Project Assignment – Sustainable Energy Technology Track

Fault Detection for Residential PV Systems

Responsible Lecturer: Malte Vogt (M.R.Vogt@tudelft.nl)

Context

With the ongoing energy transition towards a more sustainable future, and the ever-increasing energy demand, the use of photovoltaic (PV) modules will increase. One advantage over other renewable energy sources is the applicability to be easily installed at smaller scale, on household rooftops. However, in these residential PV systems proper monitoring is mostly lacking, and only AC output power is known. Due to the lack of monitoring, defects in such systems can often go undetected for longer periods of time resulting in decreasing system efficiency and reliability. A variety of faults could occur such as open-circuit faulting due to loose cabling. Sometimes this is accompanied by a reduction in power output that without careful examination is hardly distinguishable from the natural power output variations due to weather variability, such as irradiance affecting current output and module temperature affecting voltage output. Without early acting these faults could develop over time which could ultimately cause fire or electrocution hazards. In this Project Assignment, you will develop an ML model that uses PV system power output to detect malfunctioning PV systems.

Purpose

By developing this assignment, you will put into practice all the learning concepts introduced so far in Lectures 1 to 4. You will develop and train ML algorithms on the data provided and use metrics to evaluate and compare their performance. This assignment covers this course learning objectives (LOs): LO3, LO4 and LO5.

Resources

You can consult with the TAs on Practicums on Week 5 and 6. You will receive feedback from a Lecturer on Week 7.

Instructions

Activities

You will be using PV data that has been generated artificially, through fault simulations. A large variety of faulty and regular (healthy) conditions has been simulated for a residential PV system. Using binary classification, the faulty system conditions can be separated from the healthy system conditions. With multiclass classification it is possible to determine the type of fault that is causing a system to malfunction.

System power output is the most important indicator as it is the main output of the system and oftentimes the only information available for a residential PV system. Additionally, the different meteorological conditions like incident solar irradiance, sun position and ambient temperature can be utilized to learn complex relations between these features. Another aspect of the system which plays a role is system age, as different fault types are more likely to occur at specific moments of the system lifetime.

The dataset provided consist of 12000 scenarios, where each scenario provides a snapshot of the weather conditions and power output. This includes 6000 regular non-faulty conditions, 2000 short-circuit conditions, 2000 (multiple) broken cell(s) conditions, 2000 broken module string conditions. Column 1-6 contain the features for this assignment. In column 7 the system status is indicated (0=healthy, 1=faulty) and column 8 gives the specific fault type (0=healthy, 1=short-circuit, 2=broken cells, 3=broken string).

Three tasks are planned for this Project Assignment: Data pre-processing, fault detection, fault classification.

Task 1
Perform data pre-processing activities, this may include data preparation, data cleaning, normalization, standardization, and feature analysis.

Task 2
Develop a binary classification model that allows you to detect faulty systems for new unseen operational scenarios. Provide arguments for the choice and design of your final model(s). Provide arguments for the assessment and validation procedure of your final model(s).

Task 3
Develop a multiclass classification model that allows you classify the operational state of a system (healthy or specific fault type) for new unseen operational scenarios. Provide arguments for the choice and design of your final model(s). Provide arguments for the assessment and validation procedure of your final model(s).

Other instructions
- You will work in pairs.
- Decisions need to be made together, but Tasks can be done individually.
- We recommend splitting the tasks. Any member must be capable of arguing any decision made.
- At least one of the models must be a *deep neural network (Lecture 6)***.**
- One report per pair. The report must follow the proposed structure with a maximum number of pages of 10.
- Deadline: Week 9.

Deliverables
1. Final Project Report (see instructions below)
2. Project Assignment Python code

Report Structure
- Members, emails, student numbers.
- Summary (less than 200 words)
- Detailed ML pipeline (include workflow figure).
- Task 1: selected options, argumentation for the selection, model(s) developed, results, validation, comparisons.
- Task 2: argumentations for the model(s) developed, validations, results, comparisons.
- Task 3: argumentations for the model(s) developed, validations, results, comparisons.
- Conclusions (less than 200 words)

Assessment Criteria
You will be evaluated based on a predefined rubric. Check the course Brightspace page to get access to the rubric.

The Project Final Report can be considered *inadmissible*, which will render a FAIL grade for the group, if
- English is not understandable (e.g., full of typos).
- Deep neural networks were not used (as one of the tested models).
- Figures are not legible.
- The report does not follow the proposed structure.

If the report is considered *admissible*:
- English will *not* render extra points.
- Quality of the Python code will *not* render extra points.

Submission Instructions
Please submit your Final Project Report in a PDF format and your Python code in Brightspace before the deadline.