# AI Development Workflow Assignment

# Part 1

## 1. Problem Definition (6 points)

Hypothetical Problem: Predicting student dropout rates  
Objectives:  
1. Flag at-risk students at least one semester early.  
2. Provide advisors with a ranked list of risk factors per student.  
3. Reduce overall dropout rate by >= 10% within two academic years.  
Stakeholders:  
- Academic advisors and the retention office  
- Students and their parents  
Key Performance Indicator (KPI):  
- Percentage reduction in semester-to-semester dropout rate among flagged students

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

Data Sources:  
1. University Learning Management System (attendance, assignment submissions)  
2. Student Information System (demographics, GPA, financial aid status)  
Potential Bias:  
- Under-representation of non-traditional or part-time students could skew predictions toward full-time cohorts  
Preprocessing Steps:  
1. Impute missing GPA values using the semester-level median  
2. Normalize continuous features (e.g., GPA, attendance %)  
3. One-hot encode categorical variables such as major and housing status

## 3. Model Development (8 points)

Model Choice: Gradient Boosting Trees (e.g., XGBoost)  
Justification: Handles mixed data types well, captures non-linear interactions, and offers feature-importance scores  
Data Splitting Strategy:  
- 70% training set  
- 15% validation set  
- 15% test set

- Stratified by dropout label

Hyperparameters to Tune:  
1. max\_depth – Controls model complexity and prevents overfitting  
2. learning\_rate – Adjusts learning speed and model convergence

## 4. Evaluation & Deployment (8 points)

Evaluation Metrics:  
1. Recall – Ensures that most at-risk students are identified  
2. AUC-ROC – Provides a balanced evaluation across classification thresholds  
Concept Drift:  
- Refers to changes in the relationship between features and target over time (e.g., changes in learning behavior after a policy shift)  
- Can be monitored using rolling-window validation or metrics like PSI (Population Stability Index)  
Technical Challenge:  
- Scalability: Real-time scoring during peak periods like registration may overload the system  
- Solution: Deploy the model in containers (e.g., Docker) with auto-scaling APIs (e.g., Kubernetes)

**Part 2: Case Study Application**

### A. Problem Scope

Problem: Predict whether a patient will be readmitted within 30 days of discharge.  
  
Objectives:  
1. Alert care-coordination team early.  
2. Reduce readmissions by >= 15%.  
  
Stakeholders:  
- Hospital administrators (quality metrics)  
- Physicians, nurses, and patients

### B. Data Strategy

Data Sources:  
- Electronic Health Records (EHR): diagnosis codes, lab results, vitals.  
- Demographics: age, gender, socioeconomic index.  
- Hospital utilization: prior admissions, length of stay.  
Ethical Concerns:  
1. Patient privacy & HIPAA compliance.  
2. Bias against underserved minorities.  
  
Preprocessing Pipeline:  
1. De-identify PHI, tokenize IDs.  
2. Impute missing labs via last-observation-carried-forward.  
3. Feature engineering: Charlson Comorbidity Index, abnormal-lab count, discharge hour.  
**C. Model Development**

Model: Logistic Regression with L1 regularization  
Justification: Simple, interpretable, avoids overfitting with sparse weights.  
Confusion Matrix (Hypothetical: 1000 patients):

|  |  |  |
| --- | --- | --- |
|  | Predicted No | Predicted Yes |
| Actual No | 760 | 90 |
| Actual Yes | 60 | 90 |

Precision = 90 / (90 + 90) = 0.50  
Recall = 90 / (90 + 60) = 0.60

### D. Deployment

Integration Steps:  
1. Package model as REST service (FastAPI).  
2. Connect to hospital EHR using HL7/FHIR interface.  
3. Trigger nightly scoring upon discharge orders.  
  
Regulatory Compliance:  
- Perform HIPAA risk assessment.  
- Log all predictions with audit trails.  
- Encrypt data in transit (TLS) and at rest (AES-256).

### E. Optimization

Method: k-fold cross-validation with early stopping on validation AUC.  
Helps prevent overfitting without sacrificing data.

**Part 3: Critical Thinking**

### A. Ethics & Bias

Biased training data can result in under-prediction of readmission risk for certain groups (e.g., uninsured patients), leading to worse outcomes.  
Mitigation Strategy:  
- Re-weight samples or apply adversarial debiasing to equalize error rates across groups.

### B. Trade-offs

Interpretability vs. Accuracy:  
Clinicians prefer interpretable models (e.g., Logistic Regression) despite higher accuracy from deep models, due to trust and compliance needs.  
Resource Constraints:  
If resources are limited, models like Logistic Regression or Gradient Boosting are preferred over deep learning due to lower compute cost.

**Part 4: Reflection & Workflow Diagram**

### A. Reflection

Most Challenging Part:  
Ensuring high-quality labeled data. EHR records can be inconsistent.  
Improvement:  
Include clinicians in labeling and integrate social-determinant data for a more holistic view.

### B. Workflow Diagram

Workflow Stages:  
1. Problem Definition ->  
2. Data Collection ->  
3. Preprocessing & Feature Engineering ->  
4. Model Training & Validation ->  
5. Evaluation ->  
6. Deployment ->  
7. Monitoring & Maintenance (feedback loop)