Supporting Information

Modeling and Optimizing Performance of PVC/PVB Ultrafiltration Membranes Using Supervised Learning Approaches

Lina Chi [a,b,c*], Jie Wang [b], Tianshu Chu [b], Yingjia Qian [a], Zhenjiang Yu [a], Deyi Wu [a], Zhenjia Zhang [a], Zheng Jiang [c], James O. Leckie [b]

[a] School of Environmental Science and Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China

[b] The Center for Sustainable Development and Global Competitiveness (CSDGC), Stanford University, Stanford, CA 94305, U.S.A

[c] Faculty of Engineering and the Environment, University of Southampton, Southampton, SO17 1BJ, UK

** Correspondent author. Tel: +86 13816632156; E-mail address: lnchi@sjtu.edu.cn*

Text S. Comparison of four supervised learning algorithms

S.1 Linear regression (LR)

LR is a simple yet powerful statistical model that approximates the response as a linear combination of predictors. It estimates the corresponding coefficient for each predictor, so that the mean squared error (MSE) $\frac{1}{m}\sum_{i=1}^{m}(\hat{y}^{(i)}-y^{(i)})^2$ is minimized over the training set.

LR has some typical advantages. (1) It accepts mixed types of predictors, such as numerical, categorical[1], or even missing values[2]. (2) It is robust for predictors with different scaling ranges, or even irrelevant predictors. (3) It is interpretable regarding how the response is related to each predictor, and here, LR can be combined easily with other statistical tests and analysis. However, major disadvantage of LR is that it uses the very strong assumption that the response has a linear relationship on the predictors and an identically, independently, and normally distributed noise. Thus, our predictions could have a very high bias if the data is not distributed in this way.

S.2 Multiple additive regression tree (MART)

MART is an improved tree-based model that combines several classification and regression trees (CARTs) with a stochastic gradient boosting technology[3]. Similar to LR, CART estimates the response based on the contributions of each predictor. However, instead of assuming the response as a linear function of predictors, CART performs multiple partitions in the space of predictors. Then for each new predictor vector, its response is estimated as the mean of all responses of the training set in the same region of the predictor space[4]. However, a single CART usually has high bias due to the piecewise partitions, as well as high variance due to the greedy search strategy. Therefore, instead, we apply MART to improve the prediction accuracy.

MART has considerable advantages. (1) It can naturally handle mixed predictor types. (2) It is robust to irrelevant predictors, different scales, and outliers. (3) It is interpretable from the trained and selected decision tree. However, MART has some disadvantages, such as high bias and variance for data with low order of correlations.

S.3 Multiple additive regression tree (NN)

NN is a popular brain-analog SL model structured with a hierarchical set of logistic regressions[5]. The prediction of logistic regression is simply a sigmoid function of that of LR, so that each prediction is inside the range (0,1). An example n-3-1 NN is shown in **Fig. S1**, where *n* is the number of predictors, *a* and *b* are weights for the input and hidden units, and *S* and + denote the sigmoid function and linear combination, respectively. Formally, the predicted response is:

$$\hat{y}^{(i)} = \sum_{j=1}^{N} b_j S(a_{0j} + \sum_{k=1}^{n} a_{kj} x_k^{(i)}) + b_0, \forall\, i = 1,\ldots,m \tag{S1}$$

where *N* is the number of hidden units and *x* is the predictor array of each training data point. The NN can be represented as all weights in the network, which can be trained with an appropriate number of epochs. During each epoch, we apply stochastic gradient descent on weights over the training set[3,6].

The unique advantage of NN is that it can handle multiple correlated responses together and find complex relationships. Also, NN is more complicated than LR and MART, and we can adjust several controlling parameters, such as the number and size of hidden layers, to achieve the optimal performance. However, NN has more limits than either LR or MART. (1) It cannot accept categorical or missing predictors[7]. (2) It is a black box, and we do not know how each predictor contributes to responses from the model parameters.

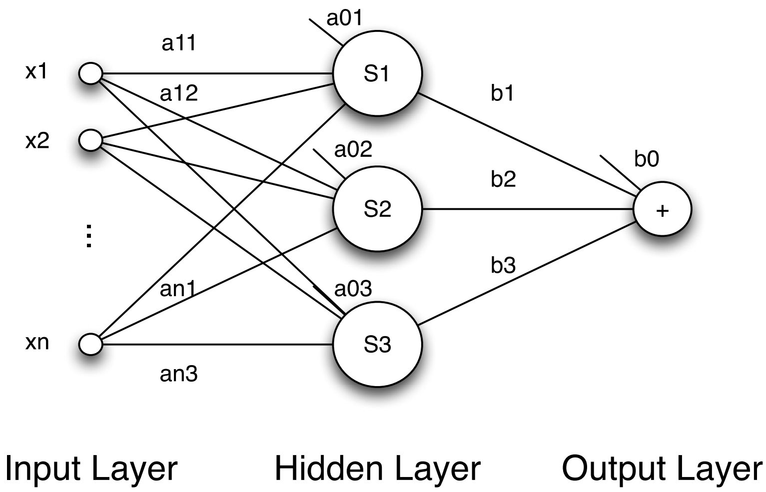

Fig. S1 Structure of an n-3-1 NN

S.4 Support vector machine (SVM)

SVM is a popular SL model using the kernel method. In particular, it replaces the predictor array with an appropriate basis function, and then to predict the response with a kernel function, indicating the similarity between the new basis function and all critical measurements (support vectors) in the training set, which can determine the decision hyper-plane in the classification problem[8]. SVM is represented as the set of all support vectors, which can be estimated by minimizing the regularized loss function, defined with a certain kernel function[9]. Vapnik et al. first introduced SVM to regression problems using the ε-insensitive loss function[10].

The advantage of SVM is that it can solve complex problems by mapping the predictors to the space with higher dimension, leading to a low bias of predictions. SVM can handle both numerical and categorical predictors, but it has some limitations, such as that it is not interpretable and that it is sensitive to irrelevant predictors and different scales.

Table S1 Fabrication process parameters and performances of the resulting PVC/PVB membranes

Sample No.	Wt% of Polymers	Wt% of PVC	Wt% of PVB	Wt% of DMAc	Wt% of Additives	Type of Additives	Temperature of Casting Solution	Evaporation Time	Temperature of Blade	Type of Coagulation Bath	Concentration of Solute in Coagulation Bath	Rejection Rate of BSA	Flux
1	15	13.5	1.5	85	0	none	60	5	60	water	0	15.4	112.4
2	15	12	3	85	0	none	60	5	60	water	0	12.6	221.52
3	15	10.5	4.5	85	0	none	60	5	60	water	0	18.3	267.67
4	15	9	6	85	0	none	60	5	60	water	0	20	308.97
5	15	7.5	7.5	85	0	none	60	5	60	water	0	22.6	356.86
6	16	14.4	1.6	84	0	none	60	5	60	water	0	35	17.8
7	16	12.8	3.2	84	0	none	60	5	60	water	0	41.6	20.43
8	16	11.2	4.8	84	0	none	60	5	60	water	0	52.4	82.3
9	16	9.6	6.4	84	0	none	60	5	60	water	0	58.5	148.54
10	16	8	8	84	0	none	60	5	60	water	0	58	151.2

11	17	15.3	1.7	83	0	none	60	5	60	water	0	61.3	5.78
12	17	13.6	3.4	83	0	none	60	5	60	water	0	62.6	13.64
13	17	11.9	5.1	83	0	none	60	5	60	water	0	65.7	82.5
14	17	10.2	6.8	83	0	none	60	5	60	water	0	70	109.4
15	17	8.5	8.5	83	0	none	60	5	60	water	0	82.4	112.7
16	18	16.2	1.8	82	0	none	60	5	60	water	0	90.8	3.86
17	18	14.4	3.6	82	0	none	60	5	60	water	0	91.6	11.67
18	18	12.6	5.4	82	0	none	60	5	60	water	0	92.1	76.57
19	18	10.8	7.2	82	0	none	60	5	60	water	0	93.8	101.22
20	18	9	9	82	0	none	60	5	60	water	0	94.3	106.73
21	13	9.1	3.9	82.5	4.5	PEG600	60	5	60	water	0	90	46.2
22	15	10.5	4.5	80.5	4.5	PEG600	60	5	60	water	0	95	114.8
23	17	11.9	5.1	78.5	4.5	PEG600	60	5	60	water	0	92.1	63.5
24	18	12.6	5.4	77.5	4.5	PEG600	60	5	60	water	0	85.2	118.1
25	20	14	6	75.5	4.5	PEG600	60	5	60	water	0	80.3	65.4
26	18	18	0	77.5	4.5	PEG600	60	5	60	water	0	100	80
27	18	14.4	3.6	77.5	4.5	PEG600	60	5	60	water	0	95.2	90.8

28	18	12.6	5.4	77.5	4.5	PEG600	60	5	60	water	0	86.7	120.6
29	18	9	9	77.5	4.5	PEG600	60	5	60	water	0	88.1	104.8
30	18	5.4	12.6	77.5	4.5	PEG600	60	5	60	water	0	58.5	196.6
31	18	1.8	16.2	77.5	4.5	PEG600	60	5	60	water	0	12.1	387.7
32	18	0	18	77.5	4.5	PEG600	60	5	60	water	0	10.3	578.4
33	18	12.6	5.4	77.5	4.5	PVPk90	60	5	60	water	0	82.4	26.5
34	18	12.6	5.4	77.5	4.5	Ca(NO3)2	60	5	60	water	0	92.5	13.3
35	18	12.6	5.4	81	1	PEG600	60	5	60	water	0	94.95	33.85
36	18	12.6	5.4	79	3	PEG600	60	5	60	water	0	99.7	46.32
37	18	12.6	5.4	77	5	PEG600	60	5	60	water	0	99.3	118.92
38	18	12.6	5.4	81	1	PVPk90	60	5	60	water	0	91.58	14.25
39	18	12.6	5.4	79	3	PVPk90	60	5	60	water	0	96.6	11.32
40	18	12.6	5.4	77	5	PVPk90	60	5	60	water	0	83.35	17.82
41	18	12.6	5.4	81	1	Ca(NO3)2	60	5	60	water	0	98.63	17.8
42	18	12.6	5.4	79	3	Ca(NO3)	60	5	60	water	0	95	117.1

43	18	12.6	5.4	77	5	2 Ca(NO3) 2	60	5	60	water	0	87	19.35
44	18	12.6	5.4	77	5	PEG600	30	5	60	water	0	95.1	30.6
45	18	12.6	5.4	77	5	PEG600	40	5	60	water	0	94.7	74.3
46	18	12.6	5.4	77	5	PEG600	50	5	60	water	0	85.4	35.7
47	18	12.6	5.4	77	5	PEG600	60	5	60	water	0	82.3	71.8
48	18	12.6	5.4	77	5	PEG600	70	5	60	water	0	87.64	82.5
49	18	12.6	5.4	77	5	PEG600	80	5	60	water	0	93.53	12.6
50	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	30	43.58	140.8
51	18	12.6	5.4	77	5	PEG600	40	30	60	DMAC	30	15.84	156.7
52	18	12.6	5.4	77	5	PEG600	40	50	60	DMAC	30	19.2	178.5
53	18	12.6	5.4	77	5	PEG600	40	70	60	DMAC	30	16.74	180.9
54	18	12.6	5.4	77	5	PEG600	40	80	60	DMAC	30	18.77	208.9
55	18	12.6	5.4	77	5	PEG600	40	120	60	DMAC	30	4.28	182.7
56	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	50	66.57	85.3
57	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	10	88.9	53.1

58	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	20	82.87	56.4
59	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	40	86.97	58.6
60	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	60	84.97	68.6
61	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	70	83.33	45.2
62	18	12.6	5.4	77	5	PEG600	40	10	60	DMAC	80	82.07	122.7
63	18	12.6	5.4	77	5	PEG600	40	10	30	water	0	89.03	78.2
64	18	12.6	5.4	77	5	PEG600	40	10	40	water	0	90.07	80.6
65	18	12.6	5.4	77	5	PEG600	40	10	50	water	0	88.66	63.5
66	18	12.6	5.4	77	5	PEG600	40	10	60	water	0	87.69	101.6
67	18	12.6	5.4	77	5	PEG600	40	10	70	water	0	87.07	68.7
68	18	12.6	5.4	77	5	PEG600	40	10	80	water	0	90.67	53.8

References for supporting information

1 J. S. Long and J. Freese, *Stata Press books*, 2006.

2 R. J. A. Little, *Journal of the American Statistical Association*, 1992, **87**, 1227–1237.

3 J. H. Friedman, *Computational Statistics & Data Analysis*, 2002, **38**, 367–378.

4 D. Steinberg and P. Colla, *The Top Ten Algorithms in Data Mining*, 2009, 179–201.

5 J. Sargolzaei, M. H. Asl and A. H. Moghaddam, *Desalination*, 2012, **284**, 92–99.

6 M. Mohri, A. Rostamizadeh and A. Talwalkar, *Foundations of Machine Learning*, Mit Press, 2012.

7 C. M. Ennett, M. Frize and C. R. Walker, *Stud Health Technol Inform*, 2001, **84**, 449–453.

8 T. Hofmann, B. Schölkopf and A. J. Smola, *Ann. Statist.*, 2008, **36**, 1171–1220.

9 K. R. Müller, S. Mika, G. Rätsch, K. Tsuda and B. Schölkopf, *IEEE Trans Neural Netw*, 2001, **12**, 181–201.

10 V. Vapnik, S. E. Golowich and A. Smola, Neural Information Processing Systems, MIT Press, Cambridge, 1997, 169-184.