Appendix: Detailed Results

Commented [DS1]: I've placed this in one document for now, but will make a separate appendix document later

1. Introduction

This document serves as an appendix to the paper titled "Evaluating human-in-the-loop machine-learning querying strategies for missing value imputation". This appendix begins with a description of the experiment carried out, the summarised training and cross-validation results, as well as the results from the Mann-Whitney U test. The decision tree-based active learners emerge as the best performing, demonstrating efficient learning and generalization across all features. The query-by-committee (QBC) algorithm shows the best This paper has shown that ML algorithms can be used for MVI and error correction, with k-NN and MICE providing efficient, and reliable estimates to missing values, with minimal training effort.

MVI performance is very good on features with lower standard deviation and cardinality, while it remains unreliable on others. The categorical features *'Material Weight'* and *'Material Part Number'*, modelled as continuous features, also provide good results, further demonstrating the versatility of the imputation methods. These observations are critical for understanding the applicability and limitations of different imputation techniques for error correction in the HVB dataset.

2. Grid search results

Hyperparameters are selected following a ten-fold cross-validation grid search for each model-feature combination. Grid searches are conducted on all available data, withholding data used for the holdout sets. The results for the decision tree, k-NN, and MLP algorithms are shown in Tables 11, 12, and 13, respectively.

Table 1: Decision Tree Grid Search Results

Dataset	Feature	Problem Type	Criterion	Max Depth	Min Samples Leaf	Splitter	Metric	Best Score
HVB	Material Weight	Regression	squared_error	10	1	best	R2	0.999998
	Material Part Number	Regression	squared_error	10	1	best	R2	0.985151
	Minimum Capacity	Regression	squared_error	15	1	random	R2	0.999973
	Capacity Throughput	Regression	absolute_error	20	1	best	R2	0.995930
	Voltage	Regression	friedman_mse	7	32	best	R2	0.752215
	Model Code	Classification	gini	10	1	random	Accuracy	0.999773
Red wine-quality	quality	Classification	gini	15	1	best	Accuracy	0.6219

Abalone	Rings	Regression	squared_error	10	32	best	R2	0.5951

The cosine distance metric is selected for most tasks and shows significantly better training performance than other distance metrics in the conducted experiments.

Table 2: k-NN Grid Search Results

Dataset	Feature	Problem Type	Metric (Distance)	n Neighbors	Weights	Metric	Best Score
HVB	Material Weight	Regression	cosine	3	distance	R2	0.995242
	Material Part Number	Regression	cosine	3	distance	R2	0.920068
	Minimum Capacity	Regression	cosine	3	distance	R2	0.994859
	Capacity Throughput	Regression	cosine	3	distance	R2	0.764981
	Voltage	Regression	cosine	11	uniform	R2	0.705323
	Model Code	Classification	manhattan	3	distance	Accuracy	0.704433
Red wine-quality	quality	Classification	cosine	11	distance	Accuracy	0.6594
Abalone	Rings	Regression	cosine	15	distance	R2	0.5878

The results for the MLP algorithm are shown in Table 12. The adaptive moment estimation (Adam) solver was selected for all MLP-based active learners, as the datasets are large [52]. The maximum iterations were tuned to 2000 to balance model fit and computational effort. L2 regularisation was tested separately, and an 'alpha' value of 0.0001 was selected.

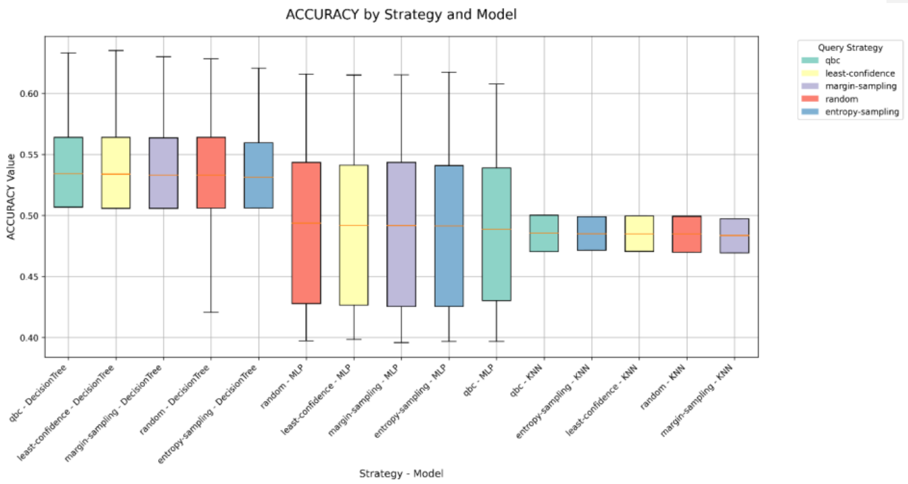
Table 3: MLP Grid Search Results

Dataset	Feature	Problem Type	Activation	Hidden Layers	Learning Rate	Metric	Best Score
HVB	Material Weight	Regression	logistic	(130, 50)	0.001	R2	0.905671
	Material Part Number	Regression	relu	(70, 10)	0.001	R2	0.374984
	Minimum Capacity	Regression	logistic	(100, 70)	0.001	R2	0.942476
	Capacity Throughput	Regression	identity	(150, 10)	0.01	R2	0.212695

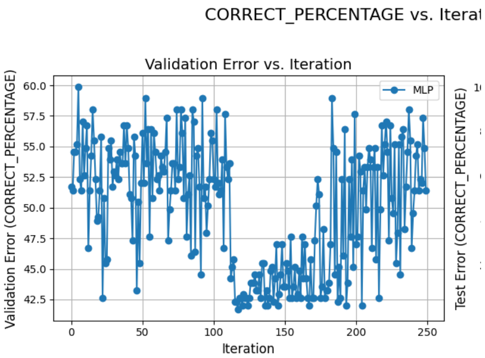
	Voltage	Regression	identity	(90, 50)	0.01	R2	0.320200
	Model Code	Classification	logistic	(70, 10)	0.001	Accuracy	0.824967
Red wine-quality	quality	Classification	logistic	(50, 130)	0.0001	Accuracy	0.575
Abalone	Rings	Regression	logistic	(80, 30)	0.0001	R2	0.2682

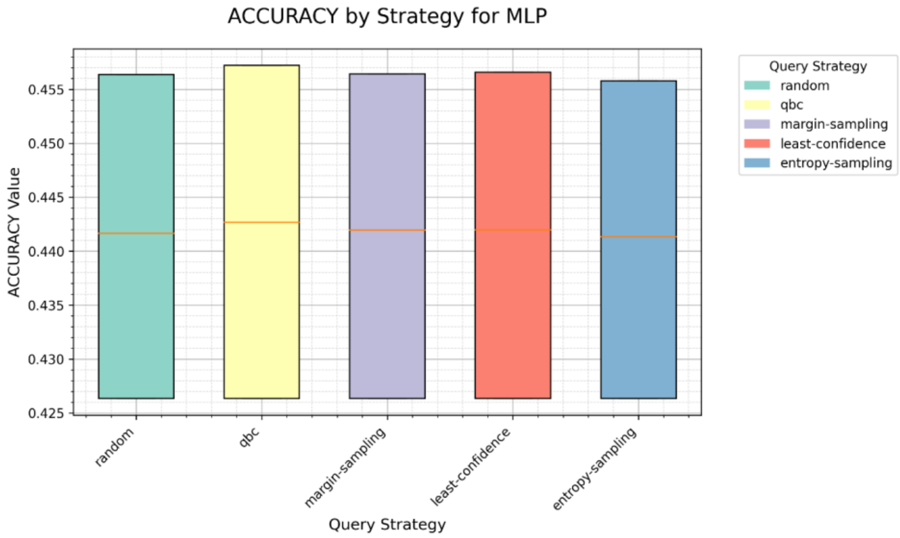
3. Additional classification results

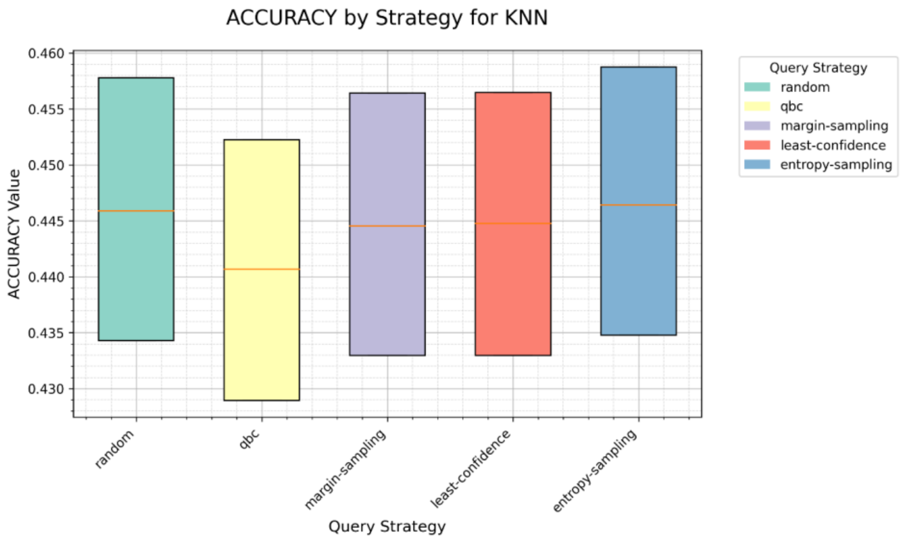
- Includes query indices, percentage of the 6 classes queried
- Include other metrics? (f1)
- Test error results (during training)

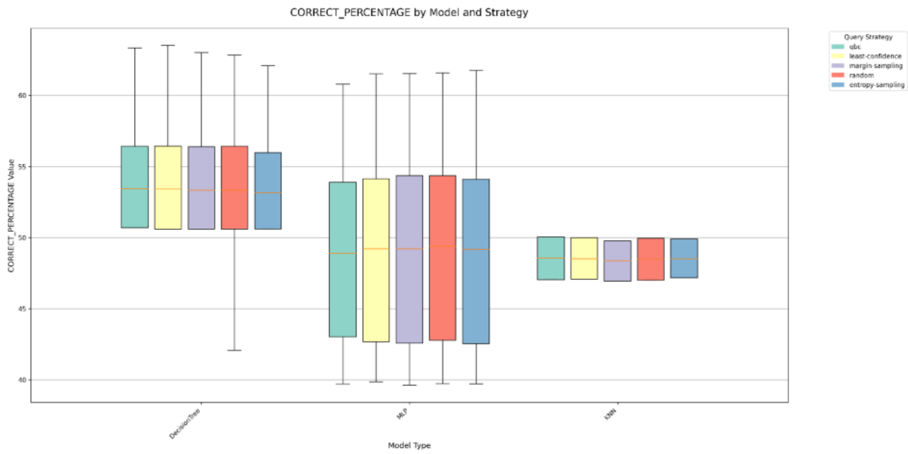


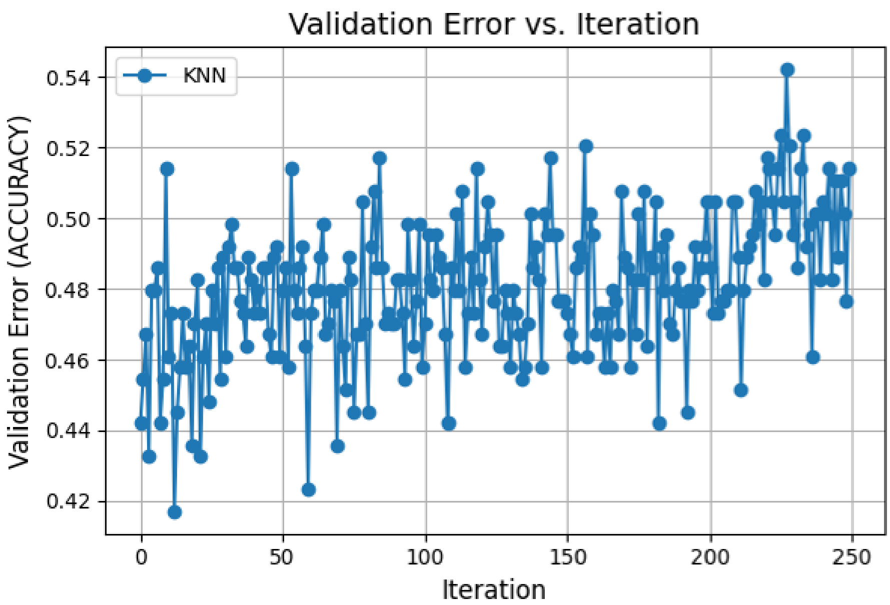
Red wine – quality




ACCURACY by Strategy for MLP
ACCURACY Value
0.455
0.450
0.445
0.440
0.435
0.430
0.425
random
qbc
margin-sampling
least-confidence
entropy-sampling
Query Strategy
Query Strategy
random
qbc
margin-sampling
least-confidence
entropy-sampling


ACCURACY by Strategy for KNN
ACCURACY Value
0.460
0.455
0.450
0.445
0.440
0.435
0.430
random
qbc
margin-sampling
least-confidence
entropy-sampling
Query Strategy
Query Strategy
random
qbc
margin-sampling
least-confidence
entropy-sampling

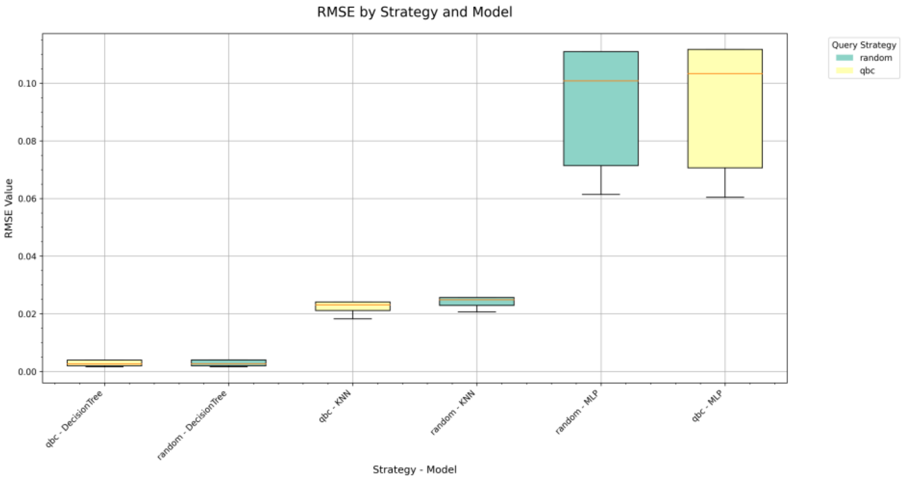


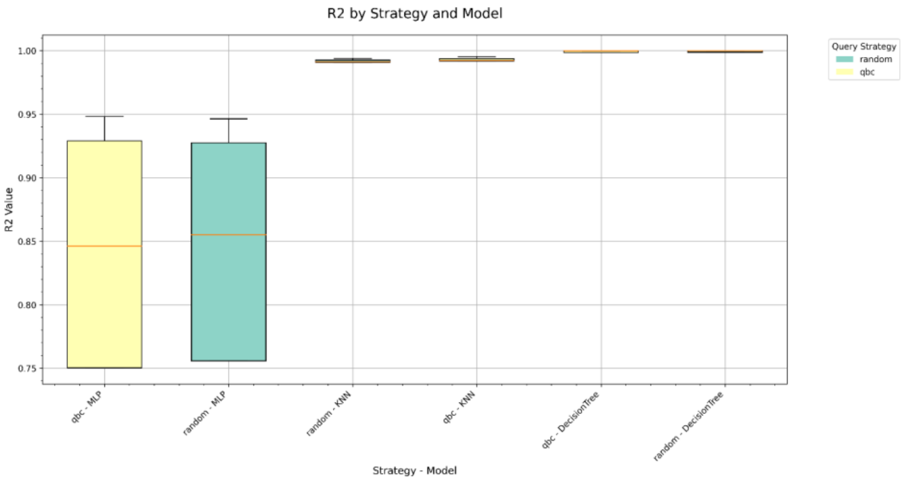


4. Additional regression results

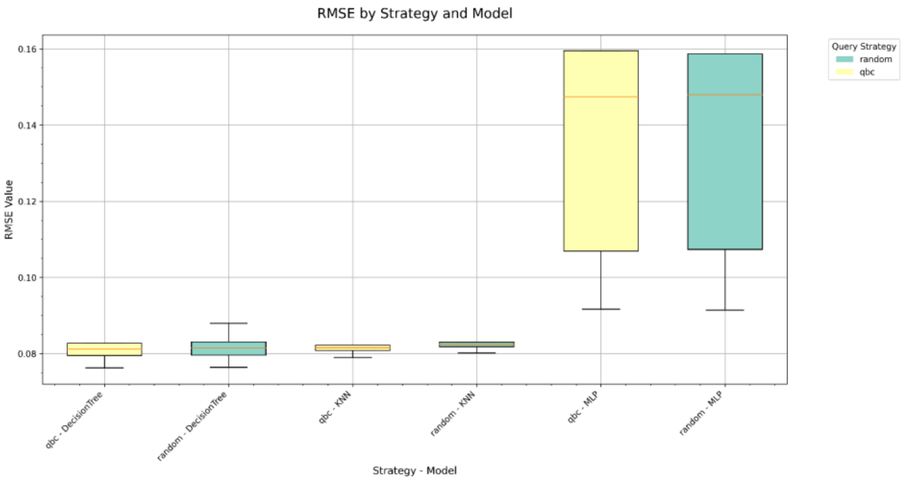
- Other metrics (MAE, MSE)
- Test error results (during training)

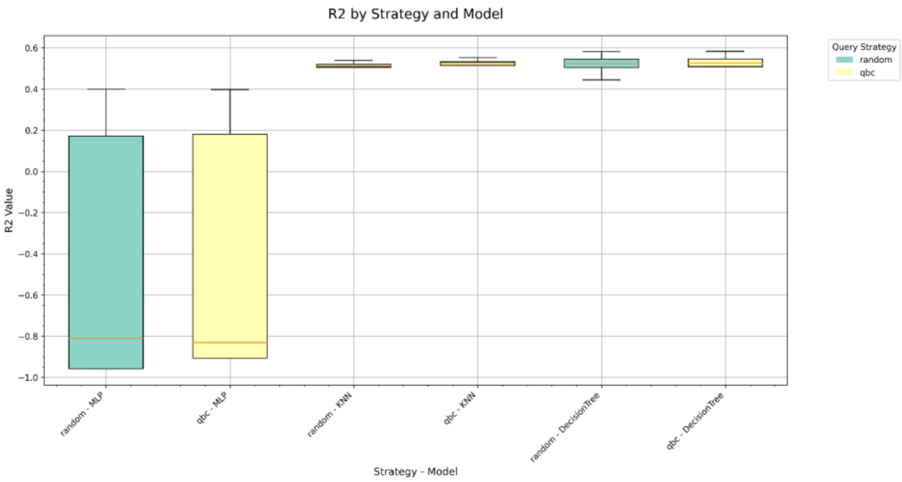
Minimum Capacity



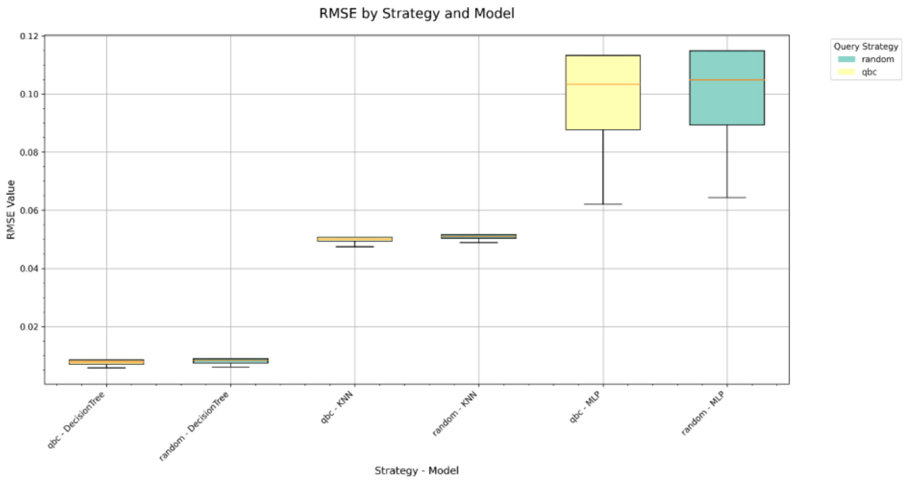


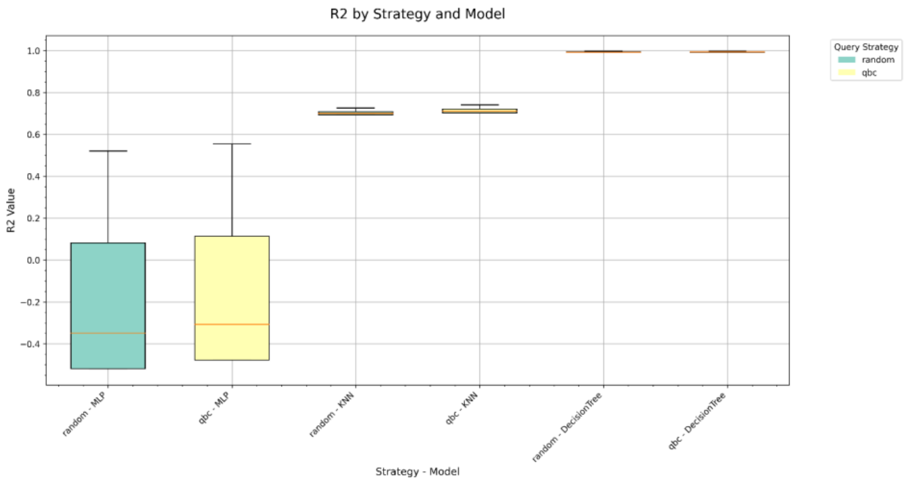
Abalone – Rings



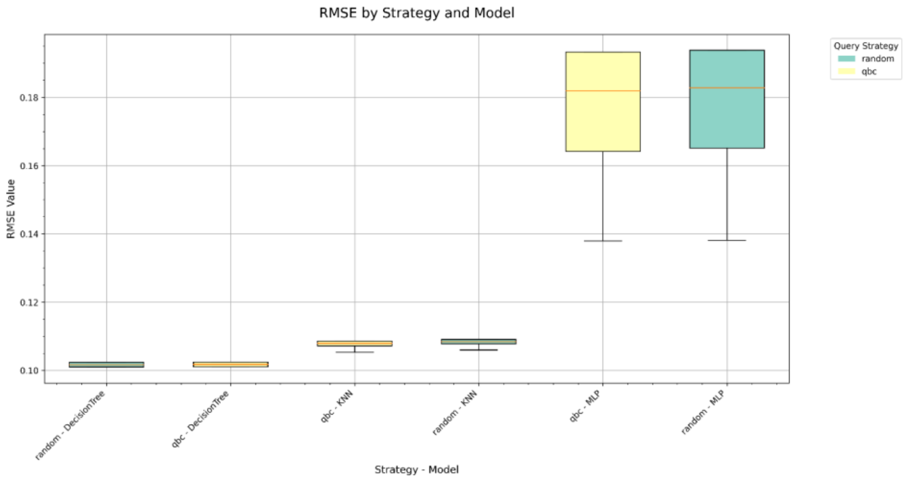


Capacity Throughput:

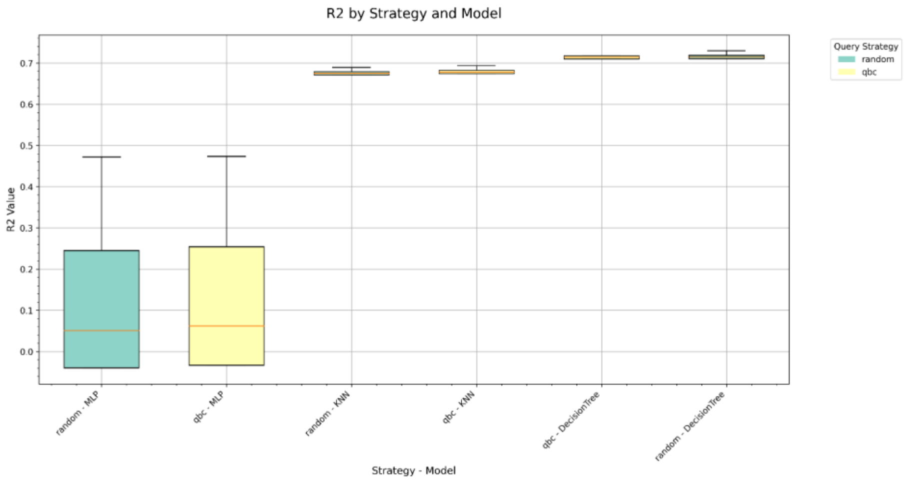

RMSE by Strategy and Model
RMSE Value
Strategy - Model
Query Strategy
random
qbc
qbc - DecisionTree
random - DecisionTree
qbc - KNN
random - KNN
qbc - MLP
random - MLP


R2 by Strategy and Model
R2 Value
Strategy - Model
Query Strategy
random
qbc
random - MLP
qbc - MLP
random - KNN
qbc - KNN
random - DecisionTree
qbc - DecisionTree

Voltage



RMSE by Strategy and Model
RMSE Value
0.10
0.12
0.14
0.16
0.18
random - DecisionTree
qbc - DecisionTree
qbc - KNN
random - KNN
qbc - MLP
random - MLP
Strategy - Model
Query Strategy
random
qbc



R2 by Strategy and Model
R2 Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
random - MLP
qbc - MLP
random - KNN
qbc - KNN
qbc - DecisionTree
random - DecisionTree
Strategy - Model
Query Strategy
random
qbc