

Quasi-Operational Testing of Real-Time Storm-Longevity Prediction via Machine Learning

AMY MCGOVERN

University of Oklahoma, Norman, Oklahoma

CHRISTOPHER D. KARSTENS

National Oceanic and Atmospheric Administration, National Weather Service, Storm Prediction Center, Norman, Oklahoma

TRAVIS SMITH

Cooperative Institute for Mesoscale Meteorological Studies, National Severe Storms Laboratory, and University of Oklahoma, Norman, Oklahoma

RYAN LAGERQUIST

University of Oklahoma, and Cooperative Institute for Mesoscale Meteorological Studies, Norman, Oklahoma

(Manuscript received 30 August 2018, in final form 17 July 2019)

ABSTRACT


Real-time prediction of storm longevity is a critical challenge for National Weather Service (NWS) forecasters. These predictions can guide forecasters when they issue warnings and implicitly inform them about the potential severity of a storm. This paper presents a machine-learning (ML) system that was used for real-time prediction of storm longevity in the Probabilistic Hazard Information (PHI) tool, making it a Research-to-Operations (R2O) project. Currently, PHI provides forecasters with real-time storm variables and severity predictions from the ProbSevere system, but these predictions do not include storm longevity. We specifically designed our system to be tested in PHI during the 2016 and 2017 Hazardous Weather Testbed (HWT) experiments, which are a quasi-operational naturalistic environment. We considered three ML methods that have proven in prior work to be strong predictors for many weather prediction tasks: elastic nets, random forests, and gradient-boosted regression trees. We present experiments comparing the three ML methods with different types of input data, discuss trade-offs between forecast quality and requirements for real-time deployment, and present both subjective (human-based) and objective evaluation of real-time deployment in the HWT. Results demonstrate that the ML system has lower error than human forecasters, which suggests that it could be used to guide future storm-based warnings, enabling forecasters to focus on other aspects of the warning system.

1. Introduction

Accurately predicting storm longevity in real time is a critical task for National Weather Service (NWS) forecasters in a variety of situations. When issuing a warning, forecasters can use longevity predictions to guide the

spatial and temporal extent of the warning. Likewise, forecasters in an outbreak situation can use longevity predictions to help focus their attention on the storms likely to last longer. Longevity prediction is also an important task for air travel, as convection closes access points to the airport, leading to long delays in the system as planes are delayed or rerouted (MacKee et al. 1999). Improved longevity prediction at airports would have economic benefit for both airlines and customers.

The NWS project called Forecasting a Continuum of Environmental Threats (FACETS; Rothfus et al. 2014,

 Denotes content that is immediately available upon publication as open access.

Corresponding author: Amy McGovern, amcgovern@ou.edu

DOI: 10.1175/WAF-D-18-0141.1

© 2019 American Meteorological Society. For information regarding reuse of this content and general copyright information, consult the [AMS Copyright Policy](https://www.ametsoc.org/PUBSReuseLicenses) (www.ametsoc.org/PUBSReuseLicenses).

2018) will change the current watch and warning system in two ways. First, it will evolve from deterministic to probabilistic forecasts; second, it will ensure a smooth transition of forecast information across spatiotemporal scales. The Probabilistic Hazard Information tool (PHI; Karstens et al. 2015, 2017, 2018) is a key part of this new paradigm, focusing on storm-based prediction. The system we propose was tested in PHI during the National Oceanic and Atmospheric Administration (NOAA) Hazardous Weather Testbed (HWT; Clark et al. 2012; Gallo et al. 2017), which is a quasi-operational naturalistic environment.

Thus, the goal of this research is to produce a longevity-prediction system that improves upon the existing predictions within PHI, using data and resources available in real time. This goal changes the approach from a purely research-based (hindcasting) system to a quasi-operational system, which adds constraints. For example, in an operational or hindcasting system, additional and higher-quality data would likely be available to improve the predictions. However, we do not study this effect, since the goal is to produce a quasi-operational system.

Most prior research on forecasting storm longevity consists of modeling studies, though some have been case studies of individual storms or outbreaks. In all cases, it has proven to be a challenging task. Within the modeling domain, it has been demonstrated that small differences in soundings can produce vastly different storm longevity. For example, Elmore et al. (2002) show similar soundings that produce storms with very different longevity, as well as very different soundings that produce storms with similar longevity. Brooks (1992) and Lilly (1990) show differences in longevity with small changes in the thermal bubble used to initiate the simulations. While not studying longevity, Dahl (2014) demonstrates that storm-scale vortices are highly sensitive to small perturbations in the initial conditions of a numerical simulation.

Other modeling studies have examined the influences of different environmental processes on storm longevity. Thorpe et al. (1982) demonstrate that strong low-level shear is needed to create long-lived storms. Rotunno et al. (1988) demonstrate that long-lived squall lines are dependent on the interaction of low-level shear and the surface cold pool. Weisman and Klemp (1982, 1986) demonstrate that wind shear and buoyancy are critical to both storm mode and longevity. Houston and Wilhelmson (2011) numerically study the issue of storm longevity in a low-shear environment and demonstrate that a deep cold pool is crucial for sustaining long-lived storms in low-shear environments. Parker (2007) shows similar

results in a moderate-shear environment. Shear is one of the environmental parameters available to our ML algorithm. Cintineo and Stensrud (2013) examine the predictability of supercells in simulation, under a variety of different initial conditions. They do not specifically examine lifetime except to note that supercells have predictable lifetimes of around 90 min. Many other storm features are extremely sensitive to the initial conditions and cannot be predicted beyond 2 h in advance.

Although fewer in number, there have been some observational studies of storm longevity. Bunkers et al. (2002) study long-lived supercells, specifically focusing on storms that last longer than four hours. They show that high wind shear and isolation from other convection are crucial for such long-lived storms. Wilson and Meigenhardt (1997) focus on storms near Cape Canaveral, Florida, examining the relationship between wind shear and the convergence zone that often causes Florida storms.

Another approach to forecasting longevity comes from algorithms such as the Thunderstorm Identification, Tracking, Analysis, and Nowcasting system (TITAN; Dixon and Wiener 1993; Li et al. 2012; Wolfson et al. 1994). Both Li et al. (2012) and Wolfson et al. (1994) use machine learning (ML) to automate part of the tracking and identification process, but neither uses ML to predict storm longevity.

MacKeen et al. (1999) is most related to our work, as they use linear regression to predict storm longevity. Their database consists of 879 storms—some single cells and some multicell clusters—near Memphis, Tennessee. Specifically, they apply both univariate and multivariate linear regression to radar- and sounding-derived variables. They demonstrate that automating the prediction of storm longevity is difficult, because there is no clear correlation between storm longevity and any set of radar- or sounding-derived variables.

This work is unique in several aspects. First, the data comprise multiple years of observations across the full continental United States (CONUS). Previous observational studies have focused on specific storms or regions of the United States. Second, the predictions were used in a real-time system: the PHI Experiment (Karstens et al. 2015, 2017), which was part of the NOAA HWT, in spring 2016 and 2017. Third, the predictions are generated by ML. Note that we have a very preliminary version of this work in McGovern et al. (2017). This paper represents a significant extension, in both sophistication of the methods and subjective and objective analysis of the results. Also, this paper adds HWT 2017 to the testing data.

TABLE 1. Training data for machine learning. All predictors are from ProbSevere. For a discussion of the ProbSevere system and “postprocessing,” see section 2a. Area, radius, and perimeter are computed from the ProbSevere polygons. AGL = above ground level.

Predictor	Units	Explanation	Postprocessed in HWT 2017?
MUCAPE	J kg^{-1}	Most unstable CAPE. CAPE is computed for parcels lifted adiabatically from all vertical levels (surface, 1000, 975 hPa, etc.), and this is the greatest value thereby obtained.	No
0–6-km bulk shear	m s^{-1}	Magnitude of 6 km AGL wind vector minus surface wind vector	
MESH	mm	Maximum estimated size of hail (based on radar reflectivity inside storm cell)	
Storm area	m^2		
Zonal storm radius	m		
Meridional storm radius	m		
Storm perimeter	m		
Current storm longevity	s		Yes
Zonal storm velocity	m s^{-1}		
Meridional storm velocity	m s^{-1}		
Storm speed (velocity magnitude)	m s^{-1}		

2. Data

Data used for this project fall into two categories. The first is training data, used to build and objectively evaluate the ML models. The second is human data, consisting of both subjective and objective evaluations from the human forecasters in HWT 2016 and 2017.

a. Training data

Because the goal of this project is to create models that can eventually be transitioned to full-time operations and evaluate those models in a quasi-operational naturalistic environment in the HWT, the training data must be available in real-time. The main source of training data is ProbSevere Cintineo et al. (2014), a real-time decision-support system for severe convection. Its main components are automated storm tracking and ML. Storm tracking is performed with segmotion (Lakshmanan et al. 2009), an algorithm in the Warning Decision Support System–Integrated Information (WDSS-II; Lakshmanan et al. 2007) software package. The tracking variable used in ProbSevere is composite reflectivity from the Multi-Radar Multi-Sensor (MRMS; Smith et al. 2016) system, which is updated every ~ 2 min. Storm objects¹ are defined as areas $\geq 20 \text{ km}^2$ with composite reflectivity $\geq 35 \text{ dBZ}$.

ProbSevere’s ML is done with a naive-Bayes classifier (NBC), which predicts the probability of severe weather (wind gust $\geq 25.7 \text{ m s}^{-1}$ or hail diameter $\geq 25.4 \text{ mm}$

or tornado) for each storm cell. These predictions are not temporally specific (e.g., a 25% tornado prediction means the system is 25% confident that the storm will produce a tornado at *some* time in the future) and ProbSevere does not automatically draw warning polygons. Thus, automated guidance on storm longevity should help the human forecasters in drawing warning polygons and focusing their attention (e.g., larger polygons and higher priority for longer-lived storms). ProbSevere’s ML uses five predictors, including MUCAPE, bulk shear, and maximum estimated size of hail (MESH). Our ML uses these and all other variables listed in Table 1.

We hypothesized that data pertaining to the near-storm environment (NSE), in addition to the storm itself, would improve predictions. Thus, we interpolate soundings from the Rapid Refresh model (RAP; Benjamin et al. 2016) to the center of each storm object. The interpolation method is nearest neighbor in space and previous neighbor in time (i.e., the entire sounding is taken from one grid cell) and the most recent RAP analysis (0-h forecast), which preserves physical consistency among the sounding variables. Then we compute 97 sounding indices for each storm object, using the Sounding and Hodograph Analysis and Research Program in Python (SHARPPy; Blumberg et al. 2017). The sounding indices include convective available potential energy (CAPE), convective inhibition (CIN), the supercell composite parameter (SCP), and many others with which forecasters are familiar. For a detailed description of all 97 indices, see Table A1 of Lagerquist et al. (2017). We omit the detailed description here because, due to limited

¹ One “storm object” is one storm cell at one time step; in other words, a snapshot of a storm cell.

computing resources, we cannot use the sounding indices in real time.

In preliminary work we also investigated the use of radar-derived variables from the MRMS. Although there is tremendous value in real-time radar data, much of the information therein is subsumed by the ProbSevere variables. Thus, using MRMS data added little predictive performance, which we determined not to be worth the additional processing time. If this were an operational or hindcasting system, not a quasi-operational one, using all available data would likely be more important.

Every ML model needs predictors (listed in Table 1) and truth, which is the actual storm longevity. Since ProbSevere includes storm tracking, we initially considered these longevities as true labels. However, it has been observed in the HWT that segmotion (the tracking algorithm used by ProbSevere) creates a large number of storm splits, where it splits what a human meteorologist would call one storm track (Harrison 2018). This is because segmotion is a real-time algorithm, meaning that it can look only at current and past data to make tracking decisions. WDSS-II includes a postevent tracking algorithm, called best track (Lakshmanan et al. 2015). “Postevent” means that best track is run in hindcast mode, allowing it to use both past and future data for tracking decisions, which reduces the frequency of track splitting. Thus, we use best track to create true labels of storm longevity.

As expected, best track significantly increases the storm longevities in our dataset, as shown in Fig. 1 for spring 2015 and 2016. For example, ~80% of ProbSevere storms live less than 60 min, which best track decreases to ~70%. We use best track only to create labels and not for predictors, which must use real-time data only. In HWT 2016, storm tracks (and thus, all predictors listed in Table 1) were taken directly from ProbSevere. In HWT 2017, storm tracks were altered by a real-time version of best track (Harrison 2018) leading to the “postprocessed” variables in Table 1.

We acquired data for every ProbSevere storm from 9 April 2015 to 24 March 2017. To make training more computationally feasible, we use only storms from April to June 2015 and 2016. We focus on the spring season, because (i) the HWT occurs in spring and (ii) the temporal frequency and spatial coverage of storm-damage reports are climatologically maximized in spring (Kelly et al. 1985). Storms are split into training, validation, and testing (Table 2). The purpose of training is to optimize the parameters of the ML model [e.g., the linear-regression coefficients in Eq. (1)]; the purpose of validation is to find the best

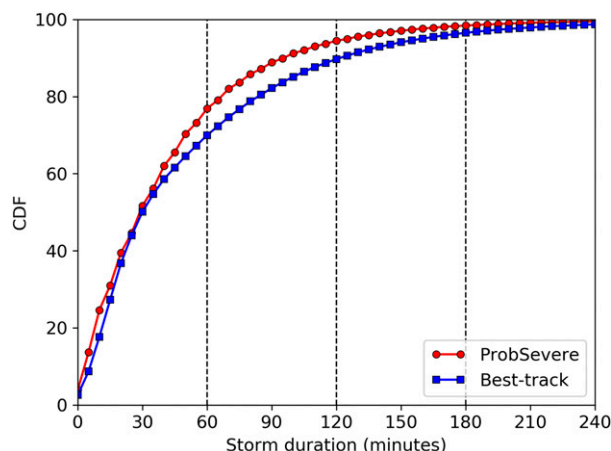


FIG. 1. Cumulative density function (CDF) of storm longevity for spring (April–June) 2015 and 2016, according to the ProbSevere and best track tracking algorithms.

hyperparameters (user-selected values that remain constant throughout training, such as the number of trees in a random forest); and the purpose of testing is to evaluate the chosen model on unseen data, which provides a reasonable expectation of future performance “in the wild” (e.g., in the HWT).

b. NOAA HWT data

Figure 2 shows a screenshot of the PHI tool (Karstens et al. 2015), which highlights the importance of longevity predictions to the forecasters. Longevity is used explicitly to determine the temporal extent of a warning/advisory and spatial extent of a warning/advisory polygon. It is also used by a separate ML algorithm that predicts the probability of severe weather for each storm cell for the predicted lifetime of the storm (Harrison 2018). In the absence of a prediction algorithm, PHI originally used a constant longevity prediction (60 min remaining), which is replaced with our ML system for the tests described in this paper.

ML predictions were integrated into PHI during HWT 2016 and 2017 (Karstens et al. 2018; Ling et al. 2017). Each year, the ML system was tested for three weeks by three NWS forecasters each week. The ML system used in the PHI experiments was an ensemble of gradient-boosted regression trees (GBRT), as described in section 3, and predictions were capped at 120 min due to logistical considerations. The ML models were trained on different data for 2016 versus 2017 (since 2017 data were not available in 2016), but the training procedures (section 3) were the same. Although the ML model results presented in this paper (RMSE, reliability, etc, shown below) are not the same as the models trained for HWT, the differences are small

TABLE 2. Training, validation, and testing periods for machine learning. One “storm object” is one storm cell at one time step.

Dataset	Period	No. of storm cells	No. of storm objects
Training	April–June 2015, April 2016	729 743	7 088 882
Validation	June 2016	120 036	1 388 231
Testing	May 2016	104 284	952 726

enough that the analysis of the results would be the same. Forecasters used the predictions in two settings: (i) displaced real-time events, where the forecasters receive pseudo-real-time data for a historical severe-thunderstorm outbreak, focusing on a single county warning area (CWA); and (ii) actual real-time events, occurring in the late afternoon and evening. A listing of these events and testing periods is provided in Karstens et al. (2018). In both settings the ProbSevere data, augmented with the longevity predictions described in this paper and other ML products (e.g., Lagerquist et al. 2017; McGovern et al. 2018), were provided to the forecasters as a first guess for issuing

severe thunderstorm warnings or subsevere products (“advisories”) at their discretion.

For each storm, forecasters expressed their level of confidence that severe wind or hail would occur within a certain time window (e.g., “0–45 min into the future”). The default time window was the storm longevity predicted by the ML system. In this work we objectively evaluate forecasters’ use of the ML longevity predictions—specifically, whether or not they modified the predictions, which is possible because their activities were logged.

3. Methods: Training and evaluation of machine learning

For the ML methods described in this paper, we use the implementation in Python’s scikit-learn library (Pedregosa et al. 2011). We chose three algorithms, based on prior experience using ML for weather prediction (e.g., McGovern et al. 2017). First, as a baseline method, we chose linear regression with elastic-net regularization (Zou and Hastie 2005). Linear regression produces an

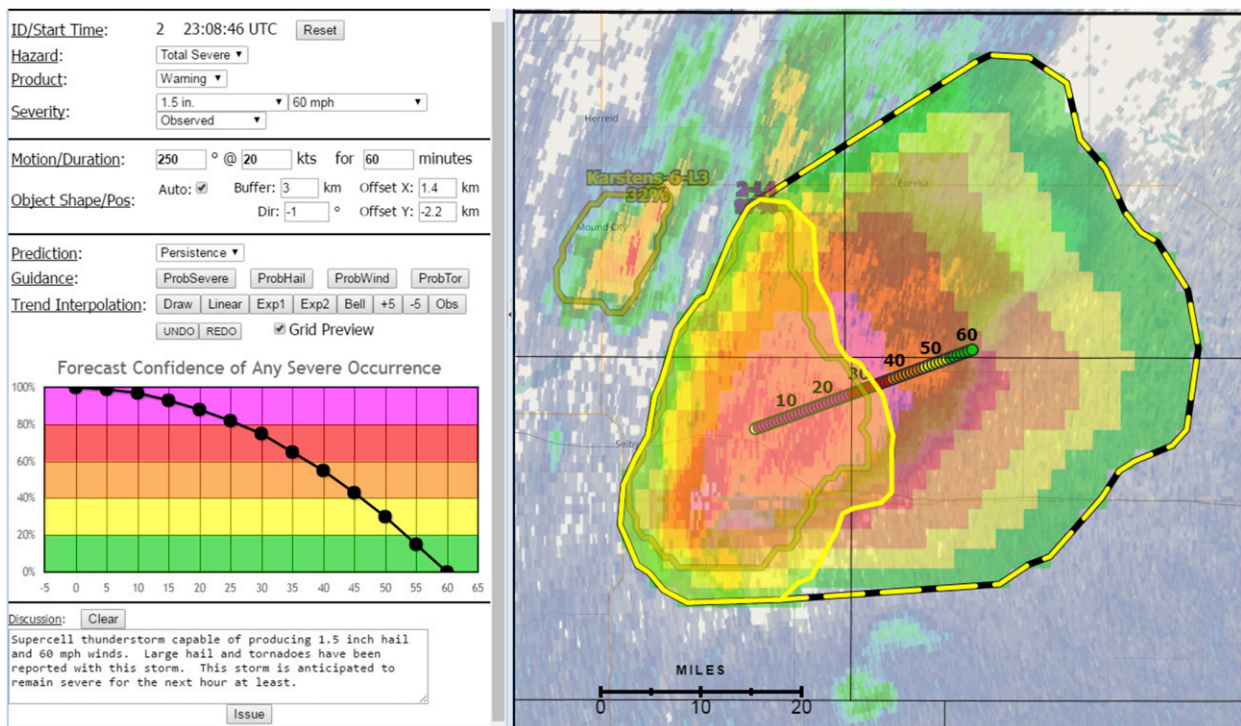


FIG. 2. Screenshot of the PHI tool, showing how predicted storm longevity may be used to determine the spatial and temporal extent of a warning. The solid yellow line in the right panel is a distance buffer around the storm of interest S ; the hatched yellow line is a warning polygon for S , which can be modified by the forecaster. The color fill in the background shows composite (column maximum) radar reflectivity; the color fill in the foreground shows severe-weather probability for S , and this field can be modified by shrinking, expanding, or changing the shape of the warning polygon. The graph in the left panel shows severe-weather probability vs lead time for S , which can be modified by many of the elements in the left panel, including the text field marked “Duration”. [Figure is from Karstens et al. (2018), their Fig. 2b.]

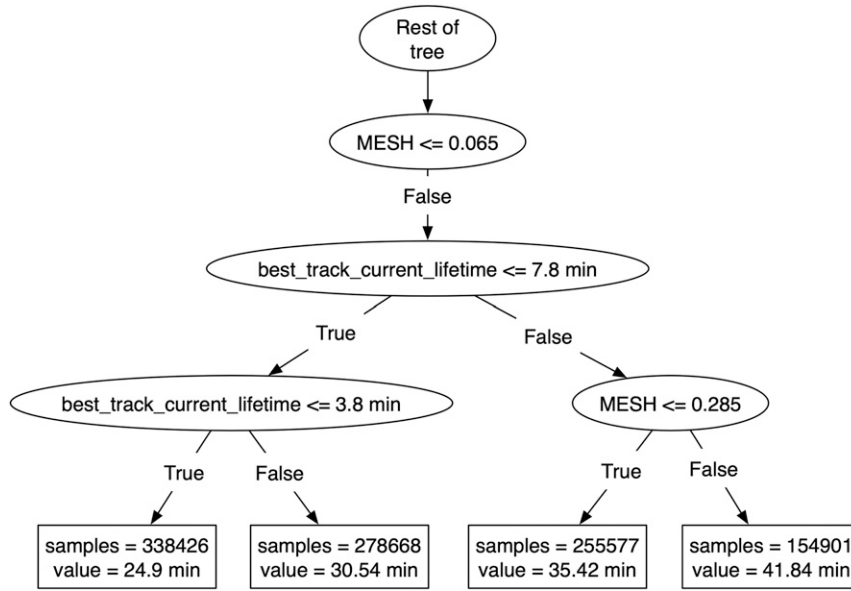


FIG. 3. Example of a regression tree. Each ellipse is a question/branch node, used to send each storm object down the appropriate branch of the tree. Once the storm object reaches a leaf node, its longevity is predicted based on storm objects in the training set that reached the same leaf node. Here best_track_current_lifetime is the current storm’s lifetime; mesh is maximum estimated size of hail (mm); mse is mean squared error (s^2); samples is the number of storm objects in the training set that reached a leaf node; and value is the forecast remaining longevity (s).

equation in the form of Eq. (1), where the x_j are predictors; β_0 and β_j are adjustable coefficients; and f is the resulting prediction, to be compared with the true value y . There are M predictors and N examples:

$$f = \beta_0 + \sum_{j=1}^M \beta_j x_j, \tag{1}$$

$$\text{loss} = \frac{1}{N} \sum_{i=1}^N \frac{1}{2} (f_i - y_i)^2 + \lambda \alpha \sum_{j=1}^M |\beta_j| + \lambda (1 - \alpha) \sum_{j=1}^M \frac{1}{2} \beta_j^2. \tag{2}$$

Without regularization, the loss function (minimized by training) is the mean squared error between the predicted and true values [the first term in Eq. (2)]. Elastic-net regularization is a combination of the lasso penalty (Tibshirani 1996), which is the second term in Eq. (2), and the ridge penalty (Hoerl and Kennard 1970, 1988), which is the third term. The variable $\lambda \in [0, \infty)$ determines the amount of regularization, and $\alpha \in [0, 1]$ determines the trade-off between the ridge and lasso penalties. Both penalties encourage the model to produce smaller regression coefficients (the β_j), and the lasso penalty specifically encourages the model to “zero out” coefficients, which effectively removes predictors from the model. Thus, elastic-net regularization encourages a simpler model, which often generalizes

better to unseen situations and noisy data. In ML terminology, regularization mitigates “overfitting” to the training data.

The second and third algorithms are decision tree based. Decision trees are popular in many applications, because they can identify the most important predictors and produce human-readable models. Figure 3 shows an example of a regression tree from one of the trees in our trained random forest. Because of the size of the full tree, this is just a small subset chosen for illustration. It shows the yes and no branches of the tree as well as the questions identified by the tree growing algorithm. At each leaf node, a regression tree predicts a constant value, which is shown in the rectangular nodes.

Decision-tree ensembles, such as random forests and GBRT, have recently been successful in many meteorological applications (Williams et al. 2008a,b; Gagne et al. 2009; McGovern et al. 2014; Williams 2014; McGovern et al. 2015; Clark et al. 2015; Elmore and Grams 2016). Ensembles usually have smaller bias and variance (mean squared error) than a single decision tree. In a random forest (Breiman 2001), each tree is trained with a bootstrap-resampled (Efron 1979) version of the training set. Thus, on average each tree sees only 63.2% of examples in the training set, which encourages diversity among the trees and improves the performance of the final ensemble. Conversely, in GBRT

(Friedman 2002), each tree is trained on the full training set. However, the k th tree is fit to the residual error from the first $k - 1$ trees, rather than fitting to the true label (observed longevity) as in random forests. Also, examples with the largest residuals are weighted the most heavily (Schapire 2003), which encourages the GBRT to improve its worst predictions.

We also experiment with bias-correcting predictions for each model, using isotonic regression (Niculescu-Mizil and Caruana 2005). Isotonic regression learns a stepwise function that maps from the base model's predictions to calibrated predictions, in a way that minimizes mean squared error [the first term in Eq. (2)]. The sole input (predictor) for isotonic regression is the base model's prediction, and the sole output is the calibrated prediction. The training set for isotonic regression must be independent of that for the base model (if the two training sets are the same, isotonic regression will be trained only on cases for which the base model performs uncharacteristically well, so it will learn to calibrate only uncharacteristically good predictions). In this work we use the validation data (Table 1) to train isotonic regression.

When evaluating forecast quality (the performance of an ML algorithm), we compare to two non-ML baselines. The first is the constant method (originally used in PHI), which is to predict a remaining longevity of 60 min for all storms. Although this baseline is easy to outperform, it is important to establish that we have improved upon the previously used method. The second baseline is "persistence," where the remaining longevity of the storm is predicted to equal its current longevity. This is also known as the Lindy effect (Goldman 1964). For example, if the storm is 15 min old, persistence predicts that it will last another 15 min, so its total longevity will be 30 min.

Performance is measured in four ways. First, we measure the root-mean-squared error (RMSE), which is $\sqrt{(1/N)\sum_{i=1}^N (f_i - y_i)^2}$, with variables defined as in Eq. (2). Second, we compare the predicted and observed cumulative density functions (CDFs) of storm longevity, which helps to identify model bias, both in an overall sense and within certain longevity ranges. Third, we compare the predicted and observed probability density functions (PDFs) of storm longevity. For this we use violin plots², which synthesize the information shown in a typical PDF and a typical boxplot. Fourth, we plot reliability curves, which show the mean observed longevity (y axis) for each bin of predicted longevity (x axis). A perfect reliability curve is the line $x = y$,

which means that the conditional expected value is always the predicted value (given a predicted longevity of T seconds, the mean observation is always T seconds). The main purpose of the reliability curve is to identify conditional bias, or bias within certain ranges of the prediction space.

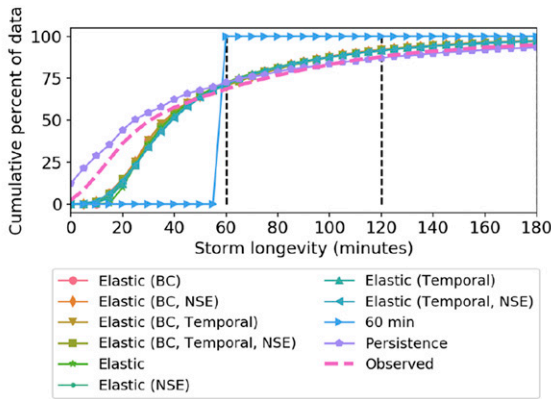
Our random forests and GBRT ensembles each contain hundreds of trees, which partly impairs human readability (although each tree alone is human readable, reading them all would take many hours and synthesizing the information thereby gleaned would be nearly impossible). The permutation method introduced by Breiman (2001) partly circumvents this problem, by quantifying the importance of each predictor. Specifically, for each predictor x_j , the training values are randomly permuted, yielding the perturbed training set X'_j . Then an already-trained model (random forest or GBRT) is used to generate predictions for X'_j . The "importance" of predictor x_j is defined as the mean squared error [first term in Eq. (2)] on the perturbed training set X'_j , minus MSE on the original training set. Thus, the most "important" predictors are those whose random permutation leads to the greatest decrease in performance (increase in MSE). Predictors can be ranked by importance, which allows some human insight into the workings of the model.

4. Results

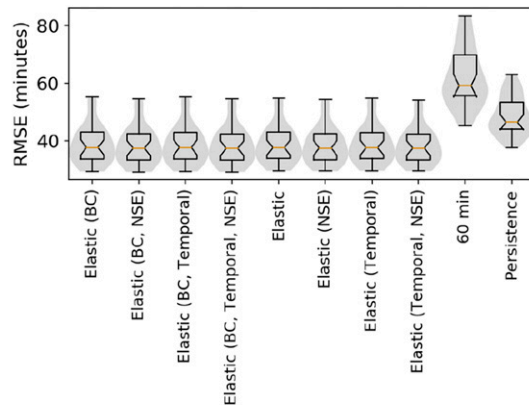
The training of elastic nets requires very little computing time, and they often perform nearly as well as more sophisticated methods. For these reasons we use elastic nets in our first experiment, to determine which predictors are necessary for training. Specifically, we train elastic nets with and without NSE data (section 2a), with and without temporal data (where the predictors in Table 1 are computed for both the current and previous time steps of the storm), and with and without bias correction (section 3). This yields eight models ($2 \times 2 \times 2$), for which the performance is shown in Fig. 4. All results in Figs. 4 and 5 are computed on the testing data, as detailed in Table 2. To create a distribution, such as shown in Figs. 4b and 5b, the RMSE is computed on each day in the testing set independently. Bootstrapping with 1000 replicates is used to create distributions.

Figure 4a shows the CDF's of observed storm longevity, predicted longevity from the eight elastic nets, and predicted longevity from the two baselines. The eight elastic nets are clustered together so tightly that they are almost impossible to distinguish. The same is true in all panels of Fig. 4, where the elastic nets have

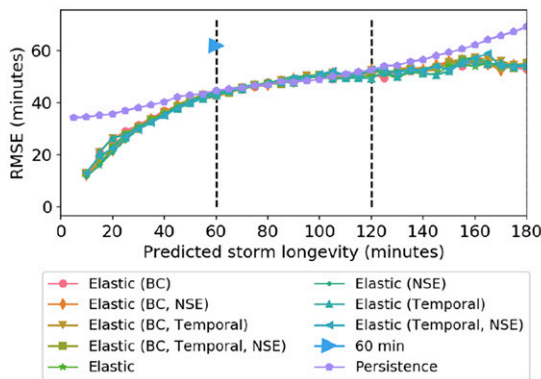
² http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.violinplot.



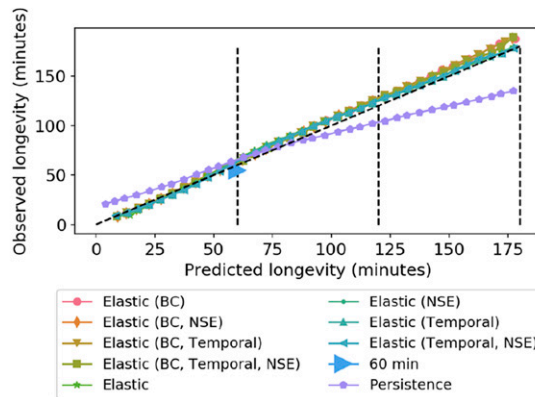
(a) CDF of observed storm longevity and predicted longevity for the eight elastic nets and the two baselines.



(b) Violin plot of RMSE showing the typical box plot with the PDF of the distribution of RMSE in grey.



(c) RMSE as a function of predicted longevity.



(d) Reliability diagram.

FIG. 4. Forecast evaluation for the eight elastic nets and two baseline methods (60 min remaining and persistence). “BC” means that the elastic net is bias corrected with isotonic regression; “temporal” means that it is trained with temporal data; and “NSE” means that it is trained with sounding indices, representing the near-storm environment.

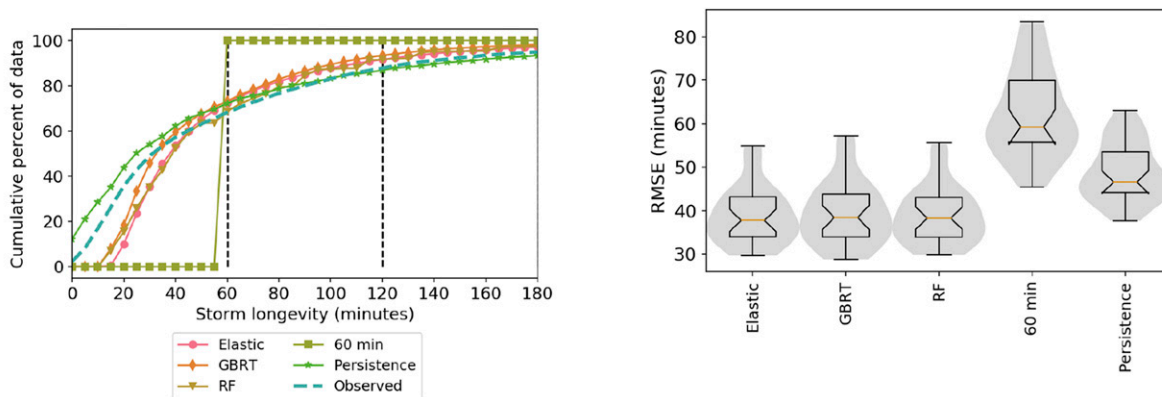
nearly identical errors but are clearly distinguishable from the baselines.

Three conclusions can be drawn from Fig. 4. First, NSE and temporal data yield very little performance gain. Also, computing NSE and temporal data takes about four minutes for each ProbSevere update (which come at intervals of about two minutes and are accompanied by MRMS data). This latency time is unacceptable, given that (i) many other storm-based predictions are included in the ProbSevere data, so their latency time is four minutes less; and (ii) thunderstorms evolve quickly, so using 4-min-older predictions can be a serious disadvantage. Thus, we chose the smallest predictor set (Table 1) for deployment in the HWT. The lack of improvement with NSE data may be surprising, but (i) these values have little temporal variance relative to the radar-derived predictors in ProbSevere that can change significantly with each two minute update

(ii) some ProbSevere predictors (MUCAPE and bulk shear) are already based on NWP soundings.

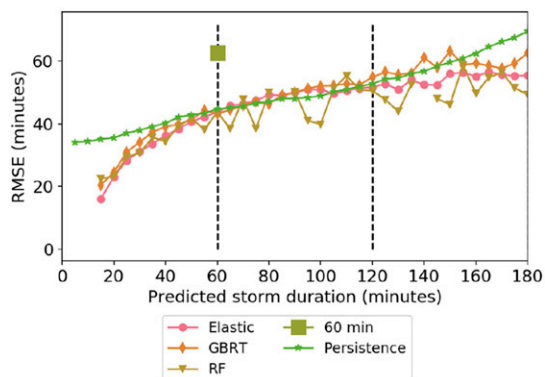
Second, although bias correction has proven important in prior experiments with this and other meteorological data, it did not provide any performance gain for this problem. Since bias correction adds computing time, which is cumbersome for real-time deployment, we did not use it in the HWT or the remaining experiments.

Third, comparing to the two baseline methods (60 min and persistence), Fig. 4 indicates the need for learning. Elastic nets with all eight parameter settings outperform the baseline methods, according to all performance metrics. As shown in Figs. 4a and 4d, persistence has a strong underforecasting bias, while the constant method has no resolution at all. As shown in Fig. 4c, although the baselines have a similar RMSE for some values of predicted longevity, their RMSE is generally much higher than the ML models.

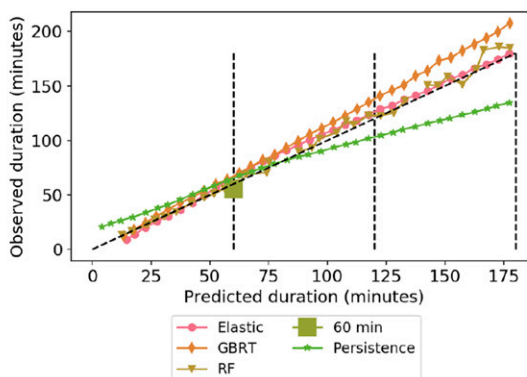


(a) CDF of observed storm longevity, predicted longevity for the three ML models and the two baselines.

(b) Violin plot of RMSE showing the typical box plot with the PDF of the distribution of RMSE in grey.



(c) RMSE as a function of predicted longevity.



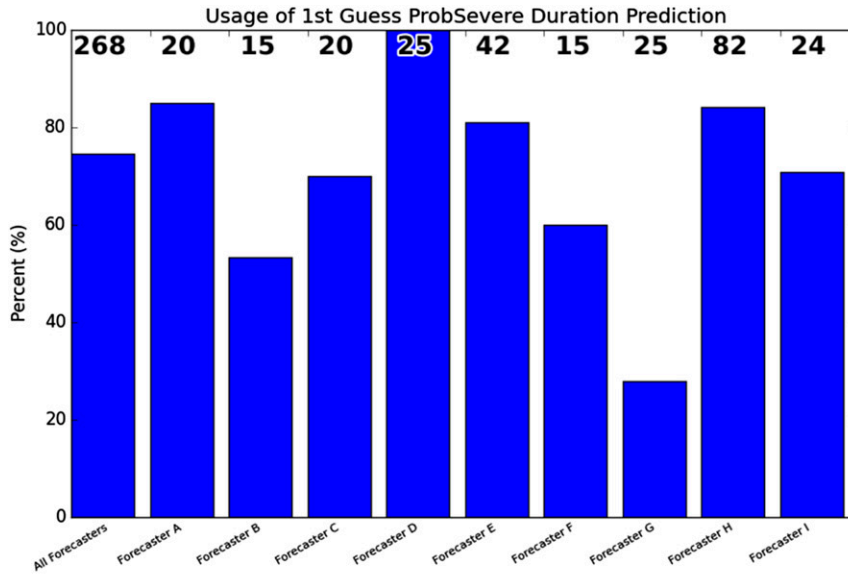
(d) Reliability diagram.

FIG. 5. Forecast evaluation for the three ML and two baseline methods (60 min remaining and persistence). “RF” means random forest. Each model was trained without temporal data, NSE data, or isotonic regression.

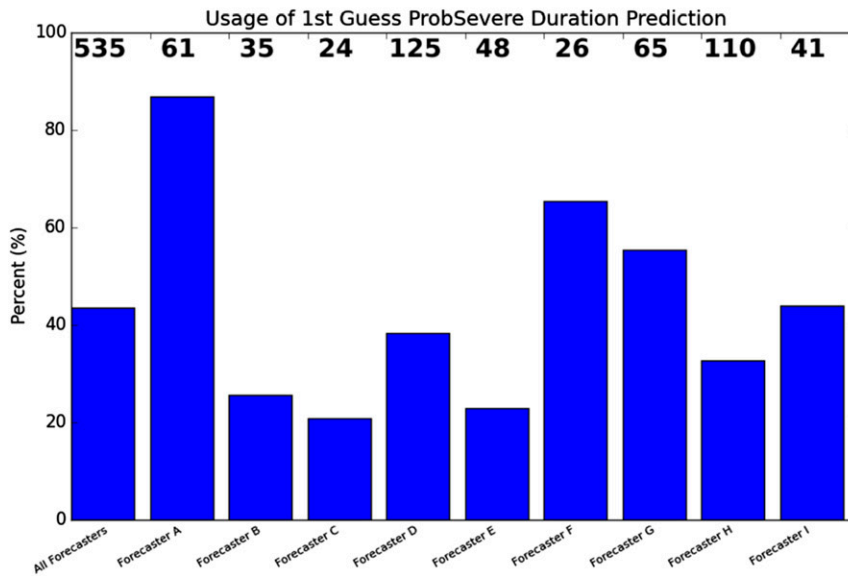
Given the recent success of decision-tree ensembles in meteorology (section 3), we hypothesized that they would perform the best of the ML models. Figure 5 compares random forests and GBRT ensembles to elastic nets. Each model is trained with only the predictors in Table 1. We used cross validation (not shown) to choose the number of trees per ensemble and maximum depth per tree. However, in choosing these values, we had to consider the computing time needed to run models in a quasi-operational naturalistic environment. There may be hundreds of storm cells in the CONUS at one time, and applying a large model to all storms could take too long. We empirically decided on 250 trees per ensemble (random forest or GBRT), with a maximum depth of five branch nodes per tree. Deeper trees or larger forests could slightly improve predictive power but at a significant computational cost, limiting their use in real time. Elastic nets used the default hyperparameters.

As shown in Fig. 5, both types of decision-tree ensembles improve the predictions, especially for short-lived storms. They are capable of predicting longevitys of 10–30 min, whereas elastic nets rarely predict less than 30 min. Figure 5 shows that, for most values of predicted longevity, the three ML models unanimously outperform the baselines and that, while the three ML models have near-perfect reliability, the decision-tree ensembles are more reliable at the lower end of the range (improve on the elastic net’s over-forecasting bias) and the elastic net is more reliable at the higher end of the range (improves on the decision trees’ underforecasting bias). We chose the GBRT for HWT deployment, since it (i) predicts short-lived storms better than the elastic net and (ii) produces smoother error graphs than the random forest.

Figure 6 shows forecasters’ usage of the predictions in 2016 and 2017. Most evident is a drastic reduction in usage frequency (by “all forecasters”), from ~75% in



(a) HWT 2016



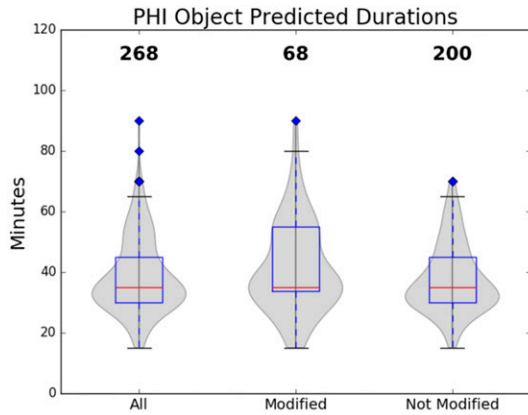
(b) HWT 2017

FIG. 6. Forecaster usage of ML longevity predictions in HWT (a) 2016 and (b) 2017. The “1st Guess ProbSevere Duration Prediction” refers to our ML predictions, and the number at the top of each bar is the total number of ML predictions seen by the forecaster. Note that (a) also appeared in [McGovern et al. \(2017\)](#).

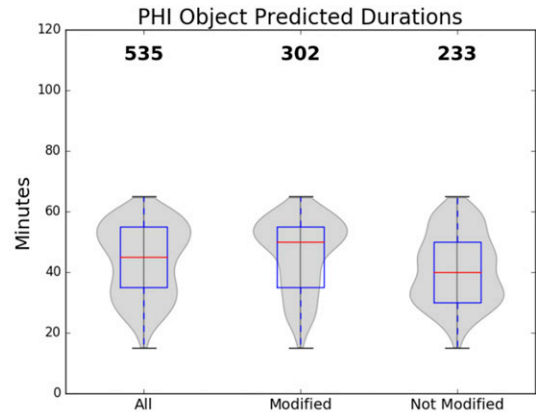
2016 to ~40% in 2017. Why did this reduction occur? [Fig. 7](#) shows, for each year, the distribution of all ML-predicted longevitys, those that were modified by humans (before modification), and those that were unmodified. The ML predictions (and observed storm durations) were generally higher in 2017 than in 2016, and in both years forecasters preferentially modified the higher predictions. This action is also evident

when the distributions are split into storms for which the forecaster issued a warning or advisory ([Fig. 7](#)).

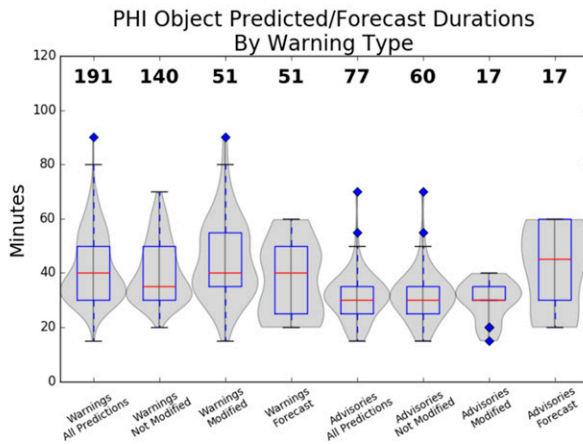
[Figure 7](#) shows *how* the ML predictions were modified, as well as how the modified and unmodified distributions compare to observations. In 2017 both the ML predictions and observations were longer than in 2016, which seems to justify the longer ML predictions. In 2016 human modification brings the



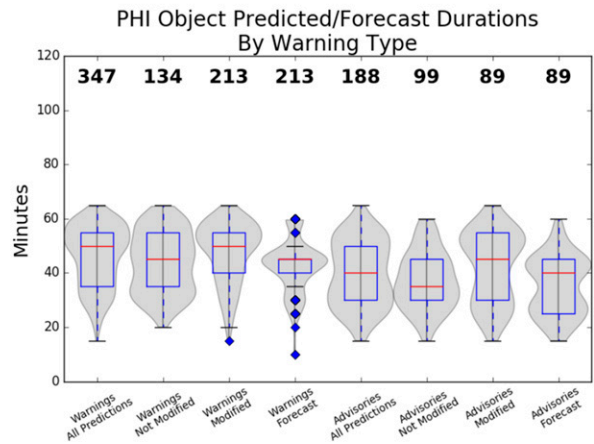
(a) Distribution of ML-predicted longevitys, those that were modified by humans (before modification), and those that were unmodified – for HWT 2016.



(b) As in panel (a) but for HWT 2017.



(c) As in panel (a) but including only storms for which the forecaster issued a warning or advisory. “Warnings / All Predictions” shows all warned storms; “Warnings / Not Modified” shows warned storms for which the ML longevity was unmodified; “Warnings / Modified” and “Warnings / Forecast” show warned storms for which the longevity prediction was modified, before and after modification respectively.



(d) As in panel (c) but for HWT 2017.

FIG. 7. Distributions of modified and unmodified ML longevity predictions. The violin plot shows a standard boxplot and PDF of the data distribution (shaded).

median prediction closer to the median observation, but in 2017 it has the opposite effect, exacerbating the under-forecasting bias of the ML system. However, in both years the ML system has slightly lower error than the humans.

Despite the large interannual difference between ML predictions, there is little interannual difference between the modified predictions. This implies that human forecasters have a preferred range of longevity values and prefer not to adjust this range much to accommodate new situations. Similar behavior is shown

in Fig. 8, especially in 2016, where the predictions are generally adjusted downward for storms with warnings and upward for storms with advisories, resulting in very similar postmodification distributions.

Figure 9 (from Harrison and Karstens 2017) shows the duration distribution for storm-based warnings (SBW). By comparison with Fig. 7, human-modified storm longevity has a very similar distribution to severe thunderstorm warnings. Thus, the human forecasters’ tendency to more frequently change/reduce

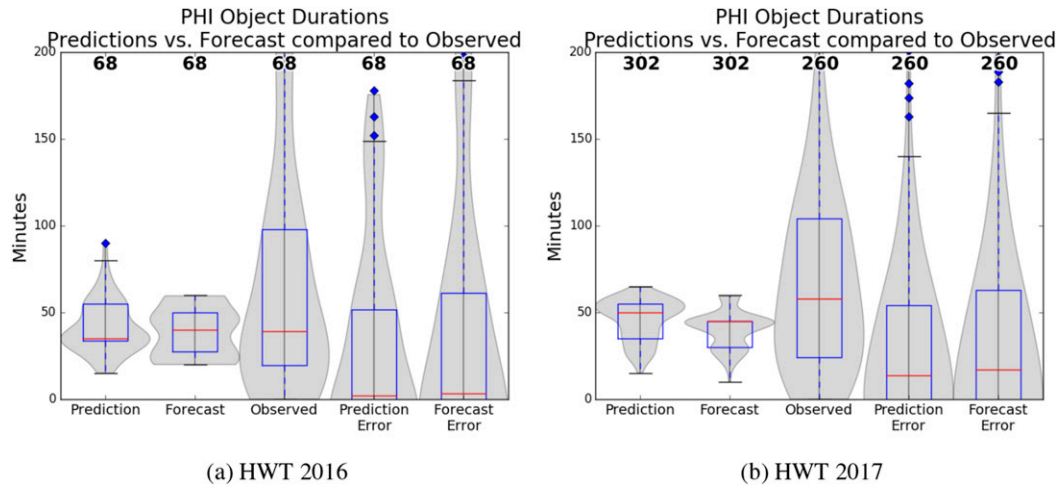


FIG. 8. Error distributions for modified and unmodified ML longevity predictions. “Prediction” and “Forecast” show storms for which the longevity was modified, before and after modification respectively. “Observed” shows the observed storm durations for this modified subset of longevity predictions. “Prediction Error” and “Forecast Error” show the distributions of absolute error resulting from subtracting each prediction and forecast with its corresponding observed value, respectively.

longevity predictions in 2017 was likely caused in part by their application of SBW-era training to the HWT experiment.

5. Discussion

Using the permutation method (section 3), we examined differences in predictor importance across the three models (elastic net, random forest, and GBRT). The models are consistent in their ranking of the most important variables. All three models choose the current storm longevity as the most important variable. This is not surprising, especially given that the persistence method (which uses only current longevity as a predictor) performs reasonably (Figs. 4 and 5). The second-most important variable (again, chosen unanimously by all three models) is MESH within the storm. This also makes sense intuitively, as storms with large hail tend to have stronger updrafts and to be longer lived. Shear was identified by many of the studies discussed in the introduction as the most important parameter and it shows up indirectly in the most important variables.

Figure 7 shows that human modification of the ML predictions slightly increased the error, implying that forecasters are perhaps better served by using the raw ML prediction for storm longevity. However, at least two outstanding questions remain. First, these distributions are based on a large number of cases, and there may be certain situations (e.g., storm modes, mesoscale regimes, or synoptic regimes) where human modification generally improves the predictions. A more detailed

analysis could reveal these regimes and other aspects of human-machine interdependence that we have not considered. Second, the ML predictions for warned storms (those associated with a warning) were generally greater than subsevere storms (those associated with advisories), and forecasters modified the ML predictions for warned storms more often (Fig. 7). It makes sense that stronger, more organized storms last longer, while weaker ones remain more transient. However, this rationale considers only the temporal dimension of storm longevity. Uncertainty in the spatial dimension

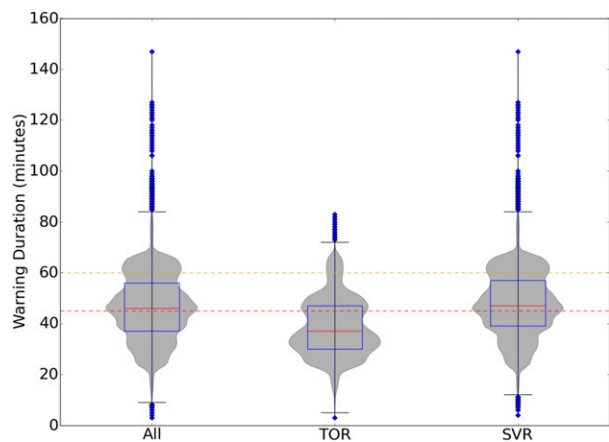


FIG. 9. Distributions of SBW durations for October 2007–May 2016. “TOR” means tornado warning; “SVR” means severe thunderstorm warning; and “All” combines both warning types. The dashed red and yellow lines are the NWS’ maximum recommended duration for TOR and SVR warnings, respectively. From Harrison and Karstens (2017).

(i.e., the forecast location of a storm) increases with lead time, which may be one reason that forecasters compressed longevity predictions into a smaller range with sound empirical basis from the SBW era, especially for warning decisions. This insight implies that spatially joining a warning with an advisory (i.e., warning for shorter lead times and advising for longer lead times thereafter) may be a viable way to transition between the current and FACETs warning paradigms, which bifurcates the plume into warning duration and storm longevity, respectively.

We have demonstrated that machine learning can be used for real-time prediction of storm longevity and provide valuable information to forecasters. As additional storm data become available in real time, and as we develop faster and more sophisticated ways to process said data, the performance of the ML system should improve. For example, although MRMS data are available in real time, processing is slow (the grids are CONUS-wide with 0.01° spacing) and our processing methods led to minimal performance gain. We anticipate that data from high-resolution convection-allowing models, and new sensing systems such as the Geostationary Operational Environmental Satellite-R Series, will allow significant performance gains. Furthermore, since HWT 2016 and 2017, the algorithms for computing velocity-derived variables (azimuthal shear and convergence) in MRMS have improved. All other MRMS variables are reflectivity-derived, so azimuthal shear and convergence may contain valuable new information. Last, we are currently working on a real-time system for classifying storm mode (e.g., supercell, multicell cluster, linear system), which will be useful to forecasters and may provide a valuable input to the longevity model.

Acknowledgments. The authors thank David Harrison for his assistance in refining the best track algorithm used in this study. The authors also thank the anonymous reviewers for their help in refining the manuscript. Funding was provided by NOAA/Office of Oceanic and Atmospheric Research under NOAA–University of Oklahoma Cooperative Agreement NA16OAR4320115, U.S. Department of Commerce. This work was supported by the NEXRAD Product Improvement Program, by NOAA/Office of Oceanic and Atmospheric Research. The statements, findings, conclusions, and recommendations are those of the authors and do not necessarily reflect the views of NOAA, the U.S. Department of Commerce, or the University of Oklahoma. The computing for this project was performed at the OU Supercomputing Center for Education and Research (OSCER) at the University of Oklahoma (OU).

REFERENCES

- Benjamin, S., and Coauthors, 2016: A North American hourly assimilation and model forecast cycle: The Rapid Refresh. *Mon. Wea. Rev.*, **144**, 1669–1694, <https://doi.org/10.1175/MWR-D-15-0242.1>.
- Blumberg, W., K. Halbert, T. Supinie, P. Marsh, R. Thompson, and J. Hart, 2017: SHARPPy: An Open Source Sounding Analysis Toolkit for the Atmospheric Sciences. *Bull. Amer. Meteor. Soc.*, **98**, 1625–1636, <https://doi.org/10.1175/BAMS-D-15-00309.1>.
- Breiman, L., 2001: Random forests. *Mach. Learn.*, **45**, 5–32, <https://doi.org/10.1023/A:1010933404324>.
- Brooks, H., 1992: Operational implications of the sensitivity of modelled thunderstorms to thermal perturbations. *Fourth AES/CMOS Workshop on Operational Meteorology*, Whistler, BC, Canada, Atmospheric Environment Service (Canada) and Canadian Meteorological and Oceanographic Society, 398–407.
- Bunkers, M., J. Johnson, J. Grzywacz, L. Czepyha, and B. Klimowski, 2002: A preliminary investigation of supercell longevity. *21st Conf. on Severe Local Storms*, San Antonio, TX, Amer. Meteor. Soc., 16.6, https://ams.confex.com/ams/SLS_WAF_NWP/techprogram/paper_47315.htm.
- Cintineo, R. M., and D. J. Stensrud, 2013: On the predictability of supercell thunderstorm evolution. *J. Atmos. Sci.*, **70**, 1993–2011, <https://doi.org/10.1175/JAS-D-12-0166.1>.
- Cintineo, J., M. Pavolonis, J. Sieglaff, and D. Lindsey, 2014: An empirical model for assessing the severe weather potential of developing convection. *Wea. Forecasting*, **29**, 639–653, <https://doi.org/10.1175/WAF-D-13-00113.1>.
- Clark, A., and Coauthors, 2012: An overview of the 2010 Hazardous Weather Testbed Experimental Forecast Program Spring Experiment. *Bull. Amer. Meteor. Soc.*, **93**, 55–74, <https://doi.org/10.1175/BAMS-D-11-00040.1>.
- , A. MacKenzie, A. McGovern, V. Lakshmanan, and R. Brown, 2015: An automated, multiparameter dryline identification algorithm. *Wea. Forecasting*, **30**, 1781–1794, <https://doi.org/10.1175/WAF-D-15-0070.1>.
- Dahl, B., 2014: Sensitivity of vortex production to small environmental perturbations in high-resolution supercell simulations. M. S. thesis, School of Meteorology, University of Oklahoma, 83 pp.
- Dixon, M., and G. Wiener, 1993: TITAN: Thunderstorm Identification, Tracking, Analysis, and Nowcasting—A radar-based methodology. *J. Atmos. Oceanic Technol.*, **10**, 785–797, [https://doi.org/10.1175/1520-0426\(1993\)010<0785:TITAA>2.0.CO;2](https://doi.org/10.1175/1520-0426(1993)010<0785:TITAA>2.0.CO;2).
- Efron, B., 1979: Bootstrap methods: Another look at the jackknife. *Ann. Stat.*, **7**, 1–26, <https://doi.org/10.1214/aos/1176344552>.
- Elmore, K., and H. Grams, 2016: Using mPING data to generate random forests for precipitation type forecasts. *14th Conf. on Artificial and Computational Intelligence and its Applications to the Environmental Sciences*, New Orleans, LA, Amer. Meteor. Soc., 4.2, <https://ams.confex.com/ams/96Annual/webprogram/Paper289684.html>.
- , D. Stensrud, and K. Crawford, 2002: Explicit cloud-scale models for operational forecasts: A note of caution. *Wea. Forecasting*, **17**, 873–884, [https://doi.org/10.1175/1520-0434\(2002\)017<0873:ECSMFO>2.0.CO;2](https://doi.org/10.1175/1520-0434(2002)017<0873:ECSMFO>2.0.CO;2).
- Friedman, J., 2002: Stochastic gradient boosting. *Comput. Stat. Data Anal.*, **38**, 367–378, [https://doi.org/10.1016/S0167-9473\(01\)00065-2](https://doi.org/10.1016/S0167-9473(01)00065-2).
- Gagne, D., A. McGovern, and J. Brotzge, 2009: Classification of convective areas using decision trees. *J. Atmos. Oceanic Technol.*, **26**, 1341–1353, <https://doi.org/10.1175/2008JTECHA1205.1>.

- Gallo, B., and Coauthors, 2017: Breaking new ground in severe weather prediction: The 2015 NOAA/Hazardous Weather Testbed Spring Forecasting Experiment. *Wea. Forecasting*, **32**, 1541–1568, <https://doi.org/10.1175/WAF-D-16-0178.1>.
- Goldman, A., 1964: Lindy's law. *The New Republic*, 34–35.
- Harrison, D., 2018: Correcting, improving, and verifying automated guidance in a new warning paradigm. M. S. thesis, School of Meteorology, University of Oklahoma, 109 pp.
- , and C. Karstens, 2017: A climatology of operational storm-based warnings: A geospatial analysis. *Wea. Forecasting*, **32**, 47–60, <https://doi.org/10.1175/WAF-D-15-0146.1>.
- Hoerl, A., and R. Kennard, 1970: Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, **12**, 55–67, <https://doi.org/10.1080/00401706.1970.10488634>.
- , and —, 1988: Ridge regression. *Encyclopedia of Statistical Sciences*, S. Kotz, Ed., Vol. 8, Wiley, <https://doi.org/10.1002/0471667196.ess2280.pub2>.
- Houston, A., and R. Wilhelmson, 2011: The dependence of storm longevity on the pattern of deep convection initiation in a low-shear environment. *Mon. Wea. Rev.*, **139**, 3125–3138, <https://doi.org/10.1175/MWR-D-10-05036.1>.
- Karstens, C., and Coauthors, 2015: Evaluation of a probabilistic forecasting methodology for severe convective weather in the 2014 Hazardous Weather Testbed. *Wea. Forecasting*, **30**, 1551–1570, <https://doi.org/10.1175/WAF-D-14-00163.1>.
- , and Coauthors, 2017: Prototyping a next-generation severe weather warning system for FACETs. *Seventh Conf. on Transition of Research to Operations*, Seattle, WA, Amer. Meteor. Soc., 8.1, <https://ams.confex.com/ams/97Annual/webprogram/Paper314063.html>.
- , and Coauthors, 2018: Development of a human-machine mix for forecasting severe convective events. *Wea. Forecasting*, **33**, 715–737, <https://doi.org/10.1175/WAF-D-17-0188.1>.
- Kelly, D. L., J. T. Schaefer, and C. A. Doswell III, 1985: Climatology of nontornadic severe thunderstorms events in the United States. *Mon. Wea. Rev.*, **113**, 1997–2014, [https://doi.org/10.1175/1520-0493\(1985\)113<1997:CONSTE>2.0.CO;2](https://doi.org/10.1175/1520-0493(1985)113<1997:CONSTE>2.0.CO;2).
- Lagerquist, R., A. McGovern, and T. Smith, 2017: Machine learning for real-time prediction of damaging straight-line convective wind. *Wea. Forecasting*, **32**, 2175–2193, <https://doi.org/10.1175/WAF-D-17-0038.1>.
- Lakshmanan, V., T. Smith, G. Stumpf, and K. Hondl, 2007: The Warning Decision Support System—Integrated Information. *Wea. Forecasting*, **22**, 596–612, <https://doi.org/10.1175/WAF1009.1>.
- , K. Hondl, and R. Rabin, 2009: An efficient, general-purpose technique for identifying storm cells in geospatial images. *J. Atmos. Oceanic Technol.*, **26**, 523–537, <https://doi.org/10.1175/2008JTECHA1153.1>.
- , B. Herzog, and D. Kingfield, 2015: A method for extracting postevent storm tracks. *J. Appl. Meteor. Climatol.*, **54**, 451–462, <https://doi.org/10.1175/JAMC-D-14-0132.1>.
- Li, N., M. Wei, B. Niu, and X. Mu, 2012: A new radar-based storm identification and warning technique. *Meteor. Appl.*, **19**, 17–25, <https://doi.org/10.1002/met.249>.
- Lilly, D., 1990: Numerical prediction of thunderstorms—Has its time come? *Quart. J. Roy. Meteor. Soc.*, **116**, 779–798, <https://doi.org/10.1002/qj.49711649402>.
- Ling, C., and Coauthors, 2017: Forecasters' mental workload while issuing Probabilistic Hazard Information (PHI) during 2016 FACETs PHI Hazardous Weather Testbeds. *33rd Conf. on Environmental Information Processing Technologies*, Seattle, WA, Amer. Meteor. Soc., J9.3, <https://ams.confex.com/ams/97Annual/webprogram/Paper314172.html>.
- MacKeen, P., H. Brooks, and K. Elmore, 1999: Radar reflectivity-derived thunderstorm parameters applied to storm longevity forecasting. *Wea. Forecasting*, **14**, 289–295, [https://doi.org/10.1175/1520-0434\(1999\)014<0289:RRDTPA>2.0.CO;2](https://doi.org/10.1175/1520-0434(1999)014<0289:RRDTPA>2.0.CO;2).
- McGovern, A., D. Gagne, J. Williams, R. Brown, and J. Basara, 2014: Enhancing understanding and improving prediction of severe weather through spatiotemporal relational learning. *Mach. Learn.*, **95**, 27–50, <https://doi.org/10.1007/s10994-013-5343-x>.
- , —, J. Basara, T. Hamill, and D. Margolin, 2015: Solar energy prediction: An international contest to initiate interdisciplinary research on compelling meteorological problems. *Bull. Amer. Meteor. Soc.*, **96**, 1388–1395, <https://doi.org/10.1175/BAMS-D-14-00006.1>.
- , K. Elmore, D. Gagne, S. Haupt, C. Karstens, R. Lagerquist, T. Smith, and J. Williams, 2017: Using artificial intelligence to improve real-time decision-making for high-impact weather. *Bull. Amer. Meteor. Soc.*, **98**, 2073–2090, <https://doi.org/10.1175/BAMS-D-16-0123.1>.
- , E. Jergensen, C. Karstens, H. Obermeier, and T. Smith, 2018: Real-time and climatological storm classification using machine learning. *17th Conf. on Artificial and Computational Intelligence and Its Applications to the Environmental Sciences*, Austin, TX, Amer. Meteor. Soc., 1.1, <https://ams.confex.com/ams/98Annual/webprogram/Paper326198.html>.
- Niculescu-Mizil, A., and R. Caruana, 2005: Predicting good probabilities with supervised learning. *Proc. 22nd Int. Conf. on Machine Learning*, Bonn, Germany, International Machine Learning Society, 625–632, <https://dl.acm.org/citation.cfm?id=1102430>.
- Parker, M., 2007: Simulated convective lines with parallel stratiform precipitation. *J. Atmos. Sci.*, **64**, 267–288, <https://doi.org/10.1175/JAS3853.1>.
- Pedregosa, F., and Coauthors, 2011: Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, **12**, 2825–2830.
- Rothfusz, L., C. Karstens, and D. Hilderbrand, 2014: Forecasting a continuum of environmental threats: Exploring next-generation forecasting of high impact weather. *Eos, Trans. Amer. Geophys. Union*, **95**, 325–326, <https://doi.org/10.1002/2014EO360001>.
- , R. Schneider, D. Novak, K. Klockow-McClain, A. E. Gerard, C. Karstens, G. J. Stumpf, and T. M. Smith, 2018: FACETs: A proposed next-generation paradigm for high-impact weather forecasting. *Bull. Amer. Meteor. Soc.*, **99**, 2025–2043, <https://doi.org/10.1175/BAMS-D-16-0100.1>.
- Rotunno, R., J. Klemp, and M. Weisman, 1988: A theory for strong, long-lived squall lines. *J. Atmos. Sci.*, **45**, 463–485, [https://doi.org/10.1175/1520-0469\(1988\)045<0463:ATFSL>2.0.CO;2](https://doi.org/10.1175/1520-0469(1988)045<0463:ATFSL>2.0.CO;2).
- Schapire, R., 2003: The boosting approach to machine learning: An overview. *Nonlinear Estimation and Classification*, D. Denison et al., Eds., Springer, 149–171.
- Smith, T., and Coauthors, 2016: Multi-Radar Multi-Sensor (MRMS) severe weather and aviation products: Initial operating capabilities. *Bull. Amer. Meteor. Soc.*, **97**, 1617–1630, <https://doi.org/10.1175/BAMS-D-14-00173.1>.
- Thorpe, A., M. Miller, and M. Moncrieff, 1982: Two-dimensional convection in non-constant shear: A model of mid-latitude squall lines. *Quart. J. Roy. Meteor. Soc.*, **108**, 739–762, <https://doi.org/10.1002/qj.49710845802>.
- Tibshirani, R., 1996: Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B*, **58**, 267–288, <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>.

- Weisman, M., and J. Klemp, 1982: The dependence of numerically simulated convective storms on vertical wind shear and buoyancy. *Mon. Wea. Rev.*, **110**, 504–520, [https://doi.org/10.1175/1520-0493\(1982\)110<0504:TDONSC>2.0.CO;2](https://doi.org/10.1175/1520-0493(1982)110<0504:TDONSC>2.0.CO;2).
- , and —, 1986: Characteristics of isolated convective storms. *Mesoscale Meteorology and Forecasting*, P. Ray, Ed., Amer. Meteor. Soc., 331–358.
- Williams, J., 2014: Using random forests to diagnose aviation turbulence. *Mach. Learn.*, **95**, 51–70, <https://doi.org/10.1007/s10994-013-5346-7>.
- , D. Ahijevych, S. Dettling, and M. Steiner, 2008a: Combining observations and model data for short-term storm forecasting. *Proc. SPIE*, **7088**, 708805, <https://doi.org/10.1117/12.795737>.
- , R. Sharman, J. Craig, and G. Blackburn, 2008b: Remote detection and diagnosis of thunderstorm turbulence. *Proc. SPIE*, **7088**, 708804, <https://doi.org/10.1117/12.795570>.
- Wilson, J., and D. Meegenhardt, 1997: Thunderstorm initiation, organization, and lifetime associated with Florida boundary layer convergence lines. *Mon. Wea. Rev.*, **125**, 1507–1525, [https://doi.org/10.1175/1520-0493\(1997\)125<1507:TIOALA>2.0.CO;2](https://doi.org/10.1175/1520-0493(1997)125<1507:TIOALA>2.0.CO;2).
- Wolfson, M., R. Delanoy, B. Forman, R. Hallowell, M. Pawlak, and P. Smith, 1994: Automated microburst wind-shear prediction. *Lincoln Lab. J.*, **7** (2), 399–426.
- Zou, H., and T. Hastie, 2005: Regularization and variable selection via the elastic net. *J. Roy. Stat. Soc. Series B Stat. Methodol.*, **67**, 301–320, <https://doi.org/10.1111/j.1467-9868.2005.00503.x>.