Summarized Model Development Pipeline for Bankruptcy Prediction

1. Choosing the Initial Models

- **Decision**: Use Logistic Regression, Random Forest, XGBoost; avoid unsupervised clustering.
    - Logistic Regression for interpretable baseline.
    - Random Forest for non-linear interactions and robustness.
    - XGBoost for high predictive power on noisy data.
    - Supervised models leverage available labels for risk probabilities.

2. Data Pre-processing

- **Decision**: Scale for LR, minimal scaling for trees, One-Hot Encode categorical, use Pipeline.
    - Scaling ensures LR stability.
    - Trees need no scaling to preserve data.
    - One-Hot Encoding maintains interpretability.
    - Pipeline prevents train/test transformation mismatch.

3. Handling Class Imbalance

- **Decision**: Use SMOTE, class_weight for LR, Stratified CV.
    - SMOTE boosts recall for rare bankrupt cases.
    - class_weight reduces compute needs for LR.
    - Stratified CV maintains class ratios.
    - Avoids leakage by applying SMOTE to training only.

4. Outlier Detection & Treatment

- **Decision**: Retain informative outliers, remove errors, Winsorize if needed, document removals.
    - Outliers may signal bankruptcy.
    - Errors (e.g., negative totals) harm model accuracy.
    - Winsorizing handles extreme sensor errors.
    - Documentation ensures transparency for audits.

5. Addressing Sampling Bias / PSI

- **Decision**: Monitor PSI, investigate if > 0.2, resample if needed.
    - PSI detects distribution shifts.
    - High PSI signals unreliable evaluation.
    - Resampling corrects biased data collection.
    - Ensures robust model performance.

6. Data Normalization

- **Decision**: Scale for LR, skip for trees, use ColumnTransformer, avoid double-scaling.
    - Scaling improves LR coefficient stability.
    - Trees are scale-invariant, preserving distributions.
    - ColumnTransformer applies scaling selectively.
    - Prevents train/test mismatch.

7. Testing for Normality

- **Decision**: Log-transform skewed ratios for LR, skip for trees, apply selectively.
    - Reduces skew for LR stability.
    - Trees unaffected by non-normality.
    - Selective transforms avoid unnecessary processing.
    - Improves LR calibration without overfitting.

8. Dimensionality Reduction (PCA)

- **Decision**: Avoid PCA by default, use only if overfitting persists.
    - Preserves feature explainability for finance.
    - Reduces overfitting risk in high dimensions.
    - PCA harms interpretability critical for audits.
    - Used only as experimental fallback.

9. Feature Engineering Choices

- **Decision**: Focus on financial ratios, prioritize domain-driven features, ensure SHAP compatibility.

    - Ratios are highly predictive.
    - Domain-driven features align with finance logic.
    - Limits arbitrary features to avoid overfitting.
    - SHAP ensures interpretable explanations.

10. Testing & Addressing Multicollinearity

- **Decision**: Use VIF, drop redundant features, consider PCA/L1 if needed.
    - VIF > 10 flags multicollinearity issues.
    - Redundant features distort LR coefficients.
    - Removal speeds up training, reduces noise.
    - PCA/L1 as fallback remedies.

11. Feature Selection Methods

- **Decision**: Correlation filtering, L1 for LR, SHAP for trees, prioritize domain features.
    - Drops highly correlated features ($r > 0.9$).
    - L1 shrinks irrelevant LR coefficients.
    - SHAP identifies low-impact tree features.
    - Domain features reduce overfitting risk.

12. Hyperparameter Tuning Methods

- **Decision**: RandomizedSearchCV for trees, GridSearchCV for LR, align with compute budget.
    - RandomizedSearchCV efficiently explores tree parameters.
    - GridSearchCV suits LR's smaller parameter space.
    - Compute alignment prevents resource waste.
    - CV variance detects overfitting.

13. Cross-Validation Strategy

- **Decision**: Use StratifiedKFold (k=5), shuffle, nested CV if compute allows.
    - StratifiedKFold preserves class balance.
    - Shuffling avoids order bias.

    - Nested CV ensures robust tuning.
    - Variance analysis ensures model stability.

14. Evaluation Metrics Selection

- **Decision**: Use ROC-AUC, Precision-Recall AUC, F1-score, Brier Score.
    - ROC-AUC evaluates overall ranking.
    - Precision-Recall AUC focuses on rare bankrupt cases.
    - F1-score balances precision and recall.
    - Brier Score checks probability calibration.

15. Evaluating Drift & Model Degradation

- **Decision**: Monitor PSI, retrain if > 0.2, track performance metrics.
    - PSI detects data drift over time.
    - Retraining addresses macroeconomic shifts.
    - Performance drop flags model issues.
    - Proactive monitoring ensures reliability.

16. Interpreting Model Results & Explainability

- **Decision**: Use SHAP, present top-K features, concise explanations.
    - SHAP provides clear local/global insights.
    - Top-K features meet audit needs.
    - Concise explanations aid decision-making.
    - Works for both linear and tree models.

17. Deployment & Retraining Decisions

- **Decision**: Retrain based on PSI > 0.2 or metric drops, use model registry, human-in-loop.
    - PSI/metric drops signal model degradation.
    - Registry tracks model versions, parameters.
    - Human oversight ensures reliable deployment.
    - Automation speeds up alerts.