VE472 Lecture 12

Jing Liu

UM-SJTU Joint Institute

Summer

Q: Why and how time series can provide information for forecasting each other?

$$
\begin{array}{c}
Y_{11}, Y_{12} \cdots\cdots Y_{1T_1} \\
Y_{21}, Y_{22} \cdots\cdots Y_{2T_2} \\
\vdots \quad \ddots \ddots \ddots \quad \vdots \\
Y_{n1}, Y_{n2} \cdots\cdots Y_{nT_n}
\end{array}
$$

- You might be attempted to construct a usual linear regression model, e.g.

$$
\begin{aligned}
y &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \varepsilon \qquad \text{where} \quad \varepsilon \sim \text{Normal}\left(0, \sigma^2\right) \\
Y_{1t} &= \beta_0 + \beta_2 Y_{2t} + \beta_3 Y_{3t} + \cdots + \beta_n Y_{nt} + Z_t \qquad \text{where} \quad Z_t \sim \text{ARIMA}
\end{aligned}
$$

but if you consider it a bit longer, you should immediately notice a problem...

可能不知道Y2t, Y3t ....

- So the usual linear model that tries to use other variables, e.g.

$$
Y_{1t} = \beta_0 + \beta_2 Y_{2,t-1} + \beta_3 Y_{3,t-1} + \cdots + \beta_n Y_{n,t-1} + Z_t
$$

lag一下，但是只include一种lag

can only involve lags, and the prediction is limited by the order of the lags.

- So clearly such predictive models by themselves are not useful!

$$Y_{1t} = \beta_0 + \beta_1 Y_{2,t-1} + \beta_2 Y_{3,t-1} + \cdots + \beta_k Y_{k,t-1} + Z_t$$

only applicable when Y1. depend on Y2., Y3., ... 而不能是Y2., Y3. , ...depend on Y1.

- However, if combining the following regression

$$Y_{1t} = \beta_0 + \beta_2 Y_{2t} + \beta_3 Y_{3t} + \cdots + \beta_n Y_{nt} + Z_t$$

with some ARIMA model (or other forecasting model) for each time series,

$$\Phi_i(\mathcal{L}^{s_i})\phi_i(\mathcal{L})\nabla^{d_i}\nabla_{s_i}^{D_i} Y_{it} = \Theta_i(\mathcal{L}^{s_i})\theta_i(\mathcal{L})\varepsilon_{it} \qquad \text{where} \quad i \neq 1$$

i: each time series

then such a regression model can give forecasts for Y_{1t} for $t > T$.

- In general, ARIMA regression can have lags as well, that is,

$$Y_{1t} = \beta_0 + \beta_2(\mathcal{L})\, Y_{2t} + \beta_3(\mathcal{L})\, Y_{2t} + \cdots + \beta_n(\mathcal{L})\, Y_{nt} + Z_t$$

where $\beta_i(\mathcal{L})$ are some polynomial lag operator. So a lot of parameters!

- It is important to notice the time series Y_{1t} depends on other times series, but other series does not depend on Y_{1t}, if not, then the above is no good!

- If there is no clear one-way dependency between our variables, then one way is to use the vector version of ARIMA (VARIMA), for example

Y: 长度为n vector

$$\mathbf{Y}_t = \boldsymbol{\varphi}\mathbf{Y}_{t-1} + \boldsymbol{\varepsilon}_t \quad \text{where} \quad \mathbf{Y}_t, \mathbf{Y}_{t-1}, \boldsymbol{\varepsilon}_t \in \mathbb{R}^n, \boldsymbol{\varphi} \in \mathbb{R}^{n\times n}.$$

- The main diagonal elements of $\boldsymbol{\varphi}$ define the relation within each time series, while the off-diagonal elements are for the ones between the time series.
- For example, $\boldsymbol{\varphi} = \begin{bmatrix} \varphi_{11} & 0 \\ \varphi_{21} & \varphi_{22} \end{bmatrix}$ gives a model where the 1st variable doesn't depend on the 2nd variable but correlated to it, thus it is totally dynamic.
- However, unlike univariate case, the invertibility of the vector polynomial lag operators is even harder to determine, and even the model is invertible, i.e.

$$\mathbf{Y}_t = \sum_{u=0}^{\infty} \boldsymbol{\varphi}^u \boldsymbol{\varepsilon}_{t-u}$$

vector ARIMA not very commonly used in practice

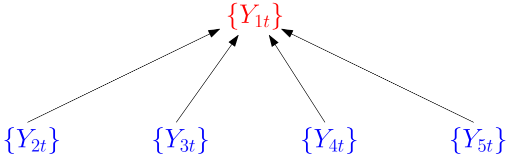
might not be unique, that is, there are multiple sets of parameters consistent with the data, so often we cannot determine the parameters numerically.

Q: How else can time series provide information for forecasting each other?

$\{Y_{1t}\}$
$\{Y_{2t}\}$
$\{Y_{3t}\}$
$\{Y_{4t}\}$
$\{Y_{5t}\}$

$\longrightarrow$

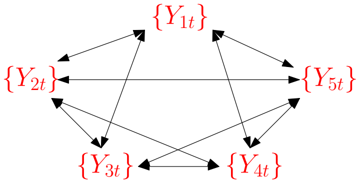
ARIMA Regression



$\{Y_{1t}\}$
$\{Y_{2t}\}$
$\{Y_{3t}\}$
$\{Y_{4t}\}$
$\{Y_{5t}\}$

$\longrightarrow$

VARIMA



- Suppose there are a number of fundamental time series

$$\{H_{\ell t}\} \qquad \text{where} \quad \ell = 1, 2, \ldots, K,$$

that govern all time series we observed and each time series we observed

$$Y_{it} = \sum_{\ell=1}^{K} \lambda_\ell H_{\ell t}$$

is a linear combination of those $\{H_{\ell t}\}$ which are often not directly observed.

- This framework is similar to discrete Fourier transform (DFT) in some ways,

$$\mathcal{F}\left(\{Y_{it}\}\right) = \mathcal{F}\left(\mathbf{y}_i\right) = \left\{ \sum_{k=0}^{K-1} Y_{i,k+1} \exp\left(-\mathrm{j}2\pi\ell k/K\right) \right\}_{\ell=0}^{K-1} = \left\{ \frac{\mathbf{b}_\ell^{\mathrm{H}} \mathbf{y}_i}{\mathbf{b}_\ell^{\mathrm{H}} \mathbf{b}_\ell} \right\}_{\ell=0}^{K-1}$$

where $\mathrm{j} = \sqrt{-1}$ and $\mathcal{B} = \{\mathbf{b}_0, \ldots, \mathbf{b}_{K-1}\}$ is the orthogonal basis of the form

$$\mathbf{b}_\ell^{\mathrm{T}} = \frac{1}{K} \begin{bmatrix} \exp\left(\mathrm{j}2\pi\ell \cdot 0/K\right) & \exp\left(\mathrm{j}2\pi\ell \cdot 1/K\right) & \cdots & \exp\left(\mathrm{j}2\pi\ell \cdot (K-1)/K\right) \end{bmatrix}$$

- Putting DFT in terms of the framework that we are planning to use, we have

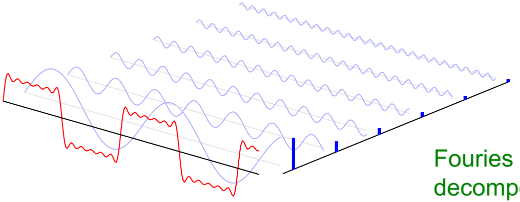
$$Y_{it} = \sum_{\ell=1}^{K} \mathrm{Re}\left(\lambda_\ell H_{\ell t}\right) \qquad \text{if } Y_{it} \in \mathbb{R} \text{ for all } i \text{ and } t,$$

only real signal

where

$$\lambda_\ell = \frac{\mathbf{b}_{\ell-1}^{\mathrm{H}} \mathbf{y}_i}{\mathbf{b}_{\ell-1}^{\mathrm{H}} \mathbf{b}_{\ell-1}} \qquad \text{and} \qquad H_{\ell t} = [\mathbf{b}_{\ell-1}]_t = \frac{1}{K} \exp\left(\mathrm{j}2\pi(\ell-1)t/K\right)$$

so λ_ℓ gives the weight, and $H_{\ell t}$ the sinusoidal shape of different frequency.



Fouries frequency decomposition

- Of course, rather than DFT, we are looking for K hidden time series so that

$$Y_{it} = \sum_{\ell=1}^{K} \lambda_\ell H_{\ell t}$$

but like DFT, we need find a basis $\mathcal{B}$, which will be our hidden time series

$$\mathcal{B} = \{\mathbf{H}_1, \dots, \mathbf{H}_K\}$$

and a way to rank the importance of a particular $\mathbf{H}_\ell$ to a particular series

$$\{Y_{it}\}$$

according to the observed data in the form of the data matrix

$$\mathbf{Y} = \begin{bmatrix} Y_{11} & Y_{12} & \cdots\cdots & Y_{1T_1} \\ Y_{21} & Y_{22} & \cdots\cdots & Y_{2T_2} \\ \vdots & \ddots & \ddots & \vdots \\ Y_{n1} & Y_{n2} & \cdots\cdots & Y_{nT_n} \end{bmatrix}$$

- To motivate it further, suppose the following data matrix is collected

$$
\mathbf{Y} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}
$$

big dataset not practical

$$
= \begin{bmatrix} 1/\sqrt{31} & 0 \\ 2/\sqrt{31} & 0 \\ 1/\sqrt{31} & 0 \\ 5/\sqrt{31} & 0 \\ 0 & 2/\sqrt{14} \\ 0 & 3/\sqrt{14} \\ 0 & 1/\sqrt{14} \end{bmatrix} \begin{bmatrix} \sqrt{91} & 0 \\ 0 & \sqrt{28} \end{bmatrix} \begin{bmatrix} 1/\sqrt{3} & 1/\sqrt{3} & 1/\sqrt{3} & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}
$$

Q: How can we do it in general?

Singular Value Decomposition

very big time series (many time series)

first K non-zero singlar values
only include non-zero singlar values

- Using the reduce SVD of the first K singular values,

$$\mathbf{Y} = \begin{bmatrix}\mathbf{U}_K & \tilde{\mathbf{U}}\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_{K\times K} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{V}_K & \tilde{\mathbf{V}}\end{bmatrix}^{\mathrm{T}}$$
$$\approx \mathbf{U}_K \boldsymbol{\Sigma}_K \mathbf{V}_K^{\mathrm{T}} = \mathbf{Y}_K$$

- Use ARIMA to forecast the K hidden components, which are given by $\mathbf{V}_K^{\mathrm{T}}$

$$\Phi_k(\mathcal{L}^{s_k})\phi_k(\mathcal{L})\nabla^{d_k}\nabla_{s_k}^{D_k} H_{kt} = \Theta_k(\mathcal{L}^{s_k})\theta_k(\mathcal{L})\varepsilon_{kt}$$

Q: Why using ARIMA on each row of $\mathbf{V}_K^{\mathrm{T}}$ is different from using it on $\mathbf{Y}$?

rows of V_k^T is orthogonal
they are independent

- To make a prediction on any of the original time series,

$$\{Y_{it}\}$$

we simply linearly combine the hidden components according to

$$\mathbf{U}_K \qquad \text{and} \qquad \boldsymbol{\Sigma}_K$$