- Ethics & Bias

How Might Biased Training Data Affect Patient Outcomes?
Biased training data can significantly impact the fairness and safety of AI in healthcare:

Underrepresentation of minority groups (e.g., racial, gender, or age-related bias) could result in inaccurate predictions for those patients.

For example, if older or low-income patients are underrepresented in the dataset, the model may underestimate their readmission risk, leading to delayed follow-up care or lack of preventive interventions.

This can widen healthcare disparities, reduce trust in AI, and potentially cause harm to vulnerable populations.

Strategy to Mitigate Bias:

Use Bias Detection and Fairness Constraints during Model Training

Tools like IBM AI Fairness 360 or Fairlearn can identify and reduce bias in model outputs.

Apply reweighting techniques or preprocessing adjustments to balance the training dataset across sensitive attributes like age, gender, or socioeconomic status.

Regularly audit model outcomes across different patient groups to ensure equitable performance.

- Trade-offs

Trade-off: Model Interpretability vs Accuracy
In healthcare, decisions must be explainable to doctors, patients, and regulators.

Highly accurate models like XGBoost or deep neural networks may act like black boxes, making it difficult to justify decisions (e.g., why a patient was marked high risk).

On the other hand, interpretable models like Logistic Regression or Decision Trees provide transparency but may sacrifice some predictive power.

Trade-off: Greater interpretability = easier to explain but possibly less accurate.
Greater accuracy = better predictions but harder to trust or regulate.

Impact of Limited Computational Resources
Hospitals with limited computing infrastructure may struggle to deploy complex models (e.g., deep learning).

This would encourage choosing lightweight, fast, and resource-efficient models such as:

Logistic Regression

Random Forests with limited depth

Gradient Boosting with tuned tree count and size

Such models can still perform well while being easier to deploy on hospital servers or low-latency environments.