BIRZEIT UNIVERSITY

Faculty of Engineering and Technology

Electrical and Computer Engineering Department

ENCS5341

MACHINE LEARNING AND DATA SCIENCE

Course Project

Predictive Modeling for Fetal Health

Partners: Raghad Afghani **ID**:1192423.

Lojain Abdalrazaq **ID**:1190707.

Instructor's Name: Dr. Yazan Abu Farha.

Section: 2.

January 24, 2024

ABSTRACT

This project focuses on the exploration and evaluation of various machine learning models for a practical predictive task in the field of fetal health. The assignment requires a careful selection of a dataset and applying different machine learning techniques to it. A baseline model, employing nearest neighbors will be evaluated, followed by two additional models which are the Logistic Regression, and Random Forest. The performance of these models will be analyzed, emphasizing the analysis of instances where errors occur to identify any interesting pattern, and the predictive capabilities of the models in the context of fetal health classification.

INTRODUCTION

Our project involved addressing a predictive task using three distinct machine learning models: **Random Forest**, **Logistic Regression**, and **K-Nearest Neighbors (KNN)** with k=1 and k=3. In addition, it involved performance evaluation of each model was carried out using metrics such as accuracy, recall, precision, F1 score, and mean squared error. The models were trained and tested on the fetal health dataset, with the best parameters determined through hyper-parameter tuning. The objective was to compare the models' performance and select the most suitable one based on the evaluation metrics.

Firstly, the following are a brief overview about the used machine learning models:

1. ***K-Nearest Neighbors (KNN):*** The K-nearest neighbors algorithm, or KNN, is a non-parametric, supervised learning algorithm, it classifies or predicts the grouping of a data point based on its proximity to neighboring points. This method is used widely in machine learning for various classification and regression tasks. The number of the nearest neighbors to a new unknown variable that has to be predicted or classified is donated by the symbol 'K'.[1]

2. ***Random Forest Model:*** Random Forest is another powerful supervised machine learning algorithm that grows and combines multiple decision trees to create a "forest". Also, it can be used for both classification and regression problems. The logic behind this method that multiple uncorrelated models (the individual decision trees) perform much better as a group than do alone. When this model is used in a classification problem, each tree gives a classification or a "vote", and the forest chooses the classification with the majority of "votes". The effectiveness of this approach lies in the low or no correlation between individual trees, minimizing errors and improving overall predictive performance. [2]

3. ***Logistic Regression:*** Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). It is used to describe and explain the relationship between one dependent binary variable and one or more. There are three main types of logistic regression; binary logistic regression, multinomial, and ordinal logistic regression.
   - **Binary logistic regression** is used to predict the probability of a binary outcome, such as yes or no, true or false, or 0 or 1.
   - **Multinomial logistic regression** is used to predict the probability of one of three or more possible outcomes.
   - **Ordinal logistic regression** used to predict the probability of an outcome that falls into a predetermined order, such as the level of customer satisfaction, the severity of a disease, or the stage of cancer.[3]

DATASET

The dataset used in this project consists of ***2126 records*** of features extracted from Cardiotocogram (CTG) exams. This dataset used in this CTGs are a simple and affordable method for assessing fetal health, enabling healthcare professionals to take preventive measures to reduce child and maternal mortality.

This dataset features extracted from CTG exams, which were then classified by expert obstetrician into 3 classes:

- **Normal** tagged by value 1.
- **Suspect** tagged by value 2.
- **Pathological** tagged by value 3.

The dataset provided 21 features that provide valuable information regarding fetal heart rate (FHR), fetal movements, uterine contractions, and other relevant factors. By analyzing this dataset, the project aims to develop predictive models that can accurately classify the fetal health status based on the extracted features. The following figure shows the first 5 rows in the dataset:

	baseline value	accelerations	fetal_movement	uterine_contractions	light_decelerations	severe_decelerations	prolongued_decelerations	abnormal_short_term
0	120.0	0.000	0.0	0.000	0.000	0.0	0.0	
1	132.0	0.006	0.0	0.006	0.003	0.0	0.0	
2	133.0	0.003	0.0	0.008	0.003	0.0	0.0	
3	134.0	0.003	0.0	0.008	0.003	0.0	0.0	
4	132.0	0.007	0.0	0.008	0.000	0.0	0.0	

5 rows × 22 columns

Figure 1 Preview of Initial Dataset Rows

In addition, features used in the dataset, the data type of each one, and the memory usage of the set. It is noticed that all columns are of type ***float64***, suggesting that the features and target variable (output) are represented as floating-point numbers. In addition, the dataset has no null value features, in other words, no missing values. The absence of null values indicates a complete dataset, and is ready for analysis. This operation done using the **data.info()** function.

To have some statistical information, the function **data.describe()** using panda library was applied, and it produced a summary that included various statistical measures for each numerical feature. The measures are as the following:

- **Count:** represents the number of non-null values in each feature.
- **Mean:** the average value of each feature.
- **Standard Deviation (SD):** for quantifying the amount of spread of a set of values.
- **Min:** the minimum value in the feature.
- **25%(25th Percentile):** it represents the value below which 25% of the data falls.
- **50%(50th Percentile):** the median, separates the higher half from the lower half.
- **75%(75th Percentile):** the value below which 75% of the data falls.
- **Max:** represents the maximum value in each feature.

For example, the “baseline value” feature has the count equals 2126, which means that there is no null values (as mentioned before), mean =133.303857, std = 9.840844, min=106, 25% = 126, 50%=133, 75%=140, and maximum value=160 as shown in the following figure:

```
data.describe()
```

	baseline value	accelerations	fetal_movement	uterine_contractions	light_decelerations	severe_decelerations	prolongued_decelerations	abnormal_short_term_variability	mean_value
count	2126.000000	2126.000000	2126.000000	2126.000000	2126.000000	2126.000000	2126.000000	2126.000000	
mean	133.303857	0.003178	0.009481	0.004366	0.001889	0.000003	0.000159	46.990122	
std	9.840844	0.003866	0.046666	0.002946	0.002960	0.000057	0.000590	17.192814	
min	106.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	12.000000	
25%	126.000000	0.000000	0.000000	0.002000	0.000000	0.000000	0.000000	32.000000	
50%	133.000000	0.002000	0.000000	0.004000	0.000000	0.000000	0.000000	49.000000	
75%	140.000000	0.006000	0.003000	0.007000	0.003000	0.000000	0.000000	61.000000	
max	160.000000	0.019000	0.481000	0.015000	0.015000	0.001000	0.005000	87.000000	

Figure 2 The summary statistics for the dataset’s features.

Next, a visual representation of the distribution of classes within the target variable “featal_health” was provided using the Seaborn library. The variable “fetal_health” was plotted on the x-axis, representing the classes 1,2, and 3. While the height of each bar indicates the frequency of occurrences in the “fetal_health” variable.

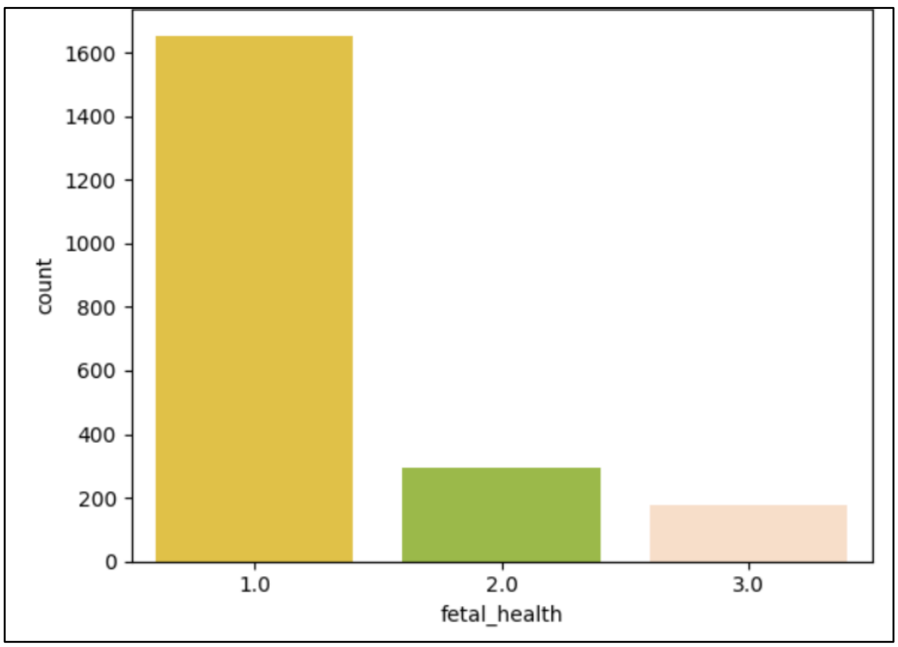

Figure 3 Distribution of the Fetal Health class.

EXPERIMENTS AND RESULTS

This section represents the details of the process of experimentation and the subsequent results generated from three predictive models; K-Nearest Neighbors (KNN), Logistic Regression, and Random Forest. The experiments extend to hyper-parameter tuning, which is considered a critical phase in refining model performance when multiple values are tested to identify the optimal configuration. Then, the matrixes such as accuracy, recall, precision, F1 score, and mean squared error serve as benchmarks for assessing model performance, ultimately guiding the selection of the most adept algorithm for the task at hand. The following sections present a detailed account of the experimentation process of each model to find the best method to suit fetal health prediction tasks.

- ***Logistic Regression:***

At first, Logistic Regression was chosen as the model for multiclass classification. ***GridSearchCV*** was then used to tune the hyper-parameters of the model. The hyper-parameters that were tuned included regularization strength, penalty type, and solver algorithm. The goal of hyper-parameter tuning was to find the optimal combination of these parameters that would maximize the model's performance. To ensure a robust evaluation, 5-fold cross-validation was employed, which involved splitting the data into five subsets and training and evaluating the model on different combinations of these subsets.

In addition, the model's performance was assessed using various metrics such as ***accuracy***, ***weighted recall***, ***weighted precision***, micro ***F1 score***, and ***mean squared error***. These metrics provided insights into different aspects of the model's classification performance and helped in determining the effectiveness of the tuned Logistic Regression model for the multiclass classification task.

The matrix shown in Figure 5 provides a comprehensive evaluation of the Logistic Regression model's performance. An accuracy of ***0.8824*** suggests that the model is performing well in classifying the test samples. The high values of recall, precision, and F1 score indicate that the model is effective in correctly identifying and distinguishing the classes. The relatively low mean squared error suggests that the predicted class probabilities are close to the true class labels.

```
********* Logistic Regression Results *********
Accuracy            : 0.8824
Recall              : 0.8824
Precision           : 0.8803
F1 Score            : 0.8824
Mean Squared Error  : 0.1505
```

Figure 4 Logistic Regression Performance Results.

- ***Random Forest:***

Then, Random Forest was chosen as the model for multiclass classification. ***GridSearchCV*** was utilized to tune the hyper-parameters of the Random Forest model. The parameters tested were (*n_estimators*) and (*max_depth*), representing the number of trees in the forest and the maximum depth of each tree, respectively. The grid search was performed using 5-fold cross-validation to evaluate different parameter combinations. The best parameter values were determined using the (*best_params*) attribute of the grid search object. Subsequently, a new Random Forest model, called (*best_RF_model*), was created with the best parameters and trained on the training data. In addition, the Random Forest model's performance was evaluated using metrics such as ***accuracy, recall, precision, F1 score***, and ***mean squared error***. These metrics provided insights into different aspects of the model's classification performance and helped assess its effectiveness for the multiclass classification task.

The results shown in Figure 8 demonstrate the strong performance of the Random Forest model. With an accuracy of 0.9420, the model achieved a high proportion of correct classifications in the test set. The recall and precision scores of ***0.9420*** and ***0.9411***, respectively, indicate the model's ability to accurately identify and distinguish between different classes, considering class imbalances. The F1 score of ***0.9420*** represents a balanced measure of precision and recall, providing an overall evaluation of the model's performance. Furthermore, the low mean squared error of ***0.0815*** suggests that the predicted class probabilities closely align with the true class labels.

```
********* Random Forest Results *********
Accuracy            : 0.9420
Recall              : 0.9420
Precision           : 0.9411
F1 Score            : 0.9420
Mean Squared Error  : 0.0815
```

Figure 5 Random Forest Performance Results.

- ***K-Nearest Neighbor:***

Finally, the K-NN model was used in training and testing the dataset. The KNN algorithm, a type of instance-based learning, was employed for classification with two different values; **K=1** and **K=3**. It calculates the distance between an input data point and its K nearest neighbors in the feature space to make the predictions. Finally, the performance of KNN with K=1 and K=3 was evaluated on the test set using various metrics which are ***Accuracy***, ***Recall***, ***Precision***, and ***F-Score***.

Now, in evaluating the K-Nearest Neighbors (KNN) model, we conducted experiments in two different values of K; K=1 and K=3. The performance matrices of each scenario are represented in the following figure. It compares the results in the key matrices, including the Accuracy, Recall, Precision, F1 Score, and the Mean Square Error, for both scenarios.

For K=1, the accuracy is 0.8981, Recall = 0.8981, Precision =0.8953, F1 Score = 0.8981, and Mean square Error = 0.1207. While K=3, the accuracy is 0.9091, Recall = 0.9091, Precision =0.9073, F1 Score = 0.9091, and Mean square Error = 0.1285.

```
********* K-Nearest Neighbors Results (k=3) *********
Accuracy (k=3)                : 0.9091
Recall (k=3)                  : 0.9091
Precision (k=3)               : 0.9073
F1 Score (k=3)                : 0.9091
Mean Squared Error (k=3)      : 0.1285
```

Figure 7 Performance results for K-NN model when K=3.

```
********* K-Nearest Neighbors Results (k=1) *********
Accuracy (k=1)                : 0.8981
Recall (k=1)                  : 0.8981
Precision (k=1)               : 0.8953
F1 Score (k=1)                : 0.8981
Mean Squared Error (k=1)      : 0.1207
```

Figure 6 Performance results for K-NN model when K=1.

ANALYSIS

At this point of identifying the most effective model for the multiclass classification task, multiple algorithms were trained and tested. Among them, the Random Forest model stands out in its performance during the evaluation process using different matrices. In this section of the report, we provide visualizations that show the correctly classified instances in green and misclassified instances in red for each feature. They provide a clear picture of how the Random Forest Model performed across different aspects of our data. The model provides high accuracy and effectiveness, making it the optimal choice for our classification task "***Fetal Health***".

From the following figures, and by comparing the Random Forest model predictions to its actual labels in the dataset, the model demonstrated high accuracy and effectiveness, making it our optimal choice for our classification task since it correctly classified the instances for most features that show a strong tendency, with the majority having values around the mode. The misclassified instances are in fewer numbers (as seen in the red bars), which is expected in a well-performing model.

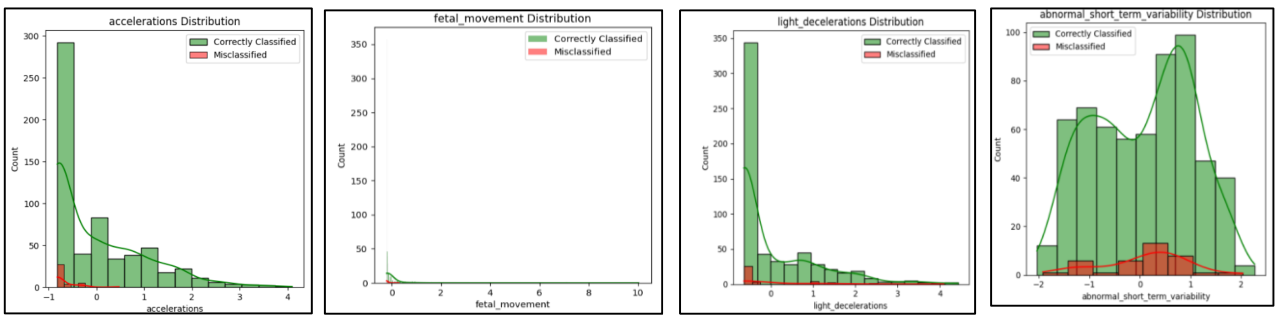

Figure 8 The Performance analysis for Random Forest Model.

In addition, the following are some interesting patterns that can be highlighted from the visuals:

- **The distribution of accelerations:** Most of the correct classifications tend to have lower values, while there are very few misclassifications. This indicates that when the acceleration value is low, it is a good predictor of correct classifications, but it doesn't play a significant role in causing misclassifications.

- **The distribution of fetal movement:** Similar to accelerations, the majority of correctly classified instances have lower values, and misclassified instances are rare.

- **Abnormal Short Term Variability Distribution**: We observe that the correctly classified instances tend to follow a normal distribution with an average value near zero. However, the misclassified instances are relatively few and scattered throughout the entire range.

- **Light Decelerations Distribution:** Most of the cases with light decelerations that are classified correctly have low values. Sometimes, there are mistakes in the classification, but they don't happen very often. This means that when the light decelerations have low values, they are usually a good indicator for predicting something.

CONCLUSION

In conclusion, multiple machine learning models were trained and tested on our dataset in order to achieve our goal in the predictive task of fetal health. Firstly, a dataset was selected, and 3 main models were used; the K-Nearest Neighbor, when K=1 and K=3, Random Forest, and Logistic Regression. Then, the performance of each model was evaluated using different evaluation matrices. In addition, it was shown and proved that the Random Forest is the most effective solution for our classification task. The performance analysis, supported by visualizations, provided valuable insights into the model's strengths and areas of success. The Random Forest model stands as a formidable choice, showcasing its superiority in handling the complexities of our dataset and making it the recommended model for future applications in similar contexts. Finally, some interesting patterns were highlighted and discussed when analyzing the performance of the Random Forest model.