Base-Stacking Prediction Project

Christine Lightfoot and Kristin Barton

December 11, 2019

1 What is the biophysical challenge being added and why is it important?

RNA molecules' three-dimensional structure dictated by their nucleotide sequence can be determined by X-ray crystallography or NMR experiments [1]. Determining an RNA'S structure is still an expensive and complex process, despite recent advances in the field [1]. This has lead scientists to take advantage of technological advances being made in machine learning. Using chemical shift data, we were able to train a model to predict base stacking interactions among ribonucleic acid residues. We were also able to train a regression model to predict solvent accessible surface areas (SASA) from the chemical shift data set.

Residue base-stacking plays a vital role in RNA structural properties and mediates RNA-RNA interactions [3]. An RNA'S three-dimensional structures is represented by a plethora of geometrical properties, including helical parameters base pairing/stacking, hydrogen bonding and backbone conformation [1]. Knowledge of whether RNA residues interact by Watson-Crick or non-Watson-Crick base-pairing and base-stacking can help a scientist find tertiary motifs, or molecular building blocks, within an RNA'S three-dimensional structure [1].

SASAs also play a vital role in determining an RNA'S tertiary structure. Obtaining solvent accessible surface areas of RNA residues in an RNA structure can help to distinguish possible functional sites and core structural regions [4]. Tertiary structure prediction is vital for understanding structure–function relationships for ribonucleic acids whose structures are unknown [2].

2 What is state of the art?

Our new model combines the results of multiple baseline models by taking the average of their y-values. We are interested to see if the model featuring the combined y-values performs more optimally than the individual baseline models.

3 How did you train your baseline and deep neural model?

3.1 Base stacking

Our model involved six baseline models and one combined model to compare them to. The baseline models were: multilayer perceptron classifier, linear discriminant analysis, K neighbors classifier, Gaussian naive Bayes, decision tree classifier, and C-support vector classifier.

A multilayer perceptron classifier is a single neuron model with the goal of developing a robust algorithm and data structure to model difficult problems, such as base-pairing/stacking. Linear discriminant analysis is a dimensionality reduction technique used for modeling differences in groups, or separating two two or more classes. A k neighbors classifier is a model that aims to implement learning based on the nearest k neighbors of each point. Gaussian naive Bayes provides a way that we can calculate the probability of a hypothesis given our prior knowledge. A decision tree classifier is a supervised machine learning model where the data is continuously split according to a certain parameter. Lastly, a C-support vector classifier is another supervised machine learning algorithm where each data point is plotted as a point in an n-dimensional space with each feature's value having a particular coordinate.

For the multilayer perceptron classifier, a randomized search was used to find the best parameters from a preset parameter space. All other models used the default hyperparameters.

The deep neural model we want to compare these baseline models to was obtained by averaging the predicted y values of all the baseline models. To classify as 0 or 1, the average y values underwent a simple rounding (where 0.5 and under becomes 0, above 0.5 becomes 1).

The training algorithm for all models was the same. First, a subset of data corresponding to one ID is removed to be used as testing set. The rest of the data is used to fit the model and the testing set is used to determine the f1 score of the model. Then, the algorithm loops so that the next ID dataset is removed for testing, and so on. We looped through every ID and averaged all of the f1 scores for each of the 7 total models.

3.2 SASA

This followed a similar process: the baseline models are a multilayer perceptron regressor with hyperparameters optimized through a randomized search of the parameter space, Multitask lasso, K-Neighbors regressor, gaussian process regressor, and decision tree regressor. There are two final models:

First is a Combined Model that averages the results of the predicted y-values of all the other 5 models.

Second is a Reduced Combined Model that only includes multilayer perceptron, multitask lasso, and K-neighbors regressor. This reduced model removes the gaussian process regressor and decision tree regressor because of how badly they perform as individual models.

4 How does your model compare to baseline models?

4.1 Base stacking

After running the program a few times, there are some mixed results. A few of the models give slightly different results each time, and the combined value is not consistently the best, but it is generally at the higher end of the scores which is promising. Three trials are listed below. The decision tree classifier and the multilayer perceptron classifier both have different values for each run. This results in slightly different combined values. There was one instance in the three trials where the combined did in fact perform better than any of the individual models.

Trial	1	2	3
MLP	0.8187	0.8383	0.8269
LDA	0.8400	0.8400	0.8400
K-NC	0.8382	0.8382	0.8382
GNB	0.7855	0.7855	0.7855
DTC	0.7974	0.8033	0.8030
C-SVC	0.8117	0.8117	0.8117
Combined	0.8374	0.8402	0.8394

4.2 SASA

After 3 runs with the SASA prediction code, it is clear that the Gaussian Process Regressor and the Decision Tree Regressor do not work at all since their R2 scores are negative. The Combined Model ends up with a negative score as well because of this. The Reduced Combined Model is much more promising. Its value is consistently higher than any of the three models which it is comprised of. While the value is still low, it shows promise for future consideration.

Trial	1	2	3
MLP	0.3318	0.3422	0.3288
MTL	0.3640	0.3640	0.3640
K-NR	0.3625	0.3625	0.3625
GPR	-22.57	-22.57	-22.57
DTR	-0.2590	-0.2880	-0.3155
Combined	-0.5436	-0.5379	-0.5470
Reduced Combined	0.4432	0.4463	0.4418

5 How can your model be improved?

A couple ways we could improve would be the following:

1. Instead of using default values for all models besides MLPClassifier and MLPRegressor, we could optimize more hyperparameters in order to get better models.

2. The models used were chosen randomly. From our trials with SASA, it is clear that careful selection of underlying models can make a substantial difference, so in future models we could choose better models to combine based on the dataset we are using.

References

[1] Bottaro, Sandro, et al. "The Role of Nucleobase Interactions in RNA Structure and Dynamics." *Nucleic Acids Research*. Nucleic Acids Research, vol. 42, no. 21, 2014, pp. 13306–13314., doi:10.1093/nar/gku972.

[2] Hajdin, Christine E et al. "On the significance of an RNA tertiary structure prediction." *RNA (New York, N.Y.)* vol. 16,7 (2010): 1340-9. doi:10.1261/rna.1837410

[3] Schulz, E.C., Seiler, M., Zuliani, C. et al. Intermolecular base stacking mediates RNA-RNA interaction in a crystal structure of the RNA chaperone Hfq. Sci Rep 7, 9903 (2017) doi:10.1038/s41598-017-10085-8

[4] Yang, Yuedong, et al. "Genome-Scale Characterization of RNA Tertiary Structures and Their Functional Impact by RNA Solvent Accessibility Prediction." *Rna* , vol. 23, no. 1, Feb. 2016, pp. 14–22., doi:10.1261/rna.057364.116.