Turbulence Intensity Adjustment Tests Tool

CFARS Site Suitability Subgroup

User's Guide

https://github.com/CFARS/site_suitability_tool

01/28/2021

USER'S GUIDE
TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.0 GENERAL INFORMATION

1.1 Turbulence Intensity Adjustment Tests Tool Overview

The purpose of this software is to support the research initiative of the CFARS Site Suitability Subgroup. Using a “closed-data” approach, a standardized testing and analysis methodology is useful to investigate a variety of turbulence intensity adjustment techniques. This tool is designed so that users can extract results to submit to the research initiative without having to share data.

1.2 License

The software includes a permissive BSD 3-Clause License.

1.3 Contributors and Points of Contact

For 2021, the point of contact for this software is Alexandra Arntsen aea@nrgsystems.com
The maintenance of this software and point of contact beyond 2021 is yet to be determined.

Programming contributors include:

Alexandra Arntsen, NRG Systems
Nikhil Kondabala, Apex
Andrew Black, Vaisala
Barrett Goudeau, NRG Systems
Nigel Swytink-Binnema, Nergica
Nicolas Jolin, Nergica

The CFARS Site Suitability Subgroup has collectively advised on and decided on tool functionality and content. Methods of adjusting turbulence intensity have not been created by the subgroup, but rather this software incorporates pre-existing methods for testing an analysis.

1.4 Organization of the User Guide

This manual is divided into 5 major sections, listed and defined below.

1.0 – *General Information*

2.0 – *Getting Started:* Describing the necessary set up of your system to run the python program.

3.0 – *Preparing Data and Configuration:* Requirements for your input data and how to fill out the configuration template.

4.0 – *Running the Tool:* How to execute the python script locally.

5.0 – *Interpreting the Output:* How to navigate and interpret the output file.

1.5 Acronyms and Abbreviations

TIATT - Turbulence Intensity Adjustment Testing Tool
TI - Turbulence intensity
RSD – Remote sensing device

2.0 GETTING STARTED

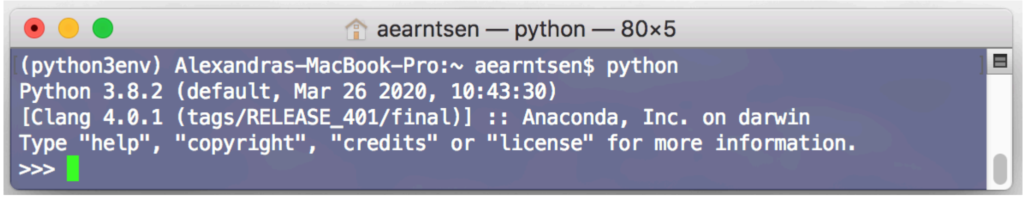
2.1 Setting up python

The turbulence intensity adjustment tests tool is a python program. To extract results from your data set, you will need to have python installed on your local machine. Furthermore, this software is compatible with python 3.

2.1.1 Confirm you have python 3 installed.

For **MacOS** open terminal and type:

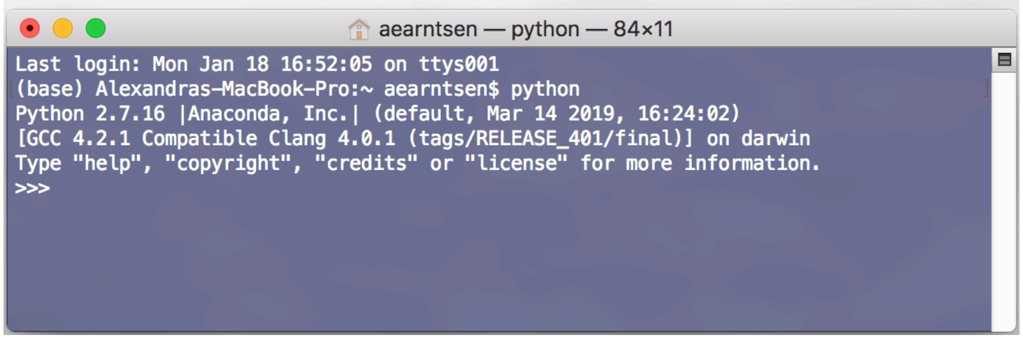
>>> python



```
(python3env) Alexandras-MacBook-Pro:~ aearntsen$ python
Python 3.8.2 (default, Mar 26 2020, 10:43:30)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

If the system reports a version of python 3 or higher, you should be ready to proceed.

However, if your system reports a version of python 2 (as seen below), you will need to update to python 3.



```
Last login: Mon Jan 18 16:52:05 on ttys001
(base) Alexandras-MacBook-Pro:~ aearntsen$ python
Python 2.7.16 |Anaconda, Inc.| (default, Mar 14 2019, 16:24:02)
[GCC 4.2.1 Compatible Clang 4.0.1 (tags/RELEASE_401/final)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

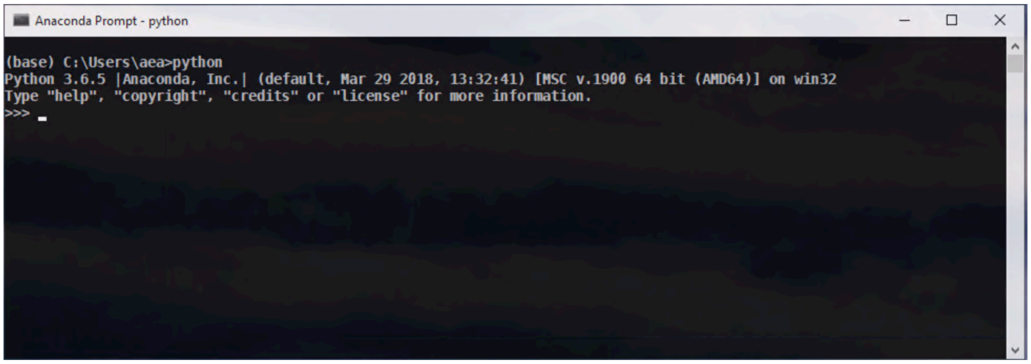
in terminal window type:

>>> conda update conda

***NOTE: If in terminal “python is not recognized as an internal or external command”, you need to add the installed python executable to your environment variable system path

For **Windows** open anaconda prompt and type:

>>> python



```
(base) C:\Users\aea>python
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

If the system reports a version of python 3 or higher, you should be ready to proceed.

However, if your system reports a version of python 2, you will need to update to python 3. Do update type:

```
>>> conda update conda
```

2.1.2 If you do not have python installed, please install python 3 for macOS or Windows

https://www.anaconda.com/products.individual

2.2 Download or update the repository

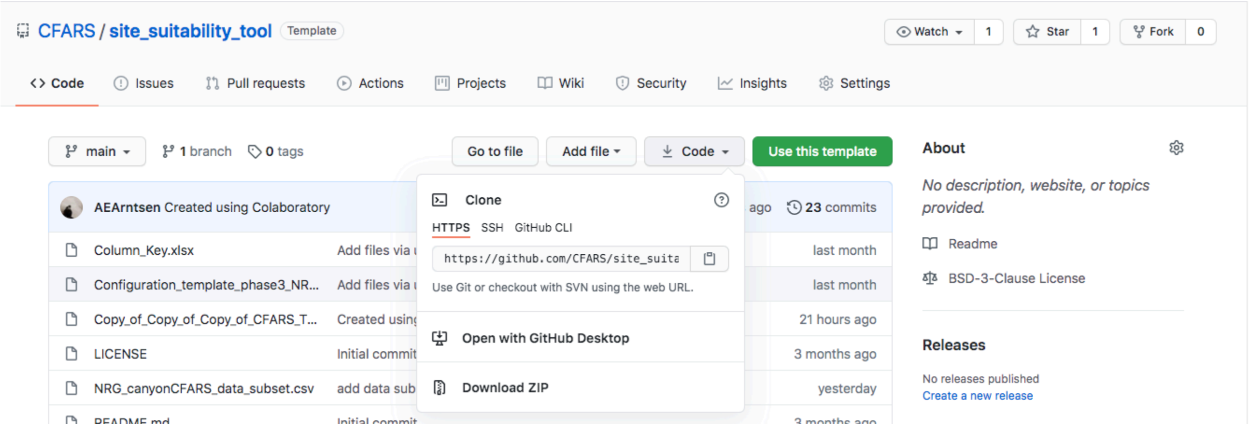
Once you confirm your python is set up correctly, you need to be sure to have a local copy of the tool repository.

https://github.com/CFARS/site_suitability_tool

If you are familiar with git, you can **clone the repository** above by in terminal or command line navigating to a directory of choice and typing

```
>>> git clone https://github.com/CFARS/site_suitability_tool.git
```

Another option is to **download the repository.** To do this, click the ⤓ Code ▾ button in the repository website and then choose Download Zip as seen in the screen shot below.



AEArntsen Created using Colaboratory ... ago 23 commits

Column_Key.xlsx	Add files via u	last month
Configuration_template_phase3_NR...	Add files via u	last month
Copy_of_Copy_of_Copy_of_CFARS_T...	Created using	21 hours ago
LICENSE	Initial commit	3 months ago
NRG_canyonCFARS_data_subset.csv	add data sub	yesterday
README.md	Initial commit	3 months ago

```
https://github.com/CFARS/site_suita
```

Use Git or checkout with SVN using the web URL.

No description, website, or topics provided.

No releases published

2.3 Install requirements

The anaconda build of python comes with most of the packages you will need to run this tool, but to be sure your version of python is configured correctly, you can use the requirements file in the repository.

In terminal or command prompt, navigate to the site_suitability_tool directory you created by either downloading or cloning the repository. This directory contains all of the files from the repository, including a file named 'requirements.txt'.

Next, type

```
>>> pipi install -r requirements.txt
```

3.0 PREPARING DATA AND CONFIGURATION

3.1 Preparing your data

The tool processes individual data sets separately, so each dataset should be a separate **.csv file** with a unique name.

1) Save each project data set to the directory that contains the tool
2) Make sure each dataset has only one row of header information

3.2 Preparing your configuration file

Each dataset requires a corresponding configuration file to be interpretable by the tool. Follow the below steps to setting up configuration files for each of your datasets.

1) In the repository directory you will find a file named 'Configuration_template.xlsx'. The first step is to **create a version of this file for each one of your unique datasets** with the dataset name included in the name of its configuration file. For example, if I have 2 datasets, 'NRG_project1.csv' and 'NRG_project2.csv' I will have 2 configuration files (copies of the template) named 'configuration_NRG_project1.xlsx' and 'configuration_NRG_project2.xlsx'.

2) Match Your Columns.

   It is important that we ensure column names are interpreted correctly by the tool. The first step in configuration is to label your data columns with the standardized column names.

   2.1 In column A, copy and paste vertically the single row header information from your project file.
   2.2 Label all the essential columns with the dropdown selection in column B of the template.
   2.3 Fill out the Site Metadata information in columns E & F
   2.4 Fill out information about filtering of the dataset in columns I & J
   2.5 Mark the version of firmware if your dataset contains ZX data

3) Refer the example files in the repository: 'Example_project_subset.csv' & its corresponding configuration file 'configuration_Example_project_subset.xlsx'

4.0 RUNNING THE TOOL

4.1 Arguments

-in location of input data file

-config	location of configuration file
-rtd	location of windcube rtd files (only necessary if running WiSE and LTERRA)
-res	location of results file
--timetestFlag	performs length of data collection analysis (note: this takes a several minutes to run)

4.2 Command in terminal/command prompt

In terminal or command type a command that follows the following structure:

```
>>> python phase3_implementation_noNDA.py -in NRG_project1_data.csv -config
configuration_NRG_project1.xlsx -rtd /Users/aearntsen/CFARS/NRG_project1/RTD -res
/Users/aearntsen/CFARS/NRG_project1/out_NRG_project1.xlsx --timetestFlag
```

5.0 INTERPRETING THE OUTPUT

5.1 Structure of the Output file

The output file is generated in the form of an excel workbook.

Tab Name	Description

5.2 Adjustment Methods

The following section provides a more in-depth description of each adjustment method testing in the tool.

Method 1
Method 2
Method 3
Method 4
Method 5
Method 6

5.3 Extrapolated TI

In