Machine learning for predicting sediment particle size distributions

Galen Egan*
gegan@stanford.edu

1 Introduction

Marine sediment (i.e., mud) particles cover approximately 70% of the Earth's surface [Dutkiewicz et al., 2015], and play a critical role in a host of environmental engineering problems. Settling rates of marine particles control global carbon sequestration rates [Alonso-González et al., 2010]. Particles often act as carriers for harmful environmental contaminants, and coastline erosion predictions rely largely on knowledge of sediment properties [Nisbet and Sarofim, 1972, Corbella and Stretch, 2012]. Global warming and associated sea level rise have added new urgency to accurately predicting these processes, a task which is generally attempted with numerical sediment transport models. However, the particle settling velocity, the most critical parameter in these models, is generally unknown. Under certain assumptions, it can be related to the particle size, but the particle diameter can vary over orders of magnitude over timescales as short as seconds [Son and Hsu, 2011]. These aggregation and disaggregation processes are commonly referred to as "flocculation". As of yet, there is no general model (i.e., a model not tuned to specific field observations) available which can predict suspended sediment particle size with sufficient accuracy.

2 Related work

Historically, this problem has been approached from a semi-empirical standpoint. One common method, proposed by Winterwerp [2002], models the number density of suspended particles through the interplay of turbulent stresses, particle collision rates, and inter-particle forces. Assuming that mass is conserved, the number density can be related to the mean particle size. Variations on this model have been proposed over the years to account for additional properties such as aggregated particle yield strength [e.g. Son and Hsu, 2011]. These models, however, have not been rigorously validated against observations of particle size; they are merely tuned within large-scale transport models to match field observations of the *net* suspended sediment concentration. In this context they can perform quite well, though it is unclear if they are accurately representing the particle flocculation dynamics.

More recently, researchers have begun investigating data-driven approaches to flocculation modeling. One study in particular [Sahin et al., 2017] used a simple neural network (one hidden layer with 10 neurons) to predict suspended sediment particle size based on suspended sediment concentration, turbulent shear rate, water temperature, and salinity. Results were promising, with $r^2 = 0.81$ between the model and observations on the test set. A similar model was successfully applied to predicting flocculated particle properties in drinking water treatment systems by Oliveira et al. [2018]. Aside from these studies, there has been relatively little effort in applying machine learning techniques to the suspended sediment flocculation problem.

3 Dataset and features

The data used in this project was collected during three one-month-long field work campaigns in South San Francisco Bay. The field work took place from July–August 2018, January–February 2019, and April–May 2019. The ground-truth data for this research consists of suspended sediment particle size distributions (PSDs), collected on a Laser In-Situ Scattering and Transmissometry meter (LISST-100x, Sequoia Scientific Inc). The PSDs were measured once per hour during each field deployment, and from these measurements we can extract a median particle diameter, d_{50}, and a variance, σ^2. Along with the PSDs, we measured (at the same time and location) hydrodynamic variables such as

*This project was completed individually by the author

the near-bottom wave-induced velocity, u_b, and the mean tidal velocity, $\overline{u}$. These measurements are susceptible to instrument noise, so time-series were despiked using standard procedures [Goring and Nikora, 2002]. We also measured water chemistry variables such as salinity, S, and temperature, T.

The unique aspect of this dataset is the additional co-located measurement of *in situ* water biology variables, which have been shown to affect flocculation in laboratory studies [Mietta et al., 2009]. Specifically, we used a ECO-FL fluorometer (SeaBird Electronics Inc) to measure chlorophyll-a concentration, $chl\text{-}a$, as a proxy for organic content. We also used an ac-9 meter (WetLabs Inc), which measures light absorption and attenuation at 9 different wavelengths to estimate particle index of refraction, n_p, the algal (organic) peak in light attenuation, a_{676}/a_{650}, and the detrital (inorganic) peak in light attenuation, a_{450}/a_{676}.

Though we deployed additional platforms to measure PSDs, hydrodynamics, and water chemistry, they were not co-located with water biology measurements due to instrument limitations. Therefore, we will restrict our analysis to the measurements taken at a single instrument platform, which resulted in 1648 examples of each of the variables listed above.

4 Methods

The goal of this project is to adequately predict the median particle diameter and PSD variance, d_{50} and σ^2 respectively, as functions of the various water quality and hydrodynamic variables for which we have measurements. The measure of success for this goal will be the coefficient of determination, r^2, between the predicted and measured d_{50}, and the predicted and measured σ^2. An example PSD is shown in figure 1.

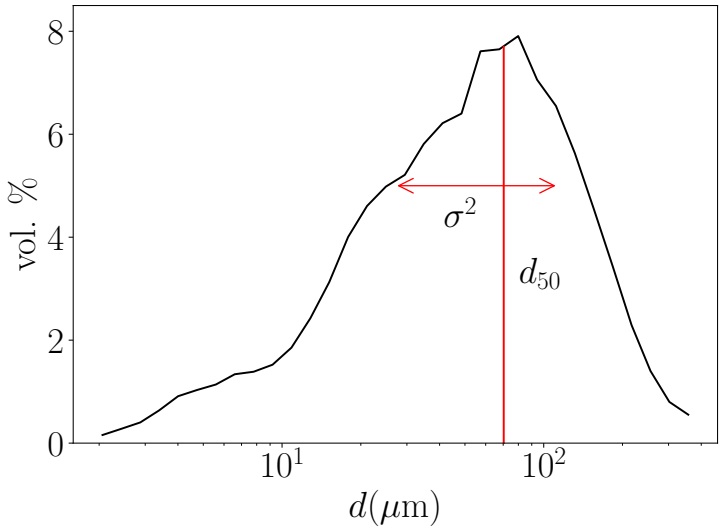

Figure 1: A sample measured particle size distribution (PSD) with the median particle diameter, d_{50}, denoted by the red vertical line, and the variance, σ^2 denoted by the red horizontal arrow.

Given that our goal is to predict a continuous variable, a regression model is most appropriate. There are numerous choices in this regard, but one relatively simple yet robust algorithm is Random Forest (RF) regression [Breiman, 2001]. RF regression works by instantiating a user-specified number of decision trees. Each of these decision trees then randomly samples (with replacement) a subset of the training data. Once trained, test data is fed to the decision trees, and the average output from all of the trees for a given test sample is taken as the model output. The RF algorithm has numerous desirable properties for our application. For example, it is relatively resistant to overfitting [Breiman, 2001], it can automatically produce an unbiased error estimate (the out-of-bag [OOB] score), and it includes a formal method for determining feature importance. Here, we will use the `scikit-learn` implementation of the RF regressor.

Once we have used RF's built-in functionality to select the most important features, we will also apply a support vector regression (SVR) algorithm to the dataset for comparative purposes. SVR works similarly to support vector machines, in that it solves a convex optimization problem to find the optimal margin classifier [Drucker et al., 1997]. Instead of classifying data points in the test set, though, SVR finds the optimal margin classifier to minimize the difference between the data points (in our case, d_{50} and σ^2) and the output of a hypothesis function, within some tolerance ε.

	FI_σ	FI_d
S	0.555	0.722
u_b	0.153	0.058
n_p	0.076	0.051
T	0.049	0.051
a_{676}/a_{650}	0.051	0.037
$chl\text{-}a$	0.050	0.031
a_{450}/a_{676}	0.037	0.025
$\overline{u}$	0.030	0.026

Table 1: Feature importance for predicting σ^2 (FI_σ) and d_{50} (FI_d) based on 1000 random forest tests run using 10 estimators and a 70/30 training/test split.

5 Results and discussion

5.1 Feature selection

Though we have 8 features available for input, they may not all be useful predictors of particle properties. From a practical standpoint, it is critical that we identify the relative feature importance, because this can inform the planning of future observational campaigns. To test which of our input features best predict d_{50} and σ^2, I ran 1000 tests using `sklearn.ensemble.RandomForestRegressor`, and averaged the feature importance vector (which is calculated by default) over the tests. The results are shown in table 1.

Clearly, the prediction is dominated in importance by the water salinity, S. The wave-orbital velocity, u_b, is more important for predicting the PSD variance than the median particle diameter, but remains the second most important predictor for d_{50} regardless. The most important biological predictor is n_p, the particle index of refraction, while the chlorophyll-a measurements and absorption spectral peak ratios (a_{676}/a_{650} and a_{450}/a_{676}) are relatively weak predictors. Therefore, the model tuning in the following section will use only the top four most important features: S, u_b, n_p, and T.

5.2 Model tuning

Now that we have selected our most important features, we can tune the model input parameters to maximize its predictive capability. For the random forest algorithm, the number of estimators (or trees) is the most critical parameter to tune. To choose an optimal number of trees, we ran tests on a training set (70% of full dataset), adding one tree to each successive test, and calculated the OOB score for each test. Because the random forest uses a "bagging" approach to train each tree, the individual trees do not train on the entire training set. The OOB score is estimated as the average predictive accuracy of all the decision trees on samples from the training set *which they did not train upon*. Therefore, it is a reliable metric for determining the model's predictive power. The OOB score is shown as a function of the number of trees used in figure 2.

The OOB score increases sharply from 10–15, and begins to level off thereafter. By approximately 32 trees, the OOB score remains flat with only minor fluctuations. Because the model becomes more expensive to run with a larger number of trees, and we see diminishing returns beyond that point, we will choose to run the model with 32 trees.

For SVR, we split the dataset into a training set (70%), cross-validation set (15%), and test set (15%). We ran the `sklearn.svm.SVR` implementation of SVR on the training set, and calculated the r^2 on the CV set for a range of C (regularization term) and ε (optimization tolerance) values. This is shown in figure 3. From this procedure, we chose $C = 2048$ and $\varepsilon = 4$, which achieve a balance between the r^2 obtained for d_{50} and the r^2 obtained for σ^2.

5.3 Results on test set

After tuning the model parameters on the training set, we can test the RF model on the remaining 30% of the data (test set) using 32 trees, and the four features that we determined most important in section 5.1. These results are shown in figure 4. The test set for SVR contains only 15% of the data because we tuned the model parameters on the CV set. These results are shown in figure 5.

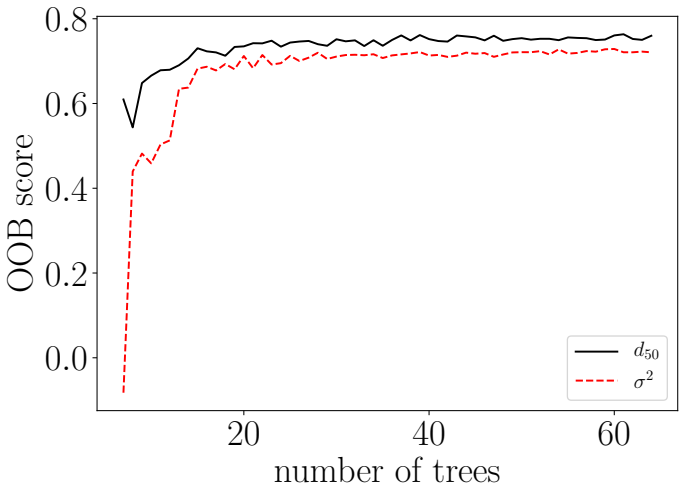

Figure 2: The OOB score as a function of number of trees in the random forest regressor, shown for predicting d_{50} (black solid line) and σ^2 (red dashed line).

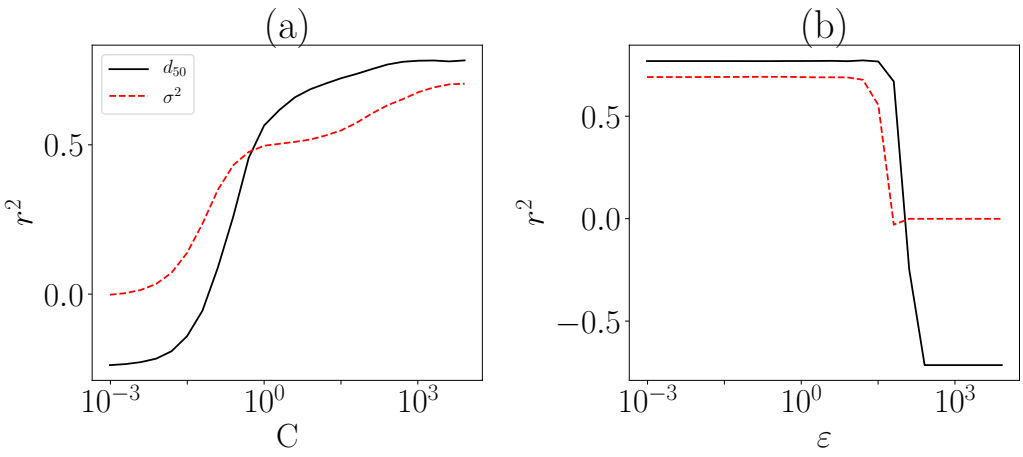

Figure 3: The coefficient of determination on the CV set when predicting d_{50} (black line) and σ^2 (red line) across a range of (a) C, and (b) ε values in the SVR model.

Both the RF and SVR algorithms show promise for predicting d_{50} and σ^2, with RF displaying slightly better results in terms of r^2 for d_{50}, and SVR performing slightly better for σ^2. The differences between algorithms, however, do not appear to be substantial: both are similarly accurate for predicting each variable, and neither tends to over- or under-predict the test set in any consistent manner. These results are encouraging, and imply that either algorithm could be successfully applied to predicting median particle diameter and PSD variance.

6 Conclusions & future work

We applied two machine learning algorithms, RF and SVR, to the problem of predicting the median diameter and variance of suspended sediment particle size distributions. Each of the algorithms achieved an accuracy of $r^2 > 0.75$ on the test sets, indicating that they can successfully represent the complex, nonlinear particle dynamics that have so far eluded many semi-empirical flocculation models.

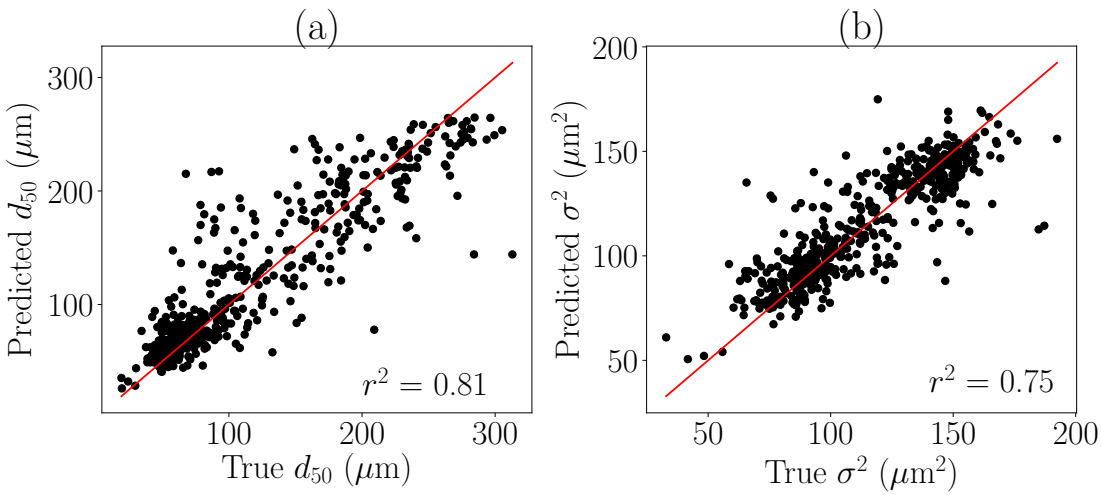

Figure 4: The random forest regression prediction compared to ground truth observations of (a) d_{50} and (b) σ^2, with the red line denoting a one-to-one fit, and r^2 indicating the coefficient of determination between the true and predicted values.

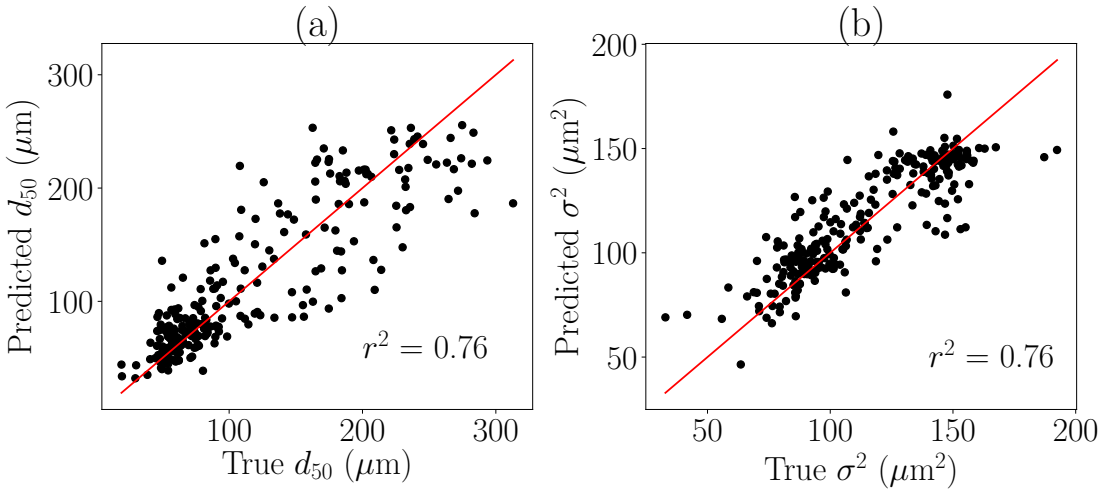

Figure 5: The SVR prediction compared to ground truth observations of (a) d_{50} and (b) σ^2, with the red line denoting a one-to-one fit, and r^2 indicating the coefficient of determination between the true and predicted values.

Building on this work, the primary question is whether or not these models are applicable to systems outside of San Francisco Bay, or even at a separate measurement location within the Bay. While it is encouraging that the two algorithms give similar results, they could each be overfitting the training data to some extent, rendering them less generalizable. To address this concern, our next step will be to incorporate additional data from San Francisco Bay and elsewhere for both testing and training of these models. With a diverse set of data upon which to train the models, we will ideally achieve a robust and generalizable prediction algorithm for use in a wide array of sediment transport models.

Code

Code used in this project can be downloaded at:
`https://stanford.box.com/s/k2z9d9z5d0xcar5667bubgz7e7vr7zwv`

References

Iván J Alonso-González, Javier Arístegui, Cindy Lee, Anna Sanchez-Vidal, Antoni Calafat, Joan Fabrés, Pablo Sangrá, Pere Masqué, Alonso Hernández-Guerra, and Verónica Benítez-Barrios. Role of slowly settling particles in the ocean carbon cycle. *Geophysical research letters*, 37(13), 2010.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Stefano Corbella and Derek D Stretch. Predicting coastal erosion trends using non-stationary statistics and process-based models. *Coastal engineering*, 70:40–49, 2012.

Harris Drucker, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161, 1997.

Adriana Dutkiewicz, R Dietmar Müller, Simon O'Callaghan, and Hjörtur Jónasson. Census of seafloor sediments in the world's ocean. *Geology*, 43(9):795–798, 2015.

Derek G Goring and Vladimir I Nikora. Despiking acoustic doppler velocimeter data. *Journal of hydraulic engineering*, 128(1):117–126, 2002.

Francesca Mietta, Claire Chassagne, Andrew J Manning, and Johan C Winterwerp. Influence of shear rate, organic matter content, ph and salinity on mud flocculation. *Ocean Dynamics*, 59(5):751–763, 2009.

Ian CT Nisbet and Adel F Sarofim. Rates and routes of transport of pcbs in the environment. *Environmental health perspectives*, 1:21–38, 1972.

Alessandra da Silva Oliveira, Verônica dos Santos Lopes, Rodrigo Braga Moruzzi, André Luiz de Oliveira, et al. Neural network for fractal dimension evolution. *Water Science and Technology*, 78(4):795–802, 2018.

Cihan Sahin, H Anil Ari Guner, Mehmet Ozturk, and Alexandru Sheremet. Floc size variability under strong turbulence: Observations and artificial neural network modeling. *Applied Ocean Research*, 68:130–141, 2017.

Minwoo Son and Tian-Jian Hsu. The effects of flocculation and bed erodibility on modeling cohesive sediment resuspension. *Journal of Geophysical Research: Oceans*, 116(C3), 2011.

Johan C Winterwerp. On the flocculation and settling velocity of estuarine mud. *Continental shelf research*, 22(9): 1339–1360, 2002.