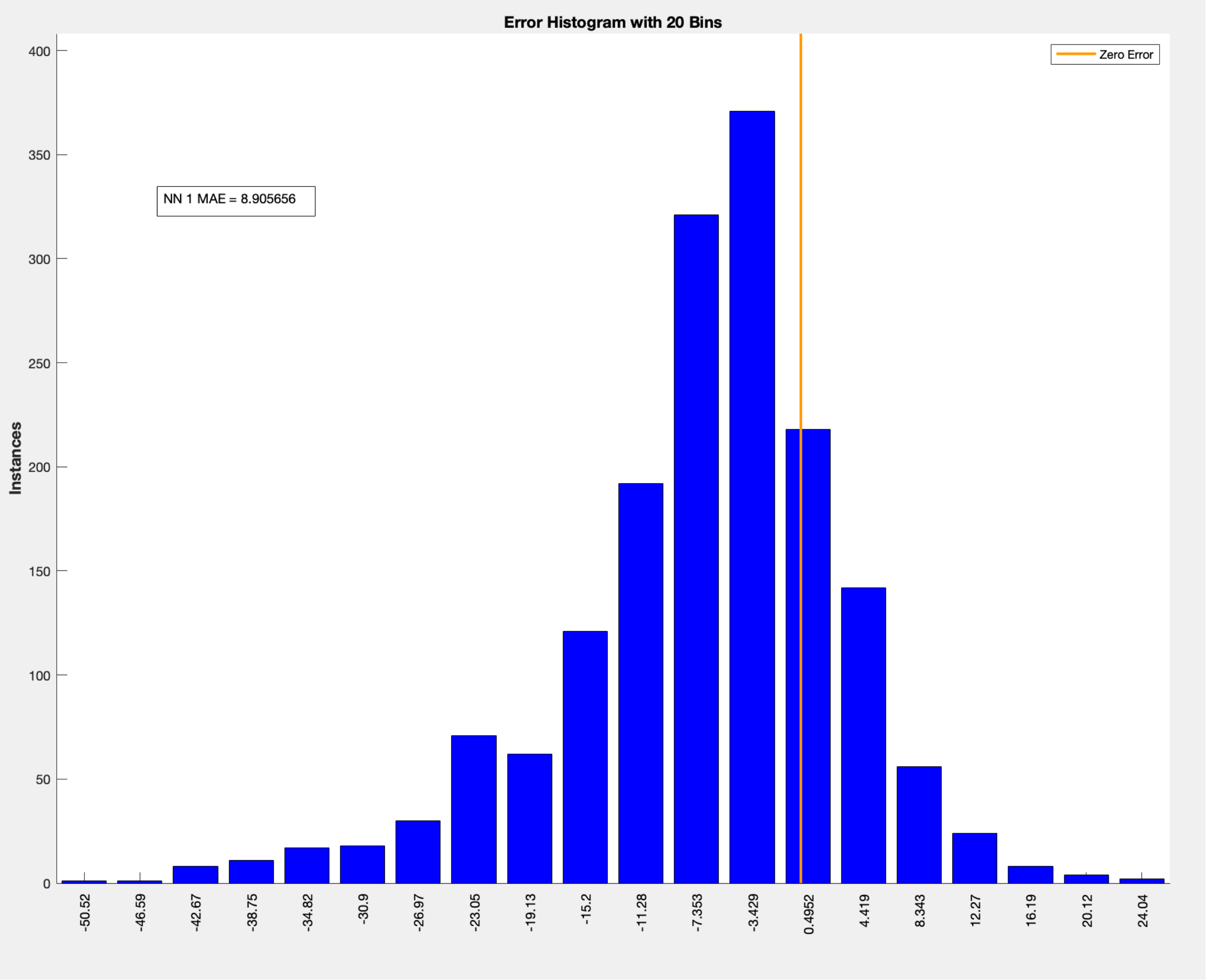
ENAS 991: Assignment 3
(Writeup)

Karthik Desingu

Problem 2

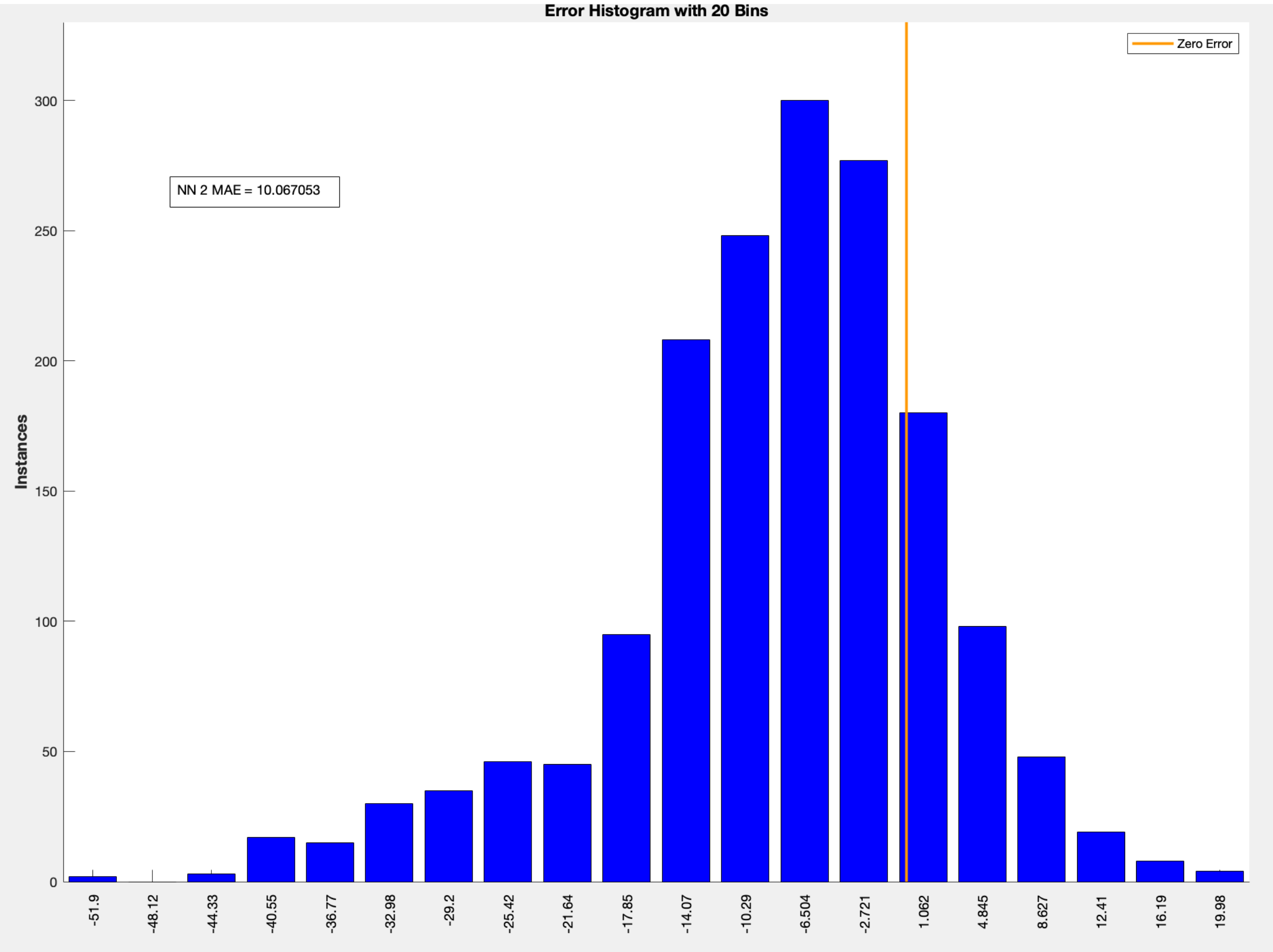
NN Configuration 1

Number of Hidden Layers	1
Size of Hidden Layers	10
Test MAE	8.906



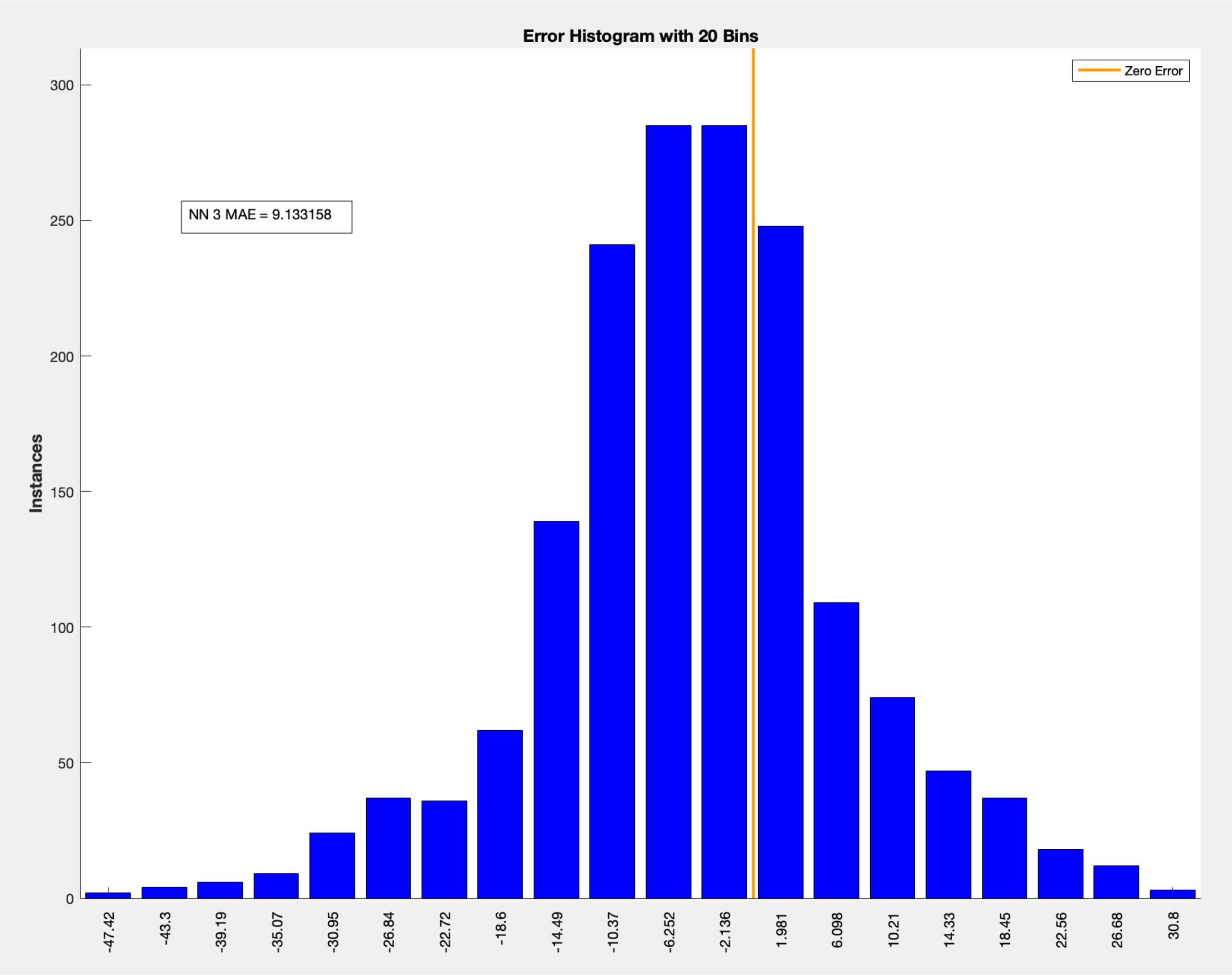
NN Configuration 2

Number of Hidden Layers	1
Size of Hidden Layers	20
Test MAE	10.067



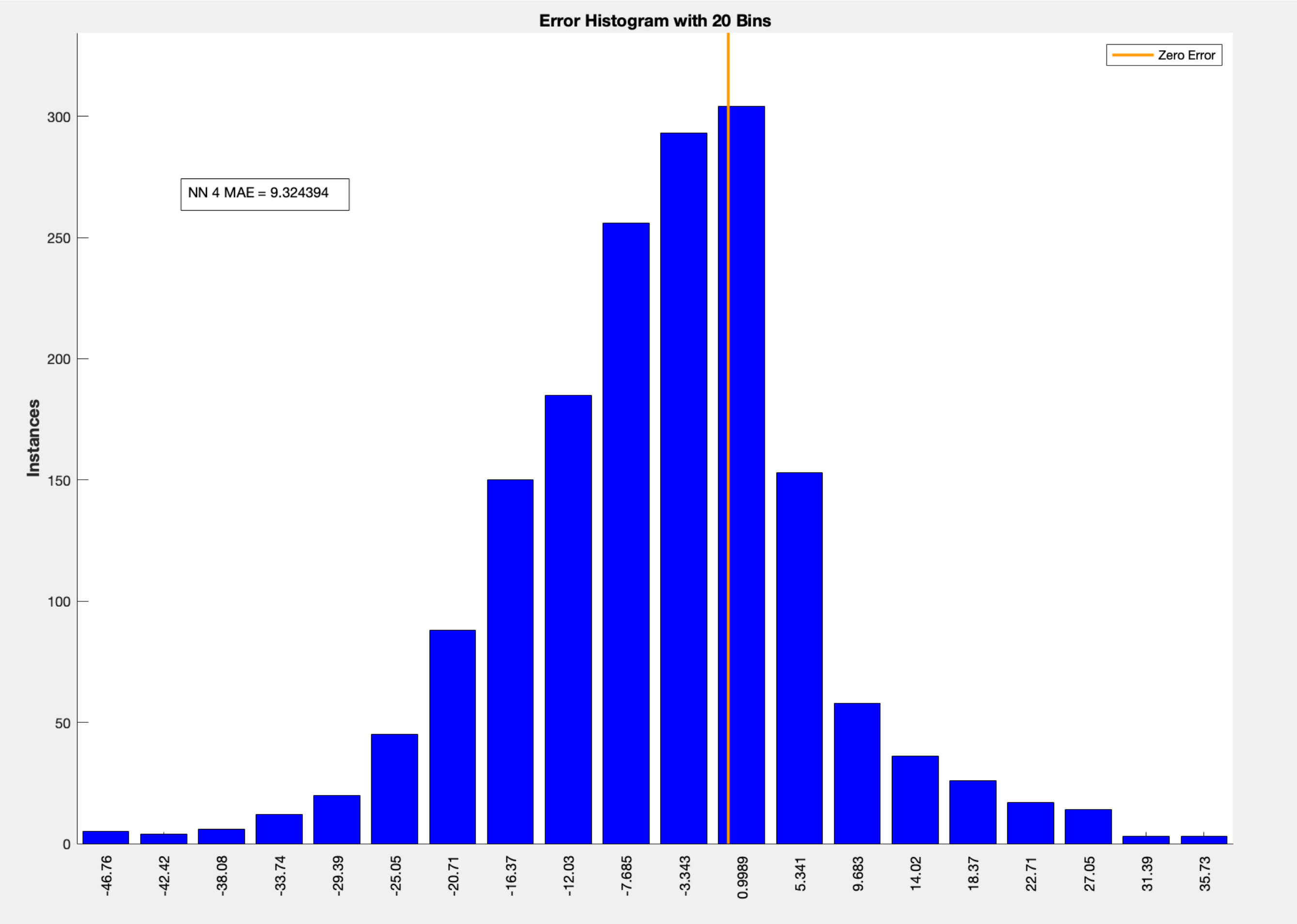
NN Configuration 3

Number of Hidden Layers	2
Size of Hidden Layers	15 → 5
Test MAE	9.133



NN Configuration 4

Number of Hidden Layers	2
Size of Hidden Layers	20 → 10
Test MAE	9.324



Four different architectures were tried: two with 2 hidden layers and two with 1 hidden layer each. The simplest architecture (NN 1) with a single 10-neuron hidden layer performs best, followed by the one with two hidden layers of 15 and 5 neurons each.

This observation is consistent with the arguments made in the following sections that the given dataset has highly correlated features, that also correlate well with the target variable itself. Hence, they have strong linear relationships, and simple NN architectures suffice to translate input features into the target variables. NN 2 is complex due to 20 neurons – which is over double the number of features. NN 4 is complex for the same reason, but the addition of a second hidden layer with only 10 features potentially simplifies the feature representation right before regression. Both NN 1 and NN 3 are relatively simpler and are able to model the (mostly) linear relationship well.

(2)

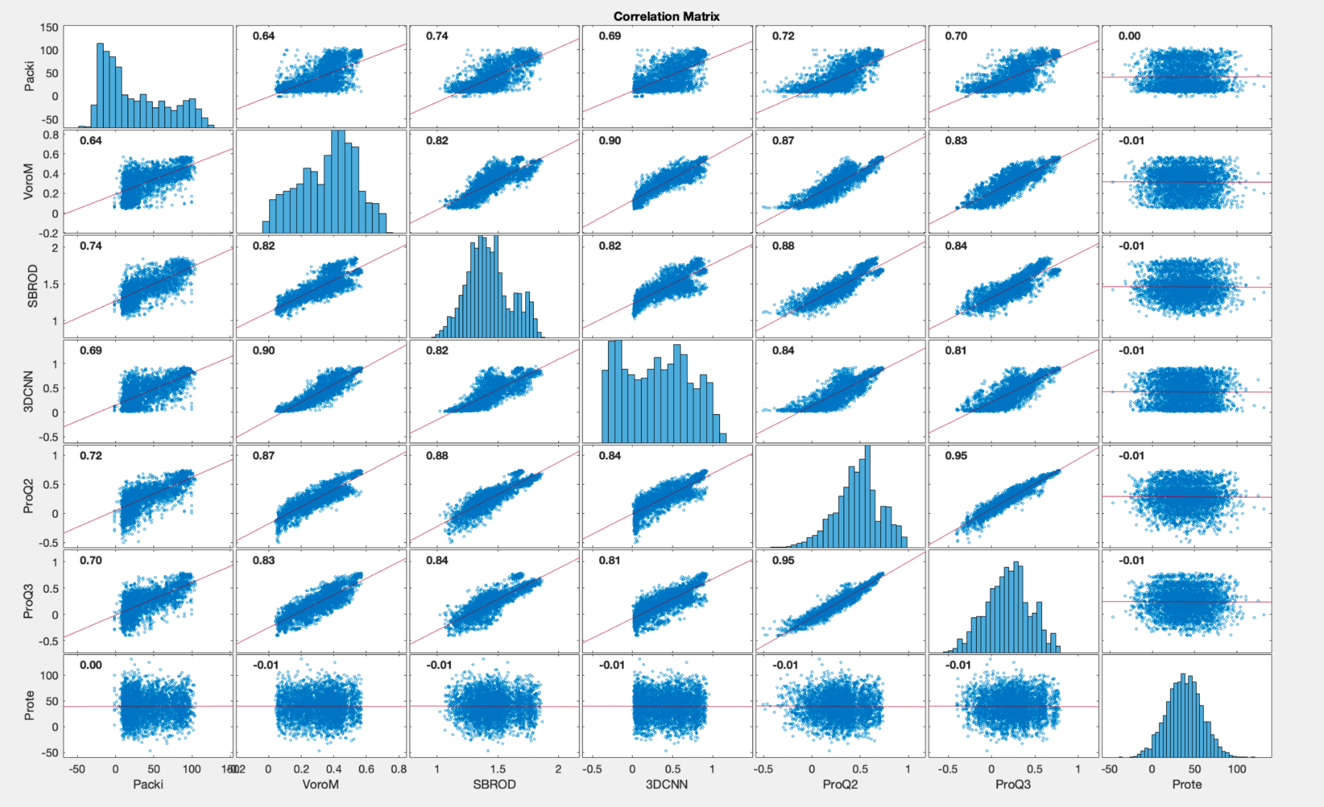
The error histogram shows a distribution that is densely distributed close to zero error, but slightly skewed toward the negative side. A direct interpretation would be that the trained NN generally tends to predict a GDT that is lower than the actual GDT value.

The fact that the histogram is more dense close to zero error is a sign that the model is performing quite well, making erroneous predictions only for some samples, while for most samples, it predicts with low error.

Furthermore, a single peak in the histogram suggests that the model is not finding or falling into local optima when training the model; if it was, then there would more than one peak in the histogram suggesting that the model has found multiple potential relationships between the features and the target variable, some of which are likely to be incorrect (local optima).

(3)

Correlation Plots of Features



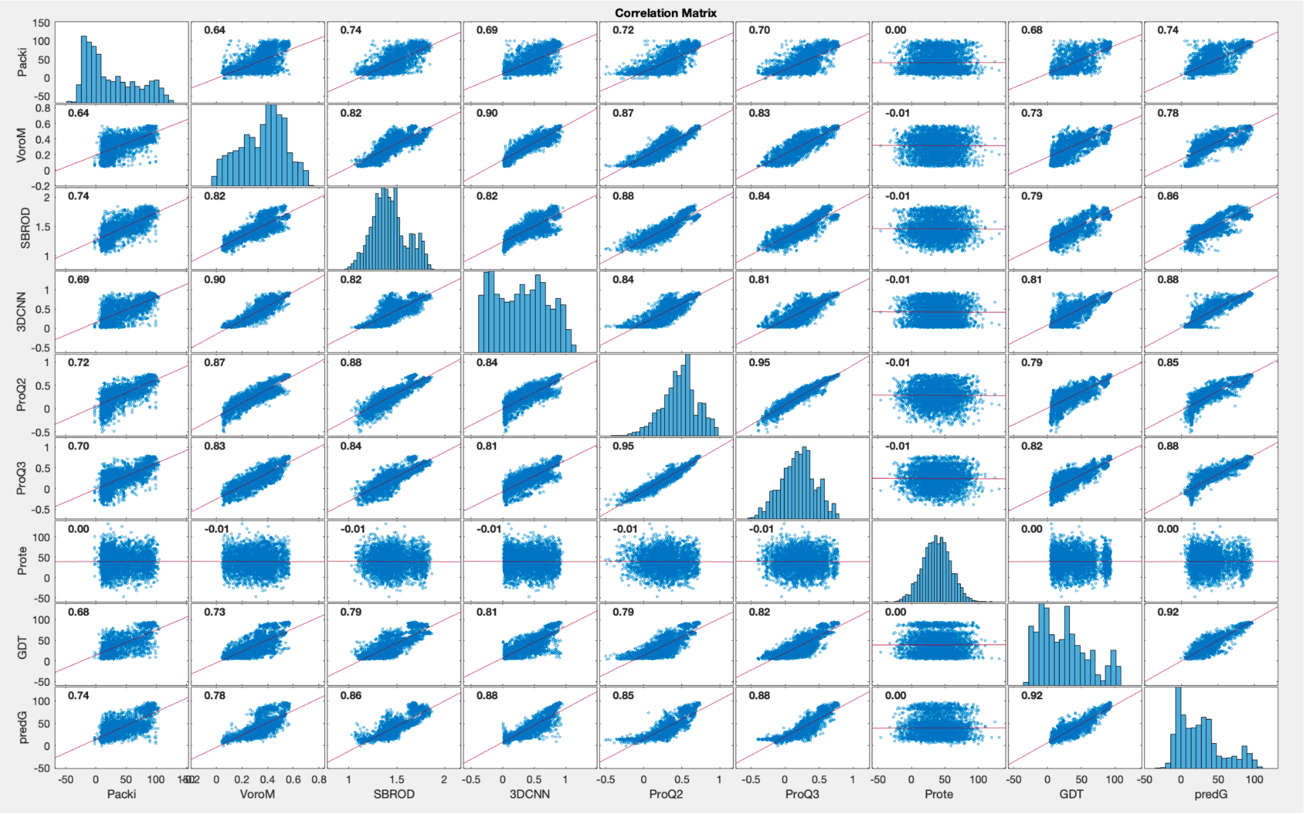
From the correlation plots, it is apparent that the **Protein Pro** feature does not correlate with any of the other features, while all other features are quite correlated with each other to a significant extent.

It would appear that the **Protein Pro** scores are fake data. Since the scores in this dataset essentially represent decoy detection scores, each computed through a different approach, they must be somewhat correlated with each other.

This is because each score in some sense tries to identify the decoy. Hence, for the same target structure, each method would tend to show a similar scoring pattern, assuming that the underlying approach to compute these scores is reasonably successful in detecting decoys.

This is further validated by the correlation plots between this feature and the target variable (both, ground truth and prediction target); they are not correlated either, suggesting that the score is in no manner representative of the protein structure and hence of its crystal structure. This feature must be random, fake, or a poor approach to decoy detection.

Correlation Plots of Features with Target and Predictions



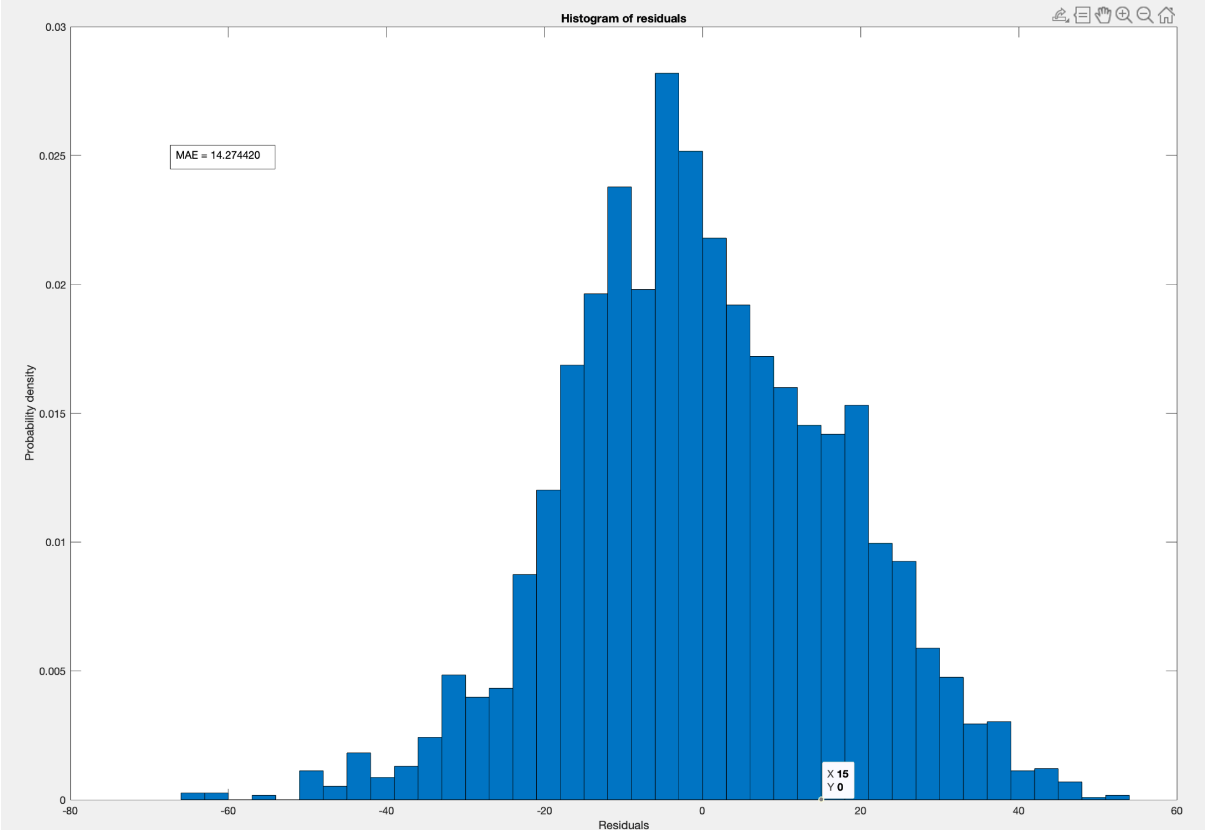
With the exception of the *protein pro* feature, the input features are well correlated with the target variable as well as the predicted target.

The predicted GDT is also strongly, positively correlated with the target GDT (p=0.92).

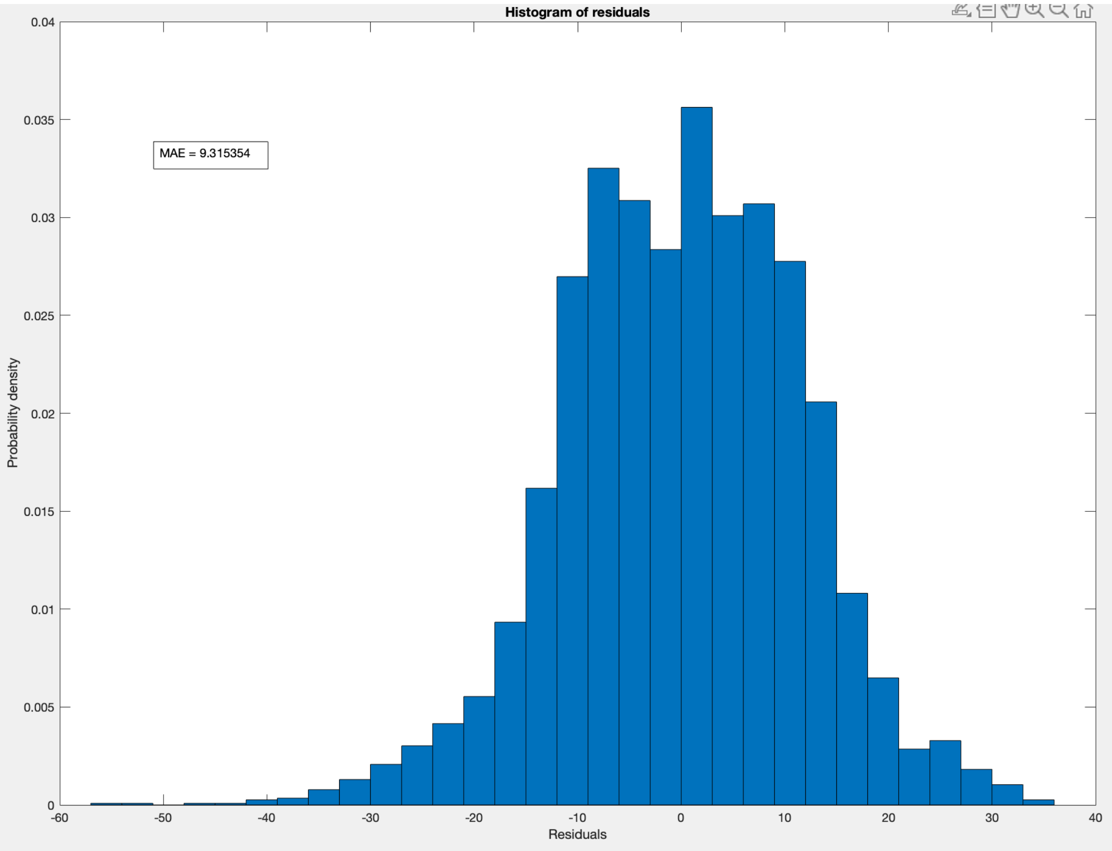
Furthermore, since the input features are highly correlated with the target, a strong linear relationship exists between the input features and the target variable. A complex representation such as that produced by a (deep) neural network may, therefore, not even be necessary to model the relationship well.

Rather, a simpler model, perhaps even a linear model (since the Pearson correlation coefficients are quite high) should be quite effective in modeling the relationship. The following plots show two linear regression models for the same data, with just one feature (ProQ3 which has the higher Pearson coefficient), and with all features but the Protein Pro feature. The MSE values are ~14.27 and ~9.31 each, comparable to (and even better than) the basic NN performance with an MAE of ~10.66.

Error Histogram for Linear Model with only ProQ3 as a feature



Error Histogram for Linear Model with all but Protein Pro as features



These observations suggest that the CASP13 dataset has a strongly linear relationship, not only between its features and the target, but also between the features themselves. The former would mean that simple neural networks should be able to reasonably regress the target variable well.

Whereas, the latter observation suggests that the scores do not have varying information to represent; they more or less represent the same information toward the prediction of protein decoy structure, and hence, it should be possible to develop ML models with a subset of them. All the more, a more effective approach to building a model with these features might be to break them down into orthogonal components, perhaps using a method such as PCA, and select a subset that represents a good portion of the variance in the dataset as features to build the model.