Practical Machine Learning for Diabetes Care

Sam Royston

Abstract—Diabetes patients must monitor and control their own blood glucose levels with the intention of approximating the blood glucose levels and dynamics of a typical person. This is an onerous task for most diabetics and this paper will survey a handful of ways that these challenges might be practically addressed with Machine Learning. The regressive versions of AdaBoost and SVMs are evaluated for feasability for blood sugar prediction in the context of intermittent data. The SVM and AdaBoost classification methods are also evaluated for their performance in predicting hypoglycemic events, which are harmful if left untreated. Possible applications of these methods are presented along with a discussion of future research topics and the many factors effecting the success of diabetes treatments.

1 Introduction

Type-1 Diabetes is an autoimmune disease that affects up to 3 million americans and many more overseas. The disease is caused by the immune system attacking the body's own pancreas and compromising it's ability to produce the hormone insulin, which is neccesary to metabolize carbohydrates. The disease is chronic and treatment consists of constant monitoring of blood glucose (BG) levels and administration of insulin via injection or subcutaneous catheter (insulin pump). In a person without diabetes, the body is able to maintain very tight control on blood glucose levels by releasing insulin in real time in response to BG changes. It has been shown that in type-1 diabetics, good blood glucose control can help prevent complications associated with the disease. A high average blood glucose (chronic hyperglycemia) can cause complications such as blindness, amputation, and heart failure. Low blood glucose, or hypoglycemia, results in temporarily diminished brain function and the release of adrenaline from the liver, which can be quite disorienting to the patient. If left untreated for an extended period of time Hypoglycemia can result in a seizure, coma, or death. For type-1 diabetics BG control is solely dependent on their ability to interpret, predict, and respond to past BG readings. Techniques that can aid in the prediction and ultimately the control of blood glucose levels will be of great help to type-1 diabetes patients.

Another quality of diabetes care that has informed the construction of this study is its inordinate cost. Diabetes care accounts for 14.9 billion in health care costs in the United States and with every incremental advance in the state of the art, costs are pushed even higher. Especially in impoverished nations, many with diabetes may not have access to the most basic supplies [8]. It is clear that even in the face of significant technological advances in the developed world, such as the continuous blood glucose monitor discussed below, cost cutting measures will play a central role in the global treatment of the disease for years to come.

The continuous glucose monitor (CGM) is a device that takes blood sugar measurements on the order of every 2 minutes. This is in stark contrast to the method of finger-sticking, which is the norm in diabetes care and is performed anywhere from 3 - 10 times daily. The advent of internet connected mobile devices in addition to advances such as the CGM introduce the possibility of analyzing BG results in real time and quickly offering actionable data.

The intention of this paper is to explore new applications of machine learning in the domain of diabetes care that caters to one or more of the following goals, each related to practicality in the real world

1) Use data from economical hardware, or in a way such that cost is minimized
2) Require minimal additional effort from the patient
3) "Get on the bandwagon" of technological

advances which will continue to improve independently of events in the field of health care; i.e. smartphones, cloud services... etc.

In an attempt to address the 1) and 2) we assess the extent to which BG data taken at typical intervals (3-10 times daily) can be used to make predictions about future blood sugars without any external information about insulin dosages and food intake. Futhermore, we discuss how we might use prior CGM data to aid in this process. For example: what useful information can be gleaned while a patient is given a CGM on loan? Due to the dynamics of health care reform and certain economic realities, it is the author's belief that a majority of diabetics will be using finger-stick methods for years to come, and this majority will persist in spite of advances such as the CGM.

2 DATASETS

In this survey three different datasets were used with the hope of learning about the behavior of learning algorithms in three contexts.

2.1 CGM Data

52514 BG measurements from a 14 year old male undergoing CGM therapy over a course of roughly 7 months. In this dataset the rate of BG measurements is not constant (ranging anyhere from every 2 minutes to every 10 minutes), and at times there are long ($\sim$ 4 hr) breaks in data ostensibly when the device is being replaced.

2.2 Finger-Stick Data

2.2.1 Comprehensive logging data

We used data from the UCI machine learning database [5] which consisted of 5064 blood sugar measurements taken at what we will consider in this study as a normal rate ($\sim$ 3-5) measurements daily. The UCI database also contains a variety of other recorded patient data, like insulin dosages, food intake, and exercise.

2.2.2 Self-Collected Data

Lastly, some experiments were done on a dataset collected by the author, which consists of 220 blood glucose measurements (and growing) taken at a rate of approximately $5-8$ times per day.

3 PRIOR WORK

The topic of applying machine learning to diabetes data is not new, but it has seen a recent uptick in interest due to the data that will be offered by more widespread usage of CGM devices and connected meters. It is the ultimate goal of many researchers to "close the loop" in glucose control by implementing an automated system that actively monitors BG and delivers insulin in real time. To that end, many current publications focus on CGM data, whereas this study is more concerned with the ability to make decisions about Fingerstick-Data.

Recent work by Marling et. al. [10] uses features derived from an underlying physiological model [7] to train an Support Vector regressor. This proves to be a good approach in terms of feature extraction, in fact outperforming some doctors in prediction tasks.

Jensen et. al. [6] chose features from a set of 2289 potential features defined by statistical measures (Linear regression, Skewedness, Kurtosis) taken over many intervals of the preceding data. A choice of 7 from this set was made based on a Separability and Correlation, (SEPCOR) analysis.

Both of these studies give results with high accuracy, but are designed specifically for CGM data and use features that are dependent on the density of CGM data and the input of other information such as Insulin dosages. Despite the inconsistencies with our experimental setup, these publications still offer valuable insight into the problem.

4 METHODS AND PROCEDURES

The toolkits numpy, scipy, and scikit-learn [3] [2] [1] were chosen to do the following data processing, and the scikit-learn functions `ensemble.AdaBoostCLassifier`, `ensemble.AdaBoostRegressor`, `tree.DecisionTreeClassifier`, `tree.DecisionTreeRegressor`, `svm.SVR`, `svm.SVC` were used as the learners. Along with the author's familiarity with these software tools, they were also selected because of their easy integration into server-side software applications should they be deemed effective. Below is a table describing the dataset and research question pairings that were explored in some capacity through this study.

Dataset	**Methods Used**	**Problem Types**
UCI (2.2.1)	AdaBoost, SVM	Hypoglycemia Classification
CGM (2.1)	AdaBoost, SVR	BG Prediction
Self-Collected (2.2.2)	AdaBoost, SVM, SVR	Hypoglycemia Classification, BG Prediction

4.1 Features

The original intention had been to leverage rich data from CGM devices to make predictions about fingerstick data, so features were designed with the purpose of being applicable to all three datasets. Thus the description the of these features is in terms of an array of BG measurements taken at a rate resembling fingerstick data. Let this array be called D. The BG measurement associatd with the feature vector D_i is denoted $m(D_i)$, and the time of measurement, as a unix time-stamp, is written as $t(D_i)$. Given this initial data, the feature vector f_i is constructed as follows:

$$f_i = \Big[m(D_{(i-1,i-3)}), \triangle t(D_{(i-1,i-3)}), t_d(D_i), \ldots$$

$$\ldots \sigma(\triangle_{pair} t(D)), \Sigma(\triangle_{pair} t(D)), \sigma(m(D)), \Sigma(m(D))\Big]$$

Where

$$m(D_{(i-1,i-3)}) = m(D_{(i-1)}), m(D_{(i-2)}), m(D_{(i-3)})$$

$$\triangle t(D_{(i-1,i-3)}) = t(D_i) - t(D_{i-1}), t(D_i) - t(D_{i-2}), \ldots$$

$$\ldots t(D_i) - t(D_{i-3})$$

$$t_d(D_i) = t(D_i) mod(86400)$$

$\triangle_{pair} t(D)$ is the set of pairwise differences in measurement time and $\sigma(\triangle_{pair} t(D))$ is the set of rolling standard deviations of $\triangle_{pair} t(D)$ taken over the past 3,5,10, and 20 samples. Similarly $\Sigma(\triangle_{pair} t(D))$ is the set of moving averages taken over the same intervals. $\sigma(m(D))$ is the set of rolling standard deviations of BG measurements over those intervals, and $\Sigma(m(D))$ are the corresponding rolling averages. Since $t(D_i)$ is a unix-timestamp, when taken modulo the number of seconds in a day, the result is a feature repsesenting the time of day: $t_d(D_i)$.

As for this choice of features, the reasoning was based on a number of factors listed below:

- Lagged-Values are the simplest possible feature selection for time series data. Also their performance is not neccessarily bad in spite of their simplicity [4]. We only look back three steps in this setting because it is questionable whether BG measurements from more than 3-4 hours in the past have a discernable effect on the current blood sugar.
- Moving-averages over different intervals could be an effective way to describe longer term trends preceding the relevant measurement, and have been shown to be effective in time-series prediction [4]
- $\triangle t(D_{(i-1,i-3)})$ is selected because it will differentiate features along these three axis as time passes, even if there are no new measurements. This allows for continuous reupdate of the results, but may also lead to some strange geometric qualities of the training data.
- The rolling standard deviation components are intended to characterize the volatility of the user's control, and whether it is increasing or decreasing leading up to the relevant measurement.

4.2 Preprocessing

In order to coerce the data into a form that could generate the features listed in section

4.1, we had to make sure that there was some sequence D to work with.

With respect to the CGM data, this meant taking a sample that would share some characteristics of finger-stick data. This was acheived in a somewhat naive fashion by taking a random sample from the CGM data at a density equivalent to the density of the self-collected blood glucose measurements.

Once our set of features was defined, we scaled all values (by column) to lie within $[0, 1]$, because the support vector model is not scale invariant.

5 Training and Results

A number of parameters need to be optimized for use in both AdaBoost and the Support Vector models, namely γ, C, n_{cl} and h. Where γ is simply another formulation of σ in the (σ, C) parameter pair associated with SVMs: $\sigma = \sqrt{\frac{1}{\gamma}}$. n_{cl} represents the number of base classifiers used by the AdaBoost model, and h is the maximum height decision tree allowed as a base classifier. We optimised these values via cross validation. Below are some selected grid search results

5.1 UCI Data

The UCI dataset was the primary dataset used for testing the potential of detecting hypoglycemic events. The one 10th of the dataset was designated as the test set, while the rest was used in 2-fold cross validation (Due to time constraints). There was some concern that this non-i.i.d. data would yield inflated cross-validation results if shuffled, so we performed this procedure with shuffled data and non-shuffled data and found there to be no significant difference, although problems may have been revealed with a greater number of cross validation folds.

A major issue with low blood sugar classification is that hypoglycemic measurements only account for some minority of the total measurements. In the UCI dataset, roughly 10% of all readings were less than 68 mg/dl, which we designate as "low". Some measures had to be taken to account for this, otherwise most classifiers would end up labeling everything false and receive an accuracy rating of $\sim 90\%$. Therefore class weights were assigned to be inversely proportional to the frequency of the hypoglycemic measurements. Furthermore, these weights were factored into the accuracy measuremnts used to asses performance.

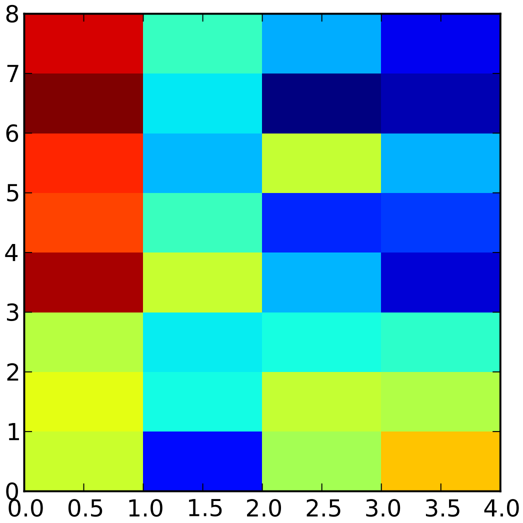

Fig. 1. Grid Search for AdaBoost classifier training on UCI data, yielding $n_{cl} = 700$ and $h = 1$. (The y-axis is $\frac{n_{cl}}{100}$)

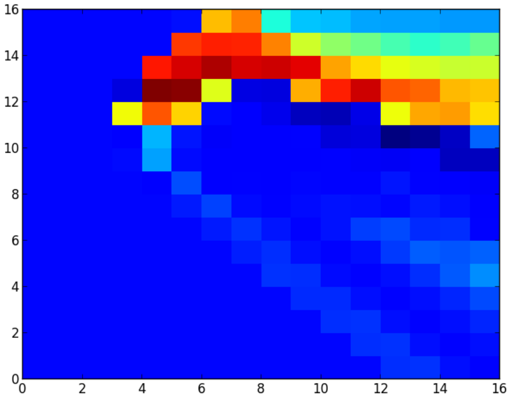

Fig. 2. Grid Search for SVM on shuffled UCI data, yielding $\gamma = 16$ and $C = 0.125$. Color is shown as a function of measured accuracy and the axes scale exponentially from 2^{-8} to 2^{8}

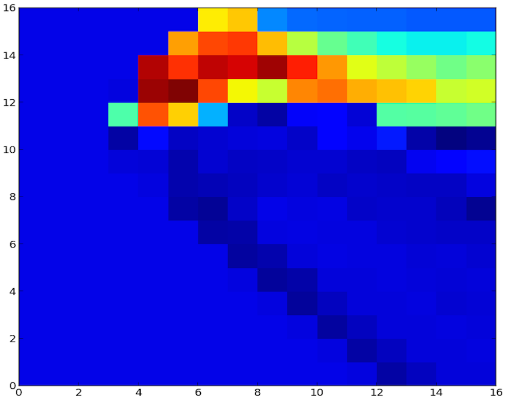

Fig. 3. Grid Search for SVM on non-shuffled UCI data, yielding $\gamma = 16$ and $C = 0.125$. Color is shown as a function of measured accuracy and the axes scale exponentially from 2^{-8} to 2^{8}

	AdaBoost	**SVM (rbf)**
Accuracy (Test)	0.53	0.55
Precision (Test)	-	0.12
Recall (Test)	-	0.67
f1-Score (Test)	-	-
Accuracy (Cross-Val)	0.61	0.59
Precision (Cross-Val)	-	0.22
Recall (Cross-Val)	-	0.62
f1-score (Cross-Val)	-	0.26

5.2 CGM Data

For the CGM dataset, it was neccesary to use the down-sampling procedure described earlier. This differentiated these features dramatically from the other datasets and based on a few tests it was easily seen that this precluded the possibility of interoperability for forward looking predictions, i.e. training on the CGM data and testing on some other fingerstick dataset. One factor to consider is that each dataset was taken from a separate patient, and there may be some things about the CGM data that must be "unlearned" before applying it elsewhere.

	R^2
AdaBoost *lookahead* $(n_{cl}, h) = (40, 20)$	0.43
SVR *lookahead* $(\gamma, \epsilon, C) = (0.7, 7, 500)$	0.21

	R^2
AdaBoost *no-lookahead* $(n_{cl}, h) = (40, 20)$	0.19

With CGM data it was possible to validate predictions at a much higher rate, and thus for every interval between test points we could track the performance of our regressor against the actual behaviour of the patient's blood glucose. Indeed, without more information about food intake and insulin administration this problem is in some sense ill-posed, but the AdaBoost regressor was still able to accurately model the behavior of the underlying BG, directly following a measurement. Thus it follows that this same regressor can accurately model the underlying BG between two known measurements separated by a reasonable amount of time. Because of this, we also ended up assessing the performance of an identical system with the ability to look one measurement ahead into the future.

Surprisingly, in this context, the top performing model was an AdaBoost Regressor with what seem to the author to be abnormally deep decision trees $(h = 20)$.

6 Discussion and Future work

The majority of the tested methods performed with very low accuracy, and for the low blood sugar identification task, approached chance. Despite this, it is important to recognize the difficulty of of these tasks and the limited

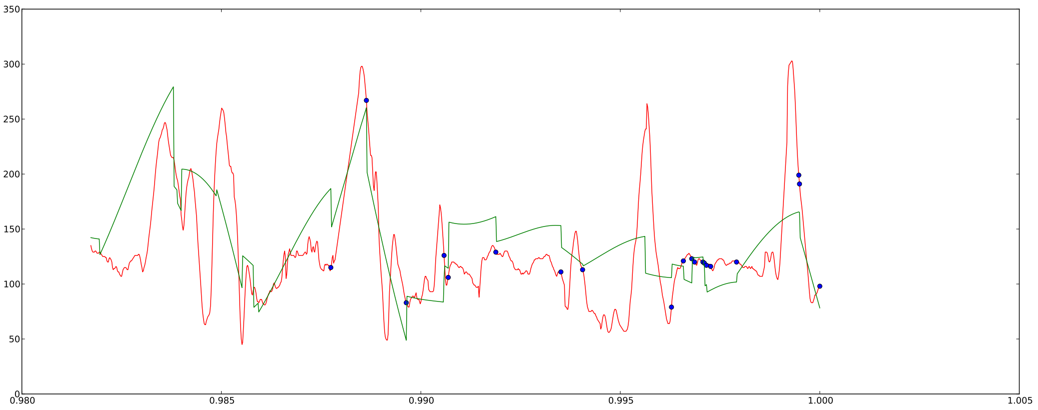

Fig. 4. Above is an SVR model (green line) intended to predict intra-measurement BG behavior based solely on the measurement samples (blue dots) over a 5 day period. The red line is the actual CGM measurements. While it is clear that this learner is not succeeding at it's task, this image helps elucidate the challenges that classifiers face using features like this. At the introduction of each new data point (red dot) the regressor is handed a vastly different feature and thus a discontinuity occurs in the predicted data.

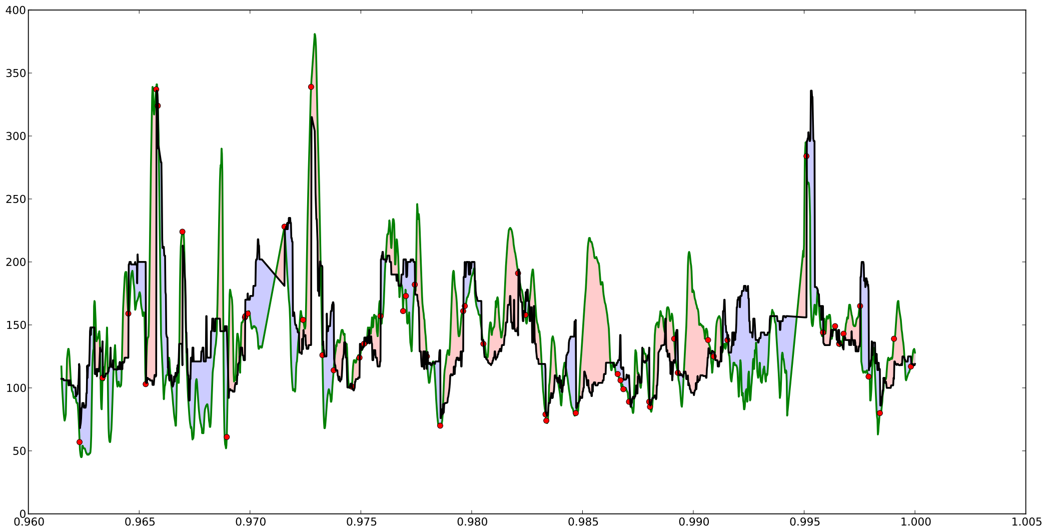

Fig. 5. Above is the only shown regressor that is technically predicting the future (black line), shown over a 6.2 day period. While it does a decent job directly after measurements (red dots), there are often large discontinuities when it has to adjust to new measurements. In contrast to any SVR model that was trained, this regressor has learned that directly after a given measurement the BG must be very similar to this measurement.

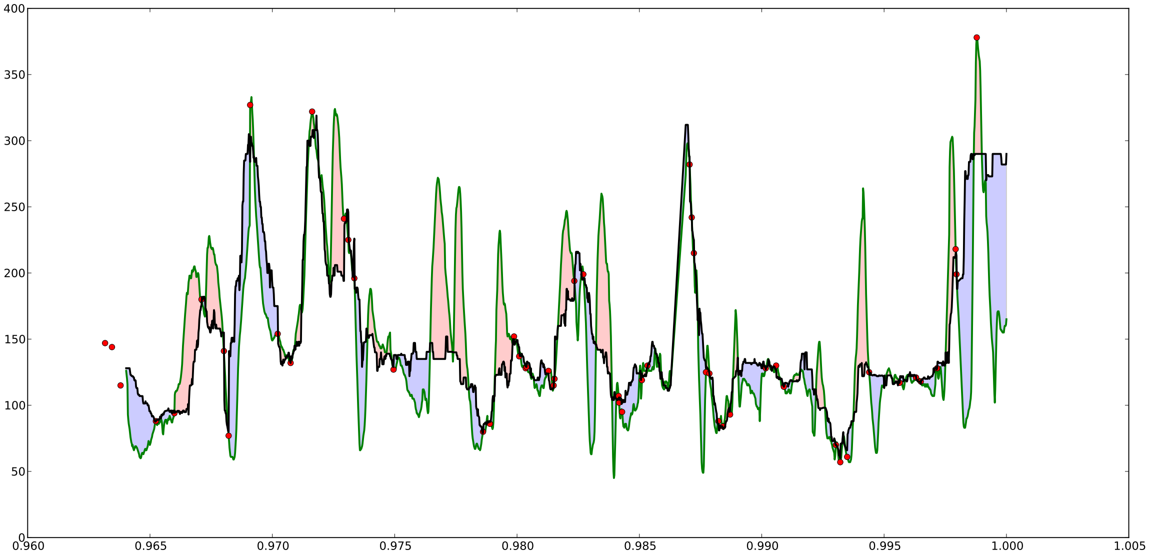

Fig. 6. The above AdaBoost Regressor (black line) with a 1-measurement look-ahead, exibited the lowest R^2 error rate. When inspected, it shows some promising behavior in predicting intra-measurement dynamics. It shuold be noted that there are certain spikes and nadirs that it misses, but they always occur over long intervals with no measurments.

information that was allowed into the feature vectors. In fact false positives (which the best hypoglycemia classifier we found had many of), are not so bad in this scenario. That said, there are many improvements to be made.

- Given the sampling method for the CGM data, some feature components may literally represent random noise and thus their utility in building a classifier for use on other data is questionable.
- The CGM sampling method itself could stand to be improved, possibly by taking into account a known distribution for BG measurements.
- There is also the question of what happens when the user begins to heed the machine's advice, which suggests research into applying on-line in this setting.
- The performance of the intra-measurement regressor should be benchmarked against the physiological models used by physicians to perform the same task.

Ultimately, the self-collected data was not big enough to have comparable results using any of the above methods, but a prototype web service has been designed to update and store new BG measurements in real-time via a smartphone app and bluetooth connected BG monitor. This service is accessible at samsbloodglucose.com and equipped with a version of the described AdaBoost Regressor at samsbloodglucose.com/predict. It is on the agenda to use smartphone capabilities to incorporate features with little or no cost to the user such as step counting and (food) purchase tracking.

7 Conclusion

This work has shown a set of new classification and regression problems relevant to diabetes care and made preliminary attempts at solving them. In the process, we have also suggested some new directions for study in the space of diabetes care with a focus on economic and quality-of-life feasibility. The author hopes that they have provided some insight into the difficulty and potential in this domain of problems and hopes that readers might find some interest in it as well.

References

[1] Fabian Pedregosa, Gal Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, douard Duchesnay. Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research, 12, 2825-2830 (2011)

[2] Jones E, Oliphant E, Peterson P, et al. SciPy: Open Source Scientific Tools for Python, 2001-, http://www.scipy.org/ [Online; accessed 2014-12-16]

[3] Stfan van der Walt, S. Chris Colbert and Gal Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation, Computing in Science and Engineering, 13, 22-30 (2011), DOI:10.1109/MCSE.2011.37 (publisher link)

[4] e Ahmed, Nesreen K. , Atiya, Amir F. , Gayar, Neamat El and El-Shishiny, Hisham(2010) 'An Empirical Comparison of Machine Learning Models for Time Series Forecasting', Econometric Reviews, 29: 5, 594 621

[5] C. Blake and C. Merz. UCI repository of machine learning databases, 1998. Available: site/path/file

[6] M. H. Jensen, T. F. Christensen, L. Tarnow., Z. Mahmoudi, M. D. Johansen, O. K. Hejlesen (2013, January). Professional Continuous Glucose Monitoring in Subjects with Type 1 Diabetes: Retrospective Hypoglycemia Detection. Journal of Diabetes Science and Technology. Volume 7(Issue 1), 135-143.

[7] Bunescu, R.; Struble, N.; Marling, C.; Shubrook, J.; and Schwartz, F. 2013. Blood glucose level prediction using physiological models and support vector regression. In Proceedings of the IEEE 12th International Conference on Machine Learning and Applications (ICMLA). Miami, FL: IEEE.

[8] Beran D, Yudkin JS, de Courten M. Access to care for patients with insulin-requiring diabetes in developing countries: case studies of Mozambique and Zambia. Diabetes Care 2005; 28: 2136-40.

[9] M. Mohri, A. Rostamizadeh, and A. Talwalkar. Foundations of Machine Learning. The MIT Press, 2012.

[10] K. Plis, R. Bunescu, C. Marling, J. Shubrook, F. Schwartz. (year, month). A Machine Learning Approach to Predicting Blood Glucose Levels for Diabetes Management. Modern Artificial Intelligence for Health Analytics. Papers from the AAAI-14

[11] F Sthl. Diabetes Mellitus Glucose Prediction by Linear and Bayesian Ensemble Modeling. (2012, December). PHd Thesis. Available: http://www.control.lth.se/documents/2012/stahl2012lic.pdf

[12] Nishimura R, LaPorte RE, Dorman JS, et al. Mortality Trends in Type 1 Diabetes: The Allegheny County (Pennsylvania) Registry 1965-1999. Diabetes Care 2001; 24: 823-7

[13] Elamin A, Altahir H, Ismail B, Tuvemo T. Clinical pattern of childhood type 1 (insulin-dependent) diabetes mellitus in the Sudan. Diabetologia 1992; 35: 645-8