DOI: 10.1111/cgf.14926

COMPUTER GRAPHICS *forum*
Volume 42 (2023), number 6, e14926

Episodes and Topics in Multivariate Temporal Data

Natalia Andrienko,[1,2] Gennady Andrienko[1,2] and Gota Shirato[1,3]

[1]Fraunhofer Institute IAIS, Sankt Augustin, Germany
gennady.andrienko@iais.fraunhofer.de
[2]City University of London, London, UK
[3]University of Bonn, Bonn, Germany

Abstract

The term 'episode' refers to a time interval in the development of a dynamic process or behaviour of an entity. Episode-based data consist of a set of episodes that are described using time series of multiple attribute values. Our research problem involves analysing episode-based data in order to understand the distribution of multi-attribute dynamic characteristics across a set of episodes. To solve this problem, we applied an existing theoretical model and developed a general approach that involves incrementally increasing data abstraction. We instantiated this general approach in an analysis procedure in which the value variation of each attribute within an episode is represented by a combination of symbols treated as a 'word'. The variation of multiple attributes is thus represented by a combination of 'words' treated as a 'text'. In this way, the the set of episodes is transformed to a collection of text documents. Topic modelling techniques applied to this collection find groups of related (i.e. repeatedly co-occurring) 'words', which are called 'topics'. Given that the 'words' encode variation patterns of individual attributes, the 'topics' represent patterns of joint variation of multiple attributes. In the following steps, analysts interpret the topics and examine their distribution across all episodes using interactive visualizations. We test the effectiveness of the procedure by applying it to two types of episode-based data with distinct properties and introduce a range of generic and data type-specific visualization techniques that can support the interpretation and exploration of topic distribution.

Keywords: visualization, visual analytics, topic modeling

CCS Concepts Categories and Subject Descriptors (according to ACM CCS): [Human-centred computing → Visual analytics]; Visualization application domains–Visual analytics

1. Introduction

Everything in the world changes over time. Data describing changes often consist of time-referenced values of one or more attributes. The process or succession of changes along time can be divided into *episodes* each of which occurs over a specific time interval and is described by attribute values referring to different time steps within this interval. Figure 1 demonstrates one of possible ways to divide continuous time series of attribute values into episodes using a sliding time window. Our research aims at finding ways to help humans understand dynamic phenomena or behaviours by analysing episode-based data.

While a line graph or other visual representation of a time series of attribute values can help a person understand the overall character of the development and identify different patterns of change, it can be difficult to get a holistic understanding of what is happening when changes are characterized by multiple attributes. To address this problem, we aim to extract interpretable patterns of change for singular attributes and then derive meaningful patterns of their joint changes.

The approach we develop and test is based on explicit representation of single-attribute temporal patterns as elements of data. Given a collection of episodes, we represent the variation of values of each individual attribute in each episode by a combination of symbols, which is treated as a 'word'. Hence, the variation of all attributes within an episode is represented by a combination of such 'words', which can be treated as a 'text'. The entire set of episodes is thus transformed to a collection of 'texts'. In natural language processing, there are topic modelling methods [VK20, AEG*23] that extract interpretable groups of semantically related words based on their co-occurrence in texts. By analogy, we expect that applying topic modelling methods to the set of 'texts' derived from the

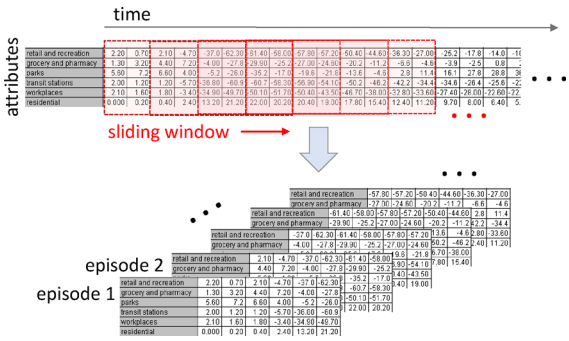

Figure 1: *Illustration of the division of a continuous multivariate time series into multiple episodes using a sliding time window.*

episode-based data will result in finding interpretable groups of related 'words' encoding single-attribute variation patterns that tend to occur together in episodes. If successful, these groups can be considered as integrated multi-attribute temporal patterns representing components of complex behaviours or stages of complex processes.

The next goal after the extraction of multi-attribute patterns is to understand when, where, and under what circumstances different patterns occur, which is essential for understanding the dynamic phenomenon or behaviour as a whole. To facilitate this understanding, we want to provide an overview and enable the exploration of the distribution of the patterns in context, including space, time, and any conditions that may affect or be affected by the process or behaviour being studied.

With this paper, we intend to make the following contribution to the visual analytics research dealing with temporal data:

- Propose a conceptual framework and a general workflow for the analysis of dynamic phenomena described by episode-based data involving time series of multiple attributes.
- Explore the opportunities for analysis of dynamic phenomena given by explicit representation of temporal patterns of attribute variation.
- Investigate the potential of using topic modelling techniques for revealing relationships between patterns and finding patterns of pattern co-occurrence.
- Demonstrate examples of visual exploration of pattern distribution for data of distinct nature.

We begin with introducing the conceptual background in Section 2 followed by an overview of the related work in Section 3. We describe the work of the investigated analysis approach in two case studies in Section 4 and then discuss our experiences and findings (Section 5). Section 6 concludes the paper.

2. Background

2.1. Key concepts

In this research, we use the term *'episode'* to refer to a short period of time that has distinct properties while being a part of a larger series or process. This aligns with common definitions of an episode in dictionaries (e.g. [MW22]). Our research focuses on data describing changes occurring within episodes with non-zero duration. The data consist of series of attribute values referring to different times between the beginnings and ends of the episodes. While the attributes can be of any type (numeric, categorical, or spatial), our current research is focusing on attributes with numeric values. We refer to this type of data as *episode-based data*.

Our research goal is to find methods and develop a visual analytics workflow for analysing episode-based data in order to understand the overall process or behaviour the episodes are parts of. Gaining a general understanding of a whole by observing its multiple parts requires abstraction. According to the theoretical model known as 'pattern theory' [AAM*21], abstraction in data analysis is achieved by finding *patterns* in data distributions. A pattern is a combination of relationships between multiple data items that allows us to consider and represent all of these items jointly as a unit, which can be described without referring to any individual items.

Episode-based data have a hierarchical structure. The entire dataset consists of descriptions of multiple episodes, each of which is composed of time series of multiple attributes. Each time series consists of multiple attribute values and their corresponding time references. To gain a general understanding, we must perform abstraction from the elementary data items (attribute values and time references) up to patterns at the highest level of the hierarchy (the distribution of dynamic properties across the entire set of episodes). We believe this requires a step-wise ascent from the bottom to the top of the hierarchy. At the lowest level, patterns are made up of data elements; at higher levels, patterns are made up of patterns from the previous levels.

At the lowest level, patterns are jointly formed by (1) temporal relations between the time steps, (2) correspondences between time steps and attribute values, and (3) relations between the attribute values. The possible types of patterns include increase, decrease, peak, trough, constancy, and fluctuations.

At the second level of the hierarchy, patterns are formed by relations of co-occurrence of temporal patterns of individual attributes, that is the single-attribute patterns appear together in episodes. For example, increase of attribute A1 may tend to co-occur with constancy of attribute A2 and decrease of attribute A3.

At the third level of the hierarchy, one needs to consider the distribution of the second-level patterns over relevant data dimensions, one of which is always time (since the data are temporal). Third-level patterns of the temporal distribution are formed by the temporal relations between the positions of the second-level patterns in time. Other relevant dimensions depend on the nature of the phenomenon reflected in the data. The data describing episodes may include contextual information about the circumstances in which the episodes occurred, their spatial positions, and/or the actors involved. The set of all such contexts is the relevant data dimension. The distribution of the second-level patterns with respect to the set of contexts needs to be analysed. Third-level patterns are formed by the links of the second-level patterns to various context properties. For example, there may be tendencies for the combination patterns to occur or not occur in certain parts of space or under specific external conditions.

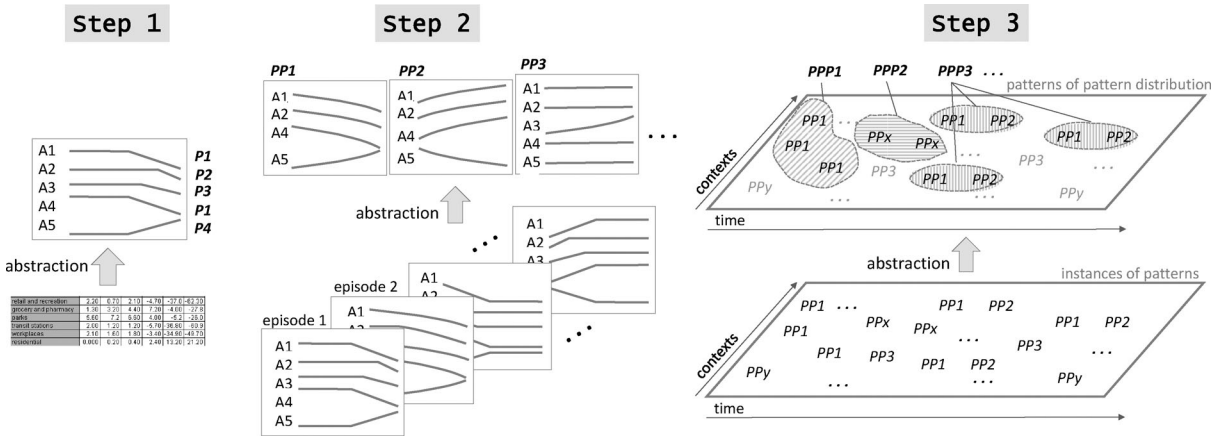

Figure 2: *A schematic representation of the idea of progressive abstraction. Step 1: Sequences of values of individual attributes are abstracted to single-attribute patterns of value variation. Step 2: Combinations of single-attribute patterns co-occurring in multiple episodes are abstracted to multi-attribute combination patterns, Step 3: Instances of combination patterns occurring throughout the data are abstracted to patterns of distribution of the combination patterns.*

This theory-based analysis workflow is schematically represented in Figure 2. In the first step of the workflow, the temporal sequences of attribute values in each episode are abstracted to single-attribute patterns $P1$, $P2$, … representing the character of the value variation. In the second step, the combinations of the single-attribute patterns co-occurring in the episodes are analysed to find multi-attribute combination patterns denoted $PP1$, $PP2$, …. The notation PP emphasizes that the combination patterns are super-patterns (i.e. patterns of a higher level of abstraction) with respect to the single-attribute patterns. After extracting the set of PP_i, the episodes are represented in terms of combinations of these super-patterns. In the third step, the distribution of the super-patterns PP_i over the set of episodes is analysed to find super-super-patterns PPP_j formed by relationships between the super-patterns resulting from their positions in the distribution. The distribution is schematically represented in Figure 2, right, as a plane where one dimension is time and the other stands for the set of relevant contexts.

Figure 2 represents the general idea of the progressive abstraction approach. It does not specify what methods can be utilized to fulfil the three steps of the workflow. In the following, we describe one of many possible ways to implement the approach. In Step 1, single-attribute value variations are represented by SAX patterns [LKWL07]. In Step 2, multi-attribute combination patterns are extracted by means of topic modelling methods [VK20]. In Step 3, patterns of the distribution of the combination patterns are discovered with the help of interactive visualizations.

2.2. Ideas for implementing the approach

According to pattern theory, one possible operation on discovered data patterns is to represent them in an aggregated manner so that they can be treated as single elements of data. This means that the combinations of data elements making up the patterns are replaced by aggregated representations, which can then be used in further analysis.

The pattern aggregation operation may be a part of an approach to finding multi-attribute combination patterns. After identifying single-attribute patterns, we can treat them as units and represent them with tokens. We can then replace the original time series of attribute values with tokens denoting the temporal patterns formed by these values. As a result, each episode is represented by a combination of pattern tokens. We can then apply a method that is suitable for analysing combinations of tokens and can find patterns formed by the tokens, such as recurring associations. A potentially suitable class of methods is topic modelling. The topics generated by these methods are, in essence, multi-token patterns, which in our case can be interpreted as multi-attribute temporal variation patterns.

Once we have an aggregated representation of the multi-attribute combination patterns in the form of topics, we can use them in further analysis. We need to analyse the distribution of the integrated patterns (topics) across the dataset to find patterns at an even higher level of abstraction and establish relationships between them. To support this by visual analytics techniques, we need to find appropriate methods to visualize the distribution of the topics over the set of episodes.

2.3. Deriving single-attribute patterns

A temporal pattern of attribute values represents, in an aggregated form, the relationships between values arranged in a chronological sequence. These relationships may be *similar* or *different*, *larger* or *smaller*, *close* or *distant*, etc. To efficiently implement our workflow, we need a method for automatically transforming value sequences into aggregated representations of the patterns of value variation along the sequences. Essentially, we need a compact and simplified machine-readable representation of a time series that can be treated as a single object (a token) in the following steps of analysis. This means that the representation must be symbolic rather than numeric.

For time series of numeric values, there is a suitable representation called Symbolic Aggregate approXimation (SAX) [LKWL07].

SAX divides each time series into a specified number w of equal-sized segments and calculates the mean value in each segment. The probability or frequency distribution of the mean values is divided into α equiprobable parts, where α is the desired size of the alphabet, that is the number of symbols to be used for encoding the time series. Each part of the distribution is given a distinct symbol from the alphabet. The mean values of the time series segments are mapped to the symbols corresponding to the parts of the value distribution in which they fit. As a result, each time series is represented by a sequence consisting of w symbols from the alphabet.

The basic idea of the SAX representation method can also be applied to values of other types of attributes if it is possible to divide the value distribution into a small number of meaningful parts that can be represented by symbols from an alphabet. For example, when attribute values represent positions in space, the space can be divided into regions, the mean positions in segments of the time series can be calculated, and these positions can be encoded with symbols corresponding to the regions containing them. This allows for the creation of a compact and simplified machine-readable representation of a time series that can be treated as a single object in the following steps of analysis.

3. Related Work

Related to our work are researches in the following areas: visual exploration of multivariate time series (MVTS) and event sequences, segmentation of MVTS, simplification of numeric time series, abstraction of temporal data, and application of topic modelling methods to non-textual data.

3.1. Visual exploration of multivariate time series

The most obvious approach to visualization of MVTS is representation of the time series of the individual variables along a common time axis in a juxtaposed, superposed, stacked, or intertwined way [JME10, BHR*19]. Another widely used approach is to apply dimensionality reduction (DR) to the combinations of values of the variables and represent the time steps by points in a two-dimensional projection space, as, for example, in MotionTrack [HWX*10] or TimeCurves [BSH*16]. Bernard et al. [BWS*12] assign colours to the positions in the projection space and represent the temporal variation of the value combinations by variation of colours along the time axis.

Fujiwara et al. [FSS*21] deal with data consisting of multiple MVTS, such as measurements recorded in different geographic locations. Originally, the data have the form of a 3D tensor. It is transformed to a matrix where each row corresponds to one time step of one MVTS. After applying DR to this matrix and obtaining a 2D projection, selected clusters of points in the projection plot are represented by colour coding in various additional views supporting interpretation of the DR results. To analyse multiple MVTS of air pollution data from different locations, Kuo et al. [KFC*22] apply non-negative matrix factorization (NMF) as a DR technique. NMF extracts combinations of chemicals that can be attributed to different sources of air pollution. The visual displays are designed to enable interpreting the results of NMF.

Algorithmic clustering of time steps with subsequent representation of the clusters by colours is also used in analysis of MVTS [GCML06]. DR can be applied to clustered and aggregated data [BWK*13].

It is worth noting that DR or clustering methods in all these works are applied to data associated with individual time steps, that is with time points lacking duration. Hence, each data item includes a single value of each attribute. Our approach to analysis of multivariate temporal data is based on dividing the data into episodes of non-zero duration. Each episode includes a sequence of values of each attribute. Analysis of such data requires different approaches.

In essence, episodes are *events*; hence, episode-based data comprise one or more sequences of events. Analysis of event sequences is an established research topic in visual analytics. Guo et al. [GGJ*22] present a comprehensive survey of the existing approaches and systems. Well known examples include LifeFlow [WGGP*11], OutFlow [WG12], and EventFlow [MLL*13]. In all these works, events are considered as atomic objects without internal structure. The main focus of analysis is arrangement of events relative to each other, which is different from our focus on the variation of values of multiple attributes *within the episodes*. Eventpad [CvW18] is designed for analysis of sequences of multivariate events, where multiple attributes characterize each event as a whole. Differently from our work, dynamic attributes whose values vary during the event life times are not considered.

3.2. Segmentation of multivariate time series

Episodes can be obtained from time series data in many different ways. The simplest approach is to use a sliding time window that defines episodes of equal length (e.g. [STKF07]). The episodes may partly overlap in time [WG11] thus smoothing transitions between consecutive patterns of value variation. This method of deriving episodes is illustrated in Figure 1. The resulting pieces of time series can be treated as multidimensional vectors to which clustering and/or dimensionality reduction (projection) methods can be applied [vWvS99, STKF07, WG11]. Other approaches define episodes based on events, for example by taking temporal buffers before and/or after detected events or time intervals from one event to another; see Monroe et al. [MLL*13] for examples in basketball data analysis. TimeMask technique [AAC*17] proposes a powerful set of query operations for defining episodes. In this paper, we apply the sliding window approach in Section 4.1 and event-based definition of episodes in Section 4.2.

Episodes can also be obtained by dividing time series into semantically meaningful segments. There exist segmentation algorithms [GYD*19], which can be combined with interactive visual techniques [BDB*16, BBB*18]. Segmentation may also be done based on clustering of time steps [BWK*13]. An earlier work [AA23] considers the problem of dividing the time span of a complex dynamic phenomenon described by *multiple MVTS* into meaningful periods that enclose different relatively stable states or development trends. Here, the segmentation is applied to all MVTS taken together. The

task is supported by a combination of clustering, aggregation, projection, and interactive visual tools for time division.

Statistical science develops methods for detecting multiple change points in a long univariate time series [NHZ16]. To utilize such methods for dividing MVTS, it is necessary to define the way of setting common breaks for all time series so as to take into account their individual change points.

3.3. Simplification of numeric time series

In our approach, short time series encapsulated in episodes are simplified and represented in an abstracted symbolic form using the SAX pattern method [LKWL07]. The method involves aggregation (averaging) of attribute values by sub-intervals and discretization of the domain of the aggregated attribute values. Alternatively to aggregation, simplification of time series can be achieved by downsampling [CS10, Ste13], which represents a time series using a smaller number of time points while striving to preserve its shape. The Douglas-Peucker algorithm originally proposed for cartographic generalization [DP73] can be used for the same purpose.

To discretize a numeric attribute, its value range is divided into bins by introducing several breaks. The ways to do this have been studied extensively in cartography (see a review by Slocum [SMKH22]) for designing classified choropleth maps. The most common approaches include natural breaks, equal length intervals, and equal size divisions. Jenks [Jen77] developed a method for calculating a statistically optimal classification. The geovisualization research community developed various interactive procedures for human-controlled discretization [AAK*21, SMKH22]. A number of discretization techniques have been developed in data mining, see a review by Garcia et al. [GLS*13]. Beyond discretization, an extensive survey of methods for simplification and compact representation of numeric time series that can be used in visualization was done by Shurkhovetskyy et al. [SAAF18].

The authors of the SAX method [LKWL07] noted that symbolic representation of time series had not received much attention in the data mining research. More recently, Bondu et al. [BBC16] proposed a more advanced variant of the SAX method that optimizes the division of the time series into sub-intervals. Shirato et al. [SAA21] divide episodes into equal intervals, as in the SAX method, but apply symbolic encoding to value trends (increase, decrease, or constancy) on the intervals rather than value aggregates.

Instead of directly working with time series as sequences of values, it may be suitable for many analysis tasks to extract features from them, that is derive attributes characterizing some aspects of the entire time series. Lubba et al. [LSK*19] evaluated about 5000 diverse features that can be computed from time series and selected a subset of 22 features that exhibit strong classification performance across a given collection of time-series problems and are minimally redundant. There are multiple software libraries for feature extraction, for example [BFF*20]. A recent trend is automatic extraction of time series features by means of artificial neural networks [ZZYG21, CTMMB22]. While such features may work very well in machine learning tasks, they are not interpretable by humans. In contrast, our goal is to extract variation patterns that are meaningful to humans.

3.4. Abstraction of temporal data

While the term 'abstraction' is often treated as a synonym to 'simplification' or 'compression', we use this term in the sense of transforming data into meaningful higher-level concepts [Sha97]. According to Aigner et al. [AMM*08, AMST11], data abstractions are qualitative values or patterns that convey key ideas. Abstraction of temporal data may be supervised or unsupervised [HÖ2]. The former means finding time intervals where data have a priori defined properties and assigning corresponding labels from a given set. Thus, there are methods for detecting predefined shapes, such as increase, decrease, constancy, peak, etc. [HÖ2, SAA23]. Abstraction can be done using domain-specific rules [CC99, AMM*08] or definitions from a domain ontology [Sha97]. Unsupervised time abstraction means that concepts are not defined in advance but need to be learned from data. A general approach is to identify similar parts in the time series by means of clustering [vWvS99, HÖ2]. In our work, unsupervised abstraction is supported by visual representations of time series transformed to SAX patterns.

Progressive abstraction means deriving higher-level concepts from earlier extracted patterns based on relationships between the latter [AAM*21]. Complex patterns can be detected automatically according to predefined rules specifying patterns to be used as building blocks and temporal relationships between these blocks [SLCB07]. Shirato et al. [SAA23] propose interactive visual interfaces to identify patterns in the temporal distribution and co-occurrences of patterns of different attributes and transitions between patterns of the same attributes. In that work, higher-level patterns are constructed in the mind of the analyst. In our current work, the derivation of multi-attribute patterns from single-attribute patterns is supported by topic modelling techniques.

3.5. Applications of topic modelling beyond text analysis

Regardless of the encoding method, the output of the procedure is a set of tokens (words) that represent single-attribute patterns. Each episode is described by a combination of tokens for multiple attributes. To find recurrent associations of tokens, we apply topic modelling methods [VK20], such as Latent Dirichlet allocation (LDA) [BNJ03] and Non-negative Matrix Factorization (NMF) [LNC*17]. In essence, these methods perform dimensionality reduction: similarly to PCA [JC16], they generate higher level features (considered as 'topics') as weighted combinations of input features, which may be, in particular, frequencies or measures of relevance of different words. Beyond analysing texts in natural languages, topic modelling methods have been successfully applied to different types of data, for example, DNA codes [LTD*16], software repositories [CTH16], taxi trips [CSZ*14], user activities in interaction with software systems [CAA*20], and team tactics in football [SAA21]. These examples demonstrate the versatility of the topic modelling methods. However, we are not aware of the uses of topic modelling for finding patterns in multivariate time series.

While there have been numerous comparative evaluation studies of different topic modelling methods in application to text data (e.g. [AYB20, EY22, AEG*23], we are not aware about studies comparing effectiveness of these algorithms for other data types. In our work, we did not plan to compare different methods. However,

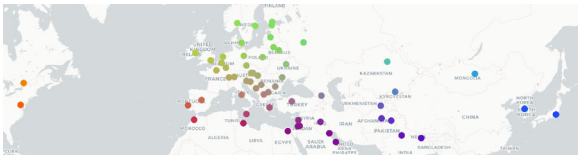

Figure 3: *The countries involved in the case study are represented on a map by their capitals.*

LDA, which worked quite well in the first case study, produced unsatisfactory results in the second case study. This lead us to try NMF, which fulfilled its task successfully.

The two case studies in which we tested our ideas and analysis workflow are described in the following section.

4. Implementation and Testing of the Approach

In the first case study, we deal with data on population mobility behaviours in different countries during the COVID-19 pandemic. In the second case study, we apply our approach to episodes extracted from football game tracking data.

4.1. Case study 1: COVID-19 mobility trends

We use a subset of the publicly available data set of mobility trends available in the Google's COVID-19 Open Data Repository [Goo22]. Google collects anonymized data from apps such as Google Maps to track changes in people's movements during the pandemic. The data consist of daily visitor numbers to specific categories of places (e.g. grocery stores, parks, train stations, etc.) relative to baseline days before the pandemic outbreak. Baseline days represent a normal value for each day of the week and are given as the median value over the 5-week period from 3 January to 6 February 2020. The rationale for using a specific normal value for each day of the week is that people usually have different routines on weekends versus weekdays. The data thus consist of the deviations from the normal values expressed in percent of the normal values. Positive values signify increased numbers of visits to a certain category of places and negative values have the opposite meaning.

For this case study, we used a subset of data containing information on 60 countries in Europe, Asia, and North America (as shown in Figure 3). The data covers a 496-day period from 17 February 2020 (Monday) to 27 June 2021 (Sunday); see Figure 4. It is divided into 69 overlapping episodes of 21 days (3 weeks) each, with a 7-day shift between consecutive episodes. This means that each episode overlaps with the previous one by 14 days. Each episode represents data for one country. The full set consists of 4130 episodes, while 10 are missing due to gaps in the data. Figure 5 shows the 21-day time series for the mobility indicators within each episode.

4.1.1. *Step 1: Generating single-attribute patterns*

We represent the episode-based time series by SAX patterns of length 5 using the alphabet $\{a, b, c, d, e\}$, where a corresponds to the lowest value interval and e to the highest value interval. In doing

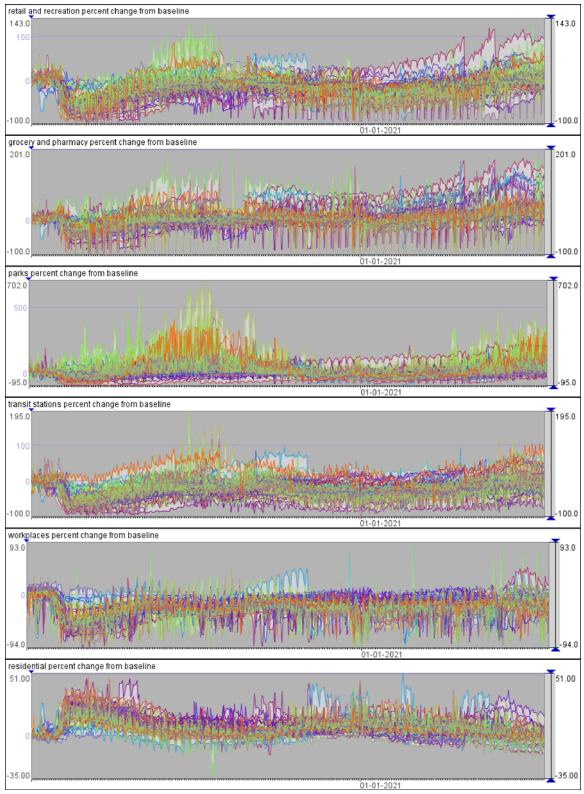

Figure 4: *The entire time series of the mobility indicators by the countries are shown on line plots. The lines have the same colours as the corresponding dots on the map showing the positions of the countries capitals (Figure 3).*

that, we skip the values for Saturdays and Sundays to disregard the irrelevant weekly variations of the mobility behaviours and consider the general trends over the 3 weeks time periods. Table A1 shows the breaks by which the value ranges of the attributes have been automatically divided into five bins so that each bin includes approximately 20% of the values. To visualize the patterns, we apply colour coding to the symbols of the alphabet that has been used in pattern generation. We use a diverging colour scale [HB03] from dark blue to dark red to encode the symbols corresponding to the value intervals ordered from the lowest to the highest. Figure 6 demonstrates the representation of the patterns.

4.1.2. *Step 2: Obtaining multi-attribute patterns*

We create pseudo-texts composed of the SAX patterns preceded by the abbreviated attribute names (e.g. 'residential' is abbreviated as 'home') and apply the topic modelling algorithm LDA to the resulting strings. After experimenting with the parameter k (number of topics), we find that $k = 9$ gives an acceptable result in terms of topic interpretability while the topics are not too numerous. The table display in Figure 7 shows the patterns that have certain weights in the topics, the minimal weight (in our example it is 0.005) being set through the slider at the bottom of the display. The interpretations of the topics that can be derived from this display are listed in Table 1.

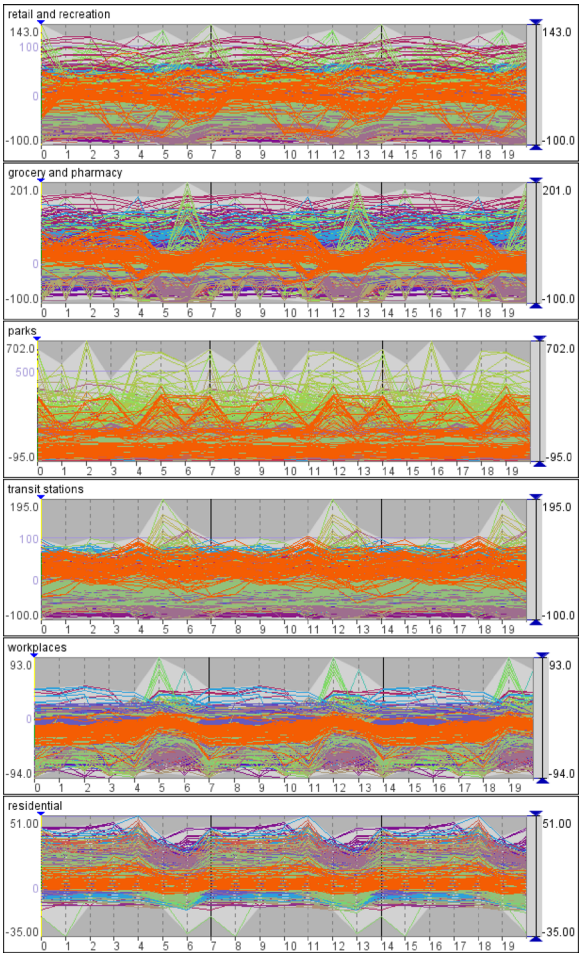

Figure 5: *The line plots show the time series of the mobility indicators by the episodes. The lines have the colours that have been earlier assigned to the countries (Figure 3). The time steps are from 0 to 20 according to the number of days passed since the beginning time of each episode.*

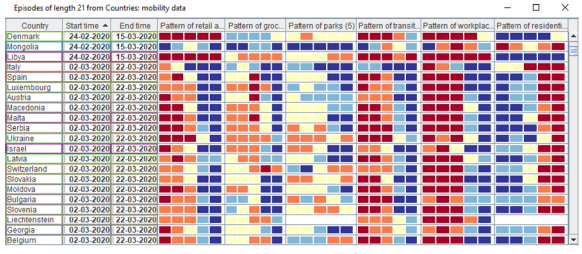

Figure 6: *A fragment of a table showing colour-coded SAX patterns of the mobility indicators in the episodes. The patterns of length five have been generated using the alphabet {a, b, c, d, e}, where a corresponds to the lowest value interval and e to the highest value interval. The symbols are represented using a diverging colour scheme from dark blue for a to dark red for e.*

Figure 7: *The topics resulting from applying the LDA algorithm.*

Table 1: *Interpretation of topics.*

N	Interpretation
0	More or less usual life
1,3,4	Average to high mobility, reduced staying at home
2,5	Average to low mobility, increased staying at home
7	Decreasing mobility, increasing staying at home
6,8	Strict 'stay at home'

It can be noted that there are groups of topics that can be interpreted similarly. The explanation is that there are multiple symbolic codes corresponding to pattern variants that have very close meanings for a human. For example, the codes *cdddd*, *cccdd*, *ddeee* and quite many others represent different variants of the pattern of increase. However, for the topic modelling algorithm, these are distinct and unrelated terms, which may belong to different topics. Therefore, running the algorithm with a lower value of the parameter k will not automatically unite semantically similar topics.

Another observation is that the topics are not 'clean' in terms of including similar or consistent patterns of the same attribute. Consider, for example, the patterns of the attribute 'home' in the topic 7. Almost all patterns represent average to high values and only one pattern represents values from the lowest range. Such inconsistent mixtures of patterns occur irrespective of the chosen number of topics k. To investigate this phenomenon in more detail, we select (by means of interactive filtering) the episodes with the symbolic pattern *aaaaa* of the attribute 'home' for which topic 7 has the highest weight among all topics. The 20 episodes satisfying the query are shown in a table view in Figure 8. We see that there are nine episodes where the lowest values of presence at home co-occurred with quite low levels of presence in the other categories of places, which seems counter-intuitive. Interestingly, eight of these episodes took place in Moldova. A possible reason may be that people did not frequently use their mobile devices while staying at home, which lead to underestimation of the people's presence. This reminds us that data

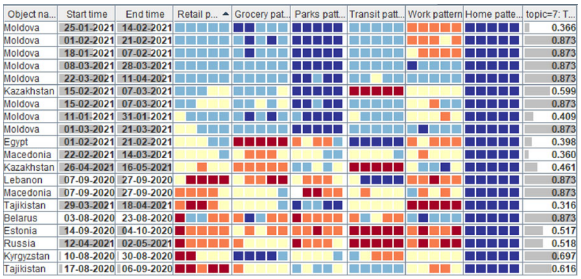

Figure 8: *A selected subset of episodes where topic 7 has the highest weight and the pattern of attribute 'home' is aaaaa. The rows of the table are arranged in the increasing order of the patterns of the attribute 'retail'.*

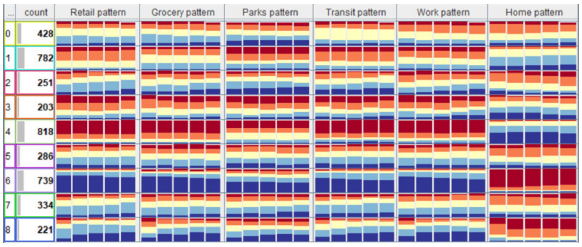

Figure 9: *The mobility patterns in the groups of episodes with different dominant topics are shown in an aggregated way by segmented bars.*

may be biased and require caution in interpreting analysis results and making inferences.

To verify and, if appropriate, refine the interpretations of the topics, it is useful to consider the groups of episodes with different dominant topics (i.e. having the highest weight) and the attribute patterns occurring in these groups of episodes. Since the episodes are numerous, the patterns need to be represented in an aggregated way. A possible way of aggregated representation of SAX patterns is demonstrated in Figure 9. The patterns of one attribute are aggregated by counting the occurrences of each symbol $\{a, b, c, d, e\}$ on each position from 1 to 5. The result is represented by a segmented bar chart where each bar corresponds to one position of a pattern and its segments represent the proportions of occurrences of the symbols $\{a, b, c, d, e\}$ in this position. The segments are painted in the colours that have been assigned to the symbols.

The display in Figure 9 confirms the interpretations of the topics and topic groups. Thus, the bar charts of the groups of episodes with dominant topics 1, 3, and 4 have large proportions of red (representing high values) for all place categories except home, which has large proportions of blue. The bars for the groups of episodes with dominant topics 6 and 8, on the opposite, have high amounts of blue for all places except home and high amounts of red for home. We also see differences between the topics within the groups in terms of the proportions of different colours; however, these differences can be treated as inessential for the interpretation.

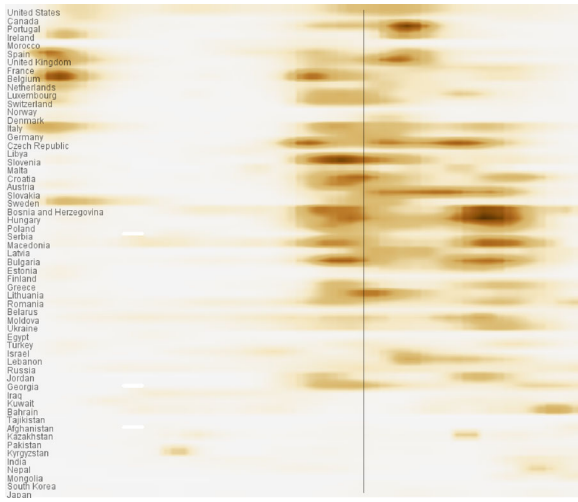

Figure 10: *The episodes are arranged in a matrix space with the rows corresponding to the countries and columns to the starting times of the episodes. The background colouring of the matrix represents the distribution of the average daily counts of the deaths due to COVID-19 (Source of the data: [Goo22]).*

4.1.3. *Step 3: Understanding the distribution of the multi-attribute patterns*

After obtaining and interpreting the topics, we want to investigate the contexts in which they occur. This includes the distribution of the topics over the set of countries and the time and their relationships to the pandemic spread indicators, such as the mortality due to COVID-19. To obtain a convenient visual representation of the context, we create an artificial matrix space (Figure 10) where the rows correspond to the countries and the columns to the start times of the episodes. To put the countries in a linear order, we apply the principal component method to the spatial positions of the country capitals and take the ordering based on the first component, as suggested by Wulms et al. [WBM*21]. With this approach, close spatial positions tend to receive close positions in the linear order. The episodes referring to each country are arranged chronologically in the corresponding row. The vertical line in Figure 10 approximately marks the time of Christmas. It is impossible to mark any date in this view precisely because the horizontal positions correspond not to individual days but to temporally overlapping episodes of 21 days length.

In this matrix space, we visualize the pandemic-caused mortality rates by shading from white for zero to dark brown for the highest values. The display in Figure 10 reveals prominent spatio-temporal patterns in the variation of the mortality rates. Thus, we observe high mortality rates in several countries of Europe (Ireland, Spain, UK, Belgium, Italy, and Sweden) in March and April 2020, relatively low values in the summer of 2020 and an increase of the deaths rates in many countries starting from October 2020.

Now we need to visualize the distribution of the topics in this context space. One possible approach is to use pie charts with sector sizes representing the topic weights for the episodes. Figure 11 shows a fragment of such a display. Please note that the topics have

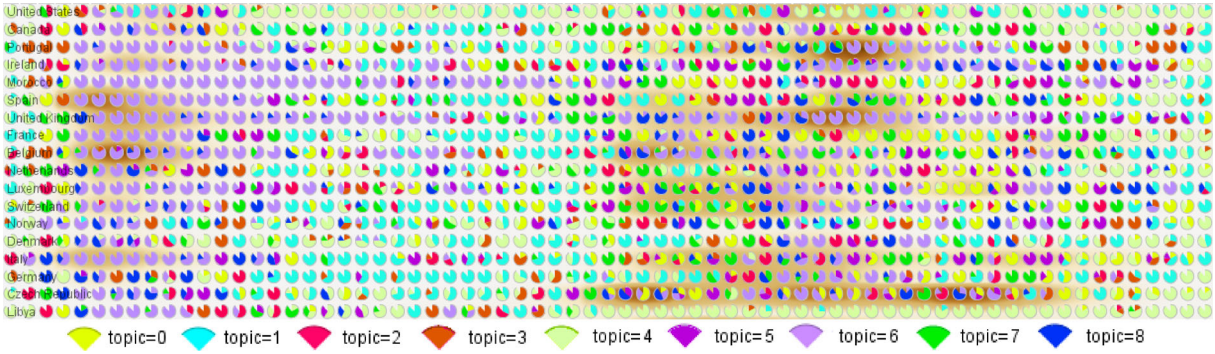

Figure 11: *A fragment of the matrix display with topic weights represented by pie charts.*

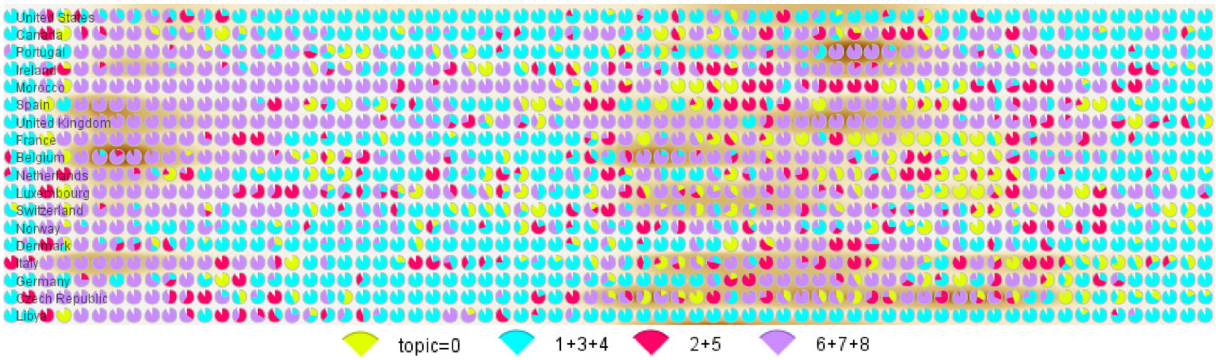

Figure 12: *A fragment of the matrix display with pie charts representing the weights of aggregated topics.*

been assigned distinct colours, which are shown in the caption of the table display in Figure 7. These colours are used for painting the corresponding sectors of the pies. The purpose of this representation is not to enable accurate perception of individual topic weights or estimation of the weight proportions (pie charts are commonly judged as poorly suited for these tasks [CM84]) but to provide an overall view of the distribution of the topic colours. To gain an overview, it is best to look at the display from a distance without paying much attention to the individual diagrams but instead perceiving the patterns of the overall colour distribution. This is possible due to the associative property of the visual variable 'colour' [Ber83].

Thus, we notice display areas with high amounts of lilac representing topic 6, which is interpreted as strict 'stay at home' regime. This topic prevails in the spring of 2020 and re-occurs in some countries in the winter and spring of 2021 when the death rates increase. The blue colour of topic 8, which is also interpreted as strict staying at home, occurs in the same periods as the lilac of topic 6, sometimes as sectors of the same pies. Topics 2 and 5 (red and magenta), both interpreted as decreased mobility and increased staying at home, often occur close in time to topics 6 and 8. High amounts of cyan representing topic 1 (average to high mobility and reduced staying at home) are observed in the summer of 2020 and also in the summer of 2021. The light green of topic 4 and brown of topic 3 also frequently occur in the same periods as topic 1.

Generally, we observe that topics with similar interpretations tend to have close positions in the matrix space, which reinforces our confidence that the topics are semantically close. This gives us a ground to aggregate semantically close topics, namely, unite the groups of topics $\{1, 3, 4\}$, $\{2, 5\}$, and $\{6, 7, 8\}$. Technically, the aggregation is done by summing up the weights of the topics of each group for each episode. Figure 12 demonstrates the appearance of the matrix display with pies after the aggregation. Here, the colours of the topics 1, 2, and 6 are assigned to the aggregates in which these topics are included.

The aggregation simplifies the perception of the patterns of colour distribution across the display. Nevertheless, to see the distributions of the individual (original or aggregated) topics more clearly, it may be beneficial to use a small multiple display as shown in Figure 13. For each topic, there is a separate matrix where the topic weights for the episodes are represented by proportional sizes of circle symbols.

Again, this view is meant not for estimation of individual values but for perceiving all circles in a matrix at once, in one instance of sight, that is as a single image [Ber83]. In so doing, we can see very prominently that the combined topics $1 + 3 + 4$ (increased mobility) and $6 + 7 + 8$ (increased staying at home) have complementary distribution patterns. The former occurs where and when the mortality rates are low, except for the early period of the pandemic spread (starting from mid-March), when lockdown regimes were introduced even in the countries whose local death rates had not yet significantly increased; see https://en.wikipedia.org/wiki/COVID-19_pandemic_in_Europe. The combined topic $6 + 7 + 8$, on the opposite, occurs almost everywhere in the period starting from mid-March and re-occurs after the summer of 2020 when the death rates increase in the majority of the countries. For the topics 0 and $2 + 5$,

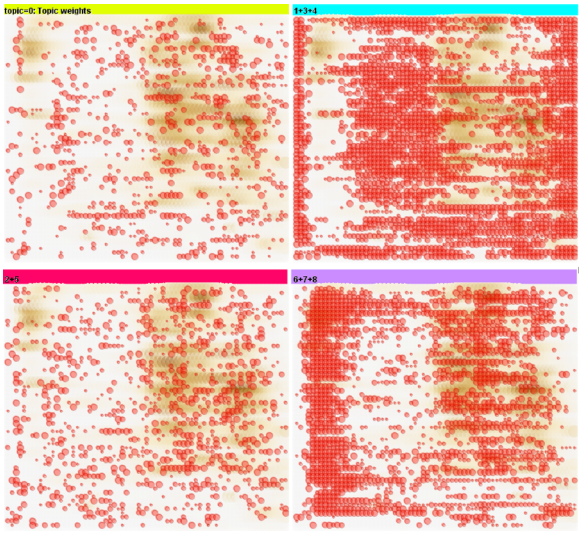

Figure 13: *A small multiple display where each matrix represents the distribution of the weights of one original or aggregated topic by proportional circle sizes.*

the small multiples display does not reveal obviously interpretable patterns. To understand the distribution of these topics, it is better to use the pie chart display (Figure 12). It shows that these topics tend to have intermediate positions between the periods of staying at home and periods of high mobility.

4.1.4. *Lessons learnt*

This investigation showed us that symbolic encoding of numeric time series, which involves division of the value range into bins, requires attention and, preferably, control from a human analyst. While there exists a sound rationale for dividing values into equal-frequency intervals [LKWL07], the analyst should examine results of automatic division to be able to interpret the codes correctly. In our case study, the middle value interval represented by the symbol *c* could be wrongly interpreted as values around zero, that is close to the pre-pandemic levels. In reality, this interval includes values around the median, which may significantly differ from zero. The meaning of the symbol *c* is defined by the interval breaks shown in the columns 2 and 3 of Table A1. While this division was suitable for our experimental study, there may be applications requiring involvement of domain knowledge and/or adopted conventions in the discretization of attribute value ranges. This can be enabled by interaction techniques that allow the effects of different divisions on the resulting symbolic patterns to be observed.

Another important lesson is that semantically close symbolic patterns are treated as completely different and unrelated by a topic modelling algorithm, which leads to generation of multiple topics with similar meanings from a human perspective. This reveals a need in interactive post-processing of topic modelling results, which includes merging of semantically close topics, as we did in Figure 12. It may also be appropriate to edit some topics to make their meanings clearer by modifying the weights of specific patterns. In our case, topic 7 could be edited by setting the weight of the pattern *aaaaa* for 'home' to zero (the weights of the remaining patterns should in this case be adjusted to make 1 in total). After such editing, the topic weights for the episodes need to be re-calculated.

4.2. Case study 2: Team behaviours in football

In this case study, we used tracking data from two football games of the German Bundesliga season 2019–2020, such that the same home team played against different guest teams. We used the original data, which included the players' and ball's trajectories, to derive time series of the following attributes that are used by FIFA to indicate team performance during a game [FIF22]:

- pressure of the defending players on the ball [AAB*17];
- pressure of the defending players on the attackers;
- percent of the attacking players in the final third of the pitch;
- depth and width of the home and guest teams on the pitch;
- stretch index of the home and guest teams;
- mean and minimal distance of the attacking players to the opponents' goal;
- minimal X-distance (i.e. distance along the pitch) of the defending players to their own goal.

The time resolution of the data is 25 steps per second, that is the time interval between consecutive time steps is 40 ms.

After excluding the time intervals when the ball was out of play from the time series, we extracted episodes of the length (duration) of 10 s starting at the moment of ball possession change as well as episodes starting 10 s before the ball possession change, that is the first and last 10 s on one team's ball possession. We skipped the time intervals where the ball possession of one team lasted for less than 9 s; however, intervals of very short ball possession (less than 1 s) were treated as parts of longer episodes of ball possession of the opponent team.

In the result, we got 250 episodes of possession start and 249 episodes of possession end, 499 episodes in total. Due to the way of episode extraction, episodes from the two categories may be overlapping or even coinciding in time. Thus, there are 21 duplicated episodes belonging to both categories. Overlapping of episode times and duplication of episodes are acceptable for our analysis, where we want to reveal differences (if any) between team behaviours at the beginning and at the end of one team's ball possession. Particularly, we want to see how teams begin their attacks, how defenders behave in response to that, and what is happening before defenders re-gain the ball. Still, we exclude the second instances of the 21 duplicated episodes from the further consideration to avoid their excessive impact on the results of topic modelling.

4.2.1. *Step 1: Generating single-attribute patterns*

In the episodes, we transform the time series of the attributes specifying team extents (depth, width, and stretch index) and distances to the goals into time series of changes (i.e. differences) with respect to the values at the beginning of the episodes. Then we generate SAX patterns of length 4 using the same alphabet as in the first use case. Table A2 shows the breaks of the attribute value ranges that were

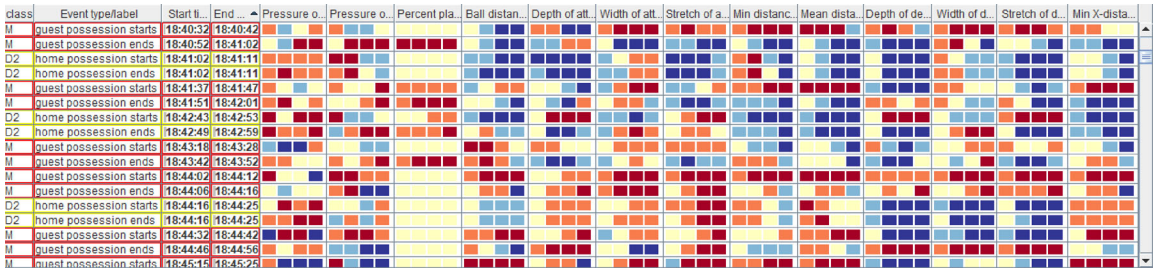

Figure 14: *A fragment of a table showing colour-coded SAX patterns of the football episodes. The patterns of length 4 were generated using the same alphabet as in the first case study.*

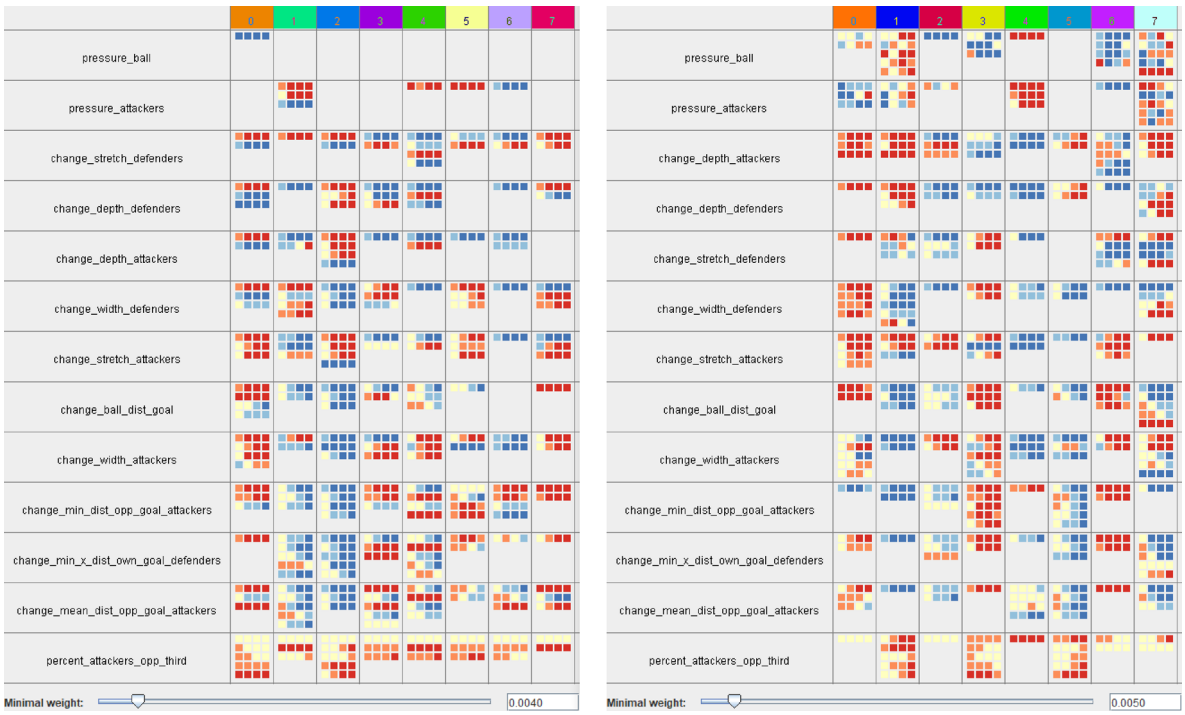

Figure 15: *Topics extracted from the football episodes by means of LDA (left) and NMF (right).*

applied for encoding the values by the symbols. A sample of the patterns can be seen in Figure 14. As in the previous case study, we construct strings including the patterns of all 13 attributes preceded by abbreviated attribute names.

4.2.2. *Step 2: Obtaining multi-attribute patterns*

In this case study, we compared the work of two topic modelling algorithms: Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorisation (NMF). We ran each of the methods with the same value of the parameter k (number of topics). Figure 15 shows the topics constructed by LDA and NMF for $k = 8$. The topics are represented by combinations of patterns having high weights.

It can be seen in Figure 15 that the topics produced by LDA (left table) are less clear than the topics resulting from NMF (right table). Thus, many cells in the left table contain both patterns of value increase (represented by shades of red) and patterns of value decrease (represented by shades of blue). This complicates interpretation of the topics. We also see that LDA did not give significant weights to the patterns of the attribute 'pressure on the ball', which mismatches our knowledge that exerting pressure on the ball is an important defensive tactics in football. The NMF topics, in contrast, are differentiated in terms of the pressure on the ball: topics 2, 3, and 6 are characterized by low pressure, topic 0 by moderate pressure, topic 4 by high pressure, and topics 1 and 7 by increasing pressure.

What concerns the patterns of the other attributes, we see a variety of attacking and defensive tactics. Thus, attackers may increase or decrease the team's depth and width on the pitch, and defenders tend to behave similarly. There are topics (0, 3, and 6) in which the ball moves away from the goal of the defending team and the attackers do not approach the opponents' goal and do not increase their presence in the final third of the pitch. The other topics represent more active attacks and corresponding defensive behaviours,

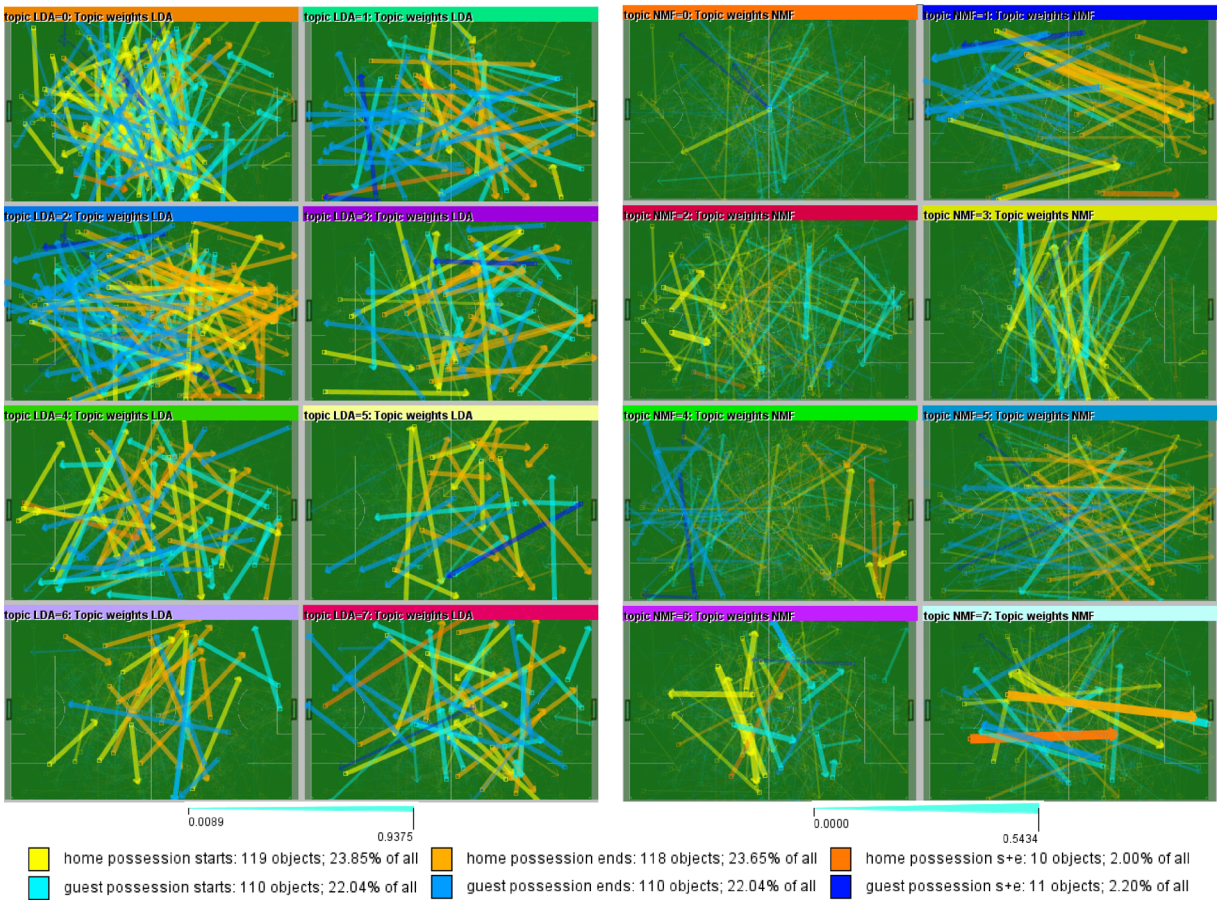

Figure 16: *The weights of the topics for the football episodes are represented on small multiple maps where each map shows the weights of one topic. The background of each map represents the football pitch. The episodes are represented by vectors (directed lines) connecting the initial and final positions of the ball. The line widths and opacity levels are proportional to the topic weights. The colours correspond to the types of the episodes. The images on the left and right show the weights of the LDA and NMF topics, respectively.*

including increased pressure on the ball and attackers when they approach the goal of the defending team. However, to understand better the behaviours represented by the topics, we need to see them in the context of the pitch in connection to the episodes.

4.2.3. *Step 3: Understanding the circumstances of the multi-attribute patterns*

The data we analyse require a domain-specific representation of the episodes and their topic weights. For this purpose, we use maps with the background representing the football pitch. The data have been transformed so that the goal of the home team is always on the left and the goal of the guest team on the right. We represent the episodes on the maps by vectors (directed lines) connecting the first and last positions of the ball, as can be seen in Figure 16. We apply colour coding to show which team possesses the ball in the episode and whether it is at the start or at the end of the ball possession; see the legend at the bottom of Figure 16. The weights of the topics are represented by proportional widths and, simultaneously, opacity levels of the vector lines. The redundant encoding of the weights improves the perception.

The position, orientation, and length of a vector not only show the overall relocation of the ball but also give some hints about the character of the episode: whether it was a swift attack along the pitch, or seeking a possibility for an attack while passing the ball across the pitch, or maintaining the ball possession while staying close to own goal, or offensive activities near the opponents' goal. We want to see whether these types of episodes would be distinguishable in terms of the topic weights.

Figure 16 contains two sets of small multiple maps. Each map represents the weights of one topic (represented by the background colour of the map caption) by line widths and opacity levels. The eight maps on the left correspond to the LDA-generated topics and the eight maps on the right to the NMF topics. The two sets of maps look very different. On the left, all vectors seem to have the same width. Indeed, for more than 76% of the episodes, the dominant LDA topic has the same weight 0.9375. Also for each of the remaining episodes, there is one LDA topic with a very high weight while the weights of the other LDA topics are close to zero. The

maps showing the weights of the LDA topics look very chaotic. We do not see any clear patterns in terms of the types and characteristics of the episodes. Each map shows a disorderly looking mixture of vectors of different origins, lengths, and orientations.

Different from what we see on the left, the brighter and thicker vectors in each map on the right have some features in common. In the map for topic 0, the vectors originate from a common point in the centre of the pitch. Many of them represent ball movements away from the defended goal, and this is consistent with the representation of topic 0 in the topic table in Figure 15, bottom. In contrast, the map for topic 1 highlights episodes (mostly at the end of ball possession, as signified by the line colours) with great advancements of the ball towards the target, which agrees with the patterns of the change of the ball and attackers distances to the goal shown in the topic table. The table also says that these episodes are characterized by the attackers increasing their presence in the opponents' third of the pitch, the defenders increasing their pressure on the ball and attackers, and both teams increasing their depth and decreasing widths. Similar behavioural patterns are observed in topics 5 and 7, and the vectors that are prominent in the corresponding maps also look similar to those in the map for topic 1.

Topic 3 expresses behaviours that are opposite to topics 1, 5, and 7. They are represented on the map by vectors oriented across the pitch and painted in the colours corresponding to the beginnings of ball possession. Topic 2 characterizes episodes in which teams' ball possession begins close to their own goals. They slowly move towards the opponents' goal stretching across the pitch while the opponents make their team more compact preparing to defend. Topic 4, in contrast, characterizes the behaviours of the teams in dramatic situations when the ball is close to the target. Both teams get more compact, and the defenders exert high pressure on the ball and attackers. Topic 6 reflects somewhat relaxed behaviours when ball possession begins close to the pitch centre, and the possessing teams move the ball closer to their own goal while stretching in width. In response, the defenders decrease their pressure on the ball while following the attackers' retreat and making their team more compact.

The maps in Figure 16 do not show clearly whether any topics prevail more for the home team or for the guest team. We would also like to compare the team behaviours in the two games for which we have data. We remind that the home team (Borussia Dortmund) was the same in both games. The guest team in the first game was FC Nuremberg and in the second game Bayern Munich. Assuming that the dominant topics of the episodes represent the main features of the teams' behaviours, we create a display of the co-occurrences of the dominant topics and the episode types; see Figure 17. On the top, the display shows the co-occurrences in both games, and the two screenshots below show the co-occurrences in the first and second games. The display consists of eight bar charts (for the eight topics) with horizontal bars oriented from right to left. The upper bar in a bar chart shows in how many episodes in total the corresponding topic was dominant. The following six bars show the frequencies of the dominance of this topic for the six types of episodes.

From the three instances of the co-occurrence display visible in Figure 17, we learn that topic 0 was rarely dominant in the episodes with the home team's possession. We also observe differences between the two games, especially in the episodes with the guest

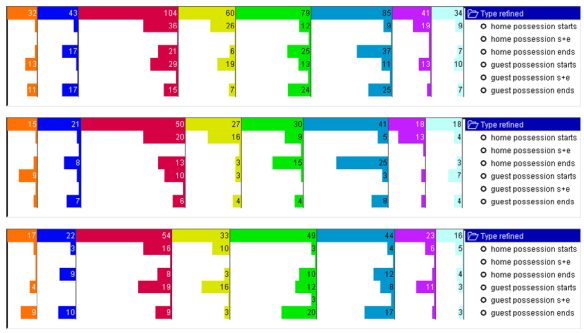

Figure 17: *Distribution of the dominant topics for the episode types in the whole dataset (top), data from game 1 (middle), and data from game 2 (bottom).*

team's possession. Under the guests' possession, the topics from 3 to 7 were dominant much more frequently in the second game than in the first. This may mean that the guest team of the first game did not vary its attacking behaviour as much as the guest team of the second game. It can also be noticed that topic 4 (green), which we interpreted as fighting close to the target, was very rarely dominant under the guest team's possession in the first game, whereas in the second game it prevailed much more frequently in the episodes with the guest team's possession than under the possession of the home team. Hence, Bayern Munich quite frequently created dangerous episodes at the goal of Borussia Dortmund. Similar whilst not so striking differences exist for topic 5 (light blue) characterized by the ball and the attackers approaching the target. This behaviour was more frequent under the home team's possession in game 1 and under the guest team's possession in game 2.

4.2.4. *Lessons learnt*

This experiment showed us that different methods of topic modelling may produce very different results and that some results may not be very useful. It is questionable whether the success or failure of a given algorithm for a given dataset can be predicted. There have been comparative studies of the efficacy of LDA and NMF in application to short texts, such as tweets. In some studies, the results of the methods were assessed as equally good [AYB20], whereas other researchers found that topics produced by NMF were more in line with human judgment [EY22]. However, whatever results may be obtained for text data, they are not necessarily transferable to non-textual applications of topic modelling. Therefore, if the chosen method does not produce an acceptable result, it may be worth trying another method.

Another useful lesson concerns preparation of the data. As we mentioned in Section 4.2.1, we transformed the original values of some of the attributes into differences from the values at the beginnings of the episodes and used the time series of the differences to generate the SAX patterns. We applied this transformation after an unsuccessful attempt to use the original time series. The resulting topics were trivial, mainly distinguishing the episodes based on the players' distances from the goals and not revealing differences in team behaviours. Therefore, transforming absolute values into

changes can be crucial in analysing behaviours. It is worth noting that the attribute values we had in the first case study were already provided as differences from the baseline values, which allowed us to analyse behaviour changes with respect to the pre-pandemic period.

5. Discussion

5.1. Application of the pattern theory

In this work, we aimed to find a way to derive a general understanding of a phenomenon reflected in episode-based data, which consist of multivariate time series encapsulated in episodes. To obtain an overall view of the phenomenon as a whole from elementary data (i.e. attribute values and time references), high abstraction is required. We applied the pattern theory [AAM*21], which posits that abstraction in data analysis is achieved through the discovery of patterns formed by relationships between data items. We developed an approach to analysis that incrementally increases the level of abstraction, beginning with the relationships between elementary data items (i.e. attribute values and time references) that form temporal variation patterns of individual attributes. To achieve the next level of abstraction, these patterns are treated as elements, and relationships between them (specifically, co-occurrence) are considered. We represented single-attribute patterns as tokens and used techniques such as topic modelling to discover patterns of token co-occurrence. However, to gain an overall understanding, the level of abstraction needs to be further increased by discovering patterns in the distribution of these token co-occurrence patterns over time and in relevant contexts.

Using the pattern theory, we devised an abstract analysis workflow that includes three steps of abstraction (Figure 2). We defined the types of patterns to be discovered in each step and the types of relationships that are involved in these patterns. This abstract workflow is transformed into a concrete work plan by choosing methods that will be used to implement each abstract operation. We chose the SAX pattern method [LKWL07] for the first step of the analysis workflow, topic modelling [VK20] for the second step, and interactive visualizations of topic distributions for the third step. We implemented this work plan for two different datasets reflecting phenomena of distinct nature and scale and found the approach to be effective. However, it is important to note that other implementations of the abstract workflow may be possible, as discussed below.

5.2. Design space in implementing the abstract workflow

Step 0: Data pre-processing. If the original data have the form of continuous time series rather than episodes, they need to be transformed to episodes. This can be done using any of the existing approaches, for example one of those mentioned in Section 3.2. We applied the sliding window approach in the first case study and event-based definition of episodes in the second. Domain knowledge may be involved in defining episodes, for example to ignore irrelevant weekly fluctuations, as in the first study, or out-of-play times, as in the second study. Besides division into episodes, data pre-processing may include data cleaning, missing value imputation, smoothing, transforming absolute values to relative, aggregation, re-sampling, and so on.

Step 1: Deriving single-attribute patterns. In this step, the task is to transform each temporal sequence of attribute values into an object that can be represented both visually, to enable human interpretation, and as a symbolic token or word, to enable computational processing in the second step. Apart from the SAX method [LKWL07] that we used in our studies, a variety of possibilities exist. One of them is to detect predefined shapes [HÖ2, SAA23] and represent them visually as shapes and symbolically by words ‘increase’, ‘decrease’, ‘peak’, etc. Another possibility is to transform the original values to changes with respect to the previous or initial value apply the SAX method to the transformed data [SAA21]. Domain-specific rules [CC99, AMM*08] or a domain ontology [Sha97] can be employed to assign human-understandable labels to time series, and these labels can represent the patterns in the following analysis. Generally, any approach producing interpretable codes or meaningful labels is suitable for this step.

Step 2: Deriving multi-attribute patterns. Here, the symbolic representations of single-attribute patterns serve as an input to a method capable to detect repeated co-occurrences of the patterns within episodes. Topic modelling methods, namely, LDA [BNJ03] and NMF [LNC*17], proved to be suitable for this purpose. While these methods are the most popular, there are many other topic modelling methods that can be potentially applied. Multiple surveys [KB18, VK20, AEG*23] discuss properties and capabilities of different methods, so that an analyst can make an informed choice. Apart from topic modelling, re-occurring combinations of single-attribute patterns represented by labels can be detected using various algorithms designed for frequent item set mining [HCXY07, LFVV19]. It should be noted, however, that these algorithms tend to produce an excessive number of patterns, which may be very challenging for the following exploration.

It may be interesting to try network analysis, specifically, community detection methods [JYL*18]. The input may be a graph with vertices corresponding to the single-attribute patterns and weighted edges connecting patterns that occurred together, the weights being the counts of the joint occurrences. Network analysis, however, provides a different kind of information than we obtained using topic modelling. It reveals strong pairwise associations, but existence of a community including three or more patterns does not necessarily mean that all these patterns often occur together.

Due to these limitations and inconveniences of the network analysis and item set mining methods, we consider topic modelling to be a better tool for fulfilling the second step of data abstraction.

Step 3: Finding patterns of distribution of multi-attribute patterns. Implementation of this step is data- and domain-specific. We propose to support this step by visualizations designed according to the nature of the data and analysis goals. The key idea is to colour-code the second-level patterns and use these colours to represent the patterns in visual displays. If the analysis goals require considering the distribution of the patterns over time, the task can be supported by variants of a time line display, as we did in our first study. The episodes are positioned along the time axis according to their

existence times. Since several patterns may be associated with one episode, the combination of these pattern can be represented by a diagram or glyph consisting of elements painted in the colours of the patterns. This can be considered as a basic design for exploring the temporal distribution of patterns. Our visualizations provide examples of using the second display dimension to represent a relevant aspect of the context in which the episodes occur. In our case, it is spatial location (country), but it is also possible to represent other kinds of context information. When the distribution of the patterns with respect to temporal cycles is of interest, polar coordinates can be used instead of Cartesian.

In our second study, the distribution of the patterns over time was not important for the analysis. We wanted to see how the second-level patterns, i.e., the topics, are related to spatial properties of the episodes, which was the relevant type of contextual information. We used a small multiples display with one panel for each topic. Within the panels, we visualized the relevant properties (namely, the ball possession and displacement vector) of the episodes significantly associated with the corresponding topics. The small multiples is a general design that can be applied to different types of data, while the visualizations within the panels depends on the nature of the episodes and analysis goals. The topic weights for the episodes shown in the panels can be represented by a suitable visual variable; we used line widths.

Hence, the general design recommendations for visually supporting the third step of abstraction include (a) timeline display of episodes, possibly, with an additional dimension representing some aspect of the context; (b) circular display with polar coordinates representing temporal positions of episodes; (c) diagrams or glyphs showing topic composition for the episodes; (d) small multiples display with panels corresponding to topics and application-specific representation of properties of the episodes associated with the topics.

5.3. Technical aspects of computational methods

In the following section, we will discuss the methods we have used in our implementation of the abstract workflow.

5.3.1. *SAX encoding*

The Symbolic Aggregate approXimation (SAX) method [LKWL07] divides the time series into equal segments and computes a single numeric value for each segment. This requires deciding how many parts to select and which aggregation function to use. The number of parts is selected according to the desired level of detail in representing a time series. In our first use case, we divided the time series into five segments, resulting in one aggregated value representing three original daily values (we remind that we took episodes consisting of the weekdays of three consecutive weeks, i.e. 15 days in total). In the second use case, we used 4 segments to represent attribute dynamics over 10 s.

The SAX method typically uses either the mean or median as the aggregation function, but different application domains may require different approaches. For example, in Schreck et al.'s work on financial time series analysis [STKF07], the raw data included stock sell transactions with amounts and prices. The transactions for each stock have been aggregated by daily intervals into the average price and total volume. Domain experts may suggest other aggregation methods such as opening and closing prices, minimum and maximum prices for the day, or the difference/ratio between opening and closing prices or between maximum and minimum values. Some of these aggregates were used by Shirato et al. [SAA21] to represent trends over parts of time series data.

After aggregating the values, the next step in the SAX method is to symbolically encode the aggregate values based on their frequency distribution. Two decisions must be made at this stage: the size of the alphabet (i.e. the number of bins to divide the distribution into) and the breaks in the range of aggregate values that determine how the distribution is divided into bins. The number of bins affects the level of detail in the resulting representation. In our case studies, we used five bins to represent values around the average, values moderately lower and moderately higher than the average, and values much lower and much higher. The classical SAX method uses equal-frequency breaks, but it may be useful to consider other options based on domain knowledge and the semantics of the attributes, as we discussed in connection with our first case study.

5.3.2. *Topic modelling*

There are two key questions to consider when using a topic modeling method: which method to choose and how to set its parameters. A detailed review of available methods can be found in [VK20]. It is worth noting that Non-Negative Matrix Factorization (NMF) has been found to be more effective than Latent Dirichlet Allocation (LDA) for short texts [AYB20, EY22], but it is unclear if this applies to non-textual data. In our case studies, LDA worked well in the first study and poorly in the second, even though the 'texts' (i.e. symbolic descriptions of the episodes) were longer in the second study. It is worth keeping in mind that LDA is a probabilistic method. It works better with large amounts of 'texts' (i.e. large number of episodes), so that term probabilities could be more reliably estimated. The number of episodes was quite low in our second study, which may be the reason of the failure of the LDA method. To determine the target number of topics, we used the approach proposed by Chen et al. [CAA*20]: running the selected method with different parameters, projecting the topics generated in the different runs into a single embedding space using a dimensionality reduction method, and exploring the topic distribution in this space, which is expected to reflect the similarities and differences between the topics. The number of visible distinct clusters in the projection suggests the potentially suitable number of topics. However, the main criterion is interpretability of the topics by a human. As it cannot be formally evaluated, we do not see a feasible way to fully automate the selection of the suitable number of topics.

5.3.3. *Visualization techniques*

The abstract analytical workflow (shown in Figure 2) is designed for a human analyst to gain insights into the behaviour of a phenomenon. All steps in the workflow require human reasoning, which should be supported by appropriate visualizations of relevant information. As the workflow is defined abstractly, it does not specify which visualizations should be used, but it suggests the types of

information that need to be visualized: (1) single-attribute variation patterns, (2) multi-attribute combination patterns, and (3) the distribution of multi-attribute combination patterns over a set of episodes.

In our example implementation, we chose to use colour encoding for the SAX representation of the single-attribute patterns because the visual variable 'colour' has a strong association capacity [Ber83], allowing multiple coloured stripes drawn close together to be efficiently perceived as a single image. This helps with the effective perception of tables displaying multiple SAX patterns, including tables of episodes (Figures 6, 8, 14) and tables of topics (Figures 7, 15). In contrast, the visual variable 'shape' is not associative [Ber83], so representing patterns by shapes instead of coloured stripes would require inefficient scanning and memorization of individual shapes, making it more difficult to interpret the overall display. Another advantage of using colour-coding is the ability to represent SAX patterns summarized by groups of episodes in the form of segmented bars, as shown in Figure 9.

We also chose to use the visual variable 'colour' to represent topics, with each topic being encoded by a distinct colour. This allows us to use charts with coloured segments to represent topic mixtures. Specifically, we used pie charts, which are compact, easily perceived as units rather than conglomerates of distinct elements, and enable the estimation of relative weights of the topics. We used the selective power of the visual variable 'colour'[Ber83] to differentiate the topics and the associative power of this variable to support the perception of topic distribution patterns from displays with multiple pie charts, as shown in Figures 11 and 12.

The ways to visualize the distribution of topics over episodes depend on the organization of the set of episodes and the patterns that can be expected based on domain knowledge. In our first case study, the episodes refer to different times over a long period and to different entities (countries). To visualize the distribution, we created a matrix with columns and rows corresponding to the times and entities and placed pie charts in the cells. We also used background shading of the cells to represent some aspect of the relevant context, enabling the investigation of the distribution in relation to the context. This kind of visualization would not be useful in the second case study, where team tactics constantly vary during a game and no meaningful temporal patterns of topic distribution can be expected. Instead, it is more appropriate to investigate how the topics are related to spatial properties of the episodes. This motivated the visualization of the episodes in the space of the football pitch. We created a small multiple display to show the distribution of the weights of each topic over the episodes (Figure 16). We also found small multiple displays to be useful in the first case study (Figure 13). As stated in Section 5.2, timeline and small multiples are basic designs that can be used for episodes of various kinds.

A limitation of all displays we used is low scalability regarding the number of topics. However, a large number of topics may also be problematic for human interpretation and analysis; therefore, an analyst should strive to generate the minimal number of topics that are easily distinguished and well understood. Another problem may be a large number of episodes, which are hard to visualize without display clutter. A possible approach to alleviate this problem is aggregation of episodes and increasing the level of detail when the user zooms in or filters the data.

5.3.4. *Software implementation*

For our studies, we utilized the implementation of the topic modelling methods from the *scikit-learn 1.3.1* [PVG*11] Python library. For the data processing and visualization, we used our in-house system V-Analytics [AAB*13], which has been developed by the authors over many years. Researchers interested in the latest version of V-Analytics are welcome to contact the authors; however, there are also state-of-the-art libraries available, such as MovingPandas [Gra19], that offer similar processing capabilities. For visualizations, Python libraries like Plotly and Bokeh, as well as the JavaScript library D3 [BOH11], can be utilized. The specific details of our software go beyond the scope of this paper, as our primary focus is on presenting the abstract workflow and providing examples of its implementation.

6. Conclusion

In our work, we applied a theoretical model [AAM*21] to develop an abstract general approach to analysing the type of data in which a set of episodes is characterized by multiple time-variant attributes. This approach involves incrementally increasing the level of data abstraction by merging multiple elements into patterns. We implemented this approach by selecting specific methods for each step and tested the resulting workflow in two case studies. In particular, we evaluated the usefulness of topic modelling methods for deriving multi-attribute combination patterns from patterns of temporal variation of individual attributes. Topic modelling has proved to be useful in two distinct case studies and can therefore be recommended for this kind of tasks.

At a broader level, our work demonstrates the feasibility and value of a theory-based approach for devising data analysis workflows and choosing appropriate methods to implement them. In the future, we plan to continue applying theoretical models proposed for visual analytics, such as theories of data patterns [AAM*21], knowledge generation [SSS*14], model building [ALA*18], and qualitative analysis [KHL21], to different types of data. Our goal is not only to find effective ways to analyse data, but also to identify and demonstrate the prescriptive potential of these primarily descriptive theoretical models.

Acknowledgements

This research was supported by Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia as part of the *Lamarr Institute for Machine Learning and Artificial Intelligence (Lamarr22B)*, and by EU in projects *SoBigData++* and *CrexData*.

Open access funding enabled and organized by Projekt DEAL.

References

[AA23] Andrienko N., Andrienko G.: It's about time: Analytical time periodization. *Computer Graphics Forum 42*, (2023), e14845. https://doi.org/10.1111/cgf.14845.

[AAB*13] Andrienko G., Andrienko N., Bak P., Keim D., Wrobel S.: *Visual analytics of movement*. Springer Science & Business Media, 2013. https://doi.org/10.1007/978-3-642-37583-5.

[AAB*17] Andrienko G., Andrienko N., Budziak G., Dykes J., Fuchs G., von Landesberger T., Weber H.: Visual analysis of pressure in football. *Data Mining and Knowledge Discovery 31*, 6 (2017), 1793–1839. https://doi.org/10.1007/s10618-017-0513-2.

[AAC*17] Andrienko N., Andrienko G., Camossi E., Claramunt C., Cordero-Garcia J. M., Fuchs G., Hadzagic M., Jousselme A.-L., Ray C., Scarlatti D., Vouros G.: Visual exploration of movement and event data with interactive time masks. *Visual Informatics 1*, 1 (2017), 25–39. https://doi.org/10.1016/j.visinf.2017.01.004.

[AAK*21] Andrienko G., Andrienko N., Kureshi I., Lee K., Smith I., Staykova T.: Automating and utilising equal-distribution data classification. *International Journal of Cartography 7*, 1 (2021), 100–115. https://doi.org/10.1080/23729333.2020.1863000.

[AAM*21] Andrienko N., Andrienko G., Miksch S., Schumann H., Wrobel S.: A theoretical model for pattern discovery in visual analytics. *Visual Informatics 5*, 1 (2021), 23–42. https://doi.org/10.1016/j.visinf.2020.12.002.

[AEG*23] Abdelrazek A., Eid Y., Gawish E., Medhat W., Hassan A.: Topic modeling algorithms and applications: A survey. *Information Systems 112* (2023), 102–131. https://doi.org/10.1016/j.is.2022.102131.

[ALA*18] Andrienko N., Lammarsch T., Andrienko G., Fuchs G., Keim D., Miksch S., Rind A.: Viewing visual analytics as model building. *Computer Graphics Forum 37*, 6 (2018), 275–299. https://doi.org/10.1111/cgf.13324.

[AMM*08] Aigner W., Miksch S., Müller W., Schumann H., Tominski C.: Visual methods for analyzing time-oriented data. *IEEE Transactions on Visualization and Computer Graphics 14*, 1 (2008), 47–60. https://doi.org/10.1109/TVCG.2007.70415.

[AMST11] Aigner W., Miksch S., Schumann H., Tominski C.: *Visualization of Time-oriented Data*. Springer (2011). https://doi.org/10.1007/978-0-85729-079-3.

[AYB20] Albalawi R., Yeap T. H., Benyoucef M.: Using topic modeling methods for short-text data: A comparative analysis. *Frontiers in Artificial Intelligence 3* (2020). https://doi.org/10.3389/frai.2020.00042.

[BBB*18] Bernard J., Bors C., Bögl M., Eichner C., Gschwandtner T., Miksch S., Schumann H., Kohlhammer J.: Combining the automated segmentation and visual analysis of multivariate time series. In EuroVA@ EuroVis (2018), pp. 49–53.

[BBC16] Bondu A., Boullé M., Cornuéjols A.: Symbolic representation of time series: A hierarchical coclustering formalization. In *Advanced Analysis and Learning on Temporal Data* (Cham, 2016), Douzal-Chouakria A., Vilar J. A., Marteau P.-F., (Eds.), Springer International Publishing, pp. 3–16. https://doi.org/10.1007/978-3-319-44412-3_1.

[BDB*16] Bernard J., Dobermann E., Bögl M., Röhlig M., Vögele A., Kohlhammer J.: Visual-interactive segmentation of multivariate time series. In *Proceedings of the EuroVis Workshop on Visual Analytics* (Goslar, DEU, 2016), Eurographics Association, pp. 31–35.

[Ber83] Bertin J.: *Semiology of Graphics*. University of Wisconsin Press, 1983.

[BFF*20] Barandas M., Folgado D., Fernandes L., Santos S., Abreu M., Bota P., Liu H., Schultz T., Gamboa H.: TSFEL: Time series feature extraction library. *SoftwareX 11* (2020), 100456. https://doi.org/10.1016/j.softx.2020.100456.

[BHR*19] Bernard J., Hutter M., Reinemuth H., Pfeifer H., Bors C., Kohlhammer J.: Visual-interactive preprocessing of multivariate time series data. *Computer Graphics Forum 38*, 3 (2019), 401–412. https://doi.org/10.1111/cgf.13698.

[BNJ03] Blei D. M., Ng A. Y., Jordan M. I.: Latent Dirichlet Allocation. *Journal of Machine Learning Research 3*, Jan (2003), 993–1022.

[BOH11] Bostock M., Ogievetsky V., Heer J.: D3: Data-driven documents. *IEEE Transactions on Visualization & Computer Graphics (Proc. InfoVis)* (2011). URL: http://vis.stanford.edu/papers/d3.

[BSH*16] Bach B., Shi C., Heulot N., Madhyastha T., Grabowski T., Dragicevic P.: Time curves: Folding time to visualize patterns of temporal evolution in data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (Jan 2016), 559–568. https://doi.org/10.1109/TVCG.2015.2467851.

[BWK*13] Bernard J., Wilhelm N., Krüger B., May T., Schreck T., Kohlhammer J.: Motionexplorer: Exploratory search in human motion capture data based on hierarchical aggregation. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2257–2266. https://doi.org/10.1109/TVCG.2013.178.

[BWS*12] Bernard J., Wilhelm N., Scherer M., May T., Schreck T.: TimeSeriesPaths : Projection-based explorative analysis of multivariate time series data. In Journal of WSCG (2012), pp. 97–106.

[CAA*20] Chen S., Andrienko N., Andrienko G., Adilova L., Barlet J., Kindermann J., Nguyen P. H., Thonnard O., Turkay C.: Lda ensembles for interactive exploration and categorization of behaviors. *IEEE Transactions on Visualization and Computer Graphics 26*, 9 (2020), 2775–2792. https://doi.org/10.1109/TVCG.2019.2904069.

[CC99] Combi C., Chittaro L.: Abstraction on clinical data sequences: an object-oriented data model and a query language based on the event calculus. *Artificial Intelligence in Medicine*

17, 3 (1999), 271–301. https://doi.org/10.1016/S0933-3657(99)00022-6.

[CM84] Cleveland W. S., McGill R.: Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association 79*, 387 (1984), 531–554. https://doi.org/10.1080/01621459.1984.10478080.

[CS10] Chia C., Syed Z.: Using adaptive downsampling to compare time series with warping. In *ICDMW 2010, The 10th IEEE International Conference on Data Mining Workshops* (2010), Fan W., Hsu W., Webb G. I., Liu B., Zhang C., Gunopulos D., Wu X., (Eds.), IEEE Computer Society, pp. 1304–1311. https://doi.org/10.1109/ICDMW.2010.94.

[CSZ*14] Chu D., Sheets D. A., Zhao Y., Wu Y., Yang J., Zheng M., Chen G.: Visualizing hidden themes of taxi movement with semantic transformation. In *2014 IEEE Pacific Visualization Symposium* (March 2014), pp. 137–144. https://doi.org/10.1109/PacificVis.2014.50.

[CTH16] Chen T.-H., Thomas S. W., Hassan A. E.: A survey on the use of topic models when mining software repositories. *Empirical Software Engineering 21*, 5 (2016), 1843–1919. https://doi.org/10.1007/s10664-015-9402-8.

[CTMMB22] Chernikov A., Tan C. W., Montero-Manso P., Bergmeir C.: Frans: Automatic feature extraction for time series forecasting, 09 2022. https://doi.org/10.48550/arXiv.2209.07018.

[CvW18] Cappers B. C., van Wijk J. J.: Exploring multivariate event sequences using rules, aggregations, and selections. *IEEE Transactions on Visualization and Computer Graphics 24*, 1 (2018), 532–541. https://doi.org/10.1109/TVCG.2017.2745278.

[DP73] Douglas D. H., Peucker T. K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization 10*, 2 (1973), 112–122. https://doi.org/10.1002/9780470669488.ch2.

[EY22] Egger R., Yu J.: A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in Sociology 7* (2022). https://doi.org/10.3389/fsoc.2022.886498.

[FIF22] FIFA: Football intelligence, 2022. [accessed 26=December-2022]. URL: https://www.fifa.com/technical/football-technology/media-releases/fifa-to-introduce-enhanced-football-intelligence-at-fifa-world-cup-2022-tm.

[FSS*21] Fujiwara T., Shilpika, Sakamoto N., Nonaka J., Yamamoto K., Ma K.-L.: A visual analytics framework for reviewing multivariate time-series data with dimensionality reduction. *IEEE Transactions on Visualization and Computer Graphics 27*, 2 (2021), 1601–1611. https://doi.org/10.1109/TVCG.2020.3028889.

[GCML06] Guo D., Chen J., MacEachren A., Liao K.: A visualization system for space-time and multivariate patterns (VIS-STAMP). *IEEE Transactions on Visualization and Computer Graphics 12*, 6 (2006), 1461–1474. https://doi.org/10.1109/TVCG.2006.84.

[GGJ*22] Guo Y., Guo S., Jin Z., Kaul S., Gotz D., Cao N.: Survey on visual analysis of event sequence data. *IEEE Transactions on Visualization and Computer Graphics 28*, 12 (2022), 5091–5112. https://doi.org/10.1109/TVCG.2021.3100413.

[GLS*13] García S., Luengo J., Sáez J. A., López V., Herrera F.: A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering 25*, 4 (2013), 734–750. https://doi.org/10.1109/TKDE.2012.35.

[Goo22] Google: Google COVID-19 open data repository, 2022. [accessed 7-September-2022]. URL: https://health.google.com/covid-19/open-data/.

[Gra19] Graser A.: MovingPandas: Efficient structures for movement data in python. *GI_Forum 1* (2019), 54–68. https://doi.org/10.1553/giscience2019_01_s54.

[GYD*19] Gharghabi S., Yeh C.-C. M., Ding Y., Ding W., Hibbing P., LaMunion S., Kaplan A., Crouter S. E., Keogh E.: Domain agnostic online semantic segmentation for multi-dimensional time series. *Data Mining and Knowledge Discovery 33*, 1 (2019), 96–130. https://doi.org/10.1007/s10618-018-0589-3.

[HÖ2] Höppner F.: Time series abstraction methods - a survey. In *Informatik Bewegt: Informatik 2002-32. Jahrestagung Der Gesellschaft Für Informatik e.v. (GI)* (2002), GI, p. 777–786.

[HB03] Harrower M., Brewer C. A.: Colorbrewer.org: An online tool for selecting colour schemes for maps. *The Cartographic Journal 40*, 1 (June 2003), 27–37. https://doi.org/10.1179/000870403235002042.

[HCXY07] Han J., Cheng H., Xin D., Yan X.: Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery 15*, 1 (2007), 55–86.

[HWX*10] Hu Y., Wu S., Xia S., Fu J., Chen W.: Motion track: Visualizing variations of human motion data. In *IEEE Pacific Visualization Symposium (PacificVis)* (2010), pp. 153–160. https://doi.org/10.1109/PACIFICVIS.2010.5429596.

[JC16] Jolliffe I. T., Cadima J.: Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 374*, 2065 (2016), 20150202. https://doi.org/10.1098/rsta.2015.0202.

[Jen77] Jenks G. F.: Optimal data classification for choropleth maps. *Department of Geographiy, University of Kansas Occasional Paper* (1977).

[JME10] Javed W., McDonnel B., Elmqvist N.: Graphical perception of multiple time series. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 927–934. https://doi.org/10.1109/TVCG.2010.162.

[JYL*18] Javed M. A., Younis M. S., Latif S., Qadir J., Baig A.: Community detection in networks: A multidisciplinary review. *Journal of Network and Computer Applications 108* (2018), 87–111.

[KB18] Kherwa P., Bansal P.: Topic modeling: A comprehensive review. *ICST Transactions on Scalable Information Systems 7* (07 2018), 159623. https://doi.org/10.4108/eai.13-7-2018.159623.

[KFC*22] Kuo Y.-H., Fujiwara T., Chou C. C., Chen C., Ma K.-L.: A machine-learning-aided visual analysis workflow for investigating air pollution data. *2022 IEEE 15th Pacific Visualization Symposium (PacificVis)* (2022), 91–100.

[KHL21] Karer B., Hagen H., Lehmann D. J.: Insight beyond numbers: The impact of qualitative factors on visual data analysis. *IEEE Transactions on Visualization and Computer Graphics 27*, 2 (2021), 1011–1021. https://doi.org/10.1109/TVCG.2020.3030376.

[LFVV19] Luna J. M., Fournier-Viger P., Ventura S.: Frequent itemset mining: A 25 years review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9*, 6 (2019), e1329.

[LKWL07] Lin J., Keogh E., Wei L., Lonardi S.: Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery 15*, 2 (Oct. 2007), 107–144.

[LNC*17] Luo M., Nie F., Chang X., Yang Y., Hauptmann A., Zheng Q.: Probabilistic non-negative matrix factorization and its robust extensions for topic modeling. In *Thirty-first AAAI Conference on Artificial Intelligence* (2017).

[LSK*19] Lubba C. H., Sethi S. S., Knaute P., Schultz S. R., Fulcher B. D., Jones N. S.: Catch22: Canonical time-series characteristics: Selected through highly comparative time-series analysis. *Data Mining and Knowledge Discovery 33*, 6 (nov 2019), 1821–1852. https://doi.org/10.1007/s10618-019-00647-x.

[LTD*16] Liu L., Tang L., Dong W., Yao S., Zhou W.: An overview of topic modeling and its current applications in bioinformatics. *SpringerPlus 5*, 1 (2016), 1–22. https://doi.org/10.1186/s40064-016-3252-8.

[MLL*13] Monroe M., Lan R., Lee H., Plaisant C., Shneiderman B.: Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2227–2236. https://doi.org/10.1109/TVCG.2013.200.

[MW22] Merriam-Webster: episode, 2022. [accessed 26=December-2022]. URL: https://www.merriam-webster.com/dictionary/episode.

[NHZ16] Niu Y. S., Hao N., Zhang H.: Multiple change-point detection: A selective overview. *Statistical Science 31*, 4 (2016), 611–623.

[PVG*11] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[SAA21] Shirato G., Andrienko N., Andrienko G.: What are the topics in football? Extracting time-series topics from game episodes. *IEEE VIS 2021 poster* (2021). URL: http://geoanalytics.net/and/papers/vis21poster.pdf.

[SAA23] Shirato G., Andrienko N., Andrienko G.: Identifying, exploring, and interpreting time series shapes in multivariate time intervals. *Visual Informatics 7*, 1 (2023), 77–91. https://doi.org/10.1016/j.visinf.2023.01.001.

[SAAF18] Shurkhovetskyy G., Andrienko N., Andrienko G., Fuchs G.: Data abstraction for visualizing large time series. *Computer Graphics Forum 37*, 1 (2018), 125–144. https://doi.org/10.1111/cgf.13237.

[Sha97] Shahar Y.: A framework for knowledge-based temporal abstraction. *Artificial Intelligence 90*, 1 (1997), 79–133. https://doi.org/10.1016/S0004-3702(96)00025-2.

[SLCB07] Sacchi L., Larizza C., Combi C., Bellazzi R.: Data mining with temporal abstractions: Learning rules from time series. *Data Mining and Knowledge Discovery 15*, 2 (oct 2007), 217–247. https://doi.org/10.1007/s10618-007-0077-7.

[SMKH22] Slocum T. A., McMaster R. B., Kessler F. C., Howard H. H.: *Thematic Cartography and Geovisualization*. CRC Press, 2022.

[SSS*14] Sacha D., Stoffel A., Stoffel F., Kwon B. C., Ellis G., Keim D. A.: Knowledge generation model for visual analytics. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (2014), 1604–1613. https://doi.org/10.1109/TVCG.2014.2346481.

[Ste13] Steinarsson S.: *Downsampling Time Series for Visual Representation*.PhD thesis, University of Iceland, 2013.

[STKF07] Schreck T., Tekušová T., Kohlhammer J., Fellner D.: Trajectory-based visual analysis of large financial time series data. *SIGKDD Explorations Newsletter 9*, 2 (dec 2007), 30–37. https://doi.org/10.1145/1345448.1345454.

[VK20] Vayansky I., Kumar S. A.: A review of topic modeling methods. *Information Systems*, *94* (2020), 101582. https://doi.org/10.1016/j.is.2020.101582.

[vWvS99] van Wijk J. J., van Selow E. R.: Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis'99)* (Oct. 1999), pp. 4–9.

[WBM*21] Wulms J., Buchmüller J., Meulemans W., Verbeek K., Speckmann B.: Stable visual summaries for trajectory collections. In *IEEE 14th Pacific Visualization Symposium (PacificVis)* (2021), pp. 61–70. https://doi.org/10.1109/PacificVis52677.2021.00016.

[WG11] Ward M. O., Guo Z.: Visual exploration of time-series data with shape space projections. *Computer Graphics Forum 30*, 3 (2011), 701–710. https://doi.org/10.1111/j.1467-8659.2011.01919.x.

[WG12] Wongsuphasawat K., Gotz D.: Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2659–2668. https://doi.org/10.1109/TVCG.2012.225.

[WGGP*11] Wongsuphasawat K., Guerra Gómez J. A., Plaisant C., Wang T. D., Taieb-Maimon M., Shneiderman B.: Lifeflow: Visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2011), CHI '11, Association for Computing Machinery, p. 1747–1756. https://doi.org/10.1145/1978942.1979196.

[ZZYG21] Zhou F., Zhou H., Yang Z., Gu L.: If2cnn: Towards non-stationary time series feature extraction by integrating iterative filtering and convolutional neural networks. *Expert Systems with Applications 170* (2021), 114527. https://doi.org/10.1016/j.eswa.2020.114527.

Appendix

Table A1: *Breaks in the division of the attribute value ranges into bins. Case study 1: COVID-19 mobility trends*

Attribute	0 *(min)*	1	2	3	4	5 *(max)*
Retail & recreation	−100	−40	−22	−10	1	143
Grocery & pharmacy	−98	−13	−3	4	16	170
Parks	−93	−23	−3	16	52	646
Transit stations	−100	−45	−30	−17	−2	96
Workplaces	−93	−41	−28	−20	−11	45
Residential	−21	1	5	9	14	43

Table A2: *Breaks in the division of the attribute value ranges into bins in the football case study*

Attribute	0 *(min)*	1	2	3	4	5 *(max)*
Pressure on ball	0.000	1.699	11.550	23.672	40.572	173.684
Pressure on attackers	5.196	117.818	156.291	188.216	233.835	487.682
Percent attackers in opp. third	0.000	0.000	0.000	20.000	47.302	100.000
Change of ball distance to defense goal	−71.930	−20.194	−7.406	−0.630	8.139	48.713
Change of depth of attackers	−29.193	−3.138	−0.600	1.140	4.478	26.445
Change of width of attackers	−36.476	−1.985	0.734	3.283	8.365	33.848
Change of stretch of attackers	−7.953	−0.653	0.147	0.861	2.421	10.193
Change of min distance of attackers to opp. goal	−54.045	−7.787	−1.880	0.408	3.286	22.679
Change of mean distance of attackers to opp. goal	−49.126	−7.222	−2.161	−0.047	1.946	23.352
Change of depth of defenders	−28.273	−4.085	−1.088	1.054	4.694	25.950
Change of width of defenders	−28.794	−5.664	−1.711	0.347	2.893	17.345
Change of stretch of defenders	−9.494	−1.635	−0.476	0.186	1.160	6.928
Change of min X-distance of defenders from own goal	−44.849	−6.321	−1.207	0.854	3.822	25.177