Predicting Sepsis Survival Using Clinical Data

- **Zhaocheng Yang**
- Instructor: Andras Zsom
- TA: Mingjun Ma
- Brown University
- Data Science Institute

- 12/11/2024
- GitHub

Introduction

What is Sepsis?	A life-threatening condition triggered by an extreme immune response to infection.
Importance	1. High mortality rate 2. Early prediction is critical for timely intervention and treatment
Objective	Build a classification model to predict patient survival using clinical features.
Data Source:	1. UC Irvine Machine Learning Repository 2. Dataset from Norwegian hospital admissions (2011-2012) of patients with sepsis-related diagnoses.

EDA Recap

Number of rows: 110,341

Number of columns: 4

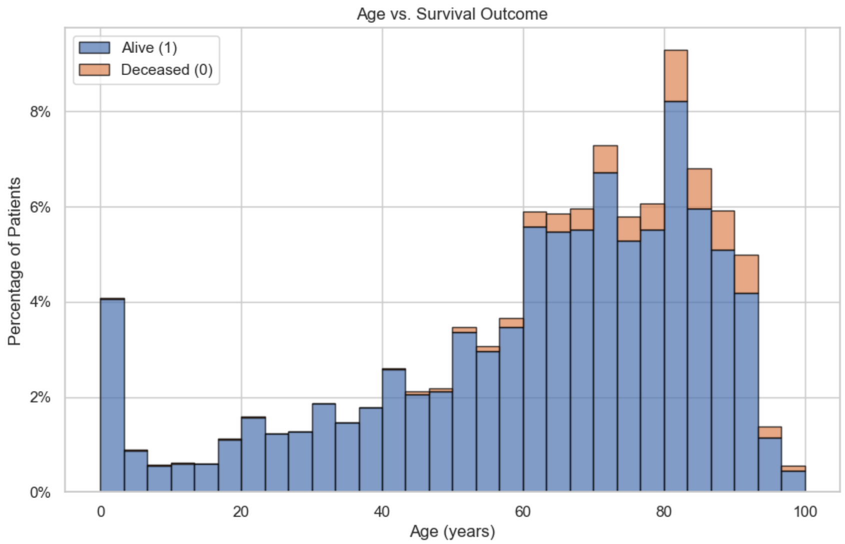
Missing Values: NO

Age: Age of the patient in years.

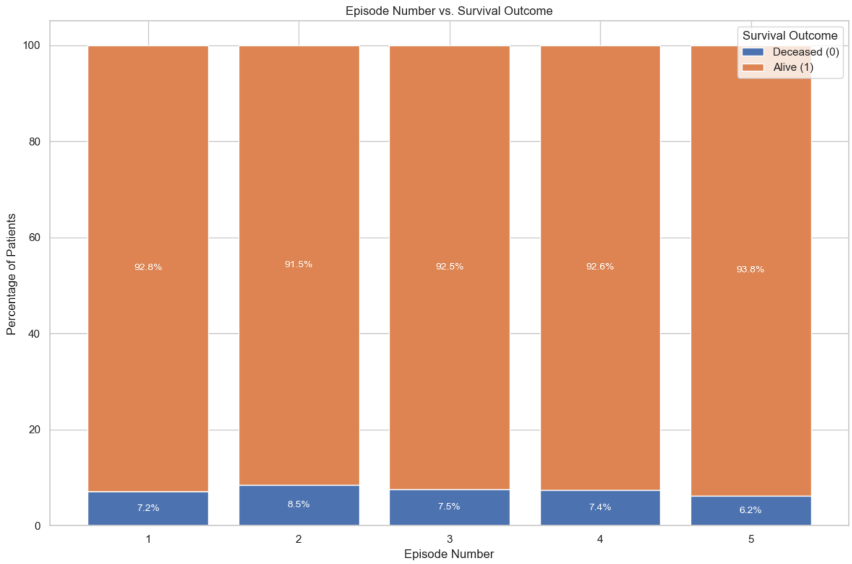
Sex: Gender of the patient. (0: male, 1: female)

Episode Number: Number of prior Sepsis episodes [1, 2, 3, 4, 5]

Hospital_Outcome: Status of the patient after 9,351 days of being admitted to the hospital. (0: Decreased, 1: Alive)



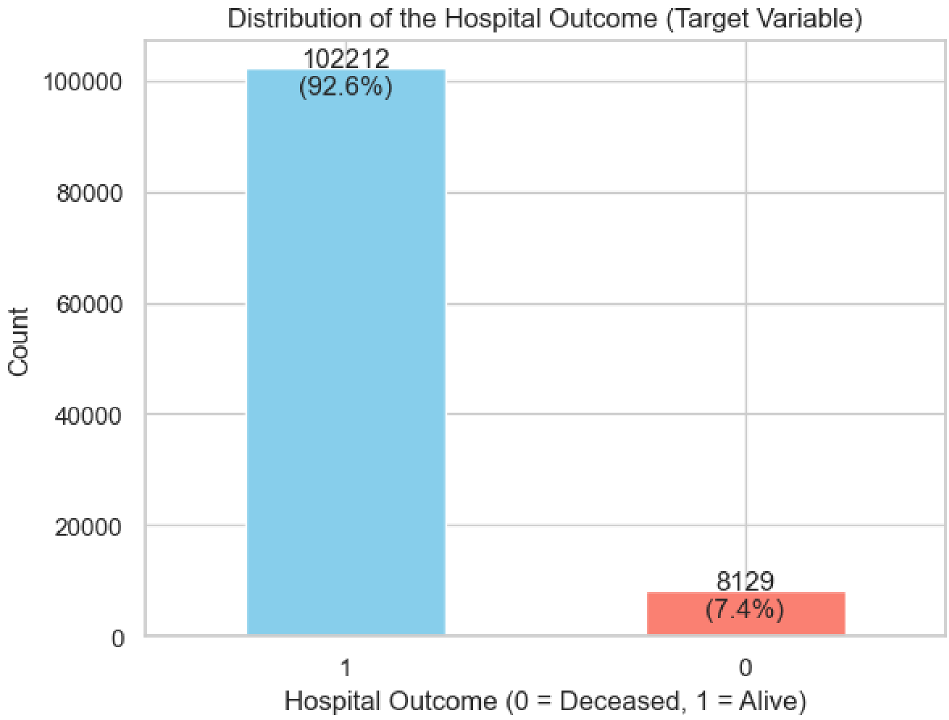
Younger patients have higher survival rates.



The mortality does not always increase with more episodes.

Data Splitting

StratifiedKFold(n_splits=4)



After the Splitting:

```
Train size: 66204 (60.00%)
Validation size: 22068 (20.00%)
Test size: 22069 (20.00%)
```

	Train	Validation	Test
1	92.6% (61326)	92.6% (20443)	92.6% (20443)
0	7.4% (4878)	7.4% (1625)	7.4% (1626)

The proportion of 0 and 1 remains consistent

ML Pipeline

ML Pipline

Numeric Transformer: MinMaxScaler() for Age

↓

Ordinal Transformer: OrdinalEncoder() for `Episode_Number`

↓

Final Standard Scaler: StandardScaler()

↓

ML Model: GridSearchCV

Cross-validation

- ML Models and their Corresponding Hyperparameters

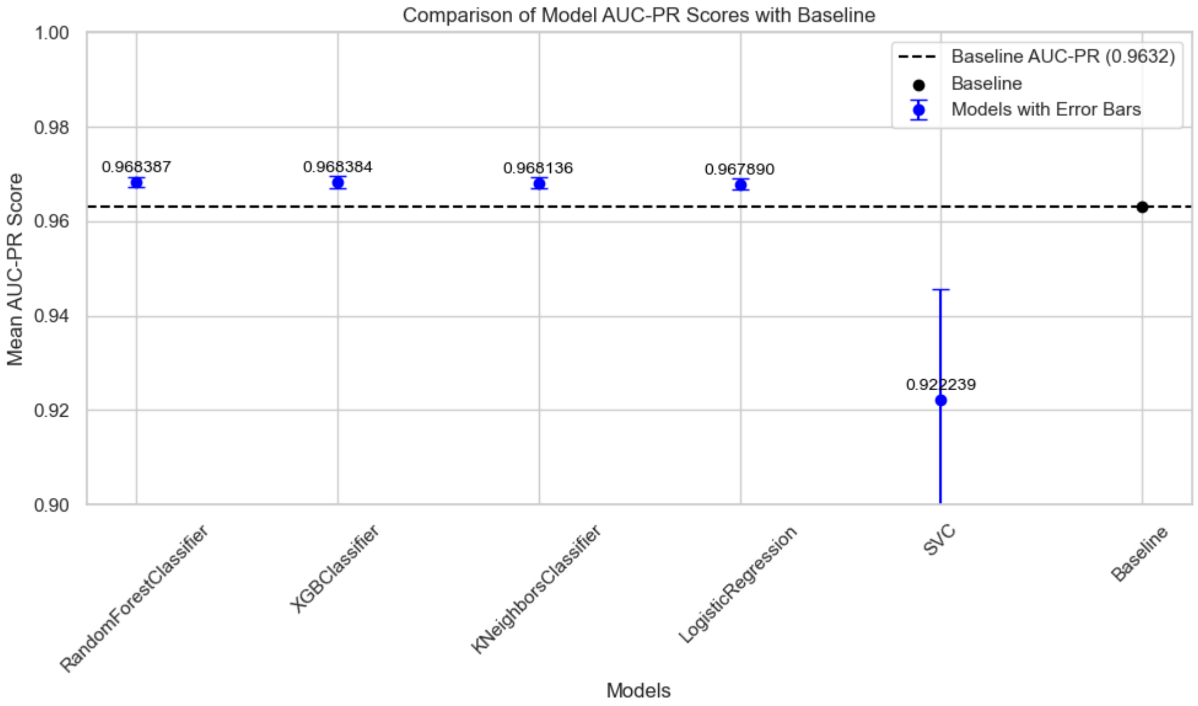
ML Model	Hyperparameters
Logistic Regression	'penalty': ['elasticnet'], 'C': [1e-2, 1e-1, 1e0, 1e1], 'l1_ratio': [0, 0.25, 0.5, 0.75, 1]
Random Forest Classifier	max_depth: [2, 3, 4, 5, 6], max_features: [0.7, 0.75, 0.8, 0.85, 0.9]
Kneighbors Classifier	n_neighbors: [1000, 1500, 1700], metric: ['euclidean', 'manhattan'], weights: ['uniform']
XGBoost Classifier	reg_alpha: [1e0, 1e1, 1e2], reg_lambda: [1e-2, 1e-1, 1e0, 1e1], max_depth: [1,3,5,7]
Support Vector Classifier	gamma: [1e-1, 1e0, 1e1], C: [1e-1, 1e0, 1e1]

Results

Evaluation metric: AUC-PR Score

- Highly imbalanced data
- Doesn't consider the true negative rate.

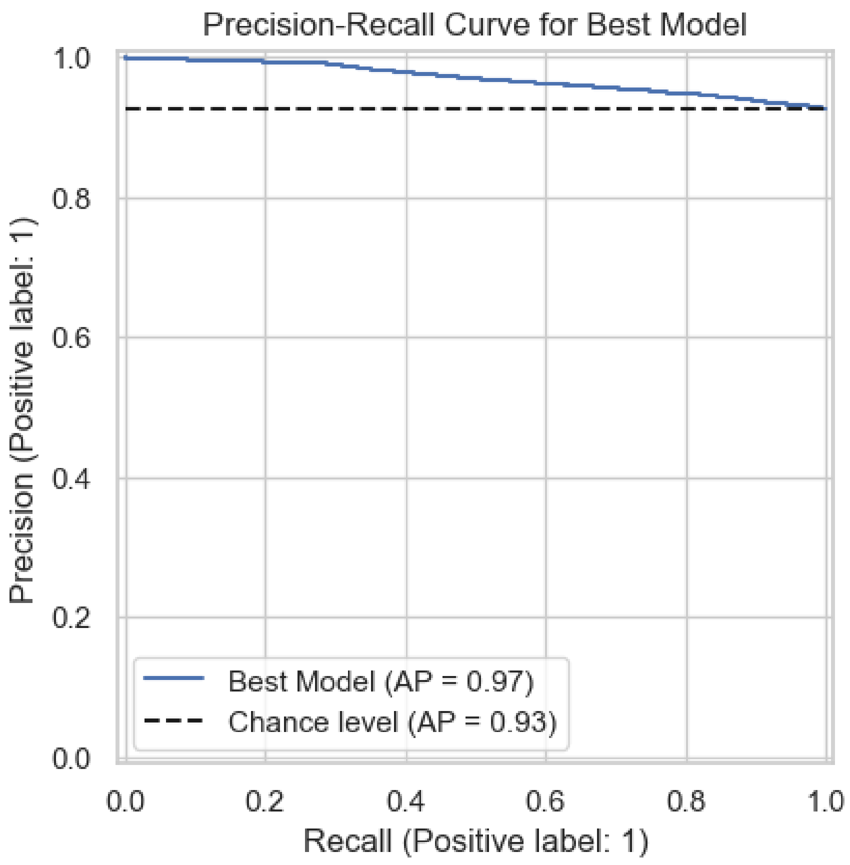
```
# Add the baseline AUC-PR
baseline = np.sum(y_test) / len(y_test)
baseline_y_pred = np.full((22069,), baseline)
precision, recall, _ = precision_recall_curve(y_test, baseline_y_pred)
baseline_AUC_PR = auc(recall, precision)
```

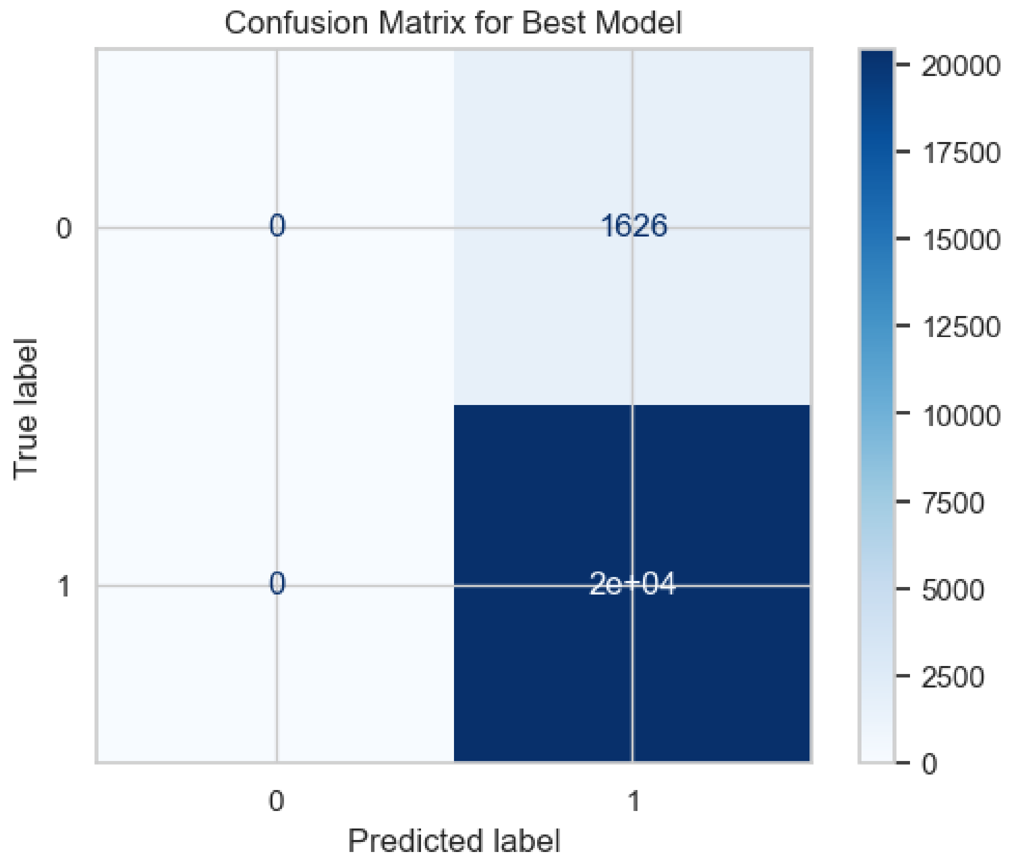


ML Model	Mean Test Score	Standard Deviation
Logistic Regression	0.9679	0.0012
Random Forest Classifier	0.9684	0.0011
Kneighbors Classifier	0.9681	0.0011
XGBoost Classifier	0.9684	0.0013
Support Vector Classifier	0.9222	0.0233

Results

Best Model:
RandomForestClassifier(max_depth=3, max_features=0.8)
(Random State: 42)

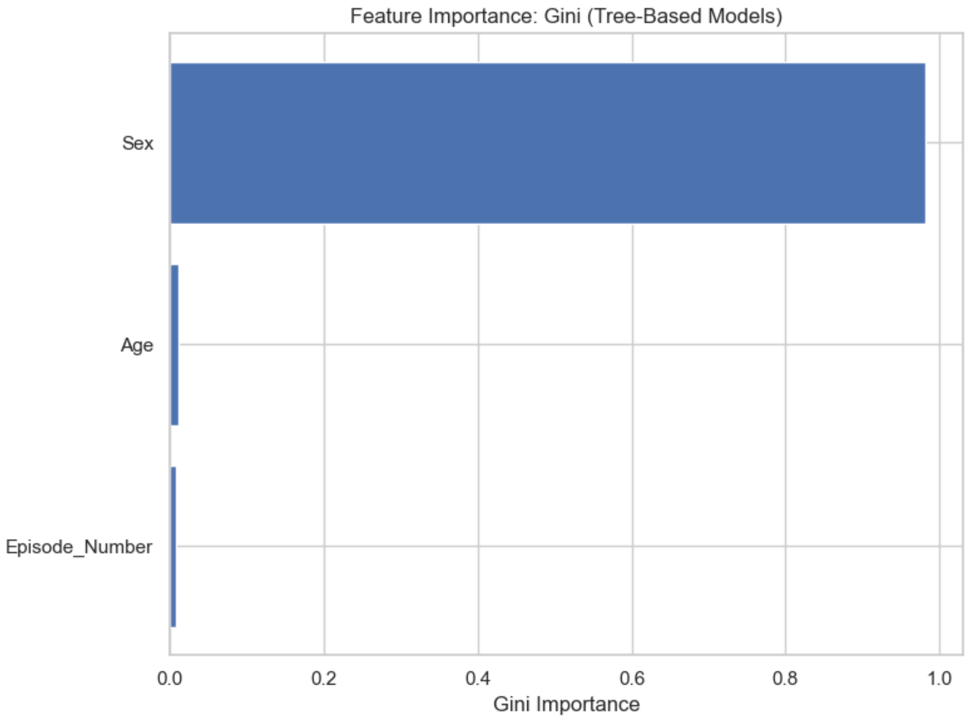




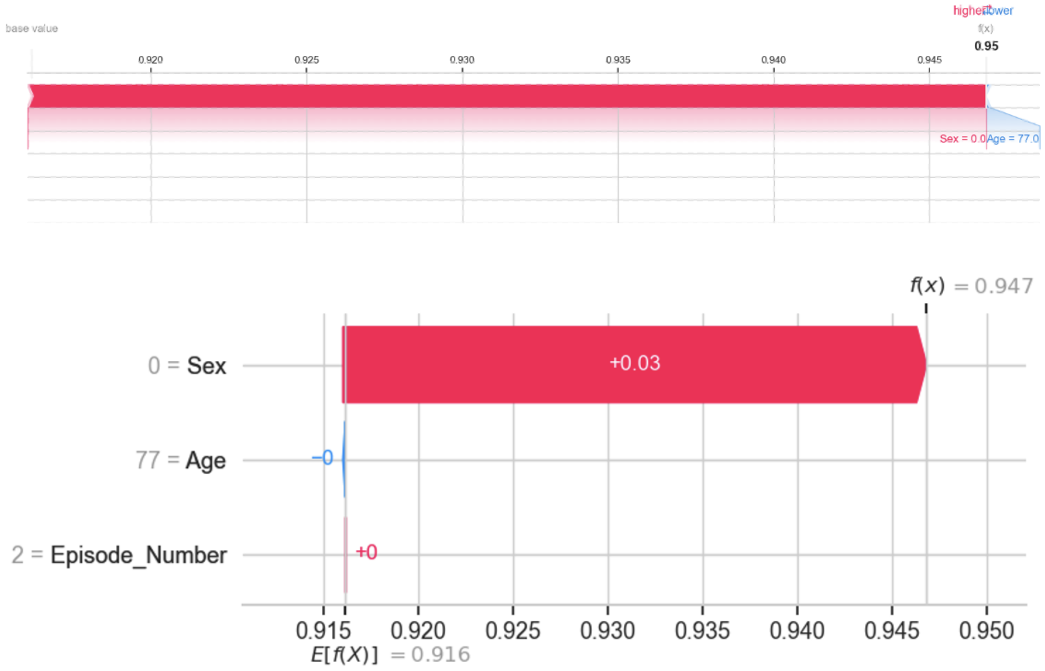
Feature Importance

- **Sex** feature has the strongest influence on the prediction
- **Age** and **Episode_Number** have almost no contributions

- Global:

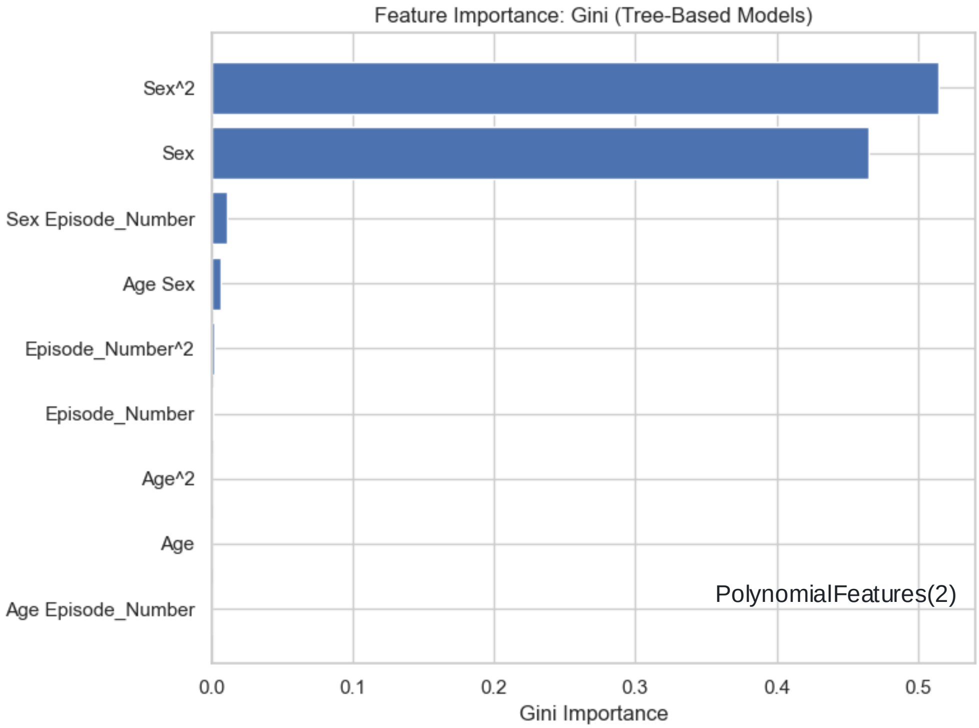


- Local (index=10000):



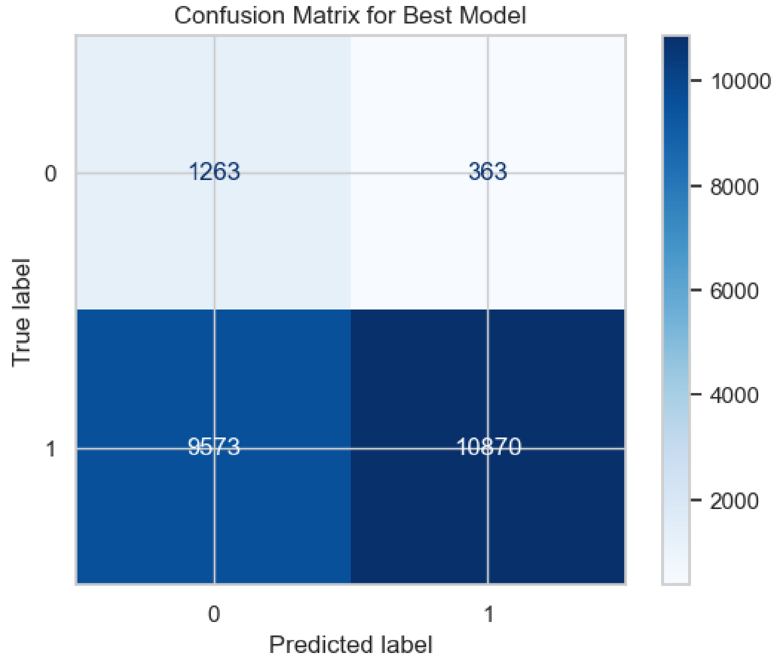
Outlook

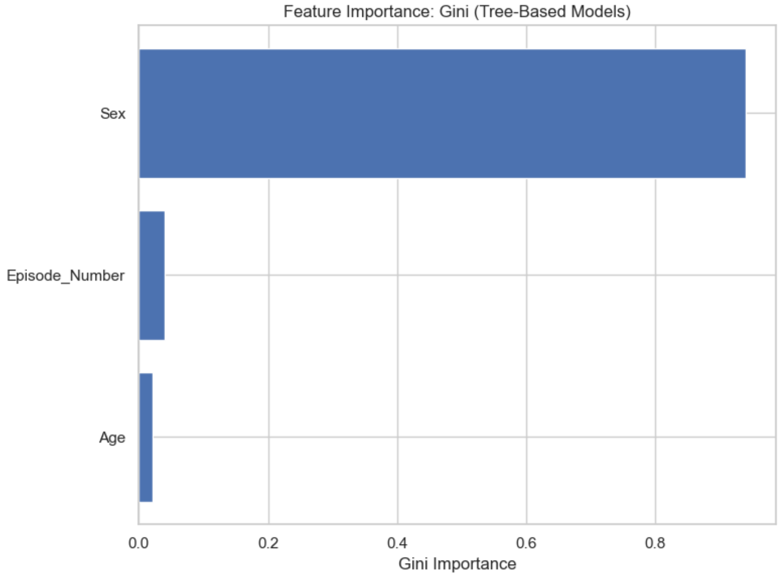
1. Add Class Weights to the Model: Modifies the contribution of each sample to the loss function. modifies the contribution of each sample to the loss function.For example, in RandomForestClassifier, set **class_weight='balanced**'
2. Resample the Dataset: Try to use **oversampling** (e.g., SMOTE) to increase the representation of the minority class or **undersampling** to reduce the dominance of the positive class.
3. Try advanced ML classification algorithms like Neural Networks or Naïve Bayes Classifiers.
4. Try to incorporate it with other datasets to increase more features relevant to the hospital outcome

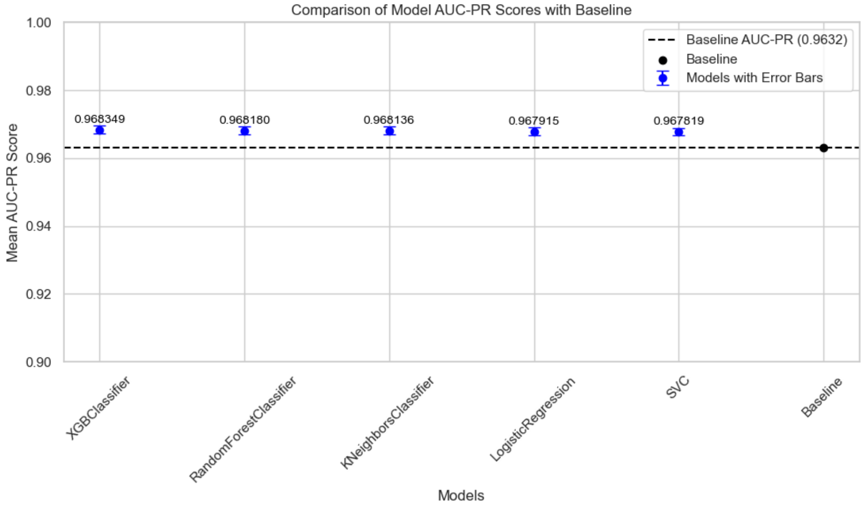


Thanks for Watching

Q&A


Confusion Matrix for Best Model
1263
363
9573
10870
True label
0
1
Predicted label
0
1
10000
8000
6000
4000
2000


Feature Importance: Gini (Tree-Based Models)
Sex
Episode_Number
Age
0.0
0.2
0.4
0.6
0.8
Gini Importance


Comparison of Model AUC-PR Scores with Baseline
Baseline AUC-PR (0.9632)
Baseline
Models with Error Bars
0.968349
0.968180
0.968136
0.967915
0.967819
Mean AUC-PR Score
1.00
0.98
0.96
0.94
0.92
0.90
XGBClassifier
RandomForestClassifier
KNeighborsClassifier
LogisticRegression
SVC
Baseline
Models