**PLP ACADEMY**

AI FOR SOFTWARE ENGINEERING

GROUP FIVE MEMBERS

* ENOCK TAREI
* FESTUS PETRUS
* SHARON KIOKO
* JECINTA NJERI
* SIDNEY BARAKA
* EMMANUEL KIMARO

2025

**PLP ACADEMY**

**COURSE: SOFTWARE DEVELOPMENT**

**SPECIALIZATION: AI FOR SOFTWARE ENGINEERING**

**COHORT: FEBRUARY COHORT**

**GROUP FIVE**

**INSTRUCTOR: MR CHACKIN**

**1. Short Answer Questions**

**Part 1: AI Development Workflow (30 points)**

**1. Problem Definition (6 points)**

**Hypothetical AI Problem:** Predicting student dropout rates in higher education institutions

**3 Objectives:**

1. **Early Identification:** Identify at-risk students by the end of their first semester to enable timely intervention
2. **Resource Optimization:** Allocate academic support resources efficiently to students who need them most
3. **Retention Improvement:** Increase overall student retention rates by 15% within two academic years

**2 Stakeholders:**

1. **Academic Advisors:** Need actionable insights to provide targeted support and counseling to at-risk students
2. **University Administration:** Requires data-driven decisions for budget allocation and strategic planning to improve institutional outcomes

**1 Key Performance Indicator (KPI):**

**Student Retention Rate at 2 Years:** Percentage of students who remain enrolled or successfully graduate within 24 months of initial enrollment (Target: 85% retention rate)

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

**2 Data Sources:**

* **Student Information System (SIS):** Academic records including GPA, course enrollment, attendance, assignment submissions, and exam scores
* **Student Support Services Database:** Financial aid status, participation in tutoring programs, counseling sessions, and extracurricular activities

**1 Potential Bias:** **Socioeconomic Bias:** Students from lower-income backgrounds may appear more likely to drop out due to financial constraints rather than academic inability. This could lead the model to unfairly flag students based on economic status rather than academic performance, potentially creating a self-fulfilling prophecy where these students receive less academic investment.

**3 Preprocessing Steps:**

1. **Missing Data Handling:** Use multiple imputation for missing GPA values and forward-fill for attendance records, while flagging chronically missing data as a potential predictor itself
2. **Feature Normalization:** Standardize numerical features (GPA, test scores) using z-score normalization to ensure equal weight across different scales
3. **Categorical Encoding:** Apply one-hot encoding for categorical variables like major, enrollment status, and participation in support programs

**3. Model Development (8 points)**

**Model Choice:** **Random Forest Classifier**

**Justification:** Random Forest is ideal for this problem because:

* Handles mixed data types (numerical and categorical) effectively
* Provides feature importance rankings to understand key dropout predictors
* Resistant to overfitting with built-in cross-validation
* Offers good interpretability for stakeholders while maintaining high accuracy
* Robust to missing values and outliers common in educational data

**Data Splitting Strategy:**

* **Training Set:** 70% of data (stratified by dropout status)
* **Validation Set:** 15% for hyperparameter tuning and model selection
* **Test Set:** 15% for final unbiased performance evaluation
* Use temporal splitting to ensure training data precedes validation/test data chronologically

**2 Hyperparameters to Tune:**

1. **n\_estimators (Number of Trees):** Tune between 100-500 trees to balance performance and computational efficiency while avoiding overfitting
2. **max\_depth:** Limit tree depth (5-15) to control model complexity and improve interpretability for academic advisors

**4. Evaluation & Deployment (8 points)**

**2 Evaluation Metrics:**

1. **Precision:** Measures accuracy of dropout predictions to minimize false positives, ensuring advisors don't waste resources on students incorrectly flagged as at-risk
2. **Recall (Sensitivity):** Measures ability to identify actual at-risk students, critical for ensuring no genuinely at-risk students are missed

**Concept Drift:** Concept drift occurs when the relationship between input features and target outcomes changes over time. In education, this could happen due to:

* Changes in admission criteria
* New academic programs
* Economic conditions affecting student behavior
* Shifts in teaching methods or technology adoption

**Monitoring Strategy:**

* Track model performance monthly using a rolling window approach
* Monitor feature distributions for significant changes
* Set up automated alerts when precision/recall drop below 80% of baseline
* Implement A/B testing to compare model versions

**1 Technical Challenge During Deployment:** **Real-time Data Integration:** Ensuring the model receives updated student data from multiple systems (SIS, LMS, financial aid) in real-time while maintaining data consistency and handling system downtime gracefully.

**Part 2: Case Study Application (40 points)**

**Scenario: Hospital Patient Readmission Risk Prediction**

**1. Problem Scope (5 points)**

**Problem Definition:** Develop an AI system to predict the likelihood of patient readmission within 30 days of discharge to reduce healthcare costs and improve patient outcomes.

**Objectives:**

1. **Reduce Readmission Rates:** Decrease 30-day readmission rates by 20% through early intervention
2. **Optimize Resource Allocation:** Prioritize post-discharge care resources for high-risk patients
3. **Improve Patient Outcomes:** Enhance care coordination and prevent adverse events post-discharge

**Stakeholders:**

1. **Healthcare Providers:** Physicians and nurses need actionable insights to plan appropriate discharge procedures and follow-up care
2. **Hospital Administration:** Requires data-driven decisions to reduce penalties from excessive readmissions and optimize operational efficiency

**2. Data Strategy (10 points)**

**Data Sources:**

* **Electronic Health Records (EHRs):** Patient demographics, medical history, diagnoses, medications, vital signs, lab results
* **Claims Data:** Insurance information, previous hospitalizations, outpatient visits, specialist consultations
* **Social Determinants:** ZIP code-based socioeconomic data, distance to healthcare facilities, insurance coverage

**2 Ethical Concerns:**

1. **Patient Privacy:** Risk of data breaches exposing sensitive health information, requiring robust encryption and access controls
2. **Algorithmic Bias:** Potential discrimination against minority populations or low-income patients if historical healthcare disparities are embedded in training data

**Preprocessing Pipeline:**

1. **Data Cleaning:** Remove duplicate records, standardize medical codes (ICD-10), validate data integrity
2. **Feature Engineering:** Create composite scores (comorbidity index, medication adherence), calculate length of stay, derive age-adjusted vital signs
3. **Missing Data Imputation:** Use clinical decision trees for missing lab values, carry-forward for medication lists
4. **Normalization:** Standardize continuous variables (age, lab values) and encode categorical variables (diagnoses, medications)

**3. Model Development (10 points)**

**Model Selection:** **Gradient Boosting Classifier (XGBoost)**

**Justification:**

* Excellent performance with structured healthcare data
* Handles missing values naturally
* Provides feature importance for clinical interpretation
* Robust to outliers common in medical data
* Supports probability calibration for risk scores

**Hypothetical Confusion Matrix:**

|  |
| --- |
| Predicted |
| No Readmit Readmit |
| Actual No Readmit 850 50 (900 total) |
| Actual Readmit 30 70 (100 total) |

**Calculated Metrics:**

* **Precision:** 70/(70+50) = 58.3%
* **Recall:** 70/(70+30) = 70.0%

**4. Deployment (10 points)**

**Integration Steps:**

1. **API Development:** Create RESTful API endpoints for real-time prediction requests
2. **EHR Integration:** Develop HL7 FHIR-compliant interfaces to extract patient data automatically
3. **Dashboard Creation:** Build clinical decision support interface showing risk scores and recommendations
4. **Workflow Integration:** Embed predictions into discharge planning workflows and care coordination systems

**HIPAA Compliance:**

* **Data Encryption:** Implement end-to-end encryption for data in transit and at rest
* **Access Controls:** Role-based authentication with audit logging for all data access
* **Minimum Necessary:** Limit data access to only information required for prediction
* **Business Associate Agreements:** Ensure all third-party vendors sign appropriate agreements

**5. Optimization (5 points)**

**Method to Address Overfitting:** **Cross-Validation with Early Stopping**

Implement 5-fold cross-validation during training with early stopping criteria. Monitor validation loss and halt training when performance on validation set begins to degrade for 10 consecutive iterations. This prevents the model from memorizing training data patterns that don't generalize to new patients.

**Part 3: Critical Thinking (20 points)**

**1. Ethics & Bias (10 points)**

**Impact of Biased Training Data:** Biased training data could lead to systematic discrimination in patient care. For example, if historical data shows lower readmission rates for certain ethnic groups due to barriers in accessing care rather than better outcomes, the model might underestimate their risk. This could result in:

* Inadequate discharge planning for minority patients
* Perpetuation of healthcare disparities
* Reduced quality of care for vulnerable populations
* Legal and ethical violations

**Strategy to Mitigate Bias:** **Fairness-Aware Model Training:** Implement demographic parity constraints during model training to ensure equal true positive rates across different demographic groups. This involves:

* Regularly auditing model performance across subgroups
* Rebalancing training data to ensure representative samples
* Using bias correction techniques like adversarial debiasing
* Involving diverse clinical teams in model validation

**2. Trade-offs (10 points)**

**Interpretability vs. Accuracy Trade-off:** In healthcare, interpretability is crucial for clinical acceptance and regulatory compliance. While complex models like deep neural networks might achieve higher accuracy, simpler models like logistic regression provide clear feature weights that clinicians can understand and trust. The trade-off involves:

* **High Interpretability:** Enables clinical validation of model decisions and builds trust but may sacrifice predictive performance
* **High Accuracy:** Better patient outcomes through superior predictions but risks clinical rejection due to "black box" nature

**Impact of Limited Computational Resources:** Limited computational resources would necessitate:

* **Simpler Models:** Choose logistic regression or decision trees over ensemble methods
* **Reduced Feature Engineering:** Limit complex feature interactions and dimensionality reduction
* **Batch Processing:** Use scheduled batch predictions instead of real-time inference
* **Cloud-Based Solutions:** Leverage external computing resources for model training while maintaining on-premises inference

**Part 4: Reflection & Workflow Diagram (10 points)**

**1. Reflection (5 points)**

**Most Challenging Part:** The most challenging aspect was balancing technical accuracy with ethical considerations and regulatory compliance, particularly in the healthcare case study. Healthcare AI requires not just high performance but also fairness, interpretability, and strict privacy protection.

**Improvements with More Time/Resources:**

* **Longitudinal Studies:** Collect multi-year data to better understand temporal patterns
* **External Validation:** Test models across multiple hospitals to ensure generalizability
* **Clinical Trials:** Conduct randomized controlled trials to measure real-world impact
* **Stakeholder Engagement:** Involve more diverse clinical teams in model development and validation

**2. Workflow Diagram (5 points)**

AI Development Workflow

├── Problem Definition

│ ├── Identify Business Need

│ ├── Define Objectives & KPIs

│ └── Identify Stakeholders

├── Data Collection & Preprocessing

│ ├── Data Source Identification

│ ├── Data Quality Assessment

│ ├── Bias Detection & Mitigation

│ ├── Missing Data Handling

│ ├── Feature Engineering

│ └── Data Normalization

├── Model Development

│ ├── Algorithm Selection

│ ├── Data Splitting

│ ├── Hyperparameter Tuning

│ ├── Cross-Validation

│ └── Model Training

├── Evaluation & Validation

│ ├── Performance Metrics

│ ├── Bias Testing

│ ├── Clinical Validation

│ └── Regulatory Review

├── Deployment

│ ├── System Integration

│ ├── API Development

│ ├── User Interface Design

│ ├── Security Implementation

│ └── Compliance Verification

└── Monitoring & Maintenance

├── Performance Monitoring

├── Concept Drift Detection

├── Model Retraining

├── Feedback Integration

└── Continuous Improvement

**Key Feedback Loops:**

* Evaluation results inform model refinement
* Deployment challenges drive preprocessing improvements
* Monitoring insights trigger model updates
* Stakeholder feedback influences objective refinement