

FORECASTING COVID-19 COUNTS AT A SINGLE HOSPITAL: A HIERARCHICAL BAYESIAN APPROACH

Alexandra Hope Lee
Tufts University

Panagiotis Lymperopoulos
Tufts University

Joshua T. Cohen
Tufts Medical Center

John B. Wong
Tufts Medical Center

Michael C. Hughes
Dept. of Computer Science, Tufts University

ABSTRACT

We consider the problem of forecasting the daily number of hospitalized COVID-19 patients at a single hospital site, in order to help administrators with logistics and planning. We develop several candidate hierarchical Bayesian models which directly capture the count nature of data via a generalized Poisson likelihood, model time-series dependencies via autoregressive and Gaussian process latent processes, and can share statistical strength across related sites. We demonstrate our approach on public datasets for 8 hospitals in Massachusetts, U.S.A. and 10 hospitals in the United Kingdom. Further prospective evaluation compares our approach favorably to baselines currently used by stakeholders at 3 related hospitals to forecast 2-week-ahead demand by rescaling state-level forecasts.

1 INTRODUCTION

The COVID-19 pandemic has created unprecedented demand for limited hospital resources across the globe. Emergency resource allocation decisions made by hospital administrators (such as planning additional personnel or provisioning beds and equipment) are crucial for achieving successful patient outcomes and avoiding overwhelmed capacity. However, at present hospitals often lack the ability to forecast what will be needed at their site in coming weeks. This may be especially true in under-resourced hospitals, due to constraints on funding, staff time and expertise, and other issues. In response to this pressing need, in this study our goal is to develop statistical machine learning approaches to forecast hospital utilization for specific hospitals. By focusing on the short-term future (1-3 weeks ahead) at specific sites, we hope predictions are directly *actionable* so hospital administrators can respond to forecasted demand.

While many efforts to forecast the spread of COVID-19 and its impact on hospitals have been publicized (Murray et al., 2020; Jewell et al., 2020; Reiner et al., 2021), they are not usable for off-the-shelf predictions for a specific hospital because they focus on whole countries, states, or regions rather than a specific site. Even if this regional forecasting were reliable, it can be of limited relevance to a particular hospital (Wong, 2020), at which new patient arrivals depend on localized conditions (e.g., case incidence at local “hot spots”). Notable efforts for hospital-level or patient-level modeling exist (Epstein & Dexter, 2020; Roimi et al., 2021), but require much more detailed data than our approach (e.g. length-of-stay for all patients at a site or other patient-level covariates).

Contributions. Our study develops and validates latent variable models to predict *distributions* over census counts at a hospital site for each future day of interest given a univariate time series of past counts. Our probabilistic methods can work even if the past census data is not fully observed (as might arise in sites relying on noisy or error-prone record-keeping processes). As a technical contribution, we show that generalized Poisson likelihoods are a better alternative to the more popular standard Poisson likelihoods for modeling hospitalization census counts. We further show that when modeling multiple hospital sites in the same region, we can use hierarchical modeling to *share* statistical strength and improve forecasts. We have released open-source Python code¹ to allow others to reproduce our analysis, making site-specific demand prediction at sites around the globe possible.

¹<https://github.com/tufts-ml/single-hospital-count-forecasting/>

2 METHODS

Suppose across T days we observe a univariate time series $y_{1:T} = [y_1, y_2, \dots, y_T]$, where $y_t \in \{0, 1, 2, \dots\}$ indicates the patient occupancy count for a specific hospital on day t . Our goal is to develop *forecasts* for the next F days (typically a few weeks ahead), given all previous observations, using a conditional probabilistic model $p(y_{(T+1):(T+F)} \mid y_{1:T})$. We consider a flexible family of latent variable models, where each timestep has a latent real value $f_t \in \mathbb{R}$. We capture dependency across time in the latent series $f_{1:T}$, and model each count y_t as conditionally independent given f_t :

$$p(f_{1:T}, y_{1:T}) = p_\alpha(f_{1:T}) \cdot \prod_{t=1}^T p_\lambda(y_t \mid f_t), \quad (1)$$

with latent-generating parameters α and count-generating parameters λ . We'll consider two options for p_α motivated by different dependency assumptions, as well as two possible likelihoods p_λ .

Likelihoods for count data. Given f_t , to generate the observed count y_t on day t , we use either the standard Poisson or the generalized Poisson distribution. In both cases, we set the mean parameter by transforming the latent f_t to a positive value via the exponential:

$$p_\lambda(y_t \mid f_t) = \text{Poisson}(\exp(f_t)), \text{ or } p_\lambda(y_t \mid f_t) = \text{GenPoisson}(\exp(f_t), \lambda) \quad (2)$$

The standard Poisson has no ability to control variance separately from the mean. The generalized Poisson (Consul & Famoye, 1992) has dispersion parameter: $\lambda \in [-1, +1]$, reducing to the standard Poisson when $\lambda = 0$. We assume a priori symmetric chances of both under- and over-dispersion, so we set the prior on λ be a normal distribution with mean 0 and standard deviation 0.3, truncated to $[-1, +1]$. Typically, on our hospital data we find that our posteriors favor under-dispersion.

Generalized Autoregressive (GAR) model. Our generalized autoregressive model is an instance of the generative model in Eq. (1) with an order- W autoregressive process to generate $f_{1:T}$:

$$p_\alpha(f_{1:T}) = \prod_{t=1}^T p_\alpha(f_t \mid f_{(t-W):(t-1)}) = \prod_{t=1}^T \mathcal{N}(f_t \mid \beta_0 + \sum_{\tau=1}^W \beta_\tau f_{t-\tau}, \sigma^2) \quad (3)$$

We place vague unimodal priors over the parameters $\alpha = \{\beta, \sigma\}$ (see App. E). Window size W is a hyperparameter selected on validation data. Our GAR model is limited to linear dependencies within the latent sequence $f_{1:T}$, but it is conceptually simple and fast to fit and evaluate.

Generalized Gaussian Process (GGP) model. We next consider a model where latents are drawn from a Gaussian process (GP), which can be seen as a simplified GP state-space model (Damianou et al., 2011; Frigola et al., 2014). The goal here is a flexible, data-driven model for non-linear trends in $f_{1:T}$ without an explosion of parameters to learn. Our model is again an instance of Eq. (1), with a GP for the prior over the latents: $f_{1:T} \sim \text{GP}(m_\alpha(t), k_\alpha(t, t'))$. We assume a constant mean $m_\alpha(t) = c$ and squared exponential covariance kernel $k_\alpha(t, t') = a^2 \exp\left(-\frac{(t-t')^2}{2\ell^2}\right)$. The parameters $\alpha = \{c, a, \ell\}$, with $a > 0$ and $\ell > 0$, are given vague unimodal priors (see App. F).

Multi-site hierarchical model. Now, consider predicting future census counts at H different hospital sites simultaneously, given the same T days of observations from each site. If all H sites share common trends (i.e. cases rising in all because they draw from similar populations), we might improve forecasts by modeling sites in the hierarchical Bayesian fashion (Gelman, 2006).

Our multi-site generative model ties the latent-sequence-generating parameters α across sites, but allows likelihood parameters λ^h to be specific to each site (indexed by h). The model factorizes as:

$$p(\{y_{1:T}^h, f_{1:T}^h\}_{h=1}^H) = \prod_{h=1}^H p_\alpha(f_{1:T}^h) p_{\lambda^h}(y_{1:T}^h \mid f_{1:T}^h). \quad (4)$$

For simplicity, all multi-site experiments use the GAR model described above, with its latent-generating parameters $\alpha = \{\beta, \sigma\}$ shared among all sites, again with vague priors (App. H).

Posterior estimation and forecasting. For all methods, we use a No-U-Turn sampler (Hoffman & Gelman, 2014) to perform Markov chain Monte Carlo approximate sampling from the posterior, as implemented using the PyMC3 toolbox (Salvatier et al., 2016). This lets us sample from the posterior over parameters and latent values: $p(\alpha, \lambda, f_{1:T} \mid y_{1:T})$ for single site models and $p(\alpha, \{\lambda^h, f_{1:T}^h\}_{h=1}^H \mid \{y_{1:T}^h\}_{h=1}^H)$ for multi-site models, gathering S posterior samples (indexed by s). We collect thousands of samples from multiple chains to avoid poor convergence.

Then, we can sample S forecasts for site h by conditioning on each posterior sample's parameters $\alpha^s, \lambda^{s,h}$ and latents $f_{1:T}^{s,h}$ when simulating the next F days (indexed by τ) from the model:

$$f_{T+\tau}^{s,h} \sim p_{\alpha^s}(f_{T+\tau}^h \mid f_{1:T+\tau-1}^{s,h}), \quad y_{T+\tau}^{s,h} \sim p_{\lambda^{s,h}}(y_{T+\tau}^h \mid f_{T+\tau}^{s,h}), \quad \tau \in 1, 2, \dots, F. \quad (5)$$

Hospital Site (MA, USA)	Single-Site GGP	Single-Site GAR	Multi-Site GAR
Beth Israel Deaconess	-3.153 ± 0.005	-3.121 ± 0.017	-3.010 ± 0.011 *
Boston Medical Center	-3.376 ± 0.002	-3.454 ± 0.025	-3.296 ± 0.014 *
Brigham & Women’s Faulkner	-2.719 ± 0.002	-2.684 ± 0.010	-2.601 ± 0.008 *
Brigham & Women’s Hospital	-3.118 ± 0.001	-2.978 ± 0.009*	-3.044 ± 0.016
Carney Hospital	-3.072 ± 0.008	-2.721 ± 0.105	-2.564 ± 0.090 *
Massachusetts General	-3.777 ± 0.020	-3.468 ± 0.018 *	-3.450 ± 0.017 *
St. Elizabeth’s	-1.980 ± 0.007	-1.312 ± 0.120 *	-1.598 ± 0.007
Tufts Medical Center	-2.937 ± 0.005	-2.822 ± 0.010	-2.728 ± 0.007 *
Hospital Site (UK)	Single-Site GGP	Single-Site GAR	Multi-Site GAR
Barts Health	-4.730 ± 0.058 *	-6.727 ± 0.221	-4.842 ± 0.078
Chelsea & Westminster	-2.937 ± 0.009 *	-3.141 ± 0.030	-5.929 ± 0.086
Imperial College Healthcare	-4.017 ± 0.019	-3.063 ± 0.178 *	-3.976 ± 0.110
King’s College Hospital	-3.012 ± 0.003	-2.749 ± 0.020	-2.695 ± 0.005 *
London North West University	-2.829 ± 0.008	-2.488 ± 0.014 *	-2.572 ± 0.009
Manchester University	-12.335 ± 0.647	-12.735 ± 0.937	-6.152 ± 0.167 *
North Middlesex University	-1.593 ± 0.008 *	-1.677 ± 0.022	-1.703 ± 0.008
Nottingham University	-3.443 ± 0.018	-3.324 ± 0.066	-2.950 ± 0.032 *
University Hospitals Birmingham	-4.385 ± 0.005 *	-4.571 ± 0.019	-4.583 ± 0.031
Univ. Hospitals of North Midlands	-3.188 ± 0.002	-2.953 ± 0.033	-2.829 ± 0.016 *

Table 1: **Heldout likelihood for retrospective evaluation across 8 sites in MA (top) and 10 sites in UK (bottom).** We report test-set log likelihood (normalized by total test days). The entry marked (*) in each row indicates the best result. For all methods, we run MCMC given all observations from the combined training-plus-validation dataset, and report likelihoods on the test set. For single-site GGP and single-site GAR, we select hyperparameters values via grid search on the validation set. For multi-site GAR, we set window size to $W = 1$ to keep runtime affordable. Tab. B.1 gives expanded results showing consistency across multiple MCMC chains.

After drawing S forecast samples, we can compute summary statistics of these samples, such as mean or median values as well as lower and upper percentiles.

3 EVALUATION

We applied our models to the task of forecasting site-specific census counts of patients with COVID-19. First, we perform a retrospective evaluation of our proposed models on two public datasets from April to July 2020 (described in App. A). These initial experiments consider data from 8 hospitals in Suffolk County, MA, U.S.A, as well as 10 hospital sites in the UK. Next, we applied the best performing models in a *prospective* validation on data from January to February 2021, comparing our methods to those currently used by stakeholders at a major hospital system in Massachusetts.

Comparison of likelihoods: Generalized vs. Standard Poisson. Supplementary Fig. B.1 shows that our proposed generalized Poisson likelihood delivers better heldout likelihoods on the MA data than the standard Poisson. We use the generalized Poisson throughout the remaining experiments.

Comparison of models: Single-Site GGP vs Single-Site GAR vs Multi-Site GAR. Table 1 provides quantitative heldout likelihood comparisons of our three candidate methods (single-site GGP, single-site GAR, and multi-site GAR), across the 8 sites in MA and 10 sites in the UK. We find that the multi-site GAR model is preferred for 6 out of 8 sites in the MA dataset, perhaps because all sites come from the same county and thus sharing information across sites works well. The UK hospital sites are much more spread out geographically, so we see less conclusive results (multi-site GAR competes well in a plurality of 4 out of 10 sites). Figure B.2 illustrates forecasts for 3 representative sites from MA, showing that the GAR models extrapolate trends better than the GGP.

Prospective validation: Methods. We performed 2-week-ahead forecasts starting on February 4, 2021 at 3 sites in Massachusetts - Melrose-Wakefield Hospital, Lowell General Hospital, and Tufts Medical Center - operated by a common healthcare system. We compare our GAR models against two baselines *currently in use* to help with 2-week-ahead planning at these sites. The first baseline is a **rescaled state-level linear regression**, where state-level census counts are predicted with standard frequentist linear regression (unknown slope, unknown intercept) from the day indicator t . We use public data released by the state and fit to the previous 28 days of data. The second baseline comes from **rescaled state-level IHME forecasts**, which uses the daily mean and 95% CI of hospitalization forecasts for the state of MA made public by the Institute for Health Metrics and Evaluation (IHME). Both baselines involve rescaling state-wide forecasts to site-specific levels. We first fit a linear

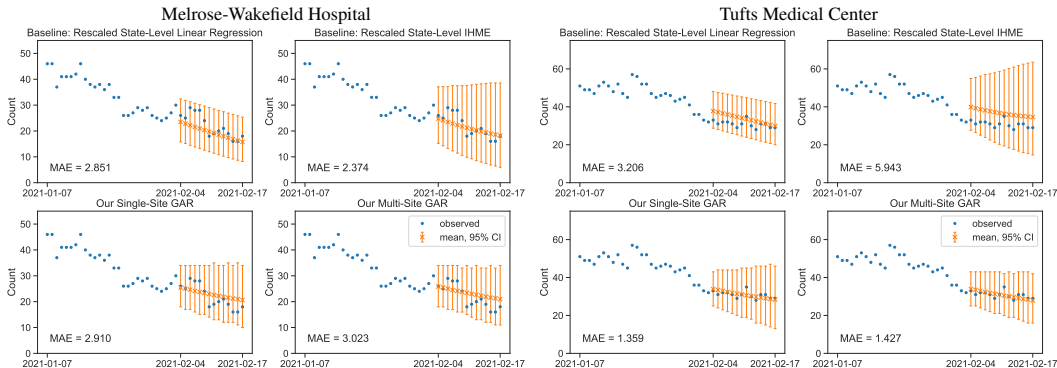


Figure 1: **Prospective evaluation of proposed models against baselines in use at two sites of a major hospital system in MA.** All proposed GAR models deliver competitive mean absolute error and *plausible* interval estimates where uncertainty increases slightly with time. See expanded results for all sites in Fig. C.1.

regression to predict the *fraction* s_t^h of the total state-level volume at each site h on each future day t , based on the previous 28 days of observed fractional volume. Then for each day, we multiply the predicted fractional volume at each site of interest by the predicted state-level count. We further estimate 95% CIs of this fraction and use these to estimate site-level 95% CIs.

Prospective validation: Results. Fig. 1 shows qualitative performance and mean absolute error (MAE) metrics on two of the sites in our prospective evaluation (see all 3 sites in Fig. C.1). All methods capture the essentially linear trend of the test period in all 3 sites. In most cases, the rescaled IHME baseline appears overly uncertain (intervals are too wide), so rescaled linear regression seems to be the better baseline. At two sites, there seems to be little difference (MAE difference within 1.0) between our GAR methods and rescaled linear regression. However, at one site (Tufts Medical) our methods appear noticeably better than the linear regression baseline: MAE improvement in these counts is larger than 1.75 patients per day. Furthermore, at all sites our methods produce the most reasonable uncertainty intervals, which sensibly *grow slightly* further into the future.

4 DISCUSSION

Advantages. Our approach is *simple yet effective*: we find that order-1 autoregressive latent variable models deliver reasonable predictions when we share latent dynamics parameters α across sites and use generalized Poisson likelihoods with flexible dispersion. Our approach may be *portable* to health systems around the world, as it relies on easy-to-collect count data and does not require much outside of a site leader’s control (such as region-level data or expensive computing resources). Our approach is *robust* to realistic scenarios where some counts may be missing or corrupted: our fully-probabilistic model allows us to properly calibrate our uncertainty in these cases. Our approach is *extensible*, relying upon widely-used probabilistic programming toolkits (Salvatier et al., 2016).

Limitations. The primary limitation of this work is that that our forecasts are purely based on statistical patterns in past census counts within the site(s) of interest. Our generalized autoregressive approach may poorly predict in cases where underlying dynamics change (e.g. if the testing period sees higher vaccination rates) or in cases where the local trend is far from linear (hence our initial interest in GPs; we plan to do follow-up work to explore GPs with appropriately learned non-linear mean functions). Additional data, especially leading indicators such as surveillance tests in the region or levels of community mobility and interaction, could improve forecasts especially when there is a *regime change* in disease dynamics (e.g., from the summer lull to the late fall 2020 surge). Other helpful data might include community demographics and density as well as referral patterns (e.g., from nursing homes). We assume that hospitals in the same geographic region will have similar trends, which may miss how nearby hospitals serve different populations and thus see diverse trends. We do not account for how hospitals may *interact* (e.g., transferring patients) to balance capacity.

While there is much more to be done, we hope this study raises interest in the single-site forecasting problem and offers a step forward for improving decision-making for local leaders.

REFERENCES

- Antoni B. Chan and Daxiang Dong. Generalized Gaussian process models. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pp. 2681–2688, Colorado Springs, CO, USA, 2011. IEEE.
- P.C. Consul and Felix Famoye. Generalized poisson regression model. *Communications in Statistics - Theory and Methods*, 21(1):89–109, 1992.
- Andreas Damianou, Michalis K. Titsias, and Neil D Lawrence. Variational Gaussian Process Dynamical Systems. In *Advances in Neural Information Processing Systems*, 2011.
- Richard H Epstein and Franklin Dexter. A Predictive Model for Patient Census and Ventilator Requirements at Individual Hospitals During the Coronavirus Disease 2019 (COVID-19) Pandemic: A Preliminary Technical Report. *Cureus*, 12(6), 2020.
- Felix Famoye. Generalized Poisson random variate generation. *American Journal of Mathematical and Management Sciences*, 17(3-4):219–237, 1997.
- Roger Frigola, Yutian Chen, and Carl Edward Rasmussen. Variational Gaussian Process State-Space Models. In *Advances in Neural Information Processing Systems*, pp. 9, 2014.
- Andrew Gelman. Multilevel (Hierarchical) Modeling: What It Can and Cannot Do. *Technometrics*, 48(3):432–435, 2006.
- Matthew D Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, pp. 31, 2014. <http://jmlr.org/papers/volume15/hoffman14a/hoffman14a.pdf>.
- Nicholas P. Jewell, Joseph A. Lewnard, and Britta L. Jewell. Predictive Mathematical Models of the COVID-19 Pandemic: Underlying Principles and Value of Projections. *JAMA*, 323(19):1893–1894, 2020.
- Neil D Lawrence. Gaussian Process Latent Variable Models for Visualisation of High Dimensional Data. In *Advances in Neural Information Processing Systems*, 2003.
- Christopher JL Murray, others, and the IHME COVID-19 health service utilization forecasting team. Forecasting the impact of the first wave of the COVID-19 pandemic on hospital demand and deaths for the USA and European Economic Area countries. *medRxiv*, 2020. <https://www.medrxiv.org/content/10.1101/2020.04.21.20074732v1>.
- Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.
- Robert C. Reiner, Ryan M. Barber, James K. Collins, Peng Zheng, Christopher Adolph, James Albright, Catherine M. Antony, Aleksandr Y. Aravkin, Steven D. Bachmeier, et al. Modeling COVID-19 scenarios for the United States. *Nature Medicine*, 27(1):94–105, 2021.
- Michael Roimi, Rom Gutman, Jonathan Somer, Asaf Ben Arie, Ido Calman, Yaron Bar-Lavie, Udi Gelbshtein, Sigal Liverant-Taub, Arnona Ziv, et al. Development and validation of a machine learning model predicting illness trajectory and hospital utilization of COVID-19 patients—a nationwide study. *Journal of the American Medical Informatics Association: JAMIA*, 2021.
- John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck. Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2:e55, 2016.
- John B. Wong. Pandemic Surge Models in the Time of Severe Acute Respiratory Syndrome Coronavirus-2: Wrong or Useful? *Annals of Internal Medicine*, 2020. <https://www.acpjournals.org/doi/10.7326/M20-1956>.

A DATASETS

Massachusetts data. We selected 8 hospital sites in Suffolk County, Massachusetts, USA targeted at adult patients with the largest volume of cases (all other sites had fewer than 50 patients per day). We used public data sourced from the Department of Public Health of the state of Massachusetts², which provided the total count of hospitalized patients with either suspected or confirmed cases of COVID-19, in both the general hospital or intensive care unit.

The observed counts for 3 representative sites (out of the 8 total) for our selected study period are shown in Fig. A.1. In this data, we see a general decline across all sites, as expected as the data is from a transitory phase of the pandemic in Massachusetts from the spring surge leading into the summer lull.

UK data. We selected 10 hospital sites in England, United Kingdom with consistent data availability and the largest volume of cases. We used public data sourced from the National Health Service of the UK³, which provided the count of beds occupied by COVID-19 patients on each day at each hospital. We excluded sites whose datasets contained either zeros or counts that had a difference of greater than 50 from the previous 4 days and next 4 days, assuming those unrealistic values were inaccurate.

B RESULTS ON RETROSPECTIVE TASKS

For our retrospective evaluation, we selected a study period of April 29, 2020 to July 6, 2020. Both MA and UK datasets had data available in this period. To evaluate our forecasts, we chose to set our future duration to $F = 14$ days (2 weeks). We selected this because 2 weeks into the future is a sensible planning horizon that allows some actionable logistics (expanding available beds, hiring traveling nursing staff) while remaining potentially predictable (forecasting the dynamics of hospitalizations caused by this disease further than 2 weeks from the present is a far more challenging problem with a low likelihood of success without incorporating other leading indicators such as testing rates and community activity levels). We set aside the last 14 days of data as the test set (used only for computing heldout likelihoods), and then treated the 14 days before that as the validation set (used for hyperparameter selection). The remaining $T = 41$ days served as our training set.

B.1 COMPARISON OF LIKELIHOODS: GENERALIZED VS. STANDARD POISSON

Our first experiment sought to justify which likelihood model between the standard and generalized Poisson best suits our observed count data. While the standard Poisson distribution is more commonly used, the generalized Poisson offers more flexibility in dispersion.

We trained the single-site generalized auto-regressive (GAR) model for all 8 sites from Massachusetts with $W = 1$ using both the standard Poisson and the generalized Poisson. Figure B.1 shows a plot of multiple estimates of the heldout log likelihoods on the validation set under both likelihoods. These multiple estimates (derived from several Monte Carlo estimates from different MCMC chains) help us understand when differences are due to the underlying model and not just luck of the draw.

The primary takeaway is that we see greater heldout likelihoods when we use the generalized Poisson (clear improvement in 6 out of the 8 sites and indistinguishable performance in the other 2). Recall that using the standard Poisson likelihood is equivalent to using the generalized Poisson with λ fixed at 0 rather than being a learned parameter. While the two models learn similar β coefficients, the learned λ parameter for the generalized Poisson concentrates well below zero for all sites, indicating the model is able to adjust for the underdispersion in the data. As a result, we don't overestimate the variance in the data and have greater certainty in our forecasts.

²<https://www.mass.gov/info-details/covid-19-response-reporting>

³<https://www.england.nhs.uk/statistics/statistical-work-areas/covid-19-hospital-activity/>

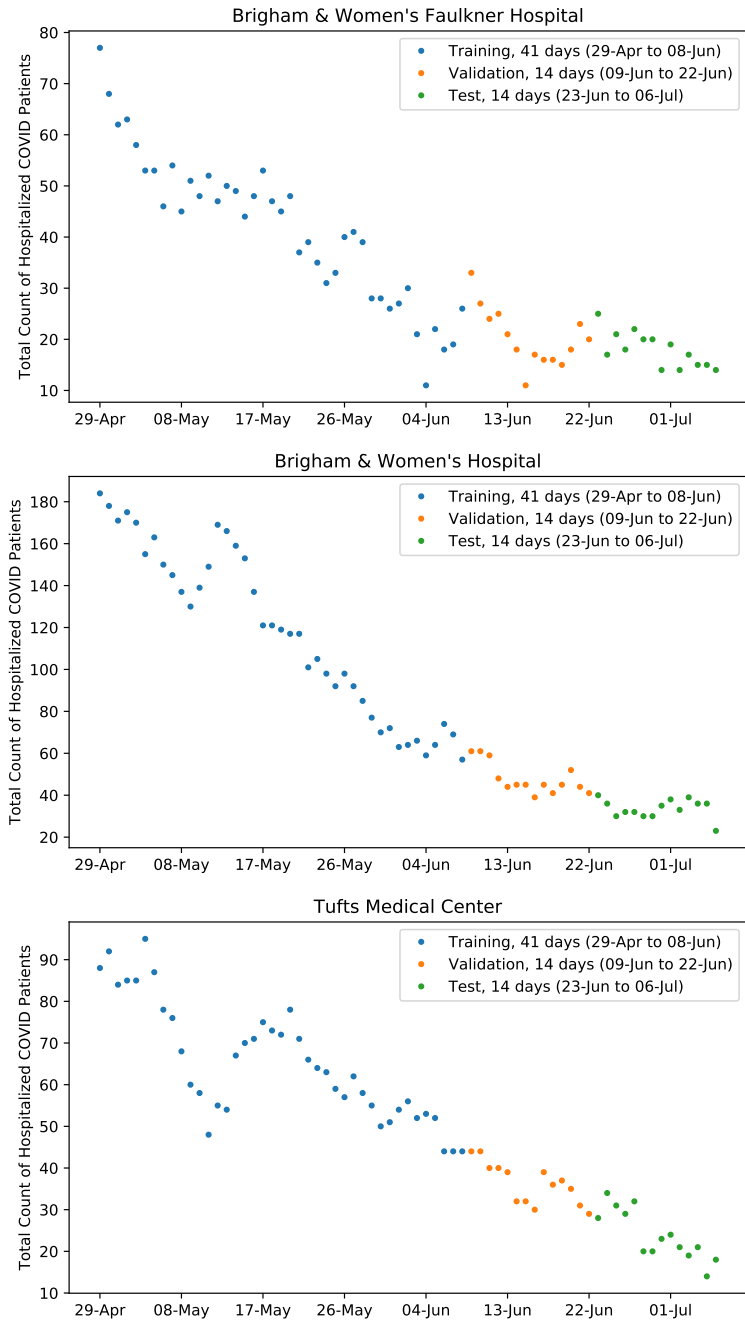


Figure A.1: **Raw data from 3 representative hospital sites in our Massachusetts dataset.** We show the training, validation, and test splits used in experiments.

B.2 COMPARISON OF LATENT FUNCTION MODELS: AR VS GP

Next, we sought to understand which of our latent function models—the autoregressive (GAR) or Gaussian process (GGP)—offered the best fit to our data. We use a generalized Poisson likelihood for all these experiments.

Experimental Setup. For both single-site models (GAR and GGP), we ran a grid search over hyperparameters. For each hyperparameter configuration, we condition on the training set when sampling from the posterior and then evaluate likelihoods on the validation set. We then combined the

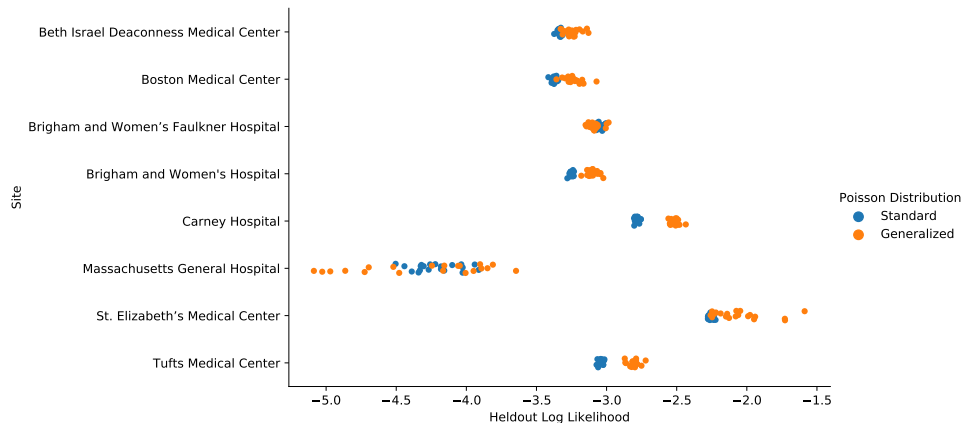


Figure B.1: **Comparison of standard and generalized Poisson likelihoods.** The generalized Poisson outperforms the standard Poisson in general, with clear improvement in six sites and indistinguishable performance at two sites (Faulkner and Mass. General). For each site, there are 20 orange points (generalized Poisson) and 20 blue points (standard Poisson), showing estimates of the count-normalized log likelihood on the validation set after training the single-site GAR model with $W = 1$ (simplest model). The 20 visualized points for each model come from 2 separate MCMC chains of 5000 samples each, divided into 10 groups of 500, to better indicate any mixing problems within chains.

training and validation sets together using the hyperparameters with the best heldout performance, and evaluated the result on the test set.

For the single-site GAR model, we searched the following set of window sizes W : [1, 2, 5, 7, 10, 14]. For the single-site GGP model, we searched the following set of time-scale prior means μ_t : [0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50].

We further evaluated the multi-site GAR model. Here, we fixed a window size of $W = 1$, avoiding an expensive grid search. We found that performance was very similar between $W = 1$ and $W = 2$, especially averaged across sites, while $W = 1$ was substantially faster to train (at least 5 times faster).

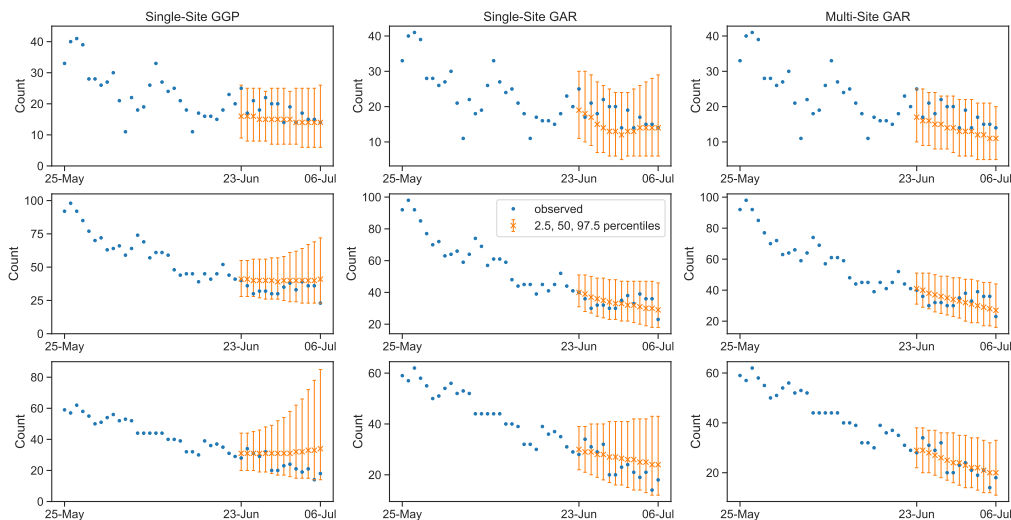


Figure B.2: **Qualitative forecasts for 3 representative hospital sites selected from the 8 available sites in our MA dataset.** From top to bottom, the 3 sites are: Brigham & Women’s Hospital, Brigham & Women’s Faulkner Hospital, and Tufts Medical Center. Each column shows one of our methods of interest: single-site GGP, single-site GAR, and multi-site GAR. We show the true observed values and the 2.5th, 50th, and 97.5th percentiles of the sampled distribution on each day ($S=5000$ total samples). In general, the GGP (left) sometimes learns very wide predictive intervals, while the multi-site GAR (right) delivers better calibrated intervals.

Hospital Site (MA, USA)	Single-Site GGP	Single-Site GAR	Multi-Site GAR
Beth Israel Deaconess	-3.153 ± 0.005 -3.148 ± 0.003	-3.121 ± 0.017 -3.127 ± 0.020	-3.010 ± 0.011 -3.011 ± 0.012 *
Boston Medical Center	-3.376 ± 0.002 -3.379 ± 0.004	-3.454 ± 0.025 -3.406 ± 0.021	-3.296 ± 0.014 -3.295 ± 0.015 *
Brigham & Women’s Faulkner	-2.719 ± 0.002 -2.719 ± 0.003	-2.684 ± 0.010 -2.671 ± 0.014	-2.601 ± 0.008 -2.583 ± 0.006 *
Brigham & Women’s Hospital	-3.118 ± 0.001 -3.114 ± 0.001	-2.978 ± 0.009 -2.948 ± 0.012 *	-3.044 ± 0.016 -3.069 ± 0.027
Carney Hospital	-3.072 ± 0.008 -3.073 ± 0.007	-2.721 ± 0.105 -2.551 ± 0.080	-2.564 ± 0.090 -2.619 ± 0.079 *
Massachusetts General	-3.777 ± 0.020 -3.775 ± 0.019	-3.468 ± 0.018 -3.463 ± 0.013	-3.450 ± 0.017 -3.490 ± 0.025 *
St. Elizabeth’s	-1.980 ± 0.007 -1.983 ± 0.010	-1.312 ± 0.120 -1.129 ± 0.122 *	-1.598 ± 0.007 -1.609 ± 0.002
Tufts Medical Center	-2.937 ± 0.005 -2.936 ± 0.004	-2.822 ± 0.010 -2.821 ± 0.016	-2.728 ± 0.007 -2.750 ± 0.007 *
Hospital Site (UK)	Single-Site GGP	Single-Site GAR	Multi-Site GAR
Barts Health	-4.730 ± 0.058 -4.706 ± 0.043 *	-6.727 ± 0.221 -6.758 ± 0.118	-4.842 ± 0.078 -4.852 ± 0.055
Chelsea & Westminster	-2.937 ± 0.009 -2.942 ± 0.013 *	-3.141 ± 0.030 -3.183 ± 0.023	-5.929 ± 0.086 -5.880 ± 0.106
Imperial College Healthcare	-4.017 ± 0.019 -4.039 ± 0.049	-3.063 ± 0.178 -3.116 ± 0.145 *	-3.976 ± 0.110 -4.100 ± 0.107
King’s College Hospital	-3.012 ± 0.003 -3.017 ± 0.003	-2.749 ± 0.020 -2.735 ± 0.014	-2.695 ± 0.005 -2.695 ± 0.007 *
London North West University	-2.829 ± 0.008 -2.807 ± 0.017	-2.488 ± 0.014 -2.506 ± 0.017 *	-2.572 ± 0.009 -2.560 ± 0.007
Manchester University	-12.335 ± 0.647 -11.275 ± 0.807	-12.735 ± 0.937 -13.028 ± 0.924	-6.152 ± 0.167 -6.021 ± 0.241 *
North Middlesex University	-1.593 ± 0.008 -1.588 ± 0.006 *	-1.677 ± 0.022 -1.656 ± 0.015	-1.703 ± 0.008 -1.708 ± 0.012
Nottingham University	-3.443 ± 0.018 -3.472 ± 0.018	-3.324 ± 0.066 -3.275 ± 0.047	-2.950 ± 0.032 -2.986 ± 0.026 *
University Hospitals Birmingham	-4.385 ± 0.005 -4.371 ± 0.010 *	-4.571 ± 0.019 -4.548 ± 0.033	-4.583 ± 0.031 -4.620 ± 0.023
Univ. Hospitals of North Midlands	-3.188 ± 0.002 -3.187 ± 0.003	-2.953 ± 0.033 -2.993 ± 0.028	-2.829 ± 0.016 -2.841 ± 0.016 *

Table B.1: **Heldout likelihood evaluation across 8 sites in MA (top) and 10 sites in UK (bottom).** We report test-set log likelihood (normalized by total test days). The entry marked (*) in each row indicates the best result. For each model, we report results from 2 separate MCMC chains to assess reliability (large differences across chains indicate convergence problems). Within each chain of 5000 samples, we gather 10 groups of 500 samples, and report the mean ± SEM to communicate uncertainty. For all methods, we run MCMC given all observations from the combined training-plus-validation dataset, and report likelihoods on the test set. For single-site GGP and single-site GAR, we select hyperparameters values via grid search on the validation set. For multi-site GAR, we set the window size to $W = 1$ to keep runtime affordable.

Qualitative Results. Figure B.2 shows qualitative forecasts for 3 representative sites produced by the three models of interest (single-site GGP, single-site GAR, and multi-site GAR).

We find in general that the GAR model is more reliable. Although the GGP can sometimes achieve better performance, it is difficult to find its ideal parameters. Because the sequences of census counts during our study period are relatively linear, likely a GP with a learnable linear mean function would have done better. Under our chosen model with a non-zero constant mean function, large time-scales are required to achieve better predictive performance. We chose a prior distribution over time-scales with larger mean and small standard deviation in order to inductively bias the posterior towards larger time scales; with a larger prior standard deviation and more freedom, the posterior time-scale might instead become smaller (improving some training predictions at the cost of generalization). The single-site GGP tends to be sensitive to the variability in the data over smaller time windows, causing the predictions curve away from the training data and prediction intervals to expand. On the other hand, the GAR learns the overall linear nature of the time series, leading to flatter predictions and intervals that expand less. The multi-site GAR achieves even greater certainty with more training data and less sensitivity to variability in the data.

Quantitative Results. Table B.1 gives the quantitative results for the Massachusetts data and UK data. We show heldout log likelihood for all three models, indicating the mean \pm the standard error of the mean (SEM) for multiple MCMC chains (to help diagnose any convergence issues).

MA Results. In the experiments on Massachusetts data, the single-site GAR performed better than the single-site GGP, and the multi-site GAR performed better than the single-site GAR. In 7 out of 8 sites, the single-site GAR achieved better performance than the single-site GGP. The multi-site GAR achieved better performance than the single-site GAR in 4 out of 8 sites, and similar performance in 3 sites. In all 8 sites, the multi-site GAR outperformed the single-site GGP. We conclude that the multi-site GAR is the best model for our data. While it isn't consistently better than the single-site GAR, it offers a much faster forecasting model because it requires us to train only one set of parameters for the entire set of sites, rather than one set of parameters for each site.

UK Results. In general, we had greater success with the Massachusetts data than the UK data. We did not see the multi-site GAR generally outperform the other models on the UK data because there was less similarity between hospital sites. While the hospital sites in the Massachusetts dataset are all in the same Massachusetts county (within about 30 miles), the hospital sites in the UK dataset are from a much larger region (within about 300 miles). Additionally, while the Massachusetts datasets are relatively smooth, the UK datasets have a lot more jumps and irregularities. For the sites that have smoother count series (like King's College Hospital, Nottingham University Hospitals, and University Hospitals of North Midlands), we see the same pattern as the Massachusetts data, where the GAR outperforms the GGP, and the multi-site model outperforms the single-site model. On the other hand, we don't see the same trends in the results for sites with more jumps and irregularities (like Chelsea & Westminster Hospital and Imperial College Healthcare). The gains from GGP to GAR and from single-site to multi-site come from tighter predictions/less uncertainty, but the datasets that are less smooth don't benefit from that.

C RESULTS ON PROSPECTIVE TASKS

We show in Fig. C.1 an expanded set of qualitative visual forecasts from our prospective evaluation on all 3 sites of the hospital system of interest in Massachusetts. The figure also indicates the mean absolute error (MAE) of each method.

A few key conclusions of our prospective evaluation are noticeable from this figure:

- All methods capture the essentially linear trend of the data in the testing period reasonably well.
- Between the two baselines (rescaled linear regression, rescaled IHME forecasts), we see consistently larger uncertainty intervals for the IHME forecasts. Rescaled linear regression appears to be the stronger baseline in terms of MAE: improvements of over 2.0 at two sites (LGH and TMC), and a difference under 0.5 at the third site (Melrose-Wakefield).
- The single-site GAR and multi-site GAR deliver similar qualitative trends and prediction accuracy. The multi-site GAR has slightly worse MAE, but this is likely insignificant (its MAE is within 0.5 of the single-site model, so the daily difference in counts is less than one patient). However, the multi-site GAR gives uncertainty intervals that are narrower while still fully capturing the true values. (Note that MAE is calculated using only the mean prediction values and doesn't account for uncertainty, which is a key goal of our forecasts.)
- Our single-site and multi-site GAR models appear to either match or outperform our baseline models on all three sites. We emphasize that our models' uncertainty intervals sensibly grow over time (unlike the frequentist linear regression baseline), accounting for larger uncertainty further in the future.

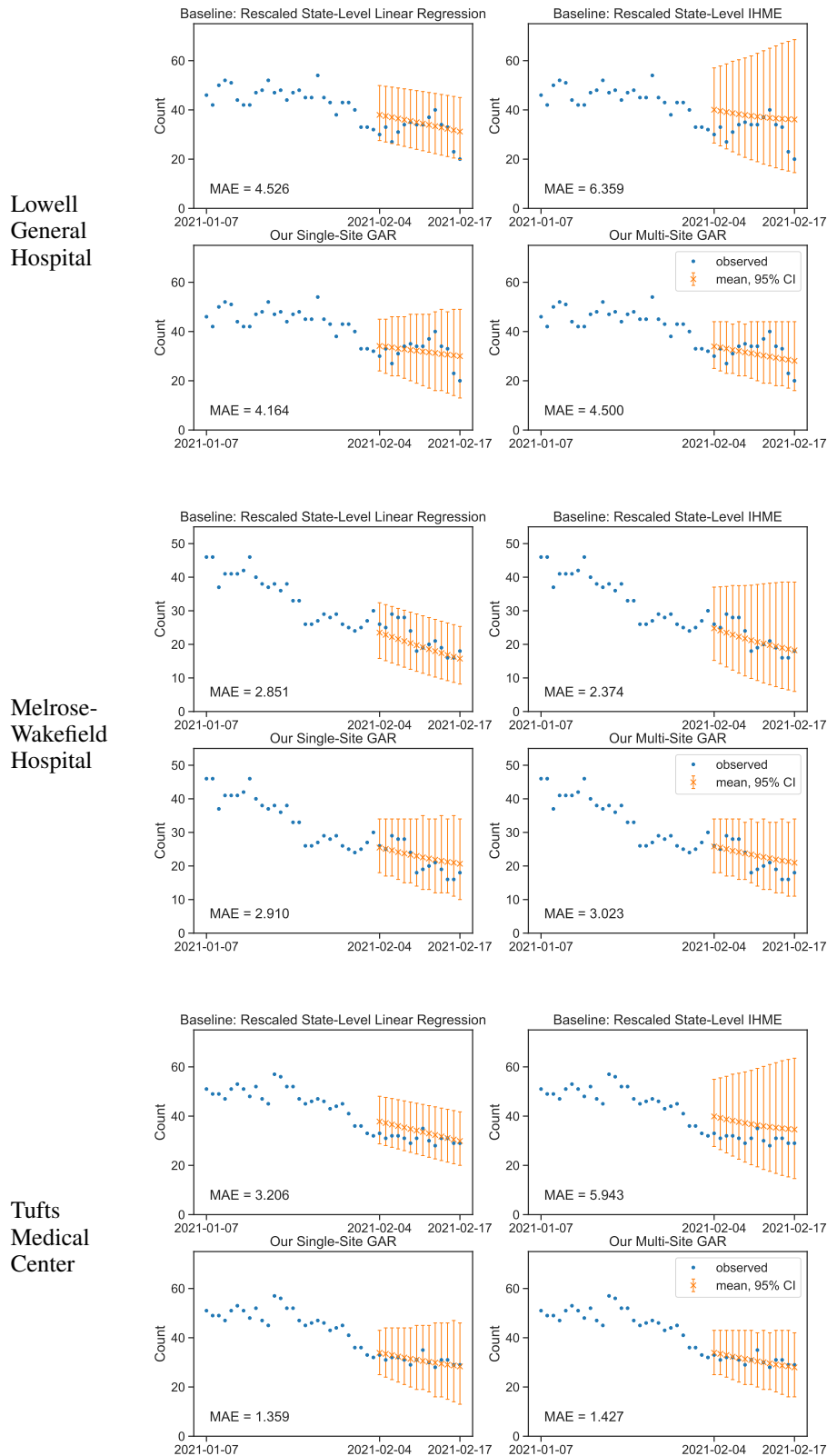


Figure C.1: Prospective evaluation of proposed models against baselines in use at all 3 hospital sites of the major hospital system we studied. All proposed GAR models deliver competitive mean absolute error and plausible interval estimates where uncertainty increases slightly with time.

D METHOD DETAILS: LIKELIHOODS FOR COUNT DATA

In this section, we review possible distributions for modeling observed data y which represents *counts* or non-negative integers: $y \in \{0, 1, 2, \dots\}$.

Poisson Likelihood. Recall the Poisson distribution over non-negative integer random variable Y has probability mass function:

$$p(Y = y | \theta) = \frac{1}{y!} \theta^y e^{-\theta}, \quad y \in \{0, 1, 2, \dots\}, \quad (6)$$

where the parameter $\theta > 0$ controls both the mean and the variance of Y :

$$\mathbb{E}[Y] = \theta, \quad \text{Var}[Y] = \theta. \quad (7)$$

Generalized Poisson Likelihood. The generalized Poisson distribution (Consul & Famoye, 1992) is an extension of the standard Poisson. Unlike the standard Poisson distribution, which has a single parameter determine both mean and variance and thus assumes equidispersion (the mean is the same as the variance), the generalized Poisson has a separate dispersion parameter that allows us to model count data that is either underdispersed (variance is less than the mean value) or overdispersed (variance is greater than the mean value).

A generalized Poisson random variable Y has the following probability mass function:

$$p(Y = y | \theta, \lambda) = \frac{1}{y!} \theta(\theta + \lambda y)^{y-1} e^{-\theta - \lambda y}, \quad y \in \{0, 1, 2, \dots\}, \quad (8)$$

where $\theta > 0$ and $\max(-1, -\frac{\theta}{4}) \leq \lambda \leq 1$. The mean and variance are given by

$$\mathbb{E}[Y] = \frac{\theta}{1 - \lambda}, \quad \text{Var}[Y] = \frac{\theta}{(1 - \lambda)^3}. \quad (9)$$

When $\lambda = 0$, the generalized Poisson reduces to the standard Poisson with mean θ . When $\lambda < 0$, the model has underdispersion; when $\lambda > 0$, the model has overdispersion.

Priors on likelihood dispersion parameter λ . For the generalized Poisson, we make the a priori assumption that the count data has symmetric chances of both under- and over-dispersion. We thus set the prior on the dispersion parameter λ to be

$$\lambda \sim \text{TruncatedNormal}(0, 0.3, \text{lower} = -1, \text{upper} = 1). \quad (10)$$

Sampling from the generalized Poisson. We generate random samples from a generalized Poisson using the Inversion Algorithm (Famoye, 1997), whereby a random sample is drawn from the uniform distribution and then plugged into the inverse of the cumulative distribution function. The inverse CDF of the generalized Poisson distribution is available in closed-form, making this possible.

E METHOD DETAILS: SINGLE-SITE GAR

First, we consider a latent autoregressive process for count data (which we call the generalized autoregressive model or ‘‘GAR’’). This model is ‘‘generalized’’ in the same sense as generalized linear models; that is, we may explore multiple possible likelihoods for count data (standard Poisson and generalized Poisson), rather than simply assuming the data is normally distributed.

Our generalized autoregressive model is an instance of the generative model $p(f_{1:T}, y_{1:T})$ in Eq. (1) with an order- W autoregressive process for the latent-generating distribution over $f_{1:T}$

$$p_\alpha(f_{1:T}) = \prod_{t=1}^T p_\alpha(f_t | f_{(t-W_t):(t-1)}), \quad W_t = \min(t - 1, W) \quad (11)$$

where the recent window size W is a hyperparameter.

For most timesteps $t > W$, we generate latent value f_t using the fully-available window of W previous values $f_{t-W:t-1}$:

$$p_\alpha(f_t | f_{(t-W):(t-1)}) = \text{NormalPDF}(\beta_0 + \sum_{\tau=1}^W \beta_\tau f_{t-\tau}, \sigma^2), \quad t \in W+1, W+2, \dots \quad (12)$$

The latent-generating parameters α for the GAR are $\alpha = \{\beta, \sigma\}$, with coefficient vector $\beta = [\beta_0, \beta_1, \dots, \beta_W]$ and standard deviation $\sigma > 0$.

For the first W timesteps, the full window is not available, and so we regress only on the available previous values:

$$p_\alpha(f_t | f_{1:(t-1)}) = \text{NormalPDF}(\beta_0 + \sum_{\tau=1}^{t-1} \beta_\tau f_{t-\tau}, \sigma^2), \quad t \in 1, 2, \dots, W \quad (13)$$

The very first value $t = 1$ is generated with mean β_0 .

A simple case of this model occurs when $W = 1$, yielding the order-1 autoregressive process, which generates each timestep’s latent value using a regression on the previous value f_{t-1} :

$$f_t | f_{t-1} \sim \text{Normal}(\beta_0 + \beta_1 f_{t-1}, \sigma^2) \quad (14)$$

Priors on latent-generating parameters α . For the GAR, our latent-generating parameters are $\alpha = \{\beta, \tau\}$. We place Normal priors over all of the coefficients β , all centered at 0 except for the coefficient on the most recent timestep, whose prior is centered at 1 as we can assume that the future is like the recent past:

$$\beta_0 \sim \text{Normal}(0, 0.1), \quad (15)$$

$$\beta_1 \sim \text{Normal}(1, 0.1), \quad (16)$$

$$\beta_2 \sim \text{Normal}(0, 0.1), \quad (17)$$

$$\dots \quad (18)$$

$$\beta_W \sim \text{Normal}(0, 0.1). \quad (19)$$

We also assume that the standard deviation of the autoregressive process should be close to 0 (since we imagine the latent process as a clean signal of an overall trend which gives rise to noisier count observations). Thus, we set the prior on the standard deviation to be:

$$\sigma \sim \text{HalfNormal}(0.1). \quad (20)$$

Advantages and Limitations. Our GAR model is advantageous due to its simple form of dependency within the latent sequence $f_{1:T}$, which makes both learning and prediction fast. It is limited in the kinds of long-term dependencies it can capture, which motivates our next class of models.

F METHOD DETAILS: SINGLE-SITE GGP

We next consider a latent Gaussian process model, building on early work on Gaussian processes (Rasmussen & Williams, 2006), GP latent variable models (Lawrence, 2003), GP state space models (Damianou et al., 2011), and extensions of GPs to general likelihoods (Chan & Dong, 2011). Our proposed GGP model uses GPs to capture correlations between latent values $f_{1:T}$. Again, this is an instance of the generative model $p(f_{1:T}, y_{1:T})$ in Eq. (1), with a GP for the prior over $f_{1:T}$:

$$f_{1:T} \sim \text{GP}(m_\alpha(t), k_\alpha(t, t')) \quad (21)$$

where we assume a constant mean and squared exponential covariance kernel:

$$m_\alpha(t) = c, \quad k_\alpha(t, t') = a^2 \exp\left(-\frac{(t-t')^2}{2\ell^2}\right) \quad (22)$$

Here, the set of latent-generating parameters α is $\alpha = \{c, a, \ell\}$, with $a > 0$ and $\ell > 0$.

Priors on latent-generating parameters α . We assume that the constant mean function c should be approximately equal to the mean of the set of observed log counts in our training set. We set the covariance amplitude a to be close to 0 to keep noise in the latent sequence f low.

Thus, we place the following priors over the Gaussian process parameters:

$$c \sim \text{TruncatedNormal}(4, 2, \text{lower} = 0) \tag{23}$$

$$a \sim \text{HalfNormal}(2) \tag{24}$$

$$\ell \sim \text{TruncatedNormal}(\mu_\ell, 2, \text{lower} = 0) \tag{25}$$

Here, μ_ℓ is a hyperparameter that controls the time-scale of dependencies. We set this via grid search on validation data. We choose a small standard deviation for the prior on ℓ to keep the posterior distribution close to the hyperparameter values being evaluated.

G METHOD DETAILS: MODEL FITTING AND EVALUATION

Our goal is to make accurate predictions for *future* data given historical counts. To assess each model’s ability to do so, we partition the data into training and evaluation sets by time rather than by random selection. Given a time series of $T + F$ counts, we treat the first T counts as *observed* data, and try to determine our prediction quality for the last F counts.

G.1 POSTERIOR ANALYSIS AND FORECASTING

Armed with our assumed generative model in Eq. (1) with either an AR or GP prior for the latent-sequence-generating distribution $p_\alpha(f_{1:T})$, our analysis goal is to take as input an observed time-series dataset of single site counts $y_{1:T}$ and use this to make useful predictions about future data $y_{(T+1):(T+F)}$. We achieve this in two steps. First, we need to draw samples of parameters α, λ and the past latent sequence $f_{1:T}$ from their posterior given the past counts $y_{1:T}$. Second, given these parameters and latents from the posterior we can either sample future counts or evaluate the likelihood of some given future counts, using the generative model defined by these parameters.

Posterior sampling. Given an observed time-series dataset of single site counts, we wish to first sample from the posterior over parameters α, λ , and latent values f : $p(\alpha, \lambda, f_{1:T} \mid y_{1:T})$. We use the No-U-Turn sampler (Hoffman & Gelman, 2014) to perform Markov chain Monte Carlo approximation of this posterior. Our NUTS sampler is implemented using the PyMC3 toolbox (Salvatier et al., 2016).

Forecast sampling. Given a single posterior sample indexed by s , with parameters α^s, λ^s and latents $f_{1:T}^s$, we can then use the generative model to draw a *forecast* of latents f and counts y for the next F days:

$$f_{T+\tau}^s \sim p_{\alpha^s}(f_{T+\tau} \mid f_{1:(T+\tau-1)}^s), \quad \tau \in 1, \dots, F \tag{26}$$

$$y_{T+\tau}^s \sim p_{\lambda^s}(y_{T+\tau} \mid f_{T+\tau}^s), \quad \tau \in 1, \dots, F \tag{27}$$

Naturally, the independence assumptions in the GAR model make the first equation above reduce to $p_{\alpha^s}(f_{T+\tau} \mid f_{(T+\tau-W):(T+\tau-1)}^s)$.

We typically draw a forecast for each of S distinct samples, using $S = 1000$ or more to be sure we’re capturing the full distribution. We can compute summary statistics of the empirical distribution over these S samples such as mean or median values as well as lower and upper percentiles.

Using the previously described sampling and forecasting methods, we obtain S samples,

$$\alpha^s, \lambda^s, f_{1:T}^s, f_{(T+1):(T+F)}^s \sim p(\alpha, \lambda, f_{1:T}, f_{(T+1):(T+F)} \mid y_{1:T}), \quad s \in 1, 2, \dots, S, \tag{28}$$

which we use to estimate the likelihood of future data given the past.

Computing Probability of Heldout Data. To score the model’s performance, we evaluate the likelihood of heldout future counts:

$$p(y_{(T+1):(T+F)} \mid y_{1:T}) = \int p(y_{(T+1):(T+F)}, f_{(T+1):(T+F)}, \lambda \mid y_{1:T}) df_{(T+1):(T+F)} d\lambda \tag{29}$$

$$= \int p_\lambda(y_{(T+1):(T+F)} \mid f_{(T+1):(T+F)}) p(f_{(T+1):(T+F)}, \lambda \mid y_{1:T}) df_{(T+1):(T+F)} d\lambda \tag{30}$$

This is an expectation with respect to a posterior that conditions on $y_{1:T}$. Given any posterior sample from the full joint $p(\alpha, \lambda, f_{(T+1):(T+F)}, f_{1:T} \mid y_{1:T})$, we can always just drop $\alpha^s, f_{1:T}^s$ to obtain sampled values $\lambda^s, f_{(T+1):(T+F)}^s$ from the posterior needed for the expectation above: $p(f_{(T+1):(T+F)}, \lambda \mid y_{1:T})$. Thus, given S samples we can compute a Monte Carlo approximation of the integral above to estimate the *log* probability of the heldout future counts:

$$\log p(y_{(T+1):(T+F)} \mid y_{1:T}) \approx \log \frac{1}{S} \sum_{s=1}^S p_{\lambda^s}(y_{(T+1):(T+F)} \mid f_{(T+1):(T+F)}^s) \tag{31}$$

$$\approx \log \frac{1}{S} \sum_{s=1}^S \prod_{\tau=1}^F p_{\lambda^s}(y_{T+\tau} \mid f_{T+\tau}^s) \tag{32}$$

This uses our chosen count likelihood probability mass function $p_{\lambda}(\cdot)$, which is typically the generalized Poisson but could be either of the functions in Sec. D.

Practical rescaling. Across different real-world sites, the scale of the heldout log likelihood values computed by the above method is likely to differ substantially between large and small sites. To facilitate a more “sensible” scale, we recommend normalizing all likelihoods by the number of observations F . We thus compute $\frac{1}{F} \log p(y_{(T+1):(T+F)} \mid y_{1:T})$. We find that in practice this makes human interpretation of these values more sensible, as the value has a consistent scale on the order of -10 to -2 regardless of the length of the window F .

H METHOD DETAILS: MULTI-SITE MODELS

We now aim to predict future census counts at H different hospital sites simultaneously, given T previous observations from each site. The intuition is that all H sites share common trends (i.e., whether they increase or decrease over short-term time scales), though there may be site-specific patterns in the observed data (e.g. one hospital may typically have many more patients than another).

We consider a multi-site generative model which ties the latent-sequence-generating parameters α across sites, but allows count-generating likelihood parameters λ^h to be specific to each site (indexed by h).

$$p(\{y_{1:T}^h, f_{1:T}^h\}_{h=1}^H) = \prod_{h=1}^H p_{\alpha}(f_{1:T}^h) p_{\lambda^h}(y_{1:T}^h \mid f_{1:T}^h) \tag{33}$$

For simplicity, for the latent-generating distribution we use the AR model described above with window size W , where its latent-generating parameters $\alpha = \{\beta, \sigma\}$ are shared among all sites.

For the count-generating distribution we use a generalized Poisson, so the site-specific parameter is $\lambda^h \in [-1, +1]$.

Priors on latent-generating parameters α . We use the same priors as for the single-site GAR model:

$$\beta_0 \sim \text{Normal}(0, 0.1) \tag{34}$$

$$\beta_1 \sim \text{Normal}(1, 0.1)$$

$$\beta_2 \sim \text{Normal}(0, 0.1)$$

...

$$\beta_W \sim \text{Normal}(0, 0.1)$$

$$\sigma \sim \text{HalfNormal}(0.1) \tag{35}$$

Priors on count-generating parameters λ . We draw each site’s parameter in i.i.d. fashion from the same truncated normal prior as before:

$$\lambda^h \sim \text{TruncatedNormal}(0, 0.3, \text{lower} = -1, \text{upper} = 1), \quad h \in 1, \dots, H \tag{36}$$

Posterior sampling for the multi-site model. As with the single-site model, for the multi-site model we can obtain S samples from the posterior using modern MCMC posterior sampling methods:

$$\alpha^s, \{\lambda^{s,h}\}_{h=1}^H, \{f_{1:T}^{s,h}\}_{h=1}^H \sim p(\alpha, \{\lambda^h\}_{h=1}^H, \{f_{1:T}^h\}_{h=1}^H \mid \{y_{1:T}\}_{h=1}^H), \quad s \in 1, 2, \dots, S. \quad (37)$$

Then for each site independently, we can draw a *forecast* of future latents $f_{(T+1):(T+F)}^{s,h}$ given the past latents for that site and the shared latent-generating parameters α^s . If a forecast count sample is needed, we can simply draw $y_{(T+1):(T+F)}^{s,h}$ from the relevant site-specific likelihood with sampled parameter $\lambda^{s,h}$.

In order to evaluate heldout likelihoods, we can compute the count-normalized log likelihood using the Monte Carlo estimates in Eq. (32), using S samples of the site-specific parameters λ^h and site-specific latents $f_{(T+1):(T+F)}^h$.