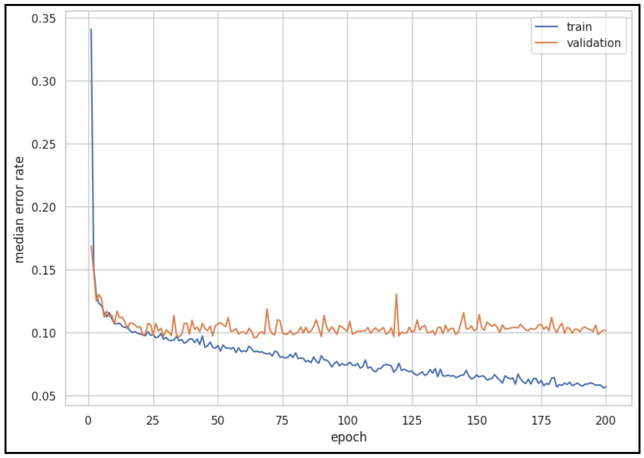
Team 52 Section C: Vaibhav Bagaria, Blaire Kim, Lily Li, Jacqualine Ji, Jay Xie

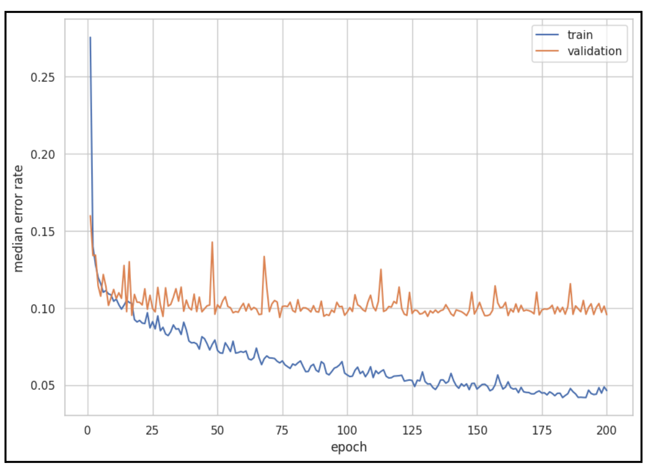
Case_1_(Part_II)_HousePricePrediction_Team 52

1) **A plot of the training errors and validation errors over epochs for a base multilayer perceptron model with 2 hidden layers of sizes 256 and 128.**



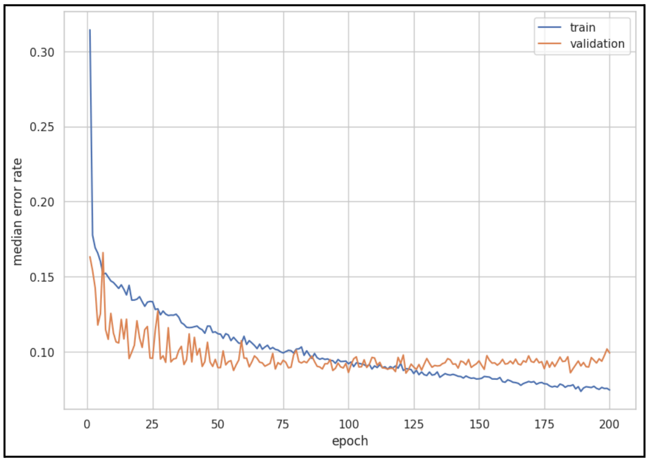
This graph shows that as the model progresses through each epoch, the training error consistently decreases, indicating improved performance on the training data. However, by the 21st epoch, the validation error begins to surpass the training error, with a validation loss of 1.868 and a median validation error rate of 0.107. This divergence suggests that the model is likely overfitting, performing well on the training data but struggling to generalize to the validation set.

1) **A plot of the training errors and validation errors over epochs for a multilayer perceptron model with 4 hidden layers of sizes 512, 256, 128, 64.**



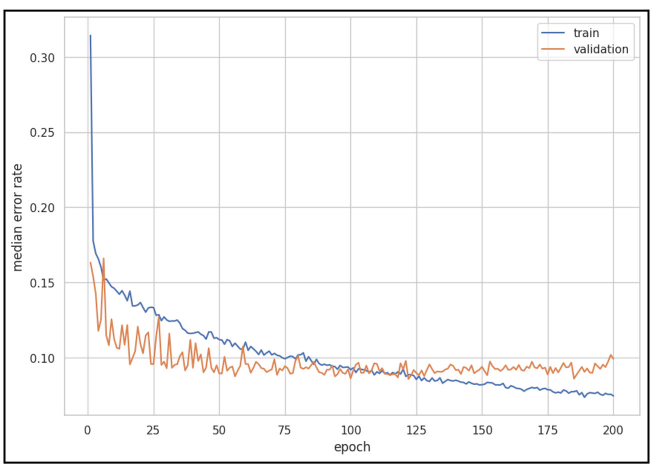
Same thing is happening here. The training error rate steadily decreases throughout the epochs, showing effective learning on the training data. However, the validation error rate stabilizes with noticeable fluctuations, staying consistently higher than the training error rate. This also indicates that while the model learns well on the training set, it may not generalize as effectively to the validation data, potentially overfitting to the data.

2) **A plot of the training errors and validation errors over epochs for a multilayer perceptron model with 4 hidden layers of sizes 512, 256, 128, 64 and norm regularization.**



After adding norm-regularization, the training error rate declines sharply at the initial stage, while validation error decreases gradually. As it goes through the 40th epoch, the validation error stabilizes and remains lower than the training error which means the adding penalty effectively resolves overfitting, improving the model performance.

3) **A plot of the training errors and validation errors over epochs for a multilayer perceptron model with 4 hidden layers of sizes 512, 256, 128, 64 and norm regularization and dropout layers.**



When we apply a norm regularization and 4 hidden layers to our prediction model, we see that the training and validation error rates decrease around the first 20 or so epochs, but validation error rate achieves convergence around epoch 75-100. After that threshold, our training error begins to diverge from our validation error line, which indicates that the model is overfitting.

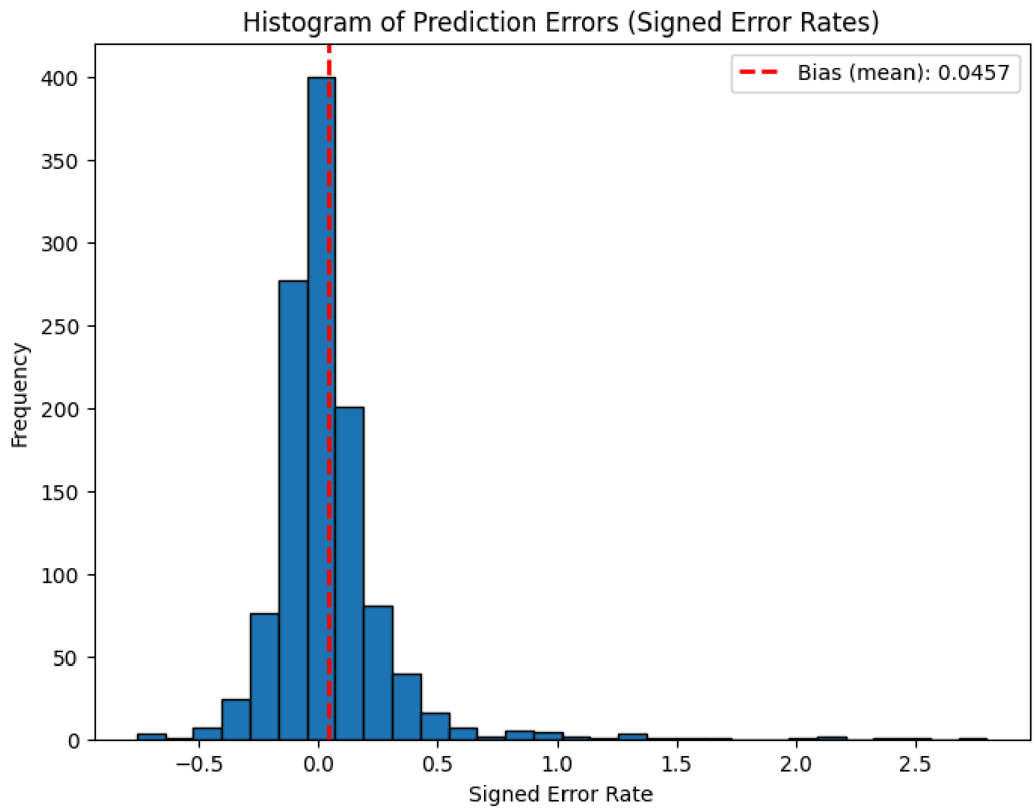
4) **A table listing all the model hyperparameters that you have tried with the corresponding validation errors that you found.**

	Learning Rate	Validation Error	Hidden Layers
0	0.0100	0.093924	NaN
1	0.0050	0.095657	NaN
2	0.0010	0.087036	NaN
3	0.0005	0.086570	NaN
4	NaN	0.097262	[256, 128]
5	NaN	0.096042	[512, 256, 128]
6	NaN	0.091190	[1024, 512, 256, 128]

This table shows the hyperparameters explored - learning rate and the number of hidden layers - along with the resulting validation errors. As shown, reducing the learning rate and increasing the complexity of hidden layer configurations generally lead to lower validation errors. However, relying solely on these adjustments may increase the risk of overfitting, so further tuning and evaluations are necessary to achieve optimal model performance systematically.

Your profit analysis of the iBuyer business model based on the predicted price on the valid data and answers to the four questions therein.

Question 1: what is the bias of the prediction errors? Include the histogram of prediction errors and the bias in your report.



The mean bias of the prediction error is 0.0457, which shows us that our house prediction ML model has a slight positive bias and tends to overestimate house price predictions to actual house price value.

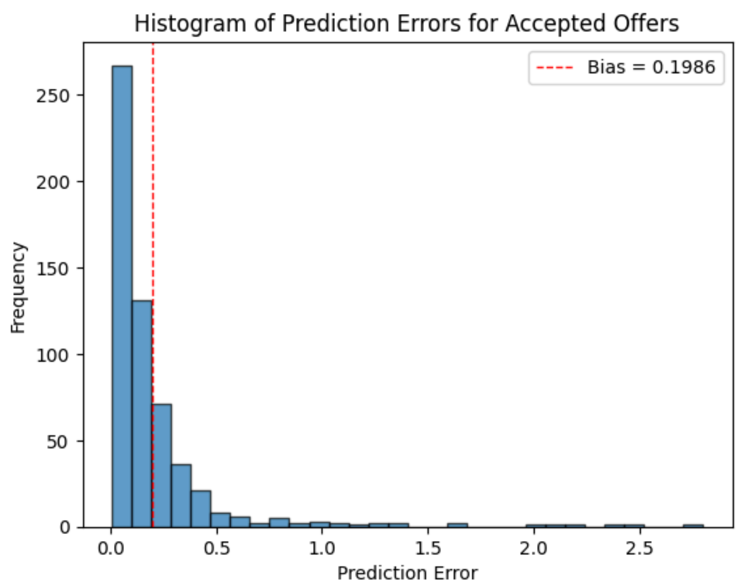
Question 2: Consider the hypothetical scenario where the offers are all accepted regardless of their values, what is the average percentage profit? Do you see a big difference compared to the profit margin α? Include your answers in the report.

The average percentage profit if all offers are accepted is 0.1228 based on our calculation, which means 0.028% larger than our target. This slight difference suggests that the actual average profit is marginally higher than the targeted profit margin.

Question 3: Based on the sale price in the valid data and the acceptance rule, what is the mean percentage profit among all accepted offers? Do you see a big difference compared to the targeted profit margin α? Include your answers in your report.

The mean percentage profit amongst all accepted offers is -0.0317, which indicates that there is a 3.17% loss on accepted offers on average. Our target profit margin of alpha is 12%. This is around a 15% decrease from our target profit margin, which indicates that our current pricing strategy at iBuyer does not meet our target. To address this, we can improve our acceptance rule threshold by increasing the current beta of 90%.

Question 4: What is the bias of the prediction errors when restricting to those properties whose owners accepted the offer? Based on the histogram and bias, can you explain your answers to Question 3?



The bias (bias_accepted = 0.1986) suggests that there’s a tendency in the accepted offers to have a positive prediction error, meaning that predictions tend to overestimate the actual prices by an average of about 19.86%.

The histogram shows the spread of these prediction errors, and a positive bias indicates that the model might need further adjustments to reduce overestimation, especially for accepted offers. If mean_percentage_profit_accepted is negative or lower than the target, this could indicate that the acceptance criteria (based on beta) are not stringent enough to maintain the desired profit margin.