Enhancing Patient Care: A Real-Time Healthcare Monitoring, Alerting, And Predictive System

Group 8

Anshit Verma, Manvi Goel, Navdha Agarwal, Padma Priya Mukkamala, Reshma Ramachandra, Sindhu Raghuram Panyam

Motivation and Problem Statement

Motivation

- Improve patient outcomes by advancing healthcare monitoring.
- Utilize Machine Learning algorithms and Frameworks such as Kafka, and Flink for proactive alerting, boosting healthcare efficiency.

Problem Statement

- Data overload: Managing growing healthcare data poses challenges
- Reactive system: Traditional approaches risk delayed responses and compromised care
- Personalisation Gap: Current systems lack personalized alerts, hindering individualized care

Project Overview

Revolutionize healthcare monitoring through a proactive and personalized healthcare monitoring system capable to handle large data

- Develop predictive models like ARIMA, Prophet, XGBoost, to forecast health conditions
- Utilize real-time big data analysis system (Kafka, Flink) to tailor alerts based on individual patient profiles
- Compare predicted behavior with real-time alerts to evaluate the effectiveness of our predictive analytics in improving patient outcomes.

Dataset

Dataset Selection:

- Utilizing the PhysioNet/Computing in Cardiology Challenge 2012 dataset for Predicting Mortality of ICU Patients.
- The dataset includes essential metrics of ICU patients.

Parameter Focus:

- Our focus is on monitoring patients using heart rate and respiratory rate parameters.
- These metrics provide valuable insights into the patients' physiological conditions.

Data preprocessing

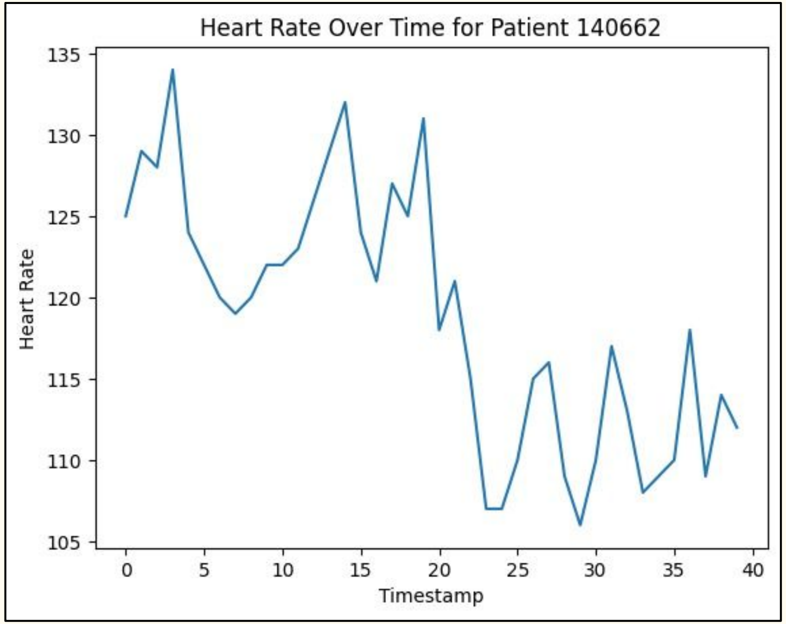
Data Preprocessing:

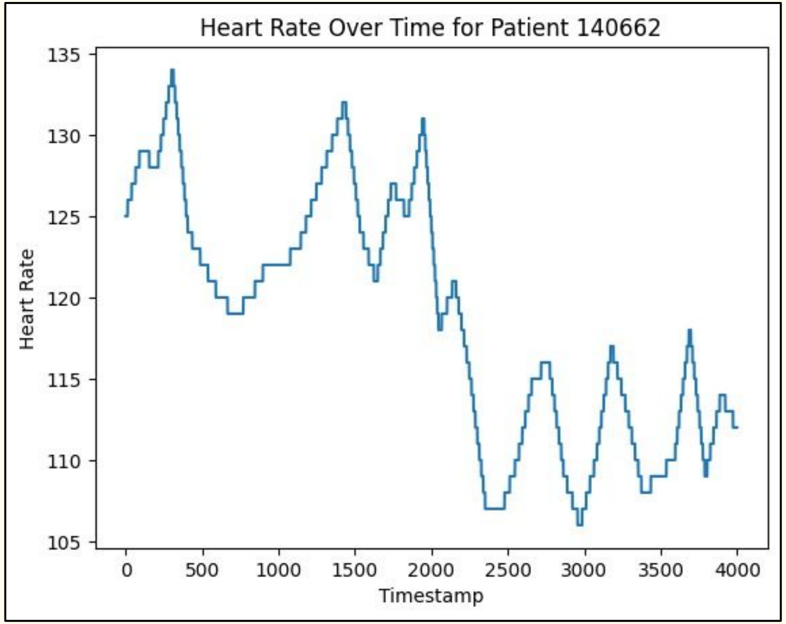
- Preprocessing involves identifying patients with both heart rate and respiratory rate data.
- The timestamps are adjusted to ensure consistency across all patients.

Interpolation for Data Augmentation:

- To expand the dataset, we interpolate the data.
- The interpolation process generates 3000 data points for each patient for improved statistical robustness.

Data interpolation



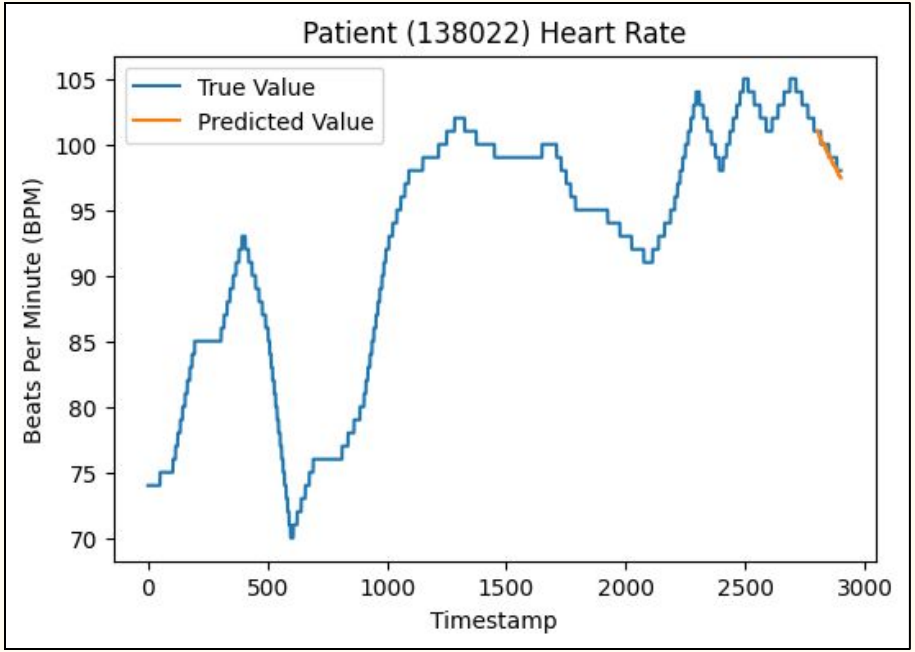


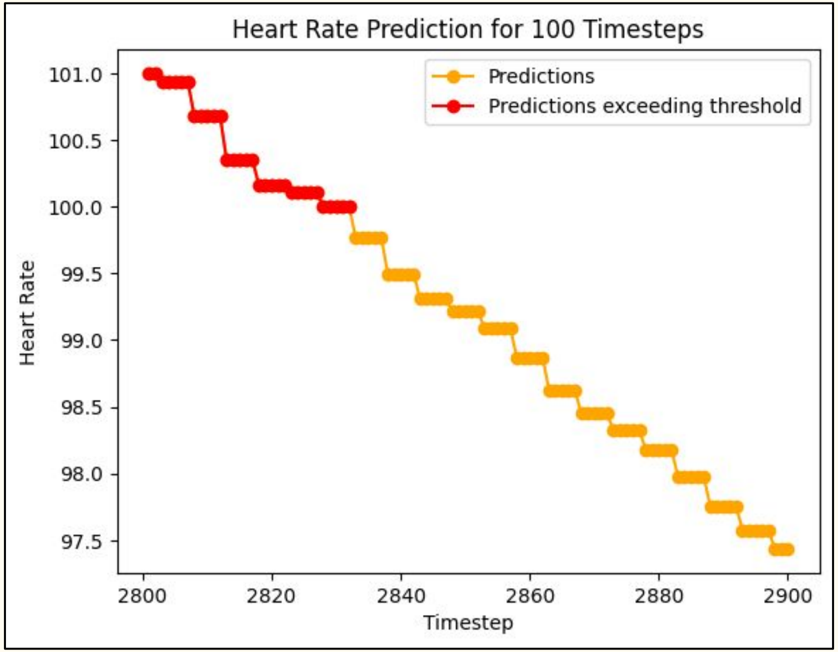
ARIMA

- ARIMA is an acronym that stands for **AutoRegressive Integrated Moving Average.** The key aspects of the model are:
  - *AR*: Autoregression. A model that uses the dependent relationship between an observation and some number of lagged observations.
  - *I*: Integrated. The use of differencing of raw observations (e.g. subtracting an observation from an observation at the previous time step) in order to make the time series stationary.
  - *MA*: Moving Average. A model that uses the dependency between an observation and a residual error from a moving average model applied to lagged observations.
- ARIMA models are a general class of models used for forecasting time series data. ARIMA models are generally denoted as ARIMA (p,d,q) where p is the order of autoregressive model, d is the degree of differencing, and q is the order of moving-average model.

ARIMA for Time Series

Heart Rate

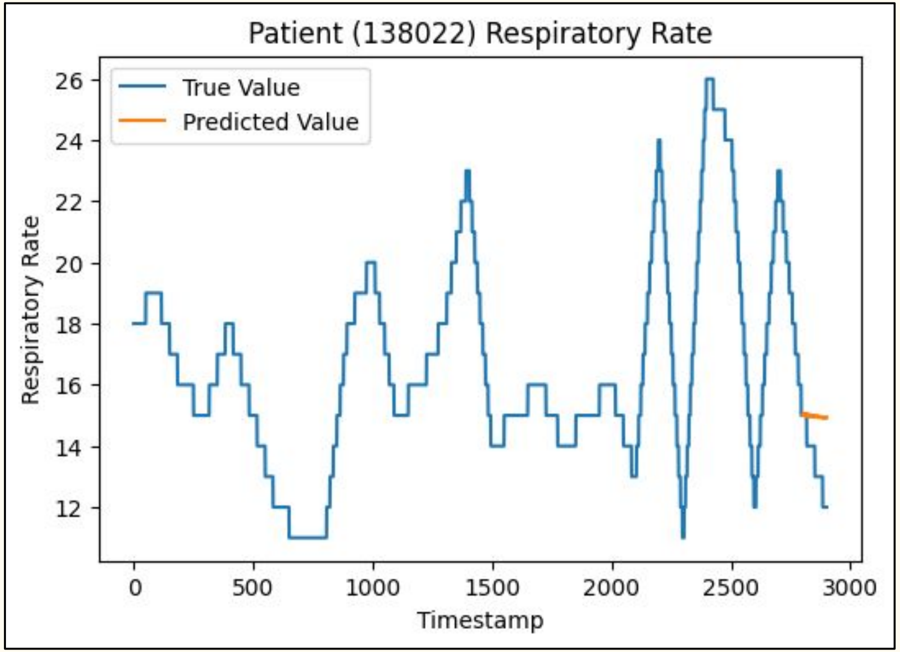


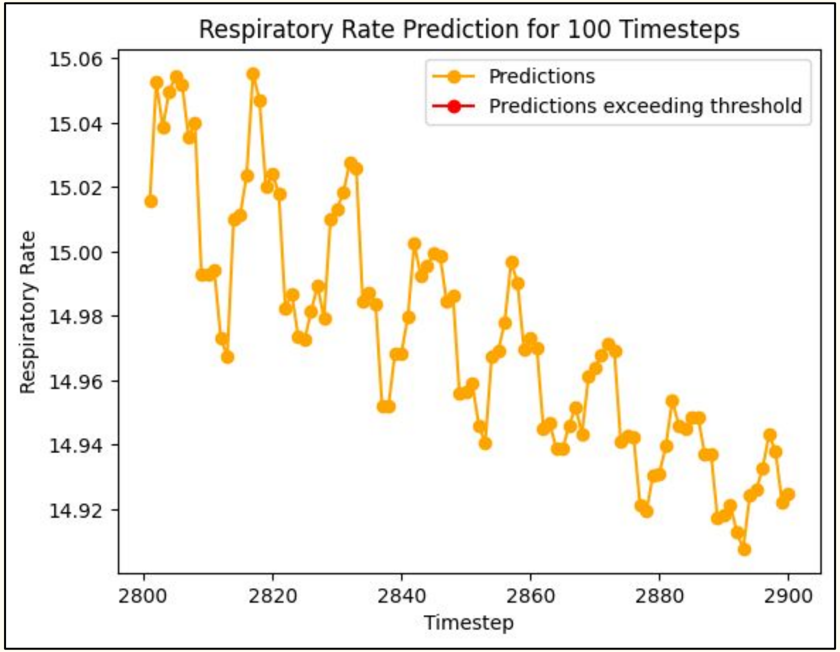


MAE = 0.35

ARIMA for Time Series

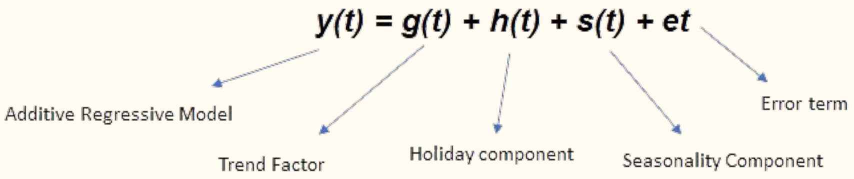
Respiratory Rate





MAE = 1.48

PROPHET



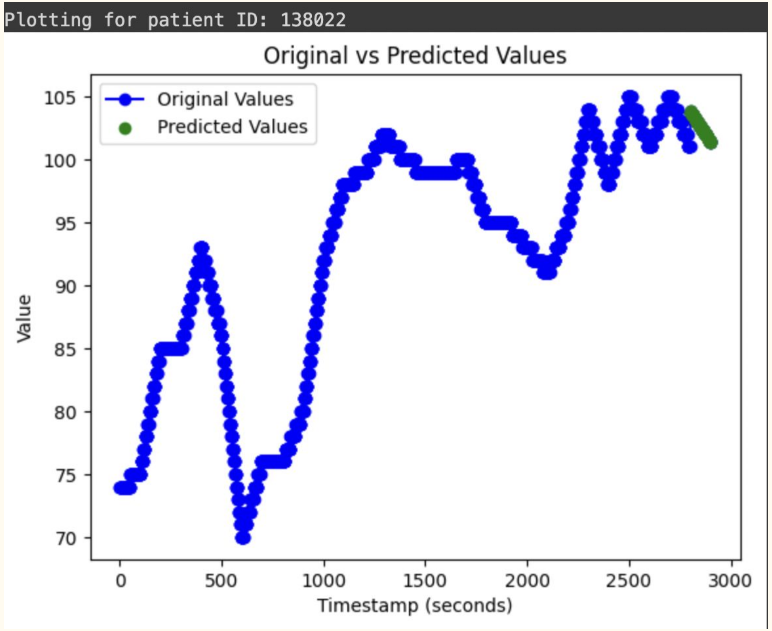
- Developed by Facebook for forecasting univariate time series data, especially effective for datasets with ***strong seasonal patterns.***
- The model adds up different factors like overall trend, seasonal patterns, and special events to make predictions.

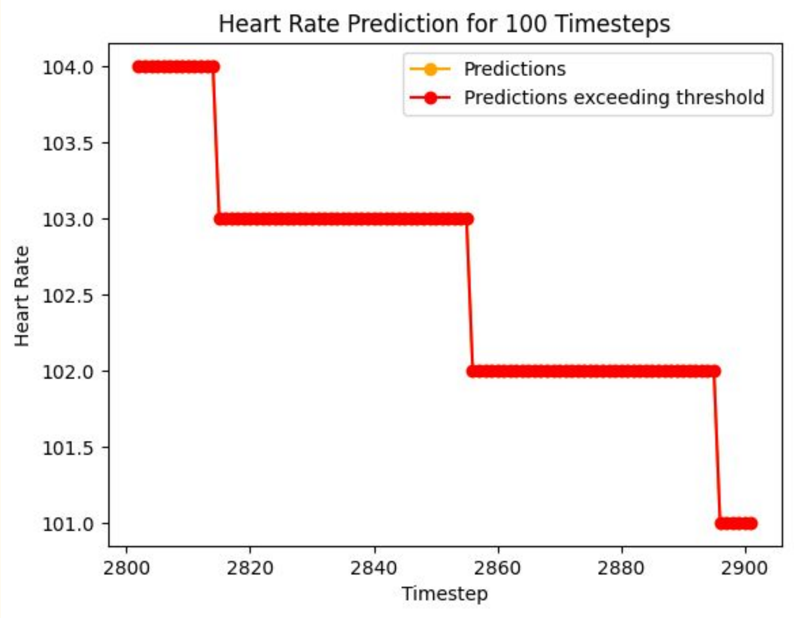
Use of Prophet in Health Monitoring:

- ***Identifying long-term trends*** in vital signs (e.g., blood pressure, heart rate) for chronic disease management.
- Spotting ***unusual patterns*** or sudden changes in vital statistics that may indicate health issues.
- Forecasting future vital sign values to provide early warnings for potential health risks.
- Tailoring patient monitoring and care plans based on individualized trends and predictions.
- Assisting in planning and resource allocation in healthcare settings by predicting future monitoring needs.

Prophet for Time Series

Heart Rate

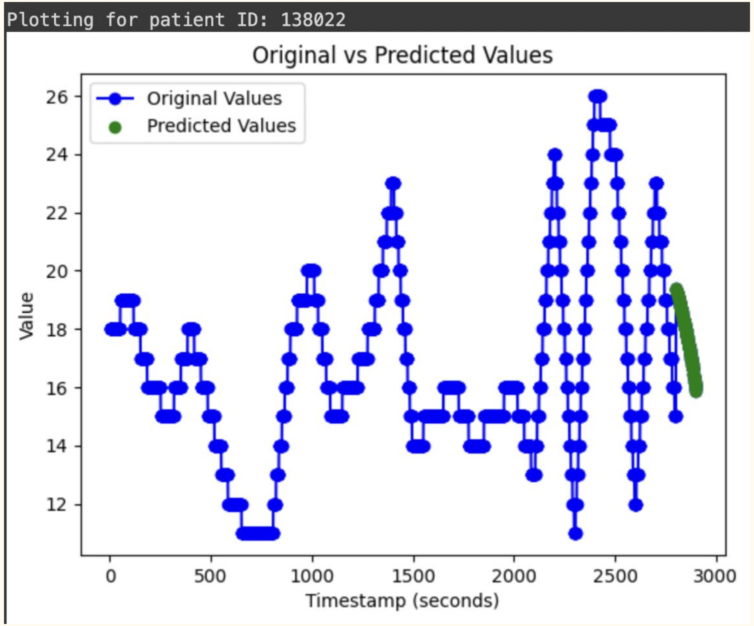


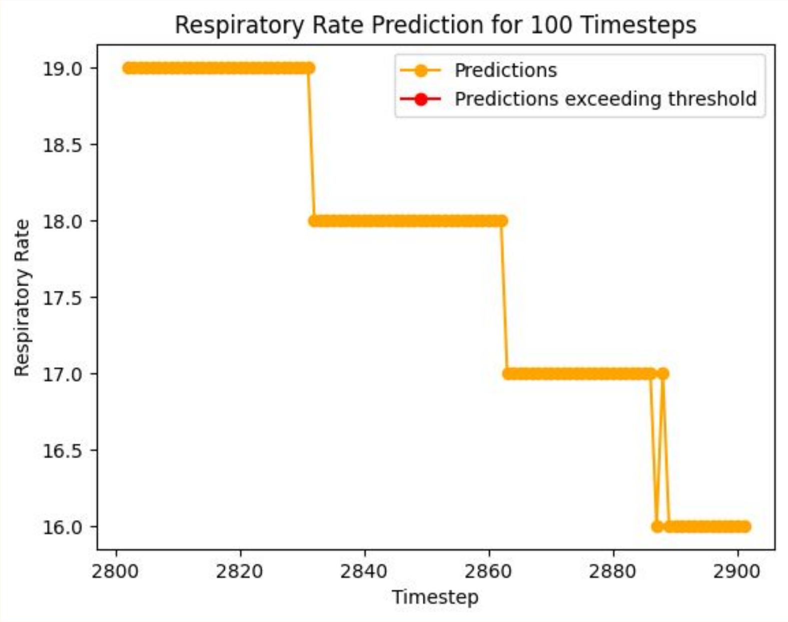


$$\mathrm{MAE} = 2.37$$

Prophet for Time Series

Respiratory Rate

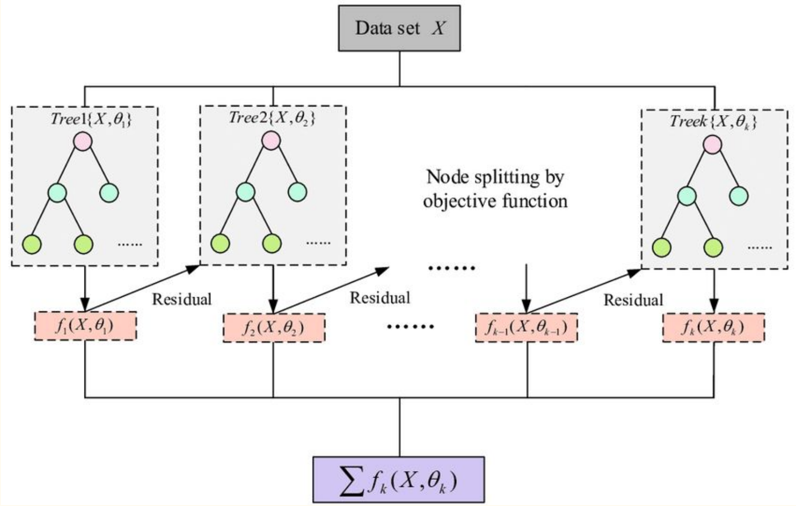




$\text{MAE} = 2.38$

XGBoost

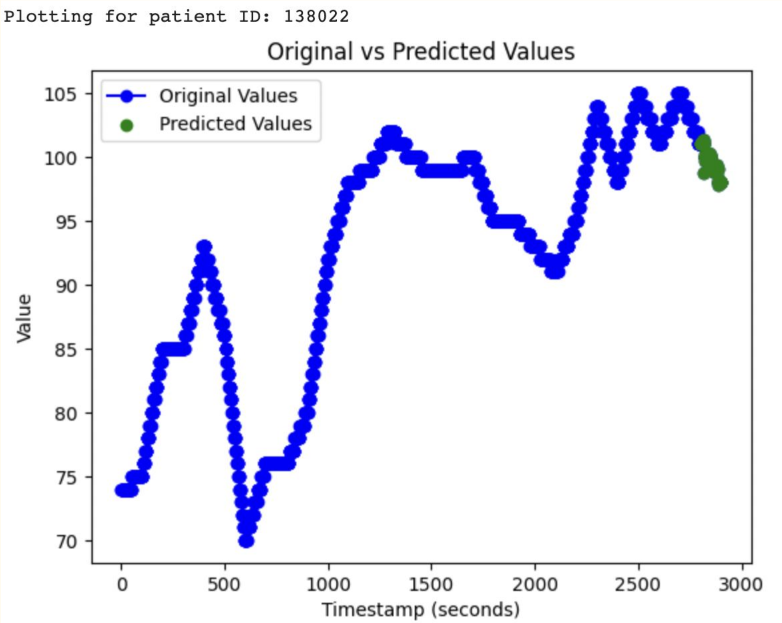
- ***eXtreme Gradient Boosting*** adds new trees by continuously splitting features.
- Adding a tree each time is actually learning a new function to fit the residual of the last prediction.
- The corresponding scores of each tree are added up to obtain the recognition prediction value of the sample.
- XGBoost is designed for ***speed and efficiency***, making it suitable for large datasets.
- It handles missing data, providing ***flexibility*** to adapt to different time series characteristics.
- It is ***robust to outliers***, making it effective for identifying key factors in time series data.

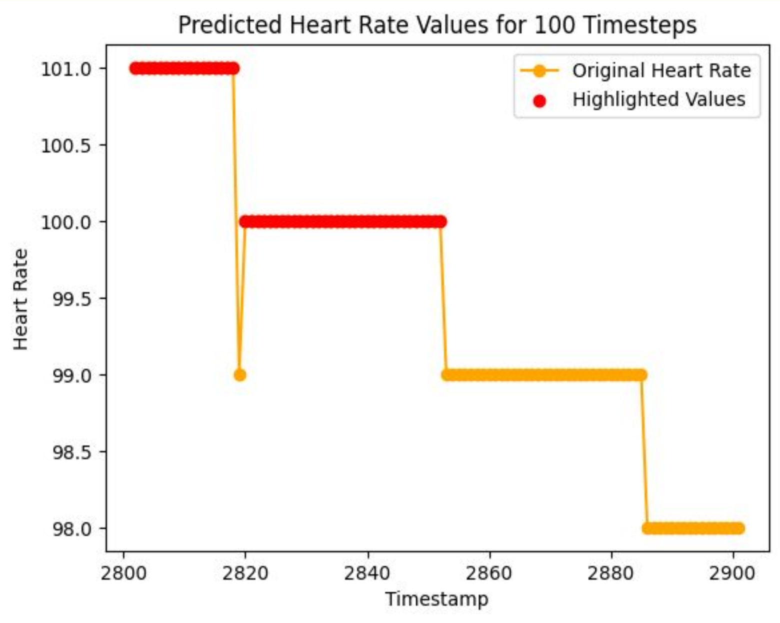


XGBoost flow chart

XGBoost for Time Series

Heart Rate

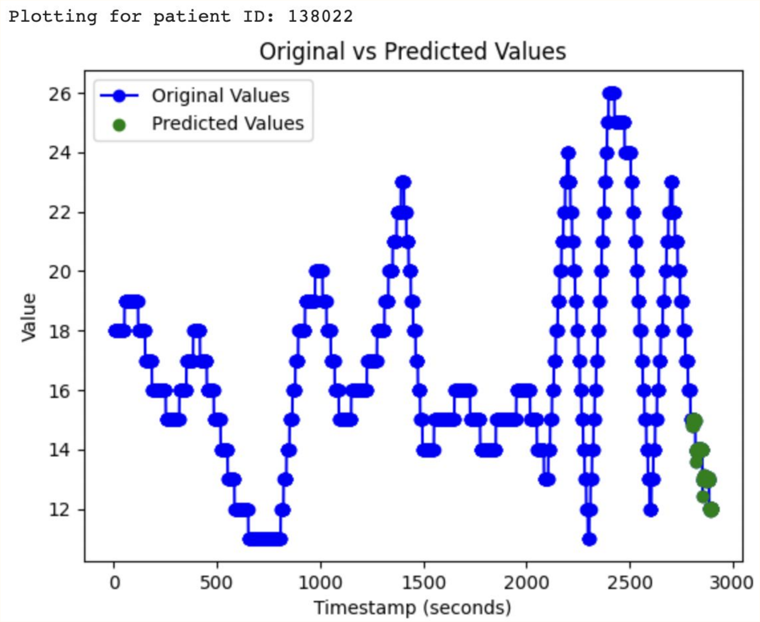


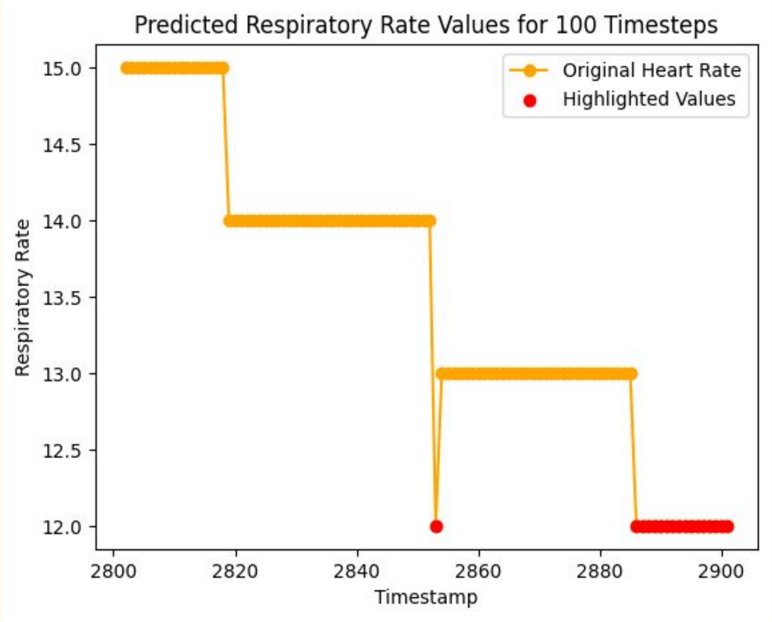


$$\mathrm{MAE} = 0.07$$

XGBoost for Time Series

Respiratory Rate





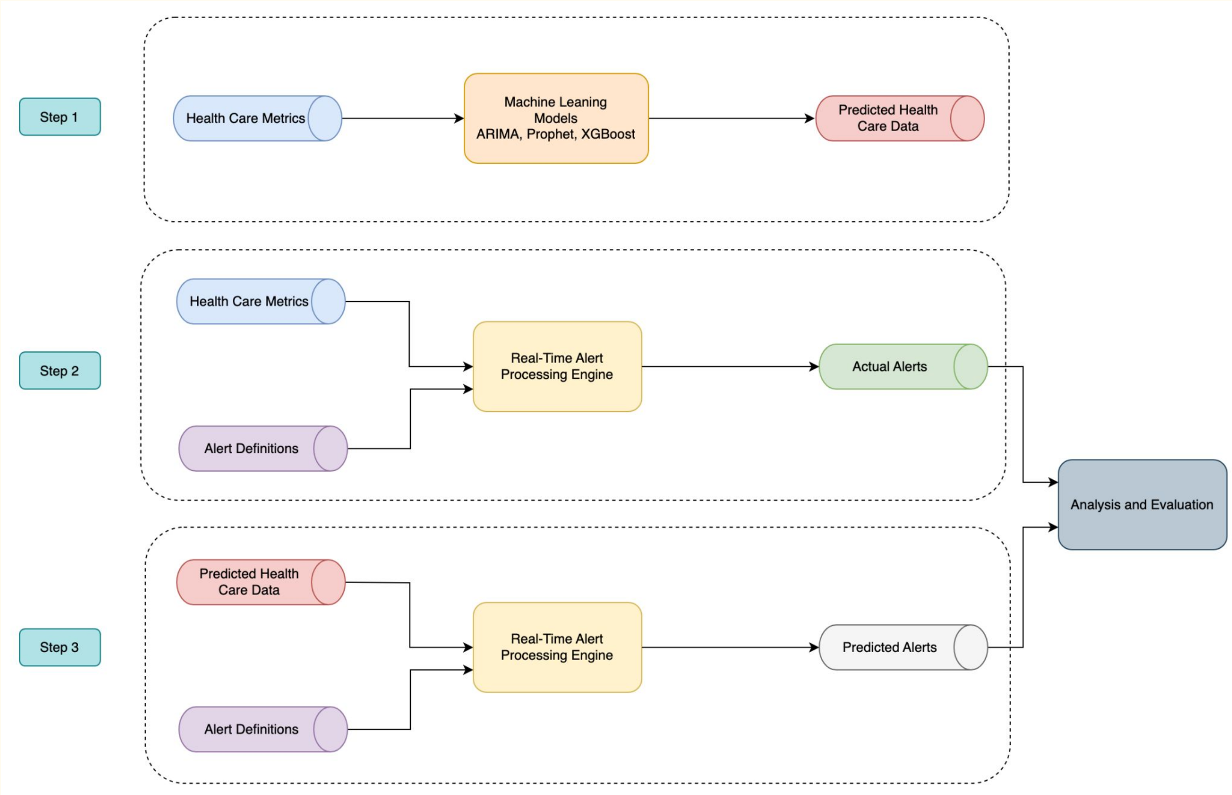
MAE = 0.05

Interpretation of the ML models

- Comparing the three predictive models, XGBoost seems to perform the best.
- Both Arima and Prophet do not work well with non-linear and highly varying data.
- XGBoost does a better job of filling in missing values and identifying unique patterns.
- This ensemble approach of using multiple predictive models ensures overall better results as compared to using only one of these models.

Model	Heart Rate MAE	Respiratory Rate MAE
Arima	0.35	1.48
Prophet	2.37	2.38
XGBoost	0.07	0.05

Real Time Big Data System



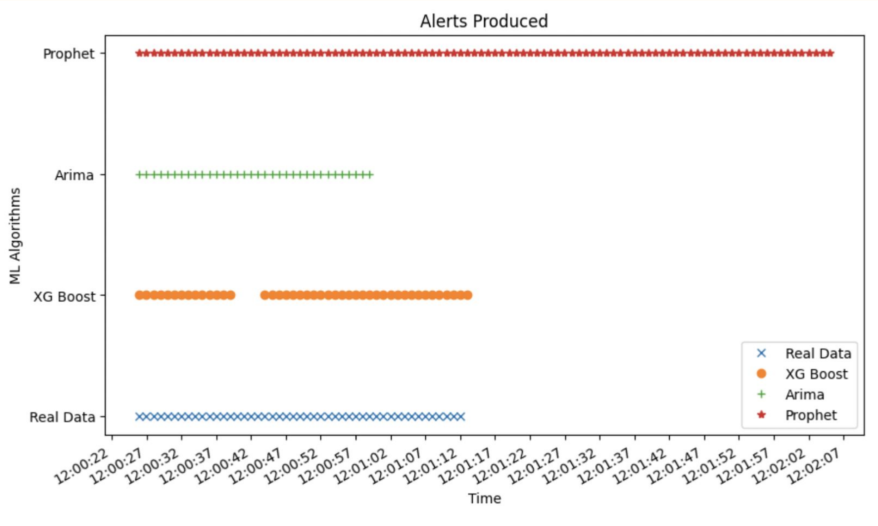

```json
{
    "time_id": "2023-11-01T12:00",
    "patient_id": "138022",
    "vital_name": "HeartRate",
    "vital_value": 74.0,
    "key": "138022#HeartRate"
}
```


Heart Rate Vital Datapoint


```json
{
    "patient_id": "138022",
    "vital_name": "HeartRate",
    "alert_threshold": 100.0,
    "alert_operator": ">",
    "violation_duration": 5,
    "key": "138022#HeartRate"
}
```


Alert Definition

Evaluation and Results



Model	Arima	Prophet	XGBoost
Accuracy	0.87	0.47	0.95
Precision	1.0	0.47	0.977
Recall	0.72	1.0	0.914
F1-Score	0.839	0.639	0.945

THANK YOU!