Supporting Information

Real-Time Detection and Classification of PFAS using Dynamic Behaviors at Liquid-Liquid Interfaces

Baishali Barua, Laura K. Dunham, Aakanksha Gadh, and Suchol Savagatrup*

Department of Chemical and Environmental Engineering, University of Arizona, 1133 E. James E. Rogers Way, Tucson, Arizona 85721

* Authors to whom correspondence should be addressed: S.S. (suchol@arizona.edu)

Keywords: sensors, microfluidics, Janus droplets, interfacial tension, PFAS

Table of Contents

1. Chemicals

Toluene, perfluorotributylamine (FC-43), perfluorooctane sulfonic acid (PFOS), and perfluroocatanoic acid (PFOA) were obtained from Sigma-Aldrich. Capstone FS-30 (25% solids in water, "Capstone") was obtained from Chemours and 2-trifluoromethyl-3-ethoxyperfluorohexane (HFE-7500) was obtained from Santa Cruz Biotechnology. Perfluorobutane sulfonic acid (PFBS) was purchased from TCI and perfluorobutanoic acid (PFBA) was obtained from Thermo Fisher. The fluorescent dye perylene (≥ 99%) was purchased from Fisher Scientific. Triton X-100 ("Triton") was obtained from Chem-Impex Int'l Inc. Negative photoresist SU-8 2150 was purchased from Fisher Scientific. Sylgard 184, a two-component system of poly(dimethylsiloxane) base and a curing agent, was obtained from Dow Chemical. All chemicals were used as received.

2. Preparation of synthetic groundwater

The synthetic groundwater (SGW) used in this study was chosen to replicate the characteristics of local water sources in Tucson, Arizona. We followed the formulation reported by Smith et al.[1] for synthetic hard water. We added the following chemicals listed in **Table S1** to Milli-Q water to prepare SGW. The chemicals were acquired from Fisher Scientific and VWR and used without further purification.

Table S1. List of chemical compounds used for SGW preparation in this study following the recipe reported by Smith et al.[1]

Chemical compounds	CAS#	Concentration (mg L^{-1})
Sodium sulfate [Na_2SO_4]	7757-82-6	28.06
Sodium bicarbonate [$NaHCO_3$]	144-55-8	22.68
Potassium bicarbonate [$KHCO_3$]	298-14-6	7.52
Potassium dihydrogen phosphate [KH_2PO_4]	7778-77-0	4.08
Calcium carbonate [$CaCO_3$]	471-34-1	83.2
Calcium chloride hexahydrate [$CaCl_2.6H_2O$]	7774-34-7	74.9
Calcium nitrate tetrahydrate [$Ca(NO_3)_2.4H_2O$]	13477-34-4	11.82
Magnesium sulfate heptahydrate [$MgSO_4.7H_2O$]	10034-99-8	100.4

3. Fabrication of Janus droplets

We employed an emulsification technique following the methodology outlined by Zarzar et al. to create monodispersed complex droplets consisting of hydrocarbon oil (H-oil) and fluorocarbon oil (F-oil) phases.[2] We used 0.1 wt% Triton solution as the continuous phase. The dispersed phase comprised of toluene with dissolved perylene as H-oil and a 9:1 mixture of HFE-7500 and FC-43 as F-oil. By maintaining the dispersed phase above the upper critical temperature (T_c = 32°C), H-oil and F-oil were miscible during emulsification. The ratio of HFE-7500 to FC-43 was selected to modulate the critical temperature (T_c) for miscibility and to ensure complete phase separation at room temperature.[3] We used the Dolomite Microfluidics setup, placed in an incubator to maintain a temperature above T_c (>32°C) during the fabrication process. Two Mitos P pressure pumps controlled the flow rates of the continuous and dispersed phases, into the flow-focusing chip, Telos 2 Reagent Chip (50 µm). Following emulsification, the droplets were cooled down to room temperature, resulting in double emulsions with equal volumes of H-oil and F-oil due to phase separation. We collected optical microscope images and analyzed the droplet diameter and dispersity using a MATLAB code.

We chose monodispersed droplets to minimize the influence of droplet size variations on optical properties, as reported by Zeininger et al.[3] Specifically, the normalized change in optical properties and emission intensity remained consistent across droplets with different diameters, depending only on internal morphology. However, the absolute emission intensity range was influenced by droplet size, highlighting the potential for signal confusion when collecting emissions from polydispersed samples without internal references. Additionally, a droplet diameter of approximately 50 µm was chosen because of the ease of integration into microfluidic-based sensors and sufficient stability during the duration of the experiment.
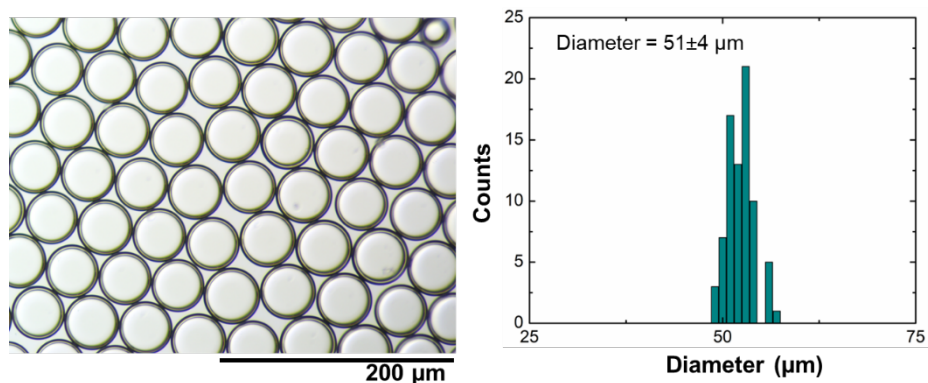
Figure S1. Top-view optical micrograph of monodispersed droplets fabricated using Dolomite Microfluidics Setup. Histogram of droplet diameter and standard deviation were analyzed from optical micrographs using a MATLAB code.

After emulsification of complex droplets in a solution of Triton, we adjusted the surfactant composition (Triton and Capstone) in the continuous phase to balance the interfacial tensions (γ_H and γ_F) in order to generate Janus droplets. We achieved the Janus state at $f_{Capstone}$ = 0.35. The value $f_{Capstone}$ is a proxy for the ratio between and γ_H and γ_F, and is defined as the ratio of the concentration of Capstone to the combined concentration of Triton and Capstone: $f_{Capstone} = \frac{[Capstone]}{[Triton]+[Capstone]}$. We confirmed the Janus state of droplets by capturing top-view and side-view optical micrographs.

4. Sensing mechanism of Janus droplets

To determine the relationship between droplet morphology and optical emission, we deposited a single layer of monodispersed Janus droplets in 350 µL of surfactant solutions. A bifurcated optical fiber was placed above the droplet layer to illuminate the droplets and capture emitted light (**Figure S2a**). The distance between droplets and the optical fiber was consistently maintained at 14 mm during data collection. The optical fiber transmits UV light at $\lambda = 405$ nm to excite the fluorescent dye and captures emission intensity at $\lambda = 475$ nm (one of the characteristic wavelengths of perylene). The normalized emission data exhibited peak intensity at a Capstone fraction ($f_{Capstone}$) near 70% (**Figure S2b**). We normalized the raw emission intensities to transform all dataset into a similar scale which helped to effectively compare between values obtained from different test runs. Also, it helped to minimize the effects of external factors, such as slight variations in number of droplets, exposure of ambient light, and slight deviation in the raw values of emission intensity.
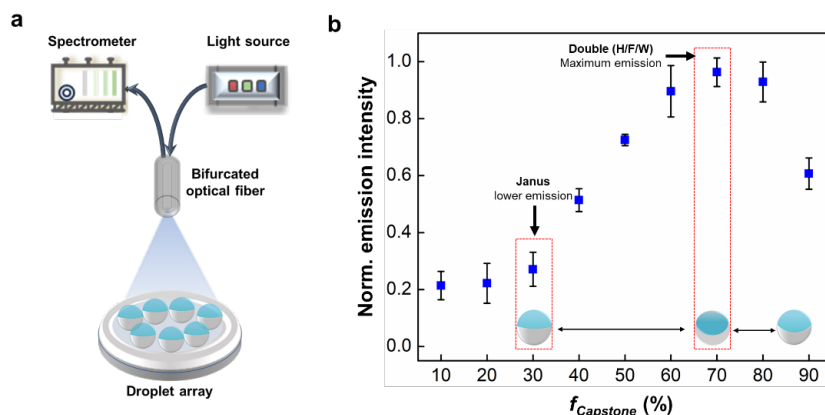

Figure S2. (a) Schematic representation of sensing mechanism of Janus droplets. (b) Light curve: emission intensity as a function of droplet morphology. Droplet morphology was controlled by altering the ratio of Triton and Capstone.

5. Fabrication of PDMS microfluidic sensing setup

PDMS microchips were patterned using conventional soft lithography.[4–6] First, silicon wafers were cleaned using a piranha solution (H_2SO_4:H_2O_2, 3:1 ratio) and subsequently dried with clean air. The wafers were then heated on a hot plate at 95°C for 15 minutes. A thin film of SU-8 2150 photoresist was deposited using a spin coater at 3000 rpm for 1 minute. The coated wafers were baked on a hot plate at 65°C for 25 minutes, followed by 95°C for 10 minutes. Exposure of UV light (λ = 365 nm) through a negative mask was then employed to pattern the photoresist. Subsequently, the wafers were baked again (65°C for 25 minutes, 95°C for 10 minutes, and 65°C for 5 minutes) and washed with PGMEA (Propylene glycol monomethyl ether acetate) to remove unexposed SU-8, resulting in the formation of SU-8 molds upon rinsing with isopropyl alcohol. We then treated the patterned wafers with 1H,1H,2H,2H perfluorooctyl trichlorosilane (FOTS) to develop hydrophobic surface. After mixing a 10:1 weight ratio of PDMS base to curing agent, PDMS solution was poured onto the patterned silicon wafers, followed by immediate placement in a vacuum desiccator to eliminate trapped air bubbles. After curing at room temperature for 24-36 hours, the cured PDMS replica was carefully peeled off from the wafer. Inlet and outlet holes were punched into the PDMS slab using disposable biopsy punches to achieve the desired microfluidic device design. Then the PDMS pieces were bonded to glass slides using an AutoGlow plasma system. Plasma treatment modified the surface chemistry, facilitating strong and permanent bonding between the PDMS with channels and the glass substrates, following thorough cleaning of both surfaces.

The real-time sensing setup includes two syringe pumps, PDMS microfluidic module, and a bifurcated optical fiber linked to a UV light source and a spectrophotometer.[7] We used two programmable syringe pumps to deliver the control solution (Triton and Capstone) and the PFAS solution (Triton, Capstone, and PFAS) through the microchannels at varying flow rates, durations, and flow ratios. We constructed three-stage PDMS microfluidic module comprising (1) a mixing channel, (2) a bubble trap, and (3) a sensing chamber, interconnected via PTFE tubes with an inner diameter of 0.5 mm.
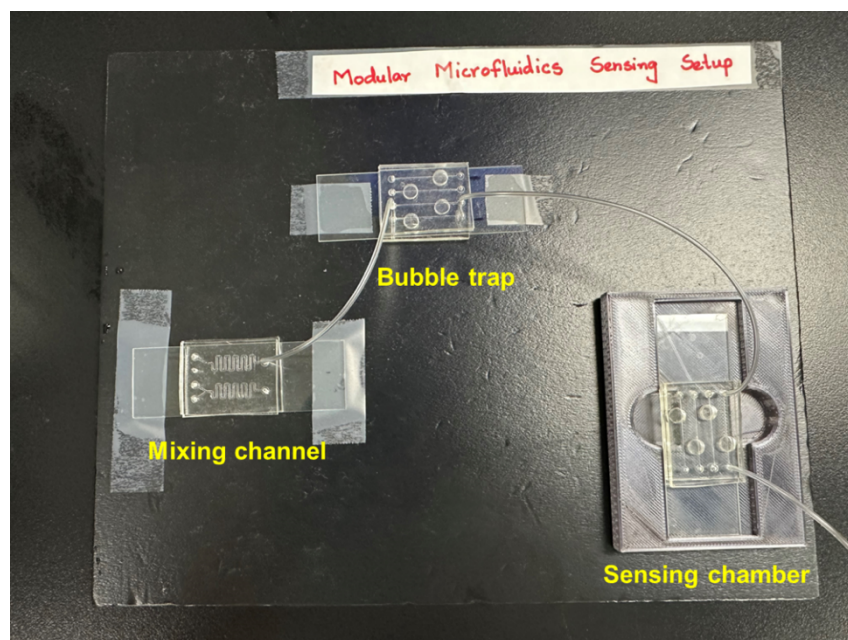
Figure S3. Images of PDMS microchips made by soft lithography process to build the three-stage microfluidics module for real-time and continuous monitoring.

The mixing channel was designed with serpentine microchannels to promote passive mixing of laminar flows. We confirmed effective mixing of two inlet streams of solutions at the outlet of the mixing channel under the experimental flow rates. Subsequently, the mixed outlet stream entered a vertical cylindrical-shaped microcavity, designed to trap air bubbles to prevent interference with measured fluorescence intensity or mechanical agitation of the droplets. The third module contains a cylindrical sensing chamber (diameter = 5 mm, depth of 3 mm), which accommodates a monolayer of Janus droplets and is linked to the inlet and outlet streams via a thin rectangular microchannel with a width of 0.5 mm and depth of 110 μm. Prior to each experiment, we flushed every PDMS module with 0.1 M NaOH solution and then rinsed it with Milli-Q water to ensure adequate hydrophilicity of the inner surfaces. We subsequently introduced a surfactant solution containing Janus droplets into the chamber to generate a single layer. For each experiment, constant total flow rate of 200 μL min^{-1} produced sufficiently fast response time (< 2 minutes) with minimal disruption to the droplet monolayer and minimum loss of droplets with the liquid flow.[7] Here we observed a consistent time delay of 25 ± 3 seconds between the input flow and the achieving equilibrium emission intensity.
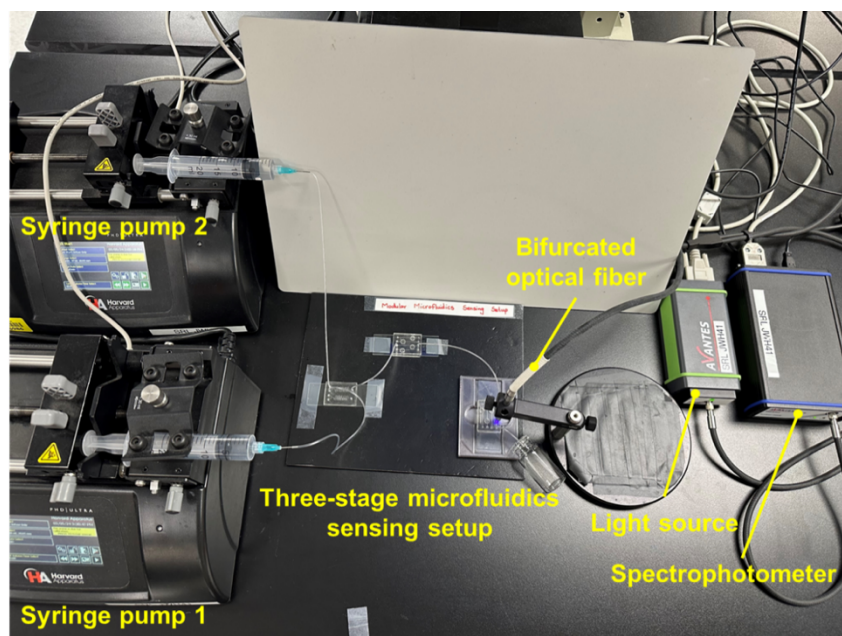
Figure S4. Real-time sensing setup of PDMS microfluidics with Janus droplets.

6. Classification model using Python

We conducted Random Forest classification using the Python programming language. The following libraries and steps are followed to implement the algorithm.

Step 1: Import necessary libraries

- `pandas`: This library is used for data manipulation and analysis. It provides data structures and functions to work with structured data.
- `numpy`: NumPy is a fundamental package for numerical computing with Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.
- `sklearn`: It is known as Scikit-learn, a machine learning library in Python. It provides simple and efficient tools for data mining and data analysis, including various machine learning algorithms and evaluation metrics.
- `matplotlib` and `seaborn`: These libraries are used for data visualization. Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Seaborn is built on top of Matplotlib and provides a high-level interface for generating informative statistical graphics.

Step 2: Load and prepare data

The dataset was loaded into a pandas DataFrame by reading the .csv file created earlier using the experimental outcomes.

Step 3: Split data into training and testing sets

The dataset was split into training and testing sets by randomly selecting two out of three samples from each experimental configuration. This step is crucial for evaluating the model's performance.

Step 4: Train the Random Forest model

The Random Forest classifier was initialized using `RandomForestClassifier` from `sklearn.ensemble` and the training data was fit. This step trains the model on the training dataset to learn patterns and relationships in the data.

Step 5: Make predictions

The trained model was used to make predictions on the testing dataset. The `predict` method of the Random Forest classifier is used to generate predictions based on the features of the test data.

Step 6: K fold cross validation

Three different training and testing splits were created and steps **4** and **5** were repeated on those splits.

Step 7: Evaluate model performance

The outcomes of three model runs were aggregated to calculate accuracy and other performance metrics such as precision, recall, and F1-score using `accuracy_score,` `classification_report,` and `confusion_matrix` from `sklearn.metrics.` Visualization of the confusion matrix was done using `seaborn` and `matplotlib.` These steps help assess how well the model performed in predicting the PFAS class.

7. Real-time emission responses of PFOS in different concentration range

By adjusting the concentration of control surfactants (Triton and Capstone), we could detect PFOS at different concentration ranges.[8] As the emission intensities depend on the interfacial tensions, the sensitivity range can be tuned. We demonstrated the utility of three different sets of Janus droplets to detect PFOS in the total range of 5 µg L^{-1} to 200 µg L^{-1}. We started with the first set of Janus droplets in 0.1 wt% Triton and 0.05 wt% Capstone and detected PFOS in the high concentration range of 120 to 200 µg L^{-1}. Then to detect different ranges of PFOS concentrations, we fabricated a second set of Janus droplets in 0.01 wt% and 0.005 wt% Capstone to target medium range of PFOS (50 to 100 µg L^{-1}) and a third set of Janus droplets in 0.001 wt% Triton and 0.0005 wt% Capstone for the low range of PFOS (5 to 40 µg L^{-1}). The real-time emission intensity as the function of concentration for the high, medium, and low ranges of PFOS are shown in **Figure S5 and 3b**. We observed that the change in normalized emission intensities for the different PFOS concentration range are different.

Table S2. List of PFOS detection range and control surfactant concentrations

Range	PFOS Concentration (µg L^{-1})	Control Solution Concentration (Triton:Capstone) (wt%)
1. High	120-200	0.1: 0.05
2. Medium	50-100	0.01: 0.005
3. Low	5-40	0.001: 0.0005
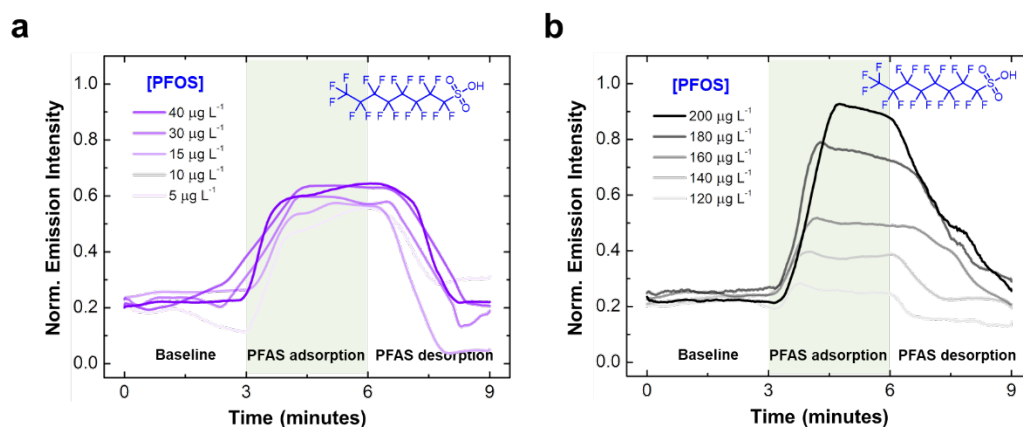
Figure S5. Real-time emission responses for PFOS for concentration range of (a) 5 to 40 µg L^{-1} and (b)120 to 200 µg L^{-1}.

8. Real-time emission responses of PFOA, PFBS, and PFBA

Similar to the PFOS detection shown in **Figure 3b**, we conducted experiments for the other three PFAS: PFOA, PFBS, and PFBA in the concentration range of 50 to 100 μg L^{-1}. We used the second set of Janus droplets for the detection of all PFAS in this concentration range. Then all the real-time, continuous emission data sets were normalized (**Figure S6**), and we extracted the four key features, which were employed for analysis and classification models.
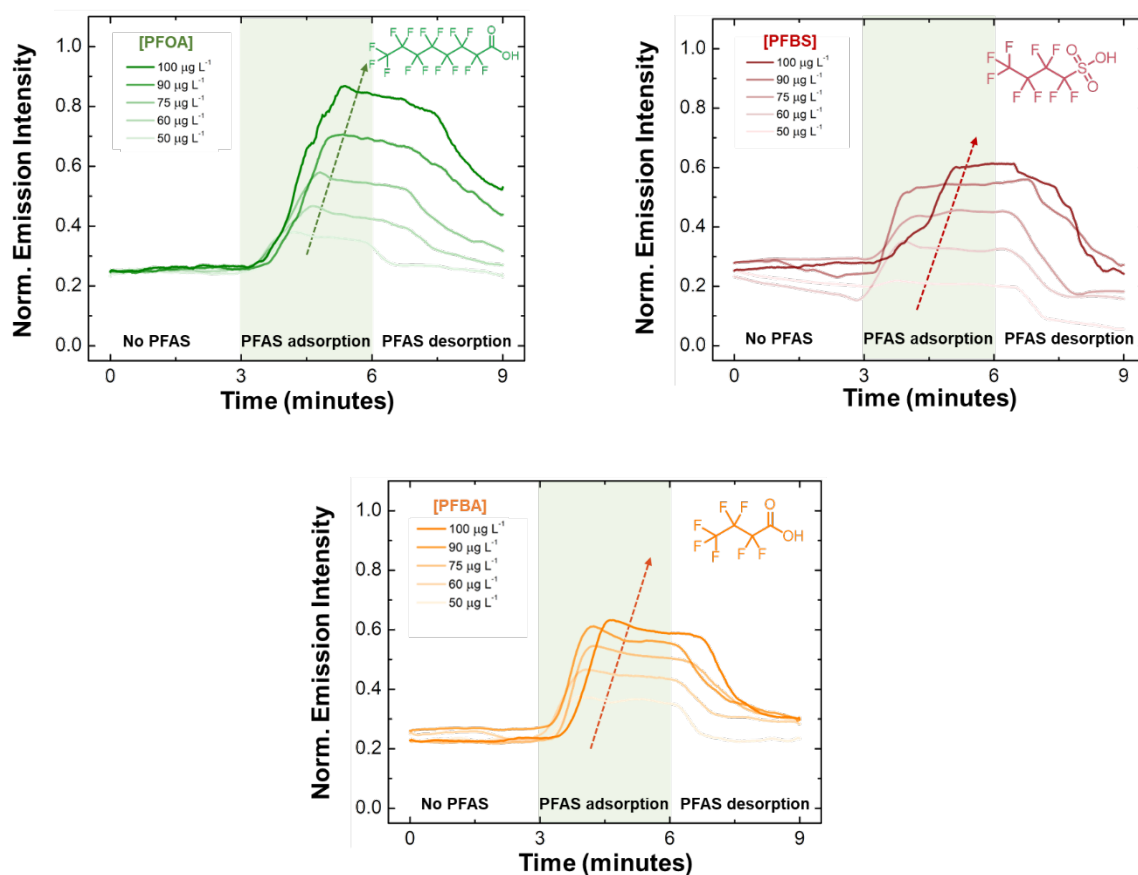
Figure S6. Real-time emission responses for PFOA, PFBS, and PFBA for concentration range 50 to 100 μg L^{-1}.

9. Principal Component Analysis

Principal Component Analysis (PCA) serves as a method for reducing the dimensionality of extensive datasets.[9] By transforming numerous variables into a smaller set that retains the majority of the original information, PCA simplifies data analysis without sacrificing substantial accuracy. It achieves this by creating principal components—new variables formed as linear combinations of the original variables. These components are designed to be uncorrelated, with the most significant information from the initial variables concentrated in the first few components. This reduction allows for easier exploration and visualization of the data, enhancing the efficiency of machine learning algorithms. Ultimately, the principal components extracted through PCA enable a clearer understanding of the data's structure and variability, making complex datasets more manageable and interpretable for better understanding.

The steps involved for performing PCA are:

Step 1: Normalization of dataset.

$$x_{new} = \frac{x - \mu}{\sigma}$$

Step 2: Calculation of covariance matrix for the features in the dataset.

$$Cov(x, y) = \frac{\sum (x_i - \bar{x}) * (y_i - \bar{y})}{N - 1}$$

Step 3: Calculation of eigen values and eigen vectors for the covariance matrix.

Step 4: Sorting of eigen values and their corresponding eigen vectors.

Step 5: Selection of top eigen values and formation of a matrix of eigen vectors.

Step 6: Transformation of the original matrix.

$$Feature\ matrix * top\ eigen\ vectors = Transformed\ data$$

PFAS in Milli-Q water

Conc. 50-100 µg L^{-1}

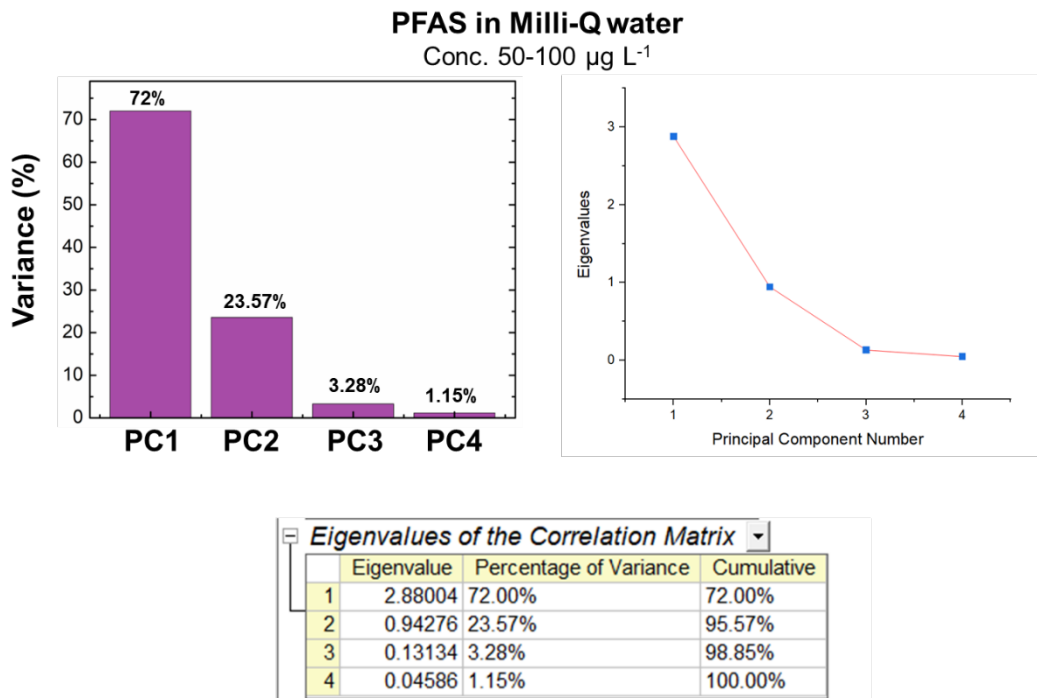|	Eigenvalue	Percentage of Variance	Cumulative
1	2.88004	72.00%	72.00%
2	0.94276	23.57%	95.57%
3	0.13134	3.28%	98.85%
4	0.04586	1.15%	100.00%

Eigenvalues of the Correlation Matrix

Figure S7. Variance percentage of principal components for PFAS classification.

10. Random Forest classification algorithm

Random Forest (RF) is a general-purpose classification and regression method that combines several randomized decision trees and aggregates their predictions by averaging the outcomes.[10,11] The algorithm is built based on the concept of the decision tree. A decision tree is a type of supervised learning algorithm that is commonly used in machine learning to model and predict outcomes based on input data. It is a tree-like structure where each internal node tests on attribute, each branch corresponds to attribute value and each leaf node represents the final decision or prediction. Even though a decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks, it has some limitations. The decision tree algorithm is slow and tends to overfit due to its reliance on a single tree structure driven by decision rules at each node. In contrast, the Random Forest algorithm addresses these shortcomings by generating multiple decision trees based on subsets of the data. This process unfolds in two phases: first, creating a random forest by generating N decision trees, and second, combining the collective predictions of each tree to make accurate predictions. We followed the following steps to perform RF classification:

Step 1: Pick M data points at random from the training set.

Step 2: Create decision trees for chosen data points (subsets).

Step 3: Each decision tree will produce a result. Analyze it.

Step 4: For classification and regression, accordingly, the final output is based on maximum likelihood for classification or averaging for regression.

The model is implemented by using the Python programming and random forest classifier in the scikit-learn package with default parameters. Here we performed 3-fold cross-validation to ensure the robustness of the classification algorithm. A total of 60 datasets was split into training and testing sets in a 67/33 ratio, respectively. Since the dataset at each concentration consists of three data points, instead of the random split, we adopted a controlled split. At each training step, two out of three data points were selected for training and one data point was used for the testing. This process ensures that the model is trained to predict the PFAS class at different concentration levels. The classification report was generated after each training and testing phase which shows the performance metrices such as precision, recall, f1-score, and overall accuracy for both training and testing datasets. The result is summarized in the table below:

Table S3: RF classification results for the four PFAS

Analyte	Average Training Results in %				Average Testing Results in %			
	Precision	Recall	f1-score	Accuracy	Precision	Recall	f1-score	Accuracy
PFOS	100	100	100		59	40	46	
PFOA	100	100	100	100	94	100	97	77
PFBS	100	100	100		100	100	100	
PFBA	100	100	100		53	67	59	

The average PFAS classification accuracy obtained is 77% where most of the reduction in accuracy was due to PFOS and PFBA misclassification. Further investigation of the confusion matrix reveals the classification model's tendency to misclassify PFOS and PFBA due to inherent similarities in the input features. Hence, the dataset was reanalyzed by repeating the steps described above by excluding PFOS and PFBA individually. In the former case, an accuracy of 100% was achieved, underscoring the model's proficiency in accurately classifying PFAS compounds excluding PFBA. In the latter case, where PFOS was excluded, the average accuracy reached 98% (**Figures S8 and S9**). This reinforced the model's capability to accurately label PFOS, PFOA, and PFBS by analyzing the four features derived from real-time data (**Figures S8 and S9**).

Table S4: RF classification results for PFOS, PFOA, and PFBS

Analyte	Average Training Results in %				Average Testing Results in %			
	Precision	Recall	f1-score	Accuracy	Precision	Recall	f1-score	Accuracy
PFOS	100	100	100		100	100	100	
PFOA	100	100	100	100	100	100	100	100
PFBS	100	100	100		100	100	100	

Table S5: RF classification results for PFOA, PFBS, and PFBA

Analyte	Average Training Results in %				Average Testing Results in %			
	Precision	Recall	f1-score	Accuracy	Precision	Recall	f1-score	Accuracy
PFOA	100	100	100		94	100	97	
PFBS	100	100	100	100	100	100	100	98
PFBA	100	100	100		100	93	96	

Confusion matrix for RF classification excluding PFBA (three trials)

Figure S8. Confusion matrix for three trial runs excluding PFBA.

Confusion matrix for RF classification excluding PFOS (three trials)

	precision	recall	f1-score	support
1	0.83	1.00	0.91	5
2	1.00	1.00	1.00	5
3	1.00	0.80	0.89	5
accuracy			0.93	15
macro avg	0.94	0.93	0.93	15
weighted avg	0.94	0.93	0.93	15

	precision	recall	f1-score	support
1	1.00	1.00	1.00	5
2	1.00	1.00	1.00	5
3	1.00	1.00	1.00	5
accuracy			1.00	15
macro avg	1.00	1.00	1.00	15
weighted avg	1.00	1.00	1.00	15

	precision	recall	f1-score	support
1	1.00	1.00	1.00	5
2	1.00	1.00	1.00	5
3	1.00	1.00	1.00	5
accuracy			1.00	15
macro avg	1.00	1.00	1.00	15
weighted avg	1.00	1.00	1.00	15

Figure S9. Confusion matrix for three trial runs excluding PFOS.

11. Linear Regression analysis for quantifying PFAS concentrations

After identifying the PFAS type, to predict the concentration of identified PFAS we performed the multiple linear regression analysis. We used a linear regression model, which is widely used and provides an easy-to-interpret mathematical formula that can generate predictions. Multiple regression is an extension of linear regression models that allow predictions of systems with multiple independent variables. Multiple regression is specifically designed to create regressions on models with a single dependent variable and multiple independent variables. Here we adopted it to predict the PFAS concentration based on the input feature information. The mathematical model can be written as following equation:

$$y_{conc} = \beta_0 + \beta_{F1}\, x_{F1} + \beta_{F2}\, x_{F2} + \beta_{F3}\, x_{F3} + \beta_{F4}\, x_{F4} + \epsilon$$

Here y_{conc} is the predicted concentration of the analyte, which is a dependent variable, β_0 is the intercept, and β_{F1}, β_{F2}, β_{F3}, β_{F4} are the coefficients of the corresponding explanatory variables (input features). ϵ represents the error, which is the deviation of the predicted value from the actual. The goal of the regression model is to determine the intercept and coefficients so that the sum of the square residuals is minimized.

Before fitting the model, pairwise scatter plots and correlations were examined to determine whether linear regression is a good fit based on the data. Those revealed that a linear trend exists between the concentration and the input variables. After that, the regression models are fitted to determine the intercept and coefficients. The fitted regression models are then used to predict the concentration. The outcome of the regression models is summarized in the table below.

Table S6: Summary of Linear regression results for the four PFAS

Analyte	Intercept		I_{ads} (F1)		I_{des} (F2)		t_{ads} (F3)		t_{des} (F4)		R^2		Mean Absolute Error
	Coefficient	P-Value	Coefficient	P-Value	Coefficient	P-Value	Coefficient	P-Value	Coefficient	P-Value	Regression Line	QQ plot line	
PFOS	32.39	0.00	0.73	0.05	-0.23	0.08	-0.04	0.84	0.20	0.04	0.99	0.96	1.70
PFOA	20.42	0.00	0.35	0.13	0.04	0.78	0.34	0.01	0.17	0.01	0.99	0.93	1.19
PFBS	18.60	0.00	0.89	0.00	0.21	0.00	-0.25	0.17	0.41	0.02	0.99	0.91	2.38
PFBA	37.64	0.00	0.24	0.72	-0.18	0.47	0.29	0.35	0.24	0.13	0.99	0.88	2.14

The table shows the intercept and coefficient values of the regression model for each PFAS, corresponding P-values, the coefficient of determination (R^2), and the mean absolute error. The P-value is a statistical number to conclude if there is a relationship between analyte concentration and the observed features. Here the P-value comes from a hypothesis test where the null hypothesis is the true value of the coefficients and intercept are equal to zero (no relationship) and vice versa for the alternate hypothesis (have relationship). The P-value is calculated based on the t-statistic where a low P-value (< 0.05) means that the coefficient is likely not to equal zero. On the other hand, a high P-value (> 0.05) means that we cannot conclude that the explanatory variable (F1, F2, F3, and F4) affects the dependent variable (concentration). The R^2 value which signifies the goodness of fit (i.e., 1 means perfect fit) is almost 1 for all four models which means the linear regression model is a good fit for the dataset. An underlying assumption of the regression model is the normality of residuals. In order to validate that, the QQ plots were generated for each PFAS type which show that residuals for all models follow the normal distribution. The R^2 values for the fitted line in QQ-plot vary from 0.88 to 0.96. The regression line for PFBA is not a good fit as three out of four coefficients show insignificant P-values even though the R^2 value is high. Also, PFBA regression model shows the largest deviation from the normality of the residuals. This finding also reinforces the outcome from RF classification where PFBA is mostly misclassified. Additionally, the mean absolute error of the models for four PFAS ranges from 1.19 to 2.38 µg L⁻

[1] which means the predicted values are close to the actual values. Therefore, the regression model can predict concentration with higher accuracy.
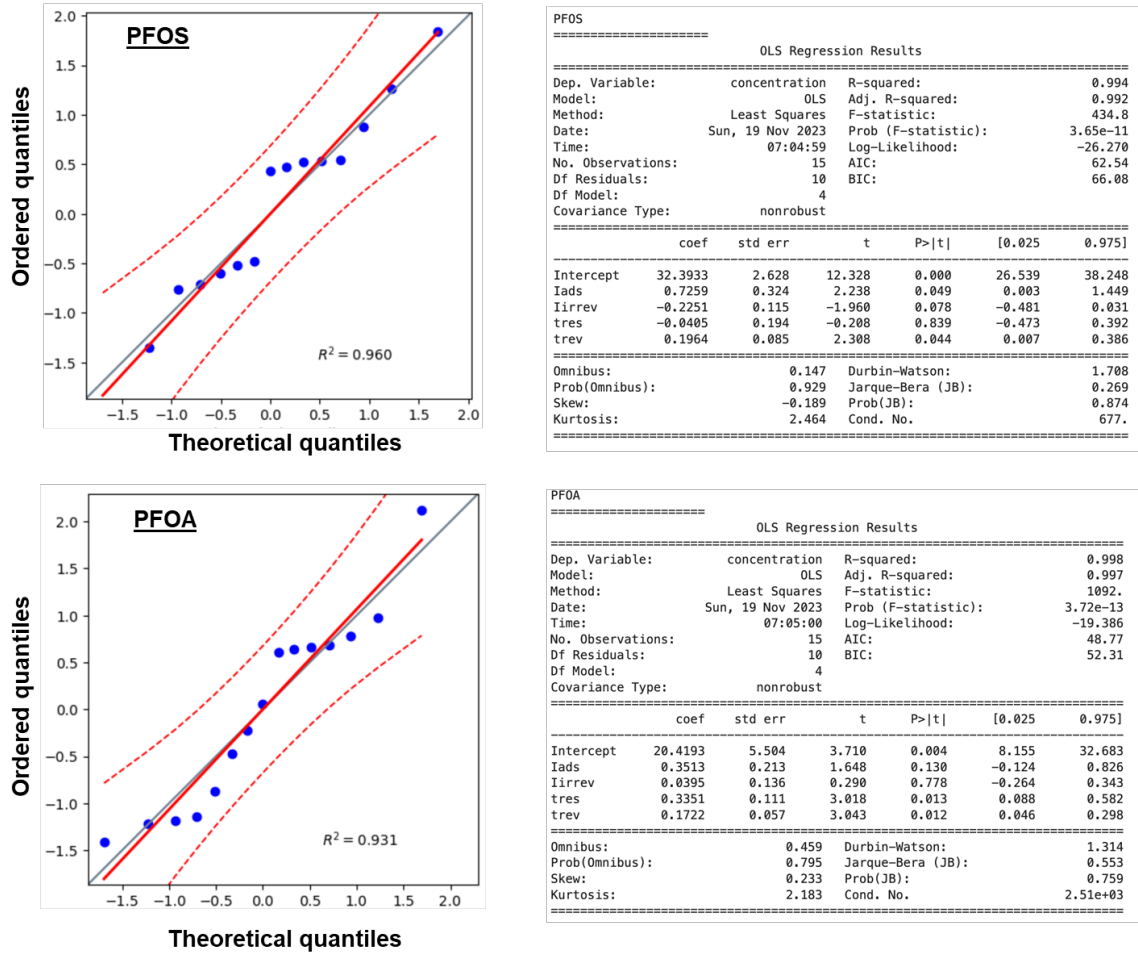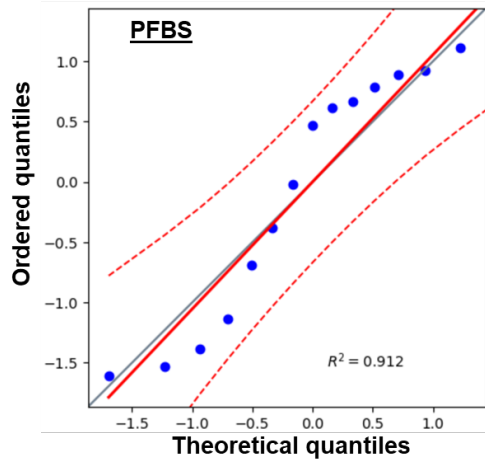

Figure S10. Regression lines for PFOS and PFOA.

PFBS

```
PFBS
=======================
                            OLS Regression Results
==============================================================================
Dep. Variable:           concentration   R-squared:                       0.992
Model:                             OLS   Adj. R-squared:                  0.988
Method:                  Least Squares   F-statistic:                     301.3
Date:                 Sun, 19 Nov 2023   Prob (F-statistic):           2.25e-10
Time:                         07:05:01   Log-Likelihood:                -29.000
No. Observations:                   15   AIC:                             68.00
Df Residuals:                       10   BIC:                             71.54
Df Model:                            4
Covariance Type:             nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     18.5969      2.202      8.444      0.000      13.690      23.504
Iads           0.8884      0.053     16.701      0.000       0.770       1.007
Iirrev         0.2079      0.028      7.491      0.000       0.146       0.270
tres          -0.2541      0.170     -1.493      0.166      -0.633       0.125
trev           0.4063      0.139      2.933      0.015       0.098       0.715
==============================================================================
Omnibus:                         4.107   Durbin-Watson:                   1.404
Prob(Omnibus):                   0.128   Jarque-Bera (JB):                1.603
Skew:                           -0.407   Prob(JB):                        0.449
Kurtosis:                        1.621   Cond. No.                         352.
==============================================================================
```
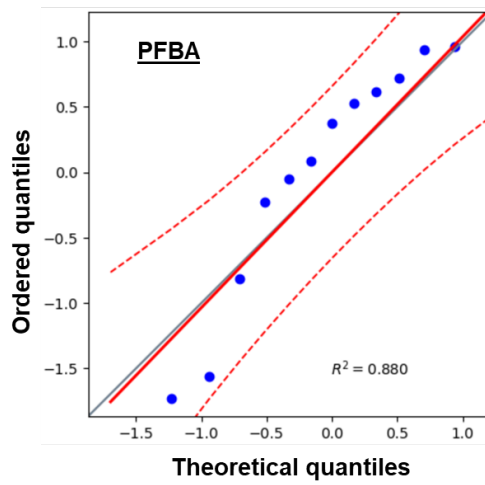

PFBA

```
PFBA
=======================
                            OLS Regression Results
==============================================================================
Dep. Variable:           concentration   R-squared:                       0.993
Model:                             OLS   Adj. R-squared:                  0.990
Method:                  Least Squares   F-statistic:                     342.2
Date:                 Sun, 19 Nov 2023   Prob (F-statistic):           1.20e-10
Time:                         07:05:01   Log-Likelihood:                -28.053
No. Observations:                   15   AIC:                             66.11
Df Residuals:                       10   BIC:                             69.65
Df Model:                            4
Covariance Type:             nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     37.6362      4.145      9.080      0.000      28.401      46.872
Iads           0.2400      0.646      0.371      0.718      -1.200       1.680
Iirrev        -0.1812      0.243     -0.744      0.474      -0.724       0.361
tres           0.2889      0.292      0.988      0.346      -0.362       0.940
trev           0.2406      0.144      1.672      0.125      -0.080       0.561
==============================================================================
Omnibus:                         2.539   Durbin-Watson:                   1.138
Prob(Omnibus):                   0.281   Jarque-Bera (JB):                1.933
Skew:                           -0.780   Prob(JB):                        0.380
Kurtosis:                        2.186   Cond. No.                         962.
==============================================================================
```

Figure S11. Regression lines for PFBS and PFBA.

12. Real-time emission responses of PFAS binary mixtures

We tested interfacial behaviors of PFAS when they are in mixtures. We prepared equimolar mixtures for four different PFAS combinations in Milli-Q water. Following the same experimental procedure (square wave input), we collected real-time emissions at different ratios for each binary mixture of PFAS. We observed competition between the PFAS at the droplet interfaces, specifically long-chain mostly dominating the interfacial properties, which led to non-linear behavior of emission change during adsorption and desorption.
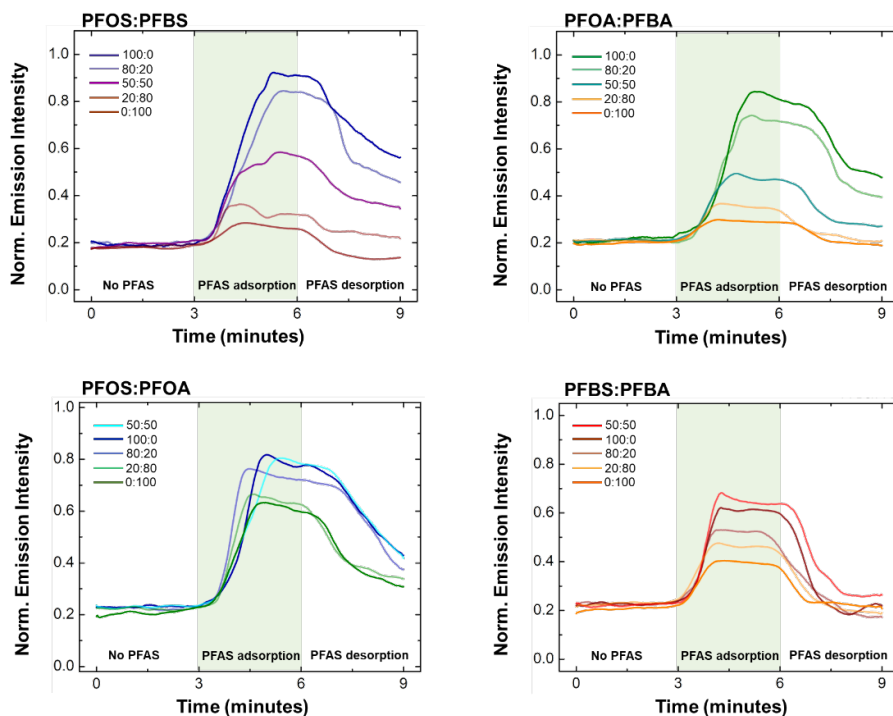
Figure S12. Real-time emission responses for PFAS binary equimolar mixtures at different ratio.
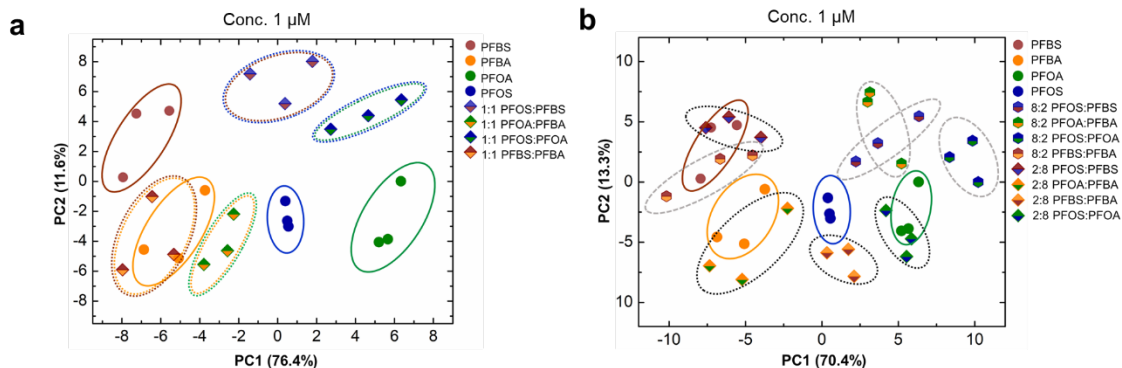

Figure S13. PCA plots for PFAS separation in binary mixtures prepared in Milli-Q water. (a) Pure PFAS and 1:1 binary mixture. (b) Pure PFAS and other ratio of binary mixtures.

13. Real-time emission responses of the four PFAS in SGW

Next, we tested PFAS in SGW containing dissolved ions. We followed the same experimental procedure (square wave input) and collected real-time emission response for the four PFAS (from 50 to 100 µg L^{-1}) in SGW. All four PFAS showed greater interfacial activity in the presence of cations. Then we performed both PCA and RF classification using the sensing data of PFAS in SGW. The confusion matrix generated utilizing the four key features of the four PFAS in the concentration range of 50 to 100 µg L^{-1}. We observed that the classification accuracy (64%) is lower than the Milli-Q water matrix. This outcome and PCA plots suggest that PFAS interfacial behavior gets affected by ions present in the system which makes it challenging to classify.
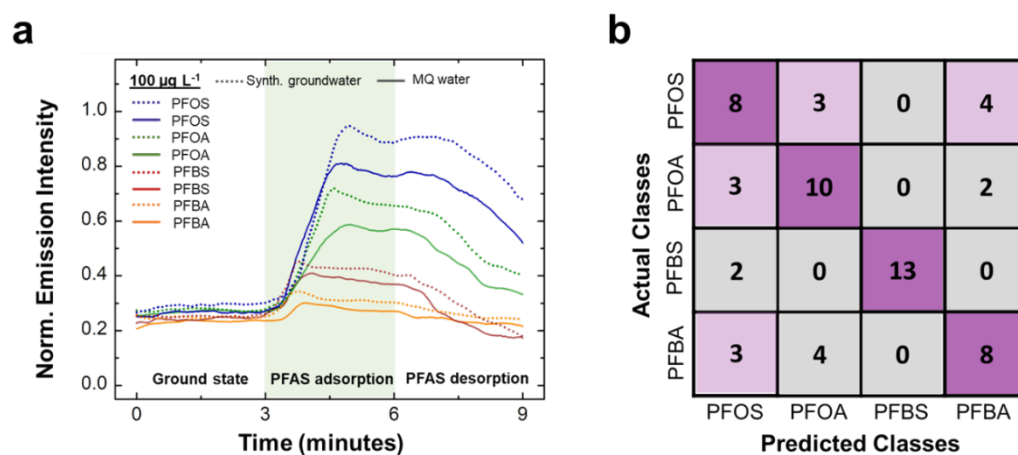

Figure S14. (a) Real-time emission responses of the four PFAS in SGW and Milli-Q water at 100 µg L^{-1}. (b) RF classification outcome for PFAS (50 to 100 µg L^{-1}) in SGW.

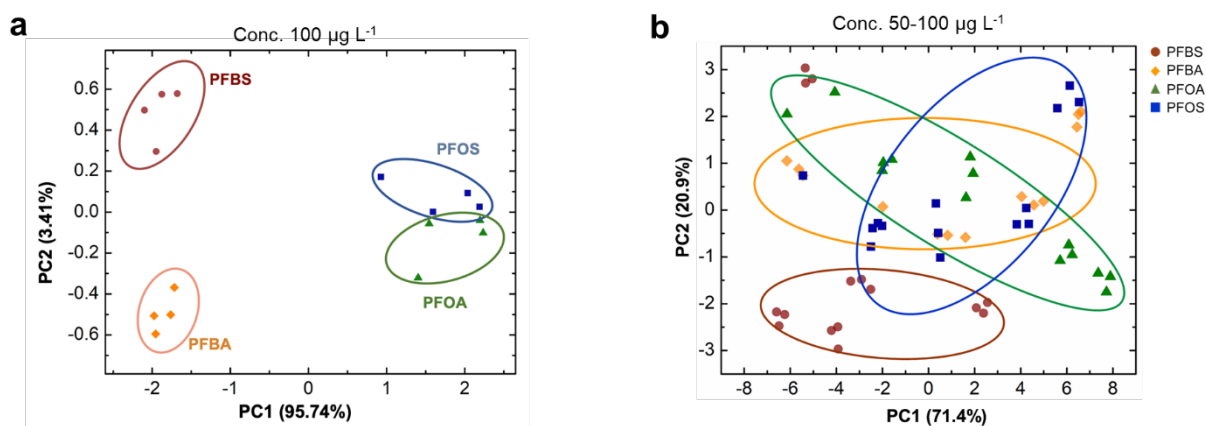**Figure 15.** PCA plots for PFAS separation in SGW. (a) for PFAS concentration of 100 µg L^{-1}. (b) within a range of concentration for 50 to 100 µg L^{-1}.

References

(1)   Smith, E. J.; Davison, W.; Hamilton-Taylor, J. Methods for Preparing Synthetic Freshwaters. *Water Res.* **2002**, *36* (5), 1286–1296. https://doi.org/10.1016/S0043-1354(01)00341-4.

(2)   Zarzar, L. D.; Sresht, V.; Sletten, E. M.; Kalow, J. A.; Blankschtein, D.; Swager, T. M. Dynamically Reconfigurable Complex Emulsions via Tunable Interfacial Tensions. *Nature* **2015**, *518* (7540), 520–524. https://doi.org/10.1038/nature14168.

(3)   Zeininger, L.; Nagelberg, S.; Harvey, K. S.; Savagatrup, S.; Herbert, M. B.; Yoshinaga, K.; Capobianco, J. A.; Kolle, M.; Swager, T. M. Rapid Detection of Salmonella Enterica via Directional Emission from Carbohydrate-Functionalized Dynamic Double Emulsions. *ACS Cent. Sci.* **2019**, *5* (5), 789–795. https://doi.org/10.1021/acscentsci.9b00059.

(4)   Kim, P.; Kwon, K. W.; Park, M. C.; Lee, S. H.; Kim, S. M.; Suh, K. Y. Soft Lithography for Microfluidics: A Review. *Biochip J.* **2008**, *2* (1), 1–11.

(5)   Friend, J.; Yeo, L. Fabrication of Microfluidic Devices Using Polydimethylsiloxane. *Biomicrofluidics* **2010**, *4* (2), 026502. https://doi.org/10.1063/1.3259624.

(6)   Raj M, K.; Chakraborty, S. PDMS Microfluidics: A Mini Review. *J. Appl. Polym. Sci.* **2020**, *137* (27), 1–14. https://doi.org/10.1002/app.48958.

(7)   Barua, B.; Durkin, T. J.; Beeley, I. M.; Gadh, A.; Savagatrup, S. Multiplexed and Continuous Microfluidic Sensors Using Dynamic Complex Droplets. *Soft Matter* **2023**, *19* (10), 1930–1940. https://doi.org/10.1039/d3sm00074e.

(8)   Trinh, V.; Malloy, C. S.; Durkin, T. J.; Gadh, A.; Savagatrup, S. Detection of PFAS and Fluorinated Surfactants Using Differential Behaviors at Interfaces of Complex Droplets. *ACS Sensors* **2022**, *7* (5), 1514–1523. https://doi.org/10.1021/acssensors.2c00257.

(9)   Archana, T. *Dimensionality Reduction and Classification through PCA and LDA*; 2015; Vol. 122.

(10)   Svetnik, V.; Liaw, A.; Tong, C.; Christopher Culberson, J.; Sheridan, R. P.; Feuston, B. P. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (6), 1947–1958. https://doi.org/10.1021/ci034160g.

(11)   Han, S.; Williamson, B. D.; Fong, Y. Improving Random Forest Predictions in Small Datasets from Two-Phase Sampling Designs. *BMC Med. Inform. Decis. Mak.* **2021**, *21* (1), 1–9. https://doi.org/10.1186/s12911-021-01688-3.