# Volatility Forecasting $\sigma$

## Stochastic vs. Machine Learning Methods

A comparative analysis of volatility forecasting approaches using EWMA, GARCH-family models, and LSTM neural networks on daily Microsoft (MSFT) returns from 2015 to 2025.

*Thomas Fouilloux, Wai Lok Law, Luv Barnwal, Aayan Zangie*

Data Science Society - 2025 AT Project Showcase

# Contents

1. Data and Methodology
2. EWMA Model
3. EWMA Lambda Selection
4. GARCH and GJR-GARCH
5. GARCH Model Selection Methodology
6. LSTM Architecture and Methodology
7. Pure LSTM Approach
8. Hybrid LSTM Approach
9. Results and Findings
10. Appendix

# Data and Methodology (1/3)

**Dataset Overview**

| Dataset | Ticker | Period | Source |
|---|---|---|---|
| Microsoft | MSFT | 2015-01-01 to 2025-12-01 | Yahoo Finance |

**Chosen Splits Splits**

| Split | Period | Purpose |
|---|---|---|
| In-Sample (Training) | 2015-01-01 to 2021-12-31 | Model estimation & LSTM training |
| Validation | 2020-01-01 to 2021-12-31 | Testing (used occasionally in LSTM tuning) |
| Out-of-Sample (Test) | 2022-01-01 to 2025-12-01 | Final performance evaluation |

**Returns:** Continuously compounded daily returns ($S_t$ = the adjusted closing price at time $t$):

$$r_t = \ln \left( \frac{S_t}{S_{t-1}} \right)$$

# Data and Methodology (2/3)

**Loss Function: QLIKE**

Following Patton (2011), we use QLIKE as our evaluation metric:

- Only QLIKE and MSE preserve robustness when using noisy volatility proxies: $h_t^* = E_{t-1}[\hat{\sigma}_t^2] = \sigma_t^2$
- QLIKE yields higher statistical power in Diebold-Mariano-West tests when compared to MSE
- Less sensitive to extreme outliers common in financial data and it penalises underestimation of volatility

**QLIKE Definition:**

$$\text{QLIKE} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{\sigma^2_{\text{realised},t}}{\sigma^2_{\text{forecast},t}} - \ln\left(\frac{\sigma^2_{\text{realised},t}}{\sigma^2_{\text{forecast},t}}\right) - 1\right)$$

- $\sigma^2_{\text{realised},t}$: Realised variance (average squared returns over forecast horizon)
- $\sigma^2_{\text{forecast},t}$: Model's variance forecast
- **Lower QLIKE = Better forecasting performance**

# Data and Methodology (3/3)

## Methodology Disclaimer

Due to the absence of easily accessible high-frequency intraday data (which is the academic standard):

- We evaluate 1-day forecasts against realised volatility over three horizons:
  - 5-days (one week of trading)
  - 20-days (one month of trading)
  - 25-days (one month of trading)

- This approach reduces the difficulty of forecasting single-day squared returns which is a very noisy estimate of realised volatility (Andersen and Bollerslev, 1998)

- Provides a more stable benchmark for model comparison

**High Frequency Data is the academic standard**, indeed, Hansen & Lunde, 2005 (p.13) explain that many utilise realised volatility computed from intraday returns. This requires high-frequency data not freely available in our case

# EWMA Model

**Exponentially Weighted Moving Average (EWMA)**

The EWMA model estimates conditional variance using a recursive function:

$$\boxed{\hat{\sigma}_t^2 = \lambda\hat{\sigma}_{t-1}^2 + (1-\lambda)r_{t-1}^2}$$

**Parameters:**

- $\hat{\sigma}_t^2$: Variance forecast for time $t$
- $\hat{\sigma}_{t-1}^2$: Previous Variance Estimate for time $t-1$ (i.e. Variance Estimate for Today)
- $r_{t-1}$: Return at time $t-1$
- $\lambda$: Decay factor $\lambda \in \{0, 1\}$ (typically between 0.85 and 0.99)
- **RiskMetrics (p.4) standard:** $\lambda = 0.94$

**Key insight:** Higher values of $\lambda$ assign more weight to past observations, resulting in smoother volatility estimates.

# EWMA Lambda Selection (1/2)

**Hyperparameter Tuning Methodology**

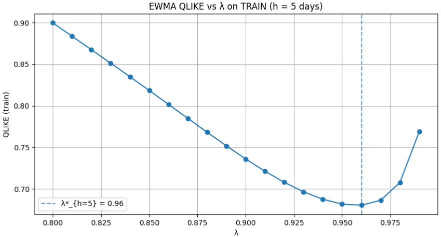
1. Calculate QLIKE for $\lambda \in \{0.80, 0.81, \ldots, 0.99\}$
2. Evaluate across three forecast horizons (5, 20, and 25 days)
3. Select optimal $\lambda$ by minimising QLIKE on the whole training set

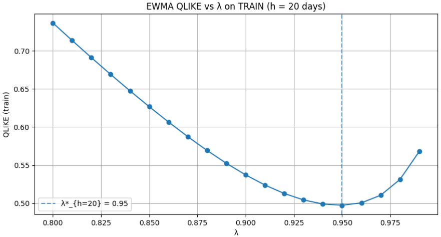
**Optimal Lambda Selection Results:**

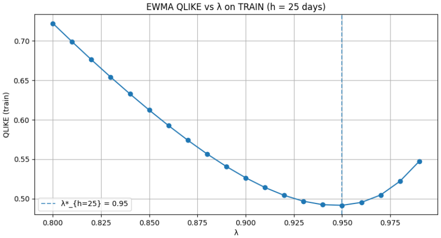
| Horizon | Optimal Lambda |
|---|---|
| 5-day | 0.96 |
| 20-day | 0.95 |
| 25-day | 0.95 |

*Note: Higher decay factors were consistently preferred across all horizons.*

# EWMA Lambda Selection (2/2)


EWMA QLIKE vs λ on TRAIN (h = 5 days)


EWMA QLIKE vs λ on TRAIN (h = 20 days)


EWMA QLIKE vs λ on TRAIN (h = 25 days)

# GARCH and GJR-GARCH (1/2)

**GARCH(1,1) Model**

Generalised Autoregressive Conditional Heteroskedasticity captures “volatility clustering”:

$$\boxed{\sigma_t^2 = \omega + \alpha\varepsilon_{t-1}^2 + \beta\sigma_{t-1}^2}$$

- $\omega > 0$: Baseline variance (constant term)
- $\alpha \geq 0$: ARCH coefficient (reaction to recent shocks)
- $\beta \geq 0$: GARCH coefficient (persistence)
- $\varepsilon_{t-1} = r_{t-1} - \mu$: Return residual at $t - 1$

**Unconditional (Long-run) Variance:**

$$\sigma_{LR}^2 = \frac{\omega}{1 - \alpha - \beta} \quad \text{(requires } \alpha + \beta < 1 \text{ for stationarity)}$$

# GARCH and GJR-GARCH (2/2)

## GJR-GARCH(1,1) Model, Capturing the Leverage Effect

Following Glosten, Jagannathan, and Runkle (1993), we add an asymmetric term:

$$\boxed{\sigma_t^2 = \omega + \alpha\varepsilon_{t-1}^2 + \gamma\varepsilon_{t-1}^2\mathbb{I}_{[\varepsilon_{t-1}<0]} + \beta\sigma_{t-1}^2}$$

**Additional Parameter:**

- $\gamma \geq 0$: Leverage coefficient (asymmetric response)
- $\mathbb{I}_{[\varepsilon_{t-1}<0]}$: Indicator function (equals 1 when returns residuals are negative, otherwise 0)

**Leverage Effect:**

- Negative shocks impact: $(\alpha + \gamma)$
- Positive shocks impact: $\alpha$

*Note: the rational behind the leverage coefficient $\gamma$ is that negative returns increase volatility more than positive returns of equal magnitude.*

# GARCH Model Selection Methodology (1/2)

**Information Criteria for Model Selection**

Following Tsay (2005), we use information criteria rather than QLIKE for GARCH selection:

**Akaike Information Criterion (AIC):**

$$\text{AIC} = 2k - 2\ln(\hat{L})$$

**Bayesian Information Criterion (BIC):**

$$\text{BIC} = k\ln(n) - 2\ln(\hat{L})$$

- $k$: Number of parameters
- $\hat{L}$: Maximised likelihood
- $n$: Sample size

**Why BIC?** Penalises model complexity more heavily than AIC, favouring parsimonious specifications which are less prone to overfitting.

# GARCH Model Selection Methodology (2/2)

## Model Selection Results

Estimated all combinations of GARCH($p, q$) and GJR-GARCH($p, o, q$) for $p, q, o \in \{1, 2, 3, 4, 5\}$:

| Rank | Model | AIC | BIC |
|---|---|---|---|
| 1 (AIC) | GJR-GARCH(5,5,1) | 6332.23 | 6403.39 |
| **1 (BIC)** | **GJR-GARCH(1,1,1)** | 6358.02 | **6385.39** |
| 2 (AIC) | GJR-GARCH(5,5,5) | 6334.15 | 6427.21 |
| **2 (BIC)** | **GARCH(1,1)** | 6366.19 | **6388.08** |

**Selected Models:**

- **Primary:** GJR-GARCH(1,1) - Best BIC
- **Benchmark:** GARCH(1,1) - Reference comparison (and second best BIC)

*Our findings are consistent with Hansen & Lunde (2001) (p.27): "We do not find much evidence that the GARCH(1,1) model is outperformed."*

# LSTM Architecture and Methodology

## Literature-Based/In-sample Hyperparameter Selection

| Parameter | Value | Justification |
|---|---|---|
| Lookback window | 22 days | Roszyk & Slepaczuk (2024) |
| LSTM units | 128 | Roszyk & Slepaczuk (2024) |
| LSTM layers | 1 | In-sample testing |
| Dropout rate | 0.2 | Srivastava et al. (2014) |
| Dense hidden units | 128 | In-sample testing |
| Dense hidden layers | 1 | In-sample testing |
| Epochs | 50 (early based on a validation set) | In-sample testing |
| Batch size | 64 | In-sample testing |
| Activation | tanh | Roszyk & Slepaczuk (2024) |
| Validation set early stopping condition | 80/20% split on the training set | In-sample testing |

**Other key Literature used:**

- Kim & Won (2018): A hybrid model integrating LSTM with multiple GARCH-type models
- Kakade et al. (2021): A Hybrid Ensemble Learning Garch-Lstm Based Approach
- Hu, Ni & Wen (2020): A hybrid deep learning approach by integrating LSTM-ANN networks with GARCH model for copper price volatility prediction

## Forward-Looking Pure LSTM

Directly predicts future realised variance over different horizons.

**Features** (all backward-looking, available at time $t$):

- **Lagged returns** (last 5 days): $r_{t-1}, r_{t-2}, \ldots, r_{t-5}$
- **Lagged squared returns** (ARCH effect): $r_{t-1}^2, r_{t-2}^2, \ldots, r_{t-5}^2$
- **Rolling volatilities**: windows of 5, 10, and 20 days
- **Rolling variances**: windows of 5, 10, and 20 days

**Target:** is the realised variance over horizon $h$ defined as per the below

$$\sigma^2_{\text{target},t} = \frac{1}{h}\sum_{i=1}^{h} r_{t+i}^2$$

where $h$ is the forecast horizon (5, 20, or 25 days).

*The model learns to predict average squared returns over the specified horizon directly from historical features exclusively derived from returns (i.e. no external data is added to the LSTM).*

# Hybrid LSTM Approach

## GARCH-LSTM Hybrid Model

We propose a novel GARCH-LSTM hybrid model: our model's objective is to predict how “wrong” GJR-GARCH forecasts is compared to realised volatility.

- This also draws from Hu, Ni & Wen (2020)'s research as they demonstrated that GARCH forecasts can serve as informative features to increase LSTM predictive power.

**Features:**

- **Same features as the pure LSTM**
- **GJR-GARCH forecasts** (persistence signal)

**Target:** Adjustment Ratio, i.e. the LSTM predicts how much to “fix” GJR-GARCH's forecast:

$$\rho_t = \frac{\sigma^2_{\text{realised},t}}{\hat{\sigma}^2_{\text{GJR},t}}$$

**Final Forecast:**

$$\boxed{\hat{\sigma}^2_{\text{hybrid},t} = \hat{\sigma}^2_{\text{GJR},t} \times \hat{\rho}_{\text{LSTM},t}}$$

# Results and Findings

**Results:**

| Horizon | EWMA | GARCH | GJR | LSTM | Hybrid | Best |
|---|---|---|---|---|---|---|
| 5-day | 0.4222 | 0.4251 | **0.3969** | 0.5828 | 0.5906 | GJR |
| 20-day | 0.2049 | 0.1903 | **0.1529** | 0.2494 | 0.2638 | GJR |
| 25-day | 0.1866 | 0.1723 | **0.1332** | 0.2050 | 0.2079 | GJR |

Table: Out-of-Sample QLIKE Loss Comparison (Lower is Better)

**Main Findings:**

- Parsimonious GARCH-family models, specifically **GJR-GARCH(1,1,1)**, outperform complex LSTM architectures for daily equity volatility forecasting in terms of QLIKE.
- Our findings align with Hansen and Lunde (2005), who found that simple GARCH(1,1) is difficult to beat in out-of-sample forecasting.
- LSTM underperformance was likely caused by the constraint of exclusively using daily returns as a basis for features. Indeed, contrary to GARCH and EWMA, LSTMs perform best with higher dimensional data. Excluding exogenous variables like VIX, volume, or sentiment may significantly limit the model's predictive ability.

# Thank You! $\sigma$

**Questions?**

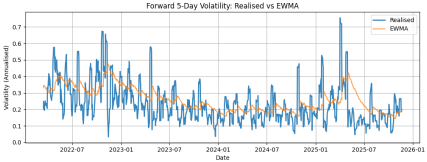
*Thomas Fouilloux, Wai Lok Law, Luv Barnwal, Aayan Zangie*

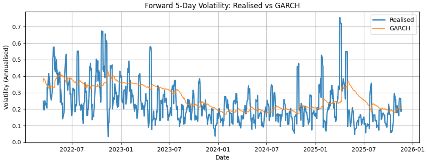
Data Science Society - 2025 AT Project Showcase

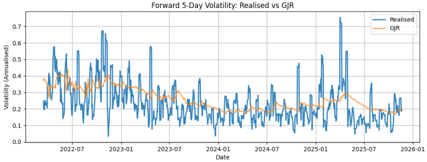
# Appendix

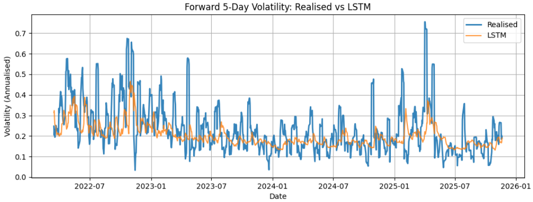
$\sigma$

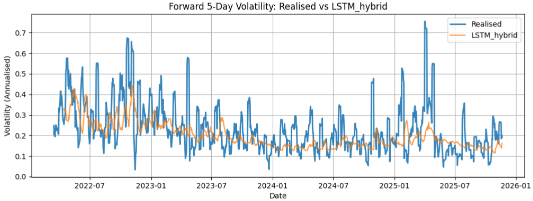
# 5-Day Volatility Forecasts





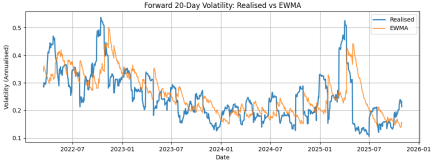


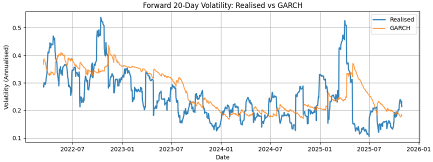


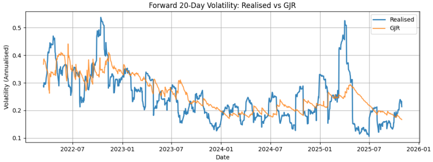


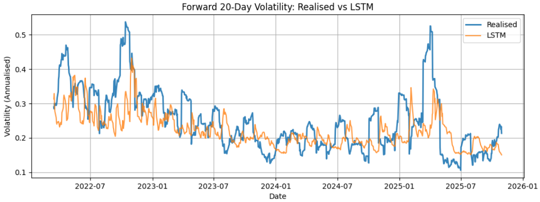
# 20-Day Volatility Forecasts

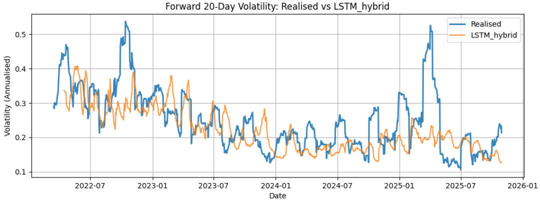
σ



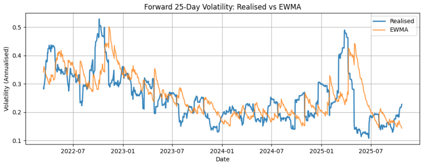


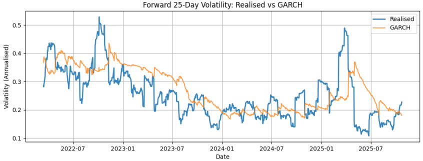


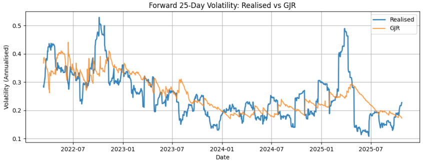


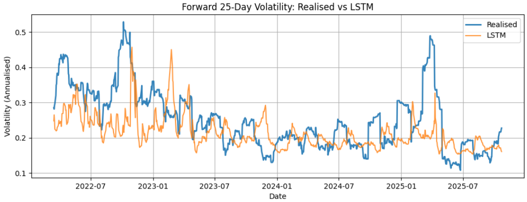


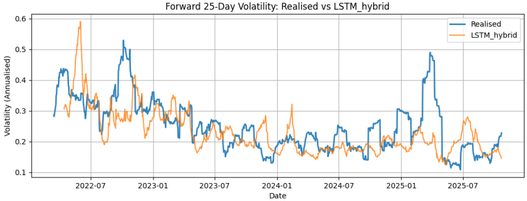
# 25-Day Volatility Forecasts


Forward 25-Day Volatility: Realised vs EWMA


Forward 25-Day Volatility: Realised vs GARCH


Forward 25-Day Volatility: Realised vs GJR


Forward 25-Day Volatility: Realised vs LSTM


Forward 25-Day Volatility: Realised vs LSTM_hybrid

# Key Observations

- **GJR-GARCH Dominance:** The leverage effect captured by the asymmetric term provides consistent forecasting improvements across all horizons.
- **LSTM Underperformance:** Both pure and hybrid architectures fail to beat GJR-GARCH. The gap is widest at the 20-day horizon (QLIKE: 0.25/0.26 vs 0.15).
- **Hybrid Approach Limitations:** Incorporating GJR forecasts as features into the LSTM degraded performance rather than improving it, suggesting the neural network failed to leverage the econometric signal effectively.
- **Horizon Sensitivity:** The performance gap is smaller at shorter horizons (5-day) compared to longer horizons (20-day and 25-day). Both LSTMs perform worse in relative terms at the 20-day horizon.
- **Similar LSTM performance at 5- and 25-day horizons:** While Pure LSTM performs better than Hybrid at all horizons, we note that the hybrid model's performance is relatively close to the pure LSTM at 5- and 25-day horizons. The hybrid model deteriorates more significantly at the 20-day horizons, leading the hybrid model to be slightly less consistent than the Pure LSTM on average.

# Conclusion

**Main Findings:**

- Parsimonious GARCH-family models, specifically **GJR-GARCH(1,1,1)**, outperform complex LSTM architectures for daily equity volatility forecasting in terms of QLIKE.
- Our findings align with Hansen and Lunde (2005), who found that simple GARCH(1,1) is difficult to beat in out-of-sample forecasting.

## Interpreting the LSTM Results

We do not consider the negative LSTM results as evidence of fundamental limitations of deep learning models for volatility forecasting, indeed, they highlight specific challenges:

1. **Noise & Sample Size:** Daily returns are noisy, and ≈2,700 observations may be insufficient for deep learning pattern recognition.
2. **Feature Engineering:** Using only price-derived features limits the LSTM. It likely requires exogenous variables (Volume, VIX, Sentiment, etc.) to add value beyond GARCH.
3. **Evaluation Proxy:** Using squared daily returns as a target introduces measurement error that disadvantages models trying to learn subtle non-linear patterns.

# Limitations and Future Work

- **High-Frequency Data:** Future work should use intraday data to construct **Realised Volatility**. Andersen and Bollerslev (1998) showed this dramatically improves evaluation accuracy compared to squared daily returns.
- **Alternative Architectures:** Transformer-based models with attention mechanisms have been used in recent literature (Soroka and Arzyn, 2025).
- **Broader Scope:** Robustness should be tested across multiple assets, sectors, and asset classes rather than a single equity (MSFT).
- **Regime Dependence:** The test period (2022-2025) covers extreme volatility shifts. Regime-switching models or separate training for high/low volatility periods could improve accuracy.
- **Data Volume:** Extending the training period or using a panel of assets would provide the larger dataset deep learning models typically require to generalise effectively.