iScience, Volume *27*

Supplemental information

Experimentally validated inverse design of multi-property Fe-Co-Ni alloys

Shakti P. Padhy, Varun Chaudhary, Yee-Fun Lim, Ruiming Zhu, Muang Thway, Kedar Hippalgaonkar, and Raju V. Ramanujan

S1. Distribution of the properties

Figure S1 shows the distribution of M_s (T), H_c (A/m), T_c (K), ρ ($\mu\Omega \cdot cm$), σ (MPa), σ_y (MPa), δ (%), H_V (HV) and *Cost* (USD/kg). It can be observed that the reported M_s, H_c, T_c, ρ, and δ values are more concentrated to values in the range 2-2.5 T, 0-500 A/m, 1150-1250 K, 0-20 $\mu\Omega \cdot cm$, and 0-10%, respectively. σ, σ_y, H_V, and *Cost* have a uniform distribution over large range of values but have comparative less data for respective higher values. From the perspective of the machine learning process for designing alloys, both attractive and unattractive property values are beneficial and reduce the bias in model building and prediction.

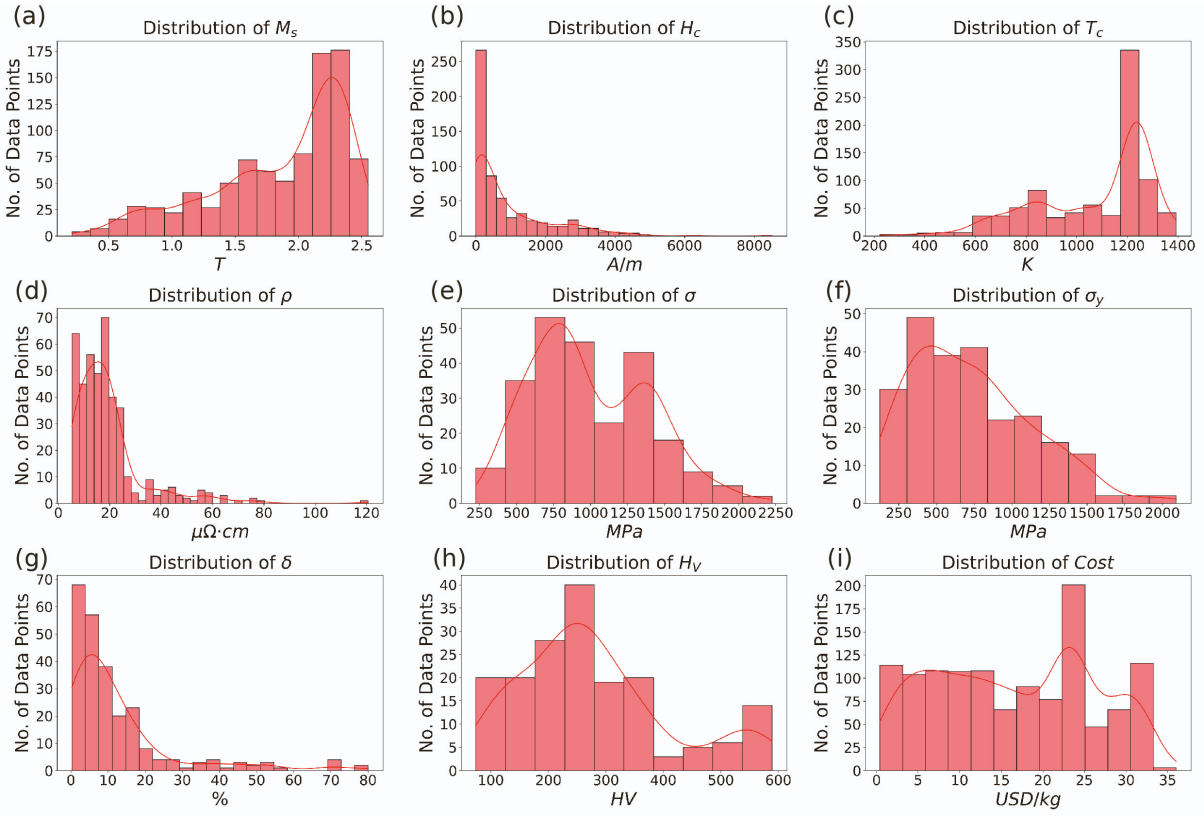

Figure S1. Distribution of (a)-(c) magnetic properties (M_s, H_c, T_c), (d) electrical resistivity (ρ), (e)-(h) mechanical properties (σ, σ_y, δ, H_V), and (w) Cost in the database. The solid continuous lines are kernel density estimations which show the probability density function of the data entries for each property.

S2. Hyperparameter optimization and model building for imputation strategy

The 12 types of ML models used in the imputation strategy, as explained in ***Section 3***, are:

(i) Linear Regression (Linear),

(ii) Ridge Regression (Ridge),

(iii) Lasso Regression (Lasso),
(iv) K-Nearest Neighbors Regression (KNNR),
(v) Support Vector Regression (SVR),
(vi) Decision Tree Regression (DTR),
(vii) Gradient Boosting Regression (GBR),
(viii) Random Forest Regression (RFR),
(ix) Extra Trees Regression (ETR),
(x) XGBoost Regression (XGBR),
(xi) Deep Neural Network Regression with same number of units in all the hidden layers (DNNR_rectangle), and
(xii) Deep Neural Network Regression with different number of units in all the hidden layers (DNNR_random).

A 5-fold cross validation hyperparameter optimization for all the models, except Linear and DNN models, was performed. For Lasso and Ridge, LassoCV and RidgeCV was used from *Scikit-Learn* respectively to optimize hyperparameters. By default, LassoCV and RidgeCV incorporate 5-fold cross validation to find the optimized alpha and for this training data (80% of total no. of data points) was used. For DNN models, the Hyperband[1] algorithm was used from *KerasTuner*[2] to obtain optimized hyperparameters. For other models, *gp_*minimize (Bayesian Optimization algorithm based on Gaussian process) from *Scikit-Optimize*[3] was used for hyperparameter optimization with a target to minimize the objective function of average of mean absolute error (MAE) for 5-fold cross validation on training data set (80% of total no. of data points). The BO algorithm followed in this function is described in **Table S1** below. According to the imputation strategy narrated in ***Section 2.2***, the list of various ML regression models used and their selected hyperparameters for optimization is shown in **Table S2**. In **Table S3**, optimized hyperparameters value for each step of the imputation strategy is reported.

Table S1. BO algorithm followed in gp_minimize from Scikit-Optimize[3]

1.	The gains are initialized, $g_i = 0$
2.	**for** t =1, 2, 3,….. **do**
3.	Each acquisition functions is optimized independently to propose a candidate X_i. A total of 10000 candidates are proposed.
4.	Out of 10000 candidates, the next X_{best} is chosen using $softmax(\eta g_i)$.

5. The surrogate model (Gaussian Process regression to model the defined objective function) is fitted with (X_{best}, Y_{best}).
6. The gains are updated, $g_i -= \mu(X_i)$
7. **end for**

Table S2. Different ML regression models and their hyperparameters used for optimization for imputation strategy

ML Regression Models	Hyperparameters for optimization
Linear	–
Ridge	alpha
Lasso	alpha
KNNR	n_neighbors, algorithm, leaf_size, p
SVR	kernel, gamma, C
DTR	criterion, splitter, max_depth
GBR	learning_rate, loss, n_estimators, max_depth, alpha
RFR	n_estimators, max_depth
ETR	n_estimators, max_depth
XGBR	eta, n_estimators, max_depth, subsample, colsample_bytree
DNNR_rectangle	num_dense_layers, units (same for all dense layers), activation, learning_rate, optimizer, batch_size, epochs
DNNR_random	units (for each dense layer), activation, learning_rate, optimizer, batch_size, epochs

Table S3. Optimized hyperparameters value for different ML regression models for imputation strategy

Mapping	ML Regression Models	Optimized Hyperparameters value
Composition to M_s (no. of data points = 907)	Linear	-
	Ridge	alpha = 7.478
	Lasso	alpha = 0.0024
	KNNR	n_neighbors = 9, algorithm = 'auto', leaf_size = 1, p = 1
	SVR	kernel = 'rbf', gamma = 0.0051, C = 100
	DTR	criterion = 'absolute_error', splitter = 'random', max_depth = 21
	GBR	learning_rate = 0.094, loss = 'quantile', n_estimators = 677, max_depth = 50, alpha = 0.55
	RFR	n_estimators = 2000, max_depth = 18
	ETR	n_estimators = 1337, max_depth = 26

	XGBR	eta = 0.149, n_estimators = 139, max_depth = 20, subsample = 1, colsample_bytree = 0.794
	DNNR_rectangle	num_dense_layers = 2, units (same for all dense layers) = 73, activation = 'relu', learning_rate = 0.0262, optimizer = 'Adam', batch_size = 43, epochs = 100
	DNNR_random	num_dense_layers (same as DNNR_rectangle) = 2, units (for each dense layer) = (156, 34), activation = 'relu', learning_rate = 0.008, optimizer = 'RMSprop', batch_size = 8, epochs = 100
Composition to $log_{10}H_c$ (no. of data points = 602)	Linear	-
	Ridge	alpha = 5.719
	Lasso	alpha = 0.102
	KNNR	n_neighbors = 3, algorithm = 'brute', leaf_size = 50, p = 1
	SVR	kernel = 'rbf', gamma = 0.194, C = 90.414
	DTR	criterion = 'absolute_error', splitter = 'random', max_depth = 18
	GBR	learning_rate = 0.036, loss = 'huber', n_estimators = 95, max_depth = 7, alpha = 0.012
	RFR	n_estimators = 227, max_depth = 11
	ETR	n_estimators = 347, max_depth = 12
	XGBR	eta = 0.022, n_estimators = 179, max_depth = 20, subsample = 0.433, colsample_bytree = 1
	DNNR_rectangle	num_dense_layers = 9, units (same for all dense layers) = 138, activation = 'tanh', learning_rate = 0.00065, optimizer = 'RMSprop', batch_size = 56, epochs = 100
	DNNR_random	num_dense_layers (same as DNNR_rectangle) = 9, units (for each dense layer) = (686, 837, 456, 609, 987, 493, 749, 423, 71), activation = 'elu', learning_rate = 0.000084, optimizer = 'RMSprop', batch_size = 71, epochs = 100
Composition to $log_{10}T_c$ (no. of data points = 874)	Linear	-
	Ridge	alpha = 48.901
	Lasso	alpha = 0.078
	KNNR	n_neighbors = 9, algorithm = 'kd_tree', leaf_size = 1, p = 1
	SVR	kernel = 'rbf', gamma = 0.132, C = 100
	DTR	criterion = 'squared_error', splitter = 'best', max_depth = 48
	GBR	learning_rate = 0.0086, loss = 'huber', n_estimators = 1000, max_depth = 41, alpha = 0.9
	RFR	n_estimators = 1, max_depth = 37

	ETR	n_estimators = 282, max_depth = 44
	XGBR	eta = 0.456, n_estimators = 148, max_depth = 20, subsample = 1, colsample_bytree = 1
	DNNR_rectangle	num_dense_layers = 4, units (same for all dense layers) = 48, activation = 'relu', learning_rate = 0.0027, optimizer = 'Adam', batch_size = 47, epochs = 100
	DNNR_random	num_dense_layers (same as DNNR_rectangle) = 4, units (for each dense layer) = (389, 111, 139, 249), activation = 'tanh', learning_rate = 0.0022, optimizer = 'Adam', batch_size = 95, epochs = 100
Composition to $log_{10}\rho$ (no. of data points = 421)	Linear	-
	Ridge	alpha = 21.868
	Lasso	alpha = 0.012
	KNNR	n_neighbors = 4, algorithm = 'ball_tree', leaf_size = 1, p = 1
	SVR	kernel = 'rbf', gamma = 0.574, C = 28.092
	DTR	criterion = 'squared_error', splitter = 'random', max_depth = 39
	GBR	learning_rate = 0.0175, loss = 'absolute_error', n_estimators = 924, max_depth = 39, alpha = 0.248
	RFR	n_estimators = 2000, max_depth = 25
	ETR	n_estimators = 383, max_depth = 16
	XGBR	eta = 0.082, n_estimators = 200, max_depth = 8, subsample = 0.403, colsample_bytree = 1
	DNNR_rectangle	num_dense_layers = 2, units (same for all dense layers) = 469, activation = 'tanh', learning_rate = 0.00019, optimizer = 'Adam', batch_size = 22, epochs = 100
	DNNR_random	num_dense_layers (same as DNNR_rectangle) = 62units (for each dense layer) = (828, 118), activation = 'tanh', learning_rate = 0.00036, optimizer = 'Adam', batch_size = 18, epochs = 100
Composition to $log_{10}\delta$ (no. of data points = 246)	Linear	-
	Ridge	alpha = 21.868
	Lasso	alpha = 0.000001
	KNNR	n_neighbors = 3, algorithm = 'auto', leaf_size = 50, p = 1
	SVR	kernel = 'rbf', gamma = 0.013, C = 100
	DTR	criterion = 'absolute_error', splitter = 'random', max_depth = 16
	GBR	learning_rate = 0.024, loss = 'absolute_error', n_estimators = 354, max_depth = 7, alpha = 0.028

	RFR	n_estimators = 390, max_depth = 40
	ETR	n_estimators = 7, max_depth = 12
	XGBR	eta = 0.0848, n_estimators = 168, max_depth = 16, subsample = 0.405, colsample_bytree = 0.213
	DNNR_rectangle	num_dense_layers = 6, units (same for all dense layers) = 431, activation = 'elu', learning_rate = 0.000084, optimizer = 'Adam', batch_size = 9, epochs = 100
	DNNR_random	num_dense_layers (same as DNNR_rectangle) = 6, units (for each dense layer) = (579, 627, 291, 431, 410, 877), activation = 'tanh', learning_rate = 0.00027, optimizer = 'Adam', batch_size = 32, epochs = 100
$log_{10}\sigma$ to $log_{10}H_V$ (no. of data points = 106)	Linear	-
	Ridge	alpha = 0.000001
	Lasso	alpha = 0.000001
	KNNR	n_neighbors = 5, algorithm = 'auto', leaf_size = 1, p = 1
	SVR	kernel = 'rbf', gamma = 0.022, C = 96.342
	DTR	criterion = 'squared_error', splitter = 'random', max_depth = 7
	GBR	learning_rate = 0.021, loss = 'squared_error', n_estimators = 947, max_depth = 1, alpha = 0.9
	RFR	n_estimators = 596, max_depth = 4
	ETR	n_estimators = 49, max_depth = 4
	XGBR	eta = 0.205, n_estimators = 92, max_depth = 1, subsample = 0.161, colsample_bytree = 0.005
$log_{10}\sigma_y$ to $log_{10}H_V$ (no. of data points = 89)	Linear	-
	Ridge	alpha = 0.391
	Lasso	alpha = 0.000001
	KNNR	n_neighbors = 12, algorithm = 'ball-tree', leaf_size = 45, p = 5
	SVR	kernel = 'sigmoid', gamma = 0.05, C = 100
	DTR	criterion = 'squared_error', splitter = 'random', max_depth = 9
	GBR	learning_rate = 0.058, loss = 'quantile', n_estimators = 521, max_depth = 34, alpha = 0.211
	RFR	n_estimators = 7, max_depth = 41
	ETR	n_estimators = 2000, max_depth = 6
	XGBR	eta = 1, n_estimators = 104, max_depth = 1, subsample = 1, colsample_bytree = 0.005
	Linear	-

Composition to $log_{10}H_V$ (no. of data points = 331)	Ridge	alpha = 37.395
	Lasso	alpha = 0.035
	KNNR	n_neighbors = 4, algorithm = 'kd_tree', leaf_size = 50, p = 1
	SVR	kernel = 'rbf', gamma = 0.085, C = 100
	DTR	criterion = 'absolute_error', splitter = 'random', max_depth = 17
	GBR	learning_rate = 0.052, loss = 'quantile', n_estimators = 280, max_depth = 50, alpha = 0.373
	RFR	n_estimators = 135, max_depth = 10
	ETR	n_estimators = 1424, max_depth = 15
	XGBR	eta = 0.033, n_estimators = 86, max_depth = 14, subsample = 0.858, colsample_bytree = 1
	DNNR_rectangle	num_dense_layers = 8, units (same for all dense layers) = 385, activation = 'tanh', learning_rate = 0.0021, optimizer = 'SGD', batch_size = 48, epochs = 100
	DNNR_random	num_dense_layers (same as DNNR_rectangle) = 8, units (for each dense layer) = (345, 623, 985, 170, 251, 416, 473, 858), activation = 'elu', learning_rate = 0.00028, optimizer = 'Adam', batch_size = 59, epochs = 100

S3. Hyperparameter optimization and performance of multi-property models

After filling the empty spaces in database using the imputation strategy, the filled database of 1208 data points was used to build multi-property predictive models as described in ***Section 2.3***. **Table S4** describes the ML regression models used and their selected hyperparameters for optimization. The optimization of hyperparameters is performed in a similar way as narrated in **Section S2.** For each of the ML regression models, the optimized hyperparameters value for the models RFR, ETR, DNNR_rectangle and DNNR_random and set hyperparameters value for the models NNR_3layers and NNR_4layers in **Table S5**.

Table S4. Different multi-property ML regression models and their hyperparameters used for optimization

ML Regression Models	Hyperparameters for optimization
RFR	n_estimators, max_depth, min_samples_split, min_samples_leaf

ETR	n_estimators, max_depth, min_samples_split, min_samples_leaf
DNNR_rectangle	num_dense_layers, units (same for all dense layers), activation, learning_rate, optimizer, batch_size, epochs
DNNR_random	units (for each dense layer), activation, learning_rate, optimizer, batch_size, epochs

Table S5. Optimized hyperparameters value for different multi-property ML regression models

ML Regression Models	Optimized/Set Hyperparameters value
RFR	n_estimators = 139, max_depth = 72, min_samples_split = 2, min_samples_leaf = 1
ETR	n_estimators = 230, max_depth = 81, min_samples_split = 2, min_samples_leaf = 1
NNR_3layers	num_dense_layers = 3, units = (128,64,32), activation = 'relu', initially learning_rate is 0.0001 but it is adaptive, optimizer = 'Adam', batch_size = 32, epochs = 96
NNR_4layers	num_dense_layers = 4, units = (256,128,64,32), activation = 'relu', initially learning_rate is 0.0001 but it is adaptive, optimizer = 'Adam', batch_size = 32, epochs = 96
DNNR_rectangle	num_dense_layers = 11, units (same for all dense layers) = 356, activation = 'tanh', learning_rate = 0.000092, optimizer = 'Adam', batch_size = 27, epochs = 100
DNNR_random	num_dense_layers (same as DNNR_rectangle) = 12, units (for each dense layer) = (1057, 936, 805, 1320, 954, 46, 705, 1173, 1474, 752, 259, 808), activation = 'tanh', learning_rate = 0.0063, optimizer = 'SGD', batch_size = 72, epochs = 100

The same hyperparameters value was used for multi-property ML regressors with composition and Wen alloy physical descriptors as input features except for epochs in NNR_3layers_Wen and NNR_4layers_Wen models which was set to 128.

As mentioned in ***Section 3.3.1***, the predicted vs. experimental values along with the performance metrics of each property for both training and testing set for the 4 models – ETR, ETR_Wen, NNR_4layers, and NNR_4layers_Wen, are shown in **Figure S2**, **Figure S3**, **Figure S4**, and **Figure S5**, respectively.

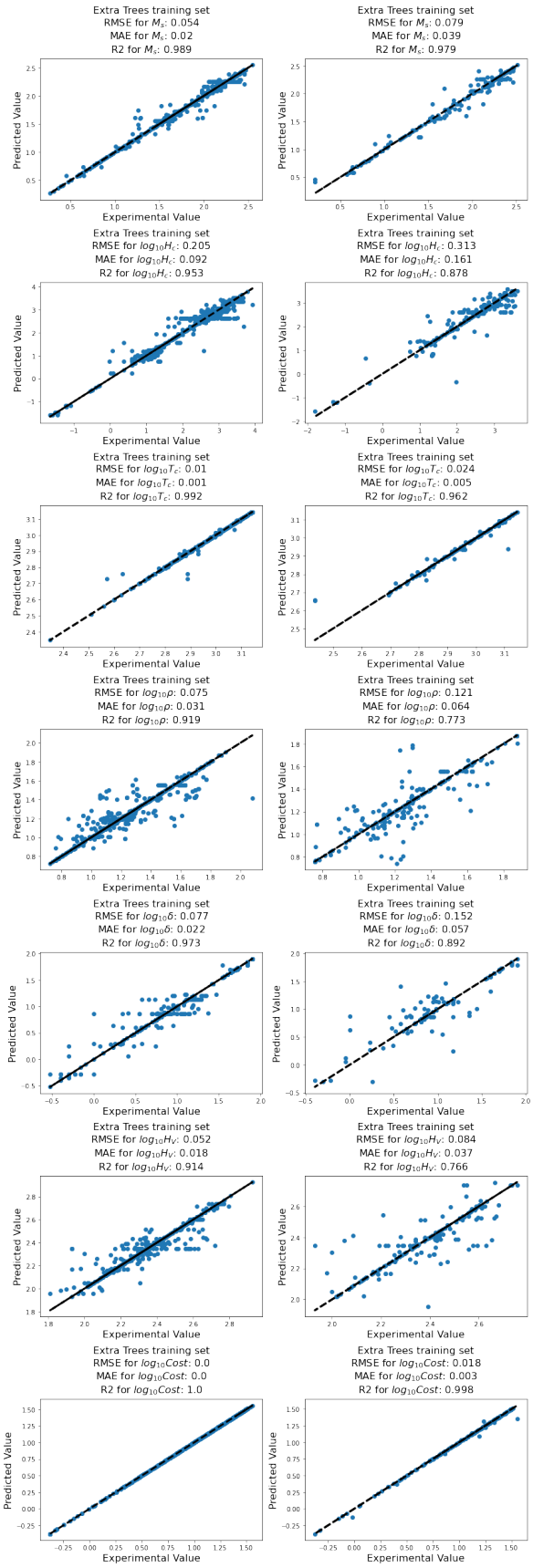

Figure S2. Predicted vs. Experimental values and performance metrics (RMSE, MAE, R^2) of M_s, $log_{10}H_c$, $log_{10}T_c$, $log_{10}\rho$, $log_{10}\delta$, $log_{10}H_V$, and $log_{10}Cost$ (in series from top to bottom) for training (all plots on left) and testing (all plots on right) sets for ETR model.

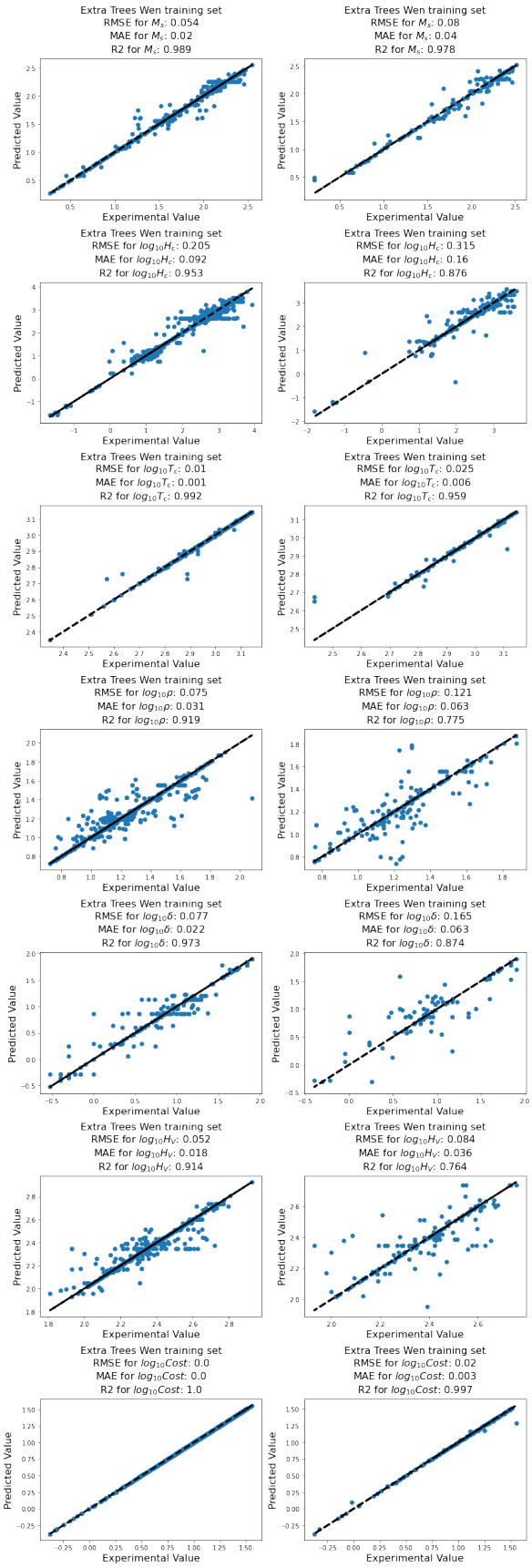

Figure S3. Predicted vs. Experimental values and performance metrics (RMSE, MAE, R^2) of M_s, $log_{10}H_c$, $log_{10}T_c$, $log_{10}\rho$, $log_{10}\delta$, $log_{10}H_V$, and $log_{10}Cost$ (in series from top to bottom) for training (all plots on left) and testing (all plots on right) sets for ETR_Wen model.

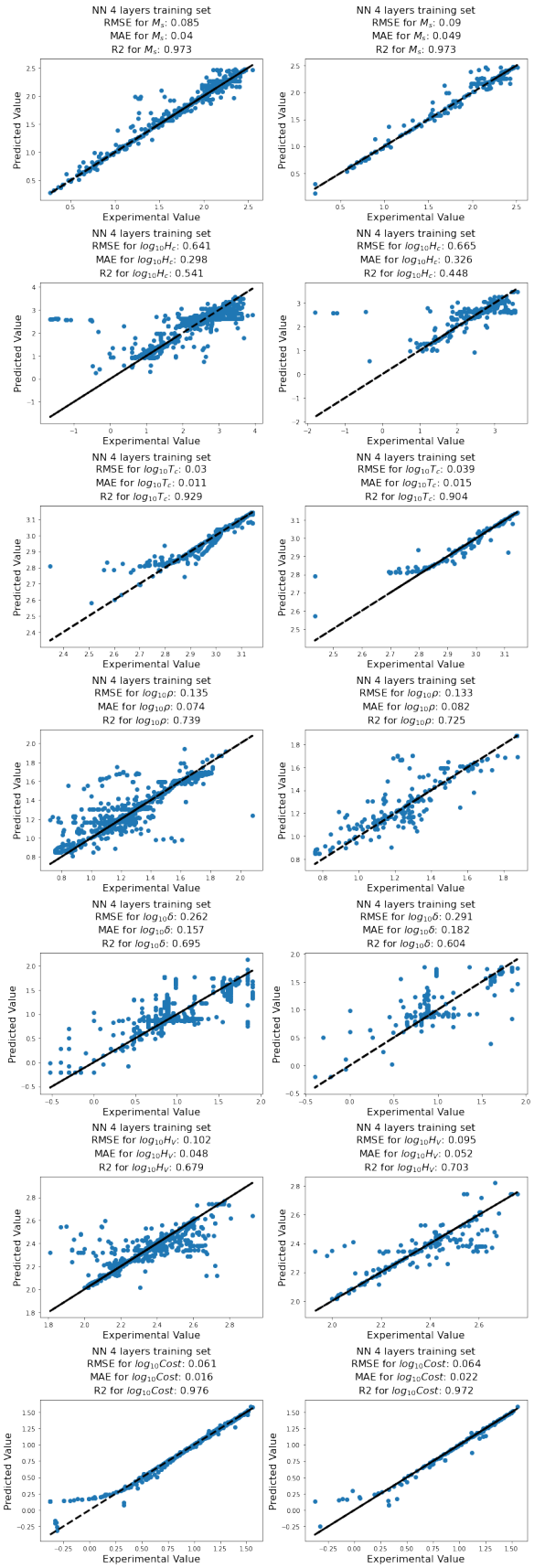

Figure S4. Predicted vs. Experimental values and performance metrics (RMSE, MAE, R^2) of M_s, $log_{10}H_c$, $log_{10}T_c$, $log_{10}\rho$, $log_{10}\delta$, $log_{10}H_V$, and $log_{10}Cost$ (in series from top to bottom) for training (all plots on left) and testing (all plots on right) sets for NNR_4layers model.

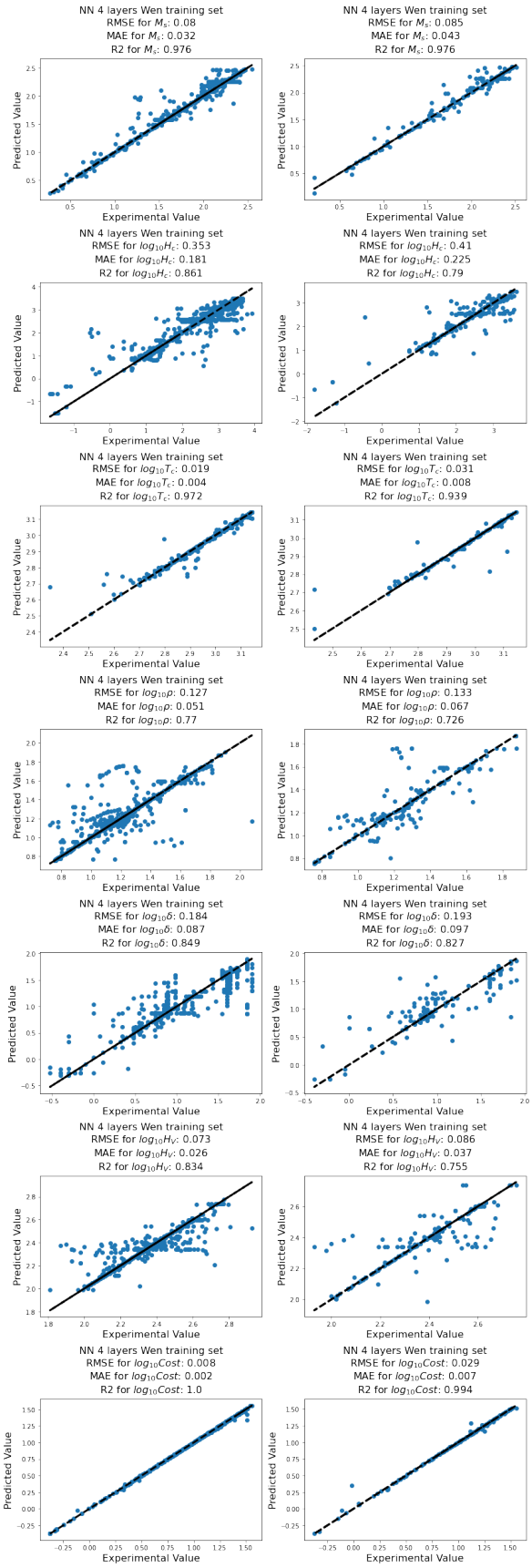

Figure S5. Predicted vs. Experimental values and performance metrics (RMSE, MAE, R^2) of M_s, $log_{10}H_c$, $log_{10}T_c$, $log_{10}\rho$, $log_{10}\delta$, $log_{10}H_V$, and $log_{10}Cost$ (in series from top to bottom) for training (all plots on left) and testing (all plots on right) sets for NNR_4layers_Wen model.

S4. Experimental Validation of Forward Design Model

The forward design model was experimentally validated using 40 compositions. Based on the predictions of the ETR model, absolute error between the predicted property value and experimental property value for each experimental composition and finally the MAE of the predictions were calculated and tabulated in **Table S6**, **Table S7**, **Table S8**, **Table S9**, **Table S10**, and **Table S11** for M_s, $log_{10}H_c$, $log_{10}T_c$, $log_{10}\rho$, $log_{10}H_V$, and $log_{10}Cost$ respectively. Similarly, the absolute errors and MAE were calculated and tabulated in Tables below for M_s, $log_{10}H_c$, $log_{10}T_c$, $log_{10}\rho$, $log_{10}H_V$, and $log_{10}Cost$, respectively, based on NNR_4layers_Wen model predictions. The deviation percentage of predictions from experimental values are calculated by the following formula:

$$Deviation\ \% = \frac{Predicted - Actual}{Predicted} \times 100$$

Using this formula, for each property the number of compositions within the ±10 % deviation range is estimated and mentioned after each table.

Table S6. Prediction, absolute error, and MAE of M_s by the ETR model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental M_s (T)	Predicted M_s (T)	Absolute error
1	10.47	39.7	49.83	1.37	1.297	0.073
2	10.47	29.76	59.77	1.18	1.212	0.032
3	10.47	19.83	69.7	1.08	1.103	0.023
4	10.46	9.91	79.63	0.95	0.967	0.017
5	20.82	9.86	69.32	1.31	1.179	0.131
6	31.07	9.81	59.12	1.29	1.404	0.114
7	41.22	9.76	49.02	1.41	1.595	0.185
8	34.49	32.69	32.82	1.6	1.91	0.31
9	41.28	48.9	9.82	1.97	2.154	0.184
10	31.13	59	9.87	1.91	2.204	0.294
11	20.87	69.2	9.93	1.88	2.025	0.145
12	10.49	79.53	9.98	1.87	1.882	0.012
13	10.49	69.55	19.96	1.69	1.85	0.16
14	10.48	59.6	29.92	1.6	1.567	0.033

15	10.48	49.64	39.88	1.5	1.394	0.106
16	51.32	38.91	9.77	2.12	2.215	0.095
17	61.26	29.03	9.71	2.09	2.261	0.171
18	71.09	19.25	9.66	2.03	2.271	0.241
19	80.82	9.57	9.61	1.92	2.268	0.348
20	71.06	9.62	19.32	1.69	2.192	0.502
21	61.21	9.67	29.12	1.43	1.95	0.52
22	51.26	9.72	39.02	1.47	1.737	0.267
23	71.08	14.43	14.49	2	2.229	0.229
24	61.24	19.34	19.42	1.94	2.175	0.235
25	51.3	24.3	24.4	1.83	2.132	0.302
26	41.25	29.32	29.43	1.77	2.051	0.281
27	31.1	34.38	34.52	1.65	1.9	0.25
28	20.84	39.5	39.66	1.51	1.578	0.068
29	41.27	39.1	19.63	1.8	2.05	0.25
30	36.19	34.29	29.52	1.75	1.999	0.249
31	31.1	29.46	39.44	1.58	1.721	0.141
32	25.97	24.61	49.42	1.44	1.456	0.016
33	20.82	19.73	59.45	1.35	1.257	0.093
34	15.65	14.83	69.52	1.17	1.144	0.026
35	41.23	19.54	39.23	1.64	1.72	0.08
36	36.18	29.39	34.43	1.71	1.864	0.154
37	31.1	39.3	29.6	1.71	2.051	0.341
38	25.99	49.27	24.74	1.73	2.082	0.352
39	20.86	59.29	19.85	1.73	1.826	0.096
40	15.69	69.38	14.93	1.76	1.952	0.192
			Mean Absolute Error (MAE)			0.183

The ETR model was able to predict the M_s of 22 compositions within ±10% deviation range compared to experimental values.

Table S7. Prediction, absolute error, and MAE of $log_{10}H_c$ by ETR model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}H_c$ (log_{10}(A/m))	Predicted $log_{10}H_c$ (log_{10}(A/m))	Absolute error
1	10.47	39.7	49.83	3.144	2.771	0.373
2	10.47	29.76	59.77	2.98	2.567	0.413
3	10.47	19.83	69.7	2.961	2.353	0.608
4	10.46	9.91	79.63	2.874	1.719	1.155
5	20.82	9.86	69.32	2.591	1.605	0.986
6	31.07	9.81	59.12	2.679	1.103	1.576
7	41.22	9.76	49.02	2.223	1.089	1.134
8	34.49	32.69	32.82	2.793	2.58	0.213
9	41.28	48.9	9.82	3.062	3.494	0.432
10	31.13	59	9.87	2.758	3.485	0.727
11	20.87	69.2	9.93	2.7	3.112	0.412
12	10.49	79.53	9.98	2.835	3.037	0.202
13	10.49	69.55	19.96	3.001	2.796	0.205
14	10.48	59.6	29.92	2.845	2.839	0.006
15	10.48	49.64	39.88	2.85	2.861	0.011
16	51.32	38.91	9.77	3.053	3.111	0.058
17	61.26	29.03	9.71	3.053	2.863	0.19
18	71.09	19.25	9.66	2.973	2.466	0.507
19	80.82	9.57	9.61	3.091	2.713	0.378
20	71.06	9.62	19.32	3.166	2.739	0.427
21	61.21	9.67	29.12	3.062	2.078	0.984
22	51.26	9.72	39.02	2.85	1.601	1.249
23	71.08	14.43	14.49	3.322	2.626	0.696
24	61.24	19.34	19.42	3.182	2.694	0.488
25	51.3	24.3	24.4	2.942	2.947	0.005
26	41.25	29.32	29.43	2.554	2.957	0.403
27	31.1	34.38	34.52	2.457	2.388	0.069
28	20.84	39.5	39.66	2.564	2.408	0.156
29	41.27	39.1	19.63	2.922	3.122	0.2

30	36.19	34.29	29.52	2.573	2.896	0.323
31	31.1	29.46	39.44	2.625	1.577	1.048
32	25.97	24.61	49.42	2.608	0.904	1.704
33	20.82	19.73	59.45	2.672	2.245	0.427
34	15.65	14.83	69.52	2.83	2.063	0.767
35	41.23	19.54	39.23	2.686	1.605	1.081
36	36.18	29.39	34.43	2.625	2.409	0.216
37	31.1	39.3	29.6	2.686	2.946	0.26
38	25.99	49.27	24.74	2.657	2.889	0.232
39	20.86	59.29	19.85	2.617	2.618	0.001
40	15.69	69.38	14.93	2.733	2.992	0.259
				Mean Absolute Error (MAE)		0.515

The ETR model was able to predict the $log_{10}H_c$ of 16 compositions within ±10% deviation range compared to experimental values.

Table S8. Prediction, absolute error, and MAE of $log_{10}T_c$ by ETR model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}T_c$ (log_{10}(K))	Predicted $log_{10}T_c$ (log_{10}(K))	Absolute error
1	10.47	39.7	49.83	3.032	3.042	0.01
2	10.47	29.76	59.77	3.005	3.01	0.005
3	10.47	19.83	69.7	2.971	2.962	0.009
4	10.46	9.91	79.63	2.948	2.899	0.049
5	20.82	9.86	69.32	2.954	2.87	0.084
6	31.07	9.81	59.12	2.945	2.926	0.019
7	41.22	9.76	49.02	2.951	2.93	0.021
8	34.49	32.69	32.82	3.009	2.991	0.018
9	41.28	48.9	9.82	3.063	3.067	0.004
10	31.13	59	9.87	3.066	3.092	0.026
11	20.87	69.2	9.93	3.103	3.137	0.034
12	10.49	79.53	9.98	3.11	3.111	0.001
13	10.49	69.55	19.96	3.09	3.105	0.015

14	10.48	59.6	29.92	3.077	3.081	0.004
15	10.48	49.64	39.88	3.06	3.057	0.003
16	51.32	38.91	9.77	3.069	3.072	0.003
17	61.26	29.03	9.71	3.064	3.056	0.008
18	71.09	19.25	9.66	3.068	3.032	0.036
19	80.82	9.57	9.61	3.018	3.006	0.012
20	71.06	9.62	19.32	2.998	2.967	0.031
21	61.21	9.67	29.12	2.943	2.848	0.095
22	51.26	9.72	39.02	2.94	2.884	0.056
23	71.08	14.43	14.49	3.007	2.999	0.008
24	61.24	19.34	19.42	3	2.992	0.008
25	51.3	24.3	24.4	3.015	2.987	0.028
26	41.25	29.32	29.43	3.006	2.986	0.02
27	31.1	34.38	34.52	3.037	2.996	0.041
28	20.84	39.5	39.66	3.037	3.02	0.017
29	41.27	39.1	19.63	3.024	3.012	0.012
30	36.19	34.29	29.52	3.003	3.001	0.002
31	31.1	29.46	39.44	3.01	2.98	0.03
32	25.97	24.61	49.42	3.001	2.966	0.035
33	20.82	19.73	59.45	2.988	2.951	0.037
34	15.65	14.83	69.52	2.96	2.874	0.086
35	41.23	19.54	39.23	2.975	2.939	0.036
36	36.18	29.39	34.43	3.024	2.978	0.046
37	31.1	39.3	29.6	3.028	3.015	0.013
38	25.99	49.27	24.74	3.064	3.043	0.021
39	20.86	59.29	19.85	3.064	3.091	0.027
40	15.69	69.38	14.93	3.088	3.114	0.026
				Mean Absolute Error (MAE)		0.026

The ETR model was able to predict the $log_{10}T_c$ of all 40 compositions within ±10% deviation range compared to experimental values.

Table S9. Prediction, absolute error, and MAE of $log_{10}\ \rho$ by ETR model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}\rho$ (log_{10}(μΩ·cm))	Predicted $log_{10}\rho$ (log_{10}(μΩ·cm))	Absolute error
1	10.47	39.7	49.83	1.422	1.042	0.38
2	10.47	29.76	59.77	1.31	1.056	0.254
3	10.47	19.83	69.7	1.292	1.055	0.237
4	10.46	9.91	79.63	1.281	1.134	0.147
5	20.82	9.86	69.32	1.352	1.164	0.188
6	31.07	9.81	59.12	1.423	1.227	0.196
7	41.22	9.76	49.02	1.589	1.372	0.217
8	34.49	32.69	32.82	1.423	1.261	0.162
9	41.28	48.9	9.82	1.246	1.069	0.177
10	31.13	59	9.87	1.193	1.122	0.071
11	20.87	69.2	9.93	1.188	1.153	0.035
12	10.49	79.53	9.98	1.196	1.239	0.043
13	10.49	69.55	19.96	1.228	1.38	0.152
14	10.48	59.6	29.92	1.369	1.147	0.222
15	10.48	49.64	39.88	1.34	1.375	0.035
16	51.32	38.91	9.77	1.217	1.148	0.069
17	61.26	29.03	9.71	1.21	1.23	0.02
18	71.09	19.25	9.66	1.279	1.342	0.063
19	80.82	9.57	9.61	1.369	1.396	0.027
20	71.06	9.62	19.32	1.479	1.438	0.041
21	61.21	9.67	29.12	1.663	1.437	0.226
22	51.26	9.72	39.02	1.667	1.437	0.23
23	71.08	14.43	14.49	1.501	1.396	0.105
24	61.24	19.34	19.42	1.435	1.267	0.168
25	51.3	24.3	24.4	1.415	1.252	0.163
26	41.25	29.32	29.43	1.412	1.262	0.15
27	31.1	34.38	34.52	1.425	1.268	0.157
28	20.84	39.5	39.66	1.528	1.253	0.275
29	41.27	39.1	19.63	1.516	1.171	0.345

30	36.19	34.29	29.52	1.41	1.272	0.138
31	31.1	29.46	39.44	1.367	1.247	0.12
32	25.97	24.61	49.42	1.346	1.299	0.047
33	20.82	19.73	59.45	1.336	1.098	0.238
34	15.65	14.83	69.52	1.375	1.111	0.264
35	41.23	19.54	39.23	1.54	1.33	0.21
36	36.18	29.39	34.43	1.391	1.254	0.137
37	31.1	39.3	29.6	1.344	1.293	0.051
38	25.99	49.27	24.74	1.358	1.27	0.088
39	20.86	59.29	19.85	1.265	1.263	0.002
40	15.69	69.38	14.93	1.258	1.239	0.019
				Mean Absolute Error (MAE)		0.147

The ETR model was able to predict the $log_{10}\rho$ of 16 compositions within ±10% deviation range compared to experimental values.

Table S10. Prediction, absolute error, and MAE of $log_{10}H_V$ by ETR model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}H_V$ (log_{10}(HV))	Predicted $log_{10}H_V$ (log_{10}(HV))	Absolute error
1	10.47	39.7	49.83	2.134	2.168	0.034
2	10.47	29.76	59.77	2.187	2.205	0.018
3	10.47	19.83	69.7	2.193	2.21	0.017
4	10.46	9.91	79.63	2.198	2.294	0.096
5	20.82	9.86	69.32	2.25	2.347	0.097
6	31.07	9.81	59.12	2.27	2.269	0.001
7	41.22	9.76	49.02	2.318	2.325	0.007
8	34.49	32.69	32.82	2.345	2.519	0.174
9	41.28	48.9	9.82	2.432	2.572	0.14
10	31.13	59	9.87	2.424	2.466	0.042
11	20.87	69.2	9.93	2.239	2.42	0.181
12	10.49	79.53	9.98	2.202	2.324	0.122
13	10.49	69.55	19.96	2.189	2.411	0.222

14	10.48	59.6	29.92	2.192	2.213	0.021
15	10.48	49.64	39.88	2.238	2.411	0.173
16	51.32	38.91	9.77	2.496	2.64	0.144
17	61.26	29.03	9.71	2.444	2.556	0.112
18	71.09	19.25	9.66	2.373	2.434	0.061
19	80.82	9.57	9.61	2.335	2.496	0.161
20	71.06	9.62	19.32	2.381	2.489	0.108
21	61.21	9.67	29.12	2.439	2.469	0.03
22	51.26	9.72	39.02	2.293	2.451	0.158
23	71.08	14.43	14.49	2.548	2.461	0.087
24	61.24	19.34	19.42	2.609	2.558	0.051
25	51.3	24.3	24.4	2.417	2.592	0.175
26	41.25	29.32	29.43	2.328	2.562	0.234
27	31.1	34.38	34.52	2.344	2.485	0.141
28	20.84	39.5	39.66	2.263	2.301	0.038
29	41.27	39.1	19.63	2.364	2.597	0.233
30	36.19	34.29	29.52	2.327	2.534	0.207
31	31.1	29.46	39.44	2.358	2.461	0.103
32	25.97	24.61	49.42	2.286	2.272	0.014
33	20.82	19.73	59.45	2.321	2.161	0.16
34	15.65	14.83	69.52	2.204	2.349	0.145
35	41.23	19.54	39.23	2.316	2.476	0.16
36	36.18	29.39	34.43	2.312	2.524	0.212
37	31.1	39.3	29.6	2.303	2.491	0.188
38	25.99	49.27	24.74	2.302	2.476	0.174
39	20.86	59.29	19.85	2.242	2.431	0.189
40	15.69	69.38	14.93	2.252	2.375	0.123
				Mean Absolute Error (MAE)		0.119

The ETR model was able to predict the $log_{10}H_V$ of all 40 compositions within ±10% deviation range compared to experimental values.

Table S11. Prediction, absolute error, and MAE of $log_{10}Cost$ by ETR model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}Cost$ (log_{10}($/kg))	Predicted $log_{10}Cost$ (log_{10}($/kg))	Absolute error
1	10.47	39.7	49.83	1.303	1.307	0.004
2	10.47	29.76	59.77	1.261	1.265	0.004
3	10.47	19.83	69.7	1.213	1.205	0.008
4	10.46	9.91	79.63	1.16	1.146	0.014
5	20.82	9.86	69.32	1.117	1.107	0.01
6	31.07	9.81	59.12	1.07	1.048	0.022
7	41.22	9.76	49.02	1.017	1.001	0.016
8	34.49	32.69	32.82	1.196	1.194	0.002
9	41.28	48.9	9.82	1.254	1.254	0
10	31.13	59	9.87	1.326	1.325	0.001
11	20.87	69.2	9.93	1.388	1.388	0
12	10.49	79.53	9.98	1.442	1.43	0.012
13	10.49	69.55	19.96	1.411	1.413	0.002
14	10.48	59.6	29.92	1.378	1.374	0.004
15	10.48	49.64	39.88	1.342	1.342	0
16	51.32	38.91	9.77	1.168	1.17	0.002
17	61.26	29.03	9.71	1.06	1.073	0.013
18	71.09	19.25	9.66	0.916	0.878	0.038
19	80.82	9.57	9.61	0.7	0.6	0.1
20	71.06	9.62	19.32	0.803	0.759	0.044
21	61.21	9.67	29.12	0.887	0.825	0.062
22	51.26	9.72	39.02	0.957	0.947	0.01
23	71.08	14.43	14.49	0.863	0.798	0.065
24	61.24	19.34	19.42	0.982	0.984	0.002
25	51.3	24.3	24.4	1.075	1.077	0.002
26	41.25	29.32	29.43	1.152	1.159	0.007
27	31.1	34.38	34.52	1.217	1.207	0.01
28	20.84	39.5	39.66	1.273	1.277	0.004

29	41.27	39.1	19.63	1.206	1.204	0.002
30	36.19	34.29	29.52	1.199	1.202	0.003
31	31.1	29.46	39.44	1.191	1.181	0.01
32	25.97	24.61	49.42	1.184	1.173	0.011
33	20.82	19.73	59.45	1.176	1.174	0.002
34	15.65	14.83	69.52	1.168	1.152	0.016
35	41.23	19.54	39.23	1.09	1.113	0.023
36	36.18	29.39	34.43	1.172	1.172	0
37	31.1	39.3	29.6	1.241	1.242	0.001
38	25.99	49.27	24.74	1.301	1.301	0
39	20.86	59.29	19.85	1.353	1.355	0.002
40	15.69	69.38	14.93	1.4	1.397	0.003
Mean Absolute Error (MAE)						0.013

The ETR model was able to predict the $log_{10}Cost$ of 39 compositions within ±10% deviation range compared to experimental values.

Table S12. Prediction, absolute error, and MAE of M_s by NNR_4layers_Wen model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental M_s (T)	Predicted M_s (T)	Absolute error
1	10.47	39.7	49.83	1.37	1.276	0.094
2	10.47	29.76	59.77	1.18	1.168	0.012
3	10.47	19.83	69.7	1.08	1.054	0.026
4	10.46	9.91	79.63	0.95	0.923	0.027
5	20.82	9.86	69.32	1.31	1.148	0.162
6	31.07	9.81	59.12	1.29	1.415	0.125
7	41.22	9.76	49.02	1.41	1.65	0.24
8	34.49	32.69	32.82	1.6	1.801	0.201
9	41.28	48.9	9.82	1.97	2.142	0.172
10	31.13	59	9.87	1.91	2.101	0.191
11	20.87	69.2	9.93	1.88	2.012	0.132
12	10.49	79.53	9.98	1.87	1.885	0.015

13	10.49	69.55	19.96	1.69	1.703	0.013
14	10.48	59.6	29.92	1.6	1.528	0.072
15	10.48	49.64	39.88	1.5	1.39	0.11
16	51.32	38.91	9.77	2.12	2.219	0.099
17	61.26	29.03	9.71	2.09	2.228	0.138
18	71.09	19.25	9.66	2.03	2.229	0.199
19	80.82	9.57	9.61	1.92	2.278	0.358
20	71.06	9.62	19.32	1.69	2.225	0.535
21	61.21	9.67	29.12	1.43	2.102	0.672
22	51.26	9.72	39.02	1.47	1.892	0.422
23	71.08	14.43	14.49	2	2.228	0.228
24	61.24	19.34	19.42	1.94	2.177	0.237
25	51.3	24.3	24.4	1.83	2.101	0.271
26	41.25	29.32	29.43	1.77	1.942	0.172
27	31.1	34.38	34.52	1.65	1.731	0.081
28	20.84	39.5	39.66	1.51	1.474	0.036
29	41.27	39.1	19.63	1.8	2.007	0.207
30	36.19	34.29	29.52	1.75	1.864	0.114
31	31.1	29.46	39.44	1.58	1.663	0.083
32	25.97	24.61	49.42	1.44	1.457	0.017
33	20.82	19.73	59.45	1.35	1.238	0.112
34	15.65	14.83	69.52	1.17	1.081	0.089
35	41.23	19.54	39.23	1.64	1.791	0.151
36	36.18	29.39	34.43	1.71	1.8	0.09
37	31.1	39.3	29.6	1.71	1.782	0.072
38	25.99	49.27	24.74	1.73	1.748	0.018
39	20.86	59.29	19.85	1.73	1.793	0.063
40	15.69	69.38	14.93	1.76	1.85	0.09
			Mean Absolute Error (MAE)			0.154

The NNR_4layers_Wen model was able to predict the M_s of 29 compositions within ±10% deviation range compared to experimental values.

Table S13. Prediction, absolute error, and MAE of $log_{10}H_c$ by NNR_4layers_Wen model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}H_c$ (log_{10}(A/m))	Predicted $log_{10}H_c$ (log_{10}(A/m))	Absolute error
1	10.47	39.7	49.83	3.144	2.656	0.488
2	10.47	29.76	59.77	2.98	2.426	0.554
3	10.47	19.83	69.7	2.961	2.17	0.791
4	10.46	9.91	79.63	2.874	1.826	1.048
5	20.82	9.86	69.32	2.591	1.74	0.851
6	31.07	9.81	59.12	2.679	1.466	1.213
7	41.22	9.76	49.02	2.223	1.459	0.764
8	34.49	32.69	32.82	2.793	2.521	0.272
9	41.28	48.9	9.82	3.062	3.285	0.223
10	31.13	59	9.87	2.758	3.237	0.479
11	20.87	69.2	9.93	2.7	3.125	0.425
12	10.49	79.53	9.98	2.835	2.962	0.127
13	10.49	69.55	19.96	3.001	2.959	0.042
14	10.48	59.6	29.92	2.845	2.927	0.082
15	10.48	49.64	39.88	2.85	2.88	0.03
16	51.32	38.91	9.77	3.053	3.189	0.136
17	61.26	29.03	9.71	3.053	3.029	0.024
18	71.09	19.25	9.66	2.973	2.919	0.054
19	80.82	9.57	9.61	3.091	2.829	0.262
20	71.06	9.62	19.32	3.166	2.597	0.569
21	61.21	9.67	29.12	3.062	2.084	0.978
22	51.26	9.72	39.02	2.85	1.684	1.166
23	71.08	14.43	14.49	3.322	2.796	0.526
24	61.24	19.34	19.42	3.182	2.724	0.458
25	51.3	24.3	24.4	2.942	2.73	0.212
26	41.25	29.32	29.43	2.554	2.631	0.077
27	31.1	34.38	34.52	2.457	2.487	0.03
28	20.84	39.5	39.66	2.564	2.64	0.076
29	41.27	39.1	19.63	2.922	3.243	0.321

30	36.19	34.29	29.52	2.573	2.708	0.135
31	31.1	29.46	39.44	2.625	2.233	0.392
32	25.97	24.61	49.42	2.608	2.014	0.594
33	20.82	19.73	59.45	2.672	2	0.672
34	15.65	14.83	69.52	2.83	1.993	0.837
35	41.23	19.54	39.23	2.686	1.963	0.723
36	36.18	29.39	34.43	2.625	2.382	0.243
37	31.1	39.3	29.6	2.686	2.739	0.053
38	25.99	49.27	24.74	2.657	2.984	0.327
39	20.86	59.29	19.85	2.617	3.086	0.469
40	15.69	69.38	14.93	2.733	3.063	0.33
				Mean Absolute Error (MAE)		0.426

The NNR_4layers_Wen model was able to predict the $log_{10}H_c$ of 16 compositions within ±10% deviation range compared to experimental values.

Table S14. Prediction, absolute error, and MAE of $log_{10}T_c$ by NNR_4layers_Wen model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}T_c$ (log_{10}(K))	Predicted $log_{10}T_c$ (log_{10}(K))	Absolute error
1	10.47	39.7	49.83	3.032	3.045	0.013
2	10.47	29.76	59.77	3.005	3.018	0.013
3	10.47	19.83	69.7	2.971	2.984	0.013
4	10.46	9.91	79.63	2.948	2.94	0.008
5	20.82	9.86	69.32	2.954	2.957	0.003
6	31.07	9.81	59.12	2.945	2.941	0.004
7	41.22	9.76	49.02	2.951	2.922	0.029
8	34.49	32.69	32.82	3.009	2.992	0.017
9	41.28	48.9	9.82	3.063	3.049	0.014
10	31.13	59	9.87	3.066	3.066	0
11	20.87	69.2	9.93	3.103	3.11	0.007
12	10.49	79.53	9.98	3.11	3.124	0.014
13	10.49	69.55	19.96	3.09	3.105	0.015

14	10.48	59.6	29.92	3.077	3.087	0.01
15	10.48	49.64	39.88	3.06	3.069	0.009
16	51.32	38.91	9.77	3.069	3.046	0.023
17	61.26	29.03	9.71	3.064	3.035	0.029
18	71.09	19.25	9.66	3.068	3.013	0.055
19	80.82	9.57	9.61	3.018	3.015	0.003
20	71.06	9.62	19.32	2.998	2.958	0.04
21	61.21	9.67	29.12	2.943	2.903	0.04
22	51.26	9.72	39.02	2.94	2.91	0.03
23	71.08	14.43	14.49	3.007	2.987	0.02
24	61.24	19.34	19.42	3	2.977	0.023
25	51.3	24.3	24.4	3.015	2.984	0.031
26	41.25	29.32	29.43	3.006	2.984	0.022
27	31.1	34.38	34.52	3.037	2.998	0.039
28	20.84	39.5	39.66	3.037	3.026	0.011
29	41.27	39.1	19.63	3.024	3.011	0.013
30	36.19	34.29	29.52	3.003	2.995	0.008
31	31.1	29.46	39.44	3.01	2.984	0.026
32	25.97	24.61	49.42	3.001	2.974	0.027
33	20.82	19.73	59.45	2.988	2.98	0.008
34	15.65	14.83	69.52	2.96	2.969	0.009
35	41.23	19.54	39.23	2.975	2.946	0.029
36	36.18	29.39	34.43	3.024	2.982	0.042
37	31.1	39.3	29.6	3.028	3.006	0.022
38	25.99	49.27	24.74	3.064	3.042	0.022
39	20.86	59.29	19.85	3.064	3.085	0.021
40	15.69	69.38	14.93	3.088	3.107	0.019
				Mean Absolute Error (MAE)		0.02

The NNR_4layers_Wen model was able to predict the $log_{10}T_c$ of all 40 compositions within ±10% deviation range compared to experimental values.

Table S15. Prediction, absolute error, and MAE of $log_{10}\ \rho$ by NNR_4layers_Wen model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}\rho$ (log_{10}(μΩ·cm))	Predicted $log_{10}\rho$ (log_{10}(μΩ·cm))	Absolute error
1	10.47	39.7	49.83	1.422	1.256	0.166
2	10.47	29.76	59.77	1.31	1.213	0.097
3	10.47	19.83	69.7	1.292	1.188	0.104
4	10.46	9.91	79.63	1.281	1.174	0.107
5	20.82	9.86	69.32	1.352	1.186	0.166
6	31.07	9.81	59.12	1.423	1.248	0.175
7	41.22	9.76	49.02	1.589	1.346	0.243
8	34.49	32.69	32.82	1.423	1.226	0.197
9	41.28	48.9	9.82	1.246	1.035	0.211
10	31.13	59	9.87	1.193	1.077	0.116
11	20.87	69.2	9.93	1.188	1.176	0.012
12	10.49	79.53	9.98	1.196	1.215	0.019
13	10.49	69.55	19.96	1.228	1.291	0.063
14	10.48	59.6	29.92	1.369	1.319	0.05
15	10.48	49.64	39.88	1.34	1.306	0.034
16	51.32	38.91	9.77	1.217	1.049	0.168
17	61.26	29.03	9.71	1.21	1.151	0.059
18	71.09	19.25	9.66	1.279	1.29	0.011
19	80.82	9.57	9.61	1.369	1.427	0.058
20	71.06	9.62	19.32	1.479	1.412	0.067
21	61.21	9.67	29.12	1.663	1.402	0.261
22	51.26	9.72	39.02	1.667	1.386	0.281
23	71.08	14.43	14.49	1.501	1.362	0.139
24	61.24	19.34	19.42	1.435	1.269	0.166
25	51.3	24.3	24.4	1.415	1.244	0.171
26	41.25	29.32	29.43	1.412	1.246	0.166
27	31.1	34.38	34.52	1.425	1.229	0.196
28	20.84	39.5	39.66	1.528	1.26	0.268
29	41.27	39.1	19.63	1.516	1.165	0.351

30	36.19	34.29	29.52	1.41	1.222	0.188
31	31.1	29.46	39.44	1.367	1.235	0.132
32	25.97	24.61	49.42	1.346	1.231	0.115
33	20.82	19.73	59.45	1.336	1.194	0.142
34	15.65	14.83	69.52	1.375	1.192	0.183
35	41.23	19.54	39.23	1.54	1.312	0.228
36	36.18	29.39	34.43	1.391	1.241	0.15
37	31.1	39.3	29.6	1.344	1.225	0.119
38	25.99	49.27	24.74	1.358	1.223	0.135
39	20.86	59.29	19.85	1.265	1.242	0.023
40	15.69	69.38	14.93	1.258	1.251	0.007
				Mean Absolute Error (MAE)		0.139

The NNR_4layers_Wen model was able to predict the $log_{10}\rho$ of 16 compositions within ±10% deviation range compared to experimental values.

Table S16. Prediction, absolute error, and MAE of $log_{10}H_V$ by NNR_4layers_Wen model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}H_V$ (log_{10}(HV))	Predicted $log_{10}H_V$ (log_{10}(HV))	Absolute error
1	10.47	39.7	49.83	2.134	2.377	0.243
2	10.47	29.76	59.77	2.187	2.359	0.172
3	10.47	19.83	69.7	2.193	2.337	0.144
4	10.46	9.91	79.63	2.198	2.318	0.12
5	20.82	9.86	69.32	2.25	2.311	0.061
6	31.07	9.81	59.12	2.27	2.315	0.045
7	41.22	9.76	49.02	2.318	2.363	0.045
8	34.49	32.69	32.82	2.345	2.503	0.158
9	41.28	48.9	9.82	2.432	2.58	0.148
10	31.13	59	9.87	2.424	2.477	0.053
11	20.87	69.2	9.93	2.239	2.395	0.156
12	10.49	79.53	9.98	2.202	2.369	0.167
13	10.49	69.55	19.96	2.189	2.381	0.192

14	10.48	59.6	29.92	2.192	2.372	0.18
15	10.48	49.64	39.88	2.238	2.386	0.148
16	51.32	38.91	9.77	2.496	2.632	0.136
17	61.26	29.03	9.71	2.444	2.64	0.196
18	71.09	19.25	9.66	2.373	2.61	0.237
19	80.82	9.57	9.61	2.335	2.505	0.17
20	71.06	9.62	19.32	2.381	2.558	0.177
21	61.21	9.67	29.12	2.439	2.544	0.105
22	51.26	9.72	39.02	2.293	2.447	0.154
23	71.08	14.43	14.49	2.548	2.59	0.042
24	61.24	19.34	19.42	2.609	2.622	0.013
25	51.3	24.3	24.4	2.417	2.594	0.177
26	41.25	29.32	29.43	2.328	2.543	0.215
27	31.1	34.38	34.52	2.344	2.479	0.135
28	20.84	39.5	39.66	2.263	2.413	0.15
29	41.27	39.1	19.63	2.364	2.614	0.25
30	36.19	34.29	29.52	2.327	2.534	0.207
31	31.1	29.46	39.44	2.358	2.44	0.082
32	25.97	24.61	49.42	2.286	2.375	0.089
33	20.82	19.73	59.45	2.321	2.339	0.018
34	15.65	14.83	69.52	2.204	2.325	0.121
35	41.23	19.54	39.23	2.316	2.447	0.131
36	36.18	29.39	34.43	2.312	2.491	0.179
37	31.1	39.3	29.6	2.303	2.514	0.211
38	25.99	49.27	24.74	2.302	2.476	0.174
39	20.86	59.29	19.85	2.242	2.409	0.167
40	15.69	69.38	14.93	2.252	2.382	0.13
				Mean Absolute Error (MAE)		0.142

The NNR_4layers_Wen model was able to predict the $log_{10}H_V$ of 39 compositions within ±10% deviation range compared to experimental values.

Table S17. Prediction, absolute error, and MAE of $log_{10}Cost$ by NNR_4layers_Wen model for 40 experimental compositions

S. No.	Fe (at.%)	Co (at.%)	Ni (at.%)	Experimental $log_{10}Cost$ (log_{10}($/kg))	Predicted $log_{10}Cost$ (log_{10}($/kg))	Absolute error
1	10.47	39.7	49.83	1.303	1.286	0.017
2	10.47	29.76	59.77	1.261	1.24	0.021
3	10.47	19.83	69.7	1.213	1.195	0.018
4	10.46	9.91	79.63	1.16	1.159	0.001
5	20.82	9.86	69.32	1.117	1.125	0.008
6	31.07	9.81	59.12	1.07	1.06	0.01
7	41.22	9.76	49.02	1.017	1.017	0
8	34.49	32.69	32.82	1.196	1.186	0.01
9	41.28	48.9	9.82	1.254	1.242	0.012
10	31.13	59	9.87	1.326	1.337	0.011
11	20.87	69.2	9.93	1.388	1.386	0.002
12	10.49	79.53	9.98	1.442	1.432	0.01
13	10.49	69.55	19.96	1.411	1.417	0.006
14	10.48	59.6	29.92	1.378	1.386	0.008
15	10.48	49.64	39.88	1.342	1.347	0.005
16	51.32	38.91	9.77	1.168	1.161	0.007
17	61.26	29.03	9.71	1.06	1.053	0.007
18	71.09	19.25	9.66	0.916	0.909	0.007
19	80.82	9.57	9.61	0.7	0.793	0.093
20	71.06	9.62	19.32	0.803	0.834	0.031
21	61.21	9.67	29.12	0.887	0.896	0.009
22	51.26	9.72	39.02	0.957	0.966	0.009
23	71.08	14.43	14.49	0.863	0.883	0.02
24	61.24	19.34	19.42	0.982	0.996	0.014
25	51.3	24.3	24.4	1.075	1.066	0.009
26	41.25	29.32	29.43	1.152	1.144	0.008
27	31.1	34.38	34.52	1.217	1.206	0.011
28	20.84	39.5	39.66	1.273	1.29	0.017

29	41.27	39.1	19.63	1.206	1.208	0.002
30	36.19	34.29	29.52	1.199	1.194	0.005
31	31.1	29.46	39.44	1.191	1.178	0.013
32	25.97	24.61	49.42	1.184	1.176	0.008
33	20.82	19.73	59.45	1.176	1.178	0.002
34	15.65	14.83	69.52	1.168	1.177	0.009
35	41.23	19.54	39.23	1.09	1.093	0.003
36	36.18	29.39	34.43	1.172	1.167	0.005
37	31.1	39.3	29.6	1.241	1.241	0
38	25.99	49.27	24.74	1.301	1.308	0.007
39	20.86	59.29	19.85	1.353	1.352	0.001
40	15.69	69.38	14.93	1.4	1.407	0.007
				Mean Absolute Error (MAE)		0.011

The NNR_4layers_Wen model was able to predict the $log_{10}Cost$ of 39 compositions within ±10% deviation range compared to experimental values.

S5. Inverse Design Strategy

S5.1. Acquisition Function

As mentioned in **Section 2.4**, negative expected improvement was used as acquisition function in *gp_minimize*. Expected Improvement (*EI*) is defined as [4,5]:

$$EI(x) = \mathbb{E}\max\left(f(x) - f(x^+), 0\right) \tag{1}$$

where $f(x^+)$ is the best function value till that point and x^+ is the location of that function value which is described as:

$$x^+ = \underset{x_i \in x_{1:n}}{argmax} f(x_i) \tag{2}$$

The EI can be evaluated analytically by using a GP model through the following equation [6,7]:

$$EI(x) = \begin{cases} \left(\mu(x) - f(x^+)\right)\Phi(\mathrm{Z}) + \sigma(x)\phi(Z) & if\ \sigma(x) > 0 \\ 0 & if\ \sigma(x) = 0 \end{cases} \tag{3}$$

$$Z = \frac{\mu(x) - f(x^+)}{\sigma(x)}$$

where $\mu(x)$ and $\sigma(x)$ are the mean and standard deviation of GP posterior distribution at x, respectively; Φ and ϕ denote the cumulative density function and probability density function, respectively. *EI* finds the global maximum of the objective function; thus, the negative expected improvement finds the global minimum.

S5.2. Evolution of Acquisition function value for ETR model

The evolution of acquisition function value for the ETR model for each target sets of BO is shown in the figure below.

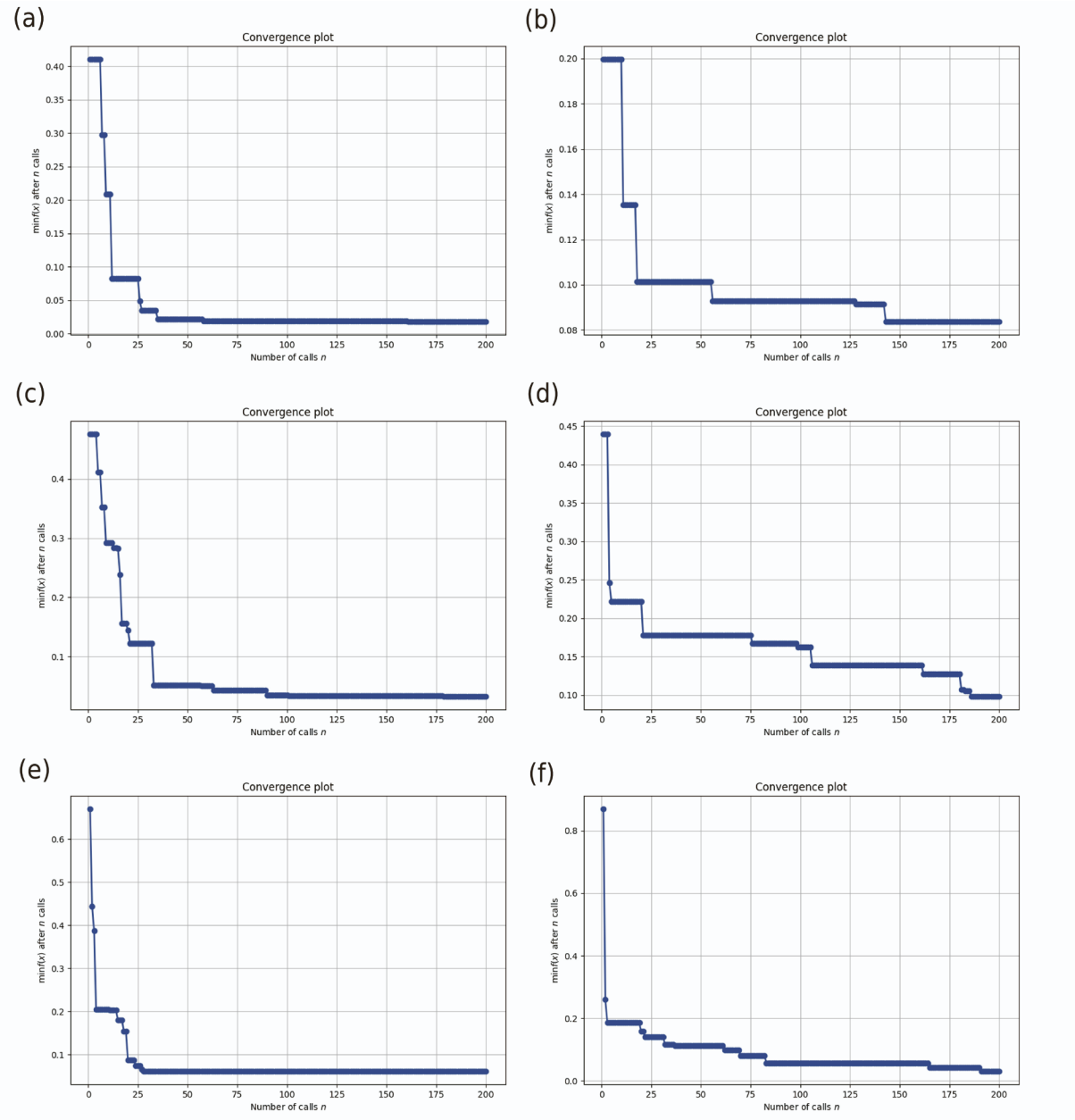

Figure S6. Evolution of acquisition function value of BO using the ETR model in the objective function for (a) target set 1, (b) target set 2, (c) target set 3, (d) target set 4, (e) target set 5, and (f) target set 6 of properties value.

S5.3. Predictions and Experimental Validation

The set target values for BO and the respective predicted composition with lowest objective value of BO by using ETR and NNR_4layers_Wen models are tabulated in **Table S18** below.

Table S18. Set targets for BO and the respective predicted composition for the MPR model

BO Set No.	Target M_s (T)	Target T_c (K)	Target *Cost* ($/kg)	MPR Model	Predicted Alloy Composition
1	1.5	970	15	ETR	$Fe_{27.3}Co_{25}Ni_{47.7}$
				NNR_4layers_Wen	$Fe_{26.9}Co_{26.3}Ni_{46.8}$
2	1.75	1000	11.5	ETR	$Fe_{44.7}Co_{15.4}Ni_{39.9}$
				NNR_4layers_Wen	$Fe_{41.7}Co_{15.9}Ni_{42.4}$
3	1.75	1143	19	ETR	$Fe_{23.5}Co_{40.8}Ni_{35.7}$
				NNR_4layers_Wen	$Fe_{26.9}Co_{47.8}Ni_{25.3}$
4	2	1070	10.5	ETR	$Fe_{50}Co_{20.8}Ni_{29.2}$
				NNR_4layers_Wen	$Fe_{46.4}Co_{25.2}Ni_{28.4}$
5	2.2	1173	10	ETR	$Fe_{61.9}Co_{22.8}Ni_{15.3}$
				NNR_4layers_Wen	$Fe_{61.5}Co_{18.9}Ni_{19.6}$
6	2.4	1173	10	ETR	$Fe_{66.8}Co_{28}Ni_{5.2}$
				NNR_4layers_Wen	$Fe_{65.1}Co_{32.4}Ni_{2.5}$

These compositions were used in their respective models to predict the properties and then compared with the approximate experimental values determined from the interactive contour plots. Using the predicted and experimental values, deviation percentage was calculated as per the following formula:

$$Deviation\ \% = \frac{Predicted - Actual}{Predicted} \times 100$$

The predicted values, approximate experimental values, and deviation percentages for M_s, T_c, *Cost*, H_c, ρ, and H_V for BO sets 1 – 5 using ETR model are tabulated in **Table S25**, **Table**

S26,**Table S27**, **Table S28**, **Table S29**, and **Table S30**, respectively. Similarly, the predicted values, approximate experimental values, and deviation percentages for M_s, T_c, *Cost*, H_c, ρ, and H_V for BO sets 1 – 5 using NNR_4layers_Wen model are tabulated in **Table S25**, **Table S26**, **Table S27**, **Table S28**, **Table S29**, and **Table S30**, respectively.

Table S19. Deviation of M_s for the predicted composition for the first 5 BO sets using ETR model

BO Set	Composition	Predicted M_s (T)	Approx. experimental M_s (T)	Deviation (%)
1	$Fe_{27.3}Co_{25}Ni_{47.7}$	1.5	1.47	2
2	$Fe_{44.7}Co_{15.4}Ni_{39.9}$	1.74	1.58	9.2
3	$Fe_{23.5}Co_{40.8}Ni_{35.7}$	1.75	1.6	8.6
4	$Fe_{50}Co_{20.8}Ni_{29.2}$	2	1.77	11.5
5	$Fe_{61.9}Co_{22.8}Ni_{15.3}$	2.22	2.01	9.5
Average Deviation (%)				8.16

Table S20. Deviation of T_c for the predicted composition for the first 5 BO sets using ETR model

BO Set	Composition	Predicted T_c (K)	Approx. experimental T_c (K)	Deviation (%)
1	$Fe_{27.3}Co_{25}Ni_{47.7}$	935.8	1002	-7.1
2	$Fe_{44.7}Co_{15.4}Ni_{39.9}$	846.3	890	-5.2
3	$Fe_{23.5}Co_{40.8}Ni_{35.7}$	1063.8	1103.7	-3.8
4	$Fe_{50}Co_{20.8}Ni_{29.2}$	872.5	1002	-14.8
5	$Fe_{61.9}Co_{22.8}Ni_{15.3}$	1047.5	1056.8	-0.9
Average Deviation (%)				-6.36

Table S21. Deviation of *Cost* for the predicted composition for the first 5 BO sets using ETR model

BO Set	Composition	Predicted *Cost* ($/kg)	Approx. calculated *Cost* ($/kg)	Deviation (%)
1	$Fe_{27.3}Co_{25}Ni_{47.7}$	15	15.28	-1.9
2	$Fe_{44.7}Co_{15.4}Ni_{39.9}$	11.6	11.1	4.3
3	$Fe_{23.5}Co_{40.8}Ni_{35.7}$	19	18.75	1.3
4	$Fe_{50}Co_{20.8}Ni_{29.2}$	10.7	11.48	-7.3

5	$Fe_{61.9}Co_{22.8}Ni_{15.3}$	10	10.2	-2
		Average Deviation (%)		-1.12

Table S22. Deviation of H_c for the predicted composition for the first 5 BO sets using ETR model

BO Set	Composition	Predicted H_c (A/m)	Approx. experimental H_c (A/m)	Deviation (%)
1	$Fe_{27.3}Co_{25}Ni_{47.7}$	3.2	385	-11931.3
2	$Fe_{44.7}Co_{15.4}Ni_{39.9}$	44.4	570	-1183.8
3	$Fe_{23.5}Co_{40.8}Ni_{35.7}$	362.1	290	19.9
4	$Fe_{50}Co_{20.8}Ni_{29.2}$	352.2	770	-118.6
5	$Fe_{61.9}Co_{22.8}Ni_{15.3}$	549.5	1350	-145.7
		Average Deviation (%)		-2671.9

Table S23. Deviation of ρ for the predicted composition for the first 5 BO sets using ETR model

BO Set	Composition	Predicted ρ (μΩ·cm)	Approx. experimental ρ (μΩ·cm)	Deviation (%)
1	$Fe_{27.3}Co_{25}Ni_{47.7}$	20.7	22.8	-10.1
2	$Fe_{44.7}Co_{15.4}Ni_{39.9}$	23.3	40.3	-73
3	$Fe_{23.5}Co_{40.8}Ni_{35.7}$	18.7	31.5	-68.4
4	$Fe_{50}Co_{20.8}Ni_{29.2}$	20.9	30	-43.5
5	$Fe_{61.9}Co_{22.8}Ni_{15.3}$	18.7	24.5	-31
		Average Deviation (%)		-45.2

Table S24. Deviation of H_V for the predicted composition for the first 5 BO sets using ETR model

BO Set	Composition	Predicted H_V (HV)	Approx. experimental H_V (HV)	Deviation (%)
1	$Fe_{27.3}Co_{25}Ni_{47.7}$	194.5	198	-1.8
2	$Fe_{44.7}Co_{15.4}Ni_{39.9}$	294.4	210	28.7
3	$Fe_{23.5}Co_{40.8}Ni_{35.7}$	236.7	188	20.6
4	$Fe_{50}Co_{20.8}Ni_{29.2}$	349	248	28.9
5	$Fe_{61.9}Co_{22.8}Ni_{15.3}$	347.1	365	-5.2
		Average Deviation (%)		14.24

Table S25. Deviation of M_s for the predicted composition for the first 5 BO sets using NNR_4layers_Wen model

BO Set	Composition	Predicted M_s (T)	Approx. experimental M_s (T)	Deviation (%)
1	$Fe_{26.9}Co_{26.3}Ni_{46.8}$	1.5	1.47	2
2	$Fe_{41.7}Co_{15.9}Ni_{42.4}$	1.75	1.56	10.9
3	$Fe_{26.9}Co_{47.8}Ni_{25.3}$	1.75	1.73	1.1
4	$Fe_{46.4}Co_{25.2}Ni_{28.4}$	2.01	1.79	10.9
5	$Fe_{61.5}Co_{18.9}Ni_{19.6}$	2.18	1.94	11
Average Deviation (%)				7.18

Table S26. Deviation of T_c for the predicted composition for the first 5 BO sets using NNR_4layers_Wen model

BO Set	Composition	Predicted T_c (K)	Approx. experimental T_c (K)	Deviation (%)
1	$Fe_{26.9}Co_{26.3}Ni_{46.8}$	943.4	1005	-6.5
2	$Fe_{41.7}Co_{15.9}Ni_{42.4}$	857	900	-5
3	$Fe_{26.9}Co_{47.8}Ni_{25.3}$	1082.4	1155	-6.7
4	$Fe_{46.4}Co_{25.2}Ni_{28.4}$	945.3	1029	-8.9
5	$Fe_{61.5}Co_{18.9}Ni_{19.6}$	942.4	1000	-6.1
Average Deviation (%)				-6.64

Table S27. Deviation of *Cost* for the predicted composition for the first 5 BO sets using NNR_4layers_Wen model

BO Set	Composition	Predicted *Cost* ($/kg)	Approx. calculated *Cost* ($/kg)	Deviation (%)
1	$Fe_{26.9}Co_{26.3}Ni_{46.8}$	15.2	15.5	-2
2	$Fe_{41.7}Co_{15.9}Ni_{42.4}$	11.5	11.6	-0.9
3	$Fe_{26.9}Co_{47.8}Ni_{25.3}$	20	19.9	0.5
4	$Fe_{46.4}Co_{25.2}Ni_{28.4}$	12.5	12.7	-1.6
5	$Fe_{61.5}Co_{18.9}Ni_{19.6}$	9.8	9.59	2.1
Average Deviation (%)				-0.38

Table S28. Deviation of H_c for the predicted composition for the first 5 BO sets using NNR_4layers_Wen model

BO Set	Composition	Predicted H_c (A/m)	Approx. experimental H_c (A/m)	Deviation (%)
1	$Fe_{26.9}Co_{26.3}Ni_{46.8}$	117.9	385	-226.5
2	$Fe_{41.7}Co_{15.9}Ni_{42.4}$	58.2	485.29	-733.8
3	$Fe_{26.9}Co_{47.8}Ni_{25.3}$	903.3	454	49.7
4	$Fe_{46.4}Co_{25.2}Ni_{28.4}$	378.3	529	-39.8
5	$Fe_{61.5}Co_{18.9}Ni_{19.6}$	506.1	1520.6	-200.5
Average Deviation (%)				-230.18

Table S29. Deviation of ρ for the predicted composition for the first 5 BO sets using NNR_4layers_Wen model

BO Set	Composition	Predicted ρ (μΩ·cm)	Approx. experimental ρ (μΩ·cm)	Deviation (%)
1	$Fe_{26.9}Co_{26.3}Ni_{46.8}$	17.3	22.8	-31.8
2	$Fe_{41.7}Co_{15.9}Ni_{42.4}$	21.5	37.8	-75.8
3	$Fe_{26.9}Co_{47.8}Ni_{25.3}$	16.8	23	-36.9
4	$Fe_{46.4}Co_{25.2}Ni_{28.4}$	18.5	26.5	-43.2
5	$Fe_{61.5}Co_{18.9}Ni_{19.6}$	18.9	27.2	-43.9
Average Deviation (%)				-46.32

Table S30. Deviation of H_V for the predicted composition for the first 5 BO sets using NNR_4layers_Wen model

BO Set	Composition	Predicted H_V (HV)	Approx. experimental H_V (HV)	Deviation (%)
1	$Fe_{26.9}Co_{26.3}Ni_{46.8}$	244.1	200.9	17.7
2	$Fe_{41.7}Co_{15.9}Ni_{42.4}$	260.4	207	20.5
3	$Fe_{26.9}Co_{47.8}Ni_{25.3}$	305.9	200	34.6
4	$Fe_{46.4}Co_{25.2}Ni_{28.4}$	360.4	220	39
5	$Fe_{61.5}Co_{18.9}Ni_{19.6}$	416.8	406.4	2.5
Average Deviation (%)				22.86

References

1 Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A. & Talwalkar, A. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research* **18**, 6765-6816 (2017).

2 O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H. & Invernizzi, L. KerasTuner. (2019).

3 Head, T., Kumar, M., Nahrstaedt, H., Louppe, G. & Shcherbatyi, I. scikit-optimize. *Zenodo* (2021). https://doi.org/10.5281/zenodo.5565057

4 Brochu, E., Cora, V. M. & De Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599* (2010).

5 Frazier, P. I. & Wang, J. in *Information Science for Materials Discovery and Design Springer Series in Materials Science* (eds Turab Lookman, Francis J. Alexander, & Krishna Rajan) Ch. Chapter 3, 45-75 (Springer International Publishing, 2016).

6 Močkus, J. in *Optimization Techniques IFIP Technical Conference* (ed G. I. Marchuk) Ch. Chapter 55, 400-404 (Springer Berlin Heidelberg, 1975).

7 Jones, D. R., Schonlau, M. & Welch, W. J. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization* **13**, 455-492 (1998). https://doi.org/10.1023/a:1008306431147