Comparative Analysis of Machine Learning Models for GDP Forecasting: A Case Study on the Indian Economy

...

Suresh Nepal [22MDT0160] and Mohana Krishnan[22MDT0047]

Introduction

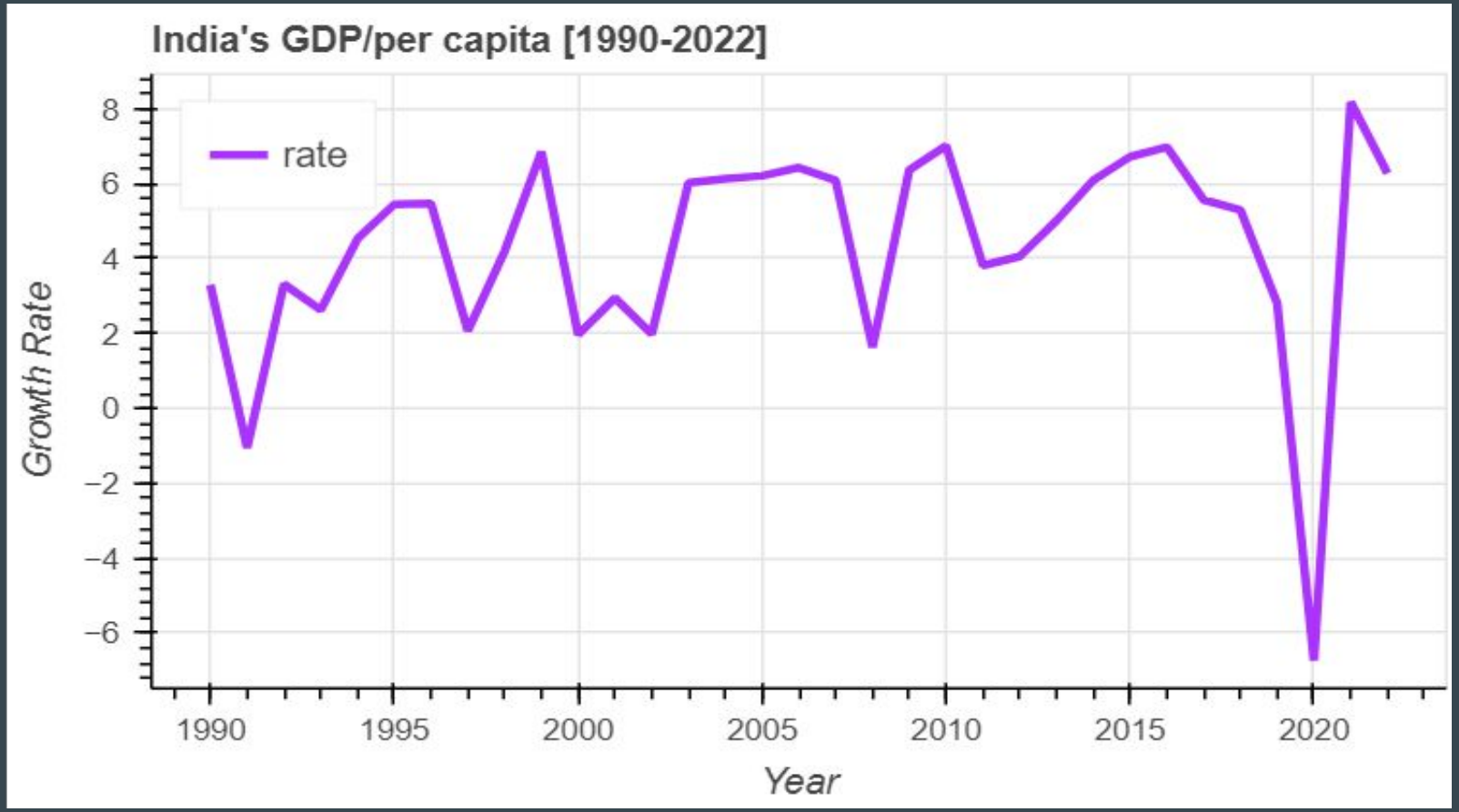
- GDP per capita is an important economic indicator that measures the average income or standard of living of the residents in a country.
- Many research and forecasting models are available with respect to GDP hence GDP per capita was chosen.
- Machine learning (ML) algorithms are been used for time series analysis in different aspects of the economy.
- For governments and policymakers to create successful macroeconomic policies, accurate GDP/GDP per capita forecasting is essential.

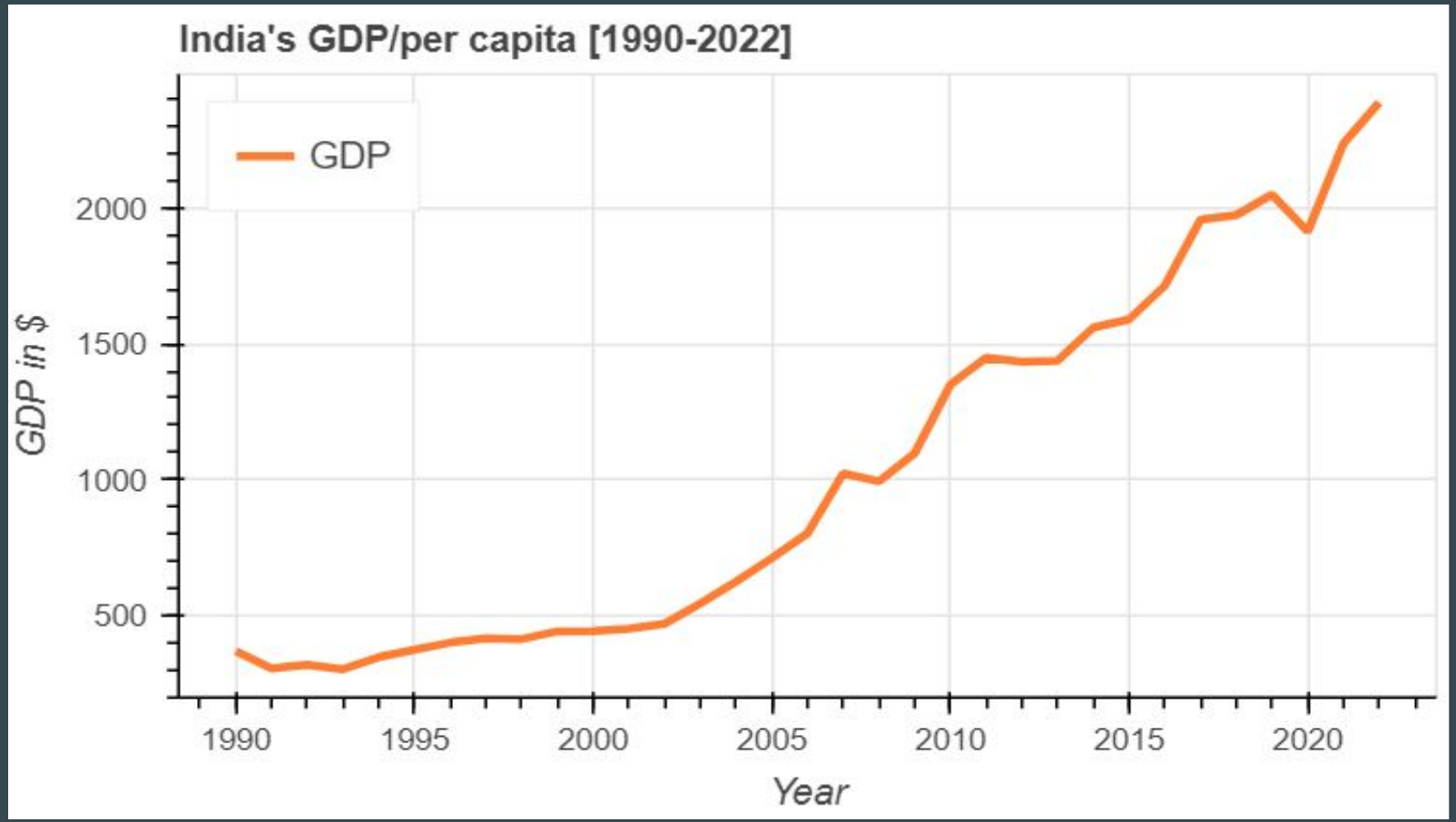
Objectives

- Basic analysis of the GDP per capita data of India (1990-2022)
- Obtain baseline with time series models such as ARIMA.
- Determine how well selected machine learning algorithms (ML algorithms that are deemed better for economic data) do at prediction of GDP per capita.

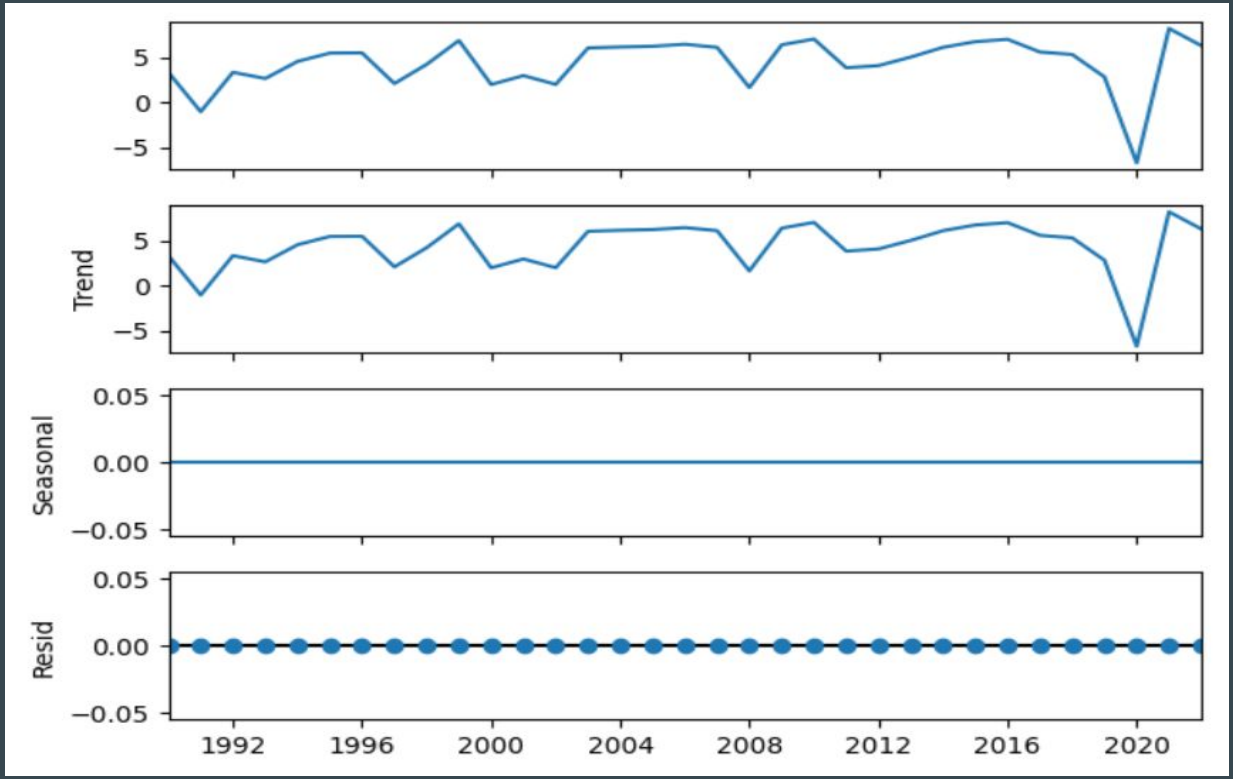
About the Data

- Data Source: World Bank website.
- Date range: 1990 - 2022.
- Yearly actual GDP per capita and the growth rate.
- Figure shows actual data.


India's GDP/per capita [1990-2022]
rate
Growth Rate
8
6
4
2
0
−2
−4
−6
1990
1995
2000
2005
2010
2015
2020
Year

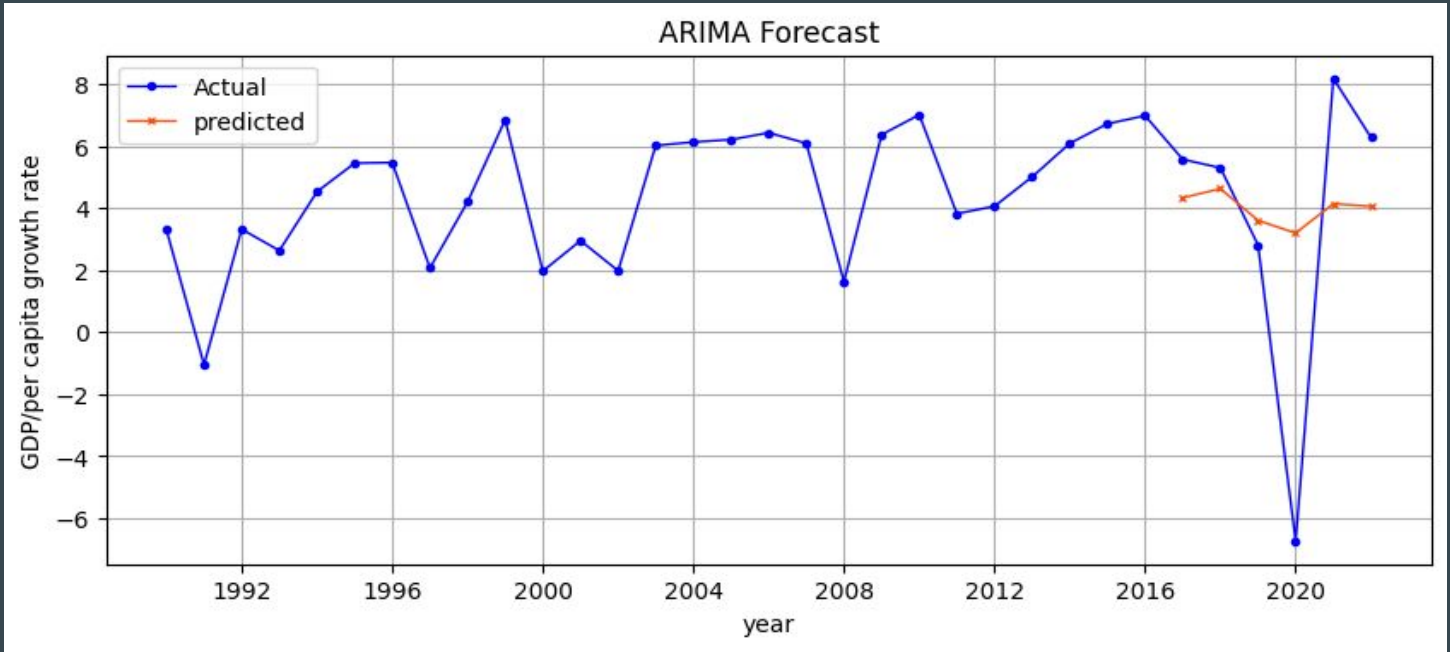

India's GDP/per capita [1990-2022]
GDP
GDP in $
2000
1500
1000
500
1990
1995
2000
2005
2010
2015
2020
Year

Seasonal Decomposition



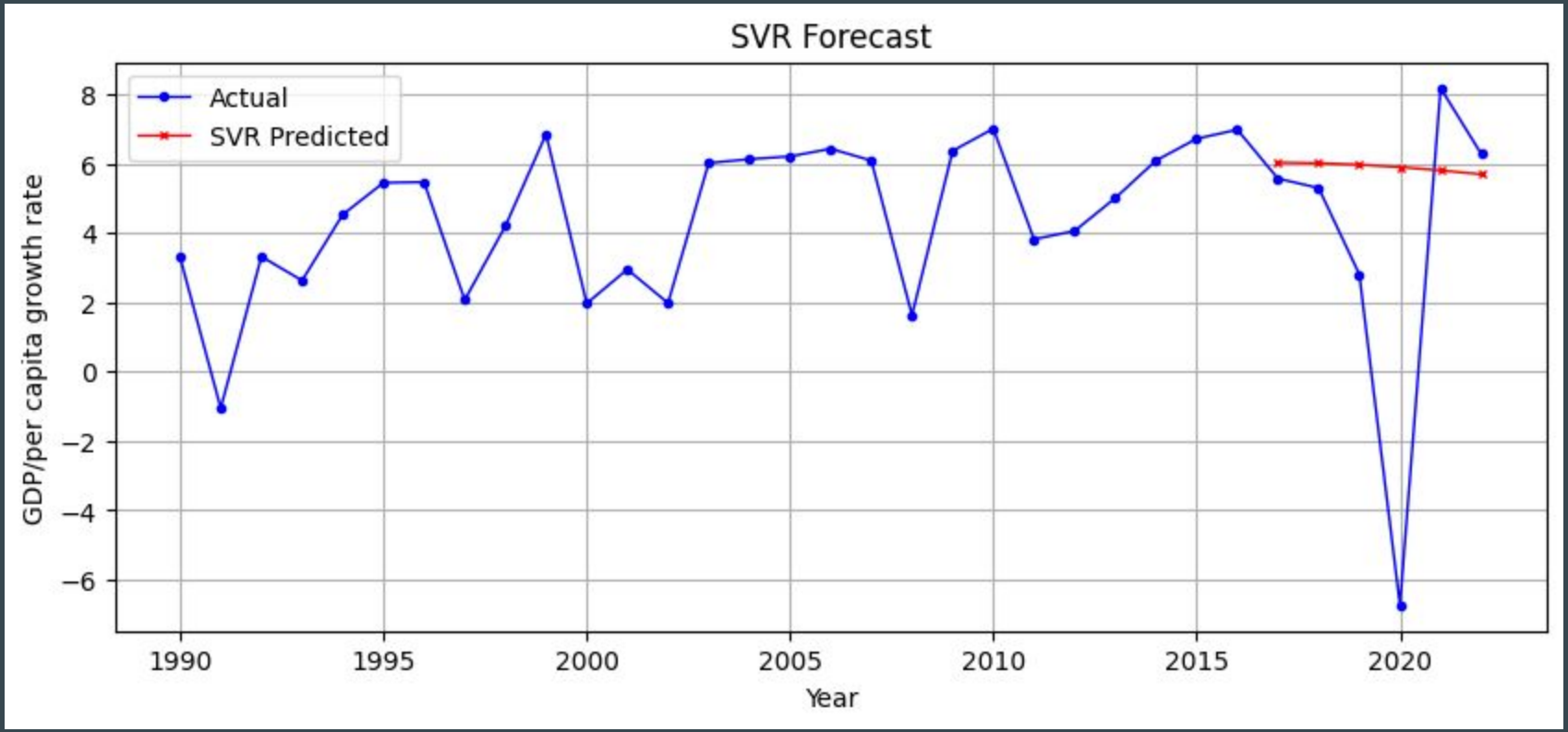
ARIMA [AutoRegressive Integrated Moving Average]

- Optimal hyperparameter (p, d, q) = (4, 0, 4)
- RSME = 4.515



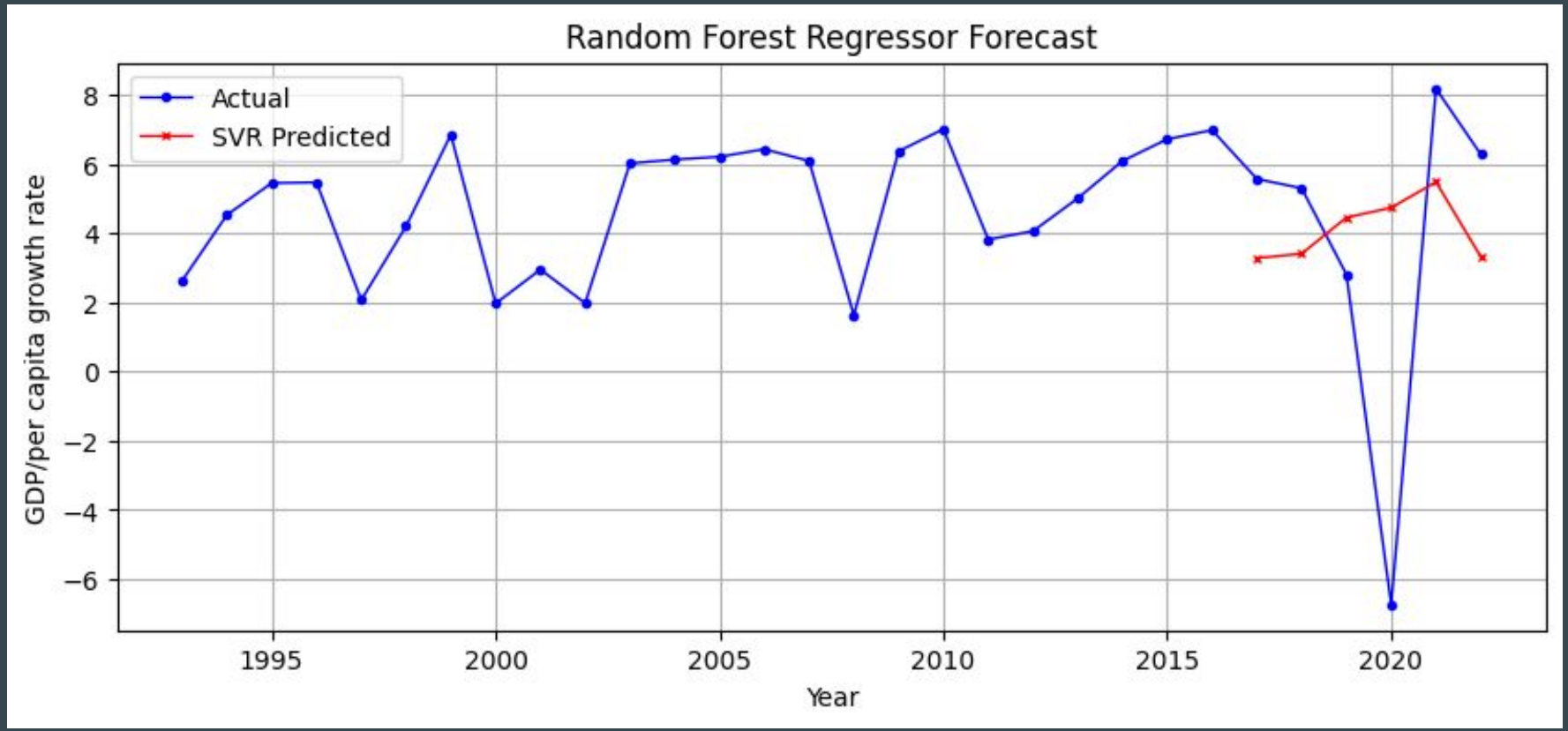
Support Vector Regressor

- SVR finds a hyperplane that best fits the data while minimizing the margin of error, and this hyperplane is used to make predictions.
- When applied to time series forecasting, SVR aims to find a regression function that captures the underlying patterns and relationships within the time series data
- Hyperparameter: Kernel function → Radial Basis Function (rbf)
- When data exhibits complex, non-linear relationships we use RBF kernel (Gaussian) instead of linear and polynomial kernel.
- RSME = 5.42


SVR Forecast
Actual
SVR Predicted
GDP/per capita growth rate
8
6
4
2
0
−2
−4
−6
1990
1995
2000
2005
2010
2015
2020
Year

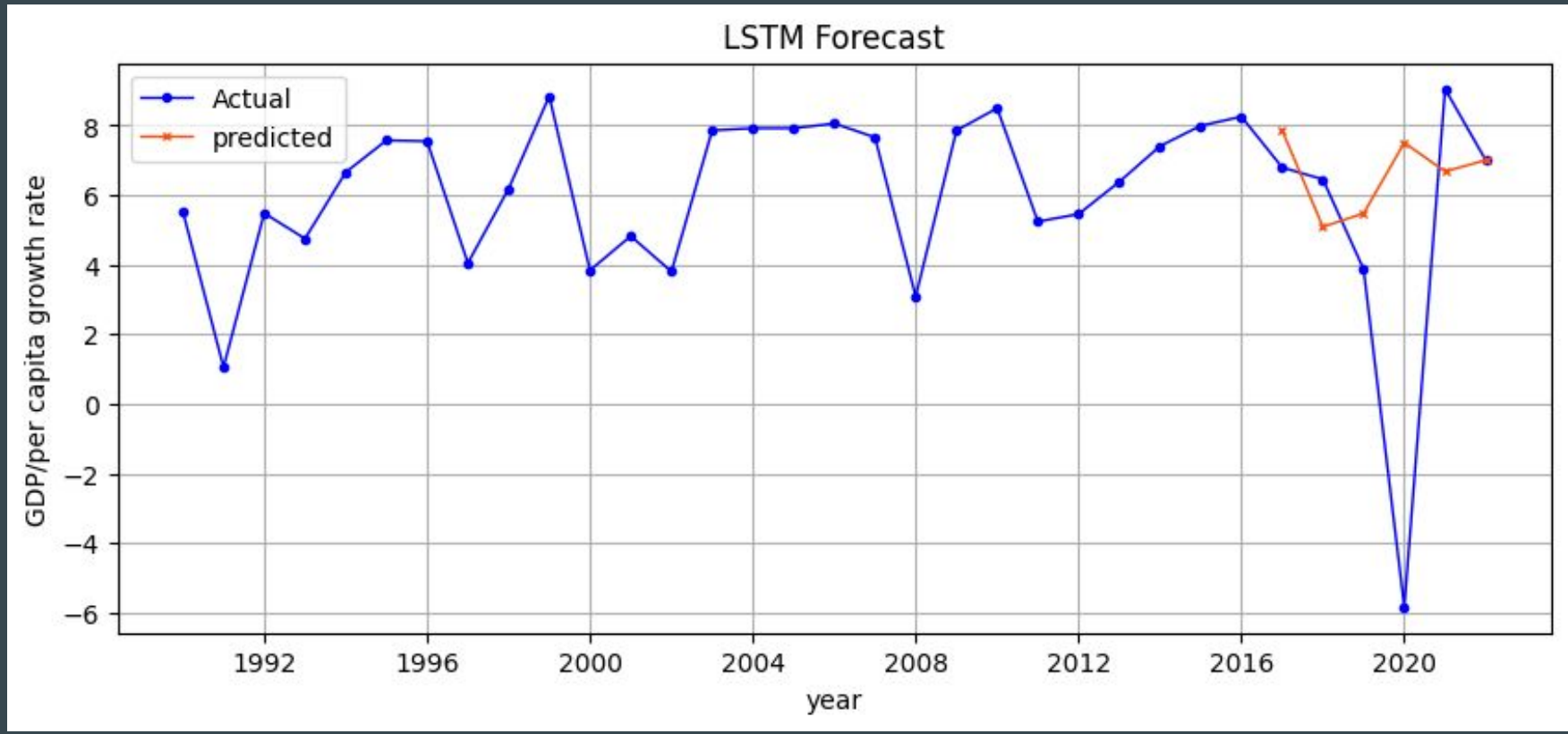
Random Forest (RF) Regressor

- Random forest regressors can be used for time series forecasting by transforming the time series data into a supervised learning problem.
- This is done by creating lagged features, which are the values of the time series at previous time steps.
- The lagged features are then used as the input features to the random forest regressor.
- The random forest regressor then learns to predict the future values of the time series based on the lagged features.
- RMSE = 5.14


Random Forest Regressor Forecast
Actual
SVR Predicted
GDP/per capita growth rate
Year
8
6
4
2
0
−2
−4
−6
1995
2000
2005
2010
2015
2020

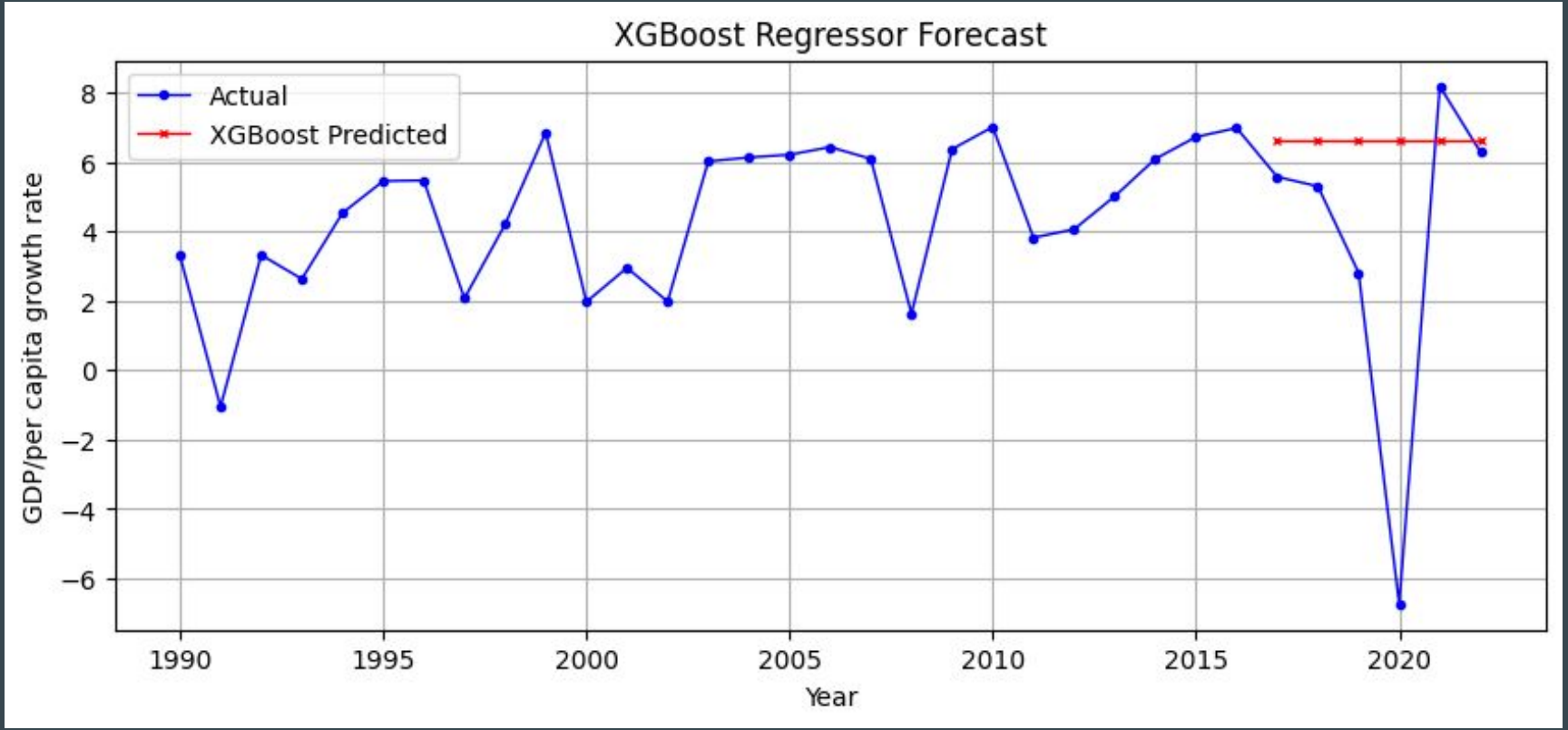
Long Term-Short Term Memory (LSTM)

- Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN).
- LSTMs are particularly effective for modeling and predicting time series data due to their ability to learn and remember information over extended periods.
- Hyperparameters, such as the number of LSTM units, the number of layers, and the learning rate, may need to be tuned to optimize performance.
- RMSE = 5.61


LSTM Forecast
Actual
predicted
GDP/per capita growth rate
8
6
4
2
0
−2
−4
−6
1992
1996
2000
2004
2008
2012
2016
2020
year

XGBoost

- XGBoost is an implementation of gradient boosting, a machine learning technique that builds a series of weak models (typically decision trees) and combines them to create a strong predictive model.
- The weak learners in XGBoost are decision trees. XGBoost builds an ensemble of trees sequentially, with each tree correcting the errors of the previous ones.
- RMSE = 5.75


XGBoost Regressor Forecast
Actual
XGBoost Predicted
GDP/per capita growth rate
8
6
4
2
0
−2
−4
−6
1990
1995
2000
2005
2010
2015
2020
Year

Summary

Model	RMSE
ARIMA	4.51
Support Vector Regressor (SVR)	5.42
Random Forest Regressor	5.15
LSTM	5.61
XGBoost Regressor	5.75

Thank You