Generative Scenario Modeling for Semiconductor Industry Cycles

Team K
Seunghwan An, Seungjin Jeon, Muhammad Aliff

Table of Contents

Introduction

Industrial Background: Why Semiconductors Are Special?

1. **Super-Cyclical Industry (Much more volatile than ordinary manufacturing)**

- Semiconductor cycles are deeper, shorter, and more volatile than most industrial sectors.
- DRAM/NAND memory prices can move 30–70% in just 3–6 months, far faster than typical commodities.
- The industry reacts quickly because production capacity is fixed in the short run; **small demand shifts cause large price swings.**

조선일보
https://www.chosun.com › tech_it › 2025/10/01

D램 가격, 6년8개월 만에 최고치... 반년 만에 4.6배 됐다

2025. 10. 1. — **D램 가격은 올 4월 전달보다 22.22% 상승**하며 오름세를 시작해 7월과 8월에는 각각 전달보다 50%, 46.15% 올랐다. 올 3월 D램 고정 거래 가격이 1.35달러 ...

Industrial Background: Why Semiconductors Are Special?

2. **High Sensitivity to Global Shocks**

- Demand, inventory levels, and prices respond immediately to:

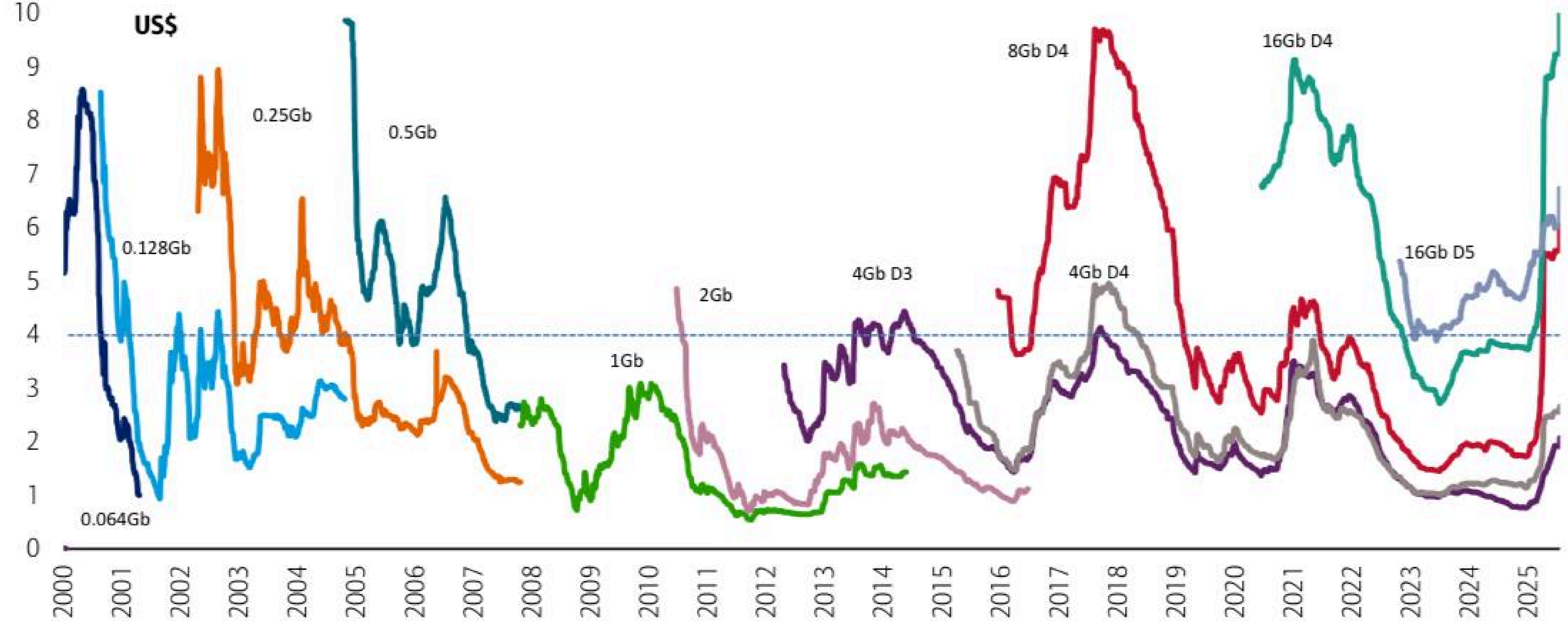
global interest rate changes, supply-chain disruptions (e.g., COVID-19), shifts in global manufacturing activity (PMI), geopolitical tensions.

- Results in repeated boom–bust cycles, **sudden collapses followed by rapid recoveries.**

Memory spot price trend

Exhibit 2: DRAM spot price – long-term trend (2000-2025)

Unprecedented spot price rally with record-high level for current mainstream DRAM 16Gb DDR5 at US$7 and 16Gb DDR4 at $10



Source: DRAMeXchange, BofA Global Research

BofA GLOBAL RESEARCH

Figure: Long-term DRAM spot price trend (2000–2025)

Industrial Background: Why Semiconductors Are Special?

3. **Structural Characteristics that Make Semiconductors Unique**

- Capital-intensive with long investment lead times (CAPEX cycles).
- Highly globalized supply chain, so shocks propagate rapidly across countries.
- Small end-market changes create large upstream fluctuations.
- Demand comes from many sectors, causing overlapping mini-cycles.

Industrial Background: Why Semiconductors Are Special?

4. **Data Limitations**

- Key indicators exist only at monthly frequency (exports, PMI, CAPEX, etc.).
- Most series have only 10–15 years of data (much shorter than typical macro datasets).
- Hard to train ML models without external macro information.

[단위 : 조원,%,억불,$]

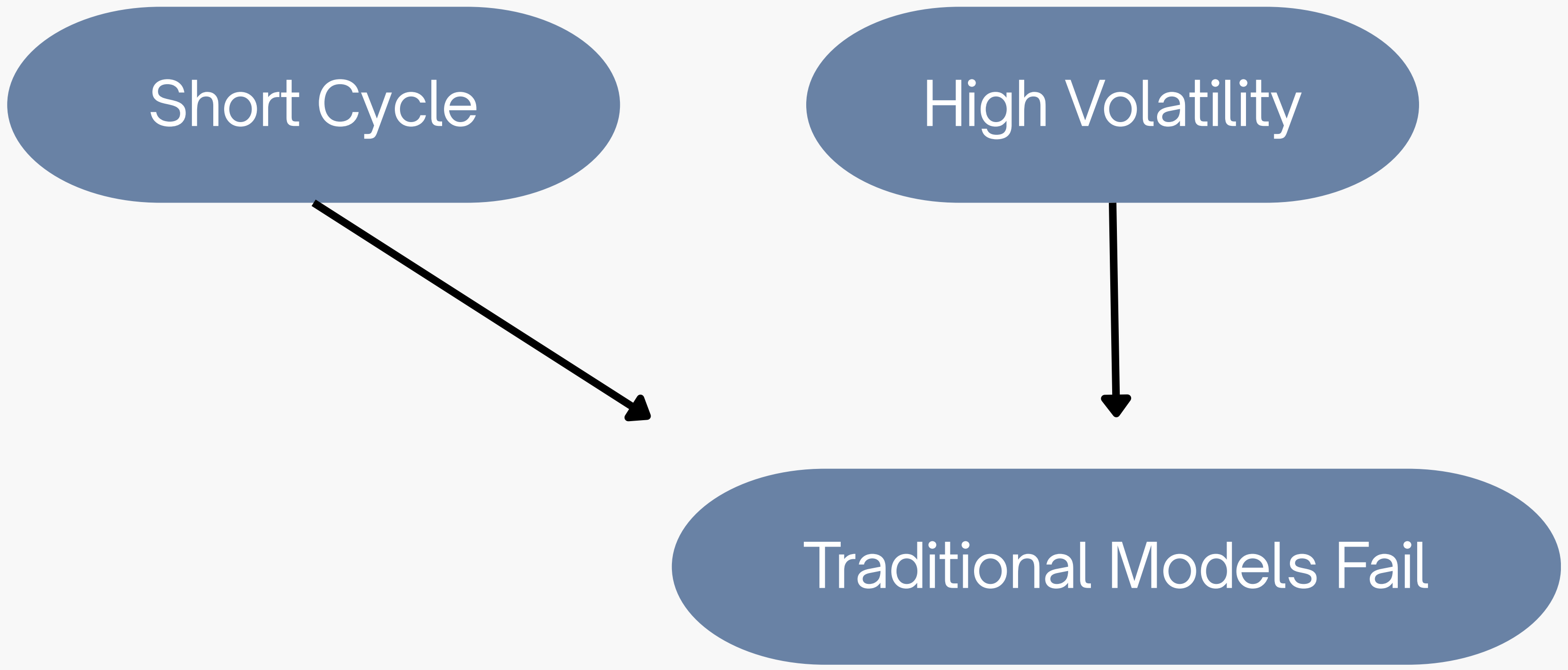
	2020	2021	2022	2023	2024
반도체 생산(조원)	159	201	224	168	-
시장점유율(%)	18	20	18	13	19
수출(억불)	992	1,280	1,292	986	1,419
수출증가율(%)	6	29	1	-24	44
수입(억불)	503	614	748	624	722
수입증가율(%)	7	22	22	-17	16
DRAM 가격($)	2	3	2	2	2

출처: 반도체산업협회, 정보통신산업진흥원 자료

Figure: Semiconductor Export Quantity (2020-2024)

Why Forecasting Semiconductor Cycles Is Hard

- Semiconductor dynamics are driven mainly by external macro shocks, not just internal patterns.
- Macro variables (interest rates, global PMI, liquidity, inflation) can reverse the cycle within months.
- Forecasting solely from past export or price data (ARIMA/LSTM) is fundamentally insufficient.



(ARIMA/LSTM can't handle macro shocks)

Limitations of Existing Time-Series Models

Univariate Models (ARIMA / Prophet / LSTM)

- Learn only from semiconductor history → cannot handle macro shocks.
- Overfit easily due to extremely small data size.

Multivariate Deep Models

- Simply concatenating macro + semiconductor inputs increases noise.
- Macro–semiconductor relationships are lagged, nonlinear, and unstable over time.
- With limited data, models quickly overfit, giving unstable predictions.

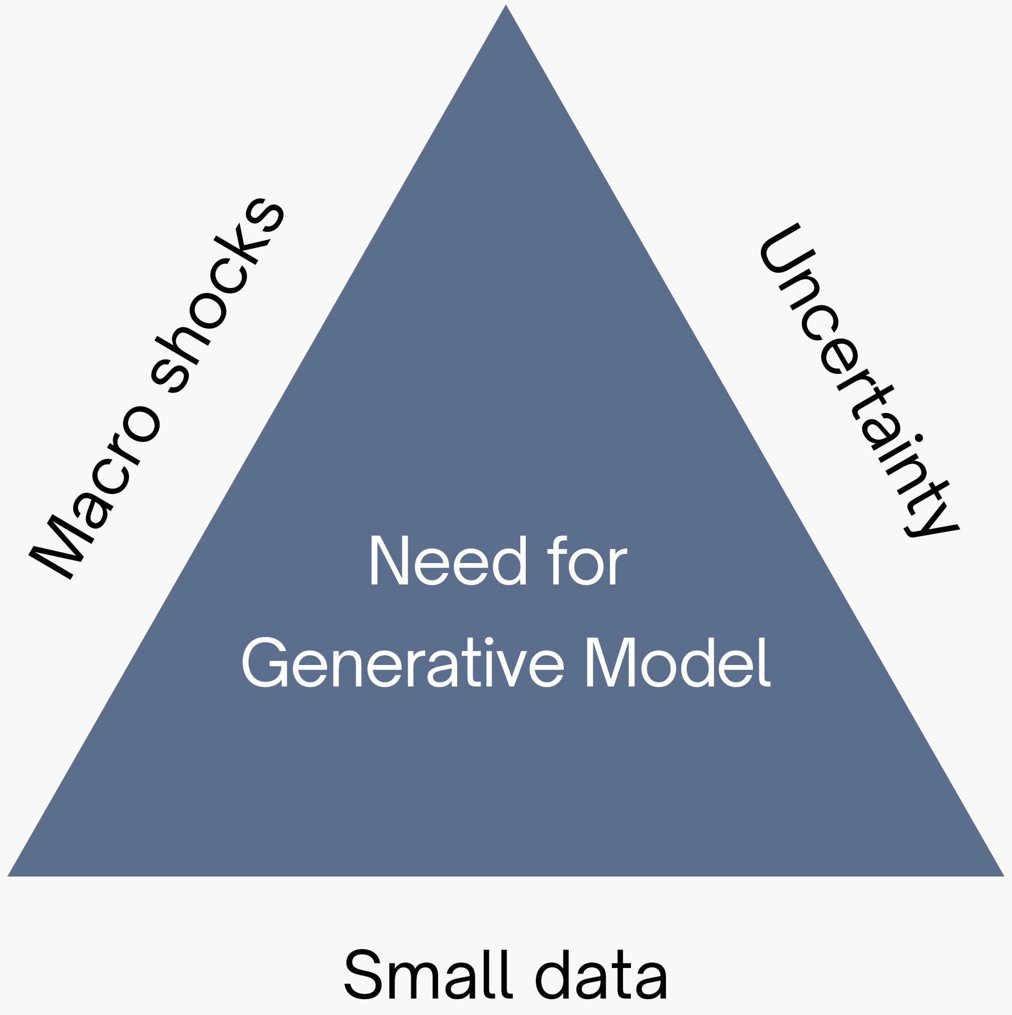
Research Gap

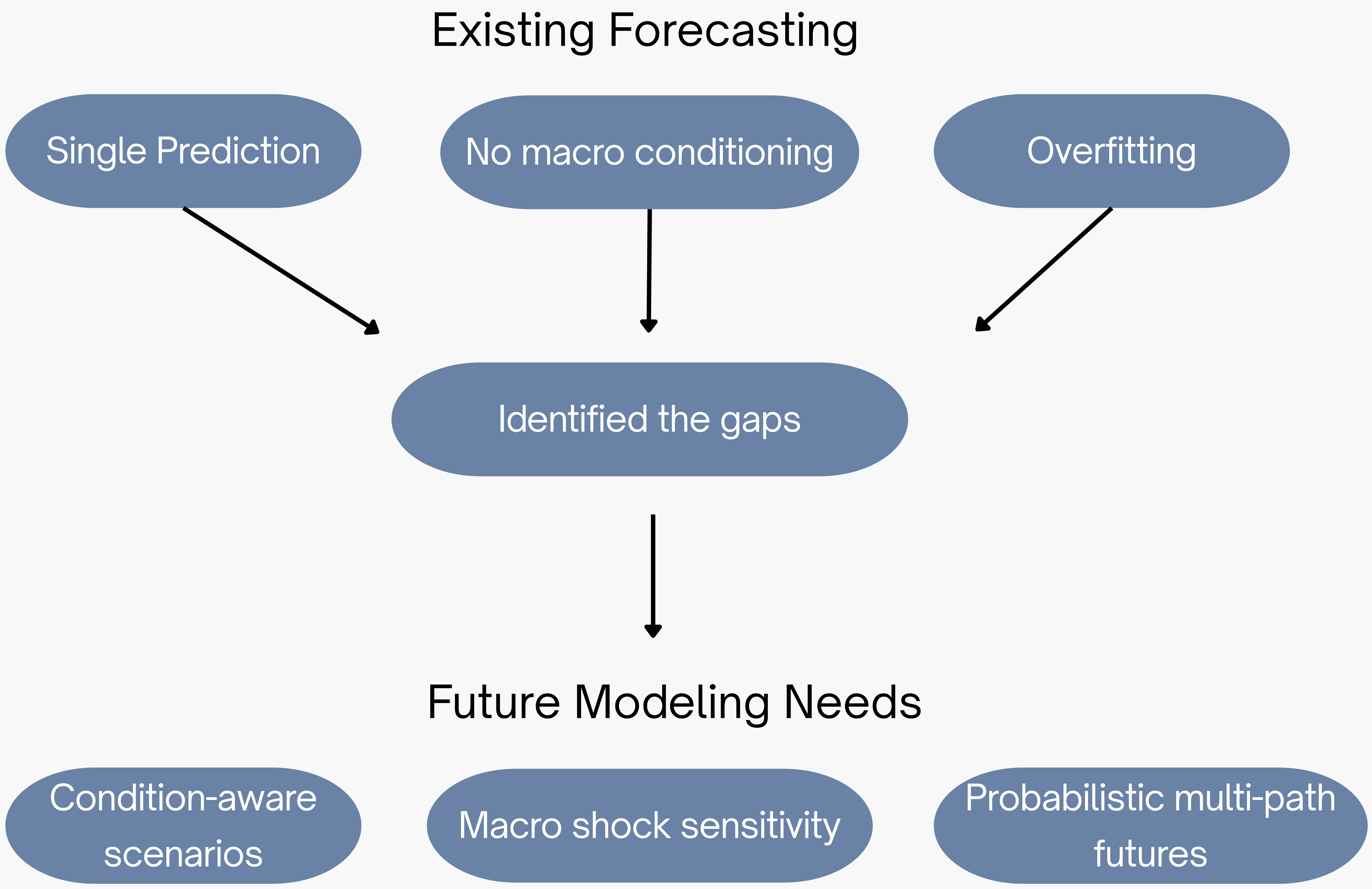
Three Core Gaps Identified:

1. **Macro shocks are essential**, but existing forecasting models do not structurally incorporate them.

2. **Very limited semiconductor data** requires models that extract structured latent representations rather than fitting raw sequences.

3. Traditional models produce one deterministic forecast, but real-world decisions **require multiple plausible scenarios** (what-if analysis).




Existing Forecasting
Single Prediction
No macro conditioning
Overfitting
Identified the gaps
Future Modeling Needs
Condition-aware scenarios
Macro shock sensitivity
Probabilistic multi-path futures

Motivation

- Semiconductor cycles are macro-driven, yet existing models treat the problem as a pure time-series task.
- Need a framework that:

Conditions on macroeconomic variables

Learns latent temporal structures

Generates diverse, scenario-based futures.

- Motivates the use of **Conditional Time-VAE**, a probabilistic, condition-aware generative model.

Dataset

Dataset for Pretrain

Federal Reserve Economic Data (FRED)

Macro VAE pretraining (1970–2025): Sentiment, Rates, Liquidity, Employment, Inflation, Output.

U.S. 10Y Treasury Yield (GS10)

M2 Money Stock (M2SL)

Unemployment Rate (UNRATE)

Consumer Price Index for All Urban Consumers (CPIAUCSL)

Industrial Production Index (INDPRO)

Purchasing Managers' Index (PMI)

Dataset for Pretrain

Transformations

- CPIAUCSL: log-difference (Δlog) → monthly inflation

Log-level

- M2SL: log level (log)
- INDPRO: log level (log)

Levels

- PMI, GS10, UNRATE: level (no transform)

Scaling

- z-score normalization (scaler fit on train split only)

Dataset for CT-VAE

From 2010 ~ 2025,

DRAM

Capital Expenditure (CAPEX)

Purchasing Managers' Index (PMI)

Export

Exchange Rate

Composite leading indicator (CLI)

ISM Manufacturing Index

Final Input Tensor (for CT-VAE)

$$x_t \in \mathbb{R}^D, \qquad D = [\text{Export, DRAM, FX, CAPEX, PMI, CLI, ISM}]$$

$$L = 36 \text{ months}, \qquad H = 12 \text{ months}$$

$$c_t \in \mathbb{R}^5,$$

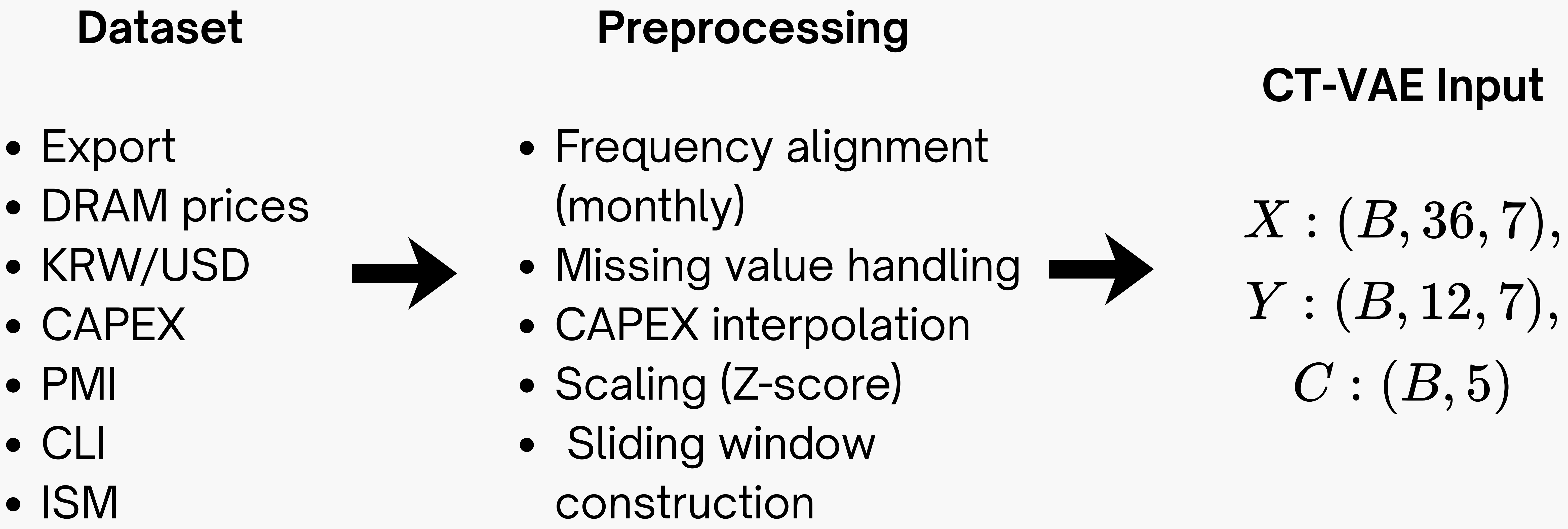
$$X : (B, 36, 7), \qquad Y : (B, 12, 7), \qquad C : (B, 5)$$

Preprocessing

- **Numeric conversion**: Convert all columns into numeric format and remove non-usable fields.
- **Log transform**: Apply log1p to CAPEX because it is skewed and sparse (quarterly).
- **Missing values**: Fill missing condition and macro variables with zero.

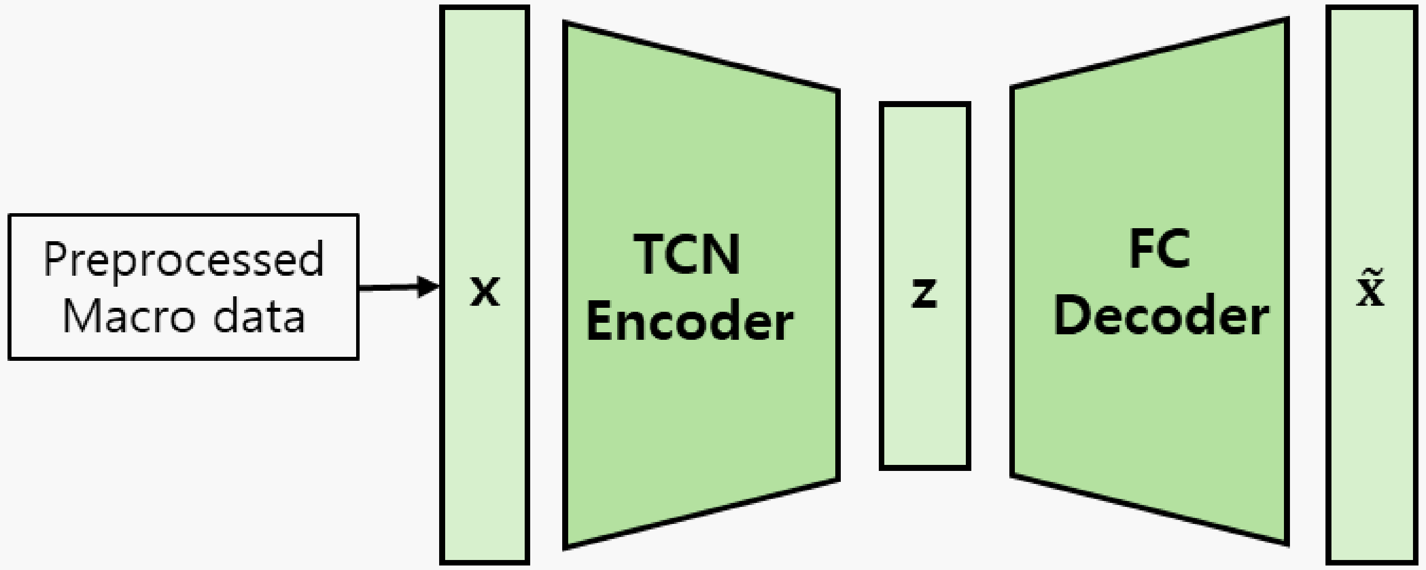
Preprocessing

- **Scaling**: Apply StandardScaler to normalize all features for neural network training.
- **Sliding window**: Convert the full dataset into sequences of 36-month input and 12-month output windows.
- **Condition extraction**: Extract 5 condition values from the last timestep of each window for baseline condition.
- **Final tensors**: Produce tensors X (B,36,7), Y (B,12,7), and C (B,5) for training.


Dataset
Export
DRAM prices
KRW/USD
CAPEX
PMI
CLI
ISM
Preprocessing
Frequency alignment (monthly)
Missing value handling
CAPEX interpolation
Scaling (Z-score)
Sliding window construction
CT-VAE Input
$X : (B, 36, 7),$
$Y : (B, 12, 7),$
$C : (B, 5)$

CT-VAE Model Architecture

Macro VAE Model Architecture



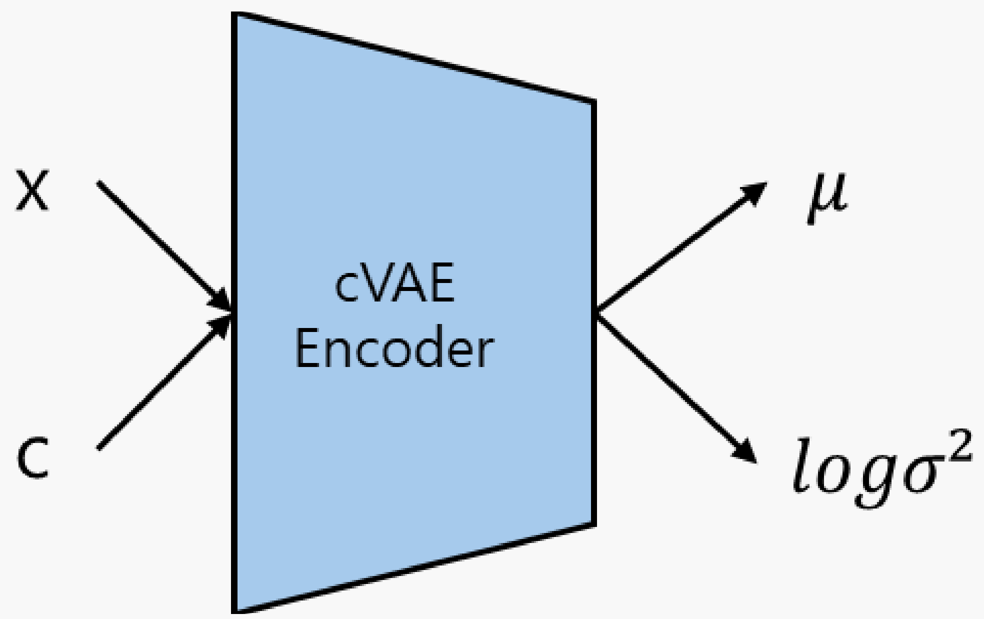
A TCN encoder summarizes a L-month macro window into a 32-dimensional latent state; an MLP head maps this state to a 12-month, 6-variable joint forecast, trained with an MSE loss and an annealed KL regularizer.

$$W = \lfloor rE \rfloor, \quad \beta(e) = \begin{cases} 0, & e \le W \\ \beta_{\max} \cdot \min\left(1, \dfrac{e-W}{E-W}\right), & e > W \end{cases}$$

e = epoch, E = total epochs, r = warmup ratio

$$\mathcal{L}(e) = \mathrm{MSE}(\hat{y}, y) + \beta(e)\,\mathrm{KL}(q_\phi(z|x) \| p(z))$$

CT-VAE Model Architecture : Encoder



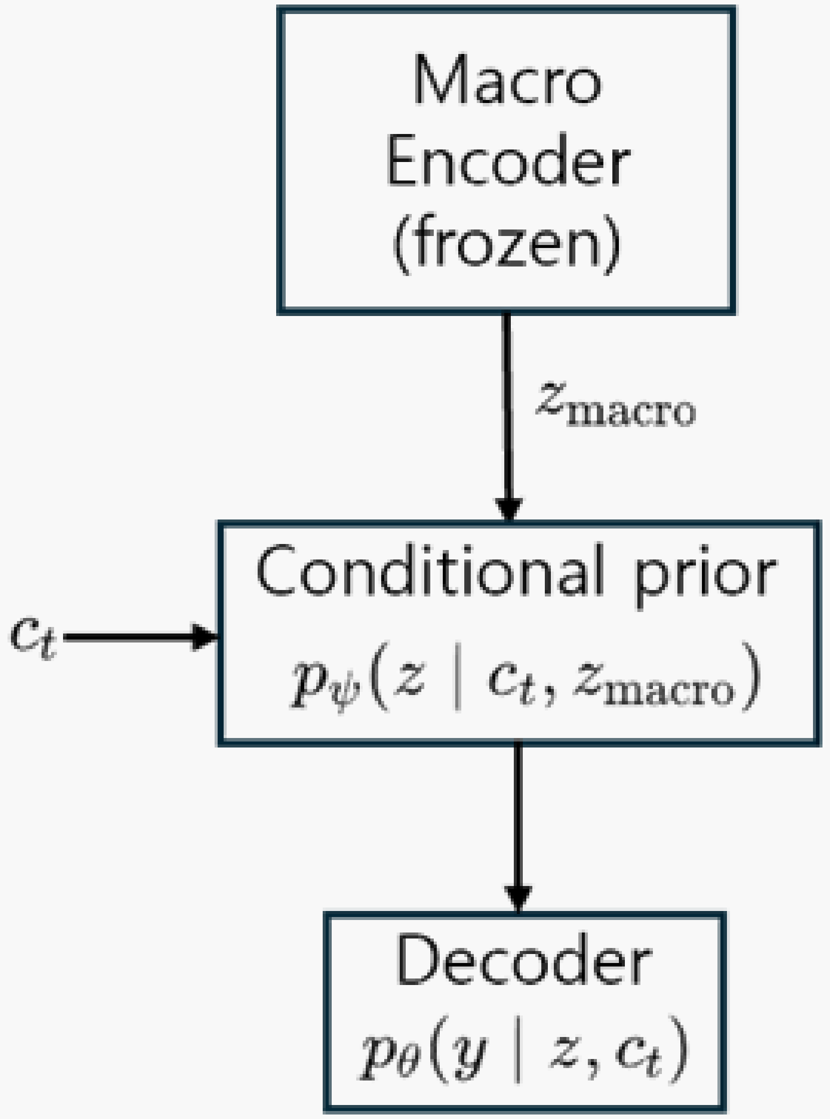
The CT-VAE encoder transforms each L-month input sequence into a 32-dimensional latent representation.

It first projects the input sequence x into a hidden feature map using a 1×1 convolution.

It then processes this representation through stacked TCN blocks whose activations are modulated by layer-wise FiLM parameters generated from the condition vector c.

$$h^{(k)} = \mathrm{TCN}^{(k)}\left(\gamma^{(k)}(c) \odot h^{(k-1)} + \beta^{(k)}(c)\right)$$

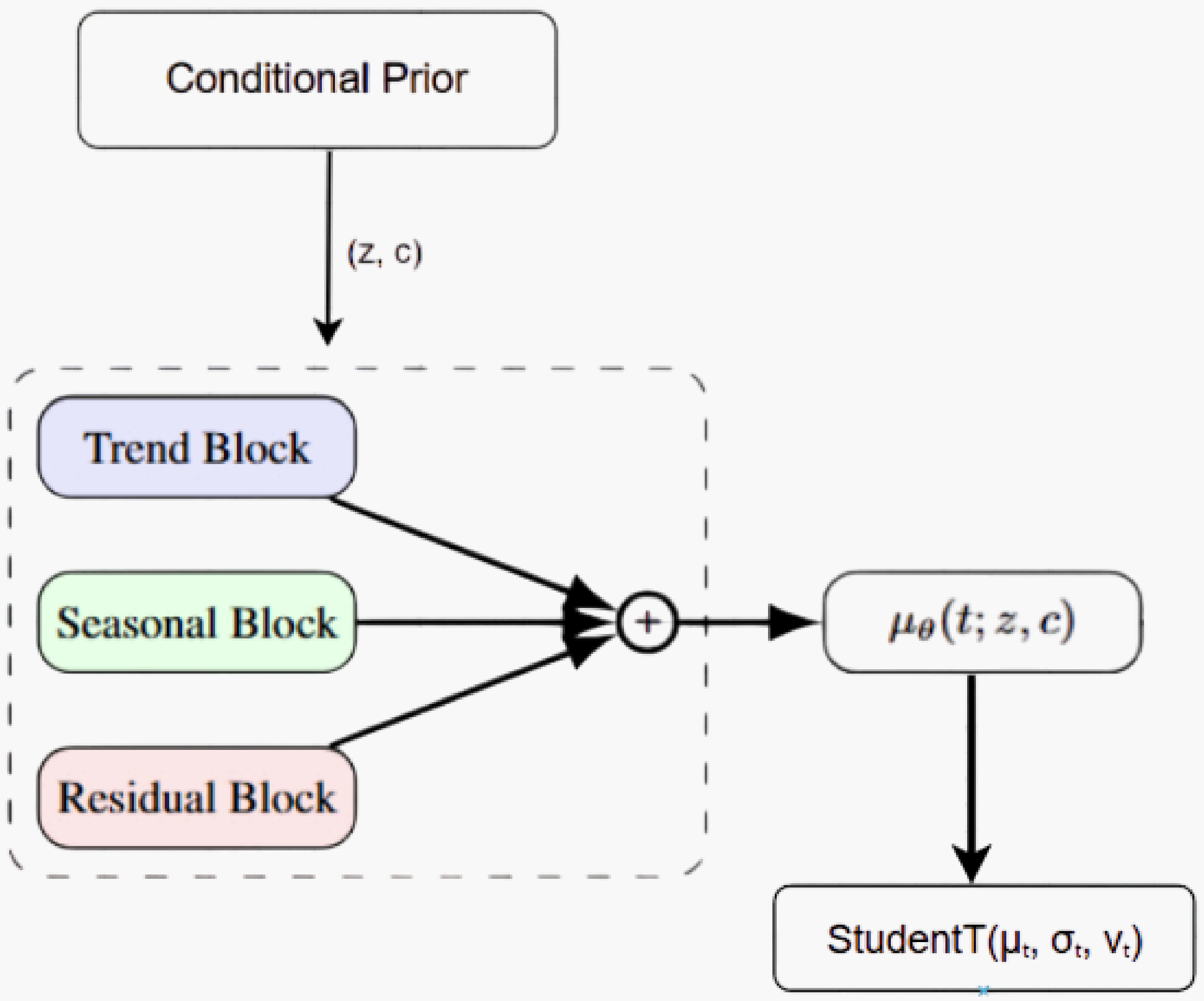
CT-VAE Model Architecture : Prior



A macro-conditioned prior is introduced: a frozen MacroEncoder produces z_{macro} from the last 36 months, and the prior network outputs a shifted Gaussian $p_\psi(z \mid c_t, z_{\text{macro}})$

The MacroEncoder is frozen during training and is only used to extract a macro-cycle embedding from the last 36 months. This embedding conditions the prior distribution but does not receive gradients.

CT-VAE Model Architecture : Decoder



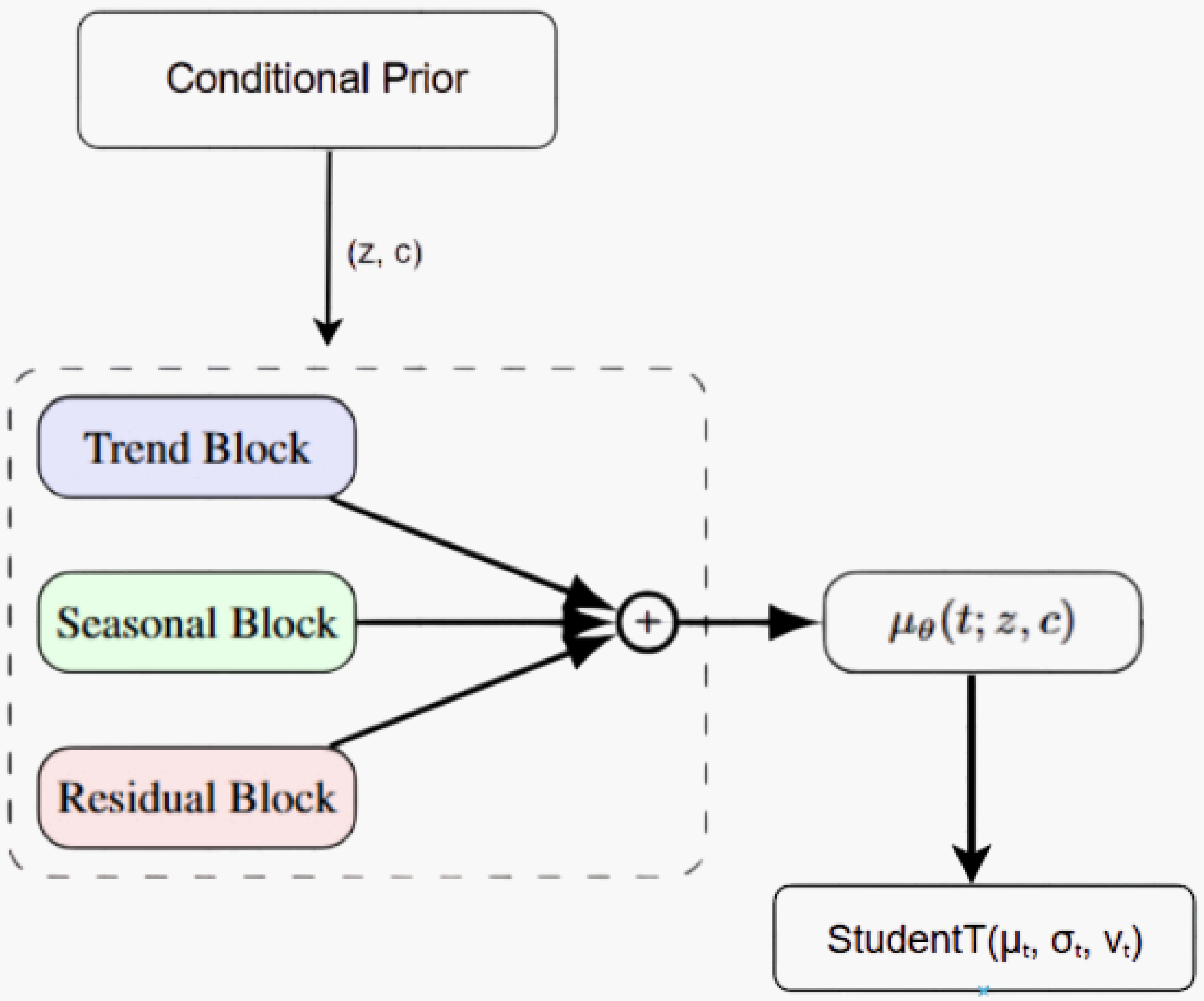
The decoder receives the latent vector z and the condition vector c_t, and produces a 12-month forecast by generating three components:

- Trend:
  - MLP predicts polynomial coefficients → smooth long-term trend
- Seasonality:
  - MLP predicts Fourier amplitudes → periodic patterns via sinusoidal bases
- Residual
  - RNN initialized from [z, c_t] captures short-term shocks and dynamics

The final mean trajectory is given by:

$$\hat{y} = \text{Trend} + \text{Seasonality} + \text{Residual}$$

CT-VAE Model Architecture : Decoder



A Student-t output layer receives the decoder features and predicts (mu, scale, tail) parameters for each timestep

- Defines heavy-tailed predictive likelihood

$$p_\theta(y_t \mid z, c_t) = \mathrm{StudentT}(\nu_t,\ \mu_t,\ \sigma_t).$$

CT-VAE Training Objective : Loss Function

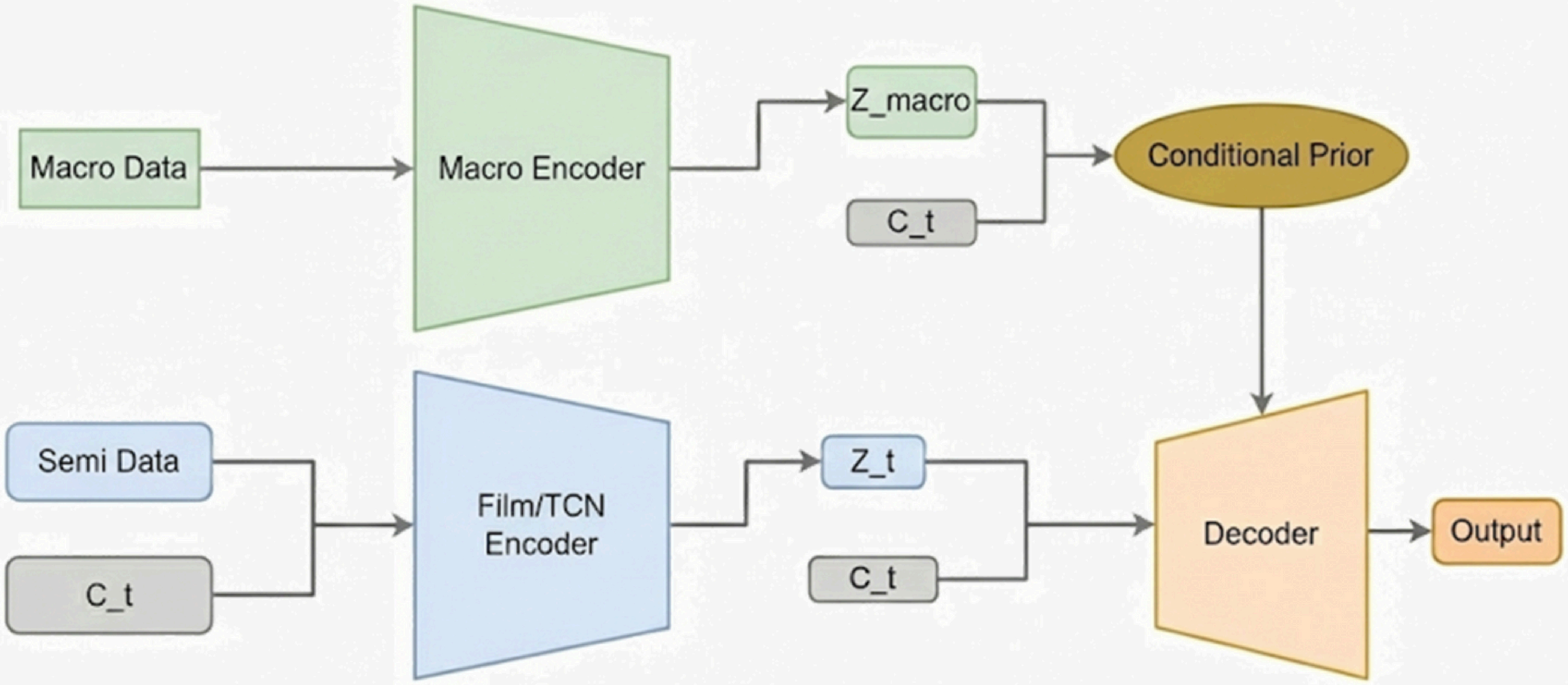
$$\mathcal{L} = \underbrace{-\mathbb{E}_{q_\phi(z \mid x,c)}\left[\log p_\theta(y \mid z, c)\right]}_{\text{Reconstruction (Student-t NLL)}} + \beta \cdot \underbrace{D_{\mathrm{KL}}(q_\phi(z \mid x, c) \parallel p_\psi(z \mid c, z_{\mathrm{macro}}))}_{\text{Macro-conditioned KL}}$$

$$p_\theta(y_t \mid z, c_t) = \mathrm{StudentT}(\nu_t,\ \mu_t,\ \sigma_t) \qquad p_\psi(z \mid c, z_{\mathrm{macro}}) = \mathcal{N}\big(\mu_p(c, z_{\mathrm{macro}}),\ \mathrm{diag}(\sigma_p^2(c, z_{\mathrm{macro}}))$$

Unlike standard cVAE formulations, our decoder replaces the Gaussian likelihood with a Student-t distribution, enabling heavy-tailed modeling of semiconductor shocks such as DRAM crashes and COVID-19 supply disruptions.

In addition, our conditional prior explicitly depends on the macro latent variable z_{macro}, extracted from a frozen TCN-based macro encoder.

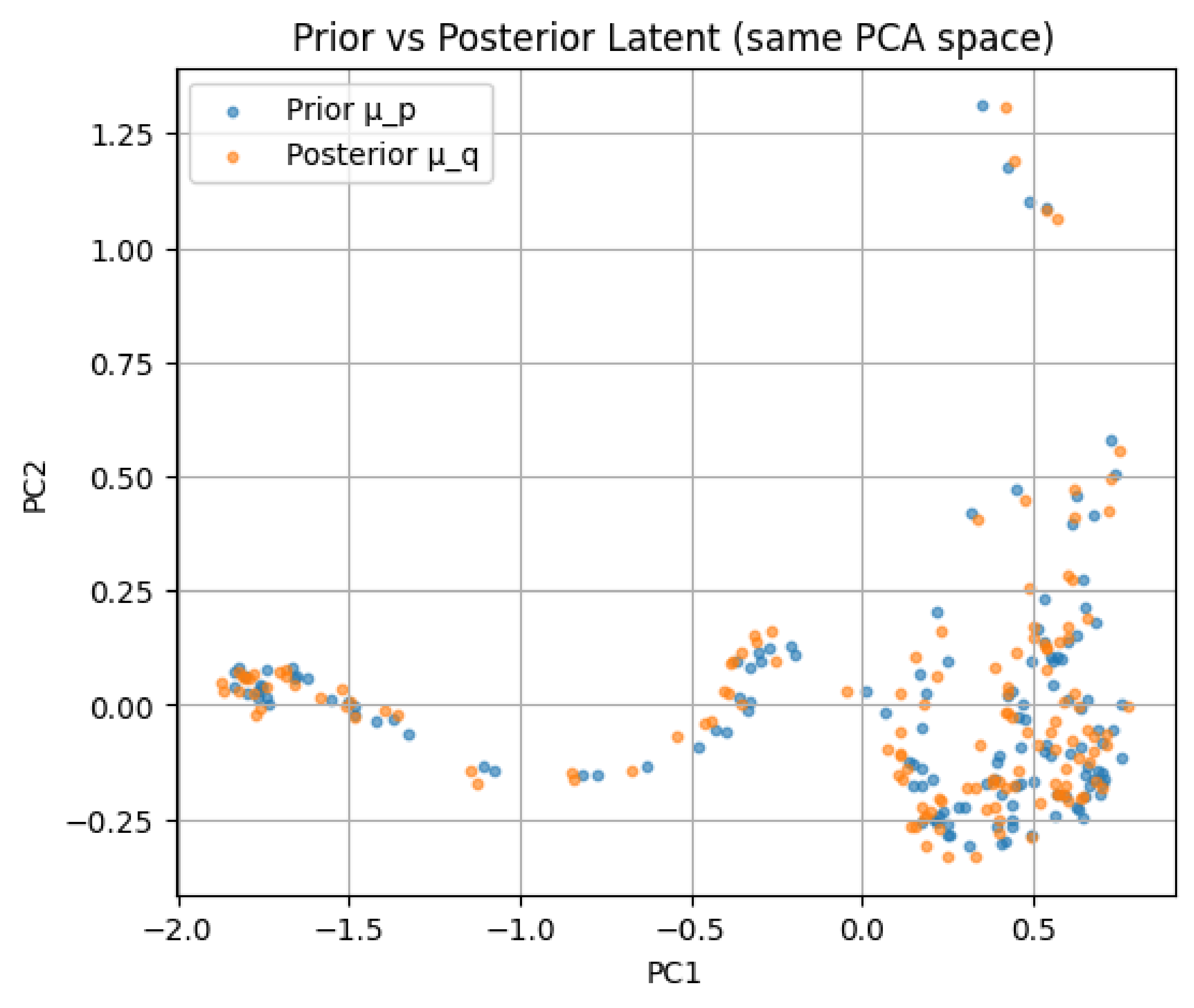
CT-VAE Model Architecture

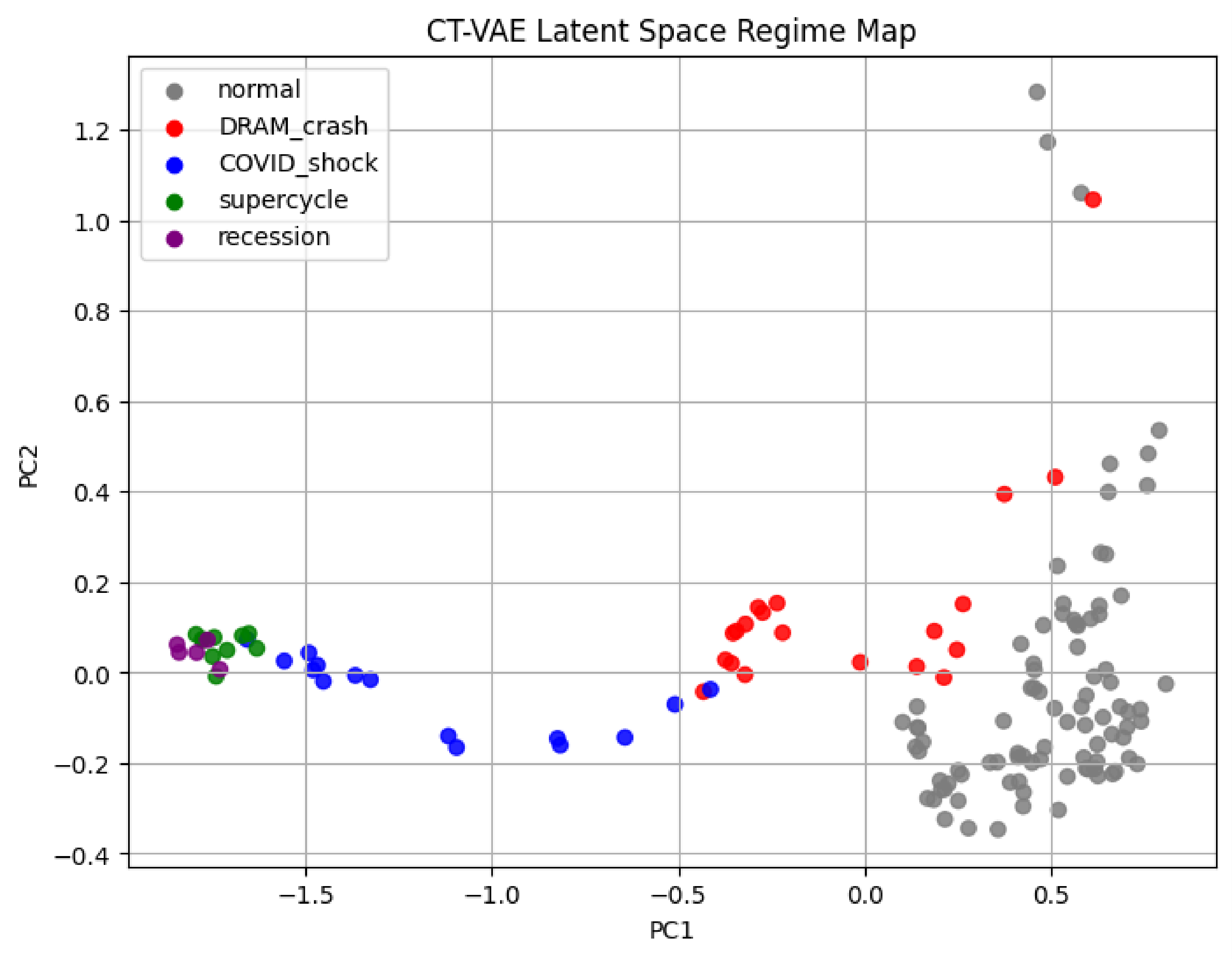


Experiment Result

Latent Space Analysis

Posterior Recon MSE: 0.1972, β = 0.3





Evaluation

Metrics : Rolling-Origin Forecast Evaluation(RMSE, NLL, CRPS, Coverage, CVaR10%)

$\beta = 0.8$

Model	MSE ↓	NLL ↓	CRPS ↓	Coverage @80 ↑	CVaR10%
CT-VAE	0.4449	-0.6540	0.1322	0.8619	-0.4257
cVAE	0.4476	-0.8000	0.1493	0.8158	-0.1765
LSTM	0.3114	0.2521	0.1714	0.8244	-
ARIMA	0.9491	8.7062	0.7676	0.2720	-

The CVaR10% results indicate that CT-VAE is the only model that accurately captures the severity of tail-risk downturns.

CT-VAE performs slightly worse than cVAE on likelihood-based metrics (e.g., NLL), but shows superior tail-risk modeling performance

Ablation

Model Variant	MSE ↓	NLL ↓	CRPS ↓	Coverage@80 ↑	Sharpness ↓	Tail Risk (P_tail) ↓
Full CT-VAE	**0.1989**	**-0.1365**	**0.4495**	**0.51**	**1.2187**	**0.14**
No Student-t	0.1765	-0.0091	0.4277	0.5833	1.3576	**0.18**
No Macro Prior	0.192	**-0.1972**	**0.435**	0.3333	**1.1685**	0.1

No Student-t

- Improves point accuracy (lowest MSE/CRPS) but becomes over-dispersed (high Sharpness, Coverage↑) and shows poor calibration and inflated tail-risk.
- → Good point forecast, bad probabilistic forecast.

No Macro Prior

- Slight gains in accuracy, but Coverage drops sharply (0.33) and intervals become too narrow (under-dispersion).
- → Over-confident model that underestimates uncertainty.

Ablation

Model Variant	MSE ↓	NLL ↓	CRPS ↓	Coverage@80 ↑	Sharpness ↓	Tail Risk (P_tail) ↓
Full CT-VAE	**0.1989**	**-0.1365**	**0.4495**	**0.51**	**1.2187**	**0.14**
No FiLM	0.1892	-0.1861	0.4251	0.4167	**1.1639**	0.09
No Decomposition	0.2033	-0.2659	0.5052	0.51	1.251	**0.22**

No FiLM

- Also improves accuracy but still under-dispersed (Coverage 0.41).
- → More confident but miscalibrated forecasts.

No Decomposition

- Point accuracy deteriorates (higher MSE/CRPS) and tail-risk worsens, despite a decent Coverage score.
- → Removing decomposition breaks structural pattern learning.

Ablation: Summary

Full CT-VAE

- Shows the most balanced performance, with near-perfect calibration (Coverage≈0.51) and stable tail-risk. This serves as the most reliable baseline.

Overall Summary

- Student-t and Decomposition are essential for stable probabilistic forecasting.
- Macro Prior and FiLM do not improve calibration and lead to miscalibrated or over-confident predictions.
- Full CT-VAE remains the most reliable model in terms of balanced accuracy, calibration, and tail-risk.

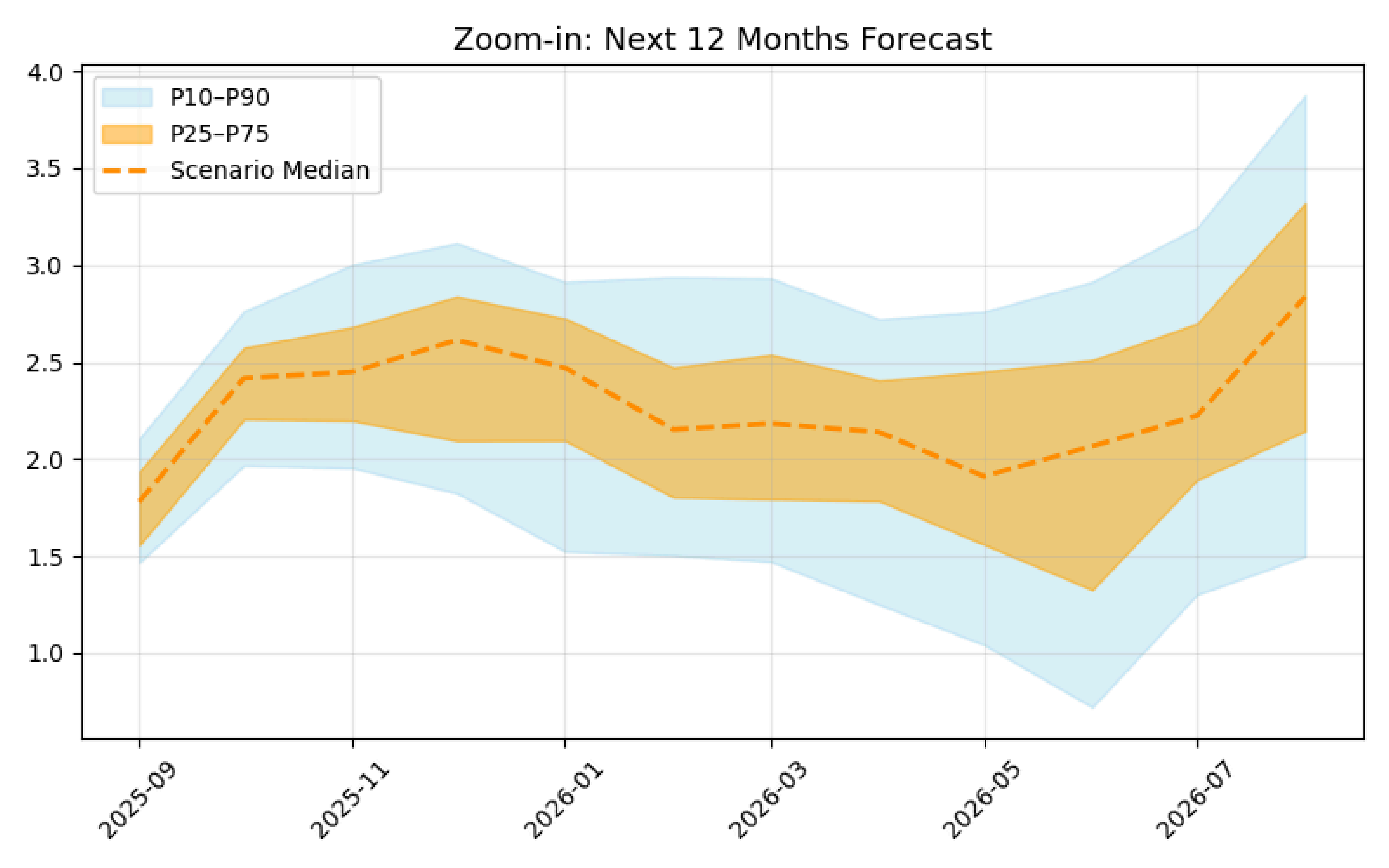
Ablation: Table

Model Variant	**MSE ↓**	**NLL ↓**	**CRPS ↓**	**Coverage@80 ↑**	**Sharpness ↓**	**Tail Risk (P_tail) ↓**
Full CT-VAE	**0.1989**	**-0.1365**	**0.4495**	**0.51**	**1.2187**	**0.14**
No Student-t	0.1765	-0.0091	0.4277	0.5833	1.3576	**0.18**
No Macro Prior	0.192	**-0.1972**	**0.435**	0.3333	**1.1685**	0.1
No FiLM	0.1892	-0.1861	0.4251	0.4167	**1.1639**	0.09
No Decomposition	0.2033	-0.2659	0.5052	0.51	1.251	**0.22**

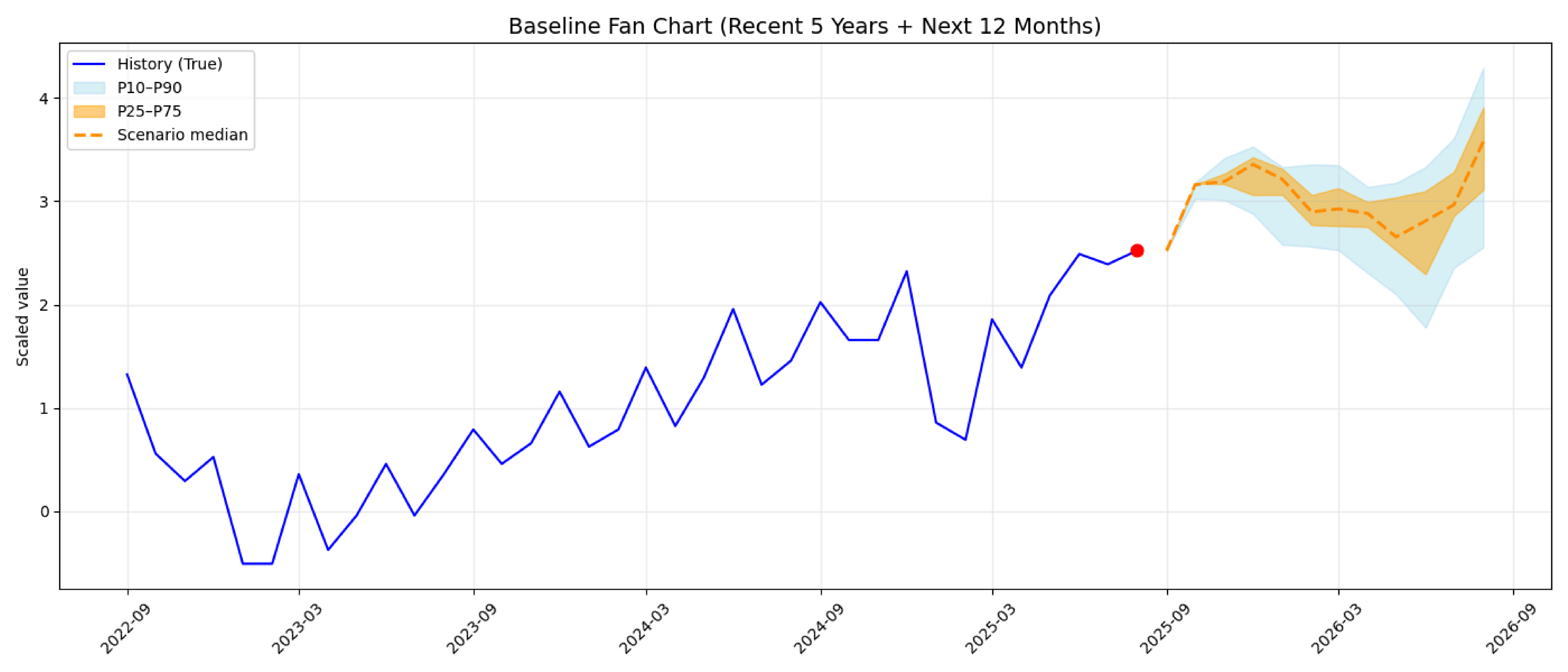
Scenario Generation

Baseline Condition (Current Economic Status) 2025-08-01

Variable	Value
Exchange Rate	1388.91 KRW/USD
GS10	4.26%
PMI	48
ISM	51.4
CLI	100.274



Scenario Generation



Scenario Generation

Risk Metrices

Metric	Baseline
P_up	0.98
Tail Risk (< -10%)	0.02
VaR_10%	0.049
ES_10%	0.011

Interpretation of Risk Metrics

- P_up (Probability of Upward Movement)
  - Likelihood of rise in next 12 months.
  - Current: Strong upward, highly stable.

- P_tail (< –10%)
  - Chance of a severe drop. (more than –10%)
  - Current: Very low downside risk.

Scenario Generation

Risk Metrices

Metric	Baseline
P_up	0.98
Tail Risk (< -10%)	0.02
VaR_10%	0.049
ES_10%	0.011

Interpretation of Risk Metrics

- VaR_10% (Value-at-Risk)
  - Worst outcome in bottom 10%.
  - Current: Still positive → resilient.
- ES_10% (Expected Shortfall)
  - Average of the deepest losses.
  - Current: Also above zero → stable tail.
- Baseline conditions indicate a highly stable upward cycle: very high upside probability, minimal tail risk, and positive outcomes even in extreme scenarios

Scenario Generation

Scenario 1 – Currency Depreciation (Exchange Rate + 10%)

Metric	Baseline	ER +10%
P_up	0.98	0.88
P_tail(<−10%)	0.02	0.08
VaR_10%	0.049	-0.065
ES_10%	0.011	-0.185

KRW Depreciation Scenario

- P_up falls (0.98 → 0.88) — depreciation is learned as a risk signal, not a recovery driver.
- Tail Risk rises (0.02 → 0.08) — large FX spikes historically align with higher volatility.
- VaR/ES turn negative — downside risk dominates.
- Mean forecast weakens slightly — model interprets this as instability, not competitiveness.

Scenario Generation

Scenario 2 – Rate Cut (GS10 -15%)

Metric	Baseline	GS10 -15%
P_up	0.98	0.8
P_tail(<−10%)	0.02	0.04
VaR_10%	0.049	0.012
ES_10%	0.011	-0.082

Why “Rate Cut Only” ≠ Recovery

- The model learned that rate cuts usually happen during downturns, not recoveries
- Up-cycles in the data only appear when multiple indicators improve together (PMI, CLI, ISM, CAPEX).
- So a rate-cut-only scenario looks like a down-cycle signal, raising Tail Risk and lowering P_up compared to the baseline

Scenario Generation

Scenario 3 – Economic Recovery Scenario (GS10 ↓ + PMI ↑ + CLI ↑ + ISM ↑ + ER ↓)

Metric	Baseline	GS10 -15%	Recovery
P_up	0.98	0.8	0.96
P_tail(<−10%)	0.02	0.04	0.02
VaR_10%	0.049	0.012	0.189
ES_10%	0.011	-0.082	0.037

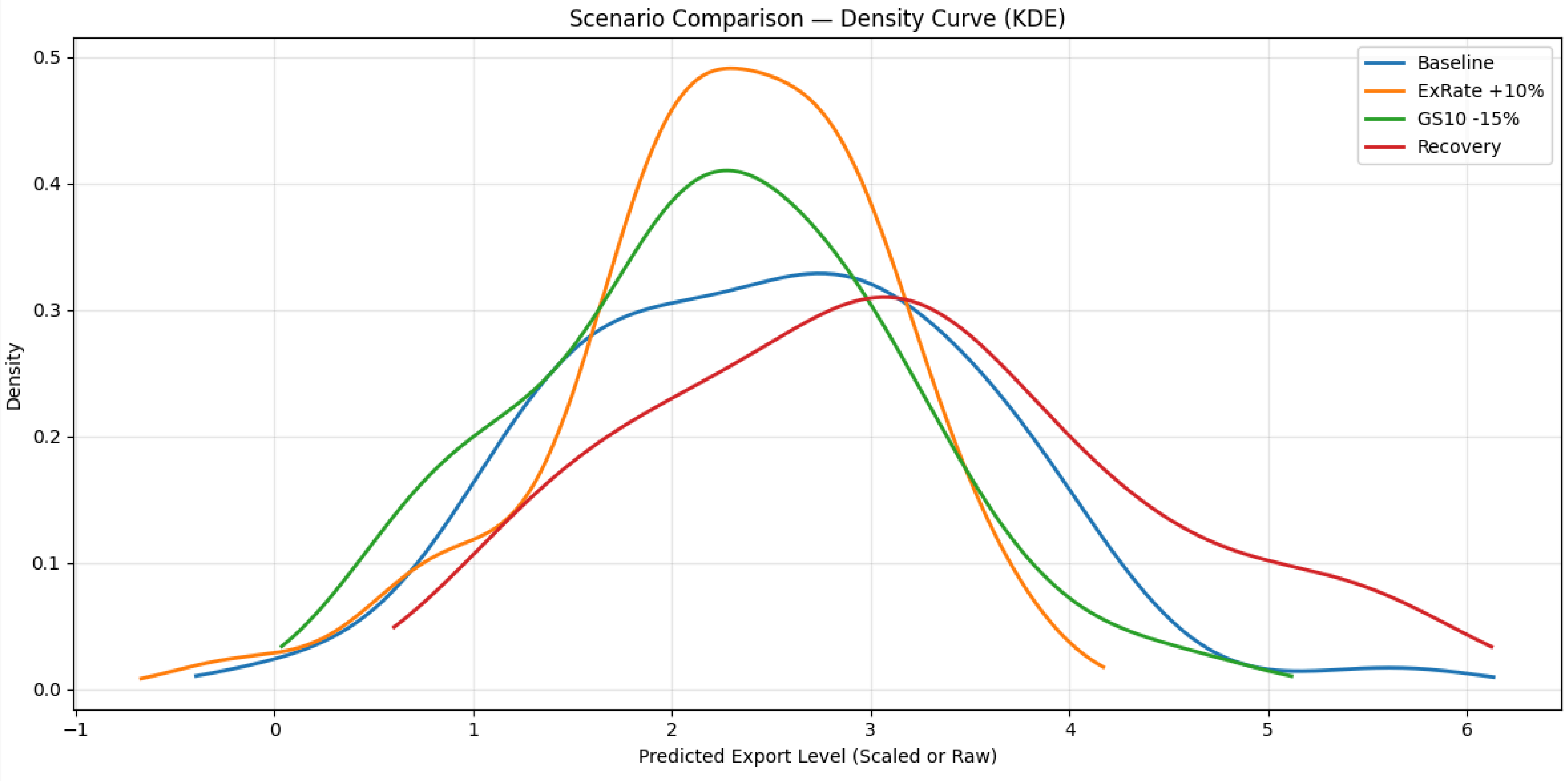
GS10 -15%, PMI +7%, CLI +2%, ISM +5%, ER -6%

True Recovery Scenario

- Recovery usually comes with a cluster of positive signals
- Tail Risk remains and VaR/ES shift to strongly positive
- A rate cut alone signals a downturn, but a coordinated improvement across PMI/CLI/ISM/CAPEX + lower rates forms the historical ‘recovery pattern’ the model recognizes.”

Scenario Generation: Comparative Summary

Metric	Baseline	ER +10%	GS -15%	Recovery
P_up	0.98	0.88	0.8	0.96
P_tail (<–10%)	0.02	0.08	0.14	0.02
VaR_10%	0.049	–0.065	–0.12	0.18
ES_10%	0.011	–0.185	–0.18	0.036



Scenario	Mean (μ)	Std (σ)	5%	95%
Baseline	2.52	1.03	1.08	3.95
ExRate +10%	2.26	0.78	0.8	3.33
GS10 -15%	2.23	0.91	0.74	3.63
Recovery	3.08	1.2	1.26	5.39

Conclusion

Conclusion

Introduced Macro-conditioned Time-VAE for semiconductor forecasting

- Strong predictive and probabilistic performance
- Enabled what-if macro-scenario forecasting

Future Work

- Global Data Augmentation
- Endogenous Scenario Generator
- Regime-Switching CT-VAE
- Multivariate Forecasting (Full Export Basket)

Q&A