Financial Time Series – Nonparametric methods in time series

Andreas Petersson

TMS088/MSA410 – May 2020

Mathematical Sciences, Chalmers University of Technology & University of Gothenburg, Sweden

Nonparametric methods

- When structure is non-linear but unknown

Nonparametric methods

- When structure is non-linear but unknown
- Highly data dependent, risk of overfitting

Nonparametric methods

- When structure is non-linear but unknown
- Highly data dependent, risk of overfitting
- No/few parameters/models $\implies$ no error distribution

General time series model

- Let $X = (X_t, t \in \mathbb{Z})$ be given by

$$X_t = m(X_{t-r}) + Z_t,$$

  where $Z \sim IID(0, \sigma^2)$ and m is an arbitrary, smooth, but unknown function

General time series model

- Let $X = (X_t, t \in \mathbb{Z})$ be given by

$$X_t = m(X_{t-r}) + Z_t,$$

  where $Z \sim IID(0, \sigma^2)$ and m is an arbitrary, smooth, but unknown function
- More generally:

$$Y_t = m(X_t) + Z_t,$$

  for two time series X and Y

General time series model

- Let $X = (X_t, t \in \mathbb{Z})$ be given by

$$X_t = m(X_{t-r}) + Z_t,$$

  where $Z \sim IID(0, \sigma^2)$ and m is an arbitrary, smooth, but unknown function
- More generally:

$$Y_t = m(X_t) + Z_t,$$

  for two time series X and Y
- For Y independent of Z, $m(X_t) \approx \mathbb{E}(Y_t | X_t)$

General time series model

- Let $X = (X_t, t \in \mathbb{Z})$ be given by

$$X_t = m(X_{t-r}) + Z_t,$$

where $Z \sim IID(0, \sigma^2)$ and m is an arbitrary, smooth, but unknown function

- More generally:

$$Y_t = m(X_t) + Z_t,$$

for two time series X and Y

- For Y independent of Z, $m(X_t) \approx \mathbb{E}(Y_t | X_t)$
- When $X = x$ is constant and independent of Z,

$$y_t = m(x) + Z_t$$

and taking the sample average yields

$$n^{-1} \sum_{t=1}^{n} y_t = m(x) + n^{-1} \sum_{t=1}^{n} Z_t.$$

Kernels

- Fine if m is smooth and $X_t \approx x$ almost constant

Kernels

- Fine if m is smooth and $X_t \approx x$ almost constant
- Otherwise: use a weighted average of y,

$$\hat{m}(x) := \sum_{t=1}^{n} w_t(x) y_t, \tag{1}$$

where $w_t(x)$ are larger for y_t with x_t close to x and smaller otherwise and $\sum_{t=1}^{n} w_t(x) = 1$.

Kernels

- Fine if m is smooth and $X_t \approx x$ almost constant
- Otherwise: use a weighted average of y,

$$\hat{m}(x) := \sum_{t=1}^{n} w_t(x) y_t, \tag{1}$$

where $w_t(x)$ are larger for y_t with x_t close to x and smaller otherwise and $\sum_{t=1}^{n} w_t(x) = 1$.

- *Kernel* $K : \mathbb{R} \to \mathbb{R}^+$. Typically a density function, $\int K(z)\ \mathrm{d}z = 1$.
- Rescale by *bandwidth*:

$$K_h(x) = h^{-1} K(xh^{-1}).$$

Still:

$$\int K_h(z)\ \mathrm{d}z = 1.$$

Kernel regression

Set

$$w_t(x) := \frac{K_h(x - x_t)}{\sum_{s=1}^{n} K_h(x - x_s)}.$$

Kernel regression

Set

$$w_t(x) := \frac{K_h(x - x_t)}{\sum_{s=1}^{n} K_h(x - x_s)}.$$

This yields the *Nadaraya–Watson kernel estimator*

$$\hat{m}(x) = \sum_{t=1}^{n} w_t(x) y_t = \frac{\sum_{t=1}^{n} K_h(x - x_t) y_t}{\sum_{t=1}^{n} K_h(x - x_t)}.$$

Kernel regression

Set

$$w_t(x) := \frac{K_h(x - x_t)}{\sum_{s=1}^n K_h(x - x_s)}.$$

This yields the *Nadaraya–Watson kernel estimator*

$$\hat{m}(x) = \sum_{t=1}^n w_t(x) y_t = \frac{\sum_{t=1}^n K_h(x - x_t) y_t}{\sum_{t=1}^n K_h(x - x_t)}.$$

Example: The *Gaussian kernel*

$$K_h(x) := (2\pi h^2)^{-1/2} \exp(-(2h^2)^{-1} x^2)$$

Kernel regression

Set

$$w_t(x) := \frac{K_h(x - x_t)}{\sum_{s=1}^{n} K_h(x - x_s)}.$$

This yields the *Nadaraya–Watson kernel estimator*

$$\hat{m}(x) = \sum_{t=1}^{n} w_t(x) y_t = \frac{\sum_{t=1}^{n} K_h(x - x_t) y_t}{\sum_{t=1}^{n} K_h(x - x_t)}.$$

Example: The *Gaussian kernel*

$$K_h(x) := (2\pi h^2)^{-1/2} \exp(-(2h^2)^{-1} x^2)$$

Example: The *Epanechnikov kernel*

$$K_h(x) := 0.75\, h^{-1}(1 - (x/h)^2) I(|x/h| \leq 1),$$

Kernel regression

Set

$$w_t(x) := \frac{K_h(x - x_t)}{\sum_{s=1}^n K_h(x - x_s)}.$$

This yields the *Nadaraya–Watson kernel estimator*

$$\hat{m}(x) = \sum_{t=1}^n w_t(x) y_t = \frac{\sum_{t=1}^n K_h(x - x_t) y_t}{\sum_{t=1}^n K_h(x - x_t)}.$$

Example: The *Gaussian kernel*

$$K_h(x) := (2\pi h^2)^{-1/2} \exp(-(2h^2)^{-1} x^2)$$

Example: The *Epanechnikov kernel*

$$K_h(x) := 0.75\, h^{-1}(1 - (x/h)^2) I(|x/h| \leq 1),$$

For the latter: $\hat{m}(x_t) \to y_t$ for $h \to 0$ and $\hat{m}(x_t) \to \bar{y}$ for $h \to \infty$

Bandwidth selection

Method (Bandwidth selection with MISE)

Minimize the *mean integrated squared error*:

$$\mathrm{MISE} := \mathbb{E}\left(\int_{-\infty}^{\infty} (\hat{m}(x) - m(x))^2 \, \mathrm{d}x\right),$$

where m is the true function and $\hat{m}$ the estimator which depends on h.

Bandwidth selection

Method (Bandwidth selection with MISE)

Minimize the *mean integrated squared error*:

$$\mathrm{MISE} := \mathbb{E}\left(\int_{-\infty}^{\infty} (\hat{m}(x) - m(x))^2 \, \mathrm{d}x\right),$$

where m is the true function and $\hat{m}$ the estimator which depends on h. Expand this to derive an optimal bandwidth that depends on unknown quantities.

Bandwidth selection

Method (Bandwidth selection with MISE)

Minimize the *mean integrated squared error*:

$$\mathrm{MISE} := \mathbb{E}\left(\int_{-\infty}^{\infty} (\hat{m}(x) - m(x))^2 \, \mathrm{d}x\right),$$

where m is the true function and $\hat{m}$ the estimator which depends on h. Expand this to derive an optimal bandwidth that depends on unknown quantities. Estimate these by preliminary smoothing, i.e., computing $\hat{m}$ with a reference bandwith selector. Common choices:

$$\hat{h}_{\mathsf{opt}} = \begin{cases} 1.06\, s\, n^{-1/5} & \text{for the Gaussian kernel,} \\ 2.34\, s\, n^{-1/5} & \text{for the Epanechnikov kernel,} \end{cases}$$

where s is the sample standard error of $(x_t)_{t=1}^{n}$.

Bandwidth selection

Method (Bandwidth selection with cross validation)

Leave-one-out cross validation omits (x_j, y_j). The other data points are used to find:

$$\hat{m}_{h,j}(x_j) := \sum_{t \neq j} w_t(x_j) y_t \approx y_j.$$

Method (Bandwidth selection with cross validation)

Leave-one-out cross validation omits (x_j, y_j). The other data points are used to find:

$$\hat{m}_{h,j}(x_j) := \sum_{t \neq j} w_t(x_j) y_t \approx y_j.$$

Repeat for the remaining $n-1$ points and set

$$\mathsf{CV}(h) := \sum_{j=1}^{n} (y_j - \hat{m}_{h,j}(x_j))^2 W(x_j),$$

where W is a nonnegative weight function satisfying $\sum_{j=1}^{n} W(x_j) = 1$.

Method (Bandwidth selection with cross validation)

Leave-one-out cross validation omits (x_j, y_j). The other data points are used to find:

$$\hat{m}_{h,j}(x_j) := \sum_{t \neq j} w_t(x_j) y_t \approx y_j.$$

Repeat for the remaining $n-1$ points and set

$$\mathsf{CV}(h) := \sum_{j=1}^{n} (y_j - \hat{m}_{h,j}(x_j))^2 W(x_j),$$

where W is a nonnegative weight function satisfying $\sum_{j=1}^{n} W(x_j) = 1$.Set $h_{\mathsf{opt}} := \arg\min_h \mathsf{CV}(h)$.

Local linear regression

- Idea: $\hat{m}(x)$ can equally be defined as

$$\arg\min_a \sum_{t=1}^{n} (y_t - a)^2 K_h(x - x_t).$$

Local linear regression

- Idea: $\hat{m}(x)$ can equally be defined as

$$\arg\min_a \sum_{t=1}^{n} (y_t - a)^2 K_h(x - x_t).$$

- The *local linear regression method*:

$$(\hat{a}, \hat{b}) = \arg\min_{a,b} \sum_{t=1}^{n} (y_t - a - b(x - x_t))^2 K_h(x - x_t)$$

- Beats kernel regression if m is twice continuously differentiable

Local linear regression

A closed form solution:

$$\hat{a} = \frac{\sum_{t=1}^{n} w_t(x) y_t}{\sum_{t=1}^{n} w_t(x)} \text{ and } \hat{b} = \frac{\sum_{t=1}^{n} \tilde{w}_t(x) y_t}{\sum_{t=1}^{n} w_t(x)}$$

where

$$w_t(x) := K_h(x - x_t)(s_{n,2}(x) - (x - x_t)s_{n,1}(x)),$$

$$\tilde{w}_t(x) := K_h(x - x_t)((x - x_t)s_{n,0}(x) - s_{n,1}(x)),$$

and

$$s_{n,j}(x) := \sum_{t=1}^{n} K_h(x - x_t)(x - x_t)^j$$

for $j = 0, 1, 2$.

Local linear regression

In practice:

$$\hat{m}(x) := \frac{\sum_{t=1}^{n} w_t(x) y_t}{\sum_{t=1}^{n} w_t(x) + n^{-2}}.$$

Local linear regression

In practice:

$$\hat{m}(x) := \frac{\sum_{t=1}^{n} w_t(x) y_t}{\sum_{t=1}^{n} w_t(x) + n^{-2}}.$$

Extend to more complex autoregressive models like

$$X_t = m_1(X_{t-1}) + m_2(X_{t-2}) + \ldots + m_k(X_{t-k}) + Z_t,$$

where $(m_i)_{i=1}^{k}$ is a sequence of smooth functions, or

$$X_t = m(X_{t-1}, \ldots, X_{t-k}) + Z_t,$$

where $m : \mathbb{R}^k \to \mathbb{R}$ is a smooth function.

Local linear regression

In practice:

$$\hat{m}(x) := \frac{\sum_{t=1}^{n} w_t(x) y_t}{\sum_{t=1}^{n} w_t(x) + n^{-2}}.$$

Extend to more complex autoregressive models like

$$X_t = m_1(X_{t-1}) + m_2(X_{t-2}) + \ldots + m_k(X_{t-k}) + Z_t,$$

where $(m_i)_{i=1}^{k}$ is a sequence of smooth functions, or

$$X_t = m(X_{t-1}, \ldots, X_{t-k}) + Z_t,$$

where $m : \mathbb{R}^k \to \mathbb{R}$ is a smooth function.
Multivariate kernels $K_h : \mathbb{R}^k \to \mathbb{R}$, e.g., multivariate Gaussian density:

$$K_h(\mathbf{x}) = (2\pi h^2)^{-k/2} (\det \Sigma)^{-1/2} \exp(-(2h^2)^{-1} \mathbf{x}' \Sigma^{-1} \mathbf{x}).$$