Machine Learning for Real-Time Prediction of Damaging Straight-Line Convective Wind

RYAN LAGERQUIST

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

AMY MCGOVERN

University of Oklahoma, Norman, Oklahoma

TRAVIS SMITH

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

(Manuscript received 20 March 2017, in final form 3 November 2017)

ABSTRACT

Thunderstorms in the United States cause over 100 deaths and $10 billion (U.S. dollars) in damage per year, much of which is attributable to straight-line (nontornadic) wind. This paper describes a machine-learning system that forecasts the probability of damaging straight-line wind (≥50 kt or 25.7 m s^{-1}) for each storm cell in the continental United States, at distances up to 10 km outside the storm cell and lead times up to 90 min. Predictors are based on radar scans of the storm cell, storm motion, storm shape, and soundings of the near-storm environment. Verification data come from weather stations and quality-controlled storm reports. The system performs very well on independent testing data. The area under the receiver operating characteristic (ROC) curve ranges from 0.88 to 0.95, the critical success index (CSI) ranges from 0.27 to 0.91, and the Brier skill score (BSS) ranges from 0.19 to 0.65 (>0 is better than climatology). For all three scores, the best value occurs for the smallest distance (inside storm cell) and/or lead time (0–15 min), while the worst value occurs for the greatest distance (5–10 km outside storm cell) and/or lead time (60–90 min). The system was deployed during the 2017 Hazardous Weather Testbed.

1. Introduction

a. Damaging straight-line wind

The three types of damaging straight-line wind are downbursts, gust fronts, and bow echoes[1] (National Severe Storms Laboratory 2016b). A downburst is an area of strong wind caused by a downdraft hitting the surface and diverging horizontally. The primary mechanisms behind downbursts are evaporative cooling and precipitation drag. Downbursts are generally categorized in two ways: macroburst versus microburst and wet versus dry. A macroburst covers a horizontal area > 4 km^2; a microburst, <4 km^2. A wet downburst is accompanied by heavy precipitation, whereas a dry downburst occurs with no precipitation, because the cloud base is high enough, and the subcloud environment is dry enough, that all hydrometeors evaporate or sublimate before reaching the ground. There are intermediate cases between wet and dry. (Fujita 1990) The typical duration and highest known horizontal winds are 5–20 min and ~60 m s^{-1} for a macroburst and 2–5 min and ~75 m s^{-1} for a microburst (National Weather Service 2016b).

Environmental conditions that favor downbursts (Rose 1996) are high static instability, leading to a strong updraft and heavy precipitation; dry (highly subsaturated) air

[1] The page cited lists five types of damaging straight-line wind, including microbursts, derechoes, and haboobs. However, a microburst is a special type of downburst, a derecho is a special type of bow echo, and a haboob is a special type of gust front (one carrying a large amount of dust).

Supplemental information related to this paper is available at the Journals Online website: https://doi.org/10.1175/WAF-D-17-0038.s1.

Corresponding author: Ryan Lagerquist, ryan.lagerquist@ou.edu

DOI: 10.1175/WAF-D-17-0038.1

below the cloud base, leading to strong evaporative cooling; moist air between the surface and dry subcloud air, which exacerbates the density difference between the near-surface air and the descending evaporatively cooled air, thus accelerating the downdraft; and frozen hydrometeors in the dry subcloud layer. The ice-to-vapor phase change (either sublimation or melting followed by evaporation) requires more latent heat than evaporation alone, which cools the surrounding air more effectively, thus accelerating the downdraft.

A gust front occurs when strong horizontal winds associated with a downburst propagate away from the storm, creating a sharp boundary between evaporatively cooled air and warmer ambient air. Convergence along the gust front causes turbulence and wind shear, which are extremely hazardous to aircraft. (Delanoy and Troxel 1993)

A bow echo is a bow-shaped line of storm cells with strong wind at its leading edge. The formation of a bow echo proceeds as follows. 1) A downburst generated by a single storm cell propagates away as a gust front. The strongest wind occurs near the midpoint of the gust front, forcing it into a bow shape. 2) Convergence along the gust front leads to new convective initiation and, on a larger scale, ascending rearward flow (perpendicular to the bow echo and opposite its motion). 3) Ascending rearward flow is balanced by descending frontward flow (the "rear-inflow jet"), which reaches the surface just behind the gust front, causing more downbursts and strengthening the gust front.

Finally, a derecho is a particularly strong bow echo or series thereof. For the precise definition of a derecho (both traditional and proposed new versions), see Corfidi et al. (2016). Derechoes have a typical duration of ~10 h, and horizontal winds can reach ~60 m s^{-1} (Storm Prediction Center 2016a). Favorable conditions for derechoes are difficult to summarize because they occur in a wide variety of environments (Coniglio et al. 2004). In general, conditions that favor downbursts also favor derechoes.

b. Current forecasting methods

Current forecasting methods focus on properties of the near-storm environment (NSE), radar signatures, and explicit wind forecasts from numerical weather prediction (NWP).

NSE properties are mainly sounding indices (single-number summaries of the vertical profile), such as the derecho composite parameter (DCP), microburst composite parameter (MCP), and wind-damage parameter (WDP). In situ soundings are very rare [less than 100 sites in the continental United States (CONUS), launching every 12 h], so these indices are usually calculated from NWP forecasts. The DCP (Storm Prediction Center 2017a; Evans and Doswell 2001) is a linear combination of downdraft convective available potential energy (CAPE), related to the potential for cold-pool development; most-unstable CAPE, related to the ability to sustain strong storms at the leading edge of a gust front; 0–6-km wind shear, related to the potential for ensuing convection to become organized; and 0–6-km wind speed, related to the potential for convective initiation downstream along the gust front. When DCP > 2, an existing mesoscale convective system (MCS) is quite likely to produce a derecho. The MCP and WDP are based on similar indices, related to the static instability and ambient wind speed of the environment. When MCP ≥ 9, microbursts are deemed "likely"; when WDP ≥ 1, there is an "enhanced" risk for damaging gust fronts from multicell clusters. (Storm Prediction Center 2017b; Blumberg et al. 2017b)

Radar-based forecasting almost always uses the current radar scan, since operational NWP models do not have the required resolution (<1 km) to depict signatures associated with damaging straight-line wind. In general, forecasters look for signatures of divergence (may indicate a downdraft hitting the surface) or large radial velocities near the surface (may indicate strong wind just below). Within the context of an MCS, forecasters often look for a strong rear-inflow jet and mesovortices. Sometimes explicit NWP forecasts (i.e., the raw wind speed output) are used as well.

The main disadvantage of sounding indices is that they are not explicit probabilities (between 0 and 1). The main disadvantage of radar is that it offers little to no lead time, because it depicts only the current situation. Also, the radar beam is usually well above the surface, where the measured radial velocity may not be strongly related to the surface wind speed. Finally, the main disadvantage of NWP models is that, even if they accurately predict the hazards associated with a given thunderstorm, they have significant errors in storm timing and location (Weisman et al. 2008; Sun et al. 2014; Gagne et al. 2015). Sometimes an existing storm cell can be matched to one in the NWP forecast, which allows NWP guidance to be used, but this is not always possible (Gagne et al. 2015). Also, errors in storm timing and location may lead to errors in the associated hazards, as a result of temporal and spatial differences in the NSE.

c. Machine learning

Machine learning (ML) allows computers to discover knowledge that has not been explicitly programmed. The goal is usually to predict one target variable y, given many predictor variables x_j. ML has many advantages over both human reasoning and NWP. For one, ML predictions are deterministic: given the same inputs x_j, an

ML model will always predict the same outcome *y*. Second, prediction time is usually very fast ($\ll$1 s to predict a new example). Third, an ML model can easily incorporate many diverse data sources (e.g., single- and dual-polarization radar, satellite, NWP output, in situ observations), which is much more difficult for an NWP model (Marriott 2012). Fourth, ML can predict the outcome of a physical process without fully understanding it. Finally, relationships learned by the model, as well as variable selection and transformation (Webb 2003, chapter 9), can be used to gain an understanding of the physical process.

Despite many applications of ML in convective meteorology [some of which are reviewed by McGovern et al. (2017)], to our knowledge only four other groups have used ML to forecast damaging straight-line convective wind. Kitzmiller et al. (1995) used radar-derived vertically integrated liquid to forecast the probability of "any severe weather" (damaging straight-line wind, tornado, or hail) on a grid within the next 20 min. Marzban and Stumpf (1998) used 23 radar-derived predictors to forecast the probability of damaging wind (straight line or tornadic) for a single storm cell. Alexiuk et al. (1999) used 22 radar-derived features to classify severe storms by their primary hazard type (straight-line wind, tornado, hail, or heavy rain). Finally, Cintineo et al. (2014) used five features (derived from radar, satellite, and NWP) to forecast the probability of "any severe weather" for a single storm cell.

We use ML to predict the probability that a storm cell will produce damaging straight-line wind, using only data available in real time. We adopt the National Weather Service (2010) definition of "damaging wind" as a gust $\geq$ 50 kt (25.7 m s^{-1}). Forecasts are made for three disjoint distance buffers (Fig. 1a) and five time windows (Fig. 1b). Our hypothesis was that, for each distance buffer and time window, we could produce unbiased forecasts that outperform climatology. Our ML system was used in real time by human forecasters during the 2017 Hazardous Weather Testbed (Clark et al. 2012).

Section 2 describes our input data and processing thereof. Section 3 describes our ML methods. Section 4 describes an experiment to optimize ML parameters, the results of which are discussed in section 5. Section 6 summarizes our findings and suggests future work.

2. Input data

a. Data sources

We use three types of input data (Table 1). Radar images from the Multiyear Reanalysis of Remotely Sensed Storms (MYRORSS; Ortega et al. 2012), as well as soundings from the Rapid Update Cycle (RUC; Benjamin et al. 2004) and the North American Regional Reanalysis (NARR; Mesinger et al. 2006), are used to create predictors for the "event" (storm-maximum wind $\geq$ 50 kt). Near-surface wind observations from the Meteorological Assimilation Data Ingest System (MADIS; McNitt et al. 2008), the Oklahoma Mesonet (McPherson et al. 2007), 1-min METARs (National Climatic Data Center 2006), and *Storm Events* (National Weather Service 2016a) are used to determine when and where the event occurred.

MYRORSS contains merged, quality-controlled data from all 143 Next-Generation Weather Radar (NEXRAD) sites in the CONUS. Merging and quality control are done by the Warning Decision Support System–Integrated Information (WDSS-II; Lakshmanan et al. 2007), which is a software package for the analysis and visualization of thunderstorm-related data. Using methods described in Lakshmanan et al. (2006), WDSS-II generates 47 variables on a 0.01° grid. These variables include reflectivity at 35 height levels (from 0 to 20 km above ground level), reflectivity at three temperature levels (−20°, −10°, and 0°C), and nine composite variables (integrating data from different heights). Composite variables are listed in Table 2.

We do not use the 35 height-level variables, because we assume that they would be mostly redundant with the temperature-level and composite variables. Also, this would nearly triple the number of predictors for ML (see discussion of radar statistics in section 2e), which would make computing requirements prohibitive.

Soundings are used to characterize the NSE, which partly dictates how a storm will evolve. We use modeled soundings because, as mentioned in section 1b, observed soundings are very sparse. The Rapid Refresh (RAP) model is widely used to create NSE datasets, such as the Storm Prediction Center (2016b) mesoanalysis. However, the RAP model did not exist during the development period (2004–11, as discussed in section 2b), so we use its predecessor, RUC. We use only 0-h analyses, because forecasts with longer lead times are usually not archived. For time steps with no RUC data, we use the NARR as a substitute.

We use wind observations from all surface-based weather stations in MADIS, which to our knowledge is the world's largest collection of station data, for model training and evaluation. We also use wind observations from the Oklahoma Mesonet and 1-min METARs, which both are of high quality and temporal resolution and are not included in MADIS. However, even with these three datasets, we achieved poor model performance because there were not enough events (storm-maximum winds $\geq$ 50 kt). Thus, we also use straight-line wind reports from *Storm Events*, all of which are measured or estimated at $\geq$50 kt. Figure 2 shows an example of the wind from all four datasets.

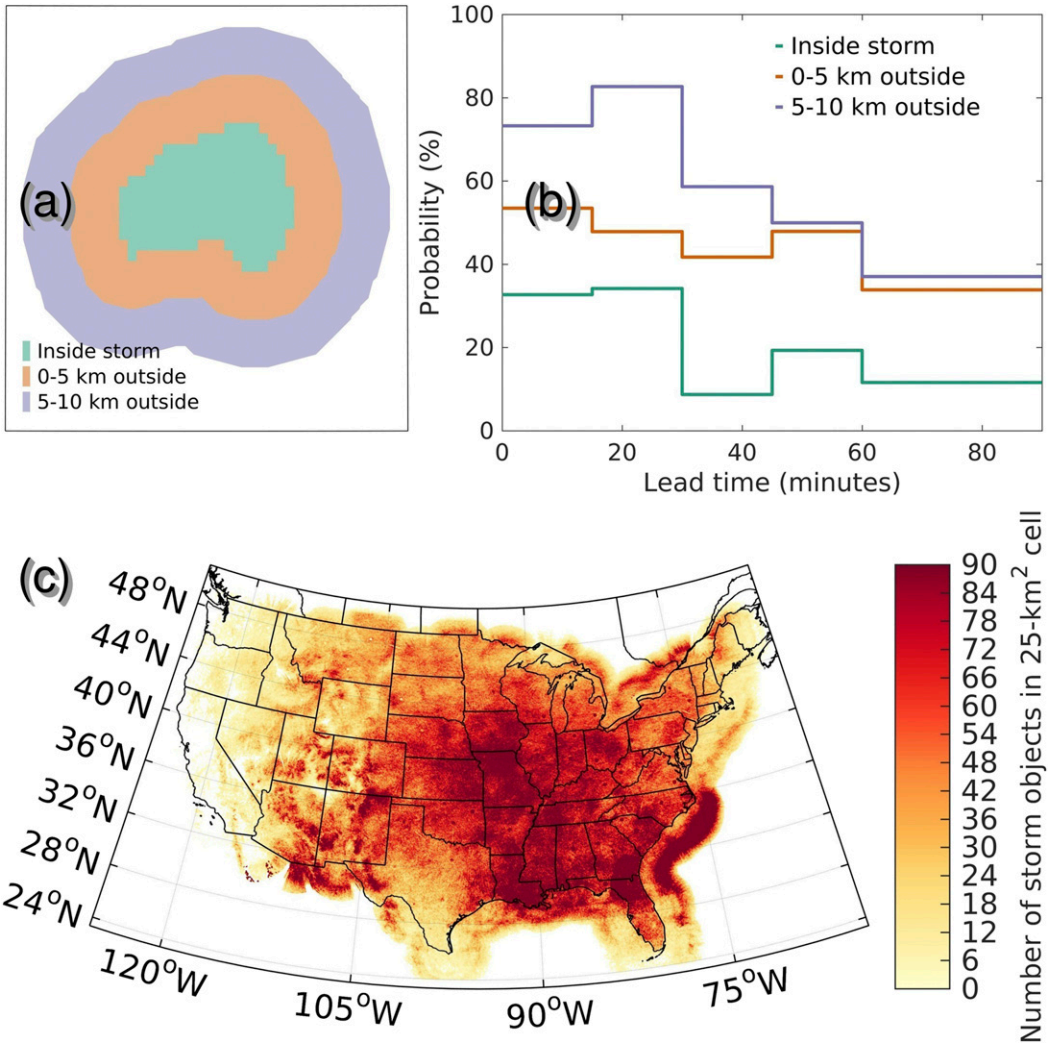

FIG. 1. Forecasts are made for (a) three disjoint distance buffers and (b) five time windows. (c) Spatial distribution of storm objects used. Counts are in 5 km × 5 km grid cells. A "storm object" is one storm cell at one 5-min time step. This should not be interpreted as a climatology, because it includes only 804 selected days.

Storm Events is a database of severe weather events, many of which are based on human reports. There are several problems with the database, one of which is that humans tend to overestimate wind speed. It is the sole responsibility of the NWS office taking the report to determine if the wind speed was *actually* ≥50 kt, and they cannot always correct for this overestimation. Second, report density increases with population density, which varies in both space and time. Third, the NWS's archiving and quality-control practices have changed over time. All

TABLE 1. Data sources. Spatial coverage of the Oklahoma Mesonet is the state of Oklahoma; all other datasets cover the CONUS.

Data type	Source	Resolution	Time period
Radar images	MYRORSS	0.01° (~1 km), 5 min	2000–11 (excluding 2009)
Model soundings	RUC	13 or 20 km, 1 h	April 1994–April 2012
	NARR	32 km, 3 h	1979–present
Near-surface wind observations	MADIS	Variable	July 2001–present
	OK Mesonet	Spatially variable, 5 min	1994–present
	1-min METARS	Spatially variable, 1 min	2000–present
	Storm Events	Variable	1955–present

TABLE 2. Radar statistics. Each statistic is computed for each variable, based only on values inside the storm object.

Variable	Spatial statistics
Low-level (0–2 km) azimuthal shear	0th percentile (min)
Midlevel (3–6 km) azimuthal shear	5th percentile
18-dBZ echo top	25th percentile (first quartile)
50-dBZ echo top	50th percentile (median)
Max estimated hail size (MESH)	75th percentile (third quartile)
−20°C reflectivity	95th percentile
−10°C reflectivity	100th percentile (max)
0°C reflectivity	Mean (first moment)
Composite (column max) reflectivity	Std dev (related to second moment)
Lowest-altitude reflectivity	Skewness (related to third moment)
Severe hail index (SHI)	Kurtosis (related to fourth moment)
Vertically integrated liquid (VIL)	

these issues are discussed in detail in Doswell et al. (2005). Nonetheless, *Storm Events* is the most consistent and complete severe weather event database available (Ashley and Black 2008), and we needed these reports to supplement the station data.

We assume that all wind observations are straight line (nontornadic), first because *Storm Events* distinguishes between the two, so presumably no tornadoes are reported as straight-line wind. Second, tornadoes are much less common (National Severe Storms Laboratory 2016a), and usually much more intense, than damaging straight-line wind. Thus, very few tornadoes hit weather stations, and those that do often destroy the anemometer.

b. Training and testing period

The datasets used (Table 1) have a common period of July 2001–December 2011, excluding 2009. However, processing all ~3500 days would have been computationally prohibitive. Thus, we filter the dataset by removing years 2001–03, for which there are many fewer wind observations; days with <30 straight-line wind reports in *Storm Events*; and days with more than five time steps of missing radar data. This leaves 804 days for training and testing, with a total of 19 951 072 storm objects (Fig. 1c). Most of the 804 days are in May–August, with a minimum of 9 in December (see Fig. A1 in the online supplement to this paper).

By using only days with ≥30 straight-line wind reports, we ensure a large number of events (storm-maximum winds ≥ 50 kt). However, there are also many nonevents, because severe-wind-producing storms are usually confined to a small area and time period.

c. Storm detection and tracking

Storm detection (the outlining of storm objects in the radar image) is performed by segmotion (Lakshmanan and Smith 2010), an algorithm in the WDSS-II package. The segmotion routine uses extended-watershed image segmentation (Lakshmanan et al. 2009) to separate the image into locally maximum areas of a certain variable. In addition, segmotion has three main input arguments: the tracking variable, threshold for the tracking variable, and minimum storm area. We use −10°C reflectivity with a 30-dBZ threshold, following Saxen (2002) and

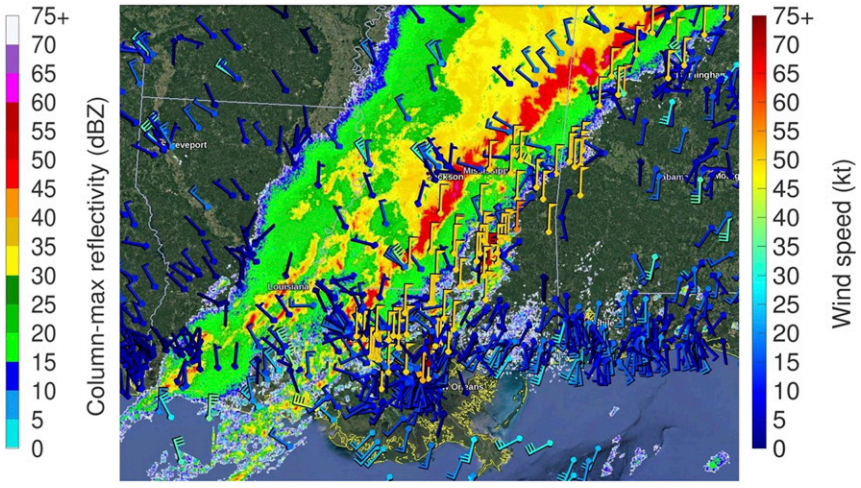

FIG. 2. Wind observations and radar mosaic. The color fill is column-maximum reflectivity at 2300:11 UTC 4 Apr 2011, and wind barbs show the maximum gust at each location over the next 90 min. Reports in *Storm Events* contain no direction, so this is depicted as due north when plotting.

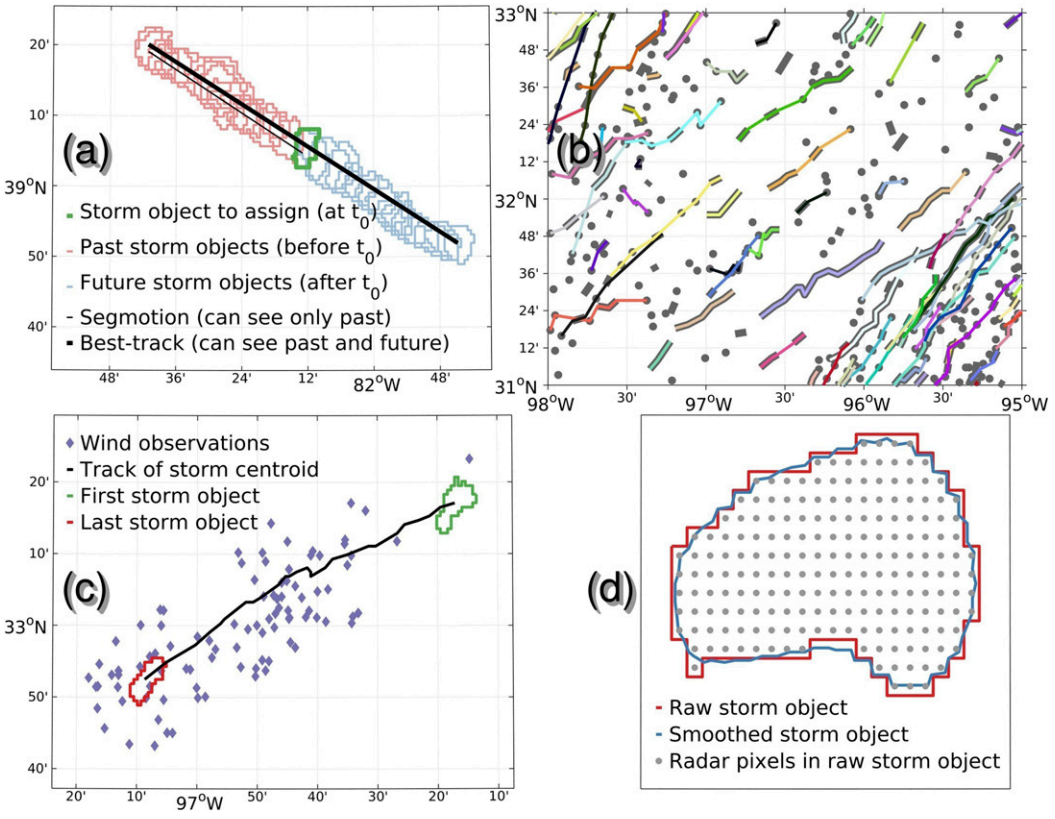

FIG. 3. (a) Difference between tracking methods. (b) Difference between tracking results. Thick gray lines underneath (thin multicolor lines on top) are segmotion (besttrack) results for a 24-h period. (c) Sample linkage output for 5–10-km distance buffer. (d) Sample storm object.

colleagues at the Cooperative Institute for Mesoscale Meteorological Studies, and a minimum area of 50 km^2. We found that smaller thresholds led to many false detections (nonthunderstorms being outlined as thunderstorms), while larger thresholds often led to many thunderstorms being merged into one object.

Tracking is done in two stages: real time and postevent. Real-time tracking is done by segmotion, which uses *K*-means clustering and assigns storm objects in the same cluster to the same track. Postevent tracking is done by the besttrack algorithm (also part of WDSS-II) (Lakshmanan et al. 2015), which assigns each storm object to its best-fitting Theil–Sen trajectory. The main difference is that to determine the track membership of a storm object at time t_0, segmotion can use only past information (data valid before t_0), whereas besttrack can use both past and future information. (See Fig. 3a.)

Tracks created by segmotion are corrected by besttrack. The main effect is that broken (incorrectly truncated) tracks are joined. This results in longer tracks (Fig. 3b), which allows storms and wind observations to be linked (section 2d) at greater lead times, which allows the ML models to forecast severe wind at greater lead times.

d. Linking storms and wind observations

For each wind observation *W* and distance buffer *D* (inside storm, 0–5 km outside, and 5–10 km outside), the procedure is as follows. 1) If *W* is not contemporaneous with a radar scan, interpolate storm objects along their respective tracks to the time of *W*. 2) Find the storm object *S* with the nearest edge. Let the corresponding track be *S**. 3) If *W* is in buffer *D* around storm object *S*, link *W* to storm track S^*. Otherwise, do not link *W* to any storm.

Sample output is shown in Fig. 3c. Linkages created by this procedure are used in section 2f to label storm objects (1, if responsible for severe wind; 0, otherwise.)

e. Calculation of predictors

There are four types of predictors, calculated for each storm object: radar statistics, storm motion, shape parameters, and sounding indices.

First, 11 statistics are calculated for each of the 12 radar variables (Table 2). These are spatial statistics, based on values inside the storm object (Fig. 3d). The same statistics are calculated for both raw values and

TABLE 3. NWP models used for sounding data. Heights listed are 2 and 10 m above ground level; 1 mb = 1 hPa.

Model	Pressure level	Variable at each pressure level	Near-surface variable
RUC	[100, 1000] mb at 25-mb intervals	Temp	2-m temp
		Relative humidity	2-m relative humidity
		Geopotential height	Surface pressure
		u wind	10-m u wind
		v wind	10-m v wind
NARR	[100, 300] mb at 25-mb intervals	Temp	2-m temp
	[350, 700] mb at 50-mb intervals	Specific humidity	2-m specific humidity
	[725, 1000] mb at 25-mb intervals	Geopotential height	Surface pressure
		u wind	10-m u wind
		v wind	10-m v wind

gradient magnitudes [$\sqrt{(\partial w/\partial x)^2 + (\partial w/\partial y)^2}$, where w is the radar variable].

Second, storm motion (speed and direction) is calculated by backward differencing [$(\mathbf{r}_n - \mathbf{r}_{n-1})/(t_n - t_{n-1})$, where $\mathbf{r}_n$ and $\mathbf{r}_{n-1}$ are the locations of the storm center at times t_n and t_{n-1}].

Third, the following shape parameters are calculated from the storm outline: (i) area; (ii) orientation of the best-fitting ellipse, which is an angle from 0° to 180°; (iii) eccentricity of the best-fitting ellipse; (iv) solidity (fraction of pixels in bounding polygon that are also in convex hull); (v) extent (fraction of pixels in bounding polygon that are also in smallest bounding rectangle); (vi) curvature (mean absolute curvature over all vertices); (vii) bending energy (sum of squared curvatures/perimeter); and (viii) compactness (perimeter^2/4$\pi \times$ area). Curvature, bending energy, and compactness are based on the smoothed polygon (Fig. 3d), because 90° angles in the raw polygon lead to unrealistically large curvature and perimeter values. All other shape parameters are based on the raw polygon.

Fourth, the RUC sounding is interpolated to the time and center of the storm object. Spatial interpolation is done by the nearest-neighbor method, and temporal interpolation is done by the previous-neighbor method (most recent RUC time step). This ensures that the whole sounding comes from the same grid cell and time step, which preserves physical consistency among the sounding variables (listed in Table 3). If the most recent model time step is from the NARR (which occurs when RUC data are missing), the NARR sounding is used instead. Only 103 of 20 019 RUC hours are missing, so this happens very infrequently.[2] After interpolating model data, the Sounding and Hodograph Analysis and Research Program in Python (SHARPpy; Blumberg et al. 2017a) is used to calculate the 97 sounding indices listed in appendix A in the online supplement.

All vectors (e.g., wind velocities, wind shears, storm motion) are decomposed into the magnitude, sine of direction, and cosine of direction. This results in 431 predictors: 264 radar statistics, three components of storm motion, nine shape parameters (orientation consists of sine and cosine only, no magnitude), and 155 sounding indices.

f. Calculation of labels

The following procedure (Fig. 4) is repeated for each storm cell, distance buffer, and time window [$t_{\min}$, $t_{\max}$]. (i) Find all wind observations linked to storm S with distance buffer D, occurring between $t_{\min}$ and $t_{\max}$. (ii) If the maximum wind observation is ≥50 kt, the label is 1; otherwise, the label is 0. This label, which indicates whether or not the storm is responsible for severe wind, is the target variable y for ML models.

g. 2017 Hazardous Weather Testbed

The Hazardous Weather Testbed (HWT) is an annual experiment managed by the SPC, the NSSL, and the NWS Forecast Office in Norman, Oklahoma. Human forecasters from the United States and abroad gather for several weeks to test new forecasting methods and technologies for convective hazards. The goal of the HWT is to accelerate the transition of research to operations. Forecasts produced by our ML system were updated every ~2 min (pursuant to new radar data) and made available ~4 min later. The combined latency time (of our forecast updates and the radar data triggering the updates) was ~6 min. Output was displayed as a probability–time graph for each storm cell (e.g., Fig. 1b). These graphs were accessed by clicking on the relevant storm cell in the probabilistic pazard information (PHI; Karstens et al. 2014) tool.

[2] In the future we will discontinue the use of NARR data. In general, using different data sources to create the same predictor variable worsens the ML performance. Since NARR data were used very infrequently in our dataset, the effect appears to have been minimal.

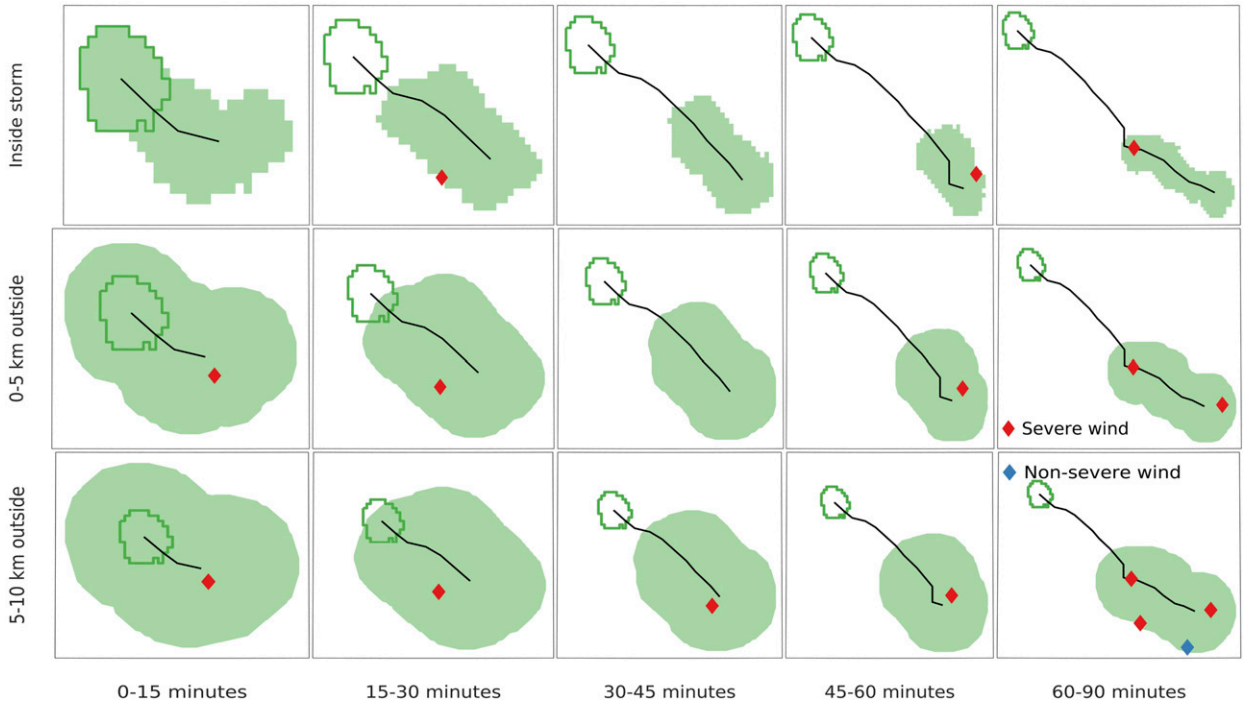

FIG. 4. Labeling procedure. The dark green polygon is the storm object S to be labeled, occurring at time t_0. The light green "blob" is the area covered by the relevant distance buffer D around S, over the relevant time window (from t_{min} to t_{max}). Diamonds are wind observations linked to S with distance buffer D, occurring between t_{min} and t_{max}.

Radar statistics (section 2e) were computed from Multi-Radar Multi-Sensor (MRMS) (Smith et al. 2016) data, rather than MYRORSS. MRMS contains the same variables on a 0.01° or 0.005° grid, but with less quality control since it must be available in real time. Sounding indices (section 2e) were computed from the latest RAP data, instead of RUC or NARR. Ideally, we would have trained models with the same data sources, but the MRMS archive was too short (about 1 yr, with considerable missing data). As in section 2c, real-time tracking was done by segmotion, and postevent tracking (only to verify forecasts after HWT) was done by besttrack.

3. Machine learning

a. Base models

1) LOGISTIC REGRESSION

Logistic regression (LR; Walker and Duncan 1967) fits a logit curve to the training data:

$$f_i = \frac{\exp\left(-\beta_0 - \sum_{j=1}^{N} \beta_j x_{ij}\right)}{1 + \exp\left(-\beta_0 - \sum_{j=1}^{N} \beta_j x_{ij}\right)}, \quad (1)$$

where f_i is the forecast probability of severe wind for the ith storm object, x_{ij} is the value of the jth predictor for the ith storm object, β_0 is the bias term, and β_j is the coefficient (weight) for the jth predictor. In addition, N is the number of predictors (431). Note as well that $f_i \in [0, 1]$, which is one reason that LR is often used to forecast probabilities.

The training algorithm [usually gradient descent; see section 4.4.3 of Mitchell (1997)] looks for weights β_j that minimize deviance:

$$D = -\frac{1}{M} \sum_{i=1}^{M} [y_i \log_2(f_i) + (1 - y_i) \log_2(1 - f_i)], \quad (2)$$

where y_i is the true label (section 2f) and f_i is the forecast probability of severe wind, for the ith storm object. In addition, M is the number of storm objects in the training set. Deviance is zero for a perfect forecast ($f_i = y_i \forall i$) and increases with the summed differences between f_i and y_i.

2) LOGISTIC REGRESSION WITH AN ELASTIC NET

Logistic regression with an elastic net (LREN; Zou and Hastie 2005) is equivalent to basic LR, except that it has a different cost function, called the penalty:

$$P = D + \lambda \sum_{j=0}^{N} \left[\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j| \right], \quad (3)$$

where D, M, and β_j are as in Eqs. (1) and (2). Here, $\lambda \in [0, \infty)$ is the regularization parameter, and $\alpha \in [0, 1]$ is the elastic-net parameter, both of which are user determined. As λ increases, the coefficient magnitudes are penalized more, which encourages the model to produce smaller coefficients. Note that α determines the balance between the L_1 penalty $\left(\sum_{j=0}^{N}|\beta_j|\right)$ and the L_2 penalty $\left(\sum_{j=0}^{N}\beta_j^2\right)$. For $\alpha = 0$, LREN simplifies to ridge regression (Hoerl and Kennard 1988). For $\alpha = 1$, LREN simplifies to lasso optimization (Tibshirani 1996). The main difference is that lasso optimization allows coefficients to be zeroed out, which explicitly removes variables from the model.

3) FEED-FORWARD NEURAL NETS

A feed-forward neural net (FFNN; Haykin 2001) contains several layers of neurons, with connections between neurons in adjacent layers (Fig. 5a). Specifically, each neuron in the kth layer is connected to all those in the $(k-1)$st and $(k+1)$st layers. Each neuron in the kth layer computes a weighted sum of its inputs from the $(k-1)$st layer, called the activation $a_j^{(k)}$:

$$a_j^{(k)} = w_{0j}^{(k-1)} + \sum_{i=1}^{N^{(k-1)}} w_{ij}^{(k-1)} z_i^{(k-1)}, \quad (4)$$

where $w_{ij}^{(k-1)}$ is the weight from the ith neuron in the $(k-1)$st layer to the jth neuron in the kth layer, $z_i^{(k-1)}$ is the output [see Eq. (5)] from the ith neuron in the $(k-1)$st layer, and $N^{(k-1)}$ is the number of neurons in the $(k-1)$st layer.

The "output" $z_j^{(k)}$ of each neuron in the kth layer is passed to neurons in the $(k+1)$st layer:

$$z_j^{(k)} = g^{(k)}(a_j^{(k)}), \quad (5)$$

where $g^{(k)}$ is the activation function for the kth layer, usually a generalized version of the step function (Fig. 6a). In our case, $g^{(k)}$ is always "tansig" for the hidden layers and "softmax" for the output layer (Fig. 6a).

We allow all predictor values to simply "pass through" the input layer. In other words, if the input layer is layer 1, $z_j^{(1)} = x_j$. This configuration, as well as the use of tansig and softmax functions, is the default in MATLAB's patternnet [MathWorks (2016); used for the experiment described in section 4].

4) DECISION TREES

A decision tree (Mitchell 1997, chapter 3) contains several layers of nodes (Fig. 5b), each classified as either a branch node or a leaf node. A branch node sends each example (storm object) down one of two branches, based on its answer to a yes-or-no question. Each question involves a predictor x_j and threshold x_{jk}, known collectively as the split point. A leaf node n^* predicts the target variable (probability of severe wind) for a new example, based on training examples that reached n^*.

During training, at each branch node, the algorithm loops through many possible split points and chooses the one leading to the greatest reduction in deviance.

5) RANDOM FORESTS

Single decision trees are often unstable, because each split point is a hard threshold involving only one variable. For example, suppose that two storm objects (S_1 and S_2) have maximum reflectivity of 70 dBZ, downdraft CAPE of 1200 J kg^{-1}, and storm motion of 20 m s^{-1}. However, S_1 has CAPE of 1499 J kg^{-1}, and S_2 has a value of 1501 J kg^{-1}. The decision tree shown in Fig. 5b would produce very different forecasts for S_1 and S_2 (5% and 60%, respectively), even though they are essentially the same. This instability causes decision trees to overfit the training data, thus generalizing poorly to new examples.

Random forests (Breiman 2001) alleviate this problem by ensembling decision trees. If the trees are diverse enough, they should overfit in different ways, so that their biases are offsetting. A random forest maintains this diversity by tree bagging, where only a subset of examples (storm objects) is used to train each tree, and feature bagging, where a subset of predictor variables is tried at each split point.

6) GRADIENT-BOOSTED TREE ENSEMBLES

Gradient boosting (Friedman 2001) is another way to ensemble decision trees. The main difference is that gradient boosting creates an additive ensemble, where the kth tree is fit to the residual from the first $(k-1)$ trees. Conversely, each tree in a random forest is fit to the same target variable (label from section 2f), and forecasts from individual trees are ensembled after training. Thus, a random forest may be trained in parallel, whereas a gradient-boosted ensemble (GBE) must be trained in series.

b. *Calibration of base models*

Calibration is necessary because most ML models produce unreliable forecasts (Niculescu-Mizil and Caruana 2005, hereafter NMC05). Reliability (REL) is defined below [cf. Eq. (2) in Hsu and Murphy (1986)], where forecast probabilities are discretized into K (usually 10) bins:

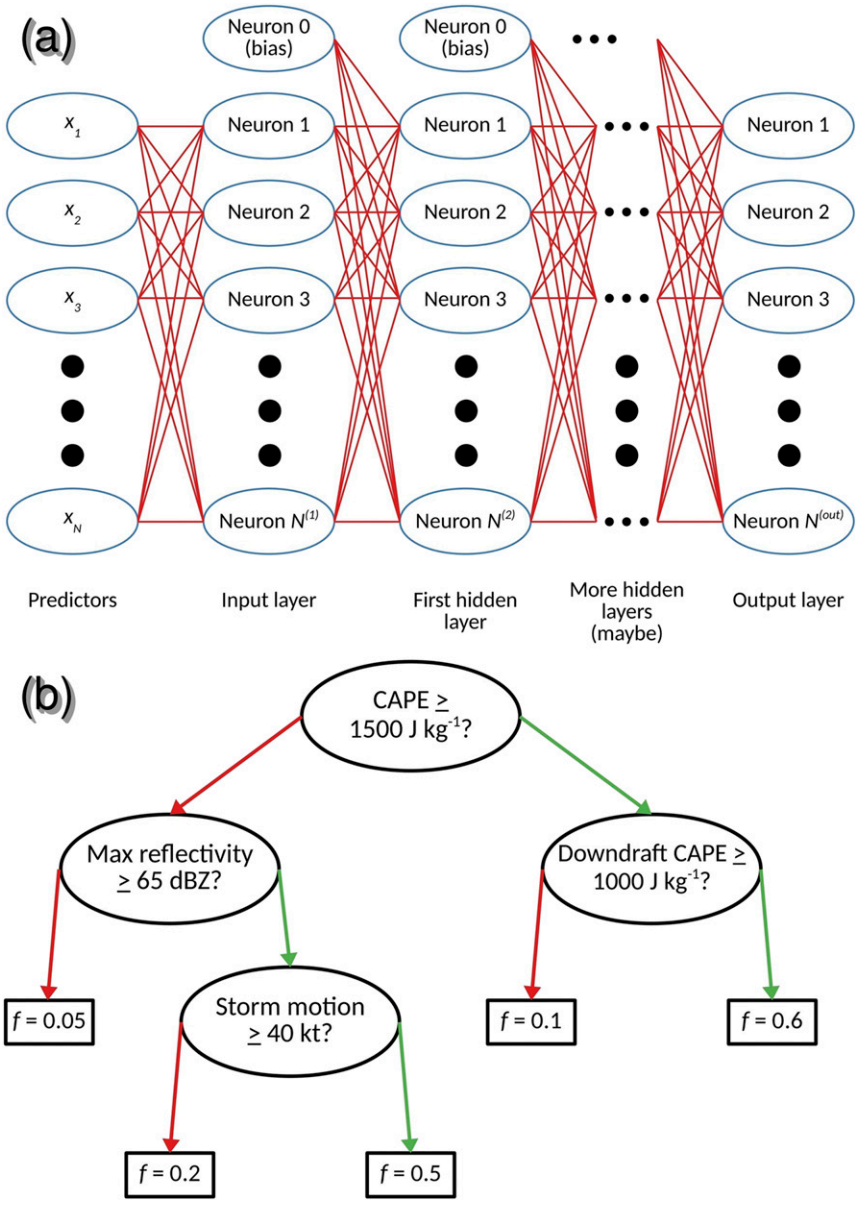

FIG. 5. (a) FFNN schematic. In the left column, each blue ellipse is a predictor variable. In all other columns, each blue ellipse is a neuron. Each red line is associated with a weight $w_{ij}^{(k-1)}$ in Eq. (4). (b) Decision-tree schematic. At each branch node (ellipse), an if–else condition is applied to storm object S. If true, S is sent down the right branch; if false, down the left branch. When S reaches a leaf node n^* (rectangle), it is given a forecast, based on training examples that reached n^*.

$$\mathrm{REL} = \frac{1}{M}\sum_{k=1}^{K} M_k (f_k - \overline{y}_k)^2, \tag{6}$$

where M is the total number of storm objects, and M_k is the number in the kth bin. For the kth bin, f_k is the mean forecast and $\overline{y}_k$ is the mean label (''conditional event frequency''). REL $\in [0, 1]$, and REL $= 0$ for a perfect model ($f_k = \overline{y}_k \forall k$).

We considered two calibration methods, which appear to be the most common: Platt scaling (Platt 2000) and isotonic regression (NMC05). Platt scaling is a sigmoid transformation of the base-model probabilities, so it works best when the base model has a sigmoid-shaped reliability curve (Fig. 6b). Isotonic regression is more general and can correct other types of poor reliability. According to NMC05, only one of the six models in

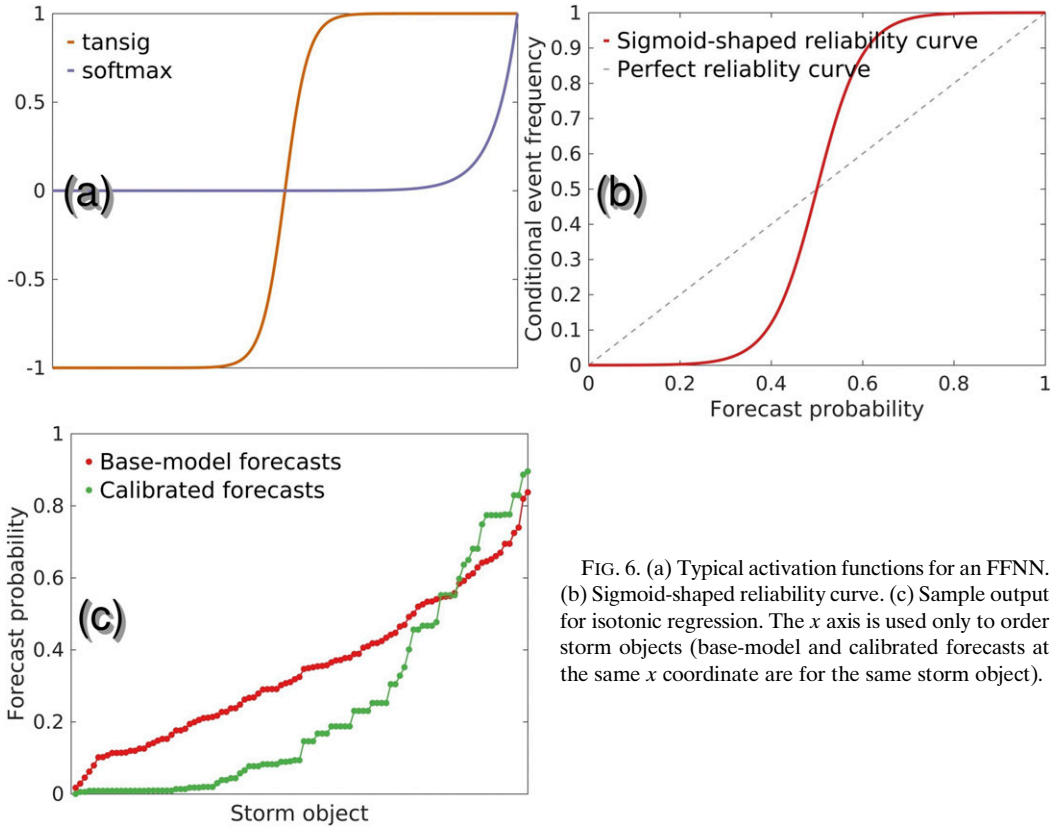

FIG. 6. (a) Typical activation functions for an FFNN. (b) Sigmoid-shaped reliability curve. (c) Sample output for isotonic regression. The x axis is used only to order storm objects (base-model and calibrated forecasts at the same x coordinate are for the same storm object).

section 3a (GBE) typically produces a sigmoid-shaped reliability curve. However, isotonic regression appears to outperform Platt scaling even in this case (their Figs. 2 and 3). Thus, we decided to use only isotonic regression.

Isotonic regression produces a mapping from each range of base-model probabilities $[\hat{f}_k, \hat{f}_{k+1})$ to the calibrated probability f_k. The training algorithm minimizes the Brier score (BS), subject to the constraint that $f_A \geq f_B$ if $\hat{f}_A \geq \hat{f}_B$ for any two storm objects A and B. In other words, isotonic regression preserves the forecast ranking, as shown in Fig. 6c. The variables related to BS are as in Eq. (2):

$$\mathrm{BS} = \frac{1}{M}\sum_{i=1}^{M}(y_i - f_i)^2 . \tag{7}$$

c. *Subsampling*

Before machine learning, we subsample the dataset. This is done for three reasons. First, the full dataset contains ~20 million storm objects (examples). Training with all 20 million examples would have taken too much memory (≫64 GB) and computing time (days for a single model). Second, the fraction of storms linked to severe wind is very small (Figs. 7a,b). In general, ML models do not learn effectively when the event of interest is so rare (Batista et al. 2004). Third, this distribution is not a good representation of the true distribution, because many storm objects are linked to only a few wind observations, which makes it plausible that they produced severe winds that were simply unobserved.

Subsampling methods are described below. Each method is run independently for each distance buffer and lead-time window $[t_{\min}, t_{\max}]$.

1) TRAINING THE BASE MODEL

We sample uniformly from the distribution of storm-maximum wind $U_{\max}$, which allows the model to learn from storms with all wind intensities. Specifically, we draw an equal number of storm objects from seven categories: 0–10, 10–20, 20–30, 30–40, 40–50, and ≥50 kt (where 1 kt = 0.51 m s^{-1}), as well as $U_{\max}$, which represents undefined results (because the storm no longer exists after $t_{\min}$). The resulting distribution is shown in Fig. 7c. Without the last category, the model would suffer from "survivor bias"; that is, it would be trained only with longer-lived storms, which tend to be stronger and more likely to produce severe wind. This would cause the model to overpredict severe wind for all time windows other than 0–15 min.

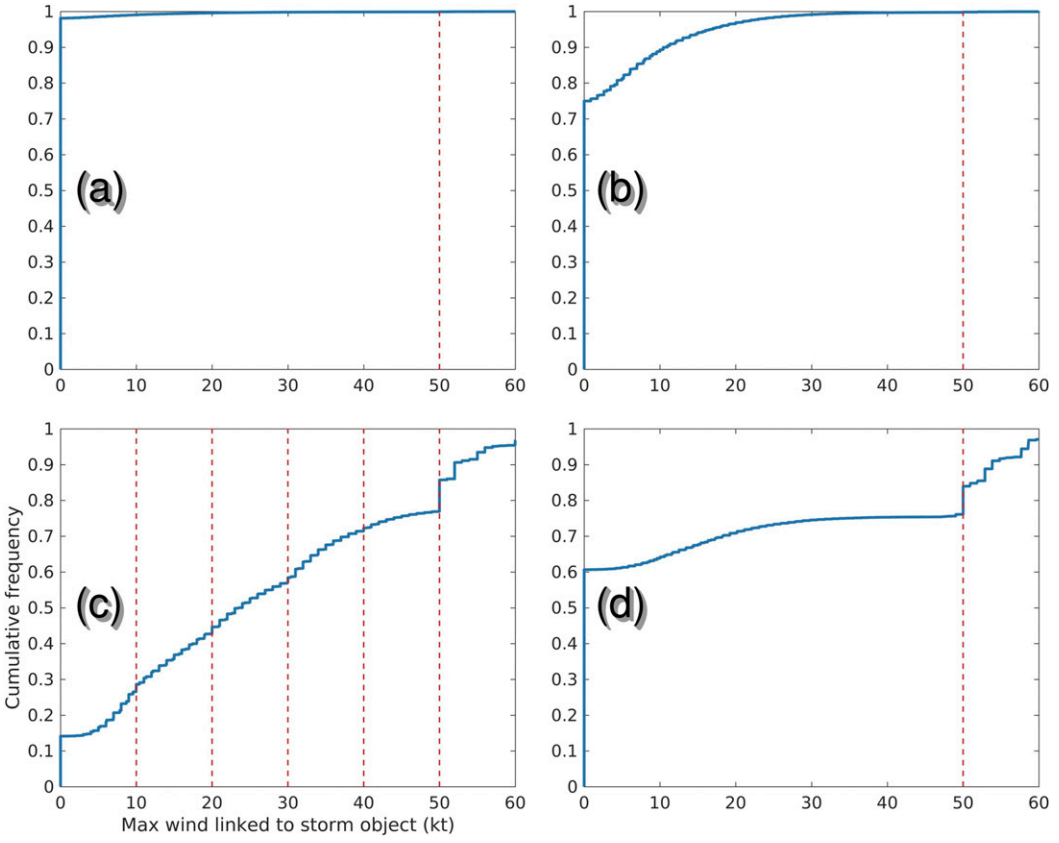

FIG. 7. Cumulative density functions (CDFs) of maximum wind linked to a storm object. (a) Full dataset at 0-km buffer distance (inside storm) and 60–90-min lead time. (b) Full dataset at 5–10-km buffer distance and 0–15-min lead time. (c) Uniform distribution and (d) best-observed distribution at 0–5-km buffer distance and 30–45-min lead time. In all cases, if a storm object has no wind observations in the relevant distance buffer and time window, its maximum wind is considered zero.

2) CALIBRATING THE BASE MODEL

The calibration set should have a U_{max} distribution similar to the true distribution. Thus, we use the ''best-observed distribution,'' which contains all storm objects meeting one of two criteria: either 1) $U_{max} \geq 50$ kt, in which case we can be confident that the storm produced severe wind, or 2) the storm is linked to enough wind observations $N_{obs,min}$ that we can be confident in either label (severe wind or no severe wind). We set $N_{obs,min}$ to 25.[3] In addition to criteria 1 and 2, the fraction of dead storms (those that do not live past t_{min}) in the subsampled data must equal the fraction in the full dataset. Again, this prevents survivor bias. The resulting distribution is shown in Fig. 7d.

3) TESTING THE MODEL

Like the calibration set, the testing set should have a U_{max} distribution similar to the true distribution. This way, testing performance is a good indicator of future performance (e.g., in real-time forecasting). Thus, we use the best-observed distribution for the testing set as well.

Also, there must be a 24-h separation between each pair of subdatasets. In other words, if storm object S is in one set, the other two sets may contain no storm object within 24 h of S.

Specifically, 50% of available storm objects (those drawn from the uniform distribution) are added to the base-model training set, 75% of the remaining storm objects (those from the best-observed distribution with a 24-h separation from the base-model training set) are

[3] In a separate experiment we found that 25 is the smallest $N_{obs,min}$ value that allows the ML models to produce well-calibrated high-probability forecasts (near 1.0) at lead times up to 60 min (see Fig. 10 and Figs. C4–C6 in the online supplement). To produce well-calibrated high-probability forecasts at 60–90 min, the required $N_{obs,min}$ value was ~100, but this reduced the size of the dataset too much. Admittedly, the choice of 25 is still arbitrary, and we do not know exactly what $N_{obs,min}$ should be to call a storm well observed with respect to near-surface winds.

added to the calibration set, and 100% of remaining storm objects (those from the best-observed distribution with a 24-h separation from the other two sets) are added to the testing set. Table 4 shows the number of examples in each set.

TABLE 4. Number of training and testing examples (storm objects) for each distance–lead-time pair. Each cell contains (in order) the number of base-model training, isotonic-regression training, and testing examples.

	Distance buffer		
Lead time (min)	Inside storm	0–5 km outside	5–10 km outside
0–15	185 286	241 800	148 686
	18 959	40 495	27 632
	5183	10 189	6919
15–30	156 256	198 752	113 529
	19 099	29 670	34 643
	5596	7297	11 970
30–45	106 728	138 231	74 789
	20 421	38 805	32 658
	8058	15 166	11 227
45–60	69 485	92 231	49 676
	22 088	42 409	31 111
	7562	10 872	8123
60–90	69 072	94 070	52 641
	46 044	104 149	103 155
	16 158	34 201	30 696

4. Experimental design

Procedures described in this section are repeated for each distance buffer and time window. Figure 8 shows how these procedures fit together.

a. Models used

We use five types of base models: logistic regression [section 3a(1)], LREN [section 3a(2)], FFNN [section 3a(3)], random forests [section 3a(5)], and GBEs [section 3a(6)]. For each base model, isotonic regression (section 3b) is used to calibrate forecast probabilities. We vary the parameters for each base-model type as described in appendix B in the online supplement.

b. Model selection

For each set of model parameters, we use K-fold cross validation, as described below.

1) Split the base-model training data (section 3c) into K mutually independent sets, called folds. Independence is defined by the 24-h criterion, as in section 3c.
2) Train K versions of the base model. The mth version is trained with all folds other than the mth, leaving the mth fold as validation.
3) Find the version with the highest area under the curve (AUC; defined in section 5) on its validation fold. Let this version of the base model be F_b. We use AUC because it is insensitive to the distribution of labels, which may vary among the validation folds.
4) Split the calibration data (section 3c) into K folds.
5) Train K versions of the isotonic-regression model. The mth version is trained with forecast probabilities generated by F_b on all folds other than the mth, leaving the mth fold as validation.
6) Find the version with the lowest BS [Eq. (7)] on its validation fold. Let this version of the isotonic-regression model be F_i. We use BS here rather than AUC, because it is the cost function for isotonic regression and a better indicator of probability calibration.
7) Together, F_b and F_i make up a "calibrated model." Find the AUC of the calibrated model on the validation fold for F_i (which is independent of the training data for both F_b and F_i). Call this AUC_v.

This procedure generates 1786 calibrated models (one logistic-regression model, 25 LREN models, 800 FFNNs, 210 random forests, and 750 GBEs). We select the one with the highest AUC_v. Again, we use the AUC because it is insensitive to the distribution of labels, which may vary among validation sets for the different models.

5. Results and discussion

Table 5 shows the testing AUC for each selected model. For each distance buffer and time window, Tables C1–C15 in the online supplement show parameters for the top models (the selected model and those for which AUC_v is not significantly different at the 95% level). Most of the top models are decision-tree based (either a GBE or random forest), which is not surprising, given the ability of decision trees to handle a large number of predictors [section 3a(4)] and the recent success of GBEs in meteorology [as highlighted in McGovern et al. (2015), the top three finishers in a recent contest to predict surface insolation were all GBEs]. No single model has prohibitive computing requirements, as each one can be trained in <2 h and applied to all active storm cells in <1 s, so there are no nonperformance-related reasons to prefer one model over another.

For each distance buffer at 0–15 min (the shortest lead times) and 60–90 min (the longest lead times), Figs. 9 and 10 show the selected model's performance on testing data. Figures C1–C6 in the online supplement show the same graphics for 15–60-min lead times. Specifically, these figures show the receiver operating characteristic (ROC) curve (Metz 1978), performance diagram (Roebber 2009), and attributes diagram (Hsu and Murphy 1986). The ROC

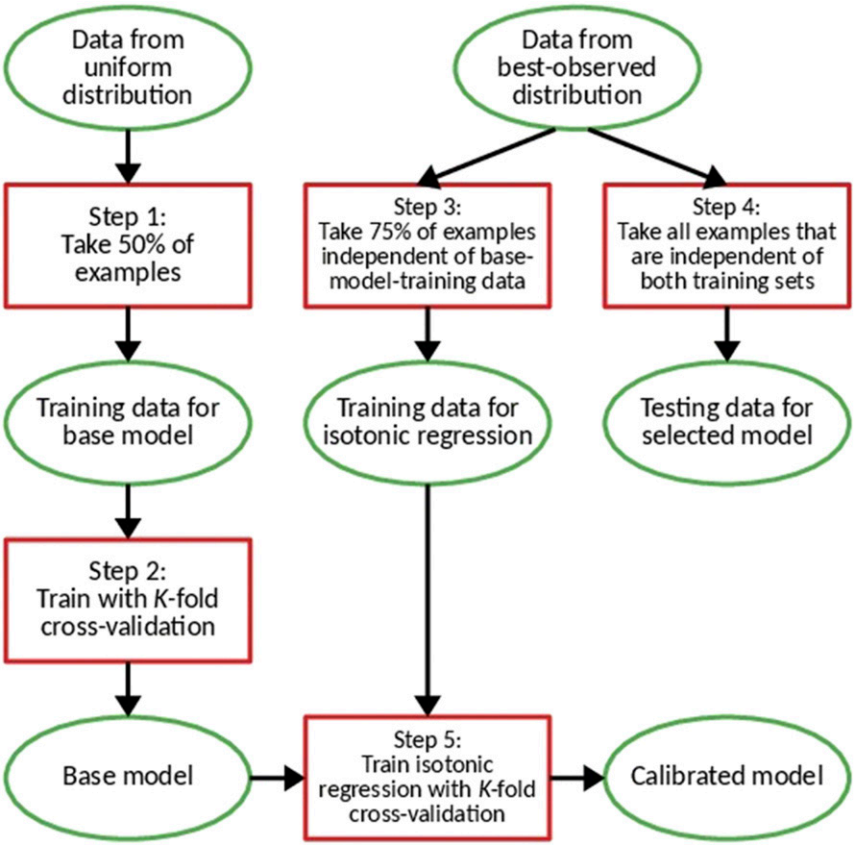

FIG. 8. Flowchart for an ML experiment. Each red box is an action, and each green ellipse is an object or set thereof. The procedure depicted in the flowchart is repeated for all three distance buffers, five time windows, and 1786 sets of model parameters. Testing data are used only to report the performance of the selected model for each distance buffer and time window.

curve shows the probability of detection (POD) versus the probability of false detection (POFD) [see Eq. (8)]. A frequently used single-number summary is AUC. When assessing the accuracy of a medical diagnosis, regardless of its frequency in the population, AUC > 0.9 is considered "excellent" and AUC > 0.8 is considered "good" (Luna-Herrera et al. 2003; Muller et al. 2005; Mehdi et al. 2011).

The performance diagram shows POD versus the success ratio (SR), overlain with contours of the critical success index (CSI) and frequency bias (FB) [Eq. (8)]. There is no standard way to judge a performance diagram, because SR and CSI change significantly with the distribution of labels (cf. Figs. 9b and 9d). However, the maximum CSI should occur with minimal bias (FB $\cong$ 1):

$$\mathrm{POD} = \frac{a}{a+c}; \quad \mathrm{POFD} = \frac{b}{b+d}; \quad \mathrm{SR} = \frac{a}{a+b};$$
$$\mathrm{CSI} = \frac{a}{a+b+c}; \quad \mathrm{FB} = \frac{a+b}{a+c}, \tag{8}$$

where a is the number of hits (event was forecast and occurred), b is the number of false alarms (event was forecast but did not occur), c is the number of misses (event was not forecast but occurred), and d is the number of correct nulls (event was not forecast and did not occur). The "event" is storm-maximum wind $\geq$ 50 kt. POD, POFD, SR, and CSI

TABLE 5. Results of the ML experiment. The top row in each cell shows the AUC on testing data for the selected model; the bottom row is the number of top models (those for which validation AUC is not significantly worse than the selected model) that are decision-tree based (either a GBE or a random forest).

Lead time (min)	Distance buffer: Inside storm	Distance buffer: 0–5 km outside	Distance buffer: 5–10 km outside
0–15	AUC = 0.9315 2 of 2	AUC = 0.9025 1 of 1	AUC = 0.9202 4 of 5
15–30	AUC = 0.9466 26 of 52	AUC = 0.9193 21 of 32	AUC = 0.9042 11 of 13
30–45	AUC = 0.9036 1 of 2	AUC = 0.8989 2 of 3	AUC = 0.9021 30 of 42
45–60	AUC = 0.8946 1 of 1	AUC = 0.8879 1 of 2	AUC = 0.8771 0 of 1
60–90	AUC = 0.8883 1 of 2	AUC = 0.8867 4 of 6	AUC = 0.9007 12 of 15

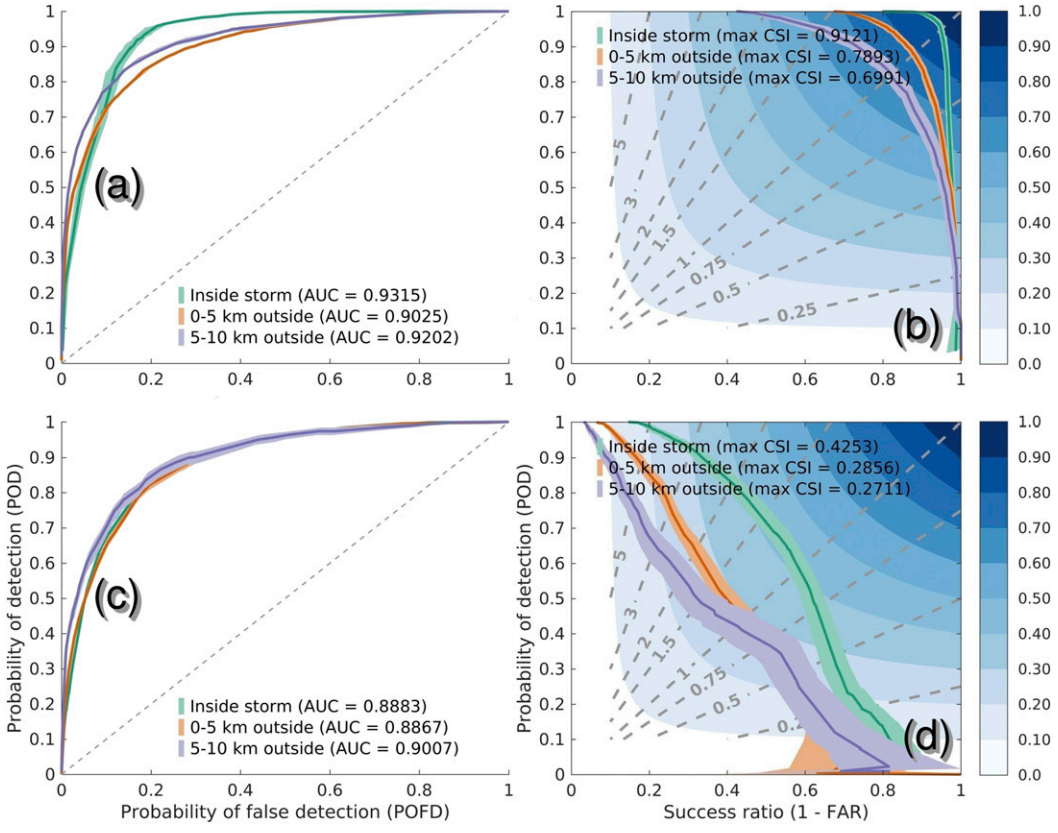

FIG. 9. ROC curves and performance diagrams for selected models at (a),(b) 0–15- and (c),(d) 60–90-min lead times. Each solid line (shaded area) is the mean (95% confidence interval), determined by bootstrapping the testing set. Dashed lines in (a) and (c) show the ROC curve for a random predictor, and dashed lines in (b) and (d) are for the frequency bias, while blue fill is CSI. Selected models (from nearest to farthest distance buffer) are a GBE, random forest, and GBE for 0–15 min; FFNN, GBE, and GBE for 60–90 min.

have a range of [0, 1]; FB has a range of [0, ∞). Higher values of POD, SR, and CSI; lower values of POFD; and FB values near 1 are considered better.

Finally, the attributes diagram plots forecast probability f versus conditional event frequency $p(y = 1|f)$. The BS [Eq. (7)] can be written in the following form (Hsu and Murphy 1986), which is more convenient for interpreting an attributes diagram:

$$\mathrm{BS} = \mathrm{UNC} + \mathrm{REL} - \mathrm{RES}; \quad \mathrm{UNC} = \bar{y}(1 - \bar{y});$$
$$\mathrm{RES} = \frac{1}{M}\sum_{k=1}^{K} M_k(\bar{y}_k - y_k)^2, \tag{9}$$

where REL, M, M_k, f_k, and $\bar{y}_k$ are as in Eq. (6). In addition, $\bar{y} \in [0, 1]$ is the overall event frequency ("climatology"), BS $\in [0, 1]$, UNC $\in [0, 0.25]$ is the uncertainty, REL $\in [0, 1]$ is the reliability, and RES $\in [0, 1]$ is the resolution. Lower values of BS and REL, and higher values of RES, are desired. UNC cannot be judged this way, because it depends only on climatology, which is outside of human control. The Brier skill score (BSS), which compares the model BS to a climatology forecast, is as follows:

$$\mathrm{BSS} = \frac{\mathrm{BS}_{\mathrm{climo}} - \mathrm{BS}}{\mathrm{BS}_{\mathrm{climo}}} = \frac{\mathrm{RES} - \mathrm{REL}}{\mathrm{UNC}}. \tag{10}$$

BSS $\in (-\infty, 1]$, and higher values are desired. BSS > 0 (RES $>$ REL) means that the model is better than climatology.

Observations from Figs. 9 and 10 and Figs. C1–C6 in the online supplement are as follows.

1) AUC, maximum CSI, and BSS decrease (worsen) with buffer distance and lead time. This makes sense, because storms impact their immediate environment more strongly, and forecast skill generally decreases with lead time.
2) In each attributes diagram, most of the reliability curve lies in the positive-skill area, where BSS > 0 (the model is better than climatology).
3) Climatology (the dashed vertical line in the attributes diagram) decreases with buffer distance and

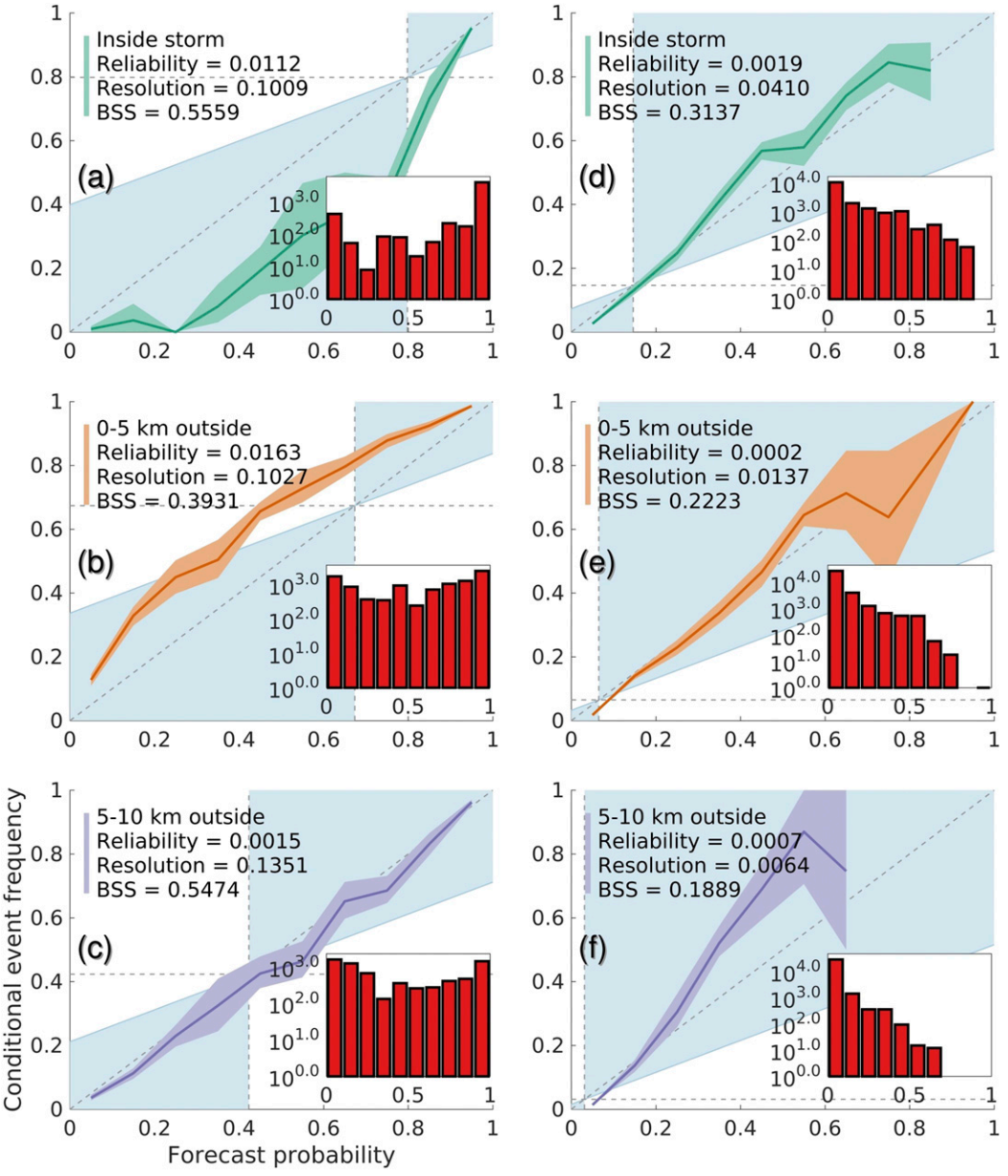

FIG. 10. Attributes diagrams for selected models at (a)–(c) 0–15- and (d)–(f) 60–90-min lead times. Each solid line (shaded area) is the mean (95% confidence interval), determined by bootstrapping the testing set. Blue shading indicates where BSS > 0, the diagonal gray line is perfect reliability, the vertical gray line is climatology $\bar{y}$, the horizontal gray line is the no-resolution line (produced by always forecasting $\bar{y}$), and the inset (red bar graph) is the forecast histogram.

lead time. This makes sense, because thunderstorms are short-lived and small-scale phenomena, so as distance and lead time increase, they become less likely to produce severe wind.

4) As climatology decreases, the forecast histogram becomes more left peaked. In other words, as the event becomes rarer, high forecast probabilities f become rarer and low f become more common, as desired.
5) AUC > 0.88 for all distance–lead-time pairs.
6) For each distance–lead-time pair, the maximum CSI occurs with little bias (FB $\cong 1$).

In general, the most important predictors are storm motion and sounding indices, rather than radar statistics and shape parameters (Lagerquist et al. 2017). However, our predictor-ranking methods are complicated, so this will be left for a subsequent paper.

The main caveat of this work is that we subsample cases for calibration and testing, using the best-observed distribution (section 3c). We use the best-observed distribution, first because it undersamples dubious null cases (storm objects with <25 wind observations, all nonsevere), where it is more likely that the storm produced severe wind that was simply unobserved. Second, without the best-observed distribution—or just criterion 1 (see section 3c)—our models learned to never forecast probabilities > 20% and usually forecast <1%. Third, calibration and testing data should be processed in the same way (models should be calibrated to increase their performance when using evaluation data). However, when human forecasters are evaluated, they do not have the benefit of undersampling poorly observed storms, so our models would cause them to issue more unverified warnings. This would cause performance metrics other than AUC to drop significantly.

6. Summary and future work

We used machine learning (ML) to forecast the probability of damaging straight-line wind (≥50 kt or 25.7 m s^{-1}) for a single storm cell. Three datasets—radar images, modeled soundings, and near-surface wind observations—were incorporated into the ML models. Forecasts were made for three distance buffers (inside storm cell, 0–5 km outside, 5–10 km outside) and five time windows (0–15, 15–30, 30–45, 45–60, and 60–90 min ahead).

For each distance buffer and time window, a two-step model was trained, consisting of a base model (to generate initial probabilities) and an isotonic-regression model (to calibrate said probabilities). After experimenting with five types of base models, we determined that the best model for most distance–lead-time pairs is an ensemble of decision trees (either random forest or gradient boosted).

We hypothesized that our models would outperform climatology for each combination of the distance buffer, time window, and forecast probability (discretized into 10 bins). According to the Brier skill score, this was achieved for all but a few of the 150 combinations. Also, the area under the ROC curve was >0.88 for all distance–lead-time pairs.

We cannot compare the performance to non-ML methods (sounding based, radar based, and explicit NWP), because they do not produce explicit probabilities (between 0 and 1). We also cannot compare directly to previous ML studies, because they either forecast straight-line wind only for severe storms (Alexiuk et al. 1999) or did not separate straight-line wind from other storm hazards (Kitzmiller et al. 1995; Marzban and Stumpf 1998; Cintineo et al. 2014). Nonetheless, we extended previous studies by forecasting straight-line wind, distinct from other hazards, for both severe and nonsevere storms; using measured wind observations in addition to human reports; using multiple data types (radar and NWP) to create predictors, which was done only by Cintineo et al. (2014); using merged (rather than single radar) data, which was done only by Cintineo et al. (2014); calibrating probabilities, which was done only by Marzban and Stumpf (1998); and using a large number of predictors (we used 431, whereas previous studies used no more than 23). Many of these advances were made possible by more computing resources, or more and better data, than were available earlier.

We are currently improving this work in several ways, which will be the subject of a future article. First, we are applying variable-selection and variable-transformation methods (e.g., forward and backward selection, principal-component analysis) to selected models to understand the physical relationships being exploited. Second, we are interpolating forecast probabilities onto a grid, accounting for storm motion. The output will be one grid per time window, rather than one forecast per storm cell, distance buffer, and time window, which will drastically reduce the cognitive load for the user. Third, we are using more advanced ML techniques. Weather data are four-dimensional and have high-level spatiotemporal relationships, which is ideal for deep learning (LeCun et al. 2015) and spatiotemporal relational probability trees (McGovern et al. 2008), respectively.

Acknowledgments. The authors thank David John Gagne and Michael Richman for extensive input throughout this project, as well as Kelton Halbert for extensive technical support with SHARPpy. This work was funded by the National Oceanic and Atmospheric Administration's (NOAA) Office of Oceanic and Atmospheric Research, under Cooperative Agreement NA11OAR4320072. The machine-learning experiment was performed at the University of Oklahoma's (OU) Supercomputing Center for Education and Research (OSCER) at OU.

REFERENCES

Alexiuk, M., N. Pizzi, and W. Pedrycz, 1999: Classification of volumetric storm cell patterns. *Canadian Conf. on Electrical and Computer Engineering*, Edmonton, AB, Canada, Institute of Electrical and Electronics Engineers, https://doi.org/10.1109/CCECE.1999.808201.

Ashley, W., and A. Black, 2008: Fatalities associated with nonconvective high-wind events in the United States. *J. Appl. Meteor. Climatol.*, **47**, 717–725, https://doi.org/10.1175/2007JAMC1689.1.

Batista, G. E. A. P. A., R. Prati, and M. Monard, 2004: A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explorations*, Vol. 6, No. 1,

20–29, Association for Computing Machinery, New York, NY, https://doi.org/10.1145/1007730.1007735.

Benjamin, S., and Coauthors, 2004: An hourly assimilation–forecast cycle: The RUC. *Mon. Wea. Rev.*, **132**, 495–518, https://doi.org/10.1175/1520-0493(2004)132<0495:AHACTR>2.0.CO;2.

Blumberg, W., K. Halbert, T. Supinie, P. Marsh, R. Thompson, and J. Hart, 2017a: 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.

——, ——, ——, ——, ——, and ——, 2017b: SHARPpy/params.py at master. Accessed 5 September 2017, https://github.com/sharppy/SHARPpy/blob/master/sharppy/sharptab/params.py.

Breiman, L., 2001: Random forests. *Mach. Learn.*, **45**, 5–32, https://doi.org/10.1023/A:1010933404324.

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.

Coniglio, M., D. Stensrud, and M. Richman, 2004: An observational study of derecho-producing convective systems. *Wea. Forecasting*, **19**, 320–337, https://doi.org/10.1175/1520-0434(2004)019<0320:AOSODC>2.0.CO;2.

Corfidi, S., M. Coniglio, A. Cohen, and C. Mead, 2016: A proposed revision to the definition of "derecho." *Bull. Amer. Meteor. Soc.*, **97**, 935–949, https://doi.org/10.1175/BAMS-D-14-00254.1.

Delanoy, R., and S. Troxel, 1993: Machine intelligent gust front detection. *Linc. Lab. J.*, **6**, 187–212.

Doswell, C. A., III, H. Brooks, and M. Kay, 2005: Climatological estimates of daily local nontornadic severe thunderstorm probability for the United States. *Wea. Forecasting*, **20**, 577–595, https://doi.org/10.1175/WAF866.1.

Evans, J., and C. A. Doswell III, 2001: Examination of derecho environments using proximity soundings. *Wea. Forecasting*, **16**, 329–342, https://doi.org/10.1175/1520-0434(2001)016<0329:EODEUP>2.0.CO;2.

Friedman, J., 2001: Greedy function approximation: A gradient boosting machine. *Ann. Stat.*, **29**, 1189–1232, https://doi.org/10.1214/aos/1013203451.

Fujita, T. T., 1990: Downbursts: Meteorological features and wind field characteristics. *J. Wind Eng. Ind. Aerodyn.*, **36**, 75–86, https://doi.org/10.1016/0167-6105(90)90294-M.

Gagne, D., A. McGovern, J. Brotzge, M. Coniglio, J. Correia, and M. Xue, 2015: Day-ahead hail prediction integrating machine learning with storm-scale numerical weather models. *29th Conf. on Artificial Intelligence*, Austin, TX, Association for the Advancement of Artificial Intelligence, https://www.aaai.org/ocs/index.php/IAAI/IAAI15/paper/viewFile/9724/9898.

Haykin, S., 2001: Feedforward neural networks: An introduction. *Nonlinear Dynamical Systems: Feedforward Neural Network Perspectives*, I. Sandberg, Ed., John Wiley and Sons, 1–16.

Hoerl, A., and R. Kennard, 1988: Ridge regression. *Encyclopedia of Statistical Sciences*, S. Kotz, Ed., Vol. 8, John Wiley and Sons, 129–136.

Hsu, W., and A. Murphy, 1986: The attributes diagram: A geometrical framework for assessing the quality of probability forecasts. *Int. J. Forecasting*, **2**, 285–293, https://doi.org/10.1016/0169-2070(86)90048-8.

Karstens, C., T. Smith, K. Kuhlman, A. Clark, C. Ling, G. Stumpf, and L. Rothfusz, 2014: Prototype tool development for creating probabilistic hazard information for severe convective phenomena. *Second Symp. on Building a Weather-Ready Nation*, Atlanta, GA, Amer. Meteor. Soc., 2.2, https://ams.confex.com/ams/94Annual/webprogram/Paper241549.html.

Kitzmiller, D., W. McGovern, and R. Saffle, 1995: The WSR-88D severe weather potential algorithm. *Wea. Forecasting*, **10**, 141–159, https://doi.org/10.1175/1520-0434(1995)010<0141:TWSWPA>2.0.CO;2.

Lagerquist, R., A. McGovern, and T. Smith, 2017: Using machine learning to predict straight-line convective wind hazards throughout the continental United States. *15th Conf. on Artificial and Computational Intelligence and Its Applications to the Environmental Sciences*, Seattle, WA, Amer. Meteor. Soc., 4.3, https://ams.confex.com/ams/97Annual/webprogram/Paper316107.html.

Lakshmanan, V., and T. Smith, 2010: Evaluating a storm tracking algorithm. *26th Conf. on Interactive Information Processing Systems*, Atlanta, GA, Amer. Meteor. Soc., 8.2, https://ams.confex.com/ams/90annual/techprogram/paper_162556.htm.

——, ——, K. Hondl, G. Stumpf, and A. Witt, 2006: A real-time, three-dimensional, rapidly updating, heterogeneous radar merger technique for reflectivity, velocity, and derived products. *Wea. Forecasting*, **21**, 802–823, https://doi.org/10.1175/WAF942.1.

——, ——, 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.

LeCun, Y., Y. Bengio, and G. Hinton, 2015: Deep learning. *Nature*, **521**, 436–444, https://doi.org/10.1038/nature14539.

Luna-Herrera, J., G. Martinez-Cabrera, R. Parra-Maldonado, J. Enciso-Moreno, J. Torres-Lopez, F. Quesada-Pascual, R. Delgadillo-Polanco, and S. Franzblau, 2003: Use of receiver operating characteristic curves to assess the performance of a microdilution assay for determination of drug susceptibility of clinical isolates of Mycobacterium tuberculosis. *Eur. J. Clin. Microbiol. Infect. Dis.*, **22**, 21–27, https://doi.org/10.1007/s10096-002-0855-5.

Marriott, R., 2012: Challenges for data assimilation—From convective-scale to climate. *Weather*, **67** (10), 277–278, https://doi.org/10.1002/wea.1982.

Marzban, C., and G. Stumpf, 1998: A neural network for damaging wind prediction. *Wea. Forecasting*, **13**, 151–163, https://doi.org/10.1175/1520-0434(1998)013<0151:ANNFDW>2.0.CO;2.

MathWorks, 2016: Pattern recognition network—MATLAB patternnet. Accessed 24 September 2016, http://www.mathworks.com/help/nnet/ref/patternnet.html.

McGovern, A., N. Hiers, M. Collier, D. Gagne, and R. Brown, 2008: Spatiotemporal relational probability trees: An introduction. *Eighth Int. Conf. on Data Mining*, Pisa, Italy, Institute of Electrical and Electronics Engineers, https://doi.org/10.1109/ICDM.2008.134.

——, D. Gagne, 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.

McNitt, J., J. Facundo, and J. O'Sullivan, 2008: Meteorological Assimilation Data Ingest System Transition Project risk reduction activity. *24th Conf. on Interactive Information Processing Systems*, New Orleans, LA, Amer. Meteor. Soc., 7C.1, https://ams.confex.com/ams/88Annual/techprogram/paper_134617.htm.

McPherson, R., andCoauthors, 2007: Statewide monitoring of the mesoscale environment: A technical update on the Oklahoma Mesonet. *J. Atmos. Oceanic Technol.*, **24**, 301–321, https://doi.org/10.1175/JTECH1976.1.

Mehdi, T., N. Bashardoost, and M. Ahmadi, 2011: Kernel smoothing for ROC curve and estimation for thyroid stimulating hormone. *Int. J. Public Health Res.*, **Special Issue**, 239–242.

Mesinger, F., and Coauthors, 2006: North American Regional Reanalysis. *Bull. Amer. Meteor. Soc.*, **87**, 343–360, https://doi.org/10.1175/BAMS-87-3-343.

Metz, C., 1978: Basic principles of ROC analysis. *Semin. Nucl. Med.*, **8**, 283–298, https://doi.org/10.1016/S0001-2998(78)80014-2.

Mitchell, T., 1997: *Machine Learning*. McGraw-Hill, 414 pp.

Muller, M., G. Tomlinson, T. Marrie, P. Tang, A. McGeer, D. Low, A. Detsky, and W. Gold, 2005: Can routine laboratory tests discriminate between severe acute respiratory syndrome and other causes of community-acquired pneumonia? *Clin. Infect. Dis.*, **40**, 1079–1086, https://doi.org/10.1086/428577.

National Climatic Data Center, 2006: Data documentation for data set 6406 (DSI-6406): ASOS surface 1-minute, page 2 data. Accessed 14 June 2016, ftp://ftp.ncdc.noaa.gov/pub/data/asos-onemin/td6406.txt.

National Severe Storms Laboratory, 2016a: Severe weather 101: Damaging winds basics. Accessed 24 September 2016, https://www.nssl.noaa.gov/education/svrwx101/wind/.

——, 2016b: Severe weather 101: Types of damaging winds. Accessed 2 August 2016, https://www.nssl.noaa.gov/education/svrwx101/wind/types/.

National Weather Service, 2010: One inch hail. Accessed 24 September 2016, http://www.nws.noaa.gov/oneinchhail/.

——, 2016a: *Storm Data* preparation. National Weather Service Instruction 10-1605, 19 pp. + appendixes, http://www.nws.noaa.gov/directives.

——, 2016b: Thunderstorm hazards—Damaging wind. Accessed 7 October 2016, http://www.srh.noaa.gov/jetstream/tstorms/wind.html.

Niculescu-Mizil, A., and R. Caruana, 2005: Predicting good probabilities with supervised learning. *22nd Int. Conf. on Machine Learning*, Bonn, Germany, International Machine Learning Society, 625–632.

Ortega, K., T. Smith, J. Zhang, C. Langston, Y. Qi, S. Stevens, and J. Tate, 2012: The Multi-Year Reanalysis of Remotely Sensed Storms (MYRORSS) project. *26th Conf. on Severe Local Storms*, Nashville, TN, Amer. Meteor. Soc., 205, https://ams.confex.com/ams/37RADAR/webprogram/Handout/Paper275486/205_ortega_et_al_myrorss.pdf.

Platt, J., 2000: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, A. J. Smola et al., Eds., MIT Press, 61–74.

Roebber, P., 2009: Visualizing multiple measures of forecast quality. *Wea. Forecasting*, **24**, 601–608, https://doi.org/10.1175/2008WAF2222159.1.

Rose, M., 1996: Downbursts. *Natl. Wea. Dig.*, **21** (1), 11–17.

Saxen, T., 2002: Forecasting C-G lightning potential at WSMR. *13th Conf. on Applied Climatology/10th Conf. on Aviation, Range, and Aerospace Meteorology*, Portland, OR, Amer. Meteor. Soc., JP1.29, https://ams.confex.com/ams/pdfpapers/39104.pdf.

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.

Storm Prediction Center, 2016a: About derechoes. Accessed 2 August 2016, http://www.spc.noaa.gov/misc/AbtDerechos/derechofacts.htm.

——, 2016b: SPC mesoscale analysis pages. Accessed 14 June 2016, http://www.spc.noaa.gov/exper/mesoanalysis/.

——, 2017a: Derecho composite parameter (DCP). Accessed 5 September 2017, http://www.spc.noaa.gov/exper/mesoanalysis/help/help_dcp.html.

——, 2017b: Microburst composite. Accessed 5 September 2017, http://www.spc.noaa.gov/exper/mesoanalysis/help/help_mbcp.html.

Sun, J., and Coauthors, 2014: Use of NWP for nowcasting convective precipitation: Recent progress and challenges. *Bull. Amer. Meteor. Soc.*, **95**, 409–426, https://doi.org/10.1175/BAMS-D-11-00263.1.

Tibshirani, R., 1996: Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc.*, **58B**, 267–288.

Walker, S., and D. Duncan, 1967: Estimation of the probability of an event as a function of several independent variables. *Biometrika*, **54**, 167–179, https://doi.org/10.1093/biomet/54.1-2.167.

Webb, A., 2003: *Statistical Pattern Recognition*. John Wiley and Sons, 496 pp.

Weisman, M., C. David, W. Wang, K. Manning, and J. Klemp, 2008: Experiences with 0–36-h explicit convective forecasts with the WRF-ARW model. *Wea. Forecasting*, **23**, 407–437, https://doi.org/10.1175/2007WAF2007005.1.

Zou, H., and T. Hastie, 2005: Regularization and variable selection via the elastic net. *J. Roy. Stat. Soc.*, **67B**, 301–320, https://doi.org/10.1111/j.1467-9868.2005.00503.x.