TAMED CANCER
ERC Stg Grant

Physiological Controls Research Center

Detection of physical activity using machine learning methods

Lehel Dénes-Fazakas, László Szilágyi, Jelena Tašić, Levente Kovács, and György Eigner

Óbuda University
Budapest
Hungary

European Research Council

Roadmap

- Introduction
- Datasets
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

Introduction: Research goals

- Determine the most accurate machine learning algorithm for detecting physical activity of patients by using blood glucose measurements
- Evaluate the accuracy and effectiveness of various machine learning algorithms
- Build a simulation framework for data generation
- Develop solution for easy implementation and testing of genuine patient data

Introduction

- Diabetes mellitus (DM): chronic metabolic disease
  - Type 1 DM (T1DM): body is unable to produce insulin internally and patients must have external insulin administration
  - Type 2 DM (T2DM): body produces insufficient insulin to reduce the blood glucose levels
- Planned physical activity helps regulate blood glucose levels and improves the metabolic system
- Machine learning application for recognizing physical activity using only available patient information
- Develop platform for massive data generation using the extended Jacobs T1DM simulator

Roadmap

- Introduction
- Dataset
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

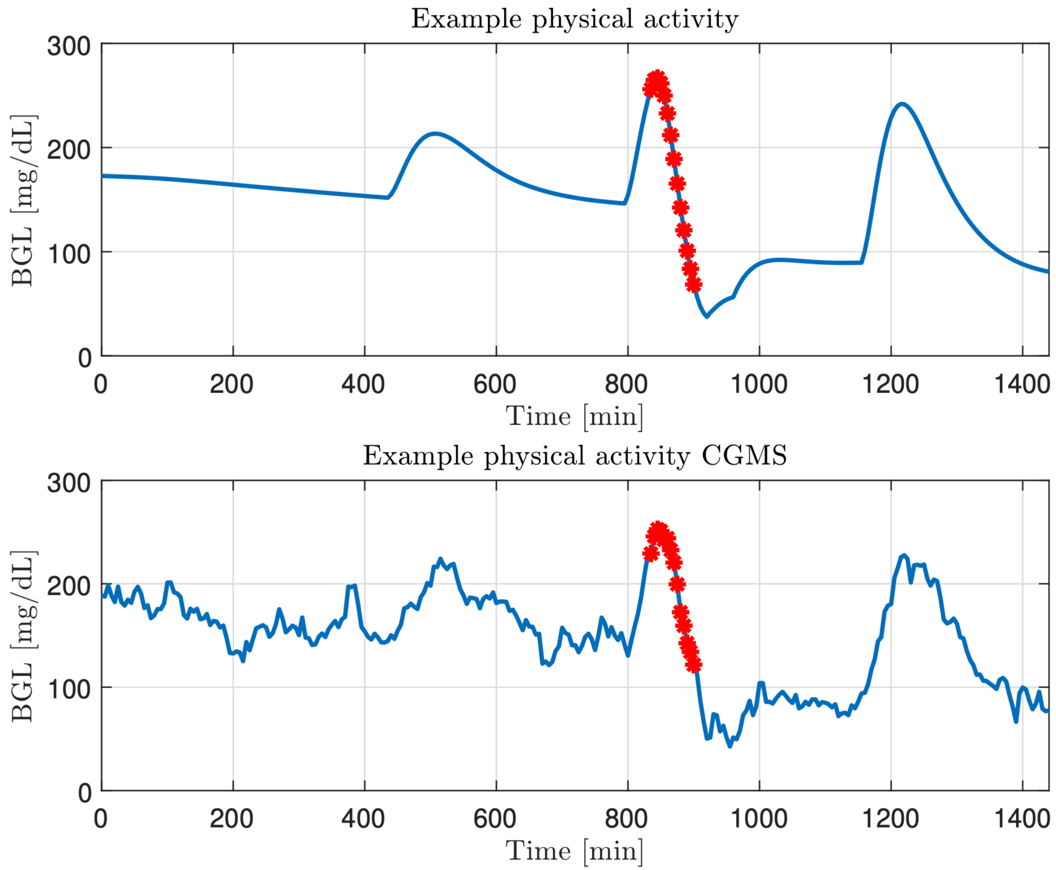
Applied dataset

- Generate synthetic data by using the extended Jacobs T1DM simulator
- Jacobs T1DM simulator:
    - employs the Cambridge-model that contains embedded physical activity sub-model
    - provides 20 virtual T1DM patients (based on 3.5-day outpatient Artificial Pancreas study)
    - used single hormone virtual patient population where the simulator expects the insulin as control input only
    - Completed with Continuous Glucose Monitoring System (CGMS) model for better data generation

Applied dataset

- Applied regimens by using randomization of time instances and amounts of CHO intake:
  - Time of meal consumption: from -30 to +90 min
  - Default time instances in the simulator: breakfast (6 am), lunch (12 pm), and dinner (6 pm)
  - Amount of breakfast: 35 ± 10 g
  - Amount at lunch: 79 ± 10 g
  - Amount at dinner: 117 ± 10 g
  - Duration of physical activity: from 30 to 90 min
  - Blood glucose level at the beginning of the day: 160 ± 20 mg/dL

Datasets



Output of Jacobs simulator model indicates the physical activity with the red section: without CGMS (top) and with CGMS (bottom)

Applied dataset

- The ground truth of the dataset:
  - Patient's body weight *w*
  - End-to-end blood glucose level change *d*:

$$d = bg(14) - bg(0)$$

  - Blood glucose level variation between consecutive sampled points:

$$dp(i) = bg(i+1) - bg(i), \text{ for any } i = 0,\dots,13$$

Applied dataset

- End-to-end blood glucose level change in all inclusive sliding windows:

$$dpp(i) = bg(5i + 4) - bg(5i), \text{ for any } i = 0,\dots,2$$

- Second order changes of the blood glucose level:

$$\begin{aligned} ap(i) &= dp(i+1) - dp(i) \\ &= bg(i+2) - 2 \times bg(i+1) + bg(i), \\ &\quad \text{for any } i = 0,\dots,12 \end{aligned}$$

- The decision *dc* is used as ground truth

Roadmap

- Introduction
- Datasets
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

Prediction

- Binary classification for each feature vector:
  - no physical activity: 0
  - physical activity: 1
- Machine learning models (multi-layer perceptrons) are used to predict the probability of physical activity
- To be interpreted in binary mode, a threshold is applied to the predicted probability
- Other models (decision tree) do not need threshold and can provide binary output directly

Roadmap

- Introduction
- Datasets
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

Machine learning models

- Machine learning algorithms:
  - Logistic Regression
  - AdaBoost Classifier
  - DecisionTree Classifier
  - Gaussian Naive Bayes
  - K-Nearest Neighbors Classifier
  - Support Vector Machines
  - Random Forest
  - Multi-Layer Perceptron Networks

Machine learning models: testing

- Implementation tools: Python v2.7 with the Scikit package
- Split of available feature vectors:
  - training dataset: 75% randomly selected
  - test dataset: 25% assigned for evaluation
- Feature vectors in the training data set were shuffled
- The trained classifiers were applied to predict the presence or absence of physical activity for all feature vectors in the test dataset

Roadmap

- Introduction
- Datasets
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

Performance evaluation

- Performance evaluation: count true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN)
- Establish statistical benchmarks
- Accuracy (ACC):

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

- Sensitivity or true positive rate (TPR):

$$TPR = \frac{TP}{TP + FN}$$

- True negative rate (TNR):

$$TNR = \frac{TN}{TN + FP}$$

Performance evaluation

- Positive prediction value (PPV):

$$PPV = \frac{TP}{TP + FP}$$

- False positive rate (FPR):

$$FPR = \frac{FP}{TN + FP}$$

- F1-score or Dice score:

$$F_1 = \frac{2 \cdot TRP \cdot TNR}{TPR + TNR} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

Roadmap

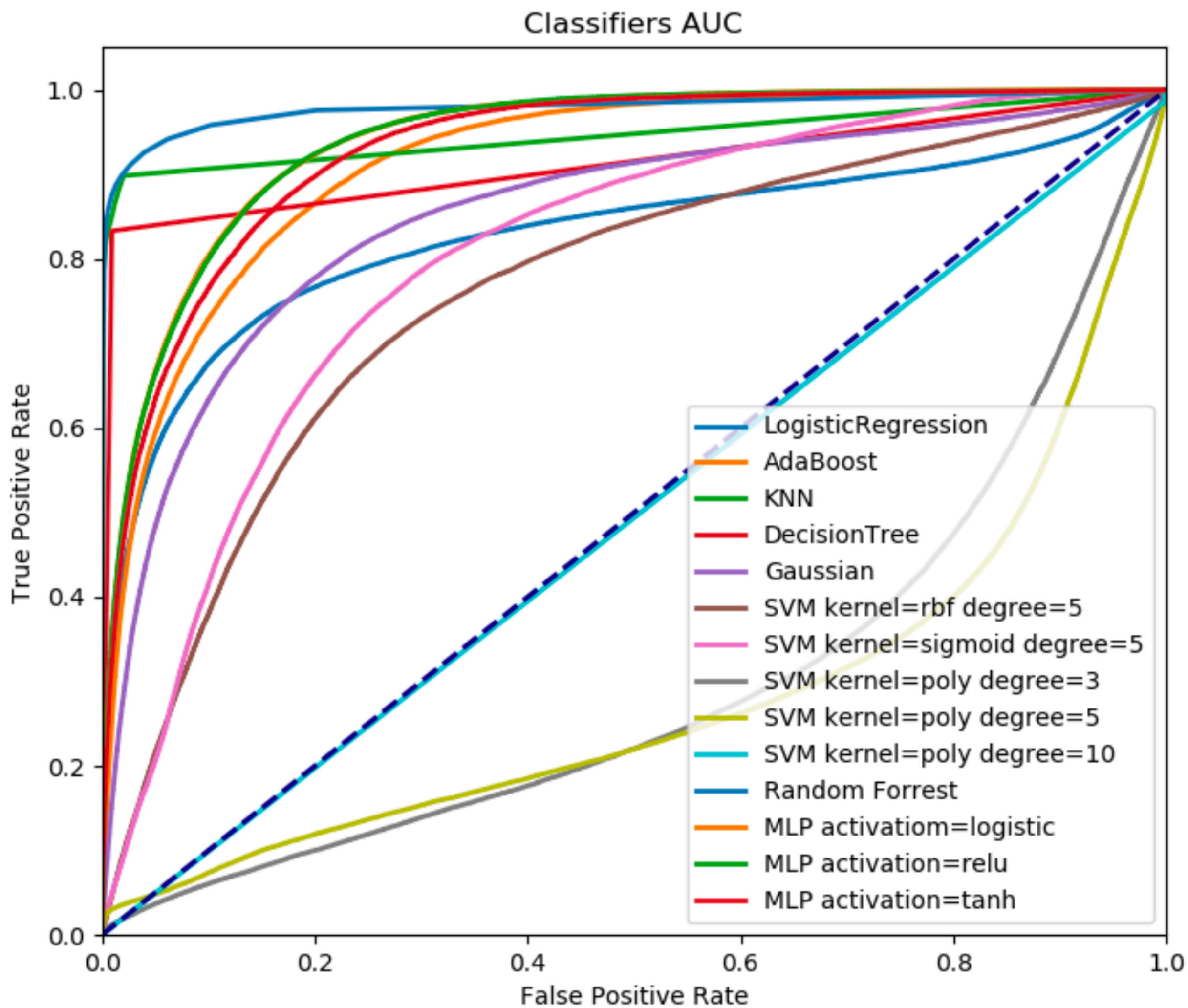
- Introduction
- Datasets
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

Confusion matrix for tested classification models

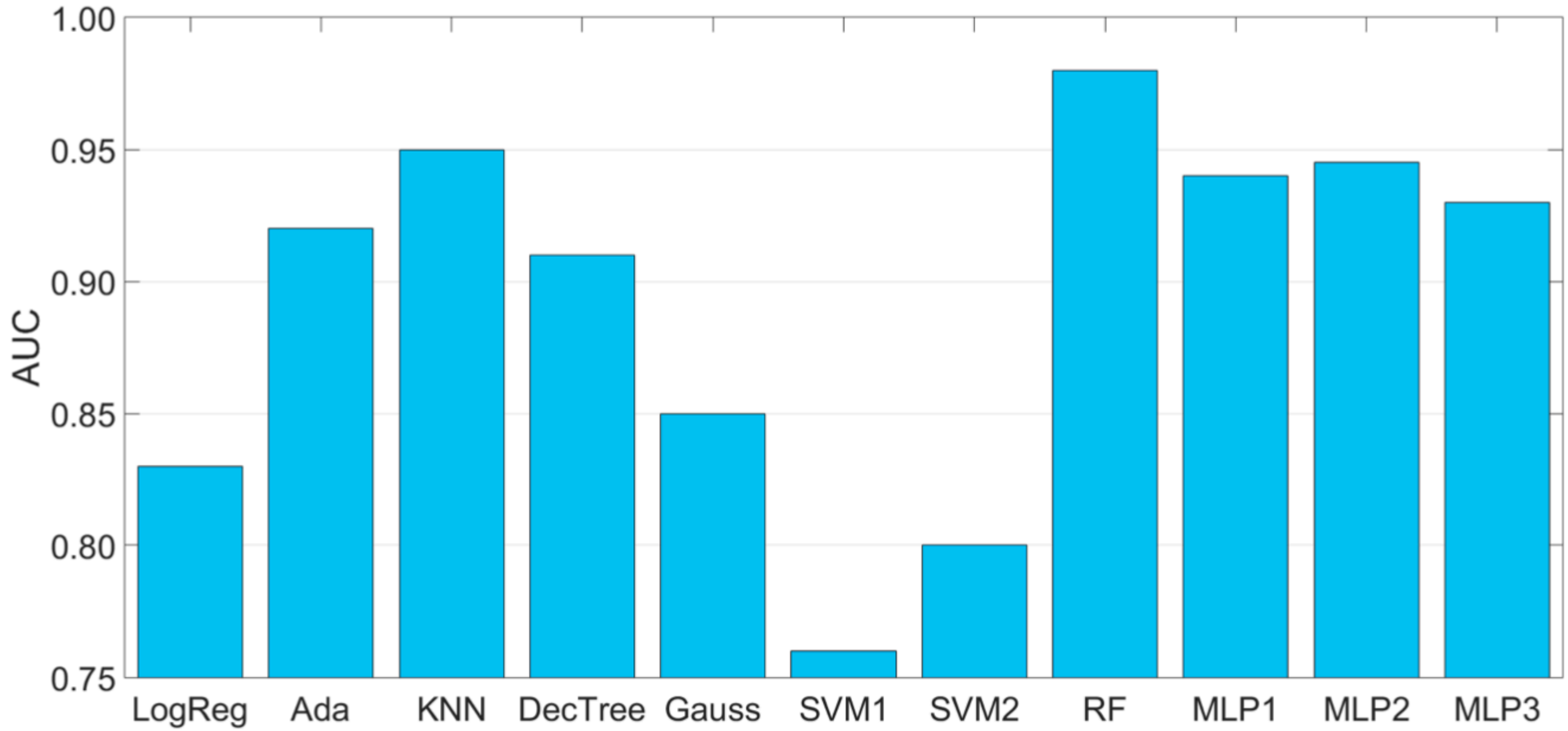
- Values are normalized in each row
- Classification is successful if the rates of true positives and true negatives are above 0.8
- Only the RF model achieved rate higher than 0.9
- Three SVM models, using polynomial kernel, predicted the opposite

True value	Predicted value: 0			Predicted value: 1		
0	LogReg	:	0.779	LogReg	:	0.221
0	Ada	:	0.833	Ada	:	0.167
0	KNN	:	0.980	KNN	:	0.020
0	DecTree	:	0.991	DecTree	:	0.009
0	Gauss	:	0.789	Gauss	:	0.211
0	SVM1	:	0.717	SVM1	:	0.283
0	SVM2	:	0.743	SVM2	:	0.257
0	SVM3	:	0.327	SVM3	:	0.673
0	SVM4	:	0.311	SVM4	:	0.689
0	SVM5	:	1.000	SVM5	:	0.000
0	RF	:	0.961	RF	:	0.039
0	MLP1	:	0.864	MLP1	:	0.136
0	MLP2	:	0.863	MLP2	:	0.137
0	MLP3	:	0.848	MLP3	:	0.152
1	LogReg	:	0.221	LogReg	:	0.779
1	Ada	:	0.167	Ada	:	0.833
1	KNN	:	0.101	KNN	:	0.898
1	DecTree	:	0.166	DecTree	:	0.834
1	Gauss	:	0.211	Gauss	:	0.789
1	SVM1	:	0.284	SVM1	:	0.716
1	SVM2	:	0.257	SVM2	:	0.743
1	SVM3	:	0.673	SVM3	:	0.327
1	SVM4	:	0.689	SVM4	:	0.311
1	SVM5	:	1.000	SVM5	:	0.000
1	RF	:	0.073	RF	:	0.927
1	MLP1	:	0.136	MLP1	:	0.864
1	MLP2	:	0.137	MLP2	:	0.863
1	MLP3	:	0.152	MLP3	:	0.848

ROC curves


Classifiers AUC
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
LogisticRegression
AdaBoost
KNN
DecisionTree
Gaussian
SVM kernel=rbf degree=5
SVM kernel=sigmoid degree=5
SVM kernel=poly degree=3
SVM kernel=poly degree=5
SVM kernel=poly degree=10
Random Forrest
MLP activatiom=logistic
MLP activation=relu
MLP activation=tanh

AUC values



1.00
0.95
0.90
0.85
0.80
0.75
AUC
LogReg
Ada
KNN
DecTree
Gauss
SVM1
SVM2
RF
MLP1
MLP2
MLP3

Accuracy

Classifier	PPV	TPR	F_1	TNR	FPR	ACC
LogReg	0.149	0.778	0.250	0.778	0.222	0.778
Ada	0.198	0.832	0.320	0.832	0.168	0.832
KNN	0.688	0.899	0.779	0.980	0.020	0.976
DecTree	**0.813**	0.833	**0.823**	**0.990**	**0.010**	**0.983**
Gauss	0.157	0.789	0.261	0.789	0.211	0.789
MLP1	0.239	0.863	0.374	0.863	0.137	0.863
MLP2	0.237	0.862	0.372	0.862	0.138	0.862
MLP3	0.217	0.848	0.346	0.848	0.152	0.848
RF	0.537	**0.926**	0.680	0.960	0.040	0.959
SMV1	0.111	0.716	0.193	0.716	0.284	0.716
SVM2	0.125	0.742	0.215	0.742	0.258	0.742
SVM3	0.024	0.326	0.044	0.326	0.674	0.326
SVM4	0.022	0.310	0.041	0.310	0.690	0.310
SVM5	0.000	0.000	0.000	1.000	0.000	0.953

Roadmap

- Introduction
- Dataset
- Prediction
- Machine learning models
- Performance evaluation
- Results and discussion
- Conclusions

Conclusions

- Introduced a machine learning based framework for detecting physical activity using features extracted from blood glucose samples taken at five minutes intervals
- Several classification models were employed using various parameter settings
- The best classifiers: *Decision Tree*, *K-Nearest Neighbors*, and *Random Forest*
- Other models may be suitable: they need additional mechanisms to avoid false positives

Acknowledgment

- This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. 679681)
- Project No. 2019-1.3.1-KK-2019-00007 has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the 2019-1.3.1-KK funding scheme
- L. Szilágyi is János Bolyai Research Fellow of the Hungarian Academy of Sciences
- L. Dénes-Fazakas was supported by the UNKP-20-2 New National Excellence Program of the Ministry for Innovation and Technology