Team Prospectors: Ensemble based semi-supervised approach to IDG-DREAM Drug-Kinase Binding Prediction Challenge

Davor Oršolić, Bono Lučić, Višnja Stepanić, Tomislav Šmuc Ruđer Bošković Institute, Zagreb, Croatia

Background

We have provided submissions for both Round 1 and Round 2 based on IDG-DREAM Drug-Kinase Binding Prediction Challenge Data: 14,492 drugs and 1,462 proteins. Both rounds were addressed by ensemble based models (Random Forest and Boosting trees), with the main effort put into feature set construction, instance selection and model selection. Final Round 2 model has been produced using *XGBoost* algorithm and feature set based on selected similarities in compound and protein space [6].

Methods

In the exposition of the methodology used for this challenge we refer to the original training set as to the compound-target interactions with valid, known Kd values, available for all challenge participants via Synapse challenge portal. The term training set is also used when referring to particular subsets of this original training set used for model optimization/selection, or developing specific models for the submissions in Round 2. We use term validation set which typically means test set obtained by training/validation set partitioning from the original training set in order to be able to test and select/optimize models. When referring to test set we usually refer to Round 1 or Round 2 test set – used for scoring submissions.

Data preprocessing & Feature engineering

For the prediction of drug-target interactions of given pairs we first retrieved protein FASTA sequences and SMILES representations of compounds from UniProt and ChemSpider, respectively. Some effort has been spent on data cleansing and preprocessing. Feature set describing compounds and targets for the initial round of modeling (Round 1) were constructed using *rcdk* and *protr* R packages, respectively [4][7]. We experimented with different subsets of compound features, but found out that maccs fingerprints gave best results. For testing and optimization of the modeling workflow we split training set of drug-kinase interactions into 70/30 ratio. In later stages (Round 1 and Round 2) we completely changed the feature set and used similarities in compounds and targets space in order to describe drug-target pair. To estimate similarities in target space, EMBOSS program with *needleall* application was used to globally align all pairs of primary protein structures [8]. To align protein sequences by the EMBOSS program, Needleman-Wunsch algorithm was used, and the EBLOSUM62 matrix for calculation of similarities, together with other default settings. Compound similarities were based on precalculated *maccs* fingerprints from R's rcdk package depending on *kekule* SMILES representation [4]. To determine similarities between all compounds, *fingerprint* package was utilized and Tanimoto coefficients were calculated based on comparison of 166 maccs keys for each of 14492 compounds [5].

Modeling

Round 1

In our Round 1b experiments we investigated different feature sets and used random training/validation splitting (70/30) of the original dataset of compound-target interaction pairs in order to improve performance of the predictive models. We have tested different types of compound descriptors, including similarities. First submitted models in

Round 1b were based on maccs fingerprints (166 features) and protein physical descriptors (41 features). We had also introduced similarity based approach in few of our Round 1b models which showed improvement in comparison with our first submission.

Round 2

In Round 2 modeling we focused more on similarity based representation of the problem. We also introduced new training/validation set definition in order to use more representative (similar) instances with respect to the interactions from the Round 2 scoring test set. For that purpose we first clustered joint training set with Round 2 test set - using *hclust* algorithm from stats package in R - into 100, 200, 400 clusters (using target and compound similarity matrices) [1]. We than formed reduced training/validation set from only those compound-target interactions that had compounds clustered together with test set compounds and targets clustered together with test set targets. We used clustering results to redefine similarity feature sets, too. Similarity feature sets were based on cluster representatives from compound space and cluster representatives from the target space, which were used as „anchors“ for similarity features on which we regressed instances from the training set. Using the optimized *XGBoost* scheme we tested models using following compound + target similarity feature sets (100+100, 200+200, 400+400)[6]. The training set with 200+200 similarity feature sets and 13,786 compound-target interactions was used to train the models for the first Round 2 submission.

Final submission - Compound-target selection for the training set

Final submission for the Round 2 and the best result was the *XGBoost* model which was trained on the training set based on compound-target interactions for which targets are one of the test targets or the targets that were clustered together with some of the test targets. This subset of interactions was further filtered to only those that have compounds clustered together with test set compounds.

Final submission - Feature construction

For the final submission we used the „anchors“ for similarity features, a subset of test targets, as well as targets from target-clusters containing test set targets which are not available in the training set. Similarly compound similarity feature set was based on the partitioning of the compound space into clusters. For compound "anchors" for similarity we used most similar compounds from clusters containing compounds from the test set (with avg.similarity>0.5). This meant that our final model was trained using 7,336 compound-interaction pairs and 207 + 194 similarities as features, from compound and target similarity matrices respectively.

Algorithms and model selection/optimization

During the course of the two rounds of the challenge we experimented with two ensemble based algorithms: we started using Random Forest (*randomForest* R package) and Round 1 submissions were based on the models produced using RF [3]. The models were based on 500 and 2000 trees, respectively, controlling for the depth/complexity of the trees by limiting the size of terminal nodes to 80 samples. For the Round 2 we used Boosting trees – or *XGBoost* algorithm implementation of R package [6]. We optimized the algorithm parameters using *train* function from *caret* package [9].

The tuning parameter grid had the following parameters: *max_depth* (maximum tree depth, default: 6) *eta* (learning rate) *gamma* (used for regularization tuning) *colsample_bytree* (column sampling, default: 1) *subsample* (row sampling, default: 1) *min_child_weight* (minimum leaf weight, default:1) This parameter optimization was performed on the reduced training set (6,450 pairs) and using 3-fold cross validation.

Discussion

We unfortunately entered the challenge very late, and have managed to produce first models on time before Round 1b was closed. This models were based on simple set of features describing compounds and targets, and large portion of the original training set was used for the model development. Our results in Round 1 were of low quality (Spearman correlation < 0.1; AUC ~ 0.55; RMSE ~ 1.1). In Round 2 we started to experiment more with compound/target similarities as features upon which regression ensemble models were based. We also used clustering of training/test interaction pairs that served several purposes: (i) as the way to extract more meaningful training and validation set for the model development; (ii) try to focus our model on the instances in the neighborhood of test set instances; (iii) use clusters as means for feature selection, as we used similarity matrices in compound/target space as features to make regression models. Our final Round 2 submission results were (Spearman correlation=0.296; AUC=0.685; RMSE=1.196) which represented significant improvement from the Round 1 results. Our findings from the Round 2 experiments show that the approach based on similarities is promising approach for the treatment of this type of the problem (large and diverse set of compounds and targets), and that learning methodology should be capable to capture highly non-linear and very localized interactions – in that respect learning models based on smaller number of samples in close proximity of test samples (learning in the neighborhood – or proximity of the actual tested interaction pairs) is better than learning from large, non-localized training set.

Authors Statement

DO implemented the code and performed calculations. TS, BL, VS, conceived and supervised the study. TS and DO wrote the text (write-up).

Command required to run the docker container:

```
docker run -it --rm -v ${PWD}/io:/input -v ${PWD}/io:/output
docker.synapse.org/syn18553372/prospectors_idg:9686257
```

References

[1] R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/. [2] Breiman, L. (2001). "Random forests." Machine learning 45.1: 5 32 [3] Liaw, A. and Wiener, M. (2002). Classification and Regression by randomForest. R News 2(3), 18--22. [4] Guha, R. (2007). 'Chemical Informatics Functionality in R'. Journal of Statistical Software 6(18) [5] Guha R. (2006). fingerprint: Functions to Operate on Binary Fingerprint Data. R package version 2.2, URL http://CRAN.R-project.org/. [6] Chen, T. and Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). ACM, New York, NY, USA, 785-794. DOI: https://doi.org/10.1145/2939672.2939785 [7] Nan Xiao, Dong-Sheng Cao, Min-Feng Zhu, and Qing-Song Xu. (2015). protr/ProtrWeb: R package and web server for generating various numerical representation schemes of protein sequences. Bioinformatics 31 (11), 1857-1859. [8] Rice, P., Longden, I. and Bleasby, A. (2000). EMBOSS: The European Molecular Biology Open Software Suite. Trends in Genetics. 16(6):276-277 [9] Kuhn, M. (2008). Caret package. Journal of Statistical Software, 28(5)