Assignment: Understanding the AI Development Workflow

Case Study: Maternal Health Risk Prediction

Author: Juliet Asiedu

Course: AI for Software Engineering

Part 1:

1. Problem Definition

Problem Statement:

Maternal health complications remain a major public health issue, especially in developing countries. The problem is to develop an AI-based system that predicts a woman's risk level during pregnancy based on vital signs and demographic data, enabling early interventions and reducing maternal mortality.

Objectives:

- To predict the risk category (low, medium, high) for pregnant women using clinical data.
- To support healthcare professionals in making informed and timely decisions.
- To help improve maternal outcomes through data-driven insights.

Stakeholders:

- Healthcare providers (doctors, midwives, nurses).
- Pregnant women receiving antenatal care.

Key Performance Indicator (KPI):

- Model accuracy (percentage of correct predictions on new maternal health data).

2. Data Collection & Preprocessing

Data Sources:

- Kaggle Maternal Health Risk Dataset (contains anonymized patient health data).
- Electronic Health Records (EHRs) from local hospitals or clinics.

Potential Bias:

If the dataset mostly includes urban hospital data, the model may underperform on rural populations where health conditions and monitoring differ. This could lead to biased predictions and unfair risk assessments.

Preprocessing Steps:

- **Handling Missing Data**: Replace missing values using mean or median imputation.
- **Normalization:** Scale features like blood pressure and temperature to a standard range for uniform model learning.
- **Encoding:** Convert categorical variables (e.g., education level or region) into numeric form for model compatibility.

3. Model Development

Chosen Model:

- Random Forest Classifier — chosen because it handles non-linear relationships well, reduces overfitting, and performs effectively on tabular health data.

Data Splitting Strategy:

- 70% Training set
- 15% Validation set
- 15% Test set

This ensures balanced evaluation and fine-tuning before final testing.

Hyperparameters to Tune:

n_estimators: Number of trees in the forest (controls performance and accuracy).

max_depth: Maximum tree depth (prevents overfitting).

4. Evaluation & Deployment

Evaluation Metrics:

Accuracy: Measures how many predictions are correct overall.

Recall: Focuses on identifying actual high-risk patients correctly — crucial in healthcare since false negatives can be dangerous.

Concept Drift:

- Concept drift occurs when the data patterns change over time (e.g., new health trends or improved antenatal practices).
- **Monitoring:** Regularly retrain the model with new hospital data and track performance metrics monthly.

Technical Deployment Challenge:

- **Scalability** — integrating the model into hospital systems that serve many users simultaneously may require cloud infrastructure or optimized APIs.

Part 2: Case Study Application

Problem Scope

Problem Statement:

The hospital faces frequent maternal readmissions due to postnatal complications such as infections, high blood pressure, or hemorrhage. The goal is to develop an AI model that predicts whether a mother is at high or low risk of readmission within 30 days of discharge.

Objectives:

- To use patient data (vital signs, delivery type, previous complications, etc.) to predict readmission risk.
- To enable timely interventions and reduce postnatal mortality and morbidity.
- To support hospital decision-making with data-driven insights.

Stakeholders:

- **Primary:** Doctors, midwives, and postnatal nurses.
- **Secondary:** Hospital administrators and patients (mothers).

Data Strategy

Data Sources:

- Electronic Health Records (EHRs): Containing patient vitals, delivery details, and discharge notes.
- Maternal Health Datasets (e.g., Kaggle or WHO open data): For model pretraining and benchmarking.

Ethical Concerns:

- **Patient Privacy:** Ensuring compliance with data protection laws by anonymizing patient identifiers.

- **Data Bias:** If historical data underrepresents certain populations (rural mothers or low-income patients), predictions may be unfair.

Preprocessing Pipeline:

- **Data Cleaning**: Remove duplicates, handle missing vitals using median imputation.

Feature Engineering:

- Compute BMI from height and weight.
- Create a “previous complications” binary flag.
- Generate “days since last visit” as a time-based feature.
- **Normalization:** Scale numerical variables (e.g., blood pressure, heart rate).
- **Encoding:** Convert categorical data like “delivery type” (C-section/Vaginal) into numeric format.

Model Development

Chosen Model:

- Logistic Regression — selected for interpretability and transparency in healthcare. It clearly shows how each factor (like high BP or infection) contributes to the risk score.

(Alternative models like Random Forest or XGBoost can later be tested for higher performance.)

Hypothetical Confusion Matrix:

	Predicted High Risk	**Predicted Low Risk**
Actual High Risk	80	20
Actual Low Risk	10	90

Precision & Recall:

- Precision = 80 / (80 + 10) = 0.89 (89%) → Of all predicted high-risk mothers, 89% were truly high risk.
- Recall = 80 / (80 + 20) = 0.80 (80%) → The model correctly identified 80% of actual high-risk mothers.

This balance is good for healthcare, where missing high-risk patients (false negatives) must be minimized.

Deployment

Integration Steps:

- Export the trained model as a .pkl or .onnx file.
- Develop an API endpoint (e.g., Flask or FastAPI) for hospital systems to send patient data and receive predictions.
- Connect API to the hospital's dashboard or mobile app for midwives.
- Set up automatic logging to track prediction trends and flag inconsistencies.

Regulatory Compliance:

- Ensure compliance with HIPAA (USA) or Ghana Health Data Protection Guidelines.
- Maintain audit trails, restrict access with user authentication, and encrypt sensitive data.

Optimization

Overfitting Solution:

- Apply k-fold cross-validation (e.g., 5-fold) to ensure the model generalizes well.
- Alternatively, use regularization (L2 penalty) in Logistic Regression to penalize overly complex patterns.

Part 3: Critical Thinking

1. Ethics & Bias

Biased data can lead to unfair predictions that disproportionately affect certain groups of mothers.

For example, if the dataset mainly contains records from urban hospitals, the AI may perform poorly on rural patients whose healthcare access and conditions differ. As a result, it might underpredict risks for these women, causing delayed interventions and higher complication rates.

Another issue is data imbalance — if fewer high-risk cases exist in the training data, the model may become insensitive to rare but serious conditions like postpartum hemorrhage.

Such biases not only reduce accuracy but also violate ethical standards of fairness and equality in maternal care.

Strategy to Mitigate Bias:

- **Data Auditing:** Before training, evaluate dataset distributions (age, region, socioeconomic status).
- **Rebalancing:** Use techniques like SMOTE (Synthetic Minority Oversampling) to balance risk categories.
- **Fairness Metrics:** Track measures like equal opportunity difference and demographic parity.
- **Transparency:** Use interpretable models (like Logistic Regression) and communicate limitations to medical staff.

2. Trade-offs

Question 1:

Discuss the trade-off between model interpretability and accuracy in healthcare.

Answer:

In healthcare, interpretability is often more important than raw accuracy.

A highly accurate but complex model (like a deep neural network) might be a "black box," making it difficult for doctors to understand why a patient is labeled high-risk.

On the other hand, interpretable models like Logistic Regression or Decision Trees might be slightly less accurate but allow clinicians to trust and explain predictions.

Therefore, the best approach is to start with interpretable models and gradually introduce more complex ones — only if they improve outcomes without sacrificing explainability.

Question 2:

If the hospital has limited computational resources, how might this impact model choice?

Answer:

Limited resources mean the hospital should use lightweight and efficient models that can run on standard hardware without cloud dependencies.

This makes Logistic Regression, Random Forest, or Gradient Boosted Trees preferable over deep neural networks, which require more memory and processing power.

Additionally, simpler models are easier to maintain, faster to retrain, and more practical in developing healthcare systems where technical infrastructure is limited.

Part 4: Reflection & Workflow Diagram

1. Reflection

The most challenging part of the workflow was data preprocessing and bias handling. The dataset had missing and unevenly distributed values, especially across age groups and risk levels. It took extra effort to clean and balance the data without losing important patterns. This stage taught me that even the most advanced AI models cannot perform well if the data foundation is weak.

With more time and resources, I would collaborate with hospitals to gather richer and more diverse maternal data. That would make the model fairer, more accurate, and applicable to real-world health settings.

2. AI Development Workflow Diagram

Problem Definition

↓

Data Collection

↓

Data Preprocessing

↓

Model Development

↓

Evaluation & Validation

↓

Deployment

↓

Monitoring & Improvement