HeartSteps V2 Anti-Sedentary Algorithm

Walter Dempsey

January 28, 2019

Abstract

This note describes the anti-sedentary algorithm to be implemented in HeartSteps V2. We compare the proposal to several competitors and present performance on the HeartSteps V1 dataset.

1 HeartSteps V2: Anti-sedentary messages

In *HeartSteps V2*, patients with blood pressure in the stage 1 hypertension range (120-130 systolic) will be provided a wrist tracker and a mobile phone application intended to help them maintain recommended levels of physical activity over the long term. In this study, one of the treatment components is an anti-sedentary message. Here, the scientific team aims to provide an average of 1.5 anti-sedentary messages per day at sedentary times. The goal of this note is to detail the algorithm—called the "HeartSteps V2 Anti-Sedentary" (HeartVerAS) algorithm—used to achieve this average constraint. We introduce competitor methods and assess comparative performance in simulations.

The proposed HeartVerAS algorithm is a specific instance of the "sequential risk time sampling" (SeqRTS) algorithm built specifically for HeartSteps V2 using a previous physical activity study. We introduce competitor SeqRTS and "block sampling" (BlockS) algorithms to assess comparative performance. All algorithms have tuning parameters, and rely on constructing a risk prediction. SeqRTS algorithms require a prediction of the remaining number of sedentary times within the block. BlockS algorithms require a prediction of the number of sedentary times within each block.

For HeartSteps V2, decision times will be every 5 minutes within a 12-hour day. We let $\mathcal{T}_d$ denote the set of decision times for day d, i.e., $|\mathcal{T}_d| = 12 \cdot 12 = 144$. At decision time t, the binary risk factor X_t is set to 1 if the user has less than 150 steps within the prior 40 minutes; otherwise the user is classified as "Not Sedentary" ($X_t = 0$). The treatment A_t is an indicator of whether an anti-sedentary message is sent at decision time t (e.g., 1 = "Send Message" and 0 = "Provide Nothing"). We assume that each day is split into 3 blocks of 4 hours each. Note, blocking of the day into 3 4-hour blocks is used by **ALL** algorithms. That is, $\mathcal{T}_d = \cup_{k=1}^{3} \mathcal{B}_{d,k}$ where $\mathcal{B}_{d,k}$ denotes the kth block on day d. Denote the size of each block $|\mathcal{B}_k| = 144/3 = 48$. At a decision time t, the user is unavailable ($I_t = 0$) if either

1. the user received an anti-sedentary message within the past hour; or
2. the user has not yet received the good morning message.
3. [WD: Currently we ignore the activity suggestions in the formulation of availability. Pedja thought it would not be an issue to receive the anti-sedentary message in a

window around the activity message. Note, that the ability to achieve the average constraint and the simulation experiments will have to re-run and could get much more complicated if we include such a constraint as I'd need to combine with the Peng's activity suggestion bandit algorithm.]

If neither conditions hold, then the user is available at time t and $I_t = 1$. Two things to note: (1) unavailability is independent of whether the user receives an activity message in prior window of time; (2) the default time for the good morning message is 8AM. However, users may change this time in the preference settings screen. This only impacts availability by restricting randomizations to the remaining part of the day. For instance, if the user sets the good morning message to 10AM, then the user is only available from 10AM to 8PM. That is, the user day is cut short by 1 hour.

In this note, we use data from a previous physical activity study [1] (here on called "HeartSteps V1 dataset") to design HeartSteps V2. HeartSteps V1 is a mobile health intervention focused on promoting physical activity. We restrict to the person-days for which there is at least one five-minute window with step count. This avoids using days in which the user may not be wearing the wrist sensor. We restrict to 9AM to 9PM as this range had the most reliable step count data. This results in 1543 12-hour person-days. We use five-fold cross validation to demonstrate the method. In particular, we randomly partition the participants into five non-overlapping sets of roughly equal size. In each of five iterations, a single subset (i.e., "test") is extracted and the remaining data for participants in the four other subsets of HeartSteps V1 are used as the training set. The training set is used to build the forecasts for each algorithm and choose the tuning parameters.

2 The Sequential Risk Time Sampling Algorithm

All SeqRTS algorithms considered here proceed as follows: at the decision time t in the k-th time block on day d, assuming the user is available (i.e., $I_t = 1$) and currently sedentary (i.e., $X_t = 1$), we deliver the treatment with probability π_t given by

$$\pi_t = \phi\left(\frac{N_1 - \sum_{s \in \mathcal{B}_{d,k}, s \leq t-1} \mathbb{1}_{\{I_s = 1, X_s = x\}} \pi_s}{1 + g(1|H_t)}\right) \tag{2.1}$$

Recall the 12-hour day is divided into 3 4-hour blocks.

1. $\phi(x) = x\mathbb{1}_{\{x \in [\epsilon, \epsilon']\}} + \epsilon \mathbb{1}_{\{x \leq \epsilon\}} + \epsilon' \mathbb{1}_{\{x \geq 1-\epsilon'\}}$ is a truncation operation to the interval $[\epsilon, \epsilon']$ such that $0 < \epsilon < \epsilon' < 1$ to ensure the output value stays within $[0, 1]$. Here we set $\epsilon = 0.005$ and $\epsilon' = 0.2$. Figure 1 plots the fraction of time sedentary within and after each hour in the day. This means that at 10AM, for example, we compute the fraction of time sedentary from 10AM until to 9PM. This is computed every hour from 9AM to 9PM (e.g., 9AM, 10AM, . . ., 8PM, 9PM). We compute this fraction as we are interested in prediction for the remainder of the day, *not* in the remainder of the hour. We see this plot is relatively flat between 0.50 at the beginning of the day and 0.55 by the final hour. Therefore, randomization probability of 0.2 (ignoring availability constraints) would yield approximately on average $144 \times 0.525 \times 0.2 = 15.12$ treatments. This is a sufficiently high upper bound to avoid undertreating. The choice was to avoid border cases where there are only a few sedentary times at the very end of the day. In this case, we want to avoid sending with very, very high probability.

Sending with chance 0.2 means that even if there are only 10 sedentary times in the day, we will likely achieve the average constraint.

2. $g(1|H_t)$ is a forecast, based on the observation history H_t, of the number of remaining available decision times at which the participant is sedentary in the current time block. So the forecast is a prediction of $\sum_{s \in \mathcal{B}_k, s \geq t+1} \mathbb{1}_{\{I_s=1, X_s=1\}}$.

3. N_1 controls the number of treatments in each block; here, the target average number of treatments for each block is set to $1.5/3 = 0.5$, and we denote this by N_1^*.

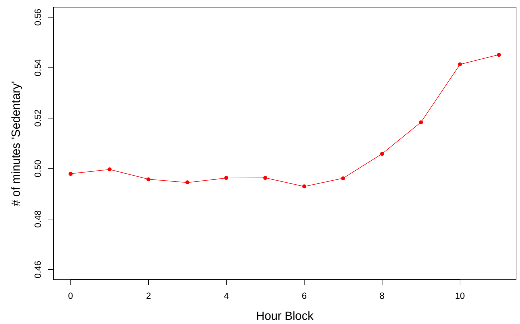

Figure 1: Fraction of time sedentary within and after current hour

Each SeqRTS algorithm requires the construction of the forecasts $(g(1|H_t))$ and the tuning parameter N_1. Here we detail how we use a training set from HeartSteps V1 to tune N_1 *for a pre-specified forecast*. To tune the parameter N_1, we use (2.1) to determine the probability π_t to generate an A_t at each available risk time. This is done for each block in each day in the training data. We then compute the average number of treatments per day. We search for the optimal tuning value of N_1 such that the computed average number of treatments within the day is close to the target constraint $3 \cdot N_1^*$. We implement two SeqRTS algorithms:

- **Baseline**: a nonparametric estimator based on the concept of a "run". When $X_t = 1$, it is likely that $X_{t+1} = 1$ since the prior 40-minute step count is used to define X_{t+1}. Since X_t and X_{t+1} have 35-minute overlap if $X_t = 1$ then X_{t+1} is likely to be 1 as well. Specifically, we fit a non-parametric estimator of the forecasting model using the training set that is a function of run length and the current hour. **Based on the simulation results, we argue that this should be the implemented algorithm for HeartSteps V2.** We now outline the math behind this estimator:

  Technical details: Given the current run length k, define K to be the remaining amount of time in the current state (so $R = K + k$ is the total run length). We then define F to be the fraction of time in the state "Sedentary" remaining in the day. Then the amount of remaining time the participant is classified as "Sedentary" is given by $\Omega = K \wedge r + F(r - K)_+$, where $\wedge$ denotes minimum operator and $(x)_+ = \max(x, 0)$ and r denotes the amount of time remaining in the current day. That is, we account for the fact that the expected run length may be greater than the remainder of the day. We assume that F is independent of both functionals of K given k and X_s. The conditional expectation can then be written as

$$\mathbb{E}[\Omega \,|\, k, X_s] = \mathbb{E}[K \wedge r \,|\, k, X_s] + \mathbb{E}[F \,|\, k, X_s] \cdot \mathbb{E}[(r - K)_+ \,|\, k, X_s].$$

Below we assume $X_s = 1$ as we only intervene when the individual is classified as "Sedentary". Each component of the conditional expectation is estimated using the Heartsteps V1 data. Figure 2 is a histogram of run lengths in state "Sedentary" and "Not Sedentary" respectively.[1] Let $\{\tilde{R}_i\}_{i=1}^m$ denote the data of run lengths used to compute the "Sedentary" plot within Figure 2. Then define $\tilde{K}_i = \tilde{R}_i - k$. Using only runs such that $\tilde{K}_i \geq 0$, we have the following estimates

$$\mathbb{E}[K \wedge r \,|\, k, X_s] \approx \frac{1}{m_k} \sum_{j=1}^{m_k} \tilde{K}_j \wedge r$$

$$\mathbb{E}[(r - K)_+ \,|\, k, X_s] \approx \frac{1}{m_k} \sum_{j=1}^{m_k} (r - \tilde{K}_j)_+$$

where $m_k \leq m$ is the number of observations satisfying the condition $\tilde{R}_i \geq k$.

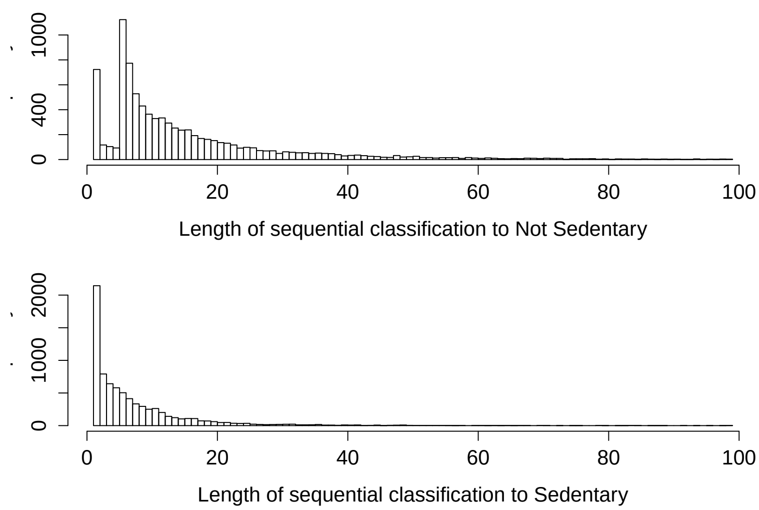

Figure 2: Run lengths of being classified as either "Not Sedentary" or "Sedentary" respectively.

We then estimate the fraction of time classified as Sedentary in the remaining window, $\mathbb{E}[F \,|\, k, X_s]$. Figure 1 plots the estimate calculated using the data from the current hour *and* all remaining hours. Here, since the figure shows this fraction is flat, we simply use the current hour estimate. For example, at time $9:15AM$, we use the estimated fraction in Figure 1 from $9:00AM$—i.e., the fraction of time sedentary from $9:00AM$ to $10:00PM$. At $9:45AM$, we use the same estimate; however, at

[1] We checked the Heartsteps V1 data for differences as functions of "time of day" and "length in study", but found very little to suggest a more complicated model was necessary.

$10:00AM$ and $10:10AM$, we use the estimated fraction at $10:00AM$—i.e., the fraction of time sedentary from $10:00AM$ to $10:00PM$.

- **Person-specific (P-S)**: an estimator that adds a person-specific random effect to the estimator of the fraction of time remaining sedentary, which personalizes predictions over time. We now outline the math behind this estimator.

  Technical details For each hour-window, we first compute the mean fraction of time sedentary in the remainder of the day given the current hour and user. Let $F_{u,d,h}$ denote this fraction for participant u on day d in hour h. We then fit a linear mixed-effects model to each hour with random intercept per hour to the training dataset. That is, let $b_{u,h} \sim N(0, \sigma^2_{h,b})$ and

  $$F_{u,d,h} \,|\, b_{u,h} \sim N(\beta_{0,h} + b_{u,h}, \sigma^2_{h,0}).$$

  We fit these models separately for each hour. Then, for each user in the test data, forecasts are then built by first estimating

  $$\hat{\mu}_{u,d+1,t} = \mathbb{E}[F_{u,d+1,h(t)} \,|\, \{F_{u,d',h(t)}\}_{d'<d}] = \hat{\beta}_{0,h(t)} + \hat{\sigma}^2_{h(t)} \mathbf{1}'_d \hat{\Sigma}^{-1}_{h(t),d} (\bar{\mathbf{F}}_{u,d,h(t)} - \hat{\beta}_{0,h(t)} \mathbf{1}_d)$$

  where $\hat{\Sigma}_{h(t),d} = \hat{\sigma}^2_{h(t),0} \cdot I_{d,d} + \hat{\sigma}^2_{h(t),b} \mathbf{1}_d \mathbf{1}_d^T$, $I_{d,d}$ is a $d \times d$ identity matrix, and $\mathbf{1}_d$ is a d-dimensional unit vector, $h(t)$ denotes the current hour for the tth decision time, and $\bar{\mathbf{F}}'_{u,d,h} = (F_{u,1,h}, \ldots, F_{u,d,h})$ is the vector of prior fractions. For the first day, the prediction is $\hat{\beta}_{0,h(t)}$. Then the prediction is $\hat{\mu}_{u,d+1,t} \cdot (T - t)$.

Both forecasts do not explicitly account for unavailability due to receiving treatment. Instead, the tuning parameter N_1 is adjusted to achieve the average constraint and to overcome potential errors in these forecasts.

3 Block-sampling algorithm

All block-sampling (BlockS) algorithms considered here proceed as follows: the average number of treatments per day in each block is $N_1^* = 0.5$. HeartSteps V1 data is used to estimate the number of expected risk times within each block, denoted by M_k for the k-th block. Then at each sedentary time within each block, a treatment is sent with probability N_1/M_k, where N_1 is a (potential) tuning parameter. Each BlockS algorithm requires the construction of the forecast $(M_k)_{k=1}^3$ of the expected number of sedentary times per block. We implement two BlockS algorithms:

1. **Baseline**: First, we set M_k equal to the mean number of sedentary times within each block across the training dataset. Then, we run a simulation on the training dataset using probability of treatment $N_1^\star/M_k$. From the simulation, we compute the average number of unavailable sedentary times per day, denoted $\hat{\Delta}$. We then re-define $M_k \leftarrow M_k - \hat{\Delta}$; that is, M_k is now set equal to the mean number of *available* sedentary times within each block. We then use this to form the randomization probability; that is, the probability of treatment at available, sedentary times is $N_1^\star/M_k$.

2. **Person-Specific**: an estimator that adds a person-specific random effect to the estimator of the fraction of time sedentary in each block, which personalizes predictions over time. We now outline the math behind this estimator.

Technical details:

Let $M_{u,d,b}$ denote the number of sedentary decision times for participant u on day d in block B. We fit a linear mixed effects model with intercept $\beta_{0,B}$ and random intercept per user-block. That is, $b_{u,B} \sim N(0, \sigma^2_{B,b})$ and

$$M_{u,d,b} \,|\, b_{u,B} \sim N(\beta_0 + b_{u,B}, \sigma^2_{B,b})$$

For each user in the test data, forecasts are then built by first estimating

$$\hat{\mu}_{u,d+1,B} = \mathbb{E}[M_{u,d+1,B)} \,|\, \{M_{u,d',B)}\}_{d'<d}] = \hat{\beta}_{0,B} + \hat{\sigma}^2_{B,b}\mathbf{1}'_d \hat{\Sigma}^{-1}_{B,d}(\bar{\mathbf{M}}_{u,d,B} - \hat{\beta}_{0,B}\mathbf{1}_d)$$

where $\hat{\Sigma}_{B,d} = \hat{\sigma}^2_{B,0} \cdot I_{d,d} + \hat{\sigma}^2_{B,b}\mathbf{1}_d\mathbf{1}_d^T$, $I_{d,d}$ is a $d \times d$ identity matrix, and $\mathbf{1}_d$ is a d-dimensional unit vector, and $\bar{\mathbf{M}}'_{u,d,B} = (M_{u,1,B}, \ldots, M_{u,d,B})$ is the vector of prior values. For the first day, the prediction is $\hat{\beta}_{0,B)}$. Then the prediction is $\hat{\mu}_{u,d+1,B}$.

We then run a simulation on the training dataset using probability of treatment $N_1/\hat{\mu}_{u,d,B}$ for user u on day d for block B. We then compute a global offset $\hat{\Delta}$, set equal the mean number of *unavailable* sedentary times within the simulation. The prediction is then the person-specific $M_{u,d+1,B} = \hat{\mu}_{u,d+1,B} - \hat{\Delta}$. The tuning parameter N_1 is set by running a simulation and computing the average number of treatments per day. We choose N_1 so this computation is closest to the target 1.5. Here, $N_1 = 0.53$.

4 Comparative performance

To compare all algorithms, we use three metrics. The first two metrics are graphical and the final metric uses a divergence function to assess divergence from uniform sampling of risk times for treatment. Recall that a training set is used to build the forecasts method as well as the select tuning parameters. Our three metrics will be evaluated on a test data set. Given the forecasts as well as the selected tuning parameters, we use the test data to generate $1,000$ treatment sequences (sequences of A_t's) per user-day. Treatment sequences A_t are generated by SeqRTS or BlockS algorithms.

In the first graphical metric, we use the 1000 generated treatment sequences to compute the total number of treatments provided per user-day. We then take the average over these treatment sequences to compute the average number of treatments per user-day. We then produce a box-plot across all user-day combinations of this average number of treatments. We expect the mean and median to agree with desired number of treatments per day, $3 \cdot N_1^* = 1.5$. We prefer methods with low variability around this mean.

In the second graphical metric, we use the same 1000 generated treatment sequences to compute the total number of treatments provided per user-day. We then take the average over days to compute an average total number of treatments provided per user. We then produce a box-plot across all users. We again expect the mean and median to agree with desired number of treatments per day, $3 \cdot N_1^* = 1.5$. We prefer methods with low variability around this mean.

To construct the final metric, we again use the $1,000$ treatment sequences and compute the fraction of time treated per decision point. That is, for day d, we compute the fraction of times the user received treatment at sedentary times per decision point. As treatment can only be provided at sedentary times, we extract these times out to construct the vector $\hat{p}_{u,d}$ (i.e., fraction of time treated for user u on day d at sedentary times). Here, we ignore availability. We believe this may be why the results are not consistent with the boxplots constructed from the other graphical metrics. Let $N_{u,d}$ be the number of sedentary times for user u on day d. A goal is uniformity across risk times. We measure this by the average Kullback-Leibler divergence from the *target quantity* $N_x/N_{u,d}$. That is,

$$\frac{1}{N_{u,d}} \sum_{i=1}^{N_{u,d}} KL\left(\hat{p}_{u,d,i}, N_1^*/N_{u,d}\right)$$

where $\hat{p}_{u,d,i}$ is the ith component of the vector $\hat{p}_{u,d}$, and $KL(q,p) = q\log(p/q) + (1-q)\log(1-p)/(1-q)$ for $p,q \in (0,1)$. Ignoring availability, uniform treatment randomization that achieves the average constraint will send an intervention at a sedentary time with probability $1.5/N_{u,d}$. The term $\hat{p}_{u,d,i}$ is the computed fraction of simulations where treatment was provided at the particular sedentary time. The KL-divergence is a distance measure on this probability. We are computing the average across all sedentary times. We then produce a box-plot of this quantity across user-days (all (u,d)'s). Smaller Kullback-Leiber divergence indicates that the sampling is closer to uniform sampling and low overall variability indicates the sampling of risk times to close to uniform sampling for all person-days.

4.1 Results

Table 1 and Figure 3a summarize the average treatment results per user-day for each method. We see minimal improvement through personalization. We replicate the table and plot for results aggregated per user; see Table 2 and Figure 3b. Figure 3c presents the Kullback-Leibler divergence per method. Overall, we see that BlockS algorithms achieve slightly better uniformity performance, but suffer from larger variance in the average number of treatments provided. **Therefore, we recommend *Baseline SeqRTS* as the HeartVerAS algorithm.** The method is easy to implement, stable, and personalizes enough to avoid potential over/under-treatment.

Method	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
Baseline (BlockS)	0.00	1.07	1.54	1.56	1.82	2.80
Person-Specific (BlockS)	0.00	1.21	1.54	1.56	1.86	5.45
Baseline (SeqRTS)	0.00	1.30	1.51	1.51	1.79	2.39
Person-Specific (SeqRTS)	0.00	1.38	1.63	1.62	1.88	2.50

Table 1: Five-fold cross validation results of Block Sampling and Sequential Risk Times Sampling (SeqRTS) algorithms: average number of treatments at sedentary times achieved by each person-day in 1,000 runs.

5 Pseudo-code for implemented SeqRTS algorithm in HeartSteps V2

Here we detail the algorithm used for HeartSteps V2. We will use the Baseline forecast and the N_1 trained via cross-validation on HeartSteps V1. We will test on the first 10 participants of HeartSteps V2 to ensure N_1 is properly tuned to achieve the average constraint. If the average number of treatments across these participant-days is more than 0.1 away from 1.5,

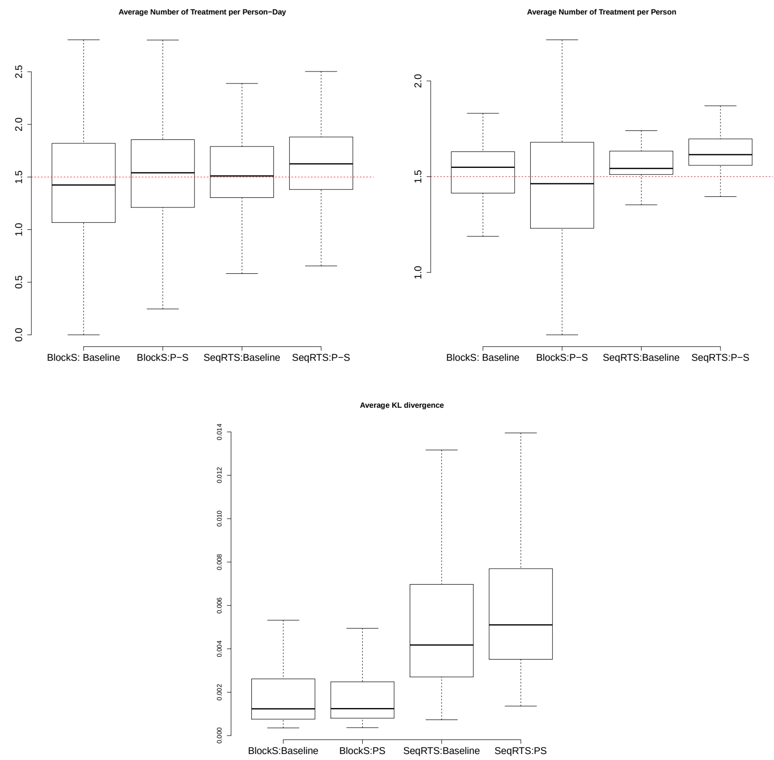

Figure 3: Five-fold cross validation results of Block Sampling and Sequential Risk Times Sampling (SeqRTS) algorithms. The average number of treatments (per person-day and per person) and the average KL divergence at sedentary times by each person-day in 1,000 runs.

Method	Min	1st Qu.	Median	Mean	3rd Qu.	Max.
Baseline (BlockS)	0.99	1.41	1.55	1.51	1.63	1.83
Person-Specific (BlockS)	0.67	1.23	1.46	1.49	1.67	2.21
Baseline (SeqRTS)	1.35	1.51	1.54	1.56	1.63	1.74
Person-Specific (SeqRTS)	1.40	1.56	1.62	1.62	1.70	1.87

Table 2: Five-fold cross validation results of Block Sampling and Sequential Risk Times Sampling (SeqRTS) algorithms: average number of treatments at sedentary times achieved by each person (averaged over person-days) in 1,000 runs.

then we will need to adjust N_1. If the deviation is below 0.2 then we will perform additional cross-validation that includes these 10 participants as a single block. If the deviation is above 0.2 then we will only use participant-days from these 10 participants when tuning N_1 and perform leave-one-out cross-validation.

We will define each user-day to be 12-hours, starting from 9AM to 9PM. To avoid issues regarding time zones, we will define the 12-hour day based on location at 9AM. If the user crosses time zones within the day, then 9PM for that day is defined at 9PM relative the geographic location at 9AM. For example, suppose a user is in Ann Arbor (EST) at 8AM and then travels to Chicago (CST). Then the end of day is 9PM EST and so the day will end at 8PM CST. The total length of day, however, will still be 12 hours. The user is unavailable for treatment if either (1) the user has received a message in the past sixy minutes or (2) the user has yet receive the good morning message from the app. We will divide the 12-hour day into 3 blocks of 4 hours each (i.e., this defines $B_{d,k}$ for the microservice). These blocks are defined relative to location at 9AM. If the user wears the fitbit outside the 12 hour range, any POST request will return $A_t = 0$ and $\pi_t = 0$ and $I_t = 0$. That is, the user will be defined as unavailable outside the 12-hour window.

At the beginning of each five minute on each participant-day in the study, there are 3 main steps. First, the main server performs a POST request in JSON format to the microservice. Second, the microservice performs some internal calculations, storing some of these calculations in a local database. Finally, the microservice returns a JSON output. We present the algorithm at a particular moment within a participant-day.

1. Every 5 minutes, the main server will ping the anti-sedentary micro-service.
2. The ping will be a POST request with the following information in JSON format
    - Unique participant identifer
    - Timestamp in military time (e.g., 10/20/2018 14:05)
    - 5-minute window step count
    - Sedentary status: three levels ("Sedentary", "Not Sedentary", "Unknown")
    - End-of-day and beginning-of-day in military time
    - Availability as determined by having sent good morning message or not
3. The microservice accepts the POST request and performs the following:
    - If sedentary status is "Not Sedentary" or "Unknown" OR availability is FALSE then return $I_t = 0$, $A_t = 0$, and $\pi_t = 0$
        - The "Unknown" status captures any syncing issues with the Fitbit wristband.
    - Otherwise, read from local database the history H_t for this participant on this day.
    - Using H_t, compute $g(1 \mid H_t)$.
    - Using H_t, compute $\sum_{s \in \mathcal{B}_{d,k}, s \leq t-1} \mathbb{1}_{\{I_s=1, X_s=x\}} \pi_s$
    - Compute π_t using above 2 inputs and equation 2.1
        - Note, if a prior POST request did not occur at some time $t' < t$ in this participant-day, then this is missing in the history. This is treated as $I_{t'} = 0$, $A_{t'} = 0$, $\pi_{t'} = 0$ and so does not affect the forecast nor the randomization probabilities.
        - For these times, we impute sedentary status to be "Unknown". This affects the definition of run length used in the forecast (see section 2). In particular, we define the run length as the number of prior times classified as "Sedentary". Therefore, having "Unknown" status breaks a sedentary run.

- We will use the first 10 participants to ensure the distribution of run lengths does not change substantially when accounting for the "Unknown" status. If it does, we will use the new data to alter the non-parametric estimate (again see section 2).

- Compute $A_t \sim \text{Bern}(\pi_t)$

4. The microservice will write to the local database for that participant a new row with the following information
   - Unique participant identifer
   - Timestamp in military time (e.g., 10/20/2018 14:05)
   - 5-minute window step count
   - Sedentary status: three levels ("Sedentary", "Not Sedentary", "Unknown")
   - Availability: I_t
   - Randomization probability: π_t
   - Treatment indicator: A_t

5. The microservice will then return the following in JSON format
   - Unique participant identifer
   - Timestamp in military time (e.g., t =10/20/2018 14:05)
   - Sedentary status: three levels ("Sedentary", "Not Sedentary", "Unknown")
   - Probability of treatment ($\pi_t \in (0, 1)$)
   - Treatment indicator ($A_t \in \{0, 1\}$)

At night, a "cleaner" algorithm will perform a participant-day sweep. The main server will perform a series of POST requests for each 5 minute window during the day with the following information:

- Unique participant identifer
- Timestamp in military time (e.g., t =10/20/2018 14:05)
- "True" sedentary status based on Fitbit cloud data: three levels ("Sedentary", "Not Sedentary", "Unknown")
- "True" 5 minute step-count based on Fitbit cloud data

The microservice will append two new columns. One that contains corrected 5-minute step count and one that contains the corrected sedentary status. This way we will maintain both the "online" data and the "batch" data for future analysis. If the main server failed during the day to perform a POST request at a particular 5 minute window, this cleaning procedure will add that row with $I_t = 0$, $A_t = 0$, $\pi_t = 0$. A binary indicator column with name "Sync Issue" will be added to clarify when this occurred during the day.

References

[1] Predrag Klasnja, Eric B Hekler, Saul Shiffman, Audrey Boruvka, Daniel Almirall, Ambuj Tewari, and Susan A Murphy. Microrandomized trials: An experimental design for developing just-in-time adaptive interventions. *Health Psychology*, 34(S):1220, 2015.