Sustainable Computing

**Tianwei Lu**

Tianwei.lu@seh.ox.ac.uk

**Abstract**

This report introduces two methods for carbon intensity forecasting for time periods of one month and one hour. Their accuracies and runtimes are compared. After these explorations, 3 different system performance optimization techniques are implemented to see the code speed up. Using a function to monitor the energy consumption and carbon emissions of the CI forecasting function. The carbon emissions of the whole project code are evaluated, the impact of the optimization is explored, finally case studies using three different locations and three different times are done.

1. **Methodology**
   1. **Procedure**

**Visualisations and initial processing**

Some visualisations are conducted and fitting a linear and an exponential decay curve are conducted, which indicate poor results, expected as the visualisations indicate non-linearities in the dataset. (I have used copilot here to check some syntaxes and used its autofill functions). It can be observed that the variance does not increase over time, hence no need for a log transformation in SARIMA forecasting.

**Monthly SARIMA model**

We define some reasonable search space for the *auto\_arima* from *pmdarima* library fitting on all monthly datapoints in the dataset.

**Monthly MLP model**

Input layer: ;

Hidden layers: 64 relu activated neurons; 32 relu activated neurons

Optimizer: Adam, minimising the MSE for validation dataset.

Data splitting: Train: 2009 to 2023; Validation: 2023 to 2024; Test: 2024 to 2025

**Hourly SARIMAX model**

For the sake of tractability and lack of computational power in the equipment, the length of the dataset which the optimizer operated on had to be reduced to the last 7200 datapoints, automated testing for differencing orders using *adf* and *ocsb* are done to save search space.

**Hourly MLP model**

Using a similar MLP model,

**Input Layer:** hour, day of the week, month, day of year, year,

**Hidden Layers:** **Layer 1:** 100 neurons, **Layer 2:** 50 neurons.

**Reading in bulk**

**SARIMA**

The *auto\_arima* parameter fitting method can be ran using multi-threads, so I ran it using 4 CPU threads, setting stepwise to *False* for parallel processing.

**MLP**

Multithreading can be applied at the feature engineering step, the model training step uses the *SciPy* and *NumPy* libraries that already implement multi-threading optimally.

**Operating on smaller datasets:**

It has been decided that data for time period 2009 to the end of 2015 to be irrelevant, due to the observable different shape that the years’ CI take. We will remove the data points that are considered irrelevant to perform the forecasting methods.

**Improving bottlenecks:**

For both methods and both time frames, there are two main sections that are consistently taking the majority of run time, importing the csv file and fitting the models. Optimization on these two aspects are to be conducted.

The csv file can be directly compiled to a more efficient form such as parquet or data frame. Both hourly and monthly forecasts can be merged into one single file so that the *resample* method to month can be conducted on the hourly dataset instead of the original dataset.

The fitting for SARIMA can be accelerated by not using the seasonality option in the auto tuning function but using Fourier variables, thanks to this method, the model is capable of converging even when using the full subset from 2015 to 2024 as training set.

MLP model can be accelerated by increasing the batch size.

Improvements from the previous optimisation methods are also included.

* 1. **Data sets**

The CI dataset for forecasting is one downloaded from canvas, which is downloaded from https://carbonintensity.org.uk

The real time carbon intensity data is accessed using https://api.carbonintensity.org.uk.

1. **Results and Discussion**

**Monthly forecasts**

A graph showing a graph of carbon

AI-generated content may be incorrect.

Figure Monthly SARIMA model forecast (no optimization)

A graph showing a line

AI-generated content may be incorrect.

Figure Monthly MLP model forecast (no optimization)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MAE** | **RMSE** | **Run time** |
| **SARIMA** | 18.09 | 20.36 | 22.8302 |
| **MLP** | 14.32 | 18.39 | 1.7061 |

**Hourly forecasts**

A graph showing a graph of a wave

AI-generated content may be incorrect.

Figure Hourly SARIMA forecast**A graph showing red lines

AI-generated content may be incorrect.**

Figure Hourly MLP forecast

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MAE** | **RMSE** | **Run time (s)** |
| **SARIMA** | 58.16 | 74.26 | 103.64 |
| **MLP** | 6.39 | 8.45 | 64.28 |

**Multithreading**

**Monthly**

**SARIMA** Yields exactly the same result

**MLP** Better run time and exactly the same result

**A graph showing a line

AI-generated content may be incorrect.**

Figure Monthly MLP forecast using multi-threading

**Hourly**

A graph showing a green line

AI-generated content may be incorrect.

Figure SARIMA hourly forecast using multithreading

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MAE** | **RMSE** | **Run time** |
| **Monthly SARIMA** | 18.09 | 20.36 | 20.8060 |
| **Monthly MLP** | 15.21 | 19.5 | 1.8597 |
| **Hourly SARIMA** | 64.43 | 83.05 | 26.32 |
| **Hourly MLP** | 6.39 | 8.45 | 59.8507 |

**Smaller datasets**

**Monthly**

**A graph showing a graph of carbon

AI-generated content may be incorrect.**

Figure Monthly SARIMA forecast (subset of data)

A graph showing a graph

AI-generated content may be incorrect.

Figure Monthly MLP foreast (subset)

**Hourly**

**A graph with red lines

AI-generated content may be incorrect.**

Figure Hourly SARIMA forecast (subset of data: 20 days)

A graph showing a graph of red and green lines

AI-generated content may be incorrect.

Figure Hourly MLP forecast (subset of data)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MAE** | **RMSE** | **Run time** |
| **Monthly SARIMA** | 24.13 | 27.22 | 10.2212 |
| **Monthly MLP** | 9.23 | 11.58 | 1.3364 |
| **Hourly SARIMA** | 182.27 | 215.06 | 117.74 |
| **Hourly MLP** | 6.38 | 8.42 | 55.9615 |

**Bottleneck improvements**

The parquet conversion and filtering takes 0.7824 seconds

**Hourly**

A blue and orange lines

AI-generated content may be incorrect.

Figure Hourly forecast SARIMA

A line graph with numbers and a line graph

AI-generated content may be incorrect.

Figure Monthly SARIMA forecast (full on optimization)

A graph showing a graph of red lines

AI-generated content may be incorrect.

Figure Hourly MLP forecast (fully optimized)

A graph showing a graph

AI-generated content may be incorrect.

Figure Monthly MLP forecast (fully optimized)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MAE** | **RMSE** | **Run time** |
| **Hourly SARIMA** | 66.99 | 84.43 | 31.72 |
| **Monthly SARIMA** | 25.96 | 28.97 | 14.86 |
| **Hourly MLP** | 6.48 | 8.55 | 49.39 |
| **Monthly MLP** | 19.52 | 25.03 | 1.37 |

1. **Conclusions**
2. **Computer Hardware and Programs**

Processor 12th Gen Intel(R) Core(TM) i5-1235U (1.30 GHz)

Installed RAM 16.0 GB (15.7 GB usable)

System type 64-bit operating system, x64-based processor

Pen and touch No pen or touch input is available for this display

IDE: visual studio code

1. **References and acknowledgements**

https://api.carbonintensity.org.uk