Predicting seasonal influenza hospitalization using an ensemble super learner: a simulation study

Jason R. Gantenberg [1,2*], Kevin W. McConeghy [2,3], Jon Steingrimsson [4], Chanelle J. Howe, [5], Andrew R. Zullo, [1,2]

1 Department of Epidemiology, Brown University School of Public Health, 121 S. Main St., Providence, RI, 02912
2 Department of Health Services, Policy and Practice, 121 S. Main St., Providence, RI, 02912
3 Providence VA Medical Center, 830 Chalkstone Ave., Providence, RI, 02908
4 Department of Biostatistics, Brown University School of Public Health, 121 S. Main St., Providence, RI 02912
5 Department of Epidemiology, Center for Epidemiology and Environmental Health, Brown University School of Public Health, 121 S. Main St., Providence, RI, 02912

* Corresponding author: jrgant@brown.edu

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur eget porta erat. Morbi consectetur est vel gravida pretium. Suspendisse ut dui eu ante cursus gravida non sed sem. Nullam sapien tellus, commodo id velit id, eleifend volutpat quam. Phasellus mauris velit, dapibus finibus elementum vel, pulvinar non tellus. Nunc pellentesque pretium diam, quis maximus dolor faucibus id. Nunc convallis sodales ante, ut ullamcorper est egestas vitae. Nam sit amet enim ultrices, ultrices elit pulvinar, volutpat risus.

Author summary

scing elit. Curabitur eget porta erat. Morbi consectetur est vel gravida pretium. Suspendisse ut dui eu ante cursus gravida non sed sem. Nullam sapien tellus, commodo id velit id, eleifend volutpat quam. Phasellus mauris velit, dapibus finibus elementum vel, pulvinar non tellus. Nunc pellentesque pretium diam, quis maximus dolor faucibus id. Nunc convallis sodales ante, ut ullamcorper est egestas vitae. Nam sit amet enim ultrices, ultrices elit pulvinar, volutpat risus.

Note to co-authors: Two authors from Sanofi may be added after they review the manuscript.

Meta

Target journal:

- *PLoS Computational Biology*

Section:

- Epidemiology and Clinical/Translational Studies

Potential editors:

- Benjamin Althouse
- Miles Davenport
- Matthew Ferrari
- Roger Kouyos
- James Lloyd-Smith

Potential reviewers:

- Ryan J. Tibshirani (co-author on paper we use for curve simulation)
- Logan C. Brooks (co-author on paper we use for curve simulation)
- Roni Rosenfeld (co-author on paper we use for curve simulation)
- Sherri Rose
- David A. Osthus
- Samrachana Adhikari
- Oleg Sofrygin (sl3 package author)
- Nima Hejazi (sl3 package author)
- Jeremy Coyle (sl3 package author – I did ask him a question about parallelization with sl3, so it may not be appropriate to have him as a reviewer)

Introduction

Between 2010 and 2017, approximately 140,000–570,000 individuals have been hospitalized and 12,000–51,000 have died annually due to seasonal influenza in the United States [1]. Being able to predict how influenza-related hospitalizations will change over time during any given influenza season can assist policymakers, public health officials, and physicians allocate resources appropriately and prepare more efficiently for changes in hospitalization rates [2].

While influenza forecasting is a still-maturing science [3,4], researchers have made considerable progress over the past decade in improving the quality of and capacity for forecasting influenza-like illness (ILI) [cite], thanks in part to the FluSight forecasting competitions sponsored by the Centers for Disease Control and Prevention (CDC) since the 2013–14 flu season [3]. Many different types of models have been used to generate forecasts, including statistical time series models [3,5], Bayesian methods [4,6], and agent-based models [4], among others. However, ensemble methods have emerged as perhaps the most promising approach to improving the accuracy and stability of epidemic predictions due to their ability to combine predictions from multiple estimators [2,7,8].

Ensembles combine predictions generated by a set of component models [7,9–11]. In some cases, ensembles aggreggate component model predictions by weighting better predictions more highly in the final ensemble prediction [7,8], though other weighting criteria can be applied [8]. The rationale for using ensemble predictions rests in their ability to borrow the strengths and discard the weaknesses of various component models. This feature tends to lead not only to more accurate predictions but to more stable ones that can be applied across a range of scenarios [8]. The CDC's primary in-season ILI forecasts are now based on an ensemble forecast generated by aggregating predictions from a growing library of individual forecasts submitted by research teams around the U.S. [3].

To date, most work has focused on ILI [3,5,6,12,13], with considerably less effort having been exerted so far on predicting influenza-related hospitalization rates [14]. Because the dynamics of flu-related hospitalizations might evolve differently over the course of an influenza season—at the very least, lagging influenza incidence by a week

or two [citation needed]—and because hospitalization rates are an independent signal of the severity of disease caused by circulating flu strains, optimizing ensembles to predict hospitalization rates can provide complementary information to ILI forecasts.

One ensemble machine learning method in particular, dubbed "super learner" [15–17], exhibits a number of desirable properties that suggest it may be a powerful tool for predicting flu hospitalizations. First, its developers have demonstrated that, asymptotically, the super learner is an oracle estimator, performing as well as the best-fitting component model and converging almost as quickly [16] [also will want to read and cite the 2003 paper of van der Laan's]. Second, this oracle property generally translates to finite samples [15–17]. Finally, several packages have been developed to implement the super learner algorithm [18,19], providing researchers easy access to a relatively large library of component models and a means to calculate cross-validated prediction risks quite easily [19].

In this study, we sought to train an ensemble learner on a distribution of simulated influenza hospitalization curves to generate predictions for three national-level seasonal target parameters based on the CDC forecasting competitions [20] to compare the performance of the ensemble learner against the best-performing component model and a naive historical average prediction for each of these targets across the 30 weeks of a typical flu season.

Methods

Empirical data

We downloaded publicly available surveillance data seasonal influenza-related hospitalizations from the CDC's FluView Interactive dashboard [21]. Specifically, we included data from the Emerging Infections Program (EIP) beginning with the 2003–2004 season and ending with the 2018–2019 season, omitting the 2009-2010 pandemic year. The EIP contains data on flu-related hospitalizations in California, Colorado, Connecticut, Georgia, Maryland, Minnesota, New Mexico, New York, Oregon, and Tennessee [21]. Since the 2009-2010 the FluSurv Network included between 3 and 6 states in addition to those included in EIP, depending on the year [21]. We elected to use only the EIP data in order to maintain consistency within the empirical data and to increase the number of flu seasons available to inform both curve simulation and parameters targets.

Typically, the CDC releases data for epiweeks 40–53 and 1–17 (approximately, October through April of the next year) [21,22]. We renumbered the epiweeks 1–30, omitting epiweek 53 as only three seasons had influenza hospitalization data recorded in this week (an artifact of leap years).

Prediction targets

Following from the CDC's Flu Sight challenge [20], we defined three season-level prediction targets: peak rate, peak week, and cumulative hospitalizations. *Peak rate* refers to the highest weekly rate of influenza-related hospitalizations throughout the course of a flu season (per 100,000 population), *peak week* to the week during which this peak rate occurs, and *cumulative hospitalizations* to the cumulative influenza-related hospitalization rate over the 30 weeks of the season (per 100,000 population) [20].

Fifteen empirical observations were available for each prediction target, corresponding to the number of flu seasons contained in the CDC's surveillance data (Table 1).

Table 1. Empirical distributions of peak hospitalization rate, peak week and cumulative hospitalization rate in the United States, 2003–2019.

Season	Peak rate	Peak week	Cumulative rate
2003-04	5.4	12.0	30.6
2004-05	1.3	19.0	13.5
2005-06	1.1	23.0	10.9
2006-07	0.5	20.0	6.2
2007-08	2.3	21.0	18.3
2008-09	0.8	20.5	7.8
2009-10	3.8	3.0	29.3
2010-11	2.1	21.0	20.1
2011-12	1.1	24.0	8.6
2012-13	5.3	14.0	43.0
2013-14	3.9	14.5	35.2
2014-15	9.0	13.0	64.2
2015-16	4.2	23.0	31.5
2016-17	5.2	21.0	62.5
2017-18	10.3	14.0	102.5
2018-19	5.4	24.0	64.9

Source: CDC FluView. Peak rate and cumulative rate expressed per 100,000 population. Pandemic influenza season 2009–2010 omitted.

Hospitalization curve simulation

To simulate a distribution of seasonal influenza hospitalization curves, we adapted an approach by Brooks et al. originally used to predict influenza-like illness [6]. First, we fit a linear trend filter [23,24] to the 15 observed influenza hospitalization curves from the EIP using the `glmgen` package in R [25]. The `glmgen` linear trend filter is a penalized method that fits a piecewise linear function to a time series, testing 50 values of the penalty (λ) [25]. In all cases, we selected the fit that used the 25th penalty value tested (S1 Fig). We then used these fits as templates for the simulated influenza hospitalization curves.

Next, we incorporated the 15 fit objects into a modified version of the curve generation scheme described in Brooks et al. [6]. Our notation borrows and follows closely from theirs.

Briefly, Brooks et al. conceptualize as seasonal influenza curve as some function plus noise [6]. Adapted to the hospitalization case, the hospitalization rate (y_i^s) in season s and week i is given by

$$y_i^s = f^s(i) + \epsilon_i^s, \epsilon \sim N(0, \tau^s),$$

where $f^s(i)$ is a hospitalization rate and ϵ_i^s is normally distributed error term with mean 0 and variance τ^s.

For each empirical season s, we use its linear trend filter fit and average the squared residuals over i to estimate τ^s:

$$(\hat{\tau}^s)^2 = \underset{i}{\text{avg}} \left[y_i^s - \hat{f}^s(i) \right]^2.$$

For each simulated curve, each of the following parameters is sampled randomly and independently:

Table 2. Input parameters to the influenza hospitalization curve generating function.

Parameter	Description
Shape	$f \sim U\{\hat{f} : \text{historical season } s\}$
Noise	$\sigma \sim U\{\hat{\tau}^{s'} : \text{historical season } s'\}$
Pacing	$\nu \sim U[0.75, 1.25]$, stretches the curve around the peak week
Peak height	$\theta \sim U\,[\theta_{min}, \theta_{max}]$
Peak week	$\mu \sim U[\mu_{min}, \mu_{max}]$

The noise parameter for each simulation is drawn separately, and may come from a season different than the season used as the shape.

$$\langle f, \sigma, \nu, \theta, \mu \rangle,$$

where f denotes a randomly selected vector of estimated hospitalization rates based on a linear trend filter fit to empirical season s, σ denotes the squared error of a linear trend filter fit to randomly sampled season s' and averaged across all weeks, ν denotes a random uniform draw from the range [0.75, 1.25], θ denotes a random draw from the vector of peak hospitalization rates based on the 15 trend filter fits, and μ denotes a random draw from the vector of peak weeks based on the 15 trend filter fits (Table 2).

The generating function for hospitalization rate in week i of simulated season sim is therefore given by:

$$f^{sim}(i) = \frac{\theta}{\max_j f(j)} \left[f\left(\frac{i-\mu}{v} + \begin{matrix} \arg\max j \\ f(j) \end{matrix} \right) \right] + \epsilon_i, \epsilon_i \sim N(0, \hat{\tau}^{s'}),$$

where $f(j)$ now denotes the vector of fitted hospitalization rates from the linear trend filter fit (randomly selected shape f) and j the integer week for which we want to retrieve the prediction from this fit (equal to i). $max_j f(j)$ denotes the peak hospitalization rate from the selected shape, whereas $\begin{matrix} \arg\max j \\ f(j) \end{matrix}$ denotes the week in which this peak occurred. The parameters θ, μ, and ν follow from their prior description. Finally, we introduce noise for each simulated weekly hospitalization rate based on the selected estimate of τ^2.

We alter the original curve formula to impose a lower bound of 0 on the hospitalization rate via the following transformation of $\hat{y}_i^s$, denoted below as $\hat{z}_i^s$:

$$\hat{z}_i^s = 0.5\Big(|\hat{y}_i^s| + \hat{y}_i^s \Big).$$

This transformation effectively preserves positive simulated hospitalization rates and sets negative hospitalization rates to 0. Negative hospitalization rates may be generated at the tails of a given season, when hospitalization rates are generally low, because the error is normally distributed in all weeks.

In all, we simulated 15,000 curves in order to generate an average of 1,000 simulated curves with shapes based on each empirical season (S2 Fig). These simulated curves may be thought of as a plausible distribution of hypothetical flu seasons that could be observed in principle but perhaps have not yet been realized [6].

Super learner

The super learner is a loss-based estimation algorithm [15–17]. It takes as inputs training data, a so-called "library" of component models (called *learners*), and a desired loss function specified to optimize model fits against a given prediction target. See Naimi & Balzer for an accessible introduction [27].

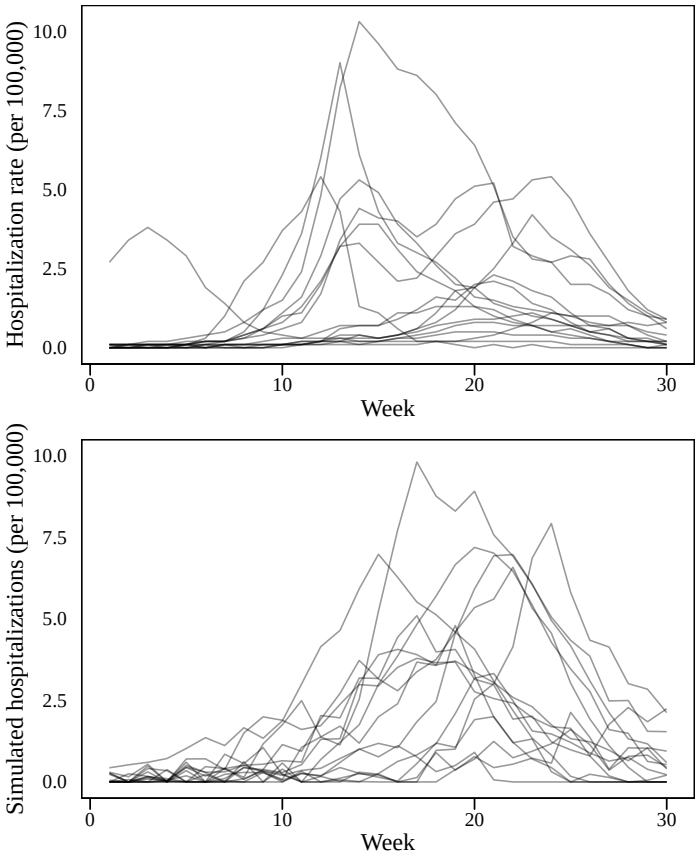


Fig 1. Empirical (top) and 15 randomly selected simulated (bottom) hospitalization curves. Empirical source: CDC, Emerging Infections Program (omitting 2009–2010 pandemic influenza season).

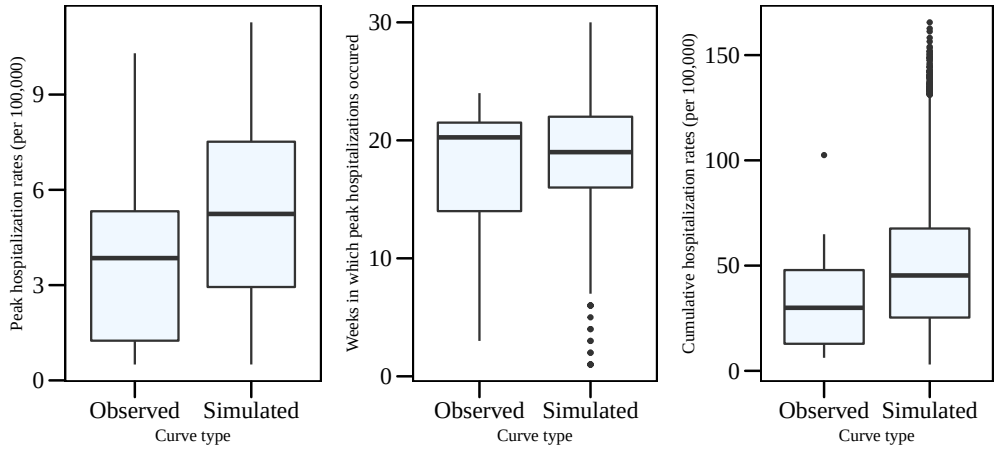


Fig 2. Empirical (N = 15) vs. simulated (N = 15,000) target distributions.

For each prediction target, we specified the same loss function and component model library and trained the super learner on the simulated distribution of hospitalization curves.

We opted to use the L_1 absolute error loss function to target the median of each prediction target distribution [17], meaning the ensemble learner would seek to minimize prediction risk by minimizing the absolute difference between a simulated instance of the prediction target and a given learner's prediction. Finally, we conducted the super learner procedure at each week using the following covariates: hospitalization rate per 100,000 population in week i, cumulative hospitalizations per 100,000 population in through week i, hospitalization rates from all prior weeks, cumulative hospitalizations from all prior weeks, the difference between the hospitalization rate in week i and the hospitalization rate in each prior week up to 5 weeks in the past, the difference between the cumulative hospitalization rate in week i and the cumulative hospitalization rate in each prior week up to 5 weeks in the past, interactions between hospitalization rate in week i and each of the cumulative hospitalization differences, and interactions between cumulative hospitalization rate in week i and each of the hospitalizaion rate differences.

The general procedures is as follows:

1. Specify the library of component learners (i.e., prediction algorithms).
2. Specify the parameter targets and loss function(s).
3. Split the simulated data into 15 folds, each fold containing simulated curves based on the same empirical template season. Splitting the data in this way accounts for the dependence between simulated seasons that rely on the same underlying shape.
4. Train the super learner on the dataset using 15-fold cross-validation, such that each fold is used once as the validation set.
5. Calculate the risk for each candidate model.
6. Generate the ensemble learner as a weighted combination of the component model predictions using a non-negative least squares regression model where model coefficients (i.e., component model weights) are normalized to sum to one and constrained to be greater than or equal to 0 and less than or equal to 1 [27].

Note to co-authors: I plan to write this out more formally as an algorithm.

In all, we trained 6X component models to each task, including variations on tuning and input parameters for the various learners (Table 2).

Target 1: Peak rate

General form:

$$E[I(Y_i = 1)A_i|\bar{A}_{i=d}, \bar{X}_{i=d}) = \beta_0 + \vec{\beta}_a\bar{A}_{i=d} + \vec{\beta}_x\bar{X}_{i=d} + \vec{\beta}_t\bar{T}(i = d)$$

where i indexes the week of the season, c indicates the current week at which we are making a prediction of the target, $Y_i = 1$ identifies the week as the week in which the peak hospitalization rate occurred, $\bar{A}_{i=d}$ indicates the history of hospitalization rates through week $i = d$, $\bar{X}_{i=d}$ indicates the history of cumulative hospitalization rates through week $i = d$, and $\bar{T}(i)$ indicates derived variables to capture time trends and/or proposed interactions between variables. Each beta coefficient topped with an arrow and including a lowercase subscript indicates the vector of coefficients implied by the corresponding variable history.

Table 3. Component models and tuning parameters.

Model	Tuning parameters	R package
Linear regression	Variables screened for inclusion first using a cross-validated lasso procedure to omit variables with zero-valued coefficients.	base
Random forest (regression trees)	number of trees = [50, 100, 200, 500], terminal node sizes = [3, 5, 10]	randomForest
Generalized additive model	gamma penalty = [1, 2, 3, 4, 5]	gam
Support vector regression	kernel = [radial, polynomial], degree = [1, 2, 3] applied only to polynomial	svm
Loss-based regression	penalty = [lasso, ridge]	glmnet
Elastic net	alpha penalty = [0.25, 0.5, 0.75]	glmnet
Neural network	number of nodes in hidden layer = [5, 10, 25, 50, 75, 100], decay = [0, 0.005, 0.1, 0.2, 0.4]	nnet
Loess	span = [0.25, 0.5, 0.75, 1]	base
Polynomial multivariate adaptive regression spline	gcv penalty = [2, 4, 6, 8, 10]	polspline

For the random forest and neural network learners, all combinations of the tuning parameters shown were proposed.

Target 2: Peak week

General form of the candidate models:

$$E(I(Y_i = 1)i|\bar{A}_{i=d}, \bar{X1}_{i=d}) = \beta_0 + \vec{\beta}_a\bar{A}_{i=d} + \vec{\beta}_x\bar{X}_{i=d} + \vec{\beta}_t\bar{T}(i)$$

where i indexes the week of the season, $I(Y_i = 1)$ is the indicator function that identifies a week as being the season's peak, $\bar{A}_{i=d}$ indicates the history of hospitalization rates through week $i = d$, $\bar{X}_{i=d}$ indicates the history of cumulative hospitalization rates through week $i = d$, and $\bar{T}(i)$ indicates derived variables to capture time trends and/or proposed interactions between variables. Each beta coefficient topped with an arrow and including a lowercase subscript indicates the vector of coefficients implied by the corresponding variable history.

Target 3: Cumulative hospitalizations

General form:

$$E[X_{i=30}|\bar{A}_{i=d}, \bar{X}_{i=d}) = \beta_0 + \vec{\beta}_a\bar{A}_{i=d} + \vec{\beta}_x\bar{X}_{i=d} + \vec{\beta}_t\bar{T}(i = d)$$

where i indexes a week of the season, c indicates the current week at which we are making a prediction of the target, $\bar{A}_{i=d}$ indicates the history of hospitalization rates through week d, $\bar{X}_{i=d}$ indicates the history of cumulative hospitalization rates through week d, and $\bar{T}(i)$ indicates derived variables to capture time trends and/or proposed interactions between variables. Each beta coefficient topped with an arrow and including a lowercase subscript indicates the vector of coefficients implied by the corresponding variable history.

Component Models

The library of component models (Table 3) will be shared across all three prediction tasks.

Note to co-authors: Subject to further updates based on candidate availability in package tlverse/sl3. Dr. Steingrimsson, the GAMs mentioned in Table 3 are all returning errors to me, saying there are fewer model parameters than degrees of

freedom. Also, I had planned at one point to include some GBMs, but I didn't feel comfortable doing so because I just didn't feel like I understood them well enough.

Results

Peak rate

The ensemble superlearner consistently performed worse—i.e., had a higher mean risk—than both the discrete super learner and the mean prediction.

Note to co-authors: See Table 4 for some preliminary results. Please note that I had to do a little additional debugging, and so these risks were calculated on the squared error loss instead of the absolute error loss. I'm not sure whether the results will change much, though anecdotally, the mean risks did seem to be different when I did some short test runs with the absolute error loss on an abbreviated stack of learners. You'll notice we see something a little strange in Table 4, too. While the super learner is supposed to have desirable oracle properties (the ensemble learner should asymptotically be very close to the discrete super learner), the super learner seems to be affected negatively by poorly fitting candidate models, so much so that it actually performs *worse* than taking a simple mean as the prediction. The discrete super learner, however, which is the best-fitting indiviudal component model in a given week exhibits more or less what I expected: didn't give us much improvement over a simple mean early in the season but performed better (generally) as the season progressed. I am not convinced the strange results with the ensemble aren't due to some poor choice of tuning parameters on my part, since many of these component models are ones I haven't worked with before. Dr. Steingrimsson, it would be great to run some of the models by you if you have time.

Note to co-authors: Table 5 may not be very informative. Usually, these numbers are shown with a risk plot, but because of some very poorly fit models in the component set, the plots weren't helpful.

Peak week

Cumulative hospitalizations

Discussion

Limitations

- λ values used in the trend filter fits were arbitrary.
- We did not include any mechanistic models in the component learner library (for instance, compartmental or agent-based models). Future work could incorporate such models, though most agent-based models may be too computationally intensive to use efficiently as component learners.

Software and code

All code is provided at ... [set up persistent DOI at Zenodo or Open Science Framework and link to Github repo for FluHospPrediction package]

Declarations

Acknowledgment

We thank Ashley Naimi, Laura Balzer, and Nicholas Reich for their comments on the study aims and the super learner approach.

Table 4. Cross-validated risks for the ensemble super learner, discrete super learner, and mean predictions of peak hospitalization rate, by week of influenza season. Estimates presented as mean risk (standard error).

Week	SuperLearner	Discrete.SL	Mean
1	8.62 (0.09)	6.26 (0.10)	8.29 (0.06)
2	11.53 (0.21)	4.60 (0.17)	8.29 (0.06)
3	12.48 (0.15)	6.40 (0.17)	8.29 (0.06)
4	9.50 (0.15)	4.19 (0.09)	8.29 (0.06)
5	10.42 (0.11)	5.59 (0.12)	8.29 (0.06)
6	9.53 (0.15)	4.53 (0.13)	8.29 (0.06)
7	10.59 (0.17)	5.79 (0.14)	8.29 (0.06)
8	13.02 (0.14)	3.52 (0.12)	8.29 (0.06)
9	12.47 (0.16)	6.70 (0.09)	8.29 (0.06)
10	11.40 (0.14)	7.04 (0.17)	8.29 (0.06)
11	10.10 (0.11)	4.31 (0.11)	8.29 (0.06)
12	11.24 (0.12)	5.57 (0.18)	8.29 (0.06)
13	10.91 (0.14)	6.00 (0.12)	8.29 (0.06)
14	10.40 (0.09)	5.53 (0.19)	8.29 (0.06)
15	9.79 (0.11)	6.10 (0.16)	8.29 (0.06)
16	10.36 (0.11)	5.36 (0.10)	8.29 (0.06)
17	11.21 (0.13)	6.38 (0.06)	8.29 (0.06)
18	13.31 (0.10)	5.23 (0.11)	8.29 (0.06)
19	12.74 (0.12)	3.23 (0.20)	8.29 (0.06)
20	10.46 (0.12)	2.32 (0.16)	8.29 (0.06)
21	11.32 (0.15)	1.60 (0.18)	8.29 (0.06)
22	11.97 (0.14)	0.97 (0.19)	8.29 (0.06)
23	9.69 (0.10)	0.45 (0.14)	8.29 (0.06)
24	13.06 (0.14)	0.70 (0.15)	8.29 (0.06)
25	11.26 (0.09)	0.66 (0.15)	8.29 (0.06)
26	13.23 (0.16)	0.72 (0.14)	8.29 (0.06)
27	12.36 (0.12)	0.49 (0.16)	8.29 (0.06)
28	11.77 (0.18)	0.48 (0.15)	8.29 (0.06)
29	9.18 (0.13)	0.53 (0.16)	8.29 (0.06)
30	9.19 (0.13)	0.64 (0.15)	8.29 (0.06)

The ensemble super learner (SuperLearner) is the prediction generated as a weighted combination of component model predictions. The discrete super learner (Discrete.SL) is the best-performing component model. The mean (Mean) is the average peak hospitalization rate across the simulated hospitalization curves.

Funding statement

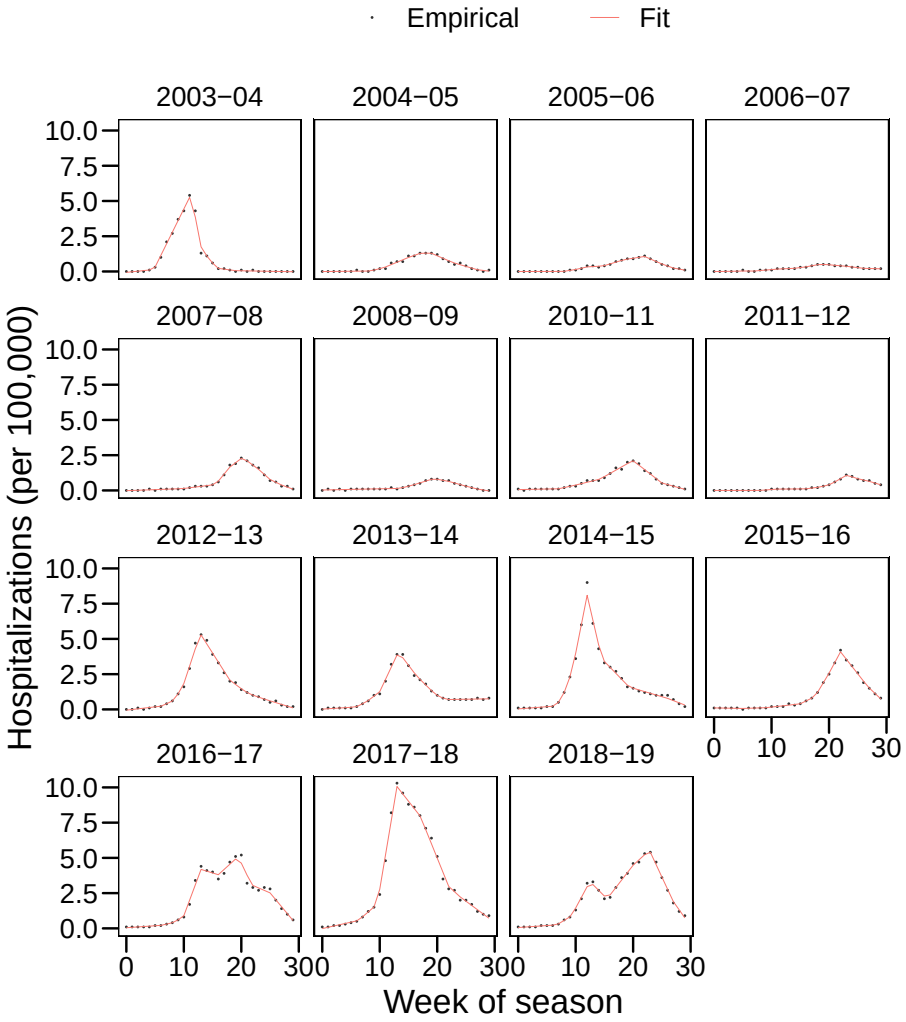
This work was funded by an unrestricted grant from Sanofi (PI: Andrew Zullo). The funders did not assist in the statistical analysis nor did they have a say in the final decision to submit the manuscript for publication.

Competing interests

[solicit competing interests from co-authors]

Supporting information

S1 Fig. Linear trend filter fits to observed influenza hospitalization curves.



S2 Table. Number of simulated curves based on each observed flu season (Emerging Infections Program).

Template season	N
2003-04	979
2004-05	1041
2005-06	965
2006-07	1059
2007-08	1044
2008-09	965
2010-11	965
2011-12	1000
2012-13	1000
2013-14	1002
2014-15	934
2015-16	1027
2016-17	1027
2017-18	985
2018-19	1007

S3 Table. Weekly cross-validated risks across the component learners used to predict peak hospitalization rate.

Week	Mean	SD	Median	Minimum	Maximum
1	18.56	54.06	10.65	6.26	437.84
2	15.85	38.85	8.88	4.60	311.26
3	4438.82	25658.32	9.55	6.40	188085.83
4	117.46	851.75	8.29	4.19	6770.43
5	23.31	52.57	9.93	5.59	357.02
6	14.51	27.31	10.00	4.53	199.83
7	10.55	4.37	9.80	5.79	35.99
8	2932.67	23043.07	9.99	3.52	182924.44
9	142.76	1032.68	9.57	6.70	8208.75
10	212.01	1592.42	10.02	7.04	12650.76
11	3769.90	29743.95	10.12	4.31	236106.63
12	36199.55	286962.66	10.56	5.57	2277740.22
13	29551.12	234391.41	8.71	6.00	1860444.63
14	14816.54	117441.71	8.98	5.53	932184.77
15	26447.32	209729.12	9.74	6.10	1664697.00
16	4522.92	35454.51	9.65	5.36	281453.44
17	18454.59	146215.35	10.25	6.38	1160580.72
18	18047.63	141087.49	8.90	5.23	1120048.71
19	314.16	2374.51	8.43	3.23	18857.79
20	370.00	2366.38	8.29	2.32	18454.21
21	17.89	57.07	8.29	1.60	378.23
22	14.81	45.73	8.29	0.97	337.61
23	9.15	10.54	8.52	0.45	64.37
24	104.03	694.42	8.29	0.70	5493.65
25	9.85	17.16	8.31	0.66	120.03
26	14.81	48.09	8.29	0.72	371.27
27	11.45	34.52	8.29	0.49	278.48
28	9.21	15.74	8.29	0.48	119.38
29	413.64	3229.58	8.29	0.53	25640.73
30	57.37	400.08	8.29	0.64	3182.27

References

1. Centers for Disease Control and Prevention. Burden of influenza. `https://www.cdc.gov/flu/about/burden/index.html`; 2020.

2. Lutz CS, Huynh MP, Schroeder M, Anyatonwu S, Dahlgren FS, Danyluk G, et al. Applying infectious disease forecasting to public health: A path forward using influenza forecasting examples. BMC Public Health. 2019;19: 1659. doi:10.1186/s12889-019-7966-8

3. Reich NG, Brooks LC, Fox SJ, Kandula S, McGowan CJ, Moore E, et al. A collaborative multiyear, multimodel assessment of seasonal influenza forecasting in the united states. Proc Natl Acad Sci U S A. 2019;116: 3146–3154. doi:10.1073/pnas.1812594116

4. Chretien J-P, George D, Shaman J, Chitale RA, McKenzie FE. Influenza forecasting in human populations: A scoping review. PLoS One. 2014;9: e94130. doi:10.1371/journal.pone.0094130

5. Biggerstaff M, Kniss K, Jernigan DB, Brammer L, Bresee J, Garg S, et al. Systematic assessment of multiple routine and near Real-Time indicators to classify the severity of influenza seasons and pandemics in the united states, 2003-2004 through 2015-2016. Am J Epidemiol. 2018;187: 1040–1050. doi:10.1093/aje/kwx334

6. Brooks LC, Farrow DC, Hyun S, Tibshirani RJ, Rosenfeld R. Flexible modeling of epidemics with an empirical bayes framework. PLoS Comput Biol. 2015;11: e1004382. doi:10.1371/journal.pcbi.1004382

7. Reich NG, McGowan CJ, Yamana TK, Tushar A, Ray EL, Osthus D, et al. Accuracy of real-time multi-model ensemble forecasts for seasonal influenza in the U.S. PLoS Comput Biol. 2019;15: e1007486. doi:10.1371/journal.pcbi.1007486

8. Ray EL, Reich NG. Prediction of infectious disease epidemics via weighted density ensembles. PLoS Comput Biol. 2018;14: e1005910. doi:10.1371/journal.pcbi.1005910

9. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning: Data mining, inference, and prediction. Springer, New York, NY; 2009. doi:10.1007/978-0-387-84858-7

10. Wolpert DH. Stacked generalization. Neural Netw. 1992;5: 241–259. doi:10.1016/S0893-6080(05)80023-1

11. Breiman L. Stacked regressions. Mach Learn. 1996;24: 49–64. doi:10.1007/BF00117832

12. McGowan CJ, Biggerstaff M, Johansson M, Apfeldorf KM, Ben-Nun M, Brooks L, et al. Collaborative efforts to forecast seasonal influenza in the united states, 2015-2016. Sci Rep. nature.com; 2019;9: 683. doi:10.1038/s41598-018-36361-9

13. Kandula S, Yamana T, Pei S, Yang W, Morita H, Shaman J. Evaluation of mechanistic and statistical methods in forecasting influenza-like illness. J R Soc Interface. 2018;15. doi:10.1098/rsif.2018.0174

14. Kandula S, Pei S, Shaman J. Improved forecasts of influenza-associated hospitalization rates with google search trends. J R Soc Interface. 2019;16: 20190080. doi:10.1098/rsif.2019.0080

15. Laan MJ van der, Polley EC, Hubbard AE. Super learner. Stat Appl Genet Mol Biol. De Gruyter; 2007;6: Article25. doi:10.2202/1544-6115.1309

16. Polley EC, Laan MJ van der. Super learner in prediction. University of California, Berkeley; 2010.

17. Polley EC, Rose S, Laan MJ van der. Super learning. In: Laan MJ van der, Rose S, editors. Targeted learning: Causal inference for observational and experimental data. New York, NY: Springer New York; 2011. pp. 43–66. doi:10.1007/978-1-4419-9782-1_3

18. Polley E, LeDell E, Kennedy C, van der Laan M. SuperLearner: Super learner prediction [Internet]. 2019. Available: `https://CRAN.R-project.org/package=SuperLearner`

19. Coyle JR, Hejazi NS, Malenica I, Sofrygin O. Sl3: Pipelines for machine learning and super learning. 2020. doi:10.5281/zenodo.1342293

20. Centers for Disease Control and Prevention. Epidemic prediction initiative. `https://predict.cdc.gov/post/59973fe26f7559750d84a843`;

21. Centers for Disease Control and Prevention. Flu view phase 3 quick reference guide [Internet]. Available: `https://gis.cdc.gov/GRASP/Fluview/FluHospRates.html`

22. Centers for Disease Control and Prevention. MMWR week overview. `https://wwwn.cdc.gov/nndss/document/MMWR_Week_overview.pdf`;

23. Kim S, Koh K, Boyd S, Gorinevsky D. $\ell 1$ trend filtering. SIAM Rev. Society for Industrial; Applied Mathematics; 2009;51: 339–360. doi:10.1137/070690274

24. Tibshirani RJ. Adaptive piecewise polynomial estimation via trend filtering. Ann Stat. Institute of Mathematical Statistics; 2014;42: 285–323. doi:10.1214/13-AOS1189

25. Arnold T, Sadhanala V, Tibshirani R. Glmgen. 2015.

26. R Core Team. R: A language and environment for statistical computing [Internet]. Vienna, Austria: R Foundation for Statistical Computing; 2020. Available: https://www.R-project.org/

27. Naimi AI, Balzer LB. Stacked generalization: An introduction to super learning. Eur J Epidemiol. 2018;33: 459–464. doi:10.1007/s10654-018-0390-z