Journal of Computer Science

Original Research Paper

Predicting Risk of Diabetes using a Model based on Multilayer Perceptron and Features Extraction

Francesca Fallucchi and Alessandro Cabroni

Guglielmo Marconi University, 44 Plinio Street, 00193 Rome, Italy

Article history
Received: 13-05-2021
Revised: 27-07-2021
Accepted: 02-08-2021

Corresponding Author:
Francesca Fallucchi
Guglielmo Marconi University,
44 Plinio Street, 00193 Rome,
Italy
Email: f.fallucchi@unimarconi.it

Abstract: Diabetes (diabetes mellitus) is a disease emerging when a person has a high blood sugar level for a prolonged period. In the healthcare context, one of the most important topic is the prevention of the disease. This study relates to diabetes prevention. In particular, it aims to produce random generated datasets according to a rule named Finnish Diabetes Risk Score and to test these datasets with a model based on Multilayer Perceptron and features extraction, to determine the diabetes risk. A second layer of the model produces the prediction. This classification layer bases on comparing the single unlabeled element (its features) against all labelled elements (their features), considering risk level similarities too. The health rule consider daily lifestyle and health parameters. We define random generated datasets to avoid privacy problems and to manage equally distributed data in order to control better the behavior of the applied model and to propose datasets to simplify the comparing of the behaviors for different models. Moreover, in this study we propose an initial hypothesis to test the explain ability of the model in terms of our datasets (input parameters, corresponding to health rule parameters), defining a method based initially based on Relevance Propagation, Deep Taylor Decomposition and testing elements features distribution. In this study, we obtain the generation of random datasets equally distributed in respect to the possible risk levels and with a mean distribution near to 0,5 (note that we manage normalized values) for the different input attributes. We define a MLP with no under fitting or over fitting problems. All accuracies values (in our scenario, definition of accuracy considers class similarity too because of ordered risk levels) for the overall model are greater than 0.939, with best result over 0.96 for 1500 labelled elements as training dataset.

Keywords: Diabetes Risk Prediction, FI Nnish Diabetes R Isk S Core, Multilayer Perceptron, Explain ability

Introduction

The diseases prevention is one of the topic of interest for healthcare. Diabetes mellitus is a chronic and lifelong metabolic disorder that occurs either when the pancreas does not secret enough insulin (type 1 diabetes), or when the body's cells do not respond to insulin, so having a high level of glucose in the blood (type 2 diabetes). In particular, we study a module to identify risks for type 2 diabetes for a person. Hence, in this study we are interested in diabetes prevention. It is an important issue considering significant human, economic and social costs (Perveen *et al*. (2019). Our work aims to define random datasets and to test them using a model based on Multilayer Perceptron (MLP) and features extraction, to determine the diabetes risk according to daily lifestyle and health parameters. These parameters are Body Mass Index (BMI), age, waist circumference, use of blood pressure medication, history of high blood glucose, physical activity, consumption of vegetables/fruits/berries and family history of diabetes. We choose a model based on MLP and a classifier based on similarity, in order to try to improve the accuracy by MLP feature extraction and some particular implementation in the classification layer (similarity between elements considering class similarities too). There are different works about this specific issue for diabetes (e.g., Xiong *et al*. (2019), Chandrakar *et al*. (2016)). We want to contribute with another possible model having these features: High level of prediction quality, initial training and testing with randomly generated data, support for future explain ability according to input attributes so to add our solution to recent studies in an analogous context (e.g., Kopitar *et al.* (2019)). About randomly generated data, in this way we have not privacy problems. With real data, we would need privacy consents and anyway of course we cannot consider open data, generally very useful (e.g., Fallucchi *et al.* (2018)). We use dataset randomly generated according to a healthcare rule named Finnish Diabetes Risk Score (FINDRISC), with the possibility to improve the model later with real data and more features too. Briefly, our contribution defines a diabetes prevention model, producing testing datasets and setting a future modality for the explain ability of predictions in respect to input attributes (features of a person). For our implementation, we use Colaboratory[1] as environment to execute our code, selecting Python™ 3 and using Tensor Processing Unit (TPU) as runtime environment. We use MLP component for features extraction with a classification component based on Cosine similarity between elements and based on diabetes risk levels similarities too. We evaluate the quality of the overall model using a definition of accuracy that consider the similarity between risk levels. We define a future implementation for explain ability of our model in term of input attributes, simply understandable by a human expert such as a Medical Doctor (MD), starting from considerations related to Layer-Wise Relevance Propagation (LRP) and Deep Taylor Decomposition (DTD) (e.g., Bach *et al.* (2015), Montavon *et al.* (2017)). Next sections organize as follows. Related work section reports some useful articles about Machine Learning (ML) techniques used in the general context of diabetes. Rule for diabetes risk section, presents FINDRISC together with the derived algorithm used to create training and testing datasets for our model. Method section describes our general solution. Accuracy section describes the accuracy definition used in our context, considering the similarity between risk levels. Architecture section describes the details of the prediction model (MLP and classification component). Experimental results section reports the results of our tests for the model from the accuracies point of view. Tables and Figures section dedicates to present also the results outlined in the paper, graphically and in tabular form. Explain ability section discusses a hypothetical solution for our model. In discussion and future work section, we outline the achieved results and our future developments.

Related Work

In Khanam *et al.* (2021), they use Pima Indian Diabetes (PID) dataset, testing seven ML algorithms for diabetes predictions. They use Waikato Environment for Knowledge Analysis (WEKA) tool too. The best results obtained are by using Logistic Regression (LR) and Support Vector Machine (SVM). They also implemented a Neural Network (NN) with two hidden layers providing 88.6% accuracy. In Tigga *et al.* (2020), the study is about diabetes risk based on lifestyles and family background. They manage 952 instances produced by questionnaire about health, lifestyle and family background. They studied the behavior of different ML algorithms applied to both this new dataset and PID dataset. Most accurate performance arises for Random Forest (RF) Classifier. In Hasan et al. (2020), they use ML models trained by PID dataset. They propose a solution based on pre-processing, K-fold Cross-Validation (KCV), Grid search for hyper-parameters, in order to select the best model among different possibilities. In future work they are interested in trying to apply their work in other medical context to verify the solution in its generality. In Contreras *et al.* (2018), there is a review about Artificial Intelligence (AI) techniques for diabetes, considering 141 articles. They study AI techniques considering three kind of problems: Learning from knowledge, exploration and discovery of knowledge, reasoning from knowledge. In particular, about first problem, they consider also the following solutions: SVM, RF, Evolutionary Algorithm (EA), Deep Learning (DL), Naïve Bayes (NB), Decision Tree (DT) and regression algorithms. They use these categories: Blood glucose control strategies; blood glucose prediction; detection of adverse glycemic events; insulin bolus calculators and advisory systems; risk and patient personalization; detection of meals, exercise and faults; lifestyle and daily-life support in diabetes management. In Swapna *et al.* (2018), there is a methodological study to classify diabetic and normal Heart Rate Variability (HRV) signals by using DL. HRV signals relate to Electro Cardio Gram (ECG) signals. The architecture considers these modules: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) and SVM for final classification starting from features extracted by CNN and LSTM components. As kernel, there is Radial Basis Function (RBF). They implement their tests, using Graphics Processing Unit (GPU) with Tensor Flow, Keras and scikit-learn. The solution is helpful for the diabetes diagnosis using ECG signals. The accuracy is 95.7%. In Miotto *et al.* (2016), they produce “deep patient”, a framework for modelling patients by features automatically extracted from an Electronic Health Record (EHR) dataset with DL techniques. Data are from Mount Sinai data warehouse. They process EHRs using a Deep Neural Network (DNN) based on Stacked Denoising Autoencoder (SDA). The solution is useful for different predictions, also for diabetes diseases. In Sisodia *et al.*

[1] https://colab.research.google.com/notebooks/welcome.ipynb

(2018), they use DT, SVM and NB for predicting early-stage diabetes, with a dataset from University of California, Irvine (UCI) repository. Performance measures are precision, accuracy, F-measure, recall and Receiver Operating Curve (ROC). They test the solution, using WEKA tool. The best accuracy is for NB algorithm. In Kavakiotis *et al.* (2017), they study a review of Machine Learning (ML) and data mining solutions for diabetes issue considering diabetic complications, genetic background and environment, healthcare and management and prediction and diagnosis too. In this context, 85% of ML algorithms are supervised algorithms and 15% are unsupervised algorithms. SVM is the most used and it has better results. In Mercaldo *et al.* (2017), they work on a method for classifying patients with diabetes, using Hoeffding Tree (HT) algorithm, also known as Very Fast Decision Tree (VFDT) algorithm. They want to classify diabetes patients using the minimum features number. They consider number of times pregnant, plasma glucose concentration at 2 hours in an oral glucose tolerance test, triceps skin fold thickness, diastolic blood pressure, 2-Hour serum insulin, BMI, diabetes pedigree function, age. They use data from UCI repository, as usual. They verify if selected features are significant to determine if a patient is diabetic. They use these classification algorithms: J48, MLP, HT, JRip, Bayesian Network (BN) and RF. Classification uses WEKA tool, obtaining a precision of 0.757 and a recall of 0.762. In Erdem *et al.* (2012), they introduce Graph Transduction Game (GTG) for the context of graph transduction. In the tests, they use also diabetes dataset from UCI to study the behavior of GTG for comparing to other methods. In Table 1, we summarize the related work presented in this section.

Rule for Diabetes Risk

To generate random data useful to test our model, we consider a healthcare rule, in particular we choice FINDRISC[2]. As initial reference for this rule, see Lindström *et al.* (2003). For the validation of the rule, see Makrilakis *et al.* (2011) and Zhang *et al.* (2014). About clinical practice guidelines, see Pottie *et al.* (2012). Other useful articles about FINDRISC are e.g., Lindström *et al.* (2010) and Noble *et al.* (2011). The rule aims to identify high-risk individuals, without doing laboratory tests. Starting from attribute related to healthcare data and lifestyle of a person, the rule calculates the diabetes risk. We consider all five risk levels in respect to score: Very low (0-3), low (4-8), moderate (9-12), high (13-20) and very high (21-26). The attributes to consider are the following: BMI (weight (kg)/height squared (m2)) (B), age (years) (A), Waist circumference (W) (differentiating for Gender (G)), Use of blood pressure medication (U), History of high blood glucose (H), Physical activity expressed in hours/week (P), Daily consumption of vegetables, fruits or berries (D), Family history of diabetes (F). The following algorithm calculates the score according to the rule:

```
score ← 0;
if ($45 \le A \le 54$) score ← score+2
else if ($55 \le A \le 64$) score ← score+3
else if ($64 < A$) score ← score+4;
if ($25 < B \le 30$) score ← score+1
else if ($30 < B$) score ← score+3;
-- centimeter
if (G=man & $94 \le W < 102$) score ← score+3
else if (G=man & $102 \le W$) score ← score+4
else if (G=woman & $80 \le W < 88$) score ← score+3
else if (G=men & $88 \le W$) score ← score+4;
if (U=yes) score ← score+2;
if (H=yes) score ← score+5;
if ($P<4$) score ← score+2;
if (D=no) score ← score+1;
if (F=yes with 2nd degree relative) score ← score+3
else if (F=yes with 1st degree relative) score ← score+5;
```

We generate datasets equally balanced in respect to the possible risk levels. We produce random data normalized to [0,1] and corresponding to the different input attributes used by the rule. These attributes correspond to the input for our model, while the calculated risk value corresponds to the right prediction for our model. Risk value is useful for training and validation set. It uses also to determine the quality of our model during the experimentation with a testing dataset.

Table 1: Summary of related work

Year	Authors	Some of the considered components
2018	Contreras *et al.*	SVM, RF, EA, DL, NB, DT
2012	Erdem *et al.*	GTG
2020	Hasan *et al.*	DT, RF, NB, MLP
2017	Kavakiotis *et al.*	SVM
2021	Khanam *et al.*	LR, SVM, NN
2017	Mercaldo *et al.*	HT
2016	Miotto *et al.*	SDA
2018	Sisodia *et al.*	DT, SVM, NB
2018	Swapna *et al.*	CNN, LSTM, SVM, RBF
2020	Tigga *et al.*	RF

Methods

After evaluating some useful papers in the context of healthcare predictions with particular interest for diabetes issue, we set our solution according to these steps:

[2] https://www.mdcalc.com/findrisc-finnish-diabetes-risk-score

1) Choice of a rule to produce significant random datasets
2) Choice of an MLP for features extraction
3) Definition of a classification component
4) Test of the solution using the same testing dataset, different training datasets and different number of extracted features

For first, we chose FINDRISC according to the references already cited about this rule. For MLP, we followed three directions: Doing initial and general tests, considering initial MLP models described in literature and using Grid Search CV of scikit-learn tool. About the last issue, we considered MLP Classifier of scikit-learn (with 200 as max number of iterations and a dataset of 1000 elements) for the following parameters: hidden_layer_sizes (with three hidden layers with 128/256/32, 256/512/32, or 512/1024/32 neurons), learning_rate_init (0.01 or 0.1), validation_fraction (0.1 or 0.2), batch_size (50 or 100). Table 2 for the results of our interest. We combined all the considerations emerged from the three-direction analysis to choice our MLP and its parameters (see architecture section for details). Generally, our model solution considers the following steps:

- Create testing dataset
- Loop on possible training dataset cardinalities
  - Create specific training dataset
  - Loop on possible number of features to extract from MLP
    - Instantiate and fit MLP
    - Extract features from last hidden layer both for training and testing datasets
    - Calculate predictions for testing dataset with a specific prediction component based on similarity between a testing element and all training elements in respect to extracted features and based on similarity between risk classes

At the end of our work, we also define a theoretical hypothesis for explain ability in respect to input values. In Fig. 1, we briefly summarize our method.

Accuracy

For the proposed overall model, we consider this accuracy definition:

$$accuracy1 = \frac{\sum_{i=1}^{np}\left(1-\frac{|PRi-RLi|}{m-1}\right)}{np} \quad (1)$$

Where:

- *np*: Total number of predictions (unlabeled elements)
- *RLi*: Right label for *i* element
- *PRI*: Prediction for *i* element
- *m*: Number of possible labels (in our study, *m* = 5)

This definition of accuracy is significant because of the order implicitly defined among the five levels of diabetes risks.

Architecture

In Fig. 2, we can see a high-level model of our solution.

MLP component is instantiated with the following structure and parameters:

- Input layer: 9 (corresponding to the number of attributes for FINDRISC)
- First dense hidden layer with 512 neurons (with batch normalization, Rectified Linear Unit (ReLU) activation function and dropout with 0.25 probability)
- Second dense hidden layer with 1024 neurons (with batch normalization, ReLU activation function and dropout with 0.25 probability)
- Third dense hidden layer with 8, 16, or 32 neurons (corresponding to the number of extracted features in the different tests) (with batch normalization, ReLU activation function and dropout with 0.25 probability)
- Fourth dense hidden layer with 5 neurons corresponding to the number of diabetes risk
- Softmax activation function to normalize MLP output to probability distribution (not used for final prediction)
- Batch Size = 100
- decay = 1e-6
- epochs = 1000
- learning Rate = 0.01
- validation Split = 0.2
- optimizer = adam
- loss = sparse_categorical_crossentropy
- metrics = accuracy

Classification component implements the following algorithm that it considers elements represented in terms of extracted features (it is important to consider that training dataset balances in respect to the possible diabetes risks):

- Loop on test dataset
- Calculate D the array of normalized Euclidean distances between the current test unlabeled element and all elements of training dataset (1)
- From D, calculate G, the array of normalized Gaussian kernel similarities between the current test unlabeled element and all elements of training dataset (2)
- Calculate probability distribution S for testing element considering G (3)
- Recalculate probability distribution adding a factor to consider the similarity between risk levels (4)

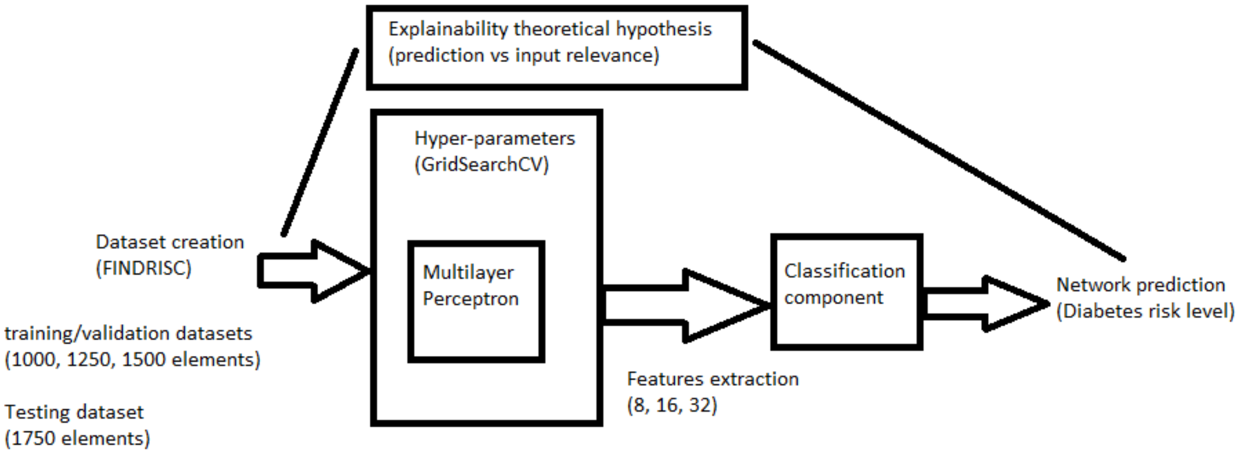

Explainability theoretical hypothesis (prediction vs input relevance)
Hyper-parameters (GridSearchCV)
Dataset creation (FINDRISC)
Multilayer Perceptron
Classification component
Network prediction (Diabetes risk level)
training/validation datasets (1000, 1250, 1500 elements)
Testing dataset (1750 elements)
Features extraction (8, 16, 32)

Fig. 1: Diagram of the method

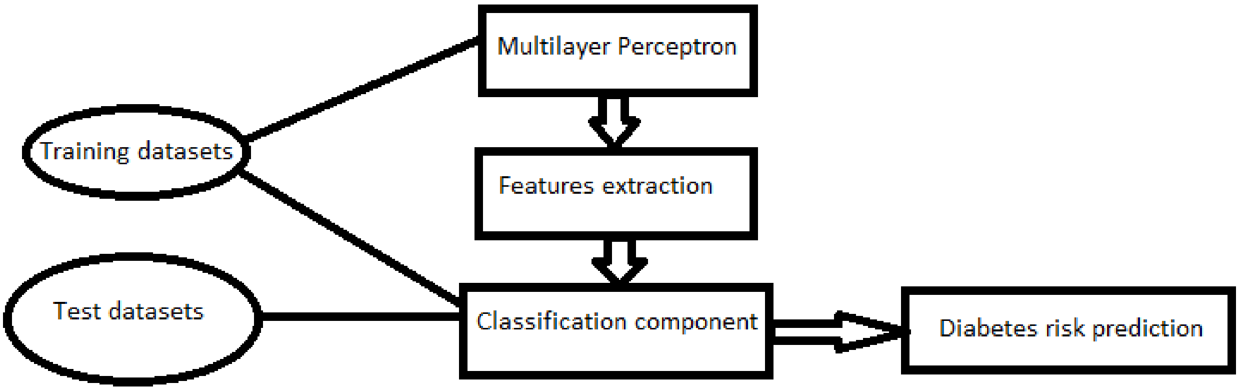


Fig. 2: Model high-level representation

Table 2: Useful results from scikit-learn Grid Search CV

Value (+/-)	Batch size	Hidden layer sizes	Learning rate init	Validation fraction
0.794 (0.057)	100	512, 1024, 32	0.01	0.2
0.748 (0.151)	100	128, 256, 32	0.1	0.2
0.681 (0.049)	50	128, 256, 32	0.1	0.2
0.602 (0.567)	100	128, 256, 32	0.1	0.1
0.600 (0.566)	100	512, 1024, 32	0.1	0.1
0.597 (0.561)	100	256, 512, 32	0.1	0.1
0.583 (0.547)	50	256, 512, 32	0.1	0.2
0.386 (0.533)	50	512, 1024, 32	0.1	0.2
0.386 (0.528)	50	128, 256, 32	0.1	0.1
0.346 (0.409)	50	256, 512, 32	0.1	0.1
0.356 (0.445)	50	512, 1024, 32	0.1	0.1

In details:

```
(1)
for trainElem=0 to numberTrainElems-1:
    D[trainElem]=EuclideanDistance(
    featuresTrainingDataSet[i],
    featuresTestingDataSet[testElem])
D=(D- min(D))/( max(D)- min(D))
(2)
for trainElem=0 to numberTrainElems-1:
    G[trainElem]=
    e^(-D[trainElem]^2/(2*sigma^2))
G=(G- min(G))/(max(G)-min(G))
```

where sigma (Gaussian kernel width) equals to 0.5

```
(3)
S=[0.0 for h in range (0,m)]
for x=0 to numberTrainElems-1:
    S[L[x]]+=G[l]
S=S/sum(S)
```

where L[x] is the is the (right) risk associated to training element x and m=5 (number of risk levels)

```
(4)
S1=copy(S)
S[0]=S1[0]+S1[1]*(m-1)/m
for h=1 to (m-1)-1:
S[h]=S1[h]+
(S1[h-1]+S1[h+1])/2*(m-1)/m
S[m-1]=S1[m-1]+S1[m-2]*(m-1)/m
S=S/sum(S)
```

Classification component considers the similarities between one unlabeled (testing) element and all labelled (training) elements, so to determine the higher probable risk level for the unlabeled element and obtaining a probability distribution. Then, classification component refines the probability distribution, considering the contributions of similar classes, in particular adding a component of probability distribution related to the nearest classes. Final prediction corresponds to argmax on the probability distribution.

Experimental Results

From Fig. 3 to 11, we present the accuracies (in terms of the classical definition) results for MLP (training and validation), in the different scenarios considering the behavior until the 1000 epochs of training. Testing dataset has always 1750 elements, while training datasets have 1000, 1250 and 1500 elements. Training datasets are used for training validation too, according to validation Split parameter. The number of extracted features is 8, 16 and 32. As we can see, usually the curve for training accuracy is quite similar and slightly better than the curve of validation accuracy. This scenario suggests that there are not significant possible problems about over fitting or under fitting.

Fig. 12 shows the results about accuracy1 (definition that it considers the similarities between classes/risk levels) for the whole model. Each curve presents the behavior of the model in terms of accuracy1 in varying the number of training elements (1000, 1250 and 1500), considering the particular number of extracted features and, as usual, with the number of testing elements set to 1750. Behaviors are similar, but the best achieves for 8 extracted features and 1500 training elements. In Fig. 13 and 14, we present the histograms representing the data distribution for the generated datasets (testing dataset of 1750 elements and training datasets of 1000, 1250 and 1500 elements). The distribution are expressed in terms of mean (average) and standard deviation for each input attribute, normalized to [0,1]. In Table 3, we present the model execution times.

We can summarize the results of our tests as follows:

- Generation of random datasets equally distributed in respect to the possible risk levels and with a mean distribution near to 0,5 for the different input attributes (standard deviation is more variable)
- Training of MLP (first layer of the model) with training accuracy curve slightly better than validation accuracy with no under fitting or over fitting problems
- All accuracy1 values for model are greater than 0.939, with best results over 0.96 for 1500 labelled elements (training dataset) and 8 extracted features

(*) attribute matching: 0-gender, 1-age, 2-BMI, 3-waist circumference, 4-use of blood pressure medication, 5-history of high blood glucose, 6-physical activity, 7-consumption of vegetables/fruits/berries, 8-family history of diabetes (for Fig. 13 and 14).

Explain Ability

In ML, we are interested in having good predictions but another important issue relates to explain why one model produces these predictions (e.g., Tjoa *et al*., 2020) and Holzinger *et al*. (2019). Therefore, in our work we are interested in understanding a prediction in respect to the input features data. Hence, a user could understand what the significant initial data affect the prediction, understanding new relations between input data and prediction and helping in validations of the results. This is particularly important for sensitive contexts such as healthcare, where it is fundamental a validation of results by human expert (e.g.,: MD). Starting from the research about LRP and DTD (Bach *et al.* (2015), Montavon *et al*. (2017) and Samek *et al*. (2019)), we define a rule to show the relevance of the single input element against the particular feature extracted from MLP. Relevance relates to the weights of the edges for the trained MLP, without biases. We also consider the weight of the different features for the classification component, using training data (known predictions for the elements) to weight the standard deviation of a feature. The idea is that if a feature has a high variation, it is more useful in establishing the distance between elements (if we consider a normalization too). Hence, it is more important for the classification. The framework under studying and implementation for explain ability of our model, bases on these forward recursive definitions:

$$R_{i,v} = norm\left(vi \cdot \sum_{f=1}^{F} \left[H\left(Rnorm_{i,0}^{f}, \sigma_f \right) \right] \right) \tag{2}$$

$$Rnorm_{i,l}^{f} = \sum_{J=1}^{C_{l+1}} \frac{Rnorm_{j,l+1}^{f} \cdot W_{i^l j^{l+1}}^{+}}{\sum_{k=1}^{C_l} W_{k^l j^{l+1}}^{+}} \tag{3}$$

$$Rnorm_{1,LF}^{f} = 1 \tag{4}$$

where:

- $(v_1,\ldots,v_n)$: Input data (normalized to [0,1])
- R: Relevance for an attribute of input data

- *Rnorm*: Relevance for explain ability component of MLP features, independently from the particular input data
- *F*: Number of extracted features from MLP
- C_{l+1}: number of neurons for layer $l + 1$ of MLP (layer 0 is for input data)
- LF: Layer of MLP used for features extraction; we consider that this layer has one neuron corresponding to the particular feature f
- $W^{+}_{i^l j^{l+1}}$: Absolute value of the weight of MLP for the edge connecting neuron i of layer l to neuron j of layer l+1
- *H*: Function (under studying) to combine the behavior of MLP and the variation of a feature in the classification component
- *norm*: Function to normalize the relevance to [0,1]

Table 3: Model execution time

	Number of labelled elements		
Number of extracted features	1000	1250	1500
8	0:03:13	0:03:12	0:03:15
16	0:03:54	0:03:54	0:03:51
32	0:04:21	0:04:23	0:04:34

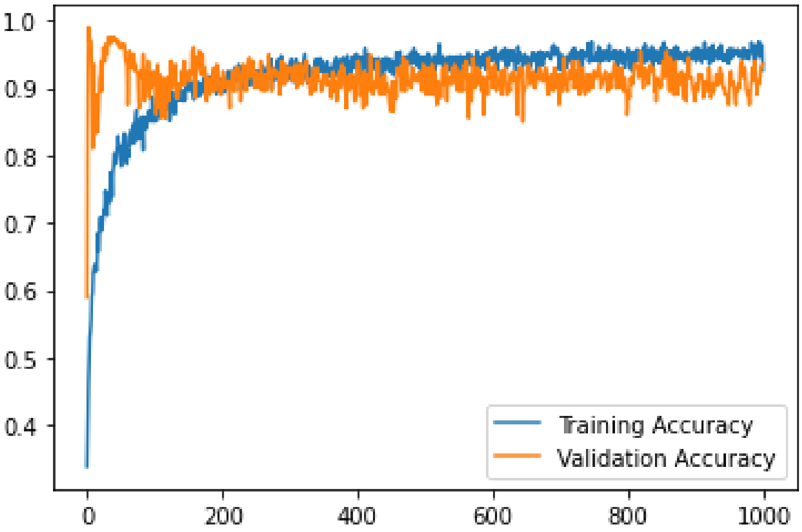


Fig. 3: Results for MLP for training data Set with 1000 elements (extracted features = 8): Accuracies vs epochs

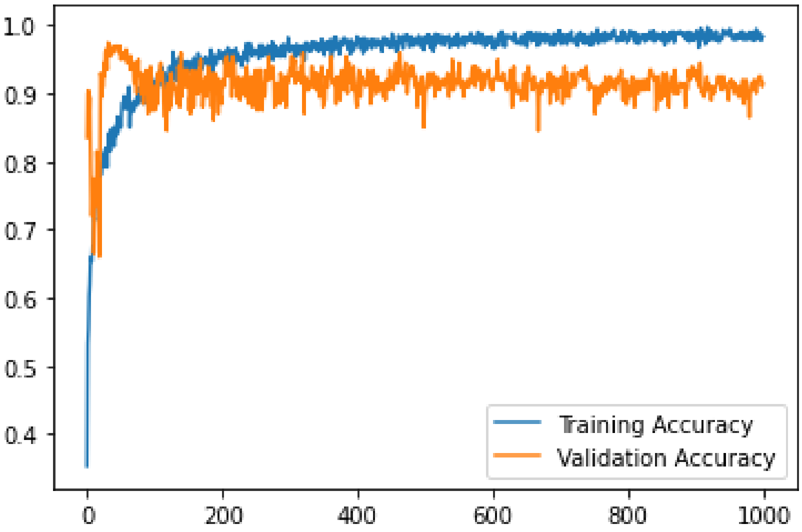


Fig. 4: Results for MLP for training data Set with 1000 elements (extracted features = 16): Accuracies vs epochs

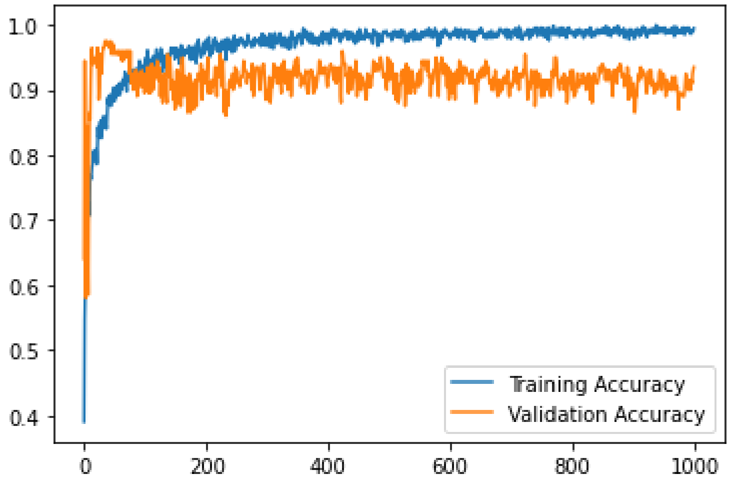


Fig. 5: Results for MLP for training data Set with 1000 elements (extracted features = 32): Accuracies vs epochs

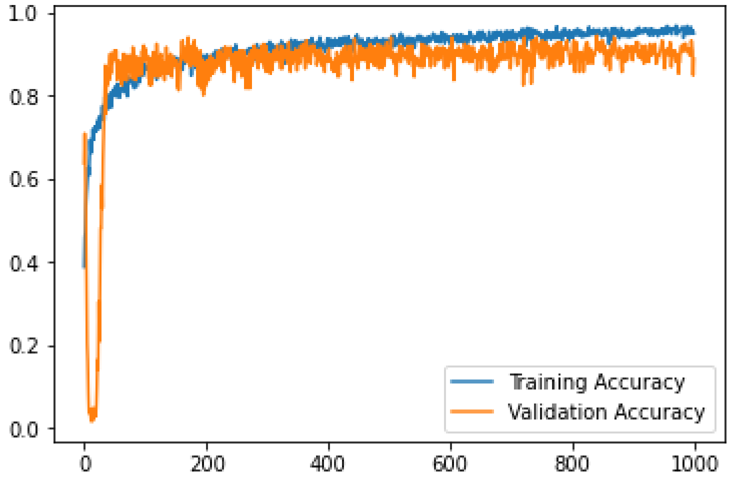


Fig. 6: Results for MLP for training data Set with 1250 elements (extracted features = 8): Accuracies vs epochs

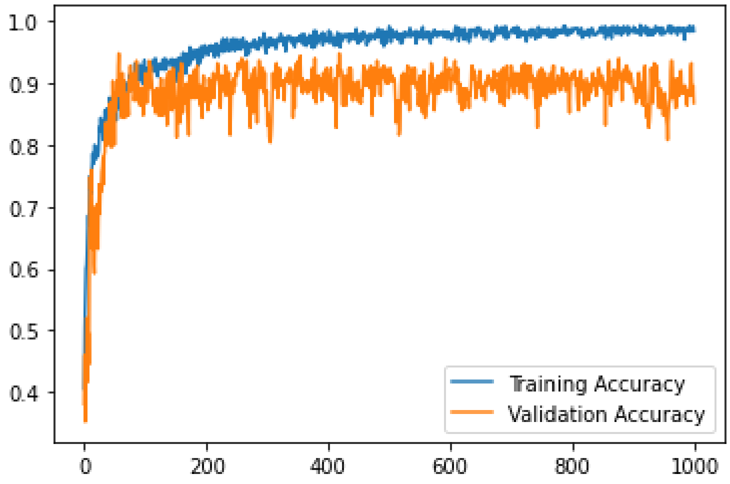


Fig. 7: Results for MLP for training data set with 1250 elements (extracted features = 16): Accuracies vs epochs

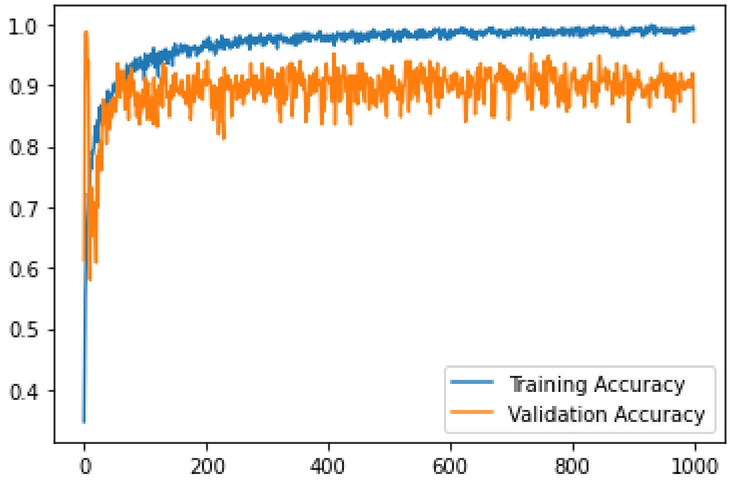


Fig. 8: Results for MLP for training data set with 1250 elements (extracted features = 32): Accuracies vs epochs

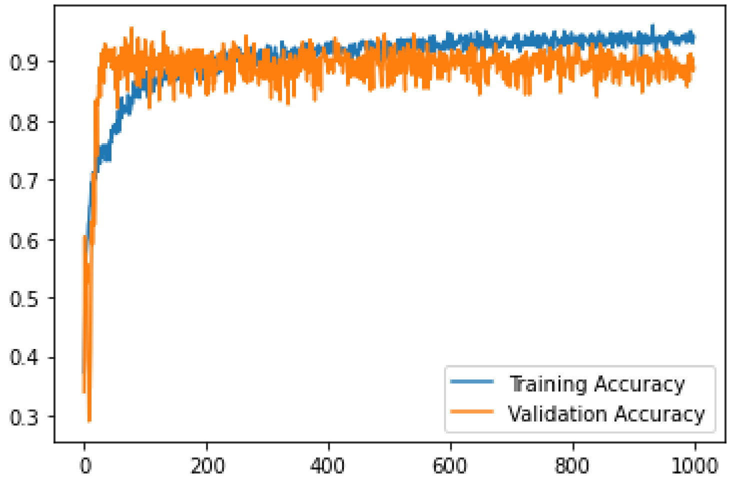


Fig. 9: Results for MLP for training data set with 1500 elements (extracted features = 8): Accuracies vs epochs

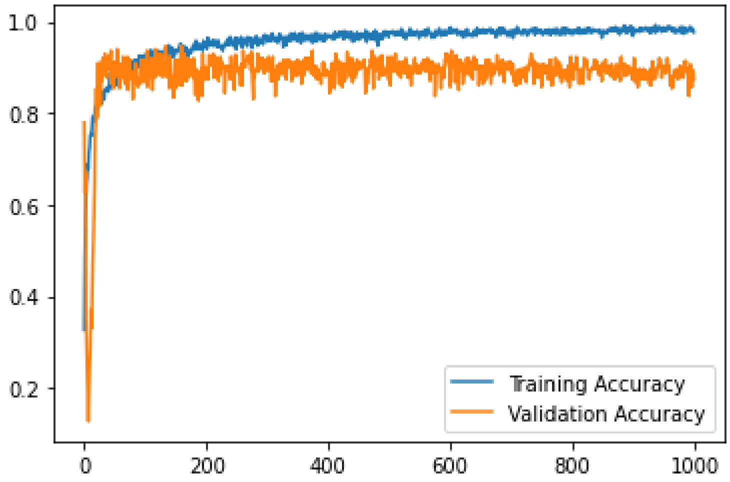


Fig. 10: Results for MLP for training data set with 1500 elements (extracted features=16): accuracies vs epoch

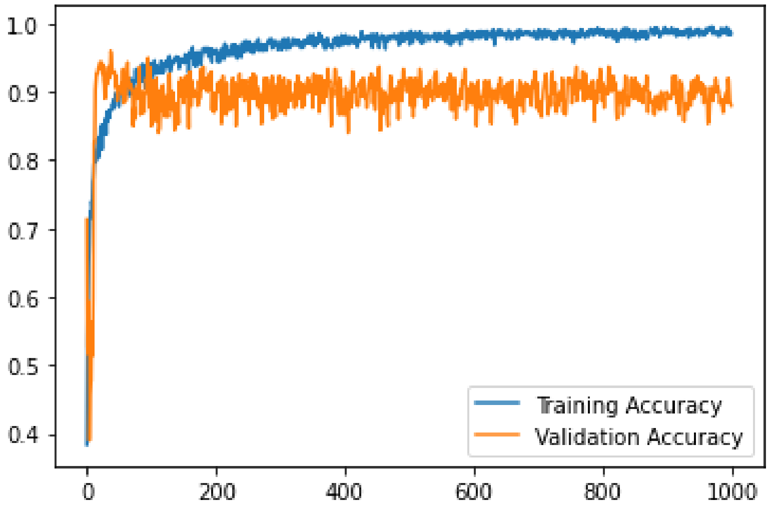
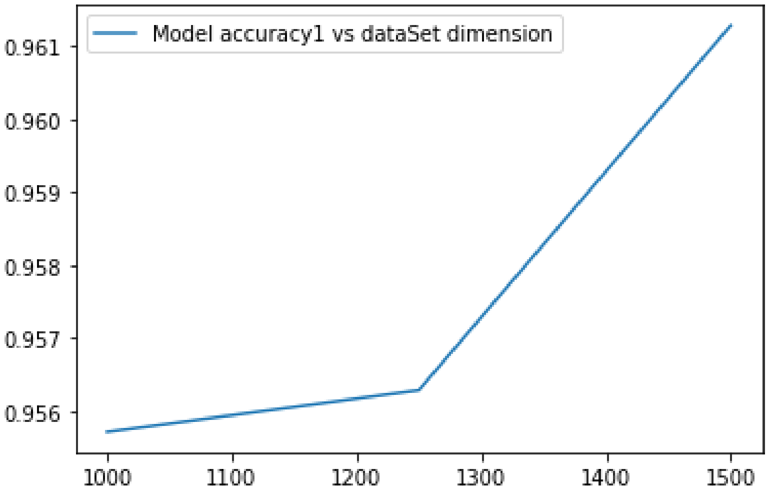
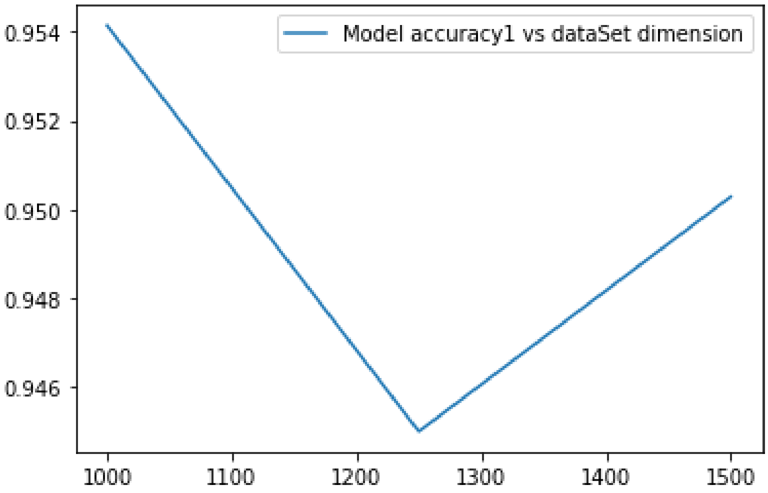


Fig. 11: Results for MLP for training data set with 1500 elements (extracted features=32): accuracies vs epochs





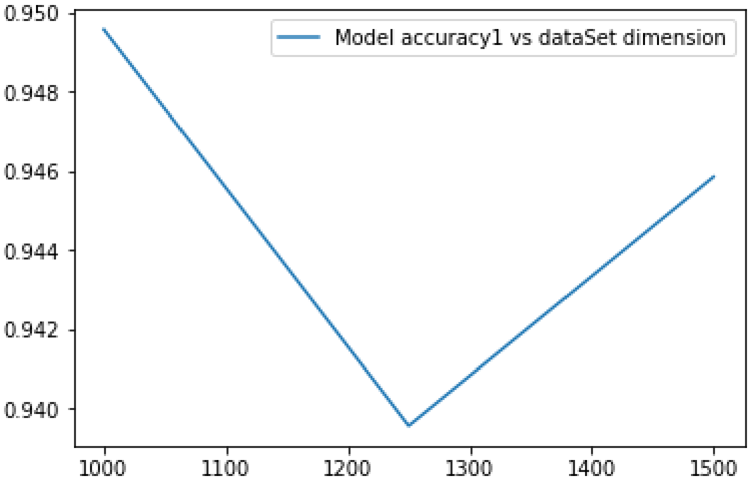


Fig. 12: Model accuracy1 vs data set cardinality (from top: Extracted features from MLP 8, 16, 32)

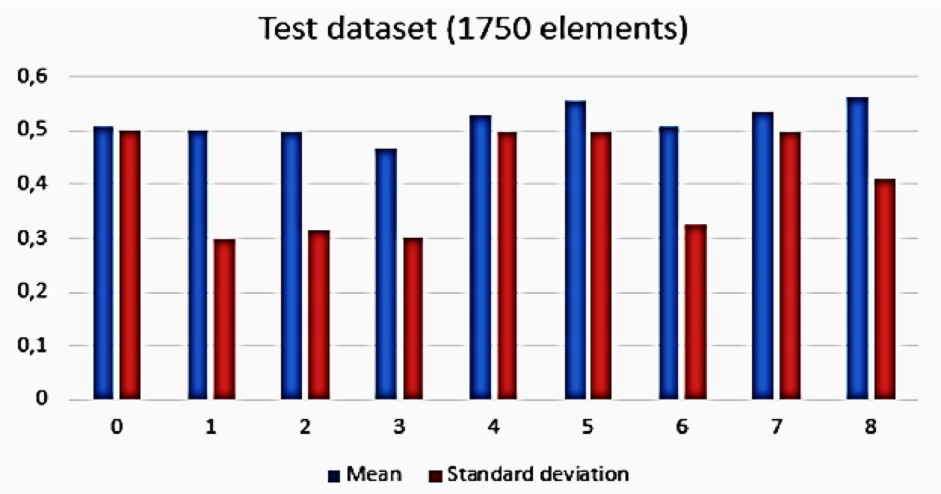
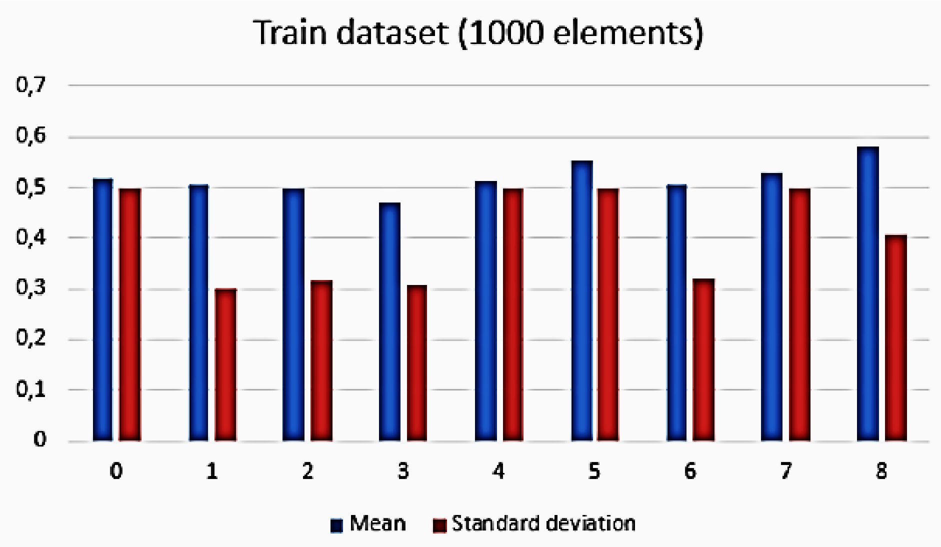


Fig. 13: Data distribution for test dataset (1750 elements) (*)



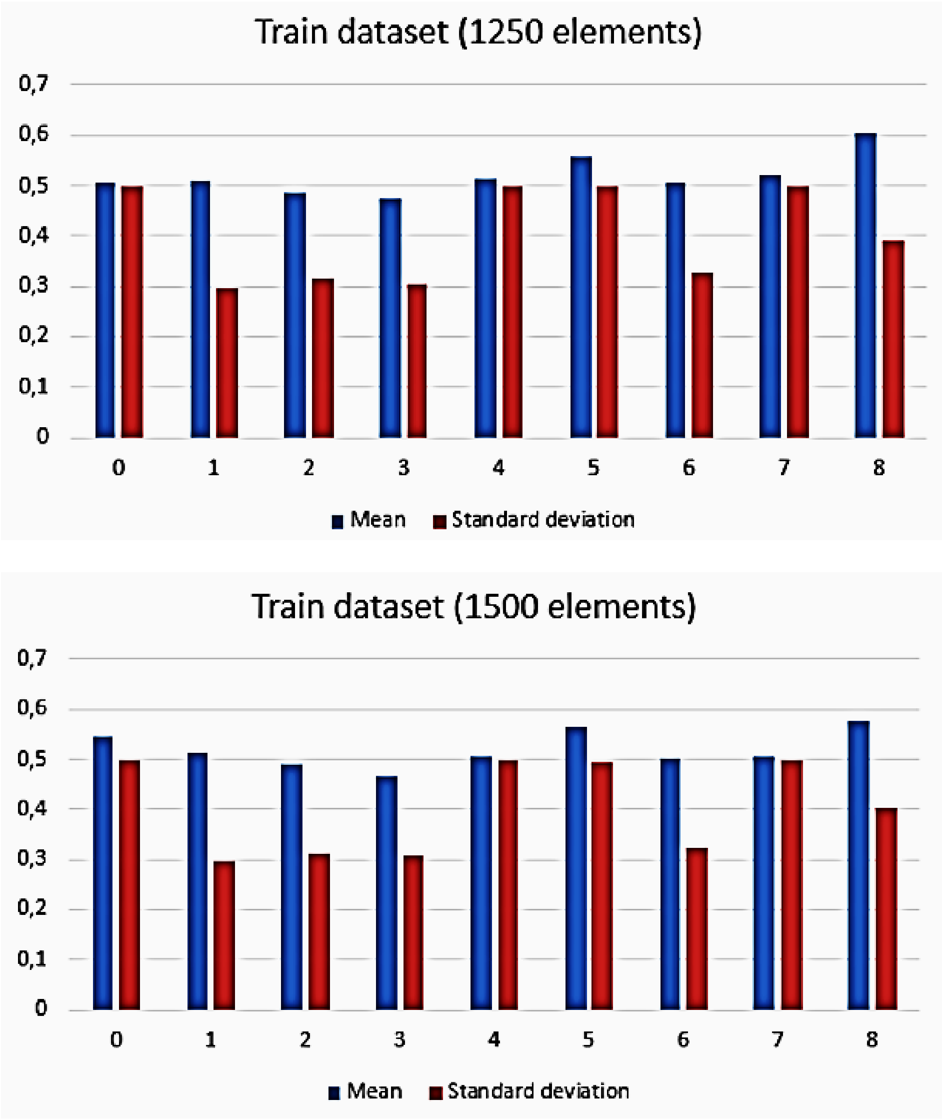


Fig. 14: Data distribution for training datasets (1000, 1250, 1500 elements) (*)

Discussion and Future Work

Diabetes is one of the most important disease. It is very important for a MD to understand the presence of a general disease in early stages. Predicting the disease at the beginning is very important to save the life of a person. Preventing a disease is another important issue that can be very useful from healthcare, social and economic point of views. Generally, DL models can predict diseases with interesting results. Our model is a contribution in trying to provide accurate results in the context of preventive medicine. In particular, it aims to predict the risk for a person to contract the disease of diabetes so to give the possibility to a MD to suggest him better lifestyles, more health controls and laboratory tests. We trained our diabetes prevention model, using randomly generated datasets produced in this study, according to FINDRISC. Same consideration is valid for testing dataset too. This is important because in this way, we are able to start with a significant dataset kernel and without any privacy problems related to real data and consents. Of course, the model could become more useful (e.g., overcoming the usage of FINDRISC) when it retrains again, adding real data during an actual usage and when it expands with a higher number of input attributes too. The level of accuracy1 for classification component is mainly due to the quality of features values extracted from MLP. We defined a MLP with no under fitting or over fitting problems. All accuracies values (the definition of accuracy considers class similarity too) for the overall model are greater than 0.939, with best results over 0.96 for 1500 labelled elements (training dataset). Another initial result of our research is the definition of a method to explain our model in terms of input attributes. In future work, we want to better analyze our model, so to define H function (see Explainability section) in order to combine the behavior of MLP with the variability of a feature in the classification component for training dataset, before implementing and testing the explain ability.

Funding Information

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Author's Contributions

Francesca Fallucchi: Designed the research plan and organized the study; coordinated the data-analysis and contributed to the writing of the manuscript.

Alessandro Cabroni: Designed the research plan and organized the study; coordinated the data-analysis and contributed to the writing of the manuscript; participated in all the experiments.

Ethics

This paper neither has been published nor is under review elsewhere.

References

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one, 10(7), e0130140. https://doi.org/10.1371/journal.pone.0130140

Chandrakar, O., & Saini, J. R. (2016, October). Development of Indian weighted diabetic risk score (IWDRS) using machine learning techniques for type-2 diabetes. In Proceedings of the 9th Annual ACM India Conference (pp. 125-128). doi.org/10.1145/2998476.2998497

Contreras, I., & Vehi, J. (2018). Artificial intelligence for diabetes management and decision support: literature review. Journal of medical Internet research, 20(5), e10775. www.jmir.org/2018/5/e10775/

Erdem, A., & Pelillo, M. (2012). Graph transduction as a noncooperative game. Neural Computation, 24(3), 700-723. https://ieeexplore.ieee.org/abstract/document/6797349/

Fallucchi, F., Petito, M., & De Luca, E. W. (2018, October). Analysing and visualising open data within the data and analytics framework. In Research Conference on Metadata and Semantics Research (pp. 135-146). Springer, Cham. doi.org/10.1007/978-3-030-14401-2_13

Hasan, M. K., Alam, M. A., Das, D., Hossain, E., & Hasan, M. (2020). Diabetes prediction using ensembling of different machine learning classifiers. IEEE Access, 8, 76516-76531. doi.org/10.1109/ACCESS.2020.2989857

Holzinger, A., Langs, G., Denk, H., Zatloukal, K., & Müller, H. (2019). Causability and explainability of artificial intelligence in medicine. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(4), e1312. doi.org/10.1002/widm.1312

Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., Chouvarda, I. (2017). Machine Learning and Data Mining Methods in Diabetes Research. Computational and Structural Biotechnology Journal, Volume 15, Pages 104-116, doi.org/10.1016/j.csbj.2016.12.005

Khanam, J. J., & Foo, S. Y. (2021). A comparison of machine learning algorithms for diabetes prediction. ICT Express. doi.org/10.1016/j.icte.2021.02.004

Kopitar, L., Cilar, L., Kocbek, P., & Stiglic, G. (2019). Local vs. Global Interpretability of Machine Learning Models in Type 2 Diabetes Mellitus Screening. In Artificial Intelligence in Medicine: Knowledge Representation and Transparent and Explainable Systems (pp. 108-119). Springer, Cham. doi.org/10.1007/978-3-030-37446-4_9

Lindström, J., & Tuomilehto, J. (2003). The diabetes risk score: a practical tool to predict type 2 diabetes risk. Diabetes care, 26(3), 725-731. https://doi.org/10.2337/diacare.26.3.725

Lindström, J., Absetz, P., Hemiö, K., Peltomäki, P., & Peltonen, M. (2010). Reducing the risk of type 2 diabetes with nutrition and physical activity–efficacy and implementation of lifestyle interventions in Finland. Public health nutrition, 13(6A), 993-999. doi.org/10.1017/S1368980010000960

Makrilakis, K., Liatis, S., Grammatikou, S., Perrea, D., Stathi, C., Tsiligros, P., & Katsilambros, N. (2011). Validation of the Finnish diabetes risk score (FINDRISC) questionnaire for screening for undiagnosed type 2 diabetes, dysglycaemia and the metabolic syndrome in Greece. Diabetes & metabolism, 37(2), 144-151. doi.org/10.1016/j.diabet.2010.09.006

Mercaldo, F., Nardone, V., & Santone, A. (2017). Diabetes mellitus affected patients classification and diagnosis through machine learning techniques. Procedia computer science, 112, 2519-2528. doi.org/10.1016/j.procs.2017.08.193

Miotto, R., Li, L., Kidd, B. A., & Dudley, J. T. (2016). Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. Scientific reports, 6(1), 1-10. https://www.nature.com/articles/srep26094

Montavon, G., Lapuschkin, S., Binder, A., Samek, W., & Müller, K. R. (2017). Explaining nonlinear classification decisions with deep taylor decomposition. Pattern Recognition, 65, 211-222. doi.org/10.1016/j.patcog.2016.11.008

Noble, D., Mathur, R., Dent, T., Meads, C., & Greenhalgh, T. (2011). Risk models and scores for type 2 diabetes: systematic review. Bmj, 343. doi.org/10.1136/bmj.d7163

Perveen, S., Shahbaz, M., Keshavjee, K., & Guergachi, A. (2019). Prognostic modeling and prevention of diabetes using machine learning technique. Scientific reports, 9(1), 1-9. doi.org/10.1038/s41598-019-49563-6

Pottie, K., Jaramillo, A., Lewin, G., Dickinson, J., Bell, N., Brauer, P., Dunfield, L., Joffres, M., Singh, H., Tonelli, M. (2012). "Recommendations on screening for type 2 diabetes in adults." CMAJ: Canadian Medical Association journal = journal de l'Association medicale canadienne, 184,15 (2012): 1687-96. doi.org/10.1503/cmaj.120732

Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., & Müller, K. R. (Eds.). (2019). Explainable AI: interpreting, explaining and visualizing deep learning (Vol. 11700). Springer Nature. https://doi.org/10.1007/978-3-030-28954-6_10

Sisodia, D., & Sisodia, D. S. (2018). Prediction of diabetes using classification algorithms. Procedia computer science, 132, 1578-1585. https://doi.org/10.1016/j.procs.2018.05.122

Swapna, G., Vinayakumar, R., & Soman, K. P. (2018). Diabetes detection using deep learning algorithms. ICT express, 4(4), 243-246. doi.org/10.1016/j.icte.2018.10.005

Tigga, N. P., & Garg, S. (2020). Prediction of type 2 diabetes using machine learning classification methods. Procedia Computer Science, 167, 706-716. doi.org/10.1016/j.procs.2020.03.336

Tjoa, E., & Guan, C. (2020). A survey on explainable artificial intelligence (xai): Toward medical xai. IEEE Transactions on Neural Networks and Learning Systems. https://ieeexplore.ieee.org/abstract/document/9233366

Xiong, X. L., Zhang, R. X., Bi, Y., Zhou, W. H., Yu, Y., & Zhu, D. L. (2019). Machine learning models in type 2 diabetes risk prediction: Results from a cross-sectional retrospective study in Chinese adults. Current medical science, 39(4), 582-588. doi.org/10.1007/s11596-019-2077-4

Zhang, L., Zhang, Z., Zhang, Y., Hu, G., & Chen, L. (2014). Evaluation of Finnish Diabetes Risk Score in screening undiagnosed diabetes and prediabetes among US adults by gender and race: NHANES 1999-2010. PloS one, 9(5), e97865. doi.org/10.1371/journal.pone.0097865