Learning Localized Spatio-Temporal Models From Streaming Data

ICML

Muhammad Osama, Dave Zachariah, Thomas Schön

Uppsala University

July 12, 2018
Many real world processes of interest are spatio-temporal in nature: weather, brain signals etc.
Problem Setup

Illustration with 1D space.

Dataset: $\mathcal{D}_n = \{ (s_1, t_1, y_1), \ldots, (s_n, t_n, y_n) \}$
Illustration with 1D space.

Predict $y$ at unobserved test points $(s, t)$
Problem Setup

Illustration with 1D space.

Contiguous unobserved regions
Illustration with 1D space.
Problem Setup

Illustration with 1D space.

Data obtained in a streaming fashion: $n = 1, 2, \ldots, N$
Problem Setup

Illustration with 1D space.

Data obtained in a *streaming* fashion: $n = 1, 2, \ldots, N$
Problem Setup

Illustration with 1D space.

Data obtained in a **streaming** fashion: $n = 1, 2, \ldots, N$
Problem Setup

Illustration with 1D space.

Data obtained in a streaming fashion: $n = 1, 2, \ldots, N$
Approach

- Mean Square Error (MSE) optimal unbiased predictor

\[ \hat{y}(s, t) = \lambda^\top(s, t)y, \]  \hspace{1cm} (1)

where \( y = \text{col}\{y_1, y_2, \ldots, y_n\} \).
Approach

- Mean Square Error (MSE) optimal unbiased predictor

\[
\hat{y}(s, t) = \lambda^\top(s, t)y,
\]  

where \( y = \text{col}\{y_1, y_2, \ldots, y_n\} \).

\[
\lambda(s, t) \triangleq \arg\min_{\lambda \in U} \mathbb{E} \left[ (y - \lambda^\top y)^2 \mid s, t \right].
\]
Approach

- Mean Square Error (MSE) optimal unbiased predictor

\[ \hat{y}(s, t) = \lambda^\top(s, t)y, \]  

(1)

where \( y = \text{col}\{y_1, y_2, \ldots, y_n\} \).

\[ \lambda(s, t) \triangleq \arg \min_{\lambda \in \mathcal{U}} \mathbb{E} \left[ (y - \lambda^\top y)^2 \mid s, t \right]. \]

- Model class

\[
\begin{align*}
\mathbb{E}[y] &= \eta, \\
\text{Cov}[y, y'] &= \phi^\top(s, t)\Theta \phi(s', t') + \theta_0 \delta(s, s')\delta(t, t')
\end{align*}
\]

(2)

where \( \Theta = \text{diag}\{\theta_1, \theta_2, \ldots, \theta_p\} \) and \( \phi(s, t) \) is a \( p \times 1 \) vector obtained from spatio-temporal basis function.
Approach

- Mean Square Error (MSE) optimal unbiased predictor

\[ \hat{y}(s, t) = \lambda^\top(s, t)y, \quad (1) \]

where \( y = \text{col}\{y_1, y_2, \ldots, y_n\} \).

\[ \lambda(s, t) \triangleq \arg \min_{\lambda \in \mathcal{U}} E \left[ (y - \lambda^\top y)^2 \mid s, t \right]. \]

- Model class

\[
\begin{cases}
E[y] = \eta, \\
\text{Cov}[y, y'] = \phi^\top(s, t) \Theta \phi(s', t') + \theta_0 \delta(s, s') \delta(t, t')
\end{cases}
\quad (2)
\]

where \( \Theta = \text{diag}\{\theta_1, \theta_2, \ldots, \theta_p\} \) and \( \phi(s, t) \) is a \( p \times 1 \) vector obtained from spatio-temporal basis function.

- Hyperparameters \( \theta = \{\Theta, \theta_0\} \) determine \( \lambda_\theta(s, t) \) in (1)
Learning Hyperparameters

- Fit model covariance $\mathbf{K}_\theta = \text{Cov}_\theta[\mathbf{y}]$ to sample covariance $\tilde{\mathbf{K}} = (\mathbf{y} - 1\eta)(\mathbf{y} - 1\eta)^\top$

$$\hat{\theta} = \arg\min_{\theta} \|\tilde{\mathbf{K}} - \mathbf{K}_\theta\|^2_{\mathbf{K}_\theta^{-1}}$$

Plug in learned $\hat{\theta}$ into $\lambda_\theta$ in (1)

$$\hat{y}_\theta(s, t) = \lambda_{\hat{\theta}}^\top(s, t)\mathbf{y}$$

$$\hat{y}_\theta(s, t) \equiv \alpha^\top(s, t)\mathbf{w}^*$$

where $\alpha(s, t) = \text{col}\{1, \phi(s, t)\}$ and $\mathbf{w}^*$ can be updated with streaming data in total runtime $\mathcal{O}(np^2)$ [4].
Covariance Model

\[ \text{Cov}[y, y'] = \phi^\top (s, t) \otimes \phi(s', t') + \theta_0 \delta(s, s') \delta(t, t') \]
Covariance Model

\[
\text{Cov}[y, y'] = \phi^\top(s, t) \odot \phi(s', t') + \theta_0 \delta(s, s') \delta(t, t')
\]

\[
\phi(s, t) = \begin{bmatrix}
\psi_0(t) \\
\psi_1(t) \\
\vdots \\
\psi_{N_t}(t)
\end{bmatrix} \otimes \begin{bmatrix}
\varphi_1(s_1) \\
\varphi_2(s_1) \\
\vdots \\
\varphi_{N_s}(s_1)
\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}
\varphi_1(s_d) \\
\varphi_2(s_d) \\
\vdots \\
\varphi_{N_s}(s_d)
\end{bmatrix}
\]

(a) Different Fourier-like components of temporal basis vector [2] (b) Different components of the spatial basis vector for two-dimensional space based on cubic b-splines.
Localized Covariance Structures

Examples of resulting covariance structure with 1D space. (a)-(c)
Contour plot of $\text{Cov}[y, y']$ where $y$ is the point marked by red cross and $y'$ is the rest of the space-time points.
Simulation: Varying Seasonalities Across Space

Noisy realization of a spatio-temporal process with 1D space.
Simulation: Varying Seasonalities Across Space

Noise realization of a spatio-temporal process with 1D space.
Simulation: Varying Seasonalities Across Space

(a) Mean Square Error (MSE) of the proposed method. (b) MSE of Gaussian Process Regression (GPR) with periodic Matérn kernel.
Simulation: Varying Seasonalities Across Space

(a) MSE of the proposed method. (b) MSE of $G_{PR}$ with spectral mixture kernel.
Real Data: Precipitation

(a) Monthly precipitation for 60 months over central United States [1].

(b)
Real Data: Precipitation

(a) Monthly precipitation for 60 months over central United States [1].
Real Data: Pacific Sea Surface Temperature

(a) Real data visualization of sea surface temperature anomalies.

(b) Time series plot showing monthly temperature anomalies for 36 months.

Monthly temperature anomalies in sea surface temperature for 36 months [3].
Conclusion

▶ We proposed a method in which a spatio-temporal predictor $\hat{y}(s, t)$ can be learned and updated sequentially as spatio-temporal data is obtained as a stream.

▶ It is capable of capturing spatially varying temporal patterns, using a non-stationary covariance model that is learned using a covariance-fitting approach.

▶ Code available at https://github.com/Muhammad-Osama/Localized-Spatio-temporal-Models

▶ Poster 143 at Hall B
Thank you


