Pattern Selection Problems in Multivariate Time-Series using Equation Discovery

Arne Koopman, Arno Knobbe, Marvin Meeng
LIACS
Universiteit Leiden
akoopman@liacs.nl

ABSTRACT

In this paper, we present a method for pattern selection in collections of patterns discovered in multivariate time-series. Because our data is continuous in nature, the pattern language we consider is somewhat out of the ordinary, compared to the common discrete patterns considered in the data mining field. An equation discovery system is employed to generate either regular algebraic equations, or more complex differential equations. As the equation discovery system generates a collection of equations per target variable, and we require equations for each variable, we are dealing with an abundance of equations, quite likely with serious levels of redundancy. The method presented here selects a subset of equations by considering to what extent the different variables are covered by the selected equations, while optimising the relevance of variables within the equations. As such, the equation selection method returns a concise set of equations, that captures the dependencies between the different time-series well, while minimizing redundancy. The work in this paper is inspired by the new InfraWatch project, which deals with high-resolution sensor data from a highway bridge. The 145 sensors (sensing structural characteristics such as stretch, vibration and temperature) are distributed fairly densely over the bridge, such that adjacent sensors are likely to show correlated signals. Especially in an exploratory setting, one would be interested in a small collection of prototype sensors with associated equations for how these prototypes are related to other sensors in the vicinity. In the experimental section, we demonstrate how the sensors can be modeled by (differential) equations, and how the equation selection method picks relevant equations that models structural properties of the bridge sensibly.

1. INTRODUCTION

This paper is concerned with multivariate time-series, specifically with data collected by a series of sensors measuring at a steady frequency. In the typical case, these sensors measure the state of a certain system at various locations, such that the measured signals will show a certain degree of correlation. Our aim is to model such correlations by means of patterns discovery. Compared to traditional pattern discovery, our data is challenging in two particular ways. First, the data is of *continuous* nature, which excludes the majority of (discrete) pattern discovery approaches. Second, the data is measured as a function of *time*, which implies a certain dependency between consecutive measurements for a given sensor. In order to deal with these two challenges, we employ an equation discovery system that is capable of finding both regular algebraic equations (the continuous aspect) as well as differential equations (the temporal aspect). The outcome of the equation discovery process is a (potentially large) collection of equations which model the value of one sensor as a function of a small number of other sensors. Such equations, and how well they approximate reality, can be used by analysts to find elementary relations between components of the observed system, and may also suggest possible redundancies in the sensor network.

The equation discovery system in question is the *Lagramge* system, developed as an extension of the earlier Lagrange system by Todorovski and Džeroski [4, 5, 13]. The system considers a grammar of well-formed equations, and constructs candidate equations in a fashion reminiscent of ILP or multi-relational systems [7, 5]. That is, the essentially structural descriptions of the equations are constructed top-down, with progressively more complex equations being built by adding new variables (the sensors) and functions. A specific difference to said relational approaches is the component of Lagramge which fits parameters of the equations by means of the downhill simplex or Levenberg-Marquart algorithms [12]. As such, the system combines a pattern discovery-inspired component with an error-minimization component. The typical setting of a Lagramge run is to select a target sensor s_x and specify a declarative bias (the equation grammar) after which the system returns a list of possible equations. Each equation includes a right-hand side which involves a number of sensors. The equations differ in the sensors involved, and in the constants that were fitted, and consequently in the error on the data.

Although Lagramge produces the desired information (a collection of equations modeling local dependencies in the observed system), it suffers from a problem that is common to most pattern discovery system: an abundance of results. This abundance comes from a number of sources. First of all, different equations may model the dependencies equally well, approximately. This redundancy is especially apparent if two or more sensors are highly correlated, such

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
KDD'10 Workshop, July 25–July 28, 2010, Washington D.C., USA.
Copyright 2010 ACM 978-1-60558-495-9/09/06 ...$10.00.

Figure 1: (left) Aerial picture of the situation of the Hollandse Brug, which connects the 'island' Flevoland to the province Noord-Holland, and the adjacent railway bridge. (right) Some of the sensors attached to the underside of the bridge.

that one of the sensors can simply be exchanged for another, without essentially affecting the model. Furthermore, there may be terms in the equation that do not substantially contribute to the overall fit, for example by having insignificant constants. Finally, there is redundancy produced by running Lagramge once for each potential target sensor (all sensors once, if need be), such that one sensor may be modeled in terms of another, and vice versa. Note that all of these instances of pattern-redundancy appear in other pattern discovery settings as well, including overlapping conditions, irrelevant conditions and so on. Observing this analogy between the equation selection problem and the pattern selection problem, we propose to select a small but relevant collection of equations, using a method that is inspired by a number of recent pattern selection methods [8, 1].

The patterns selection methods mentioned are centered around the idea of greedy forward selection of the patterns. Starting with an empty set of patterns C, all remaining patterns are considered in turn, and a pattern f is added if it improves the quality of $C \cup f$ substantially. This process continues until either some stopping-criterion is met, or all patterns have been included. In the case of patterns (which are typically interpreted as binary features), the *quality* of $C \cup f$ is often a measure of the joint entropy of $C \cup f$ [8], or the mutual information between C and f [1]. As an alternative, more syntax-oriented interpretations of quality may be employed, for example to optimize coverage of all items in itemsets. This is one of the approaches that we will assume in our equation selection method. Replacing patterns by equations, we will assume that a specific equation selected in C accounts for all sensors that appear in the equation. Therefore, a new equation that features only sensors that do not yet appear in any of the elements of C is a desirable addition to C. Our method thus adds equations that introduce as many new sensors as possible.

Our work on equation discovery in multivariate time-series is inspired by a recently started project, called InfraWatch [6]. In this project, we deal with 145 sensors attached to, or embedded in, the concrete of a large highway bridge in the Netherlands (see Section 2). These sensors measure the wheather and traffic load on various locations of the bridge at a frequency of 100Hz. Because the sensors are distributed over the bridge, and vibrations are conducted through the rigid structural elements, nearby sensors will be correlated. Furthermore, sensors of different nature (e.g., stretch vs. vibration) may be co-located, such that potentially differential equations may be required to model the difference in physical properties these types of sensors measure. Especially for exploratory purposes, an analysts would be served be a concise, yet informative set of sensors. Although our main application in this paper is related to the InfraWatch project, one could apply the same techniques to other data of similar kind. One example can be found in the Adaptive System Management problem [9, 10], where large collection of *monitors* are continuously measuring the state and health of different components in an IT-system, and clear numeric dependencies, even of differential nature, between the load in certain components exist.

2. INFRAWATCH

The InfraWatch project is centred around an important Dutch highway bridge: the Hollandse Brug. This bridge is located between the Flevoland and Noord-Holland provinces, at the place where the Gooimeer joins the IJmeer (see Figure 1 on the left). It was opened in June 1969, and in April 2007 it was announced that measurements would have shown that the bridge did not meet the quality and security requirements. Repairs were launched in August 2007 and a consortium of companies has installed a monitoring configuration underneath the Hollandse Brug with the main aim to collect data for evaluating how the bridge responds. The sensor network is part of the strengthening project which was necessary to upgrade the bridge its capacity by overlay-

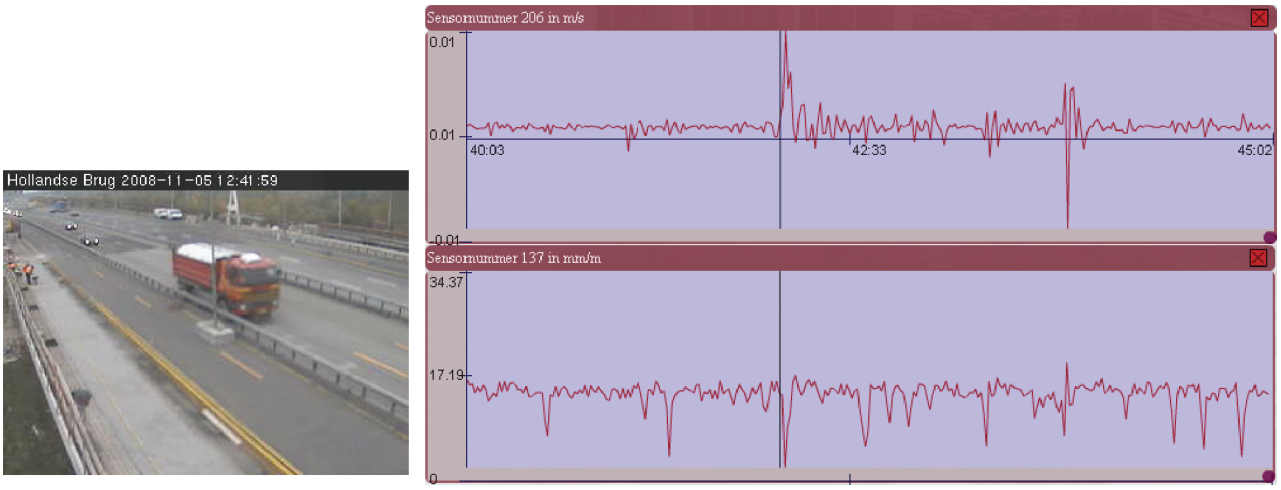

Figure 2: Example of a truck passing the single camera located on the bridge. The graphs show the signal of two sensors, with a vertical bar indicating the time that corresponds to the shown video frame.

ing.

The monitoring system comprises 145 sensors that measure different aspects of the condition of the bridge, at several locations along the bridge (see Figure 1 on the right). The following types of sensors are employed:

- 'geo-phones' (vibration sensors) that measure the vertical movement of the bottom of the road-deck as well as the supporting columns.
- strain-gauges embedded in the concrete and attached to the outside, measuring horizontal stress in two directions.
- thermometers embedded in the concrete and attached to the outside.

Furthermore, there is a weather station, and a video-camera that provides a continuous video stream of the actual traffic on the bridge. Additionally, there are plans to monitor the adjacent railway bridge.

Prior to the start of the InfraWatch project, an initial monitoring application was developed by a team of students, that allows the visual inspection of both video and sensor information. The application allows the user to navigate through a selected time-frame, and display the traffic passing over the bridge, while the data over one or more sensors is displayed in synchronised fashion (see Figure 2). The user can select the nature of the sensor as well as the location of it, which does not necessarily have to correspond with the location of the camera. Using this application, it is fairly easy to already observe some patterns in the data. For example, the vertical load data nicely corresponds with heavy vehicles passing.

3. EQUATION SET SELECTION

As part of our first efforts on this data, we aim to find characteristic sensors within the whole sensor sytem S that can be regarded as a representation of a set of other sensors. For example, if sensor s_1 demonstrates the same behaviour as sensors s_2 and s_3, if suffices to consider only s_1 as a prototype for these three sensors.

In order to determine the behaviour of the sensors, we consider the measurement data that is gathered from each sensor. That is, for each sensor s, we have a signal consisting of a stream of continous data that represents some measured aspect of the bridge. We simply denote the signal value of sensor i at timestamp t by: $s_i(t)$. Within the measurement period T, we know at every timestamp $t \in T$ the $s_i(t)$ for each sensor.

Based on these values, we can now find equations that predict the signal values of one sensor based on the reading of others. In theory, we can apply any class of equation that described relations between the sensors. However, as our interest is in finding simple relations between sensors, we initially focus on linear equations. In other words, an equation f_x that approximates s_x over time has the following form:

$$f_x(t) = c_0 + \sum_{s_y \in S} c_y \cdot s_y(t)$$

where $s_y \in S, x \neq y, c_y \in \mathbb{R}$.

We denote the length of an equation f, $L(f)$, as the number of $s_y \in S$, for which $c_y \neq 0$.

Additionaly, we are not so much interested in *any* equation that describes relationships between sensors, but rather in *good* equations. Therefore, we restrict the set of equations to those equations that are able to closely approximate the signal value of the target sensor:

$$\sum_{t \in T} |s_x(t) - f_x(t)| \leq \epsilon \cdot |T|$$

where ϵ is the error threshold. The term $|T|$ compensates for differences in the size of the time window considered, and thus allows a definition of ϵ independent of $|T|$. Given the set of signals, we can now find a set of candidate equations C that match these requirements, by means of Lagramge.

This typically results in many equations, and even worse, many of these equations describe the same behaviour for the same set of sensors. This makes this problem essentially a pattern subset selection problem, and we therefore focus on the selection of a subset of equations that reduces the redundancy of this equation set.

Algorithm 1 ForwardSelection

FowardSelection(C)

1. $F = \varnothing; Q = 0$
2. **for all** $f \in\downarrow C$ **do**
3. $\quad F' = F \cup f$
4. $\quad Q' = 0$
5. $\quad$ **for all** $f' \in F'$ **do**
6. $\quad\quad Q' = Q' + \mathit{quality(f)}$
7. $\quad$ **end for**
8. $\quad$ **if** $Q' > Q$ and $\mathit{overlap}(F') = 0$ **then**
9. $\quad\quad F = F'; Q = Q'$
10. $\quad$ **end if**
11. **end for**
12. **return** F

One source of redundancy comes from the possible presence of irrelevant terms in the equations. For example, given the two equations,

$$f_x(t) = 1.0 \cdot s_y(t)$$

and

$$f_x(t) = 1.0 \cdot s_y(t) + 0.00001 \cdot s_z(t)$$

the latter might produce a smaller error, but is worse in the sense that it includes s_z that does not contribute significantly to the modeling of s_x, and is likely to play a more important role when paired to another target sensor. In order to specify a preference for equations with relevant terms (such as the first example), we define a quality measure for equations that is based on the scalars c_y. Although defining such a measure can be done in various ways, including rather sophisticated statistical tests for the contribution of each term, we have opted for a more straightforward approach here. The following definition states how the quality of an equation depends on the scales of its parameters:

$$\mathit{quality}(f) = \frac{1}{\sum_{c \in f} |\log |c||}.$$

In other words, we prefer equations with scalars c closer to 1. We assume here that signals are all within farily similar domains, which is the case in our data. Furthermore, we prefer c values to be closer to 1 (i.e. $\log |c|$ close to 0), to indicate a direct dependence.

Given our set of candidate equations C, can we find a subset $F \subseteq C$ such that it leads to the highest total quality? As the quality cannot be negative, we can trivially select all equations to achive a maximum quality. However, this would provide a lot of redundancy, as many of these equations will describe the same relationship between sensors. Therefore, we restrict this subset such that a relationship between s_x and s_y is described at most once. That is, for each pair of sensors (s_x, s_y) there is at most one equation f_x or f_y such that:

$$f_x = c_y \cdot s_y(t) + \ldots, \text{ or}$$
$$f_y = c_x \cdot s_x(t) + \ldots$$

In order to derive an interesting set of equations, one option is to evaluate all suitable subsets of C, and select that one that has the highest quality. How would this scheme perform?

For each set of selected equations, F, we need to check the quality for each $f \in F$ that can be done linearly in the length of f: $\mathcal{O}(L(f))$. For the complete set, this becomes $\mathcal{O}(|F| \cdot |S|)$. Furthermore, we need to check for every pair $(f_1, f_2) \in F \times F$ if it does not overlap: $\mathcal{O}(|S|^2)$. In total, this thus becomes for each set: $\mathcal{O}(|F| \cdot |S| + |F|^2 \cdot |S|^2) = \mathcal{O}(|F|^2 \cdot |S|^2)$ for each selected set. Clearly, there are $\mathcal{P}$(C) number of all possible subsets of equations. The total therefore becomes $\mathcal{O}(\mathcal{P}(C) \cdot (|F|^2 \cdot |S|^2))$. Clearly, an exhaustive search is far from feasible. Typically, $\mathcal{P}(C)$ would be by far the biggest factor in this equation, making it the main target to minimise.

Our alternative therefore is to perform a heuristic search through the C search space. In this search, we utilise a forward selection scheme in which we evaluate the candidate equations in order and select only those that contribute to the total quality (see Algorithm 1).

In our approach, we order the set of candidate equations, denoted by $\downarrow C$, on length either:

- ascending: $f_1 > f_2 \leftrightarrow L(f_1) > L(f_2)$, or
- descending $f_1 < f_2 \leftrightarrow L(f_1) > L(f_2)$.

Each candidate equation is then added to F and checked whether this increases the overall quality. After all candidates are evaluated, the resulting F is returned.

Due to overlap, a new candidate equation might not be considered for inclusion in the resulting set while it might provide with a good quality increase. Alternatively, we can then apply a pruning strategy on F. That is, we can check for every candidate pattern whether some of the already selected equations can be removed from the set (see Algorithm 2).

Algorithm 2 ForwardSelectionWithPruning

FowardSelectionWithPruning(C)

1. $F = \varnothing; Q = 0$
2. **for all** $f \in\downarrow C$ **do**
3. $\quad F' = F \cup f; Q' = 0$
4. $\quad$ **for all** $f' \in F'$ **do**
5. $\quad\quad Q' = Q' + \mathit{quality(f)}$
6. $\quad$ **end for**
7. $\quad$ **if** $Q' > Q$ and $\mathit{overlap}(F') = 0$ **then**
8. $\quad\quad F = F'; Q = Q'$
9. $\quad$ **else**
10. $\quad\quad$ **for all** $f'' \in F' \setminus f'$ **do**
11. $\quad\quad\quad F'' = F' \setminus f''$
12. $\quad\quad\quad Q'' = 0$
13. $\quad\quad\quad$ **for all** $f''' \in F''$ **do**
14. $\quad\quad\quad\quad Q'' = Q'' + \mathit{quality(f)}$
15. $\quad\quad\quad$ **end for**
16. $\quad\quad\quad$ **if** $Q'' > Q$ **then**
17. $\quad\quad\quad\quad F = F''; Q = Q''$
18. $\quad\quad\quad$ **end if**
19. $\quad\quad$ **end for**
20. $\quad$ **end if**
21. **end for**
22. **return** F

Table 1: Characteristics of the 145 used sensors.

sensor type	$\#sensors$	location X-axis
1: geophone Z-axis	34	$\{1, 4\}$
2b: strain X-axis embedded	16	$\{6, 7\}$
2p: strain X-axis attached	34	$\{0, 2, 3, 4, 5\}$
3b: strain Y-axis embedded	28	$\{6, 7\}$
3p: strain Y-axis attached	13	$\{3, 5\}$
4b: temperature embedded	10	$\{7\}$
4p: temperature attached	10	$\{5\}$

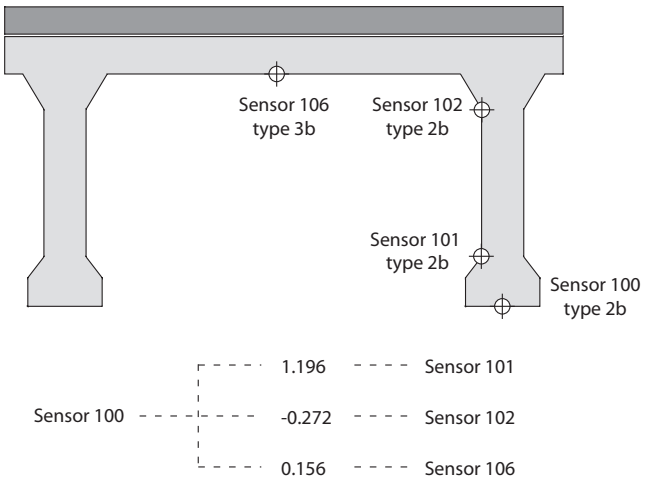

Figure 3: An example equation set shown in situ of the Hollandse Brug, please refer to Table 1 for the sensor type description.

4. EXPERIMENTS

In our experiments, we have used sensor measurement data derived from the *Hollandse Brug* in the Netherlands, as part of the InfraWatch project. This setup consist of 145 sensors. As can be seen in Table 1, there are 7 basic sensor types for which one a priori can expect that members to behave similarly. Therefore, in our first experiments, we have selected one target sensors from each of the 7 types.

In our preliminary experiments, we have aquired data representing 5 minutes of measurements, taken on the 24th of October in 2008, of all 145 sensors at 100 Hz, leading to 50 Mb of data. In our first experiments, we have downsampled this file with averaging to 1Hz, resulting in 297 distinct records.

Given each distinct target sensor, we use this dataset and Lagramge to fit equations on the target sensor. An equation grammar was used that produces linear functions of the form $f_x(t) = c_0 + \sum c_y \cdot s_y(t)$, as discussed. Note that this grammar can be easily upgraded to more complex, higher-order equations, were one so inclined. The amount of data available, and the expected nature of relationships between sensors plays a role in this decision also.

In our experiments, we have set the maximal prediction error to $1.0 \cdot 10^{-5}$. Unless reported otherwise, we have limited the search depth to 6, that is, at most 5 sensors and one constant c_0 can appear in the equation. Using a heuristic sum squared error-based beam search, we then obtain the 1000 best equations for each sensor type. In total, we therefore have 7000 candidate equations. In Table 2, examples are shown of the kinds of equation sets discovered, for two sensors: s_{100} and s_{301}. Also reported for each equation is the sum squared error that is obtained when approximating the target sensor.

4.1 Regular Equations

The candidate set of equations is ordered at the start of our forward selection algorithm. In our experiments we applied an order based on the length of the equations, either ascending or descending. With this forward selection scheme, we obtain much smaller equation sets out of the original candidate set. That is, sets with either 193 or just 1 equation, respectively, and with respect to the candidate set, we obtain reduction ratios of 2.8% and 0.014%. While these reductions seem very good, we should also focus on the resulting quality of the sets. In order to make this assesment, we take a look at the total quality, the sum of all quality values, of the resulting equation set. We see that we obtain a total quality of $48.1 \cdot 10^3$ and 105, for the ascending and descending order repectively. Moreover, we can have a look at how much of the sensor system is covered by these equation sets, which is 174 sensor pairs and 5 sensor pairs respectively.

However, such a forward selection algorithm with an overlap restriction is likely to select equations early on in the search process that might conflict with better candidates later on. Therefore, we have applied the pruning strategy for both candidate orders to observe its results.

As for the quality, we see that pruning leads to much better results in both the ascending and descending case. In Figure 5 (left), we depict the increase of quality during the run of the algorithm. As more candidates are being evaluated, we see that some of them can be added successfully to the equation set to increase its quality. Both orders show a similar increase in quality, although the descending approach leads to a slightly higher quality. The maximum obtained qualities are $1.57 \cdot 10^6$ and $1.72 \cdot 10^6$, for ascending and descending respectively.

We see the effect of pruning more clearly when looking at how the *cover* behaves over time (Figure 5 (right)). By cover, we mean the total number of unique sensors appearing in the right-hand side of the selected equations. Depicted for both candidate orders, we see how pruning affects the cover in a non-monotonic manner in favour of increasing the quality of the complete set of equations. In this respect, the descending-ordered candidate set gradually leads to larger covers of the sensor system, while the ascending order seems to peak early on in the search process. This eventually results in a cover of 40 and 154 for ascending and descending respectively.

We depict an example equation set in Figure 3. This result is obtained when using pruning on the descending-ordered candidate set as described earlier on. We show the sensors that are related to sensor 100, namely 101, 102, and 106:

$$f_{100}(t) = 23.30 + 1.196 \cdot s_{101}(t) - 0.272 \cdot s_{102}(t) + 0.156 \cdot s_{106}(t)$$

Note that all selected sensors fall within the same segment of the complete bridge (a total of 7 segments). Furthermore, all sensors are of the embedded type, which indicates that

Table 2: Examples of the first 5 equations found by Lagramge for sensors 100 and 220.

$$f_{100} = -4.799 + 1.16 \cdot s_{101} - 0.2364 \cdot s_{102} + 0.1773 \cdot s_{104} + 0.1104 \cdot s_{108} - 268.8 \cdot s_{233} \quad (SSE = 0.1084)$$
$$f_{100} = 16.98 + 1.143 \cdot s_{101} - 0.2896 \cdot s_{102} + 0.1347 \cdot s_{104} + 0.1221 \cdot s_{106} + 166.0 \cdot s_{223} \quad (SSE = 0.1086)$$
$$f_{100} = 16.70 + 1.14 \cdot s_{101} - 0.293 \cdot s_{102} + 0.1335 \cdot s_{104} + 0.1204 \cdot s_{106} + 128.8 \cdot s_{207} \quad (SSE = 0.109)$$
$$f_{100} = 11.8 + 1.150 \cdot s_{101} - 0.3028 \cdot s_{102} + 0.1469 \cdot s_{104} + 0.09248 \cdot s_{106} - 84.352 \cdot s_{219} \quad (SSE = 0.1092)$$
$$f_{100} = -4.812 + 1.152 \cdot s_{101} - 0.2431 \cdot s_{102} + 0.1916 \cdot s_{104} + 0.1069 \cdot s_{108} - 55.15 \cdot s_{201} \quad (SSE = 0.1094)$$
...
$$f_{301} = 2.215 - 4.291 \cdot 10^{-6} \cdot s_{105} + 3.502 \cdot s_{235} + 0.7977 \cdot s_{317} \quad (SSE = 1.888 \cdot 10^{-5})$$
$$f_{301} = 2.275 - 4.215 \cdot 10^{-6} \cdot s_{105} + 4.795 \cdot s_{236} + 0.7916 \cdot s_{317} \quad (SSE = 1.892 \cdot 10^{-5})$$
$$f_{301} = 2.251 - 4.206 \cdot 10^{-6} \cdot s_{105} + 3.273 \cdot s_{241} + 0.7939 \cdot s_{317} \quad (SSE = 1.895 \cdot 10^{-5})$$
$$f_{301} = 2.289 - 4.199 \cdot 10^{-6} \cdot s_{105} + 2.09 \cdot s_{233} + 0.7902 \cdot s_{317} \quad (SSE = 1.901 \cdot 10^{-5})$$
$$f_{301} = 2.328 - 4.2240 \cdot 10^{-6} \cdot s_{105} - 0.8519 \cdot s_{224} + 0.7865 \cdot s_{317} \quad (SSE = 1.902 \cdot 10^{-5})$$
...

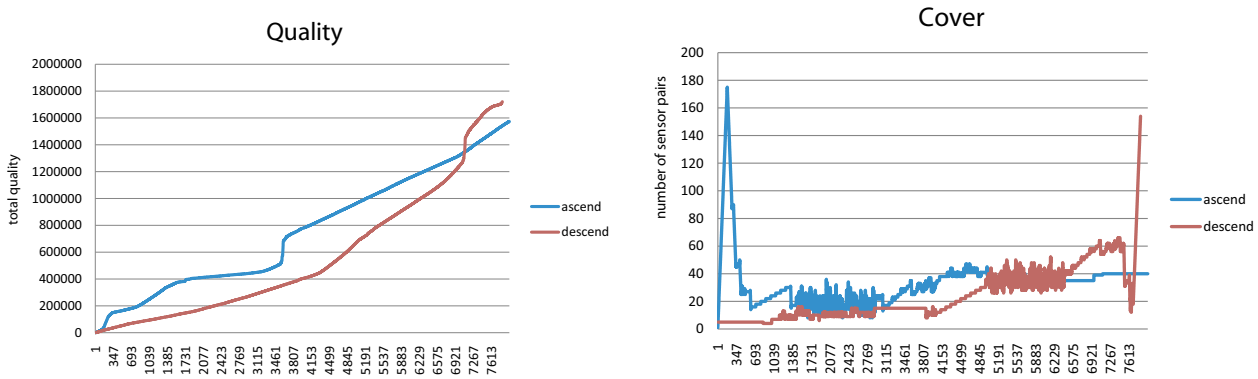

Figure 5: The increase of the quality (left) and the cover (right) of the equation set, both as a function of the evaluated candidate equations when using pruning.

this type of signal differs from the attached type in terms of physical characteristics.

When looking at the original signals, we see that sensor 102 is indeed an inverted signal opposed to sensor 100, and that 101 is quite similar to 100, and 106 is indeed also positively correlated. All signals show peaks occuring at similar times, which are our events of interest (see Figure 4). In this setup, we would select sensor 100 as a prototype to demonstrate which events are occurring in the sensor system.

4.2 Differential Equations

Apart from the regular algebraic grammar used in the previous section, we can use more elaborate grammars in order to fit more complex equations on the data. For example, we can use differential equations to approximate the target signal. This can actually be a very good way to model time dependent aspects of the sensor system, as it includes the temporal aspect more directly in the equations. To this end, we have also used the above sensor data to fit the following type of equation. Again, we have opted for relatively simple, though differential, models of the data:

$$\frac{df_x}{dt} = c_0 + \sum c_y \cdot s_y(t)$$

Like before, we have used a beam search with the same parameters to find the 1000 best equations that minimise the prediction error for the target sensor. The first five differential equations found are shown in Table 3. For equation selection on the differential equations, we see similar results as for the linear case. For ascending and descending, the forward selection without pruning leads to a (relatively small) quality of 983.2 and 27.6, a cover of 36 and 6, and an equation subset size of 35 and 2, respectively. As a demonstration, the two equations in this last subset are as follows:

$$\frac{df_{100}}{dt} = 128.7 - 0.40 \cdot s_{162}(t) + 5551 \cdot s_{240}(t) - 13.7 \cdot s_{318}(t)$$
$$\frac{df_{100}}{dt} = 32.8 + 0.12 \cdot s_{116}(t) - 1316 \cdot s_{209}(t) - 3.34 \cdot s_{317}(t)$$

Again, we see much better results when applying the pruning strategy on the candidate equation set. For ascending and descending, the forward selection with pruning leads to a quality of $7.57 \cdot 10^5$ and $1.39 \cdot 10^6$, a cover of 26 and 56, and a equation subset size of 10 and 45, respectively.

From this we can conclude that in terms of the number of selected equations in both cases we obtain very good reduction ratios, 0.5% and 0.02% without pruning, and 0.14% and 0.64% with pruning.

4.3 Select Target Sensors

In the previous experiments, we have used a set of 7 sensors, one for each type, as targets for which equations were

Table 3: Examples of the first 5 differential equations found by Lagramge for sensor 100.

$$\frac{df_{100}(t)}{dt} = 12.95 + 0.1115 \cdot s_{116}(t) - 1374.8 \cdot s_{209}(t) - 1.6136 \cdot s_{318}(t) \quad (SSE = 9.418)$$
$$\frac{df_{100}(t)}{dt} = -1.540 + 0.1076 \cdot s_{116}(t) - 1260.9 \cdot s_{209}(t) \quad (SSE = 9.43133)$$
$$\frac{df_{100}(t)}{dt} = 27.61 + 0.1614 \cdot s_{117}(t) - 1147.1 \cdot s_{209}(t) - 2.7610 \cdot s_{317}(t) \quad (SSE = 9.442)$$
$$\frac{df_{100}(t)}{dt} = 27.64 + 0.1682 \cdot s_{117}(t) - 1359.4 \cdot s_{209}(t) - 3.168 \cdot s_{318}(t) \quad (SSE = 9.483)$$
$$\frac{df_{100}(t)}{dt} = 16.63 + 0.1458 \cdot s_{117}(t) - 1099.2 \cdot s_{209}(t) - 1.787 \cdot s_{319}(t) \quad (SSE = 9.50)$$
...

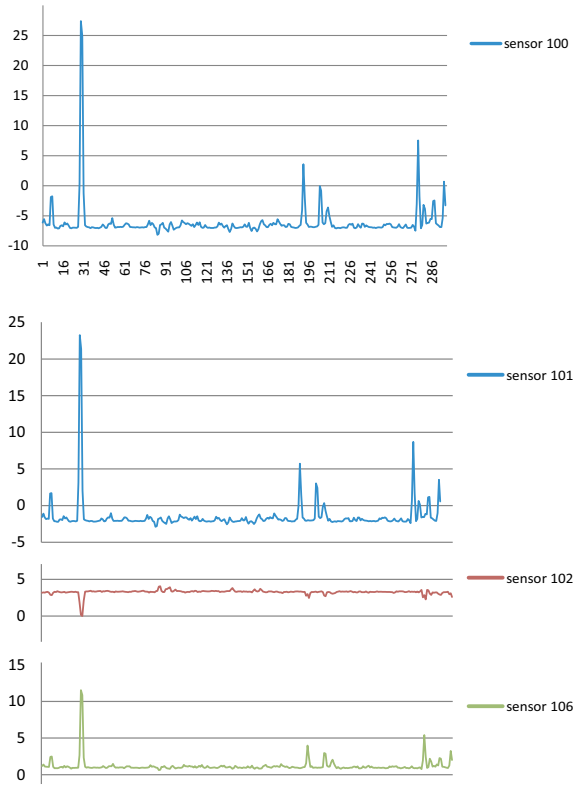

Figure 4: The sensor measurements for the signals that correspond to the example shown in Figure 3.

fitted. However, it could be that this background knowledge steers the search process significantly in a biased direction. Therefore, we have derived a set of candidate equations for each sensor in the system as target.

Moreover, we have reduced the search depth of Lagramge such that exactly one sensor can be paired with one target sensor. This allows the search space to be kept reasonable, while it still leads to over 20 thousand candidate equations. In addition, having only one sensor in the equation shows more clearly the direct relation between the sensors.

For this, we have used the best performing variant of our algorithm: with pruning and a descending-ordered candidate set. This resulted in an equation set of size 223. As for the target sensors, a set of 26 distinct sensors were selected as representations for the complete system. The resulting equation set had a quality of $3.74 \cdot 10^7$ and covered 196 pairs of the sensor system.

How does this relate to the case of hand-picking a set of target sensors? When selecting a set of target sensors from the complete sensor space, we see that not all sensor types are represented as targets. If we break these down to sensor type, we get the following distribution:

type	description	targets	equations
1	geophone Z-axis	8	12
2b	strain X-axis embedded	0	0
2p	strain X-axis attached	1	1
3b	strain Y-axis embedded	1	1
3p	strain Y-axis attached	0	0
4b	temperature embedded	6	14
4p	temperature attached	10	195

When visually inspecting the sensor signals, we see that those sensor types that are not so well described by other sensors tend to have more prototypes in the resulting set. For example, when inspecting the signals corresponding to sensor type 1, the geophone sensors, we see that all chosen target sensors of this type fluctuate quite a lot (see Figure 6).

5. CONCLUSION

Increasingly, physical systems are being equipped with sensors that in some form monitor its behaviour. In this paper, we focus on one such system in particular, the *Hollandse Brug*, which in the context of the *InfraWatch* project has been equipped with a multitude of sensors that measure aspects such as vibrations and strain. The aim of the project is to use the derived stream of sensor data to find interesting patterns and features that can be considered characteristic for its short and long-term behaviour.

In this paper, the focus was on finding a small set of interesting sensors within the large set of available sensors. As monitoring the complete system at once is very hard, we have proposed to model dependencies between sensors, and find characteristic sensors that can serve as a prototype for other sensors. The monitoring and exploratory analysis of the data can then be restricted to this subset of sensors. The prototype sensors should ideally contain the same features (peaks, response to traffic, etc.) as all the other sensors they represent.

Dealing with numeric data is a hard task for pattern mining techniques. However, we propose to shift the context slightly. Instead of grouping sensors based on their discrete events, we propose to group those sensors that can

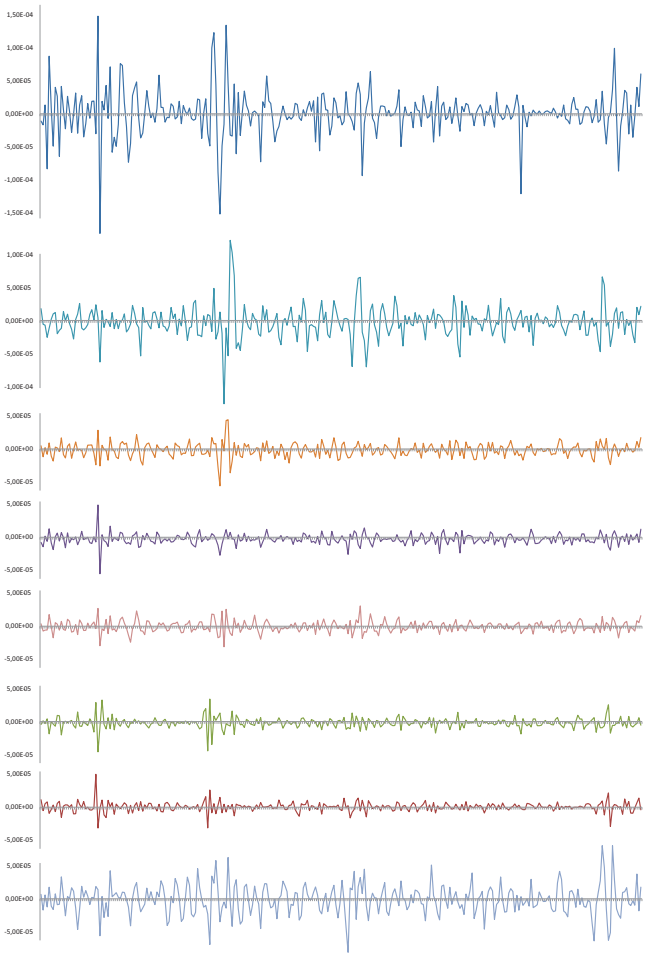

Figure 6: The signal values of some of the geophone sensors, that are hard to group. As expected, in contrast to other sensors, we see that the events are not as clearly present in the signals

be used well to describe other sensors in the form of equations. Although many specific implementations can be used, our inital results focus on linear and first-order differential equations. In these first experiments, we see that adjacent sensors, those that are likely to measure similar events, are indeed grouped by our approach.

When looking at the signal behaviour within the group we see that distinct peaks, which indicate distinct traffic loads, occur at similar time points, and stand out clearly from the noise in the signal.

Acknowledgements

The InfraWatch project is funded by the Dutch funding agency STW, under project number 10970. Access to sensor data of the Hollandse Brug is kindly provided by Strukton b.v.. Specifically Bas Obladen, Carlos Bosma and Hessel Galenkamp from Strukton were instrumental in setting up the sensors and data acquisition facilities, as well as arranging access to the data.

6. REFERENCES

[1] Bringmann, B., Zimmermann, A. *The Chosen Few: On Identifying Valuable Patterns*, In Proceedings ICDM 2007.

[2] M. Dejori, H.H. Malik, F. Moerchen, N.C. Tas, and C. Neubauer, 2009 *Development of Data Infrastructure for the Long Term Bridge Performance Program*, In Proceedings of Structures '09, Austin, USA.

[3] E. Doupal, R. Calderara, 2004, *Weigh-In-Motion*, In Proceedings of First International Conference on Virtual and Remote Weigh Stations, Orlando.

[4] Džeroski, S. and Todorovski, L. (1993) Discovering dynamics. In Proc. Tenth International Conference on Machine Learning, pages 97-103. Morgan Kaufmann, San Mateo, CA.

[5] Džeroski, S. and Todorovski, L. (1995) Discovering dynamics: from inductive logic programming to machine discovery. Journal of Intelligent Information Systems, 4: 89-108.

[6] Knobbe, A., Blockeel, H., Koopman, A., Calders, T., Obladen, B., Bosma, C., Galenkamp, H., Koenders, E., Kok, J. *InfraWatch: Data management of large systems for monitoring infrastructural performance*, In Proceedings IDA 2010, Tucson, USA, 2010

[7] Knobbe, A. *Multi-Relational Data Mining*. IOS Press, Amsterdam, 2006. http://www.kiminkii.com/thesis.pdf.

[8] Knobbe, A., Ho, E.K.Y. *Maximally-Informative k-Itemsets, and their Efficient Discovery*, In Proceedings KDD 2006, 2006.

[9] A. Knobbe, 1997, *Data Mining for Adaptive System Management*, In Proceedings of PAKDD '97, London.

[10] A. Knobbe, Bart Marseille, Otto Moerbeek, Daniël M.G. van der Wallen, *Results in Adaptive System Management*, Benelearn'98

[11] G. Meijer, *Smart Sensor Systems*, 2008, ISBN: 978-0-470-86691-7, Hardcover, 404 pages.

[12] Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T. *Numerical Recipes.* Cambridge University Press, Cambridge, MA, 1986.

[13] Todorovski, L. and Džeroski, S. (1997) Declarative bias in equation discovery. In Proc. Fourteenth International Conference on Machine Learning, pages 376-384. Morgan Kaufmann, San Mateo, CA.