Supplementary Information

METHODS

Generative process details

We provide details on the generative process for self-tracked mHealth cycle lengths, which draws per-user specific parameters from population level shared priors:

- **Observed variables:** Observed cycle length $d_{i,c}$, with $c = \{1, \cdots, C_i\}$ cycle lengths for each individual $i = \{1, \cdots, I\}$. Each true cycle length (for user i, cycle c, out of the number of skipped cycles j) is drawn from a Poisson distribution, $d_{i,j,c} \sim p\left(d_{i,j,c} \middle| \lambda_i\right) = Pois\left(d_{i,j,c} \middle| \lambda_i\right)$. The sum of independent Poissons is a different Poisson distribution, so the observed cycle length ($d_{i,c} = \sum_{j=0}^{s_{i,c}+1} d_{i,j,c}$) is also drawn from a Poisson, conditioned on the number of skipped cycles, $d_{i,c} \sim Pois\left(\lambda_i\left(s_{i,c}+1\right)\right)$.
- **Latent variables:** $s_{i,c}$ denotes the number of skipped (not reported) cycles, with $c = \{1, \cdots, C_i\}$ cycle lengths for each individual $i = \{1, \cdots, I\}$. The number of skipped cycles is drawn from a truncated Geometric distribution with a maximum number of skipped cycles S, $s_{i,c} \sim p(s|\pi_i) = \frac{\pi_i^s(1-\pi_i)}{\sum_{s=0}^{S}\pi_i^s(1-\pi_i)} = \frac{\pi_i^s}{\sum_{s=0}^{S}\pi_i^s} = \frac{\pi_i^s(1-\pi_i)}{\left(1-\pi_i^{(S+1)}\right)}$ for $s \in N$.
- **Parameters $\boldsymbol{\lambda_i}$:** the Poisson rate parameters for each individual $i = \{1, \cdots, I\}$. Per-user Poisson rate parameters λ_i are drawn from a population-level Gamma distribution $\lambda_i \sim p(\lambda|\kappa,\gamma) = \frac{\gamma^\kappa}{\Gamma(\kappa)}\lambda^{\kappa-1}e^{-\gamma\lambda}$ for $\lambda > 0$ and $\kappa, \gamma > 0$.

- **Hyperparameters of the Poisson rate parameter:** κ, γ of a Gamma distribution prior for the Poisson rate at the population level.
- **Parameters $\boldsymbol{\pi_i}$:** the probability of skipping a cycle for each individual $i = \{1, \cdots, I\}$. The probability of an individual skipping a cycle is drawn from a population-level Beta distribution $\pi_i \sim p(\pi|\alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\pi^{\alpha-1}(1-\pi)^{\beta-1}$, for $\pi \in [0,1]$ and $\alpha, \beta > 0$.
- **Hyperparameters of the geometric distribution parameters:** α, β of the population level Beta distribution prior on skipping probabilities.

Inference details

Given a dataset of C_i cycle lengths for I users, we perform hyperparameter inference via type-II maximum likelihood estimation. We compute a Monte Carlo (MC) approximation to the negative log-likelihood: $-ln\big(p(d|u)\big) = -ln(\sum_i p(d_i|u))$. Due to the impossibility of integrating out the number of skipped cycles $s_{i,c}$ analytically, we compute a MC approximation to each cycle length likelihood $p(d_i|u)$ with M samples

$$p(d_i|u) = \frac{1}{M}\sum_m p(d_i|\theta_m)\,, \theta_m \sim p(u)$$

where u represents the hyperparameters $[\alpha, \beta, \kappa, \gamma]$ of the distributions from where samples θ_m, representing the parameters $[\lambda_m, \pi_m]$, are drawn. We compute the probability $p(d_i|\theta_m)$ by integrating out the probability of skipping $s_{i,c}$, which is drawn from a truncated geometric distribution:

$$p(d_i|\theta_m) = \prod_{c=1}^{C_i} p(d_{i,c}|\theta_m) = \prod_{c=1}^{C_i}\sum_{s=0}^{S} p(d_{i,c}|\lambda_m, s)p(s|\pi_m)$$

$$= \prod_{c=1}^{C_i}\sum_{s=0}^{S} \left(\left(\lambda_m(s+1)\right)^{d_{i,c}} e^{-\lambda_m(s+1)}/d_{i,c}!\right)\left(\frac{\pi_m^s(1-\pi_m)}{\sum_{s=0}^{S}\pi_m^s(1-\pi_m)}\right)$$

$$= \prod_{c=1}^{C_i} \frac{\lambda_m^{d_{i,c}} e^{-\lambda_m}}{d_{i,c}!} \sum_{s=0}^{S}\left((s+1)^{d_{i,c}} e^{-\lambda_m s}\right)\left(\frac{\pi_m^s}{\sum_{s=0}^{S}\pi_m^s}\right)$$

$$= \prod_{c=1}^{C_i} \phi(\lambda_m) \frac{\sum_{s=0}^{S}(s+1)^{d_{i,c}}\left(\pi_m e^{-\lambda_m}\right)^s}{\sum_{s=0}^{S}\pi_m^s}$$

$$= \prod_{c=1}^{C_i} \phi(\lambda_m) \frac{\sum_{s=0}^{S}(s+1)^{d_{i,c}}\left(\pi_m e^{-\lambda_m}\right)^s}{\frac{1-\pi_m^{S+1}}{1-\pi_m}}$$

$$= \prod_{c=1}^{C_i} \frac{1-\pi_m}{1-\pi_m^{S+1}} \phi(\lambda_m) \sum_{s=0}^{S}(s+1)^{d_{i,c}}\left(\pi_m e^{-\lambda_m}\right)^s$$

where $d_{i,c}$ represents one cycle length c for a given user i, C_i is the number of cycles for user i, S is the maximum value of s, and ϕ is the Poisson density.

Prediction details

In order to update our predictions of per-user cycle length as each subsequent day passes, we are interested in the posterior of the next reported cycle length d^*, conditioned on previous cycle lengths d_i for a user i and the day of the current cycle $d_{current}$,

$$p(d^*|d^* > d_{current}, d_i, \hat{u}) = \frac{p(d^*, d^* > d_{current}|d_i, \hat{u})}{p(d^* > d_{current}|d_i, \hat{u})} = \frac{p(d^*|d_i, \hat{u})I(d^* > d_{current})}{p(d^* > d_{current}|d_i, \hat{u})}$$

where we explicitly indicate that $p(d^*, d^* > d_{current}|d_i, \hat{u}) = 0$ if $d^* \leq d_{current}$.

In addition to characterizing the full distribution, we are interested in computing the expectation of the conditional predictive posterior as a point estimate for the next cycle length,

$$E[p(d^*|d^* > d_{current}, d_i, \hat{u})] = \sum_{d^*} d^* p(d^*|d^* > d_{current}, d_i, \hat{u})$$

$$= \sum_{d^*} d^* \frac{p(d^*|d_i, \hat{u})I(d^* > d_{current})}{p(d^* > d_{current}|d_i, \hat{u})}$$

$$= \frac{\sum_{d^*} d^* p(d^*|d_i, \hat{u})I(d^* > d_{current})}{p(d^* > d_{current}|d_i, \hat{u})}$$

$$= \frac{\sum_{d^* > d_{current}} d^* p(d^*|d_i, \hat{u})}{p(d^* > d_{current}|d_i, \hat{u})}$$

$$= \frac{\sum_{d^*=d_{current}+1}^{D} d^* p(d^*|d_i, \hat{u})}{\sum_{d^*=d_{current}+1}^{D} p(d^*|d_i, \hat{u})}$$

The key term above is $p(d^*|d_i, \hat{u})$:

$$p(d^*|d_i, \hat{u}) = \frac{\int d\lambda d\pi q(\lambda)b(\pi) \sum_{s^*} p(s^*|\pi)p(d^*|s^*, \lambda)p(d_i|\lambda, \pi)}{\int d\lambda d\pi q(\lambda)b(\pi)p(d_i|\lambda, \pi)},$$

where d_i are the cycle lengths for a user i and s_i are the number of skipped cycles for a user, and d^*, s^* are the next reported cycle length and next number of skipped cycles,

respectively. For the truncated geometric distribution on skipping probabilities, we compute the above as

$$p(d^*|d_i, \hat{u}) = \frac{\sum_{m=1}^{M} \frac{1-\pi_m}{1-\pi_m^{S+1}} \sum_{s^*=0}^{S} \pi_m^{s^*} p(d^*|s^*, \lambda_m) p(d_i|\lambda_m, \pi_m)}{\sum_{m=1}^{M} p(d_i|\lambda_m, \pi_m)} .$$

We compute $p(d^*|d_i, \hat{u})$ for a range of cycle length days $d^* = \{0, \cdots, D\}$, normalizing appropriately over d^* for each value of $d_{current}$, using $p(d_i|\lambda_m, \pi_m)$ and $p(d^*|s^*, \lambda_m) = Pois\big(\lambda(s^* + 1)\big)$ (i.e., the Poisson PMF), where we must also normalize $p(d^*|s^*, \lambda)$ over $d^* = \{0, \cdots, D\}$.

Simulated data

In order to assess the ability of our model to recover skipped cycles, we separately train our model on simulated cycle length data for 10,000 users (with $C = 10$ cycles each), generated from our proposed generative process. We then take two cohorts of users: those who have never skipped a cycle in their history, and those who have skipped a cycle in their history. Note that we have access to ground truth cycle length and skipping information in this simulated case. For a sample user from each of these cohorts, we predict their probabilities of possible cycle skips $p(s^*|\hat{u}, d_i, d^* > d_{current})$ for the 11th cycle, utilizing the inferred population-wide hyperparameters $\hat{u}$ and individual cycle length histories d_i. Results in **Figure 3** and **Supplementary Figure 1** use this simulated dataset.

Implementation details

We optimize the negative log-likelihood $-ln\big(p(d|u)\big) = -ln(\sum_i p(d_i|u))$ with respect to hyperparameters u via stochastic gradient descent. Specifically, we utilize Adam[1], an adaptive gradient method. All models have been implemented using PyTorch, and trained with minibatches of size 100. All neural network-based models are trained (with dropout) on the observed cycle lengths for the whole cohort. Predictions are based on each per-user available cycle lengths.

Since we sequentially predict next cycle length, our train-test split is over the number of cycle lengths available, i.e., we train the models on C cycles and predict the $C + 1$th cycle, where $C = \{2, \cdots, 10\}$.

For reproducibility, we provide the settings for priors, learning rate, and other details for each of the models below:

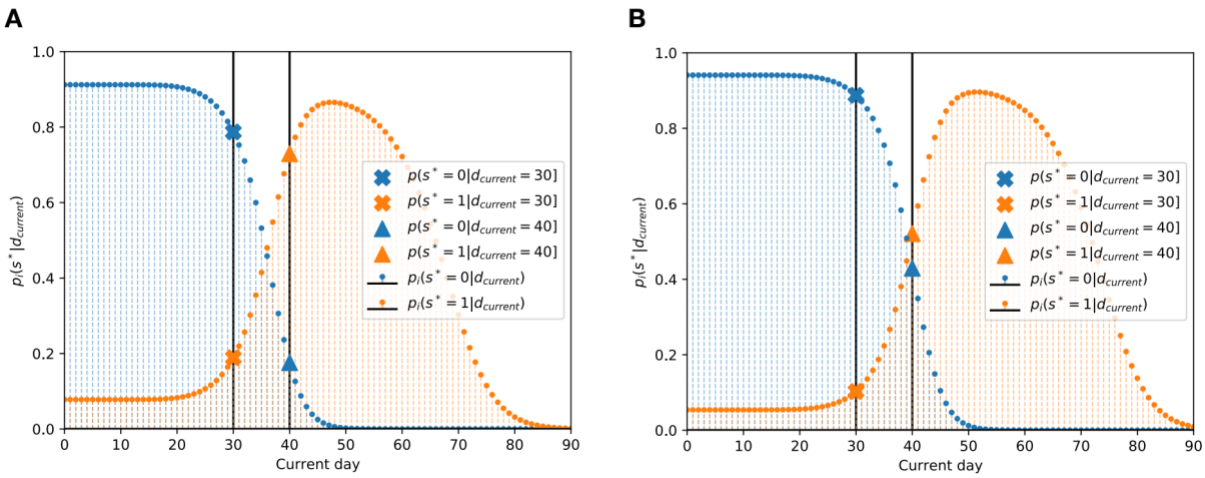
- CNN: number of layers = 1, kernel size = 3, stride = 1, padding = 0, dilation = 1, nonlinearity = tanh, dropout = 0.9, training criterion = MSE, epoch convergence criteria as maximum number of epochs = 1000, loss convergence criteria $\epsilon_{loss} = 1e-3$, optimizer = Adam, learning rate = 0.01.
- RNN: number of layers = 1, hidden size = 3, nonlinearity = tanh, dropout = 0.9, epoch convergence criteria as maximum number of epochs = 1000, loss convergence criteria $\epsilon_{loss} = 1e-3$, optimizer = Adam, learning rate = 0.01.
- LSTM: number of layers = 1, hidden size = 3, nonlinearity = tanh, dropout = 0.9, epoch convergence criteria as maximum number of epochs = 1000, loss convergence criteria $\epsilon_{loss} = 1e-3$, optimizer = Adam, learning rate = 0.01.

- Proposed model: $u_0 = [\kappa_0 = 180, \gamma_0 = 6, \alpha_0 = 2, \beta_0 = 20]$, $S = 10$ (for both inference and prediction), $M = 1000$ (for both inference and prediction), epoch convergence criteria as maximum number of epochs = 1000, loss convergence criteria $\epsilon_{loss} = 1e - 3$, optimizer = Adam, learning rate = 0.01.
- Proposed model (s=0): same as above, with $S = 10$ in inference but $S = 0$ for next cycle length prediction.

RESULTS

Our model flexibly detects possible cycle skips

We display our model's ability to detect skipped cycles by illustrating the probabilities of possible cycle skips, shorthanded as $p(s^*|d_{current})$, on simulated data in **Supplementary Figure 1**. **(a)** showcases a simulated user who has skipped in their history, and **(b)** showcases a simulated user who has never skipped in their history. The vertical lines represent specific days of the next cycle (days 30 and 40), and the markers represent the predicted probability of skipping zero or one cycle on those days.



Supplementary Figure 1: Individual posterior predictive probability of skipping upcoming cycle, $p_i(s^*|d_{current})$, over current day of next cycle $d_{current}$ for two users from simulated data: one who has skipped a cycle in their history **(a)** and one who has never skipped a cycle **(b)**. Our personalized model detects differences in predicted skipping behavior for the two users. Blue and orange curves represent probabilities of skipping zero or one cycle, respectively; markers indicate probability of skipping zero or one cycle on day 30 or 40 of the upcoming cycle. Note that users can also skip more than one cycle. For both example users, we see that the probability of having skipped zero cycles in the upcoming cycle ($p_i(s^*=0|d_{current})$) is high until day 30. However, past day 30, the model detects that the user **(a)** who has skipped in their history is more likely to have skipped the upcoming cycle than for the user **(b)** who has never skipped. This demonstrates how the model takes into account the previous non-skipping behavior of this user. Because data in this experiment is simulated, we know that the user in **(a)** does actually skip the next cycle, while the user in **(b)** does not. Our inferred probabilities recover this, showing that our model can accurately detect when a user is likely to have skipped an upcoming cycle based on their individual cycle length histories and update these beliefs over time.

We choose days 30 and 40 because 30 days is around the average cycle length for these two simulated users, and day 40 represents when the user has surpassed their typical cycle length. These plots showcase how our model is able to detect differences in the underlying skipping phenomena between the two users: as each user passes their 'typical' cycle length without tracking, the model adjusts their likelihood of having skipped tracking a cycle based on their previous skipping behavior. Specifically, **(a)** for the user who has skipped in their history, their probability of skipping one cycle on day 40 is around 0.8, and their probability of skipping zero cycles on day 40 is around 0.2, a significant drop from a near 0.8 probability on day 30. This showcases how the model is able to incorporate knowledge about this user having previously skipped in computing their propensity to skip their next cycle. In comparison, **(b)** the user who has never skipped has a probability of skipping one cycle on day 40 of around 0.5 – it is not as clear that this user may have skipped a cycle, because they have never skipped before (\ie this might be an occasional long cycle for this user, which may occur across menstruators in response to other internal or external stimuli).

While we focus on $s^* = 0{,}1$ in **Supplementary Figure 1**, note that this behavior holds analogously for $s^* = 2$ and beyond. For instance, $p(s^* = 2)$ is low early in the next cycle and peaks past day 60, similar to how $p(s^* = 1)$ starts low and peaks past day 30. The ability to detect and alert users of potential tracking artifacts is important not only to accurately predicting the occurrence of the next cycle, but also to improving the design of mHealth apps as well as the quality of mHealth data for menstrual health research.

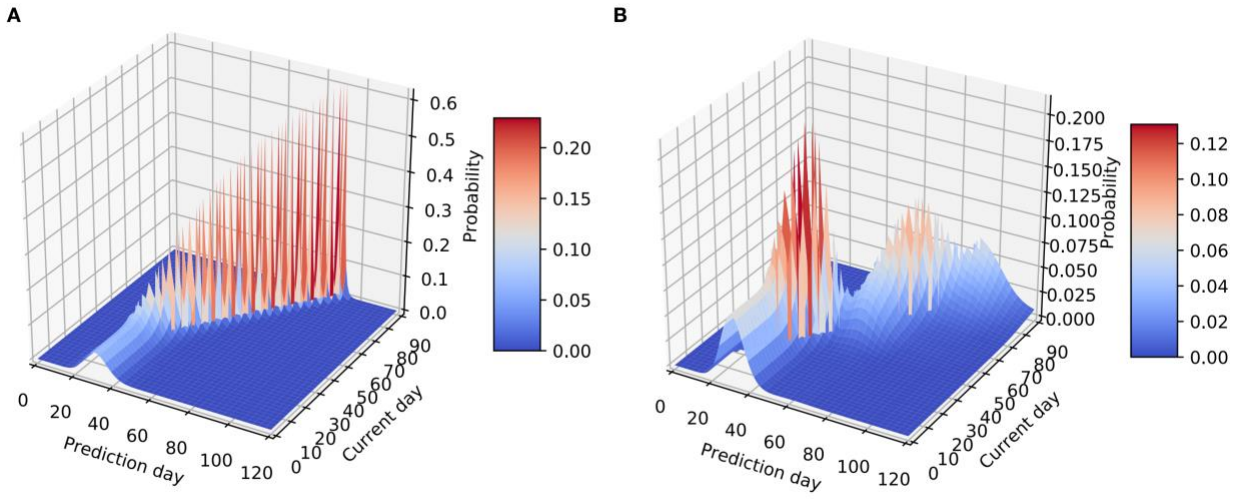
Posterior predictive distribution for cycle length is interpretable and representative of data

In **Supplementary Figure 2**, we showcase our model's posterior predictive distribution for cycle length $p(d^*|\hat{u}, d_i, d^* > d_{current})$, i.e., the probabilistic next cycle length predictions provided by our model, for a specific user (learned as per their previous cycle length history) as the days of the next cycle proceed.

In particular, **Supplementary Figure 2** shows the probability (z-axis) of a user's next cycle being of an specific length (x-axis) for the current day of the cycle (y-axis), assuming **(a)** that their next observed cycle is truth (no skipped cycles, $s = 0$) or **(b)** that their next observed cycle may contain skipped cycles (possible skipped cycles, $s \geq 0$). We are able to accurately update our model's cycle length predictions by updating its beliefs about the likelihood of skipping a cycle over time.

When **(a)** we assume the next cycle is truth, the posterior predictive distribution is unimodal; however, when **(b)** we assume the next cycle may not be truth, the posterior predictive distribution is multimodal, with peaks around $d^* = 30,60,90$.

Such multimodality occurs as a result of (i) conditioning on the day of the next cycle $d_{current}$ and (ii) the explicit modeling of cycle skips, s. This multimodal distribution mirrors the skipping phenomena observed in the dataset – when a user passes their `typical' cycle length (around 30 days), they may have skipped tracking of a cycle. The multimodal posterior predictive distribution is not only easily interpretable, but is also crucial to representing self-tracking artifacts in mHealth data and providing accurate cycle length predictions.



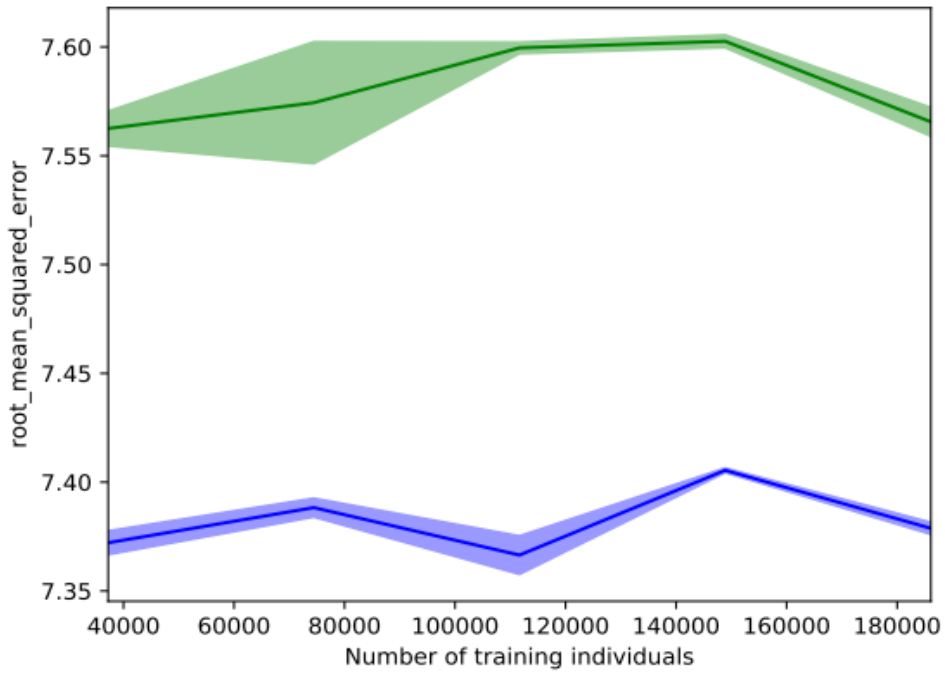
Supplementary Figure 2: Posterior predictive distribution for cycle length over prediction day d^* (i.e., what the next reported cycle is predicted to be) and current day $d_{current}$ (i.e., day in next cycle) for the same user from menstruator data, assuming either that next observed cycle is truth **(a)** or that next observed cycle may contain skipped cycles **(b)**. **(a)** When we assume the next observed cycle is true as reported ($s = 0$), our posterior predictive distribution is unimodal. The probability of the next cycle length is peaked around 30 until around day 30 of the next cycle, after which the peak moves consistently to the right, indicating that our cycle length predictions are consistently increasing past day 30 and not adjusting for the likelihood of skipped cycles. **(b)** When we account for the possibility of skipped cycles with $s \geq 0$, our posterior predictive distribution is multimodal. Prior to day 30 of the next cycle, the distribution is similarly peaked around 30 days, as with the $s = 0$ case. However, when the cycle passes day 30, the distribution shows a peak around day 60, indicating the possibility that a user may have skipped a cycle. This behavior holds analogously past

day 60. Our explicit modeling of cycle skips allows us to identify when a user may have missed tracking a cycle.

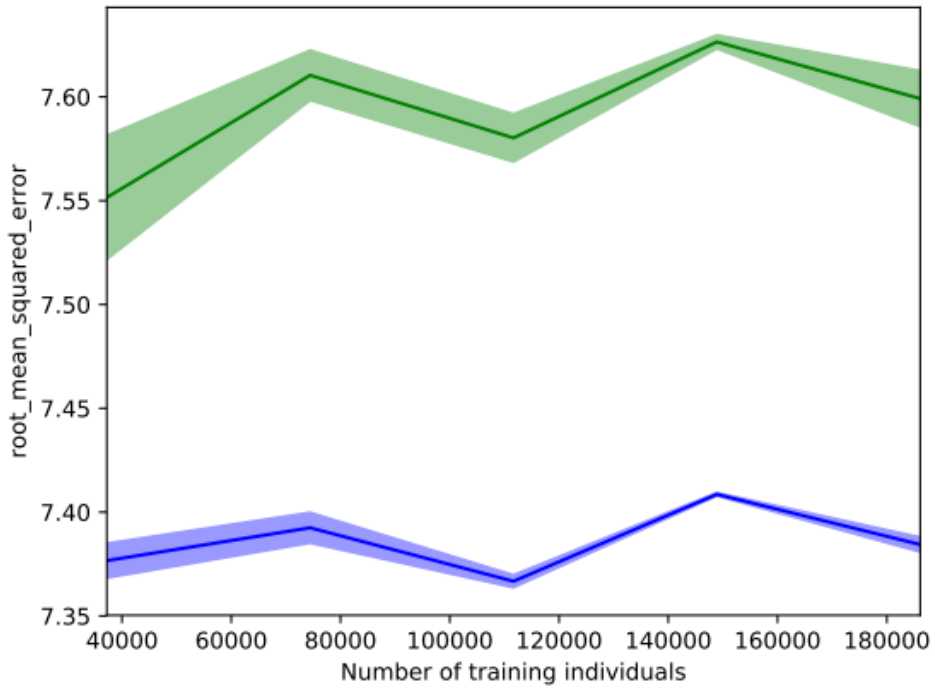
Performance stability across different priors

For the results presented in the main text, we utilize a prior $u_0 = [\kappa_0 = 180, \gamma_0 = 6, \alpha_0 = 2, \beta_0 = 20]$, from which we draw our initial $\theta = [\lambda, \pi]$. This is informed by expert knowledge about average cycle length (around 30 days) and the likelihood of skipping (relatively low) in our dataset.

In order to assess the impact of the prior, we also test training the model on different ones, namely a uniform prior on π (no prior knowledge on skipping likelihood), as well as a less informative (i.e., flatter) prior on both λ and π. We showcase the prediction RMSE results on day 0 of the next cycle for both priors in **Supplementary Figure 3** and **Supplementary Figure 4**, where the blue line represents results for $s \geq 0$ and the green line represents results for $s = 0$. Note that these results look similar in magnitude and spread as the prior we have chosen, and we therefore conclude that our method is stable to different choices of priors.



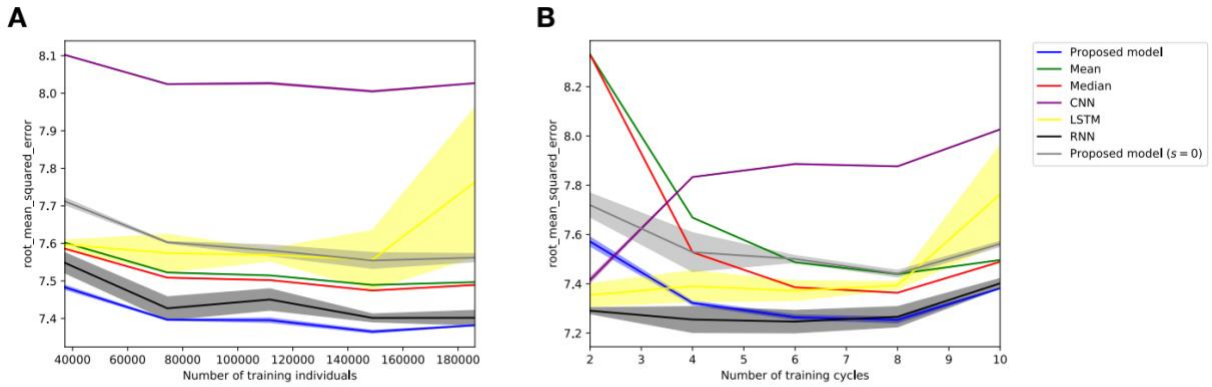
Supplementary Figure 3: Prediction RMSE over number of training individuals for a less informative (i.e., a more uncertain) prior on λ and π, $u_0 = [60,2,0.01,0.1]$.



Supplementary Figure 4: Prediction RMSE over number of training individuals for a less informative prior on λ and a completely uniformative (i.e., uniform) one on π, $u_0 = [60,2,1,1]$.

Performance stability across different dataset sizes and ordering of cycles

To demonstrate our model's robustness across different dataset sizes, we showcase prediction RMSE results across different numbers of individuals, I (left) and training cycles, C (right) in **Supplementary Figure 5**. We see that our model performance is robust to different I and C values – our model's prediction RMSE remains around 7.5 even with relatively small I or C.

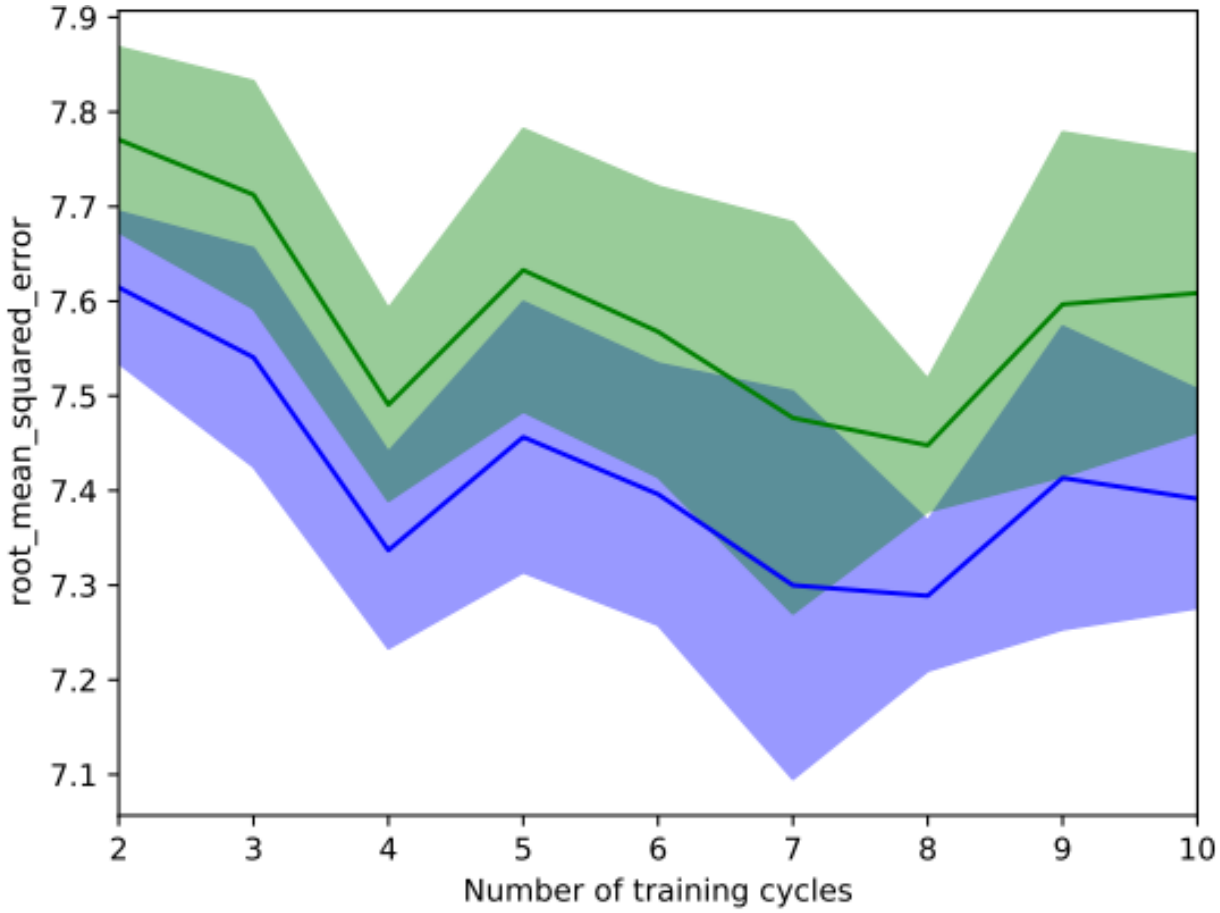


Supplementary Figure 5: Prediction RMSE for proposed model and baselines on day 0 over number of individuals, I **(a)** and number of training cycles, C (on the full set of I) **(b)**. $C = 2$ means 2 input cycles were used to predict the third and so on. **(a)** Our model outperforms summary statistic-based and neural network-based baselines on day 0 when we account for skipped cycles (blue line), across all subsets of I. In addition, our model produces sharper estimates (lower variance) and is stable across I – with less than 40,000 users, we have an RMSE less than 7.5. **(b)** Our model is robust to different C, as shown by consistent RMSE with at least 4 training cycles. Note that all models experience some fluctuations in RMSE depending on number of training cycles; this is due to data randomness, see **Supplementary Figure 6.**

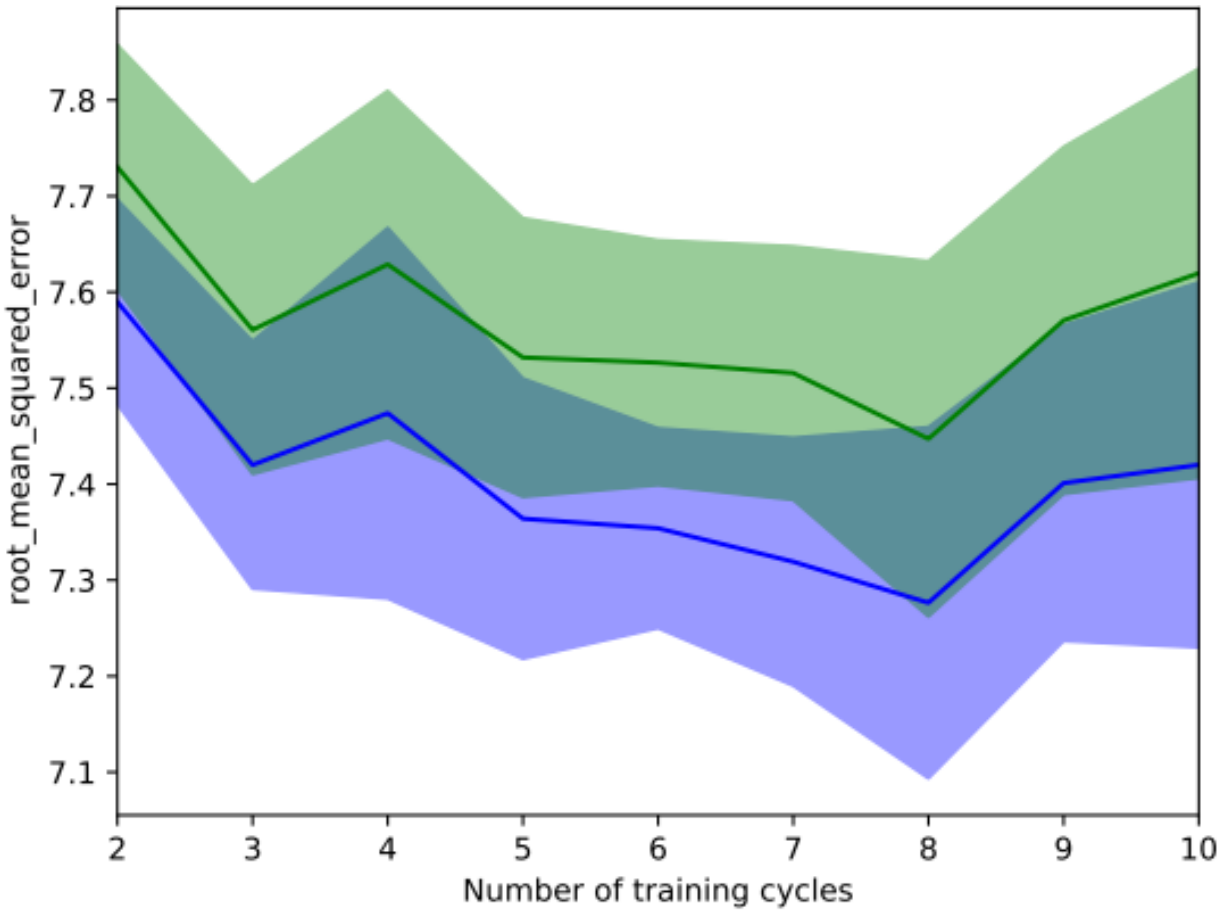
While our model performance is generally stable to dataset size as in **Supplementary Figure 5**, we note also that there is some very small magnitude fluctuation in performance with $C = 10$.

This is due to data randomness – that is, since we utilize the first C cycles in each training subset, there may be users who happened to have less adherent tracking near the end of their tracking history (i.e., with $C = 10$), resulting in a small uptick in prediction RMSE. To showcase this, we perform an experiment utilizing $I = 10{,}000$ users across 10 runs of our model; for each run, we randomly draw $I = 10{,}000$ users from the full dataset, train our model, and compute predictions. The results of this experiment averaged over the 10 runs are shown in **Supplementary Figure 6**, where we see that there is some fluctuation in prediction RMSE across C (not just for $C = 10$), verifying that the small fluctuation for $C = 10$ on the full dataset is an artifact of data randomness.

To further test the dependency of model predictive performance on the ordering of the observed training cycles, we also run the same experiment with a random shuffling of a user's cycle history before selecting the first C cycles for training. We showcase these results in **Supplementary Figure 7** and see again that there are small fluctuations in performance across C, verifying further the impact of data randomness. This also showcases the negligible effect of choosing to either take the first C cycles without shuffling (as in **Supplementary Figure 6**) or with shuffling (as in **Supplementary Figure 7**).



Supplementary Figure 6: Prediction RMSE over number of training cycles, averaged over 10 runs of different randomly-drawn datasets of $I = 10{,}000$ users.

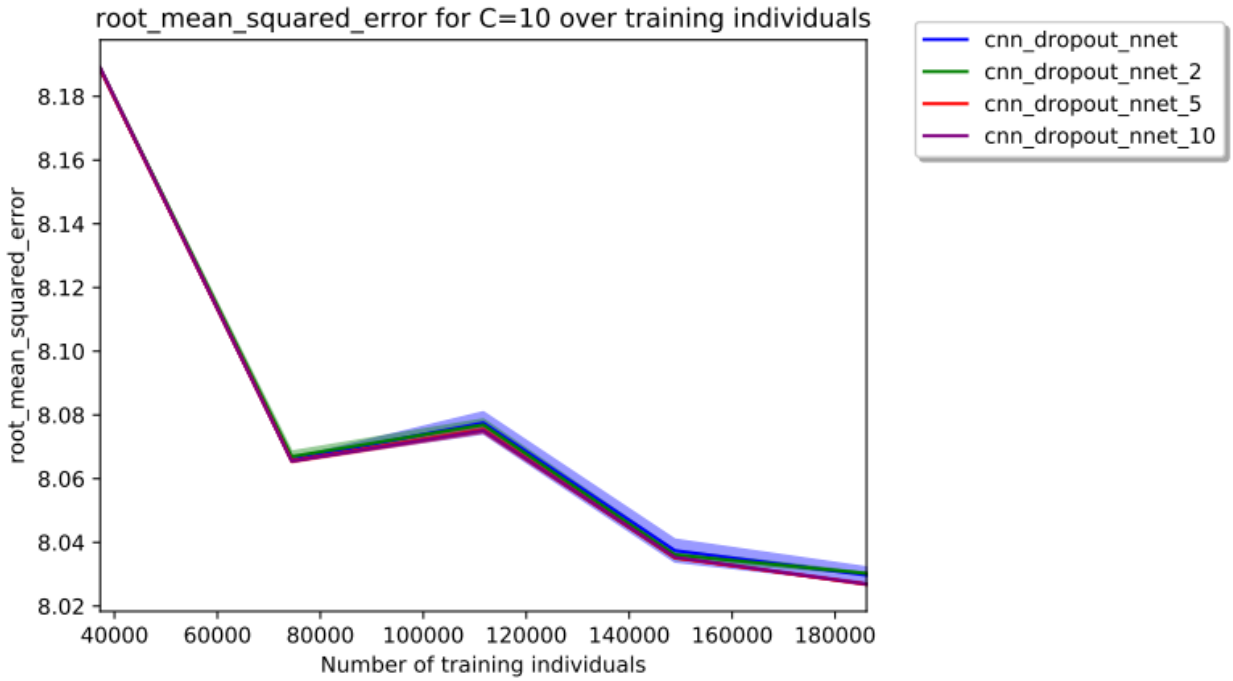


Supplementary Figure 7: Prediction RMSE over number of training cycles, averaged over 10 runs of different randomly-drawn datasets of $I = 10{,}000$ users. Here, before we take the first C cycles from each user, we randomly shuffle them.

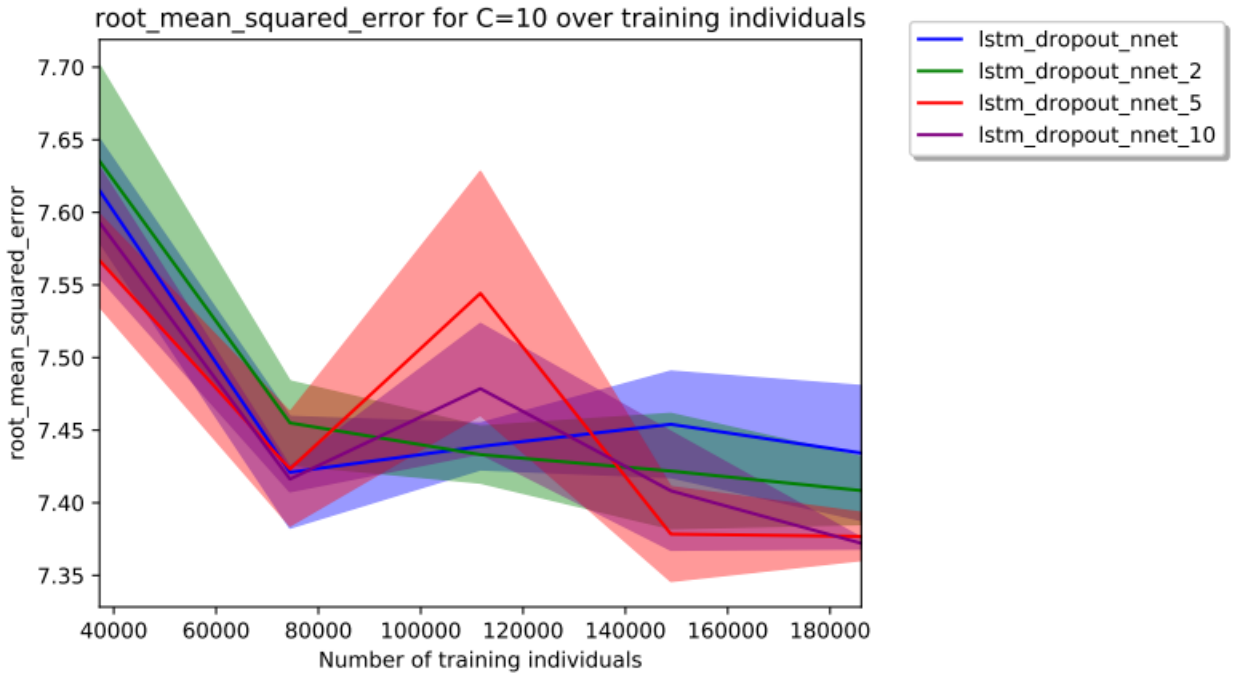
Baseline results with different neural network settings

In the results of our main manuscript, we utilize neural network-based baselines with one layer and a kernel size or hidden size of 3.

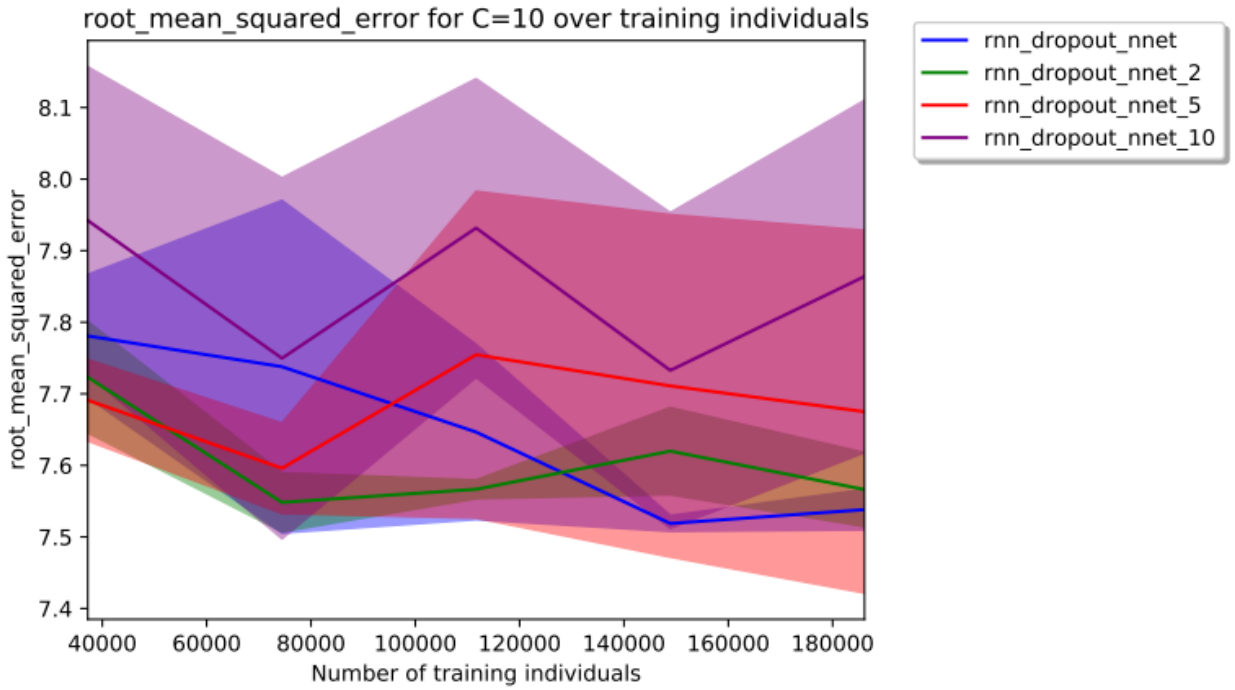
To assess the performance of neural network-based baselines with different settings, we test (i) different numbers of layers and (ii) different kernel and hidden sizes (using a kernel or hidden size equal to the number of training cycles C instead of fixed at 3). **Supplementary Figure 8, Supplementary Figure 9,** and **Supplementary Figure 10** showcase the performance RMSEs across I for 1, 2, 5, and 10-layer CNNs, LSTMs, and RNNs, respectively (with fixed kernel or hidden size of 3).



Supplementary Figure 8: Prediction RMSE over number of individuals for CNNs with 1, 2, 5, and 10 layers (blue, green, red, and purple lines, respectively) and a kernel size of 3.

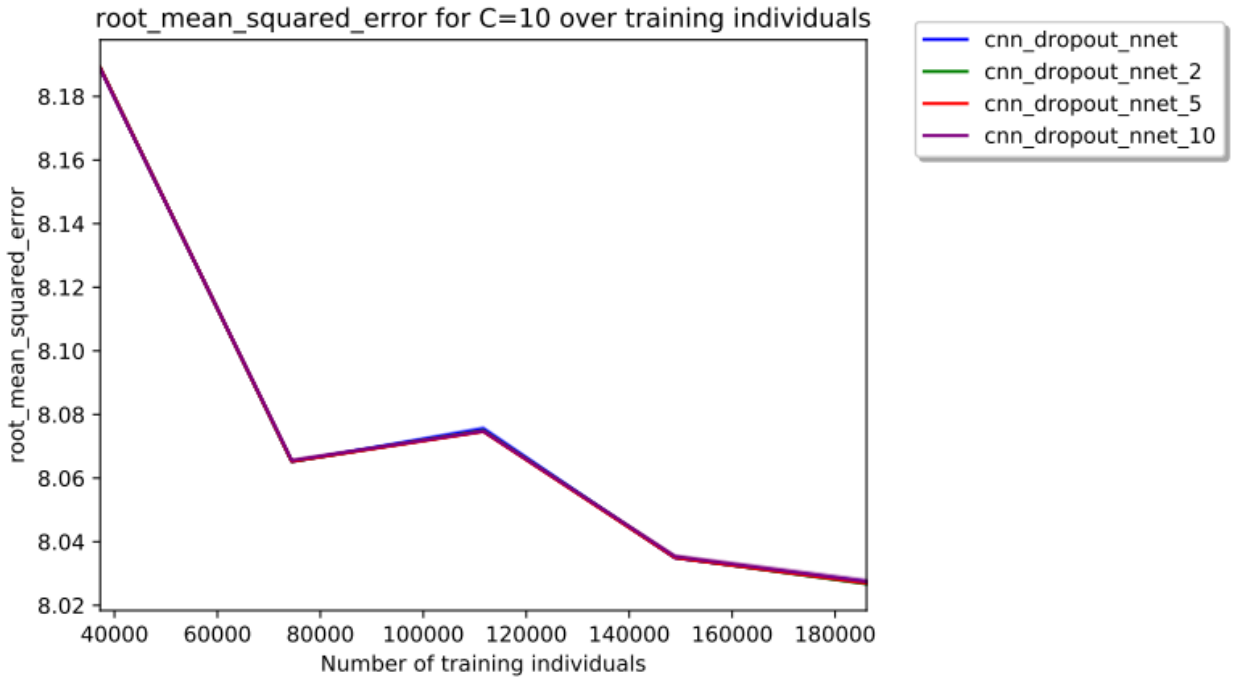


Supplementary Figure 9: Prediction RMSE over number of individuals for LSTMs with 1, 2, 5, and 10 layers (blue, green, red, and purple lines, respectively) and a hidden size of 3.

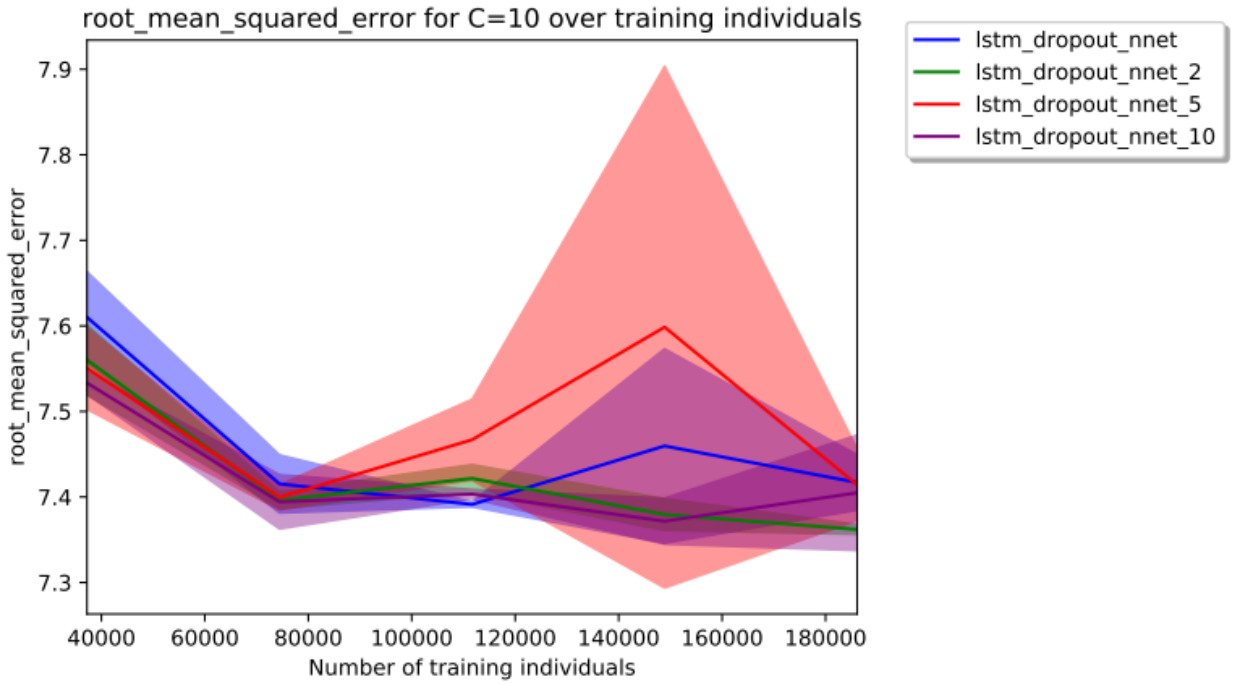


Supplementary Figure 10: Prediction RMSE over number of individuals for RNNs with 1, 2, 5, and 10 layers (blue, green, red, and purple lines, respectively) and a hidden size of 3.

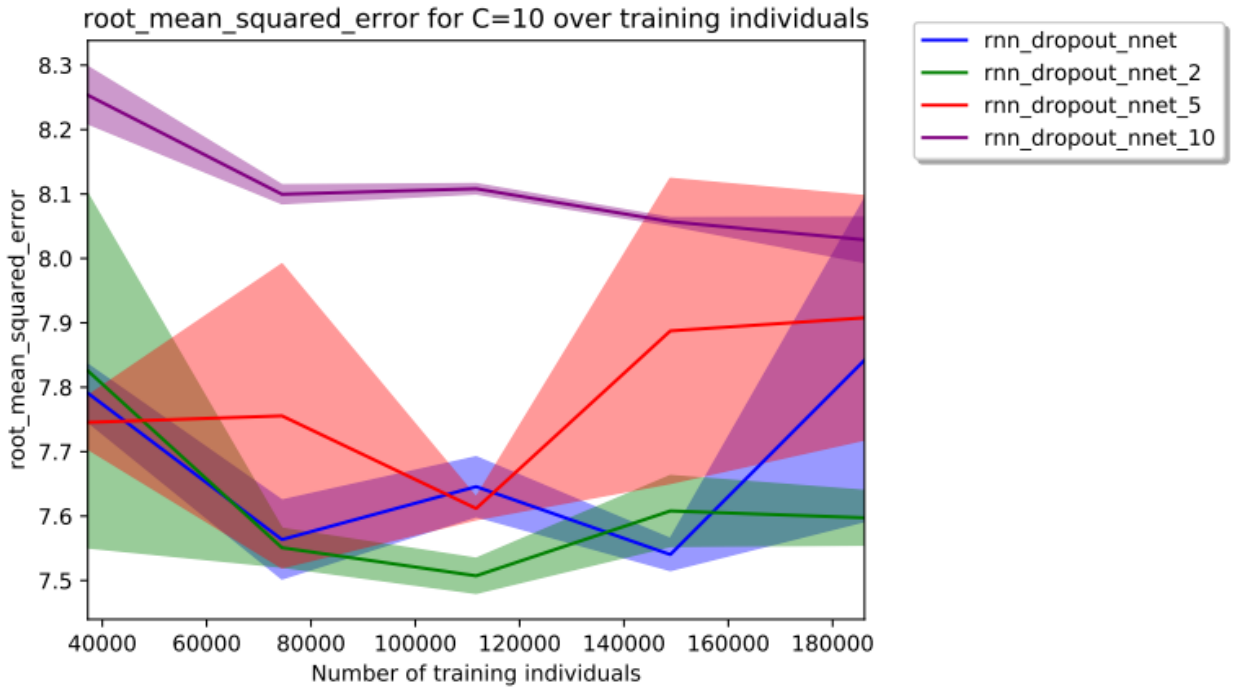
Supplementary Figure 11, Supplementary Figure 12, and **Supplementary Figure 13** showcase the performance RMSEs across I for 1, 2, 5, and 10-layer CNNs, LSTMs, and RNNs, respectively (with kernel or hidden size of $C = 10$). We see that across the number of layers and kernel or hidden size of 3 or $C = 10$, the prediction RMSE is stable, with average differences of at most 0.5 between different settings. Therefore, we conclude that one-layer neural networks, with fixed kernel or hidden size of 3, are reasonable baselines.



Supplementary Figure 11: Prediction RMSE over number of individuals for LSTMs with 1, 2, 5, and 10 layers (blue, green, red, and purple lines, respectively) and a hidden size of $C = 10$.



Supplementary Figure 12: Prediction RMSE over number of individuals for LSTMs with 1, 2, 5, and 10 layers (blue, green, red, and purple lines, respectively) and a hidden size of $C = 10$.



Supplementary Figure 13: Prediction RMSE over number of individuals for RNNs with 1, 2, 5, and 10 layers (blue, green, red, and purple lines, respectively) and a hidden size of $C = 10$.

REFERENCES

[1] D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*. 2017.