April/May 2021

Aggregate Count Explicit-Duration Hidden Markov Model (ACED-HMM) for COVID-19 Hospital Trajectories

CS RAs: Gian Marco Visani, Alexandra Hope Lee, Cuong Nguyen
Tufts Medical PIs: John B. Wong, David M. Kent, Joshua T. Cohen
CS PI: Michael C. Hughes

Manuscript PDF: https://arxiv.org/pdf/2105.00773.pdf (please cite if you find useful! Currently under peer review)

Python code: https://github.com/tufts-ml/aced-hmm-hospitalized-patient-trajectory-model

Contact: Mike Hughes - mike (AT) michaelchughes.com www.michaelchughes.com

Motivation

Build a **probabilistic** model that can forecast how a hospitalized COVID-19 patient will move through various stages of care (in general ward, in ICU, in ICU on the ventilator)

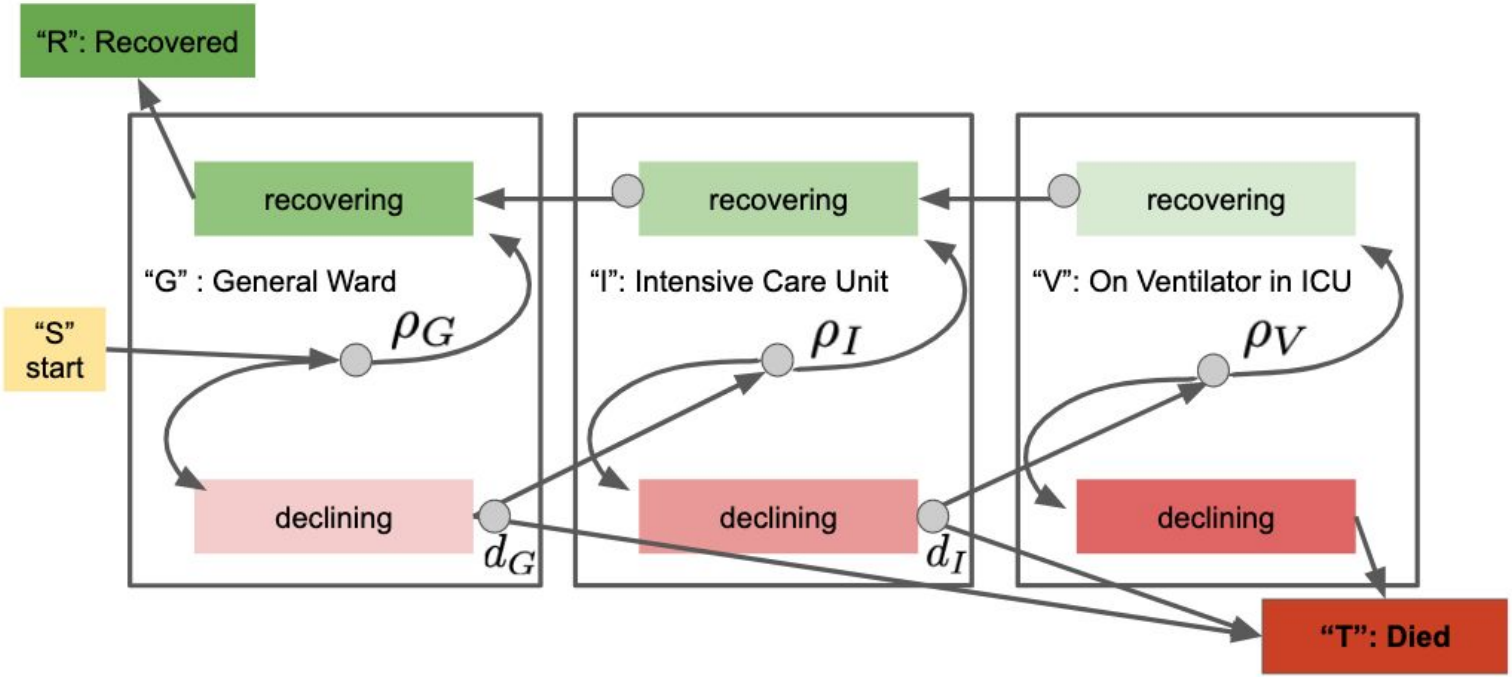
To be **portable** to health systems around the world, we'll assume access only to aggregated daily counts of resource usage (number of occupied beds in general ward, in ICU, on ventilator)

Using this model, we can:

- **fit parameters** to aggregated daily count data from a specific region
- **forecast** future daily counts to help understand future demand for resources
- **assess the societal value** of possible interventions (e.g. would decreasing admissions by X% help California avoid a lockdown in late 2020?)

Proposed model:

Aggregate Count Explicit Duration HMM (ACED-HMM)

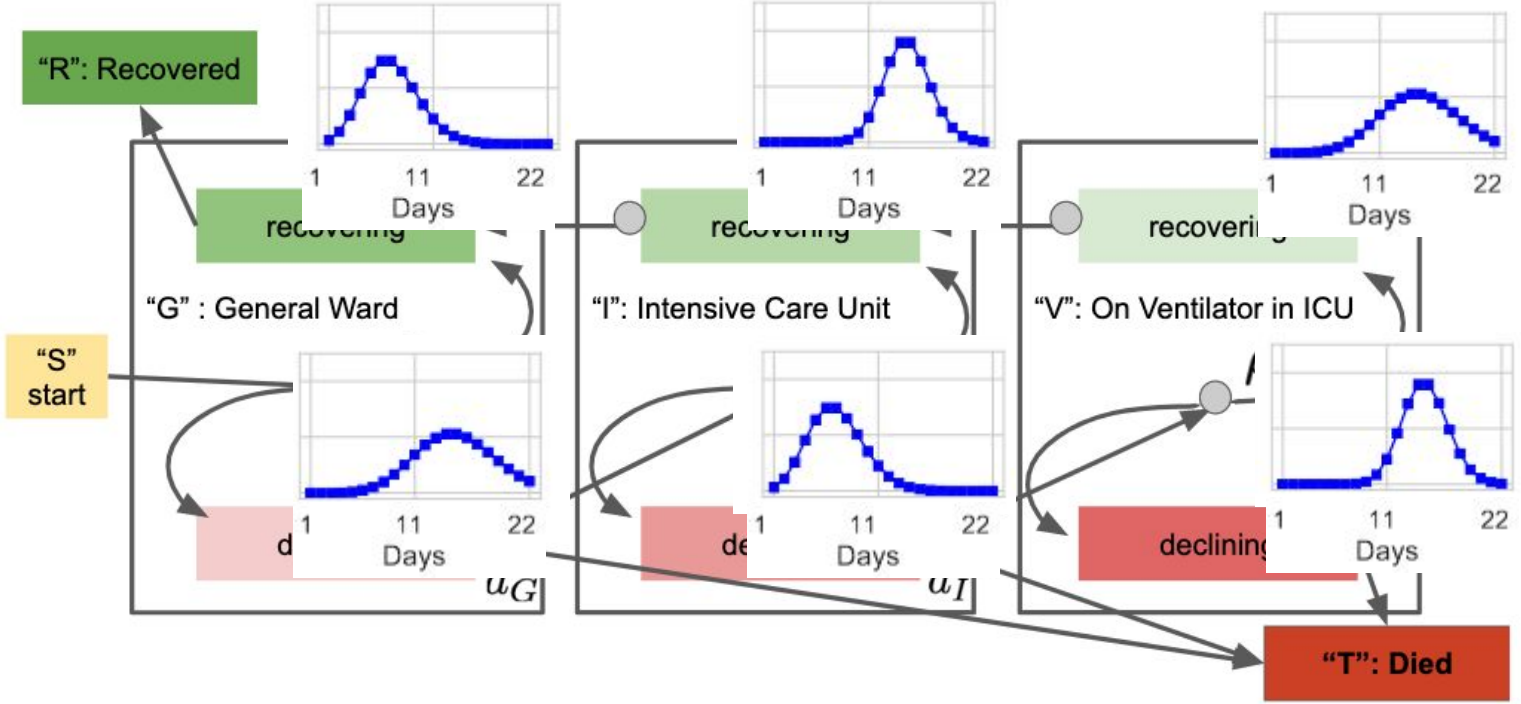


Two key components:

- **1. Transition model** : Proba. of recovery / death after each stage

Proposed model:

Aggregate Count Explicit Duration HMM (ACED-HMM)



Two key components:

- 1. Transition model : Proba. of recovery / death after each stage
- **2. Duration model** : Length of time in each stage / health state

Model parameters to define

```
"proba_Recovering_given_InGeneralWard": 0.1,
"proba_Recovering_given_OffVentInICU": 0.1,
"proba_Recovering_given_OnVentInICU": 0.1,

"proba_Die_after_Declining_OffVentInICU": 0.4,
"proba_Die_after_Declining_OnVentInICU": 1.00,
```

ρ_G
ρ_I
ρ_V
d_G

Transition probabilities

```
"pmf_duration_Declining_InGeneralWard": {
        "1": 0.5, "2": 0.25, "3": 0.125, "4": 0.0625, "5": 0.0625},
"pmf_duration_Recovering_InGeneralWard": {
        "1": 0.5, "2": 0.25, "3": 0.125, "4": 0.0625, "5": 0.0625},
"pmf_duration_Declining_OffVentInICU": {
        "1": 0.5, "2": 0.25, "3": 0.125, "4": 0.0625, "5": 0.0625},
"pmf_duration_Recovering_OffVentInICU": {
        "1": 0.5, "2": 0.25, "3": 0.125, "4": 0.0625, "5": 0.0625},
"pmf_duration_Declining_OnVentInICU": {
        "1": 0.5, "2": 0.25, "3": 0.125, "4": 0.0625, "5": 0.0625},
"pmf_duration_Recovering_OnVentInICU": {
        "1": 0.5, "2": 0.25, "3": 0.125, "4": 0.0625, "5": 0.0625}
```

Explicit duration probabilities

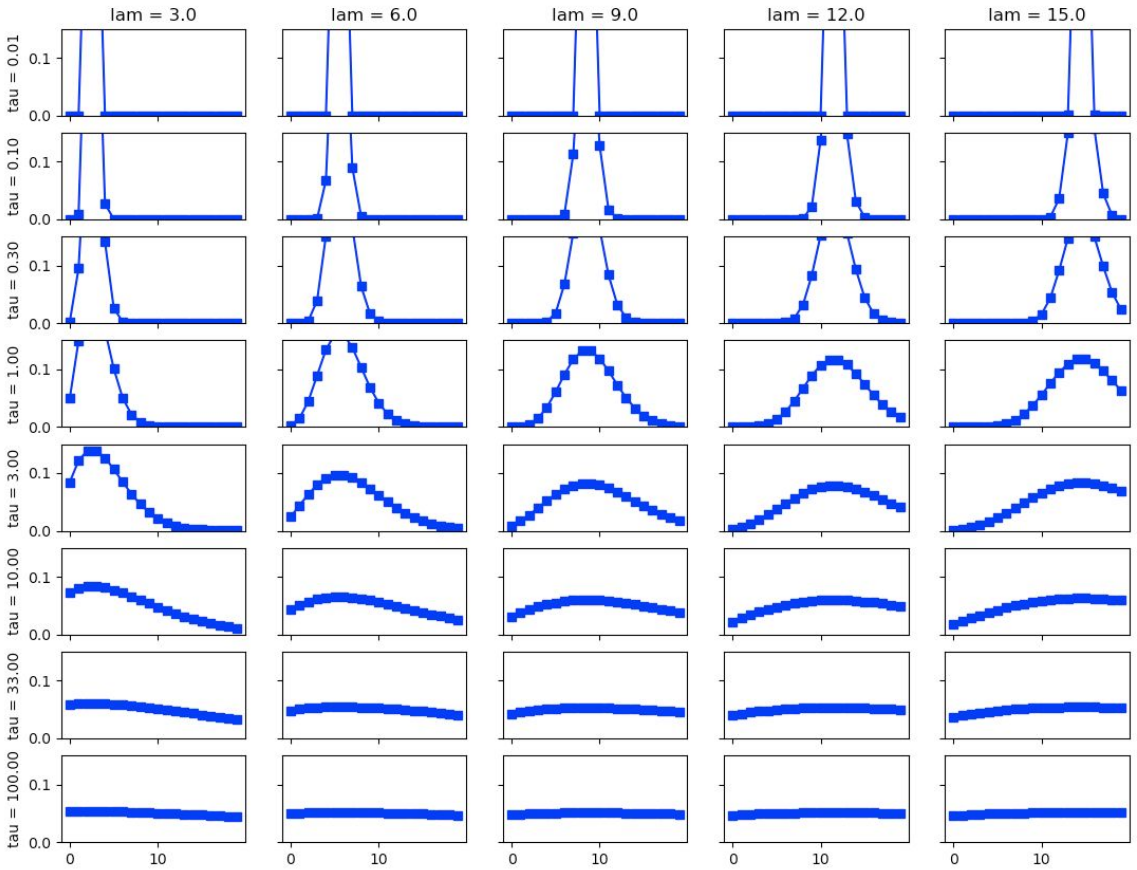
A Simple Smooth Parametrization of Durations

- lambda controls the "mean" of the distribution
- tau controls the entropy of the distribution (is it uniform or is it sharply peaked)

$$[\ell_1, \dots \ell_D] = \text{poisson.logpmf}([1 \dots D], \lambda)$$

$$p(d|\lambda, \tau) = \text{Cat}\left(\text{softmax}\left(\frac{\ell_1, \dots \ell_D}{\tau}\right)\right)$$

Expressive prior while reducing the number of parameters we need to learn from 20+ to 2.



How to adapt parameters to data: Approximate Bayesian Computation - MCMC

Markov chain Monte Carlo without likelihoods

Paul Marjoram*, John Molitor*, Vincent Plagnol†, and Simon Tavaré†‡

Proceedings of National Academy of Sciences (PNAS), 2013
https://www.pnas.org/content/100/26/15324/

MCMC Without Likelihoods. In this section we describe an MCMC approach that is the natural analog of algorithm B in that no likelihoods are used or estimated in its implementation. It is based on the following steps:

F1. If now at θ propose a move to θ' according to a transition kernel $q(\theta \to \theta')$.
F2. Generate $\mathcal{D}'$ using model $\mathcal{M}$ with parameters θ'.
F3. If $\mathcal{D}' = \mathcal{D}$, go to *F4*, and otherwise stay at θ and return to *F1*.
F4. Calculate

$$h = h(\theta, \theta') = \min\left(1, \frac{\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right).$$

F5. Accept θ' with probability h and otherwise stay at θ, then return to *F1*.

The stationary distribution of the chain is indeed $f(\theta|\mathcal{D})$, as is demonstrated below.

How to adapt parameters to data: Approximate Bayesian Computation - MCMC

Markov chain Monte Carlo without likelihoods

Paul Marjoram*, John Molitor*, Vincent Plagnol†, and Simon Tavaré†‡

Proceedings of National Academy of Sciences (PNAS), 2013
https://www.pnas.org/content/100/26/15324/

MCMC Without Likelihoods. In this section we describe an MCMC approach that is the natural analog of algorithm B in that no likelihoods are used or estimated in its implementation. It is based on the following steps:

F1. If now at θ propose a move to θ' according to a transition kernel $q(\theta \to \theta')$.

F2. Generate $\mathcal{D}'$ using model $\mathcal{M}$ with parameters θ'

F3″. If $\rho(S', S) \leq \varepsilon$, go to *F4*, and otherwise stay at θ and return to *F1*,

F4. Calculate

$$h = h(\theta, \theta') = \min\left(1, \frac{\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right).$$

F5. Accept θ' with probability h and otherwise stay at θ, then return to *F1*.

The stationary distribution of the chain is indeed $f(\theta|\mathcal{D})$, as is demonstrated below.

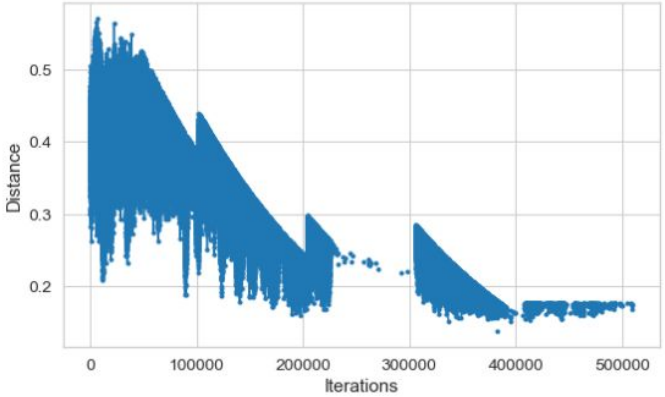


Figure D.1: **Trend of accepted distanced on one run of ABC for ACED-HMM on South Tees hospital data.** Each data point indicates a set of parametrs that has surpassed the first stage of acceptance (distance below ε). Around iteration 230,000 the algorithm got stuck in a local optimum, where few proposals were getting accepted. The resetting of ε after iteration 300,000 helped the algorithm get unstuck from the local optimum.

Distance decreases (fit to observed data gets better) as iterations go on.

Accept threshold epsilon gradually decayed over time to enforce better and better fits

Fit to counts from a specific region

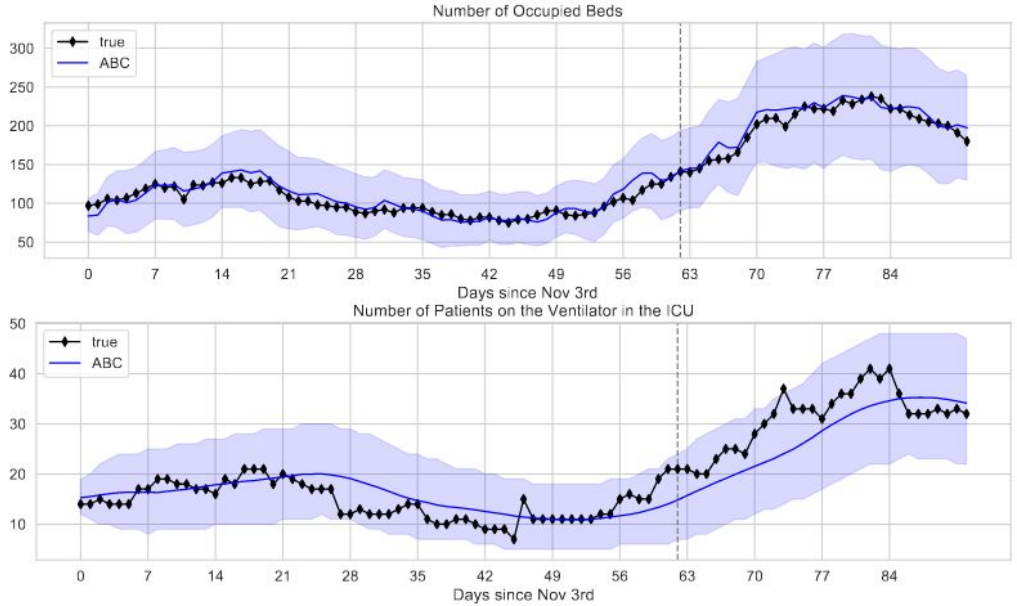


Figure 4: **Fit and forecasts on Occupied Beds and Ventilator counts for South Tees hospital in the UK.** Parameters were selected via ABC to fit 2 months of counts (November 3rd to January 3rd, left of the vertical dashed line). Shaded intervals show the 2.5th and 97.5th percentiles of 2000 posterior samples.

Quantitative error assessment

Train on data from Dec. 11 - Jan. 11, 2021
Evaluate on Jan. 11 - Feb 11, 2021

		In general ward G MAE	*In ICU (on and off ventilator)* I + V MAE
MA	ACED-HMM + ABC	65.0	15.7
	ACED-HMM + Prior	868.6	293.6
	AR-Poisson	498.0	162.2
	IHME	1066.1	104.0
	Mean Test y	1141.6	392.5
SD	ACED-HMM + ABC	9.2	4.5
	IHME	71.8	17.7
	Mean Test y	102.5	34.9
UT	ACED-HMM + ABC	20.8	18.2
	IHME	332.9	105.0
	Mean Test y	272.9	164.3

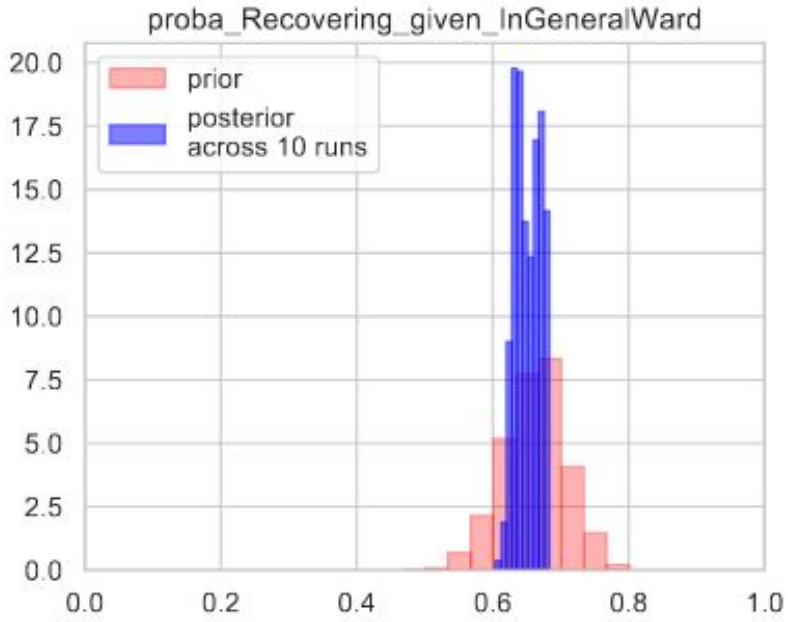
On an “average” day in MA, ICU has 390 patients
Our method’s ICU count is off by 16 patients
IHME’s is off by over 100

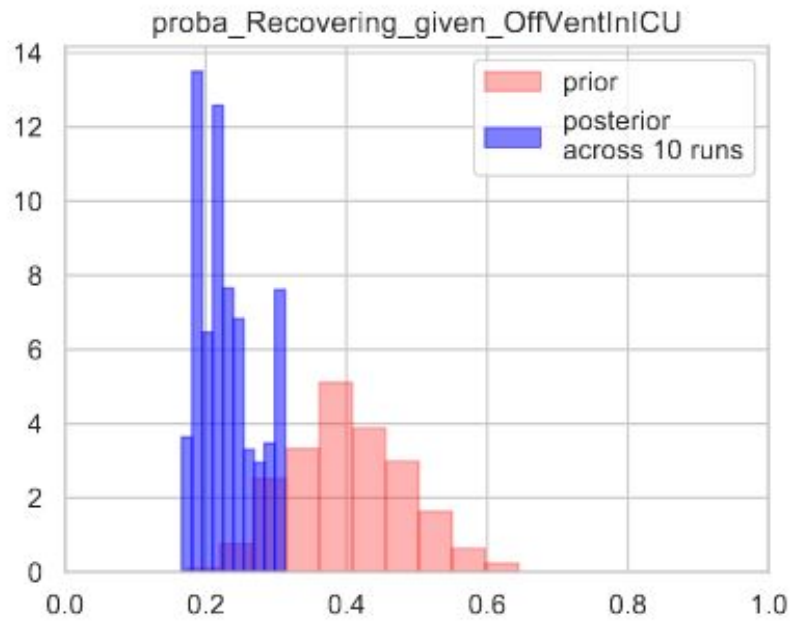
On an “average” day in SD, ICU has 35 patients
Our method’s ICU count is off by 5 patients
IHME’s is off by over 17

On an “average” day in UT, ICU has 164 patients
Our method’s ICU count is off by 18 patients
IHME’s is off by over 100

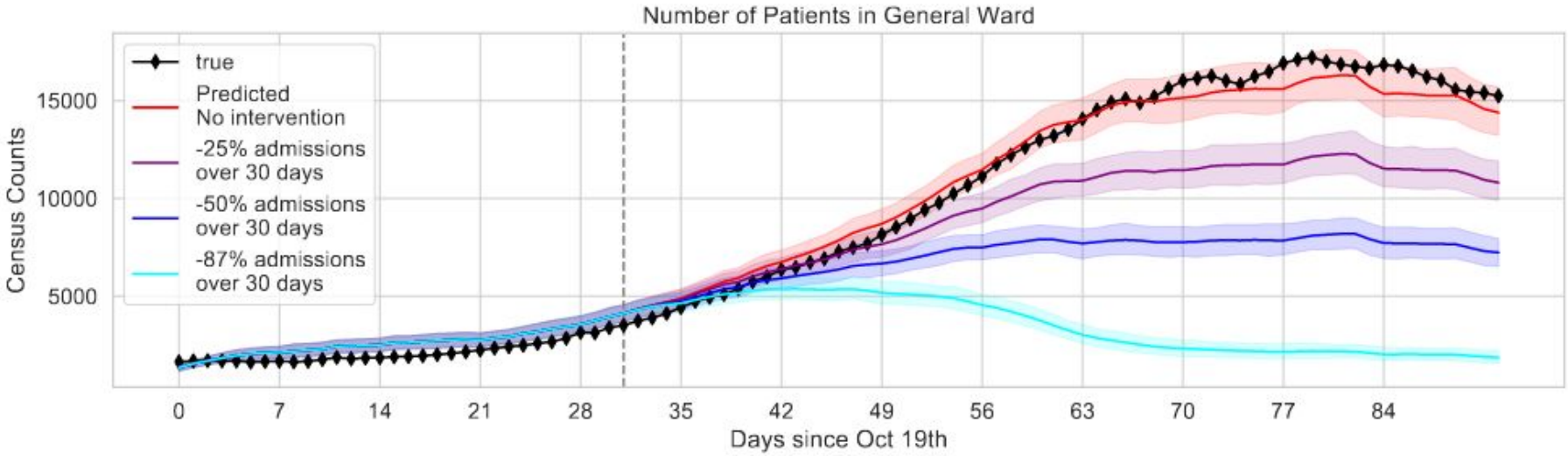
* Not really a “fair” comparison. Our method is given true admissions in test period. See manuscript’s Table B1

Learned distributions over parameters





Forecasts under several “what-if” scenarios



See our [manuscript Sec. 6.6]

Next Steps

- Support you using our ACED-HMM “hospital” model
  - Develop a tutorial and a “recipe” for how a new user could apply this model to a new region
  - What scenarios are you interested in? What help do you need from us?
- Applications of the hospital model
  - Try out model in another region of the globe - India? Japan? Continental Europe?
  - Try out assessments of other possible interventions
- Extensions of the hospital model
  - Incorporate key covariates that impact patient trajectories (transitions and durations)
    - Age - could be simulated using local data about who is coming into the hospital
    - Vaccination status - could be simulated using local data about vaccination rates
- Finalize development of an “infected population” model
  - Allow forecasting of the admission counts, rather than assuming known
  - Model would consume data about local Rt and predict who will need to go to hospital
  - Could be fit to local testing data as well as local admissions data
- Link two models together