

# Industrial Equipment Energy Efficiency Estimation and Performance Deviation Detection Using IoT Enabled Energy Metering

## **Author:**

Gabriel Stechschulte  
Kellerstrasse 38  
CH-6005 Luzern  
gabriel.stechschulte@hslu.ch

## **Lecturer:**

Dr. Braulio Barahona Garzon  
Lucerne School of Engineering and Architecture  
Technikumstrasse 21  
CH-6048 Horw  
braulio.barahonagarzon@hslu.ch

## **Co-lecturer:**

Antoine Finck  
CLEMAP  
Lavaterstrasse 66  
CH-8002 Zürich  
antoine@clemap.ch

## **Date:**

Spring Semester 2022  
Lucerne, 03 June 2022

A Master Thesis submitted as a requirement of the Master of Science in  
Applied Information and Data Science programme from Lucerne University  
of Applied Sciences and Arts

## **Management Summary**

In a production environment, monitoring the energy consumption of industrial machines is critical to assess the energy efficiency, productivity, and to ensure that the machine in operation does not result in high operating costs. Collaborating with an industrial partner offering metering hardware, power consumption data over a period of 10 days in a Swiss media and print company is collected and analyzed. Gaussian Processes, a Bayesian non-parametric learning algorithm, is used to model a time series of the energy consumption of the machines. Training a Gaussian Process model on a data set that represents the “best operating conditions” and or before any energy intervention process is applied yields an “energy baseline model”. Using this model, the next day energy consumption is predicted and compared against the actual values. Using statistical process control methodology, the posterior predictive distribution is used to monitor the degree of severity and uncertainty in instantaneous and cumulative changes in machine energy consumption. An example of this approach is presented for a paper disposal machine. Furthermore, to prepare the energy baseline model for deployment on the industrial partner’s infrastructure, a Docker container is created encapsulating the model’s parameters and inference phase.

# Contents

<b>List of Figures</b>	<b>5</b>
<b>List of Tables</b>	<b>6</b>
<b>List of Data Sources</b>	<b>7</b>
<b>List of Abbreviations</b>	<b>8</b>
<b>1 Introduction</b>	<b>11</b>
1.1 Motivation . . . . .	11
1.1.1 Industry Partner - CLEMAP . . . . .	11
1.1.2 Monitoring Energy Efficiency . . . . .	11
1.1.3 Machine Performance Deviation . . . . .	12
1.1.4 Switzerland Energy Policy . . . . .	13
1.2 Problem Statement . . . . .	13
1.3 Research Questions . . . . .	13
1.4 Project Tasks . . . . .	14
1.5 Organization of Thesis . . . . .	14
<b>2 Background and Related Work</b>	<b>16</b>
2.1 Overview of Energy Efficiency Estimation Methods . . . . .	16
2.2 Estimating and Monitoring Energy Efficiency in Industry . . . . .	16
2.3 Overview of Performance Deviation Detection Methods . . . . .	18
2.4 Performance Deviation Detection Methods in Industry . . . . .	19
<b>3 Methodology and Evaluation Metrics</b>	<b>21</b>
3.1 Definition and Characteristics of Time Series . . . . .	21

3.2	Introduction to Gaussian Processes for Time Series . . . . .	22
3.3	Covariance and Mean Functions . . . . .	24
3.3.1	Squared Exponential Kernel . . . . .	24
3.3.2	Periodic Kernel . . . . .	25
3.3.3	Rational Quadratic Kernel . . . . .	26
3.3.4	Kernel Composition . . . . .	26
3.3.5	Locally Periodic Kernel . . . . .	28
3.4	Advantages and Disadvantages of Gaussian Processes . . . . .	28
3.5	Evaluation Metrics . . . . .	29
<b>4</b>	<b>Datasets</b>	<b>32</b>
4.1	CLEMAP Client Dataset . . . . .	32
4.1.1	CLEMAP Exploratory Data Analysis . . . . .	33
4.2	HIPE Industrial Energy . . . . .	38
4.3	Similarity and Differences of Data Sources . . . . .	39
<b>5</b>	<b>Results</b>	<b>40</b>
5.1	Experimentation Setup . . . . .	40
5.2	Kernel Design and Composition . . . . .	41
5.3	Kernel Design Decomposition . . . . .	42
5.4	Energy Baseline Model Experiment Performance Metrics . . . . .	44
5.5	Assumptions and Limitations of Models . . . . .	44
<b>6</b>	<b>Monitoring Energy Efficiency and Performance Deviation Detection using the Baseline Model</b>	<b>46</b>
6.1	Evaluating Machine Energy Efficiency and Performance using Control Charts	46

<b>7 Model Deployment Preparation</b>	<b>49</b>
7.1 Docker for Machine Learning Deployment . . . . .	50
7.2 Developing a Docker Container . . . . .	51
<b>8 Conclusion</b>	<b>52</b>
8.1 Summary of Main Results . . . . .	52
8.2 Discussion with CLEMAP's Client . . . . .	52
8.3 Reflection on Research Questions . . . . .	53
8.4 Next Steps . . . . .	54
<b>9 Bibliography</b>	<b>56</b>
<b>A Appendix</b>	<b>59</b>
<b>B Appendix</b>	<b>60</b>
<b>C Appendix</b>	<b>61</b>

## List of Figures

1	Three samples, each from an <i>RBF</i> kernel with different hyperparameters denoted by the table legend. Notice how $\ell$ affects the smoothness of the function.	25
2	Three samples, each from a <i>Per</i> kernel with different hyperparameters denoted by the table legend. Notice how $p$ affects the periodicity of the function and $\ell$ affects the “wiggliness” of the variations.	26
3	Entsorgung ACF. The time series is discretized and averaged at 1 hour intervals. Therefore, the ACF shows significant autocorrelations between 10 and 12 hours and 22 and 24 hours with the remaining lags decreasing in amplitude and significance.	27
4	Three samples, each from a <i>RQ</i> kernel with different hyperparameters denoted by the table legend. Notice how $\alpha$ affects the local variations of the function while $\ell$ affects the “wiggliness” of the variations.	27
5	Three samples, each from a <i>LocPer</i> kernel with different hyperparameters denoted by the table legend. Notice how the product of the two kernels results in a periodic structure with variations in each cycle.	29
6	CLEMAP sensor measurement setup. Hourly load energy consumption (kWh) for each machine over the course of 10 days.	33
7	Paper disposal machine	34
8	Paper disposal hourly load profile. The color bar on the right indicates the average kW consumed for that hour over the 10 day period.	35
9	Entire time series for the paper disposal machine.	35
10	(a) Production week time series at 10 minute aggregations. (b) ACF correlogram indicates significant periodic cycles at about 12 and 24 hours, respectively. For example, $(75 * 10)/60 = 12.5$	36
11	(a) Random profile of non-operational hours from 17:30 - 05:00 (b) Random peak load profile from 08:00 - 13:00	37
12	Experimentation workflow for developing and refining GP energy baseline models.	41
13	Cumulatively adding more complex kernels. The dashed line represents the training and validation split. (a) $K_{LocPer24}$ , (b) $K_{LocPer12}$ , and (c) $K_{LocPer24} + K_{LocPer12} + K_{RQ}$	43

14	Benchmark reporting (training) period SPC charts. . . . .	47
15	Using the test data, the paper disposal machine SPC charts are visualized. The top plot is the instantaneous residuals whereas the bottom is the CuSum of residuals. . . . .	48
16	Original time series plotted for further analysis after the SPC charts and GP model identified a significant decrease in energy consumption indicated by the red shaded area. . . . .	49
17	Historical registry of paper disposal machine performance deviations. . . . .	49
18	Hypothetical “MLOps” deployment process. Figure taken from [29]. . . . .	50
19	Interpolation and extrapolation for the main terminal GP model at a 30 minute time aggregation. The demand for energy during the afternoon on Friday tapered off much quicker than what the baseline model had predicted and thus is the reason for the poor evaluation metrics. . . . .	59
20	10 minute frequency of the R707LV printer displays some periodicity but is not identifiable by the ACF correlogram and <i>Per</i> kernel. . . . .	59

## List of Tables

1	List of items being metered by CLEMAP. A system may be composed of several machines, all of which are being metered. . . . .	32
2	List of items being metered by KIT at IPE and their purpose. . . . .	38
3	GP evaluation metrics for each machine and time aggregation. The learning rate was set to 0.1, and the optimization loop consisted of 100 iterations. The metrics in the table represent the lowest and or highest score achieved during the experimentation phase. . . . .	44
4	Branches and directories of the GitHub repository developed for the thesis. .	60

## List of Data Sources

Category	Source	Accessed
HIPE Industrial Production Energy	Karlsruhe Institute of Technology	27.10.2021
CLEMAP Swiss Media and Printing Production Energy	CLEMAP	01.12.2022

## List of Abbreviations

**A** ampere

**ACE** Average Coverage Error

**ACF** autocorrelation function

**ADAM** adaptive moment estimation

**API** application programming interface

**BBMM** Blackbox Matrix-Matrix multiplication

**CAS** compressed air systems

**CI** confidence interval

**CI/CD** Continuous integration and delivery

**CLEMAP** Clever Energy Mapping

**CO<sub>2</sub>** carbon dioxide

**CuSum** cumulative sum

**EDA** exploratory data analysis

**EE** energy efficiency

**EEE** energy efficiency estimation

**EEM** energy efficiency intervention measure

**ETL** extract transform and load

**ETS** swiss emissions trading scheme

**EU** European Union

**EWMA** exponentially weighted moving average

**GAM** Generalized Additive Model

**GHG** greenhouse gases

**GMM** Gaussian Mixture Model

**GPU** graphical process unit

**GPs** Gaussian Processes

**HYPE** High Resolution Industrial Production Energy

**HVAC** heating ventilation and air-conditioning

**Hz** hertz

*I* Current

**IEA** International Energy Agency

**IP** internet protocol

**IPE** Institute of Data Processing and Electronics

**ISO** International Organization for Standardization

**IoT** Internet of Things

**KIT** Karlsruhe Institute of Technology

**KPIs** key performance indicators

**LCL** lower control limit

*LocPer* locally periodic

**MA** moving average

**MAPE** Mean Absolute Percentage Error

**MI** mutual information

**MLR** multiple linear regression

**MSE** Mean Squared Error

**MV** measurement and verification

**MW** Megawatts

**NNs** neural networks

*P* active power

**PDD** performance deviation detection

**PI** prediction interval

*Per* periodic kernel

**QC** quality control

*RBF* radial basis function

**RMSE** root mean squared error

**RO** rolling outlier

*RQ* rational quadratic

**SPC** statistical process control

**STL** seasonal-trend-loess

**SVM** support vector machine

**UCL** upper control limit

*V* Voltage

*S* apparent power

*Q* reactive power

**VAV** variable air volume

**VM** virtual machine

*W* Watts

**WEC** World Energy Council

**a.c** alternating current

**kW** kilowatts

**kWh** Kilowatthour

**lr** learning rates

*PF* power factor

**r-squared** coefficient of determination

**sq.ft** square foot

# 1 Introduction

## 1.1 Motivation

### 1.1.1 Industry Partner - CLEMAP

This thesis is in collaboration with Clever Energy Mapping (CLEMAP)<sup>1</sup>, a company offering hardware in the form of energy metering sensors for a wide range of electrical appliances and equipment. Complementing their hardware, CLEMAP also offers software products in the form of a data management platform and a software application to provide their customers with the metering data measured by their sensors to act as a basis for analysis, and support and device management.

CLEMAP's sensors belong to Internet of Things (IoT), an emerging technology consisting of sensors embedded in physical objects that are linked through wired and wireless networks (routers), often using the internet protocol (IP), e.g. using an IP address. In Industry 4.0, these IoT devices are enhancing the metering and sub-metering capabilities of industrial equipment and production processes by collecting not only granular, multi-measurement historical data sets, but also streaming data not previously available for real-time analysis [1]. CLEMAP's meters have the ability to collect a range of information such as Voltage ( $V$ ), Current ( $I$ ) in ampere ( $A$ ) (40A to 6kA), active power ( $P$ ), power factor ( $PF$ ), reactive power ( $Q$ ), and apparent power ( $S$ ) on each phase of a three phase load at a frequency of 12 hertz (Hz).

With the advancements in sensor technology and increasing availability of large quantities of power-usage and energy consumption data from industrial equipment and commercial production, CLEMAP is looking to expand upon their portfolio of software based services, utilizing IoT, into energy efficiency estimation (EEE) based tools that analyze and monitor energy consumption and identify equipment or processes where energy is “wasted” and or “deviating” from their expected behavior.

### 1.1.2 Monitoring Energy Efficiency

First and foremost, what is energy efficiency (EE)? The terminology of EE is broad and definitions can differ from organization to organization and by economic sectors. The International Organization for Standardization (ISO) defines energy efficiency as a “ratio or other quantitative relationship between an output of performance, service, goods or energy, and an input of energy” [2] while the International Energy Agency (IEA) and the World Energy Council (WEC) defines energy efficiency as a “reduction in the energy input of a given service or level of economic activity” [3]. In the ISO definition, EE is a quantitative

---

<sup>1</sup><https://en.clemap.ch/>

measure between an output and input *as is*. Subsequently, the IEA and WEC defines EE as a quantitative measure *given* a reduction in the energy input.

In the organization definitions stated above, EE involves analyzing inputs and outputs which can be difficult in practice as it is not always clear what is defined or measured as an output. As well, smart meters are only logging the energy inputs that the physical device consumes. In this thesis, EE is defined as the amount of energy the equipment or production process consumes *as is* for a given time period with respect to a baseline. It is important to note that in this work, the output of a machine is unknown.

Energy managers cite a number of benefits attributable to energy metering and monitoring. Fundamentally, they stem from the “you cannot manage what you do not measure” adage [4]. Building off of this adage, in regard to industry 4.0 and CLEMAP’s ambitions, the motivation is to develop an energy consumption baseline for a subset of machines being monitored by CLEMAP’s sensors. Being able to accurately model the underlying physical process of a machine is critical to assessing EE and productivity metrics. The development of a model prior to any energy intervention project, new production process or machine components is called an energy baseline model as it establishes an energy consumption baseline / benchmark *as is*.

With the energy baseline model, at a certain time point, a machine’s energy consumption is predicted and then compared to the actual measured value. The difference between the prediction and actual value can be monitored and analyzed over time. Thus, creating value add in assuring the machines that are in production do not cause excessive negative externalities such as high operating costs [5], and are operating nominally. Additionally, a well established benchmark can help companies measure and improve their performance not only within the organization, but also to compare with benchmarks established by similar firms in their industry.

### 1.1.3 Machine Performance Deviation

Subsequently, in industry, once you have established a benchmark, and as a production manager or engineer wanting to employ better energy management practices, you want to know if a piece of machinery is deviating away from the benchmark. Complementing the energy baseline model, performance deviation detection (PDD) can provide such information with the production manager’s domain expertise by detecting if a machine is in an abnormal state, that is, consuming more or less energy than what was expected—compared to the benchmark. Not only can this provide insights into emerging patterns of excessive energy consumption, but it can also provide insights into machine degradation or failure [6]. The economic aspect in the context of the equipment life cycle, for just maintenance and support for machine tools, accounts for 60 to 75 percent of the total life cycle cost [7]. Therefore, PDD can complement the energy baseline model throughout the entire machine life cycle.

#### **1.1.4 Switzerland Energy Policy**

In the Swiss industrial sector, large energy consuming and greenhouse gases (GHG) intensive companies are required to participate in the swiss emissions trading scheme (ETS), an accounting system to ensure companies have complied with statutory obligations and to auction off emissions. This system was introduced in 2008 and merged with the European Union (EU) ETS in 2020, along with a carbon dioxide (CO<sub>2</sub>) levy in order to curb GHG [8]. Larger companies which are not regulated by the ETS can enter into an agreement with the Federal Office, the canton, or third-party government mandated energy agencies to commit themselves to reduce GHG emissions; ultimately allowing them to be exempted from the CO<sub>2</sub> levy and to obtain full or partial refund of the renewable energy network surcharge [9]. Demand side policy is motivating businesses to employ better energy management practices to not only reduce CO<sub>2</sub> emissions, but to also reduce operating costs in the short and long run.

## **1.2 Problem Statement**

Per Section 1.2, the development of an energy baseline model forms the foundation for monitoring machine energy efficiency and for identifying when a machine is deviating from nominal behavior. Using data measured with CLEMAP's smart energy meters, how can an energy baseline model be utilized to monitor energy efficiency over time and also detect deviations in machine performance?

## **1.3 Research Questions**

In a study leading up to this thesis, a list of plausible research questions were outlined. However, conclusions for every research question were not reached due to time constraints and or focusing on tasks of higher priority. Therefore, below, a list of the relevant research questions to this report are outlined.

1. Smart energy meters only provide information in regard to inputs used by a piece of equipment and without access to “output” data from the sensors, how can a baseline energy efficiency or benchmark be estimated from energy data alone? Furthermore, how frequently should this benchmark be updated?
2. Could additional data be added or engineered to enhance the solution to research question one such as production data, financial information, temporal aggregations, maintenance logs or weather data?
3. Upon addressing research question one, can performance deviation detection methods, on top of energy efficiency estimation, be implemented to ensure benefits to that particular piece of equipment in the production process life cycle?

4. What information from the results of the algorithms, hypothetically, needs to be transmitted back to the production manager or equipment operator in addressing questions one and three so they can make informed production actions?

In Appendix C, a full list of the research questions identified in the preliminary study are outlined. Furthermore, in Section 8, a reflection on the full list of research questions is given.

## 1.4 Project Tasks

To address the problem statement and research questions stated above, the following main tasks are outlined:

- Identify equipment being metered by CLEMAP where an energy baseline model can be applied.
- With the measurement data, perform a series of experiments to model the equipment from the bullet point above, using a hypothetical reporting period and perform predictions one day ahead.
- Assess the quality of the model using deterministic and probabilistic evaluation metrics and select the best performing model according to those metrics—this model is then the energy baseline model.
- Use statistical process control (SPC) charts to monitor EE over time, and to visualize and detect periods of deviations in performance.
- To prepare for a prototype deployment on CLEMAP’s infrastructure and environment, create a docker container of the trained energy baseline models.

## 1.5 Organization of Thesis

With the introduction, motivation, problem statement, and research questions now set, the remainder of the thesis is organized as follows: Section 2 gives an overview and application of methods pertaining to EEE and PDD in industry. In Section 3, building off of the related work, the methodology used in this thesis is presented along with the evaluation metrics to assess the quality of the model. Furthermore, advantages and disadvantages of the proposed methodology are given in Section 3.4. Then, in Section 4, two data sets are introduced where an example of an exploratory data analysis (EDA) workflow is given for a particular machine within the CLEMAP data set. Also, the similarity and differences between the open source and CLEMAP data is given. With the data set and proposed methodology now given, Section 5 contains results and a decomposition of the energy baseline model is described. Subsequently, in Section 6, monitoring EE and performing PDD using the energy baseline

model is explained using a paper disposal machine. In Section 7, an overview of machine learning deployment and operations is given. In this overview, container technologies are discussed, namely Docker, and a Docker container is developed for the Gaussian Process models. Lastly, Section 8, provides a summary of the main results, answers to the research questions, and comparison and next steps of the current research is outlined.

## 2 Background and Related Work

### 2.1 Overview of Energy Efficiency Estimation Methods

Simona et al. [10] provides a framework for the development of a comprehensive and cost-effective energy management solution in industrial plants. The framework uses continuous monitoring and comparison of an industrial *processes* expected behavior to its actual behavior. Indeed, in the literature and in industry, many practitioners monitoring and or estimating the energy efficiency of buildings or industrial equipment often follow the same framework and is enumerated below:

1. Collection of metered data.
2. A baseline / benchmark model and reporting period is defined.
3. Continuous monitoring and comparison of baseline consumption to actual consumption.

First, before any analysis can take place, a metering solution must be in place to collect the desired measurement data. Secondly, in step two, the reporting period is defined as the EE *as is* and provides a baseline from which to measure, analyze, test and improve [11]. A model, trained on the reporting period constitutes as the baseline, or benchmark, defined as the energy consumption that characterizes the starting situation of the processes. Its role is fundamental in the assessment of EE and other key performance indicators (KPIs). Often, a baseline model in the form of a least squares multiple regression model, and more recently, additional methods such as neural networks (NNs) or support vector machine (SVM) have been used [12]. However, the studies below show that multiple linear regression (MLR) seems to still be the choice of model used in industry, and in particular, when additional independent variables are correlated with the EE. Lastly, step three, involves analyzing the difference of the energy baseline model's predicted energy consumption to the actual consumption. Step three is discussed below in Section 2.3 in more detail.

### 2.2 Estimating and Monitoring Energy Efficiency in Industry

In the literature, there is a vast amount of research and applications on electrical load forecasting. However, given our definition of EE and goal of developing an energy baseline model, the focus in this thesis is on operationalizing the predictions to monitor EE. That is, instead of simply communicating forecasted energy consumption, the aim is to also communicate a machine's EE and deviation in performance. In industry, this is often the difference between actual energy consumption versus the baseline model predictions. Namely, at predefined time points in the future,  $x_{t+n}$ , the equipment's input is predicted using the baseline model

and is then compared to the actual result when the process is measured at that time step  $x_{t+n}$  [13].

Trager et.al [13] used SPC, a technique often used in manufacturing in order to determine whether a process is changing based upon recent measurements, to monitor and analyze deviations of the energy consumption of two commercial buildings. Building one is a 100,000 square foot (sq.ft) 1990's vintage office building in California. This building is a high tech sector three-story office building with significant lab loads. The building is primarily variable air volume (VAV) reheat system, with additional cooling via fan coils serving high load areas. Building two is a 200,000 sq.ft 1980's vintage office building near Chicago, Illinois. The 12-story all-electric office building is mostly tenant occupied office space with a detached non-enclosed three-level parking garage. Using six months of data gathered from the two commercial buildings, an seasonal-trend-loess (STL) regression model was developed to predict one hour ahead for each time step. The entire prediction is then moved ahead one hour, so that at each time step in the procedure, one hour of data is predicted. The six months of training data acts as the baseline reporting period in which future predicted energy consumption is analyzed against. Then, using the residuals—the difference between the predictions and actual values—the authors use control charts to monitor EE and analyze deviations in building consumption which is talked about more in Section 2.4. The STL model, had Mean Absolute Percentage Error (MAPE) values of 4.94% and 19.2% for building one and two, respectively, when predicting one hour ahead.

Benedetti et al. [12] in a case study with a pharmaceutical manufacturing plant located in central Italy demonstrated the importance of monitoring and controlling energy performance in compressed air systems (CAS) using novel methodology based on meter data and a energy baseline definition using MLR and control charts (see Section 2.4). The plant was equipped with five screw compressors divided into two groups installed in separate sections of the plant. Group one hosted two compressors, whilst group two hosted the remaining three compressors. To model the baseline energy performance of the system, a baseline reporting period of one year at 15 minute intervals taking into account other independent variables such as compressed air production, external temperature, external humidity and pressure is chosen which displays the *best* operating performance of the CAS. Notably, when adding compressed air production as a covariate, the MLR has an coefficient of determination ( $r^2$ ) of 0.934%. Then, using this reporting period, a regression model is developed which acts as the energy baseline model. In order to monitor the energy performance, the authors evaluate the residual between actual energy consumption and the prediction of the baseline model.

Granderson et al. [14] used two months of electrical load data (August-September) from a small office building complemented with outdoor air temperature to train an energy baseline-modeling agent. Then, the model was used to predict the load for the next two months (October-November). For the first three weeks, the root mean squared error (RMSE) of the hourly load is 1.6 kilowatts (kW). Then, in the subsequent weeks, the RMSE doubled to 3.2 kW. Investigation into this new behavior showed that the building had switched into a “heating mode” at the end of October. The building manager had stated this should not happen due to the warmer weather, and as a result, a new heating ventilation and air-

conditioning (HVAC) control change was implemented to deter these type of events from happening.

Nikula et al.[15] compared actual boiler efficiency in power stations with its expected efficiency which is an estimate of the highest historical efficiency in the corresponding process state using an identification period based on MLR. The process state is a particular state which is defined as a function of the variables selected using mutual information (MI). Two industrial boilers were analyzed with a capacity of  $300 \text{ MW}_{th}$  and  $160 \text{ MW}_{elec}$ , and  $220 \text{ MW}_{th}$ , respectively.  $\text{MW}_{th}$  indicates the power is from thermal energy while  $\text{MW}_{elec}$  indicates the power is from electricity. Each boiler's time series was sampled at an interval of 1hr. Boiler one included 143 variables, trained on 810 hours of data, and was tested (monitored) using 70 hours. Boiler two included 125 variables, trained on 72.5 hours, monitored using 3.5 hours. Using MI, the top two features were chosen and were discretized. The step size in the discretized feature represents a process state. Therefore, in the MLR model, the dependent variable represents the expected efficiency and the features step sizes define the process states. The difference between the actual and expected efficiency was monitored using control controls which is talked about in Section 2.4 and the two features chosen using MI were analyzed as a potential root cause of deviations.

Benedetto et al [16], in another application, using metered data from three commercial buildings, complemented with outdoor weather data and binary indicators of holidays, developed an energy baseline model to compare the energy consumption before and after an energy efficiency intervention measure (EEM). Daily load profiles were first identified through the means of a Gaussian Mixture Model (GMM). Then, a Generalized Additive Model (GAM), using the load profiles of the previous seven days, outdoor temperature, sun altitude, wind speed, and a holiday flagging variable is used to model the electricity consumption. This baseline model is then used to compute the counterfactual energy consumption, i.e., the predicted energy consumption if no EEM had taken place. Using the proposed methodologies, upon the EEM in the commercial building, the model achieved an RMSE of 13%, 9.1%, and 7.8%, and predicted a decrease in consumption Kilowatthour (kWh) of 8.6%, 10.9%, 8.3% for building 1, 2, and 3, respectively.

### 2.3 Overview of Performance Deviation Detection Methods

In the previous section, [13][12][15], had used a form of charting, built off of the baseline model, to identify when a machine or building was deviating from benchmark efficiency. In industry, this charting is often called SPC, or quality control (QC), and monitors the residuals from the baseline model to measure the statistical significance of the actual energy use deviating from the predicted energy use. Thus, using SPC charts, operators and or production managers of the equipment can gain insight into EE over time and near real-time current deviations in performance [11].

However, as the field of PDD is vast, it must be acknowledged that analyzing the residuals from regression models is not the only way to perform PDD. Rather, it is a natural extension

of time series regression models. Other techniques, primarily using classification and clustering methods, have been proposed and used to perform PDD which will be briefly described here.

Zhang et al. [17] used GMM techniques to provide recognition of operational states, extraction of load profiles, and to detect emerging faults for industrial turbines. To detect emerging faults, a normal operation state was recognized using a Bayesian GMM technique. Then, with this normal operation cluster a GMM was used to define an ellipse boundary to the cluster and when a new data point fell outside of the ellipse boundary it was identified as a fault. This method, which is now in production, allows the production managers and engineers to analyze, in real time, the performance of their equipment. Gallagher et al. [18] implemented a clustering based PDD in the framework of measurement and verification (MV) to maintain energy savings. Du et al. [19] combined NNs to detect abnormalities in the energy consumption of the HVAC system and used subtractive clustering to classify the abnormalities for diagnostic purposes.

## 2.4 Performance Deviation Detection Methods in Industry

As the main motivation in this thesis is centered around the idea of an energy baseline model, below, PDD based on the use of SPC is demonstrated.

Using the residuals from the STL model in [13], three control charts—a moving average (MA), rolling outlier (RO), and a dual measure control chart combining the metrics of the MA and RO charts—are used. The MA chart assumes that the mean of the process should be stationary, and computes the mean of the prediction process residual as a two week rolling average (336 hours). Using this method, an upper control limit (UCL) and lower control limit (LCL) are calculated to determine when a process is “out of control” and is defined as:

$$UCL = \mu + \frac{3\sigma}{\sqrt{n * w}} \quad (1)$$

$$LCL = \mu - \frac{3\sigma}{\sqrt{n * w}} \quad (2)$$

where  $\sigma$  is the standard deviation and  $n$  is the average of the process within the rolling window  $w$  where  $w = 336$ . Points lying outside of the UCL or LCL indicate the process is deviating from the expected behavior. The RO chart calculates the 99% percentile of residuals in the preceding week, with a point being labeled “hard to predict” if the residual lies in the 99%. Combining the RO and MA chart allows one to identify a change in mean of the MA chart and corresponding analysis into which point(s) caused the change indicated by the RO chart. This charting methodology flagged a problem with the HVAC system of the building, and upon inspection, the engineer identified excessive overnight cycling of the HVAC system—resulting in unnecessary energy consumption and equipment usage.

Building off of the MLR model developed in the case study with the pharmaceutical manufacturing plant [12], control charts were developed to identify the EE of the CAS over time. In this study, the authors used instantaneous and the cumulative sum (CuSum) of residuals. Instantaneous and the CuSum of residuals is given by,

$$\text{Instantaneous} = \Delta E(t) = E_{act}(t) - E_{pred}(t) \quad (3)$$

$$\text{CuSum} = \Delta EC(t) = \sum_{t=0}^t \Delta E(t) \quad (4)$$

where  $E_{act}(t)$  is the actual energy consumption of the system at time  $t$  and  $E_{pred}(t)$  is the predicted energy consumption at time  $t$ . The difference between these two is the residual, whereas the CuSum is the cumulative sum of residuals from  $t = 0$  to  $t = n - 1$ . In addition, UCL and LCL were also defined similar to [13] to determine the significance of the CAS deviations. However, in this case study, the control limits were defined by  $2\sigma * \Delta E(t)$ .

Using these two control charts, the authors identified three different operating conditions over a time period of one year. The charts were then reviewed by the operators of the CAS where the first two were related to a maintenance and process intervention; respectively. Lastly, the third operating condition represented an evident malfunctioning of a compressor that remained stuck in stand-by for three consecutive days.

The control charts utilized by [15] were developed off of the exponentially weighted moving average (EWMA) statistic which gives less weight to older observations and provides an indication of the direction of the boilers performance and the symptoms of possible malfunctions and is defined by:

$$y_t = y_{t-1} + \lambda e_t \quad (5)$$

where  $y_t$  is the EWMA at time  $t$ ,  $\lambda$  is a constant between 0 and 1 which determines the memory length,  $e_t$  is the observed change  $\hat{y}_t - y_{t-1}$  and  $\hat{y}_t$  is the observed MA of the sample at time  $t$ . Subsequently, UCL and LCL were developed using the EWMA statistic to identity the significance in deviations of efficiency:

$$T \pm \sqrt{\left(\frac{\lambda}{2-\lambda}\right)\sigma} \quad (6)$$

where  $T$  is the target value and represents the mean in the process history and  $\sigma$  is the standard deviation in the process history. The baseline model, complemented with the EWMA control chart, identified a series of “out of control” points that signaled the corner-fired boiler was operating below its expected efficiency. As a root cause analysis, the variables used in the MLR and time period leading up to and after the out of control points were analyzed. In conclusion, it was found that the independent variables were exhibiting more variance than in the baseline period used to train the model.

## 3 Methodology and Evaluation Metrics

### 3.1 Definition and Characteristics of Time Series

A time series is a sequential set of data points indexed by time  $t$  with some output  $y$ . It is typically defined as a set of vectors  $y(t), t = 0, 1, 2, \dots, n$  where  $t$  represents the time elapsed and  $y$  is the output. A time series is *univariate* if it consists of single variable over equal periods of time. For example, monthly CO<sub>2</sub> measurements is a *univariate* time series. Furthermore, more variables may be added to a time series to make it *multivariate*. A *multivariate* time series consists of multiple time-dependent variables where each variable may also have some dependency on other the variables. Time series can be modelled via discrete or continuous time series models. Continuous time series are recorded at every instance of time, even when the measured variable can only take a discrete set of values (company sales, number of unemployed persons). Discrete time series are when observations are taken only at specific times, usually equally spaced, even if the measured variable is a continuous variable. In this thesis, a continuous time series is discretized by aggregating the time series into equally spaced intervals of 10 and 30 minutes upon which the continuous variables Watts ( $W$ ),  $V$ , and  $I$  are analyzed.

Within a time series, there are typically four components to be considered which may influence the analysis or modeling. Not all time series contain all four components, and certain models are aimed at modeling time series with only a subset of these components. Thus, the EDA and subsequent model selection are important phases of time series analysis. Below, the main components of a time series are defined.

**Trend:** A trend is a general tendency of a time series to increase, decrease or stagnate over time [20]. For example, global warming, a time series referring to the global temperature, exhibits an increasing trend.

**Seasonal:** Seasonality is when a time series is influenced by seasonal factors (e.g., the quarter of the year, the month, or day of the week). Time series longer in duration may have multiple seasonality components. For example, airline data on the number of passengers flying exhibits multiple seasonality—vacation and holiday periods [20].

**Cyclical:** The cyclical variation in a time series describes data that exhibits rises and falls that are *not of fixed period*. The duration of a cycle depends on the type of industry or business being analyzed. For example, electrical load profiles of residential buildings often exhibit a cyclical pattern as the demand for energy *cycles* throughout the day, but is typically not of a fixed period, i.e., people continue living in the building for a long period of time.

**Irregular:** This component is unpredictable. Every time series has some unpredictable component that makes it a random variable. In prediction, the objective is to “model” all the components to the point that the only component that remains unexplained is the random component. This component is also sometimes called the *residual* [20].

In this thesis, it is expected that the industrial equipment will show a strong cyclical pattern as the loads of the equipment will be influenced by production process schedules; in turn influenced by the “business week”. In Section 4.1.1, an exploratory analysis will be conducted to visualize the components of the machine’s time series.

## 3.2 Introduction to Gaussian Processes for Time Series

Gaussian Processes (GPs) belong to the family of *Bayesian non-parametric models* and offer a principled, interpretable, and intuitively specified way for conducting probabilistic inference of non-linear time series. Non-parametric methods do not assume a fixed parametric form for the prediction function, but instead try to estimate the function itself (rather than the parameters) directly from the data.

GPs can be viewed as a generalization of a Gaussian distribution in  $\mathbb{R}^n$  to a space of functions. More specifically, a GP is a way to define distributions over functions with the assumption that the function values at a set of  $M > 0$  inputs,  $f = [f(x_1), \dots, f(x_M)]$ , is jointly Gaussian with a mean ( $\mu = m(x_1), \dots, m(x_M)$ ) and covariance  $\sum_{i,j} = K(x_i, x_j)$ , where  $m$  is the mean function and  $K$  is a positive definite kernel [21]. A positive definite kernel is a generalization of a positive definite matrix where the matrix is symmetric and all its eigenvalues are positive.

Thus, the two central components of a GP are the mean  $m(x)$  and covariance kernel function  $k(x_i, x_j)$ . The former represents the value we expect for our function before seeing the data, and the latter, the beating heart of a GP, specifies the correlation between any pair of outputs and therefore determines the properties of the function that it generates. The kernel allows one to encode prior knowledge about the similarity of the two input vectors  $x_i$  and  $x_j$ , i.e., if we know that  $x_i$  is similar to  $x_j$ , then the model can be encouraged to make the predicted output at both locations  $f(x_i)$  and  $f(x_j)$  to be similar. As the problem in this thesis is concerned with time series, the informativeness of past observations in explaining current data is a function of how long ago the past observations were observed [22]. The covariance matrix for a set of locations  $x = \{x_1, x_2, \dots, x_n\}$  is defined as:

$$K(x, x') = \begin{pmatrix} k(x_1, x_1) & k(x_1, x_2) & \dots & k(x_1, x_n) \\ k(x_2, x_1) & k(x_2, x_2) & \dots & k(x_2, x_n) \\ \vdots & \vdots & \vdots & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & \dots & k(x_n, x_n) \end{pmatrix} \quad (7)$$

This means that the entire function evaluation, associated with points in  $x$ , is drawn from a multivariate Gaussian distribution [22]:

$$p(y(x)) \sim \mathcal{N}(m(x), K(x, x')) \quad (8)$$

where  $y = \{y_1, y_2, \dots, y_n\}$  are the dependent function values, evaluated at locations  $x_1, \dots, x_n$ ,

$m$  is a *mean function* and  $K$  is a kernel function, again evaluated at the locations of the  $x$  variables. If one believes there is noise (which there often is) associated with the observed function values  $y_i$ , then a noise term can be directly incorporated into the covariance function:

$$K(x, x') + \sigma^2 I \quad (9)$$

where  $I$  is the identity matrix since noise is expected to be uncorrelated from sample to sample; hence noise only needs to be added to the diagonal of  $K$  [22]. The  $\sigma^2$  is a hyperparameter representing the noise variance.

A GP with a given *mean function*  $m$  and *covariance function*  $K$  is a prior that can generate functions; hence the common definition of a GP as a “prior distribution over functions”. Now that the central components of a GP have been laid out, the following describes how to perform inference. To perform inference with GPs from some input (test data)  $X_*$  given some training data  $D = \{(x_i, y_i)\}$  assuming i.i.d additive Gaussian noise with variance  $\sigma_n^2$ , the prediction and observations are expressed as a joint distribution following the GP prior [22]:

$$\begin{bmatrix} y \\ y_* \end{bmatrix} \sim \mathcal{N}(m, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}) \quad (10)$$

where  $y_*$  is some output test data and  $X_*$  is the input test points. One can think of the evaluation of test points  $X_*$  at new locations as *augmenting* the observed data  $X$  with the test data  $X_*$  and  $y_*$  in the joint distribution.

With the product rule, one can relate the joint distribution to the conditional distribution to obtain the posterior predictive distribution [21]:

$$p(f_*|X_*, X, y) = \mathcal{N}(f_*|m_*, \sum_*) \quad (11)$$

with mean and covariance given by:

$$m^* = m(x_*) + K(X_*, x)(K(X, X) + \sigma_n^2)^{-1}(y - m(x)) \quad (12)$$

$$\sum_* = K(X_*, X_*) - K(X_*, x)[K(X, X) + \sigma_n^2]^{-1}K(X, X_*) \quad (13)$$

Gaussian Process models have a number of parameters  $\theta$  (sometimes called hyperparameters) resulting from the mean and kernel function that must be *marginalized* in order to perform inference. These parameters are talked about in Section 3.3. As a GP is within the Bayesian framework, priors are first assigned to these hyperparameters using distributions utilizing

our domain knowledge. Kernel parameters and incorporating expert opinion through the use of the autocorrelation function (ACF) is described in Section 3.3.2. Ideally, with prior distributions assigned to the hyperparameters, we would like to use Bayes rule to compute the posterior to find the *most likely* parameters:

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \quad (14)$$

However, the marginal likelihood  $p(y|\theta)$  is almost always intractable and we must therefore use approximation techniques [21]. To find the *most likely* hyperparameters, the marginal log-likelihood is maximized through gradient-based optimization and is given by the form:

$$\log P(y|X, \theta) = -\frac{1}{2}y^T(K_y + \sigma_n^2)^{-1} - \frac{1}{2}\log|K_y + \sigma_n^2| - \frac{n}{2}\log(2\pi) \quad (15)$$

In Section 5.1, the experimentation setup pertaining to the type of optimizer used, number of training iterations, and learning rate is given. Although the above is technical, GPs have an intuitive procedure for training and performing inference. First, prior beliefs are encoded using kernel functions. Second, condition on what data has been observed. Third, optimize the parameters. Lastly, use the posterior predictive distribution for performing inference.

### 3.3 Covariance and Mean Functions

As outlined above, the kernel design is a vital step in GP model design and offers an opportunity to incorporate domain knowledge into the model. Below, a few kernels and their hyperparameters that will be useful in modeling non-linear time series will be introduced and subsequently, an explanation on how to compose kernels through addition or multiplication will be provided.

#### 3.3.1 Squared Exponential Kernel

First, the *squared exponential* or *radial basis function (RBF)* kernel is one of the most widely used kernels for real-valued inputs. Notably, the *RBF* is a stationary kernel that encodes a high degree of smoothness in the function space [21], and hence can be used to design a GP which produces functions that can be smooth,

$$K_{RBF} = \sigma^2 \exp\left(-\frac{\|x - x'\|}{2\ell^2}\right) \quad (16)$$

where  $\ell$  is the lengthscale and controls the “wigginess” of the function and  $\sigma^2$  is the overall variance / outputscale ( $\sigma$  is also known as the amplitude) of the function. In Figure 1 below,

three functions are sampled and visualized from an *RBF* kernel; each with different length and outputscale values.

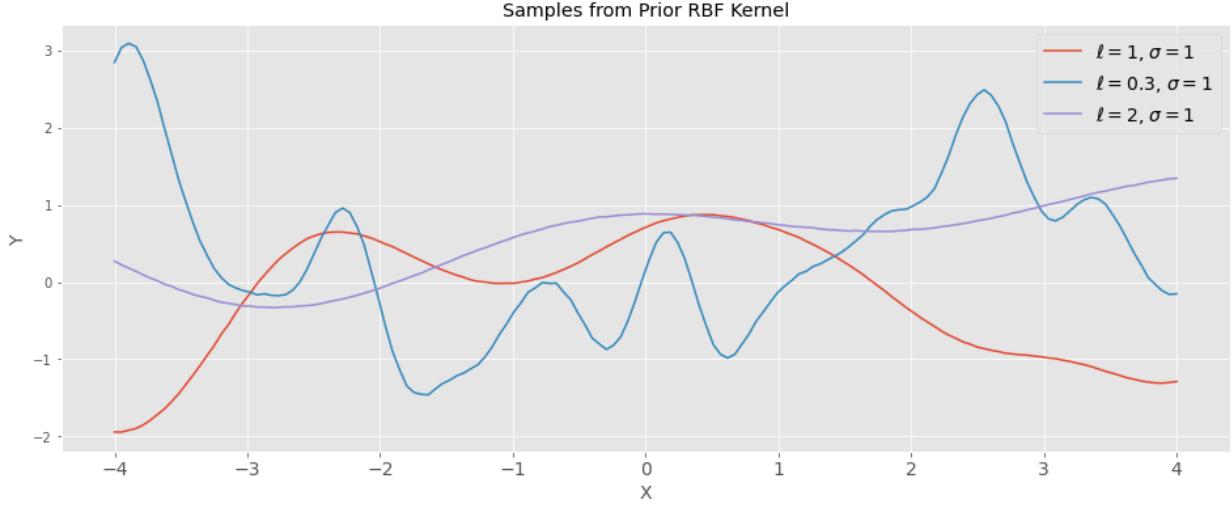


Figure 1: Three samples, each from an *RBF* kernel with different hyperparameters denoted by the table legend. Notice how  $\ell$  affects the smoothness of the function.

### 3.3.2 Periodic Kernel

The *periodic kernel* (*Per*) captures repeating structures [21], which can model seasonalities influenced by business cycles and or human behavior. For example, home electricity consumption often displays daily seasonalities. The *Per* kernel has the form

$$K_{per} = \exp\left(-\frac{2}{\ell^2} \sin^2\left(\pi \frac{r}{p}\right)\right) \quad (17)$$

where  $p$  is the period. Again,  $\ell$  and  $\sigma^2$  are the lengthscale and outputscale respectively. Figure 2 below visualizes the repeating structures of the *Per* kernel. Important in this thesis is how the period is chosen. In a time series, the correlation between any pair of outputs can be analyzed empirically using the ACF. The ACF compares the time series with itself at a certain lag. Namely:

$$\hat{p}(k) = \frac{\sum_{s=1}^{n-k} (x_{s+k} - \bar{x})(x_s - \bar{x})}{\sum_{t=1}^{n-k} (x_t - \bar{x})^2} \quad (18)$$

Thus, using the ACF coefficients, the interval of periods that show significant autocorrelations are chosen as the prior belief for the time series cyclical period length. Increasing the period  $p$  increases the distance between repetitions. Figure 3 below shows an example of

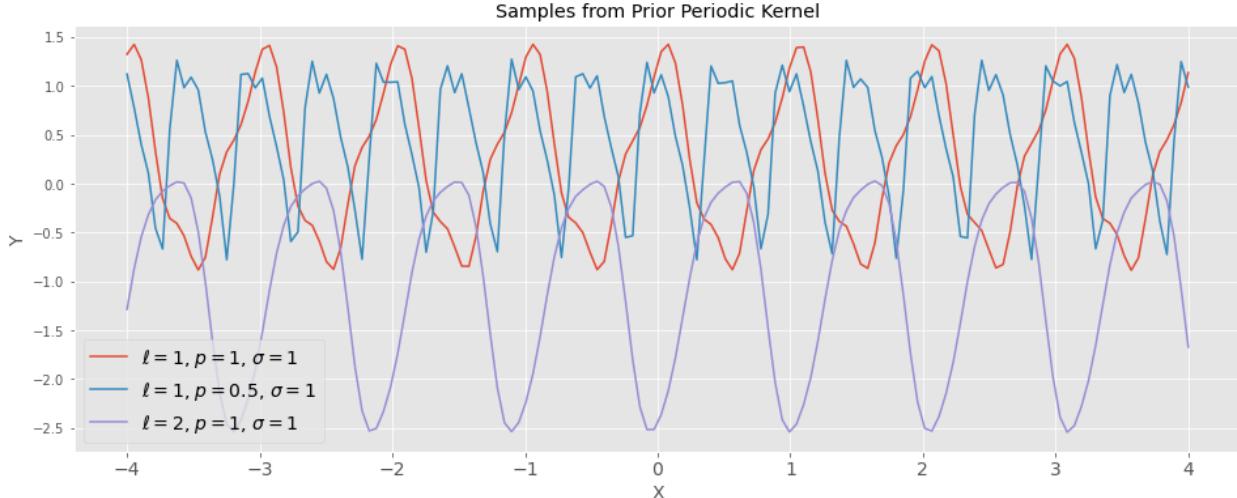


Figure 2: Three samples, each from a *Per* kernel with different hyperparameters denoted by the table legend. Notice how  $p$  affects the periodicity of the function and  $\ell$  affects the “wiggliness” of the variations.

the ACF correlogram for the paper disposal machine and how one can use the coefficients at certain lags as a prior belief for modeling the cyclical component.

### 3.3.3 Rational Quadratic Kernel

The *rational quadratic* (*RQ*) can be interpreted as the sum of many *RBF* kernels of different lengthscales. Similar to the *RBF*, the *RQ* kernel encodes smoothness in the function space, but with the additional flexibility of having both local variations and long term variations [23].

$$K_{RQ} = \sigma^2 \exp\left(1 + \frac{(x - x')^2}{2\alpha\ell^2}\right) \quad (19)$$

where  $\alpha$ , also known as the scale mixture, determines how much local variations from the smaller lengthscales contribute to the overall variation. By using this kernel, one can model non-periodic trends of the underlying physical process and interpret the hyperparameters  $\ell$  as either a non-periodic hourly or daily trend depending on how large or small the value of  $\ell$  is respectively. Three samples of the *RQ* kernel are visualized in Figure 4.

### 3.3.4 Kernel Composition

Kernels can also be combined through addition and multiplication given the result is a positive definite kernel [21]. Therefore, by combining kernels, more complex covariance structures can be designed. The addition of two kernels  $K_1(x, x')$  and  $K_2(x, x')$  is analogous to the probabilistic “OR” operation, i.e., two points  $x_1$  and  $x_2$  are considered highly correlated if they

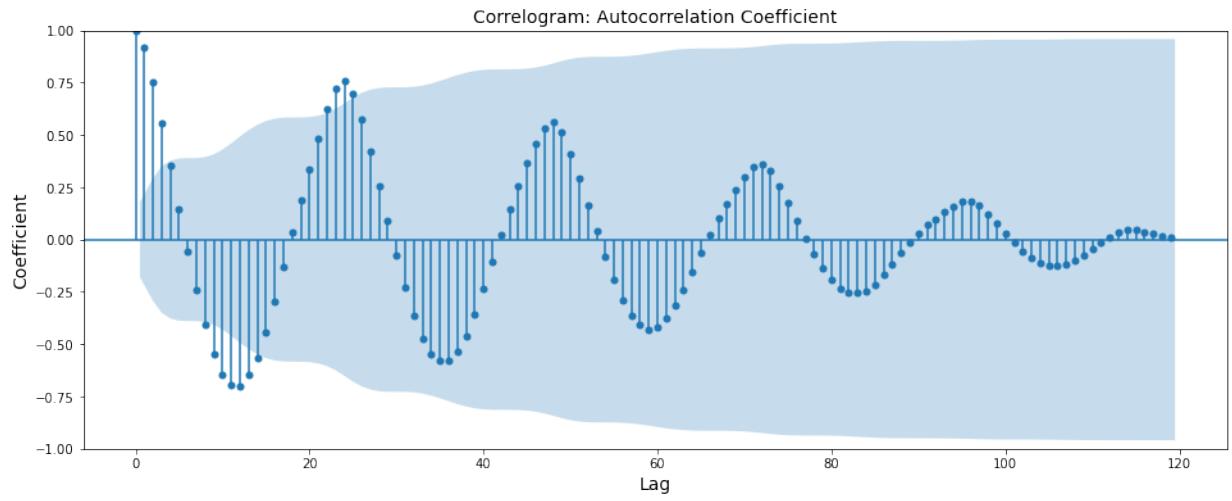


Figure 3: Entsorgung ACF. The time series is discretized and averaged at 1 hour intervals. Therefore, the ACF shows significant autocorrelations between 10 and 12 hours and 22 and 24 hours with the remaining lags decreasing in amplitude and significance.

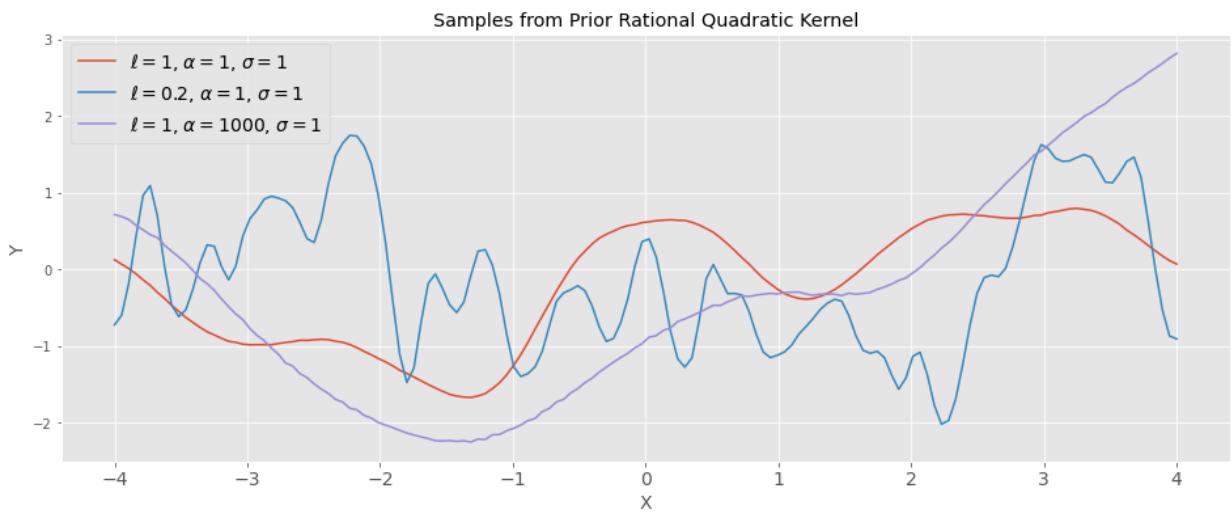


Figure 4: Three samples, each from a  $RQ$  kernel with different hyperparameters denoted by the table legend. Notice how  $\alpha$  affects the local variations of the function while  $\ell$  affects the “wiggliness” of the variations.

are highly correlated in  $K_1 \vee K_2$ . The multiplication of the two kernels is analogous to the probabilistic “AND” operation, i.e., two points  $x_1$  and  $x_2$  are considered highly correlated if they are highly correlated in  $K_1 \wedge K_2$ . The resulting kernel given by the addition or multiplication of the two kernels is given by:

$$K(x, x') = K_1(x, x') + K_2(x, x') \quad (20)$$

$$K(x, x') = K_1(x, x') * K_2(x, x') \quad (21)$$

### 3.3.5 Locally Periodic Kernel

In time series analysis, the periodic structure of the underlying function may change and/or may not be consistent over time. For example, Figure 10 shows a daily repeating cycle that changes its structure over the course of the week. Thus, one may want to incorporate this locally changing periodic structure “belief” into the kernel design [23]. To do this, a *locally periodic* (*LocPer*) is introduced. The *LocPer* kernel is a product of the  $K_{RBF}$  and  $K_{Per}$ :

$$K_{LocPer} = \sigma^2 \exp\left(-\frac{2\sin^2(\pi(x - x')/p)}{\ell_{Per}^2} \exp\left(-\frac{(x - x')^2}{\ell_{RBF}^2}\right)\right) \quad (22)$$

The periodic component correlates points that are far away from each other, but still in the same phase of a cycle. The  $K_{RBF}$  decorrelates the two points and how quickly the points are decorrelated is determined by the hyperparameter  $\ell_{RBF}$ . Remember,  $\ell_{RBF}$  determines the “wiggleness” of the function; therefore smaller  $\ell_{RBF}$  corresponds to faster changing periodic cycles and vice versa. The  $K_{LocPer}$  is of use in modeling industrial machine equipment energy consumption as production processes are typically influenced by human behavior. Below, in Figure 5, three samples are generated from a *LocPer* kernel.

## 3.4 Advantages and Disadvantages of Gaussian Processes

In this thesis, Gaussian Processes for time series was chosen for the advantages to (1) perform non-parametric probabilistic inference to model non-linear time series, (2) allow for an intuitive model interpretation by way of *decomposition* of the kernels, and (3) incorporate background information in the form of *kernel design*.

However, GPs also have disadvantages. The main one being a result of the kernel  $K$ , that when performing inference and likelihood evaluation, to compute the weights, the method has a computational complexity of  $\mathcal{O}(N^3)$ . That is, the time complexity is cubic in the number of points  $|N|$  [21]. However, the open source software used in this thesis [24] for implementing Exact Gaussian Processes (ExactGP) is based on Blackbox Matrix-Matrix multiplication (BBMM) [25] to reduce the time complexity down to  $\mathcal{O}(N^2)$ . Another disadvantage of GPs

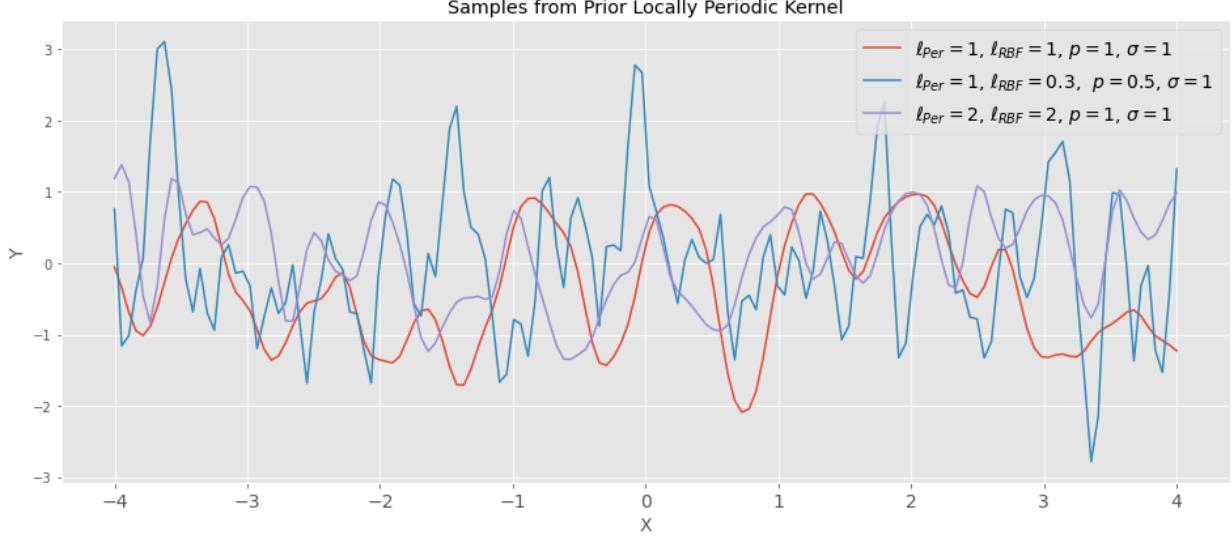


Figure 5: Three samples, each from a *LocPer* kernel with different hyperparameters denoted by the table legend. Notice how the product of the two kernels results in a periodic structure with variations in each cycle.

is the choice of kernel design. Knowing which kernel to use to model the underlying process and how to compose them can be a challenging task. However, using other people’s domain knowledge and other methods such as the ACF makes the task more manageable.

### 3.5 Evaluation Metrics

To evaluate the performance of the model, the data set is first split into training and testing sets. The training set consists of four days of data whereas the testing set consists of one day (24 hours) worth of data. Then, as GPs are probabilistic models, to evaluate model quality, not only should the accuracy of the model’s predictions be computed, but also how precise the predictions are using the posterior predictive distribution. Thus, deterministic and probabilistic error metrics are used in this thesis. The following metrics are used to evaluate the performance and quality of the model: Mean Squared Error (MSE), RMSE, MAPE, Average Coverage Error (ACE), and Pinball Loss.

- **MSE:** Also known as the average squared loss, measures the average squared distance the predictions are from the actual values. A larger MSE indicates that the data points are dispersed more widely around the mean, whereas a smaller MSE suggests the inverse. In regression, the smaller the MSE, the better. MSE is given by:

$$\frac{1}{n} \sum_{i=1}^n (\hat{y}_t - y_t)^2 \quad (23)$$

where  $\hat{y}_t$  is the forecasted value at time  $t$  and  $y_t$  is the actual value at time  $t$ .

- *RMSE*: Is the square root of MSE. The square root function turns the metric back into the same units as the dependent variable  $y$ . Again, the smaller the RMSE, the better and is defined as:

$$\sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_t - y_t)^2} \quad (24)$$

where  $\hat{y}_t$  is the forecasted value at time  $t$  and  $y_t$  is the actual value at time  $t$ .

- *MAPE*: Measures the accuracy of the forecast as a percentage and offers an intuitive way to communicate the quality of a model to non-technical audiences. For example, a 15% MAPE would mean that, on average, the model is off the true value by 15%. The smaller the MAPE, the better, and is defined as:

$$\frac{1}{n} \sum_{i=1}^n \frac{|y_t - \hat{y}_t|}{A_t} \quad (25)$$

where  $y_t$  is the actual value at time  $t$  and  $\hat{y}_t$  is the forecasted value at time  $t$ . The absolute difference is then divided by the actual value  $y_t$  to obtain a scale-independent measure.

- *ACE*: Sampling functions from the posterior predictive distribution, one can obtain prediction interval (PI). In this thesis, two standard deviations (95%) is used. ACE measures the proportion of actual values within the PI and is bounded between [0, 1]:

$$I_t = \begin{cases} 1 & \text{if } P_t \in [\hat{L}_t, \hat{U}_t] \\ 0 & \text{if } P_t \notin [\hat{L}_t, \hat{U}_t] \end{cases} \quad (26)$$

$$UC = \frac{1}{|T|} \sum_t I_t \quad (27)$$

where  $P_t$  is the actual value at time  $t$ , and  $I_t$  is a binary indicator of whether the PI  $[\hat{L}_t, \hat{U}_t]$  contains  $P_t$ . Since ACE measures a proportion of the actual values contained by the PI, this proportion measures the discrepancy between the percentage of points contained by the PI and the confidence interval (CI) of the PI; here two standard deviations represents a 95% CI [23]. However, ACE can be misleading due to the fact that a wide PI can cover all the actual data points—resulting in a high score. Therefore, ACE is complemented by the Pinball loss.

- *Pinball Loss*: Measures the sharpness of the PI which evaluates the precision of the prediction and how sharp (or tight) the PI is around the actual value [23]. Pinball loss is defined as:

$$\text{Pinball}(q, t) = \begin{cases} (1-q)(\hat{Q}_t(q) - P_t) & \text{for } P_t < \hat{Q}_t(q) \\ (q)(P_t - \hat{Q}_t(q)) & \text{for } P_t \geq \hat{Q}_t(q) \end{cases} \quad (28)$$

where  $\hat{Q}_t(q)$  is the predicted  $q^{th}$  quantile at time  $t$ . The pinball loss is calculated at every time step  $t$  and is then averaged. The final pinball loss metric is calculated by averaging the pinball loss for 99 percentiles. Thus, what is desired, is a prediction with high ACE and a low Pinball loss which indicates a probabilistic prediction is accurate and with low uncertainty.

During the experimentation phase, as outlined in Section 5.2, different model and kernel designs were compared such that the better model was the one that minimized MSE, MAPE, RMSE, and Pinball loss, while subsequently maximizing ACE. Also, in this thesis, the metrics MSE, MAPE, and Pinball loss were computed using scikit-learn’s [26] metric library. Furthermore, ACE and RMSE were calculated from scratch.

## 4 Datasets

### 4.1 CLEMAP Client Dataset

The client of CLEMAP operates within the Swiss printing and media industry where the equipment has been fitted with state of the art CLEMAP sensors on a range of their production equipment. The machines, location, and purpose are described in Table 1:

Machine/Component	System	Amperes	Location	Floor	Purpose
Gesamtmessung		1600	Bau II	0	Main terminal
Hauptluftung	HVAC	250	Bau II	-1	Main ventilation
Kältemaschiene		200	Bau II	1	Refrigeration
Printer (Drückmaschine)	XL106	315	Bau II	2	Heidelberg Speedmaster printer
UV Scan	XL106	160	Bau II	1	UV scan on XL106
UV Sigmaline EG		315	Bau II	0	UV lamp
Stahl Folder		63	Bau II	0	Steel folding
Printer (Drückmaschine)	R707LV	160	Bau II	2	Manroland printer
Puderabsauger	R707LV	25	Bau II	2	Powder extraction
Vari Air	R707LV	100	Bau II	2	Printer variational air
Trockner	R707LV	160	Bau II	2	Dryer
UV Feinabgäng		125	Bau II	0	UV fine particle extractor
Papier Entsorgung		125	Bau II	-1	Paper disposal
UV 1.0G		125	Bau II	1	UV server room
UV 2.0G		125	Bau II	2	2 <sup>nd</sup> floor
UV 3.0G		125	Bau II	3	3 <sup>rd</sup> floor
UV 4.0G		160	Bau II	4	4 <sup>th</sup> floor
UV EG		125	Bau II	0	UV ground floor

Table 1: List of items being metered by CLEMAP. A system may be composed of several machines, all of which are being metered.

All machines and the main terminals are connected to an alternating current (a.c) grid with three phases. Subsequently, the data collected is:  $V$ ,  $I$ ,  $W$ ,  $P$ ,  $Q$ ,  $PF$ , and  $S$  on each phase of the three phase load. The measurements were originally collected at a frequency of 12Hz from October 7<sup>th</sup> until October 18<sup>th</sup> for a total length of 10 days. A three phase current at 12Hz indicates that every second, there should be 12 cycles being measured for each phase 1, 2 and 3.

As outlined in Section 3.4, and due to the GPyTorch Gaussian Process time complexity of  $\mathcal{O}(n^2)$ , the time series is discretized by aggregating the data into 10 and 30 minute intervals and taking the average value. This aggregation significantly reduces the length of the original 12Hz time series while still allowing for a more granular level of analysis compared to the current literature in Section 2.

#### 4.1.1 CLEMAP Exploratory Data Analysis

Here, an introductory EDA is performed on some of the machines and components that will be modeled in Section 5. First, a bar plot broken down by machine energy consumption and hour of the day for the 10 days is visualized for a general overview of the measurement setup.

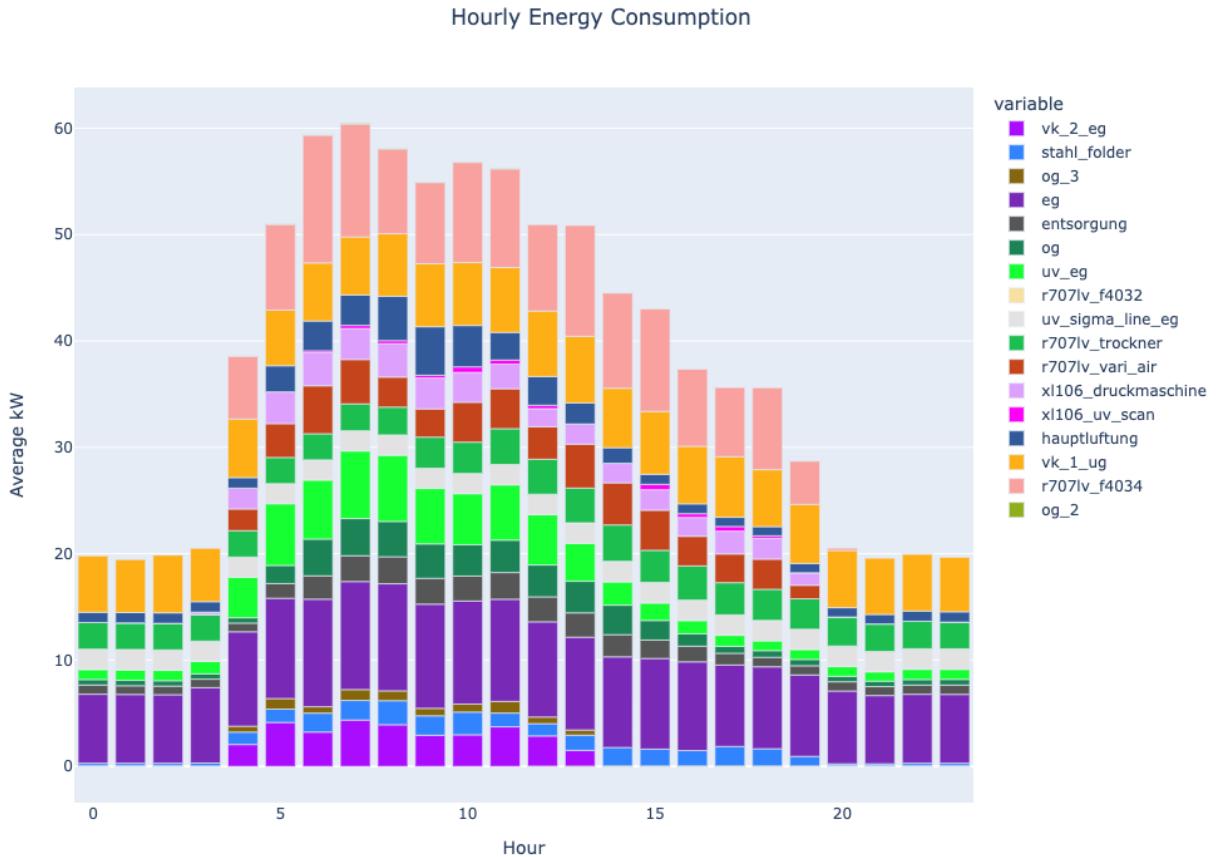


Figure 6: CLEMAP sensor measurement setup. Hourly load energy consumption (kWh) for each machine over the course of 10 days.

Using the meta data from Table 1 and the bar chart above, it is evident that machines on the ground, first, and second floor affect the energy being metered on their respective floors (UV 1.0G, etc.). Likewise, the meter on the main terminal is measuring all of the energy. The majority of the components being metered, with the exception of the Trockner, UV Sigmaline EG and VK 1.UG, are effected by the hour of the day. In the system R707LV, the Manroland printer demanded little to no energy, while the remaining three components represent a large share of total energy demanded with the powder extraction machine consuming the most. Subsequently, in the XL106 system, the Heidelberg printer represents the majority of energy demanded by the system.

For the sake of brevity, only the EDA for the paper disposal machine (Figure 7) will be shown below. For a full machine analysis, please refer to the `eda` directory on the `experiments` branch of the GitHub repository. Information and a link to the repository can be found in Table 4 of Appendix B.



(a) Outside of machine.



(b) Output of machine.

Figure 7: Paper disposal machine

Building off of the bar plot visualizations above and the characteristics of time series in Section 3.1, it is likely that the paper disposal machine has a cyclical pattern. Figure 8 is an hourly load profile heat map of the paper disposal machine. It is evident there is a daily periodicity from Monday to Friday with little to no demand on the weekend. The heat map indicates a daily and potential weekly pattern; though more data is needed to confirm the weekly periodicity hypothesis.

Next, in Figure 9, a plot of the time series at a 10 minute interval is visualized where 2021-10-08 is a Friday and 2021-10-18 is a Monday. The load profile shows a daily, and potentially sub-daily periodic pattern. Also, the 10 days of data does not split into an even two weeks of data. Rather, there is only observations for one full production week. This inconsistency does not allow for an analysis or modeling of the weekly periodicity.

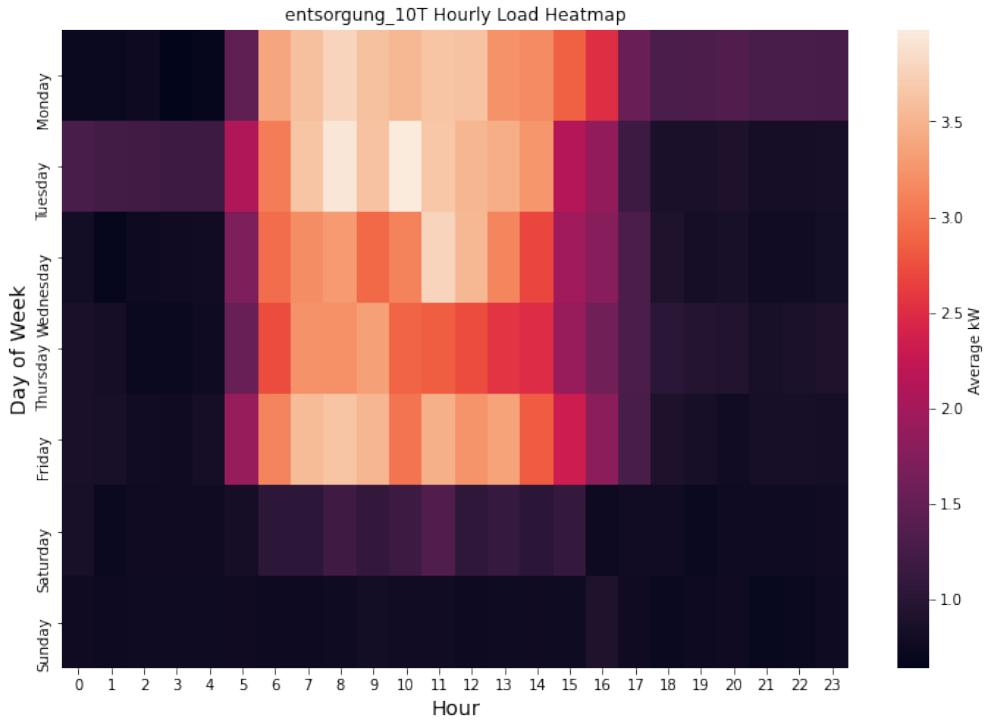


Figure 8: Paper disposal hourly load profile. The color bar on the right indicates the average kW consumed for that hour over the 10 day period.

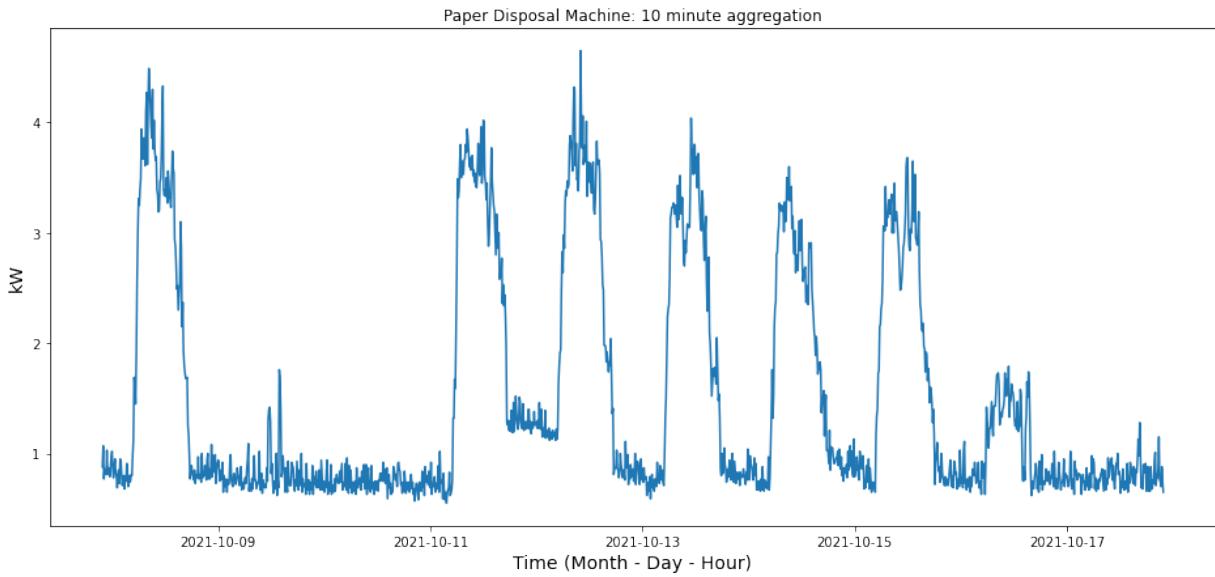


Figure 9: Entire time series for the paper disposal machine.

Therefore, during the modeling phase, only the production week (2021-10-11 through 2021-10-15) is used. Using this time series, the periodicity is verified empirically using the ACF; as explained in Section 3.3.2. Below, in Figure 10, the production week time series is visualized

along with the respective ACF correlogram. Looking at the ACF and time series plot, the sub-daily and daily periodicity is verified at about 12 and 24 hours, respectively. Also, on some days, such as the 11<sup>th</sup>, 13<sup>th</sup> and 14<sup>th</sup>, the load profile slowly tapers off compared to the other days. Furthermore, during non-operational hours, the machine seems to be “on” and in a “stand-by” mode.

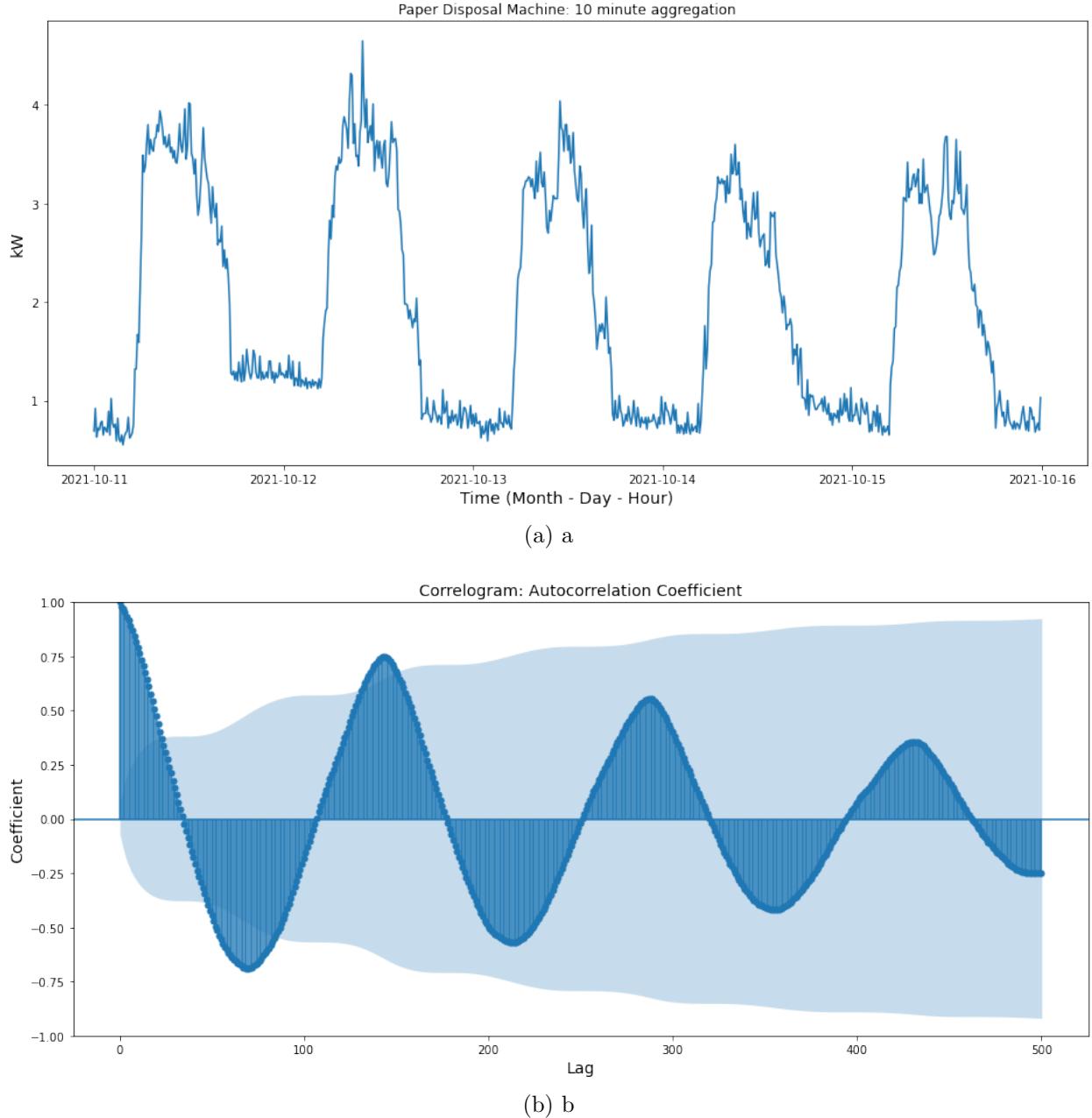


Figure 10: (a) Production week time series at 10 minute aggregations. (b) ACF correlogram indicates significant periodic cycles at about 12 and 24 hours, respectively. For example,  $(75 * 10) / 60 = 12.5$

Finally, the original time series is split into random, shorter time samples to gain a better visual understanding of the original 12Hz frequency and cycle patterns. Indeed, in Figure 11, a repeating cyclical pattern is identified during for, what could be called a “stand-by” state, the non-operational time of the machine from 17:30 on October 11<sup>th</sup> until 05:00 on October 12<sup>th</sup>. Subsequently, (b) the duration of the peak load, shows a less identifiable cycle.

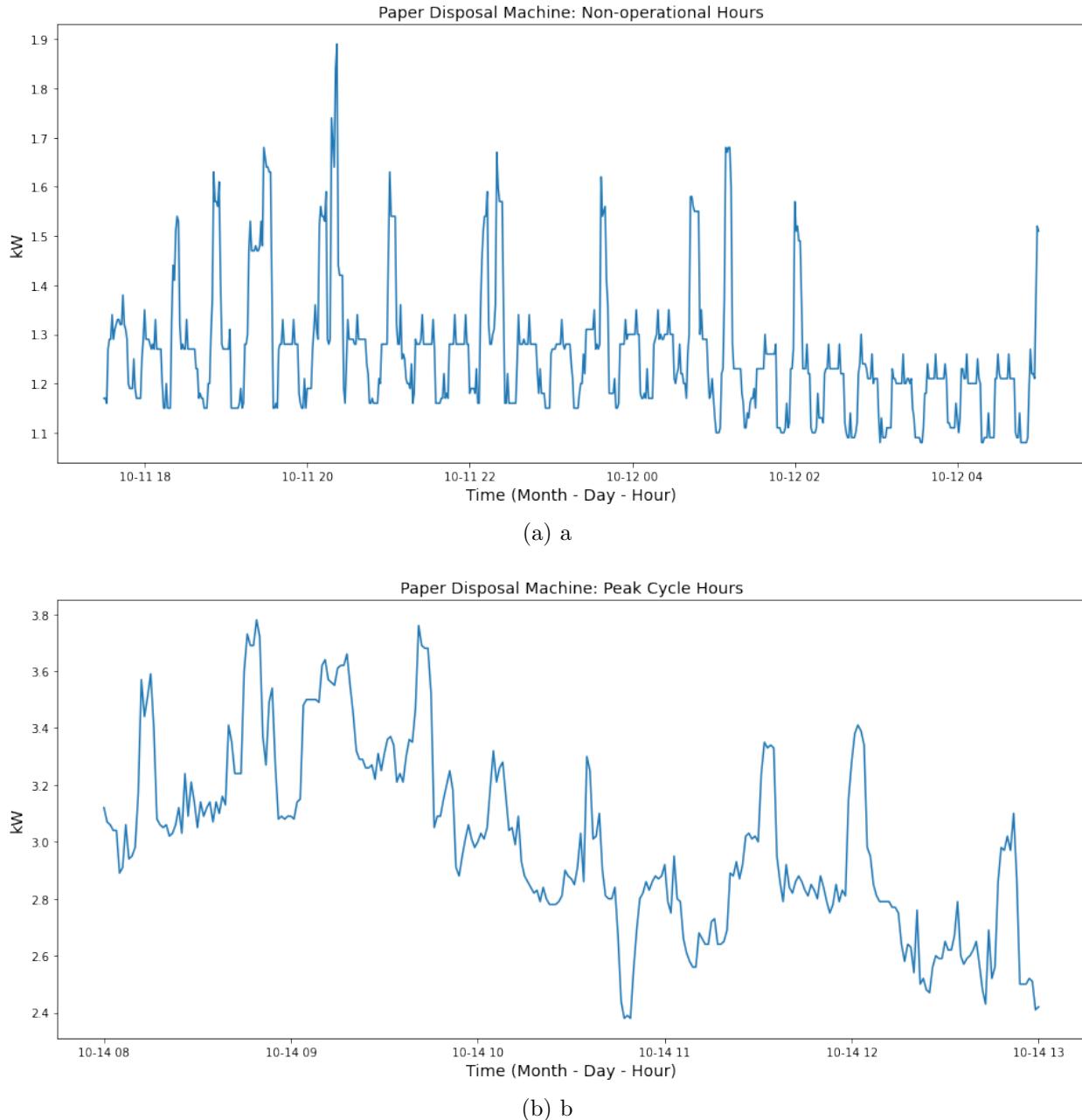


Figure 11: (a) Random profile of non-operational hours from 17:30 - 05:00 (b) Random peak load profile from 08:00 - 13:00

## 4.2 HIPE Industrial Energy

Complementing CLEMAP's client machine-level data, a second source of data, the High Resolution Industrial Production Energy (HIPE) energy status data set [27], will also be analyzed to ensure the scalability of the proposed methodologies in section 4. By developing the algorithms on a range of industrial equipment, the methods developed in this thesis can give vital feedback on the difficulties and opportunities of the scalability and feasibility of models built on specific data sets to new data of similar applications.

The HIPE data set comes from the Institute of Data Processing and Electronics (IPE) of Karlsruhe Institute of Technology (KIT) in Germany which operates an electronics production site. It produces electronic systems for particle physics, battery systems, and medical applications in small batches, i.e., less than 1,000 pieces. Several machines have been instrumented with smart meters in which the machines are either connected to one phase or three phase power. The machines and their purpose are outlined in Table 2 below.

Machine/Component	Purpose
Pick and place unit	Placement of electronic components, such as resistors and microcontrollers, on a printed circuit board (PCB). Energy consumption depends on the quantity of components per PCB and on the number of boards
Soldering oven	Components soldering to PCB. Energy consumption depends on throughput speed and temperature
Washing machine	Cleaning of PCB. Energy consumption depends on temperature and process duration
Screen printer	Printing of material layers to interconnect electronic components via thick-film technology
Vacuum Pump 1	Auxiliary machines to generate vacuum for other machines such as PickAndPlaceUnit. Energy consumption depends on vacuum demand
Vacuum Pump 2	Auxiliary machines to generate vacuum for other machines such as PickAndPlaceUnit. Energy consumption depends on vacuum demand
High Temperature Oven	Heats up to 1200 °C, fixing layers for thick-film technology. Energy consumption depends on temperature and heating duration
Vacuum Oven	Oven with vacuum chamber
Chip Saw	Separation of chips of a silicon wafer. Energy consumption depends on the wafer thickness
Chip Press	Heat treatment of surfaces under high pressure, e.g., for multi-layered PCB. Energy consumption depends on pressure and temperature

Table 2: List of items being metered by KIT at IPE and their purpose.

All machines and the main terminals are monitored using EEM-MA600 energy meters and are connected to an ac-grid with three phases. The main measurements of interest are a three month time series at a five second resolution of the electrical quantities:  $V$ ,  $I$ ,  $A$ ,  $PF$ ,  $S$ , and  $Q$ . Additionally, data related to harmonic distortions was also collected and can be used to quantify its relationship with energy consumption. A power system's ability to perform at optimal levels is compromised when harmonic distortion enters the system. It creates inefficiencies in equipment operations due to the increased need for power consumption [28].

As the goal is to evaluate the scalability and feasibility of the methods introduced in Section 2 and Section 3, the EDA on the HIPE data is similar to that of the CLEMAP dataset. Therefore, refer to the `hipeloadprofiles` directory within the `eda` directory on the `experiments` branch in the GitHub repository. See Table 4 in Appendix B for more details.

### 4.3 Similarity and Differences of Data Sources

The HIPE dataset was chosen because of various similarities such as the form of energy being electrical, metering of three phase power, longer duration (3 months) and low time sampling (5 seconds), relatively similar load profiles for a subset of equipment, and similar electrical measurements.

Where the data sets differ, are in regard to the load profiles of the following machines: chip press, pick and place unit, high temperature oven, vacuum oven, screen printer and vacuum pump 2. The load profiles of these machines are either linearly increasing, constant, and or contain too little data for analysis. Therefore, for determining the applicability and feasibility of the models and methods, the following machines will be utilized; vacuum pump 1, soldering oven, and main terminal.

## 5 Results

### 5.1 Experimentation Setup

An energy baseline model using the methodology described in Section 3 will be developed for the following machines, at 10 and 30 minute time aggregations, to analyze EE and perform PDD: main terminal, main ventilation, paper disposal, uv server room, and uv ground floor. The implementation of Gaussian Process Regression is with the open source package GPyTorch [24]; a highly efficient and modular implementation of GPs built on top of PyTorch<sup>2</sup>, with graphical process unit (GPU) acceleration.

As multiple machines, each with two time aggregations, are being modeled, an experimentation workflow is introduced to track and log results pertaining to the evaluation metrics outlined in Section 3.5 and the GP kernel design described in Section 3.3.5. The flowchart in Figure 12 provides the overview for the experimentation setup.

Starting at the top, a machine’s measurement data provided by CLEMAP’s meters is queried where an EDA is performed similar to that in Section 4.1.1. Subsequently, the GP model and kernel design begins using the results from the EDA as our prior belief. Here, kernels pertaining to cyclical patterns and local variations are designed. At the same time, the data set is split into a training and validation set. The training set, representing data from 11.10.2021-15.10.2021, is used for the optimization of the kernel hyperparameters. For optimization, the gradient-based method adaptive moment estimation (ADAM) [21] is used at different learning rates (lr) and iterations (see Table 3 for results). Then, with the optimized hyperparameters, inference is performed using the posterior predictive distribution. The composite kernel is decomposed into its individual kernel components to analyze the effect of each kernel. This decomposition allows one to investigate if increasing the model complexity (by introducing more kernels) contributes to a decrease or increase in uncertainty (using Pinball loss). Lastly, the evaluation metrics are calculated on the predictions  $\hat{y}$  and test data  $y_{test}$  and logged in the database. Based on these metrics, it is determined if further kernel designs and or model refinements need to be tested. If yes, then repeat, else save model parameters and end process. The end of this workflow results in an energy baseline model ready for analyzing EE and performing PDD for the given machine.

---

<sup>2</sup><https://pytorch.org>

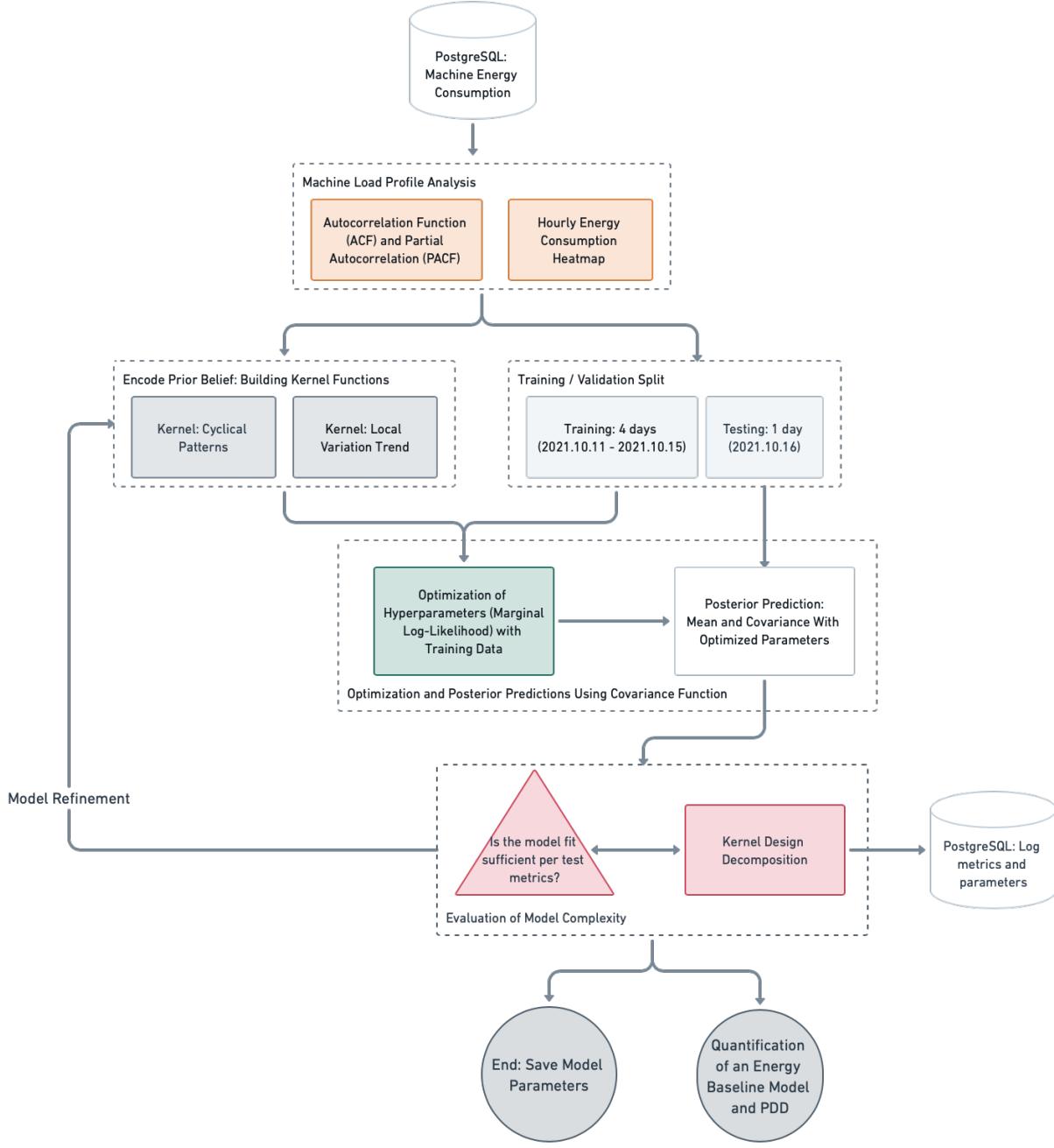


Figure 12: Experimentation workflow for developing and refining GP energy baseline models.

## 5.2 Kernel Design and Composition

In this section, the kernel design is outlined for the paper disposal machine using the results from the EDA and kernels described in Section 3.3. Recall that the GP model uses time as input and kW (Watts / 1000) as the output. Before the kernel design and composition is

outlined, a particularity of GP models is the normalization and or standardization of data before hyperparameter optimization. Thus, the inputs (time) is normalized between  $[0, 1]$  and the output (kW) is standardized to have zero mean and unit variance. The inputs and outputs are then inverse transformed to their original scale before calculating evaluation metrics.

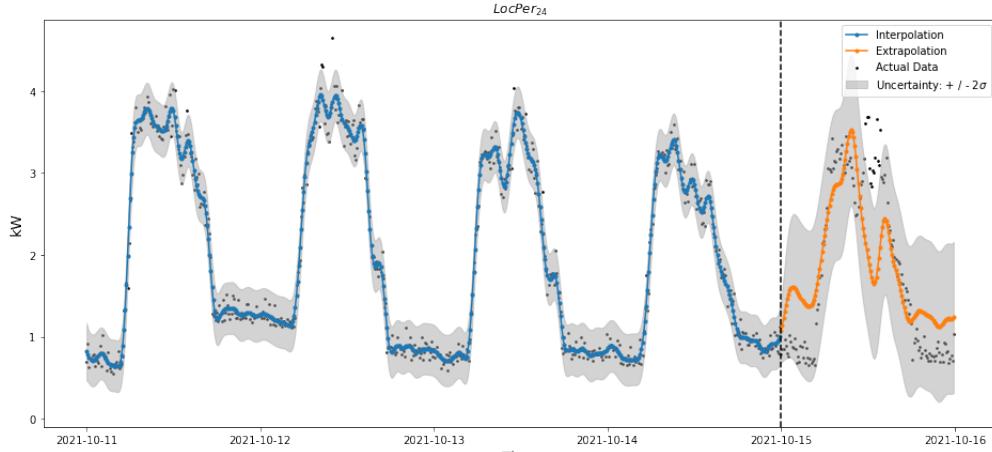
In Section 3.3, four different kernels were introduced, one of those being a product kernel, to model non-linear time series. Depending on the significant coefficients from the ACF, the addition of two locally periodic  $K_{LocPer}$  kernels with period intervals  $p$  of  $10 : 14$  and  $22 : 26$  are typically used. The two  $LocPer$  kernels allow one to model not only the daily and half-daily cycles of the machines, but also the changing periodic shape over time. The  $RQ$  kernel models any local and non-periodic trends in the time series. Finally, these kernels are combined through the addition operation, which can be seen as identifying a correlation between two time points if any of the component kernels indicate a high correlation at those points. The addition operator results in the following kernel:

$$K_{LocPer24} + K_{LocPer12} + K_{RQ} \quad (29)$$

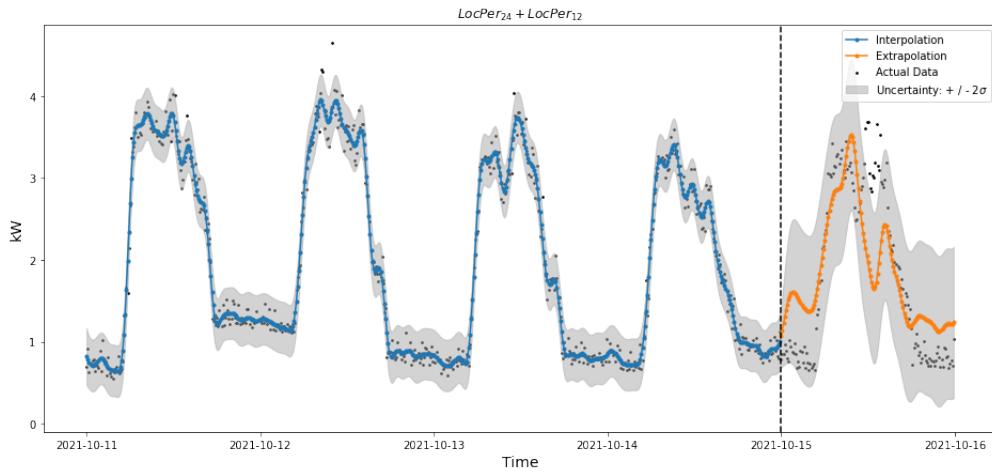
where  $K_{LocPer24}$  and  $K_{LocPer12}$  are product kernels consisting of a  $K_{Per}$  and  $K_{RBF}$ . Here, for the period  $p$ , it is denoted as  $p = 24$  and  $p = 12$  as this represents our prior hypothesis. However, in GPyTorch, an interval is defined according to the significant ACF coefficients which allows for greater flexibility. Furthermore, combinations of the individual kernel components in (29) are tested, i.e.,  $K_{LocPer24} + K_{RQ}$ ,  $K_{LocPer12} + K_{RQ}$ , etc., until all combinations are reached. Experimenting and logging evaluation metrics with simpler kernel compositions allows one to judge the complexity of the model. If a simpler kernel design results in *better* evaluation metrics, then the simpler kernel composition shall be chosen.

### 5.3 Kernel Design Decomposition

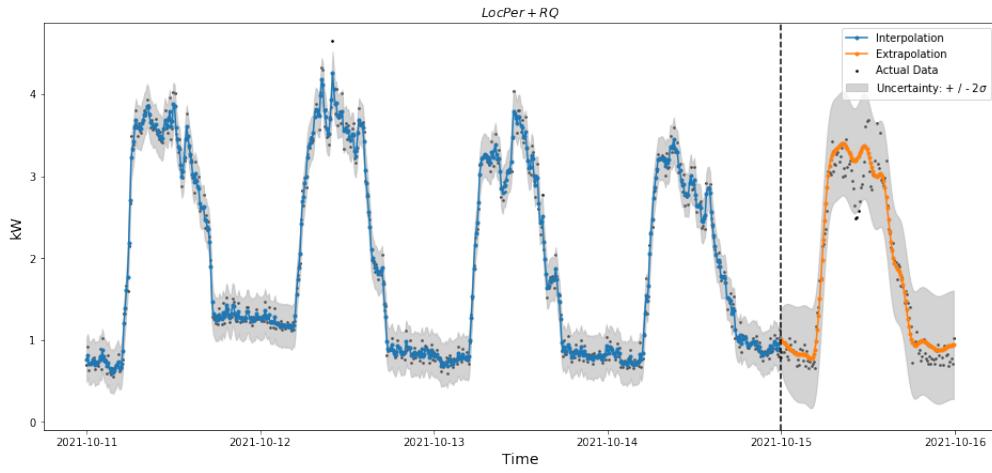
To illustrate the effect each kernel has on the composite kernel, the data is cumulatively fit with more kernels until the composite kernel in (29) is reached [23]. In Figure 13, starting with the  $K_{LocPer24}$ , the model is able to capture the repeating daily periodicity with variations from day to day, i.e., not every cycle is the same. However, when it comes to extrapolation, uncertainty grows significantly indicated by the PI and predicts rather poorly. The addition of the  $K_{LocPer12}$  results in hardly any difference. This supports the conclusion that the addition of the  $LocPer12$  kernel increases complexity but doesn't increase the quality of the model and therefore, it is suggested to leave this kernel out of the final kernel design. Indeed, referring to Table 3, the best kernel design according to the evaluation metrics is a composite kernel consisting of  $K_{LocPer24} + K_{RQ}$ . Lastly, with the addition of  $K_{RQ}$ , the composite kernel in (29) is reached. The addition of this kernel results in a tighter PI as a result of having more parameters, and thus greater flexibility to model non-periodic and irregularities in the data.



(a) a



(b) b



(c) c

Figure 13: Cumulatively adding more complex kernels. The dashed line represents the training and validation split. (a)  $K_{LocPer24}$ , (b)  $K_{LocPer12}$ , and (c)  $K_{LocPer24}+K_{LocPer12}+K_{RQ}$

## 5.4 Energy Baseline Model Experiment Performance Metrics

Following the kernel decomposition, the evaluation metrics and elapsed time (seconds to train) are queried from the PostgreSQL database to determine which kernel design resulted in the best performance according to Section 3.5. The query results from modeling the industrial partner data set is shown below.

Device	Aggregation	Kernel	MSE	MAPE	RMSE	ACE	Pinball	Elapsed Time
Paper disposal	10	$K_{LocPer24} + K_{RQ}$	0.046	0.117	0.215	0.99	0.082	6.23
	30	$K_{LocPer24} + K_{RQ}$	0.032	0.079	0.179	1.0	0.062	2.67
Main terminal	10	$K_{LocPer24} + K_{LocPer12} + K_{RQ}$	169.354	0.445	13.014	0.757	4.913	8.60
	30	$K_{LocPer24} + K_{LocPer12} + K_{RQ}$	163.175	0.447	12.774	0.771	4.806	2.96
EG	10	$K_{LocPer24} + K_{LocPer12} + K_{RQ}$	2.582	0.117	1.607	0.80	0.455	11.08
	30	$K_{LocPer24} + K_{LocPer12} + K_{RQ}$	2.63	0.115	1.622	0.792	0.443	2.97
HVAC	10	$K_{LocPer24} + K_{RQ}$	0.462	0.132	0.68	0.931	0.171	8.69
	30	$K_{LocPer24} + K_{RQ}$	0.287	0.132	0.536	0.958	0.173	2.43
OG 1	10	$K_{LocPer24} + K_{RQ}$	0.195	0.178	0.441	0.965	0.146	8.84
	30	$K_{LocPer24} + K_{RQ}$	0.162	0.155	0.402	0.912	0.132	2.47
UV EG	10	$K_{LocPer24} + K_{LocPer12} + K_{RQ}$	2.599	0.364	1.612	0.91	0.457	11.23
	30	$K_{LocPer24} + K_{LocPer12} + K_{RQ}$	2.068	0.296	1.438	0.917	0.378	2.97

Table 3: GP evaluation metrics for each machine and time aggregation. The learning rate was set to 0.1, and the optimization loop consisted of 100 iterations. The metrics in the table represent the lowest and or highest score achieved during the experimentation phase.

As can be seen, time aggregations of 30 minutes typically results in lower MSE, MAPE, RMSE and pinball loss, and a higher ACE. Likewise, time aggregations of 30 minutes takes less time to train than 10 minute aggregations. This is due to the number of data points  $n$  and the time complexity stated in Section 3.4. Furthermore, when the kernel design is more complex (involves more kernels), the amount of seconds to train the model also increases. The paper disposal machine has the best evaluation metrics at both 10 and 30 minute aggregations with OG 1, HVAC, UV EG, and EG, following respectively. The evaluation metrics for the main terminal is an interesting example. Referring to Figure 19 in the appendix, the baseline model predicted a longer duration load profile than what actually happened. Thus, the reason for the poor evaluation metrics and in particular, Pinball loss. However, if more data was collected, it is possible the kernels will learn such a behavior.

## 5.5 Assumptions and Limitations of Models

Before an example on how to use the baseline model for monitoring energy efficiency and performing PDD is presented, the assumptions and limitations of the baseline model are presented here. One of those being the model training period. Ideally, one should have a duration longer than one full production week to analyze and to choose this period from. Indeed, most of the related work in Section 2 had access to several months of data. As previously discussed, Benedetti et al. [12] used one years worth of data to train their energy

baseline model. As a result, they were able to identify three different load profile patterns by analyzing the model’s residuals. The different load profile patterns supports the idea that by considering a longer reporting period, a wider range of production processes and periodicity’s can be modeled. This conclusion does not hold true when considering only one full production week. The work conducted in this thesis was meant to be an exploration / proof of feasibility, and therefore access to more data was limited. Nonetheless, a production line is an heterogeneous environment in which a machine may be capable of producing a variety of goods. Therefore, when training a model on past data, the heuristic is that the future production of goods will also be similar—which may not be a valid conjecture. For example, in a meeting with CLEMAP’s client, the production leader had stated that their production is based on “jobs”, i.e., they produce in batches according to their customer’s needs at a given time.

In addition, a limitation of the current baseline model implementation is that it does not take into account other covariates or production processes. Rather, it is only an extrapolation based on the past. As reviewed in Section 2, additional data such as production output, harmonic distortions or meteorological is often incorporated into the model as a covariate. The additional data may not only provide for more precise forecasts in energy consumption, but also allows for a root cause analysis for when a piece of equipment experiences a decrease in EE. In another example, a meeting with CLEMAP’s client production leader had stated the company plans the next week’s production on the Thursday before. Therefore, there is the possibility to include the production as a covariate for some of the models. However, this information came too late in this thesis to incorporate. Additionally, after speaking with the production leader, it is clear the production processes are complex and can vary from order to order. The heterogeneous production environment can make false positives more likely when changing production environments and other variables are not taken into account.

A further limitation of Gaussian Processes is that they have difficulty in extrapolating when the underlying physical process of a machine displays less evidence of time series components Section 3.1. For example, the Manroland printer in Figure 20 of the appendix displays less evidence of cyclical patterns. Likewise, the underlying function has “kinks” which is very difficult to model as most kernels generate smooth functions as shown in Section 3.3. Therefore, when the time series displays abrupt changes, i.e., kinks or discontinuities in the underlying function, GPs with the standard kernels presented in this thesis perform poorly in extrapolation.

## 6 Monitoring Energy Efficiency and Performance Deviation Detection using the Baseline Model

### 6.1 Evaluating Machine Energy Efficiency and Performance using Control Charts

In this section, it is discussed on how to use the energy baseline model to identify a benchmark reporting period, monitor EE and perform PDD. Building off of [11], the choice of the benchmark reporting period in which the model is trained on is crucial as it provides a reference to which predictions are compared to. In [12], the authors trained their MLR on a period of one year and plotted the instantaneous and cumulative residuals. These two SPC charts allowed the authors to analyze periods of *best* CAS performance indicated by the residuals fluctuating around a mean of zero. Ultimately, the baseline model was then retrained on a time period of 36 days representing the *best* performance. Here, an example of a workflow for choosing the benchmark reporting period for the paper disposal machine is given.

Using the kernel design according to the evaluation metrics in Table 3, the model is trained on 10 minute aggregated data from October 11<sup>th</sup> until October 15<sup>th</sup>. In Figure 14, the top plot visualizes the instantaneous residuals of the in sample model fit, i.e., the difference of the predictions of the baseline model with the actual values of the benchmark reporting period. The bottom plot visualizes the cumulative residuals of the in sample fit.

Where the SPC charts developed in this thesis differ from that in the literature and in industry is that the energy baseline model is a probabilistic model, and therefore, we have access to the uncertainty in our predictions. Where current SPC charts only take into account the point predictions and standard deviation of residuals to define UCL and LCL (as described in Section 2.3), our proposed SPC charts utilizes the posterior distribution to define the control limits. Namely, by using the 95% PI, a probabilistic approach can be used to determine the statistical significance of deviations in performance and change in EE over time.

A red point in the top plot indicates that, at that time point  $t$ , the actual value was greater than  $2\sigma$  away from the mean prediction  $\hat{\mu}_t$  and vice versa. Looking back at the time series plot in Figure 10, one can potentially identify these outlier points. Then, in the CuSum plot, MA trend lines (1hr and 6hr) are plotted to aid the identification of a benchmark reporting period. It can be seen that the slope of the residuals slightly increases until about the end of the day on October 12<sup>th</sup>, then is constant, and finally shows a slight decrease. However, in both charts, the mean of the residuals fluctuates around the zero bound, and thus the justification as using these four days for the benchmark reporting period is upheld.

With a benchmark reporting period now chosen, predictions for every 10 minutes for the next 24 hours are computed. The same SPC charts are then visualized in Figure 15 as

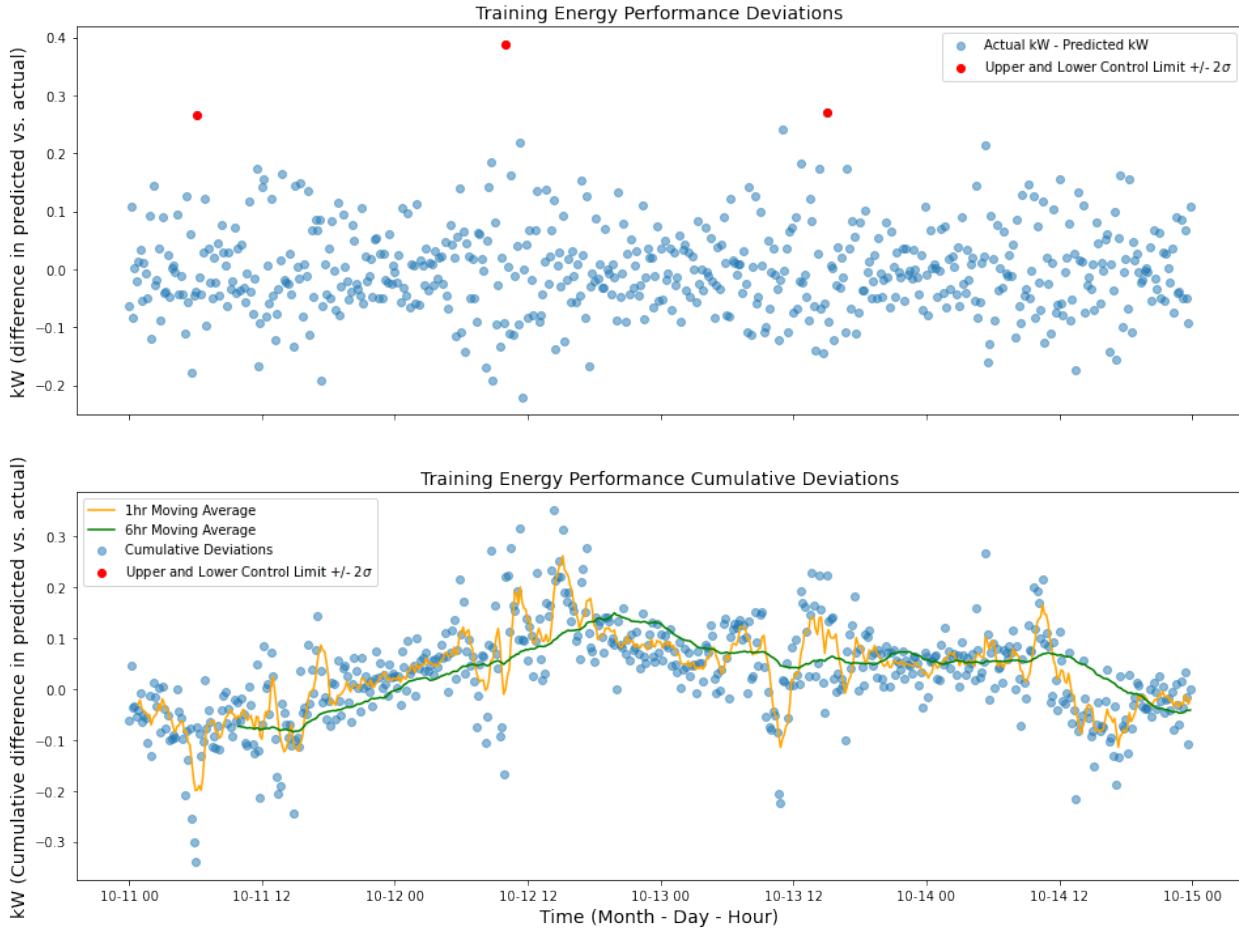


Figure 14: Benchmark reporting (training) period SPC charts.

a hypothetical example to what the operators / production leaders may receive in a real production environment. In the top chart, around 10:30a.m the paper disposal machine was consuming much less energy than what was deemed plausible by the GP model. Indicated by the red dots, these values lie in succession and are  $2\sigma$  away from the expected value. Subsequently, there is a reversion back to what seems to be the nominal operating conditions. The CuSum plot, indicated by the change in slope starting at about 8:30a.m until 11:45a.m, identifies this deviation as a large deviation in machine energy consumption. With the SPC charts having identified a change in EE as a result of a significant decrease in energy consumption, the original time series is plotted again in Figure 16. Indeed, indicated by the red shading, there was a significant decrease in energy consumption.

Furthermore, when a change in EE is identified, the time, machine, actual value, and degree of deviation (indicated by the z-score; how many standard deviations away a value is from the mean) is logged in a database where a historical registry of machine deviations can be built. Operations and maintenance teams can analyze this table to discuss each event, and the more data that is predicted and compared to the benchmark, the more valuable the registry becomes. For example, Figure 17 refers to the deviations that occurred on 15.10.2021, the

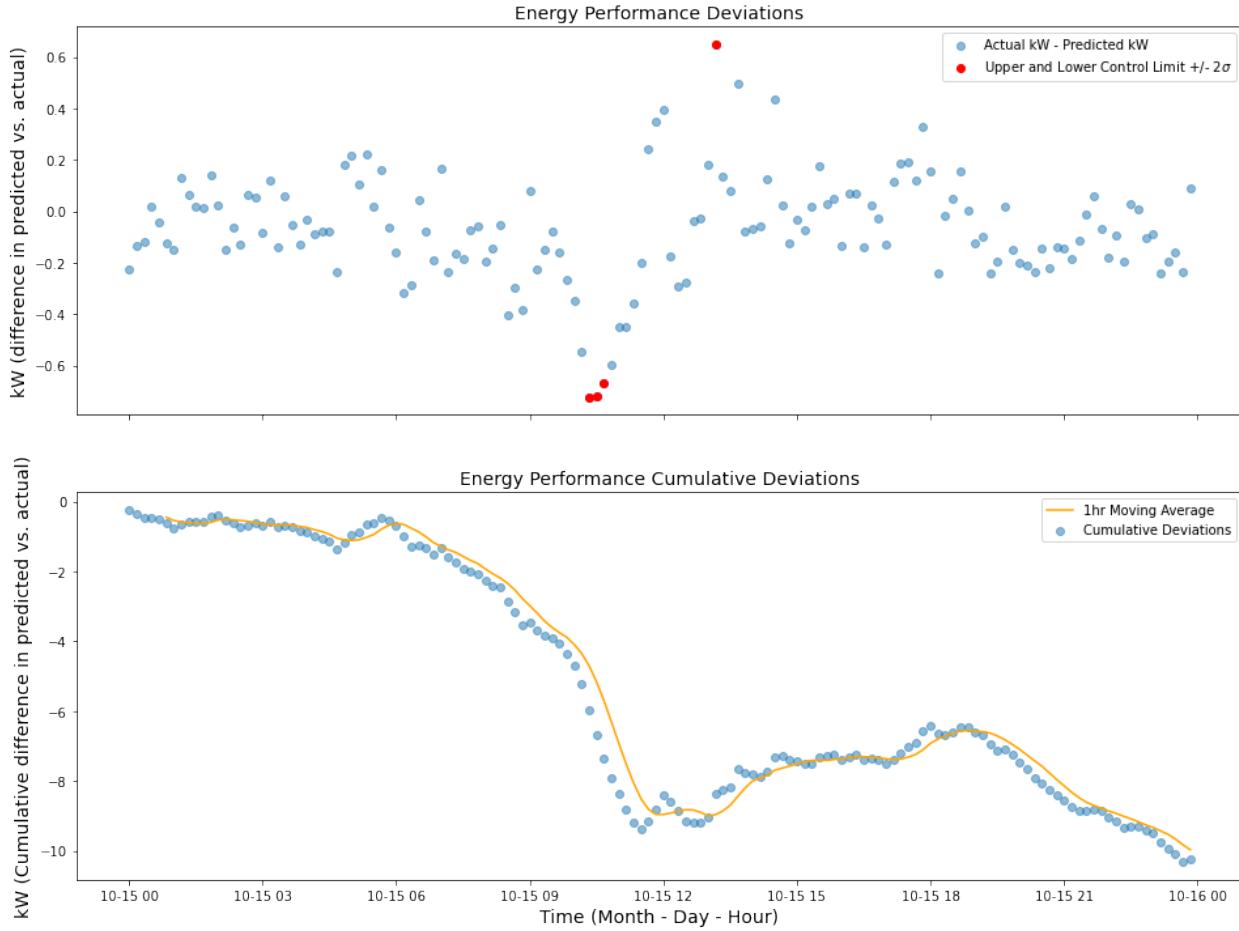


Figure 15: Using the test data, the paper disposal machine SPC charts are visualized. The top plot is the instantaneous residuals whereas the bottom is the CuSum of residuals.

type of machine, and the degree of deviation. To communicate z-scores to a non-technical audience, a color coding scheme is used where red indicates the actual value landed  $3\sigma$  away, yellow is  $2\sigma$ , and green is  $1\sigma$ . The arrows indicate in which direction the deviation occurred, i.e., an energy decrease or increase.

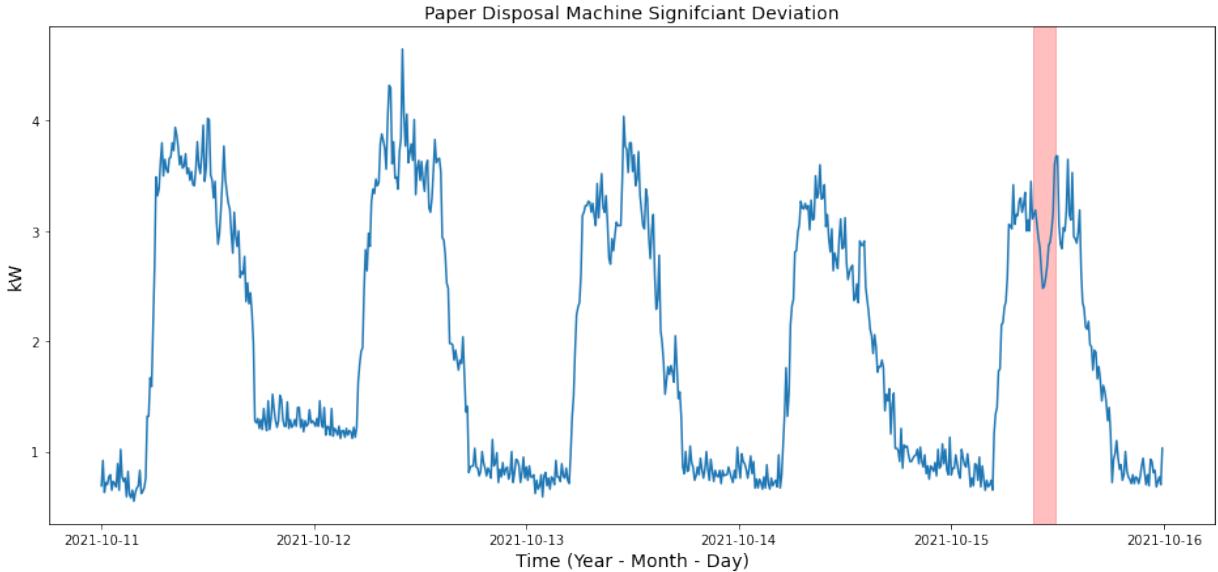


Figure 16: Original time series plotted for further analysis after the SPC charts and GP model identified a significant decrease in energy consumption indicated by the red shaded area.

time	machine	degree
10/15/21 13:10	Paper disposal	● ↑
10/15/21 10:20	Paper disposal	● ↓
10/15/21 10:30	Paper disposal	● ↓
10/15/21 10:40	Paper disposal	● ↓

Figure 17: Historical registry of paper disposal machine performance deviations.

## 7 Model Deployment Preparation

Per the experimentation workflow in Figure 12, the final step is to save the model’s parameters (state) to a .pth file. Within this file is the full “raw” state of the learned GPyTorch model parameters. Saving this to a file allows one to simply load the model’s state back into an ExactGP module and perform inference without the need for any further training. To prepare the GPyTorch model for a prototype deployment in CLEMAP’s environment and infrastructure, the focus will be on developing a Docker container for the *inference* phase which consists only of the ExactGP module, model state, and kernel and data utilities. In the sections below, aspects of machine learning deployment, in particular model preparation and Docker containers, as well as how a Docker container is developed is presented.

## 7.1 Docker for Machine Learning Deployment

In order to maximize business value, the deployment of new systems into a production environment must be done smoothly and at low risk. Continuous integration and delivery (CI/CD) pipelines are a set of modern software engineering best practices to achieve the goal of releasing applications more often and faster, while also better controlling quality, risk, and costs [29]. CI/CD also applies to machine learning deployment workflows and is a critical part of the “DevOps” side of machine learning, also known as “MLOps”. An example of a “MLOps” deployment process could contain the following steps:

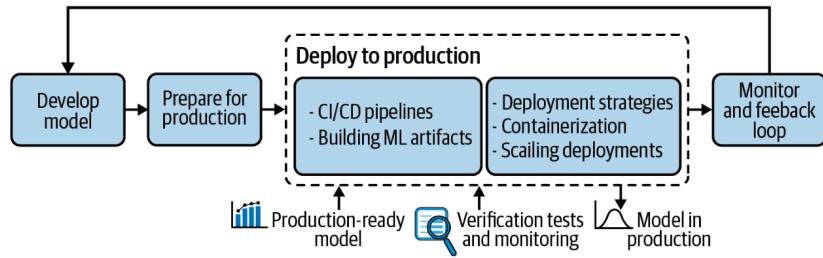


Figure 18: Hypothetical “MLOps” deployment process. Figure taken from [29].

With the energy baseline models now developed, the focus will be on the *prepare for production* phase. To prepare a model for deployment means to manage the version of a model such that an exact description of the environment (including, for example, all the Python libraries used as well as their versions, the system dependencies that need to be installed, etc.). In addition to this metadata, deployment of a model to production should automatically and reliably rebuild the environment on the target machine [29].

To achieve these goals, containerization technology is used. This technology bundles the application together with all of its related configuration files, libraries, and dependencies that are required to run across different environments [29]. The specific software used for containerization in this thesis is Docker. Docker<sup>3</sup> is an open source platform that offers container technologies, lightweight alternatives to a virtual machine (VM), allowing applications to be deployed in independent, self-contained environments, matching the exact requirements of the environment that the model was developed on [29]. Containerization of models is becoming a popular solution to the difficulties of dependencies when deploying machine learning models.

Additionally, there are two ways to approach model deployment in the *deploy to production* phase of Figure 18. Namely,

- Batch learning: The whole data set is available before training starts. For example, daily or weekly scheduled jobs / re-training of models.

---

<sup>3</sup><https://www.docker.com/>

- Online learning: The data set arrives sequentially in an unbounded stream. For example, a model predicts and recursively updates for each data point.

The models in this thesis were developed using the batch learning architecture as the entire data set was available before training. Although there are two ways to approach model deployment, the problem of model versioning, dependencies, and meta data still persists. Therefore, containerization technology is still utilized for both batch and online learning.

Putting the inference phase into a container also allows for the orchestration of other containers in CLEMAP’s environment. For example, their extract transform and load (ETL) process is in a container, and by also having an inference container, these two can be orchestrated together such that the inference container is orchestrated (“runs”) with the ETL container.

## 7.2 Developing a Docker Container

There are three main components, which are developed successively, to build a Docker container:

1. Dockerfile: a blueprint for building images.
2. Docker image: a template for running containers.
3. Docker container: the container itself, which is the packaged piece of software / application.

In the Dockerfile, is a set of commands such as installation of python requirements from the `requirements.txt` file and which file to run. In the main branch of the GitHub repository, this is the `__main__.py` file in the `src` directory. After defining the Dockerfile, a Docker image, a lightweight standalone executable package of software that includes everything needed to run the application, is built. Upon successful building of the Docker image, the container can be ran, which will run the `__main__.py` file. Subsequently, there is the possibility of pushing the Docker container to Docker Hub, a platform similar to GitHub where repositories can be created to manage containers, to be available for use by the CLEMAP team. However, in practice, many organizations do not do this due to privacy and security concerns.

## 8 Conclusion

### 8.1 Summary of Main Results

Benchmarking energy consumption and identifying deviations using an energy baseline model can be a scalable framework that can also be adapted to different machines in an industrial setting (as shown in Section 2). The specific method chosen for modeling in this thesis, namely Gaussian Processes, is a powerful non-parametric way for modeling non-linear time series. The construction of kernels allows one to model a wide variety of processes. Furthermore, using probabilistic methods allows for the communication of uncertainty in the SPC charts and forecasted energy consumption using the posterior predictive distribution. Whereas more commonly in the literature, single point predictions are used which doesn't allow for analyzing the prediction intervals.

Using the metering data provided by CLEMAP, an energy consumption benchmark is established by training a Gaussian Process energy baseline model with a specially designed kernel composition on a reporting period that represents the “best operating conditions” for a piece of equipment and where the residuals fluctuate around a mean of zero. From there, using the posterior predictive distribution and SPC methodology, the EE is monitored and analyzed to identify when a machine is deviating away from the expected behavior. Thus, not only is a probabilistic energy baseline model useful in benchmarking energy consumption, but it also provides value to the equipment operator in identifying a significant deviation in EE. As a proof of concept, an example was shown using data from the paper disposal machine. Using the methods proposed, the baseline model and SPC charts identified a large decrease in energy consumption.

### 8.2 Discussion with CLEMAP’s Client

The proof of concept was presented to the production leader and technician. They had stated they were not too sure what could have caused such a significant decrease, but it could be related to the Manroland printer (R707LV), Heidelberg Speedmaster (XL106) printer, or the steel folding machine. Furthermore, when presented the time series plots, they were surprised that the machine was cycling over night, when in fact the machine should not be consuming energy.

Additionally, the results of the energy baseline models were achieved with only time as the input. Referring to Section 2, additional data is often used to improve the predictive accuracy of the model and to provide additional insights other variables that affect EE. Data is typically compiled through two forms: (1) additional sensors, and (2) non-sensor based. In the discussion with CLEMAP’s client, the production leader schedules the next week’s production on the Thursday before—this represents the amount of goods the company expects to produce. The production leader is interested in using this data—combined with

the energy data—to produce productivity metrics. Likewise, this production data could be used as an input, in addition to the time input, in the GP to improve the quality of the energy baseline model.

### 8.3 Reflection on Research Questions

Here, a reflection on the full list of research questions outlined in Appendix C is presented. The reflection next to the corresponding number indicates which research question is being reviewed upon.

1. Using only the measurement data provided by CLEMAP, a benchmark of the energy consumption is defined and modeled using an energy baseline model which represents the energy characterization of the “starting situation” before any new production process, equipment component, or energy intervention measure is implemented. As a manufacturing environment is heterogeneous, an energy consumption benchmark should be updated *after* a new production process, energy intervention measure, etc. is implemented. Updating the benchmark after allows one to measure the counterfactual, i.e., what would have happened. After this counterfactual is measured, then the benchmark should be updated. Furthermore, several benchmarks may need to be developed for a piece of equipment *if* that equipment is capable of dual outputs. For example, in the printing industry, it is common for a manufacturer to have a printer that is capable of printing magazines, books, newspapers, etc. Therefore, if the manufacturer schedules these jobs in cohorts, then a benchmark should be estimated for each cohort.
2. As discussed in Section 5.5 and reviewed in Section 2, additional data such as production output, harmonic distortions or meteorological is often incorporated into the model as a covariate. The additional data may not only provide for more precise forecasts in energy consumption, but also allows for a root cause analysis for when a piece of equipment experiences a decrease in EE. That is, when equipment deviates from the expected behavior, the covariates can also be analyzed to determine if there is a correlation with the deviation in EE. Furthermore, production data can allow one to calculate KPIs. The KPIs, in turn, act as a goal for the production staff and incentive the team to meet specific performance metrics.
3. For proof of feasibility, it was determined that a batch setting was the most cost effective way of developing the models. In a batch setting, the entire data set  $D$  is available before training starts. Therefore, this allowed CLEMAP to give us a “data dump” which doesn’t incur additional costs for the company in the form of increased technical support and compute. However, it is possible with CLEMAP’s application programming interface (API) to develop the model in an online setting, i.e., the data arrives sequentially in an unbounded stream. Though, this type of modeling is more demanding in regard to compute power, API requests, and technical skill sets—all resulting in additional costs for CLEMAP

4. In Section 7.2, the open source software, Docker, is described and used to implement a “container” for the GP model which would then be utilized for deployment on CLEMAP’s infrastructure. This container represents the *inference* phase, and allows CLEMAP to incorporate and or develop this container into their production environment as needed.
5. In ensuring the energy baseline model does not overfit and is capable of generalizing to new unseen data, the simple “holdout” method is used. That is, the data is split into two sets, namely the training and validation set. In the example with the paper disposal machine, training was done with data from Monday to Thursday (four days) and validation was performed on Friday (one day). Furthermore, since a gradient based optimization algorithm was used, early stopping is used to stop the optimization process. By analyzing the training and validation loss together, the optimization can be stopped when a divergence occurs between the training and validation loss. This stopping prevents the model from learning too much information about the training set [21].
6. In the preliminary study, it was outlined that there would be a larger focus on the “DevOps” and model deployment of the energy baseline model. However, the direction in the main thesis focused on more of a “condition and monitoring” approach to EEE and PDD. Therefore, no specific conclusion was reached regarding the cost factors associated with the model deployment as a model was never deployed in production on CLEMAP’s infrastructure. However, as a compromise, a Docker container was developed to prepare the model for deployment on CLEMAP’s infrastructure. This technique of containerizing models using Docker has become the standard for scaling and monitoring machine learning deployments.
7. Statistical process control methodologies in the form of charts and statistical measures (instantaneous change and CuSum) are used to perform performance deviation detection. Thus, PDD complements the baseline model in research question one. Furthermore, using the posterior predictive distribution of the Gaussian Process model, it is proposed to use the 95% prediction interval to assess the range of plausible values instead of restricting oneself to only the standard deviation of the point estimates.
8. The SPC charts and PDD registry are shared with the production manager. The charts allow the manager to analyze instantaneous deviations in EE and deviations over time. Building off of the charts, a historical deviation registry (database) is developed to allow for further analysis on the data regarding performance deviations. For example, as more data is compiled in the registry, the maintenance staff can then analyze machine EE by observing sequences of increased energy consumption at certain time periods.

## 8.4 Next Steps

Building off of the work completed in this thesis, the “next steps” to be conducted include:

1. Deploy the energy baseline model in CLEMAP’s development environment using the Docker container developed in Section 7.2. This deployment phase allows one to test the generalizability of the model on new, unseen data and to perform further model refinements.
2. Incorporating the expected amount of produced goods provided by the production leader on prior Thursdays as a covariate in the energy baseline model. This could enhance the predictive accuracy of the baseline model to produce more reliable forecasts and ability in performing PDD.
3. Defining productivity metrics using the baseline model from step two above could be calculated. For example, the production leader schedules the next week’s production on the Thursday before—this represents the amount of goods the company expects to produce. Then, using the energy baseline model, the day ahead energy consumption could be forecasted for each day. Using the production leader’s forecast and the energy baseline model’s energy consumption forecast, productivity metrics such as energy consumed per unit of produced good could be calculated.
4. Building off of step three and following the same framework in the thesis, the difference between the forecasted productivity metrics and the actual productivity metric could be monitored to identify at which process or machine the productivity metric is deviating from the forecast.

## 9 Bibliography

- [1] F. Shrouf and G. Miragliotta, “Energy management based on internet of things: Practices and framework for adoption in production management,” *Journal of Cleaner Production*, vol. 100, pp. 235–246, 2015. DOI: 10.1016/j.jclepro.2015.03.055.
- [2] International Organization for Standardization. “Energy efficiency and renewable energy sources — common international terminology — part 1: Energy efficiency.” (2015), [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:13273:-1:ed-1:v1:en>. (accessed: 08.09.2021).
- [3] W. E. Council. “Energy efficiency: A recipe for success.” (2010), [Online]. Available: [https://www.worldenergy.org/assets/downloads/PUB\\_Energy\\_Efficiency\\_-\\_A\\_Recipe\\_For\\_Success\\_2010\\_WEC.pdf](https://www.worldenergy.org/assets/downloads/PUB_Energy_Efficiency_-_A_Recipe_For_Success_2010_WEC.pdf). (accessed: 04.11.2021).
- [4] V. Tutterow, S. Schultz, M. Corporation, and J. Ygdall, “Making the case for energy metering and monitoring at industrial facilities,” *ACEEE Summer Study on Energy Efficiency in Industry*, pp. 166–174, 2011. [Online]. Available: <https://www.aceee.org/files/proceedings/2011/data/papers/0085-000064.pdf>.
- [5] T. Adenuga, K. Mpofu, and R. I. Boitumelo, “Energy efficiency analysis modelling system for manufacturing in the context of industry 4.0,” *Science Direct*, vol. 80, pp. 735–740, 2019. DOI: <https://doi.org/10.1016/j.procir.2019.01.002>.
- [6] S. Emec, J. Krüger, and G. Seliger, “Online fault-monitoring in machine tools based on energy consumption analysis and non-invasive data acquisition for improved resource-efficiency,” *Science Direct*, vol. 40, pp. 236–243, 2016. DOI: 10.1016/j.procir.2016.01.111.
- [7] K. Efthymiou, N. Papakostas, D. Mourtzis, and G. Chryssolouris, “On a predictive maintenance platform for production systems,” *Science Direct*, vol. 3, pp. 221–226, 2012. DOI: 10.1016/j.procir.2012.07.039.
- [8] S. F. A. Office. “Evaluation of incentive effect of emissions trading scheme federal office for the environment.” (2017), [Online]. Available: <https://www.efk.admin.ch/en/publications/security-and-environment/transport-and-environment/2737-evaluation-of-incentive-effect-of-emissions-trading-scheme-federal-office-for-the-environment.html>. (accessed: 31.08.2021).
- [9] S. Confederation. “Switzerland’s seventh national communication and third biennial report under the unfccc fourth national communication under the kyoto protocol to the unfccc.” (2018), [Online]. Available: [https://unfccc.int/sites/default/files/resource/624078315\\_Switzerland-NC7-BR3-1-CHE\\_NC7\\_BR3\\_2018.pdf](https://unfccc.int/sites/default/files/resource/624078315_Switzerland-NC7-BR3-1-CHE_NC7_BR3_2018.pdf). (accessed 30.08.2021).
- [10] C. Simona, A. Luca, I. Vito, M. Fabrizio, and U. Stefano, “Methodology development for a comprehensive and cost-effective energy management in industrial plants,” in *Energy Management Systems*, G. Kini, Ed., InTech, Aug. 1, 2011, ISBN: 978-953-307-579-2. DOI: 10.5772/20490. [Online]. Available: <http://www.intechopen.com/books/energy-management-systems/methodology-development-for-a->

comprehensive-and-cost-effective-energy-management-in-industrial-plant  
(visited on 05/07/2022).

- [11] J. S. Oakland, *Statistical process control*, 6th ed. Burlington, MA: Butterworth-Heinemann, 2008, 458 pp., ISBN: 978-0-7506-6962-7.
- [12] M. Benedetti, F. Bonfà, V. Intronà, A. Santolamazza, and S. Ubertini, “Real time energy performance control for industrial compressed air systems: Methodology and applications,” *Energies*, vol. 12, no. 20, p. 3935, Oct. 17, 2019, ISSN: 1996-1073. DOI: 10.3390/en12203935. [Online]. Available: <https://www.mdpi.com/1996-1073/12/20/3935> (visited on 05/04/2022).
- [13] D. J. Jason Trager Paul K. Wright, “Tightening the constraints faster: A statistical process control approach to commissioning in buildings.,” *ACEEE*, vol. 11, pp. 321–331, 2014. [Online]. Available: <https://www.aceee.org/files/proceedings/2014/data/papers/11-416.pdf#page=1>.
- [14] J. G. bibinitperiod J. E. Phillip N. Price Mary Ann Piette, “Automated measurement and verification of transactive energy systems, load shape analysis, and consumer engagement,” Lawrence Berkeley National Laboratory, 2014. [Online]. Available: <https://escholarship.org/uc/item/6tr9s31d#main>.
- [15] R.-P. Nikula, M. Ruusunen, and K. Leiviskä, “Data-driven framework for boiler performance monitoring,” *Applied Energy*, vol. 183, pp. 1374–1388, 2016, ISSN: 0306-2619. DOI: <https://doi.org/10.1016/j.apenergy.2016.09.072>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0306261916313745>.
- [16] B. Grillone, G. Mor, S. Danov, J. Cipriano, and A. Sumper, “A data-driven methodology for enhanced measurement and verification of energy efficiency savings in commercial buildings,” *Applied Energy*, vol. 301, p. 117502, 2021. DOI: doi:10.1016/j.apenergy.2021.117502.
- [17] Y. Zhang, C. Bingham, M. Martínez-García, and D. Cox, “Detection of emerging faults on industrial gas turbines using extended gaussian mixture models,” *International Journal of Rotating Machinery*, vol. 2017, e5435794, 2017. DOI: 10.1155/2017/5435794.
- [18] C. Gallagher, K. Leahy, P. O’Donovan, K. Bruton, and D. O. Sullivan, “A cloud computing measurement and verification 2.0 application for automated, near real-time energy savings quantification and performance deviation detection,” *Energy and Buildings*, vol. 185, 2019. DOI: 10.1016/j.enbuild.2018.12.034.
- [19] Z. Du, B. Fan, X. Jin, and J. Chi, “Fault detection and diagnosis for buildings and hvac systems using combined neural networks and subtractive clustering analysis,” *Building and Environment*, vol. 73, pp. 1–11, 2014. DOI: 10.1016/j.buildenv.2013.11.021.
- [20] M. Dettling, *Applied time series analysis*, 2020. [Online]. Available: [https://ethz.ch/content/dam/ethz/special-interest/math/statistics/sfs/Education/Advanced%20Studies%20in%20Applied%20Statistics/course-material-1921-Zeitreihen/ATSA\\_Script\\_v200504.pdf](https://ethz.ch/content/dam/ethz/special-interest/math/statistics/sfs/Education/Advanced%20Studies%20in%20Applied%20Statistics/course-material-1921-Zeitreihen/ATSA_Script_v200504.pdf).

- [21] K. P. Murphy, *Probabilistic Machine Learning: An Introduction*. MIT Press, 2022. [Online]. Available: <https://probml.github.io/pml-book/book1.html>.
- [22] S. Roberts, M. Osborne, M. Ebden, S. Reece, N. Gibson, and S. Aigrain, “Gaussian processes for time-series modelling,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 371, no. 1984, p. 20110550, Feb. 13, 2013, ISSN: 1364-503X, 1471-2962. DOI: 10.1098/rsta.2011.0550. [Online]. Available: <https://royalsocietypublishing.org/doi/10.1098/rsta.2011.0550> (visited on 05/13/2022).
- [23] G. Arpino, H. Shih, B. Wessel, and E. Martins, *Gaussian processes for probabilistic electricity price forecasting*, 2021. [Online]. Available: [https://gabrielarpino.github.io/files/MEC\\_forecast-4.pdf](https://gabrielarpino.github.io/files/MEC_forecast-4.pdf).
- [24] J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson, “Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration,” in *Advances in Neural Information Processing Systems*, 2018.
- [25] J. Gardner, G. Pleiss, K. Q. Weinberger, D. Bindel, and A. G. Wilson, “Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration,” in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018. [Online]. Available: <https://proceedings.neurips.cc/paper/2018/file/27e8e17134dd7083b050476733207ea1-Paper.pdf>.
- [26] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [27] S. Bischof, H. Trittenbach, M. Vollmer, D. Werle, T. Blank, and K. Böhm, “HIPE: An Energy Status Data Set from Industrial Production,” *Proceedings of the Ninth International Conference on Future Energy Systems*, pp. 599–603, 2018. DOI: 10.1145/3208903.3210278.
- [28] *Harmonic solution guide*, 2021. [Online]. Available: [https://www.mtecorp.com/wp-content/uploads/Harmonic-Solution-Guide-06\\_21.pdf](https://www.mtecorp.com/wp-content/uploads/Harmonic-Solution-Guide-06_21.pdf) (visited on 05/24/2022).
- [29] L. Dreyfus-Schmidt, *Introducing MLOps*. S.l.: O'Reilly Media, Inc., 2020, OCLC: 1202550935, ISBN: 978-1-4920-8329-0. [Online]. Available: <https://learning.oreilly.com/library/view/~9781492083283/?ar?orpq&email=%5Eu> (visited on 05/16/2022).

## A Appendix

This appendix contains figures that do not flow and or belong in the main body of the thesis. Figure 19 is the result of a Gaussian Process model fit and predictions for the main terminal. Figure 20 is the production week time series for the Manroland (R707LV) printer.

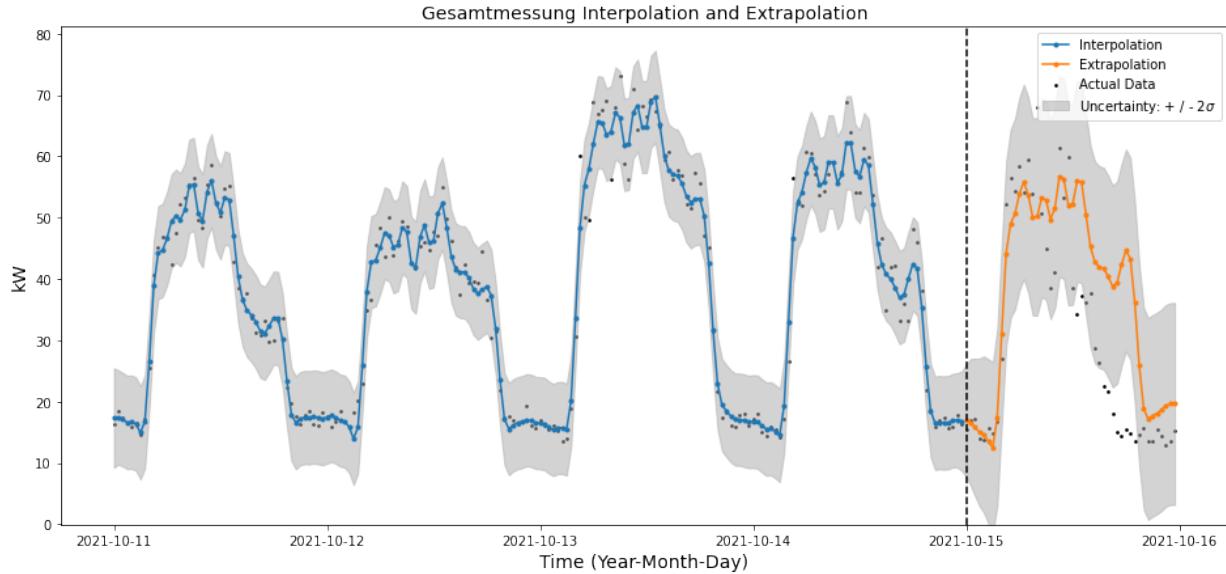


Figure 19: Interpolation and extrapolation for the main terminal GP model at a 30 minute time aggregation. The demand for energy during the afternoon on Friday tapered off much quicker than what the baseline model had predicted and thus is the reason for the poor evaluation metrics.

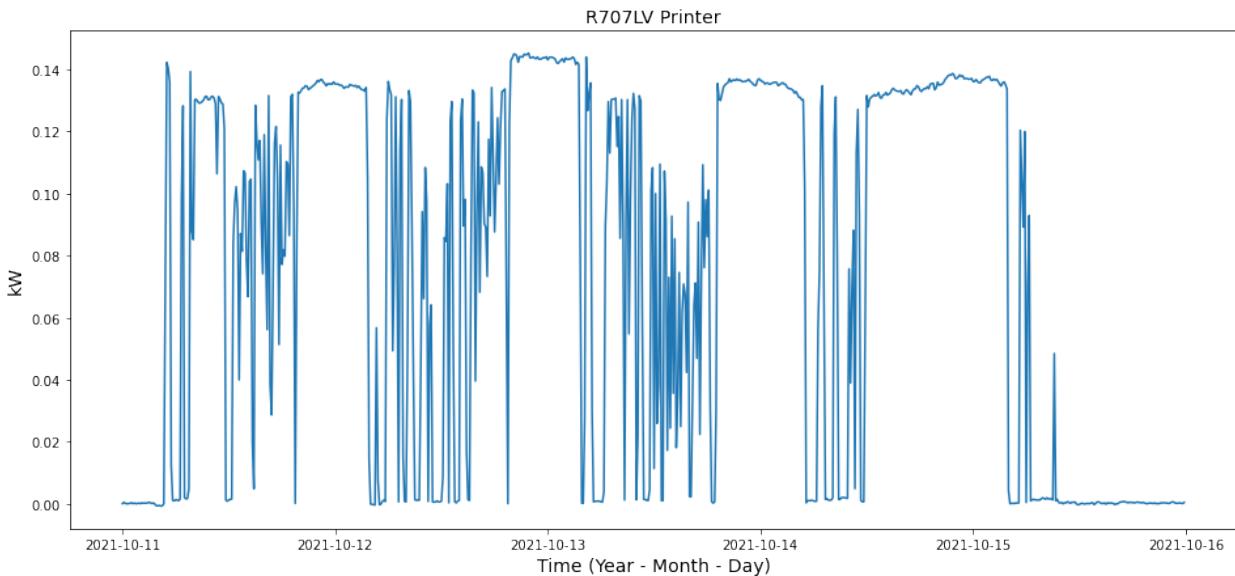


Figure 20: 10 minute frequency of the R707LV printer displays some periodicity but is not identifiable by the ACF correlogram and *Per* kernel.

## B Appendix

The following appendix contains information about the GitHub repository and specific directories within the code base. The main repository can be found at <https://github.com/GStechschulte/energy-efficiency>. The table below outlines the purpose of the main directories and their corresponding branch.

Branch	Directory	Contains
main	src	main Python scripts for running experiments, SPC charts, and utility functions
experiments	eda	CLEMAP and HIPE data set exploratory data analysis
experiments	gp validation	CLEMAP and HIPE Gaussian Process modeling

Table 4: Branches and directories of the GitHub repository developed for the thesis.

## C Appendix

This appendix contains the full list of plausible research questions identified in the preliminary study (the study leading up to the main thesis, i.e., this report).

1. Smart energy meters only provide information in regard to inputs used by a piece of equipment and without access to “output” data from the sensors, how can a baseline EE or benchmark be estimated from energy data alone? Furthermore, how frequently should this benchmark be updated?
2. Could additional data be added or engineered to enhance the solution to research question one such as production data, financial information, temporal aggregations, maintenance logs or weather data?
3. What architecture and tools are best suited for deploying the EEE and PDD models on streaming data?
4. What cloud services and or open-source libraries are needed to productively and cost-effectively deploy the developed EEE and PDD model into production?
5. What regularization techniques are most effective for the EEE and PDD models in ensuring model performance on new, unseen data?
6. Considering scalability and cost, what is the performance of the model deployment work flows and the cost factors associated with model deployment?
7. Upon addressing research question one, can PDD methods, on top of EEE, be implemented to ensure benefits to that particular piece of equipment in the production process life cycle?
8. What information from the results of the algorithms, hypothetically, needs to be transmitted back to the “production manager” in addressing questions one and seven so he/she can make informed production actions?

## **Declaration of Sole Authorship**

“I hereby confirm that I \_\_\_\_\_:

- have written this Thesis independently and without the help of any third party.
- have provided all the sources and cited all the literature I used.
- will protect the confidentiality of the client and respect the copyright regulations of Lucerne University of Applied Sciences and Arts.”