Evaluating Short-Term Regime Prediction in the Korean Equity Market: A Large-Scale Study of Decision-Rule Dependence

Yeo-Hoon Yoon, Kyung-Sung Kim

ABSTRACT

This study presents a large-scale benchmark and a decision-rule-aware evaluation protocol for short-term regime classification in the Korean stock market. Reliable evaluation in this setting is difficult because reported performance is sensitive to sample construction, look-ahead bias, class imbalance, and the decision rule used to convert model scores into predicted classes. The benchmark comprises 8,732,898 stock-day observations of 4,225 KOSPI and KOSDAQ common stocks from January 2012 to February 2026, with six three-class regime targets defined by ±5% and ±10% thresholds on 1-, 5-, and 10-trading-day forward returns. The protocol combines survivorship-bias-mitigating sampling, chronological splits with a 20-trading-day purge gap, training-only preprocessing, leakage checks, and full class-distribution reporting. Given severe class imbalance, models are evaluated using macro F1 and up/down directional F1 rather than accuracy alone. We report reference results for two tree-based baselines, LightGBM and XGBoost, and six tabular deep learning models. Under the default argmax rule, the deep learning models are more recall-oriented on extreme regimes, whereas LightGBM is more conservative. However, after validation-based threshold tuning, LightGBM's directional F1 improves substantially, and the relative ranking narrows or reverses on several targets. These results show that apparent model superiority depends materially on the decision rule. The contribution of this study is therefore a reproducible benchmark and evaluation protocol, not a claim of universal model superiority or tradable profitability.

☞ Keywords: Korean stock market; Short-term regime classification; Stock-day benchmark; Decision-rule sensitivity; Imbalanced classification; Tabular deep learning

1. Introduction

Forecasting short-term movements in the stock market is a long-standing problem in financial engineering and asset management. The task is difficult because short-horizon returns are characterized by low signal-to-noise ratios, non-stationarity, regime shifts, and changing market microstructure. Recent financial machine learning research shows that nonlinear models can capture complex interactions among financial predictors that are difficult to represent with linear specifications [4]. However, in short-term stock prediction, reported model performance is often affected not only by model architecture but also by the evaluation design itself.

A central concern is evaluation reliability. When future returns are used to define prediction targets, performance can be overstated if explanatory variables contain future information, if training and test periods overlap, or if preprocessing statistics are estimated using information outside the training period [9]. In addition, short-term regime targets are usually imbalanced because neutral states dominate the sample, whereas large upward or downward movements occur relatively infrequently. Under such conditions, accuracy or default argmax classification may hide a model's ability to detect extreme regimes, so imbalance-aware metrics are necessary for reliable evaluation [6].

In the Korean stock market, prior research has examined stock prediction using index-level data, selected stocks, sentiment variables, and machine learning models. Recent work has also incorporated Korean financial news sentiment into portfolio construction and investment-performance analysis [10]. However, large-scale stock-day benchmarks covering both KOSPI and KOSDAQ common stocks remain limited. This lack of a common

benchmark makes it difficult to determine whether observed performance differences are due to model capability, sample construction, target definition, class imbalance, or the decision rule used to convert model scores into predicted classes.

This study addresses this gap by constructing a large-scale benchmark for short-term regime prediction in the Korean stock market. The benchmark consists of 8,732,898 stock-day observations from 4,225 KOSPI and KOSDAQ common stocks from January 2012 to February 2026. Six three-class regime targets are defined using combinations of prediction horizons — 1, 5, and 10 trading days — and return thresholds — ±5% and ±10%. Each target classifies a stock-day into up, neutral, or down according to its forward return.

Using the same 112 rolling technical indicators, the same chronological train–validation–test split, and the same leakage-aware preprocessing procedure, this study evaluates two tree-based baselines, LightGBM and XGBoost, and six tabular deep learning models, including MLP, ResNet, FT-Transformer, and TabNet variants. Since recent tabular learning research shows that deep learning models and tree-based models can perform differently depending on data and evaluation settings, this comparison is conducted as a standardized benchmark rather than as a claim of universal model superiority [3].

The purpose of this study is to examine how model comparison changes under a decision-rule-aware evaluation protocol. Under the default argmax rule, tabular deep learning models tend to show higher recall on extreme up and down regimes, whereas LightGBM behaves more conservatively and predicts the neutral class more frequently. However, when LightGBM's decision thresholds are tuned on the validation set, its directional F1 improves substantially, and the ranking between model families narrows or reverses on several targets. This result suggests that apparent model superiority in imbalanced regime prediction can depend materially on the decision rule.

Throughout this paper, a regime refers to a threshold-defined short-horizon return state, not a latent macro-financial regime estimated by a state model. The empirical task is therefore short-term regime classification rather than direct trading-strategy evaluation. The prediction signal is interpreted as an end-of-day classification formed after observing day-t information, and issues such as next-day execution, transaction costs, position sizing, and tradable profitability are outside the scope of this study. The contribution of the paper is a reproducible Korean stock-day benchmark and a decision-rule-aware evaluation protocol for comparing short-term regime prediction models.

2. Literature Review

Research on stock market prediction in Korea has examined a range of data sources and modeling approaches, including technical indicators, firm- or industry-level samples, textual information, and machine learning models. Recent Korean-market studies have increasingly incorporated non-price information and machine learning into prediction or portfolio-construction settings. For example, KR-FinBERT-based news sentiment has been incorporated into mean-variance portfolio construction [10], and machine-learning-based stock prediction has been examined in relation to investment strategy performance [8]. These studies show that Korean stock prediction research has moved beyond simple price-based forecasting toward richer data sources and nonlinear modeling. However, much of the existing work still relies on selected indices, specific industries, limited groups of stocks, or restricted sample periods. This limits its use as a common benchmark for comparing model families across the broader Korean equity market.

Recent financial machine learning research also emphasizes that nonlinear models can capture complex interactions among financial predictors that are difficult to represent with linear specifications. Machine learning models, including tree-based and neural network approaches, can improve empirical asset-pricing prediction by modeling nonlinear relations among predictors [4]. In tabular prediction tasks, gradient-boosted decision tree models remain strong baselines, while deep learning models specialized for tabular data have been proposed as competitive alternatives. ResNet-style architectures and FT-Transformer can serve as strong tabular deep learning baselines, while neither tree-based models nor deep learning models are universally superior across all datasets [3]. TabNet has similarly been proposed as an attention-based architecture for tabular learning [1]. These studies suggest that comparisons between tree-based models and tabular deep learning models should be conducted under standardized inputs, splits, and evaluation rules rather than interpreted as general claims of model superiority.

In market-regime prediction, the key issue is not only model

architecture but also how the regime itself is defined. Traditional regime studies often rely on latent-state approaches, such as regime-switching models, Hidden Markov Models, or volatility-state classifications. These approaches are useful for explaining market states retrospectively, but their regime definitions are not always directly aligned with short-horizon prediction tasks. By contrast, this study defines regimes as threshold-based forward-return states — up, neutral, and down — using fixed return thresholds and prediction horizons. This design makes the regime labels operationally interpretable and reproducible, which is important when the purpose is to compare prediction models under a common benchmark rather than to estimate latent macro-financial states.

Another important issue in financial prediction is evaluation reliability. When future returns are used to define labels, reported performance can be overstated if future information leaks into explanatory variables, if training and test periods overlap, or if preprocessing statistics are computed using validation or test information. Time-ordered splits, backtesting-overfitting control, and leakage prevention are emphasized as essential in financial machine learning [9]. In addition, short-term regime classification is typically affected by severe class imbalance because neutral observations dominate the sample, while large upward or downward movements are relatively rare. Under such conditions, accuracy alone can overstate performance, and class-sensitive metrics are needed to evaluate whether models actually detect extreme regimes. A representative discussion of imbalanced learning highlights the limitations of majority-class-biased evaluation [6].

Taken together, recent studies provide useful evidence on Korean stock prediction, financial machine learning, and tabular deep learning. However, they do not provide a common large-scale Korean stock-day benchmark in which sample construction, regime-target definitions, leakage-aware preprocessing, imbalance-aware metrics, and decision rules are fixed simultaneously. This study addresses this gap by treating model comparison as an evaluation-protocol problem rather than as a model-superiority contest.

3. Data Construction and Verification

3.1 Data and Input Variables

The raw dataset covers the period from January 2, 2012 to February 27, 2026 and consists of four data blocks: stock metadata, daily Open-High-Low-Close-Volume (OHLCV) data, pre-market/regular-session/after-market trading microstructure data, and short-selling-related data. The analysis sample is restricted to common stocks listed on the KOSPI and KOSDAQ markets. Product-type securities such as ETFs, ETNs, and REITs are excluded because they follow different trading and short-selling rules. The sample is not limited to currently listed securities; instead, all common stock-day observations with available OHLCV records during the analysis period are included, subject to the coverage of the source data.

The raw datasets are merged using OHLCV records as the base table. Stock metadata are joined by ticker code, while trading-microstructure and short-selling variables are joined by ticker code and trading day. Nine trading-microstructure columns with excessive missingness and limited relevance to the present prediction task are removed. The resulting stock-day table contains 8,732,898 rows, 81 columns, and 4,225 securities. Rows for which horizon-specific forward returns cannot be computed are excluded from both training and evaluation.

Although the raw table includes ticker codes, trading dates, price levels, and a market-cap-equivalent auxiliary variable derived from price and outstanding shares, these variables are not used as direct model inputs. The input feature set is restricted to 112 rolling technical indicators computed from OHLCV, trading-microstructure, and short-selling data. These indicators are calculated over short-, medium-, and long-term rolling windows of 3, 5, 10, 20, and 60 trading days to reduce dependence on absolute price or volume scales across stocks.

The 112 input variables are grouped into six categories: price and trend indicators, momentum indicators, volatility indicators, volume and trading-value indicators, session-structure indicators, and short-selling indicators. Price and trend indicators include returns, moving-average deviations, moving-average crossovers, and Bollinger Z-scores. Momentum indicators include RSI, Stochastic K, Williams %R, and cumulative returns. Volatility indicators include true range, ATR, return volatility, and intraday range. Volume and value indicators include volume deviation, trading-value deviation, turnover, OBV, and PVT. Session-structure indicators summarize pre-market, regular-session, and after-market volume and value shares. Short-selling indicators include short-selling value share, short-selling pressure, and deviations from the average short-selling

price. The complete list of indicators, formulas, and rolling windows is provided in Appendix C.

To preserve temporal consistency, all input variables for stock-day t are computed only from information available by the end of day t. This includes after-market session variables and official short-selling statistics, which are finalized after the regular close. The prediction signal is therefore treated as an end-of-day signal formed after observing day-t information. Since regime labels are defined using forward returns from day t onward, no input variable incorporates information dated later than day t. This construction is intended to prevent look-ahead bias and maintain a time-consistent prediction setting [9]. All eight reference models — two tree-based baselines and six tabular deep learning models — use the same 112 input variables.

3.2 Data Verification

Because this study constructs a benchmark from a large-scale stock-day panel, data integrity and temporal consistency are verified before conducting the reference model comparison. We first examine whether the merged dataset preserves the intended sample structure throughout preprocessing. The final big table contains 8,732,898 rows, 81 columns, and 4,225 securities over the period from January 2, 2012 to February 27, 2026. The row count remains unchanged after the technical-indicator computation step, indicating that feature construction does not introduce unintended sample loss.

We then verify the chronological structure of the training, validation, and test samples. The split is designed so that the maximum training date precedes the minimum validation date, and the maximum validation date precedes the minimum test date. In addition, a 20-trading-day purge gap is imposed around each split boundary to reduce the risk of temporal leakage and sample overlap, which are common concerns in financial time-series prediction [9]. This design ensures that model estimation and evaluation are conducted under a strictly time-ordered setting.

To further confirm that the split procedure does not generate duplicated evaluation units, we examine overlaps in (ticker, trading day) pairs across the training, validation, and test samples. The overlap counts between training and validation, validation and test, and training and test are all zero. This verification is important because stock-day panels can otherwise contain repeated or adjacent observations that blur the distinction between estimation and evaluation periods.

We also verify that variables related to future returns are used only for regime-label construction and are not included in the model input set. The input variables are restricted to features computed from information available up to the end of each stock-day. In addition, missing-value imputation and scaling parameters are estimated only from the training sample and then applied unchanged to the validation and test samples. This training-only preprocessing procedure prevents validation or test information from entering the input distribution during model preparation.

Finally, the security-level treatment of the panel is summarized in Table 1. The benchmark does not apply post-hoc filtering based on current listing status. Instead, all common stock-day observations with available OHLCV records in the source data are included as candidates for the panel, subject to the coverage limits of the source data. This treatment is intended to reduce survivorship bias while maintaining a transparent and reproducible sample-construction rule.

(Table 1 Security-day-level sample treatment)

Item	Treatment
Delisted securities	Included if OHLCV observations exist in the source data
Currently-listed-only restriction	Not applied (no post-hoc survival filtering)
Rows with infeasible forward returns	Excluded from both training and evaluation
Newly listed securities (initial 60 trading days)	Rows that do not satisfy long-window indicators are excluded from the evaluation sample
Trading halts and missing values	Rolling technical indicators are computed strictly from past observations; missing values during training are imputed using training-sample statistics

4. Benchmark Protocol

In this study, a regime denotes a threshold-defined forward-return state (up, neutral, or down) rather than a latent, model-inferred market state such as a regime-switching state [5]; each three-class label is determined solely by whether a forward return crosses fixed thresholds.

We design six regime classification targets as combinations of three prediction horizons (1, 5, and 10 trading days) and two thresholds (±5%, ±10%). For each stock and trading day, the future return over the prediction horizon is computed as the close-to-close percentage change between the current day's close and the close at the end of the horizon. A three-class label is then assigned: up if the future return exceeds the positive threshold, down if it falls below the negative threshold, and neutral otherwise. The six combinations are 1-day ±5%, 1-day ±10%, 5-day ±5%, 5-day ±10%, 10-day ±5%, and 10-day ±10%, designated as target IDs regime_1d_thr5, regime_1d_thr10, regime_5d_thr5, regime_5d_thr10, regime_10d_thr5, and regime_10d_thr10, respectively. The 10-day ±5% target is used as the main target because it provides a comparatively balanced extreme-regime distribution (both extreme classes occupy approximately 18% of the sample) and represents an operationally interpretable short-horizon regime definition. The 1-day ±10% target has extreme-class proportions of approximately 1% or less, and is treated as a challenge target for rare-regime capture in the analysis below. Because the 5- and 10-trading-day targets are defined on overlapping forward-return windows, consecutive labels for the same stock are not independent; the evaluation treats each stock-day as one observation and does not explicitly model this within-split dependence. Future work could further address this issue using block-bootstrap procedures or non-overlapping resampling designs [9].

The test-set class distribution of the six targets defined above is reported in Table 2. As the threshold strengthens from 5% to 10%, the extreme-class ratios fall sharply, and as the horizon lengthens from 1 to 5 and 10 trading days, the extreme-class ratios increase again. The training and validation samples exhibit essentially the same pattern; absolute counts are summarized in Appendix A. In particular, the 1-day ±10% target has extreme-class ratios of 0.42% (down) and 0.98% (up), making it the most rare among the six targets, and is classified as a challenge target in this study.

(Table 2 Test-set class distribution by regime target)

Target	Down	Neutral	Up
1-day ±5%	53,152 (2.74%)	1,814,666 (93.58%)	71,364 (3.68%)
1-day ±10%	8,075 (0.42%)	1,912,081 (98.60%)	19,026 (0.98%)
5-day ±5%	240,161 (12.49%)	1,433,452 (74.57%)	248,686 (12.94%)
5-day ±10%	66,164 (3.44%)	1,757,425 (91.42%)	98,710 (5.13%)
10-day ±5% (main)	358,826 (18.87%)	1,183,035 (62.23%)	359,358 (18.90%)
10-day ±10%	133,023 (7.00%)	1,596,284 (83.96%)	171,912 (9.04%)

These class-distribution patterns directly motivate the choice of evaluation metrics. Prior research on multiclass classification evaluation points out that simple accuracy can overestimate the discriminative power of a model in problems with a dominant majority class, and recommends reporting class-wise precision, recall, and their harmonic mean F1 [11]. A representative survey on imbalanced learning similarly emphasizes that, when detecting extreme classes is crucial, accuracy alone is vulnerable to majority-class bias and distribution-robust metrics such as F1, G-mean, and AUC should be used [6]. In our benchmark, the regime targets generally have a high neutral-class share, and especially in short-horizon, high-threshold targets the neutral share reaches around 90%, indicating severe class imbalance; thus simple accuracy may not adequately reflect the model's ability to identify extreme classes. We therefore report four metrics together. First, macro precision and macro F1 over the three classes (up, neutral, down). Second, a binary directional F1 with the up class as positive and the neutral and down classes combined as negative, hereafter the up directional F1. Third, a binary directional F1 with the down class as positive and the neutral and up classes combined as negative, hereafter the down directional F1. Fourth, the arithmetic mean of the up directional F1 and the down directional F1, which we refer to as the mean directional F1.

The mean directional F1 mitigates the problem of overestimating overall performance when predictions concentrate on the neutral class, and is used as a metric that separately evaluates the ability to capture the up and down extreme classes. As a trivial reference point,

a majority-class classifier that always predicts the dominant neutral class attains an up and down directional F1 of zero by construction; any positive directional F1 therefore reflects discrimination above this floor. For brevity, in tables we abbreviate the mean directional F1 as mDir-F1 and its difference as ΔmDir-F1. Since F1 alone is not sufficient to characterize how a model behaves, this benchmark additionally reports precision and recall for the extreme classes. This is consistent with the recommendation of the classification evaluation literature that the same F1 value with "high precision and low recall" implies a different operational meaning than "low precision and high recall," and that precision and recall should be reported separately to expose such differences [11].

5. Reference Baseline Protocol

All reference models are trained, validated, and evaluated using the same data split, which together with the feature definitions and random seeds constitutes the evaluation protocol. After applying the purge gap, the training period is 2012-01-02 to 2021-12-02 (5,124,124 rows, 2,443 trading days), the validation period is 2022-02-03 to 2023-11-29 (1,411,677 rows, 451 trading days), and the test period is 2024-01-30 to 2026-02-27 (1,943,407 rows, 504 trading days). Note that the 1,943,407 rows correspond to the test period immediately after the time-series split. The final per-target evaluation sample is then obtained by excluding (i) rows at the end of the sample period for which the horizon-specific forward return is not yet observable — so that the number of excluded rows grows with the prediction horizon — and (ii) early rows that lack sufficient history for the long-window (up to 60-trading-day) indicators. Because the first exclusion is horizon-dependent, the evaluation sample size differs slightly across the six targets. All numbers reported in Table 2, Table 3, and Appendix B are based on this final per-target model-evaluation sample. Each model is selected on the validation set based on macro precision, and final reporting on the test set uses the model with the highest validation macro precision. The choice of macro precision as the (disclosed) model-selection criterion reflects the priority of restraining excessive false positives in the extreme classes, given the heavily imbalanced class distribution of the regime targets. As a sensitivity check, we also repeated model selection using validation macro F1 and validation mean directional F1; the main ranking between the best deep learning model and the best baseline remained unchanged.

The reference models include two strong tree-based baselines for tabular data — LightGBM and XGBoost, representative gradient-boosted decision tree baselines for tabular prediction [2], [7] — together with six tabular deep learning models representing the three axes of depth, residual connection, and attention. The deep learning group consists of two simple multilayer perceptrons (MLP-2Layer, MLP-3Layer), two residual-connection variants (MLP-ResNet, ResNet), and two attention-based models, FT-Transformer and TabNet, which represent recent tabular deep learning architectures [1], [3]. All models use the same 112 rolling technical indicators as inputs and the same training/validation/test splits. The deep learning optimizer is AdamW, and the loss function is class-weighted cross-entropy; the full hyperparameter settings for all eight models are reported in Appendix D. As a trivial reference floor under the protocol, a majority-class classifier attains zero up and down directional F1 by construction; the learned baselines below are therefore read relative to this floor rather than as competitors for a superiority claim.

The default training epoch is 20, but in some models we observe a phenomenon in which validation performance gradually improves in the latter half of training (hereafter, late-peaking). To conservatively check this latter-half stability, only the model-target runs that satisfy two pre-specified conditions — first, the best epoch occurring at epoch 18 or later, and second, the last-epoch validation macro precision being within 0.002 of the best — are extended up to 30 epochs. The extension only updates the best deep model on the 1-day ±10% target from FT-Transformer to ResNet, while on the other five targets the best model and test macro F1 remain unchanged. All results in this study therefore use the model weights at the epoch with the highest validation macro precision among models trained for either 20 or 30 epochs under the above criteria.

Inspecting the epoch-wise learning curve of the MLP-2Layer model on the main target (10-day ±5%), the training loss decreases monotonically with training, and validation macro F1 and mean directional F1 quickly enter a stable region after the first few epochs and remain low-variance thereafter. This suggests that learning stabilizes early on the main target and that results are not particularly sensitive to the choice of training epoch. Due to space constraints, this paper presents only a qualitative description of the learning curve.

6. Reference Baseline Results

Table 3 reports reference test performance for each of the six regime targets, contrasting the best deep learning model and the best tree baseline selected by validation macro precision. The two tree

baselines, LightGBM and XGBoost, perform comparably: across the six targets their test macro F1 differs by within about 0.03 and neither dominates (Appendix F), so LightGBM is reported as the representative tree baseline in the main tables and the full XGBoost results are given in Appendix F. The full validation-set macro F1 of all reference models is reported in Appendix E, where a deep learning model attains the highest validation score on every target; because Table 3 selects by validation macro precision (Section 5) rather than the macro F1 shown there, the Best deep model differs from the highest-F1 column on 5-day ±5% and 10-day ±10%. The best tree baseline behaves, under the default argmax rule, as a conservative, neutral-centered classifier with low recall on the extreme classes, which limits macro F1; test macro F1 ranges from approximately 0.30 to 0.36. Under the argmax rule the GBDT baseline tends toward a majority-biased classifier on the extreme classes. We report the differences between the two model groups in macro F1 and mean directional F1 (ΔmF1, ΔmDir-F1) descriptively; because the absolute magnitudes of these differences depend on extreme-class proportions, we describe the differences without interpreting them as evidence of model superiority. All metrics in this section are classifier-level regime-identification performance, not evidence of tradable profitability; execution assumptions such as next-day open prices, transaction costs, and position sizing are outside the scope of this benchmark.

(Table 3 Best deep learning vs. best baseline model on test set, with slice robustness (5-seed mean))

Target	Best deep	Deep mF1	Base mF1	Deep mDir-F1	Base mDir-F1	ΔmF1 / ΔmDir-F1	Slice wins (yr/mkt)
1-day ±5%	FT-Transformer	0.3923	0.3169	0.1712	0.0279	+0.0754 / +0.1433	3/3, 3/3
1-day ±10% (challenge)	ResNet	0.3346	0.3135	0.0652	0.0044	+0.0211 / +0.0608	3/3, 3/3
5-day ±5%	FT-Transformer	0.4426	0.3551	0.3033	0.1575	+0.0875 / +0.1458	3/3, 3/3
5-day ±10%	FT-Transformer	0.3962	0.2983	0.1943	0.0182	+0.0979 / +0.1761	3/3, 3/3
10-day ±5% (main)	MLP-2Layer	0.4457	0.3616	0.3323	0.1553	+0.0841 / +0.1770	3/3, 3/3
10-day ±10%	FT-Transformer	0.4120	0.3245	0.2391	0.0605	+0.0875 / +0.1786	3/3, 3/3

Reporting key slice-level reference numbers, the main target (10-day ±5%) shows the following pattern under the argmax rule. In yearly slices, deep learning's mean directional F1 is 0.3476 (2024), 0.3199 (2025), and 0.3220 (the partial 2026 period through February), staying above 0.31 in all three slices, while the baseline reaches only 0.1652, 0.1478, and 0.1853, respectively. In market slices, deep learning's mDir-F1 on the main target is 0.3052 (KOSPI) and 0.3578 (KOSDAQ), above the baseline's 0.1143 and 0.1890 in both markets. A similar pattern holds for the 1-day ±5% target; for the challenge target (1-day ±10%), absolute values are small (around 0.07) but the direction of the slice-level difference is consistent across slices. The market-slice analysis evaluates the same model trained on the entire-market panel on each market subsample, rather than retraining a separate model per market. These slice comparisons are reported under the argmax rule; their dependence on the decision rule is the subject of Section 7.

The difference between the two model groups under argmax reflects a difference in the precision–recall trade-off rather than absolute superiority. On the main target 10-day ±5%, LightGBM attains relatively high extreme-class precision (0.3996 up, 0.4481 down) but low extreme-class recall (0.0498 up, 0.1476 down). MLP-2Layer attains precision 0.2785 / recall 0.2909 on the up class and precision 0.3192 / recall 0.4692 on the down class, trading some precision for higher extreme-class recall and F1. Extreme-class F1 thus moves from 0.0886 to 0.2846 for the up class and from 0.2220 to 0.3800 for the down class between the two reference models. Relative to a majority-class classifier, which attains zero directional F1 by construction, both learned baselines exhibit non-trivial directional discrimination above the floor; the magnitudes, however, are modest. This pattern recurs across all six targets (see Appendix B for the detailed confusion matrix on the main target).

On the 1-day ±10% challenge target, although the extreme-class share is below 1%, the gap in extreme-class recall between the two

reference models is the most pronounced among the six targets. LightGBM's extreme-class recall is 0.71% on the up class and 0.25% on the down class, whereas the best deep learning model ResNet attains an up recall of 37.31% and a down recall of 37.27%. At the same time, the deep model's extreme-class precision on this target is very low (0.0448 up, 0.0383 down). This illustrates that class-weighted loss can increase false positives while lifting recall, and that in a regime region almost invisible to accuracy-based or argmax classification the choice of operating point dominates the apparent result. The result on this target should therefore be read not as a practical trading signal but as a motivation for the decision-rule sensitivity analysis in Section 7.

7. Protocol Sensitivity

This section examines how stable the reference results of Section 6 are to the choice of a single random seed and — most importantly for evaluation — to the baseline's decision rule. Slice stability was addressed in Section 6 (Table 3 and the accompanying narrative); here we present two remaining checks in turn — training stability and decision-rule sensitivity — and consolidate the results in Table 4.

To check that the single-seed results are not coincidental, we repeatedly trained the best deep learning model for each of the six regime targets with five different random seeds. Across all six targets, the standard deviation of the five-run test macro F1 ranges from 0.0010 to 0.0060, indicating stable training; on the main target 10-day ±5%, the five-run test macro F1 is 0.4457 with a standard deviation of 0.0024, below 0.003 (see the second column of Table 4).

Most importantly, to test whether the conservative argmax behavior of the baseline is an artifact of the decision rule, we conducted threshold tuning on LightGBM for all six regime targets: we grid-searched the up-class probability threshold and the down-class probability threshold on the validation set so as to maximize mean directional F1, and applied the optimal thresholds on the test set. Under the tuned rule a stock-day is labeled up if its up-class probability exceeds the up threshold and down if its down-class probability exceeds the down threshold; if both thresholds are exceeded, the class with the larger margin above its threshold is chosen, and if neither is exceeded the prediction defaults to neutral. The validation-best thresholds were (up, down) = (0.20, 0.15) for 10-day ±5%, (0.15, 0.15) for 5-day ±5%, (0.12, 0.12) for 10-day ±10%, and (0.10, 0.10) for each of 1-day ±5%, 1-day ±10%, and 5-day ±10%, and the resulting macro F1 and mean directional F1 of the tuned LightGBM are reported in the last column of Table 4. Threshold tuning substantially improved the extreme-class recall and mean directional F1 of LightGBM on all six targets, indicating that argmax-based evaluation can understate the directional potential of the baseline. For example, on the 1-day ±10% challenge target the mean directional F1 of LightGBM rose from 0.0044 under argmax to 0.1206 after tuning, and on 10-day ±10% from 0.0605 to 0.2391.

The cross-model comparison after threshold tuning can be summarized as follows. On the main target (10-day ±5%), the best deep model still exceeds tuned LightGBM in both macro F1 and mean directional F1 (0.4457 vs. 0.3924 in macro F1 and 0.3323 vs. 0.3287 in mean directional F1, based on Tables 3 and 4). Across the remaining targets, threshold tuning reverses the mean-directional-F1 ranking on three targets (1-day ±5%, 1-day ±10%, and 5-day ±10%); on 5-day ±5% the deep model remains slightly higher (0.3033 vs. 0.2962), while on 10-day ±10% the two models are virtually tied (0.2391 vs. 0.2391). In macro F1, tuned LightGBM exceeds the best deep model on four of the six targets (1-day ±5%, 1-day ±10%, 5-day ±10%, and 10-day ±10%). The relative ordering between the two model families therefore depends materially on the decision rule. For a benchmark, the operational implication is that results must be reported under an explicit, common decision rule: comparing one family under argmax against another under tuned thresholds — or vice versa — can invert conclusions. This is the central motivation for fixing a decision-rule-aware evaluation protocol. This threshold-tuning exercise is designed as a diagnostic analysis of decision-rule dependence, not as a fully optimized comparison across all model families. Under class imbalance, the optimal decision threshold depends on class priors and misclassification costs and is not fixed at the argmax of the predicted probabilities. Adjusting the decision threshold is therefore a recognized component of imbalanced classification [6]. We therefore tune only the baseline's decision threshold as a controlled probe of whether the argmax ranking is stable. Because the deep learning models are not threshold-tuned in the same manner, the tuned-LightGBM results should be interpreted as evidence that argmax-based rankings are sensitive to the decision rule, rather than as a final ranking of model families.

(Table 4 Robustness summary: multi-seed stability and LightGBM threshold tuning)

Target	5-seed mF1 (mean ± sd)	LightGBM tuned (mF1 / mDir-F1)
1-day ±5%	0.3923 ±	0.4405 / 0.1975

Target	5-seed mF1 (mean ± sd)	LightGBM tuned (mF1 / mDir-F1)
	0.0050	
1-day ±10% (challenge)	0.3346 ± 0.0010	0.4098 / 0.1206
5-day ±5%	0.4426 ± 0.0043	0.4284 / 0.2962
5-day ±10%	0.3962 ± 0.0045	0.4371 / 0.2109
10-day ±5% (main)	0.4457 ± 0.0024	0.3924 / 0.3287
10-day ±10%	0.4120 ± 0.0060	0.4200 / 0.2391

8. Ethical Considerations

The benchmark is constructed entirely from market-level data — daily OHLCV, intraday trading-microstructure aggregates, and short-selling statistics — together with security metadata, and contains no personally identifiable information about individual investors, so record-level privacy risk is minimal. The underlying market data are obtained from the Korea Exchange (KRX) and standard market-data providers and are used under their respective terms; the 112 input features are derived from these series using standard, publicly documented technical-indicator formulas, and no proprietary third-party series are redistributed.

To limit selection and survivorship bias, the panel includes all KOSPI and KOSDAQ common stock-day rows for which observations exist in the source data, including securities delisted during the study period, rather than only currently listed names (Section 3). The severe class imbalance of the regime targets is handled explicitly through full class-distribution reporting and imbalance-aware metrics — macro F1 and up/down directional F1 — rather than accuracy alone, so that the benchmark does not reward majority-class prediction. Residual coverage limits inherited from the source data are stated in Section 3 so that users can interpret results within those bounds.

Because the prediction setting evaluates end-of-day directional regime classification rather than executable trades, the benchmark and its reference results are intended for methodological evaluation, not as investment advice or a tradable signal, and the precision–recall and decision-rule analyses are reported to characterize classifier behavior rather than to recommend trading strategies. Users should account for transaction costs, execution constraints, and applicable regulations before drawing any operational conclusion.

9. Conclusion and Availability

We constructed a large-scale benchmark with leakage-aware construction and verification for short-term regime classification on a Korean stock-day panel, defined six operationally interpretable regime targets and an evaluation protocol for imbalanced extreme-regime detection, and reported reference results for two tree baselines and six tabular deep learning models under identical inputs and splits. Three points summarize the study. First, the construction and verification pipeline — survivorship-bias-mitigating sampling, time-ordered splits with a purge gap, training-only preprocessing, and stepwise overlap and leakage checks — provides a reproducible basis on which future Korean regime-classification work can be compared. Second, the evaluation protocol, by reporting up/down directional F1 alongside macro F1 and class-wise precision and recall, exposes extreme-regime detection behavior that aggregate accuracy hides under severe class imbalance. Third, under this protocol the relative ordering of the two model families is decision-rule-dependent.

Under the default argmax rule the tabular deep learning reference models are more recall-oriented on the extreme classes while the tree baseline is more conservative and precision-oriented; for example, on the main target 10-day ±5% the up- and down-class recalls of the baseline are 4.98% and 14.76%, while those of the deep model are 29.09% and 46.92% (Appendix B), and on the 1-day ±10% challenge target the baseline's extreme-class recall remains below 1% whereas the deep model attains roughly 37% (Section 6). Once the baseline's decision thresholds are tuned, however, the gap narrows or reverses: tuned LightGBM exceeds the best deep model in mean directional F1 on three targets, remains slightly below on two, and is virtually tied on one, and exceeds it in macro F1 on four of six targets. We therefore make no claim of universal model superiority.

The central, reproducible message of this study is that credible comparison requires a common decision rule and a leakage-controlled protocol: the apparent advantage of one model family can be an artifact of the evaluation choice rather than a property of the model. The benchmark and protocol presented here provide a useful basis for future studies on Korean stock market regime classification and highlight the importance of evaluating directional-regime identification alongside conventional classification metrics. The

empirical results should not be interpreted as evidence of directly tradable profitability.

Availability. The benchmark protocol is specified for reproduction through the construction and verification rules in Section 3, the technical-indicator definitions and rolling windows in Section 3.1 and Appendix C, the model hyperparameters in Appendix D, the six regime-target definitions in Section 4, and the fixed chronological split dates and random seeds in Section 5, with preprocessing statistics fit on the training sample only. Access to the underlying raw market data depends on the terms of the Korea Exchange and commercial data providers, and the implementation code for the construction pipeline, regime labeling, evaluation protocol, and decision-rule tuning is available from the authors for research use upon reasonable request.

References

[1] Arik, S. Ö., & Pfister, T. (2021). TabNet: Attentive interpretable tabular learning. Proceedings of the AAAI Conference on Artificial Intelligence, 35(8), 6679–6687.

[2] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.

[3] Gorishniy, Y., Rubachev, I., Khrulkov, V., & Babenko, A. (2021). Revisiting deep learning models for tabular data. Advances in Neural Information Processing Systems, 34, 18932–18943.

[4] Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. The Review of Financial Studies, 33(5), 2223–2273.

[5] Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. Econometrica, 57(2), 357–384.

[6] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263–1284.

[7] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30, 3146–3154.

[8] Kim, S. (2023). A comparison of stock price prediction for global automotive companies using machine learning models [in Korean]. Journal of the Korean Data Analysis Society, 25(1), 249–263.

[9] López de Prado, M. (2018). Advances in financial machine learning. Wiley.

[10] Park, J., Jung, M., Kim, H., & Kim, S. (2024). An analysis of investment performance for portfolio selection models reflecting news sentiment analysis: Focusing on the Korean stock market [in Korean]. Journal of the Korean Operations Research and Management Science Society, 49(4), 57–72.

[11] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. Information Processing & Management, 45(4), 427–437.

Appendix

A. Class Distribution by Target (Training and Validation)

Target	Split	Down	Neutral	Up
1-day ±5%	Train	163,061 (3.18%)	4,748,406 (92.67%)	212,657 (4.15%)
	Valid	42,814 (3.03%)	1,319,527 (93.47%)	49,336 (3.49%)
1-day ±10%	Train	26,000 (0.51%)	5,040,971 (98.38%)	57,153 (1.12%)
	Valid	4,448 (0.32%)	1,395,145 (98.83%)	12,084 (0.86%)
5-day ±5%	Train	738,715 (14.42%)	3,616,007 (70.57%)	769,402 (15.02%)
	Valid	213,694 (15.14%)	1,021,510 (72.36%)	176,473 (12.50%)
5-day ±10%	Train	212,731 (4.15%)	4,608,214 (89.93%)	303,179 (5.92%)
	Valid	59,640 (4.22%)	1,288,594 (91.28%)	63,443 (4.49%)
10-day ±5%	Train	1,085,959 (21.19%)	2,942,804 (57.43%)	1,095,361 (21.38%)
	Valid	318,884 (22.59%)	836,376 (59.25%)	256,417 (18.16%)
10-day ±10%	Train	419,180 (8.18%)	4,173,593 (81.45%)	531,351 (10.37%)
	Valid	124,681 (8.83%)	1,174,660 (83.21%)	112,336 (7.96%)

Note: Column definitions — Target is the regime target; Split indicates the training or validation period; Down, Neutral, and Up give the count and, in parentheses, the share of each regime class within that split.

B. Extreme-Class Confusion Matrix on the Main Target (10-day ±5%)

Class	Model	TP	FN	FP	TN	Precision	Recall	F1
Up	MLP-2Layer (deep)	104,524	254,834	270,722	1,271,139	0.2785	0.2909	0.2846
Up	LightGBM (base)	17,906	341,452	26,900	1,514,961	0.3996	0.0498	0.0886
Down	MLP-2Layer (deep)	168,369	190,457	359,064	1,183,329	0.3192	0.4692	0.3800
Down	LightGBM (base)	52,947	305,879	65,207	1,477,186	0.4481	0.1476	0.2220

C. The 112 Technical Indicators

The 112 input features are generated from 21 indicator families, each computed over five rolling windows (3, 5, 10, 20, and 60 trading days), giving 105 features, together with a Bollinger Z-score at two windows (20 and 60 days) and five cross-window indicators, for 112 features in total. Each family and its formula are listed below; the window suffix (for example, _20d) denotes the rolling window. All features for a stock-day t use information available only up to day t, so no feature incorporates look-ahead information relative to the forward-return targets.

Group	Indicator family	Formula (rolling, window n)	Windows (days)
Price / trend	return / return_mean	n-day rolling mean of daily return (close_t/close_{t-1} - 1)	3, 5, 10, 20, 60
	sma_gap	(close - SMA_n) / SMA_n (moving-average gap)	3, 5, 10, 20, 60
	bollinger_z	(close - SMA_n) / STD_n (Bollinger Z-score)	20, 60
	sma_cross	SMA_a / SMA_b - 1 (fast/slow MA cross)	5/20, 10/60
Momentum	rsi	RSI(n) (Wilder relative strength index)	3, 5, 10, 20, 60
	stoch_k	stochastic %K over n days	3, 5, 10, 20, 60
	williams_r	Williams %R over n days	3, 5, 10, 20, 60
	range_position	position of close within the n-day high-low range	3, 5, 10, 20, 60
Volatility	return_vol	n-day rolling std of daily return	3, 5, 10, 20, 60
	atr	n-day rolling mean of true range	3, 5, 10, 20, 60
	intraday_range_mean	n-day rolling mean of (high - low) / close	3, 5, 10, 20, 60
	volatility_regime	STD_a / STD_b - 1 (short/long volatility ratio)	5/20
Volume / value	volume_gap / value_gap	(volume or value - SMA_n) / SMA_n	3, 5, 10, 20, 60
	turnover_gap	(turnover - SMA_n) / SMA_n	3, 5, 10, 20, 60
	obv_flow / pvt_flow	n-day difference of OBV / PVT	3, 5, 10, 20, 60
	volume_cross	SMA_vol_a / SMA_vol_b - 1	5/20

Group	Indicator family	Formula (rolling, window n)	Windows (days)
Session structure	pre_qty_share_avg / after_qty_share_avg	n-day mean of pre-/after-market volume share	3, 5, 10, 20, 60
	pre_value_share_avg / after_value_share_avg	n-day mean of pre-/after-market value share	3, 5, 10, 20, 60
Short-selling	shortsell_share_avg	n-day mean of short-selling value share	3, 5, 10, 20, 60
	shortsell_pressure	(short-selling - SMA_n) / SMA_n	3, 5, 10, 20, 60
	shortsell_cross	SMA_short_a / SMA_short_b - 1	5/20

D. Reference-Model Hyperparameters

Model	Type	Key hyperparameters	Loss	Optimizer	Epochs / stop
LightGBM	GBDT (CPU)	n_estimators=500, lr=0.05, num_leaves=127, subsample=0.8, colsample=0.8, reg_lambda=1.0	multiclass logloss	—	<=500, early-stop 50
XGBoost	GBDT (CPU)	n_estimators=500, lr=0.05, max_depth=8, subsample=0.8, colsample=0.8, reg_lambda=1.0	softprob	—	<=500, early-stop 50
MLP-2Layer	MLP (GPU)	hidden=[512, 256], dropout=0.2, batch=16384	weighted CE	AdamW (lr=1e-3, wd=1e-4)	20
MLP-3Layer	MLP (GPU)	hidden=[512, 256, 128], dropout=0.2, batch=12288	weighted CE	AdamW (lr=8e-4, wd=1e-4)	20
MLP-ResNet	MLP-ResNet (GPU)	d_hidden=256, n_blocks=4, dropout=0.2, batch=16384	weighted CE	AdamW (lr=1e-3, wd=1e-4)	20
ResNet	ResNet-Tab (GPU)	d_hidden=384, n_blocks=6, dropout=0.15, batch=12288	weighted CE	AdamW (lr=8e-4, wd=1e-4)	20
FT-Transformer	Transformer (GPU)	d_token=64, n_blocks=3, n_heads=4, dropout=0.15, batch=4096	weighted CE	AdamW (lr=7e-4, wd=1e-4)	20
TabNet	TabNet (GPU)	n_d=32, n_a=32, n_steps=5, gamma=1.5, lambda_sparse=1e-4, batch=8192	weighted CE	AdamW (lr=1e-3, wd=1e-4)	20

E. Validation-Set Macro F1 of the Reference Models

Target	MLP-2L	MLP-3L	MLP-Res	ResNet	FT-Tr	TabNet	XGB
1-day ±5%	0.3732	0.3769	0.3700	0.3842	0.4115	0.3661	0.3325
1-day ±10%	0.3273	0.3300	0.3312	0.3549	0.3443	0.3250	0.3384
5-day ±5%	0.4347	0.4370	0.4307	0.4309	0.4161	0.4281	0.3319
5-day ±10%	0.3701	0.3740	0.3723	0.3709	0.4039	0.3636	0.3267
10-day ±5% (main)	0.4557	0.4524	0.4430	0.4458	0.4263	0.4513	0.3433
10-day ±10%	0.3997	0.4042	0.4004	0.4077	0.3863	0.3946	0.3157

F. XGBoost Tree-Baseline Results (Test Set)

Target	Accuracy	Macro precision	Macro recall	Macro F1
1-day ±5%	0.9359	0.5951	0.3407	0.3372
1-day ±10%	0.9860	0.6004	0.3365	0.3373
5-day ±5%	0.7502	0.5481	0.3619	0.3436
5-day ±10%	0.9142	0.5649	0.3388	0.3296
10-day ±5% (main)	0.6360	0.5000	0.3848	0.3607
10-day ±10%	0.8396	0.5312	0.3403	0.3193

Author Biography

Yeo-Hoon Yoon

Ph.D. Student, Graduate School of AI, Seoul School of Integrated Sciences & Technologies (aSSIST)

Research Interests: AI, Time-Series Data Forecasting, Digital Transformation, Generative AI, Data Analysis

E-mail: yh.yoon@stude.assist.ac.kr

Kyung-Sung Kim

Chair Professor, The Institute for Industrial Policy Studies Visiting Professor, Seoul School of Integrated Sciences & Technologies(aSSIST)

Honorary Dean of AI School

Ph.D. in Education, University of California, Los Angeles (UCLA)

Research Interests : AI, Statistical Analysis, Psychometrics

E-mail : kskim3@assist.ac.kr