Technical Report: Data Processing, Predicting, Sector-Specific Prediction

Assignment 2 GitHub repo

Borbala Kovacs, Muheng Li

February 15, 2026

1 Data Processing

1.1 Label Engineering

The construction of the `fast_growth` target variable follows a strict three-step logic to ensure economic relevance and statistical robustness.

1.1.1 The Measure (Input)

We define growth (g_i) as the **2-year logarithmic change in sales**.

- **Why Sales?** It is the most reliable "top-line" metric available (unlike profits, which can be negative for high-growth startups).
- **Why Log?** It treats growth symmetrically (e.g., doubling size is the mathematical inverse of halving size).
- **Why 2 Years?** To filter out short-term noise and seasonal spikes.

$$g_i = \ln(\text{Sales}_{2014}) - \ln(\text{Sales}_{2012})$$

1.1.2 The Threshold (Benchmark)

We benchmark growth only against **surviving firms**.

$$\tau = 75\text{th Percentile of } \{g_j \mid \text{Firm } j \text{ is active in 2014}\}$$

Logic: A firm is only "fast growth" if it outperforms 75% of the market survivors.

1.1.3 The Decision Logic (Final Label)

The binary target y_i is assigned based on performance relative to survivors and survival status itself.

$$y_i = \begin{cases} 1 & \text{if Status} = \text{Active AND } g_i > \tau \\ 0 & \text{if Status} = \text{Active AND } g_i \leq \tau \\ 0 & \text{if Status} = \text{Exit (Bankruptcy/Liquidation)} \end{cases} \quad (1)$$

Crucial Decision: Exiting firms are hard-coded as 0. This prevents survivorship bias by ensuring the model learns that "death" is the opposite of "growth."

1.2 Feature Engineering

To predict high-growth potential, we adapted the feature set from standard bankruptcy prediction models, hypothesizing that the same financial drivers (liquidity, profitability, leverage) would signal success rather than distress.

1.2.1 Financial Variables ("Hard" Data)

We engineered key financial ratios from the raw balance sheet and P&L data:

- **Size & Growth:** `ln_sales` (Log of Total Sales) and `sales_growth_lag1` (Previous year's growth). *Rationale:* Controls for the "base effect" (small firms grow faster) and momentum (growth begets growth).
- **Profitability:** `profit_margin` (Net Income / Sales) and `roa` (Return on Assets).
- **Liquidity & Leverage:** `curr_ratio` (Current Assets / Current Liab.) and `debt_equity`.

1.2.2 Management Quality ("Soft" Data)

We constructed proxy variables for human capital and governance structure:

- **Management Structure:** `ceo_count` (Number of Managing Directors) and `foreign_management` (Binary flag for international leadership).
- **Demographics:** `ceo_age` and `gender_diversity` (Flag for female presence in leadership).

1.2.3 Data Transformation Pipeline

To ensure model stability, the following transformations were applied:

- **Winsorization:** Financial ratios were winsorized at the 1st and 99th percentiles to cap extreme outliers common in SME data.
- **Imputation:** Missing numerical values were imputed with the yearly median. Categorical variables were imputed with a "missing" token to preserve information about data quality.
- **Encoding:** Categorical features (Industry, Region) were One-Hot Encoded to allow non-linear tree models to split effectively on specific sectors.

GitHub Link: 01_data_prep_fast_growth.ipynb

2 Predicting

2.1 Model Specification

To systematically evaluate the drivers of high-growth firms, we developed a hierarchy of five predictive models ranging from simple linear baselines to non-linear machine learning algorithms.

2.1.1 Linear Specifications (Logit)

We estimated three Logistic Regression models of increasing complexity to isolate the marginal contribution of different feature sets:

- **Model 1 (Baseline):** A parsimonious model relying solely on structural fundamentals—Firm Size ('ln_sales'), Momentum ('sales_growth'), Profitability ('profit_margin'), and Industry fixed effects. This serves as the benchmark for predictive performance.
- **Model 2 (Financials):** Expands the baseline by incorporating balance sheet health indicators, specifically liquidity ('curr_ratio') and leverage ('debt_equity'). It also introduces Firm Age and foreign management indicators to test for maturity and governance effects.
- **Model 3 (Full Interaction):** The most complex linear specification. It includes all variables from M2 plus HR characteristics ('ceo_count', 'gender_diversity') and squared terms/interactions (e.g., Age^2, $Size \times Age$) to capture non-linear returns to scale and maturity.

2.1.2 Machine Learning Benchmarks

To address potential overfitting in M3 and capture complex non-linearities, we employed two algorithmic approaches:

- **LASSO (L1 Regularization):** We applied Least Absolute Shrinkage and Selection Operator (LASSO) to the feature set of M3. *Rationale:* M3 contains many correlated interaction terms. LASSO automatically performs feature selection by shrinking coefficients of non-predictive variables to zero, enhancing out-of-sample stability.
- **Random Forest (RF):** A non-parametric ensemble method trained on the raw variable set. *Rationale:* Unlike Logit models, Random Forest naturally learns non-linearities and complex high-order interactions (e.g., specific growth patterns in young, high-debt manufacturing firms) without requiring manual feature engineering.

2.2 Model Selection

To determine the optimal predictive model, we evaluated five candidate specifications of increasing complexity using a 5-fold cross-validation framework. The selection process prioritized both calibration accuracy (Root Mean Squared Error) and discrimination ability (Area Under the Curve).

2.2.1 Evaluation Framework

We employed a 5-fold cross-validation (CV) strategy on the training set. For each fold:

- **RMSE (Root Mean Squared Error):** Used to assess the calibration of predicted probabilities. A lower RMSE indicates that the predicted probability of growth is closer to the true outcome (0 or 1).
- **AUC (Area Under the ROC Curve):** Used to assess the model's ability to rank firms correctly. A higher AUC indicates a better separation between high-growth and non-growth firms.

2.2.2 Selection Rationale

- **Linear Models (M1-M3):** We observed diminishing returns to complexity in the linear framework. While M3 offered a marginal improvement over the baseline (AUC 0.648 vs 0.635), the addition of complex interaction terms did not yield a breakthrough in predictive power.
- **LASSO:** The regularized LASSO model performed nearly identically to the full M3 logit model (AUC 0.649 vs 0.648). This suggests that while M3 is complex, it is not suffering from severe overfitting that regularization would correct.
- **Random Forest (Champion):** The Random Forest model significantly outperformed all linear specifications, achieving the highest AUC (0.671) and the lowest RMSE (0.372). This confirms that the relationship between firm characteristics (especially Age and Size) and high growth is non-linear and best captured by tree-based methods.

2.3 Loss Function

To align the predictive model with business objectives, we engineered a custom loss function that reflects the asymmetric costs of classification errors in the context of growth equity screening.
For task 1, we applied a loss function of FP = 1 and FN = 2. For task 2, we applied a loss function of FP = 1 and FN = 5.

2.3.1 Business Context

In early-stage investment and credit screening, the cost of errors is not symmetric:

- **Missed Opportunity (False Negative):** Failing to identify a high-growth firm is highly costly. It represents a lost "unicorn" or a missed major revenue stream.
- **Wasted Effort (False Positive):** Incorrectly flagging a non-growing firm incurs a processing cost (manual due diligence), but this cost is relatively low compared to the value of a missed deal.

2.3.2 Mathematical Definition

We formalized this trade-off into a loss function where the penalty for a False Negative (FN) is significantly higher than for a False Positive (FP).

Let y be the true class (1 = Fast Growth, 0 = Non-Growth) and $\hat{y}$ be the predicted class. The loss L is defined as:

$$L(\hat{y}, y) = \begin{cases} 2 & \text{if } y = 1 \land \hat{y} = 0 \quad \text{(False Negative)} \\ 1 & \text{if } y = 0 \land \hat{y} = 1 \quad \text{(False Positive)} \\ 0 & \text{if } y = \hat{y} \quad \text{(Correct Prediction)} \end{cases} \tag{2}$$

The total loss for a given probability threshold τ over a dataset of size N is:

$$\text{Total Loss}_\tau = \sum_{i=1}^{N} \left(2 \cdot I(y_i = 1, \hat{p}_i < \tau) + 1 \cdot I(y_i = 0, \hat{p}_i \geq \tau)\right) \tag{3}$$

2.4 Optimal Threshold Search (Cross-Validation)

To determine a robust decision boundary that generalizes well to unseen data, we employed a cross-validation sweeping strategy. Rather than fitting a single threshold to the aggregated training data, we estimated the optimal threshold τ^* as the average of the optimal cut-offs derived from each cross-validation fold.

2.4.1 Search Algorithm

The optimization process was executed as follows for the Random Forest model:

1. **Fold Iteration:** We performed 5-fold cross-validation. For each fold $k \in \{1, \ldots, 5\}$:
   - The model was trained on the training partition.
   - Probabilities $\hat{p}_k$ were predicted on the fold's validation (test) set.
2. **Threshold Sweep:** Within each fold, we swept through candidate thresholds $\tau \in [0, 1]$ (derived from the ROC curve operating points).
3. **Loss Minimization:** For each fold k, we identified the local optimal threshold τ_k^* that minimized the expected business loss:

$$\tau_k^* = \underset{\tau}{\operatorname{argmin}} \left(2 \cdot FN(\tau) + 1 \cdot FP(\tau)\right)_k \tag{4}$$

4. **Aggregation:** The final global threshold τ_{final} was calculated as the arithmetic mean of the fold-specific thresholds:

$$\tau_{final} = \frac{1}{K} \sum_{k=1}^{K} \tau_k^* \tag{5}$$

2.4.2 Rationale

This "average-across-folds" approach ensures stability. By selecting the threshold based on multiple independent validation sets, we mitigate the risk of the decision boundary being driven by outliers or specific distributional quirks present in a single data partition.
GitHub Link: 02_predicting_fast_growth.ipynb

3 Specific Sectors Predicting

This part details the methodology and implementation for predicting high-growth firms within two distinct industrial sectors: **Manufacturing** and **Services**. The objective is to deploy a Random Forest classifier optimized not just for accuracy, but for a specific business loss function that heavily penalizes missed opportunities.

3.1 Methodology

3.1.1 Sample Selection and Stratification

The dataset was filtered to isolate firms belonging to specific NACE industry codes:

- **Manufacturing:** Codes [26, 27, 28, 29, 30]

- **Services:** Codes [33, 55, 56]

Firms outside these categories were excluded to ensure the models learned sector-specific growth dynamics.

3.1.2 Asymmetric Loss Function

A custom loss function was defined to reflect the business reality where missing a "unicorn" (Fast Growth firm) is more costly than investigating a false lead.

$$Loss = 5 \times FN + 1 \times FP \tag{6}$$

Where FN is a False Negative (missed growth) and FP is a False Positive. This 5:1 ratio drives the optimization of the decision threshold.

3.1.3 Pipeline Architecture

To ensure rigor and prevent data leakage, we implemented a 'scikit-learn' Pipeline:

- **Dynamic Feature Handling:** The code automatically separates numerical and categorical columns for each sector.
- **Preprocessing:**
  - Numerical features are imputed with the median.
  - Categorical features are imputed with a constant and One-Hot Encoded (ignoring unknown categories to prevent test-time errors).
- **Model:** A Random Forest Classifier with 'class_weight='balanced'' to handle the rarity of high-growth firms.

3.2 Optimization Strategy

The training process utilized a two-stage optimization strategy to balance statistical robustness with business utility.

3.2.1 Stage 1: Hyperparameter Tuning (AUC)

We used 'GridSearchCV' with 5-fold Stratified Cross-Validation to tune the Random Forest parameters ('n_estimators', 'max_depth', 'min_samples_leaf').

- **Metric:** ROC-AUC.
- **Goal:** To find the model configuration that best *ranks* firms from high to low probability of growth, independent of any specific decision threshold.

3.2.2 Stage 2: Threshold Optimization (Business Loss)

Once the best model structure was found, we tuned the decision threshold τ.

- **Method:** We generated clean out-of-sample probabilities for the training set using 'cross_val_predict'.

- **Optimization:** We swept thresholds from 0.01 to 0.99 to find the τ^* that minimized the custom loss function ($5 \cdot FN + 1 \cdot FP$).
- **Rationale:** This ensures the threshold is tuned to the model's generalization behavior, not its training set memorization.

3.3 Implementation Code

The following snippet demonstrates the core logic used to separate sector data and execute the rigorous cross-validation loop.

```
# Loop through sectors to train separate models
for sector in ['Manufacturing', 'Services']:
    # 1. Dynamic Type Detection
    X = data_sector[features_to_use]
    num_cols = X.select_dtypes(include=['int64', 'float64']).columns
    cat_cols = X.select_dtypes(include=['object', 'category']).columns

    # 2. Build Pipeline
    preprocessor = ColumnTransformer(...)
    pipeline = Pipeline([('prep', preprocessor), ('rf', RandomForestClassifier())
    ])

    # 3. Optimize Hyperparameters (AUC)
    grid = GridSearchCV(pipeline, param_grid, cv=5, scoring='roc_auc')
    grid.fit(X_train, y_train)

    # 4. Optimize Threshold (Loss Function) via Cross-Validation
    y_probas = cross_val_predict(grid.best_estimator_, X_train, y_train, cv=5,
    method='predict_proba')
    best_thresh, min_loss = find_optimal_threshold_cv(y_train, y_probas[:, 1])
```

Listing 1: Sector-Specific Modeling Loop

3.4 Results and Interpretation

The analysis generates a "Detailed Output" for each sector, providing:

- **AUC Score:** Measuring the model's raw predictive power.
- **Optimal Threshold:** The probability cutoff that minimizes financial loss.
- **Average Loss:** The realized cost per firm in the test set.
- **Feature Importance:** Identifying whether "Hard" (Financial) or "Soft" (Management) variables drive growth in that sector.

GitHub Link: 03_sectors_predicting.ipynb