Zayed University
ZU Scholars

All Works

1-1-2018

A Refinement of Lasso Regression Applied to Temperature Forecasting

Bruce Spencer
University of New Brunswick

Omar Alfandi
Zayed University

Feras Al-Obeidat
Zayed University, feras.al-obeidat@zu.ac.ae

Follow this and additional works at: https://zuscholars.zu.ac.ae/works

Part of the Computer Sciences Commons

Recommended Citation

Spencer, Bruce; Alfandi, Omar; and Al-Obeidat, Feras, "A Refinement of Lasso Regression Applied to Temperature Forecasting" (2018). *All Works*. 242.
https://zuscholars.zu.ac.ae/works/242

This Conference Proceeding is brought to you for free and open access by ZU Scholars. It has been accepted for inclusion in All Works by an authorized administrator of ZU Scholars. For more information, please contact scholars@zu.ac.ae.

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 130 (2018) 728–735

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Sustainable Energy Information Technology (SEIT 2018)

A Refinement of Lasso Regression Applied to Temperature Forecasting

Bruce Spencer[a,*], Omar Alfandi[b,c], Feras Al-Obeidat[b]

[a] *University of New Brunswick, Fredericton, Canada*
[b] *Zayed Universty, Abu Dhabi, UAE*
[c] *University of Göttingen, Göttingen, Germany*

Abstract

Model predictive controllers use accurate temperature forecasts to save energy by optimally controlling heating, ventilation and air conditioning equipment while achieving comfort for occupants. In a "smart" building, *i.e.* one that is outfitted with sensors, temperature forecasts are computed from data gathered by these sensors. Recently, accurate temperature forecasts have been generated using relatively few observations from each sensor. However, long sensor histories are available in smart houses. In this paper we consider improving forecast accuracy by using up to 24 hours of quarter-hourly readings. In particular, we overcome forecast inaccuracy that arises from the "one standard error" heuristic (1SE) in lasso regression. When there are many historical observations, low variance in the error estimations can result in excessively high values for the lasso hyperparameter λ. We propose the *midfel* refinement of lasso regression, which adjusts λ based on the shape of the error curve, resulting in improved forecast accuracy. We illustrate its effect in a setting where lasso regression is used to select sensors based on forecast accuracy. In this setting, midfel lasso regression using many historical observations has two effects: its improves accuracy and uses fewer sensors. Thus it potentially reduces costs arising both from energy usage and from sensor installation.

© 2018 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the Conference Program Chairs.

Keywords: Home Sensor Network, Temperature Forecasting, Lasso regression, Feature Selection, Model Predictive Control, Energy Efficiency, Internet of Things

1. Introduction

According to recent studies, about 40% of energy produced worldwide is consumed by buildings, and more than half of this is used by Heating, Ventilation and Air Conditioning (HVAC) systems [1,2]. Pan *et al.* [3] point out that, due to thermal inertia, it is more efficient to maintain temperature in a room or building than to raise or lower the temperature. Accurate temperature forecasts can help reduce energy usage in buildings by using future values of temperature when deciding whether or not to activate the HVAC [4]. Moreno *et al.* [5] achieve estimated energy savings of 20% in a realistic

* Corresponding author
E-mail address: bspencer@unb.ca

1877-0509 © 2018 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the Conference Program Chairs.
10.1016/j.procs.2018.04.127

situation based on the presence of persons in a room. Yuan *et al.*[6] achieve 20% savings while exploiting thermal inertia when assigning rooms for meetings by scheduling contiguous meetings in the same room.

Model Predictive Controllers (MPC), which produce a control signal for HVAC systems, minimize a cost function based on energy consumption. The cost function takes into account a prediction horizon and a control horizon[7]. The prediction horizon used in practice depends on how much data is needed by the HVAC controller to achieve acceptable comfort while reducing energy consumption.

Our previous work[8,9,10,11,12,13] uses linear models and generalized linear models to generate accurate temperature forecasts. This previous work used various amounts of historical sensor data, and was not often able to improve forecast accuracy by using sensor histories longer than four hours. In this paper we explore this question: Can more data lead to more accurate forecasts? We provide up to 24 hours of data which allows the model the opportunity to incorporate cyclical patterns that occur at the same time each day.

More observations of the sensor data tends to lower variance in the estimations of prediction error. Lasso regression uses a heuristic called *one-standard error* (1SE) based on this variance. We suggest a refinement of this heuristic called *midfel*, that can improve forecast accuracy when the variance of the estimated forecast error is low. Midfel lasso regression improves forecast accuracy over lasso regression. Improved forecast accuracy can lead to energy savings.

We apply midfel lasso regression in a scenario where sets of sensors are considered[11,12]. In addition to improving forecast accuracy we show midfel lasso regression can reduce the number of needed sensors. This can lead to reductions in costs of sensor installation and maintenance.

In the remainder of the paper we review temperature forecasting based on generalized linear regression with lagged sensor data. We discuss the 1SE heuristic for setting the hyperparameter λ. We then explain the midfel heuristic which uses both the variance in the forecast error and the shape of the error curve to adjust λ. We analyse experimental evidence showing improved forecast error from fewer sensors. We discuss the potential for applications of midfel lasso regression for the Internet of Things, where long histories of sensor data may be available.

2. Background

2.1. Data from a Smart Home

The SML House[4] competed in the Solar Decathalon 2012 competition[14], using 88 sensors and 49 actuators. In this paper and in our previous work, we use a publicly available subset of this data[15], reporting values during March and April 2012 from 18 sensors every quarter-hour. The sensors reported are listed in Table 1.

Wi	wind speed	Pcp	precipitation
Tw	twilight indicator	P	sun irradiance measured by a pyranometer
TP	predicted temperature	LL	lights in the living room
TL	living room temperature	LD	lights in the dining room
TD	dining room temperature	HL	humidity in the living room
T	external temperature	HD	humidity in the dining room
SW	sun on the west wall	H	external humidity
SS	sun on the south wall	CL	carbon dioxide sensor in the living room
SE	sun on the east wall	CD	carbon dioxide sensor in the dining room

Table 1: Sensors in the SML house

.

2.2. Linear and Lasso Regression

The forecasting methods in this paper are based on linear regression. Given a set of independent predictor variables $x_1, ..., x_n$ and a dependent variable y of interest that we want to forecast, we seek parameters $\beta_0, ..., \beta_n$ so that $\beta_0 + \Sigma_{i=1}^n \beta_i x_i$ is a good approximation of y. When presented with a set of m instances of each x_i, called $x_{i,j}$ and the corresponding instances y_j, we select the β_i parameters to minimize the residual sum of squares (RSS):

$$\Sigma_{j=1}^m (\beta_0 + \Sigma_{i=1}^n \beta_i x_{i,j} - y_j)^2$$

Lasso regression [16] minimizes RSS + λ $\Sigma_{j=1}^m |\beta_j|$ where λ is a tuning parameter that balances the emphasis between reducing error and using small β coefficients, Some β may reduce to zero, which deselects that variable x, thus endowing lasso regression with a method of pruning predictors in the model.

For lasso regression, we use the R library `glmnet` [17,18,19]. In this implementation, a value of λ is selected to use when building a model based on all of the test data. The final λ is selected from one of many values of λ used during ten-fold cross validation. The training data is partitioned into ten sets randomly. For each partition, a model is built in which nine sets are used to build a linear regression model, and one is held back. This gives ten different models for each λ. Each model thus has ten error estimates. From these ten, a mean and stardard deviation is computed. The λ with the minimal mean error estimate is called λ-`Min`. If λ-`Min` is chosen as the final λ, the final model tends to be overfit. To avoid this, the 1SE heuristic selects a larger λ whose mean is within one standard deviation of the ten error estimates for λ-`Min`. This choice is called the λ-`1SE`. Because λ-`1SE` is larger than λ-`Min`, more β are set to zero, which helps avoid overfitting. See Figure 1. This shape of the error curve in this figure is similar to others encountered when forecasting temperature from sensor readings.

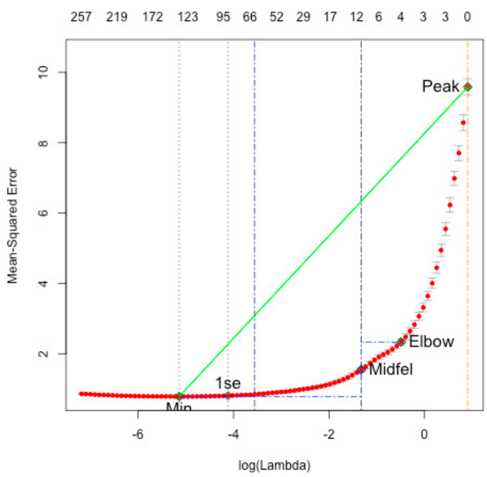

Fig. 1: Forecasting the internal temperature five hours into the future based on 24 hours of observations from four sensors: Tw, T, SW and P. Each point on the error curve is the estimated error for some selected λ. It is computed during 10-fold cross validation. For each fold, a mean squared error (MSE) is computed based the model's predicted temperature and the observed temperature. The mean of these is shown within an interval ranging from one standard error above to one standard error below. The minimal MSE is labelled `Min`. We use λ-`Min` to refer to the λ that gives rise to this `Min` error. We use λ-`1SE` to refer to the λ whose error is within one standard deviation of the ten error estimates for λ-`Min`. In Section 4 we discuss the other labeled points: Midfel, Elbow and Peak.

3. Models Using Lagged Sensor Readings

When creating a model from which to forecast temperatures, we provide multiple historical readings from each sensor. Given a history of b time periods, where readings are taken every quarter-hour, we provide $b + 1$ lagged readings from each of s sensors, which includes the current period at lag 0. Let $x_{k,t}$ be the t^{th} observation for sensor k

counting from the first observation at time $t = 1$, as it appears in the training data. Let y_t be the internal temperature the house at time t. We are given observations over the m time periods in the training data. We create a linear a model for each future period f. We define the RSS as

$$\text{RSS}(f) = \Sigma_{t=b+1}^{m}(\beta_{f,0} + \Sigma_{g=0}^{b}\,\Sigma_{k=1}^{s}\,\beta_{f,k,g}x_{k,t-g} - y_{f+t})^2$$

In this equation, t starts at b+1 because there are no observations for the lagged readings for the first b data points. Using lasso regression, we choose values for the coefficients $\beta_f = \{\beta_{f,0}\} \cup \{\beta_{f,k,g} \,|\, g = 0, \ldots, b,\ k = 1, \ldots, s\}$ where g identifies the lag and k identifies the sensor. The coefficients in β_f specify a model for each future interval f. We use two different forecast horizons; h is either 12 or 48 future time periods, i.e. 3 or 12 hours.

The coefficients are computed on the training data which is the first 2/3 of the data. Once they are computed, we switch over to using test data, which is the final 1/3 of the data. Thus x and y below refer to observations in the test data and m to the number of observations in the test data. We report the root mean squared error (RMSE) for each future interval f. In our experiments $f = 1, \ldots, 12$ for forecasts three hours into the future, and $f = 1, \ldots, 48$ for forecasts to 12 hours.

$$\text{RMSE}(f) = \sqrt{1/(m-b)\,\Sigma_{t=b+1}^{m}(\beta_{f,0} + \Sigma_{g=0}^{b}\,\Sigma_{k=1}^{s}\,\beta_{f,k,g}x_{k,t-g} - y_{f+t})^2}$$

We report error metrics on all forecasts f over the forecast horizon h, including Mean RMSE $= 1/h\,\Sigma_{f=1}^{h}\text{RMSE}(f)$ and Maximal RMSE = $max_f\text{RMSE}(f)$.

4. The Midfel Refinement of Lasso Regression

Lasso regression is a successful accurate forecasting method, which uses a simple and elegant mechanism to prevent the model from overfitting the data. As explained in Section 2.2, it is based on the variance observed when estimating the forecast error during cross-fold validation. In particular, it focuses on the mean and the variance that occurred when using the λ with the minimal error. It selects the λ-`1SE` that has a slightly higher but error, but one that is limited, based on this variance.

The midfel refinement of lasso regression, like lasso regression, uses the 1SE method. Unlike lasso regression, it also uses the shape of the error curve. It is particularly effective when the forecast model is based on a large amount of data from a stable situation, because then the variance used by 1SE tends to be small. In this case the 1SE heuristic can be too conservative. Some larger value of λ can have better performance. These larger values constrain the β coefficients more, which tends to set more of them to zero. Thus, using a larger value of λ can help lasso regression to eliminate predictors from the model. This allows us to exploit a resource that is quite often free: long sensor histories. However, larger values of λ can often lead to larger prediction errors, so the shape of the error curve must be considered. We want to carefully limit how much further we increase λ beyond the 1SE value. Given an error curve, an experienced person can estimate a good value of lambda, but how can this be done in general by an algorithm?

The midfel refinement of lasso regression is based on looking for the point where the error starts to increase quickly, and limiting λ to less than half of that additional error.

We refer again to Figure 1. In this model there are four sensors, and for each there is one predictor reporting the current value of the sensor and four more predictors per hour reporting the previous 24 hours, giving a total of $4 \times (1 + 4 \times 24) = 388$ predictors in this model. The number of selected predictors, *i.e.* those that have non-zero coefficients, depends on the choice of a value for λ. For each of λ-`Min`, λ-`1SE`, λ-`Midfel`, the number of selected predictors is 130, 94, and 12, respectively, as shown in the labels above the top of the graph. The forecast error from using λ-`1SE` is 0.89°C and the forecast error using λ-`midfel` is 0.85°C, an improvement of almost 5%.

To find λ-`Midfel` a line is drawn from the minimal MSE to the first peak in the error curve at larger values of λ, labelled `Peak`. There may be several peaks on the error curve, but we consider only the first one to the right of `Min`. The point on the error curve that is furthest from this line is labelled `Elbow`, because it represents the point where the curve bends upwards. The point labelled `Midfel` is the point on the error curve whose error is midway between `Min` and `Elbow`. The name midfel derives from "**mid**way from the **f**irst **el**bow".

When using the midfel lasso regression, the choice of λ may need a futher adjustment. In our experiments, we choose a value between λ-`1SE` and λ-`Midfel`. The between these ratio is called the *midfel balance*, defined as

$$\lambda_{\texttt{balance}} = exp(log(\lambda_{\text{1SE}}) + (log(\lambda_{\texttt{Midfel}}) - log(\lambda_{\text{1SE}})) * \texttt{balance}).$$

The midfel balance can be set to a fixed value between 0 and 1, or it can be determined during hyperparameter training. In experiments described in the next section, the midfel balance is set to 20%, so it is closer to λ-1SE. In Figure 1, λ-balance is shown as a blue line and it selects 65 predictors.

5. Experimental Results

We perform two sets of experiments to investigate the effect of midfel lasso regression on the data from the SML house. In all cases, we use 24 hours of sensor data. For all of the midfel runs, we use a balance parameter of 0.2.

5.1. Midfel Lasso Regression compared with Lasso Regression on Small Sets of Sensors

A modest number of tests is run for small sets of sensors to compare 1SE lasso regression and midfel lasso regression. Using each of the two regression methods, we forecast temperatures 48 times, to make up 12 hours of quarterly-hour forecasts. We calculated the root mean squared error for each forecast and report the maximal of these 48 RMSE values. We repeat this test 52 times selecting different sets of sensors. The results are shown in Table 2.

In 34 of these 52 cases, the lower error was computed by the `midfel` method, in one case the errors were identical in the first four decimal places, and in 17 cases the lower error was computed using the 1SE method. When using one or two sensors, the advantage was enjoyed by 1SE and midfel about evenly. Of the 35 rows in Table 2 for one or two sensors, 1SE regression's error is lower 16 times and midfel regression's is lower 18 times. For the 17 cases where more than two sensors are used, midfel holds the advantage 16 times. This seems to indicate that midfel is better able to take advantage of the variety of information offered by the different sensors.

5.2. Eliminating Sensors with Midfel Regression

In Section 4 we saw that one of the advantages of midfel regression was a reduction in the number of predictors in the model, as illustrated in Figure 1. It would be useful if we could somehow exploit this reduction so that we could use midfel regression to entirely eliminate the need for a specific sensor. This would occur, for instance, if the coefficients for all lagged values of this sensor were set to zero by the midfel restriction. Moreover, we would need this sensor to be eliminated for each of the quarter-hourly forecast models. This likelihood of this coincidence seems remote. However, all is not lost. When forecasting temperature from a large number of sensors, there is some redundancy in the information from various sensors. Perhaps we can swap some sensors for others in such a way that we can eliminate some sensors, while retaining others that offer similar information. In this way, we may be able to find sensors to remove without compromising accuracy.

In recent work[11] we propose a technique for finding a set of sensors that gives accurate forecasts. This technique performs a best-first search through the space of all possible sets of sensors. In this paper we repeat our earlier work, using both 1SE and midfel lasso regression. We are motivated to perform this experiment based on our observation in the previous section, which suggested that midfel regression tends to outperform lasso regression when at least a few sensors are available.

We perform two experiments, one creating quarter-hourly forecasts for the next 3 hours, and the other for the next 12 hours. Error is meaured as maximal RMSE over these forecasts. In each case we use both 1SE lasso regression and midfel lasso regression. The results, in Tables 3 and 4, show sets of sensors in the order of decreasing lengths, arranged so that the sensors can be removed one by one to generate the next set in the sequence. The sequence is constructed so that the error increases as the number of sensors decreases. Given this sequence, one can decide between how much error one can tolerate and how many sensors one wants to install. The largest set of sensors in each sequence also has the property that no additional sensor gives a smaller error.

For 3 hour forecasts using lasso regression, the set TP+TL+SW has lowest error at 0.4643. For midfel lasso regression, we achieve a lower error, 0.4581, with a smaller set of sensors, TW+TL. For 12 hour forecasts using lasso regression, the set TD+T+SW+SS+P produces an error of 1.0959. Using midfel lasso regression, we achieve a lower error, 1.084, using a smaller set of sensors, Tw+T+P+LD. The results show that midfel regression is able to produce more accurate forecasts using fewer sensors. This saves two kinds of costs. First saves the cost of installing some sensors. Second it potentially saves energy by generating a more accurate forecast for the model predictive controller.

	Sensors	1SE	Midfel	Difference
1:	CD	2.5455	2.5554	-0.0099
2:	CL	2.5160	2.5861	-0.0701
3:	H	3.1396	3.1344	0.0052
4:	HD	3.0811	3.0877	-0.0066
5:	HL	3.3464	3.3464	0.0000
6:	LD	2.3153	2.2773	0.0380
7:	LL	2.3052	2.3074	-0.0022
8:	P	2.2440	2.2500	-0.0060
9:	Pcp	2.2818	2.2792	0.0026
10:	SE	3.2255	3.2097	0.0158
11:	SS	3.0446	2.9537	0.0909
12:	SW	2.2329	2.2037	0.0292
13:	T	1.1889	1.1898	-0.0009
14:	TD	1.2862	1.2865	-0.0003
15:	TL	1.2657	1.2683	-0.0026
16:	TP	1.4686	1.4631	0.0055
17:	Tw	2.1978	2.1892	0.0086
18:	Wi	2.0400	2.0321	0.0079
19:	T+CD	1.2064	1.2068	-0.0004
20:	T+CL	1.2090	1.2111	-0.0021
21:	T+H	1.9424	1.9474	-0.0050
22:	T+HD	1.4801	1.4854	-0.0053
23:	T+HL	1.7692	1.7739	-0.0047
24:	T+LD	1.1743	1.1710	0.0033
25:	T+LL	1.1335	1.1315	0.0020
26:	T+P	1.1115	1.1110	0.0005
27:	T+Pcp	1.4426	1.4149	0.0277
28:	T+SE	1.2599	1.2531	0.0068
29:	T+SS	1.2049	1.2085	-0.0036
30:	T+SW	1.1626	1.1632	-0.0006
31:	TD+T	1.1626	1.1605	0.0021
32:	TL+T	1.1673	1.1665	0.0008
33:	TP+T	1.1726	1.1735	-0.0009
34:	Tw+T	1.1232	1.1205	0.0027
35:	Wi+T	1.2354	1.2095	0.0259
36:	T+P+CD	1.1436	1.1328	0.0108
37:	T+P+CL	1.1477	1.1433	0.0044
38:	T+P+H	1.8902	1.7834	0.1068
39:	T+P+HD	1.6432	1.5906	0.0526
40:	T+P+HL	1.7433	1.6980	0.0453
41:	T+P+LD	1.1111	1.1103	0.0008
42:	T+P+LL	1.1330	1.1292	0.0038
43:	T+Pcp+P	1.6442	1.5845	0.0597
44:	T+SE+P	1.2691	1.2475	0.0216
45:	T+SS+P	1.1730	1.1499	0.0231
46:	T+SW+P	1.1064	1.1121	-0.0057
47:	TD+T+P	1.1108	1.1081	0.0027
48:	TL+T+P	1.1156	1.1141	0.0015
49:	TP+T+P	1.1490	1.1323	0.0167
50:	Tw+T+P	1.1099	1.0888	0.0211
51:	Wi+T+P	1.3033	1.2581	0.0452
52:	Tw+T+SW+P	1.1180	1.0981	0.0199

Table 2: Maximal RMSE for 12-hour forecasts, comparing 1SE and midfel lasso regression over various sets of sensors. The tests include all of the 18 individual sensors, and other promising combinations, including 17 combinations of two sensors, 16 combinations of three sensors and one combination of four. They are ordered first by size and second alphabetically by the names of the sensors.

Table 3: Best-First search based on 24 hours of data, for the next 3 hours

(a) Fitting strategy `1SE`

Maximal RMSE	Sensors	Remove Next
0.4643	TP+TL+SW	SW
0.4653	TP+TL	TP
0.4696	TL	

(b) Fitting strategy `midfel-0.2`

Maximal RMSE	Sensors	Remove Next
0.4248	Tw+TL+SW+SS+CD	CD
0.4307	Tw+TL+SW+SS	SW
0.4407	Tw+TL+SS	SS
0.4581	Tw+TL	Tw
0.487	TL	

6. Comparison with Related Work

The SML team[20], who provided the data that we analyse, generated temperature forecasts using this same data. They also considered subsets of sensors, from among these: internal temperature (TD and TL), irradiance (P), internal

Table 4: Best-First search based on 24 hours of data, for the next 12 hours

(a) Fitting strategy `1SE`

Maximal RMSE	Sensors	Remove Next
1.0959	TD+T+SW+SS+P	SS
1.0995	TD+T+SW+P	TD
1.1064	T+SW+P	SW
1.1115	T+P	P
1.1889	T	

(b) Fitting strategy `midfel-0.2`

Maximal RMSE	Sensors	Remove Next
1.084	Tw+T+P+LD	LD
1.0888	Tw+T+P	Tw
1.111	T+P	P
1.1898	T	

humidity (HD and HL), and precipitation (PCP). They used combination of forecast models based on artificial neural networks. Based on their results, a selection of three sensors gave the lowest errors: internal temperature, solar irradiance, and a time-categorical variable. Our results show some agreement: temperature was the most important, and humidity and precipitation were not of any help. Unlike their result, we found that the pyranometer was not of any help. They seem not to have used sensors that we found were helpful, including the sun on each wall, the CO_2 in the dining room, and the living room lights.

The SML team reports[21,4] accuracy when forecasting temperature differences over future quarter-hour intervals, using data from two of their 88 sensors: internal temperature and sun irradiance, as well as a time categorical variable. Forecasting temperature differences is an area of future work for midfel regression.

Feature extraction shares some similarities with feature selection. Feature extraction is the process of defining new features from existing ones. It proceeds by selecting those features with good predictive accuracy, and repackaging them into linear combinations that are considered new features. Partial least squares and principal component analysis are two feature extraction techniques[22,23]. We used the same SML data for partial least squares and principle components[10]. Using four historical readings per sensor, we found the RMSE forecast error for both methods to be about for 1.7°C for twelve-hour forecasts. The results were similar for eight historical readings per sensor. In comparison, we achieved a much lower error of about 1.1°C here using both 1SE lasso regression and midfel lasso regression. Note that these two lasso regression methods were able to take advantage of 24 hours of historical data, whereas the feature extraction techniques did not improve when four more hours of historical data were added to the intial four hours.

7. Conculsions and Future Work

We present midfel lasso regression, a refinement of lasso regression that computes a larger value of λ than the standard 1SE value. The midfel refinement uses both the estimate of variance of the minimal forecast error and the shape of this error curve over various values of λ. A parameter balances between these two criteria. When the balance parameter is set to 0, the 1SE value for λ is selected. In this case, midfel lasso regression computes exactly the same models as lasso regression. When the midfel balance parameter is well selected, midfel lasso regression can outperform lasso regression.

We empirically study a sitation where the data is relatively stable: temperature forcasting in a smart house. Many hours of observations are readily available. In two experiments we have shown that midfel lasso regression reduces the overfitting that is observed for lasso regression, and improves accuracy. Improved accuracy can reduce energy costs. Our predictors are based on sensor readings. Since the predictors are correlated, we use a best first search technique to select a small set of sensors, which also saves on capital costs.

The incentive for this this paper is to find a way to effectively use long sensor histories. In the Internet of Things, sensor networks are prevalent, and long sensor histories are frequently available. Long forecasts horizons are also important, such as delaying heating events until fluctuating energy costs are forecast to be low. We will investigate the performance of midfel lasso regression for various sensor histories and forecast horizons.

Acknowledgements

The authors acknowledge the support of their organizations, especially Zayed University's RIF fund 17063.

References

1. P. Morosan, R. Bourdais, D. Dumur, J. Buisson, Building temperature regulation using a distributed model predictive control, Energy and Buildings (2010) 14451452.
2. L. Pérez-Lombard, J. Ortiz, C. Pout, A review on buildings energy consumption information, Energy and Buildings 40 (2008) 394398.
3. D. Pan, D. W. Y. Yuan, X. Xu, Y. Peng, X. Peng, P.-J. Wan, Thermal inertia:towards an energy conservation room management system, in: A. Greenberg, K. Sohraby (Eds.), INFOCOM, IEEE, 2012, pp. 2606 – 2610.
4. F. Zamora-Martínez, P. Romeu, P. Botella-Rocamora, J. Pardo, On-line learning of indoor temperature forecasting models towards energy efficiency, Energy and Buildings 83 (2014) 162–172.
5. M. V. Moreno, M. A. Zamora, A. F. Skarmeta, User-centric smart buildings for energy sustainable smart cities, Transactions on Emerging Telecommunications Technologies 25 (1) (2014) 41–55. doi:10.1002/ett.2771.
6. Y. Yuan, D. Pan, D. Wang, X. Xu, Y. Peng, X. Peng, P.-J. Wan, A study towards applying thermal inertia for energy conservation in rooms, ACM Trans. Sen. Netw. 10 (1) (2013) 7:1–7:25. doi:10.1145/2529050.
7. J. Álvarez, J. Redondo, E. Camponogara, J. Normey-Rico, M. Berenguel, P. Ortigosa, Optimizing building comfort temperature regulation via model predictive control, Energy and Buildings 57 (2013) 361 – 372.
8. B. Spencer, O. Alfandi, Forecasting internal temperature in a home with a sensor network, in: Machine Learning and Data Mining for Sensor Nets, 2016. doi:10.1016/j.procs.2016.04.259.
9. B. Spencer, F. Al-Obeidat, Temperature forecasts with stable accuracy in a smart home, in: SEIT Sustainable Energy Information Technology Conference, 2016. doi:10.1016/j.procs.2016.04.160.
10. B. Spencer, F. Al-Obeidat, O. Alfandi, Short term forecasts of internal temperature with stable accuracy in smart homes, International Journal of Thermal and Environmental Engineering 13 (2) (2016) 81–89.
11. B. Spencer, F. Al-Obeidat, O. Alfandi, Selecting sensors when forecasting temperature in smart buildings, Procedia Computer Science 109 (Supplement C) (2017) 777 – 784, 8th International Conference on Ambient Systems, Networks and Technologies, ANT-2017 and the 7th International Conference on Sustainable Energy Information Technology, SEIT 2017, 16-19 May 2017, Madeira, Portugal. doi:https://doi.org/10.1016/j.procs.2017.05.321.
12. B. Spencer, F. Al-Obeidat, O. Alfandi, Accurately forecasting temperatures in smart buildings using fewer sensors, Personal and Ubiquitous Computingdoi:10.1007/s00779-017-1103-4.
13. F. Al-Obeidat, B. Spencer, O. Alfandi, Consistently accurate forecasts of temperature within buildings from sensor data using ridge and lasso regression, Future Generation Computer Systems, in press (2018).
14. United States Department of Energy, Solar Decathlon Europe Competition, `http://www.solardecathlon.gov` (2012).
15. UCI, Sml2010 data set, `https://archive.ics.uci.edu/ml/datasets/SML2010` (2010).
16. R. Tibshirani, Regression shrinkage and selection via the lasso: a retrospective, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 58 (1) (1996) 267–288.
17. J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent, Journal of Statistical Software 33 (1) (2010) 1–22.
18. N. Simon, J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for cox's proportional hazards model via coordinate descent, Journal of Statistical Software 39 (5) (2011) 1–13.
19. J. Friedman, T. Hastie, N. Simon, R. Tibshirani, Package `glmnet`: Lasso and Elastic-Net Regularized Generalized Linear Models Ver 2.0-, `https://cran.r-project.org/web/packages/glmnet/glmnet.pdf` (March 15 2016).
20. M. V. Moreno, J. Santa, M. A. Zamora, A. F. Skarmeta, Context-aware energy efficiency in smart buildings, in: 17th International Conference on Ubiquitous Computing & Ambient Intelligence (UCAmI 2013), 2013. doi:10.1109/WF-IoT.2015.7389140.
21. F. Zamora-Martínez, P. Romeu, P. Botella-Rocamora, J. Pardo, Towards energy efficiency: Forecasting indoor temperature via multivariate analysis, Energies 6 (9) (2013) 4639. doi:10.3390/en6094639.
22. R. W. Bjørn-Helge Mevik, The `pls` Package: Principal Component and Partial Least Squares Regression in R, Journal of Statistical Software 18 (2) (2007) 1–24. doi:10.18637/jss.v018.i02.
23. B.-H. Mevik, R. Wehrens, K. H. Liland, pls: Partial Least Squares and Principal Component Regression, r package version 2.5-0 (2015).