Transportation Market Rate Forecast Using Signature Transform

Haotian Gu * Tim Jacobs † Philip Kaminsky † Xin Guo †‡ Xinyu Li ‡§

January 11, 2024

Abstract

Currently, Amazon relies on third parties for transportation marketplace rate forecasts, despite the poor quality and lack of interpretability of these forecasts. While transportation marketplace rates are typically very challenging to forecast accurately, we have developed a novel signature-based statistical technique to address these challenges and built a predictive and adaptive model to forecast marketplace rates. This novel technique is based on two key properties of the signature transform. The first is its universal nonlinearity which linearizes the feature space and hence translates the forecasting problem into a linear regression analysis; the second is the signature kernel which allows for comparing computationally efficiently similarities between time series data. Combined, these properties allow for efficient feature generation and more precise identification of seasonality and regime switching in the forecasting process. Preliminary result by the model shows that this new technique leads to far superior forecast accuracy versus commercially available industry models with better interpretability, even during the period of Covid-19 and with the sudden onset of the Ukraine war.

1 Introduction

Overview. Linehaul transportation costs make up a significant portion of overall Amazon transportation costs. To manage these costs, Amazon has developed a variety of tools to manage linehaul capacity mix and procurement. One key input to all of these models is the forecast of transportation marketplace rates, which however are notoriously difficult to forecast – they are driven a number of factors: the ever-changing network of tens of thousands of drivers, shippers of all sizes with a mix of occasional, seasonal, and regular demand, a huge set of brokers, traditional and digital exchanges, and local, regional, national, and international economic factors of all kinds. In addition, the transportation marketplace frequently goes through fundamental shifts – whether because of wars, pandemics, fuel prices, or due to shifting international trade patterns.

Although Amazon has purchased externally-created forecasts for some time, these forecasts are neither explainable nor sufficient/accurate to the specific Amazon needs. To address this challenge, we built a forecasting model based on time series data to predict weekly marketplace rates for the North America market, at both the national and the regional levels. Our approach incorporates an innovative statistical technique capable of efficiently capturing significant fluctuations in transportation marketplace rates.

*Department of Mathematics, University of California, Berkeley. **Email:** haotian_gu@berkeley.edu

†Middle Mile Marketplace Science, Amazon Research. **Email:** {timojaco,philipka,xnguo}@amazon.com

‡Department of Industrial Engineering & Operations Research, University of California, Berkeley. **Email:** {xinguo, xinyu_li}@berkeley.edu

§Corresponding author

The key challenge in time series forecasting. Time series data consists of sequential observations recorded over time and is ubiquitous: finance, economics, transportation, weather, and energy prices. Given time series data, forecasting additional data points is critical for informed decision-making and process optimization in almost every organization and industry.

Time series prediction models such as Autoregressive Integrated Moving Average (ARIMA) (Shumway et al. 2017) and Exponential Smoothing (Gardner Jr 2006) assume that the time series are stationary, which is not the case in the marketplace rates. Moreover, ARIMA has limited ability to capture seasonality and long-term trends (Kumar and Vanajakshi 2015), and Exponential Smoothing may be insufficient for abrupt changes or outliers and produce unstable forecasts (Taylor 2004). Furthermore, these methods rely solely on historical data of the time series, which is inadequate in capturing the causal relation between the economic factors and the marketplace rates. Meanwhile, machine learning algorithms such as Long Short-Term Memory Neural Networks (Yu et al. 2019) and Gated Recurrent Units (Chung et al. 2014), though capable of capturing nonlinear relationship and complex patterns in time series data, will require substantially more training data which is not available in our case.

Indeed, one of the main challenges in analyzing time series data is their ever-changing statistical properties, due to factors including changes in business and economic cycles, shifts in policy, or changes in market conditions. In our case, the marketplace itself has recently experienced shifts in *regimes* and *seasonality* (Hamilton 1989), in terms of volatility, trends, and cyclical patterns, partly due to the Covid-19 pandemics and the Ukraine-Russian war.

Machine learning models and signature transform. Much of statistical learning theory relies on finding a feature map that embeds the data (for instance, samples of time series) into a high-dimensional feature space. Two requirements for an ideal feature map are *universality*, meaning that non-linear functions of the data are approximated by linear functionals in the feature space; and *characteristicness*, meaning that the expected value of the feature map characterizes the law of the random variable. It is shown that with the technique of the signature transform these two properties are in duality and therefore often equivalent (Simon-Gabriel and Schölkopf 2018). This is the main inspiration of our proposed signature-based forecasting technique for our forecast models.

Originally introduced and studied in algebraic topology (Chen 1954, 1957), the *signature transform*, sometimes referred to as the *path signature* or simply *signature*, has been further developed in rough path theory (Lyons et al. 2007, Friz and Victoir 2010), introduced for financial applications (Lyons et al. 2014, Arribas 2018, Lyons et al. 2019, Kalsi et al. 2020) and machine learning (Yang et al. 2015, Xie et al. 2017, Li et al. 2017, Bonnier et al. 2019, Kidger and Lyons 2020), and most recently to time series data analysis (Morrill et al. 2020). Given any continuous or discrete time series, their signature transform produces a vector of real-valued features that extract information such as order and area, and explicitly considers combinations of different channels. The signature of time series uniquely determines the time series, and does so in a computationally efficient way. Most importantly, every continuous function of a time series data may be asymptotically approximated by a linear functional of its signature. In other words, signature transform linearizes the otherwise complicated feature space, and thus is a powerful tool for feature generation and pattern identification in machine learning.

Our work. We propose a novel signature-based statistical technique for the time series forecasting problem. This is based on two key properties of the signature transform. The first is the universal nonlinearity of the signature transform which linearizes the features space of the time series data and hence translates the forecasting problem into a linear regression analysis; to avoid issues of overfitting and co-linearity and to improve the forecast accuracy, we adopt the two-step LASSO for the regression analysis (Belloni and Chernozhukov 2013). The second is the signature kernel which allows for computationally efficient comparison of similarities between time series data. Technically, this is to identify different "signature feature maps", the statistical counterpart of identifying different distributions for a given time series data, albeit in the linearized features space from the signature transform. Combined, this leads to our signature based two-step LASSO approach with adaptive weights via the signature kernel (Chevyrev and Oberhauser 2022). This signature-transform-based technique (Friz and Hairer 2020) for data analysis allows for efficient feature generation and more precise identification of seasonality and regime switching embedded in the data.

Preliminary analysis shows that our forecast model presents superior performance than commercially available forecast models, while providing significantly better interpretability, despite the onset of Covid-19 and the current Ukraine war.

2 Preliminary on Signature Transform

2.1 Signatures of Continuous Paths

We begin with the definition of signatures of continuous paths.

Notation Let $\mathbb{R}^{d_1}\otimes\mathbb{R}^{d_2}\otimes\cdots\otimes\mathbb{R}^{d_n}$ denote the space of all real tensors with shape $d_1\times d_2\times\cdots\times d_n$. Define a binary operation called *tensor product*, denoted by $\otimes$, which maps a tensor of shape $(d_1,\ldots,d_n)$ and a tensor of shape $(e_1,\ldots,e_m)$ to a tensor of shape $(d_1,\ldots,d_n,e_1,\ldots,e_m)$ via $(A_{i_1,\ldots,i_n},B_{j_1,\ldots,j_m})\mapsto A_{i_1,\ldots,i_n}B_{j_1,\ldots,j_m}$. When applied to two vectors, it reduces to the outer product. Let $\left(\mathbb{R}^d\right)^{\otimes k}=\mathbb{R}^d\otimes\cdots\otimes\mathbb{R}^d$, and $v^{\otimes k}=v\otimes\cdots\otimes v$ for $v\in\mathbb{R}^d$, in each case with $k-1$ many $\otimes$.

Definition 1 *Let $a<b\in\mathbb{R}$, and $X=\left(X^1,\ldots,X^d\right):[a,b]\to\mathbb{R}^d$ be a continuous piecewise smooth path. The signature of X is then defined as the collection of iterated integrals*

$$\begin{aligned}\mathrm{Sig}(X)&=\left(\int_{a<t_1<\cdots<t_k<b}\mathrm{d}X_{t_1}\otimes\cdots\otimes\mathrm{d}X_{t_k}\right)_{k\geq 0}\\&=\left(\left(\int_{a<t_1<\cdots<t_k<b}\mathrm{d}X_{t_1}^{i_1}\cdots\mathrm{d}X_{t_k}^{i_k}\right)_{1\leq i_1,\ldots,i_k\leq d}\right)_{k\geq 0},\end{aligned}\tag{1}$$

where $\otimes$ denotes the tensor product, $\mathrm{d}X_t=\frac{\mathrm{d}X_t}{\mathrm{d}t}\,\mathrm{d}t$, and the $k=0$ term is taken to be $1\in\mathbb{R}$. The truncated signature of depth N of X is defined as

$$\mathrm{Sig}^N(X)=\left(\int_{a<t_1<\cdots<t_k<b}\mathrm{d}X_{t_1}\otimes\cdots\otimes\mathrm{d}X_{t_k}\right)_{0\leq k\leq N}.\tag{2}$$

Remark 1 *The signature may in fact be defined more generally on paths of bounded variation (Friz and Hairer 2020), but the above definition suffices for our purposes.*

Example 1 *Suppose $X:[a,b]\to\mathbb{R}^d$ is the linear interpolation of two points $x,y\in\mathbb{R}^d$, so that $X_t=x+\frac{t-a}{b-a}(y-x)$. Then its signature is the collection of tensor products of its total increment:*

$$\mathrm{Sig}(X)=\left(1,y-x,\frac{1}{2}(y-x)^{\otimes 2},\frac{1}{6}(y-x)^{\otimes 3},\ldots,\frac{1}{k!}(y-x)^{\otimes k},\ldots\right),\tag{3}$$

which is independent of a,b.

Example 2 *Suppose $X : [a, b] \to \mathbb{R}$ is a one-dimensional smooth path. Then its signature is the collection of powers of its total increment:*

$$\mathrm{Sig}(X) = \left(1, X(b) - X(a), \frac{1}{2}\left(X(b) - X(a)\right)^2, \frac{1}{6}\left(X(b) - X(a)\right)^3, \ldots, \frac{1}{k!}\left(X(b) - X(a)\right)^k, \ldots\right), \tag{4}$$

which is independent of $X(t), t \in (a, b)$. Furthermore, when $X(t)$ is a random process, the expected signature

$$\mathbb{E}\left[\mathrm{Sig}(X)\right] = \left(1, \mathbb{E}\left[X(b) - X(a)\right], \frac{1}{2}\mathbb{E}\left[\left(X(b) - X(a)\right)^2\right], \ldots, \frac{1}{k!}\mathbb{E}\left[\left(X(b) - X(a)\right)^k\right], \ldots\right), \tag{5}$$

whenever it exists, describes precisely the moments of $X(b) - X(a)$. Thus, for a high-dimensional stochastic process $X(t)$, the expected signature, naturally form the generalization of the moments of the process. In other words, for a stochastic process $X(t)$, its expected signature characterizes the law of $X(t)$ up to tree-like equivalence, as proved in Chevyrev and Lyons (2016).

Example 2 shows that the signature for a one-dimensional path, only depends on its total increment. In general, it implies that the signature of a path itself may not carry sufficient information to fully characterize the path. Nevertheless, this problem may be resolved by considering the *time-augmented* version of the original path.

Definition 2 *Given a path $X : [a, b] \to \mathbb{R}^d$, define the corresponding time-augmented path by $\widehat{X}_t = (t, X_t)$, which is a path in $\mathbb{R}^{d+1}$.*

Theorem 1 (Uniqueness (Hambly and Lyons 2010)) *Let $X : [a, b] \to \mathbb{R}^d$ be a continuous piecewise smooth path. Then $\mathrm{Sig}(\widehat{X})$ uniquely determines X up to translation.*

In fact, the signature not only determines a path uniquely up to translation, but also *linearizes* any continuous functions of the path, as stated in the next theorem.

Theorem 2 (Universal nonlinearity (Arribas 2018)) *Let F be a real-valued continuous function on continuous piecewise smooth paths in $\mathbb{R}^d$ and let $\mathcal{K}$ be a compact set of such paths. Furthermore assume that $X_0 = 0$ for all $X \in \mathcal{K}$. (To remove the translation invariance.) Let $\varepsilon > 0$. Then there exists a linear functional L such that for all $X \in \mathcal{K}$, $|F(X) - L(\mathrm{Sig}(\widehat{X}))| < \varepsilon$.*

This *universal nonlinearity* is the key property of the signature transform and important for applications in feature augmentations. See Lyons et al. (2014), Li et al. (2017), Morrill et al. (2020) for examples.

Note that the signature by definition is a infinite dimensional tensor. In practice, one can only compute the truncated signature Sig^N in Equation (2) up to some depth N. The next result guarantees that reminder terms in the truncation decay factorially.

Theorem 3 (Factorial decay (Lyons et al. 2007)) *Let $X : [a, b] \to \mathbb{R}^d$ be a continuous piecewise smooth path. Then*

$$\left\| \int_{a<t_1<\cdots<t_k<b} \mathrm{d}X_{t_1} \otimes \cdots \otimes \mathrm{d}X_{t_k} \right\| \leq \frac{C(X)^k}{k!}$$

where $C(X)$ is a constant depending on X and $\|\cdot\|$ is any tensor norm on $\left(\mathbb{R}^d\right)^{\otimes k}$.

The next property about signatures, Theorem 4, is the *invariance to time reparameterisations.* It implies that the signature encodes the data by its arrival order and independently of its arrival time. This is a desired property in many applications such as hand-writing recognition Yang et al. (2015), Xie et al. (2017).

Meanwhile, there is an interesting interplay between Theorem 2 and Theorem 4: In a problem where time parameterisations are irrelevant, it suffices to compute the signature of X by Theorem 4; However, if time parameterisation is important, then according to Theorem 2, applying the signature transform to the time-augmented path $\widehat{X}$ ensures that parameterisation-dependent features are still learned.

Theorem 4 (Invariance to time reparameterisations (Lyons et al. 2007)) *Let $X : [0,1] \to \mathbb{R}^d$ be a continuous piecewise smooth path. Let $\psi : [0,1] \to [0,1]$ be continuously differentiable, increasing, and surjective. Then* $\mathrm{Sig}(X) = \mathrm{Sig}(X \circ \psi)$.

2.2 Signatures of Streams of Data

To define and compute signatures of discrete data streams, one can simply do linear interpolations and then apply signature transforms.

Definition 3 *The space of streams of data is defined as*

$$\mathcal{S}\left(\mathbb{R}^d\right) = \left\{\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) : \boldsymbol{x}_i \in \mathbb{R}^d, n \in \mathbb{N}\right\}.$$

Given $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \mathcal{S}\left(\mathbb{R}^d\right)$, the integer n is called the length of $\boldsymbol{x}$. Furthermore for $a, b \in \mathbb{R}$ such that $a < b$, fix

$$a = u_1 < u_2 < \cdots < u_{n-1} < u_n = b. \tag{6}$$

Let $X = \left(X^1, \ldots, X^d\right) : [a,b] \to \mathbb{R}^d$ be continuous such that $X_{u_i} = \boldsymbol{x}_i$ for all i, and linear on the intervals in between. Then X is called a linear interpolation of $\boldsymbol{x}$.

Definition 4 *Let $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \mathcal{S}\left(\mathbb{R}^d\right)$ be a stream of data. Let X be a linear interpolation of $\boldsymbol{x}$. Then the signature of $\boldsymbol{x}$ is defined as* $\mathrm{Sig}(\boldsymbol{x}) = \mathrm{Sig}(X)$, *and the truncated signature of depth N of $\boldsymbol{x}$ is defined as* $\mathrm{Sig}^N(\boldsymbol{x}) = \mathrm{Sig}^N(X)$.

Note that by Theorem 4, the signature of a stream of data is independent of the choice of u_i in Equation (6) in a linear interpolation. Meanwhile, by Theorem 2, in order to learn parameterisation-dependent features, one can apply the signature transform to the time-augmented data stream $\widehat{\boldsymbol{x}} = (\widehat{\boldsymbol{x}}_1, \ldots, \widehat{\boldsymbol{x}}_n)$, where $\widehat{\boldsymbol{x}}_i = (\boldsymbol{x}_i, t_i) \in \mathbb{R}^{d+1}$, and t_i is the time when the data point $\boldsymbol{x}_i$ arrives.

Let $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \mathcal{S}\left(\mathbb{R}^d\right)$ be a data stream of length n in $\mathbb{R}^d$. Then $\mathrm{Sig}^N(\boldsymbol{x})$ has

$$M(d,N) := \sum_{k=0}^{N} d^k = \frac{d^{N+1}-1}{d-1} \tag{7}$$

components. In particular, the number of components does not depend on the length of data stream n. The truncated signature maps the infinite-dimensional space of streams of data $\mathcal{S}\left(\mathbb{R}^d\right)$ into a finite-dimensional space of dimension $\left(d^{N+1}-1\right)/(d-1)$. Thus the signature is an efficient way to tackle long streams of data, or streams of variable length. Efficient recursive algorithms for signature computation have been developed in the literature Bonnier et al. (2019). See Appendix A for details.

3 Forecast with Signature Transform

In this section, we propose a signature-based statistical technique for the time series forecast problem. This is based on two key properties of the signature transform. The first is the universal nonlinearity of the signature transform which linearizes the feature spaces of the time series data and hence translates the forecasting problem into a linear regression analysis; to avoid issues of overfitting and co-linearity and to improve the forecast accuracy, we adopt the two-step LASSO in the regression analysis. The second is the signature kernel which allows for comparing similarities between time series data. This measure of similarity leads to a novel LASSO-based approach in which weights are adapted according to the similarities between time series data and captures more precisely seasonality and regime switching embedded in the data.

3.1 Time Series Forecast Probelm

The forecast problem involves two time series $\{\boldsymbol{x}_\tau\}_{\tau\in\mathbb{N}}$ and $\{y_\tau\}_{\tau\in\mathbb{N}}$: $\boldsymbol{x}_\tau \in \mathcal{X} \subset \mathbb{R}^d$ is a d_0-dimensional vector consisting of observable factors at time τ, and takes value from an admissible factor space $\mathcal{X}$; meanwhile $y_\tau \in \mathcal{Y} \subset \mathbb{R}$ is the target variable one aims to predict at time τ, taking value from an admissible set $\mathcal{Y}$. The general goal of the forecast problem is to find a model $f^* \in \mathcal{F} \subset \{f | f : \mathcal{X} \to \mathcal{Y}\}$ such that $f^*(x_\tau) \approx y_\tau$, where $\mathcal{F}$ is the class of all admissible models.

Specific to the freight market rate forecast problem (which we currently obtain from a commercial service, DAT), y_τ is the market rate, while $\boldsymbol{x}_\tau$ represents the key economic factors that drive the supply and demand in the freight market: factors from the market supply side include information regarding supply of drivers and trucks and fuel/oil prices, and some market demand factors are imports, agriculture information, manufacturing activities, housing indexes, and railway transport. A comprehensive list of collected factors is given in Appendix B.

Given the data up to time $t-1$: $\{(\boldsymbol{x}_\tau, y_\tau)\}_{\tau\in[t-1]}$, in order to make prediction for y_t from $\boldsymbol{x}_t$, one standard approach to find $f_t^* \in \mathcal{F}$ is by solving the following optimization problem:

$$f_t^* \in \underset{f\in\mathcal{F}}{\arg\min} \left\{ \frac{1}{t-1} \sum_{\tau=1}^{t-1} L\left(f(\boldsymbol{x}_\tau), y_\tau\right) \right\}, \tag{8}$$

where $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is a loss function measuring the difference between the model prediction $f(\boldsymbol{x}_\tau)$ and the actual y_τ. Once f_t^* is obtained, the prediction of y_t is given by $\widehat{y}_t := f_t^*(\boldsymbol{x}_t)$. Then in the next time step $t+1$, the model will be retrained by solving Equation (8) using the data up to time t.

3.2 Signature-based Two-step LASSO

Besides the "external" factors $\boldsymbol{x}_\tau$, most time series forecasting approaches, such as ARIMA, also construct "internal" features from the history of y_τ. Those "internal" features may help to characterize the trend, momentum and stationarity of y_τ. As suggested by the universal nonlinearity (Theorem 2), signature features are suitable candidates for the "internal" features.

More specifically, for any time step $\tau \in \mathbb{N}$ and time window size $l \in \mathbb{N}$, denote $y_{\tau-l:\tau-1} := (y_{\tau-l}, \cdots, y_{\tau-1})$ as the slice of the time series $\{y_t\}_{t\in\mathbb{N}}$ from time $\tau - l$ to $\tau - 1$. The feature vector for predicting y_τ consists of both the economic factors $\boldsymbol{x}_\tau$ and the depth-N signature features $\mathrm{Sig}^N(y_{\tau-l:\tau-1})$. We denote the concatenation of those two sets of features as $\left[\boldsymbol{x}_\tau, \mathrm{Sig}^N(y_{\tau-l:\tau-1})\right]$, whose dimension is denoted by d.

The universal nonlinearity theorem of the signature transform linearizes the feature space, thus one can aim to fit a *linear* model to the features. Since the dimension d of the feature vector may be relatively large compared to the number of historical samples, especially when the time step t is small, solving the problem with ordinary least square can easily cause the problem of over-fitting. To resolve this issue, we adopt the LASSO approach, the key idea of which Tibshirani (1996), Zhao and Yu (2006), Zou (2006) is to add an L_1-regularization to model coefficients in the ordinary least square objective. This L_1-regularization will encourage the sparsity of model coefficients, and prevent the over-fitting problem.

Mathematically, the LASSO regression is to solve the following optimization problem:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda} \in \underset{\boldsymbol{\theta}\in\mathbb{R}^d}{\arg\min} \left\{ \frac{1}{T-1} \sum_{\tau=1}^{t-1} \left(y_\tau - \left[\boldsymbol{x}_\tau, \mathrm{Sig}^N(y_{\tau-l:\tau-1})\right] \cdot \boldsymbol{\theta} \right)^2 + \lambda \|\boldsymbol{\theta}\|_1 \right\}. \tag{9}$$

Here the constant λ, called the regularization parameter, controls the sparsity of coefficients: a higher value of λ leads to a smaller number of nonzero coefficients in $\widehat{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda}$.

The over-fitting problem can be further controlled with the two-step LASSO. The first step is to select the factors by solving the LASSO regression in Equation (9), and get $\widehat{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda}$. In the second step, an OLS with only the selected factors is applied. That is, given the LASSO estimator $\widehat{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda}$, the subsequent OLS refitting is to find $\overline{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda}$ such that:

$$\overline{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda} \in \underset{\mathrm{supp}[\boldsymbol{\theta}]=\mathrm{supp}\left[\widehat{\boldsymbol{\theta}}_{\mathrm{LASSO},t}^{\lambda}\right]}{\arg\min} \left\{ \frac{1}{T-1} \sum_{\tau=1}^{t-1} \left(y_\tau - \left[\boldsymbol{x}_\tau, \mathrm{Sig}^N(y_{\tau-l:\tau-1})\right] \cdot \boldsymbol{\theta} \right)^2 \right\}. \tag{10}$$

This two-step LASSO estimation procedure has been shown to produce a smaller bias than LASSO for a range of models Belloni and Chernozhukov (2013), Chételat et al. (2017).

3.3 Adaptive Two-step LASSO via Signature Kernel

In the approach of two-step LASSO in Equation (10), each historical sample is given equal weight in the optimization problem to obtain the model at time t. However, this equal-weight scheme may fail to account for changes of regime or seasonality. A more suitable scheme would be to put more weight on data from the period which shares more similarity with data from the current period. In the case of forecasting models with signature transform, comparing similarities of data translates to identifying "signature feature maps". This is the statistical equivalence of identifying different distributions for a given set of data, albeit in the linearized features space from the signature transform. To this end, we propose a novel LASSO-based-approach called `AdaWeight`, based on the signature kernel Chevyrev and Oberhauser (2022). It is to adaptively identify seasonality and regime, and assign weights to historical samples accordingly. Details are given as follows.

Signature kernel. First, recall Theorem 2, where the signature feature map

$$\Phi : X \mapsto \mathrm{Sig}(\widehat{X}) \tag{11}$$

is a universal feature map from the path space to the linear space of signatures Chevyrev and Oberhauser (2022). To avoid computation over a large space of functions, let us kernelize the signature feature map Φ in Equation (11), and define the signature kernel $k(x,y) := \langle \Phi(x), \Phi(y) \rangle$, as suggested in Chevyrev and Oberhauser (2022). Here $\langle \cdot, \cdot \rangle$ is the inner product on the linear space of signatures. Now consider two discrete time series $\boldsymbol{x}$ and $\boldsymbol{y}$. In order to measure the similarity between $\boldsymbol{x}$ and $\boldsymbol{y}$, consider the distance induced by the signature kernel (Sriperumbudur et al. 2010, Berlinet and Thomas-Agnan 2011, Gretton et al. 2012, Király and Oberhauser 2019, Chevyrev and Oberhauser 2022),

$$d_{\mathrm{Sig}}(\boldsymbol{x},\boldsymbol{y}) = k\left(\boldsymbol{x},\boldsymbol{x}\right) - 2k(\boldsymbol{x},\boldsymbol{y}) + k\left(\boldsymbol{y},\boldsymbol{y}\right), \tag{12}$$

where $k(\boldsymbol{x},\boldsymbol{y}) = \langle \Phi(\boldsymbol{x}), \Phi(\boldsymbol{y}) \rangle = \langle \mathrm{Sig}(\boldsymbol{x}), \mathrm{Sig}(\boldsymbol{y}) \rangle$ denotes the inner product between $\mathrm{Sig}(\boldsymbol{x})$ and $\mathrm{Sig}(\boldsymbol{y})$ defined in Definition 4. Small $d_{\mathrm{Sig}}(\boldsymbol{x},\boldsymbol{y})$ implies a higher similarity between patterns in $\boldsymbol{x}$ and $\boldsymbol{y}$, which suggests that $\boldsymbol{x}$ and $\boldsymbol{y}$ come from the same regime and share the similar seasonality. In practice, one may truncate the signature to depth N when computing Equation (12), and we denote the distance computed from the depth-N truncated signature by $d_{\mathrm{Sig},N}$.

Adaptive weight via signature kernel. Next, we adapt the weights in the two-step LASSO according to the signature kernel, this is called `AdaWeight.Sig`.

To start, define $\boldsymbol{z}_\tau := (\boldsymbol{x}_\tau, y_\tau)$ for any $\tau \in \mathbb{N}$; for any $\tau_1, \tau_2 \in \mathbb{N}$ and $\tau_1 < \tau_2$, denote $\boldsymbol{z}_{\tau_1:\tau_2} := (\boldsymbol{z}_{\tau_1}, \cdots, \boldsymbol{z}_{\tau_2})$ as a slice of the time series $\{\boldsymbol{z}_t\}_{t\in\mathbb{N}}$ from time τ_1 to τ_2. At each time t, `AdaWeight.Sig` takes the current observed historical samples $\{(\boldsymbol{x}_\tau, y_\tau)\}_{\tau\in[t-1]}$ as input, and outputs an adaptive weight vector

$$\boldsymbol{W}_t := (w_{t,1}, w_{t,2}, \cdots, w_{t,t-1}) \in \mathbb{R}^{t-1}_{\geq 0}, \tag{13}$$

with $\sum_{\tau=1}^{t-1} w_{t,t-1} = 1$. The main idea of `AdaWeight.Sig` is to assign a higher weight $w_{t,\tau}$ to the sample $(\boldsymbol{x}_\tau, y_\tau)$ if the seasonality pattern near time τ is more similar to the seasonality pattern near the current time t. It is important to emphasize that the weight vector $\boldsymbol{W}_t$ is recomputed by the `AdaWeight.Sig` module every time a new sample arrives, allowing it to adapt to changes in the recent data samples.

`AdaWeight.Sig` takes four hyper-parameters: a window size $l \in \mathbb{N}$, a signature depth $N \in \mathbb{N}$, a temperature parameter $\gamma \geq 0$, and the distance metric $d_{\mathrm{Sig},N}$ defined in Equation (12). The kernel-based distance metric $d_{\mathrm{Sig},N}$ measures the similarity between two slices of the (d_0+1)-dimensional time series $\{\boldsymbol{z}_t\}_{t\in\mathbb{N}}$.

Algorithm 1 Adaptive Weight via Signature Kernel (`AdaWeight.Sig`)

1: **Input**: window size l, signature depth N, temperature parameter γ, kernel-based distance metric $d_{\mathrm{Sig},N}$ (12), data set $D = \{(\boldsymbol{x}_\tau, y_\tau)\}_{\tau\in[t]}$.
2: **for** $\tau = 1, 2, \cdots, t$ **do**
3: Compute the distance δ_τ between the truncated depth-N signatures of $\boldsymbol{z}_{t-l:t}$ and $\boldsymbol{z}_{\tau-l:\tau}$:

$$\delta_\tau := d_{\mathrm{Sig},N}\left(\boldsymbol{z}_{\tau-l:\tau}, \boldsymbol{z}_{t-l:t}\right). \tag{14}$$

4: **end for**
5: **for** $\tau = 1, 2, \cdots, t$ **do**
6: Compute the weight w_τ according to

$$w_\tau := \frac{\exp(-\gamma \cdot \delta_\tau)}{\sum_{\tau=1}^{t} \exp(-\gamma \cdot \delta_\tau)}. \tag{15}$$

7: **end for**
8: **Output**: the weight vector $(w_1, \cdots, w_t)$.

Finally, we integrate `AdaWeight.Sig` outlined in Algorithm 1 with the LASSO approach in Equation (10). The full algorithm is summarized in Algorithm 2.

4 Numerical Experiments

The key numerical findings presented below indicate that this new signature-based forecasting approach achieves high predictive accuracy, particularly when compared to industry benchmarks. Effectiveness of the signature kernel is also verified through the experiment.

Algorithm 2 **Adaptive LASSO with Signature Features and Signature Kernel**

1: **Input**: regularization parameter λ, window size l, signature depth N, temperature parameter γ, kernel-based distance metric $d_{\mathrm{Sig},N}$ in Equation (12).

2: **Initialize**: initial data set $D = \{(\boldsymbol{x}_\tau, y_\tau)\}_{\tau\in[t_0]}$

3: **for** $t = t_0+1, t_0+2, \cdots$ **do**

4: **Regime identification**: Run `AdaWeight.Sig` (Algorithm 1) on the data set D, with window size l, signature depth N, temperature parameter γ, and kernel-based distance metric $d_{\mathrm{Sig},N}$ in Equation (12). Output the adaptive weight $\boldsymbol{W}_t := (w_{t,1}, w_{t,2}, \cdots, w_{t,t-1}) = \mathtt{AdaWeight.Sig}(D; l, N, \gamma, d_{\mathrm{Sig},N})$.

5: **2-step LASSO**: Apply the 2-step LASSO method on the current data set D, with adaptive weight $\boldsymbol{W}_t$ and regularization parameter λ:

$$\widehat{\boldsymbol{\theta}}^{\lambda}_{\mathrm{LASSO},t} \in \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^d} \left\{ \sum_{\tau=1}^{t-1} w_{t,\tau}\cdot\left(y_\tau - \left[\boldsymbol{x}_\tau, \mathrm{Sig}^N(y_{\tau-l:\tau-1})\right]\cdot\boldsymbol{\theta}\right)^2 + \lambda\|\boldsymbol{\theta}\|_1 \right\}. \tag{16}$$

$$\overline{\boldsymbol{\theta}}^{\lambda}_{\mathrm{LASSO},t} \in \mathop{\arg\min}_{\mathrm{supp}[\boldsymbol{\theta}]=\mathrm{supp}\left[\widehat{\boldsymbol{\theta}}^{\lambda}_{\mathrm{LASSO},t}\right]} \left\{ \sum_{\tau=1}^{t-1} w_{t,\tau}\cdot\left(y_\tau - \left[\boldsymbol{x}_\tau, \mathrm{Sig}^N(y_{\tau-l:\tau-1})\right]\cdot\boldsymbol{\theta}\right)^2 \right\}. \tag{17}$$

6: Given $\boldsymbol{x}_t$, make prediction on y_t:

$$\widehat{y}_t = \left[\boldsymbol{x}_t, \mathrm{Sig}^N(y_{t-l:t-1})\right]\cdot\overline{\boldsymbol{\theta}}^{\lambda}_{\mathrm{LASSO},t}.$$

7: Observe the actual y_t, and update the data set D: $D = D\cup\{(\boldsymbol{x}_t, y_t)\}$.

8: **end for**

Model relative test errors. For short-term predictions ($\leq$ 8 weeks), the relative test errors for most regions are less than 5%, while the errors are less than 10% for long-term predictions ($>$ 8 weeks). The detailed summary can be found in Table 3.

Comparison between industry models and our model predictions. We then compare the performance of our national level predictions with the standard industry predictions for April 2021 - November 2021 time period. Both our model and industry predictions are made three-month (twelve-week) ahead of the time, and our monthly predictions are obtained by aggregating weekly predictions. The detailed results are listed in Table 1, demonstrating that the our predictions (with relative error around 2%) are superior to standard industry predictions (with relative error around 20%).

Necessity of adaptive signature kernel. To demonstrate the necessity of adaptive signature kernel, we compare the predictions form Algorithm 2 with the predictions without using the signature kernel in Equation (10). The results can be found in Table 4. These predictions are made for Los Angeles on October 24, 2021. Evidently from Table 4, the errors without the signature kernel are larger ($>$ 8%) even for short-term predictions. In contrast, the signature kernel method better captures the seasonality, and obtains smaller relative error ($<$ 5%) for short-term predictions, further illustrating the effectiveness of incorporating the signature kernel in the forecast model.

Our model real-time performance. As a case study, we examine the real-time performance of our model predictions made on June 25, 2022. Our model predictions are compared with actual rates across multiple locations. As Table 1 below shows, our model predictions obtain errors less than 5%. (More examples for

Table 1: Comparing national-level 3-month-ahead industry predictions with our model predictions

Month	Actual rates	Industry prediction 3-month-ahead	% Error	Our model prediction 3-month-ahead	% Error
Apr'21	\$2.44	\$1.89	23%	\$2.39	2%
May'21	\$2.51	\$1.82	27%	\$2.45	2%
Jun'21	\$2.53	N/A	N/A	\$2.48	2%
Jul'21	\$2.57	\$2.18	15%	\$2.53	2%
Aug'21	\$2.61	\$2.21	15%	\$2.58	1%
Sep'21	\$2.71	\$2.23	18%	\$2.70	1%
Oct'21	\$2.72	\$2.36	13%	\$2.69	1%
Nov'21	\$2.72	\$2.38	13%	\$2.71	1%

earlier predictions show similar results, including the one made on October 24, 2021 and the one made on June 25, 2022, presented in Table 5 and Table 2, with prediction errors consistently less than 5%).

Table 2: Our model predictions posted on June 25 2022

Region	Week	Prediction	Actual	% Error
N.A.	**7/3/2022**	1.93	1.95	-1.34%
	7/10/2022	1.93	1.95	-1.12%
	7/17/2022	1.95	1.94	0.37%
	7/24/2022	1.96	1.93	1.25%
	7/31/2022	1.95	1.94	0.90%
A	**7/3/2022**	1.78	1.80	-0.68%
	7/10/2022	1.76	1.80	-2.06%
	7/17/2022	1.76	1.76	0.04%
	7/24/2022	1.74	1.72	1.46%
	7/31/2022	1.73	1.66	4.03%
B	**7/3/2022**	1.26	1.26	0.39%
	7/10/2022	1.27	1.21	4.68%
	7/17/2022	1.25	1.21	3.44%
	7/24/2022	1.25	1.21	3.28%
	7/31/2022	1.25	1.24	1.25%

Region	Week	Prediction	Actual	% Error
C	**7/3/2022**	1.10	1.10	0.40%
	7/10/2022	1.09	1.08	0.70%
	7/17/2022	1.08	1.02	5.32%
	7/24/2022	1.04	1.02	2.01%
	7/31/2022	1.01	1.02	-0.60%
D	**7/3/2022**	1.95	1.92	1.86%
	7/10/2022	1.96	1.94	0.99%
	7/17/2022	1.98	1.98	-0.28%
	7/24/2022	2.03	2.01	1.06%
	7/31/2022	2.07	2.01	2.86%
E	**7/3/2022**	1.76	1.79	-1.70%
	7/10/2022	1.75	1.76	-0.75%
	7/17/2022	1.74	1.76	-0.94%
	7/24/2022	1.73	1.76	-1.89%
	7/31/2022	1.73	1.75	-1.01%

Table 3: Relative test error for model predictions across different regions.

	N.A.	**A**	**B**	**C**	**D**	**E**	**F**
1-week ahead	1.06%	1.88%	1.53%	1.38%	2.25%	1.26%	1.53%
2-week ahead	1.53%	1.93%	2.76%	1.92%	3.27%	2.08%	2.06%
3-week ahead	1.55%	2.95%	2.18%	1.19%	3.55%	2.21%	2.07%
4-week ahead	1.61%	3.83%	1.87%	1.50%	4.97%	2.67%	2.34%
5-week ahead	1.33%	2.57%	2.73%	1.05%	4.47%	2.99%	1.32%
6-week ahead	1.44%	2.86%	2.96%	1.85%	1.59%	2.78%	1.20%
7-week ahead	1.64%	2.33%	5.69%	1.73%	2.63%	2.77%	2.51%
8-week ahead	1.86%	1.71%	5.25%	3.64%	8.35%	2.55%	2.64%
9-week ahead	1.88%	2.99%	4.86%	2.55%	8.55%	3.69%	2.15%
10-week ahead	2.30%	3.80%	3.31%	4.10%	8.58%	2.13%	2.53%
11-week ahead	2.60%	4.71%	5.54%	4.33%	9.66%	2.06%	2.85%
12-week ahead	2.38%	5.65%	5.25%	4.76%	9.68%	2.74%	5.33%
13-week ahead	2.58%	6.14%	6.74%	4.15%	9.20%	2.40%	6.86%
14-week ahead	2.84%	6.79%	3.68%	3.31%	8.16%	4.73%	7.72%
15-week ahead	3.14%	6.87%	6.28%	3.50%	8.66%	3.21%	7.97%
16-week ahead	2.83%	7.63%	6.73%	4.01%	8.57%	3.06%	8.14%

References

Imanol Perez Arribas. Derivatives pricing using signature payoffs. arXiv preprint arXiv:1809.09466, 2018.

Alexandre Belloni and Victor Chernozhukov. Least squares after model selection in high-dimensional sparse models. Bernoulli, 19(2):521–547, 2013.

Alain Berlinet and Christine Thomas-Agnan. Reproducing Kernel Hilbert Spaces in Probability and Statistics. Springer Science & Business Media, 2011.

Patric Bonnier, Patrick Kidger, Imanol Perez Arribas, Cristopher Salvi, and Terry Lyons. Deep signature transforms. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, pages 3105–3115, 2019.

Kuo-Tsai Chen. Iterated integrals and exponential homomorphism. Proceedings of the London Mathematical Society, s3-4(1): 502–512, 1954.

Kuo-Tsai Chen. Integration of paths, geometric invariants and a generalized baker-hausdorff formula. Annals of Mathematics, 65(1):163–178, 1957.

Didier Chételat, Johannes Lederer, and Joseph Salmon. Optimal two-step prediction in regression. Electronic Journal of Statistics, 11(1):2519–2546, 2017.

Ilya Chevyrev and Terry Lyons. Characteristic functions of measures on geometric rough paths. Annals of Probability, 44(6): 4049–4082, 2016.

Ilya Chevyrev and Harald Oberhauser. Signature moments to characterize laws of stochastic processes. Journal of Machine Learning Research, 23(176):1–42, 2022.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.

Table 4: Comparison between predictions with different seasonality adjustments.

Region	Week	Actual	Without seasonality	% Error	Signature seasonality	% Error
G	**10/31/21**	3.37	3.30	-2.31%	3.32	-1.53%
	11/7/21	3.45	3.27	-5.19%	3.34	-3.22%
	11/14/21	3.46	3.25	-5.88%	3.38	-2.15%
	11/21/21	3.52	3.22	-8.65%	3.37	-4.25%
	11/28/21	3.48	3.16	-9.28%	3.35	-3.66%
	12/5/21	3.49	3.15	-9.85%	3.31	-5.15%

Peter K Friz and Martin Hairer. A course on rough paths. Springer, 2020.

Peter K Friz and Nicolas B Victoir. Multidimensional stochastic processes as rough paths: theory and applications, volume 120. Cambridge University Press, 2010.

Everette S Gardner Jr. Exponential smoothing: The state of the art—part ii. International journal of forecasting, 22(4): 637–666, 2006.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. Journal of Machine Learning Research, 13(1):723–773, 2012.

Ben Hambly and Terry Lyons. Uniqueness for the signature of a path of bounded variation and the reduced path group. Annals of Mathematics, pages 109–167, 2010.

James D Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. Econometrica: Journal of the econometric society, pages 357–384, 1989.

Jasdeep Kalsi, Terry Lyons, and Imanol Perez Arribas. Optimal execution with rough path signatures. SIAM Journal on Financial Mathematics, 11(2):470–493, 2020.

Patrick Kidger and Terry Lyons. Signatory: differentiable computations of the signature and logsignature transforms, on both cpu and gpu. In International Conference on Learning Representations, 2020.

Franz J Király and Harald Oberhauser. Kernels for sequentially ordered data. Journal of Machine Learning Research, 20(31): 1–45, 2019.

S Vasantha Kumar and Lelitha Vanajakshi. Short-term traffic flow prediction using seasonal arima model with limited input data. European Transport Research Review, 7(3):1–9, 2015.

Chenyang Li, Xin Zhang, and Lianwen Jin. Lpsnet: a novel log path signature feature based hand gesture recognition framework. In Proceedings of the IEEE International Conference on Computer Vision Workshops, pages 631–639, 2017.

Terry Lyons, Hao Ni, and Harald Oberhauser. A feature set for streams and an application to high-frequency financial tick data. In Proceedings of the 2014 International Conference on Big Data Science and Computing, pages 1–8, 2014.

Terry Lyons, Sina Nejad, and Imanol Perez Arribas. Numerical method for model-free pricing of exotic derivatives in discrete time using rough path signatures. Applied Mathematical Finance, 26(6):583–597, 2019.

Terry J Lyons, Michael Caruana, and Thierry Lévy. Differential equations driven by rough paths. Springer, 2007.

James Morrill, Adeline Fermanian, Patrick Kidger, and Terry Lyons. A generalised signature method for multivariate time series feature extraction. arXiv preprint arXiv:2006.00873, 2020.

Robert H Shumway, David S Stoffer, Robert H Shumway, and David S Stoffer. Arima models. Time series analysis and its applications: with R examples, pages 75–163, 2017.

Carl-Johann Simon-Gabriel and Bernhard Schölkopf. Kernel distribution embeddings: Universal kernels, characteristic kernels and kernel metrics on distributions. Journal of Machine Learning Research, 19(1):1708–1736, 2018.

Bharath K Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert RG Lanckriet. Hilbert space embeddings and metrics on probability measures. The Journal of Machine Learning Research, 11:1517–1561, 2010.

James W Taylor. Smooth transition exponential smoothing. Journal of Forecasting, 23(6):385–404, 2004.

Robert Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1):267–288, 1996.

Zecheng Xie, Zenghui Sun, Lianwen Jin, Hao Ni, and Terry Lyons. Learning spatial-semantic context with fully convolutional recurrent network for online handwritten chinese text recognition. IEEE transactions on pattern analysis and machine

Table 5: Our predictions posted on October 24, 2021.

Region	Week	Prediction	Actual	% Error
N.A.	**10/31/21**	2.69	2.70	-0.49%
	11/7/21	2.69	2.69	0.30%
	11/14/21	2.70	2.69	0.43%
	11/21/21	2.73	2.73	-0.06%
	11/28/21	2.76	2.78	-0.54%
	12/5/21	2.76	2.77	-0.29%
A	**10/31/21**	1.18	1.16	2.23%
	11/7/21	1.19	1.16	2.48%
	11/14/21	1.20	1.19	0.76%
	11/21/21	1.24	1.25	-0.11%
	11/28/21	1.25	1.18	5.55%
	12/5/21	1.26	1.19	5.65%
B	**10/31/21**	2.00	1.97	1.40%
	11/7/21	2.00	1.94	2.97%
	11/14/21	1.99	1.91	4.31%
	11/21/21	2.00	1.93	3.62%
	11/28/21	2.06	2.03	1.43%
	12/5/21	2.13	2.03	5.03%
C	**10/31/21**	3.08	3.16	-2.50%
	11/7/21	3.10	3.10	-0.06%
	11/14/21	3.14	3.08	1.97%
	11/21/21	3.16	3.16	0.08%
	11/28/21	3.19	3.20	-0.21%
	12/5/21	3.24	3.16	2.46%

Region	Week	Prediction	Actual	% Error
G	**10/31/21**	3.32	3.37	-1.53%
	11/7/21	3.34	3.45	-3.22%
	11/14/21	3.38	3.46	-2.15%
	11/21/21	3.37	3.52	-4.25%
	11/28/21	3.35	3.48	-3.66%
	12/5/21	3.31	3.49	-5.15%
D	**10/31/21**	2.73	2.80	-2.37%
	11/7/21	2.74	2.79	-1.82%
	11/14/21	2.79	2.81	-0.72%
	11/21/21	2.81	2.83	-0.70%
	11/28/21	2.89	2.84	1.65%
	12/5/21	2.92	2.89	1.07%
E	**10/31/21**	2.54	2.55	-0.52%
	11/7/21	2.52	2.49	1.24%
	11/14/21	2.52	2.43	3.83%
	11/21/21	2.60	2.45	6.26%
	11/28/21	2.60	2.51	3.82%
	12/5/21	2.61	2.51	4.29%
F	**10/31/21**	2.60	2.62	-0.60%
	11/7/21	2.61	2.57	1.49%
	11/14/21	2.59	2.59	0.05%
	11/21/21	2.62	2.59	1.11%
	11/28/21	2.61	2.62	-0.30%
	12/5/21	2.61	2.63	-0.69%

intelligence, 40(8):1903–1917, 2017.

Weixin Yang, Lianwen Jin, and Manfei Liu. Chinese character-level writer identification using path signature feature, dropstroke and deep cnn. In 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pages 546–550. IEEE, 2015.

Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. Neural computation, 31(7):1235–1270, 2019.

Peng Zhao and Bin Yu. On model selection consistency of lasso. The Journal of Machine Learning Research, 7:2541–2563, 2006.

Hui Zou. The adaptive lasso and its oracle properties. Journal of the American statistical association, 101(476):1418–1429, 2006.

A Computation of Signature Transform

Signature transform of a data stream can be computed in an efficient and tractable way, with the help of Chen's identity (Lyons et al. 2007). We start by introducing the following $\boxtimes$ operation:

With $A_0 = B_0 = 1$, define $\otimes$ by

$$\boxtimes : \left(\prod_{k=1}^{N} \left(\mathbb{R}^d\right)^{\otimes k}\right) \times \left(\prod_{k=1}^{N} \left(\mathbb{R}^d\right)^{\otimes k}\right) \to \prod_{k=1}^{N} \left(\mathbb{R}^d\right)^{\otimes k},$$
$$(A_1, \ldots A_N) \boxtimes (B_1, \ldots, B_N) \mapsto \left(\sum_{j=0}^{k} A_j \otimes B_{k-j}\right)_{1 \leq k \leq N}. \tag{18}$$

Chen's identity (Friz and Victoir 2010) states that the image of the signature transform forms a group structure with respect to $\boxtimes$. That is, given a sequence of data $(x_1, \ldots, x_L) \in \mathcal{S}\left(\mathbb{R}^d\right)$ and some $j \in \{2, \ldots, L-1\}$,

$$\mathrm{Sig}^N\left((x_1, \ldots, x_L)\right) = \mathrm{Sig}^N\left((x_1, \ldots, x_j)\right) \boxtimes \mathrm{Sig}^N\left((x_j, \ldots, x_L)\right).$$

Furthermore, from Example 1, the signature of a sequence of length two can be computed explicitly from the definition. Letting

$$\exp : \mathbb{R}^d \to \prod_{k=1}^{N} \left(\mathbb{R}^d\right)^{\otimes k}, \quad \exp : v \to \left(v, \frac{v^{\otimes 2}}{2!}, \frac{v^{\otimes 3}}{3!}, \ldots, \frac{v^{\otimes N}}{N!}\right), \tag{19}$$

then

$$\mathrm{Sig}^N\left((x_1, x_2)\right) = \exp\left(x_2 - x_1\right)$$

Chen's identity further implies that the signature transform can be computed by

$$\mathrm{Sig}^N\left((x_1, \ldots, x_L)\right) = \exp\left(x_2 - x_1\right) \boxtimes \exp\left(x_3 - x_2\right) \boxtimes \cdots \boxtimes \exp\left(x_L - x_{L-1}\right). \tag{20}$$

Equation (20) implies that computing the signature of an incoming stream of data is efficient and scalable. Indeed, suppose one has obtained a stream of data and computed its signature. Then after the arrival of some more data, in order to compute the signature of the entire signal, one only needs to compute the signature of the new piece of information, and tensor product it with the previously-computed signature.

Improving computational efficiency. Recall from Equation (20) that the signature may be computed by evaluating several $\boxtimes$ in Equation (18) and exp in Equation (19). We begin by noticing that the key component in the computation is to evaluate

$$\left(\prod_{k=1}^{N} \left(\mathbb{R}^d\right)^{\otimes k}\right) \times \mathbb{R}^d \to \prod_{k=1}^{N} \left(\mathbb{R}^d\right)^{\otimes k}, \quad A, z \mapsto A \boxtimes \exp(z).$$

Instead of computing $A \boxtimes \exp(z)$ conventionally through the composition of exp and $\boxtimes$, Kidger and Lyons (2020) suggests to speed up the computation by Horner's method. More specifically, it is to expand

$$A \otimes \exp(z) = \left(\sum_{i=0}^{k} A_i \otimes \frac{z^{\otimes(k-i)}}{(k-i)!}\right)_{1 \leq k \leq N},$$

so that the k-th term can be computed by

$$\sum_{i=0}^{k} A_i \otimes \frac{z^{\otimes(k-i)}}{(k-i)!} = \left(\left(\cdots\left(\left(\frac{z}{k} + A_1\right) \otimes \frac{z}{k-1} + A_2\right) \otimes \frac{z}{k-2} + \cdots\right) \otimes \frac{z}{2} + A_{k-1}\right) \otimes z + A_k.$$

As proved in Kidger and Lyons (2020), this method has uniformly (over d, N) fewer scalar multiplications than the conventional approach, and in fact reduces the asymptotic complexity of this operation from $\mathcal{O}\left(Nd^N\right)$ to $\mathcal{O}\left(d^N\right)$. Furthermore, this rate is asymptotically optimal, since the size of result (an element of $\prod_{k=1}^{N}\left(\mathbb{R}^d\right)^{\otimes k}$), is itself of size $\mathcal{O}\left(d^N\right)$.

B Market Supply and Demand Factors

Our analysis and experiment utilizes over a hundred of national and regional market supply and demand factors, downloaded from the governmental public websites (Federal Reserve Bank, Bureau of Labor Statistics, etc.), as well as industrial databases (DAT, Logistic Manager, etc). The time range for the data is from 2018 to 2022. The typical factors selected by LASSO include consumer price index, industry production index, loaded TEUs, housing index, oil and gas drilling, logistic managers' index, employment information, weather, and other market benchmarks.

C Verification Letter from Amazon

My name is Maneesh Jyoti and I am the Director of Relay Products and Technology with Amazon's Transportation Services organization. I am writing to verify that the paper, entitled "Transportation Market Rate Forecast Using Signature Transform" and submitted to the INFORMS Journal on Applied Analytics for review, represents work done at Amazon and used for our trucking operations.

This work started during the summer of 2021 by the middle mile Product Research and Optimization Science (mmPROS) and Relay Products and Technology teams and focused on estimating the cost of surface transportation up to 16 weeks in the future. This new approach uses a signature transform methodology to capture the influence of business cycles and heterogeneity of the marketplace. This method improved prediction accuracy by fivefold and has an estimated annualized saving of approximately $50MM.