Stephen Xu

August 1, 2023

Wind and Solar Aggregation Report

**Task Description**

This project aims to aggregate and perform basic analysis on historical ERCOT Wind and Solar Data. This tool and its outputs will help answer curtailment questions asked by upper management and assist traders in their daily activities.

Note: An old version of this task was developed by Bryan Ding in his internship during the Summer of 2021. Currently, it is broken. My newer project features cleaner, more readable code, faster runtime, bug fixes, and a new functionality to aggregate the Resource-level information.

**Solution Design**

1. Script Location: *\\pzpwcmfs01\CA\11\_Transmission Analysis\ERCOT\101 - Misc\CRR Limit Aggregates\Python Scripts\Wind and Solar Aggregation*
2. Output Location: *\\pzpwcmfs01\CA\11\_Transmission Analysis\ERCOT\101 - Misc\CRR Limit Aggregates\Data*

This can be adjusted manually by editing the *output\_path* variables in each of the Python scripts.

1. Running *Resource\_Level.py* aggregates the historical Resource Level data and outputs the result to a CSV.
2. Similarly, running *Wind\_Solar\_Aggregator.py*  aggregates the historical wind and solar data and outputs the result to a CSV.
3. **Important**: Each of the Python scripts contain a *version* variable – line 20 and 17 in *Resource\_Level.py* and *Wind\_Solar\_Aggregator.py*, respectively. As indicated in the comment above, version 1 aggregates Solar Data and version 2 aggregates Wind Data. Adjust accordingly.

**Important Prerequisite Information**

The Python scripts for this project read the Excel sheets located at *\\pzpwcmfs01\ca\11\_Transmission Analysis\ERCOT\04 - Monthly Updates\101 - Misc\01 - General\Wind Forecast Monthly*.

The Excel sheets in this folder all contain images, which the Python pandas library can have difficulties reading. Be sure that the **openpyxl** package is at least version **3.1.2.** You can verify if the version on your computer satisfies this requirement by creating a new Python script, importing openpyxl and printing “*openpyxl.\_\_version\_\_*”*.* If you do not meet this requirement, follow the instructions below:

1. Open the Anaconda Prompt from the start menu.
2. Type in the following command: *pip install openpyxl==3.1.2*
3. Press Enter and see if the installation was successful. If you get denied by a permission error, restart at step 1 and run the Anaconda Prompt as an administrator. Use an SA account (or ask IT) if prompted.
4. If you cannot get steps 1-3 to work, you can create a new conda environment in a folder that you have write access to. Type the following command:  
    *conda create --name {env\_name}*Be sure to replace *env\_name* with your desired environment name.
5. From here, install all necessary packages to that environment via *conda install* *{pkg\_name}.* In particular, you will need the command

*conda install openpyxl=3.1.2*

You will also need the Pandas library. I don’t believe the Pandas version is too important here – the default one should suffice.

1. Be sure to redirect the IDE you use to develop Python to run these scripts using the new conda environment.

Sample Output

