Advanced Data Analysis and Machine Learning

INTEL Wireless Sensor Network

Level A2

Martin Kostelník, Marianne Jakonen, Ahmed Ansari

Introduction

The dataset[1] contains data collected from 54 sensors deployed in the Intel Berkeley Research lab. The collection period ranges from 28.2.2004 to 5.4.2004 and contains a total of roughly 2.3 million data samples. Each sensor collects data every 31 seconds and the data collected are: temperature, humidity, light and voltage. Each sensor also has a unique ID and stores the data with a timestamp and measurement ID.

Modelling goal

As per the assignment, the end goal is to use PCA to detect seasonality in the data. This will be done on both weekly and monthly basis. We will also study the correlation between the individual variables and create a dynamic model to predict future values for the most correlated variable. When we have this model, we can calculate how much ahead we can predict, it's window frame and the optimal sampling frequency.

Data summary

As described in the introduction section, we have roughly 2.3 million data samples from 54 sensors placed at a different location. The scheme of the data is shown in the following table..

Name	Data Type	Unit
Date	String	-
Time	String	-
Epoch	Int	-
Mote ID	Int	-
Temperature	Float	° C
Humidity	Float	%
Light	Float	Lux
Voltage	Float	V

When loading the dataset, we transform the Date and Time columns. We combine these into one column containing both using the correct data type from pandas library.

Value ranges

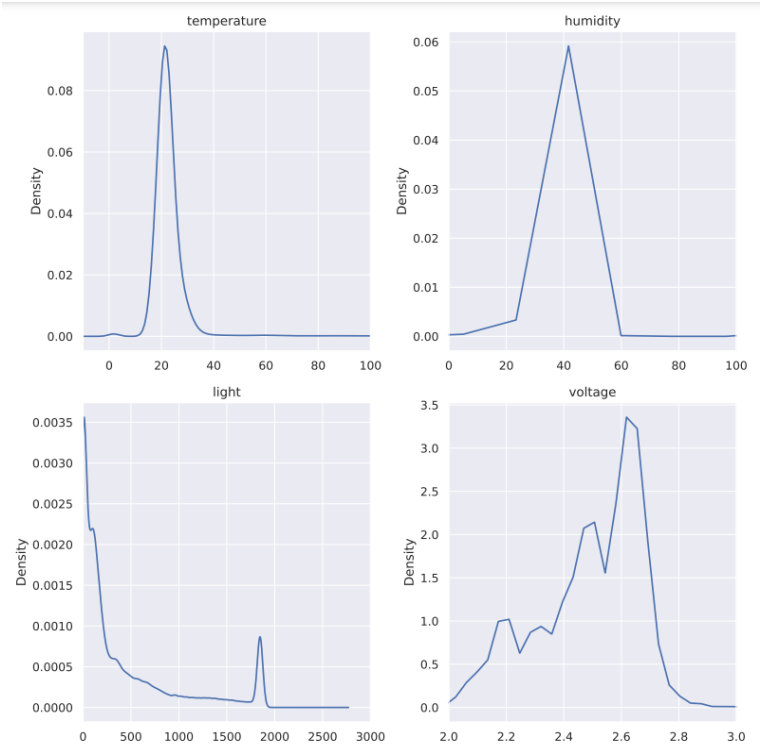
Now let's take a look at the value ranges of each column. We can find theoretical value ranges for some of the columns in the dataset documentation. There are no limits to the temperature column. We also find the actual value ranges present in the dataset for each column. These can be seen in the table below. We can see that for some columns, there are nonsense values that need to be removed and this also shows that there might be many outliers in the data.

[1] Dataset publicly available at http://db.csail.mit.edu/labdata/labdata.html

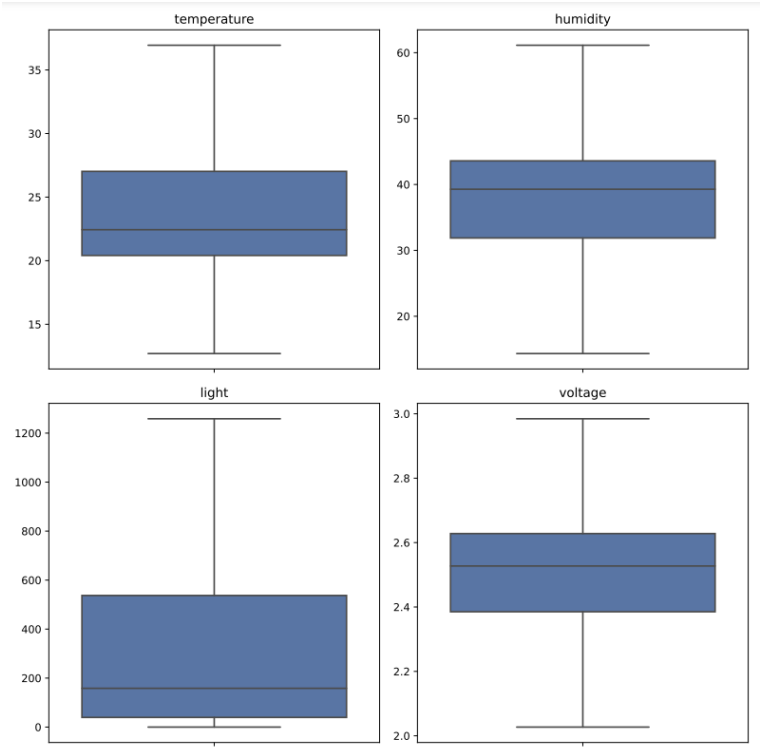
Name	Min_T	Max_T	Min_R	Max_R
Date	28.2.2004	5.4.2004	28.2.2004	5.4.2004
Time	00:00	23:59	00:58:15	11:02:32
Epoch	0	-	0	65535
Mote ID	1	54	1	65407
Temperature	-273.15	-	-38.4	385.568
Humidity	0	100	-8983.13	137.512
Light	0	-	0	1847.36
Voltage	2	3	0.009101	18.56

Distribution

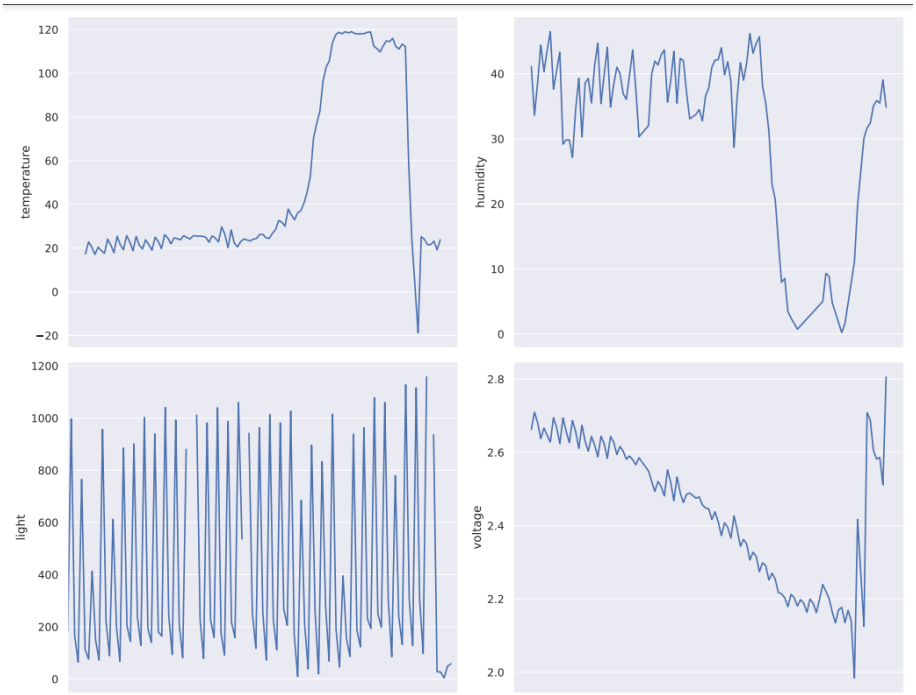
There are four columns that contain numerical values. These are temperature, humidity, light and voltage. For now, we are only going to be interested in those columns. Let's first take a look at the kernel density estimation across the value ranges of each of these columns. Note that data were cleaned of outliers behind the theoretical ranges.



Now visualize the distribution using boxplots. We can more clearly see where the mean sits and it seems to be around the usual values for all variables.



At last, let's take a look at the trend in the data over the measuring period. To create this figure, the dataset was averaged over 8 hour periods. We can see a spike in temperature and change in the humidity at around the same time. Light amount seems to be changing based on the daily activity in the office. And lastly, as expected, the voltage in the batteries has a downward trend until the batteries are probably replaced.



Data preprocessing:

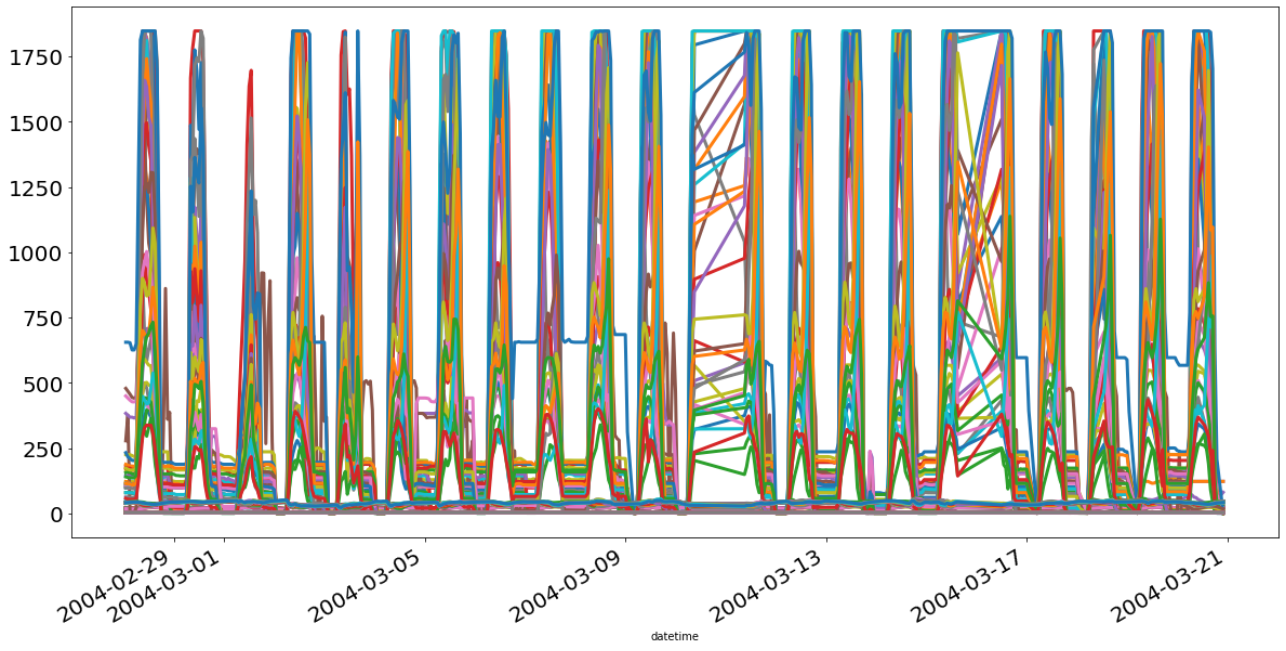
At first some basic filtering was performed on dataset based on theoretical minimum and maximum values of different variables to remove outliers. For example theoretical maximum for humidity measurements is 100% so all values above this were removed.

Data was then rearranged by sensor position id (”mote_id”) and measured variable (temperature, humidity, light, voltage). Dataframe was sorted based on datetime variable.

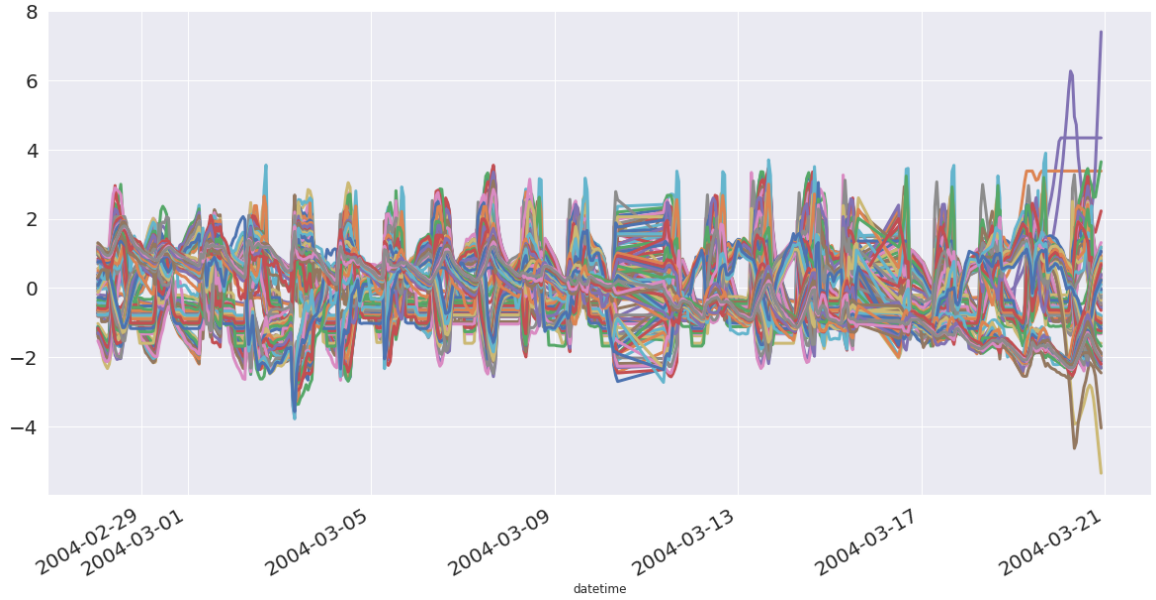
Original dataset has 2313682 rows and 58 columns (variables), measurement interval is 31 seconds. To reduce the size of Dataset it was resampled using 2H sampling interval. Missing values were removed from dataset as follows:

- Rows where all values were missing were dropped
- Rows where more then 80% of data was missing were dropped
- Colums where more than 10 values were missing were dropped

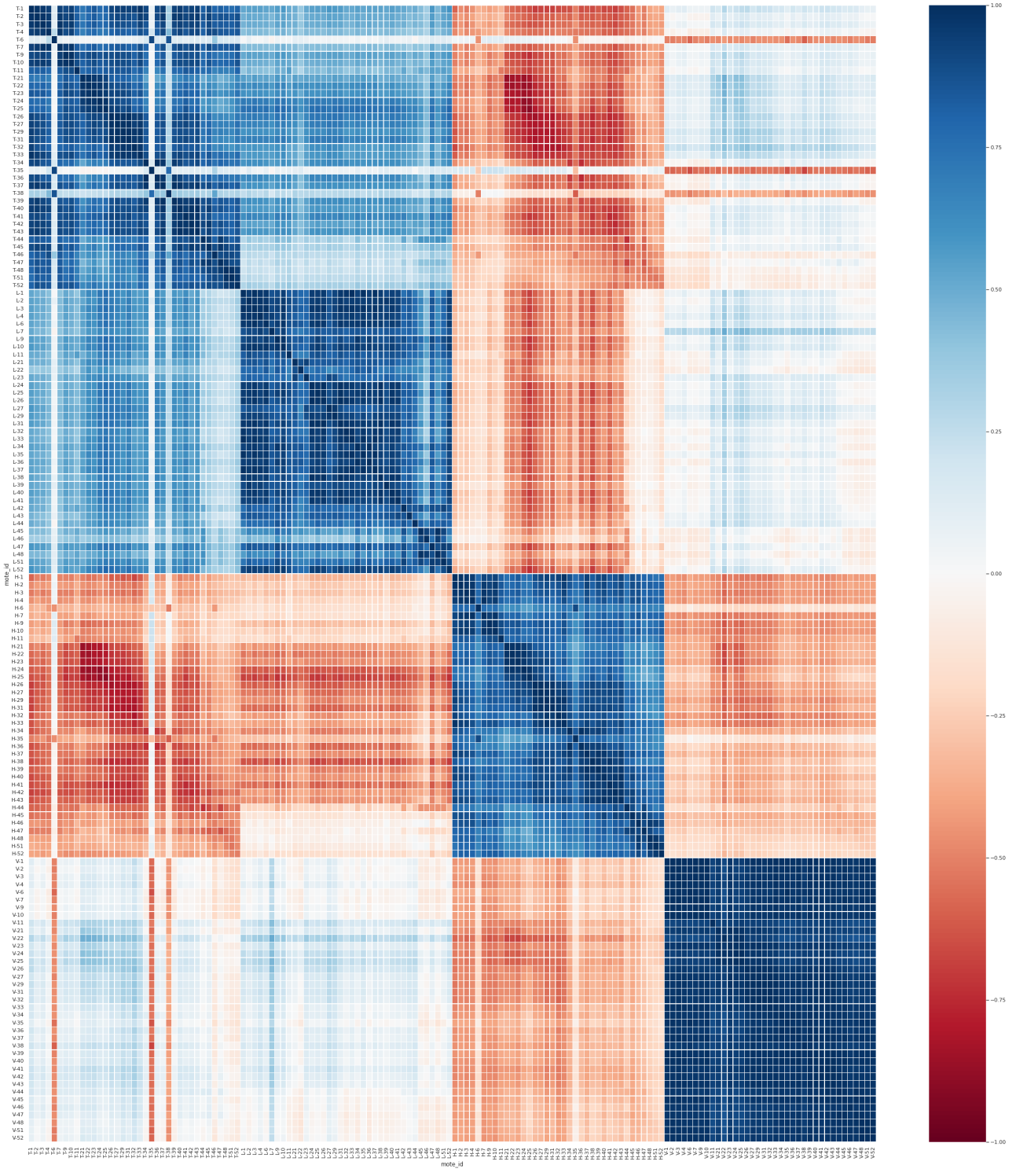
The dataset was reduced to 461 rows and the amount of variables was increased to 158 as a result of rearranging the dataset by sensor positions. Remaining missing values were interpolated. After removing missing values, the data was plotted. As seen from the figure, the different variables are in different scales.



Data was scaled and centered by using standardization, where the mean is reduced from observations and then divided by standard deviation. The result of standardization is seen in the figure below.


8
6
4
2
0
−2
−4
2004-02-29
2004-03-01
2004-03-05
2004-03-09
2004-03-13
2004-03-17
2004-03-21
datetime

Correlation matrix

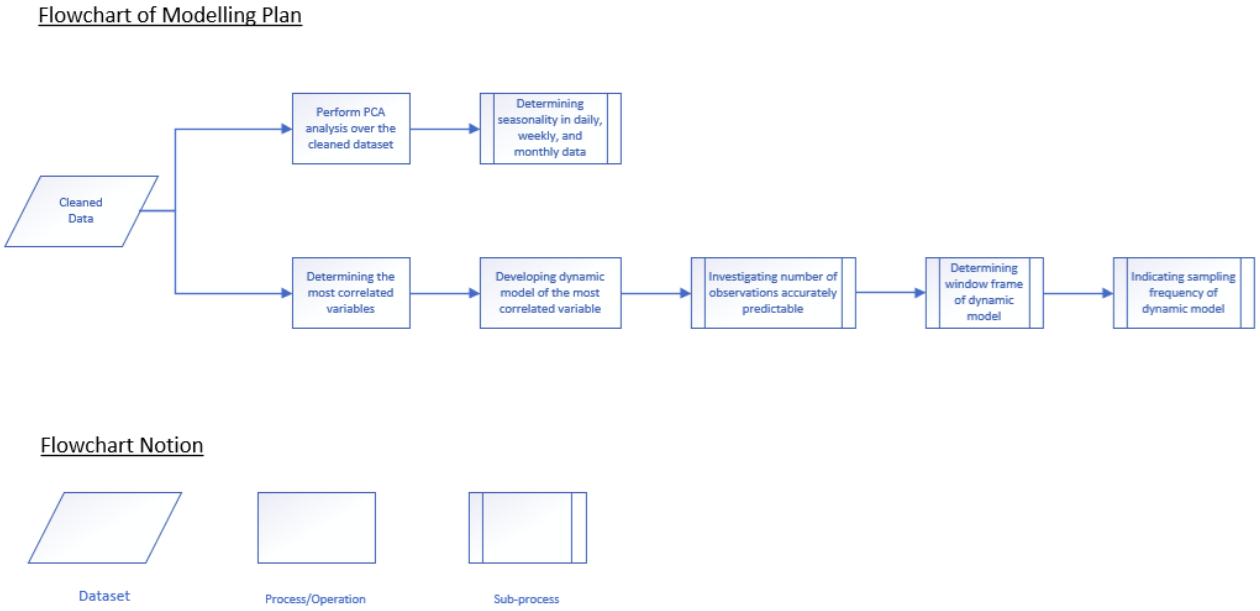


Modelling plan:

Goal of the project is to create dynamic models that can be used to predict future values of observed variables. The seasonality in data is investigated with PLS on weekly and monthly basis. Aim of the modelling is to:

a) Define the timeframe in which predictions can be made accurately
b) Define the dynamic model window frame
c) Define the maximum frequency of sampling that still allows to make accurate predictions

The modelling plan is illustrated in the figure below.



Seasonality detection

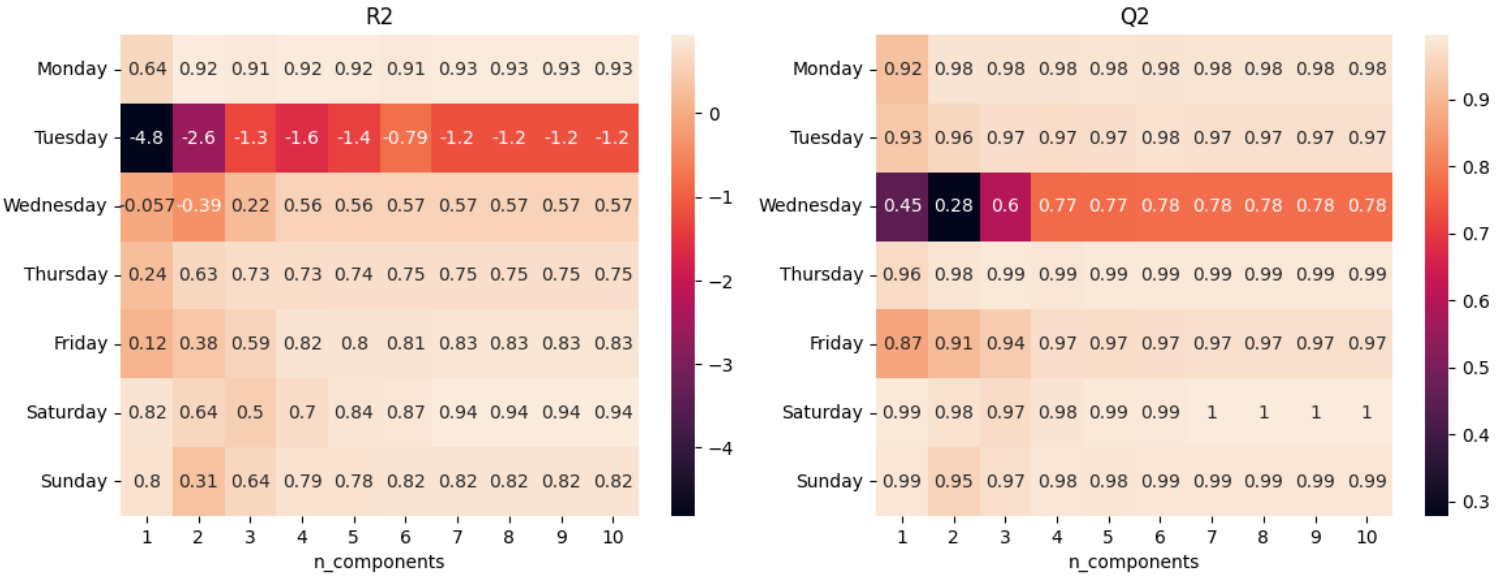
To detect the seasonality in data, a two hour resampling frequency was chosen. This assures that there are very few missing values. The missing values that persisted were then interpolated. Once we have the data ready, we have to find the most correlated variable. Five most correlated variables are in the following table.

H-26	92.74
H-30	92.09
H-31	91.16
H-27	90.42
H-29	89.72

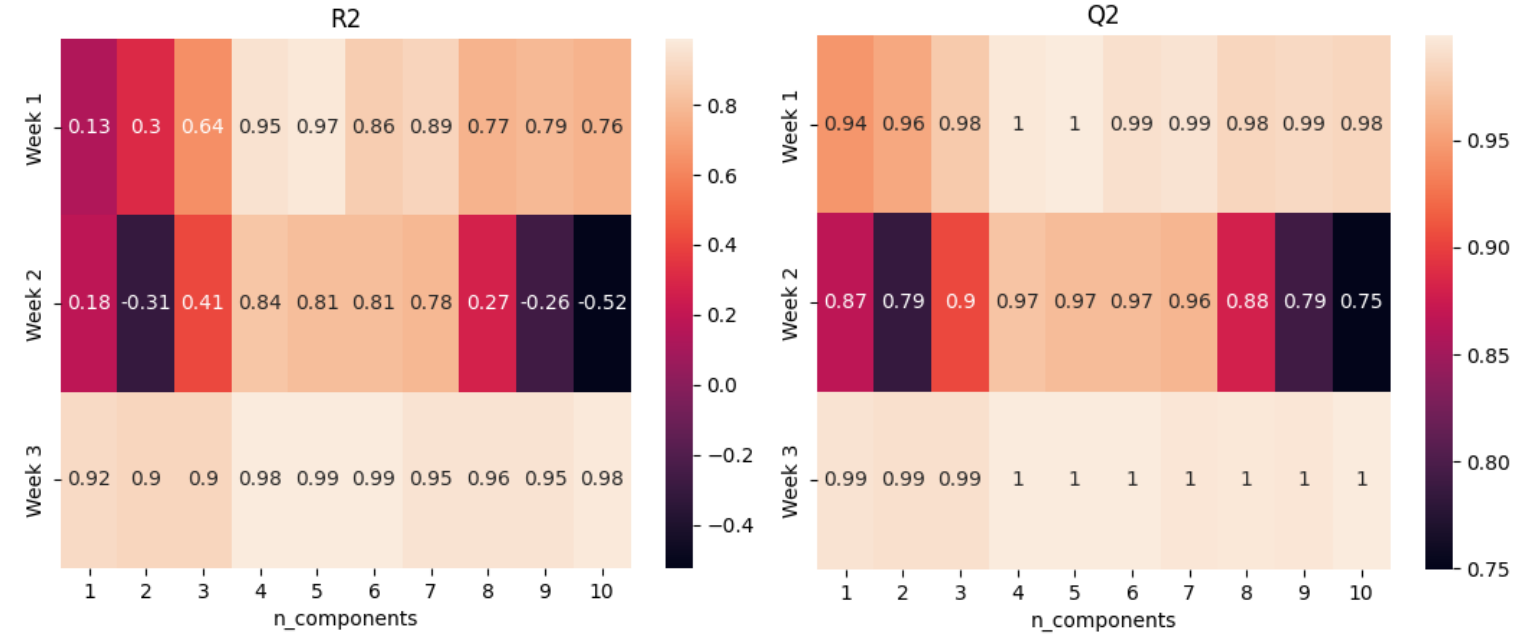
We can see that the most correlated variables are humidity. Which would make sense considering it is tied closely to temperature. Let's choose the humidity of sensor 26 as the variable we will be predicting.

After cleaning, we have data spanning from 28.4.2004 to 20.3.2004. I selected the week starting from 1.3.2004, which is Monday. We will use PLS for the prediction and we will try different number of components (1 to 10). Since the data was resampled on a two hour basis, we only have 12 measurements for each day. I decided to use an 8/4 train/test split. The train data was scaled and the mean and std values were used to scale the test data.

The PLS model was fitted with the training data and then used to predict the testing data. Once we had the predictions, R2 and Q2 scores were calculated and they can be seen in the figures below.



Now for the monthly seasonality. Since we only have data from 28.2. to 20.3., we can create three week long windows. The approach is then the same as in weekly analysis. The R2 and Q2 scores are in the figures below.



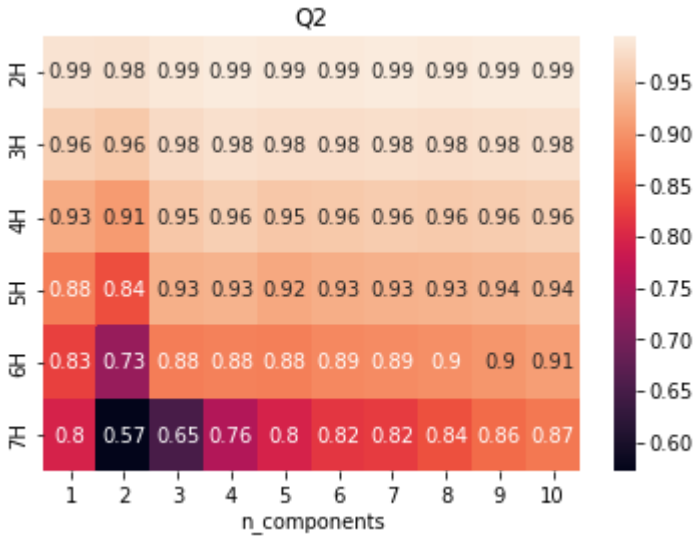
In conclusion, when we take a look at the R2 plot in the weekly seasonality, Tuesday data fit very poorly into the model. However we were able to make very accurate predictions for it according to the Q2 values. Wednesday has seen a better fit than Tuesday but also worse than the other days. On the other hand, the predictions are way worse. We can say that there is a seasonal factor during the selected week and the tuesday and wednesday data differ from the rest.

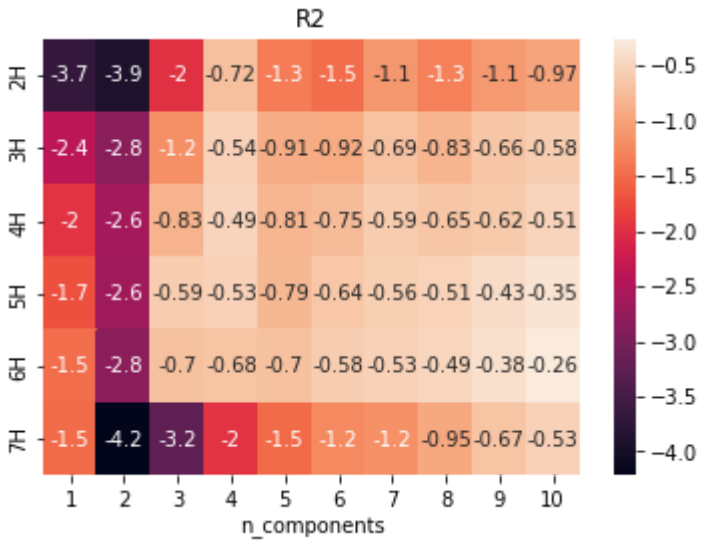
The monthly analysis shows a not so good fit for week 1 with low or high number of components, but improves when we look at the models with 4 or 5 components. The fit for the second week is even worse but again, it improves using the right amount of components, as does the prediction performance.

As a whole, we detected seasonality only for two days in the weekly analysis (one if we only consider prediction quality) and a little bit of seasonality in the monthly analysis. If we select the right number of components, which would be 6 in our case, we might be able to use one global model for all data.

Prediction accuracy for future predictions:

Prediction accuracy for future predictions was tested and evaluated for 2-8 hour predictions. Most correlated variable H-26 was used as y variable. Length of calibration set was 24h. Number of latent variables was 1-10. Q2's and R2's were plotted for the different combinations of latent variables and lengths of prediction periods.




R2
2H
3H
4H
5H
6H
7H
-3.7 -3.9 -2 -0.72 -1.3 -1.5 -1.1 -1.3 -1.1 -0.97
-2.4 -2.8 -1.2 -0.54 -0.91 -0.92 -0.69 -0.83 -0.66 -0.58
-2 -2.6 -0.83 -0.49 -0.81 -0.75 -0.59 -0.65 -0.62 -0.51
-1.7 -2.6 -0.59 -0.53 -0.79 -0.64 -0.56 -0.51 -0.43 -0.35
-1.5 -2.8 -0.7 -0.68 -0.7 -0.58 -0.53 -0.49 -0.38 -0.26
-1.5 -4.2 -3.2 -2 -1.5 -1.2 -1.2 -0.95 -0.67 -0.53
1 2 3 4 5 6 7 8 9 10
n_components
−0.5
−1.0
−1.5
−2.0
−2.5
−3.0
−3.5
−4.0