Predicting Community Assemblage in The Rocky Intertidal of Acadia National Park

Model Evaluation Report

DSE6311 – SU2 2025

Nick Lagoni

8/19/2025

1 Hyperparameter Tuning

1.1 Random Forest (untransformed and Logit-transformed)

The hyperparameters I am tuning for my Random Forest model are the number of trees and the number of predictor candidates randomly sampled at each split. For the number of trees, I created a simple sequence beginning at 1 tree and ranging to 1000 trees, escalating each time by a value of 50. This gave me a solid range without becoming *too* computationally expensive (though I did not quantify model efficiency and certainly noticed some slow down as I increased this range). To tweak the number of predictors randomly sampled, I created another sequence, this time beginning at a value of 2 predictors (1 felt too simple) and escalating by 2 up to the total number of predictors available. I felt that I should limit the top end, but since my loop is designed to minimize the testing error, not training error, I felt that it was acceptable to leave this in for now, even if it is computationally expensive. I then looped through these sequences using a nested for() loop (perhaps I should look into hashing if I can remember how) and recorded the R^2, MSE, and RMSE value from each iteration, overwriting the last only if the MSE/RMSE values were lower. This optimized for test performance.

1.2 XGBoost

I used the same nested for loop idea for tuning my XGBoost hyperparameters, but I used a simpler grid setup, rather than the lengthy and granular ranges I built for Random Forest, as XGBoost has more hyperparameters. The hyperparameters I tuned were:

- Learning rate — I tested values from small to moderate, 0.05, 0.1, and 0.2. (high values would probably just fit to my large number of zeroes)

- Max tree depth — I kept this relatively shallow (3, 5, and 7), again to avoid overfitting to the sparse data and relatively small number of years.
- Row subsampling — I tested some moderate subsampling of 0.7 and 0.8, as well as just sampling the entire set with a value of 1. Since these values are repeated samplings of the same plots over years I thought introducing some stochasticity here might be helpful.
- Random feature sampling per tree — Again I tested 0.7, 0.8, and 1 here. I do not fully understand what the purpose of this hyperparameter is aside from introducing more randomness and trying to give other variables a "shot" over the most impactful ones, so I'm not confident in this.

2 Results

2.1 Current "Best" Model

My current best performing model, as shown in the summary stats in Table 1, is the XGBoost using untransformed data (though I did do feature engineering and time-lagging prior to modeling). Both XGBoost and the untransformed Random Forest model performed well and showed some moderate improvement when compared to baseline estimating purely via lagged terms, without any actual modeling. The logit-transformed Random Forest performed very poorly, which is contradictory to my earlier exploration on the non-lagged data. I think I may have misunderstood that transformation at some point in this process, but at this point I think I can rule it out. I also conducted several out-of-bag Random Forest models, before realizing that, without splitting off a year explicitly for testing, I couldn't actually answer the research question I am hoping to. I am fairly confident with where both my XGBoost and untransformed Random Forest models are at, as they both are producing solid R^2 values and fairly low test RMSE values, and both are showing improvements over the baseline. This suggests to me that they are not overfit and are complex enough to show some merit. I am a bit concerned about the Random Forest, as it is preferring a smaller number of trees and randomly subsampled predictors, but I suppose that is just protecting against overfitting. XGBoost is preferring a

mixed bag when it comes to hyperparameters, with moderate values across the board, which could suggest that it is reasonably well fit, or it could indicate that a more aggressive model would get easily stuck on the zero inflated data.

Table 1 - Summary table comparing Random Forest, Logit-transformed Random Forest, and XGBoost model performance over baseline predictions generated directly from the 1-year lag terms. Untransformed Random Forest and XGBoost show moderate improvements in performance over the baseline, which is already a strong predictor for the following year. Logit-transformed Random Forest (with back-transformed MSE and RMSE values) shows a dramatic decrease in performance compared to baseline. XGBoost performs the best for predicting both Ascophyllum and Fucus percent cover on future years / held out test data.

Random Forest and xgBoost vs Baseline Performance						
	R^2			RMSE		
	Model	Baseline	Difference	Model	Baseline	Difference
Untransformed Random Forest						
Ascophyllum	0.880	0.829	0.051	12.259	14.959	2.701
Fucus	0.774	0.770	0.004	17.403	18.935	1.532
Logit Transformed Random Forest						
Ascophyllum	0.746	0.779	−0.033	33.675	2.368	−31.307
Fucus	0.373	0.686	−0.312	47.751	2.825	−44.926
xgBoost						
Ascophyllum	0.885	0.829	0.056	11.719	14.959	3.241
Fucus	0.788	0.770	0.017	17.095	18.935	1.840

3 Discussion

3.1 Next Steps

My next steps are to build a more robust custom function for cross-validation in my XGBoost model. I think that building out a large and granular sequence like I had for my Random Forest may help me squeeze a bit more performance out of it. I wanted to explore linear regression models as well but I have so far been unable to transform my data to meet their assumptions, so for now the only two I have been able to dive deep on are the very forgiving models like Random Forest and XGBoost. I've explored some other gradient boosting options as well but XGBoost has been the best fit for me so far.

4 Appendix A: Data Dictionary

4.1 NETN Rocky Intertidal Long Term Monitoring Protocol Data Dictionary

VARIABLE NAME	DESCRIPTION	UNITS / SCALE	TRANSFORMATION	TYPE
LOC_NAME	Site name within Acadia National Park where survey was conducted (6 total sites)	Categorical (string)	None	Identifier
START_DATE	Date of annual survey for that site/year	Date (MM/DD/YYYY)	None	Identifier
ROCKWEED_FUCUS_SPP	Percent cover of rockweed (*Fucus* spp.)	Percent cover (%)	Logit transformed	Numeric
KNOTTED_WRACK_A_NODOSUM	Percent cover of knotted wrack (*Ascophyllum nodosum*)	Percent cover (%)	Logit transformed	Numeric
BARNACLE_E_G_S_BALANOIDES	Percent cover of barnacles (*e.g.*, *Semibalanus balanoides*)	Percent cover (%)	Logit transformed	Numeric
MUSSEL_E_G_MYTILUS_EDULIS	Percent cover of mussels (*e.g.*, *Mytilus edulis*)	Percent cover (%)	Logit transformed	Numeric
IRISH_MOSS_CHONDRUS_MASTOCARPUS	Percent cover of Irish moss (*Chondrus crispus*, *Mastocarpus stellatus*)	Percent cover (%)	Logit transformed	Numeric
KELP_E_G_LAMINARIA_ALARIA	Percent cover of kelp (*e.g.*, *Laminaria* spp., *Alaria* spp.)	Percent cover (%)	Logit transformed	Numeric
DULSE_PALMARIA_PALMATA	Percent cover of dulse (*Palmaria palmata*)	Percent cover (%)	Logit transformed	Numeric
LAVER_PORPHYRA_SPP	Percent cover of laver (*Porphyra* spp.)	Percent cover (%)	Logit transformed	Numeric
SEA_LETTUCE_ULVA_LACTUCA	Percent cover of sea lettuce (*Ulva lactuca*)	Percent cover (%)	Logit transformed	Numeric
ARTICULATED_CORALLINES	Percent cover of articulated coralline algae	Percent cover (%)	Logit transformed	Numeric
CRUSTOSE_NON_CORALINE	Percent cover of crustose non-coralline algae	Percent cover (%)	Logit transformed	Numeric
OTHER_ALGAE_GREEN	Percent cover of other green algae species	Percent cover (%)	Logit transformed	Numeric
OTHER_ALGAE_RED	Percent cover of other red algae species	Percent cover (%)	Logit transformed	Numeric
FUCUS_EPIBIONT	Percent cover of epibionts on *Fucus* spp.	Percent cover (%)	Logit transformed	Numeric
ASCOPHYLLUM_EPIBONT	Percent cover of epibionts on *Ascophyllum nodosum*	Percent cover (%)	Logit transformed	Numeric
OTHER_INVERTEBRATE	Percent cover of other sessile invertebrate species (non-listed)	Percent cover (%)	Logit transformed	Numeric
ROCK	Percent cover of bare rock substrate	Percent cover (%)	Logit transformed	Numeric
OTHER_SUBSTRATE	Percent cover of substrate types not listed elsewhere	Percent cover (%)	Logit transformed	Numeric
NOT_SAMPLED	Percent cover of quadrat area not surveyed or unable to be sampled	Percent cover (%)	Logit transformed	Numeric

COMMON_PERIWINKLE_LITTORINA_LITTOREA	Abundance of common periwinkle (*Littorina littorea*) per plot	Count	None	Numeric
SMOOTH_PERIWINKLE_LITTORINA_OBTUSATA	Abundance of smooth periwinkle (*Littorina obtusata*) per plot	Count	None	Numeric
ROUGH_PERIWINKLE_LITTORINA_SAXATILIS	Abundance of rough periwinkle (*Littorina saxatilis*) per plot	Count	None	Numeric
DOGWHELK_NUCELLA_LAPILLUS	Abundance of dogwhelk (*Nucella lapillus*) per plot	Count	None	Numeric
LIMPET_TECTURA_TESTUDINALIS	Abundance of limpet (*Tectura testudinalis*) per plot	Count	None	Numeric
COMMON_PERIWINKLE_LITTORINA_LITTOREA_MEAN_MEASURE	Mean shell length of common periwinkle (average of up to 10 individuals if abundance > 10)	Millimeters (mm)	None	Numeric
SMOOTH_PERIWINKLE_LITTORINA_OBTUSATA_MEAN_MEASURE	Mean shell length of smooth periwinkle	Millimeters (mm)	None	Numeric
ROUGH_PERIWINKLE_LITTORINA_SAXATILIS_MEAN_MEASURE	Mean shell length of rough periwinkle	Millimeters (mm)	None	Numeric
DOGWHELK_NUCELLA_LAPILLUS_MEAN_MEASURE	Mean shell length of dogwhelk	Millimeters (mm)	None	Numeric
LIMPET_TECTURA_TESTUDINALIS_MEAN_MEASURE	Mean shell length of limpet	Millimeters (mm)	None	Numeric