Project 6: Trend event pattern mining

Assigned to:

Background:

This project focuses on using unsupervised learning to mine trend-event patterns from clinical data to develop prognosis models. The goal is to identify hidden trends and events in patient data, such as lab results or treatments, which can provide insights into future health outcomes without labeled data. This approach can be applied to identify patterns that lead to certain events (like disease onset, recovery, or deterioration) by analyzing temporal data trends.

Problem Statement:

Develop a trend-event pattern mining framework that can aid in personalized treatment/prognosis.

Tasks:

- Conduct a review of existing unsupervised learning techniques used in healthcare, particularly for prognosis and trend analysis. Explore clinical studies that discuss common health events (e.g., disease onset, progression, or treatment response) and how such events can be inferred from data trends.
- Analyze the available data by looking at time-series trends in lab results, medications, and other clinical records. Investigate how patterns emerge over time and identify temporal correlations between different features (e.g., a rise in certain lab markers leading to an event like organ failure).
- Engineer time-based features such as moving averages, trend slopes, and time-lagged variables to capture the progression of health states over time. Additionally, explore methods for event detection and clustering of similar patterns, such as changes in vital signs or lab test trajectories.
- Apply unsupervised learning techniques such as clustering or pattern mining algorithms to identify common trends and event sequences. Evaluate the discovered patterns by comparing them with known clinical insights and by interpreting their relevance to prognosis.
- Document all aspects of the project, from data preprocessing to the discovered patterns and trends. Include explanations of how the trends align with clinical events and how they can be used in prognosis models. Report performance on

suitable evaluation metrics. Ensure the unsupervised nature of the model is well articulated, along with the limitations and potential for further refinement.

Deliverables:

- A pattern mining algorithm implemented for CureMD's EMR data that finds trend-event patterns in an unsupervised manner.
- Analyze and develop a prognostic use case that demonstrates the clinical utility of your designed
- An evaluation of your model's performance using suitable metrics - custom metrics can be used after approval
- A project report summarizing the entire project, including the background, prior art, problem statement, methodology, results, and conclusions. This documentation should be clear and organized for future reference and replication.
- A presentation summarizing the key findings, methodologies, and outcomes of the project.
- Properly documented source code.

Recommended reading
- https://sci-hub.se/https://ieeexplore.ieee.org/abstract/document/8904774
- https://sci-hub.se/https://dl.acm.org/doi/abs/10.1145/3412841.3441937
- https://sci-hub.se/https://dl.acm.org/doi/abs/10.1145/2339530.2339578
- https://sci-hub.se/https://dl.acm.org/doi/abs/10.1145/2508037.2508044

Project Engineering Guidelines
Here is a suggested research and development engineering cycle to follow:

- First do a thorough literature survey of the prior art related to the project and then shortlist the ones based on some justifiable rationale for this use case
- Understand the models at depth including mathematics so you know what is happening inside the state-of-the-art method
- Find suitable datasets or scientifically generate them for this use case
- Build a software architecture diagram including class and sequence diagrams you were taught in OOP
- Build your pipeline in real-time by using big data technologies if needed
- Bonus would be awarded if you did model engineering with strong mathematics background instead of just doing API calls like programmers
- Pick or create your testing and performance evaluation metrics and then measure them to show the inner working and behavior of your models on real-world

scenarios. Moreover, some notion of accuracy and effectiveness of such systems need to be established.

- Add comments in the code for documentation purpose, code with no comments will be penalized heavily.