

new

October 4, 2022

1 Import libs that we'll need for the project:

```
[ ]: import pandas as pd
import matplotlib.pyplot as plt
from numpy import percentile
import numpy as np
from sklearn.preprocessing import MinMaxScaler
import datetime
import matplotlib.dates as mdates
import operator

from funcs import *
```

2 Split the project into steps

1. Read the file
2. Remove Outliers
 1. Quartiles
 2. Deviation
 3. By hand in special cases
3. Normalization and standardization of the data
 1. Z-Score
 2. Min Max
4. Split the data into train, test and validation sets
5. Train the NN
6. Adjust parameters

2.1 Read the file and analyze data:

```
[ ]: data_df = pd.read_csv("Proj1_Dataset.csv", sep=",", decimal=".")

data_df=parse_date_time(data_df)

data_df.to_csv("Proj1_Dataset_changed.csv")

#plot_data(data_df, temperature=True, CO2=True, PIR=True, light=True)
```

```
#scatter_plot(data_df, temperature=False, CO2_PIR=False, light=False)
```

3 Remove Outliers and Interpolate data

```
[ ]: print(data_df.describe())
```

While using 'describe()' we realized the following:

- The dataframe is 10129 rows long and some values return a count of, for example, 10127 rows, therefore we need to fill in the data where this values are missing
- If we look closely to the mean and max values of each column we realize that there are some outliers due to the discrepancy of the values. Due to the ammount of data we'll simply drop the entire row where the value is found.

3.0.1 Drop Outliners

```
[ ]: data_df = drop_outliners(data_df, threshold=6,  
                             column_to_remove_outliers=  
                             ["S1Temp", "S2Temp", "S3Temp",  
                              "CO2", "PIR1", "PIR2", "S1Light",  
                              "S2Light", "S3Light"])
```

3.0.2 Interpolate NaN

```
[ ]: # interpolate NaN values  
data_df = data_df.interpolate(method='linear', limit_direction='forward',  
                               ↪axis=0)
```

4 Normalize data

```
[ ]: # Normalize data with MinMaxScaler
```