

TECHNICAL UNIVERSITY OF DENMARK



APPLIED STATISTICS AND STATISTICAL SOFTWARE (02441)

HTK Case: Energy performance of buildings

JANUARY 17, 2023

AUTHORS

Marie Murmann Kragh
S203566

Lasse Schnell Danielsen
S203512

Johanna Munch Haraldsdottir
S204657

Summary

The goal of this report is to identify public buildings in Høje Taastrup municipality with lacking insulation. To determine this, data describing building energy consumption and weather conditions were investigated, through an statistical analysis.

The analysis was based on a physical model. This approach allowed for the estimation of the insulation of a building by considering both the temperature difference between both sides of the wall and the building heat loss. The building heat loss was estimated by the normalized energy consumption. To model this, a general linear model (GLM) was constructed. Here additional factors were considered, such as humidity, wind-speed, and whether it was weekend or not.

The results of the statistical analysis revealed a large difference in the heat consumption between the public buildings. An ordered list of all the buildings ranked by their insulation was presented, and it was recommended that the buildings with both poor insulation and a high energy consumption should be prioritized for improvement. Building "69469107" was identified as an ideal candidate, since it was both large and one of the top 5 worst insulated buildings.

In addition to this, a difference was found between the energy consumption on working days and weekends, suggesting that some buildings turn off the heat on weekends while others do not. Therefore, it is recommended that more buildings in the municipality, if possible, turn down the heat when not used.

Contents

1	Introduction	1
2	Description of Data	2
2.1	Descriptive analysis	2
3	Data cleansing	6
4	Statistical Analysis	8
4.1	Model assumption and selection	9
5	Results	11
5.1	Interactions	12
5.2	Design/regression matrix and ranking of buildings	13
6	Conclusion	16

1 Introduction

Høje Taastrup municipally is committed to reducing energy consumption in its public building. As a cost-saving measure, the municipality aims to prioritize retrofitting efforts on those buildings that have the poorest energy performance. This is identified through a statistical analysis. Through the application of physics, it is known that the heat loss through a simple wall can be calculated using the formula:

$$Q_{heat} = U_A(T_{indoor} - T_{outdoor}) \quad (1)$$

A generalized linear model is used to estimate the average amount of insulation, U_A for the entire building, while taking climatic variables into account. The report aims to provide valuable insight to guide the municipality's retrofitting efforts.

2 Description of Data

This case investigates how different variables affect heat consumption in public buildings, with the aim of renovating the ones with the worst amount of insulation. The experiment was conducted for 97 buildings, represented by the ID. The consumption has been noted for each building on each date. The consumption is a mean of multiple readings for each building on each day. All consumption data has been made from meters with a minimum of 121 records. Consumption is the one dependent variable as it is the target of investigation, being dependent on the 12 remaining independent variables (Table 1).

Table of data		
Date	The date of the measurement	format DDMMYY
ID	Uniuqe ID given to each building	83-level factor
Consumption	Difference in heat consumption between daily readings	Continuos variable [MWh]
Temp	Temperature outside	Continuous variable
Dew_pt	Dew point	Continuous variable
Hum	Humidity level	Continuous variable
Wind_spd	Wind speed	Continuous variable
Dir	Wind direction	16 - level factor
Vis	Visibility	Continuous variable
Pressure	Air pressure	Continuous variable
Cond	Weather condition	10 -level factor
Fog	Noticeable fog	2-level factor
Rain	Noticeable rain	2 - level factor

Table 1: Table of the dependent variable "consumption" and the 12 independent variables from the data provided.

2.1 Descriptive analysis

To present the data the temperature is plotted as a function of the consumption below (figure 1). A clear grouping by color, which represents different buildings(ID), can be seen. From equation 1 is a linear relationship between heat consumption and temperature difference seen, which could explain the grouping. A linear relationship is also illustrated below (Figure 2). Where it is only focused on one building with the ID; 78185925. the variance in residual distance seems quite high (Figure 1). When comparing with the illustration of all buildings (Figure 1) one notice that an increase in the slope, carries a higher likeliness of bigger variance in residual distance and therefore more scattered data points. One thing which could be causing this problem is the differences in the size of the buildings. Generally it also seems that temperature difference has an influence on the consumption at each building.

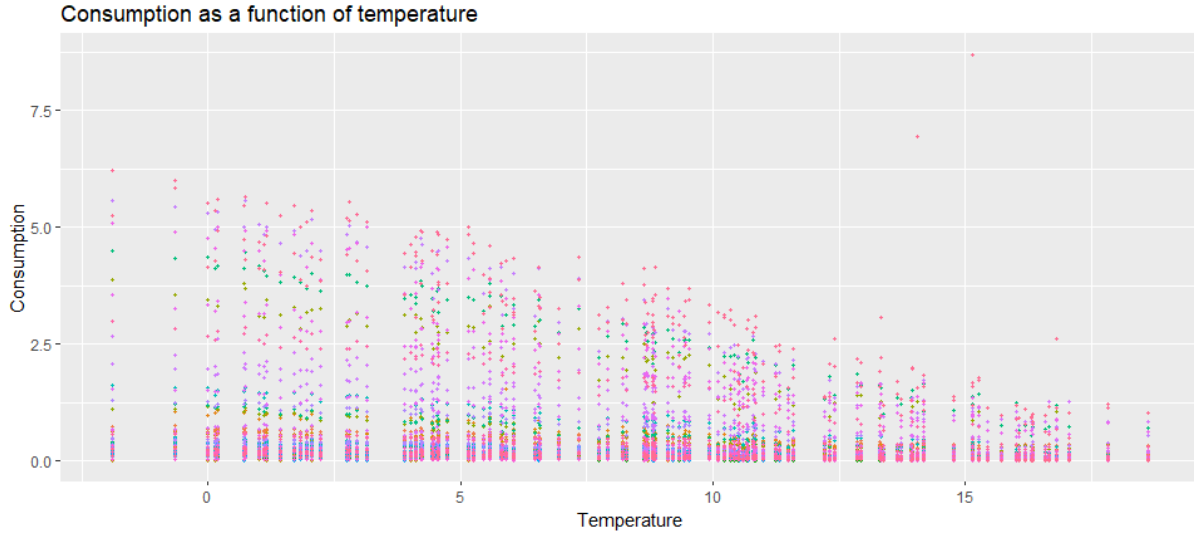


Figure 1: Scatterplot of the consumption against the temperature difference, colored by ID.

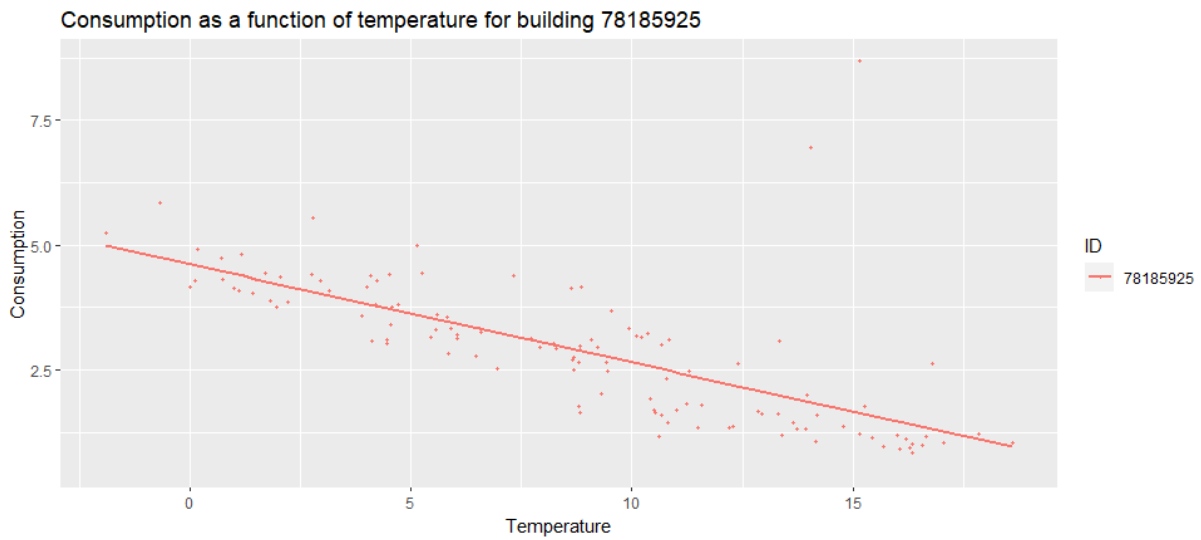


Figure 2: Scatterplot of building ID: 78185925, showing the lineary tendency assumed by equation 1..

To investigate the data several boxplots of the data is created where the heat consumption for each building is plotted (Figure 3). From the boxplot a difference between the buildings is noticed. However, nothing final can be concluded from his plot itself, it simply shows how the datapoints for each building is located.

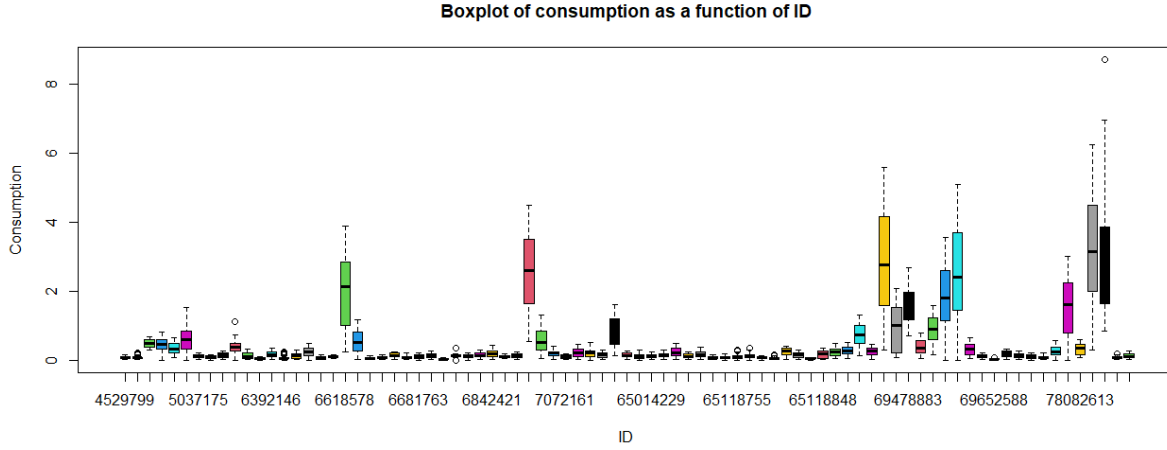


Figure 3: Boxplot of the heat consumption as a function of ID, showing the consumption data for each building .

From the data it is known that the buildings are of different sizes, which could explain the big difference in heat consumption shown in Figure 1 and 3. In attempt to decrease the variance of residuals the consumption is normalized so the building can be compared (Table 2, Figure 4). If the buildings aren't normalized would one recommend the biggest buildings for retrofitting.

Additional variable		
Ncons	Normalized heat consumption as difference between daily readings	Continuous value

Table 2: Table of the additional variables made from already existing variables from table 1.

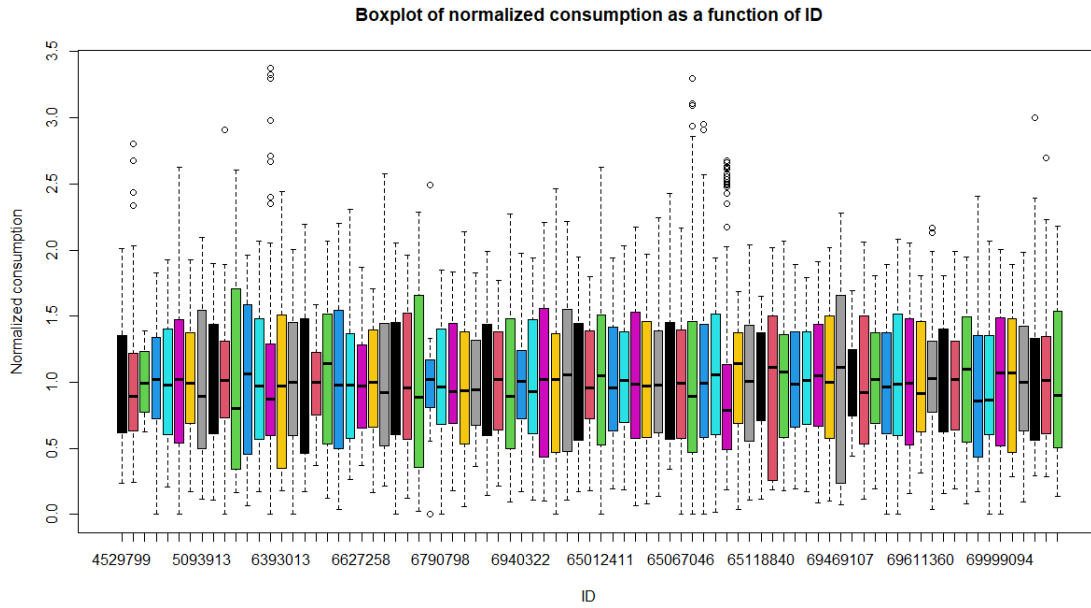


Figure 4: Boxplot of the heat normalized consumption as a function of ID, showing the normalized consumption data for each building .

The normalization is done by dividing the consumption for the corresponding building at each specific date with the mean consumption of each building for all the dates. When comparing the boxplot from the non-normalized boxplot and the normalized, the scale seems much more appropriate supporting a more relevant and valid comparison between each building (Figure 3, 4). To further investigate this new variable is the temperature difference plotted as a function of the normalized heat consumption (table 2).

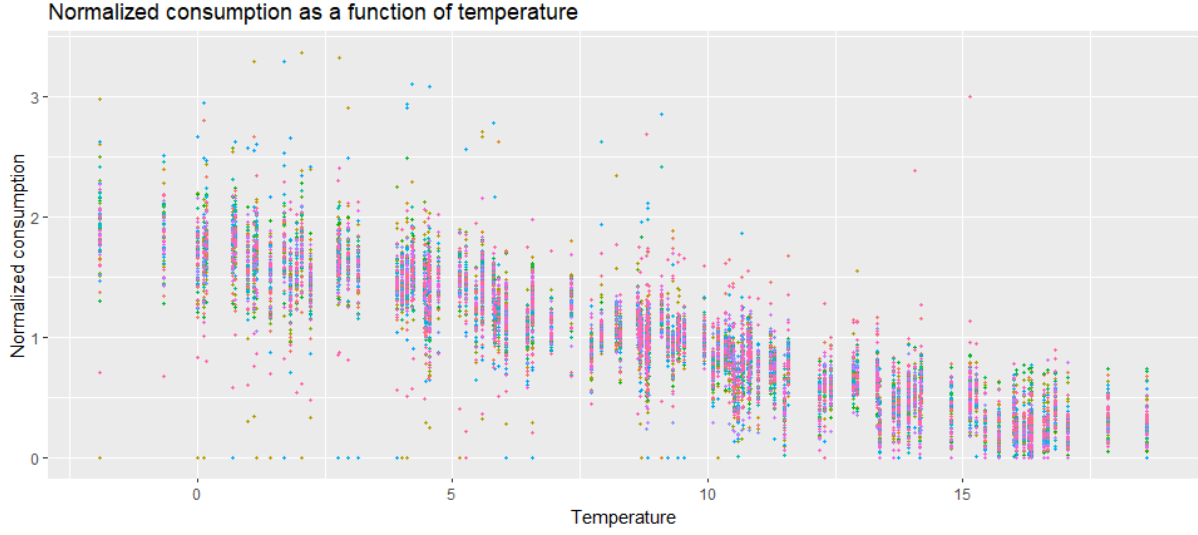


Figure 5: Scatterplot of the normalized consumption against date, colored by each building.

Compared to figure 1 is the variance of each building now smaller. A more clear linear tendency is seen, which also indicates that the point distribution in figure 1 has been affected by the size of the buildings and by normalizing the data is the buildings comparable.

It is additionally examined if any of the climatic variables correlate, and therefore should be examined in the model selection. When looking at figure 6 is a clear correlation and linear tendency seen. This indicates that the dew point depends on the temperature, which also corresponds to reality.

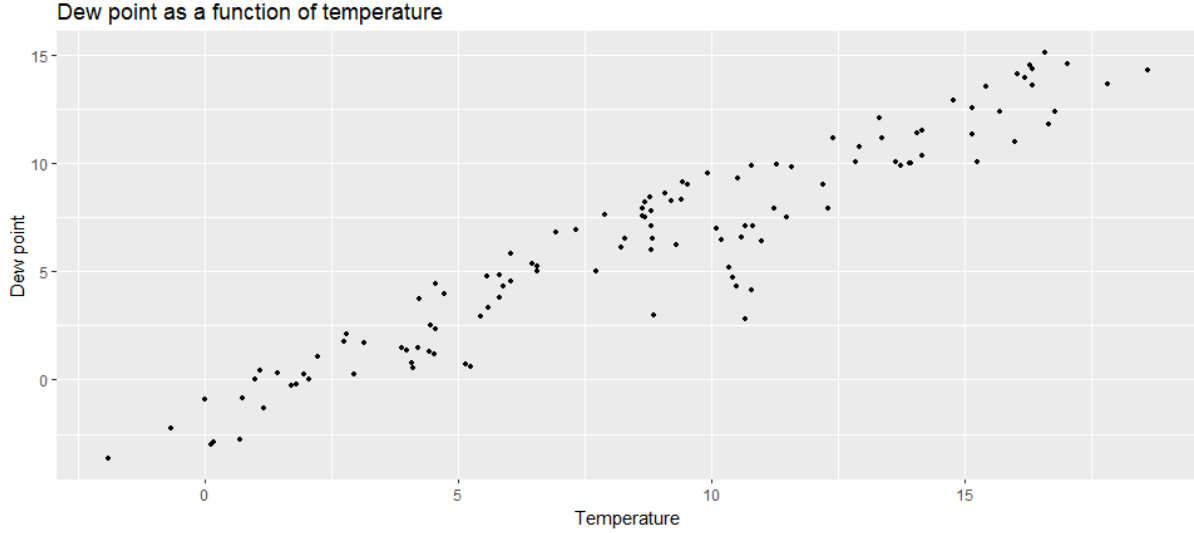


Figure 6: Date against normalized consumption and normalized mean temperature difference.

It is further investigated if the graphs for normalized mean temperature difference and the normalized heat consumption follow one another, when plotted against date. The normalized temperature has been calculated in the same way as the heat consumption.

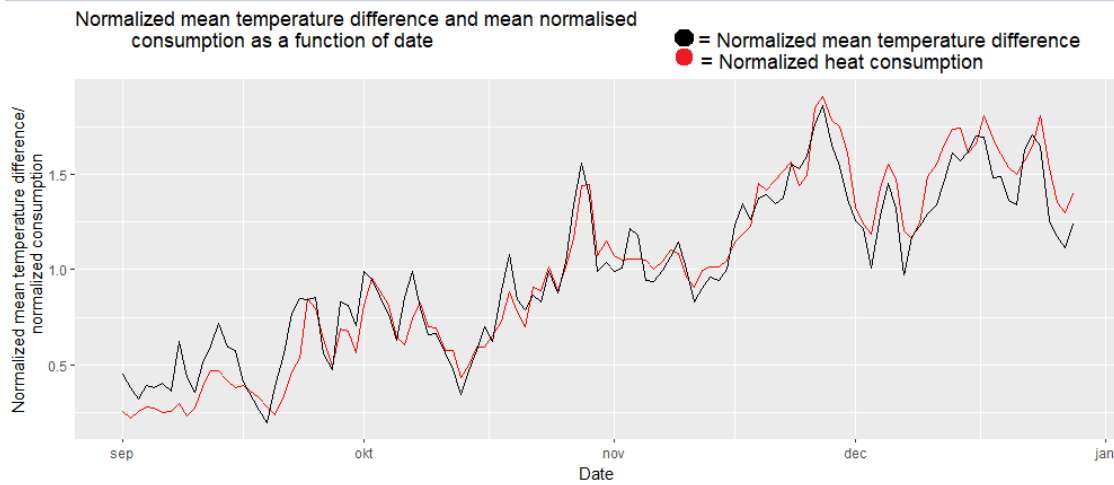


Figure 7: Date against normalized consumption and normalized mean temperature difference.

A correlation between the normalized heat consumption and normalized temperature is seen (figure 7). This indicates that the normalized mean temperature difference follows the heat consumption, and that the normalized heat consumption interacts with the mean normalized temperature difference.

3 Data cleansing

The data analysed in this report, come from two different data sources. These data sources have been merged and cleaned, before the statistical and descriptive analysis has been performed.

The first data source is from WUunderground, that provides almost hourly climate data. This data has been

cleaned where all columns with pure NAs or fixed values has been excluded, this includes the columns "Thunder" and "Wind chill". For each day is the mean value for the continuous variables and the mode of the factor variables are calculated.

The second data source provides daily readings for the district heating meters in 97 buildings in Høje Taastrup municipality. From this data three columns are kept, "ID", "Time" and "Reading". To avoid gaps in the data set buildings with less than 121 readings been excluded. Not all the readings were taken at the same time points for all the days, to correct for this has the daily readings been used to interpolate a reading at 11.59pm for each day. After the data cleansing and merging of the two data frames we have 9794 rows and 83 buildings remaining in the data set. In table 3 below is a summary of the joined data tables seen.

In the next session a given data frame has been used to ensure that the reports can be compared. Opposite to the generated data frame the variable snow have been removed in the given data table. Apart from this, there are only small differences in the values between the two data frames. A difference is for example seen in condition, where there is 3071 observations of scattered clouds in the cleansed data frame and 3066 in the given data frame.

date	dir	cond	fog	rain	snow
Min. :2018-09-01	SE :1079	Scattered Clouds: 3071	0:9462	0:9462	0:9794
1st Qu.:2018-10-01	South : 996	Mist :2407	1: 332	1: 332	
Median :2018-10-30	SW : 996	Clear : 1743			
Mean :2018-10-30	West : 913	Mostly Cloudy :913			
3rd Qu.:2018-11-29	East : 830	Fog : 664			
Max. :2018-12-28	ESE : 830	Partly Cloudy : 249			
	(Other):4150	(Other) : 747			

temp	dew_pt	hum	wind_spd	vis	pressure
Min. :-1.800	Min. :-3.600	Min. :49.00	Min. : 3.713	Min. : 1.965	Min. : 985.8
1st QU :4.579	1st Qu.: 2.190	1st Qu.:73.10	1st Qu.:11.305	1st Qu.:11.667	1st Qu.:1011.1
Median : 8.884	Median : 6.792	Median :82.30	Median :15.102	Median :17.645	Median :1017.6
Mean : 8.731	Mean : 6.315	Mean :81.01	Mean :16.356	Mean :20.601	Mean :1016.5
3rd Qu.:12.833	3rd Qu.: 9.947	3rd Qu.:89.30	3rd Qu.:20.786	3rd Qu.:29.571	3rd Qu.:1022.4
Max. :18.500	Max. :15.583	Max. :98.39	Max. :41.929	Max. :50.000	Max. :1040.2

id	cons
Min. : 4529799	Min. :0.00000
1st Qu.: 6627217	1st Qu.:0.07673
Median :65005112	Median :0.15164
Mean :37890916	Mean :0.43622
3rd Qu.:69429582	3rd Qu.:0.33675
Max. :78673711	Max. :8.00929

Table 3: Summary of joined data table.

4 Statistical Analysis

To perform a statistical analysis a modelselection was performed. Based on AIC, Anova and Ancova results the minimum model was found along with an investigation of the models residuals. The maximal model consist of the normalized consumption as function of ID, temperature difference [tempdif], wind speed [wind_spd], humidity [hum] and weekend/workingdays [weekend] with first order interactions.

From the data provided new variables are created. One of the variables created is temperature difference as this variable is in equation 1 (Table 4). This is simply made by subtracting the outside temperature provided in the data from the 21 degrees Celsius provided as the inside temperature.

Additional variables		
Tempdif	Temperature difference between inside- and outside temperature	Continuous variable
Weekend	Representation of date by differentiation between working/weekend days	2-level factor

Table 4: Table of the additional variables made from already existing variables.

The variable weekend are created as a representations of the variable date. The new variables are listed in table 4 and is present in the maximum model. The reason for adding the weekend/workingday variable as a

factor is based on observations in the data (Figure 8), showing a tendency which probably is not explained by randomness, hence a target for investigation.

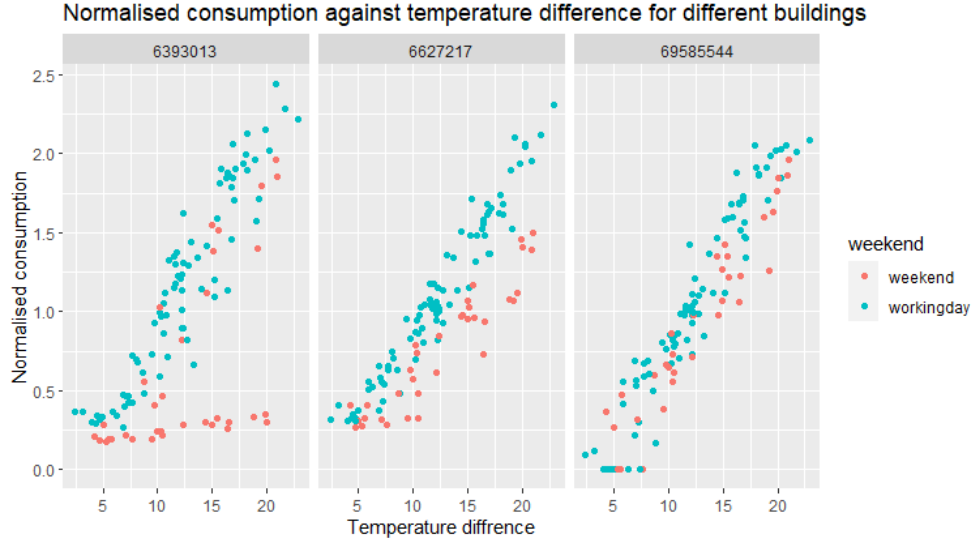


Figure 8: Plot of data for three different buldings, showing thee difference in slope depending on the factor weekend or workingsdays.

4.1 Model assumption and selection

The maximum model was reduced by performing forward and backwards selection simultaneously by an automatic command in R called step using F-statistics. As a result, only significant variables should appear in the maximum models. Firstly the model assumptions are check for the reduced maximum model (Figure 9). When checking for normality in the residuals the plots appears okay. In the QQ plot the start and end of the data appears respectively below and above the lineary tendency. However, the model is assumed to fuffill the model assumptions enough for further investigation and use.

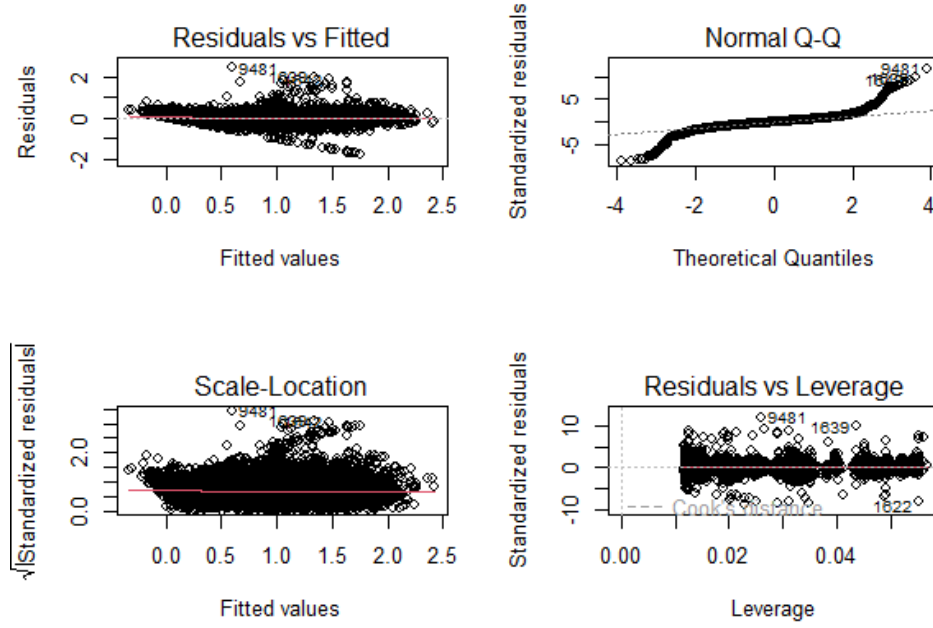


Figure 9: Plot of the regression made from the maximum model after being reduced by forward-backward selection.

As the sample size of our data is very high, there will be a high statistical power possibly making variables and interactions significant which in practice might not be the case. As a result, it is crucial that an investigation of the relevance of each variable is made before making the maximum model. When performing the backwards-forwards selection on the maximum model a model reduced to only significant variables and interactions is made (Table 5).

	Sum Sq	Df	F value	Pr(>F)
ID	0.01	82	0.00	1.0000
tempdif	1859.37	1	44274.02	<2.2e-16
wind_spd	11.99	1	285.41	<2.2e-16
weekend	40.99	1	976.10	<2.2e-16
hum	7.33	1	174.49	<2.2e-16
ID:tempdif	101.78	82	29.56	<2.2e-16
tempdif:weekend	4.09	1	97.28	<2.2e-16
tempdif:hum	4.17	1	99.35	<2.2e-16
tempdif:wind_spd	2.42	1	57.62	3.5e-14
wind_spd:hum	2.50	1	59.60	1.3e-14
wind_spd:weekend	1.46	1	34.66	4.1e-09
ID:weekend	32.73	82	9.50	<2.2e-16
Residuals	400.57	9538		

Table 5: ANCOVA made on the reduced maximum model after forward-backwards selection.

As suspected, many variables and interactions appear as significant. ID is not significant, however this is kept in the model as the AIC increases very much when removing it (Table 6). Hence, a better fit of the model on the data is prioritized.

	df	AIC
Reduced Max_model	258.00	-3134.47
Reduced Max_model % ID	176.00	-1583.74

Table 6: AIC of the reduced maximum model with and without the variable ID.

From the analysis of covariance (ANCOVA) (Table 5), some interactions does not necesarry have a relevance in practice. As a result, the interaction `wind_spd:weekend` is removed. An ANCOVA is performed on the new model (Table 7).

	Sum Sq	Df	F value	Pr(>F)
ID	0.01	82	0.00	1.0000
tempdif	1898.18	1	45039.04	<2.2e-16
wind_spd	11.99	1	284.40	<2.2e-16
weekend	40.99	1	972.67	<2.2e-16
hum	7.08	1	167.92	<2.2e-16
ID:tempdif	101.75	82	29.44	<2.2e-16
tempdif:weekend	3.38	1	80.26	<2.2e-16
tempdif:hum	4.21	1	99.93	<2.2e-16
tempdif:wind_spd	2.34	1	55.43	1.1e-13
wind_spd:hum	1.76	1	41.81	1.1e-10
ID:weekend	32.73	82	9.47	<2.2e-16
Residuals	402.02	9539		

Table 7: ANCOVA made on the reduced model after removing irrelevant interactions.

All variables and interactions left are significant as suspected. Hence, are they kept in the model. From the residual plot some outliers was present (Figure 9) - outlier 9481 appeared in all residual plots, hence is this removed with the aim of increasing the models fit, as outliers could carry leverage. However, uutlier 9481 might not have a big influence when looking at the leverage plot (figure 9). The final model is then as shown below (equation 2) with a multiple R-sqaured value at 0.8519 and independently and identically distributed residuals;

$$\begin{aligned}
Y_{ghi} = & \mu_0 + \mu_g + \mu_h + \mu_{h,g} + \beta_1 \cdot tempdif_i + \beta_2 \cdot wind_spd_i + \beta_3 \cdot hum_i \beta_g \cdot tempdif_i + \beta_h \cdot tempdif_i \\
& + \beta_4 \cdot tempdif_i \cdot hum_i + \beta_5 \cdot tempdif_i \cdot wind_spd_i + \beta_6 \cdot wind_spd_i \cdot hum_i + \varepsilon_{gi} \quad \varepsilon \sim N(0, \sigma^2) \quad \varepsilon \text{ is i.i.d} \quad (2)
\end{aligned}$$

where

$$g = ID \forall ID \text{ in dataframe}$$

$$h = workingday_weekend \forall workingday_weekend \text{ in dataframe}$$

5 Results

The final model is found and defined as written in equation 2. From the final model the tendency and results can now be presented.

5.1 Interactions

The interaction between temperature difference and weekend or working day is significant (table 7). To analyse this interaction a figure created where the normalized consumption is plotted as a function of the temperature difference split into weekend and working day (figure 10).

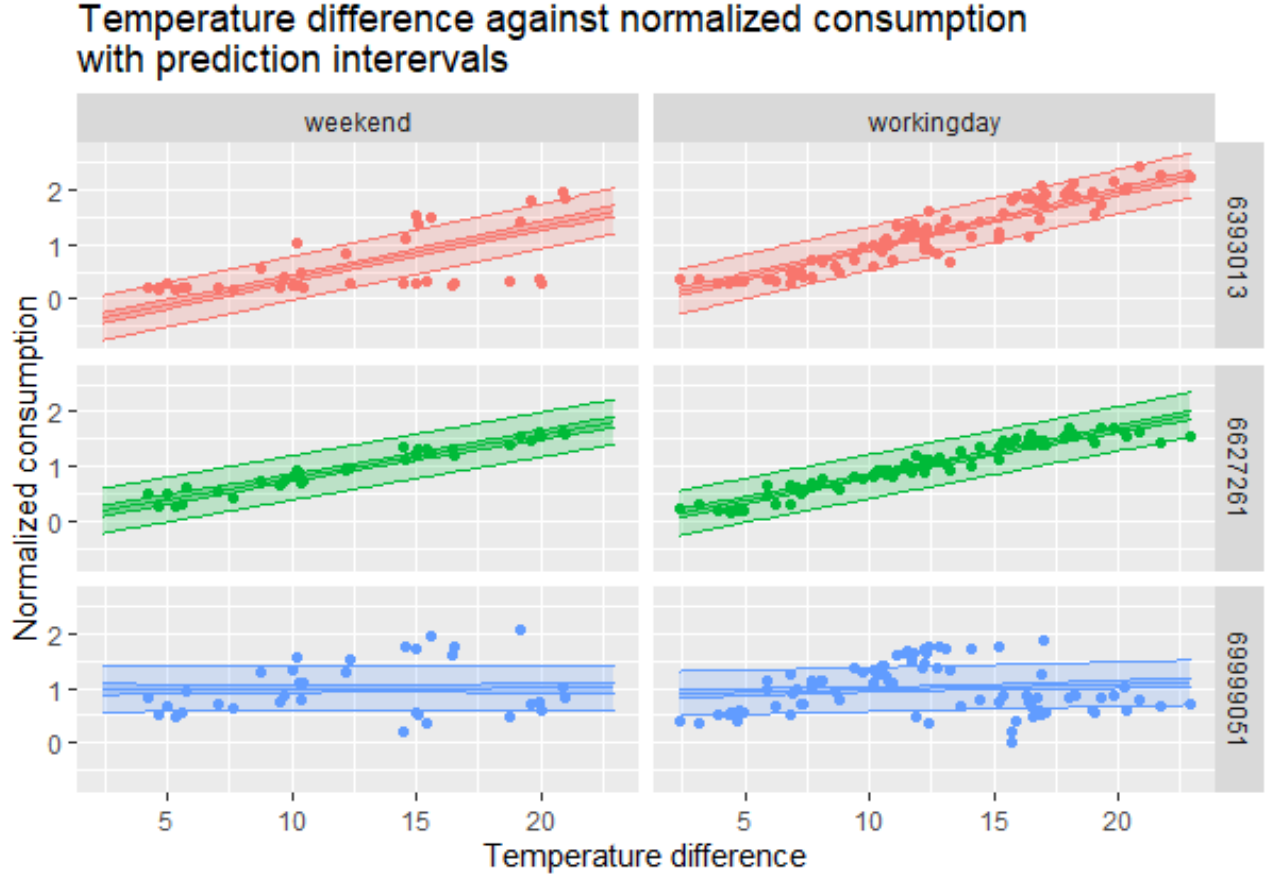


Figure 10: Temperature difference against normalized consumption with prediction and confidence intervals split into weekend and working day

Even though there isn't seen a big difference in normalized consumption for these three buildings, are the heat consumption very different for some buildings (figure 8). There is a difference in the intercept and slope depending on if it is the weekend, or a workingday. In the model this change in intercept can be observed as there is a first order interaction between ID and it being weekend or not. In addition, the change in slope can be observed from the first order interaction between temperature difference and it being weekend or not. Here the slope coefficient for it being a weekday was found to be 0.08 with a confidence interval at $[0.070:0.084]$, meaning that, based on out mode, weekdays have a higher consumption than weekends for an increase in temperature difference.

The prediction and confidence interval fits okay for both the weekend and the workingdays. There are some outliers but as the R-squared value is 0.8519 could this explain some of the outliers being outside the prediction interval.

5.2 Design/regression matrix and ranking of buildings

A design matrix is made. The design matrix carries information of the different parameters/variables from the model. Based on the final model a design matrix is made with the aim of investigation of the slopes[1]. Continuous variables, humidity and wind_spd known from interactions with temperature difference, are used as a constant. The constant is found as the mean of each interaction. As this information is now logged in the matrix an estimate and corresponding standard error is found for each ID. From these estimates of slopes the buildings can be ranked by the slope. From equation 1 is the slope an interpretation of the isolation in the buildings. The confidence interval is noted as well as the 0.975 and 0.025 quantile of the estimated slope (table 8, 9).

	ID	Slope	Sd. Error	$Q_{0.975}$	$Q_{0.025}$	Type	Sum consumption [MWh]
1	6681894	0.131	0.003833	0.138	0.123	Fritidsaktiviteter	4.463
2	69469107	0.130	0.003833	0.138	0.123	Driftsbygninger	108.190
3	5325295	0.122	0.003833	0.130	0.115	Driftsbygninger	14.417
4	6618580	0.118	0.003833	0.125	0.110	Biblioteker	62.435
5	65118812	0.113	0.003833	0.121	0.106	Institutioner for ældre	7.586

Table 8: Table of the 10 worst isolated buildings, with nr. 1 being the worst isolated.

	ID	Slope	Sd. Error	$Q_{0.975}$	$Q_{0.025}$	Type	Sum consumption [Mwh]
1	69999051	0.001	0.003845	0.009	-0.006	Skolefritidsordninger	33.187
2	6790785	0.033	0.003830	0.040	0.025	Integrerede daginstitutioner	16.643
3	4839509	0.039	0.003833	0.047	0.031	Skoler	58.068
4	6567326	0.045	0.003833	0.052	0.037	Integrerede daginstitutioner	12.907
5	69478883	0.050	0.003833	0.058	0.043	Idræts- og svømmehaller	186.475

Table 9: Table of the 10 best isolated buildings, where 1 is the best isolated building.

Based on the slopes recommendations on the prioritizing of retrofitting can be made. Buildings with the largest slope coefficient should be prioritized for renovation, as they are the worst isolated, hence more heat is needed to outbalance the outside climate. From the sum consumption for each building the consumption of each building is noted. It could be argued that building ID: 69469107 should be renovated before building ID: 6681894 despite the lower slope coefficient as building ID: 69469107 has a much higher consumption and a almost identical slope. When visualizing the 3 best and worst isolated buildings the slopes relation to isolation can be seen (Figure 11).

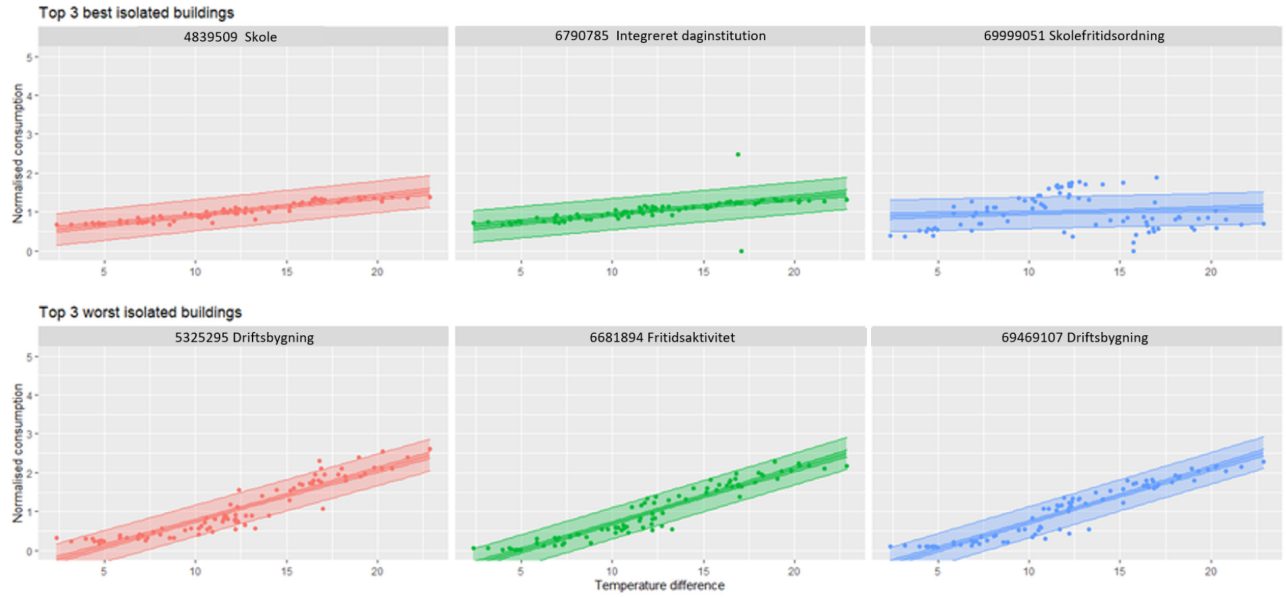


Figure 11: Plot of the three best and worst isolated buildings. Raw data is plotted along with the confidence- and prediction interval made from the model. Worst and best isolated buildings are shown to the left, second best and worst in the middle and 3rd best and worst to the right.

The slope clearly has a more flat tendency when looking at the best isolated. When comparing to the worst isolated buildings a much more increased slope can be seen. The models prediction and confidence interval is plotted along with the raw data. Generally, the models fits the data very well, however some buildings are not as good described by the model as others. The best isolated building ID: 69999051 has much more scatter points beyond the prediction interval of the model compared to the 2nd and 3rd best isolated, which points fits more nicely with the model (Figure 11). A multiple R-squared value at 0.8519 could explain some of this difference, as the model is a generalization. Furthermore, from the calculated slopes, we can observe a big difference in insulation across buildings (Figure 12).

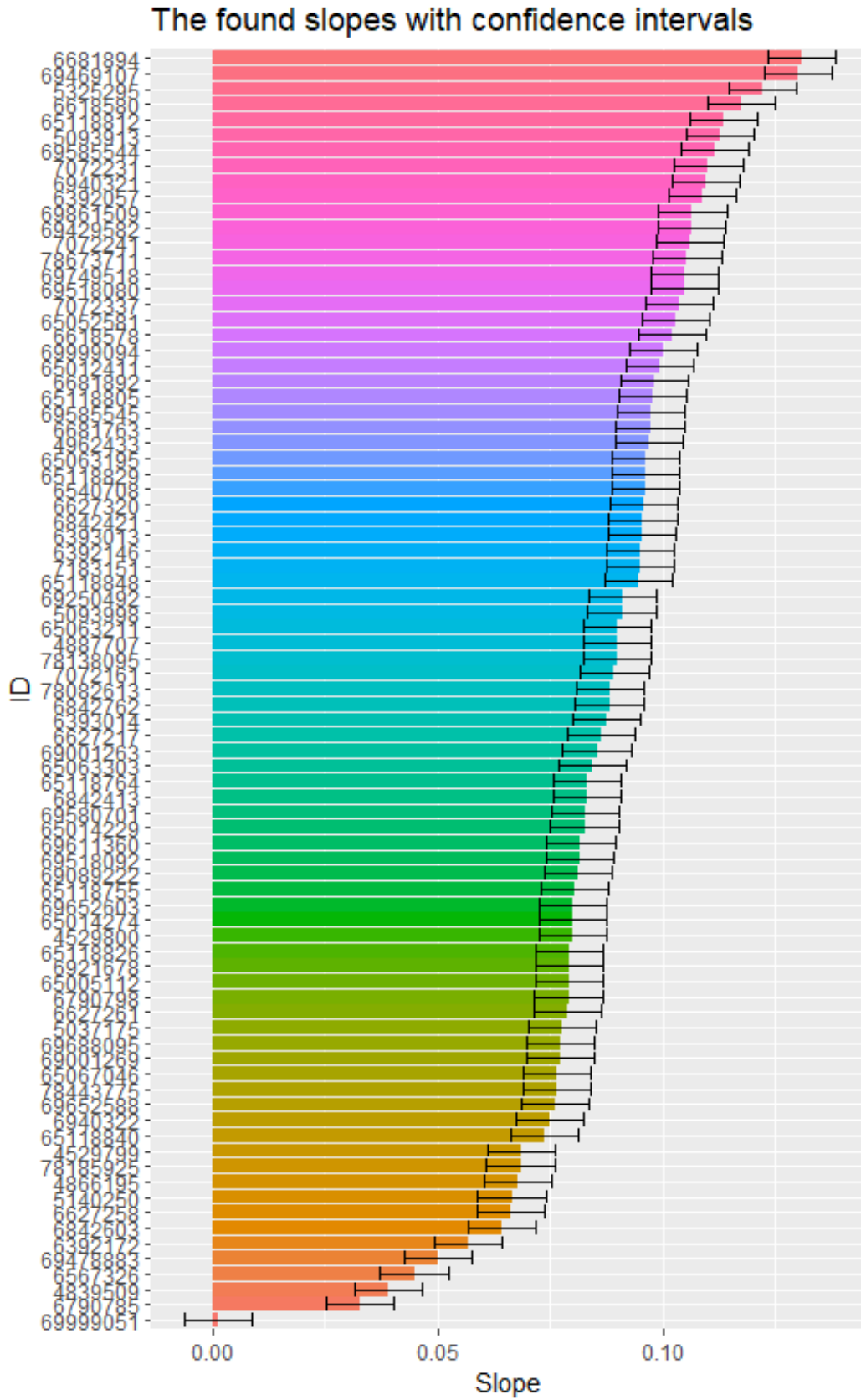


Figure 12: Plot of all estimated slopes for each ID and their confidence intervals ($\alpha=0.05$).

To sum up the results, there is a difference in the heat consumption depending on if it is a weekend or a workingday. The buildings shown in table 8 would be the ones recommend for retrofitting, within the list of those five buildings, it could make sense to take into account the sum of consumption for each building, when prioritizing between those five.

6 Conclusion

A significant difference in heat consumption among various buildings analyzed has been revealed. An ordered list of all the buildings, ranked according to their insulation levels, was presented. In particular, building 69469107, which is both large and one of the top 3 worst insulated buildings, emerges as an ideal candidate for improvement. The generated model generally fits the data well, but some buildings are not as accurately described. The multiple R-squared value of 0.8519 supports this observation.

Furthermore, a distinct difference in energy consumption between working days and weekends was uncovered by the study. This suggests that some buildings turn off the heat during the weekends while others do not. This highlights the importance of energy conservation efforts, especially during non-working days. Therefore, it is recommended that more buildings in the municipality, if feasible, should turn down the heat during weekends in order to conserve energy and reduce consumption.

References

- [1] L. E. Christiansen, “Vital capacity-with addon vital capacity,” 2019.

Appendix

1. List of all buildings ranked by isolation ability
2. Code for cleaning data
3. Code for model selection
4. Code for creating plots
5. Code for working with found model

1. List of all buildings ranked by isolation ability

	ID	Slope	Sd. Error	$Q_{0.975}$	$Q_{0.025}$	Type	Sum consumption [MWh]
1	6681894	0.131	0.003833	0.138	0.123	Fritidsaktiviteter	4.463
2	69469107	0.130	0.003833	0.138	0.123	Driftsbygninger	108.190
3	5325295	0.122	0.003833	0.130	0.115	Driftsbygninger	14.417
4	6618580	0.118	0.003833	0.125	0.110	Biblioteker	62.435
5	65118812	0.113	0.003833	0.121	0.106	Institutioner for ældre	7.586
6	5093913	0.113	0.003833	0.120	0.105	Integrerede daginstitution PRIVATE	9.339
7	69585544	0.112	0.003833	0.119	0.104	Skoler	287.748
8	7072231	0.110	0.003833	0.118	0.103	Andre ejendomme	24.880
9	6940321	0.110	0.003833	0.117	0.102	Fritids- og ungdomsklubber	67.950
10	6392057	0.109	0.003833	0.116	0.101	Aktivitets- og samværstilbud	6.037
11	69861509	0.107	0.003833	0.114	0.099	Integrerede daginstitutioner	10.301
12	69429582	0.106	0.003833	0.114	0.099	Skoler	326.486
13	7072241	0.106	0.003824	0.114	0.099	Integrerede daginstitutioner	24.494
14	78673711	0.105	0.003833	0.113	0.098		15.190
15	69749518	0.105	0.003833	0.112	0.097	Fritidsaktiviteter	12.678
16	69518080	0.105	0.003833	0.112	0.097	Idræts- og svømmehaller	44.833
17	7072337	0.104	0.003833	0.111	0.096	Integrerede daginstitutioner	16.536
18	65052581	0.103	0.003833	0.110	0.095	Fritids- og ungdomsklubber	26.672
19	6618578	0.102	0.003833	0.110	0.095		221.613
20	69999094	0.100	0.003824	0.108	0.093	Andre kulturelle opgaver	177.310
21	65012411	0.099	0.003816	0.107	0.092	Integrerede daginstitutioner	13.209
22	6681892	0.098	0.003833	0.106	0.091	Tomme ejendomme	15.589
23	65118805	0.098	0.003833	0.105	0.090	Dagleje	8.349
24	69585545	0.097	0.003833	0.105	0.090	Andre kulturelle opgaver	37.672
25	6681763	0.097	0.003838	0.105	0.090	Integrerede daginstitution PRIVATE	12.206
26	4962433	0.097	0.003836	0.105	0.090	Træningscentre	68.705
27	65063195	0.096	0.003833	0.104	0.089	Foreb. foranst for børn og unge	14.243
28	65118829	0.096	0.003833	0.104	0.089	Længerevarende botilbud	17.745
29	6540708	0.096	0.003833	0.104	0.089		8.549
30	6627320	0.096	0.003833	0.103	0.088		9.497
31	6842421	0.096	0.003833	0.103	0.088	Skolefritidsordninger	23.406
32	6393013	0.095	0.003833	0.103	0.088	Integrerede daginstitutioner	14.619
33	6392146	0.095	0.003833	0.103	0.088	Integrerede daginstitutioner	20.099
34	7183151	0.095	0.003833	0.103	0.088	Driftsbygninger	97.928
35	65118848	0.095	0.003833	0.102	0.087	Fritids- og ungdomsklubber	20.362
36	69250492	0.091	0.003833	0.099	0.084	Andre kulturelle opgaver	29.265
37	5093998	0.091	0.003833	0.098	0.083	Længerevarende botilbud	17.344
38	65063211	0.090	0.003833	0.097	0.082	Integrerede daginstitutioner	19.995
39	4887707	0.090	0.003833	0.097	0.082	Længerevarende botilbud	40.500
40	78138095	0.090	0.003833	0.097	0.082	Andre ejendomme	371.297
41	7072161	0.089	0.003833	0.097	0.082	Sundhedstjeneste	11.830
42	78082613	0.088	0.003833	0.096	0.081	Stadion og idrætsanlæg	37.608
43	6842762	0.088	0.003833	0.096	0.081	Tomme ejendomme	14.469
44	6393014	0.088	0.003824	0.095	0.080	Fritids- og ungdomsklubber	28.114
45	6627217	0.086	0.003833	0.094	0.079		7.197
46	69001263	0.085	0.003833	0.093	0.078	Integrerede daginstitutioner	27.381
47	65063303	0.084	0.003833	0.092	0.077	Integrerede daginstitutioner	8.333
48	65118764	0.083	0.003801	0.091	0.076	Integrerede daginstitution PRIVATE	13.826
49	6842413	0.083	0.003833	0.091	0.076	Integrerede daginstitutioner	18.532
50	69580701	0.083	0.003824	0.090	0.075	Plejecentre	221.705

	ID	Slope	Sd. Error	$Q_{0.975}$	$Q_{0.025}$	Type	Sum consumption [MWh]
51	65014229	0.083	0.003833	0.090	0.075	Hjælpemiddeldepoter	14.191
52	69611360	0.082	0.003833	0.089	0.074	Fritids- og ungdomsklubber	13.700
53	69518092	0.082	0.003833	0.089	0.074	Pleje og omsorg [..]	104.273
54	69089222	0.081	0.003833	0.089	0.074	Længerevarende botilbud	86.512
55	65118755	0.080	0.003802	0.088	0.073	Beboelsesejendomme	10.317
56	69652603	0.080	0.003833	0.088	0.073	Integrerede daginstitutioner	22.061
57	65014274	0.080	0.003833	0.088	0.073	Integrerede daginstitutioner	16.866
58	4529800	0.080	0.003833	0.087	0.072	Integrerede daginstitutioner	8.684
59	65118826	0.079	0.003833	0.087	0.072	Beboelsesejendomme	29.238
60	6921678	0.079	0.003833	0.087	0.072	Plejecentre	299.204
61	65005112	0.079	0.003833	0.087	0.072	Integrerede daginstitutioner	17.984
62	6790798	0.079	0.003833	0.087	0.072	Andre kulturelle opgaver	12.857
63	6627261	0.079	0.003833	0.086	0.071	Længerevarende botilbud	17.230
64	5037175	0.078	0.003833	0.085	0.070	Længerevarende botilbud	13.450
65	69688095	0.077	0.003833	0.085	0.070	Integrerede daginstitutioner	16.287
66	69001269	0.077	0.003833	0.085	0.070	Tomme ejendomme	32.132
67	65067046	0.077	0.003824	0.084	0.069	Andre ejendomme	9.597
68	78443775	0.076	0.003833	0.084	0.069	Integrerede daginstitutioner	8.637
69	69652588	0.076	0.003833	0.084	0.069		3.599
70	6940322	0.075	0.003833	0.083	0.067	Integrerede daginstitutioner	23.576
71	65118840	0.074	0.003833	0.081	0.066	Andre ejendomme	6.263
72	4529799	0.069	0.003833	0.076	0.061	Integrerede daginstitutioner	8.828
73	78185925	0.068	0.003858	0.076	0.061	Idræts- og svømmehaller	333.546
74	4866195	0.068	0.003824	0.075	0.061	Længerevarende botilbud	52.014
75	5140250	0.066	0.003827	0.074	0.059	Længerevarende botilbud	45.145
76	6627258	0.066	0.003833	0.074	0.059	Integrerede daginstitutioner	9.862
77	6842603	0.064	0.003833	0.072	0.057	Integrerede daginstitutioner	12.917
78	6392172	0.057	0.003815	0.064	0.049	Integrerede daginstitutioner	8.148
79	69478883	0.050	0.003833	0.058	0.043	Idræts- og svømmehaller	186.475
80	6567326	0.045	0.003833	0.052	0.037	Integrerede daginstitutioner	12.907
81	4839509	0.039	0.003833	0.047	0.031	Skoler	58.068
82	6790785	0.033	0.003830	0.040	0.025	Integrerede daginstitutioner	16.643
83	69999051	0.001	0.003845	0.009	-0.006	Skolefritidsordninger	33.187

Table 10: Slope for all the buildings, worst insulated buildings first

2. Code for cleaning data

```
#### Introduction ####
rm(list=ls())

library(tidyverse)
library(ggplot2)
library(car)
library(stringr)
library(xtable)

load("./WUndergroundHourly.RData")

# Remove all columns without data (only NA columns)
clima <- select(WG,!c("wind_gust" , "wind_chill",
                    "heat_index" , "precip",
                    "precip_rate", "precip_total"))

# Remove columns with fixed values
clima <- select(clima, !c("hail","thunder","tornado"))

# making separate column for date and time
clima[c('date','time')] <- str_split_fixed(clima$date,' ',2)

# Picking the correct datatype for the columns
clima <- mutate(clima,
  across(c(dir, cond, fog, rain, snow, date, time),factor)
)

### Finding mode and mean for each date ###
clima_by_date <- group_by(clima, date)

# Define function to calculate mode
mode <- function(factors){
  factors <- factors[factors != ""]
  max <- factors %>%
    table() %>%
    which.max() %>%
    as.data.frame() %>%
    rownames() %>%
    factor()

# If it does not have a value set it to None for clarity
  if (max == ""){
```

```

    return(factor("None"))}
else {
  return(max)
}
}

# Assign the mode and mean using aggregate functions
clima_mean_mode <- clima_by_date %>%
  summarise(
    across(c(dir, cond, fog, rain, snow),mode),
    across(c(temp, dew_pt, hum, wind_spd, vis, pressure), ~mean(.,na.rm=T))
  )

#### Read in the energy performance of the building ####
# Find all the files in ./data
data_files <- dir("./data", full.names=T)

# Read them into a single dataframe
# \x00 is set to be ignored since it for some reason is at the end of each file
energy <- NULL
for (i in seq_along(data_files)){
  data <- read.table(data_files[i], sep=";", skipNul=TRUE) %>%
    select(V1, V2, V4)
  energy <- bind_rows(energy, data)
}

# Renaming the dataframe
energy <- energy %>%
  rename(id=V1, time=V2, reading=V4)

## Exclude meters with less than 121 records
# find the records with 121 records
id_to_keep <- group_by(energy, id) %>%
  summarise(n=n()) %>%
  filter(n==121)
# only keep these
energy <- filter(energy, id %in% id_to_keep$id)

# set the correct datatypes
energy <- mutate(energy,
  reading = as.numeric(gsub(",", ".", reading)))
# Date CET/CEST refers to winter-/summer-time
energy$time <- as.POSIXct(strptime(energy$time,"%d-%m-%Y %H.%M"))

```



```

# Make data to approximate new values at 11:59:00
days <- unique(as.Date(energy$time))
time <- "11:59:00"
new_time_date <- NULL
for(i in seq_along(days)){
  days_time <- paste(days[i],time) %>%
    as.POSIXct()
  new_time_date <- append(new_time_date, days_time)
}

new_time_date

## Approximate new values at 11:59:00
id_time_cons <- NULL
all_id <- unique(energy$id)

for (i in seq_along(all_id)){
  # select values with the correct id
  energy_for_id <- filter(energy, id == all_id[i]) %>%
    arrange(time)

  # Approximate new values
  # It returns NA if trying to approximate
  # 2018-12-29 11:59:00 CET, its not seen
  # Here the largest number is used
  approx <- approx(energy_for_id$time,
                   energy_for_id$reading, xout=new_time_date,
                   rule = 2)

  # Assign the new values to a temp df
  time <- as.Date(approx$x)
  reading <- approx$y
  id <- rep(all_id[i],length(time))
  temp_df <- data.frame(time,id,reading)

  # add them to the id_time_cons df
  id_time_cons <- bind_rows(temp_df,id_time_cons)
}

# make consumption array.

```

```

# calculated by taking the day before MINUS the day
# this means that the last day gets NaN
consumption <- group_by(id_time_cons, id) %>%
  arrange(time) %>%
  mutate(cons=lead(reading)-reading)

# ungroup and rename time to date and remove reading
consumption <- ungroup(consumption, id)
consumption <- rename(consumption,date=time) %>% select(!"reading")

# Join the two datasets
clima_mean_mode <- mutate(clima_mean_mode, date=as.Date(date))
joined <- inner_join(clima_mean_mode, consumption, by="date")

#Remove first date due to NANS, and end due to estimation
joined <- filter(joined, date != "2018-08-31")
joined <- filter(joined, date != "2018-12-29")

# Rows
nrow(joined)
# Summary
summary(joined)
# Remaining meters == ids?
nrow(unique(select(joined, id)))
# 83 ids

```

3. Code for model selection

```
#### Loading and mutating/formatting the data ####
## Reading in the libraries
rm(list=ls())
library(tidyverse)
library(ggplot2)
library(car)
library(stringr)
library(xtable)
library(lubridate)
library(gridExtra)
library("xlsx")

## Defining get prediction interval function
get_pred_conf <- function(D, fit){
  ID <- levels(D$ID)
  weekend <- levels(D$weekend)
  n <- 100
  tempdif <- seq(min(D$tempdif), max(D$tempdif), length.out = n)

  wind_spd <- mean(D$wind_spd)
  hum <- mean(D$hum)
  dew_pt <- mean(D$dew_pt)
  pressure <- mean(D$pressure)

  new_data <- expand.grid("ID" = ID,
                        "tempdif" = tempdif,
                        "wind_spd" = wind_spd,
                        "hum" = hum,
                        "dew_pt" = dew_pt,
                        "pressure" = pressure,
                        "weekend" = weekend,
                        stringsAsFactors = T
  )

  pred <- predict(fit, interval = "prediction", newdata = new_data)
  pred <- as.data.frame(pred)
  pred <- cbind(new_data, pred)

  conf <- predict(fit, interval = "confidence", newdata = new_data)
  conf <- as.data.frame(conf)
  pred$conf_lwr <- conf$lwr
  pred$conf_upr <- conf$upr
```

```

    return (pred)
  }

## Reading in the data
D <- read.csv("merged_data.csv", header=TRUE)

## Setting the datatypes
# Setting the factors
D <- mutate(D, across(c(dir, cond, fog, rain, ID),factor))
# Setting the dates
D$date <- as.POSIXct(D$date, tz = "UTC")

## Mutating the data
# Adding temp difference column to the dataframe
D$tempdif <- 21 - D$temp

# Adding the weekends as a column
is_weekend <- function(date){
  number_day_df <- wday(date, label=T)
  number_day_char <- as.character(number_day_df)
  return(number_day_char)
}
D <- mutate(D, weekday=is_weekend(date)) %>%
  mutate(weekend = ifelse(weekday %in% c("lø", "sø", "Sat", "Sun"), "weekend", "workingday")) %>%
  mutate(weekend = factor(weekend))

# getting start/end of month int (mutating the date variable)
D <- mutate(D, dag=str_split_fixed(date, "-", 3)[ ,3])
D <- mutate(D, start_or_end = ifelse(as.integer(dag)<15, "START", "END")) %>%
  mutate(start_or_end = factor(start_or_end))

# find the normalised data
mean_each <- group_by(D, ID) %>%
  summarise(mean_each = mean(consumption))
D_with_mean <- inner_join(mean_each, D, "ID")
D <- mutate(D_with_mean, ncons = consumption/mean_each)

## Removing non important columns
D <- select(D, !c("temp", "mean_each", "dag", "weekday"))

### Maximum model

```

```

D_scope <- select(D, ID, ncons, tempdif, wind_spd, weekend, hum)
fit_scope <- lm(ncons~. ,D_scope)
#fit_old <- step(fit_scope, scope = ~.^2 , k=log(nrow(D_scope)), test = "F")
fit_old <- lm(formula = ncons ~ ID + tempdif + wind_spd + weekend + hum +
              ID:tempdif + tempdif:weekend + tempdif:hum + tempdif:wind_spd +
              wind_spd:hum + wind_spd:weekend + ID:weekend, data = D)

## Removing -wind_spd:weekend
fit <- update(fit_old, .~. -wind_spd:weekend)

AIC(fit,fit_old)
anova <- anova(fit,fit_old)

## Residual plot
#par(mfrow=c(2,2))
#plot(fit)

## Removing outliers
outliers <- c(9481)
D <- filter(D, !row_number() %in% outliers)

## Residual plot
fit <- update(fit_old, .~. -wind_spd:weekend)
#par(mfrow=c(2,2))
#plot(fit)

## Plot Weekend V nonWeekend
pred <- get_pred_conf(D, fit)

show <- c("6681894","69469107","5325295")
pred_id <- filter(pred, ID %in% show )
D_sub <- filter(D, ID %in% show)

#as.vector(sample(D$ID, size=1, replace=T))
#
ggplot(pred_id, aes(x=tempdif, y=fit, col=ID)) +
  geom_line(size=0.5) +
  labs(y="Normalized consumption", x="Temperature difference") +
  geom_ribbon(alpha=0.2, aes(x=tempdif, y=fit, col=ID, ymax =upr, ymin =lwr, fill=ID)) +
  geom_ribbon(alpha=0.4, aes(x=tempdif, y=fit, col=ID, ymax =conf_upr, ymin =conf_lwr, fill=ID)) +
  theme(legend.position = "none") +

```

```

    geom_point(data=D_sub, aes(x=tempdif, y=ncons, col=ID)) +
    facet_grid(ID~weekend) +
    ggtitle("Temperature difference against normalized consumption
with prediction interervals")

### TOP worst V best

pred <- get_pred_conf(D, fit)
D_wd <- filter(D, weekend == "workingday")
pred_wd <- filter(pred, weekend == "workingday")

# Top 3 best
show <- c("69999051", "6790785", "4839509")
pred_id <- filter(pred_wd, ID %in% show )
D_sub <- filter(D_wd, ID %in% show)

best <- ggplot(pred_id) +
  geom_line(size=0.4, aes(x=tempdif, y=fit, col =ID)) +
  labs(y="Normalised consumption", x="") +
  geom_ribbon(alpha=0.3, aes(x=tempdif, y=fit, col=ID, ymax =upr, ymin =lwr, fill=ID)) +
  geom_ribbon(alpha=0.4, aes(x=tempdif, y=fit, col=ID, ymax =conf_upr, ymin =conf_lwr, fill=ID)) +
  geom_point(data=D_sub, aes(x=tempdif, y=ncons, col=ID)) +
  facet_grid(~ID) +
  coord_cartesian(ylim = c(0, 5)) +
  ggtitle("Top 3 best isolated buildings") +
  theme(legend.position = "none")

# top 3 worst
show <- c("6681894", "69469107", "5325295")
pred_id <- filter(pred_wd, ID %in% show )
D_sub <- filter(D_wd, ID %in% show)

worst <- ggplot(pred_id) +
  geom_line(size=0.4, aes(x=tempdif, y=fit, col =ID)) +
  labs(y="Normalised consumption", x="Temperature difference") +
  geom_ribbon(alpha=0.3, aes(x=tempdif, y=fit, col=ID, ymax =upr, ymin =lwr, fill=ID)) +
  geom_ribbon(alpha=0.4, aes(x=tempdif, y=fit, col=ID, ymax =conf_upr, ymin =conf_lwr, fill=ID)) +
  geom_point(data=D_sub, aes(x=tempdif, y=ncons, col=ID)) +
  facet_grid(~ID) +
  coord_cartesian(ylim = c(0, 5)) +
  ggtitle("Top 3 worst isolated buildings") +
  theme(legend.position = "none")

```

```

grid.arrange(best,worst)

### Why pick weekend

show <- c("6681894","4529800","6393013")
D_subset <- filter(D, ID %in% show)
ggplot(D_subset) +
  geom_point(aes(x=tempdif, y=ncons, col=weekend)) +
  facet_grid(~ID) +
  labs(x="Temperature diffrence", y="Normalised consumption") +
  ggtitle("Normalised consumption against temperature difference for the 3 worst")

#4529800, 6393013

####matrix

#### Design matrix ####
### 262 columns in total
### range 90 - 171 is isolation

summary.fit <- summary(fit)$coefficients
summary.df <- as.data.frame(summary.fit)
fit_cor <- summary(fit,correlation = TRUE)

summary.df$row <- 1:nrow(summary.df)

isolation <- grep(pattern="ID\\d+:tempdif",x=rownames(summary.df))
(summary.df)[isolation,]

A <- cbind(0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,
           diag(82),
           0,0,0,0,0,0,0,0,0,0,
           0,0,0,0,0,0,0,0,0,0,

```

```

0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0)
A <- rbind(0, A)
A <- rbind(A,0,0,0)
A[,84] <- rep(1,length(A[,1]))
A[,171] <- rep(mean(D$hum),length(A[,1]))
A[,172] <- rep(mean(D$wind_spd),length(A[,1]))
A[86,170] <- c(1)

# Making the estimate of slope and st.error

est <- A %*% fit_cor$coefficients[,1]
var_est <- A %*% fit_cor$cov.unscaled %*% t(A) * fit_cor$sigma^2
coef <- data.frame(ID=c(levels(D$scope$ID),0,0,"workingday"), Slope = est,
sd.error=sqrt(diag(var_est)))
wd_coef <- filter(coef, ID == "workingday")

coef <- filter(coef, !ID %in% c(0,0,"workingday"))

n <- nrow(D)
p <- nrow(as.data.frame(fit_cor$coefficients[,1]))
# Adding the confidence interval to dataframe
coef <- mutate(coef, Qup = Slope + qt(0.975,n-p)*sd.error)
coef <- mutate(coef, Qlow = Slope - qt(0.975,n-p)*sd.error)

## Adding characteristics to the buildings
char <- read.xlsx("HTK_building_data_share.xlsx",1,header=FALSE)
char <- rename(char,ID=X1) %>% select(ID,X2)
coef <- left_join(coef,char,by="ID")

D_by_id <- group_by(D, ID) %>% summarise(sum_consumption=sum(consumption))
coef
coef_with_sum <- left_join(coef,D_by_id, "ID")
coef_sum <- rename(coef_with_sum,Type=X2, "Sum consumption"=sum_consumption,

```



```

"Sd. Error" =sd.error, "$Q_{0.025}$" = Qlow, "$Q_{0.975}$" = Qup)
coef_sum
summary(fit)

coef_sum <- mutate(coef_sum, Type=str_split_fixed(Type, " ",2)[,2])
# selecting 10 best and worse buildings
best10 <- arrange(coef_sum, Slope)[1:5,]
worst10 <- arrange(coef_sum, desc(Slope))[1:5,]

xtable(best10, digits = 3)
xtable(worst10, digits = 3)

coef_sorted <- arrange(coef_sum, Slope) %>% mutate(ID = factor(ID)) %>% rename(small='$Q_{0.025}$',
  big='$Q_{0.975}$')
coef_sorted_2 <- mutate(coef_sorted, ID=fct_reorder(ID, Slope))

ggplot(coef_sorted_2) +
  geom_col(aes(y=ID, x=Slope, fill=ID)) +
  geom_errorbar(aes(y=ID, x=Slope, xmin=small, xmax=big)) +
  theme(legend.position = "none") +
  ggtitle("The found slopes with confidence intervals")

```

4. Code for creating plots

```
#### Loading and mutating/formatting the data ####
## Reading in the libraries
rm(list=ls())
library(tidyverse)
library(ggplot2)
library(car)
library(stringr)
library(xtable)
library(lubridate)

## Reading in the data
D <- read.csv("merged_data.csv", header=TRUE)

## Setting the datatypes
# Setting the factors
D <- mutate(D, across(c(dir, cond, fog, rain, ID),factor))
# Setting the dates
D$date <- as.POSIXct(D$date, tz = "UTC")

## Mutating the data
# Adding temp difference column to the dataframe
D$tempdif <- 21 - D$temp

# Adding the weekends as a column
is_weekend <- function(date){
  number_day_df <- wday(date, label=T)
  number_day_char <- as.character(number_day_df)
  return(number_day_char)
}
D <- mutate(D, weekday=is_weekend(date)) %>%
  mutate(weekend = ifelse(weekday %in% c("lø", "sø"), "weekend", "workingday"))

# getting start/end of month int (mutating the date variable)
D <- mutate(D, dag=str_split_fixed(date, "-", 3)[ ,3])
D <- mutate(D, start_or_end = ifelse(as.integer(dag)<15, "START", "END")) %>%
  mutate(start_or_end = factor(start_or_end))

## Removing non important columns

# find the normalised data
mean_each <- group_by(D, ID) %>%
  summarise(mean_each = mean(consumption))
```

```

D_with_mean <- inner_join(mean_each, D, "ID")
D <- mutate(D_with_mean, ncons = consumption/mean_each)

#### Plots for descriptive analysis ####

## START/END IMPORTANT

# PLOT Tempdif and mean normalised consumption
# (both divided by their means)
# against date
cons_mean <- summarise(D, mean(ncons)) %>% as.double()
D_by_date <- group_by(D, date)
D_plot <- summarise(D_by_date, mncons = mean(ncons)/cons_mean)
temp_mean <- summarise(D, mean(tempdif)) %>% as.double()
D2_plot <- mutate(D, mtempdif = tempdif/temp_mean)

ggplot(D_plot) +
  geom_line(aes(x=date, y=mncons), col="Red", size=0.4) +
  geom_line(data = D2_plot, aes(x=date, y=mtempdif),size=0.4) +
  theme(legend.box.background = element_rect(color="red", size=2)+
  labs(x="Date", y="Normalized temperature difference/ mean normalised consumption
  \n (divided by their means)") +
  ggtitle("Tempdif and mean normalised consumption
  (both divided by their means)
  against date")

## Pairsplot ##
#plot(Dplot, col=Dplot$ID, main = "Pairsplot of data")

## Histogram of consumption ##
hist(Dplot$ncons, col="orange", border = "brown", m
ain = "Normalized consumption", xlab="Normalized consumption")

## Scatterplot ##
#It looks like end/start date has an effect
ggplot(D,aes(x=date, y=ncons,col=ID)) +
  geom_point(size=0.8) + theme(legend.position = "none") +
  labs(y="Normalized consumption", x= "Date", title ="Date as a function of normalized consumption")

#nconc ~ tempdif

```

```

ggplot(D, aes(x=temp, y=ncons, col=ID))+
  geom_point(size=0.8)+ theme(legend.position = "none")+
  labs(y="Normalized consumption", x= "Temperature",
  title ="Temperature as a function of normalized consumption")

ggplot(D, aes(x=temp, y=consumption, col=ID))+
  geom_point(size=0.8)+ theme(legend.position = "none")+
  labs(y="Consumption", x= "Temperature", title ="Temperature as a function of consumption")

ggplot(filter(D,ID %in% c("78185925")),aes(x=temp, y=consumption,col=ID)) +
  geom_point(size=0.8)+
  labs(y="Consumption", x= "Temperature", title ="Temperature as a function of consumption",
  legend= D$ID) +
  geom_smooth(method=lm,alpha=0)

ggplot(D, aes(x=tempdif, y=ncons, col=ID))+
  geom_point(size=0.8)+ theme(legend.position = "none")+geom_smooth(method=lm, size=0.5, alpha=0)+
  labs(y="Normalized consumption", x= "Temperature difference",
  title ="NormalisedTemperature difference as a function of normalized consumption")

# But it is explained by the temperature fluxating. Maybe we are modelling the noise
ggplot(D,aes(x=date, y=tempdif,col="black")) +
  geom_point(size=1, col="black") + theme(legend.position = "none") +
  labs(y="Temperature difference", x= "Date", title ="Date as a function of Temperature difference")

plot(Dplot$ncons ~ Dplot$ID, col=c(1:83), main="Boxplot of normalized consumption as a function of ID",
  ylab = "Normalized consumption", xlab="ID")

# PLOT Tempdif and mean normalised consumption
# (both divided by their means)
# against date
cons_mean <- summarise(D, mean(ncons)) %>% as.double()
D_by_date <- group_by(D, date)
D_plot <- summarise(D_by_date, mncons = mean(ncons)/cons_mean)
temp_mean <- summarise(D, mean(tempdif)) %>% as.double()
D2_plot <- mutate(D, mtempdif = tempdif/temp_mean)

ggplot(D_plot) +

```

```

geom_line(aes(x=date, y=mncons), col="Red", size=0.4) +
geom_line(data = D2_plot, aes(x=date, y=mtempdif),size=0.4) +
labs(x="Date", y="Normalized mean temperature difference/
      normalized consumption") +
ggtitle("Normalized mean temperature difference and mean normalised
      consumption as a function of date)
ggplot(D,aes(x=temp, y=dew_pt,col="black")) +
geom_point(size=1, col="black") + theme(legend.position = "none") +
labs(y="Dew point", x= "Temperature", title ="Dew point as a function of temperature")

```

5. Code for working with found model

```
#### Loading and mutating/formatting the data ####
## Reading in the libraries
rm(list=ls())
library(tidyverse)
library(ggplot2)
library(car)
library(stringr)
library(xtable)
library(lubridate)
library("xlsx")

## Reading in the data
D <- read.csv("merged_data.csv", header=TRUE)

## Setting the datatypes
# Setting the factors
D <- mutate(D, across(c(dir, cond, fog, rain, ID),factor))
# Setting the dates
D$date <- as.POSIXct(D$date, tz = "UTC")

## Mutating the data
# Adding temp difference column to the dataframe
D$tempdif <- 21 - D$temp

# Adding the weekends as a column
is_weekend <- function(date){
  number_day_df <- wday(date, label=T)
  number_day_char <- as.character(number_day_df)
  return(number_day_char)
}
D <- mutate(D, weekday=is_weekend(date)) %>%
  mutate(weekend = ifelse(weekday %in% c("lÃ", "sÃ", "Sat", "Sun"), "weekend", "workingday")) %>%
  mutate(weekend = factor(weekend))

# getting start/end of month int (mutating the date variable)
D <- mutate(D, dag=str_split_fixed(date, "-", 3)[ ,3])
D <- mutate(D, start_or_end = ifelse(as.integer(dag)<15, "START", "END")) %>%
  mutate(start_or_end = factor(start_or_end))

# find the normalised data
mean_each <- group_by(D, ID) %>%
```

```

    summarise(mean_each = mean(consumption))
D_with_mean <- inner_join(mean_each, D, "ID")
D <- mutate(D_with_mean, ncons = consumption/mean_each)

## Removing non important columns
D <- select(D,!c("temp","mean_each","dag","weekday"))

#### Removing outliers ####

# Setting the found model up
fit_before <- lm(ncons ~ ID + tempdif + wind_spd + hum + dew_pt +
  pressure + weekend + ID:tempdif + tempdif:weekend + wind_spd:hum +
  tempdif:dew_pt + wind_spd:dew_pt + tempdif:hum + hum:dew_pt +
  ID:weekend + dew_pt:weekend + hum:weekend + dew_pt:pressure,
  data = D)

# Removing outliers
par(mfrow=c(2,2))
plot(fit_before)
outliers <- c(9841, 1639,9478,9477)
D <- filter(D, !row_number() %in% outliers)
fit_before <- lm(ncons ~ ID + tempdif + wind_spd + hum + dew_pt +
  pressure + weekend + ID:tempdif + tempdif:weekend + wind_spd:hum +
  tempdif:dew_pt + wind_spd:dew_pt + tempdif:hum + hum:dew_pt +
  ID:weekend + dew_pt:weekend + hum:weekend + dew_pt:pressure,
  data = D)
par(mfrow=c(2,2))
plot(fit_before)

# Fitting the model again after
D_scope <- select(D, ID, ncons, tempdif, wind_spd, hum, dew_pt, pressure, weekend)
fit_scope <- lm(ncons~. ,D_scope)
fit <- step(fit_scope, scope = ~.^2 , k=log(nrow(D_scope)), test = "F")
# lm(formula = ncons ~ ID + tempdif + wind_spd + hum + dew_pt +
#     pressure + weekend + ID:tempdif + tempdif:weekend + wind_spd:hum +
#     ID:weekend + tempdif:dew_pt + wind_spd:dew_pt + tempdif:hum +
#     hum:dew_pt + wind_spd:weekend + dew_pt:weekend + hum:weekend +
#     dew_pt:pressure, data = D_scope)

summary(fit)
AIC(fit, fit_before)

```



```

0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,
0)
A <- rbind(0, A)
A[,84] <- rep(1,length(A[,1]))

# Making the estimate of slope and st.error
est <- A %>% fit_cor$coefficients[,1]
var_est <- A %>% fit_cor$cov.unscaled %>% t(A) * fit_cor$sigma^2
coef <- data.frame(ID=levels(D_scope$ID), Slope = est, sd.error=sqrt(diag(var_est)))

# Adding the confidence interval to dataframe
coef <- mutate(coef, Qup = Slope + qnorm(0.975)*sd.error)
coef <- mutate(coef, Qlow = Slope - qnorm(0.975)*sd.error)

## Adding characteristics to the buildings
char <- read.xlsx("HTK_building_data_share.xlsx",1,header=FALSE)
char <- rename(char,ID=X1) %>% select(ID,X2)
coef <- left_join(coef,char,by="ID")

# selecting 10 best and worse buildings
worst10 <- arrange(coef, Slope)[1:10,]
top10 <- arrange(coef, desc(Slope))[1:10,]

summary(fit)

worst10
top10

```