# Oxygen Model (v02) Trained and tuned (D1+M1)

*Name of model:* trained\_model1\_O2\_optim\_weightedJAN25.pth  
csv files: train\_data\_no\_head\_outer\_corner\_O2.csv & test\_data\_no\_head\_outer\_corner\_O2.csv

mean squared error – weighted  
*Optuna Results:*

{

"best\_hyperparameters": {

"lr": 0.0009140692945143792,

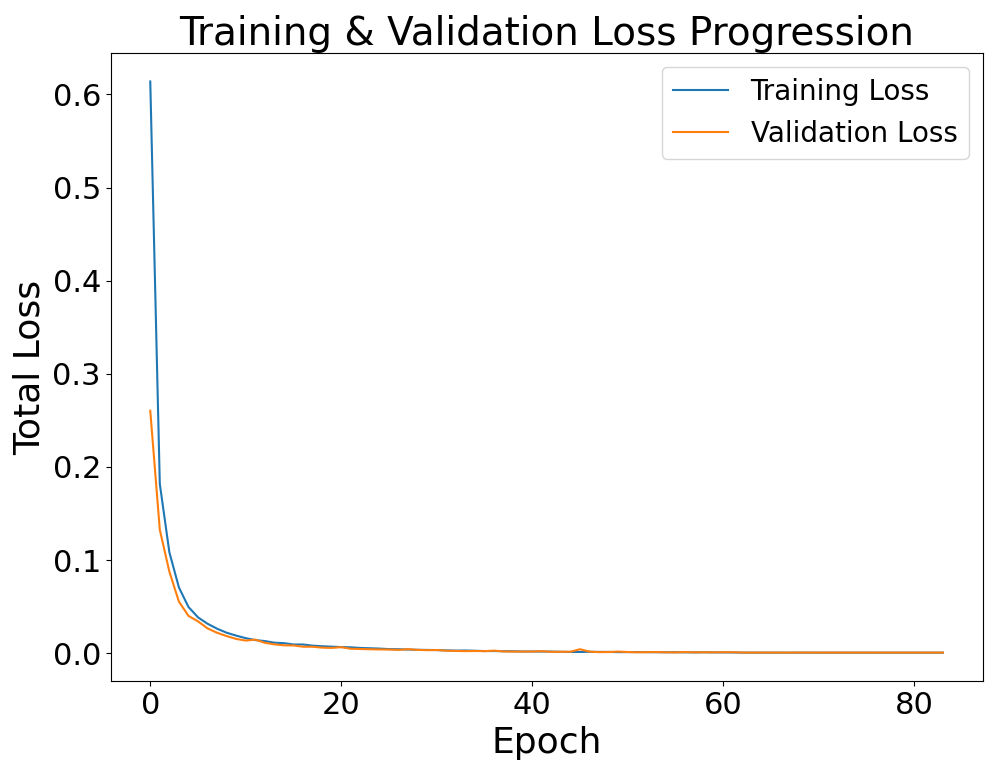
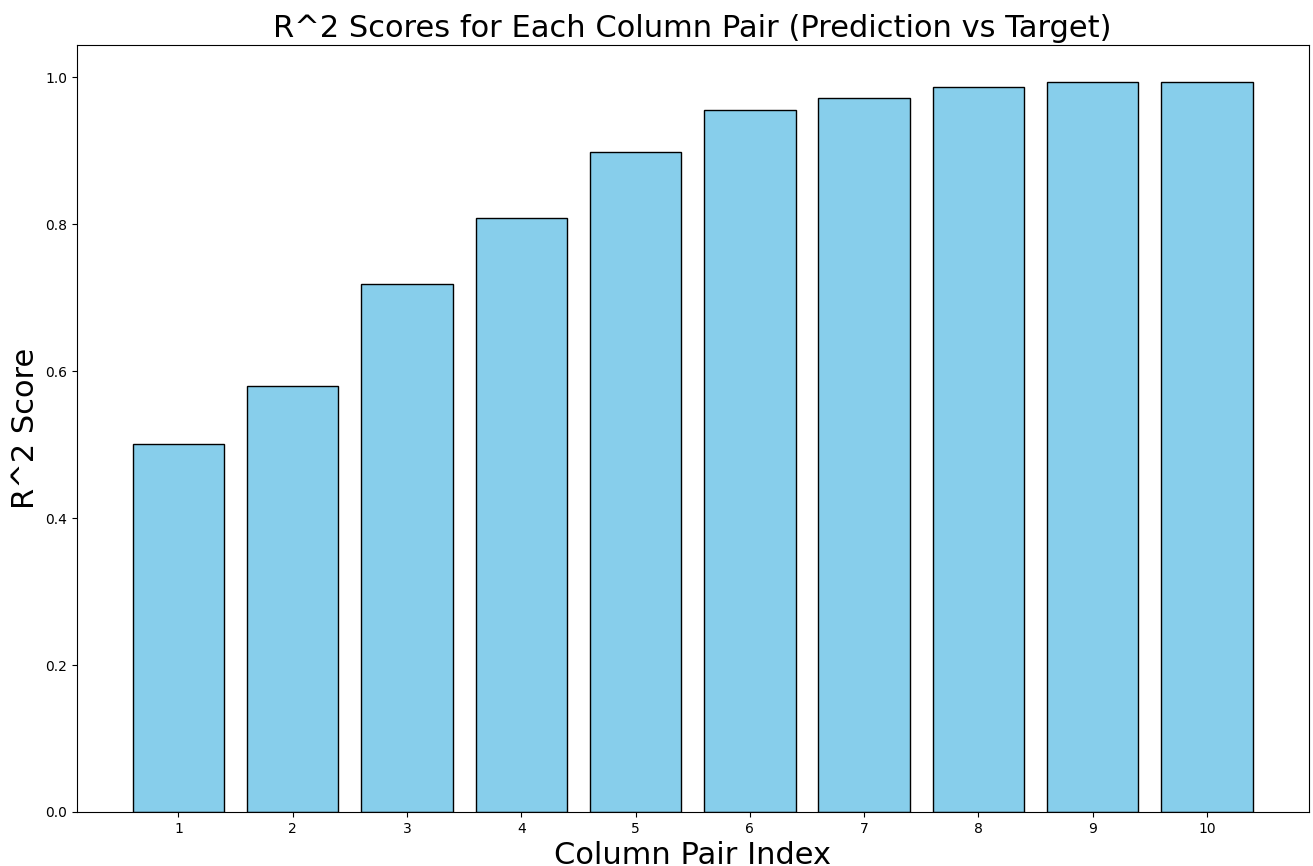
"batch\_size": 16,

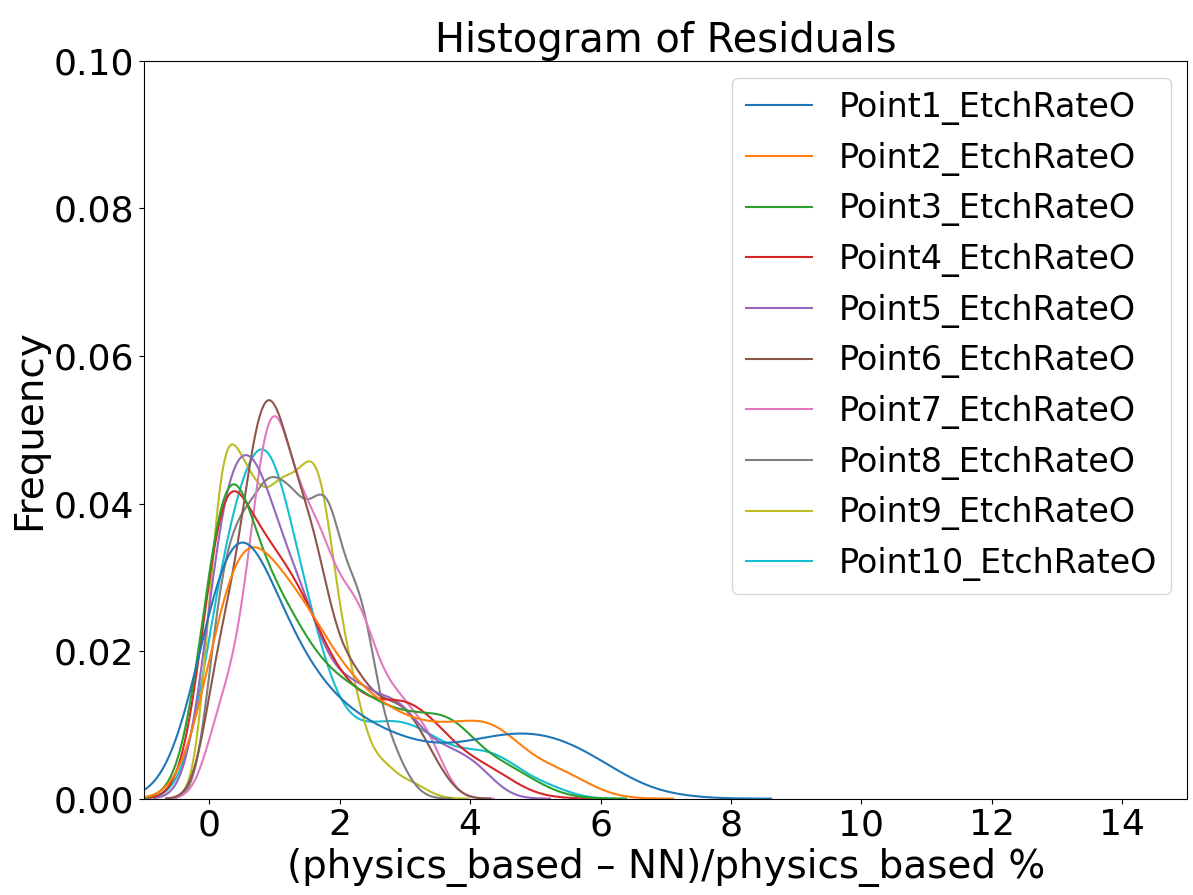
"weight\_decay": 0.000051992439535701895,

"h1": 13,

"num\_layers": 3

}



# Argon Model (v02) Trained and tuned (D2+M2)

*Name of model:* trained\_model1\_Ar\_optim\_weightedJAN25.pth  
csv files: train\_data\_no\_head\_outer\_corner\_Ar.csv & test\_data\_no\_head\_outer\_corner\_Ar.csv  
mean squared error – not weighted  
*Optuna Results:*

{

"best\_hyperparameters": {

"lr": 0.0009518706463849217,

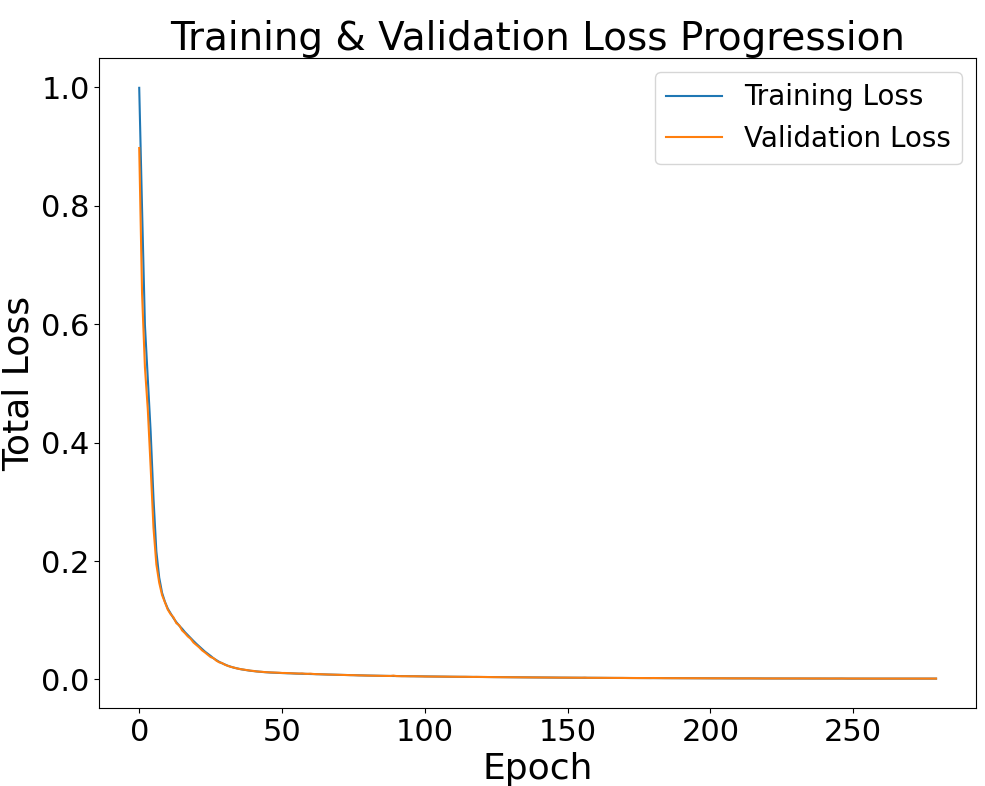
"batch\_size": 16,

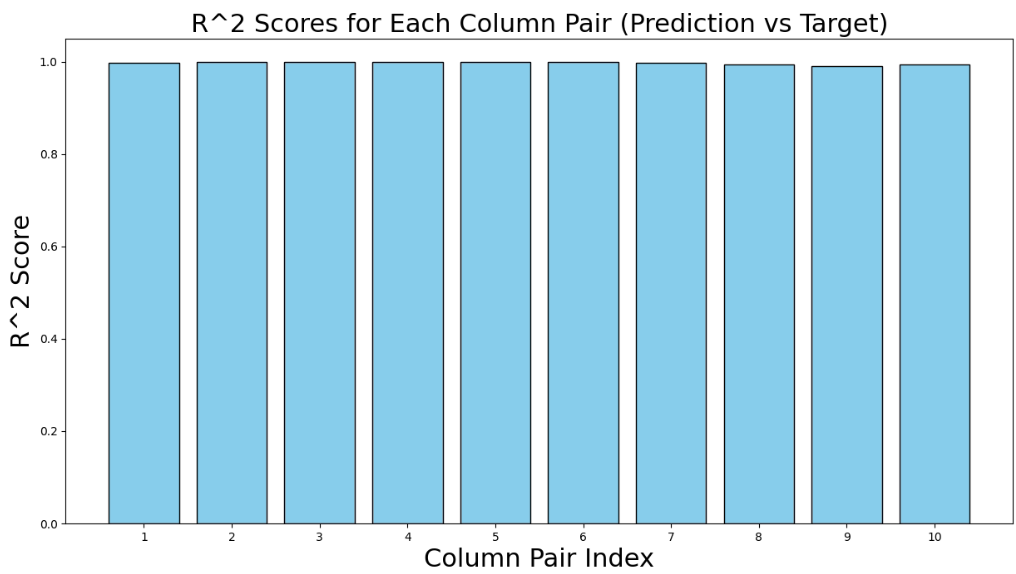
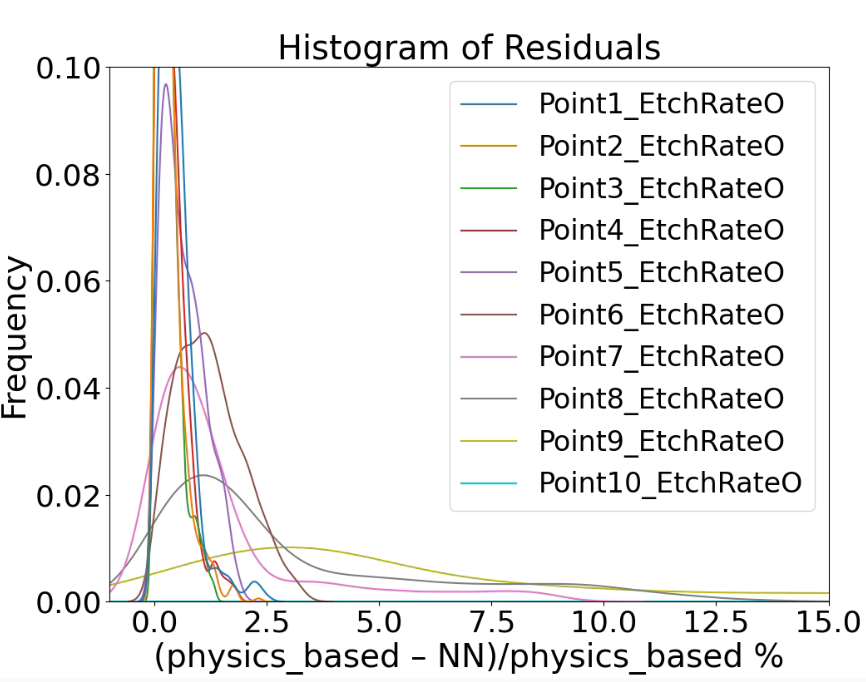
"weight\_decay": 1.0131506615948022e-05,

"h1": 13,

"num\_layers": 3

}

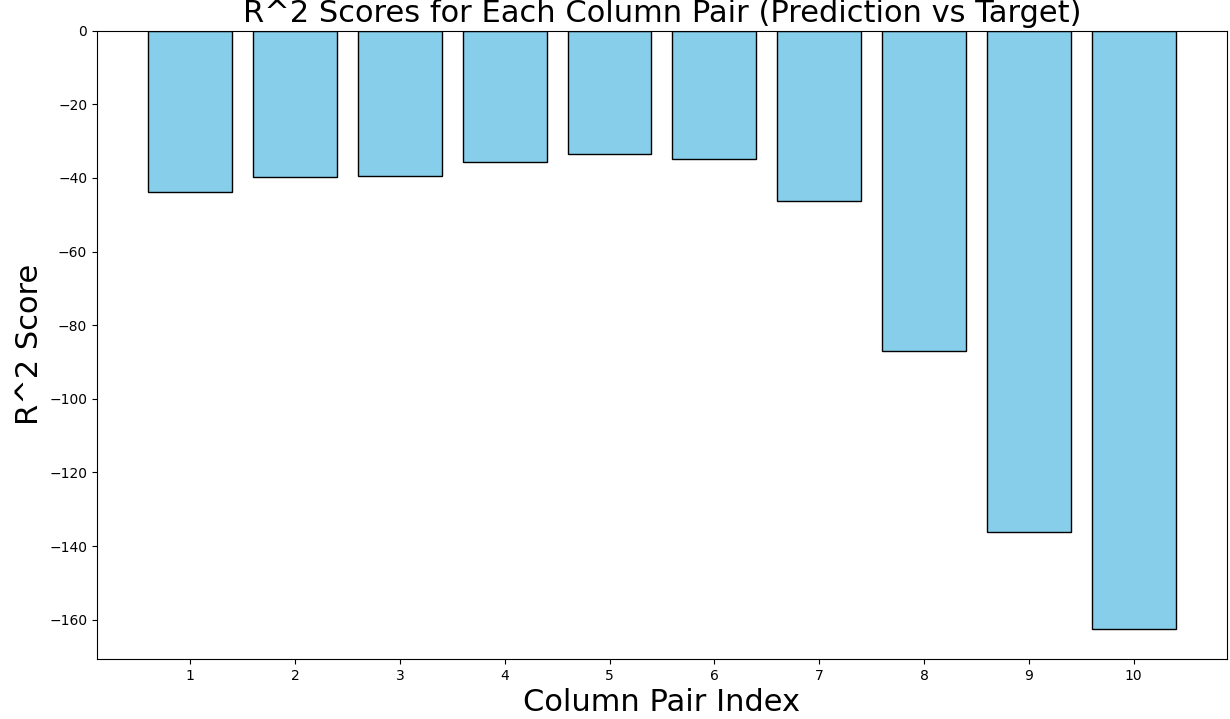
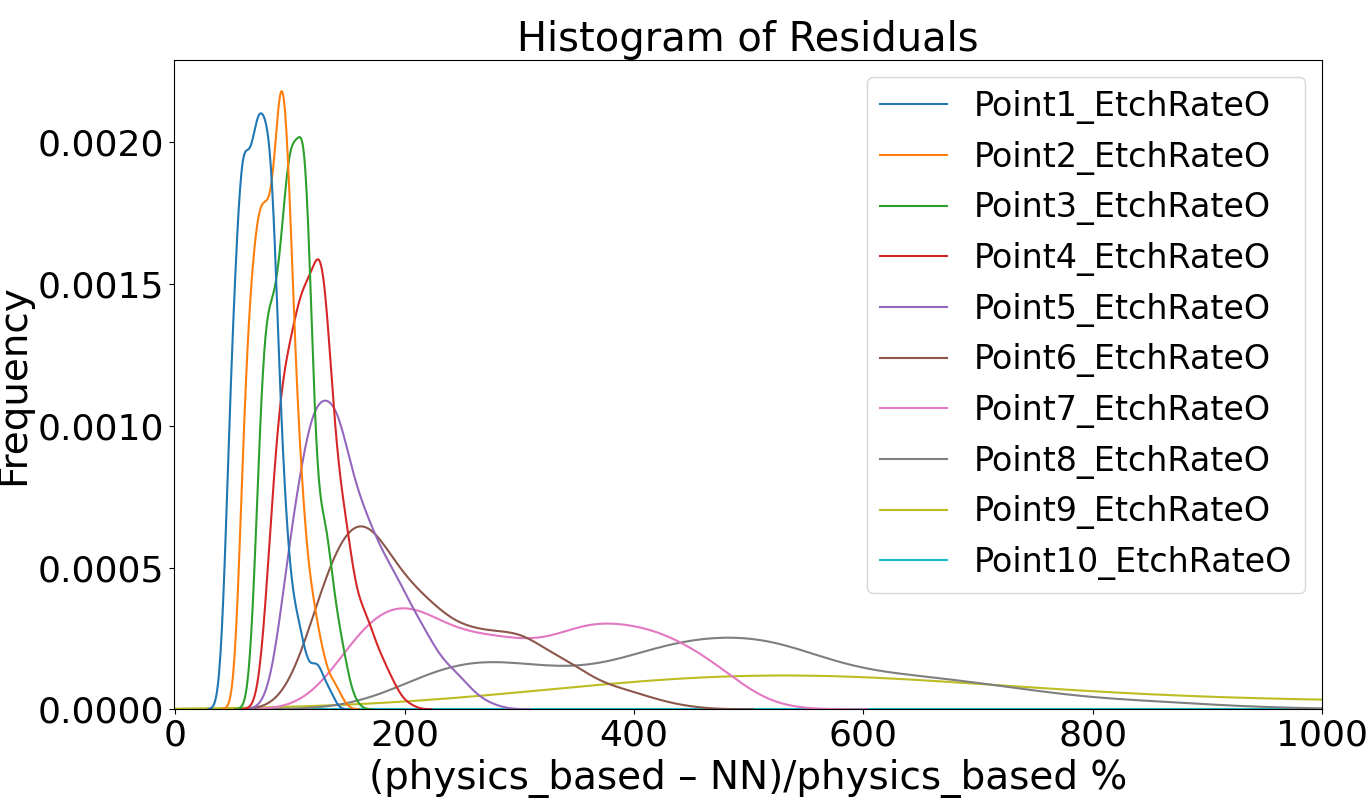


****

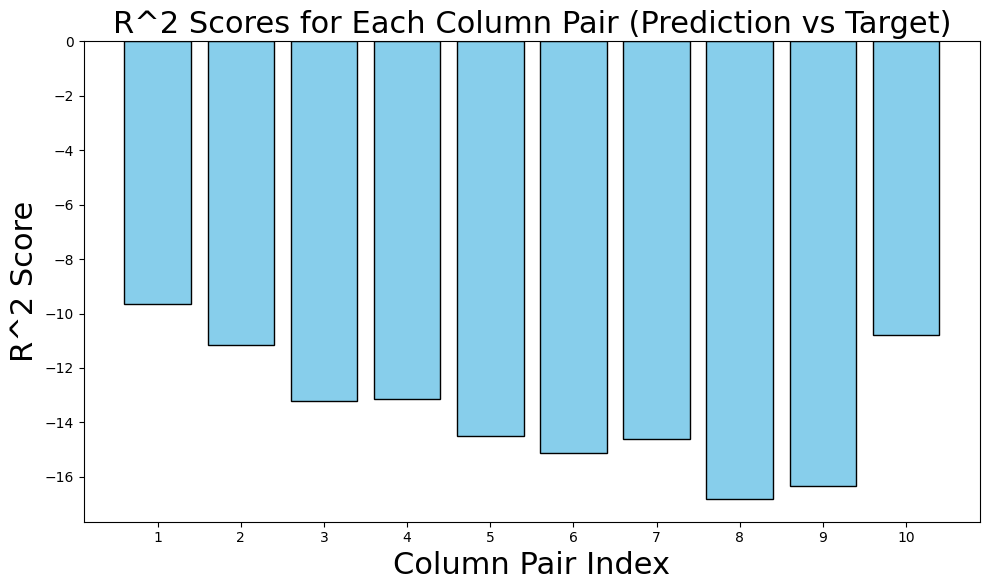
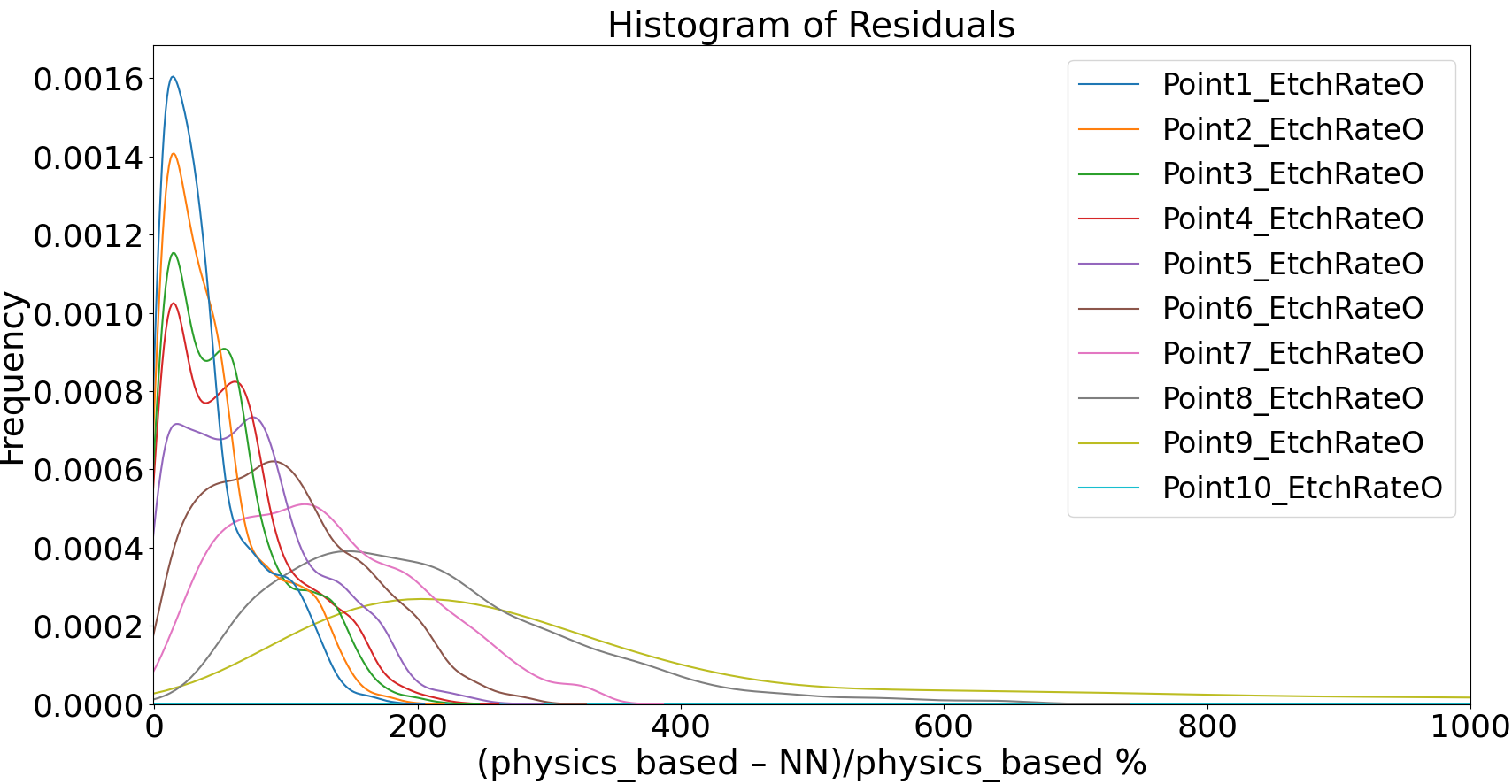
# Oxygen Model (v02) test on Argon Data (M1 on D2)

jsonfile: column\_stats02\_Ο2.json  
csv\_files with predictions also saved: unscaled\_predictions1\_ofM1toD2(train/test).csv

Using the **test** data (D2):

Using the **train** data (D2):

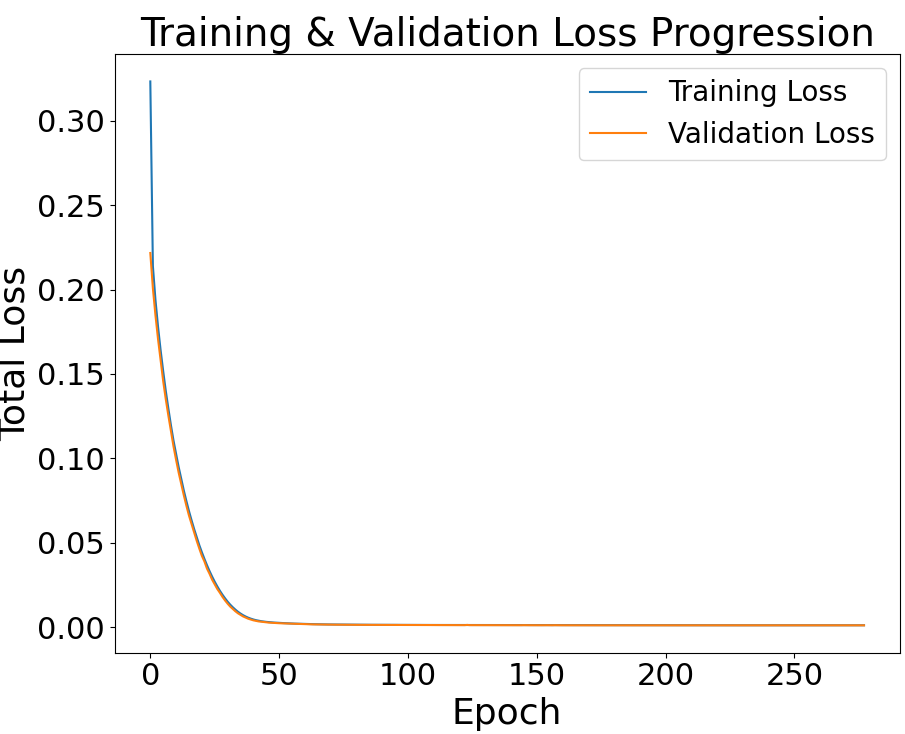
# Train The Oxygen Model (M1) with the whole Argon model data (D2) -> M12

Neural network loaded: trained\_model1\_O2\_optim\_weightedJAN25.pth

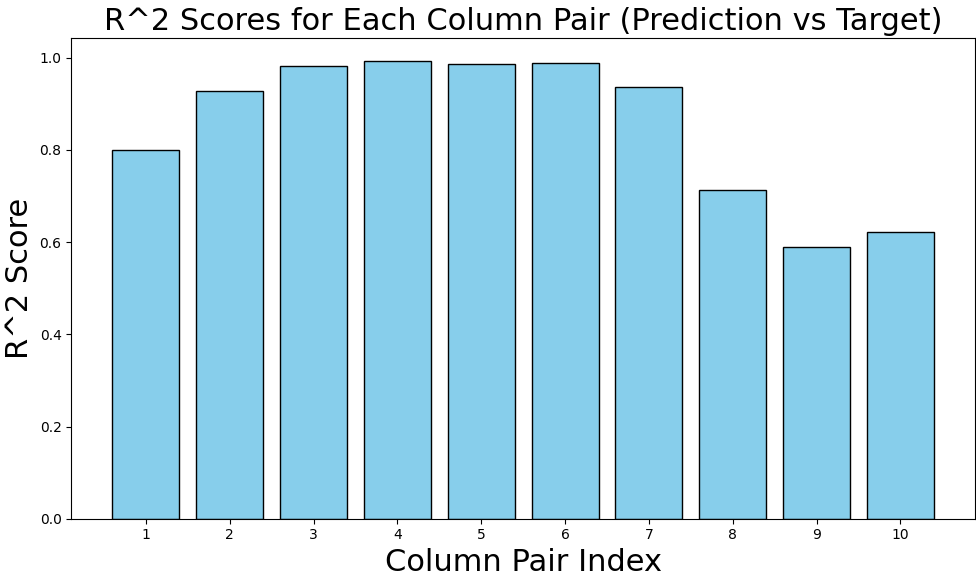
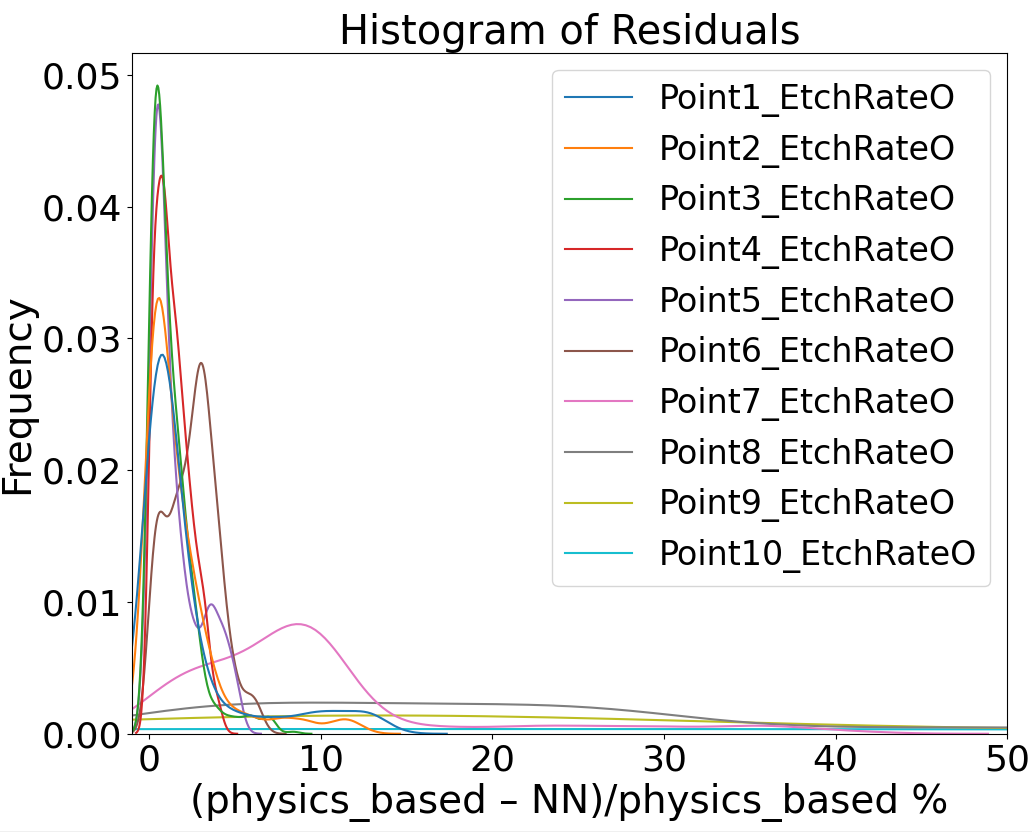
Name of M12 to be saved: M12\_trained\_model1\_O2\_with\_Ar\_data.pth

Jsonfile used: column\_stats02\_O2.json (same as M1)

Learning rate = 0.0009518706463849217/**20**



Test the model with the test data from Argon model

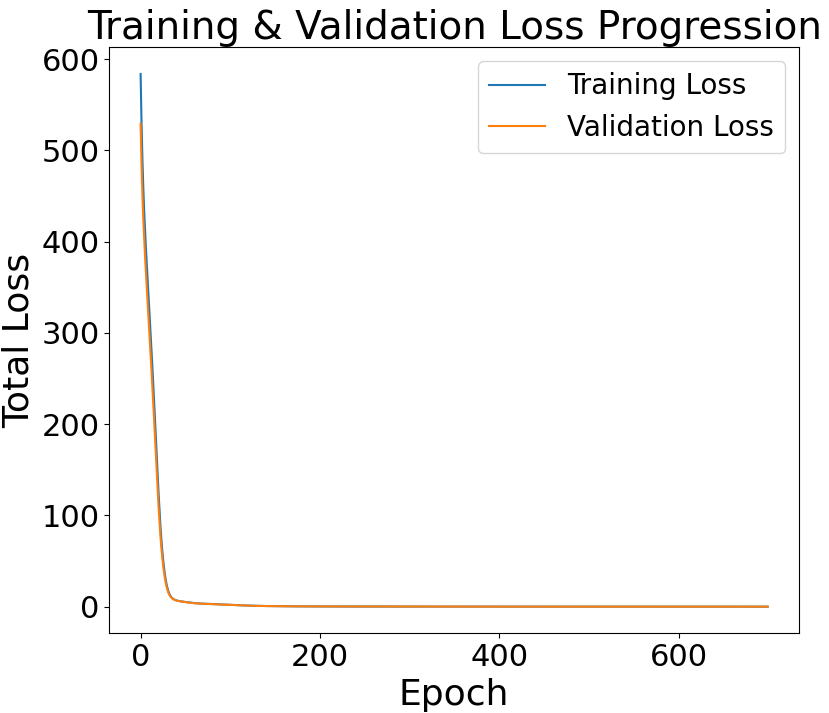
 

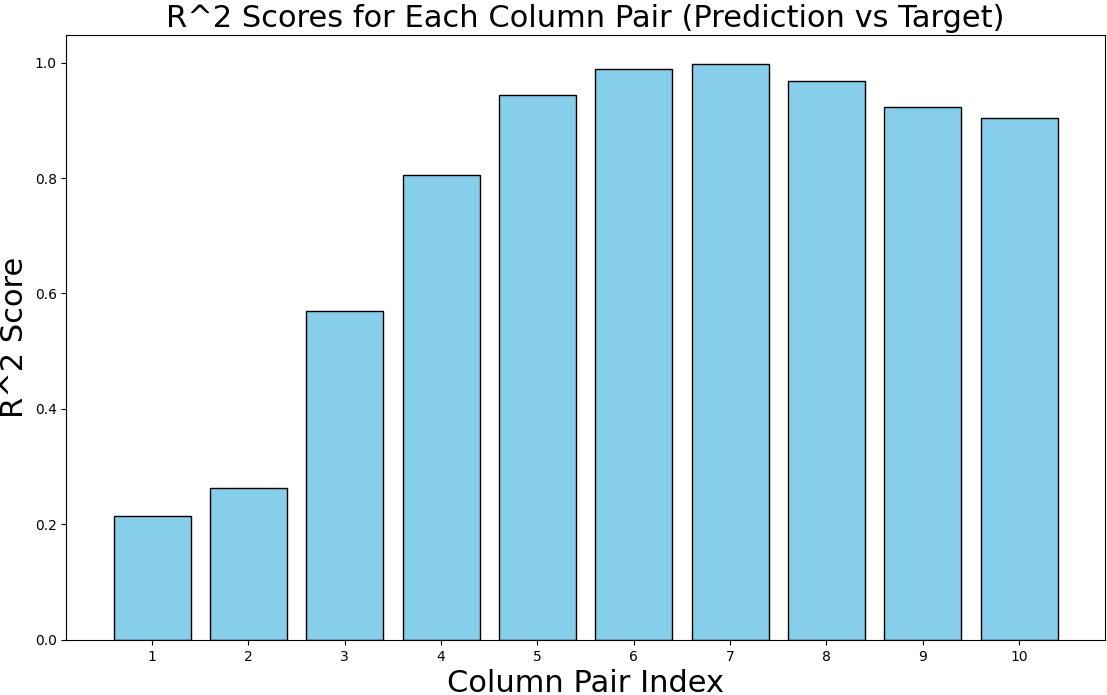
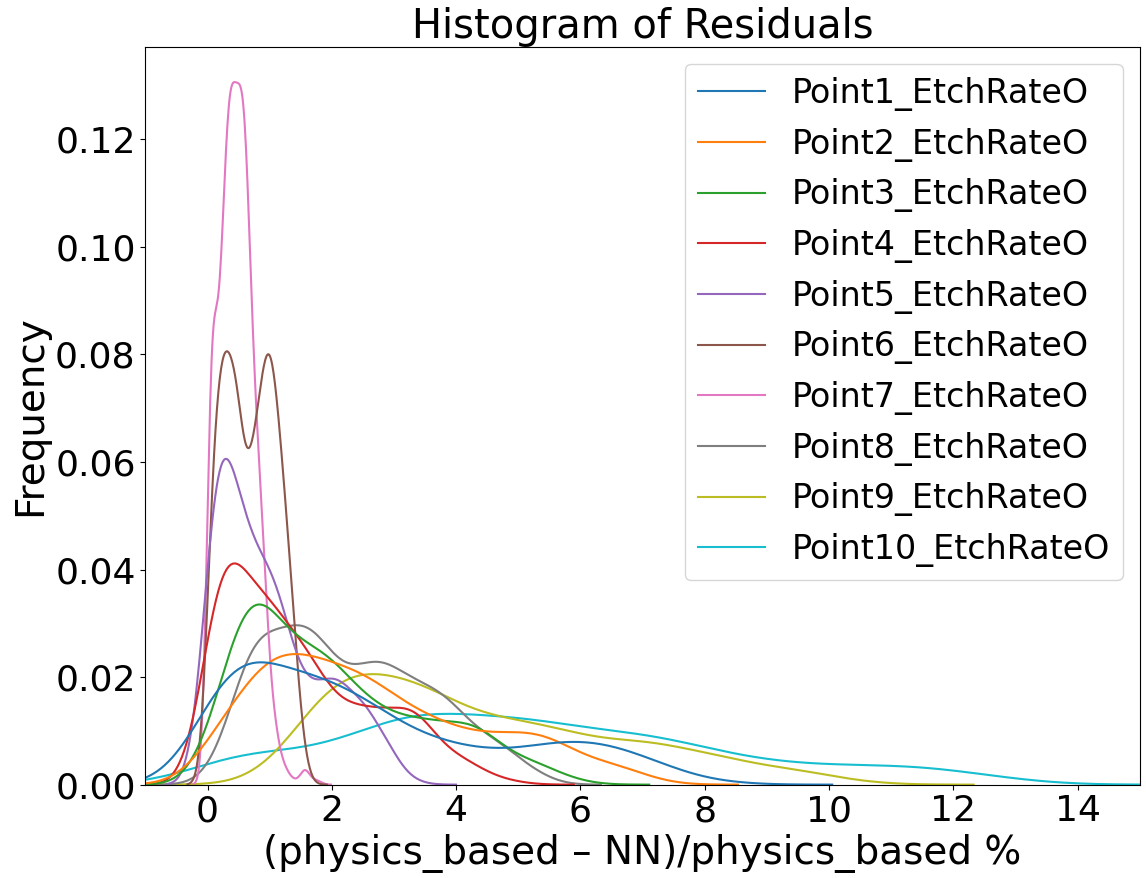
We would expect here M12 to perform better on D2 than M2, but this does not happen if you see the M2 on D2 section above. Let’s try doing it the otherway around. Take the Ar model (M2) and test it on (D1)

Argon Model on Oxygen Dataset (M2 on D1) -> M21

A lot of epochs required on training

Lr = 0.0009518706463849217/20



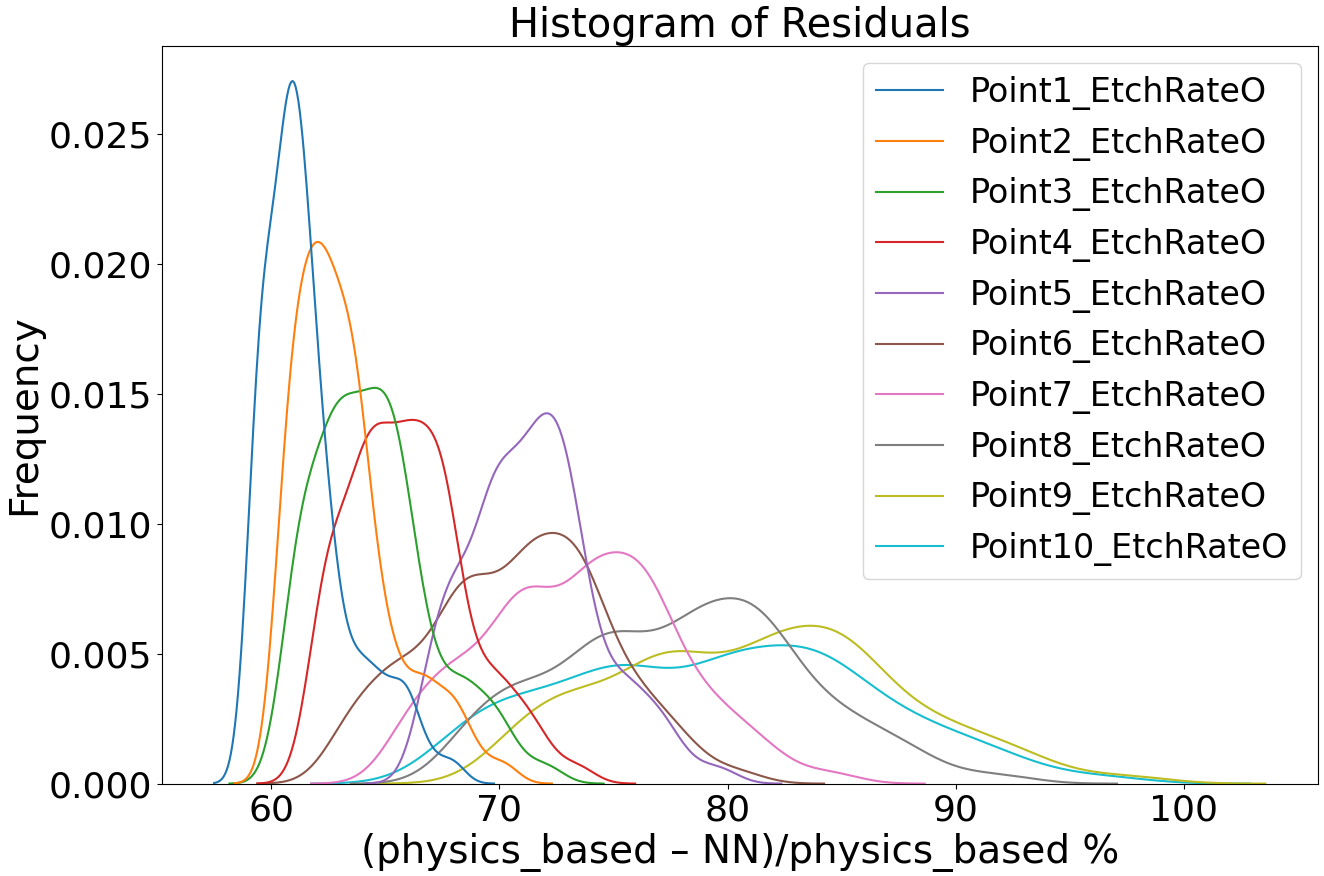
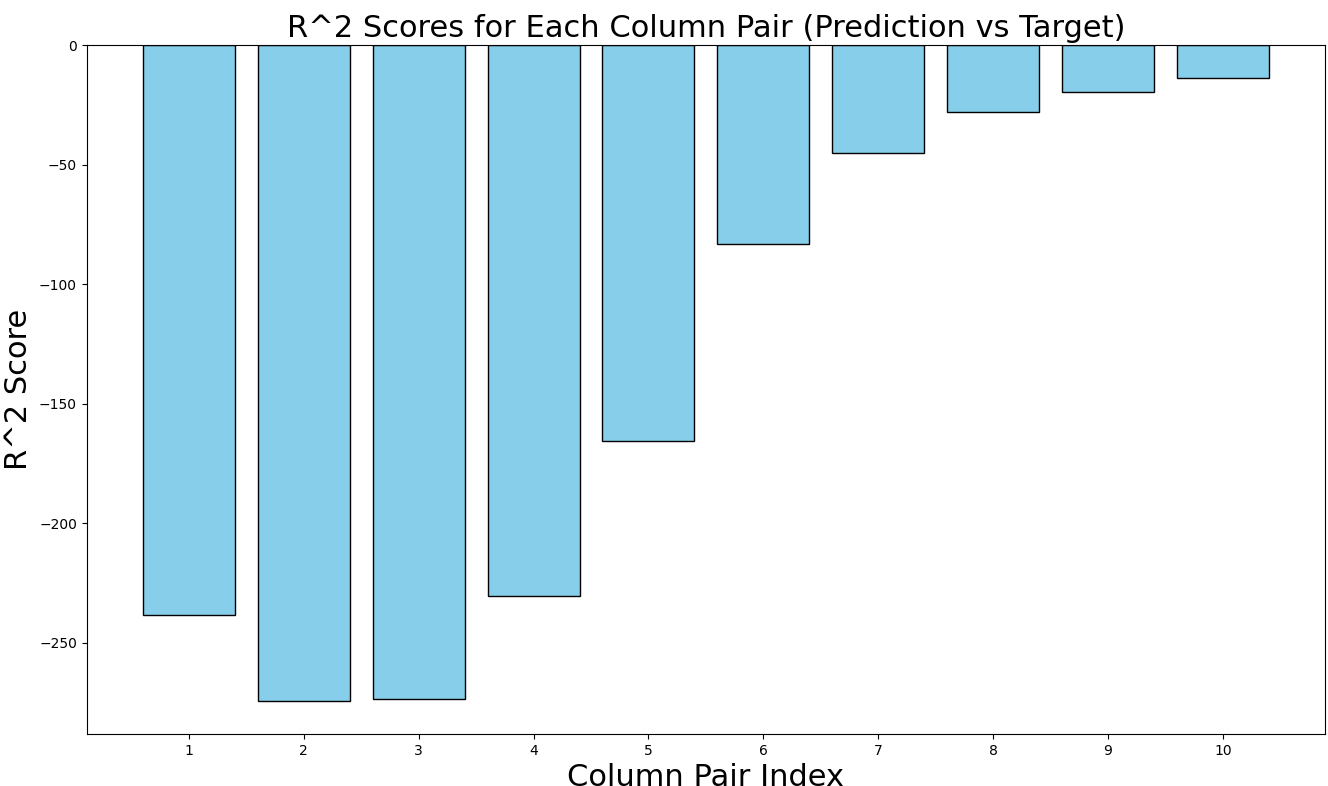
 

~ same performance as oxygen neural network on oxygen data (M1 on D1)

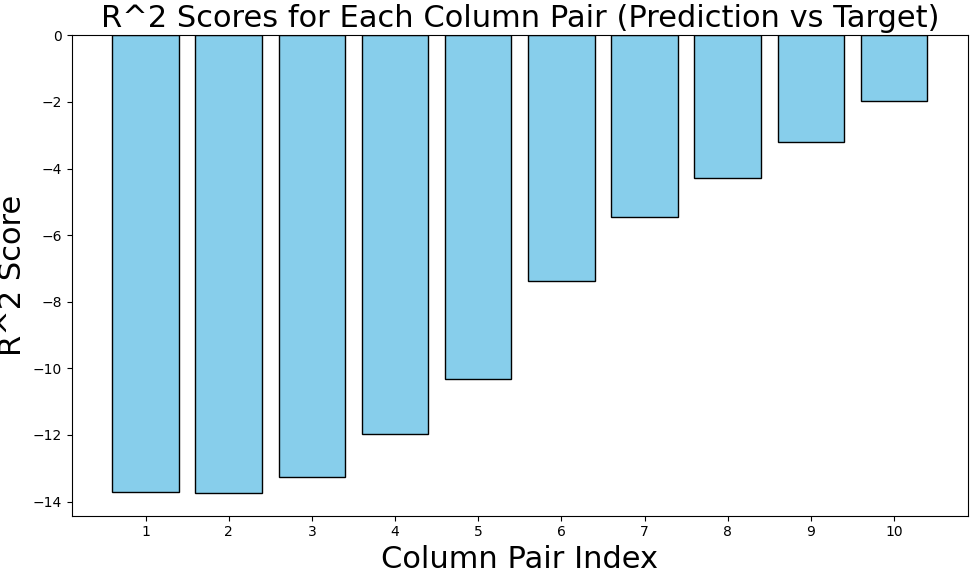
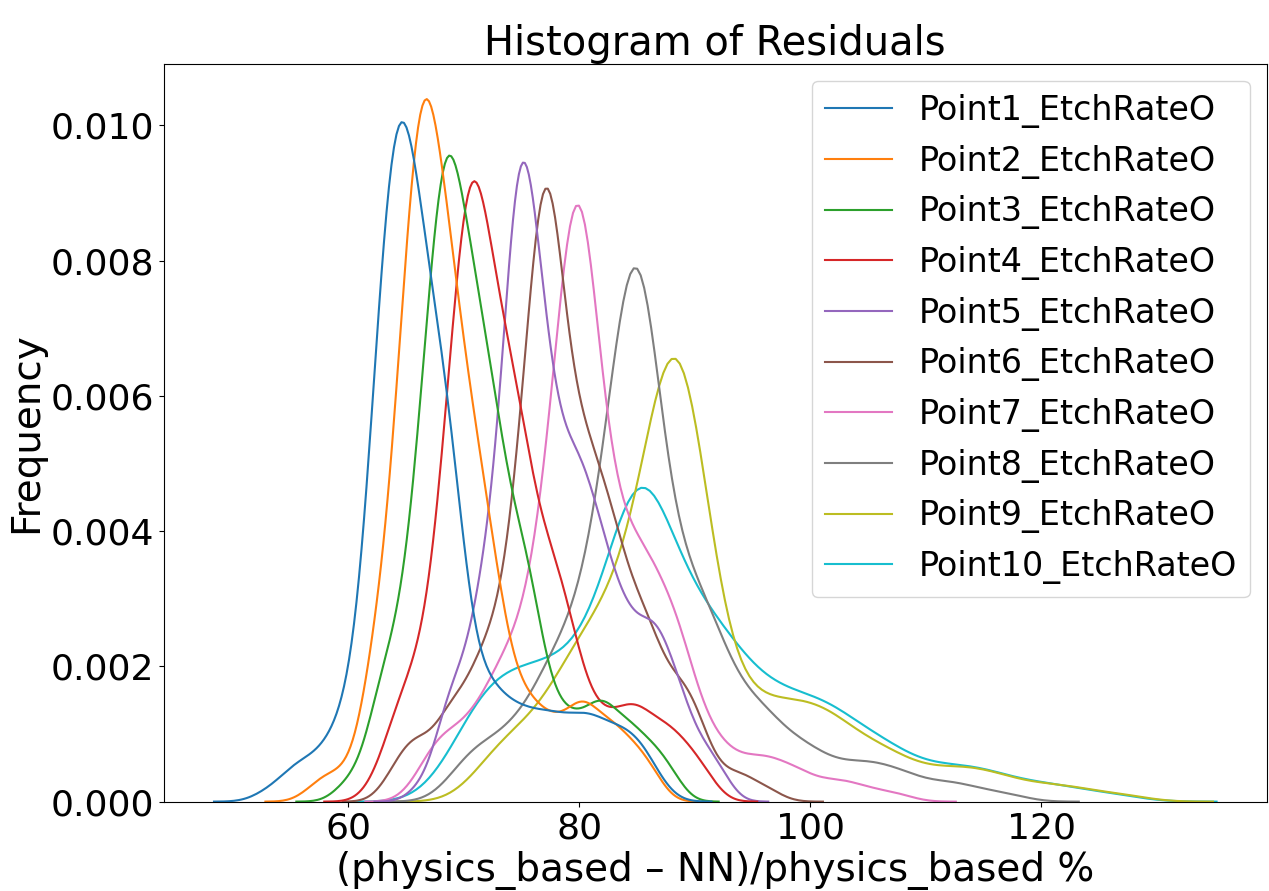
# Evaluate M12 on (D1 – Oxygen Dataset)

M12\_trained\_model1\_O2\_with\_Ar\_data.pth   
column\_stats02\_O2.json (same as M1)

Evaluation on the test data:



Evaluate on the train set:

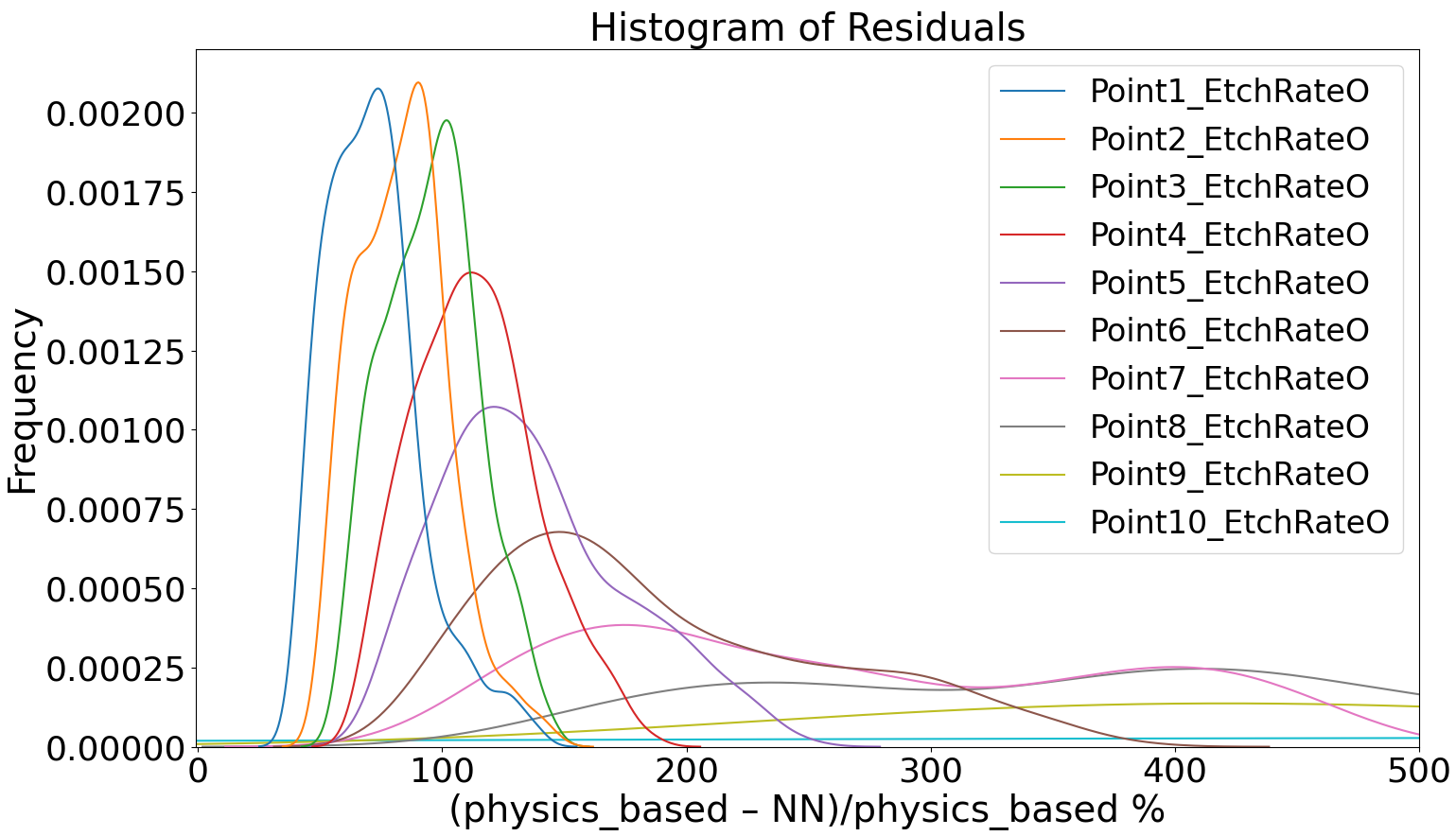
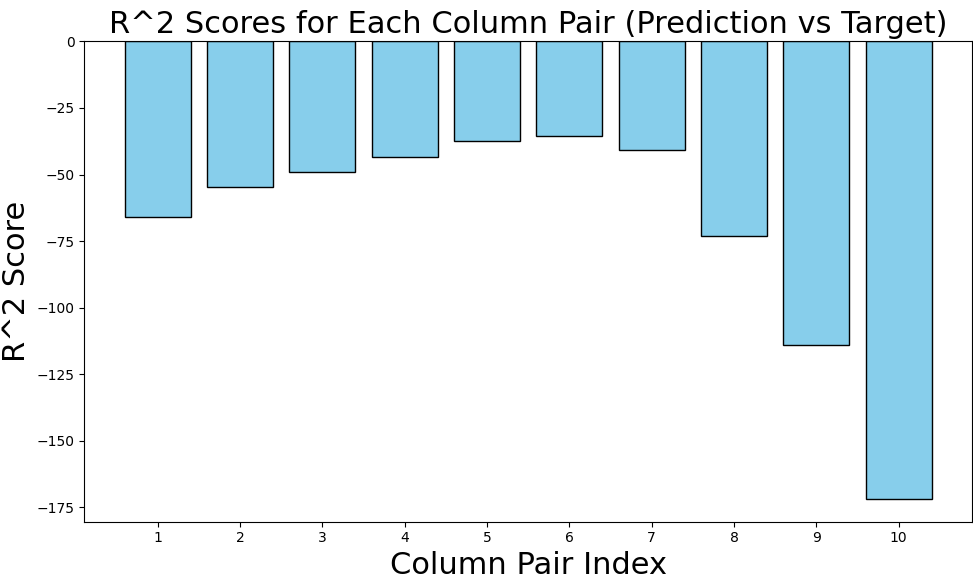
 

Evaluate M21 on D2 - Argon Dataset

M21\_trained\_model1\_Ar\_with\_O2\_data.pth

column\_stats02Ar.json

Test Data:



Train Data:

