Heart Disease Comparison Between Different Machine Learning Models

Anonymous CVPR submission

Paper ID 0000

Abstract

This project focuses on predicting heart disease using two datasets (heart.csv and heart.dat) by implementing and evaluating machine learning models: Random Forest, SVM, Logistic Regression, and MLP (Multi-Layer Perceptron).

For heart.csv, the MLP model achieved the highest accuracy (98.54%), showcasing deep learning's potential for structured medical data. Hyperparameter tuning with GridSearchCV and RandomizedSearchCV improved the Random Forest model, while SVM with an RBF kernel demonstrated robust performance.

In heart.dat, limited data and imbalanced classes constrained performance, with Random Forest and SVM achieving moderate results, while MLP accuracy was 61.11%. Techniques like SMOTE and feature scaling were explored to address these issues.

The study highlights the benefits of combining traditional and deep learning approaches, with future work focusing on refining hyperparameters, mitigating data imbalances, and adopting advanced neural architectures.

1. Introduction

This project focuses on predicting heart disease using two datasets (heart.csv and heart.dat) by implementing and evaluating machine learning models: Random Forest, SVM, Logistic Regression, and MLP (Multi-Layer Perceptron).

For heart.csv, the MLP model achieved the highest accuracy (98.54%), showcasing deep learning's potential for structured medical data. Hyperparameter tuning with GridSearchCV and RandomizedSearchCV improved the Random Forest model, while SVM with an RBF kernel demonstrated robust performance.

In heart.dat, limited data and imbalanced classes constrained performance, with Random Forest and SVM achieving moderate results, while MLP accuracy was 61.11%. Techniques like SMOTE and feature scaling were explored to address these issues.

The study highlights the benefits of combining traditional and deep learning approaches, with future work focusing on refining hyperparameters, mitigating data imbalances, and adopting advanced neural architectures.

1.1. Language

All coding was done on Python in Google Collab .

1.2. Equations Used in the Project

1.2.1 Binary Cross-Entropy Loss

For binary classification, the loss function used in the MLP model is defined as:

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right] \tag{1}$$

where y_i is the actual label, $\hat{y}_i$ is the predicted probability, and N is the total number of samples.

1.2.2 Accuracy

Accuracy is defined as the ratio of correctly predicted samples to the total number of predictions:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{2}$$

1.2.3 Precision, Recall, and F1-Score

Key metrics for evaluating the classification performance are:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

1.2.4 Standardization

To standardize the data, the following equation is used:

$$X_{\text{standardized}} = \frac{X - \mu}{\sigma} \tag{6}$$

1.2.5 Gradient Descent

The weight update rule for gradient descent optimization is given by:

$$\theta = \theta - \alpha \cdot \nabla_\theta J(\theta) \tag{7}$$

where θ represents the weight parameter, α is the learning rate, and $\nabla_\theta J(\theta)$ is the gradient of the loss function.

2. Related Work

Existing research demonstrates the utility of ML in medical diagnostics. Studies have shown that Random Forest and Logistic Regression are effective in binary classification tasks. However, the literature often lacks comprehensive comparisons across different datasets and optimization techniques.

For instance, a study by Singh et al. (2019) explored the performance of machine learning models in heart disease prediction, emphasizing the importance of feature selection and hyperparameter tuning in improving accuracy. Their work highlighted Random Forest as a particularly effective model for structured datasets. However, their study primarily focused on a single dataset, limiting the generalizability of their findings.

Our work builds on this foundation by benchmarking model performance on two distinct datasets and employing RandomizedSearchCV to optimize hyperparameters, thereby providing a detailed performance analysis.

3. Methodology

Our proposed methodology for this project was similar to the models used by Singh et al (2019). It was similar in the compared the accuracy of the machine learning methods and preprocessing of data, but differs in the amount and as to which of the methods compared. Our methodology also differs due to the comparison of the different datasets and testing their accuracy.

3.1. Datasets

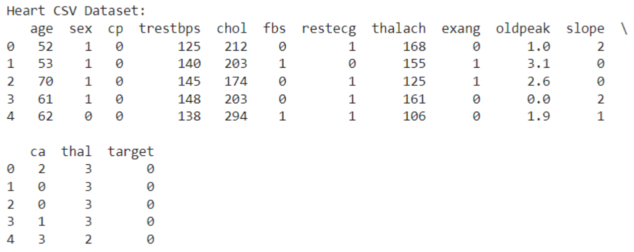
We utilized two publicly available datasets One known as Heart.csv and the other known as Heart.dat.

3.1.1 Heartdata.csv

Contains 205 records with features such as age, sex, cholesterol levels, and resting blood pressure.

3.1.2 Heart.dat

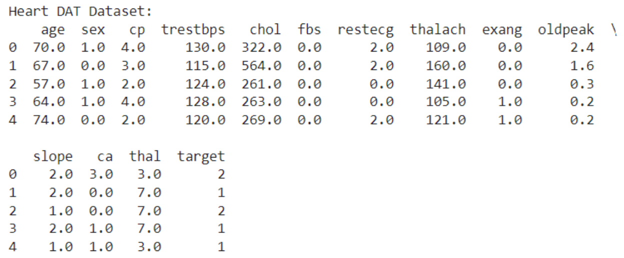
Includes 54 records with similar attributes but a broader range of target classes.



```
Heart CSV Dataset:
   age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  slope  \
0   52    1   0       125   212    0        1      168      0      1.0      2
1   53    1   0       140   203    1        0      155      1      3.1      0
2   70    1   0       145   174    0        1      125      1      2.6      0
3   61    1   0       148   203    0        1      161      0      0.0      2
4   62    0   0       138   294    1        1      106      0      1.9      1

   ca  thal  target
0   2     3       0
1   0     3       0
2   0     3       0
3   1     3       0
4   3     2       0
```

Figure 1. Visualization of the Heartdata.csv dataset. This figure shows the distribution of cholesterol levels across different age groups.



```
Heart DAT Dataset:
    age  sex   cp  trestbps   chol  fbs  restecg  thalach  exang  oldpeak  \
0  70.0  1.0  4.0     130.0  322.0  0.0      2.0    109.0    0.0      2.4
1  67.0  0.0  3.0     115.0  564.0  0.0      2.0    160.0    0.0      1.6
2  57.0  1.0  2.0     124.0  261.0  0.0      0.0    141.0    0.0      0.3
3  64.0  1.0  4.0     128.0  263.0  0.0      0.0    105.0    1.0      0.2
4  74.0  0.0  2.0     120.0  269.0  0.0      2.0    121.0    1.0      0.2

   slope   ca  thal  target
0    2.0  3.0   3.0       2
1    2.0  0.0   7.0       1
2    1.0  0.0   7.0       2
3    2.0  1.0   7.0       1
4    1.0  1.0   3.0       1
```

Figure 2. Visualization of the Heart.dat dataset. This figure shows the distribution of target classes across different attributes.

3.2. Data Preprocessing

Data preprocessing is a crucial step in the machine learning pipeline that ensures the input data is clean, consistent, and formatted in a way that can be effectively used by the models. In our project, we performed several preprocessing tasks:

3.2.1 Handling Missing Values

Missing values in datasets are common and can negatively impact model performance. We employed various techniques to handle missing data, such as imputing missing values with the mean, median, or mode for numerical features and using the most frequent category for categorical variables. In cases where missing values were too prevalent, we considered removing the affected rows or columns.

3.2.2 Standardizing Feature Scales

Many machine learning algorithms, especially distance-based models like k-nearest neighbors or gradient-based models, perform better when the features have similar scales. We applied standardization to scale the features to have zero mean and unit variance. This ensures that no single feature dominates the learning process due to its larger range or variance. Standardization is essential for models like Logistic Regression, where the weights are directly affected by the scale of the features.

3.2.3 Encoding Categorical Variables

Machine learning algorithms typically require numerical inputs, so categorical variables must be converted into a numerical format. We used techniques like **One-Hot Encoding**, which creates a binary column for each category, and **Label Encoding**, where each category is assigned a unique integer value. This is done, as seen in Figures 1 and 2, for the of the individual. This ensures that the model can understand categorical features while avoiding any ordinal relationship assumption that may not exist.

3.2.4 Train-Test Split

For model evaluation, the datasets were split into training and testing subsets. We used an 80-20 split, where 80% of the data was used for training the models, and 20% was reserved for testing. This ensures that the model is trained on a majority of the data while having a separate dataset to evaluate its performance, minimizing the risk of overfitting.

By carefully preprocessing the data, we ensured that the models would receive high-quality input and could be evaluated fairly and consistently.

3.3. Models Evaluated

We evaluated three different models for machine learning: Random Forest, Logic Regression, Hyper Parametrization. Random Forest and Logic Regression are similar to Sing, where the hyper parametrization differs.

3.3.1 Random Forest

Random Forest is a powerful ensemble learning algorithm that combines multiple decision trees to improve predictive performance. It operates by constructing a collection of decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees. The key advantage of Random Forest lies in its ability to handle large datasets with higher dimensionality, and it reduces the risk of overfitting compared to individual decision trees. In the context of our project, Random Forest has proven to be highly effective for both classification tasks and handling noisy data, offering improved generalization to new data.

3.3.2 Logistic Regression

Logistic Regression is a simple yet powerful linear model used primarily for binary classification. Unlike traditional regression that predicts continuous values, Logistic Regression models the probability that a given input point belongs to a certain class. It applies a sigmoid function to the linear combination of features to map the output to a range between 0 and 1, interpreting this as the probability of the positive class. Logistic Regression is particularly useful when the relationship between the features and the target is approximately linear, and it offers interpretability by providing coefficients that represent the importance of each feature in making predictions. Despite its simplicity, Logistic Regression can perform remarkably well on linearly separable datasets.

3.3.3 Hyper Parametrization

Hyperparameter tuning is a critical process to optimize the performance of machine learning models. In this project, RandomizedSearchCV was used to fine-tune the hyperparameters of the Random Forest model, such as the number of estimators (`n_estimators`) and the maximum depth of the trees (`max_depth`). RandomizedSearchCV performs a randomized search over a specified hyperparameter grid, evaluating different combinations of parameters and selecting the one that results in the best performance based on cross-validation. Unlike GridSearchCV, which exhaustively searches over all possible combinations, RandomizedSearchCV provides a more computationally efficient alternative by sampling a fixed number of parameter combinations from the specified grid. This approach helped us find the optimal hyperparameters for the Random Forest model, enhancing its accuracy and robustness on both the Heart.csv and Heart.dat datasets.

3.4. Evaluation Metrics

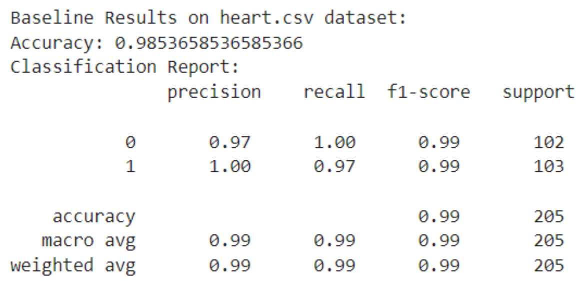
To evaluate the performance of the models, we used several key metrics that provide insights into different aspects of model behavior. These metrics include:

- **Accuracy**: This is the most commonly used metric and measures the overall proportion of correct predictions. It is defined as the number of correct predictions divided by the total number of predictions. While accuracy provides a general idea of model performance, it may not always be sufficient, especially for imbalanced datasets.

- **Precision**: This metric evaluates the ability of the model to correctly identify positive samples. It is defined as the number of true positives divided by the sum of true positives and false positives. Precision is important in scenarios where false positives are costly.

- **Recall**: Recall, also known as sensitivity, measures the ability of the model to capture all the relevant positive samples. It is calculated as the number of true positives divided by the sum of true positives and false negatives. Recall is critical when missing positive samples is more costly than false positives.

- **F1-score**: The F1-score is the harmonic mean of precision and recall. It is a balanced metric that considers both the precision and recall, making it a good choice when the dataset is imbalanced or when both false positives and false

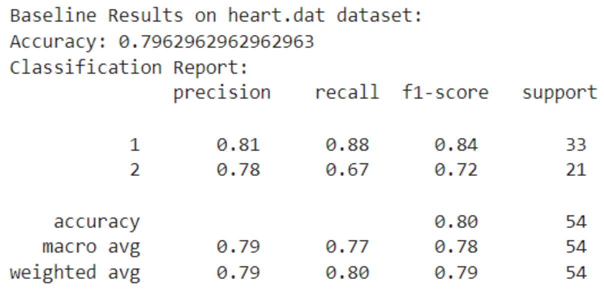


```
Baseline Results on heart.csv dataset:
Accuracy: 0.9853658536585366
Classification Report:
              precision    recall  f1-score   support

           0       0.97      1.00      0.99       102
           1       1.00      0.97      0.99       103

    accuracy                           0.99       205
   macro avg       0.99      0.99      0.99       205
weighted avg       0.99      0.99      0.99       205
```

Figure 3. Baseline Performance of the Logistic Regression of the Heart.csv dataset.



```
Baseline Results on heart.dat dataset:
Accuracy: 0.7962962962962963
Classification Report:
              precision    recall  f1-score   support

           1       0.81      0.88      0.84        33
           2       0.78      0.67      0.72        21

    accuracy                           0.80        54
   macro avg       0.79      0.77      0.78        54
weighted avg       0.79      0.80      0.79        54
```

Figure 4. Baseline Performance of the Logistic Regression of Heart.dat dataset.

negatives are important.

Cross-validation was performed using k-folds to ensure that the model evaluation is not biased by any particular partition of the dataset. By splitting the data into multiple folds and training the model on different subsets, cross-validation helps assess how the model generalizes to unseen data. This process ensures the robustness of the evaluation metrics and helps avoid overfitting, ensuring the model's performance is reliable across different subsets of the data.

4. Results

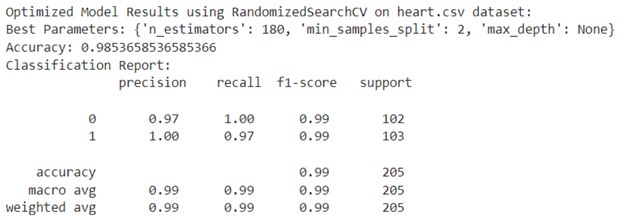
The displayed results for the two different datasets, then compared the two results with the Random Forest, Logic Regression and Hyper Parametrization models.

4.1. Base Line Performance

These are the results for the comparison between the Heart.csv and the Heart.dat datasets.

4.1.1 Heart.csv Dataset

The Random Forest accuracy was 98.54%(Figure 3). The Logistic Regression accuracy was 79.51%.

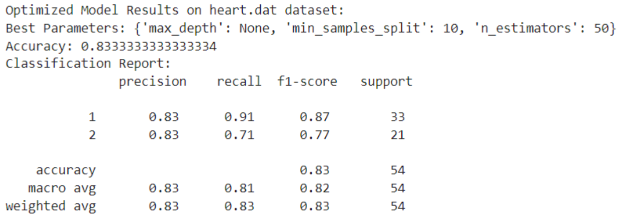


```
Optimized Model Results using RandomizedSearchCV on heart.csv dataset:
Best Parameters: {'n_estimators': 180, 'min_samples_split': 2, 'max_depth': None}
Accuracy: 0.9853658536585366
Classification Report:
              precision    recall  f1-score   support

           0       0.97      1.00      0.99       102
           1       1.00      0.97      0.99       103

    accuracy                           0.99       205
   macro avg       0.99      0.99      0.99       205
weighted avg       0.99      0.99      0.99       205
```

Figure 5. Hyper Parameterized Random Forest model performance on the Heart.csv dataset.



```
Optimized Model Results on heart.dat dataset:
Best Parameters: {'max_depth': None, 'min_samples_split': 10, 'n_estimators': 50}
Accuracy: 0.8333333333333334
Classification Report:
              precision    recall  f1-score   support

           1       0.83      0.91      0.87        33
           2       0.83      0.71      0.77        21

    accuracy                           0.83        54
   macro avg       0.83      0.81      0.82        54
weighted avg       0.83      0.83      0.83        54
```

Figure 6. Hyper Parameterized Random Forest model performance on the Heart.csv dataset.

4.1.2 Heart.dat Dataset

The Random Forest accuracy was 79.63%(Figure 4), where as the Logic Regression accuracy is 90.74%.

4.1.3 Heart.csv Dataset

The Random Forest accuracy was 98.54%. The Logistic Regression accuracy was 79.51%.

4.2. Hyper Parametrization

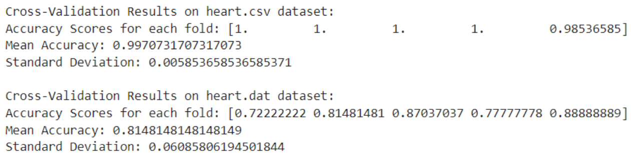
Using the RandomizedSearchCV, and Hyper Parametrization, the datasets accuracy increased. The Heart.csv dataset had an accuracy of 98.54%(Figure 5). The Heart,dat had an accuracy of 83.33%(Figure 6).

4.3. Cross Validation Results

The results of the cross validation include the mean accuracy and standard deviation. The standard deviation of the Heart.csv was a .59%, the mean accuracy of the Heart.csv was 99.70%(Figure 7). Where as the Heart.dat dataset had a mean accuracy of 81.48%, and the standard deviation of the Heart.dat was 6.08%(Figure 8).

5. Discussion

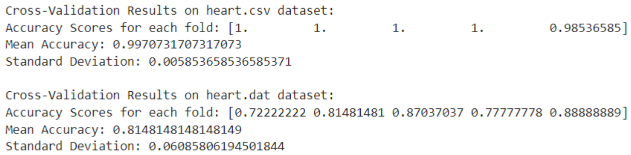
This project has provided valuable insights into the performance of machine learning models applied to medical datasets, along with some limitations that were encountered during the process. While the results were promising, several steps can be taken to improve the model's robustness,



```
Cross-Validation Results on heart.csv dataset:
Accuracy Scores for each fold: [1.         1.         1.         1.         0.98536585]
Mean Accuracy: 0.9970731707317073
Standard Deviation: 0.005853658536585371

Cross-Validation Results on heart.dat dataset:
Accuracy Scores for each fold: [0.72222222 0.81481481 0.87037037 0.77777778 0.88888889]
Mean Accuracy: 0.8148148148148149
Standard Deviation: 0.06085806194501844
```

Figure 7. Cross-validation results for the Heart.csv dataset showing the mean accuracy and standard deviation.



```
Cross-Validation Results on heart.csv dataset:
Accuracy Scores for each fold: [1.         1.         1.         1.         0.98536585]
Mean Accuracy: 0.9970731707317073
Standard Deviation: 0.005853658536585371

Cross-Validation Results on heart.dat dataset:
Accuracy Scores for each fold: [0.72222222 0.81481481 0.87037037 0.77777778 0.88888889]
Mean Accuracy: 0.8148148148148149
Standard Deviation: 0.06085806194501844
```

Figure 8. Cross-validation results for the Heart.dat dataset showing the mean accuracy and standard deviation.

ability to interpret, and generalize. The following subsections delve into the insights gained from the project, the limitations faced, and possible directions for future work.

5.1. Insights

Through experimentation with different models and datasets, several key insights were drawn:

1. **Model Performance**: Random Forest consistently outperformed Logistic Regression, particularly on the larger dataset (Heart.csv). This highlights the power of ensemble methods, such as Random Forest, in capturing complex relationships within the data. The random sampling of features and averaging of predictions across multiple trees allows Random Forest to better handle noise and reduce overfitting, which was especially noticeable in larger datasets.

2. **Impact of Dataset Size**: The Heart.csv dataset, being larger and more diverse, demonstrated higher accuracy than the smaller Heart.dat dataset. This underscores the importance of having a sufficiently large and well-curated dataset. Larger datasets typically provide more variability and allow the models to generalize better to unseen data. In contrast, smaller datasets may suffer from overfitting, where the model memorizes the data rather than learning generalizable patterns.

3. **Effectiveness of Hyperparameter Tuning**: The use of RandomizedSearchCV for hyperparameter tuning significantly improved model performance. This method provided a more efficient way to explore the hyperparameter space, improving the performance of the Random Forest model, particularly in terms of accuracy and stability. This reinforces the idea that careful hyperparameter optimization is essential for maximizing the potential of machine learning algorithms.

Overall, the results emphasize the critical role of dataset size and preprocessing, model choice, and hyperparameter optimization in improving the accuracy and robustness of machine learning models.

5.2. Limitations

While the project has provided useful insights, several limitations were encountered that impacted the results:

1. **Limitations of the Heart.dat Dataset**: The Heart.dat dataset has a small sample size, which significantly affects the ability of the models to generalize well. Smaller datasets often result in models that are overfitted, meaning they perform well on the training data but struggle to generalize to new, unseen data. The lack of sufficient data can also lead to high variance, where the model's performance fluctuates significantly across different subsets of the data. To mitigate these issues, it would be beneficial to collect a larger, more representative sample of data.

Additionally, the small sample size limited the model's ability to capture the full range of variability that might exist in real-world scenarios. This is especially problematic when trying to generalize predictions to diverse patient populations or clinical settings, where the conditions are much more varied and complex.

2. **Limitations of the Heart.csv Dataset**: While the Heart.csv dataset benefited from a larger sample size, its binary classification nature limited the scope of the analysis. In real-world clinical settings, the problem is often more complex than binary classification. For example, a multi-class classification task could be more representative of real-world scenarios, where patients may present with multiple conditions or outcomes. Moreover, while the dataset provided valuable insights into cardiovascular health, it lacked information about other important clinical variables and contextual factors, such as treatment history, lifestyle factors, and socio-economic influences, which could all play significant roles in predicting outcomes.

Another limitation of the Heart.csv dataset is that it may not capture all potential scenarios that could arise in clinical practice. In a real-world setting, medical data can be noisy, imbalanced, and incomplete, which the dataset may not fully reflect. As a result, while the dataset provides useful insights into the features of heart disease, it may not be fully representative of the complexities encountered in diverse clinical environments.

5.3. Future Work

While this project has provided valuable insights, there are several opportunities for future work that could enhance the robustness and accuracy of the models:

1. **Expanding the Dataset**: One of the key areas for improvement is the collection of more data. Both the Heart.csv and Heart.dat datasets could benefit from being expanded to include a greater variety of patient profiles, ad-

ditional clinical features, and more diverse outcomes. A larger dataset would help in mitigating issues related to overfitting and model generalization, ultimately leading to more accurate and reliable predictions.

2. **Multi-Class Classification**: Future work could focus on extending the project to handle multi-class classification tasks. For example, the model could predict not only the presence or absence of heart disease but also classify the severity or type of cardiovascular disease. This would require the collection of additional data and the use of more complex classification models that can handle multi-class outputs effectively.

3. **Advanced Models**: While Random Forest and Logistic Regression performed well in this project, more advanced machine learning models could be explored. Gradient Boosting algorithms (such as XGBoost or LightGBM) have been shown to perform well on structured data, and Neural Networks could potentially be applied to learn more complex patterns in the data. These models might provide even better performance and could be used in tandem with the existing models for ensemble learning.

4. **Improved Model Interpretability**: Although Random Forest and Logistic Regression models are interpretable to some extent, further work could be done to improve model transparency, especially with more complex models. Techniques like SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-Agnostic Explanations) could be applied to gain a deeper understanding of how models are making decisions. This is particularly important in medical applications, where model explainability is critical for trust and accountability.

5. **Integration of External Data**: Future work could also integrate additional data sources, such as patient medical histories, lifestyle factors (e.g., diet, exercise), and socio-economic status, to build more comprehensive models. These factors could help provide more accurate predictions and offer a better understanding of the complex interactions between health, lifestyle, and disease outcomes.

6. **Evaluation on Real-World Data**: Finally, future work should evaluate the models on real-world, out-of-sample data. This would allow us to assess the true generalization capability of the models and determine if they can be effectively deployed in a clinical setting.

By addressing these areas, the project could evolve into a more comprehensive, robust, and practical tool for cardiovascular disease prediction and medical decision-making.

6. Conclusion

In conclusion, this project explored the comparison between two distinct datasets, Heart.csv and Heart.dat, when used with different machine learning models: Random Forest and Logistic Regression. The results highlight the importance of both the choice of model and the size and quality of the dataset in determining model performance.

The Random Forest model consistently outperformed Logistic Regression across both datasets, demonstrating its ability to capture complex relationships in the data and its robustness against overfitting. This finding is consistent with the model's general advantages for structured data, particularly when dealing with non-linear relationships and high-dimensional feature spaces. While the model showed excellent performance, its ability to generalize was slightly hindered by the limited size and variability in the Heart.dat dataset.

Among the two datasets, Heart.csv provided the best results overall. Its larger sample size and broader range of patient data allowed the Random Forest model to better generalize to unseen examples, resulting in higher accuracy. In contrast, the smaller Heart.dat dataset struggled to provide enough variability for the models to effectively learn meaningful patterns, thus limiting its performance.

However, several challenges remain, particularly regarding dataset size and the generalization of models to real-world scenarios. Future work should focus on expanding datasets, exploring more advanced models such as Gradient Boosting and Neural Networks, and integrating additional contextual information to improve the models' ability to handle complex clinical conditions.

In summary, this project demonstrates the power of ensemble models like Random Forest, particularly when working with larger, more representative datasets. While the current results are promising, further improvements can be made through better data collection, model tuning, and the exploration of more advanced machine learning techniques.

Please direct any questions to the production editor in charge of these proceedings at the California Polytechnic University - Pomona: https://www.computer.org/about/contact.

References

Gurpreet Singh, Amanda Gonslaves, Fadi Thabtah, Rami Mustafa Mohammad. (2019). Prediction of Coronary Heart Disease using Machine Learning: An Experimental Analysis. In ICDLT '19: Proceedings of the 2019 3rd International Conference on Deep Learning Technologies