An Algorithmic Framework for Predicting Side-Effects of Drugs

Nir Atias and Roded Sharan

Blavatnik School of Computer Science
Tel Aviv University
Tel Aviv 69978, Israel
{atiasnir,roded}@post.tau.ac.il

Abstract. One of the critical stages in drug development is the identification of potential side effects for promising drug leads. Large scale clinical experiments aimed at discovering such side effects are very costly and may miss subtle or rare side effects. To date, and to the best of our knowledge, no computational approach was suggested to systematically tackle this challenge. In this work we report on a novel approach to predict the side effects of a given drug. Starting from a query drug, a combination of canonical correlation analysis and network-based diffusion are applied to predict its side effects.
We evaluate our method by measuring its performance in cross validation using a comprehensive data set of 692 drugs and their known side effects derived from package inserts. For 34% of the drugs the top scoring side effect matches a known side effect of the drug. Remarkably, even on unseen data, our method is able to infer side effects that highly match existing knowledge. Our method thus represents a promising first step toward shortcutting the process and reducing the cost of side effect elucidation.

Key words: Prediction, Canonical correclation analysis, Network diffusion, Drug targets

1 Introduction

Systems medicine is an emerging discipline in systems biology that aims at integrating clinical databases with large scale molecular interaction data to elucidate diseases and drugs [1]. Applications of such approaches range from predicting gene-disease associations and drug-target relations [2] to discovering new drugs [1].

Beyond the development of new drug leads, a critical stage in drug development is the identification of side effects that result from treatment with the drug. Drug safety has gained much attention in recent years, and has become a serious bottleneck in drug development, leading to the reduction in the number of newly approved drugs despite the enormous research efforts invested in drug discovery [3]. The elucidation of adverse reactions may occur long after the approval of a drug, as in the case of rosiglitazone maleate (Avienda ®), and can even lead to discontinuing the use of the drug, as in the case of rofecoxib (Vioxx ®) (see also [4]).

The only attempt we are aware of to predict side effects is due to Xie et al. [5]. They used protein-ligand binding predictions to identify off-targets for a given drug. The latter were used to pinpoint known pathways that are likely to be affected by the drug and consequently predict its side effects. This approach depends on protein structure information and accurate pathway information, which greatly limits its applicability. In particular, biological processes involved in side effect reaction to treatment are still largely unknown and inferring side effects, even when given the respective drug targets, remains a formidable task [6].

In contrast to the sparse work on side effect prediction, the related area of elucidating gene-disease and drug-target associations has become very active in recent years. State of the art methods for predicting gene-disease associations are based on the observation that genes that cause similar diseases tend to lie close to one another in a network of protein-protein interactions [7, 8]. Given a query disease, genes causing similar diseases are identified, and a network-based computation is used to prioritize candidate genes according to their proximity to this initial set [9–11]. Several methods have been suggested for drug-target prediction. Campillos et al. [2] construct a comprehensive drug-side effect data set and use it, in conjunction with chemical properties, to define a similarity metric between drugs. Given a query drug, they identify similar drugs and propose their targets as candidate targets for the drug. Yildirim et al. [12] examine a drug-target network in which drugs are connected based on shared targets and find that drug cluster according to the Anatomical Therapeutic Chemical (ATC) classification. Despite the insights offered by this network, no prediction scheme was suggested. A somewhat related work by Yang et al. [13] uses text mining to highlight genes responsible for serious adverse drug reactions. Finally, Kutalik et al.[14] integrate gene expression data and drug response data under different cell lines. They identify co-modules of genes and drugs with similar behavior across a subset of the cell lines, leading to the prediction of new drug targets.

Here we present a first systematic approach for predicting side effects for drugs. Our approach combines two algorithms to predict side effects. The first

algorithm is based on canonical correlation analysis which is used to obtain a low dimensional subspace that jointly contains drug-side effect associations and molecular data on drugs, such as their chemical structure. Data on new drug queries are projected onto this subspace and an efficient algorithm is used to identify corresponding side effect vectors that best correlate with the projected data. The second algorithm is based on diffusion in a side effect similarity network. Starting from a prior solution that is based on the side effects of drugs that are similar to the query, a diffusion process is used to obtain final scores that are smooth over the network.

We evaluate our method by measuring its performance in 20-fold cross validation using a comprehensive data set of 692 drugs and their known side effects derived from package inserts. For 34% of the drugs the top scoring side effect matches a known side effect of the drug; for almost two thirds of the drugs our method infers a correct side effect among the five top ranking predictions. In comparison, applying the algorithm to randomized instances, "correct" predictions are obtained for only 10% (top ranking) or 32% (among the five top ranking) of the drugs. We further validate our method in a blind test on ∼450 drugs that were not part of the initial data, but for which some side effect information exists in the literature. Remarkably, even on these unseen data, our method is able to infer side effects that highly match existing knowledge: for 45% of the drugs, a correct side effect is included among the five top ranking predictions. Finally, we show the utility of our method in drug target elucidation. We make predictions for over 4,000 drugs for which no side effect information is readily available. We then show a significant correlation between the side effect similarity and target similarity among these drugs. Not only does this agree with a previous study that used this correlation to predict drug targets [2], but importantly it suggests that target prediction algorithms can be applied also in the vast regime of drugs whose side effects have not been mapped to date.

2 Algorithmic Approach

We present two novel algorithms for predicting side effects, which are then combined to yield the final ranking of side effects for a given drug. The first algorithm is based on canonical correlation analysis. It requires as input an attribute matrix describing the drugs. In a training phase it learns a linear projection of the attribute and side effect data onto a joint low-dimensional space such that per drug, the correlation between the projected vectors of attributes and side effects is maximized. This projection is then used to infer the side effects of a test drug. The second algorithm is based on diffusion in a side effect similarity network. Given a query drug, the algorithm first identifies side effects of similar drugs. Starting from these side effects, a diffusion process is executed to obtain a final ranking that is smooth over the side effect network.

In the following we denote the number of drugs by n and the number of side effects by m. We assume that we are given as input a drug attribute matrix $R_{p\times n}$, in which each drug is described by a set of p attributes; a drug-side

effect association matrix $E_{m\times n}$; and an attribute vector q for a query drug. In a preprocessing step we normalize the rows of E and R to have mean 0.

2.1 Canonical Correlation Analysis

In canonical correlation analysis we aim to uncover and exploit the correlation between the two data sets that represent the drugs, R and E in our case, by projecting these data sets into a joint space and using the projection for the prediction task. We assume that corresponding vectors in each of the data sets should be highly correlated under some joint representation. Intuitively, our objective is to find two projection matrices, $(W_E)_{m\times k}$ and $(W_R)_{p\times k}$, that project E and R onto a common k-dimensional subspace in which the correlations between projected vectors corresponding to the same drugs are maximized. The projection vectors are chosen so that the set of projected vectors under each of the data sets will be orthonormal.

Formally, the problem is defined as follows:

$$\begin{aligned} &\max_{W_E,W_R} \operatorname{Tr}\left(W_E^T E R^T W_R\right), \qquad \text{subject to} \\ &W_E^T E E^T W_E = W_R^T R R^T W_R = I \end{aligned} \tag{1}$$

where $\operatorname{Tr}(M)$ is the trace of M. As shown in the Appendix, the resulting optimization problem can be solved by reducing it into an eigenvector problem on an appropriately defined matrix, and using the k eigenvectors with the largest eigenvalues to define the projection.

To avoid over-fitting and to account for numerical instabilities we use a regularized version of CCA [15]. The regularization takes additional regularization factors η_E and η_R which are used to penalize the norm of the column vectors of W_E and W_R. Instead of using two regularization factors we follow Wolf et al. [16] and use a single additional regularization parameter, η, and the largest eigenvalues, λ_E and λ_R, of EE^T and RR^T, respectively (see Appendix).

Finally, we use the projection matrices to compute a score vector for the query drug. To this end, the attribute vector q of the query drug is projected onto the subspace identified by the CCA: $q_{proj} = W_R^T \cdot q$. In accordance with the goal of CCA, we seek a corresponding side effect vector v whose projection maximizes the correlation to q_{proj}. Formally, we seek:

$$\max_{v} \frac{q_{proj}^T W_E^T v}{|q_{proj}| \left|W_E^T v\right|} \tag{2}$$

The maximum is achieved when $W_E^T v = q_{proj}$; however, as W_E^T projects v into a smaller subspace, the system of equations is under-determined. To obtain a unique solution, f, we use the pseudoinverse of W_E^T, denoted by $\left(W_E^T\right)^{\dagger}$. In general, a pseudoinverse is computed using singular value decomposition, but here we can use the specific structure of W_E to compute it more efficiently using matrix multiplication, as detailed in the Appendix.

2.2 Diffusion-based Prediction

The second algorithm that we use is based on a diffusion process in a side effect similarity matrix, aiming to score side effects so that: (i) prior information is taken into account; and (ii) similar side effects receive similar scores. Such an approach was applied successfully for predicting disease-causing genes [10].

Formally, given a similarity matrix between side effects (S) and a prior information vector y, we seek a score vector f which satisfies:

$$f = \alpha S \cdot f + (1 - \alpha)\, y \tag{3}$$

where $\alpha \in [0, 1]$ is a parameter reflecting the relative importance of the two (possibly contradicting) requirements on f.

We build S based on E, by measuring the Jaccard coefficient between the sets of drugs associated with each side effect. Formally, let $\Gamma(s)$ denote the set of drugs associated with side effect s. Then the similarity between side effects i and j is given by the Jaccard coefficient of their corresponding drug sets:

$$\tilde{S}_{i,j} = \frac{|\Gamma(i) \cap \Gamma(j)|}{|\Gamma(i) \cup \Gamma(j)|}. \tag{4}$$

To account for the different similarity profiles of different side effects, we normalize the similarities by setting $S_{i,j} = \tilde{S}_{i,j} / \sqrt{P_i \cdot P_j}$, where $P_i = \sum_j \tilde{S}_{i,j}$.

The computation of the prior vector is based on a similarity function between drugs. The latter is computed using R and its specific definition depends on the attribute data at hand, as described in Section 3.1. Let $D_{q,d}$ denote the similarity between the query drug q and any other drug d. We apply a nearest neighbor approach, defining the prior value for side effect s as the highest similarity score $D_{q,d}$ between a drug d and the query, across all drugs associated with s: $y_s = \max_{d \in \Gamma(s)} \{D_{q,d}\}$.

In [17] it is shown that if the eigenvalues of S are in $[-1, 1]$ (which is the case under our normalization) then f can be computed using an iterative process

$$f^0 = y; \quad f^t = \alpha S \cdot f^{t-1} + (1 - \alpha)\, y \tag{5}$$

which efficiently converges to the analytical solution: $f = (I - \alpha S)^{-1} (1 - \alpha)\, y$.

2.3 Merging Score Vectors

Invoking the CCA based prediction and the diffusion based prediction yields two score vectors. Different strategies for merging these two vectors into a single ranking can be applied. Merging the two score vectors directly is problematic as the scores are not necessarily comparable. We follow ideas from Lin et al. [18], who use a logistic function for the merging. The logistic function is a monotonic transformation of the score, thus preserving the relative ranking of each algorithm on the one hand, while rescaling the scores to the same range on the other hand.

Formally, given score vectors s_1 and s_2, with mean values $\bar{s}_1$ and $\bar{s}_2$, respectively, the combined score vector is given by:

$$\text{score}(s_1, s_2) = \frac{1}{2}\left(\frac{1}{1+e^{-(s_1-\bar{s}_1)}} + \frac{1}{1+e^{-b-a(s_2-\bar{s}_2)}}\right) \quad (6)$$

where a and b are two free parameters which adjust between the two scoring systems.

2.4 Parameter Tuning and Performance Evaluation

The prediction algorithm has several parameters. Two parameters are used by the CCA algorithm: η – the regularization parameter, and k – the dimension of the subspace to which the data are projected. One parameter is used by the diffusion algorithm: α – the relative weight of the prior term vs. the smoothing term. Two final parameters, a and b, control the merging of the two score vectors.

We tune the parameters using grid search in a cross validation setting. Specifically, in each iteration of a 20-fold cross validation, 5% of the drugs serve as a test set and their side effect associations are hidden; 5% additional drugs serve as an internal test set to tune the parameters; the rest 90% of the drugs are used for training. First, the parameters of the two algorithms, η, k and α, are learned, maximizing the performance of each algorithmic variant separately on the internal test data. Next, the mixing parameters a and b are learned. Finally, the learned parameters are used to evaluate the performance of the algorithm on the test data. We note that in each cross validation iteration, the CCA projection and the side effect similarity network are recomputed.

We measure the quality of the predictions by computing a precision-recall curve for varying numbers of predictions per drug. Given a desired number of predictions, k, we consider the union of the top k ranking predictions for all drugs and compute: (i) *precision* – the percent of correct predictions; and (ii) *recall* – the percent of true side effects that were recovered. To summarize the curve we compute the area under it, as well as the area under its leftmost section where the recall is smaller than 0.2. To resolve cases in which several side effects attain the same score, we adjust the ranks of these side effects to be their average (unadjusted) rank.

To assess the significance of the results obtained by the algorithm, we applied it also to randomized instances of the data. The randomization was performed by permuting the columns of the drug-side effect association matrix E, thus randomizing the relations between drugs and their side effect vectors, while preserving the distribution of side effects in the data.

3 Results

3.1 Data Retrieval and Similarity Computations

Drugs and their associated side effects were obtained from SIDER [19], an online database containing drug-side effect associations extracted from package inserts

using text mining methods [2]. This data set spans 880 drugs, 1382 side effects, and 61,102 drug-side effect associations. Drugs and side effects vary greatly in their number of associations. Some effects are present in almost all drugs (e.g., dizziness, edema and nausea), while others are associated with very few drugs (e.g. flashbacks, rectal polyp); and similarly for drugs. Thus, we filtered from the association data drugs and side effects that lie at the top 10% (greater than 151 associations for drugs and 127 associations for side effects), as well as side effects and drugs having less than two association. The resulting drug-side effect network contained 692 drugs, 680 side effects and 12,871 associations. These data were represented in a binary association matrix, E, where $E_{s,d} = 1$ if and only if drug d is associated with side effect s.

The prediction algorithm can be applied with various drug attribute schemes, drug similarity measures and side effect similarity measures. For drugs, we experimented with two supporting data sets: (i) chemical hashed fingerprints; and (ii) NCI-60 drug response data for the different drugs under different cell lines [14]. For side effects, we based our similarity computation on their sets of associated drugs (see Section 2).

Chemical data based computation. Structures for the drugs molecules were downloaded from PubChem [20]. Hashed fingerprints based on these chemical structures were computed using the open source Chemistry Development Kit (CDK) [21, 22]. The description matrix, R, used by the CCA prediction algorithm, is the matrix whose columns are the hashed fingerprints.

The similarity score between drugs, used by the diffusion algorithm, was calculated according to the Tanimoto 2D score between the two fingerprints, which is equal to their Jaccard coefficient. Formally, let r^d denote the hashed fingerprint for drug d ($r_i^d \in \{0,1\}$, $i \in 1 \ldots 1024$). The similarity score between two drugs, j and l, is given by:

$$D_{j,l}^{(chem)} = \text{Tanimoto}\left(r^j, r^l\right) = \frac{\sum_i \left(r_i^j \cdot r_i^l\right)}{\sum_i \left(r_i^j + r_i^l - r_i^j \cdot r_i^l\right)} \tag{7}$$

Response data based computation. We downloaded the drug response data used in [14] from `http://serverdgm.unil.ch/bergmann/PingPong.html`. The data were used to build the description matrix R. An entry in R lists the concentration of a drug that is needed to achieve 50% growth inhibition under a certain cell line ($\log(\text{GI}_{50})$). Missing data were replaced by the mean response to the drug over all cell lines. The similarity score between drugs, used by the diffusion algorithm, was calculated according to the Pearson correlation between the corresponding response profiles.

3.2 Chemical Structure Based Prediction Performance

In our first application of the algorithm we used the drug chemical structure information as supporting data. We tested the algorithm in a 20-fold cross validation setting, where in each cross validation iteration 5% of the data were

hidden, serving as a test set, and the other 95% served as a training set. Within the training set, an internal cross validation was conducted to train the parameters of the algorithm as described in Section 2.4.

Overall, for 34.7% (240) of the 692 drugs the algorithm ranked first one of the known side effects of these drugs. For 63.4% (439) of the drugs, a correct side effect was ranked among the top five scoring side effects. In comparison, when applying our algorithm to randomized instances of these data, for only 68.1 ($\pm$7.69, 9.85%) of the drugs, on average, the top ranking side effect matched a known side effect of the drug; and only 225.1 ($\pm$12.8, 32.5%) of the drugs, on average, had a known side effect among the top five ranking side effects. These marked differences are also reflected in the areas under the curve: 0.119 on the real data and 0.0524 ($\pm$0.0009) at random (see Figure 1A and Table 1).

We further compared the performance of the combined algorithm to those of applying the CCA or diffusion-based computations by themselves. As evident from the results in Figure 1A and Table 1, the combined algorithm outperforms the diffusion-based variant and is marginally better than the CCA based variant in all evaluation measures.

Some side effects are more prevalent than others and shared across many drugs. To examine the impact of side effect frequency on the prediction task, we have devised an algorithm that randomly ranks side effects according to their frequency distribution. The algorithm scores side effects by iteratively choosing side effects according to their empirical distribution in the training data, each time incrementing their score. As shown in Figure 1A this algorithm performs worse than all other variants, suggesting that the prevalence of side effects is not sufficient to explain association with drugs.

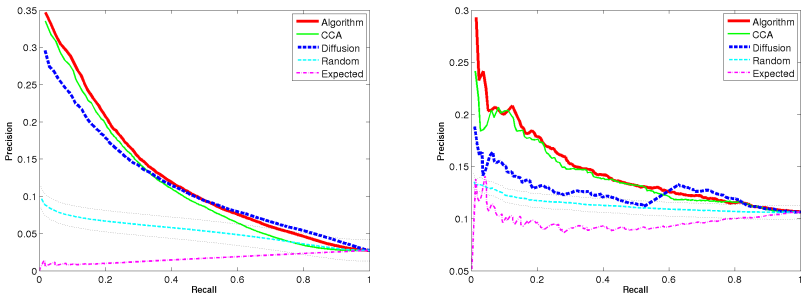


Fig. 1. Performance evaluation. Dotted lines depict standard deviation for random curves. (A) Performance comparison using chemical structures as supporting data. (B) Performance comparison using drug response as supporting data.

Table 1. Performance statistics of the different algorithmic variants and a comparison to a random application. *Top1* lists the number of drugs having a known side effect ranked highest. *Top5* lists the number of drugs having at least one known side effect among the 5 highest ranking side effects. *Area* is the total area under the precision-recall curve; and *Area20* is the area under the leftmost (recall < 0.2) section of the precision-recall curve. The best result in each row appears in bold.

Data Set	Result	Combined alg.	CCA	Diffusion	Expected	Random
Chemical	Top1	**240**	232	206	0	68.16±7.69
	Top5	**439**	430	407	0	225.1±12.8
	Area	**0.1190**	0.1095	0.1111	0.0168	0.0524±0.0009
	Area20	**0.0483**	0.0465	0.0412	0.0017	0.0145±0.0005
Response	Top1	**17**	14	11	3	7.92±2.36
	Top5	**29**	26	25	23	24.86±3.23
	Area	**0.1419**	0.1382	0.1241	0.097	0.1122±0.005
	Area20	**0.0373**	0.035	0.0275	0.0204	0.0236±0.0024

3.3 Response Based Prediction Performance

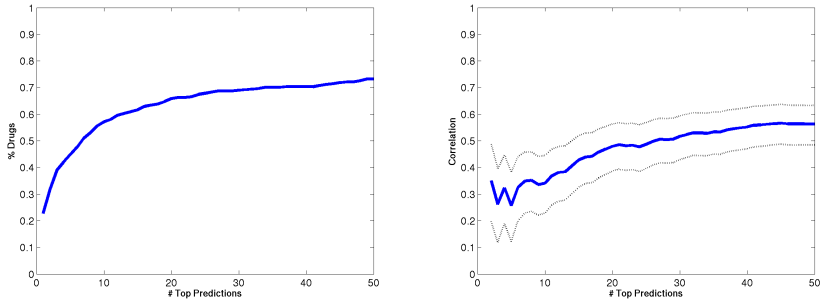
We additionally applied our algorithm using the drug response data. As the response information was not available for many of the drugs, the application was limited to 58 drugs, spanning 188 side effects. The algorithm ranked one of the known side effects highest for 17 (29%) of the drugs. For 29 (50%) drugs a correct side effect was ranked among the top 5 scoring side effects. These results significantly outperformed the random expectation (see Table 1). Precision-recall curves for the different algorithmic variants are displayed in Figure 1B. As for the chemical structure data, the combined algorithm outperformed diffusion based variant significantly and is marginally better than the CCA variant. The randomized algorithm based on side effect expectancy based on the occurrence distribution performs worse than all other variants.

3.4 A Large Scale Blind Test

To further validate our approach, we downloaded from DrugBank [23, 24] a compilation of 4,335 drugs that were not available in SIDER. Chemical structures and hashed fingerprints for these new drugs were computed as described in section 3.1, and side effect rankings were calculated using the combined algorithm.

To evaluate the results of our prediction algorithm, we used the Hazardous Substances Data Bank (HSDB), an online peer reviewed database focusing on toxicology of potentially hazardous chemicals (see [25]). For 448 drugs that had matching records in HSDB, the text in the Human Health Effects section was downloaded and a simple textual search scheme was applied to extract annotated side effects. For 102 (22.8%) of the drugs, the side effect that was ranked highest by our algorithm was also associated to the corresponding drug in HSDB (see Figure 2A). For 201 (44.9%) of the drugs, one or more of the 5 top scoring side effects were confirmed by HSDB.

We believe that the accuracy in the validation is in fact higher, as only exact string matches were considered in the textual search and the side effect data are far from complete. To support this assertion, we calculated the correlation between the number of validated predictions and the length of the textual record in HSDB. For the 201 validated drugs mentioned above, we found a significant correlation between the quality of predictions and the amount of available information (Pearson $r = 0.25$, $p < 2.3e-4$). The correlation increases as more predictions are taken into account (see Figure 2B).



A B

Fig. 2. A blind test. (A) Percentage of new drugs with validated predictions in HSDB. (B) Correlation between number of validated predictions and the amount of information available for corresponding drugs. Dotted lines show the 95% confidence interval.

3.5 Using Side Effect Predictions for Drug Target Elucidation

In a seminal paper, Campillos et al. [2] have shown that drugs with similar side effects are likely to share molecular targets. Exploiting this correlation they were able to predict new targets for drugs. However, their analysis was limited to drugs with known side effects. Our method has the potential to overcome this limitation as long as some molecular data is available on the drug in question.

To demonstrate the utility of our method in drug target elucidation, we applied it to predict the side effects of 4,335 drugs from DrugBank that do not have side effect information in SIDER. We then computed the correlation between two drug similarity matrices: one that is based on comparing the top k predicted side effects (via a Jaccard coefficient), and another that is based on comparing known drug targets (via a Jaccard coefficient). The Pearson correlation between the two similarity matrices varied for varying k, reaching a peak of 0.084 for $k = 13$ (see Figure 3). This correlation was significantly higher than the random expectation (shuffling the drug-target associations while maintaining the same number of associated targets per drug). Expectedly, the correlation was lower than that observed for the drugs whose side effects are known (from SIDER).

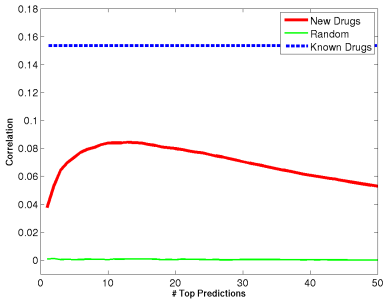


Fig. 3. Correlations of side-effect-based and target-based similarities.

4 Conclusions

Our contribution in this paper is three fold: (i) We show that computational prediction of side effects of drugs is possible. We present an approach that combines correlation based analysis with network diffusion, achieving very high retrieval accuracy. In cross validation we are able to accurately predict side effects for up to two thirds of the drugs; in a blind test we are able to confirm our predictions for almost half of the drugs. (ii) We demonstrate the use of different data sets, such as chemical structure and cell line response, for the prediction task. The use of different data sets could potentially increase the sensitivity and specificity of the predictions. (iii) We find a significant correlation between the similarity of the predicted side effects of drugs and their targets, indicating the potential utility of our algorithm in drug target identification.

Several extensions of our work are possible. The CCA algorithm that we presented is limited to the analysis of one descriptive data set at a time. It is possible that using generalized canonical correlation analysis one could extend the method to take into account multiple data sets. The descriptive data used came from two sources: chemical structure information and cell line response data. Other sources of descriptive data could be used, most notably gene expression data in response to drug treatment such as those cataloged by the Connectivity Map project [1].

In summary, we believe that our algorithm constitutes a first step toward shortcutting the process of side effect identification in the development of new drugs.

References

1. Lamb, J., Crawford, E., Peck, D., Modell, J., Blat, I., Wrobel, M., Lerner, J., Brunet, J., Subramanian, A., Ross, K., Reich, M., Hieronymus, H., Wei, G., Armstrong, S., Haggarty, S., Clemons, P., Wei, R., Carr, S., Lander, E., Golub, T.: The connectivity map: using gene-expression signatures to connect small molecules, genes, and disease. Science **313**(5795) (2006) 1929–35

2. Campillos, M., Kuhn, M., Gavin, A., Jensen, L., Bork, P.: Drug target identification using side-effect similarity. Science **321**(5886) (2008) 263–6
3. Billingsley, M.: Druggable targets and targeted drugs: enhancing the development of new therapeutics. Pharmacology **82**(4) (2008) 239–44
4. Moore, T., Cohen, M., Furberg, C.: Serious adverse drug events reported to the food and drug administration, 1998-2005. Arch Intern Med **167**(16) (2007) 1752–9
5. Xie, L., Li, J., Bourne, P.: Drug discovery using chemical systems biology: identification of the protein-ligand binding network to explain the side effects of cetp inhibitors. PLoS Comput Biol **5**(5) (2009) e1000387
6. Need, A., Motulsky, A., Goldstein, D.: Priorities and standards in pharmacogenetic research. Nat Genet **37**(7) (2005) 671–81
7. Oti, M., Snel, B., Huynen, M., Brunner, H.: Predicting disease genes using protein-protein interactions. J Med Genet **43**(8) (2006) 691–8
8. Franke, L., van Bakel, H., Fokkens, L., de Jong, E., Egmont-Petersen, M., Wijmenga, C.: Reconstruction of a functional human gene network, with an application for prioritizing positional candidate genes. Am J Hum Genet **78**(6) (2006) 1011–25
9. Kohler, S., Bauer, S., Horn, D., Robinson, P.N.: Walking the interactome for prioritization of candidate disease genes. Am J Hum Genet **82**(4) (Apr 2008) 949–958
10. Vanunu, O., Sharan, R.: A propagation-based algorithm for inferring gene-disease assocations. In: German Conference on Bioinformatics. (2008) 54–52
11. Wu, X., Jiang, R., Zhang, M.Q., Li, S.: Network-based global inference of human disease genes. Mol Syst Biol **4** (2008) 189
12. Yildirim, M.A., Goh, K.I., Cusick, M.E., Barabasi, A.L., Vidal, M.: Drug-target network. Nat Biotechnol **25**(10) (Oct 2007) 1119–1126
13. Yang, L., Xu, L., He, L.: A citationrank algorithm inheriting google technology designed to highlight genes responsible for serious adverse drug reaction. Bioinformatics **25**(17) (Sep 2009) 2244–2250
14. Kutalik, Z., Beckmann, J.S., Bergmann, S.: A modular approach for integrative analysis of large-scale gene-expression and drug-response data. Nat Biotechnol **26**(5) (May 2008) 531–9
15. Leurgans, S.E., Moyeed, R.A., Silverman, B.W.: Canonical correlation analysis when the data are curves. Journal of the Royal Statistical Society. Series B (Methodological) **55**(3) (1993) 725–740
16. Wolf, L., Donner, Y.: An experimental study of employing visual appearance as a phenotype. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. (2008) 1–7
17. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Scholkopf, B.: Learning with local and global consistency. In: Advances in Neural Information Processing Systems 16, MIT Press (2004) 321–328
18. Lin, W.H., Hauptmann, A.: Merging rank lists from multiple sources in video classification. In: Proc. IEEE International Conference on Multimedia and Expo ICME '04. Volume 3. (2004) 1535–1538 Vol.3
19. Kuhn, M., Campillos, M., Letunic, I., Jensen, L., Bork, P.: A side effect resource to capture phenotypic effects of drugs, submitted. Available at: `http://sideeffects.embl.de/`
20. Wheeler, D.L., Barrett, T., Benson, D.A., Bryant, S.H., Canese, K., Chetvernin, V., Church, D.M., Dicuccio, M., Edgar, R., Federhen, S., Feolo, M., Geer, L.Y., Helmberg, W., Kapustin, Y., Khovayko, O., Landsman, D., Lipman, D.J., Madden,

T.L., Maglott, D.R., Miller, V., Ostell, J., Pruitt, K.D., Schuler, G.D., Shumway, M., Sequeira, E., Sherry, S.T., Sirotkin, K., Souvorov, A., Starchenko, G., Tatusov, R.L., Tatusova, T.A., Wagner, L., Yaschenko, E.: Database resources of the national center for biotechnology information. Nucleic Acids Res **36**(Database issue) (Jan 2008) D13–D21
21. Steinbeck, C., Han, Y., Kuhn, S., Horlacher, O., Luttmann, E., Willighagen, E.: The chemistry development kit (cdk): an open-source java library for chemo- and bioinformatics. J Chem Inf Comput Sci **43**(2) (2003) 493–500
22. Steinbeck, C., Hoppe, C., Kuhn, S., Floris, M., Guha, R., Willighagen, E.L.: Recent developments of the chemistry development kit (cdk) - an open-source java library for chemo- and bioinformatics. Curr Pharm Des **12**(17) (2006) 2111–2120
23. Wishart, D.S., Knox, C., Guo, A.C., Cheng, D., Shrivastava, S., Tzur, D., Gautam, B., Hassanali, M.: Drugbank: a knowledgebase for drugs, drug actions and drug targets. Nucleic Acids Res **36**(Database issue) (Jan 2008) D901–D906
24. Wishart, D.S., Knox, C., Guo, A.C., Shrivastava, S., Hassanali, M., Stothard, P., Chang, Z., Woolsey, J.: Drugbank: a comprehensive resource for in silico drug discovery and exploration. Nucleic Acids Res **34**(Database issue) (Jan 2006) D668–D672
25. Wexler, P.: Toxnet: an evolving web resource for toxicology and environmental health information. Toxicology **157**(1-2) (Jan 2001) 3–10

Appendix

Solving the CCA Optimization Problem

Given two descriptive matrices E and R, CCA aims at finding two projection matrices W_E and W_R so that the following correlation is maximized:

$$\begin{aligned} &\max_{W_E,W_R} \operatorname{Tr}\left(W_E^T E R^T W_R\right), \qquad \text{subject to} \\ &W_E^T E E^T W_E = W_R^T R R^T W_R = I \end{aligned} \tag{8}$$

Denote $C_{EE} = EE^T, C_{ER} = ER^T, C_{RE} = RE^T$ and $C_{RR} = RR^T$. Consider first the case where each of the projection matrices is a single vector, and define the following optimization problem:

$$\max_{w_e,w_r} \frac{w_e^T C_{ER} w_r}{\sqrt{w_e^T C_{EE} w_e \cdot w_r^T C_{RR} w_r}} \tag{9}$$

Since the expression to optimize is invariant under scaling of the projections w_e and w_r, one can fix the two terms in the denominator to 1 and optimize the numerator. The resulting Lagrangian is:

$$\mathcal{L}\left(\lambda_e, \lambda_r, w_e, w_r\right) = w_e^T C_{ER} w_r - \frac{\lambda_e}{2}\left(w_e^T C_{EE} w_e - 1\right) - \frac{\lambda_r}{2}\left(w_r^T C_{RR} w_r - 1\right)$$

Taking derivatives and comparing to zero we find that $\lambda_e = \lambda_r = \lambda$ and, consequently, that w_r can be expressed as:

$$w_r = \frac{C_{RR}^{-1} C_{RE} w_e}{\lambda} \tag{10}$$

and that w_e is the solution to the generalized eigen problem:

$$C_{ER}C_{RR}^{-1}C_{RE}w_e = \lambda^2 C_{EE}w_e \tag{11}$$

Let W_R be the matrix whose columns are the vectors solving Eq. 10, and let W_E be the matrix whose columns are eigenvectors solving Eq. 11. Then

$$\begin{aligned}\mathrm{Tr}\left(W_E^T C_{ER} W_R\right) &= \sum_{i=1}^{k} w_{e,i}^T C_{ER} w_{r,i} \\ &= \sum_{i=1}^{k} \frac{w_{e,i}^T C_{ER} C_{RR}^{-1} C_{RE} w_{e,i}}{\lambda_i} \\ &= \sum_{i=1}^{k} \frac{\lambda_i^2 w_{e,i}^T C_{EE} w_{e,i}}{\lambda_i} = \sum_{i=1}^{k} \lambda_i\end{aligned}$$

Thus choosing eigenvectors corresponding to the k largest eigenvalues will maximize the objective of Eq. 1.

It remains to show that this solution respects the optimization constraints. The constraints of the Lagrangian ensure that the entries along main diagonal of $W_E^T EE^T W_E$ and $W_R^T RR^T W_R$ are equal to one. To show that the off-diagonal elements of these matrices are zero, we apply the Cholesky decomposition to C_{EE} and C_{RR} (both are symmetric): $C_{EE} = L_{EE}L_{EE}^T$ and $C_{RR} = L_{RR}L_{RR}^T$. Denoting $u_e = L_{EE}^T w_e$ and $A = L_{EE}^{-1} C_{ER} \left(L_{RR}^T\right)^{-1}$, we can reformulate Eq. 11 as a standard eigen problem:

$$L_{EE}^{-1} C_{ER} \left(L_{RR}^T\right)^{-1} L_{RR}^{-1} C_{RE} \left(L_{EE}^T\right)^{-1} u_e = AA^T u_e = \lambda^2 u_e \tag{12}$$

As AA^T is symmetric, its eigenvectors u_e are orthogonal, implying that for $i \neq j$: $w_{e,i}^T EE^T w_{e,j} = w_{e,i}^T L_{EE} L_{EE}^T w_{e,j} = u_{e,i}^T u_{e,j} = 0$.

In the regularized version of CCA, the terms C_{EE} and C_{RR} in Eq. 9 are replaced with

$$\begin{aligned}C_{EE}^* &= \left(EE^T + \eta\lambda_E I\right) \\ C_{RR}^* &= \left(RR^T + \eta\lambda_R I\right)\end{aligned} \tag{13}$$

Computing a Side Effect Vector with Highest Correlation

We wish to efficiently compute the vector $f = \left(W_E^T\right)^\dagger q_{proj}$. Using the notation above, $u_e = L_{EE}^T w_e$, and in matrix form, $U_E = L_{EE}^T W_E$. Substitute that into the equation above we get:

$$f = \left(\left(L_{EE}^T\right)^{-1} U_E\right)^\dagger q_{proj} \tag{14}$$

Since L_{EE}^T is invertible, the pseudoinverse of $(L_{EE}^T)^{-1}$ is L_{EE}^T. Since U_E has linearly independent columns, its pseudoinverse is equal to $\left(U_E^T U_E\right)^{-1} U_E^T$. It

follows that

$$f = \left(U_E^T U_E\right)^{-1} U_E^T L_{EE}^T q_{proj} = U_E^T L_{EE}^T q_{proj}$$
$$= L_{EE} U_{EE} q_{proj} = C_{EE} W_E q_{proj}$$

Thus f can be computed using simple matrix multiplication.