Validation of Numerical Model Improvements through Public Data Sets and Code

Caroline Draxl,[1] Joseph Lee,[1] Larry Berg,[2] Will Shaw[2]
[1]National Renewable Energy Laboratory (NREL)
[2]Pacific Northwest National Laboratory (PNNL)

Virtual Wind Integration Workshop, November 11–12, 2020

iea wind

Content

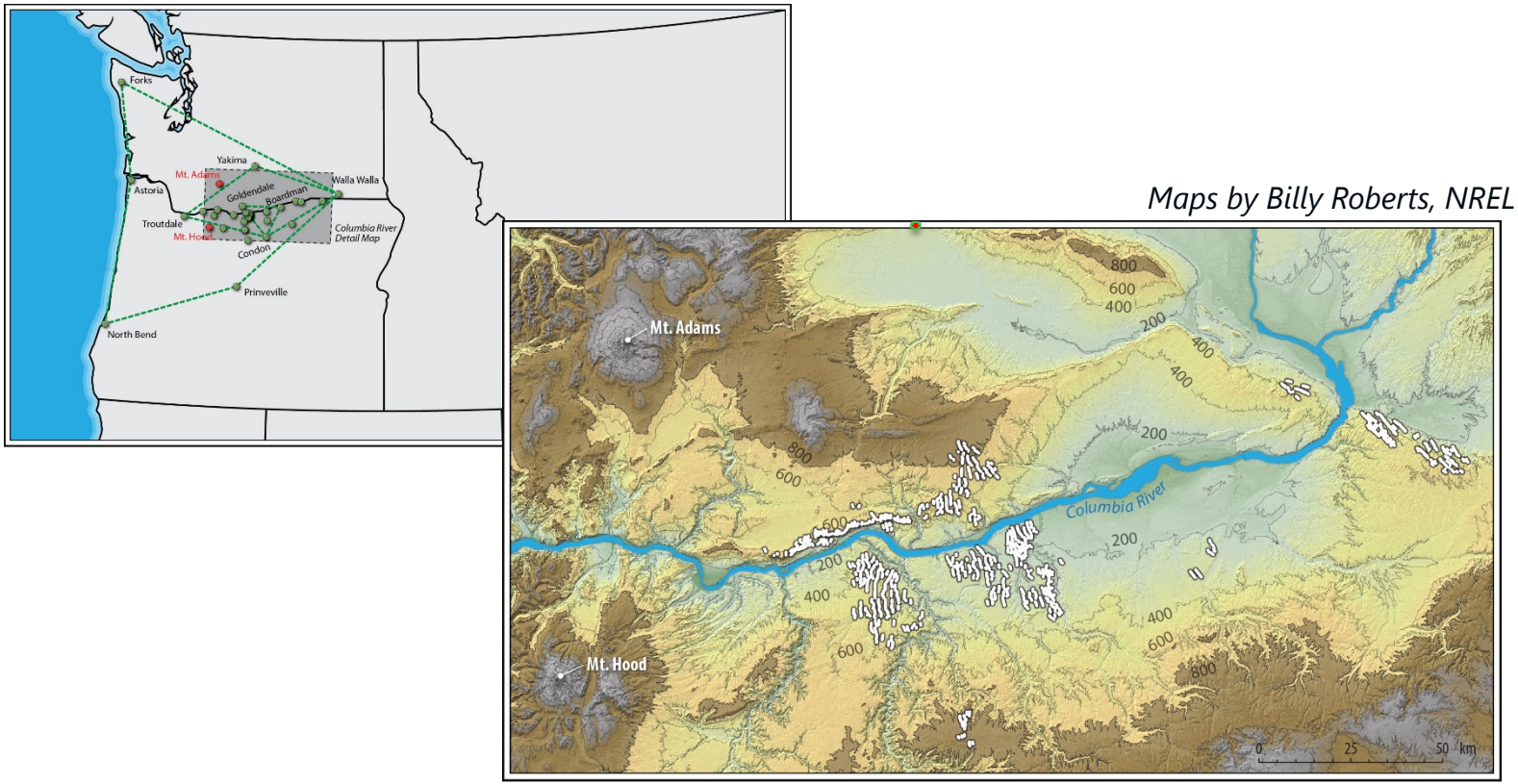
- Verification and Validation (V&V) process of the Second Wind Forecast Improvement Project (WFIP2)

- Introduction to a validation framework developed for IEA Wind Task 36 (Subtask 1.3)

- Solicit feedback regarding:
  - Dissemination of framework
  - Approach, case selection, and usefulness
  - Engaging partners in this effort.

iea wind

Improve Wind Speed Forecasts in the Turbine Rotor Layer in Complex Terrain



Maps by Billy Roberts, NREL

iea wind

WFIP2 Verification and Validation Goals

Provide tools, methods, and guidance to enable repeatable, metrics-based assessment of Weather Research and Forecasting (WRF) and High-Resolution Rapid Refresh (HRRR) models for analysis and forecasting of mesoscale weather phenomena that are important for wind energy in the Columbia River Gorge and the continental United States.

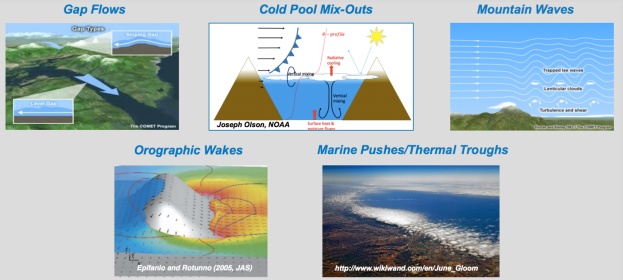
Verification: Verification is concerned with checking the mechanics of the software code rather than checking that the model's physics are correct.

Validation: Validation is determining the degree to which the model represents the real world for a particular application.

iea wind

Validation Approach

WFIP2 weather taxonomy:



Key variables and metrics:

- 80-m wind speed
- Wind power
- Bulk rotor layer statistics (RMSE, bias, MAE, percentage improvement)
- Wind ramp metric.

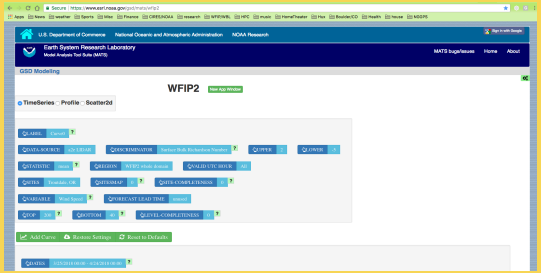
Event log

Regular V&V meetings to discuss and coordinate results

Experiment to Model Analysis Table (EMAT):

- What, where, when?
- What are the dominant physics?
- How do we see this in measurements?
- What are the metrics we should use?

Validation tool and database:

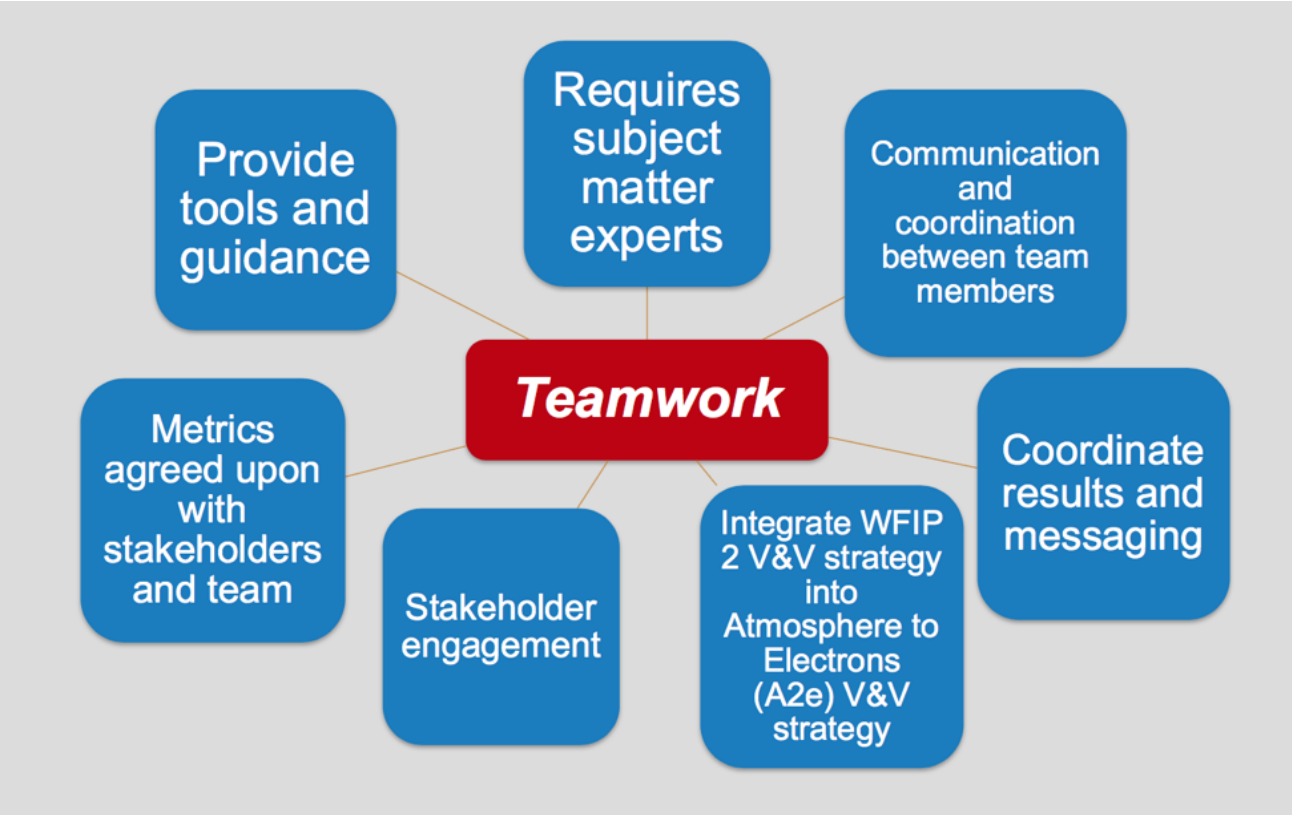


Workshops to compare validation results and test EVS tool

Common case study data set to test validation code

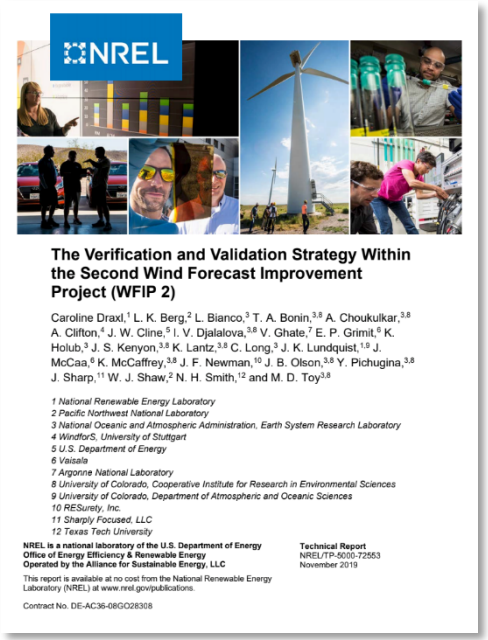
iea wind

Verification and Validation Summary and Conclusion



iea wind

Publications



Olson, J. B., and Coauthors, 2019: Improving Wind Energy Forecasting through Numerical Weather Prediction Model Development. *Bull. Amer. Meteor. Soc.*, **100**, 2201–2220, https://doi.org/10.1175/BAMS-D-18-0040.1.

William L. Oberkampf and Christopher J. Roy. 2010. *Verification and Validation in Scientific Computing* (1st. ed.). Cambridge University Press, USA.

iea wind

Public Validation Code

Experience:

We tested a case study in WFIP2:

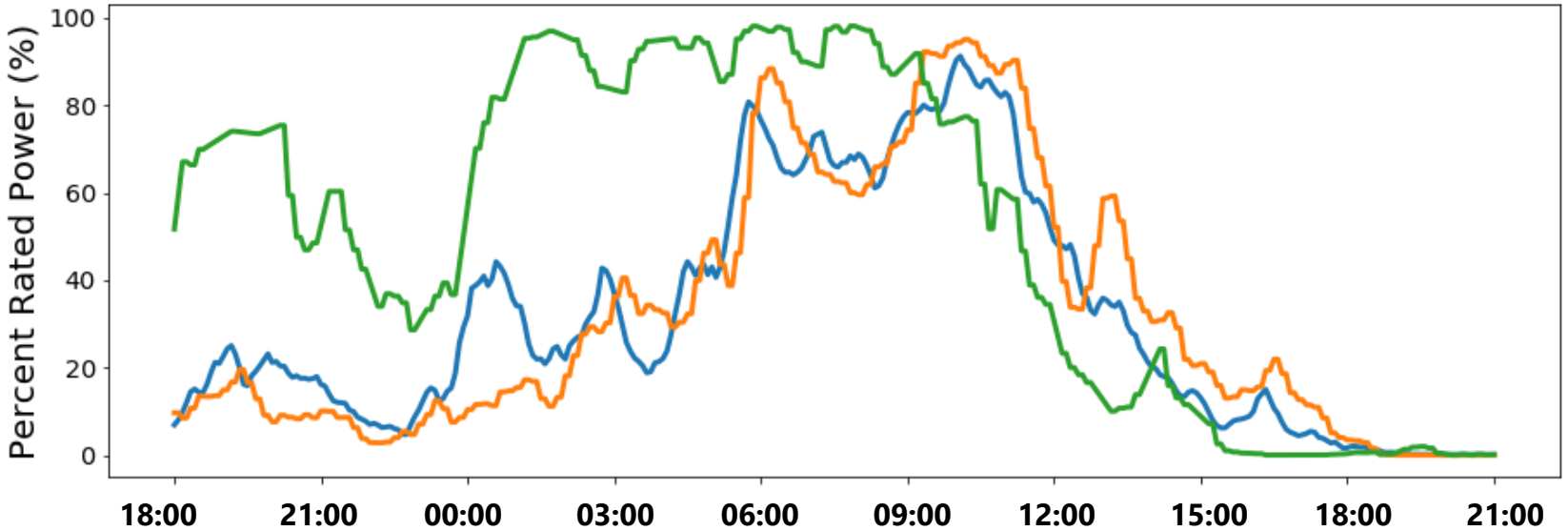
- We provided a model time series with hourly time stamps.
- We told the group the observation location and where to find the observations.
- Everyone used their own scripts to calculate root mean square error (RMSE) and bias.
- Results were provided.
- Different results were obtained because of wrong interpretations of the time stamp.
- And different averaging techniques were used in the horizontal and vertical.

Proposed validation framework:

- Offer a code that can be used by anyone.
- Use data from simulations and observations and use phenomena that are relevant for wind energy.
- Ensure that data are freely available.
- Can be reproduced (use WRF model), but it is also extendable for other NWP model output.
- Evaluate improvement: Provide two model simulations and observations.
- Document the case in the literature.

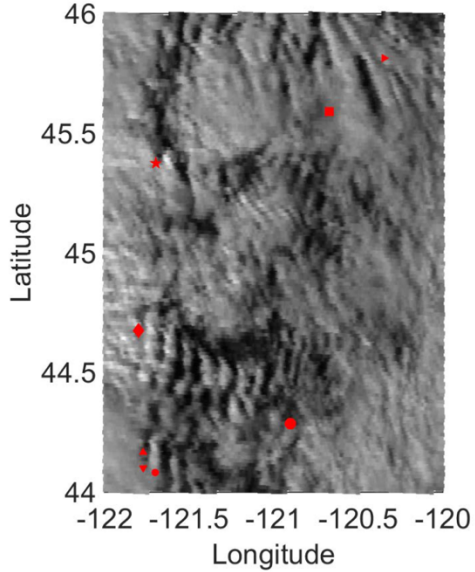
iea wind

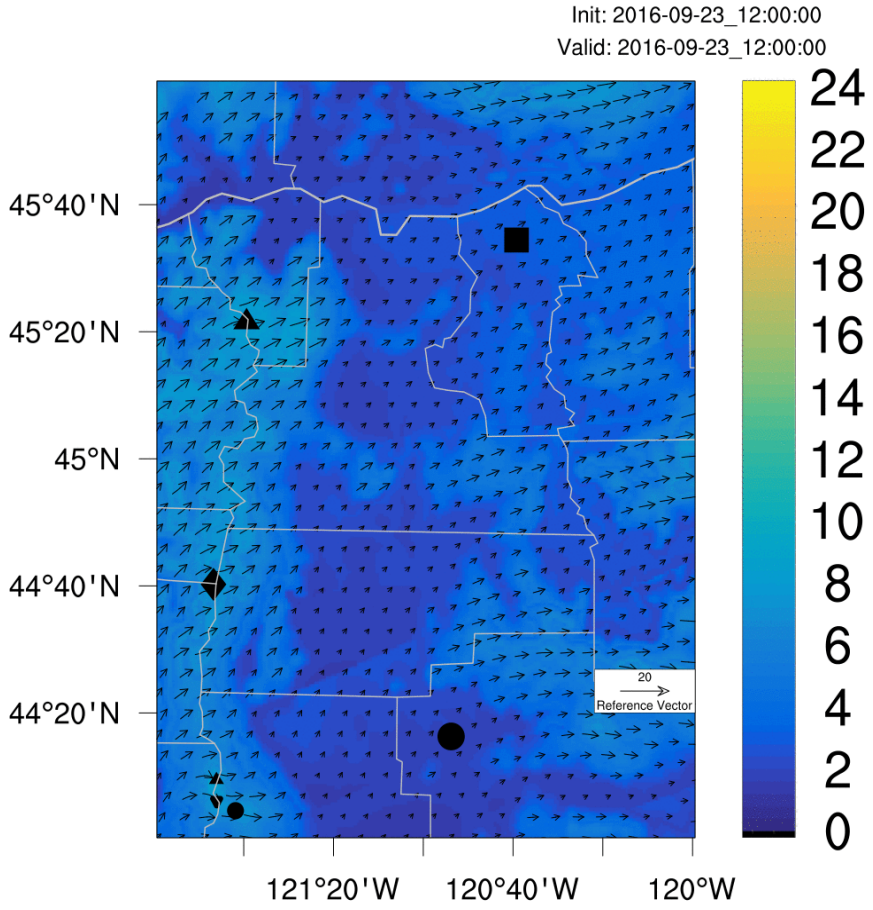
Fluctuations in Power Output from Wind Power Plants



iea wind

Simulated Waves





Draxl, C., et al, Mountain waves impact wind power generation, *Wind Energ. Sci. Discuss.*, https://doi.org/10.5194/wes-2020-77, in review, 2020.

iValidate

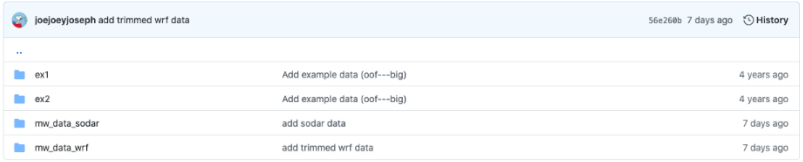
1) Install Python 3
2) Clone GitHub repository to your machine.
3) Edit config.yaml
4) Put data sets in the specified path.
5) Run.

https://github.com/joejoeyjoseph/nwtc-ivalidate/tree/dev

iea wind

joejoeyjoseph / **nwtc-ivalidate**

forked from somerandomsequence/nwtc-ivalidate



joejoeyjoseph add trimmed wrf data — 56e260b 7 days ago — History

..		
ex1	Add example data (oof---big)	4 years ago
ex2	Add example data (oof---big)	4 years ago
mw_data_sodar	add sodar data	7 days ago
mw_data_wrf	add trimmed wrf data	7 days ago

Directory where your data live

Directory with scripts that extract data
(e.g., WRF; can be changed to your data set)

Directory with scripts that calculate metrics

Directory with scripts that perform
quality control

dev | 2 branches | 0 tags | Go to file | Code

This branch is 22 commits ahead of somerandomsequence:master. | Pull request | Compare

joejoeyjoseph add sodar py file — 47f6b2f 7 days ago — **43** commits

data	add trimmed wrf data	7 days ago
inputs	add sodar py file	7 days ago
metrics	add wrf vs sodar mw case	4 months ago
notebooks	update nb	7 days ago
plotting	validate at different heights	4 months ago
prepare	add wrf vs sodar mw case	4 months ago
.gitignore	add gitignore	5 months ago
README.md	Update docs	4 years ago
TODO.md	Separate file for todo	4 years ago
compare.py	validate at different heights	4 months ago
config.yaml	add comments in yaml	3 months ago
i-validate_user_guide_2020-07.pdf	add user guide	3 months ago
output.json	debug compare.py	5 months ago
requirements.txt	add requirements.txt	3 months ago

Metrics Directory

dev ▾ nwtc-ivalidate / metrics / Go to file

This branch is 22 commits ahead of somerandomsequence:master. Pull request Compare

joejoeyjoseph add wrf vs sodar mw case 25e99fb on Jun 17 History

..

README.md	Update docs	4 years ago
__init__.py	Initial version	5 years ago
bias.py	add wrf vs sodar mw case	4 months ago
bias_pct.py	add other metrics	5 months ago
mae.py	add demo nb	5 months ago
mae_pct.py	add other metrics	5 months ago
rmse.py	add other metrics	5 months ago

README.md

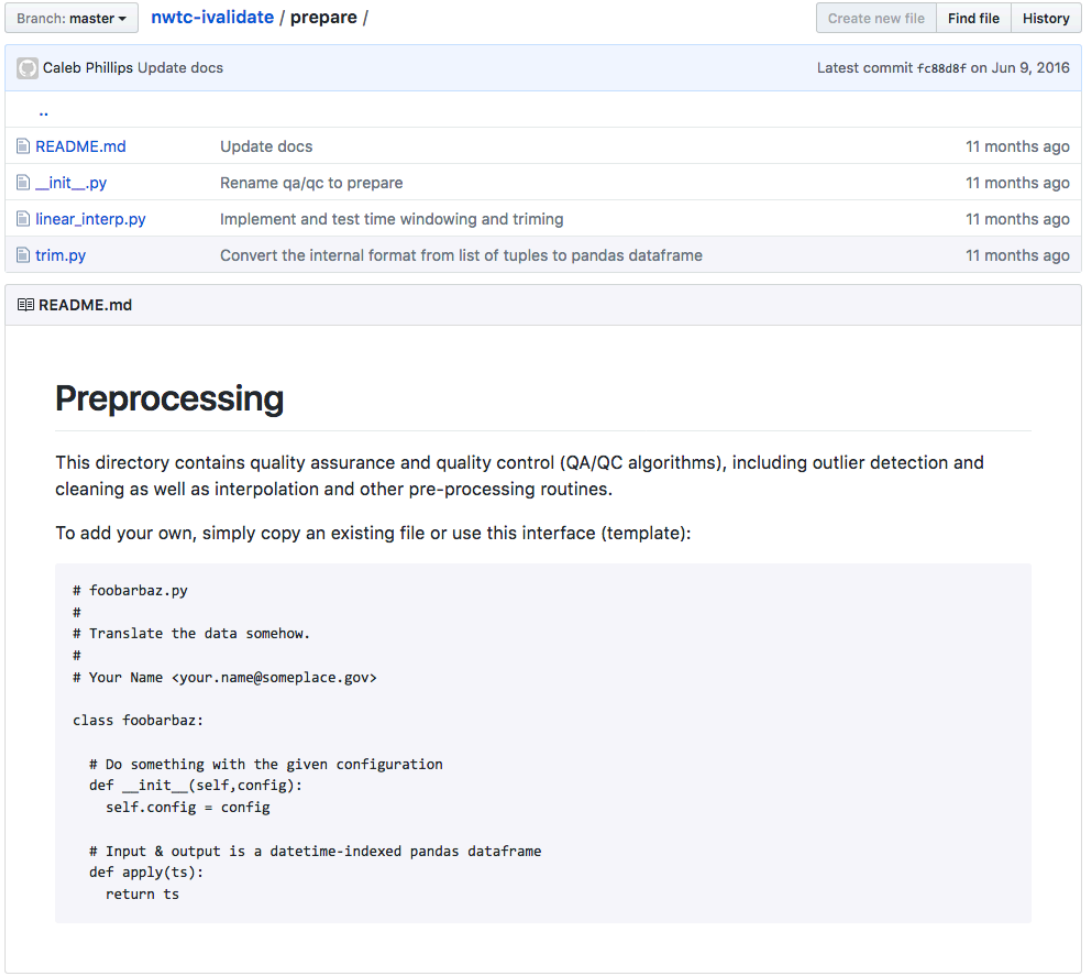
Metrics

This directory contains algorithms to compute timeseries comparison metrics.

To add your own, simply copy an existing file or use this interface (template):

```
# foobarbaz.py
#
# The best metric.
#
# Your Name <your.name@someplace.gov>

class foobarbaz:

  # This function takes two datetime-indexed pandas dataframes as input
  # and returns a single float value
  def compute(self,x,y):
    return None
```

Prepare Directory: Quality Control



Config.yaml:

Defines what you would like to compute

```
python compare.py
config.yaml > output.json
```

Grid point/ location

QC criteria

Time series

Which metrics should be computed

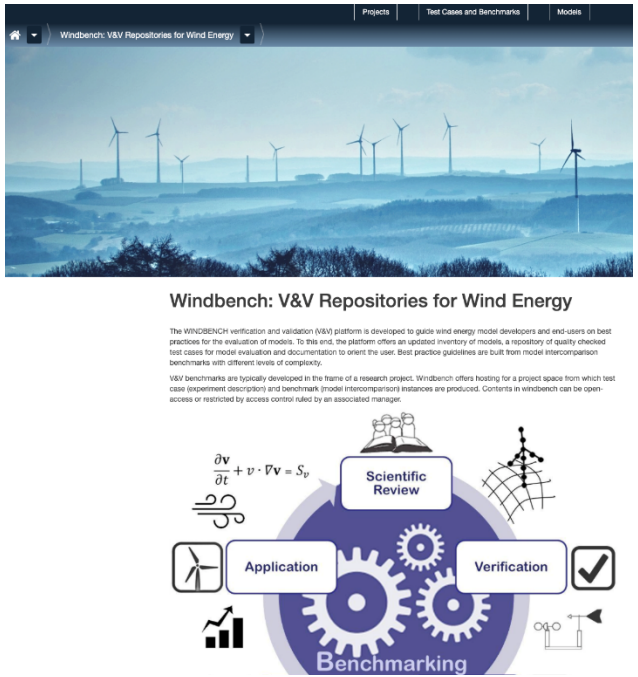
Basic data set 1

Data set 2 to which data set 1 is compared

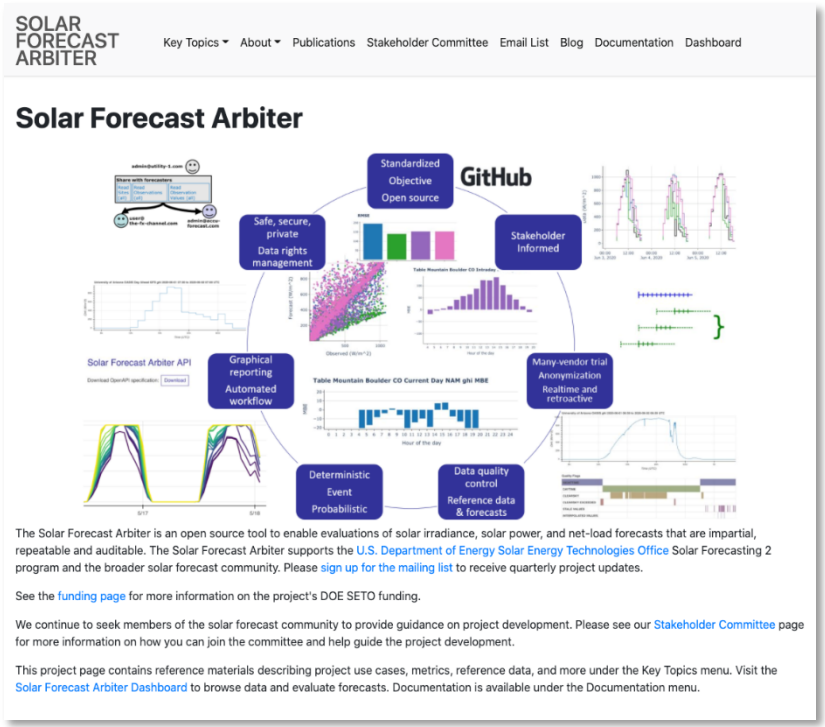
58 lines (52 sloc) | 1.11 KB

```yaml
# validation location
location:
  # latitude and longitude in degrees
  lat: 45.57451
  lon: -120.74734

prepare:
  # align the length of validation period of two data sets
  - trim:
      lower: -1000.0
      upper: 1000.0
  - linear_interp:
      #n: 23
      # resolution of input data
      period: 300 # every 300 second, according to WRF output's 5-minute resolution

time:
  window:
    # validation period start time
    lower: 2016-09-23 12:00:00
    # validation period end time
    upper: 2016-09-25 12:00:00
  trim: left # can be left or right

# validation metrics to be calculated and plotted
metrics:
  - rmse
  - bias
  - bias_pct
  - mae
  - mae_pct

levels:
  # height level of validation above ground level
  height_agl:
    - 40
    - 80
    - 120

# "truth" data input
left:
  name: sodar
  # data directory
  path: data/mw_data_sodar
  format: sodar_netcdf
  # variables to validate
  var: wind_speed

# data input to be compared with the "truth" data input
right:
  - name: wrf
    # data directory
    path: data/mw_data_wrf
    format: wrf_netcdf
    # variables to validate
    var:
      - u
      - v
```

Collaborations and Synergies



IEA Wind Task 31 Wakebench



iea wind

We Need Your Input!

We propose a validation framework that is reproducible, can be adapted by many users, includes metrics that are state of the art and stakeholder focused, can be extended, can be used for any numerical model output, and is freely available.

- Is this tool useful?
- What applications would you use it for?
- Is it straightforward to use? Which issues did you run into?
- Are there any applications that you would need it for that it currently cannot address?
- Which additional metrics would you want to see in this tool?
- Would you use the tool in the future with the additional metrics? Why, why not?
- How should we disseminate it?

Messner, JW, Pinson, P, Browell, J, Bjerregård, MB, Schicker, I. Evaluation of wind power forecasts—An up-to-date view. *Wind Energy*. 2020; 23: 1461– 1481. https://doi.org/10.1002/we.2497

iea wind

www.IEAWindForecasting.dk

Caroline Draxl

Caroline.Draxl@nrel.gov