Report team 22

BBoxx Lithium-Ion Battery Field Data Challenge

Participants: Victor Bossard, Etienne Beauchamp, Elias Galiounas, Maria Varini, Nicole Schauser

Introduction/Goals

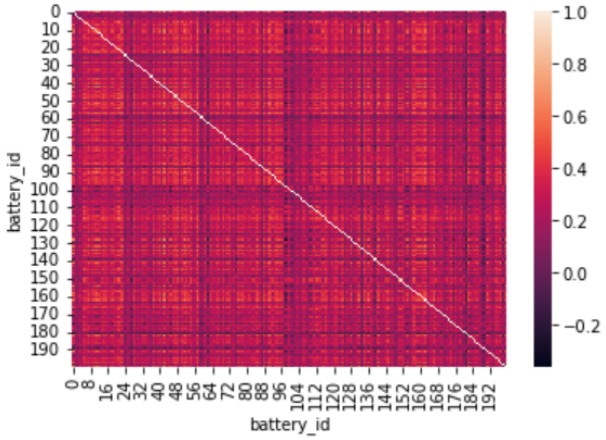
BBox operates off-grid Lithium-Ion Batteries in combination with solar panels in sub-Saharan Africa. Battery data of those systems is transferred over the air. Understanding how people use these batteries is essential for the development of future systems and also to understand battery degradation better.

The core of this challenge is the characterization and clustering of usage profiles. The usage of the devices plays an important role to understand how the batteries age.
Goal 1: find the number of usage groups from the dataset.
An excellent approach should allow being applied on many thousands of devices with reasonable computational complexity.
Goal 2: Based on these profiles come up with a method to generate new operational profiles based on the dataset.
Goal 3: visualization of the raw data and/or your approach and/or your results.

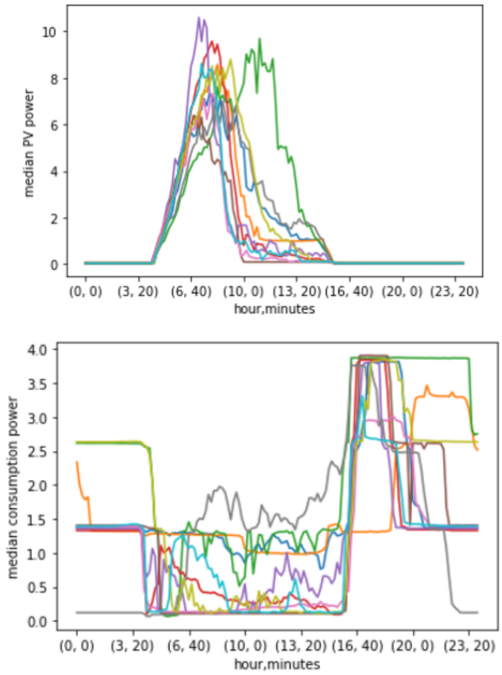
Data Discovery

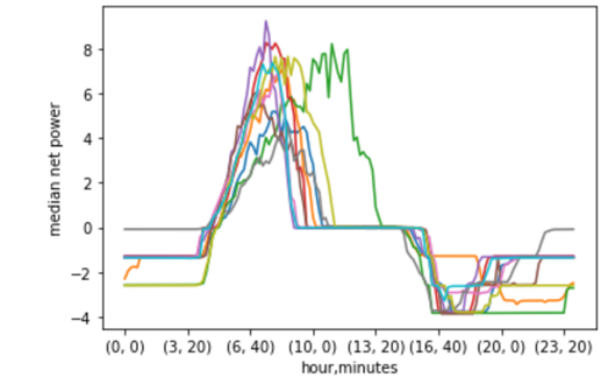
The dataset is composed of a multivariate time series, collected from 200 batteries being operated in sub-saharan Africa over a period up to 7 months. Those batteries are coupled to a photovoltaic panel (PV) and are used to store electricity for home usage.
The dataset is rather large: ~20 million observations of different measurements, namely battery voltage, current, temperature and state-of-charge, PV current and voltage, as well as consumption data. The data are sampled frequently, normally every 2 to 10 minutes for each battery.
From the data available, we are able to derive the power generated by the PV, and the power taken by the end-user.
Visualization of the raw data is rather complicated. However, it can be expected that the data from different batteries are in fact highly correlated and with strong daily patterns; this is due to the shared day-night cycle among the users (which are located roughly in the same geographical area), and consequently their living habits (time of wake, time of sleep).

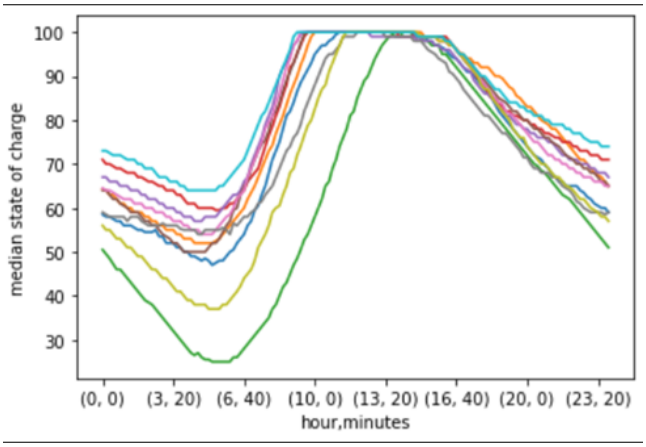
This is what we observe from the plot below, showing the correlation (Pearson one) of the consumption power time series. The mean correlation is close to 0.3. Few plants are negatively correlated to the other, and we could from now on split the former from the latter as two separate groups.



From the data, a median daily profile of consumption power, PV power, net power (where positive is for charging and negative for discharging) can be derived and plotted as follows.

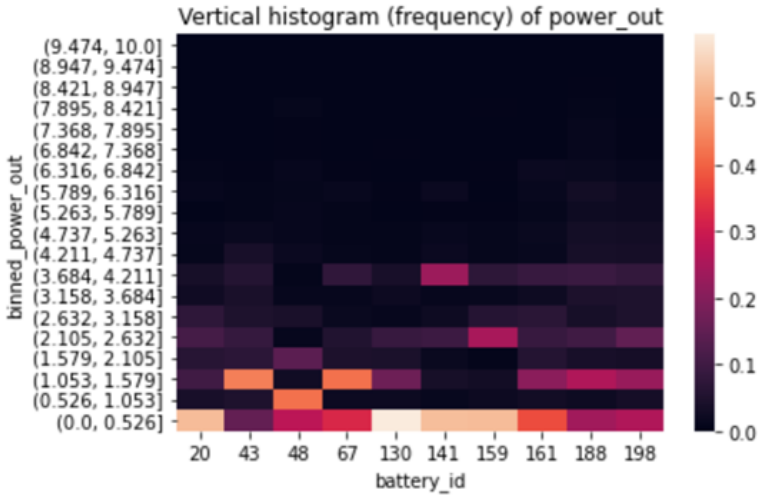




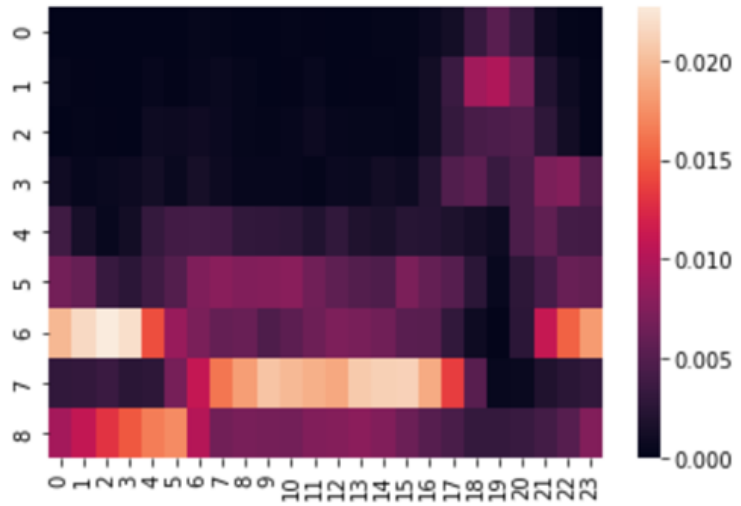


In these plots, some median consumption power profiles stand out: for example, in some instances of operation during night, large power spikes, low consumption during daylight, and noisy patterns can be found. Accordingly, we can see interesting differences in the median state of charge.

From this visualization only, the dispersion of daily profiles around the median cannot be inferred directly. If for instance we are interested in the power taken by the end-user, we can study the distribution of values for each battery. But in this way, time patterns are lost. For each battery, we can bin the time-series by power_out and by time of the day, to generate a histogram of values assumed by this battery along the day.

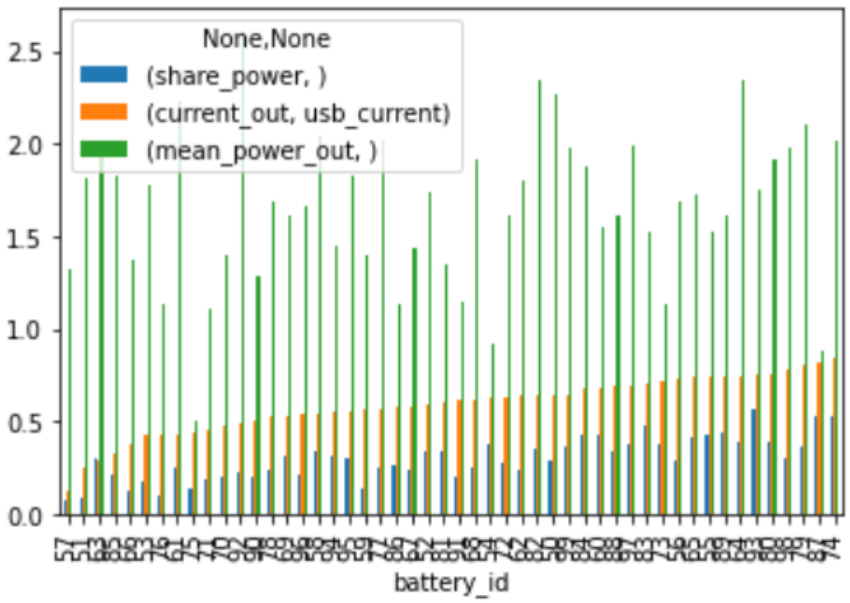


2D histogram (frequency) of power_out for one battery



From those plots, some “power modes”, or “power_level” emerge, which differ according to the usage of each battery. Our assumption is that those modes are linked to the appliances used in the house.

The only additional information about those “power modes” is the field ‘usb_power’. In fact ‘usb_power’ is contained in ‘power_out’, meaning that ‘power_out’ = ‘usb_power’ + ‘an_other_power’. We can for instance compute the Pearson correlation between usb power and power_out. This correlation (in orange below) ranges from 0.1 to 0.8. The share of usb_power or power_out (in blue below) ranges from 0.1 to 0.5. The difference in usage according to each household emerges once again.



Unfortunately, not much else can be inferred purely based on data analysis and visualization, and for this reason, several approaches are documented in the following sections to further extract information from the data.

A) Static approach using aggregate statistics

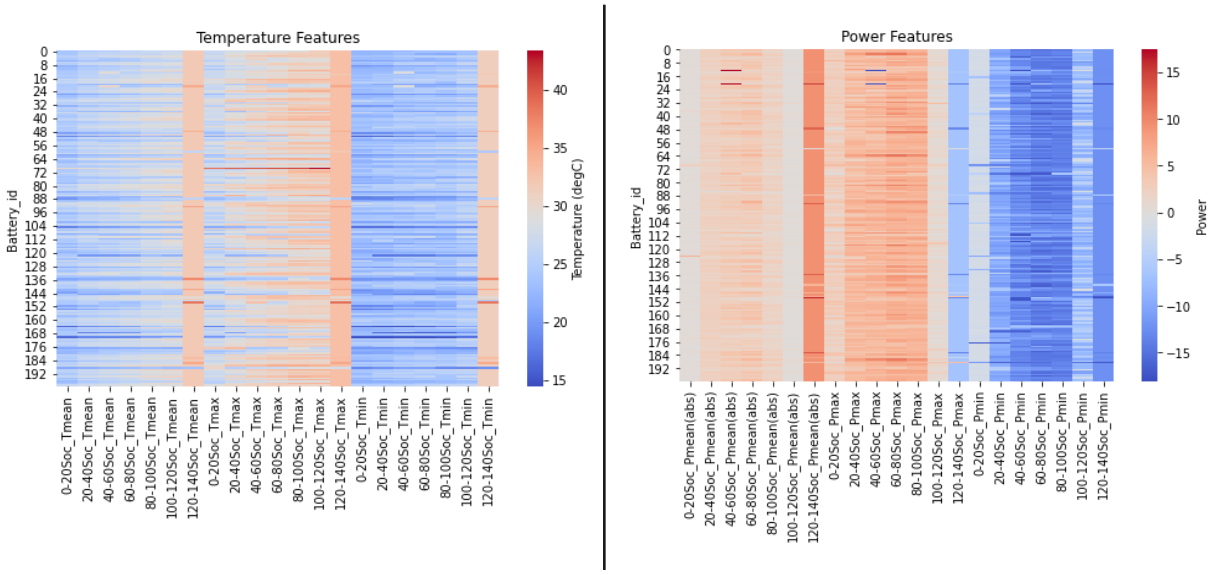
It is scientifically known that battery operation at extreme states-of-charge, either close to the top or bottom-of-charge, is most damaging to battery cells. High temperatures and power loadings are also known to have a strong influence on degradation. With a view towards extracting operating features which can be used to cluster batteries according to their propensity to degrade, this approach considers aggregate statistics of temperature and power at various SoC bins. In essence, the time dimension of the data is disregarded and for every cell the following parameters are obtained:

- At bins of 20% SoC, from 0 to 140% (i.e. 7 bins)
    - Mean temperature ‘Tmean’ (with duration-based weighing)
    - Maximum temperature ‘Tmax’(99th percentile)
    - Minimum temperature ‘Tmin’ (1st percentile)
    - Mean absolute power ‘Pmean’ (with duration-based weighing)
    - Maximum power ‘Pmax’ (99th percentile)*
    - Minimum power ‘Pmin’ (1st percentile)*

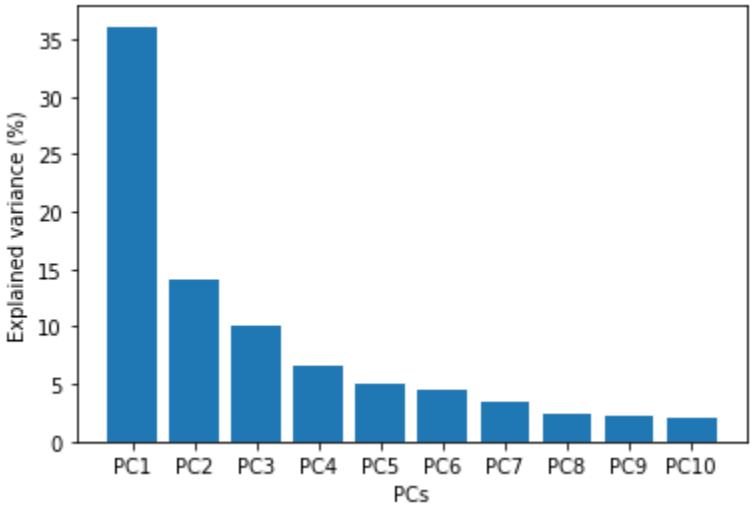
* it should be noted that positive power is discharging and negative power is charging. Maximum and minimum power values tend to be positive and negative respectively, with the exception of certain SoC bins.

Feature visualization and dimensionality reduction

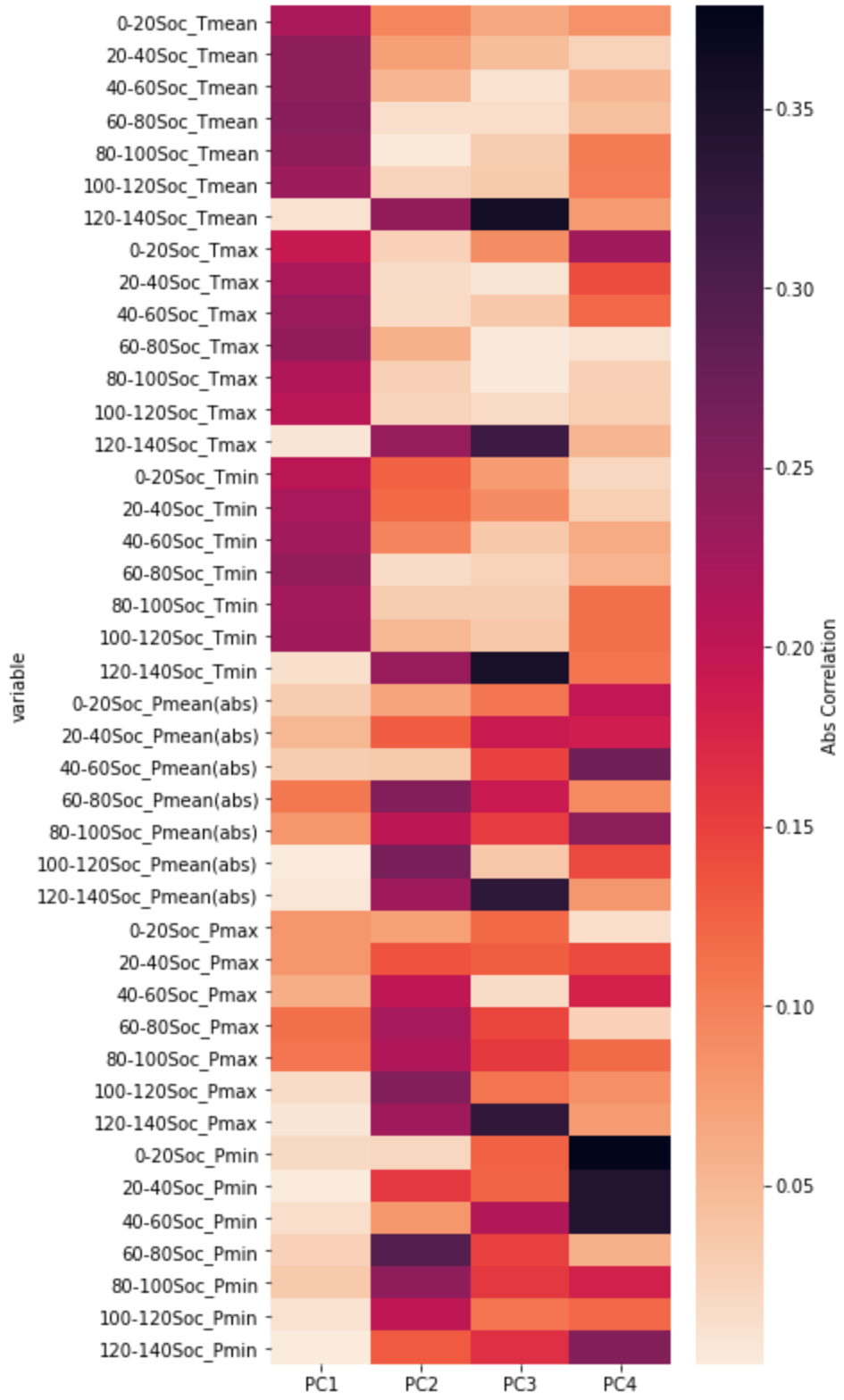
A total of 42 features were extracted from the raw data for each battery as demonstrated below.



To reduce the dimensionality of the problem, Principal Component Analysis (PCA) was performed on the features. The first 10 Principal Components (PCs) are plotted below against their explained variance (% of the feature variance they capture).



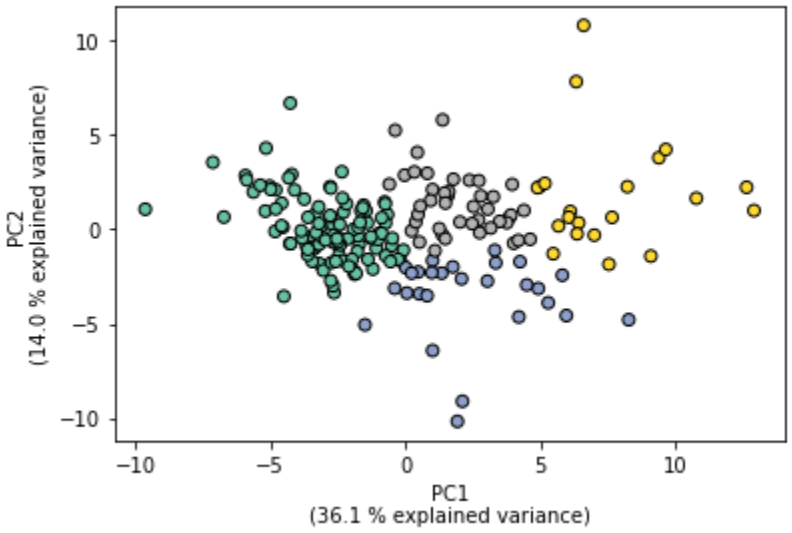
To understand how much each feature contributes to each PC, the correlation of individual features to individual PCs is plotted below. From the plot it can be seen that PC1 is mostly correlated with temperature-based features and PC2 is mostly related to power-based features.

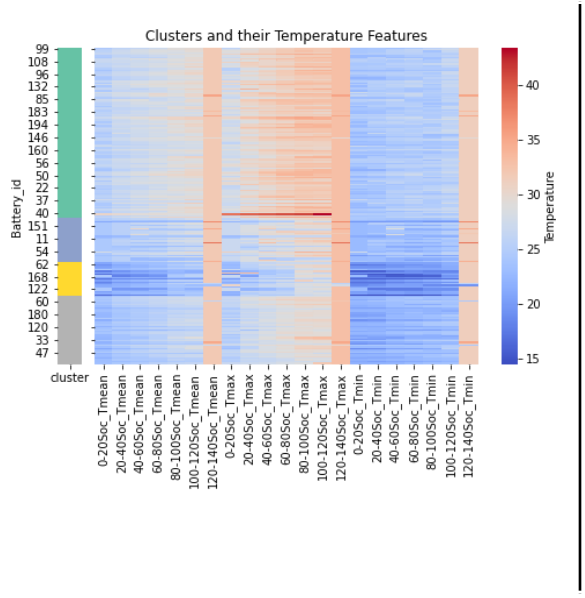


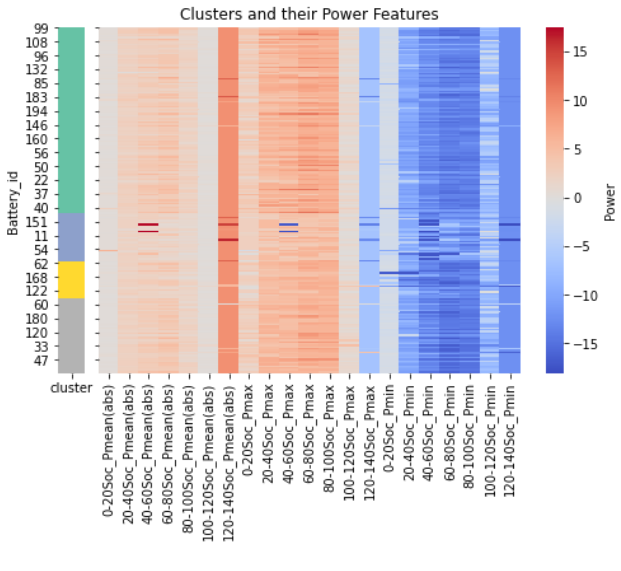
From the above evaluation it was decided to use PC1 and PC2 alone to cluster the batteries. These principal components together capture approximately 50% of the total variance in this

specific feature space, while they are also reasonably explainable having been linked to temperature and power respectively.

A K-means algorithm was run to produce 4 clusters. The number of clusters to compute was arbitrarily defined, and further developments of this work should aim to define a metric based on which the number of clusters can be optimized. The computed clusters and their PC1 and PC2 characteristics are shown below, followed by the temperature and power features grouped per cluster.







The following observations can be made from the clustered feature plots:

- The green cluster is linked with higher maximum temperatures.

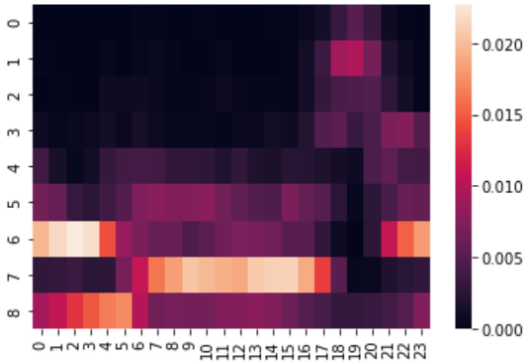
- The yellow cluster is linked with lower temperatures overall, especially lower mean and min temperatures.
- The blue and gray clusters have different temperature characteristics than the other two, but are similar to each other. They are differentiated by their power characteristics, as the blue cluster appears to contain a number of power spikes across mean absolute power, max power and mean power.

In the framework of the analysis performed, it can be expected that the clusters produced differ from each other in the impact they have on the lifetime of the batteries. Clusters representing usage under different temperatures emerge, where temperature is known to be a critical variable impacting the lifetime of batteries. Particularly at high temperatures, aging rates are prone to accelerate (in first approximation) following an Arrhenius-like trend. Additionally, clusters characterized by power spikes emerge, differentiating batteries that would otherwise belong to the same group due to temperature similarities. Power spikes are associated with the occurrence of momentarily high voltage, current, or both, and are also expected to accelerate the degradation of batteries that are subjected to them.

B) Approaches considering the time dimension

B1) Usage-based (power) only

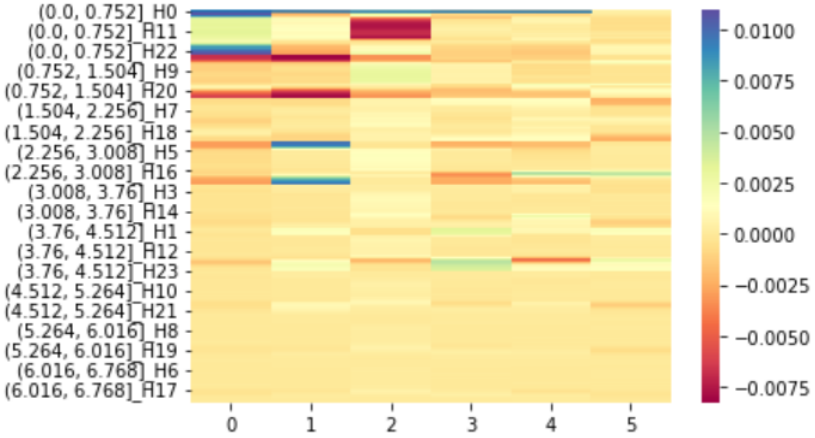
We tried to cluster only the usage. Remember that we can create from our data a histogram image of distribution on consumption power as a function of time of the day. For one battery, we bin a day per hour, and we bin power_out in 20 bins. Then for each day of the data we measure the frequency of observation of a data point inside the bins. We get an image like this.



Now we can have similar images for every battery. The question of clustering consumption data is now simplified to the question of clustering those images.

We can use PCA to reduce the dimensionality of the problem and give us insight about which part of this image is important and explains the variability of our dataset. We decided to keep only the first 6 principal components of the PCA, even if the 3 first are far more important

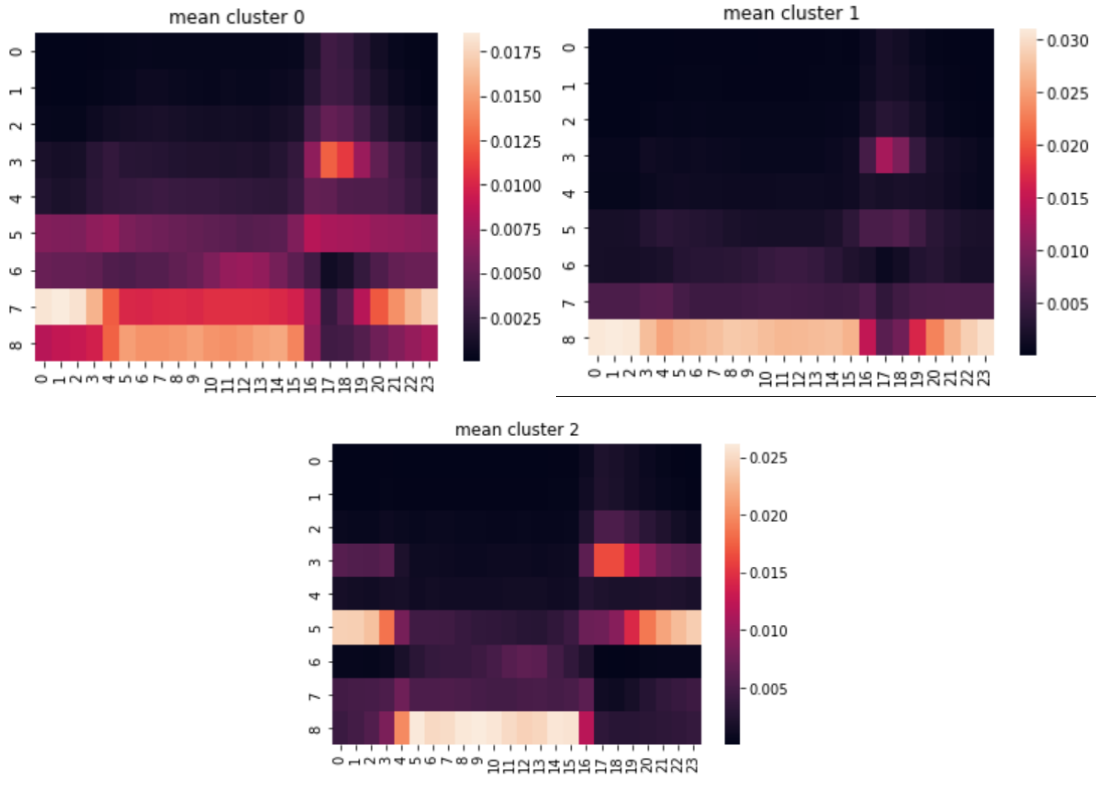
(explain +70% variance data). We can have an intuition about principal components of the PCA by looking at how our basic features project in the PC space.

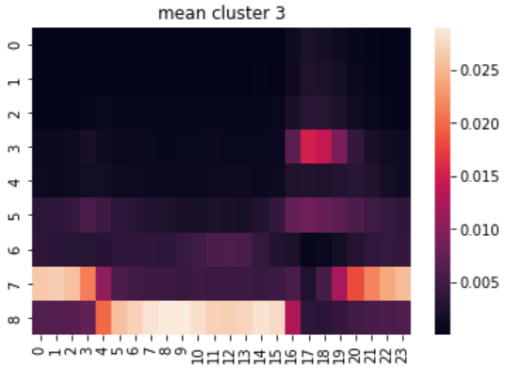


Here we can conclude that what explains most of the variability of the observed power_out are (from more important to least important):

- Values taken at night and early in the morning. Here again we can observe different modes. Some batteries don't deliver power at night, others do so at a constant level all night.
- Values taken in the middle of the day, with low power.
- Spikes of high power (cf PC3, PC4) that usually happens at the end of the day.

We are now able to cluster the consumption data into 4 groups (Kmean algorithm, elbow method). And we can show for each group the typical consumption profile.



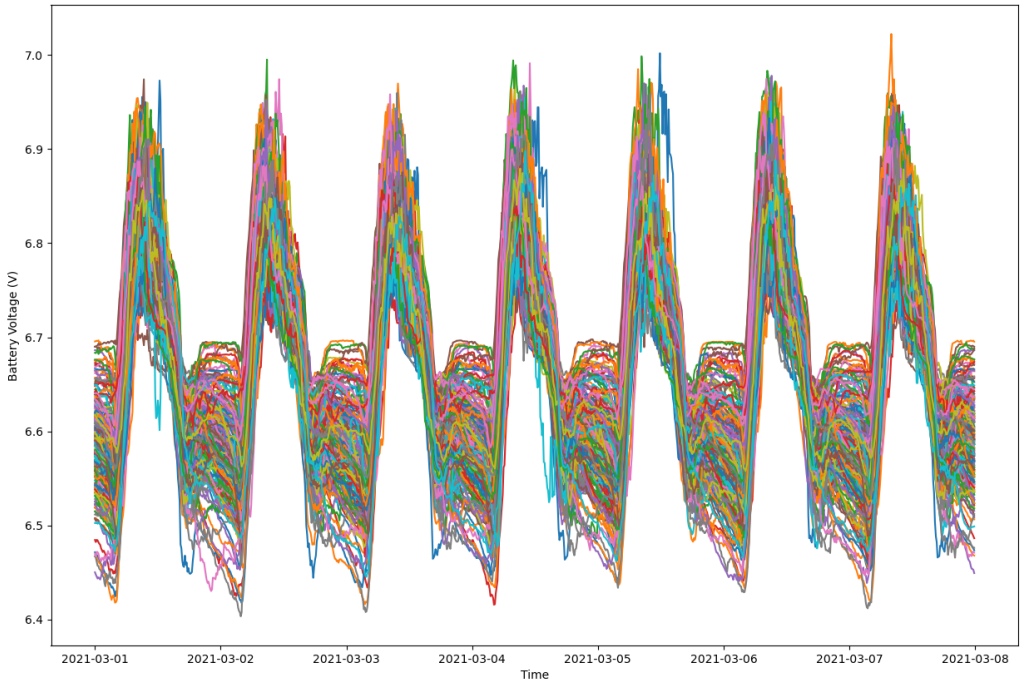


B2) Multivariate time-series clustering

Clustering techniques minimally use features from data to characterize its elements by partitioning it. In the case where the data is time-stamped, it is possible to use the temporal dimension of the data to find similarities in time-related trends. Such techniques are referred to as “time series clustering”.

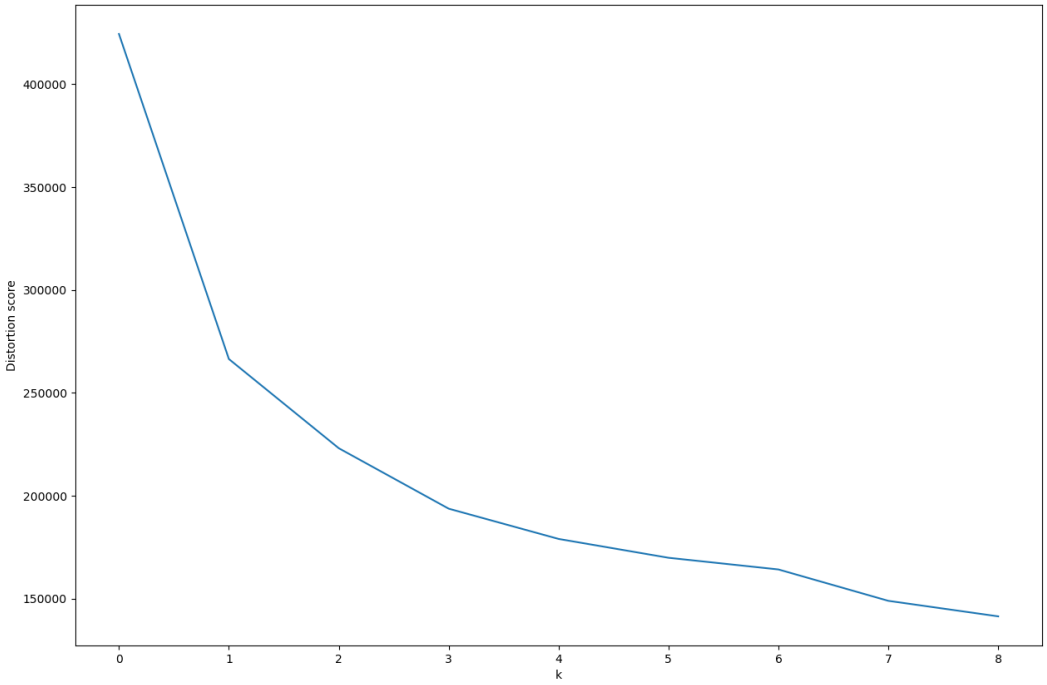
To rapidly generate results to be able to meet the deadlines of the challenge the k-means clustering technique, which is fairly popular and well documented, was used. While this algorithm is better suited for smaller datasets, it can be used as a first exploratory approach. This is what was done in the Multivariate Time Series Clustering notebook that can be found on the repo.

To perform such an analysis, the data must first be prepared. Data was resampled to a frequency of 10 minutes. This reduces the large quantity of data to analyze without losing too much information. Furthermore, this step makes the standardization of the timestamps easier afterwards.Data was then grouped according to the day of the week, the hour and minute when the data was sampled. This grouped data was then averaged to have an average week. The next figure shows the average of all the weeks combined for battery voltage for each battery id.

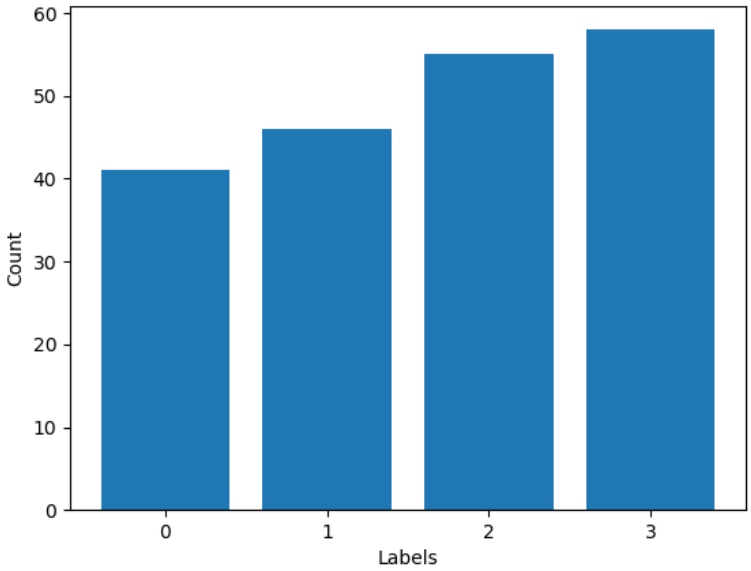


Data then needs to be of the same size. The longest time series was found amongst the data and its timestamps were used as a reference. All the other battery time series were reindexed to this longest time series with the nearest points. NaN were also replaced by doing an interpolation of data points surrounding the missing values.

In order to find the optimal number of clusters, the k-means algorithm was run with various numbers of clusters. The elbow method was then used to identify a somewhat good number of clusters. This naive technique computes what's called the distortion score which is presented in the following figure.

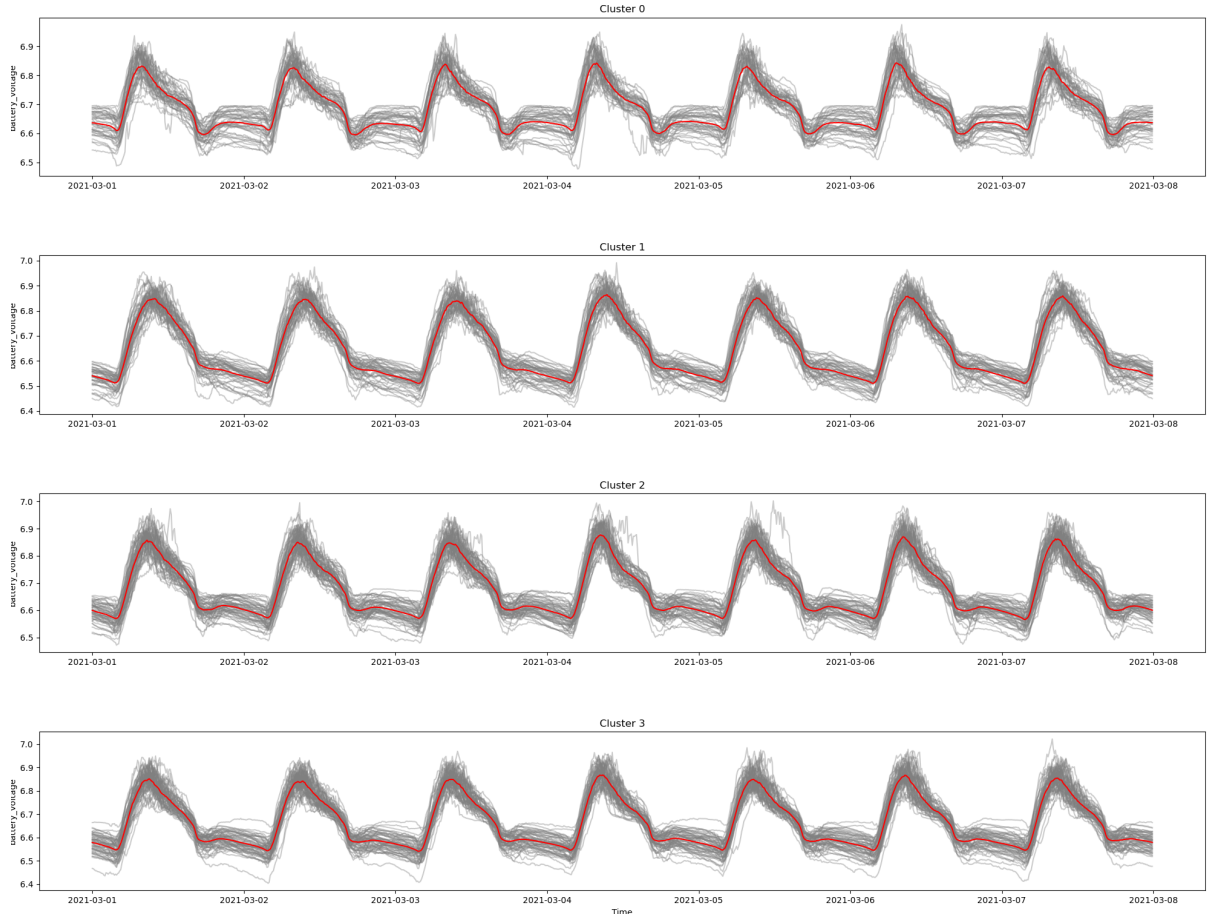


Using this graph indicates that 2 to 4 clusters should give decent results. The number of counts was then studied to make sure the labels were somewhat equally distributed.

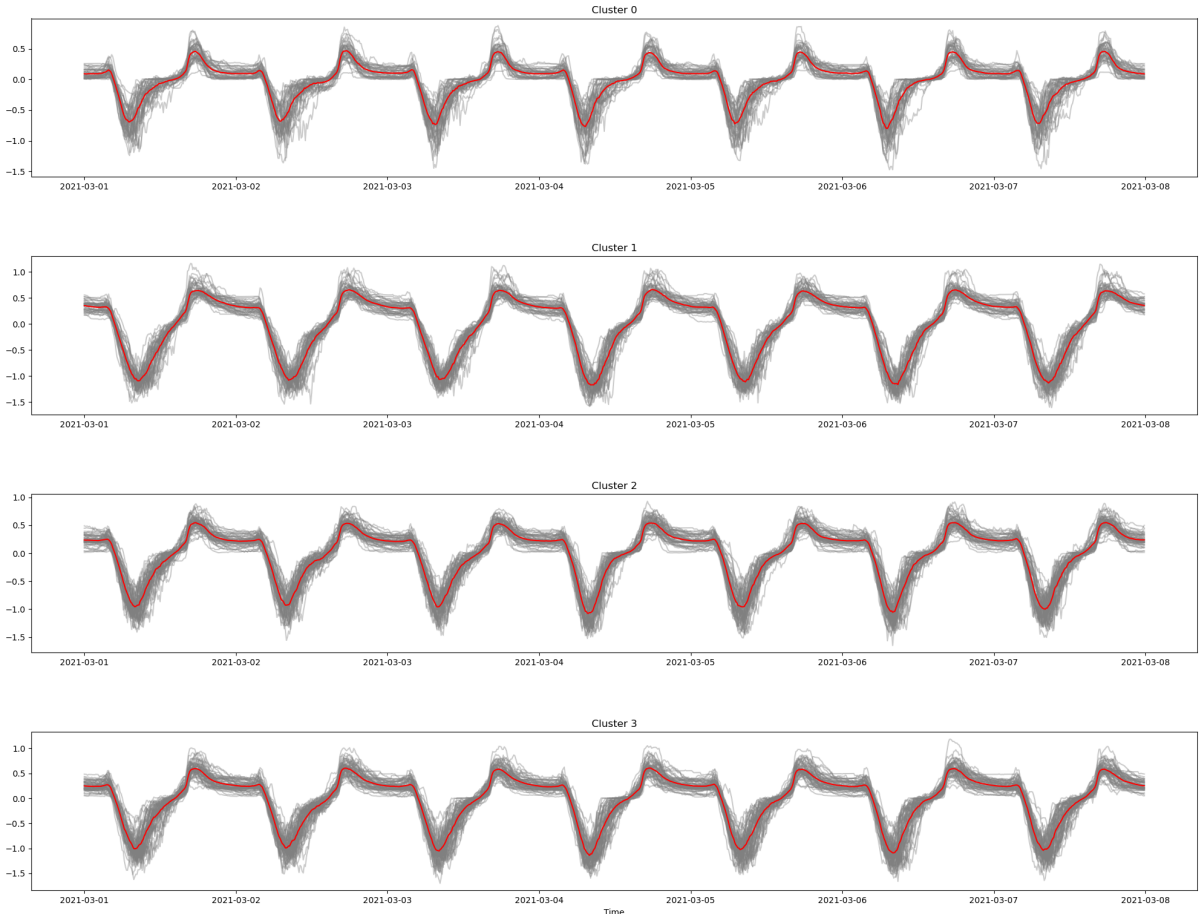


The mean of the studied variables for each cluster were then plotted to search for possible trends. Unfortunately, as it can be seen on the following figures (respectively voltage, current

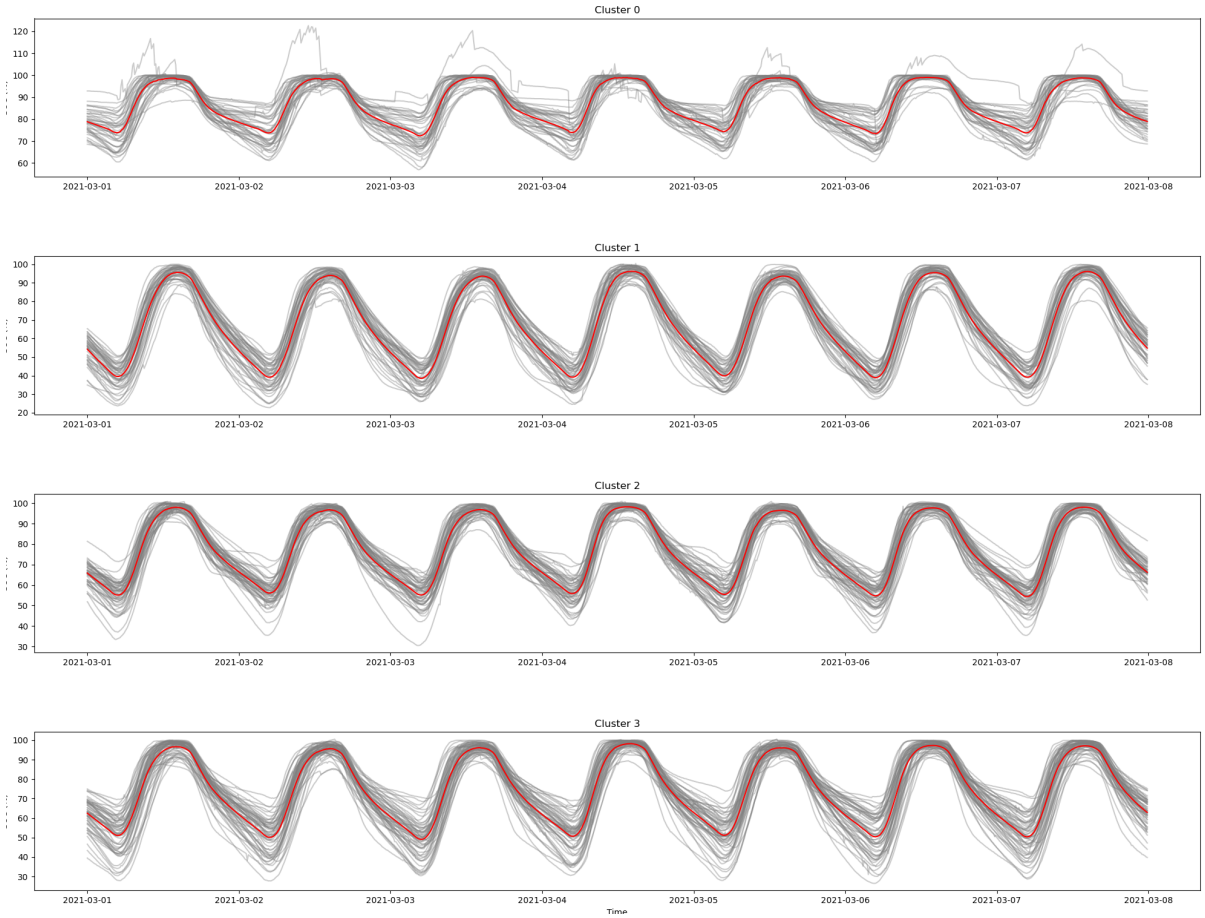
and SOC), trends amongst the clusters are fairly similar. Statistics were derived from these clusters to investigate further possible differences.



	battery_voltage		
id	mean	percentile_25	percentile_75
0	6.683488	6.64103	6.706876
1	6.647369	6.609991	6.685812
2	6.676917	6.63764	6.713175
3	6.662073	6.630844	6.704069

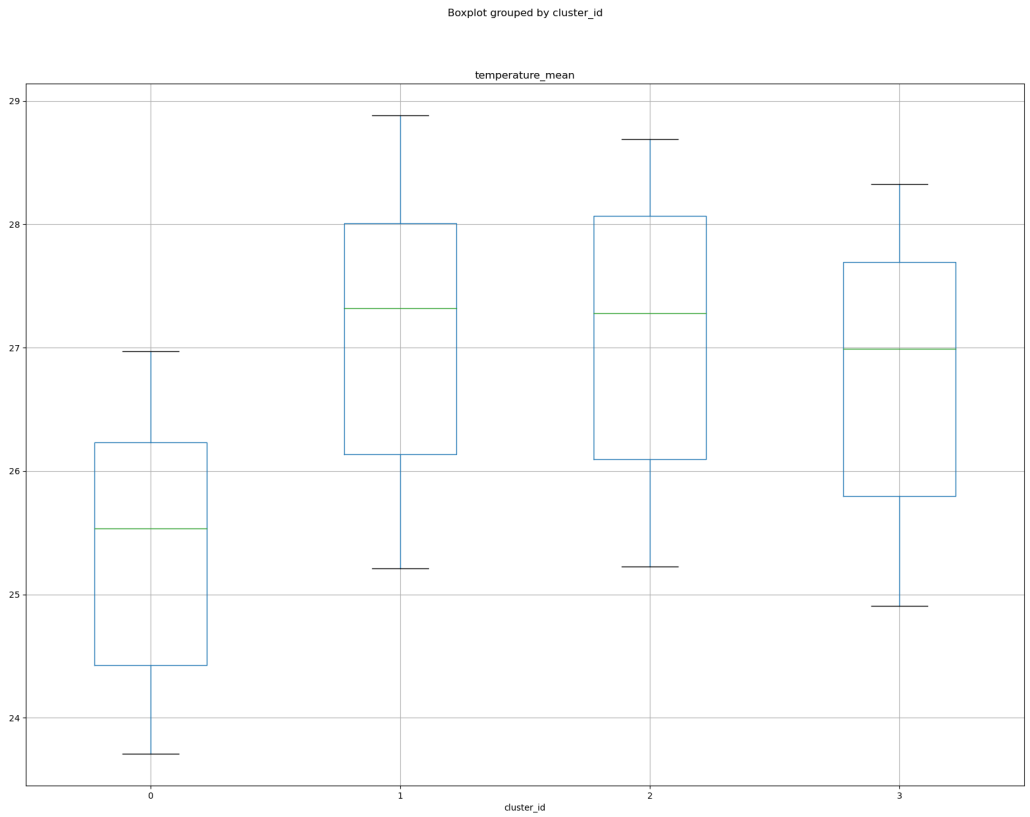


	current			
Cluster	mean	percentile_25	percentile_75	percentile_99
0	-0.014323	-0.156704	0.143333	0.410711
1	-0.007074	-0.197717	0.191273	0.529851
2	-0.007901	-0.173103	0.165093	0.460662
3	-0.007786	-0.180318	0.165192	0.46431



	SOC			
id	mean	percentile_25	percentile_75	percentile_99
0	86.818915	75.288854	87.225886	95.977454
1	67.680511	62.135433	75.89947	89.282317
2	77.23667	71.25229	84.173101	95.3729
3	74.415608	69.592011	82.411667	93.714692

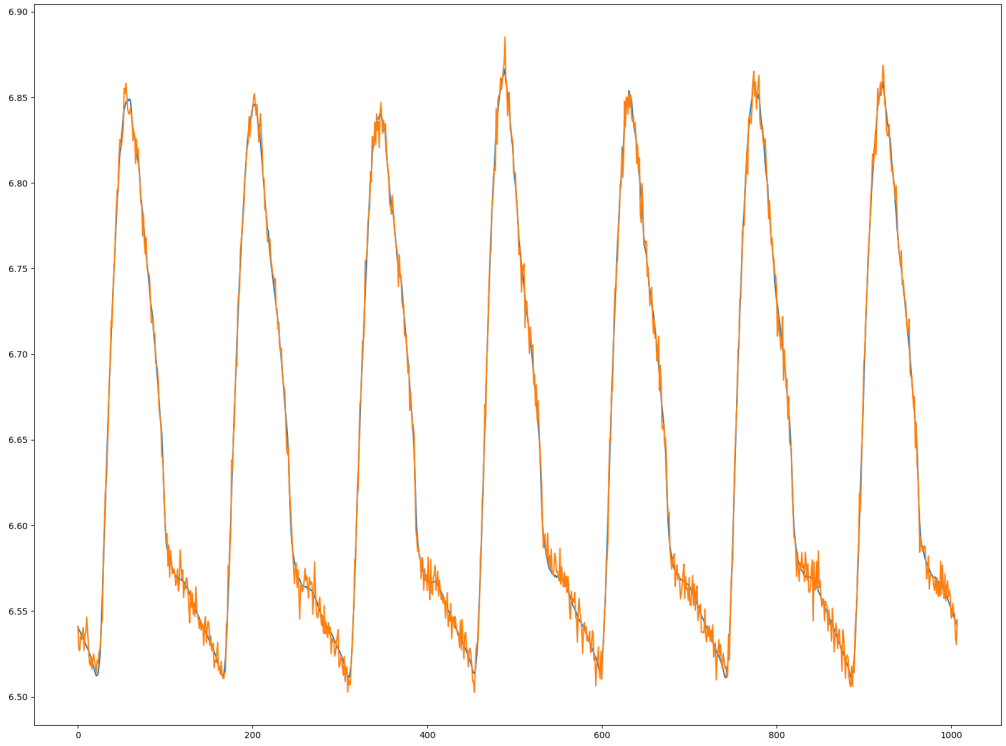
Temperature plays an important role in a battery's performance and state of health. Therefore, this variable was also studied. The next figure presents the distribution of average weekly temperatures for each cluster:



The first cluster is very different from C1, C2 and C3. This is especially interesting considering this cluster’s voltage, current and SOC is, in absolute, the highest on average.

Operating profile generation

It’s also possible to generate operational profiles from the clusters.With the same notebook (Multivariate Time Series Clustering) profiles can be randomly generated along a selected cluster’s mean values. The next figure shows an example of a weekly battery voltage generated from the first cluster.



Interactive Data Visualization

As a way to interactively display the results of the hereinabove documented analysis, an interactive dashboard was created using Jupyter Dash and dependent libraries. The reader is encouraged to run the relevant code in the submitted github repository, to generatie the interactive dashboard in their browser. A static image of the dashboard is shown below.

BBOXX LITHIUM-ION BATTERY FIELD DATA CHALLENGE

Team 22: Nicole Schauser, Victor Bossard, Etienne Beauchamp, Elias Galiounas, Maria Varini

TIME SERIES ANALYSIS

In this section, the raw data provided can be investigated interactively both in the form of a time series and as a statistic (box plot):

y-axis selection

voltage

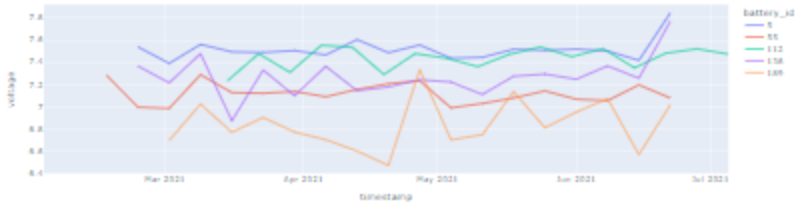
number batteries to plot

Select...

rolling aggregation

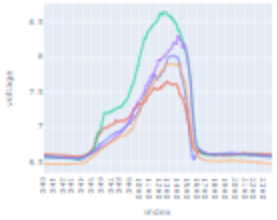
week

Raw time series



As general statistics of the data, the mean daily average and a monthly box plot of the chosen y-axis selection is displayed below.

Median voltage profile



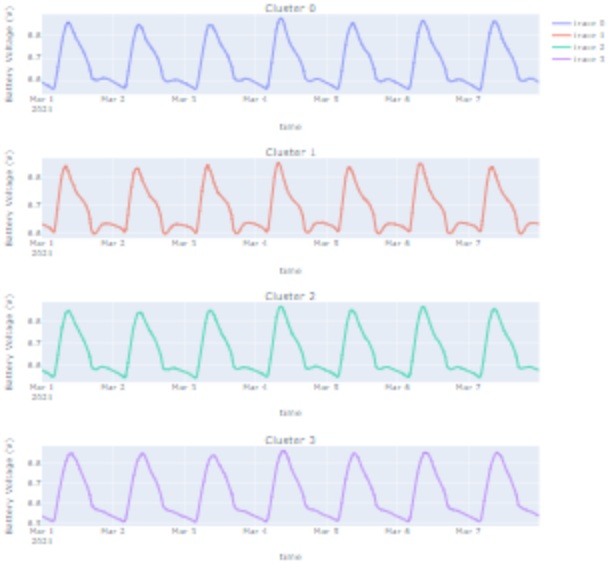
Mean voltage per battery per month

CLUSTERING ANALYSIS RESULTS

In this section, the raw data from the battery cycling are grouped with the chosen (relevant) number of clusters. Specifically, the results from the average weekly profile are presented here below for different variables.

y-axis selection

battery_voltage_mean



EXTRACTING USER PROFILES

From each of the clusters obtained, a time series of (average) voltage, current, state-of-charge and temperature is derived. These data can be used as input to a battery tester in the lab, to be able to reproduce actual user profiles from the field. These data are available for downloading:

DOWNLOAD

The dashboard is composed of three main sections; ‘Time series analysis’, ‘Clustering results’, and ‘Extracting user profiles’.

In the first section, time series data (already cleaned and re-sampled) can be visualized by choosing the variable and the time interval of interest. The possibility of a comparison between (randomly chosen) batteries is also provided. Ideally, in place of this dropdown list, another one should have been provided with ‘battery_id’, and the chance of selecting multiple ids up to the user’s choice; unfortunately, this was not successfully implemented in time.
This section also contains some statistics of the data, such as the daily median of the variable of choice, and its box plot across the entire dataset.
In this way, a rather broad overview of the raw data can be performed with the use of just a few plots.

In the second section, the results of the multivariate time-series analysis are presented as the mean of battery voltage, current, SOC and temperature for each cluster identified. With the dropdown list, it is possible to switch between the different parameters, and visualize all of them for each cluster.
Ideally, these clusters could be recalculated according to a selection of parameters (for instance, the time interval, temperature or specific battery ids of interest).These profiles represent the grouped user profiles extracted from the dataset provided.This dynamic update of the results would be very interesting as a future development.

In the third and final section, a download button is provided to obtain a .csv file of the results of the clustering displayed in the section above. Assuming a basic cell cycler, a temperature chamber, and single Li-ion cells of the same chemistry as the ones in the dataset are available , the data provided can be used to set up the equipment, and to replicate the extracted user parameters in a lab setting. For example, the current vs. time series would be fed to the battery cycler, together with the maximum and minimum voltage of the profile as safety limits, and the temperature chamber would be set on the average temperature of the profile.

Conclusions / Points of improvement

As a closing remark, this year’s BatteryDev hackathon was a fun and challenging experience. In the end, our team has successfully answered the three main goals of the challenge.

- The static approach and time series clusters provide a semi-automatic clustering method of usage groups where the number of clusters can be deduced with rule of thumbs. This is an answer to the first identified goal of the challenge
- The time series analysis and resulting clusters were used to generate new weekly profiles answering the second goal of the project.
- All of the analysis and visualizations presented answer the last goal of the challenge. Furthermore, the prepared dashboard synthesizes the results in an interactive interface.

Future improvements were also noted :

- In the static approach using aggregate statistics, the number of clusters to compute was arbitrarily defined, and further developments of this work should aim to define a metric based on which the number of clusters can be optimized.
- A score card could eventually be implemented. The idea would be to generate a number or a color on a scale to provide the user with qualitative information of how much stress the chosen usage profile would have on the battery lifetime. Assuming certain thresholds of temperature, current and SOC as adverse for the battery's lifetime (for example: temperature above 45 C Celsius, current above rated nominal capacity), different approaches (based on simple data analysis, or from the clusters' statistic analysis) could have been taken, all based on quantifying how much time the battery is spending above these thresholds.
- The dashboard could be further refined. For example, in the "Time series" section, a 'battery_id' dropdown list could be implemented to allow the user to manually choose which of the batteries they want to compare.
- The cluster analysis would ideally be updated with time. Furthermore, the score card mentioned above would ideally be inserted in the last section of the dashboard ('Extracting user profiles') with a dropdown list which would allow users to select the profile of interest (among the ones emerged from the clustering) to score.