DREAM Olfaction Prediction Challenge

Bence Szalai[1,2,*], Gábor Turu[1,2*], Péter Várnai[1], László Hunyady[1,2]
[1]Department of Physiology, Faculty of Medicine, Semmelweis University, Budapest, Hungary
[2]Laboratory of Molecular Physiology, MTA-SE, Budapest, Hungary
[*]correspondence: bence.szalai@eok.sote.hu
gabor.turu@eok.sote.hu

Summary Sentence:
We used a linear model using the descriptors provided by organizers and fingerprint similarities between the train set and a set of odor molecules available online, and predicted the perceptual descriptors for each individual by correcting our model on the similarities between the individuals and between the substances.

Background/Introduction:
We used a general linear model in our predictions. To increase the predictive power of our model, we added some modification to the model: we predicted the descriptors for individuals from the weighted average of descriptor values based on the similarity between the subjects. To train our model we also used weighs of training examples based on the similarity between the substances. To predict the average value of a descriptor, we averaged the predicted values for individuals. To predict the standard deviation of a descriptor we have taken advantage of the polynomial relationship between the average value and standard deviation.

Methods:
Data preprocessing:
We used the merged data from train set and leaderboard set to train our final model. Missing values (NaNs) from training set/leaderboard set were omitted.
Feature selection: The features we used for predictions consisted of descriptors provided by organizers and fingerprint similarities. Fingerprints were calculated as Morgan fingerprints with radius value of 5, using rdkit libraries. Similarity features were calculated between the provided molecules and a collection of odorants which consisted of the molecules themselves, and a list of known odor molecules which have been collected from two online sources, http://www.immuneweb.org/articles/fragrancelist.html and www.odour.org.uk. This gave us 2437 features in addition to the descriptors. We also included the squares of the feature values to include some nonlinearity into the regression. The total number of initial features was 14612, and this has been reduced during feature selection. For feature selection we utilized RandomizedLasso algorithm from sklearn library. The feature selection was done on averaged target data. The average target data included 1/1000 dilution only for intensity/strength, and both low and high data averaged for the other descriptors. The descriptor values were omitted if the intensity was zero for the given compound.
Predicting the descriptors for each subject (subchallenge1):
Our linear model was in the form of

$$y_{d,s} = X_d \times W_{d,s}$$

where $y_{d,s}$ denotes the vector of descriptor values for descriptor d and subject s, X_d denotes the matrix of features after feature selection and $W_{d,s}$ represents the linear model. We used linear regression with l2 regularization (Ridge regression to fit our models for each descriptor d and subject s. We fit our regression separately for each test sample, and used Morgan similarities between the given test sample and training examples as weights. We observed that predicting descriptors for individuals was much harder task than predicting the average values. To overcome this problem, we fitted our model not on the original $y_{d,s}$ vector, but on a weighted average of the

49 subjects, $y'_{d,s}$. This weighted average for a subject was calculated the following:

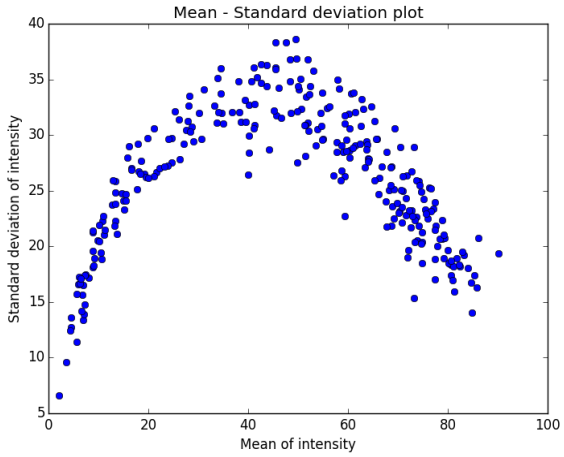
$$y'_{d,s} = \frac{1}{n} \sum_{i=1}^{n} r_{d,i,s} \times y_{d,i}$$

where i denotes the different subjects and $r_{d,i,s}$ denotes the Pearson's correlation coefficient between vectors $y_{d,s}$ and $y_{d,i}$ for a given descriptor d.

These manipulations of the data can lead to changes of the distribution of the predicted values compared to the real values. To overcome this problem, we normalized the predicted values to the distribution of the original individual training data. To do this, first we normalized the predicted values by subtracting the mean and by dividing by the standard deviation. Then we multiplied this normalized vector by the standard deviation of the original training data and added the mean of the training data. This normalization process did not improve predictions for subchallenge1 (as the predictions are evaluated by the correlation between predicted and real values) but was necessary for subchallenge2.

P**redicting the averages and standard deviations of descriptors (subchallenge2):**

To calculate the average value of a given substance and given descriptor, we calculated the average for the 49 subjects from the individual predictions. To get the standard deviation it would have been straightforward to calculate the standard deviation from the values of the 49 subjects. This calculation lead to relatively bad predictions. We hypothesized that subjects' scores for a given descriptor should follow normal distribution. However, since the scores are restricted to values between 0 and 100, low or high population averages lead to distorted distributions which result in decreased detected averages and standard deviations. If we simulate normal distributions where values lower than zero and higher than 100 are set to 0 and 100 respectively, the resulting average vs. standard deviation plots follow parabola-like distributions, and the standard deviation has the maximum at average of 50. Plotting the actual data show very similar relationship (see the figure for intensity data).



We approximated the relationship with a parabola using the following equation:

$$y = \frac{-y_{max}}{2500} \times x^2 + \frac{y_{max}}{25} \times x$$

where y_{max} is the maximal value of the standard deviation. Since only the correlation between the the standard deviation values is evaluated in the challenge , we set y_{max} arbitrary to 1 for all of the descriptors, and calculated the standard deviations based on the averages from equation. This could be corrected with the average standard deviations of the given descriptor if the actual values were addressed.

Conclusion/Discussion: We applied simple Ridge regression model for the prediction of odor descriptors. The prediction was improved by using similarity features, randomized Lasso feature

selection and weighting the training samples with the similarity scores. Instead of using the individual data for training, we decided to average those subjects' scores who had good scoring correlation. The average was weighted by the correlation scores. The reasoning behind this is that only one measurement for each individual with each compound does not provide data with good enough reliability. However, the measured data for each individual is not the only data we have, since we know that every given subject is part of a population. Indeed, correction with the data from the individuals which have similar odor sensing properties, led to better crossvalidation results.
We hypothesized that compounds with similar molecular structure would stimulate similar set of odor receptors and would lead to similar perception by the subjects. Therefore we created a new set of features, and these features consisted of Morgan fingerprint similarities. We calculated the similarities between all the compounds and also included a number of known odorants. The inclusion of similarity features improved the predictions.
The odor molecules have very diverse molecular structures. Its possible that different features might determine the same descriptor values for molecules with very different structure. To include this during the training, we weighted the training samples with their similarity to the given test sample.

References:

We performed our work in Python 2.7 using the following packages:
NumPy, SciPy, Pandas, scikit-learn, rdkit, pubchempy, BeautifulSoup

Authors Statement:

Bence Szalai: Implementing the weighted average for subjects into the linear model and implementing the mean based approximation of standard deviation.
Gábor Turu: Feature selection and implementing the weighting of training data.
Péter Várnai and László Hunyady: Supervising the project