Estimating Violation Risk for Fisheries Regulations

Hans Chalupsky[1], Robert DeMarco[2], Eduard H. Hovy[3], Paul B. Kantor[2], Alisa Matlin[2], Priyam Mitra[2], Birnur Ozbas[2], Fred S.Roberts[2], James Wojtowicz[2], Minge Xie[2]

[1] USC Information Sciences Institute, CA, USA
[2] Rutgers, the State University of New Jersey, NJ, USA
[3] Carnegie Mellon University, PA, USA

Abstract. The United States sets fishing regulations to sustain healthy fish populations. The overall goal of the research reported on here is to increase the efficiency of the United States Coast Guard (USCG) when boarding commercial fishing vessels to ensure compliance with those regulations. We discuss scoring rules that indicate whether a given vessel might be in violation of the regulations, depend on knowledge learned from historical data, and support the decision to board and inspect. We present a case study from work done in collaboration with USCG District 1 (HQ in Boston).

Keywords: Regulatory compliance, Coast Guard, Fisheries, Machine learning, Statistical models

1   Introduction

This paper describes a targeted risk-based approach to enforcing fisheries laws in the United States Coast Guard First District 1 (USCG D1), based in Boston, Massachusetts. The work is a joint project of the Laboratory for Port Security (based at Rutgers University) and the Command, Control and Interoperability Center for Advanced Data Analysis (CCICADA, a US nationwide consortium headed by Rutgers).

Fisheries rules and regulations have been established through a complex process whose key aims include preservation of the fisheries biomass. The primary mission of the fisheries law enforcement program is to maintain a balanced playing field among industry participants (professional fishing companies) through effective enforcement of the regulations. Over the years USCG D1 has developed an approach to fisheries

law enforcement, which among other things includes scheduling fishing vessel inspections using a scoring matrix. In this paper we describe a project aimed at validating and extending the scoring matrix by further refining the ability to determine the risk target profile of active vessels within the population of the First District.

Our research seeks a model that determines which vessels pose a higher safety risk through non-compliance with safety codes and which vessels are most likely to be contravening fishing laws and regulations. The main measure of effectiveness explored here, "boarding efficiency" (BE), is defined as the fraction of recommended boardings that yield either a fishery or a safety violation. We also formulate other measures of effectiveness and study approaches to improving them.

Currently the USCG determines whether to board a fishing vessel using a rule called OPTIDE (created by LCDR Ryan Hamel and LT Ryan Kowalske of USCG D1), which constructs a score by assigning points to known factors describing a vessel, such as the time since last boarding and the vessel's history of fisheries violations. The OPTIDE system recommends boarding if the sum of points exceeds a threshold. The developers of the method used expert opinion to select the factors in the rule, and to set their relative weights. The scoring matrix was developed using expert knowledge. This paper addresses the question: Can naïve researchers using methods of data analysis approach the effectiveness of such expert rules?

The USCG made available 11 years of data on USCG boarding activities and violations incurred by commercial fishing vessels. Our project studied introducing other features, such as weather, seasonality, fish price, fish migration, key fish species, home port, and detailed vessel history. The project team worked with economic data such as fish market prices and considered socio-economic factors such as family fishing boats in comparison to large commercial fishing vessels and fishers' attitudes toward law enforcement. We looked at the seasonal variation in boardings and outcomes. In the analysis, fisheries violations were separated from safety violations.

Machine learning methods were used to seek other features, or combinations of present and added features, that might lead to decision rules increasing the BE. In addition, alternative models for the boarding decision were considered. One model poses a choice of which boat to board, within a set of K alternatives. Section 2 describes this approach. Another approach sought regression models that derive alternative weights for the same features used in OPTIDE. This method is discussed in detail in Section 3. Section 4 discusses alternative goals, including balanced deterrence, balanced policing, and balanced maintenance of safe operations. Here we discuss alternative measures of effectiveness, e.g., violations found per hour rather than per boarding. We also discuss alternative decision strategies: random strategies; varying the number of boats used based on weather, season, or economics; alternative searching protocols to find the candidate vessels for boarding.

2   RIPTIDE: A Machine Learning Approach

In this section we describe a scoring rule, RIPTIDE, which loosely stands for Rule Induction OPTIDE. RIPTIDE extends OPTIDE by learning a more fine-grained and

data-driven prediction and ranking model from past activity data, using a machine learning approach. Using the best model found so far, RIPTIDE outperforms OPTIDE by up to 75% with regard to a specific scoring rule, described in more detail below. A software package implementing RIPTIDE can be used to experiment with the learned models, and can be applied to rank operational data.

The OPTIDE rule was built based on expert judgment and intuition. It is an abstraction of a set of features that a commanding officer will routinely consider when deciding whether to board a vessel. However, to our knowledge, there had been little or no optimization of the rule based on historical data.

To extend OPTIDE, we used a data-driven machine learning approach to learn a classification model from historic boarding activity data. RIPTIDE uses machine learning to automatically find regularities in past boarding activity data and encodes them in a model (or classifier) that can then be used to rank new, previously unseen candidate boarding opportunities. The classifier takes a single (new) data instance and applies the previously learned model to assign the new instance to one of two classes (e.g., "violation" or "no violation"). In doing so, the classifier estimates a probability that may be interpreted as the "confidence" of the prediction. This estimate is based on how well the model performed for similar cases on the training data. These probabilities can then be used to rank instances, as does the OPTIDE risk score.

Machine learning is built upon two core principles, data representation and generalization. First, every data instance is represented in a computer-understandable form. This is generally done by engineering a set of features or attribute/value pairs that carry relevant information and that can be either directly observed or computed from the data. In the generalization phase, the classifier uses many data instances for which the class is known as training data, and seeks regularities in that data that allow it to predict the class of a new data instance. There are many different data representation schemes and learning algorithms that can be used (see, e.g., [2, 5, 9] for an overview).

For RIPTIDE, we chose a learning algorithm called a boosted decision tree that is a good general-purpose tool for problems with a small to medium number of features. One advantage of decision trees is that the learned models are (large) 'if-then-else' statements that can be inspected by humans, and that are therefore to some extent understandable. This is useful for comparison to a rule-based approach such as OPTIDE, as the experts want to be able to decide whether they should trust such a model. Other learning methods such as support vector machines or neural nets produce largely if not completely opaque models, which can be judged only by their input/output behavior.

Classification performance can be improved by combining multiple classifiers that were trained using different algorithms, features, sections of the data, etc. One such strategy is called boosting. In boosting, instead of learning a single decision tree, we learn multiple trees on different subsets of the training data. An algorithm such as AdaBoost [4] (for Adaptive Boosting) then learns the "best" weights for combining the results of those individual decision trees into an overall boosted decision tree. For our currently best-performing classifier (Model 58), boosting improves performance on a boarding tradeoff task (described below) by about 25%.

Some 10,000 boarding activities from 2002 to the end of 2011 were used as training data and a set of about 1000 boardings in 2012 was used as a held-out test set to evaluate the models. To use a classifier such as RIPTIDE, one must set a threshold, which we can estimate from the training data. If the estimated probability of finding a violation is above the threshold, we recommend boarding a vessel; otherwise, not. Let TP be the number of true positives, that is, cases where the score is above threshold, and the boarding in fact found a violation; the remaining cases where the classifier says "board" are the false positives FP. Standard measures of effectiveness (MOEs) for classifiers are recall R (the percentage of vessels having some violation that are flagged for boarding), precision $P = TP/(TP+FP)$ measuring the fraction of true decisions, and their harmonic mean, known as the F1 value: $F1 = 2*P*R/(P+R)$. Picking a low probability threshold will give high recall but low precision; conversely, a high threshold will give high precision but low recall. Every choice represents a tradeoff between TP and FP, and what is acceptable depends on external factors such as task objectives and resources. Using a generic rule such as maximizing R or F1 value will generally not give the best compromise in practical applications.

The best way to compare classifiers without setting a threshold is to plot ROC (Receiver Operating Characteristic) curves. An ROC curve shows the true positive rate (or recall) plotted against the false-positive rate, that is the ratio of false predictions to the number of non-violating vessels, for each possible threshold point. The curve shows a tradeoff space showing how many more false positives one must accept to get additional true positives.

We can use the area under the ROC curve to compare different classifiers; a higher area under the curve generally means a better classifier. Figure 1 shows a comparison of ROC curves for OPTIDE and Model 58 for the held-out test data covering the year 2012. Both models have more or less identical area under the curve (AUC) of about 0.65, This shows that they are doing better than random choice (the dotted line with an AUC of 0.5), but not very much so, indicating that there is not a very strong signal in the data to begin with. Model 58 is doing significantly better at picking up the higher yield boardings (the bump at the beginning of the curve), but it loses that advantage towards lower-risk boardings. It also is much more fine-grained than OPTIDE, a feature we will explore in more detail below.

In the current formulation of OPTIDE, for values of the score, the yield distribution is very flat, which can be seen in the long straight sections of the OPTIDE ROC curve. About 84% of all boardings fall in a very narrow band of yield close to the threshold level. This means a large number of ships are apparently indistinguishable. Our analysis of the data suggests that there are no standout "red flags" that positively indicate that a ship might be in violation of some regulation. Even among vessels having the highest risk score, only one third of boardings yield a violation. This means we cannot assign a strong meaning to any of the OPTIDE risk categories.

Instead of focusing on absolute risk scores with a global interpretation, we explore an alternative MOE: How well can a model select among a small set of alternative vessels? For example, a set of ships may be encountered more or less simultaneously, calling for an informed decision as to which ships to board, given available time and resources. Technically, this calls for ranking the boats in the small candidate set.
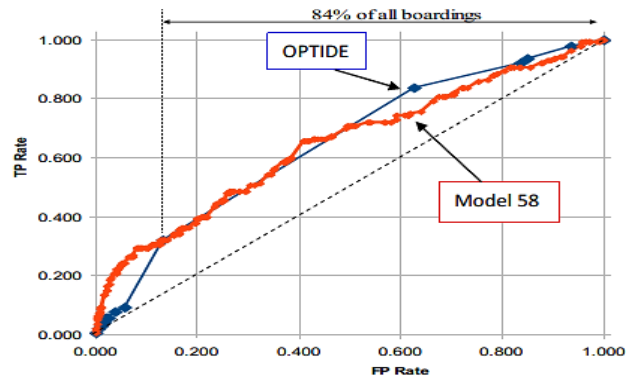
Fig. 1. ROC curves for OPTIDE and Model 58 for the held-out test data in 2012. Model 58 is a weighted combination of 20 different tree models, found using AdaBoost.

To evaluate ranking performance we consider the following MOE. Given a test set of boarding activities such as the 2012 held-out set, we randomly pick a set (or bucket) of size k and rank the elements in the bucket according to our model. We then pick the top-ranked boarding activity in the bucket (choosing randomly in case of ties) and test whether it actually had a violation or not. We repeat this experiment many times and compute the fraction of trials in which we picked a winner (i.e., a boarding with a violation). The probability of picking a winner is strongly dependent on the bucket size, since smaller buckets have a smaller chance of containing a vessel with a very high score. For example, for the held-out set of 1002 boardings of which 14% yielded a violation, the probability that a random set of two boardings contains at least one with a violation is about 26%, for 5 it is 53%, for 10 it is 78% and for 20 it is 97% (almost certain). Note that this high probability doesn't mean that it is easier to find one with a violation; that aspect still requires a good ranking function to find the best item in the bucket. Since all of our analysis is based on data collected under historical boarding policies, and, more recently, OPTIDE, the practical implications of the findings in this section remain to be explicated in future work, which our USCG partners are currently undertaking in exploration of our new ideas.

Table 1 shows the results of these experiments. It compares our currently best model, Model 58, to OPTIDE and two other models. Model 58 includes features not used in OPTIDE, such as distance to coast and vessel subtype. An alternative model (Model 57) omits a feature (distance to coast) and still a third model (Model 48) adds something called observed activity as a feature. The top of Table 1 shows standard AUC and Max-F1 metrics, and all models perform fairly similarly. In the lower portion, we show results on ranking experiments with bucket sizes ranging from 2 to 50. We find that our best model improves up to 76% over OPTIDE for a bucket size of 20, where we have an almost 45% chance to pick a winner, and even for a more realistic bucket size of 10, the improvement is still a good 38%. This shows that the apparently small advantage of RIPTIDE at higher levels of yield can become a substan-

tial improvement if it is possible to batch the candidate vessels and choose the most likely one to board.

Table 1. Evaluation results for OPTIDE and several alternate models

	Random	OPTIDE	Model 57	Model 48	Model 58	58 vs. OPTIDE
N-Thresh		15	135	191	206	
Max-F1		0.301	0.300	0.310	0.328	+9.0%
AUC		0.648	0.626	0.656	0.646	-0.3%
Bucket Size			Choose 1 of k			
5	0.135	0.210	0.217	0.236	0.243	+15.9%
10	0.135	0.237	0.279	0.311	0.328	+38.5%
15	0.135	0.244	0.328	0.364	0.393	+60.9%
20	0.135	0.251	0.363	0.403	0.443	+76.4%
25	0.134	0.261	0.399	0.440	0.484	+85.1%
30	0.135	0.276	0.422	0.466	0.516	+86.8%
35	0.135	0.290	0.447	0.488	0.542	+86.6%
40	0.134	0.307	0.464	0.505	0.567	+84.7%
50	0.137	0.336	0.492	0.542	0.601	+78.9%

We have developed a small RIPTIDE software suite that can be used to classify and rank potential boardings based on the best models found so far, and to retrain models if necessary. RIPTIDE builds upon the Weka toolkit [5] and adds a number of methods for data translation and various other tasks. RIPTIDE is purely Java based and can be run on Linux, MacOS and Windows platforms

Using the RIPTIDE approach in practice will require the users to retrain the machine learning models at regular intervals, perhaps on a yearly basis, to ensure that significant changes in behavior are incorporated. This would be an uncomplicated task, as long as the basic set of features to consider remains the same or similar. The actual implementation of RIPTIDE is experimentally underway at the USCG.

3   DE-OPTIDE

In this section, we describe an alternative approach that utilizes regression methods in statistics and the historical data to derive alternative weights for the same features used in OPTIDE. Based on this approach, a new decision rule was developed, called Data-Enhanced OPTIDE (DE-OPTIDE). We compare its performance with the original OPTIDE rule.

An underlying assumption of OPTIDE is that probability of a violation is related to an underlying score that is a weighted sum of some predictor variables $X_1, X_2, ..., X_n$ (i.e., features used in the OPTIDE rule). The decision is made to board if the score

exceeds a threshold. This assumption, plus potential random errors, leads us directly to a statistical model called a logistic regression model (see [6]). Logistic regression is an instance of a generalized linear model [1, 8]. It allows one to analyze and predict a discrete outcome (known as a response variable), such as group membership, from a set of variables (known as predictor variables) that may be continuous, discrete, dichotomous, or a mix of any of these. Generally, the response variable is dichotomous, such as presence/absence or success/failure. In our case the response variable is the violation indicator (presence/absence) of a vessel.

When sample data from such a model are available, we can perform a statistical analysis to estimate the unknown coefficients and thus estimate the relationship between the response and predictor variables. We can then use the logistic regression model to predict the category to which new individual cases are likely to belong.

We assume a violation is related to an underlying latent score S which is a weighted sum of some predictor variables plus potential errors, i.e., $S = W_1X_1 + W_2X_2 + \ldots + W_nX_n +$ error, where the *Ws* are weights describing the contributions of the feature and the random "error" follows a normal distribution with mean 0 and variance σ^2. As with the tree-based rules, if the score of a vessel exceeds a certain threshold value, the vessel should be boarded. Mathematically, these assumptions lead to the aforementioned logistic regression [3,10]. We used logistic regression and the data set available to us to estimate the coefficients W_1, W_2, …, W_n and we then used these weights to create a new decision rule. Since the new decision rule uses the same features as in the original OPTIDE rule but their weights are determined by the historical data, we call the new rule a Data-Enhanced OPTIDE (DE-OPTIDE) rule.

We note that in the original OPTIDE matrix, all of the features are categorical. Although some of them are naturally continuous, they are categorized or binned for the analysis, which may cause some loss of information. We therefore performed an additional analysis using the same set of features, but retaining continuous values for some of the features. Using the continuous versions does somewhat improve the performance of the DE-OPTIDE rule. In treating the features as continuous, we employed standard imputation techniques for missing data.

In our analysis, we randomly split the entire boarding data set available to us into two subsets: 50% used for training and 50% used for validating. We fit the logistic regression model to the training data and used the estimated probabilities to determine a new decision rule. Then we applied the new rule to the remaining 50% of data to assess its effectiveness. In the new decision rule, the threshold for boarding was chosen by either setting a required percentage of vessels to be boarded, or setting a target boarding efficiency. To control variation caused by the random 50-50 splitting, the calculations were repeated 10 times. Therefore, the results we describe do not correspond to a single unique boarding rule.

Starting with just categorical data, we explored the relationship between the Boarding Efficiency BE and the percentage of recorded boardings (that is, the fraction of all records in the data set for which boarding is recommended, at a given threshold). Results are shown in Figure 2. When applied to the data that was not used to train the model, DE-OPTIDE yields a somewhat higher or similar BE compared to OPTIDE for almost the entire range of recorded boarding percentages. For DE-OPTIDE, effi-

ciency ranges from 20% to 35%, and setting the threshold to reduce the number of boardings yields higher efficiency. This is because the rule ranks vessels by their probability of yielding violations. Therefore, when fewer are boarded, the average chance of finding a violation is higher. In choosing the threshold for the decision rule one may need to take into account not just efficiency but also the fraction of recorded boardings.
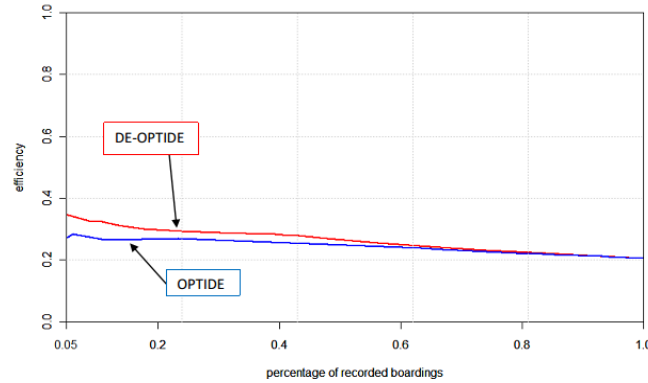

Fig. 2. Boarding Efficiency vs. percentage of recorded boardings using both OPTIDE and DE-OPTIDE for different thresholds (test data 50%). The results are based on 10 repetitions of the random selection of training data.

We also compared the efficiency of DE-OPTIDE with that of OPTIDE using another MOE. The threshold for DE-OPTIDE was chosen based on examining the efficiency of the procedure over different percentages of recorded boardings. We found that efficiency for DE-OPTIDE with a decreasing percentage of recorded boardings starts to increase when the percentage of recorded boardings is less than 10%. Thus, we chose the threshold corresponding to 10% of recorded boardings for DE-OPTIDE.

We also explored an alternative way of selecting the threshold for OPTIDE, i.e., letting threshold correspond to 10% of the recorded boardings (RBs), as we did with DE-OPTIDE. We found that the efficiency of the DE-OPTIDE procedure reaches 32%, compared to 24% efficiency of OPTIDE when using an adjusted threshold (due to our data omitting values for some of the OPTIDE features) and 27% if we use OPTIDE with threshold corresponding to 10% of RBs. We recognize that the USCG would not cut boardings to one tenth of the current level. However, some combination of this rule in a randomized or mixed strategy for boarding might be effective. Note that selecting vessels for boarding purely at random yields only 16% efficiency.

Figure 3 presents the ROC curves for both the OPTIDE and DE-OPTIDE rules. This plot helps to illustrate the performance of these two decision rules as the threshold is varied over the entire range of possible values. The ROC curve for OPTIDE has an area of 0.576 under the curve, while that for DE-OPTIDE has AUC = 0.605. Again, this indicates that the DE-OPTIDE rule is somewhat better than the OPTIDE rule. These plots are based on a single random selection of the training data. Plots from nine other repetitions are similar.
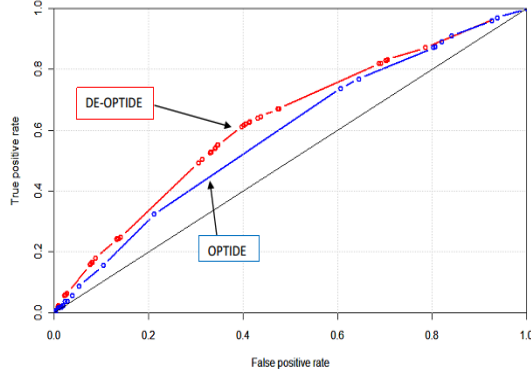
Fig. 3. The ROC curves for both the OPTIDE and DE-OPTIDE rule for various choices of thresholds (test data =50%). The plots are each based on a single run. Plots for 9 other runs show the points for DE-OPTIDE lying almost always above those for OPTIDE itself.

Next we used logistic regression treating certain features as continuous. We computed the relationship of the BE to the percentage of recorded boardings under the modified DE-OPTIDE rule using some continuous features, a rule we call DE-OPTIDE-C. DE-OPTIDE-C achieves better efficiency than OPTIDE. For OPTIDE, efficiency ranges from 20% to 30%. For DE-OPTIDE-C efficiency rises to almost 35% at levels below 10% of recorded boardings. As with the discussion of batching in Section 2, it is not known whether the set of candidates could be expanded enough for such a lower fraction of sightings to yield an acceptable number of boardings.

We also compared the efficiency of DE-OPTIDE-C to that of OPTIDE using alternative ways of setting the threshold. The efficiency of the DE-OPTIDE-C procedure reaches 34%, compared to 32% for DE-OPTIDE.

4 Other Approaches

In this section we consider other MOEs, e.g., violations per hour of enforcement activity rather than violations per boarding. We also mention alternative decision strategies: random strategies; changing the number of patrol boats based on factors such as weather, season, or economics; and varying the protocols for finding candidates for boarding.

4.1 Other Ways of Measuring Effectiveness

The models discussed so far consider all violations to be equally important. From the perspective of deterrence, this is plausible. But in terms of economic impact on fisheries and lives saved it may be more appropriate to group violations into classes $i=1,2,\ldots,I$ and seek to maximize the sum $\Sigma w_i x_i$ where x_i is the number of violations in class i. For this to be meaningful the weights must be defined on an interval or ratio scale, and not be simply ordinal [12,13].

The "denominator" in the MOE has been "boardings." Alternatively, we may want to measure effectiveness against time. Time is spent both in boarding and in seeking the next candidate. The choice of which to use will lead to different decisions. Suppose (based on the scoring rule) Vessel A has estimated 12% yield (probability a violation will be found) and the predicted time for the boarding is 4 hours. Vessel B has 15% yield and predicted boarding time 6 hours. If efficiency is violations per boarding (VPB), Vessel A has 0.12 VPB, and Vessel B has 0.15 VPB. We prefer to board Vessel B. If efficiency is violations per hour (VPH), then Vessel A has 0.12/4=0.03 VPH, and Vessel B has 0.15/6=.025 VPH. So we prefer to board Vessel A. In fact, boarding time varies randomly, according to some rule that could be estimated from data. One might also include in the denominator time spent seeking the next candidate.

4.2   Other Kinds of Enforcement Strategies

The OPTIDE-class rules discussed here are deterministic. Randomized strategies make it harder for intentional violators. The variation in goals discussed in Section 4.1 might be incorporated into a randomized mixture: e.g. 30% of time use OPTIDE, 40% of time use VPB, and 30% of time use VPH.

We can model the boarding decision as a choice between boarding and seeking further targets. For simplicity we suppose that a patrol boat meets a fishing vessel every T minutes, and must immediately decide whether to board it. That the decision to board must be made immediately is based on observations from [7] that fishermen can and do modify their behavior when they observe Coast Guard boats, seeking to limit the violations found if boarded. One boat every T minutes is a simplifying model of the random rate at which a patrol will encounter fishing vessels.

Suppose the yield p varies uniformly from 0 to 1. Suppose boarding takes time tT. What value of p should be the threshold for boarding? It can be shown that under certain assumptions, the optimal choice is

$$p = \frac{(2t + 2) - \sqrt{(2t + 2)^2 - 4t^2}}{2t}$$

As boarding time tT increases, the threshold yield p increases. This confirms the intuition that the longer boarding takes, the pickier one must be in boarding. More realistic models for $T, t,$ and the distribution of p can be developed from log data.

Finally, we considered patrol strategies, using analogies to ecology where the limiting resource is the energy available to predators [11]. In particular, we have compared pure pursuers and pure searchers. The former expend little or no energy in seeking food; they wait until sufficiently valuable prey (sufficiently risky vessel) is in sight and then act (e.g. anolis lizards). Pure searchers (e.g., warblers) spend time and energy prowling to seek food; when they sight it they decide whether to try to catch it and in that case spend little time on pursuit. We studied when a pure searcher should adopt the patient strategy of waiting for the "best" type of food (vessel with highest

risk score) or the impatient strategy of waiting for a while for the "best" type of food and then choosing what is available.

4.3   Bringing in Other Goals of Fisheries Law Enforcement

In addition to efficiency of boardings, fisheries law enforcement seeks other goals: balanced deterrence, balanced policing, and balanced maintenance of safe operations. To balance deterrence, the USCG might seek to board all vessels at least once a year. This would require, at times, boarding a low yield vessel. When should this be done? Should the rule depend on recent prior boardings? Suppose Vessel A has an estimated yield of 13% and has been boarded twice in the past year while Vessel B has a 15% yield and has been boarded six times in the past year. In some cases we might prefer to board A rather than B. We might want to board neither, and wait for some boat that has not been examined in two years.

We have developed a simple model representing a tradeoff between balance and yield. The score is based on three parameters, $y(v)$ = the yield assigned to Vessel v, $D(v)$ = days since Vessel v was last boarded, and α, a model parameter. The modified score is $S(v) = y(v) + \alpha D(v)$. The probability $y(v)$ depends on an initial class probability for that boat and on its boarding history. The class probability reflects differences that affect the probability of violation. Explicitly, we take y for a vessel with b past boardings and u "successful" past boardings to be $y = f(b,u) + .05Z$ where Z is uniformly distributed between -1 and 1, and $f(b,u)$ is presumed to come from observed data.

We ran simulations of this model, with five candidates per day, selected uniformly at random from the 100 vessels having the highest score at the start of the day. We do not simply take the five with highest scores because they might not all be accessible: the patrol might stay in a particular area and not all boats are fishing each day. Running the model 20 times for 1095 simulated days (3 years), and for each α between .0001 and .001 (incrementing α by .0001), we found the average output. A scatter plot comparing average number of observed violations over the entire 3-year period to average number of vessels boarded in the last year of the simulation can offer predictions on what the outcome might be under different scoring rubrics. Future work will consider more general scoring metrics.

5   Conclusions

Our analysis supports several conclusions. First, the existing OPTIDE approach extracts a nearly optimal rule based on the data that are used in it. The ROC curves produced by state of the art techniques for learning rules are somewhat above the curve for the existing OPTIDE rule. If the number of vessels considered could be increased, operation at a higher threshold for boarding would likely result in discovering a larger absolute number of violations per year, contributing to both fishery management and safety goals. Second, automated methods, as described in this paper, can be used to extract optimal rules by analysts who have no subject area expertise in this domain.

Indeed, such methods can find decision rules that perform as well as, or somewhat better than, models that require substantial knowledge of the data and domain expertise to develop. This means that as the USCG considers adding additional variables to the rules that trigger boardings, the automated methods used here can assess, in advance, the effectiveness of using that additional data. All that is required is to develop a data set in which the values of those new variables are reported along with the existing key variables and the results of the boarding. Finally, we have identified ways in which the objectives of the scoring rule work can be made more complex and closer to the operational realities of the USCG. Preliminary theoretical work has produced simple models showing how to include those realities in the computation of the more sophisticated yield representing complex goals of fisheries law enforcement.

We presented the results described here to USCG D1 in a briefing to the highest-level Coast Guard leadership. The results were very well received and are in the process of being implemented in USCG D1. In addition, the USCG Research and Development Center is working with D1 to explore modifications in the methods that would make them applicable to other Coast Guard districts around the country.

References

1. Agresti, A.: Categorical Data Analysis. Wiley-Interscience, New York (2002)
2. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2007)
3. Finney, D.J.: Probit Analysis (3rd edition). Cambridge University Press, Cambridge, UK, (1971)
4. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Thirteenth International Conference on Machine Learning, 148-156, San Francisco (1996)
5. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: An update. SIGKDD Explorations, Vol. 11, Issue 1 (2009)
6. Hilbe, J. M.: Logistic Regression Models. Chapman & Hall/CRC Press, London (2009)
7. King, D.M., Porter, R.D., Price, E.W.: Reassessing the value of U.S. Coast Guard at-sea fishery enforcement. Ocean Development & International Law, vol. 40, pp. 350-372. Taylor and Francis, London (2009)
8. McCullagh, P., Nelder, J.A.: Generalized Linear Models (Second Edition). Chapman and Hall, London (1989)
9. Mitchell, T.: Machine Learning. McGraw Hill, New York (1997)
10. Morgan, B.J.T.: Analysis of Quantal Response Data. Chapman and Hall, London (1992)
11. Roberts, F.S., Marcus-Roberts, H.: Efficiency of energy use in obtaining food II: Animals. In: Marcus-Roberts, H., Thompson, M. (eds.), Life Science Models, pp. 286-348. Springer-Verlag, New York (1983)
12. Roberts, F.S.: Limitations on conclusions using scales of measurement. In: Barnett, A., Pollock, S.M., Rothkopf, M.H. (eds.), Operations Research and the Public Sector, pp. 621-671. Elsevier, Amsterdam (1994)
13. Roberts, F.S.: Measurement Theory, with Applications to Decisionmaking, Utility, and the Social Sciences. Cambridge University Press, Cambridge, UK (2009)