Testing Effects of Potential Adversary Manipulations of Ship-Tracking Data

Neil C. Rowe[1][0000-0003-2612-0062] , Gabriel Lipow[1], Aroshi Ghosh[1],

and Lahari Yallapragada[1]

[1] U.S. Naval Postgraduate School, Monterey CA 9340, USA
ncrowe@nps.edu (corresponding author), gabylipow@gmail.com,
yoshigho@mit.edu, laharily@gmail.com

Abstract. This work addressed ways of making machine learning more robust against adversaries trying to manipulate its input data, as with input sensor data an adversary can at least partially control. The approach we explored was training and comparing alternative models for the same machine-learning data. Our project tested this on real ship-tracking data from the public AIS (Automated Identification System) database available from the U.S. Coast Guard for U.S. coastal waters. After filtering out stationary ships and tracking errors, we built ship tracks and tested to identify the ten most important features of tracks that help classify ship types, with the goal of identifying the types of ships not reporting or falsely reporting their identities. We then systematically perturbed the data to various degrees to see how that affected the ship-type classifications with eight standard machine-learning methods. We were particularly interested in nonlinear effects that did not uniformly decrease classification accuracy with the degree of perturbation. Such effects could be exploited by adversaries trying to deceive with their ships. Our results did find some interesting weaknesses in the methods, and our methodology is general enough to be applied to other kinds of tracking data.

This paper appeared in the International Conference on Computational Science and Computational Intelligence, Research Track on Cyber Warfare, Cyber Defense, and Cyber Security, December 2024.

Keywords: Machine Learning, Adversaries, Ships, Tracking, Classification, Testing, Perturbations

1 Introduction

The machine-learning subarea of artificial intelligence has had many successes in analysis of data. However, its methods can be impeded by an adversary who can control the data. This could occur with deception during data collection, such as when an adversary camouflages imagery, fakes electronic signals, or emplaces malware on systems to generate false information. It can also occur on when an adversary breaks into a database and changes it deliberately. Such data manipulation can cause machine learning of

false rules and trends about an adversary. For military applications, usually it is applied to classification problems in which we must decide whether we detect a threat and what kind it is, and the adversary can manipulate the detection threshold by small changes to parameters.

Countering these methods can be difficult. Neural-network models in particular are so complex mathematically that they are too hard for humans to understand, so understanding why and how an adversary manipulation works is usually impossible. Confirming the provenance of data is a challenging forensic problem due to the complexity of our digital systems. Attempts to identify the parts of a neural network that most contribute to a surprising conclusion (called "salience" of the conclusion) are sometimes possible for imagery [3], but are not reliable against other kinds of data, especially against a deceptive adversary as with security data [18]. Retraining a learned model on every possible adversary manipulation could require considerable time.

One promising approach is to run multiple machine-learning models on some suspect data and compare their conclusions. If the models have been well-trained on trusted data, disagreements they have about conclusions on new data should be rare, and large numbers of disagreements are suspicious. Despite the current tendency to make neural networks the default method for machine learning because of their often-high accuracy, other methods can have similar accuracy on most data. In particular for classification problems, the simple linear and logistic models, decision trees, and Bayesian inference can provide a baseline of performance for machine learning that can flag obvious failures of a neural network. This approach we investigate in this work.

Adversaries manipulating data are performing a kind of deception, so it is important to them that their manipulations remain undiscovered. Thus they usually perturb the data just enough to cause misclassifications, while it still looks like real data. Thus to provide a good test of effects of data manipulations, we should perturb data minimally and look for unusual effects. This is what we have done.

This research used data from ship tracking provided by the U.S. Coast Guard from their Marine Cadastre site (www.marinecadastre.gov). It holds extensive data about commercial ships and pleasure craft in U.S. coastal waters, obtained from shore-based radar and satellite data, with increasing amounts of the latter. Tracking data provides much useful data about the purposes and missions of ships, some of it quite subtle.

Section 2 describes our ship data and preliminary analysis. Section 3 describes our experiments with perturbation on this data. We follow this in Section 4 with a few conclusions. More details are in [10].

2 Obtaining Ship Tracking Data from AIS and Setting Up Analysis

The AIS ship-tracking data from the Marine Cadastre site has nearly complete records of nonmilitary ships in U.S. coastal waters over specified periods. For our experiments, we used one week of data from 12/1/21 – 12/7/21 which included records of 23,787 ships. Some data came from traditional shore-based radar tracking, and some came from transponders in the ships. Ships report this data at varying intervals: commercial

ships every few seconds, and pleasure craft every few hours. Records gave ship identifications, classifications, dimensions, timestamps, and positions; some records also gave speed, current draft, and cargo type. Ship names are not unique, so ship identification codes came from their IMO number, radio call sign, or MMSI number. If a ship had more than one, we concatenated them for the code. IMO numbers are associated with larger vessels, call signs with older vessels, and MMSI numbers with pleasure craft.

Nine major ship types occurred in our data sample: Cargo ships were the most common at around 1200, followed by tugs, pleasure craft, tankers, "other" craft such as dredges, research, and rescue vessels, fishing vessels, and passenger vessels. Two ships were incorrectly identified by AIS as military. Around 80 ships were of type "unknown", meaning no information was provided by the ship. It can occur when ship transmission fails or the ship deliberately hides information such as illegal fishing or smuggling operations [1]. For illegal activities, the data reported is itself suspect, and blockchain methods [4] will not help improve its accuracy.

Some numerical attributes had wide ranges of values. Speed is an example since many ships are in port or anchored. Applying the logarithm to these values would help, but usually we are uninterested in distinguishing low speeds for ships. Thus we applied the logistic function $f(x) = 1/(1 + e^{-s(x-c)})$ because it maps values onto to a range of 0 to 1 with most differences for the middle of the range around x = c. The logistic function is a traditional output transformation with neural networks. The parameters *s* (slope) and *c* (center) can adjust the function to the particular data needed.

Not all attributes needed the logistic function. The ratio of ship width to length was a good metric for classification of "unknown" ship types, and useful in its raw form. Another example is the fraction of time that ships are significantly moving, since most ships are usually anchored most of the time. We considered that a ship was anchored if no data was collected over at least three hours.

2.1 Qualitative Analysis of AIS Ship Tracks

For each ship, we created a track record of the timestamps, latitudes, and longitudes. For the seven days analyzed, 44,594,916 ship records were in the raw track data. Outliers did occur that were clearly erroneous data, as when a ship suddenly moved far away. To eliminate them, we compared the distances between each pair of successive points in the track records. Overall:

- 31,762,404 records (71.4%) were discarded because the ship moved less than 0.001 degrees (363 feet) from the last record.
- A few tracks were discarded if the ship never moved more than 0.001 degrees at any time (apparently ships only in port or at anchor).
- 37,678 (0.08%) were discarded because the ship moved more than 100 knots from the last record.
- 203 (0.0005%) were discarded because they duplicated the last record.
- 12,695,818 records (28.5%) were kept for track data.

Once we sorted tracks in sequence, we saw gaps where either radar could not detect the ship or the transponder did not provide location data. To fix this, we calculated the average speed along the track to check whether the ship followed the general pattern of the previous path. 8,197 ships had records but no track because the records indicated the ship did not move significantly or moved impossibly fast. We found 44,873 occurrences of possible inflection points in tracks (significant changes in speed or direction): 43,755 successive track-record pairs having a time gap of at least 3 hours after eliminating stationary points, and 1,118 showing a change in the acceleration vector of at least 0.003, where velocity was defined as a change in latitude degrees per second.

After removing errors in the data, the remaining track data was aggregated into track records and plotted, and track properties were extracted as described in **Table 1**. To create images, we used the Python library Geopandas to define approximate land regions to provide some orientation, with teal-colored dots representing major ports. We colored the track points from purple (earliest times) to yellow (latest times).

Fig. 1 shows a tugboat traveling around Puget Sound in Washington State: many routes are repeated quite precisely over the week. The blue dots are centers of major ports. Passenger vessels with scheduled routes also have distinctive patterns with many parallel paths (**Fig. 2**); this shows ferry routes from Long Beach, California, to Santa Catalina Island. Fishing vessels had the most complex routes, as they maneuvered to stay within preferred fishing grounds (**Fig. 3**). Sailing vessels also have distinctive tracks with a jagged appearance due to their tacking.

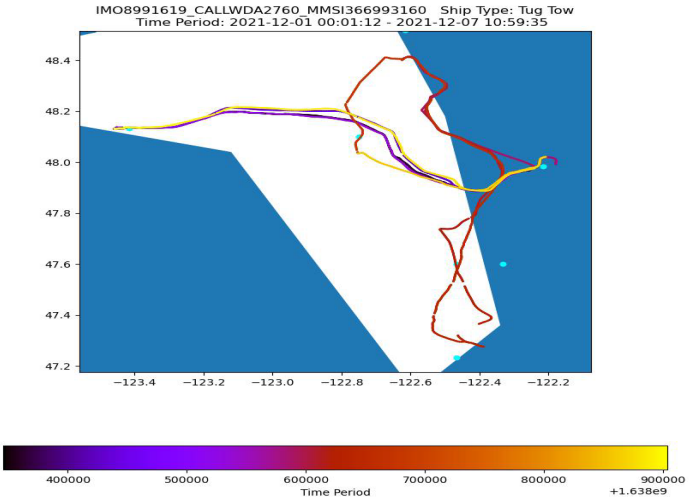

Fig. 1. Example tugboat track.

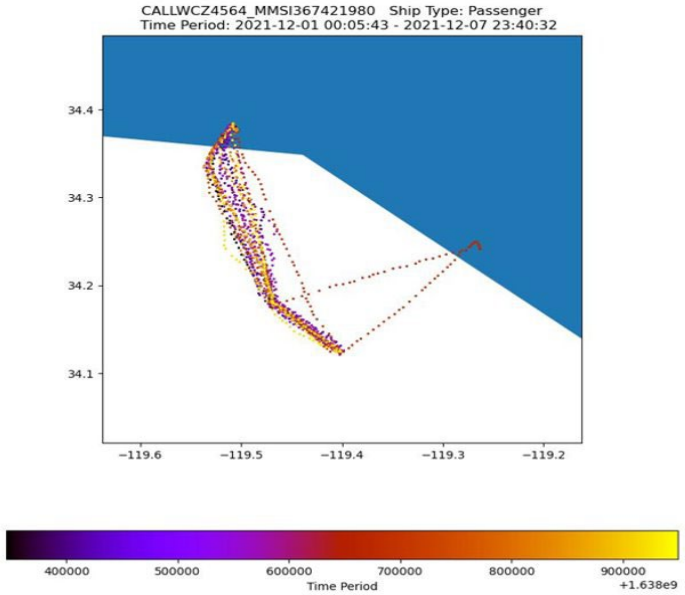

Fig. 2. Example passenger-vessel track.

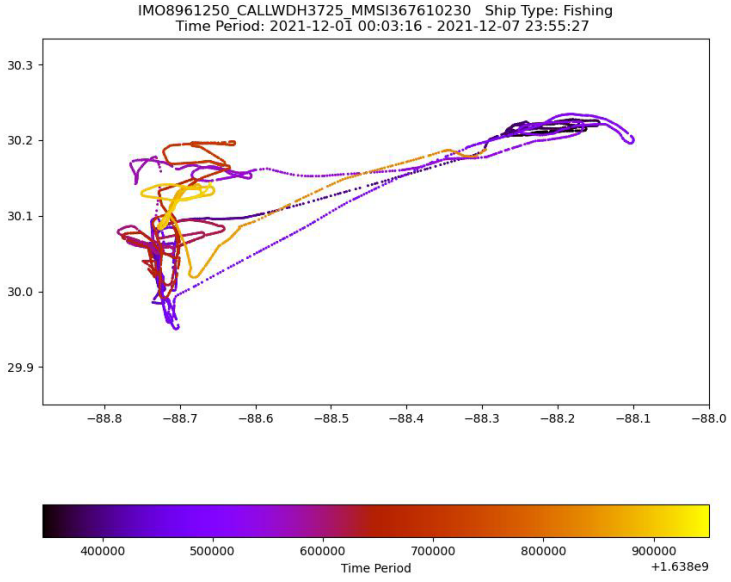

Fig. 3. Example fishing-vessel track.

2.2 Track Metadata

Preliminary analysis indicated that ship behavior and anomaly recognition are better assessed from properties of tracks rather than those of individual ship reports. We collected all the records for each ship and computed 12 properties for each track. A similar approach was used in [2] with different properties. A total 44,594,916 records were made in the seven days, which we aggregated into 15,590 nontrivial tracks. **Table 1.** shows the track attributes we chose after testing to represent the strongest phenomena in the tracks. Eight were numeric and two (the third and fourth) were strings; ship key was the primary key, and ship type was the target (independent) variable in most of our experiments. Most numeric values are modified by applying the logistic function as described. Weights were computed by a linear fit to sample training data and represent the importance of the attribute to predicting ship type.

The number of acceleration points counts the number of points in a track where a ship accelerated more than a threshold of 0.0003 nautical miles per second square. Accelerations suggest changes in plans or goals and are only typical of certain kinds of ships and certain circumstances. The mean distance to the nearest port estimates how far the ship ventures to sea. The number of straight segments was obtained by recursively splitting the path into halves until the pieces were all within a threshold of being straight according to a least-square fit.

We tested other metrics applied to the tracks but found them inadequate at predicting both ship and anomalies. Some attributes like cargo type only applied to one kind of ship. The average number of neighboring ships along a track, using the count of all ships in the same latitude/longitude bin, was intended to estimate the nearest port distance, but was unreliable because of the many ships that stay close to their home port for their assigned activities. We also studied the ratio of the distance between start and end of a track to the distance traveled along the track, which measures the straightness of the track, but found that the number of straight segments was a better predictor of ship type and track anomalousness.

2.3 Setting Up Machine Learning from Ship Data

With the attributes of **Table 1.** , machine-learning models performed significantly better at inferring ship type than with raw data. For this project, we tested several machine-learning methods, with most focus on Naïve-Bayes, logistic-regression, neural networks, and random-forest models. Good training of neural networks requires considerable time, making it harder to scale up efficiently. The best-performing method was random-forest. We suspect this was because random-forest models are good for multiclass identification problems with conflicting clues like this one. They create multiple decision trees which output predicted classes on which a majority vote is taken. This helped our task because ship types are defined by their function, not their appearance, and ships are often repurposed.

Table 1. Track attributes used in experiments for input to machine learning.

Attribute name	Data type	Parame-ters	Weight
Ship key (IMO+Call+MMSI), used only as primary data key	String		--
Ship type (primary independent variable, at least initially)	String		--
Type of ID: Has IMO, has call sign but no IMO, or has MMSI only	String		0.67
Radio receiver class of ship	String		0.59
Ratio of total track time to total time interval of recording, after removing gaps of more than 3 hours or gaps at beginning and end of the period in the cleaned data	Numeric		0.00
Logistic function of number of straight segments in the track	Numeric	Center at 5	0.05
Logistic function of average of reported speed over the ship track	Numeric	Center at 6 knots with slope 0.2	0.68
Logistic function of standard deviation of reported speed over the ship track	Numeric	Center at 2 knots with slope 0.1	0.44
Logistic function of number of points of significant acceleration, with function center at 2	Numeric	Center at 2	0.04
Logistic function of length of ship, with function center at 30 meters	Numeric	Center at 30 meters	1.16
Ratio of width of ship to length of ship	Numeric		0.89
Logistic function of mean distance to nearest port over all locations in the track, with function center at 0.1 latitude degrees	Numeric	Center at 0.1 latitude degrees	0.29

Based on the AIS vessel types and group codes provided by the Marine Cadastre Project, over 1,000 different ship types occur in the AIS data. These can be grouped into eight categories: cargo, fishing, military, passenger, pleasure craft / sailing, tanker, tug tow, and "other." 20% of the ships in order random sample did not identify their type in their AIS transponder data.

Distinguishing cargo ships and tankers that refuse to identify themselves was difficult because these two ships have similar sizes and behave similarly as transport vessels. After combining the cargo and tanker classes, our accuracy did increase from 83.1% to 93.9%. However, we can still differentiate cargos and tankers based on the type of freight they carry: Tankers transport oils, liquids, and gases, while cargo ships carry regular freight. However, we did not use the cargo attribute in the experiments reported in Section 3 since cargo was not consistently reported.

The ships labeled as “other” were difficult to identify because they showed considerable variation. Because of this, we did most of our experiments without the “other” ship type. Doing this, we obtained weights shown in **Table 1.** from a linear fit to the attributes to see which were providing the most information for the model. Ship size was the most important, followed by mean speed. This was expected as most shipping vessels are big and faster, whereas pleasure craft and fishing vessels are smaller and slower.

Besides predicting ship type, we created a model to predict the receiver class and the type of ID (IMO, CALL, or MMSI). With a random-forest model of the receiver class, we got 96.9% accuracy and for the ship’s ID type, 83.7%. This helped verify that the data was accurate and to possibly identify ships providing false data.

We also looked for coincidences between ships, defined as a pair of ships that have visited the same location at least once during a day [13]. This requires comparing locations of every pair of ships for each day, a time-consuming process. To simplify it, we created a list of geographical bins a ship visited each day and then compared the bins between ships. This required checking 60,762,025 unique ship pairs out of 15,590 ships for our data sample, a large but not unreasonable number. Using a granularity of 1.0 degree when creating the bins, we found 60,441,464 pairs with no coincidences, 239,272 pairs with one coincidence, 53,073 pairs with 2 coincidences, 16,268 pairs with 3 coincidences, and 6,913 pairs with 4 coincidences. By ignoring ports and reducing the granularity to 0.1 or 0.01 or comparing every hour instead of every day, we could identify ships possibly engaged in activities like smuggling cargo. We could further reduce the time taken by a bisection search when comparing ship bins.

3 Effects of Data Manipulations on Machine Learning for Ship Classification

3.1 Introduction

National security has long depended upon quickly determining the identity of a ship, friendly or otherwise. Using machine-learning models to classify ships has been done with sensor data like analysis of satellite imagery [17] and sonar signatures [8]. In recent years, data produced by AIS enables better classification of ships, provided it is reported honestly. Other research has tested “fuzzy” rules for ship classifiers, and in particular in recognizing small motorboats used by drug traffickers to circumvent patrols [12]. Other methods have been used to detect illegal fishing by ships that have deliberately turned off their transponders [16].

However, classifiers can misclassify and misrecognize targets if their input data is manipulated by an adversary. Such manipulations can be damaging when undetected, which becomes an increasing threat with increasingly complex classification models, due to the sensitive tuning of such models and the comparative simplicity in modifying and distorting the data a model is to classify. Deception is easiest to achieve when much of the information about the operation of the software is unclear to operators [11]. Note

we address deception in the raw data when self-reported, not just the training data, though changing just the training data is an easier way to deceive if possible.

Simpler models should simplify identification of data manipulation for classification infrastructure and make the models themselves more robust against attacks. To that end, we tested nine relatively simple models to determine what effect small perturbations had on classification; perturbations are reported to be the best clue to adversarial data manipulation [7]. Adversaries manipulating our data do not want to be discovered, so their manipulations tend to be small. We tested the degree to which small perturbations to ship attributes could significantly affect the accuracy of the classification of ships.

Testing used WEKA (https://www.weka.io) to develop and test nine classification models The models were Bayes net, decision tables, J48 decision trees, JRip, IBk, logistic neural network as used for ship classification in [5], naïve Bayes as used in [15], random forest, and simple logistic. They were trained on metadata created as described in Section 2, and then tested against data based on the original training data but with random perturbations to data elements to fool the model into misclassifying ships.

3.2 Methodology

We experimented with the final state of the data generated as described in Section 2, including the weights shown in **Table 1.** . Some experimentation was needed to find the best parameters for the applications of the logistic function to the numeric data so that each parameter had approximately an equal variability, since this tends to speed learning.

After preliminary tests, we realized that some of our original data had errors because it gave impossible or inconsistent values for the attributes. This can happen because of self-reporting in incorrect formats. A Python program was written to remove the faulty data. Approximately 500 ships reporting impossible data were removed from the testing.

A Python program was written to perturb values of eight attributes of the metadata and observe the effects. The attributes perturbed were rows 5 through 12 of **Table 1.** . (Note these include aggregates of track data; an adversary would need to make consistent changes over a track to change the aggregates without creating suspicion.) Following this, a Java program tested each perturbed dataset with the WEKA interface, and then output a confusion matrix for each test, which was then summarized to get an average for every column. Finally, a Java program created a series of bar charts for easy comparison. Graphs were created of the F_1 measure to represent accuracy, defined by $2PR/(P+R)$ where P represents precision and R represents recall.

3.3 Effects of Perturbations

The first experiment used scaled random perturbations of the numeric attributes according to the formulae $X' = X + k(X - M)$ and $X' = X + k(X - m)$, where X is the value to be perturbed, X' is the new value, M is the maximum of the attribute range, m is the minimum of the attribute range, and k is a random number between 0 and 1. After this

testing, we decided that random perturbations were not informational enough to give useful results.

Perturbations were then modified to increment k uniformly by 0.1 from 0 to 1 and from -1 to 0. This enabled us to better locate unusual effects on attributes. Also, it was decided that "unknown" ship types should be added the dataset in these experiments, since many of those ships had a strongly preferred classification already by our trained model which we could assign.

We looked for nonlinear effects of perturbations where the effect was not consistently the same ratio of the degree of the perturbation. In other words, situations where a Taylor-series approximation of the effect of a small perturbation fails significantly. Such situations were rare in our data, but could be quite strong; often they were associated with qualitative changes in the tracks such as elimination of a turn. In some cases, we actually saw an increase in accuracy with degree of perturbation. These are situations that an adversary with something to hide could exploit.

Example nonlinear effects are shown in **Fig. 4** and **Fig. 5**. Expected linear effects should be symmetrical about the center of the plot at 0.0, and monotonically decreasing on both sides. More results for other attributes and machine-learning methods are shown in [10].

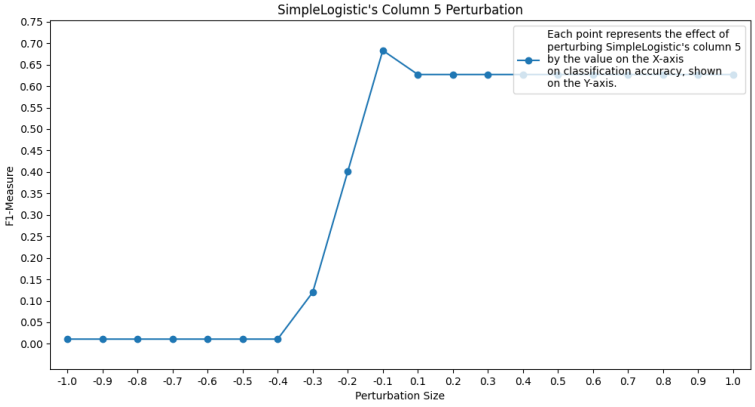

Fig. 4. Perturbation of column 5 for simple logistic models.

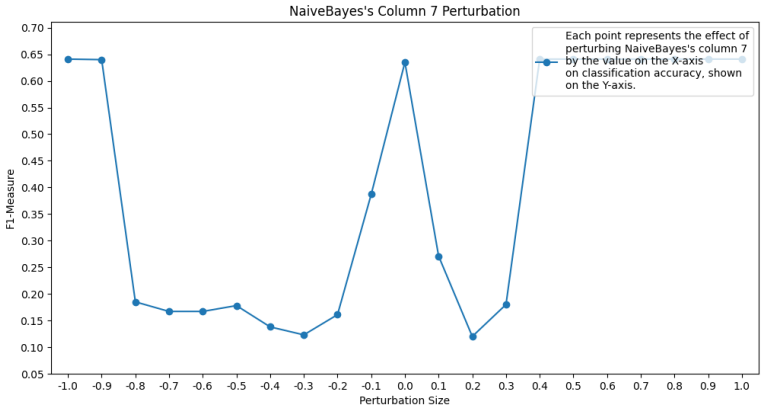

Fig. 5. Perturbation of column 7 for Naïve-Bayes machine learning

3.4 Multi-Column Perturbation

We also studied the effect of two simultaneous perturbations on different attributes since an attack victim would likely find these harder to diagnose. **Fig. 6** shows the varying effects of the machine-learning algorithms when simultaneously perturbing the ratio of width to length (the second row from the bottom of **Table 1.**) and the distance of the nearest port (the last row). (Ignore the incorrect legends on the figures.) These sometimes surprising effects could be exploited by an adversary. However in general, two perturbations were undesirable, since the effect of the second perturbation rarely had the same strength of a good first one and could undo the first effect.

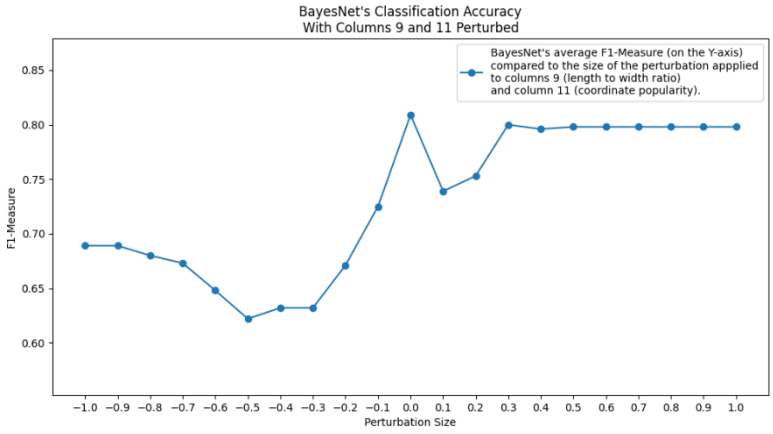

Fig. 6. Effects of a double perturbation on a Bayes Net machine-learning model

4 Conclusions

4.1 Discussion of the Perturbation Tests

Our results showed significant differences between not only classification accuracy because of perturbation, but also in what characteristics each model considers more important, with coordinate popularity, how much the ship moves between transponder reports), and the ratio of width to length being the most important overall. However, some additional trends were visible. Three ship types were hard to classify: passenger types, "unknown" types, and "other" types. We did address the "unknown" ships by assigning their most likely class for the final tests.

Some models clearly were more resistant to perturbation than others, with Bayes network and the decision tables being more robust, although decision tables were still vulnerable to perturbations of mean distance to nearest port, with a nearly 90% drop in accuracy in outliers. Random-forest and IBk (nearest-neighbor) models were little affected by random perturbations, and most severely affected were logistic, naïve Bayes, and logistic-regression models. The J48 decision tree ranked average for these tests.

The second round of tests with evenly spaced perturbations gave us a better idea about the sensitivity of each model, with more sensitive models being more easily perturbed than insensitive models. It was here we saw unusual tendencies in our data, with some tests showing that classification accuracy jumped at some points, sometimes even above the baseline accuracy without any data perturbation. This was especially noticeable in the Bayesian models (net and naïve). Other unusual results included large decreases in accuracy for the simple logistic and normal logistic models, and our inability to cause the random-forest model to significantly misclassify ships at all, since we could only cause a drop in accuracy from 100% to 99.7% accuracy with large negative perturbations. Even more unusually, in many models, the importance of each attribute seemed to change with every perturbation in a nonlinear way. That is, the performance of the naïve Bayes model with a perturbation of -1.0 was significantly worse (about 70% worse) than with a perturbation of 1.0.

Most interesting were the nonlinear perturbation effects we observed, in which either classification accuracy dropped inconsistently and asynchronously with increasing perturbations, or less often, where classification accuracy actually rose with increasing perturbations. The first phenomenon could be caused by a diminishing return on accuracy. If many formerly correct classifications are made incorrect by a small perturbation, then each additional perturbation causes significantly fewer ships to move from being classified correctly to being classified incorrectly. Different models had different tipping points in accuracy.

4.2 General Conclusions

We have shown that machine-learning methods usually work well for identifying ship types in real data of U.S. coastal waters. This will help when ships fail to provide information about their ship type because of faulty communications or deliberate attempts to deceive (as with smuggling, illegal fishing, illegal weapons transfers [14], and

military reconnaissance). An open-source tool, WEKA, made it straightforward to compare nine methods.

A key part of our work was testing the effects of perturbations on real data to see how it affected classification accuracy. Though most perturbations had a linear and symmetric effect with the size of the perturbation, some had a variety of interesting nonlinear effects. These represent possible threats to machine-learning methods that a military adversary could exploit. We itemized the effects of perturbations on each key attribute of ship tracks in [10]. All models were vulnerable to sufficiently large perturbations in a way that greatly reduced classification accuracy. However, large perturbations are easily noticeable on data inspection and will not fool anyone.

We did not find any major vulnerabilities in the machine-learning methods we tested in regard to perturbations of their input data. We thus confirm their effectiveness in future work in analysis of tracking data. Nonetheless, future work needs to investigate further the sensitivities we found in some of the machine-learning methods and analyze further how an adversary might exploit them. It also should be expanded to cover some of the more complex neural-network methods popular today, especially those that are slow and difficult to train although very accurate [6]. More data should be collected, especially outside of U.S. coastal waters, to allow testing on the rare but important kinds of suspicious maritime activity. The methods have already been extended to aircraft tracking with the data of the ATS-B system [9] where automated analysis could be more valuable because of the greater speed of the vehicles.

Acknowledgments. This research was supported by funding from the Naval Postgraduate School, Naval Research Program (PE 0605853N/2098).

Disclosure of Interests. The authors have no competing interests.

References

1. Agrafiotis, P., Doulamis, A., Doulamis, N., & Georgopoulos, A.: Multi-sensor target detection and tracking system for sea ground borders surveillance. In: Proceedings of the 7th International Conference on Pervasive Technologies Related to Assistive Environments, PETRA, pp. 1-7. https://hdl.handle.net/20.500.14279/14295 (May 2014)
2. Anneken, M., Jousselmez, A.-L., Roberty, S., & Beyerery, J.: Synthetic trajectory extraction for maritime anomaly detection. In: 2018 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 1048-1053. 10.1109/CSCI46756.2018.00204 (December 2018)
3. Hu, G., Dixit, C., Luong, D., Gao, Q., & Cheng, L.: Salience guided pooling in deep convolutional networks. In: 2019 IEEE International Conference on Image Processing (ICIP), pp. 360–364. https://doi.org/10.1109/ICIP.2019.8802973 (2019)
4. Hu, Q., Han, W., & Zhang, H.: Ship identity authentication security model based on blockchain. In: 4th International Conference on Data Science and Information Technology, pp. 135–142. https://doi.org/10.1145/3478905.3478933 (July 2021)
5. Kosta, L., Irvine, J., Seaman, L., & Xi, H.: Many-target, many-sensor ship tracking and classification. In: 2019 IEEE High Performance Extreme Computing Conference (HPEC), pp. 1-7. 10.1109/HPEC.2019.8916332 (2019)

6. LeClerc, M., Tharmarasa, R., Forea, M., Boury-Brisset, A. C., Kirubarajan, T., & Duclos-Hindie, N.: Ship classification using deep learning techniques for maritime target tracking. In: 2018 21st International Conference on Information Fusion (FUSION), pp. 737-744 (2018)
7. Li, L., Chen, X., Bi, Z., Xie, X., Deng, S., Zhang, N., Tan, C., Chen, M., & Chen, H.: Normal vs. adversarial: Salience-based analysis of adversarial samples for relation extraction. In: Proceedings of the 10th International Joint Conference on Knowledge Graphs, pp. 115-120. https://doi.org/10.1145/3502223.3502237 (December 2021)
8. Meir, T., M., Sutin, A., & Salloum, H.: Decision learning algorithm for acoustic vessel classification. Homeland Security Affairs, Vol. 4, No. 3 (April 2012)
9. Rowe, N., Allen, B., Zhou, J., Flores, A., & Das, A.: Distributed combat identification of interesting aircraft. In: International Command and Control Research and Technology Symposium, Pensacola, FL. https://faculty.nps.edu/ncrowe/ncrowe_iccrts18_distributed_combat_id_revised.htm (November 2018)
10. Rowe, N., Lipow, G., Ghosh, A., & Yallapragada, L.: Testing effects of potential adversary manipulations of ship-tracking data on success of machine learning of ship types. NPS Technical Report NPS-CS-23-005 (December 2023)
11. Rowe, N. & Rrushi, J.: Introduction to cyberdeception. Springer, Chaum Switzerland (2016)
12. Sierra, E., & Contreras J.: Classification of small boats using fuzzy classifier. In: 2015 Annual Conference of the North American Fuzzy Information Processing Society, Redmond, WA, US (August 2015)
13. Sollish, A., Everton, S., Rowe, N., & Porter, N.: Finding correlations between ship positions. In: International Command and Control Research and Technology Symposium, Pensacola, FL. https://faculty.nps.edu/ncrowe/oldstudents/iccrts18_sollish.htm (November 2018)
14. Wallace, T., & Mesko, F.: The Odessa Network: Mapping Facilitators of Russian and Ukrainian Arms Transfers. Technical report, C4ADS. Retrieved October 28, 2023, from https://c4ads.org/reports/the-odessa-network/ (September 2013)
15. Wang, W., Chu, X., Jiang, Z., & Liu, L.: Classification of ship trajectories by using naïve Bayesian algorithm. In: 5th International Conference on Transportation Information and Safety (ICTIS), pp. 466-470. 10.1109/ICTIS.2019.8883562 (July 2019)
16. Welch, H., Clavelle, T., White, T., Cimino, M., Van Osdel, J., Hochberg, T., Kroodsma, D., & Hazen E.: Hot spots of unseen fishing vessels. *Science Advances*, Vol. 8, No. 44 (November 2, 2022)
17. Yan, Z., Song, X., & Yang, L.: Research on AIS data aided ship classification in spaceborne SAR images. In: Proceedings of the 2022 11th International Conference on Computing and Pattern Recognition,179-185. https://doi.org/10.1145/3581807.3581833 (November 2022)
18. Zhou, A., Liu, X., Ye, D., & Zhu, T.: Adversarial attacks and defense in deep learning: From a perspective of cybersecurity. ACM Computing Surveys, Vol. 55, No. 8, Article 163 (December 2022)