KEY FIGURES TO EDUCATE THE DISTRICT HEATING CUSTOMER

EXPLORING NEW POSSIBILITIES OF BASIC DISTRICT HEATING DATA

by Alexander Nuorimaa

Thesis for the degree of Master of Science
Thesis advisors: Kerstin Sernhed and Axel Johansson

To be presented, with the permission of the Faculty of Engineering of Lund University, for public criticism in meeting at the Department of Energy Sciences on Monday, the 17th of June 2024 at 11:00.

This degree project for the degree of Master of Science in Engineering has been conducted at the Division of Thermal Power Engineering, Department of Energy Sciences, Faculty of Engineering, Lund University.

Supervisor at the Division of Energy Science Engineering was Kerstin Sernhed

Assistant supervisor at the Division of Energy Science Engineering was Axel Johansson.

Examiner at Lund University was Marcus Thern.

The project was carried out in cooperation with the District Heating department of Landskrona Energi AB.

© Alexander Nuorimaa 2024
Department of Energy Sciences
Faculty of Engineering
Lund University

issn: <0282-1990>
LUTMDN/TMHP-24/5572-SE

Typeset in LaTeX
Lund 2024

Sammanfattning

Underlaget för detta examensarbete härstammar från två platser, ett nytt tillägg till Fjärrvärmelagen och Landskrona Energi AB's (LEAB's) önskan om en bättre kundkommunikation. Lagen anger att ett fjärrvärmeföretag ska förse sina kunder med en jämföring av sina nyckeltal mot kunder med liknande konsumptionsprofil. LEAB kommuniserar prisförändringar och uppdaterade nyckeltal årligen till sina industrikunder igenom ett utskick benämnt kundbudgeten. Detta dokument förnyas under arbetets gång för att öka kundnytta och främja bättre kommunikation mellan LEAB och deras kunder.

För att möjliggöra detta utvecklas mjukvara som kan modellera antingen individuella kunder, kundsegment eller hela kundbasen. Utöver detta så kommer mjukvaran användas för att ta fram nya nyckeltal. Då en tydlig mall för hur kunder ska grupperas och jämföras inte har blivit satt än så kommer förslag ges på hur industrin kan följa det nya regelverket.

Resultatet av detta arbete är enligt mig bästa praxis för hur den nya fjärrvärmelagen kan följas för att ge kunder en jämförelse av relevanta nyckeltal mot liknande kunder. En ny kundbudget föreslås där nya och tidigare generaliserade nyckeltal beräknas individuellt och förklaras. Underlaget för både grupperingen och graderingen är de nya nyckeltalen. Möjligheterna med enbart tillgång till basal förbrukningsdata undersöks och därigenom utvecklas innovativa metoder för att normalisera värmeanvändning och för att uppskatta tappvarmvatten.

Detta har gjorts möjligt med begränsade kunnskaper inom programmering i Python, på en bärbar dator. Den mest relevanta koden har bifogats för att kunna minska glappet mellan mjukvaruutvecklare och fjärrvärmeföretagen. Med sina resurser och breda kompetens har de möjlighet att utveckla djupare analyser för kunderna, vilket främjar båda sidor.

Abstract

The basis for this thesis work is derived from two sources, a new Swedish District Heating (DH) law and the internal operation of Landskrona Energi AB (LEAB). The law states that a DH company should present their customers with a comparison of relevant key figures to customers with similar user profiles. LEAB communicates pricing changes, a cost estimation for the coming year and newly calculated key figures with their industry customers yearly with a send out referred to as the customer budget. This document is to be renewed by this thesis, increasing customer value and promoting better communication between LEAB and their customers.

To enable this, a software model will be developed where individual customers, customer segments or the entire customer base can be analysed. Beyond this, the software model will be used to calculate different key figures throughout the thesis work. As the framework for the grouping and comparison profiles are yet to be determined, a suggestion will be made for how the industry can comply.

The results from this thesis is what we believe to be best practise for how to comply with the new Swedish DH law, to give the customers a comparison of relevant key figures against similar customers. A new customer budget is proposed where new and previously standardized key figures are individually estimated and explained. The basis for this customer grouping and labeling are the newly developed key figures. The possibilities with access to only basic consumption data is explored through which innovative methods for heat load normalization and hot tap water estimation, amongst others, are developed.

This has been carried out with limited programming knowledge in Python, with a laptop. The most relevant code has been appended to the thesis to help bridge the gap between software developers and the DH industry. With their resources and wide competence, the DH industry has the possibility to make and provide deeper analyses to their customers, benefiting both sides.

Acknowledgements

I would like to extend a thank you to my supervisors at Lund Uneversity, Axel Johansson and Kerstin Sernhed. Your support has guided me through the almost overwhelming process that is the thesis work.

Mattias Olsson and Eva Klint at Landskrona Energi AB have been very helpful with lending me their time in several meetings. This thesis work would not have been possible without your generosity in providing data for your District Heating grid, thank you!

Contents

List of Figures

List of Tables

Nomenclature

Latin letters

C - constant used for estimating substation delta T
CF - correction factor used for heat load normalization
c_p - specific heat of water
E - heating consumption
e - Euler's constant
e_ind - energy index
HDD - heating degree days
k - shape parameter of the Weibull probability function
ΔT - temperature difference across customer substation
$T_{balance}$ - balance temperature
$T_{outdoor}$ - outdoor temperature
V - volume of district heating carrier water
x - feature (explaining variable)
$\mathbf{X}$ - vector of features
y - observation (actual outcome)
$\mathbf{y}$ - vector of observations
$\hat{y}$ - prediction from model
z - z-score

Greek letters

β - regression coefficients
ϵ - residual from regression
λ - scale parameter of the Weibull probability function
ρ - density
σ - standard deviation

Abbreviations and Terms

DH - District Heating
DWOT - Dimensioning Winter Outdoor Temperature
ML - machine learning
SMHI - Swedish Meteorological and Hydrological Institute
SNI-code - Swedish standard industrial classification code
XGBoost - extreme gradient boosting decision tree ML algorithm

1 Introduction

This introductory section aims to set the background for the thesis. It also explains the issues at hand, defines the scope of the thesis and defines the research questions.

1.1 Background and Context

In 2022, there was an addition to the Swedish DH law (fjärrvärmelagen [19]), mandating that DH customers should have access to a comparison profile where they are compared to similar customers. This requirement necessitates the definition of customer groups and key figures for comparison. This addition came from the legislation EIFS 2022:3 [8]. Although the law has been introduced, as of the writing of this thesis, no strict definitions have been announced yet for how to group customers or what to compare them by.

Landskrona Energi AB (LEAB) is a DH provider in Landskrona, located in southern Sweden, they are currently not equipped with a designated framework for complying with this yet to be determined amendment to the DH law. To communicate with their customers, LEAB utilizes a yearly communication tool known as the customer budget. The customer budget serves multiple purposes: it provides an estimation of the DH costs for the coming year, communicates the new prices and communicates the customer's new heat load signature. This customer budget is provided to their industry customers, who have a more complex pricing model compared to villa customers (the private sector).

1.2 Problem statement

The objective of this thesis is to propose a solution to the problems described above by answering the following three questions:

1. What is valuable to a District Heating customer?
2. What customer value can be generated using basic consumption data from LEAB?
3. Is it possible for LEAB to comply and succeed with the new District Heating law using the available data?

Firstly, to be clear, LEAB is not violating any laws by not having these comparison profiles ready. Even though the law states this, the guidelines for how to implement these procedure is not set yet and the law can not be enforced.

Value for a customer is defined as a satisfied need or wish identified in interview studies where consumers have been questioned about their view on their current relationship to DH and their DH provider. The primary needs identified were to have a better future DH cost estimations to use for budgeting and to better understand the way they are charged.

As this is a study focusing on data processing and exploration, needs correlating to price model structures are omitted due to limitations. Rather, needs which can be satisfied with better communication and enlightenment will be the target. In performing this data exploration, a framework will be developed to comply with the EIFS 2022:3 legislation which states that a DH customer should have access to comparison profile to customers in the same user category. The platform chosen to communicate this comparison profile is the customer budget LEAB provides their industry customers with. For villa customers, similar information could be accessible through LEAB's website.

1.3 Literature review of current research

District heating is a well established industry, recent advancements in data collection and processing have opened new avenues for research. This section provides a concise overview of current research on the needs of DH customers and data driven forecasting of DH consumption.

Zhang et al. carried out a study where they attempted to make long term forecasts of both energy consumption and peak power demand. Even though their study focused on electrical consumption, their results and conclusions are applicable to DH. They found that XGBoost could generate predictions which deviated little from the ground truth. Their methodology included a large data set containing many features such as macro-economic and demographic ones. Using XGBoost for feature extraction they were able to identify relevant features to explain their observed data.

Wang et al. carried out a study where they investigate unsupervised ML to cluster customers by consumption patterns. A strong correlation between outdoor temperature and heat power consumption may mask important underlying behavioural factors according to the authors. They conclude that clustering customers by heat consumption aids utilities in identifying irregularities, and mitigating energy misuse. Moreover, it facilitates consumers to optimize energy usage, thereby reducing expenses [20].

The DH systems of the Nordic countries are well regarded internationally and known for their high standards and reliability [7]. They also boast for a low environmental impact with their high use of renewable energy sources. While this is true, there is a lot of room for improvement. The Swedish DH industry are said to be lacking progress in digitalisation and by the nature of the scale of DH grids in cities, they operate as monopolies on an unregulated market. Many customers feel unable to impact their relation to their DH company and experts think a lack of external forces hold back the development of Swedish DH. Gorroño-Albizu and de Godoy investigate different institutional structures for DH in Sweden and Denmark through interviews with experts within academia and the industry and a literature study. They list DH as an import part of a fossil free future and that there are many cases of countries where the misuse of the market monopoly hinders proper development. Their findings indicate transparency and good customer communication from DH companies are key to develop well functional DH systems in both countries [7].

As for customer communication, Sernhed, Gåverud and Sandgren performed an interview study where they interviewed customers to several DH companies in Sweden. The aim was to investigate the perception DH customers in Sweden had on the increasing complexity of DH price models. They found that amongst the most important points, customers wanted to be able to make a budget for their DH costs. They also wanted to understand the way they were being charged [15].

1.4 Scope of the Study

The available data for this work includes a yearly dataset of heat consumption for 2023. Additionally heating season data for the winter of 2022/2023, which includes data on power, flow, and energy usage.

This thesis focuses on exploring new methodologies. Given the time constraints, these methods are not being tested as thoroughly as would be ideal. While this approach allows for the discovery of potentially innovative solutions, it also means there could be more optimal methods that haven't been fully evaluated. This exploratory phase is crucial for identifying promising directions for future, more rigorous investigations.

1.5 Theory

By explaining key concepts and underlying theory, this segment ensures readers have the necessary foundation to grasp the concepts and discussions that follows. To begin with, a short introduction in DH and relevant concepts for a general DH grid. Then the price model LEAB uses to bill their customers is presented and key figures used will be explained.

1.5.1 District heating

District heating is a widely adopted and is an integral part of Sweden's energy system, contributing significantly to the country's efforts to reduce emissions in the heating sector. It serves as a cornerstone technology, providing reliable and efficient heating solutions to residential, commercial, and industrial sectors. In this section, the reader is introduced to the relevant aspects of district heating. Fredriksen and Werner has written a book about District Heating and Cooling from which this introductory is drawn [6].

The district heating principle

In order to deliver heat to their customers, DH companies use hot water as a carrier in a pressurised grid. The heat within the grid is supplied either by a production plant, which could be a boiler or a heat pump, or it could be excess heat from process industries. There are even geological heat sources for district heating. The energy source for Swedish DH is almost exclusively renewable. At the point of delivery, the hot water from the grid enters a customer substation which amongst other things, contains a heat exchanger that extracts heat from the DH grid into the customers closed heating system.

1.5.2 Environmental data from LEAB's DH production

The values used to calculate emissions for a customers DH usage are taken from the industry organization Energiföretagen (The Energy Companies). The most recent available data is however from 2022, it is available as an excel file [10]. The values used are the following: Greenhouse gas (GHG) emissions from combustion: $56.6\, g\, CO_2\, Eq/kWh$, GHG emissions from transportation of fuel: $3.2\, g\, CO_2\, Eq/kWh$, primary energy factor: 0.05, share of fossil fuels in the energy mix: 0.2 %.

The GHG emissions account for the emissions from transportation and combustion of non renewable fuels, oil in LEAB's case. The primary energy factor shows how much of the energy is derived from fuels what have not been altered by humans. Fossil oil is an example of primary energy and excess industry heat is an example of secondary energy. If purely primary energy would be used and the conversion had en efficiency of 50 % for example, the resulting DH would have a primary energy factor of 2.

The heat power signature

A heat power signature describes the dependency of DH power load on outdoor temperature within a defined system boundary, which could be a specific customer, a customer group, or an entire DH grid. Power consumption is plotted against outdoor temperature, typically resulting in a declining slope as heating needs decrease with increasing temperatures.

Supply and return temperatures

The supply temperature is set by the DH provider and the return temperature is dictated by the efficiency of the heat extracted by the customers. Typical supply temperatures can range from 80-110 °C and even lower for the next generation of DH. Annual Swedish averages for return and supply temperatures are 47 °C and 87 °C respectively [6]. The difference between the supply and the return temperature is called the ΔT.

There are several reasons for a DH company to strive towards a low return temperature, a high ΔT. An important common component of a DH grid which excels with low return temperatures is the flue gas condenser, which extracts heat from the flue gases at the production site. These condensers acts like a heat exchanger between the hot flue gases and the return flow from the DH grid. As the amount of extracted heat from the flue gases is proportional to the temperature difference between the two mediums, a lower return temperature equals a possibility to extract more energy from the flue gases.

1.5.3 Fair District Heating and The Pricing Dialogue

When a new customer is to be added to the grid, there are a few operations with high associated costs. Digging and installing the pipes which connects the customer substation to the grid is expensive. As is the substation in itself, the equipment needed to regulate the flow and heat extraction from the grid and the customers closed loop is intricate. Since DH usually requires customers to make a significant investment when connecting, opting out can be challenging even if DH prices become non competitive compared to other heat sources. This phenomenon has been identified by several organizations within Sweden's building and energy sectors.

To address this issue, two initiatives REKO fjärrvärme (Fair DH) [18] and Prisdialogen (The Pricing Dialogue) [14] were established by these organizations. These initiatives provide a form of branding for DH providers, similar to how food products are ecologically branded. Prisdialogen acts as a bridge between DH companies and their customers, aiming to empower the customers by ensuring that member companies adhere to guidelines designed to create reasonable pricing.

Fair DH seeks to enhance transparency within the DH industry. Members of Fair DH are required to make their products easily comparable to other local heating options, helping potential customers make well-informed decisions.

Fixed rates

As installation of DH has a high cost associated to it, it is commonly shared between a DH company and the customer. All customers in LEAB's DH grid charged a fixed rate, having a non weather dependent income is important for the DH business model to be economically sustainable [15]. The social sustainability of the fixed rate is questioned by the customers, who wants a bill purely reflecting their energy use. One major weakness of such a price model is that it enables customers to install alternative heating solutions weakening DH sales [15].

Flow rate

Charging customers for the flow of DH water through the customer substation is one alternative used in the industry to encourage an efficient grid. As a lower flow yields a lower return temperature for a given energy extraction, it is of high importance to a DH provider to have an efficient grid. By charging their industry customers 3.5 kr/m^3 LEAB creates this incentive. As a comparison, table 1.1 lists a few local DH providers and their flow rates [4] [11] [12] [23].

Table 1.1: Flow rates used by different local DH providers

DH provider	Flow rate [kr/m^3]	Seasons used	Correction factor
Kraftringen	8.70	All seasons	Yes
E.ON	8.36	All seasons	Yes
Öresundskraft	5.21	Winter	No
Landskrona Energi	3.50	Winter	No

The correction factor is used to compensate for lower supply temperatures and in both cases are designed to reduce the flow cost to about 60 % at a supply temperature of 80 °C and starts to regulate the below 100 °C [4] [11]. Just as LEAB, all companies listed only charge their industry customers flow.

1.5.4 The LEAB price model

This section accounts for the structure of LEAB's different price models. At the moment, LEAB has two different price models, one for industry customers and one for non-industry customers. The price model for industry customers in itself has four different categories [12]. This section gives a description of the industry price model and its constituents as this is the more complex price model out of the two. The villa customers have a simpler price model which will be briefly explained at the end of this section. Appendix B provides the document containing the pricing terms of LEAB (in Swedish). Appendix F provides the contract terms for industry customers of LEAB's DH grid (in Swedish).

Heat load signature

The procedure that LEAB uses to set their customers heat load signature is described below in section 1.5.6. Once the heat load signature is determined, the fixed rates are set as a base price and a power rate. The power rate is correlated to the heat load signature. These heat load signature ranges, in kW, with LEAB's corresponding power rates are listed in table 1.2.

Table 1.2: Fixed rates based on heat load signature

Heat load signature [kW]	Fixed rate [kr/year]	Power rate [kr/kW]
0-60	400	788
61-175	8 000	665
176-1400	27 506	540
1400+	119 059	464

Energy rates

Consumed energy has a varying price throughout the seasons, divided into winter, summer and spring/autumn. The reason being that the production costs increase with the heat load demand. Unlike many other commodities, DH does not become cheaper to produce with increasing volumes, for a given production portfolio. Figure 1.3 below depicts the different seasonal energy rates LEAB uses.

Table 1.3: Seasonal energy rates

Season	**Time Period**	**Cost [kr/MWh]**
Autumn/Spring	Apr-May, Sep-Oct	285
Summer	Jun-Aug	100
Winter	Nov-Mar	545

Villa customers

Villa customers are not debited according to the method described above. Instead, they are charged using a simpler pricing model, which is easier to communicate to individuals who are less familiar with energy concepts. This model includes only a fixed yearly rate and a constant energy price that does not change throughout the year. While this thesis primarily addresses the industry pricing model, it will also explore the option of joint pricing models in the discussion.

1.5.5 The customer budget

As mentioned previously, the customer budget is a yearly send out to industry customers containing pricing information and a cost prognosis for the next year. In its current state, the customer budget displays the heat power and heat load signature from the latest heating season. The winter of 2022/2023 in this case. It also lists an estimation of energy usage for 2024 and lastly, a calculation matrix with the price model applied on the customer usage profile. The usage estimation is the same consumption as of the present year. The following page is a recreation of LEAB's current layout of a customer budget, all customer unique data is excluded in this thesis due to GDPR.

Your annual District Heating costs

We have estimated your facility's annual district heating costs for a normal year in accordance with the 2024 price list. The power fee and the fixed fee are based on your facility's power demand at DWOT (Dimensioned Winter Outdoor Temperature, -12 °C) and measurements in your facility during the winter period 2022-2023.

Customer: Unknown

Customer number: unknown

Facility: Unknown

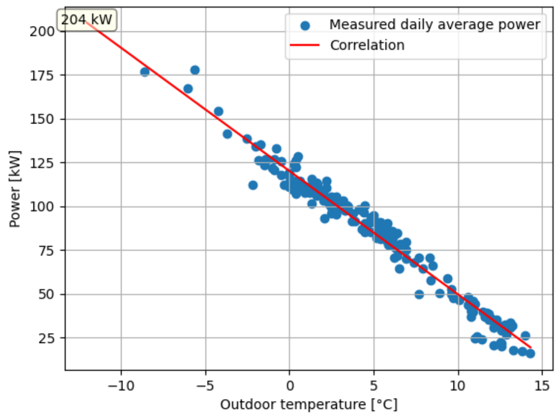
ID: Unknown

Customer type: Unknown

Annual consumption: 529 MWh

Heat load signature: 204 kW

Signature determined by: Linear regression



District heating costs for 2024

Energy price:	Price [kr/MWh]		Estimated consumption [MWh]		Cost [kr]
Winter (jan-mar, nov-dec):	545	x	350	=	190 750
Autumn/spring (apr-may, sep-oct):	285	x	132	=	37 620
Summer (jun-aug):	100	x	47	=	4 700
Sum					233 070

Power rate:	Price [kr/kW]		Power [kW]		Cost [kr]
Power rate:	540	x	204	=	111 780
Fixed rate [kr/year]:	Category		176-1400 kW	=	27 506
Sum					139 286

Flow tax nov-mar:	Price [kr/m³]		Estimated consumption [m³]		Cost [kr]
Flow tax [kr/m³]	3.5	x	5 618	=	19 663
Sum					19 663

Sum: 392 019 kr/year

Average energy price: 741 kr/MWh (VAT excluded)

1.5.6 Customer communication

Lygnerud et al. conducted a study where they evaluate the future business model of DH. In doing so, they concluded that the main result of the study was that digitalisation of the industry and education of its customers is a key future [13]. Optimizing a DH grid is not feasible if its users are not equipped with the knowledge for how to use it properly. Furthermore, in an article where several Swedish DH companies and customers were interviewed, Sernhed, Gåverud and Sandgren recorded that customers felt a greater need to be educated and informed about their price model as the complexity increased [15]. The importance of explaining incentives in the price model is highlighted to help customers understand the desired behaviors. It appears as if both DH companies and their customers have identified a need of better communication between the two.

LEAB have several channels for communicating with their customers. In line with Prisdialogen, they hold annual meetings with larger housing customers to discuss pricing matters and address complaints. They also provide information on 'Mina Sidor', an individual page where information is accessible for a customer. Additionally, the customer budget, which is a topic in this thesis, provides industry customers with yearly information about costs for the coming year.

1.5.7 The heat load signature

The heat load signature LEAB uses is a form of extrapolation to a set value, the Dimensioning Winter Outdoor Temperature (DWOT). the theory for regression and extrapolation follows shortly. The viability of the regression is evaluated. If the resulting linear regression yields poor results, there is a list of options that LEAB follows to set the heat load. Using adjusted historical data, key figures for similar customers or lastly, highest measured historical peaks. In this thesis, it will either be set by linear regression or by the highest recorded daily average power.

In figure 1.1, the heat power signature of a customer on LEAB's DH grid can be seen as the blue dots. The Swedish agency Boverket commissioned SMHI in 2016 to make a chart over DWOT's for different regions in Sweden. This chart lists -10.5 °C as the lowest average to expect over a 24 h period in Landskrona [2]. But the choice for this parameter is free to be set for a DH company, as long as they treat all customers equal. LEAB has set their DWOT to a value of -12 °C.

1.5.8 Linear Regression

Linear regression is a statistical model which assumes a linear relationship between a dependent variable and one or more independent variables. Since the relationship rarely is perfectly described with a straight line, there is an error term included, a residual. In the case of heat load signatures, the DH load would be the dependent variable. If the daily average power for observation i is denoted y_i, the average outdoor temperature as x_i and the residual as ϵ_i, a relation for the given point can be formulated [5].

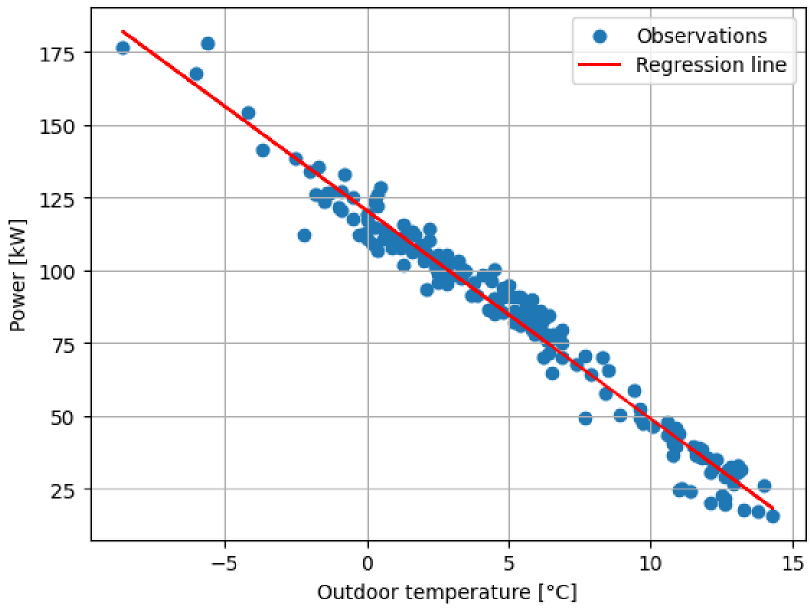


Figure 1.1: Example of a heat power signature for data over heating season. Produced with linear regression

$$y_i = \beta_0 + x_i\,\beta_1 + \epsilon_i \tag{1.1}$$

The term β_0 is included as the *intercept*, which is the value of y when $x = 0$, since there are several occurrences of x and y, the independent and dependent variables can be written as vectors and equation 1.1 can instead be written as follows.

$$\mathbf{y} = \beta_0 + \mathbf{X}\,\beta_1 + \boldsymbol{\epsilon} \tag{1.2}$$

The aim of the model is to set the scalar values β_0 and β_1 such that the sum of the residuals

$$\sum_{i=1}^{N} \epsilon_i = \sum_{i=1}^{N} (y - \hat{y})_i \tag{1.3}$$

is minimized. Where $\hat{y}_i$ is the i-th predicted value. When setting the heat load signature, the temperature x is set to the DWOT as defined by equation 1.4.

$$\hat{y}_{x=DWOT} = \beta_0 + DWOT\,\beta_1 \tag{1.4}$$

Once β_0 and β_1 is trained for a specific customer, it can be used to make predictions. In the context of the heat load signature, the subscribed power would be calculated with equation 1.4 [5].

1.5.9 Model evaluation

When evaluating the models, both the linear regressions and the XGBoost regression models, the coefficient of determination will be used throughout the thesis. Going forward, it will be referred to as R^2. This is according to Chicco, Warrens and Jurman the most descriptive measure of model performance. There are many options of model evaluation which calculates model errors, which could be relevant information. But if a single measure is to be used, they claim that the R^2 score is best suited as it gives a relative measure for how well the observations can be described by the fitted model [3].

1.5.10 Normal year correction

As DH consumption is weather dependent, a normalization is needed to make different years with different temperature distributions comparable. However, there exists a base load for DH that consists of non weather dependent behaviour, such as hot water preparation. This part of the heat load profile needs to be identified and removed before normalizing. A step by step normalization of a DH load is given below in equation 1.5 - 1.7 [17].

$$E_{heat} = E_{tot} - E_{base} \tag{1.5}$$

$$E_{heat,norm} = \frac{E_{heat}}{CF} \tag{1.6}$$

$$E_{tot,norm} = E_{heat,norm} + E_{base} \tag{1.7}$$

E_{heat} is the weather dependent part of the DH load, E_{base} is the identified hot tap water load, which will be referred to as the base load going forward. E_{tot} is the total DH load for the considered period. CF is the correction factor used for normalization. The common method in the DH industry for calculating this correction factor is using heating degree days (HDD). This method is more general in its application as it only considers outdoor temperature and indoor temperatures. But factors like the climate shell of the considered building, wind chills and solar irradiation can affect the heating load at a given outdoor temperature. In order to take factors like these into consideration, an energy index can be used instead. An explanation for the two options with pros and cons is given below.

The process of identifying E_{base} is described in greater detail in subsection 1.5.13.

Heating degree days - The HDD number is calculated during the heating season, it is the difference between the daily average outdoor temperature and the balance temperature. The reason for using a balance temperature instead of the actual desired one is that there are miscellaneous heat sources such as solar radiation and electrical appliances.

$$HDD = \sum_{i=1}^{N} T_{balance} - T_{outdoor,i} \tag{1.8}$$

Equation 1.8 describes how to calculate the HDD number, N is the total number of days during the heating season. LEAB uses 17°C for $T_{balance}$.

Energy index - The Swedish Meteorological and Hydrological Institute (SMHI) has developed an energy index which considers several factors for calculating the correction factor CF. These are the type of building, i.e. office building, multi residential building, villa. The year of construction, which building standard was used and a characterisation of the buildings climate shell. This characterisation takes into account the number of window panes and the proportion of windows to the total wall area and the ventilation technology used. Depending of the use of the building, there are different balance temperatures, for example a multi residential building constructed in the 1980's has an balance temperature of 15°C whereas one constructed in the 2010's could have a balance temperature of 13°C. The reason being that building standards have change and the climate shell as well.

Correction factor *CF* - When calculating CF, the procedure is similar when using either HDD or energy index. In order to use CF as in equation 1.6, it is calculated with HDD's as follows.

$$CF_i = \frac{HDD_{i,actual}}{HDD_{i,norm}} \tag{1.9}$$

Or in the case of using energy index for normalizing.

$$CF_i = \frac{e\_ind_{i,actual}}{e\_ind_{i,norm}} \tag{1.10}$$

1.5.11 XGBoost

XGBoost is a powerful machine learning algorithm used for predictive modeling. It combines multiple decision trees to form a strong overall model. These decision trees, called weak learners, individually have limited predictive power, but together they create a robust model. The process, known as gradient boosting, involves adding trees one by one, with each new tree correcting the errors made by the previous ones. XGBoost uses a learning rate to control the contribution of each tree, helping to prevent over fitting and improve generalization. It also includes regularization techniques to avoid overly complex models. Designed for efficiency, XGBoost can handle large datasets [9]. Zhang et al. implemented a few different configurations of XGBoost to make monthly predictions over several years. Predictions for electrical load and peak power in their case. They did find that XGBoost was an adequate algorithm for long-term load predictions [22]. Throughout this thesis, the XGBoost Regressor model will be used.

1.5.12 Normalization by machine learning

An alternative method for heat load normalization is investigated in this thesis, with using ML. By learning daily heat load patterns an algorithm is able to make predictions for how a customer would behave during a normal year.

1.5.13 Base heat load identification

Equation 1.5 states the need to identification of a base heat load if a meaningful normalisation of heat usage should be made. This thesis uses an experimental method which dynamically sets an individual balance temperature for each customer. The balance temperature is then used as a threshold temperature for when consumption is assumed to be non weather dependent, which would be the base load.

1.5.14 Hotter and colder years

By using the normal year temperatures, a hotter and colder year will be emulated. These values are derived from the recordings of yearly average temperatures of SMHI [16]. Figure 1.2 shows a plot of yearly average temperatures, a rolling yearly average and the difference between those.

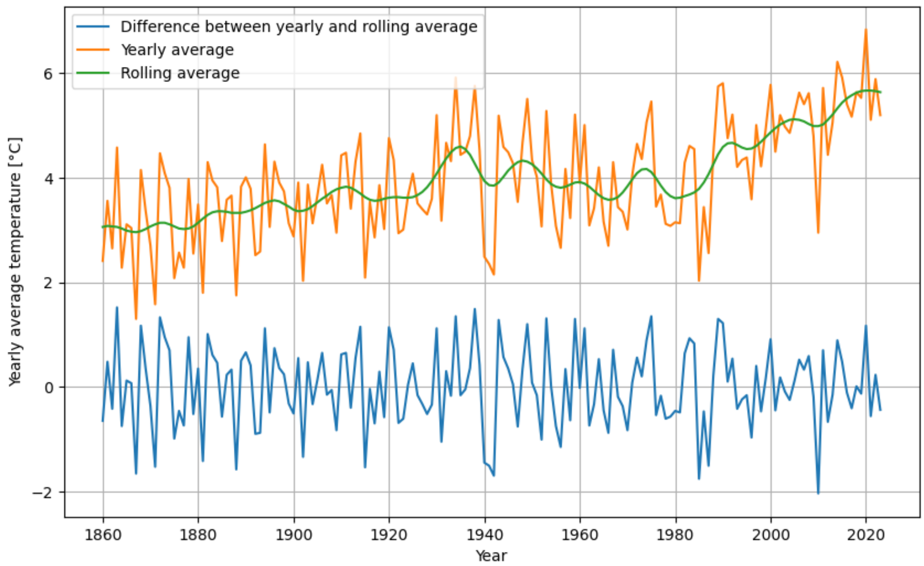


Figure 1.2: Yearly average temperatures of Sweden

1.5.15 Data

LEAB has contributed with several datasets for the execution of this thesis, this section gives a description of the contents of each dataset.

Heating season data winter 2022/2023

LEAB provided daily averages for the heating season October 2022 to Mars 2023 for their entire customer base of 2 854 customers, where 864 of them are industry customers. For each measurement, an ID was provided in order to separate different customers. When making the heat load calculations throughout the thesis, this dataset is used. The contents of table 1.4 gives an overview of the data structure in the heating season data from 2022/2023, all values are given as daily averages.

Table 1.4: Heating season data winter 2022/2023

Feature	**Energy usage**	**Flow**	$\mathbf{T_{Outdoor}}$	**Power**	**Specific flow**
Unit	kWh	m^3/h	°C	kWh/h	m^3/kWh

Heat consumption 2023

Another dataset of daily averages for heat consumption for the entire year of 2023 for the complete customer base.

Meta data

Complementary meta data including consumer code, number of apartments, building area and an SNI-code which is the Swedish standard industrial classification of enterprise activities has been appended as well. Inspecting the meta data, it is incomplete and no meaningful conclusions would be able to be drawn from analysing it.

Outdoor temperature

LEAB takes hourly temperature readings at a station located near one of their production plants in the harbour in Landskrona. An Excel file with hourly averages for 2023 has been shared for this thesis.

Normal year data

In order to create a prognosis for their customer budgets, LEAB use a dataset of normal year temperature data. The data has a daily resolution and includes the data points displayed in table 1.5.

Table 1.5: Available normal year data from LEAB

Feature	T_{actual}	T_{norm}	HDD_{actual}	HDD_{norm}	e_ind_{actual}	e_ind_{norm}
Unit	°C	°C	degree days	degree days	-	-

Heat load signatures by LEAB

The heat load signatures LEAB generated 2024 are available for comparison of those calculated in the thesis.

Cleaning data

For most real life data, there are errors along the way which can lead to corrupted data. In order to allow meaningful analysis of any sort, these need to be handled.

Outliers will be identified and removed by the use of a statistical measure called the z-score. The z-score is a form of centering and normalization of a distribution [1]. This is achieved by subtracting the mean of the distribution and then dividing by the standard deviation. The general formula is given by equation 1.11.

$$z_i = \frac{x_i - \mu}{\sigma} \tag{1.11}$$

The z-score z_i for a particular observation x_i can in other terms be seen as a measure of the number of standard deviations σ the observation deviates from the mean μ by. If the z-score is applied to a normal distribution, an observation would have a 99 % chance of being within three standard deviations away from the mean. As the data in this thesis is assumed to be mostly dependent on outdoor temperature, a normal distribution would be a poor model.

If a linear regression is performed to model the distribution as a line, the residual can be taken for each observation. A residual is the difference between the actual observation and the predicted outcome, as seen in equation 1.12.

$$r_i = y_i - \hat{y}_i \tag{1.12}$$

The variable x in equation 1.11 is replaced with the residual r_i. Once the z-scores are calculated, outliers can be removed by removing all data with a z-score greater than a threshold value. As 99 % of all residuals could be expected to fall within three deviations from the mean if the residuals follow a normal distribution, the threshold is set to four standard deviations.

Substation temperature difference - The temperature difference across the substation, also known as the ΔT is not measured directly for the period 2022-2023. It can however be estimated as a yearly average with equation 1.13.

$$\Delta T = \sum_{i=1}^{N} \frac{E_i}{V_i} \cdot C \tag{1.13}$$

Or in other terms, the aggregated energy consumption in [MWh] divided by the aggregated flow measured in [m^3] multiplied by the constant C. This constant can be derived by making two assumptions:

1. The DH water going through the customer substation has a constant specific heat $c_p = 4.2\,\mathrm{kJ/(kg\,K)}$, or $1.2 \cdot 10^{-6}$ MWh/(kg K) which could be more appropriate as the delivered energy is given in units of MWh.

2. The DH water going through the customer substation has a constant density $\rho = 1000\,\mathrm{kg/m^3}$.

C is defined with equation 1.14.

$$C = \frac{1}{\rho\, c_p} \quad \left[\frac{\mathrm{m^3}}{\mathrm{kg}} \cdot \frac{\mathrm{kg\,K}}{\mathrm{MWh}}\right] \tag{1.14}$$

A unit analysis can be performed to validate equation 1.13:

$$\frac{\cancel{\mathrm{MWh}}}{\cancel{\mathrm{m^3}}} \cdot \frac{\cancel{\mathrm{m^3}}}{\cancel{\mathrm{kg}}} \cdot \frac{\cancel{\mathrm{kg}}\,\mathrm{K}}{\cancel{\mathrm{MWh}}} = \mathrm{K}$$

Once validated, the constant C is calculated:

$$C = \frac{1}{1.2 \cdot 10^{-6} \cdot 1000} = 861.2$$

1.5.16 Categorizing the customer base

By the nature of having different price models and different categories within the industry price model, LEAB has a categorization with five different groups. These five customer groups will be used throughout the thesis. The non-commercial customers will be referred to as the villa customers. The industry customers are categorized based on their heat load signature as follows:

- **Industry customer group 1**: Heat load signature ranging from 0-60 kW
- **Industry customer group 2**: Heat load signature ranging from 61-175 kW
- **Industry customer group 3**: Heat load signature ranging from 176-1 400 kW
- **Industry customer group 4**: Heat load signature of 1 400+ kW

Customer group characterization and labeling

Regardless of the metrics used to divide a customer group, empirical intervals must be established. Following the standard practice for grading energy efficiency, a scale from A to G is applied. Once a metric is calculated for a customer group, a histogram is created to visualize the distribution for the given metric. After visually examining the histogram, an appropriate distribution model is selected. For the purposes of this thesis, the Weibull distribution is chosen. To be clear, throughout the thesis, a customer group is defined by the heat load signature and a customer label is assigned within a group based on the relative performance measured by the key figure.

Histogram

When making a histogram, data is sorted in to different bins. The width of the bin would represent the range it includes from the observed variable. The height would range from 0-1, which represents the likelihood of an observation falling in to that bin. Making this a good tool for visual determination of distribution model.

Weibull distribution

The general two parameter Weibull probability function is defined by equation 1.15.

$$\frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-(x/\lambda)^k} \tag{1.15}$$

The two parameters for this probability function are:

1. **Scale (λ) -** Sets the width of the distribution, lower value means a taller more narrow distribution. A high value flattens it out with a low peak. As the scale approaches zero it indicates a high concentration around one value for the distribution.

2. **Shape ($\boldsymbol{k}$) -** Dictates the shape of the distribution, a value higher than one shifts the distribution to the left. Lower than one and the distribution is shifted to the right. If it equals one, the distribution is instead an exponential distribution.

And x is the explaining variable, the outdoor temperature in the context of this thesis.

Estimating distribution parameters - When estimating these parameters for the different distributions, Pythons *scipy* library was used.

Probability density function PDF - A PDF models the likelihood of a certain observation having a certain outcome, measured from 0-1. The result is a curve which models a distribution with the estimated parameters and an area equal to one.

2 Method

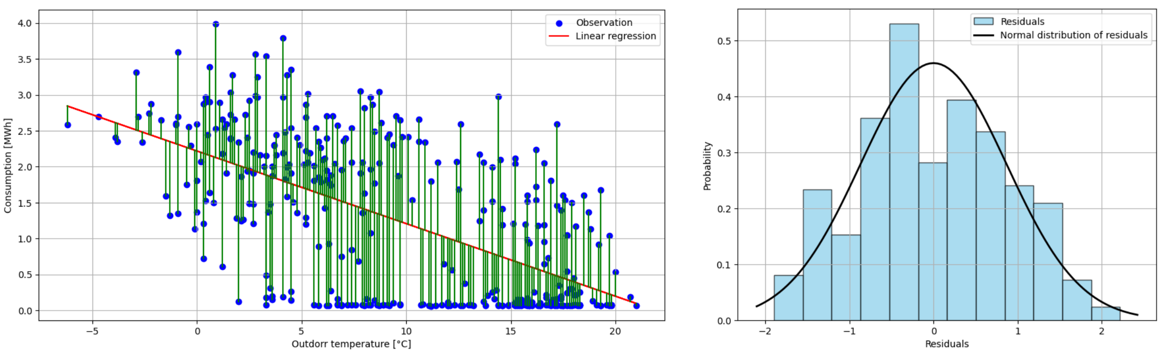
In this chapter, the methodology employed in this thesis is presented, detailing the systematic approach utilized to address the research objectives effectively. Building upon the theoretical framework established in the preceding sections, this chapter outlines the research design, data preparation methods, and analytical techniques employed to investigate the research questions. By providing a transparent and rigorous account of the methodology, this chapter ensures the validity, reliability and repeatability of the study's findings.

2.1 Data

There are several different data sets used throughout the thesis, some of which need to be pre-processed before usage. What follows is an explanation of the measures taken for each data set.

2.1.1 Cleaning data

For both the heating season data winter 22/23 and the consumption data for 2023, the z-score method described in section 1.5.15 was applied to each customer data set. An assumption that the residuals from the regression and the observations followed a normal distribution was made. Figure 2.1a shows a linear regression performed on one customer, the residuals are illustrated as the green lines connecting each observation to the regression line. Figure 2.1b shows a histogram and the normal distribution for said residuals for a customer of heat consumption 2023. All regressions performed for cleaning was done with the outdoor temperature as the explaining variable.



(a) Regression line with residuals

(b) Histogram with normal distribution for residuals

Figure 2.1: A linear regression and the distribution of its residuals

Once the theory is verified, the method for outlier cleaning was applied to all customers separately for both the heating season 22/23 set and the 2023 consumption set. For the heating season set, both the measured power and the flow was cleaned from outliers. The consumption data only included heat consumption which was cleaned from outliers. When a data set was cleaned from outliers, the average R^2 is calculated both before and after removal of outliers as a performance measure.

Both data sets were scanned for duplicate entries. In the case of only one additional measurement for a date, the average of the five prior values was calculated as a comparison. If the five prior measurements did not exist nor belong to the same customer, the five subsequent measurements were examined instead. If those readings belonged to another customer as well, all data for the customer at hand was deleted from the set as they had too few measurements for valid analysis to be performed. A diagram explaining the duplicate removal procedure is shown in figure 2.2. The python code for cleaning the full year data is given in Appendix A.3, the procedure is identical for the heating season data. The difference being that the procedure is performed on bot flow and heat consumption data as both are used from the set.

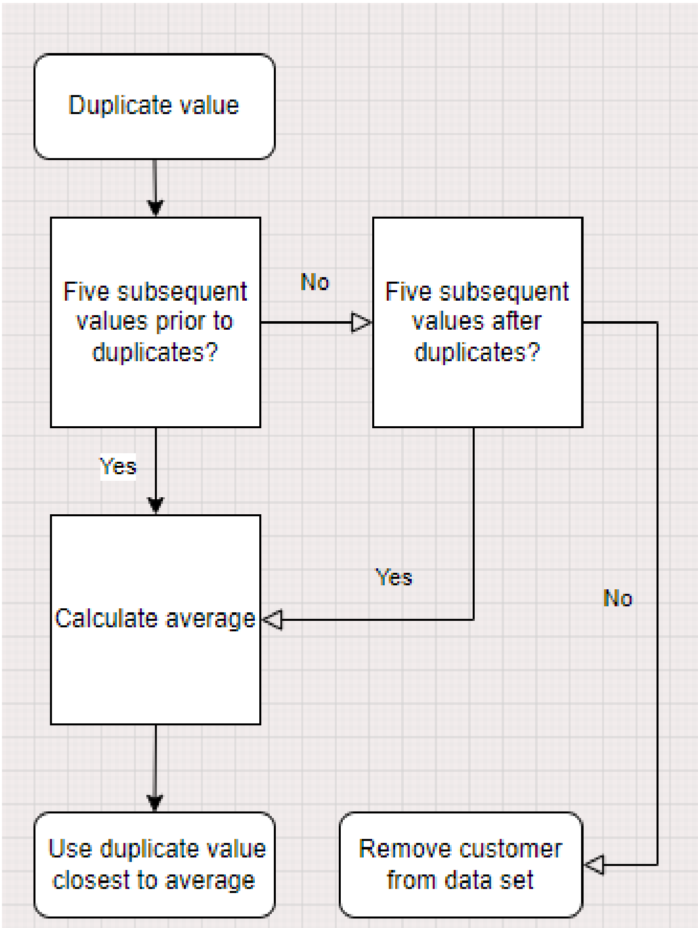


Figure 2.2: An illustration of the process for correcting duplicate values

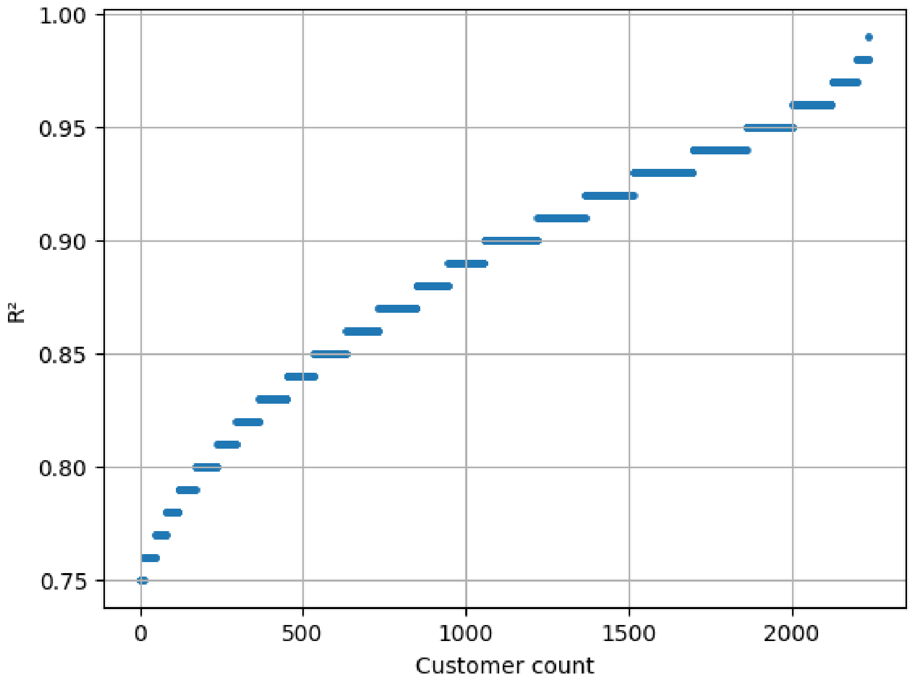


Figure 2.3: Sorted illustration of top 80 % of R^2 for heat load signature correlation

2.2 The customer budget

In order to produce an alternative customer budget for LEAB, the process of creating the original one had to be mapped first. The endeavour of creating this automatically creates a structure for making customer budgets similar to those produced by LEAB today. The *FPDF* library in Python was used to automatically format and insert text and figures in to a PDF. In Appendix A.2, the script for generating customer budgets is provided.

2.2.1 Setting the heat load signature

As this thesis makes cost prognosis for 2024 with consumption data for 2023, the heat load signature would be calculated using heating season data from the winter 23/24. However, the only heating season data available is from the winter 22/23. The method for the heat load signature calculations begins with removing measurements reading zero, with the justification of the heat load signature describing temperature dependent heat consumption.

A linear regression is performed on 80 % of the available data for the customer as a training set. The remaining 20 % is then used for verifying and calculating R^2. A regression with $R^2 > 0.75$ was deemed valid for extrapolation which was the next step. This decision was driven by choosing to include the upper 80 % of the customers heat power signatures. Figure 2.3 shows the calculated R^2 after excluding the worst 20 %. For consistency, the threshold for all regression models was set to 75 %.

The array of explaining variables was changed from the actual outdoor temperature to an array of equal length but starting at the DWOT of -12 °C. The script for calculating the heat load signature and generating the heat power diagram for the heating season is appended in Appendix A.1. For those customers who had a poor correlation between DH power and outdoor temperature, the highest recorded measurement was set as their heat load signature.

Validating the calculated heat load signatures

The calculated heat load signatures were validated by comparing them to the signatures used by LEAB in 2024. This comparison involved calculating the differences between the LEAB-provided signatures and those derived in this thesis. Both methods primarily utilize linear regression, but differences in data cleaning procedures and threshold criteria for model strength may exist. Alternative methods employed by LEAB for weak regression models were not applied in this thesis.

2.2.2 Customer categorization

In practice, a script was written which at first checked to compare the customer identification number with the signatures LEAB provided. If the customer number matched one of those, the heat load signature was evaluated and the customer was assigned the appropriate industry customer group. Those customers who were not matched with this list were automatically assigned to the villa customer group. The short script used for automatically assigning a group for each customer can be seen in Appendix A.8.

2.2.3 Price model

After setting the heat load signatures, a customer had a matrix created with one row for each day of the year 2024. Each column would represent an item of the price model. Looping through the months, the corresponding season of the current month would be evaluated. This allowed the variable energy rate and the flow rate to be appended. The fixed power rate and the subscription fee was divided evenly between all months. In reality, these two fixed rates are actually divided by number of days so it varies slightly between different months.

2.2.4 Balance temperature and base load

As stated by equations 1.5 - 1.7, the base load has to be set for normalization by conventional methods. For each customer that is eligible for the method, the base heat load is calculated as the average heat consumption at temperatures greater than the balance temperature. Validity of the method was defined by linear regression with an $R^2 > 0.75$ for a particular customer. The balance temperature is then calculated by making a curve fit to the heat power signature. The curve fit is made with the *lowess* function from *statsmodel.nonparametric.smoothers_lowess* and the *numpy* library gradient function was used to evaluate the "derivative". In reality, it calculated the slope of the smoothed curve fit.

Once the curve fit was performed, the slope of the curve was calculated for as many points as possible. The greatest slope was identified and as a threshold, the balance temperature was defined as the lowest temperature where the slope had decreased to 30 % of the greatest value. The selection of 30 % as the threshold was done by experimenting with different percentages. It was not uncommon for the heat power signature to level out at the coldest temperatures. In order for this method to not mistake those values for the balance temperature, the first 40 values of the derivative array were excluded from the analysis. Once the derivative was evaluated and the first value to fall below the set threshold was identified, the temperature array could be evaluated at the same index to obtain the balance temperature.

Finally, the consumption data could be filtered to only include measurements at temperatures above the balance temperature and the daily base load could be calculated as the average.

2.2.5 Cost prognosis of the coming year

The budget for the coming year was calculated for each post of the price model from subsection 1.5.4. As a basis for cost calculations, a normalized consumption profile using the consumption data from 2023 was done for each customer.

2.2.6 Heat load normalization

Normalization of the heat load was performed both by energy index, HDD and with ML. When using energy index and HDD, the base load E_{base} was identified and removed. Thereafter equation 1.5 - 1.7 was applied. When normalization was performed by ML, a XGBoost algorithm was trained on the actual data with consumption as a dependent variable and the outdoor temperature as the feature. The algorithm was trained on each customer individually and then asked to predict the usage pattern on the normal year temperature data. This predicted behaviour would then be used as normalized heat consumption. Beyond using the algorithm on the normal year data, it is used on the hotter and colder winter emulation to generate the consumption span which in turn is used for the actual cost prognosis.

Flow data was not included in the data set for 2023, the flow measurements from the heating season 22/23 was used in its place. The flow costs will prove to represent a small portion of the total costs so the substitution is of small significance.

Once the consumption pattern is established for the normal year as well as the hotter and colder scenarios, the price model is applied to it in order to generate the cost prognosis.

2.2.7 Base heat load estimation

The base heat load was defined as the average consumption at temperatures above $T_{balance}$ for each individual customer. This balance temperature is individually set by the methodology described below.

2.2.8 Balance temperature estimation

The heat power signature of the cleaned full year data set was used, the procedure is given in Appendix A.4.

2.2.9 Substation delta T

In order to estimate the ΔT for each customer, equation 1.13 was applied on the heating season 22/23 data set. A histogram was done to illustrate the distribution of $\Delta T's$, a Weibull distribution was deemed an appropriate distribution. A PDF was calculated and plotted on top of the histogram for verification. The total set of $\Delta T's$ was divided by the customer groups and each subset was the same way, with a histogram and a PDF.

2.2.10 Labeling customer groups

Within each customer group, further division was performed by splitting each groups $\Delta T's$ in to seven different subgroups of equal size. A customer would then get an energy performance label from A-G with the A-label being given to those with the highest ΔT. These labels were then used to create an energy performance sticker for each customer. The only exception was the single customer in the 1 400+ kW customer group. The code used to labeling customer groups is given in Appendix A.5.

2.3 The budget proposition

The proposed budget will include the key metrics relevant for customer display, those that could give customers valuable insight in to their performance. The budget prognosis will be provided with a monthly resolution and a clear division in cost posts as well as fixed and variable costs. Scenarios for hot and cold years are included with monthly cost estimations to provide the customer with a span of coming costs and further drive the point of DH consumption's weather dependency. There will be a clear description of all metrics introduced in the budget and how the customer was assigned a group and an energy performance grade. The new budget also includes environmental impacts of the customers DH consumption, making it quantifiable for any future investment analyzes.

3 Results

This section provides a clear and structured presentation of the results in this thesis.

3.1 Data preparation

The different data sets used throughout the thesis had to be examined and cleaned before using, this section accounts for the extent of the cleaning necessary.

3.1.1 The full year data set for 2023

Table 3.1 lists the results of pre-processing the 2023 full year data set.

Description	**Details**
Initial NaN entries for consumption	665 NaN entries removed (over 17 customers)
Total customers after removing NaN entries	2 931 customers
Duplicate entries	3 472 duplicate values (across 2 918 customers)
Customers removed due to insufficient data for duplicate removal	30 customers
Total customers after removing duplicates and insufficient data	2 898 customers
Customers removed due to no consumption recorded	3 customers
Average R^2 for linear regressions	Increased from 0.86 to 0.88
Outliers removed	1 791 entries removed (across 1 018 customers)

Table 3.1: Summary of data pre-processing for 2023 full year data set

3.1.2 The heating season data for winter 2022/2023

The results for pre-processing of the heating season data 2022/2023 is given in table 3.2.

Description	**Details**
Initial data set	No NaN values, 2 duplicate cases
Outliers removed (energy consumption)	957 entries removed (across 628 customers)
Average R^2 (energy consumption)	Increased from 0.81 to 0.84
Outliers removed (flow data)	1,683 entries removed (across 1,177 customers)
Average R^2 (flow data)	Increased from 0.83 to 0.85
Customer count change	Dropped from 2 854 to 2 846 (customers with no recorded consumption data removed)

Table 3.2: Summary of data pre-processing for heating season data set of 2022/2023

3.1.3 Normal year temperature data

The data set containing normal year temperatures was good to use as it was delivered, no preparation was needed.

3.2 Heat load signatures

One calculated heat load signature with its corresponding linear regression is depicted as in figure 3.1. Table 3.3 provides details regarding the actual calculations of the heat load signature.

Description	**Details**
Total valid customer data sets	2 846
Heat load signature set by linear correlation	2 261 customers
Average R^2 for linear regression	0.89
Heat load signature set by highest recorded daily average power	585 customers

Table 3.3: Summary of heat load signature calculations

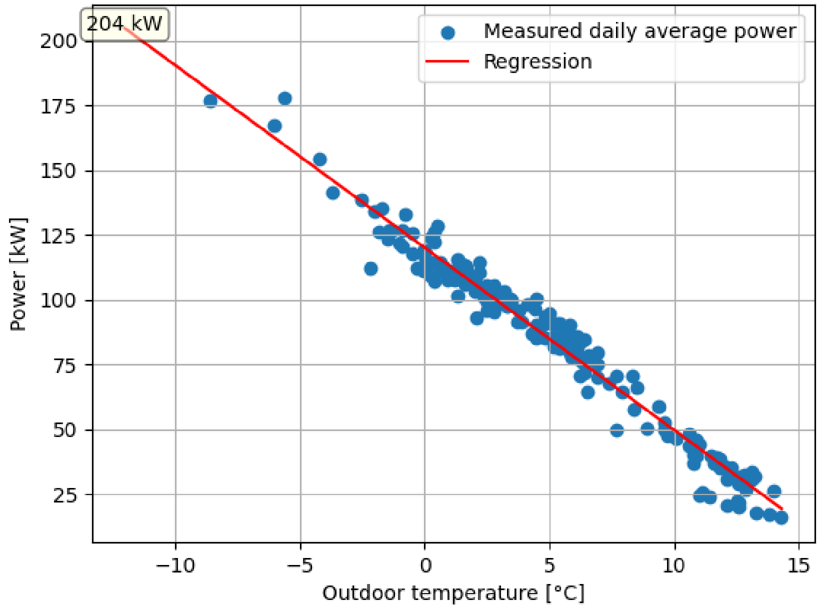


Figure 3.1: Linear regression and interpolated heat load signature at -12 °C

3.2.1 Comparison to LEAB's heat load signatures

In total, a heat load signature was set for 2 846 customers. When comparing to the set of heat load signatures LEAB had previously set. Figure C.1 and C.2 in Appendix C shows the differences of between the heat load signatures set by LEAB and the ones calculated in this thesis when using linear regression and highest daily average measure power respectively. 185 heat load signatures were a perfect match and 41 of them deviated by 10 kW or more from the one LEAB had set. Out of those 41, 31 were set by the highest measured daily average power, meaning that the linear regression had an $R^2 < 0.75$.

3.3 Consumption prognosis and normalization

A consumption prognosis was performed on the normal year temperature data. At first, the model was trained on the consumption and temperature data from 2023. Once trained, the model was given the normal year temperature data. The results from one of the customers can be seen in figure 3.2 where the temperature curves are included.

Figure 3.3 is the result of normalization of the consumption data represented be the blue line in figure 3.2. The three different plots are obtained by using the three different methods HDD, energy index and ML.

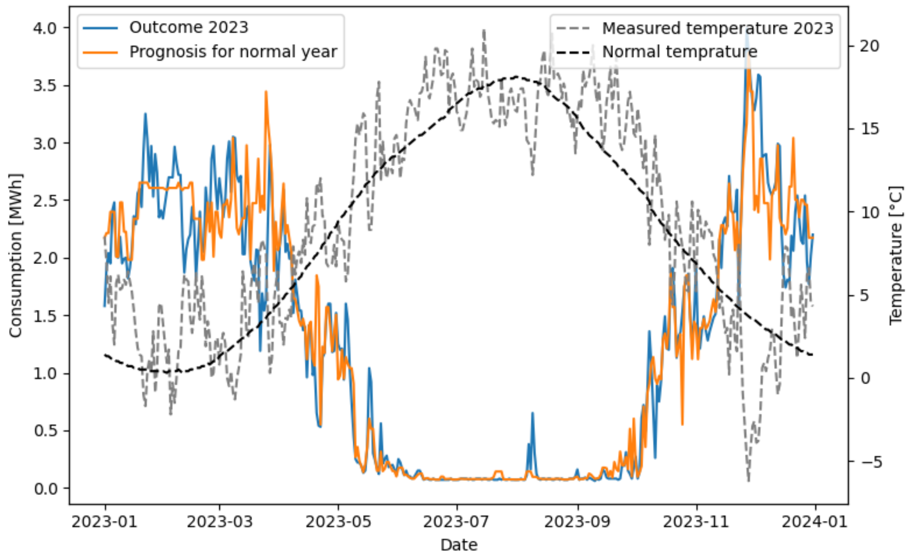


Figure 3.2: Predicted consumption for normal year and actual consumption 2023 for one customer, temperature data included

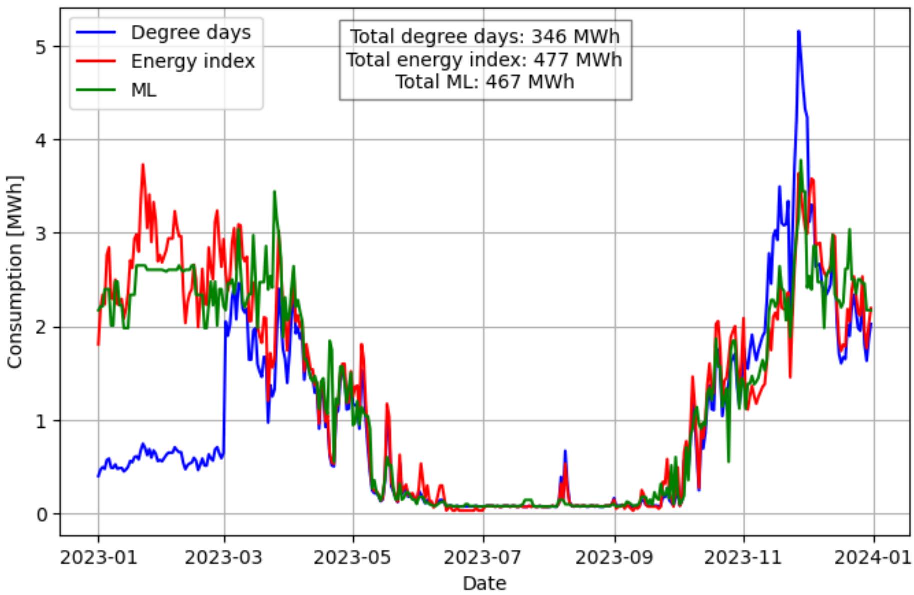


Figure 3.3: Comparison of normalization by ML, degree days and energy index with aggregated heat loads

3.3.1 Balance temperature and base load estimations

In total, the balance temperature was estimated for 2 088 customers and the average calculated balance temperature was 15.7 °C. Figure E.3 displays a division into the customer groups in figure for the estimated balance temperatures used for setting the base heat loads. An example of a balance temperature identification is given in figure 3.4. The lowest estimation for a balance temperature is seen in figure 3.5.

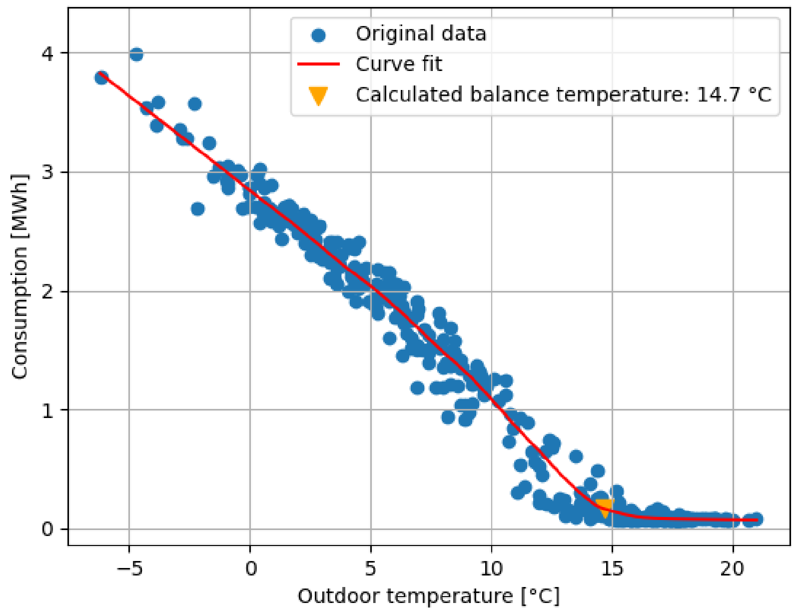


Figure 3.4: Balance temperature from curve fit on consumption data

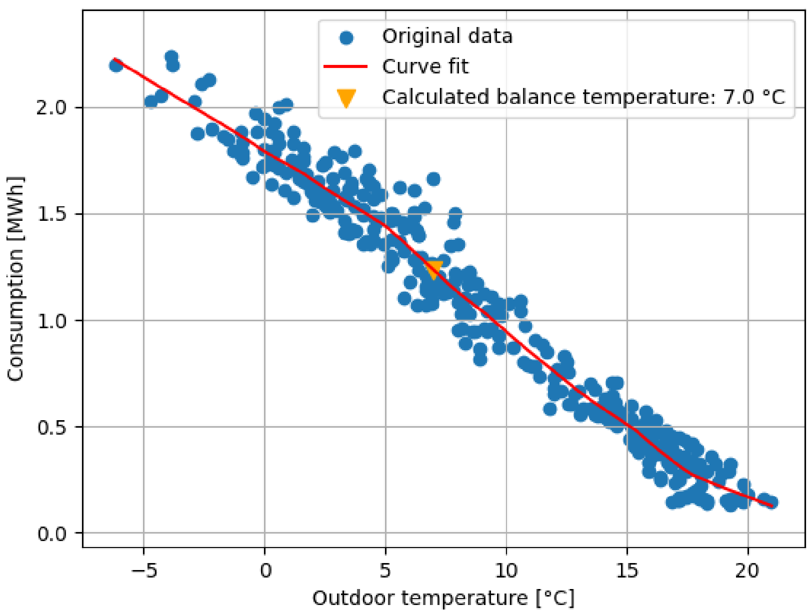
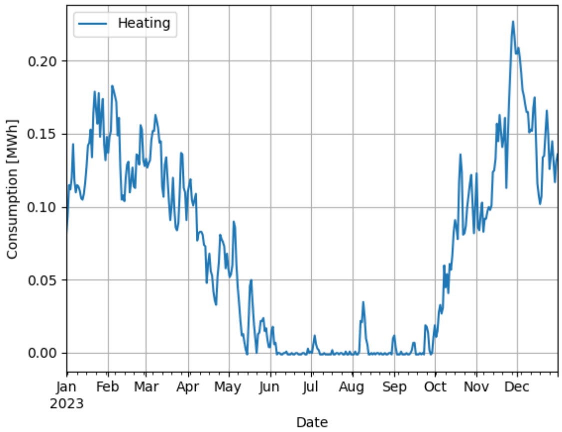


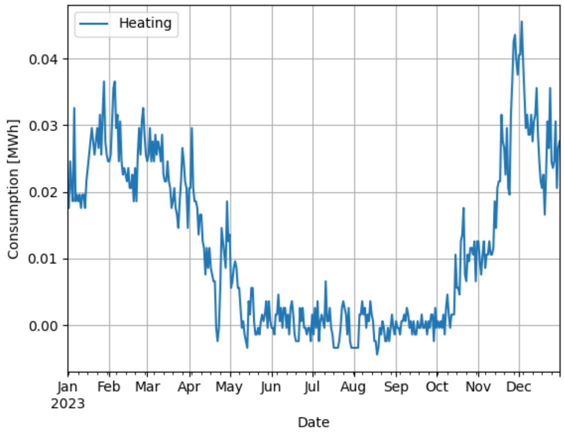
Figure 3.5: The lowest estimated balance temperature

3.3.2 Base load calculation

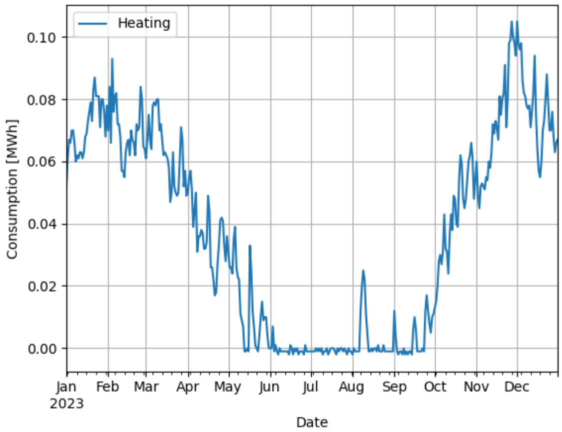
After identifying the balance temperature, the base load is calculated and removed from the consumption data. A selection of four random customers who have had their base load calculated and subtracted is shown in figure 3.6.



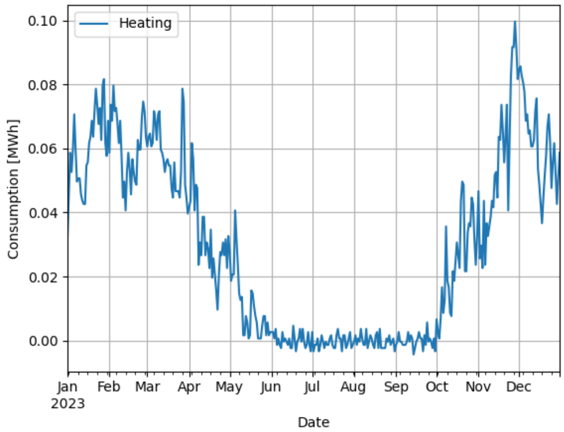
(a) Heating consumption for a randomly selected customer, calculated balance temperature: 15.4°C



(b) Heating consumption for a randomly selected customer, calculated balance temperature: 12.7°C



(c) Heating consumption for a randomly selected customer, calculated balance temperature: 16.1°C

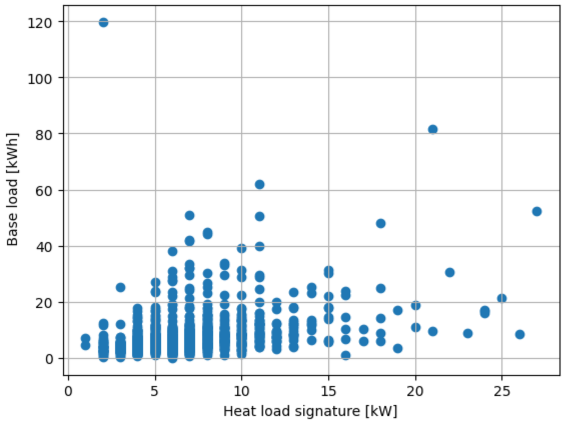


(d) Heating consumption for a randomly selected customer, calculated balance temperature: 16.1°C

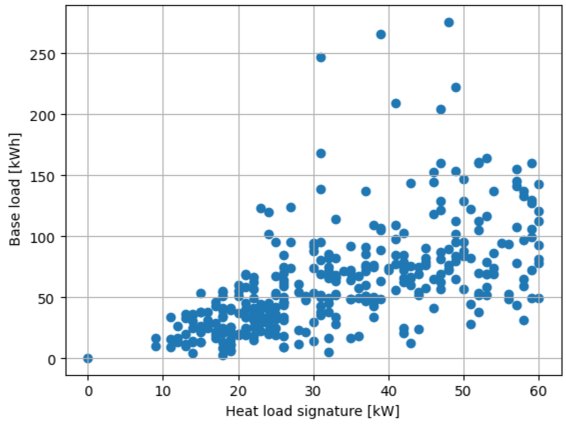
Figure 3.6: A display of the heating consumption for four randomly selected customers with their corresponding balance temperatures

3.3.3 Base load in relation to heat load signature

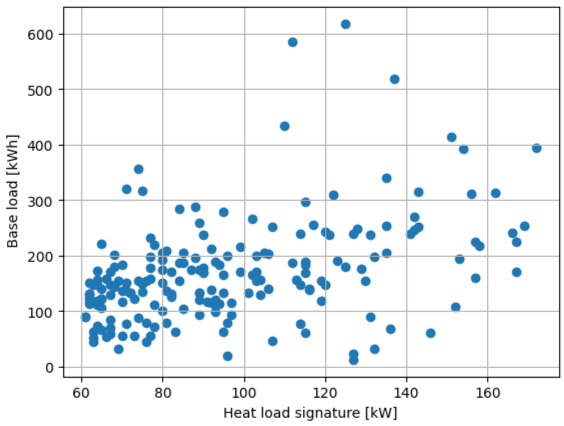
The calculated base loads in relation to the heat load signatures can be seen in figures 3.7a - 3.7d. Estimated aggregated yearly base heat load is plotted against the corresponding heat load signature for each considered customer segment.



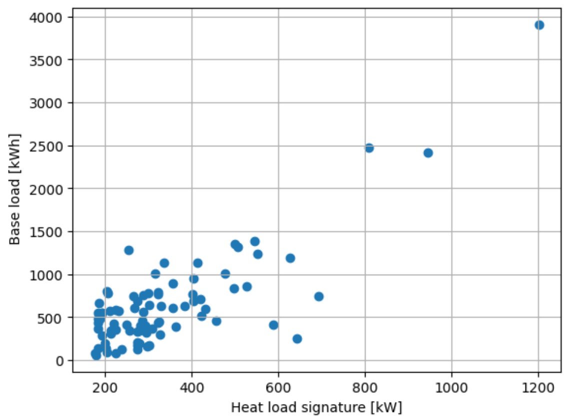
(a) Scatter plot of base loads against subscribed power for villa customers



(b) Scatter plot of base loads against subscribed power for industry group 1



(c) Scatter plot of base loads against subscribed power for industry group 2



(d) Scatter plot of base loads against subscribed power for industry group 3

Figure 3.7: Scatter plots illustrating the relation between estimated base load and calculated subscribed power for the different customer groups

3.3.4 Normal year and scenarios prognosis

Figure 3.8 shows the different temperature distributions for the normal year, a hot year and a cold year.

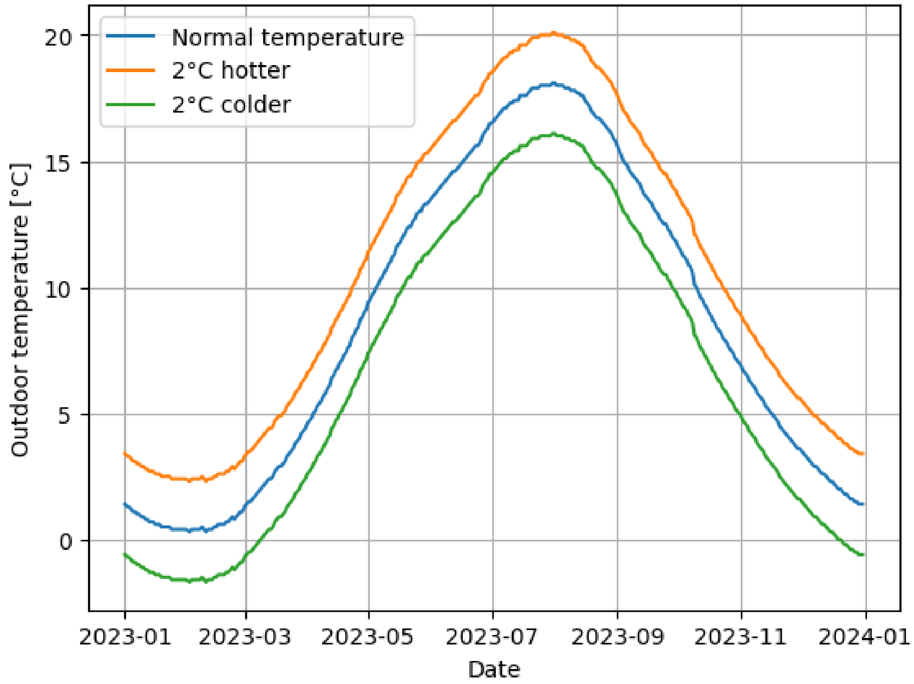


Figure 3.8: Temperature distributions for normal, hot and cold year

The already trained ML model is then given the temperature distributions and produces the consumption profiles seen in figure 3.9.

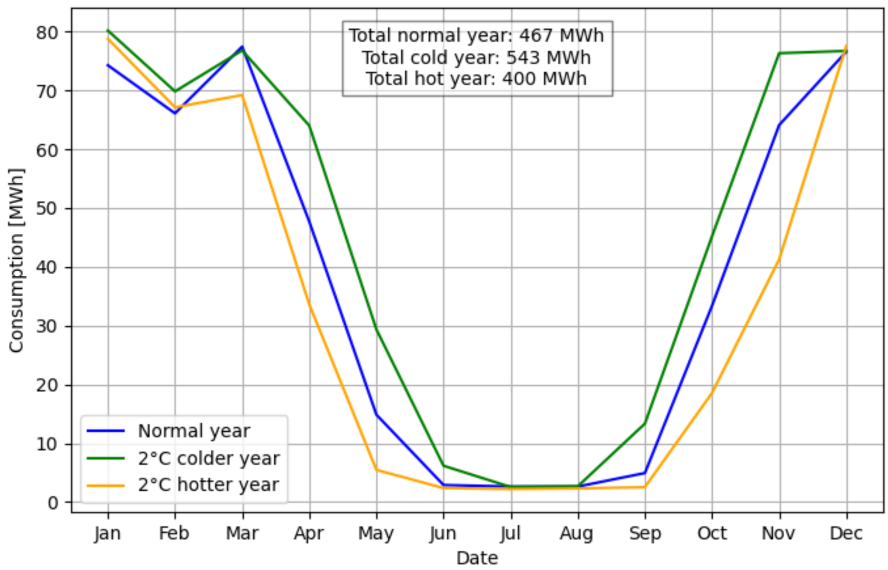


Figure 3.9: Monthly aggregated consumption predictions using ML for normal, hot and cold year

After applying the price model to the consumption profiles, the costs are calculated and are illustrated in figure 3.10.

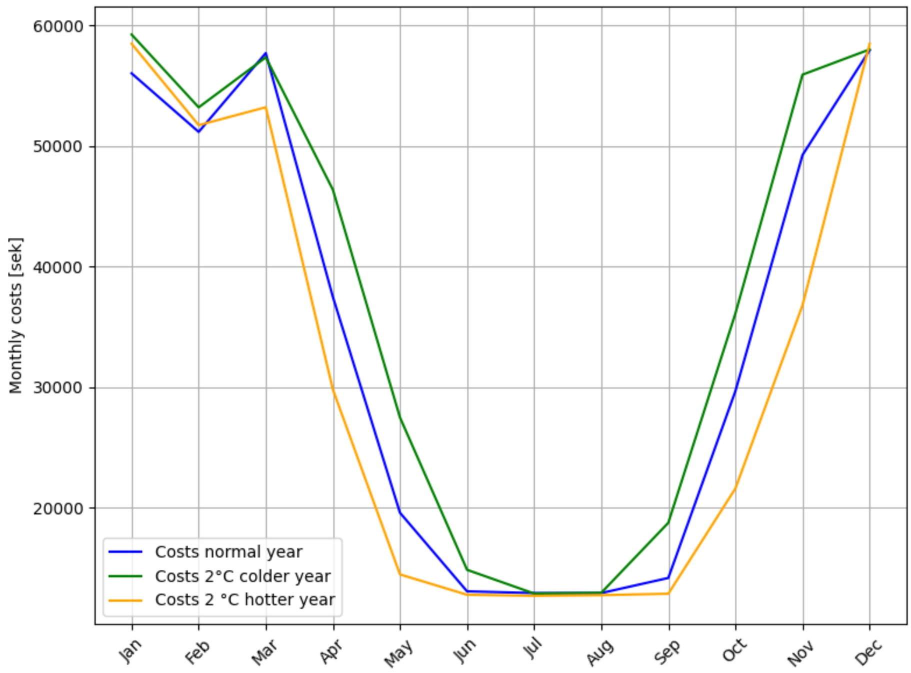


Figure 3.10: Monthly aggregated costs for normal, hot and cold year

When presenting the resulting costs for the normal year in figure 3.10 to the customer, two different methods are used for a more detailed budget. These more detailed budgets are shown in figure 3.11 as a table.

	Fixed fee [kr]	Power cost [kr]	Energy cost [kr]	Flow cost [kr]	Total [kr]
Jan	2,292	9,180	40,484	4,055	56,011
Feb	2,292	9,180	36,029	3,653	51,154
Mar	2,292	9,180	42,216	4,004	57,692
Apr	2,292	9,180	25,997	0	37,469
May	2,292	9,180	8,093	0	19,565
Jun	2,292	9,180	1,570	0	13,042
Jul	2,292	9,180	1,427	0	12,899
Aug	2,292	9,180	1,427	0	12,899
Sep	2,292	9,180	2,682	0	14,154
Oct	2,292	9,180	18,187	0	29,659
Nov	2,292	9,180	34,941	2,825	49,238
Dec	2,292	9,180	41,748	4,707	57,927
					Yearly total: 411,711 kr

Figure 3.11: Detailed budget of a normal year cost prognosis as a table

3.4 Customer performance

The chosen key figure for evaluating customer performance, ΔT, is plotted for all LEAB's customer groups in figure 3.12. Figure 3.13 shows a graph of the histogram and PDF for all customers collectively.

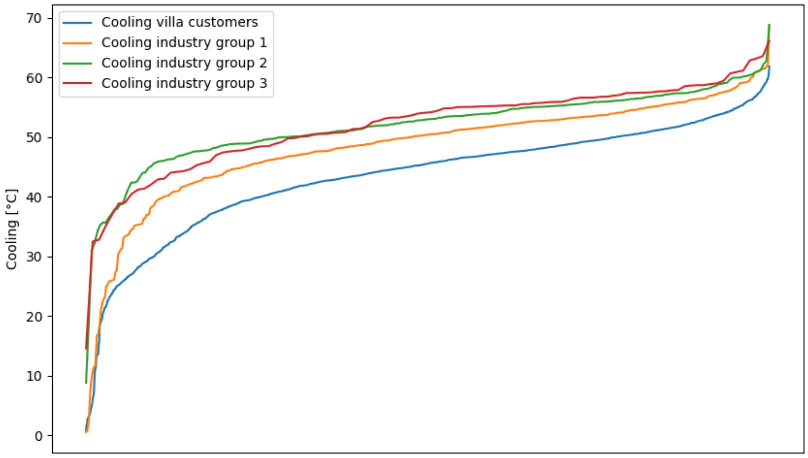


Figure 3.12: Estimated ΔT for the different customer segments

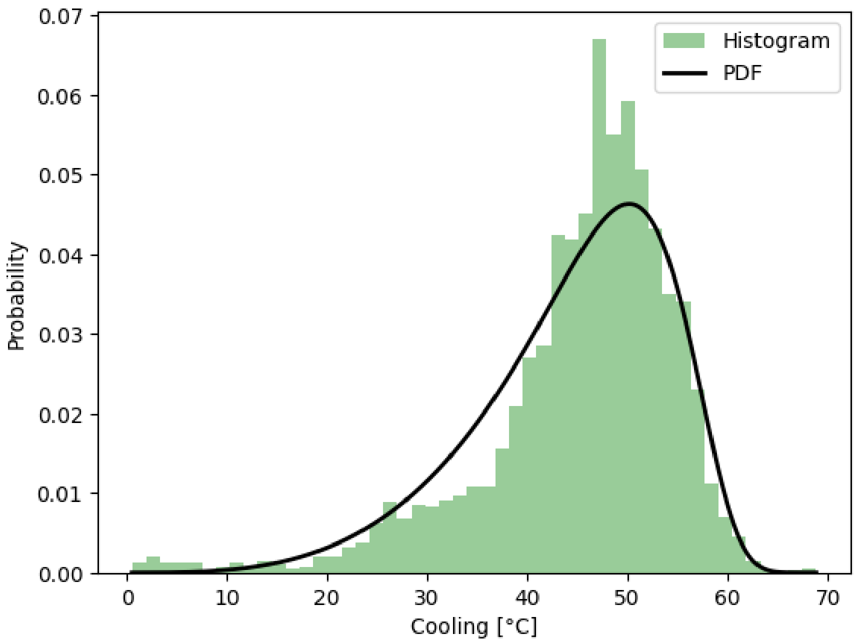
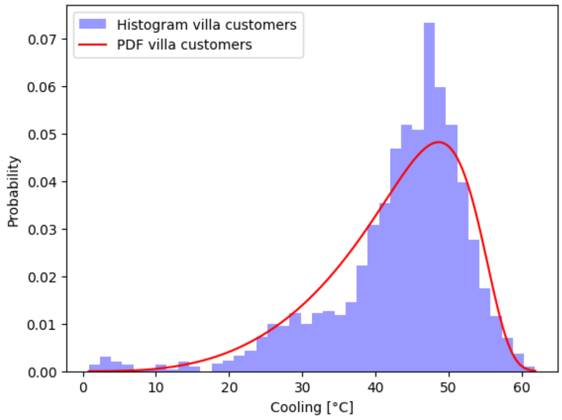


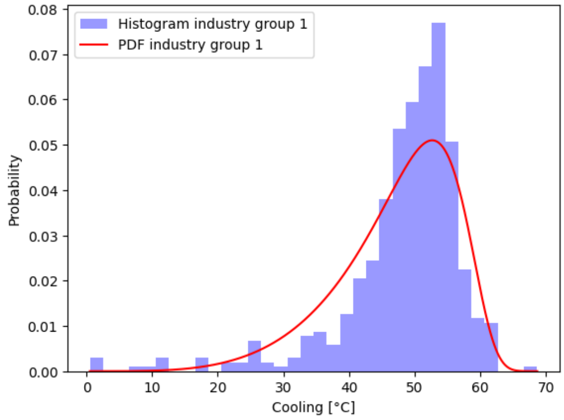
Figure 3.13: Histogram and probability density function for the whole customer base

Histogram and distribution of the different customer segments

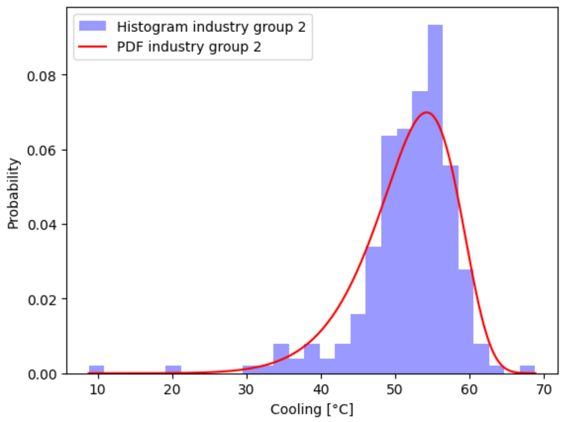
When creating the individual histograms and PDF's from the data in figure 3.12 the sub figures in figure 3.14 were the result. Table 3.4 gives the individual shape and scale parameters for the different PDF's. Industry customer group 4 had too few customers assigned for any meaningful observations to be made and was therefore excluded from the thesis.



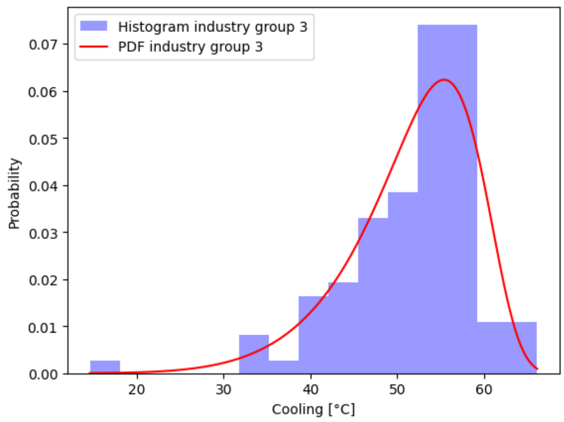
(a) Histogram and probability density function for the villa customers



(b) Histogram and probability density function for industry customer group 1



(c) Histogram and probability density function for industry customer group 2



(d) Histogram and probability density function for industry customer group 3

Figure 3.14: Histogram and distribution for the different customer segments

3.4.1 Comparison of the different probability density functions

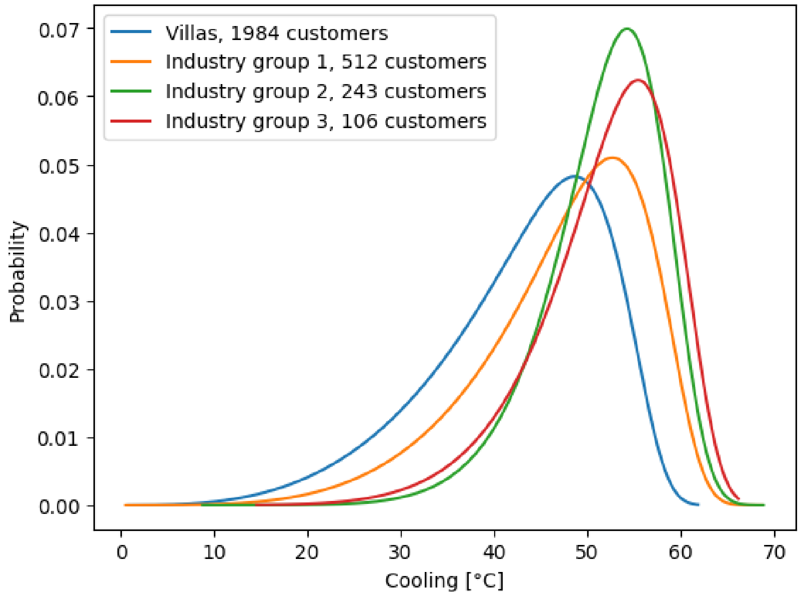


Figure 3.15: A comparison of all the probability density functions

Table 3.4: Weibull parameters for different customer groups

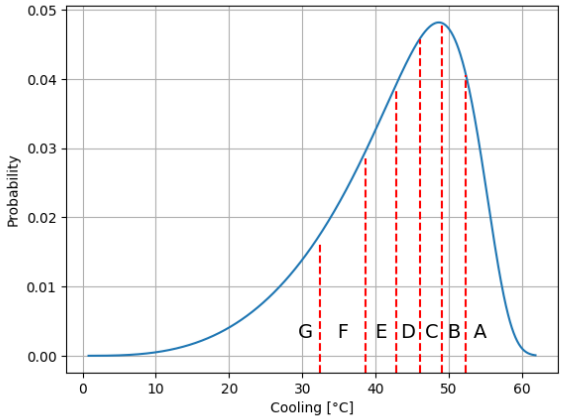
Customer Group	Shape	Scale
All customers	0.3	0
Villa customers	0.3	0
Industry group 1	0.3	0
Industry group 2	0.7	0
Industry group 3	0.5	0

Table 3.5: Quantiles for different customer groups [°C]

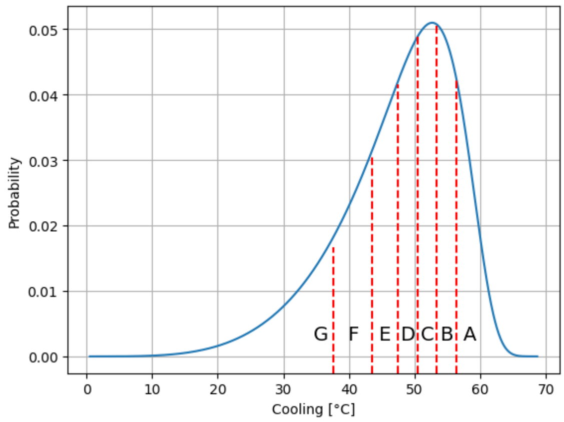
Customer Group	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**
All customers	34.2	40.4	44.5	48.0	51.1	54.4
Villa customers	32.5	38.6	42.8	46.1	49.1	52.2
Industry group 1	37.7	43.5	47.4	50.5	53.3	56.3
Industry group 2	45.0	48.8	51.4	53.5	55.6	57.9
Industry group 3	44.0	48.6	51.6	54.1	56.4	59.0

3.4.2 Labeling within the customer groups

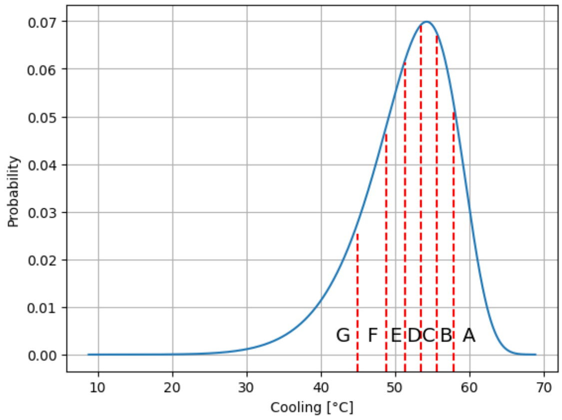
The different thresholds for the labeling intervals withing each customer group is shown in figure 3.16. Figure 3.16a is for villa customers, 3.16b for industry customer group 1, 3.16c for industry customer group 2 and finally 3.16d for industry customer group 3. The different quantiles are further given in table 3.5.



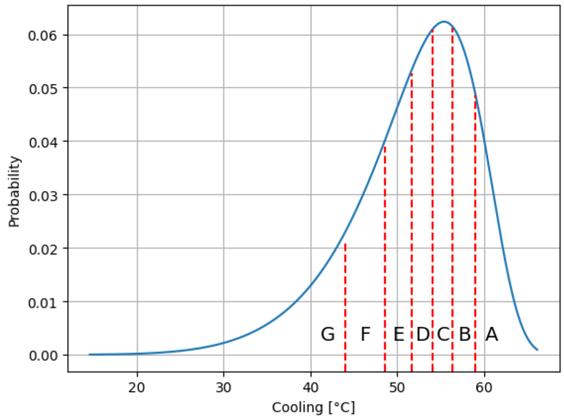
(a) Quantiles for villa customers



(b) Quantiles for industry customer group 1



(c) Quantiles for industry customer group 2

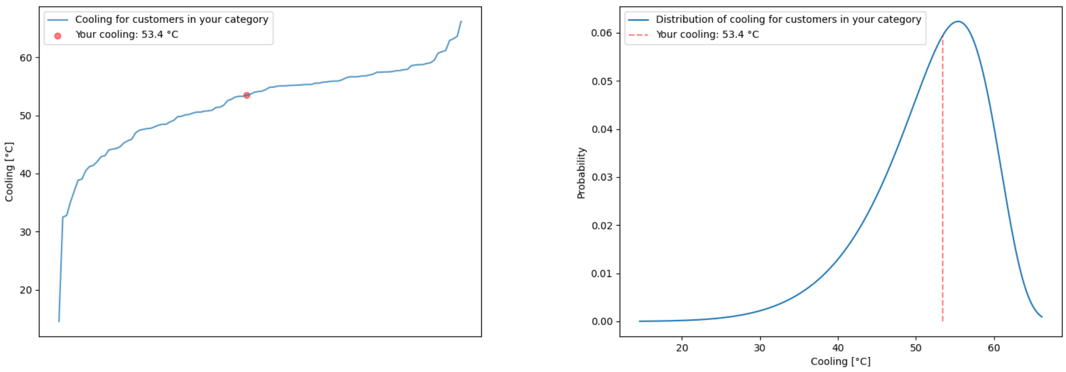


(d) Quantiles for industry customer group 3

Figure 3.16: Histogram and distribution for the different customer segments with quantiles illustrated

3.4.3 Communicating customer performance

Figure 3.17a and figure 3.17b illustrates a customers cooling either on a sorted set of the cooling data or in a Weibull distribution respectively.



(a) Illustrating a customers cooling on the sorted set of their customer group

(b) Illustrating a customers cooling on a Weibull distribution of their customer group

Figure 3.17: Two alternative ways of displaying cooling for a particular customer

3.4.4 Energy performance sticker

Figure 3.18 is an example of how a customer can view their energy performance in relation to other customers of similar size. In this case, the customer had received a D label, which is illustrated by the extra thick border around that bar.

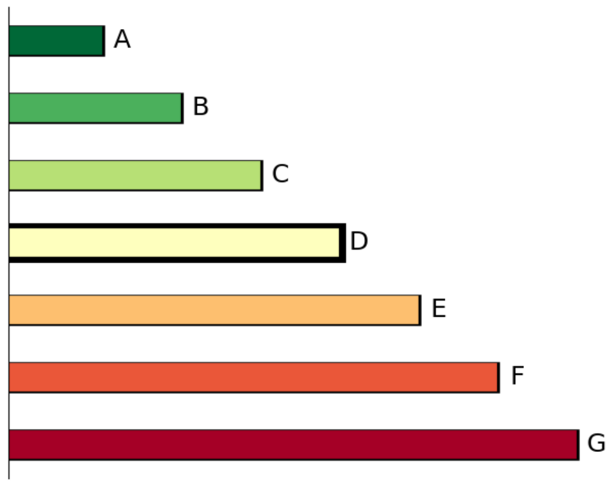


Figure 3.18: An energy performance sticker

3.5 Correlation analysis

Meta data from section 1.5.15 was appended to the heating season data set. An analysis was performed on the industry customers as one set and the villa customers as one set. Figure D.1 and D.2 in Appendix D are the matrices for industry and villa customers respectively. Table D.1 translates the different columns in the matrices.

3.6 The final customer budget

The following pages show the renewed and improved customer budget.

3.7 Easy implementation

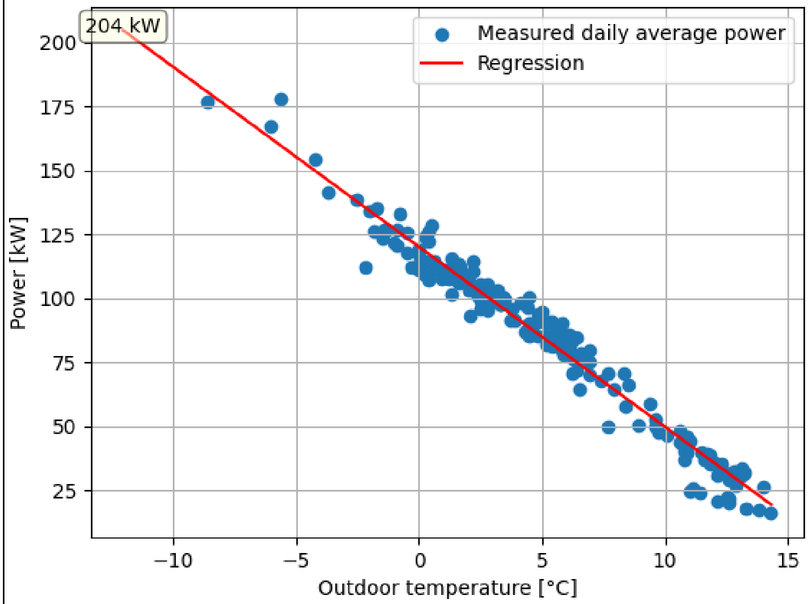
To produce the results listed in this chapter, the author estimates that around 200 hours of programming in python was needed, at an amateur level. The scripts are estimated take something in the magnitude of a few hours to execute with the current, non optimized code.

Summary of your key figures

We have estimated your yearly consumption during a normal year using machine learning to establish the pattern from your consumption during 2023. Prices are for the year of 2023, the power rate and the fixed rate are based on your heat load signature which is calculated from your consumption during the heating season 2022-2023.

These are calculated key figures and estimations for your consumption and costs during a normal year:

Estimated consumption, normal year: 467 MWh
Estimated costs, normal year: 411,711 SEK
Estimated consumption, hot year: 400 MWh
Estimated costs, hot year: 375,430 SEK
Estimated consumption, cold year: 543 MWh
Estimated costs, cold year: 452,930 SEK
Estimated annual hot tap water usage: 32 MWh
Estimated yearly hot water consumption cost: 11,039 SEK
Estimated balance temperature: 14.7°C
Heat load signature: 204 kW
Signature set by: Linear regression
Coefficient of determinartion for regression: 0.96
Your comparison price is: 880 kr/MWh

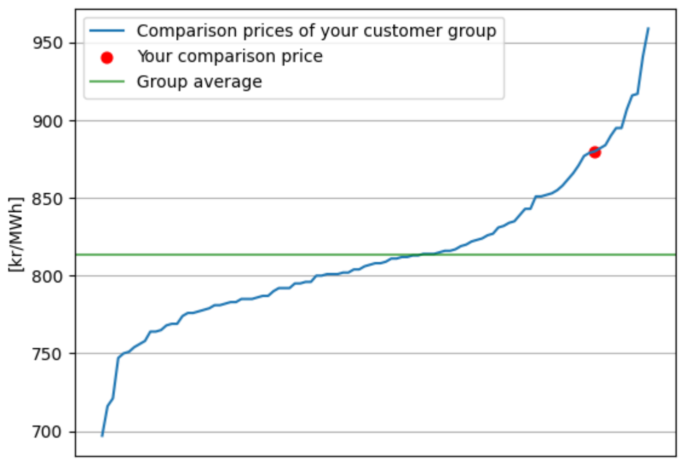


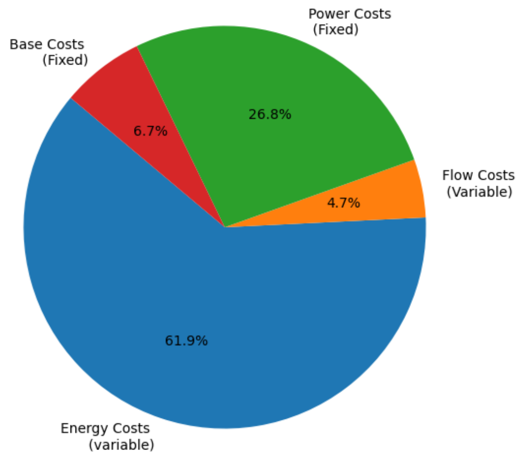
Your heat load signature is primarily set by estimating your power at a Dimensioning Winter Outdoor Temperature (DWOT) of -12°C. This estimation is performed by training an algorithm to estimate your power usage as a function of outdoor temperature. If the model is descriptive enough, it is used to establish your heat load signature, otherwise, the heat load signature is set by using your highest measured daily average power. The heat load signature sets the fixed rates for you price model, as our investment increases with the size of your substation.

Predicted costs for a normal year, by month

Your costs for a normal year has been estimated by analyzing your outdoor temperature dependency from the previous year. The same usage pattern has then been applied to a normal year in order to provide a comparable cost. To complement this, we have included scenarios for colder and hotter years as well, they are located at the end of the document. In the figures below you can see your normal year costs displayed per price model item and per month. You can also see the distribution between fixed and variable costs in the circle diagram. Your comparison price has been calculated as your annual cost divided by your annual heat consumption, it is displayed in relation to the comparison prices of customers in your customer group. We explain what your customer group is on the following page.

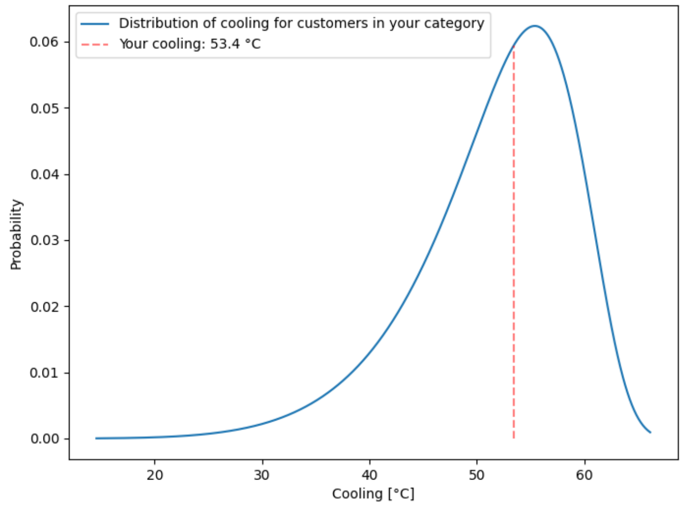
	Fixed fee [kr]	Power cost [kr]	Energy cost [kr]	Flow cost [kr]	Total [kr]
Jan	2,292	9,180	40,484	4,055	56,011
Feb	2,292	9,180	36,029	3,653	51,154
Mar	2,292	9,180	42,216	4,004	57,692
Apr	2,292	9,180	25,997	0	37,469
May	2,292	9,180	8,093	0	19,565
Jun	2,292	9,180	1,570	0	13,042
Jul	2,292	9,180	1,427	0	12,899
Aug	2,292	9,180	1,427	0	12,899
Sep	2,292	9,180	2,682	0	14,154
Oct	2,292	9,180	18,187	0	29,659
Nov	2,292	9,180	34,941	2,825	49,238
Dec	2,292	9,180	41,748	4,707	57,927
					Yearly total: 411,711 kr

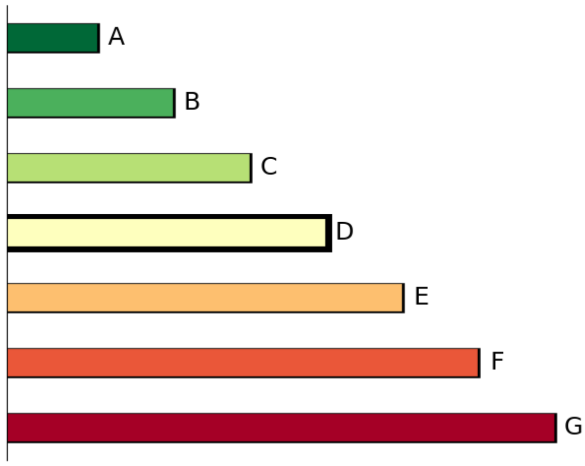




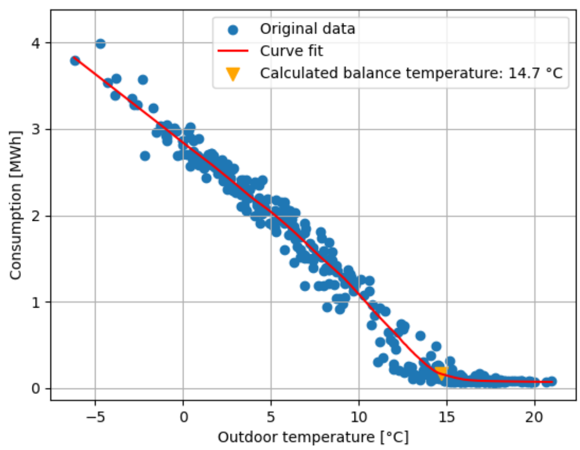
Your key figures for comparison

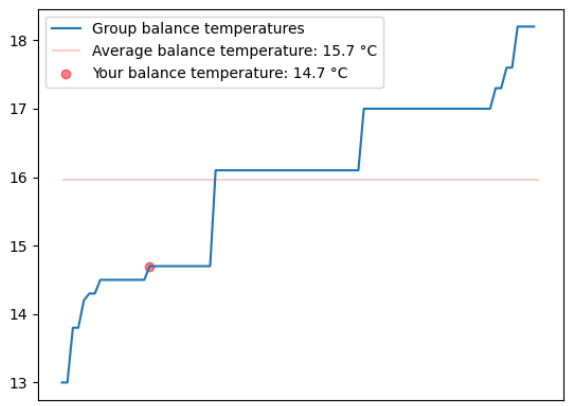
An important part of a District Heating grid is to have a low return temperature, or high cooling in customer substation. This is important since lower return temperatures help extract more energy from the fuel used at the production site and combats wasting resources. In accordance with the Swedish District Heating law, you have been grouped with similar customers on our District Heating grid to provide a performance comparison. Comparison within the group is done by evaluating the efficiency of your substation in relation to the rest of the group. This efficiency is measured by how well it extracts heat from the grid, the higher cooling it has, the better the performance. Your customer group is set by your heat load signature and the comparison is provided throug the two figures below. You can see how well your substation cools incoming District Heating compared to the group. Your energy performance certificate to the right compares you to your customer group.



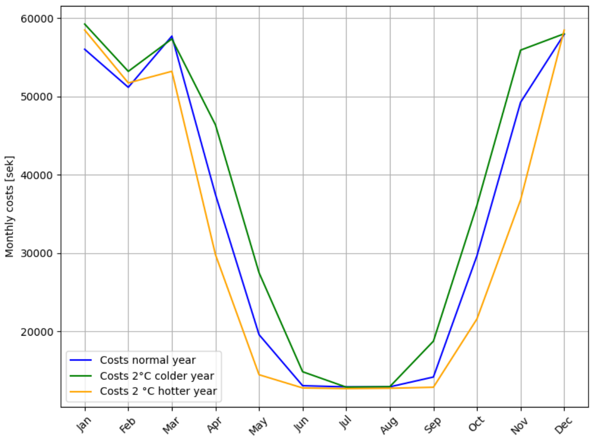


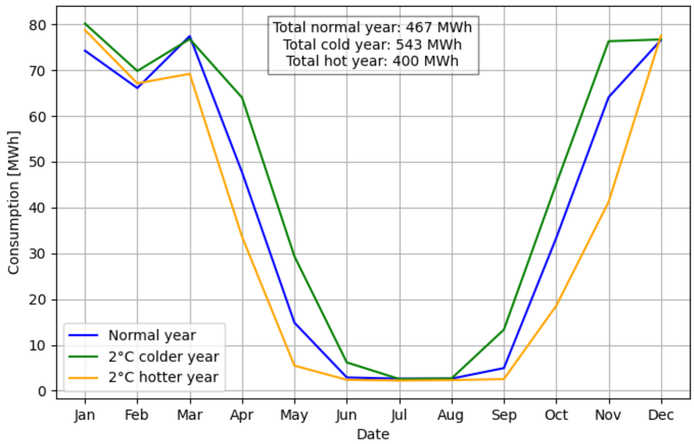
Your balance temperature, meaning the temperature where you do not need additional heating, has been individually set, a low balance temperature indicates a good climate shell and low heating need. This has been done by analyzing your heating consumption and evaluating the relation between heat consumption and the outdoor temperature. This is illustrated by the figure to the left below. To the right is a comparison to the rest of the customers in your group.





Estimated consumption and costs for normal year with scenarios





Both figures above are values given as monthly sums. We have used machine learning to establish a pattern for your heating consumption. This pattern is then used on temperature data for a normal year as well as for scenarios of hotter and colder years, which resulted in the consumption profiles seen to the right above. These consumption predictions are then used with your assigned price models to estimate costs for the coming year, which are shown in the left figure. The table on the first page is a higher resolution of the normal year cost prediction

Environmental data and your CO2 emissions

The following is environmental data for our DH production for the year of 2022. Our grid has a primary energy factor of 0.05, fossil fuels have a primary energy factor greater than one, recycled heat has a value of 0. Waste and residual products have a value smaller than 1, a small primary energy factor is therefore good. Our grid uses 0.2% fossil fuels in the energy mix and every kWh has an average emission of 59.8 g CO2/kWh. A normal year consumption for your facility therefore emits 27 kg of CO2.

The basis for your pricing is provided below

Your energy rates for winter, autumn/spring and summer respectively: 545, 285, 100 kr/MWh. Flow tax: 3.5 kr/m3. Based on your category 176-1400 kW, you are charged an annual fixed fee of 27506 kr and an additional power fee of 540 kr/kWh based on your heat load signature.

Communication is the future

If you have any questions about the information provided in this document, please reach out to discuss them with us. Based on the new analyses we can perform, these measures give us new insight in to what could be optimized for a better performance of your substation which in turn would lead to lower costs and a better user experience for you. There is additional information we could provide you with if you would consider sharing data like heated area for example. Additionally, all metrics provided in this document can be provided as a comparison to your customer group, contact us if you wish for additional analyses. Our contact information is available at the bottom of each page, take care.

4 Discussion

This chapter aims to view the methodology of the thesis through a criticizing lens and to provide insight in to the findings of the thesis.

4.1 An industry opportunity

In Appendix A, I have included the most relevant code for how I conducted this thesis work. As I mention in section 3.7, the procedure is not that resource demanding and I believe that the results can be worth wile. I leave it here hoping that it might inspire the DH industry to implement data processing like this on their own. Hopefully, this thesis shows that it is not all that resource intensive to generate additional customer value using already available data. My hope is for the industry to strive further in their efforts for transparency and the empowerment of their customers.

4.2 Comments on methodology choices

There were many choices to be made when developing the methodology for this thesis, some more straight forward than others. This section will give the reader an insight in to how I navigated these choices.

4.2.1 Data processing

The z-score was chosen for outlier removal as it utilizes a linear regression for residual calculations. Since the data at hand has a great temperature dependence, it seemed like a good choice to clean data using a method which takes advantage of linear dependency.

4.2.2 Choice of ML algorithm

After reading the conclusions from Zhang et al. and Kadiyala and Kumar, it can be concluded that careful tuning of the ML algorithm would benefit the results. The usage of the generic XGBoost model did provide useful results either way in the context of this thesis. It should also be noted that a major downside with XGBoost which i didn't learn until late in to the thesis work is it's issues with extrapolation. Perhaps it is possible, but I didn't manage to figure out how to use XGBoost to extrapolate outside of the training feature range. In other words, the model flat lines when tasked to predict outside the range of the feature variable it was trained on. This became evident when I attempted to use a XGBoost Regressor to perform the heat load signature calculations and the extrapolated heat power signature went flat after the lowest recorded temperature. With this said, if you study figure 3.8 and 3.9 you can see that during January and February, the temperature four the cold year is lower than the lowest temperature in the normal year series. But still, the predicted consumption is greater for the cold year than for the normal year.

In this thesis, the XGBoost model was used with preset parameter settings. Despite this, the model still performed adequately and was effective for demonstration purposes within the larger framework of the proposed service methodology. It did however behave unexpectedly when predicting consumption of the hot year scenario for example. If figure 3.9 is studied, you can see that the model, predicts higher consumption for the hot year compared to the normal year for some months. Why this is, I do not know.

4.2.3 An engineering approach to estimating balance temperature and base load

When developing the methodology for the balance temperature estimation, an idea of evaluating the slope of the heat power signature was the seed. The method of fitting a curve and evaluating the slope went almost as expected, there were however more NaN values in the derivative array than anticipated. The choice of evaluating the greatest slope and using as a benchmark seems appropriate, but setting the threshold for what percentage of the benchmark slope should define the balance temperature was purely experimental.

Several different percentages were tried and the distribution of estimated balance temperatures was visually investigated. The goal was to have as few estimations as possible in the two extreme "tails" which can be seen in figure E.3. Investigating the figures in Appendix E, figures E.1 and E.2 show quite low estimations when using a threshold of 20 % and 25 % respectively. On the other hand, figures E.4 and E.5 put quite few customers at below 12 °C in balance temperature. Finally, figure E.6 with a threshold at 50 % seems to be too high as the villa customers are estimated to have a balance temperature of 14.2 °C.

Looking at the page of the energy index guide from SMHI, there is an accessible PDF named *Teknisk Typhusguide Energi-Index.pdf (338 kB, pdf)* where you can find that they list different balance temperatures for different buildings. Office buildings are recommended to have balance temperatures set to 7 °C - 11 °C. Multi family buildings are listed to range from 13 °C - 17 °C depending on standard.

With these things in mind, any of the choices of 30 % - 40 % for the threshold could be argued for. In reality, a methodology for this sort of evaluation should be more thought through if it were to be applied in a more meaningful way. As the greatest implication of this estimation is for display to a curious customer, it could still serve a purpose as it's main goal is to put the customer in relation to other similar customers.

Following the balance temperature, the base heat load is directly defined in this thesis as the average of all consumption at outdoor temperatures above the balance temperature. Disregarding how the balance temperature was set, there should probably be a similar outlier removal of the base load measurements before taking the average. Figure 3.6b is an example of a customer with many observations of negative consumption after the subtraction of the base heat load. This could be because of the lack of cleaning, but as the estimated balance temperature is listed at 12.7 °C, the script could also have underestimated the balance temperature. This would in turn lead to an overestimated base heat load.

For certain customers, the dependency on the outdoor temperature was constant. The regression line was a good fit even up to the highest recorded temperature. This made it difficult for the method of evaluating the derivative for determining balance temperature as there never was any clear break in the slope. An example is given in figure 3.5 where the balance temperature seems rather ambiguous.

As a final remark on the base heat loads, the sub figures in figure 3.7 show the relationship between the the heat load signature and the estimated base heat load. Or in other words, the size of the facility versus the amount of hot tap water used. For industry customer group 1-3, there is a slight trend of increasing base load with increasing size, which would be expected. The villa customers however seem harder to analyze in this matter, partly because the range in heat load signature is quite small.

4.2.4 The hot and the cold year scenarios

The basis for the scenarios which can be seen in figure 1.2 is in fact +1.5 °C and - 2.0 °C as maximal deviations in the yearly average from the rolling average. The reason for the ± 2 °C was the simplicity of a whole number being used symmetrically. The resolution could have been swapped from yearly to seasonal and a more careful analysis of the variance could be performed, but you have to draw the line somewhere.

4.2.5 Choice of distribution model

Even though the scale parameters for all Weibull distributions were estimated to be zero, as shown in table 3.4, figure 3.14 demonstrates a reasonably good fit. The primary focus of this thesis is not on the intricacies of statistical mathematics but rather on developing a suggested methodology for customer segmentation.

4.2.6 Additional data for consumption feature extraction

Wang et al. mention the usage of weather data such as wind speed, relative humidity, and solar radiation [20]. It would be interesting to expand the dimensions of features for and perform a proper feature extraction, which would most likely lead to a more accurate model. If feature extraction was relevant, XGBoost would be able to perform one, quite well according to Zhang et al. [22]. Wang, Hong and Piette added noise to data for real world accuracy, which could have been an interesting inclusion but it wouldn't have changed the results of the thesis [21].

4.2.7 Poor heat load signature calculations

A few of the heat load signatures were off by a lot, as seen in figures C.1 and C.2. This could be due to recorded consumption data being for the building stage of a customer location and the only heat used is for construction heat like curing concrete. If so, LEAB might have already set a heat load signature for their process which could be significantly higher than what I calculated.

4.3 Suggestions for future research

Throughout the exploration of this thesis, as is often the case, other interesting opportunities have been unraveled where further research would be of interest. The following is a list of those worthy to mention.

4.3.1 Additional data for more complex analyzes

Knowing heated area could further help categorizing customers either by W/m^2 or perhaps just by the size of the area. Either way, for villa customers and for multi dwelling buildings, it would be a relevant key figure as there are SMHI guidelines for heating power W/m^2 for different types of buildings.

With for example access to power capacity and sizes of heat exchangers in a customer substation it might be possible to predict hot tap water load more accurately. If the outdoor temperature was well below the DWOT you could assume the heat exchanger is working at maximum capacity and estimate the tap water load with the current consumption minus the rating for the heat exchanger.

Over all, having more complete records of meta data would open many doors for interesting analyzes, studying the correlation matrices in figure D.1 and D.2 you can see that there are correlating features within the existing incomplete meta data set.

4.3.2 Deeper analyses

Internal investigation of aggregated system performance could be done by DH companies. Evaluation of return on investment for different yearly scenarios could be performed by alteration of the temperature data used to predict consumption. It would be interesting to investigate further needs of the DH industry and to dig deeper into the possibilities of further digitalisation.

4.4 Recommendations

By writing this thesis, I have identified a few things that I'd like to address. Firstly, comparing the different flow rates in table 1.1, it seems like LEAB has quite some room to increase their flow rates. Kraftringen and E.on even charge for flow the whole year and have about twice the flow rate of LEAB, which could make an actual difference in customer incentives.

5 Conclusions

In this thesis, I have exemplified how to illustrate a customer and its performance and how to display their environmental strain through their DH usage. All using simple means available to any and all who wish to make the same endeavour.

The methodology of grouping and labeling customers developed in this thesis does comply with the new DH law in grouping and labeling customers. Firstly grouping customers by size, i.e. heat load signature, is relevant as it is an already used metric. For the more detailed division when labeling, the cooling was chosen as it is an important metric for the DH company and a greater illustration of a customers performance in this metric aims to underline this importance. Beyond this comparison profile, the new customer budget also provides information about the DH productions primary energy factor and its corresponding CO_2 emissions. Once the new DH law eventually defines how to label and compare customers, LEAB will have a head start and can set a good example for the rest of the industry.

As for the possibility of heat load normalization using machine learning, the results were interesting. It can be seen in figure 3.3 that the XGBoost algorithm was able to produce similar results to the energy index. But also that further model development would be preferred as the model behaves unexpectedly at times. Instead of using the previous years consumption pattern for calculating a customer cost proposition for the coming year as LEAB did previously, the new format of a normal hot and cold year communicates the strong weather dependency of their DH costs.

By simply comparing the old customer budget in the introduction to the new one presented in the results, I believe it is quite evident that customer value has been created. As Sernhed, Gåverud and Sandgren had found in their study regarding customer perspectives, there existed a customer need to be able to create better budgets for DH. The deeper break down of the pricing calculations should help them gain a greater insight in to how they are billed too. LEAB gains value from this too, greater customer satisfaction is always to strive for as a business. Beyond that, they have gained tools to look deeper in to the characteristics of their own DH grid too. I also believe that exposing their customers to more information will generate larger volumes of questions from their customers, allowing LEAB to start communication and further educate their customers.

The new methodologies developed for balance temperature estimation turned out to yield rather interesting results too. It would be fun to use the method on a data set with a known truth, where actual hot tap water load is measured. This is, according to me, another example of interesting analyzes available to an energy company which does not require large investments or subscription fees to extract. These methodologies could all benefit from a wider inclusion of meta data and higher resolution data, for example hourly averages instead of daily.

As a final remark, many of the methodologies developed in this thesis is already available for energy providers on the market today. But what is interesting is what could be achieved during just one masters thesis with the help of open source programming. I believe that this thesis should empower energy producers to start incorporating more of these methodologies on their own since they already are in possession of their own data and a simple laptop could open many doors for making interesting analyzes. What errors and faults in the grid could be unveiled?

Bibliography

[1] Hervé Abdi. "Z-scores". In: *Encyclopedia of measurement and statistics* 3 (2007), pp. 1055–1058.

[2] Boverket. *Öppna data - Dimensionerande vinterutetemperatur (DVUT 1981-2010) för 310 orter i Sverige*. 2017. URL: https://www.boverket.se/sv/om-boverket/publicerat-av-boverket/oppna-data/dimensionerande-vinterutetemperatur-dvut-1981-2010/ (visited on 21/05/2024).

[3] D Chicco, MJ Warrens and G Jurman. "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation". In: *PeerJ* (2021). DOI: https://doi.org/10.7717/peerj-cs.623.

[4] E.ON AB. *Fjärrvärmekostnadens delar*. 2023. URL: https://www.eon.se/foeretag/vaerme-och-kyla/for-foretag-som-har-fjarrvarme/fjarrvarmepriser#fjarrvarmekostnad (visited on 24/06/2024).

[5] J. Ferré. "3.02 - Regression Diagnostics". In: *Comprehensive Chemometrics*. Ed. by Steven D. Brown, Romá Tauler and Beata Walczak. Oxford: Elsevier, 2009, pp. 33–89. ISBN: 978-0-444-52701-1. DOI: https://doi.org/10.1016/B978-044452701-1.00076-4. URL: https://www.sciencedirect.com/science/article/pii/B9780444527011000764.

[6] Svend Fredriksen and Sven Werner. *District Heating and Cooling*. Åkergränden 1, 226 60 Lund: Studentlitteratur, 2013.

[7] Leire Gorroño-Albizu and Jaqueline de Godoy. "Getting fair institutional conditions for district heating consumers: Insights from Denmark and Sweden". In: *Energy* 237 (2021), p. 121615. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2021.121615. URL: https://www.sciencedirect.com/science/article/pii/S0360544221018636.

[8] Ulrika Hesslow and Maria Werleskog. *Föreskrift EIFS 2022:3*. 2022. URL: https://ei.se/om-oss/publikationer/publikationer/foreskrifter-fjarrvarme-och-fjarrkyla/2022/foreskrift-eifs-20223 (visited on 15/03/2024).

[9] Akhil Kadiyala and Ashok Kumar. "Applications of python to evaluate the performance of decision tree-based boosting algorithms". In: *Environmental Progress & Sustainable Energy* 37.2 (2018), pp. 618–623. DOI: https://doi.org/10.1002/ep.12888. eprint: https://aiche.onlinelibrary.wiley.com/doi/pdf/10.1002/ep.12888. URL: https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/ep.12888.

[10] Raziyeh Khodayari. *Miljövärdering av fjärrvärme*. 2023. URL: https://www.energiforetagen.se/statistik/fjarrvarmestatistik/miljovardering-av-fjarrvarme/ (visited on 07/06/2024).

[11] Kraftringen AB. *Priser: Fjärrvärme.* 2023. URL: https://www.kraftringen.se/foretag/varme-kyla/fjarrvarme-kraftringen/fjarrvarmepris/ (visited on 24/06/2024).

[12] Landskrona Energi AB. *Fjärrvärmepriser.* 2024. URL: https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/ep.12888.

[13] Kristina Lygnerud et al. "District heating in the future - thoughts on the business model". English. In: *Energy* 278 (2023). ISSN: 0360-5442. DOI: 10.1016/j.energy.2023.127714.

[14] Prisdialogen. *Varför Prisdialogen?* (n.d.) URL: https://www.prisdialogen.se/om-prisdialogen/varfor-prisdialogen/ (visited on 05/06/2024).

[15] Kerstin Sernhed, Henrik Gåverud and Annamaria Sandgren. "Customer perspectives on district heating price models". In: *International Journal of Sustainable Energy Planning and Management* 13 (2017), 47–60. DOI: 10.5278/ijsepm.2017.13.4. URL: https://journals.aau.dk/index.php/sepm/article/view/1637.

[16] SMHI. *Climate indicator - Temperature.* 2015. URL: https://www.smhi.se/en/climate/climate-indicators/climate-indicators-temperature-1.91472 (visited on 21/04/2024).

[17] SMHI. *SMHI Energi-Index och Graddagar – normalårskorrigering värme.* 2022. URL: https://www.smhi.se/professionella-tjanster/hallbara-stader/smhi-energiindex-och-graddagar-normalkorrigering-varme-1.185236 (visited on 11/04/2024).

[18] Svensk FJärrvärme. *REKO FJÄRRVÄRME.* https://www.energiforetagen.se/globalassets/energiforetagen/energifakta/fjarrvarme/reko-fjarrvarme-riktlinjer-kundrelationer.pdf. Accessed: 04-02-2024. 2008/2015.

[19] Sveriges riksdag. *Fjärrvärmelag (2008:263).* https://www.riksdagen.se/sv/dokument-och-lagar/dokument/svensk-forfattningssamling/fjarrvarmelag-2008263_sfs-2008-263/. Accessed: 30-01-2024. 2008/2015.

[20] Chao Wang et al. "New methods for clustering district heating users based on consumption patterns". In: *Applied Energy* 251 (2019), p. 113373. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2019.113373. URL: https://www.sciencedirect.com/science/article/pii/S0306261919310475.

[21] Zhe Wang, Tianzhen Hong and Mary Ann Piette. "Building thermal load prediction through shallow machine learning and deep learning". In: *Applied Energy* 263 (2020), p. 114683. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2020.114683. URL: https://www.sciencedirect.com/science/article/pii/S0306261920301951.

[22] Tingze Zhang et al. "Long-Term Energy and Peak Power Demand Forecasting Based on Sequential-XGBoost". In: *IEEE Transactions on Power Systems* 39.2 (2024), pp. 3088–3104. DOI: 10.1109/TPWRS.2023.3289400. URL: https://ieeexplore.ieee.org/abstract/document/10163761.

[23] Öresundskraft AB. *Fjärrvärmepriser i Helsingborg och Ängelholm.* 2023. URL: https://www.oresundskraft.se/foretag/fjarrvarme/priser-fjarrvarme/ (visited on 24/06/2024).

Appendix A

Python code

A.1 Heat load signature calculations

```
# Defining a function which calculates the head load at DUT, generates the power signature and returns the R squared
# Data needs to be named exactly "Verklig temperatur" and "Ber knad dygnsmedeleffekt" for this to work.

def heat_load_signature(customer_dfs, output_path):
    """
    Generate a DataFrame containing the subscribed power for each customer.

    Args:
        customer_dfs (list): List of DataFrames, where each DataFrame represents the data for a specific customer.
        output_path (str): String containing the assigned folder for heat load signature plots.

    Returns:
        subscribed_power (list): List containing the ID, the subscribed power, calculation method and r squared for each customer.
    """

    # Importing packages
    import numpy as np
    from sklearn.linear_model import LinearRegression
    import matplotlib.pyplot as plt
    import os
    import pickle
    from sklearn.metrics import mean_squared_error, r2_score
    from sklearn.model_selection import train_test_split

    subscribed_power = [] # 'customer_ID', 'subscribed_power', 'correlation_method', 'r_squared'

    print("Calculating subscribed power")

    # Iterate over each customer DataFrame
    for ID, df in customer_dfs:
        # Extracting data for the current customer
        df = df[df['Ber knad dygnsmedeleffekt'] != 0]

        if not df.empty:
            X = df['Verklig temperatur'] # Temperature data
            y = df['Ber knad dygnsmedeleffekt']  # Average daily
```

```
power data
        DUT = -12  # The dimensioning winter outdoor temperature
        X_DUT = np.linspace(DUT, np.max(X), num=len(y))  # New temperature array going from DUT to max temperature reading

        # Reshaping for training, needed to use the current libraries
        X = X.values.reshape(-1, 1)
        y = y.values.reshape(-1, 1)
        X_DUT = X_DUT.reshape(-1, 1)

        seed = 45
        test_size = 0.2
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=seed)

        # Create and fit the linear regression model
        model = LinearRegression()
        model.fit(X_train, y_train)

        # Calculating and evaluating the r squared
        predictions = model.predict(X_test)
        r_squared = np.round(r2_score(y_test, predictions), 2)
        threshold = 0.75 # Setting threshold for acceptable r squared for correlation

        # Extrapolating the heat load pattern to DUT
        X_DUT_predictions = model.predict(X_DUT)

        # Determining if the r squared is high enough to use extrapolation for subscribed power.
        # Otherwise, the highest recorded power is used
        if r_squared >= threshold:
            # Extracting the subscribed power from this extrapolation
            sub_power = X_DUT_predictions[0][0].astype(int)
            corr_method = 'Linj r korrelation'
        else:
            sub_power = np.max(y).astype(int)
            corr_method = 'H gsta avl sta dygnsmedeleffekt'
    else:
        sub_power = 0
        corr_method = 'Ingen dygnsmedeleffekt avl st'
        r_squared = 0

    # Append the subscribed power value and the customer ID to the DataFrame
    subscribed_power.append([ID, sub_power, corr_method, r_squared])

    # Creating plot of heat load signature
    plt.figure()
    plt.scatter(X, y, label='Uppm tt dygnsmedeleffekt')
    plt.plot(X_DUT, X_DUT_predictions, color='red', label='Korrelation')

    # Annotate the plot with the value of the subscribed power at DUT
```

```
        plt.annotate(f'{sub_power:.2f} kW', (X_DUT[0], sub_power),
                    xytext=(0, 0), textcoords='offset points',
                    ha='center',
                    bbox=dict(boxstyle='round,pad=0.3', fc='lightyellow', alpha=0.5))

        # Setting axis and labels for the plot
        plt.xlabel('Utomhustemperatur [ C ]')
        plt.ylabel('Effekt [kW]')
        plt.title(f'Ber kad dygnsmedeleffekt f r kund-ID: {ID}, R\u00B2 = {r_squared}')
        plt.legend()
        plt.grid(True)

        # Creating a unique path for the output
        output = os.path.join(output_path, f'{ID}.png')

        plt.savefig(output)
        plt.close()

    # Save variables to a file

    with open("//Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/Subscribed_power.pkl", "wb") as f:
        pickle.dump((subscribed_power), f)

    return subscribed_power
```

Listing A.1: Heat load signature calculations

A.2 Customer budget generation

```
def generate_my_budget(subscribed_power, norm_consumption, base_load_costs, yearly_costs, balance_temps):
    """
    Generate customer budgets based on price model, forecasted consumption and the subscribed power.

    Args:
        subscribed_power (list): List of lists, where each list represents the data for a specific customer. Also includes ID.
        norm_consumption (list): List of lists, each row contains consumption patterns for normal year, hot and cold year.
        base_load_costs (list): List with calculated base load costs for each customer where the method was applicable.
        yearly_costs (list): List with costs calculated for an entire year, different scenarios.
        balance_temps (list): List of calculated balance temperatures for each customer where the mathod was applicable.

    Returns:
        Nothing (Null). Saves a PDF at the target location for each customer input
    """

    from fpdf import FPDF
```

```python
    from PIL import Image
    import numpy as np
    import pandas as pd

    for row in subscribed_power:
        try:
            ID = row[0]
            sub_power = row[1]
            corr_method = 'Linear regression'  ## MAKE THIS RIGHT IF ACTUAL USE OF THE SCRIPT IN THE FUTURE!
            r_squared = np.round(row[3], 2)

            price_model = [row for row in price_models if row[2] == ID]
            price_df = price_model[0][0]

            comparison_price = [row[1] for row in comparison_prices if row[0] == ID]
            comparison_price = comparison_price[0]

            pd.to_datetime(price_df['Date'])
            price_df.set_index('Date')

            summer_energy = int(price_df[price_df['Date'] == '2023-06-01']['Energy price'])
            autumn_spring_energy = int(price_df[price_df['Date'] == '2023-04-01']['Energy price'])
            winter_energy = int(price_df[price_df['Date'] == '2023-01-01']['Energy price'])
            flow_tax = int(price_df[price_df['Date'] == '2023-01-01']['Flow tax'])
            power_tax = int(price_df[price_df['Date'] == '2023-01-01']['Power tax'])
            yearly_fixed = int(price_model[0][1])
            customer_category = price_model[0][3]

            yearly_costs_row = [item for item in yearly_costs if item[0] == ID]
            yearly_costs_row = yearly_costs_row[0]
            ID, norm_sum, cold_sum, hot_sum = yearly_costs_row

            base_load = [thingy[1] for thingy in base_loads if thingy[0] == ID]
            base_load = int(base_load[0] * 365)

            base_load_cost = [entry[1] for entry in base_load_costs if entry[0] == ID]
            base_load_cost = base_load_cost[0]

            balance_temp = [specimen[1] for specimen in balance_temps if specimen[0] == ID]
            balance_temp = balance_temp[0]

            # Extracting normalized consumption df for current ID
            normalised_list = [model for model in norm_consumption if model[0] == ID]

            MWh = int(normalised_list[0][2]['y_pred_norm'].sum())
```

```python
        MWh_cold = int(normalised_list[0][2]['y_pred_cold'].sum())
        MWh_hot = int(normalised_list[0][2]['y_pred_hot'].sum())

        # Specifying layout, unit, format
        pdf = FPDF('P', 'mm', 'A4')
        # Creating a page
        pdf.add_page()
        # Definie color
        color = (40, 140, 39)

        pdf.image('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Pictures/LEAB_logga.png', 0, 0, 80)

        pdf.set_font('helvetica', 'B', 10)
        pdf.cell(0, 2 * pdf.h - 65, 'Landskrona Energi AB   Gasverksgatan 2   kundservice@landskronaenergi.se   0418 - 47 36 20', align='C')

        # Inserting header
        pdf.set_font('helvetica', 'B', 16)
        pdf.set_text_color(*color)
        pdf.set_y(35)
        pdf.cell(0, 20, 'Summary of your key figures', align='L')

        # Reset font and text color for subsequent cells
        pdf.set_font('helvetica', '', 10)  # Default font
        pdf.set_text_color(0, 0, 0)  # Default color

        # Defining input
        info_string = """We have estimated your yearly consumption during a normal year using machine learning to establish the pattern from your consumption during 2023. Prices are for the year of 2023, the power rate and the fixed rate are based on your heat load signature which is calculated from your consumption during the heating season 2022-2023."""

        MWh = "{:,}".format(MWh)
        MWh_cold = "{:,}".format(MWh_cold)
        MWh_hot = "{:,}".format(MWh_hot)
        norm_sum = "{:,}".format(int(norm_sum))
        hot_sum = "{:,}".format(int(hot_sum))
        cold_sum = "{:,}".format(int(cold_sum))
        base_load_cost = "{:,}".format(int(base_load_cost))
        sub_power = "{:,}".format(sub_power)

        customer_IDs = [f'Estimated consumption, normal year: {MWh} MWh',
                        f'Estimated costs, normal year: {norm_sum} SEK',
                        f'Estimated consumption, hot year: {MWh_hot} MWh',
                        f'Estimated costs, hot year: {hot_sum} SEK',
                        f'Estimated consumption, cold year: {MWh_cold} MWh',
                        f'Estimated costs, cold year: {cold_sum} SEK',
```

```
                        f'Estimated annual hot tap water usage: {base_load} MWh',
                        f'Estimated yearly hot water consumption cost: {base_load_cost} SEK',
                        f'Estimated balance temperature: {balance_temp} C ',
                        f'Heat load signature: {sub_power} kW',
                        f'Signature set by: {corr_method}',
                        f'Coefficient of determinartion for regression: {r_squared}',
                        f'Your comparison price is: {comparison_price} kr/MWh']

        if r_squared < 0.75:
            customer_IDs.append(f'Your signature was set by {corr_method} instead as your consumption could not be explained by our regression model.')

        # Set position for multi-cell text
        x = 10
        y = pdf.get_y() + 18  # Add some padding below the previous content

        pdf.set_font('helvetica', '', 10)
        pdf.set_xy(x, y)
        pdf.multi_cell(0, 5.5, info_string)

        # Set position for the first row
        x = 10
        y = pdf.get_y() + 2  # Add some padding below the previous content

        # Inserting the image of the heat load
        image_path_heat_load = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Data/Kundbudgetar/{ID}.png'

        # Read the image to get its dimensions
        image = Image.open(image_path_heat_load)
        image_width, image_height = image.size

        # Add the heat load plot to the PDF
        width = image_width * 0.19
        pdf.image(image_path_heat_load, x=pdf.w - width - x + 5, y=y + 10, w=width)

        prediction_string = 'These are calculated key figures and estimations for your consumption and costs during a normal year:'

        pdf.set_font('helvetica', '', 12)
        pdf.set_xy(x, y)
        pdf.cell(0, y - 60, prediction_string)

        # Starting position for the rectangle and text
        x, y = 12, 80
        row_height = 6.7
        num_entries = len(customer_IDs)

        rect_height = row_height * num_entries + 3
```

```python

        # Draw the rectangle
        pdf.rect(x, y + 6, 75, rect_height)

        # Set font and position for the rows
        pdf.set_font('helvetica', '', 8)

        for customer_id in customer_IDs:
            y += row_height  # Move to the next row
            pdf.set_xy(x, y)
            pdf.cell(0, 10, customer_id)

        string_explain_heat_load = 'Your heat load signature is primarily set by estimating your power at a Dimensioning Winter Outdoor Temperature (DWOT) of -12 C . This estimation is performed by training an algorithm to estimate your power usage as a function of outdoor temperature. If the model is descriptive enough, it is used to establish your heat load signature, otherwise, the heat load signature is set by using your highest measured daily average power. The heat load signature sets the fixed rates for you price model, as our investment increases with the size of your substation.'

        pdf.set_font('helvetica', '', 10)
        y += 15
        pdf.set_xy(x,y)
        pdf.multi_cell(0, 5.5, string_explain_heat_load)

        # Inserting cost table of monthly costs
        image_path_cost_table = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Monthly_costs_table/cost_table_{ID}.png'
        image = Image.open(image_path_cost_table)
        image_width, image_height = image.size

        pdf.add_page()

        pdf.image('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Pictures/LEAB_logga.png', 0, 0, 80)

        # Add cost table to the PDF
        width = image_width * 0.16
        pdf.image(image_path_cost_table, x=pdf.w - width - x - 20, y=y - 95, w=width)

        pdf.set_font('helvetica', 'B', 16)
        pdf.set_text_color(*color)
        pdf.set_xy(x, pdf.get_y() + 30)  # Adjust the height accordingly
        pdf.cell(0, 10, 'Predicted costs for a normal year, by month')

        costs_text = 'Your costs for a normal year has been estimated by analyzing your outdoor temperature dependency from the previous year. The same usage pattern has then been applied to a normal year in order to provide a comparable cost. To complement this, we have included scenarios for colder and hotter years as well, they are located at the end of the document. In the
```

```
 figures below you can see your normal year costs displayed per price model item and per month. You can also see the distribution between fixed and variable costs in the circle diagram. Your comparison price has been calculated as your annual cost divided by your annual heat consumption, it is displayed in relation to the comparison prices of customers in your customer group. We explain what your customer group is on the following page.'

        pdf.set_text_color(0, 0, 0)
        pdf.set_font('helvetica', '', 10)
        pdf.set_xy(x,y - 130)
        pdf.multi_cell(0, 5, costs_text)

        # Inserting pie chart of cost distribution fixed vs variable
        image_path_pie_chart = f"//Users/alexandernuorimaa/Documents/Examensarbete_Progg/pie_charts_cost_posts/{ID}.png"
        image = Image.open(image_path_pie_chart)
        image_width, image_height = image.size

        width = image_width * 0.13
        pdf.image(image_path_pie_chart, x=pdf.w - width - x- 10, y=y + 5, w=width)

        image_path_comparison_price = f"/Users/alexandernuorimaa/Documents/Examensarbete_Progg/comparison_prices/{ID}.png"
        image = Image.open(image_path_comparison_price)
        image_width, image_height = image.size

        width = image_width * 0.15
        pdf.image(image_path_comparison_price, x=pdf.w - width - x - 95, y=y, w=width)

        pdf.set_xy(x, y)
        pdf.set_y(pdf.get_y() + 92)
        pdf.set_font('helvetica', 'B', 10)
        pdf.cell(0, 0, 'Landskrona Energi AB   Gasverksgatan 2   kundservice@landskronaenergi.se   0418 - 47 36 20', align='C')

        pdf.add_page()

        pdf.image('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Pictures/LEAB_logga.png', 0, 0, 80)

        pdf.set_y(pdf.get_y() + 35)

        pdf.set_font('helvetica', 'B', 16)
        pdf.set_text_color(*color)
        pdf.cell(0, 0, 'Your key figures for comparison', align='L')

        cooling_info_string = 'An important part of a District Heating grid is to have a low return temperature, or high cooling in customer substation. This is important since lower return temperatures help extract more energy from the fuel used at the production site and combats wasting resources. In accordance with the Swedish District Heating law, you have been grouped with similar customers on our District Heating grid to provide a
```

```
performance comparison. Comparison within the group is done by evaluating the efficiency of your substation in relation to the rest of the group. This efficiency is measured by how well it extracts heat from the grid, the higher cooling it has, the better the performance. Your customer group is set by your heat load signature and the comparison is provided throug the two figures below. You can see how well your substation cools incoming District Heating compared to the group. Your energy performance certificate to the right compares you to your customer group.'

        pdf.set_y(pdf.get_y() + 7)

        image_path_cooling = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl de_weibull_f rdelat/{ID}.png'
        image = Image.open(image_path_cooling)
        image_width, image_height = image.size
        width = image_width * 0.13
        pdf.image(image_path_cooling, x=pdf.w - width - x - 85, y=y - 85, w=width)

        image_path_energy_performance = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Energy_performance/{ID}.png'
        image = Image.open(image_path_energy_performance)
        image_width, image_height = image.size
        width = image_width * 0.13
        pdf.image(image_path_energy_performance, x=pdf.w - width - x + 10, y=y - 85, w=width)

        pdf.set_font('helvetica', '', 10)
        pdf.set_text_color(0, 0, 0)  # Default color
        pdf.multi_cell(0, 5, cooling_info_string, align='L')

        balance_temp_info = 'Your balance temperature, meaning the temperature where you do not need additional heating, has been individually set, a low balance temperature indicates a good climate shell and low heating need. This has been done by analyzing your heating consumption and evaluating the relation between heat consumption and the outdoor temperature. This is illustrated by the figure to the left below. To the right is a comparison to the rest of the customers in your group.'

        pdf.set_xy(x, y)
        pdf.set_y(y - 5)
        pdf.multi_cell(0, 5, balance_temp_info, align='L')

        try:
            image_path_balance_temp = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Balance_temperatures/{ID}.png'
            image = Image.open(image_path_balance_temp)
            image_width, image_height = image.size
            width = image_width * 0.14
            pdf.image(image_path_balance_temp, x=pdf.w - width - x - 100, y=y + 18, w=width)

            image_path_individual_btemp = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/individual_balance_temp_comparison/{ID}.png'
```

```
                image = Image.open(image_path_individual_btemp)
                image_width, image_height = image.size
                width = image_width * 0.14
                pdf.image(image_path_individual_btemp, x=pdf.w - width - x - 5, y=y + 18, w=width)

            except Exception as e:
                 print('No balance temperature calculated for', ID)

            pdf.set_xy(x, y)
            pdf.set_y(pdf.get_y() + 93)
            pdf.set_font('helvetica', 'B', 10)
            pdf.cell(0, 0, 'Landskrona Energi AB   Gasverksgatan 2   kundservice@landskronaenergi.se   0418 - 47 36 20', align='C')

            pdf.add_page()

            pdf.image('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Pictures/LEAB_logga.png', 0, 0, 80)

            pdf.set_y(pdf.get_y() + 35)

            pdf.set_font('helvetica', 'B', 16)
            pdf.set_text_color(*color)
            pdf.cell(0, 0, 'Estimated consumption and costs for normal year with scenarios', align='L')

            try:
                image_path_norm_costs = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/norm_hot_cold_costs/{ID}.png'
                image = Image.open(image_path_norm_costs)
                image_width, image_height = image.size
                width = image_width * 0.10
                pdf.image(image_path_norm_costs, x=pdf.w - width - x - 110, y=y - 128, w=width)

            except Exception as e:
                 print('No heating load for', ID)

            image_path_consumption_hot_cold_norm = f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/consumption_prognosis_hot_cold_norm/{ID}.png'
            image = Image.open(image_path_consumption_hot_cold_norm)
            image_width, image_height = image.size
            width = image_width * 0.135
            pdf.image(image_path_consumption_hot_cold_norm, x=pdf.w - width - x, y=y - 134, w=width)

            norm_costs_info = 'Both figures above are values given as monthly sums. We have used machine learning to establish a pattern for your heating consumption. This pattern is then used on temperature data for a normal year as well as for scenarios of hotter and colder years, which resulted in the consumption profiles seen to the right above. These consumption predictions are then used with your assigned price models to estimate costs for the coming year, which are shown in the left figure. The table on the first page is a higher resolution of the normal year cost prediction'
```

```
        pdf.set_xy(x, y)
        pdf.set_y(pdf.get_y() - 64)

        pdf.set_font('helvetica', '', 10)
        pdf.set_text_color(0, 0, 0)
        pdf.multi_cell(0, 5, norm_costs_info)

        kotten_header = 'Environmental data and your CO2 emissions'

        pdf.set_xy(x, y)
        pdf.set_y(pdf.get_y() - 31)
        pdf.set_font('helvetica', 'B', 16)
        pdf.set_text_color(*color)
        pdf.cell(0, 0, kotten_header)

        MWh = int(MWh)
        emissions = int(MWh * 0.0598)

        kotten_text = f'The following is environmental data for our DH production for the year of 2022. Our grid has a primary energy factor of 0.05, fossil fuels have a primary energy factor greater than one, recycled heat has a value of 0. Waste and residual products have a value smaller than 1, a small primary energy factor is therefore good. Our grid uses 0.2% fossil fuels in the energy mix and every kWh has an average emission of 59.8 g CO2/kWh. A normal year consumption for your facility therefore emits {emissions} kg of CO2.'

        pdf.set_xy(x, y)
        pdf.set_y(pdf.get_y() - 24)
        pdf.set_font('helvetica', '', 10)
        pdf.set_text_color(0, 0, 0)

        pdf.multi_cell(0, 5, kotten_text)

        pricing_info = 'The basis for your pricing is provided below'

        pdf.set_xy(x, y)
        pdf.set_y(pdf.get_y() + 8)
        pdf.set_font('helvetica', 'B', 16)
        pdf.set_text_color(*color)
        pdf.cell(0, 0, pricing_info)

        pricing_text = f'Your energy rates for winter, autumn/spring and summer respectively: {winter_energy}, {autumn_spring_energy}, {summer_energy} kr/MWh. Flow tax: 3.5 kr/m3. Based on your category {customer_category}, you are charged an annual fixed fee of {yearly_fixed} kr and an additional power fee of {power_tax} kr/kWh based on your heat load signature.'

        pdf.set_xy(x, y)
        pdf.set_y(pdf.get_y() + 14)
        pdf.set_font('helvetica', '', 10)
        pdf.set_text_color(0, 0, 0)

        pdf.multi_cell(0, 5, pricing_text)
```

```python

            communication_text = 'Communication is the future'
            pdf.set_xy(x, y)
            pdf.set_y(pdf.get_y() + 37)
            pdf.set_font('helvetica', 'B', 16)
            pdf.set_text_color(*color)
            pdf.cell(0, 0, communication_text)

            final_text = 'If you have any questions about the information provided in this document, please reach out to discuss them with us. Based on the new analyses we can perform, these measures give us new insight in to what could be optimized for a better performance of your substation which in turn would lead to lower costs and a better user experience for you. There is additional information we could provide you with if you would consider sharing data like heated area for example. Our contact information is available at the bottom of each page, take care.'

            pdf.set_xy(x, y)
            pdf.set_y(pdf.get_y() + 43)
            pdf.set_font('helvetica', '', 10)
            pdf.set_text_color(0, 0, 0)

            pdf.multi_cell(0, 5, final_text)

            pdf.set_xy(x, y)
            pdf.set_y(pdf.get_y() + 91)
            pdf.set_font('helvetica', 'B', 10)
            pdf.cell(0, 0, 'Landskrona Energi AB   Gasverksgatan 2   kundservice@landskronaenergi.se   0418 - 47 36 20', align='C')

            # Saving the PDF
            pdf.output(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/MY_OWN_BUDGETS/my_own_{ID}.pdf')

        except Exception as e:
            print(f"The PDF could not be generated for {ID}, {e}")
```

Listing A.2: Customer budget generator

A.3 Cleaning full year data

```python
import pandas as pd
import numpy as np
import pickle as pkl
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

file_path = '/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Data/FD2_20230101_20231231_204074942.txt'
data = []
non_float_values = []
# Open the file in read mode
with open(file_path, 'r', encoding='ISO-8859-1') as file:
    # Iterate through each line in the file
    for line in file:
```

```python
        # Split the line into a list of values using semicolon (;) as the separator
        values = line.strip().split(';')
        try:
            values[2] = float(values[2])
            data.append(values)
        except Exception as e:
            print(e)
            non_float_values.append(values)

IDs_non_float = [row[0] for row in non_float_values]

# Removing the first entry as it is the header of the column
IDs_non_float = IDs_non_float[1:]

print('Number of customers with blank entries:', len(set(IDs_non_float)), '\n',
     'Number of blank entries:', len(IDs_non_float), '\n',
     'Number of rows remaining:', len(data), '\n',
     'All customer IDs:', set(IDs_non_float))

column_names = ['Plats', 'Datum', 'F rbrukning MWh', '3', '4']
df = pd.DataFrame(data, columns=column_names)
df['Plats'] = df['Plats'].astype(int)
df.drop(columns=['3', '4'], inplace=True)

# Initialize an empty list to store information about duplicates
duplicate_info = []
removed_customers = []

# Iterate over each group
for (ID, _), group_df in df.groupby(['Plats', 'Datum']):
    # If there are multiple entries in the group, indicating duplicates
    if len(group_df) > 1:
        try:
            first_duplicate_index = group_df.index[0]
            try:
                if df.iloc[first_duplicate_index - 5]['Plats'] == ID:
                    avg = df['F rbrukning MWh'].iloc[first_duplicate_index - 5:first_duplicate_index - 1].mean()
                elif df.iloc[first_duplicate_index + 6]['Plats'] == ID:
                    avg = df['F rbrukning MWh'].iloc[first_duplicate_index + 2:first_duplicate_index + 6].mean()
            except Exception as e:
                print(f'{ID} had too few measurements, removing from DataFrame')
                df_removed = df[df['Plats'] == ID]
                removed_customers.append(df_removed)
                df = df[df['Plats'] != ID]

            differences = np.abs(group_df['F rbrukning MWh'] - avg)

            max_difference_index = differences.idxmax()

            removed_entry = group_df.loc[max_difference_index]

```

```
            duplicate_info.append({
                'Plats': ID,
                'Date': group_df['Datum'].iloc[0],
                'Duplicates': group_df,
                'Removed_Entry': removed_entry,
                'Calculated_average': avg
            })

            df = df.drop(max_difference_index)

        except Exception as e:
            print(f'Error: {e} for {ID}')

path_norm = '/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Data/normal_temp.xlsx'
temp_df = pd.read_excel(path_norm)
temp_df['Datum'] = pd.to_datetime(temp_df['Datum'])
temp_df = temp_df[::-1]

logged_removal_energy = []
df_filtered = pd.DataFrame(columns=df.columns)

for ID, original_df in df.groupby('Plats'):
    try:
        original_df['Datum'] = pd.to_datetime(original_df['Datum'])
        original_df = original_df.merge(temp_df, on='Datum', how='left')

        # Assigning feature and target
        X = original_df['Uppm tt temp.'].values.reshape(-1, 1)  # Reshape X for sklearn
        y = original_df['F rbrukning MWh'].values

        # Fit linear regression model
        model = LinearRegression()
        model.fit(X, y)

        # Calculate R^2 score before removing outliers
        r2_before = np.round(model.score(X, y), 3)

        # Predictions and residuals
        y_pred = model.predict(X)
        residuals = y - y_pred

        # Calculate mean and standard deviation of residuals
        mean_residual = np.mean(residuals)
        std_residual = np.std(residuals)

        # Calculate z-scores for residuals
        z_scores = (residuals - mean_residual) / std_residual

        # Define threshold for filtering
        threshold = 4

        # Filter values based on z-scores
        filtered_indices = np.where(np.abs(z_scores) <= threshold)[0]

        # Remove values from y that differ by more than three
```

```
   standard deviations in residual
        df_cleaned = original_df.iloc[filtered_indices]

        df_filtered = pd.concat([df_filtered, df_cleaned], ignore_index=True)

        if len(df_cleaned) < len(original_df):
            # Calculate R^2 score after removing outliers
            r2_after = np.round(model.score(df_cleaned['Uppm tt temp.'].values.reshape(-1, 1),
                                            df_cleaned['F rbrukning MWh'].values), 3)

            # Storing the removed rows
            df_removed = original_df.loc[~(np.abs(z_scores) <= threshold), ['Datum', 'F rbrukning MWh', 'Uppm tt temp.']]

            # Append results to logged_removal list
            logged_removal_energy.append((ID, r2_before, r2_after, len(original_df) - len(df_cleaned), df_removed))

    except Exception as e:
        print(f'Error for {ID}: {e}')

# Creating a list of data frames for grouped customer data
grouped_list = []

for ID, group_df in df_filtered.groupby('Plats'):
    grouped_list.append([ID, group_df])

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/yearly_data.pkl', 'wb') as f:
    pkl.dump(grouped_list, f)
```

Listing A.3: Full year data cleaning

A.4 Calculating balance temperature and base load

```

from statsmodels.nonparametric.smoothers_lowess import lowess
import pandas as pd
import numpy as np
import pickle as pkl
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
import warnings

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/yearly_data.pkl', 'rb') as f:
    dfs = pkl.load(f)

# Suppress all runtime warnings
warnings.filterwarnings("ignore", category=RuntimeWarning)
```

```

balance_temperatures = []

for ID, df in dfs:
    try:
        # Extract heating season data for linear regression
        df_heating = df[df['Datum'].dt.month.isin([1, 2, 3, 10, 11, 12])]
        x_reg = df_heating['Uppm tt temp.'].values.reshape(-1, 1)
        y_reg = df_heating['F rbrukning MWh'].values.reshape(-1, 1)

        # Create lin reg model and train, make predictions and calculate r2
        model = LinearRegression()
        model.fit(x_reg, y_reg)
        y_pred = model.predict(x_reg)
        r2 = r2_score(y_reg, y_pred)

        # Only curve fit and set balance temperature for customers who exhibit a temperature dependent consumption
        if r2 > 0.75:
            # Sorting by outdoor temperature and excluding zeros
            df_sorted = df.sort_values(by='Uppm tt temp.')
            df_sorted = df_sorted[df_sorted['F rbrukning MWh'] != 0]

            x_data = df_sorted['Uppm tt temp.'].values.reshape(-1, 1)  # Reshape to a column vector
            y_data = df_sorted['F rbrukning MWh'].values

            # Perform LOWESS smoothing
            smoothed = lowess(y_data, x_data.ravel(), frac=0.25)

            # Calculate the derivative
            derivative = np.gradient(smoothed[:, 1], smoothed[:, 0])

            # Create a tuple array
            tuple_array = np.array([derivative, x_data.ravel()])

            # Transpose the tuple array to align derivative and x_data
            tuple_array = tuple_array.T

            # Drop rows with NaN values
            filtered_tuple_array = tuple_array[~np.isnan(tuple_array).any(axis=1)]

            # Extract filtered derivative and x_data
            # Excluding the first few measurements as some derivatives are low at the coldest temperatures
            filtered_derivative = filtered_tuple_array[40:, 0]
            filtered_x_data = filtered_tuple_array[40:, 1]

            # Find the largest negative value of the derivative
            max_negative_value = np.min(filtered_derivative)

            # Find the index where the derivative is closest to the desired value
            percentage = 0.3
```

```
            desired_value = max_negative_value * percentage
            index = np.argmax(filtered_derivative > desired_value)

            # Get the corresponding x value
            x_value = filtered_x_data[index]

            balance_temperatures.append((ID, x_value))

            # Plot the original data and the smoothed curve
            plt.scatter(x_data, y_data, label='Original data')
            plt.plot(smoothed[:, 0], smoothed[:, 1], color='red', label='Curve fit')
            plt.scatter(x_value, np.min(y_data) - 0.1, marker="v", color='red', s=75, label=f'Calculated balance temperature: {x_value:.1f} C ')
            plt.ylim(np.min(y_data) - 0.2, np.max(y_data) * 1.1)
            plt.legend()

            plt.grid('On')
            plt.xlabel('Outdoor temperature [ C ]')
            plt.ylabel('Consumption [MWh]')
            plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Balance_temperatures/{ID}.png')
            plt.close()

    except Exception as e:
        print(e, ID)

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/balance_temps.pkl', 'wb') as f:
    pkl.dump(balance_temperatures, f)

import sys
sys.path.append('/Users/alexandernuorimaa/Documents/Examensarbete_Progg')

from My_functions.clustering_by_price_model import group_customers

grouped_IDs = group_customers(balance_temperatures)

villa_IDs = [row[0] for row in grouped_IDs if row[2] == 'Villa']
grp_one_IDs = [row[0] for row in grouped_IDs if row[2] == '1-60 kW']
grp_two_IDs = [row[0] for row in grouped_IDs if row[2] == '61-175 kW']
grp_three_IDs = [row[0] for row in grouped_IDs if row[2] == '176-1400 kW']
grp_four_IDs = [row[0] for row in grouped_IDs if row[2] == '1401- kW']

balance_villa = [row[1] for row in balance_temperatures if row[0] in villa_IDs]
balance_one = [row[1] for row in balance_temperatures if row[0] in grp_one_IDs]
balance_two = [row[1] for row in balance_temperatures if row[0] in grp_two_IDs]
balance_three = [row[1] for row in balance_temperatures if row[0] in grp_three_IDs]
balance_four = [row[1] for row in balance_temperatures if row[0] in
```

```
    grp_four_IDs]

sorted_balance_villa = sorted(balance_villa)
sorted_balance_one = sorted(balance_one)
sorted_balance_two = sorted(balance_two)
sorted_balance_three = sorted(balance_three)
sorted_balance_four = sorted(balance_four)

avg_balance_villa = np.average(sorted_balance_villa)
avg_balance_one = np.average(sorted_balance_one)
avg_balance_two = np.average(sorted_balance_two)
avg_balance_three = np.average(sorted_balance_three)
avg_balance_four = np.average(sorted_balance_four)

try:
     for ID, b_temp in balance_temperatures:
         if ID in villa_IDs:
             avg_bal = np.average(sorted_balance_villa)
             plt.plot(sorted_balance_villa)
             plt.hlines(avg_bal, 0, len(sorted_balance_villa), alpha
    =0.2, color='r', label=f'Average balance temperature: {
    avg_balance_temp:.1f}  C ')

             index = np.searchsorted(sorted_balance_villa, b_temp)
             plt.scatter(index, b_temp, color='red', alpha=0.5, marker
    ='o', label=f'Your balance temperature: {b_temp:.1f}  C ')

         elif ID in grp_one_IDs:
             avg_bal = np.average(sorted_balance_one)
             plt.plot(sorted_balance_one)
             plt.hlines(avg_bal, 0, len(sorted_balance_one), alpha
    =0.2, color='r', label=f'Average balance temperature: {
    avg_balance_temp:.1f}  C ')

             index = np.searchsorted(sorted_balance_one, b_temp)
             plt.scatter(index, b_temp, color='red', alpha=0.5, marker
    ='o', label=f'Your balance temperature: {b_temp:.1f}  C ')

         elif ID in grp_two_IDs:
             avg_bal = np.average(sorted_balance_two)
             plt.plot(sorted_balance_two)
             plt.hlines(avg_bal, 0, len(sorted_balance_two), alpha
    =0.2, color='r', label=f'Average balance temperature: {
    avg_balance_temp:.1f}  C ')

             index = np.searchsorted(sorted_balance_two, b_temp)
             plt.scatter(index, b_temp, color='red', alpha=0.5, marker
    ='o', label=f'Your balance temperature: {b_temp:.1f}  C ')

         else:
             avg_bal = np.average(sorted_balance_three)
             plt.plot(sorted_balance_three)
             plt.hlines(avg_bal, 0, len(sorted_balance_three), alpha
    =0.2, color='r', label=f'Average balance temperature: {
    avg_balance_temp:.1f}  C ')

             index = np.searchsorted(sorted_balance_three, b_temp)
             plt.scatter(index, b_temp, color='red', alpha=0.5, marker
```

```
    ='o', label=f'Your balance temperature: {b_temp:.1f}  C ')

        plt.legend()
        plt.xticks([])
        plt.savefig(f'/Users/alexandernuorimaa/Documents/
    Examensarbete_Progg/individual_balance_temp_comparison/{ID}.png')
        plt.close()

except Exception as e:
    print(e, ID)

for ID, df_base_load in dfs:
    try:
        bal_temp = [row[1] for row in balance_temperatures if row[0]
    == ID]
        bal_temp = bal_temp[0]

        hw_consumption = df_base_load[df_base_load['Uppm tt temp.']
    > bal_temp]
        hw_avg = hw_consumption['F rbrukning MWh'].mean()
        df_base_load['F rbrukning uppv rmning'] = df_base_load['
    F rbrukning MWh'] - hw_avg

        x = df_base_load['Datum']
        y = df_base_load['F rbrukning uppv rmning']

        plt.plot(x, y, label='Heating')

        plt.ylabel('Consumption [MWh]')
        plt.xlabel('Date')
        plt.grid(True)
        plt.legend()

        plt.savefig(f'/Users/alexandernuorimaa/Documents/
    Examensarbete_Progg/heating_plots/{ID}')
        plt.close()

    except Exception as e:
        print('No balance temp for', ID)
```

Listing A.4: Base load calculations

A.5 Calculating cooling, customer grouping and generating energy performance sticker

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import pickle as pkl
from scipy import stats

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/
    Variables/heating_season_data.pkl', 'rb') as f:
    clean_dfs = pkl.load(f)

```

```
import sys
sys.path.append('/Users/alexandernuorimaa/Documents/Examensarbete_Progg')

from My_functions.clustering_by_price_model import group_customers

grouped_IDs = group_customers(clean_dfs)

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/Subscribed_power.pkl', 'rb') as f:
    sub_powers = pkl.load(f)

# Calculating constant for delta T estimation

density = 1000
cp_kj = 4.18
kj_to_MWh = 1 / (3600 * 1000)

constant = 1 / (density * cp_kj * kj_to_MWh)

print(f'The constant is: {constant:.2f}')

delta_Ts = []
zero_flow = []

for ID, delta_df in clean_dfs:
    flow = delta_df['F rbrukning kbm'].sum()
    energy = delta_df['F rbrukning MWh'].sum()

    if flow > 0:
        delta_T = energy * constant / flow

        delta_Ts.append((ID, delta_T))
    else:
        zero_flow.append((ID, energy))

deltas = [row[1] for row in delta_Ts]
deltas_sorted = sorted(deltas)

villa_IDs = [row[0] for row in grouped_IDs if row[2] == 'Villa']
grp_one_IDs = [row[0] for row in grouped_IDs if row[2] == '1-60 kW']
grp_two_IDs = [row[0] for row in grouped_IDs if row[2] == '61-175 kW']
grp_three_IDs = [row[0] for row in grouped_IDs if row[2] == '176-1400 kW']
grp_four_IDs = [row[0] for row in grouped_IDs if row[2] == '1401- kW']

deltas_one = [row[1] for row in delta_Ts if row[0] in grp_one_IDs]
sorted_deltas_one = sorted(deltas_one)
deltas_one_avg = np.average(deltas_one)

for ID, delta_T in delta_Ts:
    if ID in grp_one_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(sorted_deltas_one, alpha=0.75, label='Cooling for customers in your category')
```

```
        index = np.searchsorted(sorted_deltas_one, delta_T)

        # Plot the dot at the corresponding position
        plt.scatter(index, delta_T, color='red', alpha=0.5, marker='o', label=f'Your cooling: {delta_T:.1f}  C ')

        plt.ylabel('Cooling [ C ]')
        plt.xticks([])
        plt.legend()

        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl des_illustration/{ID}')
        plt.close()

deltas_two = [row[1] for row in delta_Ts if row[0] in grp_two_IDs]
sorted_deltas_two = sorted(deltas_two)
deltas_two_avg = np.average(deltas_two)

for ID, delta_T in delta_Ts:
    if ID in grp_two_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(sorted_deltas_two, alpha=0.75, label='Cooling for customers in your category')

        index = np.searchsorted(sorted_deltas_two, delta_T)
        # Plot the dot at the corresponding position
        plt.scatter(index, delta_T, color='red', alpha=0.5, marker='o', label=f'Your cooling: {delta_T:.1f}  C ')

        plt.ylabel('Cooling [ C ]')
        plt.xticks([])
        plt.legend()

        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl des_illustration/{ID}')
        plt.close()

deltas_three = [row[1] for row in delta_Ts if row[0] in grp_three_IDs]
sorted_deltas_three = sorted(deltas_three)
deltas_three_avg = np.average(deltas_three)

for ID, delta_T in delta_Ts:
    if ID in grp_three_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(sorted_deltas_three, alpha=0.75, label='Cooling for customers in your category')

        index = np.searchsorted(sorted_deltas_three, delta_T)
        # Plot the dot at the corresponding position
        plt.scatter(index, delta_T, color='red', alpha=0.5, marker='o', label=f'Your cooling: {delta_T:.1f}  C ')

        plt.ylabel('Cooling [ C ]')
        plt.xticks([])
```

```python
        plt.legend()

        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl des_illustration/{ID}')
        plt.close()

deltas_four = [row[1] for row in delta_Ts if row[0] in grp_four_IDs]
sorted_deltas_four = sorted(deltas_four)
deltas_four_avg = np.average(deltas_four)

for ID, delta_T in delta_Ts:
    if ID in grp_four_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(sorted_deltas_four, alpha=0.75, label='Cooling for customers in your category')

        index = np.searchsorted(sorted_deltas_four, delta_T)
        # Plot the dot at the corresponding position
        plt.scatter(index, delta_T, color='red', alpha=0.5, marker='o', label=f'Your cooling: {delta_T:.1f}  C ')

        plt.ylabel('Cooling [ C ]')
        plt.xticks([])
        plt.legend()

        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl des_illustration/{ID}')
        plt.close()

deltas_villa = [row[1] for row in delta_Ts if row[0] in villa_IDs]
sorted_deltas_villa = sorted(deltas_villa)
deltas_villa_avg = np.average(deltas_villa)

for ID, delta_T in delta_Ts:
    if ID in villa_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(sorted_deltas_villa, alpha=0.75, label='Cooling for customers in your category')

        index = np.searchsorted(sorted_deltas_villa, delta_T)
        # Plot the dot at the corresponding position
        plt.scatter(index, delta_T, color='red', alpha=0.5, marker='o', label=f'Your cooling: {delta_T:.1f}  C ')

        plt.ylabel('Cooling [ C ]')
        plt.xticks([])
        plt.legend()

        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl des_illustration/{ID}')
        plt.close()

x_villa = np.linspace(np.min(deltas_villa), np.max(deltas_villa), 1000)
x_one = np.linspace(np.min(deltas_one), np.max(deltas_one), 1000)
x_two = np.linspace(np.min(deltas_two), np.max(deltas_two), 1000)
```

```python
x_three = np.linspace(np.min(deltas_three), np.max(deltas_three), 1000)

# Calculating probability density function for each distribution
pdf_villa = stats.exponweib.pdf(x_villa, *stats.exponweib.fit(deltas_villa, floc=0))
pdf_one = stats.exponweib.pdf(x_one, *stats.exponweib.fit(deltas_one, floc=0))
pdf_two = stats.exponweib.pdf(x_two, *stats.exponweib.fit(deltas_two, floc=0))
pdf_three = stats.exponweib.pdf(x_three, *stats.exponweib.fit(deltas_three, floc=0))
# pdf_four = stats.exponweib.pdf(x, *stats.exponweib.fit(deltas_four))

# Define the percentiles you want to calculate
percentiles = [1/7, 2/7, 3/7, 4/7, 5/7, 6/7]

# Function to calculate quantiles based on PDF
def calculate_quantiles(pdf, x_values):
    cdf = np.cumsum(pdf) / np.sum(pdf)
    quantiles = [np.interp(p, cdf, x_values) for p in percentiles]
    return quantiles

# Calculate quantiles for each distribution based on PDF
quantiles_all = calculate_quantiles(pdf, x_all)
quantiles_villa = calculate_quantiles(pdf_villa, x_villa)
quantiles_one = calculate_quantiles(pdf_one, x_one)
quantiles_two = calculate_quantiles(pdf_two, x_two)
quantiles_three = calculate_quantiles(pdf_three, x_three)

# Assign quantiles to variables
q1_all, q2_all, q3_all, q4_all, q5_all, q6_all = quantiles_all
q1_villa, q2_villa, q3_villa, q4_villa, q5_villa, q6_villa = quantiles_villa
q1_one, q2_one, q3_one, q4_one, q5_one, q6_one = quantiles_one
q1_two, q2_two, q3_two, q4_two, q5_two, q6_two = quantiles_two
q1_three, q2_three, q3_three, q4_three, q5_three, q6_three = quantiles_three

for index, (ID, delta_T) in enumerate(delta_Ts):
    if ID in villa_IDs:
        if delta_T > q6_villa:
            delta_Ts[index] = (ID, delta_T, 'A')
        elif delta_T > q5_villa:
            delta_Ts[index] = (ID, delta_T, 'B')
        elif delta_T > q4_villa:
            delta_Ts[index] = (ID, delta_T, 'C')
        elif delta_T > q3_villa:
            delta_Ts[index] = (ID, delta_T, 'D')
        elif delta_T > q2_villa:
            delta_Ts[index] = (ID, delta_T, 'E')
        elif delta_T > q1_villa:
            delta_Ts[index] = (ID, delta_T, 'F')
        else:
            delta_Ts[index] = (ID, delta_T, 'G')

    elif ID in grp_one_IDs:
```

```
        if delta_T > q6_one:
            delta_Ts[index] = (ID, delta_T, 'A')
        elif delta_T > q5_one:
            delta_Ts[index] = (ID, delta_T, 'B')
        elif delta_T > q4_one:
            delta_Ts[index] = (ID, delta_T, 'C')
        elif delta_T > q3_one:
            delta_Ts[index] = (ID, delta_T, 'D')
        elif delta_T > q2_one:
            delta_Ts[index] = (ID, delta_T, 'E')
        elif delta_T > q1_one:
            delta_Ts[index] = (ID, delta_T, 'F')
        else:
            delta_Ts[index] = (ID, delta_T, 'G')

    elif ID in grp_two_IDs:
        if delta_T > q6_two:
            delta_Ts[index] = (ID, delta_T, 'A')
        elif delta_T > q5_two:
            delta_Ts[index] = (ID, delta_T, 'B')
        elif delta_T > q4_two:
            delta_Ts[index] = (ID, delta_T, 'C')
        elif delta_T > q3_two:
            delta_Ts[index] = (ID, delta_T, 'D')
        elif delta_T > q2_two:
            delta_Ts[index] = (ID, delta_T, 'E')
        elif delta_T > q1_two:
            delta_Ts[index] = (ID, delta_T, 'F')
        else:
            delta_Ts[index] = (ID, delta_T, 'G')

    elif ID in grp_three_IDs:
        if delta_T > q6_three:
            delta_Ts[index] = (ID, delta_T, 'A')
        elif delta_T > q5_three:
            delta_Ts[index] = (ID, delta_T, 'B')
        elif delta_T > q4_three:
            delta_Ts[index] = (ID, delta_T, 'C')
        elif delta_T > q3_three:
            delta_Ts[index] = (ID, delta_T, 'D')
        elif delta_T > q2_three:
            delta_Ts[index] = (ID, delta_T, 'E')
        elif delta_T > q1_three:
            delta_Ts[index] = (ID, delta_T, 'F')
        else:
            delta_Ts[index] = (ID, delta_T, 'G')
    else:
        delta_Ts[index] = (ID, delta_T, 'A')

# Define data
labels = ['G', 'F', 'E', 'D', 'C', 'B', 'A']  # Reversed order for A at the top
lengths = [21, 18, 15, 12, 9, 6, 3]  # Lengths of the staples
colors = np.linspace(0, 1, len(labels))  # Colors from green to red

# GENERATING ENERGY PERFROMANCE STICKERS
for ID, delta_T, grade in delta_Ts:
    try:
```

```python
        # Plot the stapled plot
        fig, ax = plt.subplots(figsize=(8, 6))

        for i, (label, length, color) in enumerate(zip(labels, lengths, colors)):
            # Set line width based on whether label matches grade
            linewidth = 28 if label == grade else 22

            # Plot bars
            ax.plot([0, length + 0.04], [i, i], color='black', linewidth=linewidth)  # Border
            ax.plot([0, length], [i, i], color=plt.cm.RdYlGn(color), linewidth=20)  # Bar
            ax.text(length + 1.3, i, label, va='center', ha='center', fontsize=18)  # Add label

        # Customize plot
        ax.set_yticklabels([])  # Hide y-axis labels
        ax.set_xticklabels([])  # Hide x-axis labels
        ax.tick_params(axis='both', length=0)  # Remove ticks
        ax.spines['right'].set_visible(False)  # Hide right spine
        ax.spines['top'].set_visible(False)  # Hide top spine
        ax.spines['bottom'].set_visible(False)  # Hide bottom spine
        ax.set_xlim(0, max(lengths) + 3)  # Set x-axis limits
        ax.set_ylim(-0.5, len(labels) - 0.5)  # Set y-axis limits
        ax.grid(False)  # Hide grid lines

        # Save plot
        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Energy_performance/{ID}')
        plt.close()

    except Exception as e:
        print(e, 'for:', ID)

for ID, delta_T, grade in delta_Ts:
    if ID in villa_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(x_villa, pdf_villa, label='Distribution of cooling for customers in your category')

        y_coordinate = pdf_villa[np.abs(x_villa - delta_T).argmin()]  # Evaluating PDF at delta_T

        plt.vlines(delta_T, ymin=0, ymax=y_coordinate, color='red', linestyle='--', alpha=0.5, label=f'Your cooling: {delta_T:.1f}  C')

        plt.ylabel('Probability')
        plt.xlabel('Cooling [ C ]')
        plt.legend()
        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl de_weibull_f rdelat/{ID}')
        plt.close()

for ID, delta_T, grade in delta_Ts:
    if ID in grp_one_IDs:
```

```
        plt.figure(figsize=(8, 6))

        plt.plot(x_one, pdf_one, label='Distribution of cooling for customers in your category')

        y_coordinate = pdf_one[np.abs(x_one - delta_T).argmin()]  # Evaluating PDF at delta_T

        plt.vlines(delta_T, ymin=0, ymax=y_coordinate, color='red', linestyle='--', alpha=0.5, label=f'Your cooling: {delta_T:.1f}  C')

        plt.ylabel('Probability')
        plt.xlabel('Cooling [ C ]')
        plt.legend()

    plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl de_weibull_f rdelat/{ID}')
        plt.close()

for ID, delta_T, grade in delta_Ts:
    if ID in grp_two_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(x_two, pdf_two, label='Distribution of cooling for customers in your category')

        y_coordinate = pdf_two[np.abs(x_two - delta_T).argmin()]  # Evaluating PDF at delta_T

        plt.vlines(delta_T, ymin=0, ymax=y_coordinate, color='red', linestyle='--', alpha=0.5, label=f'Your cooling: {delta_T:.1f}  C')

        plt.ylabel('Probability')
        plt.xlabel('Cooling [ C ]')
        plt.legend()
        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl de_weibull_f rdelat/{ID}')
        plt.close()

for ID, delta_T, grade in delta_Ts:
    if ID in grp_three_IDs:
        plt.figure(figsize=(8, 6))

        plt.plot(x_three, pdf_three, label='Distribution of cooling for customers in your category')

        y_coordinate = pdf_three[np.abs(x_three - delta_T).argmin()]  # Evaluating PDF at delta_T

        plt.vlines(delta_T, ymin=0, ymax=y_coordinate, color='red', linestyle='--', alpha=0.5, label=f'Your cooling: {delta_T:.1f}  C')

        plt.ylabel('Probability')
        plt.xlabel('Cooling [ C ]')
        plt.legend()
```

```python
        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/fl de_weibull_f rdelat/{ID}')
        plt.close()
```

Listing A.5: Cooling calculations

A.6 XGBoost training on consumption pattern

```python
import pandas as pd
import pickle as pkl
import matplotlib.pyplot as plt

path_norm = '/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Data/normal_temp.xlsx'
temp_df = pd.read_excel(path_norm)
temp_df['Datum'] = pd.to_datetime(temp_df['Datum'])
temp_df = temp_df[::-1]
temp_df.head()

temp_df_m = temp_df.copy()

temp_df_m['Datum'] = pd.to_datetime(temp_df_m['Datum'])
temp_df_m.set_index('Datum', inplace=True)
temp_df_m = temp_df_m.resample('M').mean()

temp_df_m['CF_deg'] = temp_df_m['Uppm tt temp.'] / temp_df_m['Normaltemp.']
temp_df_m['CF_ind'] = temp_df_m['Energi-Index, aktuellt'] / temp_df_m['Energi-Index, normalt']

temp_df_m.plot(y=['CF_deg', 'CF_ind'], kind='bar')

temp_df.reset_index(inplace=True)
temp_df_m.reset_index(inplace=True)


# Extract month number from the date column in both DataFrames
temp_df['M nad'] = temp_df['Datum'].dt.month
temp_df_m['M nad'] = temp_df_m['Datum'].dt.month

# Merge the monthly averages DataFrame with the original DataFrame based on the month
temp_df_merged = pd.merge(temp_df, temp_df_m[['M nad', 'CF_ind', 'CF_deg']], on='M nad', how='left')

# Drop the 'Month' column from the merged DataFrame
temp_df_merged.drop(columns=['M nad'], inplace=True)

# Display the merged DataFrame
print(temp_df_merged)

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/yearly_data.pkl', 'rb') as f:
    yearly_df = pkl.load(f)

import xgboost as xgb
```

```python
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt

trained_models = []

for item in yearly_df:
    try:
        # Extract ID, usage data, temperature data, and dates
        ID = item[0]
        usage_merged = item[1]

        # Prepare features and target variables
        y = usage_merged['F rbrukning MWh'] # Target variable

        # Preparing normal temperature data, with a hotter and colder year
        X_norm = usage_merged['Normaltemp.'].values.reshape(-1, 1)
        X_cold = X_norm - 2
        X_hot = X_norm + 2

        # Split the data into training and testing sets
        X_train, X_test, y_train, y_test = train_test_split(X_norm, y, test_size=0.2, random_state=42)

        # Train the XGBoost model
        model = xgb.XGBRegressor()
        model.fit(X_train, y_train)

        # Make predictions
        y_pred = model.predict(X_test)

        # Evaluate the model
        mse = mean_squared_error(y_test, y_pred)

        # Train the XGBoost model on actual temperature data and usage data
        model = xgb.XGBRegressor()
        model.fit(X_train, y_train)

        # Make predictions on actual temperature data
        y_pred_actual = model.predict(X_test)

        # Make predictions on new temperature data
        y_pred_norm = model.predict(X_norm)
        y_pred_cold = model.predict(X_cold)
        y_pred_hot = model.predict(X_hot)

        # Save y_pred_norm along with dates as a DataFrame
        y_pred_df = pd.DataFrame({'Datum': usage_merged['Datum'], 'y_pred_norm': y_pred_norm, 'y_pred_cold': y_pred_cold, 'y_pred_hot': y_pred_hot})

        # Create a figure and axes objects
        fig, ax1 = plt.subplots(figsize=(8, 5))

        # Plot actual data and predictions on new temperature data on ax1
```

```
        ax1.plot(usage_merged['Datum'], y, label='Outcome 2023')
        ax1.plot(usage_merged['Datum'], y_pred_norm, label='Prognosis for normal year')

        # Set labels and title for ax1
        ax1.set_xlabel('Date')
        ax1.set_ylabel('Consumption [MWh]')
        ax1.legend(loc='upper left')

        # Create a secondary y-axis for temperature data
        ax2 = ax1.twinx()

        # Plot temperatures on ax2
        ax2.plot(usage_merged['Datum'], usage_merged['Uppm tt temp.'], color='gray', linestyle='--', label='Measured temperature 2023')
        ax2.plot(usage_merged['Datum'], usage_merged['Normaltemp.'], color='black', linestyle='--', label='Normal temprature')

        # Set label for ax2
        ax2.set_ylabel('Temperature [ C ]')
        ax2.legend(loc='upper right')

        # Rotate x-axis labels for better readability
        plt.xticks(rotation=45)

        # Adjust layout to prevent clipping of labels
        fig.tight_layout()

        # Show the plot
        plt.savefig(f'/Users/alexandernuorimaa/Documents/Examensarbete_Progg/plot_f rbrukning_norm/{ID}')
        plt.close()

        save_model = [ID, model, y_pred_df]

        trained_models.append(save_model)

    except Exception as e:
        print(f"Error processing data for ID {ID}: {e}")

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/usage_models.pkl', 'wb') as f:
    pkl.dump(trained_models, f)

with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/base_loads.pkl', 'rb') as f:
    base_loads = pkl.load(f)

CF_deg = temp_df_merged['CF_deg']
CF_ind = temp_df_merged['CF_ind']

CF_deg = CF_deg.replace(0, 1)
CF_ind = CF_ind.replace(0, 1)

# Making plots of prediction of energy consumption for ML, energy index normalisation and degree day normalisation
for ID, df in yearly_df[:5]:
```

```
    trained_model = [row for row in trained_models if ID == row[0]]
    y_base = [row[1] for row in base_loads if row[0] == ID]

    y_pred_norm = trained_model[0][2]['y_pred_norm']
    x = df['Datum']
    y_heat = df['F rbrukning MWh'] - y_base

    # Reset indices
    y_heat = y_heat.reset_index(drop=True)
    CF_deg = CF_deg.reset_index(drop=True)
    CF_ind = CF_ind.reset_index(drop=True)

    y_norm_deg = y_heat / CF_deg + y_base
    y_norm_deg = y_norm_deg.dropna()

    y_norm_ind = y_heat / CF_ind + y_base
    y_norm_ind = y_norm_ind.dropna()

    plt.figure(figsize=(8, 5))
    plt.plot(x, y_norm_deg, label='Degree days', color='blue', linestyle='--')
    # plt.plot(x, df['F rbrukning MWh'], label='Utfall 2023', color='purple', linestyle='--')
    plt.plot(x, y_norm_ind, label='Energy index', color='red', linestyle='--')
    plt.plot(x, y_pred_norm, label='ML', color='green', linestyle='--')

    # Calculate total sum for each plot
    sum_deg = int(y_norm_deg.sum())
    sum_ind = int(y_norm_ind.sum())
    sum_ml = int(y_pred_norm.sum())

    # Add text box with total sum to the top middle of the plot
    plt.text(0.5, 0.9, f'Total degree days: {sum_deg} MWh\nTotal energy index: {sum_ind} MWh\nTotal ML: {sum_ml} MWh',
             horizontalalignment='center', verticalalignment='center', transform=plt.gca().transAxes, bbox=dict(facecolor='white', alpha=0.5))

    plt.grid('On')
    plt.legend()
    plt.xlabel('Date')
    plt.ylabel('Consumption [MWh]')

# Making plots of prediction of energy consumption for ML, energy index normalisation and degree day normalisation
for ID, df in yearly_df:
    trained_model = [row for row in trained_models if ID == row[0]]

    y_pred_norm = trained_model[0][2]['y_pred_norm']
    y_pred_cold = trained_model[0][2]['y_pred_cold']
    y_pred_hot = trained_model[0][2]['y_pred_hot']

    x = df['Datum']

    plt.figure(figsize=(8, 5))
    plt.plot(x, y_pred_norm, label='Normal year', color='green',
```

```
linestyle='--')
    plt.plot(x, y_pred_cold, label='2 C  colder year', color='blue',
linestyle='--')
    plt.plot(x, y_pred_hot, label='2 C  hotter year', color='red',
linestyle='--')

    # Calculate total sum for each plot
    sum_norm = int(y_pred_norm.sum())
    sum_cold = int(y_pred_cold.sum())
    sum_hot = int(y_pred_hot.sum())

    # Add text box with total sum to the top middle of the plot
    plt.text(0.5, 0.9, f'Total normal year: {sum_norm} MWh\nTotal
cold year: {sum_cold} MWh\nTotal hot year: {sum_hot} MWh',
             horizontalalignment='center', verticalalignment='center'
, transform=plt.gca().transAxes, bbox=dict(facecolor='white',
alpha=0.5))

    plt.legend()
    plt.grid('On')
    plt.xlabel('Date')
    plt.ylabel('Consumption [MWh]')

    plt.savefig(f'/Users/alexandernuorimaa/Documents/
Examensarbete_Progg/consumption_prognosis_hot_cold_norm/{ID}')
    plt.close()
```

Listing A.6: Consumption pattern training

A.7 Price model generation

```
# The extrapolated subscribed power is used as input
def generate_price_model(subscribed_power):
    """
    Generate price model DataFrames based on the subscribed power for
     each customer.

    Args:
        subscribed_power (list): List of DataFrames, where each
DataFrame represents the data for a specific customer.

    Returns:
        price_models (list): List of tuples containing the price
model DataFrames for each customer, with yearly rate, customer ID'
s and the power base.
    """
    # Relevant imports
    import pandas as pd
    import pickle

    print("Generating price models")

    price_models = [] # 'price model', 'yearly rate', 'ID', 'power
basis'

    # Iterate over each customer DataFrame
```

```
    for entry in subscribed_power:
        ID, sub_power, corr_method, r_squared = entry

        # Creating price model DataFrame with dates from the DataFrame
        price_model = pd.DataFrame()
        price_model['Date'] = pd.date_range(start='2023-01-01', end='2023-12-31', freq='D')

        # Create boolean masks for winter, summer, and spring/autumn months
        winter_mask = price_model['Date'].dt.month.isin([11, 12, 1, 2, 3])
        summer_mask = price_model['Date'].dt.month.isin([6, 7, 8])

        energy_price = [285, 545, 100] # kr/MWh (Spring/autumn, winter, summer)
        flow_tax = [0, 3.5, 0] # kr/m^3
        power_taxes = [788, 665, 540, 464] # kr/kW
        base_price = [400, 8000, 27506, 119059] # kr/ r

        # Use boolean masks to assign power and base taxes throughout the year
        price_model['Energy price'] = float(energy_price[0])
        price_model.loc[winter_mask, 'Energy price'] = float(energy_price[1])
        price_model.loc[summer_mask, 'Energy price'] = float(energy_price[2])

        # Boolean masks to assign flow taxes
        price_model['Flow tax'] = float(flow_tax[0])
        price_model.loc[winter_mask, 'Flow tax'] = float(flow_tax[1])
        price_model.loc[summer_mask, 'Flow tax'] = float(flow_tax[2])

        # Assigning power tax [sek/kW] based on subscribed power
        power_tax = 0
        base = 0
        if sub_power <= 60:
            power_tax = power_taxes[0]
            base = base_price[0]
            power_basis = '0-60 kW'
        elif sub_power <= 175:
            power_tax = power_taxes[1]
            base = base_price[1]
            power_basis = '61-175 kW'
        elif sub_power <= 1400:
            power_tax = power_taxes[2]
            base = base_price[2]
            power_basis = '176-1400 kW'
        elif sub_power > 1400:
            power_tax = power_taxes[3]
            base = base_price[3]
            power_basis = '>1400 kW'

        price_model['Power tax'] = power_tax

        # Append the price model DataFrame to the list of price models
```

```
            price_models.append([price_model, base, ID, power_basis])

    # Save variables to a file
    with open("/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/price_models.pkl", "wb") as f:
        pickle.dump((price_models), f)

    print("Done calculating price models, wihoooo!")

    return price_models
```

Listing A.7: Price model generation

A.8 Assigning customer group by heat load signature

```
def group_customers(customer_dfs):
    """
    Groups customers based on what price model they are asigned by leab.

    Args:
        customer_dfs (list): List of ID's and customer usage DataFrames.

    Returns:
        grp_IDs (list): List of ID's with corresponding csutomer category.
    """

    import pandas as pd
    import numpy as np
    import pickle as pkl

    with open('/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Variables/Subscribed_power.pkl', 'rb') as f:
        sub_powers = pkl.load(f)

    path_name = '/Users/alexandernuorimaa/Documents/Examensarbete_Progg/Data/LEAB_EFFEKTSIGNATUR.xlsx'
    industry_subs = pd.read_excel(path_name)
    ID_industry = industry_subs['Plats-ID']
    ID_industry = ID_industry.to_list()

    grp_IDs = [('Plats', 'sub_power', 'category')]

    for row in customer_dfs:
        ID = row[0]
        sub_power = [row[1] for row in sub_powers if row[0] == ID]
        try:
            sub_power = sub_power[0]
            if ID in ID_industry:
                if sub_power <= 60:
                    grp_IDs.append((ID, sub_power, '1-60 kW'))
                elif sub_power <= 175:
```

```
                    grp_IDs.append((ID, sub_power, '61-175 kW'))
                elif sub_power <= 1400:
                    grp_IDs.append((ID, sub_power, '176-1400 kW'))
                else:
                    grp_IDs.append((ID, sub_power, '1401- kW'))
            else:
                grp_IDs.append((ID, sub_power, 'Villa'))
        except Exception as e:
            print(e, ID, sub_power, row)

    return grp_IDs
```

Listing A.8: Customer grouping by subscribed power

Appendix B

Pricing terms of LEAB

Prisvillkor för näringsverksamhet i Landskrona

Prisvillkoren gäller från och med 2019-01-01 och tillsvidare för fjärrvärmeleveranser, från Landskrona Energi AB, som används i näringsverksamhet eller annan likartad verksamhet, exempelvis bostadsrättsföreningar. I detta dokument behandlas prisvillkor för fjärrvärme, i övriga frågor hänvisar vi till nu gällande "Allmänna avtalsvillkor för leverans av fjärrvärme som används i näringsverksamhet".

Fjärrvärmepriset för näringsidkare består av en ***Effektdel*** (grundavgift och effektavgift) och en ***Energidel*** (energiavgift samt flödestaxa).

Effektdel

Grundavgift
Grundavgiften är en fast årlig avgift som periodiseras över kalenderårets dygn och kunden debiteras fast avgift för motsvarande antal dygn på varje månadsfaktura. Avgiften baseras på debiterad dygnsmedeleffekt.

Effektavgift - definition
Dygnsmedeleffekt definieras som dygnsanvändning (kWh) dividerat med 24h, effekten uttrycks i kW. Effektbehovet baseras på uppmätt dygnsmedeleffektuttag då uppvärmningsbehov förestår. Uppmätt dygnsmedeleffekt ställs mot aktuell dygnsmedeltemperatur, vilket ger ett samband mellan utetemperatur och fastighetens effektbehov. Tillsammans bildar dessa värden en för fastigheten unik effektsignatur som utgör grunden för hur stor den debiterande dygnsmedeleffekten och grundavgiften blir.

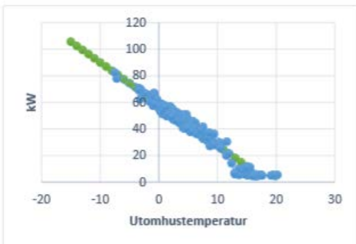
Debiterande dygnsmedeleffekt
Landskrona Energi kommer per den 1 januari varje kalenderår att revidera den debiterade dygnsmedeleffekt som fastigheten har behov av vid utetemperaturen -12°C. Kunden ska informeras om den nya debiterande effekten i samband med information om eventuella förändringar gällande pris och prisvillkor för kommande kalenderår.

Med debiterad dygnsmedeleffekt avses den effekt som kan förväntas behövas vid dimensionerande utetemperatur, -12°C. Den debiterade dygnsmedeleffekten beräknas med hjälp av en linjär analys av kundens effektanvändning. Beräkningen baseras på dygnsmedelvärden för fastigheten som uppmätts under närmast föregående vinterperiod då fastigheten uppvisat uppvärmningsbehov.

Den debiterade dygnsmedeleffekten avrundas till närmaste heltal. Om förändringen är 5 % eller större, alternativt 3 kW eller större, mot föregående år sätts en ny debiterande dygnsmedeleffekt för kommande år. Kostnaden för debiterad dygnsmedeleffekt periodiseras över kalenderårets dygn och kunden debiteras effektdelen för motsvarande antal dygn på varje månadsfaktura.

Om det av någon anledning inte går att fastställa en rimlig debiterad dygnsmedeleffekt enligt linjär analys, om t.ex. fastigheten är nyansluten, eller om det saknas historiska data, sker prissättningen enligt en debiterad dygnsmedeleffekt framräknad enligt följande prioritering:

a. Enligt linjär analys
b. Justerad datamängd för den linjära analysen inkluderande historiska data
c. Nyckeltal för liknande fastigheter/verksamheter
d. Enligt högsta uppmätta dygnsmedeleffekt



Energidel

Energipris
Inom respektive period gäller energipris enligt vid var tid gällande prislista.
Kostnaden för energi baseras på faktisk uppmätt energiförbrukning och faktureras en gång i månaden.

Energipriset är uppdelat på tre perioder:

vinterperiod	(januari - mars och november - december)
vår- och höstperiod	(april - maj, september - oktober)
sommarperiod	(juni - augusti)

Flödestaxa
Flödestaxan är baserad på hur mycket fjärrvärmevatten som behöver pumpas igenom anläggningen. Det vill säga ju effektivare anläggning desto mindre flöde och kostnad, vilket medför ett effektivare fjärrvärmenät. Flödestaxan gäller enbart under vinterperioden, november till mars. Flödet debiteras, månadsvis, enligt gällande prislista och baseras på en kostnad kr / m^3, (kronor per kubikmeter).

Avläsning och debitering
Avläsning och debitering görs med hjälp av fjärravläsning en gång per månad, i övrigt enligt § 4 och § 5 i "Allmänna avtalsvillkor för leverans av fjärrvärme som används i näringsverksamhet".

Provning av mätare
Vid misstanke om felaktig mätning kan provning av mätare begäras enligt § 4 i Allmänna avtalsvillkor för leverans av fjärrvärme som används i Näringsverksamhet. Om avvikelsen från rätt värde inte är större än +/- 5 % debiteras en avgift, kontakta Landskrona Energi för vidare information.

Anslutningsavgift
Anslutningsavgift baseras på effektuttag och byggkostnad, dvs. kostnaden för anläggning/anslutning av fjärrvärme. Vid förändringar av framdragning, ökad kapacitet, eller annan förändring ombesörjer leverantören sådan åtgärd, mot att leverantörens kostnader och/eller anslutningsavgift erläggs av kunden. I övrigt enligt § 3 i "Allmänna avtalsvillkor för leverans av fjärrvärme som används i näringsverksamhet".

Byggvärme
För nybyggnadsprojekt kan under vissa förutsättningar byggvärme användas. Särskilda villkor gäller, kontakta Landskrona Energi för vidare information.

Markvärme
Då möjlighet finns kan vi leverera returvärme för markvärme enligt särskilda förutsättningar, kontakta Landskrona Energi för vidare information.

Förhandling och medling enligt fjärrvärmelagen
(SFS 2008:263) Enligt fjärrvärmelagen har kunden rätt att begära förhandling och under vissa förutsättningar ansöka om medling angående priset för fjärrvärme eller kapaciteten hos en anslutning till fjärrvärmeverksamheten.

Övriga villkor
För fjärrvärmeleveransen gäller "Allmänna avtalsvillkor för Näringsidkare"

Appendix C

Evaluating calculated heat load signatures

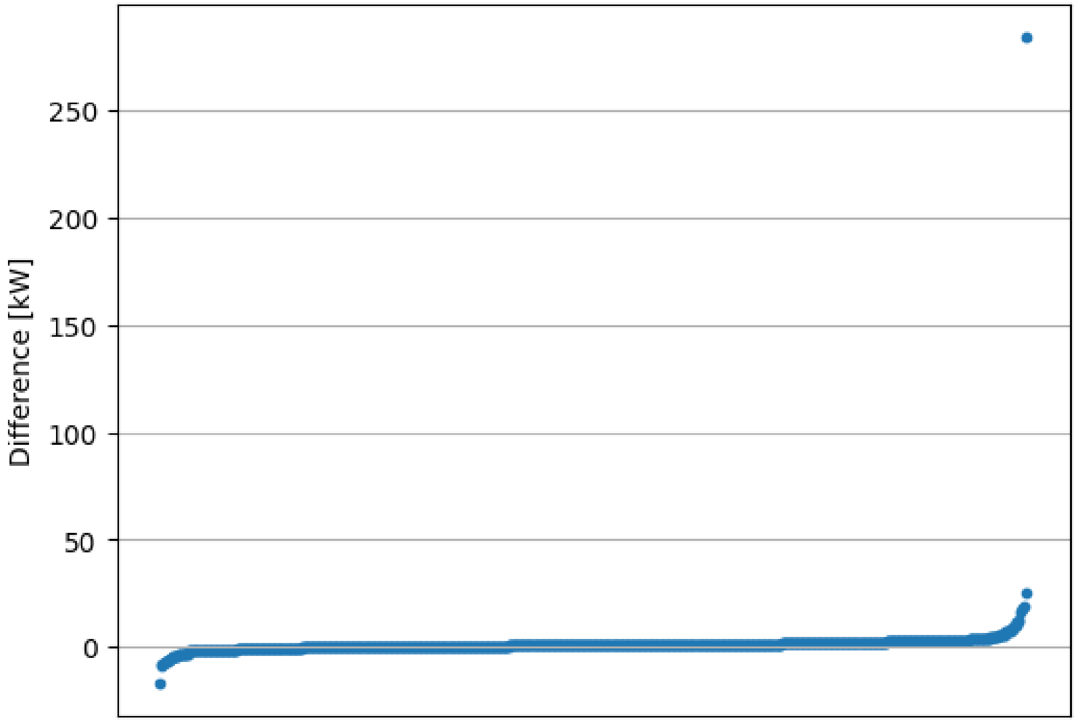


Figure C.1: Differences in calculated heat load signatures for those that had it assigned by linear regression vs LEAB

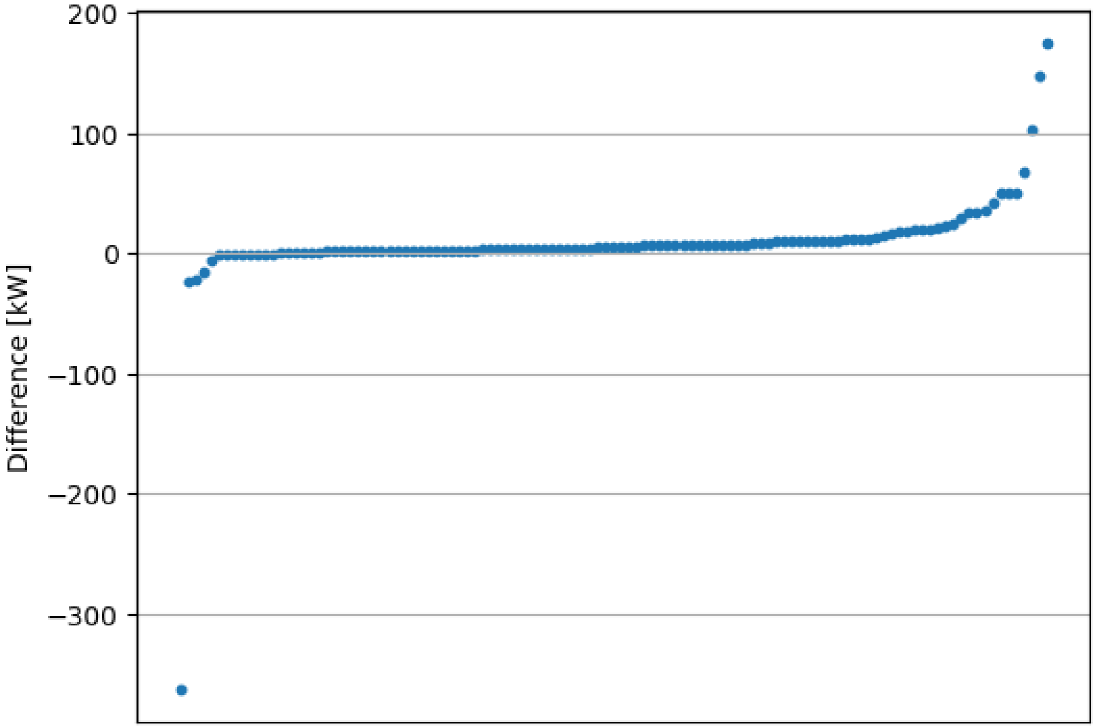


Figure C.2: Differences in calculated heat load signatures for those that had it assigned by highest measured daily average power vs LEAB

Appendix D

Correlation matrices for industry customers and villa customers

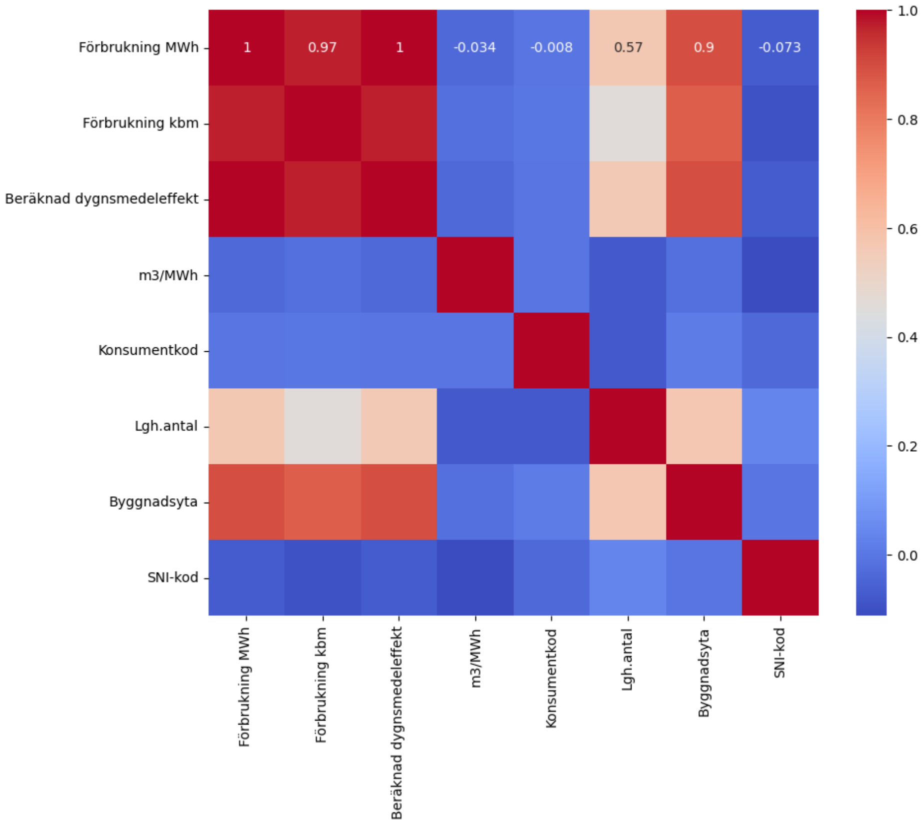


Figure D.1: Correlation matrix for the whole industry customer subgroup

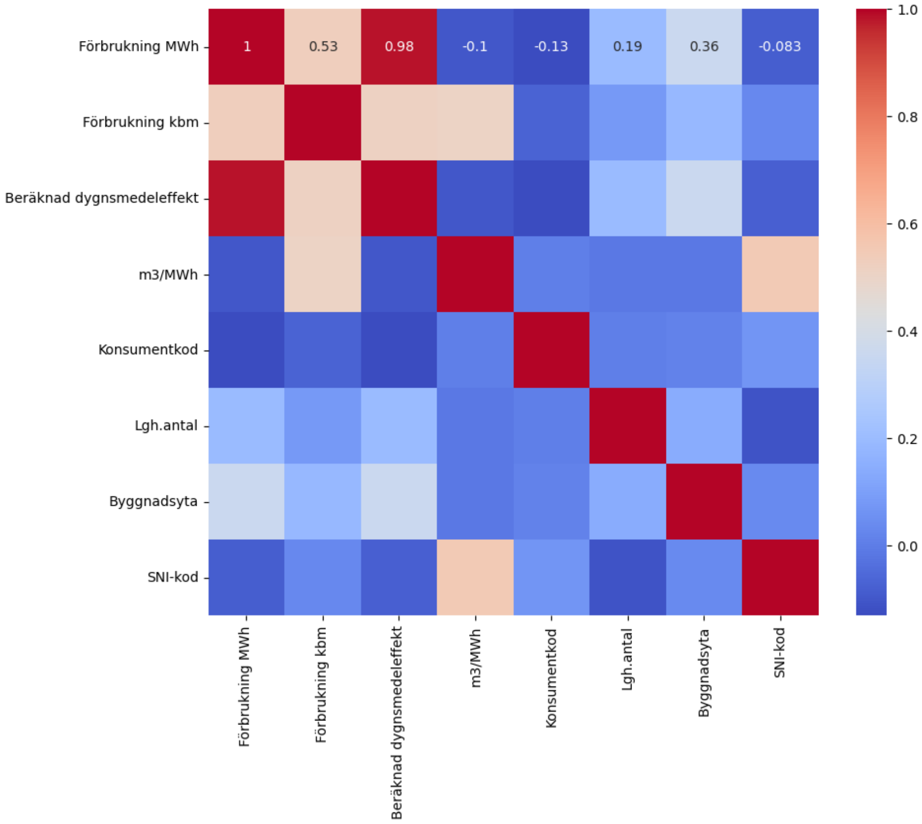


Figure D.2: Correlation matrix for the villa customer subgroup

D.1 Translation table for correlation matrices

Table D.1: List of Swedish terms and their English translations

Swedish	**English**
Förbrukning MWh	Consumption MWh
Förbrukning kbm	Consumption m^3
Beräknad dygnsmedeleffekt	Calculated daily average power
m^3/MWh	m^3/MWh
Konsumentkod	Consumer code
Lgh. antal	Number of apartments
Byggnadsyta	Building area
SNI-kod	Swedish Standard Industrial Classification Code

Appendix E

Different thresholds for setting balance temperatures

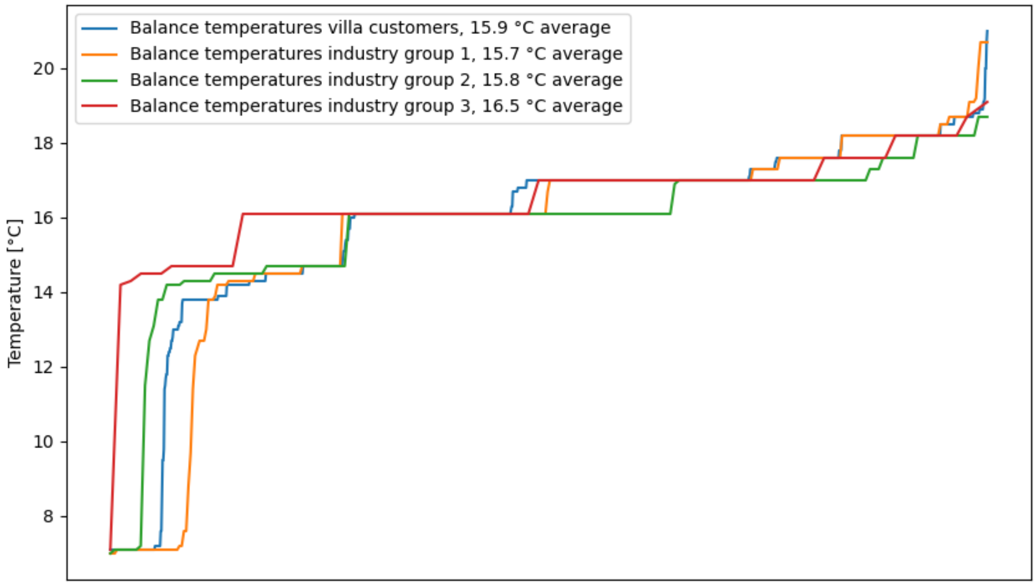


Figure E.1: Using 20 percent of greatest heat power load slope as a threshold for setting the balance temperature

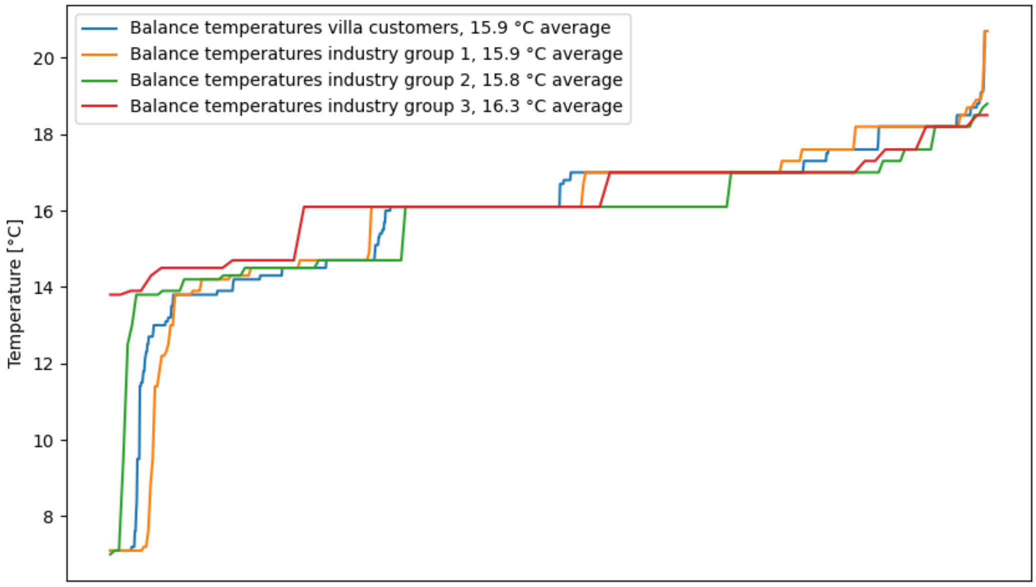


Figure E.2: Using 25 percent of greatest heat power load slope as a threshold for setting the balance temperature

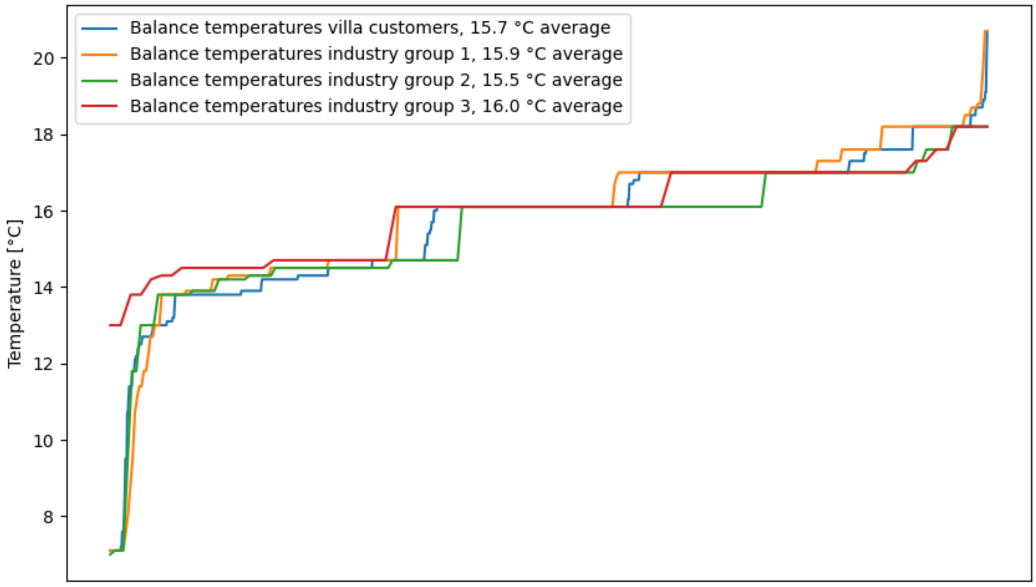


Figure E.3: Using 30 percent of greatest heat power load slope as a threshold for setting the balance temperature, this was the threshold used for balance temperature estimations

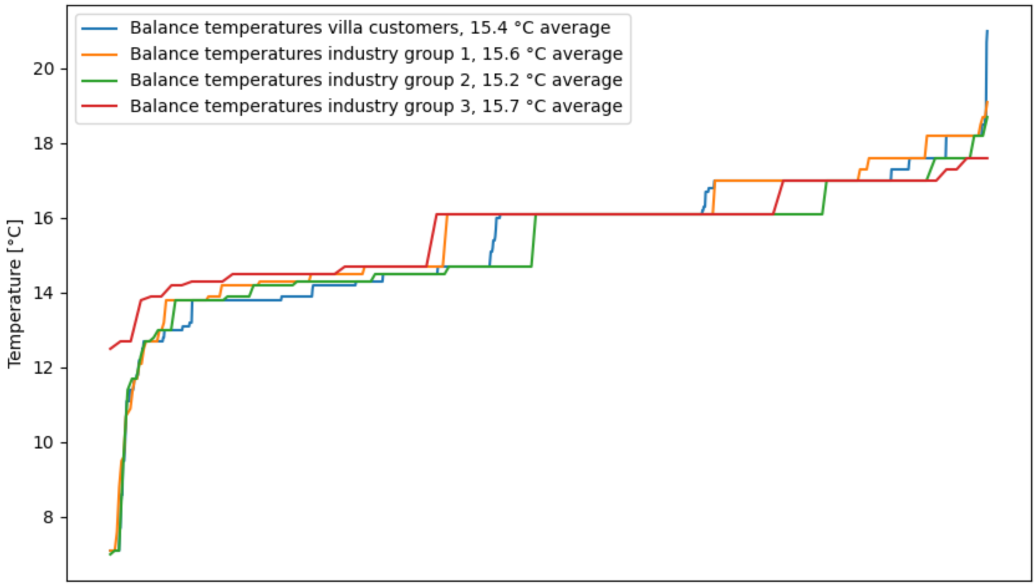


Figure E.4: Using 35 percent of greatest heat power load slope as a threshold for setting the balance temperature

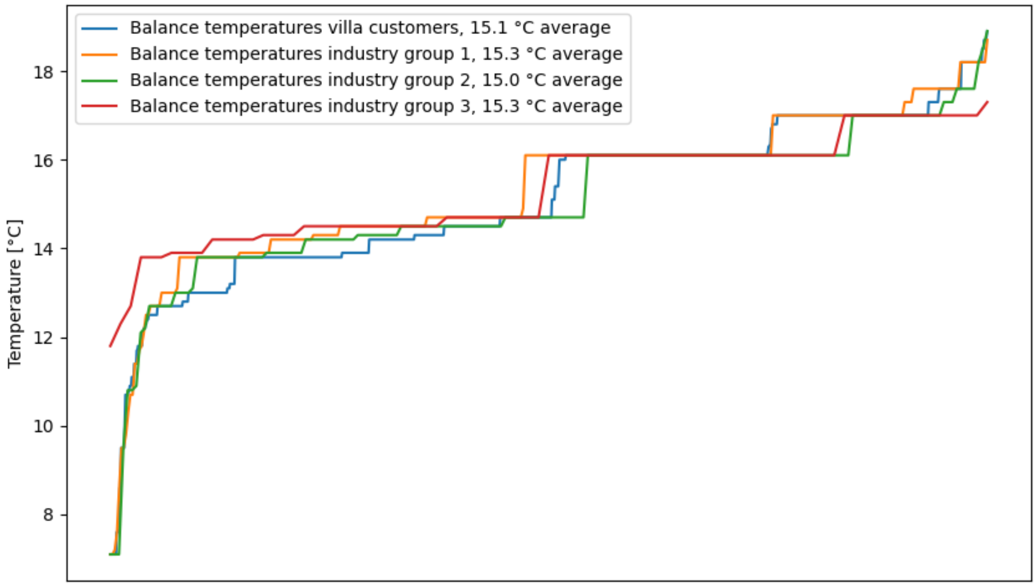


Figure E.5: Using 40 percent of greatest heat power load slope as a threshold for setting the balance temperature

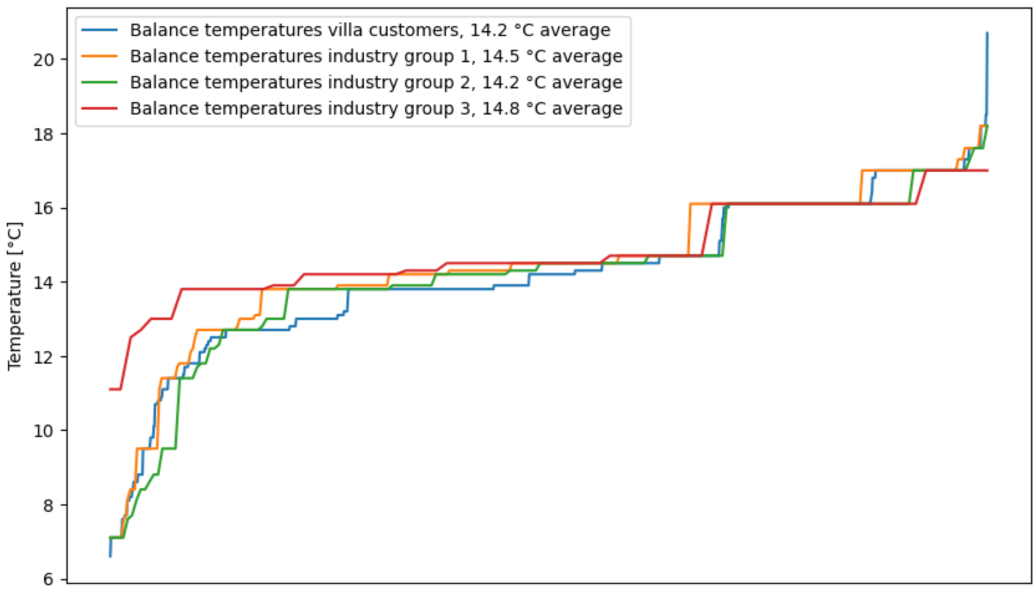


Figure E.6: Using 50 percent of greatest heat power load slope as a threshold for setting the balance temperature

Appendix F

Contract terms of LEAB for industry customers

ALLMÄNNA AVTALSVILLKOR

för leverans av fjärrvärme som används i näringsverksamhet

Dessa allmänna avtalsvillkor har den 9 mars 2023 fastställts av Värmemarknadskommittén, ett samarbetsorgan mellan Energiföretagen Sverige och Fastighetsägarna Sverige, HSB Riksförbund, Hyresgästföreningen, Riksbyggen samt Sveriges Allmännytta.

1. Inledande bestämmelser

1.1. Dessa allmänna avtalsvillkor avser leverans av fjärrvärme som används i näringsverksamhet eller annan likartad verksamhet (exempelvis bostadsrättsföreningar). För leverans av fjärrvärme som används för enskilt bruk liksom för leveranser av energi från kunder till fjärrvärmenätet, gäller andra villkor.

1.2. Om inte annat avtalats skriftligen med kunden ska dessa allmänna avtalsvillkor tillämpas. Denna bestämmelse återkommer i ett antal villkor i dessa allmänna avtalsvillkor, men denna upprepning är inte avsedd att ha annan innebörd än en påminnelse om bestämmelsen i denna punkt.

1.3. Om kunden innehar fastigheten eller del av fastigheten med annan rätt än äganderätt ska vad som föreskrivs i dessa villkor gälla i tillämpliga delar. Kunden ska tillse att leverantören gentemot fastighetsägaren tillförsäkras samma rättigheter som om kunden varit fastighetens ägare.

Definitioner

1.4. I dessa avtalsvillkor avses med

- fastighet: fastighet som kunden äger eller har nyttjanderätt till inklusive byggnader och andra anläggningar oavsett om byggnaderna eller anläggningarna utgör fast eller lös egendom.

- fjärrvärmecentral: aggregat som överför värme från leverantörens anläggning till fastighetens värmesystem.

- kundens anläggning: anläggning för mottagande och distribution av värme, som kunden äger eller har nyttjanderätt till, inom fastigheten efter leveransgräns.

- leveransgräns: gräns mellan leverantörens anläggning och kundens anläggning.

- leverantörens anläggning: anläggning för produktion eller distribution av fjärrvärme, som leverantören äger eller har nyttjanderätt till, fram till leveransgräns eller till tredje man. I leverantörens anläggning ingår också mätare och sådan kommunikationsutrustning för mätvärden som leverantören äger.

- mätare: instrument som är utformat för mätning av värmeenergi enligt definition i vid var tid gällande föreskrift från Energimarknadsinspektionen.

- mätpunkt (leveranspunkt enligt EiFS 2022:4) punkt där värmeenergin förs över från fjärrvärmenätet till fjärrvärmekundens anläggning.

- skriftligen: med skriftligen avses i dessa allmänna avtalsvillkor meddelande skickat per ordinarie post, via e-post eller annat elektroniskt medel till av kunden uppgiven kontaktuppgift.

2. Anläggning

Gemensamma bestämmelser

2.1. Leverantören bestämmer tryck, temperatur, andra dimensioneringsdata för fjärrvärmeleverans samt tekniskt utförande för fjärrvärmecentral och leverantören har rätt att förändra dessa. Om sådana förändringar skulle medföra behov av åtgärder för att bibehålla fjärrvärmecentralens funktion ska kostnaderna ersättas av leverantören, med avdrag för den värdestegring som utbytet medför, beräknat på skillnaden i värde för den utbytta och nyanskaffade utrustningen.

2.2. Part får inte använda sin anläggning så att skada eller störning kan uppkomma på motpartens anläggning eller för andra kunder.

2.3. Part ska utan dröjsmål till motparten anmäla driftstörningar, läckage och andra liknande omständigheter som kan beröra motpartens anläggning samt sådana omständigheter som kan påverka de förhållanden som avtalats.

Leverantörens anläggning

2.4. Leverantören drar, efter samråd med kunden, fram erforderliga ledningar till en av leverantören vald leveransgräns och i förekommande fall även ledningar till tredje man.

Om kunden önskar annan leveransgräns, annat läge eller ökad kapacitet kan leverantören ombesörja sådan åtgärd mot ersättning för de merkostnader som därvid uppkommer.

2.5. Inkoppling till leverantörens anläggning får inte utföras utan leverantörens tillstånd.

2.6. Leverantören har rätt att inom fastigheten installera och bibehålla mätare, reglerutrustning för effektbegränsning jämte annan för kontroll av energianvändningen erforderlig utrustning samt i förekommande fall fjärrvärmecentral och tillhörande utrustning. Kunden bekostar driftel för dessa anläggningar. Plats för utrustningen väljs av leverantören, efter samråd med kunden, och ska upplåtas utan kostnad för leverantören. Mätaren, som tillhandahålls av leverantören, förblir dennes egendom och får endast hanteras av denne. I en nödsituation har kunden rätt att stänga ventiler vid leveransgräns, även om de tillhör leverantören.

Kunden ska se till att utrustning enligt ovan är lätt åtkomlig för leverantören.

2.7. I de fall leverantören har installerat en mätare med signalutgång ska kunden ges möjlighet till egen registrering, förutsatt att det inte medför en säkerhetsrisk för leverantören eller andra olägenheter. Inkoppling ska ske enligt leverantörens anvisning.

Kundens anläggning
2.8. Kunden installerar, bekostar, äger samt svarar för underhållet av kundens anläggning.

2.9. Alla förekommande arbeten på kundens anläggning som väsentligt kan påverka avtalade tryck, temperaturer och andra dimensioneringsdata, ska utföras enligt leverantörens bestämmelser.

2.10. Den part som bekostat för mätaren erforderlig utrustning, såsom el- och rörinstallation för mätaren, mätartavla, mätarskåp och mätarledningar, bekostar även underhåll och utbyten av dessa.

2.11. Leverantörens inkoppling eller kontroll befriar inte kunden från ansvar för installationen eller skyldighet att rätta till brister i denna.

3. Upplåtelse av utrymme för leverantörens anläggningar samt tillträde till dessa
3.1. Kunden är skyldig att till leverantören upplåta det utrymme inom fastigheten som behövs för leverantörens anläggning.

3.2. Kunden ska på begäran av leverantören utan särskild ersättning teckna servituts- eller nyttjanderättsavtal eller medverka till att leverantören erhåller ledningsrätt, för leverantörens anläggning på fastigheten. Kunden ska medverka till att tecknat servituts- eller nyttjanderättsavtal skrivs in i fastighetsregistret. Kostnaden för inskrivning och ledningsförrättning betalas av leverantören.

3.3. Vad som avses i punkterna 3.4–3.8 gäller inte i den utsträckning parterna har träffat servitutsavtal, nyttjanderättsavtal eller om leverantören har erhållit ledningsrätt, med annan innebörd än vad som framgår av nämnda punkter.

3.4. Leverantören har rätt att få tillträde till fastigheten för utförande av arbeten på leverantörens anläggning, såsom avläsning/mätning, installations-, reparations-, service- och underhållsarbeten, om inte annat särskilt har avtalats. Häri ingår även en rätt för leverantören att fälla träd och buskar, som utgör fara eller hinder för leverantörens anläggning. Nämnda arbeten ska om möjligt utföras under vardagar mellan kl. 08.00 och kl.18.00. Kunden ska underrättas om arbeten minst fem arbetsdagar i förväg. Detta gäller dock inte avhjälpande av fel och brister vars åtgärdande inte kan anstå. Leverantören ska på egen bekostnad vidta de åtgärder som behövs för att minimera störningar.

Berör arbeten direkt bärande byggnadsdelar eller för fastighetens funktion viktiga installationer som tillhör kunden, får arbetet inte utföras utan kundens godkännande. Vid akut felavhjälpning får dock åtgärd vidtas utan godkännande. Leverantören ska i sådant fall utan dröjsmål underrätta kunden om åtgärden.

3.5. Kunden ska på begäran mot kvitto överlämna de nycklar, koder och liknande som behövs för tillträde till fastigheten. Leverantören ska på ett betryggande sätt förvara och får inte till obehöriga utlämna vad som sålunda överlämnats. Försummar leverantören sina skyldigheter enligt denna punkt ska denne ersätta kunden för eventuell skada jämte kostnad för byte av lås eller liknande som är en följd av denna försummelse.

3.6. Kunden får inte utan leverantörens samtycke uppföra byggnad, ändra marknivån eller vidta annan åtgärd inom fastigheten som hindrar eller försämrar funktionen eller driften av leverantörens anläggning eller försvårar tillträde till denna.

3.7. Om kunden inom fastigheten kräver flyttning eller annan ändring av leverantörens anläggning, ska leverantören medverka till detta, om inte väsentligt hinder föreligger. Av flyttningen eller ändringen föranledda kostnader ska betalas av kunden.

Om flyttningen eller ändringen föranleds av från myndighet eller kommunalt organ framställt åtgärdskrav på fastigheten (till exempel krav på hissanordning, nödutgång eller dylikt) och avser berörd del av leverantörens anläggning för distribution till tredje man, är leverantören skyldig att på egen bekostnad ombesörja sådan flyttning eller ändring inom fastigheten av denna del om inte väsentligt hinder föreligger mot flyttningen eller ändringen. Vad som nu anges beträffande leverantörens kostnadsansvar gäller inte om kravet på åtgärd från myndighet eller kommunalt organ är föranlett av kundens om- eller tillbyggnad av fastigheten eller av annan åtgärd vidtagen av kunden. Kunden ska i sådant fall bära kostnaden för flyttningen eller ändringen.

3.8. Om leverantörens anläggning inte är i funktion och inte heller i framtiden ska användas, ska leverantören på egen bekostnad avlägsna anläggningen om kunden begär det och anläggningen innebär väsentligt men för kunden och det inte är oskäligt att avlägsna anläggningen. I första hand ska dock plombering av leverantörens anläggning övervägas.

4. Mätning, insamling av mätvärden, information om energianvändningen, faktureringsinformation och fakturering
Mätning
4.1. Av kunden använd värmeenergi registreras av leverantörens mätare.

4.2. För de fall mätaren har en flödesmätare med ett eget räkneverk och ett integreringsverk och registreringarna i dessa inte överensstämmer ska flödesmätarens registrering gälla.

4.3. Om part har skäl att ifrågasätta mätarens funktion ska denne utan oskäligt dröjsmål informera den andra parten om detta.

4.4. Kunden har rätt att begära kontroll av mätaren. Leverantören ska då informera kunden om den beräknade kostnaden för kontrollen och princip för betalningsansvar. Kontrollen ska verkställas av leverantören, som ska intyga att kontrollen har utförts av opartisk sakkunnig.

Om det vid kontrollen konstateras, att mätarens registrering avviker mer än vad som godtas enligt vid var tid gällande myndighetsföreskrifter, ska mätvärdena rättas och fjärrvärmeleveransen ska anses motsvara de rättade värdena. Kontrollen ska i sådant fall bekostas av leverantören.

Om det vid kontrollen konstateras att mätarens registrering är godtagbar, ska kunden ersätta leverantören med högst de kostnader som denne har haft för kontrollen.

Insamling av mätvärden
4.5. Leverantören ska för varje mätpunkt samla in mätvärden efter varje leveransperiods slut.

Insamling enligt första stycket ska ske med regelbundna intervall och minst en gång per månad.

Leverantören ska, utöver vad som anges i första och andra stycket, för varje mätpunkt samla in mätvärden vid

1. nyanslutning,

2. vid permanent frånkoppling,

3. vid mätarbyte,

4. vid avflyttning och

5. vid inflyttning.

Insamling ska även ske vid partsbyte eller nytecknande av leveransavtal.

Insamling av mätvärden ska ske genom fjärravläsning om inte annat följer av vid var tid gällande föreskrift från Energimarknadsinspektionen.

Information om energianvändningen
4.6. Leverantören ska senast 15 dagar efter leveransperiodens slut enligt punkten 4.5 ovan lämna information till kunden om mängden levererad värmeenergi för leveransperioden.

Leverantören ska tillhandahålla mätvärden på sätt som leverantören beslutar, exempelvis via internetbaserad media (Mina sidor), via faktura, sms, e-post, genom pappersutskrift eller enligt överenskommelse med kunden. Tidpunkten för insamling av mätvärden ska framgå av informationen om energianvändningen.

Fakturering
4.7. Fakturering av uppmätt levererad värmeenergi ska ske i efterskott och ska baseras på insamlade och rapporterade mätvärden. Om inte annat har avtalats med kunden ska fakturering ske minst en gång per kvartal.

Leverantören ska erbjuda kunden elektronisk faktura.

4.8. Vid prisförändring får det nya priset tillämpas från den beräknade mätarställningen, vid den tidpunkt då det nya priset träder i kraft.

Om mätvärde saknas eller har brister
4.9. Om ett faktureringsgrundande mätvärde från en mätare saknas eller har brister eller om fel har skett vid insamling av mätvärden, ska leverantören, oavsett vad som anges ovan, basera fakturering på beräknad energianvändning.

Leverantören ska meddela kunden att faktureringen är grundad på beräknad energianvändning och hur mätvärdet har beräknats. Leverantören ska även till kunden lämna information om orsaken till att mätvärde från mätare inte kunnat erhållas vid normal insamling av mätvärden.

4.10. I de fall beräknad energianvändning utgör grund för fakturering ska leverantören vid beräkningen utgå från kundens tidigare uppmätta energianvändning och användningsprofil samt övriga kända omständigheter. Fakturan ska innehålla uppgifter om hur mätvärdet har beräknats.

4.11. Om fel har uppstått vid mätning, insamling av mätvärden, beräkning av energianvändning eller fakturering ska korrigering ske, dock inte för längre tid tillbaka än tre år från det att felet blev känt av båda avtalsparterna. Om någon part uppenbarligen har känt till omständigheter av betydelse för mätning, beräkning av energianvändning eller fakturering, utan att underrätta den andra parten, får dock korrigering ske för längre tid.

Faktureringsinformation
4.12. Kundens faktura ska utformas på ett tydligt sätt. Fakturan ska innehålla de uppgifter som anges i vid var tid gällande föreskrift från Energimarknadsinspektionen.

Information om historisk energianvändning
4.13 Leverantören ska utan särskild kostnad för kunden tillhandahålla uppgifter om energianvändning som minst omfattar de senaste tre åren eller leveransavtalets löptid, om denna är kortare. Dessa uppgifter ska motsvara de intervaller för vilka faktureringsinformation har framställts.

4.14 Leverantören ska utan särskild kostnad för kunden även tillhandahålla uppgifter om energianvändning per dag, vecka, månad och år för en period som minst omfattar de senaste två åren eller leveransavtalets löptid, om denna är kortare.

4.15 Information om historisk användning enligt 4.13 och 4.14 ska göras tillgänglig varje kvartal om kunden begär det och i annat fall minst två gånger per år. Informationen lämnas på sätt som leverantören beslutar, exempelvis via internetbaserat media (Mina sidor), via faktura, sms, e-post, genom pappersutskrift eller enligt överenskommelse med kunden.

5. Betalning och säkerhet
5.1. Betalning ska vara leverantören tillhanda senast på den i leverantörens faktura angivna förfallodagen, vilken infaller tidigast 30 dagar efter det att leverantören har avsänt fakturan.

5.2. Sker inte betalning i rätt tid har leverantören rätt att av kunden, förutom fakturabeloppet, fordra ränta enligt räntelagen (1975:635) och ersättning för de kostnader som är förenade med dröjsmålet. Hit räknas även kostnader för betalningspåminnelse och inkassokrav.

5.3. Om leverantören har anledning att befara att kunden inte kommer att fullgöra sina betalningsförpliktelser, har leverantören rätt att begära godtagbar säkerhet eller förskottsbetalning för fortsatt leverans.

6. Hinder för leveransavtalets fullgörande
Part är inte skyldig att fullgöra leveransavtalet om fullgörandet väsentligt försvåras till följd av hinder som denne inte råder över. Som hinder räknas krig, myndighetsbeslut,

omfattande driftstörningar, arbetskonflikt, brand, störningar i allmänna transportväsendet eller annat av part ej vållat förhållande som väsentligt inverkar på leveransavtalets fullgörande och som part inte kunnat förutse och vars menliga inverkan part rimligen inte kunnat undanröja.

7. Avstängning, avbrott och begränsning av leverans
7.1. Försummar kunden att betala förfallna avgifter eller att i annat hänseende fullgöra sina skyldigheter enligt leveransavtal och försummelsen inte är av ringa betydelse, får leverantören stänga av leveransen till kunden, om leverantören inte genom skriftlig anmaning har kunnat åstadkomma rättelse. Avstängning får dock inte ske om detta medför hälsorisk för tredje man.

7.2. Avser försummelsen betalning ska kunden ges skälig tid, minst 15 dagar räknat från skriftlig anmaning, att betala innan avstängning får ske.

7.3. Återinkoppling ska ske först när anledning till avstängning inte längre föreligger samt leverantörens kostnader för avstängning och återinkoppling ersatts av kunden.

7.4. Leverantören har rätt att avbryta leverans av fjärrvärme för att genomföra en åtgärd som syftar till; att undvika personskada eller omfattande sakskada, eller för att säkerställa en god leveranssäkerhet. Leverantören har även rätt att tillfälligt avbryta leveransen för att bygga ut fjärrvärmeverksamheten.

7.5. Om fjärrvärme endast kan levereras i begränsad omfattning ska tillgänglig fjärrvärme fördelas mellan kunder på objektiva grunder. Leverantören har rätt att i eller i anslutning till fjärrvärmecentralen montera utrustning för sådan fördelning.

Om leverantören önskar montera utrustningen i kundens anläggning ska leverantören inhämta kundens godkännande. Om kunden inte godkänner montering i kundens anläggning gäller följande:

Om leverantören istället monterar utrustningen utanför kundens anläggning ska kunden ersätta leverantören med hälften av den eventuella dokumenterade merkostnaden, dock högst med ett belopp som inte överstiger hälften av vid var tid gällande prisbasbelopp enligt socialförsäkringsbalken, exklusive moms.

Ersättning utgår endast om:

a. Kunden efter minst tre kontaktförsök inte svarat på leverantörens begäran om att installera utrustning i kundens anläggning, eller
b. Kunden nekar leverantörens begäran utan godtagbart skäl. Godtagbart skäl kan till exempel vara att installation i kundens anläggning har negativ påverkan på av annan part utställda garantier eller orsakar annan allvarlig olägenhet för kunden.

7.6. Om part kan förutse annat än kortvarigt avbrott i leveransen, respektive leveransens mottagande, ska motparten underrättas i god tid genom personligt meddelande, e-post, via anslag eller på annat lämpligt sätt.

8. Ersättning för skada

8.1. Leverantören är skyldig att ersätta skada som denne åstadkommer på kundens fastighet eller anläggning genom vårdslöshet vid utförande av arbete på leverantörens anläggning. Leverantören ansvarar även för skada som på grund av underlåten eller bristfällig vård av leverantörens anläggning uppstår på fastigheten eller kundens anläggning.

8.2. Om skada uppkommer på kundens mark i samband med tillsyn, underhåll, ombyggnad och/eller reparation av leverantörens anläggning ska leverantören återställa marken i ursprungligt eller likvärdigt skick. Om marken inte kan återställas till ursprungligt eller likvärdigt skick och om detta innebär skada för kunden som inte är av ringa betydelse ska leverantören utge skälig ersättning för skadan.

8.3. Kunden har rätt till ersättning för skada till följd av sådan störning eller inskränkning i leveransen, som inte beror på någon av i punkterna 7.1, 7.4 och 7.5 angivna omständigheter om skadan beror på leverantörens vårdslöshet.

8.4. Leverantören utger endast ersättning för person- eller sakskada. Rätten till ersättning omfattar inte ren förmögenhetsskada eller följdskada vid person- eller sakskada, om inte sådan skada förorsakats genom grov vårdslöshet från leverantörens sida.

8.5. Kunden ska utan dröjsmål och senast inom två år från det att skadan inträffade underrätta leverantören om anspråk på ersättning. Om så inte sker har kunden förlorat sin rätt till ersättning för skadan.

8.6. Den skadelidande parten ska vidta skäliga åtgärder för att begränsa sin skada. Försummar han det kan ersättningen reduceras i motsvarande mån.

8.7. Om leverantören anlitar entreprenör för att helt eller delvis fullgöra sitt åtagande mot kund, inklusive installations-, reparations-, service- och underhållsarbeten, ska leverantören ansvara för entreprenörens arbete så som för eget arbete.

9. Giltighet och ändring av allmänna avtalsvillkor
9.1. Dessa allmänna avtalsvillkor gäller tills vidare.

9.2. Leverantören har rätt att ersätta dessa allmänna avtalsvillkor med ny version under förutsättning att ändringarna beslutats i samråd mellan ovan angivna branschorganisationer eller om det följer av lag eller annan författning. Underrättelse ska vid beslutad ändring av dessa allmänna avtalsvillkor ske minst två månader före ikraftträdandet.

9.3. Om inte annat avtalats upphör leveransavtalet tidigast tre månader efter skriftlig uppsägning från kunden eller 12 månader efter skriftlig uppsägning från leverantören. Saklig grund ska föreligga vid uppsägning från leverantörens sida. Med saklig grund avses bland annat väsentligt avtalsbrott från kundens sida eller betydande försämring av leverantörens affärsmässiga förutsättningar för att tillhandahålla fjärrvärme enligt leveransavtal, som inte endast är tillfällig.

10. Prisändringar
10.1. Om inte annat har avtalats ska prisändringar inte ske oftare än en gång per år. Prisändringar får inte ske retroaktivt. Leverantören har rätt att ensidigt ändra gällande prisvillkor. Leverantören ska skriftligen meddela kunden om ändringen senast två månader före den tidpunkt de ändrade villkoren ska börja gälla.

10.2. Prisändringar enligt 10.1 får dock ske oftare än en gång per år om det motiveras av externa faktorer – såsom skatte- och avgiftsförändringar, force majeure eller väsentliga förändringar orsakade av myndighetsbeslut som förändrar förutsättningarna för prissättningen.

11. Förhandling och medling enligt fjärrvärmelagen (SFS 2008:263)
Enligt fjärrvärmelagen har kunden rätt att begära förhandling och under vissa förutsättningar ansöka om medling angående priset för fjärrvärme eller kapaciteten hos en anslutning till fjärrvärmeverksamheten.

12. Avtalsöverlåtelse
Kunden får inte överlåta detta avtal utan leverantörens skriftliga godkännande.

13. Tvist

Tvist ska avgöras av svensk allmän domstol.