Final Report: Predicting Heart Disease with Machine Learning

This report summarizes the exploration of machine learning models for predicting heart disease in a dataset. The primary goal was to achieve the highest possible accuracy while considering interpretability as a secondary factor between the models.

Data Preprocessing

The data ("HeartDisease.csv") was loaded and analyzed for missing values. Feature scaling was applied to numerical features to ensure all features were on a similar scale for model training. Categorical features were identified for potential one-hot encoding if necessary.

Model Evaluation Strategy

Two evaluation strategies were employed: train-test split and K-fold cross-validation (Kfold). Train-test split provides a quick performance estimate but can be susceptible to the specific data split. Kfold provides a more robust estimate of model performance by evaluating on multiple data splits.

Models Evaluated

The following machine learning models were evaluated for their ability to predict heart disease:

- Logistic Regression
- Random Forest (Tuned)
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)
- XGBoost (Tuned)

Results

Train-Test Split

The initial evaluation used a train-test split (80% training data, 20% testing data). Here's a summary of the results:

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Logistic Regression	0.8361	0.8667	0.8125	0.8387	0.8373
Random Forest (Tuned)	**0.9016**	**0.9062**	**0.9062**	**0.9062**	**0.9014**
SVM	0.8525	0.8710	0.8438	0.8571	0.8529
KNN	0.8689	0.8750	0.8750	0.8750	0.8685
XGBoost (not tuned)	0.8525	0.8710	0.8438	0.8571	0.8529

K-Fold Cross-Validation

K-fold cross-validation (Kfold with k=5) was used to provide a more robust estimate of model performance.

Here's a summary of the Kfold CV results for all models, including the tuned hyperparameters for Random Forest and XGBoost:

Model	Mean Accuracy (Std. Dev)	Mean Precision (Std. Dev)	Mean Recall (Std. Dev)	Mean F1-Score (Std. Dev)	Mean AUC-ROC (Std. Dev)	Tuned Hyperparameters
Logistic Regression	0.8311 (0.0372)	0.8237 (0.0549)	0.8840 (0.0362)	0.8509 (0.0281)	0.8251 (0.0404)	N/A
Random Forest (Tuned)	**0.8277 (0.0565)**	**0.8226 (0.0615)**	**0.8807 (0.0607)**	**0.8484 (0.0448)**	**0.8242 (0.0591)**	n_estimators=150, max_depth=8
SVM	0.8245 (0.0554)	0.8150 (0.0715)	0.8922 (0.0434)	0.8487 (0.0375)	0.8185 (0.0605)	N/A

CHOOSING THE BEST MODEL

Based on the evaluation results, the **tuned Random Forest** model achieved the **highest mean accuracy (0.8277) with K-fold cross-validation (Kfold with k=5)**. Here's a table summarizing the key considerations:

Model	Train-Test Split Accuracy	Kfold CV Mean Accuracy	Interpretability	Tuned Hyperparameters
Logistic Regression	0.8361	0.8311 (Std. Dev: 0.0372)	High	N/A

Random Forest (Tuned)	**0.9016**	**0.8277 (Std. Dev: 0.0565)**	Moderate	n_estimators=150,
				max_depth=8

SVM	0.8525	0.8245 (Std. Dev: 0.0554)	Moderate	N/A
KNN	0.8689	0.8179 (Std. Dev: 0.0476)	Moderate	N/A
XGBoost (not tuned)	0.8525	0.7945 (Std. Dev: 0.0333)	Low	N/A

Why K-Fold Cross-Validation Was Chosen Over Train-Test Split

While train-test split offers a quick and straightforward approach, K-fold cross-validation was chosen for several reasons:

- **More Robust Performance Estimation:** Train-test split relies on a single

random split of the data, which can be susceptible to the specific split chosen. Kfold addresses this by iteratively training and testing on different folds, providing a more comprehensive and less biased estimate of the model's generalizability on unseen data.
- **Efficient Use of Data:** Train-test split typically holds out a portion of the data for testing, which is not used for training. Kfold utilizes all available data for training and testing across folds, maximizing the information used to learn the model and assess its performance. This is particularly beneficial for datasets with limited samples.
- **Reduced Variance:** Train-test splits can suffer from high variance in performance estimates, especially with smaller datasets. A single split might not capture the model's true capabilities. Kfold averages the performance across multiple splits, leading to a more stable and reliable estimate.

In the case of predicting heart disease, a reliable understanding of the model's performance on new patients is crucial. Kfold helps achieve this by providing a more robust and generalizable evaluation compared to a single train-test split.

Why Tuned Random Forest is the Best Choice

Here's why Tuned Random Forest is the best choice for this specific application:

- **Accuracy:** Achieved the highest mean accuracy on K-fold CV, providing a more robust measure of performance that is likely to generalize well to unseen data.
- **Interpretability:** While not as interpretable as Logistic Regression, Random Forest offers better interpretability than complex models like XGBoost. Feature importance analysis can help understand which features contribute most to the model's predictions.
- **Hyperparameter Tuning:** Tuning the hyperparameters (n_estimators=150, max_depth=8) improved the model's performance compared to the untuned version.

Overall, the tuned Random Forest model offers a good balance between accuracy, interpretability, and gaining insights from the data through feature importance analysis.