AI Development Workflow: Predicting 30-day Patient Readmission

Group Project Submission

Includes: Problem definition, data strategy, modeling, deployment plan, ethics and reflections.

Generated artifacts: training script, Flask serve script, trained model (rf_model.joblib), ROC curve image.

Part 1: Short Answer Questions - Problem Definition

Problem: Predict patient 30-day readmission risk after hospital discharge. Objectives: 1. Identify high-risk patients at discharge. 2. Reduce readmission rate via targeted interventions. 3. Provide interpretable risk scores to clinicians. Stakeholders: Clinicians; Hospital administration. KPI: AUC on holdout test set; operational KPI: reduction in 30-day readmission rate.

Part 1: Data Collection & Preprocessing

Data sources: 1) Electronic Health Records (EHR) 2) Demographics & administrative data. Potential bias: Selection/representation bias if data comes from a single hospital. Preprocessing steps: 1. Missing value handling (median imputation + missingness flags). 2. Feature encoding and normalization (one-hot, ordinal, scaling). 3. Time-based feature engineering and leakage control (use only data available at discharge).

Part 1: Model Development & Evaluation

Model choice: Random Forest or Gradient Boosting (LightGBM/XGBoost) for tabular clinical data. Data split: Temporal split recommended; otherwise stratified 70/15/15. Hyperparameters to tune: n_estimators, max_depth. Evaluation metrics: AUC-ROC; Precision & Recall (or Precision-Recall AUC). Concept drift monitoring: track metrics over time; use PSI/KL for feature distributions; retrain when performance degrades.

Part 2: Case Study - Problem Scope

Problem statement: Build an AI system to predict probability of readmission within 30 days at discharge. Objectives: risk score at discharge; prioritize patients; decrease readmission rate. Stakeholders: Care managers, physicians, quality teams, patients, payers.

Part 2: Data Strategy & Ethical Concerns

Data sources: EHR (admissions, labs, meds), administrative (demographics, insurance), historical utilization, social determinants. Ethical concerns: 1. Patient privacy and data protection (PHI handling, de-identification). 2. Bias & fairness: model may reinforce disparities via proxies. Preprocessing pipeline highlights: de-identify, time-window creation, feature engineering (prior admissions, lab summaries, discharge destination), imputation, encoding, temporal split.

Part 2: Model Development - Example

Model selected: Gradient Boosting (LightGBM/XGBoost) or RandomForest for portability. Hypothetical confusion matrix (example): TP=120, FP=80, FN=30, TN=770. Precision = 120/(120+80) = 0.60. Recall 120/(120+30) = 0.80. Interpretation: captures 80% of readmits but 40% false positive rate among flagged patients; trade-off depends on intervention cost.

Part 2: Deployment & Compliance

Integration steps: model packaging, serve as API, EHR integration, monitoring/logging, retraining pipeline, clinician UI with explanations (SHAP). Compliance: De-identify data for development, encrypt PHI in transit and at rest, BAAs with vendors, access controls and auditing to meet HIPAA requirements. Optimization to address overfitting: cross-validation + early stopping, regularization, feature selection, and gather more diverse training data.

Part 3: Ethics & Bias

Biased training data effects: Underrepresentation leads to underprediction of risk for some subgroups, worsening disparities. Mitigation: Audit performance across subgroups; apply reweighting, stratified training, and involve clinical/ethics stakeholders.

Part 3: Trade-offs

Interpretability vs accuracy: Healthcare often requires interpretability for trust. Prefer simpler interpretable models if accuracy gains from complex models are marginal. Use SHAP/LIME carefully. Limited compute: Choose lighter models (logistic regression, small RF), compress models, precompute features offline, or centralize serving.

Part 4: Reflection & Workflow Diagram

Reflection: The most challenging part is ensuring fairness and avoiding bias; it requires multidisciplinary review and continuous monitoring. Improvements with more resources: multi-site data collection, automated monitoring/retraining, clinician-in-the-loop validation, richer SDoH data.

Appendix - Files

Appendix: Generated files in the project folder: - data_simulation_and_train.py - serve_model.py - requirements.txt - README.md - rf_model.joblib (trained model) - roc_curve.png (ROC image) - AI_Development_Workflow_Readmission_Report.pdf (this report)