

## Electronic Supplementary Information for Machine Learning Analysis to Classify Nanoparticles from Noisy spICP- TOFMS Data

Raven L. Buckman, Alexander Gundlach-Graham

Department of Chemistry, Iowa State University, Ames, Iowa

Address correspondence: alexgg@iastate.edu

### Table of Contents

<i>Instrument/Analysis Conditions</i> .....	S2
Table S1: Preparation of Mixture Samples – Neat Suspensions.....	S2
Table S2: Preparation of Mixture Samples – Mixtures .....	S2
Table S3: Instrument Parameters .....	S2
Table S4: Isotopes for Quantification with Microdroplets .....	S3
<i>Supervised Machine Learning</i> .....	S4
Table S5: Figures of Merit and Machine Learning Parameters for Supervised Learning .....	S4
<i>Semi-Supervised Machine Learning</i> .....	S6
Figure S1: Data Processing Workflow .....	S6
Table S6: Semi-Supervised Machine Learning Parameters.....	S7
<i>Time Traces from spICP-TOFMS</i> .....	S8
Figure S2: Single-Particle Time Traces .....	S8
<i>Unsupervised Machine Learning</i> .....	S9
Figure S3: tSNE Unsupervised Learning.....	S9
<i>Confusion Matrix Normalization</i> .....	S10
Figure S4: Non-Normalized Confusion Matrix .....	S10
<i>Precision-Recall Curves</i> .....	S11
Figure S5: Precision-Recall Curves for SSML Models .....	S11
<i>Mixture Analysis</i> .....	S12
Figure S6: Classification of Dilutions Samples – All Classes.....	S12
Table S7: ANOVA Test Results for Figure S5.....	S13
<i>Particle Type Specific Detection Limit Filtering</i> .....	S14
Figure S7: Classification of Dilution Samples – Detection Limit Filtering .....	S15
Table S8: ANOVA Test Results for Figure S6.....	S16

## Instrument/Analysis Conditions

Table S1: Preparation of Mixture Samples - Standards

Particle Sample Type	Stock PNC (particles mL <sup>-1</sup> )
Cerium (IV) Oxide (ENP)	3.03E+05
Ferrocerium Mischmetal (INP)	3.38E+05
Bastnaesite/Parisite Mineral (NNP)	7.74E+05

Table S2: Preparation of Mixture Samples – Mixtures

Sample	PNC (particles mL <sup>-1</sup> )	Dilution Factor - ENP	Dilution Factor - INP	Dilution Factor - NNP
E1	2.85E+05	1.43	14.5	31.1
E2	1.67E+05	2.15		
E3	1.21E+05	4.30		
E4	8.96E+04	10.8		
E5	7.91E+04	21.5		
E6	7.22E+04	108		
I1	2.48E+05	10.8	1.45	
I2	1.59E+05		2.90	
I3	1.27E+05		5.80	
I4	9.74E+04		14.5	
I5	7.54E+04		29.0	
I6	5.85E+04		145	

Table S3: Instrument Parameters

Spray Chamber	Baffled cyclonic quartz
Nebulizer	PFA Prep Fast
Sample Uptake (μL min <sup>-1</sup> )	45
Nebulizer flow (L min <sup>-1</sup> )	0.81
Auxiliary gas flow (L min <sup>-1</sup> )	1.14
Cool gas flow (L min <sup>-1</sup> )	13.3
RF Power (W)	1515
Add. gas flow Ar/He (L min <sup>-1</sup> )*	0.013/0.34
Transport Efficiency (%)	4
Torch/Injector	iCAP Q Quartz torch with 1.5 mm quartz injector
Sampling Depth (mm)	4.98
Sampler/Skimmer Cone	iCAP Q Nickel Sampler (X Series) with Skimmer (with insert)
Notch filter masses (Th)	17.2, 30.0, 36.5, 40.5
He collision cell flow (mL min <sup>-1</sup> )	5.6
Averaged TOF spectrum time resolution	1.2
Number TOF spectra averaged per time point	100

\*Additional gases are for the desolvation of droplets with the online microdroplet calibration system.

Table S4: Isotopes used for quantification, element concentrations in droplets for online microdroplet calibration, and absolute sensitivities (TofCts g<sup>-1</sup>) from droplet-derived signals. The multi-element calibration solution for online microdroplet calibration was prepared using single-element standards (High-Purity Standards, SC, USA). All dilutions were prepared gravimetrically (ML204T/A00, Mettler-Toledo, Greifensee, Switzerland) using 2% sub-boiled, trace-metal grade nitric acid (Fisher Scientific, Fair Lawn, NJ, USA) as the diluent.

<b>Isotopes used for quantification</b>	<b>Element concentrations in droplets (ng mL<sup>-1</sup>)</b>	<b>Absolute sensitivities (TofCts g<sup>-1</sup>)</b>
<sup>57</sup> Fe	62.5	1.62E+15
<sup>139</sup> La	32.8	2.59E+17
<sup>140</sup> Ce	30.5	2.78E+17
<sup>141</sup> Pr	31.9	3.20E+17
<sup>144</sup> Nd, <sup>146</sup> Nd	31.8	1.42E+17
<sup>232</sup> Th	32.5	3.65E+17
<sup>133</sup> Cs*	31.3	2.22E+17

\*Cs was used for the determination of solution uptake and was not quantified in NPs.

## Supervised Machine Learning

Thirty different supervised machine learning models were tested as a comparison against the SSML model. We report these methods and figures of merit to demonstrate that the supervised learning models did not perform at the same level as the first SSML model, i.e. all supervised methods performed with an AUC value below that of the first SSML model (~0.96). These models were tested using the *Classification Learner*<sup>TM</sup> in MATLAB and by direct coding with hyperparameter optimization. Results are summarized in Table S5.

Table S5: Figures of merit for supervised machine learning models including the learner type, accuracy (ACC), weighted-average AUC values from the ROC curve and additional model parameters.

Learner Type	ACC (%)	AUC (Avg)	Additional Parameters
Tree: Fine	78.5	0.91	MaxNumSplits: 100, SplitCriterion: Gini's Diversity Index
Tree: Medium	75.2	0.89	MaxNumSplits: 20, SplitCriterion: Gini's Diversity Index
Tree: Coarse	70.2	0.85	MaxNumSplits: 4, SplitCriterion: Gini's Diversity Index
Linear Discriminant	55.9	0.82	CovarianceStructure: Full
Naive Bayes	62.4	0.75	Distribution: Gaussian
Naive Bayes	69.0	0.83	Distribution: Kernel, KernelType: Gaussian, Support: Unbounded
SVM	64.2	0.84	KernelFunction: Linear, KernelScale: Automatic, BoxConstraint: 1, MulticlassMethod: OneVOne, StandardizeData: True
SVM	36.2	0.53	KernelFunction: Quadratic, KernelScale: Automatic, BoxConstraint: 1, MulticlassMethod: OneVOne, StandardizeData: True
SVM	38.1	0.51	KernelFunction: Cubic, KernelScale: Automatic, BoxConstraint: 1, MulticlassMethod: OneVOne, StandardizeData: True
SVM	75.5	0.90	KernelFunction: Gaussian, KernelScale: 0.61, BoxConstraint: 1, MulticlassMethod: OneVOne, StandardizeData: True
SVM	69.9	0.87	KernelFunction: Gaussian, KernelScale: 2.4, BoxConstraint: 1, MulticlassMethod: OneVOne, StandardizeData: True
SVM	60.7	0.84	KernelFunction: Gaussian, KernelScale: 9.8, BoxConstraint: 1, MulticlassMethod: OneVOne, StandardizeData: True
KNN	78	0.82	NumNeighbors: 1, Distance: Euclidean, DistanceWeight: Equal, StandardizeData: True
KNN	79.2	0.92	NumNeighbors: 10, Distance: Euclidean, DistanceWeight: Equal, StandardizeData: True
KNN	78.7	0.92	NumNeighbors: 100, Distance: Euclidean, DistanceWeight: Equal, StandardizeData: True
KNN	71.1	0.86	NumNeighbors: 10, Distance: Cosine, DistanceWeight: Equal, StandardizeData: True
KNN	79.2	0.92	NumNeighbors: 10, Distance: Cubic, DistanceWeight: Equal, StandardizeData: True
KNN	78.5	0.90	NumNeighbors: 10, Distance: Euclidean, DistanceWeight: Squared inverse, StandardizeData: True
Ensemble	79.2	0.92	EnsemblePreset: Boosted Trees, EnsembleMethod: AdaBoost, LearnerType: Decision Tree, MaxNumSplit: 20, NumLearn: 30, LearnRate: 0.1, Predictors: ALL

Ensemble	78.5	0.90	EnsemblePreset: Bagged Trees, EnsembleMethod: Bag LearnerType: Decision Tree, MaxNumSplit: 5610, NumLearn: 30, Predictors: ALL
Ensemble	55.5	0.82	EnsemblePreset: Discriminant, EnsembleMethod: Subspace, LearnerType: Discriminant, NumLearn: 30, SubspaceDimension: 3, Predictors: ALL
Ensemble	54.5	0.72	EnsemblePreset: KNN, EnsembleMethod: Subspace, LearnerType: Nearest Neighbors, NumLearn: 30, SubspaceDimension: 3, Predictors: ALL
Ensemble	78.0	0.92	Ensemble Preset: RUSBoosted Trees, Ensemble Method: RUSBoost Type: Decision Tree, MaxNumSplit: 20, NumLearn: 30, LearningRate: 0.1, Predictors: ALL
Neural Network	77.7	0.92	Preset: Narrow Neural Network, NumConnect: 1, FirstLayerSz: 10, Activation: ReLU, IterationLim: 1000, Lambda: 0, Stdz: True
Neural Network	80.3	0.93	Preset: Medium Neural Network, NumConnect: 1, FirstLayerSz: 25, Activation: ReLU, IterationLim: 1000, Lambda: 0, Stdize: True
Neural Network	80.3	0.93	Preset: Wide Neural Network, NumConnect: 1, FirstLayerSz: 100, Activation: ReLU, IterationLim: 1000, Lambda: 0, Stdize: True
Neural Network	79.9	0.93	Preset: Bilayer Neural Network, NumConnect: 2, FirstLayerSz: 10, Second Layer Size: 10, Activation: ReLU, IterationLim: 1000, Lambda: 0, Stdize: True
Neural Network	80.8	0.93	Preset: Trilayer Neural Network, NumConnect: 3, FirstLayerSz: 10, Second Layer Size: 10, Third Layer Size: 10, Activation: ReLU, IterationLim: 1000, Lambda: 0, Stdize: True
Kernel	58.1	0.57	Preset: SVM Kernel, Learner: SVM, NumExpDim: Auto, Lambda: Auto, Kernel Scale: Auto, MultiClassMethod: OneVOne, IterationLim: 1000
Kernel	57.1	0.59	Preset: Logistic Regression Kernel, Learner: Logistic Regression, NumExpDim: Auto, Lambda: Auto, Kernel Scale: Auto, MultiClassMethod: OneVOne, IterationLim: 1000
Neural Network	80.3	0.93	Preset: Medium Neural Network, NumConnect: 1, First Layer Size: 25, Activation: ReLU, IterationLim: 1000, Lambda: 0, Stdize: True

# Semi-Supervised Machine Learning

Figure S1: Flow chart describing data processing, model optimization, and sample analysis.

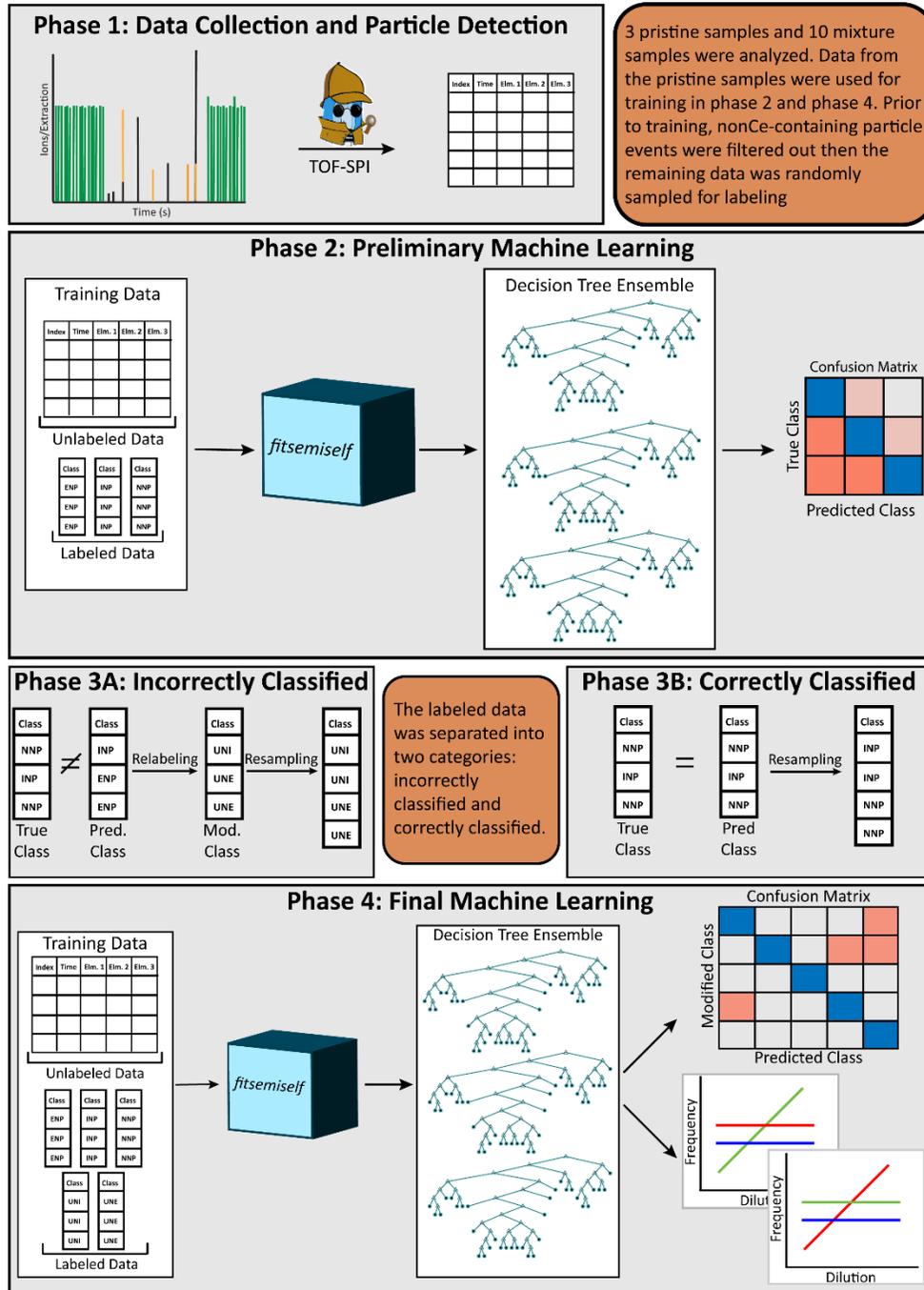
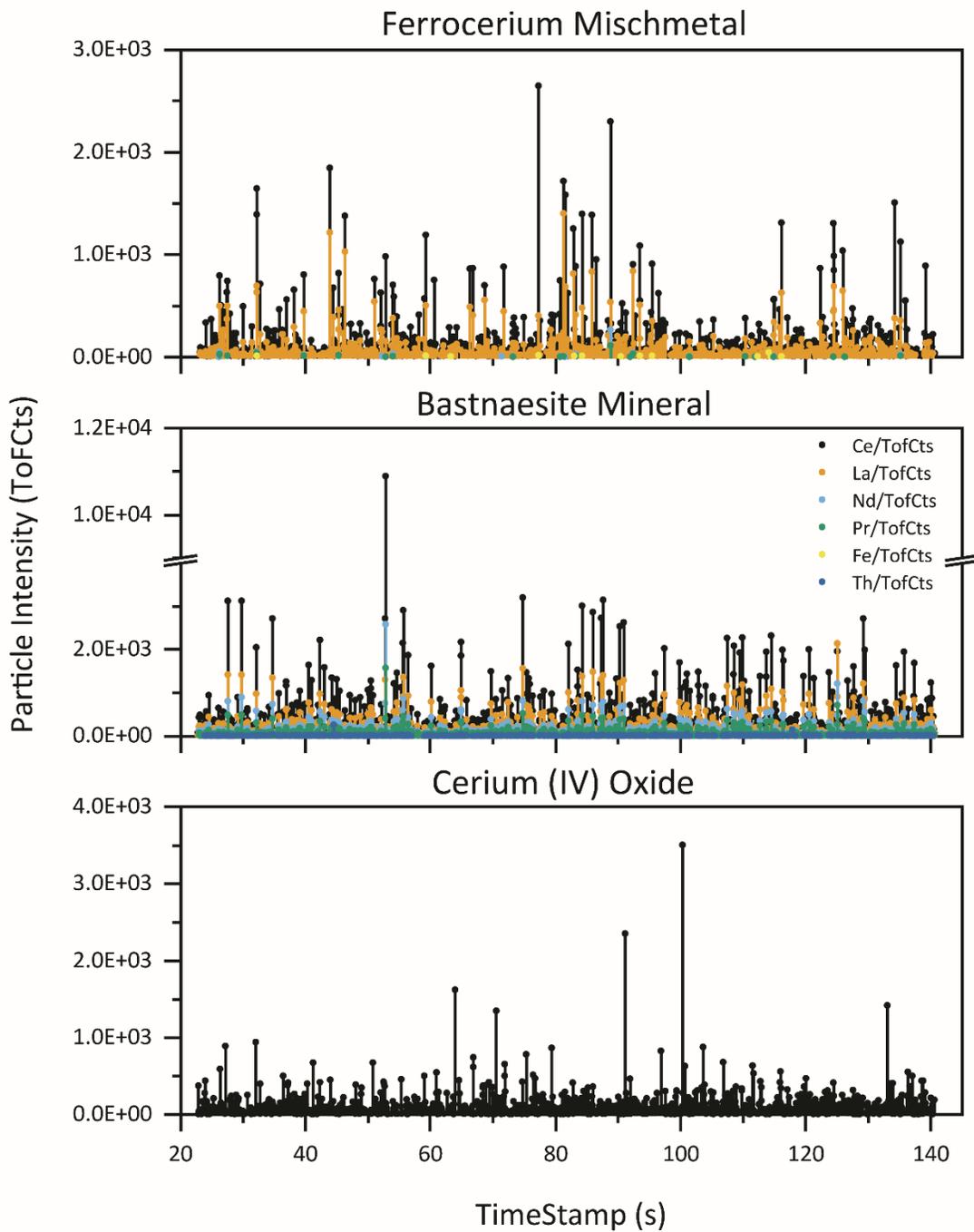


Table S6: Parameters used for semi-supervised machine learning models. Differences between the first and second model are indicated by the use of a semi-colon. Italicized fonts are indicative of the functions used in MATLAB.

Parameter	Value
<i>fitsemiself</i>	
Learner	<i>templateEnsemble</i>
IterationLimit	1.00E+03
ScoreThreshold	-0.1
CategoricalPredictors	'all'
ClassNames	{'ENP','INP','NNP'}; {'ENP','INP','NNP','UNE','UNI'}
PredictorNames	{'Fe (g)', 'La (g)', 'Ce (g)', 'Th (g)', 'Nd (g)', 'Pr (g)'}
ObservationsIn	'rows'
<i>templateEnsemble</i>	
Method	'bag'
Nlearn	500
Learner	<i>templateTree</i>
Nprint	'off'
Type	'classification'
FResample	1
Replace	'on'
Resample	'on'
LearnRate	1
<i>templateTree</i>	
MaxNumSplits	'n-1'
MergeLeaves	'off'
MinLeafSize	1
MinParentSize	2
NumVariablesToSample	'all'
PredictorSelection	'allsplits'
Prune	'off'
PruneCriterion	'error'
Reproducible	TRUE
SplitCriterion	'gdi'
Surrogate	'off'
Type	'classification'
AlgorithmForCategorical	'exact'
MaxNumCategories	3; 5

### Time Traces from spICP-TOFMS

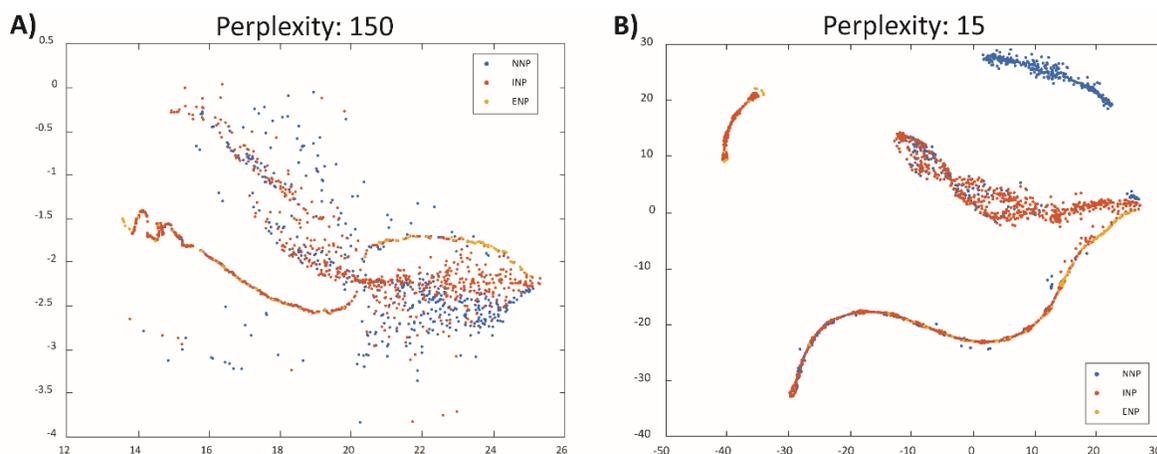
Figure S2: spICP-TOFMS time traces of pristine nanoparticles suspensions. Spikes represent a measured nanoparticle. Spikes with multiple colors are representative of multi-elemental nanoparticle signals.



### *Unsupervised Machine Learning*

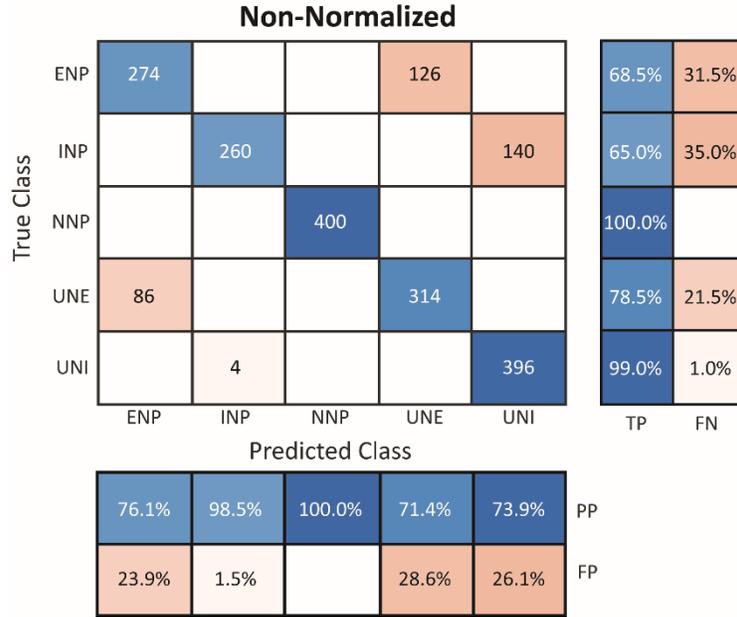
An unsupervised machine learning model, t-stochastic neighbor embedding (tSNE), was tested using the neat suspension data from ENPs, INPs and NNPs. For this model, we used the Euclidean distance function and manipulated the perplexity argument in an effort to extract distinct clusters for ENPs, INPs and NNPs. In Figure S3A and S3B , we show results using perplexity values of 150 and 15, respectively. In either case (and in all those we tested), the tSNE unsupervised learning model does not show usable clusters for the three particle types; therefore, tSNE was deemed to be an ineffective model type for classification of Ce-NPs.

Figure S3: An example of t-stochastic neighbor embedding performed with the pristine sample data. Distance function was set to Euclidean and the perplexity was set to 150 (A) and 15 (B).



### Confusion Matrix Normalization

Figure S4: The non-normalized confusion matrix representing the performance of the second SSML model with resampling.



It is important to note that this confusion matrix is misleading and not representative of the true performance of the SSML model because the particle events are resampled; therefore, the matrix must be normalized in order to compare the performance of the first and second SSML models. To normalize the matrix, we multiplied each matrix component by the actual number of particles in each class and divided by the number of samples used (Eq. S1).

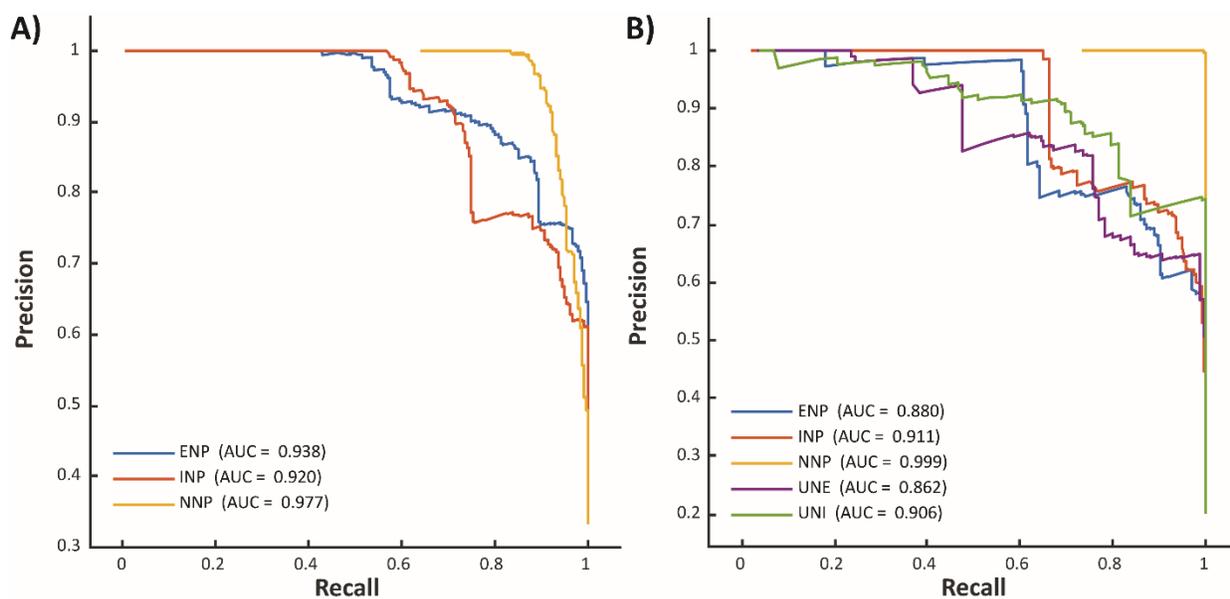
$$N_w = \frac{M_{i,j} * N}{k} \quad \text{Eq. S1}$$

Where  $N_w$  is the weighted number of particles,  $M_{i,j}$  is matrix value,  $N$  is the true number of events without resampling and  $k$  is the total number events sampled from each class (i.e. 400). This normalization preserves the percentages of true-positives (TP) and false-negatives (FN) but adjusts percentages of positive-predictions (PP) and false-predictions (FP) to be representative of the true model performance.

### Precision-Recall Curves

In machine learning metrics, precision is used to measure how many of positive predictions are truly positive. Recall (a.k.a. sensitivity, true positive rate) is the measure of the number of true positives that are correctly classified as positive. Together, precision and recall can be used to assess model performance with respect to a minor class in an imbalanced model; a precision-recall (PR) curve is often used as visualization of this assessment. As with ROC curves, we may use the area under the PR curve for a quantitative comparison of model performance (the closer the AUC value is to 1, the better the model performance).<sup>1, 2</sup> As seen in Figure S5, the PR curve improves for classification NNPs and remains similar for the ENP and INP classification from the first to second SSML models.

Figure S5: Precision-recall curves for the first (A) and second (B) SSML models with AUC values shown for each of the particle classes used in the model. The weighted-averages were 0.945 and 0.912 for A and B, respectively.



*Mixture Analysis*

Figure S6: The number of particles that were classified by the second machine learning model vs. the number of calculated, or theoretically desired, engineered (A) and incidental (B) particles. Error bars show the variability of the model in the number of particles in each class depending on which particle events were sampled for the training sets. Figure includes the unclassifiable particle classes and the number of particle events that fell below 49 ag.

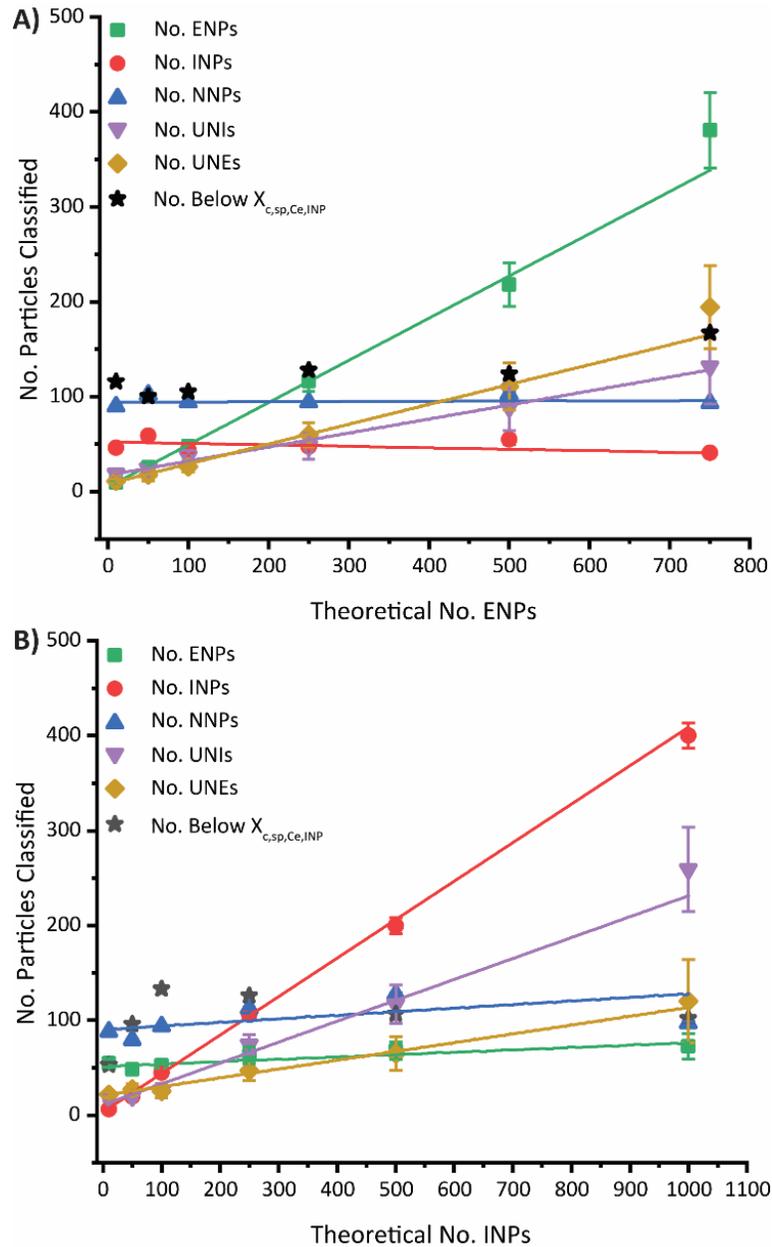


Table S7: ANOVA test results for nanoparticle classification, by semi-supervised machine learning; results were calculated with an alpha of 0.05.

**A) Increasing ENPs PNC with constant INP and NNP background**

		DF	Sum of Squares	Mean Square	F Value	Prob > F
No. ENPs	Model	1	325.81	325.81	776.21	9.87E-6
	Error	4	1.68	0.42		
	Total	5	327.49			
No. INPs	Model	1	30.96	30.96	3.39	0.14
	Error	4	36.55	9.14		
	Total	5	67.50			
No. NNPs	Model	1	1.30	1.30	0.07	0.81
	Error	4	78.86	19.72		
	Total	5	80.17			
No. UNIs	Model	1	22.28	22.28	101.78	5.43E-4
	Error	4	0.88	0.22		
	Total	5	23.16			
No. UNEs	Model	1	49.14	49.14	247.10	9.57E-5
	Error	4	0.80	0.20		
	Total	5	49.94			

**B) Increasing INPs PNC with constant ENP and NNP background**

		DF	Sum of Squares	Mean Square	F Value	Prob > F
No. ENPs	Model	1	4.41	4.41	8.57	0.043
	Error	4	2.06	0.51		
	Total	5	6.47			
No. INPs	Model	1	1859.57	1859.57	1411.27	3.00E-6
	Error	4	5.27	1.32		
	Total	5	1864.84			
No. NNPs	Model	1	511.62	511.62	2.67	0.18
	Error	4	767.35	191.84		
	Total	5	1278.98			
No. UNIs	Model	1	73.72	73.72	52.83	1.90E-3
	Error	4	5.58	1.40		
	Total	5	79.30			
No. UNEs	Model	1	15.28	15.28	87.15	7.33E-4
	Error	4	0.70	0.18		
	Total	5	15.99	4.41		

### Particle Type Specific Detection Limit Filtering

A particle-type detection limit is the minimum signal of a major element ( $j$ ) required to produce a measurable signal (at a given confidence level) of a minor element ( $k$ ) from a particle type that contains elements  $j$  and  $k$  at a given ratio ( $R_{j:k}$ ).<sup>3</sup> The particle-type detection limit is calculated based on Poisson-Normal statistics and also depends on the critical value of the minor element (i.e.  $L_{C,sp,k}$ ) in a given particle type. A generic particle-type detection limit ( $L_{D,sp,j,j-k}$ ) expression is given in Equation S2, where  $z_{1-\beta}$  is the one-sided z-score for a false-negative  $\beta$  value. This beta value is usually set to 5% to achieve a 95% confidence limit for the detection limit.

$$L_{D,sp,j,j-k} = \left( \frac{z_{1-\beta} + \sqrt{z_{1-\beta}^2 + 4L_{C,sp,k}}}{2} \right)^2 R_{j:k} \quad \text{Eq. S2}$$

In our analysis of Ce-containing NNPs (i.e. from a bastnaesite mineral sample), both Ce:La and Ce:Nd showed good correlation. The signal (ToFCounts) ratio of Ce:La was 2.11 and the mass ratio was 1.99. For the Ce:Nd ratio, the ToFCounts ratio was 3.89 and the mass ratio was 2.02. Because particle-type detection limits are calculated based on Poisson statistics, they are calculated in the signal domain. Example calculations for  $L_{D,sp,Ce,Ce-La}$  and  $L_{D,sp,Ce,Ce-Nd}$  with a beta value of 5% and experimental  $L_{C,sp,k}$  and  $R_{j:k}$  values from the NNP neat suspension are given below in Eq. S3 and S4, respectively.

$$L_{D,sp,Ce,Ce-La} = \left( \frac{1.64 + \sqrt{2.69 + 4(7.64)}}{2} \right)^2 * 2.11 = 28.9 \text{ ToFCounts} \quad \text{Eq. S3}$$

$$L_{D,sp,Ce,Ce-Nd} = \left( \frac{1.64 + \sqrt{2.69 + 4(6.92)}}{2} \right)^2 * 3.89 = 49.7 \text{ ToFCounts} \quad \text{Eq. S4}$$

The particle-type detection limits correspond to Ce masses of 100 ag and 172 ag, respectively, for the detection of Ce-La and Ce-Nd signatures in NNPs. These particle-type detection limits indicate the minimum signal from Ce required to measure La or Nd in the Ce-NNPs with 95% confidence. If Ce is measured as a Ce-only smNP with Ce signal greater than  $L_{D,sp,Ce,Ce-La}$ , then the particle is classified as an “ENP.” However, if a Ce-only smNP is recorded with Ce signal less than  $L_{D,sp,Ce,Ce-La}$ , then the particle is classified as “Ce-only undefined.” Ce-INPs are characterized as having both Ce and La. With particle-type detection limit filtering, if a dual-metal Ce-La particle is measured, and the Ce signal is greater than  $L_{D,sp,Ce,Ce-Nd}$ , then the particle is classified as an “INP.” However, if a Ce-La mmNP particle is recorded with Ce signal less than  $L_{D,sp,Ce,Ce-Nd}$ , then this particle is not large enough to be certain that Nd would have been measured if the particle were an NNP; therefore, we cannot confidently assign a class to the particle event and it is classified as “CeLa undefined.” Any Ce-containing particles with measurable Nd signals are classified as “NNPs.”

The results of the particle-type detection limit classification are summarized in Figure S7. The slopes were also statistically tested with an ANOVA test and results are given in Table S8. As with semi-supervised machine learning, the slope of the ENPs was significantly different from zero and

the detection limit classification performs as expected with a constant INP and NNP background. Additionally, the particle-type specific detection limits showed that the slope of the NNPs was the only significantly different from zero when increasing the INP's PNC. The ENPs trendline had an  $R^2$  of 0.667 and a p-value of 0.047, similar to the results of the semi-supervised machine learning model performance. However, the maximum number of incidental particles classified by the detection limit classification, 274, was small in comparison to the maximum number classified by machine learning, 400.

Figure S7: Scatter plot of particle event classification, by particle type specific detection limits, as a function of increasing ENP number (A) and INP number (B).

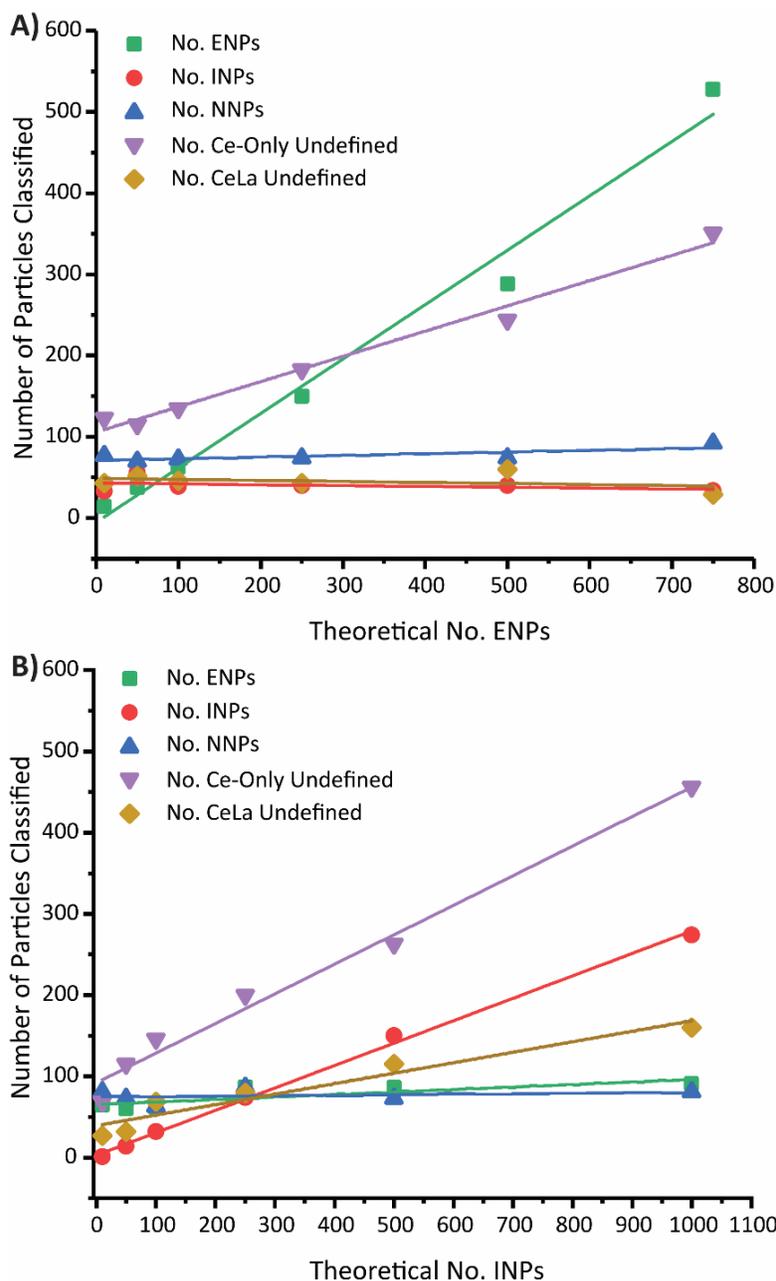


Table S8: ANOVA test results for nanoparticle classification, by particle-type specific detection limits; results were calculated with an alpha of 0.05.

**A) Increasing ENPs PNC with constant INP and NNP background**

		DF	Sum of Squares	Mean Square	F Value	Prob > F
No. ENPs	Model	1	191984.15	191984.15	248.31	9.48E-5
	Error	4	3092.69	773.17		
	Total	5	195076.83			
No. INPs	Model	1	45.68	45.68	0.62	0.48
	Error	4	295.65	73.91		
	Total	5	341.33			
No. NNPs	Model	1	180.93	180.93	5.73	0.075
	Error	4	126.40	31.60		
	Total	5	307.33			
No. UNIs	Model	1	41320.05	41320.05	241.36	1.00E-4
	Error	4	684.78	171.20		
	Total	5	42004.83			
No. UNEs	Model	1	68.08	68.08	0.58	0.49
	Error	4	469.42	117.36		
	Total	5	537.50			

**B) Increasing INPs PNC with constant ENP and NNP background**

		DF	Sum of Squares	Mean Square	F Value	Prob > F
No. ENPs	Model	1	676.90	676.90	8.01	0.047
	Error	4	337.93	84.48		
	Total	5	1014.83			
No. INPs	Model	1	54626.38	54626.38	1533.87	2.54E-6
	Error	4	142.45	35.61		
	Total	5	54768.83			
No. NNPs	Model	1	24.76	24.76	0.30	0.61
	Error	4	332.58	83.14		
	Total	5	357.33			
No. UNIs	Model	1	94954.87	94954.87	261.59	8.55E-5
	Error	4	1451.97	362.99		
	Total	5	96406.83			
No. UNEs	Model	1	11935.89	11935.89	51.80	1.97E-3
	Error	4	921.61	230.40		
	Total	5	12857.50			

1. Y. Ma and H. He, *Imbalanced learning : foundations, algorithms, and applications*, IEEE Press, Piscataway, NJ, 2013.
2. Z.-H. Zhou, *Ensemble methods : foundations and algorithms*, Taylor & Francis, Boca Raton, FL, 2012.
3. S. E. Szakas, R. Lancaster, R. Kaegi and A. Gundlach-Graham, *Environmental Science: Nano*, 2022, **9**, 1627-1638.