MidTerm Project

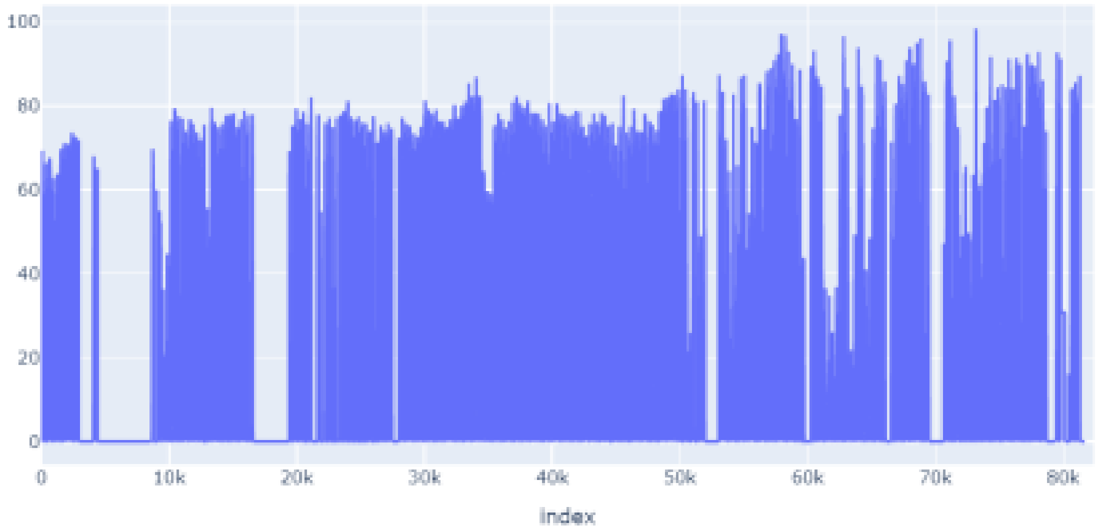
Due date : Jun 23, 2023, 23:59 Hrs.

Data to be used: e7-htr-current.csv

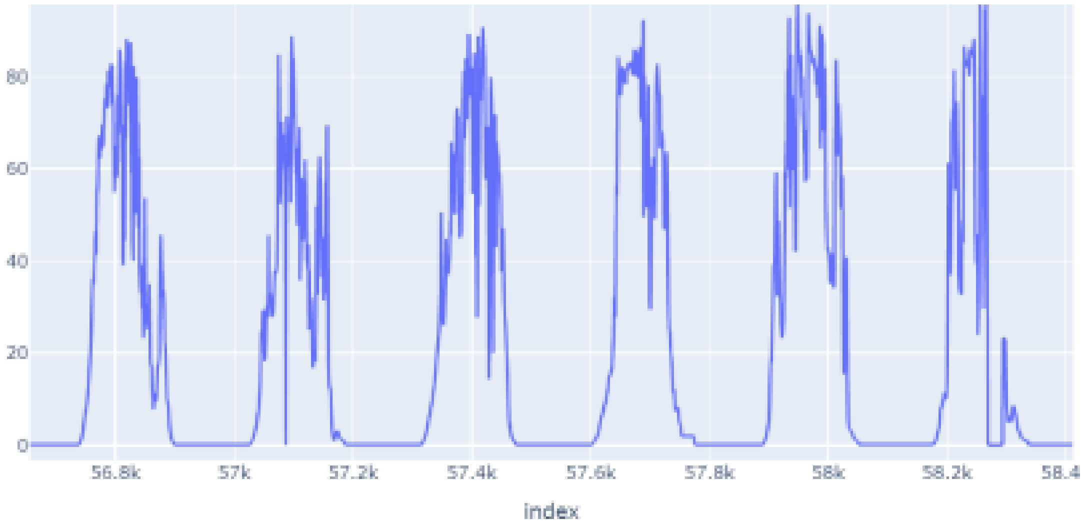
Data Description

- Transformer current data, sampled every 5 minutes
- Data source: a solar power generation site

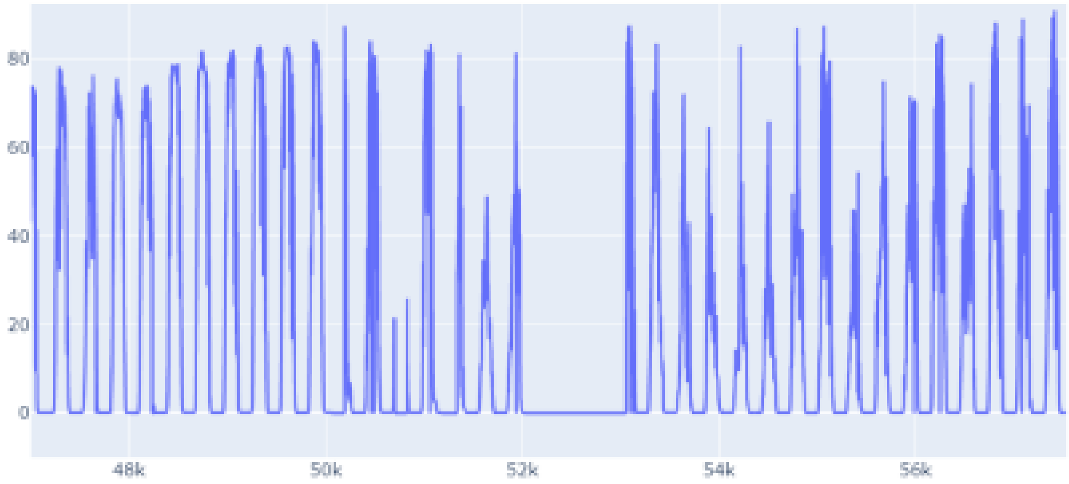
Line plot of all data points (the complete data – about 280 days).



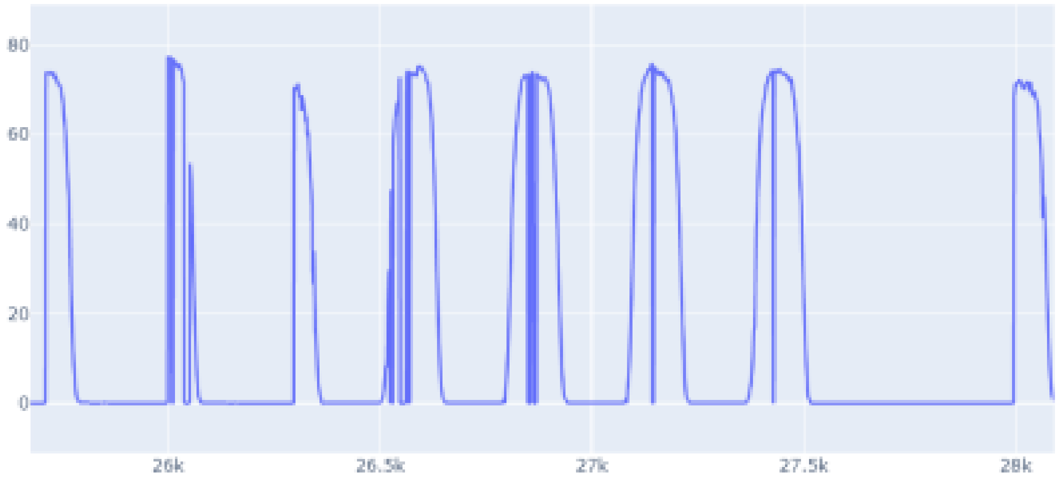
Days on which the data was choppy (bad) due to various reasons



Days on which data could not be sampled due to various reasons



Relatively good days! There may be slightly noisy readings, but seemingly easy to fix.



The challenge:

- Can the data be “de-noised”? That is, can the data set’s quality be improved with the help of the data itself?

The task: Data Quality Improvement

1. Identify the ‘good’, ‘bad’ and ’missing’ days
2. Make the ‘good days’ ‘really good’, by fixing the observed data problems
3. Use data from about 80% of the ‘good’ days to create an ML model.
4. Validate the model on the remaining 20% of the “good” days (How will you validate? What metrics will you use for validation?)
5. Use the ML model thus created to “fix” the data from the “bad” and “missing” days
6. Create appropriate metrics, and a plot of the “improved data set”, and compare it – using the metrics, and visually - with the original.

Suggestions and hints:

- Understand and use the Python library **plotly** for interactive data exploration. • The data covers about 280 days of operation of the transformer. Treat the data for one day as one set of observations – so you have about 200 sets of observations (after accounting for completely missing data) that can be used to predict the ‘current’ through the transformer at any given ‘active’ time of the day.

- What independent features will be required to predict the ‘current’ through the transformer at any given ‘active’ time of the day? Feature Engineering required?
- How can you automate the process of identifying ‘good’, ‘bad’ and ‘missing data’ days? • How can you use **descriptive statistics** to assist in some of the analysis steps?

***