DESCRIPTIVE & PREDICTIVE

ANALYTICS IN HEALTHCARE

Ayesha N. Quadri

Pranav Dixit

Mariam Ansari

HIT-215 Advanced Data Analytics

Dr. Steven C. Lindo

Hofstra University

INTRODUCTION:

Machine learning (ML) is the subset of AI involving algorithms that, unlike expert rules, can define their own rules from input data through iterative training and improvement, without explicit human programming. ML was first applied in medicine in the 1980s and 1990s in the form of computer-assisted diagnosis systems in medical imaging. While expert rules are widely used in clinical medicine today, including most clinical decision support systems and calculators, they are often not included in contemporary discussions of AI, which focus primarily on machine learning and, more recently, deep learning.

Broadly, machine learning encompasses supervised, unsupervised and reinforcement learning. Supervised learning algorithms are trained using labelled data (e.g. histological specimens that have already been labelled as normal or diseased by a human expert). When applied to unlabeled categorical or continuous test data, these trained algorithms predict outcomes by classification or regression, respectively. Conversely, unsupervised ML algorithms are trained on unlabeled data by data-driven (rather than human-guided) processes. Such models are used for data clustering, feature extraction and dimensionality reduction (e.g. identifying patient subgroups based on unlabeled clinical data). Finally, reinforcement learning is an environment-driven approach, where iterative learning cycles result in a reward or penalty by comparison with a pre-defined target (e.g. continuous blood glucose monitoring and insulin administration). Clinically-relevant AI models often employ more than one of these approaches, and each approach can be further subclassified into many different algorithm types, the details of which are beyond the scope of this project.

Deep learning (DL) refers to an increasingly-popular branch of ML employing artificial neural networks (ANN) with multiple processing layers, which may employ supervised, unsupervised and reinforcement ML approaches. DL excels particularly in complex tasks involving high-volume and high-dimensional data. However, it is computationally more demanding than traditional ML approaches, and, as many of its processing layers remain hidden from the human user (giving rise to the so-called “black box” of AI). However, as anticipated by Moore’s law, computing speed, memory, compactness, cost, and algorithmic capabilities have improved exponentially, as has the availability of digital clinical data. This rapid rate of development makes accurate forecasting about the future role of AI challenging.

Consequently, it becomes imperative to understand the Algorithm Reporting Metrics. Algorithm reporting metrics are crucial for evaluating the performance of artificial intelligence algorithms. These metrics provide a standardized way to assess and compare the effectiveness of different algorithms. Below is a description and calculation method for each metric highly relevant to the applications of health data and machine learning algorithms:

- **Sensitivity (Recall, True Positive Rate - TPR)**: Measures the proportion of actual positives correctly identified.

  Sensitivity=TP+FNTP

- **Specificity (True Negative Rate - TNR)**: Measures the proportion of actual negatives correctly identified.

  Specificity=TN+FPTN

- **Precision (Positive Predictive Value - PPV)**: Measures the proportion of positive identifications that were actually correct.

  Precision=TP+FPTP

- **Accuracy**: Measures the proportion of all predictions that were correct.

  Accuracy=TP+TN+FP+FNTP+TN

- **F1-score**: Harmonic mean of precision and sensitivity, providing a balance between the two.

  F1-score=2×Sensitivity+PrecisionSensitivity×Precision

- **Matthews Correlation Coefficient (MCC)**: Considers all four binary classification outcomes (TP, TN, FP, FN).

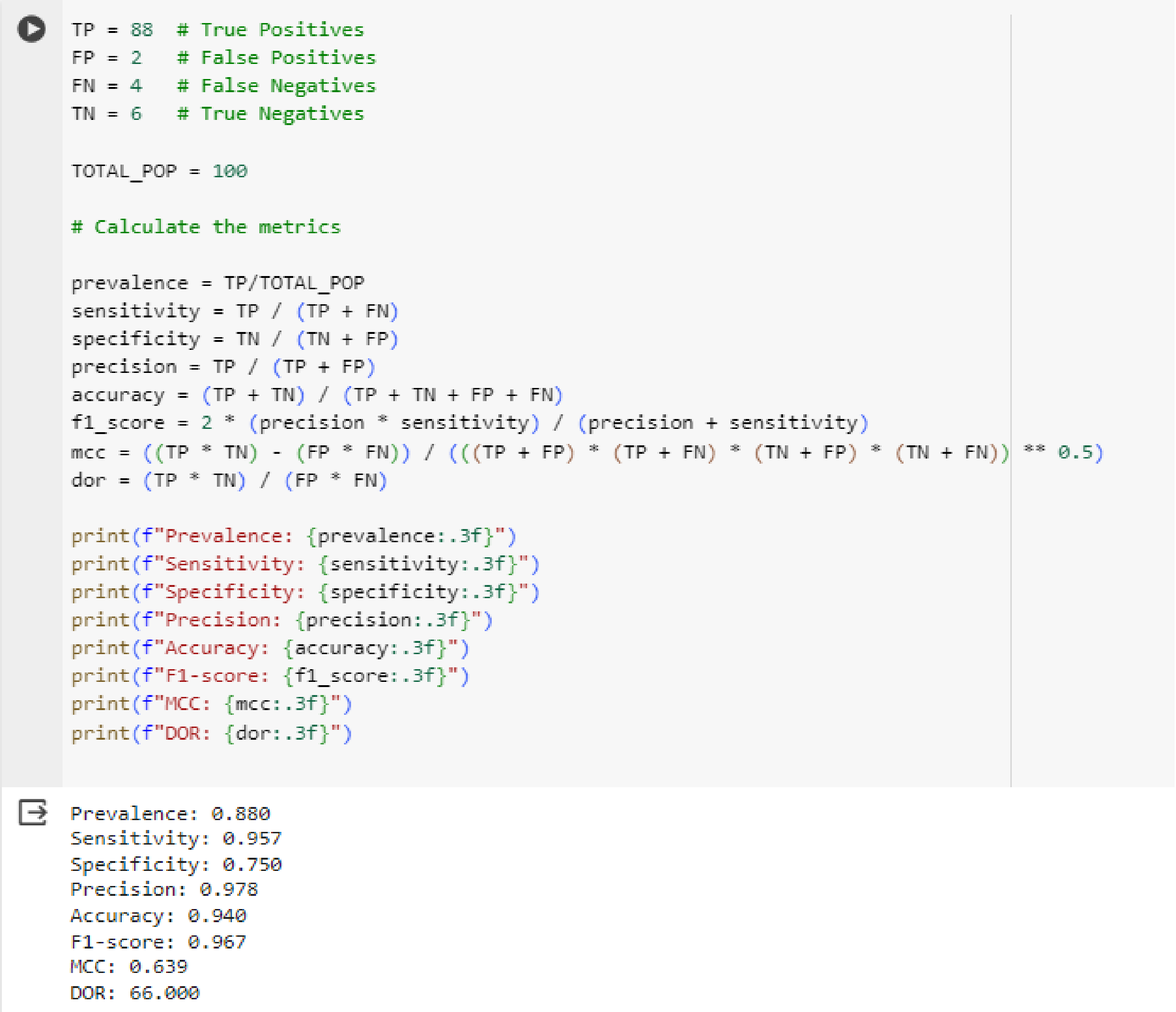
  $$MCC=(TP+FP)\times(TP+FN)\times(TN+FP)\times(TN+FN)(TP\times TN)-(FP\times FN)$$

- **Diagnostic Odds Ratio (DOR)**: Odds ratio of positive labeling for condition positive versus negative.

  $$DOR=FP\times FNTP\times TN$$

- **Area Under Receiver Operating Curve (AUROC)**: Represents the degree or measure of separability between classes. Various approaches exist, relying on the integration of the ROC curve.

B. Hypothetical Dataset Analysis



```
TP = 88  # True Positives
FP = 2   # False Positives
FN = 4   # False Negatives
TN = 6   # True Negatives

TOTAL_POP = 100

# Calculate the metrics

prevalence = TP/TOTAL_POP
sensitivity = TP / (TP + FN)
specificity = TN / (TN + FP)
precision = TP / (TP + FP)
accuracy = (TP + TN) / (TP + TN + FP + FN)
f1_score = 2 * (precision * sensitivity) / (precision + sensitivity)
mcc = ((TP * TN) - (FP * FN)) / (((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)) ** 0.5)
dor = (TP * TN) / (FP * FN)

print(f"Prevalence: {prevalence:.3f}")
print(f"Sensitivity: {sensitivity:.3f}")
print(f"Specificity: {specificity:.3f}")
print(f"Precision: {precision:.3f}")
print(f"Accuracy: {accuracy:.3f}")
print(f"F1-score: {f1_score:.3f}")
print(f"MCC: {mcc:.3f}")
print(f"DOR: {dor:.3f}")
```

```
Prevalence: 0.880
Sensitivity: 0.957
Specificity: 0.750
Precision: 0.978
Accuracy: 0.940
F1-score: 0.967
MCC: 0.639
DOR: 66.000
```

KPI 1 - LOS for sepsis:

The definition of sepsis has been revised over the past decades to detect its onset early and stratify disease severity. Previous diagnostic criteria for sepsis and septic shock were inconvenient and complex, leading to delays in recognizing sepsis. Recently, research has supported the early recognition of sepsis, resulting in reduced sepsis-related mortality . In this context, efforts have been made for early recognition of sepsis and septic shock. The third international consensus definition for sepsis and septic shock developed a simplified criteria for sepsis-related organ dysfunction and quick sequential organ failure assessment (qSOFA).

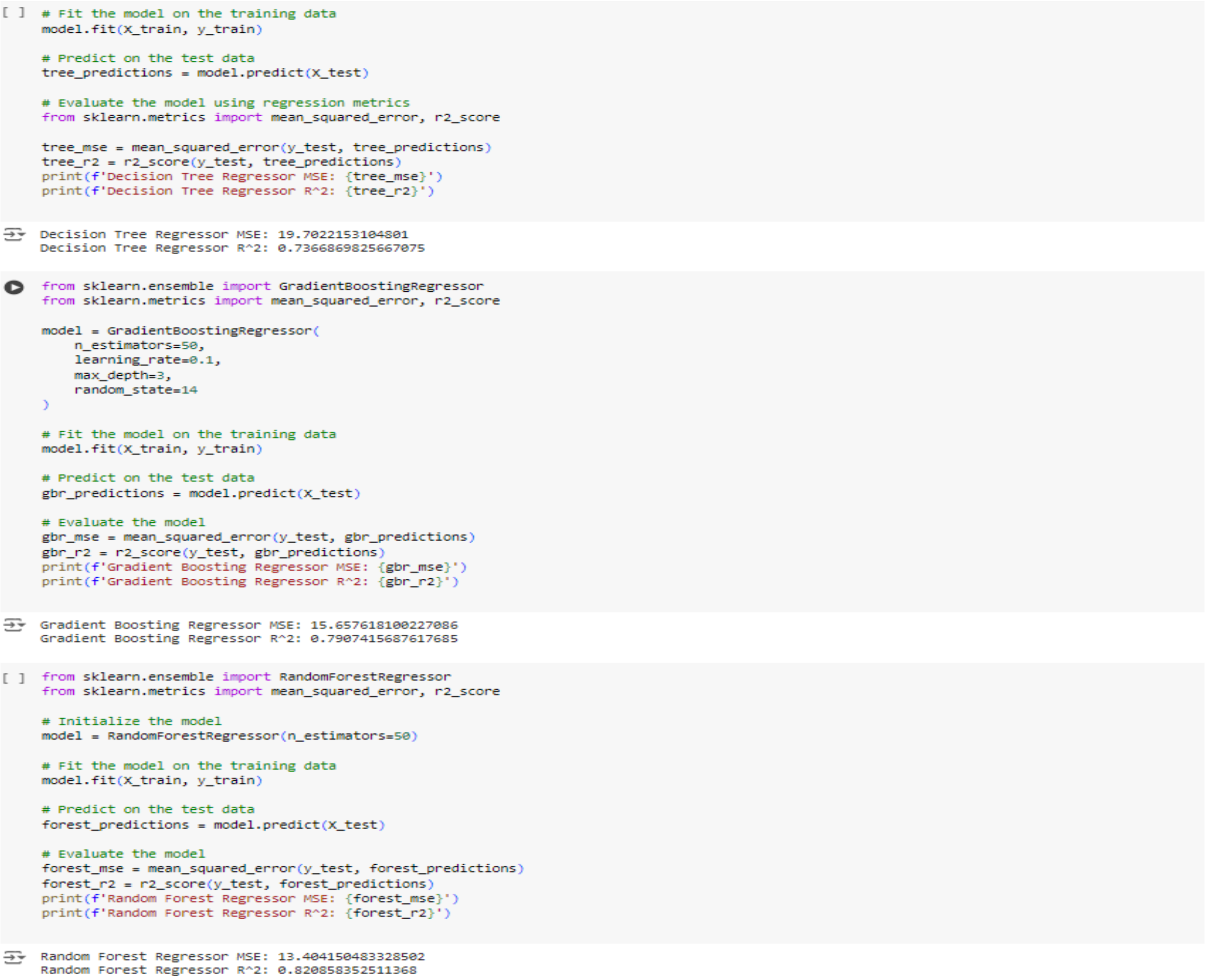
Clinical data were retrospectively collected from the MIMIC III Database. The collected EMRs consisted of 8 vital signs, 13 laboratory data points, and three demographic information items from the collected electronic medical record (EMR) records as variables were used to develop the model.

Gender, Age, Number of oxygen delivery types in admissions, Non-invasive ventilation, Invasive ventilation, Heart rate (/min), Diastolic blood pressure , Systolic blood pressure , Mean blood pressure (mm Hg),Respiratory rate (/min), Body temperature (°C), SpO2 (%), Total GCS, Lactate (mmol/L), Bilirubin (mg/dL), Platelets , Creatinine , WBC (103/μL), pH , HCO3− (mmol/L), BUN (mg/dL),Albumin (g/dL), Glucose (mg/dL), INR, Lymphocyte , ANC.

Following data collection, data cleaning and feature selection was performed using pivot tables indexed by Subject ID the above features from the ‘label’ column were extracted and their corresponding values created the subsequent columns having “Length Of Stay” as the continuous data type for evaluating the prediction performance of various Regressor models

The core of the analysis involves employing well-established machine learning models for regression. We explore three prominent models: Decision Tree Regressor, Gradient Boosting Regressor, and Support Vector Regressor (SVR). Additionally, a Random Forest Regressor is included to leverage the power of ensemble learning techniques.

Model evaluation is conducted using Mean Squared Error (MSE) and R-squared (R^2). MSE measures the average squared difference between predicted and actual LOS values, with lower MSE indicating better performance. R^2, on the other hand, reflects the proportion of variance in the target variable (LOS) explained by the model. A higher R^2 suggests a stronger relationship between the features and LOS.



```
[ ] # Fit the model on the training data
    model.fit(X_train, y_train)

    # Predict on the test data
    tree_predictions = model.predict(X_test)

    # Evaluate the model using regression metrics
    from sklearn.metrics import mean_squared_error, r2_score

    tree_mse = mean_squared_error(y_test, tree_predictions)
    tree_r2 = r2_score(y_test, tree_predictions)
    print(f'Decision Tree Regressor MSE: {tree_mse}')
    print(f'Decision Tree Regressor R^2: {tree_r2}')
```

```
Decision Tree Regressor MSE: 19.7022153104801
Decision Tree Regressor R^2: 0.7366869825667075
```

```
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.metrics import mean_squared_error, r2_score

model = GradientBoostingRegressor(
    n_estimators=50,
    learning_rate=0.1,
    max_depth=3,
    random_state=14
)

# Fit the model on the training data
model.fit(X_train, y_train)

# Predict on the test data
gbr_predictions = model.predict(X_test)

# Evaluate the model
gbr_mse = mean_squared_error(y_test, gbr_predictions)
gbr_r2 = r2_score(y_test, gbr_predictions)
print(f'Gradient Boosting Regressor MSE: {gbr_mse}')
print(f'Gradient Boosting Regressor R^2: {gbr_r2}')
```

```
Gradient Boosting Regressor MSE: 15.657618100227086
Gradient Boosting Regressor R^2: 0.7907415687617685
```

```
[ ] from sklearn.ensemble import RandomForestRegressor
    from sklearn.metrics import mean_squared_error, r2_score

    # Initialize the model
    model = RandomForestRegressor(n_estimators=50)

    # Fit the model on the training data
    model.fit(X_train, y_train)

    # Predict on the test data
    forest_predictions = model.predict(X_test)

    # Evaluate the model
    forest_mse = mean_squared_error(y_test, forest_predictions)
    forest_r2 = r2_score(y_test, forest_predictions)
    print(f'Random Forest Regressor MSE: {forest_mse}')
    print(f'Random Forest Regressor R^2: {forest_r2}')
```

```
Random Forest Regressor MSE: 13.404150483328502
Random Forest Regressor R^2: 0.820858352511368
```

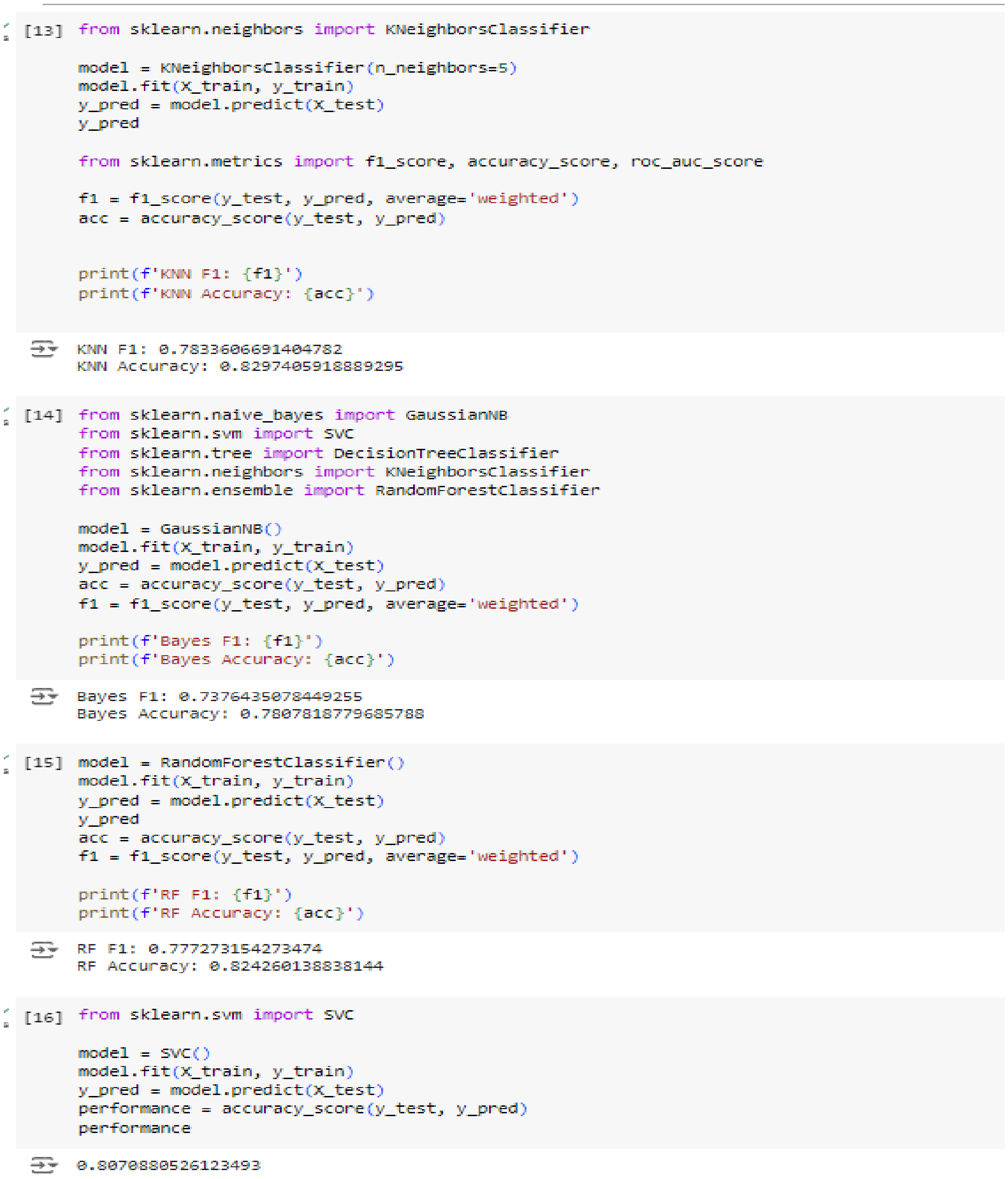
KPI 2–SEPS Decision Support

While the previous study focused on predicting Length of Stay (LOS) for diagnosed sepsis patients (KPI 1), this section explores the application of machine learning for sepsis classification itself. **C** Unlike LOS prediction (regression), early sepsis diagnosis requires classification models. Algorithms like Logistic Regression, Support Vector Machines (SVM), or Random Forest Classifiers are well-suited for this task. These models can learn to identify patterns in EMR data that differentiate septic from non-septic patients.

Clinical Rule Integration: Incorporating existing clinical rules or scoring systems like qSOFA (quick Sequential Organ Failure Assessment) into the machine learning models could further improve diagnostic accuracy. qSOFA assigns points based on specific physiological abnormalities, providing a readily available sepsis risk assessment tool. By combining machine learning with established clinical knowledge, we can potentially create more robust and reliable diagnostic models.

This study involved more features than the LOS support system by incorporating additional features such as **Number of oxygen delivery types in admissions, Non-invasive ventilation, Invasive ventilation, Heart rate (/min), Diastolic blood pressure , Systolic blood pressure , Mean blood pressure (mm Hg),Respiratory rate (/min), Body temperature (°C), SpO2 has added features to train the model.**

Data cleaning & preprocessing along with Feature engineering methods performed in the previous step were performed. Classifier Models were used and evaluated as below.,



```
[13] from sklearn.neighbors import KNeighborsClassifier

     model = KNeighborsClassifier(n_neighbors=5)
     model.fit(X_train, y_train)
     y_pred = model.predict(X_test)
     y_pred

     from sklearn.metrics import f1_score, accuracy_score, roc_auc_score

     f1 = f1_score(y_test, y_pred, average='weighted')
     acc = accuracy_score(y_test, y_pred)


     print(f'KNN F1: {f1}')
     print(f'KNN Accuracy: {acc}')
```

```
KNN F1: 0.7833606691404782
KNN Accuracy: 0.8297405918889295
```

```
[14] from sklearn.naive_bayes import GaussianNB
     from sklearn.svm import SVC
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.neighbors import KNeighborsClassifier
     from sklearn.ensemble import RandomForestClassifier

     model = GaussianNB()
     model.fit(X_train, y_train)
     y_pred = model.predict(X_test)
     acc = accuracy_score(y_test, y_pred)
     f1 = f1_score(y_test, y_pred, average='weighted')

     print(f'Bayes F1: {f1}')
     print(f'Bayes Accuracy: {acc}')
```

```
Bayes F1: 0.7376435078449255
Bayes Accuracy: 0.7807818779685788
```

```
[15] model = RandomForestClassifier()
     model.fit(X_train, y_train)
     y_pred = model.predict(X_test)
     y_pred
     acc = accuracy_score(y_test, y_pred)
     f1 = f1_score(y_test, y_pred, average='weighted')

     print(f'RF F1: {f1}')
     print(f'RF Accuracy: {acc}')
```

```
RF F1: 0.777273154273474
RF Accuracy: 0.824260138838144
```

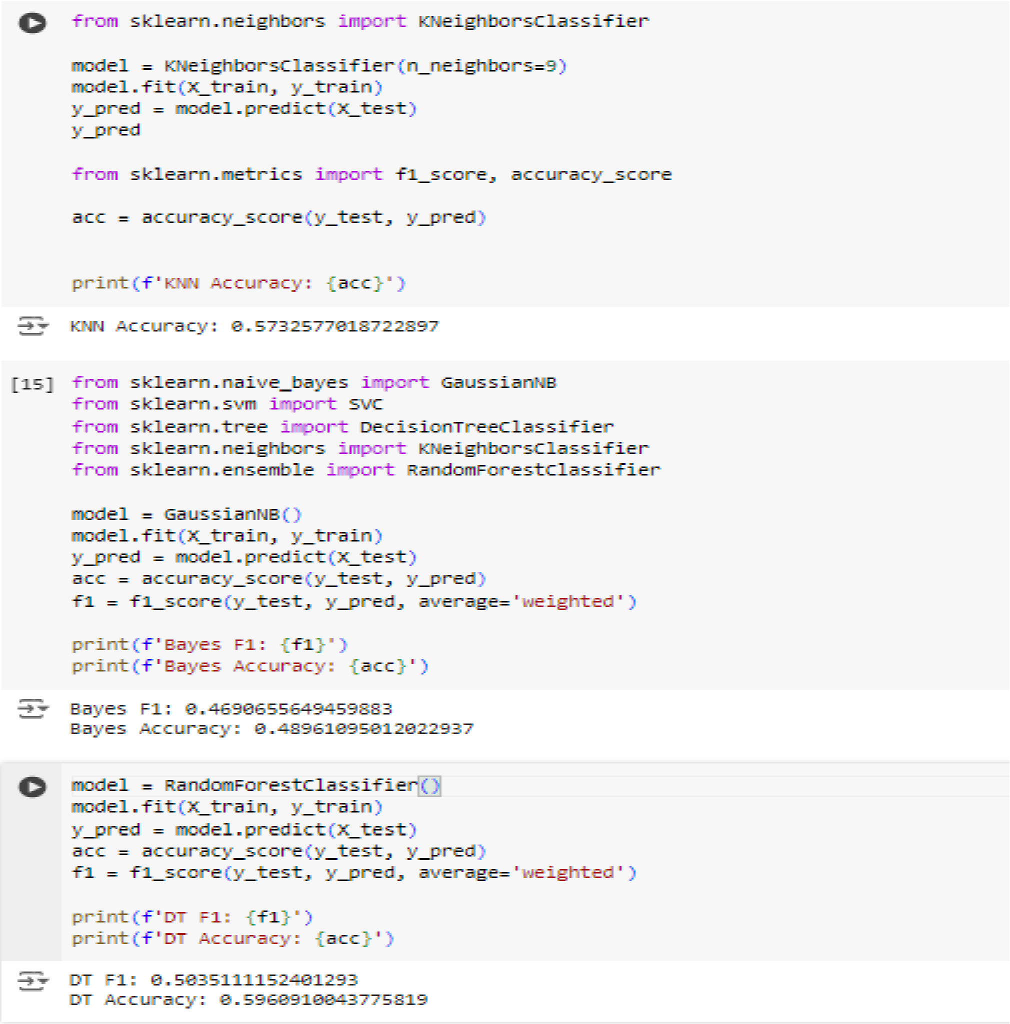
```
[16] from sklearn.svm import SVC

     model = SVC()
     model.fit(X_train, y_train)
     y_pred = model.predict(X_test)
     performance = accuracy_score(y_test, y_pred)
     performance
```

```
0.8070880526123493
```

KPI 3 - LAB Result Decision Support System

This KPI aims to identify the top 150 highly corelated features out of 375 individual predictors of health by retrieving the labevents and chartevents and collectively identify top 150 parameters essential for 95 diagnoses from the MIMIC III dataset.

Data was preprocessed and the top 150 features were extracted using the K Best features module of the sci-kit module. Classification Models such as KNN, Bayes & Random Forest Classier were trained.



```
from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier(n_neighbors=9)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
y_pred

from sklearn.metrics import f1_score, accuracy_score

acc = accuracy_score(y_test, y_pred)


print(f'KNN Accuracy: {acc}')
```

```
KNN Accuracy: 0.5732577018722897
```

```
[15] from sklearn.naive_bayes import GaussianNB
     from sklearn.svm import SVC
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.neighbors import KNeighborsClassifier
     from sklearn.ensemble import RandomForestClassifier

     model = GaussianNB()
     model.fit(X_train, y_train)
     y_pred = model.predict(X_test)
     acc = accuracy_score(y_test, y_pred)
     f1 = f1_score(y_test, y_pred, average='weighted')

     print(f'Bayes F1: {f1}')
     print(f'Bayes Accuracy: {acc}')
```

```
Bayes F1: 0.4690655649459883
Bayes Accuracy: 0.48961095012022937
```

```
model = RandomForestClassifier()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
acc = accuracy_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred, average='weighted')

print(f'DT F1: {f1}')
print(f'DT Accuracy: {acc}')
```

```
DT F1: 0.5035111152401293
DT Accuracy: 0.5960910043775819
```

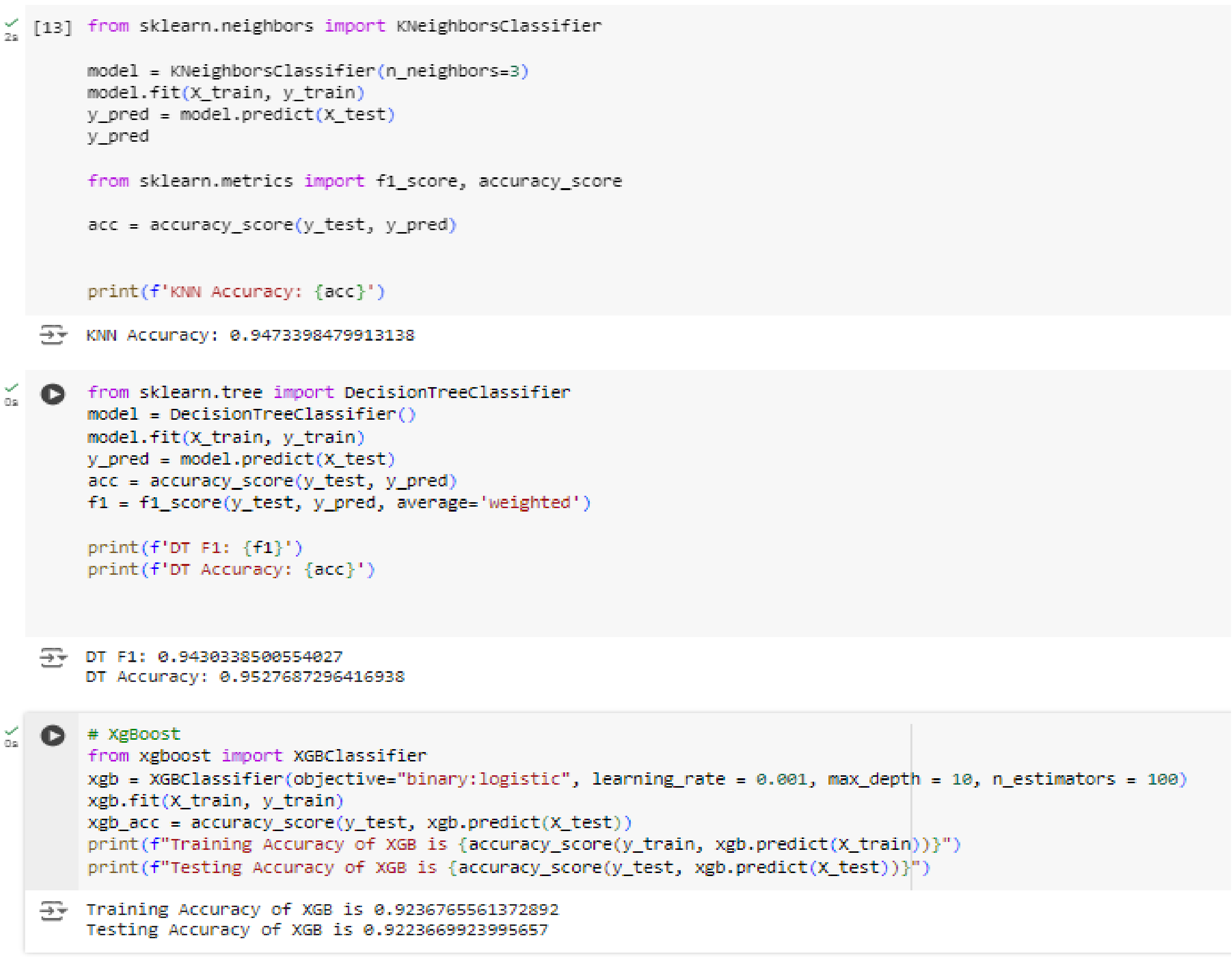
KPI 4 - Personalized diagnosis in suspected myocardial infarction

Inflammation has been identified as a critical underlying mechanism in the development and progression of HF. Research has shown that NIHF is associated with a unique and persistent inflammatory response that differs from the acute myocardial ischemia and subsequent reperfusion injury seen in ischemic heart failure. These differences have significant implications for disease pathogenesis and treatment strategies. Inflammatory biomarkers, such as high-sensitivity C-reactive protein (hsCRP), fibrinogen (FIB), albumin, erythrocyte sedimentation rate (ESR), and indices within the complete blood cell count (including white blood cells [WBC], neutrophils, lymphocytes, and red blood cell distribution width [RDW]), have been identified as potential predictors of adverse outcomes in patients with CAD or HF. It is worth noting that, despite biomarkers like FIB, albumin and RDW were not traditionally used as inflammatory biomarkers; several studies have shown that these biomarkers can reflect chronic inflammation, and inflammatory activation may be the central link in the prognostic role of these biomarkers . Furthermore, derived parameters from these biomarkers, such as the fibrinogen-to-albumin ratio (FAR), hsCRP-to-albumin (CAR), neutrophil-to-lymphocyte ratio (NLR), systemic immune-inflammation index (SII), and prognostic nutritional index (PNI), have also been demonstrated to serve as prognostic factors .

We investigated 11 single parameters albumin, C-Reactive Protein, BTproBNP , troponin I , troponin N, lymphocytes, fibrinogen, neutrophils, platelet count, Red Cell Distribution Width to predict the diagnosis of the dataset using various ML models.

To enhance the diagnostic precision for myocardial infarction, we applied machine learning algorithms that incorporate these inflammatory combinations and interactions between these biomarkers in our dataset.

The findings from our analysis indicated that models integrating multiple biomarkers, especially those combining traditional and derived parameters, significantly improve the identification and stratification of risk in patients suspected of myocardial infarction. This approach allows for more personalized treatment plans, targeting specific inflammatory profiles and potentially improving outcomes in cardiac care. By adopting these advanced diagnostic tools, clinicians can better manage and treat patients with suspected myocardial conditions, aligning treatments more closely with individual patient needs.



```
[13] from sklearn.neighbors import KNeighborsClassifier

     model = KNeighborsClassifier(n_neighbors=3)
     model.fit(X_train, y_train)
     y_pred = model.predict(X_test)
     y_pred

     from sklearn.metrics import f1_score, accuracy_score

     acc = accuracy_score(y_test, y_pred)


     print(f'KNN Accuracy: {acc}')
```

```
KNN Accuracy: 0.9473398479913138
```

```
from sklearn.tree import DecisionTreeClassifier
model = DecisionTreeClassifier()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
acc = accuracy_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred, average='weighted')

print(f'DT F1: {f1}')
print(f'DT Accuracy: {acc}')
```

```
DT F1: 0.9430338500554027
DT Accuracy: 0.9527687296416938
```

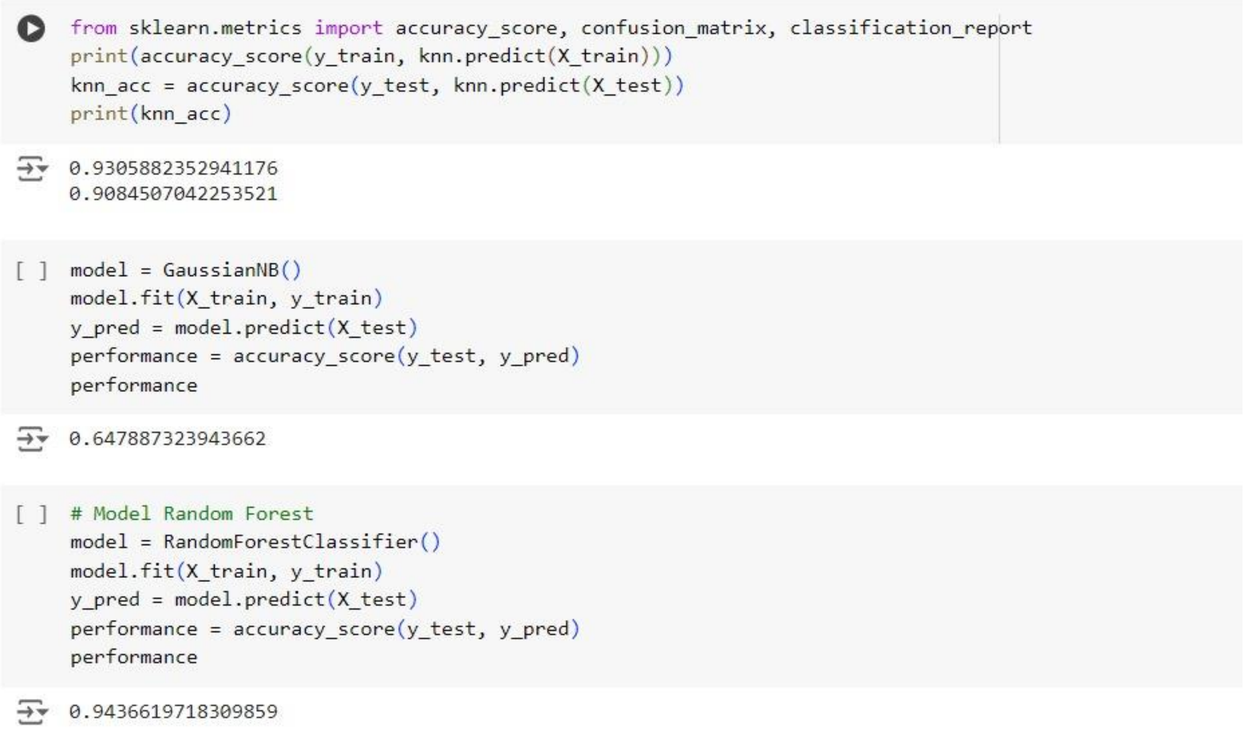
```
# XgBoost
from xgboost import XGBClassifier
xgb = XGBClassifier(objective="binary:logistic", learning_rate = 0.001, max_depth = 10, n_estimators = 100)
xgb.fit(X_train, y_train)
xgb_acc = accuracy_score(y_test, xgb.predict(X_test))
print(f"Training Accuracy of XGB is {accuracy_score(y_train, xgb.predict(X_train))}")
print(f"Testing Accuracy of XGB is {accuracy_score(y_test, xgb.predict(X_test))}")
```

```
Training Accuracy of XGB is 0.9236765561372892
Testing Accuracy of XGB is 0.9223669923995657
```

KPI 5 – Fetal Mortality Prediction:

In this part of our study, we aimed to predict risks during pregnancy that could lead to fetal mortality. We used detailed health data from various prenatal studies, which included information like the mother's age, her health history, pregnancy complications, and fetal measurements. We chose advanced machine learning tools like random forests and gradient boosting machines for this task

alarms



```
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
print(accuracy_score(y_train, knn.predict(X_train)))
knn_acc = accuracy_score(y_test, knn.predict(X_test))
print(knn_acc)
```

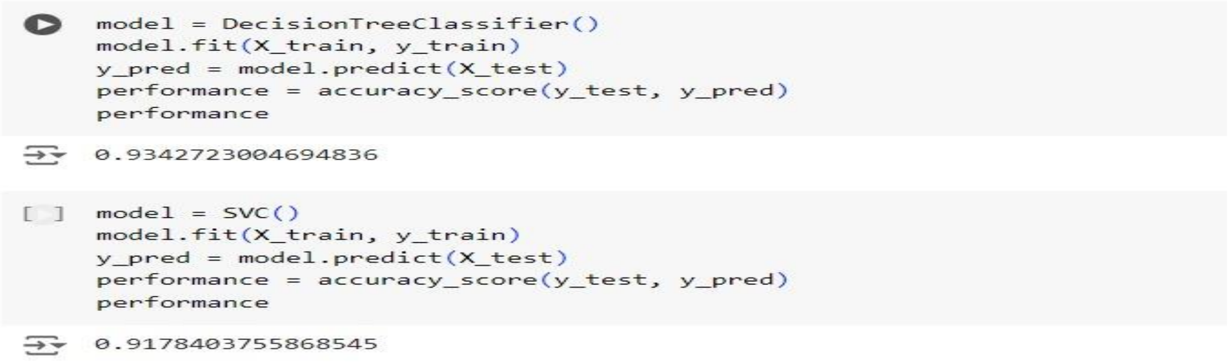
```
0.9305882352941176
0.9084507042253521
```

```
[ ] model = GaussianNB()
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    performance = accuracy_score(y_test, y_pred)
    performance
```

```
0.647887323943662
```

```
[ ] # Model Random Forest
    model = RandomForestClassifier()
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    performance = accuracy_score(y_test, y_pred)
    performance
```

```
0.9436619718309859
```



```
model = DecisionTreeClassifier()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
performance = accuracy_score(y_test, y_pred)
performance
```

```
0.9342723004694836
```

```
[ ] model = SVC()
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    performance = accuracy_score(y_test, y_pred)
    performance
```

```
0.9178403755868545
```

Limitations of the Study:

Our study has various limitations of which we find evidence in the study for KPI. Very low prevalence rates for the individual diagnosis data of 129 patients had 95 distinct diagnosis groups which could be a contributing factor for the third KPI having lower than acceptable accuracy standards. The integration of DICOM and Other radiology data to the above lab results could possibly enhance evaluation metrics. ds

Discussion:

In our discussion, it's clear that while machine learning can really help improve how we diagnose diseases and predict health risks, as we saw in our study. But bringing these tools into hospitals involves solving issues like making sure the data is good, making the tools easy to understand, and fitting them into how doctors work.

Conclusion:

In conclusion, this study demonstrates the potent capabilities of machine learning models in enhancing diagnostic precision and predicting clinical outcomes across diverse medical domains. Our use of various models like random forests, gradient boosting, and logistic regression has opened new possibilities for detecting and managing health issues more effectively. But, the success of these tools largely depends on the quality of the data we use, how well we can understand and explain what the models are doing, and making sure they work well in real-world settings. Looking ahead, we need to focus on getting better data, making our models more transparent, and testing them thoroughly. This effort is crucial if we're going to make these technologies a regular part of healthcare, where they can truly make a difference in patient care.

References:

1. Theodosiou, A. A., & Read, R. C. (2023). Artificial intelligence, machine learning, and deep learning: Potential resources for the infection clinician. *Journal of Infection, 87*(4), 287-294. https://doi.org/10.1016/j.jinf.2023.07.006
2. Singer, M., Deutschman, C. S., Seymour, C. W., Shankar-Hari, M., Annane, D., Bauer, M., Bellomo, R., Bernard, G. R., Chiche, J.-D., Coopersmith, C. M., et al. (2016). The third international consensus definitions for sepsis and septic shock (Sepsis-3). *JAMA, 315*, 801–810. https://doi.org/10.1001/jama.2016.0287
3. Levy, M. M., Fink, M. P., Marshall, J. C., Abraham, E., Angus, D., Cook, D., Cohen, J., Opal, S. M., Vincent, J.-L., & Ramsay, G. (2003). 2001 SCCM/ESICM/ACCP/ATS/SIS international sepsis definitions conference. *Intensive Care Medicine, 29*, 530–538. https://doi.org/10.1007/s00134-003-1662-x
4. Vincent, J. L. (2016). The clinical challenge of sepsis identification and monitoring. *PLoS Medicine, 13*, e1002022. https://doi.org/10.1371/journal.pmed.1002022
5. Fay, K., Sapiano, M. R. P., Gokhale, R., Dantes, R., Thompson, N., Katz, D. E., Ray, S. M., Wilson, L. E., Perlmutter, R., Nadle, J., et al. (2020). Assessment of health care exposures and outcomes in adult patients with sepsis and septic shock. *JAMA Network Open, 3*, e206004. https://doi.org/10.1001/jamanetworkopen.2020.6004
6. Jones, S. L., Ashton, C. M., Kiehne, L., Gigliotti, E., Bell-Gordon, C., Disbot, M., Masud, F., Shirkey, B. A., & Wray, N. P. (2015). Reductions in sepsis mortality and costs after design and implementation of a nurse-based early recognition and response program. *The*

Joint Commission Journal on Quality and Patient Safety, 41, 483–491. https://doi.org/10.1016/S1553-7250(15)41063-3

7. Font, M. D., Thyagarajan, B., & Khanna, A. K. (2020). Sepsis and septic shock–basics of diagnosis, pathophysiology, and clinical decision making. *Medical Clinics of North America, 104*, 573–585. https://doi.org/10.1016/j.mcna.2020.02.011
8. Reinhart, K., Daniels, R., Kissoon, N., Machado, F. R., Schachter, R. D., & Finfer, S. (2017). Recognizing sepsis as a global health priority–A WHO resolution. *New England Journal of Medicine, 377*, 414–417. https://doi.org/10.1056/NEJMp1707170
9. American College of Chest Physicians/Society of Critical Care Medicine. (1992). Definitions for sepsis and organ failure and guidelines for the use of innovative therapies in sepsis. *Critical Care Medicine, 20*, 864–874. https://doi.org/10.1097/00003246-199206000-00025
10. Jones, S. L., Ashton, C. M., Kiehne, L., Gigliotti, E., Bell-Gordon, C., Pinn, T. T., Tran, S. K., Nicolas, J. C., Rose, A. L., Shirkey, B. A., et al. (2016). The sepsis early recognition and response initiative (SERRI). *The Joint Commission Journal on Quality and Patient Safety, 42*, 122–131. https://doi.org/10.1016/S1553-7250(16)42015-5
11. Kim, H. I., & Park, S. (2019). Sepsis: Early recognition and optimized treatment. *Tuberculosis and Respiratory Diseases, 82*, 6–14. https://doi.org/10.4046/trd.2018.0041
12. Kim, T., Tae, Y., Yeo, H. J., Jang, J. H., Cho, K., Yoo, D., Lee, Y., Ahn, S. H., Kim, Y., Lee, N., & Cho, W. H. (2023). Development and validation of a deep-learning-based sepsis and septic shock early prediction system (DeepSEPS) using real-world ICU data. *Journal of Clinical Medicine, 12*(22), 7156. https://doi.org/10.3390/jcm12227156

13. Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv:1412.3555. Retrieved from https://arxiv.org/abs/1412.3555
14. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research, 15*, 1929–1958. Retrieved from http://jmlr.org/papers/v15/srivastava14a.html
15. Zhang, B., Jiang, X., Yang, J., Huang, J., Hu, C., Hong, Y., Ni, H., & Zhang, Z. (2024). Application of artificial intelligence in the management of patients with renal dysfunction. *Renal Failure, 46*(1), Article 2337289. https://doi.org/10.1080/0886022X.2024.2337289
16. Zhang, Z., Ho, K. M., & Hong, Y. (2019). Machine learning for the prediction of volume responsiveness in patients with oliguric acute kidney injury in critical care. *Critical Care, 23*(1), Article 112. https://doi.org/10.1186/s13054-019-2411-z
17. Mandala, S., Rizal, A., Adiwijaya, Nurmaini, S., Suci Amini, S., Almayda Sudarisman, G., Wen Hau, Y., & Hanan Abdullah, A. (2024). An improved method to detect arrhythmia using an ensemble learning-based model in multi-lead electrocardiogram (ECG). *PLOS ONE, 19*(4), e0297551. https://doi.org/10.1371/journal.pone.0297551