating a Cypher query*), the rest run on the semantic query engine. For example, the `count` aggregation is more efficient (in both time and memory) when performed by Node.js rather than by the graph database server.

Aggregation functions are modular, and a developer familiar with basic JavaScript and the code structure of the semantic API can easily add new aggregations (e.g., an aggregation that searches for the minimum value of a certain attribute can be implemented in just a few lines of code). Future work on the semantic API will build on this modularity by providing simpler ways of including these functions that do not require modification of the query engine itself.



```
{
  "select": ...,
  "declare": {…},
  "aggregate": {…}
}
```

```
MATCH (a:Class1)-[:REL]->...
WHERE a.name IS ...
RETURN DISTINCT b ...;
```

```
{
  "keys":[
    "b.name"...
  ],
  ...
}
```

```
{
  "name": ...,
  "count": 1
}
```

Figure 4.2.: Detailed schematic of the Semantic API query process. Individual ordered steps in the algorithm are numbered and labeled in bold.

4.2.4. The benefits of a semantic API

Catering to users with varying levels of technical skill

To enable use of the semantic API by bench researchers and laypersons, we have designed a graphical query builder incorporated into the VenomKB website. The interface for this tool is shown in **Figure 4.3**. Users complete a form, where one segment is devoted to each of the three sections of a semantic query (`select`, `declare`, and `aggregate`). Since `declare` and `aggregate` can each consist of zero or more statements, the user can add or remove any number of those statements, as desired. We designed the wording for each section of the interface carefully, to clearly represent the semantic meaning of each section and illustrate to the user how to “tell” the API what they are looking to retrieve.

Reducing time, effort, and errors

Even for experienced users of a given data schema, aggregating items across multiple complex types with various constraints is a tedious and time-consuming task, usually requiring many lines of code and a nontrivial amount of debugging. Arguably, using the semantic API (either by writing a semantic query manually or by defining one using the form-based interface on the VenomKB website) takes only a fraction of the time. Furthermore, the form of a semantic query is designed to reflect the intuitive data-needs of the user, and therefore theoretically requires less effort to translate that need into a format that the software can successfully interpret. The semantic API also has the potential to reduce user errors in data retrieval—an issue that is pervasive in database use and causes downstream problems that are sometimes impossible to detect [228]. This advantage is largely the result of two factors: (1) The user only interacts with the API a single time, whereas with traditional database appplications they may have to perform several successive queries to retrieve the desired result, and (2) the ontology itself acts as a safeguard against errors, preventing incompatible data types from being “joined” and by unambiguously specifying exactly which relationships exist between related data elements. Due to the current implementation of the semantic API, this is a passive process—we extract data from the ontology to populate the graph database, and all of the previously defined restrictions within the ontology dictate the structure of the graph.

Facilitating integrative data translation

Projects like the Biomedical Data Translator [46] and the CDISC standards [128] aim—among other things—to allow disparate systems that conduct transactions on biomedical data to be able to communicate with one another unambiguously and dynamically [9]. The Semantic API is capable of addressing this same goal. Since the format of a semantic query is not tied in any way to the domain of application, semantic queries can be constructed identically for every Semantic API that follows the format we defined. An application that is designed to communicate with a Semantic API should be able to communicate with *any* Semantic API, and therefore should also be able to integrate data from multiple Semantic APIs. One major caveat that needs to be addressed is how an instance of a Semantic API will inform remote systems about the domain content it serves. This could take the format of something as simple as an XML manifest file, or as complex as a complete distribution of the OWL ontology on which the Semantic API is running. Another (major) caveat is that this functionality depends entirely on adoption and implementation of the Semantic API as a generalizable standard by the larger research community.

4.2.5. Complete example

Consider this meaningful question, which a user may wish to answer using the semantic API:

> *Which venoms are indicated in treating osteosarcoma, and how many proteins in those venoms are known to demonstrate this indication?*

One way of writing a corresponding query is as follows:

```
example: {
  "select": [{"Species": "name"}, "Protein"],
  "declare": {
    "SystemicEffect": [
      {
        "attribute": "name",
        "operator": "equals",
        "value": "Osteosarcoma"
      }
    ]
  },
  "aggregate": {
    "distinct": {
      "attribute": "name",
      "class": "Species"
    },
    "count": {
      "class": "Protein"
    }
  }
}
```

The semantic API generates the following Cypher query and executes it on the Neo4j graph database:

```
MATCH(s:Species)-[:SPECIES_HAS_PROTEIN]->(p:Protein) \
    -[:INFLUENCES_SYSTEMIC_EFFECT]->(e:SystemicEffect)
WHERE e.name = 'Osteosarcoma'
RETURN DISTINCT s.name, p
```

Neo4j responds with the following:

```
{
  "keys": [
    "s.name",
    "p"
  ],
  "length": 2,
  "_fields": [
    "Crotalus viridis viridis",
    {
      "identity": {
        "low": 177224,
        "high": 0
      },
      "labels": [
        "Protein"
      ],
      "properties": {
        "name": "Zinc metalloproteinase-disintegrin-like crovidisin"
           ,
        "vkbid": "P4722144",
        "aa_sequence": "AMVTKNNGDLDKSGTECRLYCKDNSPGQNNS..."
        "score": {
          "low": 2,
          "high": 0
        },
        "UniProtKB_id": "P0C7N3"
      }
    }
  ],
  "_fieldLookup": {
    "s.name": 0,
    "p": 1
  }
}
```

Finally, the semantic API parses these data into a simplified JSON object, and applies the `count` aggregation on the object labeled `p`, yielding:

```
[
    {
        "name": "Crotalus viridis viridis"
    },
    {
        "count": 1
    }
]
```

Clearly, this response indicates that one species in VenomKB (*Crotalus viridis viridis*) contains venom that has been shown to treat osteosarcoma, and 1 protein in its venom is known to exhibit that activity. By slightly tweaking the `select` clause in the query, the user can determine the name of that same protein.

4.2.6. Task-based evaluation of the semantic API for computational toxinology

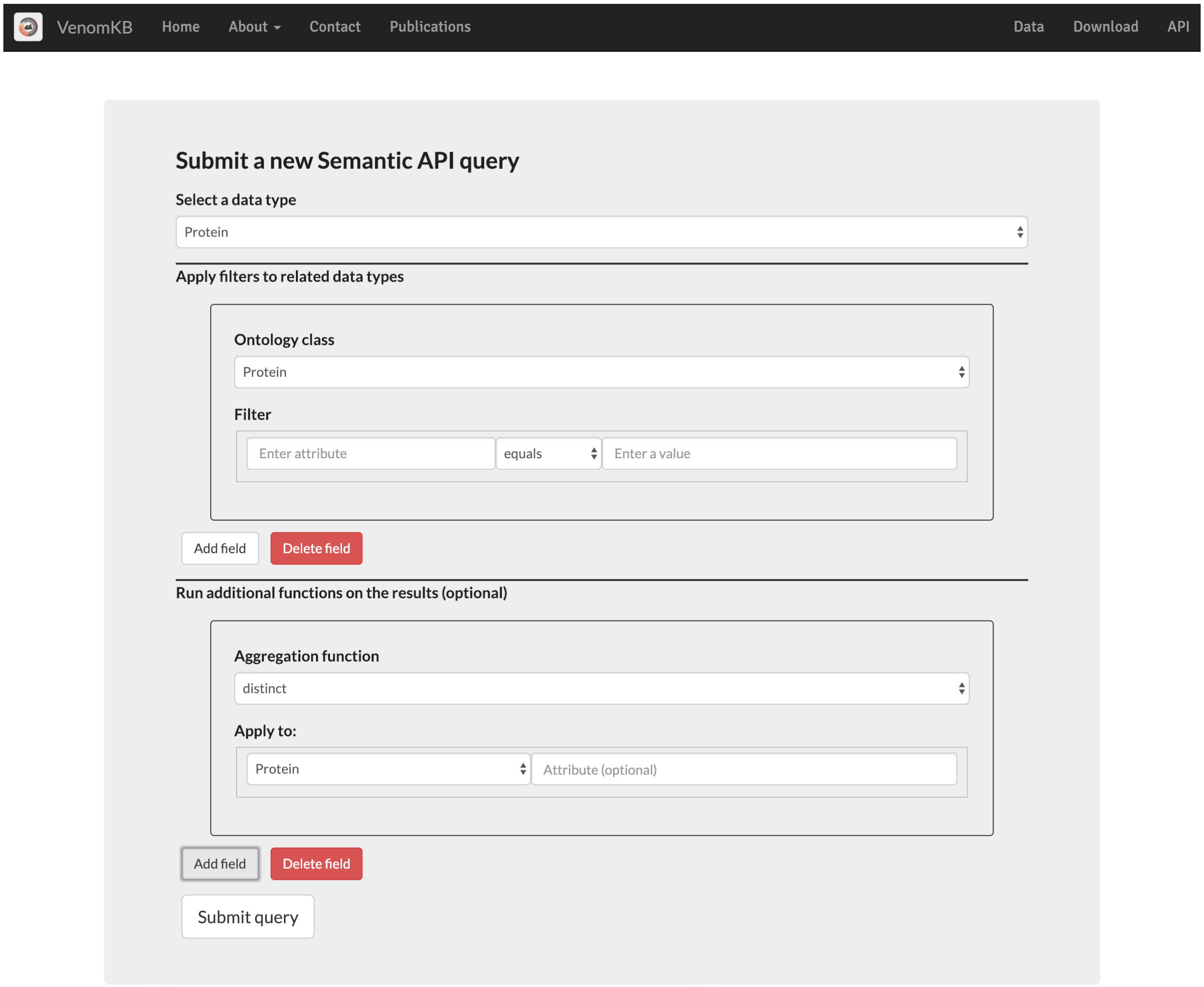
One of the major goals of computational toxinology is to make advanced computational techniques accessible to non-informaticians, but research groups comprised mainly of molecular biologists and/or field biologists often do not have members with an advanced understanding of how to manipulate complex data models[5]. Even for users who do have this type of expertise, many venom-associated questions with relatively simple semantics can require a great deal of work to answer effectively. Furthermore, issues with venom nomenclature, conflicting and nonstandardized entries in existing public databases, and a paucity of molecular data make computer-aided large scale data retrieval and analysis virtually impossible in most contexts concerning venoms. Given the need for reproducibility in modern biomedical science [158, 253] (particularly when the end-goal is the discovery and development of new drugs to treat human diseases), it is crucial for the toxinology community to have a "common ground" for conducting research. The semantic API was designed with the intention of solving these issues related to the accessibility and normalization of venom data.

To aid advanced users in learning the structure and format of a semantic query—and how to translate a meaningful research question into a query that retrieves the correct result(s)—we have provided a gallery of diverse questions and their JSON representation, as well as the result given by the semantic API. For example, consider the following question:

Do any proteins from cone snail (Conus) venom treat neuralgia?

One way to write this as a semantic query would be

[5] Conversely, computational labs typically don't employ members who know how to appropriately handle reagents and 'wet lab' experimental protocols.



Submit a new Semantic API query

Select a data type: Protein

Apply filters to related data types

Ontology class: Protein

Filter: Enter attribute | equals | Enter a value

Run additional functions on the results (optional)

Aggregation function: distinct

Apply to: Protein | Attribute (optional)

Figure 4.3.: Graphical user interface for building Semantic API queries.

```
{
    "select": "Species",
    "declare": {
        "Species": [
            {
                "attribute": "name",
                "operator": "contains",
                "value": "Conus"
            }
        ],
        "SystemicEffect": [
            {
                "attribute": "name",
                "operator": "equals",
                "value": "Neuralgia"
            }
        ]
    },
    "aggregate": {
        "exists": true
    }
}
```

In other words, the user is looking for species in VenomKB that satisfy two criteria: (1) Their name contains the string `Conus`, and (2) they are linked to a `SystemicEffect` with the name `Neuralgia`. The aggregation `exists` returns the boolean value “true” if any matches are found[6], rather than returning a list of the species. Other examples are provided in the source repository for VenomKB.

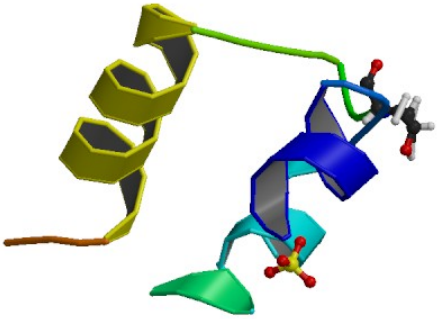
Since the semantic API is meant to solve a new type of need, the best way to evaluate its success and utility is through specific case studies. Here, we provide narrative walkthroughs of several such case studies, including examples of (1) retrieving known individual associations, (2) aggregating multiple fragmented associations to provide a “bigger picture” of the

[6] At the time of writing, this query does evaluate as “true” when submitted to the semantic API.

nature of venoms and their therapeutic effects, and (3) providing context and plausibility to novel associations suggested in the analysis of `VenomSeq` data.

Insulins in cone snail venom

Conus geographus



C. geographus peptide Con-Ins G1[7]

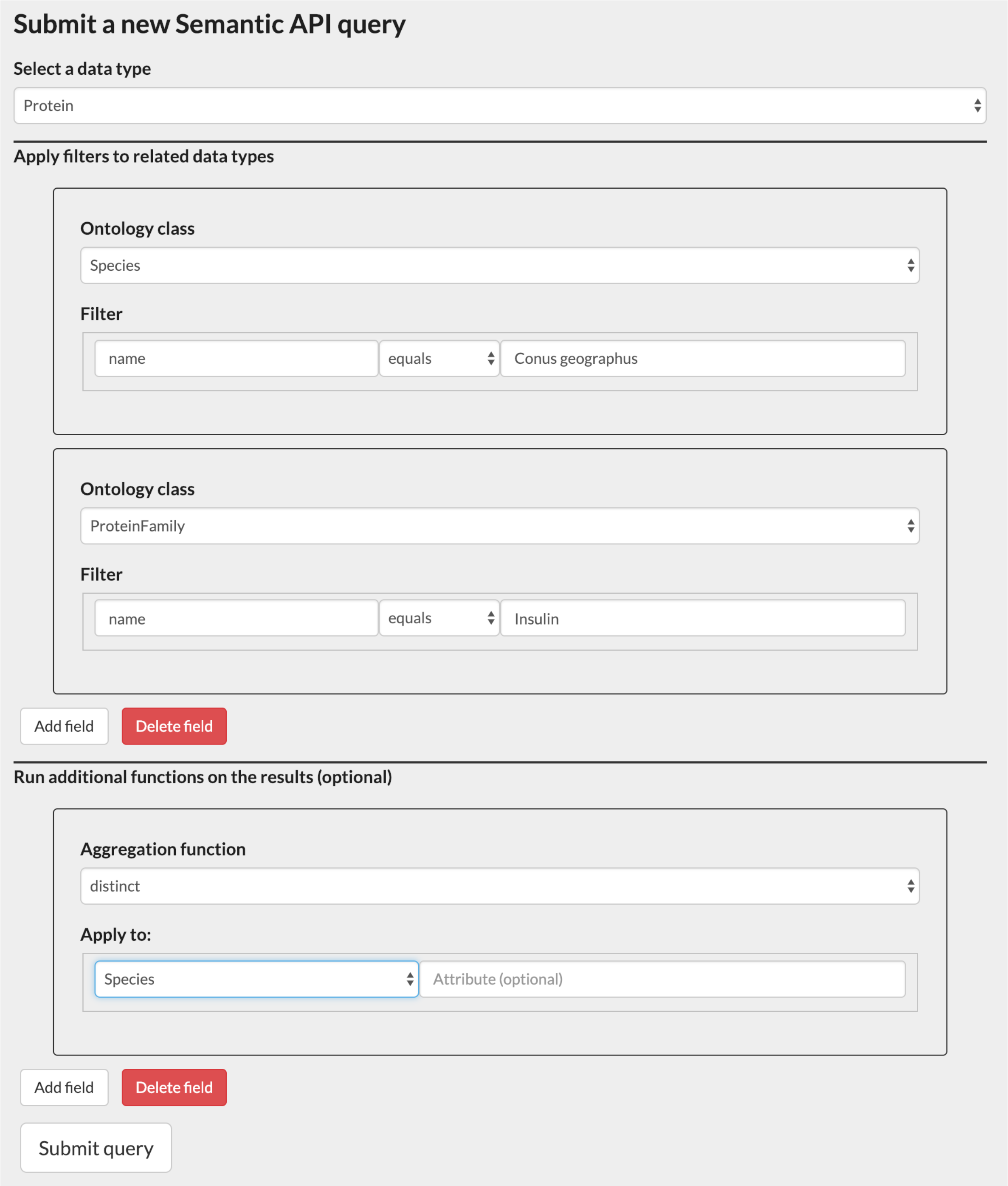
In 2016, Menting *et al* reported the groundbreaking discovery that a small insulin peptide purified from the venom of *C. geographus* contains a human insulin receptor binding motif, and demonstrated that it strongly mimics the signaling functions of human insulin within humanized mouse cells [168]. This paper was preceded by the discovery by Safavi-Hemami *et al* (including many of the same coauthors) that *C. geographus* releases the insulin into the water in order to "stun" prey fish by inducing hyperglycemic shock [216], along with evidence that similar insulin-like peptides were present in a number of other *Conus* species.

[7] PDB ID: 5JYQ [168]

Its size, stability, and potency make it an excellent candidate for developing better insulin treatments for patients with diabetes. Being a substantial recent finding in the toxinology community, we tested whether this association could be easily retrieved with the Semantic API:

```
{
    "select": "Species",
    "declare": {
        "Species": [
            {
                "attribute": "name",
                "operator": "equals",
                "value": "Conus geographus"
            }
        ],
        "Pfam": [
            {
                "attribute": "name",
                "operator": "equals",
                "value": "Insulin"
            }
        ]
    },
    "aggregate": {
        "distinct": {
            "class": "Species"
        }
    }
}
```

The semantic API returns 6 *C. geographus* peptides from VenomKB that are annotated as members of the "Insulin" family: Con-Ins G1b, G1, G1c, G3, G3b, and G2b. We can then explore these proteins individually, using either the web application or the REST API (e.g., `venomkb.org/api/proteins/P7637538` for Con-Ins G1b). For comparison, **Figure 4.5** shows the same query constructed using the graphical query builder interface.



Submit a new Semantic API query

Select a data type

Protein

Apply filters to related data types

Ontology class

Species

Filter

name | equals | Conus geographus

Ontology class

ProteinFamily

Filter

name | equals | Insulin

Run additional functions on the results (optional)

Aggregation function

distinct

Apply to:

Species | Attribute (optional)

Figure 4.5.: Querying the Semantic API for insulins in *C. geographus* venom, as described in the text. Note that the interface closely resembles the JSON version of the query, while being more accessible to non-expert users.

This is a good demonstration of the information retrieval capacity of the Semantic API, but we decided to see whether a follow-up query can be used to make novel discovery, by asking the question

Do other cone snail species also produce insulin-like peptides?

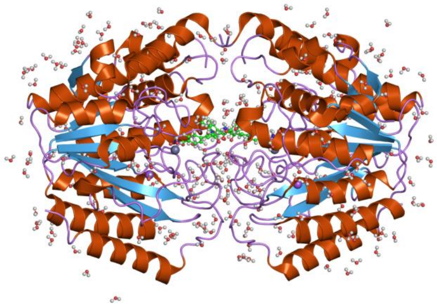
which we can pose using the query

```
{
    "select": "Species",
    "declare": {
        "Species": [
            {
                "attribute": "name",
                "operator": "contains",
                "value": "Conus"
            }
        ],
        "Pfam": [
            {
                "attribute": "name",
                "operator": "equals",
                "value": "Insulin"
            }
        ]
    },
    "aggregate": {
        "distinct": {
            "class": "Species"
        }
    }
}
```

In addition to *C. geographus*, the response to this query gives 8 other *Conus* species that produce at least 1 insulin or insulin-like peptide. Of these 9 total species, 8 are discussed in [216]. However, the 9th species—*C. victoriae*—does not show up in either of these aforementioned studies, as its venom transcriptome has only recently been characterized [201], and is therefore not discussed alongside the other species known to produce insulin-like peptides at the earlier date of publication—essentially a type of selection bias resulting from media coverage on the *Conus* insulin phenomenon occurring before the most recent discovery. The Semantic API, however, uses ontology reasoning alone to retrieve data, which is free from sources of bias like these[8].

Scorpio maurus venom and HDAC-like activity

Scorpio maurus



Histone Deacetylase 8 crystal structure

Histone deacetylase (HDAC) inhibitors are a class of drugs that have long been used to treat various psychiatric and neurological disorders, and have recently been investigated for therapeutic activity against various cancers, inflammatory conditions, and parasitic illnesses. In the results to our previous benchmarking of `VenomSeq` (see §3.3.3), we noticed strong connectivity between *S. maurus* (Israeli gold scorpion) venom and several known HDAC inhibitors. Although they are grouped together as a single class of therapeutic drugs, this is

[8] Ontologies are, of course, subject to other kinds of biases, which are covered extensively in the literature [80, 152].

misleading—different HDACs[9] have highly divergent functions, and since HDAC inhibitors only bind to certain types of HDACs, it is highly unlikely to find a venom that is similar to all (or most) types of HDAC inhibitors that are currently known. This suggests that either alternate classification schemes should be adopted in the CMap dataset, or that we need to define new metrics for summarizing connectivity at the PCL level that account for sub-class structure within PCLs.

We chose to investigate these phenomena using the Semantic API, both for the purpose of exploring possible mechanisms of the HDAC-like activity we observed, and to show that semantic querying can be used to quickly provide supporting evidence when interesting putative associations do not pass statistical significance.

4.2.7. Using the semantic API in other research contexts

Although the semantic API was designed specifically to enable discovery from VenomKB and `VenomSeq`, the principles on which it operates (as well as the benefits it provides) conceivably can be extended to virtually any domain of scientific research. The main prerequisite for adapting the semantic API to a new domain is a correctly formatted and populated OWL ontology. We developed the Venom Ontology using Protégé, but any appropriate software that is OWL-compatible and includes validation tools should suffice.

One of the tasks we have planned for continued development of semantic APIs involves creating a standalone version that can be easily reused for arbitrary applications and is not tied to the structure or implementation of VenomKB. We also aim to construct a more ef-

[9] They are generally named HDAC 1 through HDAC 11, and further grouped into classes I through IV, based on homology to the originally discovered HDACs in yeast [62].

ficient internal graph representation of the ontology to remove the dependency on Neo4j. While Neo4j and other popular graph database implementations are an incredible tool that enabled rapid prototyping of our first version of the semantic API, we only make use of a limited subset of the features of graph database servers. This suggests that we can vastly improve performance of the semantic API by reimplementing the important core algorithms of graph database servers in a slim software package written in an efficient compiled programming language (like C++ or Rust). For now, the current implementation is sufficient for low-throughput applications as a proof-of-concept to demonstrate the importance and advantages of a semantic API.

4.3. Automating discovery with VenomKB

One of the main purposes of this dissertation is to show exactly how underutilized computational methods can increase productivity in venom research. Traditionally, the main use for computers was to automate repetitive tasks and computations, and they are still used ubiquitously for this today. As the quantities of data used in various applications have grown by many orders of magnitude, and the tasks needed to analyze these data have increased substantially in complexity, computers have grown to be used for the new purpose of *interpreting* and *summarizing* the results of their own computations. This can be seen in most genomic applications—the individual steps performed in genomic analyses are, fundamentally, the same tasks that computers have been used for since the 1960s or earlier (sorting, searching, indexing data, performing simple hypothesis tests and arithmetic operations, etc.), but newer techniques are required in order to draw meaningful conclusions from these computations (performing ontology annotation, generating null models using Monte Carlo methods, phenotyping using machine learning, and others).

These patterns can be seen throughout VenomKB. For example, the connectivity analysis data we generated for 25 venoms used in `VenomSeq` yielded 11,841,175 τ scores[10], and the primitive operations used to compute these values are simple enough that the algorithm can be described in detail on approximately 1 page of text (see **Appendix C.1**). Yet, although τ is meant to be a normalized score that can be easily interpreted by humans, the sheer quantity of data makes manual analysis of these results virtually impossible.

The Semantic API provides a means for performing semi-automatic discovery on these results. Given that the user knows *what* they want (and how to compose a semantic query), the Semantic API should be able to interpret the meaning of their request and extract the appropriate data. In §4.2.6, we show particular examples of how this can be used for validating existing knowledge and for making new discoveries based on data, like that produced by `VenomSeq` as mentioned above. This functionality represents a substantial step forward in the ability to perform translation from raw molecular data into the effects that compounds have on human health.

Other tools that are part of VenomKB contribute to automated and semi-automated discovery, too. As far as we can determine, VenomKB's REST API is the first venom-centric web API that allows programmatic access to venom data for batch analyses and comparison, and therefore allows integration of VenomKB into other biomedical data applications. We are continuing to add new features to VenomKB, including the ability to submit/modify data using API calls, as well as advanced searching and filtering functions (aside from those

[10] The CMap reference dataset we used includes 473,647 expression signatures.

provided by the Semantic API).

Chapter 5.

Conclusions

In the preceding chapters, I have mainly described my efforts in toxinology as a series of individual research studies (that, granted, build on eachother sequentially). However, I view this work as a single body of research that can be decomposed into a series of connected modules. Epistemologically, this dissertation should be considered a description of computational toxinology from the perspective of a translational bioinformatics specialist. Pragmatically, the software, algorithms, and data generation/analysis pipelines in the dissertation can be grouped together as "VenomKB and related tools". To demonstrate the interconnectedness of these components, I will now take the opportunity to summarize from a macro level.

5.1. Summary of findings and original contributions

The contributions of this body of research to the fields of informatics, toxinology, and systems pharmacology can be broadly grouped into 3 categories: (1.) Conceptual advances in designing a translational infrastructure for drug discovery from venoms, (2.) discovery of new therapeutic associations between venoms (and their components) and human disease, and (3.) Newly engineered tools and resources that fill specific needs in the toxinology and biomedical data science communities.

Conceptual advances. Translational bioinformatics is still a relatively young field, and is rich with opportunities for expansion to tasks that are generally neglected with regards to computational methods. As I mentioned earlier in this text, computational toxinology is a great example of this—a field with a long history of incredible scientific productivity that is receiving increasing recognition for its contributions to biomedicine, yet in need of attention from informatics researchers and computational biologists. Beyond solely demonstrating that broad application of informatics *can* be used to facilitate and evaluate therapeutic discoveries related to venoms, we highlight the present needs to enable continued advances in this line of research. Specifically, renewed efforts in open-access publications of well-annotated next-generation sequencing data from venomous species—as well as widely adopted standards in nomenclature, data representation, and data dissemination—can invigorate computational toxinology and help ensure a steady flow of new discoveries that benefit human health and our understanding of the natural world.

New therapeutic associations. In §3.3.3 we found two promising (and plausible) new associations that merit further experimental validation, and produced a large quantity of data that can be further analyzed to discover more new associations. The first of these is a link between *A. lobata* venom and both PPAR agonist and ATPase inhibitor drugs. These associations are supported by both the connectivity (CMap) analysis and the master regulator (msVIPER) analysis, and are further validated by known molecular activities of certain argiotoxins, which are responsible for much of the bioactivity of *Argiope* venoms. The second is a link between *S. maurus* venom and FGFR inhibitor drugs (also supported by both connectivity analysis and master regulator analysis), which are anticancer drugs that are currently of great interest to the pharmaceutical industry. Although current research on

FGFR inhibitors is incredibly promising, we only know of a few examples of suitable FGFR inhibitor drug candidates to test in clinical trials, which is exactly why FGFR inhibitor-like activity in venoms is such an important finding.

New tools and resources. VenomKB, `VenomSeq`, and the utilities that accompany them are intended to be substantial resources for toxinologists, pharmacologists, students and educators, and even laypersons. Every piece of code comprising these studies is released under open-source licenses, and we strongly encourage reuse, modification, and improvements. VenomKB is meant to be informative by enabling access to previous venom discoveries and data in a centralized location that works in coordination with (rather than in opposition to) other biomedical databases related to natural toxins and drug discovery. `VenomSeq` is designed to generate much-needed perturbational differential expression data from venoms in a reproducible and economically efficient way, and we provide this data for 25 venomous species both for making an initial set of new discoveries and for acting as a proof-of-concept to potential stakeholders that may want to implement `VenomSeq` themselves.

The Semantic API is not only a tool for improving accessibility to new venom discoveries and validation of existing discoveries, but also a generalizable software paradigm for computer scientists and knowledge engineers. We designed it to serve a particular need in computational toxinology (rapidly retrieving and structuring venom knowledge enabled by ontological inference), but quickly noticed its novelty and possible uses in other domains. We have shown particular benefits it provides to both expert users of data schemas and users with limited computational experience, and also used it to validate the specific findings from our analysis of `VenomSeq` data. Even more broadly, we propose the Semantic API as a new application of ontologies integrated with high-throughtput sequencing data, and show that they can continue to provide valuable and novel insights with clear practical benefits to researchers and to biotechnologists.

5.2. Limitations and future work

Like most "big data"-driven biomedical science initiatives, VenomKB, `VenomSeq`, and the other tools related to them (Venom Ontology, the Semantic API, etc.) are intended to be used 'at-scale'. VenomKB contains only 6236 proteins, 632 species, and 5 genome pages, which is an ideal size for testing purposes, but these numbers are dwarved by the actual numbers of venomous species and proteins that exist in nature. Fortunately, next generation sequencing projects to characterize venom transcriptomes and venomous animal genomes are being conducted at a constantly increasing rate. Aside from NGS, current data needs within toxinology include improving the systematics of certain clades of venomous animals[1], comparative biochemical data of venoms and venom components (e.g., LD_{50}, molecular weight, dissociation constants, etc.), and the ability to cross-reference existing venom databases covering more granular areas (ArachnoServer and Conoserver are two of these). Additionally, VenomKB can benefit from a better representation of the ontology it is built on—for example, by adding a tool to the website that renders a certain data element within the ontology hierarchy, in a similar manner to QuickGO's "Graph View" of Gene Ontology terms [107].

`VenomSeq` can generally be improved by expansion in 3 areas: (1.) New venoms, (2.) new human cell lines, and (3.) using purified venom peptides (rather than crude venoms, as

[1] Scorpions, in particular, suffer from poorly standardized nomenclature and taxonomy, but other clades of venomous species do as well.

we use now). We are currently working with collaborators to expand `VenomSeq` to purified peptides with the intention of serving two purposes: to define a sequencing-based assay for discovering *which* proteins in a venom bind to (and alter the biochemistry of) human cells, and for performing the same type of perturbational gene expression analysis that we have only done with crude venoms to this point. It will be important to compare the data for crude venoms to the data for their corresponding purified proteins—for example, do venom proteins tend to contribute synergistically to the expression levels of individual genes, or are they optimized such that each they tend to alter the expression of unique non-overlapping sets of genes?

Overall, VenomKB and `VenomSeq` deserve a great deal of continued development. This dissertation provides the theoretical and conceptual foundations of what can be turned into a major initiative in biomedical data science. Ideally, I will be able to develop VenomKB into a major self-sustaining independent research endeavor that can be a source of new data for computational toxinologists as well as a contributor of theoretical and translational insights into therapeutic uses for venoms. Eventually, I would also like to expand to other natural product classes, to take more complete advantage of the vast diversity of bioactive (and potentially therapeutic) compounds available in nature.

Bibliography

[1] Ruben Abagyan and Maxim Totrov. High-throughput docking for lead generation. *Current opinion in chemical biology*, 5(4):375–382, 2001.

[2] Gregory P Adams and Louis M Weiner. Monoclonal antibody therapy of cancer. *Nature biotechnology*, 23(9):1147, 2005.

[3] Jean-Pierre Albrand, Martin J Blackledge, Franck Pascaud, Michelle Hollecker, and Dominique Marion. Nmr and restrained molecular dynamics study of the three-dimensional solution structure of toxin fs2, a specific blocker of the l-type calcium channel, isolated from black mamba venom. *Biochemistry*, 34(17):5923–5937, 1995.

[4] Mariano J Alvarez, Yao Shen, Federico M Giorgi, Alexander Lachmann, B Belinda Ding, B Hilda Ye, and Andrea Califano. Functional characterization of somatic mutations in cancer using network-based inference of protein activity. *Nature genetics*, 48(8):838, 2016.

[5] Gregory CA Amos, Takayoshi Awakawa, Robert N Tuttle, Anne-Catrin Letzel, Min Cheol Kim, Yuta Kudo, William Fenical, Bradley S Moore, and Paul R Jensen. Comparative transcriptomics as a guide to natural product discovery and biosynthetic gene cluster functionality. *Proceedings of the National Academy of Sciences*, 114(52):E11121–E11130, 2017.

[6] Darío Antolín-Amérigo, Carmen Moreno Aguilar, Arantza Vega, and Melchor Alvarez-Mon. Venom immunotherapy: an updated review. *Current allergy and asthma reports*, 14(7):449, 2014.

[7] Ted T Ashburn and Karl B Thor. Drug repositioning: identifying and developing new uses for existing drugs. *Nature reviews Drug discovery*, 3(8):673, 2004.

[8] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25, 2000.

[9] Christopher P Austin, Christine M Colvis, and Noel T Southall. Deconstructing the translational tower of babel. *Clinical and translational science*, 2018.

[10] Jerry Avorn. The $2.6 billion pill—methodologic and policy considerations. *New England Journal of Medicine*, 372(20):1877–1879, 2015.

[11] Sahar Awwad and Ukrit Angkawinitwong. Overview of antibody drug delivery. *Pharmaceutics*, 10(3):83, 2018.

[12] Albert-László Barabási. Scale-free networks: a decade and beyond. *science*, 325(5939):412–413, 2009.

[13] Frederique Bard, Catherine Cannon, Robin Barbour, Rae-Lyn Burke, Dora Games, Henry Grajeda, Teresa Guido, Kang Hu, Jiping Huang, Kelly Johnson-Wood, et al. Peripherally administered antibodies against amyloid β-peptide enter the central nervous system and reduce pathology in a mouse model of alzheimer disease. *Nature medicine*, 6(8):916, 2000.

[14] Courtenay Bartholomew. Acute scorpion pancreatitis in trinidad. *Br Med J*, 1(5697):666–668, 1970.

[15] Katia Basso, Adam A Margolin, Gustavo Stolovitzky, Ulf Klein, Riccardo Dalla-Favera, and Andrea Califano. Reverse engineering of regulatory networks in human b cells. *Nature genetics*, 37(4):382, 2005.

[16] Sean Bechhofer, Raphael Volz, and Phillip Lord. Cooking the semantic web with the owl api. In *International Semantic Web Conference*, pages 659–675. Springer, 2003.

[17] Dave Beckett and Brian McBride. Rdf/xml syntax specification (revised). *W3C recommendation*, 10(2.3), 2004.

[18] Timothy M Beissinger and Gota Morota. Medical subject heading (mesh) annotations illuminate maize genetics and evolution. *Plant methods*, 13(1):8, 2017.

[19] Emilio Benfenati, Andrey A Toropov, Alla P Toropova, Alberto Manganaro, and Rodolfo Gonella Diaza. Coral software: Qsar for anticancer agents. *Chemical biology & drug design*, 77(6):471–476, 2011.

[20] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society. Series B (Methodological)*, pages 289–300, 1995.

[21] Ellen L Berg, Eugene C Butcher, and Jennifer Melrose. Biomap characterization of biologically active agents, December 2 2003. US Patent 6,656,695.

[22] Juliana L Bernardoni, Leijiane F Sousa, Luciana S Wermelinger, Aline S Lopes, Benedito C Prezoto, Solange MT Serrano, Russolina B Zingali, and Ana M Moura-da Silva. Functional variability of snake venom metalloproteinases: adaptive advantages

in targeting different prey and implications for human envenomation. *PLoS One*, 9(10):e109651, 2014.

[23] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific american*, 284(5):34–43, 2001.

[24] Arthur Beyder and Gianrico Farrugia. Targeting ion channels for the treatment of gastrointestinal motility disorders. *Therapeutic advances in gastroenterology*, 5(1):5–21, 2012.

[25] Jari Björne, Suwisa Kaewphan, and Tapio Salakoski. Uturku: drug named entity recognition and drug-drug interaction extraction using svm classification and domain knowledge. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 651–659, 2013.

[26] Judith A Blake and Carol J Bult. Beyond the data deluge: data integration and bio-ontologies. *Journal of biomedical informatics*, 39(3):314–320, 2006.

[27] Jeffrey M Blaney and Eric J Martin. Computational approaches for combinatorial library design and molecular diversity analysis. *Current opinion in chemical biology*, 1(1):54–59, 1997.

[28] Olivier Bodenreider and Robert Stevens. Bio-ontologies: current trends and future directions. *Briefings in bioinformatics*, 7(3):256–274, 2006.

[29] Gadi Borkow, JoséMaría Gutiérrez, and Michael Ovadia. Isolation and characterization of synergistic hemorrhagins from the venom of the snake bothrops asper. *Toxicon*, 31(9):1137–1150, 1993.

[30] David Brown and Giulio Superti-Furga. Rediscovering the sweet spot in drug discovery. *Drug discovery today*, 8(23):1067–1077, 2003.

[31] Justin Eliot Busch, Albert Deirchow Lin, Patrick John Graydon, and Maureen Caudill. Ontology-based parser for natural language processing, April 11 2006. US Patent 7,027,974.

[32] Erin C Bush, Forest Ray, Mariano J Alvarez, Ronald Realubit, Hai Li, Charles Karan, Andrea Califano, and Peter A Sims. Plate-seq for genome-wide regulatory network analysis of high-throughput screens. *Nature communications*, 8(1):105, 2017.

[33] William S Bush and Jason H Moore. Genome-wide association studies. *PLoS computational biology*, 8(12):e1002822, 2012.

[34] AJ Butte and S Ito. Translational bioinformatics: data-driven drug discovery and development. *Clinical Pharmacology & Therapeutics*, 91(6):949–952, 2012.

[35] Juan J Calvete, Libia Sanz, Yamileth Angulo, Bruno Lomonte, and José María Gutiérrez. Venoms, venomics, antivenomics. *FEBS letters*, 583(11):1736–1743, 2009.

[36] Juan J Calvete, Libia Sanz, Davinia Pla, Bruno Lomonte, and José María Gutiérrez. Omics meets biology: application to the design and preclinical assessment of antivenoms. *Toxins*, 6(12):3388–3405, 2014.

[37] Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L Madden. Blast+: architecture and applications. *BMC bioinformatics*, 10(1):421, 2009.

[38] Delroy Cameron, Olivier Bodenreider, Hima Yalamanchili, Tu Danh, Sreeram Vallabhaneni, Krishnaprasad Thirunarayan, Amit P Sheth, and Thomas C Rindflesch. A graph-based recovery and decomposition of swanson's hypothesis using semantic predications. *Journal of biomedical informatics*, 46(2):238–251, 2013.

[39] Nicholas R Casewell, Wolfgang Wüster, Freek J Vonk, Robert A Harrison, and Bryan G Fry. Complex cocktails: the evolutionary novelty of venoms. *Trends in ecology & evolution*, 28(4):219–229, 2013.

[40] Yan Chen, Melitta Bilban, Carolyn A Foster, and Dale L Boger. Solution-phase parallel synthesis of a pharmacophore library of hun-7293 analogues: A general chemical mutagenesis approach to defining structure- function properties of naturally occurring cyclic (depsi) peptides. *Journal of the American Chemical Society*, 124(19):5431–5440, 2002.

[41] Tiejun Cheng, Qingliang Li, Zhigang Zhou, Yanli Wang, and Stephen H Bryant. Structure-based virtual screening for drug discovery: a problem-centric review. *The AAPS journal*, 14(1):133–141, 2012.

[42] Artem Cherkasov, Eugene N Muratov, Denis Fourches, Alexandre Varnek, Igor I Baskin, Mark Cronin, John Dearden, Paola Gramatica, Yvonne C Martin, Roberto Todeschini, et al. Qsar modeling: where have you been? where are you going to? *Journal of medicinal chemistry*, 57(12):4977–5010, 2014.

[43] Vivian G Cheung, Michael Morley, Francisco Aguilar, Aldo Massimi, Raju Kucherlapati, and Geoffrey Childs. Making and reading microarrays. *Nature genetics*, 21(1s):15, 1999.

[44] Ekaterina Galkina Cleary, Jennifer M Beierlein, Navleen Surjit Khanuja, Laura M McNamee, and Fred D Ledley. Contribution of nih funding to new drug approvals 2010–2016. *Proceedings of the National Academy of Sciences*, 115(10):2329–2334, 2018.

[45] Sara E Cnudde, Mary Prorok, Francis J Castellino, and James H Geiger. Metal ion determinants of conantokin dimerization as revealed in the x-ray crystallographic structure of the cd 2+/mg 2+–con-t [k7γ] complex. *JBIC Journal of Biological Inorganic Chemistry*, 15(5):667–675, 2010.

[46] Biomedical Data Translator Consortium. Toward a universal biomedical data translator. *Clinical and translational science*, 2018.

[47] William Cookson, Liming Liang, Gonçalo Abecasis, Miriam Moffatt, and Mark Lathrop. Mapping complex disease traits with global gene expression. *Nature Reviews Genetics*, 10(3):184, 2009.

[48] Geoffrey A Cordell. Biodiversity and drug discovery—a symbiotic relationship. *Phytochemistry*, 55(6):463–480, 2000.

[49] Nick Cox, James R Kintzing, Mark Smith, Gerald A Grant, and Jennifer R Cochran. Integrin-targeting knottin peptide–drug conjugates are potent inhibitors of tumor cell proliferation. *Angewandte Chemie International Edition*, 55(34):9894–9897, 2016.

[50] Giorgio Cozza, Paolo Bonvini, Elisa Zorzi, Giorgia Poletto, Mario A Pagano, Stefania Sarno, Arianna Donella-Deana, Giuseppe Zagotto, Angelo Rosolen, Lorenzo A Pinna, et al. Identification of ellagic acid as potent inhibitor of protein kinase ck2: a successful example of a virtual screening application. *Journal of medicinal chemistry*, 49(8):2363–2366, 2006.

[51] Lu Cui, Yu Wang, Zhihong Liu, Hongzhuan Chen, Hao Wang, Xinxin Zhou, and Jun Xu. Discovering new acetylcholinesterase inhibitors by mining the buzhongyiqi decoction recipe data. *Journal of chemical information and modeling*, 55(11):2455–2463, 2015.

[52] Jennifer C Daltry, Wolfgang Wüster, and Roger S Thorpe. Diet and snake venom evolution. *Nature*, 379(6565):537, 1996.

[53] Bernard de Bono, Karen Rothfels, L Castagnoli, MG Williams, and Bijay Jassal. Signaling by fgfr [homo sapiens]. *Reactome*, 2007.

[54] Ricardo C Rodríguez de la Vega and Lourival D Possani. Overview of scorpion toxins specific for na+ channels and related peptides: biodiversity, structure–function relationships and evolution. *Toxicon*, 46(8):831–844, 2005.

[55] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. Large scale distributed deep networks. In *Advances in neural information processing systems*, pages 1223–1231, 2012.

[56] Arnold L Demain. Prescription for an ailing pharmaceutical industry. *Nature biotechnology*, 20(4):331, 2002.

[57] Paul M Dewick. *Medicinal natural products: a biosynthetic approach*. John Wiley & Sons, 2002.

[58] Priyanka Dhiman, Neelam Malik, and Anurag Khatkar. 3d-qsar and in-silico studies of natural products and related derivatives as monoamine oxidase inhibitors. *Current neuropharmacology*, 16(6):881–900, 2018.

[59] Daniel A Dias, Sylvia Urban, and Ute Roessner. A historical overview of natural products in drug discovery. *Metabolites*, 2(2):303–336, 2012.

[60] Michael Dickson and Jean Paul Gagnon. Key factors in the rising cost of new drug discovery and development. *Nature reviews Drug discovery*, 3(5):417, 2004.

[61] Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe Batut, Mark Chaisson, and Thomas R Gingeras. Star: ultrafast universal rna-seq aligner. *Bioinformatics*, 29(1):15–21, 2013.

[62] Milos Dokmanovic, Cathy Clarke, and Paul A Marks. Histone deacetylase inhibitors: overview and perspectives. *Molecular cancer research*, 5(10):981–989, 2007.

[63] Jürgen Drews. Drug discovery: a historical perspective. *Science*, 287(5460):1960–1964, 2000.

[64] Joel T Dudley, Marina Sirota, Mohan Shenoy, Reetesh K Pai, Silke Roedder, Annie P Chiang, Alex A Morgan, Minnie M Sarwal, Pankaj Jay Pasricha, and Atul J Butte. Computational repositioning of the anticonvulsant topiramate for inflammatory bowel disease. *Science translational medicine*, 3(96):96ra76–96ra76, 2011.

[65] Mathias Dunkel, Melanie Fullbeck, Stefanie Neumann, and Robert Preissner. Supernatural: a searchable database of available natural compounds. *Nucleic acids research*, 34(suppl_1):D678–D683, 2006.

[66] Sébastien Dutertre, Ai-Hua Jin, Irina Vetter, Brett Hamilton, Kartik Sunagar, Vincent Lavergne, Valentin Dutertre, Bryan G Fry, Agostinho Antunes, Deon J Venter, et al.

Evolution of separate predation-and defence-evoked venoms in carnivorous cone snails. *Nature communications*, 5:3521, 2014.

[67] D Eastwood, L Findlay, S Poole, C Bird, M Wadhwa, M Moore, C Burns, R Thorpe, and R Stebbings. Monoclonal antibody tgn1412 trial failure explained by species differences in cd28 expression on cd4+ effector memory t-cells. *British journal of pharmacology*, 161(3):512–526, 2010.

[68] Totta Ehret, Francesca Torelli, Christian Klotz, Amy B Pedersen, and Frank Seeber. Translational rodent models for research on parasitic protozoa—a review of confounders and possibilities. *Frontiers in cellular and infection microbiology*, 7:238, 2017.

[69] Keren Ettinger, Gadi Cohen, Tatjana Momic, and Philip Lazarovici. The effects of a chactoid scorpion venom and its purified toxins on rat blood pressure and mast cells histamine release. *Toxins*, 5(8):1332–1342, 2013.

[70] Antonio Fabregat, Konstantinos Sidiropoulos, Phani Garapati, Marc Gillespie, Kerstin Hausmann, Robin Haw, Bijay Jassal, Steven Jupe, Florian Korninger, Sheldon McKay, et al. The reactome pathway knowledgebase. *Nucleic acids research*, 44(D1):D481–D487, 2015.

[71] Scott Farrar and D Terence Langendoen. A linguistic ontology for the semantic web. *GLOT international*, 7(3):97–100, 2003.

[72] Scott Federhen. The ncbi taxonomy database. *Nucleic acids research*, 40(D1):D136–D143, 2011.

[73] Garret A FitzGerald. Drugs, industry, and academia, 2008.

[74] Alexander Fleming. On the antibacterial action of cultures of a penicillium, with special reference to their use in the isolation of b. influenzae. *British journal of experimental pathology*, 10(3):226, 1929.

[75] Richard Frank and Richard Hargreaves. Clinical biomarkers in drug discovery and development. *Nature reviews Drug discovery*, 2(7):566, 2003.

[76] Carol Friedman, Philip O Alderson, John HM Austin, James J Cimino, and Stephen B Johnson. A general natural-language text processor for clinical radiology. *Journal of the American Medical Informatics Association*, 1(2):161–174, 1994.

[77] Raoul Frijters, Marianne Van Vugt, Ruben Smeets, René Van Schaik, Jacob De Vlieg, and Wynand Alkema. Literature mining for the discovery of hidden connections between drugs, genes and diseases. *PLoS computational biology*, 6(9):e1000943, 2010.

[78] Da-Hua Fu, Wei Jiang, Jian-Ting Zheng, Gui-Yu Zhao, Yan Li, Hong Yi, Zhuo-Rong Li, Jian-Dong Jiang, Ke-Qian Yang, Yanchang Wang, et al. Jadomycin b, an aurora-b kinase inhibitor discovered through virtual screening. *Molecular cancer therapeutics*, 7(8):2386–2393, 2008.

[79] Stephen P Gardner. Ontologies in drug discovery. *Drug Discovery Today: Technologies*, 2(3):235–240, 2005.

[80] Pascale Gaudet and Christophe Dessimoz. Gene ontology: pitfalls, biases, and remedies. In *The Gene Ontology Handbook*, pages 189–205. Springer, 2017.

[81] Anna Gaulton, Anne Hersey, Michał Nowotka, A Patrícia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J Bellis, Elena Cibrián-Uhalte, et al. The chembl database in 2017. *Nucleic acids research*, 45(D1):D945–D954, 2016.

[82] Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/-variance dilemma. *Neural computation*, 4(1):1–58, 1992.

[83] Janet C Gonder. Select agent regulations. *ILAR journal*, 46(1):4–7, 2005.

[84] José María Gutiérrez, Alexandra Rucavado, Teresa Escalante, and Cecilia Díaz. Hemorrhage induced by snake venom metalloproteinases: biochemical and biophysical mechanisms involved in microvessel damage. *Toxicon*, 45(8):997–1011, 2005.

[85] Inbal Halperin, Dariya S Glazer, Shirley Wu, and Russ B Altman. The feature framework for protein function annotation: modeling new functions, improving performance, and extending to novel applications. *BMC genomics*, 9(2):S2, 2008.

[86] Yun Hao, Kayla Quinnies, Ronald Realubit, Charles Karan, and Nicholas P Tatonetti. Tissue-specific analysis of pharmacological pathways. *CPT: Pharmacometrics & Systems Pharmacology*, 2018.

[87] Yun Hao and Nicholas P Tatonetti. Predicting g protein-coupled receptor downstream signaling by tissue expression. *Bioinformatics*, 32(22):3435–3443, 2016.

[88] JEFFREY B Harborne. Classes and functions of secondary products from plants. *Chemicals from plants*, pages 1–25, 1999.

[89] Olaf Hartig and Jun Zhao. Using web data provenance for quality assessment. In *CEUR Workshop Proceedings*, 2009.

[90] Alan L Harvey. Natural products in drug discovery. *Drug discovery today*, 13(19-20):894–901, 2008.

[91] David Harvey, Philip Bardelang, Sara L Goodacre, Alan Cockayne, and Neil R Thomas. Antibiotic spider silk: Site-specific functionalization of recombinant spider silk using "click" chemistry. *Advanced Materials*, 29(10):1604245, 2017.

[92] Duane C Hassane, Monica L Guzman, Cheryl Corbett, Xiaojie Li, Ramzi Abboud, Fay Young, Jane L Liesveld, Martin Carroll, and Craig T Jordan. Discovery of agents that eradicate leukemia stem cells using an in silico screen of public gene expression data. *Blood*, 111(12):5654–5662, 2008.

[93] Quan-Yuan He, Quan-Ze He, Xing-Can Deng, Lei Yao, Er Meng, Zhong-Hua Liu, and Song-Ping Liang. Atdb: a uni-database platform for animal toxins. *Nucleic acids research*, 36(suppl_1):D293–D297, 2007.

[94] Quanze He, Wenjun Han, Quanyuan He, Linju Huo, Jingjing Zhang, Yong Lin, Ping Chen, and Songping Liang. Atdb 2.0: A database integrated toxin-ion channel interaction data. *Toxicon*, 56(4):644–647, 2010.

[95] Dominik Heckmann, Tim Schwartz, Boris Brandherm, Michael Schmitz, and Margeritta von Wilamowitz-Moellendorff. Gumo–the general user model ontology. In *International Conference on User Modeling*, pages 428–432. Springer, 2005.

[96] Jeffrey Heer, Stuart K Card, and James A Landay. Prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430. ACM, 2005.

[97] Haley Hieronymus, Justin Lamb, Kenneth N Ross, Xiao P Peng, Cristina Clement, Anna Rodina, Maria Nieto, Jinyan Du, Kimberly Stegmaier, Srilakshmi M Raj, et al. Gene expression signature-based chemical genomic prediction identifies a novel class of hsp90 pathway modulators. *Cancer cell*, 10(4):321–330, 2006.

[98] Keiichi Hiramoto, Yurika Yamate, Hiromi Kobayashi, Masamitsu Ishii, Takanori Miura, Eisuke F Sato, and Masayasu Inoue. Effect of the smell of seirogan, a wood creosote, on dermal and intestinal mucosal immunity and allergic inflammation. *Journal of clinical biochemistry and nutrition*, 51(2):91–95, 2012.

[99] Christoph Hock, Uwe Konietzko, Johannes R Streffer, Jay Tracy, Andri Signorell, Britta Müller-Tillmanns, Ulrike Lemke, Katharina Henke, Eva Moritz, Esmeralda Garcia, et al. Antibodies against β-amyloid slow cognitive decline in alzheimer's disease. *Neuron*, 38(4):547–554, 2003.

[100] Reinhard Hohlfeld. Biotechnological agents for the immunotherapy of multiple sclerosis. principles, problems and perspectives. *Brain: a journal of neurology*, 120(5):865–916, 1997.

[101] Hennie R Hoogenboom. Selecting and screening recombinant antibody libraries. *Nature biotechnology*, 23(9):1105, 2005.

[102] Andrew L Hopkins and Colin R Groom. The druggable genome. *Nature reviews Drug discovery*, 1(9):727, 2002.

[103] George Hripcsak, Jon D Duke, Nigam H Shah, Christian G Reich, Vojtech Huser, Martijn J Schuemie, Marc A Suchard, Rae Woong Park, Ian Chi Kei Wong, Peter R Rijnbeek, et al. Observational health data sciences and informatics (ohdsi): opportunities for observational researchers. *Studies in health technology and informatics*, 216:574, 2015.

[104] Po-Ssu Huang, Scott E Boyken, and David Baker. The coming of age of de novo protein design. *Nature*, 537(7620):320, 2016.

[105] James P Hughes, Stephen Rees, S Barrett Kalindjian, and Karen L Philpott. Principles of early drug discovery. *British journal of pharmacology*, 162(6):1239–1249, 2011.

[106] Philip Hunter. The paradox of model organisms: The use of model organisms in research will continue despite their shortcomings. *EMBO reports*, 9(8):717–720, 2008.

[107] Rachael P Huntley, David Binns, Emily Dimmer, Daniel Barrell, Claire O'Donovan, and Rolf Apweiler. Quickgo: a user tutorial for the web-based gene ontology browser. *Database*, 2009, 2009.

[108] Yasumasa Ishida, Yasutoshi Agata, Keiichi Shibahara, and Tasuki Honjo. Induced expression of pd-1, a novel member of the immunoglobulin gene superfamily, upon programmed cell death. *The EMBO journal*, 11(11):3887–3895, 1992.

[109] Laurent S Jespers, Andy Roberts, Stephen M Mahler, Greg Winter, and Hennie R Hoogenboom. Guiding the selection of human antibodies from phage display repertoires to a single epitope of an antigen. *Bio/technology*, 12(9):899, 1994.

[110] Zhen Jiang and Robert Gentleman. Extensions to gene set enrichment. *Bioinformatics*, 23(3):306–313, 2006.

[111] William L Jorgensen. The many roles of computation in drug discovery. *Science*, 303(5665):1813–1818, 2004.

[112] Raja Jothi and Balaji Raghavachari. Approximation algorithms for the capacitated minimum spanning tree problem and its variants in network design. *ACM Transactions on Algorithms (TALG)*, 1(2):265–282, 2005.

[113] Florence Jungo, Lydie Bougueleret, Ioannis Xenarios, and Sylvain Poux. The uniprotkb/swiss-prot tox-prot program: A central hub of integrated venom protein data. *Toxicon*, 60(4):551–557, 2012.

[114] Marek Jutel, Werner J Pichler, Dejan Skrbic, Adrian Urwyler, Clemens Dahinden, and UR Müller. Bee venom immunotherapy results in decrease of il-4 and il-5 and increase of ifn-gamma secretion in specific allergen-stimulated t cell cultures. *The Journal of Immunology*, 154(8):4187–4194, 1995.

[115] Quentin Kaas, Jan-Christoph Westermann, and David J Craik. Conopeptide characterization and classifications: an analysis using conoserver. *Toxicon*, 55(8):1491–1509, 2010.

[116] Quentin Kaas, Rilei Yu, Ai-Hua Jin, Sebastien Dutertre, and David J Craik. Conoserver: updated content, knowledge, and discovery tools in the conopeptide database. *Nucleic acids research*, 40(D1):D325–D330, 2011.

[117] Minoru Kanehisa, Miho Furumichi, Mao Tanabe, Yoko Sato, and Kanae Morishima. Kegg: new perspectives on genomes, pathways, diseases and drugs. *Nucleic acids research*, 45(D1):D353–D361, 2016.

[118] Martin Karplus and J Andrew McCammon. Molecular dynamics simulations of biomolecules. *Nature Structural and Molecular Biology*, 9(9):646, 2002.

[119] Leonard Katz and Richard H Baltz. Natural product discovery: past, present, and future. *Journal of industrial microbiology & biotechnology*, 43(2-3):155–176, 2016.

[120] Thomas B Kepler and Timothy C Elston. Stochasticity in transcriptional regulation: origins, consequences, and mathematical representations. *Biophysical journal*, 81(6):3116–3136, 2001.

[121] Mahmud Tareq Hassan Khan, I Orhan, FS Şenol, M Kartal, B Şener, M Dvorská, K Šmejkal, and T Šlapetová. Cholinesterase inhibitory activities of some flavonoid derivatives and chosen xanthone and their molecular docking studies. *Chemico-Biological Interactions*, 181(3):383–389, 2009.

[122] R Manjunatha Kini and Herbert J Evans. Effects of snake venom proteins on blood platelets. *Toxicon*, 28(12):1387–1422, 1990.

[123] Curtis D Klaassen and John B Watkins. *Casarett and Doull's toxicology: the basic science of poisons*, volume 5. McGraw-Hill New York, 1996.

[124] Isaac S Kohane, Atul J Butte, and Alvin Kho. *Microarrays for an integrative genomics.* MIT press, 2002.

[125] Dušan Kordiš and Franc Gubenšek. Adaptive evolution of animal toxin multigene families. *Gene*, 261(1):43–52, 2000.

[126] Alexandru Korotcov, Valery Tkachenko, Daniel P Russo, and Sean Ekins. Comparison of deep learning with multiple machine learning methods and metrics using diverse drug discovery data sets. *Molecular pharmaceutics*, 14(12):4462–4475, 2017.

[127] Dimitrios A Koutsomitropoulos, Ricardo Borillo Domenech, and Georgia D Solomou. A structured semantic query interface for reasoning-based search and retrieval. In *Extended Semantic Web Conference*, pages 17–31. Springer, 2011.

[128] Wolfgang Kuchinke, Jozef Aerts, SC Semler, and Christian Ohmann. Cdisc standard-based electronic archiving of clinical trials. *Methods of information in medicine*, 48(05):408–413, 2009.

[129] Werner Kühlbrandt. Biology, structure and mechanism of p-type atpases. *Nature reviews Molecular cell biology*, 5(4):282, 2004.

[130] Jayashri Kulkarni, Kathryn A Garland, Antonietta Scaffidi, Barbara Headey, Robyn Anderson, Anthony de Castella, Paul Fitzgerald, and Susan R Davis. A pilot study of hormone modulation as a new treatment for mania in women with bipolar affective disorder. *Psychoneuroendocrinology*, 31(4):543–547, 2006.

[131] Justin Lamb, Emily D Crawford, David Peck, Joshua W Modell, Irene C Blat, Matthew J Wrobel, Jim Lerner, Jean-Philippe Brunet, Aravind Subramanian, Kenneth N Ross, et al. The connectivity map: using gene-expression signatures to connect small molecules, genes, and disease. *science*, 313(5795):1929–1935, 2006.

[132] Thomas O Larsen, Jørn Smedsgaard, Kristian F Nielsen, Michael E Hansen, and Jens C Frisvad. Phenotypic taxonomy and metabolite profiling in microbial drug discovery. *Natural product reports*, 22(6):672–695, 2005.

[133] Mette Laursen, Jonas Lindholt Gregersen, Laure Yatime, Poul Nissen, and Natalya U Fedosova. Structures and characterization of digoxin-and bufalin-bound na+, k+-atpase compared with the ouabain-bound complex. *Proceedings of the National Academy of Sciences*, 112(6):1755–1760, 2015.

[134] Andreas Hougaard Laustsen. Toxin synergism in snake venoms. *Toxin Reviews*, 35(3-4):165–170, 2016.

[135] Antonio Lavecchia. Machine-learning approaches in drug discovery: methods and applications. *Drug discovery today*, 20(3):318–331, 2015.

[136] Dana R Leach, Matthew F Krummel, and James P Allison. Enhancement of antitumor immunity by ctla-4 blockade. *Science*, 271(5256):1734–1736, 1996.

[137] Ki Won Lee, Ann M Bode, and Zigang Dong. Molecular targets of phytochemicals for cancer prevention. *Nature Reviews Cancer*, 11(3):211, 2011.

[138] Kuen-Haur Lee, Hsiang-Ling Lo, Wan-Chun Tang, Heidi Hao-yun Hsiao, and Pei-Ming Yang. A gene expression signature-based approach reveals the mechanisms of action of the chinese herbal medicine berberine. *Scientific reports*, 4:6394, 2014.

[139] Tong Ihn Lee, Nicola J Rinaldi, François Robert, Duncan T Odom, Ziv Bar-Joseph, Georg K Gerber, Nancy M Hannett, Christopher T Harbison, Craig M Thompson, Itamar Simon, et al. Transcriptional regulatory networks in saccharomyces cerevisiae. *science*, 298(5594):799–804, 2002.

[140] Norman L Letvin and Bruce D Walker. Immunopathogenesis and immunotherapy in aids virus infections. *Nature medicine*, 9(7):861, 2003.

[141] Richard J Lewis and Maria L Garcia. Therapeutic potential of venom peptides. *Nature reviews drug discovery*, 2(10):790, 2003.

[142] Qingliang Li, Tiejun Cheng, Yanli Wang, and Stephen H Bryant. Pubchem as a public resource for drug discovery. *Drug discovery today*, 15(23-24):1052–1057, 2010.

[143] Mark A Lindsay. Finding new drug targets in the 21st century. *Drug discovery today*, 10(23-24):1683–1687, 2005.

[144] Christopher A Lipinski. Lead-and drug-like compounds: the rule-of-five revolution. *Drug Discovery Today: Technologies*, 1(4):337–341, 2004.

[145] Carolyn E Lipscomb. Medical subject headings (mesh). *Bulletin of the Medical Library Association*, 88(3):265, 2000.

[146] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15(12):550, 2014.

[147] Scott J Lusher, Ross McGuire, René C van Schaik, C David Nicholson, and Jacob de Vlieg. Data-driven medicinal chemistry in the era of big data. *Drug discovery today*, 19(7):859–868, 2014.

[148] Y Lussier and C Friedman. Biomedlee: a natural-language processor for extracting and representing phenotypes, underlying molecular mechanisms and their relationships. *ISMB: 2007*, 2007.

[149] Chao Lv, Xueting Wu, Xia Wang, Juan Su, Huawu Zeng, Jing Zhao, Shan Lin, Runhui Liu, Honglin Li, Xuan Li, et al. The gene expression profiles in response to 102 traditional chinese medicine (tcm) components: a general template for research on tcms. *Scientific reports*, 7(1):352, 2017.

[150] Jose M Vazquez-Naya, Marcos Martinez-Romero, Ana B Porto-Pazos, Francisco Novoa, Manuel Valladares-Ayerbes, Javier Pereira, Cristian R Munteanu, and Julian Dorado. Ontologies of drug discovery and design for neurology, cardiology and oncology. *Current pharmaceutical design*, 16(24):2724–2736, 2010.

[151] Dik-Lung Ma, Daniel Shiu-Hin Chan, and Chung-Hang Leung. Molecular docking for virtual screening of natural product databases. *Chemical Science*, 2(9):1656–1665, 2011.

[152] Alexander Maedche and Steffen Staab. Ontology learning for the semantic web. *IEEE Intelligent systems*, 16(2):72–79, 2001.

[153] Derek D Mafong and Robert R Henry. Exenatide as a treatment for diabetes and obesity: implications for cardiovascular risk reduction. *Current atherosclerosis reports*, 10(1):55–60, 2008.

[154] Anita Malhotra, Simon Creer, John B Harris, Reto Stöcklin, Philippe Favreau, and Roger S Thorpe. Predicting function from sequence in a large multifunctional toxin family. *Toxicon*, 72:113–125, 2013.

[155] James Malone, Robert Stevens, Simon Jupp, Tom Hancocks, Helen Parkinson, and Cath Brooksbank. Ten simple rules for selecting a bio-ontology. *PLoS computational biology*, 12(2):e1004743, 2016.

[156] Ilona Mandrika, Peteris Prusis, Sviatlana Yahorava, Kaspars Tars, and Jarl ES Wikberg. Qsar of multiple mutated antibodies. *Journal of Molecular Recognition: An Interdisciplinary Journal*, 20(2):97–102, 2007.

[157] John Mann. Natural products in cancer chemotherapy: past, present and future. *Nature Reviews Cancer*, 2(2):143, 2002.

[158] Tiina Manninen, Jugoslava Aćimović, Riikka Havela, Heidi Teppola, and Marja-Leena Linne. Challenges in reproducibility, replicability, and comparability of computational models and tools for neuronal and glial networks, cells, and subcellular structures. *Frontiers in neuroinformatics*, 12, 2018.

[159] Christopher D Manning, Christopher D Manning, and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

[160] Rachael A Maplestone, Martin J Stone, and Dudley H Williams. The evolutionary role of secondary metabolites—a review. *Gene*, 115(1):151–157, 1992.

[161] Gary Marcus. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631*, 2018.

[162] Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC bioinformatics*, 7(1):S7, 2006.

[163] William Markland, Arthur Charles Ley, and Robert Charles Ladner. Iterative optimization of high-affinity protease inhibitors using phage display. 2. plasma kallikrein and thrombin. *Biochemistry*, 35(24):8058–8067, 1996.

[164] John Markwell and David W Brooks. "link rot" limits the usefulness of web-based educational materials in biochemistry and molecular biology. *Biochemistry and Molecular Biology Education*, 31(1):69–72, 2003.

[165] Brian McBride. Jena: A semantic web toolkit. *IEEE Internet computing*, 6(6):55–59, 2002.

[166] Catherine A McCarty, Rex L Chisholm, Christopher G Chute, Iftikhar J Kullo, Gail P Jarvik, Eric B Larson, Rongling Li, Daniel R Masys, Marylyn D Ritchie, Dan M Roden, et al. The emerge network: a consortium of biorepositories linked to electronic medical records data for conducting genomic studies. *BMC medical genomics*, 4(1):13, 2011.

[167] Catherine A McCarty and Russell A Wilke. Biobanking and pharmacogenomics. *Pharmacogenomics*, 11(5):637–641, 2010.

[168] John G Menting, Joanna Gajewiak, Christopher A MacRaild, Danny Hung-Chieh Chou, Maria M Disotuar, Nicholas A Smith, Charleen Miller, Judit Erchegyi, Jean E Rivier, Baldomero M Olivera, et al. A minimized human insulin-receptor-binding motif revealed in a conus geographus venom insulin. *Nature structural & molecular biology*, 23(10):916, 2016.

[169] DW Miller, AD Jones, JS Goldston, MP Rowe, and AH Rowe. Sex differences in defensive behavior and venom of the striped bark scorpion centruroides vittatus (scorpiones: Buthidae), 2016.

[170] Paolo Missier, Khalid Belhajjame, and James Cheney. The w3c prov family of specifications for modelling provenance metadata. In *Proceedings of the 16th International Conference on Extending Database Technology*, pages 773–776. ACM, 2013.

[171] Milan Mladenović, Alexandros Patsilinakos, Adele Pirolli, Manuela Sabatino, and Rino Ragno. Understanding the molecular determinant of reversible human monoamine oxidase b inhibitors containing 2 h-chromen-2-one core: Structure-based and ligand-based derived three-dimensional quantitative structure–activity relationships predictive models. *Journal of chemical information and modeling*, 57(4):787–814, 2017.

[172] Scott T Moe, Daryl L Smith, Yongwei Eric Chien, Joanna L Raszkiewicz, Linda D Artman, and Alan L Mueller. Design, synthesis, and biological evaluation of spider toxin (argiotoxin-636) analogs as nmda receptor antagonists. *Pharmaceutical research*, 15(1):31–38, 1998.

[173] Russell J Molyneux, Stephen T Lee, Dale R Gardner, Kip E Panter, and Lynn F James. Phytochemicals: the good, the bad and the ugly? *Phytochemistry*, 68(22-24):2973–2985, 2007.

[174] Emily Mullin. Animal venom database could be boon to drug development. *Forbes*, 2015.

[175] Bernard Munos. Lessons from 60 years of pharmaceutical innovation. *Nature reviews Drug discovery*, 8(12):959, 2009.

[176] Aliyu Musa, Laleh Soltan Ghoraie, Shu-Dong Zhang, Galina Glazko, Olli Yli-Harja, Matthias Dehmer, Benjamin Haibe-Kains, and Frank Emmert-Streib. A review of connectivity map and computational approaches in pharmacogenomics. *Briefings in bioinformatics*, 19(3):506–523, 2017.

[177] Prudence Mutowo, A Patrícia Bento, Nathan Dedman, Anna Gaulton, Anne Hersey, Jane Lomax, and John P Overington. A drug target slim: using gene ontology and

gene ontology annotations to navigate protein-ligand target space in chembl. *Journal of biomedical semantics*, 7(1):59, 2016.

[178] National Center for Complementary and Integrative Health. Natural products research—information for researchers, 2017.

[179] Nature Publishing Group. All natural. *Nature chemical biology*, 3(7):351, 2007.

[180] David J Newman and Gordon M Cragg. Natural products as sources of new drugs from 1981 to 2014. *Journal of natural products*, 79(3):629–661, 2016.

[181] Linh T Ngo, Joseph I Okogun, and William R Folk. 21st century natural product research and drug development and traditional medicines. *Natural product reports*, 30(4):584–592, 2013.

[182] Alexandra C Nica and Emmanouil T Dermitzakis. Using gene expression to investigate the genetic basis of complex disorders. *Human molecular genetics*, 17(R2):R129–R134, 2008.

[183] Sachiko Nitta and Keiji Numata. Biopolymer-based nanoparticles for drug/gene delivery and tissue engineering. *International journal of molecular sciences*, 14(1):1629–1654, 2013.

[184] Andrew E Nixon, Daniel J Sexton, and Robert C Ladner. Drugs derived from phage display: from candidate identification to clinical practice. 6(1):73–85, 2014.

[185] Bernard V North, David Curtis, and Pak C Sham. A note on the calculation of empirical p values from monte carlo procedures. *American journal of human genetics*, 71(2):439, 2002.

[186] Natalya Fridman Noy, Monica Crubézy, Ray W Fergerson, Holger Knublauch, Samson W Tu, Jennifer Vendetti, and Mark A Musen. Protégé-2000: an open-source ontology-development and knowledge-acquisition environment. In *AMIA... Annual Symposium proceedings. AMIA Symposium*, volume 2003, pages 953–953. American Medical Informatics Association, 2003.

[187] Anika Oellrich, Nigel Collier, Tudor Groza, Dietrich Rebholz-Schuhmann, Nigam Shah, Olivier Bodenreider, Mary Regina Boland, Ivo Georgiev, Hongfang Liu, Kevin Livingston, et al. The digital revolution in phenotyping. *Briefings in bioinformatics*, 17(5):819–830, 2015.

[188] Fatih Ozsolak and Patrice M Milos. Rna sequencing: advances, challenges and opportunities. *Nature reviews genetics*, 12(2):87, 2011.

[189] Nataraj S Pagadala, Khajamohiddin Syed, and Jack Tuszynski. Software for molecular docking: a review. *Biophysical reviews*, 9(2):91–102, 2017.

[190] Adrià Pérez, Gerard Martínez-Rosell, and Gianni De Fabritiis. Simulations meet machine learning in structural biology. *Current opinion in structural biology*, 49:139–144, 2018.

[191] Seth Pettie and Vijaya Ramachandran. An optimal minimum spanning tree algorithm. *Journal of the ACM (JACM)*, 49(1):16–34, 2002.

[192] Sandy S Pineda, Pierre-Alain Chaumeil, Anne Kunert, Quentin Kaas, Mike WC Thang, Lien Li, Michael Nuhn, Volker Herzig, Natalie J Saez, Ben Cristofori-Armstrong, et al. Arachnoserver 3.0: an online resource for automated discovery, analysis and annotation of spider toxins. *Bioinformatics*, 1:3, 2017.

[193] Janet Piñero, Àlex Bravo, Núria Queralt-Rosinach, Alba Gutiérrez-Sacristán, Jordi Deu-Pons, Emilio Centeno, Javier García-García, Ferran Sanz, and Laura I Furlong. Disgenet: a comprehensive platform integrating information on human disease-associated genes and variants. *Nucleic acids research*, page gkw943, 2016.

[194] Pimolpan Pithayanukul, Jiraporn Leanpolchareanchai, and Patchreenart Saparpakorn. Molecular docking studies and anti- snake venom metalloproteinase activity of thai mango seed kernel extract. *Molecules*, 14(9):3198–3213, 2009.

[195] Norman I Platnick and Robert J Raven. Spider systematics: past and future. *Zootaxa*, 3683(5):595–600, 2013.

[196] Jaroslaw Polanski. Receptor dependent multidimensional qsar for modeling drug-receptor interactions. *Current medicinal chemistry*, 16(25):3243–3257, 2009.

[197] Cristiano Gonçalves Ponte, Eduardo Lira Nóbrega, Vanessa Câmara Fernandes, Wilmar Dias da Silva, and Guilherme Suarez-Kurtz. Inhibition of the myotoxic activities of three african bitis venoms (b. rhinoceros, b. arietans and b. nasicornis) by a polyvalent antivenom. *Toxicon*, 55(2-3):536–540, 2010.

[198] Mette H Poulsen, Simon Lucas, Tinna B Bach, Anne F Barslund, Claudius Wenzler, Christel B Jensen, Anders S Kristensen, and Kristian Strømgaard. Structure–activity relationship studies of argiotoxins: selective and potent inhibitors of ionotropic glutamate receptors. *Journal of medicinal chemistry*, 56(3):1171–1181, 2013.

[199] Rona R Ramsay, Marija R Popovic-Nikolic, Katarina Nikolic, Elisa Uliassi, and Maria Laura Bolognesi. A perspective on multi-target drug discovery and design for complex diseases. *Clinical and translational medicine*, 7(1):3, 2018.

[200] Raj R Rao and William S Kisaalita. Biochemical and electrophysiological differentiation profile of a human neuroblastoma (imr-32) cell line. *In Vitro Cellular & Developmental Biology-Animal*, 38(8):450–456, 2002.

[201] Samuel D Robinson, Qing Li, Pradip K Bandyopadhyay, Joanna Gajewiak, Mark Yandell, Anthony T Papenfuss, Anthony W Purcell, Raymond S Norton, and Helena Safavi-Hemami. Hormone-like peptides in the venoms of marine cone snails. *General and comparative endocrinology*, 244:11–18, 2017.

[202] Samuel D Robinson and Raymond S Norton. Conotoxin gene superfamilies. *Marine drugs*, 12(12):6058–6101, 2014.

[203] Tiago Rodrigues, Daniel Reker, Petra Schneider, and Gisbert Schneider. Counting on natural products for drug design. *Nature chemistry*, 8(6):531, 2016.

[204] Joseph Romano, Victor Nwankwo, and Nicholas Tatonetti. Venomkb v2.0: A knowledge repository for computational toxinology. *bioRxiv*, page 295204, 2018.

[205] Joseph D Romano and Nicholas P Tatonetti. Venomkb, a new knowledge base for facilitating the validation of putative venom therapies. *Scientific data*, 2:150065, 2015.

[206] Joseph D Romano and Nicholas P Tatonetti. Using a novel ontology to inform the discovery of therapeutic peptides from animal venoms. *AMIA Summits on Translational Science Proceedings*, 2016:209, 2016.

[207] Joseph D Romano, William G Tharp, and Indra Neil Sarkar. Adapting simultaneous analysis phylogenomic techniques to study complex disease gene relationships. *Journal of biomedical informatics*, 54:10–38, 2015.

[208] Nina Rønsted, Matthew RE Symonds, Trine Birkholm, Søren Brøgger Christensen, Alan W Meerow, Marianne Molander, Per Mølgaard, Gitte Petersen, Nina Rasmussen, Johannes Van Staden, et al. Can phylogeny predict chemical diversity and potential medicinal activity of plants? a case study of amaryllidaceae. *BMC evolutionary biology*, 12(1):182, 2012.

[209] Steven A Rosenberg, James C Yang, and Nicholas P Restifo. Cancer immunotherapy: moving beyond current vaccines. *Nature medicine*, 10(9):909, 2004.

[210] Jinlong Ru, Peng Li, Jinan Wang, Wei Zhou, Bohui Li, Chao Huang, Pidong Li, Zihu Guo, Weiyang Tao, Yinfeng Yang, et al. Tcmsp: a database of systems pharmacology for drug discovery from herbal medicines. *Journal of cheminformatics*, 6(1):13, 2014.

[211] David Ruau, Michael Mbagwu, Joel T Dudley, Vijay Krishnan, and Atul J Butte. Comparison of automated and human assignment of mesh terms on publicly-available molecular datasets. *Journal of biomedical informatics*, 44:S39–S43, 2011.

[212] Tim Ruder, Kartik Sunagar, Eivind AB Undheim, Syed A Ali, Tak-Cheung Wai, Dolyce HW Low, Timothy NW Jackson, Glenn F King, Agostinho Antunes, and Bryan G Fry. Molecular phylogeny and evolution of the proteins encoded by coleoid (cuttlefish, octopus, and squid) posterior venom glands. *Journal of molecular evolution*, 76(4):192–204, 2013.

[213] Jeffrey D Rudolf, Xiaohui Yan, and Ben Shen. Genome neighborhood network reveals insights into enediyne biosynthesis and facilitates prediction and prioritization for discovery. *Journal of industrial microbiology & biotechnology*, 43(2-3):261–276, 2016.

[214] FE Russell. When a snake strikes. *Emerg Med*, 22:21–43, 1990.

[215] Andrey Rzhetsky, David Wajngurt, Naeun Park, and Tian Zheng. Probing genetic overlap among complex human phenotypes. *Proceedings of the National Academy of Sciences*, 104(28):11694–11699, 2007.

[216] Helena Safavi-Hemami, Joanna Gajewiak, Santhosh Karanth, Samuel D Robinson, Beatrix Ueberheide, Adam D Douglass, Amnon Schlegel, Julita S Imperial, Maren Watkins, Pradip K Bandyopadhyay, et al. Specialized insulin is used for chemical warfare by fish-hunting cone snails. *Proceedings of the National Academy of Sciences*, 112(6):1743–1748, 2015.

[217] Veronica Salmaso and Stefano Moro. Bridging molecular docking to molecular dynamics in exploring ligand-protein recognition process: An overview. *Frontiers in pharmacology*, 9, 2018.

[218] Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513, 2010.

[219] Justin O Schmidt. *Insect defenses: adaptive mechanisms and strategies of prey and predators.* SUNY Press, 1990.

[220] Valerie A Schneider, Tina Graves-Lindsay, Kerstin Howe, Nathan Bouk, Hsiu-Chuan Chen, Paul A Kitts, Terence D Murphy, Kim D Pruitt, Françoise Thibaud-Nissen,

Derek Albracht, et al. Evaluation of grch38 and de novo haploid genome assemblies demonstrates the enduring quality of the reference assembly. *Genome research*, 2017.

[221] Martijn J Schuemie, Preciosa M Coloma, Huub Straatman, Ron MC Herings, Gianluca Trifirò, Justin Neil Matthews, David Prieto-Merino, Mariam Molokhia, Lars Pedersen, Rosa Gini, et al. Using electronic health care records for drug safety signal detection: a comparative evaluation of statistical methods. *Medical care*, pages 890–897, 2012.

[222] Su K Seo, Jae H Choi, Young H Kim, Woo J Kang, Hye Y Park, Jae H Suh, Beom K Choi, Dass S Vinay, and Byoung S Kwon. 4-1bb-mediated immunotherapy of rheumatoid arthritis. *Nature medicine*, 10(10):1088, 2004.

[223] Jeff Sevigny, Ping Chiao, Thierry Bussière, Paul H Weinreb, Leslie Williams, Marcel Maier, Robert Dunstan, Stephen Salloway, Tianle Chen, Yan Ling, et al. The antibody aducanumab reduces aβ plaques in alzheimer's disease. *Nature*, 537(7618):50, 2016.

[224] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003.

[225] Marina Sirota, Joel T Dudley, Jeewon Kim, Annie P Chiang, Alex A Morgan, Alejandro Sweet-Cordero, Julien Sage, and Atul J Butte. Discovery and preclinical validation of drug indications using compendia of public gene expression data. *Science translational medicine*, 3(96):96ra77–96ra77, 2011.

[226] Arvind Sivasubramanian, Aroop Sircar, Sidhartha Chaudhury, and Jeffrey J Gray. Toward high-resolution homology modeling of antibody fv regions and application to antibody–antigen docking. *Proteins: Structure, Function, and Bioinformatics*, 74(2):497–514, 2009.

[227] Gregory Sliwoski, Sandeepkumar Kothiwale, Jens Meiler, and Edward W Lowe. Computational methods in drug discovery. *Pharmacological reviews*, 66(1):334–395, 2014.

[228] John B Smelcer. *Understanding user errors in database query.* PhD thesis, University of Michigan, 1990.

[229] William Leo Smith and Ward C Wheeler. Venom evolution widespread in fishes: a phylogenetic road map for the bioprospecting of piscine venoms. *Journal of Heredity*, 97(3):206–217, 2006.

[230] Michael Spedding. New directions for drug discovery. *Dialogues in clinical neuroscience*, 8(3):295, 2006.

[231] Reisa A Sperling, Paul S Aisen, Laurel A Beckett, David A Bennett, Suzanne Craft, Anne M Fagan, Takeshi Iwatsubo, Clifford R Jack Jr, Jeffrey Kaye, Thomas J Montine, et al. Toward defining the preclinical stages of alzheimer's disease: Recommendations from the national institute on aging-alzheimer's association workgroups on diagnostic guidelines for alzheimer's disease. *Alzheimer's & dementia*, 7(3):280–292, 2011.

[232] Kristina Spiess, Andreas Lammel, and Thomas Scheibel. Recombinant spider silk proteins for applications in biomaterials. *Macromolecular bioscience*, 10(9):998–1007, 2010.

[233] MJ Stone and DH Williams. On the evolution of functional secondary metabolites (natural products). *Molecular microbiology*, 6(1):29–34, 1992.

[234] Aravind Subramanian, Rajiv Narayan, Steven M Corsello, David D Peck, Ted E Natoli, Xiaodong Lu, Joshua Gould, John F Davis, Andrew A Tubelli, Jacob K Asiedu, et al. A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. *Cell*, 171(6):1437–1452, 2017.

[235] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

[236] Peng Gang Sun, Lin Gao, and Shan Han. Prediction of human disease-related gene clusters by clustering analysis. *International journal of biological sciences*, 7(1):61, 2011.

[237] Kartik Sunagar, Eivind AB Undheim, Holger Scheib, Eric CK Gren, Chip Cochran, Carl E Person, Ivan Koludarov, Wayne Kelln, William K Hayes, Glenn F King, et al. Intraspecific venom variation in the medically significant southern pacific rattlesnake (crotalus oreganus helleri): biodiscovery, clinical and evolutionary implications. *Journal of proteomics*, 99:68–83, 2014.

[238] Don R Swanson. Fish oil, raynaud's syndrome, and undiscovered public knowledge. *Perspectives in biology and medicine*, 30(1):7–18, 1986.

[239] Siang Yong Tan and Yvonne Tatsumura. Alexander fleming (1881–1955): discoverer of penicillin. *Singapore medical journal*, 56(7):366, 2015.

[240] Nicholas P Tatonetti, P Ye Patrick, Roxana Daneshjou, and Russ B Altman. Data-driven prediction of drug effects and interactions. *Science translational medicine*, 4(125):125ra31–125ra31, 2012.

[241] Haythum O Tayeb, Evan D Murray, Bruce H Price, and Frank I Tarazi. Bapineuzumab and solanezumab for alzheimer's disease: is the 'amyloid cascade hypothesis' still alive? *Expert opinion on biological therapy*, 13(7):1075–1084, 2013.

[242] Nicholas K Terrett, Mark Gardner, David W Gordon, Ryszard J Kobylecki, and John Steele. Combinatorial synthesis—the design of compound libraries and their application to drug discovery. *Tetrahedron*, 51(30):8135–8173, 1995.

[243] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

[244] Nicholas Thomford, Dimakatso Senthebane, Arielle Rowe, Daniella Munro, Palesa Seele, Alfred Maroyi, and Kevin Dzobo. Natural products for drug discovery in the 21st century: Innovations for novel drug discovery. *International journal of molecular sciences*, 19(6):1578, 2018.

[245] Wen Torng and Russ B Altman. 3d deep convolutional neural networks for amino acid environment similarity analysis. *BMC bioinformatics*, 18(1):302, 2017.

[246] Paolo Tosco and Thomas Balle. Open3dqsar: a new open-source software aimed at high-throughput chemometric analysis of molecular interaction fields. *Journal of molecular modeling*, 17(1):201–208, 2011.

[247] John Tsai. For better or worse: Introducing the gnu general public license version 3. *Berkeley Tech. LJ*, 23:547, 2008.

[248] Dmitry Tsarkov and Ian Horrocks. Fact++ description logic reasoner: System description. In *International Joint Conference on Automated Reasoning*, pages 292–297. Springer, 2006.

[249] Heather D VanGuilder, Kent E Vrana, and Willard M Freeman. Twenty-five years of quantitative pcr for gene expression analysis. *Biotechniques*, 44(5):619–626, 2008.

[250] VK Vyas, RD Ukawala, M Ghate, and C Chintha. Homology modeling a fast tool for drug discovery: current perspectives. *Indian journal of pharmaceutical sciences*, 74(1):1, 2012.

[251] Michael Wainberg, Daniele Merico, Andrew Delong, and Brendan J Frey. Deep learning in biomedicine. *Nature biotechnology*, 36(9):829, 2018.

[252] Peter H Walls and Michael JE Sternberg. New algorithm to model protein-protein recognition based on surface complementarity: Applications to antibody-antigen docking. *Journal of molecular biology*, 228(1):277–297, 1992.

[253] Dagmar Waltemath and Olaf Wolkenhauer. How modeling standards, software, and initiatives support reproducibility in systems biology and systems medicine. *IEEE Trans. Biomed. Engineering*, 63(10):1999–2006, 2016.

[254] Jim Webber and Ian Robinson. *A programmatic introduction to neo4j*. Addison-Wesley Professional, 2018.

[255] Nicole Gray Weiskopf and Chunhua Weng. Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research. *Journal of the American Medical Informatics Association*, 20(1):144–151, 2013.

[256] Matthew E Welsch, Scott A Snyder, and Brent R Stockwell. Privileged scaffolds for library design and drug discovery. *Current opinion in chemical biology*, 14(3):347–361, 2010.

[257] Julian White, David Warrell, Michael Eddleston, Bart J Currie, Ian M Whyte, and Geoffrey K Isbister. Clinical toxinology—where are we now? antivenoms. *Journal of Toxicology: Clinical Toxicology*, 41(3):263–276, 2003.

[258] RA Wilke, H Xu, JC Denny, DM Roden, RM Krauss, CA McCarty, RL Davis, T Skaar, J Lamba, and G Savova. The emerging role of electronic medical records in pharmacogenomics. *Clinical Pharmacology & Therapeutics*, 89(3):379–386, 2011.

[259] Nigel Williams. How to get databases talking the same language. *Science*, 275(5298):301–302, 1997.

[260] Jeannette M Wing. Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences*, 366(1881):3717–3725, 2008.

[261] David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 46(D1):D1074–D1082, 2017.

[262] Andrew M Wollacott, Luke N Robinson, Boopathy Ramakrishnan, Hamid Tissire, Karthik Viswanathan, Zachary Shriver, and Gregory J Babcock. Structural prediction of antibody-april complexes by computational docking constrained by antigen saturation mutagenesis library data. *Journal of Molecular Recognition*, page e2778, 2019.

[263] Wilson Wong, Wei Liu, and Mohammed Bennamoun. Ontology learning from text: A look back and into the future. *ACM Computing Surveys (CSUR)*, 44(4):20, 2012.

[264] Guoxiang Xie, Robert Plumb, Mingming Su, Zhaohui Xu, Aihua Zhao, Mingfeng Qiu, Xiangbao Long, Zhong Liu, and Wei Jia. Ultra-performance lc/tof ms analysis of medicinal panax herbs for metabolomic research. *Journal of separation science*, 31(6-7):1015–1026, 2008.

[265] Tao Xie, Sicheng Song, Sijia Li, Liang Ouyang, Lin Xia, and Jian Huang. Review of natural product databases. *Cell proliferation*, 48(4):398–404, 2015.

[266] Tianhua Yan, Qiang Fu, Jing Wang, and Shiping Ma. Uplc-ms/ms determination of ephedrine, methylephedrine, amygdalin and glycyrrhizic acid in beagle plasma and its application to a pharmacokinetic study after oral administration of ma huang tang. *Drug testing and analysis*, 7(2):158–163, 2015.

[267] Yang Yang, Malgorzata A Mis, Mark Estacion, Sulayman D Dib-Hajj, and Stephen G Waxman. Na v 1.7 as a pharmacogenomic target for pain: Moving toward precision medicine. *Trends in pharmacological sciences*, 2018.

[268] Lixia Yao, Yiye Zhang, Yong Li, Philippe Sanseau, and Pankaj Agarwal. Electronic health records: Implications for drug discovery. *Drug discovery today*, 16(13-14):594–599, 2011.

[269] Yi Yue, Gang-Xiu Chu, Xue-Shi Liu, Xing Tang, Wei Wang, Guang-Jin Liu, Tao Yang, Tie-Jun Ling, Xiao-Gang Wang, Zheng-Zhu Zhang, et al. Tmdb: A literature-curated database for small molecular compounds found from tea. *BMC plant biology*, 14(1):243, 2014.

[270] Xian Zeng, Peng Zhang, Weidong He, Chu Qin, Shangying Chen, Lin Tao, Yali Wang, Ying Tan, Dan Gao, Bohua Wang, et al. Npass: natural product activity and species source database for natural product research, discovery and tool development. *Nucleic acids research*, 46(D1):D1217–D1222, 2017.

[271] Bo Zhang, Yingxue Fu, Chao Huang, Chunli Zheng, Ziyin Wu, Wenjuan Zhang, Xiaoyan Yang, Fukai Gong, Yuerong Li, Xiaoyu Chen, et al. New strategy for drug

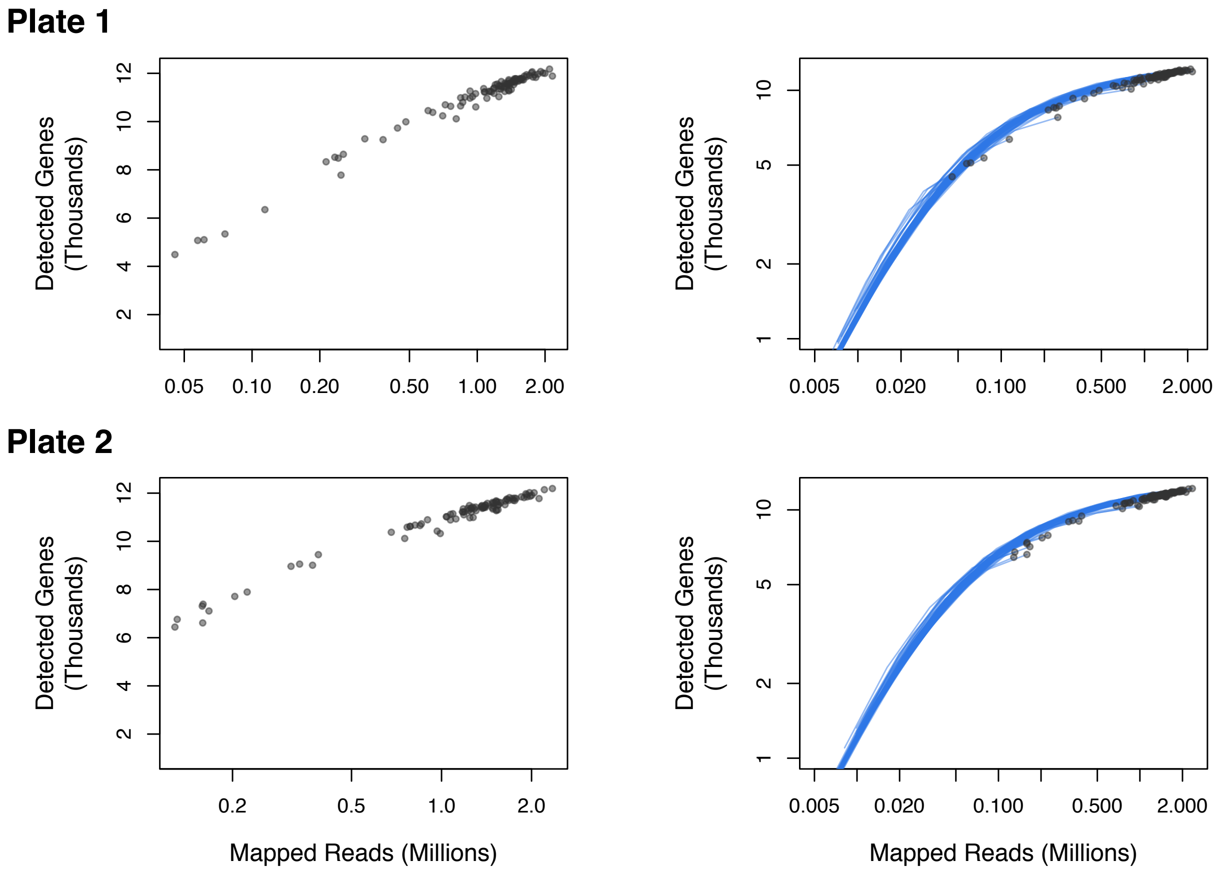
discovery by large-scale association analysis of molecular networks of different species. *Scientific reports*, 6:21872, 2016.

[272] Rui Zhang, Nivedha Manohar, Elliot Arsoniadis, Yan Wang, Terrence J Adam, Serguei V Pakhomov, and Genevieve B Melton. Evaluating term coverage of herbal and dietary supplements in electronic health records. In *AMIA Annual Symposium Proceedings*, volume 2015, page 1361. American Medical Informatics Association, 2015.

[273] Nadine Ziemert and Paul R Jensen. Phylogenetic approaches to natural product structure prediction. In *Methods in enzymology*, volume 517, pages 161–182. Elsevier, 2012.

Appendix A.

PLATE-Seq quality control data



(a) Library Complexity

(b) Saturation Analysis

Figure A.1.: Quality control plots. (a.) Number of detected genes (mapped reads ≥ 2) as a function of the total number of mapped reads per sample. (b.) Saturation analysis by *in silico* subsampling. Original data points are indicated by the black dots.

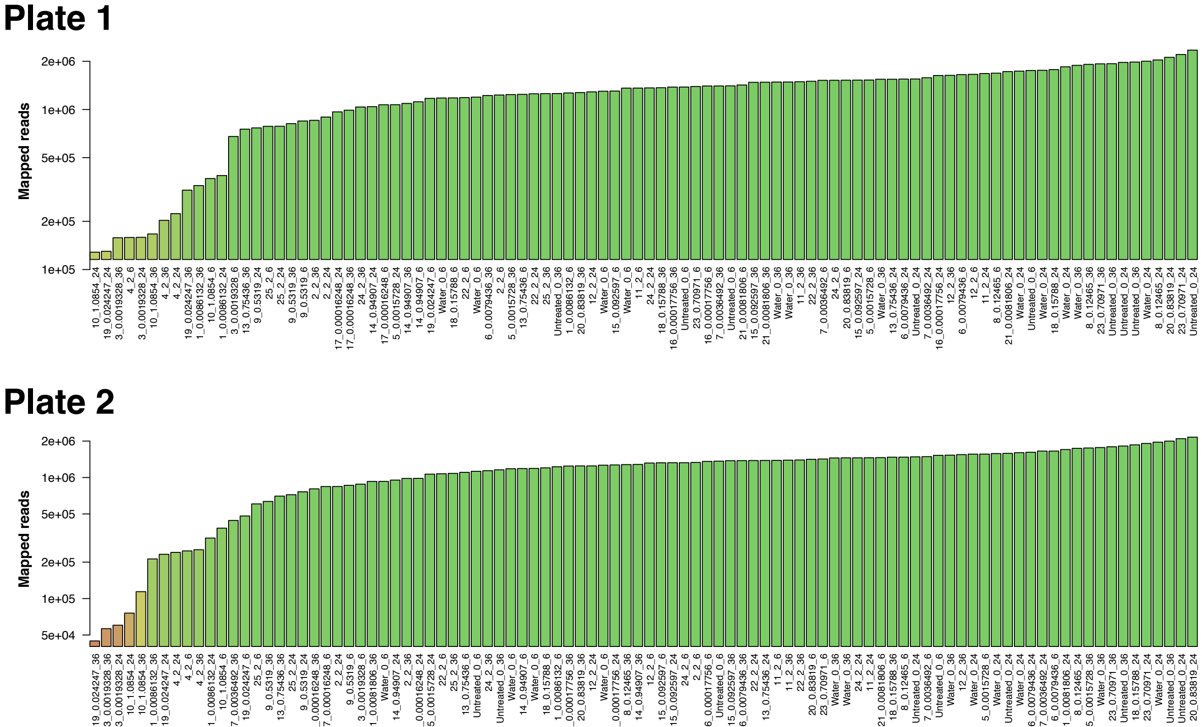


Plate 1

Plate 2

Figure A.2.: Barplot showing the number of mapped reads per sample.

Plate 1:

Well	Venom	Conc. (uG/uL)	Time (Hrs)
A1	1	0.008 6	6
B1	2	2.000 0	6
C1	3	0.001 9	6
D1	4	2.000 0	6
E1	5	0.001 6	6
F1	6	0.007 9	6
G1	7	0.003 6	6
H1	8	0.124 7	6
A2	9	0.531 9	6
B2	10	1.085 4	6
C2	11	2.000 0	6
D2	12	2.000 0	6
E2	13	0.754 4	6
F2	14	0.949 1	6
G2	15	0.092 6	6
H2	16	0.000 2	6
A3	17	0.000 2	6
B3	18	0.157 9	6
C3	19	0.024 2	6
D3	20	0.838 2	6
E3	21	0.008 2	6
F3	22	2.000 0	6
G3	23	0.709 7	6
H3	24	2.000 0	6
A4	25	2.000 0	6
B4	Water	–	6
C4	Water	–	6
D4	Water	–	6
E4	Water	–	6
F4	Untreated	–	6
G4	Untreated	–	6
H4	Untreated	–	6
A5	1	0.008 6	24
B5	2	2.000 0	24
C5	3	0.001 9	24
D5	4	2.000 0	24
E5	5	0.001 6	24
F5	6	0.007 9	24
G5	7	0.003 6	24
H5	8	0.124 7	24
A6	9	0.531 9	24
B6	10	1.085 4	24
C6	11	2.000 0	24
D6	12	2.000 0	24
E6	13	0.754 4	24
F6	14	0.949 1	24
G6	15	0.092 6	24
H6	16	0.000 2	24
A7	17	0.000 2	24
B7	18	0.157 9	24
C7	19	0.024 2	24
D7	20	0.838 2	24
E7	21	0.008 2	24
F7	22	2.000 0	24
G7	23	0.709 7	24
H7	24	2.000 0	24
A8	25	2.000 0	24
B8	Water	–	24
C8	Water	–	24
D8	Water	–	24
E8	Water	–	24
F8	Untreated	–	24
G8	Untreated	–	24
H8	Untreated	–	24
A9	1	0.008 6	36
B9	2	2.000 0	36
C9	3	0.001 9	36
D9	4	2.000 0	36
E9	5	0.001 6	36
F9	6	0.007 9	36
G9	7	0.003 6	36
H9	8	0.124 7	36
A10	9	0.531 9	36
B10	10	1.085 4	36
C10	11	2.000 0	36
D10	12	2.000 0	36
E10	13	0.754 4	36
F10	14	0.949 1	36
G10	15	0.092 6	36
H10	16	0.000 2	36
A11	17	0.000 2	36
B11	18	0.157 9	36
C11	19	0.024 2	36
D11	20	0.838 2	36
E11	21	0.008 2	36
F11	22	2.000 0	36
G11	23	0.709 7	36
H11	24	2.000 0	36
A12	25	2.000 0	36
B12	Water	–	36
C12	Water	–	36
D12	Water	–	36
E12	Water	–	36
F12	Untreated	–	36
G12	Untreated	–	36
H12	Untreated	–	36

Plate 2:

Well	Venom	Conc. (uG/uL)	Time (Hrs)
A1	1	0.008 6	6
B1	2	2.000 0	6
C1	3	0.001 9	6
D1	4	2.000 0	6
E1	5	0.001 6	6
F1	6	0.007 9	6
G1	7	0.003 6	6
H1	8	0.124 7	6
A2	9	0.531 9	6
B2	10	1.085 4	6
C2	11	2.000 0	6
D2	12	2.000 0	6
E2	13	0.754 4	6
F2	14	0.949 1	6
G2	15	0.092 6	6
H2	16	0.000 2	6
A3	17	0.000 2	6
B3	18	0.157 9	6
C3	19	0.024 2	6
D3	20	0.838 2	6
E3	21	0.008 2	6
F3	22	2.000 0	6
G3	23	0.709 7	6
H3	24	2.000 0	6
A4	25	2.000 0	6
B4	Water	–	6
C4	Water	–	6
D4	Water	–	6
E4	Water	–	6
F4	Untreated	–	6
G4	Untreated	–	6
H4	Untreated	–	6
A5	1	0.008 6	24
B5	2	2.000 0	24
C5	3	0.001 9	24
D5	4	2.000 0	24
E5	5	0.001 6	24
F5	6	0.007 9	24
G5	7	0.003 6	24
H5	8	0.124 7	24
A6	9	0.531 9	24
B6	10	1.085 4	24
C6	11	2.000 0	24
D6	12	2.000 0	24
E6	13	0.754 4	24
F6	14	0.949 1	24
G6	15	0.092 6	24
H6	16	0.000 2	24
A7	17	0.000 2	24
B7	18	0.157 9	24
C7	19	0.024 2	24
D7	20	0.838 2	24
E7	21	0.008 2	24
F7	22	2.000 0	24
G7	23	0.709 7	24
H7	24	2.000 0	24
A8	25	2.000 0	24
B8	Water	–	24
C8	Water	–	24
D8	Water	–	24
E8	Water	–	24
F8	Untreated	–	24
G8	Untreated	–	24
H8	Untreated	–	24
A9	1	0.008 6	36
B9	2	2.000 0	36
C9	3	0.001 9	36
D9	4	2.000 0	36
E9	5	0.001 6	36
F9	6	0.007 9	36
G9	7	0.003 6	36
H9	8	0.124 7	36
A10	9	0.531 9	36
B10	10	1.085 4	36
C10	11	2.000 0	36
D10	12	2.000 0	36
E10	13	0.754 4	36
F10	14	0.949 1	36
G10	15	0.092 6	36
H10	16	0.000 2	36
A11	17	0.000 2	36
B11	18	0.157 9	36
C11	19	0.024 2	36
D11	20	0.838 2	36
E11	21	0.008 2	36
F11	22	2.000 0	36
G11	23	0.709 7	36
H11	24	2.000 0	36
A12	25	2.000 0	36
B12	Water	–	36
C12	Water	–	36
D12	Water	–	36
E12	Water	–	36
F12	Untreated	–	36
G12	Untreated	–	36
H12	Untreated	–	36

Table A.1.: Layout of samples in 2 96-well plates for PLATE-Seq.

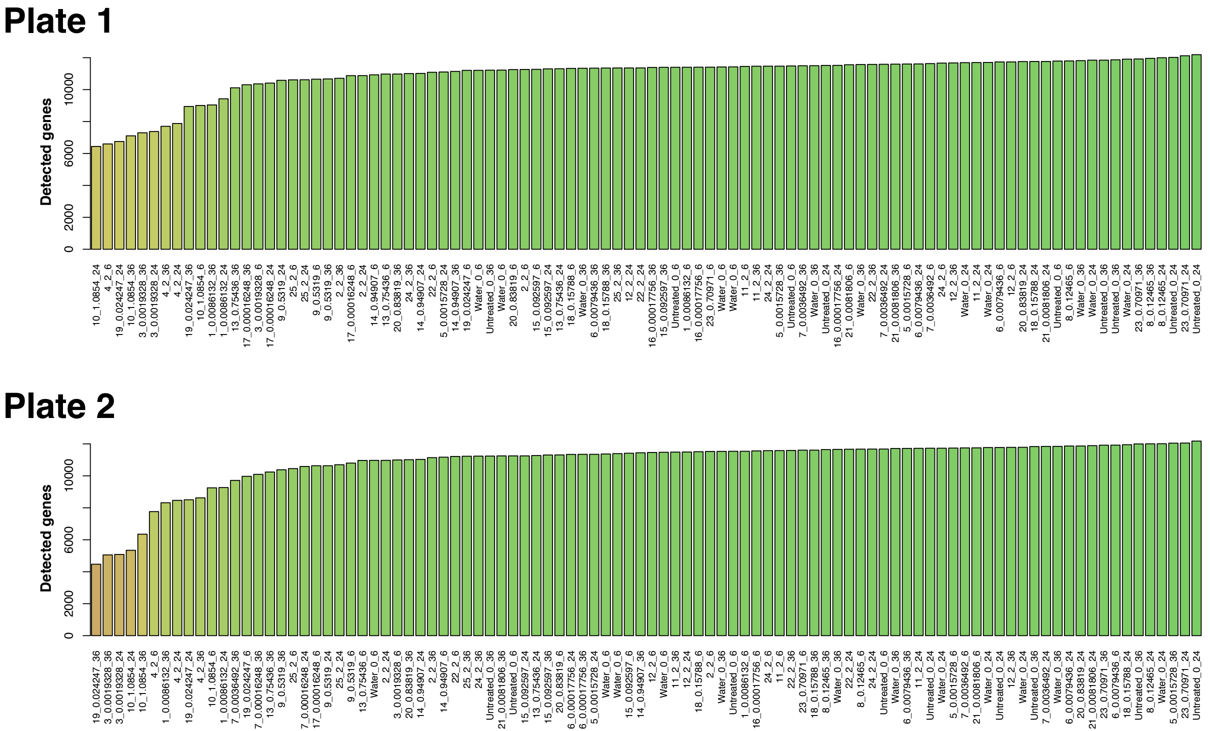
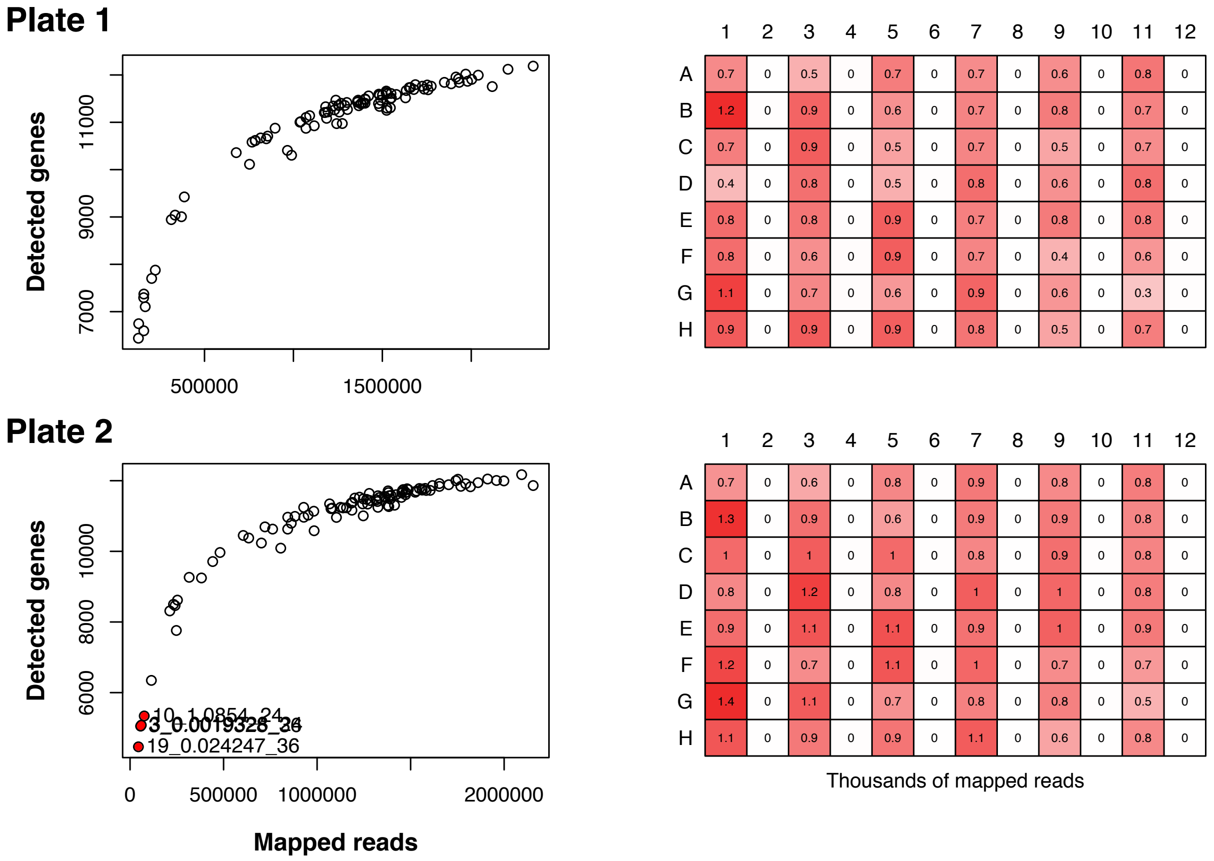


Figure A.3.: Barplot showing the number of detected genes per sample.



	1	2	3	4	5	6	7	8	9	10	11	12
A	0.7	0	0.5	0	0.7	0	0.7	0	0.6	0	0.8	0
B	1.2	0	0.9	0	0.6	0	0.7	0	0.8	0	0.7	0
C	0.7	0	0.9	0	0.5	0	0.7	0	0.5	0	0.7	0
D	0.4	0	0.8	0	0.5	0	0.8	0	0.6	0	0.8	0
E	0.8	0	0.8	0	0.9	0	0.7	0	0.8	0	0.8	0
F	0.8	0	0.6	0	0.9	0	0.7	0	0.4	0	0.6	0
G	1.1	0	0.7	0	0.6	0	0.9	0	0.6	0	0.3	0
H	0.9	0	0.9	0	0.9	0	0.8	0	0.5	0	0.7	0

	1	2	3	4	5	6	7	8	9	10	11	12
A	0.7	0	0.6	0	0.8	0	0.9	0	0.8	0	0.8	0
B	1.3	0	0.9	0	0.6	0	0.9	0	0.9	0	0.8	0
C	1	0	1	0	1	0	0.8	0	0.9	0	0.8	0
D	0.8	0	1.2	0	0.8	0	1	0	1	0	0.8	0
E	0.9	0	1.1	0	1.1	0	0.9	0	1	0	0.9	0
F	1.2	0	0.7	0	1.1	0	1	0	0.7	0	0.7	0
G	1.4	0	1.1	0	0.7	0	0.8	0	0.8	0	0.5	0
H	1.1	0	0.9	0	0.9	0	1.1	0	0.6	0	0.8	0

Thousands of mapped reads

(a) Mapped reads x Genes

(b) Spike-ins

Figure A.4.: Detected genes and spike-ins. (a.) Association between the number of mapped reads and detected genes for each of the 96 analyzed samples. (b.) Heatmap showing the number of reads (thousands) mapping to spike-ins for each of the samples.

Appendix B.

VenomSeq JSON schema

```
{
  "$schema": "http://json-schema.org/draft-06/schema#",
  "$id": "http://venomkb.org/schemas/venomseq-schema.json",
  "title": "VenomSeq data",
  "description": "Structured data corresponding to VenomSeq run on a
      single venom",
  "type": "object",
  "properties": {
    "experiment-description": {
      "type": "string"
    },
    "investigators": {
      "type": "array",
      "items": {
        "type": "string",
      }
    },
    "release-date": {
      "type": "string",
      "format": "date-time",
      "description": "The date on which these data were published"
    },
    "sequencing-platform": {
      "type": "string",
      "description": "Name of sequencing platform used to generate
         venom perturbation data (e.g., \"Illumina HiSeq X\")"
    },
    "reference-datasets": {
      "type": "array",
      "items": { "type": { "$ref": "#/definitions/reference-dataset"
          } }
    },
    "cell-types": {
      "type": "array",
      "items": { "type": { "$ref": "#/definitions/cell-type" } }
    },
    "venom": { "$ref": "#/definitions/venom" },
  },
  "definitions": {
    "reference-dataset": {
      "type": "object",
      "description": "Dataset used for comparison (e.g.,
         Connectivity Map data)",
      "properties": {
        "url": { "type": "string" },
        "name": { "type": "string" },
        "citation": { "type": "string" },
      }
    },
    "cell-type": {
      "type": "object",
      "properties": {
        "name": { "type": "string" },
        "species": {
          "type": "string",
          "description": "Cell type species of origin (usually human
              for VenomSeq)"
        },
        "morphology": { "type": "string" },
        "venomseq-data": { "type": { "$ref": "#/definitions/cell-
           type/venomseq-experiment" } }
      },
      "definitions": {
        "venomseq-data": {
          "type": "object",
          "properties": {
            "dosage": { "type": "number" },
            "dosage-unit": { "type": "string" },
            "genes-up": {
              "type": "array",
              "items": { "type": { "$ref": "#/definitions/cell-type/
                 venomseq-data/gene-dif" } }
            },
            "genes-down": {
              "type": "array",
              "items": { "type": { "$ref": "#/definitions/cell-type/
                 venomseq-data/gene-dif" } }
            },
            "raw-data": {
              "type": "string",
              "description": "URL linking to raw count data"
      }
     },
     "definitions": {
      "gene-dif": {
       "type": "object",
       "properties": {
        "entrez-gene-id": { "type": "number" },
        "base-mean": { "type": "number" },
        "log2-fold-change": { "type": "number" },
        "lfc-se": { "type": "number" },
        "test-stat": { "type": "number" },
        "pvalue": { "type": "number" },
        "padj": { "type": "number" },
        "symbol": { "type": "string" },
       }
      }
     }
    }
   }
  },
  "venom": {
   "type": "object",
   "properties": {
    "species": { "type": "string" },
    "common-name": { "type": "string" },
    "venomkb-id": {
     "type": "string",
     "description": "VenomKB identifier for the species from
       which this venom is derived"
    },
    "format": {
     "type": "string",
     "description": "E.g., lyophilized, fresh, frozen, etc.,
       with additional details like where stored and
       temperature"
    },
    "date-obtained": {
     "type": "string",
     "format": "date-time"
    }
   }
  }
 }
}
```

Appendix C.

Listing of algorithms

C.1. Connectivity analysis

Algorithm C.1.1 (*Connectivity score algorithm*). Given a query $\{q = (q_{up}, q_{down}) \,|\, q \subset \mathcal{G}\}$ and a reference database of gene-wise Z-scores $\mathbf{R} \in \mathbb{R}^{N \times M}$—where $\mathcal{G}$ is the set of all genes in the human genome, N is the number of expression signatures in $\mathbf{R}$, and M is the number of genes (or probe sets) reported in $\mathbf{R}$—computes *connectivity scores* that correspond to the signed enrichment of the genes in q versus the scores in $\mathbf{R}$. This algorithm is derived from the methods used by the Connectivity Map team, described in [234].

1. [Find $\mathbf{V}_{q\mathbf{r}}$.] Let $t_{\text{up}} = |q_{\text{up}}|$. Define $\mathbf{r}_i := \text{row}_i\,\mathbf{R}\,(i = 1, \ldots, N)$—the row of $\mathbf{R}$ corresponding to expression profile i. For each profile, construct a vector $\mathbf{V}_{q_{\text{up}}\mathbf{r}_i}$, which is the ranks of the members of q_{up} within $\mathbf{r}_i$:

$$\mathbf{V}_{q_{up}\mathbf{r}_i}[j] := \left\lfloor \frac{t_{up}}{M} \sum_{k=0}^{M} \mathbb{1}\left\{\mathbf{r}_i[k] \leq q_{up}[j]\right\} \right\rfloor$$

2. [Compute a and b.] For each i, compute the following two quantities:

$$a_{\text{up}} = \max_{j=1}^{t} \left[\frac{i}{t} - \frac{\mathbf{V}_{q_{\text{up}}\mathbf{r}_i}(j)}{N} \right]$$

$$b_{\text{up}} = \max_{j=1}^{t} \left[\frac{\mathbf{V}_{q_{\text{up}}\mathbf{r}_i}(j)}{N} - \frac{(j-1)}{t} \right]$$

3. [Compute ES.] For each i, compute the enrichment scores of the genes in q_{up} and q_{down}:

$$ES^{\text{up}}_{q\mathbf{r}_i} = \begin{cases} a_{\text{up}} & \text{if } a_{\text{up}} > b_{\text{up}} \\ -b_{\text{up}} & \text{if } a_{\text{up}} < b_{\text{up}} \end{cases}$$

4. [Repeat for “down” gene sets.] Do steps **1.** through **3.** again, substituting “down” for "up" in q, t, a, b, and ES. We now have both up and down enrichment scores between q and each of the signatures in $\mathbf{R}$.

5. [Compute WCSs.] For each i, find the weighted connectivity score between q and $\mathbf{r}_i$. We introduce sparsity by setting the connectivity score to 0 when the "up" and "down" enrichment scores do not have opposing signs.

$$w_{\mathbf{qr}_i} = \begin{cases} (ES^{\text{up}}_{\mathbf{qr}_i} - ES^{\text{down}}_{\mathbf{qr}_i})/2 & \text{if } \text{sgn}(ES^{\text{up}}_{\mathbf{qr}_i}) \neq \text{sgn}(ES^{\text{down}}_{\mathbf{qr}_i}) \\ 0 & \text{otherwise} \end{cases}$$

6. [Normalize WCSs.] To reduce bias, we normalize connectivity scores both by cell type and by perturbagen class (e.g., small molecule, genetic loss-of-function, genetic overexpression, etc.). Let $\text{sgn}(x)$ be the sign function $\frac{\mathrm{d}}{\mathrm{d}x}|x|$. μ^{+}_{ct} is the mean of all WCSs with the same cell type c and perturbagen type t as $w_{\mathbf{qr}_i}$ that are positive valued (and similarly for μ^{-}_{ct} with negative WCSs of the same cell and perturbagen type).

$$NCS_{\mathbf{qr}_i} = \begin{cases} w_{\mathbf{qr}_i}/\mu^{+}_{ct} & \text{if } \text{sgn}(w_{\mathbf{qr}_i}) > 0 \\ w_{\mathbf{qr}_i}/\mu^{-}_{ct} & \text{otherwise} \end{cases}$$

7. [Find τ.] τ is the signed connectivity, where the magnitude of a score represents the percentile of an NCS in the context of all other NCSs.

$$\tau_{\mathbf{qr}_i} = \text{sgn}(NCS_{\mathbf{qr}_i})\frac{100}{N}\sum_{j=1}^{N}[|NCS_{j\mathbf{r}_i}| < |NCS_{j\mathbf{r}_i}|]$$

∎

C.2. VIPER and aREA-3T[1]

Algorithm C.2.1 (*msVIPER*). Performs virtual inference of protein-activity by enriched regulon analysis as described by Alvarez *et. al.* in [4].

1. [Initialize.] Let `GES` be a matrix of gene expression, where rows represent genes (or probe sets) and columns represent samples. The matrix values can be either relative

[1] These two algorithms were written and originally described by members of the Califano Lab at Columbia University. I have included them here for completeness, but all credit goes to the original authors.

expression (e.g., fluorescence) or absolute expression (e.g., gene counts). Let `REGUL` be an adjacency matrix corresponding to a regulatory network for the cell type represented by `GES`, generated using the ARACNe algorithm [15]. The values of `REGUL` roughly correspond to measures of coregulation (based on mutual information) between pairs of genes.

2. [Create signatures.] For the perturbational state of interest p, perform Student's t-test on each row of `GES`, comparing columns corresponding to samples in p to columns representing "control" samples p^0 (i.e., unperturbed). Let `TSTAT` be the list of computed t-statistics and `PVAL` be the list of p-values.

3. [Normalize signatures.] Normalize the signature values by converting to estimated Z-scores. Let `QNORM()` be a function that converts a vector of numbers into their respective quantile scores (e.g., $F_X(x) := Pr(X \leq x)$).

$$\texttt{SIG} \leftarrow \texttt{QNORM}\left(\frac{\texttt{PVAL}}{2}\right) * \operatorname{sgn}(\texttt{TSTAT})$$

4. [Generate null model.] Perform step 3. 1000 times on a version of `GES` with columns (samples) randomly shuffled (producing a matrix with dimensions $|\texttt{SIG}| \times 1000$, with correlation structures between genes preserved). Store the result in matrix `NULL`. (If fewer than 5 samples are present, generate the null model by permuting rows [genes] instead.)

5. [Run `aREA`.] Perform the `aREA-3T` algorithm on `SIG`:

$$\texttt{RES} \leftarrow \texttt{aREA3T(SIG, REGUL)}$$

6. [Run `aREA` on null model.] Perform the `aREA-3T` algorithm on `NULL`:

$$\texttt{TMP} \leftarrow \texttt{aREA3T(NULL, REGUL)}$$

7. [Estimate statistical significance.] `RES` and `TMP` are vectors of regulon enrichment scores for the true data and the null model, respectively. Let $\texttt{RES[}i\texttt{]}$ be the enrichment score of reg_i in the observed data, and $\texttt{TMP[}i\texttt{,:]}$ be the vector of enrichment scores of reg_i under the null model. The p-value for each regulon reg_i is given by $\Pr\left(|\texttt{TMP[}i\texttt{,:]}| \geq |\texttt{RES[}i\texttt{]}|\right) \times 2$. Compute each p-value and store the result in a vector `PVAL`. Compute a vector of FDR-corrected p-values using the Benjamini-Hochberg procedure [20] (in the `R` programming language, this can be performed using the `p.adjust()` function), and store the result in a vector `FDR`.

8. [End.] Return `RES`, `PVAL`, and `FDR` to the user (each should be of the same length, and each index i corresponds to reg_i). ▮

Algorithm C.2.2 (*aREA-3T*). Performs analytic Rank-based Enrichment Analysis using a 3-tailed approach. Given a regulon object and an expression signature (see **Algorithm C.2.1**), tests whether the genes within each regulon shift towards the front or the back of the rank-sorted expression signature. The 3-tailed test separately computes 1-tailed and 2-tailed enrichment statistics and integrates the two values using the Mode of Regulation statistic (see [4] for further details).

1. [Initialize.] Let `SIG` be a gene expression signature of Z-scores and `REGUL` be a regulon, as described above.
2. [Make vectors of ranks.] Construct a vector of ranks—named `RANK2`—such that $\{\texttt{SIG}[\texttt{RANK2}[i]] \geq \texttt{SIG}[\texttt{RANK2}[i-1]] \quad \forall i \in [1, |\texttt{SIG}|]\}$. Similarly, construct another vector of ranks (`RANK1`) using the magnitudes (absolute values) of the Z-scores. These will be used to compute ES_2 and ES_1 below, respectively.
3. [Find 1-tailed enrichment.] For each regulon $\text{reg}_i \in \texttt{REGUL}$, compute an enrichment statistic ES_1 as the mean of the quantile scores of members of reg_i within `RANK1`.
4. [Find 2-tailed enrichment.] Perform step **3.** using `RANK2` instead of `RANK1`.
5. [Determine Mode of Regulation.] The Mode of Regulation is used to weight the contributions of ES_1 and ES_2 in the final enrichment score ES. Define three Gaussian random variables: G_1 (repressed targets), G_2 (activated targets), and G_3 (non-monotonically regulated targets), and estimate their parameters using whichever method is preferred[2].
6. [Integrate for 3-tailed enrichment.] The enrichment score for each regulon is the sum of ES_1 and ES_2 weighted by the magnitude of MoR:

$$\texttt{ES} \leftarrow |\text{MoR}|ES_2 + (1 - |\text{MoR}|)ES_1$$

▮

C.3. The Semantic API

Algorithm C.3.1 (*Semantic query*). Interprets a “semantic query” request submitted via the Semantic API.

[2] The authors of aREA-3T use the `mixtools` package for the R programming language.

1. **[Parse user query.]** Create an empty list CLASSES. Iterate over elements of select, declare, and aggregate in the query, and push any encountered ontology classes to the end of CLASSES. Delete duplicate entries from CLASSES. Set $k \leftarrow |\texttt{CLASSES}|$. If the query contains a declare key, create an empty list CONSTRAINTS and fill it by iterating over elementsof declare, parsing according to the template

$$\langle \mathit{class} \rangle \mid \langle \mathit{attribute} \rangle \mid \langle \mathit{operator} \rangle \mid \langle \mathit{value} \rangle$$

and appending the JSON object to the end of CONSTRAINTS. Copy the value for the query's select key to an object variable named SELECT.

2. **[Find subgraph.]** Let $C = (V, E)$ be the graph representing class relations in the OWL ontology of interest, and let $V' = \Big\{\{v_1', v_2', \ldots, v_k'\},\ V' \subset V\Big\}$ be the set of vertices corresponding to the members of CLASSES. Find the subgraph with the fewest possible edges[3] $C' = (V', E')$, which corresponds to the generalized distance $d(v_1', \ldots, v_k')$. When k is small, this can be approximated efficiently by finding the shortest paths between all pairs of nodes in V' and then taking the union of those paths.

3. **[Construct MATCH clause.]** Build a string MATCH by walking the nodes and edges of C' until all nodes and edges have been visited. Append (⟨a_i⟩:⟨CLASSES[i]⟩) at each node, and -[:⟨$\mathit{relation}_j$⟩]-> at each edge, where a_i is an alphanumeric variable name referring to class_i. Nodes and edges may be visited more than once, but a_i and CLASSES[i] must be reused consistently.

4. **[Construct WHERE clause.]** If CONSTRAINTS is defined, iterate over its elements, convert each element to a Cypher-formatted string (e.g., s.name contains 'Conus'), matching the variable name at the beginning of the string to the corresponding variable name in the MATCH clause for the ontology class of interest. Join each of these strings with the delimiter ' AND ', and append the result to the end of the string 'MATCH '. If CONSTRAINTS is not defined, do nothing.

5. **[Construct RETURN clause.]** Convert the value of SELECT to a string formatted as a Cypher RETURN clause, replacing ontology class names class_i with their respective variable names a_i.

6. **[Preprocess data aggregations.]** If the query contains a key aggregate, iterate over its keys and process them as defined for each aggregation function. For example, distinct: { class: 'a_i' } should be handled by inserting DISTINCT into the RETURN clause, between RETURN and a_i. Aggregations that are meant to be applied to the result of the Cypher query should be saved for later.

[3] Sometimes called the *Steiner tree*.

7. [Assemble Cypher query.] Create a new string `QUERY` by appending the values of `MATCH`, `WHERE`, and `RETURN`.

8. [Execute query.] Initiate a new transaction with the graph database server using the contents of `QUERY`. Retrieve the server's response to the query and store it in `RESULT`. Trim unnecessary metadata included in the server's response.

9. [Perform remaining aggregations.] If any aggregation functions (sorting, counting, etc.) remain, handle them as specified, and modify the contents of `RESULT` accordingly.

10. [Return results to user.] Send `RESULT` to the user. ▮

Appendix D.

Code availability

All code used in the experiments described within this dissertation is open source and publicly accessible. Most of the software I wrote is available from my GitHub profile, located at `https://github.com/JDRomano2`. This includes the web application and API code for VenomKB, the code used to produce and evaluate the Venom Ontology, and all code and data related to the analysis of `VenomSeq` results. All production-quality releases of pertinent code are mirrored on the Tatonetti Lab GitHub profile, which can be found at `https://github.com/tatonetti-lab`. Additionally, I have included Digital Object Identifiers (DOIs) in the dissertation text pointing to "frozen" versions of the different software packages as they existed at the time of publication (usually hosted on figshare). Some of the features related to the semantic API are not yet ready for deployment on the VenomKB website, but all progress is tracked on the VenomKB source repository and can be run locally. I am happy to provide additional data, documentation, or help as needed, and all questions pertaining to the code itself should be directed to me.