Supplement (S1 Text) for "Accuracy of Real-Time Multi-Model Ensemble Forecasts for Seasonal Influenza in the U.S."

Nicholas G Reich, Craig McGowan, Teresa Yamana, Abhinav Tushar, Evan L Ray,
Dave Osthus, Sasikiran Kandula, Logan Brooks
Willow Crawford-Crudell, Graham Casey Gibson, Evan Moore, Rebecca Silva
Matthew Biggerstaff, Michael A Johansson, Roni Rosenfeld, Jeffrey Shaman

November 5, 2019

1 Component models

See Table A.

2 Supplemental evaluation metrics

2.1 Bias and MSE

We compared the `FSNetwork-TTW` ensemble model's accuracy in 2017/2018 to the top-performing models from each team in the training phase. We used the metrics of root mean-squared error (RMSE) and average bias to measure accuracy of point estimates. Note that the ensemble weights were optimized solely to maximize log-score, so these accuracy scores are not an indicator of how well the ensemble could do if it were optimized to minimize point-estimate error. Consistent with the CDC scoring rules, we only evaluated point estimates within the "scoring bounds" specific to each target, region, and season (see Methods in main manuscript).

Overall, the `FSNetwork-TTW` model ranked second in both RMSE and average bias, behind the `LANL-DBM` model (Figure A). All selected models showed a negative bias (i.e. underestimation, on average) of the targets on the wILI scale (week-ahead incidence and peak percentage). The `CU-EKF_SIRS` model showed particularly low bias for 1- and 2-week-ahead foreacasts, although greater variability led to lower ranks for RMSE.

In general, these results suggest that using separate weighting schemes for point estimates and predictive distributions may be valuable.

Team	Model Abbr	Model Description	Ext. Data	Mech. Model	MM Ens.
FSNetwork	EW	Equal Weights (number of estimated weights = 0)			x
	CW	Constant Weights (20)			x
	TTW	Target-Type Weights (40)			x
	TW	Target Weights (140)			x
	TRW	Target-Region Weights (1,540)			x
CU	EAKFC_SEIRS†	Ensemble Adjustment Kalman Filter SEIRS[1]	x	x	
	EAKFC_SIRS†	Ensemble Adjustment Kalman Filter SIRS[1]	x	x	
	EKF_SEIRS†	Ensemble Kalman Filter SEIRS[2]	x	x	
	EKF_SIRS†	Ensemble Kalman Filter SIRS[2]	x	x	
	RHF_SEIRS†	Rank Histogram Filter SEIRS[2]	x	x	
	RHF_SIRS†	Rank Histogram Filter SIRS[2]	x	x	
	BMA	Bayesian Model Averaging[3]			
Delphi	BasisRegression*	Basis Regression (`epiforecast` defaults)[4]			
	DeltaDensity1*	Delta Density (`epiforecast` defaults)[5]			
	EmpiricalBayes1*	Empirical Bayes (conditioning on past 4 weeks)[6, 4]			
	EmpiricalBayes2*	Empirical Bayes (`epiforecast` defaults)[6, 4]			
	EmpiricalFuture*	Empirical Futures (`epiforecast` defaults)[4]			
	EmpiricalTraj*	Empirical Trajectories (`epiforecast` defaults)[4]			
	DeltaDensity2*	Markovian Delta Density (`epiforecast` defaults)[5]			
	Uniform*	Uniform Distribution			
	Stat	Ensemble (combination of 8 Delphi models)[5]			x
LANL	DBM	Dynamic Bayesian SIR Model with discrepancy[7]		x	
ReichLab	KCDE	Kernel Conditional Density Estimation[8]			
	KDE	Kernel Density Estimation and penalized splines[9]			
	SARIMA1	SARIMA model without seasonal differencing[9]			
	SARIMA2	SARIMA model with seasonal differencing[9]			
FluSight	unweighted_avg	Average of all models submitted to the CDC[10]			x

Table A: List of models, with key characteristics. New ensemble models introduced by this paper are indicated with the prefix FSNetwork. Component models contributed by individual teams are grouped by team with team-specific prefixes as follows: CU = Columbia University, Delphi = Carnegie Mellon, LANL = Los Alamos National Laboratory, ReichLab = University of Massachusetts Amherst, FluSight = CDC challenge organizers. The FluSight model was not included in the collaborative multi-model ensemble, but is used as a reference multi-model ensemble in the analysis. The 'Ext data' column notes models that use data external to the ILINet data from CDC. The 'Mech. model' column notes models that rely to some extent on a mechanistic or compartmental model of infectious disease transmission.[11] The 'MM Ens.' column indicates models that are multi-model ensembles. Note that some of the components were not designed as standalone models (marked with *) and others used single-model ensemble methodologies (marked with †) (see Methods for more details). S(E)IRS abbreviations stand for Susceptible (Exposed) Infectious Recovered Susceptible models of disease transmission and SARIMA stands for Seasonal Auto-Regressive Integrated Moving Average Model (see references for details).

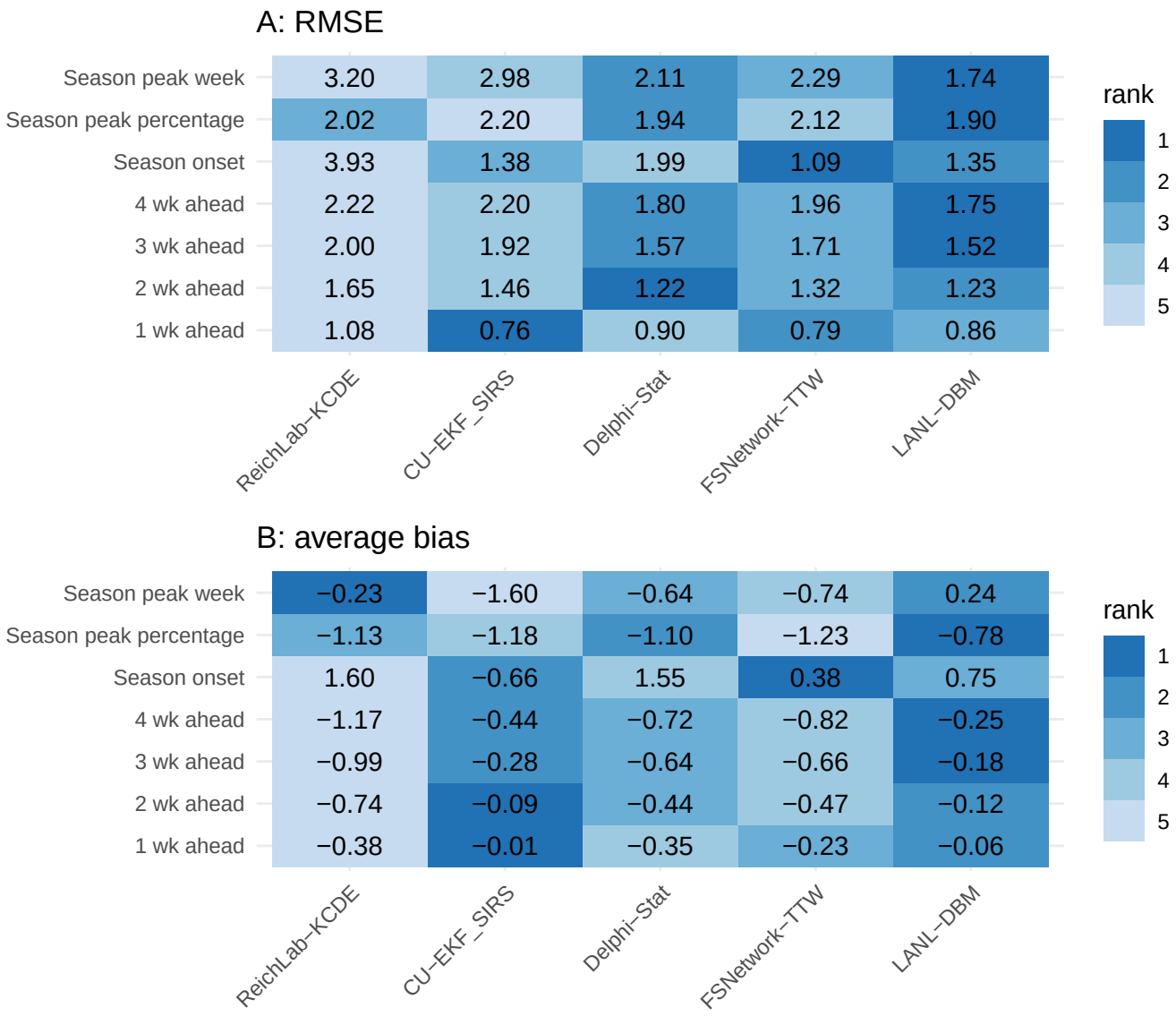

Figure A: Root mean squared error (RMSE, panel A) and bias (panel B) by target for selected models (with rank) in the 2017/2018 season. Evaluations are for all weeks in the 2017/2018 season. Models are sorted with lowest RMSE on right.

2.2 Probability integral transform

The Probability Integral Transform (PIT) is an evaluation metric that can be used to assess the calibration of a predictive model. A common application of PIT is testing whether a set of values from an unknown target distribution can be accurately modeled by one or more predictive distributions. In brief, statistical theory tells us that if we plug the set of observed values (which come from the unknown target distribution) into the cumulative distribution function of the predicted distribution, the output, otherwise known as the PIT values, should be uniformly distributed if the predicted distribution matches the true distribution.[12, 13] Therefore, looking at the PIT values provides a quantitative and qualitative assessment of the predictive model calibration by comparing the shape of the histogram of PIT values to a uniform distribution. Intuitively, the PIT measures how often a model's probabilistic assessment is true, i.e. does an event that the model says has a 10% chance of occuring really only occur 10% of the time. Systematic deviations from the expected uniform distribution may indicate lack of calibration in some aspects of the predictive model.

In our application, we evaluate the set of all probabilistic forecasts of the five targets on the wILI scale (1 through 4 week-ahead wILI percent and the peak percentage) from the `FSNetwork-TTW` model using PIT. For the 2017/2018 influenza season, we obtained a PIT value from each predictive distribution based on region, target, and week of season. As in other evaluations presented here, we only considered forecasts from the time-period of interest for the CDC, depending on the timing of the peak for each region-season. We rounded each PIT value to the nearest tenth of a decimal place and plotted them on a historgram with ten bars, one for each decile of the Uniform(0,1) distribution. We computed a Monte Carlo confidence interval under the null hypothesis that the PIT values are independent and identical draws from a Uniform(0,1) distribution, conditional on the number of PIT values.

In the 2017/2018 season, the `FSNetwork-TTW` model showed good calibration for all week-ahead targets (Figure C). The models appeared to be slightly better calibrated for shorter forecast horizons (i.e. 1- and 2-week ahead) than for longer horizons. Forecasts for the peak percentage were less well calibrated, with more forecasts than expected occuring in both low and high tails of the predictive distribution.

Across all training seasons, the `FSNetwork-TTW` model showed some lack of calibration for all targets considered (Figure B). In particular, the predictive distributions appeared to be too wide, with eventually observed values falling in the central region of the distribution more than expected. A slight negative bias is evident as well, from the skewness of the PIT figures, with the observed values more likely to fall under the median of the distribution for 2-, 3-, and 4-week-ahead forecasts. However, these forecasts were evaluated on only seven seasons worth of data, and given that the forecasts were better calibrated in a larger-than-usual season such as 2017/2018 (Figure C) suggests that the model in the training phase may have been appropriately allowing for the possibility of such a large season.

All in all, there could be some improvement in model calibration as measured by PIT. However, to date, this has not been designated as an explicit target for optimization of the ensemble weighting schemes.

3 Updated weights incorporating 2017/2018 performance

See Figure D.

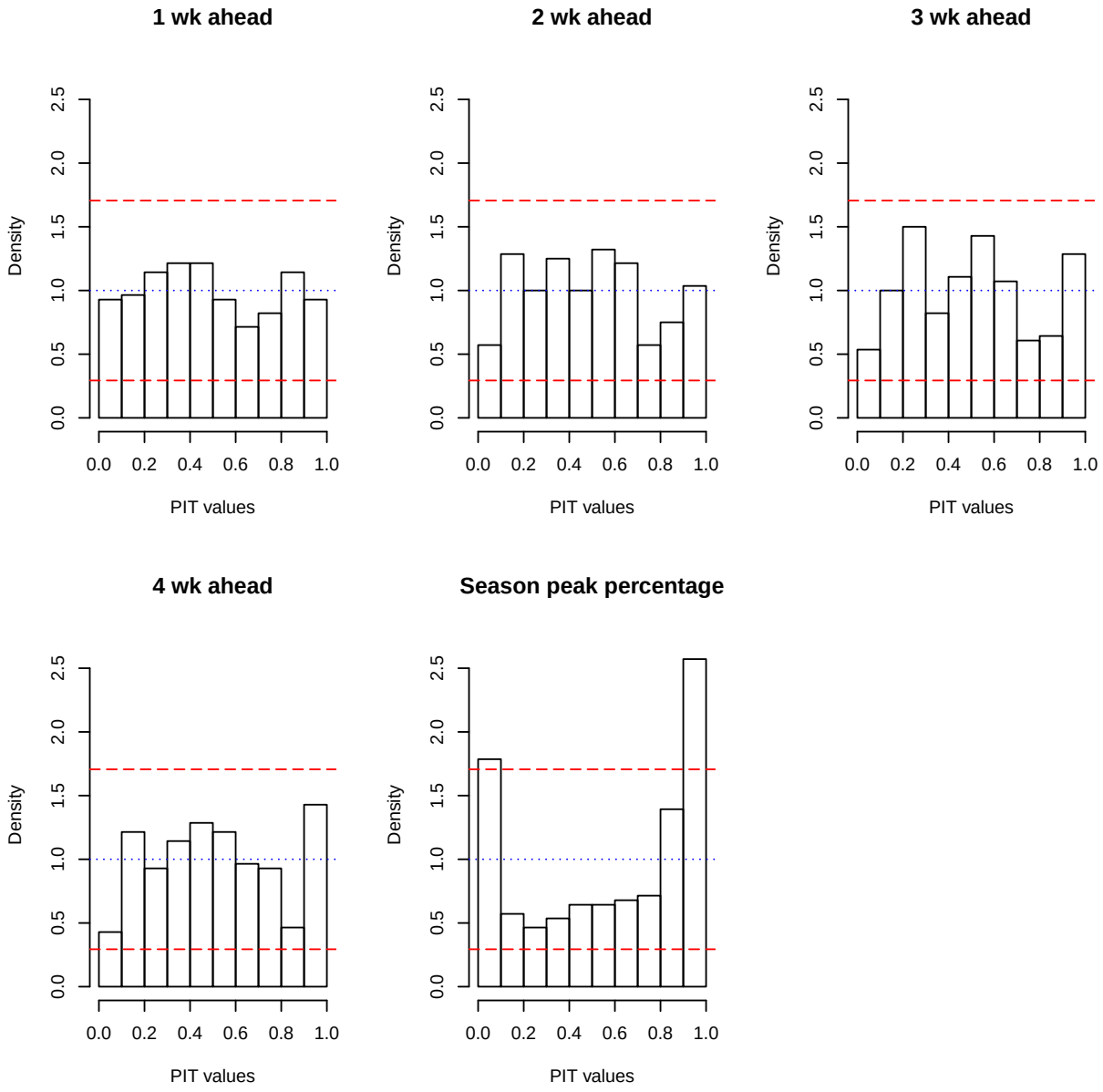

Figure B: Probability integral transform histograms by target for the FSNetwork-TTW model in the 2017/2018 season. If the model is well-calibrated, the histogram should resemble a uniform distribution, i.e. all the bars should be level at y=1. The red dashed lines represent a Monte Carlo confidence interval under the null hypothesis that the PIT values follow a Uniform(0,1) distribution. The fact that some of the bars for the season peak percentage target lie outside the CI bounds suggest that the model shows significantly weak calibration for that particular target.

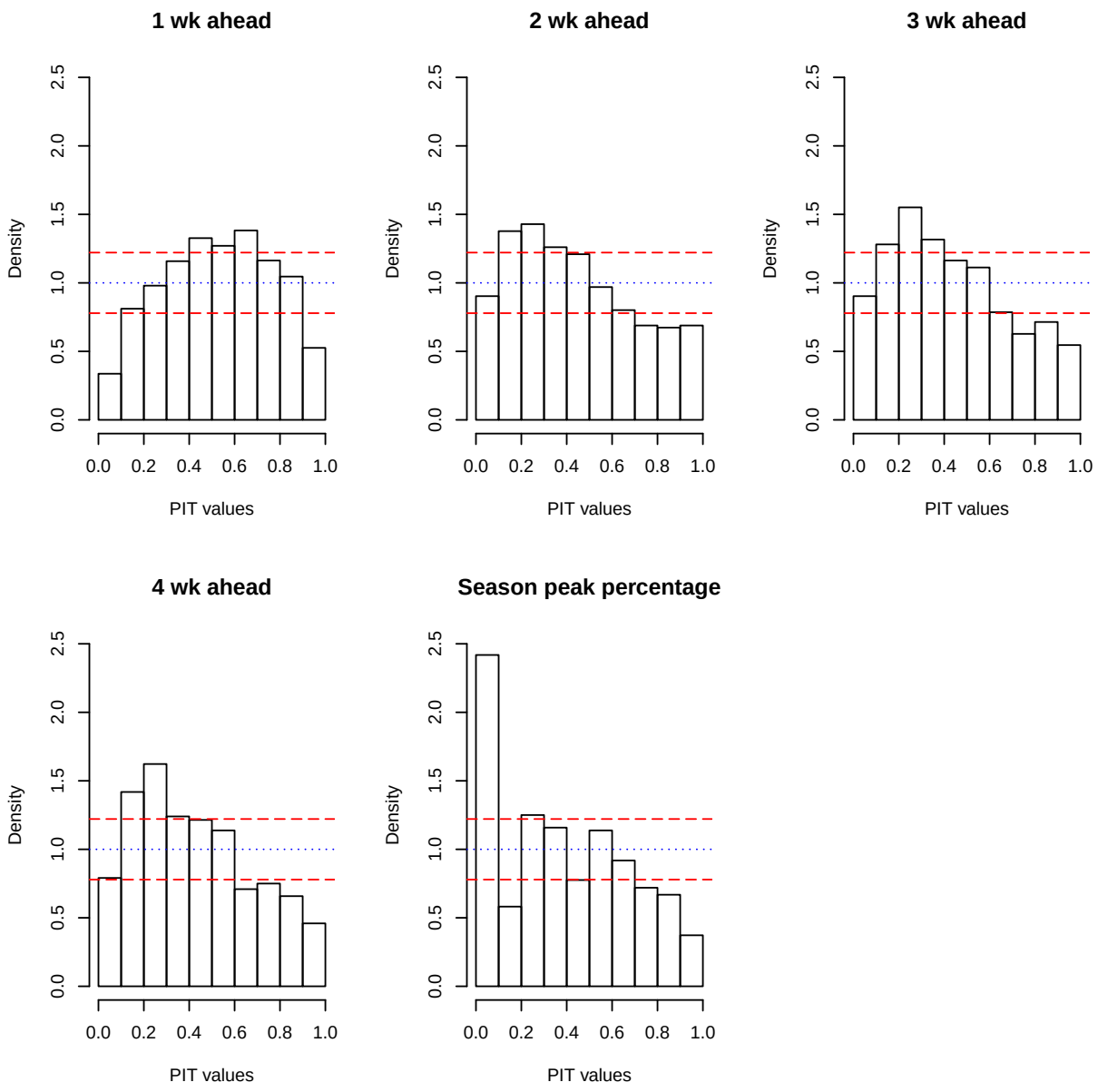

Figure C: Probability integral transform histograms by target for the FSNetwork-TTW model in all training seasons: 2010/2011 through 2016/2017. If the model is well-calibrated, the histogram should resemble a uniform distribution, i.e. all the bars should be level at y=1. The red dashed lines represent a Monte Carlo confidence interval under the null hypothesis that the PIT values follow a Uniform(0,1) distribution. The intervals are narrower here than in Figure B because there are more observations from all the training seasons combined than in the one testing season. The model is showing some significant lack of calibration for all targets. In particular for the season peak percentage, substantially more observations were in the lowest 10% of the predictive distributions than would have been expected due to chance.

Figure D: The change in the estimated weight for each model after including the 2017/2018 performance results.

4 Comparison of ensemble model performance in testing and training phases

To compare the training and test phase performance of the ensemble models, we plotted their relative performance (Figure E). This shows that the `FSNetwork-TTW` model had the highest overall score in both the training and test phase.

Figure E: Overall test and training phase performance scores for the five ensemble models. Displayed scores are averaged across targets, regions, and weeks, and plotted separately for selected models. Model ranks within each row are indicated by color of each cell (darker colors indicate higher rank and more accurate forecasts) and the forecast score (rounded to three decimal places) is printed in each cell.

5 Forecast accuracy across the 2017/2018 test season

We compared the average weekly accuracy of three models –`FSNetwork-TTW`, `FSNetwork-EW`, and `ReichLab-KDE`– over time in the 2017/2018 test season (Figure F). This comparison showed that a performance-weighted ensemble, `FSNetwork-TTW`, consistently outperformed an equally weighted ensemble, `FSNetwork-EW`, by a small margin and a seasonal average model, `ReichLab-KDE`, by a larger margin. Accuracy for week-ahead targets was substantially lower in the middle of the season.

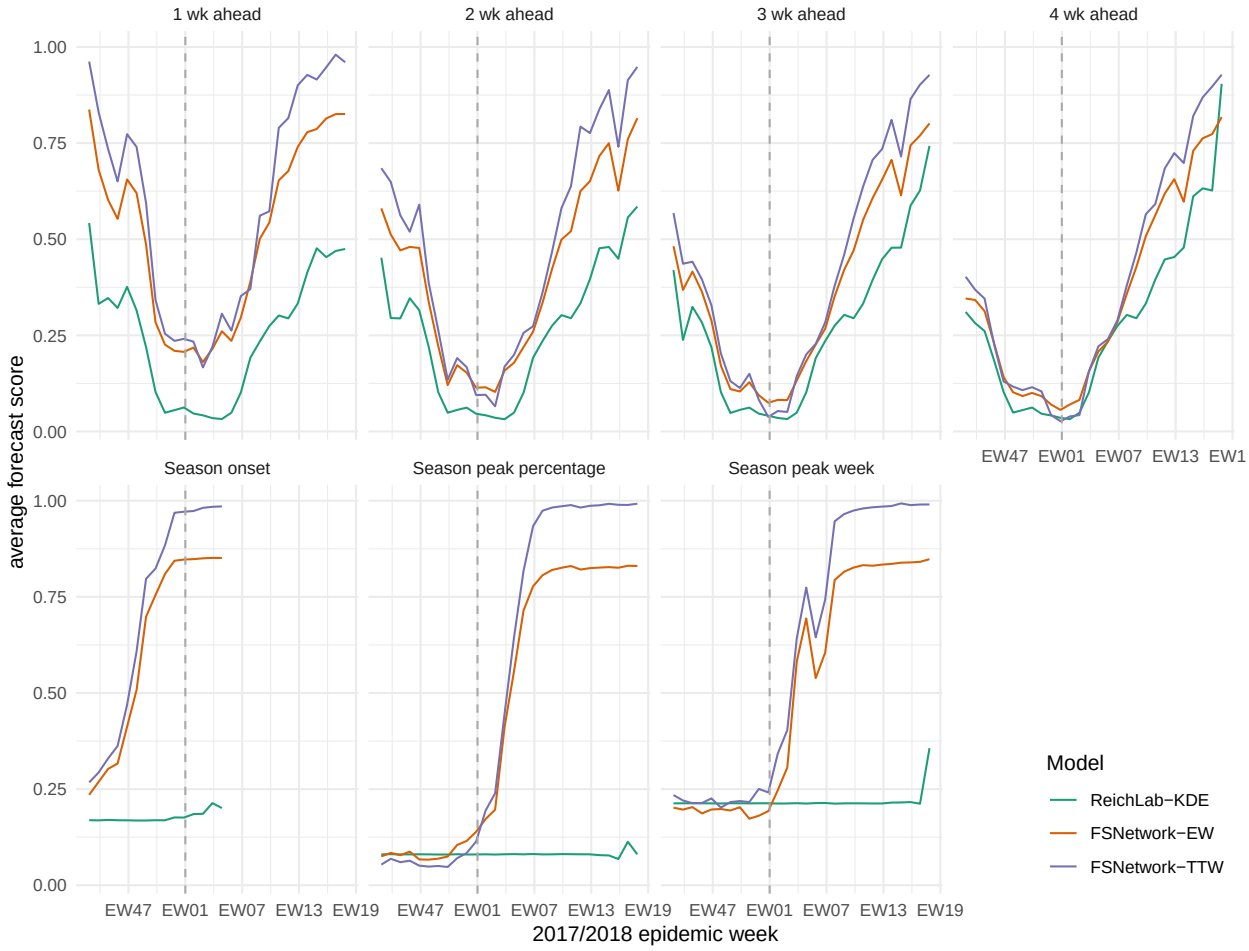

Figure F: Forecast scores averaged across all regions for three selected models in the 2017/2018 test season. The x-axis shows the epidemic week (EW) of the season. Scores are only shown for scored weeks according to FluSight guidelines (see Methods in original manuscript). Therefore, season onset scores are truncated six weeks after the last season onset occurred. A vertical dashed line indicates January 1, 2018.

6 EM Algorithm for Weighted Density Ensembles

The following is adapted from [14, 15] and describes the use of the Expectation Maximization (EM) algorithm for constructing weighted average ensemble models in the context of infectious disease forecasting. Our goal is to develop a weighted density ensemble that combines the full predictive distributions in such a way as to optimize the score of the resulting model average.

To use the EM algorithm to find optimal weights, we formulate the question as a missing data problem. We consider a data generating process in which an observed target is generated from $f(z)$ by choosing one of the $f_c(z)$ component distributions as a random draw from a multinomial distribution with probabilities π_c. Here we supress the subscripts for target, region and week (t, r, and w) for simplicity. The problem is that we do not know, for each observed datapoint z_i^*, which component this observation was drawn from. However, we can make a best guess, conditional on the data and our current estimates of π_c, of how often each component was chosen. This is the "E step". Then, based on these guesses, we can update our estimate of π_c. This is the "M-step".

The "E step" of the EM algorithm we can think of as determining, for each component c, the expected number of times for each of our observed N datapoints that component c was chosen as the contributor to $f(z)$:

$$\mathbf{E}[model_c|data] = \sum_i \frac{\pi_c f_c(z_i^*)}{f(z_i^*)} \tag{1}$$

Heuristically, we can think of the expression $f_c(z)$ equivalently as $Pr(z|model_c)$ or in words the likelihood of seeing the value z given that component c is the "chosen" model.

The "M step" of the EM algorithm simply calculates, conditional on the "complete data", i.e. the z_i^* and the estimated number of times each component was chosen, the fraction of times each method was chosen. Therefore, if we simply divide the quantity from the "E-step" by N, our total number of observations, we obtain a new estimate of this probability:

$$\pi_c^{(k+1)} = \frac{1}{N}\mathbf{E}[model_c|data] \tag{2}$$

$$= \frac{1}{N}\sum_i \frac{\pi_c^{(k)} f_c(z_i^*)}{f(z_i^*)} \tag{3}$$

Assume that we have a set of C fitted predictive densities "evaluated at" observed data z_i^* for $i = 1, ..., N$. In our application, we let the $f_c(z_i^*)$ be computed as the probabilities associated with the modified scores as described in the main manuscript. As an example, for season peak percentage and the short-term forecasts, probabilities assigned to wILI values within 0.5 units of the observed values are included as correct, so the modified score becomes $f_c(z_i^*) = \int_{z_i^*-.5}^{z_i^*+.5} f_c(z|\mathbf{x})dz$. We will notate these scores as $f_c(z_i^*|\mathbf{x})$. There will be $C \cdot N$ total observations, as each model must have an associated score (a probability, between 0 and 1) for each observed data point.

We wish to obtain a set of optimal weights $\tilde{\pi} = \{\tilde{\pi}_1, \tilde{\pi}_2, ..., \tilde{\pi}_C\}$ for combining the models such that $\forall c\ \tilde{\pi}_c \geq 0$ and $\sum_{c=1}^{C} \tilde{\pi}_c = 1$. The weights can be used to then combine the component models into an ensemble model

as

$$f(z|\pi) = \sum_{c=1}^{C} \pi_c f_c(z).$$

We define a function $\ell(\pi)$ that computes a log-likelihood of the resulting ensemble as follows:

$$\ell(\pi) = \frac{1}{N} \sum_{i=1}^{N} \log f(z_i|\pi).$$

Below, we define one procedure to obtain a set of weights for the ensemble.

Algorithm 1 Degenerate Expectation Maximization (DEM) algorithm

1: **procedure** DEM(...)
2: Initialize $\pi_c^{(0)}$ such that $\forall c\ \pi_c^{(0)} \geq 0$ and $\sum_{c=1}^{C} \pi_c^{(0)} = 1$
3: Set $t = 0$
4: Set $\Delta = 1$, or another arbitrary constant.
5: Set ϵ to be a very small positive number strictly less than Δ.
6: **while** $\Delta > \epsilon$ **do**
7: Set $t = t + 1$
8: Update weights, $\forall c$, $\pi_c^{(t)} = \frac{1}{N} \sum_{i=1}^{N} \frac{\pi_c^{(t-1)} f_c(z_i)}{f(z_i|\pi^{(t-1)})}$
9: Set $\Delta = \frac{\ell(\pi^{(t)}) - \ell(\pi^{(t-1)})}{|\ell(\pi^{(t)})|}$
10: **return** $\tilde{\pi} = \tilde{\pi}^{(t)}$

And note that in Algorithm 1, Step 9 it should always be the case that $\ell(t) \geq \ell(t-1)$.

We note that this application of the EM algorithm is a very simple example of the standard EM, which in general does not guarantee a global maximum. However, this particular log-likelihood function (a sum of logarithms of weighted sums) is a convex function of its parameters, the π_c, and convexity ensures that any local maximum is a global maximum [16]. In particular, we can ensure that there is a global, finite maximum and that the EM finds it by including a uniform component and making the algorithm start with all nonzero weights.

References

[1] Sen Pei and Jeffrey Shaman. Counteracting structural errors in ensemble forecast of influenza outbreaks. *Nature Communications*, 8(1):925, dec 2017.

[2] Wan Yang, Alicia Karspeck, and Jeffrey Shaman. Comparison of Filtering Methods for the Modeling and Retrospective Forecasting of Influenza Epidemics. *PLOS Computational Biology*, 10(4):e1003583, apr 2014.

[3] Teresa K. Yamana, Sasikiran Kandula, and Jeffrey Shaman. Individual versus superensemble forecasts of seasonal influenza outbreaks in the United States. *PLOS Computational Biology*, 13(11):e1005801, nov 2017.

[4] Logan C Brooks, David C Farrow, Sangwon Hyun, Ryan J Tibshirani, and Roni Rosenfeld. epiforecast: Tools for forecasting semi-regular seasonal epidemic curves and similar time series. https://github.com/cmu-delphi/epiforecast-R, 2015.

[5] Logan C. Brooks, David C. Farrow, Sangwon Hyun, Ryan J. Tibshirani, and Roni Rosenfeld. Nonmechanistic forecasts of seasonal influenza with iterative one-week-ahead distributions. *PLOS Computational Biology*, 14(6):e1006134, jun 2018.

[6] Logan C. Brooks, David C. Farrow, Sangwon Hyun, Ryan J. Tibshirani, and Roni Rosenfeld. Flexible Modeling of Epidemics with an Empirical Bayes Framework. *PLOS Computational Biology*, 11(8):e1004382, aug 2015.

[7] Dave Osthus, James Gattiker, Reid Priedhorsky, and Sara Y Del Valle. Dynamic Bayesian influenza forecasting in the United States with hierarchical discrepancy. *Bayesian Analysis*, 2018.

[8] Evan L. Ray, Krzysztof Sakrejda, Stephen A. Lauer, Michael A. Johansson, and Nicholas G. Reich. Infectious disease prediction with kernel conditional density estimation. *Statistics in Medicine*, 36(30):4908–4929, sep 2017.

[9] Evan L. Ray and Nicholas G. Reich. Prediction of infectious disease epidemics via weighted density ensembles. *PLOS Computational Biology*, 14(2):e1005910, feb 2018.

[10] Craig McGowan, M Biggerstaff, Michael A. Johansson, K Apfeldorf, M Ben-Nun, L Brooks, M Convertino, M Erraguntla, D Farrow, J Freeze, S Ghosh, S Hyun, S Kandula, J Lega, Y Liu, N Michaud, H Morita, J Niemi, N Ramakrishnan, EL Ray, NG Reich, P Riley, J Shaman, R Tibshirani, A Vespignani, Q Zhang, and Carrie Reed. Collaborative efforts to forecast seasonal influenza in the United States, 2015-2016. *Nature Scientific Reports*, 9(683), 2019.

[11] Matt J Keeling and Pejman Rohani. *Modeling infectious diseases in humans and animals*. Princeton University Press, 2011.

[12] John E Angus. The probability integral transform and related results. *SIAM review*, 36(4):652–654, 1994.

[13] Francis X Diebold, Todd A Gunther, and Anthony S Tay. Evaluating density forecasts with applications to financial risk management. *International Economic Review*, 39(4):863–883, 1998.

[14] R Rosenfeld. The EM Algorithm. http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/11761-s97/WWW/tex/EM.ps, 1997.

[15] R Rosenfeld. The "degenerate EM" algorithm for finding optimal linear interpolation coefficients λ_i. https://www.cs.cmu.edu/~roni/11761/Presentations/degenerateEM.pdf, 2007.

[16] Thomas McAndrew and Nicholas G. Reich. Adaptively stacking ensembles for influenza forecasting with incomplete data. 2019.