22 September 2023

Temporal and spatial analysis of mobile app data

Orest Bucicovschi[1], *David A. Meyer*[1,2], *David P. Rideout*[1], *Asif Shakeel*[1], *Jiajie Shi*[1]
[1]Department of Mathematics
University of California, San Diego
[2]Theoretical Sciences Visiting Program
Okinawa Institute of Science and Technology Graduate University
orest@gmx.net, dmeyer@ucsd.edu, dp.rideout@pm.me
asif.shakeel@gmail.com, jis254@ucsd.edu

Overview

The NetMob23 Challenge provided data on (interpolated, relative) usage for 68 mobile services during each 15 minute interval within each of a number of $100m \times 100m$ *tiles* covering the 20 major metropolitan regions in France for the 77 days 16 March 2019 to 31 May 2019 [1]. This immense (more than 4×10^{11} data points) dataset contains a wealth of information; in this report we describe methods for extracting temporal and spatial patterns from these data and apply them to problems of anomaly detection and analysis, to population dynamics, and to generation of synthetic data.

State space models for temporal analysis

We begin with a brief summary of linear Gaussian state space models and their estimation [2]. Suppose we have a timeseries $y : [T] \to \mathbb{R}^k$ for $0 < T, k \in \mathbb{N}$, and we believe this timeseries to be dependent on an unobserved variable with its own timeseries $\alpha : [T] \to \mathbb{R}^n$ for $0 < n \in \mathbb{N}$, where the relation between the *observations*, y, and the *states*, α, is linear:

$$y_t = Z\alpha_t + \epsilon_t. \tag{1}$$

(so $Z \in \mathbb{R}^{k\times n}$), where the noise is Gaussian, $\epsilon_t \sim \mathcal{N}\big(0, H_t\big)$, with $0 \leq H_t \in \mathrm{Sym}(\mathbb{R}^{k\times k})$, the covariance matrix.

Suppose further that the time evolution of the state is also linear, with Gaussian noise:

$$\alpha_{t+1} = T\alpha_t + R\eta_t \tag{2}$$

(so $T \in \mathbb{R}^{n\times n}$), where $\eta_t \sim \mathcal{N}\big(0, Q_t\big)$, with $0 \leq Q_t \in \mathrm{Sym}(\mathbb{R}^{m\times m})$ for $n \geq m \in \mathbb{N}$, so $R \in \mathbb{R}^{n\times m}$.

Finally, suppose that the system is initialized with $\alpha_1 \sim \mathcal{N}\big(a_1, P_1\big)$, with $a_1 \in \mathbb{R}^n$ and $0 \leq P_1 \in \mathrm{Sym}(\mathbb{R}^{n\times n})$.

Notice that the model defined by (1) and (2), and this initial condition, specifies that (y, α) has a multivariate Gaussian distribution. Let $Y_t = (y_1, \ldots, y_t)$. The goal of the *Kalman filter* algorithm [3] is to compute the (necessarily Gaussian) distribution of y_{t+1}

conditional on Y_t. To this end we consider the (also necessarily Gaussian) distributions of α_t and α_{t+1} conditional on Y_t and define

$$\begin{array}{ll} a_{t|t} = \mathsf{E}[\alpha_t|Y_t] & \\ P_{t|t} = \mathsf{Var}[\alpha_t|Y_t] & \text{and} \end{array} \quad \begin{array}{l} a_{t+1} = \mathsf{E}[\alpha_{t+1}|Y_t] \\ P_{t+1} = \mathsf{Var}[\alpha_{t+1}|Y_t] \end{array}.$$

Then the *forecast error*,

$$v_t = y_t - \mathsf{E}[y_t|Y_{t-1}] = y_t - Za_t, \tag{3}$$

which has variance

$$F_t = \mathsf{Var}[v_t|Y_{t-1}] = ZP_tZ^\mathsf{T} + H_t. \tag{4}$$

And in terms of v_t and F_t we can write

$$\begin{array}{ll} a_{t|t} = a_t + P_tZ^\mathsf{T}F_t^{-1}v_t & \\ P_{t|t} = P_t - P_tZ^\mathsf{T}F_t^{-1}ZP_t & \text{and} \end{array} \quad \begin{array}{l} a_{t+1} = Ta_{t|t} \\ P_{t+1} = TP_{t|t}T^\mathsf{T} + RQ_tR^\mathsf{T} \end{array}. \tag{5}$$

With (3), (4), and (5) we can recursively predict the state α_{t+1} conditional on Y_t, and hence also predict y_{t+1}.

Next we would like to determine the distributions of the states α_t, given Y_T, *i.e.*, to find $\hat{\alpha}_t = \mathsf{E}[\alpha_t|Y_T]$ and $V_t = \mathsf{Var}[\alpha_t|Y_T]$. Let

$$L_t = T - TP_tZ^\mathsf{T}F_t^{-1}Z, \tag{6}$$

which can be computed as we compute the F_t and P_t with the Kalman filter above. Also, set $r_T = 0 \in \mathbb{R}^n$ and $N_T = 0 \in \mathbb{R}^{n\times n}$, and then we can recursively (decreasing) compute:

$$\begin{array}{ll} r_{t-1} = Z^\mathsf{T}F_t^{-1}v_t + L_t^\mathsf{T}r_t & \\ N_{t-1} = Z^\mathsf{T}F_t^{-1}Z + L_t^\mathsf{T}N_tL_t & \text{and} \end{array} \quad \begin{array}{l} \hat{\alpha}_t = a_t + P_tr_{t-1} \\ V_t = P_t - P_tN_{t-1}P_t \end{array}. \tag{7}$$

(6) and (7) constitute the *state smoothing* algorithm.

Notice that the predictions for the states, and thence the observations, in the Kalman filter, and also the estimates for the states in the smoothing algorithm, depend on the parameters in Z, T, H_t, and Q_t (and also on a_1 and P_1). To complete the calculation we find the maximum likelihood values of these parameters. Writing all the parameters of the model as θ, the log-likelihood of the observations is

$$\log \mathcal{L}(Y_T|\theta) = -\frac{Tk}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{T}\log|\det F_t| + v_t^\mathsf{T}F_t^{-1}v_t. \tag{8}$$

We compute the θ^* that maximizes (8), and then use that to compute the smoothed states. Doing this recursively allows us to update the parameters with each new observation, and can be used in the prediction setting.

State space models for the NetMob23 data

To analyze the NetMob23 timeseries we have to deal with the time change on 31 March 2019 that jumps from 02:00 to 03:00. If we take clock time as our index, this means we are missing four 15 minute time intervals of data, namely 02:15, 02:30, 02:45, and 03:00. The filtering and smoothing algorithms can be modified to deal with this, simply by setting $Z = 0$ for those timesteps t in (3)–(7). This is one of the advantages of this method for timeseries analysis.

Two striking features of the NetMob23 data are the daily and weekly approximate periodicities. A simple linear Gaussian state space model for the timeseries of any single scalar value in the NetMob23 data that captures the daily periodicity is

$$\begin{aligned} y_t &= \mu_t + \delta_t + \epsilon_t & \epsilon_t &\sim \mathcal{N}(0, \sigma_\epsilon^2) \\ \mu_{t+1} &= \mu_t + \xi_t & \xi_t &\sim \mathcal{N}(0, \sigma_\xi^2) \\ \delta_{t+1} &= -(\delta_t + \cdots + \delta_{t-(m_1-2)}) + \eta_t & \eta_t &\sim \mathcal{N}(0, \sigma_\eta^2), \end{aligned} \tag{9}$$

where μ_t is the *local level* and δ_t is the daily periodic component, so $m_1 = 4 \times 24$. In this model the states are

$$\alpha_t = (\mu_t, \delta_t, \delta_{t-1}, \ldots, \delta_{t-m_1+2})$$

so $Z = (\,1 \quad 1 \quad 0 \quad \cdots \quad 0\,)$, while

$$T = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & -1 & \cdots & -1 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix} \quad \text{and} \quad R = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix}.$$

Finally, the parameters in the model are the covariance matrices for the noise, namely

$$H = (\sigma_\epsilon^2) \quad \text{and} \quad Q = \begin{pmatrix} \sigma_\xi^2 & 0 \\ 0 & \sigma_\eta^2 \end{pmatrix}.$$

A more comprehensive model than (9) would also include the weekly approximate periodicity. Although this periodicity is less regular than the daily approximate periodicity, were we to be using the model to forecast, it would be important to include. In the example applications we will consider, however, including only the weekly component gives a sufficiently good fit to the data, as shown in Figure 1, which plots the download volume for Netflix in Paris during March 2019. In this figure the data are in red, while the ± 2 standard deviation (approximately 95%) confidence interval for the model is shown in blue. Almost of the data points fall within this interval.

Notice, however, that there is at least one data point far outside the blue confidence interval, on March 29. Figure 2 shows the day before and after this point, which in this plot

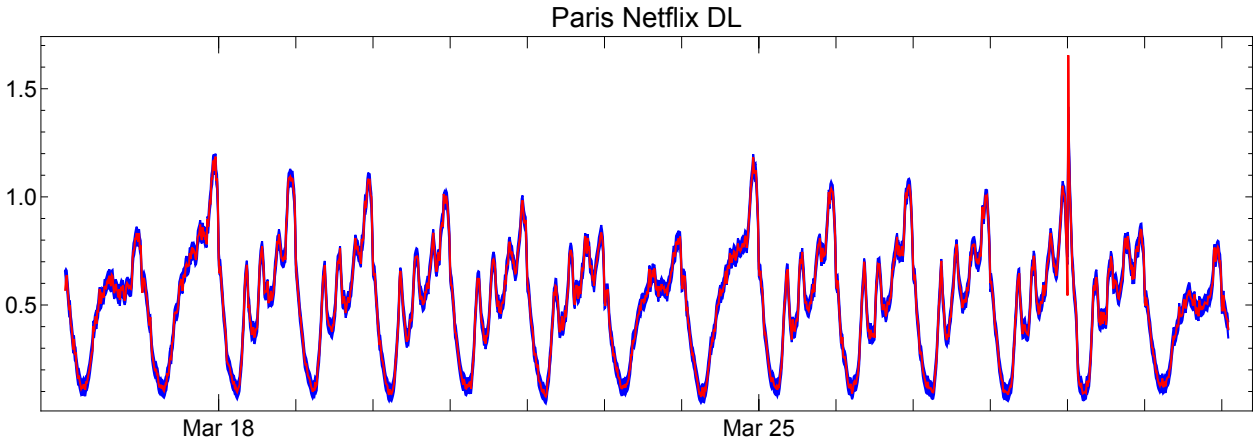

Fig. 1. Netflix download (DL) data aggregated for Paris over 15 minute intervals is shown in red. The estimated state space model with its 95% confidence intervals is shown in blue. Vertical unit is 10^{10}.

can be identified as the download volume in the 00:00 to 00:15 time interval on March 29. Further inspection reveals that the download volume had been decreasing as usual before midnight, whence it shot up, and then stayed comparatively high for the next couple of hours. This constitutes an anomaly, and seems to call for a social explanation. To confirm that it is not just a statistical fluctuation, we compare with the same data for another city, Bordeaux, shown on the right in Figure 2. Since the same anomaly occurs there and then, we must try to explain it. On March 29, 2019, Netflix released its third French language series, *Osmosis* [4]; and Netflix typically releases its new shows at midnight [5]. We believe this explains the anomalous download pattern after midnight on March 29, 2019. This example illustrates the use of a state space model to locate an anomaly more subtle than, for example, service outages [1].

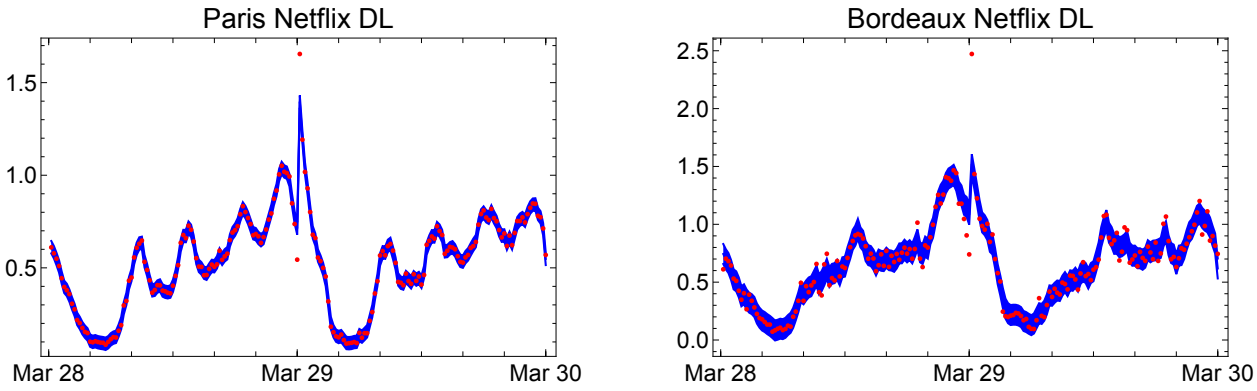

Fig. 2. Netflix download (DL) data around 00:00 March 29 for Paris and Bordeaux. Vertical units are 10^{10} and 10^9, respectively. (The larger variances for Bordeaux are due to the smaller volumes.)

Earth mover's distance for spatial analysis

In any time interval, the usage of each mobile app has a geographical distribution, approx-

imated in the NetMob23 data by a function $q : \mathcal{T} \to \mathbb{N}$, where $\mathcal{T}$ is the set of $100m \times 100m$ tiles provided [1]. It is useful to be able to quantify by how much this distribution differs from some other distribution, *e.g.*, population.

The earth mover's distance (EMD) between two non-negative functions p and q on a discrete metric space (X, d) is determined by computing a flow f_{ij} which we think of as being from the larger to the smaller ℓ^1 norm function [6]. That is, the flow should satisfy:

$$\begin{aligned} \sum_j f_{ij} &\le p_i \\ \sum_i f_{ij} &\le q_j \\ \sum_{i,j} f_{ij} &= \min\{\sum_i p_i, \sum_j q_j\} \end{aligned} \tag{10}$$

and it should minimize $\sum_{i,j} f_{ij} d_{ij}$. Let f^* be the minimizing flow satisfying the conditions (10). Then

$$\mathrm{EMD}(p, q) = \frac{\sum_{i,j} f^*_{ij} d_{ij}}{\sum_{i,j} f^*_{ij}}. \tag{11}$$

When p and q are probability distributions, the inequalities in (10) are saturated by f^*, and the denominator in (11) is just 1. In the more general setting of unnormalized p and q, the interpretation is that (some of) the larger total amount of "earth" gets moved to achieve the smaller total amount distribution with minimum effort, and then the excess "earth" is removed with no cost. For example, suppose we have these two distributions:

0	2
0	2

3	1
1	0

Making the distance between adjacent tiles 1, and between diagonal tiles $\sqrt{2}$, the minimum flow here is for 1 in the tile with 3 to move right, and 1 to move diagonally down; and for the 1 in the lower left tile to move right. The amount of work done is then $2 + \sqrt{2}$, since the remaining 1 in the tile with 3 evaporates with no cost. Since the total flow is 4 (the 3 that move plus the 1 in the upper right corner that stays there), the EMD between these two distributions is $(2 + \sqrt{2})/4$.

Comparing NetMob23 data with the population distribution

The idea that cell phone data can be used to track human mobility goes back to shortly after cell phones became ubiquitous in certain populations [7]. The NetMob23 data does not allow us to track the positions of individual users over time, nor does it include call data, both approaches that have been used to analyze mobility in the past. It does, however, allow us to measure the geographical distribution of users of any one of the included mobile

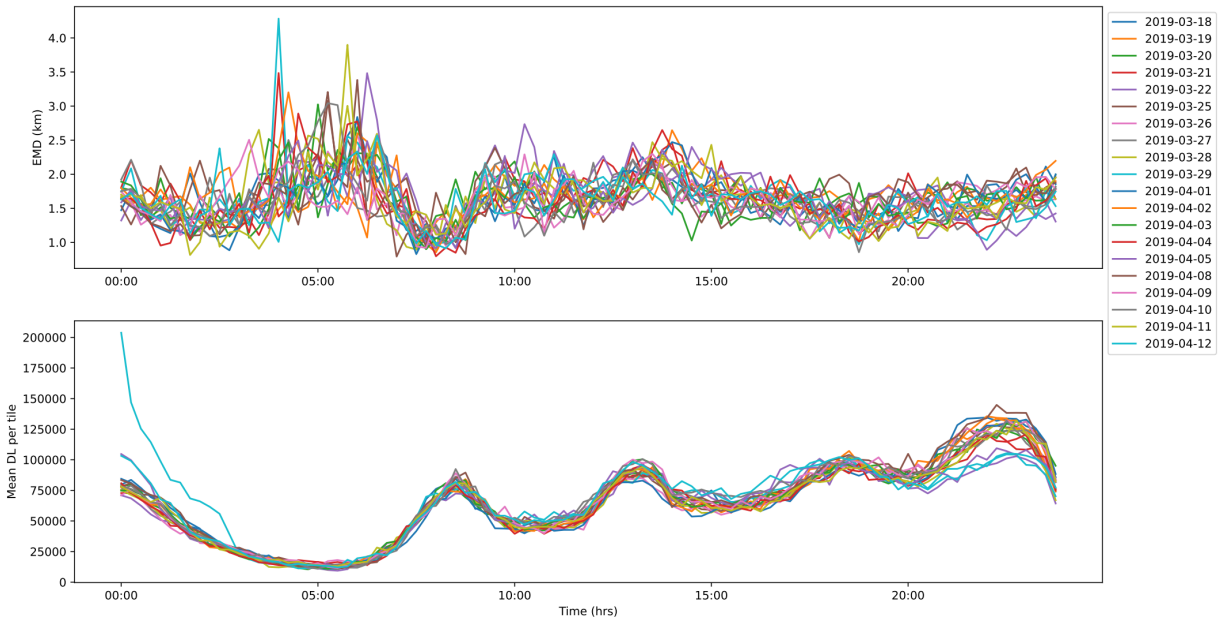

Fig. 3. Netflix download (DL) timeseries for 20 weekdays in Paris: volume (lower) and EMD (upper).

services at 15 minute time intervals. This distribution can be compared with the underlying population distribution to obtain information about collective mobility.

We use the population data for 2019 from INSEE [8]. For each IRIS (Ilots Regroupés pour l'Information Statistique; geographical regions with populations of about 2000) we compute the population density. Then to each tile in the NetMob23 data we assign the population density of the IRIS containing the tile's centroid. To reduce the computational load we aggregate tiles into macrotiles of dimension 40×40, and compute the population of each macrotile from the population densities assigned to the tiles comprising it. We define p to be the population distribution normalized by the total population.

To compare with this macrotile population distribution we also aggregate app usage of the corresponding 40×40 tiles, and then let q be this normalized by the total usage of that particular app. Now we can compute $\mathrm{EMD}(p, q)$ during each 15 minute time interval. Figure 3 shows the resulting Netflix DL EMD timeseries for weekdays in Paris, above a plot of the Netflix DL volume. We notice again the anomalously large volume after midnight on March 29, 2019 (the cyan curve in the lower plot). We also notice the consistently increased volumes around 08:00, 13:00, 18:00, and from 21:00 to 23:00, which we may attribute to people watching Netflix during meals, and in the late evening. The EMD timeseries show that the difference from the underlying population distribution is smallest in the morning, and largest at lunchtime, and in the very early morning, although since the volume is so low then that the variance is very large.

Other apps show different patterns, and differences between regions. Figure 4 shows the corresponding plots for Uber uploads (UL) on weekends in Lyon, and in Marseille. The

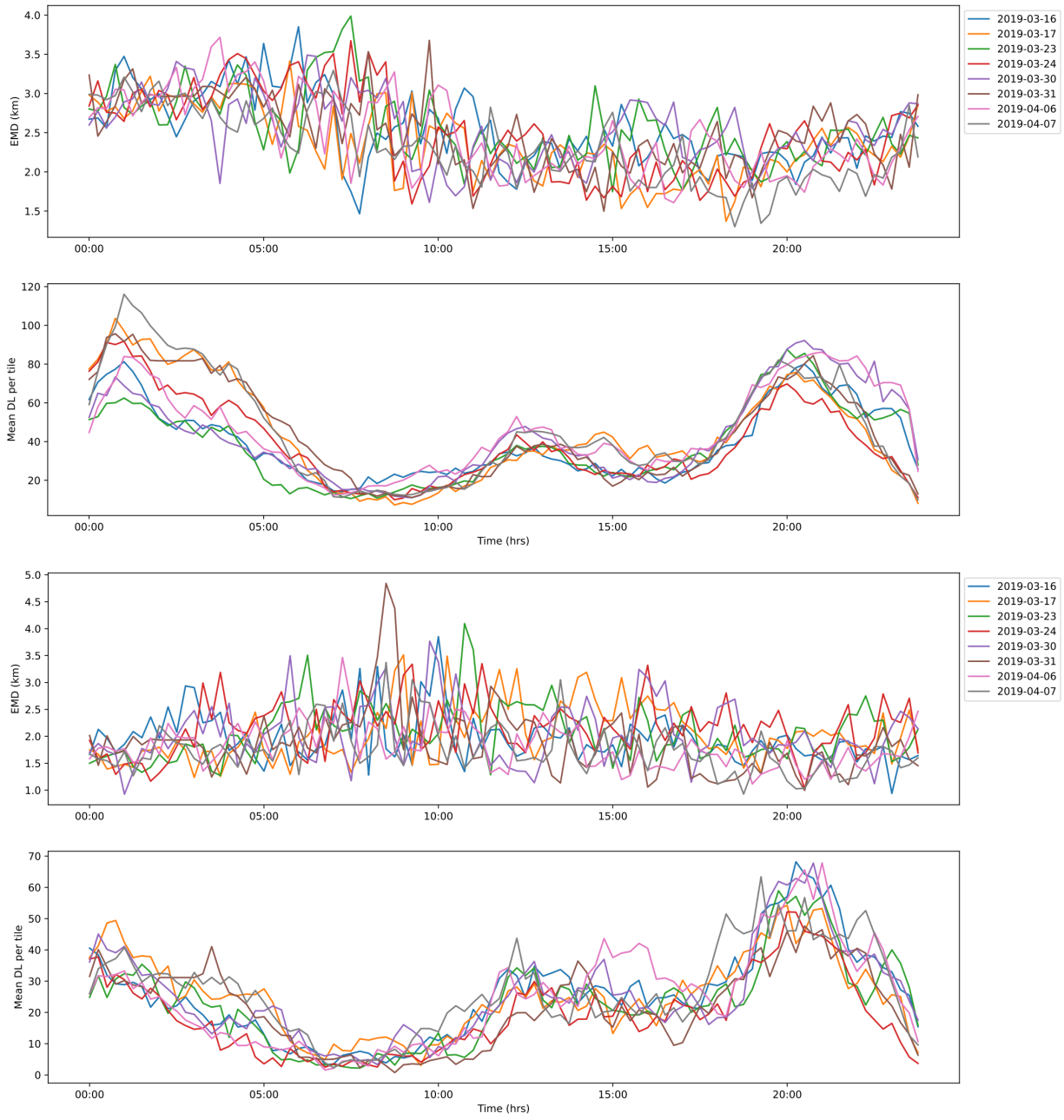

Fig. 4. Uber upload (UL) EMD and volume timeseries for 8 weekend days in Lyon (top) and Marseille (bottom).

volume pattern is similar for these two metropolitan areas, although about double in Lyon, which is plausible given its larger 'greater Lyon' population [8]. More interestingly, the EMD patterns are different, with larger distances for Lyon, but also larger distances at night than during the day, unlike Marseille which has approximately constant (albeit heteroskedastic) EMD. This is just one example of the EMD revealing cultural differences between different geographical regions.

Flows and spatial anomalies

The EMD can also be used to compare app usage distributions in two time intervals. In this setting the total app usage typically differs in different time intervals, and we would lose information if we normalized the distributions. Thus we optimize with the *inequalities* in (10), and it is as if when the total usage decreases from one time interval to another, existing users move to the new distribution and the excess users turn off; while when the total usage increases from one time interval to another, the existing users move towards the new distribution and new users turn on at locations where necessary. At the core of the calculation of an EMD is the computation of the minimizing flow f^* in (10–11). While it does not necessarily describe the real movement of users, it does describe their *effective* movement.

We can use this flow to analyze events that are localized in time and space. For example, la cathédrale Notre-Dame de Paris caught fire on April 15, 2019, at about 18:15 [9]. In Figure 5 we illustrate the flow computed for the difference between the Instagram usage in the 18:15-18:30 time interval and the 18:30-18:45 time interval. On the left is the flow on April 15, with red arrows showing flow from one tile to another, and red tiles indicating locations of new users. There is clearly an increase in usage on both banks of the Seine, on l'île de la Cité, where la cathédrale Notre-Dame de Paris is located, and just upriver on l'île Saint Louis, as well as a general flow toward this area. To support our interpretation that this is users reacting to the fire, on the right is the same flow for a week earlier, April 8, at the same time. Then the flow is generally northeastward, with a decrease in users indicated by the green tiles.

Fig. 5. Optimal flow of Instagram users around Notre Dame at the time of the fire, on April 15, 2019, and at the same time the previous week.

Extensions

Having estimated a state space model from data (y_t), it is straightforward to generate *simulated data*, conditional on the real data: one simply adds Gaussian noise of estimated variances to each $Z\hat{\alpha}_t$. Thus a second application of this formalism could be to create simulated mobile app use data. We emphasize, however, that it is hard to imagine any simulation procedure, and certainly not this one, that would preserve small, but explainable anomalies like the Netflix downloads on March 29, 2019.

Although we did not use it in our examples, y_t need not be a scalar. It could be the data for a single app listed for all the tiles in one city, for example, or the data for multiple apps. That is, the formalism extends to multivariate time series. Multi-geographical-variate state space models can include dependence of the variables at one tile on the variables of other tiles at the previous timestep in the equation for α_{t+1}. Once estimated, such models will imply a flow, which can be compared with the optimal flow from the EMD calculation to obtain richer patterns to compare between times and places as in the examples we have shown.

Finally, both the EMD and flow calculations can be done not only for geographically different regions, but for regions differing on some other dimension, *e.g.*, wealth, or education, or age. Similarly, exogenous covariates like these can be included into state space models. Thus each of these methods can be used for further exploration of interactions between social factors and mobile app use.

Acknowledgements

DAM's research on this project was conducted partially while visiting the Okinawa Institute of Science and Technology (OIST) through the Theoretical Sciences Visiting Program (TSVP).

References

[1] Orlando E. Mariínez-Durive, Sachit Mishra, Cezary Ziemlicki, Stefania Rubrichi, Zbigniew Smoreda and Marco Fiore, "The NetMob23 Dataset: A high-resolution multi-region service-level mobile data traffic cartography", `arXiv:2305.06933v2 [cs.NI]`.

[2] James Durbin and Siem Jan Koopman, *Time Series Analysis by State Space Methods*, 2nd edition (Oxford: Oxford University Press 2012).

[3] Rudolph E. Kálmán, "A new approach to linear filtering and prediction problems", *Journal of Basic Engineering* **82** (1960) 35–45.

[4] Catherine Balle, Carine Didier, Stéphanie Guerrin, Emmanuel Marolle and Michel Valentin, "'Hanna', 'Osmosis', 'The Highwaymen' ... les séries et films à regarder ce week-end", *Le Parisien* (29 March 2019); `https://www.leparisien.fr/culture-loisirs/series/hanna-osmosis-the-highwaymen-les-series-et-films-a-regarder-ce-week-end-29-03-2019-8042511.php`.

[5] Netflix, Centre d'aide, "Quand Netflix diffuse-t-il les nouvelles séries et nouveaux films

?", https://help.netflix.com/fr/node/118959.

[6] Gaspard Monge, "*Mémoire sur la théorie des déblais et des remblais*", *Histoire de l'Académie Royale des Sciences de Paris, avec les Mémoires de Mathématique et de Physique* (1781) 666–704.

[7] Carlo Ratti, Dennis Frenchman, Riccardo M. Pulselli and Sarah Williams, "Mobile landscapes: Using location data from cell phones for urban analysis", *Environment and Planning B: Planning and Design* **33** (2006) 727–748.

[8] Institut national de la statistique et des études économiques, Population en 2019, https://www.insee.fr/fr/statistiques/6543200/.

[9] Stéphane Joahny, "Incendie de Notre-Dame de Paris : pourquoi la piste d'une défaillance électrique est privilégiée", *Le Journal de Dimanche* (21 April 2019); https://www.lejdd.fr/Societe/incendie-de-notre-dame-de-paris-pourquoi-la-piste-dune-defaillance-electrique-est-privilegiee-3894124.