Cardiac Risk Prediction using Classical Machine Learning Techniques

Martinus Boom
University of Coimbra, Coimbra, Portugal
uc2024154464@student.uc.pt

Abstract

Using clinical data, this project compares machine learning models for cardiac risk prediction with and without SMOTE data balance. Sensitivity was the primary emphasis of the investigation in order to accurately identify high-risk patients, and balancing the data resulted in slightly lower specificity but increased sensitivity. Wilcoxon-Holm and Friedman statistical tests verified that the models differed significantly, and it was concluded that the optimum balance for clinical usage was offered by logistic regression, while the neural network achieved the highest sensitivity.

1 Introduction

Heart disease detection at an early stage proves essential because this condition continues to be a major cause of death across the globe. Machine learning systems detect high-risk patients through their ability to recognize clinical data patterns which appear before health incidents.

The research project studies multiple supervised learning algorithms which will be used to analyze structured clinical data for heart disease risk prediction. The dataset was preprocessed to handle missing values, balance class distribution, and ensure feature consistency before the models training, which included Decision Tree, Logistic Regression, Neural Network, K Nearest Neighbors and Naïve Bayes, that also went through tuning and validation on a stratified cross-validation, with sensitivity defined as the primary metric given its medical relevance, while other complementary metrics such as specificity, accuracy, and AUC were also analyzed to provide a complete performance assessment.

Finally, the main goal is to compare and statistically evaluate the performance of these models in predicting cardiac risk, identifying which approach provides the best generalization while minimizing diagnostic errors.

2 Materials and Methods

2.1 Dataset

The study used the `cardiacRisk.csv` dataset containing 1,469 patient records with ten clinical attributes and one binary target variable (`Event`) indicating the occurrence of a cardiac event.

2.2 Data Preprocessing

The process of data preprocessing served as a fundamental step to protect data integrity and achieve dependable model outcomes. Missing values that took the value of `-1` were substituted with the median value from each respective feature because medians provide stronger resistance to extreme values while maintaining central tendency without being affected by outliers. `StandardScaler` was used to standardize numerical features only for models that need feature scale adjustments like Neural Networks and K-Nearest Neighbors because tree-based methods do not require any scaling.

SMOTE, a technique that creates synthetic samples of the minority class, was only applied to the training folds during cross-validation to avoid any data leakage because the target variable has a discrepancy of 31% positive cases against 69% negative cases. In order to maintain the original class distribution, the dataset was divided into 70% training data and 30% testing data using stratified sampling.

2.3 Models and Hyperparameter Tuning

The experiment tested five algorithms to determine their performance outcomes: Decision Tree, Logistic Regression, Neural Network, K-Nearest Neighbors and Naïve Bayes. The models underwent hyperparameter optimization through Repeated Stratified 5-Fold Cross-Validation, which included two repetitions to minimize random data partition effects while keeping equal group representation across folds to improve performance estimation for various training and validation splits.

- **Decision Tree:** maximum depth, minimum samples per leaf, criterion (*gini* or *entropy*), and pruning factor (*ccp_alpha*).
- **Neural Network:** number and size of hidden layers, learning rate, regularization term (*alpha*), and batch size.
- **Logistic Regression:** regularization strength (*C*) and penalty type (*L1* or *L2*).
- **KNN:** number of neighbors, distance metric (Manhattan or Euclidean), and weighting strategy (uniform or distance-based).

- **Naïve Bayes:** smoothing factor (*var_smoothing*).

2.4 Evaluation Metrics

Medical environments need to detect high-risk patients through sensitivity because it serves as the primary metric for identifying those who require immediate intervention to prevent major health issues from going undetected. These models choose sensitivity as their main goal because they need to identify most genuine cardiac events although this method produces an increased number of incorrect positive results.

Additionally, specificity was used to determine if emphasizing sensitivity would lead to an unacceptable rise in false positive results. F1-score to balance recall and precision, accuracy to assess overall consistency, and AUC to assess models' performance across different thresholds. This evaluation method allowed assessing both disease detection accuracy and prediction reliability, which doctors can apply to their work.

Sensitivity was used as the evaluation metric to determine optimal parameters during cross-validation thus, these parameters were applied for final model training. The method aligns optimization with clinical objectives to minimize incorrect negative results in cardiac risk assessments.

2.5 Statistical Analysis

A paired bootstrap approach was used for the statistical analysis in order to assess variability while preserving test sample alignment, allowing for a fair model comparison without relying on assumptions about a normal distribution, and because every participant participated in every model evaluation, a repeated-measures design was used. Wilcoxon tests with Holm correction were used to find significant differences between individual model pairs after the Friedman test was employed as the non-parametric technique to determine overall model differences.

3 Results

The two-part results show the outcomes from models that trained without data balancing and from models which used SMOTE for training. The statistical tests were applied only to the balanced setup to determine whether the differences between models were statistically significant.

3.1 Without SMOTE

Cross-validation on TRAIN

Model	AUC	SE	SP	F1	ACC
Decision Tree	0.669	0.572	0.806	0.571	0.733
Neural Network	0.776	0.707	0.696	0.599	0.699
Logistic Regression	0.850	0.586	0.902	0.649	0.804
KNN	0.851	0.533	0.923	0.624	0.802
Naïve Bayes	0.823	0.197	0.982	0.318	0.738

Table 1: Cross-validation results without SMOTE on the training set.

Best parameters (without SMOTE)
Decision Tree: ccp_alpha=0.0, criterion=gini, max_depth=9,min_samples_leaf=1.
Neural Network: alpha=1e-5, batch_size=128, hidden=(16),learning_rate_init=0.0001.
Logistic Regression: C=0.01,penalty=l2.
KNN: n_neighbors=15,p=1,weights=uniform.
Naïve Bayes: var_smoothing=1e-9.

Final Test set

Model	AUC	SE	SP	F1	ACC
Decision Tree	0.695	0.547	0.822	0.564	0.737
Neural Network	0.749	0.635	0.747	0.578	0.712
Logistic Regression	0.836	0.540	0.924	0.632	0.805
KNN	0.815	0.533	0.957	0.655	0.825
Naïve Bayes	0.799	0.190	0.990	0.313	0.741

Table 2: Test performance without SMOTE.

3.2 With SMOTE

Cross-validation on TRAIN

Model	AUC	SE	SP	F1	ACC
Decision Tree	0.763	0.707	0.712	0.602	0.711
Neural Network	0.825	0.818	0.691	0.656	0.730
Logistic Regression	0.830	0.762	0.738	0.651	0.746
KNN	0.838	0.730	0.783	0.661	0.767
Naïve Bayes	0.821	0.205	0.981	0.328	0.740

Table 3: Cross-validation results with SMOTE on the training set.

Best parameters (with SMOTE)
Decision Tree: ccp_alpha=0.0, criterion=gini, max_depth=3,min_samples_leaf=20.
Neural Network: alpha=1e-5, batch_size=128, hidden=(16),learning_rate_init=0.0001.
Logistic Regression: C=0.01,penalty=l1.
KNN: n_neighbors=13,p=2,weights=uniform.
Naïve Bayes: var_smoothing=1e-9.

Final Test set

Model	AUC	SE	SP	F1	ACC
Decision Tree	0.738	0.409	0.928	0.521	0.766
Neural Network	0.765	0.723	0.622	0.564	0.653
Logistic Regression	0.819	0.708	0.809	0.664	0.778
KNN	0.785	0.672	0.812	0.643	0.769
Naïve Bayes	0.800	0.197	0.993	0.325	0.746

Table 4: Test performance with SMOTE.

3.3 Statistical tests on Test set

Global Friedman tests

Metric	χ^2	p-value
AUC	3443.112	$< 10^{-6}$
Sensitivity	3627.364	$< 10^{-6}$
Specificity	3813.701	$< 10^{-6}$

Table 5: Friedman tests across models on the test set (with SMOTE).

Pairwise Wilcoxon with Holm correction

All pairwise comparisons between models are statistically significant ($p_{Holm} < 0.05$), confirming performance differences detected by the Friedman test.

Model 1	Model 2	p_{raw}	p_{Holm}
	AUC		
Logistic Regression	Bayesian	5.56×10^{-131}	2.22×10^{-130}
Logistic Regression	KNN	9.94×10^{-165}	5.97×10^{-164}
Logistic Regression	Neural Network	3.33×10^{-165}	3.33×10^{-164}
Neural Network	Decision Tree	3.33×10^{-165}	3.33×10^{-164}
Bayesian	KNN	6.09×10^{-90}	6.09×10^{-90}
	Sensitivity		
Logistic Regression	Bayesian	3.31×10^{-165}	3.31×10^{-164}
Logistic Regression	KNN	5.42×10^{-124}	1.08×10^{-123}
Logistic Regression	Neural Network	1.38×10^{-39}	1.38×10^{-39}
Bayesian	KNN	3.32×10^{-165}	3.31×10^{-164}
Decision Tree	Neural Network	3.32×10^{-165}	3.31×10^{-164}
	Specificity		
Logistic Regression	Bayesian	3.32×10^{-165}	3.32×10^{-164}
Logistic Regression	KNN	1.81×10^{-07}	1.81×10^{-07}
Logistic Regression	Neural Network	3.32×10^{-165}	3.32×10^{-164}
Neural Network	Decision Tree	3.32×10^{-165}	3.32×10^{-164}
Bayesian	KNN	3.32×10^{-165}	3.32×10^{-164}

Table 6: Pairwise Wilcoxon tests with Holm correction on the test set (with SMOTE).

4 Discussion

According to the experimental data, all models performed consistently in predicting cardiac risk, with AUC values on the test set ranging from 0.74 to 0.82. Although it led to slight decreases in specificity and accuracy rates, the SMOTE technique improved the system's ability to detect minority events, which in turn improved model sensitivity. Through the creation of synthetic samples, the sensitivity scores of the Decision Tree and Neural Network models rose from 0.55 and 0.64 to 0.71 and 0.72, respectively, improving their capacity to detect uncommon risk patterns. As a result, these models saw the largest performance gains.

The Neural Network was statistically determined to be the best-performing model based on the Friedman and Wilcoxon-Holm tests, and it had the highest sensitivity (0.72) of any classifier. This demonstrates that the neural model was the most successful at identifying individuals at high risk, which is consistent with the medical goal of reducing false negatives. Its lower specificity (0.62) and accuracy (0.65), which are indicative of its greater false positive rate, render it less dependable for autonomous deployment in the absence of further calibration.

From a clinical perspective, the overall balance was better with the Logistic Regression model. It had a sensitivity that was about the same as the Neural Network's (0.71) with an AUC of 0.82, but it had higher specificity (0.81) and accuracy (0.78). For open and transparent decision support in the healthcare sector, this performance pattern shows that Logistic Regression generated consistent predictions with interpretable coefficients and remained constant across both configurations.

5 Conclusion

The research evaluated different supervised learning algorithms to predict cardiac risk from clinical data through two separate experiments which included both balanced and unbalanced datasets with SMOTE. The results showed that dataset balancing led to small specificity reductions yet it improved the models' ability to detect patients at high risk. The study results show that medical decision support systems need to handle class imbalance during their development process because it affects predictive model performance.

Among all investigated models, the Neural Network demonstrated the maximum sensitivity and was statistically chosen as the best-performing model. However, Logistic Regression offered a more steady and interpretable trade-off between sensitivity, specificity, and accuracy, making it better suited for clinical use. Future study should include evaluating the models on bigger, independent clinical datasets, as well as investigating ensemble strategies that integrate many models to improve sensitivity while maintaining specificity.

References

[1] Scikit-learn Documentation. Repeated Stratified K-Fold Cross-Validation. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RepeatedStratifiedKFold.html

[2] Imbalanced-learn Documentation. Synthetic Minority Over-sampling Technique (SMOTE). Available at: https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.SMOTE.html

[3] SciPy Documentation. Wilcoxon Signed-Rank Test. Available at: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.wilcoxon.html

[4] GeeksforGeeks. Friedman Test – Non-parametric Statistical Test. Available at: https://www.geeksforgeeks.org/dsa/friedman-test/

[5] GPT Tutor Pro. Machine Learning Evaluation Mastery: How to Use Bootstrap for Model Evaluation and Comparison. Available at: https://gpttutorpro.com/machine-learning-evaluation-mastery-how-to-use-bootstrap-for-model-evaluation-and-comparison/