Jacob Jones, Amanda Haffner, Isabel Haas, Sarah Hurmez
Project Paper
MAT 422

Pima Indians Diabetes Dataset Prediction Models

1. Introduction

Diabetes impacts 30 million individuals in the United States, and an additional 84 million Americans are reported to have prediabetes. The disease is recognized for its potential to cause nerve damage in extremities and harm to vital organs such as the kidneys and eyes [1]. Unfortunately, the Pima Indian population has one of the highest prevalences of diabetes of any population in the world, and type 2 diabetes and obesity occur at a much younger age than in the general population [2]. Thus, it is critical to study and test for these diseases, especially in high-risk populations, so that contributing factors may be identified early and patients can minimize possible health complications.

Supervised machine learning techniques can be used for modeling and accurately predicting whether or not a patient in the dataset has diabetes based on diagnostic measures. The diagnostic measurements used were: age (in years), BMI (weight in kg/(height in m)^2), glucose levels (plasma glucose concentration at 2 hours in an oral glucose tolerance test), number of times pregnant, diastolic blood pressure, insulin levels, diabetes pedigree function (heritability), skin thickness, and outcome (whether they had diabetes: 1, or not: 0). The six supervised machine learning algorithms used are K Nearest Neighbors (KNN), Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, and Logistic Regression. The goal of this study is to determine the best algorithm to use for this dataset by comparing the values of the F1-score, precision score, and recall score from the six methods.

The dataset used is from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to predict, through diagnostic measurements, the likelihood of a patient having diabetes. Several constraints were placed on selecting these instances from a larger database. In particular, all patients listed are females at least 21 years old of Pima Indian heritage. Diabetes records from patients were obtained from two sources: an automatic electronic recording device and paper records.

The subsequent sections are organized as follows: Section 2 delves into pertinent research studies. The proposed methodologies are presented in Section 3. Section 4 outlines the experimental setups and presents the findings. Comparative analysis is addressed in Section 5, and Section 6 is dedicated to concluding remarks.

2. Related Work

This dataset has been utilized in previous research to explore potential vital signs that may indicate the genetic contribution to diabetes risk. The Pima Indians of the Gila River Indian Community have likely been the most studied group for the contributing factors and health implications of diabetes.

Previous research using this data set includes a paper by Gandhi and Prajapati (2014) that focuses on diabetes prediction using feature selection and classification. The authors evaluate the performance of the SVM classifier on the dataset by testing data mining algorithms to see their prediction accuracy in diabetes data classification [3]. Nilashi, et al. (2017) adds to the literature on the usage of machine learning techniques to improve diabetes prediction. Their method, in contrast, focuses on clustering, noise removal, and classification approaches through the use of SOM, PCA, and NN for each element, respectively [4].

Another study by Weyer et al. (1999), explores how type 2 diabetes develops over time by tracking 17 Pima Indians as they transition from normal blood sugar levels to diabetes. The findings show that early on, there's an increase in body weight, a drop in the body's ability to use insulin for glucose, and a decrease in the insulin response to glucose. As diabetes progresses, these issues worsen, accompanied by more weight gain and increased glucose production. In contrast, those who maintain normal blood sugar levels show an increase in insulin response despite some weight gain [5]. The study suggests that problems with insulin and glucose occur early in diabetes development and should be targeted for prevention.

The importance of improving prediction accuracy in diabetes risk assessment guides our study in exploring advanced deep learning methods. The investigation seeks to highlight the performance of predicting diabetes outcomes, addressing the critical need for accurate and early detection in diabetes management. The investigation will also add to the existing literature by determining the best machine-learning strategies to improve prediction accuracy.

3. Proposed Methodology

Focusing on the prediction of diseases, the present study uses k-Nearest Neighbor (k-NN), Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, and Logistic Regression methods. These methodologies are addressed in the following sections.

3.1 Dataset

The Pima Indian Diabetes dataset, provided by the National Institute of Diabetes and Digestive and Kidney Diseases, predicts diagnostically whether a patient has diabetes using specific diagnostic measurements included in the dataset. The selection of these instances from a larger database was subject to specific constraints, ensuring that all included individuals were female, at least 21 years of age, and of Pima Indian heritage. This dataset has been extensively utilized in previous research to explore potential vital signs that might indicate the presence of diabetes in patients. Comprising 768 training instances, each instance is characterized by 8 features and a class variable that provides the corresponding label (as depicted in Table 1). These features encompass essential health indicators: 1) number of times pregnant, 2) plasma glucose concentration, 3) diabetes pedigree function, 4) triceps skinfold thickness (mm), 5) diastolic blood pressure (mmHg), 6) 2-hour serum insulin (mU/ml), 7) body mass index (kg/m2), and 8) years of age. The class variable assumes a binary value of 0 or 1, where 0 indicates a healthy individual and 1 signifies a diabetic patient.

Table 1: Description of the features of the Pima Indian diabetes dataset

Variable	Feature Label	Variable Type	Range
X1	Number of times pregnant	Integer	0-17
X2	Plasma glucose concentration in a 2-hour oral glucose tolerance test	Real	0-199
X3	Diastolic blood pressure	Real	0-122
X4	Triceps skin fold thickness	Real	0-99
X5	2-hour serum insulin	Real	0-846
X6	Body mass index	Real	0-67.1
X7	Diabetes pedigree function	Real	0.078-2.43
X8	Age	Real	21-81
Y	Class	Real	0,1

3.2 Exploratory Data Analysis

We start this section off by getting a deeper understanding of the Pima Indians Diabetes dataset to see if we can get an idea of what we need to do next. We first ran a code that displayed the first 5 rows from the Pima Indians Diabetes dataset so we could get familiar with what it looked like. Next, we wanted to get the shape of the data. The result was 768 rows and 9 columns, which means there are 768 training instances with 9 different variables. Lastly, we ran a code to help us understand how the data was spread across the table. It provides each variable's count, mean, standard deviation, minimum, 25th percentile, 50th percentile, 75th percentile, and maximum. We noticed that some columns have a minimum value of zero, which doesn’t make sense medically. There are also some variables with oddly high maximum values. We can’t draw any conclusions yet, but this is enough proof that it is necessary to clean this data.

When cleaning the data, we decided to look at three things. Duplicate values, empty (NULL) values, and checking for and replacing zero values for certain variables [6]. After running the necessary code, we concluded no duplicate or empty (NULL) values. Next, we checked certain variables that had values of zero. The independent variables we checked were Glucose, Blood Pressure, Skin Thickness, Insulin, and BMI because it is medically impossible to record a value of zero for any of these. After checking for values of zero, we replaced them. One way to do this is by analyzing the histogram distribution for each independent variable. If the distribution for an independent variable is normal, we replace it with its mean value, and if the distribution for an independent variable is skewed, we replace it with its median value [7]. After evaluating the histograms for each independent variable, we concluded that glucose and blood pressure had normal distribution while skin thickness, insulin, and BMI had skewed distribution.

3.3 Supervised Machine Learning Algorithms

k-Nearest Neighbors (k-NN). The k-Nearest Neighbors (k-NN) algorithm is rooted in the fundamental notion that data points sharing similar characteristics tend to belong to the same class. The operation of k-NN begins with the selection of an appropriate distance metric, often including choices such as the Euclidean distance or the Manhattan distance. The algorithm then identifies the 'k' data points within the dataset with the shortest distances to the data point under consideration. These 'k' nearest neighbors are subsequently used in a majority voting scheme to predict the class of the data point. Incorporating k-NN into the research methodology involves careful consideration of hyperparameters, most notably the value of 'k.' The choice of 'k' directly impacts the model's performance and necessitates systematic exploration to identify the optimal configuration for the dataset. Additionally, the selection of an appropriate distance metric is a critical decision and must align with the dataset's characteristics. Feature scaling is also recommended to ensure equitable contributions of features to the distance calculations, particularly when features exhibit disparate scales.

Naive Bayes. The Naive Bayes algorithm, a probabilistic classification method, addresses the core objective of diagnostically predicting the presence of diabetes in patients. Leveraging the principles of probability and Bayesian statistics, Naive Bayes offers a powerful and straightforward approach to classification tasks. It employs Bayes' theorem to estimate the probability of a data point belonging to a particular class by considering the conditional probabilities of the features given that class. The "naive" assumption it makes is that features are conditionally independent, simplifying the calculations. By calculating the likelihood of each feature occurring in each class and combining these probabilities with the prior probabilities of each class, Naive Bayes determines the most probable class for the data point. This approach is particularly well-suited for the high-dimensional diagnostic measurements in this research, making it computationally efficient and suitable for dealing with large datasets.

Support Vector Machine (SVM). The Support Vector Machine (SVM) algorithm is a powerful supervised learning tool well-suited for classification tasks, including complex and high-dimensional datasets. SVM's operation involves identifying the optimal hyperplane that effectively separates data points into distinct classes. This is guided by the maximum margin principle, focusing on maximizing the space between the support vectors—those data points closest to the hyperplane. The choice of the optimal hyperplane relies on the positions and characteristics of these support vectors. Furthermore, SVM's versatility extends to handling both linearly separable and non-linearly separable data, thanks to its utilization of kernel functions that map data into higher-dimensional spaces, where separability becomes more evident. Margin maximization is at the core of SVM, enhancing its robustness and generalization capabilities. It also employs regularization to combat overfitting, ensuring the model can effectively generalize to unseen data. SVM brings several advantages, including its adeptness in dealing with high-dimensional data, resilience in the face of outliers, and flexibility to adapt to various kernel functions, making it an invaluable asset for diagnostic prediction in this research.

Decision Tree. The Decision Tree algorithm is a versatile and interpretable machine learning method that can effectively handle classification problems, making it well-suited for this research. The Decision Tree algorithm constructs a tree-like model where each internal node represents a decision based on a feature attribute, and each leaf node signifies a class label. The algorithm recursively partitions the data based on the features that best discriminate between classes. The decision-making process follows certain rules or criteria, such as Gini impurity or information gain, to determine the most informative features and their data-splitting thresholds. One of the distinct advantages of Decision Trees is their transparency and interpretability. Researchers and domain experts can readily understand the decision-making process by examining the tree structure and rules. This transparency can be invaluable for gaining insights into which features are most influential in diagnosing diabetes in patients. Decision Trees can handle categorical and numerical data and deal with missing values without requiring complex preprocessing. This flexibility is particularly advantageous when working with healthcare datasets like the Pima Indian Diabetes dataset.

Random Forest. The Random Forest algorithm is a robust and versatile ensemble learning method known for its prowess in classification tasks, making it a fitting choice for this study. The Random Forest algorithm assembles multiple decision trees to form a "forest." These individual decision trees are constructed using random subsets of the data and random subsets of the features, introducing diversity and mitigating overfitting concerns. Each tree in the ensemble provides a classification, and the final prediction is determined through a majority vote or weighted averaging of the individual tree outputs. Random Forest's strength lies in its ability to mitigate overfitting and enhance the model's generalization capabilities. Aggregating predictions from multiple trees produces more stable and accurate results than a single decision tree. Furthermore, it inherently provides a measure of feature importance, helping to identify the most significant diagnostic indicators of diabetes in patients. This algorithm is particularly advantageous when dealing with high-dimensional and complex datasets, where it captures intricate patterns and relationships.

Logistic Regression. The Logistic Regression algorithm is a fundamental and widely used method for binary classification tasks. Logistic Regression models the probability of an event occurring, such as the presence of diabetes, as a function of one or more independent variables. The logistic function, also known as the sigmoid function, transforms the output into a probability score between 0 and 1. This probability score signifies the likelihood of a given data point belonging to a specific class. One of the distinguishing features of Logistic Regression is its simplicity and interpretability. The model produces coefficients for each feature, offering insights into the impact of individual diagnostic measurements on the likelihood of diabetes. Researchers can readily interpret the coefficients to understand the relationship between features and the odds of a patient having diabetes. Moreover, Logistic Regression is efficient and computationally lightweight, making it well-suited for both small and large datasets. It also handles categorical and numerical data gracefully, and its predictions are readily interpretable.

4. Experiment Setups and Result Discussion

This section will present the experimental setting, experimental findings, and analysis.

4.1 Experimental Setting

This experiment was carried out on a Macbook Pro 1.4 GHz Quad-Core Intel

Core i5 processor running at 2133 MHz with the graphics display of Intel Iris Plus Graphics 645 1536 MB.

To evaluate the model's performance, we implemented several machine learning algorithms for classification, such as K Nearest Neighbors (KNN), Naive Bayes, Support Vector Machine (SVM), Decision Tree, and Random Forest, on a diabetes dataset. The code performs data preprocessing steps like handling missing values and outliers, then resumes by splitting the data into training and test sets. The evaluation metrics consist of confusion matrices, precision, recall, and F1-score. The F1-score is a useful indicator of a model's accuracy that takes into account both precision and recall. Recall quantifies the percentage of actual positives that were correctly predicted, while precision quantifies the accuracy of the positive predictions. The purpose of this study is to determine how well these algorithms predict the existence or lack of diabetes based on the data that has been provided. Each algorithm is fitted to the training data, and predictions are made on the test set. An overview of each model's performance is provided by the classification reports and confusion matrices.

The parameter setting: The K Nearest Neighbors (KNN) algorithm was tested with various hyperparameters. The number of neighbors that are taken into account, or n_neighbors, was tested with values ranging from 15 to 25. The distance metric, or p, was tested with values of 1 (Manhattan) and 2 (Euclidean). The weight of neighbors, or 'weights', was tested with both 'uniform' and 'distance'. The metric options that were investigated were 'euclidean','manhattan', and'minkowski'. The configuration that performed best was one that had n_neighbors set to 19, p as 1, and 'uniform' weights. In the case of the Naive Bayes method, the parameter was the var_smoothing parameter, which manages data smoothing. In this context, the range between 1 and 0.01 was investigated. For the Support Vector Machine (SVM) model: 'poly', 'rbf', and'sigmoid' were tested for the kernel parameter, which determines the type of hyperplane used for classification; values of 50, 10, 1.0, 0.1, and 0.01 were investigated for the regularization parameter, C;'scale' was set for the gamma parameter, which influences the kernel coefficient; and the best configuration was found with the 'rbf' kernel, C set to 1.0, and gamma set to'scale'. For the Decision Tree, the max_depth parameter was varied between values of 5, 10, 20, and 25. The min_samples_leaf parameter, which establishes the minimum number of samples needed to be a leaf node, was tested between values of 10, 20, 50, 100, and 120. The 'criterion' parameter, which determines the function to measure the quality of a split, was investigated between 'gini' and 'entropy.' The optimal setup consisted of max_depth of 5, min_samples_leaf of 10, and applying the 'gini' criterion. Finally, n_estimators, or the number of trees in the forest, was set to 1800 in the Random Forest model. The max_features parameter, which indicates the number of features to take into account when searching for the best split, did not explicitly specify any values in the analysis. The models were evaluated using performance metrics like accuracy, precision, recall, or F1-score on validation sets. The best hyperparameter configurations were chosen to achieve the best predictive performance for the dataset.

4.2 Experimental Result Analysis

This section will present a detailed analysis of the different models tested within the experiment using the dataset for detecting the risk of diabetes. Before discussing the ideal model, we will discuss the results we got from analyzing the confusion matrix of each model. Looking at **Figures 1-6,** we are presented with several confusion matrices that can be interpreted by looking at the different sectors and analyzing their percentages. The way the sectors are broken up is that the top left corner is True Positive (TP), top right is False Positive (FP), bottom left is false negative (FN) , and the bottom right is True Negative (TN). In this context, a true positive case means an accurate prediction of diabetes, which can then be further broken down into factors that can determine the best model. An ideal model has a high percentage of TP and TN cases. From these metrics, we can derive the accuracy, precision, recall, and F1- score by analyzing and performing mathematical operations on the matrix. Based solely on the matrix figures, it is evident that KNN and SVM have the highest percentages in TP and TN cases. In diabetes confusion matrix analysis, true positive (TP) and true negative (TN) cases are important because they show how accurately the model can identify real positive and negative instances. Specifically, in the context of diabetes diagnosis, TP indicates how accurately the model can identify people with diabetes among those who actually have the condition, while TN indicates how accurately the model can identify people without diabetes among those who are truly free of the disease. High TP and TN rates indicate that the model is reliable in differentiating between the presence and absence of diabetes. These metrics are important because they shed light on the model's performance and possible implications in real-world healthcare settings. Moreover, TP and TN rates are related to metrics like sensitivity, specificity, accuracy, and precision, which impact the evaluation of the model's overall effectiveness in managing and diagnosing diabetes.

To further support our findings, we will proceed by looking at **Table 2** and analyzing the values that were collected in the analysis for the strongest model. Previously the matrix displayed the strongest models to be KNN and SVM, which we will continue to analyze using the values collected. Precision is defined as the ratio of correctly predicted positive findings to all results; a recall score of one is required for the categorization process to be perfect; the best recall value in this study was obtained using the SVM model with a score of 0.68. F1-score is calculated as the weighted average of accuracy and recall scores. This criterion takes into account both FP and FN. A high F1-score indicates that the predictor has few FP and few FN. In this case, the predictor identifies serious threats while avoiding false alarms. As with accuracy, the best recall value in information retrieval experiments should be one. As with any other assessment criterion, the best F1-score obtained using SVM was 0.70. In classification analysis, an F1-score of 1 is considered perfect. Based on this analysis, the SVM has an overall stronger model which was proven in both the matrix and results comparison data.

Figure 1:
KNN Confusion Matrix Model

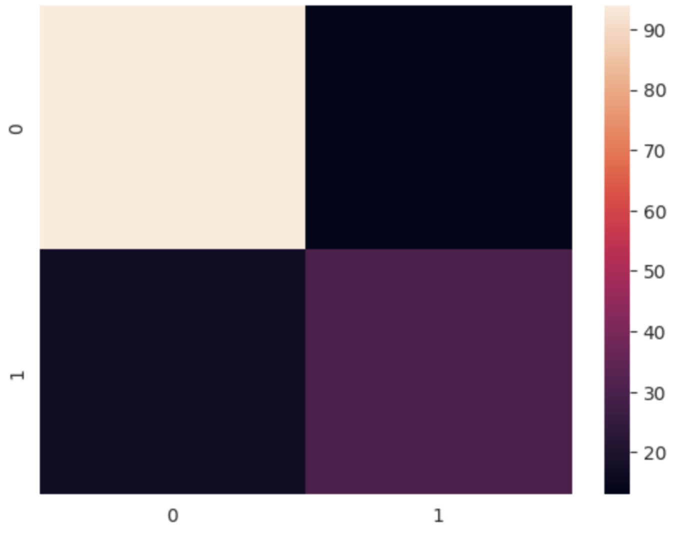

Figure 2:
Naive Bayes Confusion Matrix Model

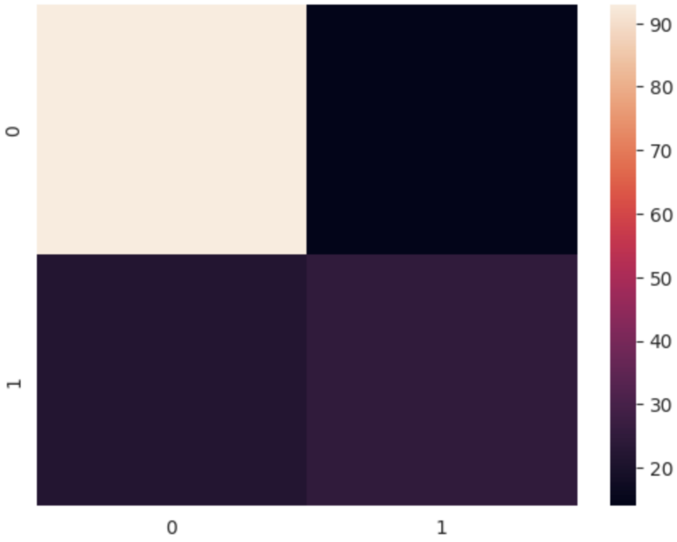

Figure 3:
SVM Confusion Matrix Model

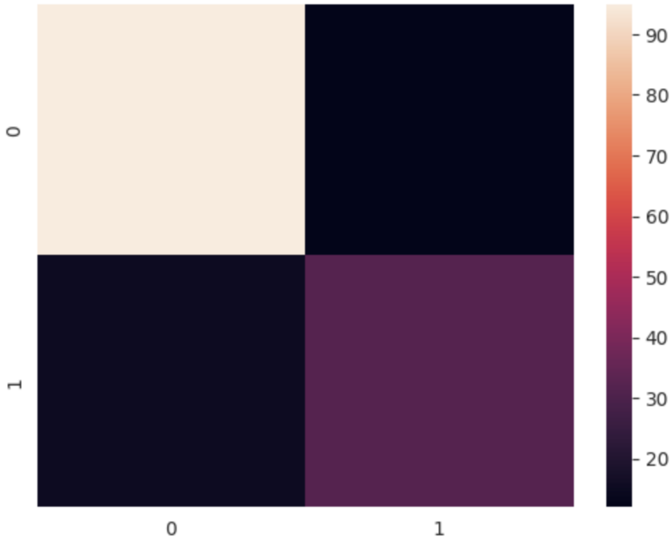

Figure 4:
Decision Tree Confusion Matrix

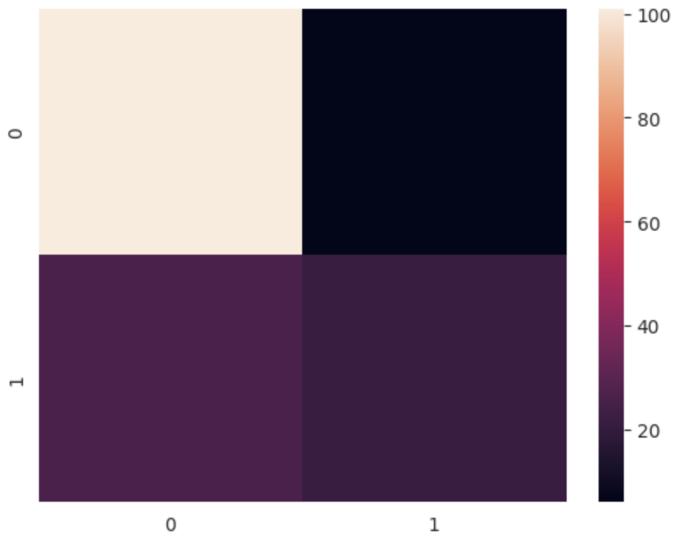

Figure 5:
Random Forest Confusion Matrix

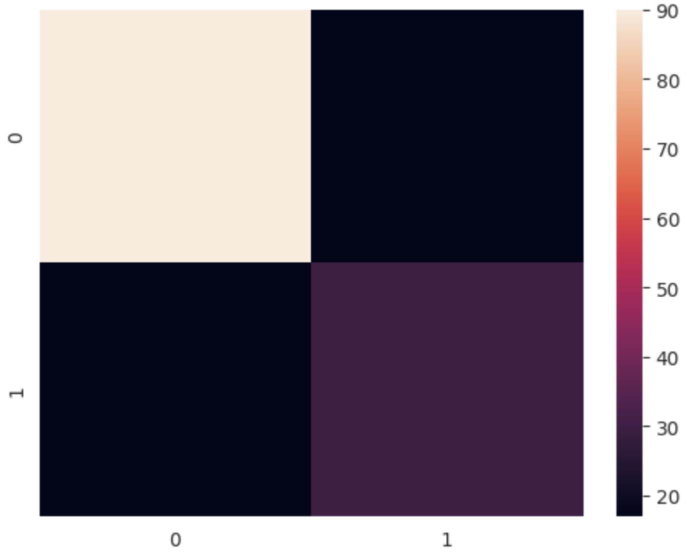

Figure 6:
Logistic Regression Confusion Matrix

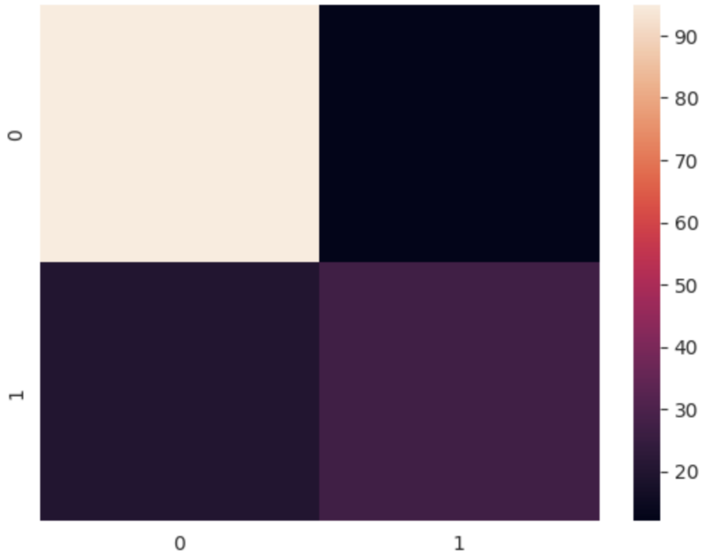

Table 2: Comparing Model Results

Model	F1 Score	Precision	Recall
KNN	0.67	0.70	0.64
Naive Bayes	0.58	0.64	0.53
SVM	0.70	0.73	0.68
Decision Tree	0.57	0.78	0.45
Random Forest	0.64	0.64	0.64
Logistic regression	0.63	0.69	0.57

5. Comparison

Table 3: Comparison Table to Recent Research

Reference	Dataset	Methods	Best Results	Accuracy	F1-Score	Precision	Recall
Sisodia & Sisodia [8]	Pima Indians Diabetes	Machine Learning	Naive Bayes	76.30%	76.0%	75.9%	76.3%
Gandhi & Prajapati [3]	Pima Indians Diabetes	Machine Learning	SVM	98.92%	—	—	—
Nilashi et al. [4]	Pima Indians Diabetes	Machine Learning	PCA-SOM-NN	92.28%	—	—	—
Our study	Pima Indians Diabetes	Machine Learning	SVM	82%	70.33%	73.0%	68.0%

In this section of the study we compare the proposed method to recent research in terms of performance measures. **Table 3** compares our study and its specific metrics to those of three other research studies that used the same dataset. The dash line (---) in the table means the researchers did not take that specific metric into consideration. One thing that our study does, unlike some of the other ones, is that we replaced the variables with values of zero (Glucose, Blood Pressure, Skin Thickness, Insulin, and BMI). Some of these other studies, [8], do not indicate if this was executed in the research/analysis so it could play part into the different outcomes. Assuming this is the case for the research done by Sisodia & Sisodia which had Naive Bayes as the best supervised machine learning algorithm with 76.30% accuracy. It is proof of the importance to replace those values of zero because we ended up with a different algorithm (SVM) and it was much more accurate with 82%. The other two research studies that we compared our results talked about how many previous research studies found SVM to be a good algorithm to start with and build on to create much more accurate results.

6. Conclusion

In this project, we wanted to determine which supervised machine learning algorithm is the best to use for modeling and predicting whether or not a patient has diabetes based on diagnostic measures. The six supervised machine learning algorithms looked at were K Nearest Neighbours (KNN), Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, and Logistic Regression. By comparing the values of the F1-score, precision score, and recall score as well as looking at the confusion matrix models, we determined that the best method to use was Support Vector Machine (SVM). SVM had the best F1 score (0.70), the second best precision score (0.73), and the best recall score (0.68). There are definitely some things that could be done to improve the success of using SVM to help with predicting whether or not a patient has diabetes, evidence of that is shown with the study by Gandhi & Prajapati [3]. Based on our findings and those from previous research, SVM is a good starting point for being the best algorithm to help predict whether or not a patient has diabetes.

Acknowledgments

The authors would like to thank anonymous reviewers for their valuable recommendations for improving the project.

Author contributions

This is a collaborative work with four authors that contribute throughout.

Ethical standard

This article does not contain any studies with human participants or animals performed by any of the authors.

Data availability

The authors declare that all data supporting the findings of this study are available on https://raw.githubusercontent.com/amandahaff/MAT422/main/diabetes.csv

References

[1] Feldman, Eva. “Disease Connection Answers May Exist within This Arizona Tribe.” *Michigan Medicine*, 23 Mar. 2020, www.michiganmedicine.org/health-lab/disease-connection-answers-may-exist-within-arizona-tribe.

[2] Ewan R. Pearson; Dissecting the Etiology of Type 2 Diabetes in the Pima Indian Population. Diabetes 1 December 2015; 64 (12): 3993–3995. https://doi.org/10.2337/dbi15-0016

[3] Gandhi, K.K., & Prajapati, N.B. (2014). Diabetes prediction using feature selection and classification.

[4] Mehrbakhsh Nilashi, Othman Ibrahim, Mohammad Dalvi, Hossein Ahmadi & Leila Shahmoradi (2017). Accuracy Improvement for Diabetes Disease Classification: A Case on a Public Medical Dataset, *Fuzzy Information and Engineering*. 9(3) (pp. 345-357). DOI: 10.1016/j.fiae.2017.09.006

[5] Weyer C, Bogardus C, Mott DM, Pratley RE. The natural history of insulin secretory dysfunction and insulin resistance in the pathogenesis of type 2 diabetes mellitus. J Clin Invest 1999;104:787–794

[6] Elgabry, Omar. “The Ultimate Guide to Data Cleaning.” *Medium*, Towards Data Science, 2 Mar. 2019, towardsdatascience.com/the-ultimate-guide-to-data-cleaning-3969843991d4.

[7] Kumar, Ajitesh. “Python - Replace Missing Values with Mean, Median & Mode.” *Analytics Yogi*, 17 Nov. 2023, vitalflux.com/pandas-impute-missing-values-mean-median-mode/.

[8] Sisodia, Deepti, and Dilip Singh Sisodia. “Prediction of Diabetes Using Classification Algorithms.” *Procedia Computer Science*, vol. 132, 2018, pp. 1578–1585, https://doi.org/10.1016/j.procs.2018.05.122.