NOVA IMS Information Management School

MDSAA

Master’s degree Program in

Data Science and Advanced Analytics

Short-term forecasting for household electricity load with dynamic feature selection using power cepstrum

Machine Learning Approaches

Luis Fernando Rodrigues Agottani

Dissertation

presented as partial requirement for obtaining the master’s degree Program in Data Science and Advanced Analytics

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

20
22

Short-term forecasting for household electricity load with dynamic feature selection using power cepstrum

Luis Fernando Rodrigues Agottani

MAA

///

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

SHORT-TERM FORECASTING FOR HOUSEHOLD ELECTRICITY LOAD WITH DYNAMIC FEATURE SELECTION USING POWER CEPSTRUM

by

Luis Fernando Rodrigues Agottani

Dissertation report presented as partial requirement for obtaining the master's degree in Advanced Analytics, with a Specialization in Data Science

Supervisor / Co Supervisor: Ian Scott, PhD

Co Supervisor*:*

November 2022

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

Luis Fernando Rodrigues Agottani

November 2022

ABSTRACT

Electrical energy is present in our civilization and has positive and negative impacts in our environment, renewable and green energy like solar and wind energy works with significant less negative environmental impacts, reducing disasters and fuel dependency. Although, the transition to renewable and green energy demands advanced technologies to manage energy distribution in society, since the clean sources are stochastic. In this study the research will be done to improve electricity consumption forecasting in a household, a tool that can help the energy distribution management by microgrids to determine the amount of energy used by consumers at a particular moment, resulting in reducing energy waste and allowing P2P energy trading. The goal of this study is to do short-term forecasting and test the ability of Power Cepstrum to select autoregressive features, the dataset used is of minute-by-minute electricity consumption in kilowatts of a single household in the town of Sceaux,, France, between December 2006 and November 2010, the model tested was the Convolutional Long Short-Term Memory neural network with selected auto-regressive feature model (CLSAF), a Convolutional Long Short-Term Memory model working with Persistence model with dynamic feature selection, the ability of Power Cepstrum to select the autoregressive feature is tested and compared to CLSAF using Autocorrelation Function to select the autoregressive feature, the results are compared either to state of art models such as ConvLSTM and Persistence model. The tests were done comparing different theta threshold, input lags, resolutions, and input length. The result show that Power Cepstrum can be used as a replacement for Autocorrelation Function, CLSAF have comparable accuracy to ConvLSTM model and better runtime performance when using y[t-1] as input lag, for 30 minutes resolution is possible to observe great difference between runtime prediction without losing accuracy performance, Power Cepstrum showed better runtime prediction when compared to autocorrelation function, also, higher input length improved models performance.

KEYWORDS

Short-term energy load Forecasting, Dynamic selection, Power Cepstrum, Machine Learning, ConvLSTM

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ConvLSTM	Convolution Long Short-Term Memory
LSTM	Long Short-Term Memory
ACF	Autocorrelation Function
HEMS	House Energy Management Systems
ANN	Artificial Neural Network
SVM	Support Vector Machine
CLSAF	ConvLSTM neural network with selected auto-regressive feature
GDP	Gross Domestic Product
FFT	Fast Fourier Transform
IFFT	Inverse Fast Fourier Transform
DFT	Discrete Fourier Transform
MAE	Mean absolute error
MAPE	Mean absolute percentage error
RMSE	Root mean square error

1. INTRODUCTION

Energy is the ability to do work, and modern civilization is only possible because people have learned to convert energy from one form to another. Electrical energy is a secondary energy source produced by converting primary energy sources such as natural gas, hydraulic, wind energy, coal, nuclear energy, and solar energy into electrical energy. To date, the production process in large, centralized power plants fueled by non-renewable fossil fuels causes problems such as environmental degradation, energy losses due to the long physical distances between generation and consumption sites, and rising prices in energy bills also due to a centralized generation system.

Electricity distribution provides the needs and comfort of human behavior in the use of space and water heating, cooling, cooking, and lighting. According to (Eurostat, 2018), the household sector accounts for 27.4% of final energy consumption in Europe, and of the total energy consumed by the household sector, 24.8% comes from electricity, from the total electricity consumption, 20.3% comes from renewable energy and waste. Several studies show the urgency of the global energy transition, according to (Steg, Perlaviciute, and van der Werff 2015), the future must aim at how to fit the distribution of renewable energy into the social system and adapt it to provide energy in a smart way for human behavior efficiently.

Microgrids infrastructures and HEMS (House Energy Management Systems) are systems used to efficiently improve energy distribution between households and are a turning point in the implementation of renewable energy in residential areas, as shown by (Zhou et al. 2016).The results show increased energy savings, a change in energy consumption behavior, and can bring savings of up to 30%, as shown by (Tuomela et al. 2021).To achieve these goals, the system depends on several tools to efficiently manage the energy load. One of the tools needed is household power load forecasting, which aims to identify the behavioral pattern of the occupants in a household.

According to (Proedrou 2021) and (Kuster, Rezgui, and Mourshed 2017), the energy load forecasting studies have used different measures for energy consumption resolution of a household, energy consumption of different household electrical devices, the date range of the records, the number of houses analyzed, the exogenous features such as weather, income, and building, the statistical model used for the predictions such as ARIMA, linear regression, SVM, or ANNs, the detailed information are presented to understand the needs to solve the actual problem.

The study presented by (Li et al. 2021) shows great results using ANNs with exogenous features. The ConvLSTM model used by the author was previously implemented by (Shi et al. 2015) using spatiotemporal features and is adapted from the LSTM model, which can retain past information to identify pattern behavior. The algorithm presented by (Li et al. 2021) uses ConvLSTM neural network with selected auto-regressive feature (CLSAF), a ConvLSTM and dynamic feature selection with autocorrelation function to identify the autoregressive feature, the autocorrelation is used to identify the degree of similarity between current time series lag with past time series lags, this method improved the scores results and runtime when performing one-step ahead forecast with one hour resolution.

A new method for identifying autoregressive features is presented by (Lauwers, Vermeersch, and de Moor 2022), according to the author's results, the Power Cepstrum can identify autocorrelated features more accurately and quickly than Autocorrelation Function used in CLSAF. In this study will be tested the ability of Power Cepstrum to select the autoregressive feature for the CLSAF model and the effects for different input lengths, different input lag selections, and different time series resolutions when compared to CLSAF using Autocorrelation Function and state of art models such as ConvLSTM, and Persistence model.

2. LITERATURE REVIEW

2.1. RESIDENTIAL LOAD PROFILE MODELS

In the interest of the study, before applying models to predict the energy load in a household, it is necessary to understand what the load profile of residential buildings is. This information can be found in the study published by (Proedrou 2021), in which the author defines each parameter that makes up the residential consumption to obtain satisfactory results in the prediction of energy load. Residential is a private residence with no commercial activity, occupied in whole or in part by one or more person during the entire data period. Load is the total electricity consumed in the home by appliances during a given time. Profile is the behavioral pattern of the household occupant represented by the electricity load during that period. These definitions are important to avoid conducting a study in a home that is not representative of a common residence, as this is a known difficulty regarding volatility behavior.

According to (Proedrou 2021), research in this area has increased since 2004, the modeling approach to solve the problem can lead to several paths. The author identifies the most important features such as sampling rate, application, and statistical techniques and classifies the approaches into categories as demonstrated in Table 2.1.

Table 2.1 Model categorization summary of the reviewed models based on their key features.

Categories according to the model's key features	**Subcategories**
Sampling rate	Low resolution (15 minutes to hours)
	Middle Resolution (1 minute to 15 minutes)
	High resolution (Hz to 1 minute)
Intended application	Demand side management
	Planning, control, and design of energy systems distributions grids and local energy efficiency strategies
	Residential load profiles

According to the author, the sampling rate indicates how detailed the profile of behavioral exposure by household residents is represented in the dataset, divided into low resolution, middle resolution, and high-resolution categories. The intended application describes the use of the model to measure demand-side management, which aims to change the energy demand behavior of a household, leading to energy savings, planning, control of energy systems and distribution networks to use different energy suppliers and predict the electricity consumption of a household or group of houses to be used in projects such as person-to-person energy trading. As an example, we can see a load profile for high-resolution, one-minute energy load information that describes occupant behavior, as we see in *Figure 2.1*.

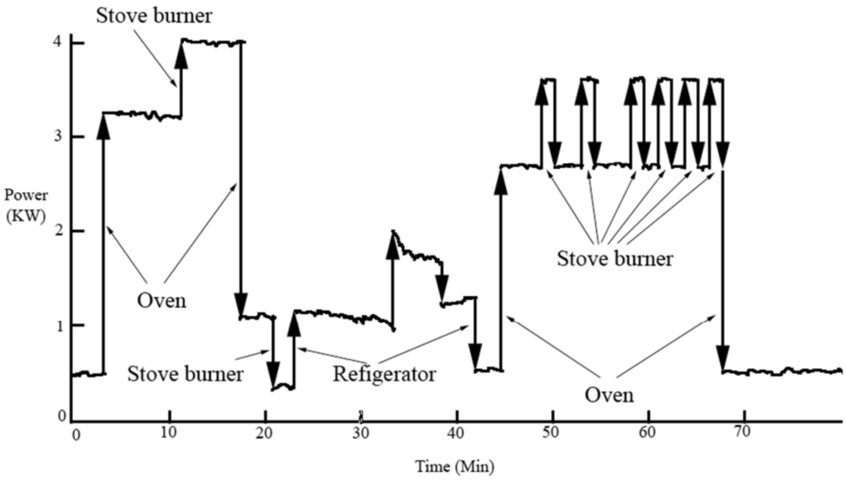


Figure 2.1 High-resolution electricity load profile presented by (Proedrou,2021).

2.2. Forecasting model approaches

The forecast model is responsible for the outcome and is a fundamental component of electricity load forecasting. As stated in the study of (Kuster, Rezgui, and Mourshed 2017), the author examined 113 different case studies with sixteen different models to determine the best methods for each electricity load forecasting problem.

First, data preprocessing is represented in 66% of the papers with four different types of preprocessing such as smoothing, filling missing values, measuring variable dependence and significance, data decomposition and classification, checking integration order and stationarity, using statistical and mathematical tools such as principal component, Pearson correlation, analysis of variance, kernel density estimation, and canonical correspondence analysis.

The prediction time frame is the time that is predicted. The model can predict the next week in different resolutions, a week in an hour or minute resolution, for the time frame there are 4 categories, very short term for less than an hour, short term for more than an hour, mid- term for a month to a season and long term for a year or more, 61.5% of the papers dealt with long term predictions and 43.6% with short term predictions, i.e. predictions for a year, a day and an hour ahead. This reflects the need for short-term forecasts for residential energy loads, or strategies for long-term forecasts in industry.

External characteristics are commonly used in studies, according to the author, and can improve results when describing human behavior. These include characteristics such as income, occupancy, electricity price, temperature, building size, precipitation, housing type, GDP, population, and others. The author has grouped these characteristics into categories such as socioeconomic, weather conditions, building type, and activity.

In the author's report, the most used models are regression models, artificial neural networks, time series ARIMA, SVM, and bottom-up models. For the evaluation, the author recommends the use of different comparison methods, such as the mean squared error, the mean absolute percentage error, or the mean percentage error. The methodology used by the author, the statistical use of each approach is provided in Appendix 1, Appendix 2, Appendix 3, Appendix 4 and Appendix 5. The summary in *Table 2.2*.

Table 2.2 Suggested approaches to energy load forecasting problems summary.

Approach	Action
Pre-processing. **Mentioned by the author:** **Pre-processing in 66% of 113 different case studies.**	Smoothing and filling missing values
	Measurement of variables dependency and significance
	Data decomposition and classification
	Check the order of integration and stationarity
	Pearson correlation, analysis of variance, kernel density estimation, and canonical correspondence analysis
Forecasting time frame. **Mentioned by the author:** **Long-term and short-term prediction represents 61,5% and 43,6% respectively the actual needs for electrical loads forecasting in buildings.**	Very short-term (Until one hour ahead)
	Short-term (One hour to one month ahead)
	Mid long-term (One month to a season ahead)
	Long-term (One year or more ahead)
Forecasting input variables	Socio-Economic (GDP, Income, Population) - 41,0% of the papers and mostly used in long-term prediction
	Weather (Temperature, Pressure, Rainfall) - 38,5% of the papers focus on short-term prediction.
	Building Type – 48,7% of the papers.
Models	Regression - 43,6% of the papers and used mostly for long-term
	Artificial neural networks - 38,5% of the papers and used in short-term and mid-term prediction

	ARIMA - 30,8% of the papers in very short and short-term prediction.
	SVM - 15,4% of the papers in very short and short-term prediction.

2.3. MODELS

2.3.1. PERSISTENCE MODEL

The persistence model is a simple method of forecast with good accuracy for small look-ahead times according to (Dutta et al. 2017), consist in predict the next step ahead to be the same as the current time step, for example, for time t with energy load y of 4 kWh, the energy load forecasting ŷ for the next step ahead t+1 will be 4 kWh as represented in Equation 1.

$$\hat{y}[t+1] = y[t]$$

Equation 1 Persistence Model.

2.3.2. ARTIFICIAL NEURAL NETWORK

The model ANN is an intelligent system inspired by humans brains, more specifically by the biological neural network, neurons, as presented by (Jain, Mao, and Mohiuddin 1996). Research on this model began in the 1940s with McCulloch and Pitts, was further developed in the 1960s by Rosenblatt with the convergence theorem of the perceptron, and the major advance occurred in 1982 with the energy approach of Hopfield with a multilayer perceptron.

A major feature is the wide range of problems in different disciplines that the model can solve: Pattern classification, clustering, function approximation, prediction, optimization, content-addressable memory, and control. The architecture of ANNs depends on neuron connection patterns and weights and is divided into two categories: Feed-forward networks and feedback recurrent networks. The main difference between the two categories lies in the learning processes. The feed-forward algorithm is memoryless, and the sequence of inputs does not affect the output; unlike feedback algorithms, where the connections between nodes form a temporal sequence that allows for temporally dynamic behavior.

2.3.3. LONG SHORT-TERM MEMORY (LSTM)

As mentioned earlier, recurrent neural networks showed improvements in solving problems, as in (Mikolov et al. 2010), where the author uses the model to tackle a speech recognition problem, although basic RNNs are not able to store long-term dependencies where the long signals are occasionally lost, as mentioned by (Hochreiter 1998).In addition to this work, long short-term memory (LSTM) algorithm was presented to the world by (Hochreiter and Schmidhuber 1997) with better memory capacity for sequential data. This was done by adding gates to the internal cell architecture, with two gate units helping to open and close access to errors in each memory cell. Over the years, researchers began to develop and improve the LSTM cell (Yu et al. 2019), and in 2015, a study by (Shi et al. 2015) presented a novel model ConvLSTM to solve a spatiotemporal sequence problem, which opened the possibility of storing more than one-dimensional information in an LSTM cell with good results.

2.3.4. CONVOLUTIONAL LONG SHORT-TERM MEMORY (CONVLSTM)

The ConvLSTM architecture works with the convolution operator used in the state-to-state and input-to-state transitions before entering the LSTM cell, as we can see in (Shi et al. 2015), the operator in Figure 2.2, the architecture in Figure 2.3 and the calculation in *Equation 2* of a ConLSTM model as follows:

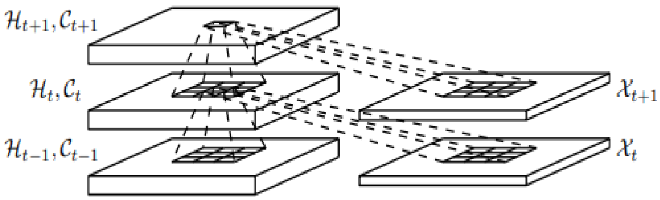


Figure 2.2 Convolution operator in the ConvLSTM model.

Source: (Shi et al. 2015

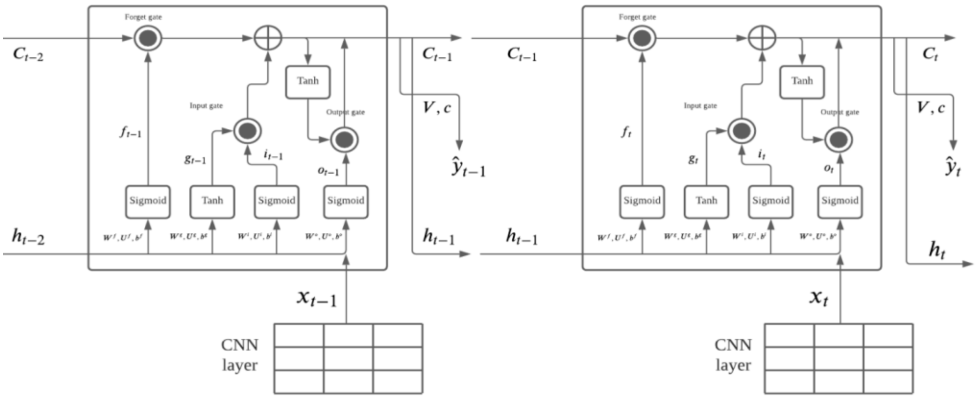


Figure 2.3 Architecture of a ConvLSTM model.

Source: (Fang et al. 2021)

$$g_t = tanh(W^g * h_{t-1} + U^g * x_t + b^g)$$

$$Forget\ gate: f_t = \sigma(U^f * x_t + W^f * h_{t-1} + b^f)$$

$$Input\ gate: i_t = \sigma(U^i * x_t + W^i * h_{t-1} + b^i)$$

$$Long - term\ state: c_t = f_t \odot c_{t-1} + g_t \odot i_t$$

$$Output\ gate: o_t = \sigma(U^o * x_t + W^o * h_{t-1} + b^o)$$

$$Short - term\ state: h_t = o_t \odot tanh(c_t)$$

$$Output: \hat{y}_t = V * h_t + c$$

Equation 2 ConvLSTM calculation.

Source: (Fang et al. 2021)

where W and U are parameters for the information from the previous hidden state and current hidden state, b is the bias term, and V is the parameter of output. The Xt is the input vector for the time series value in LSTM; however, the Xt will take the convolutional layer as input in the ConvLSTM2D model.

2.4. CONVLSTM IN ENERGY LOAD FORECASTING

The use of ConvLSTM showed good performance in the spatiotemporal domain and as we can see in the study (Li et al. 2021), the architecture presented good results in the energy load for one step ahead forecasting problems and one hour resolution. Compared to the benchmark models ARIMA and Persistence, the model improved the result by 9.6% and 5.6%, respectively.

The author calculated another model for the benchmark, the CLSAF with the best score of the models according to the author, compared to ConvLSTM an improvement of 8%, the CLSAF works with the combination of the dynamic feature selection by using the autocorrelation function with the ConvLSTM, the reason for implementing this process according to the author is to avoid overfitting and cover gaps of human behavior occupancy, it is possible to better understand the architecture with the flowchart from Figure 2.4:

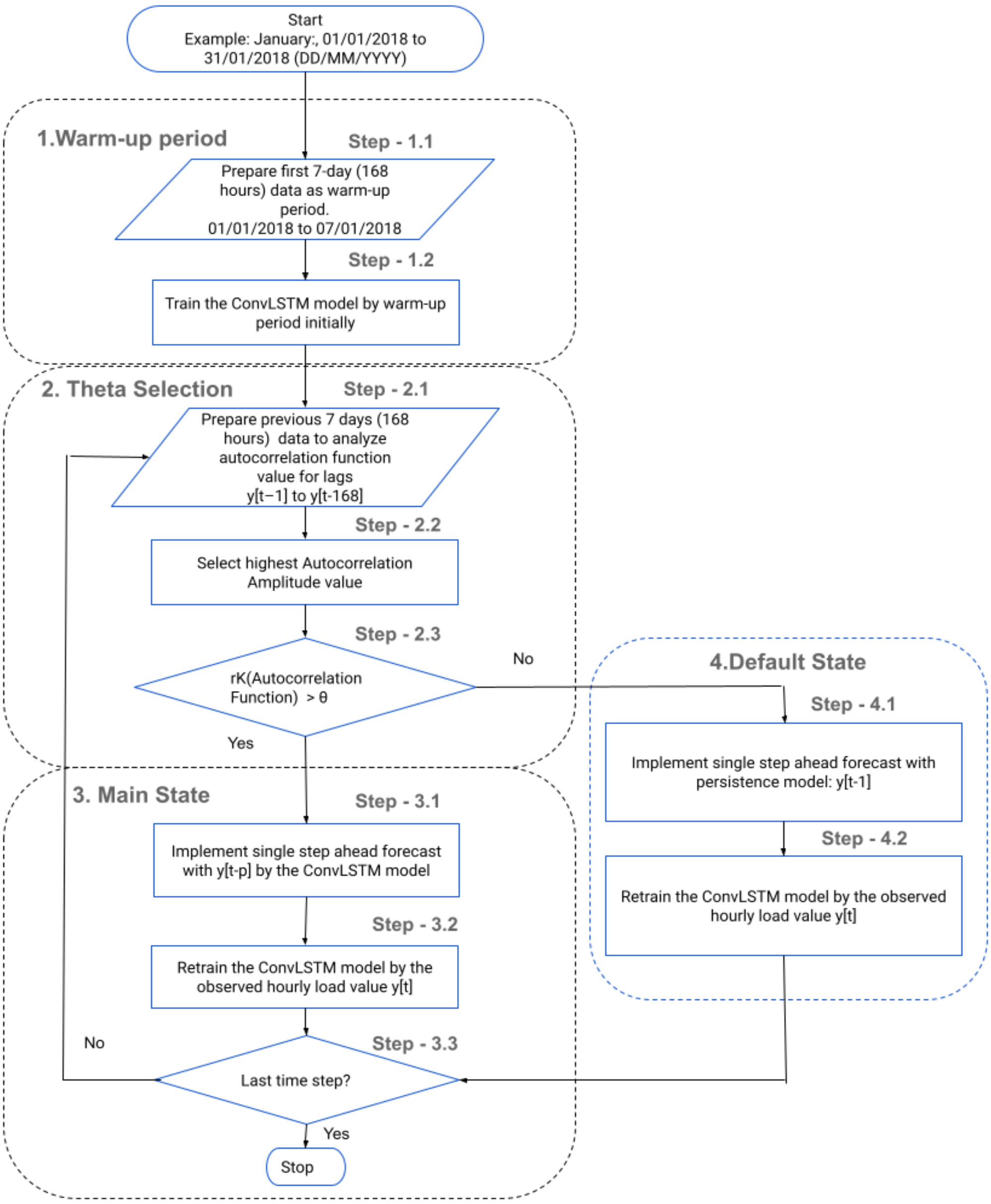


Figure 2.4 The CLSAF model consists of a standard ConvLSTM model with two additions: the autocorrelation-based algorithm, which dynamically selects the most-suited prior load, and the "default" state which ensures robust forecast during periods of overfitting.

Source: Adapted from (Li et al. 2021)

The algorithm is divided into start, warm-up (1), theta selection (2), main state (3), and default state (4). The dataset includes hourly energy load information and additional features such as temperature, absolute humidity, wind speed, binary weekday/weekend, and sin(local time[t]) that correlate with human behavior.

2.5. Theta Selection

The selection of the theta is the key point for this model. The author uses the autocorrelation function, which is a valuable tool to identify the patterns behavior of energy loads between current and past loads, and is defined in *Equation 3*:

$$rk = \frac{\sum_{t=k+1}^{n} \quad (X_t - \underline{X})(X_{t-k} - \underline{X})}{\sum_{t=1}^{n} \quad (X_t - \underline{X})^2}$$

Equation 3 Autocorrelation Function.

Source: (Li et al. 2021)

where X is the electricity load, t is the current time step position in the time series, $\underline{X}$ is the mean value and k is the past time step position in the time series.

Based on the study of (Sood and Koprinska 2010), the author compares the amplitude of the autocorrelated lag and the theta threshold. If the highest amplitude is higher than the theta threshold, the ConvLSTM model uses the selected lag to predict the next step ahead, otherwise the persistence model is used. For example, in Figure 2.5, the values from applying the autocorrelation function for 24 previous lags to determine the highest autocorrelation amplitude are compared with the theta threshold that are shown in dashed red.

In example (a), the second lag after the first, that is autocorrelated, is higher than the theta threshold, so the ConvLSTM model would be used to forecast the next time step ahead for this time step, unlike in example (b) where the highest autocorrelation amplitude is lower than the theta threshold and the persistence model would be used forecasting the next time step ahead for this time step.

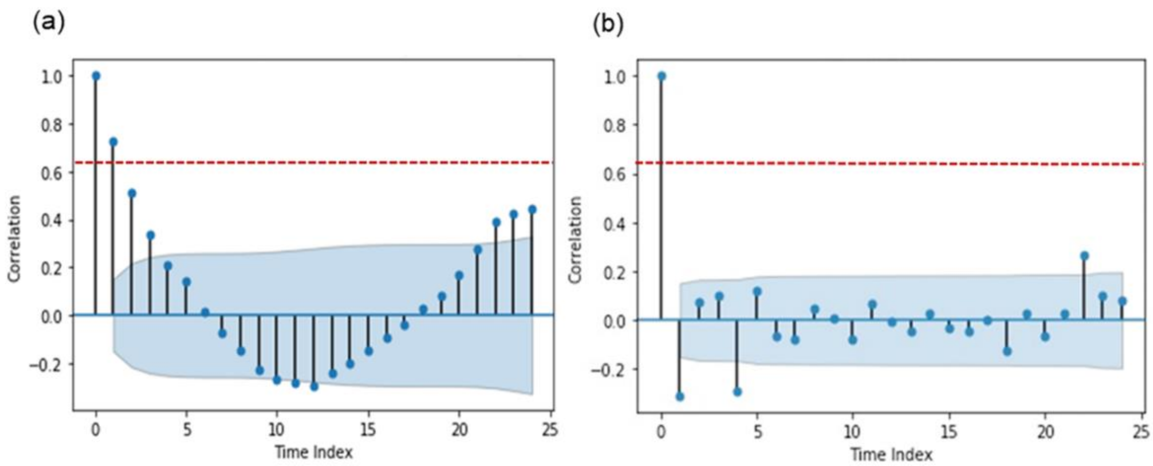


Figure 2.5 Autocorrelation Function applied to two different time steps as example, using 24 hours as previous steps.

Source: (Li et al. 2021)

2.6. Evaluation of model Accuracy

According to (Hyndman 2006), there are four types of forecast scores. The mean absolute error, known as MAE, is the easiest to calculate and understand for a single time series, the only problem is that you cannot compare it to other time series. The mean absolute percent error, MAPE, does not need to be scaled and fixes the problem of MAE, it is possible to compare results in multiple time series, but for values equal to zero in the time series, the results tend to be infinite. Relative error metrics are independent, but when divided by zero, we have the same problem as MAPE.

The evaluation metric used by the author (Li et al. 2021) is the normalized root mean square error, arguing that it is better to calculate errors at low values, presented in Equation 4.

$$CV - residual(\%) = \frac{\sqrt{\frac{1}{N-1}\sum_{t=1}^{N} \quad (y_t - \hat{y}_t)^2}}{\bar{y}}$$

Equation 4 CV - Residual (%).

Source:(Li et al. 2021)

Where N is the time series length, y_t is the observed time step ahead, $\hat{y}_t$ is the time step ahead forecasted and $\bar{y}$ is the time series energy load mean.

2.7. Results Presented by the Author

The results published by the author show an improvement in accuracy, minimum error, and maximum error when comparing the CLSAF model to the other 4 benchmark models listed in Table 2.3.

Table 2.3 Results from models evaluated. *Source:* (Li et al. 2021)

Model Name	Mean Value	Minimum	Maximum
Persistence	61.2	6.3	141.4
SW-ARIMA	64.1	6.4	201.5
SW-ETS	63.7	6.4	188.4
SW-SVR	62.0	6.3	162.3
ConvLSTM	57.9	6.2	131.1
CLSAF	53.3	5.9	115.8

The author used the theta threshold with the lowest average CV-Residual (%) of 20 sampled apartments in 3 seasons. The results for the theta threshold with the lowest CV-Residual(%) in Figure 2.6.

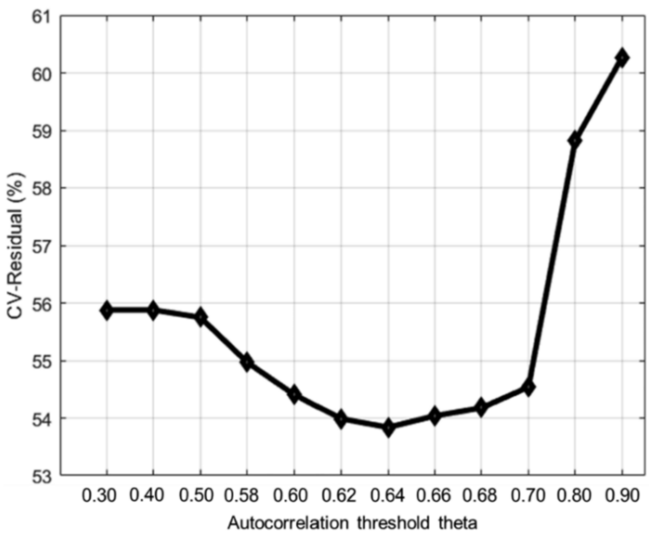


Figure 2.6 Results for theta threshold.

Source: (Li et al.,201).

The author observed in results that the higher the diurnal pattern higher the accuracy of the prediction model, to measure the strength of the diurnal pattern the author used DFT, formulated in *Equation 5*:

$$Yk = \sum_{n=0}^{N-1} y_n e^{\frac{-2\pi kni}{N}}$$

Equation 5 Discrete Fourier Transform.

Source:(Li et al. 2021)

Where N is the time series length and k the frequency.

The author observed that another factor in the pattern behavior could be impacting the model accuracy, the volatility of the electricity consumption, the CV Observation that is calculated by *Equation 6*:

$$CV - Observation(\%) = \frac{\sqrt{\frac{1}{N-1}\sum_{i=1}^{N} (y_i - \bar{y}_i)^2}}{\bar{y}}$$

Equation 6 CV-Observation.

Source:(Li et al. 2021)

Where N is the time series length, y_t is the observed time step ahead, $\bar{y}$ is the time series energy load mean.

The results observed by the author comparing the CV-Residual(%) with the volatility of the electricity consumption and the diurnal pattern behavior in Figure 2.7.

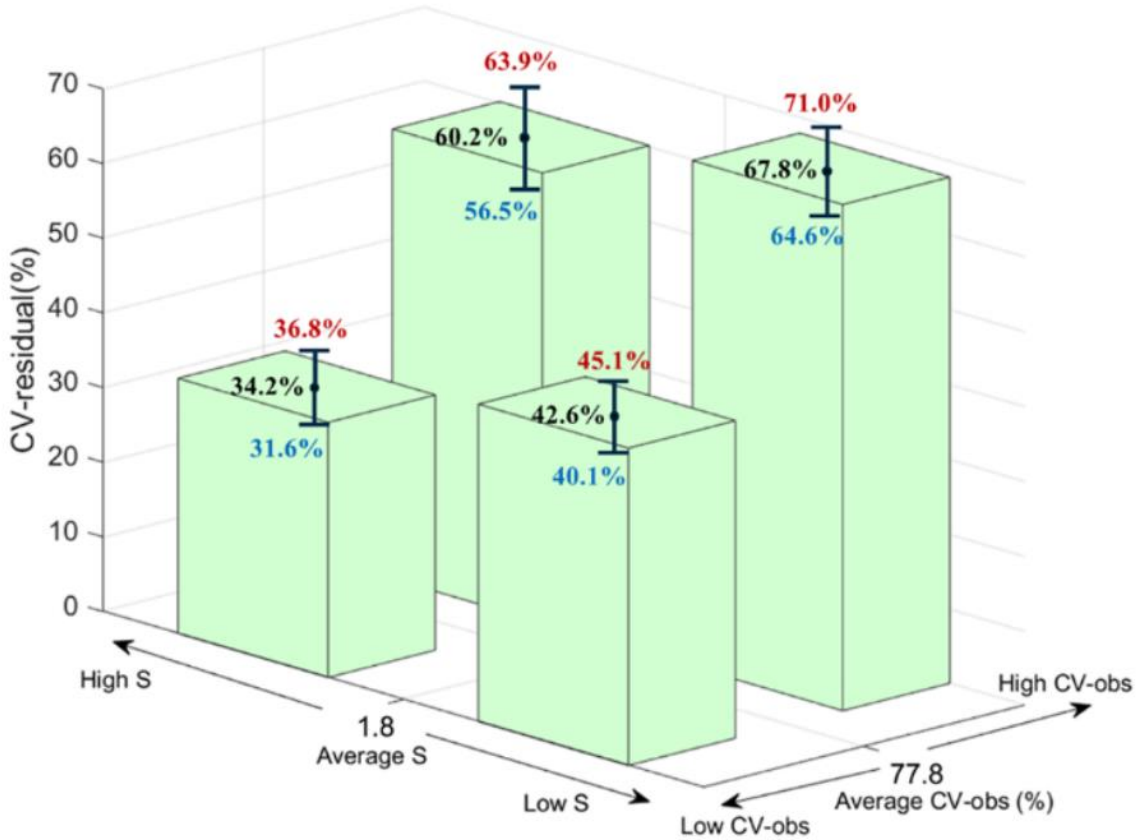


Figure 2.7 The results from the author indicating that lower CV-Observation and strength of diurnal pattern (S) resulted in better accuracy or low CV-residual(%).

Source: (Li et al.,2021).

2.8. CEPSTRUM

The Power Cepstrum is the result of calculating the inverse Fourier transform (IFT) of the logarithm of the estimated signal spectrum, a tool for searching for periodic structures in frequency spectra. The Power Cepstrum is a tool used in 1963 by B. P. Bogert and M. J. Healy to characterize seismic echoes from earthquakes and bomb blasts and is now used to solve various problems such as human speech, radar signals, and autoregressive system identification (Lauwers, Vermeersch, and de Moor 2022), with improvements to a faster system on a larger scale.

Other positive results using power cepstrum were observed in (Kalpakis, Gada, and Puttagunta 2001). The Cepstral coefficients for feature extraction improved cluster accuracy with no computational differences between small and long time series. The Power Cepstrum, Cy(k), is the result from *Equation 7*.

$$Cy(k) = |IFFT\{log(|FFT\{f(t)\}|^2)\}|^2, \text{ or:}$$

$$\Phi y(e^{tw}) = |FFT(y(t))|^2$$

$$\Phi y'(e^{tw}) = log(\Phi y(e^{tw}))$$

$$Cy(k) = |IFFT(\Phi y'(e^{tw}))|^2$$

Equation 7 Power Cepstrum.

Source:(Li et al. 2021)

Where FFT is the Fast Fourier Transform and IFFT is the Inverse Fast Fourier Transform. The calculation of the FFT and IFFT was done in python with the numpy.fft package.

2.9. SUMMARY

The objective of this study is to implement the Power Cepstrum analysis presented by (Lauwers, Vermeersch, and de Moor 2022) to replace the Autocorrelation Function in step 2.1, as shown in the flowchart of the CLSAF model in Figure 2.4, and compare the results for CV -residual(%), runtime, different lag sizes, and input sizes to understand the effect of each parameter on the CV -residual(%) results.

The dataset was selected using the (Proedrou 2021) list of recommendations for different types of datasets and profiles. According to the results presented by (Kuster, Rezgui, and Mourshed 2017), the best dataset to test the ANN model would be a short-term time frame for forecasting, one hour ahead, with a low resolution, maximum the one-hour time step, with global energy load activity in time unit representing the behavior of household residents, with available time index with day, month, year and time and location to obtain weather information for exogenous characteristics.

3. METHODOLOGY

3.1. INTRODUCTION TO METHODOLOGY

The methodology contains all the information about the data used in this study and the approach used to evaluate the different parameters of the model in different steps.

3.2. DATA

The data used in this study consist of minute-by-minute electricity consumption in kilowatts of a single household in the town of Sceaux (7 km from Paris, France) between December 2006 and November 2010 (47 months) from the source Georges Hebrail (georges.hebrail '@' edf.fr), Senior Researcher, EDF R&D, Clamart, France, and weather data for the city of Sceaux from the source WorldWeatherOnline (https://pypi.org/project/wwo-hist/) between August 2008 and August 2010, which include the forecast temperature (°C) for the next hour, the next hourly step, and a Boolean column indicating the weekend days.

For the analysis, the dataset was resampled in hourly steps aggregating the mean for Global Active Power, in the matter of the study, to get all seasons, it (?) will be used data for one year, 2009, since it is the only complete year with data. The first and last record from the dataset in Table 3.1 and energy load time series in Figure 3.1.

Table 3.1 The first and last line of the dataset used to train and evaluate algorithms.

Datetime	Mean Global_Active_Power (Kw/minute)	Temperature for the next hour (°C)	Weekend	Next Hour
2009-01-01 00:00:00	0.534933	-2	0	1
…	…	…	…	…
2009-12-31 23:00:00	1.690500	1	0	0

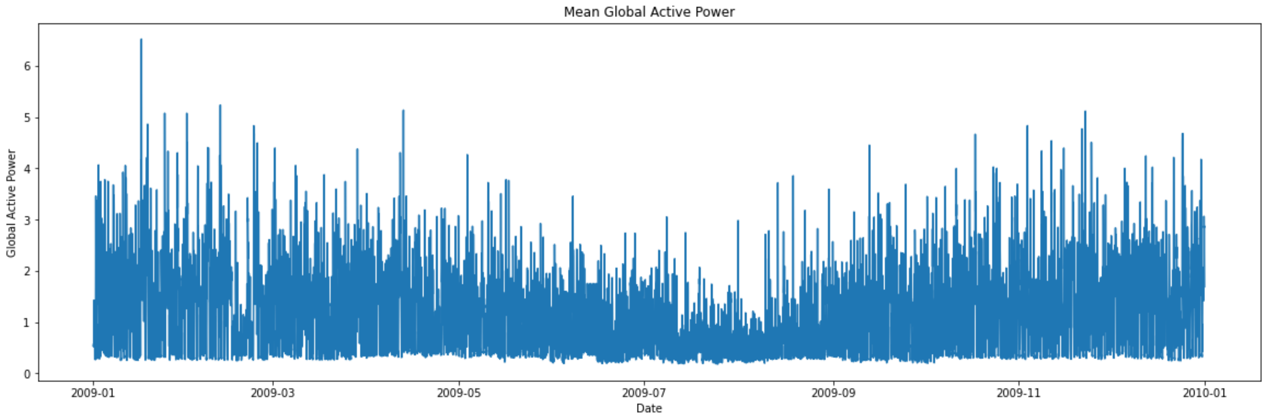


Figure 3.1 The hourly average kW/minute electricity demand load for the city of Sceaux from January 2009 to December 2009.

3.3. MODEL

The goal is to use a short-term forecasting model to predict the one-hour step ahead of the Global Active Power for each month of the year. The model proposed to solve this problem in this study is a ConvLSTM with a dynamically selected auto-regressive feature and a default state with a persistence model, the CLSAF model created by (Li et al. 2021).

The differences in this proposed model are that the auto-regressive lag is determined by the Power Cepstrum, as used by (Lauwers, Vermeersch, and de Moor 2022) and (Kalpakis, Gada, and Puttagunta 2001) to address the problem with less runtime and better accuracy. In this study, further variations are performed compared to the tests made by (Li et al. 2021), the boxes in red will be evaluated with different configurations as shown in Figure 3.2. The model was developed in python, the code is available in github.com/luisfernandoagottani/ConvLSTM-with-autoregressive-feature-selection-for-

Short-term-forecasting, the model run was made in a computer with CPU 6 Core 2.70 GHz Intel Core i7-10850H (Hyper-Threaded), memory ram of 32370 MB, Dell Inc. model Latitude 5411.

In the following sections the model procedure is explained.

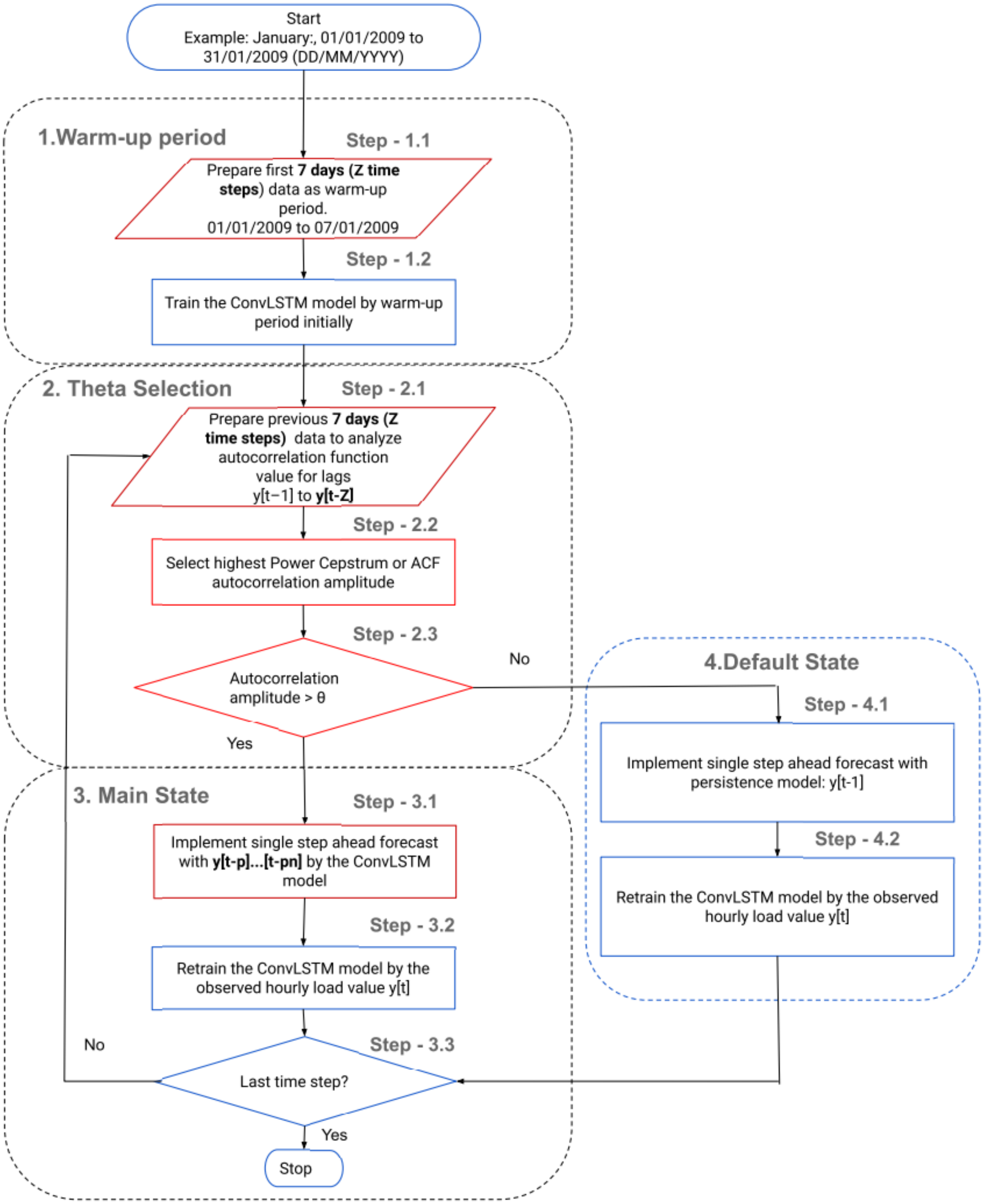


Figure 3.2 Flowchart of CLSAF Adapted with Power Cepstrum used in the present study.

3.3.1. START

The model is tested separately for each month of the year, following flowchart from Figure 3.2 a date is used as an example, between 01/01/2009 to 31/01/2009, which is divided into four weeks, the first week is used to warm up and the last three weeks are used to test and retraining in each hourly step, the dataset used has a resolution of one hour time step, which in this example gives a total of 744 records, which corresponds to 744 hours in 31 days. The external features are weather forecast results.

3.3.2. WARM-UP

The warm-up period was used to train the ConvLSTM model for the first seven days of the month before starting to predict the next step.

3.3.2.1. WARM-UP - STEP 1.1 – SELECT THE FIRST 7 DAYS

Select the first seven days of the month to train the Conv-LSTM and warm up the model. In this case, records from 01/01/2009 to 07/01/2009 are selected, giving a total of 168 records if the resolution is one hour and 4 features with a shape of (168,4), as represented in Figure 3.3.

Figure 3.3 Represents the data split to the warm-up period to train the ConvLSTM and forecasting with the model retrain period.

3.3.2.2. WARM-UP - STEP 1.2 – TRAIN THE CONVLSTM MODEL

After selecting the first 7 days for the warm-up period, was trained the ConvLSTM model as demonstrated in *Figure 3.4*:

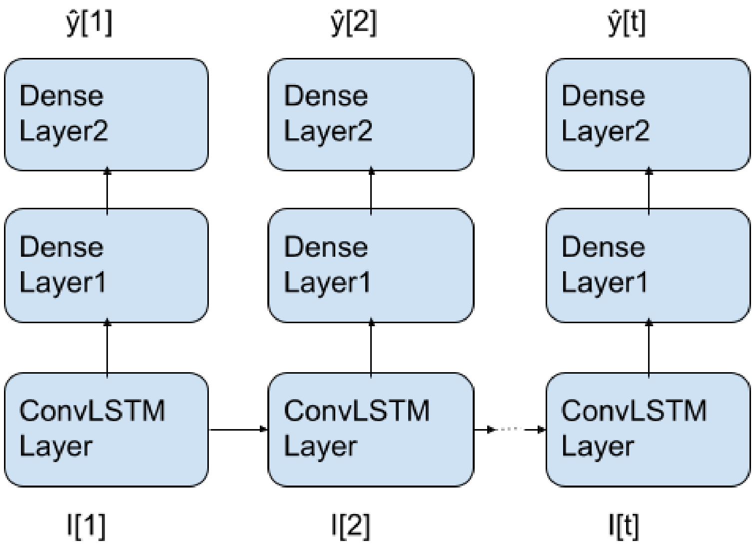


Figure 3.4 Sequential architecture of the ConvLSTM model for the warm-up period.

Source: (Li et al. 2021)

In this study, is used the ConvLSTM2D from Keras, which maps the convolutional neural network with the LSTM model used for image classification with matrix multiplication before the cell enters the LSTM gate, is used as in (Shi et al. 2015) to better capture the spatiotemporal correlation of their data since we use a dataset with multidimensional features. The ConsLSTM configuration can be found in **Appendix 6.** The input shape depends on the benchmark models being evaluated as demonstrated in Table 3.4. The output shape will always be one step ahead.

3.3.3. THETA SELECTION

The theta selection will be evaluated with Power Cepstrum or Autocorrelation Function to perform the autoregressive feature selection.

3.3.3.1. THETA SELECTION - STEP 2.1 - PREPARE THE PREVIOUS 7 DAYS

Prepare the previous 7 days, for one hour resolution would be y[t] to y[t-168], to apply the autocorrelation function or Power Cepstrum.

The Power Cepstrum amplitude is calculated using Equation 7 and applied to the previous 7 days for autocorrelation analysis, an example with the time 2009-08-20 at 00:00:00, the Power Cepstrum values of previous 168 records in Figure 3.5:

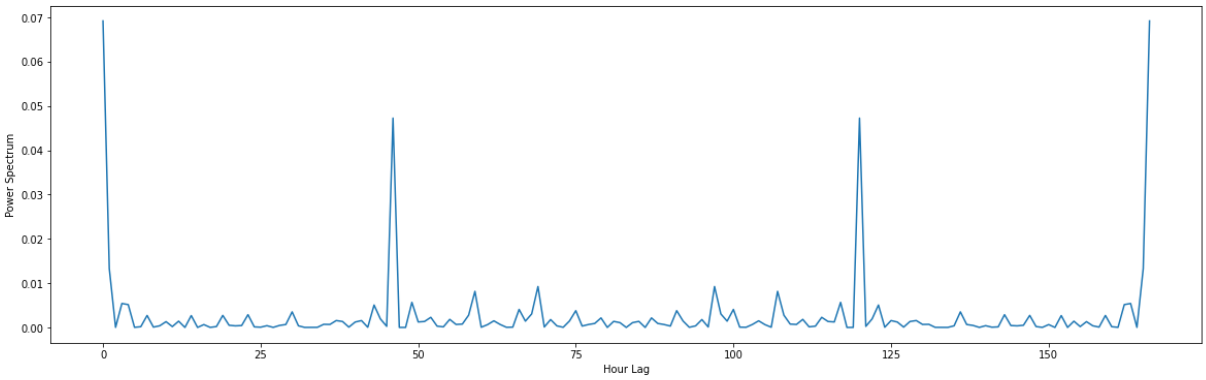


Figure 3.5 The Power Cepstrum values calculated for the previous 168 hours on the current time of 2009-08-20 at 00:00:00.

The same process was performed to evaluate the autocorrelation amplitude with the autocorrelation function (ACF) from Equation 3, applied to the 168 hours (7 days) for the autocorrelation analysis, an example with the time 2009-08-20 at 00:00:00 in Figure 3.6:

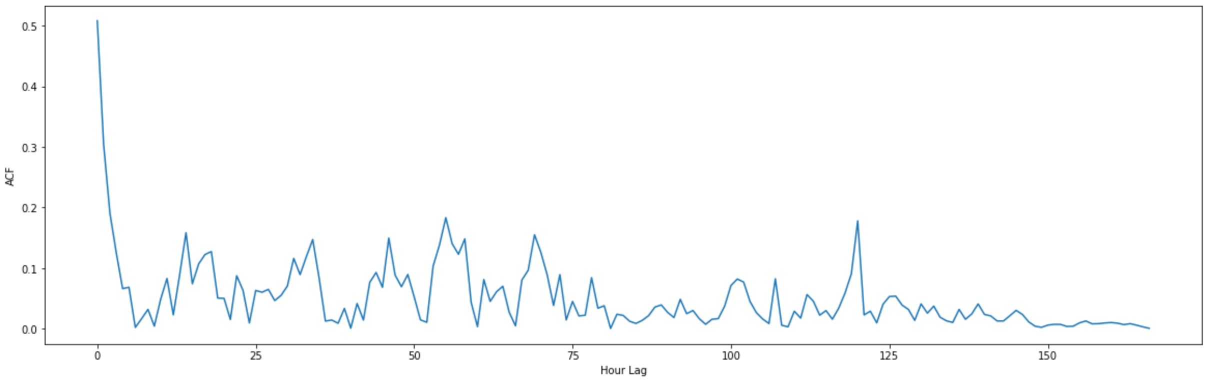


Figure 3.6 The autocorrelation function amplitude values calculated for the previous 168 hours on the current time of 2009-08-20 at 00:00:00.

3.3.3.2. THETA SELECTION - STEP 2.2 – SELECT THE HIGHEST AMPLITUDE

Applying the autocorrelation function or Power Cepstrum and selecting the highest Autocorrelation Amplitude of y[t-168] to y[t-1], the first lag is not analyzed because it is correlated with itself.

The result of the **highest** Power Cepstrum amplitude for each time step in the time series using the previous 168 hours (7 days) of each time step between January 2009 and December 2009 in Figure 3.7:

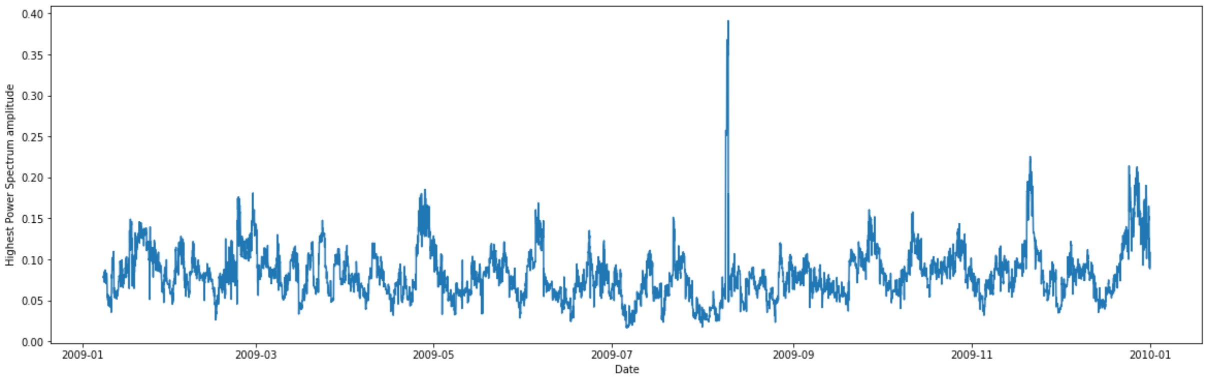


Figure 3.7 The highest Power Cepstrum value considering the previous 168 hours for each hour from January 2009 to December 2009.

The result of the **highest** ACF amplitude for each time step in the time series using the previous 168 hours (7 days) of each time step between January 2009 and December 2009 in Figure 3.8:

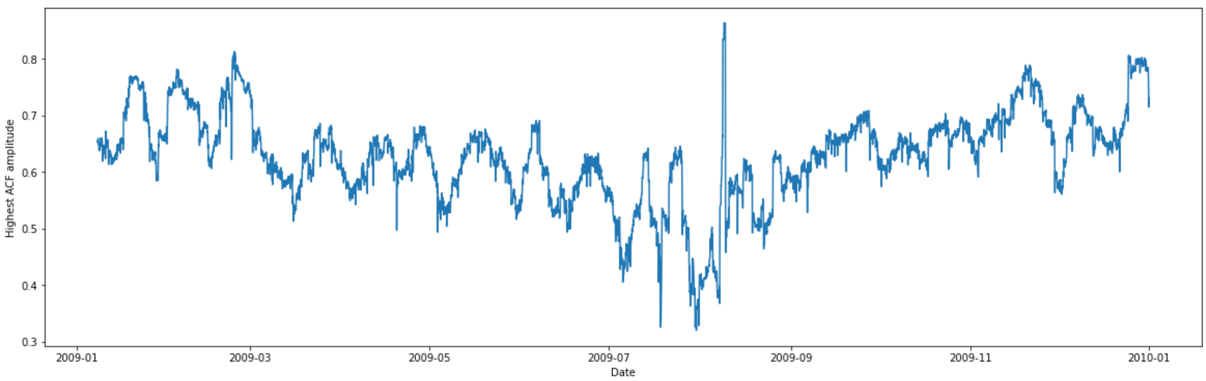


Figure 3.8 The highest ACF value considering the previous 168 hours for each hour from January 2009 to December 2009.

3.3.3.3. Theta Selection - step 2.3 – Verify if rK value is higher than the theta threshold

After determining the highest autocorrelated amplitude lag was compared the value to the theta threshold manually set for the entire period. Higher theta values increase the frequency of using ConvLSTM model to forecast, to compare the Power Cepstrum and Autocorrelation Function the theta threshold will be defined according to the % of theta using the Main State model. As represented in Table 3.2.

Table 3.2 Simulate % of lags used according to theta threshold.

Theta Threshold	**% of time steps using Main State to forecast**	**% of time steps using Default State to forecast**
0.6	85%	15%
0.7	60%	40%
…	…	
0.95	15%	85%

3.3.4. MAIN STATE

The main state runs the ConvLSTM if the Boolean from step 2.3 is True.

3.3.4.1. MAIN STATE - STEP 3.1 – IMPLEMENT SINGLE-STEP AHEAD FORECASTING WITH CONVLSTM

The ConvLSTM runs the single-step forecast, the parameters used in the ConvLSTM model are in **Appendix 6**. The autoregressive input is determined according to the configuration being tested described in Benchmark models Table 3.4, in this study will be tested as autoregressive input the lag with highest autocorrelation amplitude, y[t-pn], same as presented by (Li et al. 2021) and will be test with always using the last step known, y[t-1]. For higher lengths will be evaluated with the last time steps known, for example, for 6 lengths, y[t-1] to y[y-6].

3.3.4.2. MAIN STATE - STEP 3.2– RETRAIN THE CONVLSTM

After forecasting the single step ahead, the ConvLSTM model is retrained with the observed hourly load, y[t].

3.3.5. DEFAULT STATE

The author created the default state to avoid overfitting, a period without occupancy, or when human behavior is different from usual.

3.3.5.1. DEFAULT STATE - STEP 4.1– IMPLEMENT SINGLE-STEP AHEAD FORECASTING WITH THE PERSISTENCE MODEL

When the highest autocorrelation amplitude is lower than the theta threshold the Boolean result is false in step 2.3 and the default state forecasts the next step ahead with the persistence model, equal to the last known time step, calculated by Equation 1.

3.3.5.2. DEFAULT STATE - STEP 4.2– RETRAIN THE CONVLSTM

After forecasting the single step ahead with the Persistence model, was retrained the ConvLSTM model with the observed hourly load, y[t].

3.4. METRIC TO EVALUATE FORECASTING ACCURACY

The metric used to evaluate the forecasting accuracy is the normalized RMSE from Equation 4 used by (Li et al. 2021).

3.5. BENCHMARK MODELS AND TEST PROCEDURE

According to (Li et al., 2021), time series with low volatility and high pattern behavior in electricity consumption led to better model accuracy. The first analysis is conducted to check the predictability of each month for 2009, after, the analysis of the theta thresholds and the differences in autoregressive selection when using Power Cepstrum or ACF are done, and the last step, different values of theta threshold are set to test the benchmark models, the tests are conducted by running the models 10 times for each benchmark model configuration presented in Table 3.4, each configuration are explained as follows:

3.5.1. INPUT AUTOREGRESSIVE FEATURES

The input autoregressive features are related to the position of the time step that is used in the input to forecast the next time when using the ConvLSTM model. As we can see in the flowchart in Figure 2.4, step 3.1 , the author (Li et al.,2021) uses as input in the ConvLSTM the time step with highest Autocorrelation amplitude lag, y[t-pn], measured in step 2.1 by ACF, in this study, other methods are tested for comparison, using also the previous step, y[t-1], every time the highest correlation amplitude is higher than theta threshold set manually, as represented in .

Table 3.3 Autoregressive features time step position configuration example

Input autoregressive features	Current Time position (hour)	Prediction Input position (hour)	Output position (hour)
y[t-1]	12	12	13 (one step ahead)
y[t-pn]	12	6 (highest autocorrelation amplitude lag)	13 (one step ahead)

.

3.5.2. AUTOCORRELATION MEASUREMENT

Autocorrelation is performed in two ways, the Autocorrelation Function (ACF) or the Power Cepstrum. This tool is used in step 2.1 of Figure 3.2.

3.5.3. INPUT SHAPE (STEP, LENGTH, FEATURES)

The input shape is related to the number of time steps, length, and features used as input to train and retrain the ConvLSTM model, steps 1.2, 3.1, 3.2, and 4.2 from Figure 3.2, are used in this study with one step, different input length for tests, and four features, as shown in Table 3.1. For higher input length the epochs for ConvLSTM Structure are higher, as described in **Appendix 6.**

3.5.4. PREVIOUS LAGS FOR AUTOREGRESSIVE ANALYSIS

The number of previous lags used to analyze the autocorrelation amplitude with the current time step by Power Cepstrum or ACF as in step 2.1 varies depending on the resolution of the time step. In this study, the previous seven days of data are used for each lag autocorrelation analysis.

3.5.5. BENCHMARK MODEL CONFIGURATION

The configuration for each model that will be evaluated is represented in Table 3.4.

Table 3.4 Benchmark Models and configurations evaluated.

Analysis Description	Model Name	Resolution	Autoregressive features	Input Length	AR Calculation	Previous days for autoregressive analysis
Autoregressive input analysis	Persistence	1 hour	y[t-1]	1	None	None
Autoregressive input analysis	ConvLSTM	1 hour	y[t-1]	1	None	7 days (168 hours)
Autoregressive input analysis	CLSAF-ACF y[t-pn]	1 hour	y[t-pn]	1	ACF	7 days (168 hours)
Autoregressive input analysis	CLSAF-ACF y[t-1]	1 hour	y[t-1]	1	ACF	7 days (168 hours)
Autoregressive input analysis	CLSAF-Power Cepstrum y[t-pn]	1 hour	y[t-pn]	1	Power Cepstrum	7 days (168 hours)
Autoregressive input analysis	CLSAF-Power Cepstrum y[t-1]	1 hour	y[t-1]	1	Power Cepstrum	7 days (168 hours)
Resolution Analysis	ConvLSTM	1 minute	y[t-1]	1	None	7 days (10080 minutes)
Resolution Analysis	CLSAF-ACF y[t-1]	1 minute	y[t-1]	1	ACF	7 days (10080 minutes)

Resolution Analysis	CLSAF-Power Cepstrum y[t-1]	1 minute	y[t-1]	1	Power Cepstrum	7 days (10080 minutes)
Resolution Analysis	ConvLSTM	30 minutes	y[t-1]	1	None	7 days (336 lags)
Resolution Analysis	CLSAF-ACF y[t-1]	30 minutes	y[t-1]	1	ACF	7 days (336 lags)
Resolution Analysis	CLSAF-Power Cepstrum y[t-1]	30 minutes	y[t-1]	1	Power Cepstrum	7 days (336 lags)
Input Length analysis	ConvLSTM	1 hour	y[t-1 to t-input length]	1, 2, 3, 4, 5, 6, 12, and 24	None	7 days (168 hours)
Input Length analysis	CLSAF-ACF y[t-1 to t-length]	1 hour	y[t-1 to t-length]	1, 2, 3, 4, 5, 6, 12, and 24	ACF	7 days (168 hours)
Input Length analysis	CLSAF-Power Cepstrum y[t-1 to t-length]	1 hour	y[t-1 to t-length]	1, 2, 3, 4, 5, 6, 12, and 24	Power Cepstrum	7 days (168 hours)

3.6. MODEL EVALUATION

The model evaluation analysis will be done by the accuracy scores calculated with Equation 4, and runtime in seconds to perform prediction. The results of CLSAF model using Power Cepstrum or Autocorrelation Function for different % of lags using ConvLSTM will be compared with ConvLSTM model, and Persistence model, when using different autoregressive features as input, different time series resolution, and different input length.

3.6.1. POWER CEPSTRUM AND ACF DIFFERENCES

Different theta threshold will be tested for CLSAF models, to better compare the models the theta threshold will be selected according to the % of lags using the ConLSTM (Main state model), the percentages of time steps are 90%, 80%, 70%, 60%, and 50%.

3.6.2. MODEL SCORE RESULTS FOR AUTOREGRESSIVE INPUT ANALYSIS

The autoregressive input will be tested with y[t-1] and y[t-pn] for one hour resolution and one input length. The autoregressive feature y[t-pn] is selected by the highest Power Cepstrum or ACF amplitude, if highest amplitude is higher than theta threshold, the lag with highest autocorrelation amplitude will be selected as input for ConvLSTM, different for y[t-1], if the Power Cepstrum or ACF amplitude is higher than the theta threshold the input lag for ConvLSTM will be always the y[t-1]. This represents step 3.1 from flowchart Figure 3.2.

3.6.3. MODEL SCORE RESULTS FOR RESOLUTION ANALYSIS

The different time series resolution will be tested for one hour, 30 minutes and one minute, using autoregressive input always as y[t-1] and one input length. This represents the step 2.1 from flowchart Figure 3.2.

3.6.4. MODEL SCORE RESULTS INPUT LENGTH ANALYSIS

The different input length will be tested for 1,2,3,4,5,6, 12 and 24 input length, using autoregressive input always as y[t-1 to t-input length] and the time series resolution for one hour. The ConvLSTM parameter changed compared to others analysis, for this one will be used 50 epochs as described in **Appendix 6**.

4. RESULTS AND DISCUSSION

The results and discussion start with the month predictable analysis using the pattern behavior of electricity consumption for each month of 2009 to compare the strength of day cycles using FFT and the volatility of the energy load of the occupant behavior for each month of 2009, according to (Li et al. 2021), time series with low volatility and high pattern behavior have better model accuracy.

The results from predictable analysis resulted in the months used to test benchmark models using the most predictable from 2009. After selecting the months, the theta thresholds are calculated to obtain the same percentage of time steps using the ConvLSTM model when CLSAF models are tested with Power Cepstrum or ACF as autoregressive feature selection, this can guarantee correct comparative analysis between the two methods. At the end, the results are analyzed for the benchmark models and commented accordingly.

4.1. MOST PREDICTABLE MONTH FOR HOUR RESOLUTION

The analysis to identify the most predictive month for this dataset is performed according to the results of (Li et al. 2021), which show better model performance for time series with strong daily pattern behavior. For this, the author used Discrete Fourier Transform from *Equation 5* and CV-Observation from *Equation 6*. The results, when applied to global active power consumption for 2009, show a pattern behavior of one and two cycles per day with higher Spectrum values, as we can see in Figure 4.1.1.

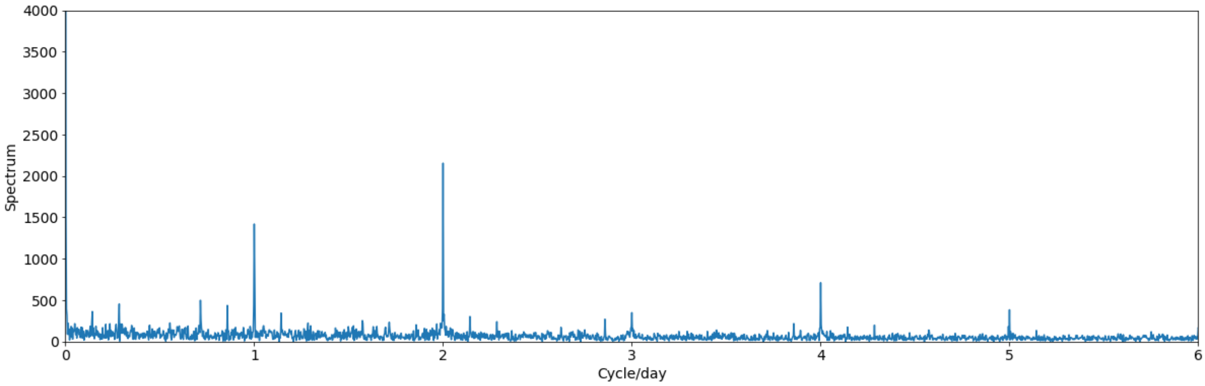


Figure 4.1.1 Spectrum analysis for global active power consumption for 2009 with a frequency of cycles per day using hour resolution.

The analysis for each month of 2009 was performed by applying the *Equation 6* and *Equation 5*, to summarize the Spectrum amplitude, an average of the amplitude for one and two cycles per day was taken and compared with the volatility of each month, which is shown in Figure 4.1.2. The results show that the best results were obtained in the months of April, May, March and December with a high Average Spectrum Amplitude and a low CV-Observation.

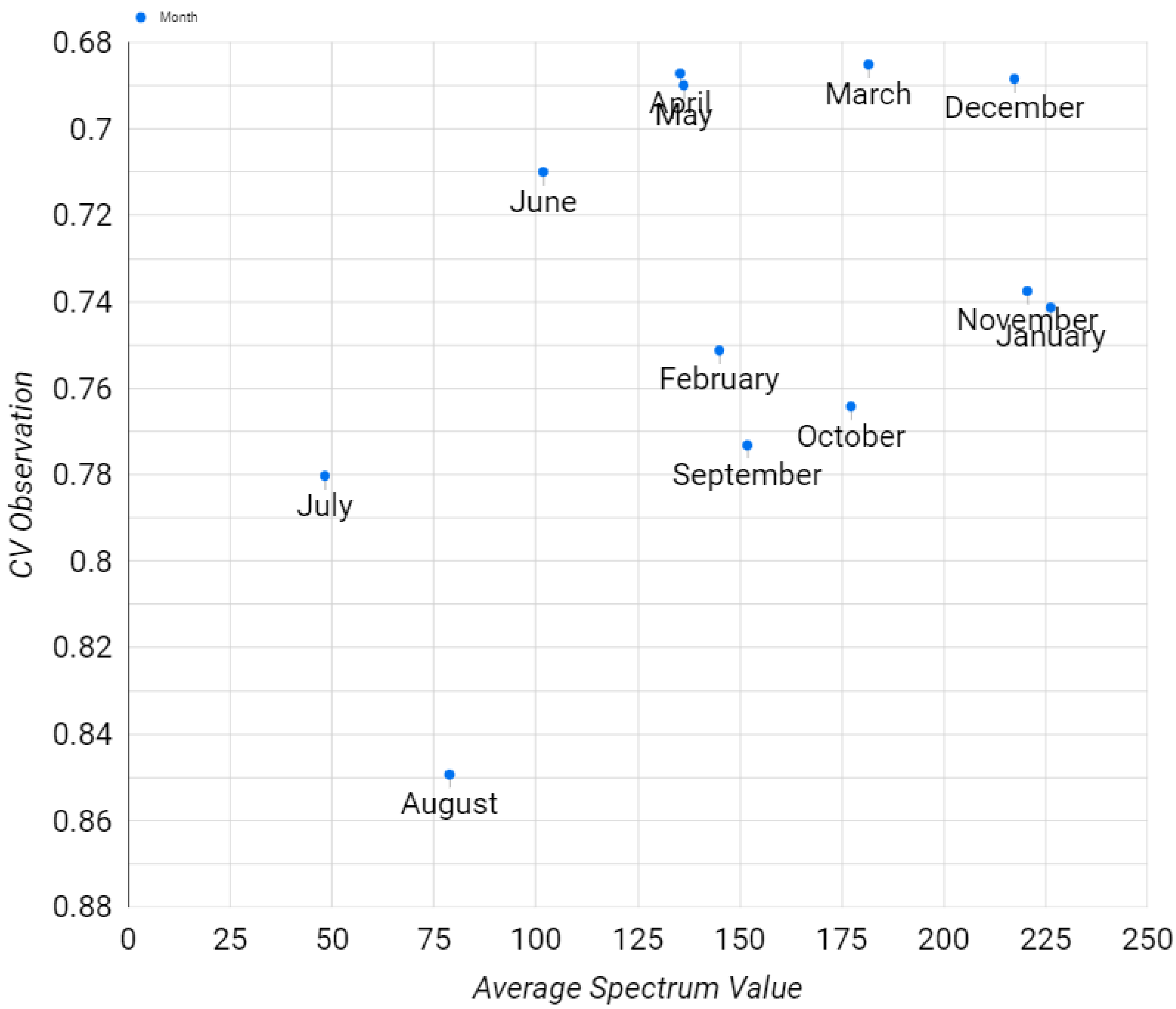


Figure 4.1.2 Scatter plot of CV Observations and Average Spectrum Value for each month of 2009. Indicating the most predictable months.

The analysis of electricity consumption behavior can be illustrated with the mean value of global active power consumption per time of day for each month, with high energy consumption observed in the morning and evening for the winter, fall, and spring seasons, as represented in Figure 4.1.3.

Month / Global_active_power													
time	Jan...	Feb...	Mar...	April	May	June	July	Aug...	Sep...	Oct...	Nov...	Dec...	Grand total
0													0.81
1													0.57
2													0.49
3													0.45
4													0.43
5													0.45
6													0.46
7													0.81
8													1.56
9													1.48
10													1.37
11													1.26
12													1.26
13													1.2
14													1.11
15													1.01
16													0.91
17													0.96
18													1.12
19													1.34
20													1.71
21													1.9
22													1.83
23													1.33
Grand total	1.41	1.25	1.23	1.14	1.01	0.83	0.62	0.66	0.99	1.14	1.27	1.36	1.08

Figure 4.1.3 Average Global Active Power for each time of the day by month of the year for the year 2009 dataset by hour.

4.2. Power Cepstrum and Autocorrelation Function differences

Each time step uses the previous lags to select the highest autocorrelated lag amplitude and compare it to the theta threshold, as shown in the flowchart in Figure 3.2. In this study, the goal is to compare the Autocorrelation Function (ACF) and Power Cepstrum as an autoregressive feature selector. To do this, it is necessary to determine the percentage of time steps using ConvLSTM for each type and month.

The test is performed with autoregressive analysis of 168 hours of previous lags for each time step in a hour resolution dataset, the % of Time step using ConvLSTM is defined as 50%, 60%, 70%, 80%, and 90%, the theta threshold is defined according to the quantity of lags with highest autocorrelated lag amplitude higher than the theta threshold value set, for example, to achieve 90% of lags using ConvLSTM in a month with 23 days (the first 7 days are used for warm-up period as described in Figure 3.2), the total of time steps available are 552 hours, where 497 hour time steps needs to have the highest autocorrelated lag amplitude from autoregressive analysis higher than the theta threshold set.

The months evaluated are those selected as the most predictable, March, April, May, and December. The full table showing the theta threshold used to evaluate each type of autocorrelation calculation and the percentage of time step usage for each month. The results in Table 4.2.1.

Table 4.2.1 Theta Threshold that will be used to test for each type of autocorrelation calculation, each % of time step usage and for each month.

					Month / Theta Threshold
Type	% of Time Step Using ConvLSTM	March	April	May	December
Power_Cepstrum	90.00%	0.051446	0.050982	0.051921	0.050136
	80.00%	0.062307	0.057665	0.058908	0.061060
	70.00%	0.070363	0.063902	0.062649	0.074317
	60.00%	0.074829	0.069820	0.068355	0.082587
	50.00%	0.083827	0.076930	0.075410	0.089002
ACF	90.00%	0.571797	0.592953	0.549751	0.646807
	80.00%	0.584284	0.599591	0.577454	0.653260
	70.00%	0.592835	0.610852	0.595212	0.666220
	60.00%	0.601250	0.625570	0.610044	0.675272
	50.00%	0.609981	0.634244	0.622375	0.691317

The analysis of the number of time steps with the highest autocorrelated lag amplitude for each computation type is shown in Table 4.2.2. The AR feature is the position of the autoregressive feature lag of the time step with the highest autocorrelation amplitude for the 168 previous hours that are above the specified theta threshold. The results show that both types have the same position, “y-2”, of the autoregressive feature position with the highest amplitude.

Table 4.2.2 Quantity of time steps with highest autocorrelation amplitude for each type, each month and for each % of time step usage.

Type	Month ▲	AR Feature	50% of Time Steps	60% of Time Steps	70% of Time Steps	80% of Time Steps	90% of Time Steps
ACF	March	y-2	289	346	404	461	519
ACF	April	y-2	277	332	387	442	497
ACF	May	y-2	289	346	404	461	519
ACF	December	y-2	289	346	404	461	519
Power_Cepstrum	March	y-85	0	0	0	2	2
Power_Cepstrum	March	y-2	289	346	404	459	517
Power_Cepstrum	April	y-2	277	332	387	442	497
Power_Cepstrum	May	y-2	289	346	404	461	519
Power_Cepstrum	December	y-2	289	346	404	461	519

The analysis comparing the time steps using ConvLSTM when performing Power Cepstrum or ACF shows that not every step is the same. For example, when 50% of the lags uses ConvLSTM for March, the results show that 67.36% of the time steps are the same when using Power Cepstrum or ACF, indicating that 32.64% of the time steps using ConvLSTM are different when comparing the two types as shown in Table 4.2.3.

Table 4.2.3 Results comparing % of same time steps that use ConvLSTM when using Power Cepstrum or ACF.

date (Year Mont...	50% of Time steps	60% of Time steps	70% of Time steps	80% of Time steps	90% of Time steps
March	67.36%	64.24%	66.67%	76.74%	91.67%
April	65.94%	65.58%	66.67%	72.46%	82.97%
May	73.61%	80.90%	81.94%	89.58%	87.50%
December	80.21%	85.07%	83.68%	76.04%	80.90%

The difference of the time steps using ConvLSTM when using Power Cepstrum or ACF was analyzed by summing the number of time steps with highest autocorrelation amplitude higher than theta threshold per hour of the day, for each % of time steps using ConvLSTM, and each type of autoregressive selector, Power Cepstrum and ACF. The analysis was done with the months of March, April, May, and December, a total of 93 days of prediction (121 days minus 28 of warm-up period).

In the Figure 4.2.1 and Figure 4.2.2 the results for Power Cepstrum and ACF are shown respectively.

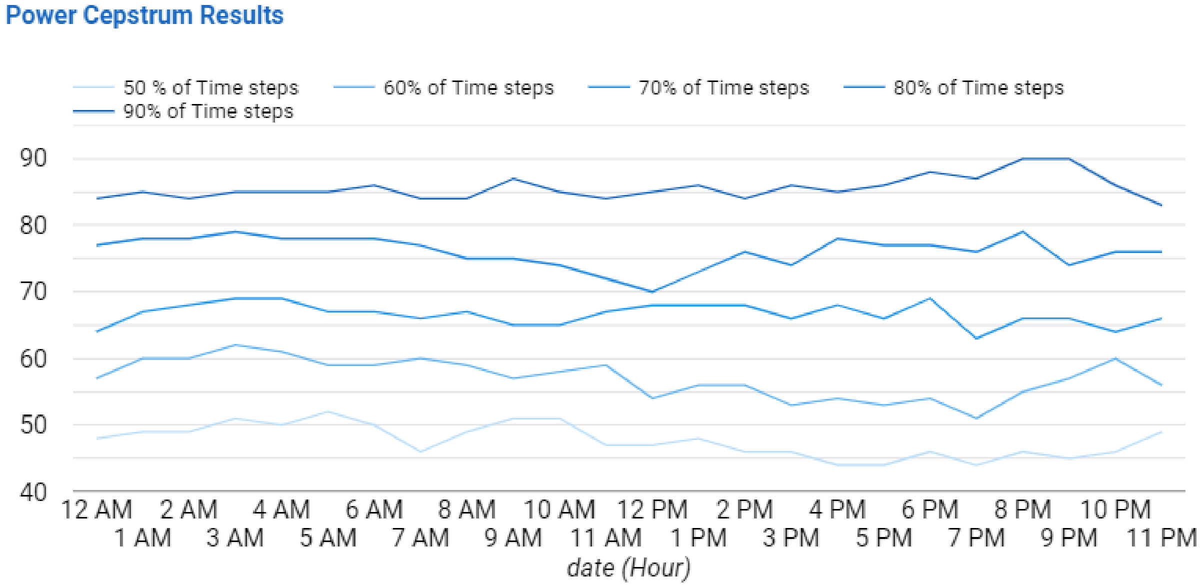


Figure 4.2.1 Sum of time steps using ConvLSTM when using Power Cepstrum for the months of March, April, May, and December.

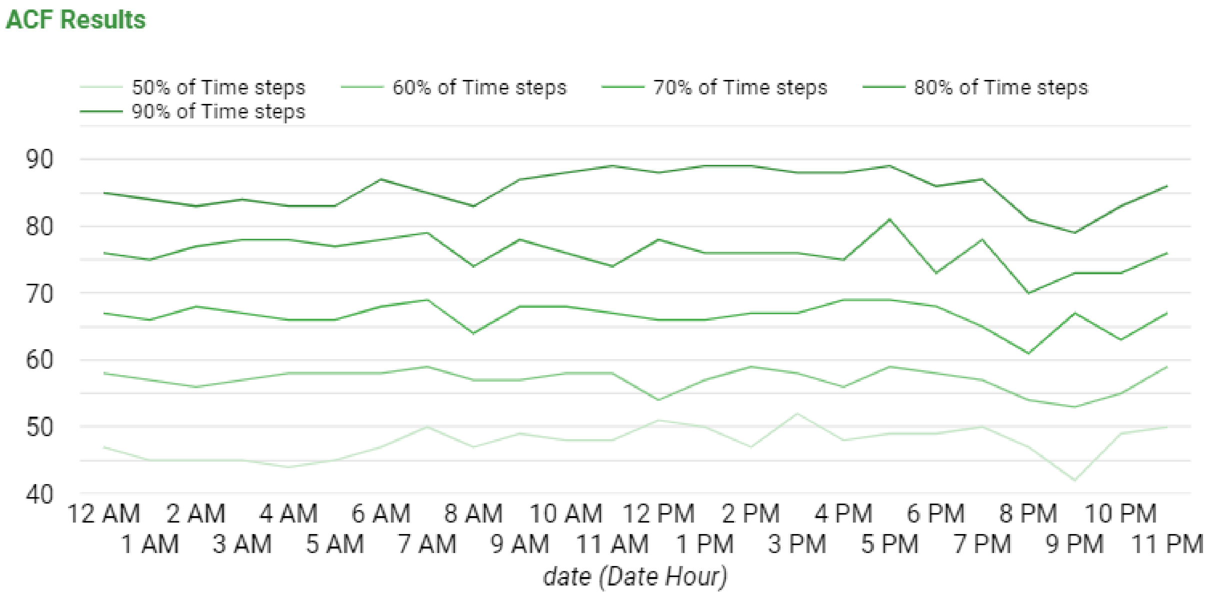


Figure 4.2.2 Sum of time steps using ConvLSTM when using ACF for the months of March, April, May, and December.

The results show that there is a pattern with a higher quantity of time steps using ConvLSTM during 9 AM and 8 PM for Power Cepstrum, according to Figure 4.1.3, these are the moments of the day with a higher global active power average for all % of time steps using ConvLSTM.

The pattern observed for ACF shows a higher quantity of time steps using ConvLSTM during the afternoon, between 10 AM and 6 PM, the time when the global active power average is lower for all % of time steps using ConvLSTM.

The Figure 4.2.3 shows the sum of time steps using ConvLSTM that are different when using Power Cepstrum or ACF as autoregressive selector, the differences are for results with 90% of time steps using ConvLSTM and for months of March, April, May, and December, a total of 93 days of prediction (121 days minus 28 of warm-up period).

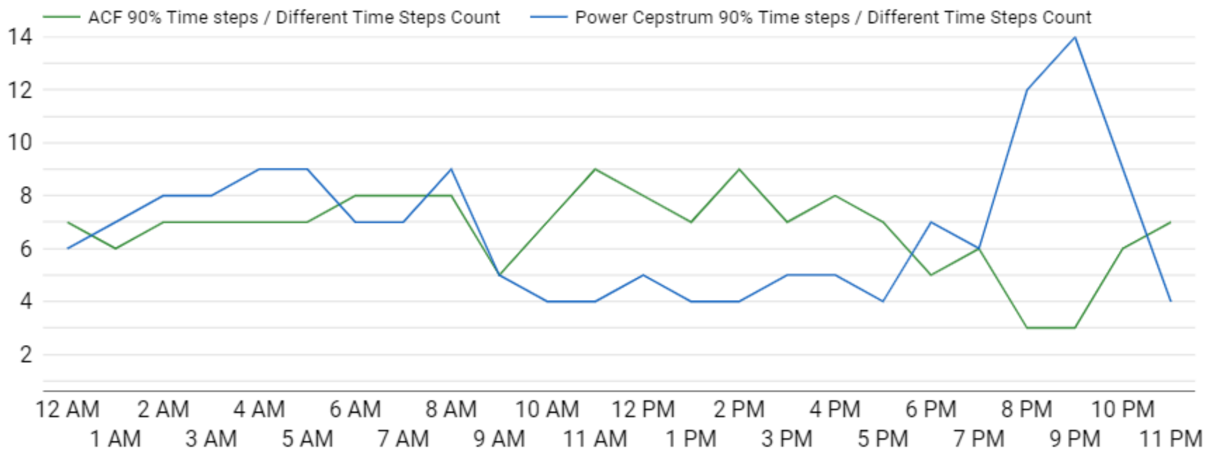


Figure 4.2.3 Quantity of different time steps using ConvLSTM when using ACF or Power Cepstrum for the months of March, April, May, and December for 90% of time steps using ConvLSTM.

The highest quantity of different time steps using ConvLSTM are between 9 AM and 5 PM with higher quantity for ACF and between 7 PM and 10 PM with higher quantity for Power Cepstrum.

This shows that when ACF is used as autoregressive selector for time steps between 9 AM and 5 PM, the autocorrelation amplitude of the highest lag for this period is higher than the autocorrelation amplitude of the highest lag between 8 PM and 10 PM, indicating a higher autocorrelation amplitude for afternoon patterns for 90% of the lags using ConvLSTM

The same reasoning for Power Cepstrum, since Power Cepstrum has high quantity of time steps selected during 8 PM and 10 PM, indicating that when using Power Cepstrum the autocorrelation amplitudes are higher for night patterns for 90% of the lags using ConvLSTM.

4.3. MODEL SCORE RESULTS FOR AUTOREGRESSIVE TIME STEP INPUT ANALYSIS

Tests were performed by repeating each configuration of the benchmark models described in Table 3.4, 10 times for each theta threshold from Table 4.2.1, evaluation is done selecting the best scores from CV -Residual (%), calculated using Equation 4.

The model with highest accuracy is the ConvLSTM model with an average score of 0.5274 in the "Grand total" column, but not so far, CLSAF - ACF using autoregressive lag as (t-1) for the input and CLSAF - Power Cepstrum using autoregressive lag as (t-1) for the input, the best scores values are 0.5313 and 0.5302, respectively, for 90% of the time steps using ConvLSTM.

When compared to the persistence model, with score of 0.5731, an improvement of 6.5% and 1.9% lower than ConvLSTM model accuracy, the positive effect is the time consumption of the CLSAF model, which is lower than ConvLSTM. The score results for 60% of the time steps using ConvLSTM for CLSAF - ACF using autoregressive lag as (t-1) and CLSAF - Power Cepstrum using autoregressive lag as (t-1), the best scores values are 0.5380 and 0.5397, respectively, maintaining the score while reducing computational cost. The scores results are similar when comparing ACF and Power Cepstrum, except that ACF performs better than Power Cepstrum at 50% of the time steps using ConvLSTM (0.5394 versus 0.5459).

It is noticeable that December is a very good month for predictions, but at low volatility the persistence model performs very well, with similar results compared to ConvLSTM and CLSAF with autoregressive lag as (t-1). For March, April and May, a higher improvement is observed when comparing the models to the persistence model, for example in March with an improvement of 14%.

The results for the CLSAF models using autoregressive lags as input, y[t-pn] show poor performance compared to Persistence and the other models. The results in *Table 4.3.1*.

Table 4.3.1 Model CV-Residual score results for hour resolution on March, May, April and December of 2009, by % of time steps using ConvLSTM.

Model	% of Time Steps	March	April	May	December	**Grand total**
ConvLSTM	100.00%	0.5188	0.5291	0.5671	0.4944	**0.5274**
CLSAF - ACF y[t-1]	90.00%	0.5268	0.5282	0.5747	0.4954	**0.5313**
	80.00%	0.5369	0.5280	0.5654	0.4992	**0.5324**
	70.00%	0.5337	0.5280	0.5788	0.4908	**0.5328**
	60.00%	0.5464	0.5381	0.5766	0.4909	**0.5380**
	50.00%	0.5519	0.5463	0.5786	0.4807	**0.5394**
CLSAF - Power Cepstrum y[t-1]	90.00%	0.5247	0.5286	0.5608	0.5067	**0.5302**
	80.00%	0.5313	0.5319	0.5709	0.4963	**0.5326**
	70.00%	0.5495	0.5356	0.5748	0.4973	**0.5393**
	60.00%	0.5571	0.5398	0.5689	0.4930	**0.5397**
	50.00%	0.5632	0.5488	0.5755	0.4961	**0.5459**
Persistence	0.00%	0.5982	0.5907	0.6011	0.5025	**0.5731**
CLSAF - ACF y[t-pn]	90.00%	0.6037	0.6191	0.6549	0.5825	**0.6151**
	80.00%	0.6064	0.6158	0.6425	0.5806	**0.6113**
	70.00%	0.6036	0.5956	0.6405	0.5699	**0.6024**
	60.00%	0.6004	0.5901	0.6336	0.5640	**0.5970**
	50.00%	0.5922	0.5904	0.6246	0.5415	**0.5872**
CLSAF - Power Spectrum y[t-pn]	90.00%	0.6012	0.6333	0.6491	0.5954	**0.6197**
	80.00%	0.6005	0.6269	0.6363	0.5886	**0.6131**
	70.00%	0.5986	0.6235	0.6244	0.5739	**0.6051**
	60.00%	0.6050	0.6149	0.6275	0.5680	**0.6039**
	50.00%	0.5991	0.6072	0.6207	0.5664	**0.5984**

The main advantage of the CLSAF model is the lower runtime consumption, with a difference of 7 seconds to execute the prediction, while maintaining model accuracy. This difference can be very significant for higher resolution, as we will see in the test with 30 minutes and 1 minute resolution.

The results show that CLSAF has better runtime performance when compared to ConvLSTM for % of time steps using ConvLSTM lower than 90%. The results to compare Power Cepstrum and ACF show that ACF has better runtime performance for 90% and 80% of time steps using ConvLSTM, but for 50% to 80% we observed similar runtime results. The complete results in Table 4.3.2.

Table 4.3.2 Time in seconds to run prediction by model and % of time steps using ConvLSTM for March, April, May, and December of 2009.

Model	% of Time Steps	Time (seconds)
CLSAF - Power Cepstrum y[t-1]	90.00%	45.32
	80.00%	43.62
	70.00%	42.21
	60.00%	39.69
	50.00%	37.62
CLSAF - ACF y[t-1]	90.00%	42.25
	80.00%	41.84
	70.00%	41.39
	60.00%	39.92
	50.00%	38.00
ConvLSTM	100.00%	45.46

The CLSAF-Power Cepstrum y[t-1] and CLSAF-ACF y[t-1] models with 70% of the time steps using ConvLSTM, from 13-04-2009 to 25-04-2009, were shown in Figure 4.3.1 and Figure 4.3.2 with detailed information about the time steps using Persistence or ConvLSTM.

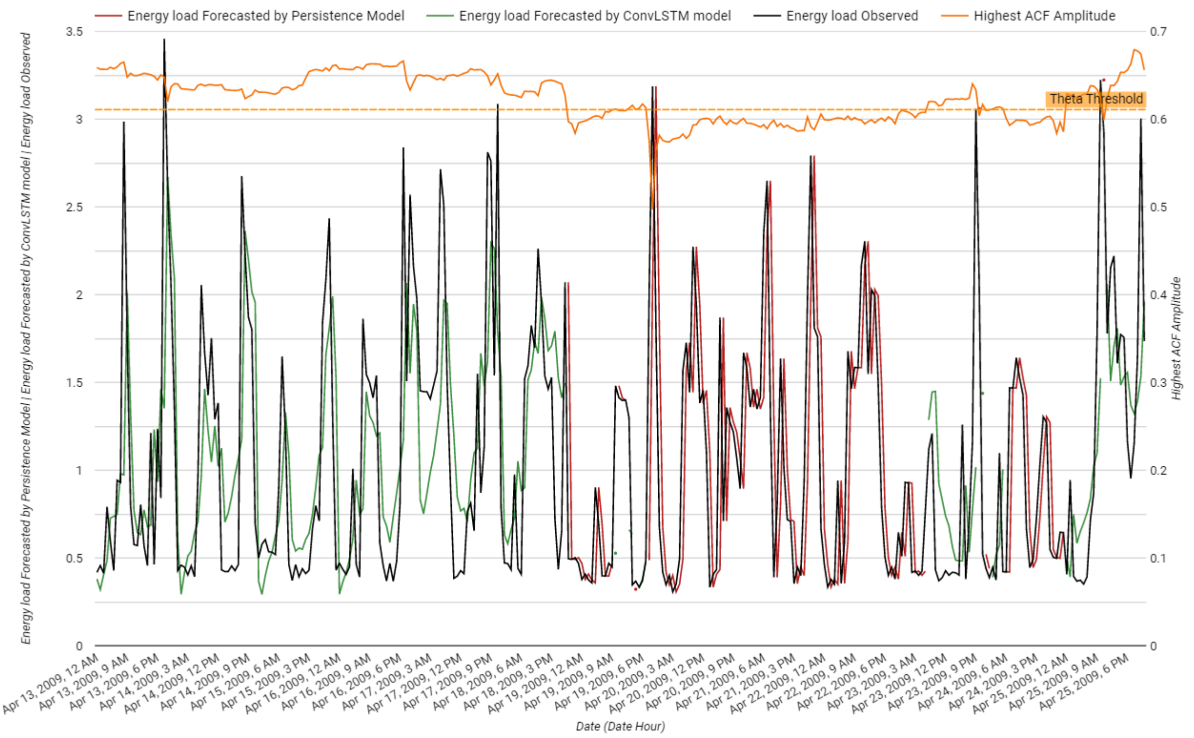


Figure 4.3.1 CLSAF-ACF y[t-1] model prediction results from 13-04-2009 to 25-04-2009, with 70% of time step using ConvLSTM which is theta threshold of 0,610852283363079 and 160 hours of warm-up.

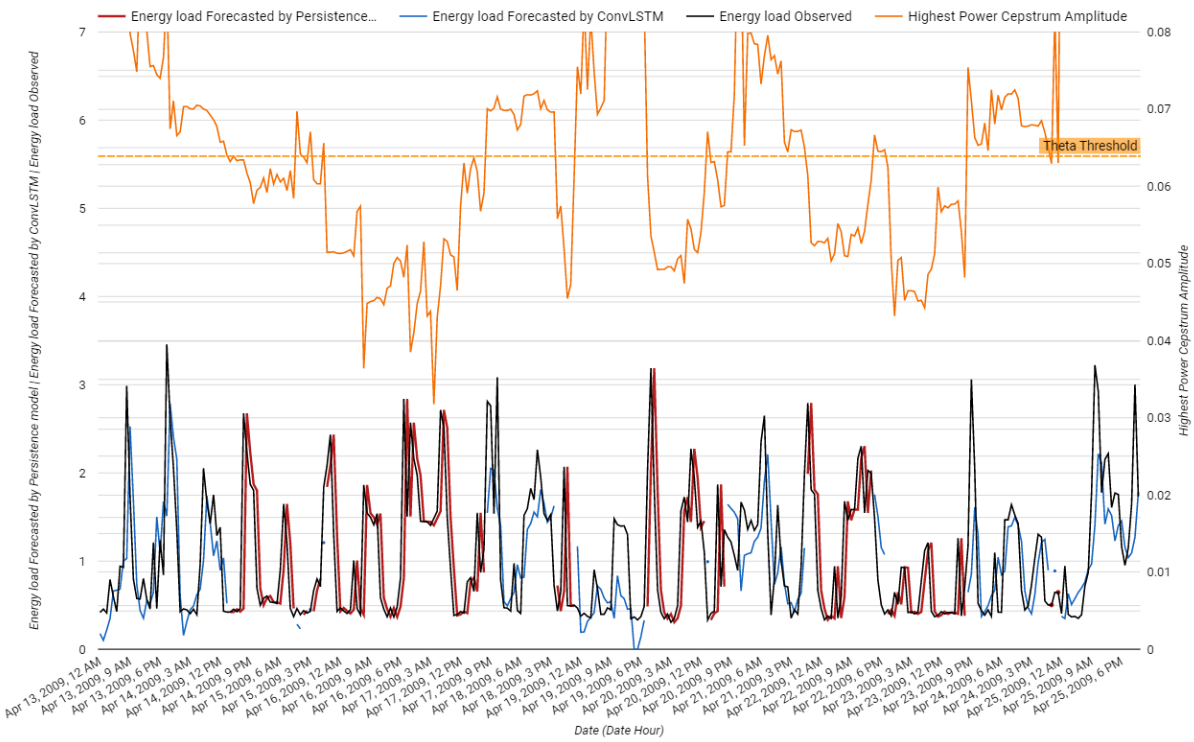


Figure 4.3.2 CLSAF-Power Cepstrum y[t-1] model prediction results from 13-04-2009 to 25-04-2009 with 70% of time step using ConvLSTM which is theta threshold of 0,0639021864467233 and 160 hours of warm-up.

The yellow dashed line indicates the theta threshold. When the highest value of the autocorrelation amplitude in the yellow line is below the theta threshold, the models use the Persistence model; in the red line, when the amplitude is higher, the models use the ConvLSTM model; in the green line for ACF; and in the blue line for Power Cepstrum.

4.4. Model score results for resolution analysis

Following the tests for hour resolution, the first step for higher resolution analysis was to identify the theta threshold for each % time steps using ConvLSTM defined before, for this analysis will be used the month of April in 2009 only and the analysis will be done for 1 minute and 30-minute resolution. The results in Table 4.4.1.

Table 4.4.1Theta threshold values for one minute and 30 minutes resolution for Power Cepstrum and ACF and for each % of time steps using ConvLSTM in April 2009.

		Resolution / Theta Threshold	
⊟ Type	% of time steps	1 Minute	30 Minute
Power_Cepstrum	90.00%	0.137822	0.072104
	80.00%	0.178384	0.080827
	70.00%	0.199799	0.087012
	60.00%	0.216199	0.092681
	50.00%	0.234025	0.098915
ACF	90.00%	0.835712	0.657827
	80.00%	0.866142	0.669192
	70.00%	0.886346	0.689335
	60.00%	0.903944	0.702136
	50.00%	0.917894	0.713100

At 30 minute resolution, the results show great performance of the CLSAF using autoregressive input lag as y[t-1] models compared to Persistence and ConvLSTM, with a CV-Residual(%) for CLSAF- Power Cepstrum using autoregressive lag as y[t-1] of 0.5182 when using 90% of the time steps with ConvLSTM, 6.3% better than the Persistence model and 1.59% better than ConvLSTM and the performance for other % of the time steps remains acceptable considering that the runtime is lower when using lower % of Time steps as we can see in Table 4.4.3. At 1 minute resolution, the results show that the models are worse than the Persistence model and are not usable. The results in Table 4.4.2.

Table 4.4.2 Model CV-Residual score results for minute and 30-minute resolution on April of 2009, by % of time steps using ConvLSTM.

		Resolution / score	
Model	% of Time steps	30 Minutes	1 Minute
Persistence	0.00%	0.5534	0.2252
ConvLSTM	100.00%	0.5266	0.2379
CLSAF - Power Cepstrum y[t-1]	90.00%	0.5182	0.2307
	80.00%	0.5234	0.2293
	70.00%	0.5277	0.2270
	60.00%	0.5304	0.2254
	50.00%	0.5401	0.2252
CLSAF - ACF y[t-1]	90.00%	0.5242	0.2317
	80.00%	0.5228	0.2317
	70.00%	0.5251	0.2308
	60.00%	0.5272	0.2294
	50.00%	0.5330	0.2306

Results for run time only for the 30-minute resolution because the persistence model was the best model for the 1-minute resolution. As expected, the time difference between the ConvLSTM and CLSAF models in running the prediction is larger at higher resolution, with an improvement of about 40% compared to 50% of the time steps. It can be observed that CLSAF when using Power Cepstrum is 11% faster compared to ACF for higher resolution.

Table 4.4.3 Time in seconds for 30-minute resolution to run prediction by model, % of time steps using ConvLSTM for April of 2009.

		Resolution / Time (seconds)
Model	% of Time Steps	30 Minutes
ConvLSTM	100.00%	124.34
CLSAF - Power Cepstrum y[t-1]	90.00%	88.99
	80.00%	84.67
	70.00%	79.56
	60.00%	77.59
	50.00%	73.23
CLSAF - ACF y[t-1]	90.00%	95.50
	80.00%	92.20
	70.00%	87.44
	60.00%	85.26
	50.00%	82.15

4.5. MODEL SCORE RESULTS FOR INPUT LENGTH ANALYSIS

The results for higher inputs were run using the same theta threshold as in Table 4.4.1, since the highest autocorrelated amplitude for the hourly resolution remains the same. The month chosen for testing is the same as for the higher resolution, April 2009. As mentioned in the methodology, the ConvLSTM configuration was changed for higher inputs as shown in **Appendix 6**, with better results observed at 50 epochs. Tests were performed for input lengths of 1, 2, 3, 4, 5, 6, 12, and 24. The results show that the score improves with higher input lengths. The best average score was observed for a 6-step input, 0.89% better than a single step input. The best score, 0.5118 CV -Residual(%), was obtained from CLSAF - Power Cepstrum with 80% of lags using the ConvLSTM model. Detailed results in Table 4.5.1.

Table 4.5.1 Model CV-Residual score results for hour resolution on different input length, in April of 2009, by % of time steps using ConvLSTM.

		Input Length / score							
⊟ Model	% of Time steps	1	2	3	4	5	6	12	24
Persistence	0.00%	0.5907	0.5907	0.5907	0.5907	0.5907	0.5907	0.5907	0.5907
CLSAF-ACF y[t-1 to t-length]	90.00%	0.5263	0.5252	0.5374	0.5276	0.5228	0.5197	0.5414	0.5445
	80.00%	0.5186	0.5213	0.5207	0.5264	0.5203	0.5206	0.5191	0.5658
	70.00%	0.5222	0.5263	0.5244	0.5236	0.5351	0.5258	0.5291	0.5906
	60.00%	0.5348	0.5447	0.5422	0.5473	0.5345	0.5396	0.5339	0.5432
	50.00%	0.5381	0.5455	0.5474	0.5466	0.5436	0.5418	0.5501	0.5445
ConvLSTM	100.00%	0.5215	0.5237	0.5249	0.5222	0.5205	0.5186	0.5171	0.5832
CLSAF-Power Cepstrum y[t-1 to t-length]	90.00%	0.5215	0.5207	0.5180	0.5213	0.5245	0.5163	0.5240	0.5826
	80.00%	0.5264	0.5219	0.5304	0.5218	0.5209	0.5118	0.5249	0.5615
	70.00%	0.5307	0.5289	0.5281	0.5296	0.5183	0.5184	0.5162	0.5761
	60.00%	0.5358	0.5388	0.5364	0.5356	0.5391	0.5264	0.5393	0.5579
	50.00%	0.5477	0.5554	0.5456	0.5502	0.5449	0.5326	0.5382	0.5495

The results for runtime show similar behavior as previous analysis, when using lower % of time steps the runtime performance of the model increases, it is possible to observe that 6 input length has the best runtime performance, this is because warm-up period for higher length have lower training time steps compared to one length input, and the matrix multiplication time consumption is not as high when compared to 24 input length. Results shown in Table 4.5.2.

Table 4.5.2 Time in seconds to run prediction for each length by model and % of time steps using ConvLSTM for April of 2009.

		Input Length / Time (seconds)							
⊟ Model	% of Time Steps	1	2	3	4	5	6	12	24
ConvLSTM	100.00%	53.40	51.71	51.35	51.26	51.28	50.71	51.88	51.44
CLSAF-Power Cepstrum y[t-1 to t-length]	90.00%	50.39	50.04	50.08	50.15	50.02	49.46	50.16	50.73
	80.00%	48.59	48.20	48.84	48.21	47.93	47.88	48.53	48.89
	70.00%	46.42	45.85	45.94	46.39	45.74	46.32	46.00	47.04
	60.00%	44.54	44.61	43.98	43.43	43.73	43.91	44.95	45.02
	50.00%	43.09	42.34	42.30	42.24	42.44	41.96	42.51	43.05
CLSAF-ACF y[t-1 to t-length]	90.00%	48.75	47.88	48.86	49.23	48.42	47.93	48.32	48.63
	80.00%	44.25	46.35	46.93	47.18	46.22	46.23	46.53	47.24
	70.00%	44.64	45.14	44.66	45.20	44.98	44.31	44.76	45.17
	60.00%	43.28	42.90	43.06	43.08	42.59	43.32	43.06	42.97
	50.00%	41.49	41.03	41.08	40.87	40.77	41.13	41.39	40.65

5. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORKS

The limitations found in this study was the time consumption to evaluate multiple theta threshold values for each month in multiple resolutions using multiple ConvLSTM parameters.

Testing in others dataset that contains different energy load behavior can lead to different conclusions and improve results for CLSAF model when occupancy is not so frequent.

Apply the model for months that are considered not good for predictions and test new ways to find good results for these months to achieve great performance in all periods of the year.

Test different configuration for input length, warm-up period, and autoregressive feature input.

Implement the CLSAF model in real world projects automatizing theta threshold optimization in the best predictable periods.

6. CONCLUSIONS

In this study, the CLSAF model was tested with Power Cepstrum to select the autoregressive feature to achieve better model accuracy and runtime performance. The tests were performed with different configurations for input lag selection, input length and resolution.

The results show that on average, more than 70% of the time steps used for Power Cepstrum and Autocorrelation Function are the same when comparing % of lags using ConvLSTM. The difference is the result of a higher Power Cepstrum amplitude for the night hours, between 8 pm to 10 pm, when compared to 9 am to 5 pm hours, the inverse rationale for ACF, that contains highest amplitude for 9 am to 5 pm when compared to 8 pm to 10 pm.

The average CLSAF scores are similar for Power Cepstrum and ACF when comparing the percentage of lags using ConvLSTM, the average CLSAF runtime was better when using Power Cepstrum for higher resolutions compared to ACF. The autoregressive feature used as input for ConvLSTM showed better performance when using the last step known lag, y[t-1] compared to the highest autocorrelated amplitude lag, y[t-n].

Results for one minute resolution showed better performance for the Persistence model, although for 30 minutes the models CLSAF y[t-1] and ConvLSTM had better accuracy when compared to persistence model.

High input length improved model accuracy when using CLSAF and ConvLSTM models. The CLSAF model with Power Cepstrum as the autoregressive feature selector, with 80% of lags using ConvLSTM, and 6 input length achieved the best model accuracy score for hour resolution when compared to different input length.

The best model performance is for CLSAF model using Power Cepstrum, using autoregressive features as y[t-1 to t-length], with 80% of time steps using ConvLSTM, and 6 length input for hour resolution in April 2019, compared to ConvLSTM and Persistence model the improvement in score is 1,3% and 13,3% respectively, the improvement in time consumption to run the all month prediction when compared to ConvLSTM was improved in 8,6%.

The CLSAF model can reduce time consumption without losing accuracy when compared to ConvLSTM. The method showed great performance using just one week of historical data, which makes it easier to apply. The theta threshold presented better results for 80% of time steps using ConvLSTM. Is possible to conclude that Power Cepstrum can replace and outperform Autocorrelation Function in accuracy and runtime.

7. BIBLIOGRAPHY

Dutta, Shreya, Yanling Li, Aditya Venkataraman, Luis M. Costa, Tianxiang Jiang, Robert Plana, Philippe Tordjman, Fook Hoong Choo, Chek Fok Foo, and Hans B. Puttgen. 2017. “Load and Renewable Energy Forecasting for a Microgrid Using Persistence Technique.” In *Energy Procedia*, 143:617–22. Elsevier Ltd. https://doi.org/10.1016/j.egypro.2017.12.736.

Fang, Zheng, Jianying Xie, Ruiming Peng, and Sheng Wang. 2021. “Climate Finance: Mapping Air Pollution and Finance Market in Time Series.” *Econometrics* 9 (4). https://doi.org/10.3390/econometrics9040043.

Hochreiter, Sepp. 1998. “The Vanishing Gradient Problem during Learning Recurrent Neural Nets and Problem Solutions.” *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems* 6 (2). https://doi.org/10.1142/S0218488598000094.

Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. “Long Short-Term Memory.” *Neural Computation* 9 (8). https://doi.org/10.1162/neco.1997.9.8.1735.

Hyndman, Rob J. 2006. “Another Look at Forecast-Accuracy Metrics for Intermittent Demand.” *Foresight: The International Journal of Applied Forecasting* 4.

Jain, Anil K., Jianchang Mao, and K. M. Mohiuddin. 1996. “Artificial Neural Networks: A Tutorial.” *Computer*. https://doi.org/10.1109/2.485891.

Kalpakis, Konstantinos, Dhiral Gada, and Vasundhara Puttagunta. 2001. “Distance Measures for Effective Clustering of ARIMA Time-Series.” In *Proceedings - IEEE International Conference on Data Mining, ICDM*. https://doi.org/10.1109/icdm.2001.989529.

Kuster, Corentin, Yacine Rezgui, and Monjur Mourshed. 2017. “Electrical Load Forecasting Models: A Critical Systematic Review.” *Sustainable Cities and Society*. https://doi.org/10.1016/j.scs.2017.08.009.

Lauwers, Oliver, Christof Vermeersch, and Bart de Moor. 2022. “Cepstral Identification of Autoregressive Systems.” *Automatica* 139. https://doi.org/10.1016/j.automatica.2022.110214.

Li, Lechen, Christoph J. Meinrenken, Vijay Modi, and Patricia J. Culligan. 2021. “Short-Term Apartment-Level Load Forecasting Using a Modified Neural Network with Selected Auto-Regressive Features.” *Applied Energy* 287. https://doi.org/10.1016/j.apenergy.2021.116509.

Mikolov, Tomaš, Martin Karafiát, Lukaš Burget, Cernocky Jan, and Sanjeev Khudanpur. 2010. “Recurrent Neural Network Based Language Model.” In *Proceedings of the 11th Annual Conference of the International Speech Communication Association, INTERSPEECH 2010*. https://doi.org/10.21437/interspeech.2010-343.

Proedrou, Elisavet. 2021. “A Comprehensive Review of Residential Electricity Load Profile Models.” *IEEE Access*. https://doi.org/10.1109/ACCESS.2021.3050074.

Shi, Xingjian, Zhourong Chen, Hao Wang, Dit Yan Yeung, Wai Kin Wong, and Wang Chun Woo. 2015. "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting." In *Advances in Neural Information Processing Systems*. Vol. 2015-January.

Sood, R, and I Koprinska. 2010. "IEEE Xplore - Electricity Load Forecasting Based on Autocorrelation Analysis." *Neural Networks (IJCNN)*.

Steg, Linda, Goda Perlaviciute, and Ellen van der Werff. 2015. "Understanding the Human Dimensions of a Sustainable Energy Transition." *Frontiers in Psychology*. https://doi.org/10.3389/fpsyg.2015.00805.

Tuomela, Sanna, Mauricio de Castro Tomé, Netta Iivari, and Rauli Svento. 2021. "Impacts of Home Energy Management Systems on Electricity Consumption." *Applied Energy* 299. https://doi.org/10.1016/j.apenergy.2021.117310.

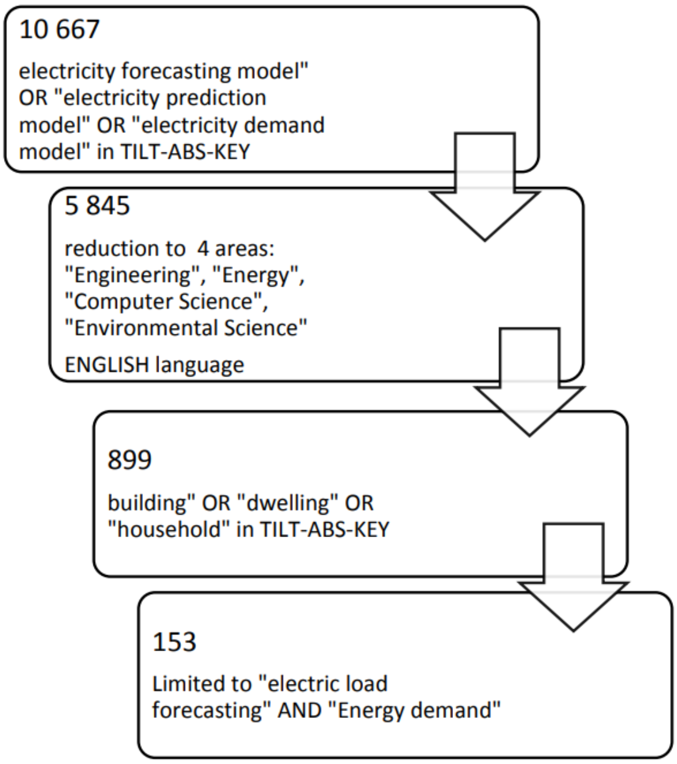
Tuomisto, Mikko. 2016. "Non-Intrusive Appliance Load Monitoring (NIALM) System with the Possibility for Users to Follow the Consumption of Individual Electricity Appliances from the Calendar." *International Journal of Energy and Power Engineering* 5 (3): 129. https://doi.org/10.11648/j.ijepe.20160503.16.

Yu, Yong, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. 2019. "A Review of Recurrent Neural Networks: Lstm Cells and Network Architectures." *Neural Computation*. https://doi.org/10.1162/neco_a_01199.

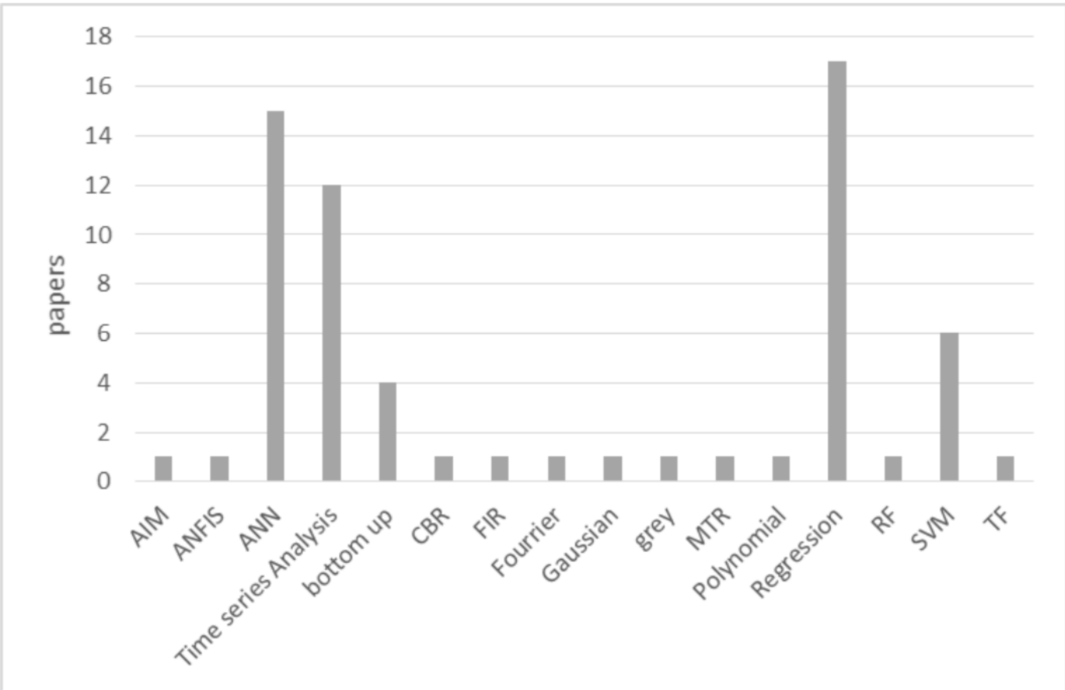
Zhou, Bin, Wentao Li, Ka Wing Chan, Yijia Cao, Yonghong Kuang, Xi Liu, and Xiong Wang. 2016. "Smart Home Energy Management Systems: Concept, Configurations, and Scheduling Strategies." *Renewable and Sustainable Energy Reviews*. https://doi.org/10.1016/j.rser.2016.03.047.

8. APPENDIX

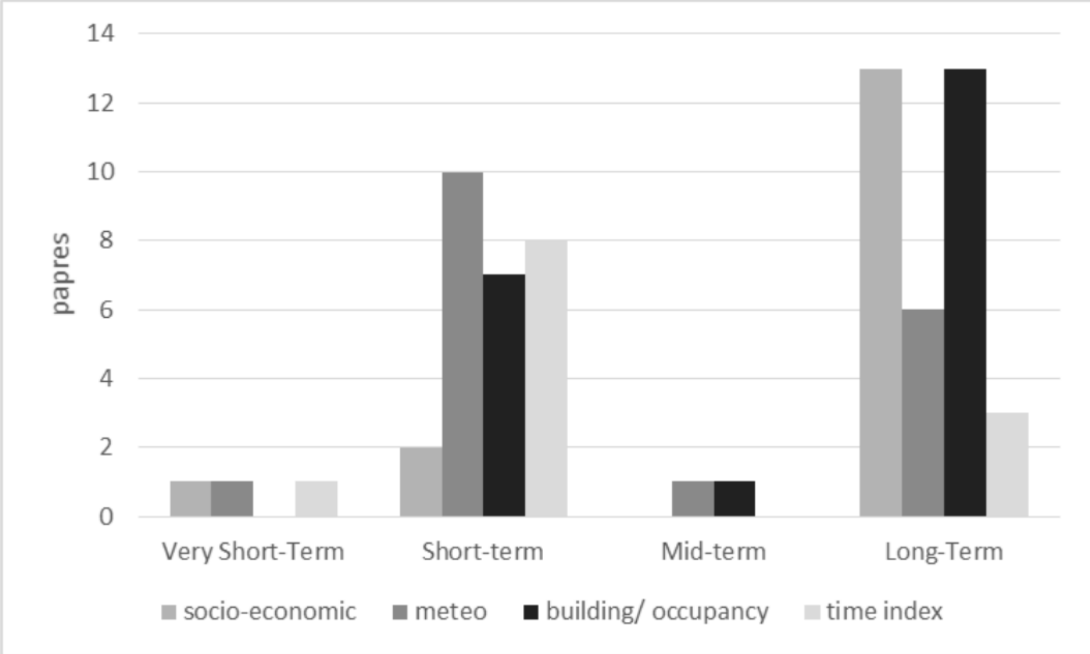
Appendix 1 *- Selection procedure presented. Source:* (Kuster, Rezgui, and Mourshed 2017).



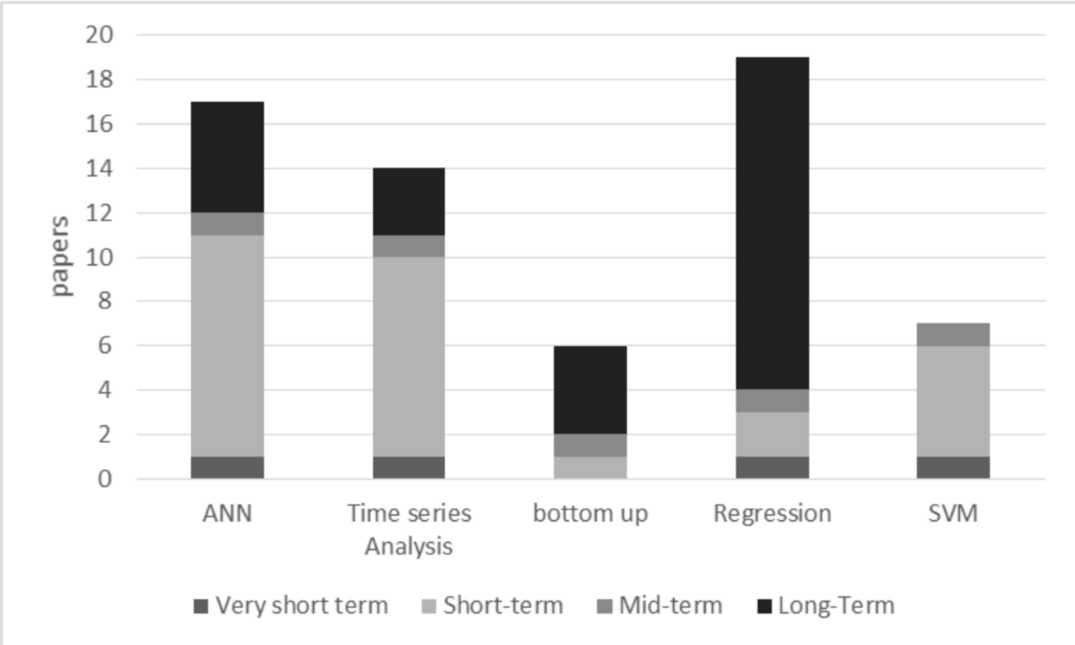
Appendix 2 *- Classified forecasting models distribution. Source:* (Kuster, Rezgui, and Mourshed 2017).



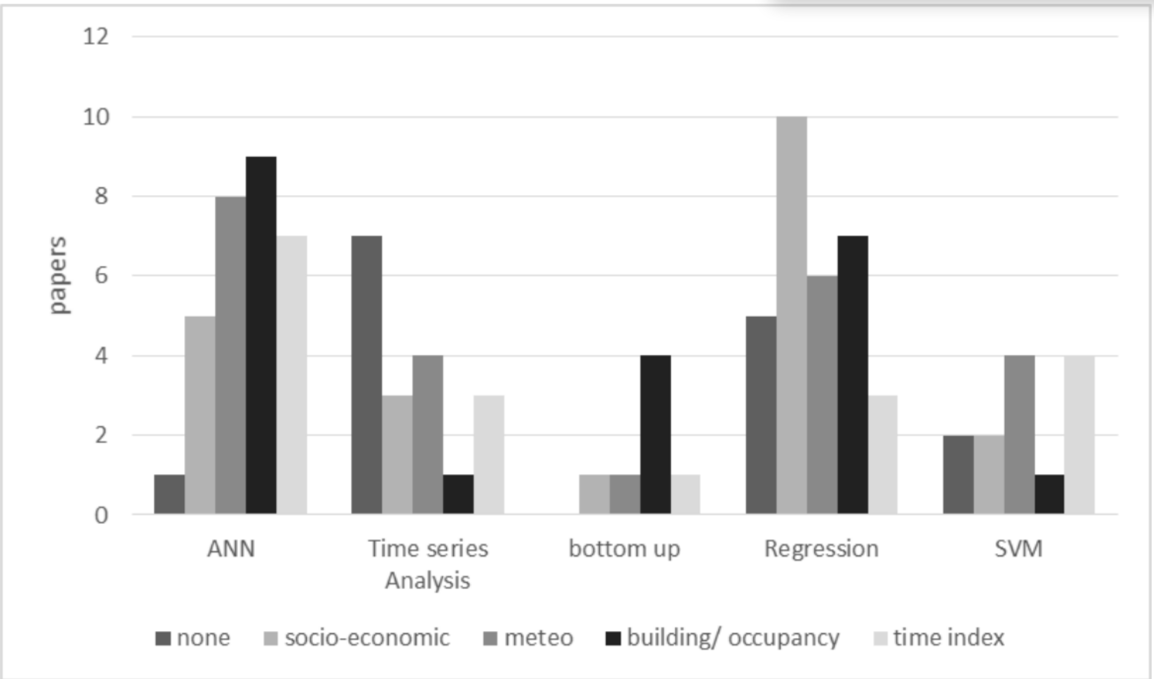
Appendix 3 - *Input distribution depending on the scale. Source:* (Kuster, Rezgui, and Mourshed 2017).



Appendix 4 – *Model distribution by resolution. Source:* (Kuster, Rezgui, and Mourshed 2017).



Appendix 5 *– Model vs inputs distribution. Source:* (Kuster, Rezgui, and Mourshed 2017).



Appendix 6 *– ConvLSTM parameters.*

Property	**Value**
Structure	One ConvLSTM2D layer and two dense layers
Filters	36
Kernel Size	(1,2)
Activation function	Relu
Nodes number of the first dense layer	4
Nodes number of the second dense layer	1
Epoch	20
Epochs testing Higher inputs	50
Batch size	1
Loss function	MSE
Optimizer	Adam
Epochs retraining	1
Input shape (steps, length, features)	(1,1,n)
Output shape (length, features)	(1,1)