Detecting Temporal Dependencies in Data

Joaquin Cuomo[1], Hajar Homayouni[2], Indrakshi Ray[1] and Sudipto Ghosh[1]

[1]Department of Computer Science, Colorado State University
[2]Department of Computer Science, San Diego State University

Abstract

Organizations collect data from various sources, and these datasets may have characteristics that are unknown. Selecting the appropriate statistical and machine learning algorithm for data analytical purposes benefits from understanding these characteristics, such as if it contains temporal attributes or not. This paper presents a theoretical basis for automatically determining the presence of temporal data in a dataset given no prior knowledge about its attributes. We use a method to classify an attribute as temporal, non-temporal, or hidden temporal. A hidden (grouping) temporal attribute can only be treated as temporal if its values are categorized in groups. Our method uses a Ljung-Box test for autocorrelation as well as a set of metrics we proposed based on the classification statistics. Our approach detects all temporal and hidden temporal attributes in 15 datasets from various domains.

Keywords

Dataset Management Systems, Statistics, Temporal attribute detection, Autocorrelation.

1. Introduction

Datasets can be temporal or non-temporal. A dataset is temporal if one or more attributes is a time sequence [1]. An example of a temporal dataset is a stock market dataset, in which each value of an attribute corresponds to the daily stock price. Time series normally present a time-dependency, meaning that a value is dependent on its past values. Time-series analysis has applications ranging from stock market prediction to digital signal processing and has been studied in statistics [1], econometrics [2], and in communications [3].

Data analysis techniques depend on the type of data. Techniques for non-temporal data, such as Support Vector Machine (SVM) [4] and Isolation Forest (IF) [5] only discover associations among attributes of individual data records and cannot be used for analyzing time-series data because associations may exist among multiple records in a time series [6]. Other approaches, such as Autoregressive Moving Average (ARIMA) [7] and Long Short-Term Memory (LSTM) [8], are more suitable for either prediction or optimization for temporal data analysis [9] techniques.

It is critical to understand the existence of temporal dependencies in a dataset in advance in order to choose the best analysis approach. Existing analysis approaches rely on domain experts to identify the type of data and to choose appropriate techniques to model the data. However, in big datasets, there can be a large number of attributes to be analyzed by the experts. Moreover, even domain experts may not be aware of temporal dependencies among a subset of attributes in a big dataset. An example is a health data warehouse to which temporal and non-temporal data is automatically loaded from multiple source hospitals through an automated Extract, Transform, Load (ETL) process [10]. Every patient can have a set of temporally dependent records, such as records related to their lab tests. Explicit temporal information, such as a timestamp that identifies when data is captured as well as attribute names that indicate temporal characteristics may change through the ETL transformation. For example, the name of the *Patient_Height* attribute may change into a random name through the transformation process. This data modification can make the temporal nature of the target attribute unknown to the researchers who are using the data for making critical decisions on disease, treatments, and medications.

To the best of our knowledge, there is no prior attempt on the detection of temporal dependencies in datasets. Such dependencies are presumed to be known beforehand, which works only for well-understood datasets. However, where domain experts lack adequate knowledge about the data characteristics, there is a need to automatically detect whether or not a dataset is temporal in order to choose the right technique and have a fully automated process. Our work fills this gap.

We developed a method to determine whether or not a dataset contains temporal attributes. Moreover, our approach automatically identifies grouping attributes. A grouping attribute is such by which we can group the dataset records and obtain intergroup temporal attributes but not intragroup. A dataset may have one or more grouping attributes. The proposed algorithm is based on a portmanteau test [1] for autocorrelation to

BICOD21: British International Conference on Databases, December 9–10, 2021, London, UK

jcuomo@colostate.edu (J. Cuomo); hhomayouni@sdsu.edu (H. Homayouni); iray@colostate.edu (I. Ray); ghosh@colostate.edu (S. Ghosh)

© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

determine the presence of temporal data. To find grouping attributes that yield temporal sequences we use a brute force approach by testing each unique value as a possible grouping attribute. Finally, we propose metrics that help determine whether the result of the portmanteau test should be accepted or rejected based on an integrated perspective of the dataset. We evaluated the proposed method on fifteen datasets, where each attribute was given a priori classification by domain experts. We demonstrated that our approach was able to discover all the temporal attributes.

The paper is organized as follows. Section 2 presents the theoretical background that forms the basis of our work. Section 3 discusses our proposed approach to detect temporal dependencies in datasets. Sections 4 and 5 describe our experiments and results using 15 different datasets. Section 7 concludes the paper.

2. Background

In this section we provide some background on time series analysis and autocorrelation theory, which is needed to understand the proposed method.

2.1. Time Series

A time series is a sequence of observations equally spaced and ordered by time [11]. Normally, these observations are not independent from each other because their relative order is important. This non-independence means that there is a temporal dependence implying that future values are influenced by past values. The classical approach for analyzing temporal series is to consider them as a combination of four components. This combination can be additive or multiplicative:

$$X_t = Trend + Seasonal + Cyclical + Irregular \quad (1)$$

$$X_t = Trend \cdot Seasonal \cdot Cyclical \cdot Irregular \quad (2)$$

where X_t is a temporal series.

A secular trend (*Trend* in Eqs. 1 and 2) describes the consistent tendency of the data over a long period. A seasonal variation (*Seasonal* in Eqs. 1 and 2) describes the periodic fluctuation within cycles. The cyclical component (referred to as *Cyclical* in Eqs. 1 and 2) describes to longer periodic fluctuations. The irregular component (*Irregular* in Eqs. 1 and 2) describes small changes that are unpredictable.

A time series is said to be stationary if its statistical properties do not change over time, that is, if it has constant mean and variance, and covariance is independent of time.

Finally, autocorrelation is a measure of the similarity of the observations at certain lag, that is, the correlation of the series with a delayed copy of itself. It gives critical information on whether a value in the series can be used to infer information about another value. A common way to analyze temporal data is to create a model that fits the data, and the most widespread technique is regression analysis, which uses autocorrelation [12]. Therefore, autocorrelation is going to be the most important metric to determine if a dataset has or does not have temporal dependence.

2.2. Testing Autocorrelation

Most of the literature on autocorrelation of time-series is about evaluating the fitness of an autoregressive model, which is done by analyzing the autocorrelation of the model's residuals. However, because we do not have prior knowledge about the data we are unable to apply these methods which require certain assumptions [13]. The most popular methods are Ljung-Box [14], Box-Pierce [15] and others like Breusch–Godfrey [16], Daniel-Peña [17] and Monte-Carlo [18] which overcomes some of the limitations of the first two [14, 15] but are more focused on time-series model's residuals. Both Ljung-Box and Box-Pierce methods are portmanteau tests which allows testing the autocorrelation of a time series at multiple lags at the same time. The null hypothesis of the test is that the data is independently distributed while the alternative hypothesis is that the data exhibits serial correlation up to any lag. The distribution of the tests approximates asymptotically to a χ^2 and the rejection of the null hypothesis will indicate to us that there is autocorrelation in our data.

The method that we used is Ljung-Box, which is a modification of Box-Pierce and it approximates better to a χ^2 [14]. The formula is:

$$Q(m) = n(n+2) \cdot \sum_{k=1}^{m} \frac{r_k^2}{n-k} \quad (3)$$

where n is the number of samples, m is the maximum lag to test for autocorrelation, and r is the autocorrelation.

The degree of freedom of the χ^2, when there is no other information about the data, should be equal to the number of lags up to where the autocorrelation is being tested. The choice of lag is difficult when no information about the data is known. The higher the lag the lower the performance of the test. Also, the lag should be a fraction of the sequence length. For example, the Stata implementation [19] uses the rule of m=min(n/2,40), while Box et al. [20] suggest m=20, and Tsay [21] suggests m=ln(n) warning that when seasonal behavior is expected, this behavior needs to be taken into consideration and lag values at multiples of the seasonality are more important. Escanciano and Lobato [22] present a portmanteau test

date	county	cases	deaths
2020-01-21	Snohomish	1	0.0
2020-01-22	Snohomish	1	0.0
2020-01-23	Snohomish	1	0.0
2020-01-24	Cook	1	0.0
2020-01-24	Snohomish	1	0.0
...	...	...	...
2021-02-02	Sweetwater	3510	33.0
2021-02-02	Teton	3151	7.0
2021-02-02	Uinta	1975	12.0
2021-02-02	Washakie	867	26.0
2021-02-02	Weston	611	5.0

date	county	cases	deaths
2021-01-29	Larimer	17914	192.0
2021-01-30	Larimer	17914	192.0
2021-01-31	Larimer	17914	192.0
2021-02-01	Larimer	18115	196.0
2021-02-02	Larimer	18160	198.0

date	county	cases	deaths
2021-01-29	Boulder	17225	232.0
2021-01-30	Boulder	17279	232.0
2021-01-31	Boulder	17329	232.0
2021-02-01	Boulder	17376	232.0
2021-02-02	Boulder	17433	232.0

Figure 1: Example of groups created by filtering by the attribute county's values. Left shows the entire dataset. Right shows the groups. This dataset has no obvious temporal dependent data until we do the grouping by 'county'. Only then, 'deaths' and 'cases' have temporal dependency.

that automatically chooses the lag.

3. Our Approach

Based on possible temporal characteristics, we categorized datasets into three types.

- No temporal dependence: Given a dataset with no temporal information, no autocorrelation is expected.
- A continuous evenly-sampled time-ordered dataset: Given a dataset that corresponds to a single time window, we can detect the temporal dependence by computing the autocorrelation of each attribute over the entire dataset.
- Temporal dependence within a grouping attribute: There is no observable temporal dependence when the dataset is considered as a whole, but the temporal dependence becomes apparent when grouped by some attribute. In such a case, we can detect the temporal dependence by computing the autocorrelation of each attribute within each group. Finding the proper grouping attribute is the main challenge in this case.

Figure 1 exemplifies the third case, where a dataset may have hidden temporal dependencies that are uncovered once the proper attribute is used to form groups. On the left, the entire dataset does not exhibit any autocorrelation for any of the attributes. On the other hand, on the right, after grouping by attribute 'county' the attributes 'deaths' and 'cases' correspond to temporal series.

Our Algorithm

We proposed an algorithm that aims to detect the data with temporal dependency. In order to do this, we split the algorithm in two stages, A and B, as shown in Figure 2.

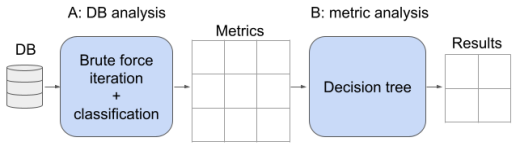

Figure 2: High-level overview of the proposed method.

In stage A, we do nested iterations over all the numeric attributes and all their unique values. We group the dataset by those values and classify all other attributes as time-dependent or not. As an example, using dataset from Figure 1, while we are at the iteration of the attribute 'county', we group by 'county', and for each group, we classify the other attributes ('date', 'cases', and 'deaths') as temporal or not. The following pseudo code describes the process, which computes a set of metrics we analyze in stage B using a decision tree to determine the temporal attributes.

```
for each attribute A do
    for each unique value x of A do
        smallDB = SELECT * WHERE A = x;
        classification(smallDB);
    end
end
```

The classification part of stage A is diagrammed in Figure 3. It consists of analyzing a single attribute and determining if it has autocorrelation. We do a Ljung-Box test to detect statistically significant autocorrelation. In parallel, we apply a threshold (0.5 in our examples) to determine if the autocorrelation is also quantitatively significant for the specific posterior use of the dataset. If both tests pass, we consider the sequence to have temporal dependency.

The metrics outputted on stage A consists of a table showing statistics of all the classification when grouping the dataset by each attribute. The rows are the attributes of the dataset and the columns are the metrics described in Table 1. To address the first and second types of dataset described at the beginning of this section, we add a row consisting of no-grouping-by-any-attribute, where we show the classification of the attributes if no grouping is done. As an example of how the metrics are computed, let us consider the dataset from Figure 1. First, we group by 'date' and classify each attribute as temporal or not. In this case, in none of the groups the attributes were classified as temporal. Next, we group by 'county' and

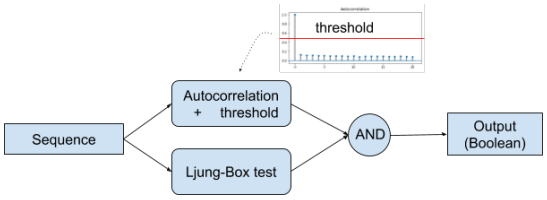

Figure 3: Diagram of proposed classification algorithm.

classify 'cases' and 'deaths' as temporal ('date' is not considered as it is not a numerical attribute). As some particular counties might have only 'cases' being classified as temporal, the average of attributes detected as temporal when grouping by 'county' is less than 2. Figure 4 shows the average is 1.93 in the resulting table.

Table 1
Metrics

Name	Description
% data	Percentage of records from groups with at least one attribute classified as temporal over the entire dataset.
groups	Count of groups with at least one attribute classified as temporal.
avg_temp_att	Average of the count of attributes classified as temporal over the groups.
std	Standard deviation of avg_temp_att.
avg_corr	Average of maximums autocorrelations over all groups. Maximum values are calculated within a group, over all attributes classified as temporal.
max_corr	Maximum autocorrelation over all attributes and groups.

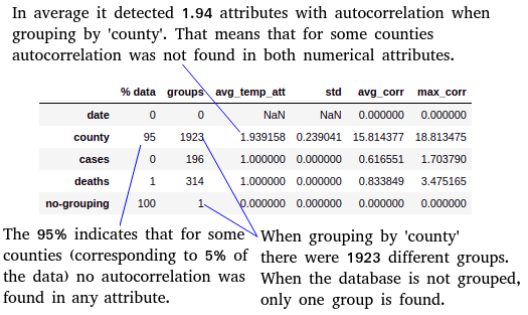

Figure 4: Example of the analysis of the metrics using example from Figure 1.

Stage B consists of analyzing the metrics from the resulting table to determine if grouping by attributes generates temporal sequences. We designed a decision tree, shown in Figure 5 to guide the analysis of the table. The tree first discards attributes with small percentage of data used, as they are considered not representative. Similarly, the attributes that produced only one group (or none) are discarded as they do not produce multiple groups with temporal dependence. Based on the definition of the metrics, only groups with some autocorrelation are being counted. For example, in Figure 6, when grouping by the 'date' attribute, none of the resulting groups presents autocorrelation. As a result, the group count metric for 'date' is equal to 0. Next, the average count of attributes with detected autocorrelation is used to discard attributes, where the larger is preferred (as far as the standard deviation is small). This condition is the primary metric to analyze the attributes, as it values more the groups that in average have more attributes with autocorrelation. Additionally, the average of the autocorrelation of each group is evaluated and those with the highest values are considered.

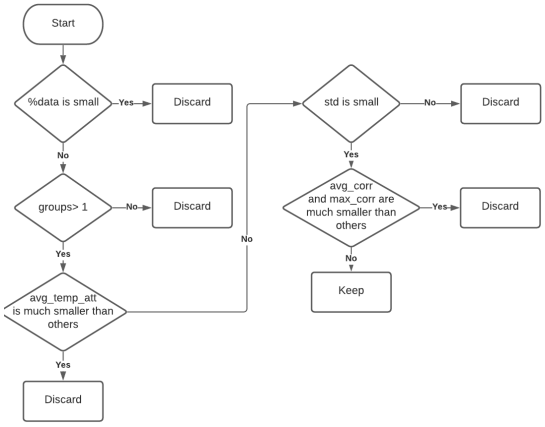

Figure 5: Decision tree to analyze results.

The final result consists of the attributes with temporal dependence along with the percentage of times it was detected as temporal over all groups.If the percentage is lower than 50% we don't consider that attribute as temporal for further analysis. Figure 6 shows an example of this result when grouping by the 'county' attribute. In this example, both 'cases' and 'death' attributes have autocorrelation and were detected as temporal in more than 50% of times over all groups. As a result, both attributes are considered as temporal if we group the dataset by 'county'.

4. Experiments

We conduct different experiments to show how the metrics we defined can help determine if there is temporal dependence in the dataset. We run the algorithm against

	Ocurrences [%]
cases	100.000000
deaths	93.915757

Figure 6: An example of attributes with temporal dependence.

15 datasets, five for each of the three categories mentioned in Section 3. For each of these categories, we explain one dataset in detail as an example. In the following, we describe the research questions that we answer through our experiments.

Q1: Can the proposed approach correctly identify attributes with temporal dependence in the datasets?

We answer this question using the domain knowledge. Domain experts label an attribute as positive or negative depending on whether or not the attribute has temporal dependency. We construct a contingency matrix and calculate the accuracy (eq. 5) and the F1-score (eq. 4) for each dataset. We include the accuracy because the F1-score is not applicable for the cases where there are no temporal attributes.

$$F1 = \frac{TP}{TP + 1/2(FP + FN)} \tag{4}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

Q2: Can the approach correctly identify grouping attributes to form multiple temporally dependent sequences?

Using same evaluation metrics, we analyzed if the approach can correctly identify attributes by which we can group the dataset records into multiple temporally-dependent sequences. For each dataset, we first identify if it has such an attribute. Typically, there are multiple possible grouping attributes. For example, in a dataset containing information about suicides all over the world, a grouping attribute can be each country, but at the same time, there could be trends related to other attributes, such as gender. Therefore, we do not do a specific analysis on each of the possible grouping attributes, but we limit the analysis to the existence or not of any. The F1-score is calculated for all the datasets.

All the datasets used in this study are publicly available and were picked to exemplify various categories (Appendix B).

5. Results

In this section, we first present the summary of the results for each dataset. Then, we explore with more details some examples for each specific case.

Figure 7 shows the scores for each dataset. Each row corresponds to a dataset, where the first five (election, incomes, countries, biomechanical, and crime) do not have temporal dependence. The next ten have temporal dependence, but only the last five have grouping attributes that produce temporal sequences (covid2, wage, market, avocado, and suicides), while the five in the middle do not (codiv1, energy1, yahoo, traffic, india). Each of these cases is indicated by column 'case' and the numbers 0,1,2 correspond to the same order of categories explained in Section 3. The columns 'FP', 'TP', 'FN', and 'TN' count the number of attributes that have been classified as temporal or not and are false positive, true positive, false negative, and true negative respectively. The columns 'ACC' and 'F1' are the accuracy and the F1-score of the classification respectively. Following, '# temp att detected' is the ratio the attributes that were correctly detected over the total number of temporal attributes. The next three columns refers to the detection of the grouping attributes, where 'grouping' indicates if the dataset has one or more grouping attributes, 'grouping detected' if the algorithm found any, and 'contingency' specifies the type of error or success. Finally, the bottom part of the table summarizes the 'contingency' column and shows the F1-score for the detection of grouping attributes.

For case 0, in two of the five datasets, there were attributes that were detected as temporal. As there is no temporal attribute, the F1-score is not applicable for this case, so only the accuracy should be taken into account. In the entire table, these are the only two cases with accuracy and F1-score (when applicable) lower than 1. In none of the datasets, an attribute was falsely considered as a grouping attribute. Moreover, in all datasets with grouping attributes, those attributes were successfully found, yielding a F1-score of 1.

5.1. No Temporal Dependencies Datasets

To exemplify this case, we have the 'elections' dataset, which consists of reported votes by county in the governor race in the US elections 2020 (Figure 8). It has 1025 entries, 2 non-numeric attributes, and 3 numerical attributes, none of which has temporal dependence.

Figure 9 shows that no autocorrelation was found, as expected.

One of the limitations of using autocorrelation, as we will discuss in Section 6.1, is that other types of relationships can also produce correlation. To illustrate this, we used the 'biomechanical' dataset (Figure 10), for which there are two false positives based on the result table of Figure 7. The dataset consist of six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine of 310 patients. Despite the lack of temporal dependence in the data, the results, as shown in Figure 11, indicates the presence of autocorrelation

	case	FP	TP	FN	TN	ACC	F1	# temp att detected	grouping	grouping detected	Contingency
elections	0	0	0	0	3	1	N/A	0/0	False	False	TN
incomes	0	0	0	0	3	1	N/A	0/0	False	False	TN
countries	0	2	0	0	16	0.88	0	2/0	False	False	TN
biomechanical	0	2	0	0	4	0.66	0	2/0	False	False	TN
crime	0	0	0	0	13	1	N/A	0/0	False	False	TN
covid1	1	0	2	0	0	1	1	2/2	False	False	TN
energy1	1	0	4	0	0	1	1	4/4	False	False	TN
yahoo	1	0	101	0	0	1	1	101/101	False	False	TN
india	1	0	2	0	0	1	1	2/2	False	False	TN
exchange	1	0	8	0	0	1	1	8/8	False	False	TN
covid2	2	0	2	0	1	1	1	2/2	True	True	TP
wage	2	0	12	0	0	1	1	12/12	True	True	TP
market	2	0	5	0	0	1	1	5/5	True	True	TP
avocado	2	0	11	0	0	1	1	11/11	True	True	TP
suicides	2	0	6	0	0	1	1	6/6	True	True	TP

	F1	FN	FP	TN	TP
grouping	1.0	0.0	0.0	10.0	5.0

Figure 7: Results summary for all datasets. Case 0, 1, and 2 corresponds to no-temporal information, no-grouping temporal information, and grouping temporal information.

state	county	current_votes	total_votes	percent
Delaware	Kent County	85415	87025	100
Delaware	New Castle County	280039	287633	100
Delaware	Sussex County	127181	129352	100
Indiana	Adams County	14154	14209	100
Indiana	Allen County	168312	169082	100

Figure 8: US elections dataset

	% data	groups	avg_temp_att	std	avg_corr	max_corr
state	0	0	NaN	NaN	0.0	0
county	0	0	NaN	NaN	0.0	0
current_votes	0	0	NaN	NaN	0.0	0
total_votes	0	0	NaN	NaN	0.0	0
percent	0	0	NaN	NaN	0.0	0
no-grouping	100	1	0.0	0.0	0.0	0

Figure 9: US elections dataset grouping attribute results

in 2 out of the 5 numerical attributes, namely, 'lumbar lordosis angle' and 'degree spondylolisthesis'.

5.2. No-grouping Temporal Datasets

The 'covid1' dataset, shown in Figure 12 has daily information about positive cases and deaths caused by COVID-19 in the United States.

The results in Figure 13 shows that no-grouping is the best option, which is correct as none of the attributes allows to form groups. The detected temporal attributes, as expected are both the number of cases and deaths, as shown in Figure 14.

5.3. Temporal Dependencies within Grouping Attributes

To illustrate this case, we used the 'covid2' dataset from Figure 15, which consists of daily deaths and positive cases of COVID-19 by county in the United States. There are two differences between 'covid1' and 'covid2' datasets. First, there are two non-numeric attributes corresponding to counties and states, which can be used to establish a geographical relation between the records. Second, there is a numerical attribute 'FIPS', which is a code to identify counties and states. Therefore, we expect this attribute not to have autocorrelation, but to be a potential grouping attribute.

The results in Figure 16 show that if we do not split the dataset in groups, none of the attributes can be considered to have temporal dependence. Instead, if we group by 'county' or 'fips' there are 2 attributes in average with autocorrelation. Despite that grouping by the attributes 'date', 'state', 'cases', 'fips' have non-zero values in the resulting table, the percentage of used data is low. Thus, we ignore grouping by these attributes. Figure 17 shows, that when grouping by 'county' the attributes 'cases', 'deaths', and 'fips' could be considered as temporal sequences. Nevertheless, we discard 'fips' as it has an occurrence of approx. 22%, which is lower than our defined threshold of 50%.

pelvic_incidence	pelvic_tilt	lumbar_lordosis_angle	sacral_slope	pelvic_radius	degree_spondylolisthesis	class
63.027818	22.552586	39.609117	40.475232	98.672917	-0.254400	Hernia
39.056951	10.060991	25.015378	28.995960	114.405425	4.564259	Hernia
68.832021	22.218482	50.092194	46.613539	105.985135	-3.530317	Hernia
69.297008	24.652878	44.311238	44.644130	101.868495	11.211523	Hernia
49.712859	9.652075	28.317406	40.060784	108.168725	7.918501	Hernia

Figure 10: Biomechanical dataset

	% data	groups	avg_temp_att	std	avg_corr	max_corr
pelvic_incidence	0	0	NaN	NaN	0.000000	0.000000
pelvic_tilt	0	0	NaN	NaN	0.000000	0.000000
lumbar_lordosis_angle	0	0	NaN	NaN	0.000000	0.000000
sacral_slope	0	0	NaN	NaN	0.000000	0.000000
pelvic_radius	0	0	NaN	NaN	0.000000	0.000000
degree_spondylolisthesis	0	0	NaN	NaN	0.000000	0.000000
class	0	0	NaN	NaN	0.000000	0.000000
no-grouping	100	1	2.0	0.0	0.506095	0.506132

Figure 11: Biomechanical dataset temporal attribute results

date	cases	deaths
2020-01-21	1	0
2020-01-22	1	0
2020-01-23	1	0
2020-01-24	2	0
2020-01-25	3	0
...	...	...
2021-01-29	25971349	436780
2021-01-30	26105263	439421
2021-01-31	26218775	441285
2021-02-01	26358607	443235
2021-02-02	26472841	446643

Figure 12: Covid-19 in the US dataset

	% data	groups	avg_temp_att	std	avg_corr	max_corr
date	0	0	NaN	NaN	0.000000	0.000000
cases	0	0	NaN	NaN	0.000000	0.000000
deaths	10	1	1.0	0.0	0.791398	1.582797
no-grouping	100	1	2.0	0.0	17.774385	17.930101

Figure 13: Covid-19 in the US dataset grouping attribute results

	Ocurrences [%]
cases	100.0
deaths	100.0

Figure 14: Covid-19 in the US dataset temporal attribute results

date	county	state	fips	cases	deaths
2020-01-21	Snohomish	Washington	53061.0	1	0.0
2020-01-22	Snohomish	Washington	53061.0	1	0.0
2020-01-23	Snohomish	Washington	53061.0	1	0.0
2020-01-24	Cook	Illinois	17031.0	1	0.0
2020-01-24	Snohomish	Washington	53061.0	1	0.0
...	...	...	...	...	...
2021-02-02	Sweetwater	Wyoming	56037.0	3510	33.0
2021-02-02	Teton	Wyoming	56039.0	3151	7.0
2021-02-02	Uinta	Wyoming	56041.0	1975	12.0
2021-02-02	Washakie	Wyoming	56043.0	867	26.0
2021-02-02	Weston	Wyoming	56045.0	611	5.0

Figure 15: Covid-19 in the US by county and state dataset

	% data	groups	avg_temp_att	std	avg_corr	max_corr
date	0	18	1.000000	0.000000	0.820709	1.681197
county	99	1927	2.159315	0.506363	16.542085	19.672386
state	9	19	1.842105	0.364642	4.278340	18.517124
fips	99	3216	1.952736	0.212202	17.516142	18.813475
cases	0	361	1.022161	0.147206	0.634591	1.703790
deaths	1	345	1.066667	0.249444	0.817670	3.475165
no-grouping	100	1	0.000000	0.000000	0.000000	0.000000

Figure 16: Covid-19 in the US by county and state grouping attribute results

	Ocurrences [%]
cases	99.792423
deaths	93.720810
fips	22.418267

Figure 17: Covid-19 in the US by county and state temporal attribute results

6. Discussion and Future Research

We proposed an approach to classify datasets based on whether or not they contain temporally dependent data. The core of our algorithm is based on the autocorrelation as the method to determine if there is a temporal dependence in a section of the data. Our algorithm relies on a

set of proposed metrics to integrally classify the dataset. Among these metrics, the percentage of data used in the analysis, the number of groups, and the average of autocorrelated sequences found were the three metrics that provided the most relevant information for making a decision. The other metrics were not used in any of the examples but we believe that they could come handy in larger datasets. For example, the standard deviation should not be too large as it would mean that there is a particular grouping attribute value with more temporal sequences than the rest, which is probably as a result of an outlier, and should be handled carefully to avoid a false positive. Both the average autocorrelation and the maximum autocorrelation are used as tiebreakers when the other metrics have same values.

Our approach could identify temporal sequences, when the sequence corresponds to the entire dataset, and also when grouping by attributes was needed. Typical datasets fall in both cases, meaning that an attribute can present autocorrelation as a whole sequence and as multiple grouped subsequences. The latter case is important because it allows to improve the data analysis. For example, if we have an outlier detection algorithm for temporal data, we may apply that to a single sequence as well as to different subsequences constructed from the same data, to increase the chance of detecting more outliers. Another use case is when the algorithm has high time complexity. In such a case, it may be better to only explore the outliers in the smaller subsequences than in the entire sequence.

6.1. Limitations

We identify the following scenarios where our approach might failed to detect temporal dependency on the attributes.

- Small sample size: when the number of samples is small, no statistical test will have enough significance.
- Unevenly-sampled data: when there is no constant time-spacing between samples. If the uneven sampling is due to missing data points and the sample size is large enough, the approach should converge to the same values as if all data points were present. However, if there is no pattern in the sampling rate, different methods should be used to calculate the autocorrelation indirectly, such as estimating the autocorrelation using the statistical approaches [23].
- Missing values: when there are null values in the data. there are many methods [24] to overcome missing values in time-series data and specifically for the Ljung-Box test[25]. However, under the assumption that we do not have prior information on the data, none of these methods can be used.
- Cross-sectional data: when there are dependence other than temporal between attribute values. Even though autocorrelation is a necessary condition to exploit temporal data information, it is not a sufficient condition to determine if the data is temporal. For example, our method will fail when a dataset has correlations that are not temporal but spatial [26].
- Non-stationarity: when time-series statistical properties vary over time. In such cases, the autocorrelation cannot be calculated using the mean and the variance but needs to be estimated. Similar methods could be used as when dealing with missing values [27].

The decision tree to analyze the metrics is currently not automated as we require a higher volume of use cases to generalize the rules. Similarly, for tuning the hyperparameters, such as the autocorrelation threshold we used to determine if an autocorrelation was significant, we need more extensive analysis and cases.

6.2. Future work

Statistical exploration and optimization We will investigate whether different types of correlations, such as Pearson, Kendall, Spearman, and estimation from the power spectral density can be used within the Ljung-Box or Box-Pierce test [28]. We will conduct a deep analysis on which autocorrelation function to use when no prior information on the data is known. Currently, the algorithm goes over all numeric attributes searching for autocorrelation. This is time consuming and should be, if possible, improved.

Working with categorical attributes Datasets may consist of categorical attributes, such as boolean labels, names, IDs, and dates. These attributes may be temporal as well. For example, a positive value for a patient with a non-curable disease is unlikely to become negative in the future. Thus, finding a way to process such attributes is important. We will use one-hot encoding to pre-process the categorical attributes.

7. Conclusions

In this paper, we have presented a technique that uses autocorrelation to determine the presence of temporal data within its attributes without any prior knowledge about the database. The algorithm was tested for different databases, including those with and without temporal dependence data, and specifically focused on databases containing hidden temporal groups. For these cases, we proposed metrics to find the grouping attributes that

unveil such hidden groups. The results show that we were able to successfully classified attributes as temporal or not, and also to find grouping attributes that form temporal groups. Finally, we discussed the limitations of the approach and potential improvement paths.

Acknowledgments

This work was supported in part by funding from NSF under Award Numbers CNS 1822118, IIS 2027750, OAC 1931363, Statnett, ARL, AMI, Cyber Risk Research, and NIST.

References

[1] P. J. Brockwell, R. A. Davis, Introduction to Time Series and Forecasting, Springer, 2008.

[2] H. Luetkepohl, M. Krätzig, In Applied Time Series Econometrics, Applied Time Series Econometrics (2004).

[3] M. Allen, The SAGE Encyclopedia of Communication Research Methods, SAGE Publications, 2017.

[4] Y. Chen, W. Wu, Application of One-class Support Vector Machine to Quickly Identify Multivariate Anomalies from Geochemical Exploration Data, Geochemistry: Exploration, Environment, Analysis 17 (2017) 231–238.

[5] Z. Cheng, C. Zou, J. Dong, Outlier Detection Using Isolation Forest and Local Outlier Factor, in: Conference on Research in Adaptive and Convergent Systems, Association for Computing Machinery, 2019, p. 161–168.

[6] H. Lu, Y. Liu, Z. Fei, C. Guan, An Outlier Detection Algorithm based on Cross-Correlation Analysis for Time Series Dataset, IEEE Access 6 (2018) 53593–53610.

[7] P. M. Maçaira, A. M. T. Thomé, F. L. C. Oliveira, A. L. C. Ferrer, Time Series Analysis with Explanatory Variables: A Systematic Literature Review, Environmental Modelling & Software 107 (2018) 199 – 209.

[8] Y. Yu, X. Si, C. Hu, J. Zhang, A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures, Neural Computation 31 (2019) 1235–1270.

[9] W. Lin, M. Orgun, G. Williams, An overview of temporal data mining (2019).

[10] H. Homayouni, S. Ghosh, I. Ray, An Approach for Testing the Extract-Transform-Load Process in Data Warehouse Systems, in: Proceedings of the 22nd International Database Engineering and Applications Symposium, IDEAS 2018, Association for Computing Machinery, 2018, p. 236–245.

[11] W. Wei, Time Series Analysis: Univariate and Multivariate Methods, volume 33, 1989.

[12] A. Dotis-Georgiou, Autocorrelation in Time-series Data, 2019. URL: https://www.influxdata.com/blog/autocorrelation-in-time-series-data/, influx Data article, accessed 25th July 2021.

[13] G. Maddala, Introduction to Econometrics, Wiley, 2001.

[14] G. Ljung, G. Box, On a Measure of Lack of Fit in Time Series Models, Biometrika 65 (1978).

[15] G. Box, D. Pierce, Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models, Journal of the American Statistical Association 72 (1970) 397–402.

[16] D. Scott, Applied Econometrics with R by Christian Kleiber, Achim Zeileis, International Statistical Review 77 (2009) 164–164.

[17] D. Peña, J. Rodríguez, A Powerful Portmanteau Test of Lack of Fit for Time Series, Journal of the American Statistical Association 97 (2002) 601–610.

[18] J.-M. Dufour, L. Khalaf, Monte Carlo Test Methods in Econometrics, 2007.

[19] S. Documentation, wntestq Portmanteau (Q) Test Description, 2019. URL: http://www.stata.com/manuals13/tswntestq.pdf, accessed on 24th July 2021.

[20] E. Ziegel, G. Box, G. Jenkins, G. Reinsel, Time series analysis, forecasting, and control, Technometrics 37 (1995) 238.

[21] R. Tsay, Analysis of Financial Time Series. Financial Econometrics, 2002.

[22] J. C. Escanciano, I. N. Lobato, An automatic Portmanteau Test for Serial Correlation, Journal of Econometrics 151 (2009) 140–149.

[23] K. Rehfeld, N. Marwan, J. Heitzig, J. Kurths, Comparison of Correlation Analysis Techniques for Irregularly Sampled Time Series, Nonlinear Processes in Geophysics 18 (2011) 389–404.

[24] I. Pratama, A. E. Permanasari, I. Ardiyanto, R. Indrayani, A Review of Missing Values Handling Methods on Time-series Data, in: 2016 International Conference on Information Technology Systems and Innovation (ICITSI), 2016, pp. 1–6.

[25] D. Stoffer, C. Toloi, A Note on the Ljung–Box–Pierce Portmanteau Statistic with Missing Data, Statistics & Probability Letters 13 (1992) 391–396.

[26] A. Zuur, Spatial Correlation, 2019. URL: http://userwww.sfsu.edu/efc/classes/biol710/spatial/spat-auto.htm, san Fransisco State University article, accessed on 24th June 2021.

[27] G. P. Nason, R. Von Sachs, G. Kroisandt, Wavelet processes and adaptive estimation of the evolutionary wavelet spectrum, Journal of the Royal Statistical Society Series B 62 (2000) 271–292.

[28] C. Chatfield, The Analysis of Time Series: An Introduction, Fourth Edition, Chapman & Hall/CRC

Texts in Statistical Science, CRC Press, 1989.

A. Code

The code used for this paper is available in GitHub:
https://github.com/JCuomo/TemporalDependenceDB

B. Datasets

- **elections**
  https://www.kaggle.com/unanimad/us-election-2020
  "governors county" file.
  General information about reporting votes to governor race by county.
- **incomes**
  https://www.kaggle.com/jonavery/incomes-by-career-and-gender
  American citizens incomes from 2015 broken into male and female statistics.
- **countries**
  https://www.kaggle.com/fernandol/countries-of-the-world
  Information on population, region, area size, infant mortality and more.
- **biomechanical**
  https://www.kaggle.com/uciml/biomechanical-features-of-orthopedic-patients
  Patient data of six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine.
- **crime**
  https://www.kaggle.com/mascotinme/population-against-crime
  FBI crime statistics for 2012 on population less than 250,000.
- **covid1**
  https://raw.githubusercontent.com/nytimes/covid-19-data/master/us.csv
  Covid cases and death statistics for USA.
- **energy**
  This dataset is proprietary and cannot be distributed.
  Daily energy delivery by Fort Collins power facility.
- **yahoo**
  https://webscope.sandbox.yahoo.com/
  "A3Benchmark all" file
  Real and synthetic time-series. The synthetic dataset consists of time-series with varying trend, noise and seasonality. The real dataset consists of time-series representing the metrics of various Yahoo services.
- **india**
  https://www.kaggle.com/muralimunna18/india-population
  Population of india by year.
- **exchange**
  https://www.kaggle.com/rohithbollareddy/foreign-exchange-in-india-yearlysource-rbi
  Exchange currencies by year.
- **covid2**
  https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-counties.csv
  Covid cases and death by County in the USA.
- **wage**
  https://kaggle.com/lislejoem/us-minimum-wage-by-state-from-1968-to-2017
  USA minimum wage by State from 1968 to 2020.
- **market**
  https://raw.githubusercontent.com/selva86/datasets/master/MarketArrivals.csv
  Indian markets quantity and price per year.
- **avocado**
  https://www.kaggle.com/neuromusic/avocado-prices
  Avocado weekly 2018 retail scan data for National retail volume (units) and price.
- **suicides**
  https://www.kaggle.com/russellyates88/suicide-rates-overview-1985-to-2016
  Worldwide suicide statistics per year.

Table 2
Description of Datasets

DB	Link	Description
elections	https://www.kaggle.com/unanimad/us-election-2020 "governors county" file.	General information about reporting votes to governor race by county.
incomes	https://www.kaggle.com/jonavery/incomes-by-career-and-gender	American citizens incomes from 2015 broken into male and female statistics.
countries	https://www.kaggle.com/fernandol/countries-of-the-world	Information on population, region, area size, infant mortality and more.
biomechanical	https://www.kaggle.com/uciml/biomechanical-features-of-orthopedic-patients	Patient data of six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine.
crime	https://www.kaggle.com/mascotinme/population-against-crime	FBI crime statistics for 2012 on population less than 250,000.
covid1	https://raw.githubusercontent.com/nytimes/covid-19-data/master/us.csv	Covid cases and death statistics for USA.
energy	This dataset is proprietary and cannot be distributed.	Daily energy delivery by Fort Collins power facility.
yahoo	https://webscope.sandbox.yahoo.com/ "A3Benchmark all" file	Real and synthetic time-series. The synthetic dataset consists of time-series with varying trend, noise and seasonality. The real dataset consists of time-series representing the metrics of various Yahoo services.
india	https://www.kaggle.com/muralimunna18/india-population	Population of india by year.
exchange	https://www.kaggle.com/rohithbollareddy/foreign-exchange-in-india-yearlysource-rbi	Exchange currencies by year.
covid2	https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-counties.csv	Covid cases and death by County in the USA.
wage	https://kaggle.com/lislejoem/us-minimum-wage-by-state-from-1968-to-2017	USA minimum wage by State from 1968 to 2020.
market	https://raw.githubusercontent.com/selva86/datasets/master/MarketArrivals.csv	Indian markets quantity and price per year.
avocado	https://www.kaggle.com/neuromusic/avocado-prices	Avocado weekly 2018 retail scan data for National retail volume (units) and price.
suicides	https://www.kaggle.com/russellyates88/suicide-rates-overview-1985-to-2016	Worldwide suicide statistics per year.