Extension Draft

Abdulrahman Zainab, 663423

June 2025

1 Forecasting Cryptocurrencies

The out-of-sample performance of different volatility forecasting models across six cryptocurrencies is reported in this section. For the quantitative evaluation of our models, we employ two benchmark metrics. First, the Root Mean Squared Error (RMSE), which calculates the square root of the average squared differences between predicted ($\hat{y}_t$) and observed (y_t) volatility over T time periods:

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}.$$

Due to its quadratic structure, RMSE effectively penalizes larger prediction errors, ensuring they are not obscured by smaller ones [Hodson, 2022]. Secondly, we use Quasi Likelihood (QLIKE), a standard metric in volatility forecasting known for its sensitivity to significant prediction errors, particularly at low volatility levels. The QLIKE loss function is calculated as:

$$\text{QLIKE} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{y_t}{\hat{y}_t} - \log\left(\frac{y_t}{\hat{y}_t}\right) - 1\right).$$

Importantly, QLIKE penalizes under predictions more heavily than over predictions. This asymmetry is particularly relevant in financial risk management contexts, where underestimating volatility typically carries more severe consequences. This robustness to noise in volatility proxies has been demonstrated by Patton [2011]. For the evaluation of volatility forecasting, we employed a rolling window approach with a forecast horizon of one day. The estimation window consisted of 30 days, after which the model was used to predict the next day (that is, day 31). This process was repeated throughout the dataset, shifting the training and testing window forward each time. A further investigation using a 60 day rolling window remains to be conducted.

The results in Table 2 and Table 1 show a clear superiority of the Local Linear Forest (LLF) variations compared to traditional econometric models such as GARCH and HAR RV, as well as to general machine learning algorithms like Random Forest (RF) and XGBoost. For instance, GARCH consistently shows the highest RMSE and QLIKE values across all cryptocurrencies; for Bitcoin (BTC), its RMSE is 0.583 compared to LassoLLF's 0.173, representing a substantial reduction in error. Similarly, GARCH's QLIKE for BTC is 8.228, whereas SparseGroupLLF achieves 0.032, indicating a dramatic improvement in predictive accuracy. HAR-RV also demonstrates higher errors than the LLF variants, with BTC's HAR-RV RMSE at 0.309 compared to LassoLLF's 0.173. Among the LLF variants, SparseGroupLLF appears to be the most robust overall, achieving the lowest QLIKE value across all six cryptocurrencies, demonstrating a consistent ability to accurately estimate volatility in a relative sense. In terms of RMSE, SparseGroupLLF obtains the lowest values for three coins: ETH (0.009), XRP (0.014), and DOGE (0.014). For the remaining three coins (BTC: 0.173, XLM: 0.012, SHIB: 0.017), LassoLLF yields the best RMSE performance, indicating its strong capability in minimizing absolute prediction errors for these specific assets. While standard LLF performs well, its RMSE values are consistently slightly higher than those of the penalized variants across almost all cryptocurrencies. These results collectively highlight the significant benefit of regularization techniques (Lasso and Sparse Group Lasso) in enhancing forecast accuracy.

when distinguishing between large cap cryptocurrencies (such as Bitcoin, Ethereum, and Ripple) and mid or meme cap coins (like Dogecoin, Stellar, and Shiba Inu), the models generally exhibit similar relative

performance. The improved accuracy of the LLF variants over traditional and general machine learning models remains consistent across both categories, suggesting their robustness in capturing diverse volatility dynamics regardless of market capitalization or liquidity. Table 2 reveals high QLIKE values for the GARCH model across all cryptocurrencies. This is primarily because the GARCH model often produces very small forecast values that are close to zero, especially during periods of low actual volatility. When these near-zero forecasts are used in the QLIKE formula denominator, even small discrepancies from the actual volatility can lead to disproportionately large QLIKE scores, indicating severe penalties for underpredicting volatility.

Table 1: Out-of-sample RMSE per model en cryptocurrency

Coin	**GARCH**	**HAR-RV**	**LLF**	**LassoLLF**	**RF**	**SparseGroupLLF**	**XGBoost**
BTC	0.5833	0.3091	0.1819	**0.1725**	0.2613	0.1739	0.2118
ETH	0.0340	0.0154	0.0090	0.0088	0.0132	**0.0087**	0.0102
XRP	0.0429	0.0244	0.0147	0.0143	0.0207	**0.0142**	0.0150
DOGE	0.0507	0.0249	0.0151	0.0152	0.0223	**0.0143**	0.0164
XLM	0.0421	0.0218	0.0119	**0.0117**	0.0191	0.0119	0.0137
SHIB	0.0513	0.0260	0.0186	**0.0168**	0.0252	0.0173	0.0220

Table 2: Out-of-sample QLIKE per model en cryptocurrency

Coin	**GARCH**	**HAR-RV**	**LLF**	**LassoLLF**	**RF**	**SparseGroupLLF**	**XGBoost**
BTC	8.2278	0.1243	0.0421	0.0341	0.0854	**0.0324**	0.0445
ETH	182.2534	0.3572	0.0360	0.0309	0.0710	**0.0288**	0.0360
XRP	154.1759	1.7378	0.0541	0.0353	0.0971	**0.0324**	0.0414
DOGE	142.8138	0.1215	0.0380	0.0328	0.0836	**0.0313**	0.0401
XLM	181.6911	2.4857	0.0334	0.0297	0.0753	**0.0286**	0.0376
SHIB	136.9702	2.8028	0.0517	0.0455	0.0956	**0.0419**	0.0539

1.1 Significance Testing of Model Performance

To assess whether the models differ significantly, we apply the Friedman test-a nonparametric alternative to repeated-measures ANOVA that compares average ranks across paired blocks [Demšar, 2006]. It tests the null hypothesis that all models perform equally in terms of QLIKE. Table 3 shows highly significant χ^2 values (all $p < 0.001$) for every cryptocurrency, so we reject equal performance and conclude clear differences across the seven models.

Table 3: Friedman test results (QLIKE) by cryptocurrency

Coin	χ^2	p-value
BTC	2638.48	< 0.001
DOGE	2854.16	< 0.001
ETH	1131.79	< 0.001
SHIB	2711.08	< 0.001
XLM	2711.87	< 0.001
XRP	2811.95	< 0.001

The Nemenyi post-hoc test then performs pairwise comparisons of those average ranks while controlling the family wise error rate [Demšar, 2006]. Table 4 shows that LLF, LassoLLF and SparseGroupLLF do not differ significantly from each other, whereas RF, GARCH and HARRV all differ sharply ($p = 0$) from the LLF family. XGBoost aligns with the LLF group ($p > 0.05$) rather than with the tree-based or GARCH/HARRV models. The full results for the other coins appear in the appendix.

Table 4: Nemenyi post-hoc p-values (QLIKE) for BTC; values < 0.001 shown as 0

Model	LLF	LassoLLF	SparseGroupLLF	RF	GARCH	HARRV	XGBoost
LLF	-	0.7185	0.0511	0	0	0	0.8139
LassoLLF	-	-	0.8032	0	0	0	0.0541
SparseGroupLLF	-	-	-	0	0	0	0.0003
RF	-	-	-	-	0	0	0
GARCH	-	-	-	-	-	0	0
HARRV	-	-	-	-	-	-	0
XGBoost	-	-	-	-	-	-	-

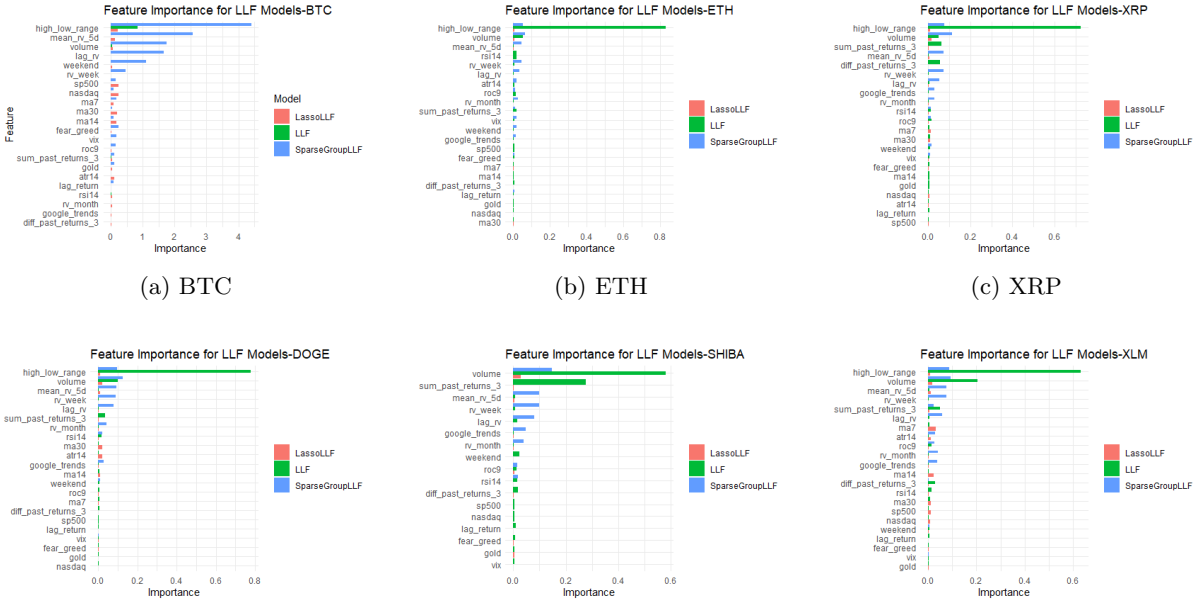
1.2 Feature importance

In Figure 1, we present the feature importance plots of the three LLF variants for the volatility forecasts of six different cryptocurrencies (BTC, ETH, XRP, DOGE, SHIB, and XLM). For almost all coins, we observe that the feature "high minus low" is dominant, with the exception of Shiba Inu. This suggests that intraday price swings are a robust predictor of future volatility, regardless of the type of coin.

We also observe that "volume" is a key feature across all cryptocurrencies, particularly for the meme and mid-cap coins SHIB, DOGE, and XLM. This aligns with the nature of these assets, which often experience hype-driven price surges, making trading volume a crucial signal. In contrast, the importance of volume is slightly lower for large-cap coins like BTC and ETH, where other macroeconomic or fundamental factors may play a more prominent role.

Three additional features that consistently rank high across most coins are "sum of past returns over three days", "mean realized volatility over 5 days" and "lagged realized volatility". This indicates that simple and interpretable metrics can provide substantial predictive power in volatility forecasting.

It is also notable that exogenous variables such as the "S&P500", "VIX", "gold", "fear and greed index", and "Google Trends" generally play a minimal or negligible role in the models. This implies that, within this framework, crypto volatility appears to be largely endogenously driven, and that traditional market sentiment indicators possess limited explanatory power.Finally, it is evident that SparseGroupLLF generally exhibits a broader but more refined variable selection pattern-assigning smaller but consistently nonzero importance to selected variables-which aligns with the intended design of this method: balancing between group structure and sparsity. LassoLLF appears to apply more aggressive selection, often activating only a few features, while the standard LLF produces a more diffuse importance distribution.



(a) BTC (b) ETH (c) XRP

(d) DOGE (e) SHIBA (f) XLM

Figure 1: Feature importance per LLF-model en cryptocurrency

2 Simulation Study: EGARCH-driven Volatility Forecasting

In this Monte Carlo simulation [Mooney, 1997], we investigate the out-of-sample performance of six different models applied to volatility forecasting: BART, LLF, LassoLLF, SparseGroupLLF, RF, and XGBoost. The simulation is designed to analyze the influence of crucial data properties, namely the sample size (n), the dimensionality of the features (d), and the degree of heteroscedasticity (σ). The data is simulated according to the EGARCH(1,1) framework, where the time varying volatility (σ_t) scales the error term in the linear regression model. The EGARCH(1,1) model is defined as:

$$\log(\sigma_t^2) = \omega + \alpha \left(\frac{|\varepsilon_{t-1}|}{\sigma_{t-1}} - \sqrt{\frac{2}{\pi}} \right) + \gamma \frac{\varepsilon_{t-1}}{\sigma_{t-1}} + \beta \log(\sigma_{t-1}^2)$$

The dependent variable is created by $y_t = X_t\beta + \epsilon_t$, whereby $\epsilon_t \sim N(0, \sigma \cdot \sigma_t)$, and only five of the p regression coefficients (β) are non-zero, which introduces a scenario with both relevant and irrelevant features. For each combination of $n \in \{250, 500, 1000\}$, $p \in \{5, 20, 40\}$, and $\sigma \in \{1, 5, 10\}$, the simulation is repeated 10 times to ensure the robustness of the results. Each simulated dataset is split into an 80% training set and a 20% test set to determine the average out-of-sample RMSE, an unbiase d measure of predictive accuracy.

Regarding individual model performance, our results show that the Local Linear Forest (LLF) variants namely, the standard LLF, LassoLLF, and SparseGroupLLF generally outperform Random Forest (RF) and XGBoost across varying levels of heteroscedasticity and feature dimensionality. A detailed analysis of Table 5 reveals that in low noise settings ($\sigma = 1$), SparseGroupLLF frequently achieves the lowest RMSE in higher-dimensional scenarios (e.g., $d = 20, 40$), while the standard LLF often performs best at lower dimensionality ($d = 5$). This underscores the strength of group structured regularization in extracting relevant features under mild stochastic volatility. at intermediate noise levels ($\sigma = 5$), both LassoLLF and SparseGroupLLF frequently demonstrate superior performance compared to the standard LLF, indicating that penalized variants offer enhanced robustness against moderate noise while preserving relevant structure. In high noise scenarios ($\sigma = 10$), the predictive performance across all three LLF variants remains highly competitive, though no single penalized model shows a clear advantage over the standard LLF. This convergence suggests that in severely noisy environments, the benefits of regularization may plateau, and that the core LLF structure itself is sufficiently robust. Regarding individual model performances, the Local Linear Forests (LLF, LassoLLF, and SparseGroupLLF) consistently demonstrate superior predictive accuracy. LLF tends to perform best in environments characterized by lower dimensionality and noise levels, while its penalized variants-LassoLLF and SparseGroupLLF-excel in scenarios

with higher values of d and σ. This highlights the effectiveness of their regularization techniques in filtering out noise and selecting relevant features, making them especially suitable for more complex data environments. BART also exhibits high robustness in environments characterized by varying noise levels and high feature dimensionality, with results often comparable to the strongest LLF variants, and even yielding the lowest RMSE in certain high noise, high-dimensional configurations (e.g., $n = 500$, $d = 20$, $\sigma = 10$). On the other hand, Random Forest (RF) and XGBoost generally underperform compared to the LLF family and BART, as reflected in their consistently higher RMSE values. This suggests that while RF and XGBoost are powerful general purpose machine learning algorithms, they may struggle to capture the structure of EGARCH driven volatility in this simulated context.

n	d	σ	BART	LLF	LassoLLF	RF	SparseGroupLLF	XGB
250	5	1	1.1448	**1.0788**	1.1032	1.7463	1.1032	1.5210
250	20	1	1.3339	1.2298	1.1832	1.9129	**1.1496**	1.6904
250	40	1	1.2639	1.3800	1.1880	2.2424	**1.0984**	1.5649
500	5	1	1.1072	**1.0425**	1.1402	1.5073	1.1402	1.2838
500	20	1	1.1729	1.1415	1.1604	1.7127	**1.0617**	1.3589
500	40	1	1.1431	1.1014	**1.0698**	1.9340	1.0374	1.4533
1000	5	1	1.0786	**1.0434**	1.0529	1.3976	1.0529	1.2886
1000	20	1	1.0722	1.0401	1.0496	1.5940	**1.0233**	1.3293
1000	40	1	1.1281	1.1335	1.0914	1.6817	**1.0564**	1.3933
250	5	5	5.2523	5.1475	5.1952	5.1887	5.1952	5.9019
250	20	5	5.5681	5.5301	5.5519	5.6015	**4.8586**	5.9059
250	40	5	5.6407	**5.5880**	5.6626	5.6342	5.8307	6.0873
500	5	5	4.8799	4.8746	**4.8692**	4.9026	4.8692	5.5009
500	20	5	5.3820	5.2788	5.3120	5.3609	**5.2307**	5.8101
500	40	5	5.2035	5.2121	**5.1522**	5.3126	5.2307	5.6058
1000	5	5	5.0827	5.0662	**5.0619**	5.1506	5.0619	5.6441
1000	20	5	4.9772	4.9805	**4.9404**	5.0907	4.9499	5.4048
1000	40	5	5.5002	5.4555	**5.3904**	5.5713	4.9772	5.8873
250	5	10	10.3650	10.1107	10.2583	10.1783	**10.2592**	11.8109
250	20	10	9.9682	**9.7184**	10.0833	9.7662	10.1239	10.6192
250	40	10	10.2931	9.9643	10.2389	9.9072	10.7172	**10.4347**
500	5	10	10.8726	10.6963	10.7053	**10.6528**	10.6972	11.9871
500	20	10	**8.9606**	9.1058	9.3386	9.2134	9.2101	9.4632
500	40	10	11.1056	**10.9678**	11.0628	11.0071	10.9840	11.4874
1000	5	10	10.3344	**10.2866**	10.3216	10.3054	10.3216	11.5662
1000	20	10	10.1252	10.0703	**10.1372**	10.1085	10.1831	11.1684
1000	40	10	10.36699	10.2080	10.2886	10.2336	**10.4707**	11.0760

Table 5: Average out-of-sample RMSE per model for different combinations of sample size n, dimension d, and volatility factor σ.

3 Regime Analysis

In this section, we seek to assess and compare the performance of a suite of volatility-forecasting models, including GARCH, HAR-RV, Local Linear Forests (LLF and its Lasso- and Sparse-Group-penalized variants), Random Forest (RF), and XGBoost across different market regimes in the Bitcoin market.

3.1 Regime Definition

We will split our data into three regimes: Bear, Sideways, and Bull, following the approach frequently employed in previous literature, such as in Agakishiev et al. [2025], and proven effective in Cortese et al. [2023]. The regime classification is determined based on the relationship between the 50-day and 200-day moving averages (MA) of the closing price. Specifically, if the 50-day MA is more than 1% higher than the 200-day MA, we classify the period as a bull regime; if it is more than 1% lower, we classify it as a bear regime. Periods where the 50-day MA falls within ±1% of the 200-day MA are classified as sideways

regimes. This methodology is inspired by Brock et al. [1992], who demonstrated that MA50/MA200 "golden" and "death" crosses serve as reliable regime indicators.

3.2 Evaluation Methods

For the quantitative evaluation of our models, we employ two benchmark metrics, RMSE and QLIKE. To statistically assess volatility predictability across market phases and validate observed differences, we apply nonparametric tests. This approach is preferred as it does not require assumptions about the underlying distribution of forecast errors, which are often non-normal and may contain outliers in financial applications. First, the Kruskal-Wallis test [Kruskal and Wallis, 1952] is utilized to determine whether significant differences in forecasting performance (based on squared errors) exist between the three distinct regime groups. Second, the Jonckheere-Terpstra test [Lunneborg, 2005] serves as an extension, specifically designed for situations where groups possess a logical order (e.g., Bull < Bear < Sideways). This test assesses for a monotonic trend in performance across these ordered regimes. Finally, the Conover-Iman post-hoc procedure [Conover and Iman, 1981], with Holm-adjusted p-values, is employed to identify which specific regime pairs exhibit statistically significant differences. While the Kruskal–Wallis and Jonckheere–Terpstra tests assume independent observations, they are commonly used in Timeseries applications [Lelwala et al., 2024, Kuryłek, 2023, Iuchi and Hamada, 2021] even when some degree of autocorrelation is present.

3.3 Results

Table 6 presents the performance of seven volatility forecast models across three different market phases: Bear, Sideways, and Bull. The models are evaluated using two metrics: RMSE and QLIKE. From the results, we can conclude that traditional models like GARCH perform significantly worse, regardless of the regime. This confirms the earlier results shown in table1 and 2. We observe that the LassoLLF and SparseGroupLLF models frequently outperform other models. Specifically, SparseGroupLLF achieves the best performance in the bull market (lowest RMSE and QLIKE), while LassoLLF demonstrates superior performance in the sideways market (lowest RMSE and QLIKE). In the bear market, LassoLLF shows better performance in terms of RMSE, whereas SparseGroupLLF achieves a lower QLIKE. A general trend among most models (excluding GARCH and XGBoost) is that they perform best in the bull market, followed by the bear market, and perform worst in the sideways market. Notably, XGBoost exhibits slightly better RMSE in the bear market than in the bull market. While LLF based models consistently perform well under all conditions, models such as Random Forest and XGBoost generally show comparatively weaker performance, particularly when the market trend is flat (sideways regime). This suggests that despite their flexibility, these models may still struggle with optimal model fitting when the underlying data exhibits minimal trend or contains significant noise.

Table 6: Forecast error metrics (RMSE and QLIKE) across different market regimes: Bear, Sideways, and Bull for various volatility forecasting models in the Bitcoin market (BTC)

Model	Bear		Sideways		Bull	
	RMSE	QLIKE	RMSE	QLIKE	RMSE	QLIKE
GARCH	0.521	7.840	0.733	7.282	0.555	8.731
HAR-RV	0.313	0.145	0.395	0.093	0.273	0.125
LLF	0.174	0.039	0.251	0.035	0.157	0.040
LassoLLF	**0.163**	0.035	**0.231**	**0.031**	0.153	0.034
RF	0.255	0.099	0.340	0.075	0.233	0.082
SparseGroupLLF	0.164	**0.031**	0.241	0.033	**0.150**	**0.033**
XGBoost	0.183	0.041	0.298	0.047	0.189	0.045

Table 7: Kruskal-Wallis-test (χ^2, p_{KW}) en Jonckheere-Terpstra-test (J, p_{JT}) per model

Model	χ^2_{KW}	p_KW	J_{JT}	p_JT
LLF	11.3380	0.0035	181 698	0.0063
LassoLLF	16.2960	0.0003	185 224	0.0010
SparseGroupLLF	23.3160	0.0000	182 257	0.0058
RF	20.2790	0.0000	186 641	0.0004
GARCH	40.1310	0.0000	180 194	0.0167
HAR-RV	7.8220	0.0200	175 969	0.0934
XGBoost	10.8430	0.0044	177 248	0.0588

To statistically validate the observed differences in model performance across regimes, we apply two nonparametric omnibus tests: the Kruskal-Wallis test and the Jonckheere-Terpstra test. Table 7 reports the test statistics (χ^2_{KW} for Kruskal-Wallis, J_{JT} for Jonckheere-Terpstra) and their corresponding p-values (p_{KW}, p_{JT}) for the squared forecast errors for each forecasting model. The Kruskal-Wallis test examines whether the distribution of squared forecast errors differs significantly between the three market regimes. For all models except HAR-RV, the p-values are well below the 5% significance threshold, indicating strong evidence that forecasting performance varies by regime. Particularly notable is the result for the SparseGroupLLF model ($\chi^2 = 23.316$, $p < 0.001$), suggesting a pronounced sensitivity to regime dynamics, consistent with its superior performance in Table 6.

On the other hand, the Jonckheere-Terpstra test evaluates whether there is a monotonic trend in forecasting errors across the ordered regimes (Bull < Bear < Sideways). Contrary to expectations, several models now do show in Table 7 statistically significant monotonic patterns, including LLF ($p = 0.0063$), LassoLLF ($p = 0.0010$), SparseGroupLLF ($p = 0.0058$), and RF ($p = 0.0004$). This suggests that not only do performance differences exist across regimes, but for some models, these differences follow a consistent directional pattern.

Table 8: Conover–Iman post-hoc p-values (Holm-corrected) for each model and regime pair

Model	Bear vs Sideways	Bear vs Bull	Sideways vs Bull
LLF	0.008	0.992	0.004
LassoLLF	0.001	0.879	0.000
SparseGroupLLF	0.000	0.180	0.000
RF	0.000	0.991	0.000
GARCH	0.000	0.002	0.000
HAR-RV	0.019	0.356	0.046
XGBoost	0.004	0.272	0.015

To further investigate which specific pairs of regimes contribute to the significant differences identified by the Kruskal-Wallis test, we apply the Conover-Iman post-hoc test with Holm-adjusted p-values. The results, reported in Table 8, show that most models exhibit statistically significant differences between Sideways and Bull or Sideways and Bear regimes. For instance, LassoLLF, SparseGroupLLF, and RF display strong significance in the Sideways-related comparisons ($p < 0.001$), but not necessarily between Bull and Bear markets.

the traditional GARCH model also demonstrates consistent significance across all pairs, including Bull vs Bear ($p = 0.002$), despite its lower forecasting accuracy overall. This supports the idea that GARCH is sensitive to structural changes, but less effective at adapting to them. In contrast, LLF-based models like LassoLLF and SparseGroupLLF show meaningful distinctions particularly when the market is in or out of a Sideways regime, confirming their robustness to varying market dynamics.

4 Conclusion

This study set out to determine whether the SparseGroupLLF variant delivers better volatility forecasts than The LassoLLF variant. The empirical crypto exercise, the EGARCH Monte Carlo simulation and the BTC regime analysis all point in the same direction: both penalised Local Linear Forests dominate

GARCH, HAR-RV, Random Forest and XGBoost by a wide margin, yet neither decisively outruns the other across every metric and setting. SparseGroupLLF produces the lowest QLIKE for every coin and achieves the best risk centred scores in bull markets, underscoring its strength when relative error and tail risk matter most. LassoLLF, in turn, often secures the lowest RMSE particularly for BTC, XLM and SHIB and proves more resilient than SparseGroupLLF when the market drifts sideways.Nemenyi post-hoc comparisons confirm that the rank difference between the two is rarely significant at conventional levels, so any superiority is context-dependent rather than absolute. In practical terms, choosing between the two hinges on the loss function: if minimising QLIKE (and hence penalising volatility underprediction) is paramount, SparseGroupLLF offers a small but persistent edge; if absolute error is the priority, LassoLLF remains the safer bet. Either way, both variants represent a substantial upgrade over traditional econometric and generic machine learning baselines, and their shared dominance highlights the value of combining local linear corrections with targeted regularisation.

References

Ilyas Agakishiev, Wolfgang Karl Härdle, Denis Becker, and Xiaorui Zuo. Regime switching forecasting for cryptocurrencies. *Digital Finance*, pages 1–25, 2025.

William Brock, Josef Lakonishok, and Blake LeBaron. Simple technical trading rules and the stochastic properties of stock returns. *The Journal of finance*, 47(5):1731–1764, 1992.

William J Conover and Ronald L Iman. Rank transformations as a bridge between parametric and nonparametric statistics. *The American Statistician*, 35(3):124–129, 1981.

Federico P Cortese, Petter N Kolm, and Erik Lindström. What drives cryptocurrency returns? a sparse statistical jump model approach. *Digital Finance*, 5(3):483–518, 2023.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30, 2006.

Timothy O Hodson. Root mean square error (rmse) or mean absolute error (mae): When to use them or not. *Geoscientific Model Development Discussions*, 2022:1–10, 2022.

Hitoshi Iuchi and Michiaki Hamada. Jonckheere–terpstra–kendall-based non-parametric analysis of temporal differential gene expression. *NAR Genomics and Bioinformatics*, 3(1):lqab021, 2021.

William H Kruskal and W Allen Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621, 1952.

Wojciech Kuryłek. The modeling of earnings per share of polish companies for the post-financial crisis periodusing random walk and arima models. *Journal of Banking and Financial Economics*, 19(1): 26–43, 2023.

EI Lelwala, WM Seamasinghe, and KMLM Gunarathna. Nonparametric approach to detecting seasonality in time series: Application of the kruskal-wallis (kw) test on tourist arrivals to sri lanka. *South Asian Journal of Business Insights*, 4(1), 2024.

Clifford E Lunneborg. J onckheere–terpstra test. *Encyclopedia of statistics in behavioral science*, 2005.

Christopher Z Mooney. *Monte carlo simulation*. Number 116. Sage, 1997.

Andrew J Patton. Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics*, 160(1):246–256, 2011.