2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE) | 978-1-6654-9260-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/IITCEE57236.2023.10090957

Comparative Performance Analysis of Quantum Algorithm with Machine learning Algorithms on Diabetes Mellitus

BBNS. Prakash[1], B. Naveen[2], MD. Akhiluzzama[3], Pothuraju Rajarajeswari[4]

[1,2,3]Bachelor of Technology, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India.

[1]praakash.79@gmail.com, [2]naveennani14341@gmail.com, [3]mdakheeluzzama@gmail.com

[4] Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India.

Email:rajilikhitha@gmail.com

Abstract- This paper shows the distinction between Quantum and Classical Machine Learning techniques appeal to a diabetic mellitus dataset. Diabetes mellitus is like series of diseases that affect the body and deplete blood sugar levels (insulin). In our bodies, glucose is an important source of energy for powering mitochondria, the cells' function that make muscles and tissues strong. We use many machine learning classifiers such as Support Vector Machine, Kernel Principal Component Analysis, Bayesian Network and Decision Tree etc. These models are able predict a certain amount of data, while in case of large data there is significant amount of error and low accuracy rate, A new method known as quantum machine learning (QML) is nothing but the collaboration of machine learning in the way of quantum algorithms the term used for machine learning algorithms executed in a quantum computer for analysis of classical data is known as quantum-enhanced machine learning. In this method we are predicting diabetes mellitus using quantum machine learning algorithms which able to handle the huge data with high accuracy rate of 97% and F1-Score of 68%.Our study having implemented models of Predicted & Enhanced models.

Index Terms - Kernel Principle Component Analysis (KPCA), Diabetes Mellitus, Support Vector Machine (SVM), Quantum Computing, Quantum Machine Learning Algorithms (QMLA).

I. INTRODUCTION

Diabetes mellitus, generally appertained to simply as diabetes, is a metabolic issue that causes high blood sugar. For the energy purpose the sugar is replaced from blood to cells by hormone insulin. With diabetes, your body either does not make enough insulin or can not effectively use the insulin it does make undressed high blood sugar from diabetes can damage your jitters, eyes, feathers, and other organs. There's a lot of increment who are suffering with diabetes from past ages, that is increased to 422 million in 2014 when compared to 108 million in 1980 .Frequency has been rising more fleetly economically developing countries than in economically balanced countries. In contrast, the likelihood of dying between the ages of 30 and 70 from any of the four major non-infectious conditions (cardiovascular conditions, cancer, habitual respiratory conditions or diabetes) dropped from 2000 to 2019 by 22% globally during the year. To reduce convolutions and ameliorate the quality of Type 2 diabetes mellitus patient's life, there is rapid growth of healthcare expenses and even some research's demonstrating the control of glycaemic level of the indicative factors to help organ damage and other complications in diabetes mellitus. With type 2 diabetes we can say in some situations blood sugar can be reduced in people so that in decreasing the ailment and risk factor. several research's developed models to predict the accurate diseases on the patients utilizing Machine Learning and Deep Learning for T2DM. All the classical approaches in regular computer of binary system have shown the uncertainty and inaccuracy in prediction. This paper is regarding the necessity of using Algorithms of Quantum Computing. Quantum processing Unit (QPU) vs Classical Binary Approached Computers. The QML and ML algorithms such as KPCA, SVM, Decision tree and Bayesian network apply and compare diabetes datasets. delicacy and performance with traditional machine learning classifiers(KPCA, decision trees, SVM, Bayesian networks). This paper is divided into his six sections, Section I deals with his preface to QML, ML and Diabetes. Section II is about case status data and prognosticating variables in the data set. Section III introduces amount and classical machine learning algorithms. Section IV describes the armature of the trial and the refinement of the analysis and results. V and VI is the difference in quantum vs classical algorithms and conclusion of the prediction

II.APPLICATIONS

There are many fields and practices that are usually getting benefitted by sentiment analysis or opinion mining some of the popular applications are

- Medical Sector
- Predicting Diabetes
- Analysing Parents Genetics

III.LITERATURE SURVEY

LITERATURE REVIEW

Author	Method/Approach	Title, Journal & Year	Findings
Jarrod R. McClean, Nicholas C. Rubin, Joonho Lee, Matthew P. Harrigan, Thomas E. O'Brien, Ryan Babbush, William J. Huggins, and Hsin-Yuan Huang.	Rapid development of quantum technology; DIGITAL CHEMISTRY EXPERIMENTS WITH QUANTUM COMPUTERS, LIMITS OF EVEN QUANTUM COMPUTATIONS IN CHEMISTRY.	**What the foundations of quantum computer science teach us about chemistry** Google Quantum AI (2021)	A direct reading of the no-fast-forwarding theorem contrasted with the unassailable predictive success of simplified chemical models raises a...number... of interesting questions. On the one hand, the no-fast-forwarding theorem tells us that even with an exponential advantage over classical simulation of quantum dynamics, a quantum computer aiming to simulate the dynamics of a full wavefunction to a fixed accuracy is bound by some multiplier of the physical time of the actual process.
William J. Huggins,Ryan A. O'Gorman, Nicholas C. Rubin, David R. Reichman, Ryan Babbush, and Joonho Lee.	We experimentally implement our scheme using up to 16 qubits in order to unbias constrained QMC calculations performed on chemical systems with as many as 120 orbitals.	**Unbiasing Fermionic Quantum Monte Carlo with a Quantum Computer** Google Quantum AI...(2020)	Theory and algorithms. Quantum Monte Carlo (QMC) approaches16,17 target the exact ground state |Ψ0 ⟩ of a many-body Hamiltonian, $H_{\text{...}}$ via imaginary time evolution of an initial state |Φ0 ⟩ with a non-zero overlap with |Ψ0 ⟩: |Ψ0 ⟩ ∝ lim$_{\tau\to\infty}$ $e^{-\tau H}$|Ψ(τ)⟩, |Ψ(τ)⟩ = $e^{-\tau H}$ |Φ0 ⟩,
M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii	Finding ground and excited states Variational Quantum Eigensolver (VQE) implementation.	**Variational Quantum Algorithms** Google Quantum AI (2019)	Schematic diagram of an ansatz. The unitary U(θ) can be expressed as a product of L unitaries Uj(θj) sequentially acting on an input state. As indicated, each unitary Uj(θj) can in turn be decomposed into a sequence of parametrized and unparametrized gates U(θ) = UL(θL) · · · U2(θ2)U1(θ1), with Uj(θj) = Y m e −iθmHmWm
JOSEPH C. BARDIN, DANIEL H. SLICHTER3, DAVID J. REILLY.	QUBITS AND QUANTUM COMPUTING. QUBITS AS RESONATORS INTERFACING A MICROWAVE SOURCE TO A QUBIT.	**Microwaves in Quantum Computing** Google Quantum AI (2020)	Modern nanofabrication makes it possible to confine and detect single electron spins in "zero-dimensional" nanostructures referred to as quantum dots (QDs). The potential that confines the electron is produced electrostatically via gate electrodes on the surface of a semiconductor, enabling the number of electrons on a dot and their coupling to the neighboring dots and reservoirs

			to be tuned by varying gate voltages
YuanSu,Dominic W. Berry, Nathan Wiebe, Nicholas Rubin, and Ryan Babbush	Circuit implementation of qubitization. Block-encoding the kinetic operator 18. Implementation choices for the interaction picture algorithm.	**Fault-Tolerant Quantum Simulations of Chemistry in First Quantization.** IEEE Access (2019)	In order to simulate molecular systems on a computer one must discretize them in some fashion. For reasons discussed in more depth in Appendix B, most molecular modeling employs a Galerkin representation which involves projecting the system onto some well behaved set of basis functions {φp(r)}. The matrix elements of the Hamiltonian operators are then given by the following integrals over these basis functions: T (m) pq = Z dr φ∗ p (r) = ∇2 2m φq (r)
Matthew P. Harrigan 1 , Kevin J. Sung1,2, Matthew Neeley 1 , Kevin J. Satzinger 1	Faster algorithms for combinatorial optimization could prove transformative for diverse areas such as logistics, finance and machine learning. Accordingly, the possibility of quantum enhanced optimization has driven much interest in quantum technologies.	**Quantum-approximate optimization of non-planar graph problems on a planar superconducting processor** Nature Physics (2020)	In this study, we consider three families of binary optimization problems typified by their graph representation. First, we study problem graphs that match the connectivity of our hardware, which we term 'hardware grid problems'. This family of problems is composed of random instances generated by sampling wjk to be ±1 for edges in the device topology or a subgraph thereof, as depicted in Fig. 1a. While formally NP-hard12 (and thus, unlikely to be efficiently solvable in the worst case), problems defined on these graphs with couplings chosen in this fashion are known to be classically efficient to exactly solve on average27. However, we study these problems here as they are a simple example of a problem that does not require routing.
Jarrod R. McClean, * Matthew P. Harrigan , Masoud Mohseni, Nicholas C. Rubin, Zhang Jiang, Sergio	The goal of this work is to highlight and improve mechanisms that occur even for low-depth quantum circuits, to enable better engineering of quantum optimization.	**Low-Depth Mechanisms for Quantum Optimization** PRX Quantum (2021)	Quantum annealing [14] works to solve this optimization problem through connection with the adiabatic principle by evolving under Hamiltonians of the form H = s(t) z f (z)|z—— z| − [1 − s(t)] i Xi,2 At a conceptual level, the graph Laplacian is the realspace Laplacian used in quantum mechanics for the kinetic energy, generalized
Boixo , Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven			to generic and discrete graphs. In particular, for optimization problems, we can consider them as evolving under a potential and kinetic energy defined by T = LG
Ryan Babbush, * Jarrod R. McClean,† Michael Newman, Craig Gidney, Sergio Boixo , and Hartmut Neven	RELATIONSHIP BETWEEN PRIMITIVE TIMES AND RUNTIME IMPLEMENTING ERROR-CORRECTED QUANTUM PRIMITIVES	**Focus beyond Quadratic Speedups for Error-Corrected-Quantum Advantage** IRJET (2018)	Many quantum algorithms are built on coherent access to primitives implemented with classical logic. For example, this classical logic might be required to compute the value of a classical cost function for optimization [12], to evaluate a function of a trajectory of some security that one is pricing with a Monte Carlo method [9], or to compute some classical criterion that flags a marked state for which one might be searching [5]. We define the runtimes of the quantum and classical algorithms as TQ = M tQ, TC = Md tC,
Sorrachai Yingchareonthawornchai , Chatchawit Aporntewan , Prabhas Chongstitvatana Z Department of Computer Engineering	THE MAPPING OF COMPACT GENETIC ALGORITHM – QUANTUM CONSTRUCTION. EXPERIMENTAL RESULT FOR QUANTUM ALGORITHM CONSTRUCTION.	**An Implementation of Compact Genetic Algorithm on a Quantum Computer** ACM (2020)	There have been the relevant attempts between quantum and genetic algorithms. For example, Quantum Genetic Optimization Algorithm has introduced an optimization of classical genetic algorithm using the principles of quantum search which provided a significant speed-up on each genetic step. There are many variations of genetic algorithms. The compact genetic algorithm is one of them.
K.I.SATZINGER, Y.JIU, A.SMITH, C.KNAPP, M.NEWMANC JONESZ, CHENC, QUNITANAX	Preparation of the ground state Measurement of topological entanglement entropy	**Realizing topologically ordered states on a quantum processor.** ACM (2022)	Our approach of directly preparing the toric code ground state also allows us to simulate the exotic braiding statistics of its quasiparticle excitations (anyons). We used a mapping between the adiabatic evolution of toric code excitations and strings of Pauli operators applied to the ground state.

IV. Theoretical Analysis

Lately, Google blazoned TensorFlow Quantum(TFQ) an open- source library for amount machine literacy, in collaboration with the University of Waterloo, X, and Volkswagen. The end of TFQ is to give the necessary tools to control and model natural or artificial amount systems. TFQ is an illustration of a suite of tools that combines amount modelling and machine literacy ways to prognosticate Diabetes analysis.

1. Convert amount data to the amount dataset: Quantum data can be represented as a multi-dimensional array of figures which is called as amount tensors. TensorFlow processes these tensors in order to represent produce a dataset for farther use.

2. Choose amount neural network models: Grounded on the knowledge of the amount data structure, amount neural network models are named. The end is to perform amount processing in order to prize information hidden in an entangled state.

3. Sample or Average dimension of amount: countries excerpts classical information in the form of samples from the classical distribution. The values are attained from the amount state itself. TFQ provides styles for comprising over several runs involving way(1) and(2).

4. Estimate a classical neural networks model :– Since amount data is now converted into classical data, deep literacy ways are used to learn the correlation between data.

V. Algorithms & Methods

In this process of analysis we have used some of the AI algorithms for data extraction and predicting purposes.

- Kernel Principal Component Analysis
- Decision Tree
- Support Vector Machine
- Bayesian Network
- Q Boost Classifier

➢ **Kernel Principal Component Analysis:**

In multiple classifiers Kernel Principal Component Analysis(KPCA) is a linear method. This method is used to diagonalize a target value of covariance matrix It can only be implied to Datasets that are linearly separable. To eject dataset into a exorbitant dimensional feature space KPCA uses kernel function which is linearly separable. The Function below represents the KPCA Model.

$$\mathbf{V^{kT}\ \phi(x) = \left(\sum_{i=1}^{N} a_i^k \Phi(x_i)\right)^T \Phi(x)}$$

We note that $\Phi(x_i)^{\mathrm{T}}\ \Phi(x)$ denotes fleck product, that is like the rudiments of the kernel Component K. It means all from left wing is to calculate be done by working with the eigenvector equation

$$\mathbf{N\lambda_a = K_a}.$$

➢ **Decision Tree:**

Learning from decision trees is a wisely used technique in data mining. The resultant need to develop a replica that analysis the rate of a point variable based on numerous input variables. Data is provided in records of the form:

$$\mathbf{(x, Y) = (x_1+x_2+x_3+\text{------}+x_k, Y)}$$

The major value, Y {\display style V} Y, is the resultant value that is to be learned, classify or conclude. The vector is proposed by using features, x1, x2, x 3 etc., that are required for the solution.

$$\mathbf{F(x) = sign(\sum_{i=0}^{N} A_n f_n(x))}$$

$$\boldsymbol{F(x) = sign(A_n * x + c)}$$

where f(n) is the purpose of detecting feature, A_n refers to weights and c is to be optimized called biasing.

- **Support Vector Machine:**

Support Vector Machine(SVM) is a supervised learning algorithm in machine learning which is vastly used in data classification and regression techniques, The classification is based on boundary which divided into categories using a hyperplane, therefore positive hyperplane and a negative hyperplane having support vectors.

The joint function is useful for extending SVMs when the data can not be separable linearly . $\boldsymbol{\max(0, 1 - y_i(w^T x_i - b))}$.

Note that y_i is the i^{th} goal (i.e,. in this case 1 or -1), and $w^T x_i$-b is the i^{th} resultant value. The value of the function is zero if the condition in (1) is full-filled, that is, if xi is to the right of the boundary. For data behind an edge, the value of the function is commensurable is to get the distance from the edge. The goal of optimization is to minimization.

$$\boldsymbol{\lambda||w||^2 + [\frac{1}{n}\sum_{i=1}^{n} \max(0, 1 - y_i(w^T x_i - b))}$$

here constraint λ > 0 dictate the trade- off between adding the periphery area and icing that the x_i taradiddle to the correct side of periphery. Therefore, for sufficiently tiny values of λ, it'll bear analogous to the hard-periphery SVM, if input data can be classifiable linearly, but will still learn if a bracket rule is feasible or not.(This parameter λ is frequently also called C, e.g., in LIBSVM, but generally refers to antipode of λ.).

- **Bayesian Network:**

Bayesian-Networks were a kind of probabilistic graphical model represents set of variables and they are these networks are from probabilistic distributions are the conditional dependencies uses the directed acyclic graph. Bayesian networks point to model conditional dependence by edges in a directed graph.

$P(X_1,X_2,\ldots,X_n)=P(X_1).P(X_2|X_1).P(X_3|X_2,X_1)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..P(X_n|X_1,X_2,\ldots X_n)$

$$=\boldsymbol{\pi(X_i|Parents(X_i))}$$

- **Q Boost Classifier**

Qboost Classifier is the quantum type of XGboost classifier which is an ensemble way of learning method used in QA (Quantum Annealing). For the safe of using the uttermost power by Dwave Quantum Annealing (QA), the term objective function is necessary for Quadratic Unconstrained Binary Optimization. To obtain this we interpret AdaBoost as replacing standard weighted error function (WEF) with a Quadratic Unconstrained Binary Optimization(QUBO).

L(A) → Loss Function
R(A) → Regularization

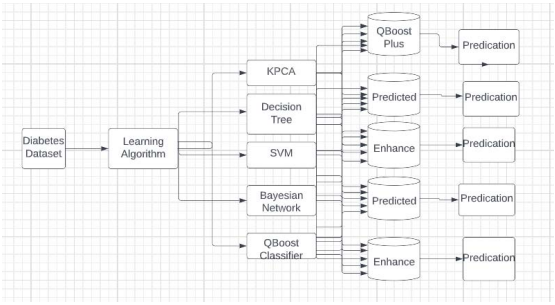
$$\boldsymbol{A^* = argmin_A(\sum_s \left(\frac{1}{N}(\sum_n^N A_n f_n(x_s) - y_s) \right)^2 + \lambda||A_0||}$$

where a(x) is a classifier which is strong one that iteratively integrates, selects, and reconstructs weak classifiers at each replication, and retrains to shrink the weighted error function(WEF). Their weights will be altered and refactored so that all weights sum to 1.

VI. Architecture Of The Experiment

This model is implemented using python 3 and QML with libraries of scikit learn. This Application Program Interface(API) is used to achieve the Dwave System QPU(quantum processing unit), where Dwave system Quantum Computing allows Jupiter notebook tool to run a quantum processing unit.

The data which is trained and applied to develop each machine learning classifiers, and performed testing data is required in order to verify and contrast the regular predicted classifiers labels with given labels known as test labels. The test labels are +1 and -1 representing diabetics with critical diabetes and 1 as diabetics with non-critical diabetes. our whole data is divided into 70% of training data and 30% of testing data. while ensuring the same percentage of the classes in each submodule(50% = =+1 and 50% = = -1)



VII. Algorithms And Evaluation:

In this algorithm we had used the classifiers of Decision Tree, Kernel principle component analysis(KPCA), Bayesian network, Support Vector Machine(SVM) with comparison of Quantum algorithm(Q Boost), the regression we have to introduce a metric to penalize differences of the prediction from the label Y.

Percent gives the maximum deviation of the prediction from the label that is not penalized. we create weak classifiers based on Decision Tree and strong classifiers as QBoost the functions of fit estimators of variables param X and parm Y. Weak Regressor based on Decision-Tree Regressor, Strong Regressor based on Classifier as QBoost Regressor, the four models Enhanced & Predicted models.

$$\boldsymbol{Precision} = \left(\frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} * \mathbf{100}\right)\%$$

$$\mathbf{Accuracy} = \left(\frac{\mathbf{TP+TN}}{\mathbf{TP+TN+FP+FN}} * \mathbf{100}\right)\%$$

$$\mathbf{Fi-Score} = \left(2 * \frac{\mathbf{Precision} * \left(\frac{\mathbf{TP}}{\mathbf{TP+FP}}\right)}{\mathbf{Precision} + \left(\frac{\mathbf{TP}}{\mathbf{TP+FP}}\right)} * \mathbf{100}\right)\%$$

$$\mathbf{Specificity} = \left(\frac{\mathbf{TN}}{\mathbf{TN+FP}} * \mathbf{100}\right)\%$$

Where FP signifies for false positive, TP represents as true positive, FN replicates false negative., and TN signifies true negative.

These tables represents the Precision, Sensitivity, Fi-Score, Specificity of the classifiers (kernel principle component analysis KPCA, SVM, Decision-tree, QBoost and weighted average, Execution time.

This entire data is predicted in percentage rate using ratio of probability and demonstrated in graphical way.

BOARD -I KPCA-SVM

Classes	Precision	Sensitivity	Fi-Score	Specificity
Diabetes with Critical Diseases	69%	75%	77%	80%
Diabetes without Critical Diseases	66%	71%	73%	78%
Weighted Avg	67%	73%	75%	79%
Accuracy	69%			
Time	0.809s			

BOARD -II DECISION TREE

Classes	Precision	Sensitivity	Fi-Score	Specificity
Diabetes with Critical Diseases	57%	67%	73%	75%
Diabetes without Critical Diseases	54%	63%	66%	69%
Weighted Avg	56%	65%	72%	71%
Accuracy	57%			
Time	0.512Xs			

BOARD -III QBOOST

Classes	Precision	Sensitivity	Fi-Score	Specificity
Diabetes with Critical Diseases	76%	81%	83%	88%
Diabetes without Critical Diseases	70%	75%	78%	81%
Weighted Avg	73%	78%	80%	85%
Accuracy	73%			
Time	0.0428s			

BOARD -IV BAYESIAN NETWORK

Classes	Precision	Sensitivity	Fi-Score	Specificity
Diabetes with Critical Diseases	**69%**	**69%**	**69%**	**69%**
Diabetes without Critical Diseases	**69%**	**68%**	**69%**	**69%**
Weighted Avg	**69%**	**69%**	**69%**	**69%**
Accuracy	**69%**			
Time	**0.379 & 17.69s**			

BOARD -V ENHANCE MODEL-I

Classes	Precision	Sensitivity	Fi-Score	Specificity
Diabetes with Critical Diseases	**67%**	**72%**	**70%**	**72%**
Diabetes without Critical Diseases	**70%**	**64%**	**67%**	**64%**
Weighted Avg	**68%**	**68%**	**68%**	**68%**
Accuracy	**68%**			
Time	**0.379 & 0.328s**			

BOARD -VI ENHANCE MODEL-II

Classes	Precision	Sensitivity	Fi-Score	Specificity
Diabetes with Critical Diseases	**71%**	**77%**	**79%**	**82%**
Diabetes without Critical Diseases	**68%**	**73%**	**75%**	**78%**
Weighted Avg	**69%**	**75%**	**77%**	**80%**
Accuracy	**69%**			
Time	**19.34s**			

Results and Conclusion :

The Following graph is the predicted model developed using mat-plot library and seaborn library which showing the accuracy, sensitivity, F1-score and specificity of the classifiers

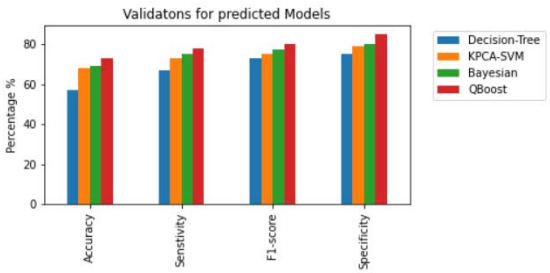
The Quantum Processing Unit(QPU) run time of the QBoost, Enhance Model 1 and Enhance Model 2 as 0.0428, 0.379 and 0.328sec respectively, from this prediction the quantum model is ~ 57 times rapid compared to classical models. To ensure Security it is very Robust and reliability in nature.

The results of the predictions are shown in Table 7. They're shown pictorially in graph shows that the produced Quantum algorithm present better in terms of Accuracy (73.4%), Sensitivity(78.1%), F1-score(80.3%) and Specificity. (85.7%) compared to other classification algorithms.

CLASSIFIERS	ACC	SN	F1	SP
QBOOST	73..356	78.152	80.264	85.692
KPCA-SVM	68.342	73.783	75.454	79.260
BAYESIAN	69.952	75.745	77.263	80.297
DECISION TREE	57.278	67.337	73.437	75.438

Table 7 : Comparison of Quantum & Classical Classifiers.

This is a new mongrel system developed from being effectiveness- tested algorithms. Kernel- grounded PCA is used to reduce indistinguishable data in the dataset. This helps to ameliorate the effectiveness in complaint discovery by furnishing high accuracy. With the other classifiers of binary computer the way quantum computing used in the quantum machine learning is both dependent on hardware and software as of quantum processing unit and QML, the Dwave should be in limit of QUBO(Quadratic unconstrained binary optimization) of achieving astonishing accuracy of 69% and 68.2% and F1-scores are 68% and 67.78% the enhance model 1 and enhance model 2 are optimized methods of QBoost classifier



Advancements:

In this Model we can even change the accuracy and precision by 10% - 15% making the function feasible which observing the patterns of the prediction in each dataset and optimizing the value in the limit of QUBO and developing an enhanced model to improve the readings which can be done by some changes in hardware field. E.g. changing the Quantum spin, intrinsic spin ½ degree of freedom of each particle in semi conductors. These are the optimal constraints for the advancements of the model.

References:

[1] Leon Kopitar, Primoz Kochbek, L. Cilar, A.Sheikh, Gregor Stigilic for Early Detection of type 2 diabetes mellitus using machine learning-based prediction models, article number: 11981, (20-July-2020), Scientific Reports.

[2] S. K. M. Wong and W. Ziarko, "On optimal decision rules in decision tables," *Bulletin of Polish Academy of Sciences,* vol33, 1985, pp. 693-696.

[3] G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller, "Scikit-learn," GetMobile Mob. Comput. Commun., vol. 19, no. 1, pp. 29–33, 2015.

[4] Danyal Maheshwari, Begona Garcia-Zapirain, Daniel Sierra-Soso for Machine learning applied to diabetes dataset using Quantum versus Classical computation, vol:10, doi={10.1109/ISSPIT51521.2020.9408944}}, 2020, Pp. 1-6.

[5] M G Dinesh and D Prabha for Diabetes mellitus using KPCA=GA – SVM feature selection techniques,2021 **DOI** 10.1088/1742-6596/1767/1/012001.

[6] Sushra Mishra, H. R. Kumar, M. Pradeep Kumar about an Enhanced and adaptive hybrid model for diabetes Mellitus *Sensors* **2020**, *20*(14), 4036; ttps://doi.org/10.3390/s20144036

[7] H. Gupta, Hideresh Varshney, Dr Tarun Kumar Sharma, Om prakash verma, regarding Comparative performance analysis of quantum machine learning with deep learning for diabetes prediction DOI: 10.1007/s40747-021-00398-7. 20-May-2021.

[8] Quan zou, Kaiyang Qu, Y. luo, Hua Tang about Predicting Diabetes Mellitus With Machine Learning Techniques Front. Genet., 06 November 2018 Sec. Computational Genomics https://doi.org/10.3389.

[9] V. Kumbargeri, Nitisha Sinha, Ch. Satyanadhan, regarding A Quantum Machine Learning Classifier Model for diabetes. DOI: 10.1007/978-981-15-9651-3_50.Feb-2021.

[10] Daniel Sierra-Sosa1,*, Juan D. Arcila-Moreno2, Begonya Garcia-Zapirain3 and Adel Elmaghraby1 about Diabetes Type 2: Poincaré Data Pre-processing for Quantum Machine Learning, Received: 17 July 2020; Accepted: 07 December 2020. DOI:10.32604/cmc.2021.013196.

[11] Z. Pawalak, for Rough Sets: Theoretical Aspects of Reasoning about Data "Dordrecht: Kluwer Academic Publishers 1991.

[12] Hang Lai, H. Huang, K. Keshvajee, Aziz Guergachi & Xin Gao, for Predictive models for diabetes mellitus using machine learning techniques, BMC Endocrine Disorders 19, Article No: 101, 15th oct 2019.

[13] C. B. D. Goes, Thiago O. Maciel, Giovani G, Pollachini, Rafel Cuenca, Juan, P.L.C. Salazar, E. I. Duzzioni, regarding QBoost for regression problems : solving partial differential Equations, https://arxiv.org/abs/2108.13346, 30th august 2021.

[14]H. Neven, V.S. Denchav, Geordie Rose, William Macready, concept- QBoost: Large scale classifier training with adiabatic quantum optimization, Jan-2012. Journal of Machine Learning Research 25:3333-348.