Predicting Dolphin Activity Using AdaBoost

Timo Pew

December 10, 2019

Sarasota Bay Dolphins

http://www.sarasotadolphin.org/intro-to-dolphin-conservation/dolphin-life/

- Dolphins of Sarasota Bay longest studied dolphin population
- Dolphins are apex predators and indicator species, studying their health gives insight to whole ecosystem
- Researchers can learn about mammal behavior and evolutionary biology by studying dolphin behavior

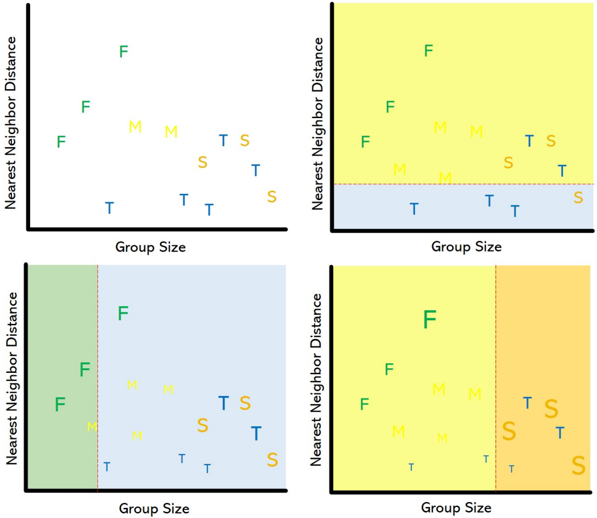
Data Collection Challenges

- Dolphin follow studies collect the water depth, habitat, group size, activity, etc. for dolphins they observe
- Lots of information to collect in short amount of time
- Activity is of particular interest, but sometimes it is difficult to tell if dolphins are feeding, so researchers mark these as 'probably feeding'

Predicting Dolphin Activity

- We hope to be able to predict dolphin activity, particularly the 194 observations recorded as ‘probably feeding’
- Our training set has 28 feeding, 674 milling, 118 socializing, and 1210 traveling observations
- We use water depth, habitat (channel, canal, gulf, pass, sandbar, or seagrass), group size, group spread, boats within 100 meters, whether or not someone was illegally close to the dolphin, and nearest neighbor distance as predictors

AdaBoost Algorithm



AdaBoost Algorithm

Let $\boldsymbol{x}_i$ denote the predictor variables of the i^{th} observation, a_i denote the activity of the dolphin at the i^{th} observation, and $T(\boldsymbol{x})$ denote our weak decision tree classifier as a function of the variables in $\boldsymbol{x}$.

Algorithm 1: SAMME Boosting

1. Initialize the weights of each observation to be $\frac{1}{n}$: $w_i^{(1)} = \frac{1}{n}$ for $i = 1, \ldots, n$
2. For b in $1, \ldots, B$
   (i) Fit a weak classifier $T^{(b)}(\boldsymbol{x})$ to the data with the weights $w_i^{(b)}$
   (ii) Compute the error of the classifier, $err^{(b)} = \sum_{i=1}^{n} w_i^{(b)} \mathbb{I}(a_i \neq T^{(b)}(\boldsymbol{x}_i))/s$, where $s = \sum_{i=1}^{n} w_i^{(b)}$
   (iii) Compute the value of the classifier, $\alpha^{(b)} = log\left(\frac{1-err^{(b)}}{err^{(b)}}\right) + log(K-1)$
   (iv) Compute new weights, $w_i^{(b+1)} = w_i^{(b)} exp\left(\alpha^{(b)} \mathbb{I}(a_i \neq T^{(b)}(\boldsymbol{x}_i))\right)$ for $i, \ldots, n$
   (v) Renormalize weights
3. The final classifier, $C(\boldsymbol{x}) = \arg\max_k \sum_{b=1}^{B} \alpha^{(b)} \mathbb{I}(k = T^{(b)}(\boldsymbol{x}))$

Prioritizing Accurate Feeding Predictions

- Traditionally, equal costs are assigned to each misclassification type and the final classifier that minimizes the misclassification error is considered the best classifier
- To reflect our desire to more accurately predict feeding dolphins correctly, we adjust the cost matrix
- We also define our own metric to 'measure' the quality of a classifier by weighting the true positive and positive predictive rates:

$$WR = (4 \cdot TP_F + TP_M + 3 \cdot TP_S + TP_T + 3 \cdot PV_F + 2 \cdot PV_M + PV_S + 2 \cdot PV_T)/17$$

- The weighted rate, WR, is between 0 and 1 where 1 would indicate perfect prediction, and 0 would indicate a classifier that misclassifies every single observation
- In some situations it may be justifiable to fix the cost matrix based on subject-expert knowledge and then use the overall misclassification rate to determine the best classification rule

Other Adjustments to the Algorithm

- We treat the cost matrix somewhat as a tuning parameter. We started with what we felt was a reasonable assignment and made changes through trial and error to produce a higher value of *WR*

	Predicted			
Truth	Feeding	Milling	Socializing	Traveling
Feeding	0.00	1.25	1.00	1.75
Milling	1.50	0.00	0.75	1.25
Socializing	1.00	1.25	0.00	1.50
Traveling	1.00	0.75	0.50	1.00

Adjusted cost matrix for decision trees

- Traditionally, the prior probabilities are set to the observed proportions, but we set them to a subject-matter expert's best guess: feeding $= 0.14$, milling $= 0.24$, socializing $= 0.10$, and traveling $= 0.52$
- Generally the decision trees consists of a single split, but we allowed that to be tuned as well

Cross-Validation Study Set-Up

- To recap, the cost matrix, number of splits, and number of weak learners can be changed
- We considered our adjusted cost matrix, a traditional equal-cost matrix; growing 10, 50, and 100 weak learning decision trees; and allowing 1, 2, or 8 splits
- For every combination of these tuning parameters we used leave-one-out and 5-fold cross-validation and compared the value of *WR* for each scenario
- This allows us to find the best values to use for our final classifier and understand how changing these different parameters affects the prediction abilities

Predicting Probably Feeding Observations

- After determining the cost matrix, number of trees, and number of splits that produce the best out-of-sample WR, we use those values to build the final classifier $C(\boldsymbol{x})$ using all the data
- We apply $C(\boldsymbol{x})$ to the probably feeding observations to predict the activity
- The estimated probability that an observation is in the k^{th} can be found with the following equation:

$$P(a_i = k) = \frac{\sum_{b=1}^{B} \alpha^{(b)} \mathbb{I}(k = T^{(b)}(\boldsymbol{x}))}{\sum_{k=1}^{K} \sum_{b=1}^{B} \alpha^{(b)} \mathbb{I}(k = T^{(b)}(\boldsymbol{x}))}$$

Results from Cross-Validation Study

		Leave-One-Out		5-Fold	
Max Splits	*B*	CM 1	CM 2	CM 1	CM 2
1	10	0.203 (0.51)	0.283 (0.63)	0.261 (0.53)	0.277 (0.62)
	50	0.203 (0.51)	0.255 (0.64)	0.231 (0.61)	0.272 (0.64)
	100	0.203 (0.51)	0.233 (0.64)	0.226 (0.62)	0.270 (0.66)
2	10	0.433 (0.59)	0.365 (0.65)	0.412 (0.57)	0.352 (0.63)
	50	0.443 (0.60)	0.364 (0.65)	0.423 (0.57)	0.347 (0.65)
	100	0.458 (0.61)	0.341 (0.65)	0.427 (0.57)	0.348 (0.66)
8	10	0.452 (0.63)	0.435 (0.65)	0.439 (0.59)	0.407 (0.63)
	50	0.465 (0.64)	0.435 (0.65)	0.466 (0.62)	0.424 (0.65)
	100	0.452 (0.64)	0.450 (0.65)	0.469 (0.62)	0.422 (0.66)

WR values from cross-validation study. Values for 5-fold cross-validation are averages over the 100 iterations. Unweighted accuracy are recorded in parenthesis

Notes from Cross-Validation Study

- The adjusted cost matrix produced better predictions overall
- Despite a single split being most commonly used, it struggled to predict feeding accurately
- We did not appear to reach the point where allowing more splits or trees produced overfitting
- Leave-one-out and 5-fold cross-validation have very comparable results
- The best combination of parameters based on this study was the adjusted cost matrix with 2 splits and 50 decision trees

Looking at Confusion Matrices from Each Cost Matrix

		Predicted			
		FE	MI	SO	TR
Truth	FE	15	3	0	10
	MI	89	86	84	415
	SO	0	3	37	78
	TR	69	16	48	1077

		Predicted			
		FE	MI	SO	TR
Truth	FE	12	3	0	13
	MI	52	140	1	481
	SO	0	21	0	97
	TR	29	12	0	1169

Confusion matrix of predictions for adjusted cost matrix on the left and standard cost matrix on the right with 50 weak learners and 2 splits

- The overall error dropped slightly, but the use of our adjusted cost matrix and *WR* mostly moved the misclassification where we were more comfortable with it

True Positive and Positive Predictive Rates for Feeding

- The highest true positive rate for feeding from the adjusted cost matrix was 15/28
- The positive predictive value for the same combination of parameters was 15/173, or just 0.087
- Although we were able to adjust the cost matrix to build a multiclass classifier that does a better job predicting feeding observations, we have to use exercise caution when predicting future feeding observations
- The results of our cross-validation study suggest that the variables currently collected do contain some information to predict dolphin behavior, but that there may be other variables that would better predict behavior, particularly feeding

Predicting Probably Feeding Observations

- Of the 194 observations recorded as probably feeding, only 40 were predicted as feeding
- There were 58 observations with a predicted probability of feeding above 0.40, only 30% of the 194
- If we had more confidence in the predictive power of our classifier, we would be able to say that the observations currently being marked as probably feeding are often not actually feeding. Unfortunately, based on our cross validation study we do not feel confident making such conclusions.

Conclusion

- We were able to adjust the AdaBoost algorithm to more accurately predict when dolphins were feeding
- However, the out-of-sample prediction accuracy for feeding was still low enough that we are not very confident in predicting future observations
- The cost matrix, number of trees, and number of splits can all be treated as tuning parameters to try and build better classifiers