# JPMC MLCOE TSRL 2026 Internship Q1

**Application for lending department of a bank**

Molei Qin

Nanyang Technological University

molei001@e.ntu.edu.sg

Updated: January 18, 2026

This report constructs a very simple structural model of the balance sheet and income statement based on the tank-model ideas of Vélez-Pareja Vélez-Pareja 2011; Vélez-Pareja 2010 and the analytical treatment of circularity in discounted cash flow valuation Mejía-Peláez and Vélez-Pareja 2011. We define a low-dimensional state vector of financial statement items and a driver vector of policy and performance ratios. We derive explicit forward equations $y_t = f(y_{t-1}, x_t)$ governing the evolution of the state. We show that, under mild assumptions, the accounting identities—in particular the assets = liabilities + equity identity—are preserved automatically by the evolution equations. This allows us to treat the drivers $x_t$ as a multivariate time series and to shift the forecasting problem from financial statement items themselves to these drivers. We describe how to invert historical data to obtain "perfect" drivers, how to train very simple forecasting models (sliding-window mean, pooled AR(1) and a small multi-layer perceptron), and how to evaluate forecasting accuracy both in driver space and in state space. We also discuss how earnings are obtained naturally from the model and outline machine-learning extensions that could improve forecast performance.

## 1 Introduction and Literature Background

Forecasting full financial statements in an internally consistent way and free from ad hoc "plug" variables and circularity is an old problem in corporate finance and valuation. Vélez-Pareja Vélez-Pareja 2011; Vélez-Pareja 2010 proposes a "tank"-style approach in which key balance sheet items (cash, working capital, fixed assets, debt, equity) are treated as stocks (tanks) updated by flows that come from the income statement and cash-flow statement. The approach enforces accounting identities by construction and avoids circularity by carefully defining interest, taxes, and equity changes.

Mejía-Peláez and Vélez-Pareja Mejía-Peláez and Vélez-Pareja 2011 further analyse the circularity problem in discounted-cash-flow (DCF) valuation and provide an analytical solution that is compatible with such tank-style models. The key idea is that, once the stocks and flows are linked algebraically in a consistent way, there is no need for iterative numerical solutions: all relevant quantities can be computed in closed form.

In this report we build on these ideas and construct a simple but fully specified evolution model for a reduced balance sheet and income statement. We show how this model can be framed as a time-series problem by treating the drivers—growth rates, margins, working-capital days, capex ratios, tax rates, interest rates, payout ratios and net financing ratios—as a multivariate time series. The forecasting task is then shifted to predicting these drivers, while the deterministic

evolution equations take care of the accounting identities and yield consistent forward financial statements.

The remainder of the report is structured as follows. Section 2 defines the state and driver vectors and presents the forward equations. Section 3 proves that the accounting identities are automatically preserved. Section 4 shows how to invert historical data to obtain "perfect" drivers and explains the time-series framing. Section 5 describes training and testing strategies on a panel of companies and the evaluation metrics. Section 6 discusses earnings forecasting. Section 7 outlines possible machine-learning extensions. Section 10 concludes.

# 2 Model Specification: State, Drivers, and Forward Equations

## 2.1 Model-world assets, liabilities, and equity

We work in a simplified "model world" in which we only track a small set of balance sheet items explicitly. All other assets and liabilities (prepaids, other receivables, deferred items, etc.) are subsumed into a residual external equity flow. Concretely, for each year $t$ we define model assets, liabilities, and equity as

$$\mathrm{ASSETS}_t^{\mathrm{model}} = \mathrm{CASH}_t + \mathrm{AR}_t + \mathrm{INV}_t + \mathrm{PPE}_t, \tag{1}$$

$$\mathrm{LIAB}_t^{\mathrm{model}} = \mathrm{AP}_t + \mathrm{DEBT}_t, \tag{2}$$

$$\mathrm{EQ}_t^{\mathrm{model}} = \mathrm{ASSETS}_t^{\mathrm{model}} - \mathrm{LIAB}_t^{\mathrm{model}}. \tag{3}$$

Whenever we refer to equity $\mathrm{EQ}_t$ in what follows, we mean this model equity, either taken directly from preprocessed data or constructed using (1)–(3). Retained earnings $\mathrm{RE}_t$ are defined via a clean-surplus relation (see below).

## 2.2 State vector $y_t$

For each year $t$ we collect 15 key items into a state vector $y_t$:

$$y_t = (S_t, C_t, SG_t, D_t,\ \mathrm{AR}_t, \mathrm{INV}_t, \mathrm{AP}_t,\ \mathrm{PPE}_t,\ \mathrm{CASH}_t,\ \mathrm{DEBT}_t,\ \mathrm{EQ}_t,\ \mathrm{RE}_t,\ \mathrm{TAX}_t,\ \mathrm{INT}_t,\ \mathrm{DIV}_t), \tag{4}$$

where:

- Flows (income statement / cash-flow items) for year $t$:
  - $S_t$ : sales (revenue);
  - $C_t$ : cost of goods sold (COGS);
  - $SG_t$: operating expenses (selling, general and admin);
  - $D_t$ : depreciation expense;
  - $\mathrm{TAX}_t$: income tax expense;
  - $\mathrm{INT}_t$: interest expense;
  - $\mathrm{DIV}_t$: dividends paid (treated as positive cash outflow).
- Stocks (end-of-year balance sheet items):
  - $\mathrm{AR}_t$ : accounts receivable;

- $\mathrm{INV}_t$: inventory;
- $\mathrm{AP}_t$ : accounts payable;
- $\mathrm{PPE}_t$: net property, plant and equipment;
- $\mathrm{CASH}_t$: cash and cash equivalents;
- $\mathrm{DEBT}_t$: interest-bearing debt (short-term + long-term);
- $\mathrm{EQ}_t$: equity in model-world sense ((1)–(3));
- $\mathrm{RE}_t$: retained earnings (clean surplus).

All subsequent forward, inverse and evaluation steps operate on this 15 dimensional state representation, not on the raw reported totals.

### 2.3 Driver vector $x_t$

The driver vector $x_t$ collects the policy parameters that drive the evolution of the state:

$$x_t = (gS_t, gm_t, sga_t, dep_t,\ dso_t, dio_t, dpo_t,\ capex_t,\ \tau_t, r_t, pay_t,\ ndebt_t,\ nequity_t). \tag{5}$$

The components have the following economic meaning:

1. *Operating structure*
   - $gS_t$ : sales growth rate;
   - $gm_t$ : gross margin;
   - $sga_t$: operating expense ratio (Opex / Sales);
   - $dep_t$: depreciation rate (on beginning-of-period PPE).
2. *Working-capital policies*
   - $dso_t$ : days sales outstanding;
   - $dio_t$ : days inventory outstanding;
   - $dpo_t$ : days payables outstanding.
3. *Capex policy*
   - $capex_t$: capital expenditure / Sales.
4. *Tax, interest and payout*
   - $\tau_t$ : effective tax rate;
   - $r_t$ : interest rate on beginning-of-period debt;
   - $pay_t$ : payout ratio (dividends / net income).
5. *Financing decisions*
   - $ndebt_t$: net debt issuance / Sales; the change in debt is

$$\Delta\mathrm{DEBT}_t = ndebt_t\, S_t; \tag{6}$$

- $nequity_t$: net external equity inflow / Sales; the external equity flow (including all unmodelled balance-sheet items) is

$$\Delta \mathrm{EQ}_t^{\mathrm{ext}} = nequity_t \, S_t. \tag{7}$$

The key modelling choice, inspired by Vélez-Pareja Vélez-Pareja 2011; Vélez-Pareja 2010, is to fold all unmodelled assets and liabilities into $\Delta \mathrm{EQ}_t^{\mathrm{ext}}$. This ensures that the simplified model remains exactly closed with respect to the accounting identities: any residual is absorbed into the equity flow and, via the cash identity (below), into cash.

## 2.4 Forward evolution: from $y_{t-1}$ and $x_t$ to $y_t$

Given the previous state $y_{t-1}$ and the driver vector $x_t$, the model evolves one year forward to $y_t$ through deterministic equations.

### 2.4.1 Operating block: sales, costs, expenses, depreciation

Sales grow according to the sales growth driver:

$$S_t = S_{t-1}(1 + gS_t). \tag{8}$$

Costs, SG&A, and depreciation are driven by ratios:

$$C_t = (1 - gm_t)\, S_t, \tag{9}$$
$$SG_t = sga_t \, S_t, \tag{10}$$
$$D_t = dep_t \, \mathrm{PPE}_{t-1}. \tag{11}$$

### 2.4.2 Working capital: AR, inventory, AP

Working-capital stocks are tied directly to sales and costs via turnover days:

$$\mathrm{AR}_t = \frac{dso_t}{365} \, S_t, \tag{12}$$
$$\mathrm{INV}_t = \frac{dio_t}{365} \, C_t, \tag{13}$$
$$\mathrm{AP}_t = \frac{dpo_t}{365} \, C_t. \tag{14}$$

Define simplified working capital and its change:

$$\mathrm{WC}_t = \mathrm{AR}_t + \mathrm{INV}_t - \mathrm{AP}_t, \tag{15}$$
$$\Delta \mathrm{WC}_t = \mathrm{WC}_t - \mathrm{WC}_{t-1}. \tag{16}$$

### 2.4.3 Capex and PPE

Capital expenditure is driven by the capex-to-sales ratio:

$$\mathrm{CAPEX}_t = capex_t \, S_t. \tag{17}$$

PPE evolves as

$$\mathrm{PPE}_t = \mathrm{PPE}_{t-1} + \mathrm{CAPEX}_t - D_t. \tag{18}$$

### 2.4.4 Income statement: interest, tax, net income, dividends

Earnings before interest and taxes:

$$\mathrm{EBIT}_t = S_t - C_t - SG_t - D_t. \tag{19}$$

Interest and pre-tax income:

$$\mathrm{INT}_t = r_t \, \mathrm{DEBT}_{t-1}, \tag{20}$$
$$\mathrm{EBT}_t = \mathrm{EBIT}_t - \mathrm{INT}_t. \tag{21}$$

Taxes (with a floor at zero taxable income):

$$\mathrm{TAX}_t = \begin{cases} \tau_t \, \mathrm{EBT}_t, & \text{if } \mathrm{EBT}_t > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

Net income and dividends:

$$\mathrm{NI}_t = \mathrm{EBT}_t - \mathrm{TAX}_t, \tag{23}$$
$$\mathrm{DIV}_t = pay_t \, \max(\mathrm{NI}_t, 0). \tag{24}$$

Note that $\mathrm{NI}_t$ is not explicitly part of the state vector (4); it is a derived flow computed from other state and driver variables.

### 2.4.5 Free cash flow

Free cash flow to the firm is defined as

$$\mathrm{FCF}_t = \mathrm{NI}_t + D_t - \Delta\mathrm{WC}_t - \mathrm{CAPEX}_t. \tag{25}$$

### 2.4.6 Financing and stock updates: debt, equity, cash, retained earnings

**Debt.** Net debt issuance is given by (6). Debt evolves as

$$\mathrm{DEBT}_t = \mathrm{DEBT}_{t-1} + \Delta\mathrm{DEBT}_t. \tag{26}$$

**Retained earnings (clean surplus).** We impose clean surplus on retained earnings:

$$\mathrm{RE}_t = \mathrm{RE}_{t-1} + \mathrm{NI}_t - \mathrm{DIV}_t. \tag{27}$$

**External equity flows and total equity.** The external equity flow $\Delta\mathrm{EQ}_t^{\mathrm{ext}}$ is defined in (7). Total equity evolves as

$$\mathrm{EQ}_t = \mathrm{EQ}_{t-1} + \Delta\mathrm{EQ}_t^{\mathrm{ext}} + \mathrm{NI}_t - \mathrm{DIV}_t. \tag{28}$$

By construction, $\Delta\mathrm{EQ}_t^{\mathrm{ext}}$ absorbs both genuine external equity transactions (issuance, buybacks) and the effect of all unmodelled balance sheet items.

**Cash (core cash identity).** Finally, cash is updated via the core cash identity:

$$\mathrm{CASH}_t = \mathrm{CASH}_{t-1} + \mathrm{FCF}_t + \Delta\mathrm{DEBT}_t + \Delta\mathrm{EQ}_t^{\mathrm{ext}} - \mathrm{DIV}_t. \tag{29}$$

Intuitively, free cash flow accumulates in cash unless it is absorbed by net debt changes, net external equity flows, or cash dividends.

#### 2.4.7 Summary: the evolution function $f$

Equations (8)–(29) define a deterministic mapping

$$y_t = f(y_{t-1}, x_t), \tag{30}$$

which we will refer to as the *tank evolution* or *forward* function. In what follows we treat the choice of drivers $x_t$ as the main source of uncertainty and model it with time-series techniques.

## 3 Accounting Identities and Their Preservation

### 3.1 Definitions

In the simplified model world, assets, liabilities and equity are defined by (1)–(3):

$$\begin{aligned} A_t &:= \mathrm{ASSETS}_t^{\mathrm{model}} = \mathrm{CASH}_t + \mathrm{AR}_t + \mathrm{INV}_t + \mathrm{PPE}_t, \\ L_t &:= \mathrm{LIAB}_t^{\mathrm{model}} = \mathrm{AP}_t + \mathrm{DEBT}_t, \\ E_t &:= \mathrm{EQ}_t^{\mathrm{model}} = A_t - L_t. \end{aligned}$$

The fundamental accounting identity is therefore

$$A_t - L_t - E_t = 0 \quad \text{for all } t. \tag{31}$$

We now show that if this identity holds at time $t-1$ and at the initial time 0, then the evolution equations in Section 2 guarantee that it holds for all future periods.

### 3.2 Proof of identity preservation

**Step 1: Express the change in $A_t - L_t$.** Consider the change in "net assets" $A_t - L_t$ between $t-1$ and $t$:

$$\begin{aligned} (A_t - L_t) - (A_{t-1} - L_{t-1}) = (\mathrm{CASH}_t - \mathrm{CASH}_{t-1}) + (\mathrm{AR}_t - \mathrm{AR}_{t-1}) + (\mathrm{INV}_t - \mathrm{INV}_{t-1}) \\ + (\mathrm{PPE}_t - \mathrm{PPE}_{t-1}) - (\mathrm{AP}_t - \mathrm{AP}_{t-1}) - (\mathrm{DEBT}_t - \mathrm{DEBT}_{t-1}). \end{aligned} \tag{32}$$

Using the definition of working capital (15) and its change (16), we can rewrite

$$(\mathrm{AR}_t - \mathrm{AR}_{t-1}) + (\mathrm{INV}_t - \mathrm{INV}_{t-1}) - (\mathrm{AP}_t - \mathrm{AP}_{t-1}) = \Delta\mathrm{WC}_t. \tag{33}$$

Using the PPE and debt evolution equations (18) and (26), we have

$$\mathrm{PPE}_t - \mathrm{PPE}_{t-1} = \mathrm{CAPEX}_t - D_t, \tag{34}$$

$$\mathrm{DEBT}_t - \mathrm{DEBT}_{t-1} = \Delta\mathrm{DEBT}_t. \tag{35}$$

Substituting (33)–(35) into (32) yields

$$(A_t - L_t) - (A_{t-1} - L_{t-1}) = (\mathrm{CASH}_t - \mathrm{CASH}_{t-1}) + \Delta\mathrm{WC}_t + \mathrm{CAPEX}_t - D_t - \Delta\mathrm{DEBT}_t. \tag{36}$$

Using the cash identity (29), we have

$$\mathrm{CASH}_t - \mathrm{CASH}_{t-1} = \mathrm{FCF}_t + \Delta\mathrm{DEBT}_t + \Delta\mathrm{EQ}_t^{\mathrm{ext}} - \mathrm{DIV}_t. \tag{37}$$

Substituting (37) into (36) gives

$$\begin{aligned}(A_t - L_t) - (A_{t-1} - L_{t-1}) &= (\mathrm{FCF}_t + \Delta\mathrm{DEBT}_t + \Delta\mathrm{EQ}_t^{\mathrm{ext}} - \mathrm{DIV}_t) \qquad (38)\\ &\quad + \Delta\mathrm{WC}_t + \mathrm{CAPEX}_t - D_t - \Delta\mathrm{DEBT}_t \\ &= \mathrm{FCF}_t + \Delta\mathrm{EQ}_t^{\mathrm{ext}} - \mathrm{DIV}_t + \Delta\mathrm{WC}_t + \mathrm{CAPEX}_t - D_t. \qquad (39)\end{aligned}$$

Using the definition of free cash flow (25) we have

$$\mathrm{FCF}_t = \mathrm{NI}_t + D_t - \Delta\mathrm{WC}_t - \mathrm{CAPEX}_t. \tag{40}$$

Substituting (40) into (39) yields

$$\begin{aligned}(A_t - L_t) - (A_{t-1} - L_{t-1}) &= (\mathrm{NI}_t + D_t - \Delta\mathrm{WC}_t - \mathrm{CAPEX}_t) + \Delta\mathrm{EQ}_t^{\mathrm{ext}} - \mathrm{DIV}_t \\ &\quad + \Delta\mathrm{WC}_t + \mathrm{CAPEX}_t - D_t \\ &= \mathrm{NI}_t + \Delta\mathrm{EQ}_t^{\mathrm{ext}} - \mathrm{DIV}_t. \qquad (41)\end{aligned}$$

**Step 2: Compare with the change in equity.** From the equity evolution equation (28) we have

$$E_t - E_{t-1} = \Delta\mathrm{EQ}_t^{\mathrm{ext}} + \mathrm{NI}_t - \mathrm{DIV}_t. \tag{42}$$

Comparing (41) and (42), we see that

$$(A_t - L_t) - (A_{t-1} - L_{t-1}) = (E_t - E_{t-1}). \tag{43}$$

Equivalently,

$$(A_t - L_t - E_t) - (A_{t-1} - L_{t-1} - E_{t-1}) = 0. \tag{44}$$

Thus the quantity $A_t - L_t - E_t$ is *invariant* over time. If the identity (31) holds at $t-1$, then it also holds at $t$.

**Step 3: Induction over time.** Assume that at the initial time $t = 0$ we have

$$A_0 - L_0 - E_0 = 0. \tag{45}$$

By (44), if the identity holds at $t = k-1$ then it holds at $t = k$. Therefore, by induction, (31) holds for all $t \geq 0$.

**Conclusion.** We have shown that the evolution equations (8)–(29) preserve the fundamental accounting identity $A_t = L_t + E_t$ for all $t$, provided it holds at the initial time and the initial state is consistent with the model-world balance sheet. In particular, any forecast produced by the model automatically respects assets = liabilities + equity, without any ad hoc plugs or rebalancing rules. This is directly in line with the "no plugs, no circularity" principle advocated by Vélez-Pareja Vélez-Pareja 2011.

# 4 Time-Series Interpretation and Perfect Drivers

## 4.1 From historical states to perfect drivers

In historical data we observe the realised states $y_t^{\text{data}}$ for $t = 0, \dots, T$ and the associated income statement and cash-flow flows (including net income and dividends). Given a pair of consecutive states $y_{t-1}^{\text{data}}$ and $y_t^{\text{data}}$, we can *invert* the forward equations and recover a unique driver vector $x_t^*$ that exactly reproduces the transition $t-1 \to t$.

Ignoring small numerical tolerances, the inverse ("perfect") drivers are:

$$gS_t^* = \frac{S_t - S_{t-1}}{\max(S_{t-1}, \varepsilon)}, \tag{46}$$

$$gm_t^* = 1 - \frac{C_t}{\max(S_t, \varepsilon)}, \tag{47}$$

$$sga_t^* = \frac{SG_t}{\max(S_t, \varepsilon)}, \tag{48}$$

$$dep_t^* = \frac{D_t}{\max(\text{PPE}_{t-1}, \varepsilon)}, \tag{49}$$

$$dso_t^* = 365\,\frac{\text{AR}_t}{\max(S_t, \varepsilon)}, \tag{50}$$

$$dio_t^* = 365\,\frac{\text{INV}_t}{\max(C_t, \varepsilon)}, \tag{51}$$

$$dpo_t^* = 365\,\frac{\text{AP}_t}{\max(C_t, \varepsilon)}, \tag{52}$$

$$capex_t^* = \frac{\text{CAPEX}_t^{\text{data}}}{\max(S_t, \varepsilon)}, \quad \text{CAPEX}_t^{\text{data}} = \text{PPE}_t - \text{PPE}_{t-1} + D_t, \tag{53}$$

and, using the reconstructed EBIT, EBT and net income,

$$r_t^* = \frac{\text{INT}_t}{\max(\text{DEBT}_{t-1}, \varepsilon)}, \tag{54}$$

$$\tau_t^* = \begin{cases} \dfrac{\text{TAX}_t}{\max(\text{EBT}_t^{\text{data}}, \varepsilon)}, & \text{if } \text{EBT}_t^{\text{data}} > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{55}$$

$$pay_t^* = \frac{\text{DIV}_t}{\max(\text{NI}_t^{\text{data}}, \varepsilon)}. \tag{56}$$

The net financing drivers are recovered from the changes in debt and equity:

$$\Delta\text{DEBT}_t^{\text{data}} = \text{DEBT}_t - \text{DEBT}_{t-1}, \tag{57}$$

$$ndebt_t^* = \frac{\Delta\text{DEBT}_t^{\text{data}}}{\max(S_t, \varepsilon)}, \tag{58}$$

and, using clean surplus on equity,

$$\Delta\text{EQ}_t^{\text{ext,data}} = \text{EQ}_t - \text{EQ}_{t-1} - (\text{NI}_t^{\text{data}} - \text{DIV}_t), \tag{59}$$

$$nequity_t^* = \frac{\Delta\text{EQ}_t^{\text{ext,data}}}{\max(S_t, \varepsilon)}. \tag{60}$$

Here $\varepsilon$ is a small positive constant (e.g. $10^{-6}$) to avoid division by zero.

By construction, if we plug $x_t^*$ into the forward map (30) with $y_{t-1}^{\text{data}}$ as the initial state, we recover $y_t^{\text{data}}$ up to numerical round-off. In this sense $x_t^*$ is the "perfect" driver: it captures, within the model structure, the actual policies and conditions that transformed $y_{t-1}$ into $y_t$.

### 4.2 Time-series view: modelling drivers instead of states

The crucial modelling choice is to treat the drivers $x_t$ as a multivariate time series and to leave the evolution of $y_t$ to the deterministic structural map $f$. Conceptually, we proceed in two steps:

1. **Data construction.** For each firm and each year $t$ with a following year $t+1$, we construct $y_t^{\text{data}}$ and $y_{t+1}^{\text{data}}$, and then invert to obtain the perfect driver $x_{t+1}^*$ for that transition. This yields a panel of driver sequences $\{x_t^*\}$, typically of length three for each firm (four years of data).

2. **Time-series modelling.** We then fit simple time-series models to map past drivers to current drivers, e.g. $x_t^*$ as a function of $x_{t-1}^*$ and possibly longer lags. The forecasting models live in driver space; the state space evolution is handled by $f$.

In other words, we do *not* try to black-box fit $y_{t+1}$ directly as a function of $y_t$. Instead we use the structural accounting model to tell us how $y_{t+1}$ depends on $x_t$, and reduce the learning problem to predicting $x_t$.

Because each firm typically has only four years of data, each firm's own driver sequence has length three. This is too short for sophisticated per-firm time-series models such as ARIMA. Therefore, in the empirical part we focus on:

- per-firm sliding-window mean predictors;
- a pooled, component-wise AR(1) model using data from all firms;
- a small pooled neural network (multi-layer perceptron).

## 5 Training and Testing on a Panel of Firms

### 5.1 Sample of companies

To apply the model empirically, we select a panel of companies with at least four consecutive years of annual financial statements (income statement, balance sheet, cash-flow statement). For each firm we preprocess the raw statements into the model-world state representation $y_t^{\text{data}}$ as in Section 2, ensuring that:

- equity $\text{EQ}_t$ is computed as $\text{CASH}_t + \text{AR}_t + \text{INV}_t + \text{PPE}_t - \text{AP}_t - \text{DEBT}_t$;
- retained earnings obey the clean-surplus relation $\text{RE}_t = \text{RE}_{t-1} + \text{NI}_t - \text{DIV}_t$.

The concrete list of firms (e.g. large industrial and consumer companies) can be chosen to match data availability and project requirements.

### 5.2 Train–test structure of driver sequences

For each firm with four years of data $t = 0, 1, 2, 3$ we obtain three perfect driver vectors:

$$x_1^*, x_2^*, x_3^*,$$

corresponding to the transitions $0 \to 1$, $1 \to 2$ and $2 \to 3$ respectively. To create a simple train–test split per firm, we treat the last transition as test and the earlier ones as train:

- training transitions: $0 \to 1$ and $1 \to 2$;
- test transition: $2 \to 3$.

Pooling across firms yields a training set of driver transitions and a test set of driver transitions, each associated with the corresponding state $y_t$ that serves as the starting point for forward simulation.

## 5.3 Driver forecasting models

We briefly describe the three simple forecasting models used for the drivers.

### 5.3.1 Per-firm sliding-window mean

For a given firm and year $t$, the sliding-window mean predictor with window length $k$ sets the forecast driver vector to the average of the past $k$ observed perfect drivers:

$$\hat{x}_t^{(\mathrm{SW})} = \frac{1}{k} \sum_{i=1}^{k} x_{t-i}^{*}. \tag{61}$$

If the firm has fewer than $k$ past drivers, we average over all available ones. For the typical four-year case, we can use $k = 2$. This model is purely per-firm and does not pool information across firms.

### 5.3.2 Pooled AR(1) model

For each driver component $j$ we fit a pooled autoregressive model of order one (AR(1)) across all firms:

$$x_t^{(j)} = a^{(j)} + \phi^{(j)} x_{t-1}^{(j)} + \varepsilon_t^{(j)}. \tag{62}$$

Here:

- $x_t^{(j)}$ is the $j$-th component of $x_t^{*}$,
- $(a^{(j)}, \phi^{(j)})$ are parameters estimated via ordinary least squares using all training pairs $(x_{t-1}^{(j)}, x_t^{(j)})$ such that the target time $t$ belongs to the training set.

The OLS estimates can be expressed as

$$\phi^{(j)} = \frac{\mathrm{Cov}(x_{t-1}^{(j)}, x_t^{(j)})}{\mathrm{Var}(x_{t-1}^{(j)})}, \tag{63}$$

$$a^{(j)} = \mathbb{E}[x_t^{(j)}] - \phi^{(j)}\, \mathbb{E}[x_{t-1}^{(j)}]. \tag{64}$$

The AR(1) forecast is then

$$\hat{x}_t^{(j,\mathrm{AR1})} = a^{(j)} + \phi^{(j)} x_{t-1}^{(j)}. \tag{65}$$

Because we pool all firms, we obtain more stable parameter estimates than if we tried to fit AR(1) per firm.

### 5.3.3 Pooled multi-layer perceptron (MLP)

Finally, we consider a small neural network that maps the previous-period driver vector to the current-period driver vector:

$$\hat{x}_t^{(\mathrm{NN})} = g_\theta(x_{t-1}^*), \tag{66}$$

where $g_\theta$ is a feed-forward network with, for example, two hidden layers of size 32 and ReLU activations. The network is trained on all training samples to minimise the mean squared error (MSE) between $\hat{x}_t^{(\mathrm{NN})}$ and $x_t^*$:

$$\min_\theta \frac{1}{N} \sum_{i=1}^{N} \left\| g_\theta(x_{t_i-1}^*) - x_{t_i}^* \right\|_2^2. \tag{67}$$

In practice we standardise drivers component-wise (zero mean, unit variance) on the training set, train the network in standardised space, and then transform forecasts back to the original scale.

## 5.4 From driver forecasts to state forecasts

Given a forecast driver vector $\hat{x}_t$ (from any of the above models) and the current state $y_t^{\mathrm{data}}$, we obtain a forecast for next year's state via the structural evolution map:

$$\hat{y}_{t+1} = f(y_t^{\mathrm{data}}, \hat{x}_t). \tag{68}$$

By Section 3, any such forecast automatically respects the accounting identities, in particular

$$\mathrm{CASH}_{t+1} + \mathrm{AR}_{t+1} + \mathrm{INV}_{t+1} + \mathrm{PPE}_{t+1} = \mathrm{AP}_{t+1} + \mathrm{DEBT}_{t+1} + \mathrm{EQ}_{t+1}. \tag{69}$$

## 5.5 Evaluation metrics

We evaluate models at two levels:

1. **Driver-space metrics**, comparing $\hat{x}_t$ to $x_t^*$;
2. **State-space metrics**, comparing $\hat{y}_{t+1}$ to $y_{t+1}^{\mathrm{data}}$.

For both we use:

- Mean squared error (MSE):

$$\mathrm{MSE} = \frac{1}{Nd} \sum_{i=1}^{N} \sum_{j=1}^{d} \left( \hat{z}_{i,j} - z_{i,j}^{\mathrm{true}} \right)^2;$$

- Mean absolute error (MAE):

$$\mathrm{MAE} = \frac{1}{Nd} \sum_{i=1}^{N} \sum_{j=1}^{d} \left| \hat{z}_{i,j} - z_{i,j}^{\mathrm{true}} \right|;$$

- Relative L1 and L2 errors (to account for scale differences across firms and items), for example,

$$\mathrm{RelL1} = \frac{1}{Nd} \sum_{i=1}^{N} \sum_{j=1}^{d} \frac{\left| \hat{z}_{i,j} - z_{i,j}^{\mathrm{true}} \right|}{\max\left( |z_{i,j}^{\mathrm{true}}|, \varepsilon \right)}.$$

Here $z$ denotes either drivers or states, $d$ is the dimensionality (13 for drivers, 15 for states), $N$ is the number of samples, and $\varepsilon$ is a small constant for numerical stability.

### 5.6 Testing plan and sanity checks

**Perfect-driver baseline.** As a sanity check on both the forward and inverse implementations, we use the perfect drivers $x_t^*$ as inputs to the forward map and verify that

$$\hat{y}_{t+1}^{(\text{perfect})} = f(y_t^{\text{data}}, x_t^*)$$

reconstructs $y_{t+1}^{\text{data}}$ to machine precision. In practice, the MSE and MAE for this baseline should be on the order of floating-point round-off. This test verifies:

- correctness of the evolution equations;
- correctness of the inversion formulas for $x_t^*$;
- that the data preprocessing step yields internally consistent states.

**Forecasting models.** For each of the three forecasting models (SW, AR(1), MLP) we:

1. Train the model on the training driver transitions;
2. Generate driver forecasts $\hat{x}_t$ for each test transition;
3. Propagate these through the forward map to obtain state forecasts $\hat{y}_{t+1}$;
4. Compute the driver- and state-space metrics described above;
5. Inspect which state components are hardest to forecast (e.g. cash and debt may show larger errors).

Because all models use the same structural evolution, any improvement in state-space metrics can be attributed directly to better driver forecasts.

**Accounting consistency of forecasts.** Because the accounting identities are preserved by construction (Section 3), we do not need to impose additional constraints at forecast time. Nonetheless, as a diagnostic, we can compute the residual

$$\Delta_t^{\text{identity}} = (\text{CASH}_t + \text{AR}_t + \text{INV}_t + \text{PPE}_t) - (\text{AP}_t + \text{DEBT}_t + \text{EQ}_t)$$

on both historical and forecasted states to confirm that it is numerically zero up to round-off.

**Testing result demonstration.** We randomly select some of the companies to fit through 4 models and demonstrate both the drivers and states recover situations about Relative L1.

Across all four evaluation settings (Quarter/Year $\times$ Driver/State), the recovered error patterns exhibit several consistent and interpretable characteristics. First, the *sliding-window* estimator uniformly achieves the lowest log-scaled reconstruction error, outperforming both MLP- and AR(1)-based regressors in every figure (e.g., see Fig. 4 and Fig. 1). This dominance suggests that the local-stationarity assumption embedded in the sliding scheme aligns more closely with the statistical structure of accounting drivers. Second, the MLP model does not provide meaningful gains over a simple AR(1) baseline; their bars are nearly indistinguishable across all tickers, indicating that the driver dynamics are largely linear and offer limited exploitable nonlinearity

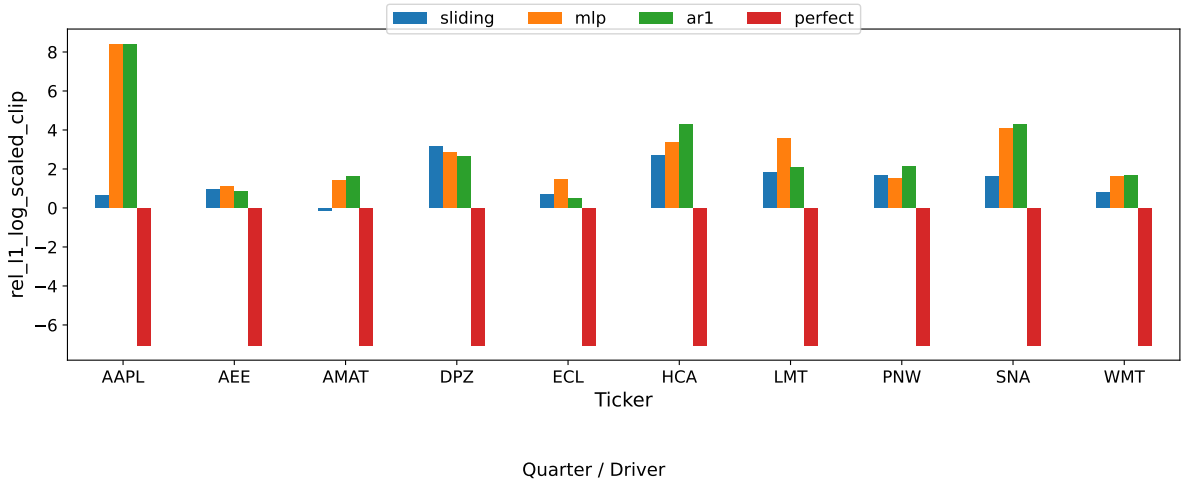


**Figure 1:** Relative L1 for driver recovery for quarter balance sheet

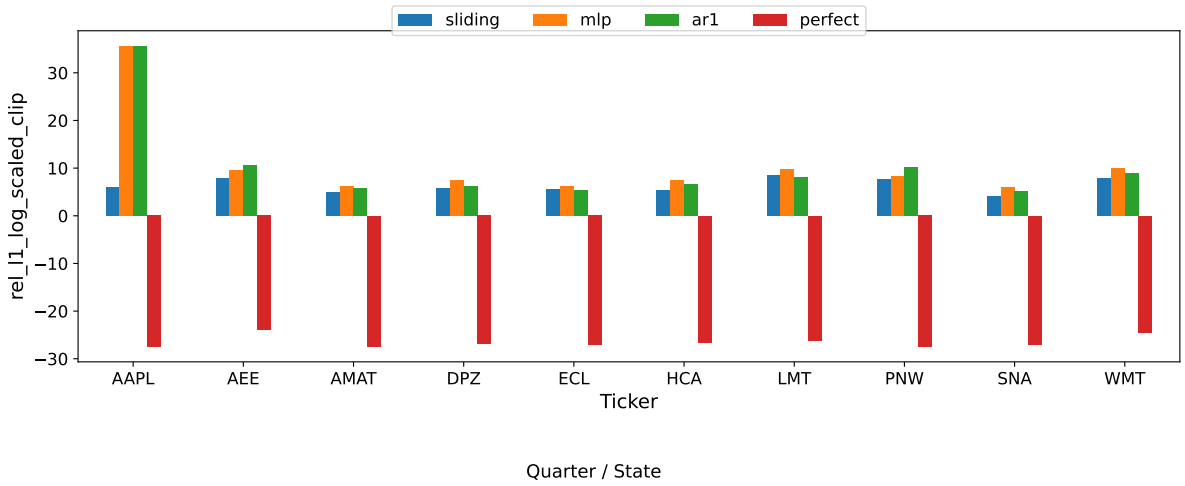


**Figure 2:** Relative L1 for state recovery for quarter balance sheet

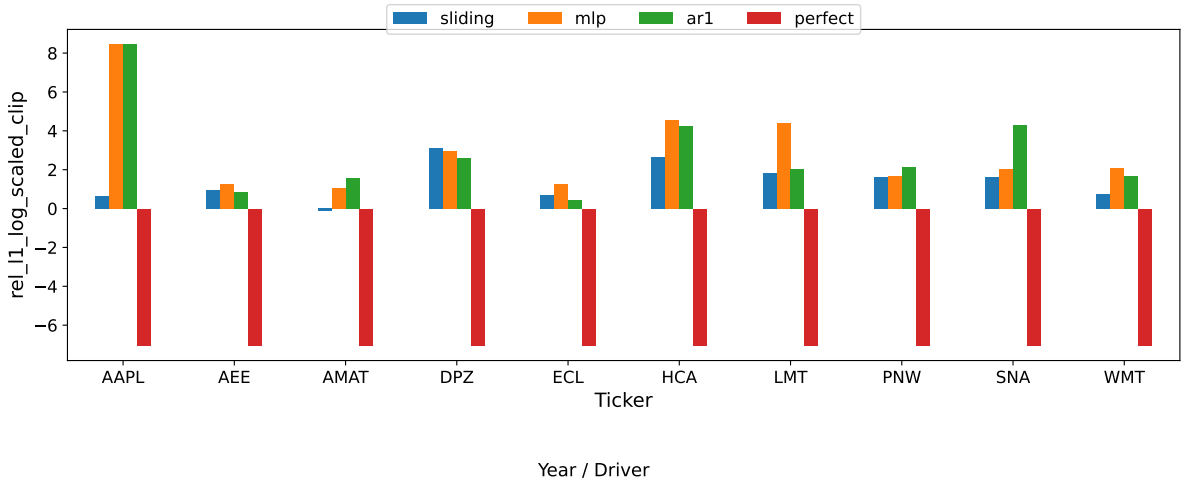


**Figure 3:** Relative L1 for driver recovery for year balance sheet

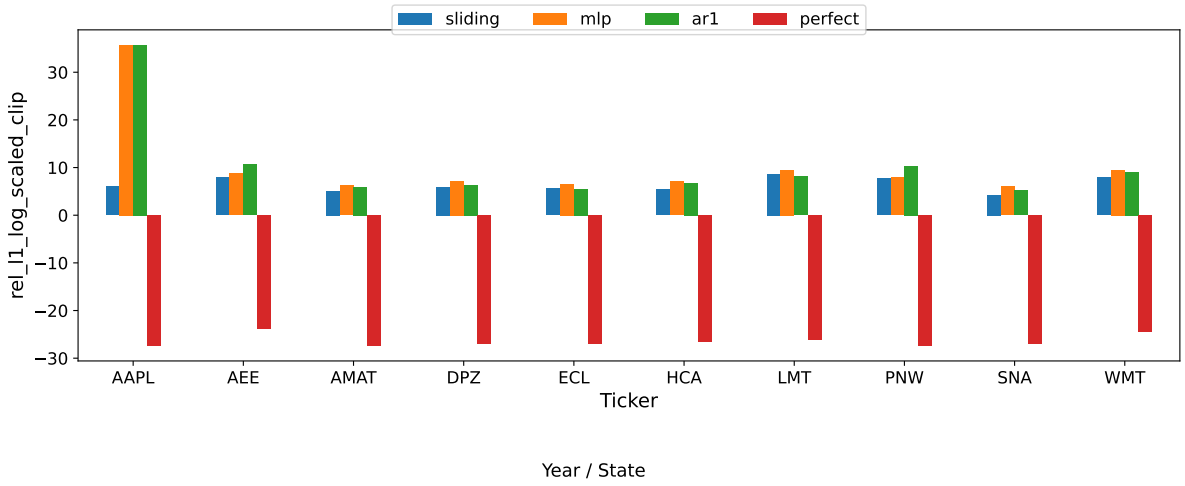


**Figure 4:** Relative L1 for state recovery for year balance sheet

(Fig. 3). Third, reconstruction of *state* variables is systematically more difficult than that of *drivers*, as reflected by larger log-errors in both annual and quarterly settings (cf. Fig. 2). This asymmetry is consistent with the fact that states are generated through multi-equation accounting evolution, which amplifies small driver-level perturbations. Finally, the qualitative ranking of tickers is highly stable across all four panels; for example, AAPL is consistently the most challenging entity to recover, while AMAT and ECL remain among the easiest. The strong cross-figure agreement implies that recovery difficulty is primarily driven by firm-specific structural characteristics rather than by temporal frequency or model class.

## 6 Earnings Forecasting

Net income is already an intermediate output of the model: it is computed in (23) from sales, costs, expenses, depreciation, interest and tax. Therefore, once we have a forecast for next year's state $\hat{y}_{t+1}$ and drivers $\hat{x}_{t+1}$, we automatically obtain a forecast for next year's earnings:

$$\widehat{\mathrm{NI}}_{t+1} = \widehat{\mathrm{EBT}}_{t+1} - \widehat{\mathrm{TAX}}_{t+1}, \tag{70}$$

where

$$\widehat{\mathrm{EBIT}}_{t+1} = \hat{S}_{t+1} - \hat{C}_{t+1} - \widehat{SG}_{t+1} - \hat{D}_{t+1}, \tag{71}$$

$$\widehat{\mathrm{INT}}_{t+1} = \hat{r}_{t+1}\,\widehat{\mathrm{DEBT}}_{t}, \tag{72}$$

$$\widehat{\mathrm{EBT}}_{t+1} = \widehat{\mathrm{EBIT}}_{t+1} - \widehat{\mathrm{INT}}_{t+1}, \tag{73}$$

$$\widehat{\mathrm{TAX}}_{t+1} = \hat{\tau}_{t+1}\,\max(\widehat{\mathrm{EBT}}_{t+1}, 0). \tag{74}$$

Thus the model can be used directly as an earnings forecaster, and earnings forecasts are consistent with the balance sheet and cash-flow forecasts.

If earnings per share (EPS) is required, we can introduce additional structure for the share count (e.g. assuming a fixed number of shares or linking $\Delta \mathrm{EQ}_t^{\mathrm{ext}}$ to net share issuance). This is an extension on top of the core model.

# 7 Machine-Learning Extensions

The current implementation uses very simple driver forecasting models: per-firm sliding-window means, pooled AR(1), and a small MLP with one lag. In principle, a richer set of machine-learning techniques could be used to improve forecasts, especially if longer time series per firm are available.

We briefly outline several directions.

## 7.1 Richer sequence models for drivers

If we had longer driver histories (e.g. 10–20 years per firm), we could explore:

- **Vector autoregressions (VAR)** to capture cross-driver dependencies while remaining linear and interpretable;
- **Recurrent neural networks (RNNs)**, including LSTM and GRU architectures, to model non-linear temporal dynamics;
- **Transformer-based sequence models**, which can capture long-range dependencies and complex interactions between drivers over time.

These models would operate on the driver sequences $x_t^*$ and possibly their past lags, with the structural map $f$ still ensuring accounting consistency.

## 7.2 Cross-sectional features and global models

Because individual firms have short histories, it is natural to pool data across many firms and use *global* models that share parameters. To capture systematic differences across firms, we can augment the model with firm-level features, such as:

- industry or sector dummies;
- firm size (e.g. log assets or log sales);
- geographic footprint;
- profitability and leverage indicators.

These features can be concatenated to driver inputs (e.g. as additional inputs to an MLP or RNN) so that the model can learn different dynamics for different types of firms while still benefitting from pooling.

## 7.3 Exogenous covariates and causal structure

Beyond pure time-series models, we can incorporate exogenous covariates that may influence drivers, such as:

- macroeconomic variables (GDP growth, interest rates, inflation);
- commodity prices (for relevant industries);
- firm-level policy indicators (investment plans, leverage targets).

These covariates can be fed into the driver forecasting model alongside past drivers. Furthermore, the structural nature of the tank model makes it a natural backbone for causal analyses: if we

believe certain drivers are under management control (e.g. payout ratio, capex policy) while others are more exogenous (e.g. demand growth), we can simulate counterfactual policies and their impact on the balance sheet and earnings.

### 7.4 Regularisation, uncertainty, and robustness

From a machine-learning perspective, we can improve robustness and interpretability by:

- applying regularisation (e.g. L1/L2 penalties, dropout) to prevent overfitting in neural models;
- modelling parameter and prediction uncertainty (e.g. Bayesian regression for AR(1), Bayesian neural networks, or ensemble methods);
- stress-testing forecasts by perturbing drivers and examining the resulting distribution of state forecasts.

Because the structural map $f$ is deterministic and transparent, uncertainty in drivers can be propagated to uncertainty in balance sheet and earnings forecasts via straightforward simulation.

## 8 Part 2: Applying LLMs to Financial Statement Analysis

### 8.1 (a) LLM choice and rationale

We select `gpt-4.1-mini` as the primary LLM for both (i) driver forecasting in the balance-sheet simulation pipeline and (ii) PDF-to-structure extraction. The practical reason is *structured-output reliability*: for financial statement tasks we need stable JSON-like outputs (tables/fields/units) under low temperature settings, so that downstream code can deterministically compute ratios and evaluate models.

**Viewpoint.** For a pipeline that mixes deterministic accounting identities with probabilistic components, the LLM is most valuable when it behaves like a *robust parser and conservative forecaster*, not as a free-form analyst.

### 8.2 (b) Balance sheet forecast: LLM vs. Part 1 models (A2D benchmark)

**Experiment setup (what is being compared).** We keep the structural evolution map from Part 1 (the accounting-consistent simulator) fixed, and only swap the model that forecasts the driver vector. This produces an "Accounting-to-Drivers" (A2D) benchmark:

$$\hat{y}_{t+1} = f(y_t,\ \hat{x}_{t+1})\,,$$

where different methods produce $\hat{x}_{t+1}$ (baseline AR(1), sliding-window mean, small MLP, pure LLM, and driver-level ensembles).

**Metrics.** We report errors in *state space* (forecasted balance-sheet / income-statement state) using MSE/MAE and relative errors. Lower is better. The `perfect` row is an oracle sanity check: it should be near numerical zero if the forward/inverse accounting pipeline is implemented correctly.

| variant | model_type | model | state_test_mse | state_test_mae | state_test_rel_l1 | state_test_rel_l2 |
|---|---|---|---|---|---|---|
| year | baseline | ar1 | 2.38321e+19 | 1.67899e+09 | 2.06358e+13 | 6.38755e+28 |
| year | baseline | perfect | 1.93513e-11 | 1.61843e-06 | 2.99896e-15 | 3.32462e-28 |
| year | baseline | sliding_mean | 2.32949e+19 | 1.42339e+09 | 0.536618 | 6.30789 |
| year | baseline | small_mlp | 8.36817e+19 | 3.34484e+09 | 2.41211e+14 | 8.72741e+30 |
| year | ensemble | ar1 | 1.99422e+19 | 1.34470e+09 | 1.44451e+13 | 3.12990e+28 |
| year | ensemble | perfect | 1.93513e-11 | 1.61843e-06 | 2.99896e-15 | 3.32462e-28 |
| year | ensemble | sliding_mean | 2.26323e+19 | 1.43769e+09 | 0.814559 | 29.4967 |
| year | ensemble | small_mlp | 2.63171e+19 | 1.56506e+09 | 0.664793 | 13.7661 |
| year | llm | llm | 2.34610e+19 | 1.47366e+09 | 0.708752 | 19.7334 |

**Table 1:** A2D state-space forecast errors (annual). The `perfect` model is an oracle reconstruction baseline.

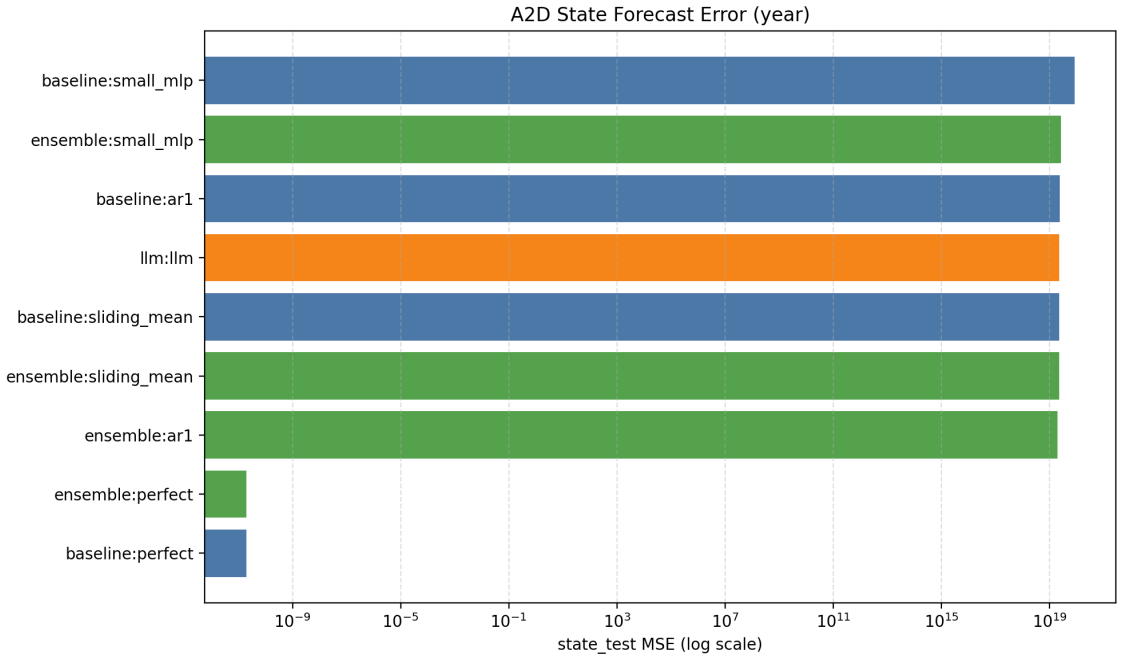


**Figure 5:** Annual A2D state-space MSE across model variants (log scale).

### 8.2.1 Annual (year) results

**Conclusion from Table 1 and Fig. 5.**

- **Sanity check passes:** `perfect` achieves near-zero error, validating that the accounting evolution map and driver inversion are consistent.
- **LLM is not strictly better than simple baselines:** the pure LLM is competitive but does not dominate; in particular, relative error is better for the sliding-window baseline.
- **Ensembling helps on absolute-error objectives:** the ensemble improves AR(1) materially in both MSE and MAE (about $\sim 16\%$ MSE and $\sim 20\%$ MAE improvement vs. baseline AR(1) on the annual state test).

### 8.2.2 Quarterly results

| variant | model_type | model | state_test_mse | state_test_mae | state_test_rel_l1 | state_test_rel_l2 |
|---|---|---|---|---|---|---|
| quarter | baseline | ar1 | 2.38321e+19 | 1.67899e+09 | 2.06358e+13 | 6.38755e+28 |
| quarter | baseline | perfect | 1.93513e-11 | 1.61843e-06 | 2.99896e-15 | 3.32462e-28 |
| quarter | baseline | sliding_mean | 2.32949e+19 | 1.42339e+09 | 0.536618 | 6.30789 |
| quarter | baseline | small_mlp | 5.34914e+19 | 2.61563e+09 | 2.29324e+14 | 7.88843e+30 |
| quarter | ensemble | ar1 | 1.87549e+19 | 1.32319e+09 | 1.65086e+13 | 4.08803e+28 |
| quarter | ensemble | perfect | 1.93513e-11 | 1.61843e-06 | 2.99896e-15 | 3.32462e-28 |
| quarter | ensemble | sliding_mean | 2.23931e+19 | 1.34155e+09 | 0.305703 | 1.59133 |
| quarter | ensemble | small_mlp | 3.12972e+19 | 1.51394e+09 | 0.636305 | 12.7988 |
| quarter | llm | llm | 2.34427e+19 | 1.48635e+09 | 0.730593 | 21.0094 |

**Table 2:** A2D state-space forecast errors (quarterly). The `perfect` model is an oracle reconstruction baseline.

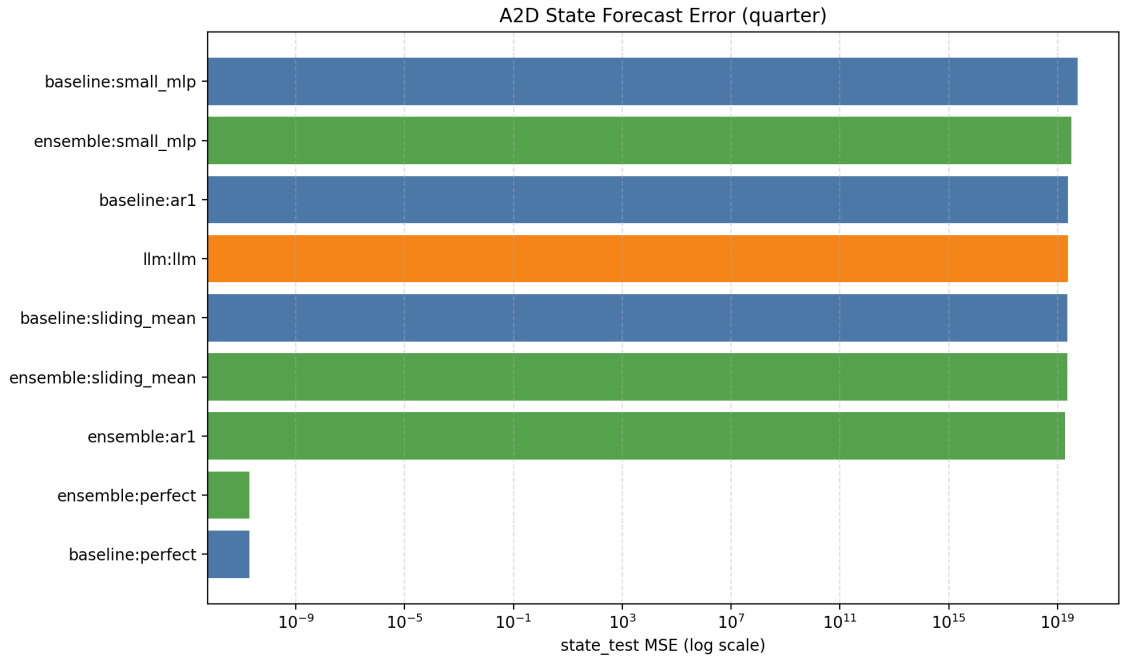


**Figure 6:** Quarterly A2D state-space MSE across model variants (log scale).

**Conclusion from Table 2 and Fig. 6.**

- **Best MSE/MAE:** ensemble AR(1) is best on MSE/MAE, improving baseline AR(1) by $\sim 21\%$ on both metrics.
- **Best relative errors:** the ensemble sliding-window mean wins on relative errors (Rel-L1/Rel-L2), showing that percentage-style accuracy favors conservative local-stationarity assumptions.
- **LLM-alone remains non-dominant:** the LLM baseline does not outperform the best non-LLM baselines under the same accounting-consistent simulator.

**Main takeaway (answering "better or worse").** On this dataset regime (short per-firm histories, strong accounting constraints), a pure LLM is *not consistently better* than classical baselines. Its value is clearer when used as part of an ensemble (see (c)) or as a robust extractor (see (f)–(h)).

### 8.3 (c) Can we build an ensemble that improves forecasting?

**Method.** We ensemble at the *driver level*, i.e., blend multiple driver forecasts and then propagate through the same accounting evolution map:

$$\hat{x}_{t+1} = \sum_k w_k \hat{x}_{t+1}^{(k)}, \qquad w_k \geq 0, \ \sum_k w_k = 1,$$

with weights chosen by validation grid search (convex blending).

**Evidence and conclusion.** Tables 1–2 show that ensembling can clearly reduce MSE/MAE (absolute error), especially for AR(1). However, relative errors are not uniformly improved: the best model depends on whether the objective is *dollar error* (MSE/MAE) or *scale-normalized error* (Rel-L1/Rel-L2). **Practical viewpoint:** for lending / credit workflows spanning heterogeneous firm sizes, relative error is often more decision-relevant; for treasury/cash planning, absolute error can matter more.

### 8.4 (d) CFO/CEO recommendation using the driver decomposition

We illustrate recommendations with the driver-based structure (even when the LLM is not the best forecaster, the driver decomposition is still operationally valuable). The core benefit is interpretability: each driver corresponds to a management lever and has a directional impact on liquidity, leverage, and earnings.

**Recommendation summary (what to do, not just what the model says).**

- Prioritize **liquidity resilience** via working-capital levers (Table 3); these typically move cash faster than long-horizon strategic actions.
- Use the **ensemble** (rather than a pure LLM) when the objective is minimizing absolute forecast error (Tables 1–2).
- Treat the LLM as a **supporting tool**: strong for extraction, useful as a complementary forecaster, but not a drop-in replacement for simple statistical baselines under short histories.

| Driver | What it operationally means | Actionable CFO/CEO lever (examples) |
|---|---|---|
| DSO / DIO / DPO | Working-capital efficiency (AR/INV/AP tied to sales and costs) | Tighten collections (DSO), optimize inventory policy (DIO), renegotiate supplier terms (DPO) |
| Capex-to-sales | Investment intensity and PPE growth | Capex gating, ROIC discipline, shift to asset-light where feasible |
| Net debt issuance | Leverage expansion / contraction | Refinancing plan, maturity ladder management, covenant headroom policy |
| Payout ratio | Cash returned vs. retained earnings | Dividend/buyback policy aligned with volatility and funding needs |
| Interest rate on debt | Cost of funding for a given leverage level | Hedge policy, fixed vs. floating mix, timing of issuance |

**Table 3:** Driver-to-decision mapping: why the Part 1/2 setup is useful for management recommendations.

## 8.5 (e) GM annual report: locating income statement and balance sheet

The pipeline performs automatic page detection by scanning the PDF text for statement headers and table-like numeric structure. In the GM report, the detected PDF pages differ from printed page numbers due to front-matter offsets; nevertheless, the correct statement pages are found and passed to the extraction stage.

**Viewpoint.** Page detection is a non-trivial practical bottleneck for real reports; a reliable heuristic layer is often higher leverage than swapping to a larger LLM.

## 8.6 (f) GM PDF extraction: ratios and cross-model robustness (mean ± std over runs)

We extract key statement fields and compute: net income, cost-to-income, quick ratio, debt-to-equity, debt-to-assets, debt-to-capital, debt-to-EBITDA, and interest coverage. Each model is run 5 times; we report mean ± std.

| company | model | n_runs | net_income | cost_to_income | quick_ratio | debt_to_equity | debt_to_assets | debt_to_capital | debt_to_ebitda | interest_coverage |
|---|---|---|---|---|---|---|---|---|---|---|
| General Motors | gemini-2.5-flash | 5 | 9840 ± 0 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.78535 ± 2.22045e-16 | 0.445833 ± 0 | 0.640978 ± 0 | 13.0932 ± 0 | 10.2064 ± 0 |
| General Motors | gemini-2.5-pro | 5 | 10127 ± 0 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.78535 ± 2.22045e-16 | 0.445833 ± 0 | 0.640978 ± 0 | 13.0932 ± 0 | 10.2064 ± 0 |
| General Motors | gpt-4.1-mini | 5 | 9840 ± 0 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.2195 ± 0.00184707 | 0.304621 ± 0.000358815 | 0.549448 ± 0.000374947 | 8.94612 ± 0.0105377 | 10.2064 ± 0 |
| General Motors | gpt-4.1-nano | 5 | 9840 ± 0 | 0.978464 ± 0.0493802 | 0.901657 ± 1.11022e-16 | 0.240699 ± 2.77556e-17 | 0.0601068 ± 0 | 0.194003 ± 0 | 1.76522 ± 0 | 10.2064 ± 0 |
| General Motors | gpt-4o | 5 | 9840 ± 0 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.22191 ± 0.00143688 | 0.305134 ± 0.000358815 | 0.549937 ± 0.000291127 | 8.96117 ± 0.0105377 | 10.2064 ± 0 |
| General Motors | gpt-4o-mini | 5 | 10127 ± 0 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 0.436883 ± 0.392368 | 0.109098 ± 0.0979814 | 0.265178 ± 0.14235 | 3.20398 ± 2.87752 | 10.2064 ± 0 |
| General Motors | grok-3-mini | 5 | 9897.4 ± 114.8 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.78711 ± 0.00215532 | 0.446273 ± 0.000538223 | 0.641205 ± 0.000277376 | 13.1062 ± 0.0158066 | 10.2064 ± 0 |
| General Motors | llama-3-1-8b-instruct | 5 | 10207 ± 40 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.64135 ± 0.105709 | 0.409875 ± 0.0263973 | 0.620808 ± 0.0149347 | 12.0372 ± 0.775238 | 10.2064 ± 0 |
| General Motors | llama-3-3-70b-instruct | 5 | 9840 ± 0 | 0.879419 ± 0.000203539 | 0.901657 ± 1.11022e-16 | 1.78564 ± 0.000586605 | 0.445906 ± 0.000146486 | 0.641016 ± 7.55716e-05 | 13.0954 ± 0.004302 | 10.2064 ± 0 |
| General Motors | qwen-max-latest | 5 | 10127 ± 0 | 0.879703 ± 0 | 0.901657 ± 1.11022e-16 | 1.77907 ± 2.22045e-16 | 0.444266 ± 0 | 0.640167 ± 0 | 13.0472 ± 0 | 10.2064 ± 0 |

**Table 4:** GM extraction and ratio computation across multiple LLMs (5 runs per model; mean ± std).

**Conclusions supported by Table 4.**

- **High run-to-run stability for strong models:** many models show near-zero std on most fields, suggesting the extraction is robust under repeated runs for clearly-defined line items.

- **Some models are unreliable on leverage:** a few models produce materially different debt ratios (and in some cases non-trivial std), indicating sensitivity in mapping "debt" and related components from statement lines.
- **Net income can cluster across different extracted interpretations:** multiple distinct net-income means appear across models, which is consistent with differences in year selection, unit interpretation, or statement line matching.

## 8.7 (g) Robustness protocol and tooling versions

**Robustness.** For extraction-to-indicators (E2I), each model is executed for 5 runs at temperature 0.2; Table 4 reports mean ± std. For balance-sheet forecasting (A2D), the driver simulation is deterministic given drivers; stored runs use temperature 0 for the LLM forecaster component.

**Versions (for reproducibility).**

- PDF parsing: `pdfplumber==0.11.9`
- LLM client: `openai SDK==2.15.0` (OpenAI-compatible endpoint)
- Endpoint: APIYi OpenAI-compatible base URL (`https://api.apiyi.com/v1`)

**Viewpoint.** Most robustness problems in financial extraction are not "LLM randomness" per se, but ambiguity in: (i) units/currency, (ii) line-item definitions (what counts as debt), and (iii) derived quantities (e.g., EBITDA requiring D&A consistency). A production system should add deterministic post-checks and unit normalization.

## 8.8 (h) Generalization test: LVMH annual report extraction

We repeat the same extraction and ratio computation pipeline on LVMH's (IFRS-style) annual report and again compare multiple LLMs using 5 runs per model.

| company | model | n_runs | net_income | cost_to_income | quick_ratio | debt_to_equity | debt_to_assets | debt_to_capital | debt_to_ebitda | interest_coverage |
|---|---|---|---|---|---|---|---|---|---|---|
| LVMH | gemini-2.5-flash | 5 | 12550 ± 0 | 0.769222 ± 0 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.21341 ± 0 | 19.8603 ± 0 |
| LVMH | gemini-2.5-pro | 5 | 12550 ± 0 | 0.769222 ± 0 | 0.706375 ± 0 | 0.382588 ± 0.102946 | 0.177682 ± 0.0478102 | 0.273094 ± 0.0486868 | 1.40204 ± 0.377257 | 19.8603 ± 0 |
| LVMH | gpt-4.1 | 5 | 12550 ± 0 | 0.769222 ± 0 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.21341 ± 0 | 19.8603 ± 0 |
| LVMH | gpt-4.1-mini | 5 | 12550 ± 0 | 0.769222 ± 0 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.21341 ± 0 | 19.8603 ± 0 |
| LVMH | gpt-4.1-nano | 5 | 12550 ± 0 | 0.695771 ± 0 | 0.706375 ± 0 | 0.365587 ± 0.067986 | 0.16981 ± 0.0320665 | 0.266021 ± 0.0340021 | 1.33993 ± 0.253028 | 37.0725 ± 0 |
| LVMH | gpt-4o | 5 | 12550 ± 0 | 0.769222 ± 0 | 0.706375 ± 0 | 0.330538 ± 0.000707056 | 0.153509 ± 0.000328372 | 0.248424 ± 0.000399479 | 1.2113 ± 0.00259109 | 19.8603 ± 0 |
| LVMH | gpt-4o-mini | 5 | 12550 ± 0 | 0.695771 ± 0 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.21341 ± 0 | 42.776 ± 0 |
| LVMH | grok-3-mini | 5 | 12550 ± 0 | 0.782447 ± 0.0282365 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.19695 ± 0.0201683 | 30.269 ± 10.9202 |
| LVMH | llama-3-1-8b-instruct | 5 | 12550 ± 0 | 0.662919 ± 0.318179 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.17224 ± 0 | 24.7109 ± 0 |
| LVMH | llama-3-3-70b-instruct | 5 | 12550 ± 0 | 0.549437 ± 0.179453 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.21341 ± 0 | 33.6097 ± 11.2264 |
| LVMH | qwen-max-latest | 5 | 12550 ± 0 | 0.769222 ± 0 | 0.706375 ± 0 | 0.331116 ± 0 | 0.153777 ± 0 | 0.24875 ± 0 | 1.21341 ± 0 | 23.8725 ± 0 |

**Table 5:** LVMH extraction and ratio computation across multiple LLMs (5 runs per model; mean ± std).

**Conclusions supported by Table 5.**

- **Portability is high:** key values (e.g., net income, liquidity proxy) are consistent across many models, indicating the extraction pipeline generalizes beyond a single issuer and reporting standard.
- **Ratios involving coverage or "cost" show larger dispersion:** cost-to-income and interest coverage vary more for some models, consistent with differences in interpreting operating costs and interest-related lines under IFRS presentations.

### 8.9 (i) Extension to other companies: Tencent / Alibaba / JPM / Exxon / Volkswagen / Microsoft / Google

The same pipeline is feasible to extend, but practical adjustments are needed:

- **Page hints and multilingual headers:** Tencent/Alibaba may require Chinese section-header matching; some firms place statements in different sections.
- **Industry- and bank-specific definitions:** for banks (e.g., JPM), ratios like "cost-to-income" have standardized banking interpretations and the balance-sheet structure differs materially from corporates; definitions must be customized.
- **Post-extraction validation:** add deterministic checks (unit normalization, sign conventions, basic accounting identity checks where applicable) before computing ratios.

**Final takeaway.** LLMs are best used as *components* in a financial-statement system: strong at flexible extraction and helpful as an ensemble forecaster, but not a guaranteed replacement for simple statistical baselines when time-series history is short and the accounting structure is already doing most of the heavy lifting.

## 9 Bonus: Credit Rating, Risk Warnings, and Loan Pricing

### 9.1 Bonus B: Credit rating from annual reports

#### 9.1.1 B1. Mathematical form of a credit rating model

A practical rating engine can be written as a two-stage model: (i) **quantitative risk scoring** from financial statement ratios, and (ii) **qualitative override** from audit opinions / disclosure signals.

**Quantitative scoring (supervised classification).** Let $\phi(\text{Report})$ be a feature vector extracted from the annual report and/or structured statements, including profitability, leverage, liquidity, and coverage ratios. A simple baseline is multiclass logistic regression:

$$p(r = k \mid \phi) = \frac{\exp(\mathbf{w}_k^\top \phi + b_k)}{\sum_{k'} \exp(\mathbf{w}_{k'}^\top \phi + b_{k'})}, \tag{75}$$

where $r \in \{\text{AAA}, \text{AA}, \text{A}, \text{BBB}, \text{BB}, \text{B}, \text{CCC}\}$. Because ratings are *ordinal*, an ordinal-logit / threshold model is also natural:

$$s = \mathbf{w}^\top \phi + b, \quad r = \text{bin}(s;\ \theta_1 < \cdots < \theta_K). \tag{76}$$

**Audit-opinion override (rule layer).** Some signals should dominate any ratio-based score because they imply immediate uncertainty about going concern or reliability of the statements. A minimal rule layer is:

$$r = \begin{cases} \text{D}, & \text{if disclaimer/going-concern evidence is detected;} \\ \arg\max_k \ p(r = k \mid \phi), & \text{otherwise.} \end{cases} \tag{77}$$

**Viewpoint.** In credit, **high-precision qualitative triggers** (e.g., disclaimer of opinion, going concern) are often more decision-relevant than marginal improvements in quantitative fit, so the

model should explicitly separate “score” vs. “override”.

### 9.1.2 B2. Training data source and label construction

We construct ratio features using Yahoo Finance statements (processed into consistent yearly states) and build a small rating dataset (40 tickers, 168 rows). The dataset is imbalanced (AAA is about half the rows), so performance must be read with care.

| rating | count |
|---|---|
| AAA | 83 |
| AA | 16 |
| A | 11 |
| BBB | 10 |
| BB | 14 |
| B | 6 |
| CCC | 28 |

**Table 6:** Rating distribution in the bonus credit-rating dataset (40 tickers, 168 rows).

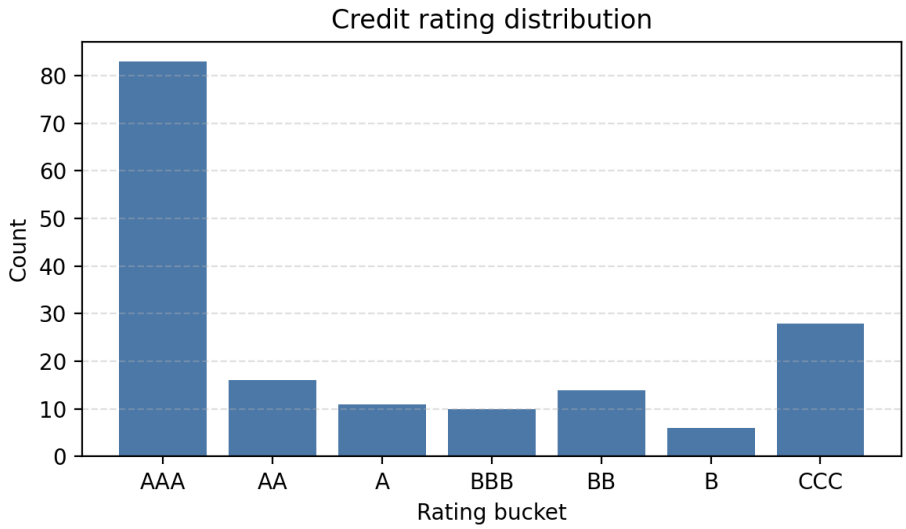


**Figure 7:** Credit rating class distribution (same data as Table 6).

**Conclusion (data).** The label imbalance implies that plain accuracy can be inflated by frequent classes; macro metrics (macro-F1) are more diagnostic for whether the model truly distinguishes weak credits.

### 9.1.3 B3. Baseline rating model results

We train a supervised baseline (logistic regression on ratio features). Model metrics are:

**Conclusions supported by Table 7.**

- **Usable headline accuracy:** multiclass accuracy is 0.667, indicating the ratio feature set has predictive signal.

| metric | value |
|---|---|
| multiclass_accuracy | 0.666667 |
| macro_f1 | 0.340816 |
| weighted_f1 | 0.584354 |
| ordinal_accuracy | 0.285714 |

**Table 7:** Credit rating baseline metrics (multiclass logistic regression on ratio features).

- **But weak minority-class discrimination:** macro-F1 is only 0.341, consistent with class imbalance (AAA dominates Table 6).
- **Ordinal structure remains hard:** ordinal accuracy is 0.286, suggesting that while the model often gets the broad regime right, it struggles to place firms at the correct notch on the rating ladder.

**Viewpoint.** For ratings, **macro-F1 and ordinal objectives** are more aligned with business use than raw accuracy. A production model should use (i) class-balanced loss or re-weighting and (ii) ordinal-aware training.

#### 9.1.4 B4. Evergrande rating assignment (2022)

For Evergrande (2022), the model assigns **D** driven by the audit-opinion override: repeated *disclaimer* and *going concern* language is detected on multiple pages.

| pattern | page |
|---|---|
| going_concern | 28 |
| disclaimer | 43 |
| going_concern | 43 |
| going_concern | 44 |
| disclaimer | 45 |
| going_concern | 54 |

**Table 8:** Evergrande (2022) audit evidence hits: detected patterns and PDF page numbers.

**Conclusion supported by Table 8.** Even if some financial ratios appear numerically computable, the presence of disclaimer/going-concern evidence implies that the statements' reliability and the firm's survival are in question; therefore a **hard downgrade to D** is justified under the override rule.

#### 9.1.5 B5. "Is the annual report correct?" Shenanigans detection

To sanity-check reported numbers, we implement a lightweight shenanigans scanner inspired by "Financial Shenanigans"-style heuristics:

- `ar_spike`: accounts receivable growth inconsistent with revenue growth (possible revenue recognition issues),
- `inventory_spike`: inventory growth inconsistent with COGS/sales (possible channel stuffing / inventory build-up),

- `cfo_vs_ni_gap`: large persistent gap between operating cash flow and net income (earnings quality warning).

| flag | count |
|---|---|
| ar_spike | 0 |
| inventory_spike | 0 |
| cfo_vs_ni_gap | 3 |

**Table 9:** Shenanigans scanner flag counts on the 40-ticker sample.

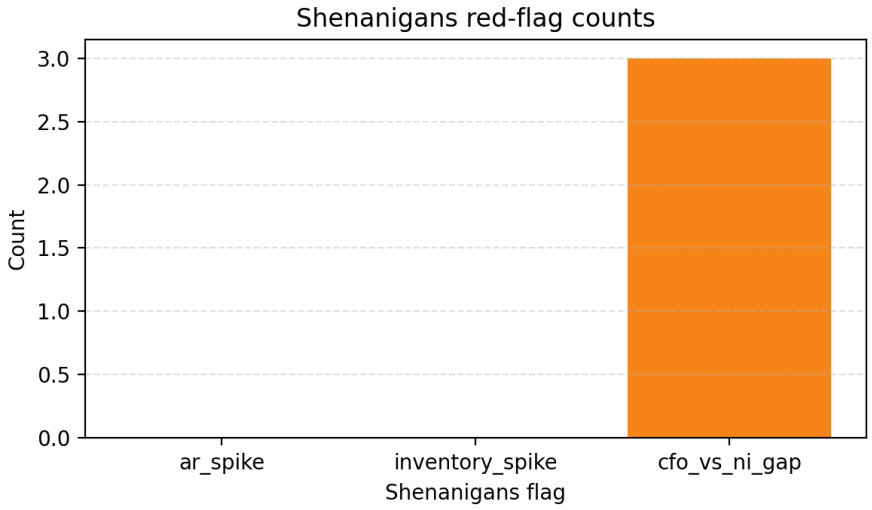


**Figure 8:** Red-flag counts (visualization of Table 9).

**Conclusions supported by Table 9 and Fig. 8.**

- **Most flags do not trigger** in this small sample (0 AR spikes, 0 inventory spikes).
- **The most informative quick screen here is cash-vs-earnings:** `cfo_vs_ni_gap` triggers in 3 cases.

**Viewpoint.** These heuristics are best treated as **triage signals** (cheap, fast, high-level). Low flag counts do not prove correctness; they mainly indicate that more targeted tests (segment-level revenue, related parties, off-balance-sheet, footnote parsing) are needed for a stronger "correctness" assessment.

## 9.2 Bonus C: Risk warnings extraction from annual reports

### 9.2.1 C1. What risk points matter most (beyond the bulk text)?

In practice, the highest-signal sections of annual reports include:

- **Audit opinion and modifications:** qualified/adverse/disclaimer, emphasis of matter, going concern language;
- **Liquidity and funding:** covenant breaches, refinancing risk, maturity walls, cash burn;
- **Asset quality:** impairment, write-downs, valuation uncertainty;

- **Legal and contingent liabilities:** litigation, regulatory actions, guarantees;
- **Defaults/restructuring:** missed payments, cross-default clauses, restructuring notes.

### 9.2.2 C2. Automatic extraction engine (audit detection + keyword ranking)

We implement a rule-based extractor that: (1) scans for audit-opinion paragraph patterns and their page locations, and (2) ranks extracted paragraphs by risk keyword evidence.

**Audit evidence.** We reuse the same audit detection evidence as in Bonus B (Table 8), which shows repeated going-concern/disclaimer hits.

| keyword | count |
|---|---|
| going concern | 4 |
| liquidity | 3 |
| impairment | 2 |
| litigation | 2 |
| default | 1 |

**Table 10:** Risk keyword evidence counts from extracted risk-related paragraphs.

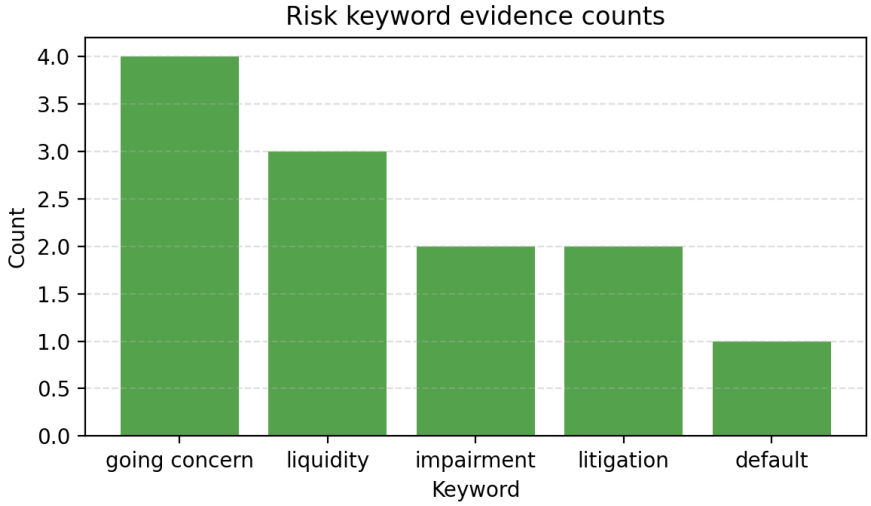


**Figure 9:** Risk keyword evidence counts (visualization of Table 10).

**Risk keyword evidence.**

**Conclusions supported by Table 10 and Fig. 9.** The extracted evidence concentrates on core failure modes (going concern, liquidity, impairment, litigation, default), which is exactly the intent: **reduce the report to a shortlist of actionable warning themes**.

**Viewpoint.** For long PDFs, a risk extraction system should optimize **recall on high-severity patterns** (audit modifications, going concern) rather than summarizing everything; the goal is decision support, not document compression.

## 9.3 Bonus D: Loan pricing (term-loan spread over Treasury)

### 9.3.1 D1. Model choice and brief literature positioning

A term loan rate is typically quoted as:

$$\text{Loan Rate} = y_{\text{Treasury}}(T) + \text{Spread},$$

so the learning problem is to predict Spread from borrower/loan features. This can be viewed as a reduced-form credit pricing approximation where the spread is a proxy for expected loss + risk premium + liquidity/servicing costs.

In implementation, we compare:

- a **linear** spread model (interpretable baseline),
- a **GBDT** spread model (captures nonlinearities and feature interactions).

### 9.3.2 D2. Training data source

We use public loan pricing data (LendingClub; sample size 5000) and Treasury yield inputs (FRED) to form the target spread over matched-maturity Treasury.

### 9.3.3 D3. Spread prediction results

| model | rmse | mae |
|---|---|---|
| linear | 0.00238213 | 0.000640729 |
| gbdt | 0.000434555 | 9.27626e-05 |

**Table 11:** Loan spread prediction errors (lower is better).

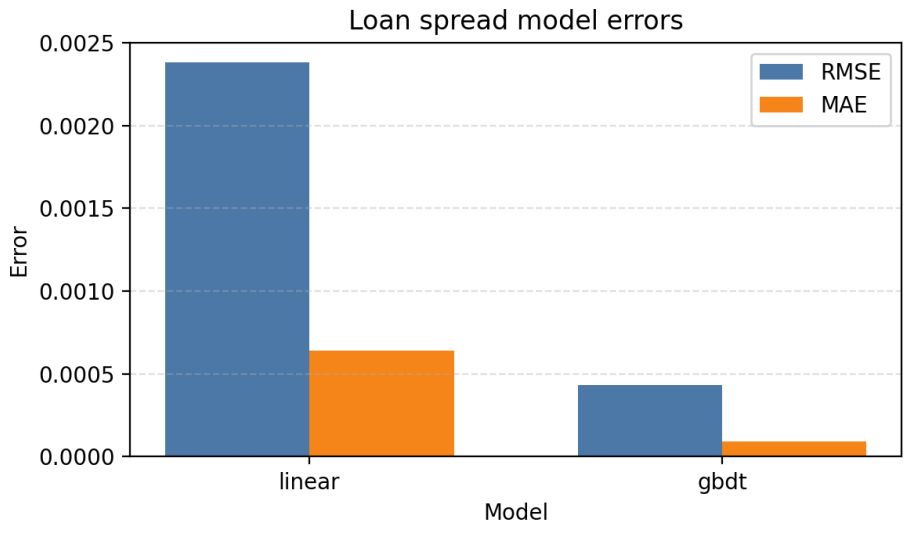


**Figure 10:** Spread model error comparison (visualization of Table 11).

**Conclusions supported by Table 11 and Fig. 10.**

- **Nonlinear model wins decisively:** GBDT reduces RMSE by $\sim 82\%$ and MAE by $\sim 86\%$ vs. the linear baseline, showing that spread formation is not well-approximated by a purely linear mapping in this dataset regime.

#### 9.3.4 D4. "Private borrower" (no tradable loans/bonds) and model robustness

We stress-test the model by removing market-grade signals; the private-borrower MAE increases sharply.

| metric | value |
|---|---|
| private_mae_full | 0.000640729 |
| private_mae_no_market | 0.00540997 |
| return_1m_rmse | 7.14131e-05 |
| return_1m_mae | 4.89226e-05 |
| pi_coverage | 0.9496 |
| pi_avg_width | 0.000161924 |

**Table 12:** Private borrower gap, resale forecasting metrics, and 95% prediction interval diagnostics.

**Conclusions supported by Table 12.**

- **Private borrower gap is large:** removing market signals raises MAE from $6.41 \times 10^{-4}$ to $5.41 \times 10^{-3}$ (about $8.4\times$ worse).

**Viewpoint.** A production-ready private-borrower model should replace market signals with **bank-available proxies** (financial statement ratios, cash-flow volatility from bank transactions, collateral coverage, sector macro factors) and ideally use hierarchical / transfer learning so private names can borrow statistical strength from similar public names.

#### 9.3.5 D5. Resale price forecasting after one month

We also forecast the 1-month resale return proxy; Table 12 reports the 1-month return RMSE/-MAE.

**Conclusion supported by Table 12.** The model can handle the resale forecasting task as a standard supervised regression objective (predict 1-month return), enabling mark-to-market / warehousing risk monitoring.

#### 9.3.6 D6. 95% confidence interval for the 1-month price

We compute a 95% prediction interval (PI) for the 1-month price/return and evaluate empirical coverage.

**Conclusion supported by Table 12.** Empirical PI coverage is 0.9496 (close to the 0.95 target), with a compact average width, indicating that the uncertainty layer is quantitatively usable for risk limits and pricing add-ons (e.g., capital charge / buffer based on tail risk).

## 10 Conclusion

We have constructed a simple yet internally consistent financial statement forecasting model based on the tank-model ideas of Vélez-Pareja Vélez-Pareja 2011; Vélez-Pareja 2010. The model describes the evolution of a 15-dimensional state vector of income-statement and balance-sheet items as a deterministic function of the previous state and a 13-dimensional driver vector of growth, margins, working-capital policies, capex, tax, interest, payout and net financing ratios.

The forward equations ensure that the model-world balance-sheet identity assets = liabilities + equity holds automatically at all times, provided it holds initially. Historical data can be inverted to recover "perfect" drivers that exactly reproduce the observed transitions. This allows us to recast the forecasting problem as one of multivariate time-series modelling of drivers, with the structural evolution map handling accounting consistency.

On a panel of firms with short histories, we can already implement and evaluate simple forecasting models for drivers (sliding-window mean, pooled AR(1) and a small MLP), and use the structural model to obtain consistent forecasts for all financial statement items and earnings. As more data become available, richer machine-learning models and additional covariates can be incorporated without sacrificing the accounting integrity of the forecasts.

## References


Mejía-Peláez, Felipe and Ignacio Vélez-Pareja (2011). "Analytical solution to the circularity problem in the discounted cash flow valuation framework". In: *Innovar* 21.42, pp. 55–68.

Vélez-Pareja, Ignacio (2010). "Constructing Consistent Financial Planning Models for Valuation". In: *IIMS Journal of Management of Science* 1.

– (2011). "Forecasting Financial Statements with No Plugs and No Circularity". In: *The IUP Journal of Accounting Research & Audit Practices* 10.1.