Lecture 13: January 16, 2018

Lecturer: Prof. Roded Sharan *Scribe: David Pellow*

13.1 Predicting drug side-effects

The drug development pipeline is long, very costly and can often end in failure. While more money and data has been put into developing new drugs, the number of approved drugs has not increased. Adverse side-effects during clinical trials are one of the reasons for non-approval of a drug and they are detected late in the development pipeline, making them very costly. Predicting side-effects in advance can greatly benefit the drug development process.

The method developed uses canonical correlation analysis (CCA) to predict drug side-effects from chemical structure [1]. An advantage of using CCA is that it is trained on combined side-effect information and predicts all side-effects together as opposed to other classifiers, for example SVM, which will give a binary prediction for each individual side-effect.

13.1.1 Mathematical introduction: canonical correlation analysis

Intuition

Given two different "views", i.e. two sets of features, into a single object, for example a set of samples, we combine the information in both. We use a projection of both feature sets into a lower-dimensional joint subspace such that the projections are maximally correlated.

Procedure

We have n samples and $p \times n$ data matrix X and $q \times n$ data matrix Y where each column is centered to have a mean of 0. Let $\Sigma_{XX} = XX^\top$, $\Sigma_{YY} = YY^\top$, $\Sigma_{XY} = XY^\top$, $\Sigma_{YX} = YX^\top$ be the covariance and cross-covariance matrices.

The projection matrices $(W_X)_{p\times k}$, $(W_Y)_{q\times k}$ ($k \leq min(p, q)$) for X and Y respectively, are computed as follows:

The columns of W_X are the eigenvectors corresponding to the top k eigenvalues solving the equation: $\Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX}w_X = \lambda^2 w_X$

For easier computation, the columns of W_Y are calculated directly as functions of the columns of W_X: $w_y = \frac{1}{\lambda}\Sigma_{YY}^{-1}\Sigma_{YX}w_X$ for the corresponding w_X and λ

To return a vector u in the lower dimensional space to vectors x', y' in the original spaces use $x' = \Sigma_{XX}W_X u$ and $y' = \Sigma_{YY}W_Y u$.

Derivation

We are looking to maximize the correlation between $W_X^\top X$ and $W_Y^\top Y$. This can be written as:

$$\underset{W_X,W_Y}{arg\,max}\ tr(W_X^\top XY^\top W_Y)\ \text{ s.t. } W_X^\top XX^\top W_X = W_Y^\top YY^\top W_Y = I \tag{13.1}$$

We come to this expression and derive the solution as follows:

Consider projecting X onto one axis using the vector w_X and projecting Y using the vector w_Y. Maximizing the correlation between the projections:

$$\underset{w_X,w_Y}{max}\frac{w_X^\top \Sigma_{XY} w_Y}{\sqrt{w_X^\top \Sigma_{XX} w_X}\sqrt{w_Y^\top \Sigma_{YY} w_Y}} \tag{13.2}$$

We can choose how to normalize w_X, w_Y since if w_X, w_Y maximize eqn. (13.2) then so do aw_X, bw_Y for $a, b > 0$. To simplify things, we choose

$$w_X^\top \Sigma_{XX} w_X = w_Y^\top \Sigma_{YY} w_Y = 1 \tag{13.3}$$

This yields the Lagrangian:

$$\mathcal{L} = w_X^\top \Sigma_{XY} w_Y + \lambda_1(1 - w_X^\top \Sigma_{XX} w_X) + \lambda_2(1 - w_Y^\top \Sigma_{YY} w_Y) \tag{13.4}$$

$$\frac{\partial \mathcal{L}}{\partial w_X} = \Sigma_{XY} w_Y - 2\lambda_1 \Sigma_{XX} w_X \overset{set}{=} 0 \implies -2\lambda_1 w_X = \Sigma_{XX}^{-1}\Sigma_{XY} w_Y \tag{13.5}$$

$$\frac{\partial \mathcal{L}}{\partial w_Y} = \Sigma_{XY}^\top w_X - 2\lambda_2 \Sigma_{YY} w_Y \overset{set}{=} 0 \implies -2\lambda_2 w_Y = \Sigma_{YY}^{-1}\Sigma_{XY}^\top w_X \tag{13.6}$$

Substituting eqn. (13.6) into eqn. (13.5):

$$(4\lambda_1\lambda_2) w_X = \left(\Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{XY}^\top\right) w_X \tag{13.7}$$

We will show that $\lambda_1 = \lambda_2$, therefore, letting $2\lambda_1 = 2\lambda_2 \triangleq -\lambda$, and $M \triangleq \Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{XY}^\top$, eqn. (13.7) is the eigen-equation $\lambda^2 w_X = M w_X$

Rearranging eqn. (13.7) gives $-2\lambda_1 w_X = \frac{-1}{2\lambda_2}\left(\Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{XY}^\top\right) w_X$. Plugging this back into eqn. (13.5) yields:

$$w_Y = \frac{-1}{2\lambda_2}\Sigma_{YY}^{-1}\Sigma_{XY}^\top w_X = \frac{1}{\lambda}\Sigma_{YY}^{-1}\Sigma_{XY}^\top w_X \tag{13.8}$$

To show $\lambda_1 = \lambda_2$:
$\frac{1}{4\lambda_1\lambda_2} w_X^\top \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{XY}^\top w_X \underset{(13.7)}{=} w_X^\top \Sigma_{XX} w_X \underset{(13.3)}{=} w_Y^\top \Sigma_{YY} w_Y \underset{(13.6)}{=} \frac{1}{4\lambda_2^2} w_X^\top \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{XY}^\top w_X$
$\implies \frac{1}{4\lambda_1\lambda_2} = \frac{1}{4\lambda_2^2} \implies \lambda_2 = \lambda_1$

To solve eqn. (13.1), each of the w_X, w_Y make up the columns of W_X, and W_Y:

$$\begin{aligned} tr(W_X^\top X Y^\top W_Y) &= \sum_{i=1}^{k} W_{X,i}^\top \Sigma_{XY} W_{Y,i} \\ &= \sum_{i=1}^{k} \frac{1}{\lambda_i} W_{X,i}^\top \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{XY}^\top W_{X,i} && \text{(eqn. (13.8))} \\ &= \sum_{i=1}^{k} \frac{\lambda_i^2}{\lambda_i} W_{X,i}^\top \Sigma_{XX} W_{X,i} && \text{(eqn. (13.7))} \\ &= \sum_{i=1}^{k} \lambda_i && \text{(eqn. (13.3))} \end{aligned}$$

Therefore, taking the columns of W_X to be the eigenvectors corresponding to the top k eigenvalues will maximize our objective.

To derive the projection from the sub-space back into the original spaces, consider vector $\mathbf{x}$ and its projection $\mathbf{u} = W_X^\top \mathbf{x}$. Now $\mathbf{u}\mathbf{u}^\top = W_X^\top \mathbf{x}\mathbf{x}^\top W_X = \mathbf{I}$ from eqn. (13.3). Let M_X be the inverse projection. So $\Sigma_{XX} W_X = \mathbf{x}(W_X^\top \mathbf{x})^\top = \mathbf{x}\mathbf{u}^\top = (M_X \mathbf{u})\mathbf{u}^\top = M_X$. Similarly, the inverse projection of W_Y is $M_Y = \Sigma_{YY} W_Y$.

13.1.2 Training data

Drug side-effects matrix

The SIDER database [13, 14] which maps drugs to their side-effects was used. This database uses text mining to associate side-effects to drugs using drug labels (`http://sideeffects.embl.de/`). After removing all drugs and side-effects with less than two associations and the 10% with the highest number of associations, 61,102 associations between 692 drugs and 1382 side-effects were used. The data was presented as a binary side-effect matrix X with $X_{ij} = 1$ iff side-effect i is associated with drug j.

Drug properties matrix

For each of the drugs, the chemical structure is obtained from the PubChem database [3, 11] (`https://pubchem.ncbi.nlm.nih.gov/`) and assigned 1024 binary structural features [22].

Structural features could be any properties of the drug molecule and its structure, for example, the identity of the atoms in it, rings, or paths, sizes and types of rings etc. The data is presented as a drug property matrix Y with $Y_{kj} = 1$ iff drug j has structural feature i.

13.1.3 Method details

The method relies primarily on CCA, which was explained in section 13.1.1, with some modifications and additions as follows:

A regularized version of CCA is used to avoid overfitting and ensure the matrices are invertible. Σ_{XX} is replaced with $\Sigma_{XX} + \eta\lambda_X\mathbf{I}$ and Σ_{YY} is replaced with $\Sigma_{YY} + \eta\lambda_Y\mathbf{I}$ where λ_X, and λ_Y are the top eigenvalues of Σ_{XX} and Σ_{YY}, respectively.

The projection matrices W_X and W_Y are learned from the drug side-effect and drug properties data and the side-effects for a new query drug q are predicted by calculating its drug property feature vector y_q, and projecting onto the joint subspace using $u_q = W_Y^\top y_q$. The resulting vector is then projected out into the drug-side effect space using $\tilde{x}_q = \Sigma_{XX}W_X u_q$. $\tilde{x}_{q,i}$ is the score of the i-th side-effect.

Network propagation

The CCA scores are combined with scores obtained using network propagation in the side-effect similarity network to try to leverage prior knowledge about similar drugs and similar side-effects. Network propagation will be explained in more detail later (section 13.3.1). Briefly, a similarity matrix S for all the side-effects is constructed with S_{ij} being the normalized Jaccard similarity between the sets of drugs associated with side-effect i and side-effect j.

For a query drug q a prior p_s for each side-effect s is calculated as the similarity between the query and the most similar drug associated with that side-effect: Letting $d(s)$ be the drugs associated with side-effect s, $p_s = \underset{d \in d(s)}{max}\, D(d, q)$.

The Jaccard similarity $D(d,q) = \dfrac{\sum\limits_i y_{d,i} \cdot y_{q,i}}{\sum\limits_i (y_{d,i} + y_{q,i} - y_{d,i} \cdot y_{q,i})}$ where $y_{d,i}, y_{q,i}$ are the values of the i-th feature of the drug property vectors for drugs d and q.

The score vector is the solution to $f = \alpha S f + (1-\alpha)p$. This can be solved exactly by an iterative algorithm that is guaranteed to converge efficiently:
$f^0 = p$
$f^t = \alpha S f^{t-1} + (1-\alpha)p$

For the final score vector the network propagation and CCA score vectors are combined: $score(\tilde{x}_q, f) = \frac{1}{2}\left(\dfrac{1}{1+e^{-(\tilde{x}_q - \mu_{\tilde{x}_q})}} + \dfrac{1}{1+e^{-(a(f-\mu_f)+b)}}\right)$ where a and b are parameters to

be learned.

13.1.4 Performance

Model training

The model is trained on the SIDER and PubChem data using 20-fold cross-validation. In each fold, 5% of the drugs and their side-effect associations are held out. Another 5% are held out as an internal test set. The parameters of the CCA and network propagation η, k and α are learned using the internal test set and then the scoring parameters a and b are learned from the remaining data. Performance of the model is then tested on the held out data.

Methods for comparison

The method is compared against each of the CCA and network propagation based methods alone, an SVM classifier trained for each side-effect and a random baseline which applies the model to a shuffled drug property matrix.

Results

The combined CCA and network propagation based method correctly predicted a known side-effect as the top scoring prediction for 35% of the drugs, and in the top 5 prediction scores for 63% of the drugs. The combined method improves slightly on each of the CCA and network propagation methods alone. The precision-recall curve showing all results is presented in Fig. 13.1.

The advantage of the joint methods which are trained over all side-effects at once over the SVM is evident in Fig. 13.1. Fig. 13.2 shows this result in more detail: predictions on each individual side-effect with the combined method are compared to the SVM method. For each algorithm the area under the precision-recall curve is measured, and for 76% of the side-effects the combined method did better than the SVM.

Predictions on unlabeled data

4,335 drugs that did not appear in the SIDER database were downloaded from DrugBank [25]. Drug property vectors were computed for each and then the method was used to predict side-effects. Of the 448 drugs that also appear in the Hazardous Substance Database (HSDB) [6], 23% of them have the top ranked prediction appearing in the HSDB, and 45% have one of the top 5 predictions in the HSDB.

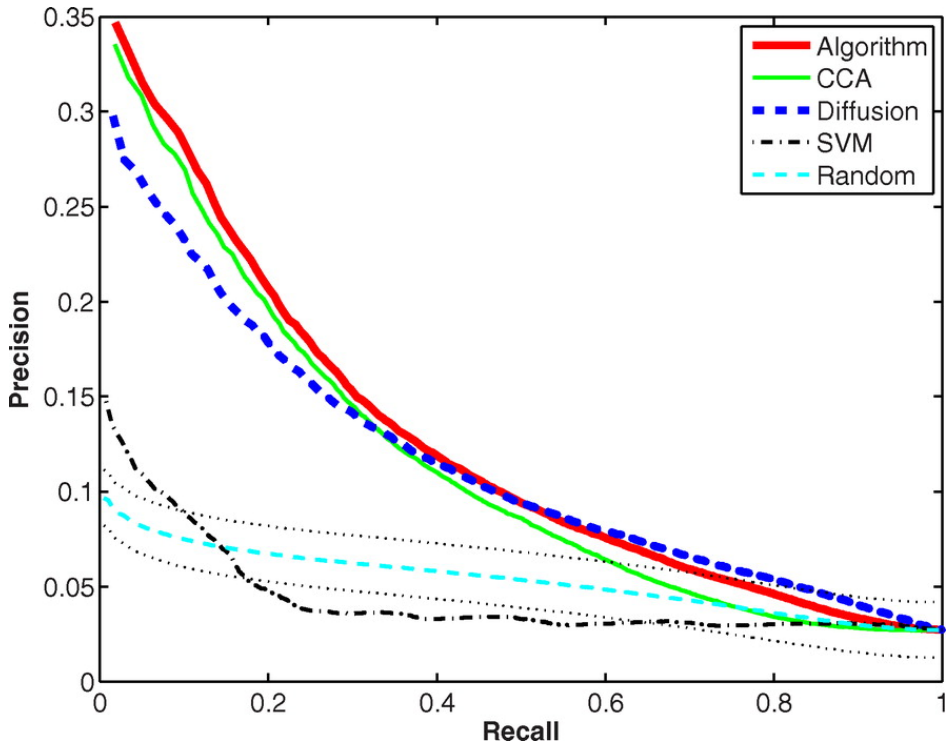

Figure 13.1: Precision-recall curve comparing performance of the different methods based on correctly predicting a known drug side-effect as the top scoring prediction [1]. Algorithm - the combined CCA and network propagation method, CCA - CCA only, Diffusion - network propagation only, SVM - an SVM classifier trained to predict each side-effect separately, Random - the random baseline using the combined algorithm trained on shuffled data.

13.2 PREDICT: Inferring drug indications

A *drug indication* is a reason, such as a set of symptoms or a disease, to treat with a drug. Successfully predicting drug indications is very useful in drug development because it allows a set of candidate experimental molecules to be filtered down to the ones that are most likely to successfully treat a certain disease. Alternatively, new indications could be predicted for existing drugs, which would enable a much faster and less expensive process to reposition that drug to treat other diseases.

PREDICT [8] is a method to infer novel drug targets using known drug-drug and disease-disease similarities to find high-scoring drug-disease matches. It performs better than naive guilt-by-association (GBA) [4] which predicts that if a drug treats two diseases then any drug which treats one of them may treat the other. It also outperforms predictions based on the "Connectivity Map", which maps out gene-expression signatures of many cell-types

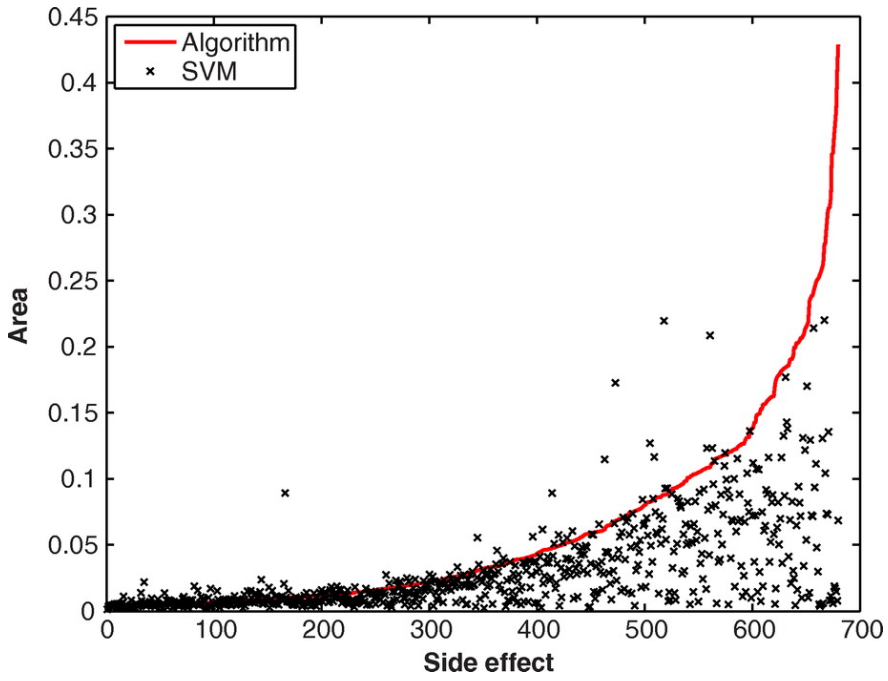

Figure 13.2: Comparison of the areas under the precision-recall curve of individual side-effects using the combined algorithm and an SVM trained for each side-effect[1]. The algorithm has higher AUC than SVM in 76% of the cases.

and states under treatment with many molecules and drugs [15].

13.2.1 PREDICT pipeline overview

A set of known drug-disease associations is collected. Because it is difficult to compare drug-disease associations to each other, PREDICT seeks drug-disease matches in which the drug and disease are most similar to a query drug and disease, respectively. For a candidate drug-disease association, drug-drug and disease-disease similarity scores are computed for a variety of drug and disease characteristics and then combined for each pair of drug and disease characteristics. These combined scores are used as features in a classifier that is trained to predict which drug-disease associations are correct. This pipeline is depicted in Fig. 13.3.

13.2.2 Training data

Gold standard drug-disease associations

313 diseases were extracted from the OMIM (Online Mendelian Inheritance in Man)

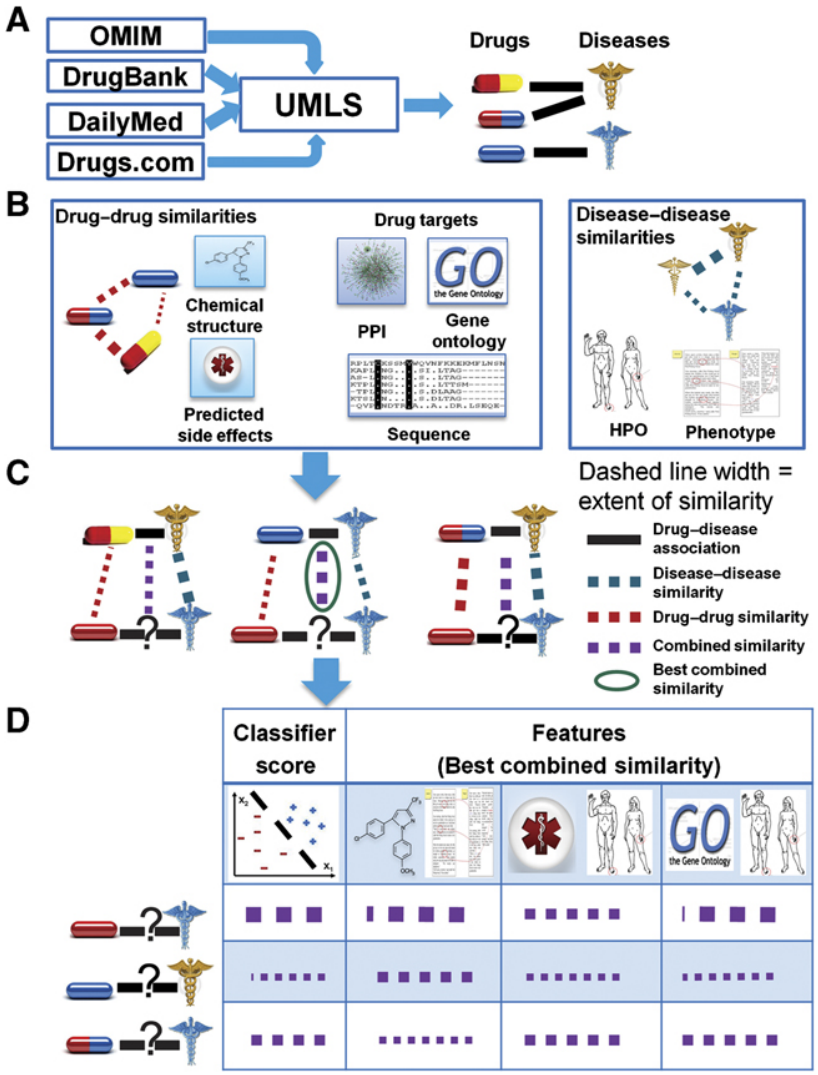

Figure 13.3: Overview of the pipeline of the PREDICT algorithm[8].
A Construct gold-standard database of known disease-drug associations.
B Compute drug-drug and disease-disease similarities based on multiple drug and disease characteristics.
C Combine similarity scores for pairs of different drug-drug and disease-disease similarities.
D Train classifier to predict whether a drug-disease association is correct.

database [9] (`https://www.ncbi.nlm.nih.gov/omim`), and mapped to UMLS (Universal Medical Language System) [2] concepts. Some drug associations are available directly in UMLS, others were extracted from drug labels available from DailyMed (`https://dailymed.nlm.nih.gov/dailymed/`) and DrugBank [25], `www.drugs.com`, and post-marketing clinical

trials. In total 1933 associations between 593 drugs and the diseases were used.

Drug-drug similarities

Five different drug-drug similarity measures were used:

1. **Similarity in chemical structure.** Similarity between vectors of structural features for pairs of drugs is scored as described previously in section 13.1.3 in the work of Atias et al [1].

2. **Similar side-effects.** For each drug the set of associated side-effects is extracted from the SIDER database [13], and, for drugs not in SIDER, the top 10 side-effect predictions by the method of Atias et al [1] described in the previous section (13.1) are used. The similarity of a pair of drugs is scored as the Jaccard score of their side-effects.

3. **Gene target sequence similarity.** The Smith-Waterman sequence alignment score between the gene targets of the drugs normalized by the geometric mean of their self-alignment scores is used.

4. **Gene target PPI network distance.** The score of a pair of drugs d, d' with gene targets g, g' corresponding to proteins p, p' in the PPI network is given by: $S(d, d') = 0.9e^{1+D(p,p')}$ where $D(p, p')$ is the shortest path distance in the PPI network [17].

5. **Gene target GO similarity.** Pairs of drugs are scored using the semantic similarity score [18] between their target genes in the GO ontology.

Disease-disease similarities

Disease-disease similarities are grouped into phenotypic similarities and genotypic similarities

(a) Phenotypic similarities

1. **Phenotype term similarity.** The Medical Subject Heading (MeSH) [16] terms for the disease in the OMIM database are compared using the similarity score from van Driel et al [23].
2. **Phenotype semantic similarity.** This scores semantic similarity [18] using the distance in the Human Phenotype Ontology (HPO) [19].

(b) Genotypic similarities

3. **Genotype differential expression score.** A pair of diseases is scored by the Jaccard similarity of the mutually up-regulated and down-regulated genes.

4. **Differential gene sequence similarity.** The sequences of differentially expressed genes from both diseases are compared using distances in the PPI network in the same way as drug-drug similarity #3.
5. **Differential gene PPI network distance.** Differentially expressed genes from both diseases are compared as in drug-drug similarity #4.
6. **Differential gene GO similarity.** Differentially expressed genes from both diseases are compared using GO similarity as in drug-drug similarity #5.

13.2.3 Classifier, features, and training

All pairs of drug-drug similarity and disease-disease similarity are used to create 10 or 20 features (depending on whether genotypic or phenoytopic disease-disease similarity is used) scoring the similarity between known drug-disease associations and a query drug-disease association.

For any drug-disease association (d_r, d_i) the value for a given feature $f \in S_{drug} \times S_{dis}$ is $Score_f(d_r, d_i) = \max_{(d'_r, d'_i)} \sqrt{(S_{drug}(d_r, d'_r) \times S_{dis}(d_i, d'_i))}$ where the disease-disease and drug-drug similarity scores are described in the previous section (13.2.2).

A logistic regression based classifier is trained on the feature vectors of the positive examples of known drug-disease associations and twice as many randomly generated negative examples that are not on the list of known drug-disease associations. 10-fold cross-validation with 10% of the drugs left out is repeated one hundred times with different randomly selected partitions.

13.2.4 Performance

Comparison to GBA

GBA (guilt-by association) [4] predicts drug associations using the associations of other drugs used to treat the same diseases. Because GBA requires known drug associations, the 10-fold cross-validation scheme is modified to hide only drug-disease associations rather than leaving out drugs entirely. GBA achieves a false positive rate of 0.13 and true positive rate of 0.77. This is compared on the ROC-curve of PREDICT in Fig. 13.4 - since the predictions are not ranked, GBA results are only a single point on the curve. The area under the curve is 0.91.

Comparison to Connectivity Map

Connectivity Map (CMap) [15] is a large-scale experimental effort to map out gene expression signatures in multiple cell lines under treatment with many different molecules at different doses. The result is a database that can be queried to try to find connections between different disease states or the effects of different treatments based on similar expression

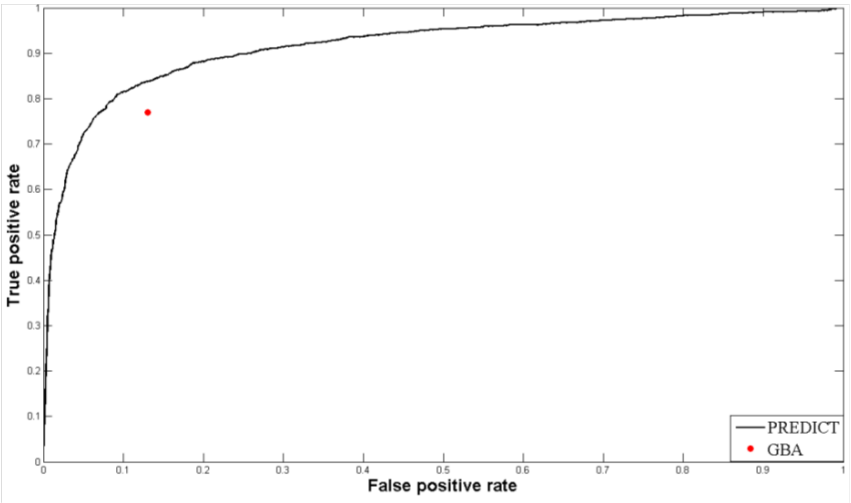

Figure 13.4: ROC-curve of PREDICT compared to the false positive vs. true positive rate of GBA [8].

signatures. For a given disease, drugs that have the strongest opposite effect on differentially expressed genes are predicted as candidate treatments.

36 gene expression signatures for diseases that were both in the OMIM database and complied with CMap requirements were obtained from ArrayExpress [12] (`https://www.ebi.ac.uk/arrayexpress/`). They correspond to 37 diseases with 266 known associations to 71 drugs. PREDICT is trained using genotypic disease-disease similarity features for comparison.

The CMap method obtains an AUC score of 0.42, while PREDICT has a much higher AUC of 0.93.

Drug repositioning predictions

The PREDICT classifier is trained using the set of all known drug-disease associations and a set of twice as many random associations that are not known to be correct and then all possible remaining drug-disease associations are scored (183,676 possible pairs). A cutoff threshold is used to cut the ranked scores and predict associations for drug-disease pairs that score above the threshold.

The threshold is chosen based on the enrichment of 782 low-quality drug-disease associations that were not used in the training data because they only appeared in one database

source. A hypergeometric test is used to test for enrichment of these associations among the predictions using different values of the cutoff. The threshold that yields the highest p-value for the test is used (see Fig. 13.5). This results in 9476 novel drug-disease association predictions, suggesting possible repositionings for 580 (out of 593) of the drugs. 39% of the low quality associations are covered using this threshold, and the p-value of the hypergeometric test for enrichment of this list is $p < 2 \times 10^{-177}$.

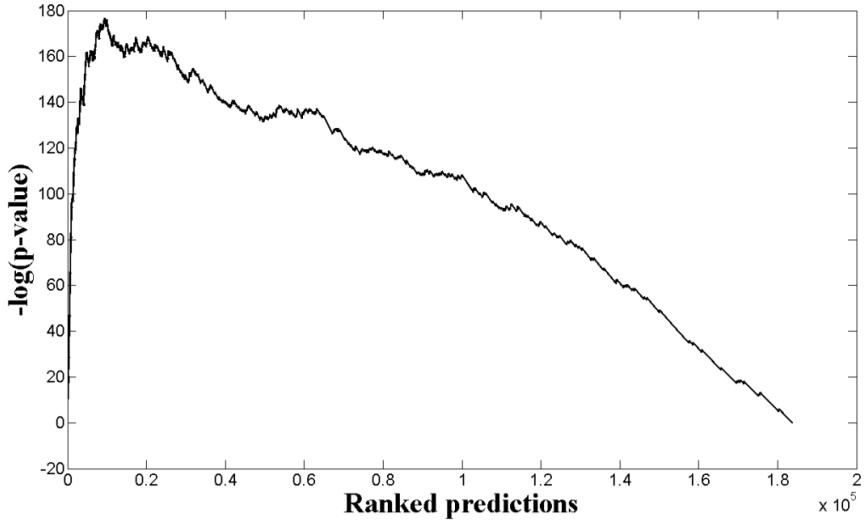

Figure 13.5: The held out low quality associations are distributed among the list of predictions. A hypergeometric test is used to test for enrichment of the known associations in the top ranked predictions for different cutoff thresholds. The p-value of the test is plotted against the number of predictions above the threshold [8].

The predicted associations are validated in two ways:

1) 1943 drug-disease associations that were currently being tested in clinical trials were compared to the list of predicted associations. 27% of the associations being tested in clinical trials are covered by the prediction, with the p-value for a hypergeometric test for enrichment of these associations in the list of predictions $p < 2 \times 10^{-220}$. 38% of the Phase-III clinical trials are covered, with a p-value on the hypergeometric test of $p < 6 \times 10^{-128}$.

2) Tissue specificity of drug targets was matched with tissue specificity of diseases. Drug targets were found to be more significantly differentially expressed in tissues affected by predicted associated diseases with an enrichment p-value of $p < 0.006$

Proof-of-concept for personalized medicine

In a personalized medicine setting, the patients genotype will be used to predict which drugs will be best for treatment. To show that PREDICT can accurately find novel drug-

disease associations using genotype data, 171 disease state gene expression signatures obtained from ArrayExpress were used to train the classifier trained with the genotypic disease-disease similarity features. Associations between 261 drugs and 114 diseases were predicted with an AUC of 0.92. 2103 novel predictions were found using the same method to choose the threshold cutoff as above. These novel predictions were validated validated using clinical trial information and tissue-disease specificity as described previously, indicating that PREDICT can accurately find novel drug treatments using genotype data.

13.3 Inferring personal drug targets

Predicting drug targets based on disease genotypes is not only useful for drug repositioning, it can also enable personalized drug targeting based on the patient-specific disease genotype. This work by Shnaps et al [21] uses network propagation techniques in the protein-protein interaction (PPI) network to identify drug targets given prior knowledge of the specific disease genotype.

The network topology is used to identify modules and pathways defined by enrichment in the disease-specific genotype. Simulated changes to this topology reveal how the gene expression is likely to change when specific genes are targeted with a drug.

13.3.1 Background: network propagation

Intuition

Network propagation methods use the topology of a network to smooth prior information about a subset of the nodes over the entire network. For example, assume you know that a small subset of genes is differentially expressed in a disease. By propagating this information along the edges of the network, other genes that are likely to play a role in the disease can be found.

Intuitively, consider a set of genes G on a network that is known to play a role in a disease. You could guess that all neighbours of these genes also play a role, but this will produce many false positives and many false negatives. Instead, simultaneously consider all paths between all nodes in the network - if network flow is propagated from the nodes in G along the edges of the network for a short time, then nodes that are highly connected to nodes in G will receive the most flow. An example of network propagation is shown in Fig. 13.6.

The network propagation process used in this method is known as random walk with restart (also called PageRank) - the flow is a mixture of flow propagated along the network and the prior information.

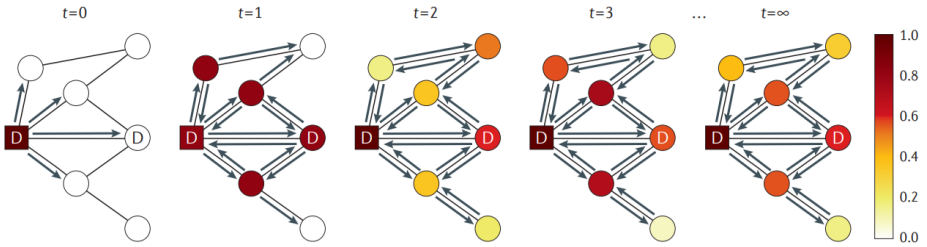

Figure 13.6: Example of the iterative network propagation process [5]. The known disease state (square node) is the source of the flow which is propagated along the network. In each step flow is propagated from each node to its neighbours until steady state is reached. The known and predicted disease associated nodes (marked with 'D') and their close neighbours end up with the highest scores.

Mathematical details

We are given a network $G = (V, E, w)$ where $v \in V$ is a node in the network, $(u, v) \in E$ is an edge from node u to node v, and w_{uv} is the weight assigned to the edge (u, v). We iteratively compute a score function $f_t(v)$ (at iteration t) for all nodes $v \in V$ using prior information giving the score $f_0(v)$ at $t = 0$:

$$f_t(v) = \alpha \left[\sum_{u|(u,v)\in E} f_{t-1}(u) w_{uv} \right] + (1 - \alpha) f_0(v)$$

At each time step the flow at a node is a mixture of the weighted sum of the flows into the node from the previous time step and the initial state of the node. In vector notation:

$f_t = \alpha W f_{t-1} + (1 - \alpha) f_0$ where f_i is an $n \times 1$ vector ($n = |V|$), and $W_{u,v} = w_{uv}$ is the $n \times n$ weight matrix.

At steady-state, $f_t = f_{t-1} \triangleq f^*$, so the steady state solution is the solution of the equation: $f^* = \alpha W f^* + (1 - \alpha) f_0 \implies f^* = (1 - \alpha)(\mathbf{I} - \alpha W)^{-1} f_0$.

$\alpha \in (0, 1)$, therefore if W is chosen so that its eigenvalues are in $[-1, 1]$ then the eigenvalues of $(\mathbf{I} - \alpha W)$ will be positive and the inverse will exist (in the next section we show how to choose W to satisfy this requirement). We call the maximum absolute value of an eigenvector of a matrix its *spectral radius* denoted $\rho(\cdot)$.

To see that the iterative formula converges to this steady state, unwind the iteration: $f_t = (\alpha W)^t f_0 + (1 - \alpha) \sum_{i=0}^{t-1} (\alpha W)^i f_0$

Again, since $0 < \alpha < 1$ and $\rho(W) \leq 1$, we have $\rho(\alpha W) < 1$ and therefore the sequence $\lim_{t\to\infty} (\alpha W)^t$ converges to zero (intuitively, think of the eigenvalues converging to zero since λ^k is an eigenvalue of $(\alpha W)^k$ if λ is an eigenvalue of αW, and we know $\lambda \in (-1, 1)$).

In the second term there is a matrix geometric series $\sum_{i=0}^{t-1} (\alpha W)^i$. Analogously to the

regular geometric series, for matrix T, $S_n = \mathbf{I} + T + T^2 + \cdots + T^n$ converges as $n \to \infty$ iff $\rho(T) < 1$, and if it does then it converges to $(\mathbf{I} - T)^{-1}$ (for a good, intuitive explanation of this point see lecture notes by Charlie Watson [24]).
Plugging this in to our equation we have that $\lim_{t\to\infty} f_t = (1-\alpha)(\mathbf{I}-\alpha W)^{-1} f_0 = f^*$ as required.

13.3.2 The method

Network propagation implementation

The method uses multiple network propagations over the PPI and compares them. A prior set P is defined as either the set of patient-specific mutated or differentially expressed genes.

Network propagation is performed as described above with the following settings of the variables:
The initial vector $f_0 = Y$ where $Y_v = \begin{cases} 1, & v \in P \\ 0, & v \notin P \end{cases}$.
The weight matrix $W_{uv} = \dfrac{w(u,v)}{\sqrt{\sum_{v'} w(u,v') \sum_{u'} w(v,u')}}$ where $w(u,v)$ is the score of the edge in the PPI network, representing its reliability.
$\alpha = 0.9$ was chosen after the results were shown to be robust over a range of different values.

Proof that $\rho(W) \leq 1$

Letting $W'_{uv} = w(u,v)$ be the matrix of PPI edge weights, then the way we have defined W can be written $W = D^{-1/2} W' D^{-1/2}$ where $D_{u,v} = \begin{cases} 0, & u \neq v \\ \sum_{v'} w(u,v'), & u = v \end{cases}$.

Two matrices A, and B are *similar* ($A \sim B$) if there is an invertible matrix P such that $P^{-1}AP = B$. Since the PPI network is connected, all entries on the diagonal of $D^{-1/2}$ are non-zero, therefore it is invertible.

We have: $W = D^{-1/2} W' D^{-1/2} \sim D^{-1/2} D^{-1/2} W' D^{-1/2} D^{1/2} = D^{-1} W'$. Now, similar matrices have the same eigenvalues, so we show that $\rho(D^{-1}W') \leq 1$.

$D^{-1}W'$ is a *stochastic* matrix, meaning that it has only non-negative entries and each of its rows sums to 1.

For any stochastic matrix A suppose there exists an eigenvalue $\lambda > 1$. But each element of Ax cannot exceed x_{max} since the rows of A are made up of non-negative elements summing to 1. On the other hand $Ax = \lambda x$, so each element of λx is also $\leq x_{max}$. Specifically, $\lambda x_{max} \leq x_{max}$, contradicting the assumption that $\lambda > 1$.

Propagation score vectors

For an individual patient, network propagation is run using their set of mutated genes as the prior set P. The score of a gene is the rank of its score after running the network propagation, $s(v) = rank(f_v^*)$. The score vector for a gene set of interest G, for example all the differentially expressed genes for the patient, contains the scores for those genes: $S_g = s(g)$. A simulated healthy score vector is calculated in the same way by randomly choosing a prior set P' that is the same size as the patient's prior P and running network propagation. Similarly, score vectors for simulated "knockout" experiments are calculated using the patient's prior set P and running network propagation on the modified network with the knockout removed from the PPI network.

Back2Healthy score

The effect of a knockout on the score vector is measured using the Back2Healthy score. Let S_p be a patient score vector, S_k a knockout score vector, S_i, $i = 1, 2..n$ be $n = 1000$ simulated healthy vectors obtained by network propagation using k random genes as the prior set (where $k = |P|$, the size of the patient's prior set).

For an individual gene g in the score vectors, $Q_{pg} = \frac{|\{S_{i,g} < S_{p,g}\}|}{n}$ and $Q_{kg} = \frac{|\{S_{k,g} < S_{p,g}\}|}{n}$ are the quantiles of the patient and knockout scores.

The *Back2Healthy score* is $b2h(S_p, S_k) = \frac{\sum_{g \in G} |Q_{pg} - Q_{kg}|}{|G|}$

13.3.3 Performance

The method was tested on acute myeloid leukemia (AML) data from The Cancer Genome Atlas (TCGA) (`https://cancergenome.nih.gov/`). Mutation and gene expression data is available for 174 AML patients.

The PPI network, obtained from HIPPIE [20] has around 190 thousand interactions between 15 thousand proteins.

Validation with known causal genes

Two sets of known AML causal genes were obtained, 10 genes from KEGG (Kyoto Encyclopedia of Genes and Genomes) [10], and 94 from COSMIC (Catalogue Of Somatic Mutations In Cancer) [7]. An additional 533 general cancer causal genes were obtained from COSMIC as well.

For all patients, the ranks of all genes after network propagation were aggregated and the top 10% were tested for enrichment with the known causal genes using a hypergeometric test. The p-value in all cases is $p < 10^{-5}$.

Personalized drug target prediction

The network is perturbed by removing each gene (excluding the patient's mutated genes that are already known), and the Back2Healthy score is computed. The top 10% of the genes (across all patients) are highly enriched with known AML drug targets obtained from DrugBank and COSMIC. When a "consensus" patient profile is constructed using commonly mutated and differentially expressed genes in AML, enrichment for known targets among the high scoring genes is insignificant. This difference is shown in Fig. 13.7, highlighting the usefulness of personalized predictions in this case.

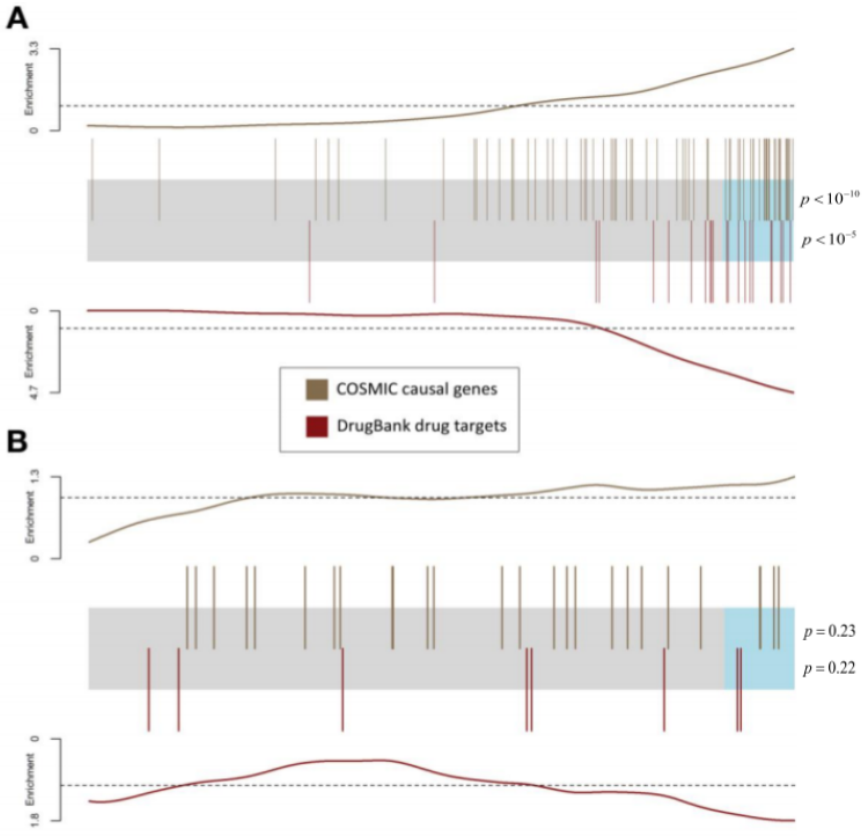

Figure 13.7: Enrichment of known AML causal genes and drug targets in the set of high scoring predicted genes. Predicted genes are represented by the grey box, with the top 10% in blue. Every overlaid bar stands for a single gene from the COSMIC and DrugBank sets of known genes. Relative enrichment of the known set among the predictions is shown by the lines above and below the predicted gene box. P-values for enrichment in the top 10% of predicted genes are shown to the right of the blue regions.
A Predictions and enrichments aggregated across individuals
B Predictions and enrichments for a consensus patient

Bibliography

[1] Nir Atias and Roded Sharan. An algorithmic framework for predicting side effects of drugs. *Journal of Computational Biology*, 18(3):207–218, 2011.

[2] Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270, 2004.

[3] Evan E Bolton, Yanli Wang, Paul A Thiessen, and Stephen H Bryant. Pubchem: integrated platform of small molecules and biological activities. *Annual Reports in Computational Chemistry*, 4:217–241, 2008.

[4] Annie P Chiang and Atul J Butte. Systematic evaluation of drug–disease relationships to identify leads for novel drug uses. *Clinical Pharmacology & Therapeutics*, 86(5):507–510, 2009.

[5] Lenore Cowen, Trey Ideker, Benjamin J Raphael, and Roded Sharan. Network propagation: a universal amplifier of genetic associations. *Nature Reviews Genetics*, 2017.

[6] George Charles Fonger. Hazardous substances data bank (HSDB) as a source of environmental fate information on chemicals. *Toxicology*, 103(2):137–145, 1995.

[7] Simon A Forbes, David Beare, Harry Boutselakis, Sally Bamford, Nidhi Bindal, John Tate, Charlotte G Cole, Sari Ward, Elisabeth Dawson, Laura Ponting, et al. Cosmic: somatic cancer genetics at high-resolution. *Nucleic Acids Research*, 45(D1):D777–D783, 2016.

[8] Assaf Gottlieb, Gideon Y Stein, Eytan Ruppin, and Roded Sharan. Predict: a method for inferring novel drug indications with application to personalized medicine. *Molecular Systems Biology*, 7(1):496, 2011.

[9] Ada Hamosh, Alan F Scott, Joanna Amberger, Carol Bocchini, David Valle, and Victor A McKusick. Online mendelian inheritance in man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Research*, 30(1):52–55, 2002.

[10] Minoru Kanehisa and Susumu Goto. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

[11] S Kim, PA Thiessen, EE Bolton, J Chen, G Fu, A Gindulyte, L Han, J He, S He, BA Shoemaker, et al. Pubchem substance and compound databases. *Nucleic Acids Research*, 44(D1):D1202–13, 2016.

[12] Nikolay Kolesnikov, Emma Hastings, Maria Keays, Olga Melnichuk, Y Amy Tang, Eleanor Williams, Miroslaw Dylag, Natalja Kurbatova, Marco Brandizi, Tony Burdett, Karyn Megy, Ekaterina Pilicheva, Gabriella Rustici, Andrew Tikhonov, Helen Parkinson, Robert Petryszak, Ugis Sarkans, and Alvis Brazma. ArrayExpress update–simplifying data submissions. *Nucleic Acids Research*, 43(Database issue):D11136, January 2015.

[13] Michael Kuhn, Monica Campillos, Ivica Letunic, Lars Juhl Jensen, and Peer Bork. A side effect resource to capture phenotypic effects of drugs. *Molecular Systems Biology*, 6(1):343, 2010.

[14] Michael Kuhn, Ivica Letunic, Lars Juhl Jensen, and Peer Bork. The SIDER database of drugs and side effects. *Nucleic Acids Research*, 44(D1):D1075–D1079, 2015.

[15] 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.

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

[17] Liat Perlman, Assaf Gottlieb, Nir Atias, Eytan Ruppin, and Roded Sharan. Combining drug and gene similarity measures for drug-target elucidation. *Journal of Computational Biology*, 18(2):133–145, 2011.

[18] Philip Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[19] Peter N Robinson, Sebastian Köhler, Sebastian Bauer, Dominik Seelow, Denise Horn, and Stefan Mundlos. The human phenotype ontology: a tool for annotating and analyzing human hereditary disease. *The American Journal of Human Genetics*, 83(5):610–615, 2008.

[20] Martin H Schaefer, Jean-Fred Fontaine, Arunachalam Vinayagam, Pablo Porras, Erich E Wanker, and Miguel A Andrade-Navarro. HIPPIE: Integrating protein interaction networks with experiment based quality scores. *PloS ONE*, 7(2):e31826, 2012.

[21] O Shnaps, E Perry, D Silverbush, and R Sharan. Inference of personalized drug targets via network propagation. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, volume 21, pages 156–167, 2016.

[22] Christoph Steinbeck, Yongquan Han, Stefan Kuhn, Oliver Horlacher, Edgar Luttmann, and Egon Willighagen. The chemistry development kit (CDK): An open-source java library for chemo-and bioinformatics. *Journal of Chemical Information and Computer Sciences*, 43(2):493–500, 2003.

[23] Marc A Van Driel, Jorn Bruggeman, Gert Vriend, Han G Brunner, and Jack AM Leunissen. A text-mining analysis of the human phenome. *European Journal of Human Genetics*, 14(5):535–542, 2006.

[24] Charlie Watson. The geometric series of a matrix. `http://www.math.uvic.ca/~dcwatson/work/geometric.pdf`, October 2015.

[25] David S Wishart, Craig Knox, An Chi Guo, Savita Shrivastava, Murtaza Hassanali, Paul Stothard, Zhan Chang, and Jennifer Woolsey. Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Research*, 34(suppl_1):D668–D672, 2006.