JOURNAL

Identifying Irregular Activity Sequences: an Application to Passive Household Monitoring

Jess Gillam[1] | **Rebecca Killick**[2] | **Simon Taylor**[2,3] |

Jack Heal[4] | **Ben Norwood**[4]

[1]STOR-i Doctoral Training Centre,
Lancaster University

[2]Department of Mathematics and Statistics,
Lancaster University

[3]School of Mathematics, University of
Edinburgh

[4]Howz, Greenheys Business Centre,
Pencroft Way, Manchester

Correspondence
Jess Gillam, STOR-i Doctoral Training
Centre, Lancaster University
Email: j.gillam@lancaster.ac.uk

Funding information

Approximately 1 in 5 people will live to see their 100^{th} birthday due to advancements in modern medicine and other factors. Over 65's constitute 42% of elective admissions and 43% of emergency admissions to hospital. Increasingly, people are turning to technology to help improve health and care of the elderly. There is mixed evidence of the success of wearables in older populations with a key barrier being adoption. In contrast, passive sensors such as infra-red motion and plug sensors have had more success. These passive sensors give us a sequence of categorical "trigger" events throughout the day. This paper proposes a method for detecting subtle changes in sequences whilst taking account of the natural day-to-day variability and differing numbers of "trigger" events per day.

KEYWORDS

categorical data, routines, home sensing

1 | INTRODUCTION

Approximately 1 in 5 people will live to see their 100^{th} birthday due to advancements in modern medicine and other factors (Age UK, 2019). While this is a great achievement, it puts pressure upon the NHS and health care services due to the care that is necessary with an ageing population. As of April 2019, there are just under 12 million people in the UK aged 65 or over (Age UK, 2019), with close to a quarter of those living by themselves. This group constitute 42% of elective admissions and 43% of emergency admissions to hospital (Age UK, 2018). It is clear that there is a growing need to help care for the elderly population as such pressures on the NHS and health care services will only increase in the future.

Increasingly, people are turning to technology to help improve health and care of the elderly. Howz is a company who uses unobtrusive passive sensors to ascertain patterns of behaviour and seeks to alert customers to help them to understand changes in their behaviours. Taylor et al. (2021) investigate changes in raw activity levels of Howz customers but do not consider the more detailed information collected about what activity is taking place. The larger changes in a persons routine can often be captured by changes in activity levels, but the more subtle changes in a loss of confidence or independence in living can be harder to discern. For example, a customer may routinely prepare hot lunch using the cooker but then switches to cold lunches as they become less confident in cooking. The activity levels will be the same but the nature of the task has changed and gives an indication of a decline in independence.

To identify these subtle changes we propose to consider a persons routine as their sequence of activated sensors per day and we wish to identify subtle changes in these routines. A significant challenge here is that a persons daily routine may have considerable variation and a varying number of activated sensors each day. Any method developed needs to be able to address these challenges.

1.1 | Howz Dataset

When considering activity data it is common to think of a wearable device that samples at 100-500 observations per second, where the type of activity needs to be identified statistically. This relies upon the users wearing such devices 24-7. In contrast, the Howz data arise from static infra-red motion sensors, door sensors and smart plug adapters. This data is sparse in comparison to wearable devices, but the sensor is labelled with a location. Examples of labelled sensors include motion sensors; landing and lounge, appliance sensors; toaster and fridge door, and door sensors; front door and bedroom. The median number of sensors per household is four.

Figure 1 presents a household provided by Howz with three sensors installed. The data are binary indications of presence or absence of activity for each sensor with a time stamp for the occurrence of each categorical observation, taken from a sensor. The households provided by Howz for this research are single occupancy. Howz advises the households to place the sensors on regularly used devices/locations. There is a wide range of number of sensor activations per day with 2 at the 5-percentile, 24 median, and 180 at the 95-percentile. Howz has had an increasing number of households since its creation in 2017 with rapid increases in customers resulting from multiple partnerships and research projects, as well as the COVID-19 pandemic. The median household has more than one year of data available.

In order to identify changes in sensor activation order day to day, we consider the nature of this data as a categorical sequence of events. We take each day to be a sequence of data, this will enable us to identify changes while considering the obvious daily structures, e.g., from figure 1 multiple triggering of the kettle followed by the bathroom followed by the front door. We define the common behaviour seen across days as a routine. An example of a regular routine would be waking up in the morning, putting on the kettle, letting the cat out and having some toast. This small routine could result in the following sequence: Hallway motion, Kettle, Back Door, Toaster. Many customers have very flexible routines, therefore, any method developed must be tolerant of alterations in this varied but regular behaviour. We aim to identify if changes in these routines occur.

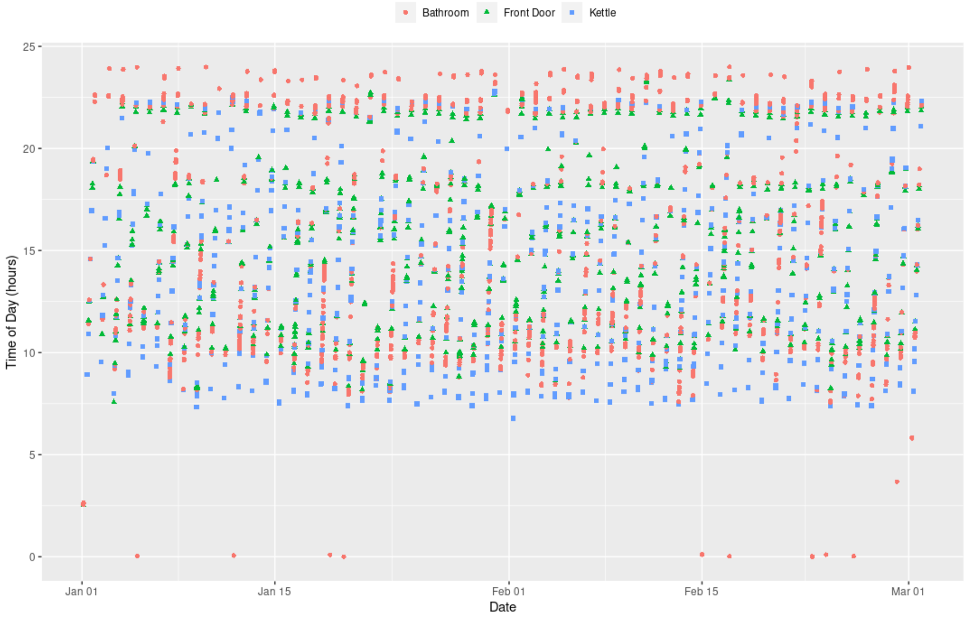

FIGURE 1 Colour-coded sensor events for a Howz household with 3 sensors; Bathroom, Front Door and Kettle.

1.2 | Motivation and Structure

A common way to detect changes in behaviour is to use changepoint detection. Höhle (2010) presents methods to detect changes in categorical time series using multi-categorical regression models where they monitor the category probabilities seen within the data. Recently, Plasse and Adams (2019) propose to detect changepoints in multiple categorical data streams that monitor the category probabilities of a multinomial distribution. Generally these approaches focus on estimating changes in the probabilities of sensors activating within a single day but will not detect changes in the order of the events.

To focus on the ordering of events, He et al. (2019) and Fradkin and Mörchen (2015) derive methods for mining sequential patterns; identifying interesting sub-sequences and their frequency within the data. However, when deciding which subsequences are important, these methods don't consider that sequences can be very varied and still be part of the same underlying behaviour. Applying this method could potentially focus on sub-sequences which are important for characterising the broad routine but are not sensitive to identifying irregular behaviour.

Dafé et al. (2015) introduce a method to identify how similar two categorical sequences subject to noise are. The method allows for sequences to not match completely and still be classed as very similar by avoiding penalising too heavily on the mismatches within. This is ideal for our application, with the stipulation that the noise we need to account for is not meaningless. Dafé et al. (2015) are motivated by very large genomic sequences of the same length. In contrast, our sequences are relatively short and a household will be unlikely to have the same number and order of events each day. Figure 1 shows there are a different number of bathroom and kettle observations in the morning across January and February.

In this paper, we tackle the challenge of identifying changes within short categorical sequences with varied lengths and differing behaviour. We approach this problem by letting each day be a sequence of events for a household. To decide if a change in behaviour has occurred, we need a notion of typical behaviour for a household. Motivated by practices in Howz, we define this as a set of regular sequences for the household where each sequence would be a day of categorical sensor events. Our contribution is in developing a new metric for assessing the similarity of two categorical sequences of potentially different lengths. Then using this similarity measure to assess whether a new day is conforming to regular behaviour or should be flagged as irregular.

We develop a novel approach that directly addresses the challenges of a small number of sensors, small and varied sequence lengths, and variation within regular sequences to detect a change in behaviour. In section 2, we describe the set up for this problem and the proposed method is developed in section 3. Section 4 provides a simulation study and section 5 gives an example using the Howz dataset. In section 6, we summarise our results and discuss potential avenues for further research.

2 | PROBLEM SET UP

To define a change in routine for a household we must first consider what is regular behaviour. The method we propose assumes we have existing data to characterise regular behaviour for a household. We take multiple weeks of data, resulting in a set of regular sequences for every household, one per day.

Formally, let R denote the set of regular sequences that form the null distribution to compare against, and T be the set of new sequences to be tested. Let $X^R \in R$ be a sequence from the regular sequence set whose length is $|X^R|$. Furthermore, let $\mathcal{L}$ denote the library set of sensors such that X^R is defined as a sequence of categorical events:

$$X^R = \{x_j \ : \ x_j \in \mathcal{L} \text{ for } j = 1, \ldots, \left|X^R\right|.\}$$

The notation similarly extends to the set of test sequences T. Reference to the regular and test sets are not essential when defining the sequence silhouettes (section 2.1) and the similarity metric (section 3). As such, explicit reference in the notation to the sequence sets is suppressed and only re-introduced where necessary. The aim is to take the vector sequences in T and assess whether, individually, they are consistent with regular behaviour (as described by set R). Typically the test set, T, will be of size 1.

In the following sections we use a similarity measure as a method for assessing if a sequence of activities is regular behaviour. However, there are a few application specific constraints on the data we must consider. Firstly, we have a small number of sequences in the set of regular sequences. This is because we want to keep the set of regular sequences up to date with the regular behaviour of the household. For example, if a person has a fall then their regular behaviour may change while they recover. If we are using, for instance, three months of regular behaviour then several new days may appear irregular due to the known change in behaviour. Once a change has been confirmed by the client we can use the new data as the regular set. Thus we use a smaller set of regular sequences that evolves over time, although how these evolve will not be addressed in this paper. We also need to acknowledge the small lengths of each sequence which vary day to day e.g., between 7 and 42 from the data in figure 1. An example of why we may see such variation is a household which spends some days indoors and some outdoors. It is also important to acknowledge some households have very varied sequence patterns. Further to the practical concerns over the structure of the data, the developed method for identifying changes in behavioral patterns should also be scalable. Specifically, the computational efficiency of the method should be sufficiently fast in order to analyze new days of data for a large number of clients over night.

We present an approach for comparing two sequences that addresses these challenges in section 2.1 before discussing how to calculate a similarity score based on these comparisons in section 3.

2.1 | Sequence Silhouettes

In contrast to numerical data, the similarity between categorical data sequences is not as trivial to define. Dafé et al. (2015) introduce a way to calculate the similarity by comparing subsequences within both the new sequence and regular sequence. This identifies the small routines (the subsequences) and bases the similarity score on if they appear within both sequences. Consider the example sequence X with $|X| = 7$ in table 1, which consists of a library of 3 sensors: door (D), kettle (K) and motion (M).

Index	1	2	3	4	5	6	7
Sequence: X	D	M	K	D	D	D	D

TABLE 1 Example of an activity sequence with Door (D), Kettle (K) and Motion (M) sensors.

First we must find all possible subsequences of X whilst preserving order. Clearly the number of subsequences increases exponentially when considering longer sequence lengths. As the number of observations per day varies considerably, we choose to impose an upper limit, $\mathcal{K} \leq |X|$, on the lengths of the subsequences to consider. The choice of $\mathcal{K}$ is examined in Appendix E.

Despite this upper bound, evaluating all possible subsequences of length $k \in 1, \ldots, \mathcal{K}$ scales poorly. Specifically, if the library set consists of $r = |\mathcal{L}|$ sensors, then there are up to $\frac{r(r^{\mathcal{K}}-1)}{r-1}$ possible subsequences to consider. To address this issue, Dafé et al. (2015) reduced the search space by using 'sequence silhouettes' that focus on the start and end of each subsequence for each possible subsequence length k. For example, the sequences DDK and DKK would be classed as belonging to the same sequence silhouette D*K.

Only counting unique silhouettes, for all $k > 1$ we would have a maximum of r^2 options and for $k = 1$ a maximum of r silhouettes. Therefore the total number of possible sequence silhouettes, P, in each sequence is $r + (\mathcal{K} - 1)r^2$.

We give Definition 2.1 for each sequence silhouette (SQS), using a general sequence, X, for the condensed search method.

Definition (SQS)

A sequence silhouette is defined by the triple $S = (s, s', k')$ where items $s, s' \in \mathcal{L}$ respectively define the first and last categorical event of the subsequence that has length $k = k' + 1$. We define the set of all possible silhouettes as $\mathcal{P} = \{S = (s, s', k') : s, s' \in \mathcal{L}, k' = 0, \dots, \mathcal{K} - 1\}$. The set of sequence silhouettes that exist for a given sequence, X with cardinality $|X|$, is defined as $\mathcal{P}_X \subseteq \mathcal{P}$ where $S \in \mathcal{P}_X$ if there exists some initial position h such that $x_h = s$ and $x_{h+k'} = s'$. It is possible that a sequence silhouette can match multiple times for a given sequence, and so we record the initial positions of all matches in the set $\mathcal{H}_{X,S}$.

Returning to the example in table 1, if we take X, an example of the sequence silhouettes found if we search through the sequence is presented in table 2.

D	M	K	D	D	D	D	s	s'	k'
$\underline{\text{D}}$	M	K	D	D	D	D	D	D	0
D	$\underline{\text{M}}$	K	D	D	D	D	M	M	0
D	M	$\underline{\text{K}}$	D	D	D	D	K	K	0
⋮			⋮			⋮	⋮	⋮	⋮
$\underline{\text{D}}$	$\underline{\text{M}}$	K	D	D	D	D	D	M	1
D	$\underline{\text{M}}$	$\underline{\text{K}}$	D	D	D	D	M	K	1
D	M	$\underline{\text{K}}$	$\underline{\text{D}}$	D	D	D	K	D	1
⋮			⋮			⋮	⋮	⋮	⋮
$\underline{\text{D}}$	M	$\underline{\text{K}}$	D	D	D	D	D	K	2
D	$\underline{\text{M}}$	K	$\underline{\text{D}}$	D	D	D	M	D	2
D	M	$\underline{\text{K}}$	D	$\underline{\text{D}}$	D	D	K	D	2
⋮			⋮			⋮	⋮	⋮	⋮

TABLE 2 Sequence silhouette graphic for sequence X in table 1; the left part of the table gives the sequence with the start and end of each silhouette (underlined) and the right part of the table gives the sequence silhouette triples (s, s', k') as defined in Definition SQS.

In section 3 we propose a similarity metric that compares a single sequence within T and compares it to all the sequences within R. We will score each silhouette from a sequence separately and combine across sequences to assess if the occurrence of a silhouette in the new sequence is statistically different from those in R.

3 | PROPOSED NEW SIMILARITY METRIC

Using the sequence silhouette framework, we design a similarity score to decide how similar two sequences are by looking at the sequence silhouettes they share and the information inside them. Table 3 gives an example of a sequence silhouette two sequences, X and Y, have in common.

Index	1	2	3	4	5	6	7	8	SQS		
Sequence: X	M	D	M	K	D	D	D	D	D	D	6
Sequence: Y	D	K	M	D	M	D	D	D	D	D	6

TABLE 3 Example of two sequences with sequence silhouette $(D, D, 6)$ with the start and end of the sequence silhouette underlined.

When making a comparison between sequences we consider all instances of a sequence silhouette. However, to increase computational efficiency we first created an indexing system which can decides whether a sequence silhouette is in the sequence.

In order to assess how similar the two sequences are we consider two factors, the presence of the sequence silhouette, and the elements within it. We take inspiration from Dafé et al. (2015) who designed a score function which penalised heavily on internal mismatches. However, for household behaviour small differences in the internal ordering are not as important (figure 1) and so we do not expect strict routines to exist. Therefore, we want to penalise a mismatch between the two subsequences in a different way than in Dafé et al.

In section 3.1 we design a new similarity score for each silhouette. Section 3.2 then proposes an adjusted score which takes into account the relative sizes of the sequences. This requires the maximum and expected scores which are not trivial to calculate and are considered in section 3.4 and 3.5 respectively.

3.1 | Similarity Score Function

The metric for the similarity score for comparing one instance of a sequence silhouette match between two sequences is given by Definition 3.1.

Definition (Similarity Score)

Consider two categorical sequences X and Y. Let $k \in \{1, \dots, \mathcal{K}\}$ be the length of the subsequence the sequence silhouette, $S = (s, s', k')$, refers to, with $k' = k - 1$. For all shared sequence silhouettes $S \in \mathcal{P}_{X,Y} = \mathcal{P}_X \cap \mathcal{P}_Y$, define $\mathcal{H}_{X,Y,S} = \mathcal{H}_{X,S} \times \mathcal{H}_{Y,S}$ to be the set of all paired initial positions $(h, g) \in \mathcal{H}_{X,Y,S}$ so that $x_h = y_g = s$ and $x_{h+k'} = y_{g+k'} = s'$.

Given scaling factors, $\beta \in \mathbb{R}_{>0}$ and $\lambda \in \mathbb{R}_{>0}$, we definethe similarity metric as,

$$\Gamma(X \mid Y, h, g, S) = \beta + \begin{cases} \sum_{c=1}^{k'-1} \gamma(x_{h+c}, y_{g+c}) & \text{if } k' > 1, \\ 0 & \text{otherwise.} \end{cases}$$

where γ is the recursive function:

$$\gamma(x_{h+c}, y_{g+c}) = \begin{cases} \gamma(x_{h+c-1}, y_{g+c-1}) + \lambda & \text{if } c \geq 1 \text{ and } x_{h+c} = y_{g+c}, \\ \gamma(x_{h+c-1}, y_{g+c-1}) & \text{otherwise.} \end{cases}$$

We give an example in table 4 to show how the inside of the sequence silhouette (SQS) is calculated. This shows that the score would be $\beta + 4\lambda$ for the silhouette match in this example. Definition 3.1 gives the similarity score for

Aligned Index	1	2	3	4	5	6	7	SQS		
Sequence: X	$\underline{\text{D}}$	M	K	D	D	D	$\underline{\text{D}}$	D	D	6
Sequence: Y	$\underline{\text{D}}$	K	M	D	M	D	$\underline{\text{D}}$	D	D	6
Score		0	0	λ	λ	2λ				

TABLE 4 Evaluation of the similarity score between two sequences based on the sequence silhouette $(D, D, 6)$, resulting in a score of $\Gamma(X \mid Y, h, g, (D, D, 6)) = \beta + 4\lambda$.

one match of a silhouette. Across all matches the similarity measure is:

$$\begin{aligned}\Gamma(X \mid Y, S) &= \sum_{(h,g)\in\mathcal{H}_{X,Y,S}} \Gamma(X \mid Y, h, g, S) \\ &= \left|\mathcal{H}_{X,S}\right| \left|\mathcal{H}_{Y,S}\right| \beta + \begin{cases} \sum_{(h,g)\in\mathcal{H}_{X,Y,S}} \sum_{c=0}^{k'-1} \gamma(x_{h+c}, y_{g+c}) & \text{if } k' > 1, \\ 0 & \text{otherwise.} \end{cases}\end{aligned} \tag{1}$$

Equation 1 is the similarity score for one silhouette for all instances between two sequences X and Y. By re-introducing reference to the set of regular sequences, the total similarity of sequence X^R against all other sequences within R with respect to a given silhouette is:

$$\Gamma(X^R \mid R, S) = \sum_{Y\in R\setminus X^R} \Gamma(X^R \mid Y, S). \tag{2}$$

The similarity metric for a test sequence from T, X^T, is likewise defined by comparing it against all sequences in R:

$$\Gamma(X^T \mid R, S) = \sum_{Y\in R} \Gamma(X^T \mid Y, S). \tag{3}$$

3.2 | Adjusted Similarity Score

Intuitively, due to the large variation in the sequence length of Howz data, large sequences are likely to have more sequence silhouettes and thus, have larger similarity scores when compared to shorter sequences. Therefore, we need to introduce a scaling method to mitigate the effect of varying sequence size. Taking inspiration from the Rand Index used in clustering (Rand, 1971; Hubert and Arabie, 1985), we correct for the larger variation in sequence length using the expected and maximum scores as follows.

Definition (ASC) The Adjusted Similarity Score for a sequence X and sequence silhouette S:

$$\text{ASC}(X \mid S) = \frac{\text{Similarity Score - Expected Score}}{\text{Maximum Score - Expected Score}} = \frac{\Gamma(X \mid S) - \mathbb{E}(\Gamma(X \mid S))}{\max(\Gamma(X \mid S)) - \mathbb{E}(\Gamma(X \mid S))}$$

In section 3.1, we defined the similarity score. In the following sections we explore the maximum and expected scores.

3.3 | Maximal Packing

To calculate a maximum score, we must first consider the maximum number of times a silhouette could be present in a sequence, X of length $|X|$, the maximal packing. A first naive attempt at an upper bound would be $|X| - k'$ as we consider the overlapping of sequence silhouettes, i.e. a sequence silhouette instance may start within another instance. For example, using X defined in table 3, there exists two silhouettes $(D, D, 2)$ which overlap between index 4 and 7. However, there are ways to improve upon this maximum bound. When considering types of overlap, there are two different types of silhouette: a silhouette where the start and end are the same ($s = s'$) and one where they are different ($s \neq s'$). For the first case, it is clear the maximum number of silhouettes occurs when the sequence is entirely the same event. The second case is less intuitive, we give the definition for the maximal packing here with the full derivation in appendix A.

Proposition 1 *(Maximal Packing) For the different cases, the maximum number of occurrences for a sequence silhouette, S, within a sequence, X, where $|X|$ is the length of the sequence,*

$$\textit{Maximal Packing} = \begin{cases} |X| - k' & k' \geq 0 \quad s = s', \\ k' \left\lfloor \frac{|X|}{2k'} \right\rfloor + w_X & k' \geq 1 \quad s \neq s'. \end{cases}$$

where $\lfloor n \rfloor$ denotes the integer part of the number n. The term w_X describes how many additional matches of the SQS after perfect overlapping. Specifically, the residual part of the sequence after accounting for perfect overlapping has length $v = |X| \mod 2k'$, where the modulo operator returns the remainder of the Euclidean division. Hence:

$$w_X = \begin{cases} 0 & v < k \\ v \bmod k' & \textit{otherwise.} \end{cases}$$

An example of maximal packing for the silhouette $(D, K, 2)$ for a sequence of length 7 would be $DDKKDDK$ with a count of 3 occurrences of the silhouette: two in one instance of perfect overlapping, and one occurrence in the remaining sequence.

3.4 | Maximum Score

Given the number of sequence silhouettes gained from the maximal packing, we next need to compute the associated maximum score. For the first case, we assume that each sequence silhouette instance within the maximally packed sequence will have a match at each element. However, this does not hold true for the second case (see table 11,

Appendix B). The overlap results in a lower score. This is because the score for each silhouette match in the $2k'$ space will get smaller, due to increased mismatches within the sequence silhouette. The following definition describes the maximum score for these cases. The derivation is given in the appendix B.

Theorem 2 *(Maximum Score) The maximum overall score for all instances of the sequence silhouette, S, in the two sequences, X, Y, with cardinality $|X|$ and $|Y|$, respectively. Therefore, the maximum score $(\max(\Gamma(X \mid Y, S))$ when comparing these sequences is:*

$$\max(\Gamma(X \mid Y, S)) = \begin{cases} \beta \ |X| \ |Y| & \text{if } k' = 0, \\ (|X| - k') \ (|Y| - k') \left(k'\beta + \frac{k'(k'-1)\lambda}{2}\right) & \text{if } k' \geq 1 \text{ and } s = s', \\ \beta k \lfloor \frac{|X|}{2k'} \rfloor k' \lfloor \frac{|Y|}{2k'} \rfloor & \text{if } k' = 1 \text{ and } s \neq s', \\ \left(M^\star \left\lfloor \frac{|Y|}{2k'} \right\rfloor + \sum_{c=1}^{w_Y} M(c)\right) \left(k' \left\lfloor \frac{|X|}{2k'} \right\rfloor + w_X\right) + A(w_X, w_Y) & \text{if } k' > 1 \text{ and } s \neq s', \end{cases}$$

where $A(w_X, w_Y)$ is defined in Appendix C to account for subsequent score matching. We define the function M by:

$$M(c) = \begin{cases} \beta + \frac{\lambda(k'-c-1)(k'+c)}{2} & k' > 1, \ 0 \leq c < k' - 1, \\ \beta & k' > 1, \ c = k' - 1, \end{cases}$$

where c has maximum value w_Y. For ease of notation we denote $M^\star$:

$$M^\star = k'\beta + \begin{cases} \lambda & \text{if } k' = 2, \\ \frac{\lambda k'(k'-1)(k'+1)}{3} & \text{if } k' > 2, \\ 0 & \text{otherwise.} \end{cases}$$

We now have a method to calculate the maximum score between two sequences, and hence, the maximum score over all the regular sequences is:

$$\max(\Gamma(X \mid S)) = \sum_{Y \in G} \max(\Gamma(X \mid Y, S)).$$

To ensure the maximal score derivation holds we add the following constraint. For a fixed $\mathcal{K}$,

$$\sum_{c=1}^{\mathcal{K}-1} \gamma(x_{h+c}, y_{g+c}) < \beta,$$

such that β and λ do not change across sequence silhouettes.

3.5 | Approximated Expected Score

For each sequence silhouette, when trying to calculate the expected score for the new sequence using the set of regular sequences, we quickly run into computational challenges. The search space would increase greatly with k, as the inside of the silhouette must be considered. As minimizing computational complexity is a key priority we choose to approximate the expected score. A reasonable approximation would need to take into account the relative probability of the silhouette whilst scaling with the number of sensors. Thus we propose estimating the expected score as follows.

Definition (Approximated Expected Score) Considering sequence $X \in R$, and sequence silhouette $S = (s, s', k)$, the approximated expected score is defined as:

$$ES(X \mid S) = \frac{r^2 P(s) P(s')}{|R|} \times \max(\Gamma(X \mid S)),$$

where r is the size of the library $\mathcal{L}$ and $P(\cdot)$ is the estimated probability of seeing a certain event using the regular behaviour sequences, i.e. the relative frequency of events. This can be defined similarly for a sequence within set T.

4 | SIMULATION STUDY

We now consider how Algorithm 1 performs on simulated data. The method is implemented in R, (R Core Team, 2020), for daily sequences under a range of scenarios, in order to test for sensitivity. We focus on the basic package offered by Howz i.e., a library consisting of 3 sensors. Let these sensors be Door (D), Kettle (K) and Motion (M) arbitrarily. We will focus on a variety of types of household starting with those with no visible routines i.e., independent data.

There are many components that we can vary to mimic the real data example in section 5. We need to consider how to simulate the entries with different probabilities for each sensor, the varied lengths of sequences and how many sequences are in the regular set. For all simulations, we report significance level $\alpha = 0.05$.

4.1 | Households with No Visible Routines

To simulate independent days, the data are generated based on the probabilities for each category. Due to the short sequences, a randomly generated sequence can exhibit wildly different probabilities than that which it was generated from. Thus rather than considering random sequences we generate a sequence such that the proportions approximately match the probabilities of the sensors being triggered and then generate random permutations of this sequence for the different days. For example, a sequence of length 10 with probabilities (0.8,0.1,0.1) would have a sequence MMMMMMMMDK which would be randomly permuted.

To keep the set of regular behaviour up to date and to reduce computation time, we consider having 8 weeks, 4 weeks and 2 weeks of regular sequences i.e. $|R|$. Let the upper bound of the length of SQS be $\mathcal{K} = 3$. Then, choose $\beta = 1$ and $\lambda = 0.5$ to ensure it fits the maximal packing constraint in section 3.4. In simulating the data, we consider varying the sequence lengths by sampling the sequence lengths uniformly between a minimum $c \in 4, 10, 25$

and maximum $d \in \{10, 25, 50\}$. The minimum lengths are greater than or equal to the number of sensors to ensure that each sensor has a chance of being seen in a daily sequence. Let the length of each sequence, X, be $|X| \sim \text{Unif}(c, d)$ where c and d are the minimum and maximum length of the sequence respectively. The elements of the categorical sequence X, x_j for $j \in \{1, \ldots, |X|\}$ are Door, Kettle and Motion with the vector ρ, of length 3, being the probabilities of triggering each of the 3 sensors independently at each j. Table 5 presents three potential independent households with 3 sensors. The vectors $\boldsymbol{\rho}_1$ denote the probabilities for each sensor and $\boldsymbol{\rho}_2$ are the changes in routine we consider.

Household	ρ_1	ρ_2
A	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	(0.5,0.4,0.1) , (0.8,0.1,0.1)
B	$(0.8, 0.1, 0.1)$	(0.7,0.15,0.15) , (0.6,0.2,0.2) , $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$
C	$(0.65, 0.25, 0.1)$	(0.8,0.1,0.1) , $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$

TABLE 5 Details of the household probability sets used in simulation.

The first set are small changes in probability, with the final one simulating a large change in behaviour. We also simulate new sequences using $\boldsymbol{\rho}_1$ to test the false detection rate.

We perform 500 replications for each set of parameters, simulating the sequence lengths and observed sensors separately each time. The values in table 6 are the proportions of sequences labelled as irregular, calculated using algorithm 1.

In general, from table 6, we see that the method has good power for this range of household behaviours. For Household B with the very small change in behaviour from (0.8,0.1,0.1) to (0.7,0.15,0.15), the method struggles to pick up the change when the minimum length of sequence is 4. This is most likely due to the small change in probability resulting in no difference in the simulated new sequence compared to regular sequences for the varying sequence lengths. The false positive rate is controlled as the number of regular sequences increases for these households. As expected, the power increases as the size of the change increases. The false positive rate also increases as the number of sequences in the regular set decreases. Given the small number of regular sequences, only a small number of scores are used to approximate the null distribution. This means we are likely to see new sequences of regular behaviour in the tails of the distribution. Therefore, this method is likely to have a higher false detection rate than the significance level when the regular set is small. These show similar results to table 6. We also varied the length of $\mathcal{K}$,

β, λ and increased the number of sensors. These did not make a material difference to the results (see appendix E for reference).

4.2 | Example of a Household with Regular Routines

We consider a household with varying lengths of sequences and a regular routine. A regular routine refers to a household with a few small common routines present in most of the days in the set of regular sequences.

We simulated each regular sequence using the sequence in table 7 as a template. The sequence has a few small routines, to give an example of regular behaviour. To simulate variations in this sequence, at each position with probability 0.4 the event is removed. This results in a noticeable difference in sequences without changing the template sequence too drastically. To simulate regular behaviour for the user, we use the same underlying process for the new sequence as the regular sequences.

We consider three changes in behaviour for this household. The first and second change use the sequence in table 7 as a template to simulate the regular sequences. At each position in the sequence there is a probability of $\frac{1}{3}$ that the event is removed and a probability of $\frac{1}{3}$ that it is replaced with an event simulated with equal probability. Alternatively, at each position in the sequence there is a probability of 0.2 that the event is removed and a probability of 0.5 that it is replaced with an event simulated with equal probability of each sensor. For someone with a regular routine, an example of a large change we would want to flag as irregular would be the loss of routine. We simulate this randomly by setting the probability of each sensor as equal, varying sequence lengths between 4 and 25 uniformly. We again perform 500 simulations for each set of parameters in the table.

The results in table 8 show similar false positive rates to the households with no visible routine. The power of the method appears to be driven by the modification of elements in a routine rather than the absence of elements which is our intention in the design of the similarity score. This is well aligned with the application to detecting changes in routines.

4.3 | Example of a Household with Strict Routines

Consider a household with a strict routine, with little variation day to day. Using the same sequence in table 7, we follow the same procedure as in section 4.2. However, at each position in the sequence up to 24, with probability 0.3 it is swapped with the event ahead of it. This gives new sequences which are more similar to the initial sequence in table 7 but reflect subtle changes in behaviour such as using the kettle before the toaster. We simulate the change in behaviour in the same way as section 4.2. We perform 500 simulations for each set of parameters.

Note that due to the strict routines simulated there will be smaller variance in the regular set. This results in a slightly higher false positive rate than in previous examples - although it is still controlled. This also means that smaller deviations will be flagged as irregular behaviour and thus the power in this case is higher than in previous setting.

Overall the simulations demonstrate that the variability in the regular set is key to controlling the false positive rate. The larger number of sequences in the regular set also decreases the false positive rate but 28 sequences is viable. The larger the regular set, the more power we have to detect smaller changes in behaviour.

5 | EXAMPLES USING THE HOWZ DATASET

We show two households provided by Howz. The first household has 49 days of data with 3 sensors; Bathroom motion, Front Door and Kettle. The second household has 88 days of data with 4 sensors; Hallway motion, Kettle, Main door and Microwave. We take two weeks as the regular set as this is typical practice within Howz. Similar to the simulation study, we set $\mathcal{K} = 3$, significance level $\alpha = 0.05$ and choose $\beta = 1$ and $\lambda = 0.5$. To calculate the estimated probability of a sensor being seen for the expected score, we use the proportion seen in the regular set.

Figure 2 presents the sensor activity and the results for our first household. From a separate analysis on overall activity and the start/end of a users day, Howz has seen that there have been no discernible changes in the data. The method proposed here echoes these results, with no silhouettes flagged as irregular. This shows the method, even with a small regular set, does not flag any changes for this household.

We report the computational time for this example, to show how fast this method performs on a household. We

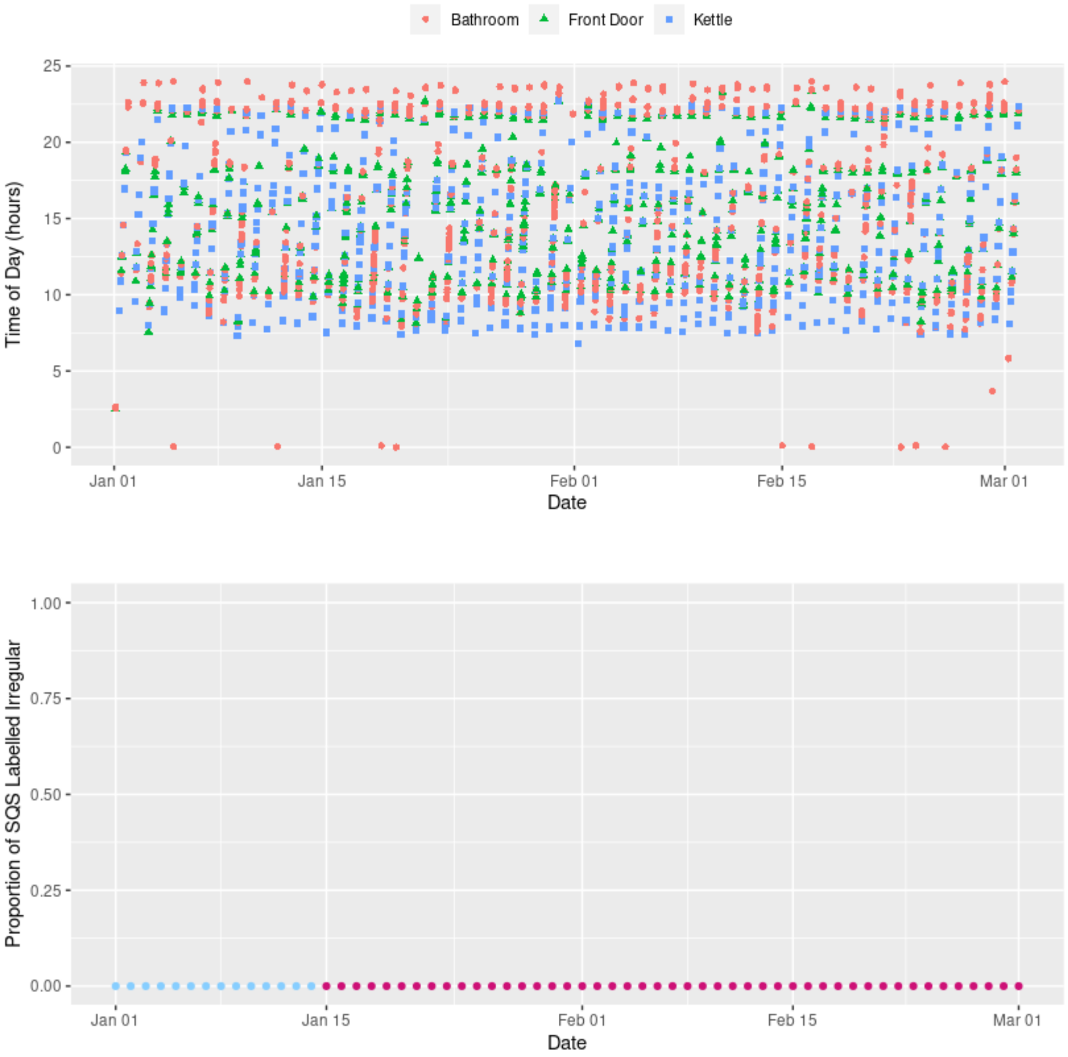

FIGURE 2 Top figure: Household with 3 sensors, showing the period between January and March. Bottom figure : Results with dots indicating regular set (light blue) and proportion of sequence silhouettes labelled irregular (red).

look at this in two parts; time taken to calculate the regular sequence similarity scores and the test for whether a new sequence is irregular or not. We expect in practice, the similarity scores for the regular sequences to be calculated offline so that we can use Irregular Activity Sequence Detection algorithm on the daily data as it comes in. Hence, the computational time for the regular sequence similarity scores for this example is approximately 112 seconds (virtual machine with 4 cores, 3Ghz processor and 16Gb RAM) and the average time to run the method on the new days i.e., the days between July 8^{th} and October 10^{th}, as around 9.9 seconds. Using simple parallelisation on the regular sequence similarity scores, using 3 cores we can reduce this to approximately 38 seconds. However, for Howz it may be more useful to parallelise over the different households, rather than within the method.

Figure 3 present results which Howz have prior knowledge of a change occurring in the household. Howz proposed this household as an example due to the fact that they had confirmed events in the middle of August and September with the household. A key feature of the events is a lack of any sensor activation, as the person was no longer living in the house. From Figure 3(top) we can clearly see a change in sensor behaviour prior to the first gap in sensor data. The household changes from activity throughout the day and night, to just morning and evening activity, dominated by the Main Door sensor. Looking closer, 3 or 4 days prior to this, we can see that the main door occurs more frequently in the evening and the kettle earlier in the morning. Whilst our analysis does not take into account the timing of the sensors, the bottom figure provides interesting additional insight. We pick up the event occurring in the middle of August with the proportion of irregular silhouettes increasing until intervention. The initial irregular silhouettes are highlighted a few days prior to the clear morning/evening change which looks to capture the more subtle changes highlighted earlier.

Leading up to and during the event, the method flags the silhouettes (Hallway, Hallway, 0) and/or (Hallway, Kettle, 1) as irregular. The first indicates that overall the Hallway movement decreases meaning that the individual is less active in the Hallway relative to the sequence lengths. In conjunction with the decrease of Hallway, Kettle as a routine this could indicate a change in the activity level in the household. This was the conclusion reached by Howz when considering other analyses of aggregated data.

The silhouettes flagged as irregular towards the end of September are (Hallway, Hallway, 0) and then (Main Door, Main Door, 0), (Main Door, Main Door, 1) respectively. The first is due to an increase in Kettle and Main Door observations and as such the methodology is expecting an increase in Hallway sensors which does not manifest. The second is due to a decrease in the main door sensors from a typical value of around 8-20 to just 2 observations in conjunction with Kettle and Hallway maintaining their usual frequency of activation.

This example shows the method is able to detect a change in routine before the change would usually be flagged by Howz alongside providing a more targeted description of what activity has changed. Naturally the description

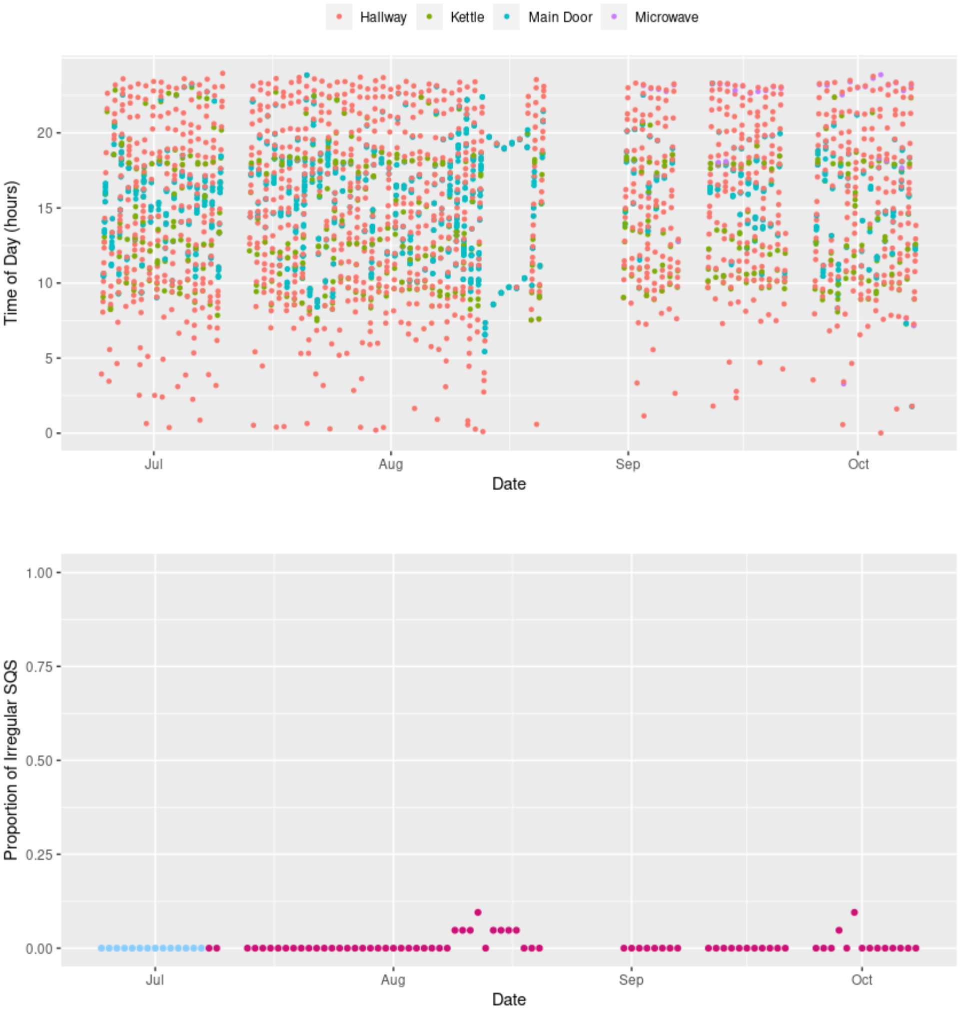

FIGURE 3 Top figure: Household with 4 sensors, showing the period between end of June and mid October. Bottom figure : Results with dots indicating regular set starting end of June (light blue) and proportion of sequence silhouettes labelled irregular (red) out of total possible for that sequence.

(Hallway, Hallway, 0) is not appropriate to send to a user and so some degree of interpretation of the change by the Howz team is required.

6 | CONCLUSION

We have presented a novel method to detect changes in the behaviour of households on a day to day basis using sparse categorical data. The method is able to detect changes in new sequences that may be varied in length, including very small lengths. In practice for the Howz dataset, we cannot expect more than 8 weeks of regular behaviour. We have demonstrated that the method works with just two weeks of regular data, although increasing the number of regular days does improve the false detection rates, as expected. The model is very adaptable such that it can be applied to different situations with different constraints by changing the upper bound on the subsequence length, $\mathcal{K}$ and the partial matching parameters, β and λ, to suit the specific purpose and risk profile of the household. For example, if it was only important that the start and end of the silhouette occurred, λ could be set to 0, for applications where the inside of the silhouette holds no meaning. In practice, we would not update the regular sequences every day, which accounts for the majority of the computation time. After the offline calculation of the scores for the regular sequences, as discussed in section 5, the method is fast enough that Howz could run all households overnight. We have shown this could be sped up further using parallelisation. However, within the industrial context, parallelisation may be better utilised over households rather than within the method itself.

A more accurate representation of the expected score would account for the potential dependence between the sensors. However, the adopted definition was favoured for computational ease, which performs well in the simulation study. The households presented here are single occupancy, a future avenue of research would be considering multiple occupancy and how whether individual changes can be identified within the household data.

Currently the method is used on daily activity sequences but it is easily extended to weekly or monthly sequences. This could potentially provide insight into the longer term behaviour of the household. However, this will have computational drawbacks. Alternatively, we could further develop the approach to incorporate the time aspect explicitly into the model. One potential approach would be taking specific periods within the day and running the analysis on these subsequences to identify changes in behaviour. This would allow us to place more emphasis on distinguishing between two sequences e.g., DKMDDDDKMK where the initial KM routine was in the morning verses at lunchtime. Another approach would be including a time component to the similarity score. The method will still pick up a change due to the sequence silhouettes around this routine changing but timing changes may be identifiable.

Within our application, the aim is to identify changes in behaviour within households such that timely interventions can take place. The algorithm presented here contributes to the first part of this aim; identifying irregular sequences. Following this identification we need decision processes in place to take this information and turn it into appropriate timely interventions tailored to the specific changes within the households. The advantage of our method is that the irregular sequences can be detected quickly such that interventions can be more timely. These early interventions can aid in improving wellbeing, supporting households that require more care, and preventing hospitalization. However, more research is required to measure the link between identifying irregular sequences and subsequent changes in wellbeing.

Data Availability

The data that support the findings of this study are available from Howz. Restrictions apply to the availability of these data, which were used under license for this study. Data are available from the industrial authors with the permission of Howz.

Acknowledgements

Gillam is grateful for the support of the Smith Institute who provided opportunities for Engineering and Physical Sciences Research Council CASE studentships with SMEs. Howz provided financial support of Gillam (through CASE funding) and Taylor (through IAA funding).

Killick is grateful for the support of the Engineering and Physical Sciences Research Council through grants EP/T014105/1 and EP/T021020/1.

A | MAXIMAL PACKING EXPLANATION

Discussed in section 3.3, there are two different types of silhouette: a silhouette where the start and end are the same $s = s'$ and one where they are different $s \neq s'$.

To consider overlap for the first case, it is clear the maximum number of silhouettes occurs when the sequence is entirely the same event.

Therefore the maximum number of times a SQS can occur is $|X| - k'$.

For the second case, the start and end events are different. The maximal way of packing the silhouettes is to take the SQS of length k and replace the first k' elements inside with the start event. For (D,K,k'):

$$\overbrace{\underbrace{D \quad D \quad D \quad D \quad \ldots \quad D}_{k'} \quad K}^{k} \quad \underbrace{K \quad \ldots \quad K}_{k'}.$$

Hence, k' is the maximum number of ways to fit a SQS into a space of length $2k'$. For a sequence of length $|X|$ divisible by $2k'$, $k' \lfloor \frac{|X|}{2k'} \rfloor$ is the maximum number of sequence silhouette occurrences. However, for sequences that do not divide by $2k'$, the residual part of the sequence has length $v = |X| \mod 2k'$, where the modulo operator returns the remainder of the Euclidean division. If $v < k$ then the residual sequence is too short for for the silhouette, but it is possible to pack additional silhouettes if $k' < v < 2k'$. Hence:

$$w_X = \begin{cases} 0 & v < k \\ v \bmod k' & \text{otherwise.} \end{cases}$$

Therefore, for the different cases, the maximum number of occurrences for a sequence silhouette for one sequence where $|X|$ is the length of the sequence,

$$\text{Maximal Packing} = \begin{cases} |X| & k' = 0 \quad s = s', \\ |X| - k' & k' \geq 1 \quad s = s', \\ k' \left\lfloor \frac{|X|}{2k'} \right\rfloor + w_X & k' \geq 1 \quad s \neq s'. \end{cases}$$

B | MAXIMUM SCORE EXPLANATION

For the first case where the SQS satisfies $s = s'$, we simply have that each sequence silhouette instance within the maximally packed sequence will have a match at each element (see table 10).

Hence, the score when comparing every instance between two sequences, X and Y, for silhouettes with the same start and end event is:

$$\max(\Gamma(X \mid Y, S) = \begin{cases} \beta \; |X| \; |Y| & \text{if } k' = 0 \text{ and } s = s', \\ (|X| - k') \; (|Y| - k') \left(k'\beta + \frac{k'(k'-1)\lambda}{2} \right) & \text{if } k' \geq 1 \text{ and } s = s', \end{cases}$$

This scoring method does not hold for the second case where the silhouette start and end events are different, $s \neq s'$. In table 11 we see that the score is reduced for each sequential aligned silhouette due to the increased number of mismatches.

To find a general formula for the second case, we consider one maximally packed instance in the new sequence and find the score compared to a maximally packed regular sequence. The results use a counter,

$$\{c \in \mathbb{N}_0 \,:\, 0 \leq c < k', k' = k - 1, k \in \{1, \ldots, \mathcal{K}\}\}.$$

Consider only $k' > 1$ as this is the first chance for overlap to start. For $c = 0$, we have the first row in table 11. The maximum possible score is $\beta + \sum_{m=1}^{k'-1} m\lambda$, as every element matches. Given the structure of the maximally packed sequence silhouettes, silhouette match $c \in \{1, \ldots, k'-1\}$ will have matches until encountering the first mismatch, and then the remainder will be mismatches. For $c = k' - 1$, everything inside the sequence silhouette will be a mismatch so the score will be β. This results in a general function, $M(c)$ for each c:

$$M(c) = \beta + \begin{cases} \sum_{m=1}^{k'-c-1} m\lambda + (k' - c - 1)c\lambda & k' > 1,\ 0 \leq c < k' - 1, \\ 0 & k' > 1,\ c = k' - 1. \end{cases}$$

Taking β out for each silhouette, the maximum score for a length of $2k'$ is:

$$M^{\star} = k'\beta + \begin{cases} \lambda & \text{if } k' = 2, \\ \underbrace{\sum_{m=1}^{k'-1} m\lambda}_{c=0} + \underbrace{\sum_{c=1}^{k'-2}\left(\sum_{m=1}^{k'-c-1} m\lambda + (k'-c-1)c\lambda\right)}_{c\,\in\,\{1,\ldots,k'-2\}} & \text{if } k' > 2, \\ 0 & \text{otherwise.} \end{cases}$$

$$= k'\beta + \begin{cases} \lambda & \text{if } k' = 2, \\ \frac{\lambda k'(k'-1)(k'+1)}{3} & \text{if } k' > 2, \\ 0 & \text{otherwise.} \end{cases}$$

We now have the maximal score for each silhouette, thus, we calculate the overall score for all instances of the sequence silhouette in the two sequences. Therefore, the maximum score when comparing sequences X and Y is:

$$\max(\Gamma(X \mid Y, S) = \begin{cases} \beta \ |X| \ |Y| & \text{if } k' = 0 \text{ and } s = s', \\ (|X| - k')\ (|Y| - k') \left(k'\beta + \frac{k'(k'-1)\lambda}{2}\right) & \text{if } k' \geq 1 \text{ and } s = s', \\ \beta k \lfloor \frac{|X|}{2k'} \rfloor k' \lfloor \frac{|Y|}{2k'} \rfloor & \text{if } k' = 1 \text{ and } s \neq s', \\ (M^{\star} \left\lfloor \frac{|Y|}{2k'} \right\rfloor + \sum_{c=1}^{w_Y} \mathrm{M}(c))(k' \left\lfloor \frac{|X|}{2k'} \right\rfloor + w_X) + A(w_X, w_Y) & \text{if } k' > 1 \text{ and } s \neq s'. \end{cases}$$

where $A(w_X, w_Y)$ is an extra term defined in Appendix C to account for subsequent score matching.

C | EXPLANATION OF EXTRA TERM IN MAXIMAL SCORE

The function $A(w_X, w_Y)$ describes the adjustment to the maximum score with the matching between the residual subsequence of X with the perfectly overlapping subsequences of Y. Specifically, we base the calculations against the maximal packing case where all of the intermediate sequence events for the silhouette are identical to the first, and assess what are the differences in the sequence score as we increase the occurrence of the last event from the right. To illustrate, table 12 presents the similarity score for the SQS $(D, K, 6)$ with all matched offsets when the intermediate events are all D (the initial match parameter β is ignored as this is already accounted for elsewhere in the maximum score formula).

Tables 13—17 presents the similarity score for the same SQS and sequence Y, but we sequentially replacing the right most intermediate event with the last matched event, i.e. K. In the final column of these tables, we note relationship of the scores with those in table 13, which consists of a swap of the first line with the appropriate iteration index line or a difference the score total.

These differences are collated into the following matrix that presents all score differences from an exhaustive investigation up to longest SQS considered in this paper ($\mathcal{K} = 7$, see Appendix E).

$$F = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -4 & -8 & -13 \\ 0 & 1 & 0 & 0 & -4 & -9 \\ 0 & 1 & 0 & 0 & 2 & -3 \\ 0 & 1 & 3 & 6 & 0 & 5 \\ 0 & 1 & 3 & 6 & 10 & 0 \end{bmatrix}$$

The values differences seen in the table for the i^{th} iteration correspond to the i^{th} column of F, up to the j^{th} row associated to the SQS length up to k'. This matrix has the property that the values sum to zero. This corresponds to the case where the lengths of sequences X and Y are both divisible by 2k, and so there are no residual subsequence after accounting for optimal packing. The adjustment term in the maximal score is then the sum of all entries in the upper-left $w_Y \times w_X$ submatrix of F, specifically:

$$A(w_X, w_Y) = \sum_{i=1}^{w_X} \sum_{j=1}^{w_Y} F_{j,i}.$$

We note that the entries in matrix F can be expressed formulaically according to:

$$F_{j,i} = \begin{cases} j(j-1) - \frac{i(i-1)}{2} & \text{if } 1 < j < i, \\ \frac{i(i-1)}{2} & \text{if } j > i, \\ 0 & \text{otherwise.} \end{cases}$$

C.1 | Final Algorithm

Using the approach described in sections 3.1-3.5, we calculate the similarity scores for a new sequence, X, within T comparing it to the regular set. This produces one similarity score for each sequence silhouette in a sequence. In order to assess whether these scores are significantly different from the regular set we need to compare the set of regular sequences to each other. We take a sequence within the regular set, R, and compare it to the remaining sequences in the set using the method in section 3. Then repeat for all the regular sequences. Note, we scale the scores for the new sequence and regular sequence, due to the extra comparison for the new score.

We estimate the null distribution of each sequence silhouette score using those calculated from the regular set via a kernel density (Racine and MacKinnon, 2007; Davison and Hinkley, 1997). We use this null distribution to obtain a p-value for each silhouette, $S \in \mathcal{P}_X$. However, if we were to simply reject the new sequence if a p-value is below significance level α, typically the size of the test will not be maintained due to multiple testing. Instead, a multiple testing procedure with the following hypothesis test is used for testing sequence X:

$$\begin{aligned} H_{0,S} &: \text{Sequence silhouette, } S \in \mathcal{P}_X\text{, is regular for the household.} \\ H_{1,S} &: \text{Sequence silhouette, } S \in \mathcal{P}_X\text{, is irregular for the household.} \end{aligned}$$

The new sequence is labelled as irregular for our application if one of the hypotheses, $|\mathcal{P}_X|$, is rejected, after correcting for the multiple hypotheses. More emphasis should be placed on correctly finding all sequences with irregular behaviour, rather than rejecting a few regular behaviour sequences. Contextually, this is to ensure the safety and well-being of the household. Hence, a multiple testing procedure which controls the False Discovery Rate (FDR) is used. We choose a procedure from Benjamini and Hochberg which focuses on controlling the FDR at significance level α using a sequentially modified Bonferroni correction. We calculate the p-values for each sequence silhouette using the estimated null distribution, before ordering them in ascending order, assign indexing $v_{(q)}$ for $q = 1, \dots, |\mathcal{P}_X|$. Let,

$$\hat{p} = \max\left(q \in \{1, \dots, |\mathcal{P}_X|\} \ : \ v_{(q)} \leq \frac{q\alpha}{|\mathcal{P}_X|}\right).$$

If $\hat{p}$ does not exist we fail to reject any of the hypotheses and consider the new sequence as regular. Else, we reject $H_1, \dots, H_{\hat{p}}$ and label the sequence as irregular. A condensed pseudo-code for the Irregular Activity Sequence Identification algorithm is given in Algorithm 1.

```
Irregular Activity Sequence Identification
  Input:       Regular set $R$ with sequences $X_i^R$ for $i = 1, \dots, |R|$.
               Test set $T$ with sequences $X_j^T$ for $j = 1, \dots, |T|$.
               Sequence silhouettes, $S \in \mathcal{P}$.
               Significance level $\alpha$.
               Parameters $\beta$ and $\lambda$ in evaluating similarity score function $\Gamma$.
  Initialise:  $U = \emptyset$, $V = \emptyset$.

Iterate For $X_j^T \in T$:

1.  For each $S \in \mathcal{P}_{X_j^T}$:
a.  Calculate the average similarity score, $\frac{\Gamma(X_j^T \mid R, S)}{|R|}$ based on equation 3.
b.  Scale using $ASC(X_j^T \mid R, S)$.
c.  For $i \in \{1, \dots, |R|\}$:
    i.   Calculate the average similarity score, $\frac{\Gamma(X_i^R \mid R, S)}{|R|-1}$ based of equation 2,
         where sequence $X_i^R$ is excluded from $R$.
    ii.  Scale using the $ASC(X_i^R \mid R, S)$ and add to set $U$.
d.  Evaluate the kernel density estimate of $U$ to define the null distribution.

e.  Calculate p-value using the density estimate and add to set $V$.
2.  Order $V$ in ascending order and assign indexing $v_{(q)}$ for the $q^{th}$ smallest p-value.
3.  Calculate,

    $$\hat{p} = \max\left(q \in \{1, \dots, |V|\} \;:\; v_{(q)} \le \frac{q\alpha}{|V|}\right).$$

If $\hat{p}$ does not exist we fail to reject $X_j^T$ as a regular sequence. Otherwise reject the null hypothesis for sequence $j$ and label as irregular.

Output: $X_j^T$ labelled as regular or irregular.
```

ALGORITHM 1 Irregular Activity Sequence Identification

D | TABLES FOR SECTION 4.1

E | EXTRA SIMULATED RESULTS FOR HOUSEHOLDS WITH NO ROUTINES FOR DIFFERENT PARAMETERS

E.1 | Varying the length of $\mathcal{K}$

We now take 56 days as the set of regular sequences and sequence lengths of minimum 4 and maximum 25, from table 6. We vary $\mathcal{K} = (3, 5, 7)$ to see how it affects the results, including $\mathcal{K} = 3$ for ease of comparison. Set $\beta = 1$ and

$\lambda = 0.5$,

Table 20 shows little difference in the false detection results for varying $\mathcal{K}$ in this example. Keeping $\mathcal{K}$ small increases the computational efficiency of the method, so this should be considered in the choice of $\mathcal{K}$.

E.2 | Varying β and λ

We take 56 days as the set of regular sequences and sequence lengths of minimum 4 and maximum 25, from table 6. We vary β and λ, including $\beta = 1$ and $\lambda = 0.5$ for ease of comparison. Table 21 shows little difference overall across the varying parameter values. As you can see the false positive rate and the small change shows a slight decrease as λ increases. However, for the larger change, $(0.6, 0.2, 0.2)$, we can see it increases very slightly and then decreases. As the difference between the results is so minimal, we choose a suitable value in the middle for the results ($\beta = 1$ and $\lambda = 0.5$).

E.3 | Increasing the Number of Sensors

As above, we take 56 days as the set of regular sequences and varying sequence lengths of minimum 4 and maximum 25. We now explore having 4 sensors to see how it affects the results. Set $\beta = 1$ and $\lambda = 0.05$ to ensure the maximal packing constraint holds. The results in the table are similar to the three sensor case. However, the method struggles to pick up the smaller changes. This is due to the small sequence lengths limiting the number of patterns that can be observed. This increases the variability within the reference set which reduces the power of the method to identify small changes.

References

Age UK, . (2018) Later life in the united kingdom. `https://www.ageuk.org.uk/globalassets/age-uk/documents/reports-and-publications/later\_life\_uk\_factsheet.pdf`. Accessed : 2018-08-22.

— (2019) Later life in the united kingdom. `https://www.ageuk.org.uk/globalassets/age-uk/documents/reports-and-publications/later_life_uk_factsheet.pdf`. Accessed : 2019-06-25.

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, **57**, 289–300.

Dafé, G., Veloso, A., Zaki, M. and Meira, W. (2015) Learning sequential classifiers from long and noisy discrete-event sequences efficiently. *Data mining and knowledge discovery*, **29**, 1685–1708.

Davison, A. C. and Hinkley, D. V. (1997) *Bootstrap methods and their application*, vol. 1. Cambridge university press.

Fradkin, D. and Mörchen, F. (2015) Mining sequential patterns for classification. *Knowledge and Information Systems*, **45**, 731–749.

He, Z., Zhang, S. and Wu, J. (2019) Significance-based discriminative sequential pattern mining. *Expert Systems with Applications*, **122**, 54–64. URL: `https://www.sciencedirect.com/science/article/pii/S0957417418308145`.

Höhle, M. (2010) Online change-point detection in categorical time series. In *Statistical modelling and regression structures*, 377–397. Springer.

Hubert, L. and Arabie, P. (1985) Comparing partitions. *Journal of classification*, **2**, 193–218.

Plasse, J. and Adams, N. M. (2019) Multiple changepoint detection in categorical data streams. *Statistics and Computing*, 1–17.

R Core Team (2020) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL: `https://www.R-project.org/`.

Racine, J. S. and MacKinnon, J. G. (2007) Inference via kernel smoothing of bootstrap p values. *Computational Statistics & Data Analysis*, **51**, 5949–5957.

Rand, W. M. (1971) Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, **66**, 846–850.

Taylor, S. A., Killick, R., Burr, J. and Rogerson, L. (2021) Assessing daily patterns using home activity sensors and within period changepoint detection. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **70**, 579–595.

Regular Sequence Probabilities	New Sequence Probabilities	$\lvert R \rvert$	Minimum Sequence Length		
			4	10	25
A: $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	56	0.038	0.02	0.00
	$(0.5, 0.4, 0.1)$		0.918	1	1
	$(0.8, 0.1, 0.1)$		0.93	1	1
A: $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	28	0.058	0.024	0.018
	$(0.5, 0.4, 0.1)$		0.918	1	1
	$(0.8, 0.1, 0.1)$		0.928	1	1
A: $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	14	0.08	0.026	0.042
	$(0.5, 0.4, 0.1)$		0.918	1	1
	$(0.8, 0.1, 0.1)$		0.928	1	1
B: $(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	56	0.022	0.024	0.018
	$(0.7, 0.15, 0.15)$		0.236	0.812	1
	$(0.6, 0.2, 0.2)$		0.928	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
B: $(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	28	0.062	0.036	0.020
	$(0.7, 0.15, 0.15)$		0.354	0.784	1
	$(0.6, 0.2, 0.2)$		0.93	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
B: $(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	14	0.070	0.048	0.032
	$(0.7, 0.15, 0.15)$		0.428	0.758	1
	$(0.6, 0.2, 0.2)$		0.922	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
C: $(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	56	0.036	0.036	0.042
	$(0.8, 0.1, 0.1)$		0.846	1	1
	$(1/3, 1/3, 1/3)$		1	1	1
C: $(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	28	0.070	0.070	0.054
	$(0.8, 0.1, 0.1)$		0.852	1	1
	$(1/3, 1/3, 1/3)$		1	1	1
C: $(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	14	0.082	0.062	0.05
	$(0.8, 0.1, 0.1)$		0.842	1	1
	$(1/3, 1/3, 1/3)$		1	1	1

TABLE 6 Each column represents (from left to right); the probabilities the regular sequences are simulated with, the probabilities the new sequences are simulated with, number of regular sequences and the different minimum lengths of the sequences. The elements of the table contain the proportion which reject at least one hypothesis. Maximum sequence length 25.

Position	1	2	3	4	4	6	7	8	9	10	11	12	13	14	15
Sequence	D	D	D	D	K	M	D	D	D	K	D	K	M	D	D
Position	16	17	18	19	20	21	22	23	24	25					
Sequence	D	K	K	D	D	D	M	K	D	D					

TABLE 7 Regular sequence template used in section 4.2 and 4.3.

New Sequence	$\lvert R \rvert$	Proportion of Sequences labelled Irregular
Simulated Like	56	0.032
Regular Sequences	28	0.044
	14	0.062
Simulated with	56	0.65
Probability $\frac{1}{3}$ Removed	28	0.70
and $\frac{1}{3}$ Replaced	14	0.73
Simulated with	56	0.798
Probability 0.2 Removed	28	0.832
and 0.5 Replaced	14	0.854
Randomly Simulated	56	0.96
with Probabilities	28	0.972
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	14	0.974

TABLE 8 Results for the flexible routine example.Each column represents (from left to right); how the new sequences are simulated, number of regular sequences and the proportion which reject at least one hypothesis.

New Sequence	$\lvert R \rvert$	Proportion of Sequences labelled Irregular
Simulated Like	56	0.048
Regular Sequences	28	0.090
	14	0.112
Probability $\frac{1}{3}$ Removed	56	0.886
and $\frac{1}{3}$ Replaced	28	0.894
	14	0.868
Randomly Simulated	56	1
with Probabilities	28	1
$(1/3, 1/3, 1/3)$	14	1

TABLE 9 Results for strict routine example. Each column represents (from left to right); how the new sequences are simulated, number of regular sequences and the proportion which reject at least one hypothesis.

Sequence: X	D	D	D	D	D				(D , D , 4)
Sequence: Y	D	D	**D**	D	D	D	**D**	D	Total
Maximum Score	β	λ	2λ	3λ					$\beta + 6\lambda$
		β	λ	2λ	3λ				$\beta + 6\lambda$
			β	λ	2λ	3λ			$\beta + 6\lambda$
				β	λ	2λ	3λ		$\beta + 6\lambda$

TABLE 10 Example of a maximum score for maximally packed sequence silhouette instance $(D, D, 4)$.

Sequence: X	D	D	D	D	K				(D , K , 4)	
Sequence: Y	D	D	**D**	D	K	K	**K**	K	Total	c
Maximum Score	β	λ	2λ	3λ					$\beta + 6\lambda$	0
		β	λ	2λ	2λ				$\beta + 5\lambda$	1
			β	λ	λ	λ			$\beta + 3\lambda$	2
				β	0	0	0		β	3

TABLE 11 Example of a maximum score for maximally packed sequence silhouette Instance $(D, K, 4)$. Counter c used in score calculation.

Sequence: X		D	D	D	D	D	D	K						(D , K , 6)
Sequence: Y	Line	D	D	D	D	D	D	K	K	K	K	K	K	Total
Maximum Score	1		λ	2λ	3λ	4λ	5λ							15λ
	2			λ	2λ	3λ	4λ	4λ						14λ
	3				λ	2λ	3λ	3λ	3λ					12λ
	4					λ	2λ	2λ	2λ	2λ				9λ
	5						λ	λ	λ	λ	λ			5λ
	6							0	0	0	0	0		0λ

TABLE 12 Iteration 1 for score calculation of example sequence silhouette $(D, K, 6)$.

Sequence: *X*	D	D	D	D	D	K	K						(D , K , 6)	
Sequence: *Y*	D	D	D	D	D	D	K	K	K	K	K	K	Total	Diff from I1
Maximum Score		λ	2λ	3λ	4λ	4λ							14λ	Line 2
			λ	2λ	3λ	4λ	5λ						15λ	Line 1
				λ	2λ	3λ	3λ	4λ					13λ	$+1\lambda$
					λ	2λ	2λ	2λ	3λ				10λ	$+1\lambda$
						λ	λ	λ	λ	2λ			6λ	$+1\lambda$
							0	0	0	0	λ		λ	$+1\lambda$

TABLE 13 Iteration 2 for score calculation of example sequence silhouette $(D, K, 6)$, the final column represents the difference in λ between Iteration 1 and Iteration 2.

Sequence: *X*	D	D	D	D	K	K	K						(D , K , 6)	
Sequence: *Y*	D	D	D	D	D	D	K	K	K	K	K	K	Total	Diff from I1
Maximum Score		λ	2λ	3λ	3λ	3λ							12λ	Line 3
			λ	2λ	3λ	3λ	4λ						13λ	-1λ
				λ	2λ	3λ	4λ	5λ					15λ	Line 1
					λ	2λ	2λ	3λ	4λ				12λ	$(+1+2)\lambda$
						λ	λ	λ	2λ	3λ			8λ	$(+1+2)\lambda$
							0	0	0	λ	2λ		3λ	$(+1+2)\lambda$

TABLE 14 Iteration 3 for score calculation of example sequence silhouette $(D, K, 6)$, the final column represents the difference in λ between Iteration 1 and Iteration 3.

Sequence: *X*	D	D	D	K	K	K	K						(D , K , 6)	
Sequence: *Y*	D	D	D	D	D	D	K	K	K	K	K	K	Total	Diff from I1
Maximum Score		λ	2λ	2λ	2λ	2λ							9λ	Line 4
			λ	2λ	2λ	2λ	3λ						10λ	-4λ
				λ	2λ	2λ	3λ	4λ					12λ	0λ
					λ	2λ	3λ	4λ	5λ				15λ	Line 1
						λ	λ	2λ	3λ	4λ			11λ	$(+1+2+3)\lambda$
							0	0	λ	2λ	3λ		6λ	$(+1+2+3)\lambda$

TABLE 15 Iteration 4 for score calculation of example sequence silhouette $(D, K, 6)$, the final column represents the difference in λ between Iteration 1 and Iteration 4.

Sequence: *X*	D	D	K	K	K	K	K						(D , K , 6)	
Sequence: *Y*	D	D	D	D	D	D	K	K	K	K	K	K	Total	Diff from I1
Maximum Score		λ	λ	λ	λ	λ							5λ	Line 5
			λ	λ	λ	λ	2λ						6λ	-8λ
				λ	λ	λ	2λ	3λ					8λ	-4λ
					λ	λ	2λ	3λ	4λ				11λ	$+2\lambda$
						λ	2λ	3λ	4λ	5λ			15λ	Line 1
							0	λ	2λ	3λ	4λ		10λ	$(+1+2+3+4)\lambda$

TABLE 16 Iteration 5 for score calculation of example sequence silhouette $(D, K, 6)$, the final column represents the difference in λ between Iteration 1 and Iteration 5.

Sequence: *X*	D	K	K	K	K	K	K						(D , K , 6)	
Sequence: *Y*	D	D	D	D	D	D	K	K	K	K	K	K	Total	Diff from I1
Maximum Score		0	0	0	0	0							0λ	Line 6
			0	0	0	0	λ						λ	-13λ
				0	0	0	λ	2λ					3λ	-9λ
					0	0	λ	2λ	3λ				6λ	-3λ
						0	λ	2λ	3λ	4λ			10λ	$+5\lambda$
							λ	2λ	3λ	4λ	5λ		15λ	Line 1

TABLE 17 Iteration 6 for score calculation of example sequence silhouette $(D, K, 6)$, the final column represents the difference in λ between Iteration 1 and Iteration 6.

Regular Sequence Probabilities	New Sequence Probabilities	$\lvert R \rvert$	Minimum	Sequence	Length
			4	8	10
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	56	0.19	0.29	0
	$(0.5, 0.4, 0.1)$		1	1	1
	$(0.8, 0.1, 0.1)$		1	1	1
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	28	0.02	0.04	0
	$(0.5, 0.4, 0.1)$		1	1	1
	$(0.8, 0.1, 0.1)$		1	1	1
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	14	0.11	0.13	0.08
	$(0.5, 0.4, 0.1)$		1	1	1
	$(0.8, 0.1, 0.1)$		1	1	1
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	56	0.11	0.13	0
	$(0.7, 0.15, 0.15)$		0.44	0.49	1
	$(0.6, 0.2, 0.2)$		0.90	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	28	0.13	0.01	0
	$(0.7, 0.15, 0.15)$		0.46	0.37	1
	$(0.6, 0.2, 0.2)$		0.91	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	14	0.08	0.04	0.02
	$(0.7, 0.15, 0.15)$		0.34	0.40	1
	$(0.6, 0.2, 0.2)$		0.79	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	56	0.13	0.07	0.31
	$(0.8, 0.1, 0.1)$		0.81	1	1
	$(1/3, 1/3, 1/3)$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	28	0.16	0.03	0.07
	$(0.8, 0.1, 0.1)$		0.76	1	1
	$(1/3, 1/3, 1/3)$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	14	0.13	0.17	0.12
	$(0.8, 0.1, 0.1)$		0.77	1	1
	$(1/3, 1/3, 1/3)$		1	1	1

TABLE 18 Each column represents (from left to right); the probabilities the regular sequences are simulated with, the probabilities the new sequences are simulated with, number of regular sequences and the different minimum lengths of the sequences. The elements of the table contain the proportion which reject at least one hypothesis. Maximum sequence length 10.

Regular Sequence Probabilities	New Sequence Probabilities	$\|R\|$	Minimum	Sequence	Length
			4	10	50
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	56	0.03	0.07	0.17
	$(0.5, 0.4, 0.1)$		1	1	1
	$(0.8, 0.1, 0.1)$		1	1	1
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	28	0.09	0.35	0.31
	$(0.5, 0.4, 0.1)$		1	1	1
	$(0.8, 0.1, 0.1)$		1	1	1
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	14	0.28	0.34	0.53
	$(0.5, 0.4, 0.1)$		1	1	1
	$(0.8, 0.1, 0.1)$		1	1	1
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	56	0.05	0.11	0.07
	$(0.7, 0.15, 0.15)$		0.46	0.96	1
	$(0.6, 0.2, 0.2)$		0.98	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	28	0.10	0.19	0.15
	$(0.7, 0.15, 0.15)$		0.68	1	1
	$(0.6, 0.2, 0.2)$		0.99	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	14	0.21	0.21	0.16
	$(0.7, 0.15, 0.15)$		0.83	0.96	1
	$(0.6, 0.2, 0.2)$		0.98	1	1
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	56	0.07	0.09	0.12
	$(0.8, 0.1, 0.1)$		0.99	1	1
	$(1/3, 1/3, 1/3)$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	28	0.14	0.16	0.17
	$(0.8, 0.1, 0.1)$		0.99	1	1
	$(1/3, 1/3, 1/3)$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	14	0.25	0.22	0.19
	$(0.8, 0.1, 0.1)$		0.99	1	1
	$(1/3, 1/3, 1/3)$		1	1	1

TABLE 19 Each column represents (from left to right); the probabilities the regular sequences are simulated with, the probabilities the new sequences are simulated with, number of regular sequences and the different minimum lengths of the sequences. The elements of the table contain the proportion which reject at least one hypothesis. Maximum sequence length 50.

Regular Sequence Probabilities	New Sequence Probabilities	$\|R\|$	$\mathcal{K}$		
			3	5	7
$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	56	0.038	0.038	0.038
	$(0.5, 0.4, 0.1)$		0.918	0.918	0.918
	$(0.8, 0.1, 0.1)$		0.930	0.930	0.930
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	56	0.022	0.028	0.040
	$(0.7, 0.15, 0.15)$		0.236	0.258	0.330
	$(0.6, 0.2, 0.2)$		0.928	0.928	0.922
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1
$(0.65, 0.25, 0.1)$	$(0.65, 0.25, 0.1)$	56	0.036	0.036	0.042
	$(0.8, 0.1, 0.1)$		0.846	0.838	0.834
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		0.994	0.994	0.994

TABLE 20 Changing k for varying sequence lengths between 4 and 25. Each column represents (from left to right); the probabilities the regular sequences are simulated with, the probabilities the new sequences are simulated with, number of regular sequences and the different minimum lengths of the sequences. The elements of the table contain the proportion which reject at least one hypothesis.

Regular Sequence Probabilities	New Sequence Probabilities	$\|R\|$	(β, λ)				
			(1,0.1)	(1,0.25)	(1,0.5)	(1,0.75)	(1,0.9)
$(0.8, 0.1, 0.1)$	$(0.8, 0.1, 0.1)$	56	0.024	0.022	0.022	0.022	0.022
	$(0.7, 0.15, 0.15)$		0.248	0.236	0.236	0.230	0.230
	$(0.6, 0.2, 0.2)$		0.924	0.926	0.928	0.922	0.922
	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$		1	1	1	1	1

TABLE 21 Changing β and λ for varying sequence lengths between 4 and 25. Each column represents (from left to right); the probabilities the regular sequences are simulated with, the probabilities the new sequences are simulated with, number of regular sequences and the different β and λ. The elements of the table contain the proportion which reject at least one hypothesis.

Regular Sequence Probabilities	New Sequence Probabilities	$\lvert R \rvert$	Minimum Sequence Length 4	10	25
$(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$	$(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$	56	0.044	0.040	0.000
	$(0.5, 0.3, 0.1, 0.1)$		0.860	1	1
	$(0.7, 0.1, 0.1, 0.1)$		0.882	1	1
$(0.7, 0.1, 0.1, 0.1)$	$(0.7, 0.1, 0.1, 0.1)$	56	0.010	0.026	0.014
	$(0.6, 0.15, 0.15, 0.1)$		0.076	0.378	1
	$(0.5, 0.2, 0.2, 0.1)$		0.502	0.988	1
	$(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$		0.998	1	1
$(0.6, 0.3, 0.05, 0.05)$	$(0.6, 0.3, 0.05, 0.05)$	56	0.026	0.040	0.044
	$(0.7, 0.1, 0.1, 0.1)$		0.906	1	0.914
	$(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$		0.998	1	0.996

TABLE 22 4 sensors for varying sequence lengths between 4 and 25. Each column represents (from left to right); the probabilities the regular sequences are simulated with, the probabilities the new sequences are simulated with, number of regular sequences and the different lengths of $\mathcal{K}$. The elements of the table contain the proportion which reject at least one hypothesis.