**Future Work**

1. **Address Class Imbalance More Rigorously**
   * In our dataset, the ratio of survivors to non‐survivors was somewhat skewed. Although basic encoding and stratified splitting were used, applying methods such as SMOTE (Synthetic Minority Over‑sampling Technique) or class‐weight adjustments in neural network training could further improve recall for the minority class without sacrificing precision.
2. **Feature Engineering & Additional Clinical Variables**
   * The current dataset provides a limited set of features (age, gender, stage, family history, etc.). Adding more nuanced clinical variables—such as detailed smoking history (pack‑years), biomarker levels (e.g., EGFR mutations), comorbidities (e.g., COPD), or imaging features extracted via radiomics—could significantly enrich the model’s predictive capability.
3. **Hyperparameter Optimization**
   * While we used default hyperparameters for many models (e.g., n\_neighbors=5 for KNN, default tree depth for Random Forest), performing a systematic grid search, randomized search, or Bayesian optimization (e.g., using scikit‑optimize or Optuna) could reveal better settings. In particular, tuning:
     + **Random Forest**: number of trees, maximum depth, minimum samples per leaf.
     + **XGBoost**: learning rate (eta), maximum depth, subsample ratio, and colsample\_bytree.
     + **Neural Network**: number of hidden layers, neurons per layer, activation functions, dropout rates, and different optimizers (Adam, RMSprop, etc.).
4. **Cross‑Validation & Ensemble Stacking**
   * We relied on a single train/test split. To obtain more robust estimates of generalization performance, implementing k‑fold cross‑validation (e.g., 5‑fold or 10‑fold) would reduce variance in our metrics. Additionally, stacking multiple base learners (e.g., combining RF, XGBoost, and Neural Network) with a meta‐learner could leverage complementary strengths and yield even higher accuracy.
5. **Model Explainability & SHAP Analysis**
   * While feature importance from tree models gives a global sense of variable impact, using SHAP (SHapley Additive exPlanations) values would allow us to explain individual predictions in detail. This is especially important in a clinical context, where understanding why the model made a certain prediction for a patient can guide treatment planning.
6. **Integration with Clinical Workflow & Deployment**
   * To transition from a proof‐of‐concept to a usable tool, the final model should be packaged into a lightweight REST API (e.g., using FastAPI or Flask) or embedded in a simple web dashboard. This would allow clinicians to input patient features and receive a real‑time survival probability. Ensuring data privacy and compliance (e.g., HIPAA or GDPR) is critical for real‐world deployment.
7. **External Validation on Independent Cohorts**
   * Our models were trained and tested solely on a single publicly available dataset. Validating performance on independent datasets (e.g., another hospital’s lung cancer registry) would test generalizability. If performance degrades, domain adaptation techniques or transfer learning could be explored to recalibrate the model to the new cohort.
8. **Time‑to‑Event (Survival) Analysis**
   * The binary “survived vs. not survived” framework loses information about *when* an adverse event (death) occurred. Implementing a Cox proportional hazards model or a deep survival network (e.g., DeepSurv) would allow us to predict hazard functions or survival curves, which is more informative for clinicians and patients than a simple classification.
9. **Incorporate Imaging Data via Multimodal Learning**
   * Ultimately, lung cancer prognosis often depends on radiographic findings (CT scans, X‑rays). A future extension could build a multimodal pipeline in which tabular clinical data are combined with convolutional neural networks (CNNs) trained on imaging data. Early feasibility studies show that radiomic features from CT can boost classification performance beyond what clinical variables alone achieve.
10. **Longitudinal Monitoring & Continual Learning**
    * Patients’ status can change over time. Implementing a rolling prediction framework where the model is periodically retrained with new data (continual learning) can help capture evolving patterns—especially as new treatments emerge or diagnostic protocols change.