Automatic Detection of Anomalies in Blood Glucose Using a Machine Learning Approach

Ying Zhu
Faculty of Business and Information Technology
University of Ontario Institute of Technology
2000 Simcoe Street North, Oshawa, Ontario, Canada, L1H 7K4
1-905-721-8668
ying.zhu@uoit.ca

Abstract

Rapid strides are being made to bring to reality the technology of wearable sensors for monitoring patients' physiological data. We study the problem of automatically detecting anomalies in the measured blood glucose levels. The normal daily measurements of the patient are used to train a hidden Markov model (HMM). The structure of the HMM — its states and output symbols — are selected to accurately model the typical transitions in blood glucose levels throughout a 24-hour period. The learning of the HMM is done using historic data of normal measurements. The HMM can then be used to detect anomalies in blood glucose levels being measured, if the inferred likelihood of the observed data is low in the world described by the HMM. Our simulation results show that our technique is accurate in detecting anomalies in glucose levels and is robust (i.e., no false positives) in the presence of reasonable changes in the patient's daily routine.

1 Introduction

Significant technological advances in wearable sensors and wireless sensor networks are offering the realization of remote personal health monitoring of patients [9, 7, 8, 12]. In particular, for individuals with diabetes, a number of new non-invasive technologies are being developed to read the blood glucose level without access to the blood, using near infrared, ultrasound and dielectric spectroscopy, e.g., [3, 4, 5]. These types of glucose sensors not only avoid the (possibly painful) piercing of the subject's finger tip, but also offer the advantage of continuous glucose monitoring, even while the subject is sleeping.

These sensors can be deployed as part of a wireless sensor network. Our model of the wireless sensor network for medical monitoring is as follows. There are two types of nodes: the sensing nodes and the central nodes. The sensing nodes measure the physiological data; they have tiny onboard processors that can carry out simple pre-processing of data locally and transmit only required data at pre-set intervals, e.g., one reading every half hour, to the central node. A central node is fixed in the infrastructure (e.g., installed on a wall in the house), and responsible for integrating and analyzing the sensed readings received. The central node is also responsible for triggering medical alerts if it detects anomalies in the sensor data. The central node may be a basestation or can communicate with a basestation via the network infrastructure, therefore as soon as alerts are produced by the central node, the appropriate parties (e.g., nurse, doctor) will be notified.

We are interested in the problem of automatically detecting anomalies in the sensed blood glucose levels so that medical alerts will be triggered whenever and only when the readings are abnormal. Whether the readings are abnormal should not depend only on the absolute numerical values, since the blood glucose levels that are considered normal can vary from person to person. Even for the same individual, the glucose level increases and decreases throughout the day, due to various factors such as food intake and sleep. A good anomaly detection scheme should detect glucose levels that are abnormal for each individual based on the *historical* readings for this person. We develop such a scheme by designing a hidden Markov model (HMM) for glucose levels throughout the day, and the HMM is trained automatically for each individual by learning its parameters using historical glucose readings for this person.

Our simulation results show that our scheme is both effective in detecting anomalies in glucose levels when they occur, and reliable in not producing false positives (a false positive is identifying an anomaly when it is not abnormal for this individual based on his/her history). Further-

	Fasting (before meals) blood glucose (mmol/L)	After-eating blood glucose (mmol/L)
Normal range	4.0 - 6.0	5.0 - 8.0
Target for most patients with diabetes	4.0 - 7.0	5.0 - 10.0

Table 1. Blood glucose levels from the Canadian Diabetes Association website [1]

more, our method demonstrates robustness in the presence of moderate changes in the daily schedule of the monitored individual.

2 Related work

To the best of our knowledge, there is no previous work on automatic anomaly detection in blood glucose sensor readings. There have been relevant studies in the area of medical monitoring using wireless sensor networks. In [6, 8], the authors propose methods to automatically detect a patient's abnormally long inactivity at home in order to trigger timely alerts in the case of medical emergencies where the patient is rendered immobile and/or unconscious. For the same purpose of monitoring patients and raising emergency alerts, an architecture for a medical sensor network is proposed in [12]. The focus of [11] is on monitoring of people in an emergency site. The authors in [13, 7] present algorithms that make use of wearable accelerometer sensors to detect the occurence of a fall and the location of the victim. In contrast to our work, these work do not address the specific problem of monitoring and detecting anomalies in blood glucose readings; moreover, they do not adopt a machine learning approach as we do in this paper.

3 Automatic glucose anomaly detection

A person's blood glucose levels are highly dynamic over time: they are affected by a variety of factors including recent food intake, insulin doses, metabolic activity, and the time of day. Table 1 is taken from the Canadian Diabetes Association website [1]; it shows the normal range of blood glucose levels and the target range for persons with diabetes. For both normal and diabetic ranges, there is a clear distinction between glucose levels after and before a meal. We will use the term *Post-meal* to refer to the time period after a meal when the glucose level rises, and the term *Fasting* for the lower-glucose duration starting right after *Post-meal* until the next meal.

We obtain a graph [2] of the fluctuation of blood glucose in humans during a 24-hour period with three meals, based on the clinical nutrition study [10], see Fig. 1. From this plot, we observe that there are three distinct states throughout a day: *Meal*, *Fasting*, and *Sleeping*. In each state, the glucose behaviour is very different: *Meal* is a high, sharp peak; *Fasting* is a valley with a lower and wider bump; *Sleeping* slopes down to a near-constant low. Thus, to detect anomalies in patients' blood glucose levels, one cannot simply impose absolute thresholds for upper and lower bounds. Not only because the normal range differs from person to person, but even for the same person, naive thresholding does not work. For example, if an upper threshold of 8 is chosen, it can catch abnormally high glucose levels right after a meal, but it will miss anomalies such as the glucose remaining high at 7 between meals or during night sleep.

Our approach is to build a hidden Markov model (HMM) to describe or model the sensor readings of blood glucose level, by taking into account the existence of the three distinct states. Next, an algorithm learns the model parameters from some training data, so that the learned HMM profiles normal daily glucose levels. This HMM can then be used to test whether new sensor glucose readings are anomalous: an inference algorithm can find the probability of that a sequence of new readings are produced by the HMM — if the probability is high, then the new readings are normal because they are supported by the HMM that profiles the norm; if the probabiliy is low, then an anomaly is detected.

3.1 Hidden Markov models

A discrete (first-order) Markov chain or process is in one of a set of N states at any time step t. Let X_t denote the state at time t. The Markov process is governed by two assumptions: Markov and stationary. The Markov assumption is that the current state X_t depends only on the previous state X_{t-1}. The stationary assumption is that the laws of changes from state to state do not themselves change over time (even though the states change over time), i.e., the conditional probability $P(X_t|X_{t-1})$ is independent of t. In summary, for all t, $P(X_t|X_{0..t-1}) = P(X_t|X_{t-1})$, and the laws dictating how the state evolves over time are entirely contained within the conditional probability $P(X_t|X_{t-1})$.

In the hidden Markov model, the state of the Markov model is described by a single discrete random variable, whose possible values are the possible states of the world. Let O_t denote the observation or output that is seen at each time step t, which is dependent only on the state at t, X_t. Each observation is affected (only) by the current state of the world — the state causes the observation or output to take on particular values — with the conditional probability distribution: $P(O_t|X_t)$. So the state of the world we are modeling is hidden from observation. We see only the O_t's and we have knowledge of the conditional distribution

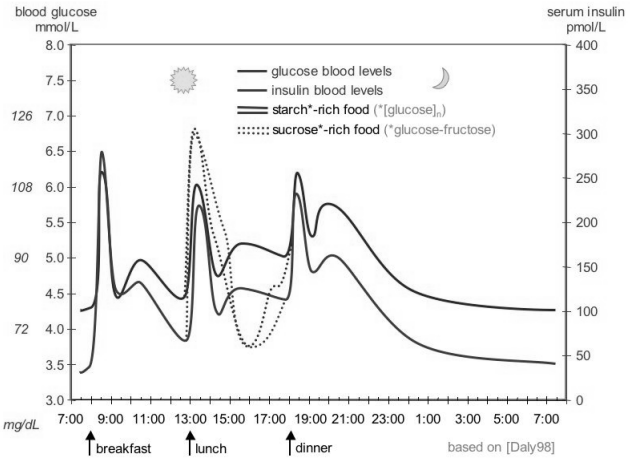

Figure 1. The top solid curve represents the fluctuation of blood glucose concentration in humans during the course of a day with three meals.

$P(O_t|X_t)$.

An HMM is thus characterized by the following elements:

- The number of states, N; i.e., there are N possible values for X_t.
- The number of distinct observation or output symbols, M; i.e., there are M possible values for O_t.
- The state transition probability distribution given by the matrix T: $T_{ij} = P(X_t = i|X_{t-1} = j)$.
- The observation probability distribution for an observation k is the diagonal matrix, B^k: $B^k_{ii} = P(O_t = k|X_t = i), 1 \leq i \leq N, 1 \leq k \leq M$.
- The initial probability distribution $P(X_0)$ over the states at time 0, denoted P_0.

Let $H = (T, B, P_0)$ denote an HMM with all its parameters.

3.2 Automatic detection of glucose anomalies

In order to build an HMM to model a person's dynamic glucose levels, we must go through the steps:

1. Determine the set of states and the number of states.
2. Determine the set of observation or output symbols.
3. Learning of the HMM parameters — use training data to optimize the model parameters so the resulting HMM best describes the given observation sequence.

We now explain each step in detail.

3.2.1 The states

Motivated by the pattern we observe in Fig. 1, we model a person's fluctuating glucose values by a 3-state HMM with the three states: *Fasting*, *Meal*, and *Sleep*. The state transition model is shown in Figure 2. The state *Sleep* is during overnight sleeping. The state *Meal* is for the short time period right after a meal is taken — it basically cosists of the high peaks (glucose rising then dropping sharply). The state *Fasting* is the time between meals — it corresponds to the gentle low bumps between the *Meal* peaks. An arrow from state i to state j is labeled with the transition probability $P(X_t = j|X_{t-1} = i)$ (in the diagram, the shorthand $P(j_t|i_{t-1})$ is used). To model reality as closely as possible (as it is depicted by Fig. 1), in our model we allow transitions in both directions between *Fasting* and *Meal* and self-loops on each state, but there are no transition from *Meal* to *Sleep* nor from *Sleep* to *Fasting* — these correspond to the reasonable assumptions that one eats some food after waking up and does not go into deep night-sleep immediately after dinner.

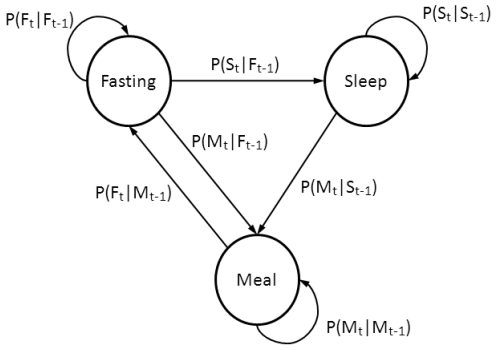

Figure 2. The state transition diagram of the HMM. Shorthand notation is used for the transition probabilities labeling the arrows between states: $P(j_t|i_{t-1})$ means $P(X_t = j|X_{t-1} = i)$; e.g., $P(M_t|F_{t-1}) = P(X_t = M|X_{t-1} = F)$.

3.2.2 Observation symbols

The observation or output at time t is the sensor reading of the blood glucose level at t. It is reasonble to set the sensor reading intervals to τ minutes (τ could be set to any number, we choose $\tau = 30$, so an observation is generated every half an hour). Every day yields a sequence of L glucose readings (e.g., for $\tau = 30$, $L = 48$). We first considered using these glucose readings directly as the observation symbols. However, on second thought, this does not seem appropriate. To see why, let us examine the glucose curve in Figure 1 (the top solid curve): consider the first *Meal* state that spans from around 07:30 to 09:30 and the immediately-following *Fasting* state approximately in [09:30 – 12:30]. Note that some of the glucose levels in the *Meal* state coincide with some of those in the *Fasting* state, e.g., glucose is the same at both 08:30 and 10:30, and at both 09:30 and 12:30. In fact, for each pair of states, there are coinciding glucose values. Since in the HMM, the current state causes particular observation symbols to be seen, it is highly ambiguous in this case which state is more likely to cause specific glucose values.

To reason this through, the observation symbols should be chosen so that the symbols emitted by each state somehow characterize this particular state, that is, they can be used to distinguish this state from other states. From the empirical data in [10], as seen in Figure 1, it is evident that a distinguishing measure of each state is the degree of dynamics in the glucose values. During the *Meal* state, the values exhibit the most dynamics: the difference between the lowest and the highest value is the greatest; in *Sleep*, there is hardly any change in the values, while *Fasting* is somewhere in between. We also observed that a window of one hour is large enough to capture the entire up-and-down dynamics in the *Meal* state. Therefore, we use the *maximum difference between glucose values* within each one-hour window as the observation symbols in our model. The one-hour window is of course shifted by τ minutes at each step, since the readings are done every τ minutes. The maximum difference within each one-hour window is calculated by taking the difference between the highest and the lowest value in that window.

3.2.3 Learning the HMM

The initial distribution P_0, basically the belief of which state the world is in at the very beginning (i.e., time 0), can simply be set to the uniform distribution, $[1/3, 1/3, 1/3]$ in this case.

It remains to learn the HMM from historic evidence — the evidence is called training data. The learning problem is essentially to determine the model parameters (T, B) that maximizes the likelihood of the training data given the model. The training data is the observation symbols obtained from the patient's normal daily glucose readings. Thus, the learned or trained HMM describes the normal glucose profile of this patient. For the HMM learning, we use the well-known *Baum-Welch* algorithm that belongs to the family of Expectation Maximization algorithms. It is an iterative algorithm that starts with an initial HMM and in every iteration, computes a new HMM based on the old HMM from the previous iteration, using the given observation sequence. The algorithm always converges and finds local maxima. Since local maxima are found (there is no known method to find the global maximum), we were careful in our choice of initial distributions (T, B). The initial transition probability distribution T included the near-zero probability of transitioning from *Sleep* state to *Fasting* state and from *Meal* state to *Sleep* state. The initial observation distribution included high probabilities for low observation symbols (recall low means low dynamics or small changes in the glucose readings) in the *Sleep* state and near-zero likelihood that high observation symbols are seen in *Sleep*.

3.3 Automatic anomaly detection

Once the learning stage is finished, the trained HMM now profiles normal glucose readings throughout the day. To automatically detect anomalies, the glucose readings received from the sensors are evaluated based on the HMM to compute the probability of these observations in the HMM model. The *Forward-Backward* algorithm [14] is used to compute this probability. If the probability is high, then the readings are normal. If the probability is too low, then

an anomaly has been detected. The specific values used in making this classification decision are determined from experimental data; the details are given in the next section.

4 Evaluation

We implemented our automatic glucose detection method in Python. The generation of the training glucose data for learning the HMM is done as follows. We obtain from [10] the normal glucose data: (1) a set of mean glucose values Z at half-hour marks throughout the day (at 00:00, 00:30, 01:00, . . ., 23:00, 23:30), and (2) the confidence intervals for these values. To generate 1-day of training data, each data value from Z is added with a random number within the corresponding confidence interval. This way, we can generate any number of days of training glucose data. We also simulate glucose readings of 100 days that contain anomalies.

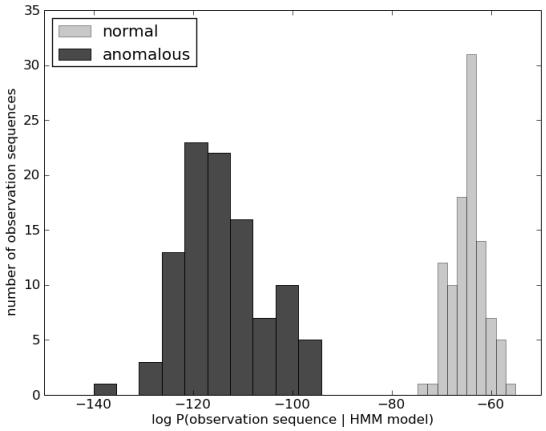

Figure 3. Histogram of log probability distribution of 100 days of normal glucose readings versus glucose readings containing anomalies

To evaluate the effectiveness of the trained HMM in detecting anomalies, the likelihood of the training data given the HMM are compared against the likelihood of the anomalous data given the HMM. The histograms of the log probabilities for both sets of data are shown in Figure 3. In this case, both data sets have 100 days of glucose readings. The y-axis is the number of observed sequences, each sequence is one day of glucose readings. It is evident that there is a clear and substantial gap between the likelihood of anomalous and normal glucose readings — hence we can simply set the classifier threshold to the log probability value in the middle of the gap, which is approximately -95.

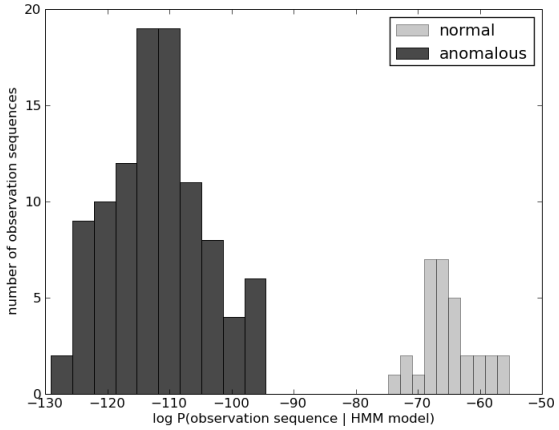

Figure 4. Histogram of log probability distribution of 30 days of normal training data versus anomalous data

So if the log probability of new glucose readings given the trained HMM is less than -95, then an anomaly is detected; otherwise they are classified as normal.

Reducing the number of days for collecting historical observations for training data does not decrease the effectiveness of the trained HMM in classifying anomalous readings, as shown in Figure 4. For this experiment, we obtain only 30 days of readings for training the HMM. The likelihood distribution of the normal data looks almost identical to that of the 100-day set. Most importantly, they are as distinct and distant from the anomalous data set as in the 100-day case; in other words, 30 days of historical observation is sufficient for learning the HMM that will be capable of detecting anomalies.

We also evaluate the ability of the HMM to still work in the realistic scenarios where the sleep/meal schedule of the patient has moderate changes, such as eating and sleeping later than usual. After the HMM is learned from the training data, new data is generated that correspond to a shift of up to two hours in the sleep/meal schedule, again for 100 days: for each day, a random shift between +2 and -2 hours is introduced into the glucose readings, to simulate realistic changes in schedule. The shifted data is evaluated for its likelihood given the HMM and compared with the normal and anomalous data, in Figure 5. It can be seen that the shifted data is still classified — correctly — as normal and is obviously distinguished from anomalous readings.

5 Conclusion

We have presented a machine learning method for automatically detecting anomalies in a person's blood glucose

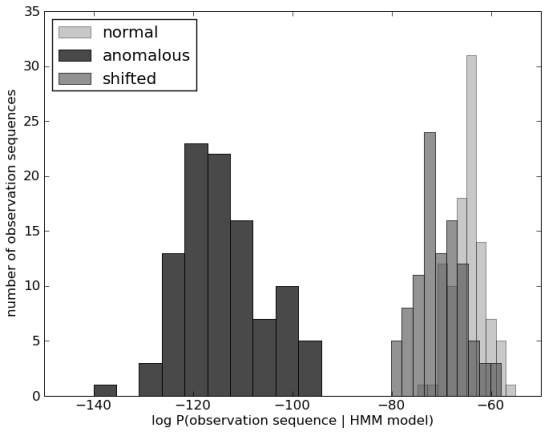

Figure 5. Histogram of log probability distribution of normal (training) data, normal but with a random shift (between -2 and +2 hours) in the schedule, and anomalous glucose readings

levels, using historical observations as a benchmark. The method is shown in simulations to be effective and robust.

References

[1] `http://www.diabetes.ca/about-diabetes/living/management/manage-glucose/`.

[2] `http://en.wikipedia.org/wiki/Blood_sugar`.

[3] Glucoband. `http://www.calistomedical.com/?cat=14`.

[4] Glucotrack. `http://www.integrity-app.com/`.

[5] Sensys glucose tracking system. `http://www.sensysmedical.com/technology/index.html`.

[6] T. Ryan Burchfield and S. Venkatesan. Accelerometer-based human abnormal movement detection in wireless sensor networks. In *HealthNet '07: Proceedings of the 1st ACM SIGMOBILE international workshop on Systems and networking support for healthcare and assisted living environments*, pages 67–69, New York, NY, USA, 2007. ACM.

[7] J. Chen, K. Kwong, D. Chang, J. Luk, and R. Bajcsy. Wearable sensors for reliable fall detection. In *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005.

[8] Paul Cuddihy, Jenny Weisenberg, Catherine Graichen, and Meena Ganesh. Algorithm to automatically detect abnormally long periods of inactivity in a home. In *HealthNet '07: Proceedings of the 1st ACM SIGMOBILE international workshop on Systems and networking support for healthcare and assisted living environments*, pages 89–94, New York, NY, USA, 2007. ACM.

[9] Akyildiz et al. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.

[10] Daly et al. Acute effects on insulin sensitivity and diurnal metabolic profiles of a high-sucrose compared with a high-starch diet. *The American Journal of Clinical Nutrition*, 67:1186–1196, 1998.

[11] Gao et al. Wireless medical sensor networks in emergency response: Implementation and pilot results. In *2008 IEEE Conference on In Technologies for Homeland Security*, 2008.

[12] Wood et al. Context-aware wireless sensor networks for assisted-living and residential monitoring. *IEEE Network*, 22(4):26–33, 2008.

[13] M. Mathie, J. Basilakis, and B.G. Celler. A system for monitoring posture and physical activity using accelerometers. In *2001 IEEE Engineering in Medicine and Biology 23rd Annual Conference*, 2001.

[14] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, 2nd edition, 2003.