Using Nearest Neighbor to Classify Activity

Cesar Suarez[1], Faye Cameron, Ph.D.[2], and B. Wayne Bequette, Ph.D.[2]
[1]University of Maryland, Baltimore County, U.S.A., cesar5@umbc.edu
[2]Rensselaer Polytechnic Institute, U.S.A., fmccamer@gmail.com, bequette@rpi.edu

Abstract*– Optimal control response for the closed-loop artificial pancreas depends on the current activity and future activity of Type 1 Diabetes (T1D) patients as this can impact the regulation of their blood glucose levels. By incorporating Smartwatch readings to the closed-loop artificial pancreas, the activity state of the patient can potentially be recognized through the common wrist actions that occur while sleeping, eating, or exercising. The goal of this project is to predict activity state by training a machine learning algorithm (K-Nearest Neighbor) on the accelerometer, gyroscope, and quaternion readings collected on the smartwatch. Activities were logged onto an Android app called DiabetesHelper. The data collected spanned 14 days and was divided into two-minute intervals. Within those two-minute intervals, the mean and standard deviations were generated for each sensor readings. The data was then randomly sampled such that each activity type had equal amount of data. 1000 iterations of the KNN was performed and the k parameter was changed to see how accuracy was impacted. The average accuracy was found from k=2 to k=10. K=2 was found to have the highest average accuracy.

Keywords: Closed-loop artificial pancreas, KNN, Smartwatch

I. Introduction

Type 1 diabetes (T1D) is characterized by the lack of insulin production from the pancreas, as a result T1D patients require external insulin infusion to regulate blood glucose [1]. One form of treatment available for T1D patients is the closed-loop artificial pancreas, shown in Figure 1. The closed-loop artificial pancreas consists of a sensor, a controller, and an insulin pump. The sensor monitors the blood-glucose levels of the patient. Based on the deviation of the patient's blood glucose level from their target blood glucose level, the controller will signal the insulin pump to administer insulin to the patient to return their blood glucose to their target level.

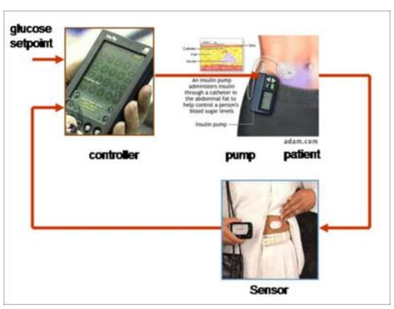

Figure 1. The regulation process for the closed-loop artificial pancreas

The issue with current artificial pancreas systems is that insulin delivery is primarily meal-time based. This form of regulation does not account for other factors that influence blood glucose levels such as sleep and exercising. The time frame in which these activities take place with one another is important to consider as well.

The goal of the Bequette lab is to add Smartwatch and phone sensor readings to the closed-loop artificial pancreas and use the reading to detect the current activity of T1D patients and with more work predict future activity and adapt to patient patterns. Using the sensor readings from wrist actions, the Smartwatch could have the potential to recognize activity state of the patient. Here we try to classify my activity based on smartwatch sensors using a K-nearest neighbor machine learning algorithm.

II. Methodology

In this section, we describe the how data was collected, and the programming involved to load the dataset, train the K-NN model, and test the accuracy of the model in making predictions [2].

The Moto 360 Smart Watch recorded accelerometer, gyroscope, and quaternion readings based on wrist actions using the DHWatchWAttitude app. The readings were then stored in local CSV files on the watch. The duration of activities such as sleeping, eating, and exercising were logged onto an Android app called DiabetesHelper, as shown in Figure 2. Activities outside of those three main activities were treated as "other".

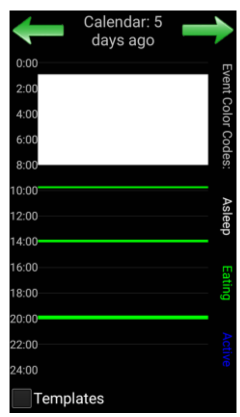

Figure 2. An example of activities being logged on the DiabetesHelper app. The white block represent time slept and the green blocks represents meal times.

Next, the files were imported into MATLAB, where they were converted to another CSV file that tagged the data to their corresponding activity type. 14 days' worth of data was collected. Each new CSV file was read as a dataframe onto

Digital Object Identifier (DOI): http://dx.doi.org/10.18687/LACCEI2018.1.1.493
ISBN: 978-0-9993443-1-6
ISSN: 2414-6390

Pandas Python with the activity type spliced and put into another dataframe.

The remaining dataframe was divided into intervals of 2 minutes. Each 2 minute interval segment was stored into an array and was filtered to make sure at least one of each sensor type was present. The array was converted back into a dataframe with the time column removed. The dataframe was then organized into groups with respect to their sensor type and the means and standard deviations were found for each sensor type.

The groupby dataframe was put into an array and reshaped into a row vector. The first 6 columns represented the means and standard deviations of the gyroscope readings. The next 6 columns represented the means and standard deviations of the accelerometer readings. The next 8 columns represent the means and stand deviations for the quaternion readings. The row vector was converted into a dataframe and the activity type was added to the end of the row of the dataframe. This process continued until a block of data was formed in a dataframe.

The dataframe was organized into groups with respect to their Activity Type and the number of rows that corresponded to each activity type was found. Based on the value of the activity type group with the smallest size, the other activity type groups were reduced so they were the same sized as the smallest activity type group through random selection. This was done because sleeping and other held majority of the data, as shown by Table 1, so each activity type had an equal chance of being predicted.

Table 1. The data distribution of the final dataframe. Other and sleeping made up the majority of the

Activity Type	Data Distribution
Sleeping	28.60%
Eating	0.89%
Exercising	0.58%
Other	69.90%

The final dataframe was split into a training set and a validation set, which each set split into two subsets. One for the input, the average and standard deviations of each sensor type, and the other, the corresponding activity type. The training data was used to build the K-NN model. The validation data was used to test the accuracy. Iterations were done to account for the random selection and the value of the k-parameter was varied from 2 to 10. 1000 iterations were taken for each k value and the average accuracy was found for each value of k. A graph was made plotting the average accuracy as the value of k increased.

III. Preliminary Results

After completing 1000 iterations for each k value, the average accuracy for each value of k was found and a graph was plotted showing the average accuracy as the value of k increased. The average accuracy of the K-NN model in predicting activity type decreased as the value of k increased.

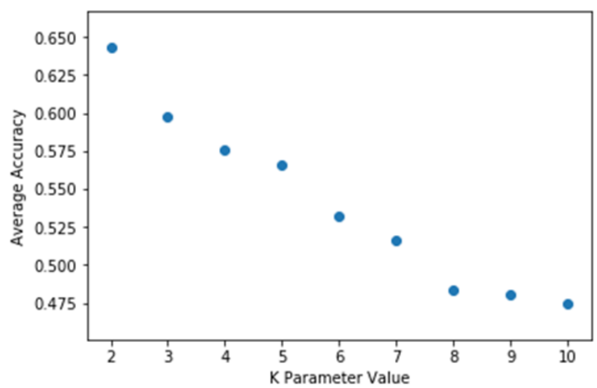

Figure 3. The average accuracy of the K-Nearest Neighbor model in predicting activity type as the value of K increases.

Table 2. The highest accuracy observed for the K-Nearest Neighbor model was 63% when k=2. The confusion matrix for this iteration is displayed below.

Actual Class	Predicted Class			
	Sleeping	Eating	Exercising	Other
Sleeping	7	1	2	1
Eating	0	11	0	0
Exercising	1	2	8	0
Other	3	5	1	2

IV. Discussion

The goal of this project was to predict activity type with the K-Nearest Neighbor model, using Smartwatch sensor readings from wrist actions. After performing 1000 iterations of K-NN and varying the value of the k parameter, the average accuracy of the K-NN model in predicting activity type decreased as the value of k increased. As shown by Figure 3, the average accuracy of the K-NN model was highest when k=2 and lowest when k=10.This could be the result of how the training data is dispersed with respect to the unclassified data points tested. Table 2 shows the iteration at k=2 that showed the highest accuracy, which was 63%. Of the four activity types, sleeping, eating, exercising, and other, the model was able to classify eating perfectly. Of the 11 data points that corresponded to exercising for that iteration, one was misclassified as sleeping and two were misclassified as eating. Sleeping also demonstrated error in classification. Of the 10 data points that corresponded to sleeping for that iteration, one was classified as eating and two were classified as exercising. The ability of the K-NN model to predict 100% of meals shows the potential of incorporating the smartwatch to the closed-loop artificial pancreas for detecting the current activity type of TD1 patients and with more work predict future activity and adapt to patient patterns.

V. Conclusion and Future Work

The data collected will serve as a baseline as the K-NN model is a simple machine learning algorithm. As Table 2 showed, when k=2 the K-NN model was able to classify 63% of free-living activity data, and 100% of meals. The next step would be trying to predict activity type based on smartwatch sensors using a more complex algorithm, neural networks.

Acknowledgment

This material is based upon the work conducted by the Bequette Lab at Rensselaer Polytechnic Institute with financial support from the NSF grant #CNS-1446751 for funding the work.

References

[1] Brownlee, J., Ph.D. (2018, February 01). Your First Machine Learning Project in Python Step-By-Step. Retrieved March 16, 2018, from http://machinelearningmastery.com/machine-learning-in-python-step-by-step/

[2] Cameron, F., Ph.D., Bequette, B. W., Ph.D., Wilson, D. M., M.D., Buckingham, B. A., M.D., Lee, H., Ph.D., & Niemeyer, G., Ph.D. (2011). A Closed-Loop Artificial Pancreas Based on Risk Management. *A Closed-Loop Artificial Pancreas Based on Risk Management, 5*(2), 368-379. Retrieved March 15, 2018, from https://www.ncbi.nlm.nih.gov/pubmed/21527108.