AI Development Workflow Assignment

Part 1: Short Answer Questions (30 points)

1. Problem Definition *(6 points)*

Hypothetical AI Problem:

"Predicting Student Dropout Rates in Universities Using Academic and Behavioral Data."

Objectives:

1. Accurately identify students at high risk of dropping out early in the semester.
2. Enable university staff to take timely interventions for at-risk students.
3. Provide a predictive dashboard for academic advisors to monitor student performance trends.

Stakeholders:

- **University Administrators** – Responsible for retention and academic performance strategies.
- **Academic Advisors** – Directly interact with students and provide counseling/interventions.

Key Performance Indicator (KPI):

- **Accuracy of dropout prediction model (e.g., >85%)** measured by comparing predicted dropout labels with actual outcomes.

2. Data Collection & Preprocessing *(8 points)*

Data Sources:

1. **Student Information System (SIS):** Includes grades, attendance, course enrollment, demographic data.
2. **Learning Management System (LMS):** Logs student engagement (e.g., logins, quiz submissions, forum participation).

Potential Bias in the Data:

- **Socioeconomic status may be underrepresented.** Students from low-income backgrounds might drop out more frequently, yet this variable may not be captured accurately in the data, leading to a biased model that underestimates dropout risk for this group.

Preprocessing Steps:

1. **Handling Missing Data:** Use mean/mode imputation for numerical/categorical fields or drop records if missing values are excessive.
2. **Normalization:** Scale numerical features (e.g., GPA, number of logins) using Min-Max scaling to bring them to a comparable range.
3. **Encoding Categorical Variables:** Convert categorical features like "major," "gender," or "course level" into numerical form using one-hot encoding or label encoding.

3. Model Development *(8 points)*

Model Choice:

- **Random Forest Classifier**
    - Justification: Handles both numerical and categorical data well, resistant to overfitting, provides feature importance, and performs well with missing data and imbalanced classes.

Data Splitting Strategy:

- **70% Training Set** – Used for model learning.
- **15% Validation Set** – Used to tune hyperparameters and prevent overfitting.
- **15% Test Set** – Used for final performance evaluation on unseen data.

Two Hyperparameters to Tune:

1. **Number of Trees (n_estimators):** Controls model robustness and accuracy. Too few trees can underfit, too many can slow computation.
2. **Maximum Tree Depth (max_depth):** Helps control overfitting. Limiting tree depth makes the model more generalizable.

4. Evaluation & Deployment *(8 points)*

Evaluation Metrics:

1. **Recall:** Important for identifying as many dropouts as possible (i.e., minimizing false negatives).
2. **F1-Score:** Balances precision and recall, especially useful if dropout cases are relatively rare.

Concept Drift:

- **Definition:** Concept drift refers to changes in the data distribution or relationship between input features and target variables over time. For example, factors influencing student dropout may evolve (e.g., shift to online learning).

- **Monitoring Strategy:** Implement regular performance checks using live data. Retrain model periodically using recent data and compare KPIs to historical benchmarks.

Technical Challenge in Deployment:

- **Scalability:** The model must be able to handle real-time predictions for thousands of students across multiple campuses. This may require using cloud infrastructure and efficient APIs to serve predictions quickly and reliably.

Part 2: Case Study Application (40 Points)

Scenario:

A hospital wants an AI system to predict patient readmission risk within 30 days of discharge.

1. Problem Scope *(5 points)*

Problem Definition:

Hospitals face financial and clinical challenges due to high patient readmission rates. The aim is to create an AI-based system that predicts whether a patient is at risk of being readmitted within 30 days post-discharge.

Objectives:

1. Predict 30-day readmission risk using patient health data.
2. Support healthcare providers in prioritizing follow-up care.
3. Reduce avoidable readmissions and associated costs.

Stakeholders:

- **Hospital Management** – Interested in cost reduction and regulatory compliance.
- **Healthcare Providers (Doctors, Nurses)** – Use model insights to personalize care plans.
- **Patients** – Benefit from proactive care and reduced risk of complications.

2. Data Strategy *(10 points)*

Proposed Data Sources:

1. **Electronic Health Records (EHRs):** Clinical notes, diagnoses, medication history, vitals, lab results.
2. **Demographic Data:** Age, gender, race, insurance status, socioeconomic factors.
3. **Hospitalization History:** Previous admissions, length of stay, comorbidities.

Two Ethical Concerns:

1. **Patient Privacy:** Sensitive health data must be securely stored and shared only with authorized personnel, in compliance with HIPAA.
2. **Bias & Fairness:** The model could disproportionately misclassify based on race, age, or socioeconomic status if historical data contains systemic biases.

Preprocessing Pipeline:

1. **Missing Data Handling:**
    - Impute missing vitals/lab results using median or previous entries.
    - Drop columns with excessive missingness.
2. **Feature Engineering:**
    - Derive "number of prior admissions in 6 months."
    - Encode "discharge diagnosis" using ICD code groupings.
    - Calculate average stay duration or medication complexity score.
3. **Normalization & Encoding:**
    - Normalize numerical features (e.g., blood pressure, lab values).
    - One-hot encode categorical fields like department or discharge type.

3. Model Development *(10 points)*

Model Choice:

- **Gradient Boosting Machine (e.g., XGBoost)**
    - Justification: Performs well on structured/tabular medical data, handles missing values efficiently, and provides feature importance for interpretability.

Confusion Matrix (Hypothetical Data):

	Predicted: No Readmission	**Predicted: Readmission**
Actual: No Readmission	820	180
Actual: Readmission	90	210

Calculations:

- **Precision** = TP / (TP + FP) = 210 / (210 + 180) = **0.538**
- **Recall** = TP / (TP + FN) = 210 / (210 + 90) = **0.7**

4. Deployment *(10 points)*

Integration Steps:

1. **API Development:** Build REST APIs to integrate predictions with the hospital's Health Information System (HIS).
2. **Batch or Real-time Scoring:** Deploy model to generate predictions at discharge time.
3. **Alert System:** Flag high-risk patients for care managers or doctors in the EHR interface.
4. **Training for Staff:** Provide onboarding sessions for staff to interpret and act on model outputs.
5. **Feedback Loop:** Collect feedback from end users for continuous improvement.

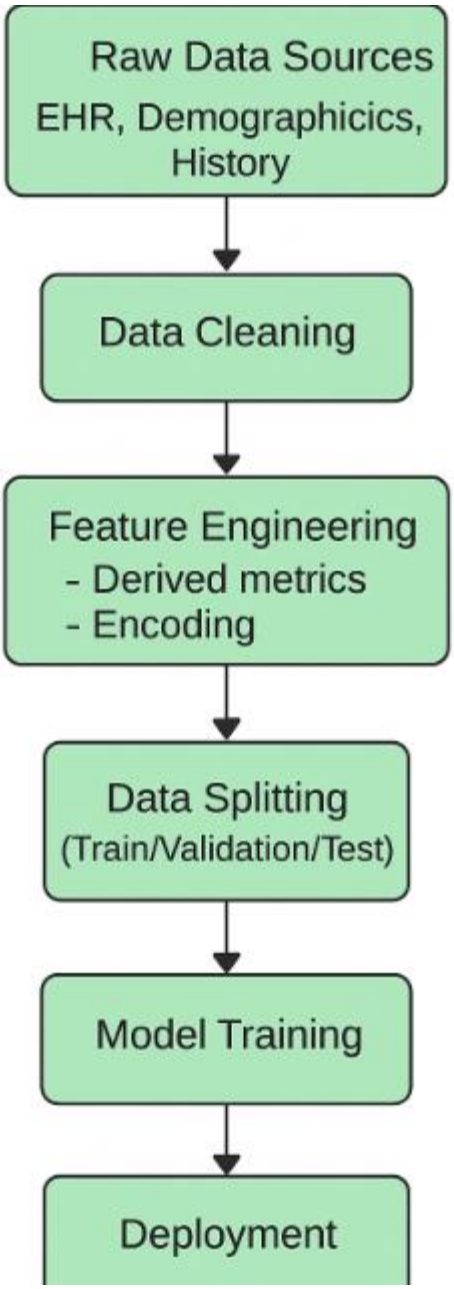
Regulatory Compliance (HIPAA):

- Ensure **data encryption at rest and in transit.**
- Apply **access controls** for data usage and model inference.
- Maintain **audit logs** of model access and predictions.
- Conduct **regular audits and assessments** to ensure model fairness and security.

5. Optimization *(5 points)*

Method to Address Overfitting:

- **Use Cross-Validation with Early Stopping:**
  Apply k-fold cross-validation during training and stop training early when validation error stops improving to prevent overfitting while maximizing performance.



Part 3: Critical Thinking (20 Points)

Ethics & Bias *(10 points)*

How might biased training data affect patient outcomes in the case study?

Biased training data can lead to **discriminatory predictions** that negatively impact patient care. For example, if the data underrepresents certain racial or socioeconomic groups, the model may fail to accurately predict readmission risk for these populations. This can result in **inequitable healthcare**, where at-risk individuals don't receive necessary interventions, while others may be

over-monitored unnecessarily. Such bias may widen existing healthcare disparities and violate ethical standards and legal obligations (e.g., HIPAA, anti-discrimination laws).

One Strategy to Mitigate This Bias:

- **Bias Auditing with Fairness Tools:** Use tools like **IBM AI Fairness 360** or **Fairlearn** to evaluate model outputs across demographic subgroups. If disparities are found, apply techniques like **re-weighting**, **data augmentation**, or **adversarial debiasing** to ensure the model treats all patient groups fairly.

Trade-offs *(10 points)*

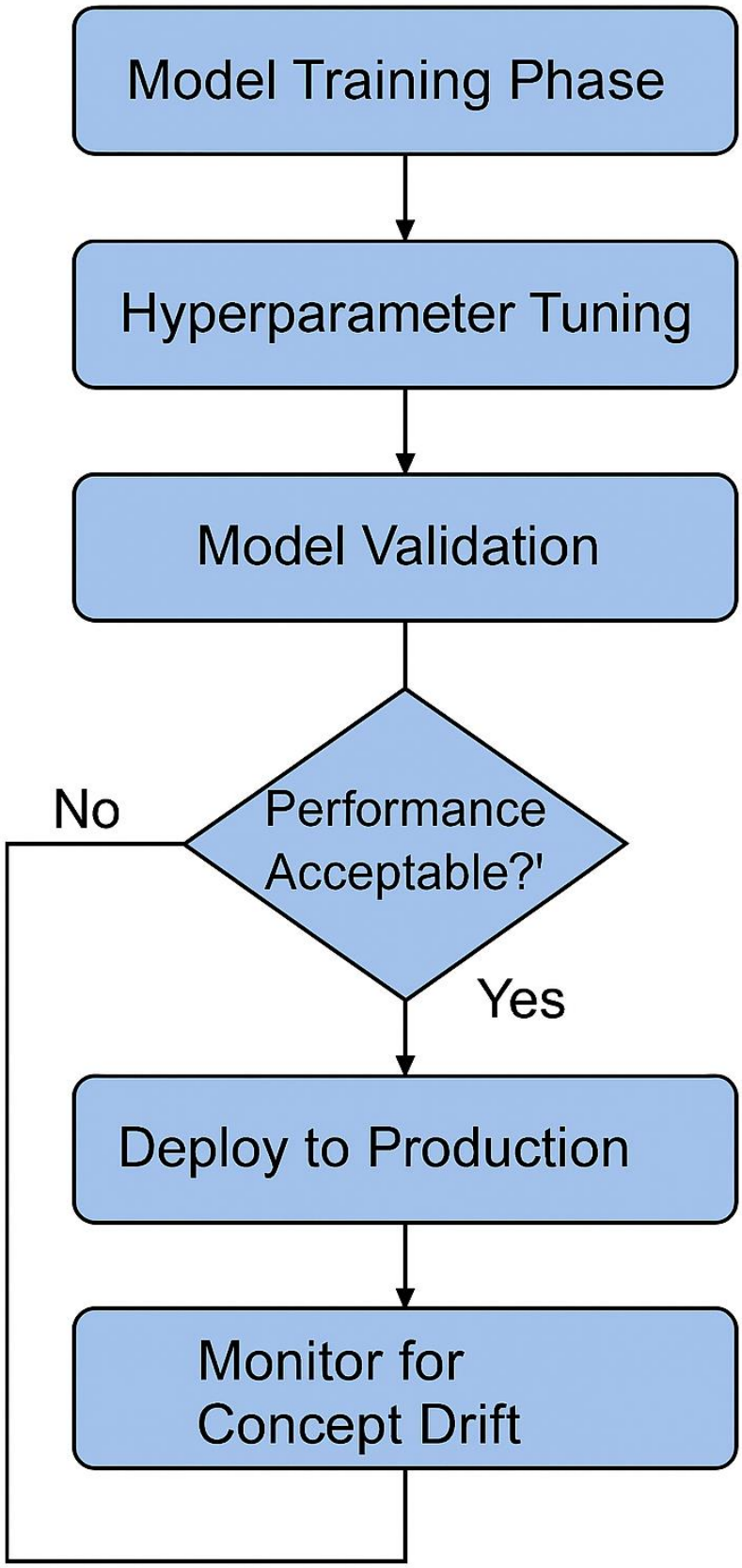
Model Interpretability vs. Accuracy in Healthcare:

In healthcare, **interpretability is critical**—doctors and nurses must understand why a model makes a prediction to trust and act on it. Highly accurate models like **deep neural networks** may outperform simpler models but are often **"black boxes"** and lack transparency. On the other hand, models like **logistic regression or decision trees** are more interpretable but might sacrifice predictive performance.

- **Trade-off:**
  Choosing a more accurate but opaque model could improve patient outcomes but reduce **clinician trust**, possibly leading to underuse or misinterpretation. In contrast, a more interpretable model might be less accurate but **more accepted** in clinical workflows.

Impact of Limited Computational Resources on Model Choice:

If the hospital has limited computational infrastructure, complex models like **neural networks or ensemble methods** may not be feasible due to their high memory and processing demands. This limitation would favor **lightweight models** such as **logistic regression**, **Naive Bayes**, or **decision trees**, which are faster to train and deploy—even if it means slightly lower accuracy. Additionally, model updates and real-time inference must be optimized for speed and cost.


Model Training Phase
Hyperparameter Tuning
Model Validation
Performance Acceptable?'
No
Yes
Deploy to Production
Monitor for Concept Drift

Part 4: Reflection & Workflow Diagram (10 Points)

Reflection *(5 points)*

What was the most challenging part of the workflow? Why?

The most challenging part was the **data preprocessing and bias mitigation phase**. This stage required not only cleaning and transforming raw data from various sources (EHRs, demographics, hospitalization records) but also carefully identifying and addressing **potential biases** that could compromise fairness and safety in healthcare predictions. Dealing with missing data, inconsistent coding in medical records, and ethical concerns around underrepresented groups made it a complex, sensitive, and resource-intensive process.

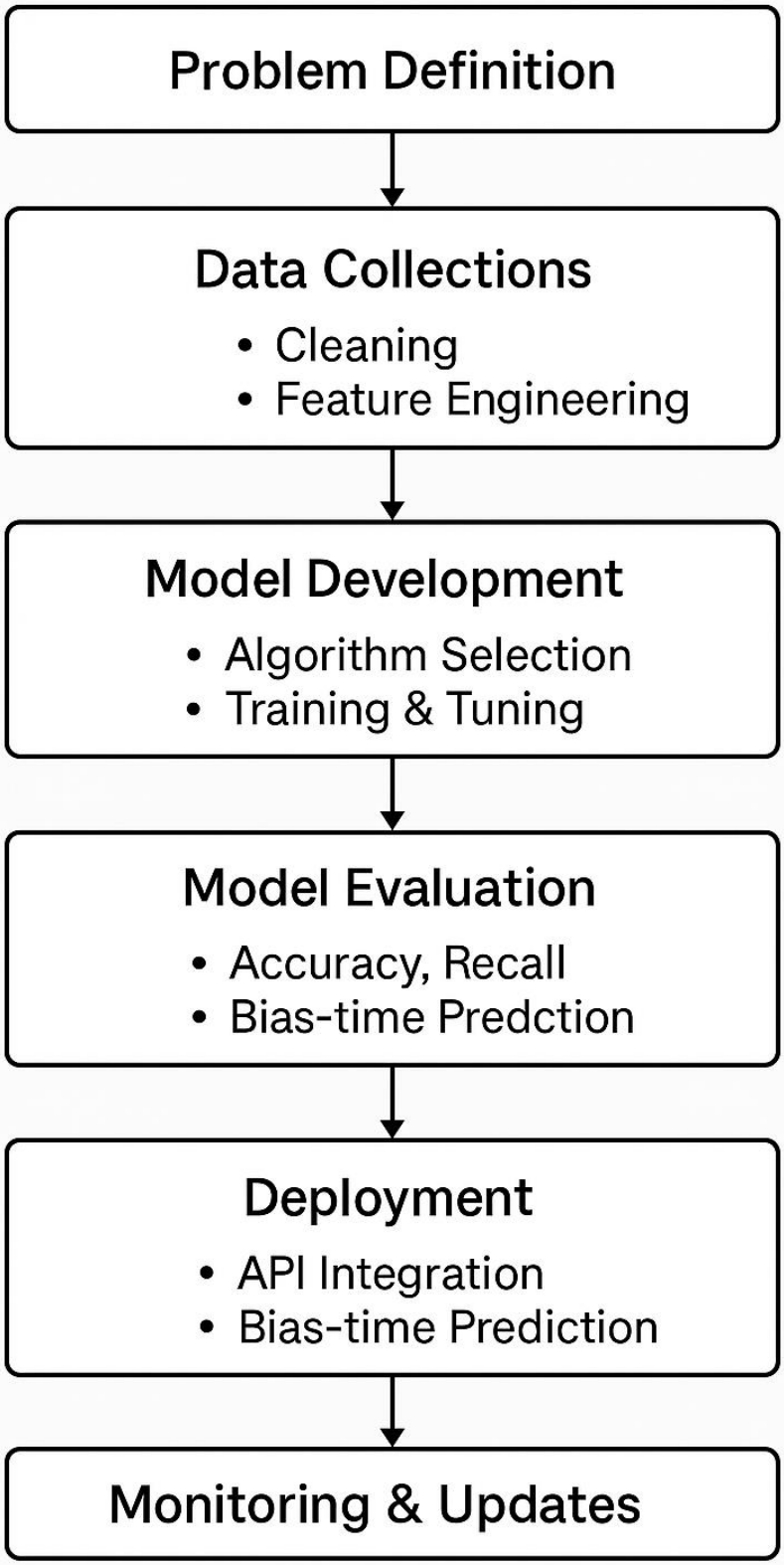
How would you improve your approach with more time/resources?

With more time and resources, I would:

- Conduct **a thorough exploratory data analysis (EDA)** using advanced visualization tools.
- Use **larger, more diverse datasets** that include underrepresented patient groups.
- Implement **automated pipelines** for preprocessing using tools like **DataRobot**, **MLflow**, or **Apache Airflow**.
- Involve **medical professionals and data ethicists** throughout the process to guide design decisions and validate the model's clinical relevance.

Diagram (5 points)

Here is a textual representation of the **AI Development Workflow Flowchart**


Problem Definition
Data Collections
• Cleaning
• Feature Engineering
Model Development
• Algorithm Selection
• Training & Tuning
Model Evaluation
• Accuracy, Recall
• Bias-time Predction
Deployment
• API Integration
• Bias-time Prediction
Monitoring & Updates