Protecting Time Series Data with Minimal Absolute Change to Forecasts

Matthew J. Schneider, Jinwook Lee
Decision Sciences and MIS, LeBow College of Business, Drexel University, Philadelphia, PA 19104
matt.schneider@drexel.edu, jinwook.lee@drexel.edu

Privacy is a social good and little attention has been paid to how data protection affects forecasting. To measure the potential severity of this problem, we derive theoretical bounds for the absolute change to forecasts generated from additive exponential smoothing models using protected data. We protect time series data by creating the k-nearest Time Series (k-nTS) Swapping and k-means Time Series (k-mTS) Shuffling methods that minimize the absolute change to forecasts while protecting privacy. We optimize an integer programming problem with perfect matching for simultaneous data swapping to trade-off the utilities between a data provider and data intruder. We apply our data protection methods to thousands of time series and find that they maintain the exponential smoothing forecasts and time series patterns (level, trend, and seasonality) better than standard data protection methods. Our findings suggest the importance of explicitly incorporating the forecasting model into the generation of protected data itself, and measuring the effectiveness of forecasting for social good initiatives with non-economic metrics that benefit society at large.

Key words: privacy, forecasting for social good, exponential smoothing, data protection, optimal data swapping

1. Introduction

1.1. Our motivation and background of the paper

Privacy is a social good. It gives power to individuals through control over their personal information (Véliz (2020)). Surprisingly, despite all the new data protection laws such as GDPR or CCPA, the number of confirmed data breaches doubled from 2019 to 2021 (Verizon (2019),Verizon (2021)), resulting in more harm to society at large. Responsible data providers should assume that all data will eventually get out and consequently, alter the data to protect individuals' identities and sensitive data before it is shared or used. Thus far, little research exist on privacy for forecasting applications (Boone et al. (2019)) and this paper is the first attempt to develop data privacy theory for forecasting, using additive exponential smoothing models.

To protect time series data, organizations frequently aggregate or add noise to their data at levels beyond usefulness. The focus is rightfully "privacy first," without explicitly incorporating how the data is used downstream. However, Schneider et al. (2018) showed that improved assumptions about the data generating process of protected data can improve the quality of the downstream

use case while preserving privacy. Specifically, the authors inserted the loss function of a widely used pricing model into their data protection method (DPM) to generate synthetic retail sales data that maintained store-wide pricing elasticities and profitability better than standard DPMs. For forecasting, the question then becomes, "How does one protect time series data to maintain similar forecasts?"

Evaluating protected data for forecasting requires balancing a multi-metric trade-off between privacy and forecasting. Recently, Rostami-Tabar et al. (2021) applied Doughnut theory to also incorporate multi-metric 'compasses' and defined Forecasting for Social Good (FSG) as "a forecasting process that aims to inform decisions that prioritise the thriving of humanity over the thriving of economies by enhancing the social foundation and ecological ceilings that impact the public as a whole on both local and global levels" (p. 3). Examples of FSG situations include forecasting regional water levels for humanitarian efforts, de-biasing forecasts to remove opportunities for discrimination, and developing new forecasting metrics that favor societal benefits over accuracy. In our case, the F in FSG is the ability of a forecasting model to maintain similar forecasts to the confidential data and the SG in FSG is the privacy level. Furthermore, FSG applies because privacy impacts the public, and individuals are no better off in recent years despite the massive financial windfalls to privacy law firms and tech companies. This leads us to develop metrics and DPMs for forecasting that primarily value the social good.

In some situations, data protection may actually increase the availability of data due to the willingness of organizations to share new information. For example, each year millions of men, women, and children are transported across the globe to be sold for slavery, forced labor, sex, illegal adoption, or domestic servitude (DHS (2021)). Consider the case of forecasting monthly cross-border flows of human trafficking counts at the local or national levels which can influence public policy. One issue is that these time series data are frequently unavailable at disaggregated time or geographical levels (Yao et al. (2021) because they relate to sensitive issues or specific persons. However, forecasting cross-border flows could improve (or newly initiate) the allocation of budgets and resources at local levels to help these trafficked victims. The FSG framework applies in this situation because there is a trade-off between privacy issues and the ability to forecast human trafficking.

Protecting data for intended use cases involving forecasting is challenging due to the repeated nature of time series data. First, a data provider needs to protect time series data at every time i in order to prevent a data intruder from targeting privacy issues in the underlying confidential time series. Second, the data provider needs to ensure that a forecaster can use the protected data to make similar forecasts to those using the confidential data for time $i+1$. We find that the first

two challenges compete with one another – that is, better data privacy at time i often results in less similar forecasts for time $i+1$.

There are serious privacy concerns due to a potential attack from data intruders that gain access to the protected (or confidential) data. We refer the readers to the literature for several examples, e.g., Garfinkel et al. (2002), Sweeney (2002), Li and Sarkar (2009), etc. For this paper, we consider two basic types of utility for a data intruder at time i: first, detection of extreme values of the protected data within a time series j and second, targeting of anomalous time series compared to all time series $j = 1, \ldots, J$. We refer the readers to relevant literature for a thorough review on anomaly detection and defining extreme values for time series, e.g., Teng (2010), Gorr and Schneider (2013), Laptev et al. (2015), Lee et al. (2017), Munir et al. (2019) and references therein.

Recent data privacy legislation such as the General Data Protection Regulation (GDPR) (European Parliament and Council of European Union (2016)) in the European Union and the California Consumer Privacy Act (CCPA) (California State Legislature (2018)) requires the protection of personal data, such as time series data related to a natural person. To protect personal data, a data provider can use data security measures and anonymization methods. Data security measures might include encryption or restricting access to users which lowers the chance of a data breach. These measures are considered pseudonymization because they are reversible (Article 29 Data Protection Working Party (2014)) and do not randomize or generalize the personal data directly and offer no privacy in the event of a data breach of the confidential data.

On the other hand, anonymization methods permanently and irreversibly alter the personal data which reduces the utility to a data intruder. Table 1 summarizes standard data protection methods reported from the Article 29 Data Protection Working Party (2014) which was a precursor to GDPR. Standard anonymization methods or data protection methods (DPMs) include suppression (removal or redaction of personal data), generalization ("generalizing, or diluting, the attributes of data subjects by modifying the respective scale or order of magnitude," p.16) , or randomization (altering "the veracity of the data in order to remove the strong link between the data and the individual," p.12). Anonymization methods may also include more sophisticated DPMs using local differential privacy (Cheu et al. (2021)) or synthetic data regression models with time series data (Schneider and Abowd (2015)), or a study of how privacy interacts with bias issues (Xu and Zhang (2021)). However, a gap in the literature exists on using DPMs for forecasting.

The main issue with DPMs found thus far in the literature is that they do not measure how forecasts change when choosing how to protect time series data. In practice, data protection is usually chosen by privacy-minded experts, such as IT or legal counsel. For example, California's "15/15" rule (Harvey (2013)) requires all time series data on household energy usage to be aggregated into groups of 15 households or more, with no one household exceeding 15% of the group's

Table 1 Standard Data Protection Methods (DPMs)

Standard DPM	Description	Category
Pseudonymization, Encryption, Hash function, Tokenization	Replacing one attribute in a record with another	Neither Generalization or Randomization
Aggregating, Averaging	Grouping attribute values so that each individual shares the same value	Generalization
Bottom- (or Top)-Coding	Grouping the bottom (or top) percent of attribute values to the same value	Generalization
Noise Addition	Adding a random number to a data point	Randomization
Swapping or Permutation	Swapping the values of data points	Randomization

total energy usage. This generalization is potentially problematic because it provides no theoretical guarantee to how forecasts change, and restricts the household energy forecasts to use top-down forecasting methods (Gross and Sohl (1990)) based on potentially unknown household energy usage proportions. Other organizations adjust time series data upward or downward for privacy reasons. For example, the United States Department of Agriculture releases public data from the National Household Food Acquisition and Purchase Survey. They require annual income and expenses to be top-coded at the 99% quantile (Economic Research Service USDA (2016)). As a result, forecasts using top-coded income are different (and likely lower) than those using the confidential data. Fildes et al. (2009) found that downward adjustments to forecasts were more effective at improving forecast accuracy than upward or small adjustments, but the findings depended on the context of the data.

Our study differs because we do not make an assessment as to whether privacy adjustments improve forecast accuracy. Instead, we establish theory that measures the adjustment (absolute change) to forecasts and tailors DPMs to minimize those adjustments in a trade-off with privacy. Using additive exponential smoothing models, we provide theoretical bounds on how different protected data at time i changes the forecasts computed at time i (intended for time $i+1$). The theoretical bounds of the absolute change to forecasts at time i also imply the upper bound of the change to forecast accuracy at time $i+1$, but not the forecast accuracy itself. With respect to how altered data at time i improves forecast accuracy at time $i+1$, Luo et al. (2018) found that additive Gaussian white noise created forecasts that were no longer accurate. Although we also find that additive noise will have a bound that depends on the largest generated random noise in any time period, their study differed because it was empirical and simulated a hacker altering the data for nefarious reasons (as opposed to a DPM tailored for social good).

1.2. Key contributions and organization of the paper

This paper provides three contributions to the literature. First, we develop mathematical models and a decision-making framework between a data provider and data intruder. Our goal is to

minimize the absolute change to forecasts while keeping data privacy at a desirable level. When it comes to the information loss in general, this topic has long been studied (e.g., see Menon and Sarkar (2007) and references therein). Recently, Baena et al. (2020) studied optimization models with applications for data privacy. Using a large scale discrete optimization, they solved nicely a cell suppression problem in order to remove vulnerable data. In our case, however, such optimization problem formulation would not be suitable since we also want to minimize the absolute change to forecasts of time series data, not only the privacy issues. Our model is an initial attempt to the challenge of protecting time series data for forecasting applications which is a known gap in the literature (Boone et al. (2019)). Our two-party framework is twofold: (i) individual decision making of a data provider to trade-off protecting privacy with maintaining similar forecasts (ii) individual decision making of a data intruder to target privacy issues in time series data. We characterize the two decision makers' judgements based on their own preferences with utility functions.

Second, we provide theoretical bounds on the maximum absolute change to forecasts for any DPM, which is defined as the absolute difference between forecasts made using the confidential data and forecasts made using the protected data. Although we only provide theory on forecasts for time $i+1$, the theory can be easily extrapolated to forecast horizons larger than one. We provide these bounds for the additive Holt-Winters exponential smoothing models in Section 2 by decomposing the forecasts. This decomposition also extends into insights beyond forecasting, such as how the underlying time series structures such as levels, trends, and seasonalities change with data protection. We provide empirical results for the DPMs in Section 6.

Third, we create a DPM that minimizes the maximum absolute change to forecasts based on a multidimensional Euclidean space of similar time series. The data provider replaces confidential data with the protected data by randomly swapping it with values from k-nearest time series or randomly permuting it within k groups of similar time series. Protected data are specified through a parameter that balances the trade-off between (a) absolute change to forecasts and (b) a data intruder's ability to target a privacy issue in a time series (represented by a utility function). Our approach allows the data provider to evaluate and optimize the DPM simultaneously while choosing the DPM in time i.

The structure of the paper is as follows. Section 2 establishes the theoretical bounds of the absolute change to forecasts for any DPM including the standard DPMs in legislation. Section 3 defines the standard DPMs and introduces our k-nearest Time Series (k-nTS) Swapping method to improve the theoretical bound of the absolute change to forecasts. It also introduces the k-means Time Series (k-mTS) Shuffling method to replace actual values with a protected value based on a group of time series rather than an individual time series. Section 4 introduces the data intruder's utility function and decision analysis. In Section 5 we present the two-party data privacy framework

and its related mathematical models for forecasting between the data provider and data intruder, and find its related optimal data swapping strategy. Our main optimization model is presented in Section 5.2. Finally, Section 6 applies our DPM to sensitive time series data and measures the trade-off between the absolute change to forecasts and data privacy. We also measure how DPMs change time series structures and conclude in Section 7.

2. Absolute Change to Forecasts from Data Protection

We define the absolute change to forecasts as $|F_{i+1} - F^*_{i+1}|$, where F_{i+1} is a forecast using the confidential data and F^*_{i+1} is a forecast using the protected data. Framing the problem in terms of absolute change to forecasts provides us with two key advantages. First, unlike traditional forecast accuracy measures such as mean squared error, absolute change to forecasts can be computed at time i (for time $i+1$) without the knowledge of the confidential value, A_{i+1}, at time $i+1$. Thus, the data provider can alter the protected data, P_i, at time i if the absolute change to forecasts is too large. Second, we can establish a priori bounds of the absolute change to forecasts with the commonly used exponential smoothing models in subsection 2.1. These bounds also imply an upper bound for the change to forecast accuracy at time $i+1$ in subsection 2.2.

2.1. Absolute Change to Forecasts for Exponential Smoothing

We use the additive Holt-Winters' equations to establish tractable bounds of the absolute change to forecasts for any data protection method. The Holt-Winters' forecasts are a function of the level (l_t), trend(b_t), and seasonality (s_t) of each time series. These components differ depending on whether the confidential data, A_t, or the protected data, P_t, was input into the equations for single exponential smoothing (SES), double exponential smoothing (DES), and triple exponential smoothing (TES). With data protection starting at time t, if $|A_i - P_i|$ is bounded by some value, say $M \in \mathbb{R}_+$ (i.e, $|A_i - P_i| \leq M$), for any integer $i \geq t$, then the absolute change to forecasts at any time points after t can be written up in terms of smoothing factors and M. Specifically, we discovered a functional form of inequalities, i.e., the bounds for the absolute change to forecasts at time $T+1$ for any $T \geq t+1$, which is a geometric series $M \sum_{n=1}^{T-t} a(\cdot) r(\cdot)^{n-1}$ where $a(\cdot)$, the initial term and $r(\cdot)$, the common ratio are both the functions of related smoothing factors α, β, γ.

2.1.1. Single Exponential Smoothing (SES) Let us begin with single exponential smoothing. We choose exponential smoothing because it is commonly used by practitioners and robust under distributional assumptions which will be required for arbitrary data protection methods (see, e.g., Gelper et al. (2009)). The forecasting equation is the following,

$$F_{t+1} = \alpha A_t + (1-\alpha) F_t,\ 0 \leq \alpha \leq 1,\ t \in \mathbb{Z}_+, \tag{1}$$

where α is the smoothing factor for the level l. We assume that the data provider begins data protection starting at time t and therefore, we let $F_{t+1} = l_t$. Then, given confidential data up to time $t-1$, we can write the level for the confidential data, l_t and the level for protected data, l_t^* as

$$\begin{array}{ll} \text{Confidential:} & l_t = \alpha A_t + (1-\alpha) l_{t-1} \\ \text{Protected:} & l_t^* = \alpha P_t + (1-\alpha) l_{t-1}. \end{array} \tag{2}$$

THEOREM 1 **(Bounded absolute change to forecasts in case of SES)**. *Suppose that data protection starts at time t. For any integer $i \geq t$, if $|A_i - P_i|$ is bounded by $M \in \mathbb{R}_+$ (i.e., $|A_i - P_i| \leq M$), then we have a recursive formula for the absolute change to forecasts bound only in terms of M and α. For any $T \geq t+1$, the bound for the absolute change to forecasts $F - F^*$ can be written up as:*

$$|F_{T+1} - F_{T+1}^*| \leq \alpha M + (1-\alpha)|F_T - F_T^*| \tag{3}$$

with the initial value $F_{t+1} - F_{t+1}^* = \alpha(A_t - P_t)$. Equivalently, we have the following

$$|F_{T+1} - F_{T+1}^*| \leq \sum_{n=1}^{T-t} (1-\alpha)^{n-1}(2\alpha - \alpha^2) M, \tag{4}$$

which is a geometric series $M \sum_{n=1}^{T-t} a(\cdot) r(\cdot)^{n-1}$ where a, the initial term and r, the common ratio are the functions of related smoothing factor α.

Proof. Suppose that data protection starts at time t. Then $F_{t+1} = l_t$ and $F_{t+1}^* = l_t^*$. Subtraction of l_t from l_t^* (in (2)) gives us the following:

$$l_t - l_t^* = \alpha(A_t - P_t) = \alpha \Delta_t, \tag{5}$$

where $\Delta_t = A_t - P_t$ and this is equivalent to the absolute change to forecasts for time $t+1$:

$$F_{t+1} - F_{t+1}^* = \alpha \Delta_t. \tag{6}$$

Next, we consider the case of data protection in both time period t and $t+1$. If we roll forward data protection to $t+1$, then we have for the confidential data:

$$l_{t+1} = \alpha A_{t+1} + (1-\alpha) l_t = \alpha A_{t+1} + (1-\alpha)(\alpha A_t + (1-\alpha) l_{t-1}), \tag{7}$$

and in case of the protected data:

$$l_{t+1}^* = \alpha P_{t+1} + (1-\alpha) l_t^* = \alpha P_{t+1} + (1-\alpha)(\alpha P_t + (1-\alpha) l_{t-1}). \tag{8}$$

By subtraction of (8) from (7) we get:

$$l_{t+1} - l_{t+1}^* = \alpha(A_{t+1} - P_{t+1}) + \alpha(1-\alpha)(A_t - P_t) = \alpha \Delta_{t+1} + \alpha(1-\alpha)\Delta_t, \tag{9}$$

which is equivalent to the absolute change to forecasts for $t+2$:

$$F_{t+2} - F^*_{t+2} = \alpha\Delta_{t+1} + \alpha(1-\alpha)\Delta_t = \alpha\Delta_{t+1} + (1-\alpha)(F_{t+1} - F^*_{t+1}) \tag{10}$$

Since data protection starts at time t (confidential data up to time $t-1$), for any time $T \geq t+1$ we have the same form as in the equation (10) only with different subscripts. Therefore, we can write the following recursion:

$$F_{T+1} - F^*_{T+1} = \alpha\Delta_T + (1-\alpha)(F_T - F^*_T). \tag{11}$$

For any integer $i \geq t$, if $|A_i - P_i|$ is bounded by $M \in \mathbb{R}_+$ (i.e., $|A_i - P_i| = |\Delta_i| \leq M$), we have the following

$$|F_{T+1} - F^*_{T+1}| \leq \alpha M + (1-\alpha)|F_T - F^*_T|. \tag{12}$$

Using the initial value $F_{t+1} - F^*_{t+1} = \alpha\Delta_t$ and the recursion, we have the following

$$\begin{aligned} |F_{T+1} - F^*_{T+1}| &\leq \sum_{n=1}^{T-t} (1-\alpha)^{n-1}(\alpha M + (1-\alpha)|F_{t+1} - F^*_{t+1}|) \\ &\leq \sum_{n=1}^{T-t} (1-\alpha)^{n-1}(2\alpha - \alpha^2)M, \end{aligned} \tag{13}$$

which completes the proof. □

Remark 1 (Data protection method for bounding $A_n - P_n$). Note that a sequence $\Delta_n = A_n - P_n$ for all $n \in \mathbb{Z}_+$ can be bounded if we use our proposed data protection methods (in Sections 3.1, 3.2). Given the number of time periods for protection is $T-t+1$, we establish that the maximal absolute change to forecasts for $T+1$ is bounded by the maximal difference ($M \in \mathbb{R}_+$) between confidential and protected data over all time periods. This implies that the bounds in absolute change to forecasts from standard DPMs in Table 1 are larger if they have a single time period where the difference between the confidential and protected data is large. For example, top-coding will have a large bound in absolute change to forecasts if the confidential data was much larger than the $p\%$ upper quantile limit at any time period. Additive noise will have a bound that depends on the largest generated random noise in any time period. Since P_n plays a significant role, the data provider can limit the maximal absolute change to forecasts in $T+1$ by carefully choosing P_n with respect to A_n.

2.1.2. Double Exponential Smoothing (DES) We turn our attention to forecasting with DES. In addition to the smoothing factor α for the level l, let us consider a smoothing factor β for the trend b (assuming there is a trend). The following is the DES formula for the confidential data without data protection:

$$\begin{aligned} l_t &= \alpha A_t + (1-\alpha)F_t \\ b_t &= \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \\ F_{t+1} &= l_t + b_t, \end{aligned} \tag{14}$$

where $0 \leq \alpha \leq 1$, $0 \leq \beta \leq 1$, and the forecast horizon is 1. Suppose that time series data up to time $t-1$ are not protected. Then, with protected data at time t we have the following equations

$$\begin{aligned} l_t^* &= \alpha P_t + (1-\alpha)(l_{t-1} + b_{t-1}) \\ b_t^* &= \beta(l_t^* - l_{t-1}) + (1-\beta) b_{t-1} \\ F_{t+1}^* &= l_t^* + b_t^*. \end{aligned} \tag{15}$$

THEOREM 2 **(Bounded absolute change to forecasts in case of DES)**. *Suppose that data protection starts at time t. For any integer $i \geq t$, if $|A_i - P_i|$ is bounded by $M \in \mathbb{R}_+$ (i.e., $|A_i - P_i| \leq M$), then we have a recursive formula for the absolute change to forecasts bound in terms of M, α and β. For any $T \geq t+1$, the bound for the absolute change to forecasts $F - F^*$ can be written up as::*

$$|F_{T+1} - F_{T+1}^*| \leq (\alpha + 2\alpha\beta)M + (1 - \alpha(1+\beta))|F_T - F_T^*|, \tag{16}$$

with the initial value $F_{t+1} - F_{t+1}^* = \alpha(1+\beta)(A_t - P_t)$. Equivalently, we have the following

$$|F_{T+1} - F_{T+1}^*| \leq \sum_{n=1}^{T-t} (1 - \alpha(1+\beta))^{n-1} (2\alpha + 3\alpha\beta - \alpha^2(1+\beta)^2) M, \tag{17}$$

which is a geometric series $M \sum_{n=1}^{T-t} a(\cdot) r(\cdot)^{n-1}$ where a, the initial term and r, the common ratio are the functions of related smoothing factors α and β.

Proof. Subtracting the corresponding equations of (15) from those of (14) gives us

$$l_t - l_t^* = \alpha(A_t - P_t) = \alpha \Delta_t, \tag{18}$$

$$b_t - b_t^* = \beta(l_t - l_t^*) = \beta\alpha(A_t - P_t) = \alpha\beta\Delta_t, \tag{19}$$

where $\Delta_t = A_t - P_t$. Since $F_{t+1} = l_t + b_t$ and $F_{t+1}^* = l_t^* + b_t^*$, we can write the absolute change to forecasts $F_{t+1} - F_{t+1}^*$ for time $t+1$ by adding (18) to (19) as the following:

$$F_{t+1} - F_{t+1}^* = (\alpha + \alpha\beta)\Delta_t = \alpha(1+\beta)\Delta_t. \tag{20}$$

For F_{t+2} and F_{t+2}^* we roll forward to protect data at time $t+1$, provided that all time series up to time $t-1$ are not protected as in case of F_{t+1}. Since $F_{t+2} - F_{t+2}^* = l_{t+1} + b_{t+1} - (l_{t+1}^* + b_{t+1}^*)$, let us present equation for each term:

$$\begin{aligned} l_{t+1} &= \alpha A_{t+1} + (1-\alpha)(l_t + b_t) \\ b_{t+1} &= \beta(l_{t+1} - l_t) + (1-\beta) b_t, \end{aligned} \tag{21}$$

$$\begin{aligned} l_{t+1}^* &= \alpha P_{t+1} + (1-\alpha)(l_t^* + b_t^*) \\ b_{t+1}^* &= \beta(l_{t+1}^* - l_t^*) + (1-\beta) b_t^*, \end{aligned} \tag{22}$$

Together with (18) and (19), subtraction of corresponding equations of (22) from those of (21) gives us:

$$\begin{aligned} l_{t+1} - l^*_{t+1} &= \alpha(A_{t+1} - P_{t+1}) + (1-\alpha)[(l_t - l^*_t) + (b_t - b^*_t)] \\ &= \alpha\Delta_{t+1} + (1-\alpha)(F_{t+1} - F^*_{t+1}). \end{aligned} \tag{23}$$

$$\begin{aligned} b_{t+1} - b^*_{t+1} &= \beta[(l_{t+1} - l^*_{t+1}) - (l_t - l^*_t)] + (1-\beta)(b_t - b^*_t) \\ &= \beta(l_{t+1} - l^*_{t+1}) - \beta\{(l_t - l^*_t) + (b_t - b^*_t)\} + (b_t - b^*_t) \\ &= \beta\left(\alpha\Delta_{t+1} + (1-\alpha)(F_{t+1} - F^*_{t+1})\right) - \beta(F_{t+1} - F^*_{t+1}) + \alpha\beta\Delta_t \\ &= \alpha\beta\Delta_{t+1} - \alpha\beta(F_{t+1} - F^*_{t+1}) + \alpha\beta\Delta_t \\ &= \alpha\beta\{\Delta_{t+1} + \Delta_t - (F_{t+1} - F^*_{t+1})\}. \end{aligned} \tag{24}$$

By adding up (23) and (24) we have the absolute change to forecasts for $t+2$ as

$$F_{t+2} - F^*_{t+2} = \alpha\Delta_{t+1} + \alpha\beta\Delta_{t+1} + \alpha\beta\Delta_t + (1-\alpha-\alpha\beta)(F_{t+1} - F^*_{t+1}). \tag{25}$$

Since data protection starts at time t (confidential data up to time $t-1$), for any time $T \geq t+1$ we have the same form as in the equation (25) only with different subscripts. Therefore, we can write the following recursion:

$$F_{T+1} - F^*_{T+1} = \alpha\Delta_T + \alpha\beta\Delta_T + \alpha\beta\Delta_{T-1} + (1-\alpha-\alpha\beta)(F_T - F^*_T). \tag{26}$$

Let $M \in \mathbb{R}_+$ such that $|A_n - P_n| = |\Delta_n| \leq M$ for all $n \in \mathbb{Z}_+$. Then (26) can be written up as

$$|F_{T+1} - F^*_{T+1}| \leq (\alpha + 2\alpha\beta)M + (1 - \alpha(1+\beta))|F_T - F^*_T|. \tag{27}$$

Using the initial value $F_{t+1} - F^*_{t+1} = \alpha(1+\beta)\Delta_t$ and recursion, we have the following

$$\begin{aligned} |F_{T+1} - F^*_{T+1}| &\leq \sum_{n=1}^{T-t} (1-\alpha(1+\beta))^{n-1} \left[(\alpha + 2\alpha\beta)M + (1 - \alpha(1+\beta))|F_{t+1} - F^*_{t+1}|\right] \\ &\leq \sum_{n=1}^{T-t} (1-\alpha(1+\beta))^{n-1}(2\alpha + 3\alpha\beta - \alpha^2(1+\beta)^2)M, \end{aligned} \tag{28}$$

which completes the proof. □

Note that from (26), we can remove some terms to express the bound only with M and α:

$$\begin{aligned} |F_{T+1} - F^*_{T+1}| &\leq \alpha M + \alpha\beta M + \alpha\beta M + (1-\alpha-\alpha\beta)|F_T - F^*_T| \\ &\leq 3\alpha M + (1-\alpha)|F_T - F^*_T|. \end{aligned} \tag{29}$$

The second inequality of (29) holds because $0 \leq \alpha\beta \leq \alpha$. Using $|F_{t+1} - F^*_{t+1}| \leq \alpha(1+\beta)M \leq 2\alpha M$, it follows that for any $T \geq t+1$, we have the following

$$\begin{aligned} |F_{T+1} - F^*_{T+1}| &\leq \sum_{n=1}^{T-t} (1-\alpha)^{n-1} \left[3\alpha M + 2\alpha(1-\alpha)M\right] \\ &= \sum_{n=1}^{T-t} (1-\alpha)^{n-1} \left(5\alpha M - 2\alpha^2 M\right) \\ &\leq \sum_{n=1}^{T-t} 5\alpha(1-\alpha)^{n-1} M, \end{aligned} \tag{30}$$

where the last inequality can be used in practice when α^2 and $\alpha\beta$ are negligible.

2.1.3. Triple Exponential Smoothing (TES) The standard Holt-Winters' triple exponential smoothing (TES) equations are as follows:

$$F_{t+1} = l_t + b_t + s_{t+1-m}, \tag{31}$$

where

$$\begin{aligned} l_t &= \alpha(A_t - s_{t-m}) + (1-\alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \\ s_t &= \gamma(A_t - l_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}, \end{aligned} \tag{32}$$

where the forecast horizon is 1, $0 \leq \alpha \leq 1$, $0 \leq \beta \leq 1$, $0 \leq \gamma \leq (1-\alpha)$, F_{t+1} is the forecast for time $t+1$ made at time t, l_t is the smoothed level, b_t is the trend, m is the frequency of the time series, and s_t is the seasonal adjustment.

However, unlike SES and DES, forecasts using TES depend on m seasonality adjustments computed from the time period before to ensure that they sum to zero. To do this continuous normalization, McKenzie (1986) (and later Archibald and Koehler (2003)) first suggests reparameterizing the above equations as follows

$$l_t = l_{t-1} + b_{t-1} + \alpha(A_t - F_t) \tag{33}$$

$$b_t = b_{t-1} + \alpha\beta(A_t - F_t) \tag{34}$$

$$s_t = s_{t-m} + \gamma(1-\alpha)(A_t - F_t), \tag{35}$$

where the initial coefficients α, β, and γ are multiplied together and only the latest seasonality adjustment is recomputed. Then, to maintain the same forecasts, we subtract

$$r_t = \frac{\gamma(1-\alpha)}{m}(A_t - F_t) \tag{36}$$

from all seasonality adjustments (including the latest in equation (35)) and add this r_t to l_t. Without loss of generality, $s(t-1)$ are the seasonality adjustments computed at time $t-1$ and s are the seasonality adjustments computed at time t. The final equations with continuous normalization for TES are as follows. The equations become

$$l_t = l_{t-1} + b_{t-1} + \alpha(A_t - F_t) + r_t \tag{37}$$

$$b_t = b_{t-1} + \alpha\beta(A_t - F_t) \tag{38}$$

$$\begin{aligned} s_t &= s_{t-m}(t-1) + \gamma(1-\alpha)(A_t - F_t) - r_t \\ &= s_{t-m}(t-1) + (m-1)r_t, \end{aligned} \tag{39}$$

followed by

$$s_{t+k-m} = s_{t+k-m}(t-1) - r_t, \text{for } k = 1, \ldots, m-1. \tag{40}$$

Note that this reparameterization implies $F_{t+1} = l_t + b_t + s_{t+1-m}$ is unchanged since r_t is both added and subtracted to l_t and s_t, respectively. Also, the seasonality factors computed at time t sum to 0 as long as the seasonality factors at time $t-1$ do. Also note that the following

$$\sum_{k=1}^{m} s_{t+k-m}(t-1) = 0 \tag{41}$$

implies

$$\sum_{k=1}^{m} s_{t+k-m} = \sum_{k=1}^{m} s_{t+k-m}(t-1) - (m+1)r_t + (m-1)r_t = 0. \tag{42}$$

Suppose that time series data up to time $t-1$ are not protected. Then, by substituting P_t we have the following subtractions with $k=1$:

$$l_t - l_t^* = \alpha(A_t - P_t) + \frac{\gamma(1-\alpha)}{m}(A_t - P_t) \tag{43}$$

$$b_t - b_t^* = \alpha\beta(A_t - P_t) \tag{44}$$

$$s_{t+1-m} - s_{t+1-m}^* = \frac{\gamma(1-\alpha)}{m}(P_t - A_t) \tag{45}$$

Using $F_{t+1} = l_t + b_t + s_{t+1-m}$, the last two terms for (43) and (45) sum to 0 and the absolute change to forecasts at time $t+1$ becomes

$$F_{t+1} - F_{t+1}^* = (\alpha + \alpha\beta)(A_t - P_t) = \alpha(1+\beta)\Delta_t, \tag{46}$$

which is the same as DES in equation (20) and only depends on Δ_t. Note that, for any time point i, all of $l_i - l_i^*, b_i - b_i^*, s_{i+1-m} - s_{i+1-m}^*$ are in terms of $\Delta_i = A_i - P_i$ with given constants α, β, γ. We also know that rolling forward to later time periods, absolute change to forecasts comes from that of the previous time periods. Therefore, it is not hard to see that the absolute change to forecasts $F - F^*$ at time $T+1$ (for any $T \geq t+1$) can be written up in terms of absolute change to forecastses $F_i - F_i^*$ and $\Delta_i = A_i - P_i$ for $i = t, t+1, \ldots, T$ up to time T.

THEOREM 3 **(Bounded absolute change to forecasts in case of TES)**. *Suppose that data protection starts at time t. For any integer $i \geq t$, if $|A_i - P_i|$ is bounded by $M \in \mathbb{R}_+$ (i.e., $|A_i - P_i| \leq M$), then we have a recursive formula for the absolute change to forecasts bound only in terms of M, α, β, γ and m. For any $T \geq t+1$, the bound for the absolute change to forecasts $F - F^*$ can be written up as::*

$$|F_{T+1} - F_{T+1}^*| \leq \left(2\alpha + 3\alpha\beta + \tfrac{\gamma(1-\alpha)(m+1)}{m}\right) M + (\alpha + \alpha\beta + \gamma(1-\alpha))|F_T - F_T^*| \tag{47}$$

with the initial value $F_{t+1} - F^*_{t+1} = \alpha(1+\beta)(A_t - P_t)$. *Equivalently, for any* $T \geq t+1$, *we have the following*

$$|F_{T+1} - F^*_{T+1}| \leq \sum_{n=1}^{T-t} r^{n-1} \left(2\alpha + 3\alpha\beta + (\alpha+\alpha\beta)^2 + \gamma(1-\alpha)\left(\frac{m+1}{m} + \alpha + \alpha\beta\right)\right) M, \tag{48}$$

where $r = \alpha + \alpha\beta + \gamma(1-\alpha)$ *and* m *designates the frequency of the time series. Note that this is a geometric series* $M\sum_{n=1}^{T-t} a(\cdot)r(\cdot)^{n-1}$ *where* a, *the initial term and* r, *the common ratio are the functions of related smoothing factors* α, β *and* γ.

Proof. Data protection starts at time t and let $\Delta_t = A_t - P_t$. Then for the absolute change to forecasts at time $t+1$, by (46), we can write $F_{t+1} - F^*_{t+1} = \alpha(1+\beta)\Delta_t$. For F_{t+2} and F^*_{t+2} we roll forward to protect data at time $t+1$. Since $F_{t+2} - F^*_{t+2} = l_{t+1} + b_{t+1} + s_{t+2-m} - (l^*_{t+1} + b^*_{t+1} + s^*_{t+2-m})$, let us present the components for both F_{t+2} and F^*_{t+2}. The components of F_{t+2} are:

$$\begin{aligned} l_{t+1} &= l_t + b_t + \alpha(A_{t+1} - F_{t+1}) + r_{t+1} \\ b_{t+1} &= b_t + \alpha\beta(A_{t+1} - F_{t+1}) \\ s_{t+2-m} &= s_{t+2-m}(t+1) + (m-1)r_{t+1} \end{aligned} \tag{49}$$

and for F^*_{t+2} we have:

$$\begin{aligned} l^*_{t+1} &= l^*_t + b^*_t + \alpha(P_{t+1} - F^*_{t+1}) + r^*_{t+1} \\ b^*_{t+1} &= b^*_t + \alpha\beta(P_{t+1} - F^*_{t+1}) \\ s^*_{t+2-m} &= s_{t+2-m}(t+1) + (m-1)r^*_{t+1}. \end{aligned} \tag{50}$$

Subtraction of corresponding equations of (50) from those of (49) gives us:

$$\begin{aligned} l_{t+1} - l^*_{t+1} &= (l_t - l^*_t) + (b_t - b^*_t) + \alpha\Delta_{t+1} - \alpha(F_{t+1} - F^*_{t+1}) + r_{t+1} - r^*_{t+1} \\ b_{t+1} - b^*_{t+1} &= (b_t - b^*_t) + \alpha\beta\Delta_{t+1} - \alpha\beta(F_{t+1} - F^*_{t+1}) \\ s_{t+2-m} - s^*_{t+2-m} &= (m-1)(r_{t+1} - r^*_{t+1}), \end{aligned} \tag{51}$$

which is followed by

$$\begin{aligned} F_{t+2} - F^*_{t+2} &= (l_{t+1} - l^*_{t+1}) + (b_{t+1} - b^*_{t+1}) + (s_{t+2-m} - s^*_{t+2-m}) \\ &= (l_t - l^*_t) + 2(b_t - b^*_t) + (\alpha+\alpha\beta)\Delta_{t+1} - (\alpha+\alpha\beta)(F_{t+1} - F^*_{t+1}) + m(r_{t+1} - r^*_{t+1}) \\ &= \left(\alpha + 2\alpha\beta + \tfrac{\gamma(1-\alpha)}{m}\right)\Delta_t + (\alpha+\alpha\beta)\Delta_{t+1} - (\alpha+\alpha\beta)(F_{t+1} - F^*_{t+1}) \\ &\quad + m\left(\tfrac{\gamma(1-\alpha)}{m}(A_{t+1} - F_{t+1}) - \tfrac{\gamma(1-\alpha)}{m}(P_{t+1} - F^*_{t+1})\right) \\ &= \left(\alpha + 2\alpha\beta + \tfrac{\gamma(1-\alpha)}{m}\right)\Delta_t + (\alpha+\alpha\beta)\Delta_{t+1} - (\alpha+\alpha\beta)(F_{t+1} - F^*_{t+1}) \\ &\quad + \gamma(1-\alpha)\left(\Delta_{t+1} - (F_{t+1} - F^*_{t+1})\right) \\ &= \left(\alpha + 2\alpha\beta + \tfrac{\gamma(1-\alpha)}{m}\right)\Delta_t + (\alpha+\alpha\beta+\gamma(1-\alpha))\left(\Delta_{t+1} - (F_{t+1} - F^*_{t+1})\right). \end{aligned} \tag{52}$$

Since data protection starts at time t (confidential data up to time $t-1$), for any time $T \geq t+1$ we have the same form as in the above equation only with different subscripts. Therefore, we can write the following recursion:

$$F_{T+1} - F^*_{T+1} = \left(\alpha + 2\alpha\beta + \frac{\gamma(1-\alpha)}{m}\right)\Delta_{T-1} + (\alpha+\alpha\beta+\gamma(1-\alpha))\left(\Delta_T - (F_T - F^*_T)\right). \tag{53}$$

Let $M \in \mathbb{R}_+$ such that $|A_n - P_n| = |\Delta_n| \leq M$ for all $n \in \mathbb{Z}_+$. Then (53) can be written up as

$$\begin{aligned} |F_{T+1} - F^*_{T+1}| &\leq \left(\alpha + 2\alpha\beta + \tfrac{\gamma(1-\alpha)}{m}\right) M + (\alpha + \alpha\beta + \gamma(1-\alpha))\left(M + |F_T - F^*_T|\right) \\ &= \left(2\alpha + 3\alpha\beta + \tfrac{\gamma(1-\alpha)(m+1)}{m}\right) M + (\alpha + \alpha\beta + \gamma(1-\alpha))|F_T - F^*_T| \end{aligned} \tag{54}$$

with the initial value $F_{t+1} - F^*_{t+1} = \alpha(1+\beta)\Delta_t$ as in (46). If we let $r = \alpha + \alpha\beta + \gamma(1-\alpha)$, then using the initial value and recursion, we can write the following

$$\begin{aligned} |F_{T+1} - F^*_{T+1}| &\leq \sum_{n=1}^{T-t} r^{n-1}\left[\left(2\alpha + 3\alpha\beta + \frac{\gamma(1-\alpha)(m+1)}{m}\right) M + r|F_{t+1} - F^*_{t+1}|\right] \\ &\leq \sum_{n=1}^{T-t} r^{n-1}\left(2\alpha + 3\alpha\beta + (\alpha+\alpha\beta)^2 + \gamma(1-\alpha)\left(\frac{m+1}{m} + \alpha + \alpha\beta\right)\right) M, \end{aligned} \tag{55}$$

which completes the proof. □

2.2. Changes to Actual Forecast Accuracy and Measured Forecast Accuracy

The Mean Absolute Error (MAE) and Mean Absolute Scaled Error (MASE) measure the actual forecast accuracy using the confidential data at a future time after the data has been protected. We average over $T-t$ time periods from time $t+1$ to T to compute forecast accuracy. Intuitively, the MASE is less than one when, on average, the MAE of a forecasting model is less than the MAE of the one-step Naïve model, where the one-step Naïve model is defined by using the last confidential value at time t to forecast the confidential value at time $t+1$. The MAE and MASE are defined as

$$\text{MAE} = \frac{1}{T-t}\sum_{i=t+1}^{T} |F_i - A_i|, \tag{56}$$

$$\text{MASE} = \frac{\frac{1}{T-t}\sum_{i=t+1}^{T} |F_i - A_i|}{\frac{1}{T-t}\sum_{i=t+1}^{T} |A_{i-1} - A_i|}, \tag{57}$$

where F_i is the forecast for time i made at time $i-1$ using the confidential data, A_i is the confidential value realized at time i, $t+1$ is the start of the forecast period, and T is the end of the forecast period. Note that the MAE and MASE can only be computed by the data provider, since A_i is confidential and never released to a forecaster. Also, we change the denominator of MASE slightly since data protection does not start until time t and it becomes more of a relative measure than the Hyndman and Koehler (2006) measure.

To evaluate the upper bound of the change in forecast accuracy due to data protection, the data provider measures the change in MAE (and MASE) between forecasts using the protected data and forecasts using the confidential data. The data provider substitutes F^*_i for F_i to compute ΔMAE and ΔMASE using the results in Section 2.1:

$$\Delta\text{MAE} = \frac{1}{T-t}\sum_{i=t+1}^{T} |F_i - A_i| - \frac{1}{T-t}\sum_{i=t+1}^{T} |F^*_i - A_i| \tag{58}$$

and

$$\Delta\text{MASE} = \frac{1}{\sum_{i=t+1}^{T} |A_{i-1} - A_i|} \left(\sum_{i=t+1}^{T} |F_i - A_i| - \sum_{i=t+1}^{T} |F_i^* - A_i| \right). \tag{59}$$

THEOREM 4 **(Change in Mean Absolute Error)**. *The change in MAE (Mean Absolute Error) is always bounded above by the average of absolute change to forecasts,* $\Delta MAE \leq \frac{1}{T-t} \sum_{i=t+1}^{T} |F_i - F_i^*|$.

Proof.

$$\begin{aligned}
\Delta\text{MAE} &= \frac{1}{T-t} \sum_{i=t+1}^{T} |F_i - A_i| - \frac{1}{T-t} \sum_{i=t+1}^{T} |F_i^* - A_i| \\
&= \frac{1}{T-t} \left(\sum_{i=t+1}^{T} |F_i - F_i^* + F_i^* - A_i| - \sum_{i=t+1}^{T} |F_i^* - A_i| \right) \\
&\leq \frac{1}{T-t} \left(\sum_{i=t+1}^{T} |F_i - F_i^*| + \sum_{i=t+1}^{T} |F_i^* - A_i| - \sum_{i=t+1}^{T} |F_i^* - A_i| \right) \\
&= \frac{1}{T-t} \sum_{i=t+1}^{T} |F_i - F_i^*|,
\end{aligned} \tag{60}$$

which completes the proof. □

The above theorem implies that for additive exponential smoothing models, the change in forecast accuracy is upper bounded by the choice of smoothing parameters α, β, γ, frequency m, and M, the upper bound of $|A_i - P_i|$ for all i . Equations (13), (30) and (55) can be used in time i to predetermine the exact bound for the change in MAE for a future time period $i+1$.

So far, we focused only on the change in forecast accuracy without considering the experience of the end user, a forecaster. Unlike the data provider, a forecaster only has access to the protected data P_i and the forecasts using the protected data, F_i^*. Hence, the forecaster only measures a forecast accuracy instead of realizing an actual forecast accuracy. We define the forecaster's Measured Mean Absolute Error (MMAE) and the Measured Mean Absolute Scaled Error (MMASE) similarly as follows:

$$\text{MMAE} = \frac{1}{T-t} \sum_{i=t+1}^{T} |F_i^* - P_i| \tag{61}$$

$$\text{MMASE} = \frac{\frac{1}{T-t} \sum_{i=t+1}^{T} |F_i^* - P_i|}{\frac{1}{T-t} \sum_{i=t+1}^{T} |P_{i-1} - P_i|}. \tag{62}$$

THEOREM 5 **(Measured Mean Absolute Error (MMAE))**. *The forecaster's Measured Mean Absolute Error (MMAE) is always bounded as* $MMAE \leq MAE - \Delta MAE + M$, *where* M, *the upper bound of* $|A_i - P_i|$ *for all* i, *can be considered the quality guarantee provided by the data provider. MMAE becomes smaller as* ΔMAE *in equation (60) increases, which acts in opposition to* M.

Proof. Using equations (56) and (58), we can write equation (61) as

$$\begin{aligned}
\text{MMAE} &= \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - P_i| \\
&= \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - A_i + A_i - P_i| \\
&\leq \tfrac{1}{T-t}\sum_{i=t+1}^{T}\{|F_i^* - A_i| + |A_i - P_i|\} \\
&= \text{MAE} - \Delta\text{MAE} + \tfrac{1}{T-t}\sum_{i=t+1}^{T}|A_i - P_i| \\
&\leq \text{MAE} - \Delta\text{MAE} + M,
\end{aligned} \tag{63}$$

which completes the proof. □

COROLLARY 1 **(Measured Mean Absolute Error (MMAE))**. *The forecaster's Measured Mean Absolute Error (MMAE) is always bounded as the following:*

$$MMAE \leq MAE + Average\ of\ |A_i - P_i| + Average\ absolute\ change\ to\ forecasts.$$

Proof. Equation (61) can be written up as

$$\begin{aligned}
\text{MMAE} &= \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - P_i| \\
&= \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - A_i + A_i - P_i| \\
&\leq \tfrac{1}{T-t}\sum_{i=t+1}^{T}\{|F_i^* - A_i| + |A_i - P_i|\} \\
&= \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - A_i| + \tfrac{1}{T-t}\sum_{i=t+1}^{T}|A_i - P_i| \\
&= \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - F_i + F_i - A_i| + \tfrac{1}{T-t}\sum_{i=t+1}^{T}|A_i - P_i| \\
&\leq \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i^* - F_i| + \tfrac{1}{T-t}\sum_{i=t+1}^{T}|F_i - A_i| + \tfrac{1}{T-t}\sum_{i=t+1}^{T}|A_i - P_i| \\
&= \text{Average absolute change to forecasts} + \text{MAE} + \text{Average of } |A_i - P_i|,
\end{aligned} \tag{64}$$

which completes the proof. □

3. Data Provider: Bounding Absolute Change to Forecasts with Data Protection

At time t, the data provider uses a data protection method to release protected data $P_{j,t}$ for each time series j based on the confidential values of all time series up until time t. Standard DPMs previously mentioned in the introduction are formulated in Table 2. As Table 2 shows, the main issue with these methods is that they choose protected values based on predefined rules and not absolute change to forecasts. However, the goal of the data provider should be to change $A_{j,t}$ to $P_{j,t}$ with the minimal absolute change to forecasts while also increasing privacy to an acceptable threshold.

In this section, we design the k-nTS (k-nearest time series) Swapping method to alter the confidential data using randomization to balance the trade-off between absolute change to forecasts and privacy. Depending on the quantity of available data, both k-nTS Swapping and standard DPMs can use rolling windows of data. Rolling windows adjust for dynamic changes in the relationships between time series. For example, if we choose a rolling window size, say n, then $x_j = (A_{j,t-n+1}, A_{j,t-n+2}, \ldots, A_{j,t-1}, A_{j,t})^T$ where $x_j \in \mathbb{R}^n$. Also, define $F(x_j)$ as the empirical cumulative

Table 2 Standard Data Protection Methods at Time t

Data Protection Method	**Description**	**Formulation**
None	Release confidential observation	$P_{j,t} = A_{j,t}$
Bottom-Coding	Bottom p percent of observations are replaced with the p quantile	$P_{j,t} = \begin{cases} B \text{ if } A_{j,t} \leq B \\ A_{j,t} \text{ if } A_{j,t} > B \end{cases}$ where $B = \inf\{x_j \in \mathbb{R} \mid F(x_j) \geq p\}$
Top-Coding	Top p percent of observations are replaced with the $1-p$ quantile	$P_{j,t} = \begin{cases} T \text{ if } A_{j,t} \geq T \\ A_{j,t} \text{ if } A_{j,t} < T \end{cases}$ where $T = \sup\{x_j \in \mathbb{R} \mid F(x_j) \leq p\}$
Additive Noise	Add a normal random number with mean zero and standard deviation σ	$P_{j,t} = A_{j,t} + r$, where $r \sim N(0, \sigma^2)$ and $\sigma^2 = E[x_j - E[x_j]^2]$

distribution function of x_j. Protection in subsequent time periods from $t+1$ to T rolls x_j forward from $x_j = (A_{j,t-n+2}, A_{j,t-n+3}, \ \ldots, A_{j,t}, A_{j,t+1})^T$ to $(A_{j,T-n+1}, A_{j,T-n+2}, \ \ldots, A_{j,T-1}, A_{j,T})^T$, respectively. Compared to standard DPMs, k-nTS Swapping matches similarly patterned time series together based on the distance of a rolling window of their past values. Then, it uses randomization to replace $A_{j,t}$ with a confidential value from a similar time series. Additionally, we design the k-means Time Series (k-mTS) Shuffling method in order to simultaneously swap multiple time series data. Subsections 3.1 and 3.2 develop the k-nTS Swapping method and k-mTS Shuffling methods, respectively.

3.1. The k-nearest Time Series (nTS) Swapping method

Let $\mathbb{X} = \{x_1, \ldots, x_J\}$ be a given set of time series data (n-vectors). For each time series $x_j \in \mathbb{R}^n$, the data provider computes a set of squared distances of the elements of the set $\mathbb{X}$. Let us define $\mathbf{dist}(x_j, x_i) = d_{j,i}$ as the distance between x_j and x_i, i.e., two distinct time series data from a given set $\mathbb{X}$. Without loss of generality, we use the Euclidean norm, or ℓ^2-norm for this paper (all norms on $\mathbb{R}^n$ are equivalent to the Euclidean norm) as a distance metric. Since our case is multivariate and partially ordered, we can get a totally ordered set based on the Euclidean distance.

Let us define $x_j^{(k)}$ as the kth nearest neighbor of x_j. Then, for a time series x_j, we have $\{d_{j,(1)}, \ d_{j,(2)}, \ldots, d_{j,(J-1)}\}$ such that $d_{j,(k)} \leq d_{j,(l)}$ for any integers $k < l$ where $d_{j,(k)} = \|x_j - x_j^{(k)}\|$. Note that $x_j^{(i)} \in \mathbb{X} \backslash \{x_j\}$ and the superscript (i) means the i^{th} order statistic of the related Euclidean distances from x_j. Thus, for a given time series vector x_j, its k-nTS (k-nearest time series) can be represented as the set $K_j = \{x_j^{(1)}, \ldots, x_j^{(k)}\}$ based on $\|x_j - x_j^{(1)}\| \leq \cdots \leq \|x_j - x_j^{(k)}\|$ or an ordered set $\{d_{j,(1)}, \ d_{j,(2)}, \ldots, d_{j,(k)}\}$.

For more efficient computation for such ordering, let us introduce a distance matrix D using the squared distances. The squared distance between x_i and x_j is given by $d_{i,j} = \|x_i - x_j\|^2$, and $d_{i,j}$ is the (i,j)th entry of D. Hence D is symmetric. (Also note that $\text{rank}(D) \leq n+2$.) For our applications, the vector $x_j \in \mathbb{R}^n$, $j = 1, \ldots, J$ where $n \ll J$, typically. Suppose that we are given a data matrix $X = [x_1, x_2, \ldots, x_J]$, $x_j \in \mathbb{R}^n$ (i.e., $X \in \mathbb{R}^{n \times J}$). We can write such data matrix X, i.e., a confidential data matrix, as the following.

$$X = [x_1, x_2, \ldots, x_J] = \begin{pmatrix} A_{1,t-n+1} & A_{1,t-n+2} & A_{1,t-n+3} & \cdots & A_{1,t-1} & A_{1,t} \\ A_{2,t-n+1} & A_{2,t-n+2} & A_{2,t-n+3} & \cdots & A_{2,t-1} & A_{2,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ A_{J,t-n+1} & A_{J,t-n+2} & A_{J,t-n+3} & \cdots & A_{J,t-1} & A_{J,t} \end{pmatrix}^T, \tag{65}$$

where $x_j = (A_{j,t-n+1}, A_{j,t-n+2}, \ldots, A_{j,t-1}, A_{j,t})^T$ where $x_j \in \mathbb{R}^n$ and also $x_j \in \mathbb{X}$.

Note that we can find the distance matrix D using the fact that $\|x_i - x_j\|^2 = (x_i - x_j)^T(x_i - x_j) = x_i^T x_i - x_i^T x_j - x_j^T x_i + x_j^T x_j$, which can be written up as the following:

$$D = \mathbf{1} diag(X^T X)^T - 2X^T X + diag(X^T X)\mathbf{1}^T, \tag{66}$$

where the symbol $\mathbf{1}$ denotes a column vector of J ones. It is easy to see that the column vector $diag(X^T X) = (\|x_1\|^2, \ldots, \|x_J\|^2)^T$. Let d_j denote the jth column of D. Then we can write the $J \times J$ distance matrix $D = [d_1, \ldots, d_J]$, where $d_j \in \mathbb{R}^J$.

In the general case $k \ll J$, for each time series x_j we sort d_j, the jth column of D from the smallest to largest components and find the kth smallest component so that we have

$$K_j = \{x_j^{(1)}, \ldots, x_j^{(k)}\}. \tag{67}$$

That is, the data provider then selects a value of k from 1 to a maximum of $J-1$ and selects the k-nearest time series. In case of $k = J$, random swapping is simply done by rearranging the components in the last row of matrix X. Let the ith nearest time series from x_j be $x_j^{(i)} = (A_{j,t-n+1}^{(i)}, A_{j,t-n+2}^{(i)}, \ldots, A_{j,t}^{(i)})^T$ where n is the length of the rolling window of past data. Then the swap of the last component of x_j with the last component of one of its k-nearest time series $x_j^{(i)}$, $i = 1, \ldots, k$ can simply be written as the following random swapping:

$$k - \text{nTS Swapping}: \; P_{j,t} = \begin{cases} A_{j,t}^{(1)} \text{ with probability } \frac{1}{k} \\ \vdots \\ A_{j,t}^{(k)} \text{ with probability } \frac{1}{k}, \end{cases} \tag{68}$$

which is equivalent to the following: the last component of x_j is randomly replaced by the last component of $x_j^{(i)} \in K_j$ with probability $\frac{1}{k}$ for $i = 1, \ldots, k$.

Algorithm 1 The k-nTS Swapping method

Require: [Initialization]

(i) [Time Series Matrix $X \in \mathbb{R}^{n\times J}$] $X = [x_1, x_2, \ldots, x_J]$, $x_j \in \mathbb{R}^n$ for $j = 1, \ldots, J$ as in (65).

(ii) [Distance Matrix D] $D = \mathbf{1}diag(X^TX)^T - 2X^TX + diag(X^TX)\mathbf{1}^T$ as in (66).

for $j = 1, 2, \ldots, J$ **do**

[Finding a set K_j for x_j] Let d_j denote the jth column of D. Sort d_j from the smallest to largest components and find the kth smallest component, followed by K_j as in (67).

[Random swapping] $x_j \leftarrow x_j^{(i)}$ (last components only) for some $i \in \{1, \ldots, k\}$ as in (68).

end for

By Algorithm 1, we can obtain X': a matrix of protected time series data at time point t for all J time series for a given rolling window size n. The k-nearest time series data protection method can be written up as the following protected data matrix

$$X' = [x'_1, x'_2, \ldots, x'_J] = \begin{pmatrix} A_{1,t-n+1} & A_{1,t-n+2} & A_{1,t-n+3} & \cdots & A_{1,t-1} & P_{1,t} \\ A_{2,t-n+1} & A_{2,t-n+2} & A_{2,t-n+3} & \cdots & A_{2,t-1} & P_{2,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ A_{J,t-n+1} & A_{J,t-n+2} & A_{J,t-n+3} & \cdots & A_{J,t-1} & P_{J,t} \end{pmatrix}^T . \tag{69}$$

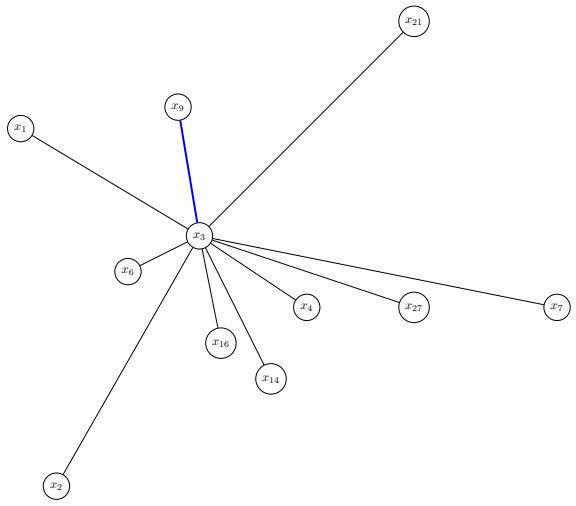
Remark 2 (Time complexity). Let $\mathbb{X} = \{x_1, \ldots, x_J\}$ be a given set of time series data (n-vectors). The time complexity to find k-nearest neighbors of each one of x_j's can be calculated by the following. First, the cost for the computation of distances $\|x_i - x_j\|$ for $j = 1, \ldots J$, $j \neq i$ is $3(J-1)n$ ($2(J-1)n$ if we use squares of distances), in either case it would be $O(Jn)$. In order to find the k-nearest neighbors we need to sort the calculated distances and it would be $O(J \log J)$. Since we typically have $J \gg n$, the cost of computation is $O(J \log J)$ for each x_j. If we use k-d tree algorithm (e.g., see Mahajan et al. (2009) and references therein), the whole process can be done in $O(J \log J)$.

We can represent each time series x_j, $j = 1, \ldots, J$ as a vector, and then put them in the graph $G = (V, E)$, which consists of a set V of vertices (or nodes) and a set E of undirected edges. In our case, we can use weighted edges to represent the Euclidean distance between associated time series vectors: $w_{i,j} = \mathbf{dist}(x_i, x_j)$. If we put all the nodes on the graph and assign weight on every edge (every pair of nodes), e.g., $w_{i,j} = \mathbf{dist}(x_i, x_j)$ for all $i \neq j$, then we will have a complete graph. As described in Figure 1, the k-nearest time series swapping method can be considered as a random edge selection problem of the graph. Figure 1 depicts the case of $k = 10$ for the k-nearest time series of x_3, where the last component of x_3 is swapped with that of x_9.

3.2. The k-means Time Series (mTS) Shuffling method

In this subsection, we treat the protection of each time series in terms of a group of time series rather than individuals. In doing so, we simultaneously swap multiple data for more efficient data

Figure 1 Random edge selection for k-nTS Swapping



Note. For a given time series x_3, its last component is replaced by the last component of x_9, that is randomly selected among x_3's 10-nearest time series data. $K_3 = \{x_1, x_2, x_4, x_6, x_7, x_9, x_{14}, x_{16}, x_{21}, x_{27}\}$.

protection. For effective protection, we use a clustering algorithm to partition given time series into multiple groups of similar ones. There are clustering algorithms widely used in practice, e.g., k-means (Lloyd (1982)), hierarchical (Corpet (1988)), isolation forest (Liu et al. (2008)) clustering methods, etc. Clustering and shuffling for data privacy has also been studied, e.g., Muralidhar and Sarathy (2006), Li and Sarkar (2013) and references therein.

There are J points in n-dimensional space, i.e., $x_j \in \mathbb{R}^n, j = 1, \ldots, J$. Our confidential data matrix is $X = [x_1, x_2, \ldots, x_J]$ as in (65). Our goal is to partition those points into k-means clusters (Lloyd (1982), first proposed in 1957), where there are k centroids $c_1, \ldots, c_k$. Each centroid c can be found by the optimization problem: $\min \sum \|c - x_j\|^2$, which minimizes the total distance to points $x_j = (A_{j,t-n+1}, \ldots, A_{j,t-1}, A_{j,t})^T$ in its cluster.

For k-means clustering, we minimize the following:

$$\min E = \sum_{i=1}^{k} \sum_{j \in Q_i} \|x_j - c_{Q_i}\|^2, \tag{70}$$

where the clusters Q_i have centers c_{Q_i} for $i = 1, \ldots, k$. The centroid c_i is the average $c_{Q_i} = \left(\sum_{j \in Q_i} x_j\right) / |Q_i|$ of the vectors in cluster Q_i.

It is well-known that the optimal clustering is an NP-hard problem even for smallest problems (see Mahajan et al. (2009)). In our applications, we start from a given set of vectors: k centroids (representatives), selected by a data provider or by random sampling, which is followed by parti-

Algorithm 2 Clustering using k-representatives

Require: [Initialization] A list of J vectors $x_1, \ldots, x_J$ and an initial list of k group representative vectors $c_1, \ldots, c_k$ out of the given J time series (given by a data provider or random sampling).

for the fixed representatives $c_i, i = 1, 2, \ldots, k$ **do**

[Partitioning vectors]

for all $x_j, j = 1, \ldots, J$ **do**

[Assignment] $\min\limits_{i=1,\ldots,k} \|x_1 - c_i\|^2 + \cdots + \min\limits_{i=1,\ldots,k} \|x_J - c_i\|^2$

end for

[Finding k clusters $Q_i, i = 1, \ldots, k$]

Based on the vector assignment by partitioning, we find the corresponding clusters by

$Q_i = \{x_j \mid \min\limits_{j=1,\ldots,J} \|x_j - c_i\|^2\}$

end for

tioning into k clusters (as in Algorithm 2). Suppose that, by the k-representative partitioning, we can obtain n_i points for each cluster $Q_i, i = 1, \ldots, k$ such as

$$Q_i = \{x_{1_i}, \ldots, x_{n_i}\}, \tag{71}$$

where $x_{1_i}, \ldots, x_{n_i}$ are the elements of the cluster i. Note that $\sum_{i=1}^{k} n_i = J$. If we put the points in a matrix as columns, then the matrix $\mathbf{Q}_i = [x_{1_i}, x_{2_i}, \ldots, x_{n_i}],\ x_{j_i} \in \mathbb{R}^n$ and the matrix $\mathbf{Q}_i \in \mathbb{R}^{n \times n_i}$.

For each cluster Q_i, there is an array between 1 and n_i (columns of matrix $\mathbf{Q}_i$). For random shuffling, we generate a random permutation, from which we get an n_i by n_i permutation matrix $\mathbf{P_i}$. (simply by permuting columns of the n_i by n_i identity matrix.) By Algorithm 3, we can obtain a protected data matrix X_i' for cluster i on a given rolling window size n. The k-means time series clustering and shuffling data protection method can be written up using the following protected data matrix for cluster i:

$$X_i' = [x'_{1_i}, x'_{2_i}, \ldots, x'_{n_i}] = \begin{pmatrix} A_{1_i,t-n+1} & A_{1_i,t-n+2} & A_{1_i,t-n+3} & \cdots & A_{1_i,t-1} & P_{1_i,t} \\ A_{2_i,t-n+1} & A_{2_i,t-n+2} & A_{2_i,t-n+3} & \cdots & A_{2_i,t-1} & P_{2_i,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ A_{n_i,t-n+1} & A_{n_i,t-n+2} & A_{n_i,t-n+3} & \cdots & A_{n_i,t-1} & P_{n_i,t} \end{pmatrix}^T, \tag{72}$$

where the row vector $(P_{1,t}, P_{2,t}, \ldots, P_{n_i,t})$ is the last row vector of $\mathbf{Q}_i\mathbf{P_i}$.

According to the partition by Algorithm 2, let us rearrange the original data matrix $X = [x_1, x_2, \ldots, x_J]$ defined in (65) into $\mathbf{X} = [\mathbf{Q_1}, \mathbf{Q_2}, \ldots, \mathbf{Q_k}]$, where each data matrix $\mathbf{Q_i} \in \mathbb{R}^{n \times n_i}$ con-

Algorithm 3 the k-mTS Shuffling method

Require: [Initialization] k clusters $Q_i, i = 1, \ldots, k$. (by Algorithm 2)

for all $x_j \in Q_i$ **do**

[Random permutation] Construct a data matrix $\mathbf{Q}_i \in \mathbb{R}^{n \times n_i}$ using $x_j \in Q_i$ as columns. ($|Q_i| = n_i, i = 1, \ldots, k$.) For each cluster Q_i, create an array between 1 and n_i and then generate a random permutation, followed by an n_i by n_i permutation matrix $\mathbf{P_i}$.

[Random shuffle] Random shuffling can be done by $\mathbf{Q}_i\mathbf{P_i}$, from which we use the last row for a protected data matrix X_i' for $i = 1, \ldots, k$.

end for

sists of time series vectors in the corresponding cluster i for $i = 1, \ldots, k$. Then we can write the following.

$$X'' = \mathbf{XP} = [\mathbf{Q_1}, \mathbf{Q_2}, \ldots, \mathbf{Q_k}] \begin{pmatrix} \mathbf{P_1} & & & \\ & \mathbf{P_2} & & \\ & & \ddots & \\ & & & \mathbf{P_k} \end{pmatrix} = [\mathbf{Q_1P_1}, \mathbf{Q_2P_2}, \ldots, \mathbf{Q_kP_k}], \tag{73}$$

where $\mathbf{P}$ is a block diagonal matrix with "randomly generated" permutation matrices $\mathbf{P_i}$ for $i = 1, \ldots, k$ (with all off-diagonal blocks being zero matrices). If we take the last row (i.e., the n-th row) of X'', with which replace that of $\mathbf{X}$, then a new matrix $\mathbf{X}' \in \mathbb{R}^{n \times J}$ can be constructed as a protected data matrix:

$$\mathbf{X}' = [X_1', X_2', \ldots, X_k'] = \begin{pmatrix} A_{1_1,t-n+1} & A_{1_1,t-n+2} & A_{1_1,t-n+3} & \cdots & A_{1_1,t-1} & P_{1_1,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n_1,t-n+1} & A_{n_1,t-n+2} & A_{n_1,t-n+3} & \cdots & A_{n_1,t-1} & P_{n_1,t} \\ A_{1_2,t-n+1} & A_{1_2,t-n+2} & A_{1_2,t-n+3} & \cdots & A_{1_2,t-1} & P_{1_2,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n_2,t-n+1} & A_{n_2,t-n+2} & A_{n_2,t-n+3} & \cdots & A_{n_2,t-1} & P_{n_2,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{1_k,t-n+1} & A_{1_k,t-n+2} & A_{1_k,t-n+3} & \cdots & A_{1_k,t-1} & P_{1_k,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n_k,t-n+1} & A_{n_k,t-n+2} & A_{n_k,t-n+3} & \cdots & A_{n_k,t-1} & P_{n_k,t} \end{pmatrix}^T, \tag{74}$$

where the row vector $(P_{1_1,t}, \ldots, P_{n_k,t}) \in \mathbb{R}^J$ is from the last row vector (the n-th row) of X'', and by shuffling of time series data in the same cluster, we obtain the protected data matrices X_i''s for clusters $i = 1, \ldots, k$.

In this section, we studied two DPMs (data protection methods) for data provider. Using these new DPMs, we develop novel mathematical models and decision making framework in Section 5, incorporating data intruder's utilities in Section 4.

4. Data Privacy: Anomaly Detection and Targeting Extreme Values

We quantify the privacy of protected data by defining a data intruder's utility function which represents the satisfaction the data intruder receives from discovering privacy issues at time i. As noted in the previous sections, we assume that data protection begins at time t, and therefore, we can write confidential time series j as an n-vector $x_j = (A_{j,i-n+1}, ..., A_{j,t-1}, A_{j,t}, \ldots, A_{j,i})^T$ and protected time series j as an n-vector $x'_j = (A_{j,i-n+1}, ..., A_{j,t-1}, P_{j,t}, \ldots, P_{j,i})^T$.

Defining Privacy Issues. The data provider uses anomaly pattern detection at time i on the confidential data $x_j = (A_{j,i-n+1}, ..., A_{j,t-1}, A_{j,t}, \ldots, A_{j,i})^T$ to define privacy issues within a time series. Privacy issues exist at time i if $u(A_{j,i}) > \tau^*$ for $j = 1, \ldots, J$ where $u(\cdot)$ is a utility function of data intruders, defined in Sections 4.1 and 4.2. The threshold τ^* can be chosen by the data provider. The threshold is based on a quantile $q = \hat{F}^{-1}_{x_j}(p)$, where $\hat{F}_{x_j}$ denotes an estimated CDF and p is 0.95, for example. When defining a privacy issue, we set $g_{j,i} = 1$ and otherwise, $g_{j,i} = 0$. Thus, the total number of privacy issues at time i is $G_i = \sum_{j=1}^{J} g_{j,i}$.

4.1. Data Intruder 1's Utility: Anomaly Detection Within Individual Time Series

Anomaly pattern detection within a protected time series $x'_j = (A_{j,i-n+1}, ..., A_{j,t-1}, P_{j,t}, \ldots, P_{j,i})^T$ at time i can be thought of as the surprise of the new value $P_{j,i}$ compared to the past patterns (e.g., seasonality adjustments, levels) within the same times series. To define surprise for series j, we fit a kernel density $\hat{f}_{x'_j}$ on the past $n-1$ observations up until time $i-1$ (each time series as an n-vector). We then estimate the density of the new protected value at time i to get a density value for $P_{j,i}$. For kernel density estimation and related topics we refer the readers to relevant literature, e.g., Azzalini (1981), Jones et al. (1996), Chen (2017) and references therein. If the density $\hat{f}_{x'_j}(P_{j,i})$ is small, this is defined as an anomaly and a privacy issue which increases the data intruder's utility. We can define the data intruder's utility for time series j at time i as the following:

$$u(P_{j,i}) = \sqrt{\frac{1}{\hat{f}_{x'_j}(P_{j,i})}}. \tag{75}$$

With some positive constant τ as a threshold, data intruders may flag x_j as anomalous if the utility function value in (75) is above the threshold: time series x'_j is marked as anomaly if $u(P_{j,i}) > \tau$ for $j = 1, \ldots, J$. This means, it looks like a good target for data intruder to attack. Such threshold can be based on a quantile $q = \hat{F}^{-1}_{x'_j}(p)$, where $\hat{F}_{x'_j}$ denotes an estimated CDF and p is large, e.g., 0.95, 0.99, etc. In this regards, for the unimodal case, a reasonable confidence interval would also be useful to target anomalous data since points outside such interval can be identified as anomalies.

4.2. Data Intruder 2's Utility: Anomaly Detection for Multiple Time Series

A data intruder may also want to simultaneously find anomalous time series, instead of investigating individual data points within time series. In this case, it would be helpful to partition all the time

series into multiple groups of similar time series, followed by anomaly detection on each group. Let us consider k-means clustering here. Suppose that there are k groups of the data, characterized by group representatives such as centroids, $c_1, \ldots, c_k$. For each cluster $Q_h \in \mathbb{R}^n, h = 1, \ldots, k$, the following metric may be useful for anomaly detection

$$g_h(x) = e^{-\|x - c_h\|^2 / \sigma_h^2}, \tag{76}$$

where $x, c \in \mathbb{R}^n$ and σ_h denotes standard deviation of data points in cluster Q_h. Note that the metric of (76) is always between 0 and 1, and its value gets closer to zero when the distance between x and c_h gets larger. Therefore, data intruders may want to use some small positive constant $0 < \epsilon \ll 1$ as a threshold, in order to target the following anomaly:

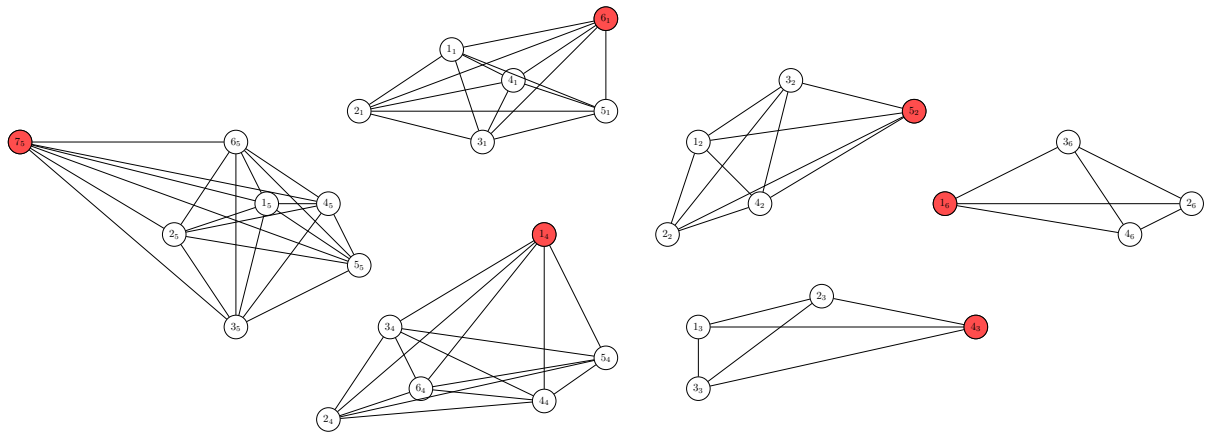
$$\{y \in Q_h \mid g_h(y) < \epsilon\}. \tag{77}$$

Note that (76) and (77) can be used for the general case, and it can easily be modified, e.g., the vectors x and c are both replaced by $\lambda^T x$ and $\lambda^T c$, $\lambda = (\lambda_1, \ldots, \lambda_n)^T \in \mathbb{R}^n_+$, $\|\lambda\|_1 = 1$. We can write the following

$$g_h(x) = e^{-\|\lambda^T x - \lambda^T c_h\|^2 / \sigma_h^{*2}}, \tag{78}$$

where σ_h^* denotes a standard deviation of $\lambda^T x, x \in Q_h$. In order to target any surprise of the new value $P_{j,i}$ at time i, we can simply use $\lambda_i = 1$ with all other components being zero.

Figure 2 Description of targeting across time series



Note. Seven clusters. Each edge has a value, e.g., distance or weighted distance between nodes for each pair.

4.3. Targeting Decisions of the Data Intruder

Define a decision at time i for time series j at as

$$\text{Decision}_{i,j} = \begin{cases} 1 & \text{if } u(P_{j,i}) > \tau \\ 0 & \text{if } u(P_{j,i}) \leq \tau, \end{cases} \tag{79}$$

where $u(\cdot)$ is defined as in (75) for targeting extreme values within time series, and for targeting anomalous time series we have the following:

$$\text{Decision}_{i,j} = \begin{cases} 0 & \text{if } g_h(x_j) \geq \epsilon \\ 1 & \text{if } g_h(x_j) < \epsilon, \end{cases} \tag{80}$$

where $g_h(\cdot)$ is defined in (78), $x_j \in Q_h$ for $h = 1, \ldots, k$. Let us define the likelihood ratio for threshold τ as

$$\text{LR}_\tau = \frac{\text{TPR}_\tau}{\text{FPR}_\tau} \tag{81}$$

where TPR_τ is the true positive rate and FPR_τ is the false positive rate. If a set of targeting decisions is better than random guessing, then $\text{LR}_\tau > 1$. The data provider attempts to decrease the maximum value of LR_τ over all values of τ subject to a minimum FPR_τ to avoid division by zero.

Table 3 illustrates all possible decisions over the values of τ. The left side of the figure represents the targeting of no time series when τ is greater than or equal to all values of $u(P_{j,i})$. The right side of the figure represents the targeting of all time series when τ is lower than all values of $u(P_{j,i})$, as in (79).

Table 3 **Decisions at time i**

Time Series	Protected Data	Decision(τ_1)	Decision(τ_2)	Decision(τ_3)	Decision(τ_4)	$\cdots$	Decision($\tau_{\min}$)
1	$P_{1,i}$				Target		Target
2	$P_{2,i}$		Target	Target	Target		Target
3	$P_{3,i}$			Target	Target		Target
4	$P_{4,i}$						Target
5	$P_{5,i}$			Target	Target		Target
6	$P_{6,i}$				Target		Target
7	$P_{7,i}$						Target
$\vdots$	$\vdots$						$\vdots$
J	$P_{J,i}$						Target

Decisions are made based on τ. τ_i and $\tau_1 > \tau_2 > \cdots > \tau_{\min}$.

For each value of τ, there is a true positive $\text{TP}_{i,j} = 1$ when $u(P_{j,i}) > \tau$ and $g_{j,i} = 1$, and there is a false positive $\text{FP}_{i,j} = 1$ if $u(P_{j,i}) \leq \tau$ and $g_j = 0$. The corresponding TPR_τ and FPR_τ are:

$$\text{TPR}_\tau = \frac{\sum_{j=1}^{j=J} \text{TP}_{i,j}}{G}, \tag{82}$$

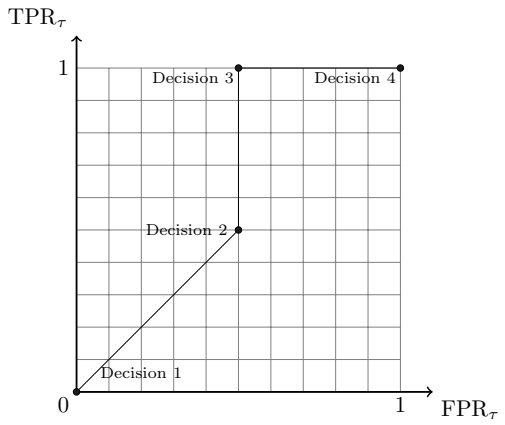
$$\text{FPR}_\tau = \frac{\sum_{j=1}^{j=J} \text{FP}_{i,j}}{J-G}. \tag{83}$$

For a decision analysis of the data intruder, the above equations can be used to compute the LR_τ in order to limit the maximum value, as previously mentioned.

For a more comprehensive decision analysis, the combination of all values of FPR_τ and TPR_τ can be represented in a Receiver Operating Characteristic (ROC) Curve. The area under the ROC Curve is the probability that a randomly chosen privacy issue ($g_{j,i}=1$) has a higher $u(P_{j,i})$ than a randomly chosen non-privacy issue ($g_{j,i}=0$). The data provider may also assume the data intruder has a limited budget and only targets anomalies up to a predetermined FPR limit=r, resulting in a Partial Area Under the ROC Curve (PAUC). PAUC_i is piece-wise linear and $0 \leq \frac{\text{PAUC}_i}{r} \leq 1$. Best-case privacy (random targeting) occurs when TPR_τ =FPR_τ for all values of τ and $\text{PAUC}_i = \frac{1}{2}r^2$. Worst-case privacy occurs when $\text{FPR}_\tau = 0$ for all decisions and $\text{PAUC}_i = r$.

For example, Figure 3 and Table 4 show the case of $J=4$ time series where $0.01 < u(P_{1,i}) < 0.40 < u(P_{2,i}) < 0.50 < u(P_{3,i}) = u(P_{4,i}) < 0.75$ which results in 4 possible values of τ. Only one set of decisions with $\tau = 0.40$ beats random targeting and has TPR_τ >FPR_τ. When $r=1$, $\text{PAUC}_i = \frac{5}{8}$ which has worse privacy than random targeting with $\text{PAUC}_i = \frac{1}{2}r^2 = \frac{1}{2}$. When $\tau = 0.40$, the set of decisions also has the highest LR_τ (worst privacy, but the best utility for data intruder) with $LR_\tau = \frac{1}{0.5} = 2$.

Figure 3 ROC Curve at time i



5. Two-Party Framework: Optimality for Effective Data Protection

The involved parties in this framework are: the data provider who chooses how to protect the data prior to release; the data intruder who seeks to attack the protected data at time i. The data intruder can be a data provider's employee, a contracted third-party vendor, or a hacker. The

Table 4 **Decisions at time i**

Time Series j	Protected Data	Decision	Privacy Issue
1	$P_{1,i}$	-	No
2	$P_{2,i}$	Target	Yes
3	$P_{3,i}$	Target	Yes
4	$P_{4,i}$	Target	No

Decisions are made based on τ with the lowest LR_τ.

primary goals of the data provider are to (a) ensure a forecaster can achieve similar forecasts with the protected data at time i, and (b) prevent the data intruder from targeting time series j at time i. The two-party data privacy framework should be used by data providers to decide the optimal protection k while engaged in the practice of forecasting.

5.1. Optimal time series selection for the k-nTS swapping using fixed-radius δ

As described in Figure 1 in Section 3.1, the k-nTS swapping can be considered a random edge selection problem for given k neighbors of each time series $j = 1, \ldots, J$. Since the selection of k plays an important role in both forecasting loss and data privacy perspectives, we need to be able to find an optimal k (i.e., optimal set K) before random swapping. For each time series $j \in J$, we can find the optimal k by the following :

$$\begin{array}{l} \max k \\ \text{subject to} \\ \|x_j - x_{(k)}\| \leq \delta \\ \|x_j - x_{(k)}\|_\infty \leq M, \end{array} \tag{84}$$

where $\delta > 0$ and $M > 0$ are given constants determined by a data provider. The first constraint is to find k neighboring time series of $x_j \in \mathbb{R}^n$ within a given radius δ from x_j, and the second constraint requires to find time series in such a way that maximum of absolute values of components of a vector $x_j - x_{(k)}$ is bounded above by M. Let $\mathbf{x}^{(j_k)} = x_j - x_{(k)}$. Then the second constraint can be written up as $\|\mathbf{x}^{(j_k)}\|_\infty = \max_i |\mathbf{x}_i^{(j_k)}| \leq M$, where $\mathbf{x}_i^{(j_k)}$ denotes the ith component of a vector $x_j - x_{(k)}$. This means that, using (84), data providers can put an upper bound M for absolute change to forecasts because the second constraint is equivalent to $|A_t - P_t| \leq M$ for all $t = 1, \ldots, n$.

This problem formulation always provides us with the best k with a given δ. This problem formulation gives us different k for each time series $j = 1, \ldots, J$, which is ideal, since k-nearest neighbors of x_j may include points too far from most of neighbors of x_j. This is because the set K_j for x_j (as defined in (67)) is based on the ordering, no matter how far from the given data point. Swapping with such irrelevant time series for data protection may result in an unacceptable absolute change to forecasts level. For this reason, it would be useful to set an upper bound on the distance from x_j only to include neighbors with acceptable absolute change to forecasts levels.

Algorithm 4 The fixed-radius δ-nTS Swapping method

Require: [Initialization] Same as in Algorithm 1.

for $j = 1, 2, \ldots, J$ **do**

[Component sorting] Sort d_j from the smallest to largest, and let $D_j = \{d_j^{(1)}, \ldots, d_j^{(J-1)}\}$.

[Finding δ-near time series] Find $d_j^{(k)} = \sup\{d \in d_j \mid d < \delta^2\}$ and let $B_j = \{x_j^{(1)}, \ldots, x_j^{(k)}\}$.

[Random swapping] (last components) $x_j \leftarrow x_j^{(i)} \in B_j = \{x_j^{(1)}, \ldots, x_j^{(k)}\}$ with probability $1/k$.

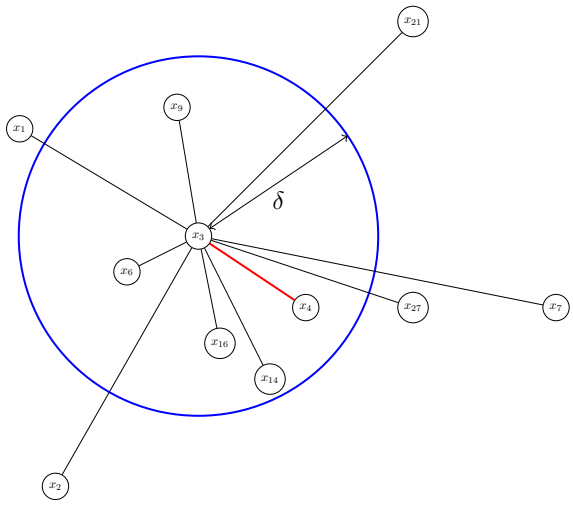
end for

Using the best k from (84), we do a random edge selection from a fixed-radius δ-near time series (as in Algorithm 4). Let us propose to topologize n-dimensional Euclidean space using open ball as the following:

$$\text{Fixed-radius } \delta\text{-near time series } x_i \in B_\delta(x_j) = \{y \in \mathbb{R}^n \mid \mathbf{dist}(x_j, y) < \delta\}, \tag{85}$$

where $x_i, x_j \in \mathbb{R}^n$ but $1 \leq i \neq j \leq J$. There is no problem also using subspace topology in case of weighted time series data only on certain time points. Figure 4 depicts the time series data within $B_\delta(x_3) = \{y \in \mathbb{R}^n \mid \mathbf{dist}(x_3, y) < \delta\} = \{x_4, x_6, x_9, x_{14}, x_{16}\}$, with which x_3 can be randomly swapped (summarized in Algorithm 4).

Figure 4 **Fixed-radius δ-near time series**



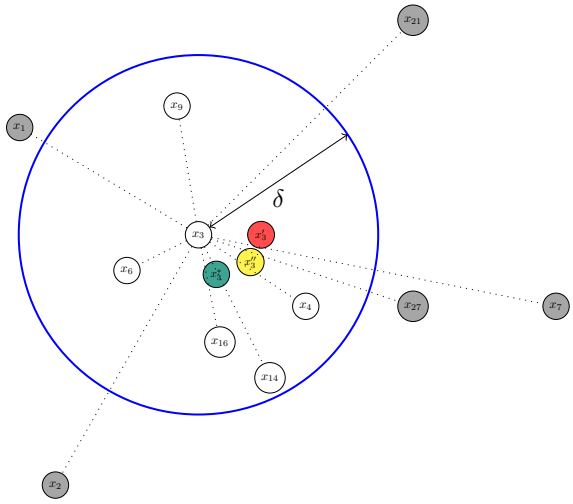
Note. The set $\{x_4, x_6, x_9, x_{14}, x_{16}\}$ consists of the time series within $B_\delta(x_3) = \{y \in \mathbb{R}^n \mid d(x_3, y) < \delta\}$. Note that we use the same example of Figure 1 that is the case of $k = 10$ for the k-nearest neighbors of x_3.

It is also reasonable to swap with artificial time series data instead of a confidential data. Let K_j denote the set of k-nearest neighbors of x_j, i.e., $|K_j| = k$. We can use a centroid of the set of vectors,

say c_j for swapping with x_j at time i. That is, we have the protected data matrix $X' = [x'_1, \ldots, x'_J]$, where $x'_j = (x_{j,1}, \ldots, x_{j,n-1}, c_{j,n})^T$ for $j = 1, \ldots, J$ for the confidential data matrix $X = [x_1, \ldots, x_J]$.

For description (see Figure 5), we consider, among others, three different representatives regarding K_j, i.e., the set of k-nearest neighbors of x_j: (i) centroid of the vertices of conv(K_j), the convex hull of K_j; (ii) centroid of the vectors in the K_j; (iii) centroid of the vectors in the $B_\delta(x_j)$, the set of δ-near time series of x_j. Note that there are a few other representatives for swapping, e.g., Chebyshev center, weighted mean center (based on the distances from the given point), etc. Swapping of the given point with an artificial data point such as a centroid removes the possibility of poor selection of a node which locates farther than most neighbors. This keeps absolute change to forecasts at an improved bound while protecting the given time series with randomization. In Section 5.2, we present an integer programming model (matching problem) for optimal swapping, where centroids are used in case of odd-numbered time series data in a cluster.

Figure 5 Alternative representatives for the group of neighbors for swapping



Note. $k = 10$ for the k-nearest neighbors of x_3, $K_3 = \{x_1, x_2, x_4, x_6, x_7, x_9, x_{14}, x_{16}, x_{21}, x_{27}\}$ (i) $x'_3 =$ centroid of conv(K_3) (ii) $x''_3 =$ centroid of $K_3 \cup \{x_3\}$ (iii) $x^*_3 =$ centroid of $B_\delta(x_3) \cup \{x_3\}$.

5.2. Minimum weight perfect matching for the k-mTS shuffling

Our goal is to minimize absolute change to forecasts while keeping data privacy issues at a desirable level. In this section, we present optimal shuffling by the use of an integer programming model for a matching problem, where simultaneous data-swapping takes place within clusters. For each cluster $Q_h, h = 1, \ldots, k$ with $|Q_h| = n_h$, we put the points in a matrix as columns, then the matrix

$\mathbf{Q}_h = [x_{1_h}, x_{2_h}, \ldots, x_{n_h}] \in \mathbb{R}^{n \times n_h}$. By Algorithm 3 in Section 3.2, we shuffle time series data within the cluster in order to obtain a protected data matrix X'_h:

$$X'_h = [x'_{1_h}, x'_{2_h}, \ldots, x'_{n_h}] = \begin{pmatrix} A_{1_h,t-n+1} & A_{1_h,t-n+2} & A_{1_h,t-n+3} & \cdots & A_{1_h,t-1} & P_{1_h,t} \\ A_{2_h,t-n+1} & A_{2_h,t-n+2} & A_{2_h,t-n+3} & \cdots & A_{2_h,t-1} & P_{2_h,t} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ A_{n_h,t-n+1} & A_{n_h,t-n+2} & A_{n_h,t-n+3} & \cdots & A_{n_h,t-1} & P_{n_h,t} \end{pmatrix}^T. \tag{86}$$

For a perfect matching problem formulation, we put the column vectors (time series) of the matrix $\mathbf{Q}_h$ on the graph G, where every pair of distinct nodes is connected by a unique edge. We then transform such complete graph into bipartite graph with edge weights in terms of the data intruder's utility, where weights ∞ are assigned to the loops (i.e., $w(\{j,j\}) = \infty$). For the weight on edge $\{j,j'\}$, for $j' \neq j$ we use the following:

$$\text{(Data intruder's utility)} \qquad w_{j,j'} = w(\{j,j'\}) = u(P_{j',i}) = \sqrt{\frac{1}{\hat{f}_{x_j}(P_{j',i})}}, \tag{87}$$

where $\hat{f}_{x_j}$ denotes a kernel density based on the past $n-1$ observations up until time $i-1$ for time series j.

Since the absolute change to forecasts $|F - F^*|$ is based on $|A_t - P_t|$ (as in Theorems 1, 2, 3), we define another edge weight on $\{j,j'\}$, for $j' \neq j$ as follows.

$$\text{(absolute change to forecasts base)} \qquad f_{j,j'} = f(\{j,j'\}) = |A_{j,t} - A_{j',t}|. \tag{88}$$

Let us consider graph G with two partitions $\mathbb{A}$ and $\mathbb{B}$, that is, $G = (V,E)$ and $V = \mathbb{A} \cup \mathbb{B}$ and $E \subseteq \{ab : a \in \mathbb{A}, b \in \mathbb{B}\}$. Then, using (87) and (88), we can write the following min-weight perfect matching problem:

$$\begin{array}{l} \min \sum_{e \in E} (w_e + f_e) x_e \\ \text{subject to} \\ x(\delta(a)) = 1, \text{ for all } a \in \mathbb{A} \\ x_e \in \{0,1\}, \text{ for all } e \in E, \end{array} \tag{89}$$

where $\delta(a) = \{ab : ab \in E, a \in \mathbb{A}, b \notin \mathbb{A}\}$. Equivalently, (89) can be written up as the following.

$$\begin{array}{l} \min \sum_{a,b} (w_{ab} + f_{ab}) x_{ab} \\ \text{subject to} \\ \displaystyle\sum_{a \in \mathbb{A}} x_{ab} = 1 \\ \displaystyle\sum_{b \in \mathbb{B}} x_{ab} = 1 \\ x_{ab} \in \{0,1\}, \end{array} \tag{90}$$

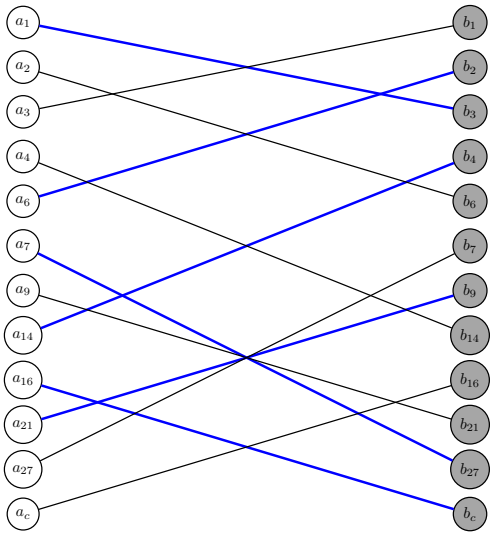
where w_{ab} denotes the data intruder's utility in (87) when time series data of $a \in \mathbb{A}$ at time i is replaced by one of the time series $b \in \mathbb{B}$. Note that $\sum f_{ab} x_{ab}$ denotes the total absolute change to forecasts base from the time series in Q_h.

Using $0 \leq \lambda \leq 1$, we can write its variation as the following.

$$\begin{aligned} &\min \textstyle\sum_{a,b} \lambda w_{ab} x_{ab} + (1-\lambda) f_{ab} x_{ab} \\ &\text{subject to} \\ &\sum_{a \in \mathbb{A}} x_{ab} = 1 \\ &\sum_{b \in \mathbb{B}} x_{ab} = 1 \\ &x_{ab} \in \{0,1\}, \end{aligned} \tag{91}$$

which means minimum intruder utility perfect matching when $\lambda = 1$, and minimum absolute change to forecasts perfect matching when $\lambda = 0$.

Figure 6 Optimal shuffling to simultaneously minimize data intruder's utility and absolute change to forecasts



Note. Taking each cluster as a complete graph, we then transform it into a bipartitie graph for the min-weight perfect matching problem. $G = (V, E)$ and $V = \mathbb{A} \cup \mathbb{B}$ and $E \subseteq \{ab : a \in \mathbb{A}, b \in \mathbb{B}\}$, where each time series j appears in both sets $\mathbb{A}$ and $\mathbb{B}$ as a_j and b_j. The centroid (yellow node) in Figure 7 is depicted as a_c on the left and b_c on the right.

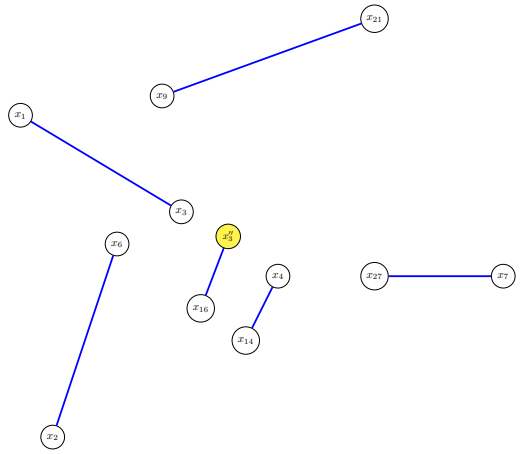
The above problem formulations provide data providers with optimal solution to minimize data intruder's utility. Since perfect matching contains $|V|/2$ edges, there is no perfect matching for a graph with an odd number of vertices. Therefore, for a cluster with an odd number of time series, we include a centroid in the set of nodes for matching problem as described in Figures 6 and 7, which are equivalent. The case in the figures can be written up as the ordered sets in Table 5.

Table 5 Example of Optimal Shuffle by Min-Weight Perfect Matching

Time Series j at time i	$A_{1,i}$	$A_{2,i}$	$A_{3,i}$	$A_{4,i}$	$A_{6,i}$	$A_{7,i}$	$A_{9,i}$	$A_{14,i}$	$A_{16,i}$	$A_{21,i}$	$A_{27,i}$	$A_{c,i}$
For data protection, swapped by	$A_{3,i}$	$A_{6,i}$	$A_{1,i}$	$A_{14,i}$	$A_{2,i}$	$A_{27,i}$	$A_{21,i}$	$A_{4,i}$	$A_{c,i}$	$A_{9,i}$	$A_{7,i}$	$A_{16,i}$

Optimal shuffle (simultaneous swapping for multiple time series data for each cluster) of the example in Figures 6 and 7.

Figure 7 Optimal shuffling as a min-weight perfect matching problem to minimize data intruder's utility



Note. Cluster with an odd number of nodes. $Q = \{x_1, x_2, x_3, x_4, x_6, x_7, x_9, x_{14}, x_{16}, x_{21}, x_{27}\}$. Inclusion of a centroid (yellow node) for perfect matching. Blue edges are the solution of the optimization problem (89), (90) or (91).

In Section 6, we do the k-mTS shuffling by solving (91) with various k and λ's over the dataset of 1,572 time series of JCRs from Ohio in 2001 Q2 to 2010 Q1 ($T = 35$ total quarters) at the US County and NAICS two-digit code aggregation level. Section 6 presents that the numerical solutions of (91) provide us effective data protection while keeping absolute change to forecasts at a desirable level.

6. Experiments

We use an employee-employer dataset from the US Census Bureau's Longitudinal Employer-Household Dynamics (LEHD) program to test our two-party data privacy framework. The dataset contains time series of the number of job creations relating to individual employees and their employers in specific industries and geographies. Identifiable information can result when a quarterly job creation has a count of 1 because it relates to a single individual. Hence, these data require protection because Title 13 of the U.S. Code[1] states that US Census Bureau data released to the public cannot contain "any identifiable information about individuals, households, or businesses."[2] For data protection reasons, these data are usually generalized (e.g., aggregated) to US county and North American Industry Classification System (NAICS) industry classifications, which results in over 50,000 time series nationwide that are available for download (https://lehd.ces.census.gov/data/). Furthermore, job creations are transformed into Job Creation Rates (JCRs) because rates are bounded (between 0 and 2) and normalize the counts across small

[1] https://www.census.gov/content/dam/Census/library/factsheets/2019/comm/2020-confidentiality-factsheet.pdf

[2] We use sanitized job creations in this paper because they have the same underlying structure as the confidential data behind the Census Bureau's secure access servers.

and large geographies. JCRs are computed by dividing each job creation by the county-industry average of employment at the beginning and end of that quarter. A JCR of 0 implies no new jobs and a JCR of 2 implies the creation of a new county-industry.

6.1. Data Protection Process

To illustrate our data protection methods, we use 1,572 time series of JCRs from Ohio in 2001 Q2 to 2010 Q1 ($T = 35$ total quarters) at the US County and NAICS two-digit code aggregation level. Of these, we eliminated 334 time series because disclosure avoidance rules at the US Census Bureau required data suppression in at least one time period. Of the 1,238 remaining time series, we use the data from time $i = 1$ to $i = 25$ to initialize the DPMs. Protected data from each DPM are released on a rolling basis from time $i = 26, \ldots, 35$ by using the actual data up until time $i = 25, \ldots, 34$, respectively. The DPMs include bottom-coding (5, 10, and 20 %), top-coding (5, 10, and 20 %), additive noise (standard deviation parameter equal to 1 and 2 standard deviations of the actual time series data), and k-mTS Shuffling ($k = 10, 15, 20, 25, 30, 35, 40, 45$ and $\lambda = 0.1, 0.3, 0.5, 0.7, 0.9$). We calculate the absolute change to forecasts and privacy at times $i = 26, \ldots, 35$ after the data is protected at time i. Note that the protected data only affects the forecasts at times $i \geq 27$ since the data protection begins at $i = 26$. Thus, in our application, $t + 1 = 27$ and $T = 35$.

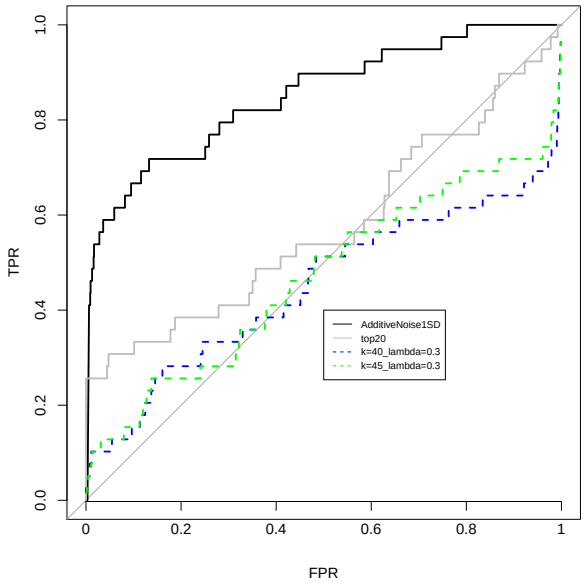
6.2. Data Privacy

Data privacy can be measured irrespective of the forecasts as it only depends on $x_j = (A_{j,i-n+1}, ..., A_{j,t-1}, A_{j,t}, \ldots, A_{j,i})^T$ and $x'_j = (A_{j,i-n+1}, ..., A_{j,t-1}, P_{j,t}, \ldots, P_{j,i})^T$. To define the privacy issues in the actual data per Section 4.1, we set τ^* equal to the 97% quantile of all J values of $u(A_{j,i})$. This results in a binary outcome, $g_{i,j}$, for each time series j at time i. We then calculate $u(P_{j,i})$ per Equation (75) and compare the intruder's utility on each time series j to $g_{i,j}$.

Figure 8 plots the ROC Curves for additive noise with 1 standard deviation (AdditiveNoise1SD), top-coding 20% (top20), and k-mTS Shuffling solution of our optimization model (91) (i.e., simultaneous swapping of multiple data within each cluster) for $k = 40, \lambda = 0.3$ and $k = 45, \lambda = 0.3$. The figure shows that a data intruder can successfully target more privacy issues when the protected data is generated with AdditiveNoise1SD or top-coding 20% compared to k-mTS Shuffling. Also, the data intruder's ability for k-mTS Shuffling is similar to random chance over the entire range of false positive rates.

Table 6 presents the Area Under the ROC Curve (AUC) and the associated 95% confidence intervals(CIs) for the DPMs. The third row shows the maximum likelihood ratio per Equation 81 when the false positive rate (FPR) is greater than 5% (ensures the data intruder will target at least some time series incorrectly). As described previously, a LR_τ greater than 1 implies that the intruder's targeting decision is better than random chance. For all DPMs, this situation occurs

Figure 8 ROC Curve at $i = T$ for DPMs



Note. Standard data protection methods dominate k-mTS Shuffling with lack of privacy.

most commonly at low FPR ranges. However, as the data intruder incurs more false positives, their performance is similar to random guessing for k-mTS Shuffling.

Table 6 Data Privacy at time $i = T$

Data Privacy	AdditiveNoise1SD	Top-coding 20%	$k = 40$, $\lambda = 0.3$	$k = 45$, $\lambda = 0.3$
AUC	0.851	0.575	0.449	0.468
95% CI	(0.779, 0.922)	(0.463, 0.688)	(0.328, 0.570)	(0.353, 0.584)
max LR_τ	11.65	6.08	2.35	2.53
TPR_τ	0.59	0.308	0.128	0.128
FPR_τ	0.051	0.051	0.055	0.051

Decisions are made based on τ with the max LR_τ.

6.3. Absolute Change to Forecasts

To compute the absolute change to forecasts, we first consider using the R package forecast (Hyndman et al. (2008)) for the exponential smoothing models, however, this package re-optimizes the initial level, trends, and seasonalities at every time period. This poses a problem for calculating the absolute change to forecasts because the absolute change to forecasts assumes the same levels, trends, and seasonalities are updated sequentially at each time period per Section 2.1. Hence, we manually code the forecasts using Equations (2) for SES, Equations (15) for DES, and Equations (37) to (40) for TES.

Tables 7, 8, and 9 present the maximum absolute change to forecasts for all exponential smoothing models across varying time periods, broken by the corresponding levels, trends and seasonalities.

The results show that additive noise has the most absolute change to forecasts and bottom-coding has the least. Also, the maximum absolute change to forecasts of top-coding seems comparable to k-mTS Shuffling. With regard to the time series structures, the maximum absolute change to forecasts seems to be driven by the maximum absolute loss in level first, then seasonality, and lastly trend. Our results indicate that the maximum absolute change to forecasts generally increases as more of the data was protected from $i = t+1$ to $i = T$ which corresponds to our findings in Theorem 1. For clarity reasons, we do not present the maximum absolute change to forecasts for DES and TES time periods earlier than $i = T$, but our findings are similar to SES and in accordance with Theorems 2 and 3. Finally, we note that maximum absolute change to forecasts increases as the additive exponential smoothing models have more components (as expected since $A_t - P_t$ shows up in all three Equations (43) to (45) in Subsection 2.1.3).

Table 7 Maximum Absolute Change to Forecasts for SES

Time	None	AdditiveNoise1SD	AdditiveNoise2SD	bottom5	bottom20	top5	top20	$k = 40$	$k = 45$
$i = t+1$	0	0.032	0.070	0.001	0.011	0.011	0.032	0.041	0.018
$i = t+5$	0	0.070	0.128	0.002	0.005	0.044	0.045	0.054	0.046
$i = T$	0	0.133	0.212	0.002	0.010	0.046	0.053	0.067	0.047

Last two columns are based on the k-mTS shuffling solutions of (91) in the cases of $k = 40, \lambda = 0.3$ and $k = 45, \lambda = 0.3$.

Table 8 Maximum Absolute Losses for Forecasts, Levels, and Trends of DES at $i = T$

Max loss	None	AdditiveNoise1SD	AdditiveNoise2SD	bottom5	bottom20	top5	top20	$k = 40$	$k = 45$
$\max\lvert F_T - F_T^*\rvert$	0	0.152	0.262	0.003	0.012	0.053	0.067	0.085	0.056
$\max\lvert l_T - l_T^*\rvert$	0	0.146	0.252	0.002	0.012	0.051	0.064	0.082	0.054
$\max\lvert b_T - b_T^*\rvert$	0	0.007	0.010	0.000	0.000	0.002	0.003	0.003	0.002

Last two columns are based on the k-mTS shuffling solutions of (91) in the cases of $k = 40, \lambda = 0.3$ and $k = 45, \lambda = 0.3$.

Table 9 Maximum Absolute Losses for Forecasts, Levels, Trends, and Seasonalities of TES at $i = T$

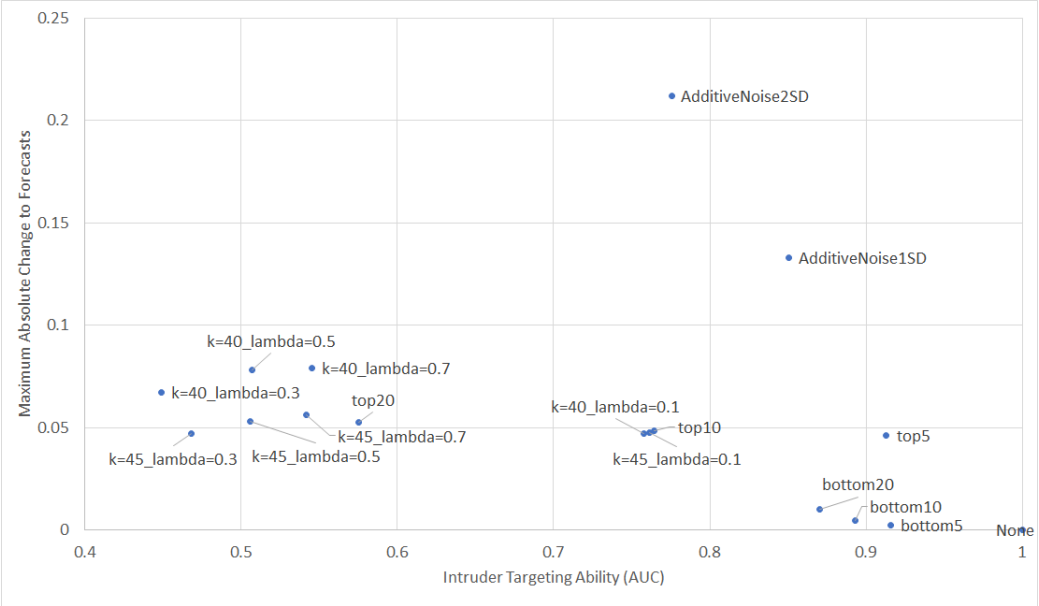
Max loss	None	AdditiveNoise1SD	AdditiveNoise2SD	bottom5	bottom20	top5	top20	$k = 40$	$k = 45$
$\max\lvert F_T - F_T^*\rvert$	0.000	0.173	0.367	0.004	0.017	0.091	0.125	0.109	0.091
$\max\lvert l_T - l_T^*\rvert$	0.000	0.173	0.293	0.003	0.014	0.062	0.074	0.093	0.063
$\max\lvert b_T - b_T^*\rvert$	0.000	0.006	0.010	0.000	0.000	0.002	0.003	0.003	0.002
$\max\lvert s_T - s_T^*\rvert$	0.000	0.031	0.065	0.002	0.004	0.037	0.049	0.058	0.036

Last two columns are based on the k-mTS shuffling solutions of (91) in the cases of $k = 40, \lambda = 0.3$ and $k = 45, \lambda = 0.3$.

6.4. Trade-off Between Data Intruder Utility and Maximum Absolute Change to Forecasts

In this subsection, we compare the trade-off between the utility of a data intruder and the maximum absolute change to forecasts. Figure 9 shows that the k-mTS Shuffling DPMs of (91) for higher values of k (40 to 45) and moderate values of λ (0.3 to 0.7) have a lower maximum absolute change to forecasts per unit of privacy. Importantly, a privacy level (AUC) of 0.5 is equivalent a data intruder randomly guessing which time series have privacy issues. Comparatively, the bottom-coding and additive noise DPMs do not seem to protect the data well. Top-coding protects the data better, but top-coding is not on the efficient frontier on the trade-off. Results were similar for both DES and TES.

Figure 9 Tradeoff at $i = T$ for Simple Exponential Smoothing (SES)



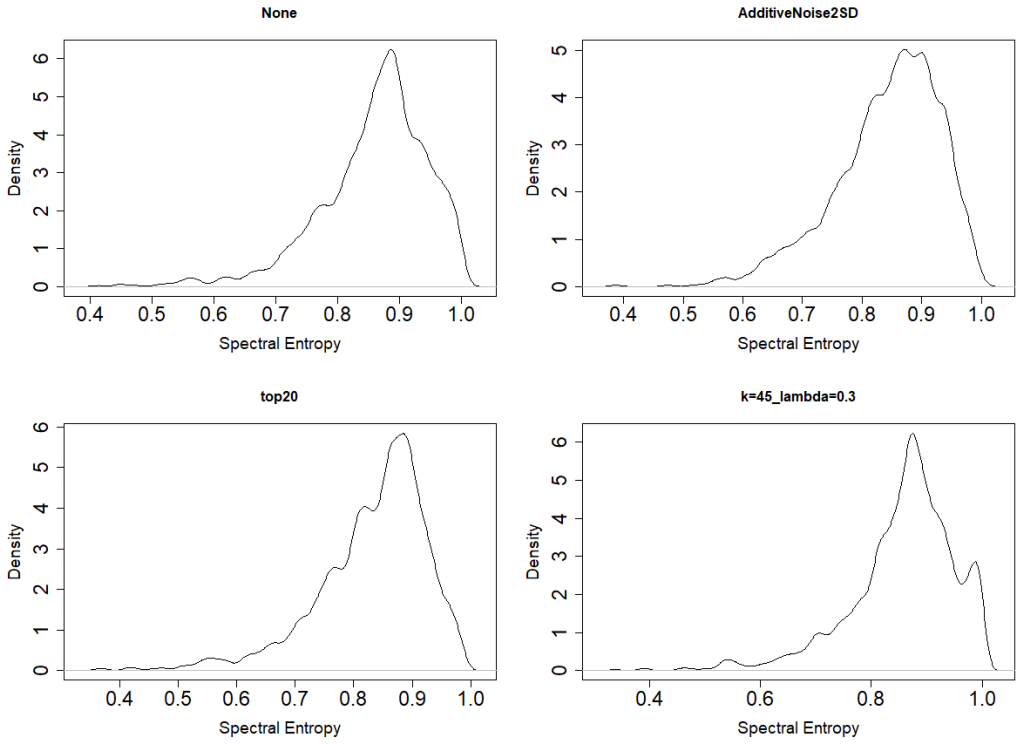
Note. Area Under the Curve (AUC) vs. Maximum Absolute Change to Forecasts for Simple Exponential Smoothing (SES) at time $i = T$.

6.5. Time Series Patterns

Since our theory provides no guarantees for improving forecast accuracy, we do not empirically measure the direction of forecast accuracy. Instead, we compare the patterns from the protected data to the actual data across all J time series. Figure 10 plots the empirical densities of the J spectral entropies of the protected data from time $i = t + 1$ to $i = T$. We can see that the k-mTS Shuffling of (91) with $k = 45$ and $\lambda = 0.3$ produces several protected time series with spectral entropies close to one, indicating that some time series have more disorder and are less forecastable (Kang et al. (2017)) for privacy protection reasons. This illustrates the purposeful trade-off (λ)

that k-mTS Shuffling made between data privacy and absolute change to forecasts to protect the time series with privacy issues.

Figure 10 Spectral Entropy of Protected Data from $i = t+1$ to $i = T$



Note. The standard deviations of the smoothing kernels for density estimation are 0.01 for all DPMs. k-mTS with $k = 45, \lambda = 0.3$ has more protected time series with spectral entropies closer to 1.

6.6. Practical Implications

In our application, we found that standard DPMs mentioned in legislation are not the best choice for minimizing the absolute change to forecasts; instead, organizations should consider using DPMs that explicitly include absolute change to forecasts in its loss function, such as k-mTS Shuffling which chooses the protected data based on the ability to maintain similar forecasts. These findings suggest the importance of explicitly incorporating forecasts (i.e., absolute change to forecasts and $|A_t - P_t|$) into the data protection itself.

Besides absolute change to forecasts, using the additive exponential smoothing models allowed data providers to also see how time series structures were affected by data protection. Thus, our research also has implications for practitioners concerned with how data protection might cause changes in the level, trends, and seasonalities of time series. Based on our limited empirical findings with only 10 time periods for data protection, we found that levels were most affected and then the seasonality coefficients. Trends were not greatly affected at low (but typical) values for α, β,

and γ, but our dataset involved job creation rates that were bounded between 0 and 2. Additive noise altered the time series structures the most while not providing much data privacy. For data protection in more than 10 time periods and more different datasets, we provide the reader with theory in Section 2.

Importantly, our solution considered the operational requirement of starting data protection at time period t with all the prior data being unprotected. We found that the maximal absolute change to forecasts was lower in earlier time periods when the data protection first started. As opposed to standard DPMs like additive noise, the k-mTS method of (91) stabilized the maximal absolute change to forecasts for all forecasting models (SES, DES, and TES) in later time periods when k was higher, indicating that more ($k = 45$) means were better for maintaining a consistent maximal absolute change to forecasts. In addition, the data provider can choose whether or not to release the protected data (and which DPM to use) during time i by simultaneously looking at the absolute change to forecasts, providing management with a ability to override data release decisions.

7. Conclusions

Given that the number of data breaches doubled in the past two years after data protection legislation, privacy protection after data collection is not enough. Zuboff (2021) states that the entire economic system of incentives is centered around the commodification of personal data and more focus needs to be placed on whether data collection addresses the genuine needs of people and communities. Similarly, the Forecasting for Social Good (FSG) framework presented in this paper placed greater emphasis on the non-economic (privacy) needs of society at large, but only tried to fix an existing problem stemming from the big data collection incentives. Further research needs to be done on privacy protection ex-ante data collection (e.g., by using the randomized response models of Warner (1965), local differential privacy techniques, or edge computing without storing user data) or question whether the data needs to be collected at all.

In this paper, the aim of the data privacy framework is to alter time series data for privacy reasons while ensuring that forecasts are similar. We provide theoretical guarantees for the maximum absolute change to forecasts as each additional time period was protected. For the decision-making process, we explicitly incorporate the perspective of the forecaster and data intruder into the protected time series data. We treat each time series as a vector and construct a data matrix, together with a distance matrix based on Euclidean geometry. We then turn it into a bipartite graph, where vertices are a time series vector, and their associated edge-weights are calculated by the data intruders' utility and absolute change to forecasts on every pair of given time series vectors. Using such bipartite graph we solve the minimum-weight perfect matching problem for optimal decision making (i.e., optimal swapping or shuffling).

Our theoretical findings are also generalizable to absolute changes to time series structures (level, trend, and seasonality) and the upper bound of the changes to forecast accuracy. To protect time series data and minimize the absolute change to forecasts when using additive exponential smoothing models, we recommend data providers use our proposed DPM of k-mTS Shuffling over standard DPMs. However, the over reliance on distance can produce unintended effects such as large differences in magnitude overwhelming identical time series structures. Follow-on research could involve improving upon the definition of closeness in the time series space, potentially reflecting the measurable time series features such as trend, seasonality, autocorrelation, and randomness.

One major shortcoming of our work is that we did not focus on whether forecast accuracy improves. Although our theory extends the bound of the absolute change to forecasts to the upper bound of the change in forecast accuracy at time $i+1$, it did not state whether forecast accuracy improved or worsened. Thus, we do not claim that minimizing the absolute change to forecasts leads to better forecasting. Instead, we leave this issue to future research and focus on the theoretical foundations of this new topic. In the future, we plan to expand our research to include other forecasting models and the empirical study of how forecast accuracy changes under different data protection methods.

We also did not apply the framework or theory to multivariate forecasting models such as neural networks (Januschowski et al. (2020); Fry and Brundage (2020)), gradient boosted models (Ke et al. (2017)), vector auto-regression models, or hybrid machine learning-exponential smoothing models (Smyl (2020)). We also plan to study related optimization models using stochastic programming techniques (Prékopa (1995); Birge and Louveaux (1997)), especially from the set theoretical approach using related probability bounds (Boros and Prékopa (1989); Boros et al. (2014)). An open question for future research is whether complex forecasting models with related probability bounds will improve forecast accuracy by weighting patterns of related time series more than the noise introduced in the protected data, not change forecast accuracy, or worsen forecast accuracy by over fitting on the protected data. We leave this area to future research.

References

Archibald BC, Koehler A (2003) Normalization of seasonal factors in winters' methods. *International Journal of Forecasting* 19:143–148.

Article 29 Data Protection Working Party (2014) Opinion 05/2014 on anonymisation techniques URL `https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf`.

Azzalini A (1981) A note on the estimation of a distribution function and quantiles by a kernel method. *Biometrika* 68:326–328.

Baena D, Castro J, Frangioni A (2020) Stabilized benders methods for large-scale combinatorial optimization, with application to data privacy. *Manag. Sci.* 66:3051–3068.

Birge J, Louveaux F (1997) *Introduction to Stochastic Programming* (Springer), URL `http://dx.doi.org//10.1007/978-1-4614-0237-4`.

Boone T, Ganeshan R, Jain A, Sanders NR (2019) Forecasting sales in the supply chain: Consumer analytics in the big data era. *International Journal of Forecasting* 35:170–180.

Boros E, Prékopa A (1989) Closed form two-sided bounds for probabilities that exactly r and at least r out of n events occur. *Math. Oper. Res.* 14:317–342.

Boros E, Scozzari A, Tardella F, Veneziani P (2014) Polynomially computable bounds for the probability of the union of events. *Math. Oper. Res.* 39(4):1311–1329.

California State Legislature (2018) California consumer privacy act of 2018 URL `https://oag.ca.gov/privacy/ccpa`.

Chen Y (2017) A tutorial on kernel density estimation and recent advances. *Biostatistics Epidemiology* 1:161 – 187.

Cheu A, Smith A, Ullman JR (2021) Manipulation attacks in local differential privacy. *J. Priv. Confidentiality* 11.

Corpet F (1988) Multiple sequence alignment with hierarchical clustering. *Nucleic acids research* 16 22:10881–10890.

DHS (2021) "what is human trafficking?". URL `https://www.dhs.gov/blue-campaign/what-human-trafficking`, accessed January 31, 2022.

Economic Research Service USDA (2016) National Household Food Acquisition and Purchase Survey (FoodAPS): Users Guide to Survey Design, Data Collection, and Overview of Datasets.

European Parliament and Council of European Union (2016) Regulation (eu) 2016/679 of the european parliament and of the council URL `https://eur-lex.europa.eu/eli/reg/2016/679/oj`.

Fildes R, Goodwin P, Lawrence M, Nikolopoulos K (2009) Effective forecasting and judgmental adjustments: an empirical evaluation and strategies for improvement in supply-chain planning. *International journal of forecasting* 25(1):3–23.

Fry C, Brundage M (2020) The m4 forecasting competition – a practitioner's view. *International Journal of Forecasting* 36:156–160.

Garfinkel R, Gopal R, Góes P (2002) Privacy protection of binary confidential data against deterministic, stochastic, and insider threat. *Manag. Sci.* 48:749–764.

Gelper S, Fried R, Croux C (2009) Robust forecasting with exponential and holt-winters smoothing. *Journal of Forecasting* 29:285–300.

Gorr WL, Schneider MJ (2013) Large-change forecast accuracy: Reanalysis of m3-competition data using receiver operating characteristic analysis. *International Journal of Forecasting* 29(2):274–281.

Gross CW, Sohl JE (1990) Disaggregation methods to expedite product line forecasting. *Journal of forecasting* 9(3):233–254.

Harvey SJ (2013) Smart meters, smarter regulation: Balancing privacy and innovation in the electric grid. *UCLA L. Rev.* 61:2068.

Hyndman R, Koehler A (2006) Another look at measures of forecast accuracy. *International Journal of Forecasting* 22:679–688.

Hyndman RJ, Khandakar Y, et al. (2008) Automatic time series forecasting: the forecast package for r. *Journal of statistical software* 27(3):1–22.

Januschowski T, Gasthaus J, Wang Y, Salinas D, Flunkert V, Bohlke-Schneider M, Callot L (2020) Criteria for classifying forecasting methods. *International Journal of Forecasting* 36:167–177.

Jones MC, Marron JS, Sheather S (1996) A brief survey of bandwidth selection for density estimation. *Journal of the American Statistical Association* 91:401–407.

Kang Y, Hyndman RJ, Smith-Miles K (2017) Visualising forecasting algorithm performance using time series instance spaces. *International Journal of Forecasting* 33(2):345–358.

Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T (2017) Lightgbm: A highly efficient gradient boosting decision tree. *NIPS*.

Laptev N, Amizadeh S, Flint I (2015) Generic and scalable framework for automated time-series anomaly detection. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* .

Lee J, Kim J, Prékopa A (2017) Extreme value estimation for a function of a random sample using binomial moments scheme and boolean functions of events. *Discret. Appl. Math.* 219:210–218.

Li XB, Sarkar S (2009) Against classification attacks: A decision tree pruning approach to privacy protection in data mining. *Oper. Res.* 57:1496–1509.

Li XB, Sarkar S (2013) Class-restricted clustering and microperturbation for data privacy. *Manag. Sci.* 59(4):796–812.

Liu F, Ting K, Zhou Z (2008) Isolation forest. *2008 Eighth IEEE International Conference on Data Mining* 413–422.

Lloyd ST (1982) Least Square Quantization in PCM. *IEEE Transactions on Information Theory* 28:129–137.

Luo J, Hong T, Fang SC (2018) Benchmarking robustness of load forecasting models under data integrity attacks. *International Journal of Forecasting* 34(1):89–104.

Mahajan M, Nimbhorkar P, Varadarajan K (2009) *The Planar k-Means Problem is NP-Hard* (In: Das S., Uehara R. (eds) WALCOM: Algorithms and Computation. WALCOM 2009. Lecture Notes in Computer Science, vol 5431. Springer, Berlin, Heidelberg.).

McKenzie E (1986) Technical note - renormalization of seasonals in winters' forecasting systems: Is it necessary? *Oper. Res.* 34:174–176.

Menon S, Sarkar S (2007) Minimizing information loss and preserving privacy. *Manag. Sci.* 53:101–116.

Munir M, Siddiqui S, Dengel A, Ahmed S (2019) Deepant: A deep learning approach for unsupervised anomaly detection in time series. *IEEE Access* 7:1991–2005.

Muralidhar K, Sarathy R (2006) Data shuffling - a new masking approach for numerical data. *Manag. Sci.* 52:658–670.

Prékopa A (1995) *Stochastic programming* (Kluwer Academic Publishers).

Rostami-Tabar B, Ali MM, Hong T, Hyndman RJ, Porter MD, Syntetos A (2021) Forecasting for social good. *International Journal of Forecasting* .

Schneider MJ, Abowd J (2015) A new method for protecting interrelated time series with bayesian prior distributions and synthetic data. *Journal of The Royal Statistical Society Series A-statistics in Society* 178:963–975.

Schneider MJ, Jagpal S, Gupta S, Li S, Yu Y (2018) A flexible method for protecting marketing data: An application to point-of-sale data. *Marketing Science* 37(1):153–171.

Smyl S (2020) A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting* 36:75–85.

Sweeney L (2002) k-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl. Based Syst.* 10:557–570.

Teng M (2010) Anomaly detection on time series. *2010 IEEE International Conference on Progress in Informatics and Computing* 1:603–608.

Véliz C (2020) *Privacy is power* (Random House Australia).

Verizon (2019) 2019 data breach investigations report. URL `https://www.verizon.com/business/resources/reports/2019-data-breach-investigations-report.pdfx`, accessed January 31, 2022.

Verizon (2021) 2021 data breach investigations report. URL `https://www.verizon.com/business/resources/reports/2021-data-breach-investigations-report.pdfx`, accessed January 31, 2022.

Warner SL (1965) Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association* 60(309):63–69.

Xu H, Zhang N (2021) Implications of data anonymization on the statistical evidence of disparity. *Manag. Sci.* .

Yao Y, Liu Y, Guan Q, Hong Y, Wang R, Wang R, Liang X (2021) Spatiotemporal distribution of human trafficking in china and predicting the locations of missing persons. *Computers, Environment and Urban Systems* 85:101567.

Zuboff S (2021) The coup we are not talking about. *The New York Times* 29.