FORECASTING INFLATION IN SOUTHEAST ASIA USING ARIMA AND MACHINE LEARNING MODELS

Assignment Type:

Time-Series Forecasting Dissertation / Technical Report

Tools Used:

Python (statsmodels, scikit-learn), IMF Data, World Bank Indicators, Jupyter Notebook

1. Introduction

Accurate inflation forecasting is critical for monetary policy and investment decisions. In Southeast Asia, volatile commodity prices and exchange rate fluctuations complicate the inflation outlook. This project compares traditional econometric models and machine learning methods to forecast monthly CPI-based inflation in three countries: Thailand, Indonesia, and Vietnam.

2. Research Questions

1. How well does the ARIMA model perform in short-term inflation forecasting compared to ML models like Random Forest and XGBoost?
2. What are the leading indicators that improve inflation forecast accuracy in emerging markets?
3. Can hybrid models outperform single-model approaches?

3. Literature Review Highlights

- **Stock & Watson (2001)**: Time series and leading indicators improve inflation prediction
- **Gourinchas & Obstfeld (2012)**: Exchange rates and global commodity prices drive inflation in EMs
- **Makridakis et al. (2018)**: Hybrid ML models outperform classical methods in economic forecasting
- **Ng & Wright (2013)**: Forecast combination yields robust predictions

4. Data Collection and Description

Variable	Frequency	Source	Notes
Consumer Price Index (CPI)	Monthly	IMF IFS	Main dependent variable
Crude Oil Price (Brent)	Monthly	U.S. EIA	Leading commodity indicator
Exchange Rate (USD)	Monthly	World Bank	Inflation pass-through effect
Money Supply (M2)	Monthly	IMF	Monetary policy indicator
Interest Rate	Monthly	IMF	Policy rate (discount or repo)
GDP Growth (Quarterly)	Quarterly	World Bank	Interpolated to monthly via quadratic match

Data Span: Jan 2008 – Dec 2023

Countries Analyzed: Thailand, Indonesia, Vietnam

5. Methodology

5.1 ARIMA Modelling (Country-Specific)

- Stationarity Check (ADF Test)
- AIC-based model selection
- Residual diagnostics and Ljung-Box test

$$ARIMA(p,d,q): CPIt = \alpha + \sum i = 1^p \phi_i \, CPIt - i + \sum_{j=1}^{q} \theta_j \, \varepsilon t - j + \varepsilon_t$$

5.2 Machine Learning Models

- **Random Forest Regressor**
- **XGBoost Regressor**
- **Features**: Lagged CPI (up to 12 months), exchange rate, oil price, interest rate, M2

5.3 Evaluation Metrics

- Root Mean Square Error (RMSE)
- Mean Absolute Percentage Error (MAPE)

- Directional Accuracy (%)

6. Results

Model Performance Comparison

Model	Country	RMSE	MAPE (%)	Directional Accuracy
ARIMA	Thailand	0.42	4.7	73.1%
RandomForest	Thailand	0.38	4.1	77.5%
XGBoost	Thailand	0.34	3.6	79.4%
ARIMA	Vietnam	0.49	5.2	70.2%
XGBoost	Vietnam	0.37	3.9	78.8%

Visuals Included

- Time series plot of CPI and forecasts (ARIMA vs ML)
- Residual plots for ARIMA models
- Feature importance chart for XGBoost
- Directional accuracy bar chart

7. Interpretation of Results

- Machine learning models, particularly XGBoost, consistently outperform ARIMA on both error metrics and directional accuracy.
- Crude oil price and exchange rate volatility rank as the top two predictors across all countries.
- ARIMA performs adequately for short-term forecasts (1–2 months ahead) but struggles with sudden shocks.

8. Policy Implications

- **Central Banks**: May benefit from hybrid model forecasting tools during monetary planning
- **Investors**: ML-based models offer more accurate CPI forecasts for hedging strategies

- **Statistical Offices**: Should integrate real-time commodity and exchange rate data in inflation models

9. Limitations

- Monthly interpolated GDP data introduces potential smoothing error
- ML models are black-box in interpretation, though SHAP values help
- External shocks (e.g., COVID-19) challenge all models' accuracy

10. Deliverables to Student

- Full 8,000-word research project formatted for dissertation
- Python Jupyter Notebook with detailed step-by-step execution
- Editable graphs in PNG and Excel
- Summary poster for classroom/academic display
- APA reference list of 18 sources
- Appendix: Data cleaning steps, code chunks, and model tuning results