**Purpose of the Model**

The model extensively explores the data of Junior Analysts, delving deep to uncover hidden patterns within the dataset. It not only identifies these patterns but also extracts vital insights that hold significance for both Stakeholders and senior management within the organization.

Employing a systematic and methodical approach, the model meticulously guides through the process in a series of intuitive Python notebooks (. ipynb). These notebooks are crafted with clear and insightful comments, designed to be comprehensible to individuals with varying levels of expertise. Whether you're a technical expert or someone new to the field, the commentary will facilitate your understanding, making the analysis accessible and transparent.

The step-by-step breakdown of the analytical journey ensures that every stage is well-documented, making it easy to follow the reasoning behind each decision. The model's transparency and comprehensibility ultimately empower stakeholders and decision-makers to grasp the intricacies of the analysis, fostering informed discussions and strategic decisions based on the derived insights.

**Data and Checks**

1. **Setting up the Environment**

To kickstart the project, a Jupyter Notebook named "electricity\_consumption" was created. The environment setup included the following steps:

* Installation of Jupyter Notebook using the command **pip install jupyter**
* Launching Jupyter Notebook with **Visual Studio Code**
* Creating a new notebook and renaming it to "electricity\_consumption"
* Importing essential Python libraries for data analysis:
  + import pandas as pd
  + import numpy as np
  + import os
  + import matplotlib.pyplot as plt
  + import seaborn as sns
  + import plotly.express as px

1. **Data Importation checks**

Before diving into the analysis, it's essential to ensure the data has been imported correctly and is in the expected format. This section outlines the steps taken to import the data and perform initial data checks. The data was imported and put in put in an object named **electricity\_consumption\_data.**

* We parsed the **Date\_hour\_start** to **datetime** format to prevent unnecessary errors downstream.
* The first two rows were skipped since they did not have important data and were making it difficult to read the header
* The data was restricted to 8760 rows because for some reason pandas was reading the entire sheet

1. **Unifying naming conventions**

All the naming conventions of the columns was changed to lower characters to ensure code consistency and maintainability. This prevents mistakes that usually arise with mixed naming conventions.

1. **Checking the statistical measures**

The keyword describe() was used to get a quick glimpse of the statistical measures of the numerical variables in the data

1. **Checking information of the data**

The keyword info() was used to have a feel of the information in the data. The information includes; the **dtypes** and the **memory usage** of the data

1. **Checking For missing values in the data**

The keyword df.isnull().sum() was used to check for null values and get its sum. This data did not have null values

1. **Checking for Outliers and Removing them**

Boxplot is usually a good choice to show outliers in a dataset. After you identify the outliers (extreme values) the next thing is to deal with them (of course depending on the needs of the analysis). In this case the extreme values were removed by using Z-score to make necessary changes and then using trim method to remove the outliers. It is imperative to build a boxplot for the second time to check if the outliers were removed as expected

1. **Checking the amount of electricity to be bought for each hour from the electricity provider (measured in kWh and subject to a minimum of zero)**

The data is first grouped by ‘date\_hour\_start’ and aggregated by sum of 'electricity\_usage\_kwh'.

1. **Checking the net solar Electricity generated**

This will be achieved by finding the difference of the solar electricity generated and the electricity used (usage).

1. **Checking the cumulative battery charge**

This will be achieved by summing the electricity available at the beginning of the year with net for each hour

1. **Finding the amount of electricity for each hour in 2020 that would have been bought from the electricity provider (measured in kWh and subject to a minimum of zero), assuming a battery had already been installed.**

To achieve this the we first pick the net negative data from the net\_electricity column (feature) and summing up their absolute values

1. **Finding the Savings Over 2020 from installing a battery compared to using the existing solar panels alone**

We first take the absolute value of the bought electricity and find the difference with net positive from the net\_electricity column (feature), thereafter you multiply with the cost $ 0.17 per kWh

1. **Building bespoke charts**

**-**We first build a chart for monthly solar generation; this is achieved by grouping the data by month with regards to solar electricity generation and aggregating by sum.

-The second chart is monthly solar usage; this is achieved by grouping the data by month with regards to solar electricity usage and aggregating by sum

-The third chart is monthly electricity purchased from the electricity provider (without battery); this is achieved by grouping the data by month with regards to solar electricity purchased (net negative) and aggregating by sum

-The last chart is monthly electricity purchased from the electricity provider (with battery); this is achieved by grouping the data by month with regards to solar purchased (net positive) usage and aggregating by sum

1. **Exporting the data for further visualization**