Power Consumption Forecasting of an EPFL Building

Corentin Jaire, Kelyan Hangard, Jennifer Abou-Najm
Team StockDataHolm
CS433 Machine Learning, EPFL, Project 2

***Abstract*—In this project, two forecasting models with different time windows are developed to accurately predict the electricity consumption of an EPFL building (CM) with photovoltaic panels using given data on the building's electrical consumption and meteorological data (irradiance and temperature). The pre-processing steps taken with the data, including handling of missing data, the feature engineering process used to generate additional relevant features, the various models designed using different ML approaches and their results are all detailed. The final models chosen, artificial neural networks with hypertuned parameters, achieve 62.09% accuracy for next-day forecasting and 62.96% accuracy for intra-day forecasting.**

I. Introduction

Electricity markets rely on contracts to determine the dispatch of power. However, as the world turns to renewable energy sources like photovoltaic panels to meet the challenges of climate change and increasing electricity demand, predicting energy demand has become more complex. Accurate forecasting is crucial for ensuring the smooth operation of the electricity market, particularly in the day-ahead market where the dispatch plan for the following day must be determined in advance. Inaccurate forecasts can have serious consequences, leading to incorrect dispatch plans and potentially costly penalties.

To address this pressing need, this project aims to develop models that can accurately predict the electricity consumption of an EPFL building with photovoltaic panels. Specifically, two forecasting models for the CM building at EPFL will be developed: one that can predict the building's energy consumption for the next 6 hours (intra-day), and another one for the following day. In order to improve the performance of these models, meteorological data, such as irradiance and temperature, will be incorporated and issues with the missing data will be addressed. Feature engineering will also be performed to generate new features that can enhance the models' performance. Afterwards, multiple models designed using different ML approaches will be implemented and compared. The results of the analysis will be presented in a clear and concise manner.

II. Pre-processing

A. Data

1) Given data : In order to forecast the electrical consumption, the laboratory provided the power measurements (on a 1-minute basis) in Watts from February, 18th 2022 to November, 14th 2022 for the CM in a Matlab data file. This data represents the electricity consumption of the building, minus the photovoltaic production of the solar panels on the building's roof. The data was then converted into a .csv file, to be used in python.

2) Additional data: Nevertheless, using only the power measurement does not allow to build a strong model. To increase the efficiency, meteorological data is important. Firstly, since there are solar panels on the top of each EPFL building, the power demand of the concerned building depends highly on the solar power production, and thus on the irradiance. Indeed, the measurements given are the total consumption (which is similar between days) minus the solar production (which depends on weather condition). Secondly, the temperature is also an important parameter, as heating systems are sensitive to this parameter and very energy-consuming.

It is possible to get irradiance and temperature data with a 10-minute time resolution through the MeteoSwiss IDAWEB database, only available for students doing an academic project. This data has been obtained for the purposes of the project.

B. Missing data

The data obtained from the lab on the CM building's energy consumption is plagued by significant gaps, particularly from mid-September to mid-October and from May to June. In total, 7.87% of the daily energy consumption and 3.51% of the hourly energy consumption data is missing.

For the complete missing days, one way to address the problem is to use data from the same day of the previous or next week to replace the missing values. This strategy allows for the maintenance of the information about the day of the week, which is known to significantly impact the building's energy consumption. In addition, the overall temperature is likely to be similar from one week to the next, so using data from the same day of the week in a nearby week should provide an adequate estimate of the missing data. The energy consumption for missing hours are replaced using the same strategy, but this time using the previous or next hour of the missing hour on the same day.

After implementing this replacement method, the proportion of missing days is reduced from 7.87% to 0.00% and the proportion of missing hours from 3.36% to 1.89%. This significantly improves the quality of the data and enables the training of the models with greater confidence.

C. Interpolation of data for temperature and irradiance

The given consumption data is recorded on a minute-by-minute basis, but the irradiance and temperature data obtained is only collected every 10 minutes, meaning there are 9 missing data points between each recorded data point.

These missing data points are inserted by using a resampling method to insert the missing data points and filling them with the previous non-null value in the dataset, ensuring that the data is properly represented and ready for analysis. This is a reasonable approach, since it is unlikely that the temperature or irradiance will change significantly over the course of 10 minutes.

III. Feature Engineering

In the feature engineering phase of the project, the focus is on creating new features from the available data in order to improve the performance of the model. As only time, energy consumption and meteorological data is available, it is essential to generate additional features that could provide the model with the information it needs to make accurate forecasts.

A. Season

Two binary features are added to the dataset, indicating whether the data corresponds to winter or summer, as the season can significantly impact energy consumption in a building, particularly when it comes to heating and cooling systems, as can be seen in 1. Including these features allows for the analysis and modeling of the influence of seasonal changes on energy consumption. However, to avoid multicollinearity problems, where the values of one column can be inferred from the values of other columns in the same row, the summer feature is dropped.

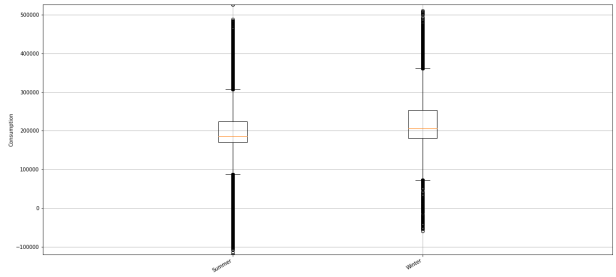

Fig. 1: Box plot of the energy consumption grouped by day of the week

B. Working days and hours

1) Weekend vs Weekday: A binary feature indicating whether a given data point occurred on the weekend is created by using the timestamp of each data point to determine whether it fell on a Saturday or Sunday, and assigning a value of 1 if it is the case and 0 otherwise.

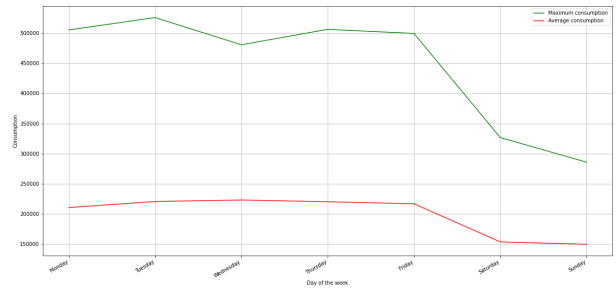

Fig. 2: Maximum and mean of the energy consumption grouped by day of the week

This feature allows the model to take into account the fact that energy consumption in the CM is lower on the weekend compared to weekdays, as can be seen in 2, since it is probably less occupied.

2) Working hours: A categorical feature day_category is created to indicate the time of day for each data point, by using the timestamp of each data point to divide the day into four categories: working hours, transition in the morning, transition in the evening, and non-working hours.

The morning transition category include data points between 3h and 6h, the working hours category between 6h and 15h, the morning transition category 15h and 19h, and the rest is considered non-working hours. This allows the model to take into account the different patterns of energy consumption that exist at different times of the day, as can be seen in fig. 3. The morning transition provide insight into the expected increase in energy use and likely includes turning on lights and other electrical equipment, as well as heating systems to warm up the space before people arrive. The working hours represent peak energy consumption as the building becomes occupied and people begin using electricity. Similarly, the evening transition represents the people leaving, hence the energy consumption diminishing.

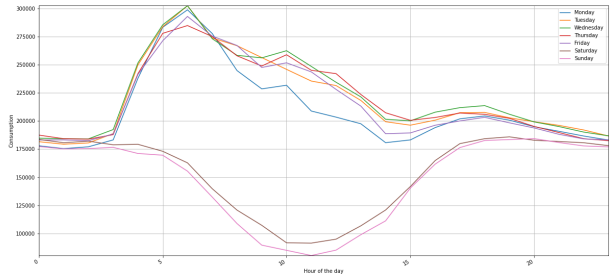

Fig. 3: Consumption profile for each day of the week

3) Holidays: A holiday binary feature is added, by using external data on holidays to identify which data points in the dataset occurred on a holiday (1 if it is the case, and 0 otherwise). It allows the model to take into account the fact that energy consumption is different on holidays compared to non-holidays, as the building is less occupied.

C. Peaks

In the dataset, small peaks in energy consumption that occur approximately once per hour for 15 minutes are observed. These peaks were likely due to heating systems that were turned on in response to the temperature in the building reaching a certain threshold. Accounting for these peaks is essential but very difficult as they do not occur periodically and depend on unknown parameters.

To account for them, they are calculated by identifying instances where the normalized value of energy consumption is between 0.5 and 0.6, that is only during non-working hours and non-transition evening hours (not capable of distinguishing them in other periods), as can be seen in red in Fig.4. The occurrence of these peaks is included as a feature in the dataset, for a deeper understanding of the patterns in energy consumption in the building, and the use of this information in analysis and modeling.

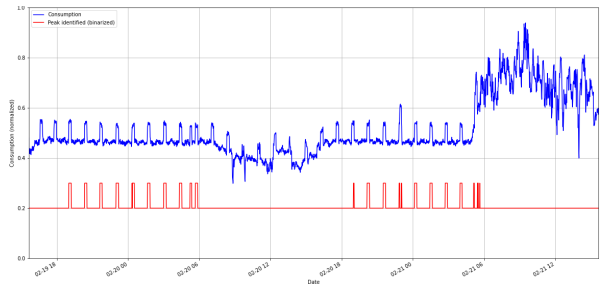

Fig. 4: Peaks identified in the consumption over 2 days (weekend day and working day)

D. Converting 1d time to 2d time

Calendar data like month and day have periodic properties, but when represented by sequential data, some of that periodicity is lost. This can be problematic in machine learning.

For instance, with a classic time display, at the end of a day, time "jump" from 23 hours to 0 hour in one step. This step of 1 hour, in real life, is the same between any two consecutive hours. However, the machine does not know that the distance between hour 23 and hour 0 is equivalent to 1 hour (and that it is a 24-hours cycle). Thus, to avoid this problem and maintain the cyclical nature of calendar data, one has to find a way to generate such a cycle.

The chosen solution is to convert time into a 2D representation by using complex numbers: $\text{Time}_{2D} = e^{\text{Time}_{1D} \times \frac{2i\pi}{60}}$. Then, instead of having one parameter with classic representation, there are two new parameters: one that is the real part of Time_{2D} and the other the imaginary part.

This principle was applied for the minutes, hours, days, months and day of week. For the days, a special attention is required depending on the length of month (28, 29, 30 or 31 days).

The following figure 5 illustrates one cycle:

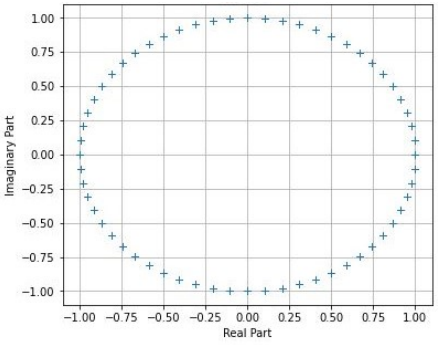

Fig. 5: 2D time representation for the minutes

E. Windowing

Three types of windows (normalized value of energy consumption at - the window time) are added to the features. The choice of windows is guided by the cyclical pattern of energy consumption:

- Window of 6 hours to catch recent power variations (only for the intra-day forecasting model)
- Window of 12 hours to capture cyclical day/night patterns (only for the intra-day forecasting model)
- Window of a day since the power consumption is very correlated from day-to-day for the same time
- Window of a week since the power consumption is very correlated from week-to-week for the same day

F. Normalization

The features are finally normalized by scaling them to have a minimum value of 0 and a maximum value of 1. This method ensures that all features are on the same scale, which helps the model learn from all the features equally. It should be noted that the mean and standard deviation were not used to normalize the features, as they are not relevant in this case.

IV. Machine learning methods

A. Comparison of multiple ML methods

Now that the preprocessing and feature engineering steps have been completed, different machine learning techniques can be applied to the data in order to build accurate forecasting models. A range of ML methods is explored and compared, including Stochastic Gradient Descent (SGD), Ridge regression, Support Vector Regression (SVR) and an Artificial Neural Network (ANN, specifically the MLP Regressor), in order to identify the best approach for predicting the energy consumption of the CM building at EPFL.

The performance of the models are assessed using evaluation metrics such as mean absolute percentage error (MAPE, adequate for forecasting a variable with a wide range of values) and the R2-score (assessing the overall fit of the model to the data and provides insight into the amount of variance in the data explained by the model) with a local validation set (splitting the data with 70% for the training set and 30% for the validation set). The parameters of the methods are hypertuned for each of the two forecasting models needed (intra-day and next-day forecasting), the difference between their features being the type of windows implemented.

Method	Parameters	Accuracy	MAPE
SGD	$\eta = 1e-5$ $n_{iter} = 5500$ $tol = 1e^{-3}$ learning-rate: adaptive	46.58%	23.5%
Ridge	$\alpha = 450$	46.73%	23.21%
SVR	$\gamma = 0.1$	56.34%	19.18%
ANN	$n_{layers} = 5$, $n_{epochs} = 4$ $n_{neurons} = 300$ learning-rate = 0.0001 $\alpha = 0.0005$ solver: *adam* activation: *relu*	62.96 %	17.75 %

TABLE I: Results obtained for each method implemented with hypertuned parameters for intra-day forecasting

Method	Parameters	Accuracy	MAPE
SGD	$\eta = 1e-4$ $n_{iter} = 5500$ $tol = 1e^{-3}$ learning-rate : adaptive	46.72%	23.55%
Ridge	$\alpha = 500$	46.33%	23.44%
SVR	$\gamma = 0.1$	54.42%	20.88%
ANN	$n_{layers} = 5$, $n_{epochs} = 6$ $n_{neurons} = 300$ learning-rate = 0.0001 $\alpha = 0.0005$ solver: *adam* activation: *relu*	62.09 %	18.09 %

TABLE II: Results obtained for each method implemented with hypertuned parameters for next-day forecasting

As can be seen in the tables I and II, for both forecasting tasking, the ANNs have a significantly higher accuracy, which is consistent with theoretical expectations, as neural networks are able to capture complex, non-linear relationships between the features. A more detailed explanation of the hypertuning of the parameters of the ANNs can be found in the section IV-B.

B. Hypertuning the parameters of the ANN

To optimize the performance of the ANNs, a thorough hyperparameter tuning process is conducted, adjusting the number of

layers, neurons, epochs, and other key parameters. Good results were achieved by carefully selecting the hyperparameters that had the most significant impact on the model's performance, and are summarized in the tables III. The particular activation *relu* is chosen for multiple reasons: it is effective at preventing overfitting in neural networks, which can be a common issue when forecasting future values based on past data and is a non-linear activation function, which means it can capture complex relationships in the data. In order to avoid overfitting, the number of epochs is adjusted as can be seen in figures 6 and 7 which shows that the optimal number of epochs is 4 for the next-day model, and 6 for the intra-day model.

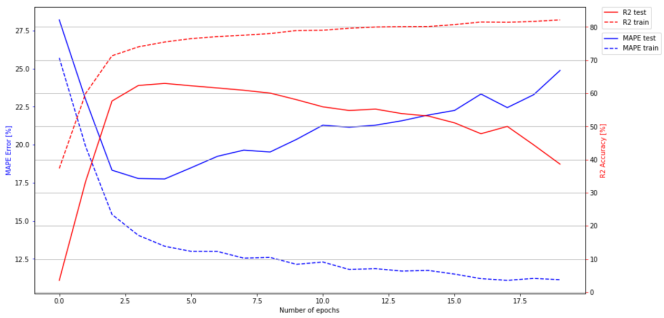

Fig. 6: Accuracies of the hypertuned next-day MLP Regressor with respect to the number of epochs

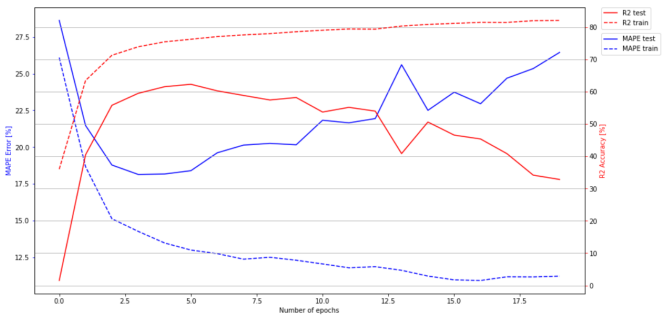

Fig. 7: Accuracies of the hypertuned intra-day MLP Regressor with respect to the number of epochs

Parameters	Intra-day	Next-day
n_{epochs}	4	6
n_{layers}	5	5
$n_{neurons}$	300	300
learning-rate	0.0001	0.0001
α	0.0005	0.0005
solver	*adam*	*adam*
activation	*relu*	*relu*
Accuracy	62.96%	62.09 %
MAPE	17.75%	18.09 %

TABLE III: Results and hyper-tuned parameters of the intra-day model and the next-day model

V. Results and Discussions

The final models chosen are the hypertuned artificial neural networks with the optimized hyper-parameters listed in the tables III. They achieve an accuracy of 62.96% and a MAPE of 17.75% for intra-day forecasting and an accuracy of 62.09% and a MAPE of 18.09% for next-day forecasting.

Some examples of predictions for the next 6 hours and 24 hours using the suitable forecasting model at midnight can be seen in the figure 8. An example of predictions on a full week (if the models are continuously fed real-time data) can also be seen in the figure 9.

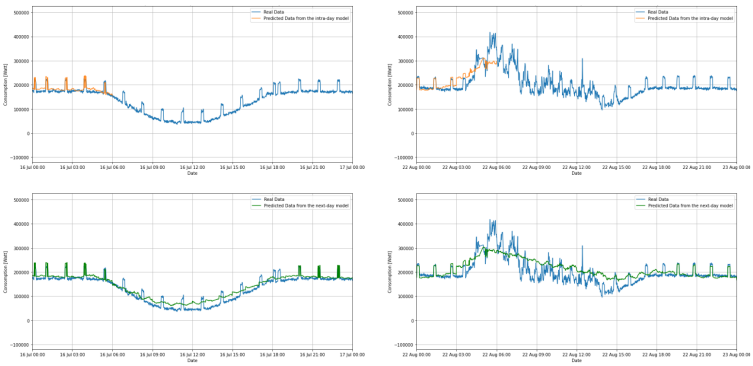

Fig. 8: One-Day Energy Consumption Model Prediction

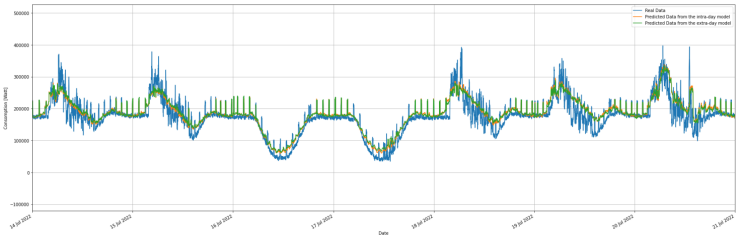

Fig. 9: Week Energy Consumption Model Prediction

There are several ways in which the performance of the forecasting models could be improved. One approach would be to incorporate additional data sources and features that may be relevant to the task, such as occupancy data or information about the building's heating and cooling systems to better model the peaks (especially during working hours). Another approach would be the use of more sophisticated ML methods, such as recurrent neural networks (specifically long-term short memory LTSM model), which could also be beneficial in improving the accuracy of the models. Finally, having one complete year of data could provide a more accurate picture of a building's energy consumption across all the different seasons.

VI. Conclusion

Motivated by the increasing use of renewable energy sources and the need for accurate forecasting tools to ensure the smooth operation of the electricity market, this project aimed to address the challenges of predicting energy demand.

To address these challenges, two forecasting models have been developed to accurately predict the power consumption of an EPFL building equipped with photovoltaic panels. The models are designed for intra-day forecasting (next 6 hours) and next-day forecasting (next 24 hours), respectively. By incorporating meteorological data, performing feature engineering, and using machine learning techniques, high levels of accuracy have been achieved in both models: 62.96% for the intra-day model and 62.09% for the next-day model. These results demonstrate the effectiveness of data-driven approaches for forecasting building energy consumption and the importance of addressing issues such as missing data and incorporating relevant features in the modeling process. These models can be adapted for use in other buildings and meet the specifications required by the laboratory.

VII. References

– A review on time series forecasting techniques for building energy consumption, Chirag Deba, Fan Zhangb, Junjing Yanga, Siew Eang Leea, Kwok Wei Shah from the National University of Singapor, 2017
– Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders, Jui-Sheng Chou, Duc-Son Tran from the National Taiwan University of Science and Technology, 2018
– Machine Learning Course CS-433, Nicolas Flammarion, Martin Jaggi, 2022