
WinFault Documentation

Release 1.0

Kevin Leahy

July 19, 2016

1	Contents	3
1.1	Note on the SCADA, Status and Warning Data	3
1.2	winfault module	4
1.3	Usage Example	4

This is a module which introduces the *WT_data* class, which enables you to import wind turbine SCADA data. From there, the fault-free SCADA data can be extracted, as well as various types of faults, according to separate status and warning data. This is then labelled as such and split into training and testing data for fault classification and prediction.

1.1 Note on the SCADA, Status and Warning Data

1.1.1 General Notes

Make sure the status, warning and SCADA data all cover the same time period. If this is not the case, some SCADA data could be labelled as having a fault/fault-free status that it does not actually have.

1.1.2 SCADA Data

The CSV file for this data is located at “Source Data\SCADA_data.csv”. It contains time-stamped operational data for the turbine. The timestamps are at 10 minute intervals which represent the average of the sensor readings over that period.

1.1.3 Status Data

There are a number of normal operating states for the turbine. For example, when the turbine is producing power normally, when the wind speed is below cut-in, or when the turbine is in “storm” mode, i.e., when the wind speeds are too high for normal generation.

There are also a large number of statuses for when the turbine is in abnormal or faulty operation. These are all tracked by status messages, contained within the “Status” data. This is split into two different sets:

- WEC status data (“Source Data\status_data_wec.csv”)
- RTU status data (“Source Data\status_data_rtu.csv”)

The WEC (Wind Energy Converter) status data corresponds to status messages directly related to the turbine itself, whereas RTU data corresponds to power control data at the point of connection to the grid, i.e., active and reactive power set points.

Each time the WEC or RTU status changes, a new timestamped status message is generated. Thus, the turbine or RTU is assumed to be operating in that state until the next status message is generated. Each turbine/RTU status has a “main status” and “sub-status” code associated with it.

Any main WEC status code above zero indicates abnormal or faulty behaviour, however many of these are not associated with a fault, e.g., status code 2 - “lack of wind”.

The RTU status data almost exclusively deals with active or reactive power set-points. For example, status 100 : 82 corresponds to limiting the active power output to 82% of its actual current output.

1.1.4 Warning Data

The “Warning” data on the turbine mostly corresponds to general information about the turbine, and usually isn’t directly related to turbine operation or safety. These “warning” messages, also called “information messages” in some of the turbine documentation, are timestamped in the same way as the status messages, and also have a “main warning” and “sub-warning” code associated with them.

Sometimes, warning messages correspond to a potentially developing fault on the turbine; if the warning persists for a set amount of time and is not cleared by the turbine operator or control system, a fault is raised and a new status message is generated.

The warning data is also split into WEC and RTU warning data:

- WEC warning data (“Source Data\warning_data_wec.csv”)
- RTU warning data (“Source Data\warning_data_rtu.csv”)

1.2 winfault module

1.3 Usage Example

Import using `import winfault`.

Create the class, and access correctly formatted SCADA data:

```
Turbine = winfault.WT_data()
scada_data = Turbine.scada_data
fault_free_data = Turbine.fault_free_scada_data
```

Access columns by typing `data['column name']`. The following gets the first 100 windspeeds of the `fault_free_data`:

```
fault_free_data['WEC_ava_windspeed'][0:99]
```

Get all the fault data using default values (type `help(Turbine.filter)` or `help(Turbine.get_all_fault_data)` for details):

```
all_faults_scada_data, feeding_fault_scada_data, \
mains_failure_fault_scada_data, aircooling_fault_scada_data, \
excitation_fault_scada_data, generator_heating_fault_scada_data = \
Turbine.get_all_fault_data()
```

Generate training and testing data for fault-free, feeding fault and mains failure faults, using specific features:

```
features = ['Time',
            'WEC_ava_windspeed',
            'WEC_ava_Rotation',
            'WEC_ava_Power',
            'WEC_ava_reactive_Power',
            'WEC_ava_blade_angle_A',
            'Inverter_averages',
            'Inverter_std_dev']

fault_data_sets = [feeding_fault_scada_data, mains_failure_fault_scada_data]
X_train, X_test, y_train, y_test, X_train_bal, y_train_bal = \
Turbine.get_test_train_data(features, fault_data_sets, fault_free_data,
                            True, 0.2)
```

genindex