Part 1: Short Answer Questions (30 points)

1. Problem Definition (6 points)

Hypothetical AI Problem: Predicting patient no-shows in hospital appointments

Objectives

1. Accurately predict which patients are likely to miss appointments.
2. Optimize clinic scheduling to reduce unused time slots.
3. Improve patient engagement through targeted interventions.

Stakeholders

1. **Hospital Scheduling Staff**: Need predictive insights to optimize appointment booking and implement overbooking strategies
2. **Healthcare Providers (Physicians/Nurses)**: Require reliable scheduling to maximize patient contact time and maintain efficient clinical workflows

Key Performance Indicator (KPI):

No-Show Prediction Accuracy at 80% Recall: The model should identify 80% of actual no-show cases to enable effective intervention while maintaining reasonable precision to avoid excessive patient outreach

2. Data Collection & Preprocessing (8 points)

Identify **2 data sources** for your problem.

Electronic Health Records (EHR) and Patient Management Systems: Patient demographics, medical history, appointment history, insurance status, and contact information

External Data Sources: Weather data, public transportation schedules, local event calendars, and socioeconomic indicators that may influence appointment attendance

Explain **1 potential bias** in the data.

Bias toward frequently visiting patients: underrepresentation of first-time patients could hurt generalization.

Outline **3 preprocessing steps** (e.g., handling missing data, normalization).

1. Handle missing values in demographic or history fields using imputation
2. Normalize features like age, lead time before appointment
3. Encode categorical data like appointment type and insurance provider

3. Model Development (8 points)

1. Choose a model (e.g., Random Forest, Neural Network) and justify your choice.

Model Choice: Logistic Regression with Feature Engineering

Justification: Logistic Regression is ideal for patient no-show prediction because:

- Provides interpretable coefficients that healthcare staff can understand and trust
- Outputs probability scores that can be used for risk stratification and intervention prioritization
- Computationally efficient for real-time predictions during appointment scheduling
- Handles imbalanced datasets well with proper class weighting
- Allows for easy integration of domain knowledge through feature engineering
- Regulatory compliance friendly due to transparency in decision-making process

2. Describe how you would split data into training/validation/test sets.

Training Set (70%): Used for model learning and parameter estimation

Validation Set (15%): Used for hyperparameter tuning and feature selection during development

Test Set (15%): Reserved for final, unbiased performance evaluation

3. Name **2 hyperparameters** you would tune and why.

- **Regularization Parameter (C)**: Controls the strength of regularization to prevent overfitting; lower values increase regularization and help the model generalize better to new patients
- **Class Weight**: Addresses class imbalance between show/no-show cases; proper weighting ensures the model doesn't simply predict all patients will show up due to the typically imbalanced nature of no-show data

4. Evaluation & Deployment (8 points)

Select **2 evaluation metrics** and explain their relevance.

- **Recall (Sensitivity)**: Critical for identifying as many actual no-show cases as possible to maximize scheduling optimization opportunities and minimize wasted appointment slots
- **Precision**: Important for ensuring targeted interventions (reminder calls, texts) are sent to patients who are actually likely to miss appointments, avoiding patient annoyance from unnecessary communications

What is **concept drift**? How would you monitor it post-deployment?

Concept drift occurs when the statistical relationship between input data and target variables changes over time in a predictive model. This means that patterns learned during training may no longer be valid when the model is deployed, causing performance to degrade.

Monitoring Strategy:

- Weekly model performance reviews comparing predicted vs. actual no-show rates
- Automated alerts when no-show prediction accuracy drops below 75%
- Monthly retraining with recent appointment data
- Seasonal model adjustments for holiday periods and weather patterns

Describe **1 technical challenge** during deployment (e.g., scalability).

Real-time Integration with Scheduling Systems: Ensuring the prediction model can process appointment bookings in real-time while accessing multiple data sources (EHR, scheduling system, external APIs) without causing delays in the appointment booking workflow or system downtime during peak scheduling hours.

PART 2:

Hospital Readmission Prediction System - Case Study Analysis

1. Problem Scope (5 points)

Problem Definition

Develop an AI system to predict the likelihood of patient readmission within 30 days of discharge to enable proactive interventions, reduce healthcare costs, and improve patient outcomes.

Objectives

- Flag high-risk patients for additional follow-up care
- Improve patient outcomes and resource allocation
- Reduce avoidable readmissions and financial penalties

Stakeholders

- **Primary:** Hospital administrators, discharge planning teams, attending physicians
- **Secondary:** Nurses, case managers, quality improvement teams
- **Tertiary:** Patients and families, insurance companies, regulatory bodies
- **Technical:** IT department, data analysts, compliance officers

2. Data Strategy (10 points)

Proposed Data Sources

Electronic Health Records (EHRs)

- Patient demographics (age, gender, socioeconomic status)
- Medical history and comorbidities
- Current admission diagnosis and procedures
- Vital signs and laboratory results
- Medication history and discharge medications
- Length of stay and discharge disposition

Administrative Data

- Insurance type and coverage details
- Previous healthcare utilization patterns
- Emergency department visits in past 6 months

- Discharge location (home, skilled nursing, rehabilitation)

Social Determinants

- ZIP code for socioeconomic indicators
- Distance from healthcare facilities
- Social support system indicators
- Housing stability markers

Ethical Concerns:

1. **Patient privacy** – sensitive health information must be protected under legal frameworks like HIPAA.
2. **Algorithmic bias** – predictions could unfairly disadvantage groups based on race, income, or disability if trained on biased data.

Preprocessing Pipeline

Data Cleaning

1. **Missing Value Handling:** Impute missing values using median for numerical, mode for categorical
2. **Outlier Detection:** Apply IQR method for vital signs and lab values
3. **Data Validation:** Check for logical consistency (e.g., age vs. diagnosis compatibility)

Feature Engineering

1. **Temporal Features:** Days since last admission, time between procedures
2. **Aggregated Features:** Average lab values during stay, medication count
3. **Risk Scores:** Calculate existing clinical risk scores (Charlson Comorbidity Index)
4. **Interaction Features:** Age × comorbidity count, length of stay × diagnosis complexity
5. **Categorical Encoding:** One-hot encoding for diagnoses, ordinal encoding for severity levels

Normalization and Scaling

1. **Numerical Features:** StandardScaler for continuous variables
2. **Feature Selection:** Remove highly correlated features (correlation > 0.9)
3. **Dimensionality Reduction:** Apply PCA if feature count exceeds 100

3. Model Development (10 points)

Model Selection: Gradient Boosting (XGBoost)

Justification

- **Performance:** Excellent performance on tabular healthcare data with mixed feature types
- **Interpretability:** Provides feature importance scores crucial for clinical decision-making
- **Handles Missing Data:** Built-in capability to handle missing values common in EHR data
- **Non-linear Relationships:** Captures complex interactions between medical conditions
- **Proven Track Record:** Widely successful in healthcare prediction tasks

Hypothetical Performance Metrics

Confusion Matrix (1000 test patients)

Predicted

Actual	No Readmit	Readmit	Total
No Readmit	780	70	850
Readmit	45	105	150
Total	825	175	1000

Calculated Metrics

- **Precision:** 105/(105+70) = 0.60 (60%)
- **Recall:** 105/(105+45) = 0.70 (70%)
- **F1-Score:** 2×(0.60×0.70)/(0.60+0.70) = 0.646
- **Specificity:** 780/(780+70) = 0.918 (91.8%)

Clinical Interpretation

The model correctly identifies 70% of patients who will be readmitted while maintaining high specificity, minimizing false alarms that could lead to unnecessary interventions.

4. Deployment (10 points)

Integration Steps

Phase 1: Technical Integration

1. **API Development:** Create RESTful API for real-time predictions
2. **EHR Integration:** Develop HL7 FHIR-compliant interfaces
3. **Database Setup:** Establish secure data pipeline from EHR to prediction system
4. **User Interface:** Develop dashboard for clinicians showing risk scores and recommendations

Phase 2: Workflow Integration

1. **Clinical Workflow Mapping:** Integrate prediction into discharge planning process
2. **Alert System:** Configure risk-based alerts 24-48 hours before discharge
3. **Training Program:** Train clinical staff on system interpretation and usage
4. **Pilot Testing:** Deploy in select units with close monitoring

Phase 3: Full Deployment

1. **Gradual Rollout:** Expand to all units over 3-month period
2. **Performance Monitoring:** Continuous tracking of prediction accuracy
3. **Feedback Loop:** Collect clinician feedback for system improvements

HIPAA Compliance Measures

Technical Safeguards

- **Access Controls:** Role-based access with multi-factor authentication
- **Audit Logs:** Comprehensive logging of all data access and model predictions
- **Encryption:** End-to-end encryption for data in transit and at rest
- **Secure Development:** Regular security testing and vulnerability assessments

Administrative Safeguards

- **Business Associate Agreements:** Contracts with all third-party vendors
- **Staff Training:** Regular HIPAA training for all system users
- **Risk Assessment:** Annual security risk assessments
- **Incident Response:** Established breach notification procedures

Physical Safeguards

- **Server Security:** Restricted access to servers hosting the system
- **Workstation Controls:** Automatic screen locks and secure workstation setup
- **Media Controls:** Secure disposal of storage media containing PHI

5. Optimization (5 points)

Overfitting Mitigation:

- Apply **cross-validation with early stopping** during training to detect when the model begins to memorize rather than generalize.

Implementation Strategy

K-Fold Cross-Validation (k=5) with Temporal Splits

- Split data chronologically to prevent data leakage
- Use 5-fold cross-validation with early stopping based on validation loss

- Monitor both training and validation metrics during model training

Technical Details

- **Early Stopping Patience:** Stop training if validation AUC doesn't improve for 50 iterations
- **Regularization Parameters:**
    - L1 regularization (alpha = 0.1) to reduce feature complexity
    - L2 regularization (lambda = 1.0) to prevent weight magnitude explosion
- **Learning Rate Schedule:** Start with 0.1, reduce by factor of 0.5 if validation loss plateaus

PART 3

Critical Thinking Analysis: Ethics, Bias, and Trade-offs in Healthcare AI

Ethics & Bias (10 points)

- **Impact of Biased Data:**
  - May lead to unfair predictions against underrepresented groups (e.g., rural, low-income, minority patients).
  - Could reduce trust and worsen healthcare disparities if high-risk patients are overlooked.
- **Bias Mitigation Strategy:**
  - Conduct regular **bias audits** across demographic groups.
  - Apply **reweighing or fairness-aware algorithms** during training to ensure equitable treatment.

Trade-offs (10 points)

- **Interpretability vs Accuracy:**
  - Complex models (e.g., neural networks): higher accuracy, but low transparency.
  - Simple models (e.g., decision trees, logistic regression): more interpretable, but might lose predictive power.
  - In healthcare, interpretability is crucial for clinician trust and accountability.
- **Limited Computational Resources:**
  - May prevent use of deep learning or large ensemble models.
  - Favors **lightweight models** like XGBoost or Random Forests—efficient, interpretable, and deployable on standard servers.

Part 4: Reflection & AI Development Workflow

1. Reflection (5 points)

Most Challenging Part of the Workflow

The most challenging aspect was balancing ethical considerations with technical performance requirements, particularly in the context of healthcare where decisions have life-or-death consequences.

Ensuring predictions didn't reflect biases in patient demographics required careful analysis and adjustments.

It's challenging because fairness metrics often conflict with optimization goals like maximizing recall or precision.

Improvement with More Time/Resources:

- Include domain experts (e.g., clinicians, ethicists) in model evaluation to ensure medical relevance and ethical compliance.
- Conduct A/B testing with different feature sets and retraining frequency to improve adaptability and long-term performance.
- Implement continuous learning pipelines to update the model in real time as new data becomes available.

2. AI Development Workflow Diagram (5 points)

[Business Understanding]
↓
[Data Collection & Access]
↓
[Data Preprocessing]
(Cleaning, Feature Engineering)
↓
[Model Selection]
↓
[Model Training & Tuning]
↓
[Evaluation & Validation]
(Confusion Matrix, Precision, Recall)
↓
[Deployment & Monitoring]
↓
[Feedback & Optimization]