Drug Discovery: Predicting Molecular Activity with Deep Learning

Introduction

During the drug discovery process, there is a need to discern how well a particular molecule will affect different biological targets. Specifically, for a given molecule can one discern how well it will interact with intended and unintended targets through deep learning? Where the targets can be a range of different molecules in the human body such as proteins or neurotransmitters. A specific use case of this might be in the development of drugs which treat mental illnesses. Often it is the case that for these drugs there is a need to have an interaction with a specific neurotransmitter (say dopamine) and not others (say serotonin).

The client is Merck pharmaceutical company. With the successful information obtained through deep learning the company may be able to more efficiently test new drugs. This would in turn reduce their R&D costs in drug production. Further, the current risk to drug companies for developing new drugs is large. It takes an average of 12 years and an average cost of $2.7 billion to bring a new drug to market, and there is no guarantee of it being profitable. Therefore, by reducing the R&D costs, the risk undertaken when developing new drugs can be lowered tremendously. Not only does this benefit the pharma company, but also the lives of patients who could benefit from the new drugs.

The following is a rough work flow by which the data will modelled:

- Data will be gathered and cleaned. Accounting for any differences in units, missing information, outliers, etc.
- Preliminary investigation will be done to check for abnormalities as well as to obtain a better picture/understanding for the general mathematical structure of the data.
- Some feature engineering and feature selection will be done. Since the dataset is large, there is a high likelihood many of the features will not be useful for learning and may lead to overfitting.
- Simple machine learning models will be run to setup a baseline for future models. Probably regression will play an important role here and may be used before implementation of neural networks.
- For advanced classification, neural networks will be utilized. This will be done using Keras and TensorFlow. Initial work and implementation will be done in

Keras and follow up will be done in TensorFlow. Some iteration will be done to try and find the most optimal model.

Data sources and cleaning

The data source being used was found entirely in the Merck Molecular Kaggle Activity Challenge webpage.It consists of the following: In the training set there are 15 csv files in the following format
Column 1: Molecule ID - it should be noted that this is an anonymous identifier made for the competition. Consequently, no knowledge of the actual molecules is known. IDs are in the format of M_1, M_2, etc.
Column 2: Activity - raw activity values which may be in different units depending on measure.
Column 3 - N: Molecular descriptors/features - As the Molecule ID's, this is also anonymous and so true knowledge of the features is given. They are in the format of D_1, D_2, etc.

In short, the 15 csv files show the activity level on a given target (out of 15) for a large number of molecules including some features of the molecules. In total, the training set consists of 2G of data, thus making loading and computations a bit slow on the entirety of the dataset.

Since this data is from a competition and is structured, there are relatively few data cleaning steps. Further, There are no missing values which need to be taken care of. However, there may be some molecules which have very few number of predictors which may need to be considered. These molecules would be unreliable in training unless those few descriptors can truly accurately predict the activity level.

One step which was taken was to determine how sparse the datasets were. Upon investigating the data, it was clear many of the feature entries were zero. Looking at the sparsity of the data can be useful for multiple reasons. If the data is extremely sparse, faster computation may be performed by utilizing numpy's sparse matrices. Knowing how sparse the data is can give some indication of the likelihood of overfitting. Since the data was so sparse, one should be careful in choosing which features to employ with machine learning. The average sparsity, or the ratio of zero values to non-zero values came out to an average of about 0.94. This implies about 94% of the data is zero.

Exploratory analysis

Exploration of the data went in the following manner. First, cleaning the data and noting it's sparsity. As a result of the sparsity, there may be ways to speed up training through numpy sparse matrices.

Second, histograms of the activity levels for each different dataset (different targets) was looked at. This gives some information about the spread of activity level for each of the molecules. There was an interesting spread which seemed to differ noticeably from target to target. That is, there was not a consistent spread for each of the targets. Some of the targets had a large spread, indicating a large feature dependence of the activity level (fig. 1). Where as some were much more localized (fig. 2), indicating the activity level acted as being more independent of the features.

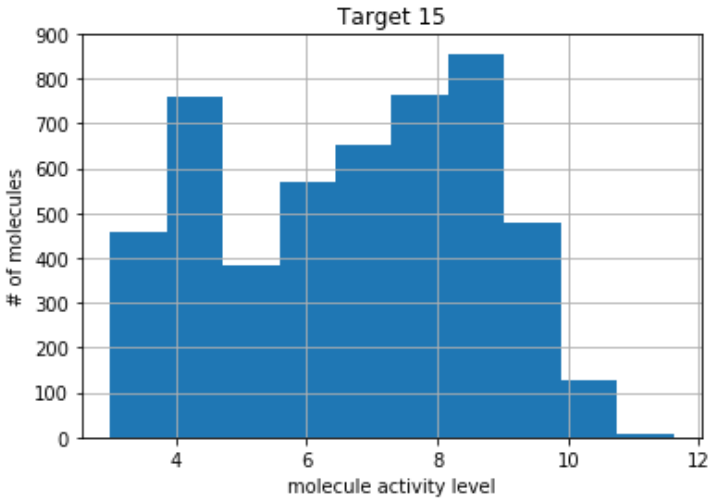

Figure 1: activity level histogram for target 15. The spread is high, indicating the features are strongly correlated to the activity level.

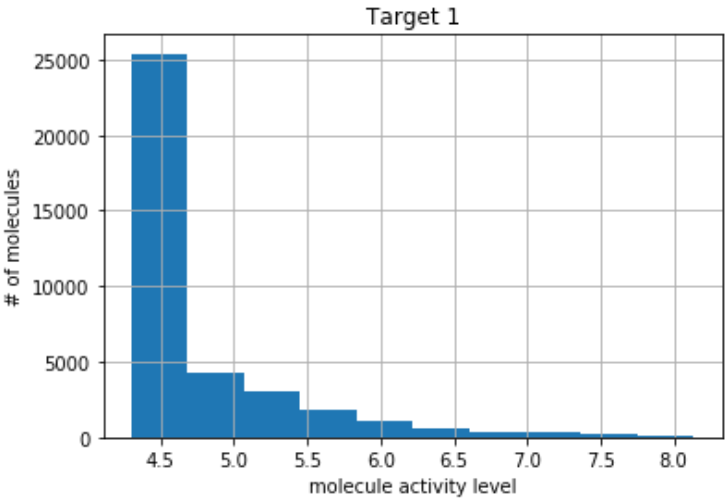

Figure 2: activity level histogram for target 1. The spread is very low, indicating the features have less importance for this target.

The next step in exploration was to determine which (if any) features had a column, but were not actually represented in the data. That is, which features were in the dataset but no molecules had. This was done by simply summing the columns and seeing which were zero. These were then subsequently dropped for the remainder of the analysis.

The last step involved determining the correlation coefficient values for all of the features on the activity level. This is done to determine which features were statistically significant in their correlation with the activity level. A confidence level of 95% was used, and the resulting data frames had thousands fewer features. Interestingly, the highest correlation was found in target 15 with a value of corr = 0.61. Further, the top correlation values in target 15 were all large compared to other targets (fig. 3). This agrees with the observation above, where target 15 had the largest variation in values. This should imply that the correlations would be strongest, which does indeed seem to be the case.

	corr	pval
3151	0.606606	0.0
1493	0.571384	0.0
1492	0.554667	0.0
2418	0.526271	0.0
1491	0.514849	0.0

Figure 3: this table shows the top five correlation values for target 5 between features and the activity level