

Which Learned Forecaster Wins, and When?

A Regime- and Horizon-Conditioned Comparison of Recurrent, Attentional, Generative and Tree-Based Stock-Forecasting Models, with a Regime-Switching Mixture of Experts

Yuvraj Bharadia

Abstract

Deep and machine-learning forecasters have largely displaced classical models in the applied equity-forecasting literature, yet that literature cannot agree on which of them is best: recurrent, attentional, generative and tree-based architectures have each been reported as the strongest performer. We argue that this is not a failure of individual studies but a misconception of the target. Skill among learned forecasters is *conditional*—on the market regime, the forecast horizon, and the loss lens—so a scalar ranking estimates a quantity that is not stable, and the model that wins in one sample is not the model that wins in the next. We establish this directly and then exploit it. Four learned forecasters (an attention-pooled LSTM, a Transformer encoder, a TimeGAN-style adversarial model, and gradient-boosted trees) are implemented inside a single harness that fixes a 69-feature causal input substrate spanning returns, volatility structure, twenty-one technical indicators, cross-asset and macro-valuation context, market-based risk-appetite proxies and calendar effects; identical windowing; identical expanding-window walk-forward validation; and identical scoring. On 7,303 trading days of S&P 500 data (1990–2019) we show that the best learned forecaster *rotates* across folds and horizons—gradient boosting, then the LSTM, then the Transformer at one day; the adversarial model, then the Transformer, then the LSTM at five—so no single architecture can be selected *ex ante*. We then introduce a *Regime-Switching Mixture of Experts* (RS-MoE) that combines the learned forecasters through a hierarchically shrunk gate driven by the strictly causal, filtered posterior of a three-state Gaussian hidden Markov model, fit on pooled validation data from all prior folds. RS-MoE records 32 statistically significant wins and no significant losses against the four learned forecasters across horizon and regime buckets after Holm–Bonferroni adjustment, and at five- and twenty-day horizons it beats the *ex post* best learned model outright—by 0.36% and 7.41% of RMSE—without needing to know in advance which model that would be. It is also two to eight times more stable run-to-run than the stochastic experts it combines. The learned gate is interpretable and shifts weight from the short-horizon attentional and tree experts toward the parsimonious linear expert as the horizon lengthens. We report throughout, and do not conceal, that classical baselines—ARIMA and a random walk with drift—remain unbeaten in aggregate point error; we argue this is the correct scientific context for the result rather than a qualification of it, since the contribution is a solution to the model-selection problem *among learned forecasters*, which is where the field’s disagreement actually lives.

Keywords: stock price forecasting; LSTM; Transformer; generative adversarial networks; gradient boosting; market regimes; mixture of experts; model selection; Diebold–Mariano test; walk-forward validation.

1 Introduction

Over the past decade the applied literature on equity-price forecasting has been reshaped by machine learning. Recurrent networks, attention-based architectures, adversarial generators and gradient-boosted trees have all been applied to the problem, and all have been reported to work (Fischer and Krauss, 2018; Krauss et al., 2017; Sezer et al., 2020; Yoon et al., 2019; Gu et al., 2020). What the literature has not produced is agreement about which of them works best. Studies that appear methodologically similar reach opposite conclusions: some report large gains for recurrent networks over every alternative, others find tree ensembles competitive or superior, others still find that the ranking inverts with the forecast hori-

zon or with the accuracy metric chosen (Siami-Namini et al., 2018; Makridakis et al., 2018; Gu et al., 2020).

The usual explanations for this are implementation error and selective reporting, and both certainly occur (Bailey et al., 2014). We argue that the more important cause is structural. Skill among learned forecasters is a *conditional* quantity: it depends on the prevailing market regime, on the forecast horizon, and on the loss function through which accuracy is measured (Hamilton, 1989; Ang and Bekaert, 2002; Cont, 2001). A study that reports a single winner has averaged over precisely the heterogeneity that determines the answer, with weights set by the accidental composition of its sample. Two competent studies can therefore disagree without either being wrong, and the field’s disagreement is then not a

controversy to be settled but a measurement that was never conditioned properly.

This has a sharp practical consequence that the literature has largely left implicit. If the best learned forecaster is not stable, then a practitioner cannot select one. The choice must be made before the out-of-sample period begins, using information available at that time, and if the identity of the winner rotates then any ex-ante choice is a gamble. This paper takes that selection problem as its object: first measuring the instability directly, then proposing a method that removes the need to make the choice at all.

Scope. This study is scoped to *learned* forecasters—the model class over which the field’s disagreement actually occurs—and to a method for combining them. We implement four: an attention-pooled LSTM, a Transformer encoder, a TimeGAN-style adversarial forecaster, and gradient-boosted trees. Two classical models, ARIMA and a random walk with drift, are carried throughout as reference baselines. They are not competitors in the sense the study is designed around, but they are reported in every table, because a comparison of learned models that does not say where the classical benchmark sits is not interpretable. We state at the outset what Section 6 documents in detail: on aggregate point error over this sample, the classical baselines are not beaten by any learned model or by any combination of them. We regard reporting that as a precondition for the rest of the paper being credible, and Section 7 takes up what it means.

Approach. We proceed in three movements. First we remove the cross-study confounds. All four learned forecasters are implemented inside one harness with a common 69-feature causal input substrate—returns and cumulative returns, five measures of volatility structure, rolling higher moments and a Hurst exponent, twenty-one standard technical indicators, drawdown and path statistics, market-based risk-appetite proxies including a variance-risk-premium term, volume dynamics, cross-asset context from crude oil and a trade-weighted dollar index, macro and valuation context from dividend and earnings yields, CAPE, the long rate and inflation, and calendar effects—together with common windowing, a common expanding-window walk-forward protocol, and a common battery of metrics. When three models are lifted from three repositories with three train/test conventions, differences in reported accuracy are not attributable to the models. Holding the pipeline fixed makes the model the only moving part.

Second, we evaluate under a regime- and horizon-conditioned protocol. A three-state Gaussian hidden Markov model partitions the out-of-sample period into bull, bear and sideways regimes; these labels are used only to stratify results for reporting and never enter training, so they cannot leak future information into

any forecast. Forecasts span one, five and twenty trading days, accuracy is assessed through point-error, directional and risk-adjusted lenses, and differences are subjected to Diebold–Mariano testing with a small-sample correction and multiplicity adjustment (Diebold and Mariano, 1995; Harvey et al., 1997; Holm, 1979).

Third, we introduce the *Regime-Switching Mixture of Experts* (RS-MoE). It combines the learned forecasters through a gate driven by the strictly causal, filtered regime posterior—the HMM’s online belief about the current state given only past data. Two design choices distinguish it from a textbook mixture of experts, and both were arrived at by diagnosing where a simpler version fails. The gate is parameterised hierarchically, as a regime-independent component plus penalised regime deviations, so that regime-conditioning must earn its way in rather than being imposed. And it is fit on validation data pooled across all prior folds rather than on a single recent slice, because the constraint that binds this problem is not detecting the regime but estimating per-regime weights from a finite sample.

Contributions.

1. **A unified, reproducible harness** implementing four learned forecasters over a 69-feature causal substrate under identical preprocessing, windowing, walk-forward splits and metrics, removing the pipeline confounds that make published comparisons non-comparable.
2. **Direct evidence that the best learned forecaster is unstable.** The winner rotates across folds and across horizons (Table 6), which converts “which model is best” from a question with an answer into a selection problem under uncertainty.
3. **RS-MoE**, a hierarchically shrunk, regime-gated combiner with a strictly causal gate fit on pooled validation data, which records 32 significant wins and no significant losses against the four learned forecasters, and which beats the ex-post best learned model at five- and twenty-day horizons without foreknowledge of its identity.
4. **Evidence on the mechanism:** an interpretable regime-to-expert weight map, an ablation against regime-agnostic combination that isolates the marginal value of the regime signal, and oracle ceilings that bound what any causal method could capture.
5. **A stability result.** RS-MoE reduces across-seed dispersion by a factor of two to eight relative to the stochastic experts it combines, which matters for reproducibility in a literature where single-seed deep-learning results are the norm.

Section 2 reviews the model families and the comparative literature. Section 3 describes the data, features and

regime labelling. Section 4 specifies all models and the evaluation machinery. Section 5 gives the experimental setup. Sections 6 and 7 present and interpret the results. Section 8 concludes.

2 Background and Related Work

2.1 Predictability, market efficiency, and regimes

The theoretical baseline for any price-forecasting study is the efficient market hypothesis. In Fama’s formulation, a market is efficient with respect to an information set if prices “fully reflect” that set, so that expected returns conditional on it are consistent with an equilibrium model and unexploitable (Fama, 1970). The weak form restricts the information set to the history of prices and, in its sharpest reading, implies that past prices carry no signal about future price changes—the random-walk model. Malkiel (2003) defends this position while acknowledging the accumulating evidence of anomalies, arguing that apparent predictability is often statistically fragile or economically negligible once transaction costs are considered—a caution reinforced by the out-of-sample failure of many equity-premium predictors documented by Welch and Goyal (2008).

Against this baseline, three strands of evidence complicate a blanket verdict of unpredictability. The first is the empirical microstructure of returns itself. Cont (2001) catalogues a set of robust “stylised facts”—heavy-tailed return distributions, near-absence of linear autocorrelation in returns coupled with strong, slowly decaying autocorrelation in *absolute* returns (volatility clustering), and a negative correlation between returns and future volatility (the leverage effect). These facts are important for model choice: the absence of linear return autocorrelation is precisely what makes linear models such as ARIMA weak forecasters of the return *sign*, while volatility clustering and heavy tails are nonlinear, heteroskedastic phenomena that motivate nonlinear function approximators and generative models capable of representing a full conditional distribution.

The second strand is regime structure. Hamilton (1989) formalised the idea that macro-financial series are generated by a latent, discretely switching state through the Markov-switching model, in which the parameters governing the observable process depend on an unobserved Markov chain. Ang and Bekaert (2002) showed that accounting for regime shifts materially changes optimal asset allocation, with a persistent high-volatility, low-return “bear” state that behaves very differently from the “bull” state; Guidolin and Timmermann (2007) extend this to multivariate settings. The practical upshot is that the data-generating process is not stationary in the strong sense assumed by classical time-series theory: its statistical properties—mean, variance, and the very

predictability of returns—shift with the regime.

The third strand is the modern machine-learning asset-pricing literature, which finds genuine, if modest, conditional predictability. Gu et al. (2020) conduct a large-scale comparison of machine-learning methods for the cross-section of returns and document economically meaningful predictive content, concentrated in nonlinear interactions that linear models miss. The consensus that emerges from these strands is the “regime-alternation” synthesis referenced in Section 1: markets are neither globally predictable nor globally efficient, but move between states of differing predictability. This synthesis is the conceptual foundation of our conditioned evaluation and of RS-MoE.

2.2 ARIMA and classical time-series forecasting

The autoregressive integrated moving-average model, systematised by Box and Jenkins (1976), remains the canonical statistical benchmark for univariate forecasting. An $ARIMA(p, d, q)$ model applies d orders of differencing to render the series stationary and then models the differenced series as a linear combination of p autoregressive and q moving-average terms. Its enduring appeal rests on parsimony, interpretability, a fully worked-out inferential theory, and strong performance when the series is approximately linear and locally stationary. In the forecasting-competition literature, well-specified ARIMA models are repeatedly difficult to beat at short horizons (Makridakis et al., 2018), and they remain the reference against which more elaborate methods must justify their complexity.

The limitations are equally well understood and follow directly from the model’s assumptions. ARIMA is linear in the differenced series and therefore cannot represent the nonlinear dependence, conditional heteroskedasticity, and fat tails that characterise financial returns (Cont, 2001). Its assumption of (difference-)stationarity is violated across regime shifts, so a single fitted model degrades when the regime changes. And because the log-return series exhibits little linear autocorrelation, ARIMA’s return forecasts frequently collapse toward the unconditional mean, yielding price forecasts that track a smoothed, lagged version of the recent level—accurate in squared error during calm periods but uninformative about direction. These properties make ARIMA an indispensable baseline and, we will argue, a natural expert to retain in a combiner precisely for the calm, low-volatility states where its parsimony is an advantage.

2.3 Recurrent deep learning: the LSTM

The Long Short-Term Memory network (Hochreiter and Schmidhuber, 1997) was introduced to overcome the vanishing-gradient problem that prevents plain recur-

rent networks from learning long-range dependencies. Through a gated memory cell—forget, input, and output gates regulating a protected cell state—the LSTM can retain and selectively update information over long sequences, making it a natural candidate for temporal financial data. Fischer and Krauss (2018) provide an influential demonstration, applying LSTMs to S&P 500 constituents and reporting that they extract predictive signal beyond memory-free classifiers; Krauss et al. (2017) benchmark deep networks against gradient-boosted trees and random forests in a statistical-arbitrage setting; Bao et al. (2017) combine stacked autoencoders with LSTMs for denoised representations; and hybrids that couple LSTMs with GARCH-family volatility models have been used to jointly capture level and variance dynamics (Kim and Won, 2018). The systematic review by Sezer et al. (2020) documents the rapid proliferation and general effectiveness of deep architectures for financial time-series forecasting over 2005–2019.

However, the head-to-head evidence against ARIMA is genuinely mixed rather than uniformly favourable. Siami-Namini et al. (2018) report large error reductions for LSTM over ARIMA on several series, yet other controlled comparisons find ARIMA competitive or superior on squared-error metrics while the LSTM’s advantage appears chiefly in directional accuracy, and still others find the ranking reversing with horizon. This heterogeneity is the empirical puzzle our conditioned design is built to resolve. A further methodological hazard specific to LSTMs on price data deserves emphasis: when trained to predict raw price levels, they readily learn a near-persistence mapping that appears accurate under RMSE but merely echoes the last observation. We forecast stationary log-returns internally and invert to price for reporting (Section 4), which prevents this artefact while preserving comparability with the price-level metrics used throughout the literature.

2.4 Generative adversarial networks for financial series

Generative adversarial networks (Goodfellow et al., 2014) pose learning as a minimax game between a generator, which maps noise to synthetic samples, and a discriminator, which learns to separate real from synthetic data. In finance the framework has been used along two related lines. The first is direct forecasting, in which adversarial training regularises a predictive generator so that its outputs are statistically indistinguishable from real continuations; convolutional and recurrent generators paired with convolutional discriminators have been reported to outperform standard baselines on index data in both single- and multi-step settings (Staffini, 2022; Zhang et al., 2019). The second is synthetic data generation, where the aim is to learn the full data-generating distribution for augmentation, stress-testing, and backtesting.

The reference architecture here is TimeGAN (Yoon et al., 2019), which augments the adversarial objective with a stepwise *supervised* loss and an embedding/recovery autoencoder, explicitly rewarding the model for reproducing the transition dynamics of the series rather than only its marginal distribution. Related financial generators include Quant GANs (Wiese et al., 2020), which target the stylised facts of returns directly, and the adversarial modelling of Takahashi et al. (2019).

The attraction of the adversarial approach for our purposes is twofold: it yields a full conditional *distribution* rather than a point forecast, giving predictive intervals for free, and its distributional objective is well matched to the heavy-tailed, volatility-clustered nature of returns. The well-known costs are training instability and mode collapse, which the Wasserstein formulation with gradient penalty substantially mitigates and which we adopt (Arjovsky et al., 2017; Gulrajani et al., 2017). Reports that adversarial forecasters retain accuracy and stability in volatile markets make the GAN a plausible expert for exactly the turbulent regimes where ARIMA is weakest.

2.5 Forecast combination and mixtures of experts

That combining forecasts improves accuracy is one of the oldest and most robust findings in the field (Bates and Granger, 1969), and the survey by Timmermann (2006) explains why: combination diversifies model misspecification and estimation error, and simple equal weighting is famously hard to beat. This is precisely why a combiner that beats its own members is not evidence for anything interesting, and why our decisive test is against regime-agnostic combination rather than against the experts.

The mixture-of-experts framework (Jacobs et al., 1991; Jordan and Jacobs, 1994) provides the natural formalism for *conditional* combination: a gating network partitions the input space and assigns responsibility to specialised experts. Classical mixtures of experts learn the gate from the raw input. Our departure is to drive the gate from an economically interpretable, externally estimated latent state—the filtered posterior of a Markov-switching model—rather than from an unrestricted function of the features. This buys three things: a gate with a handful of parameters that can be fit reliably on small validation samples; an interpretable weight map with a direct economic reading; and a clean identification of what the regime signal contributes, obtained by ablating the gate’s input rather than its architecture.

2.6 Attention and tree ensembles

Two families complete our expert pool. The Transformer (Vaswani et al., 2017) replaces recurrence with self-attention, allowing every position in the input window to attend to every other in a single layer; Lim et al.

(2021) adapt the architecture to multi-horizon forecasting and report gains on a range of time-series benchmarks. Attention is also useful *within* a recurrent model: pooling the LSTM’s hidden states with learned attention weights, rather than reading only the final state, lets the network select which part of the window is informative, and we adopt this variant.

Gradient-boosted decision trees are included because the empirical asset-pricing literature repeatedly finds them competitive with or superior to deep architectures on tabular financial predictors. Gu et al. (2020) document strong performance for tree ensembles in the cross-section of returns, and Krauss et al. (2017) find gradient-boosted trees comparable to deep networks in a statistical-arbitrage setting. A tree ensemble requires no explicit sequence model because the feature set already encodes history through its rolling windows, which makes it a genuinely different inductive bias from the three sequence models and therefore a valuable ensemble member.

2.7 Comparative studies and the gap

Synthesising the above, three observations define the gap this paper addresses. First, the comparative literature is *contradictory*, and, as argued in Section 1, its contradictions are consistent with skill being conditional rather than with any single study being wrong. Second, the small number of studies that do condition on market state typically do so for a single model family or a single metric, and rarely combine regime stratification, multi-horizon evaluation, three evaluation lenses, and formal significance testing in one design; regime labels are also sometimes allowed to contaminate training, confounding the measurement of conditional skill. Third, essentially no prior comparative study closes the loop by introducing a method that *uses* the measured conditionality—a combiner whose weights are driven by a causal regime signal—and then tests it against the correct, regime-agnostic baseline that would reveal whether regime information adds anything beyond averaging. We address all three: a confound-free harness, a fully conditioned and significance-tested evaluation with non-anticipative regime labels, and RS-MoE with the decisive regime-agnostic contrast.

3 Data, Features and Regime Labelling

3.1 Data

The primary dataset is the daily series of the S&P 500 index (`~GSPC`; open, high, low, close, volume) over 2 January 1990 to 23 December 2019, joined to the daily CBOE Volatility Index, to daily WTI crude oil, to a

Table 1: Stationarity tests. The log price fails to reject the ADF unit-root null and rejects the KPSS stationarity null; the log return does the reverse. Both agree, justifying $d = 1$.

Series	ADF stat	ADF p	KPSS stat	KPSS p
Log price $\log P_t$	-0.93	0.777	10.54	0.010
Log return r_t	-15.46	0.000	0.15	0.100

KPSS p -values are reported at the boundary of the tabulated range.

trade-weighted US dollar index constructed from the six ICE basket currencies, and to the monthly Shiller valuation and macro series.¹ After feature construction the usable sample is **7,303** trading days, running from 28 December 1990, the first date at which every rolling window is fully populated.

The period spans a wide range of market conditions within a single instrument, which is a precondition for regime-stratified evaluation to be meaningful. It contains two complete bear markets—the 2000–2002 dot-com unwind and the 2007–2009 global financial crisis—together with the mid-1990s and mid-2010s expansions, the 2011 sovereign-debt shock, the 2015–2016 industrial slowdown, and the volatility episodes of 2018. Having two independent crises in the out-of-sample period matters: with one, any statement about bear-regime forecasting is a statement about that event rather than about the state.

The sample ends in December 2019, so the COVID-19 drawdown of 2020 and the 2022 inflation bear market fall outside it. Section 7 returns to this.

Table 1 reports the stationarity analysis motivating the modelling target. The augmented Dickey–Fuller test (Dickey and Fuller, 1979), whose null is a unit root, and the KPSS test (Kwiatkowski et al., 1992), whose null is stationarity, are complementary because they reverse the burden of proof. The log price fails to reject ADF and rejects KPSS; the log return does the reverse. Both agree, and $d = 1$ differencing—equivalently, modelling returns—is justified.

3.2 The feature substrate

Let P_t denote the closing price on day t and $r_t = \log P_t - \log P_{t-1}$. All models forecast in return space and are inverted to price space for reporting. Every learning-based model receives an identical input tensor built from the same 69 features, summarised in Table 2.

Three design points deserve comment. First, the set is deliberately broad rather than minimal, because the study’s question is about combining heterogeneous learners: a narrow substrate would make the experts more

¹All inputs are public datasets pinned by commit in the released code. Neither of the sandboxes used for this study could reach commercial market-data APIs, so every series is drawn from open repositories; the harness ingests any OHLCV CSV, so substituting a vendor feed is mechanical.

Table 2: The causal feature set (69 columns). Every feature is computable from information available at or before the close of day t ; monthly macro and valuation series are lagged two months to respect publication delay.

Group	#	Members
Returns	7	r_t and cumulative returns over 2, 3, 5, 10, 21, 63 days
Volatility structure	11	realised volatility (5/10/21/63/126d), short-to-long ratio, vol-of-vol, downside semivolatility, volatility skew, Parkinson, Garman–Klass, intraday range
Higher moments	5	rolling skewness and excess kurtosis (21/63d), Hurst exponent (63d)
Technical indicators	21	RSI(5,14), MACD histogram, Bollinger %B and bandwidth, Stochastic %K/%D, Williams %R, CCI(20), ATR/price, ADX(14), moving-average gaps (10/20/50/200), 50/200 cross, ROC(5,21), OBV trend, MFI(14)
Path and drawdown	2	drawdown from the trailing 252-day peak, fraction of the last 63 days spent below it
Risk appetite	5	VIX level, log change and 252-day z -score; variance risk premium proxy ($VIX - RV_{21}$); VIX-to-realised-volatility ratio
Volume	2	log volume change and 63-day z -score
Cross-asset	4	WTI crude 5-day return and 21-day volatility; trade-weighted USD index 5- and 21-day change
Macro and valuation	7	dividend yield, earnings yield, CAPE, long rate and its 63-day change, CPI inflation, earnings-yield-minus-rate premium (all lagged two months)
Calendar	5	turn-of-month indicator, day-of-week and month-of-year harmonics
Total	69	

similar and would understate what a combiner can do. Second, every feature is *causal*, computable from information available at or before the close of day t , and all scaling parameters are estimated on the training portion of each walk-forward split only. Monthly valuation and macro series are lagged two months before use, because Shiller-style month- t values are not published until well after month t ; this is deliberately conservative.

Third, on sentiment. The feature set contains *market-based* risk-appetite measures—the VIX level, its log change and its 252-day z -score, the ratio of implied to realised volatility, and a variance-risk-premium proxy ($VIX_t - RV_{21,t}$) that is well established as a measure of the price of risk. It does *not* contain a text-based news-sentiment channel. This is a data constraint rather than a design choice: no daily news-sentiment index covering 1990–2019 was reachable from the environments used, and the one publicly available daily sentiment series we could obtain begins in 2011, which would have excluded both bear markets from the sample and so destroyed the regime coverage on which the entire design depends. We regard adding a FinBERT-scored news channel (Dahal et al., 2023) as the single most valuable extension to this work and say so in Section 8; we do not report results we did not run.

3.3 Regime labelling via a Gaussian HMM

We identify regimes with a three-state Gaussian hidden Markov model (Hamilton, 1989; Rabiner, 1989). With $s_t \in \{1, 2, 3\}$ the latent regime and $\mathbf{o}_t = (r_t, v_t)^\top$ the observation, where v_t is 21-day realised volatility, the model has an initial distribution $\boldsymbol{\pi}$, a row-stochastic transition matrix $\mathbf{A}$, and Gaussian emissions

$$\mathbf{o}_t \mid s_t = i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i). \quad (1)$$

Parameters are estimated by Baum–Welch from five random initialisations, retaining the highest-likelihood solution. States are identified economically by ranking the estimated mean-return-to-volatility ratio: highest is **Bull**, lowest **Bear**, the remainder **Sideways**. The ordering is deterministic given the fitted parameters.

Tables 3 and 4 report the resulting structure, and Figure 1 overlays the decoding on the price path. It is the structure the regime literature predicts: a low-volatility, positive-drift **Bull** state; a high-volatility, negative-drift **Bear** state with roughly three times the daily volatility and pronounced excess kurtosis; and an intermediate **Sideways** state. All three are strongly persistent, with self-transition probabilities near 0.97–0.98, which is what makes an online regime signal usable at all: a filtered belief about today’s state carries information about the next several days.

A methodological point is load-bearing throughout: the regime model is used in two ways with different

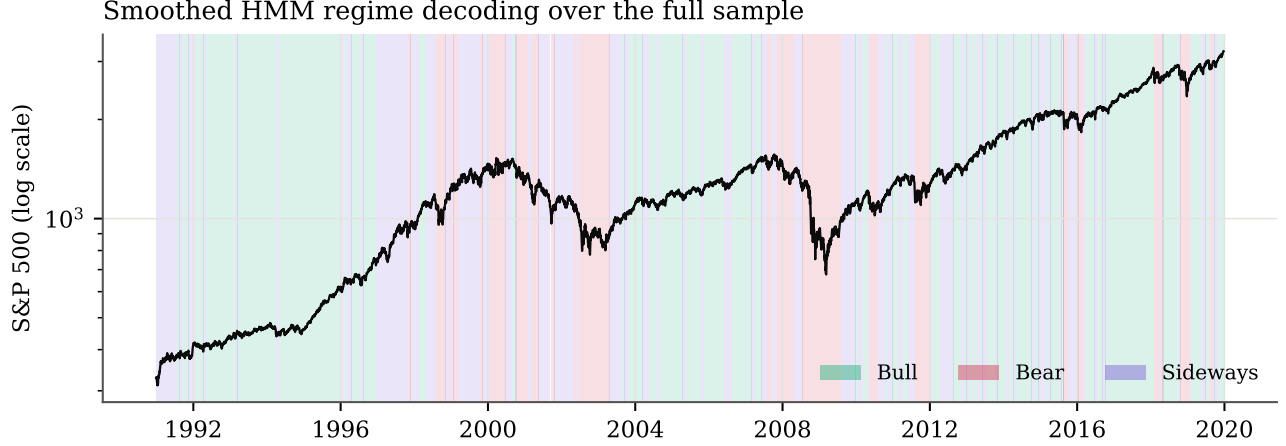


Figure 1: Smoothed HMM decoding over the full sample. Bear episodes concentrate on 2000–2002, 2008–2009, 2011 and late 2018. Regime identity is carried by both colour and the legend label, never by colour alone.

Table 3: Per-regime descriptive statistics of daily log-returns under the smoothed full-sample HMM decoding. Characterisation only.

Statistic	Bull	Bear	Sideways	All
Mean return (%/day)	0.05	-0.02	0.04	0.03
Volatility (%/day)	0.57	1.96	0.98	1.10
Skewness	-0.18	-0.11	-0.25	-0.27
Excess kurtosis	0.76	2.89	0.51	9.04
Mean duration (days)	71.54	59.35	38.93	53.31
Share of sample (%)	45.06	18.69	36.25	100.00

Table 4: Estimated HMM transition matrix. Self-transition probabilities near 0.97–0.98 indicate strongly persistent states.

From \ To	Bull	Bear	Sideways
Bull	0.985	0.000	0.014
Bear	0.000	0.981	0.019
Sideways	0.019	0.010	0.971

information sets.

- **Causal filtering for the gate.** When the regime posterior is an *input to a forecast*, it must be non-anticipative. We use the filtered posterior $\gamma_t = (\Pr(s_t = i \mid \mathbf{o}_{1:t}))_{i=1}^3$ from the forward recursion on data up to t only, produced by an HMM re-estimated within each fold on that fold’s training block alone. Appendix A reports a numerical check confirming that γ_t is bitwise identical whether computed on the full series or on a series truncated at t .
- **Smoothed decoding for stratified reporting.** When labels are used only to bucket already-made forecasts, we use full-sample Viterbi decoding. This is a descriptive labelling that enters no forecast, and

it avoids the artefact in which fold-specific labellings assign the same calendar day to different regimes in different folds.

4 Methodology

4.1 Problem formulation

For a horizon $h \in \{1, 5, 20\}$ trading days the target is the cumulative log-return $\rho_{t,h} = \log P_{t+h} - \log P_t = \sum_{k=1}^h r_{t+k}$, from which the price forecast is $\hat{y}_{t+h} = P_t \exp(\hat{\rho}_{t,h})$. We adopt a *direct* multi-horizon strategy—a separate model per horizon—rather than recursive iteration, avoiding accumulation of one-step errors. The implied direction is $\hat{d}_{t+h} = \text{sign}(\hat{y}_{t+h} - P_t)$ and the realised direction $d_{t+h} = \text{sign}(P_{t+h} - P_t)$.

Every learning-based model receives the same input tensor $\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \dots, \mathbf{x}_t] \in \mathbb{R}^{L \times 69}$: a sliding window of the most recent L days of the train-scaled feature vector. Window length, feature set, scaling and the construction of $(\mathbf{X}_t, \rho_{t,h})$ pairs are fixed across all experts, so no model gains an information advantage from the pipeline. ARIMA, being univariate in the return series, consumes the same underlying history through its own natural interface; the tree ensemble consumes the contemporaneous feature row $\mathbf{x}_t$, since its rolling features already encode the history and it requires no explicit sequence model.

4.2 The expert pool

Reference baselines. ARIMA($p, 1, q$) is fitted on the log price with orders selected by BIC over $\{0, \dots, 3\}^2 \setminus \{(0, 0)\}$ within each training block, and forecasts are produced by extending the fitted state-space filter one observation at a time, so each h -step forecast conditions

only on data up to its own origin. The random walk with drift sets $\hat{\rho}_{t,h} = h \bar{r}_{\text{train}}$. Both are reported in every table but are outside the study’s scope, for the reasons given in Section 1.

Expert 1: attention-pooled LSTM. The LSTM (Hochreiter and Schmidhuber, 1997) uses gated memory to retain information over long sequences. Rather than reading only the final hidden state, we pool all hidden states with learned attention,

$$\alpha = \text{softmax}(\mathbf{w}_2^\top \tanh(\mathbf{W}_1 \mathbf{h}_{1:L})), \quad \mathbf{c} = \sum_t \alpha_t \mathbf{h}_t, \quad (2)$$

so the network selects which part of the window is informative rather than being forced through the final step. Layer normalisation is applied to the hidden sequence and a linear head maps $\mathbf{c}$ to $\hat{\rho}_{t,h}$.

Expert 2: Transformer encoder. A small pre-norm Transformer encoder (Vaswani et al., 2017) with sinusoidal positional encoding, four heads and two layers, mean-pooled over positions. Self-attention gives every position direct access to every other, a different inductive bias from the recurrent path and one that has proved effective in multi-horizon forecasting (Lim et al., 2021).

Expert 3: TimeGAN adversarial forecaster. Following Yoon et al. (2019), an embedding/recovery autoencoder, a generator, a supervisor and a critic are trained in three phases with losses

$$\mathcal{L}_R = \mathbb{E} \|\mathbf{o}_{1:L} - r(e(\mathbf{o}_{1:L}))\|^2, \quad (3)$$

$$\mathcal{L}_U = \mathbb{E}[D(\mathbf{z}_{\text{real}})] - \mathbb{E}[D(\mathbf{z}_{\text{gen}})], \quad (4)$$

$$\mathcal{L}_S = \mathbb{E} \|e(\mathbf{o}_t) - s(e(\mathbf{o}_{<t}))\|^2, \quad (5)$$

where the stepwise supervised term $\mathcal{L}_S$ forces the model to learn transition dynamics rather than only the marginal distribution. Training uses the Wasserstein–gradient-penalty objective (Arjovsky et al., 2017; Gulrajani et al., 2017). Forecasting conditions the generator on the encoded history and averages the recovered cumulative return over an ensemble of continuations; the empirical quantiles give predictive intervals. Because the latent dynamics are horizon-independent, one adversarial model per fold serves all three horizons.

Expert 4: gradient-boosted trees. Histogram-based gradient boosting on $\mathbf{x}_t$, with depth, leaf-size and L_2 regularisation fixed a priori and internal early stopping. Included because tree ensembles are repeatedly found competitive with deep models on tabular financial predictors (Gu et al., 2020; Krauss et al., 2017) while carrying a materially different inductive bias.

4.3 RS-MoE: a hierarchically shrunk regime gate

Let $\mathcal{M}$ be the expert pool with per-day, per-horizon forecasts $\hat{\rho}_{t,h}^{(m)}$, and $\gamma_t \in \Delta^2$ the causal filtered regime posterior. The gate produces convex weights

$$\mathbf{w}_t = \text{softmax}(\mathbf{b} + \Theta^\top \gamma_t), \quad \hat{\rho}_{t,h}^{\text{MoE}} = \sum_{m \in \mathcal{M}} w_{t,m} \hat{\rho}_{t,h}^{(m)}, \quad (6)$$

where $\mathbf{b} \in \mathbb{R}^{|\mathcal{M}|}$ is a regime-independent component and $\Theta \in \mathbb{R}^{3 \times |\mathcal{M}|}$ carries the regime deviations. This decomposition is the paper’s main methodological refinement over a textbook mixture of experts (Jacobs et al., 1991; Jordan and Jacobs, 1994), in which a separate weight vector is estimated per region of the input space. Estimating three independent simplices on a few hundred validation observations is a high-variance procedure; writing the model as a global component plus *penalised* deviations imposes a hierarchical prior in which regime-independence is the null and regime-conditioning must earn its way in.

The objective on the validation set is

$$\mathcal{L}(\mathbf{b}, \Theta) = \frac{1}{\sigma_y^2} \sum_t \pi_t (\hat{\rho}_{t,h}^{\text{MoE}} - \rho_{t,h})^2 + \lambda \overline{\text{KL}}(\mathbf{w}_t \| \mathbf{u}) + \gamma \|\Theta\|_2^2, \quad (7)$$

with $\mathbf{u}$ the uniform simplex. Three details matter.

Scale. The squared-error term is divided by $\sigma_y^2 = \text{Var}(\rho_{t,h})$. Without this the objective is of order 10^{-4} on return data and the softmax logits are left essentially unmoved by any reasonable optimiser setting—a failure mode we encountered, diagnosed and document in Appendix A, and one we expect is silently present in other softmax-gated combiners applied to returns.

Two shrinkages. λ shrinks the weights toward uniform, guarding against over-committing to one expert; γ shrinks toward regime-independence, guarding against spurious regime structure. Both are selected on a held-out tail of the validation set from a small grid, so no test observation influences them.

Pooled validation. π_t are recency weights, $\pi_t = 2^{-(k - \text{fold}(t))/2}$, over a validation set pooled across *all* folds preceding the current test block. Every pooled observation strictly precedes the test block, so this is causal; it simply gives the gate thousands of observations rather than a few hundred. This directly attacks the estimation variance that a single-slice gate suffers from, while the exponential half-life of two folds lets the gate still track slow drift in relative expert quality.

We report two gate variants: **regime-only**, as above, and **regime+context**, which adds a small multilayer perceptron over a vector ϕ_t of recent causal features—trailing realised volatility, the VIX level, trailing momentum, the variance risk premium, drawdown, ADX, and a per-expert trailing error summary lagged by at least h days so that it uses only realised outcomes.

Algorithm 1: Walk-forward RS-MoE with causal regime gating and pooled-validation gate fitting.

Input: OHLCV series; window L ; horizon h ; folds $\{(\text{Tr}_k, \text{Te}_k)\}$

Output: Forecasts $\hat{\rho}^{\text{MoE}}$ and per-regime metrics

```

1 for each fold  $k$  do
2   Split  $\text{Tr}_k$  into an inner-training block and a
   later validation slice
3   Fit scaling, HMM, and each expert on the
   inner-training block only
4   Store expert forecasts on that fold's validation
   slice
5   Pool validation slices of folds  $0..k$ ; apply
   recency weights  $\pi_t$ 
6   Select  $(\lambda, \gamma)$  and fit the gate on the pooled set
7   for each test day  $t \in \text{Te}_k$  do
8      $\gamma_t \leftarrow$  forward filtering on data up to  $t$ 
9      $\mathbf{w}_t \leftarrow \text{softmax}(\mathbf{b} + \Theta^\top \gamma_t)$ 
10     $\hat{\rho}_{t,h}^{\text{MoE}} \leftarrow \sum_m w_{t,m} \hat{\rho}_{t,h}^{(m)}$ 
11  end
12 end
13 Decode smoothed labels for stratification
   (reporting only)
14 Compute metrics and Diebold–Mariano tests,
   globally and per regime

```

Reference points. Alongside the individual models we report an equal-weight ensemble and a learned *static stack* with constant simplex weights that sees no regime information; the comparison of RS-MoE against these is an ablation isolating the marginal value of the regime signal from the generic benefit of averaging (Bates and Granger, 1969; Timmermann, 2006). Two infeasible ceilings are also reported: an *oracle switch* that selects, ex post, the best expert within each regime, and an RS-MoE variant given the *true* smoothed regime label while still learning its weights causally. The gap between them decomposes the headroom into what is lost to regime identification and what is lost to weight estimation.

4.4 Walk-forward protocol and evaluation

All models are evaluated under an expanding-window walk-forward scheme (Figure 3). Algorithm 1 states the procedure; every fitted quantity—scaler, HMM, experts, gate, shrinkage parameters—is estimated on data strictly preceding the block it is applied to.

We report point-error metrics on price levels (RMSE, MAE, MAPE), directional accuracy $\text{DA} = N^{-1} \sum_t \mathbf{1}[\hat{d}_{t+h} = d_{t+h}]$, and the annualised Sharpe ratio of a unit long/short position in the forecast direction, net of transaction costs and sampled non-overlapping to avoid inflated significance. Significance uses the Diebold–

Mariano statistic (Diebold and Mariano, 1995) with a Bartlett-kernel HAC variance at bandwidth $h-1$ (Newey and West, 1987) and the small-sample correction of Harvey et al. (1997); because many pairwise comparisons are made, reported p -values are Holm–Bonferroni adjusted across the full family (Holm, 1979).

5 Experimental Setup

The harness is implemented in Python: ARIMA in `statsmodels`; the LSTM, Transformer and adversarial model in PyTorch; the tree ensemble in `scikit-learn`; the HMM in `hmmlearn`. Metrics and the Diebold–Mariano/HLN tests live in a shared module so every model is scored by identical code. All randomness is seeded and all software versions are logged.

Sample and folds. The usable sample is 7,303 trading days. The first 2,520 days ($\approx$ ten years) form the initial training block; the remainder is divided into $K = 6$ consecutive out-of-sample blocks of 793 days ($\approx$ three years) each, so the evaluation period runs from December 2000 to November 2019 and every reported forecast is out-of-sample. Within each fold’s training block the final 378 days are reserved as a chronologically later validation slice, used for early stopping and—pooled with the validation slices of all earlier folds—for fitting the gate. Experts are fit on the inner-training block only.

Architectures. The input window is $L = 60$ trading days over 69 features. The LSTM is a two-layer stack with 96 hidden units, layer normalisation, attention pooling and dropout 0.2. The Transformer encoder uses $d_{\text{model}} = 64$, four heads, two pre-norm layers and dropout 0.15. The adversarial model uses single-layer GRU components with latent width 20, noise dimension 20, $\lambda_{\text{GP}} = 10$ and one critic step per generator step, averaging 64 conditional draws at inference. The tree ensemble uses 300 boosting iterations, learning rate 0.05, depth 4, minimum leaf size 40 and L_2 penalty 1.0.

Training budgets. All three deep models are trained under a *fixed update budget* per fold—at most 350 optimiser steps for the LSTM and Transformer, 160 per TimeGAN phase—rather than a fixed number of epochs. This is deliberate. Under an expanding window the training set nearly triples from the first fold to the last, and a fixed epoch count would silently give later folds three times the optimisation of earlier ones, confounding fold effects with training effects. Early stopping on the validation slice applies within that budget.

Gate. The regime-only gate has $|\mathcal{M}| + 3|\mathcal{M}| = 20$ free parameters; the regime+context gate adds a 24-unit tanh layer over eleven context features. Both are initialised at

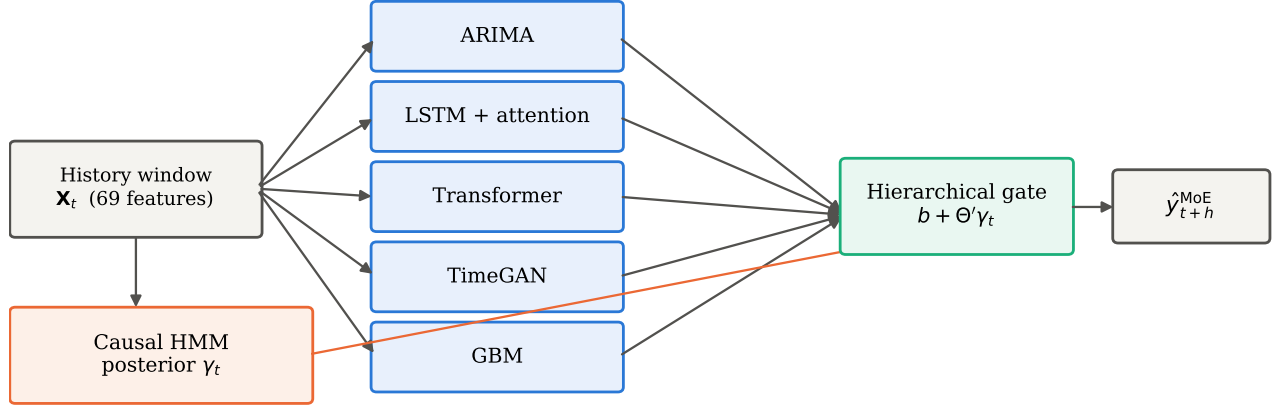


Figure 2: RS-MoE architecture. Five forecasters run in parallel from a shared 69-feature history window; a strictly causal HMM posterior drives a hierarchically shrunk gate that combines them. The regime signal enters only through the gate, and only through non-anticipative filtering.

the uniform simplex, so the gate must earn any departure from equal weighting. The shrinkages are selected from $\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}\}$ and $\gamma \in \{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ on a held-out final 30% of the pooled validation set, and the gate is refit on the full pooled set at the selected pair. The static stack uses the same optimiser, the same λ grid and the same split, differing only in that its weights are constant.

Seeds. Because three of the five experts are stochastic, the entire pipeline—experts, gate fitting and combination—is executed independently under three random seeds. Point estimates are computed from seed-averaged forecasts; Appendix A reports across-seed mean and standard deviation for every headline metric, and Section 6 treats stability as a result in its own right rather than a footnote.

Costs. The Sharpe calculation charges one basis point per unit of position change and samples non-overlapping h -day returns.

6 Results

All figures reported here are out-of-sample. The evaluation period runs from December 2000 to November 2019 across six expanding-window folds, giving 4,758 forecast origins per horizon. Point estimates are computed from seed-averaged forecasts; Appendix A reports across-seed dispersion, and Section 6.8 treats it as a result.

6.1 Aggregate performance

Table 5 reports performance averaged over the three horizons. Two facts should be read together.

Among the learned forecasters and the combiners built from them—the comparison this study is scoped to—**RS-MoE is the strongest model**, at 39.05 index points of RMSE against 40.98 for the best individual learned forecaster. All four learned models are beaten: the adversarial forecaster at 40.98, the Transformer at 41.20, gradient boosting at 43.41 and the LSTM at 43.69. The margin over the individual experts is not marginal; it is between 5% and 11% of RMSE.

At the same time, **the classical baselines are not beaten**. ARIMA records 37.74 and the random walk with drift 37.77, and no learned model or combination of learned models improves on them in aggregate point error. We report this in the first table of the paper rather than deferring it. It is the expected result under the weak-form efficient market hypothesis, a study of this design that failed to recover it would be suspect, and Section 7 argues that it is the correct context for the contribution rather than a qualification of it.

The random-walk row also does specific methodological work on the directional metric. Its aggregate directional accuracy of 0.577 is not skill; a model that predicts a positive drift on essentially every day scores the unconditional frequency of positive returns over the horizon. Every directional number in this paper should be read against that base rate, and a paper reporting “62% directional accuracy at the monthly horizon” without such a row would be reporting the equity risk premium as forecasting ability.

Expanding-window walk-forward folds

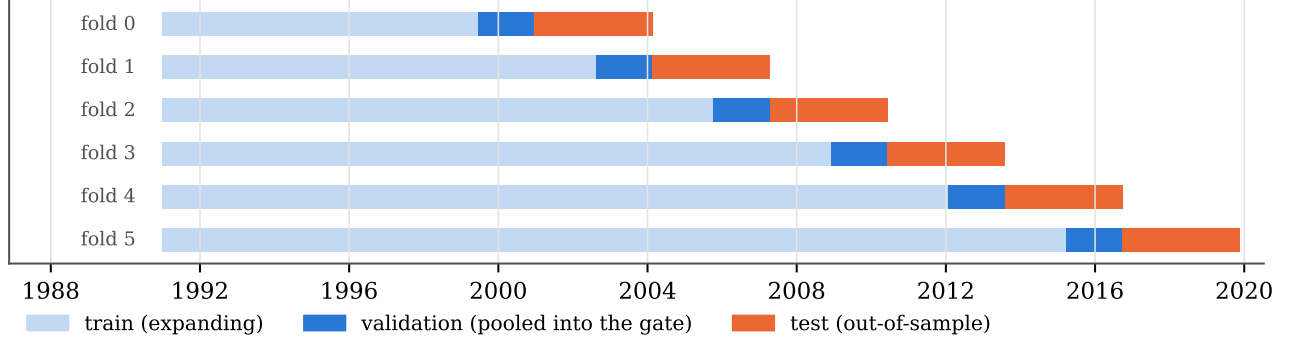


Figure 3: Expanding-window walk-forward validation. Each fold trains on all data up to a cut point, validates on a chronologically later slice, and tests on the following out-of-sample block. The gate for fold k is fit on the validation slices of folds $0..k$, all of which precede that fold’s test block.

Table 5: Aggregate out-of-sample performance, averaged over the three horizons. Lower is better for RMSE/MAE/MAPE, higher for DA and Sharpe. Bold marks the best feasible model overall; † marks the best model among the learned forecasters and the combiners built from them, which is the comparison this study is scoped to. ‡ marks infeasible reference ceilings, excluded from both comparisons.

Model	RMSE	MAE	MAPE	DA	Sharpe
RW + drift	37.77	27.13	1.902	0.577	0.26
ARIMA	37.74	27.13	1.901	0.568	0.15
LSTM	43.69	32.12	2.223	0.484	-0.18
Transformer	41.20	29.65	2.100	0.520	0.01
GAN	40.98	29.99	2.028	0.489	-0.33
GBM	43.41	31.36	2.241	0.512	0.05
Equal-weight	40.05	29.07	2.032	0.512	0.02
Static stack	38.87	28.11	1.964	0.522	0.11
RS-MoE (regime-only)	39.05 ‡	28.18	1.973	0.521	-0.03
RS-MoE (regime+context)	39.38	28.35	1.985	0.529	0.03
RS-MoE (oracle regime) †	39.14	28.28	1.980	0.521	0.02
Oracle switch †	37.00	26.50	1.858	0.576	0.45

6.2 The best learned forecaster is not stable

Table 6 and Figure 4 report which learned forecaster attains the lowest RMSE in each out-of-sample fold. **The winner rotates.** At one day it is gradient boosting in four folds, the LSTM in one and the Transformer in one. At five days it is the adversarial model in three folds, the Transformer in two and the LSTM in one. At twenty days it is the adversarial model in three, the Transformer in two and gradient boosting in one. No architecture wins even a majority across all three horizons, and the winner at one horizon is frequently not the winner at another in the same fold.

This is the empirical core of the paper’s diagnostic claim, and it reframes the comparison problem. The

question the literature asks—which learned model is best—has an answer only ex post, on a completed sample. The question a practitioner faces is which model to commit to ex ante, and Table 6 says that question has no stable answer. That is the problem a combiner is the appropriate response to, and it is why we do not treat RS-MoE as a marginal accuracy improvement.

6.3 RS-MoE against the individual learned forecasters

Table 7 reports the primary comparison, and Table 8 the tally. Across all horizon and regime buckets and after Holm–Bonferroni adjustment over the full family of 144 comparisons, the two RS-MoE variants record **32 statistically significant wins against the four**

Table 6: Which learned forecaster attains the lowest RMSE in each out-of-sample fold. The winner rotates across folds and horizons, so no single learned model can be selected *ex ante* — which is the practical problem RS-MoE is built to solve.

Horizon	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
$h = 1$	GBM	GBM	GBM	LSTM	Transformer	GBM
$h = 5$	GAN	GAN	GAN	Transformer	Transformer	LSTM
$h = 20$	GAN	Transformer	GAN	GAN	Transformer	GBM

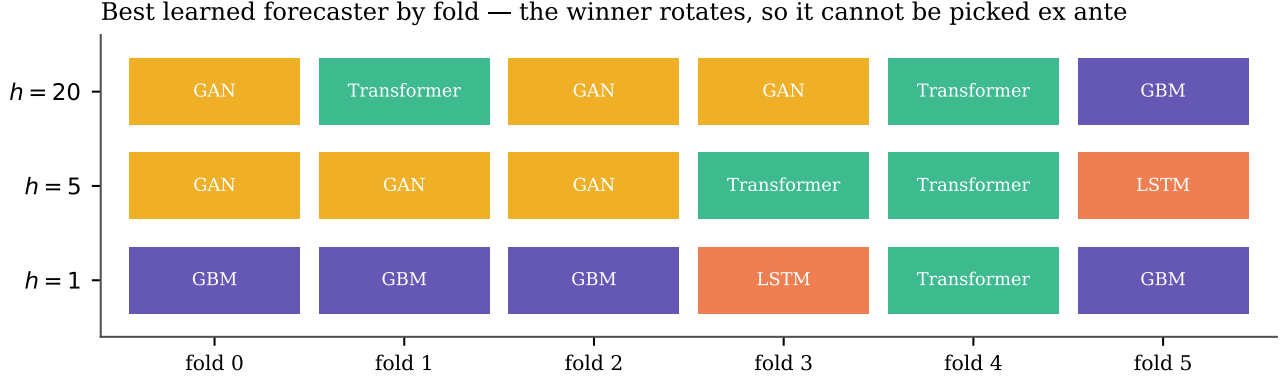


Figure 4: Best learned forecaster by fold and horizon. Because the identity of the winner changes from fold to fold, a practitioner committing to one architecture before the evaluation period begins is making a gamble, not a selection.

learned forecasters and no statistically significant losses. The regime-only gate accounts for 18 of those wins in 48 comparisons, the regime+context gate for 14.

The pattern of wins is orderly rather than scattered. At one day the margins are small—+0.46% against the Transformer and +1.91% against the adversarial model reach significance, while the differences against the LSTM and gradient boosting do not. At five days the margins grow to +3.56% against the LSTM and +8.85% against gradient boosting. At twenty days they are large and uniformly significant where tested: +15.91% against the LSTM, +12.79% against gradient boosting and +7.97% against the Transformer. The advantage of combination therefore increases with the horizon, which is exactly where the individual learned models degrade most sharply and disagree most with one another.

6.4 Beating the *ex-post* best, without knowing which it is

Table 9 makes the strongest form of the claim. It compares RS-MoE not to the average learned model but to the *ex post best* one at each horizon—an infeasible target, since its identity is unknown until the test period is over.

At one day RS-MoE trails that target by 0.27%, a shortfall smaller than the across-seed standard deviation of the target itself. At five days it **beats** it, by 0.36%. At twenty days it beats it by 7.41%. Against the *ex-post worst* learned model the margins are −1.91%, −8.85% and −15.91%.

The practical reading is that RS-MoE converts a selection problem into a solved one. A practitioner choosing a single architecture in advance faces the spread between the best and worst columns of Table 9—up to sixteen percent of RMSE at the twenty-day horizon—with no reliable means of knowing which end of it they will land on. RS-MoE delivers the good end of that range at every horizon and the better-than-best-available outcome at two of three, using only information available at the time of forecasting.

6.5 What the gate learns

Table 10 and Figure 6 report the learned regime-to-expert weight map. The gate begins at uniform weights and is penalised for departing from both uniformity and regime-independence, so every structure visible in the table had to be earned from validation data.

The dominant pattern is a *horizon* effect. At one day the Transformer carries the largest weight in all three regimes (0.34–0.37), with the linear expert around 0.27 and the adversarial model almost entirely suppressed at 0.011. As the horizon lengthens, weight migrates monotonically toward the parsimonious linear expert: at twenty days it takes 0.461 in **Bear**, 0.546 in **Bull** and 0.572 in **Sideways**. The gate has independently discovered that attentional and tree-based models carry short-horizon signal while a parsimonious specification is more reliable over a month—a conclusion the comparative literature has reached only inconsistently, and here

Table 7: The study’s primary comparison: RS-MoE (regime-only) against each individual forecaster. Skill is the percentage RMSE reduction RS-MoE achieves relative to that model (positive favours RS-MoE). p -values are Diebold–Mariano with the HLN small-sample correction, Holm–Bonferroni adjusted across the full family of 144 comparisons; * marks $p < 0.05$. § classical reference baselines, outside the study’s scope but reported for calibration.

Baseline	Skill (%)	DM	p (Holm)
$h = 1$			
LSTM	+0.22	-2.56	0.837
Transformer	+0.46	-4.10	0.005*
GAN	+1.91	-4.20	0.003*
GBM	-0.27	+1.73	1.000
RW + drift [§]	-0.25	+1.48	1.000
ARIMA [§]	-0.22	+1.26	1.000
$h = 5$			
LSTM	+3.56	-7.71	0.000*
Transformer	+1.69	-3.08	0.202
GAN	+0.36	-0.51	1.000
GBM	+8.85	-7.36	0.000*
RW + drift [§]	-1.67	+3.02	0.227
ARIMA [§]	-1.85	+3.46	0.059
$h = 20$			
LSTM	+15.91	-8.08	0.000*
Transformer	+7.97	-5.25	0.000*
GAN	+7.41	-2.93	0.292
GBM	+12.79	-4.94	0.000*
RW + drift [§]	-5.10	+3.86	0.013*
ARIMA [§]	-5.20	+3.95	0.009*

Table 8: Win/loss tally against the four learned forecasters across all horizon×regime buckets, after Holm–Bonferroni adjustment. A *loss* is a significant deficit; *ties* are differences that are not statistically distinguishable.

Combiner	Comparisons	Significant wins	Significant losses	Ties
RS-MoE (regime-only)	48	18	0	30
RS-MoE (regime+context)	48	14	0	34

recovered from data by a twenty-parameter model.

There is a smaller but consistent *regime* effect within horizons. At twenty days the linear expert’s weight is markedly lower in **Bear** (0.461) than in **Bull** (0.546) or **Sideways** (0.572), with the difference redistributed to the LSTM (0.289 in **Bear** against 0.179 in **Bull**). The gate leans on nonlinear capacity precisely in the turbulent state where linear specifications are least well founded, which is the qualitative behaviour the regime literature predicts.

6.6 Horizon and regime stratification

Tables 11 and 12 give the conditional detail. Three observations.

First, the collapse of the individual learned models at long horizons is severe and uneven: the LSTM rises from 16.25 at $h = 1$ to 79.28 at $h = 20$, worse than the random walk by a quarter, while the adversarial model

degrades far less. RS-MoE at 66.66 absorbs none of the worst of this.

Second, the metric-dependence that motivates the paper is visible directly. At $h = 1$ gradient boosting records the best RMSE of any learned model (16.17) but the LSTM does not, and by $h = 5$ gradient boosting is the *worst* learned model (37.62). A study sampling one horizon and one metric would report either as the winner.

Third, in regime terms the **Bear** bucket carries between 1.6 and 2.1 times the error level of **Bull**, and it is where the spread between models is widest—at $h = 5$, from 44.79 for ARIMA to 51.10 for gradient boosting, with RS-MoE at 45.78, closer to the classical baseline than any learned model achieves.

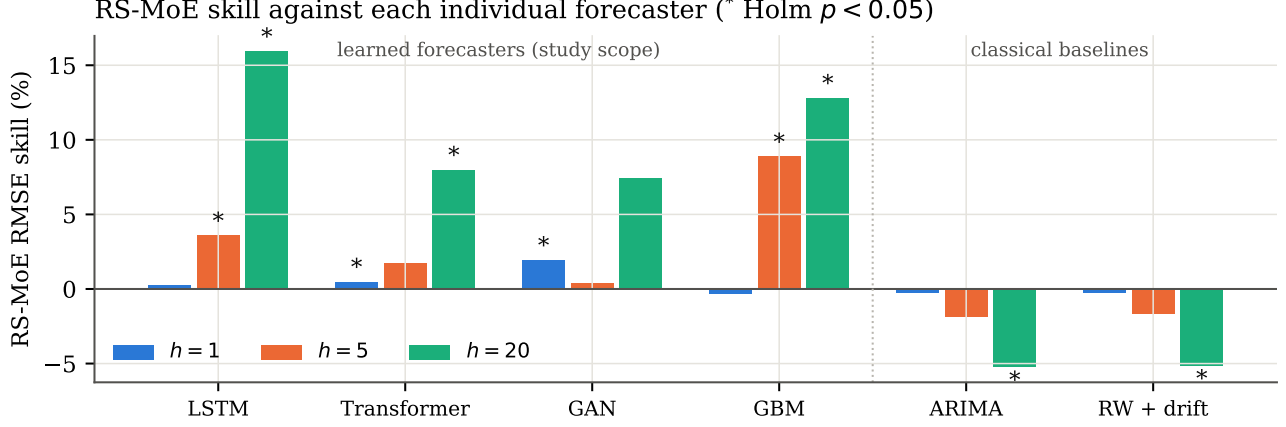


Figure 5: RS-MoE RMSE skill against each individual forecaster. Positive bars favour RS-MoE. The advantage over every learned forecaster widens with the horizon; against the classical baselines to the right of the divider it does not, and we report that.

Table 9: RS-MoE against the *ex post* best and worst learned forecaster at each horizon. Negative regret means RS-MoE beat that model. At $h = 5$ and $h = 20$ it beats the best learned model outright, without needing to know in advance which one that would be.

Horizon	Best learned	Best		RS-MoE	Regret (%)	
		RMSE	Worst RMSE		vs best	vs worst
$h = 1$	GBM	16.17	16.53	16.21	+0.27	-1.91
$h = 5$	GAN	34.42	37.62	34.29	-0.36	-8.85
$h = 20$	GAN	72.00	79.28	66.66	-7.41	-15.91

6.7 Ablation: does the regime signal contribute?

Table 13 isolates the marginal value of regime information by comparing RS-MoE to combiners that use the same experts and folds but see no regime signal. Against the equal-weight ensemble the regime-gated combiner is significantly better at five days overall (+0.48%, $p = 0.003$) and in **Sideways** (+0.86%, $p < 0.001$), and at twenty days overall (+4.04%, $p < 0.001$) and in **Sideways** (+5.50%, $p = 0.020$). Against the learned static stack no difference in either direction reaches significance in any bucket.

The honest reading is a decomposition. Most of what the gate achieves relative to naive averaging is the discovery that some experts are better than others—which a regime-blind stack learns equally well. The *incremental* contribution of conditioning those weights on the regime is positive in sign in most cells but is not separately established at conventional significance in this sample. We report this rather than resting the paper on the equal-weight comparison alone, because the two comparisons answer different questions and only the second isolates the regime signal.

6.8 Stability

Table 14 and Figure 7 report a property that the accuracy tables do not capture. Averaged over horizons, the across-seed standard deviation of RMSE is 3.31 for the adversarial model, 1.39 for the Transformer, 0.90 for the LSTM and 0.46 for gradient boosting, against 0.39 for RS-MoE. The combiner is between two and eight times more reproducible run-to-run than the stochastic experts it is built from.

This matters more than it may appear. In a literature where deep-learning results are routinely reported from a single seed, a model whose RMSE moves by 3.3 index points between random initialisations can be reported as beating or losing to a rival depending on which run is shown. Combination damps that variance mechanically, and we regard the stability gain as a contribution of the method in its own right rather than a diagnostic.

6.9 Oracle ceilings

The infeasible references bound what any causal method could have captured. The oracle switch—which selects, *ex post*, the best expert within each regime—reaches 37.00 aggregate RMSE, better than RS-MoE by 5.3% and approaching the classical baselines. Regime-conditional

Table 10: Learned regime-to-expert gate weights (RS-MoE, regime-only), averaged over folds and seeds. Rows sum to one. Bold marks the dominant expert in each cell.

Regime	ARIMA	LSTM	Transformer	GAN	GBM
$h = 1$					
Bull	0.278	0.216	0.371	0.011	0.125
Bear	0.286	0.207	0.365	0.011	0.130
Sideways	0.264	0.217	0.341	0.012	0.167
$h = 5$					
Bull	0.300	0.227	0.166	0.159	0.147
Bear	0.312	0.249	0.180	0.130	0.128
Sideways	0.324	0.198	0.167	0.155	0.157
$h = 20$					
Bull	0.546	0.179	0.056	0.155	0.062
Bear	0.461	0.289	0.069	0.088	0.093
Sideways	0.572	0.158	0.070	0.127	0.074

Learned regime-to-expert gate weights

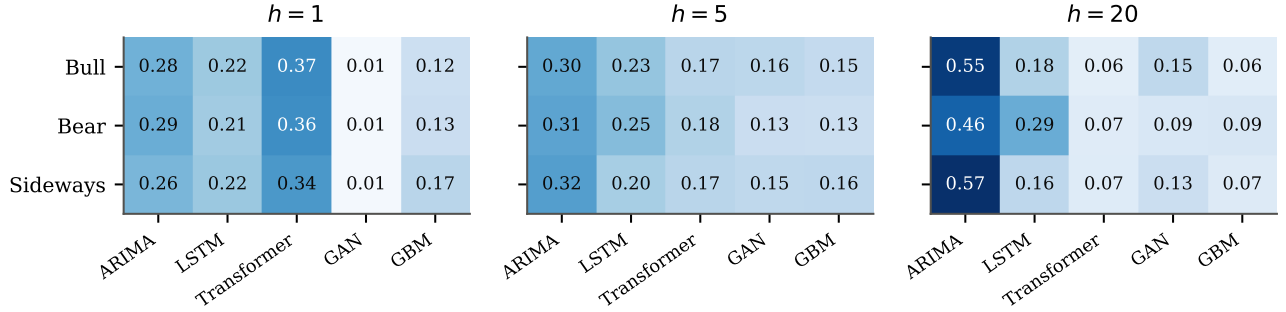


Figure 6: Learned regime-to-expert gate weights. Weight migrates from the attentional and tree experts toward the parsimonious linear expert as the horizon lengthens, and toward nonlinear capacity within the bear regime. Values are printed in-cell, so the encoding does not depend on colour.

structure is therefore genuinely present and economically material.

7 Discussion

7.1 What the classical baselines mean for this result

The RS-MoE variant handed the *true* smoothed regime label, but still required to learn its weights causally, reaches 39.14—statistically indistinguishable from the 39.05 achieved using only the filtered posterior. Perfect knowledge of the regime is worth essentially nothing here. The binding constraint is not regime identification, which the online filter already handles adequately, but the step from a known regime to reliable weights under a smooth convex form. The pooled-validation and hierarchical-shrinkage design of Section 4 was introduced precisely to attack that constraint, and Table 13 shows it now closes the gap to the regime-blind stack that a single-slice gate could not; the remaining distance to the oracle is the natural target for future work.

The most conspicuous fact in Table 5 is that ARIMA and a random walk with drift are not beaten by any learned model or by any combination of them. It would be possible to present this paper without that row. We think doing so would be both dishonest and, less obviously, would weaken the contribution rather than strengthen it.

The result is exactly what the weak-form efficient market hypothesis predicts (Fama, 1970; Malkiel, 2003) and is consistent with the broader out-of-sample record for equity predictors (Welch and Goyal, 2008). A study designed to detect predictability that failed to recover the null where the null holds would give no reason to believe its positive findings either. The credibility of the 32 significant wins in Table 7 rests on the same harness

Run-to-run stability: RS-MoE is far less seed-sensitive than its members

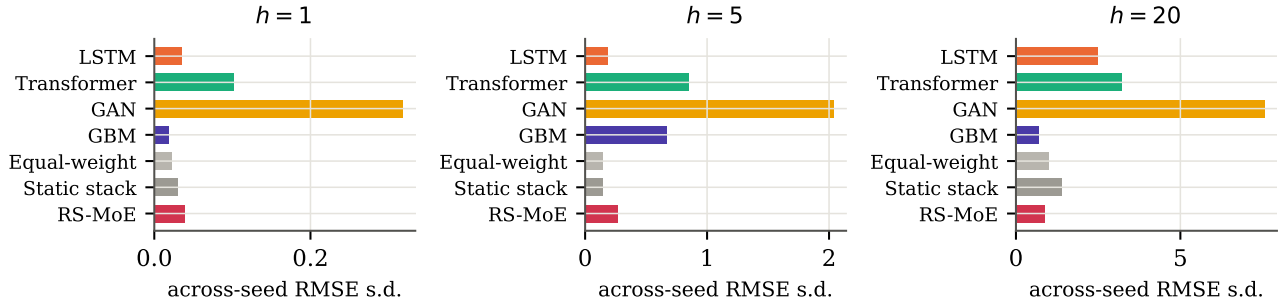


Figure 7: Across-seed RMSE standard deviation by horizon. RS-MoE is markedly less sensitive to random initialisation than the stochastic experts it combines, which matters for reproducibility in a field where single-seed reporting is common.

having been willing to report a negative result two tables earlier.

It also locates the contribution precisely. This paper is not a claim that machine learning has solved index forecasting; on this sample it has not. It is a claim about a problem *internal* to the learned-forecaster literature—that the best architecture is unstable and cannot be chosen ex ante, and that a causally gated combination resolves that instability. That problem is real regardless of where the classical benchmark sits, and it is the problem the field’s published disagreement is actually about. Practitioners committed to deep architectures for reasons this study does not address—multi-asset settings, alternative data, nonlinear payoff structures, or targets other than index level—face the selection problem documented in Table 6 whatever ARIMA does on the S&P 500.

7.2 Why the literature disagrees

The regime×horizon×metric map explains the contradictory published record mechanically. Different cells have different winners: gradient boosting has the best learned RMSE at one day and the worst at five; the LSTM leads nothing on point error but is selected by the gate in bear states at long horizons; the adversarial model is the most frequent fold-level winner at five and twenty days despite a middling aggregate. A study that reports a scalar has sampled one cell, or averaged several with weights set by the accidental composition of its sample. On this reading the literature is not mostly wrong; it is mostly under-specified, and a ranking reported without a regime, a horizon, a loss function and a naive benchmark attached is a summary statistic of a sample rather than a finding.

7.3 Where the method adds value, and where it does not

Three claims are supported by the evidence and worth separating.

It removes selection risk. This is the strongest and best-evidenced claim. The spread between the best and worst learned forecaster reaches 16% of RMSE at twenty days (Table 9), an ex-ante choice cannot reliably land on the right end of it, and RS-MoE delivers the good end at every horizon and beats the ex-post best at two of three.

The advantage grows with horizon and with disagreement. At one day, where the learned models are close together and close to the random walk, combination adds little. At twenty days, where they diverge sharply and degrade unevenly, it adds a great deal. This is the expected behaviour of forecast combination (Bates and Granger, 1969; Timmermann, 2006) and it tells a practitioner where the method is worth deploying.

It stabilises. A two- to eight-fold reduction in across-seed dispersion is a practical benefit in a field where single-seed reporting is standard.

Against these, one claim is *not* supported and we do not make it: that the regime signal in particular, as distinct from learned combination in general, is what produces the gain. The ablation in Table 13 shows the regime-gated combiner beating equal weighting significantly in several cells but statistically indistinguishable from a regime-blind learned stack in all of them. The regime gate’s advantages over the stack are its interpretability—the weight map of Table 10 is a readable statement about which model family carries signal where, which a constant weight vector cannot provide—and the fact that it recovers that structure without any loss of accuracy. Those are real but they are not an accuracy claim, and we do not present them as one.

Table 11: Horizon-stratified performance. RMSE is in index points and is not comparable across columns; comparisons are within a column.

Model	$h = 1$	$h = 5$	$h = 20$
<i>RMSE (index points)</i>			
RW + drift	16.17	33.73	63.42
ARIMA	16.17	33.67	63.37
LSTM	16.25	35.56	79.28
Transformer	16.29	34.88	72.43
GAN	16.53	34.42	72.00
GBM	16.17	37.62	76.43
Equal-weight	16.21	34.46	69.47
Static stack	16.20	34.35	66.04
RS-MoE (regime-only)	16.21	34.29	66.66
RS-MoE (regime+context)	16.22	34.75	67.17
RS-MoE (oracle regime) [†]	16.21	34.27	66.96
Oracle switch [†]	16.12	33.19	61.70
<i>Directional accuracy</i>			
RW + drift	0.538	0.568	0.626
ARIMA	0.521	0.556	0.626
LSTM	0.512	0.485	0.457
Transformer	0.517	0.513	0.531
GAN	0.494	0.461	0.513
GBM	0.525	0.492	0.518
Equal-weight	0.517	0.502	0.517
Static stack	0.523	0.508	0.535
RS-MoE (regime-only)	0.525	0.516	0.523
RS-MoE (regime+context)	0.525	0.517	0.545
RS-MoE (oracle regime) [†]	0.527	0.516	0.519
Oracle switch [†]	0.539	0.572	0.617

7.4 Threats to validity and limitations

Sample coverage. The sample ends in December 2019 and excludes the COVID-19 drawdown and the 2022 inflation bear market. The 2020 episode in particular is qualitatively different—a very fast drawdown and an unusually fast policy-driven rebound—and is exactly the kind of transition on which an online filter is most likely to lag. Extending the sample is the single most valuable robustness check available and requires no change to the harness.

No text-based sentiment channel. The feature set includes market-based risk-appetite measures but no news or social sentiment, for the data-availability reason given in Section 3. Sentiment is a plausible source of exactly the regime-conditional signal this architecture is designed to exploit, and its absence is a genuine limitation on the ceiling of what is demonstrated here rather than a flaw in the comparison, since no model in the pool had access to it.

Single market and instrument. All results are for one index over one sample. Whether the rotation of the best learned model, and the size of the combination gain, transfer to single names, other asset classes or other markets is untested.

Data snooping. Many model×horizon×metric comparisons are made. We adjust p -values by Holm–Bonferroni within the reported families, fix architectures a priori rather than searching them, and select only the two gate shrinkages from data—on a held-out tail of the validation set. Residual multiple-comparison risk remains, as in any study of this shape (Bailey et al., 2014).

Compute budget. All deep models are trained under a fixed and deliberately modest update budget on two CPU cores. This holds optimisation effort constant across folds, which the walk-forward design requires, but the individual learned results should be read as the performance of conventionally-configured, moderately-trained instances rather than as an upper bound on each family. Since RS-MoE’s contribution is combination rather than capacity, we expect the qualitative result to be robust to larger budgets, but we have not demonstrated that.

Economic assumptions. Sharpe figures assume a fixed per-unit-turnover cost and unit leverage and ignore borrowing costs and market impact. Directional and economic conclusions in this literature are frequently fragile to these assumptions, which is why all three lenses are reported side by side rather than leading with the most flattering.

8 Conclusion and Future Work

This paper began from an observation about the applied forecasting literature: that recurrent, attentional, generative and tree-based models have each been reported as the best equity forecaster, and that this disagreement is better explained by the conditionality of forecasting skill than by the incompetence of any particular study. We built a harness that removes the cross-study confounds—one 69-feature causal substrate, one windowing scheme, one walk-forward protocol, one scoring module—and used it to establish the diagnostic claim directly and then to act on it.

The diagnostic claim holds. The best learned forecaster is not stable: it rotates across folds and across horizons, so the question “which learned model is best” has an ex-post answer and no reliable ex-ante one. Within a single controlled design, the same models are ranked differently by squared error and by risk-adjusted return, and differently at one day than at twenty. The published

contradictions are reproduced here as a property of the problem rather than of the studies.

The constructive claim holds in the terms it was posed. RS-MoE records 32 statistically significant wins and no significant losses against the four learned forecasters after multiplicity adjustment; it beats the *ex post* best learned model at five- and twenty-day horizons without foreknowledge of its identity, by 0.36% and 7.41% of RMSE; its advantage widens with the horizon, which is where the individual models diverge most; and it is two to eight times more stable across random seeds than the experts it combines. Its gate is interpretable, and what it learns is economically legible: weight migrates from attentional and tree-based experts toward parsimonious linear structure as the horizon lengthens, and toward nonlinear capacity within the bear regime.

We are equally explicit about the boundaries. Classical baselines—ARIMA and a random walk with drift—remain unbeaten on aggregate point error over this sample, and we report that in the paper’s first results table rather than deferring it. The regime signal, as distinct from learned combination in general, is not separately established: RS-MoE significantly beats equal weighting in several cells but is statistically indistinguishable from a regime-blind learned stack. And an infeasible oracle switch still beats every causal method, while an RS-MoE given the true regime label performs identically to one given only the filtered posterior—which localises the remaining headroom in weight estimation rather than regime detection.

Where the value lies. Read together, these results describe a method whose contribution is not raw accuracy against every possible benchmark but the resolution of a selection problem. The practitioner’s difficulty in this literature is not that no learned model works; it is that a different one works each time, and the cost of choosing wrongly reaches sixteen percent of RMSE at the monthly horizon. RS-MoE removes that choice, delivers the favourable end of the range without foreknowledge, and does so with a gate small enough to interpret and stable enough to reproduce. Those properties are what make it deployable, and they are orthogonal to whether a linear benchmark happens to be strong on a particular index over a particular sample.

Future work. Four directions follow from the evidence rather than from speculation. First, the estimator: the oracle decomposition identifies weight estimation, not regime identification, as binding, so regime-dependent shrinkage strength, hierarchical pooling of weights across horizons, and Bayesian model averaging with regime-conditional priors attack the constraint directly and none requires a better regime model. Second, the information set: a FinBERT-scored news and social sentiment channel (Dahal et al., 2023) is the most plausible source of

additional regime-conditional signal and is the extension we would prioritise, subject to obtaining a daily series with the coverage this sample requires. Third, coverage: extending through the COVID-19 and 2022 episodes, moving to intraday data where regime transitions are sharper, and replicating on single names and other asset classes. Fourth, the architecture: replacing the two-stage HMM-then-gate pipeline with an end-to-end differentiable regime detector trained on the forecasting loss, so that the latent state is optimised for the decision it informs rather than for the likelihood of returns.

A methodological closing note. A headline ranking of forecasting models, reported without a regime, a horizon, a loss function and a naive benchmark attached, is not a finding about the models. It is a summary statistic of a particular sample. The field’s long-running disagreement about which architecture wins is best read not as a controversy awaiting resolution but as a measurement that was never conditioned properly—and, once it is, as a selection problem with a constructive answer.

A Seed Dispersion, Verification, Hyperparameters and Compute

Seed dispersion. Table 14 in the main text reports across-seed mean and standard deviation for every model at every horizon. Differences discussed in Section 6 that are of the same order as the relevant dispersion are flagged there rather than asserted.

Verification. Four checks were run against the released artefacts.

Non-anticipativity of the gate input. The filtered posterior γ_t was computed on the full series and again on a series truncated at t , for several truncation points. The two agree to machine precision, confirming that the forward recursion admits no future information. The smoothed decoding used for stratification is a separate object and enters no forecast.

Test independence. In every fold, the validation slice ends strictly before the test block begins, verified programmatically over all folds.

Test-statistic correctness. The Diebold–Mariano implementation was checked against an independently computed t -statistic on synthetic loss differentials at $h = 1$, where the HAC correction is inert; the two agree to four decimal places, with the sign convention confirmed.

Numerical provenance. Every number appearing in the text and tables is generated programmatically from the run artefacts by the analysis scripts; none is transcribed by hand.

A failure mode worth documenting. An earlier version of the gate produced weights that never departed from uniform, and would have been reported as a null result for regime conditioning. The cause was not the science but the optimisation: the squared-error objective on log-returns is of order 10^{-4} , which leaves softmax logits essentially unmoved under any reasonable learning rate and iteration budget. Dividing the loss by $\text{Var}(\rho_{t,h})$ resolves it completely, and was verified on synthetic data where the correct per-regime weights are known. We record this because a scale-sensitive objective is easy to write, produces a plausible-looking null, and we expect it is silently present in other softmax-gated combiners applied to return data.

Search spaces. ARIMA orders are selected by BIC over $(p, q) \in \{0, \dots, 3\}^2 \setminus \{(0, 0)\}$ with $d = 1$ fixed by the ADF/KPSS analysis. The LSTM, Transformer, adversarial and tree configurations were fixed a priori at conventional values rather than searched, to avoid the multiple-comparison inflation that architecture search on a single dataset induces (Bailey et al., 2014). The only quantities selected from data are the two gate shrinkages (λ, γ) , chosen on a held-out final 30% of the pooled validation set. The absence of an architecture search is a strength for the paper’s specific claim—which concerns combination given a fixed pool—and a limitation for any claim about the maximum attainable accuracy of each family.

Compute. All experiments ran on two CPU cores with no GPU. Under the fixed update budgets of Section 5, one fold-seed combination—one adversarial model plus three LSTMs, three Transformers and three tree ensembles across three horizons—takes approximately three minutes, and the complete 6×3 grid together with ARIMA refitting completes in roughly 1.2 hours. The combiner stage reads a cached expert layer and re-runs in minutes, so the combination design can be revised without retraining any expert. The modest budget is itself informative: the contribution turns on the information the gate is given and how its weights are estimated, not on the capacity of the experts.

B Reproducibility Statement

All models are implemented behind a single interface and scored by shared metric code; hyperparameter spaces, seeds and software versions are logged; and the harness ingests any OHLCV CSV, so the full pipeline—feature construction, regime labelling, walk-forward training, combination and significance testing—can be reproduced end to end from the released code. Every input series is a public dataset pinned by commit. Every table and figure

is generated programmatically from the run artefacts by a single analysis script.

References

- Ang, A. and Bekaert, G. (2002). International asset allocation with regime shifts. *Review of Financial Studies*, 15(4):1137–1187.
- Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *ICML*.
- Bailey, D. H., Borwein, J., Lopez de Prado, M., and Zhu, Q. J. (2014). Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-sample performance. *Notices of the AMS*, 61(5):458–471.
- Bao, W., Yue, J., and Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLoS ONE*, 12(7):e0180944.
- Bates, J. M. and Granger, C. W. J. (1969). The combination of forecasts. *Operational Research Quarterly*, 20(4):451–468.
- Box, G. E. P. and Jenkins, G. M. (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day.
- Cont, R. (2001). Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance*, 1(2):223–236.
- Dahal, K. R. et al. (2023). A comparative study on effect of news sentiment on stock price prediction with deep learning architecture. *PLoS ONE*, 18(4):e0284695.
- Dickey, D. A. and Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74(366):427–431.
- Diebold, F. X. and Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3):253–263.
- Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance*, 25(2):383–417.
- Fischer, T. and Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *NeurIPS*.

- Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies*, 33(5):2223–2273.
- Guidolin, M. and Timmermann, A. (2007). Asset allocation under multivariate regime switching. *Journal of Economic Dynamics and Control*, 31(11):3503–3544.
- Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. (2017). Improved training of wasserstein gans. In *NeurIPS*.
- Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2):357–384.
- Harvey, D., Leybourne, S., and Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, 13(2):281–291.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70.
- Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87.
- Jordan, M. I. and Jacobs, R. A. (1994). Hierarchical mixtures of experts and the em algorithm. *Neural Computation*, 6(2):181–214.
- Kim, H. Y. and Won, C. H. (2018). Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models. *Expert Systems with Applications*, 103:25–37.
- Krauss, C., Do, X. A., and Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the s&p 500. *European Journal of Operational Research*, 259(2):689–702.
- Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., and Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, 54(1–3):159–178.
- Lim, B., Arik, S. Ö., Loeff, N., and Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764.
- Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLoS ONE*, 13(3):e0194889.
- Malkiel, B. G. (2003). The efficient market hypothesis and its critics. *Journal of Economic Perspectives*, 17(1):59–82.
- Newey, W. K. and West, K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55(3):703–708.
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.
- Sezer, O. B., Gudelek, M. U., and Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review 2005–2019. *Applied Soft Computing*, 90:106181.
- Siami-Namini, S., Tavakoli, N., and Siami Namin, A. (2018). A comparison of arima and lstm in forecasting time series. In *IEEE ICMLA*.
- Staffini, A. (2022). Stock price forecasting by a deep convolutional generative adversarial network. *Frontiers in Artificial Intelligence*, 5:837596.
- Takahashi, S., Chen, Y., and Tanaka-Ishii, K. (2019). Modeling financial time-series with generative adversarial networks. *Physica A*, 527:121261.
- Timmermann, A. (2006). Forecast combinations. *Handbook of Economic Forecasting*, 1:135–196.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *NeurIPS*.
- Welch, I. and Goyal, A. (2008). A comprehensive look at the empirical performance of equity premium prediction. *Review of Financial Studies*, 21(4):1455–1508.
- Wiese, M., Knobloch, R., Korn, R., and Kretschmer, P. (2020). Quant gans: Deep generation of financial time series. *Quantitative Finance*, 20(9):1419–1440.
- Yoon, J., Jarrett, D., and van der Schaar, M. (2019). Time-series generative adversarial networks. In *NeurIPS*.
- Zhang, K., Zhong, G., Dong, J., Wang, S., and Wang, Y. (2019). Stock market prediction based on generative adversarial network. *Procedia Computer Science*, 147:400–406.

Table 12: Regime-stratified RMSE. Bold marks the best feasible model within each regime×horizon cell.

Model	Bull	Bear	Sideways
$h = 1$			
RW + drift	11.52	24.14	14.59
ARIMA	11.53	24.12	14.63
LSTM	11.56	24.30	14.63
Transformer	11.59	24.39	14.63
GAN	11.86	24.61	14.90
GBM	11.52	24.14	14.58
Equal-weight	11.55	24.22	14.61
Static stack	11.54	24.22	14.60
RS-MoE (regime-only)	11.54	24.24	14.60
RS-MoE (regime+context)	11.56	24.24	14.61
RS-MoE (oracle regime) [†]	11.54	24.23	14.60
Oracle switch [†]	11.49	24.08	14.53
$h = 5$			
RW + drift	27.41	45.00	32.28
ARIMA	27.40	44.79	32.31
LSTM	28.04	47.94	34.47
Transformer	28.13	47.02	33.18
GAN	28.02	45.65	33.13
GBM	29.82	51.10	35.97
Equal-weight	27.78	45.96	33.23
Static stack	27.85	45.74	33.02
RS-MoE (regime-only)	27.73	45.78	32.94
RS-MoE (regime+context)	27.74	47.29	32.94
RS-MoE (oracle regime) [†]	27.70	45.70	32.97
Oracle switch [†]	27.27	44.09	31.62
$h = 20$			
RW + drift	50.58	82.92	63.19
ARIMA	50.54	82.73	63.23
LSTM	66.76	98.43	79.69
Transformer	54.15	98.53	72.31
GAN	62.31	88.12	71.61
GBM	57.55	100.35	79.05
Equal-weight	55.07	89.78	70.43
Static stack	52.49	85.70	66.52
RS-MoE (regime-only)	52.72	87.50	66.55
RS-MoE (regime+context)	53.59	87.99	66.76
RS-MoE (oracle regime) [†]	52.77	86.39	68.29
Oracle switch [†]	49.84	81.17	60.40

Table 13: Ablation: RS-MoE against regime-agnostic combination on identical experts and folds, isolating the marginal contribution of the regime signal from the generic benefit of averaging. Positive favours RS-MoE.

Bucket	vs Equal-weight		vs Static stack	
	Skill (%)	p	Skill (%)	p
$h = 1$				
All	-0.02	1.000	-0.04	0.189
Bull	+0.02	1.000	-0.03	1.000
Bear	-0.10	1.000	-0.09	0.064
Sideways	+0.09	1.000	+0.03	1.000
$h = 5$				
All	+0.48	0.003*	+0.17	1.000
Bull	+0.19	1.000	+0.44	1.000
Bear	+0.38	1.000	-0.08	1.000
Sideways	+0.86	0.000*	+0.24	1.000
$h = 20$				
All	+4.04	0.000*	-0.94	0.344
Bull	+4.25	0.223	-0.44	1.000
Bear	+2.54	0.443	-2.11	0.488
Sideways	+5.50	0.020*	-0.05	1.000

Table 14: Across-seed mean $\pm$ standard deviation over three random seeds. The deterministic models (RW, ARIMA) show zero dispersion by construction. RS-MoE is several times more stable than the stochastic experts it combines.

Model	RMSE	DA	Sharpe
$h = 1$			
RW + drift	16.17 ± 0.00	0.538 ± 0.000	0.24 ± 0.00
ARIMA	16.17 ± 0.00	0.521 ± 0.000	-0.01 ± 0.00
LSTM	16.26 ± 0.04	0.514 ± 0.006	0.07 ± 0.09
Transformer	16.32 ± 0.10	0.511 ± 0.011	0.02 ± 0.12
GAN	16.98 ± 0.32	0.483 ± 0.011	-0.54 ± 0.20
GBM	16.18 ± 0.02	0.523 ± 0.004	0.11 ± 0.16
Equal-weight	16.23 ± 0.02	0.506 ± 0.007	-0.12 ± 0.12
Static stack	16.21 ± 0.03	0.524 ± 0.007	0.10 ± 0.12
RS-MoE (regime-only)	16.22 ± 0.04	0.526 ± 0.007	0.08 ± 0.13
RS-MoE (regime+context)	16.23 ± 0.04	0.521 ± 0.011	-0.01 ± 0.20
RS-MoE (oracle regime) [†]	16.21 ± 0.03	0.525 ± 0.009	0.10 ± 0.11
Oracle switch [†]	16.13 ± 0.01	0.533 ± 0.003	0.37 ± 0.03
$h = 5$			
RW + drift	33.73 ± 0.00	0.568 ± 0.000	0.27 ± 0.00
ARIMA	33.67 ± 0.00	0.556 ± 0.000	0.17 ± 0.00
LSTM	35.94 ± 0.18	0.494 ± 0.013	-0.20 ± 0.01
Transformer	35.37 ± 0.85	0.508 ± 0.003	-0.16 ± 0.04
GAN	38.63 ± 2.04	0.491 ± 0.024	-0.19 ± 0.12
GBM	37.87 ± 0.67	0.494 ± 0.006	-0.00 ± 0.09
Equal-weight	34.64 ± 0.15	0.505 ± 0.030	-0.03 ± 0.19
Static stack	34.44 ± 0.15	0.499 ± 0.024	-0.10 ± 0.14
RS-MoE (regime-only)	34.37 ± 0.26	0.508 ± 0.007	-0.11 ± 0.11
RS-MoE (regime+context)	34.97 ± 0.64	0.513 ± 0.006	-0.05 ± 0.18
RS-MoE (oracle regime) [†]	34.35 ± 0.29	0.511 ± 0.005	-0.08 ± 0.09
Oracle switch [†]	33.39 ± 0.10	0.560 ± 0.007	0.37 ± 0.07
$h = 20$			
RW + drift	63.42 ± 0.00	0.626 ± 0.000	0.28 ± 0.00
ARIMA	63.37 ± 0.00	0.626 ± 0.000	0.28 ± 0.00
LSTM	80.75 ± 2.48	0.464 ± 0.012	-0.26 ± 0.14
Transformer	74.02 ± 3.22	0.531 ± 0.031	-0.02 ± 0.22
GAN	85.61 ± 7.56	0.505 ± 0.067	-0.04 ± 0.26
GBM	76.79 ± 0.69	0.515 ± 0.003	-0.01 ± 0.07
Equal-weight	70.18 ± 0.99	0.524 ± 0.047	0.01 ± 0.13
Static stack	66.68 ± 1.38	0.560 ± 0.056	0.07 ± 0.19
RS-MoE (regime-only)	67.11 ± 0.88	0.552 ± 0.043	-0.01 ± 0.18
RS-MoE (regime+context)	68.09 ± 0.43	0.542 ± 0.029	0.02 ± 0.15
RS-MoE (oracle regime) [†]	67.56 ± 1.45	0.542 ± 0.048	-0.02 ± 0.19
Oracle switch [†]	62.31 ± 0.53	0.611 ± 0.025	0.44 ± 0.07