Using Multi-task Learning to Predict Signaling and Regulatory Pathways

Venkata Krishna Pillutla
Rohan Varma
Petar Stojanov

1 Introduction

Various laboratory and computational techniques have been used to link genes to disease and resistance to therapy. With the explosion of high-throughput biological data, analyses of GWAS, DNA and RNA (expression) sequencing data, copy number data, clinical data as well as functional experiments have all played a significant role in the discovery of genes with a potential functional role in diseases such as cancer and infectious and genetic diseases (15, 6, 12, 16). Because of the complexity of these diseases and the unpredictable effect of treatment, understanding how these potential driver genes interact to regulate the cell under malignant and therapeutic conditions is instrumental in developing better clinical approaches in the future. With the increasing availability of publicly available gene expression and clinical data, there is a growing need of developing methods that will use this information to explore potentially clinically relevant mechanisms.

Reconstruction of protein networks and inferring their signaling and regulatory pathways from biological high-throughput data is a large problem space, and a series of methods have been described that address various aspects of it. (11, 13, 5, 4, 8). Some methods such as (4) aim at using gene expression data and statistical relationships between genes' expression profiels to reconstruct intercation networks for complex species. While having a concise network provides a broad idea about the neighborhood of a protein, it does not provide information about specific pathways that are targeted under conditions such as drug perturbations or other external stress.

DREM and SDREM (11, 13, 8) predict transcriptional and regulatory pathways by integrating time-series gene expression data and static protein-protein interaction data. The aim of these methods is to discover pathways that represent the cell's response to a disease or a drug-specific perturbation and thus nominate members of this network which could be its targets. SDREM was shown to do this successfully for H1N1 and H5N1 strains of flu (10). However, SDREM and other methods such as ResponseNET (5) work with a single protein-protein interaction network per each condition, thus not taking advantage of possible similarities between different conditions. For example, for many drugs in the clinic and the laboratory the target molecules are well established, so it is fairly safe to assume that the pathway that the drug triggers is similar across different experiments on different cells from the same lineage. SDREM and DREM are therefore more suitable when this type of drug information is not available and we are not working with potentially closely related conditions. ResponseNET uses a flow algorithm formulated as a linear program to find pathways between genetic hits from external cellular stress to transcription factors and differentially expressed genes. However, this framework is difficult to adapt to new problem settings because imposing any specific constraints on the network structure has to be represented as constraints in the linear program which could make the algorithm too computationally intensive.

With the growth of publicly available biological data, the availability of experiments of related tissue types or perturbation agents has increased significantly. For example, the The Cancer Genome Atlas (https://tcga-data.nci.nih.gov/tcga/dataAccessMatrix.htm) now has sequenced the RNA of many patients with matched clinical data. Furthermore, even more of this information is publicly available

because of the large amount of work published in the field of cancer genome analysis. This creates a need for a method that can use this data to search for drug-related regulatory pathways, using some of its potential advantages such as multiple patients from the same tumor type being treated with the same or similar medication.

Recent biotechnological breakthroughs have enabled the production of a large and powerful dataset LINCS funded by the NIH (http://www.lincscloud.org/, citation not yet available), which contains the gene expression data of a selected 1000 genes, in 77 cell types, across up to 45000 drug and genetic perturbations (such as knock-out) agents. Most of the cell types are cancer cells, some of which are well studied cancer cell lines (such as MCF7, VCAP and PC3). Since multiple cell types have been treated with the same drug, the assumption is that these cell types will respond to the treatment by activating/supressing similar pathways. While working with the above-mentioned TCGA data above is a major long-term project, the LINCS dataset presents an opportunity to develop a method that augments the learning process by sharing information between cell types when modeling the networks. Furthermore, it is a chance to model networks as responses to cancer therapy and hopefully nominate unique and common gene members that are responsible for drug resistance. In this project, we will focus on using multitask-learning techniques to regulatory and signaling networks in cancer and analyze their performance.

A new method also from Carnegie Mellon University (MT-SDREM) (in press) builds on SDREM and DREM to adapt them to using multi-task learning (17) to share parameters between three related flu strains and thus take advantage of the fact that these strains trigger common pathways. Namely, the priors for the transcription factors that are passed to DREM are calculated jointly for all three strains. This method shows successful application of multi-task learning to a pathogenic case, and it would be useful to extend its features to: 1. Work with static expression profiles from many experiments and 2. Redefine the multi-task learning target function to one that deals with multiple cell types being treated with the same drug.

2 Problem Representation and data

Our multitask learning problem aims to find a set of pathways from sources to targets from the large PPI network, where for each drug experiment, a task represents a different cell line from the same tumor type. As mentioned in the previous section, for MT-SDREM (where the tasks are different flu strains), the parameter sharing happens at the regulatory level because it only affects transcription factors. However, it could be that many transcription factors regulate multiple different pathways that may not all participate in drug response. In our redefinition of the multi-task problem, one of the main features is encouraging sharing nodes between the graphs in our target function. and introduce a constraint for this in the target function.

2.1 Network Representation

Similarly to MT-SDREM, we will integrate the LINCS data with static perturbation-independent protein-protein interaction database by setting up a network as follows:

1. For each cell type (or condition C) we represent a signaling and regulatory network as a graph, where the nodes are the proteins (genes) and the edges are protein-protein (PPI) and protein-DNA interactions. Initially we will not use directed edges since assigning direction to protein interactions may be beyond the scope of this project. This graph will have candidate sources S, candidate targets T. The sources are the proteins that the drug interacts with on the cell surface, and the targets are the proteins whose concentration in the cell is affected as a response to the drug (approximated by gene expression). The goal is to find one or more traversals from sources to targets using the least amount of nodes, and nodes that are common across multiple similar cell types.

2. We initialize the signaling component of the graph with a static interaction dataset I_c which we assume is a superset of the current cell's protein-protein interactions, by combining several large public datasets (Biogrid (9), String (7), ENCODE (2) etc.). We identify the sources S with which the drug interacts on the surface of the cell. We will do this by correlating the expression profile of the drug treatment on a cell type with the expression profile of the knock-out experiments, using Pearson

correlation or mutual infromation. A complete correlation or anti-correlation will indicate that the gene which was subject to the KO experiment is a candidate source of the graph that the drug targets.

3. A significant difference in the framework of MT-SDREM and our current formulation is that for MT-SDREM the transcription factor is the end-point of the drug's response whereas in our case the targets are genes whose regulation changed subject to the drug, the signaling pathway and the transcription factor. In this setup, we have a way of identifying targets regulated by relevant transcription factors directly from the expression data. For each transcription factor knock-out that was performed with LINCS, we construct edges between the transcription factor and the differentially expressed genes that resulted in its KO experiment. These edges represent protein-DNA interactions and will be the initial state of the regulatory component of the network, and the top differentially expressed genes will be the candidate regulatory targets T. In our formulation we can have unexplained targets, and we are looking for pathways that explain as many of the differentially expressed targets as possible.

2.2 Data

We downloaded the Broad Institute LINCS level4 gene expression dataset from the LINCS cloud (http://apps.lincscloud.org/). From initial inspection we concluded that that various cell lines have different number of perturbations performed. We decided to work with the same tumor type (prostate cancer), and we identified two cancer cell lines, VCAP and PC3, for which there are 4000 knockout experiments. The level 4 gene expression data represents the differential expression of the 1000 hallmark genes captures in a Z score for each gene and each experiment.
Although we are trying to infer the sources (the molecules that the drug directly interacts with) from the data, we picked experiments for drugs that have been tested on these cell lines before or are used in the clinic to treat prostate cancer (disulfiram, docetaxel, ketoconazole, vinblastine, doxorubicin, metformin, parthenolide, bicalutamide). For these drugs the sources are known and they can be either single proteins (such as androgen receptor AR for bicalutamide) or protein families (such as aldehyde dehydrogenase for disulfiram and tubulin proteins for docetaxel). We inferred the sources as described in (2) of the previous subsection, and we evaluated thisprocedure by matching against known drug targets. The true target was among the true sources for only three of the nine drugs that we tested. However, because there are many unknown side-effects of drug treatment we decided to work with the targets we inferred and see if we obtain biologically meaningful pathways with this approach.

3 Multi-task Algorithm

We represent each cell type (in our case VCAP and PC3) as separate tasks for each drug experiment. Here we describe the multi-task learning that aims to address the following key features: 1. Node-sharing between conditions (networks) - because of the assumption that the same drug affects similar pathways in the two cell types. 2. Targets are differentially expressed genes and we need to penalize unexplained targets, as well as transcription factors which do not specifically regulate targets of the drug in question.
In this we define an objective function for this multitask problem. To solve it, we first use BFS to find k paths between each source-target pair, and then we use a greedy method (described below) to search these paths and evaluate the objective function.

3.1 Notation

- C: set of all conditions - in our case the two different cell lines for a particular drug experiment
- T_c: set of targets of a condition $c \in C$
- P_c^t: set of paths connecting $c \in C$ to target $t \in T_c$; p will refer to any path in the network
- $h(p)$: cost of a path defined as probability of a path, i.e. product of probabilities of edges in the path
- S: subgraph of the network chosen by the algorithm

- $I_S(p)$: 1 if $p \in S$ and 0 otherwise, i.e. $I(p \in S)$
- $n(p_1, p_2)$: number of nodes common to paths p_1, p_2
- $N(S)$: total number of nodes present in the all paths contained in S, with each node counted only once
- tf: a transcription factor
- $\mathcal{T}_c$: set of TFs of condition $c \in C$
- $\mathcal{P}_c^{tf}$: set of paths connecting $c \in C$ to $tf \in \mathcal{T}$
- $TF(S)$: set of transcription factors in the network induced by S
- $T(tf)$: set of targets attached to transcription factor tf
- α is a parameter deciding how important it is for paths to have common nodes: to be decided by cross-validation
- $a \rightarrow b$ denotes an edge from a to b

3.2 Objective function

$$\begin{aligned}
\max_S \quad & \sum_{c\in C}\sum_{t\in T_c}\sum_{p\in P_c^t} I_S(p)h(p) \\
& +\lambda_1 \sum_{c_1\in C}\sum_{t_1\in T_{c_1}}\sum_{p_1\in P_{c_1}^{t_1}}\sum_{c_2\in C}\sum_{t_2\in T_{c_2}}\sum_{p_2\in P_{c_2}^{t_2}} n(p_1,p_2)^\alpha I_S(p_1)I_S(p_2) \\
-\lambda_2 N(S) \quad & +\lambda_3 \sum_{c\in C}\sum_{t\in T_c} I\Big(\sum_{p\in P_c^t} I_S(p) > 0\Big) \\
& +\lambda_4 \sum_{tf\in TF(S)} \frac{\sum_{t\in T(tf)} I(t\in S)}{|T(tf)|}
\end{aligned}$$

The first term is to ensure we select smaller, better paths, since $h(p)$ lies between 0 and 1. The second term encourages similarity across tasks. The third term is to penalize a large number of nodes in the induced network, and the fourth term (λ_3 term) is to encourage explanation of all targets. The λ_4 term is to penalize targets that are attached to a TF but are not required to be explained (we would like to impose a constraint that an active TF activates all the targets it is attached to and this is a soft way of doing it). To put differently, a TF that explains n targets is better than a TF that explains n targets but also has other connections that are not targets.

We can simplify the objective as follows:

$$\begin{aligned}
\max_S \quad \sum_{p\in S} h(p) \quad & +\lambda_1 \sum_{p_1\in S}\sum_{p_2\in S} n(p_1,p_2)^\alpha - \lambda_2 N(S) + \lambda_3 \sum_{c\in C}\sum_{t\in T_c} I(|S\cap P_c^t| > 0) \\
& +\lambda_4 \sum_{tf\in TF(S)} \frac{\sum_{t\in T(tf)} I(t\in S)}{|T(tf)|}
\end{aligned}$$

In the following, we denote the objective as $f(S)$.

3.3 Algorithm

It is known in biology that one TF may regulate several targets. So, there exist many more source-target paths than the number of source-TF paths. Consequently, searching in the space of paths from the sources to TFs and then looking at all targets attached to the TFs will be benefical. This step is also biologically motivated by the fact that TFs bind to specfic DNA sequences.

The overall algorithm is given in algorithm 3.3 We trade-off rigour for simplicity in the description in algorithm 3.3, our greedy procedure. The first step of algorithm 3.3 finds k best paths using a BFS with a limited queue (for reasons of efficiency).

Algorithm 1 Overall Algorithm

1: Search Space: For every (c, tf), find k best paths from the network with BFS
2: Search Procedure: Algorithm 3.3

Algorithm 2 Greedy Algorithm

Input: k paths for each (c, tf) pair, ordering τ of (c, tf) pairs
Output: set of paths, S

```
S = φ
for (c, tf) in ordering τ do
    while S changes do
        Find best path p_1 from c to tf to add to S
        Find best path p_2 ∈ S to remove from S
        Add(p_1, S), Remove(p_2, S), or leave S unchanged, whichever leads to the highest objective function
    end while
end for
return S
```

3.3.1 Picking an ordering

The order in which (c, tf) pairs are traversed is important in determining the quality of the optimum. We describe three schemes, and the pros and cons of each.

- Random ordering: This is the easiest to implement but several (c, tf) pair may have no paths. Hence, even considering such a pair is extra work.
- Use a heuristic such as average probability of all paths from c to tf. This method overcomes the problem with random orders. But it is deterministic and the best we can do is the local optimum from this ordering.
- Importance sampling using the above heuristic value as weights. This method prefers shorter paths because shorter paths have larger probability. Hence, we use importance sampling, but with a heuristic normalised for length. Recall that the probability of a path is $h(p) = \prod_{e \in p} h(e)$. The heuristic we use is $\exp(-\frac{1}{|p|}(\sum_{e \in p} - \log(h(e))))$ which, in other words, $(\prod_{e \in p} h(e))^{\frac{1}{|p|}}$

3.4 Analysis of Algorithm 3.3

Convergence

Algorithm 3.3 converges because in each iteration of the for loop, there is a strict increase in the objective function value. Since the objective cannot increase in an unbounded manner, the iterations converge.

Complexity

Algorithm 3.3 saves work over the brute-force, exponential algorithm in two places:

1. Step 1 of the algorithm: Instead of looking at all possible paths, we look at the k best paths.
2. Algorithm 3.3: The brute-force algorithm would have to look at all possible $n_{c,tf}!$ orderings where $n_{c,tf}$ is the number of (c, tf) pairs. Instead, we use a sampling based procedure to fix and ordering.
3. Each iterations of the for loop of Algorithm 3.3 does not look at all possible subsets, but instead tries to construct a local optimum by adding or removing one set at a time.

3.5 Comparison to existing work

MT-SDREM (in press) that builds on DREM and SDREM using multi-task procedures is similar to the proposed method in certain ways but is different is several. MT-SDREM finds paths using a BFS, and this is where the similarity ends. MT-SDREM tries to greedily orient edges whose direction has not been specified. We leave undirected edges as is- we try to find a subgraph of paths to encourage overlap between selected paths.

3.6 Discussion: Similarity to Stochastic Coordinate Ascent

In more ways than one, our algorithm looks similar to Stochastic Coordinate Ascent (14), Algorithm 3.6. SCA picks a coordinate with some chosen probability. In algorithm 3.3, we pick a (c, tf) pair. In SCA, a one-dimensional sub-problem is solved in the selected coordinate. In our setting, analogously, we look to add or delete paths from the selected (c, tf) pair. Our discrete analogue of the step length for SCA, $1/L_i$ is the number of steps, i.e., the number of times the inner while loop runs.

This is a very powerful observation because we can use tricks in literature about SCA to our method, such as order to consider vertices in, etc.

Algorithm 3 Random Coordinate Descent [Nesterov]

Input: $x_0 \in R^n$ //starting point
Output: x

1: set $x = x_0$
2: **for** $k = 1, ...$ **do**
3: choose coordinate $i \in \{1, 2, \ldots, n\}$ w.p. p_i
4: update $x^{(i)} = x^{(i)} - \frac{1}{L_i} \nabla f_i(x)$
5: **end for**
6: **return** x

4 Results

Molecular and protein pathways are generally very difficult to validate, especially outside the wet lab. One simple approach that does not involve biological experiments is matching against a set of gene-sets that have been curated based on experimental information from previous literature. One such set is Gene Ontology (1), which we downloaded from the MSIGDB website (3). These 1400 gene-sets contain various types of biological pathways some of which are cancer-related. We define a validation metric a set of significant q values (< 0.1), derived from p values: $p = 1 - H(k - 1, K, N - K, n)$ where H is the hypergeometric CDF, k is the number of overlapping genes between a discovered pathway and a GO geneset, K is the number of genes in the GO geneset, N is number of all known genes (20000 in the human genome), and n is the total number of genes in the discovered pathway. This gives some form of statistical evaluation of overlap of genesets. In order to create a performance metric for our method, we did the following:

1. For each drug, took the pathways discovered and collapsed them to a unique set of genes.

2. For each of the 1400 GO gene-sets, we calculated a p value between the collapsed set of discovered genes and a GO gene-set.

3. We used Benjamini-Hochberg multiple hypotheses correction across the 1400 p values to get q values. The number of significant overlaps is then the number of q values that are less than 0.1.

In order to assess the effectiveness of the Multi-Task approach, we ran the algorithm in two modes. In the first mode we ran it as described, with the objective rewarding shared nodes between conditions. For the second mode we turned this off, which eliminated the multi-task aspect. We compared

the percent increase in the number of significant overlaps from single-task to multi-task over five different drugs for five separate runs. Table 1 shows the percent increase for each drug.

Drug:	Drug 1	Drug 2	Drug 3	Drug 4	Drug 5
Avg. % Increase - correlation:	3.4	6.25	0.77	0.6	13
Avg. % Increase - mutual info. :	2.10	2.25	4.52	-0.24	-3.37

Table 1: Average percent increase of significant overlapping genesets in multi vs. single task

From the table, one can appreciate that when we use correlation to infer the sources, on average we observe an increase in number of genesets that our discovered genes overlap with significantly. From this we can loosely interpret that when we encourage node sharing between the conditions, we are more likely to find pathways that are biologically plausible. Figure 1 shows an example of a pathway that we found:

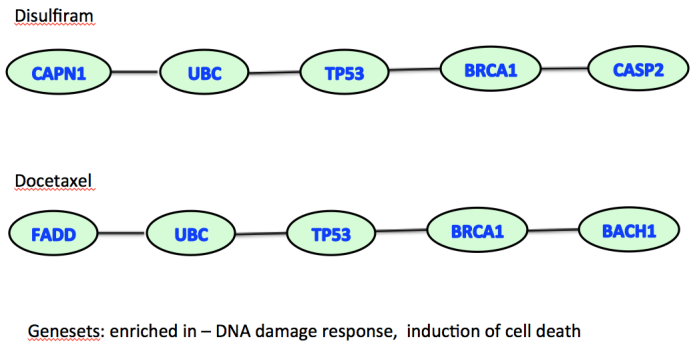

Figure 1: An example of two significant pathways that were found for the first two drugs.

These two pathways share three of their nodes, and the shared genes (UBC, TP53, BRCA1) are well known tumor suppressors. These pathways were also enriched in GO genesets for DNA damage repair and cell death, which are pathways that typically get activated when cell division is targeted, a common effect of these two drugs.

4.1 Cross-validation for Parameter Values

The objective function has five parameters in all that can be adjusted as per requirements. We came up with intial guesses for the parameter values and did a cross-validation on near-by values. The metric used was significant overlap with the GO genesets. 4 of 9 drugs were used for the cross-validation procedure and all the tests were performed on 9 drugs. Table 2 contains some sample output we obtained in deciding on parameter values:

Table 2: Sample Cross-validation output

Drug	α	λ_1	λ_2	λ_3	λ_4	Performance
9	1	0.1	-1	0.1	5	7
9	1	0.1	-1	0.05	5	0
9	1	0.1	-0.5	0.1	5	38
9	1	0.1	-0.5	0.05	5	33
6	1	0.1	-1	0.1	5	9
6	1	0.1	-1	0.05	5	1
6	1	0.1	-0.5	0.1	5	40
6	1	0.1	-0.5	0.05	5	35

5 Discussion

We have showed that multi-task learning works effectively when we infer sources using Pearson correlation, by discovering pathways that have a more significant overlap with previously studied curated gene-sets. One future direction would be combining multiple cell lines and drugs that have similar effect and see if the multitask learning improves further. An important limitation to our framework is that we use a pre-defined protein-protein interaction network that is not derived from cancer. Enhancing this network with models inferred from gene expression may be a useful next step in improving the quality of the pathways that we discover. Furthermore, correlation might not be the best way to infer sources even though we observe improvements in the multi-task algorithm, so exploring better methods for inferring drug interacting partners from expression data may be another useful next step.

References

[1] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[2] ENCODE Project Consortium et al. The encode (encyclopedia of dna elements) project. *Science*, 306(5696):636–640, 2004.

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

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

[5] Esti Yeger-Lotem, Laura Riva, Linhui Julie Su, Aaron D Gitler, Anil G Cashikar, Oliver D King, Pavan K Auluck, Melissa L Geddie, Julie S Valastyan, David R Karger, et al. Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity. *Nature genetics*, 41(3):316–323, 2009.

[6] Ryan D Morin, Maria Mendez-Lago, Andrew J Mungall, Rodrigo Goya, Karen L Mungall, Richard D Corbett, Nathalie A Johnson, Tesa M Severson, Readman Chiu, Matthew Field, et al. Frequent mutation of histone-modifying genes in non-hodgkin lymphoma. *Nature*, 476(7360):298–303, 2011.

[7] Damian Szklarczyk, Andrea Franceschini, Michael Kuhn, Milan Simonovic, Alexander Roth, Pablo Minguez, Tobias Doerks, Manuel Stark, Jean Muller, Peer Bork, et al. The string database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic acids research*, 39(suppl 1):D561–D568, 2011.

[8] Marcel H Schulz, William E Devanny, Anthony Gitter, Shan Zhong, Jason Ernst, and Ziv Bar-Joseph. Drem 2.0: Improved reconstruction of dynamic regulatory networks from time-series expression data. *BMC systems biology*, 6(1):104, 2012.

[9] Andrew Chatr-aryamontri, Bobby-Joe Breitkreutz, Sven Heinicke, Lorrie Boucher, Andrew Winter, Chris Stark, Julie Nixon, Lindsay Ramage, Nadine Kolas, Lara ODonnell, et al. The biogrid interaction database: 2013 update. *Nucleic acids research*, 41(D1):D816–D823, 2013.

[10] Anthony Gitter and Ziv Bar-Joseph. Identifying proteins controlling key disease signaling pathways. *Bioinformatics*, 29(13):i227–i236, 2013.

[11] Anthony Gitter, Miri Carmi, Naama Barkai, and Ziv Bar-Joseph. Linking the signaling cascades and dynamic regulatory networks controlling stress responses. *Genome research*, 23(2):365–376, 2013.

[12] Dan A Landau, Scott L Carter, Petar Stojanov, Aaron McKenna, Kristen Stevenson, Michael S Lawrence, Carrie Sougnez, Chip Stewart, Andrey Sivachenko, Lili Wang, et al. Evolution and impact of subclonal mutations in chronic lymphocytic leukemia. *Cell*, 152(4):714–726, 2013.

[13] Hai-Son Le and Ziv Bar-Joseph. Integrating sequence, expression and interaction data to determine condition-specific mirna regulation. *Bioinformatics*, 29(13):i89–i97, 2013.

[14] Yurii Nesterov. Efficiency of coordinate-descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 2013.

[15] Jacqueline I Goldstein, L Fredrik Jarskog, Chris Hilliard, Ana Alfirevic, Laramie Duncan, Denis Fourches, Hailiang Huang, Monkol Lek, Benjamin M Neale, Stephan Ripke, et al. Clozapine-induced agranulocytosis is associated with rare hla-dqb1 and hla-b alleles. *Nature communications*, 5, 2014.

[16] Michael S Lawrence, Petar Stojanov, Craig H Mermel, James T Robinson, Levi A Garraway, Todd R Golub, Matthew Meyerson, Stacey B Gabriel, Eric S Lander, and Gad Getz. Discovery and saturation analysis of cancer genes across 21 tumour types. *Nature*, 2014.

[17] Rich Caruana. *Multitask learning*. Springer, 1998.