Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Adaptable Regression Method for Ensemble Consensus Forecasting

John K. Williams, Peter P. Neilley, Joseph P. Koval and Jeff McDonald

The Weather Company, Andover, MA

john.williams@weather.com

Abstract

Accurate weather forecasts enhance sustainability by facilitating decision making across a broad range of endeavors including public safety, transportation, energy generation and management, retail logistics, emergency preparedness, and many others. This paper presents a method for combining multiple scalar forecasts to obtain deterministic predictions that are generally more accurate than any of the constituents. Exponentially-weighted forecast bias estimates and error covariance matrices are formed at observation sites, aggregated spatially and temporally, and used to formulate a constrained, regularized least squares regression problem that may be solved using quadratic programming. The model is re-trained when new observations arrive, updating the forecast bias estimates and consensus combination weights to adapt to weather regime and input forecast model changes. The algorithm is illustrated for 0-72 hour temperature forecasts at over 1200 sites in the contiguous U.S. based on a 22-member forecast ensemble, and its performance over multiple seasons is compared to a state-of-the-art ensemble-based forecasting system. In addition to weather forecasts, this approach to consensus may be useful for ensemble predictions of climate, wind energy, solar power, energy demand, and numerous other quantities.

Introduction

Combining information from multiple forecasts—for instance, via bias correction followed by weighted averaging—has long been known to produce deterministic consensus predictions that are generally more accurate than any of the constituent inputs. Consensus methods have been successfully used in a wide range of disciplines including finance, economics and biomedicine (Clemen 1989). Within computer science, "blending" or "stacking" multiple machine-learned models (Wolpert 1992, Breiman 1996, Kuncheva 2004) has proven to be highly effective, with famous successes such as winning the Netflix prize. The focus of this paper is on developing a skillful, practical method for producing an automated ensemble consensus of weather forecasts, exploiting the increasing number of physics-driven numerical weather prediction (NWP) models now operationally available to provide a single authoritative forecast. Accurate weather forecasts are important for daily planning by individuals and families as well as for decision support across a broad range of endeavors including public safety, transportation, energy generation and management, retail logistics, emergency preparedness, and many others. Indeed, Lazo et al. (2011) estimated that weather variability is responsible for a $485 billion annual impact on U.S. economic activity, some portion of which could be mitigated by improved weather forecasts.

Individual physics-based predictive models are limited by the accuracy of observations used for initialization and by their imperfect representations of physical processes, and are subject to both systematic and chaotic excursions. Human forecasters commonly combine multiple sources of forecast guidance, drawing on training and deep experience to account for nonstationarity due to seasons, variable weather patterns, and daily heating and cooling cycles; spatial inhomogeneity in NWP models' predictive performance, which vary as a function of latitude, altitude, land cover, terrain, proximity to water and other factors; ensemble changes, as NWP models are frequently updated; and errors in verifying observations.

Given their promise for reducing forecaster workload and generating more accurate predictions, automated consensus methods that attempt to intelligently combine multiple weather or climate predictions have received considerable attention in recent years. Thompson (1977) argued that a weighted linear combination of two imperfectly correlated scalar weather forecasts could reduce the error variance by 20%. Krishnamurti et al. (1999) fit linear regressions to model ensemble forecast anomalies and showed improved mean-squared error (MSE) for seasonal climate, global weather, and hurricane forecasts during subsequent testing periods. An operational Australian consensus system used a recent lookback period to estimate biases for several forecast models or derived model output statistics (MOS) products, and combined them using weights in-

versely proportional to their mean absolute errors (MAEs); the resulting predictions for several meteorological quantities were shown to be significantly better than existing guidance or operational forecast products (Woodcock and Engel 2005, Engel and Ebert 2007). The system was later extended to provide gridded forecasts (Engel and Ebert 2012). DelSole (2007) discussed ridge regression in a Bayesian framework and used it to combine seasonal-scale NWP model predictions. Peña and van den Dool (2008) also utilized ridge regression and incorporated information from neighboring pixels to diminish negative combination weights in consolidating sea surface temperature forecasts. The Dynamically Integrated ForeCast (DICast®) system, originally developed at the National Center for Atmospheric Research (Gerding and Myers 2003, Myers and Linden 2011, Koval et al. 2015), dynamically adjusts input forecast bias estimates and updates combination weights using stochastic gradient descent. DICast is the core of the forecast blending methodology currently used by The Weather Company to provide billions of unique forecasts to global users every day. The stochastic gradient descent approach is computationally efficient, requires little storage, and adapts to changes in weather regimes or changing NWP models. However, (1) adding or removing input forecasts is not straightforward; (2) a missing forecast model run requires either using an older forecast or setting its weight to zero, neither of which is optimal; (3) constraining weights or specifying "preferred" weights is not naturally included; and (4) the influence of erroneous observations cannot easily be corrected after they are incorporated.

This paper presents an adaptable regression (AR) approach to dynamic, spatiotemporal consensus forecasting that addresses DICast's limitations. It incorporates three fundamental innovations: (1) formulating the constrained, regularized regression with "target" combination weights and weight bounds as a quadratic programming problem, and providing an approximate but fast solution method; (2) incorporating spatiotemporal neighborhood information through exponential moving averages and aggregation of forecast input error covariance matrices and biases; and (3) allowing modulation of bias estimates by a multiplicative factor. The result is a flexible consensus forecasting methodology with improved performance.

Regression methodology

In the weather forecasting ensemble consensus context, the input forecasts for various meteorological variables are generated on diverse geospatial grids at operational centers all over the world, arrive asynchronously, and are downscaled to the desired time resolution and to the location of weather stations that provide "truth" observations. The time at which a forecast is produced is called its "generation" time, the future time to which it applies is the "valid" time, and the difference between them is the "lead" time. For simplicity of notation, we initially assume a single target meteorological variable, forecast location and valid/lead time in the description below. For consensus forecast generation at time t, the ensemble of p input forecasts may be represented as vectors $\mathbf{f}_t = \{f_t(i) | i = 1,..,p\}$, and the system learns corresponding input forecast biases $b_t(i)$ and weights $w_t(i)$, producing consensus forecasts through the weighted average

$$(1)\quad \boldsymbol{F_t} = \sum_{i=1}^{p} \big(f_t(i) - b_t(i)\big) w_t(i) = (\mathbf{f}_t - \mathbf{b}_t)^{\mathrm{T}} \mathbf{w}$$

where $\sum_{i=1}^{p} w_t(i) = 1$, i.e., $\mathbf{1}^{\mathbf{T}}\mathbf{w} = 1$. Here $\mathbf{1}$ represents the vector of 1's and T denotes the matrix transpose. If $\boldsymbol{v_t}$ is the verifying observation obtained at the forecast valid time, then the error in the consensus forecast $\boldsymbol{F_t}$ is

$$(2)\quad \begin{aligned} \boldsymbol{E}_t &= \boldsymbol{F}_t - \boldsymbol{v}_t = \sum_{i=1}^{p} (f_t(i) - b_t(i) - v_t(i)) w_t(i) \\ &= \sum_{i=1}^{p} d_t(i) w_t(i) = (\mathbf{d_t})^{\mathbf{T}} \mathbf{w} \end{aligned}$$

where $d_t(i)$ is the bias-corrected error for input forecast *i*. Below, we describe how the bias and the error covariance matrix may be estimated from a database of historical forecast errors and how appropriate consensus weights are computed. Although the procedure is described using concepts from weather forecasting, it applies to a broad class of consensus prediction problems where an ensemble of forecasts is continually generated and verified against observations.

Estimating bias

The bias of each input forecast is an estimate of its systematic, or expected, error (Wilks 2005). Weather forecast errors are functions of location, lead time, valid time of day, season, and weather regime, to name a few. Moreover, NWP forecast models frequently update their resolution, data assimilation and physics schemes, causing their performance statistics to change over time. To accommodate these dependencies, the bias for each input forecast i is computed for each location, lead time (t_{lead}) and valid time of day based on that forecast's past performance. Let $\{e_1(i), e_2(i), \dots, e_n(i)\}$ denote observed input forecast errors from generation times $\{t_1(i), t_2(i), \dots, t_n(i)\} < t - t_{\mathrm{lead}}$. (The inequality reflects the time lag until a verifying observation is available.) Then for a bias "learning rate" $\gamma \in [0,1)$, the AR method estimates the bias for input forecast i at generation time t via the "modulated" exponential moving average (EMA)

$$(3)\quad b_t(i) = \mu \frac{\sum_{k=1}^{n} (1-\gamma)^{t-t_k(i)} e_k(i)}{\sum_{k=1}^{n} (1-\gamma)^{t-t_k(i)}}$$

When $\mu = 1$ and $\gamma = 0$, eq. (3) is simply the standard (unweighted) mean of the latest n forecast errors; as γ grows larger, recent errors are given more influence than older ones. Thus, small values of γ reduce the impact of random error on the bias estimate, whereas increasing γ diminishes the representativeness error (since recent forecast statistics are presumably more like those that will characterize the future forecast period); ideally, a happy medium can be identified empirically. The non-standard formulation of the EMA in eq. (3) allows for temporally unequally-spaced error observations and, by explicitly normalizing by the exponential weights, avoids the "spinup" problem exhibited by the more commonly used iterative formulation. The multiplicative factor $\mu \in [0,1]$ allows the bias to be "modulated" (nudged toward 0) to address the fact that, due to nonstationarity, a backward-looking average may tend to overestimate the magnitude of systematic error in subsequent forecasts, especially when the input forecasts are presumably designed to be unbiased. Said another way, multiplying by $\mu < 1$ provides the effect of a Bayesian prior centered around 0; more generally, if the prior is centered around ρ, the term $(1-\mu)\rho$ should be added to the right-hand side of eq. (3).

Estimating the error covariance

Let the bias-corrected input forecast errors be denoted by $d_k(i) = e_k(i) - b_k(i)$, where $\mathbf{b}_k = \mathbf{b}_{t_k}$, i.e., the bias estimate from eq. (3) that would have been available at forecast generation time t_k. The AR method estimates the error covariance matrix $\mathbf{C_t}$ via

$$(4) \quad C_t(i,j) = \kappa(t_1, \ldots, t_n) \frac{\sum_{k=1}^{n} (1-\eta)^{t-t_k} \boldsymbol{d_k}(i) \boldsymbol{d_k}(j)}{\sum_{k=1}^{n} (1-\eta)^{t-t_k}}$$

where η is the covariance learning rate and κ is a normalizing functional that is set to 1 for simplicity, since it does not affect the linear regression solution described below. Of course, $C_t(i,j) = C_t(j,i)$, so eq. (4) does not need to be computed for all error covariance matrix entries. Similar to the bias equation, this formulation addresses the nonstationarity of forecast ensemble performance by weighting more recent samples of the bias-corrected error more heavily than older ones. If $\eta = 0$ and $\kappa = n/(n-1)$, eq. (4) reduces to the standard definition of the sample covariance.

Formulating the regression problem

If the forecast module error processes were stationary, the appropriate choice of learning rates would be $\gamma = \eta = 0$, reducing eqs. (3) and (4) to the usual bias and sample covariance computations, respectively. If additionally the generation times satisfied $t_k(i) = t_k(j)\ \forall\ i,j$ and $k = 1, \ldots, n$, we could find the *a posteriori* least-squares value for $\mathbf{w}$ by minimizing the sum of squared errors from eq. (2):

$$(5) \quad \begin{aligned} &\operatorname{argmin}_{\mathbf{w}} \textstyle\sum_{k=1}^{n} ((\mathbf{d_k})^{\mathrm{T}}\mathbf{w})^2 \ \text{subj. to}\ \mathbf{1}^{\mathrm{T}}\mathbf{w} = 1 \\ &= \operatorname{argmin}_{\mathbf{w}} \|\mathbf{D}\mathbf{w}\|^2 \ \text{subj. to}\ \mathbf{1}^{\mathrm{T}}\mathbf{w} = 1 \\ &= \operatorname{argmin}_{\mathbf{w}} \mathbf{w}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{w} \ \text{subj. to}\ \mathbf{1}^{\mathrm{T}}\mathbf{w} = 1 \\ &= \operatorname{argmin}_{\mathbf{w}} \tfrac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{C}\mathbf{w} \ \text{subj. to}\ \mathbf{1}^{\mathrm{T}}\mathbf{w} = 1 \end{aligned}$$

Here $\mathbf{D}$ is the matrix having k^{th} row $(\mathbf{d_k})^{\mathrm{T}}$, and $\mathbf{C}$ is the error covariance matrix. Eq. (5) has the form of a quadratic programming problem. The AR method is based on the assumption that eq. (5) remains a good method for computing $\mathbf{w}$ when γ and η are allowed to be nonzero in the bias and error covariance estimates, respectively, providing a tradeoff between the impacts of random error (reduced by small learning rates) and non-representativeness (reduced via large learning rates) in the context of the nonstationary forecasting problem.

Constraints and regularization

Since we assume that the input forecasts in the ensemble are chosen to have positive skill, we generally restrict the solution $\mathbf{w}$ of eq. (5) to also satisfy $\mathbf{w} \geq 0$. In fact, we can specify lower and upper bounds $\mathbf{w}^L \in [0,1]^{\boldsymbol{p}}$ and $\mathbf{w}^U \in [0,1]^{\boldsymbol{p}}$ so long as $\mathbf{1}^{\mathrm{T}}\mathbf{w}^L \leq 1 \leq \mathbf{1}^{\mathrm{T}}\mathbf{w}^U$. Additionally, we may wish to specify a "goal" weight $\mathbf{w}^G$ and a diagonal regularization matrix $\mathbf{R} = \alpha\, \mathbf{I}_{\boldsymbol{p}\times\boldsymbol{p}} + \beta\, \mathrm{diag}(\mathbf{C})$. (See Peña and van den Dool 2008 for a detailed exploration of ridge regression in the context of consensus prediction.) The deviation of a solution $\mathbf{w}$ from $\mathbf{w}^G$ is then quantified by

$$(6) \quad \begin{aligned} &\tfrac{1}{2}\left\|\mathbf{R}^{1/2}(\mathbf{w} - \mathbf{w}^G)\right\|^2 = \tfrac{1}{2}(\mathbf{w} - \mathbf{w}^G)^{\mathrm{T}}\mathbf{R}(\mathbf{w} - \mathbf{w}^G) \\ &= \tfrac{1}{2}[\mathbf{w}^{\mathrm{T}}\mathbf{R}\mathbf{w} - 2(\mathbf{w}^G)^{\mathrm{T}}\mathbf{R}\mathbf{w} + (\mathbf{w}^G)^{\mathrm{T}}\mathbf{R}\mathbf{w}^G] \end{aligned}$$

Since the last term of eq. (6) does not depend on $\mathbf{w}$, the constrained, regularized version of the regression problem specified in eq. (5) is

$$\mathbf{(7)} \quad \begin{aligned} &\operatorname{argmin}_{\mathbf{w}} \tfrac{1}{2}\mathbf{w}^{\mathrm{T}}(\mathbf{C} + \mathbf{R})\mathbf{w} - (\mathbf{w}^{\boldsymbol{G}})^{\mathrm{T}}\mathbf{R}\mathbf{w} \\ &\text{subj. to}\ \mathbf{1}^{\mathrm{T}}\mathbf{w} = 1 \ \text{and}\ \mathbf{w}^L \leq \mathbf{w} \leq \mathbf{w}^U. \end{aligned}$$

Increasing α will generally drive the solution of eq. (7) towards $\mathbf{w}^G$, subject to the constraints. When $\mathbf{w}^G = 0$, increasing β will move the solution towards the "inverse variance" solution, i.e., weighting each input forecast in inverse proportion to its error variance—the optimal solution when the input forecast errors are independent. Thus, eq. (7) provides the ability to add information to the regression problem, constrain the solutions, and diminish the risk of overfitting, which could lead to poor generalization and poor forecast accuracy. As an example, a human fore-

caster "over the loop" could place a lower limit on the weight of an input forecast for extrinsic reasons; or specify diminishing goal weights $w^G(i)$ for lead times near the end of the range of input forecast *i* to mitigate the discontinuity at the lead time when that input forecast disappears from the ensemble; or increase α or β to make the solution weights more temporally or spatially consistent. In the experimental results below, we use a fixed value of $\alpha = 10^{-6}$ to help insure that **C** + **R** is well-conditioned, and adjust β based on empirical sensitivity results.

Solution

The quadratic program in eq. (7) can be solved using standard optimization libraries, e.g., MATLAB's "quadprog" function. Additionally, if $\mathbf{w}^L = -\infty$ and $\mathbf{w}^U = \infty$, we may write the Lagrangian for eq. (7) as

$$(8)\quad \boldsymbol{L} = \tfrac{1}{2}\mathbf{w}^{\mathbf{T}}(\mathbf{C}+\mathbf{R})\mathbf{w} - (\mathbf{w}^{\boldsymbol{G}})^{\mathbf{T}}\mathbf{R}\mathbf{w} + \lambda(\mathbf{1}^{\mathbf{T}}\mathbf{w} - 1)$$

where λ is the Lagrange multiplier. Setting the derivatives of *L* with respect to **w** and λ equal to zero, and using the fact that **C** + **R** is symmetric, the solution to eq. (7) may be obtained by solving the equation

$$(9)\quad \begin{bmatrix}\mathbf{C}+\mathbf{R} & \mathbf{1}\\ \mathbf{1}^{\mathbf{T}} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{w}\\ \lambda\end{bmatrix} = \begin{bmatrix}\mathbf{R}\mathbf{w}^{G}\\ 1\end{bmatrix}$$

for **w**. When the **R** is nontrivial, eq. (9) is typically well-conditioned and straightforward to solve using a linear equation solver such as MATLAB's "linsolve." If $\mathbf{w}^L = 0$ and $\mathbf{w}^U \geq 1$, eq. (9) may be applied iteratively, omitting input forecasts whose weights violate the limits and then re-solving for the remaining weights to obtain an approximate but fast solution to eq. (7). When $\mathbf{w}^L > 0$ or $\mathbf{w}^U < 1$, a slightly more complicated version of eq. (9) may be useful. After one or more iterations of eq. (9), let *A* be the set of "clipped" input forecast indices (active constraint indices), *I* denote the remaining (inactive constraint) indices, $\mathbf{C}(I, A)$ be the submatrix of C consisting of rows in *I* and columns in *A*, and $\mathbf{w}(A)$ denote the clipped weights assigned. Then, using the fact that **R** is diagonal, we may partition the matrices in eq. (8) and solve for $\mathbf{w}(I)$ via

$$(10)\quad \begin{bmatrix}\mathbf{C}(\boldsymbol{I},\boldsymbol{I}) + \mathbf{R}(\boldsymbol{I},\boldsymbol{I}) & \mathbf{1}\\ \mathbf{1}^{\mathbf{T}} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{w}(\boldsymbol{I})\\ \boldsymbol{\lambda}\end{bmatrix} = \begin{bmatrix}\mathbf{R}(\boldsymbol{I},\boldsymbol{I})\mathbf{w}^{\boldsymbol{G}}(\boldsymbol{I}) - \mathbf{C}(\boldsymbol{I},\boldsymbol{A})\mathbf{w}(\boldsymbol{A})\\ \mathbf{1} - \mathbf{1}^{\mathbf{T}}\mathbf{w}(\boldsymbol{A})\end{bmatrix}$$

Eq. (10) may be applied repeatedly, with any weights that violate the constraints being "clipped" and assigned to *A* until all weights satisfy the constraints. This method may be extended to an exact solution using the approach described in Bro and De Jong (1997).

Aggregation

The discussion up to this point has focused on determining weights and biases for a single location, generation and lead time. However, in many situations, input forecast error statics are similar for neighboring locations and times. Aggregating error covariance and bias estimates provides a reduction in random error at the expense of incurring an increase in representativeness error. The representativeness error is limited by defining a neighborhood (or kernel) of limited displacements in location, altitude, generation time and valid time of day, and by restricting the contribution of neighbors. Thus, for $\zeta_C \in [0,1]$ and covariance matrices computed at the nearest M_C neighborhood points denoted with superscripts, we may compute

$$(11)\quad \bar{\mathbf{C}}_t^0 = (1-\zeta_C)\mathbf{C}_t^0 + \zeta_C \frac{\mathbf{1}}{M_C}\sum_{\boldsymbol{k}=\mathbf{0}}^{M_C}\mathbf{C}_t^k$$

The simple average in eq. (11) may be replaced with variable weights for each neighbor based on its distance via a Gaussian kernel function, for instance. The aggregate $\bar{\mathbf{C}}^0$ is used in place of **C** in the regularized regression problem described by eq. (7). An aggregated $\bar{\mathbf{b}}$ may be defined similarly and used in place of **b** in producing the consensus forecast via eq. (1). However, bias aggregation has shown limited value in our temperature forecasting experiments and is not used in the results presented below.

AR consensus forecasting system

To summarize, the AR consensus forecasting method involves the following steps:

(1) maintain a historical dataset of input forecast values and verifying observations at target sites;
(2) at a forecast generation time, compute input forecast biases and error covariances via eqs. (3) and (4);
(3) aggregate error covariances (optionally, biases) via eq. (11) and compute the regularization matrix R;
(4) solve the quadratic program in eq. (7) for the consensus weights, e.g., using eqs. (9) and (10); and
(5) produce the consensus forecast via eq. (1).

The AR method requires specification of a number of parameters, including the bias and error covariance learning rates, bias modulation parameter, aggregation coefficients, regularization parameters, and weight constraints and goal. Weight constraints and goal will generally be specified based on extrinsic criteria. For the others, the AR system may learn optimal parameters via offline training between forecast cycles to minimize root mean squared error (RMSE) or another skill metric over the past several forecast days, for instance. However, for the results presented below, we do not vary the parameters as a function of time, location, or input forecast.

Results

The AR consensus forecasting system was demonstrated using hourly temperature forecasts from an ensemble of 22 input forecasts including both NWP output and derived MOS products. The dataset spanned the period 14 November 2014 through 11 October 2015, with a few missing days. Surface temperature measurements from over 1200 ground weather station ("METAR") locations in the continental U.S were used as truth observations.

To evaluate performance, forecasts for 0-72 hour lead times generated at 0900 UTC each day for the METAR sites were compared with the observed surface temperatures. Results for several methods and for several choices of the AR method's adaptable parameters were computed and compared. These included the single best-performing input forecast (BF); the best input forecast with bias correction (BFB); the equal weights consensus (EW), which computes the simple mean of the bias-corrected input forecasts; the inverse variance weights method (VAR) mentioned earlier; the DICast stochastic gradient descent method (SGD); and adaptable regression (AR). For AR, five sets of parameters are shown, denoted AR000, AR100, AR010, AR001 and AR111, where the first digit represents whether or not bias modulation was used, the second whether or not regularization was used and the third whether or not covariance aggregation was used. For these AR results, the bias modulation μ = 1 or 0.8, regularization parameter β = 0 or 0.1, and error covariance aggregation proportion ζ_C = 0 or 0.7; the bias aggregation proportion ζ_b = 0.0 is fixed for all four. All 10 methods were allowed to "spin up" from 14 November 2014 – 31 January 2015, and then evaluated for 1 February – 11 October 2015. The AR results use a lookback period of up to 91 days (3 months), whereas the other methods are allowed to use all previous data. Cross-validation was not appropriate for this evaluation, since forecast consensus weights and biases were determined at each timestep from past data, then used to make a forecast 0-72 hours into the future which was subsequently verified. However, all experiments for determining good parameters were performed using a small number of odd-hour forecast lead times, whereas only performance comparison results for even forecast lead times are shown.

Results of the evaluation are summarized in Table 1 using the performance of the equally-weighted average of bias-corrected input forecasts, EW, as a reference. Although EW is a simple technique, the 22 input forecasts were selected based on their skill, so it is a valid benchmark. The first column contains the overall RMSE computed over all days, sites and lead-times; the other columns show the median and 90th percentile, respectively, of RMSEs computed over all days for each site and lead-time. The input forecast used for the best forecast (BF) evaluation was the one with the smallest RMSE over the entire dataset. Bias-correcting it using a dynamic bias calculation similar to eq. (3), with $\gamma = 0.05$ and $\mu = 1$, reduced RMSE substantially—more than 10%, as shown by the BFB row in the table. Averaging all of the bias-corrected input forecasts provided another large performance jump of about 10%, as shown by the EW RMSEs. These results clearly confirm the value of both dynamic bias correction and averaging the input forecasts, both of which have been well-established in the weather forecasting literature.

The inverse variance weighting (VAR) method reduces RMSE by over 1% from the EW results, and the DICast stochastic gradient descent algorithm (SGD) improves on EW by nearly 3%. Without bias modulation, regularization or covariance aggregation, the AR method reduces RMSE by a further 0.5% or more, as shown by the AR000 row. In this and all other AR runs shown in this paper, the bias and error covariance learning rates were fixed at $\gamma = 0.05$ and $\eta = 0.03$; the goal weight $\mathbf{w}^G = 0$; and the weight limits were $\mathbf{w}^L = 0$ and $\mathbf{w}^U = 1$. Incorporating bias modulation, regularization, covariance aggregation, or all three together results in successive improvements, with AR111 representing an improvement of nearly 6% over the EW benchmark. While this improvement is less impressive than the large gains achieved by dynamic bias correction and simple ensemble forecast averaging, the improvement of AR111 over SGD—the current state-of-the-art—is more than the improvement of SGD over EW. Since performance gains are increasingly hard-won as forecast accuracy improves, we consider this a successful outcome. AR001 (baseline plus covariance aggregation) achieves roughly 80% of this improvement, whereas bias modulation and regularization have smaller but still significant effects.

Table 1: Performance comparison results represented as percentages relative to EW performance, as described in the text.

	Rel. Tot. RMSE (%)	Rel. Median RMSE (%)	Rel. 90 pctile RMSE (%)
BF	128.6	125.2	127.9
BFB	110.8	111.3	109.4
EW	100.0	100.0	100.0
VAR	98.7	98.8	98.8
SGD	97.3	97.3	97.8
AR000	96.7	96.8	97.1
AR100	96.4	96.6	96.7
AR010	96.1	96.2	96.5
AR001	94.7	94.7	95.2
AR111	94.3	94.3	94.8

To further illustrate the contributions of the adaptable bias modulation, regularization, and covariance aggregation parameters, Figure 1 shows how the RMSE changes as μ, β, and ζ_C are varied in turn from the AR000 configuration of $\mu = 1$, $\beta = 0$, and $\zeta_C = 0$ (dotted lines), or from the AR111 selection of $\mu = 0.8$, $\beta = 0.1$, and $\zeta_C = 0.7$ (solid lines). These results were computed from only four representative lead times—11, 27, 39 and 55 hours—to limit computation time. Decreasing μ from 1.0 to 0.8 reduced RMSE by about 0.04% in both AR000 and AR111 scenarios. Beginning with the AR000 configuration, increasing the regularization factor β reduced the RMSE by about 1% for values between 0.3 and 1.0; in AR111, however, regularization reduced RMSE by only 0.1% at $\beta = 0.1$, and increased it for $\beta > 0.3$. Thus, regularization appears less valuable, and is optimal at a smaller parameter value, when covariance aggregation is used. The proportion of aggregated covariances, ζ_C, had the largest effect on performance; for values between 0.7 and 0.9, it reduced RMSE by about 2% from AR000, and in the AR111 scenario was responsible for a decrease of about 1.5%.

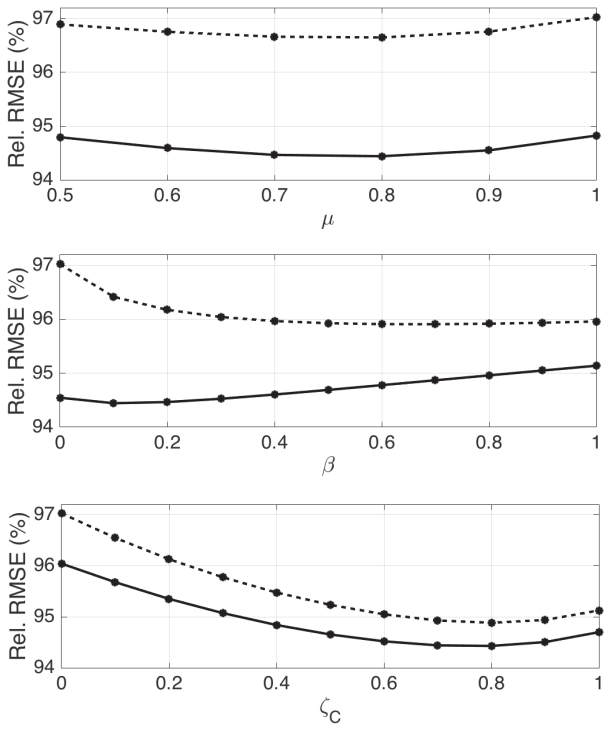

Figure 1: AR performance sensitivity results for bias modulation (top), regularization (middle) and covariance aggregation (bottom) parameters, as described in the text.

Finally, we consider how the AR improvements in consensus forecast accuracy vary spatially and temporally. Figure 2 shows a spatial comparison of the RMSE of AR111 relative to EW, where the RMSE is computed for each site over all days and even hour forecast lead times from 0-72 hours. The AR111 consensus performs better at every site, as indicated by the fact that no values exceed 100%. The typical improvement of AR111 over EW is around 6%. However, much greater gains were made at many sites along the west coast and several in the intermountain west. We hypothesize that many of these sites are in or near highly variable terrain, and that the input forecasts in the ensemble are both less skillful and more variable there, making a careful choice of weights more beneficial. A similar analysis of monthly RMSE shows that AR111 scored better than EW in every month: over 7% better in February-March, about 5.5% better in April and May, and nearly 4% better in June – September. Thus, AR111 shows the largest relative improvement over EW in the winter months, which have the greatest temperature forecast errors, and less in the less volatile summer months when both AR111 and EW RMSEs were about 30% lower than their winter highs.

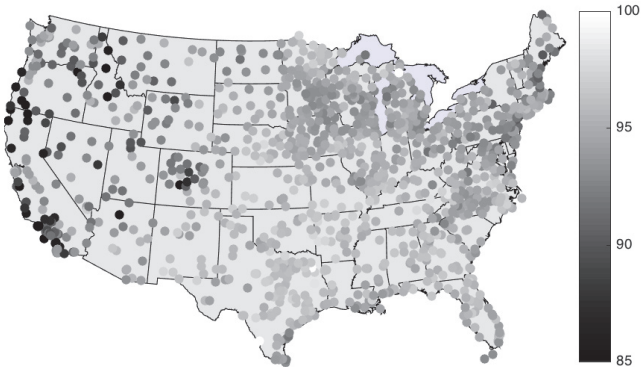

Figure 2: AR111 RMSE as a percentage of EW RMSE for each site, with values represented via the colorscale shown at right.

Conclusion

This paper has motivated and described an adaptable regression technique for consensus prediction, couching it in the context of ensemble-based deterministic weather forecasting. The method requires a history of past observations and input forecast values. An exponential decay factor is used to discount the influence of older performance data in both bias and error covariance computations, and input forecast error covariance matrices and biases from neighboring sites and times may be aggregated to reduce the effect of random "noise" in the input forecasts and observations. At each forecast generation time, biases and error covariance matrices are assembled for the available input forecasts. Weight bounds and a preferred weight solution, or goal, may be specified. The input forecast error covariance matrix diagonal is inflated to provide regularized re-

gression ("ridge regression") that mitigates overfitting by driving the solution toward the specified goal weights. Similarly, the computed biases are "modulated" by a multiplicative factor between 0 and 1 to accommodate a prior belief that the bias estimates should tend toward 0. Using these ingredients, the AR method produces a quadratic program for each site whose solution is the set of combination weights to be used in producing the final consensus forecast via a weighted average of the bias-corrected input forecasts. Between forecast generation cycles, the adaptable algorithm parameters may be adjusted based on the most recent performance data, allowing the system to adapt to seasonal, synoptic, or NWP model changes.

The AR method was illustrated using 0-72 hour, multi-season consensus temperature forecasting for over 1200 METAR sites in the continental U.S. based on a 22-member forecast ensemble. AR showed much better performance than the best individual input forecast, a simple average of the bias-corrected input forecasts, or inverse variance weighting, and it showed more modest but still significant improvement over the legacy DICast method. Additionally, sensitivity tests showed that bias modulation, regularization and covariance aggregation improved RMSE, but an investigation of different parameter sets showed that their effects were not additive. The biggest improvement appeared to be produced by error covariance aggregation, while regularization demonstrated significant improvement when aggregation wasn't used and bias modulation had a smaller but consistent effect. Further improvements may result from dynamically selecting AR parameter values by testing alternatives empirically between forecast cycles, or via methods along the lines of the *L*-curve criterion for ridge regression (Hansen 1992, Calvetti et al. 2004). However, our initial exploration of these ideas has not identified a performance benefit that merits the additional computational cost.

Given the widespread use and importance of consensus prediction, even a marginal improvement in accuracy can have significant benefit for a number of domains. For instance, accurately forecasting weather can help communities better prepare for and mitigate the damaging effects of severe weather on lives and property, and consensus forecasts of climate impacts are important for informed long-term planning. Short and mid-term forecasts of temperature, wind and solar radiation are becoming increasingly important to utilities managing portfolios of wind and solar farms in addition to traditional power plants, balancing production with consumer power demand (which is also highly influenced by weather). Numerous other endeavors, from transportation to retail logistics, will also benefit from more accurate predictions.

References

Breiman, L. 1996. Stacked Regressions. *Machine learning*, 24(1): 49-64.

Bro, R. and De Jong, S. 1997. A Fast Non-negativity-constrained Least Squares Algorithm. *Journal of Chemometrics* 11(5): 393-401.

Engel, C. and Ebert, E. E. 2012. Gridded Operational Consensus Forecasts of 2-m Temperature over Australia. *Wea. Forecasting* 27(2): 301-322.

Calvetti, D.; Reichel, L.; and Shuibi, A. 2004. L-curve and Curvature Bounds for Tikhonov Regularization. *Numerical Algorithms* 35(2-4): 301-314.

Clemen, R. T. 1989. Combining Forecasts: A Review and Annotated Bibliography. *International Journal of Forecasting*, 5(4), 559-583.

Engel, C. and Ebert, E. 2007. Performance of Hourly Operational Consensus Forecasts (OCFs) in the Australian Region. *Wea. Forecasting*, 22, 1345–1359.

Gerding, S. and Myers, B. 2003: Adaptive Data Fusion of Meteorological Forecast Modules. In 3rd American Meteorological Society Conference on Artificial Intelligence Applications, Paper 4.8.

Hansen, P. C. 1992. Analysis of Discrete Ill-posed Problems by Means of the L-curve. *SIAM Review*, 34, 561-580.

Koval, J. P.; Rose, B.; Neilley, P.; Bayer, P.; McDonald, J.; Cassanova, B.; Winn, D.; Yang, E.; and Celenza, J. 2015. 1-15 Day Weather Forecast Guidance at The Weather Company. In 27th American Meteorological Society Conference on Weather Analysis and Forecasting, Paper 12B.8.

Krishnamurti, T. N.; Kishtawal, C. M.; LaRow, T. E.; Bachiochi, D. R.; Zhang, Z.; Williford, C. E.; Gadgil, S.; and Surendran, S. 1999. Improved Weather and Seasonal Climate Forecasts from Multimodel Superensemble. *Science* 285: 1548–1550.

Kuncheva, L. I. 2004. *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley & Sons.

Lazo, J. K.; Lawson, M.; Larsen, P. H.; and Waldman, D. M. 2011: U.S. Economic Sensitivity to Weather Variability. *Bull. Amer. Meteor. Soc.* 92: 709–720.

Myers, W., and Linden, S. 2011. A Turbine Hub Height Wind Speed Consensus Forecasting System. In 91st American Meteorological Society Annual Meeting, 22-27.

Peña, M. and van den Dool, H. 2008. Consolidation of Multimodel Forecasts by Ridge Regression: Application to Pacific Sea Surface Temperature. *Journal of Climate* 21(24): 6521-6538.

Thompson, P. D. 1977. How to Improve Accuracy by Combining Independent Forecasts. *Mon. Wea. Rev.* 105: 228–229.

Wilks, D. S. 2011. *Statistical Methods in the Atmospheric Sciences, 3rd Ed.*: Academic Press.

Wolpert, D. H. 1992. Stacked Generalization. *Neural Networks*, 5: 241–259.

Woodcock, F. and Engel, C. 2005. Operational Consensus Forecasts. *Wea. Forecasting* 20: 101–111.