Computational Curation of Open Science Data

Maxim Grechkin

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2018

Reading Committee:

Bill Howe, Chair

Walter L. Ruzzo

Hoifung Poon

Program Authorized to Offer Degree:
Computer Science & Engineering

University of Washington

Abstract

Computational Curation of Open Science Data

Maxim Grechkin

Chair of the Supervisory Committee:
Associate Professor Bill Howe
Information School

Rapid advances in data collection, storage and processing technologies are driving a new, data-driven paradigm in science. In the life sciences, progress is driven by plummeting genome sequencing costs, opening up new fields of bioinformatics, genomics, and systems biology. The return on the enormous investments into the collection and storage of the data is hindered by a lack of curation, leaving significant portion of the data stagnant and underused. In this dissertation, we introduce several approaches aimed at making open scientific data accessible, valuable, and reusable.

First, in the Wide-Open project, we introduce a text mining system for detecting datasets that are referenced in published papers but are still kept private. After parsing over 1.5 million open access publications, Wide-Open has identified hundreds of datasets overdue for publication, 400 of them were then released within one week.

Second, we propose a machine learning system, EZLearn, for annotating scientific data into potentially thousands of classes without manual work required to provide training labels. EZLearn is based on an observation that in scientific domains, data samples often come with natural language descriptions meant for human consumption. We take advantage of those descriptions by introducing an auxiliary natural language processing system, training it together with the main classifier in a co-training fashion.

Third, we introduce Cedalion, a system that can capture scientific claims from papers,

validate them against the data associated with the paper, then generalize and adapt the claims to other relevant datasets in the repository to gather additional statistical evidence. We evaluated Cedalion by applying it to gene expression datasets, and producing reports summarizing the evidence for or against the claim based on the entirety of the collected knowledge in the repository. We find that the claim-based algorithms we propose outperform conventional data integration methods and achieve high accuracy against manually validated claims.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I am incredibly grateful to my adviser Bill Howe. Bill was an inspiration throughout the years at UW, helping set the research direction and always questioning me about the problems, methods, and approaches. I have enjoyed our brainstorming sessions and am grateful to Bill for broadening my horizons. Bill's interdisciplinary position gave me opportunities to meet people across campus and join DataLab at Information School. I wish to thank people in DataLab who have worked with me and provided helpful comments on my work including other Bill's students: Emily Furst, Dylan Hutchinson, Shrainik Jain, Dominik Moritz, and Luke Rodriguez. I am grateful to Po-shen Lee, Jevin West, and Sean Yang who introduced me to Viziometrics.

I have very warm memories about my internship at Microsoft Research, working with Hoifung Poon who became a good friend and adviser. I am deeply grateful to Hoifung for the support and guidance he provided over the following four years, working with me on almost every project I did since that time. I am thankful to Sumit Gulwani, who gave me a chance to work with an incredible PROSE team at MSR on program synthesis. I have learned a lot from Sumit.

Larry Ruzzo and David Ribes have my gratitude for serving on my committee and providing invaluable feedback.

My graduate journey began in computational biology, and I am indebted to Su-In Lee for guiding me during those years. Su-In, along with other Lee Lab members including Sonya Alexandrova, Safiye Celik, Peter Ney, PJ Velez were inspiring and I am especially grateful to Ben Logsdon and Scott Lundberg for the insightful discussions.

I am thankful to my undergraduate advisers Igor Menshov and Alexander Ilushin, who

have first introduced me to research and launched me on this wonderful path.

I would like to thank all of my collaborators, co-authors, friends, and colleagues who have helped me throughout my studies: Maryam Fazel, Andrew Gentles, Bernease Herman, Vu Le, Mark Marron, Mikal Mayer, Tyler McCormick, Alex Polozov, Jian Peng, Rishabh Singh, Gustavo Soares, Ben Taskar, Sheng Wang, Daniela Witten, and Benjamin Zorn.

Allen School is an amazing place for research and I have truly enjoyed my time here. I am deeply grateful to Lindsay Michimoto and Elise deGoede Dorough for the their support.

Finally, I'm deeply grateful to my friends and family who have supported me over the years. I would not have been here if it were not for your support.

Chapter 1

INTRODUCTION

Data-intensive paradigm in science

With the beginning of the 21st century, the data-intensive paradigm [113] has been transforming science. Oceanography studies are equipping ships with high-throughput sensors, collecting terabytes of data to be processed and analyzed back on shore (e.g. UW eScience Oceanography collaboration [230]). In Astronomy, projects like the Sloan Digital Sky Survey (SDSS) [69] are systematically scanning the night sky, generating hundreds of terabytes of high-resolution images of stars and galaxies. In Physics, experiments like the Laser Interferometer Gravitational-Wave Observatory (LIGO) [1] and Large Hadron Collider (LHC) are processing streams of hundreds of gigabytes of data per second, making automated decisions on what to keep for future analysis. Biology has been transformed by an explosion of sequencing and other bio-technologies. Less than 20 years after the completion of Human Genome Project, NCBI Sequence Read Archive (SRA) [154] stores over 10 PB of sequencing data.

All these data lead to specialization, which is a key attribute of the data-intensive paradigm: the scientists and engineers producing the data are no longer the same as those who are performing the analysis [113]. This separation leads to the fact that data needs to be shared or transferred from producers to consumers. According to Jim Gray, the separation means that the data-intensive scientific process can be decoupled into three components: capture, curation, and analysis [113]. Here, curation is not strictly necessary, but it is a common step that occurs in some situations as we will show later. Data and knowledge curation, which includes questions like data storage, quality control and discoverability (e.g., problems that arise after the data capture, but before the actual analysis takes place) is

going to be the main topic of this thesis.

According to Renée Miller, data curation includes all the processes needed for principled and controlled data creation, maintenance, and management, together with the capacity to add value to data [172]. A narrower definition of curation from BioCreAtIvE glossary states that the aim of curation is to "transform information contained in free text (scientific literature) to information stored in form of a structured database record." We will adopt a broader view and consider curation to be all activities that deal with organizing the data, performing quality control procedures, making the data discoverable and pre-processing it for easier digestion by a downstream analysts. The separation between the curator and analyst is the key abstraction here — multiple researchers can use the same dataset and so it is advantageous to avoid repeating the curation step. Furthermore, curation will often deal with making data useful for *future* uses, making the problem harder as it can be difficult to predict ahead of time what aspects of the data will be of value.

For the purposes of this thesis, we adopt the following working definition of data curation introduced by Cragin et al [49]: *Data curation is the active and on-going management of data through its lifecycle of interest and usefulness to scholarship, science, and education; curation activities enable data discovery and retrieval, maintain quality, add value, and provide for re-use over time.*

What are the circumstances that make the curation problem interesting for us? Projects like SDSS, LIGO, and LHC are predicated on exclusive access to special equipment – data capture is centralized (telescopes and particle accelerators are extremely expensive) and curation problem is solved at the source. This centralization makes it possible, for example, for SDSS to provide convenient access to its publicly released data, through specially developed API against consistent internal data. In contrast, Molecular Biology is extremely decentralized — hundreds or even thousands of labs around the world can afford to have their own data capture equipment, develop their own measurement protocols, and collect data about incredibly diverse biological questions. This diversity and decentralization exacerbates the curation problem — even though all the data can be stored in the public centralized repos-

itory (e.g., NCBI Gene Expression Omnibus (GEO)), metadata standards are lacking and each submitted dataset is unique in its own way. *Semantic heterogeneity* arising in these scenarios is the main focus of several projects in this thesis (see Chapter 4).

The main goal of improving curation is making *productive* data sharing and reuse easier, but why is that important? Specifically, in this context we understand data sharing to mean transfer of data (or access to data) between research groups, potentially located in different universities and even countries. Data sharing activities within a single research group is excluded as it is presumably facilitated by in-person discussions and explanations reducing the need for curation activities. Unlike intra-lab data sharing, public data sharing allows people unaffiliated with data produces to perform secondary analyses and reproduce primary results. Both of these aspects are crucially important: reproducing the original findings increases the overall quality of science being done, helps catch errors earlier and disseminates knowledge about the methodology. Secondary analysis opportunities are even broader: publicly shared datasets make studies possible that were not possible before (e.g., by combining two datasets from different competing labs), democratize access to scientific research opportunities (e.g., people with access to modest commodity computational resources can now advance scientific knowledge, without access to expensive data producing labs) and spur advances in processing algorithms by people who would not have considered studying them.

If we accept that curation is important, who should be responsible for the curation process in a decentralized model employed by molecular biology? From the data capture side, producers are numerous and getting them all to agree to the same standards and workflow practices might be challenging (see Section 2.5 for discussion of workflow management systems). On the analysis side, analysts are expected to use multiple disparate datasets and it is inevitable that the same datasets are going to be reused time and time again, making duplication of efforts on the analyst side an undesirable effect of any system that will push curation to the analysts. In this thesis, we argue that a third-party, running centralized curation services at the level of the entire repository, is the right way to approach this problem.

Thesis statement: *Public centralized scientific data repositories enable data reuse and secondary analyses. Machine learning techniques, deployed as repository-scale curation services, can improve data reuse, facilitate data discovery, enforce policy compliance, and enable reproducibility.*

Public repositories of transcriptomics data and curation problem

An increasing amount of data is being produced by researchers in life science every year. A significant portion of this data is ultimately ending up in public large-scale repositories like NCBI Gene Expression Omnibus (GEO). Established in 2000, GEO contains over 90 thousand datasets collected in Series (GEO Series, with accession numbers[1] starting with "GSE") with over 2 million processed samples as of February 2018. While the datasets submitted to GEO have to be "MIAME-compliant", it doesn't mean that they are computationally discoverable or curated. The MIAME standard (Minimum Information About a Microarray Experiment, [28]) does not require that the annotations be machine-readable or use a controlled vocabulary, making large-scale computational studies challenging. GEO curators also manually assemble submitted Series into Datasets (accession numbers starting with "GDS") that provide comparison and analytics tools called GEO Profiles. Unfortunately, this curation process is struggling to catch up with an explosive growth of datasets being submitted to GEO (see Figure 1.1a).

As a part of project called *Wide-Open* (see Chapter 3), we have processed 1.5 million papers from PubMed Central Open Access dataset attempting to asses usage of data from GEO repository. In addition to general information (e.g. title, journal, publication date) we have extracted a list of GEO datasets used by each paper using a regular expression search for `GSE[0-9]+` regex. This approach allows us to estimate usage data of GEO datasets by open-acess publications.

We observe (Figure 1.1b) that there has been an increase in the number of *integrative*

[1] an *accession number* is a unique identifier, associated with a record in the public data repository, such as GEO.

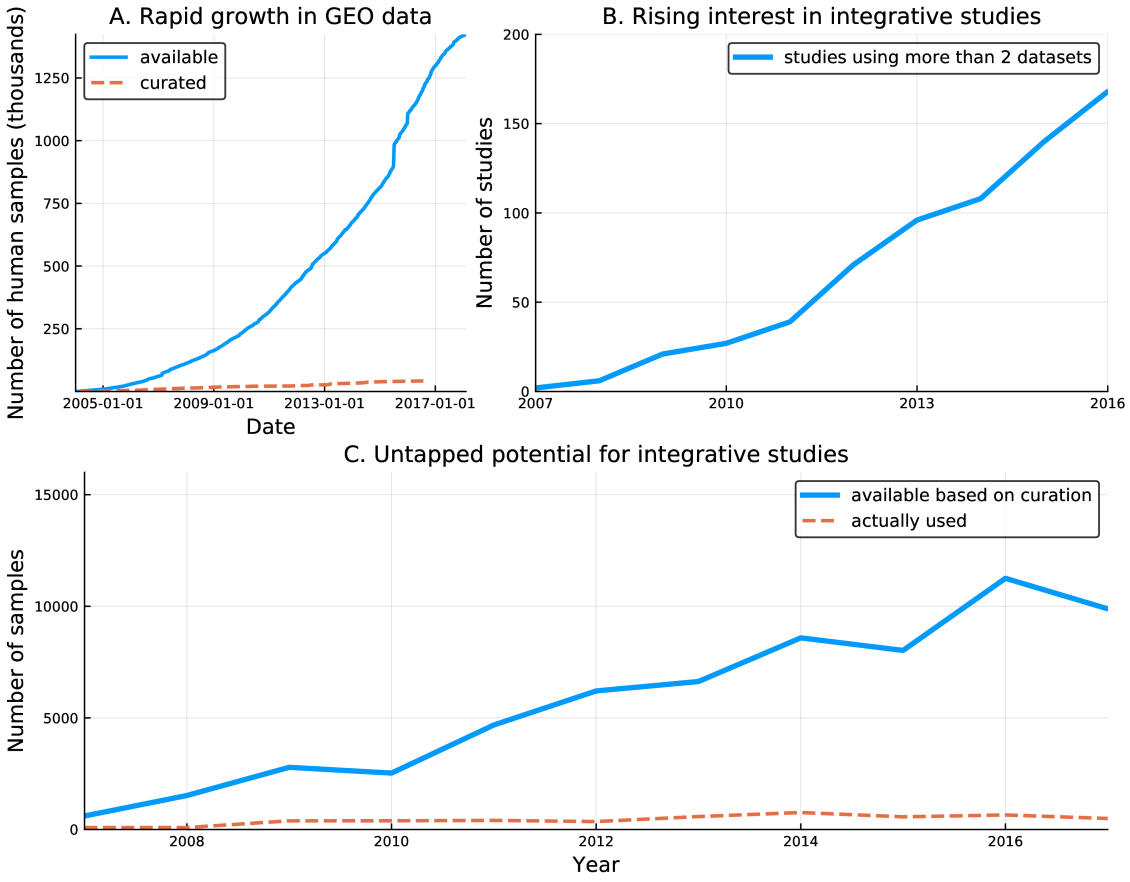


Figure 1.1: a) Total number of human samples submitted and curated in GEO. Curation is defined as being assembled into Geo DataSets (GDSs). b) Number of integrative studies found in the PubMed Centeral corpus of open access papers plotted over years. Only studies using datasets from Affymetrix U133 Plus 2.0 platform were considered. c) The median number of samples used by a integrative study using data from Affymetrix U133 Plus 2.0 platform vs number of available samples with matching tissue (based on *EZLearn* tissue labels).

papers (we define them as those that reference at least 3 different GEO Series). This increae is an encouraging finding — one of the goals of GEO was to facilitate research collaborations and data reuse, and integrative papers are one indicator of such reuse taking place. In contrast, when we look at the median number of samples used by integrative papers, we notice a rather slow uptake of data by researchers. Comparing this slow growth with a general trend of an increasing number of samples available every year, our finding suggests that there might be some factor that is inhibiting researchers from using more of these data.

There might be several potential reasons for this poor reuse:

- *Statistical challenges related to integrating multiple datasets.* Aggregating data from multiple sources presents a multitude of statistical challenges: different microarray platforms can have different numbers of probes, different platforms might provide results on different scales (e.g. RNA-Seq vs microarrays), different experimental conditions might perturb the gene expression in different ways, batch effects related to different labs processing samples, or data required for analysis might be simple missing (e.g. another microarray platform is not sensitive enough to detect splice variants). All of those are real challenges that, in our opinion, highlight the need for better statistical tools in our toolbox. Computational biology and bioinformatics community has already developed methods to deal with batch effects [126], approaches to deal with different platform scales [192] and even different numbers of probes among them [39].

- *Shift of research interests over time.* Perhaps older datasets are just no longer interesting to researchers? However, a simple statistic suggests that this might not be the case: over 35% of the third-party[2] datasets referenced by open access papers since 2013 are five years old or older. Additionally, we would argue that at the very least, control samples from older studies remain valuable as controls, provided there are researchers can easily find those extra samples and incorporate them in the study using appropriate

[2]defined here as those that were released at least two months before the paper was published, a conservative definition

statistical tools.

- *Lack of curation.* It is challenging to estimate the impact of curation on data reuse, but rough estimates (comparing number of re-uses of the series that are part of a curated dataset vs stand-alone ones[3]) and observational studies [194] suggest that curation does indeed help increase reuse probability.

We have attempted to estimate the potential effect of curation using the following experiment: using tissue annotations from *EZLearn* project (Chapter 4), we have estimated an upper bound of the number of possible samples that each integrative paper could have used then compared to what they have actually used (Figure 1.1c). The estimate is based on the number of samples available in the GEO repository annotated with the same tissue type as was used in the paper *and* published earlier than the most recent dataset referenced by the paper. That is, later datasets could not have been used by the author, and datasets with different tissue type were not relevant. While there are multiple reasons why this is an upper bound (e.g. some of the tissue-matched samples might be unusable due to their experimental condition or data quality issues), it is instructive to observe that the number of datasets that are used is not growing nearly as fast as our estimated upper bound. This suggests that curation might indeed be a bottleneck - more data is readily available in the repository than is used by integrative papers, motivating our thrust of work in the direction of data curation.

Reproducibility efforts

Reproducibility and replication are issues of great importance in science, and have motivated significant work on the topic. Larson and Sandberg [146] found in 2006 that less than half of the samples submitted to GEO had raw data attached to them that satisfied basic quality control metrics. A significant increase in the number of samples submitted to GEO since that

[3] a caveat here is that datasets chosen for curation might be more "interesting" to begin with

time suggests that absence of data might no longer be a limiting factor. Out of 125 papers submitted to Biostatistics from July 2009 to July 2011, only 21 contained corresponding source code or data and 5 have passed an optional reproducibility review offered by the journal editors [190]. An attempt [124] to computationally reproduce 18 studies from Nature Genetics was successful in reproducing only 8 results (two of them were "reproduced in principle" and six more were "reproduced partially or with some discrepancies"). While half of the failed reproductions were due to outright missing data, another half was due to methods not being described in enough detail or missing software. A case study attempting to reproduce a computational biology paper [83] estimated that it took a novice user 280 hours to reproduce results of the paper while having access to the first author of the paper under consideration to ask questions. While one could envision a completely automated computational solution for such in-depth reproducibility (likely an AI-complete problem), it is prudent to have higher-level screening systems in place that would check the plausibility of the claims based on the provided data and cross-check them against other datasets available in the repository.

Several recent projects [191, 107] are attempting to summarize off-the-shelf methods for assisting with producing reproducible research (e.g., containers, source code version control, data provenance tools), but those efforts face issues similar to workflow tracking systems. The database community is also actively involved (e.g., Computational Reproducibility tutorial at SIGMOD 2012 [77]).

In this dissertation, we propose a system (Cedalion[4]) that provides a way to express and execute data-driven claims from research papers (e.g. gene X's expression is higher in group A vs group B) against their source datasets. Furthermore, in some circumstances Cedalion can even *generalize* the claim, finding other related dataset from the GEO repository (leveraging our own tissue annotation from *EZLearn* project) and automatically validating the claim in them. We believe that a system like Cedalion that can both help with assessing

[4]In Greek mythology, Cedalion was a servant of Hephaestus [65].

reproducibility of old studies and help make new studies more reproducible from the start by encoding the claims made in a formal language is useful.

Outline

The rest of this dissertation is organized as follows:

- Chapter 2 will provide an overview of data curation and exploration efforts related to the field of genomics.

- Chapter 3 will introduce *Wide-Open*, a text mining system we use to track data citations in reuse in open access literature. In addition to elucidating the patterns of data reuse, *Wide-Open* also allows us to track datasets that are "overdue" for publication — they are referenced from a published papers, but are not made available themselves.

- Chapter 4 will introduce *EZLearn*, a general data curation framework leveraging a concept of *organic supervision* to produce high-quality data annotations in scientific domains. We will demonstrate that *EZLearn* outperforms prior state of the art supervised approaches without using any manually labeled data.

- In Chapters 5 and 6 we will explore two computational approaches for analyzing genomics datasets obtained by *Wide-Open* and curated by *EZLearn*. Specifically, we will introduce DISCERN — a method for detecting differentially regulated genes based on data samples gathered from two conditions (e.g. cancer and normal) and Pathway Graphical Lasso — a method for reconstructing gene regulatory networks using additional "pathway assumption."

- Chapter 7 will introduce Cedalion, a system that allows interested parties to express scientific claims from papers, validate them against the data associated with the paper, then generalize and adapt the claims to other relevant datasets in the repository to gather additional statistical evidence.

- We conclude in Chapter 8 by discussing limitations of current work and looking forward at potential next steps.

Chapter 2

RELATED WORK

Data curation spans wide range of different methods and approaches, both general-purpose and domain-specific. In this section, we will review prominent related work in general data curation field, taking special interest in the approaches that deal with curating gene expression data. We will then discuss curation approaches specifically aimed at scientific data, including tools for knowledge discovery and workflow management.

2.1 General data curation

Knowledge base construction (KBC) is the process of populating a knowledge base with information extracted from documents and structured sources [206]. Some data curation tasks can be viewed as a special case of KBC, where, for example, a tissue curation system for GEO will collect a knowledge base of facts of the form (sample, tissue). Furthermore, KBC can be viewed as part of the curation process aimed at increasing the value of the raw data by constructing semantically meaningful knowledge bases on top of it (e.g., Literome [196] project curated abstracts of scientific papers by extracting implied facts from them).

DeepDive [182] is a hybrid system for Knowledge Base Construction and Data Curation that leverages advances from fields of database research and machine learning. DeepDive democratizes the use of *distant supervision* [173] by making it easy for users to write feature extractors (functions that extract information from raw data, e.g. extracting entities from the document, or extracting a phrase from a sentence that is between two entities) and supervision rules (e.g. if FreeBase mentions some fact, then there is a decent probability it is true, but lack of mention doesn't prove anything).

Shin et al [219], authors describe methods for speeding up incremental KBC in the

DeepDive framework. They employ both standard database techniques (incremental view maintenance using DRed algorithm [101]) and novel approaches for the inference stage of the DeepDive system: 1) They precompute a number of preliminary samples ahead of time, that are then used in a Metropolis-Hastings scheme to get a correct distribution. 2) Once the precomputed samples are exhausted, they employ variational approach. Specifically, they draw a number of samples from the original factor graph, construct the covariance matrix and then solve the inverse covariance estimation problem [9] to get a sparse approximation, producing a reduced factor graph that is then used for the direct inference.

Ratner et al [205] describe the feature construction portion of the DeepDive-like system. Data Programming is a paradigm that allows analysts to define a set of noisy labeling functions (heuristics) that can each annotate a portion of the dataset of interest. An inference algorithm is then run on the data to simultaneously learn confidence in each labeling function (and structure between them) and produce a final 'consensus' labeling of the dataset. One of the most interesting aspects of this work is ability to handle explicit hints about dependencies among labeling functions. The system supports *similar*, *fixing*, *reinforcing*, and *exclusive* dependency predicates that are then converted into factors in the labeling function factor graph. While promising, data programming also has limitations: some of the datasets used as examples in the paper required over a hundred labeling functions, and the example that used just 7 functions had only 7% coverage of the dataset.

Clearly, a more unsupervised approach would reduce the effort needed for curation, but it is very challenging to achieve in a general case. In this work, we will present a method that takes advantage of multiple modalities that are sometimes available in the data. Specifically in the case of gene expression, we will use both raw gene expression data and human provided text descriptions to derive the final labels.

A different approach to data curation is presented in [227]. Data Tamer is a system for scalable, incremental data curation in an enterprise setting, with an input from a large group of non-programmer domain experts (DEs). In Data Tamer, an administrator (DTA) designates a set of input *sites* (collections of records with key-value pairs) and wants to have

a combined relational database at the end with a pre-defined or learned schema. The paper considers several use-cases ranging from a fixed output schema in a health services application, to a completely free-form web aggregated data without any predefined schema. Major tasks performed by Data Tamer are deduplication and attribute mapping. Deduplication is achieved by training a Naïve Bayes classifier to detect duplicate pairs, followed by clustering of duplicate entities. Data Tamer also maps attributes from different sites into a common schema by leveraging a set of pluggable experts (four built-in experts are fuzzy string matching, TF-IDF cosine similarity between data values, Jaccard similarity for categorical valued columns, and Welch's t-test for real-valued ones).

A separate category of methods is being developed for *data cleaning* [45]. Systems like ActiveClean [142] provide tools for data scientists to actively inspect their data while building machine learning models to get rid of or repair *dirty* data (e.g. outliers, missing data). Data cleaning tools are a necessary and important step in the pipeline that sits above data curation — cleaning tools expect to get a labeled dataset, while curation system like *EZLearn* and DeepDive are aimed at producing those datasets. At the same time, cleaning tasks might be very domain dependent. In biology, for example, there is a wide array of tools for quality control of data produced by microarray [253] and RNA-Seq experiments [128].

2.2 Curating Gene Expression data

Large-scale independent microarray studies [163, 221, 179, 51, 111, 144] suggest that there is a growing interest in understanding tissue-specific or subtype-specific (e.g., in case of cancer) differences among biological samples. On the other hand, there has been a limited number of integrative studies on the topic [114, 71, 97, 212], highlighting the need for curation.

A prominent prior work on the topic of data curation, as applied to gene expression datasets is work by Lee at al[153], which introduced Unveiling RNA Sample Annotation (URSA). URSA a supervised classifier method, meaning that it requires labeled training data to create a model. In their work, Lee at al have manually constructed an annotated dataset of over 14 thousand samples they have used for training URSA.

URSA method consists of two parts: a support vector machine (SVM) one-vs-all classifier that predicts probabilities of the sample being part of a particular tissue and a Bayesian network-based correction (BNC) [12]. An SVM [29] is a discriminative classifier [156] that takes as input positive and negative examples of particular tissue type (selected in an ontology-aware manner) and finds a linear decision boundary that the maximizes margin between the two classes. An ontology-aware sample selection is done by considering all descendants of a particular term as positive examples, excluding all ancestors and then considering all the remaining samples as negative examples. An SVM trained with this procedure will be a one-vs-all SVM — it will be able discern if a particular sample is, say *leukocyte*, but will not be able to provide a correct label if the answer is negative. Thus, a separate SVM needs to be trained for every term we want to distinguish.

A Bayesian Network Correction is done to promote coherence between predictions of all the term-specific SVMs. For example, if an SVM classifier responsible for *leukocyte* gives 0.7 probability of the *leukocyte* label being correct, classifier for *blood* is reporting 0.9, and classifier for *liver* is also at 0.7, then BNC will prefer to choose *leukocyte*, as it is better supported by predictions for the related tissues (*blood* in this example). BNC works by constructing a Bayesian network that mirrors the ontology tree structure (with the *whole body* node at the top, and specific tissue types as leaf nodes at the bottom) and adding an observed node for each tissue representing an output from the SVM classifier. With this construction, latent variable nodes represent true label, which are inferred using standard BN inference methods [148].

One major shortcoming of the URSA method is that it is a supervised method that requires labeled training data. Manually labeling data is a costly process, especially in the case of scientific data curation, where some non-trivial level of domain knowledge is required and the task cannot be easily outsourced to a crowdsourcing service like Amazon Mechanical Turk. Additionally, based on empirical observations, URSA tends to very confidently predict general terms (e.g. *blood*).

	GEO search		*EZLearn*	
Term	Precision	Recall	Precision	Recall
”liver”	40%	57%	97%	89%
”kidney”	59%	100%	100%	90%

Table 2.1: Comparison of GEO search vs *EZLearn* as validated against manually curated samples from [153]

2.3 Repository access tools

NCBI provides the authoritative tool for querying GEO repository[1] that consists of a web interface as well as an API. While comprehensive (all data available in GEO is by definition exposed through the official interfaces), it lacks support for basic semantic search functions. For example, if you want to search for microarray expression data from a specific human tissue, then you would use a query similar to this one: `liver[All Fields] AND "Homo sapiens"[Organism] AND GPL570[All Fields] AND ("gsm"[Filter])`. This query would execute a full-text search across description fields of all Human samples from Affymetrix U133 Plus 2.0 platform (GPL570). Using *EZLearn* on the other hand, one can just query tissue annotations directly, obtaining a precise set of samples curated to a specific tissue. Table 2.1 demonstrates that *EZLearn* is far superior in terms of both precision/recall compared to regular GEO search.

If one wants to do an analysis of GEO data, then there are two interfaces available: GEO Profiles and GEO2R. GEO Profiles is an integrated tool based on manual curation efforts undertaken by GEO curators. It is operating over processed GEO data taking advantage of manual annotation, so features like “compare expression of gene X between smokers and nonsmokers in dataset Y” are readily available. The downside of GEO Profiles is that it can only operate over limited subset of GEO data that was manually processed by GEO curators.

[1] https://www.ncbi.nlm.nih.gov/geo/

GEO2R on the other hand is a tool that allows users to explore non-curated datasets: users are invited to manually split a dataset into sections of interest, and run some analysis that are available in R package *limma*. Under the hood, GEO2R is constructing and executing an R script that actually fetches the data and performs the analysis, hence the name.

In 2008, Liu et al [159] have introduced TiGER — a database of tissue specific expression and regulation. TiGER was based on NCBI EST dataset containing data from just 30 tissues and allows easy visualization of things like observed tissue-specific gene expression and predicted tissue-specific transcription factor (TF) interactions.

2.4 Hypothesis Generation and Discovery

Public repositories of diverse data like GEO make the idea of automating science seem to be within reach. Just like a human researcher, an AI system can gain the partial understanding of the world based on publicly available data (not to mention reading the published papers), formulate a set of new hypothesis on the boundary of this knowledge and execute experiments (physical or computational) to prove or disprove them. In fact, over a decade ago, a robot scientist named Adam [137] was already on the way to achieving just this goal. Adam was capable of formulating new hypothesis and carrying out wet-lab experiments to validate them, making the first novel scientific discovery in 2009 [136]. While Adam was specialized to the specific field (yeast functional genomics), a system that would rely on computational experiments or an interface with a cloud experiment provider [108] can potentially explore a much wider variety of research directions, while being completely confined to a computational agent.

Duggan et al. [65] describe Haphaestus, a system that can potentially allow scientists to encode their data-intensive research directions as *virtual experiments* (VE). Scientific hypotheses, expressed as VEs, can then be executed against the public data repository, with Haphaestus taking care of performing the statistical analysis and deciding on the optimal plan for executing the expressed query. Multiple challenges remain, two of the major ones being curation quality and p-hacking concerns. Similar concerns will be applicable to many

automated scientific discovery systems [225], so it is prudent to look into each separately.

First, the current state of data curation in biomedicine might be lacking for such systems to be deployed at the repository-scale. As shown in the previous section, simply querying the repository for samples from a particular tissue type already gives results of poor quality. Tissue type is going to be just a small part of a much broader experimental covariate (e.g, using Experimental Factor Ontology [165]) curation system that will be needed to address this issue. EZLearn (Chapter 4) can play a role in helping solve at least part of this curation challenge.

Second, any system that would provide large degree of automation for hypothesis mining will be susceptible to p-hacking issues [123, 263]. P-hacking, or data dredging [223] is commonly shunned, but widely used practice of tourturing the data until it confesses [121]. Any algorithm or scientist evaluating a large number of hypotheses using statistical tests would invariably find something "significant" just by random chance. If one would run millions of tests, number of these *false discoveries* can far outstrip number of real things. Developing approaches to bound false discovery rate is a fruitful area of research in statistics, but in practical matters, it is prudent to control number of *baseless* hypothesis tested. Baseless is an important distinction here, as our prior level of confidence in a set of hypotheses is strongly related to our false discovery rate [123]. Recognizing this potential issue, we will focus only on claims made by published papers in our discussion of automatic claim execution in Chapter 7.

2.4.1 Hypothesis Discovery Tools in Genomics

Several systems were recently proposed in the functional genomics space, implemented as Shiny R applications, but they all suffer limitations due to the heterogeneous and unstructured nature of the datasets in GEO. ScanGEO [140] is limited to manually curated portion of GEO, comprising of less than 10% of all samples. ShinyGEO [66] allows one to analyze any available dataset but is limited to one dataset at the time analysis and requires users to provide exact mapping of their query to the underlying schema of the data through a

GUI interface. GEOracle [58] uses machine learning and text-mining techniques to extract perturbation experiments from GEO datasets and visualize the results of the differential expression. None of the approaches support automatic extension of an analysis to related data or work directly from the scientific claim to formulate the data-analysis hypothesis.

2.5 Workflow systems as an alternative to repository curation services

A possibly attractive solution to the problem of data curation would be to encourage or require everyone to use a standardized set of tools to capture their data processing pipelines, providing curation as a by-product as every step will be tracked in such a *workflow system* [33, 183, 10, 207]. For example, Synapse system by Sage Bionetworks "allows researchers to share and describe data, analyses, and other content" by providing them tools to "describe these data or analyses, where they come from, and how to use them." Tracking each data sample from it's creation through all transformation until the final product will surely give us better understanding of what is going on in the dataset, but what are the downsides of such systems?

First, of course, is an increased burden on the users — workflow systems by definition will disrupt the usual analysis workflow. In a world where data sharing is not yet universally practiced [194, 112, 72], imposing additional barriers can be detrimental. In contrast, in this thesis, we emphasize server-side tools that operate repository-wide to support curation and reproducibility. This is a conscious trade-off — we believe that getting as much of the "dirty" data into the repository as possible using carrots (with automatic curation after the fact) is better than trying to get "clean" data with a stick.

Second, curation is inherently forward-looking process, trying to optimize utility of the data for *future* use cases. It is thus difficult to predict what annotations are going to be needed, making a system that tries to capture as much information as possible pushing into the regime of diminishing returns. On the other hand, a system like EZLearn (Chapter 4) that processes raw unstructured data without relying on manually provided labels can tolerate *ontology drift* that is naturally expected of bleeding edge scientific endeavours.

2.6 Summary

The methods presented in this thesis will be set apart from prior work along the following dimensions: In *EZLearn* (Chapter 4) we will demonstrate that the curation can be a repository-side service, working without manual supervision or altering the workflow of data providers in any way. In *Wide-Open* (Chapter 3) we will reinforce this idea, showing that policy compliance can be enforced at the repository level as well. In Cedalion (Chapter 7) we will demonstrate that a way to avoid p-hacking concerns during hypothesis processing is to be grounded in the claims already made in the published papers by starting with addressing reproducibility issues first.

Chapter 3

ENFORCING OPEN DATA POLICY COMPLIANCE USING TEXT-MINING

3.1 Tracking dataset references in open access literature

Advances in sequencing and other bio-technologies have led to an explosion of biological data. Figure 3.1 shows the remarkable growth in the number of gene expression samples in the NCBI Gene Expression Omnibus (GEO) repository [47]. As of February 2017, GEO contains 80,985 public datasets and 2,097,543 samples, spanning hundreds of tissue types in thousands of organisms. Making such a wealth of data publicly available not only facilitates replication, but also generates new opportunities for discovery by jointly analyzing multiple datasets [209].

Consequently, journals and repositories have increasingly embraced the open-data policies. For example, PLoS journals require authors to "make all data underlying the findings described in their manuscript fully available without restriction" [24]. GEO requests that authors should inform them "as soon as your manuscript is published so that we can release your records and link them with PubMed". Enforcing such policies, however, largely relies on manual efforts. Authors often forget to notify repositories when their papers get published. Repositories such as GEO resort to periodically checking private datasets to determine if they should be released, and call upon users to notify them of overdue ones. Still, the lag between the date the paper is published and the date the data is released is significant and appears to grow over time.

To help address the opportunity cost of this "hidden data", and to reduce the burden of manually keeping track of the release process for authors and repository administrators, we developed *Wide-Open*, a general approach that applies text mining to automatically detect

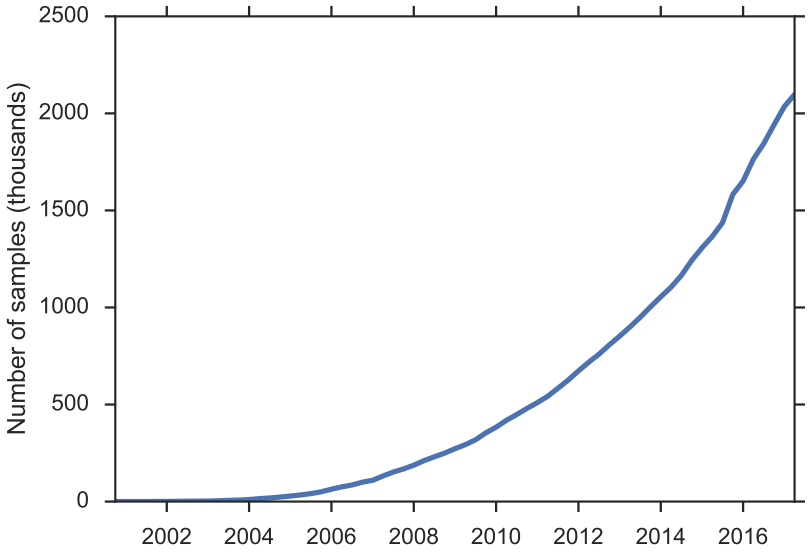


Figure 3.1: Number of samples in the NCBI Gene Expression Omnibus (GEO).

overdue datasets in a public repository. *Wide-Open* first scans PubMed articles for dataset unique identifiers (UIDs) using regular expressions. It then determines the validity of each candidate UID, and for valid UIDs, whether the corresponding datasets have been released. To determine if the dataset has been released, *Wide-Open* calls the repository's Web API for accessing datasets and searches for signature textual patterns in the query result. When there exists a database that indexes many publicly released datasets, *Wide-Open* will first check the UIDs using the database to minimize unnecessary Web API calls.

To evaluate the effectiveness of this approach, we applied it to two popular NCBI repositories: Gene Expression Omnibus (GEO) and Sequence Read Archive (SRA). To scan PubMed text for accession numbers, *Wide-Open* uses the regular expression `GSE[0-9]+` for GEO, and `SRX[0-9]+` for SRA. For each candidate accession number, *Wide-Open* first checks GEOmetadb [266] for GEO, and SRAdb [267] for SRA. A hit means that the dataset has been released. If not, *Wide-Open* calls the Web APIs for GEO (`https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=<accession>`) and SRA (`https://www.ncbi.nlm.nih.gov/sra/?term=<accession>`). The resulting page will then be parsed to determine if the accession

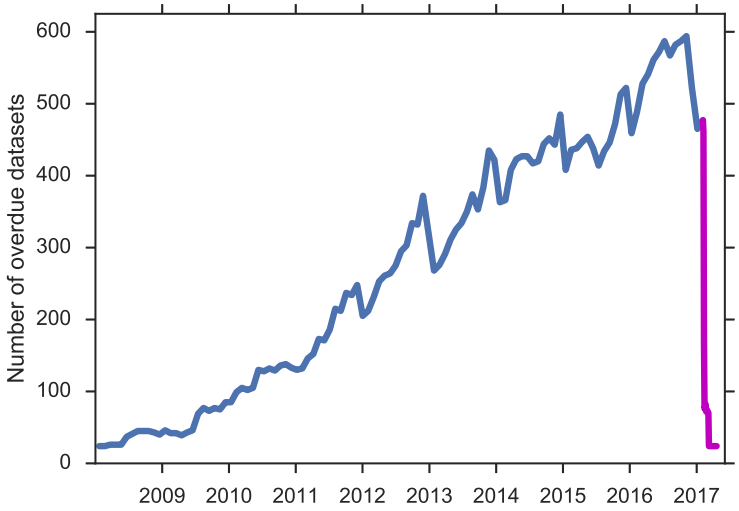


Figure 3.2: Number of GEO datasets overdue for release over time, as detected by *Wide-Open*. We notified GEO of the standing list in February 2017, which led to the dramatic drop of overdue datasets (magenta portion), with four hundred datasets released within the first week.

number is valid, and if so, whether the dataset is public or private. In the last case, the dataset remains private after being cited in a published article, which means that it is most likely to be overdue.

Specifically, for GEO, *Wide-Open* looks for strings such as "Could not find a public or private accession," which signifies an invalid accession number, as well as strings such as "is currently private," which signifies that the dataset is private. For SRA, the process is similar. The details can be found in our open-sourced code.

Wide-Open identified a large number of overdue datasets in GEO and SRA. Figure 3.2 shows the number of overdue GEO datasets over time. For each time point, we show the number of datasets referenced in prior publications but not yet released at the time of publishing. Notwithstanding some fluctuation, the number has been steadily rising since

the advent of next-gen sequencing. The oldest paper that references an overdue dataset was published in 2010. We have notified GEO of the overdue datasets *Wide-Open* had identified in February 2017. We received a prompt acknowledgement and noticed a dramatic drop in the number shortly after our exchange (the magenta portion; about 400 datasets were released within the first week). We applaud the quick action by GEO, and take this response as a promising sign that an automatic monitoring system like *Wide-Open* could help accelerate the release process. Out of the 473 datasets identified by *Wide-Open* on February 2017, 455 have been released by GEO since. Of the remaining eighteen candidates, only one is a true precision error (the accession number candidate GSE17200 actually refers to a soil name). Among the other seventeen cases, fourteen were identified due to typos by the authors who cited a wrong accession number, while the remaining three were legitimate datasets that could not be released either due to incomplete submission or privacy issues. In other words, *Wide-Open* attained a precision of 97%, even with author errors considered.

Wide-Open identified 84 overdue SRA datasets, as of March 2017. Next, we plan to contact SRA and work with them on verification and release of these datasets as well.

The time lag between submission and release has also steadily risen (Figure 3.3). GEO datasets that became public in 2006 took an average of 87 days from submission to release, whereas in 2016 the average delay was over 8 months. GSE2436 was submitted to GEO in March 2005 and was not made public until November 2016, an 11-year wait. While longer reviewing cycles might explain part of this increase[106], it seems clear that the rapid growth in the number of datasets would tax the manual release process and ultimately make it unsustainable.

While the initial progress is promising, much remains to be done. We need full text access to identify published datasets, which limits our current monitoring to the open access subset of PubMed Central (PMC). As of February 2017, this subset contains about 1.5 million papers, which is a small subset of PMC (4.2 million) and a fraction of PubMed (26 million). There are various ways to substantially increase the number of full-text articles for monitoring, thanks to the open-access movement. Publishers are increasingly open to grant-

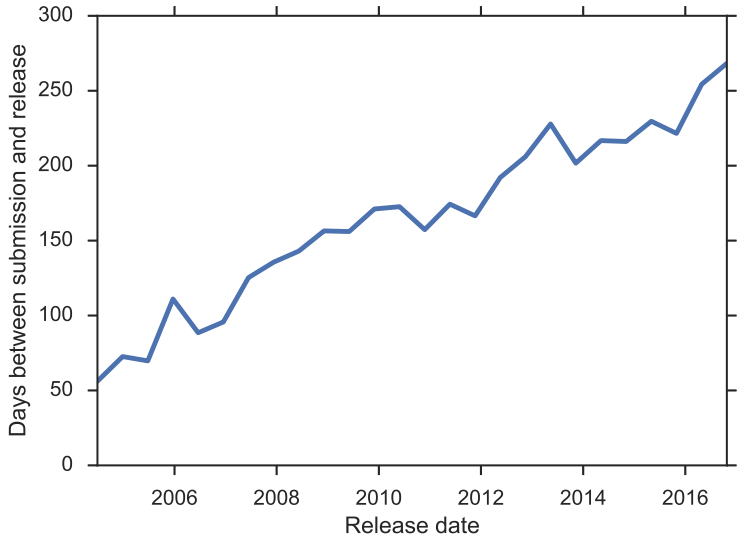


Figure 3.3: Average delay from submission to release in GEO.

ing text-mining licenses (see, e.g. http://text.soe.ucsc.edu/progress.html). Through our collaborators, we begin to have access to many more full-text articles on which we plan to run *Wide-Open* next. The number of private datasets is rather large. For example, GEO currently has over 10,000 datasets that remain private. We expect that many more overdue datasets could be identified with access to additional full-text articles.

Extending *Wide-Open* to a new repository consists of three simple tasks: creating regular expressions for dataset identifiers, identifying the Web API for dataset access, and adapting the query-result parser to distinguish between invalid IDs, datasets that have been released, and datasets that remain private.

3.2 Summary

In this chapter we have introduced *Wide-Open*, a text-mining system for detecting dataset references from full texts of open access papers and tracking their publication status. *Wide-Open* has been scanning papers from PubMed Center since February 2017, detecting hundreds

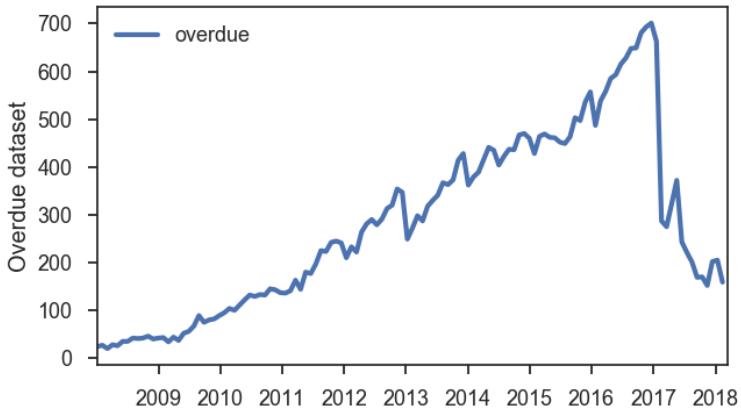


Figure 3.4: *Wide-Open* tracking result from February 2018, showing an initial drop in the number of overdue datasets, together with newly discovered ones.

of overdue datasets in GEO and SRA repositories. An updated version of the tracking plot is shown on Figure 3.4, demonstrating that *Wide-Open* is still actively detecting new overdue datasets.

The work in this chapter was first reported in [95]. *Wide-Open* is available under an open source license on `https://github.com/wideopen/datawatch/`. It is hosted at `https://wideopen.github.io/datawatch/` and keeps monitoring publications and identifying overdue datasets.

Chapter 4

AIDING DATA DISCOVERY IN THE REPOSITORY BY BUILDING CURATION CLASSIFIERS

4.1 Introduction

The confluence of technological advances and the open data movement [176] has led to an explosion of publicly available datasets, heralding an era of data-driven hypothesis generation and discovery in high-value applications [194]. A prime example is *open science*, which promotes open access to scientific discourse and data to facilitate broad data reuse and scientific collaboration [80]. In addition to enabling reproducibility, this trend has the potential to accelerate scientific discovery, reduce the cost of research, and facilitate automation [209, 156].

However, progress is hindered by the lack of consistent and high-quality annotations. For example, the NCBI Gene Expression Omnibus (GEO) [47] contains over two million gene expression profiles, yet only a fraction of them have explicit annotations indicating the tissue from which the sample was drawn, information that is crucial to understanding cell differentiation and cancer [105, 102]. As a result, only 20% of the datasets have ever been reused, and tissue-specific studies are still only performed at small scales [194].

Annotating data samples with standardized classes is the canonical multi-class classification problem, but standard supervised approaches are difficult to apply in these settings. Hiring experts to annotate examples for thousands of classes such as tissue types is unsustainable. Crowd-sourcing is generally not applicable, as annotation requires domain expertise that most crowd workers do not possess. Moreover, the annotation standard is often revised over time, incurring additional cost for labeling new examples.

While labeled data is expensive and difficult to create at scale, unlabeled data is usually

in abundant supply. Many methods have been proposed to exploit it, but they typically still require labeled examples to initiate the process [25, 169, 73]. Even zero-shot learning, where the name implies learning with no labeled examples for *some* classes, still requires labeled examples for related classes [186, 224].

In this chapter, we propose *EZLearn*, which makes annotation learning easy by exploiting two sources of *organic supervision*. First, the annotation classes generally come with a lexicon for standardized references (e.g., "liver", "kidney", "acute myeloid leukemia cell" for tissue types). While labeling individual data samples is expensive and time-consuming, it takes little effort for a domain expert to provide a few example terms for each class. In fact, in sciences and other high-value applications, such a lexicon is often available from an existing ontology. For example, the Brenda Tissue Ontology specifies 4931 human tissue types, each with a list of standard names [98]. Second, data samples are often accompanied by a free-text description, some of which directly or indirectly mention the relevant classes (e.g., the caption of a figure, or the description for a gene expression sample). Together with the lexicon, these descriptions present an opportunity for exploiting distant supervision by generating (noisy) labeled examples at scale [173]. We call such indirect supervision "organic" to emphasize that it is readily available as an integral part of a given domain.

In practice, however, there are serious challenges to enact this learning process. Descriptions are created for general human consumption, not as high-quality machine-readable annotations. They are provided voluntarily by data owners and lack consistency; ambiguity, typos, abbreviations, and non-standard references are common [153, 209]. Multiple samples may share a text description that mentions several classes, introducing uncertainty as to which class label is associated with which sample. Additionally, annotation standards evolve over time, introducing new terms and evicting old ones. As a result, while there are potentially many data samples whose descriptions contain class information, only a fraction of them can be correctly labeled using distant supervision. This problem is particularly acute for domains with numerous classes and frequent updates, such as the life sciences.

To best exploit indirect supervision using all instances, *EZLearn* introduces an auxiliary

text classifier for handling complex linguistic phenomena. This auxiliary classifier first uses the lexicon to find exact matches to teach the main classifier. In turn, the main classifier helps the auxiliary classifier improve by annotating additional examples with non-standard text mentions and correcting errors stemming from ambiguous mentions. This co-supervision continues until convergence. Effectively, *EZLearn* represents the first attempt in combining distant supervision and co-training, using text as the auxiliary modality for learning (Figure 4.1).

To investigate the effectiveness and generality of *EZLearn*, we applied it to two important applications: functional genomics and scientific figure comprehension, which differ substantially in sample input dimension and description length. In functional genomics, there are thousands of relevant classes. In scientific figure comprehension, prior work only considers three coarse classes, which we expand to twenty-four. In both scenarios, *EZLearn* successfully learned an accurate classifier with zero manually labeled examples.

While standard co-training has labeled examples from the beginning, *EZLearn* can only rely on distant supervision, which is inherently noisy. We investigate several ways to reconcile distant supervision with the trained classifier's predictions during co-training. We found that it generally helps to "remember" distant supervision while leaving room for correction, especially by accounting for the hierarchical relations among classes. We also conducted experiments to evaluate the impact of noise on *EZLearn*. The results show that *EZLearn* can withstand a large amount of simulated noise without suffering substantial loss in annotation accuracy.

4.2 Related Work

A perennial challenge in machine learning is to transcend the supervised paradigm by making use of unlabeled data. Standard unsupervised learning methods cluster data samples by explicitly or implicitly modeling similarity between them. It cannot be used directly for classification, as there is no direct relation between learned clusters and annotation classes.

In semi-supervised learning, direct supervision is augmented by annotating unlabeled

examples using either a learned model [181, 25] or similarity between examples [265]. It is an effective paradigm to refine learned models, but still requires initialization with sufficient labeled examples for all classes. Zero-shot learning or few-shot learning relax the requirement of labeled examples for some classes, but still need to have sufficient labeled examples for *related* classes [186, 224]. In this regard, they bear resemblance with domain adaptation [23, 55] and transfer learning [187, 202]. Zero-shot learning also faces additional challenges such as novelty detection to distinguish between known classes and new ones.

An alternative approach is to ask domain experts to provide example annotation functions, ranging from regular expressions [110] to general programs [205]. Common challenges include combating low recall and semantic drifts. Moreover, producing useful annotation functions still requires domain expertise and substantial manual effort, and may be impossible when predictions depend on complex input patterns (e.g., gene expression profiles).

EZLearn leverages domain lexicons to annotate noisy examples from text, similar to distant supervision [173]. However, distant supervision is predominantly used in information extraction, which considers the single view on text [200, 189]. In *EZLearn*, the text view is introduced to support the main annotation task, resembling co-training [25]. The original co-training algorithm annotates unlabeled examples in batches, where *EZLearn* relabels all examples in each iteration, similar to co-EM [181].

4.3 EZLearn

Let $X = \{x_i : i\}$ be the set of data samples and C be the set of classes. Automating data annotation involves learning a multi-class classifier $f : X \rightarrow C$. For example, x_i may be a vector of gene expression measurements for an individual, where C is the set of tissue types. Additionally, we denote t_i as the text description that accompanies x_i. If the description is not available, t_i is the empty string.

Algorithm 1 shows the *EZLearn* algorithm. By default, there are no available labeled examples (x, y^*) where $y^* \in C$ is the true class for annotating $x \in X$. Instead, *EZLearn* assumes that a lexicon L is available with a set of *example terms* L_c for each $c \in C$. We do not

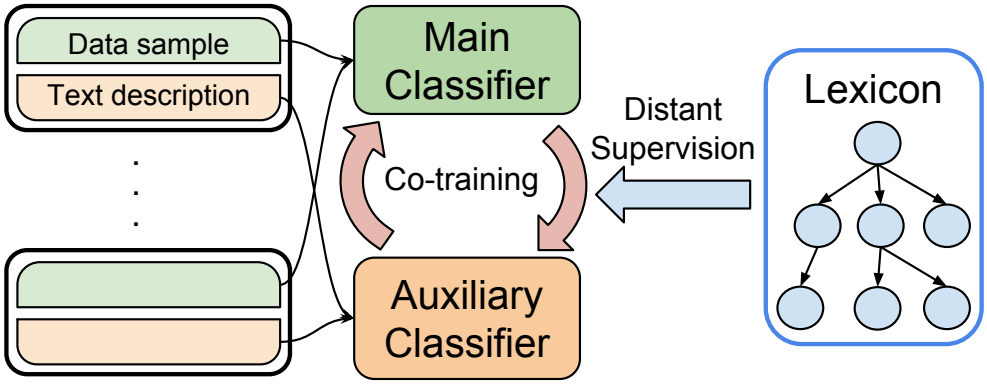


Figure 4.1: The *EZLearn* architecture: an auxiliary text-based classifier is introduced to bootstrap from the lexicon (often available from an ontology) and co-teaches the main classifier until convergence.

assume that L_c contains every possible synonym for c, nor that such terms are unambiguous. Rather, we simply require that L_c is non-empty for any c of interest. We use L_c's for distant supervision in *EZLearn*, by creating an initial labeled set D^0, which consists of all (x_i, t_i, c) where the text description t_i explicitly contains at least one term in L_c.

To handle linguistic variations and ambiguities, *EZLearn* introduces an auxiliary classifier $f_T : T \rightarrow C$, where $T = \{t_i : i\}$. At iteration k, we first train a new main classifier f^k using D^{k-1}. We then apply f^k to X and create a new labeled set D^k_T, which contains all (t_i, c) where $f^k(x_i) = c$. We then train a new text classifier f^k_T using D^k_T, and create the new labeled set D^k with all (x_i, c) where $f^k_T(t_i) = c$. This process continues until convergence, which is guaranteed given independence of the two views conditioned on the class label [25]. Empirically, it converges quickly.

For samples with distant-supervision labels, a classifier (main or auxiliary) might predict different labels in an iteration. Since distant supervision is noisy, reconciling it with the classifiers prediction could help correct its errors. The `Resolve`$(\cdot)$ function is introduced for this purpose. The direct analog of standard co-training returns distant-supervision labels if they are available (`Standard`). Conversely, `Resolve` could ignore distant supervision and

Algorithm 1 *EZLearn*

Input: Data samples X, text descriptions T, annotation classes C, and lexicon L containing example terms L_c for each class $c \in C$.

Output: Trained classifiers $f : X \to C$ (main) and $f_T : T \to C$ (auxiliary).

Initialize: Generate initial training data D^0 as all (x_i, t_i, c) for $x_i \in X$, $t_i \in T$, where t_i mentions some term in L_c.

for $k = 1 : N_{iter}$ **do**

$f \leftarrow \texttt{Train}_{\texttt{main}}(D^{k-1})$; $D_T^k \leftarrow \texttt{Resolve}(f(X), D^0)$

$f_T \leftarrow \texttt{Train}_{\texttt{aux}}(D_T^k)$; $D^k \leftarrow \texttt{Resolve}(f_T(T), D^0)$

end for

always return the classifier's prediction (`Predict`). Alternatively, `Resolve` may return all labels (`Union`) or the common ones (`Intersect`).

However, none of the above approaches consider the hierarchical relations among the label classes. Suppose that the text mentions both `neuron` and `leukemia`, whereas the classifier predicts `leukocyte` with high confidence. Our confidence in `leukemia` being the correct label should increase since `leukemia` is a subtype of `leukocyte`, and our confidence in `neuron` should decrease. We thus propose a more sophisticated variant of `Resolve` that captures such reasoning (`Relation`). Let c_1, c_2 be the two labels from distant supervision and classifier prediction, respectively. If c_1 and c_2 are the same, `Relation` returns $c = c_1 = c_2$. If they have a hierarchical relation, `Relation` will return the more specific one (i.e., the subtype). Otherwise, `Relation` returns none. If distant supervision or the classifier prediction assigns multiple labels to a sample, `Relation` will return results from all label pairs. (In domains with no hierarchical relations among the classes, `Relation` is the same as `Intersect`.)

We can use any classifier for $\texttt{Train}_{\texttt{main}}$ and $\texttt{Train}_{\texttt{aux}}$. Features for the main classifier are domain-specific and can be what any supervised approach might use. For the text classifier, we use standard n-gram features, which are effective in both applications we evaluated, though it is possible to tailor them for specific domains. Typically, the classifiers will take a

parametric form (e.g., $f(x) = f(x, \theta)$) and training with a labeled set D amounts to minimize some loss function L (i.e., $\theta^* = \arg\min_\theta \sum_{(x,y^*)\in D} L(f(x, \theta), y^*)$).

Generally, a classifier will output a score for each class rather than predicting a single class. The score reflects the confidence in predicting the given class. *EZLearn* generates the labeled set by adding all (sample, class) pairs for which the score crosses a hyperparameter threshold. We chose 0.3 in preliminary experiments, which allows up to 3 classes to be assigned to a sample. In all iterations, a labeled set might contain more than one class for a sample, which is not a problem for the learning algorithm and is useful when there is uncertainty about the correct class.

Method	**# Labeled**	**# All**	**AUPRC**	**Prec@0.5**	**Use Expression**	**Use Text**	**Use Lexicon**	**Use EM**
URSA	14510	0	0.40	0.52	yes	no	no	no
Co-EM	14510	116895	0.51	0.61	yes	yes	no	yes
Dist. Sup.	0	116895	0.59	0.63	yes	yes	yes	no
EZLearn	0	116895	**0.69**	**0.86**	yes	yes	yes	yes

Table 4.1: Comparison of test results between *EZLearn* and state-of-the-art supervised, semi-supervised, and distantly supervised methods on the CMHGP dataset. We reported the area under the precision-recall curve (AUPRC) and precision at 0.5 recall. *EZLearn* requires no manually labeled data, and substantially outperforms all other methods. Compared to URSA and co-EM, *EZLearn* can effectively leverage unlabeled data by exploiting organic supervision from text descriptions and lexicon. *EZLearn* amounts to initializing with distant supervision (first iteration) and continuing with an EM-like process as in co-training and co-EM, which leads to further significant gains.

4.4 Application: Functional Genomics

Different tissues, from neurons to blood, share the same genome but differ in gene expression. Annotating gene expression data with tissue type is critical to enable data reuse for cell-

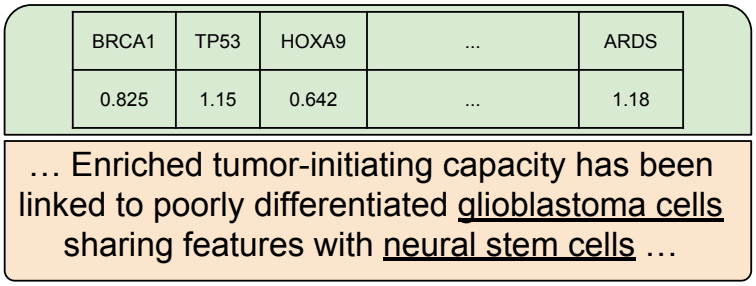


Figure 4.2: Example gene expression profile and its text description in Gene Expression Omnibus (GEO). Description is provided voluntarily and may contain ambiguous or incomplete class information.

development and cancer studies [209]. Lee et al. manually annotated a large dataset of 14,510 expression samples to train a state-of-the-art supervised classifier [153]. However, their dataset only covers 176 tissue types, or less than 4% of classes in BRENDA Tissue Ontology. In this section, we applied *EZLearn* to learn a far more accurate classifier that can in principle cover all tissue types in BRENDA. (In practice, the coverage is limited by the available unlabeled gene expression samples; in our experiments *EZLearn* learned to predict 601 tissue types.)

Annotation task The goal is to annotate gene expression samples with their tissue type. The input is a gene expression profile (a 20,000-dimension vector with a numeric value signifying the expression level for each gene). The output is a tissue type. We used the standard BRENDA Tissue Ontology [98], which contains 4931 human tissue types. For gene expression data, we used the Gene Expression Omnibus (GEO) [68], a popular repository run by the National Center for Biotechnology Information. Figure 4.2 shows an example gene expression profile with text description in GEO. We focused on the most common data-generation platform (Affymetrix U133 Plus 2.0), and obtained a dataset of 116,895 human samples. Each sample was processed using UPC to minimize batch effects and normalize the expression values to [0,1] [192]. Text descriptions were obtained from GEOmetadb [266].

Main classifier We implemented $\texttt{Train}_{\texttt{main}}$ using a deep denoising auto-encoder (DAE) with three LeakyReLU layers to convert the gene expression profile to a 128-dimensional vector [248], followed by multinomial logistic regression, trained end-to-end in Keras [44], using L2 regularization with weight $1e-4$ and RMSProp optimizer [237].

Auxiliary classifier We implemented $\texttt{Train}_{\texttt{aux}}$ using fastText with their recommended parameters (25 epochs and starting learning rate of 1.0) [130]. In principle, we can continue the alternating training steps until neither classifier's predictions change significantly. In practice, the algorithm converges quickly [181], and we simply ran all experiments with five iterations.

Systems We compared *EZLearn* with URSA [153], the state-of-the-art supervised method that was trained on a large labeled dataset of 14,510 examples and used a sophisticated Bayesian method to refine SVM classification based on the tissue ontology. We also compared it with co-training [25] and co-EM [181], two representative methods for leveraging unlabeled data that also use an auxiliary view to support the main classification. Unlike *EZLearn*, they require labeled data to train their initial classifiers. After the first iteration, high-confidence predictions on the unlabeled data are added to the labeled examples. In co-training, once a unlabeled sample is added to the labeled set, it is not reconsidered again, whereas in co-EM, all of them are re-annotated in each iteration. We found that co-training and co-EM performed similarly, so we only report the co-EM results.

Evaluation The BRENDA Tissue Ontology is a directed acyclic graph (DAG), with nodes being tissue types and directed edges pointing from a parent tissue to a child, such as `leukocyte` $\rightarrow$ `leukemia cell`. We evaluated the classification results using *ontology-based precision and recall*. We expand each singleton class (predicted or true) into a set that includes all ancestors except the root. We then measure precision and recall as usual: precision is the proportion of correct predicted classes among all predicted classes, and recall

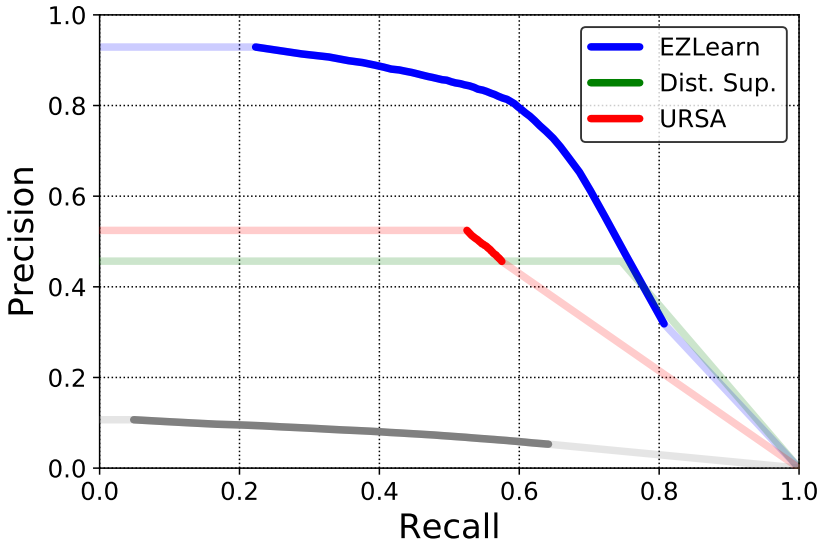
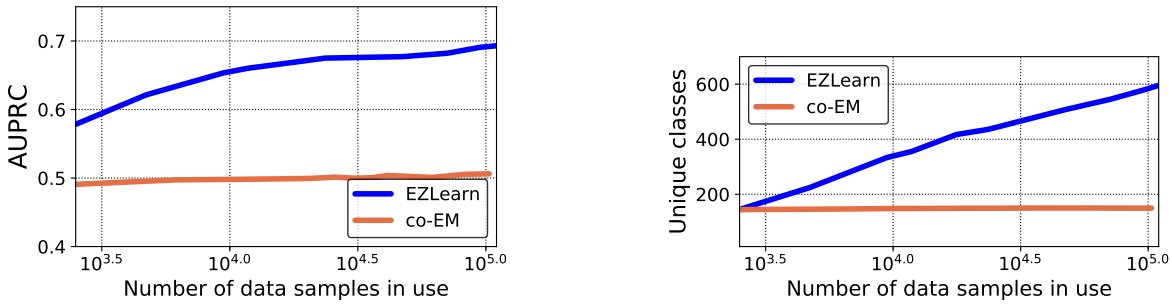


Figure 4.3: Ontology-based precision-recall curves comparing *EZLearn*, distant supervision, URSA, and the random baseline (gray). Extrapolated points are shown in transparent colors.

is the proportion of correct predicted classes among true classes, with ancestors included in all cases. This metric closely resembles the approach by Verspoor et al. [247], except that we are using the "micro" version (i.e., the predictions for all samples are first combined before measuring precision and recall). If the system predicts an irrelevant class in a different branch under the root, the intersection of the predicted and true sets is empty and the penalty is severe. If the predicted class is an ancestor (more general) or a descendent (more specific), the intersection is non-empty and the penalty is less severe, but overly general or overly specific predictions are penalized more than close neighbors. We tested on the Comprehensive Map of Human Gene Expression (CMHGP), the largest expression dataset with manual tissue annotations [238]. CMHPG used tissue types from the Experimental Factor Ontology (EFO) [165], which can be mapped to the BRENDA Tissue Ontology. To make the comparison fair, 7,209 CMHGP samples that were in the supervised training set for URSA were excluded from the test set. The final test set contains 15,129 samples of 628 tissue types.



(a) (b)

Figure 4.4: (a) Comparison of test accuracy with varying amount of unlabeled data, averaged over fifteen runs. *EZLearn* gained substantially with more data, whereas co-EM barely improves. (b) Comparison of number of unique classes in high-confidence predictions with varying amount of unlabeled data. *EZLearn*'s gain stems in large part from learning to annotate an increasing number of classes, by using organic supervision to generate noisy examples, whereas co-EM is confined to classes in its labeled data.

Results We report both the area under the precision-recall curve (AUPRC) and the precision at 0.5 recall. Table 4.1 shows the main classification results (with `Resolve` = `Relation` in *EZLearn*). Remarkably, without using any manually labeled data, *EZLearn* outperformed the state-of-the-art supervised method by a wide margin, improving AUPRC by an absolute 27 points over URSA, and over 30 points in precision at 0.5 recall. Compared to co-EM, *EZLearn* improves AUPRC by 18 points and precision at 0.5 recall by 25 points. Figure 4.3 shows the precision-recall curves.

To investigate why *EZLearn* attained such a clear advantage even against co-EM, which used both labeled and unlabeled data and jointly trained an auxiliary text classifier, we compared their performance using varying amount of unlabeled data (averaged over fifteen runs). Figure 4.4(a) shows the results. Note that the x-axis (number of unlabeled examples in use) is in log-scale. Co-EM barely improves with more unlabeled data, whereas *EZLearn* improves substantially from 2% to 100% of unlabeled data.

To understand why this is the case, we further compare the number of unique classes predicted by the two methods. See Figure 4.4(b). Co-EM is confined to the classes in its labeled data and its use of unlabeled data is limited to the extent of improving predictions for those classes. In contrast, by using organic supervision from the lexicon and text descriptions, *EZLearn* can expand the classes in its purview with more unlabeled data, in addition to improving predictive accuracy for individual classes. The gain seems to gradually taper off (Figure 4.4(a)), but we suspect that this is an artifact of the current test set. Although CMHGP is large, the number of tissue types in it (628) is still a fraction of that in the BRENDA Tissue Ontology (4931). Indeed, Figure 4.4(b) shows that the number of its predicted classes keeps climbing. This suggests that with additional unlabeled data *EZLearn* can improve even further, and with additional test classes, the advantage of *EZLearn* might become even larger.

We also evaluated on the subset of CMGHP with tissue types confined to those in the labeled data used by URSA and co-EM, to perfectly match their training conditions. Unsurprisingly, URSA and co-EM performed much better, attaining 0.53 and 0.67 in AUPRC, respectively (though URSA's accuracy is significantly lower than its training accuracy, suggesting overfitting). Remarkably, by exploiting organic supervision, *EZLearn* still outperformed both URSA and co-EM, attaining 0.71 in AUPROC in this setting.

EZLearn amounts to initializing with distant supervision (first iteration) and continuing with an EM-like process as in co-training and co-EM. This enables the main classifier and the auxiliary text classifier to improve each other during learning (Figure 4.5). Overall, compared to distant supervision, adding co-training led to further significant gains of 10 points in AUPRC and 23 points in precision at 0.5 recall (Table 1).

If labeled examples are available, *EZLearn* can simply add them to the labeled sets at each iteration. After incorporating the URSA labeled examples [153], the AUPRC of *EZLearn* improved by two absolute points, with precision at 0.5 recall increasing to 0.87 (not shown in Table 1).

Compared to direct supervision, organic supervision is inherently noisy. Consequently, it

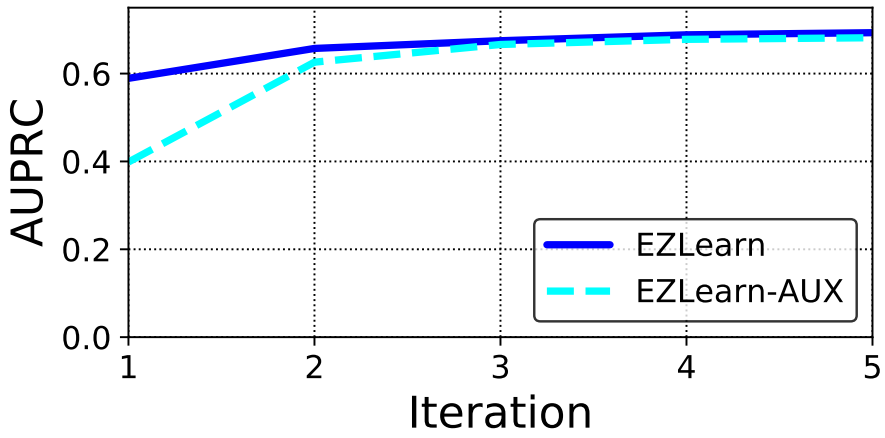


Figure 4.5: Comparison of test accuracy of the main and auxiliary classifiers at various iterations during learning.

`Resolve`	`Standard`	`Predict`	`Union`	`Intersect`	`Relational`
# Classes	623	329	603	351	601
AUPRC	0.59	0.64	0.59	0.66	**0.69**

Table 4.2: Comparison of test results and numbers of unique classes in high-confidence predictions on the Comprehensive Map of Human Gene Expression by *EZLearn* with various strategies in resolving conflicts between distant supervision and classifier prediction.

is generally beneficial to reconcile classifier prediction with distant supervision when they are in conflict, as Table 4.2 shows. `Standard` (always choosing distant supervision when available) significantly trailed the alternative approach that always picks classifier's prediction (`Predict`). `Union` predicted more classes than `Intersect` but suffered large precision loss. By taking into account of hierarchical relations in the class ontology, `Relation` substantially outperformed all other methods in accuracy, while also covering a large number of classes.

To evaluate *EZLearn*'s robustness, we simulated noise by replacing a portion of the initial distant-supervision labels with random ones. Figure 4.6 shows the results. Interestingly,

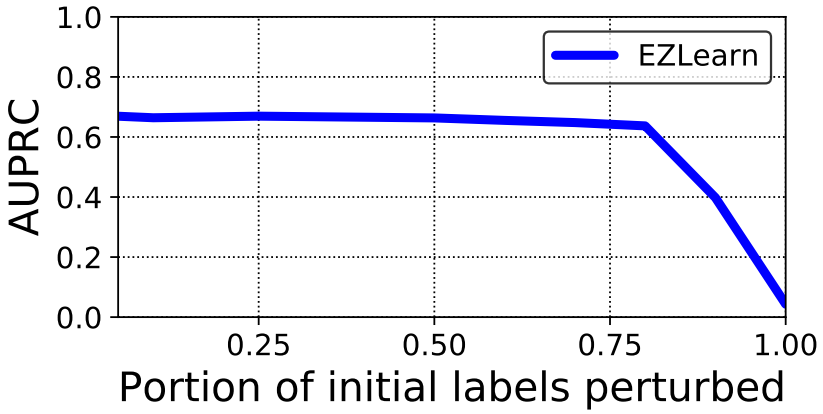


Figure 4.6: *EZLearn*'s test accuracy with varying portion of the distant-supervision labels replaced by random ones in the first iteration. *EZLearn* is remarkably robust to noise, with its accuracy only starting to deteriorate significantly after 80% of labels are perturbed.

EZLearn can withstand a significant amount of label perturbation: test performance only deteriorates drastically when more than 80% of initial labels are replaced by random ones. This result suggests that *EZLearn* can still perform well for applications with far more noise in their organic supervision.

4.5 Application: Scientific Figure Comprehension

Figures in scientific papers communicate key results and provide visual explanations of complex concepts. However, while text understanding has been intensely studied, figures have received much less attention in the past. A notable exception is the Viziometrics project [151], which annotated a large number of examples for classifying scientific figures. Due to the considerable cost of labeling examples, they only used five coarse classes: `Plot`, `Diagram`, `Image`, `Table` and `Equation`. We exclude the last two as they do not represent true figures. In practice, figure-comprehension projects would be much more useful if they include larger set of specialized figure types. To explore this direction, we devised an ontology where `Plot`, `Diagram`, and `Image` are further refined into a total of twenty-four classes, such as `Boxplot`,

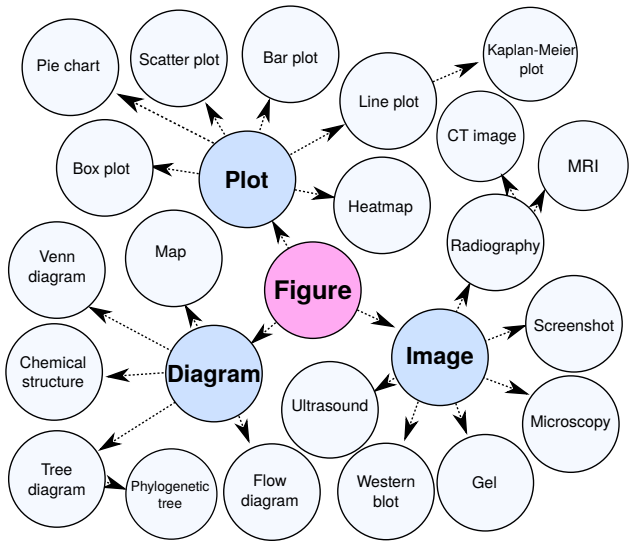


Figure 4.7: The Viziometrics project only considers three coarse classes `Plot`, `Diagram`, and `Image` for figures due to high labeling cost. We expanded them into 24 classes, which *EZLearn* learned to accurately predict with zero manually labeled examples.

`MRI` and `PieChart` (Figure 4.7). *EZLearn* naturally accommodates a large and dynamic ontology since no manually labeled data is required.

Annotation task The goal is to annotate figures with semantic types shown in Figure 4.7. The input is the image of a figure with varying size. The output is the semantic type. We obtained the data from the Viziometrics project [151] through its open API. For simplicity, we focused on the non-composite subset comprising single-pane figures, yielding 1,174,456 figures along with free-text captions for use as distant supervision. As in the gene expression case, captions might be empty or missing.

System Each figure image was first resized and converted to a 2048-dimensional real-valued vector using a convolutional neural network [109] trained on ImageNet [57]. We follow [117] and use the ResNet-50 model with pre-trained weights provided by Keras [44]. We used the same classifiers and hyperparameters as in the functional genomics application. We used

a lexicon that simply comprises of the names of the new classes, and compared *EZLearn* with the Viziometrics classifier. We also compared with a lexicon-informed baseline that annotates a figure with the most specific class whose name is mentioned in the caption (or root otherwise).

Evaluation We followed the functional genomics application and evaluated on ontology-based precision and recall. Since the new classes are direct refinement of the old ones, we can also evaluate the Viziometrics classifier using this metric. To the best of our knowledge, there is no prior dataset or evaluation for figure annotation with fine-grained semantic classes as in Figure 4.7. Therefore, we manually annotated an independent test set of 500 examples.

	Lexicon	Viziometrics	Distant Supervision	*EZLearn*
AUPRC	0.44	0.53	0.75	**0.79**
Precision@0.5	0.31	0.43	0.87	**0.88**

Table 4.3: Comparison of test results between *EZLearn*, the lexicon baseline, the Viziometrics classifier, and distant supervision on the test set of 500 images manually labeled using an ontology from Figure 4.7.

Results *EZLearn* substantially outperformed both the lexicon-informed baseline and the Viziometrics classifier (Table 4.3). The state-of-the-art Viziometrics classifier was trained on 3271 labeled examples, and attained an accuracy of 92% on the coarse classes. So the gain attained by *EZLearn* reflects its ability to extract a large amount of fine-grained semantic information missing in the coarse classes. Figure 4.8 shows example annotations by *EZLearn*, all chosen from figures with no class mention in the caption.

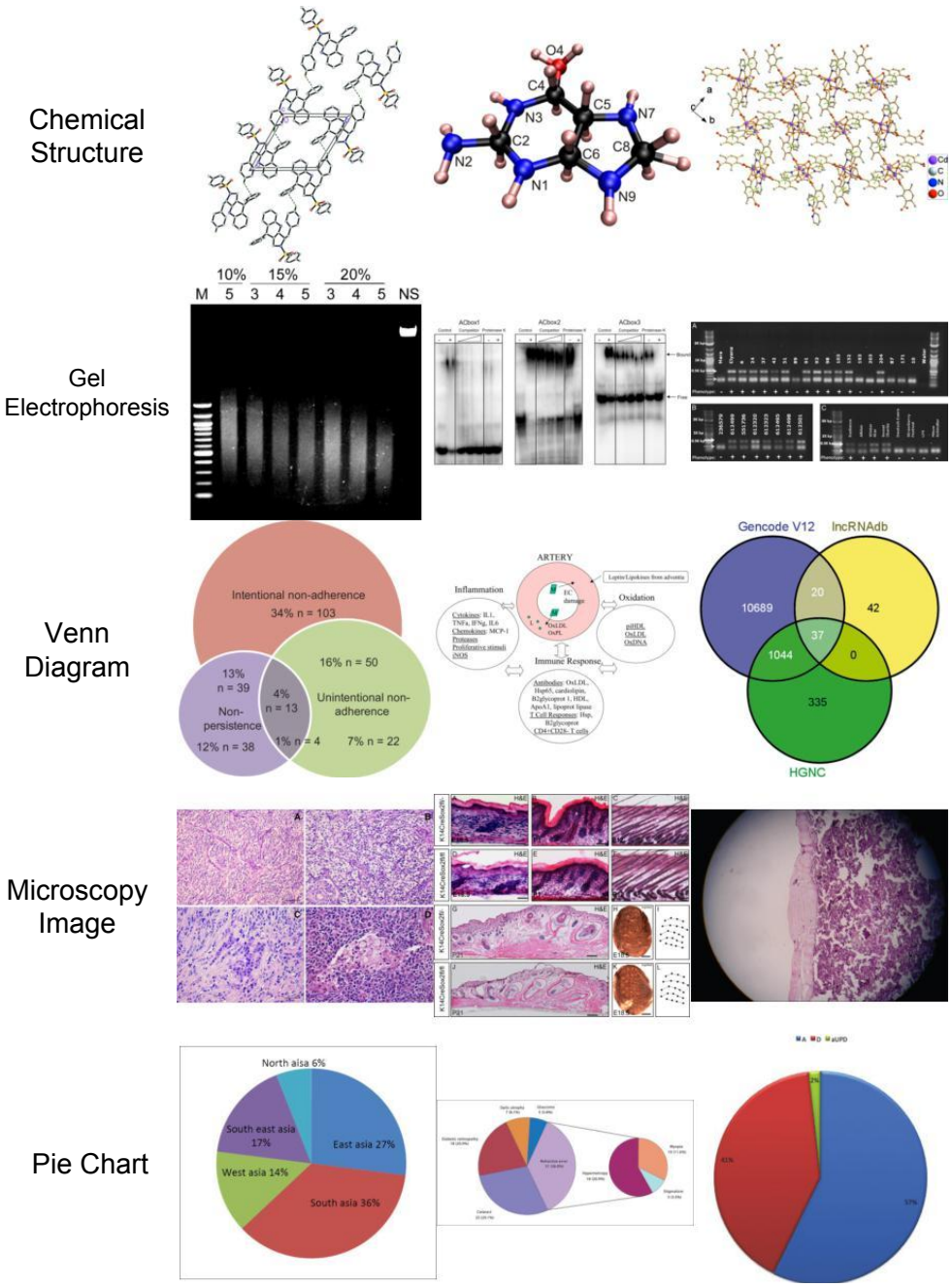


Figure 4.8: Example annotations by *EZLearn*, all chosen among figures with no class information in their captions.

4.6 Summary

In this chapter we proposed *EZLearn* for automated data annotation, by combining distant supervision and co-training. *EZLearn* is well suited to high-value domains with numerous classes and frequent update. Experiments in functional genomics and scientific figure comprehension show that *EZLearn* is broadly applicable, robust to noise, and capable of learning accurate classifier without manually labeled data, even outperforming state-of-the-art supervised systems by a wide margin. *EZLearn* annotations for tissue types in GEO repository, as well a processed dataset collected for the experiments will form basis for claim verification and generalization work presented in Chapter 7.

The work in this chapter was first reported in [94]. *EZLearn* is available under an open source license at `https://github.com/maximsch2/EZLearn.jl`. *EZLearn* is additionally using an ontology parser, available at `https://github.com/maximsch2/OBOParse.jl`.

Chapter 5

RECONSTRUCTING GENE REGULATORY NETWORKS BASED ON PUBLIC DATASETS

5.1 Introduction

Gaussian graphical models (GGMs) provide a compact representation of the statistical dependencies among variables. Learning the structure of GGMs from data that contain the measurements on a set of variables across samples has significantly facilitated data-driven discovery in a diverse set of scientific fields. For example, biologists can gain insights into how thousands of genes interact with each other in various disease processes by learning the GGM structure from gene expression data that measure the mRNA expression levels of genes across hundreds of patients. Existing algorithms for learning the structure of GGMs lack scalability and interpretability, which limits their utility when there is a large number of variables. Most learning algorithms perform $O(p^3)$ computations per iteration, where p denotes the number of variables; consequently they are impractical when p exceeds tens of thousands. Furthermore, a network based on a large number of variables can be difficult to interpret due to the presence of a large number of connections between the variables.

To resolve these challenges, we propose the *pathway graphical lasso* (PathGLasso) framework, which consists of the incorporation of pathway-based constraints and an efficient learning algorithm. We assume that we are given a set of pathways *a priori*, and that each pathway contains a (possibly overlapping) subset of the variables. We assume that a pair of variables can be connected to each other only if they co-occur in at least one pathway. Figure 5.1 illustrates a simple example network of 8 variables: $\{x_1, x_2, x_3\}$ in Pathway 1, $\{x_2, x_3, x_4, x_5, x_6\}$ in Pathway 2 and $\{x_6, x_7, x_8\}$ in Pathway 3. By incorporating the pathway constraints, we can effectively reduce the search space of network structures by excluding nonsensical edges.

Pathway constraints have the potential to improve structure learning of GGMs in several applications. In the context of gene regulatory networks, one can make use of pathway databases such as Reactome [50] that specify sets of genes that are likely work together. Making use of such pathways in learning the network can yield results that are more meaningful and interpretable. In computational neuroscience, when learning an interaction network of brain activation from fMRI data, we can use our prior knowledge that nearby brain regions are likely to interact with each other [74]. In computer vision, in which each pixel in an image corresponds to a variable in a network, one can generate overlapping pathways by grouping nearby pixels; this has been shown to be an effective prior in several applications [115]. For example, Figure 5.2 compares network estimates of the true 2D lattice network for the unconstrained graphical lasso model and the pathway constrained graphical lasso model (5.2) when each pathway contains nearby variables.

The key idea in this chapter is that we define certain edges to be non-existent only when the corresponding variables are not together in any of the pathways. Many of the potential edges within a pathway can also end up becoming zero. The pathway constraints provide a way of reducing the search space of structure learning. They do not determine the structure to a large extent.

In this chapter, we present a learning algorithm that takes advantage of the pathway assumption in order to deliver a dramatic improvement in performance relative to existing approaches. We make use of a block-coordinate descent approach, in which we update each pathway individually. We apply a message-passing algorithm in order to enforce the correct solution jointly across all pathways.

5.2 Pathway Constrained Sparse Inverse Covariance Estimation

5.2.1 Preliminaries

Suppose that we wish to learn a GGM with p variables based on n observations $\mathbf{x}^1, \ldots, \mathbf{x}^n \overset{\text{i.i.d.}}{\sim} N(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ is a $p \times p$ covariance matrix. It is well known that $(\mathbf{\Sigma}^{-1})_{jj'} = 0$ for

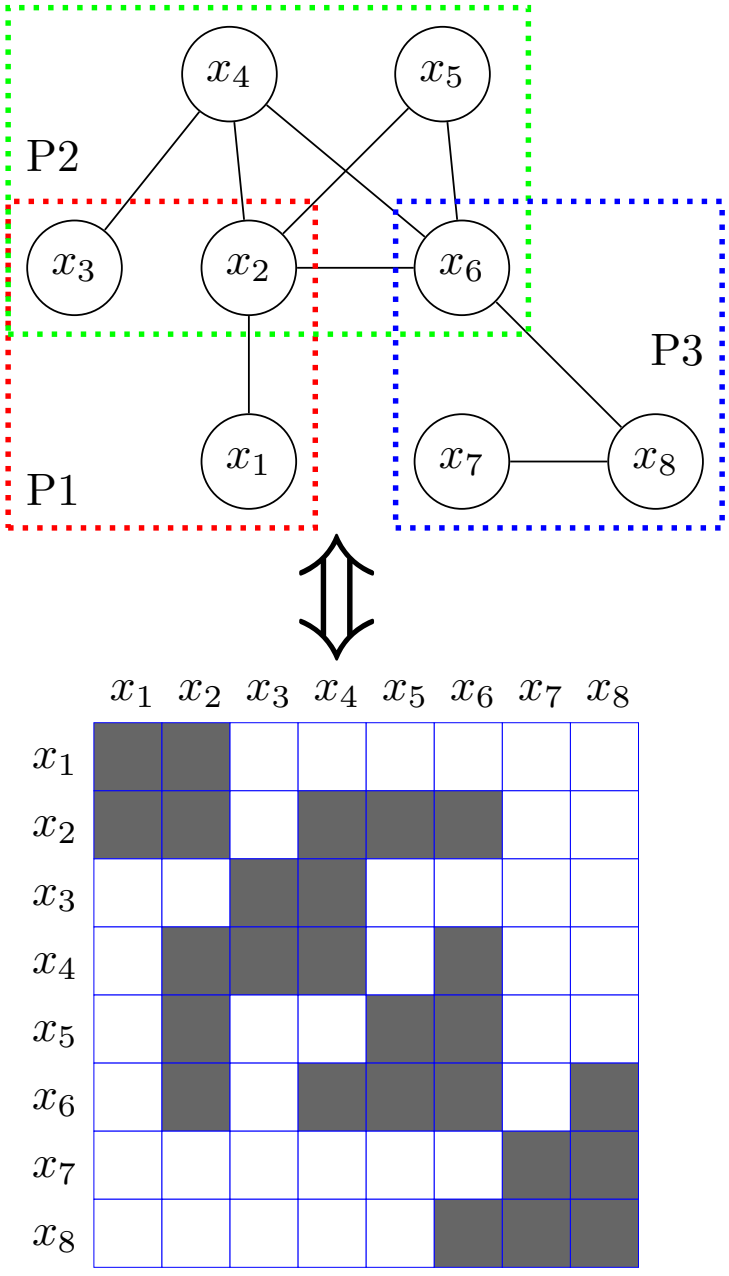


Figure 5.1: Graphical representation of pathways (top) and the corresponding precision matrix (bottom).

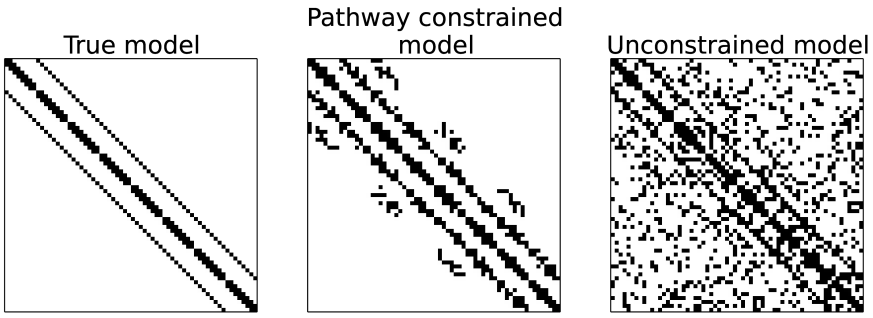


Figure 5.2: Comparison of learned networks between the pathway graphical lasso (middle) and the standard graphical lasso (right). The true network has the lattice structure (left).

some $j \neq j'$ if and only if X_j and $X_{j'}$ are conditionally independent given X_k with $k = \{1, \ldots, p\} \setminus \{j, j'\}$ [166, 147]. Hence, the non-zero pattern of $\mathbf{\Sigma}^{-1}$ corresponds to the graph structure of a GGM. In order to obtain a sparse estimate for $\mathbf{\Sigma}^{-1}$, a number of authors have considered the *graphical lasso* optimization problem [259, 8, 78]:

$$\begin{aligned} &\underset{\Theta}{\text{minimize}} - \log\det(\Theta) + \text{trace}(S\Theta) + \lambda\|\Theta\|_1 \\ &\text{subject to } \Theta \succeq 0, \end{aligned} \tag{5.1}$$

where S is the empirical covariance matrix and $\lambda > 0$ is an l_1 regularization parameter. We denote the estimate of the inverse covariance matrix by Θ throughout the chapter.

5.2.2 Pathway Graphical Lasso Problem

Consider a set of edges within k pathways $P_1, ..., P_k$: $\mathcal{F} = \bigcup_{t=1}^{k} \{(i,j) | i,j \in P_t\}$. We assume that edges that are outside of $\mathcal{F}$ are set to zero. This modifies the graphical lasso problem (5.1) as:

$$\begin{aligned} &\underset{\Theta}{\text{minimize}} \quad - \log\det(\Theta) + \text{trace}(S\Theta) + \lambda\|\Theta\|_1 \\ &\text{subject to} \quad \Theta_{ij} = 0,\ (i,j) \notin \mathcal{F};\ \ \Theta \succeq 0. \end{aligned} \tag{5.2}$$

To the best of our knowledge, this is a novel problem, and none of the existing algorithms for learning GGMs can solve (5.2) directly. However, many existing approaches for solving (5.1) either support (e.g., QUIC) or can easily be adapted to support (e.g., HUGE) a per-variable regularization scheme. Setting specific regularization parameter values in (5.1) to some very large number (say 10^{10}) effectively forces the corresponding values in Θ to be zero. We observed in our experiments that methods that employ active set heuristics can get a significant performance boost from such a setting.

5.2.3 Related Work

Our proposal, PathGLasso, decomposes the original problem (5.2) into a set of smaller overlapping problems, and uses a divide-and-conquer approach to optimize the local marginal likelihood with a modified sparsity-inducing penalty. This novel combination of ideas differentiates PathGLasso from previous approaches to learn GGMs.

Several authors attempted to optimize the local marginal likelihood of a handful of nearby variables for parameter estimation in GGMs with fixed structures [256, 171]. It was proven that this type of local parameter estimation produces a globally optimal solution under certain conditions [175]. Though these papers adopted a similar idea of exploiting the conditional independence of a set of variables from the rest of the network given their Markov blanket, they solve a fundamentally different problem. PathGLasso learns a pathway-constrained structure of the network in addition to estimating individual parameters. Moreover, the l_1 regularization that we employ for structure learning makes these previous approaches inapplicable to our setting.

Another approach [118] first partitions variables into non-overlapping pathways by using a clustering algorithm, and then estimates a network for each pathway. In contrast, in this work we assume that overlapping pathway information is provided to us, although it could alternatively be estimated from the data, for example by running a clustering algorithm with soft assignment. The approach of [118] is not applicable to our problem, because combining independently estimated networks from overlapping pathways into a global network can

lead to a non-positive definite solution. Like [118], PathGLasso is agnostic of the specific optimization algorithm for learning the network within each pathway.

There are methods that aim to infer *modules*, sets of densely connected variables. Many approaches attempt to learn a network with a prior that induces modules [6], which makes it significantly less efficient than without the prior. To address this, [38] proposed a method that can jointly learn modules and a network among modules. Although this method achieves scalability and interpretability, it does not learn a network of individual variables. PathGLasso addresses both of these shortcomings.

Finally, a number of methods have been proposed to solve the l_1 penalized sparse inverse covariance estimation problem (5.1). One such algorithm [78] uses row-wise updates on the dual problem by solving a lasso problem at every step. The lasso problem is solved using a coordinate-descent algorithm that takes advantage of an active set heuristic, reducing computational load for sparse matrices. In Section 5.4, we provide a comparison to an efficient implementation of this method, provided by the R package HUGE [262]. Another paper [119] proposes a quadratic approximation based algorithm (QUIC) that achieves super-linear convergence rates as well as significant performance improvements due to clever partitioning of variables into free and fixed sets.

In the following section, we propose a novel learning algorithm for pathway constrained sparse inverse covariance estimation, and demonstrate that it shows significant improvement in run time compared to general off-the-shelf methods for (5.1) that we modified to solve (5.2).

5.3 PathGLasso Learning Algorithm

5.3.1 Overview

We employ a version of the block-coordinate descent approach to solve the optimization problem (5.2); a discussion of the convergence properties of the standard block-coordinate descent is provided in [241]. In each iteration, we update the parameters that correspond to

one pathway, with all of the other parameters held fixed. Consider updating the parameters in the pathway P_1. After re-arranging the variables, the $p \times p$ inverse covariance matrix Θ takes the form:

$$\Theta = \begin{pmatrix} A & B & 0 \\ B^T & C & D \\ 0 & D^T & E \end{pmatrix} \tag{5.3}$$

where $\Theta_1 = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix}$ contains the parameters in P_1, $\Theta_2 = \begin{pmatrix} C & D \\ D^T & E \end{pmatrix}$ contains the parameters in the rest of the pathways, and C corresponds to the subset of variables that are in the intersection of P_1 and all other pathways.

5.3.2 Updating Each Pathway

We show that updating the parameters in P_1 with all of the other parameters held fixed boils down to estimating a $p_1 \times p_1$ inverse covariance matrix, where p_1 is the number of variables in P_1. This is not obvious, because P_1 overlaps with other pathways (C in (5.3)). To update the parameters in P_1, we need to solve the following optimization problem:

$$\begin{aligned} &\underset{A,B,C}{\text{minimize}} - \log\det(\Theta) + \text{trace}(S\Theta) + \lambda\|\Theta\|_1 \\ &\text{subject to } \Theta \succeq 0. \end{aligned} \tag{5.4}$$

Applying the Schur complement decomposition, we obtain

$$\det\Theta = \det E \cdot \det(\Theta_1 - \Delta), \tag{5.5}$$

where

$$\Delta = [0;\ D] \cdot E^{-1} \cdot [0\ \ D^T]. \tag{5.6}$$

Given that D and E are fixed, the optimization problem (5.4) is equivalent to the following problem:

$$\begin{aligned} \underset{\Theta_1}{\text{minimize}} &- \log\det(\Theta_1 - \Delta) + \text{trace}(S_1(\Theta_1 - \Delta)) \\ &+ \lambda\|\Theta_1\|_1 \\ \text{subject to } &\Theta_1 - \Delta \succeq 0, \end{aligned} \tag{5.7}$$

where S_1 is the portion of the empirical covariance matrix corresponding to P_1.

Let $\Omega = \Theta_1 - \Delta$. Since our estimate of Θ is always positive definite, E is also positive definite as it is the principal submatrix of Θ. Thus, constraining Ω to be positive definite will guarantee the positive definiteness of Θ. Then, (5.7) is equivalent to the following optimization problem:

$$\begin{aligned} &\underset{\Omega}{\text{minimize}} - \log\det(\Omega) + \text{trace}(S_1\Omega) + \lambda\|\Omega + \Delta\|_1 \\ &\text{subject to } \Omega \succeq 0. \end{aligned} \tag{5.8}$$

Note that (5.8) is the graphical lasso problem with a "shifted" l_1 penalty. This means that our block-coordinate update can make use of any algorithm for solving the graphical lasso problem, as long as it can be adapted to work with the shifted penalty. In this work, we used the DP-GLASSO (dual-primal graphical lasso) algorithm [168], which works well with restarts and guarantees a positive definite solution at each iteration.

5.3.3 Probabilistic Interpretation

The marginal distribution of the variables that are in P_1 is Gaussian with mean zero and precision matrix $\Omega = \begin{pmatrix} A & B \\ B^T & C - D \cdot E^{-1} \cdot D^T \end{pmatrix}$, where Θ denotes the true precision matrix of the entire distribution partitioned as in (5.3). Then, the optimization problem (5.8) can be viewed as maximizing the marginal likelihood of the variables in P_1 with adjustments in the regularization term. That term makes it possible to take into account the variables that are outside of P_1. For example, in Figure 5.1, even if x_2 and x_3 are separately connected with x_4, maximizing the marginal likelihood of P_1 would induce an edge between x_2 and x_3 because x_4 is outside of P_1. Δ in (5.8) informs the algorithm that the connection between x_2 and x_3 can be explained away by x_4 when optimizing the marginal likelihood of P_1.

5.3.4 Marginalization of More Than One Pathway

In Section 5.3.2, we showed that updating the parameters for a given pathway requires the computation of Δ (5.6), a function of all of the other pathways. We could compute Δ directly by inverting a potentially very large matrix E (5.3), and performing two matrix multiplications. This corresponds to marginalizing all other pathways at once. In this section, we show that when more than two pathways are present, it is possible to avoid computing the matrix inverse of E explicitly, by instead marginalizing the pathways one-at-a-time.

As an example, we consider a very simple case of three pathways that form a linear chain,

$$\Theta = \begin{pmatrix} A & B & 0 & 0 \\ B^T & C & D & 0 \\ 0 & D^T & E & F \\ 0 & 0 & F^T & G \end{pmatrix}. \tag{5.9}$$

Suppose that we want to update the top left pathway, corresponding to the matrix $\Theta_1 = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix}$. Following the arguments in Section 5.3.2, computing (5.6) involves inverting the matrix $\begin{pmatrix} E & F \\ F^T & G \end{pmatrix}$. Instead, we note that

$$\begin{aligned} \det\Theta &= \det(G) \cdot \det \begin{pmatrix} A & B & 0 \\ B^T & C & D \\ 0 & D^T & E - FG^{-1}F^T \end{pmatrix} \\ &= \det(G) \cdot \det(E - FG^{-1}F^T) \\ &\quad \cdot \det \begin{pmatrix} A & B \\ B^T & C - D(E - FG^{-1}F^T)^{-1}D^T \end{pmatrix}. \end{aligned} \tag{5.10}$$

Recall that our goal is to update the top left pathway in (5.9), with D, E, F, and G held

fixed. Therefore, we can re-write (5.10) as

$$\begin{aligned}\det\Theta &= \det(G)\cdot\det(E - FG^{-1}F^T)\cdot\det(\Theta_1 - \Delta)\\ &= \text{const}\cdot\det(\Theta_1 - \Delta),\end{aligned}$$

where

$$\Delta = \begin{pmatrix} 0 & 0 \\ 0 & D(E - FG^{-1}F^T)^{-1}D^T \end{pmatrix}. \tag{5.11}$$

Using the arguments in Section 5.3.2, we see that it is possible to update Θ_1 using a shifted graphical lasso problem of the form (5.8).

We note that the computations in (5.10) allowed us to derive the form of Δ in (5.11) without needing to invert the matrix $\begin{pmatrix} E & F \\ F^T & G \end{pmatrix}$. Instead, computing Δ simply required inverting two smaller matrices, G and $E - FG^{-1}F^T$. This is an example of a more general principle: marginalizing pathways one-at-a-time leads to a dramatic improvement in performance over the naive approach of marginalizing all pathways at once outlined in Section 5.3.2. This general principle holds in the case of more than three pathways, and in fact will lead to much greater computational improvements as the number of pathways grows.

In (5.11), the term $FG^{-1}F^T$ can be interpreted as a *message*, a piece of information needed to marginalize out the variables that are within pathway 3 and not in the other pathways. In Section 5.3.5, we show that it is possible to cleverly re-use these messages in order to speed up computations.

5.3.5 Message-Passing Approach

A naive application of the idea described in Section 5.3.4 would require computing $O(k^2)$ messages per iteration, where k is the number of pathways. This is because in each iteration, we update all k pathways, and each update requires marginalizing over $k-1$ other pathways.

In fact, we can drastically speed up computations using a divide-and-conquer message passing scheme. This approach relies on the careful re-use of messages across pathway updates. Using such a scheme, we need to compute only $O(k\log k)$ messages per iteration.

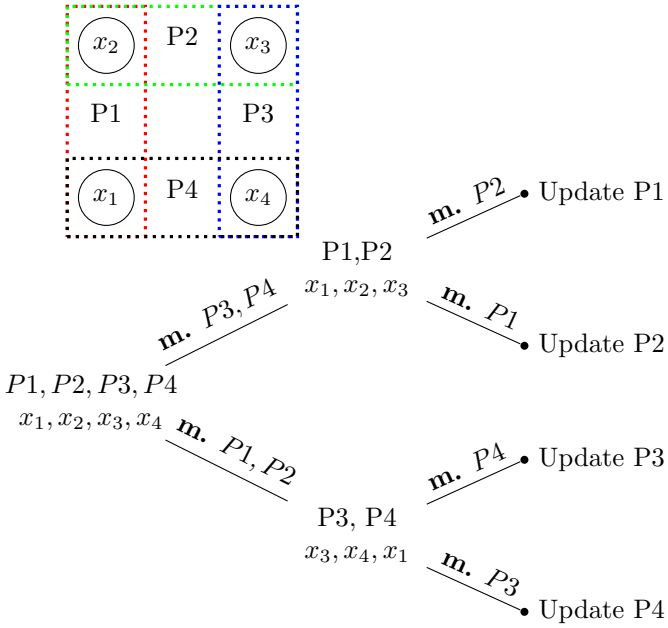


Figure 5.3: Example with 4 pathways forming a cycle **m.** means marginalization.

An example is shown in Figure 5.3. In the special case of pathways that form a tree structure, we can further improve this approach to compute only $O(k)$ messages per iteration.

5.4 Experiments

Since there is no learning algorithm designed to efficiently solve (5.2), we compared PathGLasso with the state-of-the-art learning algorithms for the graphical lasso problem (5.1) – QUIC [119] and HUGE [262]. Although neither of these competitors solves (5.2) directly, we adapt them to solve (5.2) by supplying a matrix of separate λ values for each entry in the inverse covariance matrix. We set $\lambda = 10^{10}$ for the entries that lie outside of the pathways, making them solve exactly the same problem as PathGLasso. We observed that supplying such a matrix improves performance of both methods due to the active set heuristics employed by these methods. Additionally, we compared with DP-GLASSO [168], the method that we used to learn parameters in each pathway (5.8), to make sure that the superior performance of PathGLasso is due to our decomposition approach as opposed to the use of

DP-GLASSO. We note that DP-GLASSO is not competitive in this setting because it does not employ active set heuristics. All comparisons were run on 4 core Intel Core i7-3770 CPU @ 3.40GHz with 8GB of RAM.

5.4.1 Synthetic datasets comparison

We compared PathGLasso with QUIC, HUGE and DP-GLASSO on 3 scenarios: 1) Cycle: Pathways form one large cycle with 50 genes per pathway with overlap size of 10; 2) Lattice: The true underlying model is a 2D lattice, and each pathway contains between 3 and 7 nearby variables; and 3) Random: Each pathway consists of randomly selected genes. For each setting, we generated a true underlying connectivity graph, converted it to the precision matrix following the procedure from [158], and generated 100 samples from the multivariate Gaussian distribution.

We observed that PathGLasso dramatically improves the run time compared to QUIC, HUGE and DP-GLASSO (Figure 5.4), sometimes up to two orders of magnitude. We note that DP-GLASSO, used as an internal solver for PathGLasso, performs much worse than both HUGE and QUIC. This is because DP-GLASSO is not as efficient as QUIC or HUGE when solving very sparse problems due to the lack of active set heuristics. This is not a problem for PathGLasso, because our within-pathway networks are small, and are much denser on average than the entire network.

In addition to varying the number of variables p (Figure 5.4), we also explored the effect of the degrees of overlap among the pathways (Figure 5.5). We denote by η the sum of sizes of all pathways divided by the total number of variables in the entire network. This can be interpreted as the average number of pathways to which each variable belongs. In a non-overlapping model, $\eta = 1$. The parameter η grows with the size of the overlap between pathways. A set of pathways from a real biological database, called Reactome [50], has $\eta = 1.95$ (see Section 5.4.2).

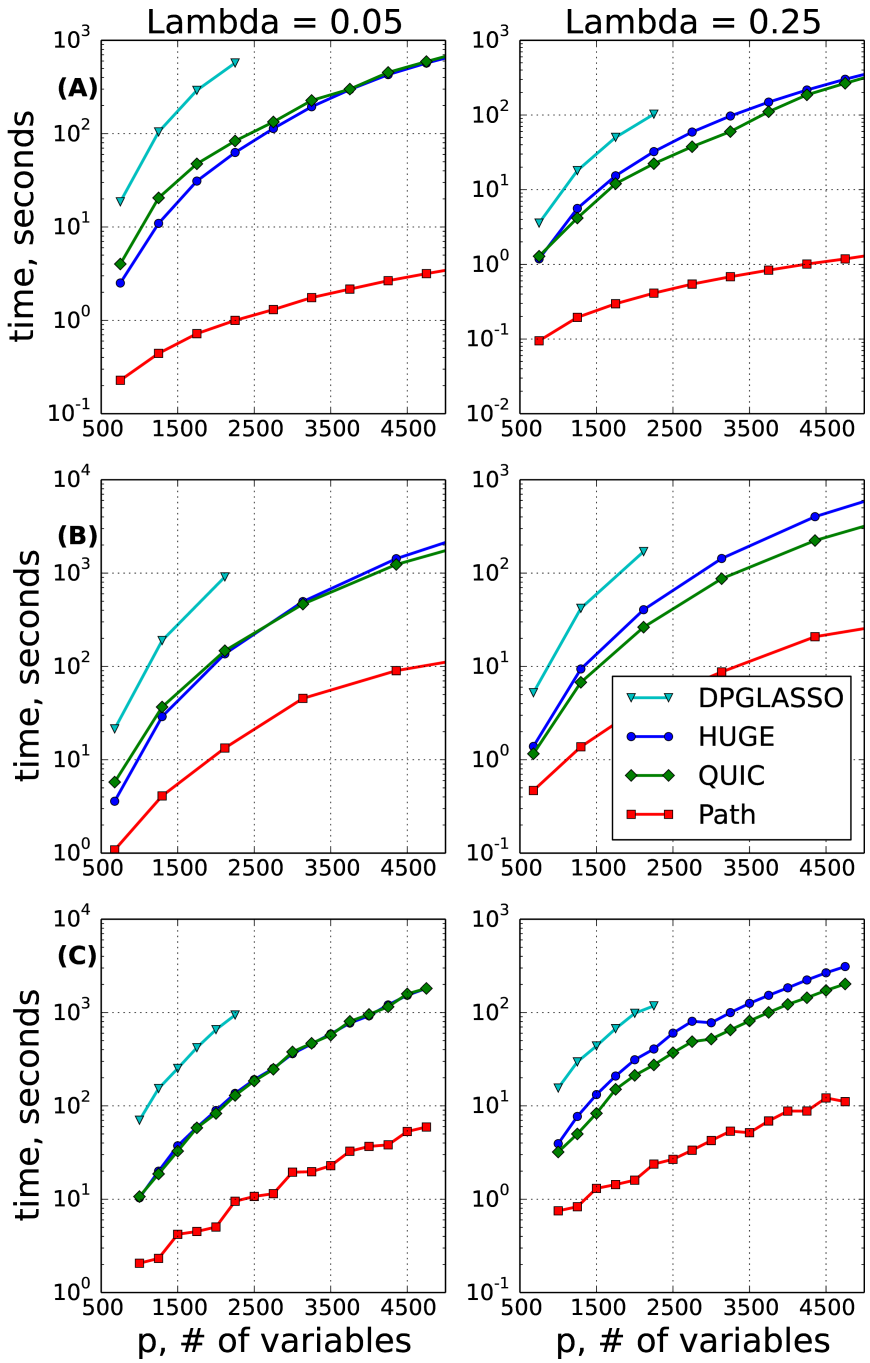


Figure 5.4: Run time (y-axis) for (A) Cycle, (B) Lattice and (C) Random (see text for details).

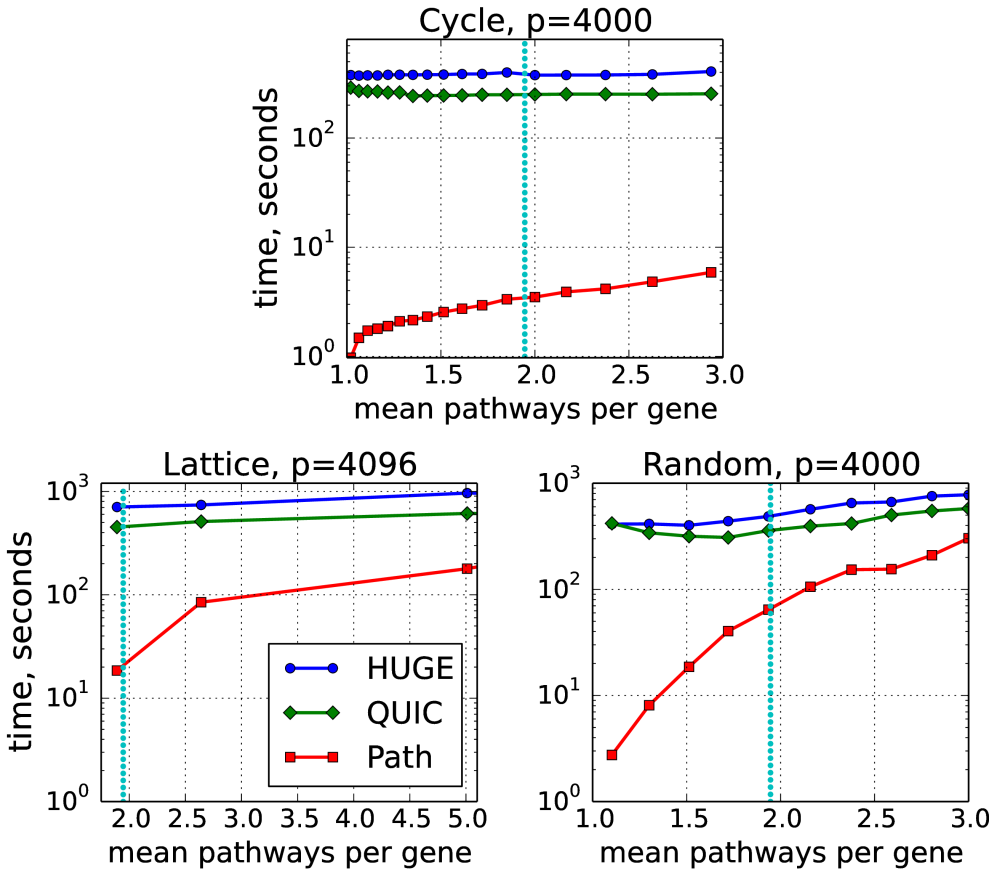


Figure 5.5: Run time for various values of η, with $\lambda = 0.1$. $\eta = 1.95$ is drawn as a dotted vertical line.

5.4.2 Real data experiments

We considered two gene expression datasets from acute myeloid leukemia (AML) studies: MILE [104] and GENTLES [86] containing 541 and 248 samples, respectively. The raw data were processed using the Affy R package, MAS5 normalized, and corrected for batch effects by ComBat. We used a widely used curated pathway database, called Reactome, that contains a set of genes for each biological pathway. We considered pathways containing fewer than 150 genes, which results in 4591 genes in 156 pathways. Large pathways are often supersets of smaller pathways; therefore, this filtering scheme allowed us to focus on finer-grained pathways. Following [119], we plotted the relative error (y-axis) against time (x-axis) in Figure 5.6A. We computed the relative error in the following way. All 3 methods (QUIC, HUGE and PathGLasso) were run for an extended period of time with small tolerance parameters. We denote by Θ^* the learned parameter that leads to the lowest value of objective function. Relative error was then computed as $|ll(\Theta) - ll(\Theta^*)| \,/\, |ll(\Theta^*)|$, where ll means the log-likelihood. Again, PathGLasso is significantly faster than HUGE and QUIC in the full range of relative errors. This experiment indicates that the choice of stopping criterion did not affect our results.

5.5 Interpretation of the Learned Network

We first checked whether the constraints imposed by the Reactome pathways improved the generalization performance of the learned network. We computed the *test* log-likelihood of the network trained on the MILE data and tested on the GENTLES data. We compared the result with random pathways created by shuffling genes among the pathways, preserving the pathway sizes and the structure among the pathways. We observed that the test log-likelihood of the original Reactome pathways is significantly higher than those of random pathways (Figure 5.6B). This result indicates that the Reactome pathways capture relevant information about the underlying network structure among genes conserved in two independent datasets.

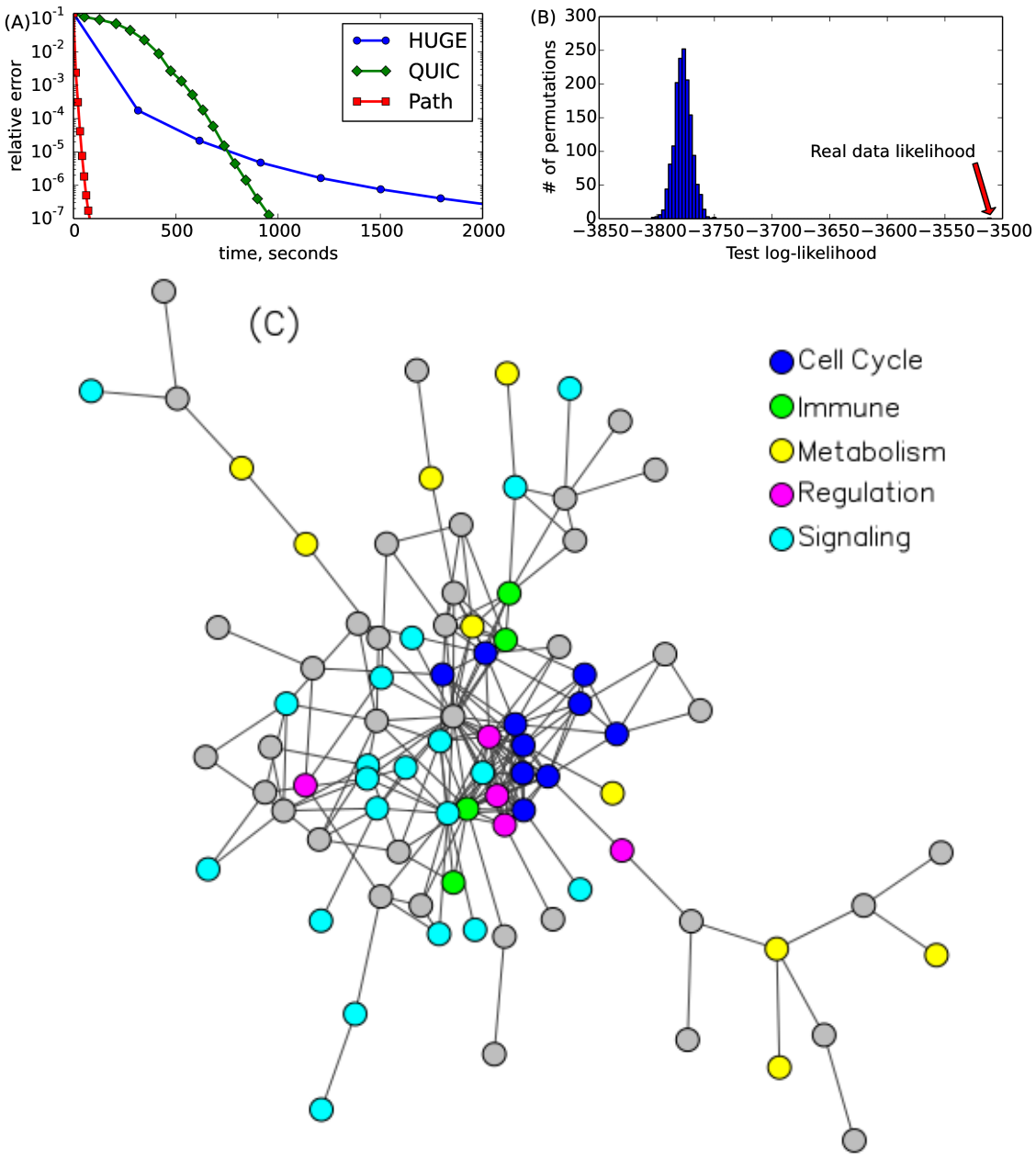


Figure 5.6: MILE data ($p = 4591, k = 156$). (A) Relative error vs time, (B) Test log-likelihood on Gentles dataset for random pathways, (C) Significant pathway interactions.

We will now show that PathGLasso provides a new way of interpreting the learned network. We identified pairs of pathways that have significant dependencies, by computing the sum of the magnitude of the edge weights that connect them, and comparing that with the quantity obtained from simulated 1500 data sets. Figure 5.6C shows a graphical representation of the dependencies among pathways. Interestingly, all of the cell cycle-related pathways are tightly connected with each other. Cancer is characterized by uncontrolled growth of cells, which is caused by deregulation of cell cycle processes [48]. One of the most densely-connected pathways is the "Cell Cycle Check Points" pathway, which is known to play a role in the central process of cancer progression by tightly interacting with many other pathways involved in the immune system, metabolism, and signaling [48].

5.6 Summary

In this chapter, we have introduced the pathway-constrained sparse inverse covariance estimation problem and a novel learning algorithm to solve it, called PathGLasso. We showed that our algorithm can be orders of magnitude faster than state-of-the-art competitors. We demonstrated that PathGLasso can leverage prior knowledge from curated biological pathways. PathGLasso uses an off-the-shelf algorithm for solving a standard graphical lasso problem as a subroutine, thus it will benefit from future performance improvements in the graphical lasso algorithms.

The work in this chapter was first reported in [92]. Implementation of PathGLasso is available under an open source license at `https://github.com/maximsch2/pglasso`.

Chapter 6

UNCOVERING NETWORK-PERTURBED GENES IN PUBLIC CANCER EXPRESSION DATASETS

6.1 Introduction

Genes do not act in isolation but instead work as part of complex networks to perform various cellular processes. Many human diseases including cancer are caused by *dysregulated* genes, with underlying DNA or epigenetic mutations within the gene region or its regulatory elements, leading to *perturbation* (topological changes) in the network [254, 214, 3, 14, 150, 161, 226]. This can ultimately impair normal cell physiology and cause disease [135, 32, 122, 105]. For example, cancer driver mutations [46, 40, 16, 13, 180, 245, 35, 56] on a transcription factor can alter its interactions with many of the target genes that are important in cell proliferation (**Figure 6.1A**). A key tumor suppressor gene can be bound by different sets of transcription factors between cancer and normal cells, which leads to different roles [161, 228, 255, 257, 264] (**Figure 6.1B**). Recent studies stress the importance of identifying the perturbed genes that create large topological changes in the gene network between disease and normal tissues as a way of discovering disease mechanisms and drug targets [135, 32, 21, 220, 103, 197]. However, most existing analysis methods that compare expression datasets between different conditions (e.g., disease vs. normal tissues) focus on identifying the genes that are *differentially expressed* [242, 75, 260]. For example, a recent review paper on biological network inference [174] emphasized that there is a lack of methods that focus on inferring the *differential network* between different conditions (e.g., distinct species, and disease conditions).

Several recent studies compare gene networks inferred between conditions based on expression datasets [254, 26, 5, 87, 99, 252, 261]. They fall into three categories: 1) Network

construction based on prior knowledge: West et al. (2012) computes the local network entropy, based on the protein interaction network from prior knowledge and expression datasets from cancer and normal tissues [254]. 2) Pairwise correlation-based networks: Guan et al. (2013) [99] proposed the *local network similarity* (LNS) method to compare the pairwise Pearson's correlation matrices of all genes between two conditions. Still other authors compared pairwise correlation coefficients for all gene pairs between conditions with different correlation measures including t-test p-values [26, 5, 252]. 3) Learning a condition-specific conditional dependency network for each condition and comparing the networks between conditions: Gill et al. (2010) proposed a method, called PLSNet, that fits a partial least squares model to each gene, computes a connectivity scores between genes, and then calculates the L_1 distance between score vectors to estimate network perturbation[87]. Zhang et al. (2009) proposed a differential dependency network (DDN) method that uses *lasso regression* to construct networks, followed by permutation tests to measure the significance of the network differences [261].

There have been approaches to identify dysregulated genes in cancer by utilizing multiple types of molecular profiles, not based on network perturbation across disease states estimated based on expression data. Successful examples use a linear model to infer each gene expression model based on copy number variation, DNA methylation, ChIP-seq, miRNAs or mRNA levels of transcription factors [155, 217, 7]. The advantages of the aforementioned methods that take only expression datasets as input to identify perturbed genes are in their applicability to diseases for which only expression data are available. In this chapter, we focus on identifying perturbed genes purely based on gene expression datasets representing distinct states, and compare our method with existing method, LNS, D-score and PLSNet.

We present a new computational method, called DISCERN (**DI**fferential **S**pars**E** **R**egulatory **N**etwork), to identify *perturbed* genes, i.e. the genes with differential connectivity between the condition specific networks (e.g., disease versus normal). DISCERN takes two expression datasets, each from a distinct condition, as input, and computes a novel *perturbation score* for each gene. The perturbation score captures how likely a given gene has a distinct set

of regulators between conditions (**Figure 6.1A**). The DISCERN method contains specific features that provide advantages over existing approaches: 1) DISCERN can distinguish direct associations among genes from indirect associations more accurately than methods that focus on marginal associations such as LNS; 2) DISCERN uses a penalized regression-based modeling strategy that allows efficient inference of genome-wide gene regulatory networks; and 3) DISCERN uses a new likelihood-based score that is more robust to the expected inaccuracies in local network structure estimation. We elaborate on these three advantages below:

First, DISCERN infers gene networks based on *conditional dependencies* among genes - a key type of probabilistic relationship among genes that is fundamentally distinct from correlation. If two genes are conditionally dependent, then by definition, their expression levels are still correlated even after accounting for (e.g., regressing out) the expression levels of all other genes. Thus, conditional dependence relationship is less likely to reflect transitive effects than mutual correlation, and provides stronger evidence that those genes are functionally related. These functional relationships could be regulatory, physical, or other molecular functionality that causes two genes expression to be tightly coupled. As a motivating example, assume that the expression levels of genes '3' and '5' are regulated by gene '1' in a simple 7-gene network (**Figure 6.1A**). This implies that the expression level of gene '1' contains sufficient information to know the expression levels of genes '3' and '5'. In other words, genes '3' and '5' are *conditionally independent* from each other and from the rest of the network given gene '1'.

Second, DISCERN uses an efficient neighborhood selection strategy based on a penalized regression to enable the inference of a genome-wide network that contains tens of thousands of genes. Penalized regression is a well established technique to identify conditional dependencies [170]. Inferring the conditional dependence relationships from high-dimensional expression data (i.e., where the number of genes is much greater than the number of samples) is a challenging statistical problem, due to a very large number of possible network structures among tens of thousands of genes. Unlike pairwise correlation, the conditional

dependence between '1' and '2' cannot be measured based on just the expression levels of these two genes. We should consider the possible networks among all genes and find the one that best explains the expression data. This involves both computational and statistical challenges. To make this process feasible, DISCERN uses a sparse regression model for each gene to select neighbors in the network [170, 251]. The use of a scalable method to infer a genome-wide conditional dependence network is a key distinguishing feature of the DISCERN method.

Finally, one of the most novel features of DISCERN is the ability to avoid the overestimation of the degree of network perturbation due to dense correlation among many genes. Revisiting the 7-gene network example (**Figure 6.1A**), assume that genes '5' and '7' are highly correlated to each other, in which case a penalized regression that imposes a sparsity penalty, such as the *lasso* method, may arbitrarily select one of them. This can result in a false positive edge between genes '1' and '7' instead of '1' and '5'. This may lead to overestimation of the perturbation of gene '1' (**Figure 6.1A**). Our network perturbation score overcomes this limitation by measuring the network differences between conditions based on the likelihood when the estimated networks are swapped between conditions - not based on the differences in topologies of the estimated networks. We demonstrate the effectiveness of this feature by comparing with methods based on the topology differences of the estimated networks.

We evaluated DISCERN on both synthetic and gene expression data from three human cancers: acute myeloid leukemia (AML), breast cancer (BRC), and lung cancer (LUAD). Integrative analysis using DISCERN on epigenomic data from the Encyclopedia of DNA Elements (ENCODE) project leads to hypotheses on the mechanisms underlying network perturbation (**Figure 6.1C**).

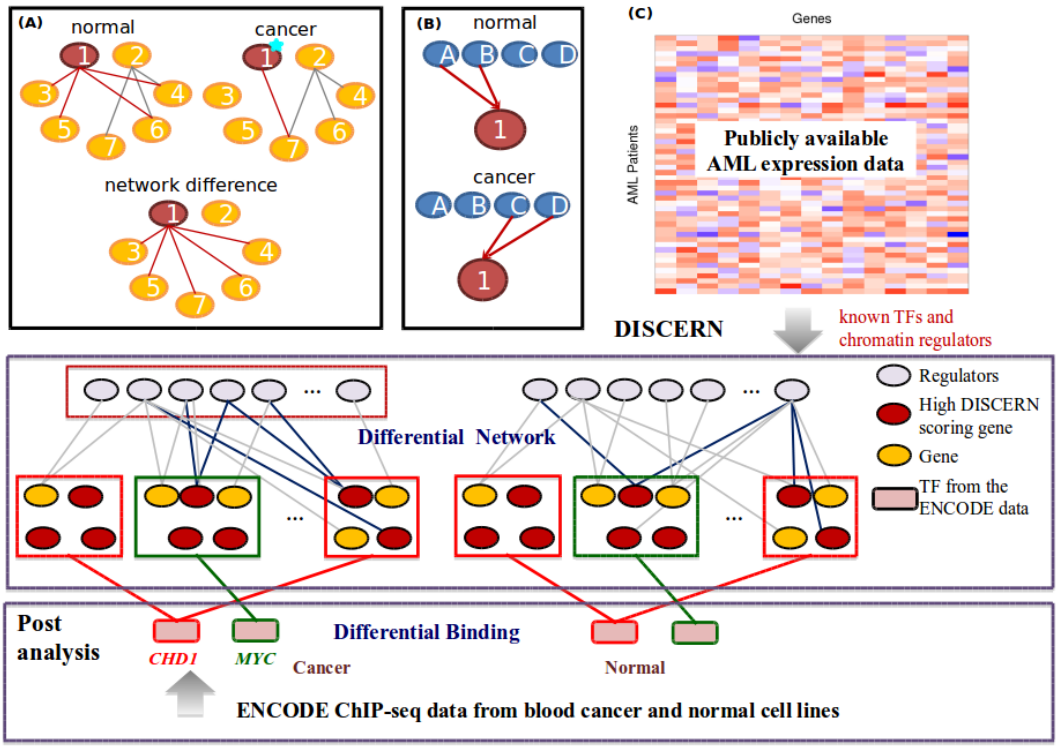


Figure 6.1: (A) A simple hypothetical example that illustrates the perturbation of a network of 7 genes between disease and normal tissues. One possible cause of the perturbation is a cancer driver mutation on gene '1' that alters the interactions between gene '1' and genes '3', '4', '5', and '6'. (B) One possible cause of network perturbation. Gene '1' is regulated by different sets of genes between cancer and normal conditions. (C) The overview of our approach. DISCERN takes two expression datasets as input: an expression dataset from patients with a disease of interest and another expression dataset from normal tissues (top). DISCERN computes the network perturbation score for each gene that estimates the difference in connection between the gene and other genes between disease and normal conditions (middle). We perform various post-analyses to evaluate the DISCERN method by comparing with alternative methods, based on the importance of the high-scoring genes in the disease through a survival analysis and on how well the identified perturbed genes explain the observed epigenomic activity data (bottom).

6.2 Results

6.2.1 Method overview

Here, we describe the DISCERN method, referring to the Methods for a full description. We postulate that a gene can be perturbed in a network largely in two ways: A gene can change how it influences other genes (**Figure 6.1A**), for example, a driver mutation on a transcription factor can affect cell proliferation pathways [46, 40, 16, 13, 180, 245, 35, 56]. A gene can change the way it is influenced by other genes, a common example being when a mutated (genetically or epigenetically) gene acquires a new set of regulators, which occurs frequently in development and cancer [161, 228, 255, 257, 264] (**Figure 6.1B**). Identifying the genes that are responsible for large topological changes in gene networks could be crucial for understanding disease mechanisms and identifying key drug targets [135, 32, 21, 220, 103, 197]. However, most current methods for identifying genes that behave differently in their expression levels between diseased and normal tissues focus on *differential expression*[242, 75, 260], rather than *differential connection* with other genes in a gene expression network (**Figure 6.1A**).

We model each gene's expression level using a sparse linear model (*lasso* regression): let $y_i^{(s)}$ be expression levels of gene i in an individual with state s, cancer ($s = c$) or normal ($s = n$), modeled as: $y_i^{(s)} \approx \sum_{r=1}^{p} w_{ir}^{(s)} x_r^{(s)}$. Here, $x_1, ...x_p$ denote *candidate regulators*, a set of genes known to regulate other genes, including transcription factors, chromatin modifiers or regulators, and signal transduction genes, which were used in previous work on network reconstruction approaches [152, 127, 215, 85] (Supplementary Table 1 from [93]). Linear modeling allows us to capture conditional dependencies efficiently from genome-wide expression data containing tens of thousands genes. Naturally, a zero weight w_{ir} indicates that a regulator r does not affect the expression of the target gene i. Sparsity-inducing regularization helps to select a subset of candidate regulators, which is a more biologically plausible model than having all regulators, and makes the problem well-posed in our *high-dimensional* setting (i.e., number of genes $\gg$ number of samples).

To determine the regulators for any given gene, we use a lasso penalized regression model [235] with the optimization problem for each lasso regression defined as:

$$\text{argmin}_{w_{i1}^{(s)},\ldots,w_{ip}^{(s)}} \sum_{j=1}^{n} \left(y_{ij}^{(s)} - \sum_{r=1}^{p} w_{ir}^{(s)} x_{rj}^{(s)} \right)^2 + \lambda \sum_{r=1}^{p} |w_{ir}^{(s)}|$$

, where $y_{ij}^{(s)}$ means the expression level of the i^{th} gene in the j^{th} patient in the s^{th} state, and $x_{rj}^{(s)}$ similarly means the expression level of the r^{th} regulator in the j^{th} patient in the s^{th} state. The second term, the L_1 penalty function, will zero out many irrelevant regulators for a given gene, because it is known to induce sparsity in the solution [235]. We normalize the expression levels of each gene and each regulator to be mean zero and unit standard deviation, a process called *standardization*, which is a standard practice before applying a penalized regression method [235, 236, 61, 211]. The difference in the weight vector between conditions, $w_i^{(n)}$ and $w_i^{(c)}$, can indicate a distinct connectivity of gene i with p regulators between the conditions. However, simply computing the difference of the weight vectors is unlikely to be successful, due to the correlation among the regulators. The *lasso*, or other sparsity-inducing regression methods, can arbitrarily choose different regulators between cancer and normal. Examining the difference in the weight vectors between conditions would therefore lead to overestimation of network perturbation.

Instead, DISCERN adopts a novel network perturbation score that measures how well each weight vector learned in one condition explains the data in a different condition. This increases the robustness of the score to correlation among regulators, as demonstrated in the next section. We call this score the DISCERN score, defined as

$$\text{DISCERN}_i = \frac{\text{-log-likelihood computed using learned weights } w^{(s)} \text{ in a different condition s'}}{\text{-log-likelihood computed using learned weights } w^{(s)} \text{ in the same condition s}}$$

. This is equivalent to

$$\text{DISCERN}_i = \frac{err_i(c,n) + err_i(n,c)}{err_i(c,c) + err_i(n,n)}$$

where $err_i(s,s') = \frac{1}{n_s} ||y_i^{(s)} - \sum_{r=1}^{p} w_{ir}^{(s')} x_r^{(s)}||_2^2$. Here n_s is the number of samples in the data from condition s. The numerator measures the error of predicting gene i's expression levels

in cancer (normal) based on the weights learned in normal (cancer). If gene i has different sets of regulators between cancer and normal, it is likely to have a high DISCERN score. The denominator plays an important role as a normalization factor, which is demonstrated by comparing with an alternative score, namely the D^0 score (**Figure 6.2A**), that uses only the numerator of the DISCERN score. We also compare with existing methods, such as LNS [99] and PLSNet [87], that compare the weight vectors between cancer and normal models where we demonstrate the advantages of the likelihood-based model that DISCERN uses.

6.2.2 Comparison with previous approaches on synthetically generated data

In order to systematically compare DISCERN with alternative methods in a controlled setting, we performed validation experiments on 100 pairs of synthetically generated datasets representing two distinct conditions. Each pair of datasets contains 100 variables drawn from the multivariate normal distribution with zero mean and covariance matrices Σ_1 and Σ_2. We divided 100 variables into the following three categories: 1) variables that have different sets of edge weights with other variables across two conditions, 2) variables that have exactly the same sets of edge weights with each other across the conditions, and 3) variables not connected with any other variables in the categories 2) and 3) in both conditions. For example, in **Figure 6.1A**, '1' is in category 1). '2', '4', '6', and '7' are in category 2), and '3' and '5' is in category 3). We describe how we generated the network edge weights (i.e., elements of Σ_1^{-1} and Σ_2^{-1}) among the 100 variables in more detail in Methods.

We compared DISCERN with 4 alternative methods to identify perturbed genes: LNS [99], D-score [252], PLSNet [87], and D^0 that uses only the numerator of the DISCERN score. Here, we do not compare with the methods to identify differentially expressed genes, such as ANOVA, because the synthetic data were generated from a zero mean Gaussian distribution. We note that the PLSNet method uses empirical p-values as the network perturbation scores, where the empirical p-value for each gene is estimated from permutation tests that generate the null distribution of the gene's score [87]. All the other methods, such as DISCERN, LNS, and D-score, do not require permutation tests (see Methods for details). To show that

DISCERN outperforms existing methods and those that use the empirical p-values obtained through permutation tests as the network perturbation scores, we developed the following methods for comparison: LNS, D-score, and D^0 followed by permutation tests to compute the empirical p-values, called pLNS, pD-score, and pD^0, respectively.

The average receiver operating characteristic (ROC) curves across 100 pairs of datasets for these methods (**Figure 6.2A**) show that DISCERN significantly outperforms all the other 7 methods – 3 existing methods (LNS, D-score, and PLSNet), and 4 methods we created for comparison (D^0, pD^0, pLNS, and pD-score). Except DISCERN, PLSNet performs the best among all existing methods. However, its run time grows too quickly as the number of variables increases, which makes it two to three orders of magnitude slower than DISCERN when run on larger data (**Figure 6.2B**). PLSNet was too slow to run on genome-scale data and therefore we did not use it for the subsequent experiments on genome-wide gene expression data from cancer patients.

We note that DISCERN does not need permutation tests to generate the null distribution of the score for each gene. All other methods improve when the empirical p-values from permutation tests are used, which indicates that the gene-level bias on the magnitude of the raw scores hurts their performance to identify perturbed genes. DISCERN significantly outperforms D^0 that uses only the numerator of the DISCERN score, which indicates that the denominator of the DISCERN score plays a role to normalize the score such that the scores of different genes can be compared to each other. Computing the empirical p-value for each gene based on the gene-specific null distribution obtained through permutation tests is not feasible on genome-wide data. To obtain a p-value of 0.05 after Bonferroni correction, we need at least $(1/0.05 \times p)$ permutation tests per gene, where p is the total number of genes, and $(1/0.05 \times p^2)$ permutation tests in total. When $p = 20,000$, this number is (4×10^9) permutation tests, which is not feasible even when using multiple processors at a reasonable cost. This is demonstrated in **Figure 6.2B** that shows the run time of PLSNet, a permutation test-based method, when applied to data containing a varying number of genes (p).

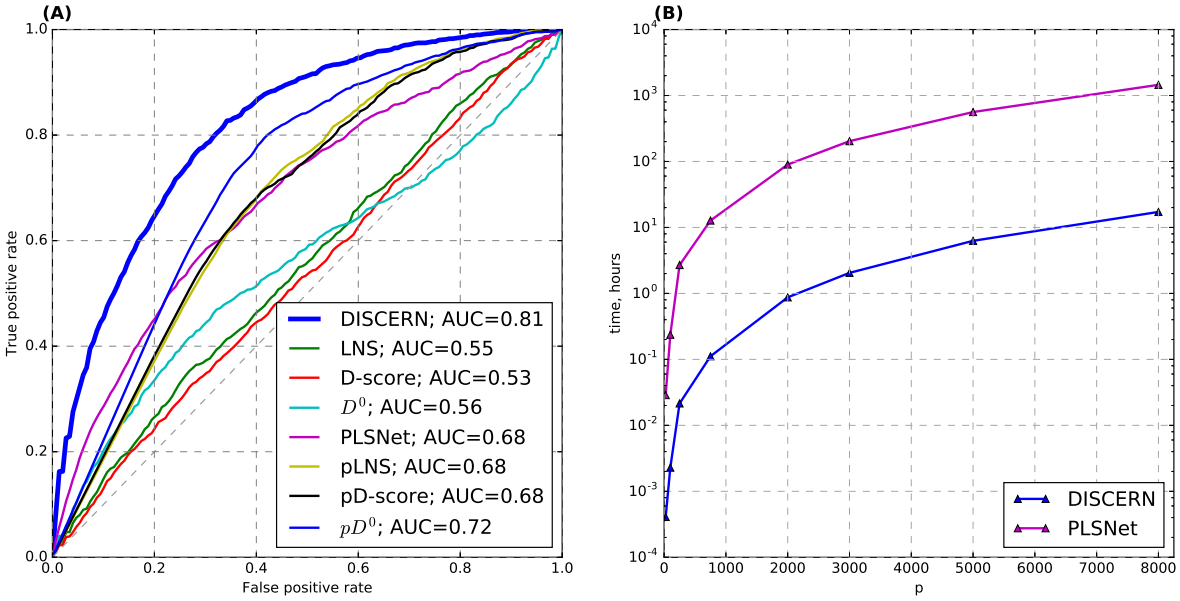


Figure 6.2: (A) Average receiver operating characteristic (ROC) curves from the experiments on synthetic data. We compare DISCERN with 7 alternative methods: 3 existing methods – LNS [99], D-score [252], and PLSNet [87] – and 4 methods we developed for comparison – pLNS, pD-score, D^0 and pD^0. (B) Comparison of the runtime (hours) between PLSNet and DISCERN for varying numbers of variables (p). The triangles mean the measured run times over specific values of p, and lines connect these measured run times. PLSNet uses the empirical p-values from permutation tests as scores, and DISCERN does not. For a large value of p, DISCERN is two to three orders of magnitude faster than PLSNet.

6.2.3 Comparison of methods on gene expression datasets

We used genome-wide expression datasets consisting of 3 acute myeloid leukemia (AML) datasets, 3 breast carcinoma (BRC) datasets and 1 lung adenocarcinoma (LUAD) dataset (**Table 1**). Details on the data processing are provided in Methods. To evaluate the performance of the DISCERN method, we compared DISCERN with existing methods that scale to over tens of thousands of genes: LNS [99] and D-score [252] that aim to estimate network perturbation, and ANOVA that measures differential expression levels between cancer and normal samples.

We first computed the DISCERN, LNS, D-score, and ANOVA scores in the 3 cancers based on the following datasets that contain normal samples: AML1, LUAD1 and BRC1 (**Table 1**). Then, we used the rest of the datasets to evaluate the performance of each method at identifying genes previously known to be important in the disease, for example, the genes whose expression levels are significantly associated with survival time in cancer. The value of the sparsity tuning parameter λ was chosen via cross-validation tests, a standard statistical technique to determine the value of λ [235]. For the chosen λ values, the overall average regression fit measured by cross-validation test R^2 was 0.493.

To remove any potential concern of the effect of standardization on genes with very low expression level, we first show that genes with low mean expression do not tend to have high enough DISCERN score to be considered in our evaluation in the next sections (see Supplementary Figure 1 from [93]). The Pearsons correlation between the mean expression before standardization and the DISCERN score ranges from 0.08 and 0.43 across datasets. Positive correlation is induced because genes with low mean expression tend to have lower DISCERN scores, indicating that genes whose expression are likely essentially noise would not be selected as high-scoring genes. To further reduce the potential concern of genes with low expression in RNA-seq data (LUAD), we applied the *voom* normalization method that is specifically designed to adjust for the poor estimate of variance in count data, especially for genes with low counts [149].

We assessed the significance of the DISCERN scores through a conservative permutation testing procedure, where we combined cancer and normal samples, and permuted the cancer/normal labels among all samples (more details in Methods). Unlike the gene-based permutation test described in the previous section, here, we generate a single null distribution for all genes, which requires a significantly less number of permutation tests (one million in this experiment). After applying false discovery rate (FDR) correction on these p-values, there are 1,351 genes (AML), 2,137 genes (BRC), and 3,836 (LUAD) genes whose FDR corrected p-values are less than 0.05. We consider these genes to be significantly perturbed genes (Supplementary Table 2 from [93]). The difference in these numbers of significant perturbed genes identified by DISCERN is consistent with a prior study that showed that lung cancer has a larger number of non-synonymous mutations per tumor than breast cancer, which has a larger number than AML [250].

Table 6.1: Gene expression datasets used in DISCERN experiments.

	Reference	# Genes	# of Tumors	# of Normal	Survival?	Platform	Accession
AML1	MILE, [104]	16853	541	73	No	Affy U133+2.0	GSE13159
AML2	Gentles, [86]	16853	515	0	Yes	Affy U133+2.0	GSE12417,GSE14468,GSE10358
AML3	Metzeler	11697	152	0	Yes	Affy U133A	GSE12417
BRC1	TCGA	10809	529	61	Yes	Agilent G4502A	Firehose 2013042100
BRC2	Metabric	10809	1981	0	Yes	Illumina HTv3	Synapse: syn1688369
BRC3	Oslo	10809	184	0	Yes	Illumina HTv3	Synapse: syn1688370
LUAD1	TCGA	17022	504	57	Yes	Illumina HiSeq	Firehose 2015110100

6.2.4 Top scoring DISCERN genes in AML reveal known cancer drivers in AML

The 1,351 genes that were predicted to be significantly perturbed between AML samples and normal non-leukemic bone marrow samples were enriched for genes causally implicated previously in AML pathogenesis (Supplementary Table 2 from [93]). This include a number of genes that we and others have previously identified as being aberrantly activated in leukemic stem cells such as BAALC, GUCY1A3, RBPMS, and MSI2 [86, 133, 231]. This is consistent with over-production of immature stem-cell like cells in AML, which is a major driver of poor prognosis in the disease. Prominent among high-scoring DISCERN genes were many HOX family members, which play key roles in hematopoietic differentiation and in the pathogenesis of AML [4]. HOX genes are frequently deregulated by over-expression in AML, often through translocations that result in gene fusions. The highest ranked gene in AML by DISCERN is HOXB3 which is highly expressed in multipotent hematopoietic progenitor cells for example. Thirteen (out of 39 known) HOX genes are in the 1,351 significantly perturbed genes (p-value: 5.99×10^{-6}).

When compared to known gene sets from the Molecular Signature Database (MSigDB) [229] in an unbiased way, the top hit was for a set of genes that are down-regulated by NPM1 (nuclephosmin 1) mutation in AML (Supporting Information 1 from [93]). NPM1 is one of three markers used in AML clinical assessment; the others are FLT3 and CEBPA that are significantly perturbed genes identified by DISCERN as well. Mutation leads to aberrant cytoplasmic location of itself and its interaction partners, leading to changes in downstream transcriptional programs that are being captured by DISCERN. Also highly significant were genes highly expressed in hematopoietic stem cells [125]. Among these were key regulators of hematopoietic system development such as KIT, HOXA3, HOXA9, HOXB3 (with the latter homeobox genes also implicated in AML etiology), as well as FLT3 which plays a major role in AML disease biology, with its mutation and constitutive activation conferring significantly worse outcomes for patients [141]. Comparison to Gene Ontology (GO) categories identified dysregulation of genes involved in hemostasis and blood coagulation, a key clinical

presentations of AML. Furthermore, GTPase activity/binding and SH3/SH2 adaptor activity were enriched among high-scoring DISCERN genes. These are pertinent to AML due to previously noted high expression in AML leukemic stem cells of GUCY1A3 and SH3BP2, both identified as perturbed genes by DISCERN [86]. However, their function has not been examined in detail, suggesting that they are potential targets for further investigation as to their role in AML disease mechanisms. Several other highly significant enrichments were for AML subtypes that are driven by specific translocations, including MLL (mixed lineage leukemia) translocation with various partners, as well as t(8;21) translocations. The latter is of particular interest, since it is primarily a pediatric AML, whereas our network analysis uses purely adult AML samples indicating the potential to uncover putative mechanisms that generalize beyond the context of the immediate disease type.

6.2.5 Top scoring DISCERN genes in lung cancer reveal biological processes known to be important in lung cancer

There are 3,836 significantly perturbed genes identified by DISCERN in lung cancer (LUAD) (Supplementary Table 2 from [93]). The 3rd and 4th highest ranked genes are ICOS (inducible costimulator) and YWHAZ (14-3-3-zeta). Both genes have known roles in disease initiation or progression in lung cancer. Polymorphisms in ICOS have been associated with pre-disposition to non-small cell lung cancer [131], while over-expression of YWHAZ is known to enhance proliferation and migration of lung cancer cells through induction of epithelial-mesenchymal transitions via beta-catenin signaling [41]. GIMAP5 (GTPase IMAP Family Member 5), another high scoring LUAD gene (11th), is consistently repressed in paired analyses of tumor vs normal lung tissue from the same patient, and encodes an anti-apoptotic protein [218]. Down-regulation of GIMAP5 in lung tumors therefore potentially facilitates their evasion of programmed cell death, one of the hallmarks of cancer.

Several of the GO biological categories enriched in 3,836 high-scoring DISCERN genes in LUAD (FDR-corrected p-value < 0.05) reflected metabolic and proliferative processes that are commonly de-regulated in solid tumors such as lung adenocarcinoma. Among these were

cellular response to stress, mitotic cell cycle, amino acid metabolism, and apoptosis (Supporting Information 1 from [93]). In fact the top-ranked gene was MCM7 (minichromosome maintenance protein 7), an ATP-dependent DNA helicase involved in DNA replication which has been implicated in carcinogenesis previously due to its function as a binding partner of PRMT6 [239]. Moreover, it was specifically identified as being a potential therapeutic target due to its over-expression in solid tumors relative to normal tissues. The high ranking of genes associated with apoptosis is consistent with the fact that there is often high rate of tumor cell death. Although the highly-ranked CARD6 (caspase recruitment domain family member 6) functions in apoptotic processes, it is also known as a regulator of downstream NF-$\kappa\beta$ signaling. Indeed, consistent with this, we found enrichment for NF-$\kappa\beta$ signaling pathway genes among high DISCERN-scoring genes in LUAD including NFKBIB (NF-$\kappa\beta$ inhibitor β) which inhibits the NF-$\kappa\beta$ complex by trapping it in the cytoplasm, preventing nuclear activation of its downstream targets. Although the role of NFKBIB in lung cancer has not been studied extensively, its related family member NFKBIA is known to be a silencer in non-small-cell lung cancer patients with no smoking history, suggesting that it could play some role in LUAD that arises through inherent genetic influences, or environmental insults other than smoking [81]. Levels of β-catenin have been known for some time to influence progression and poor prognosis in LUAD, potentially through its role in differentiation and metastasis from primary tumor sites [258]. We found that components of β-catenin degradation pathways - including most notably CTNNBIP1 (β-catenin interacting protein 1) - ranked among the most significant DISCERN genes in our LUAD analysis.

When comparing to other sets of genes in MSigDB, we also found targets of transcription factors including MYC, which is often de-regulated in solid tumors (either by mutation or copy number variation), and targets of the polycomb repressive complex gene EZH2. The developmental regulator EZH2 functions through regulation of DNA methylation [249], and has been implicated in B-cell lymphomas through somatic mutations [177], promotion of transformation in breast cancer [138], as well as progression in prostate cancer [246]. Interestingly, the most highly dys-regulated gene set identified by comparison to GO categories

in LUAD was one related to NGF (nerve growth factor)-TrkA signaling. There are a few reports on the relevance of this axis to cancers including neuroblastoma, ovarian cancer, and a possible role in promoting metastasis in breast cancer. However, its striking appearance as the most significant hit for high-ranking DISCERN genes suggests that it merits study in lung cancer.

6.2.6 Top scoring DISCERN genes in breast cancer reveal biological processes known to be important in breast cancer

Here, we did the functional enrichment analysis with 2,137 genes identified by DISCERN to be significantly perturbed in breast cancer (BRC) (Supplementary Table 2 from [93]). BRC showed perturbation of distinct genes and sets of genes in comparison to LUAD, as well as similarities. Again, these included GO biological processes that one would generically expect to be over-activated in a solid tumor, such as translation intiation, cell cycle, proliferation, and general cellular metabolic processes. As with LUAD, targets of MYC were enriched in high-scoring DISCERN genes in BRC. Another high-scoring group in BRC was comprised of genes that are highly correlated with each other, but with this relationship de-regulated by BRCA1 mutation [198]. Additional significant overlaps were identified with luminal A, luminal B, HER2-enriched, and basal-like breast cancer subtype-specific genes that are associated with clinical outcomes [188], and genes associated with ER-positive breast cancer [244]. The 3rd highest ranked DISCERN gene was BRF2 (TFIIB-related factor 2). BRF2 is a known oncogene in both breast cancer and lung squamous cell carcinoma, and a core RNA polymerase III transcription factor that senses and reacts to cellular oxidative stress [90]. A GO category associated with NGF (nerve growth factor)-TrkA signaling shows the highest overlap with DISCERN genes in BRC (p-value: 3.16×10^{-104}). NGF-TrkA signaling is upstream of the canonical phosphatidylinositol 3-kinase (PI3K)AKT and RASmitogen-activated protein kinase (MAPK) pathways, both of which impinge on cell survival and differentiation. In the context of breast cancer, over-expression of TrkA has been connected to promoting growth and metastasis, as an autocrine factor, presumably due to its influence on PI3K-AKT

and RAS/MAPK [143]. TrkA is reportedly over-expressed in breast carcinoma relative to normal breast tissue in a majority of cases [2], supporting the high-ranking of genes in this pathway by DISCERN. Taken together, these results indicate that DISCERN highly ranks genes that are connected to known phenotypic and survival-associated processes in breast cancer. However, intriguingly the top DISCERN gene was CLNS1A (chloride nucleotide-sensitive channel 1A). This chloride channel gene has not, to our knowledge, been implicated in pathogenesis in any cancer, although it is a member of the BRCA1-related correlation network noted above. In fact there appear to have been few studies of its function although Entrez gene notes that it performs diverse functions.

6.2.7 DISCERN scores reveal survival-associated genes across multiple cancer types

In this section, we focus on the quantitative assessment of DISCERN and the comparison with LNS and D-score in terms of how much the identified genes are enriched for genes implicated to be important in the disease. Specifically, genes whose expression levels are significantly associated positively or negatively with survival time are often considered to be associated with tumor aggression. Identifying such genes has been considered as an important problem by a number of authors, where breast cancer was one of the first cancers to show promise in terms of identifying clinically relevant biomarkers [185, 243]. Here, we evaluated DISCERN based on how well it reveals survival-associated genes identified in an available independent dataset.

We chose the datasets with measures of patient prognosis: AML2, BRC2, and LUAD1. AML2 and BRC2 were not used for computing any scores (DISCERN, LNS, D-score, and ANOVA). For each of these datasets we computed the survival p-values based on the Cox proportional hazards model [233] measuring the association between each gene's expression level and survival time. We defined survival-associated genes as the genes whose expression levels are associated with survival time based on the Cox proportional hazards model (p-value < 0.01) (Supplementary Table 3 from [93]).

We considered the genes whose DISCERN scores are significantly high at FDR corrected

p-value < 0.05 in each cancer: 1,351 genes (AML), 2,137 genes (BRC), and 3,836 genes (LUAD). We first computed the Fisher's exact test p-values to measure the statistical significance of the overlap between these significantly perturbed genes and survival-associated genes in each of three cancers. For each cancer, we compared with existing methods to detect network perturbation – LNS and D-score – when exactly the same number of top-scoring genes were considered (**Figure 6.3A-C**). Since these numbers of genes were chosen specifically for DISCERN, there is a chance that LNS and D-score would show a higher enrichment for survival-associated genes if different numbers of top-scoring genes were considered. As discussed in the previous section, performing the gene-based permutation tests to estimate the confidence of each gene's score in genome-wide data is not feasible. Instead, we compared the Fisher's exact test p-values of the three methods across a range of numbers of top-scoring genes from 0 to N **Figure 6.3D-F**. It is pretty clear that neither LNS nor D-score would be better than DISCERN in revealing survival-associated genes, even when different numbers of top-scoring genes were considered across all cancer types.

ANOVA is a well-established method to identify *differentially expressed* genes across distinct conditions; DISCERN LNS, and D-score are methods to identify *differentially connected* genes across conditions. Therefore, the purpose of the comparison with ANOVA is not to evaluate DISCERN in identifying survival-associated genes as perturbed genes. The purpose is to compare between differentially expressed genes (that are commonly considered important) and perturbed genes estimated by the three methods (DISCERN, LNS, and D-score), in terms of the enrichment for genes with potential importance to the disease. For ANOVA, in **Figure 6.3A-C**, we considered 8,993 genes (AML), 7,922 genes (BRC) and 13,344 genes (LUAD) that show significant differential expression between cancer and normal samples at FDR corrected p-value < 0.05. The perturbed genes identified by DISCERN are more associated with survival than differentially expressed genes captured by ANOVA in AML and LUAD (**Figure 6.3**).

In addition to the comparison with other methods – LNS and D-score – we also compare with frequently mutated genes and genes annotated to be involved in the respective

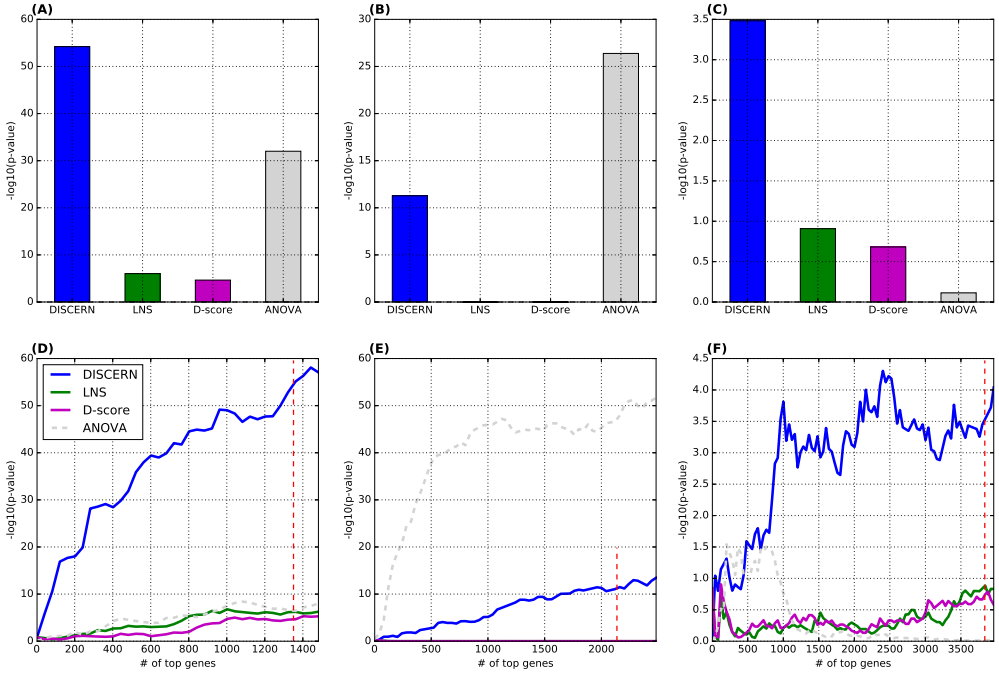


Figure 6.3: The significance of the enrichment for survival-associated genes in the identified perturbed genes. We compared DISCERN with LNS and D-score based on the Fisher's exact test p-value that measures the significance of the overlap between N top-scoring genes and survival-associated genes in each of three cancers. (A)-(C) We plotted $-\log_{10}$(p-value) from the Fisher's exact test when N top-scoring genes were considered by each method in 3 datasets: (A) AML ($N = 1,351$), (B) BRC ($N = 2,137$), and (C) LUAD ($N = 3,836$). For ANOVA, we considered 8,993 genes (AML), 7,922 genes (BRC) and 13,344 genes (LUAD) that show significant differential expression at FDR corrected p-value < 0.05. (D)-(F) We consider up to 1,500 (AML), 2,500 (BRC), and 4,000 (LUAD) top-scoring genes in each method, to show that DISCERN is better than LNS and D-score in a range of N value. The red-colored dotted line indicates 1,351 genes (AML), 2,137 genes (BRC), and 3,836 genes (LUAD) that are identified to be significantly perturbed by DISCERN (FDR < 0.05). We compare among the 4 methods consisting of 3 methods to identify network perturbed genes (solid lines) and ANOVA for identifying differentially expressed genes (dotted line) in 3 cancer types.

cancer. We considered the following three gene sets: 1) a gene set constructed based on the gene-disease annotation database, Malacards [204], 2) genes known to have cancer-causing mutations based on the Cancer Gene Census [82], and 3) genes predicted to have driver mutations identified by MutSig [150] applied to The Cancer Genome Atlas (TCGA) data for the respective cancer type. The Malacards (gene set #1) and TCGA driver gene sets (#3) are generated for each cancer type – AML, breast cancer, or lung cancer. For example, for Malacards, we used the genes that are annotated to be involved in AML in Malacards to compare it with DISCERN genes identified in AML. Similarly, for the TCGA driver gene sets (#3), we used the AML TCGA data to identify the frequently mutated genes that are likely driver genes, and compared with high DISCERN-scoring genes in AML. We used the breast cancer TCGA data for BRC, and lung cancer TCGA data for LUAD. The Cancer Gene Census (CGC) gene set is a rigorously defined set of genes with multiple sources of evidence that its genes are cancer drivers in a single or multiple cancers.

For each cancer type, we compared these three sets of genes with the perturbed genes identified by DISCERN – 1,351 (AML), 2,137 (BRC), and 3,836 (LUAD) genes with high DISCERN scores – on the basis of the significance of the enrichment for survival-associated genes. Supplementary Figure 2 in [93] shows that the perturbed genes identified by DISCERN are more significantly enriched for survival-associated genes.

6.2.8 Prognostic model based on high DISCERN-scoring genes

In this section, we evaluated the DISCERN score based on how well it identifies genes that are predictive of patient prognosis. Here, we test the possibility of using the network perturbed genes identified by DISCERN as prognostic markers. For the cancer types with at least three data sets (AML and BRC; see **Table 1**), we construct a survival time prediction model using the genes with significant DISCERN scores (AML: 1,351 genes, BRC: 2,137 genes) identified based on one data set (Data # 1: AML1 and BRC1) as described in the previous subsection. Then, we trained the prediction model using one of the other datasets (Data #2: AML2 and BRC2) not used for the computation of the DISCERN score. Finally, we tested the

prediction accuracy on the third data set (Data #3: AML3 and BRC3).

We controlled for clinical covariates whose data are available – age in case of AML and age, grade and subtype in case of BRC – by adding them as unpenalized covariates into our elastic net Cox regression model. We trained the Cox regression model using Data #2 and tested the survival prediction model on Data #3. Since we evaluated the survival prediction in separate data (AML3 and BRC3) that were not used when training the survival prediction model, using more predictors, e.g., by adding clinical covariates, does not necessarily improve the prediction performance. Adding more predictors often leads to a higher chance of overfitting. Our survival prediction model based on the high DISCERN-scoring genes works at least as well as models based on the genes contained in the previously established prognosis markers, such as Leukemic Stem Cell score (LSC) [86] for AML and MammaPrint signature (with ∼70 genes) [88] for BRC, as shown in **Figure 6.4**. The c-index in AML is 0.669 with standard error (se) being 0.031 (**Figure 6.4B**); in BRC, the c-index is 0.668 (se: 0.027) (**Figure 6.4D**). The DISCERN-based expression marker with clinical covariates makes better predictions than when clinical covariates alone are used.

6.2.9 DISCERN explains epigenomic activity patterns in cancer and normal cell types more accurately than alternative methods

One of the possible mechanisms underlying network perturbation identified in gene expression datasets representing different conditions (e.g., cancer and normal) is the following: A transcription factor (TF) 'X' binds to a gene 'Y"s promoter or its enhancer region in cancer but not in normal (or vice versa). Then, 'X' or its co-regulator could be an expression regulator for 'Y' in cancer but not in normal (or vice versa), and Y is identified as a perturbed gene (i.e., a high DISCERN-scoring genes). It is possible that 'X"s binding information is not available and 'X"s protein level is not reflected in its mRNA expression level; thus we cannot expect the DISCERN score of a gene inferred from expression data to be perfectly correlated with whether the gene has a differential biding of a certain TF, inferred from ChIP-seq or DNase-seq data. However, the degrees of correlation between the network perturbation score

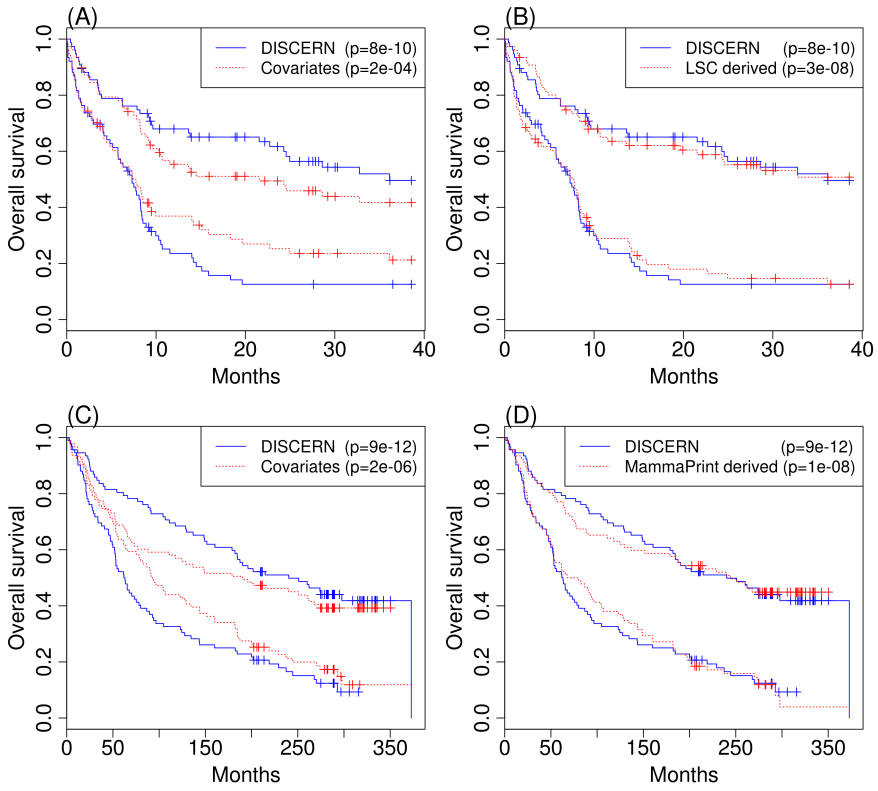


Figure 6.4: The Kaplan-Meier plot showing differences in the survival rate measured in AML3 (A and B) and BRC3 (C and D) between the two patient groups with equal size, created based on the predicted survival time from each prediction model. We consider the model trained based on the top N (N=1,351 for AML; N=2,137 for BRC) DISCERN-scoring genes and clinical covariates (blue), and the model trained based on only clinical covariates (red) (panels A and C for AML3 and BRC3, respectively). (B) The panel shows the comparison with the model trained using genes comprising 22 genes previously known prognostic marker, called LSC [86], along with the clinical covariates (red). (D) The panel shows the comparison with the model trained using 67 genes from the MammaPrint prognostic marker (70 genes) [88] along with the clinical covariates. We used 67 genes out of 70 genes that are present in our BRC expression datasets. P-values shown in each plot are based on the logrank test (red).

(DISCERN, LNS or D-score) of a gene and whether a TF differentially binds to the gene can be a way to evaluate the network perturbation scoring methods.

To determine whether or how much our statistical estimates of network perturbation reflects perturbation of the underlying TF regulatory network, we queried epigenomic data from ENCODE project. Two of the ENCODE cell lines – NB4 (an AML subtype [145]) and CD34+ (mobilized CD34 positive hematopoietic progenitor cells) – are closest to AML and normal conditions, and the DNase-seq data from these cell lines are available. We used the DNase-seq data from NB4 and the position weight matrices (PWMs) of 57 TFs available in the JASPAR database [210] to find the locations of the PWM motifs that are on the hypersensitive regions. This is a widely used approach to estimate active binding motifs using DNase-seq data, when ChIP-seq data are not available. We identified the locations of these PWM motifs on the hg38 assembly by using the FIMO [91] method (p-value $\leq 10^{-5}$). We then intersected these motif locations with hypersensitive regions identified by the DNase-seq data for each TF. We repeated for the other cell line CD34+.

For each TF, we measured how well the DISCERN score of a gene can predict the *differential* binding of the TF in active enhancer regions (marked by H3K27Ac) within 15kbs of the transcription start site (TSS) of the gene (**Figure 6.5A-C**) and 5kb of the gene between blood cancer and normal cell lines (NB4 and CD34+) (Supplementary Figure 3 from [93]). We show that the DISCERN score can reflect differential binding of most of the TFs better than existing methods to identify network perturbation (LNS and D-score) and a method to identify differentially expressed genes (ANOVA). As a way to summarize these results across all 57 TFs, we computed the Pearson's correlation between the score of each gene and the proportion of TFs that differentially bind to that gene out of all TFs that bind to that gene. **Figure 6.5D** shows that DISCERN detects genes with many TFs differentially bound between cancer and normal better than the other network perturbation detection methods (LNS and D-score) and ANOVA.

Considering hypersensitive sites identified by DNase-seq data as the indication of "general" binding of TFs or other DNA-associated proteins, we assume that a gene is differentially

bound if there is a DNase signal within a 150bp window around its TSS in one condition (cancer or normal), but not in the other condition. We observe that the DISCERN scores of the genes that are differentially bound are significantly higher than those of the genes that are not (**Figure 6.5E**). These results suggest that DISCERN identifies possible regulatory mechanisms underlying network perturbation more accurately than existing network perturbation detection methods (LNS and D-Score) and a method for identifying differential expression levels (ANOVA).

As a specific example, STAT3 has been shown to differentially regulate the mRNA expression of BATF in myeloid leukemia but not in normal condition [216]. We found that STAT3 differentially binds to BATF in the AML cell line but not in the normal cell line based on our differential binding analysis using the DNase-seq/motif data, as described above (Supplementary Table 4 from [93]). Interestingly, DISCERN identifies BATF as a perturbed gene in AML (FDR corrected p-value < 0.05). DISCERN also identifies STAT3 as the strongest regulator for BATF in AML expression data, but STAT3 is not selected as an expression regulator in normal expression data (Supporting Information 1 from [93]). Interestingly, LNS and D-Score detect STAT3 as an expression regulator of BATF in both conditions, not as a differential expression regulator.

Two of the Tier 1 ENCODE cell lines – K562 (chronic myeloid leukemia cell line) and GM12878 (a lymphoblastoid cell line) – correspond to blood cancer and normal tissues as well [234]. Tier 1 data contain the largest number of TFs with ChIP-seq datasets, which allows us to perform this kind of analyses using ChIP-seq datasets for these TFs. We repeated the same analysis with these cell lines and showed similar results (see Supplementary Figures 4 and 5 from [93]).

6.2.10 Combining DISCERN with ENCODE data improves the enrichment of known pathways

Additionally, we investigated whether one can use DISCERN as a filtering step to increase the power in a pathway enrichment analysis. We consider hypersensitive sites identified

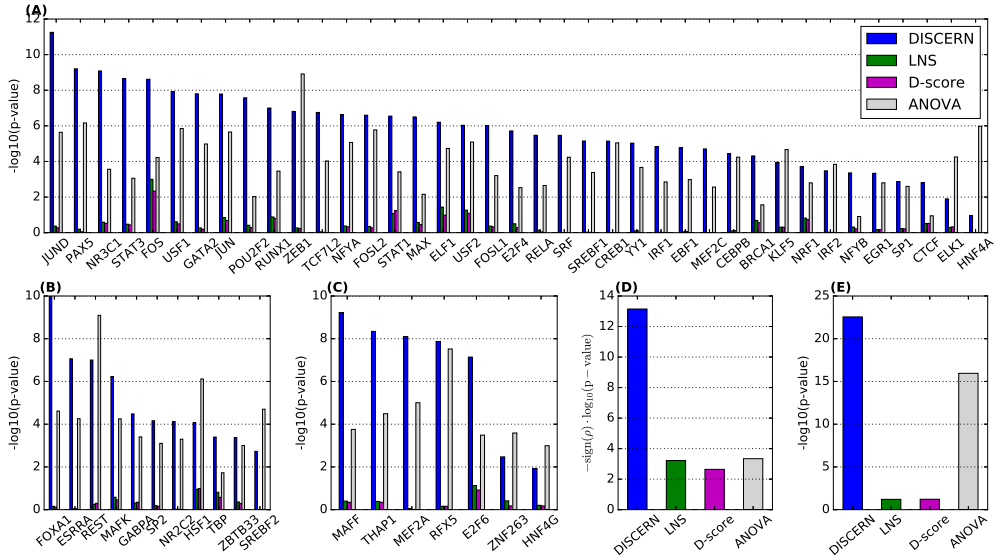


Figure 6.5: Kolmogorov-Smirnov test p-value measuring the significance of the difference in score between genes differentially bound by the corresponding transcription factor (TF) (x-axis) and those not differentially bound by the corresponding TF. We performed the one-sided test with an alternative hypothesis that differentially bound genes have higher scores; thus high $-\log_{10}$(p-value) means that high-scoring genes tend to show differential binding. The TFs are divided into the 3 sets: (A) TFs that are known to be associated with leukemia, (B) TFs that are known to be associated with cancer, and (C) TFs that are currently not known to be associated with cancer or leukemia, based on the gene-disease annotation database Malacards [204]. (D) Comparison of the p-values for the Pearson's correlation between the score of each gene and the proportion of differential TFs out of all TFs bound to the genes. (E) Kolmogorov-Smirnov test (one-sided as above) p-value measuring the significance of the difference in score between the genes with differential binding purely based on the DNase-seq data and those not. Here, a differentially bound gene is defined as a gene that has a DNase signal within a 150bp window around its TSS in one condition but not in the other condition.

by DNase-seq data as the indication of "general" binding of TFs or other DNA-associated proteins, and important regulatory events. As describe above, we identified differentially regulated genes between AML and normal cell lines (NB4/ CD34+) by identifying gene that have DNase-seq peaks within 150bp around the TSS in one condition (cancer or normal), but not in the other condition. There are 3,394 differentially regulated genes selected based on the DNase-seq data, of which 339 are significant DISCERN genes (Supporting Information 1 from [93]). Presumably, these disease specific targets should be enriched for pathways or categories that will help us understand mechanisms underlying the disease. Alternatively, some targets may be spurious, especially considering the use of cell lines that are not a perfect match to healthy and diseased bone marrow samples and experimental noise.

Here we attempt to identify differentially regulated genes between AML and normal samples, by integrating the information on the DNase-seq data (i.e., differentially bound genes) and significantly perturbed genes identified by DISCERN based on the expression datasets from AML samples and normal non-leukemic bone marrow samples. To show that combining these two pieces of information helps us to identify pathways that are specifically active in one condition not in the other, we compared the significance of the enrichment for Reactome pathways measured in fold enrichment between 1) 339 differentially bound DISCERN genes (intersection of 3,394 differentially bound genes and high DISCERN-scoring genes), and 2) 3,394 differentially bound genes. Supplementary Figure 6 in [93] shows that for most of the pathways, using the intersection of differentially bound and perturbed genes increases the fold enrichment compared to when differentially bound genes were used (Wilcox p-value $< 7 \times 10^{-5}$).

Among the pathways, 'platelet activation signalling and aggregation' shows significant improvement in fold enrichment: 1) when differentially bound DISCERN genes were used ($f = 2.9$; FDR q-value = 0.01), compared to 2) when differentially bound genes were used ($f = 1.03$). It has been shown that the interactions between platelets and AML cells have considerable effects on metastasis, and the various platelet abnormalities have been observed in AML and other leukemias [76]. G-alpha signalling-related pathways also show significant

boost in fold enrichment when DISCERN was used as a filtering mechanism for differentially bound genes. 'G_q signalling pathway' shows significant increase in fold enrichment: 1) when differentially bound DISCERN genes were used ($f = 2.16$;FDR q-value $= 0.05$), compared to 2) when differentially bound genes were used ($f = 0.92$). '$G_{12/13}$ signalling pathway' shows significant improvement in fold enrichment: 1) when differentially bound DISCERN genes were used ($f = 3.4$; q-value < 0.03), compared to 2) when differentially bound genes were used ($f = 1.5$). These pathways have been implicated in leukemias [222].

6.3 Methods

6.3.1 Data Preprocessing

Raw cell intensity files (CEL) for gene expression data in AML1, AML2, and AML3 were retrieved from GEO [11] and The Cancer Genome Atlas (TCGA). Expression data were then processed using MAS5 normalization with the Affy Bioconductor package [84], and mapped to Enztrez gene annotations [164] using custom chip definition files (CDF) [52], and batch-effect corrected using ComBat [126] implemented in package `sva` from CRAN.

BRC1 expression data were accessed through Broad Firehose pipeline (build 2013042100). We checked whether BRC1 processed by Firehose shows evidence of batch effects. We confirmed that the first three principal components are not significantly associated with the plate number (which we assumed to be a batch variable), which indicates no strong evidence of batch effects. BRC2 and BRC3 were accessed through Synapse (syn1688369, syn1688370). All probes were then filtered and mapped using the illuminaHumanv3.db Bioconductor package [67]. Probes mapped into the same genes were then collapsed by averaging if the probes being averaged were significantly correlated (Pearson's correlation coefficient greater than 0.7).

LUAD1 expression data were accessed through Broad Firehose pipeline (build 2015110100). Genes which had a very weak signal were filtered out of the LUAD1 data. We then applied the *voom* normalization method that is specifically designed to adjust for the poor estimate

of variance in count data, especially for genes with low counts [149]. The voom algorithm adjusts for this variance by estimating precision weights designed to adjust for the increased variance of observations of genes with low counts. This would stabilize the estimated distribution of RSEM values in the LUAD data, making it more normally distributed. Since LUAD data comes from different tissue source sites, we have applied batch-effect correction using ComBat.

For all datasets, only probes that are mapped into genes that have Entrez gene names were considered. **Table 6.1** shows the number of samples and genes used in each dataset. For AML1, BRC1, and LUAD1 that were used for score computation, we splitted each dataset into two matrices, one with only cancerous patients and one with normal patients. These matrices are normalized to 0-mean, unit-variance gene expression levels for each gene, before each network perturbation score (DISCERN, LNS, and D-score) was computed, which is a standard normalization step for accurately measuring the difference in the network connectivity. For methods that measure the differential expression levels (ANOVA), such normalization was not applied.

Lastly, candidate regulators are identified from a set of 3,545 genes known to be transcription factors, chromatin modifies, or perform other regulatory activity, which have been used in many studies on learning a gene network from high-dimensional expression data [152, 127, 215, 85] (Supplementary Table 1 from [93]).

6.3.2 DISCERN score

DISCERN uses a likelihood-based scoring function that measures for each gene how much likely the gene is differently connected with other genes in the inferred network between two conditions (e.g., cancer and normal). We model each gene's expression level based on a sparse linear model. Let $y_i^{(s)}$ be a *standardized* expression levels of gene i in an individual with a condition s (cancer or normal) modeled as: $y_i^{(s)} \approx \sum_{r=1}^{p} w_{ir}^{(s)} x_r^{(s)}$, where $x_1^{(s)}, ...x_p^{(s)}$ denote standardized expression levels of candidate regulator genes in a condi-

tion s. Standardization is a standard practice of normalizing expression levels of each gene to be mean zero and unit stadard deviation before applying penalized regression method [235, 236, 61, 211]. To estimate weight vector $w_i^{(s)}$ *lasso* [235] optimizes the following objective function: $\text{argmin}_{w_{i1}^{(s)},\ldots,w_{ip}^{(s)}} \sum_{j=1}^{n} \left(y_{ij}^{(s)} - \sum_{r=1}^{p} w_{ir}^{(s)} x_{jr}^{(s)} \right)^2 - \lambda \sum_{r=1}^{p} |w_{ir}^{(s)}|$, where the subscript j in the formula iterates over all patients, used as training instances for *lasso*. Here, $y_{ij}^{(s)}$ corresponds to the expression level of the i^{th} gene in the j^{th} patient in the s^{th} state and $x_{ij}^{(s)}$ similarly corresponds to the expression level of the i^{th} regulator in the j^{th} patient in the s^{th} state. The second term, the L_1 penalty function, will zero out many irrelevant regulators for a given gene, because it is known to induce sparsity in solution [235].

After estimating $w_i^{(s)}$ for each s, the DISCERN score measures how well each weight vector learned on one condition explains the data in the other condition, by using a novel model selection criteria defined as:

$$\begin{aligned} \text{DISCERN}_i &= \frac{\text{per sample -log-likelihood based on } w_i^{(s)} \text{ on data in the other condition s'}}{\text{per sample -log-likelihood based on } w_i^{(s)} \text{ on data in the same condition s}} \\ &= \frac{err_i(c,n) + err_i(n,c)}{err_i(c,c) + err_i(n,n)}, \end{aligned} \tag{6.1}$$

where $err_i(s,s') = \frac{1}{n_s} ||y_i^{(s)} - \sum_{r=1}^{p} w_{ir}^{(s')} x_r^{(s)}||_2^2$. Here n_s is the number of samples in the data from condition s. The numerator in Eq (6.1) measures the error of predicting gene i's expression levels in cancer (normal) based on the weights learned in normal (cancer). If gene i has different sets of regulators between cancer and normal, it would have a high DISCERN score. The denominator plays an important role as a normalization factor. To show that, we defined an alternative score, namely the D^0 score that uses only the numerator of the DISCERN score, Eq (6.1):

$$\begin{aligned} D_i^0 &= err_i(c,n) + err_i(n,c) \\ &= \frac{1}{n_c} ||y_i^{(c)} - \sum_{r=1}^{p} w_{ir}^{(n)} x_r^{(c)}||_2^2 + \frac{1}{n_n} ||y_i^{(n)} - \sum_{r=1}^{p} w_{ir}^{(c)} x_r^{(n)}||_2^2 \end{aligned} \tag{6.2}$$

The first step of calculating the DISCERN score and D^0 score is to fit a sparse linear model (such as *lasso* [235]) for each gene's expression level. We used the *scikit-learn* Python

package (version 0.14.1) to calculate these scores with the values of the sparsity tuning parameters λ chosen by using the 5-fold cross-validation tests.

ANOVA score computation

Analysis of Variance (ANOVA) is a standard statistical technique to measure the statistical significance of the difference in mean between two or more groups of numbers. For each gene, the 1-way ANOVA test produces a p-value from the F-test, which measures how significantly its expression level is different between conditions (e.g., cancer and normal). The ANOVA score was computed as negative logarithm of a p-value, obtained from 1-way ANOVA test using `f_oneway` function in `scipy.stats` Python package.

PLSNet score computation

PLSNet score attempts to measure how likely each gene is differently connected with other genes between conditions. It was computed using dna R package version 0.2_1 [87]. The network perturbation score for each gene is computed based on the empirical p-value from 1,000 permutation tests.

LNS score computation

In Guan et al. (2013) [99], the authors defined the local network similarity (LNS) score for gene i that is defined as correlation of the Fisher's z-transformed correlation coefficients between expression of gene i and all other genes between two conditions:

$$LNS_i = \mathrm{corr}(\mathrm{arctanh}(c_{ij}^n), \mathrm{arctanh}(c_{ij}^c)), \tag{6.3}$$

where c_{ij}^s represents the correlation coefficient between expression levels of genes i and j in condition $s = n$ for normal and $s = c$ for cancer.

D-score score computation

For synthetic data analysis, we have also introduced a D-score, computed as following (as used in Wang et al. (2009) [252]):

$$D_i = \|d_{ij}^n - d_{ij}^c\|_1, \tag{6.4}$$

where d_{ij}^s is a normalized correlation (normalized to have zero mean and unit variance across genes) between genes i and j in condition s, also known as Glass' *d score* [89].

Synthetic data generation

We generated 100 pairs of datasets, each representing disease and normal conditions. Each pair of datasets contains 100 variables drawn from the multivariate normal distribution with zero mean and covariance matrices Σ_1 and Σ_2. Each dataset contains n_1 and n_2 samples, respectively, where n_1 is randomly selected from uniform distribution between 100 and 110, and n_2 is from uniform distribution between 16 and 26. This difference in n_1 and n_2 reflects the ratio of the cancer samples and normal samples in the gene expression data (**Table 6.1**).

For each of the 100 pairs of datasets, we divided 100 variables into the following three categories: 1) variables that have different sets of edge weights with other variables across two conditions, 2) variables that have exactly the same sets of edge weights with each other across the conditions, and 3) variables not connected with any other variables in the categories 2) and 3) in both conditions. For example, in **Figure 6.1A**, '1' is in category 1) (i.e., perturbed genes). '2', '4', '6', and '7' are in category 2), and '3' and '5' is in category 3). In each of the 100 pairs of datasets, the number of genes in category #1 (perturbed genes), p, is randomly selected from uniform distribution between 5 and 15. The number of genes in each of the other two categories #2 and #3 is determined as $(100 - p)/2$.

We describe below how we generated the network edge weights (i.e., elements of Σ_1^{-1} and Σ_2^{-1}) among the 100 variables. To ensure that only the genes in #1 have differing edge weights between two conditions, we generated two $p \times p$ matrices, X_1 and X_2, with elements randomly drawn from a uniform distribution between -1 and 1. Then, we generated

symmetric matrices, $X_1^\intercal X_1$ and $X_2^\intercal X_2$, and added positive values to the diagonal elements to these symmetric matrices, if its minimum eigenvalue is negative – a commonly used method to generate positive definite matrices [27]. They become submatrices of Σ_1^{-1} and Σ_2^{-1} for these p variables. Similarly, we generate a common submatrix for the variables in category #2 – variables that have the same edge weights with other variables across conditions. Variables in category #3 have identity matrix as the inverse covariance matrix among the variables in that categories. Finally, we added mean zero Gaussian noise to each element of Σ_1^{-1} and Σ_2^{-1}, where the standard deviation of the Gaussian noise is randomly selected between 0.5 and 5.

This procedure allows having datasets of varying levels of difficulty in terms of high-dimensionality and network perturbation, which provides an opportunity to compare the average performances of the methods in various settings.

Conservative permutation tests

To generate a conservative null distribution, we performed permutation tests by randomly re-assigning cancer/normal labels to each sample, preserving the total numbers of cancer/normal samples. The correlation structure among genes would be preserved, because every gene is assigned the same permuted label in each permutation test. We then computed the DISCERN score for a random subset of 300 genes. We repeated this process to get over one million DISCERN scores to form a stable null distribution, which was used to compute empirical p-values.

Identifying survival-associated genes

For the survival-associated genes enrichment analysis, we first computed the association between survival time and each gene expression level. Genes that had a p-value from the Cox proportional hazards model (computed using *survival* R package) smaller than 0.01 were considered significantly associated with survival. These include 1,280 genes (AML), 1,891 genes (BRC) and 1,273 genes (LUAD) (Supplementary Table 3 from [93]). Statistical

significance of the overlap with top N DISCERN, LNS, D-score and ANOVA -scoring genes was computed by using the Fisher's exact test based on the hypergeometric distribution function from *scipy.stats* Python package [129].

Gene sets previously known to be important in cancer

We presented the results on the comparison with three sets of genes that are known to be important in cancer (Supplementary Figure 2 from [93]). Here, we describe how we obtained these gene sets. First, Malacards genesets were constructed based on the data from malacards.org website accessed in September 2012. Second, we used a set of 488 genes we downloaded from Catalogue of Somatic Mutations in Cancer website (CGC) [82]. For each cancer type, we considered the intersection between this list and the genes that are present in the expression data. Finally, a set of genes likely to contain driver mutations selected by MutSig was defined as those that pass $q < 0.5$ threshold based on 20141017 MutSig2.0 report from Broad Firehose.

Cross-dataset survival time prediction

To evaluate the performance of the DISCERN score on identifying genes to be used in a prognosis prediction model, we trained the survival prediction model using one dataset and tested the model on an independent dataset (**Figure 6.4**). To train the survival prediction model, we used the elastic net regression ($\alpha = 0.5$) using `glmnet` CRAN package (version 1.9-8). Available clinical covariates – age for AML, and age, grade and subtype for BRC – were added as unpenalized covariates. Regularization parameter λ was chosen by using the built-in cross-validation function. Testing was always performed in the independent dataset with held-out samples from the dataset that was not used for training. For comparison, we trained the prediction model using 22 LSC genes [86] with age in AML, and 67 genes from the 70-gene-signature [88] (3 genes from the signature were missing in the dataset we were using) with clinical covariates (age, stage, and subtype) in BRC, as shown in **Figure 6.4B** and **Figure 6.4D**, respectively.

Epigenomics analysis

The Encyclopedia of DNA Elements (ENCODE) is an international collaboration providing transcription factor binding and histone modification data in hundreds of different cell lines [70]. Data for ENCODE analysis were accessed through the UCSC Genome Browser data matrix [132] and processed using the BedTools and pybedtools packages [199, 53]. Two of the ENCODE cell lines – NB4 (an AML subtype [145]) and CD34+ (mobilized CD34 positive hematopoietic progenitor cells) – are closest to AML and normal conditions, and the DNase-seq data from these cell lines are available.

For each cell line, we used the DNase-seq data and the position weight matrices (PWMs) of 57 transcription factors (TFs) available in the JASPAR database [210] to find the locations of the PWM motifs that are on the hypersensitive regions. We identified the locations of these PWM motifs on the hg38 assembly by using FIMO [91] (p-value $\leq 10^{-5}$). We then intersected these motif locations with hypersensitive regions identified by the DNase-seq data for each TF. We repeated this process to identify active binding motifs of the 57 TFs in each of the cell lines, NB4 and CD34+.

For each TF, we identified the genes the TF differentialy binds to between cancer and normal cell lines. We assumed that a certain TF is bound near a gene if the center of the peak is in the active enhancer regions (marked by H3K27Ac) within 15kbs of the transcription start site (TSS) of the gene or the 5kb around the gene's transcription start site. We show that for most of the TFs, differentially bound genes have significantly high DISCERN scores than those not (**Figure 6.5A-C**).

The differential regulator score for each gene was computed by taking the number of differentially bound TFs and dividing it by the total number of TFs bound to the gene in any condition. We show that the differential regulator score is highly correlated with the DISCERN score (**Figure 6.5D**). For DNase-based analysis (**Figure 6.5E**), we defined a gene to be differentially regulated if hypersensitive sites detected by DNase-seq are within 150bp upstream of the gene in one condition and not in another.

Reactome enrichment and DISCERN filtering

A set of 605 Reactome pathways was downloaded through Broad Molecular Signature Database (MSigDB) [229]. We postulate that hypersensitive sites identified by DNase-seq in a particular cell line indicate the regions where important regulatory events occur, such as transcription factor binding. We constructed the list of differentially regulated genes by comparing the hypersensitive sites identified by DNase-seq data between cancer and normal cell lines within 150bp upstream from TSS of each gene. For each pathway, we computed the fold enrichment ($= \frac{\text{number of genes in the intersection of two groups of genes}}{\text{number of genes in the intersection by random chance}}$) that measures the significance of the overlap between genes in the pathway and the identified differentially regulated genes. We compared the fold enrichment with when the genes in the intersection of differentially regulated genes and 1,351 significantly perturbed genes identified by DISCERN were used (Supplementary Figure 6 in [93]). To reduce the noise, we only considered the pathways that had ≥ 5 genes in the overlap before filtering. The p-values were then FDR corrected for multiple hypothesis testing. Although p-values would measure the significance of the overlap between a gene set with a pathway, we used the enrichment fold as a measure of the significance of the overlap because we compared a set of genes with another set much smaller size.

6.4 Summary

In this chapter, we have presented a general computational framework for identifying the *perturbed* genes, i.e., genes whose network connections with other genes are significantly different across conditions, and tested the identified genes with statistical and biological benchmarks on multiple human cancers. Our method outperforms existing alternatives, such as LNS, D-score, and PLSNet, based on synthetic data experiments and through biological validation performed using seven distinct cancer genome-wide gene expression datasets, gathered on five different platforms and spanning three different cancer types – AML, breast cancer and lung cancer. We have demonstrated that DISCERN is better than other methods for

identifying network-perturbation in terms of identifying genes known to be or potentially important in cancer, as well as genes that are subject to differential binding of transcription factor according to the ENCODE DNase-seq data. We have also demonstrated a method to use DISCERN scores to boost signal in the enrichment test of targets of differential regulation constructed using DNase-seq data available through the ENCODE Project.

The work in this chapter was first reported in [93]. The resulting DISCERN score for each gene in AML, BRC and LUAD, the implementation of DISCERN, and the data used in the study are freely available on our website http://discern-leelab.cs.washington.edu/.

Chapter 7

ENABLING REPRODUCIBILITY BY USING CLAIM-AWARE DATA INTEGRATION

7.1 Introduction

In response to increasing awareness of reproducibility issues in many fields of science [124, 83, 167, 123], funding agencies and publishers have made significant investments to create public data repositories (e.g. [68]). In some fields, authors are required to submit data supporting their findings as a condition of publication or funding [24]. But while these policies have been successful in forcing researchers to make data public, there is little evidence that the data are being reused to any significant degree. For example, the number of datasets deposited in the Gene Expression Omnibus (GEO) has been growing superlinearly over time, but the number of datasets used in integrative studies (those referring at least 3 different datasets) has been lagging behind (Figure 7.1). Moreover, the number of datasets referenced per paper has not increased significantly (with mean rising from 1.7 to just 2.3 over the last 10 years), suggesting that the friction of finding, cleaning, and integrating other people's data remains prohibitively expensive. Cultural pressures also limit the reuse of data; those who reuse other people's data have been labeled "data parasites" [160, 96].

Existing services over these public data repositories amount to little more than keyword search [37], which itself performs poorly. A search for datasets for common tissue types in GEO, for example "liver", yields precision of just 40% and recall of 57% (estimated by comparing search results with manually curated tissue annotations provided by Lee et al. [153]). Some projects have attempted to improve dataset discovery in open data repositories by automatically profiling data at scale, but these approaches offer no help in integrating data once relevant data is found, and do nothing to encourage reuse in the first place [153, 94].

Projects aimed at improving reproducibility tend to emphasize tools for scientists to use beginning from early phases of their data analysis, requiring significant changes to researchers' behavior and workflow [190, 191, 77].

We advocate a more direct approach that does not rely on influencing researchers' behavior. Researchers publish papers and upload datasets as usual. Each claim from the paper can be represented in a machine-readable form (automatically extracted in some cases, or manually expressed where necessary), and the experimental conditions implied by the claim are aligned to the schema of a relevant dataset. Once the claim is expressed and the schema has been aligned to it, a statistical test based on the extracted claim can be applied and the results aggregate. We apply this process first to the authors' own uploaded data, addressing the schema misalignment and information extraction issues resulting from inconsistent use of terms between the text of the paper and the attributes in the data itself. We extract a set of candidate claims and possible schema matchings, then prune this set of candidates to those that provide statistical evidence in support of the claim. Then we can attempt to *generalize* the claim by gathering relevant subsets of data that were collected under consistent experimental conditions and re-running the relevant statistical test. An overview of our approach is shown on Figure 7.2.

Example 1 Researcher uploads a new paper reporting a significant correlation between the expression levels of gene X and gene Y in the lung tissue of smokers. The researcher also uploads the dataset D on which the analysis was carried out. For simple claims like this one, we can often extract candidate claims directly from the paper. For more complex claims, authors, repository administrators, reviewers, or reproducibility researchers can express the claim in a domain-specific language for capturing simple statistical tests. In this case, the claim is written as $X \sim Y \mid lung, smoker \in D$. The more general claim implied by this result is that this correlation holds for *all* smokers, so we can drop the dataset reference and just write $X \sim Y \mid lung, smoker$. But the dataset uses an uninformative numeric code to indicate smokers; the terms used in the paper do not necessarily match the attribute names in the

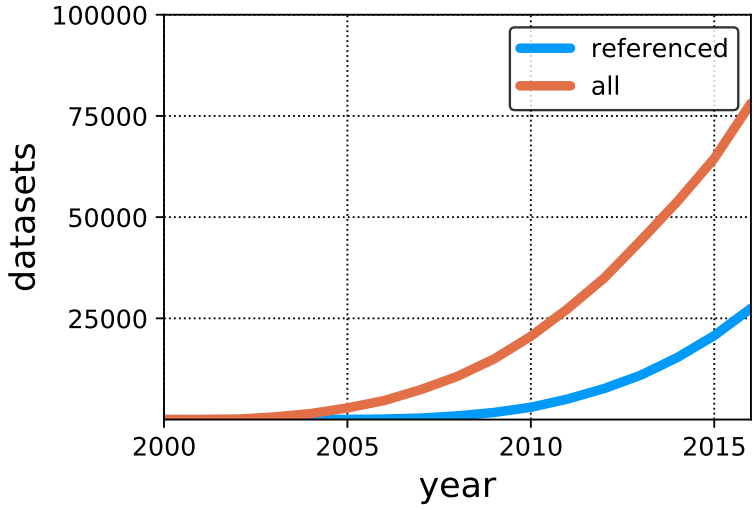


Figure 7.1: Cumulative number of datasets available in GEO over the years, compared with the number of datasets ever cited by reuse papers in Open Access portion of PubMed Central. Obtained by automatic text mining following the method described in Piwowar et al.

dataset. We therefore *adapt* this claim by finding a mapping between the schema implied by the claim and the schema of the dataset. Our approach is to assume that claims are true on the uploaded dataset, and select from among candidate extracted claims and candidate mappings by evaluating the statistical test. Finally, we can validate the generalized claim by finding relevant data items in the broader repository, imputing the conditions (e.g., smoker) when necessary using a Bayesian model, and evaluating a broader experiment. The result is a report showing the confidence interval of the claim against the fused knowledge in the overall repository.

A key challenge is in the semantic heterogeneity between a) the language used in the paper to describe the claim, b) the schema of the dataset associated with the paper, and c) the schema of other relevant datasets. *Our key insight is that we can use the stated claim to help resolve semantic heterogeneity in all three contexts.*

The claim language is naturally extensible; each new type of claim needs only a corresponding statistical test that can be evaluated over the datasets available in a repository.

The basic idea — to use the validity of the statistical test to disambiguate difficult problems in semantic information extraction and data integration — is independent of the nature of the claim. In this work, we study correlation claims (that two variables are correlated under certain conditions) and mean shift claims (that a single variable has statistically different average values under two different conditions) These two types of claims are sufficient to model the results in hundreds of papers in our application of interest.

We use the term "mean shift" to refer to a very general type of claim that claims a statistically significant difference between two variables under similar conditions: treatment outcomes in biomedicine, algorithm performance in computer science, Likert scale survey responses in user studies, economic outcomes in political science, etc. Our approach does not depend crucially on the specific statistical test; for example, a chi square test compares means in frequency for categorical variables, and could be added to the language. We focus on t-tests in this work, as it covers the great majority of claims we see in practice.

The term replicability typically refers to repeating the exact experiment presented in a paper, while reproducibility implies collecting new data to support or challenge a previous conclusion [36]. We are addressing both concepts: we first attempt a simplified (yet automated) replication strategy on the authors' own data, then attempt an automated reproducibility study using other data in the repository. To avoid overloading the term reproducibility, we will refer to this second step as *generalization*.

In this work, we consider this approach in the context of transcriptomics studies that compare gene expression levels across treatment groups, demographic groups, etc. Multiple methods for measuring gene expression data exist (RNA-seq and Microarray chips between the two most prominent); these methods have matured significantly, but were originally prone to severe reproducibility concerns. In some cases, the lab technician operating the equipment was a better predictor of the outcome than the treatment being studied! In part due to this history of poor reproducibility, the Gene Expression Omnibus was established to promote data sharing and improve reuse. But as shown in Figure 7.1, the rate of reuse is still low: The number of datasets is growing superlinearly over time, but instances of reuse

for those datasets is growing much more slowly.

To evaluate our algorithms, we use a set of correlation claims in genomics extracted by Poon et al [196] as well as a corpus of manually encoded mean-shift claims. Cedalion then provides a way to represent these claims in a domain specific language, and use these claim expressions to invoke repository-wide meta-experiments. Cedalion is enabled in part by our previous work on machine learning techniques to provide high-quality tissue annotations for human samples in GEO [94]. We use these annotations to filter out irrelevant datasets from the repository during the claim generalization step. The future combined system build using this components can take scientific papers and a public repository as input and produce a report summarizing the evidence for or against the claims in the paper.

Some other proposed systems attempt to automate the discovery process itself, under the banner of *hypothesis generation* [65, 42, 225]. However, these approaches have a fundamental vulnerability to finding and reporting spurious results — so-called p-hacking. That is, mining for correlations in a large data repository will always produce strong but meaningless signals. In contrast, our system is conservative in the sense that we are only gathering evidence related to published claims that have undergone peer review.

Our contributions are as follows:

- A model of the data integration problem in the context of replicability and reproducibility studies against public data repositories.

- A claim-aware algorithm *ClaimCheck* for replicability that uses the statistical test implied by the claim to prune the search space for schema matching.

- An algorithm *ClaimJump* that uses shared information between datasets to impute missing attributes for use in data integration and generalize the claim for validation against experiementally valid datasets.

- An end-to-end system prototype Cedalion that combines both algorithms together with

our prior work from [196, 94] to generate reports summarizing the evidence for and against the genomics claim.

- An evaluation of these methods against Gene Expression Omnibus repository, including a new assessment of hundreds of claims in published papers. An example of the role Cedalion can play: A 2013 study [134] found a link between the Chemokine ligand 9 gene (CXCL9) and acute rejection of kidney transplants using various datasets available at the time. Cedalion automatically found and integrated relevant data from 2014 *uploaded after the original study was published* to confirm the results, suggesting that Cedalion can be used to continuously and automatically monitor claims as new data becomes available.

The rest of this chapter is organized as follows. In Section 7.2 we consider related work. In Section 7.3 we describe the claim validation problem setting precisely and derive the specific sub-problems we address in this chapter. In Section 7.4 we describe the core algorithms addressing the sub-problems, specialized where noted to our application setting. In Section 7.5 we evaluate these algorithms against a gene expression repository, finding that our methods provide a way to include claims in the solution for large-scale scientific data integration and claim validation problems. In Section 7.6 we discuss our conclusions and position them within a broader context of improving the robustness of science based on statistical arguments.

7.2 Related work

Hypothesis Discovery Duggan et al. provide a cogent discussion of the hypothesis generation problem and propose a solution, but no system has been built, and the approach is fundamentally vulnerable to spurious results and p-hacking [65]. Chirigati et al. describes a novel topology-based method for finding patterns across datasets with shared time and space; this approach is compatible with our framing of the problem, but again lacks guidance from hypotheses developed by domain experts and is therefore vulnerable to spurious results, as

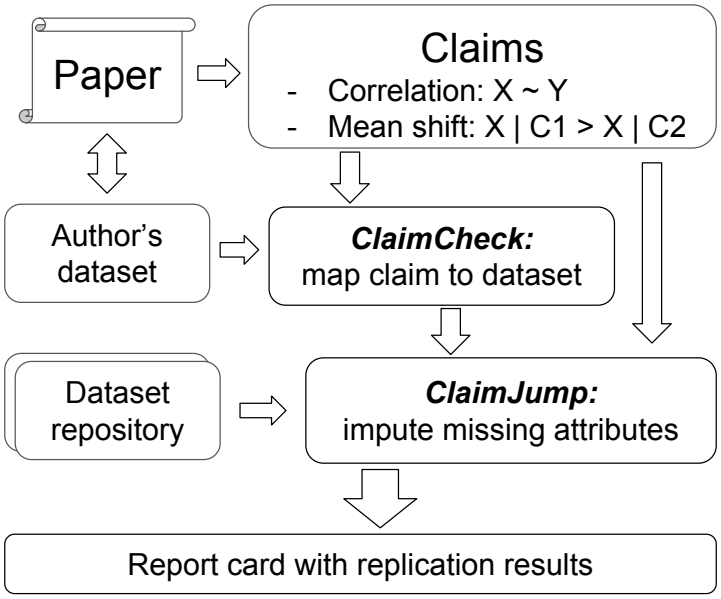


Figure 7.2: Overview of Cedalion system

has been pointed out [42]. Zhao et al. developed new methods for controlling for spurious results by adjusting the family-wise error rates [263]. Spangler et al. [225] mines knowledge from papers and uses this information to identify potential new hypotheses to guide future research, but makes no effort to incorporate public data to verify the claims.

Workflow systems A number of systems aim to provide upstream tools to standardize the analysis process and improve reproducibility as a result, but require that scientists change their behavior to use them [33, 183, 10]. In contrast, we emphasize server-side tools that operate repository-wide to support reproducibility studies.

Code-sharing systems A number of systems allow researchers to remain in their preferred programming environments, but provide services for sharing and reproducibility. Jupyter Notebooks [139] simplify code sharing and data sharing, and have been widely adopted, but make no effort to adapt result to other data, nor link the communicable claims made in

papers to the code and data used to implement the analysis; they are "expert tools." The noWorkflow system augments the Python programming environment to collect provenance information [178, 193]. ReproZip and an earlier system CDE monitor access to files and shared libraries, automatically packaging them to allow re-execution in different environments [100, 43].

Reproducibility Studies Several papers have described studies to measure the extent of the reproducibility problem in various fields, but these studies are primarily manual, or limit their studies to papers whose authors have conformed to strict guidelines for reproducibility [124, 83]. In recognition of these concerns, various fields are enacting top-down standards to enforce reproducibility mandates, including SIGMOD reproducibility efforts [77].

Knowledge Fusion Web-scale knowledge extraction [62, 63, 64] presents opportunities driven by the fact that the same information is repeated multiple times across different web pages. In the scientific domain, we do not have the same degree of redundancy to exploit; studies are expensive and researchers are strongly incentivized to publish unique, novel results. Survey papers and meta-analysis studies are exceptions, but usually only cover a tiny fraction of the overall hypothesis space in the literature and require intense manual effort to explore a single question. Complementary to deep-yet-manual meta-analysis studies, we emphasize broad-yet-automated tools to increase the value of shared data repositories even for those questions where no meta-analysis is likely to be undertaken.

Data Cleaning A large number of methods and tools for data cleaning have been proposed, but these approaches focus on errors in the datasets (typically in the enterprise setting), generally assume the existence of ground truth information that can inform cleaning, and do not consider the statistical claims as a source of information to guide cleaning [54, 142].

Data integration The problem of matching schemas between different relations has been extensively studied (c.f. [201, 20]), but most approaches tend to rely on information only available in conventional enterprise settings, such as a large amount of instance-level data [59], engineered schemas with rich constraints, expressive query languages, or explicit requirements for Extract Transform Load (ETL) pipelines. In our "weakly structured" setting, we do not have access to any of this information, but we do have access to task-specific information in the form of peer-reviewed scientific claims.

Semantic parsing The problem of claim extraction from free text can be cast as a semantic parsing problem where a logical form is being induced directly from text. Approaches to semantic parsing range from unsupervised [195] to supervised by question-answer pairs [15] to a supervision through schema matching in a large labeled database [31]. We consider these methods complementary to our core approach of using statistical claims to guide information extraction and data integration.

Hypothesis validation and discovery systems in genomics Several systems were recently proposed in the functional genomics space, implemented as Shiny R applications, but they all suffer limitations due to the heterogeneous and unstructured nature of the datasets in GEO. ScanGEO [140] is limited to manually curated portion of GEO, comprising of less than 10% of all samples. ShinyGEO [66] allows one to analyze any available dataset but is limited to one dataset at the time analysis and requires users to provide exact mapping of their query to the underlying schema of the data through a GUI interface. GEOracle [58] uses machine learning and text-mining techniques to extract perturbation experiments from GEO datasets and visualize the results of the differential expression. None of the approaches support automatic extension of an analysis to related data or work directly from the scientific claim to formulate the data-analysis hypothesis.

Meta-analysis studies in genomics Tseng et al [240] have analyzed over 300 transcriptomics meta-analysis studies, finding that 66% of them focused on gene differential expression (encoded by our "mean shift" claim type) and further 10% focused on gene co-expression (encoded by our correlation claim type). Microarray meta-analysis is fraught with issues in data quality, heterogeneity, batch effects [203, 30]. We ignore those issues in the current study to focus on our claim-centric methods; we do so by using a consistent corpus of datasets collected from a single genomic expression platform [94]. Extension to a wider range of platforms is an engineering exercise that is left as future work.

7.3 Problem Definition

In this section we make the problem setting precise and define the two sub-problems we address in this paper.

A repository $\mathcal{R}$ is a population of data items: people in the case of a repository of survey results, or tissue samples in the case of a repository of gene expression studies. All data items in the repository have a (potentially very large) set of *unobserved* attributes $X \cup L$, where X is a set of numeric measured attributes over which statistical claims are asserted (e.g., income in the case of surveys or gene expression level for a particular gene) and L is a set of *labeling* attributes; e.g., demographic information, tissue types, or experimental conditions.. That is, we can consider the entire repository as a large relation with an unobserved "true" schema.

A *dataset* $D \subset \mathcal{R}$ is a subset of the repository with a set of *observed* attributes $X_D \cup L_D$. Metadata provided at the dataset level are captured as constant attributes. For example, if a dataset consists entirely of liver tissue samples, a constant attribute `tissue` is assumed, with a value of `liver` for every record in the dataset.

We assume every labeling attribute is binary; non-binary labeling attributes can be pivoted into binary attributes. For example, an attribute `hair color` with domain {`blond`, `red`, `black`} can be pivoted into three binary attributes `hair color:blond`, `hair color:red`, and `hair color:black` all with domain {`true`, `false`}. We will at times refer to a binary

attribute as a *selector*.

A *general claim* is a pair (Q, p), where Q is a boolean query over the unobserved numeric attributes X and p is a predicate over the labeling attributes L. The semantics of a general claim is an assertion that the query Q evaluates to true on a virtual dataset consisting of all data items in $\mathcal{R}$ that satisfy the predicate p.

An *observational* claim is a triple (Q, p', D): a general claim associated with a specific dataset $D \subset \mathcal{R}$. The semantics of an observational claim is an assertion that query Q evaluates to true over the portion of the dataset D that satisfies the predicate p'. Often times, the claim from a paper will correspond to the entire dataset released with the paper and in those cases predicate p' is vacuously true. But we still need to perform schema mapping in this case because Q can contain selectors to talk about two sub-population of the entire dataset.

Since the query Q and the predicate p are expressed over the unobserved attributes X and L respectively, while each dataset D has a schema $X_D \cup L_D$, the query Q cannot be executed directly over any particular D. This complication reflects the fact that the terminology used in the paper to express the claim may or may not be aligned with the attributes of the dataset — the claim in the paper is designed to be interpreted by a human reader and the dataset may involve coded attributes, abbreviations, technical nomenclature, or other differences. A key challenge is to resolve this semantic heterogeneity by finding the mapping from $X \cup L$ to $X_D \cup L_D$ intended by the author that demonstrates the validity of Q on D.

Example of semantic heterogeneity A study of the dataset GSE31210 [184] notes: "In ALK-positive tumors, 30 genes, including ALK and GRIN2A, were commonly overexpressed...". But the corresponding dataset includes the attribute `gene alteration status` with a domain including {`ALK-fusion +`, `EGFR/KRAS/ALK -`}, making it neccessary to distinguish `ALK-fusion +` corresponding to ALK-positive samples. Tumor samples are annotated with the `tissue: primary lung tumor` selector, also presenting a matching problem.

The observational claim asserts that the provided data supports the claim. The *general*

form of the claim asserts that the observational claim is not limited to only the experimental data, but will hold true for all data items that match the experimental conditions. Usually both are desired: without the general claim, the result only applies to the observed sample, which is only an interesting statement if the entire population was observed — a rare situation. But a general claim without an observational claim corresponds to an untested hypothesis, with no supporting experimental data.

We distinguish the two claims even though they are closely related because we will handle them differently: We assume that the observational claim is true, and use this assumption to assist with information extraction and data integration (see Algorithm 2). We do not assume the general claim is true; we find relevant data items from the broader repository and use them to assess the validity of the claim (see Algorithm 3).

In typical cases, the queries associated with claims will be associated with a probability rather than a boolean truth value. That is, claims are true or false only in a probabilistic sense, so it is natural to talk about the probability of the query being true. The types of claims we will consider will be interpreted as statistical tests returning a probability. The corresponding boolean query can be constructed by assigning a threshold, e.g., a p-value threshold of 0.05 associated with significance tests.

7.3.1 Problem Statements

We consider two sub-problems within this framework:

1. **Claim-Aware Query Rewriting** Given an observational claim (Q, p, D), find a query Q' and a predicate p' such that $Q(p(\mathcal{R})) = Q'(p'(D))$, where $p(D)$ is shorthand for $\{x | x \in D, p(x)\}$.

2. **Claim Generalization** Given a general claim (Q, p), derive a new observational claim (Q_i, p_i, D_i) for each compatible dataset $D_1, D_2, ...$ in $\mathcal{R}$, resolving the semantic heterogeneity between D and D_i, then aggregate the results to assess the validity of the claim.

For claim-aware query rewriting, we *assume* the claim is true to assess the rewritings; for claim generalization we will model the uncertainty in the rewritings and *assess* the validity of the claim. This approach exploits the fact that we have access to peer-reviewed claims in addition to author-provided datasets.We will describe this process in more detail in Section 7.4.1.

For claim generalization, we will *impute* missing attributes for relevant datasets in the repository based on the machine learning model trained in the original dataset. A Bayesian model will jointly evaluate the quality of the imputation as well as validity of the claim. This approach exploits the fact that we have a large repository of data with a coherent population. We will describe this process in more detail in Section 7.4.2.

7.3.2 Application Scenarios

In their general form, these sub-problems have very little information available; we instantiate these problems in two problem scenarios where we make additional assumptions. We will develop and evaluate algorithms for the Gene Expression scenario in Section 7.4.

Scenario: A Repository of Surveys A repository of survey results (e.g., ICPSR [232]) contains datasets associated with social science studies. We assume a subset of the attributes in each dataset corresponding to demographic data are explicitly aligned, or can be easily matched using conventional techniques. For example, Chicago Regional Household Travel Inventory (CRHTI, ICPSR 34910) is a survey of travel behaviour of individuals of Chicago area, containing both demographic information and attributes like the travel mode used. Similarly, New York, New Jersey, Connecticut Regional Travel - Household Interview Survey (RT-HIS, ICPSR 35294) contains similar information for a different region, with potentially different encoding for travel modes. The semantic coherence between these two studies suggests an opportunity for Cedalion to validate claims made in one city against data collected in another.

Scenario: A Repository of Gene Expression Data For each dataset in a repository of gene expression data, the numeric measure attributes correspond to gene expression levels for a large set of genes. It is again possible to align the gene attributes - while different naming conventions exist and probe mapping can depend on the choice of the particular probes, a reasonable baseline is possible by processing all datasets through a small number of consistent pipelines. Remember, that our goal is not to exactly replicate all the intricate details of the original research claim, but rather assist the user in quickly getting a rough estimate of its validity and generality in light of the entire repository. We therefore assume that the alignment between the numeric attributes have no uncertainty. The heterogeneity between datasets stems from the labeling attributes, which are study-specific and defined by the authors.

As a first step, we restrict ourselves to two largest types of claims made by the *primary* genomics literature:[1] correlation claims between two genes in the same set of patients, and mean shift claims comparing the distribution of gene expression levels of two subsets of patients under different experimental conditions.

The query rewriting problem therefore reduces to finding a mapping between the schema implied by the predicate in a claim and the schema of a given dataset. A simple string-matching approach on attribute names is the best we can assume using conventional data integration approaches; we do not have instance-level data available for the unobserved schema over which the predicate is expressed.

The claim generalization problem is simplified when there is a statistical relationship between the measured attributes and the labeling attributes. In the biological case, we can use the genetic information to impute certain kinds of labels, even when these labels are not provided by the author. For example, the gender of the patient can be inferred from the gene expression data, as can more environmental effects, such as whether the patient smokes. In the survey scenario, the analogous situation is when demographic information

[1]Secondary, or computational, analysis will usually employ significantly more sophisticated methods that are out of scope for this work.

(e.g., age, job title) can be used to impute survey responses (e.g., income). We will use imputation rather than schema matching for claim generalization. There may or may not be sufficient information in the repository to reliably impute a given attribute, so our approach (Algorithm 3) carefully tracks the uncertainty about the imputation to detect those cases where imputation is impossible.

7.3.3 Claim Expression Language

To describe and reason about claims, we introduce a domain-specific language called *ClaimQL*. The language has two components: a claim language for expressing claims themselves and a transformation language for expressing computations involving those claims.

The grammar for claims is as follows:

$$
\begin{aligned}
\langle \text{claim} \rangle &\models \langle \text{test} \rangle \mid \langle \text{test} \rangle \text{ on } \langle \text{ds} \rangle \mid \langle \text{claim} \rangle, \langle \text{claim} \rangle \\
\langle \text{test} \rangle &\models \langle \text{attr} \rangle \sim \langle \text{attr} \rangle | \langle \text{pred} \rangle \mid \langle \text{attr} \rangle | \langle \text{pred} \rangle > \langle \text{attr} \rangle | \langle \text{pred} \rangle \\
\langle \text{pred} \rangle &\models \langle \text{attr} \rangle = \langle \text{val} \rangle \mid \langle \text{pred} \rangle, \langle \text{pred} \rangle
\end{aligned}
$$

A *predicate* is an attribute-value pair or conjunction of attribute-value pairs. In our initial system, a statistical *test* is either a correlation test (written $X \sim Y$) or a mean shift test (written $X > Y$). Each test can be interpreted as a boolean condition, but is also associated with a numeric value. For a mean shift test, the numeric value is the difference in means. For a correlation test, the numeric value is the correlation. A test also includes the predicates on which it applies. For a correlation test, there is only a single predicate, since the correlation must be between attributes in the same dataset: they both must have the same number of elements, and batch effects will dominate if we test the correlation between two separate datasets. For a mean shift test, there can be predicates associated with each of the two attributes being compared. For example, smokers vs. non-smokers, or those receiving a treatment vs. control.

Scenario: Social Science A study might want to explore correlation between income and prevalence of smoking, using the data D from a survey in Illinois. The claim will be expressed as $smoking \sim income \text{ on } D$

Scenario: Genomics Similarly, a study might want to explore the expression of particular gene X in lung tissue of smokers vs non-smokers, writing $X \mid lung, smoker > X \mid lung, non\text{-}smoker$ to denote their claim.

Scenario: Computer Science A VLDB paper might want to introduce a new query optimization technique and measure its impact on the runtime of TPC-H queries. While we are used to those benchmarks being deterministic, this is still a clear mean shift query: $time \mid optimization{=}1 > time \mid optimization{=}0$. Performing multiple runs of the benchmark and averaging them, or running the same benchmark against a range of different hardware configurations are all ways of collecting a dataset against which this claim is expressed. Lack of established data sharing practices in the field preclude this sort of analysis, but the claim is nonetheless present in the same form.

To connect the claim expression language with the repository structure and selectors, we use the following function.

The function AllSelectors returns the schema implied by the claim, defined as

$$\begin{aligned}
&\textsc{AllSelectors}((X \sim Y|S_0, S_1, ...)) = \{X, Y, S_0, S_1, ...\} \\
&\textsc{AllSelectors}((X|R_0, ...) > (Y|S_0, ...)) = \{X, R_0, ...Y, S_0, ...\} \\
&\textsc{AllSelectors}(C_1, C_2) = \textsc{AllSelectors}(C_1) \cup \\
&\qquad\qquad\qquad\qquad \textsc{AllSelectors}(C_2)
\end{aligned}$$

We will also call AllSelectors on a dataset, in which case it simply returns the set of all possible selectors of the (binarized) dataset.

7.4 Claim-Aware Algorithms

In this section, we present an algorithm *ClaimCheck* for claim-aware query rewriting, and an algorithm *ClaimJump* for claim generalization.

7.4.1 Claim-Aware Query Rewriting

In this section, we describe an approach to query rewriting problem in the gene expression setting, formulate it as an optimization problem, and describe an algorithm to solve it.

A claim (Q, p) involves a set of attributes $X_Q L_p$. To evaluate the claim on the dataset D provided by the author, we must find a mapping between the attributes $X_Q L_p$ and the schema of D.

This problem is related to schema matching and mapping in a conventional data integration context: The unobserved schema is akin to a mediated schema in a global-as-view framework [60]. Recall we assume that all attributes are binary, such that both `gender:male` and `gender:female` are separate binary attributes. Under this assumption, we can model the problem as a) finding a schema mapping between the mediated schema and the observed schema, then b) rewriting the query using this mapping. However, there are some complications in our setting: First, there is no instance-level data for the unobserved schema, preventing us from using state-of-the-art instance-level schema matchers (e.g., BigGorilla [120]). Second, we do not have access to any auxiliary information about the attributes, such as types, value constraints, key constraints, nesting or grouping structures among attributes [162]; our labels are simply strings associated with data items. Third, the attribute names can be domain-specific jargon, or worse, arbitrary strings with no obvious semantic content. For example, linguistic matching will not help us with a claim made for the dataset GSE28654 comparing group called “UMZAP-70+” and group called “MTZAP-70-” (the correct mapping is ”class2” and ”class1” respectively).

As a result of these complications, the only schema-level information we can use is the string itself. We will therefore consider a fuzzy string similarity algorithm as a baseline

approach.

More generally, our approach is to search through the space of possible mappings, optimizing some objective function that expresses the quality of the mapping. The baseline approach uses an objective function that provides a similarity score based on fuzzy string matching. As mentioned, we cannot use instance-level information to help us choose a mapping since we do not have instance data available for the unobserved schema [59].

Our approach is instead to consider all possible mappings, evaluate Q under each mapping, then choose the mapping that returns the highest probability of the claim being true. In this case, we are assuming that Q is a statistical test that returns a score rather than a boolean value. We cast this task as an optimization problem that combines fuzzy string matching as well as statistical claim validity, then solve it.

We formulate the query rewriting task as the following optimization problem.

$$\tilde{M} = \arg\max_{M \in \mathbb{M}} f(Q, p, D, M) \tag{7.1}$$

where $\tilde{M}$ is the result mapping we seek, Q, p are the query and predicate representing a claim such that Q returns a probability that the claim is true, D is a dataset, $\mathbb{M}$ is a set of all possible mappings between attributes in the unobserved schema implied by claim $C = (Q, p)$ and the observed schema of the dataset D, and f is an objective function defined separately for different approaches.

We consider three objective functions, corresponding to three different methods:

Baseline string matching As a baseline approach, we use fuzzy string matching on column names. Here, we define f to be the sum of the inverse Levenshtein distances (scaled between 0 and 1) between each mapped pair of attribute names:

$$F_{\text{string}}(Q, p, D, M) = \sum_{s, s' \in M} \text{ILD}(s, s')$$

Simple claim-based matching In case of purely claim-based matching, f will be defined by evaluating the query on D, subject to p, under the mapping M. The result of this query

indicates the probability that the claim is true:

$$F_{\text{claim}}(Q, p, D, M) = Q(p(M(D)))$$

where $M(D)$ indicates the dataset D with attributes renamed according to the mapping M.

And finally, our proposed algorithm *ClaimCheck* combines both methods in a weighted sum objective function:

$$F_{ClaimCheck}(Q, p, D, M) = Q(p(M(D))) + \alpha \sum_{s,s' \in M} \text{ILD}(s, s')$$

where α is a trade-off parameter, set to 1 for the main results presented in the paper. We further evaluate the sensitivity of *ClaimCheck* to this parameter with additional experiments.

Implementation Algorithm 2 details the steps taken by *ClaimCheck* to recover the best mapping. We start by mapping those attributes from the unobserved schema that have exact matches in the observed schema to reduce the search space. We then attempt to find a maximum value of the objective function (corresponding to an optimal mapping). This is a discrete optimization problem, and thus computational cost is of concern. We propose three approaches to solving the problem (corresponding to three different implementations of OPTIMIZE function):

- Naïve approach. We can exhaustively enumerate all possible mappings and find the optimal one. While clearly costly in general for datasets with large number of selectors, we do find that for most of the datasets we have examined this approach is viable.

- We employ a *stochastic optimization* package called HyperOpt that implements a sequential model-based optimization (SMBO) approach based on a Tree-structured Parzen Estimator [17, 18, 19]. HyperOpt maintains a statistical model that aims to approximate the objective function based on the history of evaluations.

- Finally, we exploit the fact that most claims are conjunctions, such that we can sort the list of claims by increasing number of selectors and iteratively solve each claim

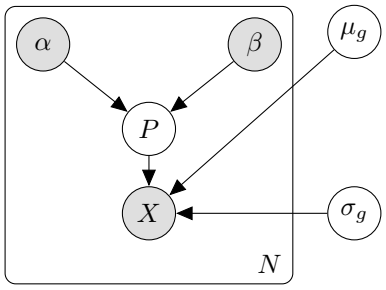


Figure 7.3: Bayesian graphical model for estimating mean with uncertain group labels. α and β are estimated from bootstrapped classifiers for predicting the group,

mapping problem separately, repairing the mapping learned in the previous step for the next one. Surprisingly, we find that this *greedy approach* does not significantly compromise performance, while providing substantial run time improvements.

7.4.2 Claim Generalization

The generalization problem requires finding a mapping between the claim schema and a schema of an unrelated dataset.

Given a claim (Q, p, D) and a another dataset in the repository D', we can consider three methods for solving the claim generalization problem:

1. **Conventional schema matching** Use instance-level and schema-level information to infer a match between D and D'. In some cases no match may exist.

2. **Imputation** Identify shared attributes between D and D' (either provided explicitly or derived from conventional schema matching techniques), then train a classifier to impute the missing attributes in D'.

3. **Combined** Use imputation to produce missing attributes, then look for correlations between the imputed attributes and the existing attributes as a source of information for conventional schema matchers.

The effectiveness of these methods in any given application will depend on the strength of the signal from instance-level data and shared attributes. In some settings, there will be high semantic coherence between all datasets in the repository, and conventional schema matching techniques will work well. In other settings, the labeling attributes will be semantically incoherent, but there may be a large number of shared attributes on which to train a classifier. In the worst case, the schemas will be entirely discordant, and there is no effective way to find an appropriate mapping.

In the gene expression scenario, the core shared attributes of gene expression data are standard, but the metadata attributes are highly study-specific. In this scenario, the imputation approach can be effective and sufficient, but schema matchers alone are likely to fail.

In the survey scenario, the core demographic data that can be assumed to be present is somewhat limited: (gender, race, age, etc.) However, a number of other attributes are likely to be frequent, if not universally shared: income, profession, national origin, etc. So there is some possibility that schema matchers can work, but a combined approach can help identify cases where attributes are coded differently in different studies. For example, political affiliation could be coded as republican = 1 and democrat = 2 in D as opposed to republican = R and democrat = D in D'. In these cases, conventional schema matchers can struggle because the domains are different, but we can first impute a column in the correct domain as a pre-processing step.

Problem Summary Given a rewritten claim (Q_s, p_s, D_s) found using the *ClaimCheck* algorithm, and given a target dataset D_t drawn from the repository, find a mapping M between D_s and D_t such that $Q_s(M(D_t)) = Q_s(D_s)$ *if the claim is true.* We cannot know if the claim is true in general; we will evaluate this method with manually validated claims as ground truth.

Solution Our solution is to train a classifier using the claim-specific dataset D_s to predict the values of each of the selectors from AllSelectors((Q_s, p_s)), then use this classifier to impute this labels on D_t so we can evaluate the query. The correctness of the algorithm relies on there being a relationship between the shared training attributes (e.g., demography data or gene expression data). We construct a Bayesian model to capture the uncertainty in the classifier and use a bootstrap approach to estimate the parameters of the model.

Specifically, given the mean classifier prediction is μ_i with variance σ_i^2 (estimated through a bootsrap process for data item i), we use the following formulas to estimate the α and β parameters of the Beta distribution.

$$\alpha = \left(\frac{1 - \mu_i}{\sigma_i^2} - \frac{1}{\mu_i} \right) \mu_i^2$$

$$\beta = \alpha \left(\frac{1}{\mu_i} - 1 \right)$$

Next, we assume that true probability of the particular data item belonging to a group is $p_i \sim Beta(\alpha_i, \beta_i)$ with a full Bayesian model given as follows:

$$x_i \sim \texttt{Normal}(\mu_g, \sigma_g) \mid p_i$$

$$p_i \sim \texttt{Beta}(\alpha_i, \beta_i)$$

$$\mu_g \sim \texttt{Normal}(\mu_0, 1)$$

$$\sigma_g \sim \texttt{half} - \texttt{Cauchy}(0, 1)$$

where α_i, β_i are parameters derived based on the uncertainty of the prediction (they can be thought of the s), μ_0 is the background mean expression value (calculated over the entire dataset D), and μ_g, σ_g are the estimated mean and standard deviation of the posterior distribution of the mean of the group value with standard priors. This model gives us the probability distribution for mean μ_g of interest, based on probabilistic group assignments encoded by α_i and β_i. The Bayesian model is illustrated in Figure 7.3.

We use this Bayesian model as the basis of Algorithm 3 which we refer to as *ClaimJump*. The *ClaimJump* algorithm estimates the validity of a mean shift claim on a new dataset. It

calls the EstimateImputedMean procedure twice to estimate the group mean for each of the groups defined by the two claim predicates using the Bayesian model. For example, for a claim $G|R > G|S$, we call EstimateImputedMean for the predicate R and the predicate S. In lines 9-10 we find common attributes already aligned between the two datasets that we will use for the imputation. In line 14 we build a classifier based on those common attributes using a bootstrap process (BuildClassifier). This means that before the classifier construction the input data is perturbed — sampled with replacement to estimate variability of the resulting classifier. We then applied the classifier to the common attributes of the target dataset, producing the group probabilities used in the Bayesian model. Note that due to the Markov Chain Monte Carlo approach used for estimating the mean in the probabilistic model, *ClaimJump* outputs an entire distribution of possible mean shifts, and not just a point estimate. As will be shown in Section 7.5, *ClaimJump* can automatically detect cases when group classifiers are uncertain and output wide high-density intervals (HDI).

Implementation We use Stan [34] to estimate posterior distribution of the model shown on Figure 7.3. Parameters α_i and β_i are estimated using 50 boostrapped logistic regression classifiers with Lasso penalty, trained using GLMNet [79]. The regularization parameter is automatically chosen by GLMNet using 3-fold cross-validation. In case when cross-validation chooses an empty model, we assume that the covariate imputation is impossible and discard the (dataset, claim) pair. This step is important to avoid imputing covariates that are independent of the gene expression data, hence avoiding making false generalizations.

Given group membership probabilities, we use Stan [34] to estimate posterior distributions of the group means. Comparing those distributions leads to accepting or rejecting the claim, based on the 95% HDI interval of the distribution (HDI interval is a Bayesian analogue for a confidence interval).

7.5 Experiments

Following our tissue annotation work (EZLearn [94]), we will focus on a corpus of 116,685 samples arranged into 3567 datasets from Gene Expression Omnibus from Affymetrix U133 Plus 2.0 platform. Raw data files for each of those samples were downloaded from GEO and consistently processed using Single-Channel Array Normalization (SCAN) [191], a method that aims to reduce the batch effects across samples. Apart from ability to query the repository for related datasets (provided by EZLearn annotations), none of the methods make any particular assumptions about it and are generally applicable to other genomics platforms and application domains.

7.5.1 Evaluating Claim-Aware Rewriting

In this experiment, we measure whether the statistical information used by the *ClaimCheck* algorithm improves precision and recall over baseline methods. Table 7.1 reports the results.

We constructed a test set of 52 datasets by finding those which a) contained the term "overexpressed" in the summary and b) for which at least one claim could be extracted from the summary or the associated paper. This process produced a set of 223 manually extracted mean shift claims. We then manually determined the correct mapping to the dataset through a careful review of the text to serve as ground truth during evaluation. Importantly, we found that some of these claims required 1:N mappings, which our current *ClaimCheck* algorithm does not support. Although we have also designed a generalization of *ClaimCheck* to consider disjunctive mappings, we have not employed it in the presented experiments and thus *ClaimCheck* could not have gotten accuracy of 100% by design. We expect that additional claims can be obtained by a general NLP-based extraction approach or even by searching the repository for a different keyword. Our goal here was just to obtain a representative sample of mean shift claims for the evaluation purposes.

First, we evaluate *ClaimCheck* against two baselines: a conventional string similarity approach doing fuzzy selector matching implemented using FuzzyWuzzy Python package and

Method	**Precision**	**Recall**	**F1**	**Accuracy**
Fuzzy matching	0.75	0.63	0.69	0.72
Claim-only	0.85	0.73	0.79	0.82
ClaimCheck	**0.86**	**0.78**	**0.82**	**0.90**

Table 7.1: Comparison of performance of different schema matching methods. *ClaimCheck* outperforms both claim-only and conventional baselines.

a claim-only approach that ignores the string similarity purely focusing on maximizing claim-matching part. This two approaches are equivalent to either setting α trade-off parameter to infinity (for fuzzy matching) or zero (for claim-only matching). We observe (Table 7.1) that *ClaimCheck* outperforms both of these approaches on our test set, achieving F1 score of 0.82 and accuracy of 90%.

Second, we evaluate sensitivity of *ClaimCheck* to the trade-off parameter α, by evaluating performance of the method for the range of alpha values spanning 0.1 to 10 on logarithmic scale (Figure 7.4). We observe that while having impact performance, *ClaimCheck* still outperforms both baselines in this wide range of α values. Additionally, it is clear that future fine-tuning of the model (e.g., by setting α to 2.2) can further improve performance of the method.

Finally, we perform run-time performance analysis of different variants of *ClaimCheck* as described in Section 7.4.1 (Table 7.2). We observe that stochastic optimization implementation based on HyperOpt can lead to an order of magnitude performance improvement with negligible loss of accuracy, while the claim-based greedy solution can surpass that leading to two orders of magnitude performance improvements at the same accuracy level as the full solution.

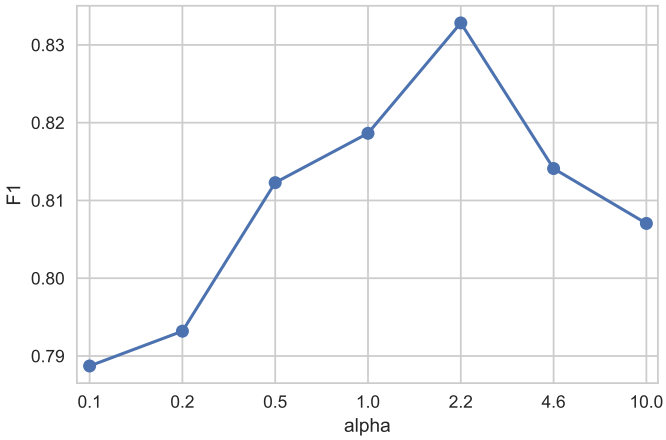


Figure 7.4: F1 score of *ClaimCheck* with varying values of tradeoff parameter α on a log scale. $\alpha = 1$ is used for experiments throughout the paper.

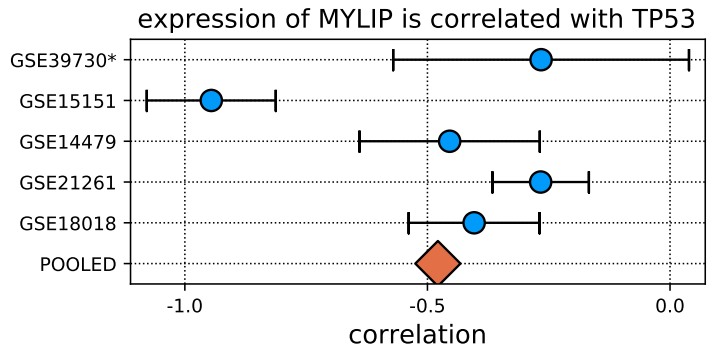
7.5.2 Evaluating ClaimJump

We first want to determine if *ClaimJump* is effective at discovering independent sources of support for a claim within the repository. We designed an experiment as follows: We take a large dataset from our corpus of manually annotated datasets (GSE28654, 112 samples) and split it into two halves randomly. Each time, we assume the claim was made against the first half and the second half is another dataset we found in the repository. We attempt to apply *ClaimJump* to generalize the claim into this second half of the original dataset. Intuitively, we should expect *ClaimJump* generalization to be successful, as the claim is evaluated on the portion of the same dataset it was made against. We repeat the process 50 times.

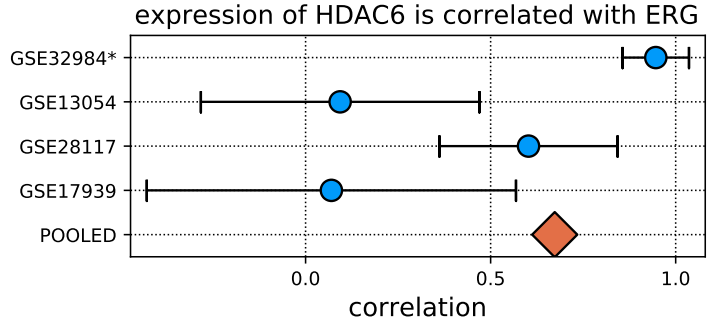
Empirically, out of 50 random splits, we observe the following distribution of outcomes:

- 95% HDI (highest density interval) is entirely above zero - successful replication. This happens for 43 out of 50 cases (86% of the time)

- 95% HDI is entirely below zero — we wrongly and confidently reject a true hypothesis. This happens for 0 out of 50 cases.



(a) Original study achieves the correlation of -0.25 between genes of interest (MYLIP and TP53). After pooling data across 4 more tissue-matched studies we see that the combined correlation coefficient is even stronger, close to -0.5.



(b) In this set of studies, original results become weaker after pooling data across additional related studies. Significant discrepancies between original results and pooled results should be a reason for a more careful look at the claims involved.

Figure 7.5: Report cards for a pair of correlation claims. Each card compares the original study effect (denoted by $*$) with computational replication studies based on tissue-matched datasets found in GEO. Pooled effect based on combining correlations from all datasets with appropriate weighting is denoted as "POOLED".

Implementation	**Time** seconds	**F1**	**Accuracy**
Naïve	1625	0.82	0.90
HyperOpt (maxEval = 500)	108	0.81	0.87
HyperOpt (maxEval = 250)	51	0.81	0.84
Greedy in claim size	**14**	0.80	0.90

Table 7.2: Performance comparison of different implementations of *ClaimCheck*. A naive search through all possible mapping is the most accurate, but computationally expensive. HyperOpt-based approach allows us to trade off accuracy and performance by tuning maximum number of allowed evaluations and Greedy approach achives the best overall balance of performance and accuracy.

- Cases when 95% HDI intersects zero suggest that our signal is not strong enough to make a conclusion, this happens 14% of the time. This is higher more than 5% of the time we would expect for 95% HDI in theory, suggesting future improvements to the model might help to improve the performance even further.

Limitations This experiment only measures positive results; the inverse evaluation is more difficult to measure, since there are myriad ways a dataset can fail to support a claim, and it would be easy for *ClaimJump* to detect them in most cases (e.g., incompatible tissue types). An evaluation of these negative results remains future work.

Synthetic experiments

Next, we perform a sensitivity analysis on synthetic data to answer two questions: First, does *ClaimJump* accurately estimate true effect sizes? Second, can *ClaimJump* still accurately

estimate the effect size of the claim when the imputation process is highly uncertain?

To answer the first question, we evaluated the ability of the model to uncover the true effect size at a predetermined confidence level (set at 70%) based on the size of the effect. The model was successful at recovering correct effect with tight bounds from this synthetic dataset of size 100 (Figure 7.6).

To answer the second question, we estimated the mean shift effect in a pair of synthetically generated datasets: one that contained a true effect size of 0.4 and another that did not contain an effect. We observed that the model correctly identified presence of the effect even when the classifier confidence was only 20%, demonstrating robustness. The model also correctly determined lack of the effect regardless of the confidence (Figure 7.8).

Limitations We do depend on the imputation classifier being well-calibrated. As demonstrated, *ClaimJump* is robust to classifier being wrong with low-confidence, but we will not be able to tolerate high-confidence incorrect predictions. We use standard cross-validation process during training to reduce the risk of this happening.

7.5.3 Applying Cedalion in Practice

In these experiments, we show how Cedalion can be used in practice as part of a process to curate a scientific repository.

First, using the correlation claims extracted by the Literome project[195], we run the end-to-end Cedalion system, and measure the support for each claim. We find that we can detect independent support corroborating 72% of the evaluated claims (Section 7.5.3).

Next, we run *ClaimJump* on our set of manually extracted mean shift claims to assess their validity using publicly available data. Out of 223 claims, we find significant supporting evidence for 24 of them and significant counter evidence against 11 of them (Figure 7.9). For the remaining claims, *ClaimJump* was unable to find enough relevant data items to assess validity. (Section 7.5.3)

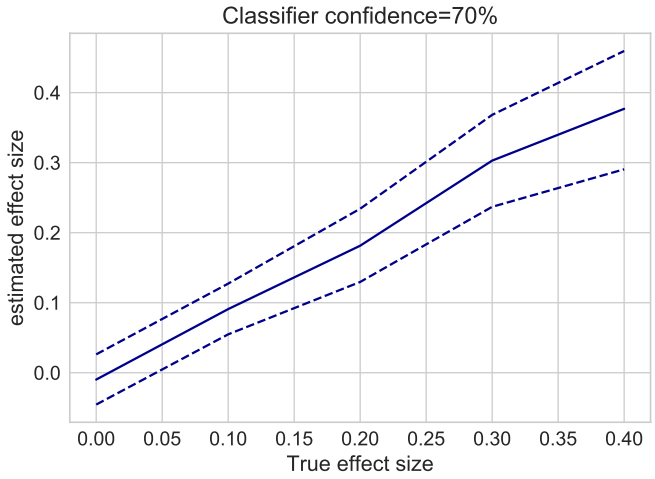


Figure 7.6: Estimates of the effect size from a Bayesian imputation model depending on the true effect size. Median of the distribution is shown as a solid line with bounds of the 95% HDI represented by dotted lines.

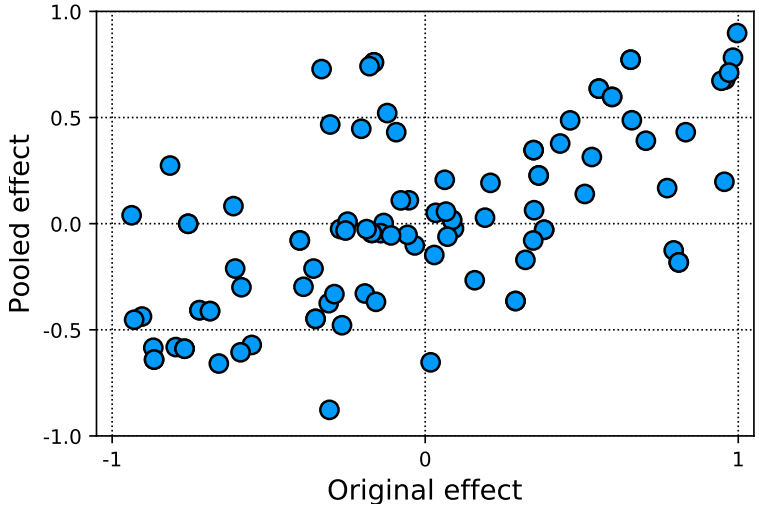


Figure 7.7: Overall scatter plot of the original vs pooled correlations of correlation claim experiments where at least one other related dataset were found. Points in the first and third quadrant are considered to be successful verifications - original and pooled results have the same sign there.

Correlation claims

We obtained correlation-type claims from the Literome automatic NLP extraction system of Poon et al [196]. Literome extractions are based just on the abstract of the paper and so do not require a dataset to be present for the statement to be extracted. For computational verification, we were interested in a set of claims for which the data is also available as part of our corpus of processed Affy U133 Plus 2.0 datasets. Joining the Literome claims with these datasets we get 166 (dataset, claim) pairs. The GEOmetadb database [266] was used to match article identifiers reported by Literome to dataset identifiers available in GEO. Only a small number of claims have datasets associated with them because: 1) not all of the claims are derived from gene expression data deposited in GEO, 2) not every dataset in GEO is associated with a correlation claim.

Out of 166 claims, we were able to match 103 claims to their original dataset and find at least one additional dataset for computing the pooled correlation. Of these 103 matches, the estimated effect size reported by Cedalion had the same sign as the original effect size for 72% (74) of the claims, suggesting agreement with the original study (Figure 7.7).

In Figure 7.5a, we inspect Cedalion results for a typical paper. The paper [208] (along with its dataset, GSE39370) states that expression of gene MYLIP is correlated with expression of gene TP53 in leukemia. As a part of our validation process, we match this dataset with four other datasets and compute the correlation between MYLIP and TP53 in each of them. We then compute the pooled correlation by weighting each correlation coefficient by the inverse of the size of the confidence interval [157].

In Figure 7.5b, we show Cedalion results for another paper where pooling data across different studies make us *less* certain about the original claim. This effect may either be a statistical artifact of the original study, a manifestation of the underlying noisy biological system, a mismatch between our method of computing correlation (*ClaimJump* and *ClaimQL* could be extended to have claims parameterized by a particular implementation of each statistical test, but this is left for future work) and the original statistical analysis in the

paper, or some other factor.

In the future, we envision that this generalization process will be part of a human-in-the-loop system, where a biologist interested in a particular claim will be able to review steps taken be Cedalion to replicate it and decide if they want to trust them. Having a high-recall system that will find all *possibly* related datasets will save time for the user, as filtering out irrelevant datasets from a list of a few is significantly easier than manually searching for relevant ones in the repository, downloading them, and processing them manually.

Mean shift claims

Next, we apply *ClaimJump* to our corpus of 223 manually curated mean shift claims to generate a "report card" for each claim. There is no ground truth for a study like this, but these examples demonstrate the potential value of the approach in practice.

First, we use *ClaimJump* to generalize each claim from the corpus to the entire set of other related datasets at once. Figure 7.9 shows results of such generalization. Each dot on the graph represents a (claim, dataset) pair, with the original effect plotted on the x-axis and the effect estimated by the Bayesian procedure from the other data in the repository on the y-axis. We note two outlier datasets — GSE50058 and GSE70102 (see Figure 7.9) for future evaluation.

Next, we evaluate a claim from dataset GSE50058 that states that the CXCL9 gene is overexpressed in rejected kidney transplants compared to the stable ones [134]. Reading the paper associated with GSE50058, we notice that the authors have used additional datasets to formulate their hypothesis before collecting their own dataset for validation. It is interesting to observe that Cedalion has found several of those datasets (dataset1 and dataset7 on Figure 7.10a) and confirmed that they indeed do support the claim from the paper. This suggests that Cedalion is capable of successfully finding related datasets and confirming their relevance. Furthermore, while dataset11 strongly supports the hypothesis and was actually collected with the same goal in mind as GSE50058 (studying differences between acutely rejected and stable renal transplants), it was not used in the original study. The explanation

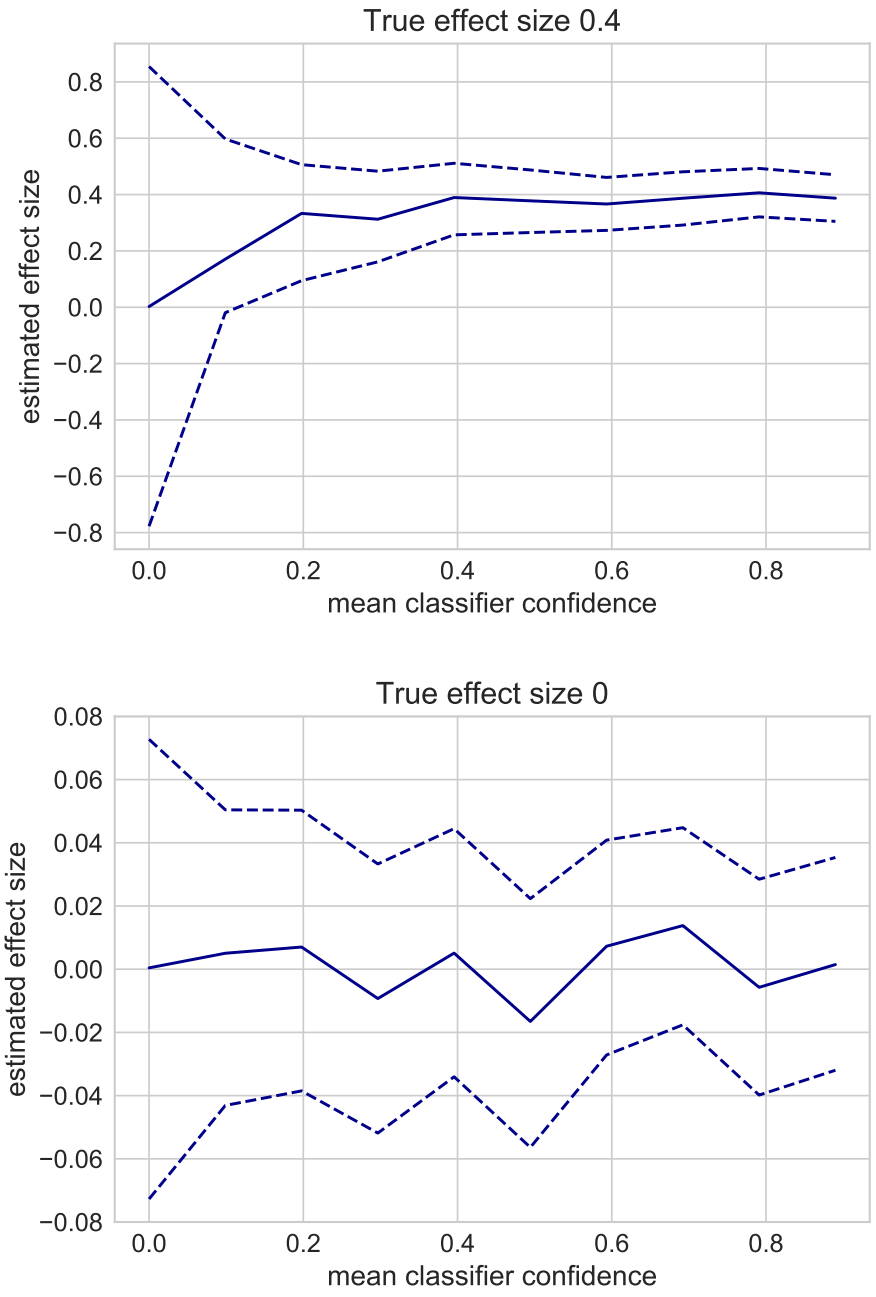


Figure 7.8: Estimates of the effect size from a Bayesian imputation model depending on the imputation classifier confidence. Median of the distribution is shown as a solid line with bounds of the 95% HDI represented by dotted lines.

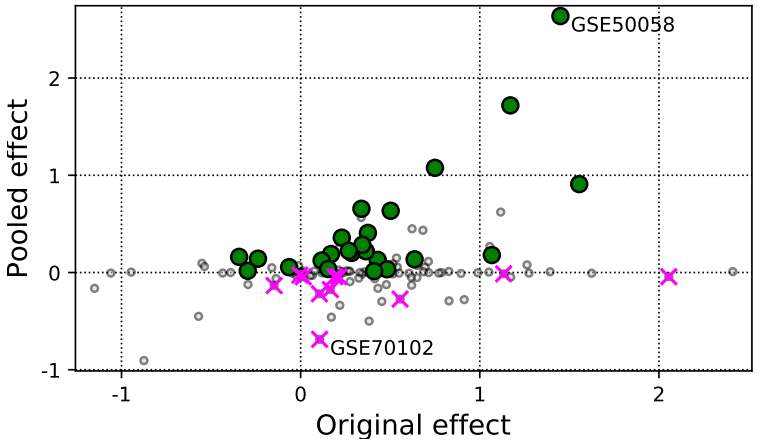
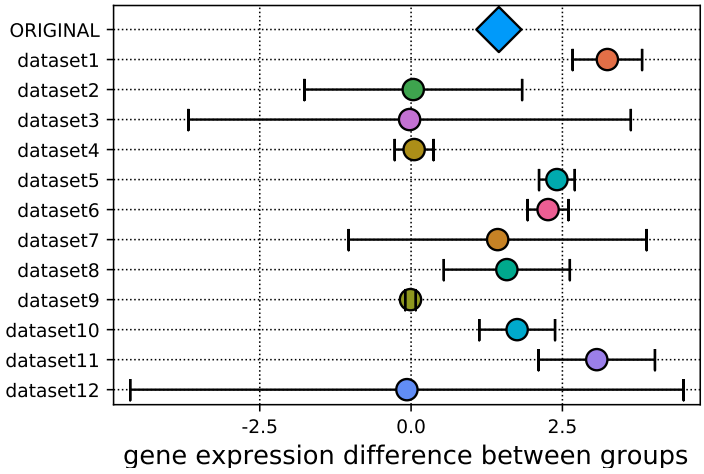


Figure 7.9: Comparison of the original and pooled mean shift effect sizes. Gray points represent claims with insufficient certainty to be considered by *ClaimJump*. Each magenta X represents a claim with 95% HDI < 0, suggesting *ClaimJump* does not support the claim. Each green circle represents a claim with 95% HDI > 0, suggesting *ClaimJump* corroborates the claim. Two outliers are annotated for future analysis.

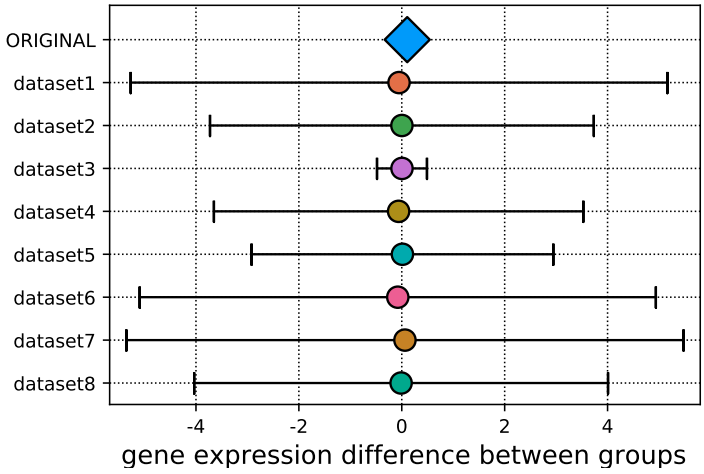
is simple: it was made public in 2014, while the original study was published in 2013.

On the other hand, Figure 7.10b shows results of the replication of the claim that achieves the largest negative effect size when the information is pooled in the entire repository [22]. Looking at the per-dataset estimates of the effect, though, we observe that they are all around zero with a small negative bias, suggesting that the global effect observed might be the result of batch effects.

Another possible explanation is that the original study in this case (GSE70102) was looking at a specific chromosomal condition (Down syndrome) with a small sample size (8 samples in total), explaining wide HDI intervals on Figure 7.10b - imputing the covariate based on classifier build from such a small sample size should be highly uncertain.



(a) Dataset GSE50058



(b) Dataset GSE70102

Figure 7.10: An example report cards for two mean shift claims, showing an effect from the original dataset, as well as estimated effects from additional related datasets.

7.6 Summary

We have introduced Cedalion, a framework for integrating scientific claims against their corresponding datasets, then validating them in the broader context of a public repository. We have demonstrated that our *ClaimCheck* algorithm is effective at mapping schemas of an arbitrarily specified claim to the schema of the underlying dataset by using the claim to aid in the matching process. We have also introduced *ClaimJump*, an algorithm for generalizing claims made against one dataset to a corpus of related dataset obtained from the public repository by training a classifier to impute missing attributes.

Future work will include improving the automation of the claim extraction process using semantic parsing NLP approaches, and using *ClaimCheck* to align low-confidence claims extracted from the full text of the paper to the schema of underlying dataset. A broader set of claim types can be supported, going either in a domain-specific (e.g., pathway enrichment) or domain-independent (e.g., generalized linear models) direction. *ClaimCheck* can be extended to consider mappings that map selector from the claim to disjunction of selectors on underlying data or even a range query (e.g., mapping "age=adult" into "age $>$ 21").

In conclusion, we believe that Cedalion is an important step to automate reproducibility and improve robustness in data-driven sciences. Explicitly stating scientific claims in a formal DSL can unlock new exciting applications such as empowering authors and reviewers to automatically generate replication report cards at the time of paper submission.

Algorithm 2 *ClaimCheck* algorithm

Input: A claim $C = (Q, p)$ and a dataset D where the claim is assumed to be true.

Output: A mapping M associating each selector in C with a selector in D.

procedure ClaimCheck(C, D)

S_c = AllSelectors(C)

S_d = AllSelectors(D)

$M = \varnothing$

for $s \in S_c$ **do**

if exists s' in S_d with $s.value == s'.value$ **then**

$M[s] = s'$

S_c -= s

end if

end for

M = Optimize(Objective, $M_0 = M$)

return M

end procedure

procedure Objective(C, D, M)

$Q, p = C$

$R = Q(p(M(D)))$

for $(s, s') \in M$ **do**

R += α ILD(s, s')

end for

return R

end procedure

Algorithm 3 *ClaimJump* algorithm

Input: A mean shift claim $C = (G \mid p_1 > G \mid p_2)$ and a dataset D where the claim is assumed to be true, another dataset D'.
Output: A vector of mean shifts, one for each draw from the Bayesian model.
procedure ClaimJump(C, D, D')
    μ_1 = EstimateImputedMean(D, p_1, D', G)
    μ_2 = EstimateImputedMean(D, p_2, D', G)
    return $\mu_1 - \mu_2$
end procedure
procedure EstimateImputedMean(D, p, D', g)
    A = $schema(D) \cap schema(D')$
    X = $\pi_A(D)$, $X' = \pi_A(D')$
    y = $[p(x)|x \in D]$
    predictions = $\varnothing$
    for i = 1 to 50 **do**
        classifier = BuildClassifier(X, y)
        prediction = Predict(classifier, X')
        predictions $\leftarrow$ predictions $\cup$ prediction
    end for
    values = $[x.g|x \in D']$
    μ = RunBayesianModel(predictions, values)
    return μ
end procedure

Chapter 8

CONCLUSIONS

This work uses the gene expression data repository (GEO) as an example to demonstrate potential capabilities of repository-scale curation services. In the thesis statement, we claim that this services "can improve data reuse, facilitate data discovery, enforce policy compliance, and enable reproducibility." To prove this point, several systems have been introduced throughout the chapters of this work, addressing each of the individual pieces of the claim. Specifically:

- In Chapters 5 and 6 we have introduced Pathway Graphical Lasso and DISCERN, two Machine Learning approaches that enable reuse by providing generically applicable techniques for learning gene regulatory networks from publicly available datasets (in case of Pathway Graphical Lasso) and even running comparative analysis on pairs of datasets (or subsets of one dataset) to find network-perturbed genes (in case of DISCERN). This methods, methods serve as on of the top-level curation services: they assume that the data is already available and discoverable and extract additional information from it in a more structured format (e.g., tissue-specific regulatory networks).

- In Chapter 4 we have shown *EZLearn*, a data curation framework that automatically assigns tissue annotation labels to samples from GEO without the need for manually annotated labels. Instead, *EZLearn* is relying on *organic supervision* naturally present in the scientific data repository. We introduce a novel combination of distant supervision and co-training that uses both underlying gene expression samples and human-provided noisy text descriptions, outperforming previous state-of-the art supervised approaches. *EZLearn* annotations can enable higher-quality query interface

to the repository than currently available, helping repository users discover the samples from the tissue they need.

- *Wide-Open*, introduced in Chapter 3, helps enforce compliance with open access policies by automatically mining open access publications for dataset references and cross-checking them against the database of publicly available ones. In just the first weeks, working together with GEO curators we have found and made available over 400 "overdue" datasets. Built on the simple idea of mining the literature for consistent identifier patterns, *Wide-Open* has already inspired follow up work [213].

- Finally, Chapter 7 introduced Cedalion, a system that enables reproducibility studies in GEO by allowing curators to explicitly encode claims made by the studies in a domain-specific language called *ClaimQL*. Cedalion then uses the claims to curate the original dataset, making the claim validation possible, and attempts to generalize the claim to other relevant datasets in the repository. To reduce bias and p-hacking concerns during the generalization, Cedalion focuses on the claims already explicitly stated in the papers (as opposed to greedily mining the data for new ones) and carefully tracks the uncertainty of the generalization process using a Bayesian model.

This confluence of tools and approaches sits squarely in the middle between data capture and analysis, providing curation services at the scale of the repository. Decentralized data-intensive fields like biology are a fertile ground for developing and deploying such techniques.

8.1 Limitations and future work

There are numerous limitations of this work (that are excellent avenues for future research), including:

- *Wide-Open* is limited by papers available in the open access portion of PubMed. Getting access to broader range of papers will undoubtedly lead to detecting even more overdue datasets.

- Similarly, *Wide-Open* can be extended to scan for references to items from other repositories with consistent identifiers. In fact, a follow-up paper has done just that and used text-mining approaches to scan literature for new fungus taxonomic names [213].

- *EZLearn* is also limited by a subset of data it processes in GEO. Limitation here is partly mechanical (getting more data will require writing specific processing code for each new platform) and partly methodological – different data platforms will have slightly different sets of genes available. Platform alignment problem could probably be solved by extending the denoising autoencoder and treating unobserved gene expression values as missing.

- Furthermore, *EZLearn* classifiers were designed to demonstrate that a tissue annotation approach that is not relying on manually labeled data can be successful and competitive with state of the art supervised approaches. The classifiers are thus simple, to avoid the need of extensive fine-tuning. With a bit of training data, much better classifiers could likely be developed.

- Pathway Graphical Lasso, while being really efficient at learning regulatory networks when the pathway structure is known, provides no guidance on what to do when the structure is unknown. Follow-up work [116] addresses this concern to some extent already, but clearly more thought could be placed in designing pathway-discovery approaches.

- Cedalion, while using prior work [196] to automatically extract some claims, would be made significantly more useful by a fully end-to-end NLP-based claim extraction system. This is clearly predicated on having access to full texts of the articles, though.

Looking forward, a combination of systems similar to those described in this work can make public data repositories into more *active* systems — instead of being just data storage services, they can perform curation an unsupervised way using techniques from Chapter 4.

Furthermore, coupled with an NLP-based claim extraction system, a data repository (e.g., GEO) could integrate with paper repository (e.g., PubMed) and automatically keep up-to-date report card for each paper, summarizing the evidence for and against claims made by the paper based on the all datasets in the repository, including those that have been released *after* the paper was published.

BIBLIOGRAPHY

[1] Alex Abramovici, William E Althouse, Ronald WP Drever, Yekta Gürsel, Seiji Kawamura, Frederick J Raab, David Shoemaker, Lisa Sievers, Robert E Spero, Kip S Thorne, et al. LIGO: The laser interferometer gravitational-wave observatory. *Science*, 256(5055):325–333, 1992.

[2] E. Adriaenssens, E. Vanhecke, P. Saule, A. Mougel, A. Page, R. Romon, V. Nurcombe, X. Le Bourhis, and H. Hondermarck. Nerve growth factor is a potential therapeutic target in breast cancer. *Cancer Res.*, 68(2):346–351, Jan 2008.

[3] Ludmil B. Alexandrov, Serena Nik-Zainal, David C. Wedge, Samuel A. J. R. Aparicio, Sam Behjati, Andrew V. Biankin, Graham R. Bignell, Niccolò Bolli, Ake Borg, Anne-Lise Bø rresen Dale, Sandrine Boyault, Birgit Burkhardt, Adam P. Butler, Carlos Caldas, Helen R. Davies, Christine Desmedt, Roland Eils, Jórunn Erla Eyfjörd, John A. Foekens, Mel Greaves, Fumie Hosoda, Barbara Hutter, Tomislav Ilicic, Sandrine Imbeaud, Marcin Imielinsk, Natalie Jäger, David T. W. Jones, David Jones, Stian Knappskog, Marcel Kool, Sunil R. Lakhani, Carlos López-Otín, Sancha Martin, Nikhil C. Munshi, Hiromi Nakamura, Paul A. Northcott, Marina Pajic, Elli Papaemmanuil, Angelo Paradiso, John V. Pearson, Xose S. Puente, Keiran Raine, Manasa Ramakrishna, Andrea L. Richardson, Julia Richter, Philip Rosenstiel, Matthias Schlesner, Ton N. Schumacher, Paul N. Span, Jon W. Teague, Yasushi Totoki, Andrew N. J. Tutt, Rafael Valdés-Mas, Marit M. van Buuren, Laura van t Veer, Anne Vincent-Salomon, Nicola Waddell, Lucy R. Yates, Jessica Zucman-Rossi, P. Andrew Futreal, Ultan McDermott, Peter Lichter, Matthew Meyerson, Sean M. Grimmond, Reiner Siebert, Elías Campo, Tatsuhiro Shibata, Stefan M. Pfister, Peter J. Campbell, and Michael R. Stratton. Signatures of mutational processes in human cancer. *Nature*, August 2013.

[4] R. A. Alharbi, R. Pettengell, H. S. Pandha, and R. Morgan. The role of HOX genes in normal hematopoiesis and acute leukemia. *Leukemia*, 27(5):1000–1008, Apr 2013.

[5] David Amar, Hershel Safer, and Ron Shamir. Dissection of regulatory networks that are altered in disease via differential co-expression. *PLoS computational biology*, 9(3):e1002955, January 2013.

[6] Christophe Ambroise, Julien Chiquet, and Catherine Matias. Inferring sparse Gaussian graphical models with latent structure. *Electronic Journal of Statistics*, 3:205–238, 2009.

[7] Piotr J. Balwierz, Mikhail Pachkov, Phil Arnold, Andreas J. Gruber, Mihaela Zavolan, and Erik van Nimwegen. ISMARA: automated modeling of genomic signals as a democracy of regulatory motifs. *Genome Research*, 2014.

[8] O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *JMLR*, 9:485–516, 2008.

[9] Onureena Banerjee, Laurent El Ghaoui, and Alexandre dAspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine learning research*, 9(Mar):485–516, 2008.

[10] Roger Barga, Jared Jackson, Nelson Araujo, Dean Guo, Nitin Gautam, and Yogesh Simmhan. The trident scientific workflow workbench. In *eScience, 2008. eScience'08. IEEE Fourth International Conference on*, pages 317–318. IEEE, 2008.

[11] Tanya Barrett, Stephen E Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F Kim, Maxim Tomashevsky, Kimberly A Marshall, Katherine H Phillippy, Patti M Sherman, Michelle Holko, Andrey Yefanov, Hyeseung Lee, Naigong Zhang, Cynthia L Robertson, Nadezhda Serova, Sean Davis, and Alexandra Soboleva. NCBI GEO: archive for functional genomics data sets–update. *Nucleic acids research*, 41(Database issue):D991–5, January 2013.

[12] Zafer Barutcuoglu, Robert E Schapire, and Olga G Troyanskaya. Hierarchical multilabel prediction of gene function. *Bioinformatics*, 22(7):830–836, 2006.

[13] Ali Bashashati, Gholamreza Haffari, Jiarui Ding, Gavin Ha, Kenneth Lui, Jamie Rosner, David G Huntsman, Carlos Caldas, Samuel A Aparicio, and Sohrab P Shah. DriverNet: uncovering the impact of somatic driver mutations on transcriptional networks in cancer. *Genome biology*, 13(12):R124, December 2012.

[14] Stephen B Baylin and Peter A Jones. A decade of exploring the cancer epigenome - biological and translational implications. *Nature reviews. Cancer*, 11(10):726–34, October 2011.

[15] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic Parsing on Freebase from Question-Answer Pairs. In *EMNLP*, volume 2, 2013.

[16] Michael F Berger, Michael S Lawrence, Francesca Demichelis, Yotam Drier, Kristian Cibulskis, Andrey Y Sivachenko, Andrea Sboner, Raquel Esgueva, Dorothee Pflueger, Carrie Sougnez, Robert Onofrio, Scott L Carter, Kyung Park, Lukas Habegger, Lauren Ambrogio, Timothy Fennell, Melissa Parkin, Gordon Saksena, Douglas Voet, Alex H

Ramos, Trevor J Pugh, Jane Wilkinson, Sheila Fisher, Wendy Winckler, Scott Mahan, Kristin Ardlie, Jennifer Baldwin, Jonathan W Simons, Naoki Kitabayashi, Theresa Y MacDonald, Philip W Kantoff, Lynda Chin, Stacey B Gabriel, Mark B Gerstein, Todd R Golub, Matthew Meyerson, Ashutosh Tewari, Eric S Lander, Gad Getz, Mark A Rubin, and Levi A Garraway. The genomic complexity of primary human prostate cancer. *Nature*, 470(7333):214–20, February 2011.

[17] James Bergstra, Dan Yamins, and David D Cox. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. In *Proceedings of the 12th Python in Science Conference*, pages 13–20, 2013.

[18] James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International Conference on Machine Learning*, pages 115–123, 2013.

[19] James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554, 2011.

[20] Philip A Bernstein, Jayant Madhavan, and Erhard Rahm. Generic schema matching, ten years later. *Proceedings of the VLDB Endowment*, 4(11):695–701, 2011.

[21] Nitin Bhardwaj, Philip M Kim, and Mark B Gerstein. Rewiring of transcriptional regulatory networks: hierarchy, rather than connectivity, better reflects the importance of regulators. *Science signaling*, 3(146):ra79, January 2010.

[22] Katherine Bianco, Matthew Gormley, Jason Farrell, Yan Zhou, Oliver Oliverio, Hannah Tilden, Michael McMaster, and Susan J Fisher. Placental transcriptomes in the common aneuploidies reveal critical regions on the trisomic chromosomes and genome-wide effects. *Prenatal diagnosis*, 36(9):812–822, 2016.

[23] John Blitzer, Mark Dredze, Fernando Pereira, et al. Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In *ACL*, 2007.

[24] Theodora Bloom, Emma Ganley, and Margaret Winker. Data access for the open access literature: PLOS's data policy. *PLoS biology*, 12(2):e1001797, 2014.

[25] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, 1998.

[26] Michael Bockmayr, Frederick Klauschen, Balazs Györffy, Carsten Denkert, and Jan Budczies. New network topology approaches reveal differential correlation patterns in breast cancer. *BMC systems biology*, 7(1):78, August 2013.

[27] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[28] Alvis Brazma, Pascal Hingamp, John Quackenbush, Gavin Sherlock, Paul Spellman, Chris Stoeckert, John Aach, Wilhelm Ansorge, Catherine A Ball, Helen C Causton, et al. Minimum information about a microarray experiment (MIAME)toward standards for microarray data. *Nature genetics*, 29(4):365–371, 2001.

[29] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[30] Patrick Cahan, Felicia Rovegno, Denise Mooney, John C Newman, Georges St Laurent, and Timothy A McCaffrey. Meta-analysis of microarray results: challenges, opportunities, and recommendations for standardization. *Gene*, 401(1):12–18, 2007.

[31] Qingqing Cai and Alexander Yates. Large-scale Semantic Parsing via Schema Matching and Lexicon Extension. In *ACL (1)*, pages 423–433, 2013.

[32] Andrea Califano. Rewiring makes the difference. *Molecular systems biology*, 7:463, January 2011.

[33] Steven P Callahan, Juliana Freire, Emanuele Santos, Carlos E Scheidegger, Cláudio T Silva, and Huy T Vo. VisTrails: visualization meets data management. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 745–747. ACM, 2006.

[34] Bob Carpenter, Andrew Gelman, Matt Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Michael A Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software*, 20:1–37, 2016.

[35] Hannah Carter, Sining Chen, Leyla Isik, Svitlana Tyekucheva, Victor E Velculescu, Kenneth W Kinzler, Bert Vogelstein, and Rachel Karchin. Cancer-specific high-throughput annotation of somatic mutations: computational prediction of driver missense mutations. *Cancer research*, 69(16):6660–7, August 2009.

[36] Arturo Casadevall and Ferric C Fang. Reproducible science, 2010.

[37] Daniel Castellani Ribeiro, Huy T. Vo, Juliana Freire, and Cláudio T. Silva. An Urban Data Profiler. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, pages 1389–1394, New York, NY, USA, 2015. ACM.

[38] Safiye Celik, Ben Logsdon, and Su-In Lee. Efficient Dimensionality Reduction for High-Dimensional Network Estimation. *International Conference on Machine Learning (ICML)*, 2014.

[39] Safiye Celik, Benjamin A. Logsdon, Stephanie Battle, Charles W. Drescher, Mara Rendi, R. David Hawkins, and Su-In Lee. Extracting a low-dimensional description of multiple gene expression datasets reveals a potential driver for tumor-associated stroma in ovarian cancer. *Genome Medicine*, 8(1):66, 2016.

[40] Michael A Chapman, Michael S Lawrence, Jonathan J Keats, Kristian Cibulskis, Carrie Sougnez, Anna C Schinzel, Christina L Harview, Jean-Philippe Brunet, Gregory J Ahmann, Mazhar Adli, Kenneth C Anderson, Kristin G Ardlie, Daniel Auclair, Angela Baker, P Leif Bergsagel, Bradley E Bernstein, Yotam Drier, Rafael Fonseca, Stacey B Gabriel, Craig C Hofmeister, Sundar Jagannath, Andrzej J Jakubowiak, Amrita Krishnan, Joan Levy, Ted Liefeld, Sagar Lonial, Scott Mahan, Bunmi Mfuko, Stefano Monti, Louise M Perkins, Robb Onofrio, Trevor J Pugh, S Vincent Rajkumar, Alex H Ramos, David S Siegel, Andrey Sivachenko, A Keith Stewart, Suzanne Trudel, Ravi Vij, Douglas Voet, Wendy Winckler, Todd Zimmerman, John Carpten, Jeff Trent, William C Hahn, Levi A Garraway, Matthew Meyerson, Eric S Lander, Gad Getz, and Todd R Golub. Initial genome sequencing and analysis of multiple myeloma. *Nature*, 471(7339):467–72, March 2011.

[41] C. H. Chen, S. M. Chuang, M. F. Yang, J. W. Liao, S. L. Yu, and J. J. Chen. A novel function of YWHAZ/-catenin axis in promoting epithelial-mesenchymal transition and lung cancer metastasis. *Mol. Cancer Res.*, 10(10):1319–1331, Oct 2012.

[42] Fernando Chirigati, Harish Doraiswamy, Theodoros Damoulas, and Juliana Freire. Data Polygamy: The Many-Many Relationships Among Urban Spatio-Temporal Data Sets. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD '16, pages 1011–1025, New York, NY, USA, 2016. ACM.

[43] Fernando Seabra Chirigati, Dennis E Shasha, and Juliana Freire. ReproZip: Using Provenance to Support Computational Reproducibility. In *TaPP*, 2013.

[44] François Chollet. Keras. https://github.com/fchollet/keras, 2015.

[45] Xu Chu, Ihab F Ilyas, Sanjay Krishnan, and Jiannan Wang. Data cleaning: Overview and emerging challenges. In *Proceedings of the 2016 International Conference on Management of Data*, pages 2201–2206. ACM, 2016.

[46] Giovanni Ciriello, Ethan Cerami, Chris Sander, and Nikolaus Schultz. Mutual exclusivity analysis identifies oncogenic network modules. *Genome research*, 22(2):398–406, February 2012.

[47] Emily Clough and Tanya Barrett. The Gene Expression Omnibus database. *Methods Mol Biol*, 1418:93–110, 2016.

[48] Kathleen Collins, Tyler Jacks, and Nikola P. Pavletich. The cell cycle and cancer. *Proceedings of the National Academy of Sciences*, 94(7):2776–2778, 1997.

[49] Melissa H Cragin, P Bryan Heidorn, Carole L Palmer, and Linda C Smith. An educational program on data curation. 2007.

[50] David Croft, Gavin OKelly, Guanming Wu, Robin Haw, Marc Gillespie, Lisa Matthews, Michael Caudy, Phani Garapati, Gopal Gopinath, Bijay Jassal, Steven Jupe, Irina Kalatskaya, Shahana Mahajan, Bruce May, Nelson Ndegwa, Esther Schmidt, Veronica Shamovsky, Christina Yung, Ewan Birney, Henning Hermjakob, Peter DEustachio, and Lincoln Stein. Reactome: a database of reactions, pathways and biological processes. *Nucleic Acids Research*, 39(suppl 1):D691–D697, 2011.

[51] Christina Curtis, Sohrab P Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M Rueda, Mark J Dunning, Doug Speed, Andy G Lynch, Shamith Samarajiwa, Yinyin Yuan, et al. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352, 2012.

[52] Manhong Dai, Pinglang Wang, Andrew D Boyd, Georgi Kostov, Brian Athey, Edward G Jones, William E Bunney, Richard M Myers, Terry P Speed, Huda Akil, Stanley J Watson, and Fan Meng. Evolving gene/transcript definitions significantly alter the interpretation of GeneChip data. *Nucleic acids research*, 33(20):e175, January 2005.

[53] Ryan K Dale, Brent S Pedersen, and Aaron R Quinlan. Pybedtools: a flexible Python library for manipulating genomic datasets and annotations. *Bioinformatics (Oxford, England)*, 27(24):3423–4, December 2011.

[54] Michele Dallachiesa, Amr Ebaid, Ahmed Eldawy, Ahmed Elmagarmid, Ihab F Ilyas, Mourad Ouzzani, and Nan Tang. NADEEF: a commodity data cleaning system. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pages 541–552. ACM, 2013.

[55] Hal Daumé III. Frustratingly Easy Domain Adaptation. *ACL 2007*, 2007.

[56] Nathan D Dees, Qunyuan Zhang, Cyriac Kandoth, Michael C Wendl, William Schierding, Daniel C Koboldt, Thomas B Mooney, Matthew B Callaway, David Dooling, Elaine R Mardis, Richard K Wilson, and Li Ding. MuSiC: identifying mutational significance in cancer genomes. *Genome research*, 22(8):1589–98, August 2012.

[57] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR 2009*, pages 248–255. IEEE, 2009.

[58] Djordje Djordjevic, Yun Xin Chen, Shu Lun Shannon Kwan, Raymond WK Ling, Gordon Qian, Chelsea YY Woo, Samuel J Ellis, and Joshua WK Ho. GEOracle: Mining perturbation experiments using free text metadata in Gene Expression Omnibus. *bioRxiv*, page 150896, 2017.

[59] AnHai Doan, Pedro Domingos, and Alon Y Halevy. Reconciling schemas of disparate data sources: A machine-learning approach. In *ACM Sigmod Record*, volume 30, pages 509–520. ACM, 2001.

[60] AnHai Doan, Alon Halevy, and Zachary Ives. *Principles of data integration*. Elsevier, 2012.

[61] Ronald D. Snee Donald W. Marquardt. Ridge Regression in Practice. *The American Statistician*, 29(1):3–20, 1975.

[62] Xin Luna Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Kevin Murphy, Shaohua Sun, and Wei Zhang. From data fusion to knowledge fusion. *Proceedings of the VLDB Endowment*, 7(10):881–892, 2014.

[63] Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. Knowledge-based trust: Estimating the trustworthiness of web sources. *Proceedings of the VLDB Endowment*, 8(9):938–949, 2015.

[64] Xin Luna Dong and Divesh Srivastava. Knowledge curation and knowledge fusion: challenges, models and applications. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 2063–2066. ACM, 2015.

[65] Jennie Duggan and Michael L Brodie. Hephaestus: Data Reuse for Accelerating Scientific Discovery. In *CIDR*, 2015.

[66] Jasmine Dumas, Michael A Gargano, and Garrett M Dancik. shinyGEO: a web-based application for analyzing gene expression omnibus datasets. *Bioinformatics*, 32(23):3679–3681, 2016.

[67] Mark Dunning, Andy Lynch, and Matthew Eldridge. *illuminaHumanv3.db: Illumina HumanHT12v3 annotation data (chip illuminaHumanv3)*. R package version 1.26.0.

[68] Ron Edgar, Michael Domrachev, and Alex E Lash. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *NAR*, 2002.

[69] Daniel J Eisenstein, David H Weinberg, Eric Agol, Hiroaki Aihara, Carlos Allende Prieto, Scott F Anderson, James A Arns, Éric Aubourg, Stephen Bailey, Eduardo Balbinot, et al. SDSS-III: Massive spectroscopic surveys of the distant universe, the Milky Way, and extra-solar planetary systems. *The Astronomical Journal*, 142(3):72, 2011.

[70] ENCODE. The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science (New York, N.Y.)*, 306(5696):636–40, October 2004.

[71] Jesse M Engreitz, Rong Chen, Alexander A Morgan, Joel T Dudley, Rohan Mallelwar, and Atul J Butte. ProfileChaser: searching microarray repositories based on genome-wide patterns of differential expression. *Bioinformatics*, 27(23):3317–3318, 2011.

[72] Benedikt Fecher, Sascha Friesike, and Marcel Hebing. What drives academic data sharing? *PloS one*, 10(2):e0118053, 2015.

[73] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE TPAMI*, 2006.

[74] D J Felleman and D C Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, N.Y. : 1991)*, 1(1):1–47, 1991.

[75] A. Field. Analysis of variance (ANOVA). *Encyclopedia of measurement and statistics.*, pages 33–36, 2007.

[76] B. Foss and O. Bruserud. Platelet functions and clinical effects in acute myelogenous leukemia. *Thromb. Haemost.*, 99(1):27–37, Jan 2008.

[77] Juliana Freire, Philippe Bonnet, and Dennis Shasha. Computational reproducibility: state-of-the-art, challenges, and database research opportunities. In *Proceedings of the 2012 ACM SIGMOD international conference on management of data*, pages 593–596. ACM, 2012.

[78] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9:432–441, 2007.

[79] J Friedman, T Hastie, and R Tibshirani. glmnet: Lasso and elastic-net regularized generalized linear models. Version1, 2013.

[80] Sascha Friesike, Bastian Widenmayer, et al. Opening science: towards an agenda of open science in academia and industry. *J. of Tech. Transfer*, 2015.

[81] M. Furukawa, J. Soh, H. Yamamoto, K. Ichimura, K. Shien, Y. Maki, T. Muraoka, N. Tanaka, T. Ueno, H. Asano, K. Tsukuda, S. Toyooka, and S. Miyoshi. Silenced expression of NFKBIA in lung adenocarcinoma patients with a never-smoking history. *Acta Med. Okayama*, 67(1):19–24, 2013.

[82] P Andrew Futreal, Lachlan Coin, Mhairi Marshall, Thomas Down, Timothy Hubbard, Richard Wooster, Nazneen Rahman, and Michael R Stratton. A census of human cancer genes. *Nature reviews. Cancer*, 4(3):177–83, March 2004.

[83] Daniel Garijo, Sarah Kinnings, Li Xie, Lei Xie, Yinliang Zhang, Philip E Bourne, and Yolanda Gil. Quantifying reproducibility in computational biology: the case of the tuberculosis drugome. *PloS one*, 8(11):e80278, 2013.

[84] Laurent Gautier, Leslie Cope, Benjamin M Bolstad, and Rafael A Irizarry. affy–analysis of Affymetrix GeneChip data at the probe level. *Bioinformatics (Oxford, England)*, 20(3):307–15, February 2004.

[85] Andrew J Gentles, Ash A Alizadeh, Su-In Lee, June H Myklebust, Catherine M Shachaf, Babak Shahbaba, Ronald Levy, Daphne Koller, and Sylvia K Plevritis. A pluripotency signature predicts histologic transformation and influences survival in follicular lymphoma patients. *Blood*, 114(15):3158–66, October 2009.

[86] Andrew J Gentles, Sylvia K Plevritis, Ravindra Majeti, and Ash A Alizadeh. Association of a leukemic stem cell gene expression signature with clinical outcomes in acute myeloid leukemia. *JAMA : the journal of the American Medical Association*, 304(24):2706–15, December 2010.

[87] Ryan Gill, Somnath Datta, and Susmita Datta. A statistical framework for differential network analysis from microarray data. *BMC bioinformatics*, 11(1):95, January 2010.

[88] Annuska M Glas, Arno Floore, Leonie J M J Delahaye, Anke T Witteveen, Rob C F Pover, Niels Bakx, Jaana S T Lahti-Domenici, Tako J Bruinsma, Marc O Warmoes, René Bernards, Lodewyk F A Wessels, and Laura J Van't Veer. Converting a breast cancer microarray signature into a high-throughput diagnostic test. *BMC genomics*, 7(1):278, January 2006.

[89] Gene V Glass. Primary, secondary, and meta-analysis of research. *Educational researcher*, pages 3–8, 1976.

[90] J. Gouge, K. Satia, N. Guthertz, M. Widya, A. J. Thompson, P. Cousin, O. Dergai, N. Hernandez, and A. Vannini. Redox Signaling by the RNA Polymerase III TFIIB-Related Factor Brf2. *Cell*, 163(6):1375–1387, Dec 2015.

[91] Charles E. Grant, Timothy L. Bailey, and William Stafford Noble. FIMO: scanning for occurrences of a given motif. *Bioinformatics*, 27(7):1017–1018, 2011.

[92] Maxim Grechkin, Maryam Fazel, Daniela Witten, and Su-In Lee. Pathway graphical lasso. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 2015, page 2617. NIH Public Access, 2015.

[93] Maxim Grechkin, Benjamin A Logsdon, Andrew J Gentles, and Su-In Lee. Identifying Network Perturbation in Cancer. *PLoS Comput Biol*, 2016.

[94] Maxim Grechkin, Hoifung Poon, and Bill Howe. EZLearn: Exploiting Organic Supervision in Large-Scale Data Annotation. *arXiv preprint arXiv:1709.08600*, 2017.

[95] Maxim Grechkin, Hoifung Poon, and Bill Howe. Wide-Open: Accelerating public data release by automating detection of overdue datasets. *PLoS biology*, 15(6):e2002477, 2017.

[96] Casey S Greene, Lana X Garmire, Jack A Gilbert, Marylyn D Ritchie, and Lawrence E Hunter. Celebrating parasites. *Nature Genetics*, 49(4):483–484, 2017.

[97] Casey S Greene and Olga G Troyanskaya. PILGRM: an interactive data-driven discovery platform for expert biologists. *Nucleic acids research*, 39(suppl 2):W368–W374, 2011.

[98] Marion Gremse, Antje Chang, Ida Schomburg, Andreas Grote, Maurice Scheer, Christian Ebeling, and Dietmar Schomburg. The BRENDA Tissue Ontology (BTO): the first all-integrating ontology of all organisms for enzyme sources. *Nucleic Acids Research*, 2011.

[99] Yuanfang Guan, Maitreya J Dunham, Olga G Troyanskaya, and Amy A Caudy. Comparative gene expression between two yeast species. *BMC genomics*, 14:33, January 2013.

[100] Philip J. Guo and Dawson Engler. CDE: Using System Call Interposition to Automatically Create Portable Software Packages. In *Proceedings of the 2011 USENIX Annual Technical Conference*, USENIX'11, Berkeley, CA, USA, 2011. USENIX Association.

[101] Ashish Gupta, Inderpal Singh Mumick, and Venkatramanan Siva Subrahmanian. Maintaining views incrementally. *ACM SIGMOD Record*, 22(2):157–166, 1993.

[102] Maria Gutierrez-Arcelus, Halit Ongen, Tuuli Lappalainen, et al. Tissue-specific effects of genetic and epigenetic variation on gene regulation and splicing. *PLoS Genet*, 2015.

[103] Naomi Habib, Ilan Wapinski, Hanah Margalit, Aviv Regev, and Nir Friedman. A functional selection model explains evolutionary robustness despite plasticity in regulatory networks. *Molecular systems biology*, 8:619, January 2012.

[104] Torsten Haferlach, Alexander Kohlmann, Lothar Wieczorek, Giuseppe Basso, Geertruy Te Kronnie, Marie-Christine Béné, John De Vos, Jesus M Hernández, Wolf-Karsten Hofmann, Ken I Mills, Amanda Gilkes, Sabina Chiaretti, Sheila A Shurtleff, Thomas J Kipps, Laura Z Rassenti, Allen E Yeoh, Peter R Papenhausen, Wei-Min Liu, P Mickey Williams, and Robin Foà. Clinical utility of microarray-based gene expression profiling in the diagnosis and subclassification of leukemia: report from the International Microarray Innovations in Leukemia Study Group. *Journal of clinical oncology : official journal of the American Society of Clinical Oncology*, 28(15):2529–37, May 2010.

[105] Douglas Hanahan and Robert A Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–74, March 2011.

[106] Chris H J Hartgerink. Publication cycle: A case study of the Public Library of Science (PLOS), Dec 2015.

[107] Les Hatton and Gregory Warr. Full Computational Reproducibility in Biological Science: Methods, Software and a Case Study in Protein Biology. *arXiv preprint arXiv:1608.06897*, 2016.

[108] Erika Check Hayden. The automated lab. *Nature News*, 516(7529):131, 2014.

[109] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[110] Marti Hearst. Noun homograph disambiguation using local context in large text corpora. *Using Corpora*, pages 185–188, 1991.

[111] Laura M Heiser, Anguraj Sadanandam, Wen-Lin Kuo, Stephen C Benz, Theodore C Goldstein, Sam Ng, William J Gibb, Nicholas J Wang, Safiyyah Ziyad, Frances Tong, et al. Subtype and pathway specific responses to anticancer compounds in breast cancer. *PNAS*, 109(8):2724–2729, 2012.

[112] Joel Herndon and Robert O'Reilly. Data Sharing Policies in Social Sciences Academic Journals: Evolving Expectations of Data Sharing as a Form of Scholarly Communication. *Databrarianship: The Academic Data Librarian in Theory and Practice*, 2016.

[113] Tony Hey, Stewart Tansley, Kristin M Tolle, et al. *The fourth paradigm: data-intensive scientific discovery*, volume 1. Microsoft research Redmond, WA, 2009.

[114] Matthew A Hibbs, David C Hess, Chad L Myers, Curtis Huttenhower, Kai Li, and Olga G Troyanskaya. Exploring the functional landscape of gene expression: directed search of large microarray compendia. *Bioinformatics*, 23(20):2692–2699, 2007.

[115] Jean Honorio, Dimitris Samaras, Nikos Paragios, Rita Goldstein, and Luis E. Ortiz. Sparse and Locally Constant Gaussian Graphical Models. In Y. Bengio, D. Schuurmans, J.D. Lafferty, C.K.I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 745–753. Curran Associates, Inc., 2009.

[116] Mohammad Javad Hosseini and Su-In Lee. Learning sparse gaussian graphical models with overlapping blocks. In *Advances in Neural Information Processing Systems*, pages 3808–3816, 2016.

[117] Bill Howe, Po-shen Lee, Maxim Grechkin, Sean T Yang, and Jevin D West. Deep Mapping of the Visual Literature. In *WWW Companion*, pages 1273–1277, 2017.

[118] Cho-Jui Hsieh, Inderjit S Dhillon, Pradeep Ravikumar, and Arindam Banerjee. A divide-and-conquer procedure for sparse inverse covariance estimation. *NIPS*, 2012.

[119] Cho-Jui Hsieh, Matyas A. Sustik, Inderjit S. Dhillon, and Pradeep Ravikumar. Sparse Inverse Covariance Matrix Estimation Using Quadratic Approximation. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2330–2338. http://nips.cc/, 2011.

[120] http://www.biggorilla.org. Biggorilla: Data integration and data preparation in python., 2017.

[121] Darrell Huff. *How to lie with statistics*. WW Norton & Company, 2010.

[122] Trey Ideker and Nevan J Krogan. Differential network biology. *Molecular systems biology*, 8:565, January 2012.

[123] John PA Ioannidis. Why most published research findings are false. *PLoS medicine*, 2(8):e124, 2005.

[124] John PA Ioannidis, David B Allison, Catherine A Ball, Issa Coulibaly, Xiangqin Cui, Aedín C Culhane, Mario Falchi, Cesare Furlanello, Laurence Game, Giuseppe Jurman, et al. Repeatability of published microarray gene expression analyses. *Nature genetics*, 41(2):149–155, 2009.

[125] T. Jaatinen and J. Laine. Isolation of hematopoietic stem cells from human cord blood. *Curr Protoc Stem Cell Biol*, Chapter 2:Unit 2A.2, Jun 2007.

[126] W Evan Johnson, Cheng Li, and Ariel Rabinovic. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics (Oxford, England)*, 8(1):118–27, January 2007.

[127] Vladimir Jojic, Tal Shay, Katelyn Sylvia, Or Zuk, Xin Sun, Joonsoo Kang, Aviv Regev, Daphne Koller, Adam J Best, Jamie Knell, Ananda Goldrath, Nadia Cohen, Patrick Brennan, Michael Brenner, Francis Kim, Tata Nageswara Rao, Amy Wagers, Tracy Heng, Jeffrey Ericson, Katherine Rothamel, Adriana Ortiz-Lopez, Diane Mathis, Christophe Benoist, Natalie A Bezman, Joseph C Sun, Gundula Min-Oo, Charlie C Kim, Lewis L Lanier, Jennifer Miller, Brian Brown, Miriam Merad, Emmanuel L Gautier, Claudia Jakubzick, Gwendalyn J Randolph, Paul Monach, David A Blair, Michael L Dustin, Susan A Shinton, Richard R Hardy, David Laidlaw, Jim Collins, Roi Gazit, Derrick J Rossi, Nidhi Malhotra, Taras Kreslavsky, Anne Fletcher, Kutlu Elpek, Angelique Bellemare-Pelletier, Deepali Malhotra, and Shannon Turley. Identification of transcriptional regulators in the mouse immune system. *Nature immunology*, 14(6):633–43, June 2013.

[128] Daniel C Jones, Walter L Ruzzo, Xinxia Peng, and Michael G Katze. A new approach to bias correction in RNA-Seq. *Bioinformatics*, 28(7):921–928, 2012.

[129] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–.

[130] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of Tricks for Efficient Text Classification. *EACL 2017*, page 427, 2017.

[131] L. Karabon, E. Pawlak, A. Tomkiewicz, A. Jedynak, E. Passowicz-Muszynska, K. Zajda, A. Jonkisz, R. Jankowska, M. Krzakowski, and I. Frydecka. CTLA-4, CD28, and ICOS gene polymorphism associations with non-small-cell lung cancer. *Hum. Immunol.*, 72(10):947–954, Oct 2011.

[132] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and a. D. Haussler. The Human Genome Browser at UCSC. *Genome Research*, 12(6):996–1006, May 2002.

[133] M. G. Kharas, C. J. Lengner, F. Al-Shahrour, L. Bullinger, B. Ball, S. Zaidi, K. Morgan, W. Tam, M. Paktinat, R. Okabe, M. Gozo, W. Einhorn, S. W. Lane, C. Scholl, S. Frohling, M. Fleming, B. L. Ebert, D. G. Gilliland, R. Jaenisch, and G. Q. Daley.

Musashi-2 regulates normal hematopoiesis and promotes aggressive myeloid leukemia. *Nat. Med.*, 16(8):903–908, Aug 2010.

[134] Purvesh Khatri, Silke Roedder, Naoyuki Kimura, Katrien De Vusser, Alexander A Morgan, Yongquan Gong, Michael P Fischbein, Robert C Robbins, Maarten Naesens, Atul J Butte, et al. A common rejection module (CRM) for acute rejection across multiple organs identifies novel therapeutics for organ transplantation. *Journal of Experimental Medicine*, 210(11):2205–2221, 2013.

[135] Jinho Kim, Inhae Kim, Seong Kyu Han, James U Bowie, and Sanguk Kim. Network rewiring is an important mechanism of gene essentiality change. *Scientific reports*, 2:900, January 2012.

[136] Ross D King, Jem Rowland, Stephen G Oliver, Michael Young, Wayne Aubrey, Emma Byrne, Maria Liakata, Magdalena Markham, Pinar Pir, Larisa N Soldatova, et al. The automation of science. *Science*, 324(5923):85–89, 2009.

[137] Ross D King, Kenneth E Whelan, Ffion M Jones, Philip GK Reiser, Christopher H Bryant, Stephen H Muggleton, Douglas B Kell, and Stephen G Oliver. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427(6971):247, 2004.

[138] C. G. Kleer, Q. Cao, S. Varambally, R. Shen, I. Ota, S. A. Tomlins, D. Ghosh, R. G. Sewalt, A. P. Otte, D. F. Hayes, M. S. Sabel, D. Livant, S. J. Weiss, M. A. Rubin, and A. M. Chinnaiyan. EZH2 is a marker of aggressive breast cancer and promotes neoplastic transformation of breast epithelial cells. *Proc. Natl. Acad. Sci. U.S.A.*, 100(20):11606–11611, Sep 2003.

[139] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian E Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica B Hamrick, Jason Grout, Sylvain Corlay, et al. Jupyter Notebooks-a publishing format for reproducible computational workflows. In *ELPUB*, pages 87–90, 2016.

[140] Katja Koeppen, Bruce A Stanton, and Thomas H Hampton. ScanGEO: parallel mining of high-throughput gene expression data. *Bioinformatics*, 33(21):3500–3501, 2017.

[141] P. D. Kottaridis, R. E. Gale, M. E. Frew, G. Harrison, S. E. Langabeer, A. A. Belton, H. Walker, K. Wheatley, D. T. Bowen, A. K. Burnett, A. H. Goldstone, and D. C. Linch. The presence of a FLT3 internal tandem duplication in patients with acute myeloid leukemia (AML) adds important prognostic information to cytogenetic risk group and response to the first cycle of chemotherapy: analysis of 854 patients from the

United Kingdom Medical Research Council AML 10 and 12 trials. *Blood*, 98(6):1752–1759, Sep 2001.

[142] Sanjay Krishnan, Jiannan Wang, Eugene Wu, Michael J Franklin, and Ken Goldberg. ActiveClean: interactive data cleaning for statistical modeling. *Proceedings of the VLDB Endowment*, 9(12):948–959, 2016.

[143] C. Lagadec, S. Meignan, E. Adriaenssens, B. Foveau, E. Vanhecke, R. Romon, R. A. Toillon, B. Oxombre, H. Hondermarck, and X. Le Bourhis. TrkA overexpression enhances growth and metastasis of breast cancer cells. *Oncogene*, 28(18):1960–1970, May 2009.

[144] 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.

[145] M. Lanotte, Martin-Thouvenin V, Najman S, Balerini P, Valensi F, and Berger R. NB4, a maturation inducible cell line with t(15;17) marker isolated from a human acute promyelocytic leukemia (M3). *Blood*, 77 (5):1080–6, Mar 1991.

[146] Ola Larsson and Rickard Sandberg. Lack of correct data format and comparability limits future integrative microarray research. *Nature biotechnology*, 24(11):1322–1323, 2006.

[147] S.L. Lauritzen. *Graphical Models*. Oxford Science Publications, 1996.

[148] Steffen L Lauritzen and Nanny Wermuth. Graphical models for associations between variables, some of which are qualitative and some quantitative. *The annals of Statistics*, pages 31–57, 1989.

[149] C. W. Law, Y. Chen, W. Shi, and G. K. Smyth. voom: Precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol.*, 15(2):R29, 2014.

[150] Michael S. Lawrence, Petar Stojanov, Paz Polak, Gregory V. Kryukov, Kristian Cibulskis, Andrey Sivachenko, Scott L. Carter, Chip Stewart, Craig H. Mermel, Steven A. Roberts, Adam Kiezun, Peter S. Hammerman, Aaron McKenna, Yotam Drier, Lihua Zou, Alex H. Ramos, Trevor J. Pugh, Nicolas Stransky, Elena Helman, Jaegil Kim, Carrie Sougnez, Lauren Ambrogio, Elizabeth Nickerson, Erica Shefler, Maria L. Cortes, Daniel Auclair, Gordon Saksena, Douglas Voet, Michael Noble, Daniel DiCara, Pei Lin, Lee Lichtenstein, David I. Heiman, Timothy Fennell, Marcin Imielinski, Bryan Hernandez, Eran Hodis, Sylvan Baca, Austin M. Dulak, Jens Lohr, Dan-Avi Landau,

Catherine J. Wu, Jorge Melendez-Zajgla, Alfredo Hidalgo-Miranda, Amnon Koren, Steven A. McCarroll, Jaume Mora, Ryan S. Lee, Brian Crompton, Robert Onofrio, Melissa Parkin, Wendy Winckler, Kristin Ardlie, Stacey B. Gabriel, Charles W. M. Roberts, Jaclyn A. Biegel, Kimberly Stegmaier, Adam J. Bass, Levi A. Garraway, Matthew Meyerson, Todd R. Golub, Dmitry A. Gordenin, Shamil Sunyaev, Eric S. Lander, and Gad Getz. Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature*, 499(7457):214–218, Jul 2013. Letter.

[151] Po-shen Lee, Jevin D West, and Bill Howe. Viziometrics: Analyzing visual information in the scientific literature. *IEEE Transactions on Big Data*, 2017.

[152] Su-In Lee, Aimée M Dudley, David Drubin, Pamela A Silver, Nevan J Krogan, Dana Pe'er, and Daphne Koller. Learning a prior on regulatory potential from eQTL data. *PLoS genetics*, 5(1):e1000358, January 2009.

[153] Tong Ihn Lee and Richard A Young. Transcriptional regulation and its misregulation in disease. *Cell*, 152(6):1237–51, March 2013.

[154] Rasko Leinonen, Hideaki Sugawara, Martin Shumway, and International Nucleotide Sequence Database Collaboration. The sequence read archive. *Nucleic acids research*, 39(suppl_1):D19–D21, 2010.

[155] Yue Li, Minggao Liang, and Zhaolei Zhang. Regression Analysis of Combined Gene Expression Regulation in Acute Myeloid Leukemia. *PLoS Comput Biol*, 10(10):e1003908, 10 2014.

[156] Maxwell W. Libbrecht and William Stafford Noble. Machine learning applications in genetics and genomics. *Nat. Rev. Genetics*, 2015.

[157] Mark W.. Lipsey and David B Wilson. *Practical meta-analysis*, volume 49. Sage publications Thousand Oaks, CA, 2001.

[158] Qiang Liu and Alexander T Ihler. Learning scale free networks by reweighted l1 regularization. In *AISTATS*, pages 40–48, 2011.

[159] Xiong Liu, Xueping Yu, Donald J. Zack, Heng Zhu, and Jiang Qian. TiGER: A database for tissue-specific gene expression and regulation. *BMC Bioinformatics*, 9(1):271, 2008.

[160] Dan L. Longo and Jeffrey M. Drazen. Data Sharing. *New England Journal of Medicine*, 374(3):276–277, 2016. PMID: 26789876.

[161] Jakob Lovén, Heather A Hoke, Charles Y Lin, Ashley Lau, David A Orlando, Christopher R Vakoc, James E Bradner, Tong Ihn Lee, and Richard A Young. Selective inhibition of tumor oncogenes by disruption of super-enhancers. *Cell*, 153(2):320–34, April 2013.

[162] Jianguo Lu, Ju Wang, and Shengrui Wang. XML schema matching. *International Journal of Software Engineering and Knowledge Engineering*, 17(05):575–597, 2007.

[163] Margus Lukk, Misha Kapushesky, Janne Nikkilä, Helen Parkinson, Angela Goncalves, Wolfgang Huber, Esko Ukkonen, and Alvis Brazma. A global map of human gene expression. *Nature biotechnology*, 28(4):322–324, 2010.

[164] Donna Maglott, Jim Ostell, Kim D Pruitt, and Tatiana Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic acids research*, 39(Database issue):D52–7, January 2011.

[165] James Malone, Ele Holloway, Tomasz Adamusiak, et al. Modeling sample variables with an Experimental Factor Ontology. *Bioninf.*, 2010.

[166] K.V. Mardia, J. Kent, and J.M. Bibby. *Multivariate Analysis*. Academic Press, 1979.

[167] Scott E Maxwell, Michael Y Lau, and George S Howard. Is psychology suffering from a replication crisis? What does failure to replicate really mean? *American Psychologist*, 70(6):487, 2015.

[168] Rahul Mazumder, Trevor Hastie, et al. The graphical lasso: New insights and alternatives. *Electronic Journal of Statistics*, 6:2125–2149, 2012.

[169] David McClosky and Eugene Charniak. Self-training for biomedical parsing. In *ACL*, 2008.

[170] Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the Lasso. *The Annals of Statistics*, 34(3):1436–1462, June 2006.

[171] Zhaoshi Meng, Dennis Wei, Ami Wiesel, and Alfred Hero III. Distributed learning of Gaussian graphical models via marginal likelihoods. *JMLR 31: 3947*, 2013.

[172] Renée J Miller. Big Data Curation. In *COMAD*, page 4, 2014.

[173] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *ACL*, pages 1003–1011, 2009.

[174] K. Mitra, A. R. Carvunis, S. K. Ramesh, and T. Ideker. Integrative approaches for finding modular structure in biological networks. *Nat. Rev. Genet.*, 14(10):719–732, Oct 2013.

[175] Yariv Dror Mizrahi, Misha Denil, and Nando de Freitas. Linear and Parallel Learning for Markov Random Fields. In *International Conference on Machine Learning (ICML)*, 2014.

[176] Jennifer C Molloy. The open knowledge foundation: open data means better science. *PLoS Biol*, 2011.

[177] R. D. Morin, N. A. Johnson, T. M. Severson, A. J. Mungall, J. An, R. Goya, J. E. Paul, M. Boyle, B. W. Woolcock, F. Kuchenbauer, D. Yap, R. K. Humphries, O. L. Griffith, S. Shah, H. Zhu, M. Kimbara, P. Shashkin, J. F. Charlot, M. Tcherpakov, R. Corbett, A. Tam, R. Varhol, D. Smailus, M. Moksa, Y. Zhao, A. Delaney, H. Qian, I. Birol, J. Schein, R. Moore, R. Holt, D. E. Horsman, J. M. Connors, S. Jones, S. Aparicio, M. Hirst, R. D. Gascoyne, and M. A. Marra. Somatic mutations altering EZH2 (Tyr641) in follicular and diffuse large B-cell lymphomas of germinal-center origin. *Nat. Genet.*, 42(2):181–185, Feb 2010.

[178] Leonardo Murta, Vanessa Braganholo, Fernando Chirigati, David Koop, and Juliana Freire. noWorkflow: capturing and analyzing provenance of scripts. In *International Provenance and Annotation Workshop*, pages 71–83. Springer, 2014.

[179] Cancer Genome Atlas Network et al. Comprehensive molecular portraits of human breast tumors. *Nature*, 490(7418):61, 2012.

[180] Sam Ng, Eric A Collisson, Artem Sokolov, Theodore Goldstein, Abel Gonzalez-Perez, Nuria Lopez-Bigas, Christopher Benz, David Haussler, and Joshua M Stuart. PARADIGM-SHIFT predicts the function of mutations in multiple cancers using pathway impact analysis. *Bioinformatics (Oxford, England)*, 28(18):i640–i646, September 2012.

[181] Kamal Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *CIKM*, 2000.

[182] Feng Niu, Ce Zhang, Christopher Ré, and Jude W Shavlik. DeepDive: Web-scale Knowledge-base Construction using Statistical Learning and Inference. *VLDS*, 12:25–28, 2012.

[183] Tom Oinn, Matthew Addis, Justin Ferris, Darren Marvin, Martin Senger, Mark Greenwood, Tim Carver, Kevin Glover, Matthew R Pocock, Anil Wipat, et al. Taverna: a

tool for the composition and enactment of bioinformatics workflows. *Bioinformatics*, 20(17):3045–3054, 2004.

[184] Hirokazu Okayama, Takashi Kohno, Yuko Ishii, Yoko Shimada, Kouya Shiraishi, Reika Iwakawa, Koh Furuta, Koji Tsuta, Tatsuhiro Shibata, Seiichiro Yamamoto, et al. Identification of genes upregulated in ALK-positive and EGFR/KRAS/ALK-negative lung adenocarcinomas. *Cancer research*, 72(1):100–111, 2012.

[185] Soonmyung Paik, Steven Shak, Gong Tang, Chungyeul Kim, Joffre Baker, Maureen Cronin, Frederick L. Baehner, Michael G. Walker, Drew Watson, Taesung Park, William Hiller, Edwin R. Fisher, D. Lawrence Wickerham, John Bryant, and Norman Wolmark. A Multigene Assay to Predict Recurrence of Tamoxifen-Treated, Node-Negative Breast Cancer. *New England Journal of Medicine*, 351(27):2817–2826, 2004. PMID: 15591335.

[186] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, 2009.

[187] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE TKDE*, 22(10):1345–1359, 2010.

[188] J. S. Parker, M. Mullins, M. C. Cheang, S. Leung, D. Voduc, T. Vickery, S. Davies, C. Fauron, X. He, Z. Hu, J. F. Quackenbush, I. J. Stijleman, J. Palazzo, J. S. Marron, A. B. Nobel, E. Mardis, T. O. Nielsen, M. J. Ellis, C. M. Perou, and P. S. Bernard. Supervised risk predictor of breast cancer based on intrinsic subtypes. *J. Clin. Oncol.*, 27(8):1160–1167, Mar 2009.

[189] Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. Cross-Sentence N-ary Relation Extraction with Graph LSTMs. *TACL*, 5:101–115, 2017.

[190] Roger D Peng. Reproducible research in computational science. *Science*, 334(6060):1226–1227, 2011.

[191] Stephen Piccolo, Adam Lee, and Michael Frampton. Tools and techniques for computational reproducibility. *bioRxiv*, page 022707, 2015.

[192] Stephen Piccolo, Michelle Withers, Owen Francis, Andrea Bild, and Evan Johnson. Multiplatform single-sample estimates of transcriptional activation. *PNAS*, 2013.

[193] Joao Felipe Pimentel, Leonardo Murta, Vanessa Braganholo, and Juliana Freire. noWorkflow: a tool for collecting, analyzing, and managing provenance from python scripts. *Proceedings of the VLDB Endowment*, 10(12):1841–1844, 2017.

[194] Heather Piwowar and Todd Vision. Data reuse and the open data citation advantage. *PeerJ*, 2013.

[195] Hoifung Poon and Pedro Domingos. Unsupervised semantic parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 1–10. Association for Computational Linguistics, 2009.

[196] Hoifung Poon, Chris Quirk, Charlie DeZiel, and David Heckerman. Literome: PubMed-Scale Genomic Knowledge Base in the Cloud. *Bioinformatics*, 2014.

[197] Teresa M Przytycka, Mona Singh, and Donna K Slonim. Toward the dynamic interactome: it's about time. *Briefings in bioinformatics*, 11(1):15–29, January 2010.

[198] M. A. Pujana, J. D. Han, L. M. Starita, K. N. Stevens, M. Tewari, J. S. Ahn, G. Rennert, V. Moreno, T. Kirchhoff, B. Gold, V. Assmann, W. M. Elshamy, J. F. Rual, D. Levine, L. S. Rozek, R. S. Gelman, K. C. Gunsalus, R. A. Greenberg, B. Sobhian, N. Bertin, K. Venkatesan, N. Ayivi-Guedehoussou, X. Sole, P. Hernandez, C. Lazaro, K. L. Nathanson, B. L. Weber, M. E. Cusick, D. E. Hill, K. Offit, D. M. Livingston, S. B. Gruber, J. D. Parvin, and M. Vidal. Network modeling links breast cancer susceptibility and centrosome dysfunction. *Nat. Genet.*, 39(11):1338–1349, Nov 2007.

[199] Aaron R Quinlan and Ira M Hall. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics (Oxford, England)*, 26(6):841–2, March 2010.

[200] Chris Quirk and Hoifung Poon. Distant Supervision for Relation Extraction beyond the Sentence Boundary. *EACL-2017*, 2017.

[201] Erhard Rahm and Philip A Bernstein. A survey of approaches to automatic schema matching. *the VLDB Journal*, 10(4):334–350, 2001.

[202] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y Ng. Self-taught learning: transfer learning from unlabeled data. In *ICML*, 2007.

[203] Adaikalavan Ramasamy, Adrian Mondry, Chris C Holmes, and Douglas G Altman. Key issues in conducting a meta-analysis of gene expression microarray datasets. *PLoS medicine*, 5(9):e184, 2008.

[204] Noa Rappaport, Noam Nativ, Gil Stelzer, Michal Twik, Yaron Guan-Golan, Tsippi Iny Stein, Iris Bahir, Frida Belinky, C Paul Morrey, Marilyn Safran, and Doron Lancet. MalaCards: an integrated compendium for diseases and their annotation. *Database : the journal of biological databases and curation*, 2013:bat018, January 2013.

[205] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data Programming: Creating Large Training Sets, Quickly. In *NIPS*, 2016.

[206] Christopher Ré, Amir Abbas Sadeghian, Zifei Shan, Jaeho Shin, Feiran Wang, Sen Wu, and Ce Zhang. Feature engineering for knowledge base construction. *arXiv preprint arXiv:1407.6439*, 2014.

[207] Michael Rubacha, Anil K Rattan, and Stephen C Hosselet. A review of electronic laboratory notebooks available in the market today. *JALA: Journal of the Association for Laboratory Automation*, 16(1):90–98, 2011.

[208] FG Rücker, AC Russ, S Cocciardi, H Kett, RF Schlenk, U Botzenhardt, C Langer, J Krauter, S Fröhling, B Schlegelberger, et al. Altered miRNA and gene expression in acute myeloid leukemia with complex karyotype identify networks of prognostic relevance. *Leukemia*, 27(2):353–361, 2013.

[209] Johan Rung and Alvis Brazma. Reuse of public genome-wide gene expression data. *Nat Rev Genet*, 14(2):89–99, Feb 2013.

[210] Albin Sandelin, Wynand Alkema, Pr Engstrm, Wyeth W. Wasserman, and Boris Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*, 32(suppl 1):D91–D94, 2004.

[211] Sylvain Sardy. On the Practice of Rescaling Covariates. *International Statistical Review*, 76(2):285–297, 2008.

[212] Patrick R Schmid, Nathan P Palmer, Isaac S Kohane, and Bonnie Berger. Making sense out of massive data by going beyond differential expression. *PNAS*, 109(15):5594–5599, 2012.

[213] Conrad L Schoch, M Catherine Aime, Wilhelm de Beer, Pedro W Crous, Kevin D Hyde, Lyubomir Penev, Keith A Seifert, Marc Stadler, Ning Zhang, and Andrew N Miller. Using standard keywords in publications to facilitate updates of new fungal taxonomic names. *IMA Fungus*, 8(2):70–73, 2017.

[214] Benjamin Schuster-Böckler and Ben Lehner. Chromatin organization is a major influence on regional mutation rates in human cancer cells. *Nature*, 488(7412):504–7, August 2012.

[215] Eran Segal, Michael Shapira, Aviv Regev, Dana Pe'er, David Botstein, Daphne Koller, and Nir Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature genetics*, 34(2):166–76, June 2003.

[216] T. Senga, T. Iwamoto, S. E. Humphrey, T. Yokota, E. J. Taparowsky, and M. Hamaguchi. Stat3-dependent induction of BATF in M1 mouse myeloid leukemia cells. *Oncogene*, 21(53):8186–8191, Nov 2002.

[217] Manu Setty, Karim Helmy, Aly A. Khan, Joachim Silber, Aaron Arvey, Frank Neezen, Phaedra Agius, Jason T. Huse, Eric C. Holland, and Christina S. Leslie. Inferring transcriptional and microRNA-mediated regulatory programs in glioblastoma. *Mol Syst Biol*, 8:605–605, Aug 2012. 22929615[pmid].

[218] Y. M. Shiao, Y. H. Chang, Y. M. Liu, J. C. Li, J. S. Su, K. J. Liu, Y. F. Liu, M. W. Lin, and S. F. Tsai. Dysregulation of GIMAP genes in non-small cell lung cancer. *Lung Cancer*, 62(3):287–294, Dec 2008.

[219] Jaeho Shin, Sen Wu, Feiran Wang, Christopher De Sa, Ce Zhang, and Christopher Ré. Incremental knowledge base construction using deepdive. *Proceedings of the VLDB Endowment*, 8(11):1310–1321, 2015.

[220] Chong Shou, Nitin Bhardwaj, Hugo Y K Lam, Koon-Kiu Yan, Philip M Kim, Michael Snyder, and Mark B Gerstein. Measuring the evolutionary rewiring of biological networks. *PLoS computational biology*, 7(1):e1001050, January 2011.

[221] Radha Shyamsundar, Young H Kim, John P Higgins, Kelli Montgomery, Michelle Jorden, Anand Sethuraman, Matt van de Rijn, David Botstein, Patrick O Brown, and Jonathan R Pollack. A DNA microarray survey of gene expression in normal human tissues. *Genome biology*, 6(3):R22, 2005.

[222] S. Siehler. Regulation of RhoGEF proteins by G12/13-coupled receptors. *Br. J. Pharmacol.*, 158(1):41–49, Sep 2009.

[223] George Davey Smith and Shah Ebrahim. Data dredging, bias, or confounding: They can all get you into the BMJ and the Friday papers. *BMJ: British Medical Journal*, 325(7378):1437, 2002.

[224] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, 2013.

[225] Scott Spangler, Angela D Wilkins, Benjamin J Bachman, Meena Nagarajan, Tajhal Dayaram, Peter Haas, Sam Regenbogen, Curtis R Pickering, Austin Comer, Jeffrey N Myers, et al. Automated hypothesis generation based on mining scientific literature. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1877–1886. ACM, 2014.

[226] Philip J Stephens, Patrick S Tarpey, Helen Davies, Peter Van Loo, Chris Greenman, David C Wedge, Serena Nik-Zainal, Sancha Martin, Ignacio Varela, Graham R Bignell, Lucy R Yates, Elli Papaemmanuil, David Beare, Adam Butler, Angela Cheverton, John Gamble, Jonathan Hinton, Mingming Jia, Alagu Jayakumar, David Jones, Calli Latimer, King Wai Lau, Stuart McLaren, David J McBride, Andrew Menzies, Laura Mudie, Keiran Raine, Roland Rad, Michael Spencer Chapman, Jon Teague, Douglas Easton, Anita Langerø d, Ming Ta Michael Lee, Chen-Yang Shen, Benita Tan Kiat Tee, Bernice Wong Huimin, Annegien Broeks, Ana Cristina Vargas, Gulisa Turashvili, John Martens, Aquila Fatima, Penelope Miron, Suet-Feung Chin, Gilles Thomas, Sandrine Boyault, Odette Mariani, Sunil R Lakhani, Marc van de Vijver, Laura van 't Veer, John Foekens, Christine Desmedt, Christos Sotiriou, Andrew Tutt, Carlos Caldas, Jorge S Reis-Filho, Samuel A J R Aparicio, Anne Vincent Salomon, Anne-Lise Bø rresen Dale, Andrea L Richardson, Peter J Campbell, P Andrew Futreal, and Michael R Stratton. The landscape of cancer genes and mutational processes in breast cancer. *Nature*, 486(7403):400–4, June 2012.

[227] Michael Stonebraker, Daniel Bruckner, Ihab F Ilyas, George Beskales, Mitch Cherniack, Stanley B Zdonik, Alexander Pagan, and Shan Xu. Data Curation at Scale: The Data Tamer System. In *CIDR*, 2013.

[228] Hannah Stower. Gene expression: Super enhancers. *Nature reviews. Genetics*, 14(6):367, June 2013.

[229] 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, and Jill P Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–50, October 2005.

[230] Jarred E Swalwell, Francois Ribalet, and E Armbrust. SeaFlow: A novel underway flow-cytometer for continuous observations of phytoplankton in the ocean. *Limnology and Oceanography: Methods*, 9(10):466–477, 2011.

[231] S. M. Tanner, J. L. Austin, G. Leone, L. J. Rush, C. Plass, K. Heinonen, K. Mrozek, H. Sill, S. Knuutila, J. E. Kolitz, K. J. Archer, M. A. Caligiuri, C. D. Bloomfield, and A. de La Chapelle. BAALC, the human member of a novel mammalian neuroectoderm gene lineage, is implicated in hematopoiesis and acute leukemia. *Proc. Natl. Acad. Sci. U.S.A.*, 98(24):13901–13906, Nov 2001.

[232] Charles Lewis Taylor. Inter-university Consortium for Political and Social Research. 1975.

[233] Terry M Therneau and Patricia M Grambsch. *Modeling Survival Data: Extending the Cox Model*. 2000.

[234] Robert E Thurman, Eric Rynes, Richard Humbert, Jeff Vierstra, Matthew T Maurano, Eric Haugen, Nathan C Sheffield, Andrew B Stergachis, Hao Wang, Benjamin Vernot, Kavita Garg, Sam John, Richard Sandstrom, Daniel Bates, Lisa Boatman, Theresa K Canfield, Morgan Diegel, Douglas Dunn, Abigail K Ebersol, Tristan Frum, Erika Giste, Audra K Johnson, Ericka M Johnson, Tanya Kutyavin, Bryan Lajoie, Bum-Kyu Lee, Kristen Lee, Darin London, Dimitra Lotakis, Shane Neph, Fidencio Neri, Eric D Nguyen, Hongzhu Qu, Alex P Reynolds, Vaughn Roach, Alexias Safi, Minerva E Sanchez, Amartya Sanyal, Anthony Shafer, Jeremy M Simon, Lingyun Song, Shinny Vong, Molly Weaver, Yongqi Yan, Zhancheng Zhang, Zhuzhu Zhang, Boris Lenhard, Muneesh Tewari, Michael O Dorschner, R Scott Hansen, Patrick A Navas, George Stamatoyannopoulos, Vishwanath R Iyer, Jason D Lieb, Shamil R Sunyaev, Joshua M Akey, Peter J Sabo, Rajinder Kaul, Terrence S Furey, Job Dekker, Gregory E Crawford, and John A Stamatoyannopoulos. The accessible chromatin landscape of the human genome. *Nature*, 489(7414):75–82, September 2012.

[235] R Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[236] Robert Tibshirani, Jacob Bien, Jerome Friedman, Trevor Hastie, Noah Simon, Jonathan Taylor, and Ryan J. Tibshirani. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):245–266, 2012.

[237] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *Coursera: NN for ML*, 2012.

[238] Aurora Torrente, Margus Lukk, et al. Identification of Cancer Related Genes Using a Comprehensive Map of Human Gene Expression. *PLOS ONE*, 2016.

[239] Gouji Toyokawa, Ken Masuda, Yataro Daigo, Hyun-Soo Cho, Masanori Yoshimatsu, Masashi Takawa, Shinya Hayami, Kazuhiro Maejima, Makoto Chino, Helen I. Field, David E. Neal, Eiju Tsuchiya, Bruce AJ Ponder, Yoshihiko Maehara, Yusuke Nakamura, and Ryuji Hamamoto. Minichromosome Maintenance Protein 7 is a potential therapeutic target in human cancer and a novel prognostic marker of non-small cell lung cancer. *Molecular Cancer*, 10(1):1–11, 2011.

[240] George C Tseng, Debashis Ghosh, and Eleanor Feingold. Comprehensive literature review and statistical considerations for microarray meta-analysis. *Nucleic acids research*, 40(9):3785–3799, 2012.

[241] Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

[242] V G Tusher, R Tibshirani, and G Chu. Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences of the United States of America*, 98(9):5116–21, April 2001.

[243] Marc J van de Vijver, Yudong D He, Laura J van't Veer, Hongyue Dai, Augustinus A M Hart, Dorien W Voskuil, George J Schreiber, Johannes L Peterse, Chris Roberts, Matthew J Marton, Mark Parrish, Douwe Atsma, Anke Witteveen, Annuska Glas, Leonie Delahaye, Tony van der Velde, Harry Bartelink, Sjoerd Rodenhuis, Emiel T Rutgers, Stephen H Friend, and René Bernards. A gene-expression signature as a predictor of survival in breast cancer. *The New England journal of medicine*, 347(25):1999–2009, December 2002.

[244] L. J. van 't Veer, H. Dai, M. J. van de Vijver, Y. D. He, A. A. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, and S. H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–536, Jan 2002.

[245] Fabio Vandin, Eli Upfal, and Benjamin J Raphael. De novo discovery of mutated driver pathways in cancer. *Genome research*, 22(2):375–85, February 2012.

[246] S. Varambally, S. M. Dhanasekaran, M. Zhou, T. R. Barrette, C. Kumar-Sinha, M. G. Sanda, D. Ghosh, K. J. Pienta, R. G. Sewalt, A. P. Otte, M. A. Rubin, and A. M. Chinnaiyan. The polycomb group protein EZH2 is involved in progression of prostate cancer. *Nature*, 419(6907):624–629, Oct 2002.

[247] Karin Verspoor, Judith Cohn, Susan Mniszewski, and Cliff Joslyn. A categorization approach to automated ontological function annotation. *Prot. Sc.*, 2006.

[248] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *ICML*, 2008.

[249] E. Vire, C. Brenner, R. Deplus, L. Blanchon, M. Fraga, C. Didelot, L. Morey, A. Van Eynde, D. Bernard, J. M. Vanderwinden, M. Bollen, M. Esteller, L. Di Croce, Y. de Launoit, and F. Fuks. The Polycomb group protein EZH2 directly controls DNA methylation. *Nature*, 439(7078):871–874, Feb 2006.

[250] B. Vogelstein, N. Papadopoulos, V. E. Velculescu, S. Zhou, L. A. Diaz, and K. W. Kinzler. Cancer genome landscapes. *Science*, 339(6127):1546–1558, Mar 2013.

[251] Martin J. Wainwright, John D. Lafferty, and Pradeep K. Ravikumar. High-Dimensional Graphical Model Selection Using L1-Regularized Logistic Regression. In *Advances in Neural Information Processing Systems*, pages 1465–1472, 2006.

[252] Kai Wang, Manikandan Narayanan, Hua Zhong, Martin Tompa, Eric E Schadt, and Jun Zhu. Meta-analysis of inter-species liver co-expression networks elucidates traits associated with common human diseases. *PLoS computational biology*, 5(12):e1000616, December 2009.

[253] Xujing Wang, Soumitra Ghosh, and Sun-Wei Guo. Quantitative quality control in microarray image processing and data acquisition. *Nucleic Acids Research*, 29(15):e75–e75, 2001.

[254] James West, Ginestra Bianconi, Simone Severini, and Andrew E Teschendorff. Differential network entropy reveals cancer system hallmarks. *Scientific reports*, 2:802, January 2012.

[255] Warren A Whyte, David A Orlando, Denes Hnisz, Brian J Abraham, Charles Y Lin, Michael H Kagey, Peter B Rahl, Tong Ihn Lee, and Richard A Young. Master transcription factors and mediator establish super-enhancers at key cell identity genes. *Cell*, 153(2):307–19, April 2013.

[256] Ami Wiesel and Alfred O Hero. Distributed covariance estimation in Gaussian graphical models. *Signal Processing, IEEE Transactions on*, 60(1):211–220, 2012.

[257] Kyoung-Jae Won, Xian Zhang, Tao Wang, Bo Ding, Debasish Raha, Michael Snyder, Bing Ren, and Wei Wang. Comparative annotation of functional regions in the human genome using epigenomic data. *Nucleic acids research*, 41(8):4423–32, April 2013.

[258] H. T. Xu, L. Wang, D. Lin, Y. Liu, N. Liu, X. M. Yuan, and E. H. Wang. Abnormal beta-catenin and reduced axin expression are associated with poor differentiation and progression in non-small cell lung cancer. *Am. J. Clin. Pathol.*, 125(4):534–541, Apr 2006.

[259] M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(10):19–35, 2007.

[260] B-H Zhang, J Liu, Q-X Zhou, D Zuo, and Y Wang. Analysis of differentially expressed genes in ductal carcinoma with DNA microarray. *European review for medical and pharmacological sciences*, 17(6):758–66, March 2013.

[261] Bai Zhang, Huai Li, Rebecca B Riggins, Ming Zhan, Jianhua Xuan, Zhen Zhang, Eric P Hoffman, Robert Clarke, and Yue Wang. Differential dependency network analysis to identify condition-specific topological changes in biological networks. *Bioinformatics (Oxford, England)*, 25(4):526–32, February 2009.

[262] Tuo Zhao, Han Liu, Kathryn Roeder, John Lafferty, and Larry Wasserman. The Huge Package for High-dimensional Undirected Graph Estimation in R. *J. Mach. Learn. Res.*, 13:1059–1062, April 2012.

[263] Zheguang Zhao, Lorenzo De Stefani, Emanuel Zgraggen, Carsten Binnig, Eli Upfal, and Tim Kraska. Controlling false discoveries during interactive data exploration. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 527–540. ACM, 2017.

[264] Xianmin Zhu, Shaad M Ahmad, Anton Aboukhalil, Brian W Busser, Yongsok Kim, Terese R Tansey, Adrian Haimovich, Neal Jeffries, Martha L Bulyk, and Alan M Michelson. Differential regulation of mesodermal gene expression by Drosophila cell type-specific Forkhead transcription factors. *Development (Cambridge, England)*, 139(8):1457–66, April 2012.

[265] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. *Technical Report CMU-CALD-02-107*, 2002.

[266] Yuelin Zhu, Sean Davis, Robert Stephens, Paul S. Meltzer, and Yidong Chen. GEOmetadb: powerful alternative search engine for the Gene Expression Omnibus. *Bioinformatics*, 24(23):2798–2800, 2008.

[267] Yuelin Zhu, Robert M. Stephens, Paul S. Meltzer, and Sean R. Davis. SRAdb: query and use public next-generation sequencing data from within R. *BMC Bioinformatics*, 14(1):19, 2013.