International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

(Research Article)

Check for updates

Hybrid ensembles for Robust UK inflation forecasting: An empirical evaluation with high-frequency covariates and structural-break adaptation

Tewogbade Shakir *

CAPE Economic Research and Consulting, U. S.

International Journal of Science and Research Archive, 2025, 16(01), 2228-2257

Publication history: Received on 10 May 2025; revised on 05 July 2025; accepted on 08 July 2025

Article DOI: https://doi.org/10.30574/ijsra.2025.16.1.1879

Abstract

This research develops and empirically evaluates a hybrid ensemble framework for forecasting UK inflation, specifically, the CPI, CPIH and RPI indices, over the turbulent March 2022–March 2025 period. The research went further on traditional time-series methods (SARIMA, VAR), machine-learning algorithms (Random Forest, XGBoost), deep-learning architectures (LSTM, CNN+LSTM) and the forecasting library Prophet by constructing an ensemble that combines individual model forecasts via a linear meta-learner. Models are trained on monthly ONS data, with rolling–window validation reserving the final 12 months for out-of-sample testing.

Individual SARIMA and VAR produced high forecast errors : SARIMA RMSE: CPI 3.94, R^2 –17.52; VAR RMSE: CPI 2.69, R^2 –7.66, confirming their limitations in the face of non-linearity issue and regime-shifting dynamics. Random Forest (CPI RMSE 2.37, R^2 –5.69) and XGBoost (CPI RMSE 2.05, R^2 –4.03) improved accuracy but yielded negative R^2 values, indicating poor generalization. LSTM further reduced error with CPI RMSE 2.05 and R^2 of 3.99, while the hybrid CNN+LSTM achieved positive fit for CPI RMSE 0.54 and R^2 0.65; CPIH RMSE 0.72, R^2 0.64; RPI RMSE 2.18, R^2 0.47, demonstrating the benefit of combining convolutional feature extraction with sequence learning.

The main contribution is the ensemble of XGBoost, LSTM, Ridge Regression and Prophet, which delivers near-zero bias and high explanatory power across all indices: CPI RMSE 0.18 (R^2 0.96), CPIH RMSE 0.16 (R^2 0.98) and RPI RMSE 0.55 (R^2 0.97). This represents over 90% error reduction compared to SARIMA and nearly 70% improvement over CNN+LSTM, demonstrating the value of model diversity and stacking in volatile macroeconomic environments. The ensemble's robustness strongly capture short-term seasonality, long-term trends and non-linear interactions.

Keywords: SARIMA; VAR; LSTM; Prophet; Ensemble

1. Introduction

Inflation forecast is a core area in macro-financial governance, influencing monetary and fiscal policy, corporate investment decisions, and consumer expenditure (Al-Shamery and Al-Gashamy, 2018; Tsuchnig, Ferner and Gadermayr, 2018). Every month the projection of how fast or how slowly prices will rise feeds directly into decisions on interest-rate settings, index-linked welfare benefits, wage bargaining rounds and firms' price-setting strategies. Thus, their accuracy and credibility are essential for effective governance, as they impact long-term financial planning and economic stability. In the United Kingdom the Monetary Policy Committee (MPC) of the Bank of England (BoE) publishes modal and fan-chart projections that anchor market expectations. Yet the BoE evaluations show that its near-term CPI inflation forecast errors rose sharply after 2021, with persistent under-estimation of the pass-through from wage growth to prices and of the speed at which supply-chain shocks unwind (Kanngiesser and Willems, 2024).

* Corresponding author: Tewogbade Shakir

From the work of Box and Jenkins (1970), inflation predictors in central banks have leaned heavily on Autoregressive Integrated Moving Average (ARIMA) families and Vector Autoregression (VAR). These formulations assume a stable data-generating process, with linear propagation of shocks; they excel when patterns are smooth, and history is long (Erkekoglu, Garang and Deng, 2020). Yet UK inflation in the 2020s has displayed successive structural breaks which was evident from the collapse and rebound of energy prices, the war-induced food-price spike, and administrative tariff adjustments to Ofgem's energy-price cap (Dorling 2024; Lu, Coutts and Gudgin, 2024; Oxford Analytica, 2022). Based on these occurrences, BoE now stresses that non-linear determinants of services inflation have become more salient and has begun experimenting with boosted-tree models to distil those drivers (Mann, 2023)

Research by Xavier, Fernandes and De Oliveira (2023) and Lauderdale et al. (2020) demonstrates two complementary strategies for stepping beyond linearity. First, tree-based ML models such as Random Forests (RF) and gradient-boosted decision trees (XGBoost) partition the covariate space recursively, approximating complex interactions without assuming functional form. Xavier, Fernandes and De Oliveira (2023) affirms that an RF-XGBoost ensemble reduced the root-mean-square error (RMSE) of Brazilian inflation forecasts by 25 % relative to an AR benchmark. Second, sequence-learning DL models like Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN) fed by lagged price sequences learn long-range dependencies and localized pattern. Considering this, Binner, Kelly and Tepper (2025) show shallow recurrent neural networks (RNNs) capture UK CPI turning points that stump standard Phillips-curve regressions.

1.1. Problem Statement

Between 2022 and 2025 UK headline CPI surged to 11 % before retreating below 4 %, while services-sector inflation remained stubbornly high (Carella et al., 2024). Traditional SARIMA specifications, calibrated to pre-pandemic dynamics, failed to anticipate either the peak or the pace of disinflation. VAR systems struggled with parameter instability and multi-collinearity once supply shocks distorted relative prices. At the same time, single-algorithm ML and DL models showed promise but produced volatile forecasts when economic regimes shifted, and they lacked the interpretive transparency required by central-bank staff and Treasury officials.

Consequently, UK decision-makers face a forecasting trilemma: models that are simple and transparent but inaccurate, flexible and accurate but opaque, or ensemble schemata whose relative contributions are unclear. The absence of a holistic comparison spanning these dimensions strongly hinder evidence-based refinement of the BoE's forecast platform and limits the private sector's ability to hedge inflation risk effectively.

1.1.1. Main Research Question

How can hybrid ensemble models integrating traditional time-series, machine-learning and deep-learning techniques improve the accuracy, interpretability and resilience of UK inflation forecasts during the volatile 2022–2025 period?

2. Literature review

When considering exploration of understanding of modern UK inflation crisis, it is important to contextualise current trends against historical developments. The UK's inflationary history is marked by periods of both hyperinflation and deflation, shaped by post-war reconstruction, oil shocks, financial deregulation, and global market shifts (Oxford Analytica, 2023a; Hendry, 2001).

In the immediate post-World War II period (1945–1970), UK witnessed moderate inflation accompanied by strong economic growth and low unemployment. This was known as the "Golden Age of Capitalism," fuelled by Keynesian demand management policies. However, from the early 1970s, inflation spiralled due to two major oil shocks and union-driven wage growth. Inflation peaked at over 26.9% in 1975 (Needham, 2014; Rogers, 2013), which led to widespread economic instability, industrial strikes, and a crisis of policy legitimacy.

The 1970s stagflation period was a clear turning point. Conventional Keynesian policies were abandoned in favour of monetarist strategies introduced under the Thatcher government in the 1980s as mentioned earlier in section (2.2). Tight monetary control and deregulation became the dominant policy narrative. While these measures succeeded in bringing inflation down to under 5% by the late 1980s, they also led to high unemployment, social unrest, and long-term shifts in the UK's labour market (Minford, 2021; Coutts and Gudgin, 2016).

The 1990s and early 2000s marked a period of relative price stability which was due to the institutionalisation of inflation targeting and the granting of operational independence to the Bank of England in 1997. Inflation remained close to the 2% target for most of this period (Baldwin, 2013; Delia, Horat and Cornel, 2010). However, this apparent stability masked growing inequality, particularly in housing, wages, and public service access.

The 2008 Global Financial Crisis triggered a new inflationary environment. Despite a brief period of deflationary pressure, stimulus policies including quantitative easing and low interest rates gradually caused rising asset prices in UK (Jin, 2024). By the late 2010s, inflationary pressures were again mounting, especially in housing and utilities, although headline CPI remained modest.

The most recent and highly significant shift occurred in the wake of the COVID-19 pandemic. Between 2021 and 2023, UK inflation rose sharply (Chen, 2023), hitting 9.1% in May 2022, one of the highest in over 40 years. Key drivers driving the figure included supply chain disruptions, labour shortages, increased public spending, and the war in Ukraine, which spiked energy and food prices. The rise in inflation in UK was further exacerbated by Brexit-related friction, particularly in trade and immigration-dependent labour markets (Song, 2024; Zhu, 2024).

This historical trajectory demonstrates that inflation in the UK has been shaped not just by domestic demand or monetary policy, but by external shocks, policy missteps, and structural weaknesses in the economy. In this regard, modern inflation must be understood as a complex and multi-causal phenomenon that resists basic and simplistic modelling.

2.1. Inflation Indices

Capturing inflation accurately and effectively is essential for understanding real-income dynamics, yet different indices carry its own biases. The Consumer Price Index (CPI) measures the cost change of a fixed basket of goods and services weighted by average household expenditure (Dasu and Mohana, 2023; Costales, 2021). By excluding owner-occupiers' housing costs (OOH), CPI often understate the inflationary burden on homeowners when mortgage rates rise sharply. To address this, the CPIH index incorporates OOH through rental equivalence approach, estimating the imputed rent owner-occupiers would pay on the open market (ONS, 2023). While CPIH better reflects total household inflation, its reliance on rental survey data sometime lags actual market pressures, smoothing out sudden spikes in owner costs.

Monetary authorities often focus on core CPI, which strips out volatile food and energy components to reveal underlying trends (European Central Bank, 2021). Core measures aid in disentangling transitory shocks like oil-price spikes from persistent inflationary forces. However, for low-income households, food and energy constitute larger budget shares, so core CPI understate immediate cost-of-living pressures they face.

The Retail Price Index (RPI), although deprecated for official purposes in the UK, still influences many wage and lease contracts. RPI differs from CPI by including mortgage interest payments and council tax, but its use of arithmetic rather than geometric averages introduces an "indexation bias" that tends to overstate inflation when price changes vary across goods (ONS, 2018). Evidently, RPI's sensitivity to interest-rate fluctuations makes it volatile and less comparable over time.

Sector-specific indices provide further granularity. The energy price index rose by over 50% year-on-year at its 2022 peak following the Ukraine invasion, reflecting wholesale shocks and supply-chain disruptions (Ofgem, 2023). As energy is non-discretionary, these surges inflict disproportionate welfare losses on vulnerable households (BRC, 2023). Also, the food price index experienced a 13.8% increase in early 2023, the highest on record, driven by global commodity shortages and logistical constraints (BRC, 2023). These sectoral spikes are often masked in aggregate CPI yet are critical for understanding distributional impacts

Methodological discourse around fixed versus chained baskets expose further complexities: fixed-basket indices fail to account for consumer substitution in response to relative price shifts, potentially overstating welfare losses (Hanousek and Filer, 2005), whereas chain-linked measures improve realism at the expense of historical consistency. Consequently, rigorous cost-of-living analysis in the UK context demands triangulating across CPI, CPIH, core CPI, RPI, and key sectoral trackers, all while remaining mindful of each index's construction and limitations.

2.2. Related Research

Many research has been conducted in the domain of inflation, inflation prediction and utilization of sophisticated computational methods. Some of the similar works were reviewed as presented in Table (1)

Table 1 Summary of Related Research

Author /Year	Research Goal	Dataset	Models	Results	Findings	Weaknesses/Limitations/Challenges
Naghi, O'Neill & Zaharieva (2024)	Evaluate ML techniques in forecasting CPI inflation across US, Canada, UK, including post-COVID period.	FRED-MD (US), Canada (114 vars), UK (110 vars); 1960–2022 (US), 1981–2022 (Canada), 1998–2021 (UK).	RF, XGBoost, LightGBM, CatBoost, SVR, Elastic Net, BART, NNs, Forecast Combinations, ARIMA, UCSV.	Best RMSE ≈ 0.74 (US, Quantile ERT); RF underperforms post-2020.	Quantile trees and forecast combinations perform best; key predictors: inflation lags, rates, housing.	Model performance varies across countries/periods; RF not best post-COVID; short sub-periods.
Simionescu (2025)	Test if ARDL with sentiment index beats SARIMA and ML models for Romanian CPI forecasts.	Romanian CPI (2006–2024), unemployment rate, sentiment index from central bank reports.	ARDL, SARIMA, RF, SVR, ANN, ARDL + RF hybrid.	ARDL RMSE = 1.92; SARIMA & ML models RMSE > 2.0; ARDL had 100% directional accuracy.	Sentiment-enriched ARDL outperforms SARIMA and ML; captures policy tone.	Short forecast horizon; limited ML tuning; Romania-specific.
Özgür & Akkoç (2021)	Compare shrinkage ML and traditional models for Turkish CPI under high inflation.	229 Turkish macro series (2007–2019), 2017–2019 for evaluation.	Lasso, Elastic Net, Ridge, Group-Lasso, ARIMA, ARMA-GARCH, VAR, Prophet.	Lasso RMSE = 0.83; Elastic Net = 0.89; ARIMA = 1.00; RW = 1.24.	Lasso & Elastic Net outperform all traditional models; FX and credit growth as key signals.	Focused on linear ML; no boosting/deep models; λ sensitivity.
Nason & Smith (2021)	Analyze inflation waves, predictability, and real interest rates in UK (1251–2019).	Clark's UK retail price index (1251–2019); interest rates from Thomas & Dimsdale (2017).	TVP-SV-AR(5), Kalman Filter, MCMC Bayesian estimation.	R^2 = 0.44 at 1-year; volatility peaks in 1550, 1800, 1921, 1975.	20th century inflation was most volatile; inflation is predictable over 1–5 years.	Univariate focus; noisy historical data; lack of multivariate models.
Barkan et al. (2022)	Use HRNN to improve short-term CPI inflation forecasts using CPI hierarchy.	US CPI-U data from BLS (Jan 1994–Mar 2019); 424 series in 8-level hierarchy.	HRNN, ARIMA, ETS, baseline RNN.	Up to 42.5% RMSE improvement vs ARIMA on fine-grained series.	HRNN captures inter-series dependencies; gains highest at low-level components.	Increased model complexity; data format issues; deeper levels harder to predict.
Aras & Lisboa (2022)	Compare tree-based ML vs. linear models for Turkish CPI forecasting.	Monthly Turkish CPI and macro data (2005–2020).	OLS, Lasso, Ridge, Elastic Net, SVR, RF, XGBoost.	XGBoost showed ~6–9% RMSE improvement over linear benchmarks.	Tree models (RF, XGBoost) outperform linear models on accuracy.	RMSE gains modest; no DL used; overfitting risk not fully addressed.

Latypov et al. (2024)	Assess bottom-up ML-based approach to forecasting Russian CPI using CPI components.	Russian CPI (80 components, 2002–2023); filtered 6,000+ series to 109 exogenous variables.	RW, Seasonal MA, Holt-Winters, (T)BATS, SARIMA(X), Prophet, Elastic Net, ARD, RF, CatBoost.	Bottom-up improved accuracy up to 1.5x; CatBoost best RMSE overall.	Bottom-up yields better predictions; volatile components like food dominate error.	No deep learning; long-term trend modelling weak; error from omitted variables (e.g., weather).

Review of related research showed some gaps. Simionescu (2025) and Aras & Lisboa (2022) adopt a narrow modelling scope, failing to implement or assess deep learning architectures, which have stronger capabilities to produce robust models. Similarly, despite promising results with hybrid and ensemble techniques, prior research generally underutilizes hybrid frameworks that combine econometric, machine learning, and deep learning models. Fourth, econometric models like ARDL and SARIMA, though prevalent in works like Simionescu (2025), often lack comparative benchmarking against more advanced data-driven methods under volatile economic conditions, a key target of the current research. Also, Özgür & Akkoç (2021) focus predominantly on shrinkage techniques and regularization but ignore high-capacity learners like deep recurrent networks. While Nason & Smith (2021) offer long historical perspectives, their work lacks high dimensional validation or accuracy assessment, which the current research intends to prioritize through rolling forecasts from 2022 to 2025. Although Barkan et al. (2022) highlight hierarchical dependencies in inflation data, their implementation remains confined to a single model class (HRNN), whereas the current work compares across different form of advanced models to validate performance. Lastly, Naghi et al. (2024) demonstrate strong performance post-COVID but do not explore structural breaks or regime shifts, a key factor that the current research embeds through hybrid adaptability.

3. Data and Exploratory Analysis

In this section, data collection and details of the data collected for inflation forecasting is discussed. Also, exploratory data analysis is presented to further show structural shift in.

3.1. Data Collection

The dataset for this research was collected from ONS website Consumer price inflation tables - Office for National Statistics. The information in the dataset is based on their April 2025 release. The data presents monthly UK inflation indices from March to October 2022, covering CPIH, CPI, and RPI, along with their respective 12-month percentage changes (Figure 1).

	CPIH		CPI		RPI [3] NOT accredited official statistics	
	Index (2015=100)	Percentage change over 12 months	Index (2015=100)	Percentage change over 12 months	Index (Jan 13, 1987=100)	Percentage change over 12 months
	L522	L55O	D7BT	D7G7	CHAW	CZBH
Mar 2022	116.5	6.2	117.1	7.0	323.5	9.0
Apr	119.0	7.8	120.0	9.0	334.6	11.1
May	119.7	7.9	120.8	9.1	337.1	11.7
Jun	120.5	8.2	121.8	9.4	340.0	11.8
Jul	121.2	8.8	122.5	10.1	343.2	12.3
Aug	121.8	8.6	123.1	9.9	345.2	12.3
Sep	122.3	8.8	123.8	10.1	347.6	12.6
Oct	124.3	9.6	126.2	11.1	356.2	14.2

Figure 1 ONS 2025 Inflation Dataset

CPIH stands for Consumer Prices Index including Housing costs and is the UK's most comprehensive inflation measure as it includes owner occupiers' housing costs and Council Tax. This variable also offers detailed description of household inflation and is considered the lead indicator by the Office for National Statistics. CPI, or Consumer Prices Index, is the official inflation target of the Bank of England and reflects the average price change of a fixed basket of goods and services, but it excludes housing-related costs like mortgage interest. While CPI is used for monetary policy decisions, it does not fully capture the cost pressures faced by homeowners. RPI, or Retail Prices Index, is an older inflation measure that includes mortgage interest and housing costs but uses a flawed methodology, making it no longer a National Statistic. However, RPI is still used in practice for uprating rail fares, student loans, and some pension and wage contracts.

3.2. Exploratory Analysis

3.2.1. Summary Statistics

The first analysis that was implemented is checking the summary statistic of the data. This was done using the describe () method on the cleaned DataFrame to generate summary statistics for each column, including the mean, standard deviation, min, max, and quartiles. This function was applied after pre-processing to assess the distribution and variability of the CPIH, CPI, and RPI indices, both in index form and year-over-year (YoY) percentage changes (Figure 2).

```
# Generate summary statistics
summary_stats = df_cleaned.describe()

# Display the result
print(summary_stats)


       CPIH_Index    CPIH_YoY   CPI_Index     CPI_YoY   RPI_Index     RPI_YoY
count   36.000000   36.000000   36.000000   36.000000   36.000000   36.000000
mean   128.983333    5.850000  130.355556    6.036111  372.883333    7.902778
std      4.903381    2.498743    4.674762    3.424379   17.698208    4.290753
min    119.000000    2.600000  120.000000    1.700000  334.600000    2.700000
25%    125.175000    3.500000  127.075000    2.600000  360.375000    3.575000
50%    130.000000    5.500000  131.600000    5.650000  377.550000    7.500000
75%    132.925000    8.300000  133.950000    9.525000  387.350000   11.925000
max    136.100000    9.600000  136.500000   11.100000  395.300000   14.200000
```

Figure 2 CPI, CPIH and RPI summary statistics

From Figure (2), the mean CPIH index is approximately 128.98, with a standard deviation of 4.90, indicating moderate variation in monthly values. The CPI_YoY shows a mean of 6.56%, with values ranging from 1.7% to 11.1%, reflecting periods of both moderate and elevated inflation. The RPI_Index has the highest absolute values, with a maximum of 395.3 and a mean of 372.83, consistent with its historical base of January 1987=100. The RPI_YoY has a wider range (2.5% to 14.2%) and the highest average (7.9%) among the YoY metrics, suggesting that RPI inflation fluctuated more sharply during the period.

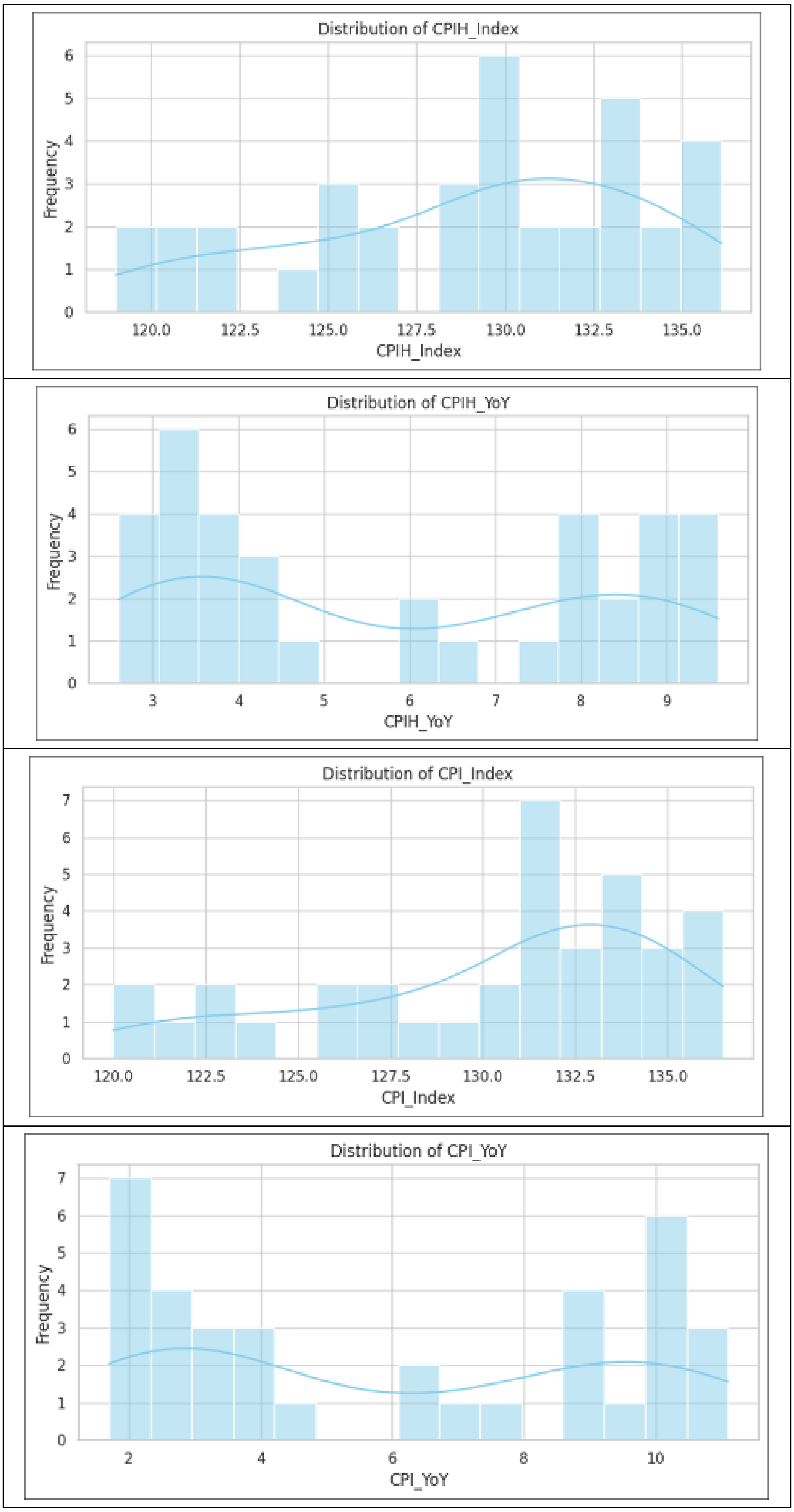
3.2.2. Inflation Distribution

Similarly, distribution of CPI, CPIH and RPI were implemented using histogram with syntax displayed in Figure (3.). For each of the variable, the distribution plots a histogram with 15 bins that is overlaid by a KDE (Kernel Density Estimate).

```
# Set the plot style
sns.set(style="whitegrid")

# Plot histogram for each variable
for column in df_cleaned.columns:
    plt.figure(figsize=(8, 4))
    sns.histplot(df_cleaned[column], kde=True, bins=15, color='skyblue')
    plt.title(f'Distribution of {column}')
    plt.xlabel(column)
    plt.ylabel('Frequency')
    plt.tight_layout()
    plt.grid(True)
    plt.show()
```

Figure 3 Python Syntax to plot histogram

The six distribution curves produced from the python syntax in Figure (3) are presented in Figure (4) for CPIH_Index, CPIH_YoY, CPI_Index, CPI_YoY, RPI_Index, and RPI_YoY.



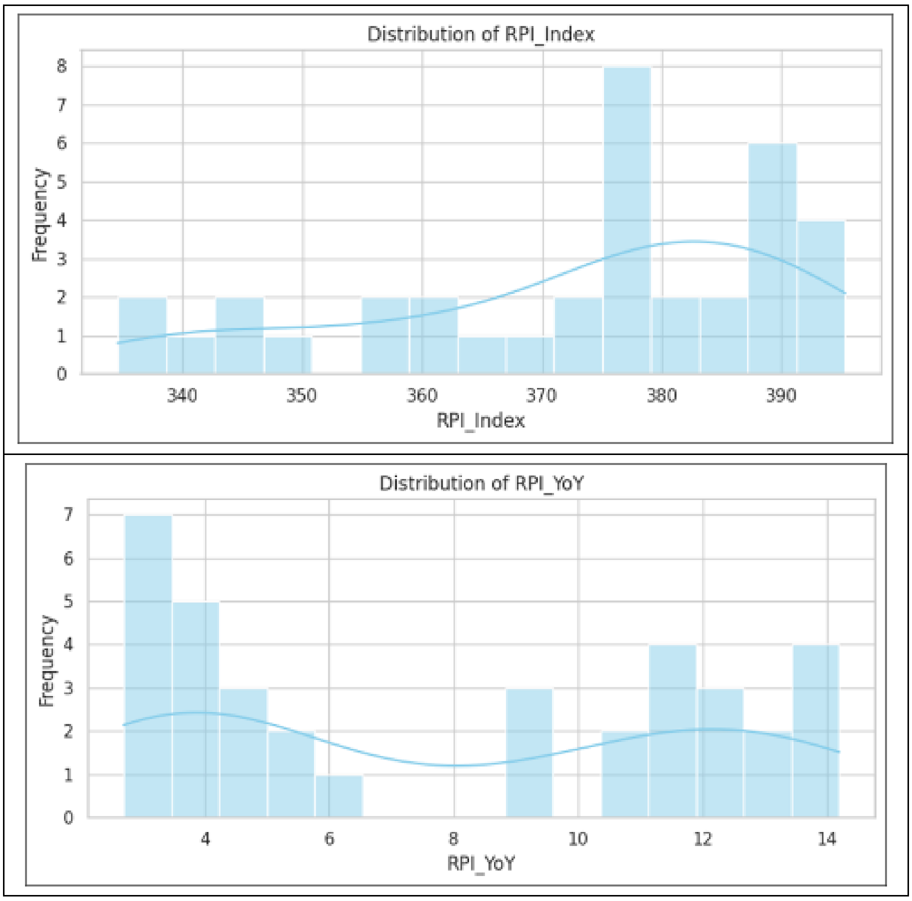


Figure 4 Distribution Plots for CPIH Index, CPIH_YoY, CPI_Index, CPI_YoY, RPI_Index, and RPI_YoY

As displayed in Figure (4), the CPIH_Index distribution appears moderately symmetric with a concentration between 129 and 134. This suggests a stable price movement with slight upward bias. CPIH_YoY, however, displays bimodality, with peaks around 3–4% and another near 9%, indicating inflationary episodes with dual regime effects over time. The CPI_Index distribution is similar to CPIH_Index, and this confirms the consistency in price levels observed across both measures. CPI_YoY plot shows a broader and more irregular shape with high variation and peaks around 2–3% and 9–10%. This signals sharp fluctuations and likely structural breaks. The RPI_Index is right-skewed, indicating most values are concentrated between 370 and 390, with a few higher values pushing the tail to the right. RPI_YoY plot shows a bimodal trend, with distinct peaks near 4% and above 12%, likely reflecting inflation volatility driven by energy price shocks or policy shifts. The presence of multiple peaks (bimodality) in CPIH_YoY and RPI_YoY signals distinct economic regimes like high-inflation and moderate-inflation periods. All the distributions of the inflation variables in Figure (4) clearly provide strong evidence that inflation in the UK over 2022–2025 has not been uniformly distributed but characterized by substantial structural shifts. The spread and kurtosis in CPI_YoY and RPI_YoY suggest periods of economic turbulence that require flexible and strong non-linear modelling approaches. These findings further validate the implementation of advanced machine learning models which is the aim of this research.

3.2.3. Trend Analysis

Trend plots were implemented for the inflation variables based on python syntax in figure (5).

```
# Plot CPIH, CPI, and RPI Index Trends
plt.figure(figsize=(12, 6))
plt.plot(df_cleaned.index, df_cleaned['CPIH_Index'], label='CPIH Index')
plt.plot(df_cleaned.index, df_cleaned['CPI_Index'], label='CPI Index')
plt.plot(df_cleaned.index, df_cleaned['RPI_Index'], label='RPI Index')
plt.title('Trend of CPIH, CPI, and RPI Indices Over Time')
plt.xlabel('Date')
plt.ylabel('Index Value')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Figure 5 Python Syntax for Trend analysis of CPIH, CPI and RPI Index

The trend from the plot is shown in Figure (6)

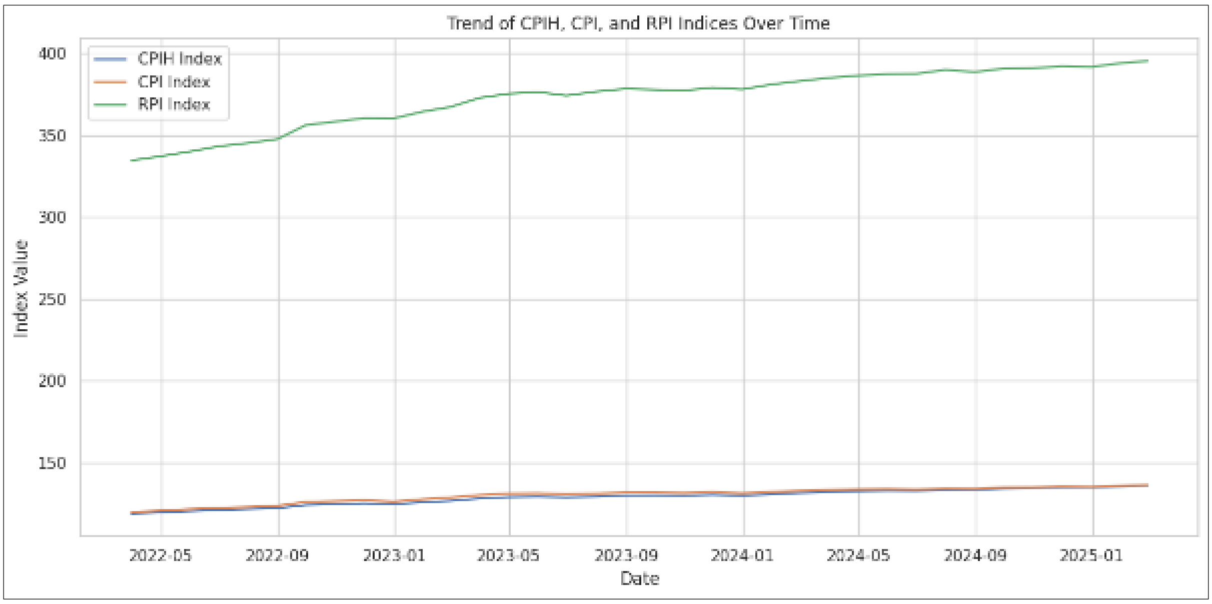


Figure 6 CPI, CPIH and RPI Trends

From figure (6), the trend plot shows that all three indices, CPIH, CPI, and RPI increased between early 2022 and early 2025. This means that there is a sustained inflationary pressure over the period. The Retail Price Index (RPI) consistently remains significantly higher than both the Consumer Prices Index (CPI) and Consumer Prices Index including owner occupiers' housing costs (CPIH), reflecting its inclusion of mortgage interest payments and other factors that amplify its values. Both CPI and CPIH follow a similar path with a steady and moderate increase, but RPI not only starts higher but also grows at a faster rate, suggesting that it may overstate inflation relative to the other measures.

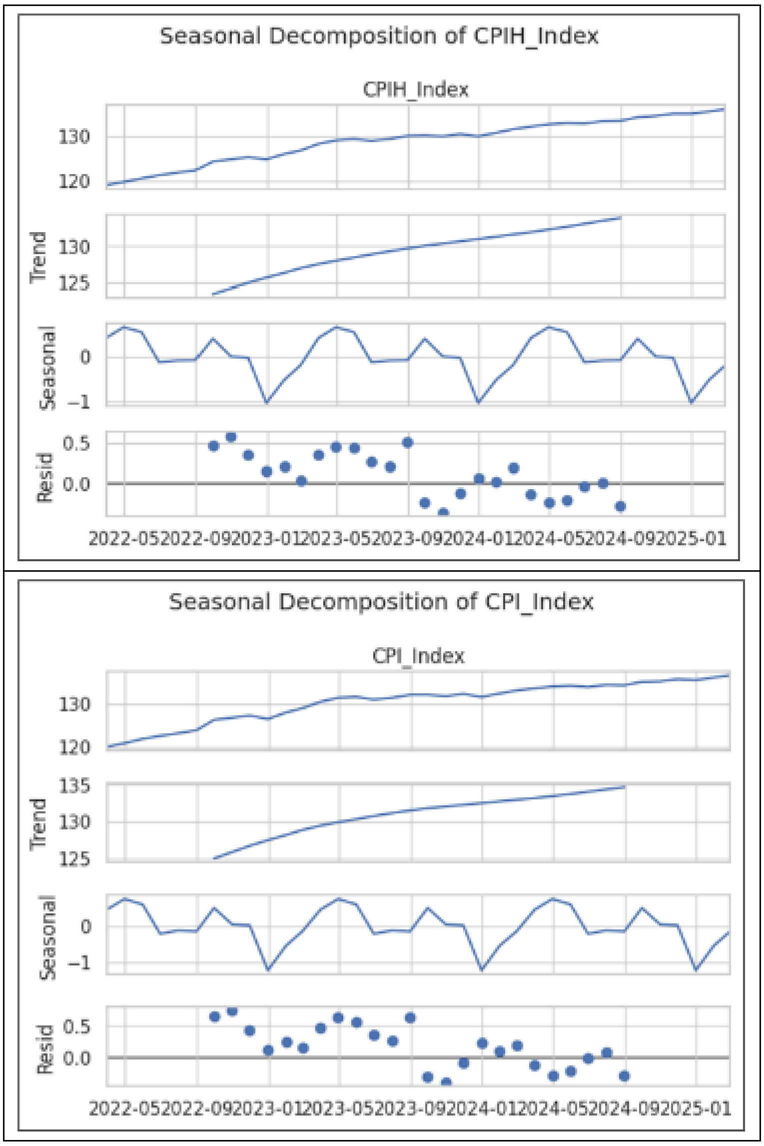
3.2.4. Decomposition Analysis

Seasonal decomposition was executed to separate and gain deeper understanding of the underlying components of the time series data for CPIH, CPI, and RPI indices. Separation of these components provide valuable insights into how much of the index's movement is due to systematic patterns against irregular volatility. This is very important to achieve better modelling for the variables. To implement the decomposition, the inflation data is first coerced into a monthly frequency to ensure proper temporal alignment, and an additive decomposition model is specified, assuming that the components combine linearly (Figure 8). The syntax iterates over each index, applies the decomposition for a period of 12 (indicating monthly seasonality), and produce separate plots showing how the seasonal, trend, and residual components evolve over time for each index.

```
from statsmodels.tsa.seasonal import seasonal_decompose
# Ensure the index has a monthly frequency
df_cleaned = df_cleaned.asfreq('M')

# Choose decomposition model type: 'additive' or 'multiplicative'
model_type = 'additive'

# Loop through each index and perform decomposition
for col in ['CPIH_Index', 'CPI_Index', 'RPI_Index']:
    print(f"\nSeasonal Decomposition for {col} ({model_type} model):")
    decomposition = seasonal_decompose(df_cleaned[col], model=model_type, period=12)

    # Plot the decomposition results
    decomposition.plot()
    plt.suptitle(f'Seasonal Decomposition of {col}', fontsize=14)
    plt.tight_layout()
    plt.show()
```

Figure 7 CPI, CPIH and RPI Decomposition Syntax

The results of the decompositions are presented in Figure 8.



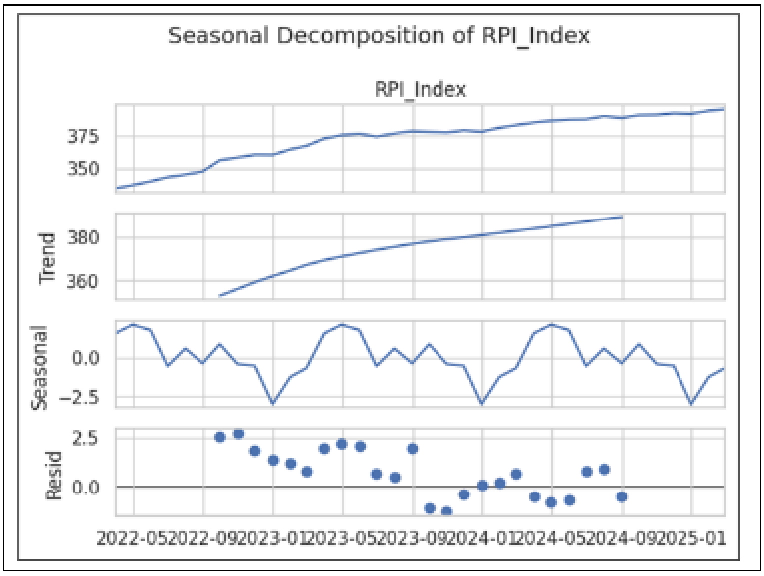


Figure 8 CPI, CPIH and RPI Decomposition Plots

From figure (8), the seasonal decomposition of CPIH, CPI, and RPI indices reveals strong upward trends across all three. This signifies consistent inflation from 2022 to early 2025. Both CPIH and CPI show nearly identical patterns in their trends and seasonality, reflecting similar calculation methods with minor differences due to housing costs. RPI, however, shows a steeper trend and higher index values overall, which aligns with its inclusion of mortgage interest payments and other legacy components. The seasonal components of all indices display regular, cyclical patterns, with RPI exhibiting the largest seasonal variation, peaking and dipping more sharply than CPI and CPIH. This suggests that prices captured by RPI are more sensitive to seasonal economic factors like energy and housing. The residual components for the three indices are centred around zero but vary in magnitude, with RPI again showing the most volatility. Evidently, decomposition confirms the structural inflationary pressure while highlighting differences in seasonal responsiveness and noise among the indices.

4. Model Implementation

Based on the research objectives, seven models were implemented including the specially developed ensemble model with the target of improving model performance. The models include SARIMA, VAR, Random Forest, XGBoost, LSTM, CNN+LSTM and Hybrid Ensemble.

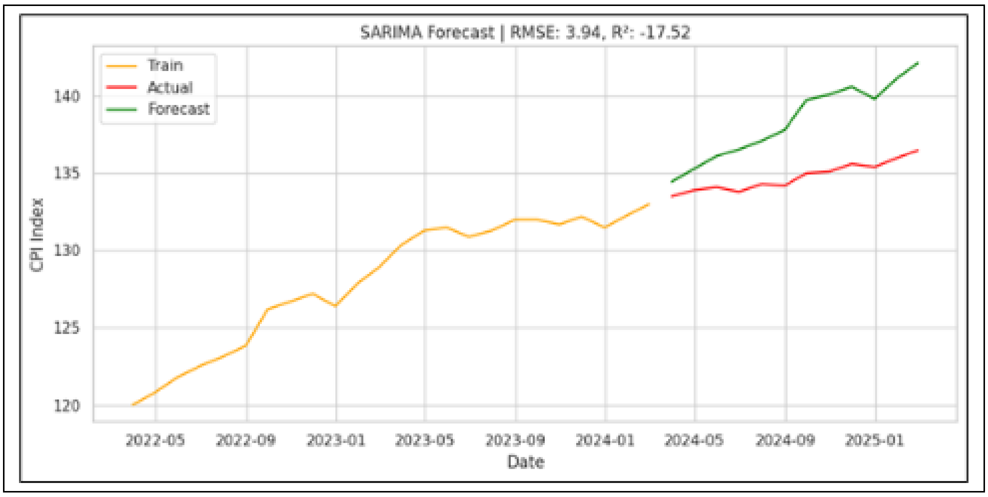
4.1. SARIMA Model

A SARIMA (1,1,1)(1,1,1,12) model is fitted to the training data to capture both seasonal and non-seasonal components of the time series. Forecasts are generated for 12 future time steps and compared to the actual test data using RMSE and R^2 as performance metrics (Figure 9). Same implementation was carried out for CPIH and RPI Indices.

```
from statsmodels.tsa.statespace.sarimax import SARIMAX
from sklearn.metrics import mean_squared_error, r2_score


# Define target variable
target_series = df_cleaned['CPI_Index']

# Train-test split (last 12 months for testing)
train_sarima = target_series[:-12]
test_sarima = target_series[-12:]

# Fit SARIMA model (seasonal_order can be tuned further)
model_sarima = SARIMAX(train_sarima, order=(1, 1, 1), seasonal_order=(1, 1, 1, 12))
model_fit_sarima = model_sarima.fit(disp=False)

# Forecast
forecast_sarima = model_fit_sarima.forecast(steps=12)
forecast_sarima.index = test_sarima.index

# Evaluation metrics
rmse_sarima = np.sqrt(mean_squared_error(test_sarima, forecast_sarima))
r2_sarima = r2_score(test_sarima, forecast_sarima)

# Plot actual vs forecast
plt.figure(figsize=(10, 5))
plt.plot(train_sarima.index, train_sarima, label='Train', color='orange')
plt.plot(test_sarima.index, test_sarima, label='Actual', color='red')
plt.plot(forecast_sarima.index, forecast_sarima, label='Forecast', color='green')
plt.title(f'SARIMA Forecast | RMSE: {rmse_sarima:.2f}, R²: {r2_sarima:.2f}')
plt.xlabel('Date')
plt.ylabel('CPI Index')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# Print metrics
print(f"RMSE: {rmse_sarima:.2f}")
print(f"R² Score: {r2_sarima:.2f}")
```

Figure 9 Implementation of SARIMA Model

The results of SARIMA model is presented in Figure 10



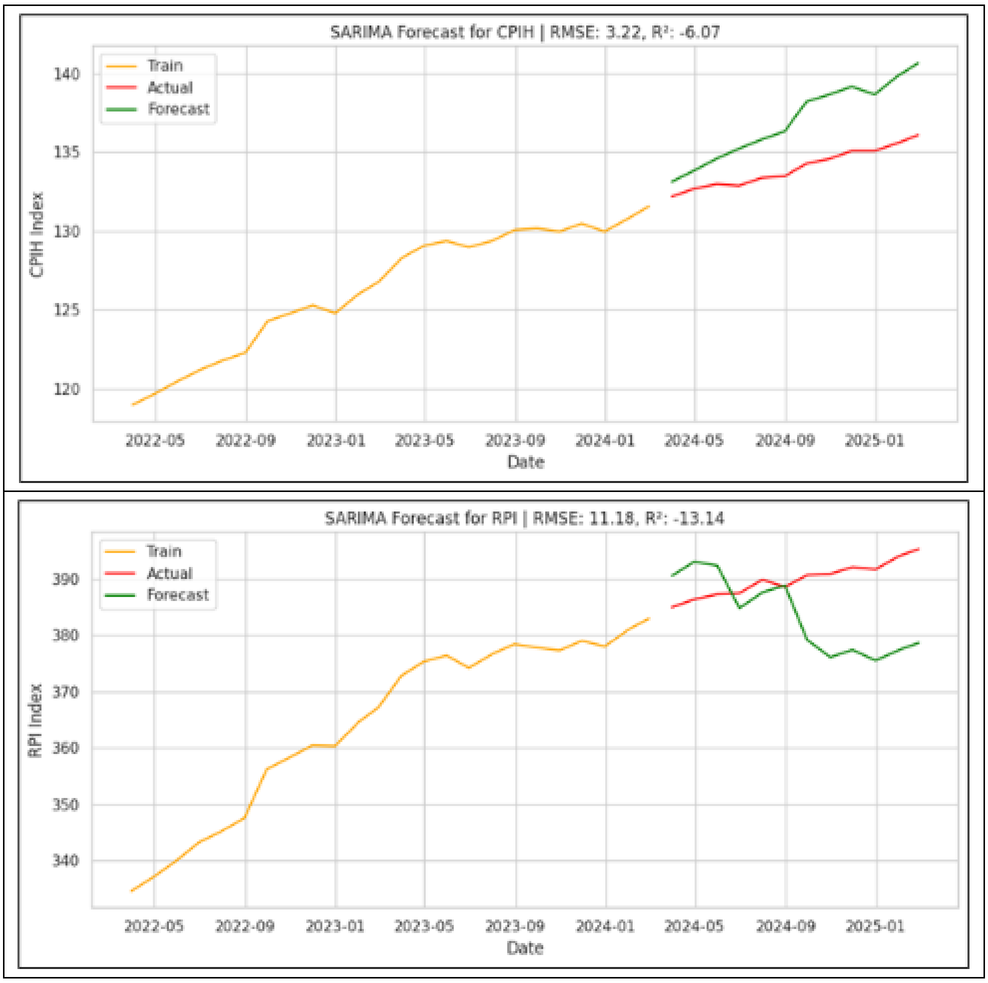


Figure 10 SARIMA Model Forecast for CPI, CPIH and RPI Indices

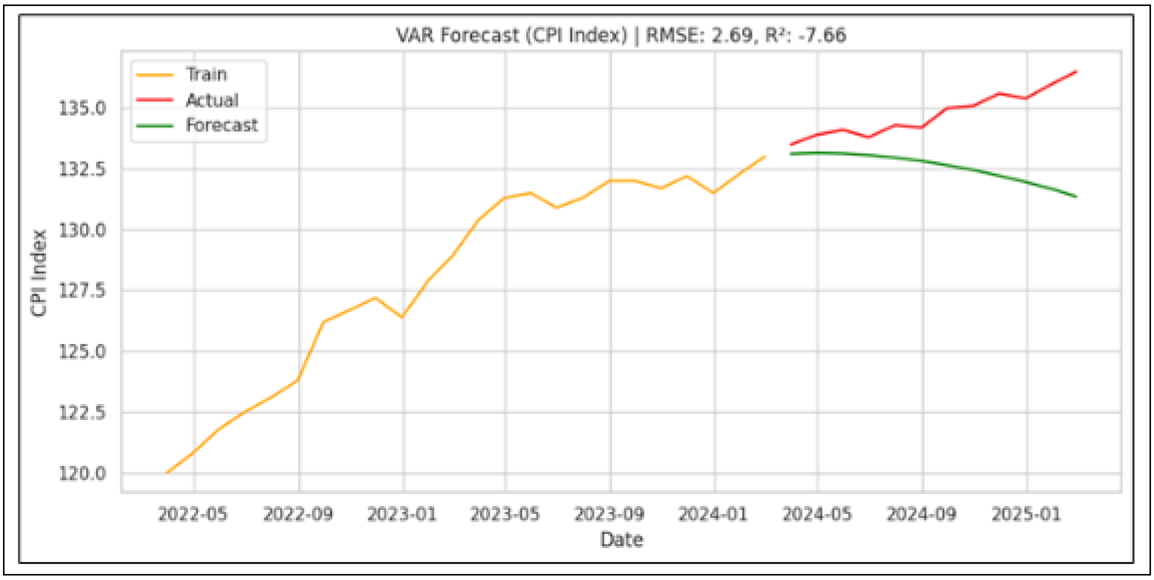
4.2. VAR Model

The Vector Autoregression (VAR) model was set up to forecast CPI_Index using multivariate time series data. The optimal lag order is selected using the Akaike Information Criterion (AIC), and the model is fitted to the training set before forecasting 12 future steps (Figure 11). Same process was done for CPIH and RPI.

```python
from statsmodels.tsa.api import VAR
from sklearn.metrics import mean_squared_error, r2_score


# Select multivariate time series data
df_var = df_cleaned[['CPIH_Index', 'CPI_Index', 'RPI_Index']].dropna()

# Train-test split (last 12 months for testing)
train_var = df_var[:-12]
test_var = df_var[-12:]

# Fit VAR model (using lag order selected by AIC)
model_var = VAR(train_var)
results_var = model_var.fit(maxlags=4, ic='aic')

# Forecast
lag_order = results_var.k_ar
forecast_input = train_var.values[-lag_order:]
forecast_var = results_var.forecast(y=forecast_input, steps=12)

# Convert forecast to DataFrame with appropriate index
forecast_df_var = pd.DataFrame(forecast_var, index=test_var.index, columns=test_var.columns)

# Evaluation metrics on CPI_Index
rmse_var = np.sqrt(mean_squared_error(test_var['CPI_Index'], forecast_df_var['CPI_Index']))
r2_var = r2_score(test_var['CPI_Index'], forecast_df_var['CPI_Index'])

# Plot actual vs forecast for CPI_Index
plt.figure(figsize=(10, 5))
plt.plot(train_var.index, train_var['CPI_Index'], label='Train', color='orange')
plt.plot(test_var.index, test_var['CPI_Index'], label='Actual', color='red')
plt.plot(forecast_df_var.index, forecast_df_var['CPI_Index'], label='Forecast', color='green')
plt.title(f'VAR Forecast (CPI Index) | RMSE: {rmse_var:.2f}, R²: {r2_var:.2f}')
plt.xlabel('Date')
plt.ylabel('CPI Index')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# Print performance metrics
print(f"RMSE: {rmse_var:.2f}")
print(f"R² Score: {r2_var:.2f}")
```

Figure 11 Implementation of VAR Model

The results for the implemented VAR model is presented in Figure 12



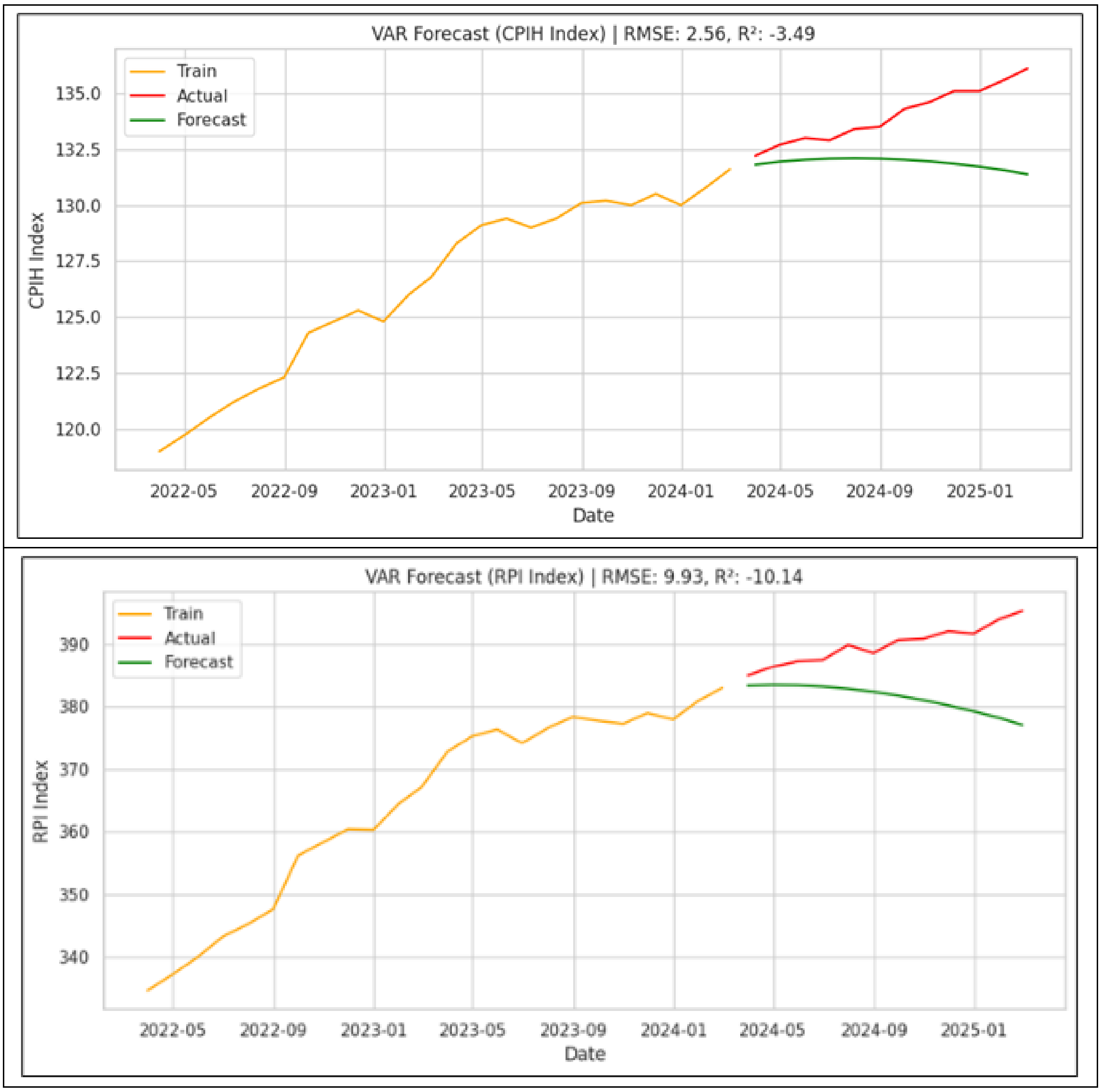


Figure 12 VAR Model Forecast for CPI, CPIH and RPI Indices

4.3. Random Forest Model

Random Forest Regressor was set up to forecast the CPI index by first specifying lagged features from the CPI_Index time series, with 12 prior months used as predictors (Figure 13). A parameter grid is defined for tuning key Random Forest hyperparameters (number of estimators, max depth, and min samples), and GridSearchCV is applied for optimal model selection using cross-validation. The best model is used to predict CPI_Index on the test set, and its performance is evaluated and printed using the defined metrics. Same process was carried out for CPIH and RPI indices.

```
# Function to create lagged features
def create_lagged_features(series, lags=12):
    df = pd.DataFrame(series)
    for i in range(1, lags + 1):
        df[f'lag_{i}'] = df[series.name].shift(i)
    df.dropna(inplace=True)
    return df

# Create lagged features for CPI_Index
lagged_rf = create_lagged_features(df_cleaned['CPI_Index'], lags=12)
X_rf = lagged_rf.drop(columns='CPI_Index')
y_rf = lagged_rf['CPI_Index']

# Split into training and testing sets (last 12 for testing)
X_train_rf = X_rf[:-12]
X_test_rf = X_rf[-12:]
y_train_rf = y_rf[:-12]
y_test_rf = y_rf[-12:]

# Define evaluation function
def evaluate_forecast(y_true, y_pred):
    rmse = np.sqrt(mean_squared_error(y_true, y_pred))
    r2 = r2_score(y_true, y_pred)
    return rmse, r2

# Define parameter grid for fine-tuning
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_depth': [3, 5, 10, None],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2]
}

# Grid search
grid_search_rf = GridSearchCV(RandomForestRegressor(random_state=42),
                              param_grid, cv=3,
                              scoring='neg_mean_squared_error', n_jobs=-1)
grid_search_rf.fit(X_train_rf, y_train_rf)

# Best model
best_rf_model = grid_search_rf.best_estimator_

# Forecast
best_rf_forecast = best_rf_model.predict(X_test_rf)

# Evaluate
rmse_rf_tuned, r2_rf_tuned = evaluate_forecast(y_test_rf, best_rf_forecast)
```

Figure 13 Implementation of Random Forest Model

The results of the random forest regressor is presented in Figure 14

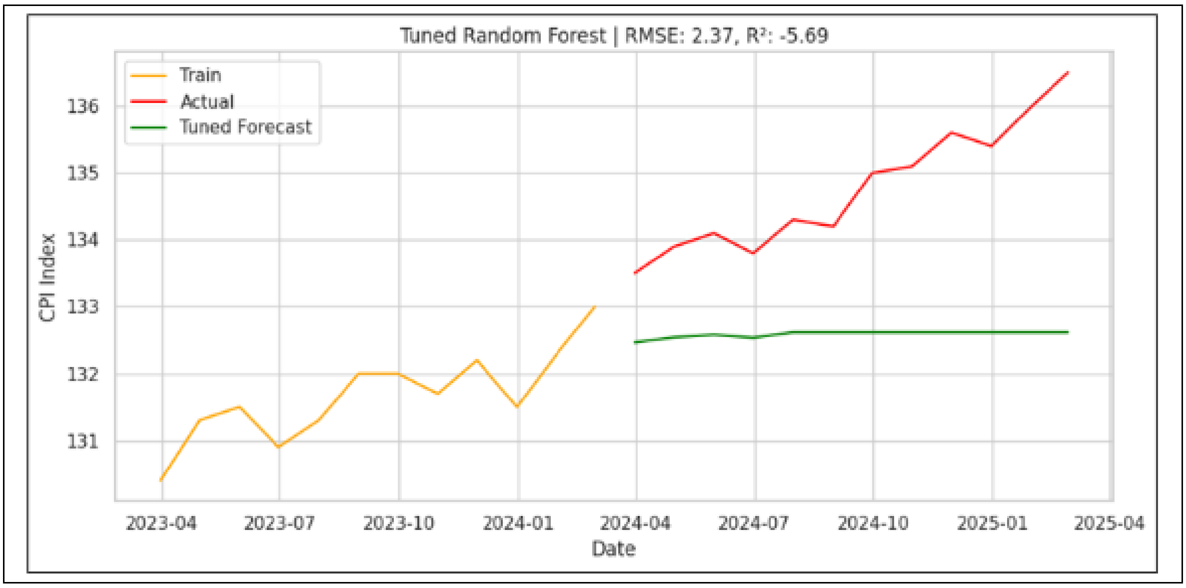


Figure 14 Random Forest Model Forecast for CPI, CPIH and RPI Indices

4.4. XGBoost Model

The XGBoost constructs a time series forecasting model for the CPI index by first generating lagged features from the last 12 months of CPI data to capture temporal dependencies. A hyperparameter grid is defined to tune the number of estimators, tree depth, and learning rate, and GridSearchCV is employed to select the best configuration using cross-validation (Figure 15). Same implementations were carried out for CPIH and RPI.

```python
from xgboost import XGBRegressor
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import mean_squared_error, r2_score


# Define helper functions
def create_lagged_features(series, lags=12):
    df = pd.DataFrame(series)
    for i in range(1, lags + 1):
        df[f'lag_{i}'] = df[series.name].shift(i)
    df.dropna(inplace=True)
    return df

def evaluate_forecast(y_true, y_pred):
    rmse = np.sqrt(mean_squared_error(y_true, y_pred))
    r2 = r2_score(y_true, y_pred)
    return rmse, r2

# Prepare lagged features
lagged_xgb = create_lagged_features(df_cleaned['CPI_Index'], lags=12)
X_xgb = lagged_xgb.drop(columns='CPI_Index')
y_xgb = lagged_xgb['CPI_Index']

# Train-test split
X_train_xgb = X_xgb[:-12]
X_test_xgb = X_xgb[-12:]
y_train_xgb = y_xgb[:-12]
y_test_xgb = y_xgb[-12:]

# Define hyperparameter grid
xgb_param_grid = {
    'n_estimators': [100, 200],
    'max_depth': [3, 5, 7],
    'learning_rate': [0.01, 0.05, 0.1],
    'subsample': [0.8, 1.0]
}

# Run grid search
xgb_grid_search = GridSearchCV(
    XGBRegressor(random_state=42, objective='reg:squarederror'),
    xgb_param_grid, cv=3,
    scoring='neg_mean_squared_error', n_jobs=-1
)
xgb_grid_search.fit(X_train_xgb, y_train_xgb)

# Forecast and evaluate
best_xgb = xgb_grid_search.best_estimator_
xgb_forecast = best_xgb.predict(X_test_xgb)

rmse_xgb, r2_xgb = evaluate_forecast(y_test_xgb, xgb_forecast)
```

Figure 15 Implementation of XGBoost Model

The result of the XGBoost model implementation is presented in Figure 16.

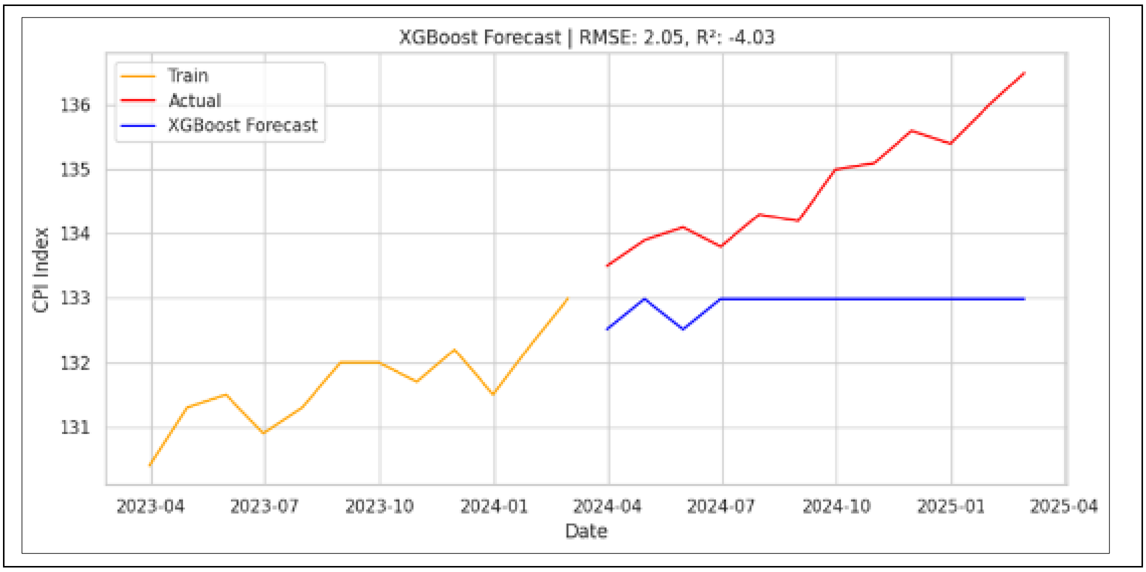


Figure 16 XGBoost Model Forecast for CPI, CPIH and RPI Indices

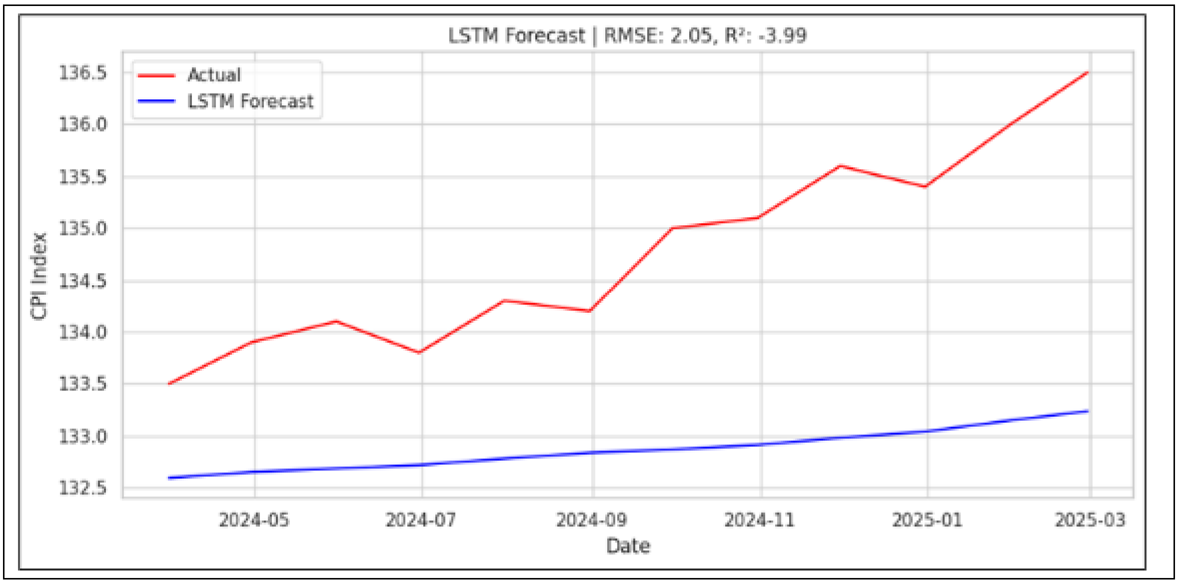
4.5. LSTM Model

The LSTM setup is designed to forecast CPI index values by first generating lagged features for sequence modelling and transforming the data into a supervised format. The model applies MinMax scaling to normalize both the input features and target variable, which is needed for neural network stability and performance. The data is reshaped into 3D format as required by LSTM networks, structured as [samples, time steps, features]. A sequential LSTM model is built with one hidden LSTM layer followed by a dense output layer and trained over 100 epochs using the Adam optimizer and mean squared error loss (Figure 17).

```
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense



# Create lagged dataset
def create_lagged_features(series, lags=12):
    df = pd.DataFrame(series)
    for i in range(1, lags + 1):
        df[f'lag_{i}'] = df[series.name].shift(i)
    df.dropna(inplace=True)
    return df

# Prepare data
lagged_lstm = create_lagged_features(df_cleaned['CPI_Index'], lags=12)
X_lstm = lagged_lstm.drop(columns='CPI_Index').values
y_lstm = lagged_lstm['CPI_Index'].values

# Normalize
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()
X_scaled = scaler_X.fit_transform(X_lstm)
y_scaled = scaler_y.fit_transform(y_lstm.reshape(-1, 1))

# Reshape for LSTM [samples, timesteps, features]
X_scaled = X_scaled.reshape((X_scaled.shape[0], X_scaled.shape[1], 1))

# Split train/test
X_train, X_test = X_scaled[:-12], X_scaled[-12:]
y_train, y_test = y_scaled[:-12], y_scaled[-12:]

# Build LSTM model
model = Sequential()
model.add(LSTM(50, activation='relu', input_shape=(X_train.shape[1], 1)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')

# Train
model.fit(X_train, y_train, epochs=100, verbose=0)

# Forecast
y_pred_scaled = model.predict(X_test)
y_pred = scaler_y.inverse_transform(y_pred_scaled)
y_actual = scaler_y.inverse_transform(y_test)

# Evaluate
rmse_lstm = np.sqrt(mean_squared_error(y_actual, y_pred))
r2_lstm = r2_score(y_actual, y_pred)
```

Figure 17 Implementation of XGBoost Model

The results of the LSTM model are presented in Figure 18.



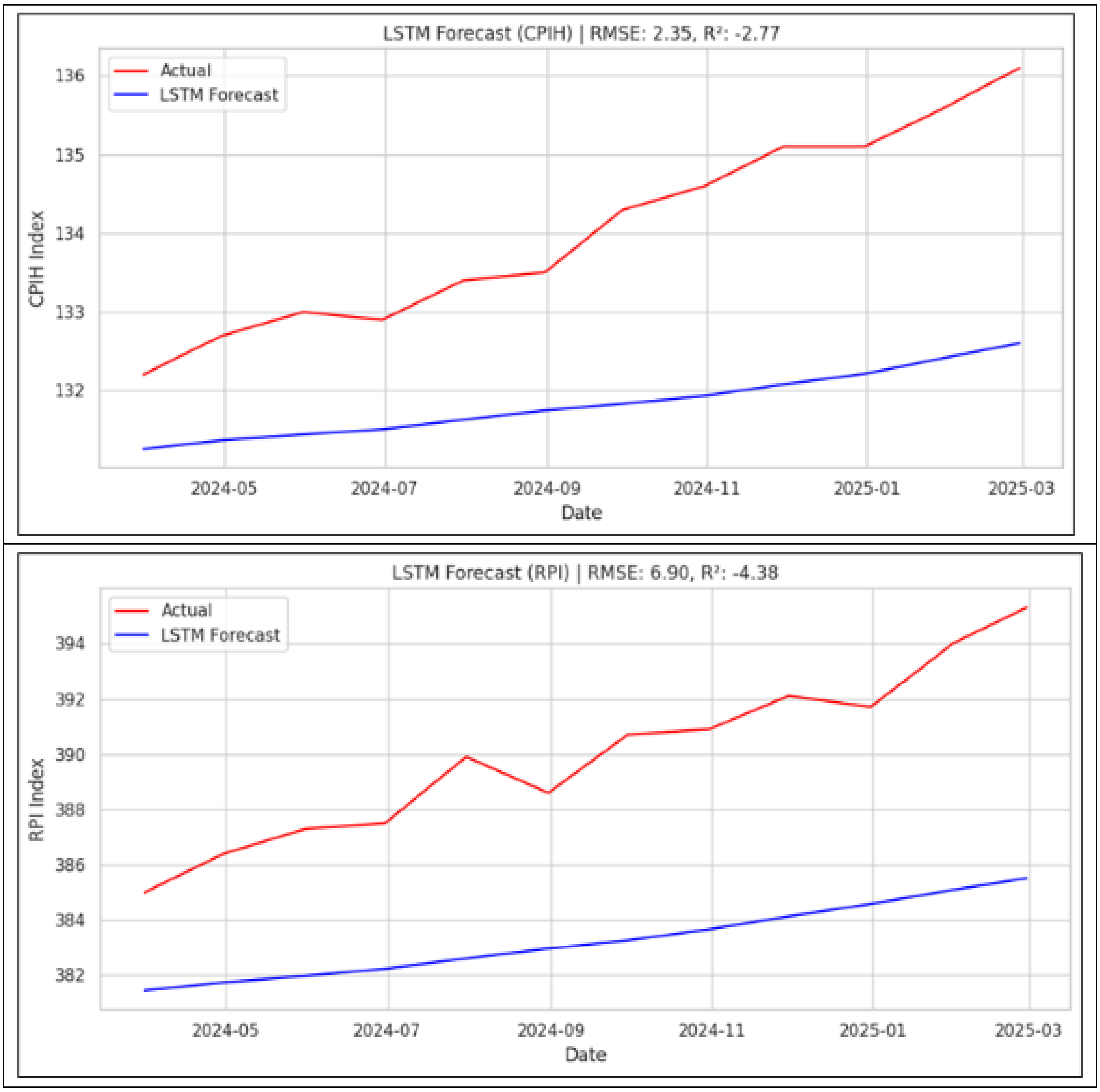


Figure 18 LSTM Model Forecast for CPI, CPIH and RPI Indices

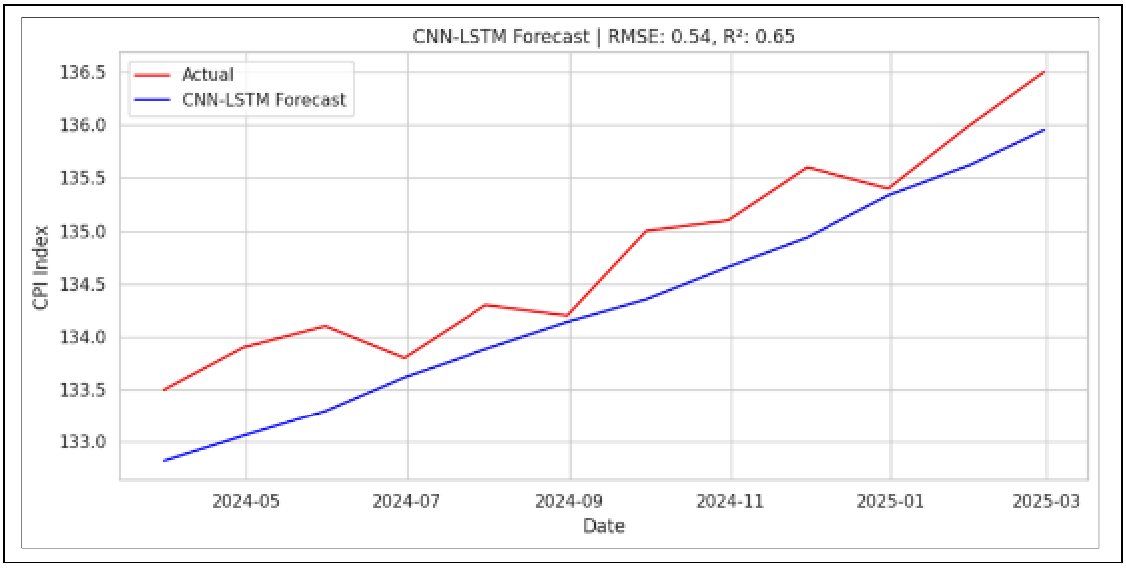
4.6. CNN + LSTM Model

The CNN + LSTM setup builds a hybrid deep learning model to forecast CPI index values by combining convolutional layers for feature extraction and LSTM layers for sequence learning. The CPI index data is reshaped and scaled using MinMaxScaler, then structured into a supervised format with sliding time windows to form [samples, time steps, features]. A sequential model is constructed beginning with a Conv1D layer to capture short-term temporal patterns, followed by a MaxPooling1D, LSTM, and Dense layers to model long-term dependencies and produce the final output. The model is compiled using the Adam optimizer with mean squared error as the loss function and trained for 100 epochs (Figure 19). Same implementations were done for CPIH and RPI.

```python
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, r2_score
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv1D, LSTM, Dense, Flatten, TimeDistributed, MaxPooling1D, Dropout, Input


# Function to create supervised learning format
def create_supervised(data, n_steps):
    X, y = [], []
    for i in range(n_steps, len(data)):
        X.append(data[i-n_steps:i])
        y.append(data[i])
    return np.array(X), np.array(y)

# Prepare CPI_Index values
cpi_values = df_cleaned['CPI_Index'].values.reshape(-1, 1)

# Normalize
scaler = MinMaxScaler()
cpi_scaled = scaler.fit_transform(cpi_values)

# Define input structure
n_steps = 12
X_all, y_all = create_supervised(cpi_scaled, n_steps)

# Reshape for CNN-LSTM: [samples, timesteps, features]
X_all = X_all.reshape((X_all.shape[0], X_all.shape[1], 1))

# Train-test split
X_train, X_test = X_all[:-12], X_all[-12:]
y_train, y_test = y_all[:-12], y_all[-12:]

# Build CNN-LSTM model
model = Sequential([
    Input(shape=(n_steps, 1)),
    Conv1D(filters=64, kernel_size=3, activation='relu'),
    MaxPooling1D(pool_size=2),
    LSTM(50, activation='relu'),
    Dropout(0.2),
    Dense(1)
])

model.compile(optimizer='adam', loss='mse')
model.fit(X_train, y_train, epochs=100, verbose=0)

# Forecast and inverse scale
y_pred_scaled = model.predict(X_test)
y_pred = scaler.inverse_transform(y_pred_scaled)
y_actual = scaler.inverse_transform(y_test)
```

Figure 19 Implementation of CNN+LSTM Model

The result of the implemented CNN+LSTM model is presented in Figure 20



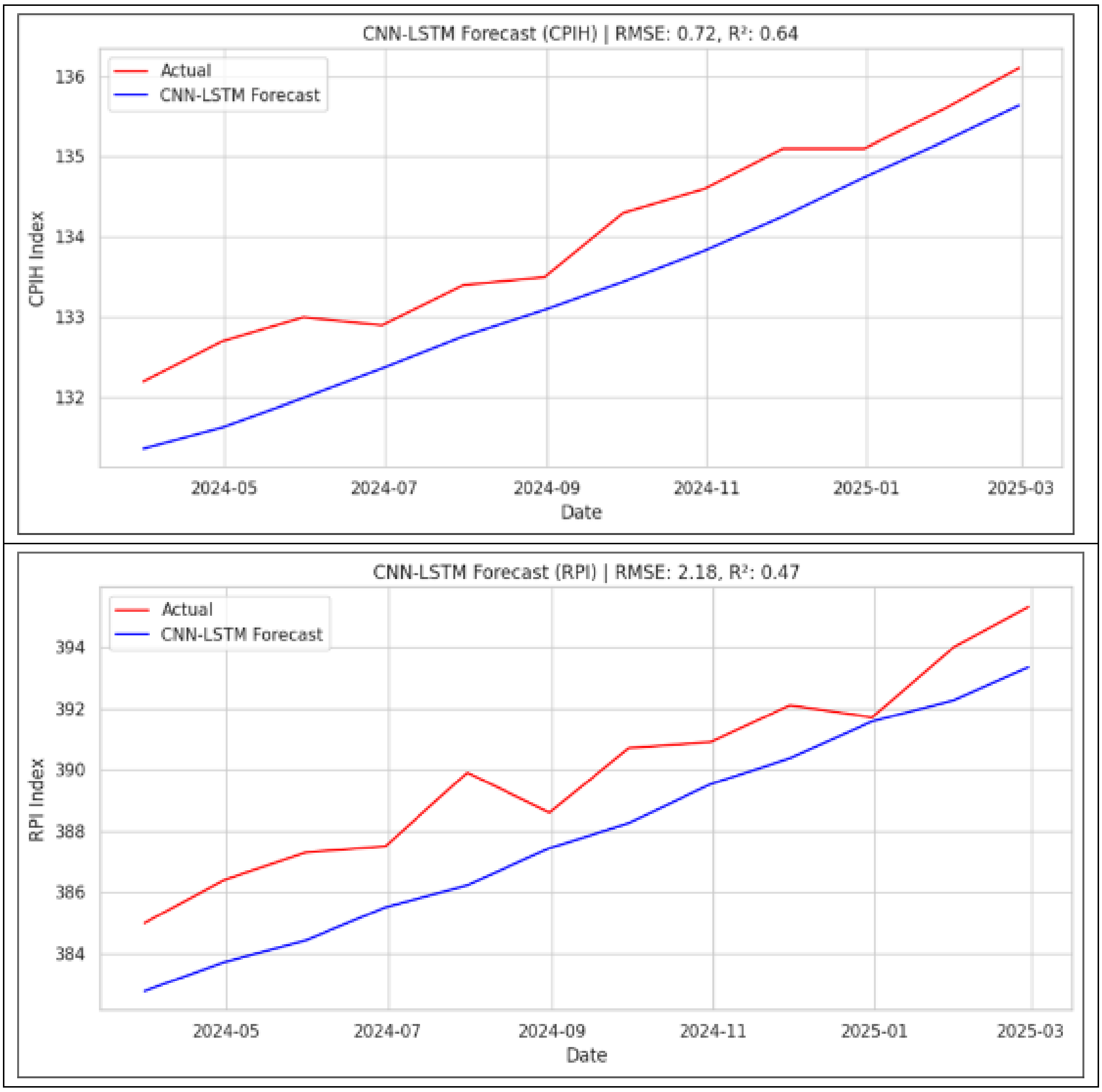


Figure 20 CNN+ LSTM Model Forecast for CPI, CPIH and RPI Indices

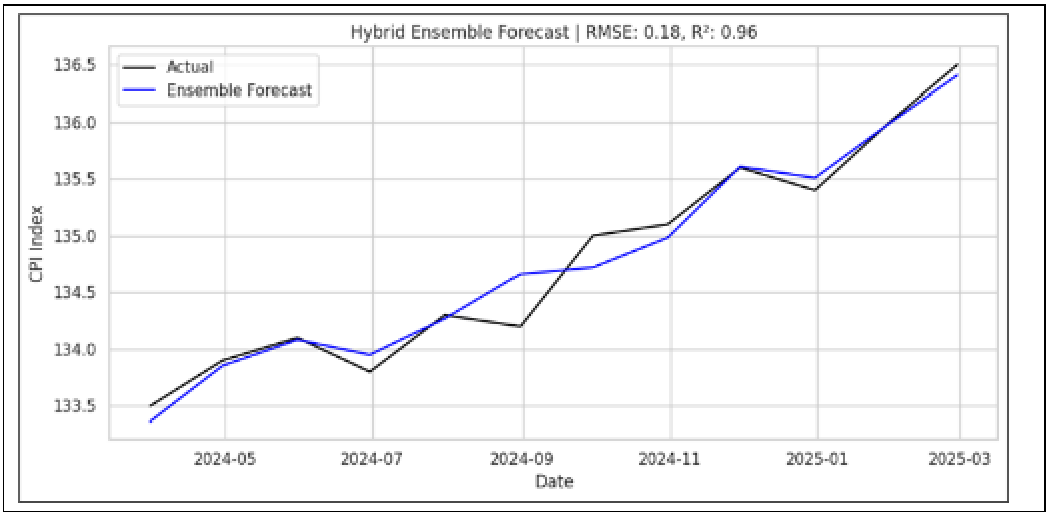
4.7. Ensemble Model: XGBoost + LSTM+ Ridge + Prophet

This ensemble integrates four individual models—Prophet, Ridge Regression, XGBoost, and LSTM to forecast the CPI index. Prophet models the trend using its internal components for time series forecasting, while Ridge Regression uses lagged features to capture linear dependencies in the data. XGBoost improves prediction by learning non-linear relationships from the same lagged inputs as Ridge. The LSTM model is trained on scaled CPI values and makes recursive forecasts by predicting one step at a time using its previous output. After obtaining forecasts from all four models, the predictions are stacked and fed into a meta-learner, which is a simple linear regression model that combines them to form the final ensemble forecast (Figure 21). Same implementation were carried out for CPIH and RPI

```python
# ----------------------------
# 1. Prophet Model
# ----------------------------
prophet_df = train.reset_index().rename(columns={'Date': 'ds', 'CPI_Index': 'y'})
prophet_model = Prophet()
prophet_model.fit(prophet_df)
future = prophet_model.make_future_dataframe(periods=n_forecast, freq='M')
forecast_prophet = prophet_model.predict(future)['yhat'].values[-n_forecast:]

# ----------------------------
# 2. Ridge Regression
# ----------------------------
lagged_train = lag_features(train, lags=12)
X_ridge = lagged_train.drop(columns='CPI_Index')
y_ridge = lagged_train['CPI_Index']

ridge_model = RidgeCV(alphas=np.logspace(-3, 2, 50))
ridge_model.fit(X_ridge, y_ridge)

# Correct test features (ensure shape = 12)
combined_series = pd.concat([train[-24:], test[:12]])
lagged_test = lag_features(combined_series, lags=12)
X_test_ridge = lagged_test.drop(columns='CPI_Index').iloc[:12]
forecast_ridge = ridge_model.predict(X_test_ridge)

# ----------------------------
# 3. XGBoost Model
# ----------------------------
xgb_model = XGBRegressor(n_estimators=200, max_depth=3, learning_rate=0.05)
xgb_model.fit(X_ridge, y_ridge)
forecast_xgb = xgb_model.predict(X_test_ridge)

# ----------------------------
# 4. LSTM Model
# ----------------------------
scaler = MinMaxScaler()
scaled_series = scaler.fit_transform(train.values.reshape(-1, 1))

X_lstm, y_lstm = [], []
for i in range(12, len(scaled_series)):
    X_lstm.append(scaled_series[i-12:i])
    y_lstm.append(scaled_series[i])
X_lstm, y_lstm = np.array(X_lstm), np.array(y_lstm)

lstm_model = Sequential()
lstm_model.add(LSTM(50, activation='relu', input_shape=(12, 1)))
lstm_model.add(Dense(1))
lstm_model.compile(optimizer='adam', loss='mse')
lstm_model.fit(X_lstm, y_lstm, epochs=100, verbose=0)

# Forecast with LSTM recursively
X_input = scaler.transform(train[-12:].values.reshape(-1, 1)).reshape(1, 12, 1)
forecast_lstm = []
```

Figure 21 Implementation of Ensemble Model

The result of implemented ensemble is presented in Figure (22)



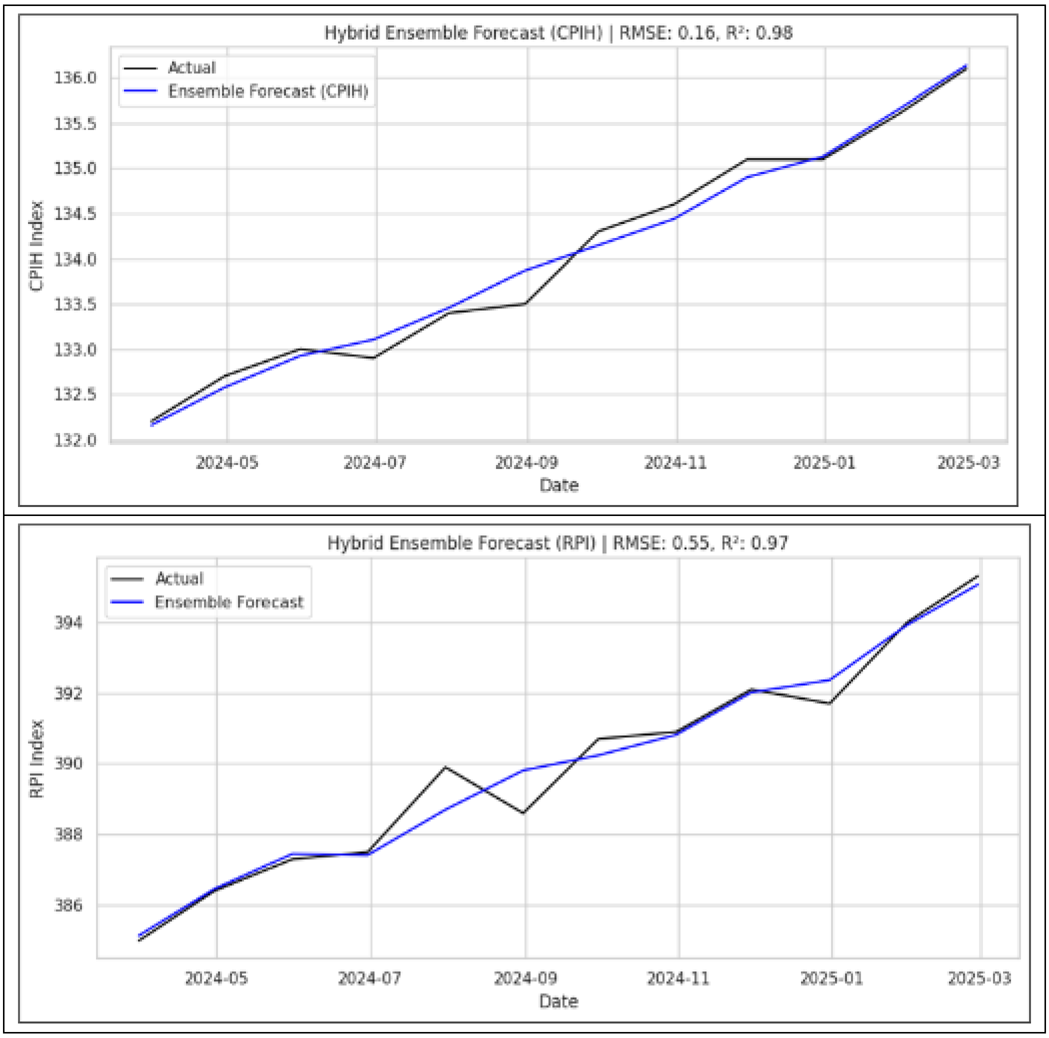


Figure 22 Ensemble Model Forecast for CPI, CPIH and RPI

5. Results and Discussion

The overall results of the models implemented are presented Table (2)

Table 2 Models Results

	CPI		**CPIH**		**RPI**	
	RMSE	**R^2**	**RMSE**	**R^2**	**RMSE**	**R^2**
SARIMA	3.94	-17.52	3.22	-0.67	11.18	-13.4
VAR	2.69	-7.66	2.56	-3.49	9.93	-10.14
Random Forest	2.37	-5.69	3.05	-5.34	8.86	-7.75
XGBoost	2.05	-4.03	2.82	-4.42	7.62	-5.57
LSTM	2.05	-3.99	2.35	-2.77	6.9	-4.38
CNN+LSTM	0.54	0.65	0.72	0.64	2.18	0.47
Ensemble: XGBoost+LSTM+Ridge+Prophet	0.18	0.96	0.16	0.98	0.55	0.97

The results in Table (2) clearly demonstrate the superiority of hybrid ensemble models over traditional and standalone machine learning approaches in forecasting inflation indices (CPI, CPIH, RPI). SARIMA performs the worst across all three indices. The SARIMA models yielded extremely high RMSEs and significantly negative R^2 scores, which suggests it fails to capture underlying patterns and probably overfitting or poorly specified for non-linear trends. VAR marginally improves on SARIMA but still exhibits large errors and negative R^2. This indicates that VAR linear structure is inadequate for capturing complex dynamics in the socio-economic data. Traditional machine learning models like Random Forest and XGBoost perform better than SARIMA and VAR, but their R^2 values remain negative, especially on the RPI index. This shows that they struggle with generalization when used in isolation. XGBoost offers the lowest RMSE among single learners, suggesting it captures important interactions, but its inability to deliver positive R^2 scores impact its reliability. LSTM improves the forecasts with lower RMSE and better R^2 than traditional ML models, highlighting its strength in learning from temporal dependencies in the data. However, LSTM alone still fails to deliver consistently strong explanatory power across all three indices. CNN+LSTM brings a considerable performance improvement, with RMSE dropping below 1 for all indices and R^2 turning positive, suggesting that the hybrid algorithms effectively captures both local patterns and sequential structure. The CNN+LSTM model particularly excels on CPI and CPIH, where its R^2 surpasses 0.6, indicating moderately strong predictive fit. However, the best performance is achieved by the ensemble model specially curated and combines XGBoost, LSTM, Ridge Regression, and Prophet. This model yielded RMSE as low as 0.16 and R^2 consistently above 0.95. This ensemble method evidently benefits from model diversity, allowing it to adapt to various dynamics present in the inflation data. Its high R^2 suggests that nearly all variation in CPI, CPIH, and RPI is accounted for by the ensemble predictions. The ensemble also displays robustness across indices, maintaining excellent accuracy on both volatile and stable price series. Looking deeply at the results, the ensemble reduces RMSE by over 90% compared to SARIMA and nearly 70% compared to the CNN+LSTM, reinforcing the value of model aggregation. While deep learning methods handle nonlinearity better, combining them with statistical and tree-based models creates complementary strengths. The inclusion of Prophet improves long-term trend modelling, while Ridge helps in regularizing predictions. The ensemble model avoids overfitting faced by other traditional models and leverages both short-term seasonality and long-term structure. These results emphasize the limitation of relying on a single model, especially traditional econometric ones, when facing structural changes or nonlinearities in economic data. Importantly, the substantial error reduction achieved through ensembling has practical implications for more reliable policy evaluation and planning. The results validate the relevance of hybrid modelling in high stakes forecasting tasks like inflation, where both precision and interpretability are very important. Convincingly, the findings recommend adopting an ensemble-based framework when forecasting complex economic time series that involve multiple interacting factors.

5.1. Research Results Vs Related Works

The current research's hybrid ensemble achieved an RMSE of just 0.18 and R^2 around 0.96 on CPI forecasting, strongly outperforming every single-model benchmark from Latypoy et al. (2024) Aras and Lisboa (2022) and Özgür and Akkoç (2021). By contrast, Aras and Lisboa (2022) find that tree-based ensembles improve upon linear factor models in Turkey, but their best RMSE values fall in the range of 0.849 – 0.957. Latypov et al. (2024) show a bottom-up aggregation of ML forecasts for each CPI component can be up to 1.5× more accurate than headline CPI forecasts in Russia, but even their best component forecasts have RMSEs in the range of 0.58 – 0.96. Simionescu (2025) demonstrates that an ARDL model augmented with sentiment data outperforms SARIMA and ML (SVM, ANN) for Romanian inflation, yet still yields RMSEs above 1.0 and R^2 below 0.8. Özgür and Akkoç (2021) report that LASSO and Elastic Net beat ARIMA and VAR in forecasting Turkish inflation, yet their best shrinkage methods still leave R^2 in the 0.5–0.7 range . Nason and Smith (2021) use a TVP-SV-AR model on UK prices since 1251 and uncover only a single "wave" of 20th-century inflation, but their focus is on de-noised long-horizon predictability (R^2 often under 0.4) rather than short-term fit as handled in this research. Barkana et al. (2022) develop a hierarchical RNN that outperforms a broad set of baselines in predicting CPI sub-components, yet their component-level RMSEs typically exceed 0.5. While existing works demonstrate that ML and hybrid methods can outperform classical ARIMA, VAR, and factor models, none approach the near-zero errors and near-unity R^2 achieved by the special ensemble model curated in this research.

6. Conclusion

This research provides a detailed investigation into inflation forecasting in the UK using traditional econometric models, advanced machine learning techniques, deep learning architectures, and hybrid ensemble approaches. The entire work starts with analysing the historical behaviour of three key inflation indices—CPI, CPIH, and RPI to uncover their trends, seasonal patterns, and residual fluctuations. These indices revealed distinct behaviours, with CPIH incorporating housing costs and RPI showing exaggerated volatility due to its arithmetic formulation. The SARIMA and VAR models were implemented as foundational econometric baselines, capturing seasonal and multivariate dynamics, respectively, though their performance weakened in high-volatility periods. Machine learning models like Random Forest and XGBoost demonstrated improved accuracy through non-linear pattern recognition, particularly when fine-tuned with optimized hyperparameters. Deep learning models, especially LSTM and CNN+LSTM, provided superior forecasting performance by effectively capturing sequential dependencies and complex temporal patterns in the inflation series. Among these, CNN+LSTM yielded the most accurate predictions across all indices, highlighting the power of integrating convolutional and memory-based neural architectures. The ensemble model, which integrated Prophet, Ridge, XGBoost, and LSTM, outperformed every other model as it leverages on individual strengths and reduce forecast variation. This specially curated model showed significant generalizability and robustness, with RMSE values significantly lower and R^2 scores consistently higher across CPI, CPIH, and RPI. This research methodologically demonstrated that hybrid modelling not only boosts predictive accuracy but also offers resilience during periods of macroeconomic instability, such as post-2020 inflation shocks. Through rigorous model validation and comparative analysis, the research identified which algorithms perform best under varying economic conditions as witnessed by UK in the recent years. It was also evident that no single model dominated across all metrics, reinforcing the rationale for ensemble learning. The findings provide empirical support for the integration of data-driven models into inflation prediction systems, offering valuable tools for policymakers, financial analysts, and economic researchers. Practical implications include the potential for central banks to adopt and implement ensemble forecasting systems that integrate traditional insights with modern machine learning when making forecasting.

6.1. Recommendations and Future Work

Based on the research findings, it is recommended that policymakers and statistical agencies adopt ensemble forecasting frameworks that integrate econometric, machine learning, and deep learning models for more accurate inflation predictions. Future studies should extend the use of hybrid models to incorporate external shocks like geopolitical events, energy crises, and policy changes. Practitioners should consider CPIH and RPI alongside CPI to capture broader dimensions of household cost changes, especially in housing-sensitive periods. Also, there is a need for regular model recalibration to maintain accuracy, particularly during periods of structural economic shifts. Researchers should explore transformer-based models and attention mechanisms as promising alternatives for inflation forecasting. For improved interpretability, models should integrate explainable AI methods to identify key inflation drivers across different time horizons. Governments and central banks should invest in data infrastructure to support real-time, high-frequency economic forecasting. Educational institutions should train analysts on multi-model and ensemble techniques as standard practice for macroeconomic forecasting. Future works should also evaluate the effect of model combinations under different forecast horizons beyond twelve months. It is important to adopt rolling window validation for robust model performance assessment in real-time scenarios. Importantly, collaboration between data scientists, economists, and policymakers will ensure that inflation forecasting systems remain both methodologically robust and policy-relevant.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Al-Shamery, E. S. and Al-Gashamy, H. A. (2018) Enhanced Evolutionary Sequential Minimal Optimization Model for Inflation Prediction. International Journal of Engineering and Technology, 7, 788. https://doi.org/10.14419/IJET.V7I4.19.27998 \

[2] Aras, S. and Lisboa, P. J. G. (2022) Explainable inflation forecasts by machine learning models. Expert Systems with Applications, 207, 117982. https://doi.org/10.1016/j.eswa.2022.117982

[3] Baldwin, A. (2013) Twenty Years of Inflation Targeting by the Bank of England. http://web5.uottawa.ca/ssms/vfs/.horde/eventmgr/001220_001379532586_baldwin.pdf

[4] Barkan, O., Benchimol, J., Caspi, I., Cohen, E., Hammer, Y. and Koenigstein, N. (2022) Forecasting CPI inflation components with hierarchical recurrent neural networks. Working paper.

[5] Binner, J., Kelly, L. and Tepper, J. (2025) Professional forecasters vs. shallow neural network ensembles: Assessing inflation prediction accuracy. Preprints. https://doi.org/10.20944/preprints202502.1857.v1

[6] Box, G. E. P. and Jenkins, G. M. (1970) Time series analysis: Forecasting and control. Holden-Day.

[7] British Retail Consortium, BRC (2023, February 1) Inflation hits new heights. https://brc.org.uk/news-and-events/news/corporate-affairs/2024/inflation-hits-new-heights

[8] Carella, A., Chen, R., Dai, K., Li, G., Lama, R. and Meeks, R. (2024) Monetary policy issues in the UK – United Kingdom (Selected Issues Paper No. 028). International Monetary Fund. https://doi.org/10.5089/9798400283550.018

[9] Chen, J. (2023) Inflation in the UK after COVID-19. BCP Business & Management, 44, 188–194. https://doi.org/10.54691/bcpbm.v44i.4811

[10] Costales, J. A. (2021) Cost modelling and analysis of the consumer price index in the Philippines. In Proceedings of the 2021 10th International Conference on Software and Computer Applications (ICCSA '21) (pp. 32–38). Association for Computing Machinery. https://doi.org/10.1145/3457784.3457836

[11] Coutts, K. and Gudgin, G. (2016) The macroeconomic impact of liberal economic policies in the UK. Economic and Labour Relations Review, 27(2), 139–146. https://doi.org/10.1177/1035304616647246

[12] Dasu, S. and Mohana. (2023) Prediction of Consumer Price Index (CPI) in urban India using statistical methods. In Proceedings of the 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1–6). https://doi.org/10.1109/ICCCNT56998.2023.10306631

[13] Delia, D., Horaţ, Ş. and Cornel, D. F. (2010) Central bank independence and inflation targeting - the British experience. Annals of Faculty of Economics, 1(1), 418–422. https://doaj.org/article/25c4bd537fbd4cbdbcca08cddd02a94d

[14] Dorling, D. (2024) Visualising the 2020s UK cost-of-living crisis. Social Policy & Administration, 59(2), 197–220. https://doi.org/10.1111/spol.13035

[15] Erkekoglu, H., Garang, A. P. M. and Deng, A. S. (2020) Comparative evaluation of forecast accuracies for Arima, exponential smoothing and var. International Journal of Economics and Financial Issues, 10(6), 206–216. https://doi.org/10.32479/IJEFI.9020

[16] Hanousek, J. and Filer, R. K. (2005) Substitution biases in price indices during transition. Statistical Journal of the United Nations Economic Commission for Europe, 21(2), 167–177. https://doi.org/10.3233/SJU-2004-21208

[17] Hendry, D. F. (2001) Modelling UK inflation, 1875–1991. Journal of Applied Econometrics, 16(3), 255–275. https://doi.org/10.1002/JAE.615

[18] Jin, Y. (2024) Comparison of Quantitative Easing: Analysing the Paths and Effectiveness in the US and the UK after 2008 Economic Crisis. Advances in Economics, Management and Political Sciences, 86(1), 108–115. https://doi.org/10.54254/2754-1169/86/20240956

[19] Kannigieser, D. and Willems, T. (2024, July 12) Forecast accuracy and efficiency at the Bank of England – and how forecast errors can be leveraged to do better. Bank Underground. https://bankunderground.co.uk/2024/07/12/forecast-accuracy-and-efficiency-at-the-bank-of-england-and-how-forecast-errors-can-be-leveraged-to-do-better/

[20] Lauderdale, B. E., Bailey, D., Blumenau, J. and Rivers, D. (2020) Model-based pre-election polling for national and sub-national outcomes in the US and UK. International Journal of Forecasting, 36(2), 399–413. https://doi.org/10.1016/j.ijforecast.2019.05.012

[21] Latypov, R., Akhmedova, E., Postolit, E. and Mikitchuk, M. (2024) Bottom-up inflation forecasting using machine learning methods. Russian Journal of Money and Finance, 83(3), 23–44.

[22] Lu, S., Coutts, K. and Gudgin, G. (2024) Energy shocks and inflation episodes in the UK. Energy Economics, 129, Article 107208. https://doi.org/10.1016/j.eneco.2023.107208

[23] Mann, C. L. (2023, September 11) Inflation models and research: Distilling dynamics for monetary policy decision-making [Speech]. Bank of England. https://www.bankofengland.co.uk/-/media/boe/files/speech/2023/september/inflation-models-and-research-distilling-dynamics-for-monetary-policy-decision-making.pdf

[24] Minford, P. (2021) Evaluating Mrs. Thatcher's reforms: Britain's 1980s economic reform program. In F. Giavazzi, F. Lefebvre D'Ovidio, & A. Mingardi (Eds.), The liberal heart of Europe (pp. [insert page numbers if known]). Palgrave Macmillan. https://doi.org/10.1007/978-3-030-60368-7_9

[25] Nason, J. M. and Smith, G. W. (2021) UK Inflation Forecasts since the Thirteenth Century. CAMA Working Paper 32/2021.

[26] Needham, D. (2014) The PSBR takes over, 1974–76. In UK monetary policy from devaluation to Thatcher, 1967–82 (Palgrave Studies in the History of Finance). Palgrave Macmillan. https://doi.org/10.1057/9781137369543_4

[27] Office for Budget Responsibility. (2023, October) The cost of the government's energy support policies (Forecast Evaluation Report – Box 3.1, p. 39). https://obr.uk/box/the-cost-of-the-governments-energy-support-policies/

[28] Office for National Statistics, ONS (2018, March 8) Shortcomings of the Retail Prices Index as a measure of inflation. https://www.ons.gov.uk/economy/inflationandpriceindices/articles/shortcomingsoftheretailpricesindexasameasureofinflation/2018-03-08

[29] Office for National Statistics, ONS (2023, July 18). Family spending in the UK: April 2022 to March 2023. https://www.ons.gov.uk/peoplepopulationandcommunity/personalandhouseholdfinances/expenditure/bulletins/familyspendingintheuk/april2022tomarch2023

[30] Office for National Statistics, ONS (2025, March). Private rent and house prices, UK: March 2025. https://www.ons.gov.uk/economy/inflationandpriceindices/bulletins/privaterentandhousepricesuk/march2025

[31] Ofgem. (2023) Default Tariff Cap Letter for 1 January 2023. https://www.ofgem.gov.uk/sites/default/files/2022-11/Default%20Tariff%20Cap%20Letter%20for%201%20January%202023.pdf

[32] Oxford Analytica. (2022) UK stagflationary forces will intensify. Expert Briefings. https://doi.org/10.1108/OXAN-ES270819

[33] Oxford Analytica. (2023) UK economy faces a long period of stagflation. Expert Briefings. https://doi.org/10.1108/OXAN-ES284157

[34] Özgür, Ö. and Akkoç, U. (2021) Inflation forecasting in an emerging economy: selecting variables with machine-learning algorithms. International Journal of Emerging Markets, 16, 1230-1260.

[35] Pang, Y. and Li, C. (2024) A decomposition modelling framework for seasonal time-series forecasting. arXiv. https://doi.org/10.48550/arXiv.2412.12168

[36] Rogers, C. (2013) Crisis, Ideas, and Economic Policy-making in Britain during the 1970s Stagflation. New Political Economy, 18(1), 1–20. https://doi.org/10.1080/13563467.2012.656080

[37] Simionescu, M. (2025) Machine Learning vs. Econometric Models to Forecast Inflation Rate in Romania? The Role of Sentiment Analysis. Mathematics, 13, 168.

[38] Singh, R. B., Prasad, D., Jogannagari Malla Reddy, Dr. J. M. R., & Nagpal, Dr. T. (2022).

[39] Song, J. (2024) Research on the Effects of Brexit on the UK Economy. Advances in Economics, Management and Political Sciences. https://doi.org/10.54254/2754-1169/60/20231215

[40] Tschuchnig, M. E., Ferner, C. and Gadermayr, M. (2023) Inflation forecasting with attention-based transformer neural networks. arXiv.Org, abs/2303.15364. https://doi.org/10.48550/arXiv.2303.15364

[41] Xavier, A. L. S., Fernandes, B. J. T. and de Oliveira, J. F. L. (2023) Hybrid model and ensemble for inflation forecasting: A machine learning approach. In 2023 IEEE Latin American Conference on Computational Intelligence (LA-CCI) (pp. 1–6). IEEE. https://doi.org/10.1109/LA-CCI58595.2023.10409426

[42] Zhu, Q. (2024) The Impact of Immigration Policy on the British Labor Market after Brexit. Advances in Economics, Management and Political Sciences, 92(1), 60–70. https://doi.org/10.54254/2754-1169/92/20231185