Supporting Information

Modeling the transplacental transfer of small molecules: A case study on poly/perfluorinated substances (PFAS)

Dimitri Abrahamsson1, Adi Siddharth1, Joshua F. Robinson1, Anatoly Soshilov2, Sarah Elmore2, Vincent Cogliano2, Carla Ng3, Elaine Khan2, Randolph Ashton4-6, Weihsueh Chiu7, Lauren Zeise2 and Tracey J. Woodruff\*1

1Department of Obstetrics, Gynecology and Reproductive Sciences, Program on Reproductive Health and the Environment, University of California, San Francisco, 490 Illinois Street, San Francisco, CA 94143, United States.

2California Environmental Protection Agency, Office for Environmental Health and Hazard Assessment, 1515 Clay St., Oakland, CA 94612, United States.

3Department of Civil and Environmental Engineering, University of Pittsburgh, 3700 O'Hara St, PA 15261, United States.

4Wisconsin Institute for Discovery, University of Wisconsin, Madison, 330 N Orchard St, Madison, WI 53715 Wisconsin

5The Stem Cell and Regenerative Medicine Center, University of Wisconsin, Madison, 1111 Highland Avenue, Madison, WI 53705 Wisconsin

6Department of Biomedical Engineering, University of Wisconsin - Madison, 1550 Engineering Drive, Madison, Wisconsin 53706, United States

7Department of Veterinary Integrative Biosciences, College of Veterinary Medicine and Biomedical Sciences, Texas A&M University, College Station, Texas 77843, United States.

\***Corresponding author**: Tracey J. Woodruff, [tracey.woodruff@ucsf.edu](mailto:tracey.woodruff@ucsf.edu)

## Text S1: Machine learning approaches and parametrization

### Support Vector Machine

SVM is a supervised machine learning model that uses the Vapnik-Chervonenkis theory 54,55 to generate the optimal hyperplane, also known as decision surface, that separates in the best possible way objects with two or more labels.52–55 The labels can be either categories (e.g., red or blue) in categorical problems or ranges of values (e.g., 2-4 or 4-8) in regression problems. One interesting feature of SMVs is that they make projections of the datapoints onto a higher-dimensional space (≥ 2D) where one can linearly separate the data into groups using specific kernel functions, e.g., linear, polynomial or radial basis.52–55 For the purposes of our exercise, we used the SVM package from scikit-learn45 applied to regression problems, called SVR. The parameters we tested and tuned in our SVM algorithm were: i) the kernel function, which specifies the kernel function to be used in the algortithm; ii) the epsilon (ε) parameter, which specifies the range of values within which no penalty is given in the training loss function when datapoints are predicted within a distance epsilon from the actual value; and iii) the C parameter, which determines the influence of misclassification on the function. Lower C values cause the optimizer to look for larger-margin hyperplanes, whereas higher C values result in smaller-margin hyperplanes. The parameters and the results of the optimization are presented in the Supporting Information Spreadsheet.

### Random Forest

Random Forest (RF) is a supervised machine learning algorithm that learns by creating a multitude of decision trees to solve a classification or a regression problem. In the case of a classification problem, the decision trees are built to distinguish between two or more categories (e.g., red, yellow or green). In regression problems, these categories can be ranges of values (e.g., 3-5 or 6-6) and the decision trees are used to calculate a specific value instead of predicting a category. The algorithm starts by randomly selecting a subset of observations from the dataset (hence the term “random”) and by creating a decision tree based on these observations. The process is then repeated multiple times creating a “forest” of decision trees, where every tree makes a prediction for a given datapoint. In classification problems, the output of the algorithm is the majority vote of all the individual tree predictions, whereas in regression problems the output of the algorithm is the average of all the predictions made by the individual trees.56 For the purposes of our exercise, we used the random forest package from scikit-learn45. The parameters we tested and tuned in our RF algorithm were: i) the number of trees in the forest (n\_estimators), ii) whether bootstrap samples are used when building trees (bootstrap = True or False), iii) the maximum depth of the tree (max\_depth), iv) the maximum number of features to consider when looking for the best split (max\_features), vi) the minimum number of samples (min\_samples\_leaf) required to be in leaf node, and the minimum number of samples required to split an internal node (min\_samples\_split) (Supporting Information Spreadsheet).

### Artificial Neural Network

Artificial Neural Networks (ANNs) are computing systems composed of a collection of interconnected nodes, called neurons. Each connection can be viewed as a synapse transmitting information from one node to another, similar to the neurons and synapses in the human brain. Neurons are the core processing units of an ANN.57 They are used to build an input layer, which collects information from our input data, and an output layer which outputs the predictions of the model. In between the input and the output layer, exist the hidden layers which perform most of the computations in the network. The neurons of the first layer are connected to those of the second layer through channels and each channel is assigned a numerical value known as weights. The inputs of the first layer are then multiplied by the weights and their result is sent and to the neurons of the second layer (hidden layer). Each of these neurons are associated with a certain value known as biases which are added to the inputs from the first layer. This information passes then through a function, called activation function, which then determines whether the neuron will be activated. Some examples of activation functions are sigmoid, exponential and rectified linear unit. This process is repeated for every layer until we reach the output layer which gives out the prediction of the model. This process is known as forward propagation. The model then compares the generated prediction to the input datapoint and returns to the hidden layers to adjust the weights and biases so that the predictions match the inputs. This process is known as backpropagation. During training, the processes of forward propagation and backpropagation are repeated multiple times and each cycle is called an epoch.57 For the purposes of our exercise, we used the Tensorflow46 platform to build the ANN and we tested (i) variations in number of layers, (ii) number of nodes per layer, (iii) activation functions, (iv) dropout layer, which is an extra layer meant to remove (dropout) unnecessary channels between nodes created during the training to avoid overfitting, (v) training optimizer, which the algorithm used in the optimization of the model during the processes of forward propagation and backpropagation, and (vi) optimizing step, which is the step at which the optimizer operates. The parameters and settings for each model together with the results of the optimization process are presented in the Supporting Information Spreadsheet.

![Chart, histogram

Description automatically generated]()

Figure S1: Distribution of log *R*CM values (central tendency) in the compiled database with descriptive statistics.

![Chart, box and whisker chart

Description automatically generated]()

Figure S2: Number of participants in each study used in the compiled database presented by chemical category. The number of participants in the collected studies ranged from 1 to 1000s with the majority of the studies being in the 10 to 100 participant range. Some of the chemical categories with the highest number of participants were thyroid hormones, plasticizers, phthalates and PFAS.

![Diagram

Description automatically generated with low confidence]()

Figure S3: Principal component analysis (A) and chemical space plots (B and C) for the compiled maternal-cord database (*R*CM) and for the PFAS database from EPA’s CompTox Chemicals Dashboard (PFAS master list v2). The plots show the principal components 1, 2 and 3 of the Mordred physicochemical properties of the chemicals in the *R*CM database (n=264) and in the PFAS database (n=7,986).

![Diagram

Description automatically generated]()

Figure S4: Chemical space plots using kernel density estimations for the compounds in the compiled maternal-cord database (*R*CM) and for the PFAS database from EPA’s CompTox Chemicals Dashboard (PFAS master list v2). Plot (A) shows PC1 and PC2 and plot (B) shows PC1 and PC3 of the Mordred physicochemical properties of the chemicals in the *R*CM database (n=264) and in the PFAS database (n=7,986).