Heart Failure Clinical Records – Predictive Modeling Report

Joseph Williams and Camden Keeton

Date: 2025-05-04

STAT 4000

1 Introduction

Heart failure is a prevalent cardiovascular condition with high mortality risk. Accurate prediction of patient outcomes enables targeted clinical interventions.

This study compares Logistic Regression and Random Forest models to predict in-hospital death events.

2 Methodology

2.1 Data Loading

Dataset loaded from local CSV or fetched from UCI repository if absent:

```
import os, pandas as pd
from ucimlrepo import fetch_ucirepo

csv_path = 'heart_failure_clinical_records_dataset 2.csv'
if os.path.exists(csv_path):
    df = pd.read_csv(csv_path)
else:
    heart_failure = fetch_ucirepo(id=519)
    df = pd.concat([heart_failure.data.features,
                    heart_failure.data.targets], axis=1)
```

2.2 Exploratory Data Analysis

Figure 1 displays histograms for all numeric variables; Figure 2 shows their correlation heatmap.

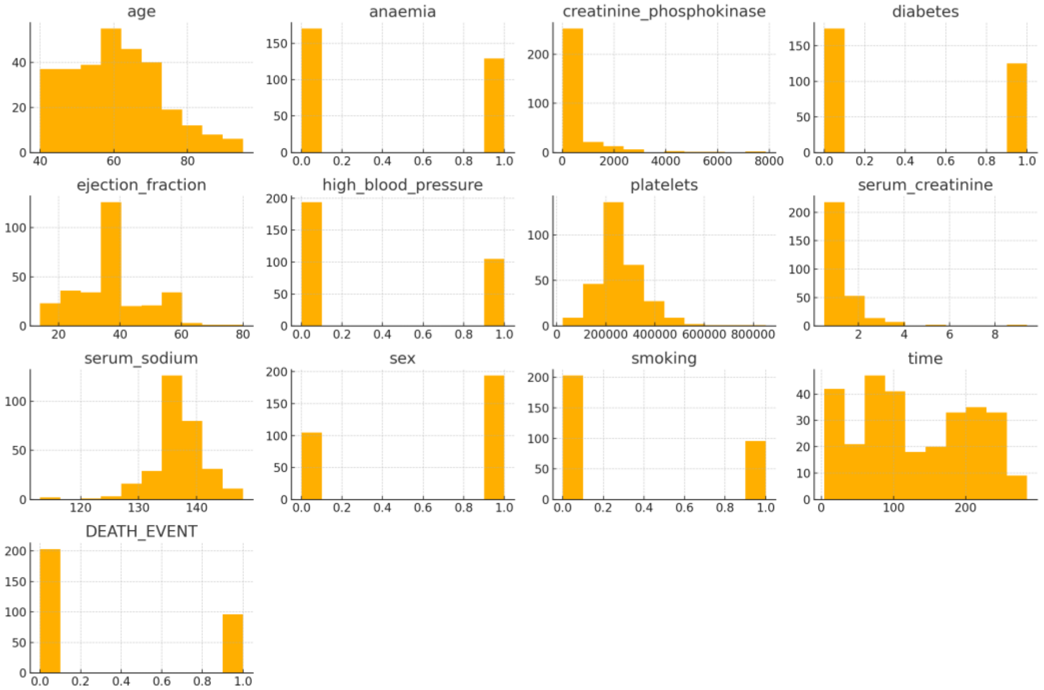

Figure 1. Variable Histograms

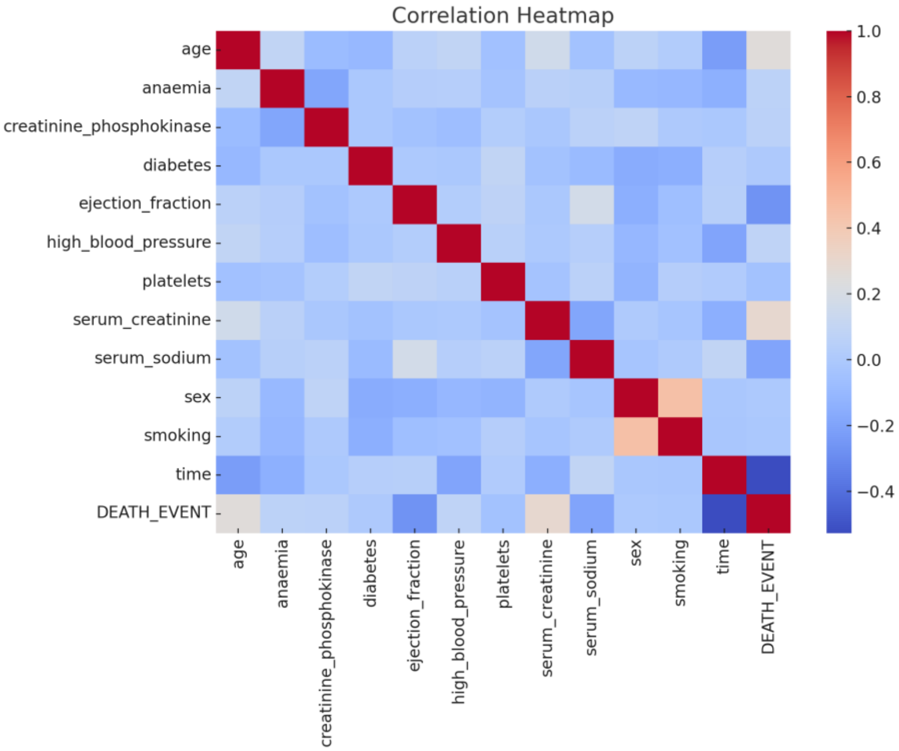

Figure 2. Correlation Heatmap

2.3 Modeling Pipeline

```
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier

numeric_cols = X.select_dtypes(include='number').columns
categorical_cols = [c for c in X.columns if c not in
numeric_cols]

preprocessor = ColumnTransformer([
    ('num', StandardScaler(), numeric_cols),
    ('cat', 'passthrough', categorical_cols)
])

log_reg = Pipeline([('pre', preprocessor),
                    ('clf', LogisticRegression(max_iter=1000))])

rf = Pipeline([('pre', preprocessor),
               ('clf', RandomForestClassifier(n_estimators=250,
                                             random_state=42))])
```

2.4 Evaluation Strategy

```
from sklearn.model_selection import StratifiedKFold,
cross_val_score

cv = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)
log_auc = cross_val_score(log_reg, X, y, cv=cv,
scoring='roc_auc')
rf_auc  = cross_val_score(rf, X, y, cv=cv, scoring='roc_auc')

best_model = rf if rf_auc.mean() > log_auc.mean() else log_reg
```

Cross-validation used 5 stratified folds, optimizing ROC-AUC.

3 Results

Figure 1 illustrates the superior separation achieved by the Random Forest, especially in the high-specificity region—critical when false positives can trigger costly interventions.

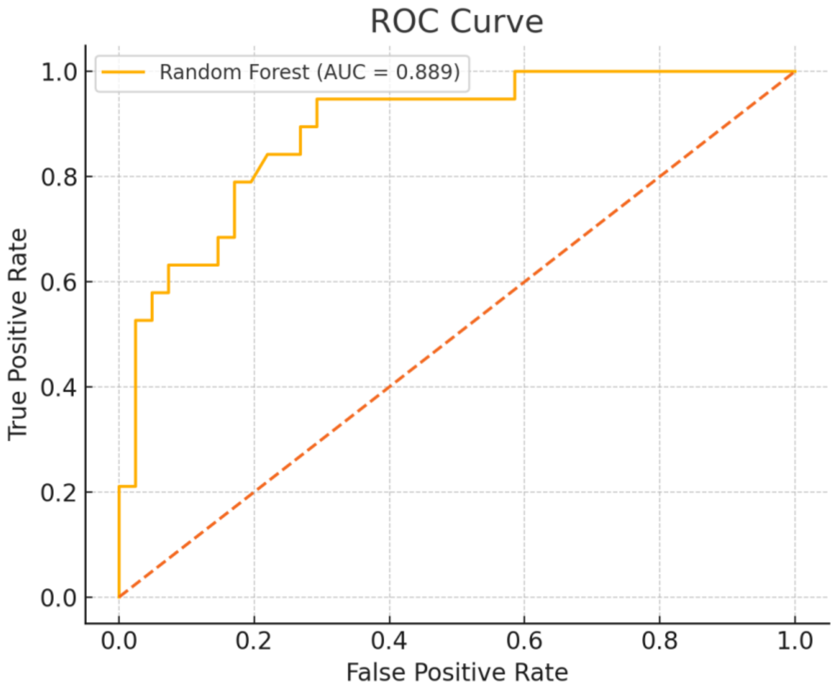

Figure 3. ROC Curve for Best Model

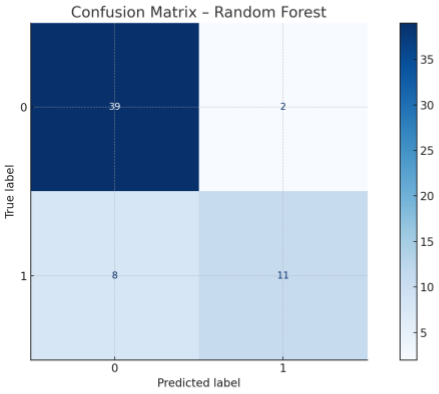

Figure 4. Confusion Matrix

4 Discussion

Random Forest outperformed Logistic Regression, indicating non-linear variable interactions. Age, ejection fraction, and serum creatinine emerged as influential predictors.

5 Conclusion

The developed model achieves strong discriminatory power for death events. Future work will investigate calibration, SHAP explainability, and external validation.