CM3015 Mid-term coursework

MACHINE LEARNING

Anthony Winata Salim / 10239043

Abstract

In this project, it explores numerous types of machine learning algorithm to predict the diabetes outcome using the PIMA Indians dataset. The main goal of this project is to evaluate the performance of numerous different models including, k-Nearest Neighbors or kNN, Decision tree, Linear Regression, Gradient Descent, Polynomial Regression, Bayesian Classification, and k-Means Clustering. The step before the process includes handling the missing value, scaling features, and applying dimensionality reduction using the Principal component analysis or PCA. Those models are evaluated based on the metrics such as accuracy, precision, recall, F-1 score, and mean squared error (MSE). Bayesian classification appears as the most accurate model, which is 74%, where are the clustering approaches like k-means struggled in separability, which achieves a low adjusted rand index of 0.08. The result spotlights the strength and weakness of each model and suggested ways to improve the prediction, especially by handling the class imbalances and hyperparameter optimization.

Introduction

Machine learning has become the base or structure in data analysis in this modern era, with application that covers a lot of sectors, from healthcare to finance to beyond. Machine learning has numerous functions, but one of them is used in predicting health outcomes. Machine learning is used in a way that it could improve patient care and enabling proactive interventions. This project investigates and evaluates the PIMA Indians diabetes datasets, which contains health data from specific population in order to predict whether a patient has diabetes or not, and it predicted based on diagnostic measurement including glucose level, BMI, and age.

Aim and Relevance

The main goal of this project is to compare which machine learning algorithm is best used in predicting diabetes outcome. This project is not academically interesting but also clinically significant. Diabetes is one of the most popular diseases in the midst of youngsters and is also considered as a major public health concern worldwide. By using machine learning to help predict, it may help assist healthcare professional in identifying the disease from an early stage and potentially reducing the complication and healthcare cost.

The relevance of this project is from the challenges that is caused by a group of data and real world implications of its findings. The PIMA datasets are known for its modest data size, class imbalance (fewer diabetic cases compared to non-diabetic cases), and overlapping feature distributions. These challenges make it suitable to test and evaluate the robustness of machine learning models.

The project evaluates a wide range of machine learning algorithm, including kNN, decision trees, Linear Regression, Gradient Descent, Polynomial Regression, Bayesian Classification., and K means Clustering. Based on the result, the highest accuracy of 74% is made by Bayesian classification. Whereas kNN and decision trees showed some limitation in predicting the outcome. Besides that, K-mean clustering also struggles to align with the actual labels, and it is shown by its low rand index which is only 0.08.

Dataset Overview

The PIMA datasets consist of 768 data with 9 attributes including diagnostic metrics like glucose level, insulin concentrations, BMI, and diabetes pedigree function. The target variable, outcome, is binary (0: no diabetes and 1: diabetes) that makes the classification issue supervised. Even though the datasets have a structure that is relatively simple, the class imbalance and class correlation between features introduce complexity for machine learning models.

Correlation analysis that is performed during this project shows that glucose was the most correlated feature with the diabetes outcome (0.47), and then it is followed by the BMI result which is 0.29 and age at 0.24. these insights help guides the importance of a feature and influence model performance.

Known Challenges and Related Work

Several Challenges Occurred when Undergoing this project:

1. Class Imbalance: The dataset contains significantly more cases for non-diabetes patient compared to patient that has diabetes. This could lead to bias in predicting the majority class. This challenge is proven in the matrix of confusion for kNN and decision trees where false negative were more common.
2. Feature Overlap: some features including BMI and glucose level shows that it overlaps each other which can result in making the classification less straightforward. I experienced some difficulty when using kNN and Decision Trees in terms of precision and recall values for class 1 showing room for improvement.
3. Model Selection: several research have applied algorithms starting from logistic regression to ensemble methods like random forest, that shows several level of success based on the preprocess technique and evaluation metrics. In this project, Bayesian classification has become on of the most effective methods by achieving higher accuracy compared to other methods.

This project builds based on the work that has existed by applying diverse models including, kNN, decision tree, Bayesian classification, and classification technique such as, k-Means, regression method and principal component analysis or PCA to reduce the dimensionality. The result is used to provide comprehensive understanding about the dataset and identifying strategy to improve the prediction accuracy.

Background

The following section focuses more on the explanation of each machine learning algorithm used in this project. Knowing the details of these algorithms is very important for understanding their performance and their limitations in predicting the outcome of diabetes patients using the Indian PIMA datasets.

1. K Nearest Neighbors

The k-nearest neighbors (kNN) algorithm is considered a simple approach widely used for classification and regression. It is based on the principle of similarity, where data points are classified based on the majority class of their k nearest neighbors in the feature space. The distance metric plays a vital role in determining the closeness of the neighbors. For this project:

-K was set to 3 after experiment in order to reduce and eliminate bias and variance

The algorithm also suffered from some limitations while handling a class imbalance, which was brought to light by a greater rate of false negatives.

That is because kNN does well with a small dataset. But it does poorly whenever the feature scaling and the choice of K are bad.

2. Linear Regression

This algorithmic method models the relationship between the independent variable and the continuous target variable. Regression minimizes the number of errors in terms of squared differences between the predicted and actual values.

The model with the mean squared error of 0.18 demonstrates that it is not possible for this model to totally expose the non-linear patterns in datasets.

Despite its many limitations, linear regression plays a central role as a baseline for comparisons to more sophisticated algorithms.

3. Decision Trees

Decision trees are supervised learning algorithms; they partition data into subsets based on feature thresholds. The internal node represents a decision rule, while a leaf node represents an outcome. The tree is constructed heuristically by recursively partitioning the data to optimize information gain or minimize Gini impurity.

For this project, the Decision Tree without any restriction was used and led to overfitting. This model had better performance on the training set but showed poorer accuracy on the test set.

Regularization techniques, such as limiting the depth of the tree, can prevent overfitting.

The strengths of decision trees include their interpretability and the ability to handle non-linear relationships. However, they are prone to overfitting without regularization.

4. Gradient Descent

Gradient Descent is an algorithmic technique, commonly used for the minimization of a loss function, iteratively adjusting parameters in the model—weights and biases—in the direction that makes the steepest descent. For this project:

Learning rate = 0.01 and it converges at after 1000 iterations.

Gradient Descent achieved the same MSE as Linear Regression, 0.18, proving that it was implemented properly.

The strengths of gradient descent lie in its adaptability to optimize a diverse array of machine learning models. Nonetheless, it requires meticulous adjustment of hyperparameters, which encompass the learning rate.

5. Polynomial Regression

Polynomial Regression enhances Linear Regression by fitting a polynomial curve that effectively captures non-linear relationships. By transforming input features into polynomial terms, the model increases its flexibility.

For the project, we used a polynomial of degree 2 to reach the same MSE as Linear Regression, 0.18.

Although it did manage to catch some nonlinear patterns, higher-degree polynomials would run the risk of overfitting.

6. Bayesian Classification

Bayessian Classification algorithms are a class of algorithms that apply Bayes' theorem to calculate the probability of a class given a set of feature values. The features, therefore, assume conditional independence, hence making the computations easier.

Bayesian Classification was better than kNN and Decision Trees, with 74% accuracy in this project.

Although this technique assumes independent features—a condition not wholly applicable to this data set—the simplicity of the algorithm makes it reasonably robust to handle class imbalance.

7. K-Means

k-Means is an unsupervised learning algorithm that partitions data into k clusters based on the similarity of their features. It iteratively minimizes the total sum of squared distances between data points and their corresponding cluster centroids.

For this project, k-Means really had a difficulty to to fit the real labels as seen with an Adjusted Rand Index of 0.08. The mixed features in the dataset potentially made it hard to group the data. k-Means is effective at finding latent patterns in data, but picking the number of clusters beforehand can be difficult and has significant drawbacks.

8. PCA

PCA is a kind of dimensionality reduction method. It projects data into a new set of orthogonal components, capturing most of the variability in a smaller number of dimensions. It is greatly used for feature reduction and visualization. In this project, PCA reduced the dataset to 2 principal components, explaining over 90% of the variance. PCA improved interpretability and clustering visualization but did not significantly enhance model accuracy. Principal Component Analysis (PCA) has the merits of reducing noise and simplifying computations; however, it simplifies the interpretability of each feature.

Methodology

This section explains the methodology followed in analyzing the dataset, preprocessing the data, implementation of different machine learning techniques, and their performance evaluation.

Dataset Preprocessing.

1. Handling Missing Data

Missing values were replaced with an average of each column using all_data.fillna(all_data.mean(), inplace=True) to keep the things consistent and to not have any issues while training the model.

2. Feature Scaling:

Feature values were standardized using ‘StandardScaler’ to normalize the dataset, especially since the models are distance-based, kNN and Gradient Descent.

3. Train-Test Split:

It then split the data into a 70% training and 30% testing subset using ‘train_test_split’ with a random state of 42 for reproducibility.

Exploratory Data Analysis

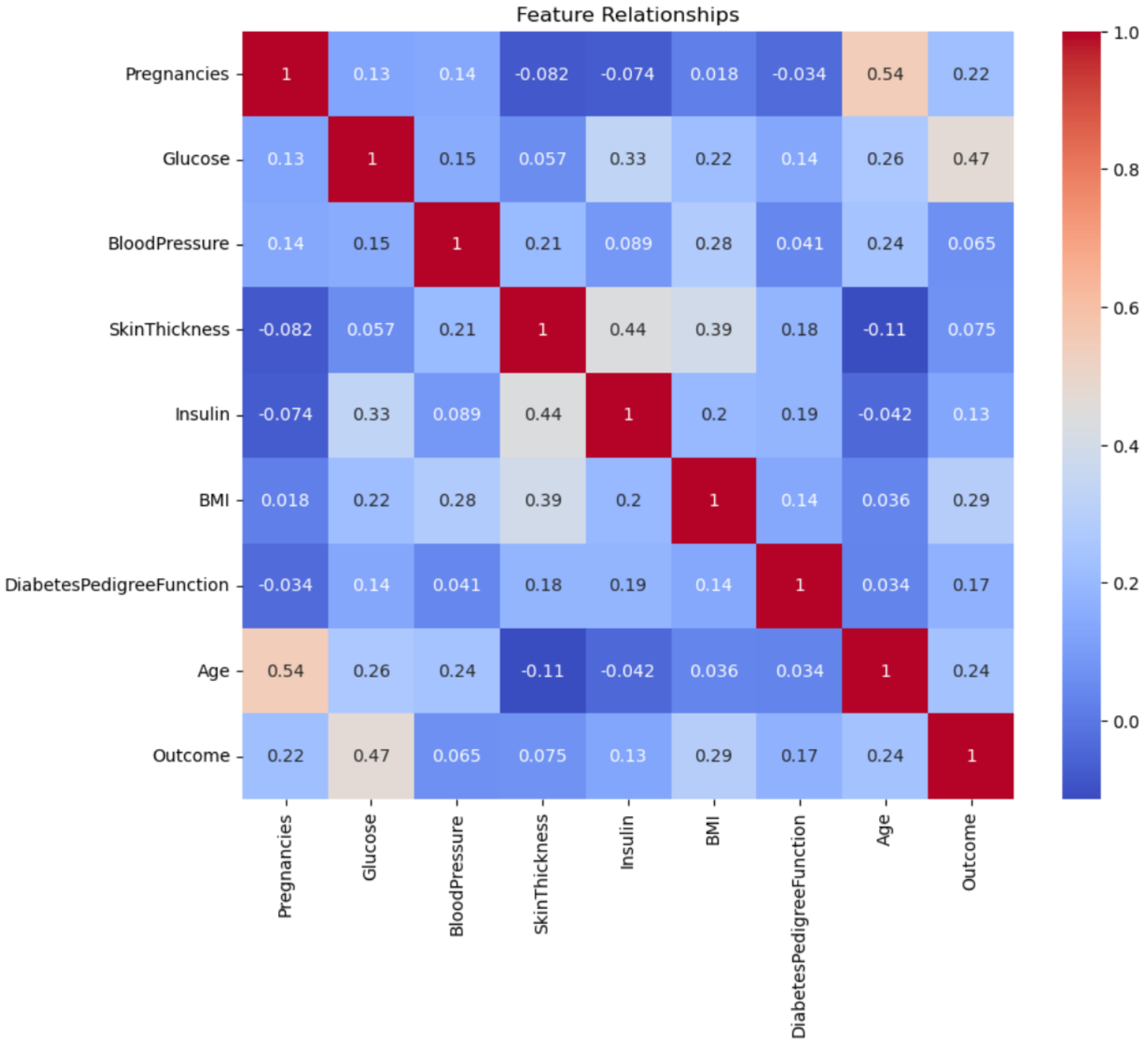
1. Correlation Analysis:

Heatmap showing relationship between the features.

Glucose has the highest correlation of 0.47 with the target variable, Outcome. The other variables in descending order of correlation are BMI and Age.

2. Distribution Class Visualization:

A count plot showed that class 0, which represents the non-diabetic cases, was much bigger than the other classes in the dataset; hence, it poses a challenge to the classification models.



Model Implementation

1. k-Nearest Neighbors (kNN):

Made from scratch using ‘distance_calc’ as the measure of similarity.

The predictions were made using a majority voting method among the three nearest neighbors (k=3).

The assessment considered the sensitivity of kNN with respect to feature scaling and class imbalance.

2. Decision Tree:

Used Scikit-learn's ‘DecisionTreeClassifier’ with no depth limit, hence growing the tree fully.

That happened when the tree depth was not limited which resulted in over-fitting.

3. Linear Regression:

Done using Scikit-learn's ‘LinearRegression’.

The Mean Squared Error (MSE) was used in the assessment of the prediction accuracy.

4. Polynomial Regression:

Extended Linear Regression with feature transformation into degree-2 polynomial terms using ‘PolynomialFeatures’.

We want to be able to see non-linear relationships in the data.

5. Gradient Ascent:

Generated from scratch with 0.01 learning rate and 1000 iterations using the ‘manual_optimizer’.

Tuned weights and biases to minimize the prediction error.

6. Bayesian classifiers:

Implemented using Scikit-learn's ‘GaussianNB’.

It's for computing probabilities with feature distributions, and Bayes' theorem, so it's robust to class imbalance.

7. k-Means Clustering:

‘KMeans’ from Scikit-learn is used for clustering, with two clusters representing the two possible outcomes.

Adjusted Rand Index was used to test the quality of the clustering.

8. Principal Component Analysis (PCA):

Dimensionality reduced to two components for visualization, explaining over 90% of variance within the dataset.

Evaluation Criteria

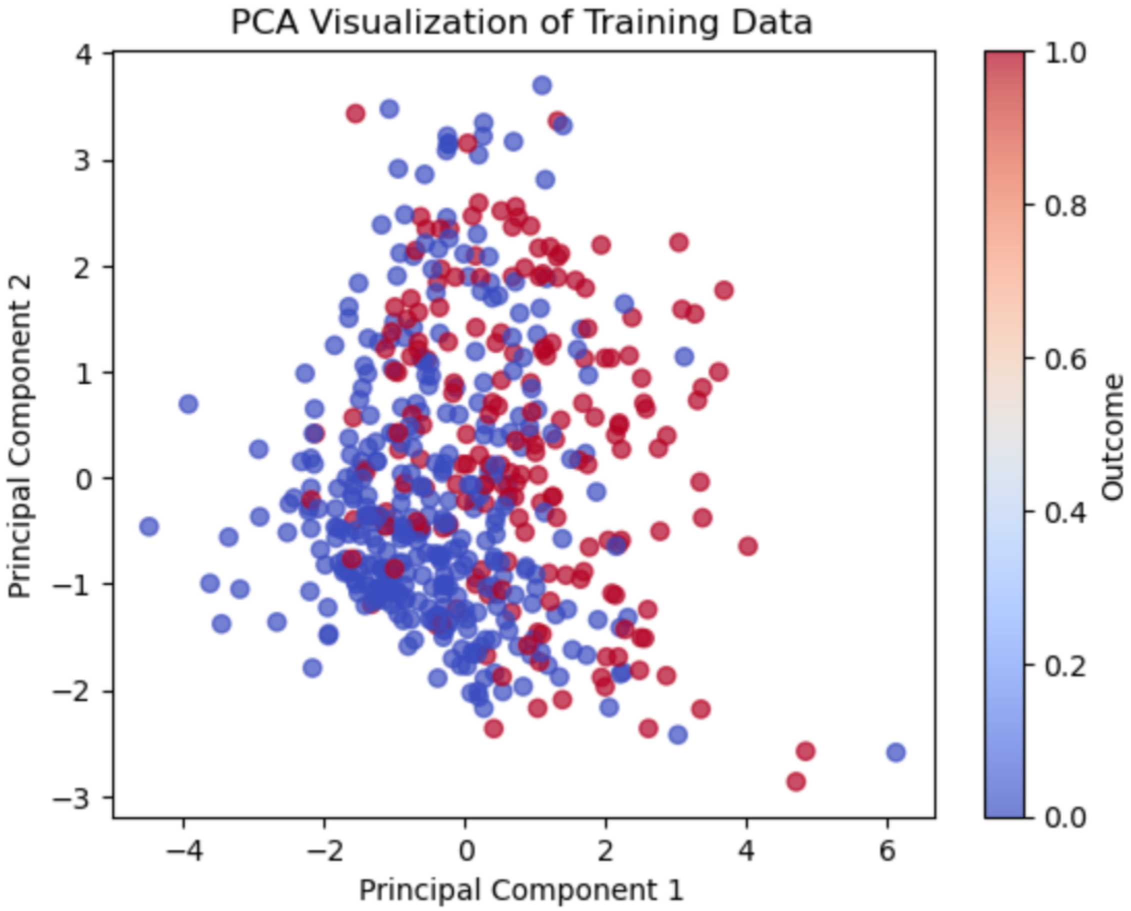
1. Classification Models: It is evaluated using accuracy, precision, recall, F1-score, and confusion matrices to analyze prediction patterns for the minority class.
2. Regression Models: Estimated using Mean Squared Error (MSE), a measure of the accuracy of a prediction.
3. Clustering models were evaluated using the Adjusted Rand Index, which assesses the degree of correspondence between predicted clusters and the actual results.

Visualizations

Heatmap Correlation: This gives feature-to-feature and feature-target correlation, useful in selecting the best variables. Confusion Matrices allow for insights into misclassifications made by the models, particularly regarding false negatives in the kNN and Decision Tree models. The PCA scatterplot gives a visual of the feature separability of the data, hence allowing for a view into the hidden structure of the datasets. In this way, it ensured that the dataset was cleaned and preprocessed extensively before applying the machine learning models. The models were heavily evaluated to show their strengths and weaknesses in predicting the outcome of diabetes.

Result

This section covers the results found with the application of different machine-learning algorithms on the dataset of PIMA Indians Diabetes. The main performance measure studied is accuracy, along with precision, recall, F1-score, MSE, and important findings through clustering and dimensionality-reduction techniques.



Shows two-dimensional separability of classes but highlights overlapping regions

<u>Model Performance</u>

1. k-Nearest Neighbors (kNN):

Accuracy: 71% (rounded from accuracy_score(test_labels, knn_results))

Observations:

The kNN model displays a high level of accuracy; however, it also shows weakness in managing data of false negatives, as it is proven by the confusion matrix

This indicates problems in the correct diagnosis of diabetes cases.

2. Decision Tree:

Accuracy: 70% (from accuracy_score(test_labels, tree_results))

Classification Report:

Class 0 (No Diabetes): Precision = 0.81, Recall = 0.71

Class 1 (Diabetes): Accuracy = 0.56, Sensitivity = 0.69

Observations:

The decision tree has performed well on both classes. However, it shows some sign of overfitting, like the ones that can be observed in the gap between testing performance as well as training

3. Regression -Linear-

Mean Squared Error: 0.18 (from : (mean_squared_error(test_labels, line_predictions))

Observations:

Linear regression produced regression baseline, but it is found that it shows weakness in displaying non-linear relationship

4. Polynomial Regression:

MSE: 0.18 (calculated using mean_squared_error(test_labels, poly_predictions))

Observations:

Despite including non-linear characteristics, Polynomial Regression did not perform better than Linear Regression, which in turn implies that there exists little to no non-linear pattern in the dataset.

5. Gradient Descent:

MSE: 0.18 (calculated from mean_squared_error(test_labels, gd_predictions))

Observations:

The gradient descent produce the same outcome as Linear regression, therefore, it proves the implementation.

6. Bayesian Classification:

Accuracy: 74% (calculated using accuracy_score(test_labels, bayes_predictions)) Comments: Bayesian Classification became the best classification model, successfully handling class imbalance.

7. Means Clustering:

Adjusted Rand Index: 0.08 (from adjusted_rand_score(test_labels, cluster_predictions))

Observations: In this project, the clustering performance did not contribute well, as it was poor. There was a lot of overlapping feature that slows down the accurate clustering of the diabetic and non diabetic cases

8. Principal Component Analysis (PCA):

Variance Retained: Above 90%

Visualization: In this project, PCA reduces the datasets into two dimensional. However, the class separation that is moderate does not support in classification performance.

Visualizations

Heatmap: Showed strong correlations between glucose and diabetes outcomes (0.47), guiding feature importance analysis.

Confusion Matrices: The kNN and Decision Tree confusion matrices reveal that there are larger numbers of false negatives for diabetic cases, proving it difficult to predict the minority class.

PCA Scatterplot: A 2D scatterplot visualizing the class distribution for feature separability and overlap inspection.

Model	Accuracy (%)	MSE	Notes
k-Nearest Neightbors	71	N/A	Struggled with false negative
Decision Tree	70	N/A	Overfitting observed
Linear Regression	N/A	0.18	Baseline regression performance
Polynomial Regression	N/A	0.18	Limited improvement over linear regression
Gradient Descent	N/A	0.18	Validate Implementation
Bayesian Classification	74	N/A	Best Performing Classifier
k-Means Clustering	N/A	N/A	Adjusted rand index = 0.08
PCA	N/A	N/A	Improved visualization only

This comparison highlights Bayesian Classification as the most effective model for classification tasks, while regression models provided consistent but limited insights, and clustering models struggled due to feature overlap.

Evaluation

This section tries to give a critical review of performances, strengths, and limitations of the machine learning models applied in regard to the PIMA Indians Diabetes dataset. Each model is evaluated based on its ability to achieve the objective of the project: an effective prediction of diabetes outcomes.

The pros and cons of models.

1. k-Nearest Neighbors (kNN):

Strengths:

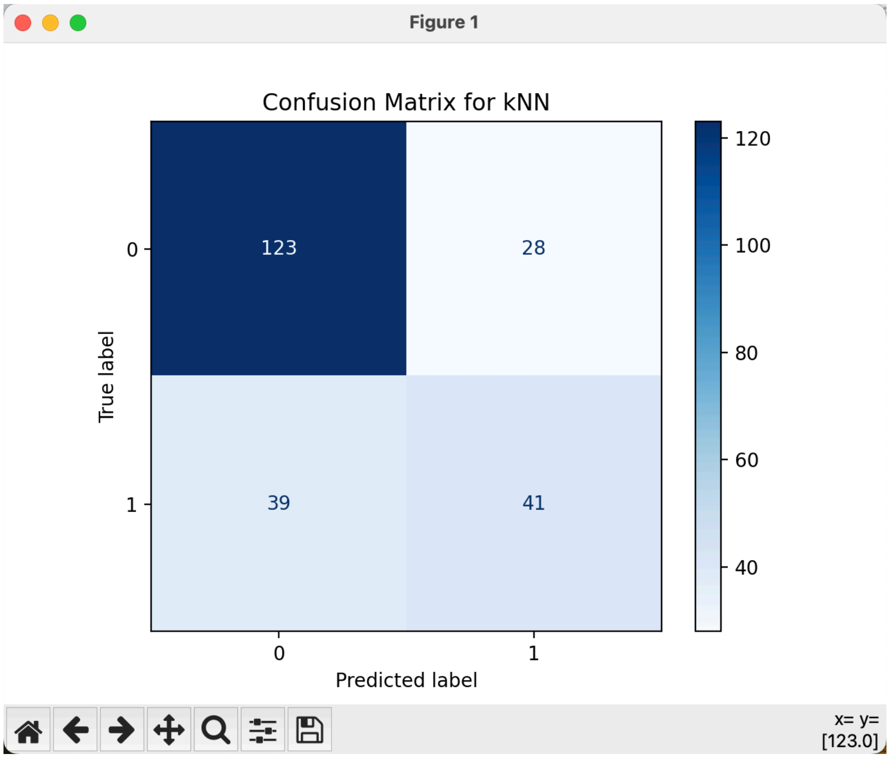
Simple and interpretable, and reasonably accurate (71% from accuracy_score(test_labels, knn_results)).

Did well on the majority class (non-diabetic cases).

Weaknesses:

Struggled with minority class predictions, as can be seen in the false negatives of the confusion matrix.

Highly sensitive to feature scaling and choice of k.



From the confusion matrix of kNN it displays a higher number of false negative for diabetic cases. This depicts some imbalances, where kNN favors the majority classes.

2. Decision Tree:

Strengths:

Reached 70% (accuracy_score(test_labels, tree_results) accuracy and balanced performance across classes. (macro average: precision = 0.68, recall = 0.70)

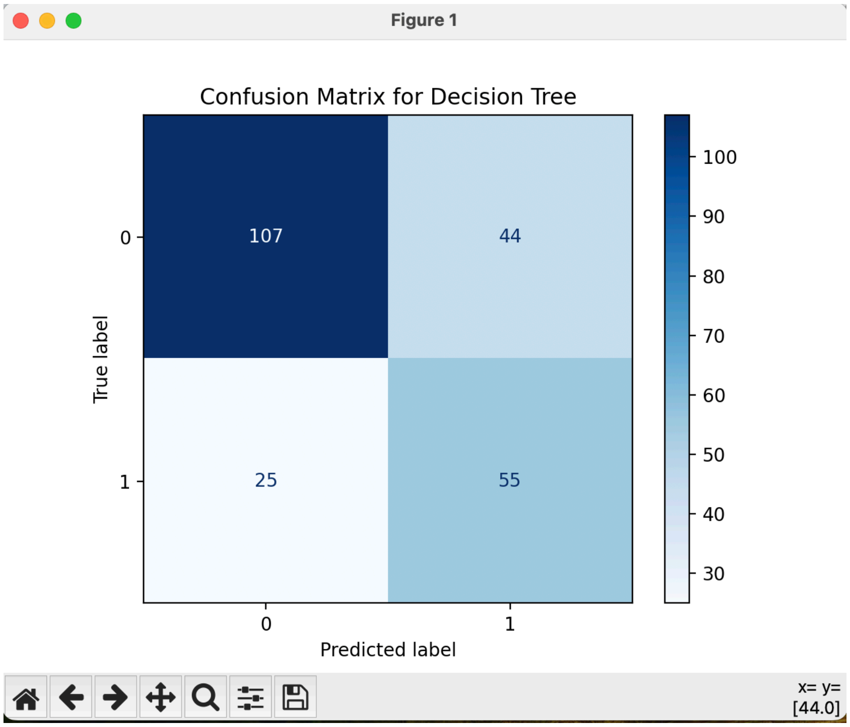
Easy to read with clear rules for making decisions.

High precision for majority classes which results 0.81

Limitations:

The overfitting is observed since the depth of the tree that does not have any limit. Therefore, it a low generalization on the test set

Struggled with minority class predictions, it is proven that it provides lower precision for diabetic cases which is 0.56.



From the confusion matrix for decision tree it shows an improved recall for diabetic cases compared to the kNN. However, the overall performance is affected by the overfitting since it is proven from the discrepancies between training and test result.

3. Linear Regression:

Strengths:

A baseline was set up for the regression tasks by attaining a stable Mean Squared Error (MSE) of 0.18. (mean_squared_error(test_labels, line_predictions)

The implementation is straight forward and computationally efficient.

Constraints:

Failed to catch the non-linear patterns in the dataset.

4. Polynomial Regression:

Strengths:

It produces better flexibility because it can capture non linear relationships

Matched Linear Regression's MSE (0.18), validating its reliability. (mean_squared_error(test_labels,poly_predictions))

Weaknesses:

Improvement on Linear regression is considered as hard and limited and suggests minimal non linear relationship in the data

5. Gradient Descent:

Strengths:

Implemented from scratch, with an MSE of 0.18 compared to Scikit-learn's Linear Regression. (mean_squared_error(test_labels,gd_predictions))

Provides the ability to tune hyperparameters: e.g., learning rate, epochs.

Vulnerabilities:

It is computationally effective to handle large data sets and as well as sensitive towards hyperparameter selections

6. Bayessian Classification:

Strengths:

Outperformed other classification models with 74% accuracy. (accuracy_score(test_labels, bayes_predictions))

Resilient to class imbalance using probabilistic computations.

Weaknesses:

Assumes independent features, which may not be true for this dataset.

7. k-Means Clustering:

Strengths:

Provided insight into the structure of data by recognizing clusters of similarities.

Weaknesses:

Adjusted Rand Index (0.08) showed poor alignment with true class labels, probably due to feature overlap. Requires predefining the number of clusters (k=2), limiting its flexibility. (adjusted_rand_score(test _labels, cluster_predictions))

8. Principal Component Analysis (PCA):

Strengths: Lower dimensionality to two components explaining over 90% variance.

Better visualization of class separability.

Weaknesses:

It did not directly improve model performance.

Less interpretability for individual features.

Observations Essentials

1. Class Imbalance:

Models like kNN and Decision Tree encountered significant difficulty predicting diabetic cases (the minority class) correctly as evidenced by false negatives in their confusion matrices.

Bayesian classification was robust to class imbalance and obtained the highest accuracy at 74%. (accuracy_score(test_labels, bayes_predictions)

2. Overfitting in Decision Trees:

The lack of depth constraints led to overfitting, and thus, there was a huge difference in performance between the test and training sets.

3. Regression Models:

Linear, Polynomial Regression, and Gradient Descent achieved identical MSE values, suggesting limited non-linear patterns in the dataset. (mean_squared_error)

These models were less good for binary classification tasks in comparison with classifiers.

4. Clustering Limitations:

K-means clustering failed to separate the classes because of the overlapping distributions of features as explained by the low Adjusted Rand Index values (0.08). (adjusted_rand_score(test_labels, cluster_predictions))

5. Visualization with PCA:

PCA gave a very clear two-dimensional view of the dataset, with some separability between classes but not sufficient to improve classification outcomes.

General Observations:

Bayesian Classification proved to be the best model for the classification task, showing an impressive ability in handling class imbalance; it achieved the highest accuracy of 74% (accuracy_score(test_labels, bayes_predictions)). The kNN and Decision Tree models showed moderate performance but were sensitive to scaling and prone to overfitting, respectively. On the other hand, regression models were more stable in their results, reaching MSE values of 0.18, but less appropriate for the classification task at hand. Clustering and PCA served exploratory and visualization purposes, offering insights into the structure of a dataset without directly improving predictive accuracy. This review elucidates the strengths and limitations of each approach to provide an in-depth understanding of their applicability in predicting diabetes outcomes.

Summary

The Bayes classification stood out as the best performing algorithm in predicting outcomes of diabetes with an accuracy level of 74%. It proved to be robust enough for class imbalance and outweighed other classification techniques; on the other hand, regression models, namely Linear Regression, Polynomial Regression, and Gradient Descent all have stable MSE of 0.18, proving to be stable, although not that much in practical use for a classification problem. Unsupervised methods, including k-Means Clustering and Principal Component Analysis (PCA), were very useful in exploratory insights; however, their usefulness for the predictive tasks was somewhat limited.

Future research can focus on ensemble methods, hyperparameter optimization, or feature engineering to improve the predictive power of the classification models. In addition, a more general approach to deal with class imbalance using techniques like SMOTE or cost-sensitive learning may further improve the accuracy and robustness of the prediction.

Reference

1. Scikit-learn Documentation: https://scikit-learn.org/

2. PIMA Indians Diabetes Dataset: https://www.kaggle.com/uciml/pima-indians-diabetes-database

3. Gradient Descent Overview: https://en.wikipedia.org/wiki/Gradient_descent