Using Machine Learning Models to Predict Heart Disease

Helen Liang, Ivy Zhao, Xiaotong Zhao

Background & Problem

- The prevalence of heart disease, a general term referring to several types of heart conditions including coronary artery disease (CAD), remains one of the leading causes of death in the United States, causing about 1 in every 5 deaths and posing a significant threat to public health.

- Early detection and intervention can significantly decrease the risk of heart disease, promoting better quality of life and reducing heart disease-related mortality.

- ***Objective:*** To perform predictions to identify patients with high risk of developing heart disease through utilization of various machine learning models as well as to speed up the diagnostic process based on medical information provided, hence allowing for early preventive measures.

Data

- This Heart Disease dataset is from **UC Irvine Machine Learning Repository**.

- Combined data from the Cleveland, Hungary, Switzerland, and VA Long Beach databases.

- Contains 920 observations, each representing a patient's medical record.

- Out of 76 variables, 14 variables are available for public use, representing demographic information, physiological measurements, and patient medical history (See Table 1).

- Outcome variable, “num”, is converted from continuous to factor with “yes” indicating presence of and “no” indicating absence of heart disease.

TABLE 1. Description of the Heart Disease Dataset, UC Irvine Machine Learning Repository, 6/30/1988

Variable Name	Description	Variable Tyoe
age	Patient age (in years)	Continuous
sex	Gender of patient	Binary
cp	Chest pain type	Ordinal
trestbps	Resting blood pressure (in mmHg)	Continuous
chol	Serum cholesterol (in mg/dl)	Continuous
fbs	Fasting blood sugar > 120 mg/dl	Binary
restecg	Resting electrocardiographic results	Ordinal
thalach	Maximum heart rate achieved	Continuous
exang	Exercise included angina	Binary
oldpeak	ST depression induced by exercise relative to rest	Continuous
slope	The slope of the peak exercise ST segment	Ordinal
ca	Number of major vessels (0-3) colored	Ordinal
thal	Thalassemia	Ordinal
num	Diagnosis of heart disease	Binary

Data (cont.)

Distribution of our dataset:

- Age (years): mean (SD) is 53.51 (9.42)
- 78.91% of patients are females (78.91%)
- 53.91% of patients have asymptomatic chest pain type
- Resting blood pressure (mmHg): mean (SD) is 132.1 (0.97)
- Serum cholesterol (mg/dl): mean (SD) is 199.1 (0.98)
- 83.37% of patients do not have fasting blood sugar > 120 mg/dl
- 60.02% of patients have normal resting electrocardiographic results
- Maximum heart rate achieved (bpm): mean (SD) is 137.5 (0.97)
- 61.04% of patients do not have exercise included angina
- ST depression induced by exercise relative to rest: 0.88 (0.97)
- 56.46% of patients have flat slope of the peak exercise ST segment
- 58.58% of patients with 0 number of major vessels colored by fluoroscopy
- 45.16% of patients do not have thalassemia
- 55.33% of patients have heart disease

- Positive correlation between "age" and "trestbps", indicating that blood pressure increases as age increases
- Negative correlation between "age" and "thalach", indicating that maximum heart rate decreases as age increases
- Negative correlation between "oldpeak" and "thalach", indicating that ST depression decreases as maximum heart rate increases
- "chol" shows little to no correlation with "thalach" and "oldpeak"

FIGURE 1. Correlation Heatmap of Continuous Variables.

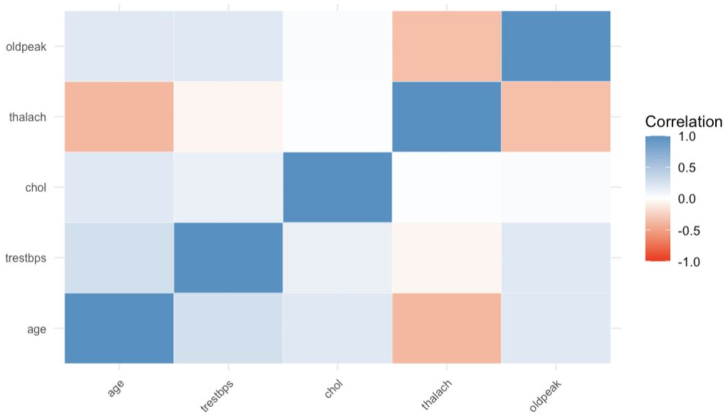


Table 1. Summary of the Heart Disease Datase (n=920), Mean(SD), Median(min, max) are repoeted for continuous variables. Frequencies(%) are reported for categorical variables.

Sociodemographic characteristics	Total (N = 920)
Age	
Mean (SD)	53.51(9.42)
Median (IQR)	54 (47,60)
Sex	
Male	194 (21.09%)
Female	726 (78.91%)
Chest Pain Type	
Typical Angina	46 (5.0%)
Atypical Angina	174 (18.91%)
Non-anginal Pain	204 (22.17%)
Asymptomatic	496 (53.91%0
Resting Blood Pressure (in mmHg)	
Mean (SD)	132.1 (0.97)
Median (IQR)	130(120,140)
Serum Cholesterol (in mg/dl)	
Mean (SD)	199.1 (0.98)
Median (IQR)	223 (175,268)
Fasting Blood Sugar > 120 mg/dl	
FALSE	692 (83.37%)
TRUE	138 (16.63%)
resting electrocardiographic results	
Normal	551 (60.02%)
Having ST-T Wave Abnormality	179 (19. 50%)
Showing Probable	188 (20.48%)
Maximum Heart Race Rchieved	
Mean (SD)	137.5(0.97)
Median (IQR)	140 (120,157)
Exercise Included Angina	
No	528 (61.04%)
Yes	337 (38.96%)
ST Depression Induced by Exercise Relative to Rest	
Mean (SD)	0.88 (0.97)
Median (IQR)	0.5 (0, 1.5)
Slope of the Peak Exercise ST Segment	
Upsloping	203 (33.22%)
Flat	345 (56.46%)
Downsloping	63 (10.31%)
Number of Major Vessels Colored by Fluoroscopy	
0	181 (58.58)
1	67 (21.68%)
2	41 (13.26%)
3	20 (6.47%)
Thalassemia	
Normal	196 (45.16%)
Fixed Defect	46 (10.60%)
Reversible Defect	192 (44.23%)
Diagosis of Heart Disease	
No	411 (44.67%)
Yes	509 (55.33%)

Data (cont.): Feature Selection

- Check for NA: Feature “ca” and “thal” are dropped, as more than half of observations have missing values for these 2 features.

  ca (number of major vessels colored): 611 missing values;

  thal (thalassemia): 486 missing values;

- Forward Stepwise Selection with Adjusted R-squared and Backward Stepwise Selection with Cp.
- 10 features are selected: age, sex, cp, chol, fbs1, restecg, thalach, exang, oldpeak, slope.

```
(Intercept)         age         sex          cp        chol        fbs1    restecg1    restecg2
 -0.3114297   0.1610103   0.3459368   0.2403301  -0.2036175   0.2374377   0.1548254   0.2555852
    thalach      exang1     oldpeak      slope2
 -0.1119386   0.1838280   0.3246255   0.1088429
```

Methods

- 5 machine learning models suitable for binary classification will be incorporated for heart disease diagnosis prediction:
    - Logistic regression
    - Support Vector Machines (SVM)
    - K-Nearest Neighbors (KNN)
    - Random Forest
    - Gradient Boosting

- The dataset will be split into training (80%) and testing (20%) sets.

- K-fold cross-validation will be implemented to tune hyperparameter in order to achieve optimal model performance since our dataset is small.

- Training error, testing error, accuracy, sensitivity, specificity, F1 score, and area under the ROC curve (AUCROC) will then be calculated respectively for both initial and tuned models.

Methods/Results - Logistic Regression

- **Logistic regression** is used as a baseline model due to its simplicity and interpretability to solve classification tasks, in this case, predicting heart disease status (0 = no heart disease, 1 = has heart disease).

- Training error rate = 0.1779
- Testing error rate = 0.2065
- Accuracy = 79.35%
- Sensitivity = 0.6986
- Specificity = 0.8559
- F1 Score = 0.7286
- AUCROC = 0.8841

```
[1] 0.1779891  <- Training Error
[1] 0.2065217  <- Testing Error
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  51  16
       yes 22  95

               Accuracy : 0.7935
                 95% CI : (0.7277, 0.8495)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 2.936e-08

                  Kappa : 0.5624

 Mcnemar's Test P-Value : 0.4173

            Sensitivity : 0.6986
            Specificity : 0.8559
         Pos Pred Value : 0.7612
         Neg Pred Value : 0.8120
              Precision : 0.7612
                 Recall : 0.6986
                     F1 : 0.7286
             Prevalence : 0.3967
         Detection Rate : 0.2772
   Detection Prevalence : 0.3641
      Balanced Accuracy : 0.7772

       'Positive' Class : no

Setting levels: control = no, case = yes
Setting direction: controls < cases
Area under the curve: 0.8841
```

Methods/Results - Logistic Regression with Regularization (cont.)

- Incorporate **elastic net regularization** technique and use **k-fold cross-validation** for tuning to avoid overfitting.
- **Alpha**: 0 to 1, elastic net combining L1 (lasso) and L2 (Ridge) regulation.
- **Lambda**: controls overall strength of regularization.
- The final values used for the model were:
  alpha = 0 (solely ridge) and lambda = 0.05

- Training error rate = 0.2486
- Testing error rate = 0.2717
- Accuracy = 72.83%
- Sensitivity = 0.6849
- Specificity = 0.7568
- F1 Score = 0.6667
- AUCROC = 0.7208

```
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  50  27
       yes 23  84

               Accuracy : 0.7283
                 95% CI : (0.6579, 0.7911)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 0.0002639

                  Kappa : 0.4376

 Mcnemar's Test P-Value : 0.6713732

            Sensitivity : 0.6849
            Specificity : 0.7568
         Pos Pred Value : 0.6494
         Neg Pred Value : 0.7850
              Precision : 0.6494
                 Recall : 0.6849
                     F1 : 0.6667
             Prevalence : 0.3967
         Detection Rate : 0.2717
   Detection Prevalence : 0.4185
      Balanced Accuracy : 0.7208

       'Positive' Class : no

[1] 0.2486413   <- Training Error

[1] 0.2717391   <- Testing Error

Area under the curve: 0.7208
```

Methods/Results - Support Vector Machines (SVM)

- **SVM:** find a hyperplane that best separates the classes in the feature space.
- **Gaussian kernel:** handling non-linear relationships between features.

- Training error rate = 0.1576
- Testing error rate = 0.1956
- Accuracy = 80.43%
- Sensitivity = 0.6849
- Specificity = 0.8829
- F1 Score = 0.7353
- AUCROC = 0.7839

```
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  50  13
       yes 23  98

               Accuracy : 0.8043
                 95% CI : (0.7396, 0.859)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 4.217e-09

                  Kappa : 0.5814

 Mcnemar's Test P-Value : 0.1336

            Sensitivity : 0.6849
            Specificity : 0.8829
         Pos Pred Value : 0.7937
         Neg Pred Value : 0.8099
              Precision : 0.7937
                 Recall : 0.6849
                     F1 : 0.7353
```

Methods/Results - Support Vector Machines with K-Fold (cont.)

- **K-fold cross-validation:** tune regularization hyperparameters to avoid overfitting.
- **Cost, C:** controls the trade-off between maximizing the margin and minimizing the classification error.
- **Sigma, σ:** controls the smoothness of the decision boundary.
- The optimal values for svm model were sigma = 0.022 and C = 1.

- Training error rate = 0.1576
- Testing error rate = 0.1902
- Accuracy = 80.98%
- Sensitivity = 0.6986
- Specificity = 0.8829
- F1 Score = 0.7445
- AUCROC = 0.7817

```
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  51  13
       yes 22  98

               Accuracy : 0.8098
                 95% CI : (0.7455, 0.8638)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 1.52e-09

                  Kappa : 0.594

 Mcnemar's Test P-Value : 0.1763

            Sensitivity : 0.6986
            Specificity : 0.8829
         Pos Pred Value : 0.7969
         Neg Pred Value : 0.8167
              Precision : 0.7969
                 Recall : 0.6986
                     F1 : 0.7445
```

Methods/Results - K-Nearest Neighbors (KNN)

- **K-Nearest Neighbors (KNN)** Used to estimate the response of a data point by capturing local patterns based on its K-nearest neighbors. The hyperparameter, K = 12, is selected by looking at the lowest test error to compute the metrics for producing the model with the best performance using KNN.

- Training error rate = 0.1671
- Testing error rate = 0.1956
- Accuracy = 80.43%
- Sensitivity = 0.7397
- Specificity = 0.8468
- F1 Score = 0.75
- AUCROC = 0.7796

```
[1] 0.1671196 <-- Training error
[1] 0.1956522 <-- Testing error
[[1]]
          actual
predicted  no yes
      no  266  51
      yes  72 347

[[1]]
          actual
predicted no yes
      no  54  17
      yes 19  94

[1] 0.8043478 <-- Accuracy
[1] 0.739726 <-- Sensitivity
[1] 0.8468468 <-- Specifivity
[1] 0.75 <-- F1 Score
Setting levels: control = no, case = yes
Setting direction: controls < cases
Area under the curve: 0.7796 <-- AUCROC
```

Methods/Results - K-Nearest Neighbors with K-Fold (cont.)

- **K-fold cross-validation:** tune regularization hyperparameters to avoid overfitting and improve model performance.

- Training error rate = 0.1752
- Testing error rate = 0.1847
- Accuracy = 82.07%
- Sensitivity = 0.7397
- Specificity = 0.8739
- F1 Score = 0.7660
- AUCROC = 0.7796

```
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  54  14
       yes 19  97

               Accuracy : 0.8207
                 95% CI : (0.7575, 0.8732)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 1.781e-10

                  Kappa : 0.6209

 Mcnemar's Test P-Value : 0.4862

            Sensitivity : 0.7397
            Specificity : 0.8739
         Pos Pred Value : 0.7941
         Neg Pred Value : 0.8362
              Precision : 0.7941
                 Recall : 0.7397
                     F1 : 0.7660
             Prevalence : 0.3967
         Detection Rate : 0.2935
   Detection Prevalence : 0.3696
      Balanced Accuracy : 0.8068

       'Positive' Class : no
```

Methods/Results - Random Forest

- **Random Forest** is an ensemble method used to create multiple decision trees using different random subsets of the data and features, and each decision will providing its opinion on how to classify the data. It's less prone to overfitting compared to decision trees.

- Training error rate = 0.1997
- Testing error rate = 0.2119
- Accuracy = 78.80%
- Sensitivity = 0.6712
- Specificity = 0.8649
- F1 Score = 0.7153
- AUCROC = 0.8754

```
Confusion matrix:
     no yes class.error
no  256  82   0.2426036
yes  65 333   0.1633166
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  49  15
       yes 24  96

               Accuracy : 0.788
                 95% CI : (0.7218, 0.8447)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 7.378e-08

                  Kappa : 0.5477

 Mcnemar's Test P-Value : 0.2002

            Sensitivity : 0.6712
            Specificity : 0.8649
         Pos Pred Value : 0.7656
         Neg Pred Value : 0.8000
              Precision : 0.7656
                 Recall : 0.6712
                     F1 : 0.7153
             Prevalence : 0.3967
         Detection Rate : 0.2663
   Detection Prevalence : 0.3478
      Balanced Accuracy : 0.7680
```

Methods/Results - Random Forest with K-Fold (cont.)

- **K-fold cross-validation** is used to tune the Random Forest model hyperparameters - **mtry.**
- **Mtry:** determines the number of variables to randomly sample as candidates at each split.
- The optimal value for **mtry = 2**.

- Training error rate = 0.0611
- Testing error rate = 0.1793
- Accuracy = 82.07%
- Sensitivity = 0.7397
- Specificity = 0.8739
- F1 Score = 0.7660
- AUCROC = 0.8963

```
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  54  14
       yes 19  97

               Accuracy : 0.8207
                 95% CI : (0.7575, 0.8732)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 1.781e-10

                  Kappa : 0.6209

 Mcnemar's Test P-Value : 0.4862

            Sensitivity : 0.7397
            Specificity : 0.8739
         Pos Pred Value : 0.7941
         Neg Pred Value : 0.8362
              Precision : 0.7941
                 Recall : 0.7397
                     F1 : 0.7660
             Prevalence : 0.3967
         Detection Rate : 0.2935
   Detection Prevalence : 0.3696
      Balanced Accuracy : 0.8068
```

Methods/Results - Gradient Boosting

- **Gradient Boosting** combines an ensemble of weak decision trees subsequently to create a better performance as a whole.

- Training error rate = 0.1182
- Testing error rate = 0.1956
- Accuracy = 80.43%
- Sensitivity = 0.7534
- Specificity = 0.8378
- F1 Score = 0.7534
- AUCROC = 0.8043

```
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no  55  18
       yes 18  93

               Accuracy : 0.8043
                 95% CI : (0.7396, 0.859)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 4.217e-09

                  Kappa : 0.5913

 Mcnemar's Test P-Value : 1

            Sensitivity : 0.7534
            Specificity : 0.8378
         Pos Pred Value : 0.7534
         Neg Pred Value : 0.8378
              Precision : 0.7534
                 Recall : 0.7534
                     F1 : 0.7534
             Prevalence : 0.3967
         Detection Rate : 0.2989
   Detection Prevalence : 0.3967
      Balanced Accuracy : 0.7956
```

Methods/Results - Gradient Boosting with K-Fold (cont.)

- **K-fold cross-validation** to tune the **Gradient Boosting** model hyperparameters
- The hyperparameters considered: the number of trees, interaction depth, shrinkage (learning rate), and the minimum number of observations in nodes.
- Accuracy was used to select the optimal model using the largest value.
- The final values used for the model were **n.trees** = **150**, **interaction.depth** = **5**, **shrinkage** = **0.01**, and **n.minobsinnode** = **10**.
- Training error rate = 0.5027
- Testing error rate = 0.1793
- Accuracy = 82.07%
- Sensitivity = 0.6849
- Specificity = 0.9099
- F1 Score = 0.7519
- AUCROC = 0.904

```
Confusion Matrix and Statistics

          Reference
Prediction  no yes
       no   50  10
       yes  23 101

               Accuracy : 0.8207
                 95% CI : (0.7575, 0.8732)
    No Information Rate : 0.6033
    P-Value [Acc > NIR] : 1.781e-10

                  Kappa : 0.6135

 Mcnemar's Test P-Value : 0.03671

            Sensitivity : 0.6849
            Specificity : 0.9099
         Pos Pred Value : 0.8333
         Neg Pred Value : 0.8145
              Precision : 0.8333
                 Recall : 0.6849
                     F1 : 0.7519
             Prevalence : 0.3967
```

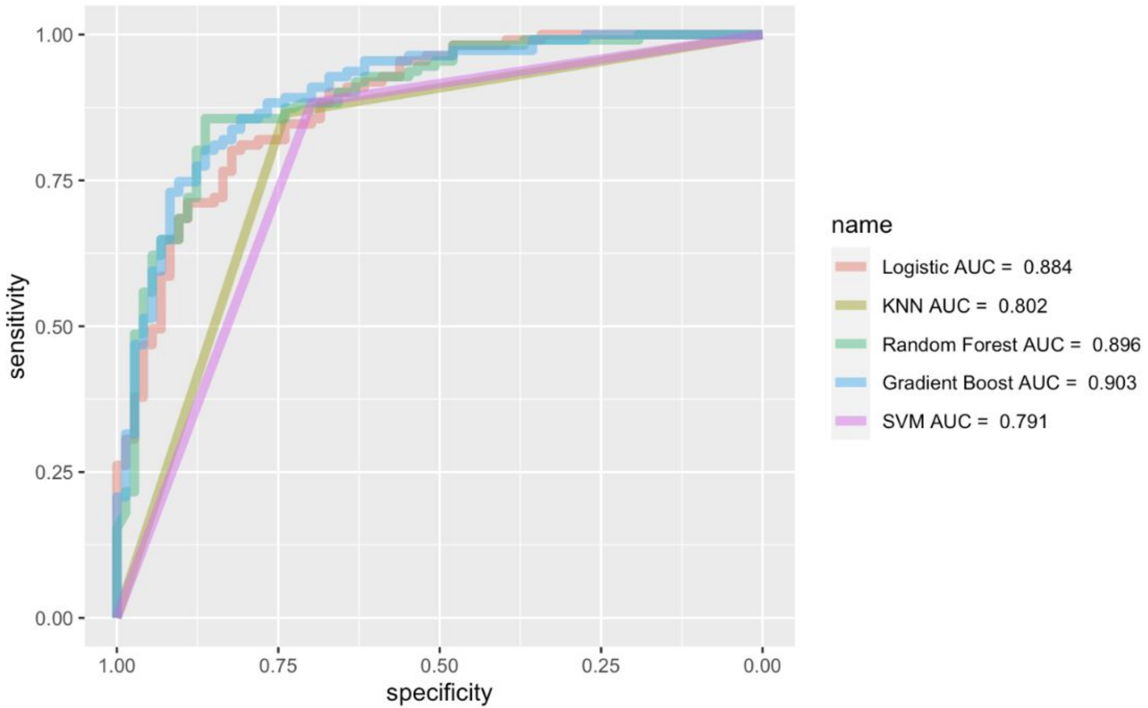
Results - Determining Final Models

Our final models are:

1. Initial logistic regression model
2. KNN tuned using K-fold
3. Random forest tuned using K-fold
4. Gradient boosting tuned using K-fold
5. SVM tuned with K-fold

Gradient boosting has the best AUCROC value = 0.903.

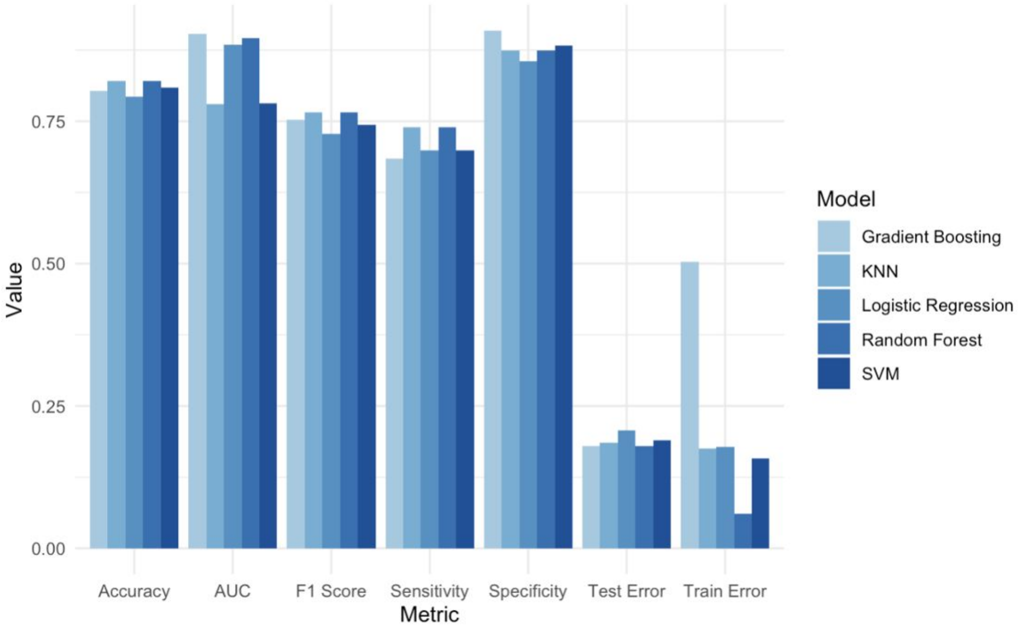
FIGURE 2. Comparison of AUCROC Curves for Final Models.



Results

- All models achieve promising performance on classification task with high accuracy above 80%.
- All models performs well on other evaluation metrics including F1 score, AUC, sensitivity, and specificity.
- The **random forest** model outperformed other methods in combination with all 7 evaluation metrics.

FIGURE 3. Comparison of Accuracy, AUC, F1 Score, Sensitivity, Specificity, Test Error, and Train Error for Final Models.



Discussion

Recommended Algorithm(s):

- **Random Forest**

  Random forest is an ensemble method and can handle overfitting problem effectively in a small dataset.

Future Analysis:

- Incorporate deep learning algorithms such as neural network which is capable of automatic feature selection to better capture non-linear relationships.
- Find larger dataset for model training.

Thanks For Watching.
Any Questions?