SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V11I8P115

Volume 11 Issue 8, 140-150, August 2024

Original Article

Data-Driven Insights into Gestational Diabetes Mellitus: Enhancing Models for Prediction by SVM Imputation for Personalized Pregnancy Care

T. Sujatha[1], K. R. Ananthapadmanaban[2]

[1,2]*Department of Computer Science, SRM Institute of Science and Technology, Vadapalani Campus, Chennai, India.*

[1]*Corresponding Author : tsujathasrm@gmail.com*

Received: 08 June 2024 Revised: 22 July 2024 Accepted: 11 August 2024 Published: 31 August 2024

Abstract *- Gestational Diabetes Mellitus(GDM) stands as a vital health concern for pregnant individuals worldwide. The onset or detection of elevated blood sugar levels during pregnancy, representing a form of glucose intolerance, characterizes it. The implications of GDM extend beyond maternal health, as it also poses risks to the developing fetus, potentially leading to adverse outcomes such as macrosomia, birth injuries, and an increased likelihood of caesarean delivery. Machine learning helps overcome the mentioned problems. This work evaluates the performance of different machine learning models as well as compares them with an existing system, particularly the K-Nearest Neighbors (KNN) model, in predicting GDM during pregnancy. This evaluation aims to determine whether KNN outperforms alternative models in accurately predicting GDM. The dataset contains 3525 records with 17 attributes, of which 16 are independent attributes, and one is an outcome attribute. For preprocessing these records, the SVM imputation method is implemented to replace missing records in the dataset. The KNN of the Lazy category produces an effective result with an accuracy of 96.96%, 97% precision, and 97% recall, which is an efficient result, and the Decision Table demonstrates the lowest efficiency with 95.97% accuracy, 96% precision, and 96% recall. The proposed system of the KNN model gives 96.96% accuracy, 97% precision, 97% recall, 97% F1-Score, 0.03 deviations, and 0.01 seconds of time complexity, whereas the existing KNN model had 85% accuracy, 83% precision, 84.96% recall, 84% F1-Score, 0.1503 errors, and 0.5355 seconds of time complexity. The work assesses classification metrics and regression metrics on multi-layer perceptron, random forest, Bayes net, decision table, and KNN models. The ultimate objective is to detect the most effective model for predicting GDM, which could improve the analysis and management of this medical complication during pregnancy.*

Keywords *- Gestational diabetes mellitus, KNN, MLP, Decision table, Random forest.*

1. Introduction

Both the mother and the unborn child are obstructed by Gestational Diabetes Mellitus (GDM), making it a maternal and fetal healthiness problem. Interventions such as lifestyle changes, medication, or insulin therapy follow a risk assessment based on clinical indicators in the traditional method of GDM management [1-4]. Nevertheless, new possibilities to improve GDM prediction, diagnosis, and management have emerged thanks to the proliferation of health data and developments in ML approaches. Healthcare solutions that are both data-driven and tailored to the individual patient are made possible by machine learning, which uses computational algorithms to identify patterns in massive datasets and provide predictions. To stratify the risk of Gestational Diabetes Mellitus (GDM), ML algorithms can examine a wide range of data, such as maternal demographics, medical history, anthropometric measurements, biochemical markers, and genetic predispositions. In comparison to more conventional risk assessment methods, ML-based systems may be able to increase the precision of GDM prediction by combining data from a variety of sources [5-7].

In addition, ML methods can help find Gestational Diabetes Mellitus (GDM) earlier by spotting minor trends in maternal health data that can be there before the disease shows up clinically [8,9]. The dangers of uncontrolled hyperglycemia during pregnancy can be reduced with early treatments, such as nutritional counseling or glucose monitoring, made possible by early prediction of GDM. Furthermore, by evaluating patient-specific data, ML algorithms can help personalize GDM management strategies, optimize treatment regimens, and improve maternal and fetal outcomes. [1-9] Multiple research studies have provided evidence for utilising machine learning models in envisaging the threat of GDM and making treatment decisions. These models incorporate a range of ML methods, each with its own set of benefits in terms of interpretability, scalability, and predictive performance. It is worth noting that GDM

prediction models can be even more accurately and broadly used when ML is combined with other computational methods like deep learning or genetic algorithms. Concerns about data quality, the interpretability of models, and clinical implementation are obstacles to ML's use in GDM prediction and management, notwithstanding the advantages. Deploying ML-based healthcare solutions also necessitates meticulously examining ethical factors, privacy problems, and legal restrictions. However, there is great promise that ML, combined with clinical knowledge, can change the game for GDM treatment, opening the door to precision and individualized therapy that caters to each patient's unique needs.

Usually, researchers apply only the traditional method for replacing the missing value for preprocessing the borrowed dataset, which has a missing value, but this proposed system focuses on utilizing the SVM model to fill the empty values in the dataset. The primary objective of SVM classification is to generate the most optimal output by utilizing a selection of models.

Below is an outline of this paper's primary contributions.

- Implement Support Vector Machine imputation to recover missing values and get a perfect dataset.
- Use a variety of classification and regression metrics to predict gestational diabetes mellitus using selected classifiers.

This work is organized as follows: Section 2 presents a review of related research, Section 3 describes materials and techniques, Section 4 displays results and analysis, Section 5 concludes the work, and Section 5 provides a description of the conclusion.

2. Literature Survey

This section focuses on analyzing prior research that pertains to the subject matter of this study. Data from 82,698 pregnant women in the Japan Environment and Children's Study birth cohort were used in this study. Statistical power, data availability, machine learning technique comparison, GDM factor discovery, decision tree-based algorithm correctness and interpretability, and GDM development factor investigation are all enhanced by big sample size [10]. The study employed Data Confederation (DC) analysis to combine health checkup data from 1502 Tsukuba City citizens with health history data from 1399 University of Tsukuba Hospital patients, improving modeling while safeguarding information (Uchitachimoto G et al., 2011). LR and GBDT achieved high recall rates and ROC-AUC scores of 0.858 and 0.856, respectively, using only health checkup data. It has LR's performance (ROC-AUC: 0.875, recall: 0.993), but GBDT's efficacy decreased because of issues with sharing private data. The ROC-AUC (0.767) and recall (0.867) in the 324 residents' health checkup data were improved by DC analysis. The study highlighted GBDT issues involving secret material by finding that LR and DC analyses may produce precise predictions with few datasets.[11]

The review found diabetic knowledge, guidelines, and medical practice deficiencies. The study described the inadequacies in diabetic ML methodology and its application to risk assessment, diagnosis, and prognosis. Through tailored risk assessment and decision assistance, ML can improve T2D therapy, especially using non-invasive variables like toenail composition, PPG signals, tongue images, and iris images for diagnosis. Risk scores and models that rely on lab test data, which may not be widely accessible and require further validation and clinical application to evaluate their usability and impact, have certain limitations.[12]

In their pregnant case-control study, they used 190 testing subjects from August 2020 and 735 training participants from August 2019 to November 2019. The study used XG Boost to identify 20 predictors from 33 variables. The prediction accuracy of the XG Boost model was 0.875, and its AUC was 0.946. The AUC and prediction accuracy of a typical LR model with four predictors were 0.752 and 0.786, respectively. DCA has shown that treating all or none of the at-risk women was not as effective as using the XG Boost model to inform treatment decisions. While both machine learning models had great calibration, the XG Boost model scored better in discrimination than the LR model.[13] which was used for predicting early-stage gestational diabetes mellitus. This study developed prediction models using three distinct algorithms and classic logistic regression. Additionally, two ensemble techniques were implemented to determine the significance of individual characteristics [8].

The Authors conducted this study using data collected from 489 patients between 2019 and 2021, ensuring they provided informed consent. This model achieved a sensitivity (95%) and a specificity (99%). It also obtained an AUC of 98%. Therefore, the clinical diagnosis system aims to save both financial resources and time by avoiding needless OGTT for those who are not at risk of GD. This will also help minimize any negative consequences [14].

The authors proposed a prospective observational study that included pregnant women aged 18–50 with gestational ages of 10–16 weeks. We excluded under-18s, twin pregnancies, known fetal abnormalities, and pre-existing edema problems. Waist measurement, hip measurement, SFT, MUAC, weight, SAT, and VAT were all predictors. We linked gestational diabetes to a high BMI, abdominal SAT, VAT, truncal SFT, waist, and gluteal hip measurements. A multivariate prediction model using a family history of diabetes, perinatal mortality, and insulin resistance discriminated GDM well (AUC of 0.860). Early identification of at-risk pregnancies in the first trimester using this approach should enable targeted interventions, suggesting its therapeutic relevance in GDM risk assessment.

The research determined that by utilizing a hybrid of machine learning, deep learning, and distributed hash table algorithms and a health governing system, engineers can improve maintenance processes and mitigate dependability issues [15, 16]. This work, compared to current screening methodologies, shows that machine learning is appealing for envisaging GDM. In order to enhance their utilization, it is crucial to emphasise the significance of quality evaluations and standardized diagnostic criteria [17]. Researchers conducted a comparative evaluation of ten machine learning classification approaches using the SMOTE technique to tackle the problem of imbalanced classes. The proposed approach has achieved outstanding outcomes. The XGBoost algorithm with SMOTE achieved 97.4% accuracy, 0.95 F1 value, and 0.87 AUC for the private dataset [18].

The authors created a CDSS model by deductive learning that is capable of explaining its decisions. This system categorizes women at risk and requires specific pregnancy interventions. It is used for maternal features and blood biomarkers. The researchers performed the necessary data preparation, followed by the artificial oversampling technique and attribute selection. Then, 5 ML models were implemented using a 5-fold sampling technique to optimize accuracy [19].

Several prediction models were compared in this study. These models included binary-class logistic regression, neural networks, and boosted decision trees. The findings indicated that the two-class boosted decision tree had greater performance compared to the others, which was attaining a 0.991 AUC. The research employed the SMOTE methodology to tackle the problem of imbalanced classes. They evaluated and compared several ML models to govern the system that achieves the maximum accuracy in predicting diabetes. The proposed approach has achieved outstanding outcomes [20].

The authors proposed a correlation study of medical and family histories to estimate GDM risk. Training data-based classification models use inference functions on illness characteristics and risk variables to predict the significance of a related factor. Experimental results suggest that the classification-based prognosis model may predict GDM early, enabling timely intervention and better management. Early detection, tailored risk assessment, integration of sophisticated technologies such as IoT and wearable sensors for early symptom recognition, and data mining and categorization are all advantages. Late diagnosis due to lack of early symptoms, complexity of risk variables, data quality and availability, and feasibility in resource-limited healthcare settings are obstacles.

This system predicted a GDM before the normal diagnostic window of 24–28 weeks. These models use clinical data, biochemical indicators, metabolites, peptides, proteins, and microRNAs. A comprehensive PubMed literature review found 109 GDM prediction ML models. Independent trials have validated only 8.3% of the models, indicating a varied predictive value. Some models have outstanding initial predictive power, especially those without independent validation, while validated models perform mixedly, highlighting the need for further refinement and validation across varied populations.

ML models may forecast early, allowing for prompt intervention, and they use several variables to improve accuracy. The challenges include the need for robust validation in different populations, the moderate predictive power of validated models, the lack of independent validation in many promising models, and the technical, logistical, and regulatory challenges of integrating these models into routine clinical practice [22].

These reviews presented an overview of the uses of ML models in predictive modelling for diabetes. It also emphasized the existing gaps in medical and technological aspects, along with the different factors involved in using ML models for decision making in diabetes [12].

The proposed approach utilizes a CNN with boosted SVM, resulting in synergistic effectiveness. The dataset they analysed comprises data from 768 patients, consisting of eight primary characteristics and a target column indicating either a "positive"/or "negative" outcome. The research was conducted using Python, and the results indicated that the deep learning model offers greater efficiency in predicting diabetes [23]. Researchers have discussed several studies utilising metabolomics and proteomics techniques to identify urine metabolites and proteins produced differently in patients with gestational diabetes mellitus. This article provided a concise summary of potential urine biomarkers that might be used to detect and diagnose gestational diabetes mellitus

The authors proposed a multi-hospital prospective observational cohort study in Nigeria collected clinical data from 253 sequentially selected pregnant women at eight to twelve weeks of gestation. The study's shortcomings include its small sample size, its design and validation in a Nigerian population, its dependence on accurate clinical data collection, and its potential for improvement with the addition of other predictor variables. All things considered, the model has the potential to improve early GDM prediction and allow for therapeutic preventive actions.

The above-related works are helpful for doing this research work.

3. Materials and Methods

In this research work, a dataset on Gestational Diabetes Mellitus (GDM) obtained from the Kaggle data repository [25] is analyzed. The dataset comprises 3525 instances with 17 attributes, each providing valuable information related to factors potentially associated with GDM.

3.1. Description of the GDM Dataset

The dataset for Gestational Diabetes Mellitus (GDM) includes 17 characteristics gathered from expectant mothers to estimate the probability of acquiring GDM.

The dataset covers several factors of a patient's medical history, present health, and lifestyle factors. It contains both continuous and categorical information**.**

3.2. Information Features of the Dataset

- Features:17
- Target variable:Outcome(GDM/Non GDM)
- Sample size:2000
- Distribution of the cases: GDM Vs Non-GDM cases

3.3. Description of the Simulation Dataset

This constructed a simulation dataset by using SVM imputation techniques to handle missing values on the features of the dataset in order to improve the resilience of our ML models. This artificial data is used to:

- Expand the sample size by adding to the current data.
- The initial dataset was unbalanced, and then the classes were balanced.
- Verify that our models
- Keeping the original datasets with 17 characteristics intact.
- Producing values for every characteristic that falls within the designated limits.
- To quarantee accurate connections between features, statistical models are utilised.
- In order to attain balance, the minority class(GDM cases) is oversampled.
- There are 229 samples in the final simulated dataset, split 50/50 between GDM and Non-GDM situations.

The table below represents the meta data of the GDM dataset. This research considers string variables from the numeric data type in the outcome of Table 1.

Figure 1 shows the following methods for predicting an optimal outcome using the below ML models.

3.4. Descriptive Characteristics for Predicting Gestational Diabetes Mellitus:

The factors that are utilized to forecast GDM fall into the following categories:

- Demographic: Case number and age
- Pregnancy History: Total Pregnancies, Previous Pregnancy Gestation, Inexplicable Prenatal Death, Large
- Offspring, or Birth Defect
- Health Metrics: Hemoglobin, HDL, OGTT, Systolic and Diastolic Blood Pressure, and BMI
- Medical History: Prediabetes, PCOS, and Family History
- Sedentary lifestyle as a lifestyle factor
- Result: GDM Condition

Table 1. Meta data of GDM

S.No	Name of the feature	Description	Data type
1	Case Number	Patient Case ID	Numeric
2	Age	Age (from 20 to 45)	Numeric
3	No of Pregnancy	{0,1,2,3,4}	Numeric
4	Gestation in Previous Pregnancy	{0,1,2}	Numeric
5	BMI	{From 13.3 to45}	Numeric
6	HDL	{15 to 70}	Numeric
7	Family History	{0=No,1=Yes}	Numeric
8	Unexplained Prenatal Loss	{0=No,1=Yes}	Numeric
9	Large Child or Birth Default	{0=No,1=Yes}	Numeric
10	PCOS	{0=No,1=Yes}	Numeric
11	Sys BP	{from 90 to 185}	Numeric
12	Dia BP	{from 60 to 124}	Numeric
13	OGTT	{from 80 to 403}	Numeric
14	Hemoglobin	{from 8.8 to 18}	Numeric
15	Sedentary Lifestyle	{0=No,1=Yes}	Numeric
16	Prediabetes	{0=No,1=Yes}	Numeric
17	Outcome (GDM /Non GDM)	{0=Non GDM,1=GDM}	String

Here a set of input features denoted as X: X={ Feature 1:'Case No', Feature 2:'Age of Patient', Feature 3: 'Pregnancy Count', Feature 4:'GDM in Previous Pregnancy', Feature 5:'BMI', Feature 6:'HDL', Feature 7:'Family History', Feature 8:' Inexplicable Prenatal Death', Feature 9:'Congenital disorder', Feature 10:'PCOS', Feature 11:' Systolic Blood Pressure', Feature 12:' Diastolic BP', Feature 13:'OGTT', Feature 14:'Hemoglobin', Feature 15:'Lifestyle Factor', Feature 16:'Prediabetes'} and the output attribute want to predict is denoted as Y: Y = { Feature 17:'Class Label (GDM / Non-GDM)'}.

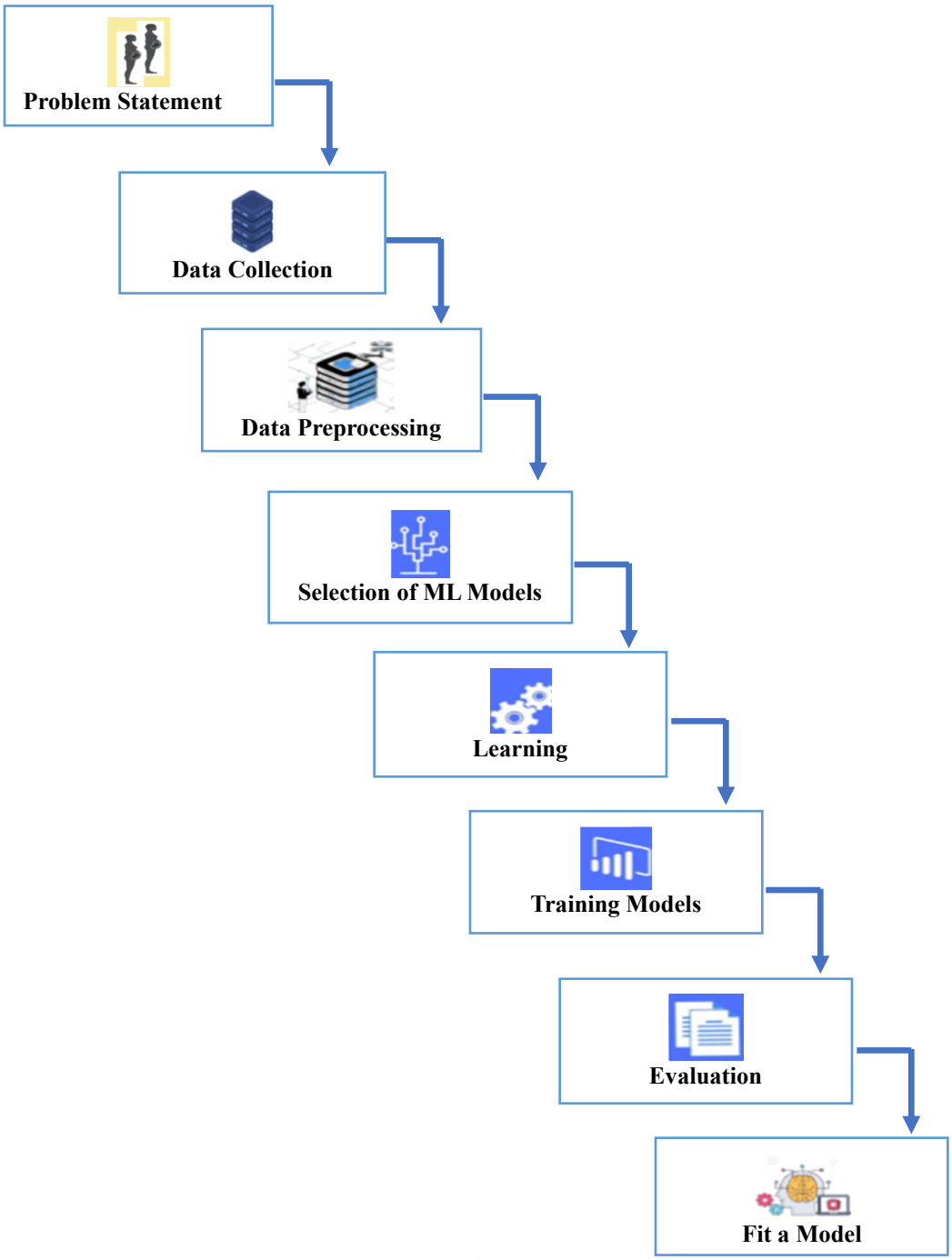

Fig. 1 Schema of the proposed system

3.5. Risk Assessment Factors

Here applied machine learning methods to a thorough risk factor evaluation:

3.5.1. Bayes Net

A Bayesian network consists of nodes (variables) and edges (dependencies). Let us denote the nodes corresponding to the features as X1, X2,..., Xn. The node representing the class label is Y. Each node has a conditional probability distribution given its parents. the probability of the class label specified all input attributes = P(Y | X1, X2, …, Xn). The probability of attributes Xi given its parents is P(Xi | Parents(Xi)). The joint probability : Prob(Xi, Xj, …, Xn, Y) = P(Y)* Π P(Xi | Parents(Xi)). This work calculates posterior to predict GDM or non-GDM, then finally computes P(Y | X1, X2, …, Xn).

3.5.2. Random Forest

Ensembling, specifically using the technique of averaging results from many trees, effectively mitigates the problem of overfitting. Moreover, this approach demonstrates excellent performance over a diverse variety of datasets, yielding high levels of accuracy. Let fun_b(x) represent the forecast of the b-th decision tree for a given input x. The expression Y_hat(x) represents the mode of the functions fun_i(x), fun_j(x), ... fun_n(x).

3.5.3. Multi-Layer Perceptron

Numerous layers of interconnected neurons make up a multilayer perceptron, a subset of Deep learning. The architecture comprises an input (1 (or) >1) and one output. Here, input and output are hidden layers. Every individual neuron computes a summation of weighted contributions, puts on an activation function, and moves the output to the subsequent layer. The forward propagation in an MLP can be expressed as follows: LT= Wx + b (linear transformation to hidden layer), h = activation(z) (apply activation function), LTh = Wh + bh (linear transformation to hidden layer), p = softmax(LTh) (softmax activation for classification), During training, adjust the weights (W) and biases (b) to minimize the prediction error. This work considers backpropagation techniques.

3.5.4. Decision Table

A decision table is a tabular representation of rules that map input conditions to output actions. It helps us make decisions based on specific combinations of input values. Each row in the table corresponds to a specific combination of input features. The columns represent input features (conditions) and output class labels (action). Let us denote the decision table as DT: DT = {(X1, X2, …, Xn, Y)}. Each entry in DT represents a specific combination of input features and the corresponding class label.

3.5.5. K-Nearest Neighbors (KNN)

The technique is a straightforward and efficient method for supervised learning (classification and regression task). It performs on the similarity between K data points. x: The new data point (input features); D: The dataset of existing data points; d(x, xi): The distance among x and each data point xi in D,K: the number of neighbors to consider. Y_hat(x) = mode (Yi) for i in K nearest neighbors (for classification), and Y_hat(x) = mean (Yi) for i in K nearest neighbors (regression). By choosing K, it affects the model's performance. The common choices include K = sqrt(n). This work considers the Euclidean distance between a novel point and all existing data points and selects the K = 1 for a better outcome.

Our machine learning models required data to be prepared through a number of important procedures. We used a unique SVM imputation technique to deal with missing values. This technique uses Support Vector Machines' ability to estimate missing data points using patterns in the available data. We used feature scaling to provide standardization for continuous variables, making sure that every feature was on a similar scale. One-hot encoding converted categorical variables, resulting in binary columns for every category. The dataset was then divided via stratified sampling into an 80% training set and a 20% test set, preserving the original class distribution of GDM and non-GDM cases. By using this method, the training and test sets are guaranteed to accurately reflect the entire dataset. Lastly, we used grid search with ten-fold cross-validation to hyperparameter tune each model in order to maximize performance. It was possible for us to develop strong and trustworthy machine learning models for GDM prediction because of this thorough data preparation procedure.

Pseudocode of the proposed system to predict GDM by SVM imputation
Input: Gestation Diabetic Data from Kaggle Dataset
Output: Fit a model for predicting Gestation Diabetic
Step 1: Let X represent the dataset containing n instances (rows) and m features (columns), X= [xij]nxm where xij is the value of feature j in instance i.
Step 2: Let Missing(xij) be a function that returns true if xij is missing
Step 3: M={(i,j) Missing(xij)=True}
Step 4: N={(i,j)Noise(xij)=True}
Step 5: X′ by excluding instances in M, N, i.e., X′=X/(M∪N) by SVM Imputation.
Step 6: Identify Step3 & Step 4 Outcome for X′
Step 7: Y=(A,B,C,D, & E) ← X' Where A=Bayes Net, B=Multi-Layer Perceptron, C= Decision Table, D= Random Forest, and E=KNN
Step 8: Apply 20 fold cross sampling technique with customizing the required parameters
Step 9: Compare (A', B',C',D', and E') Where, A'=O(A), B'= O(B), C'=O(C), D'=O(D), and E'=O(C) O=Results of evaluation metrics.
Step 10: Repeat Steps 7 to 9 until get an expected result
Step 11: Fit a model

The above Pseudocode is considered for this research work to yield a better outcome to predict GDM for the considered models. This is implemented by Python in colab and Weka 3.8.6 tool for predicting an optimal outcome by using below ML models. The above input features have missing values on BMI, HDL, OGTT and SysBp features. So, one of the powerful supervised learning algorithms is implemented for the data imputation methodology. Here, the SVM imputer class was utilised by Python-Scikit to learn how to replace missing values.

2. Procedure for Data Imputation by SVM Imputation method
Step1: Start imputation process
Step 2: Dataset denoted as D: {X= X1, X2…Xn},
Step 3: Compute the Missing dataset denoted as Mij=Xij∈X, Compute Complete dataset denoted as Cij=Xij∈X
Step 4: Target Variable T: Xj, Xj+1, Xj+2…XZ
Step 5: Set Standard scaler
Step 6: Train model
Step 7: Impute missing values to T
Step 8: Integrate from step 7
Step 9: Repeat step7 and 8 until Mij=0
Step 10: Stop

This work governs the classification evaluation metrics and regression evaluation metrics below.

$$\text{Accuracy} = \frac{(\text{TP}+\text{TN})}{(\text{TP}+\text{TN}+\text{FP}+\text{FN})} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \quad (3)$$

$$\text{False Positive Rate} = \frac{\text{FP}}{\text{FP}+\text{FN}} \quad (4)$$

$$\text{F1} - \text{Score} = \frac{2*(\text{Precission}*\text{Recall})}{(\text{Precision}+\text{Recall})} \quad (5)$$

$$\text{MCC} = \frac{(\text{TP}*\text{TN}-\text{FP}*\text{FN})}{\sqrt{(\text{TP}+\text{FP})*(\text{TP}+\text{FN})*(\text{TN}+\text{FP})*(\text{TN}+\text{FN})}} \quad (6)$$

$$\text{Kappa Statistic} = \frac{2*(\text{TP}*\text{TN}-\text{FP}*\text{FN})}{(\text{TP}+\text{FP})*(\text{TP}+\text{FN})*(\text{TN}+\text{FP})*(\text{TN}+\text{FN})} \quad (7)$$

ROC curve= It is plotted with the TPR on the y-axis and the FPR on the x-axis

PR curve= It is plotted with PPV values on the y-axis and TPR values on the x-axis

$$\text{Mean Absolute Error} = \frac{1}{\text{n}}\sum_{\text{i}=1}^{\text{n}}|\text{x}_\text{i} - \text{m(X)}| \quad (8)$$

Here, m(X)=average value of the data, n=no of data, and xi=data values

$$\text{Root Mean Square Error} = \frac{\sqrt{\sum_{\text{i}=1}^{\text{Num}}\|\text{y(i)}-\hat{\text{y}}\text{(i)}\|^2}}{1^{\text{Num}}} \quad (9)$$

Here, Num= No of data points, $y(i) = i$th measurement, and $\hat{y}(i)$ =corresponding prediction.

$$\text{Relative Absolute Error} = E_i = \frac{\sum_{j=1}^{n}|P_{ij}-T_j|}{\sum_{j=1}^{n}|T_j-\bar{T}|} \quad (10)$$

$$\text{Root Relative Square Error} = E_i = \frac{\sum_{j=1}^{n}|P_{ij}-T_j|2}{\sum_{j=1}^{n}|T_j-\bar{T}|2} \quad (11)$$

Here, TP=True Positive, TN=True Negative, FN=False Negative, FP=False Positive.

4. Results and Discussion

This section focuses on the outcome and analysis of Gestational Diabetes Mellitus (GDM Data Set). This work implemented the proposed model on Python (version 3.8) and Java platform using configurations of a 12th Gen Intel (R) Core (TM) i5-12450H at 2.00 GHz, 16.0 GB, a 64-bit operating system, and an x64-based processor. The data used had information about gestational diabetes mellitus. This work considered the Decision Table (DT), Bayes Net (BN), Multi-Layer Perceptron (MLP), Random Forest (RF), and K-Nearest Neighborhood learning algorithms to find the best machine learning model for predicting GDM.

Table 2 below presents the performance of classification metrics on the selected models. Compared to other models, the Lazy KNN produces an efficient outcome with 96.96% accuracy. The DT shows the least efficiency, which is 95.97% accuracy. The BN, MLP, and RF also lie above 96% accuracy; the KNN, BN, and RF show the same as well as the best outcome, yielding an efficient outcome with 97% precision. The DT and MLP have the same as well as the least efficiency, which is 96% of precision value; the KNN, BN, and RF show the same best outcome, which yields an efficient outcome of 97% recall. The DT and MLP have the same as well as the least efficiency, which is 96% of the recall value; the BN, MLP, DT, and RF show the same as well as the best outcome, which is 99% of ROC. The KNN shows the least outcome, which is 97% of the ROC; the BN, MLP, DT, and RF show the same as well as the best outcome, which is 99% of the PRC value. The KNN shows the least outcome, which is 96% of the PRC value; the BN has the highest value, which is 94% of the kappa value. The DT has the least as well, at 92% of kappa. The MLP, RF, and KNN show the same kappa (93%). The BN, RF, and KNN have the same high value, 97% of the F1 Score value. The MLP and DT also have the least, which is 96% of the F1 Score value. The BN and KNN have the same high value (94% of MCC value). The MLP and RF have the same 96% MCC value. The DT model shows the least value (92% of MCC).

Table 2. Classification and regression metrics

S.No	Classifier	Accuracy	Precision	Recall	ROC	PRC	Kappa	F1-Score	MCC	MAE	RRSE	RAE	RRSE	Time
1	Bayes Net (BN)	96.91%	0.97	0.97	0.99	0.99	0.94	0.97	0.94	0.03	0.17	6.45%	35.05%	0.09
2	Multi-Layer Perceptron (MLP)	96.43%	0.96	0.96	0.99	0.99	0.93	0.96	0.93	0.04	0.18	7.68%	36.45%	7.75
3	Decision Table (DT)	95.97%	0.96	0.96	0.99	0.99	0.92	0.96	0.92	0.05	0.15	11.09%	31.69%	0.5
4	Random Forest (RF)	96.82%	0.97	0.97	0.99	0.99	0.93	0.97	0.93	0.03	0.13	7.02%	26.53%	1.16
5	KNN	96.96%	0.97	0.97	0.97	0.96	0.93	0.97	0.94	0.03	0.17	6.45%	35.72%	0.01

Table 3. Existing model vs proposed model

Existing Model							
S.No	**Existing Model**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**Error**	**Time**
1	Ensemble Method	94%	94%	94.24%	94%	0.06	1.64
2	Random Forest	93%	93%	92.39%	92%	0.08	0.66
3	Logistic Regression	92%	90%	91.60%	91%	0.08	0.14
4	KNN	85%	83%	84.96%	84%	0.15	0.54
5	SVM	82%	83%	82.49%	82%	0.18	0.19
Proposed model							
S.No	**Classifier**	**Accuracy**	**Precision**	**Recall**	**F1- Score**	**Error**	**Time**
1	SVM Imputation with Bayes Net	96.91%	97%	97%	97%	0.03	0.09
2	SVM Imputation with Multi-Layer Perceptron	96.43%	96%	96%	96%	0.04	7.75
3	SVM Imputation with Decision Table	95.97%	96%	96%	96%	0.05	0.5
4	SVM Imputation with Random Forest	96.82%	97%	97%	97%	0.03	1.16
5	SVM Imputation with KNN	96.96%	97%	97%	97%	0.03	0.01

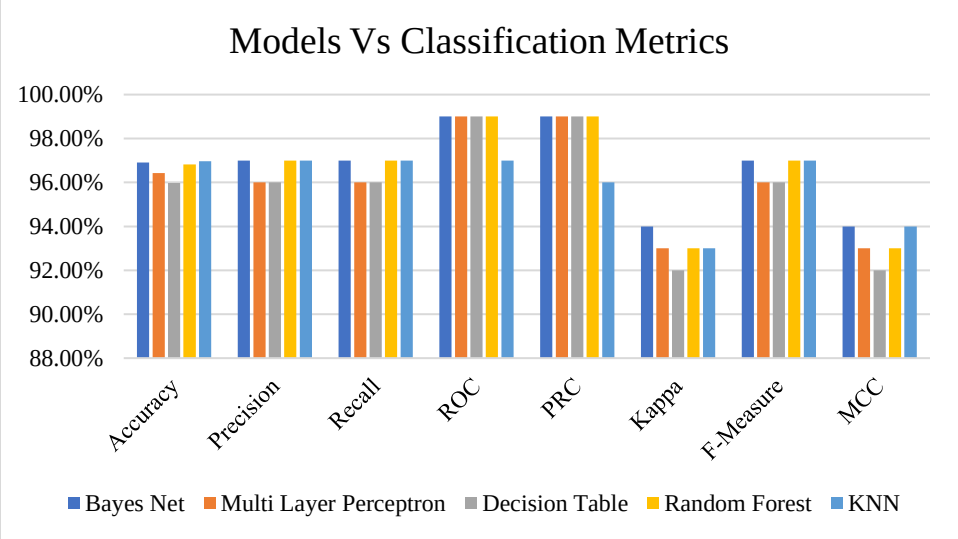

Fig. 2 Graphical representation of models vs Classification metrics

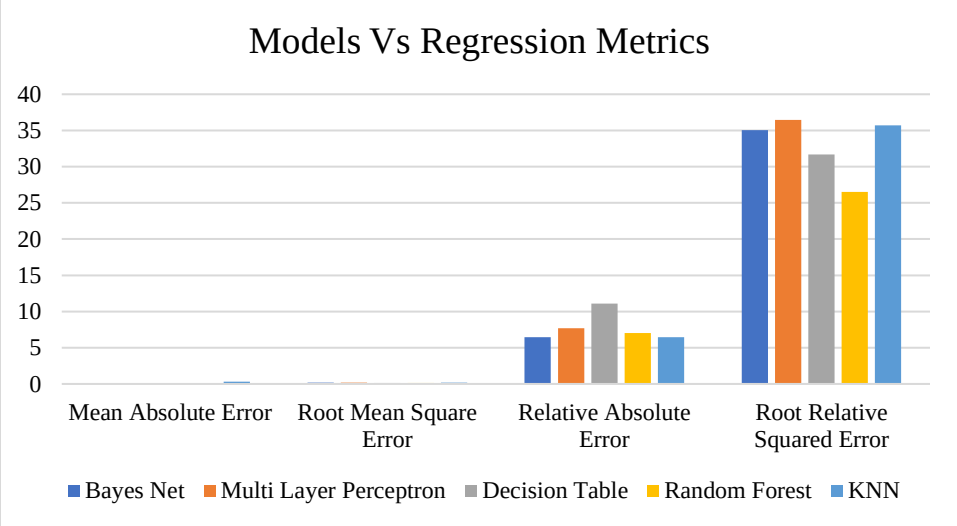

Fig. 3 Graphical representation of models vs Regression metrics

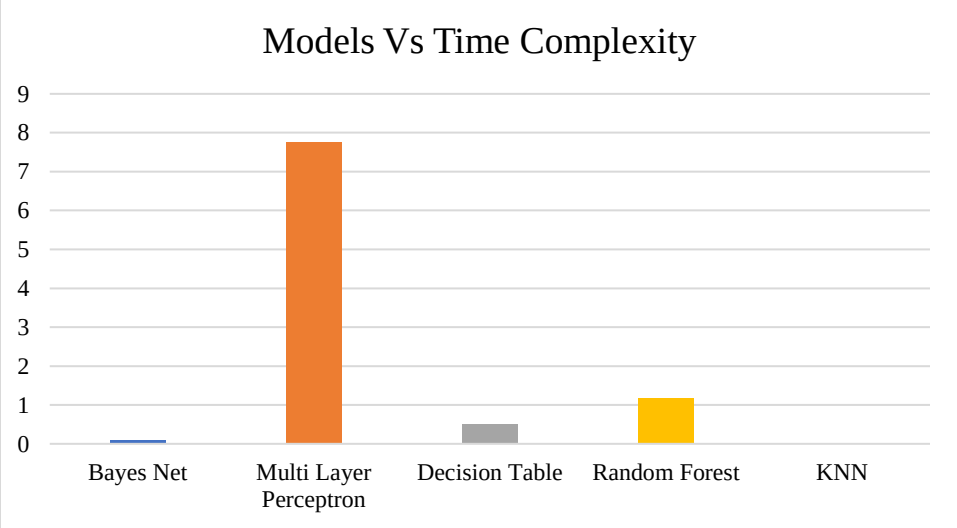

Fig. 4 Graphical representation of models vs Time complexity

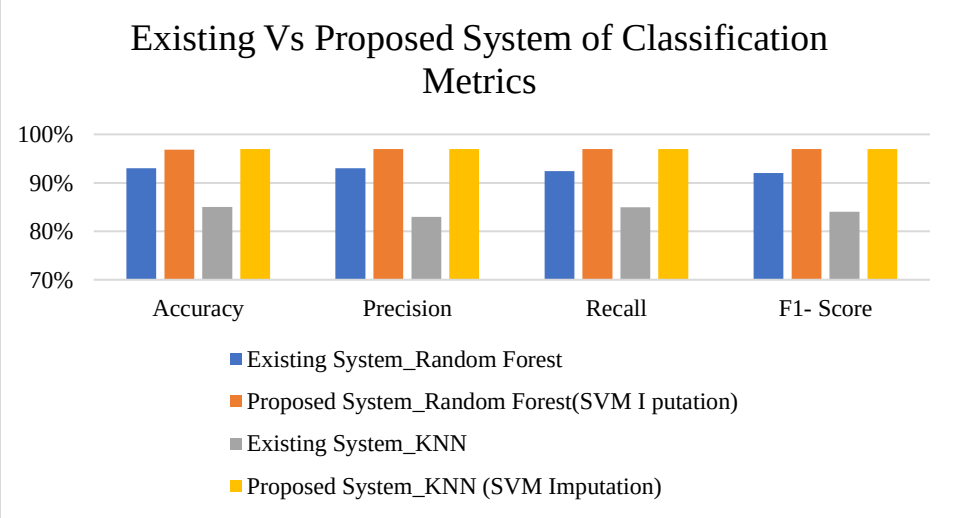

Fig. 5 Graphical representation of existing model vs Proposed model

Figure 4 shows the performance of regression metrics on the selected learning models. The figure shows that all models have the lowest deviation values. The BN, RF, and KNN have the same high performance, which is 0.03 of MAE. The MLP shows the lowest performance, which is 0.04 MAE, and DT shows a 0.05 MAE value. The BN and KNN have the same high performance, which is 0.17 of the RMSE. The MLP shows the least performance, which is 0.18 RMSE, and the DT shows a 0.15 RMSE value. The BN and KNN have the same high performance, which is 6.45% of the RMSE. The DT shows the least performance, which is 11.09 RMSE. The MLP and RF show 7.68% and 7.02 RMSE values, respectively. The RF has the best outcome, which is a 26.53% RRSE value. The other models have RRSE values above 30%. The MLP has the worst outcome compared with other models, which is a 36.45% RRSE value. The BN and KNN show below 1 second for making their models. The MLP takes more time to make its model, which is 7.75 seconds. The DT and RF models take 0.5 and 1.16 seconds, respectively.

The above table 3 shows the comparisons between the existing [2] and the proposed SVM imputation system. The existing Ensemble Method had accuracy=94%, precision= 94%, recall=94.24%, F1-Score=94%, Deviation =0.0575, and 1.64 seconds for the time complexity of its model. The Random Forest had 93% accuracy, 93% precision, 92.39% recall, 92% F1-score, 0.0760 deviation, and 0.6632 seconds for the time complexity of its model. The logistic regression had 92% accuracy, 90% precision, 91.60% recall, 91% F1-score, 0.0839 errors, and 0.1376 seconds for creating its model. The KNN had 85% accuracy, 83% precision, 84.96% recall, 84% F1-score, 0.1503 errors, and 0.5355 seconds for making its model. The SVM had 82% accuracy, 83% precision, 82.49% recall, 82% F1-score, 0.1750 errors, and 0.1864 seconds for creating its model.

The above Figure 5 shows the existing and proposed system outcomes. This proposed random forest and KNN model system produces the best outcome compared with an existing model. The proposed system of Random Forest has 96.82% accuracy, 97% precision, 97% recall, 97% F1 score, 0.03 deviations, and 1.16-time complexity, but the existing model of Random Forest had 93% accuracy, 93% precision, 92.39% recall, 92% F-measure, 0.0760 deviation, and 0.6632 seconds for time complexity. The proposed KNN model has secured an accuracy (96.96%), precision (97%), recall (97%), F1-Score (97%), deviations (0.03), and 0.01 seconds of time complexity, whereas the existing KNN model had 85% accuracy, 83% precision, 84.96% recall, 84% F1-Score, 0.1503 errors, and 0.5355 seconds for making its model. Our proposed system shows that all the models have the highest performance compared with all the existing models.

5. Conclusion

This research concludes that the KNN and Random Forest have more or less the same outcome and also perform well compared with other models. The KNN has shown efficient outcomes with the lowest deviations. This system recommends that the KNN model best predicts gestational diabetes mellitus.

Support Vector Machines (SVMs) can incur significant processing costs, particularly when dealing with extensive datasets. The duration of training substantially increases as the size of the dataset expands. Extensive training time is necessary, particularly when working with high-dimensional data or intricate models. As the number of features grows, the system's complexity increases, and it may be prone to overfitting. Effective feature selection is essential. It is susceptible to noisy data. If the dataset includes a substantial quantity of noise, the model's performance may deteriorate. It does not provide a guarantee of finding the global optimum. The resolution is contingent upon the selected kernel and hyperparameters. Researchers can attempt to address the aforementioned constraints associated with using the SVM imputation approach in the future.

References

[1] A. Sumathi, S. Meganathan, and B. Vijila Ravisankar, "An Intelligent Gestational Diabetes Diagnosis Model Using Deep Stacked Autoencoder," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3109-3126, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] A. Sumathi, and S. Meganathan, "Ensemble Classifier Technique to Predict Gestational Diabetes Mellitus (GDM)," *Computer Systems Science and Engineering*, vol. 40, no.1, pp. 313-325, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] A. Ghasemi, S. Zahediasl, and F. Azizi, "Application of Machine Learning to Predict Gestational Diabetes Mellitus Risk," *Journal of Diabetes Investigation*, vol. 9, no. 5, pp. 1105-1113, 2018.

[4] C. Laine, S.R. Bailey, and J. Ma, "A Machine Learning-Based Model for Individualized Prediction of Gestational Diabetes Mellitus Risk," *Plos One*, vol. 15, no. 7, 2020.

[5] M.T. Segura, and A. Sanchez, "Application of Machine Learning Techniques for Prediction of Gestational Diabetes Mellitus: A Systematic Review," *Medicine*, vol. 57, no. 1, 2021.

[6] Grzegorz Żabiński et al., "Multiclassifier Majority Voting Analyses in Provenance Studies on Iron Artifacts," *Journal of Archaeological Science*, vol. 113, pp. 1-15, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Gabriel Cubillos et al. "Development of Machine Learning Models to Predict Gestational Diabetes Risk in the First Half of Pregnancy," *BMC Pregnancy Childbirth*, vol. 23, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yi-Xin Li et al., "Prediction of Gestational Diabetes Mellitus at the First Trimester: Machine-Learning Algorithms," *Archives of Gynecology and Obstetrics*, vol. 309, pp. 2557-2566, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] Iswaria Gnanadass, "Prediction of Gestational Diabetes by Machine Learning Algorithms," *IEEE Potentials*, vol. 39, no. 6, pp. 32-37, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[10] Masahiro Watanabe et al., "Prediction of Gestational Diabetes Mellitus Using Machine Learning from Birth Cohort Data of the Japan Environment and Children's Study," *Scientific Reports*, vol. 13, no. 1, pp. 1-11, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Go Uchitachimoto et al., "Data Collaboration Analysis in Predicting Diabetes from a Small Amount of Health Checkup Data," *Scientific Reports*, vol. 13, no. 1, pp. 1-8, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Ashwini Tuppad, and Shantala Devi Patil, "Machine Learning for Diabetes Clinical Decision Support: A Review," *Advances in Computational Intelligence*, vol. 2, pp. 1-24, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Xiaoqi Hu et al., "Prediction Model for Gestational Diabetes Mellitus Using the XG Boost Machine Learning Algorithm," *Frontiers in Endocrinology*, vol. 14, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Burçin Kurt et al., "Prediction of Gestational Diabetes Using Deep Learning and Bayesian Optimization and Traditional Machine Learning Techniques," *Medical & Biological Engineering & Computing*, vol. 61, pp. 1649-1660, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[15] Yipeng Wang et.al, "Identify Gestational Diabetes Mellitus by Deep Learning Model from Cell-Free DNA at the Early Gestation Stage," *Briefings in Bioinformatics*, vol. 25, no. 1, pp. 1-13, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Mohammad Shahin et al., "Using Machine Learning and Deep Learning Algorithms for Downtime Minimization in Manufacturing Systems: An Early Failure Detection Diagnostic Service," *The International Journal of Advanced Manufacturing Technology*, vol. 128, pp. 3857-3883, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[17] Zheqing Zhang et al., "Machine Learning Prediction Models for Gestational Diabetes Mellitus: Meta-Analysis," *Journal of Medical Internet Research*, vol. 24, no. 3, pp. 1-15, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Hosam El-Sofany et al., "A Proposed Technique Using Machine Learning for the Prediction of Diabetes Disease through a Mobile App," *International Journal of Intelligent Systems*, vol. 2024, no. 1, pp. 1-13, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] Yuhan Du et al., "An Explainable Machine Learning-Based Clinical Decision Support System for Prediction of Gestational Diabetes Mellitus," *Scientific Reports*, vol. 12, pp. 1-14, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] Chun-Yang Chou, Ding-Yang Hsu, and Chun-Hung Chou, "Predicting the Onset of Diabetes with Machine Learning Methods," *Journal of Personalized Medicine*, vol. 13, no. 3, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[21] Sumathi Amarnath, Meganathan Selvamani, and Vijayakumar Varadarajan, "Prognosis Model for Gestational Diabetes using Machine Learning Techniques," *Sensors and Materials*, vol. 33, no. 9, pp. 3011-3025, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] Daniela Mennickent et al., "Machine Learning-Based Models for Gestational Diabetes Mellitus Prediction before 24-28 Weeks of Pregnancy: A Review," *Artificial Intelligence in Medicine*, vol. 132, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] P. Nagaraj, and P. Deepalakshmi, "Diabetes Prediction Using Enhanced SVM and Deep Neural Network Learning Techniques: An Algorithmic Approach for Early Screening of Diabetes," *International Journal of Healthcare Information Systems and Informatics*, vol. 16, no. 4, pp. 1-20, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[24] Bruno Basil et al., "A First Trimester Prediction Model and Nomogram for Gestational Diabetes Mellitus Based on Maternal Clinical Risk Factors in a Resource-Poor Setting," *BMC Pregnancy Childbirth*, vol. 24, no. 1, pp. 1-8, 346. [CrossRef] [Google Scholar] [Publisher Link]

[25] Sumathi Santhosh, Gestational Diabetes Mellitus (GDM Data Set), Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/sumathisanthosh/gestational-diabetes-mellitus-gdm-data-set