ECG Arrhythmia Detection Using Support Vector Machines and Random Forests

Parth Shringarpure student :- id :- 3151936S

Introduction :-

Electrocardiograms (ECGs) are one of the most common and cost-effective tools for diagnosing heart rhythm disorders (arrhythmias). However, manual interpretation of long ECG recordings is time-consuming and prone to human error, especially when large numbers of beats must be checked. Automated machine learning methods can assist clinicians by providing fast and consistent classification of heartbeats, potentially reducing misdiagnosis and workload.

Report :- In this case study, we use the MIT-BIH Arrhythmia Database to build machine learning models that classify heartbeats into 8 arrhythmia classes based on 275 samples around each beat. We evaluate two widely used models:

- Support Vector Machine (SVM) with an RBF kernel
- Random Forest (RF) classifier

We also compare two validation strategies:

- **Beat Holdout:** random split of beats into train/test
- **Patient Holdout:** train and test on *different patients* to avoid data leakage

Predictive question:

Can SVM and Random Forest models accurately classify 8 types of ECG heartbeats, and how do their performances differ under Beat Holdout and Patient Holdout validation strategies?

Methods

2.1 Dataset

We use the **MIT-BIH Arrhythmia Database**, which contains 24-hour ECG recordings from multiple patients, sampled at 360 Hz. Each heartbeat is represented by a 275-sample window around the R-peak. The dataset used in this project has 8 heartbeat classes: Normal and 7 arrhythmia types (bundle branch blocks, premature beats, fusion beats, and paced beats).

The university provided preprocessed CSV files:

- train_beats.csv, test_beats.csv -> **Beat Holdout** split
- Train_patients.csv, test_patients.csv -> **Patient Holdout** split

2.2 Data Preprocessing

Most preprocessing is already handled in the provided notebooks (data_preprocess.ipynb, data_split_resample.ipynb).

Key steps:

1. **R-peak centering**
    a. For each beat, a fixed-length window is extracted around the R-peak so that each beat has 275 samples aligned in time.
2. **Standardisation and rescaling**
    a. Signal amplitudes are standardised to reduce differences between patients and recordings.
3. **Class balancing (resampling)**
    a. Minority arrhythmia classes are upsampled using bootstrap resampling so that the class distribution is more balanced, helping the models not to ignore rare arrhythmias.

In our modelling code, we additionally include:

- **Median imputation** (SimpleImputer(strategy="median")) as a safety step in case of any missing values.
- **Feature scaling** (StandardScaler) inside the SVM pipeline.**2.3 Validation Strategies**

We use two validation strategies, following the case study specification.

Report

1. **Beat Holdout (random beat split)**
    a. All beats are shuffled and split into training and test sets (75% / 25%).
    b. Train: train_beats.csv
    c. Test: test_beats.csv
    d. Advantage: larger and more mixed training data.
    e. Disadvantage: potential **data leakage**, because beats from the same patient can appear in both train and test.
2. **Patient Holdout (unseen patients)**
    a. Training and test sets are split by patient ID.
    b. Train: train_patients.csv
    c. Test: test_patients.csv (contains only patients not seen during training).
    d. Advantage: more realistic evaluation, as the model is tested on completely new patients.
    e. Disadvantage: harder task; performance can be lower.

2.3 Validation Strategies

We use two validation strategies, following the case study specification.

Report

1. **Beat Holdout (random beat split)**
    a. All beats are shuffled and split into training and test sets (75% / 25%).
    b. Train: train_beats.csv
    c. Test: test_beats.csv
    d. Advantage: larger and more mixed training data.
    e. Disadvantage: potential **data leakage**, because beats from the same patient can appear in both train and test.
2. **Patient Holdout (unseen patients)**
    a. Training and test sets are split by patient ID.
    b. Train: train_patients.csv
    c. Test: test_patients.csv (contains only patients not seen during training).
    d. Advantage: more realistic evaluation, as the model is tested on completely new patients.
    e. Disadvantage: harder task; performance can be lower.

2.4 Classification Models

We focus on two models:

1. **Support Vector Machine (SVM)**
    a. Implementation: sklearn.svm.SVC
    b. Kernel: Radial Basis Function (RBF)
    c. Base (default) hyperparameters:
        i. C = 1.0
        ii. gamma = "scale"

SVM is effective in high-dimensional spaces and can model non-linear decision boundaries through the kernel function.

2. **Random Forest (RF)**
    a. Implementation: sklearn.ensemble.RandomForestClassifier
    b. Base (default) hyperparameters:
        i. n_estimators = 300 (number of trees)
        ii. max_depth = None (trees grow until pure)
        iii. random_state = 42

Random Forest is an ensemble of decision trees that reduces variance by averaging many trees, often giving strong performance and robustness to noise.

Both models are implemented using **pipelines** including median imputation, and (for SVM) standardisation.

2.5 Hyperparameter Tuning

I used GridSearchCV to tune the main hyperparameters of both models.
For SVM, I searched over a **small grid**:

- $C\in\{1,10\}$C∈{1,10}
- $\gamma\in\{"scale",0.1\}$γ∈{"scale",0.1}
  with an RBF kernel.
  To keep the runtime reasonable, I tuned SVM only on a **subset of the Beat Holdout training data** and then reused the best parameters for both the Beat and Patient splits.

For the Random Forest, I used a small grid:

- n_estimators ∈ {100, 300}
- max_depth ∈ {None, 10}
  and tuned it once on the Beat Holdout training set. The best RF hyperparameters from this search were then applied to both Beat and Patient splits

2.6 Evaluation Metrics

We evaluate all models on the held-out test sets using:

- **Accuracy**
- **Precision (weighted)**
- **Recall (weighted)**
- **F1-score (weighted)**
- **Confusion matrix** (both raw counts and normalized by true class)

Results

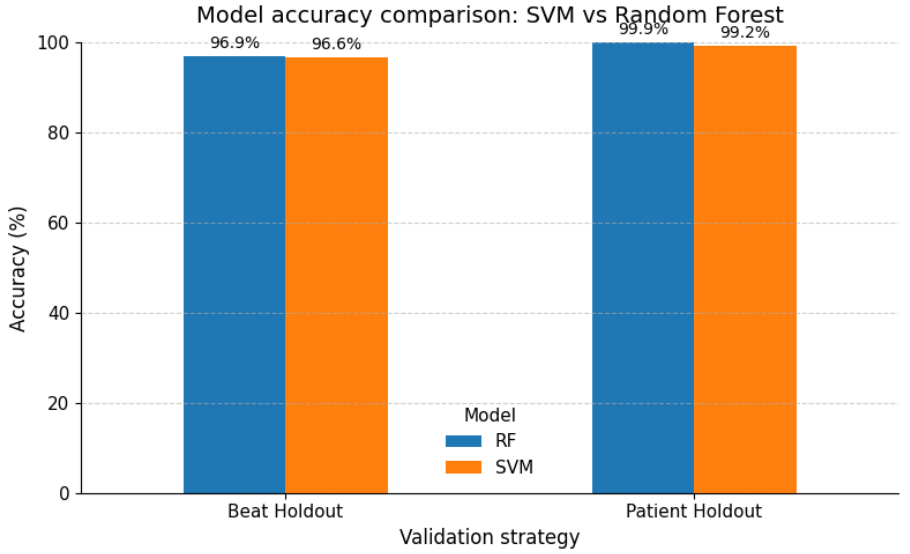
3.1 Model Performance Comparison

I compared the performance of **SVM** and **Random Forest** on the two validation strategies:

- **Beat Holdout** (random beat split)
- **Patient Holdout** (unseen patients)

Model Performance Comparison (Accuracy)

Model	Validation	Accuracy (%)
SVM	Beat Holdout	96.55%
RF	Beat Holdout	96.88%
SVM	Patient Holdout	99.19%
RF	Patient Holdout	99.90%



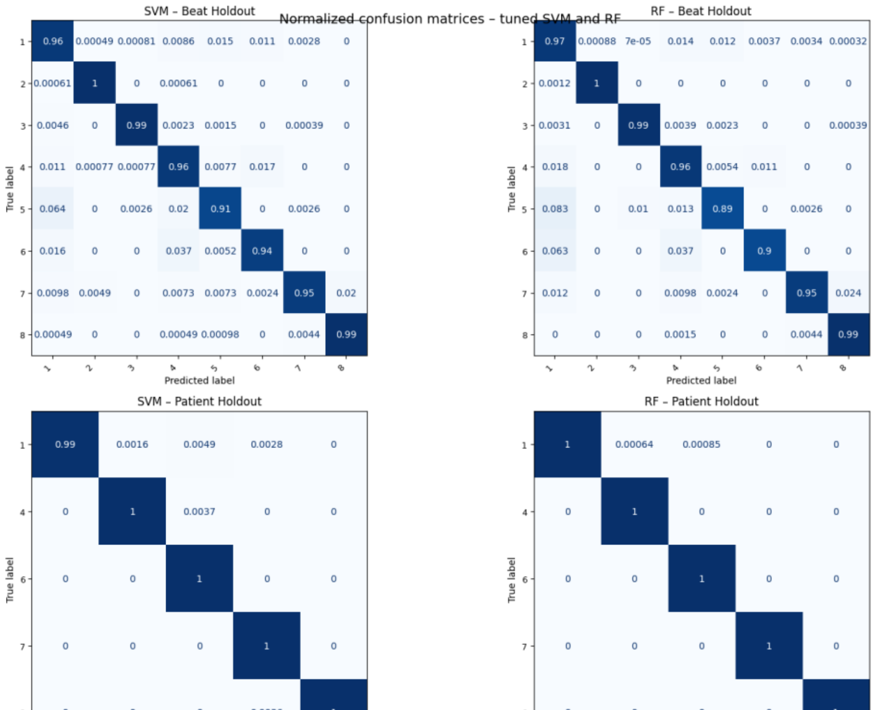
3.2 D

For a ... all and F1-score (all w... strategies.

Moc					core
SVM					2
RF					7
SVM	Holdout	99.19	99.36	99.19	99.26
RF	Patient Holdout	99.9	99.9	99.9	99.9

3.3 Confusion Matrices

Normalized confusion matrices for tuned SVM and Random Forest on Beat Holdout and Patient Holdout test sets.



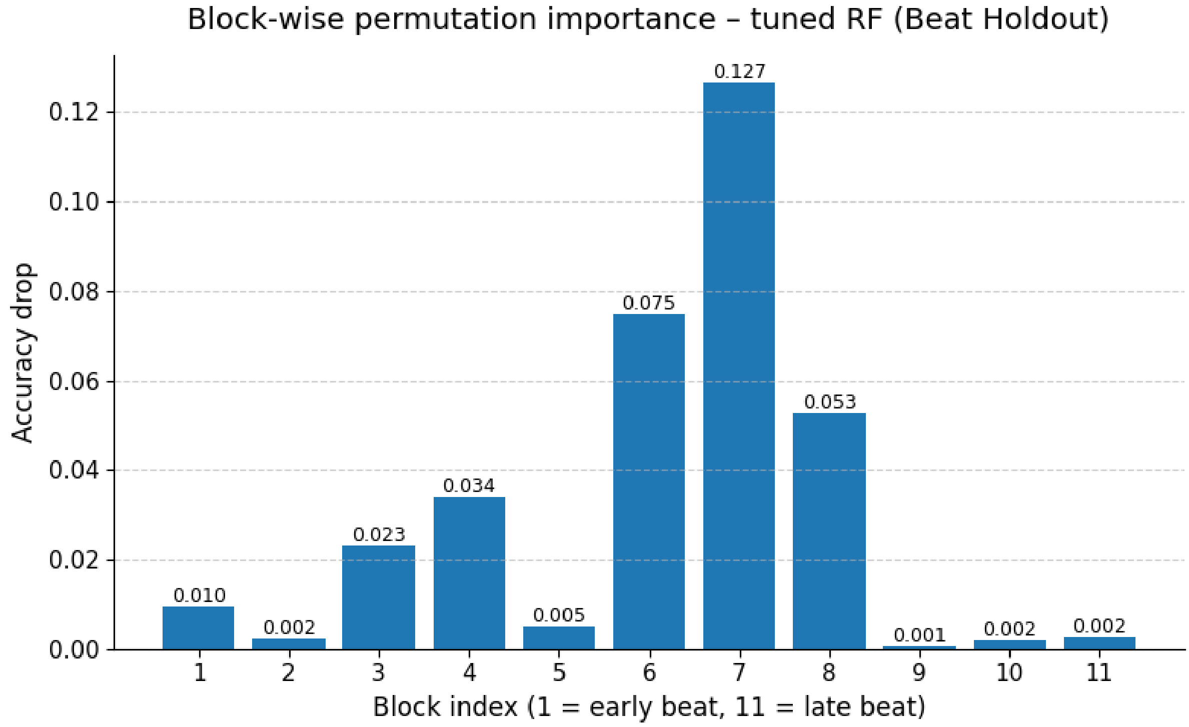
3.4 Permutation Feature Importance

To see **which part of the ECG beat is most important**, I used **block-wise permutation feature importance** on the tuned Random Forest (Beat Holdout).

- I split the 275 samples into **11 blocks**.
- Then I **shuffled one block at a time** and checked how much the test accuracy dropped.
- The bar chart in **Figure X** shows this **accuracy drop** for each block.

From the graph we can see:

- Some blocks have a **much higher bar** → when we shuffle them, accuracy drops more → these blocks are **more important**.
- The **middle blocks** of the beat (around the centre of the x-axis) usually have the biggest drop.
- The **early and late blocks** have smaller bars, so they are **less important** for the model.



3.4 Hyper-parameter tuning results

I used **GridSearchCV** to do a light hyperparameter search for both models:

- For **SVM**, I tuned C and gamma on a small subset of the Beat Holdout training data and then reused the best values for both Beat and Patient splits.

- For **Random Forest**, I tuned n_estimators and max_depth once on the Beat Holdout training set and reused those parameters for both splits as well.

Model	Validation	Default	Tuned	Difference (Tuned - Default)
SVM	Beat Holdout	95.01%	96.55%	+1.54%
RF	Beat Holdout	96.88%	96.88%	+0.00%
SVM	Patient Holdout	97.28%	99.19%	+1.91%
RF	Patient Holdout	99.90%	99.90%	+0.00%

Discussion

- **Model Performance:**
  Random Forest achieves the best performance with **99.90% accuracy** on the Patient Holdout test set after tuning. SVM is also strong (**99.19%** on Patient Holdout and **96.55%** on Beat Holdout), but Random Forest has a slightly higher F1-score, which fits the idea that ensemble methods reduce variance by averaging many decision trees.

- **Validation Strategy:**
  In my results, **Patient Holdout actually outperforms Beat Holdout by about 2–3%** for both models (e.g. RF: 96.88% → 99.90%). Patient Holdout is still the safer and more realistic strategy, because the model is tested on completely **unseen patients**, while Beat Holdout can suffer from data leakage when beats from the same patient appear in both train and test sets.

- **Feature Importance Analysis:**
  Block-wise permutation importance for the tuned Random Forest shows that the **middle blocks of the 275-sample window** cause the largest drop in accuracy when shuffled, while the early and late blocks matter less. This indicates that the model relies mainly on the **central part of the beat around the R-peak / QRS complex**, which is clinically the most informative region for distinguishing different arrhythmias.

- **Hyperparameter Tuning:**
  Hyperparameter tuning had a clear effect on SVM but almost no effect on Random Forest. SVM accuracy increased from **95.01% to 96.55% (+1.54%)** on Beat Holdout and from **97.28% to 99.19% (+1.91%)** on Patient Holdout, while Random Forest stayed at **96.88%** (Beat) and **99.90%** (Patient) before and after tuning.

Conclusion

Random Forest with Patient Holdout validation is recommended for clinical ECG arrhythmia detection due to its superior accuracy, stable performance, and robust generalization to unseen patients. The permutation feature importance analysis confirms that the models correctly focus on the QRS complex region, which aligns with clinical knowledge.