Solar Power Plant Case Presentation

Rezso Roland Gimesi

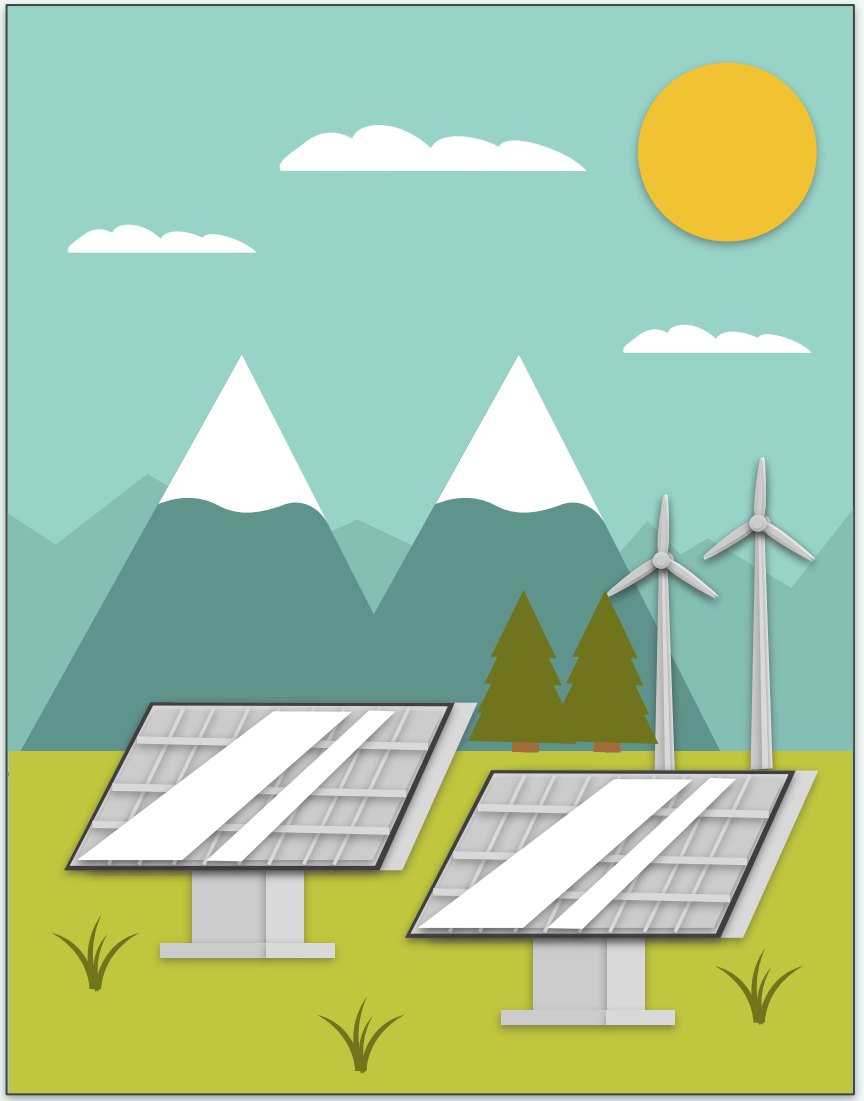
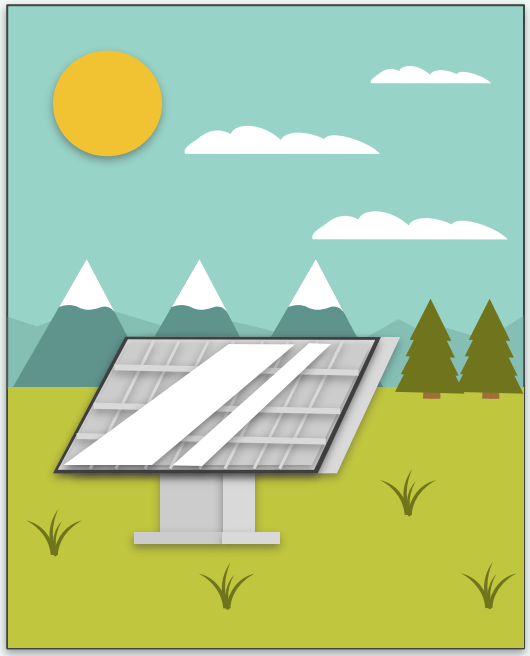


Table of contents

Personal Introduction

Rezso Roland Gimesi

- BSc Commerce and Marketing
- 1,5 Digital Account Manager – data analysis
- MSc Information Management, specialized in data analysis
- 1 year Student Worker in Data Analysis at Siemens Gamesa



The Case

Energy company is a renewable energy company that provides green energy through designing, building and operating photovoltaic power plants.

Dataset:

The energy production of 3 power plants over 3-year period.

Questions:

Quantitative	Qualitative	Additional
• What is the DC capacity of each power plant in kWp? • What is the ~ yearly performance ratio of the plants? • What percentage of the data is invalid?	• Explain invalid data • If there is seasonal difference in the data, explain why	• How would you visualize data to inform non-subject experts? • Other details about the parks? • Can you estimate approximate location?

Analytical Toolkit

Excel

Data structure manipulation

Python

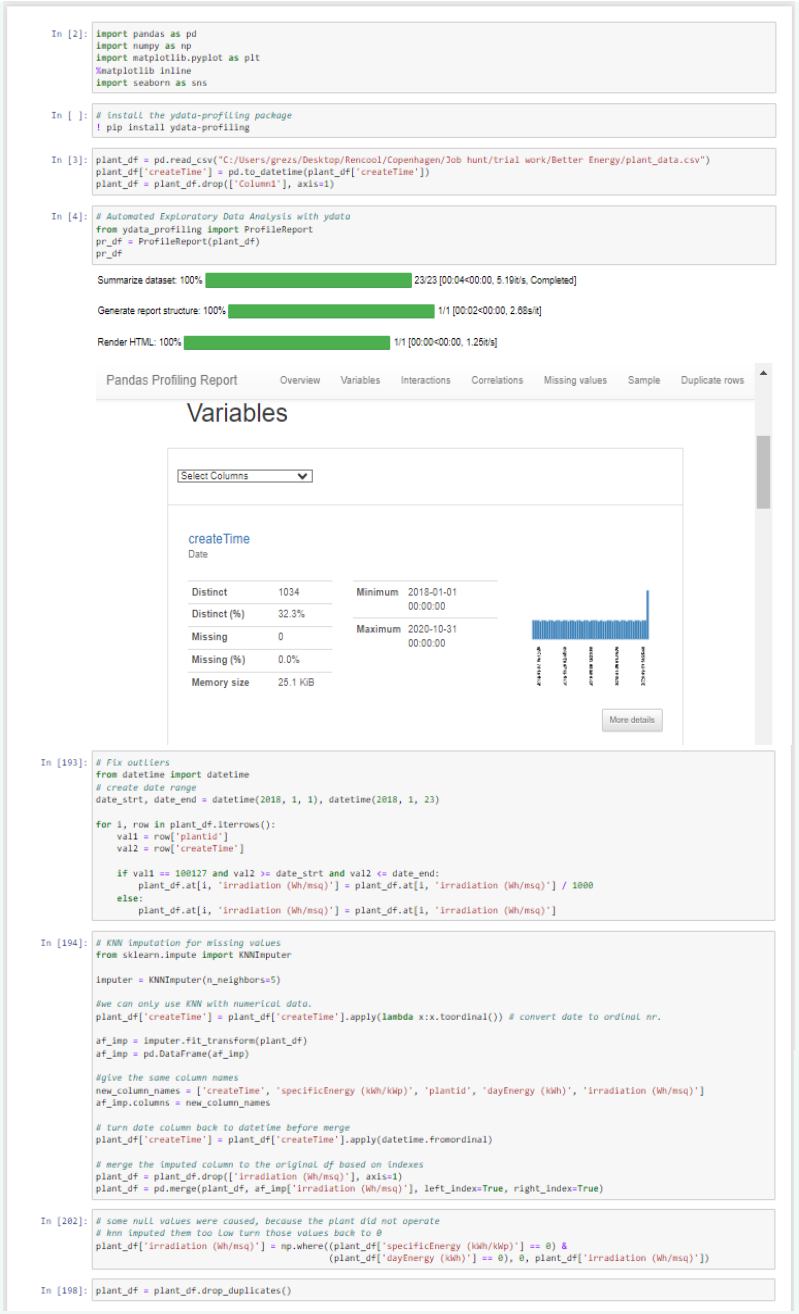
Exploratory Data Analysis, data manipulation,

Power BI

Data analysis, data visualization

Analysis

1. Automated EDA with ydata
2. Basic data manipulation
   - Fixed outliers, duplicates
   - Imputation of null values
3. Added 2 columns
   - kWp (peak performance of the plants)
   - Plant number



```
In [2]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        %matplotlib inline
        import seaborn as sns

In [ ]: # install the ydata-profiling package
        ! pip install ydata-profiling

In [3]: plant_df = pd.read_csv("C:/Users/grezs/Desktop/Rencool/Copenhagen/Job hunt/trial work/Better Energy/plant_data.csv")
        plant_df['createTime'] = pd.to_datetime(plant_df['createTime'])
        plant_df = plant_df.drop(['Column1'], axis=1)

In [4]: # Automated Exploratory Data Analysis with ydata
        from ydata_profiling import ProfileReport
        pr_df = ProfileReport(plant_df)
        pr_df
```

Summarize dataset: 100% 23/23 [00:04<00:00, 5.19it/s, Completed]

Generate report structure: 100% 1/1 [00:02<00:00, 2.68s/it]

Render HTML: 100% 1/1 [00:00<00:00, 1.25it/s]

Pandas Profiling Report — Overview | Variables | Interactions | Correlations | Missing values | Sample | Duplicate rows

Variables

Select Columns

createTime
Date

Distinct	1034	Minimum	2018-01-01 00:00:00
Distinct (%)	32.3%	Maximum	2020-10-31 00:00:00
Missing	0		
Missing (%)	0.0%		
Memory size	25.1 KiB		

More details

```
In [193]: # Fix outliers
          from datetime import datetime
          # create date range
          date_strt, date_end = datetime(2018, 1, 1), datetime(2018, 1, 23)

          for i, row in plant_df.iterrows():
              val1 = row['plantId']
              val2 = row['createTime']

              if val1 == 100127 and val2 >= date_strt and val2 <= date_end:
                  plant_df.at[i, 'irradiation (Wh/msq)'] = plant_df.at[i, 'irradiation (Wh/msq)'] / 1000
              else:
                  plant_df.at[i, 'irradiation (Wh/msq)'] = plant_df.at[i, 'irradiation (Wh/msq)']

In [194]: # KNN imputation for missing values
          from sklearn.impute import KNNImputer

          imputer = KNNImputer(n_neighbors=5)

          #we can only use KNN with numerical data.
          plant_df['createTime'] = plant_df['createTime'].apply(lambda x:x.toordinal()) # convert date to ordinal nr.

          af_imp = imputer.fit_transform(plant_df)
          af_imp = pd.DataFrame(af_imp)

          #give the same column names
          new_column_names = ['createTime', 'specificEnergy (kWh/kWp)', 'plantId', 'dayEnergy (kWh)', 'irradiation (Wh/msq)']
          af_imp.columns = new_column_names

          # turn date column back to datetime before merge
          plant_df['createTime'] = plant_df['createTime'].apply(datetime.fromordinal)

          # merge the imputed column to the original df based on indexes
          plant_df = plant_df.drop(['irradiation (Wh/msq)'], axis=1)
          plant_df = pd.merge(plant_df, af_imp['irradiation (Wh/msq)'], left_index=True, right_index=True)

In [202]: # some null values were caused, because the plant did not operate
          # knn imputed them too low turn those values back to 0
          plant_df['irradiation (Wh/msq)'] = np.where((plant_df['specificEnergy (kWh/kWp)'] == 0) &
                                                      (plant_df['dayEnergy (kWh)'] == 0), 0, plant_df['irradiation (Wh/msq)'])

In [198]: plant_df = plant_df.drop_duplicates()
```

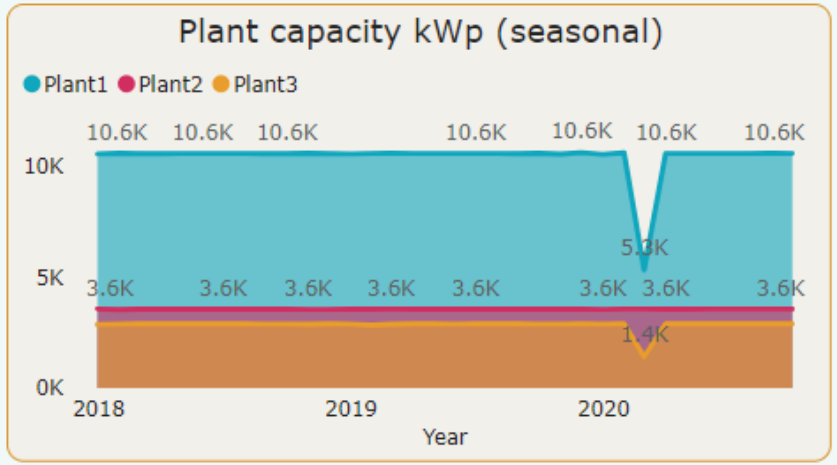
Notebook available here

Analysis – Quantitative

What is the DC capacity of each power plant in kWp?

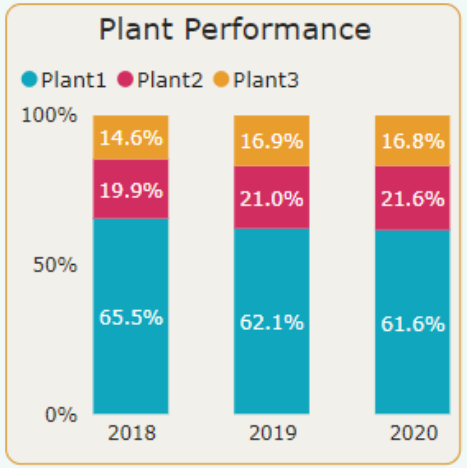
Plant1: 10.6 kWp
Plant2: 3.6 kWp
Plant3: 2.9 kWp

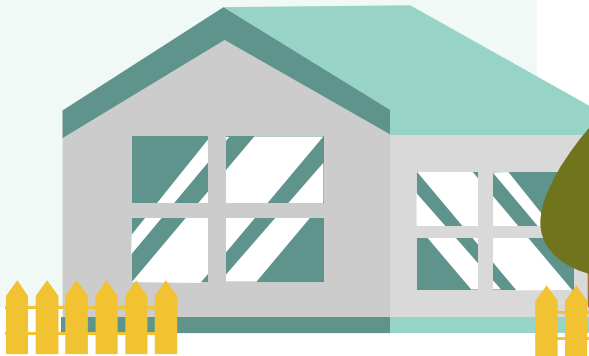
- 75% decrease in capacity between March 2020 for Plant1 & Plant3
- Produced kWh stayed intact



What is the ~ yearly performance ratio of the plants?

- Plant performance ratio is almost the same
- 2018 Plant 1 produced a bit more which causes the higher share.





Analysis – Quantitative /Qualitative

What percentage of the data is invalid? Explain invalid data.

Duplicates (3%):

Last day of the dataset got appended at the end of every month

Same day exists 34 times for all 3 Plants

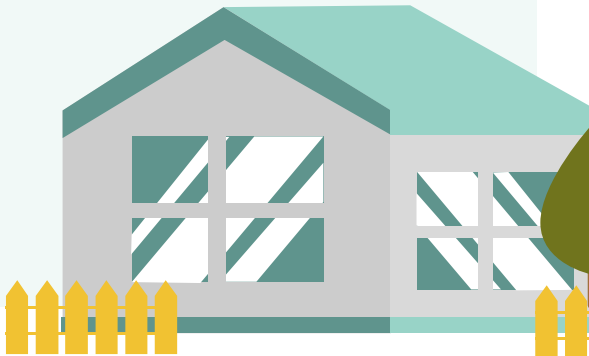
→ Should be deleted

Irrationally high irradiation (0.1%):

Higher than Solar constant (1370 W/m2) – can't be right

Decimal point seem to shifted to the right 3 digits

→ Should be normalized



Analysis – Quantitative /Qualitative

What percentage of the data is invalid? Explain invalid data.

Null values (gray area, 0.7%):

Informative or not?

- System is not operating – informative
- Missing by fault – non-informative

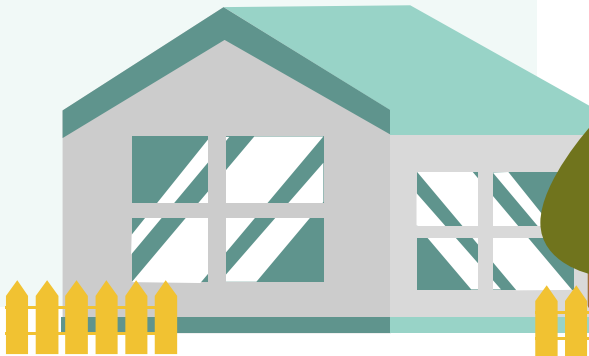
→ Informative: replace with 0
Non-inf.: imputation (KNN)

Column 1 (16.6%):

Faulty index, restarts at every month because of wrong appending

→ Should be deleted

20 % of all data can be considered invalid



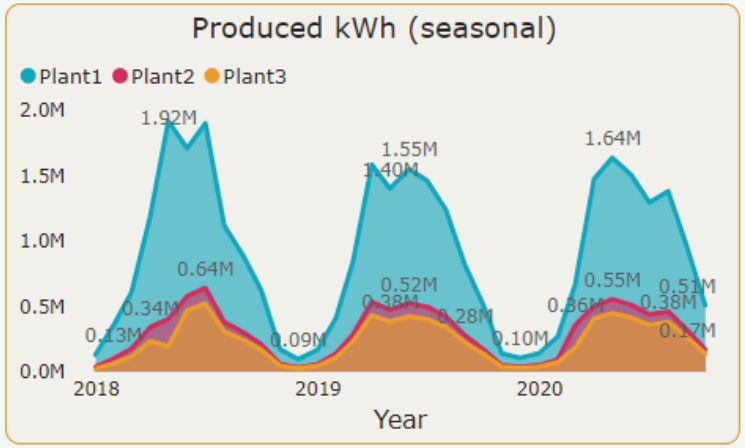
Analysis – Qualitative

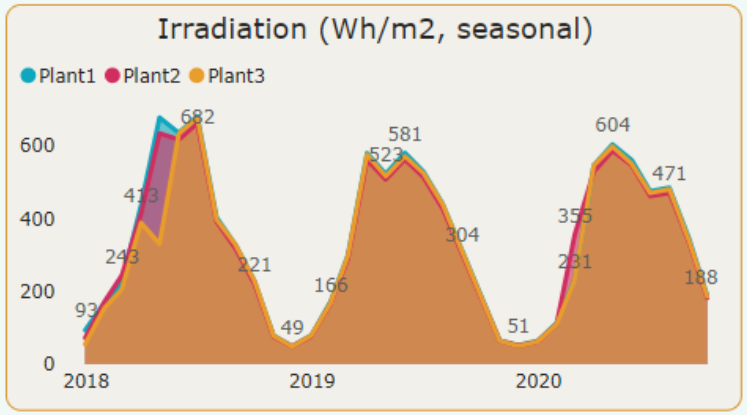
If there is seasonal difference in the data, explain why.

There is seasonal difference in the data.

- Caused by the sun's different availability
- Performance is much lower in the winter months as:
  1. days are shorter,
  2. less sunny,
  3. sunlight received from a less direct angle.

Energy produced changes with the amount of irradiation available.



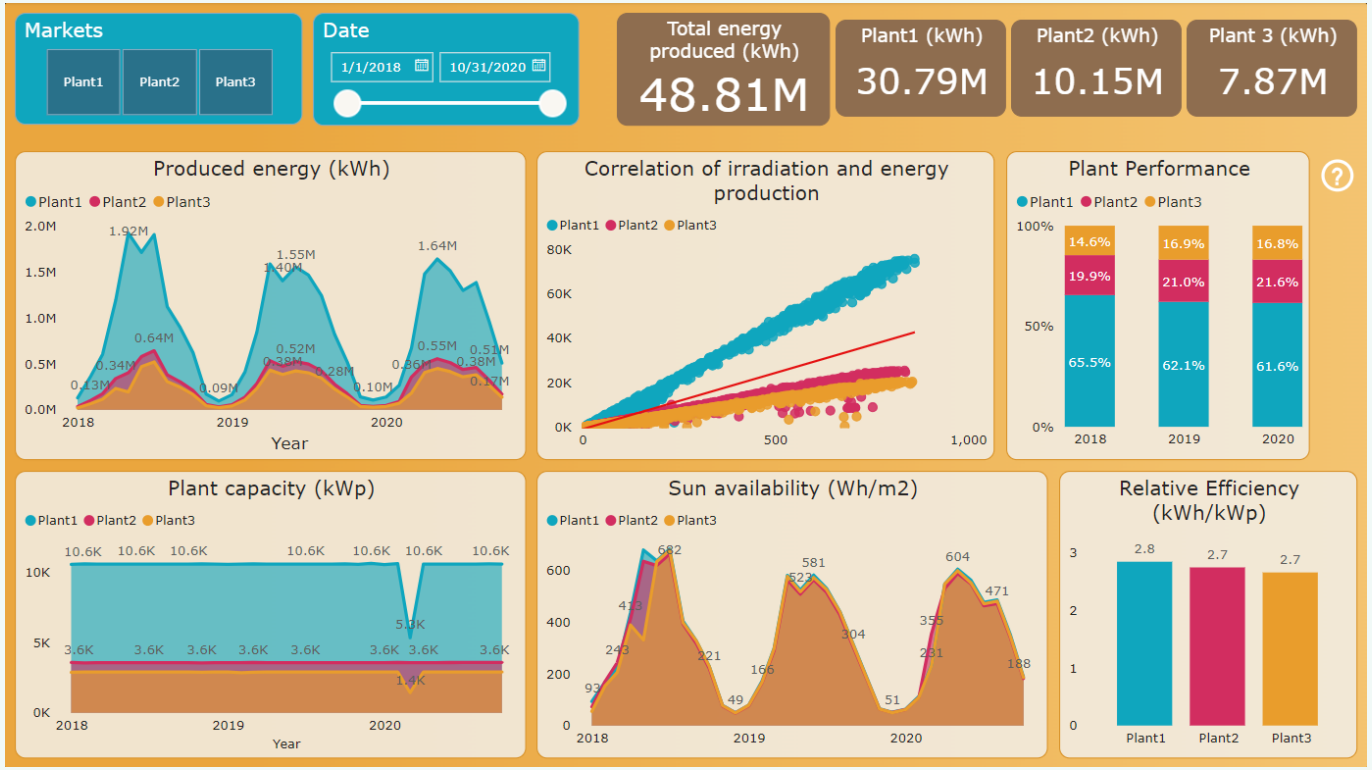


Analysis – Additional

How would you visualize data to inform non-subject experts?

Key principles:

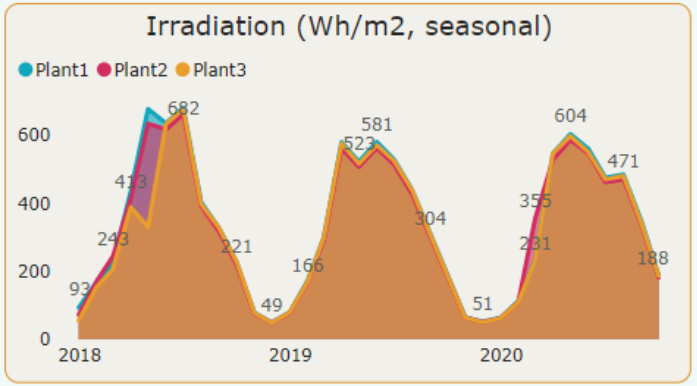
- Get requirements straight with key users
- Power BI - Interactive visualizations
- Avoid complicated terminology
- Main metrics on one page (+drill down)
- Easily understandable structure (filters, totals, etc.)
- Available documentation

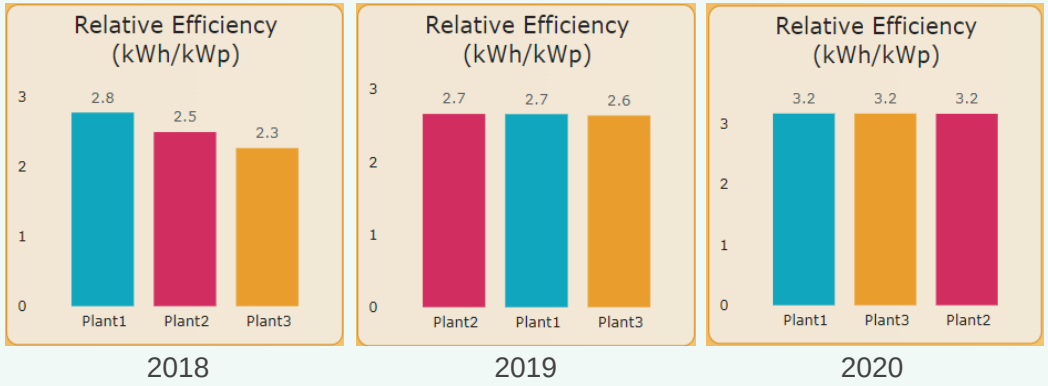


Dashboard available here

Analysis – Additional

What other details can you infer about the parks?
Can you estimate their apprx. location?





Overall:

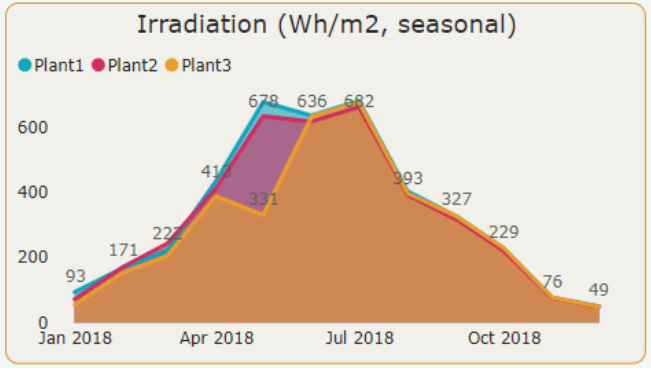
- Plants are located relatively close to each other (irradiation)
- Same technology – efficiency rate almost identical (except for a short period 2018)
- Plant 1 being the largest – Plant 3 the smallest

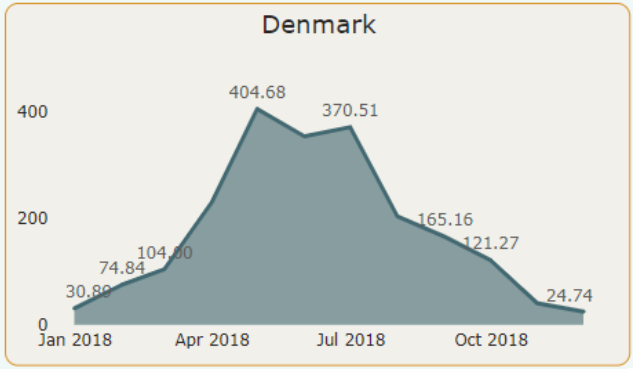
Location:

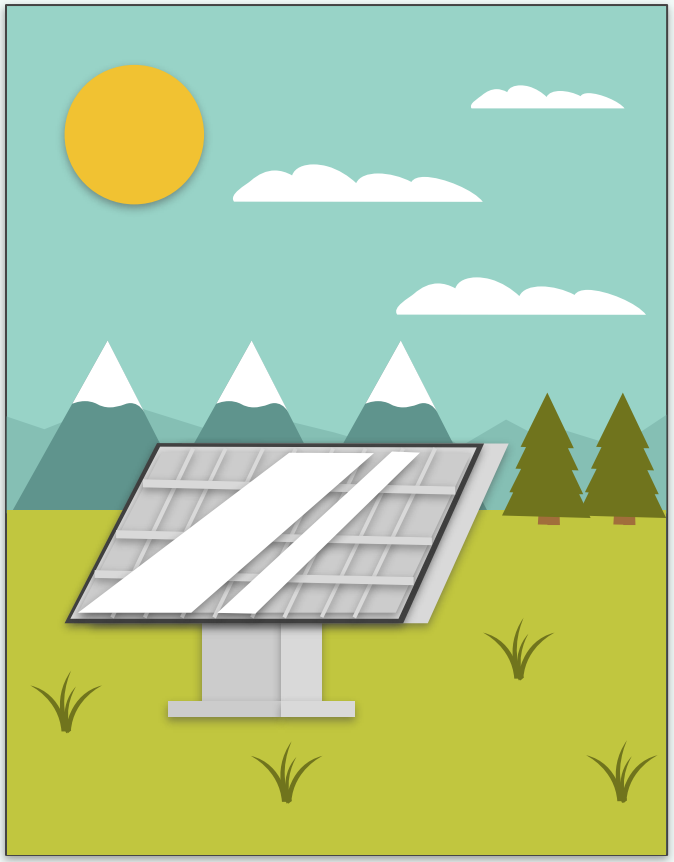
Possible if historical weather data is available

- Type of Irradiance ?
- BE main markets: Northern Europe / Ukraine
- NSRDB database – Daily GHI

Most likely located around Denmark.







Summary

Quantitative:

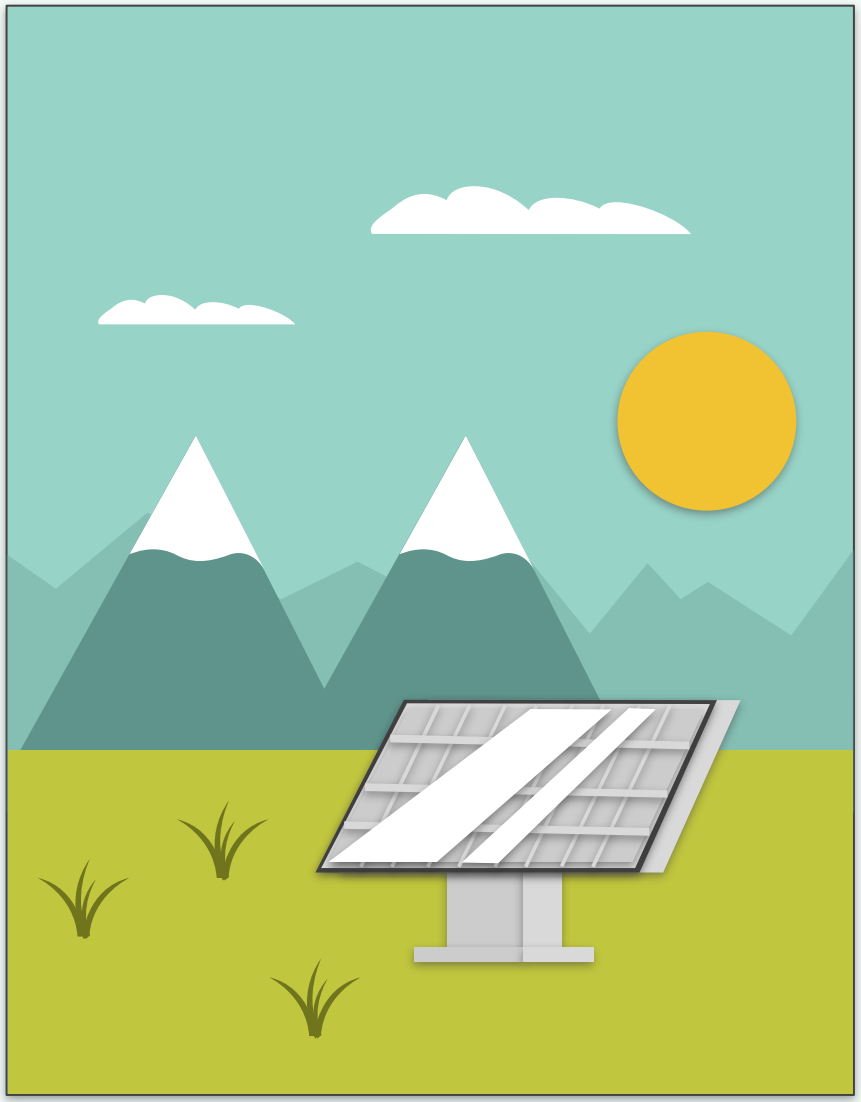
- Estimated capacity for each PV
- Plotted the plants' relative performance ratio
- Identified the amount of invalid/faulty data

Qualitative:

- Explained the reason of invalidity
- Explained seasonality in the data

Additional:

- Outlined other details about the parks
- Estimated their apprx. location



Thank you for your attention!

Do you have any questions?