Model Prediction F1 racing

GOAL:

- Forecast 5 future lap times for a F1 driver for a specific race using data from 2014-2023

APPROACH:

- Statistical Learning
- Machine Learning

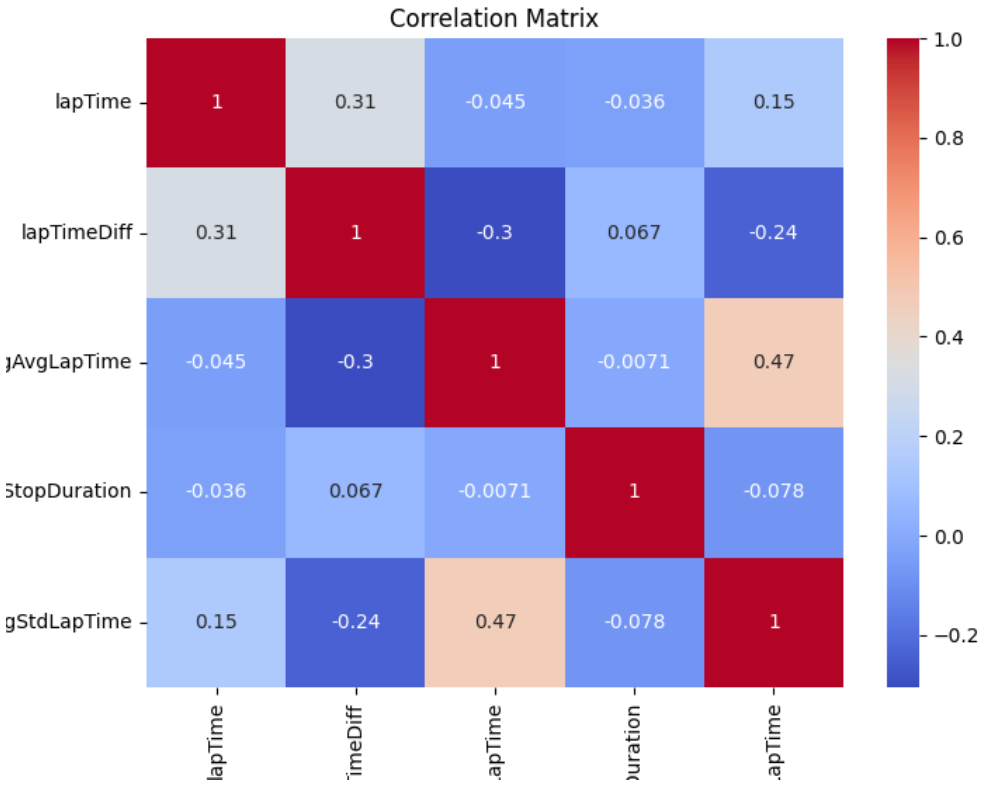
ERROR METRICS:

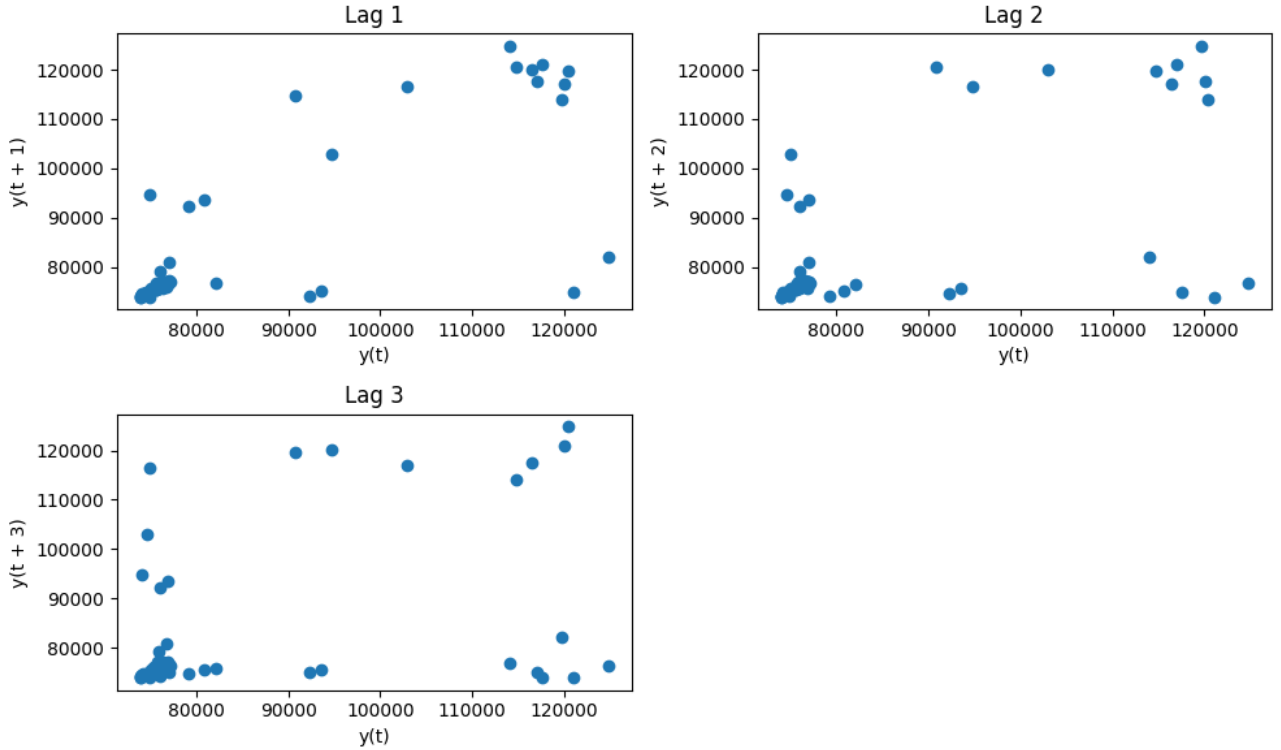
- RMSE (Root Mean Squared Error)
- MAE (Mean Absolute Error)

STEPS:

- Data Aggregation
- Data Visualization
- Model Selection
- Model Tuning
- Comparison of Models

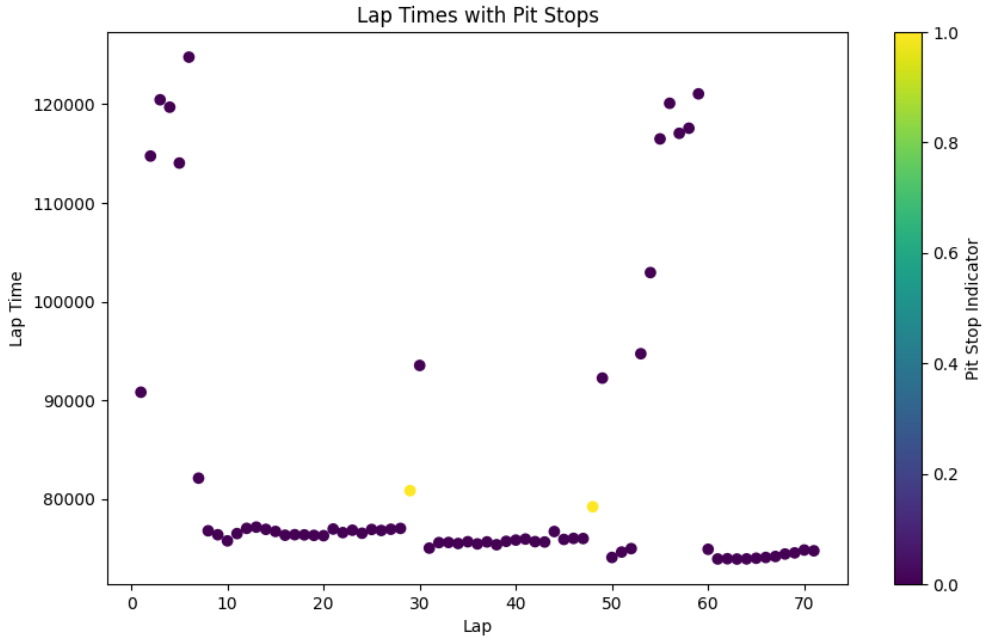
Data Visualization

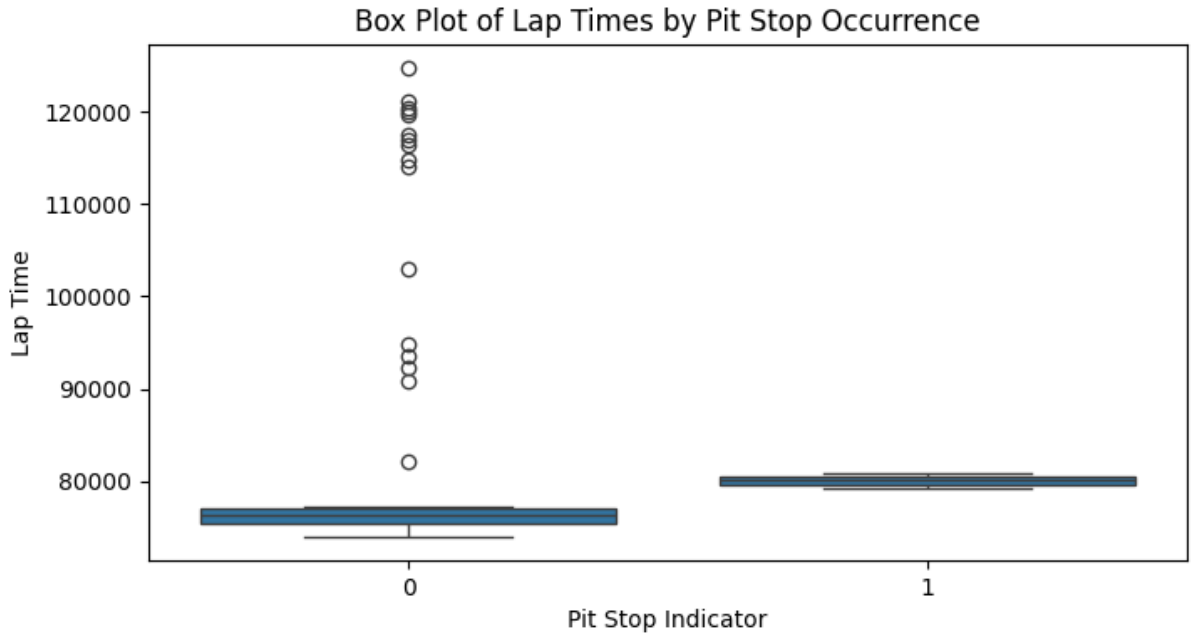




- Moderate positive and negative correlation among some variables
- If lag plot is linear, we can infer that the underlying structure is of the autoregressive model and autocorrelation is present
- If lag plot is of elliptical shape, we can say that the underlying structure represents some continuous periodic function
- Not clear from the lag plots the distribution of the data

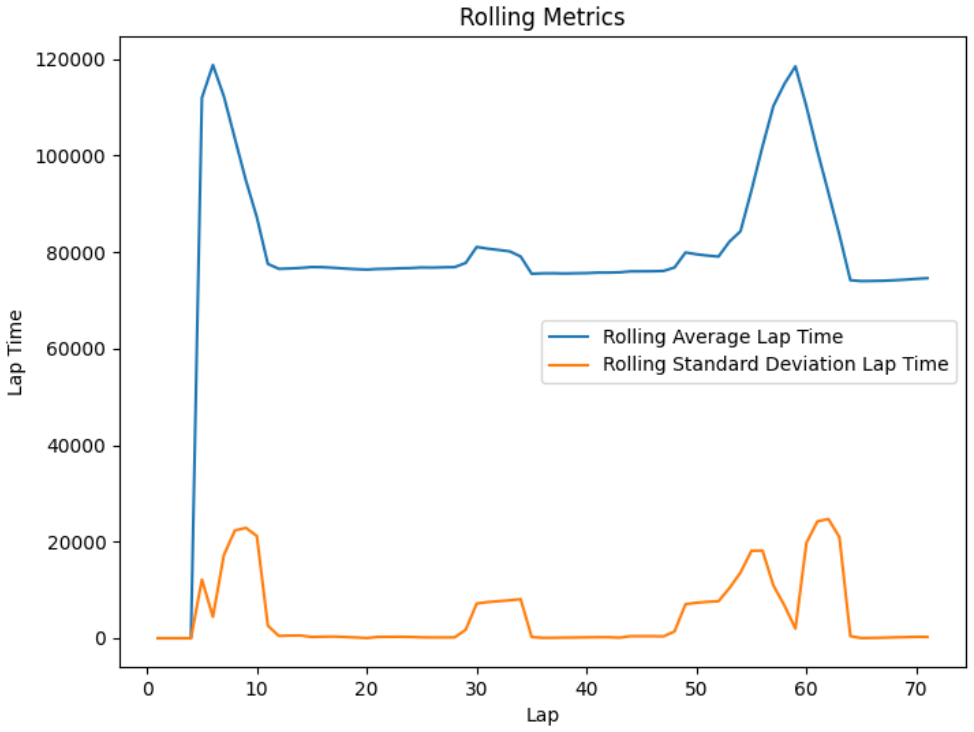
Data Visualization

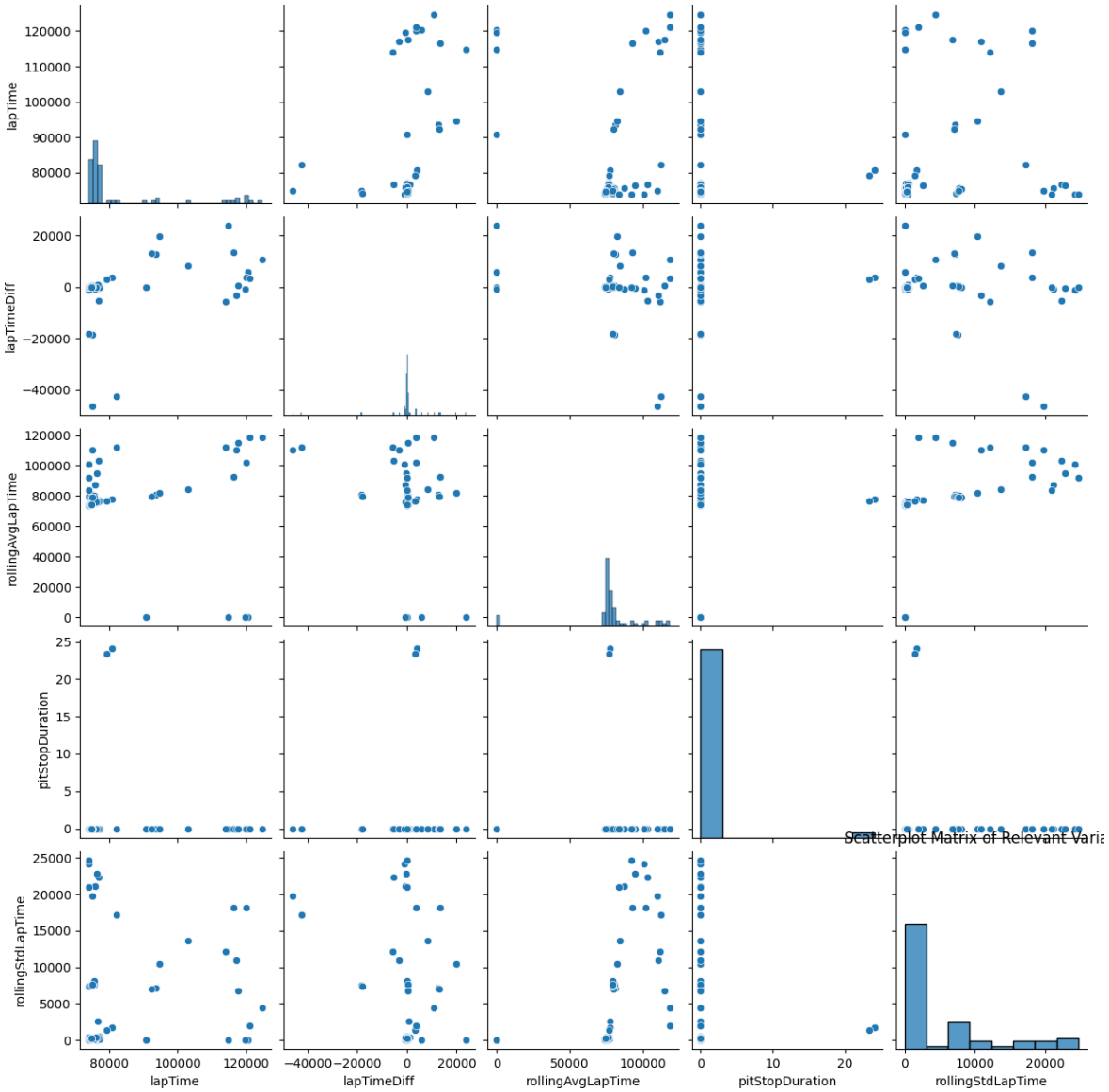




The presence of sudden peaks in the lap times can be attributed to pit stops during the race. These pit stops lead to abrupt increases in lap times compared to the general trend. By incorporating pit stop information into our dataset, we effectively capture and explain these periodic spikes in lap times.

Data Visualization

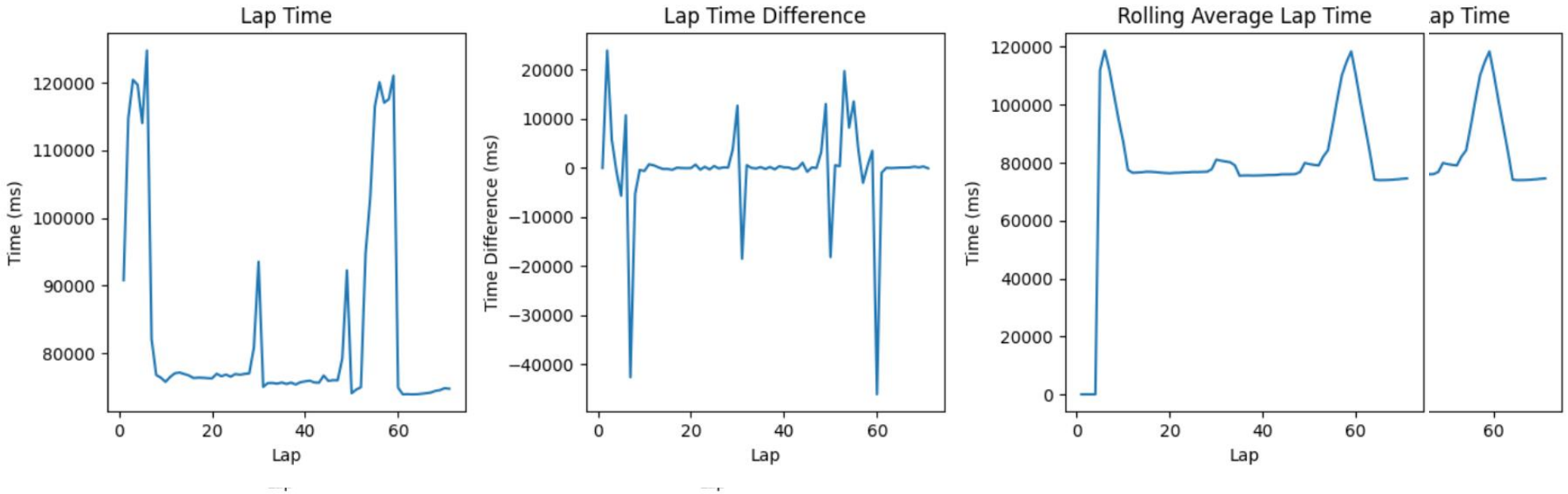


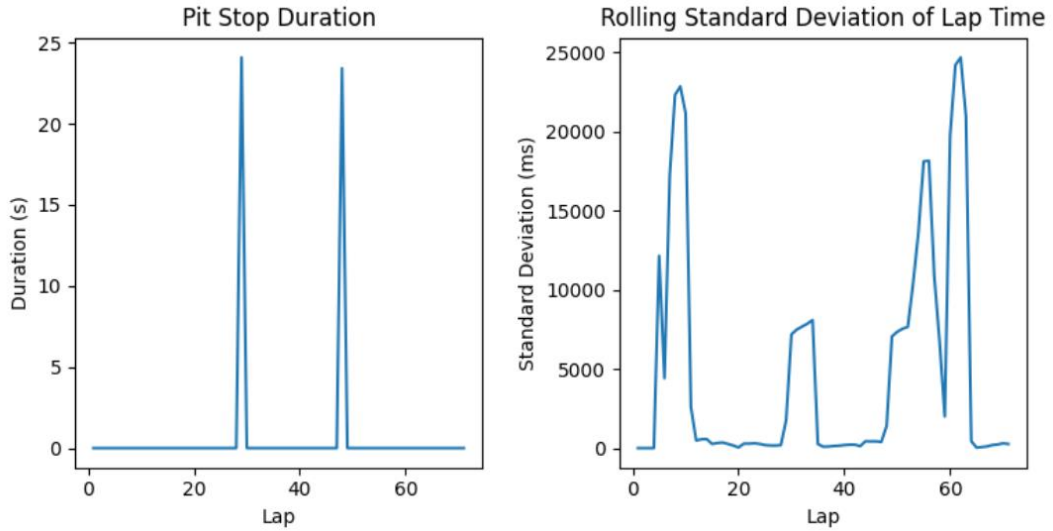


- The rolling average lap time captures the overall trend and smoothens out the short-term fluctuations

- The rolling standard deviation, on the other hand, measures the variability of lap times within a sliding window

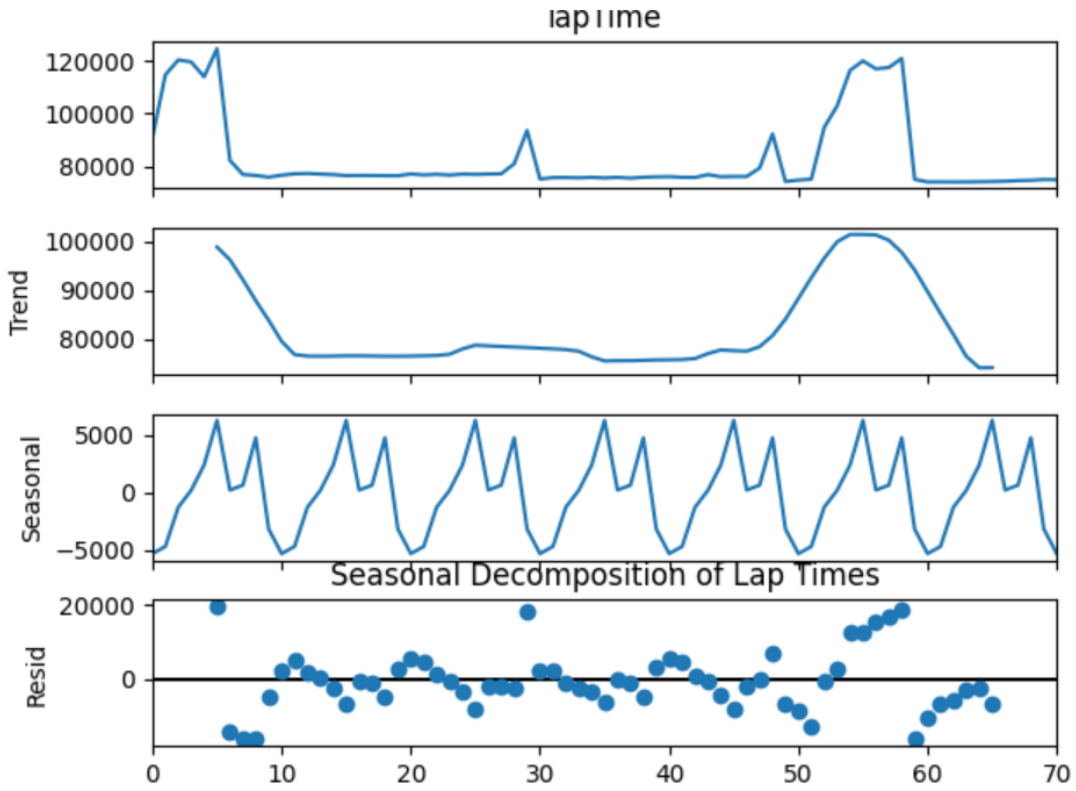
Data Visualization

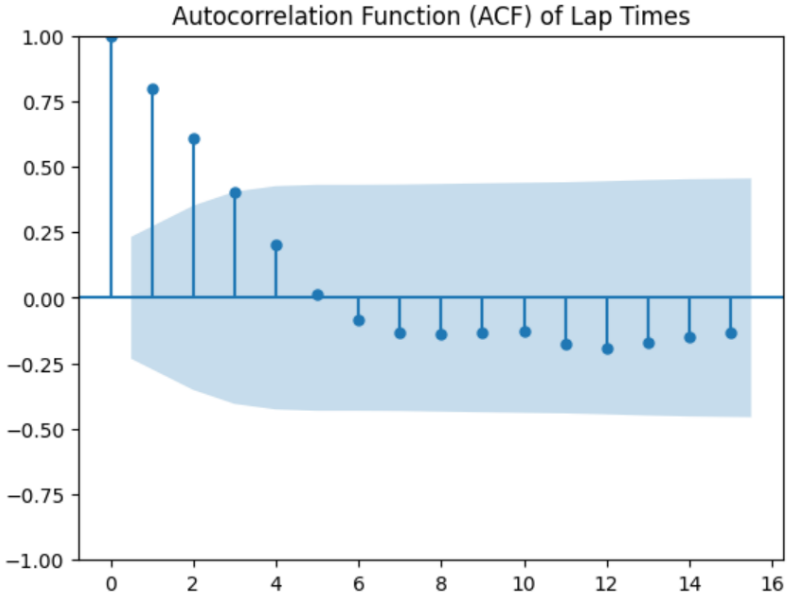


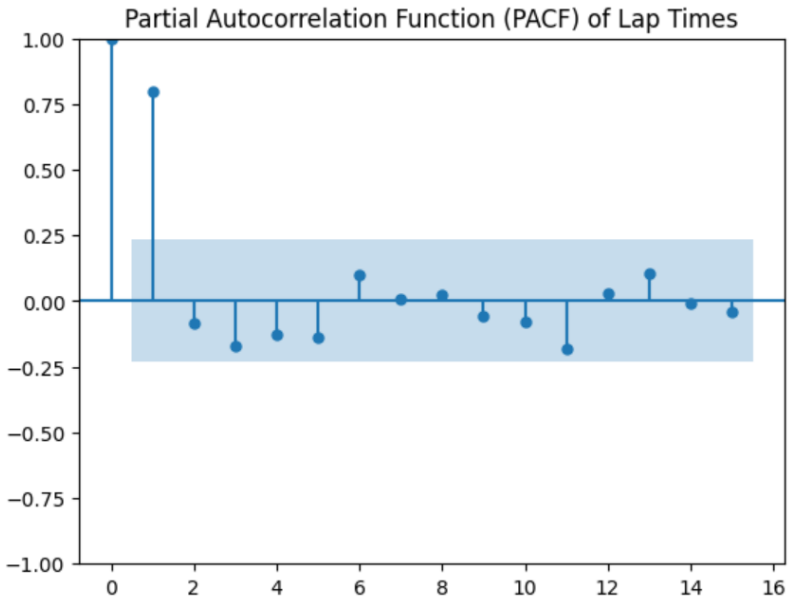


- We can observe that the standard deviation remains relatively constant, and the mean does not exhibit significant shifts over time
- Additionally, there is an absence of strong seasonality patterns indicating stationarity

Data Visualization







- ADF test used for stationarity
- One variable is found non-stationary and is made stationary by using differences
- From ACF plots it can be seen that high correlation values with the few initial lags of lap times

Error Metrics for Models

- RMSE and MAE as evaluation metrics is appropriate for this regression task of predicting lap times
- RMSE gives a higher penalty to larger errors, while MAE is more interpretable and less sensitive to outliers.
- Data has few examples and hence advanced auto-regression methods like VAR or deep learning methods like Attention based LSTMs won’t be needed and simpler models can suffice
- Multivariate time series forecasting and for this both these models can function well. (due to less data, ARIMA can conduct good multivariate predictions even though it is mainly good for univariate settings).

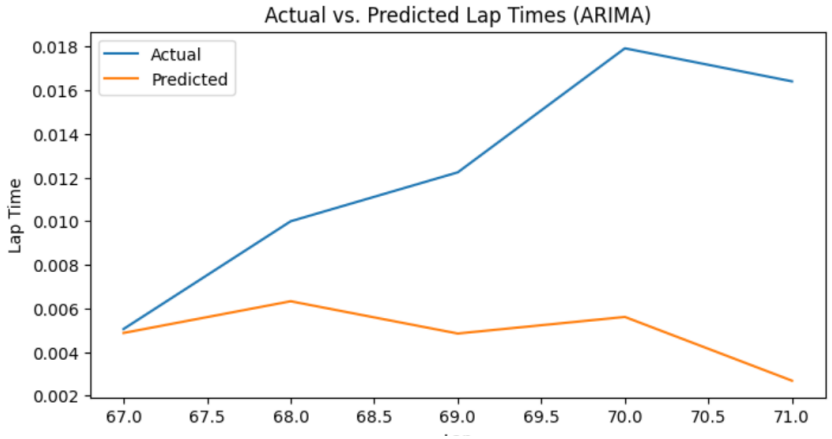
ARIMA Model Specification

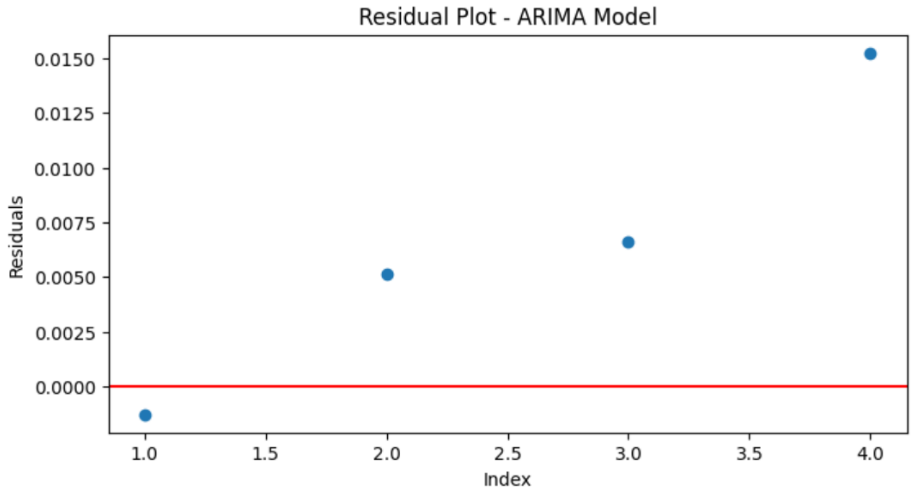
Best Hyperparameters with lowest RMSE in train set:

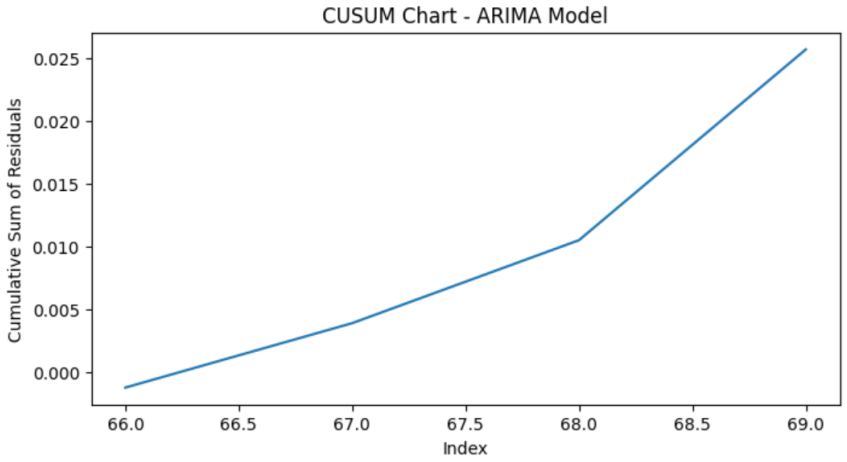
(p,d,q) = (2,1,0) ⟶ ARIMA(2, 1, 0) model

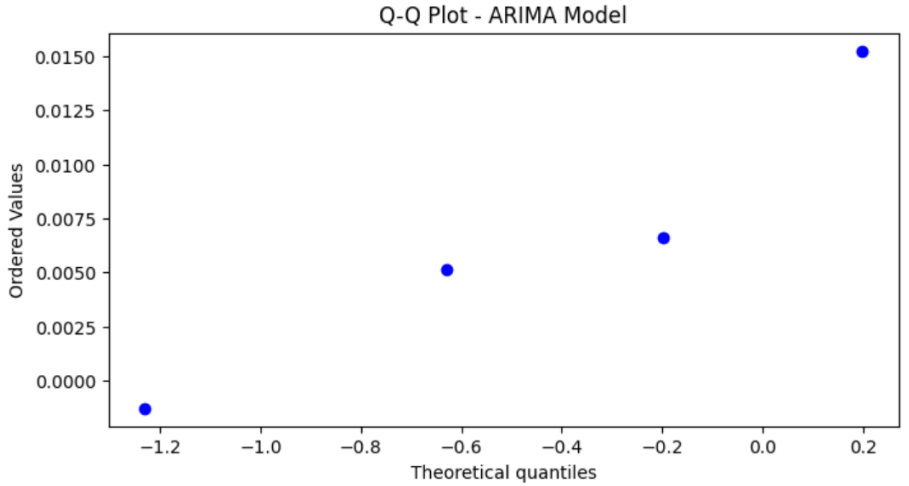
Test RMSE = 0.009
Test MAE = 0.007

ARIMA Model Results



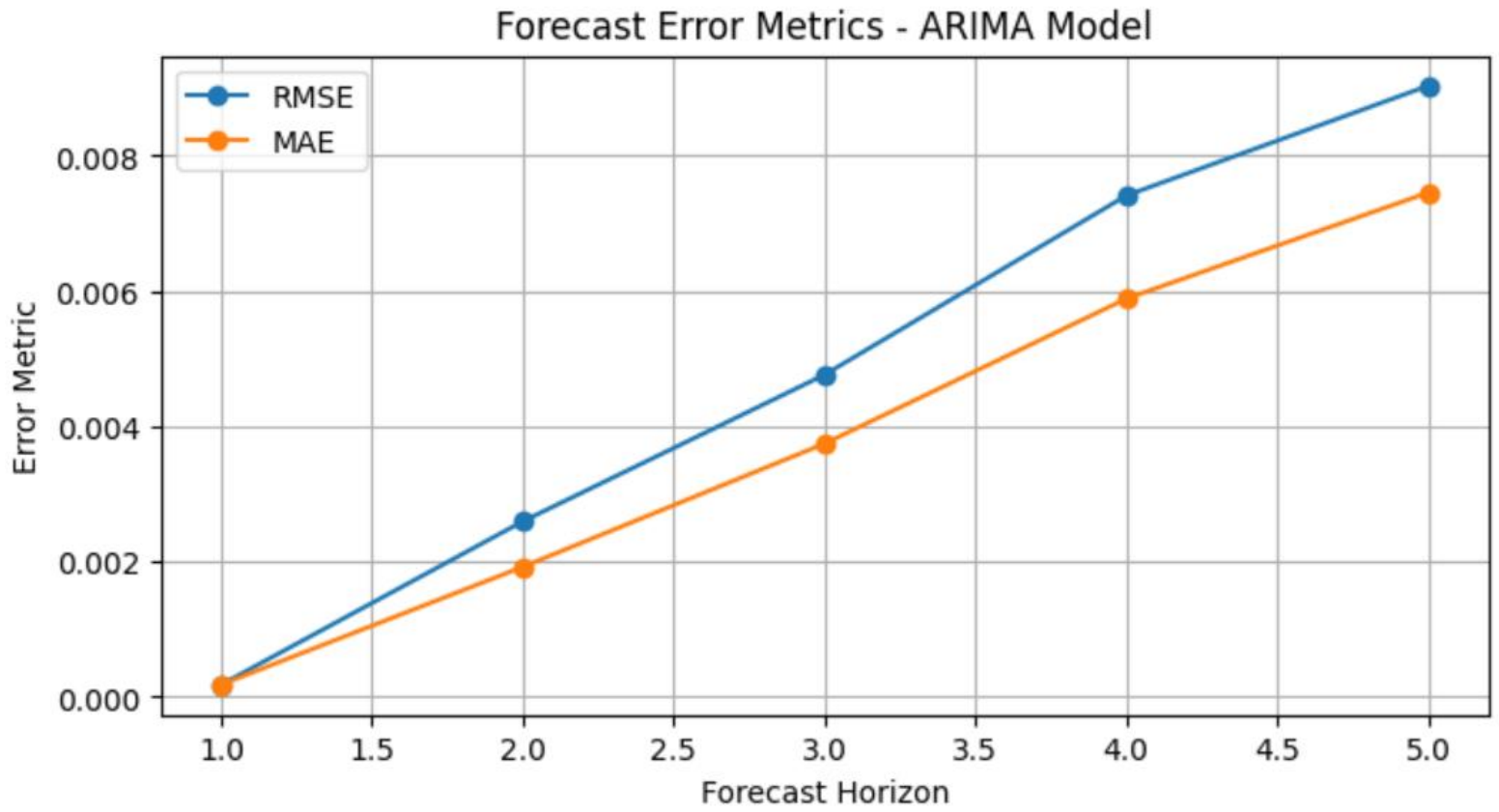






- CUSUM chart and test is used to monitor whether a process is drifting away from its mean. The cumulative sums stay within a region near the expected value of zero
- Q-Q plot can be used to quickly check the normality of the distribution of residual errors - here it can be compared to Gaussian distribution

ARIMA Model Results



LSTM Model Specification

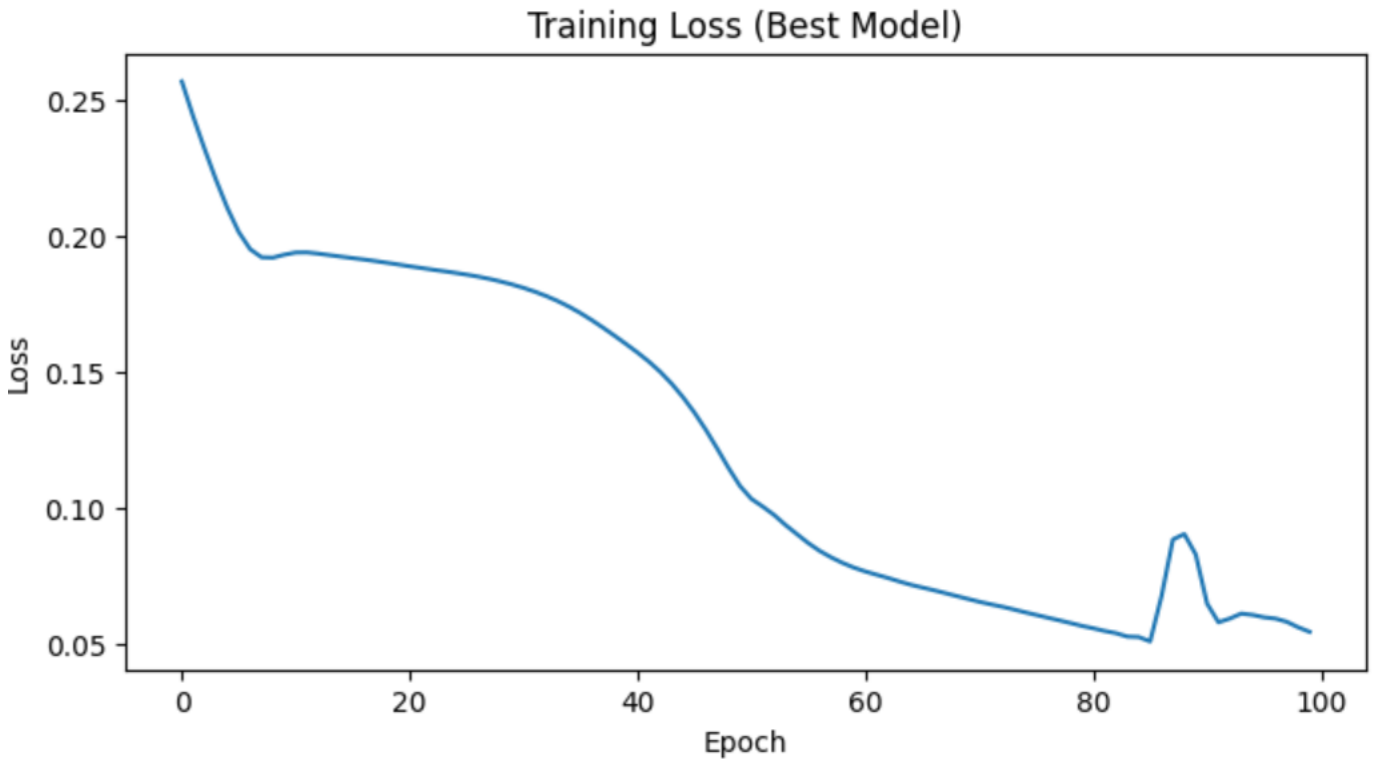
hidden_dim = 32
num_layers = 2

num_epochs = 100
batch_size = 16

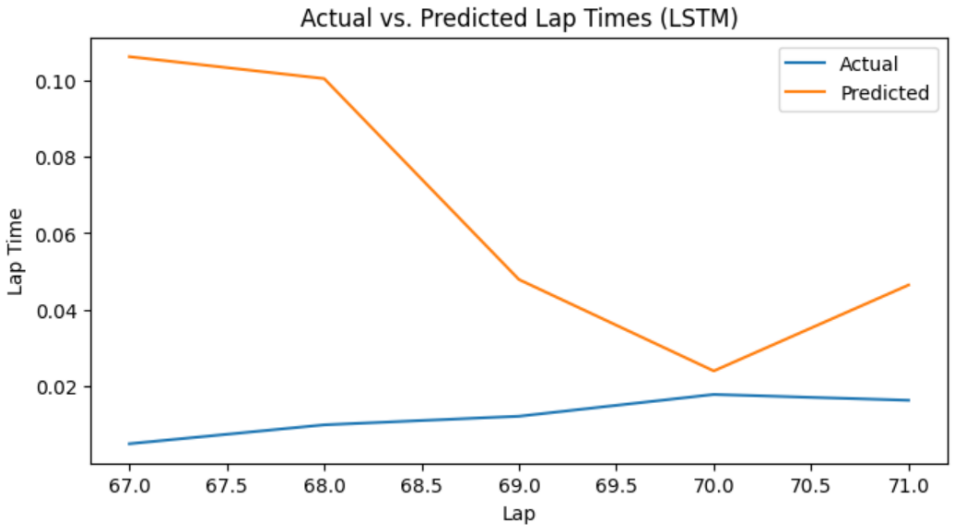
window_sizes = [5, 10, 15, 20]

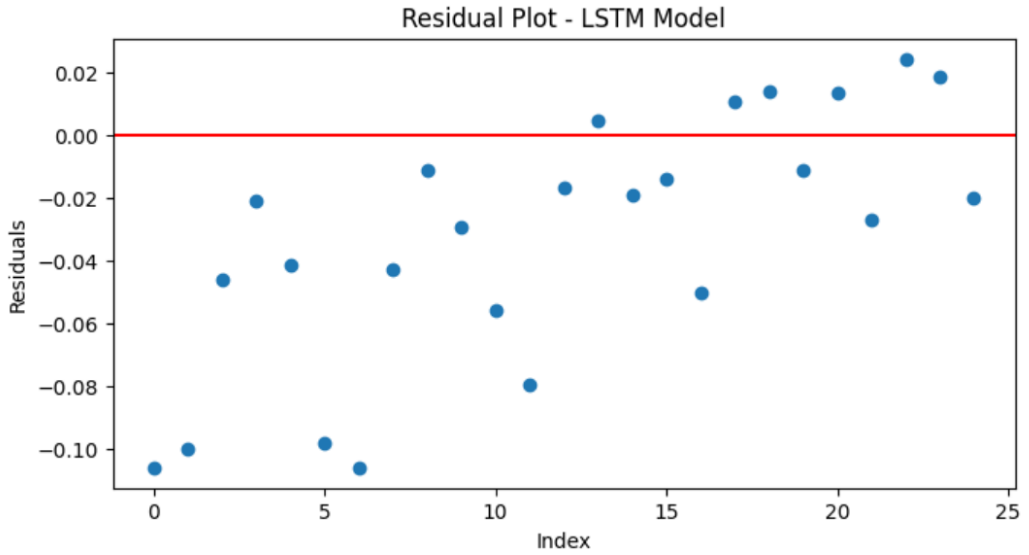
Best Window Size = 20
Best RMSE = 0.0393

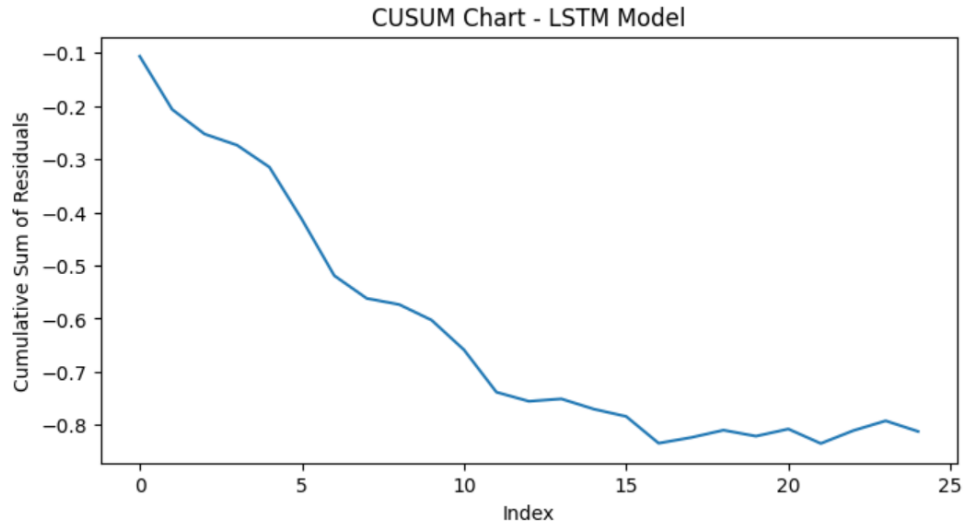
Test RMSE = 0.0641
Test MAE = 0.0526

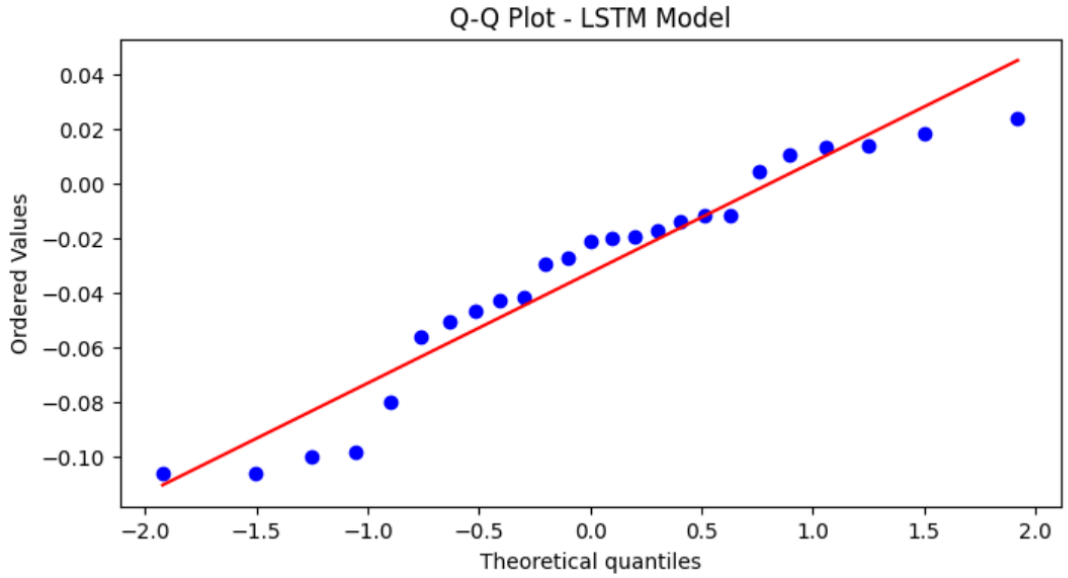


LSTM Model Results

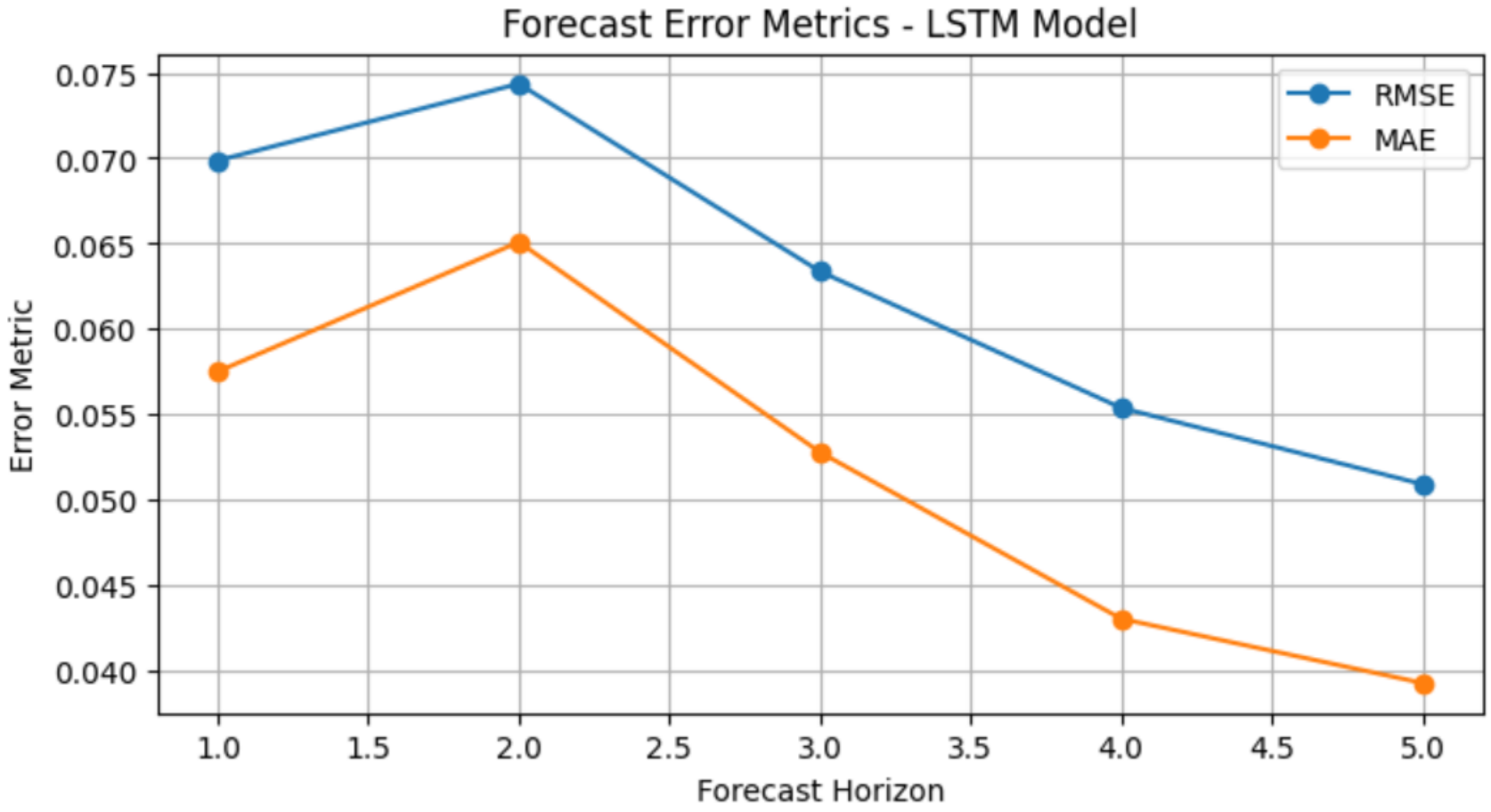








LSTM Model Results



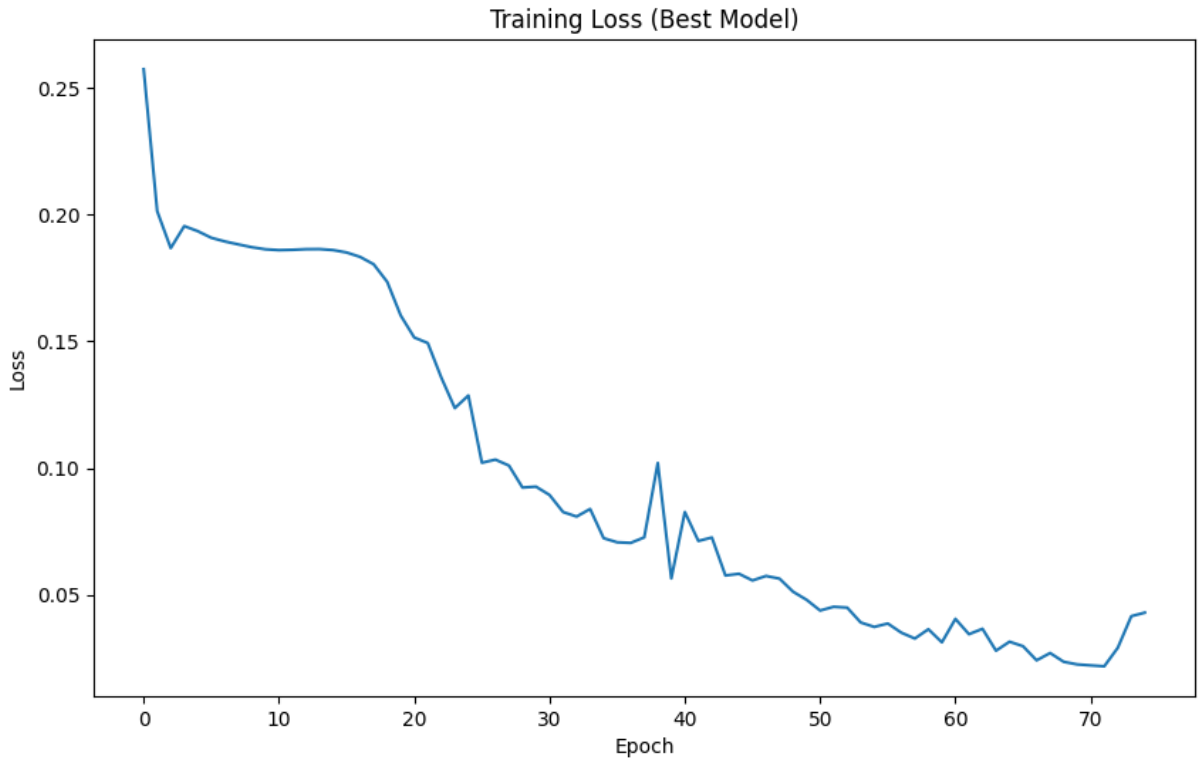
LSTM Model Tuning (Optuna)

hidden_dim = 52
num_layers = 4
learning rate = 0.003

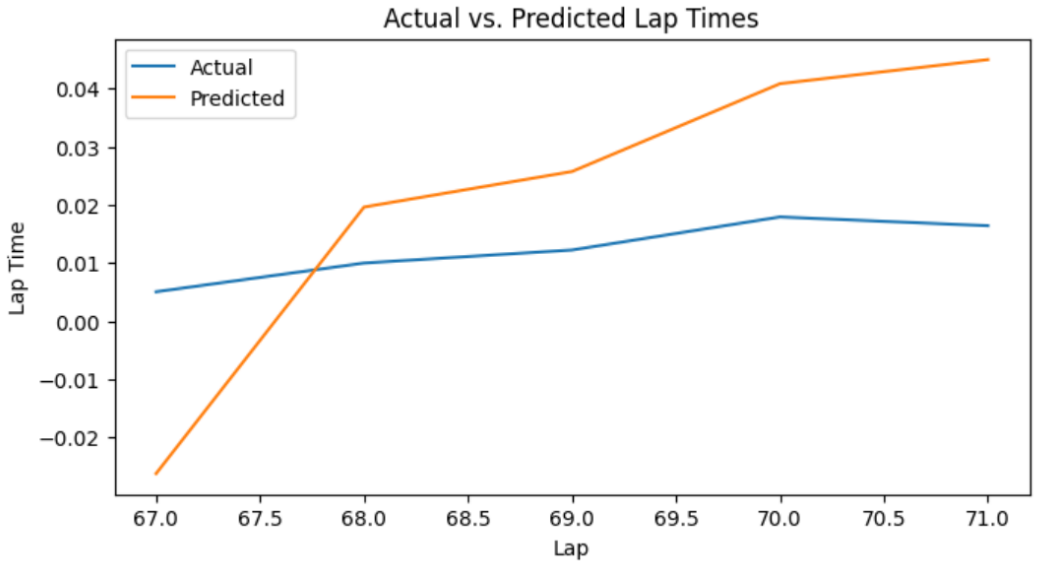
num_epochs = 100
batch_size = 16

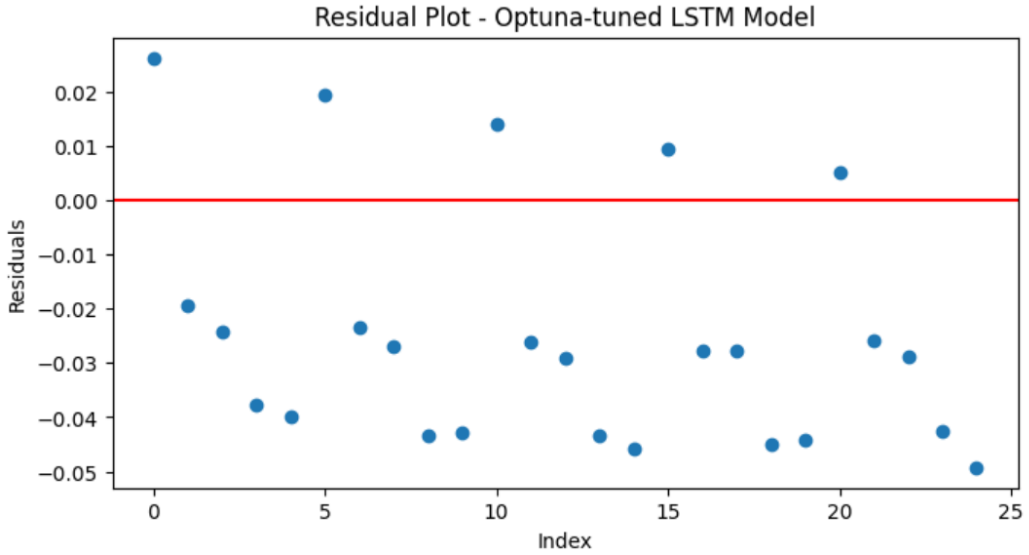
Best Window Size = 19
Best RMSE = 0.0167

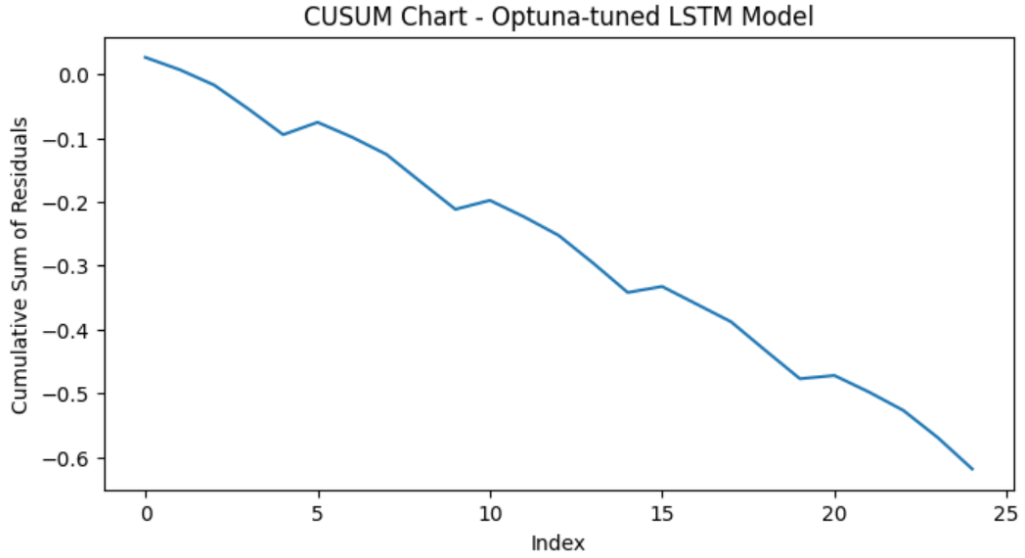
Test RMSE = 0.0227
Test MAE = 0.02117

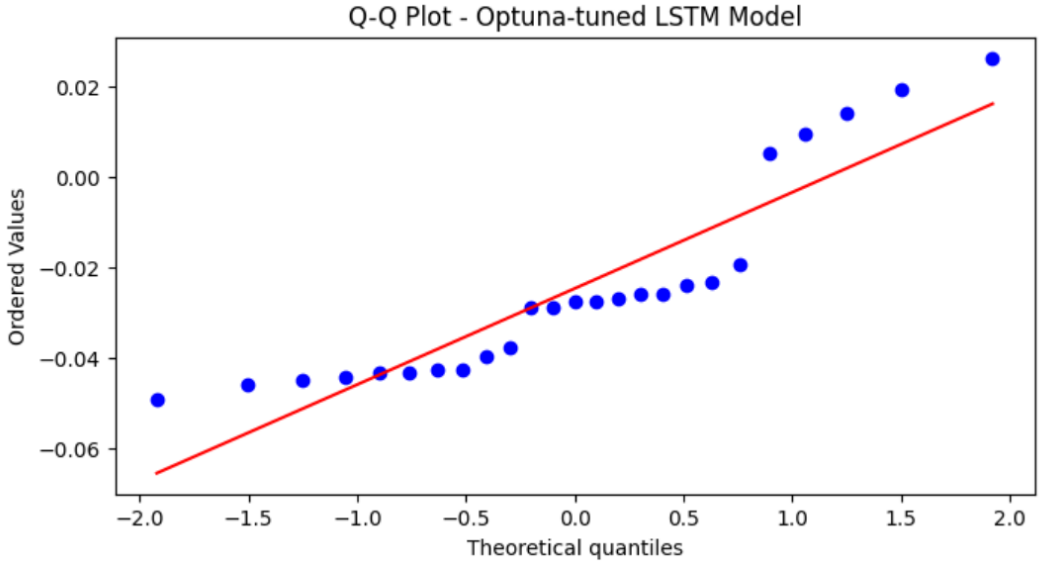


LSTM Model Results (with hyperparameter tuned - Optuna)

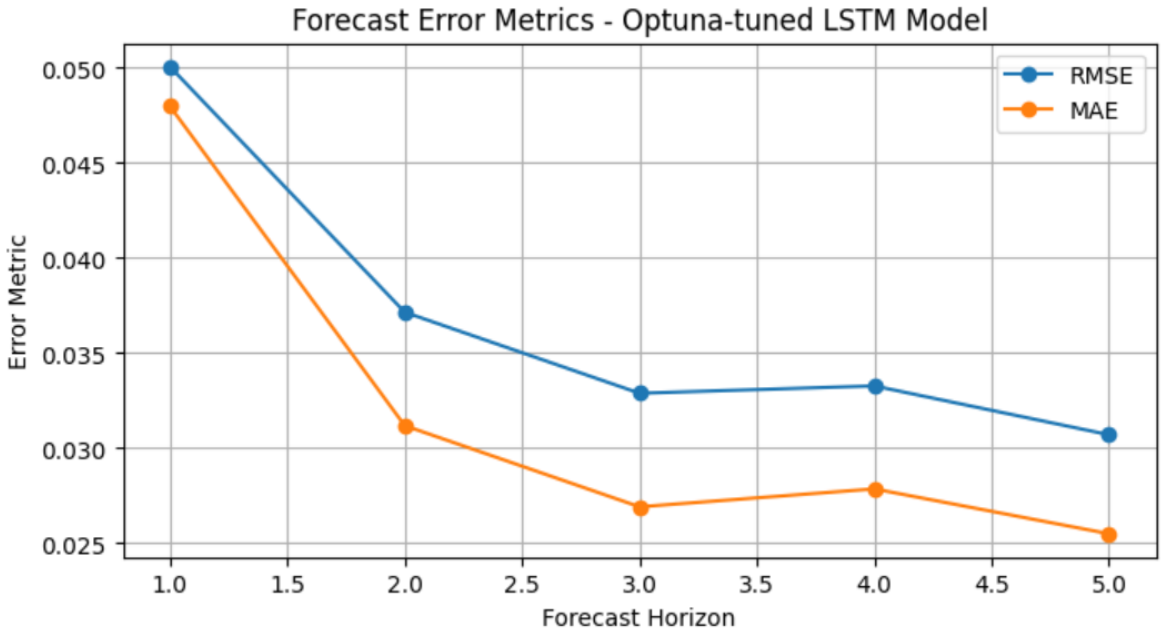


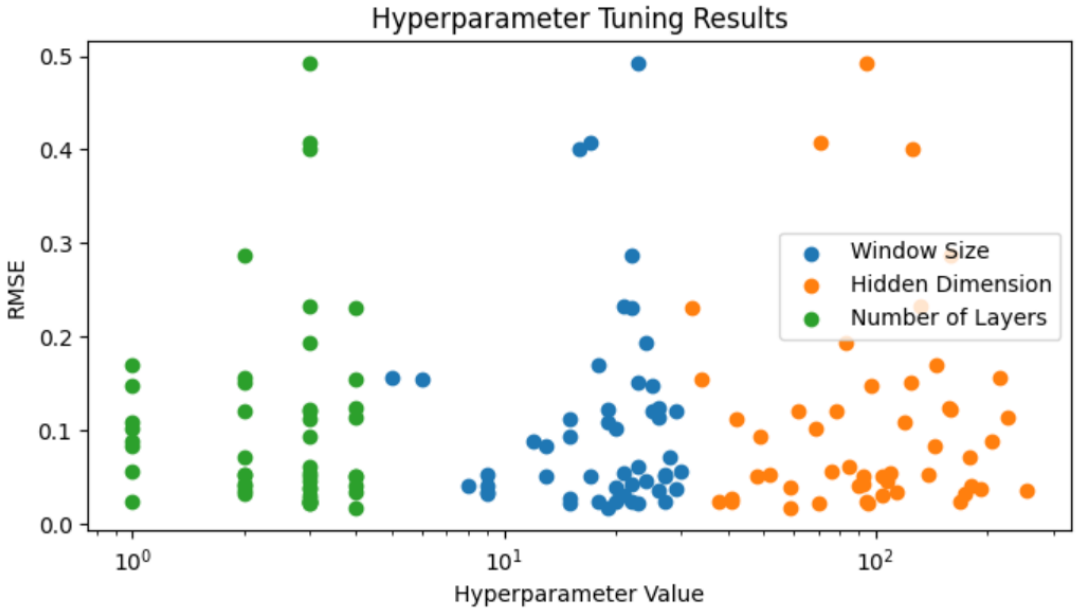






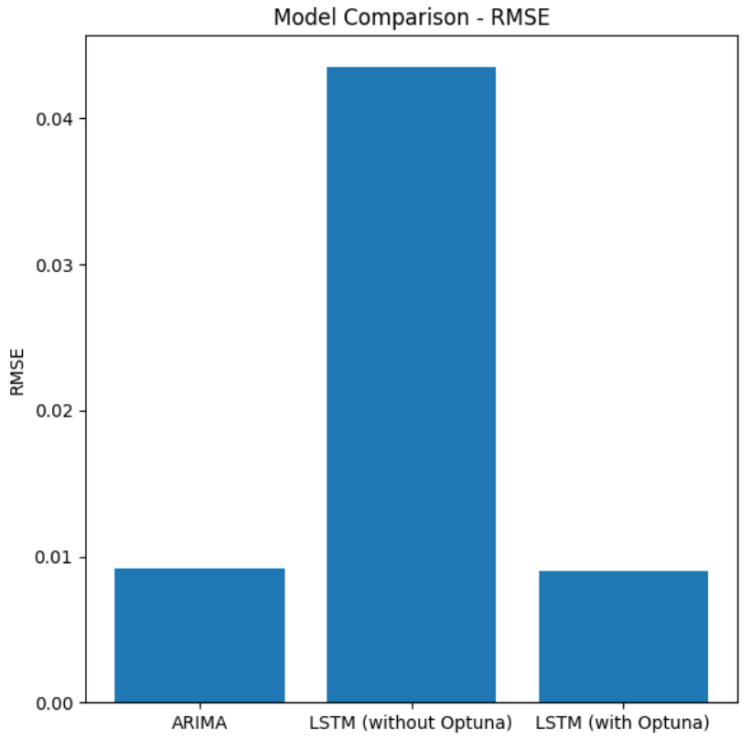
LSTM Model Results (with hyperparameter tuned)

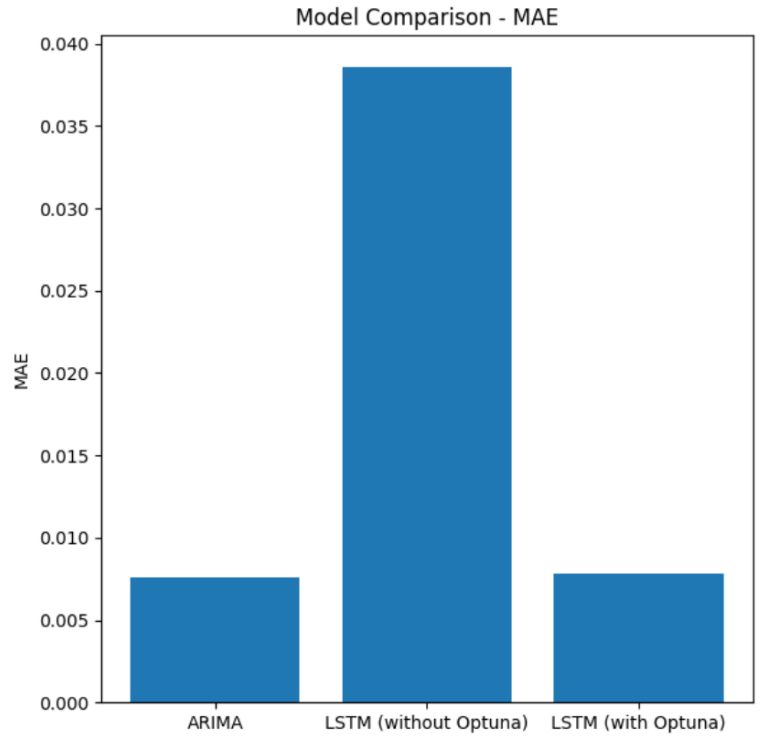




Hyperparameter Tuning results from Optuna

Model Comparison





Model	RMSE	MAE
ARIMA	0.0091	0.0075
LSTM	0.0434	0.0385
LSTM (Optuna Tuned)	0.0089	0.0078

Summary

- ***Stationarity*** - Any non-stationarity was addressed through differencing
- ***Logical feature elimination*** - Irrelevant features were logically eliminated from the initial dataset by removing columns that did not vary with changing lap numbers or provide useful information about lap times
- ***Feature engineering*** - New variables like lap time differences, rolling averages, and rolling standard deviations were created to capture trends, seasonality, and variability in lap times. Pit stop indicators and durations were also incorporated to better explain periodic lap-time spikes
- ***Normalizing and de-normalizing data*** - Min-max normalization was applied to scale the features to a common range, Use of window size and its tuning - For the LSTM model, the optimal window size (number of previous lags) was determined through systematic tuning, evaluating performance across different window lengths
- ***Hyperparameter tuning with Optuna*** - Automated hyperparameter tuning using Optuna library was employed for the LSTM
- ***RMSE and MSE metrics*** - Model accuracy was evaluated using the RMSE and MAE on the held-out test set. RMSE penalizes larger errors more heavily, while MAE is more interpretable and robust to outliers

Summary

- Most of the data series were stationary
- *RMSE* and *MAE* have lower values for ARIMA vs LSTM but similar when we tune LSTM with Optuna
- ARIMA is much faster than LSTM in terms of convergence and fitting
- LSTM might perform better with more data
- LSTM hyper parameters were optimized with Optuna rather than Grid Search

Thank You