ISSN: 2994- 9459

Research Article

Current Research in Statistics & Mathematics

Exploring Time Series Randomness

Pierpaolo Massoli*

Directorate for Methodology and Statistical Process Design (DCME), Italian National Institute of Statistics (ISTAT), Italy.

***Corresponding Author**
Pierpaolo Massoli, Directorate for Methodology and Statistical Process Design (DCME), Italian National Institute of Statistics (ISTAT), Italy.

Submitted: 2024, Mar 04; **Accepted**: 2024, Apr 08; **Published**: 2024, Apr 22

Citation: Massoli, P. (2024). Exploring Time Series Randomness. *Curr Res Stat Math, 3*(1), 01-07.

Abstract
Assessing the randomness within time series becomes challenging in the case of large-scale datasets. This novel approach leverages the efficiency of Locality Sensitive Hashing in detecting the repeating patterns over time as well as different time series. By breaking each time series down into pre-defined blocks, the solution set consists of pairs of similar blocks in accordance with the metric the proposed method approximates. As a consequence, the estimation of the aforementioned randomness turns into a pattern recognition problem, insofar as the more patterns are repeated over time, the more predictable the data becomes. Therefore, a simple measurement of the overall randomness of the time series in the input dataset is obtained by counting the identified similar blocks. Following the detection of similar patterns, the mutual information exchanged across the blocks of every detected pair is investigated to validate the results. A case study concerning a selection of different financial market indices is discussed to evaluate the potential of the proposed algorithm.

Keywords: Time Series Randomness, Locality Sensitive Hashing, Random Projections, Mutual Information, Pattern Recognition

1. Introduction

Time series is an extensive topic which covers various fields including behavioural sciences, finance, engineering, medicine and environmental monitoring. Evaluating randomness in such data can be challenging, particularly with large-scale datasets. State-of-the-art methods for analyzing time series often struggle to cope with the dimensions of the typical statistical registers which contain them. There are various approaches to testing for randomness, depending on the context of the data. From the point of view of statistical analysis, various non-parametric tests have been introduced to evaluate the degree of disorder in time series, such as runs testing [1, 2]. Symbolic dynamics and the more general communication theory are widely adopted to device tests for randomness which do not assume any probability distribution hypotheses [3-5]. Testing for the randomness of multiple time series becomes an increasingly complicated task as the number of time series grows as is the case of many real-world applications. This key-aspect is reported in where the randomness of financial markets is investigated. Another popular approach for evaluating the complexity of time series is given by the concept of entropy which is used as a measure of the degree of regularity of the time series by catching if the variations over time arise from a random process [6-8]. High values of entropy correspond to chaotic time series whereas low values of entropy indicate a high degree of regularity. A very popular algorithm for calculating the approximate entropy has been proposed by Pincus [9] while a faster sample entropy has been proposed by Richman and Moorman [10]. In this direction, mutual information can be used as a measure of association between random variables to capture the non-linear dependency between time series as in the case of financial markets [11, 12]. Although these methods are widespread, the use of non-parametric tests as well as entropy-based estimations simply provide an overall measure of the degree of disorder which is rather useless for detecting recurring patterns over time embedded into the time series. Recurring patterns in time series are data structures which are repeated at regular intervals within a time series. Patterns such as: trend, seasonality and periodicity running through the whole time series can provide valuable insights into the behaviours of the underlying phenomena over time. Relevant research regarding the pattern recognition topic has been carried out by Tanake et al. [13] who investigate the innermost structure of time series for identifying characteristic patterns called motifs. In a deep learning context, a deep convolutional neural network architecture is proposed by Zhang et al.[14] for multivariatetime series clustering. The problem of finding recurring patterns is approached by means of a deep belief network algorithm [15]. In order to detect similarities between time series another increasingly popular approach to get rid of this curse of dimensionality in the case of large-scale data is the Locality Sensitive Hashing (LSH) approach. This is a grounded tool that approximates the search for similar entities in high-dimensional spaces by mapping data into integers by using suitable hash functions [16]. In virtue of its computational efficiency, the LSH has been addressed to the problem of finding similar time series by approximating the well-known Dynamic Warping Distance [17, 18].

The patterns to be identified in this study are sequences with a length shorter than the length of the time series. These patterns called blocks are obtained by breaking the time series down into sequences, each of the same length in accordance with a pre-defined time window. This is achieved by shifting each sequence one element at a time. The objective is to detect similarities between such blocks within the time series dataset which, subsequent to the subdivision into blocks, has become even larger in size. The exploration of the randomness within a a time series dataset is faced by identifying repeated patterns inside similar blocks as they reveal local behaviours to investigate. As a consequence, the use of the aforementioned algorthms may be ineffective necessitatting more suitable methods for this purpose. On the basis of the LSH, this study aims to explore the randomness by detecting the near-duplicate ordered pairs of blocks by leveraging the LSH potential. Subsequently, the mutual information for each pair of patterns belonging to the solution set is calculated to evaluate the goodness of the solution. Finally, the count of the aforementioned pairs is summarized using a well-known composite index method to identify which time series contains the greatest number of similar blocks from the past.

2. Notations and Theoretical Background

For the convenience of the reader, the notation used in the following is listed below.

T	length of the time series
N	number of time series
L	length of the time window
C	overall number of blocks in the input dataset
$\mathbf{X}$	input dataset $(C \times L)$ of sequences
$\mathbf{x}_k = \{x_{(k-1)+1}, x_{(k-1)+2}, \ldots, x_{(k-1)+L}\}$	kth sequence (block) $\mathbf{x}_k \in \mathbb{R}^L$ of the input dataset
μ_k	average of values in the sequence $\mathbf{x}_k$
$h(\cdot)$	generic hash function
$sim(\mathbf{x}_k, \mathbf{x}_h)$	similarity measure between the sequences $\mathbf{x}_k$ and $\mathbf{x}_h$
n	number of hyperplanes in the Random Projections algorithm
$\mathbf{P}_j$	projection matrix $(L \times n)$ $(j = 1, 2, \ldots, H)$
H	number of i.i.d. hash functions
B	number of bands of the LSH search for near-duplicates
R	number of hash codes in each band
$\mathbf{M}$	similar block counts matrix $(N \times N)$
σ	similarity threshold $[0, 1]$
π	probability of being a pair of duplicates
$\mathcal{H}(\cdot)$, $\mathcal{MI}(\cdot)$	entropy and mutual information
MPI	composite index

2.1 The LSH-Family of The Random Projections

In data science Locality Sensitive Hashing (LSH) refers to a method designed for an approximate similarity search in high-dimensional spaces where traditional search methods become computationally expensive. There are several metrics that LSH encompasses for finding near-duplicates by means of a suitable family of hash functions $h(\bullet)$ which establish a relation between two input data points $(x_k,x_h) \in \mathbf{X}$ and the probability of sharing the same hash code: $sim(x_k,x_h) = Pr[h(x_k) = h(x_h)]$. The choice of the hash function determines the metric to approximate. Every family associates input data to integers which are thought of as being buckets with the purpose of hashing is to group similar data points together into the same bucket so that neighboring data fall into the same bucket with a high probability while data which are likely to be distant in the input space belong to different buckets. In a database context, this facilitates the detection of pairwise similar observations in accordance with varying degrees of similarity. The LSH family known as Random Projections adopted in this study is tailored for evaluating the cosine similarity between numerical sequences This family implements the Johnson-Lindenstrauss lemma which states that data belonging to high dimensional spaces can be projected onto a lower-dimensional space nearly preserving pairwise distances. In order to carry out this task, a set of randomly generated hyperplanes in the input space is used to project every sequence onto a lower dimensional space. Each hyperplane is considered as a decision boundary so that neighboring data points are inserted into the same bucket while they are inserted into different buckets if they are not neighbors. To be more precise, by geberating a matrix $\mathbf{P}$ with elements $\{p_{ij}\} \sim \mathcal{N}(0, 1)$ which has as many rows as the dimensions of the input space and a number of columns equal to the pre-defined number n of hyperplanes the hash code of the sequence x_k is given by setting every *ith* element of the vector-matrix product $\langle x_k,\mathbf{P}\rangle$ equal to 0 if the product of the sequence and the *ith* column of the matrix is negative and equal to 1 otherwise. The distribution $\mathcal{N}(0, 1)$ denotes a standard normal distribution with mean 0 and standard deviation 1. The number of hyperplanes affects the maximum number of buckets to which the data points are associated: In practical applications the typical values that ensure hashing with a reduced number of collisions are $n = 32$ or $n = 64$. By multiplying every input sequence of L elements by a sequrnce of H randomly generated $(L\times n)$ matrices $\{\mathbf{P}_1,\mathbf{P}_2, \ldots ,\mathbf{P}_H\}$, the input dataset is transformed into a dataset of *signatures* which are sequences of H i.i.d. hash codes. As a result the input dataset is transformed into a ($L \times H$) *signature matrix* which is elaborated in the following

2.2. Near-Duplicates Search

Subsequent to the creation of the signatures matrix in order to speed up the nearduplicates search, LSH shrinks the signatures into B bands. Each band consists of R adjacent hash codes combined together so that the relation H = BR holds. Similar sequences are finally detected by sorting the ($N \times B$) *banded matrix* and sequentially scanning it B times. Every pair of consecutive signatures with at least one corresponding equal band inidcates a pair of near-duplicate input sequences. The probability of being a pair of similar objects with a similarity value σ is given by:

$$\pi = 1 - (1 - \sigma^R)^B \qquad (1)$$

It is widely reported in the literature that the LSH is an approximate method that can give rise to *false duplicates* in the solution. The rate of the same is usually controlled by an appropriate tuning process of the hyperparameters.

2.3. Mutual Information: Basic Concepts

Mutual information is a well-known measure of the dependency in time series analysis. This measure accounts for nonlinear dependencies and requires no specific theoretical probability disibution assumption in order to be estimated. Mutual information as well as correlation are measures of association between variables, but they capture different aspects of the relationship. Correlation measures the strength and direction of a linear relationship between two continuous variables while mutual information measures the amount of information exchanged between variables which captures any type of relationship. It is obvious that they are complementary measures describing different aspects of the association between two random variables (X,Y). Mutual infotmation is related to entropy $\mathcal{H}(\bullet)$ as reported below:

$$\mathcal{MI}(X,Y) = \mathcal{H}(X) + \mathcal{H}(Y) - \mathcal{H}(X,Y) \qquad (2)$$

which can be normalized by dividing it by $max(\mathcal{H}(X),\mathcal{H}(Y))$. Equation 2 indicates that mutual information gains as the degree of regularity increases, implying that observing one variable provides a better prediction in relation to the other. Therefore information sharing within time series can be investigated by estimating the mutual information exchanged by their similar detectec blocks. Randomness is higher in less probable events, i.e. high degrees of randomness may emerge, following the detection of a few pairs of similar blocks.

3 Randomness Evaluation

The proposed algorithm is developed in the steps which follow:

3.1 Similar Blocks Detection

Consider an input dataset **X** with N time series each one being T time periods long. A time window of length $L < T$ is used to break every input series down into blocks $x^{(k)} = \{x_{(k-1)+1} - \mu_k, x_{(k-1)+2} - \mu_k, \ldots, x_{(k-1)+L} - \mu_k\}$ by shifting them one period k at a time so that the total number of resulting blocks C from the input dataset is equal to $N(T - L + 1)$. The variable µk is the average of all values in the block. The thus created input dataset is transformed into the signatures matrix as described in Section 2.

3.2. Optimization of the Solution

The LSH-family of Random Projections approximates the pairwise cosine similarity between the blocks. The solution set is composed by all pairs with a high probability of being similar with a high degree of similarity. Due to the probabilistic nature of the LSH, the presence of false duplicates must be controlled by carefully selecting the parameters $\{H,B,R\}$. Their setting is generally a critical aspect of the nearest neighbors search insofar as a wrong setting could compromise the goodness of the solution. The parameters in the algorithm proposed here are therefore set to achieve an almost zero false negatives rate in opposition to a probable higher false positives rate. In order to lower the rate of false positives, the number of the pairs detected can be reduced by filtering out all the pairs whose cosine similarity is below a pre-defined threshold τ from $\mathcal{M}$.

3.3 Evaluating the Randomness

The evaluation of randomness is carried out by comparing all the blocks pertaining to the real time series in the input dataset with the blocks from a false time series entirely made up of random values (*white noise*) which is inserted into the input dataset. The basic idea is that the more the real time series move away from the white noise the more predictable they are, reflected in a higher number of pairs of similar blocks detected. Due to the high level of the artificial disorder inserted by creating a white noise time series, pairs of similar patterns between the same and the real time series are not expected, or if detected, they should be a few or exhibit low values of the exchanged mutual information. The number of pairs of similar patterns is high if two time series show similar movements over time while on the contrary the expected number of pairs is low if one of the two series (or both) shows completely random movements. This furnish a simple evaluation of the reliability of the proposed method. The easiest summary of the solution set $\mathcal{M}$ is represented by the collection of overall counts of pairs of similar blocks gathered in a square matrix **M** of order $N + 1$. The generic cell (i,j) represents the number of pairs of blocks from the past belonging to the time series i which are similar to those belonging to the time series j forward in time . An important remark is that the cosine similarity aSsumes positive as well as negative values which indicate movements in a concurrent or opposite direction respectively while mutual information is a positive measure in both cases. A further attempt to summarize the results can be achieved by averaging pwe column all the numbers of pairs belonging to the same time series. To be more precise, every row of the count matrix is ranked by using the well-known synthesis method *MPI* [19]. This index is a *penalized* mean of the values related to the statistical units for which a ranking is coomputed. The penalty of this row-wise mean depends on the variability of the values aggregated. This method requires standardized (per column) values obtained by transforming them into z-scores having mean 100 and standard deviation 10 so that the (standardized) values of MPI fall into the interval [90,110] for the 95% of cases. The aim of applying this composite index method is to highlight the predictable behaviour of all real time series with respect to white noise.

4. Application to Financial Markets

Notwithstanding the fact that the proposed approach can be adopted in any field of research, a field of certain interest is finance in which the evaluation of the uncertainty is of considerable importance. As it is known from the literature, financial markets are not perfectly efficient implying that the exploitation of past information is not useless [20]. As market efficiency is closely related to the randomness of the same, it is worth detecting the repeated patterns of price movements over time [21, 22]. On the basis of these concepts, a dataset containing various financial market indices is explored as a case study by comparing the count of the pairs concerning real time series with those generated randomly (white noise).

4.1. The Input Dataset

Consider a dataset comprising daily values of N = 27 market indices (see Table 1). All data was collected via web scraping. The downloaded time series may have covered different time periods ranging from *2000-01-03 to 2024-02-15,* resulting in varying lengths as indicated in Table 1. The time window used to divide all the series into blocks is set to L = 10 days.The presence of missing data has to be managed prior to starting the proposed method, thus, it was necessary to identify missing data in each time series. Blocks containing at least one missing value were excluded from the analysis. No further data preprocessing was carried out. For this study, the input dataset contains a total of C = 161868 blocks to be investigated for similarities. Stabdard pairwise comparisons for similar blocks would require approximately 1.31×10^{10} comparisons, rendering the search computationally infeasible.

N	Index	Description	T
1	AORD	Ordinaries	6100
2	AXJO	S1&PASX 200	6095
3	BFX	BEL 20	6164
4	BSESN	S&P BSE SENSEX	5948
5	BUK100P	Cboe UK 100	3397
6	DJI	Dow Jones Industrial Average	6069
7	FCHI	CAC 40	6167
8	FTSE	FTSE 100	6093
9	GDAXI	DAX Performance-Index	6127
10	GSPC	S&P 500	6069
11	GSPTSE	S&PTSX Composite index	6060
12	HSI	HANG SENG	5945
13	IMOEX.ME	MOEX Russia Index	2699
14	IXIC	NASDAQ Composite	6069
15	JKSE	IDX Composite	5867
16	KLSE	FTSE Bursa Malaysia KLCI	5916
17	KS11	KOSPI Composite Index	5949
18	N100	Euronext 100 Index	6170
19	N225	Nikkei 225	5911
20	NYA	NYSE Composite (DJ)	6069
21	NZ50	S&PNZX 50 Index Gross & Gross	5204
22	RUT	Russell	6069
23	STI	STI Index	6035
24	STOXX50E	ESTX 50 PR.EUR	4234
25	TWII	TSEC weighted	5919
26	VIX	CBOE Volatility Index	6069
27	WHITE.NOISE	-	7385
28	XAX	NYSE AMEX Composite Index	6069

Table 1: World market indices in the input dataset

4.2. LSH Hyperparameters Setting

In this case the setting provides that every block is signed with a sequence of H = 200 i.i.d. hash codes. Every hash is a n = 32 long integer which is a sufficient length (in bits) for hashing the blocks with a low number of collisions. Each signature is grouped into B = 50 bands of R = 4 hashes combined in *bitwise XOR.* Due to these hyperparameters the probability of detecting a pair of blocks with a cosine similarity equal to 0.8 equates to 1 (see Equation 1) dropping the probability of false negatives to zero while the probability of false positives increases. In order to avoid this inflation of positives in the solution set a similarity threshold equal to σ = 0.85 is used. Pairs with ae cosine similarity *sim* such that $|sim| < 0.85$ are excluded.

4.3. Randomness Estimation

The number of pairs in the set $\mathcal{M}$, which are considered acceptable, is therefore equal to $M = 233531$. These pairs concern patterns of price movements in either concordant directions or in opposite directions with a high degree of cosine similarity. In both cases, the significant result is that the mutual information exchanged between the blocks in each pair also reaches high values, as well as reported in Table 2, implying that past information or other time series are also exploitable for analysis.

By summarizing the count matrix introduced in Section 2, the overall percentage of similar blocks concerning real-time series with respect to the total of the detected pairs is approximately equal to 99.94%. It is important to underline that this result provides a measure of the goodness of the solution as opposed to a measure of the randomness of the entire input dataset. Moreover, Table 2 indicates that pairs of repeated blocks across the time series over time are mutually informative. Hence in order to explore the randomness within the time series dataset, the matrix M containing the number of repeated patterns is summarized using the MPI index. The value of the index, as well as the rank, are reported in Table 3.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.8194	1.0000	1.0000	0.9985	1.0000	1.0000

Table 2: Mutual information in the solution set normalized values.

The index aggregates per column the values of each row *i* of the count matrix to indicate the contribution of past blocks of the time series *i* to all the other time series moving forward in time. As it can be seen from Table 3, the highest value of the MPI reveals that the highest number of blocks pertains to the *BSESN - S&P BSE SENSEX* market index. On the contrary, it is evident that the lowest number of exploitable blocks belongs to the *white noise*, rendering the approach an effective tool for exploring randomness.

A simple example of application of the results obtained is the evaluation of trading strategies by using back-testing approaches widely used in finance. As a matter of fact, pairs of similar patterns $\{x_{(k-1)+1} - \mu_k, x_{(k-1)+2} - \mu_k, \ldots, x_{(k-1)+L} - \mu_k\}$ of price fluctuations relative to their block averages μ_k imply similar patterns of the prime differences and as a consequence the simple L-days *return sign* $sign(r_k)$ is related to the sign of the differencee $(x_{(k-1)+L} - x_{(k-1)+1})$ so that a positive difference yields a positive return and viceversa. In this case study, pairs with a positive cosine similarity have a concordance of their return signs equal to 99.65% while pairs with a negative cosine similarity as well as opposite return signs are 99.48%.

Market index	MPI.value	MPI.rank
AORD	104.0175456	4
AXJO	100.0505033	19
BFX	104.9495307	2
BSESN	107.7398143	1
BUK100P	87.3645177	26
DJI	101.6488485	13
FCHI	101.1772325	15
FTSE	100.5406199	17
GDAXI	102.4744143	9
GSPC	97.3825360	22
GSPTSE	100.4525230	18
HSI	101.8429500	12
IMOEX.ME	87.2236976	27
IXIC	101.1821435	14
JKSE	103.1989557	7
KLSE	103.3817906	6
KS11	100.7769930	16
N100	96.6143371	23
N225	101.8955299	11
NYA	99.8565148	20
NZ50	102.8206462	8
RUT	97.7420234	21
STI	103.8062985	5
STOXX50E	91.1342641	25
TWII	104.7133149	3
VIX	92.6223118	24
WHITE.NOISE	83.8343378	28
XAX	102.0155606	10

Table 3: MPI index

5. Conclusion

The proposed approach explores the randomness of a set of time series by turning the exploration into a pattern recognition problem. The patterns to be detected are contained within sub-sequences (blocks) of a pre-defined length which are shorter than the shortest time series in the dataset so that the search for repeated patterns over time is reduced to the detection of similar blocks. In this approach time series values are transformed into differences between the values in the block and their average value. In real-world applications, searching for such similar patterns can become intractable when the input dataset is large. Hence, in this study, it has been proposed to use the Locality Sensitive Hashing technique, which is effective for the accomplishment of this task. The advantage of using the specific family of hashing functions of Random Projections is that it allows an approximate yet fast evaluation of the cosine similarity between the input blocks. In any case, the setup of the time window hyperparameter constitutes an important aspect. Long time windows can lead to the detection of a low number of repeated patterns, while, on the contrary, short time windows can lead to similar insignificant patterns. In order to test the effectiveness of the proposed approach, it was applied to the case of financial markets for which the evaluation of randomness is crucial. The case study examined concerns a number of time series of different financial indices which, given their length, constitute a valid test for the computational efficiency of the proposed method. The pairs of similar blocks detected reveal repeated patterns over time across the time series. High degrees of cosine similarity (in absolute value) are related to high degrees of the mutual information exchanged. This implies that the underlying random processes pertaining to these repeated patterns act the same even if they are related to different time periods. Due to this fact, the high percentage of pairs of similar blocks involving only true time series compared to the percentage of pairs involving blocks belonging to the white noise used as a reference series provides an empirical measure of randomness of the time series dataset under investigation. As it is well-known, in the case of financial markets, the more of these patterns are identified the less the market is efficient as patterns from the past can be exploited for analyticl purposes. Pairs of near-duplicate blocks are gathered into the square matrix of their counts which indicate the number of blocks from the past belonging to each time series similar to blocks forward in time belonging to the same or another time series in the dataset. This matrix is summarized by using a well-known synthesis method in order to obtain the overall contribution that each time series gives to all the others. This approach allows for an effective exploration if a fake, i.e., completely random, series such as the white noise proposed in this study as a reference time series is inserted into the dataset. As expected, the case study results showed that the block pairs detected mostly concern the real-time series of the input dataset. In conclusion, the results obtained confirm that the proposed approach provides a valid tool for a fast exploration of the randomness within a set of time series.

References

1. Bradley, J. V. (1960). *Distribution-free statistical tests* (Vol. 60, No. 661). United States Air Force.
2. Siegel, S. (1957). Nonparametric statistics. *The American Statistician, 11*(3), 13-19.
3. Brida, J. G., & Punzo, L. F. (2003). Symbolic time series analysis and dynamic regimes. *Structural Change and Economic Dynamics, 14*(2), 159-183.
4. Kurths, J., Schwarz, U., Witt, A., Krampe, R. T., & Abel, M. (1996, June). Measures of complexity in signal analysis. In *AIP Conference Proceedings* (Vol. 375, No. 1, pp. 33-54). American Institute of Physics.
5. Keller, K., & Lauffer, H. (2003). Symbolic analysis of high-dimensional time series. *International Journal of Bifurcation and Chaos, 13*(09), 2657-2668.
6. Varga, J. (1996). Tests for randomness in multiple financial time series. *In Modelling Techniques for Financial Markets and Bank Management* (pp. 259-271). Physica-Verlag HD.
7. Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal, 27*(3), 379-423.
8. Bandt, C., & Pompe, B. (2002). Permutation entropy: a natural complexity measure for time series. *Physical review letters, 88*(17), 174102.
9. Pincus, S. M. (1991). Approximate entropy as a measure of system complexity. *Proceedings of the national academy of sciences, 88*(6), 2297-2301.
10. Richman, J. S., & Moorman, J. R. (2000). Physiological time-series analysis using approximate entropy and sample entropy. *American journal of physiology-heart and circulatory physiology, 278*(6), H2039-H2049.
11. Cover, T. M. (1999). *Elements of information theory*. John Wiley & Sons.
12. Dionisio, A., Menezes, R., & Mendes, D. A. (2004). Mutual information: a measure of dependency for nonlinear time series. Physica A: *Statistical Mechanics and its Applications, 344*(1-2), 326-329.
13. Tanake, Y., Iwamoto, K., & Uehara, K. (2005). Discovery of time-series motif from multi-dimensional data based on MDL principle. *Machine Learning, 58*, 269-300.
14. Zheng, Y., Liu, Q., Chen, E., Ge, Y., & Zhao, J. L. (2014, June). Time series classification using multi-channels deep convolutional neural networks. In International conference on web-age information management (pp. 298-310). Cham: *Springer International Publishing*.
15. Song, J., Gan, Z., & Carin, L. (2016, June). Factored temporal sigmoid belief networks for sequence learning. *In International Conference on Machine Learning* (pp. 1272-1281). PMLR.
16. Rajaraman, A., & Ullman, J. D. (2011). *Mining of massive datasets. Autoedicion.*
17. Yu, C., Luo, L., Chan, L. L. H., Rakthanmanon, T., & Nutanong, S. (2019). A fast LSH-based similarity search method for multivariate time series. *Information Sciences,*

476, 337-356.
18. Zhang, M. (2021, July). Time Series Search Based on Locality Sensitive Hashing. In *2021 3rd International Conference on Applied Machine Learning (ICAML)* (pp. 57-61). IEEE.
19. Mazziotta, M., & Pareto, A. (2016). On a generalized non-compensatory composite index for measuring socio-economic phenomena. *Social indicators research, 127*, 983-1003.
20. Malkiel, B. G. (2003). The efficient market hypothesis and its critics. *Journal of economic perspectives, 17*(1), 59-82.
21. Lo, A. W., & MacKinlay, A. C. (1988). Stock market prices do not follow random walks: Evidence from a simple specification test. *The review of financial studies, 1*(1), 41-66.
22. Shiller, R. J. (1981). Do stock prices move too much to be justified by subsequent changes in dividends?.