Lab 4: Analyzing Solar Energy Data (Time series)

1 Lab Aims

In this lab, you will use your new knowledge to analyze a real dataset related to solar energy (see Figure 1). To succeed, you will need to import data, and answer questions using data visualization tools.

Figure 1 depicts a high-level view of the electricity generation process, from a module of solar panels to the power grid. Solar energy is directly converted into electricity through the photoelectric effect. Whenever materials such as Silicon (the most common semiconductor material used in solar panels) are exposed to light, photons (subatomic particles of electromagnetic energy) are absorbed before releasing free electrons and as a result, creating a Direct Current (DC). In fact, by using an inverter, the DC is converted into Alternating Current (AC) and fed into the power grid, where it can be distributed to homes.

There are different types of solar power systems, such as off-grid systems and direct photovoltaic systems that are illustrated in Figure 2.

The dataset used in the examples is a customized dataset using solar radiation measurements from the Measurement and Instrumentation Data Center (MIDC) of the U.S. National Renewable Energy Laboratory (NREL). The selected station is located at the University of Nevada - Las Vegas (UNLV) and the in-use data includes measurements for the year between 2021 and 2022 (1 year). Moreover, this dataset is 1-minute resolution data with 21 variables related to meteorological and other relevant data: “ambient temperature”, “wind speed”, “wind direction”, “Global Horizontal Irradiance (GHI)”, “Direct Normal Irradiance (DNI)”, “Diffuse Horizontal Irradiance (DHI)”, “zenith and azimuth angles”, “airmass”, among many others.

The main target of this work is to analyze the level of solar radiation versus both date and time, based on last year’s measurements. This study can be useful to know whether solar energy batteries (Power storage and distribution as illustrated in Figure 3) will be reasonable to use in the future. This can be possible by predicting the solar radiation level.

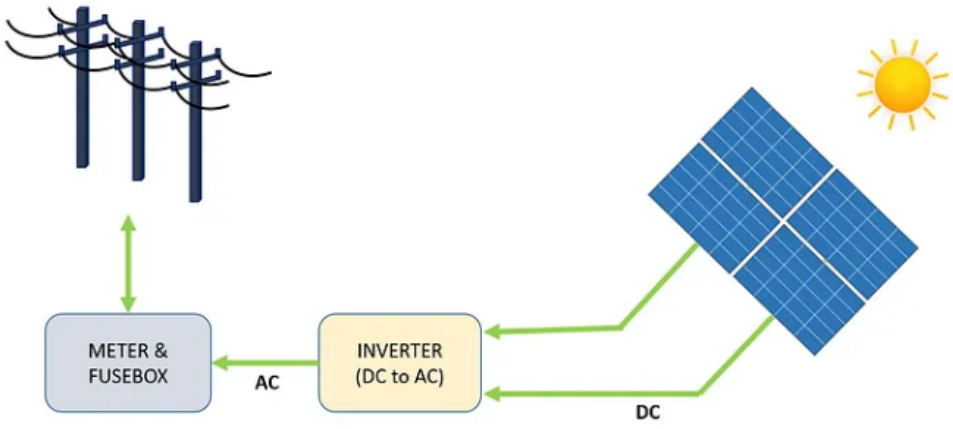
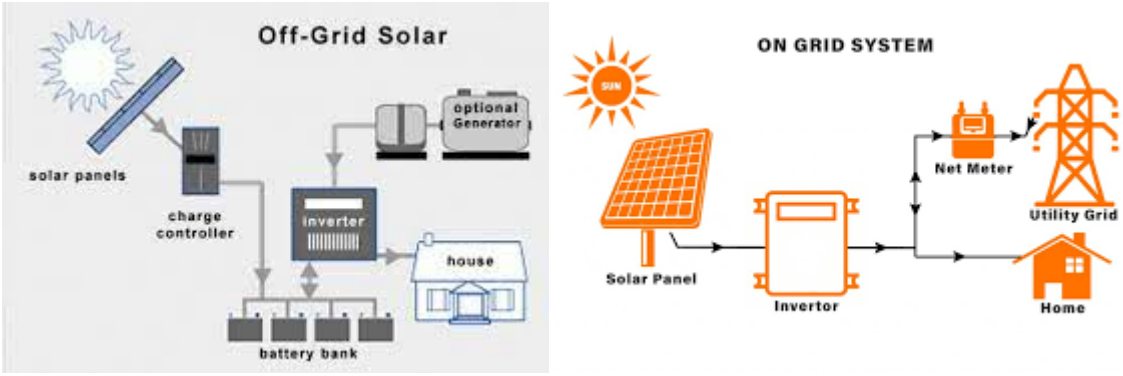


Figure 1: Grid connected Solar System



((a)) Off-Grid Solar System ((b)) On Grid Solar System

Figure 2: Types of solar power systems

1.1 Load the data

- Your first assignment is to import the required Python libraries that you might need to complete this lab such as Pandas.
- Then, load the corresponding data file (**'solar2021UNLV.csv'**) into `df_Solar` data-frame. Which column do you have to use as the index? and why?

1.2 Review the data

In this part, we will review the dataset, in order to allow you first to understand what you have "in your hand".

(i) What is the size of the loaded data frame?

(ii) Print the first and last 10 rows of the `df_Solar` data frame.

(iii) List the columns names included in this dataset and their data type.

(iv) Which features might include outliers? You have to justify this with a visual graph (boxplot).

1.3 Manipulating Data

(v) As you may have noticed, the dataframe df has an extra column. Remove unwanted column(s).

(vi) Check for and remove all rows with NaN, or missing values in the `df_ref` Data-Frame.

(vii) Check for and remove all duplicate rows in this data frame.

1.4 Data analysis & visualisation

(viii) What is the largest value for *'GHI'* and its corresponding *'date'*?

(ix) What is the mean value for *'GHI'* per month and per season?

Figure 3: An example of an implemented solar energy system

(x) Plot the variation of *'GHI'* for a specific day. Then, another plot for five consecutive days from a specific day. You can get the GHI for a given day in the time-series by slicing the dataframe as indicated in the following: *df_ref['2021-06-01':'2021-06-02']['ghi']*

(xi) Plot the variation of *'GHI'* by day, and month. Can you do it without adding new columns related to day and month? If yes, how?

(xii) Plot the variation of *'GHI'*, *'DHI'*, *'DNI'* and temperature versus the day date as a subplot for each case.

(xiii) Plot the variation of temperature versus radiation *'GHI'*.

(xiv) Plot the correlation matrix using a heatmap for a clear understanding of the correlation among all dataset parameters. Use the Pandas "corr" function to derive the correlation matrix. Store the result in a variable called "corr". These values answer the questions on the 'relationship' between columns. A perfect positive correlation yields a value of +1, whereas a perfect negative correlation yields a value of -1.

- Notice the left-to-right diagonal in the correlation table generated above. Why is the diagonal filled with 1s? Please explain.
- Still looking at the correlation table above, notice that the values are mirrored; values below the 1 diagonal have a mirrored counterpart above the 1 diagonal? Please explain.
- Many variable pairs present a correlation close to zero. What does that mean?
- Which variables have a stronger correlation? Can you list them?

(xv) Now, you should do another plot for the correlation between 'GHI' radiation and the other parameters. Let us indicate that the "corr" function uses the "pearson" by default.

- Which variables have a stronger correlation with *'GHI'*? Can you list them?
- Which variables don't have a correlation with *'GHI'*? Can you list them?
- Additionally, You should do the same with other correlation methods implemented by Pandas as "Kendall Tau correlation coefficient" and "Spearman rank correlation".
- What can you conclude from the obtained results?

(xvi) Create a scatter plot to show the relationship between '*Azimuth Angle*' (on the x-axis) and '*GHI*' (on the y-axis). Can you see any interesting pattern in the scatter plot?

(xvii) Create a scatter plot to show the relationship between '*Zenith Angle [degrees]*' (on the x-axis) and '*GHI*' (on the y-axis). Can you see any interesting pattern in this scatter plot?

(xviii) Create a scatter plot to show the relationship between '*Azimuth Angle*' (on the x-axis) and '*Zenith Angle [degrees]*' (on the y-axis). Now, try to use the *'GHI'* column to color-code the points. Did the obtained result verifies your previous conclusion?

(xix) Displaying a 2D KDE plot with the corresponding KDE plots for *'GHI'* and temperature. Similarly, do another 2D KDE plot between '*"wind speed"* and '*"wind direction"*.

(xx) Plot the distribution of *"wind speed"*, *"wind direction"*, and *"temperature"* as subplots.

(xxi) PLot a two-dimensional (2D) KDE plot that shows the distribution in values for *'DHI'* and *'DNI'*.

1.5 Investigating differences

(xxii) In fact, the *'GHI'* column can be used to understand the difference between solar and not solar days. Based on the histograms above:

- Do solar periods have higher or lower values for *'GHI'* (relative to non-solar periods)?
- Which season seems to have a larger range of potential values?

(xxiii) Create a single figure containing the KDE plots (that show the density) of *'DHI'* and *'DNI'*.

(xxiv) Provide a pair plot that plots all pairwise relationships in the dataset by using the corresponding method of the "Seaborn" library. What can you conclude?

1.6 More Investigation

- Create a 3D scatter plot variables (x,y,z): where 'DNI', 'DHI', and 'GHI' represents the axes x, y, and z, respectively. You will need the following code:

```
ax = plt.axes(projection='3d')
ax.scatter3D (x,y,z)
```

- What can you conclude from this plot?

1.7 Down and up-sampling time-series data

When assessing solar resources, you may need a different time resolution than your data for a particular part of the analysis. In those cases, it is possible to down-sample and up-sample the data at different temporal resolutions using two different methods within the Pandas library called *'resample'* and *'asfreq'*. Depending on your needs, you will opt for one or the other. Regardless of the method, both of them require a DataFrame with d*atetimeindex* either time-aware (localized) or time-naive (not localized). What is the difference between them? Try to explain the difference in detail.

- Plot the variation of *'GHI'* versus 30-minutes, 1 hour, and week period intervals. What can you conclude?

- Plot the variation of *'GHI'*, *'DHI'*, *'DNI'* and temperature versus two weeks period intervals as subplots.