ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC BÁCH KHOA

CHƯƠNG 6: PHÂN TÍCH DỮ LIỆU

CHUỖI THỜI GIAN

Khoa Công nghệ thông tin

Tài liệu tham khảo

[1] Rob J Hyndman and George Athanasopoulos, Forecasting: Principles & Practice, available online at https://otexts.com/fpp2/

[2] Slides used for the course offered at Monash University for ETC3550 during semester 1, 2018" (https://github.com/robjhyndman/ETC3550Slides/releases)

[3] https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html

[4] Các nguồn khác trên internet

Nội dung

6.1. Giới thiệu về dữ liệu chuỗi thời gian

6.2. Xử lý thông tin thời gian với Pandas

6.3. Phân rã dữ liệu chuỗi thời gian

6.4. Mô hình hoá dữ liệu chuỗi thời gian cho bài toán dự báo

6.1. Giới thiệu về dữ liệu chuỗi thời gian

Dữ liệu chuỗi thời gian (time series data) là gì

- Là một chuỗi các giá trị được xếp thứ tự theo thời gian
- Phổ biến trong các lĩnh vực tự nhiên – xã hội
- Các ví dụ:
  - Tỉ giá USD/EUR được ghi lại từng ngày qua nhiều năm

(x-rates.com)

Dữ liệu chuỗi thời gian (time series data) là gì

- Là một chuỗi các giá trị được xếp thứ tự theo thời gian
- Phổ biến trong các lĩnh vực tự nhiên – xã hội
- Các ví dụ:
  - Nhịp tim của bệnh nhân được máy đo ghi nhận theo từng phút

(researchgate.net)

Tầm quan trọng của dữ liệu chuỗi thời gian

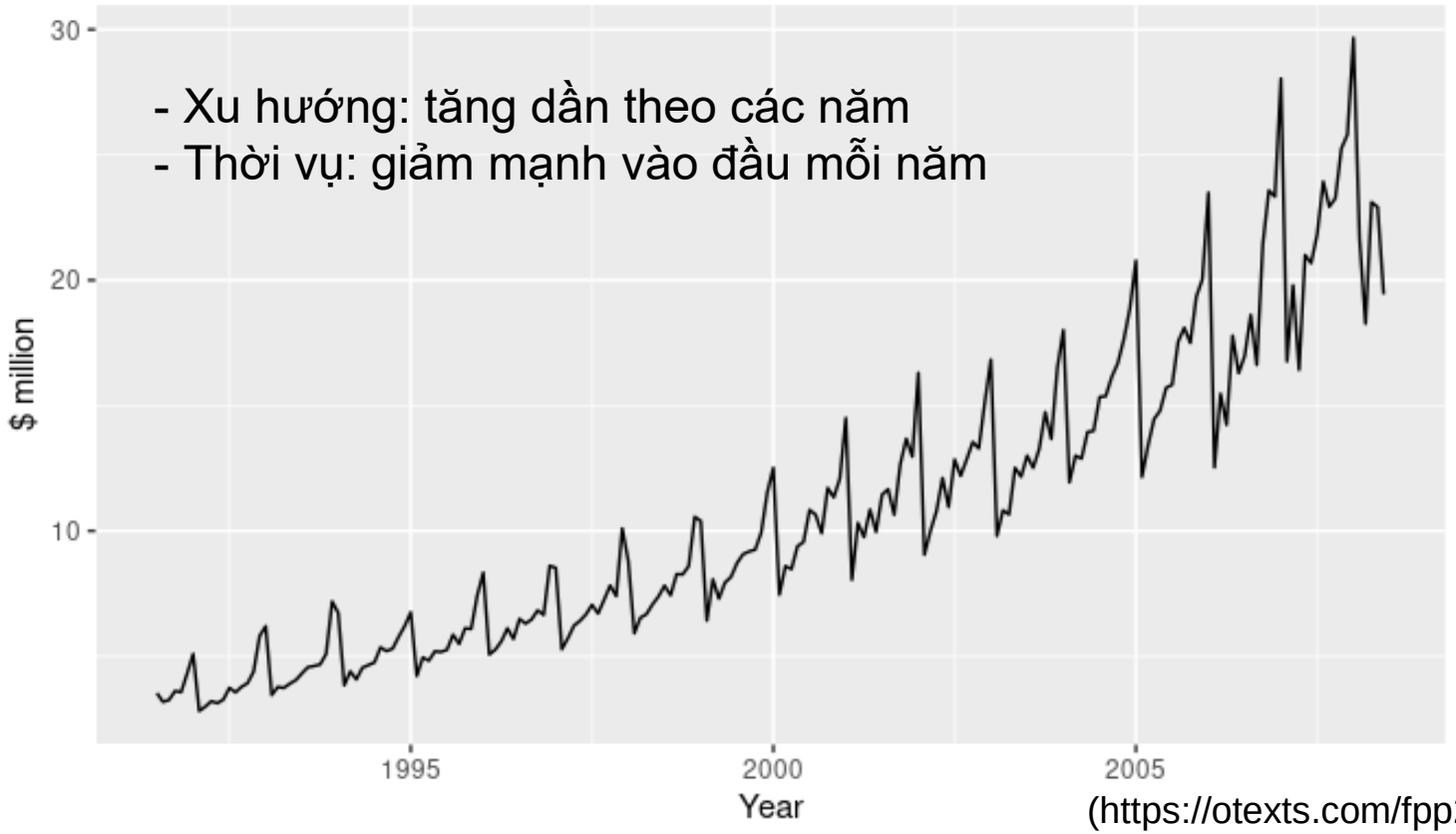
- Có đặc điểm là thường biến thiên theo chu kỳ và có xu hướng
- Ẩn bên dưới những biến thiên của dữ liệu là một số yếu tố cần phải phát hiện nhằm:
  - Hiểu về quá trình đã tạo ra dữ liệu quan sát
  - Dự báo dữ liệu tương lai (vd: tỉ giá)
  - Phân lớp dữ liệu (vd: phân loại nhịp tim bình thường/bất thường, hay theo trạng thái ngủ/suy nghĩ/vận động…)

Các thành phần của dữ liệu chuỗi thời gian

1. Trend (xu hướng): xảy ra khi dữ liệu tăng (hoặc giảm) trong dài hạn

2. Seasonal (thời vụ): xảy ra khi dữ liệu bị ảnh hưởng bởi các yếu tố thời vụ (ví dụ: thời điểm trong năm, ngày trong tuần). Thời vụ có tần suất cố định.

Ví dụ: Antidiabetic drug sales



(https://otexts.com/fpp2/tspatterns.html)

Các thành phần của dữ liệu chuỗi thời gian (tt)

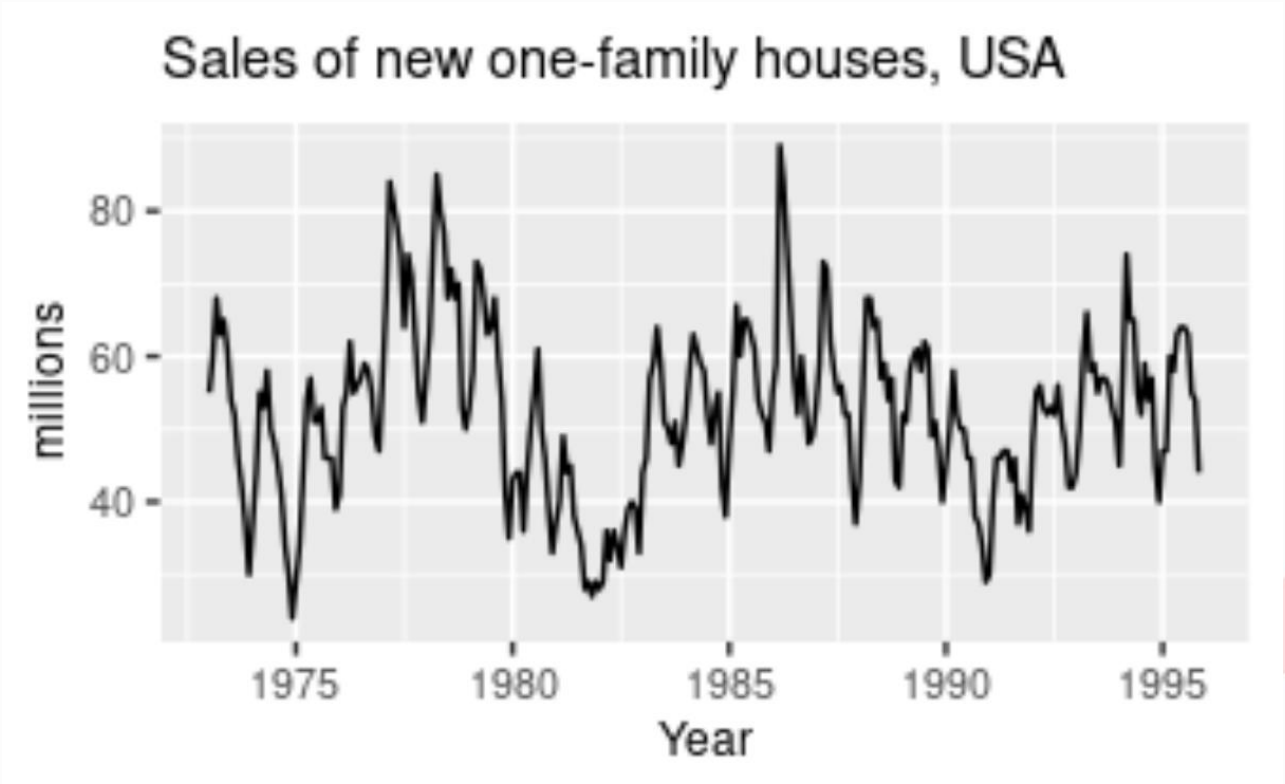
3. Cyclic (chu kỳ): xảy ra khi dữ liệu tăng và giảm không theo một tần suất cố định. Những biến động này thường do điều kiện kinh tế và thường liên quan đến "chu kỳ kinh doanh". Chu kỳ của những biến động này thường kéo dài ít nhất là 2 năm.

Chú ý:

- Nhiều chuỗi thời gian bao gồm các thành phần xu hướng, chu kỳ và thời vụ.
- Khi chọn một phương pháp dự báo, trước tiên cần xác định sự có mặt của các thành phần trên trong dữ liệu, sau đó chọn một phương pháp có thể nắm bắt các thành phần một cách chính xác.

(https://otexts.com/fpp2/tspatterns.html)

Các ví dụ

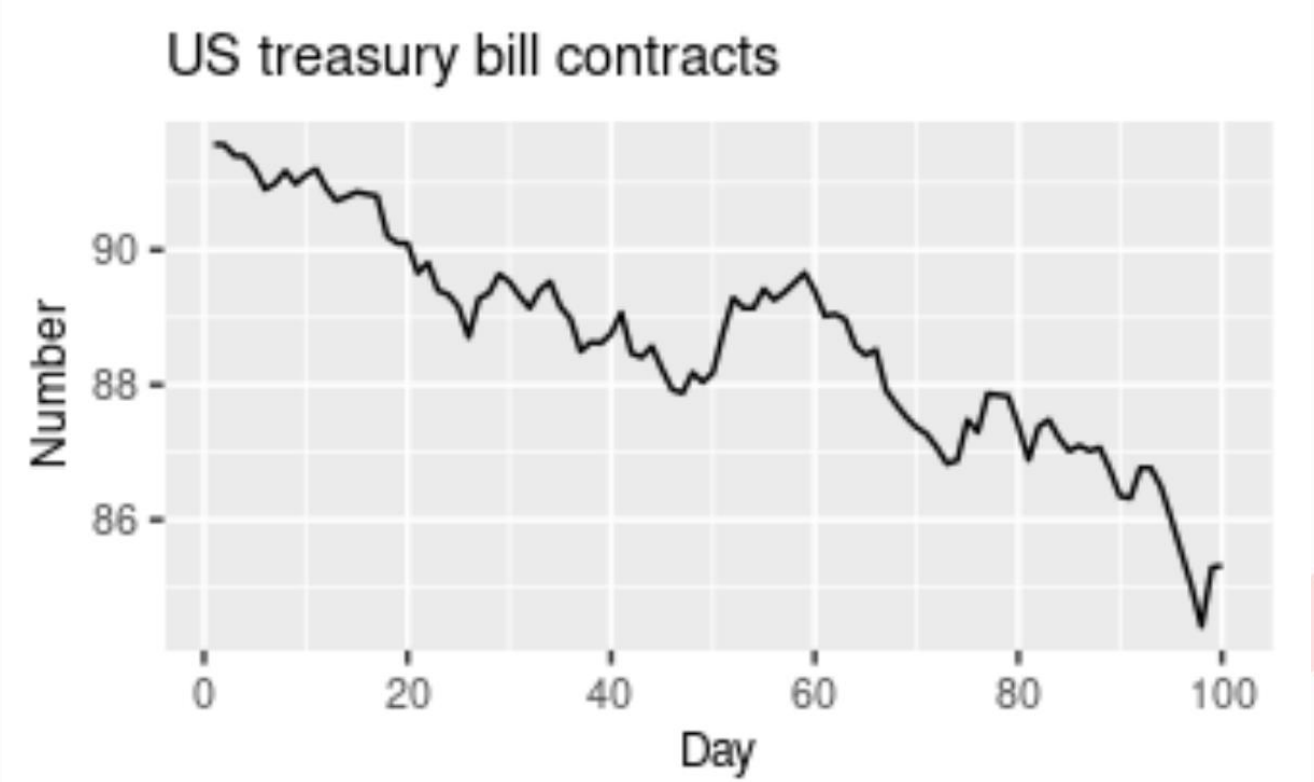


Doanh số bán nhà hàng tháng cho thấy:

- Tính thời vụ mạnh trong mỗi năm
- Một số biến thiên mang tính chu kỳ mạnh với khoảng thời gian khoảng 6–10 năm
- Không có xu hướng rõ ràng trong dữ liệu trong giai đoạn này

(https://otexts.com/fpp2/tspatterns.html)

Các ví dụ (tt)

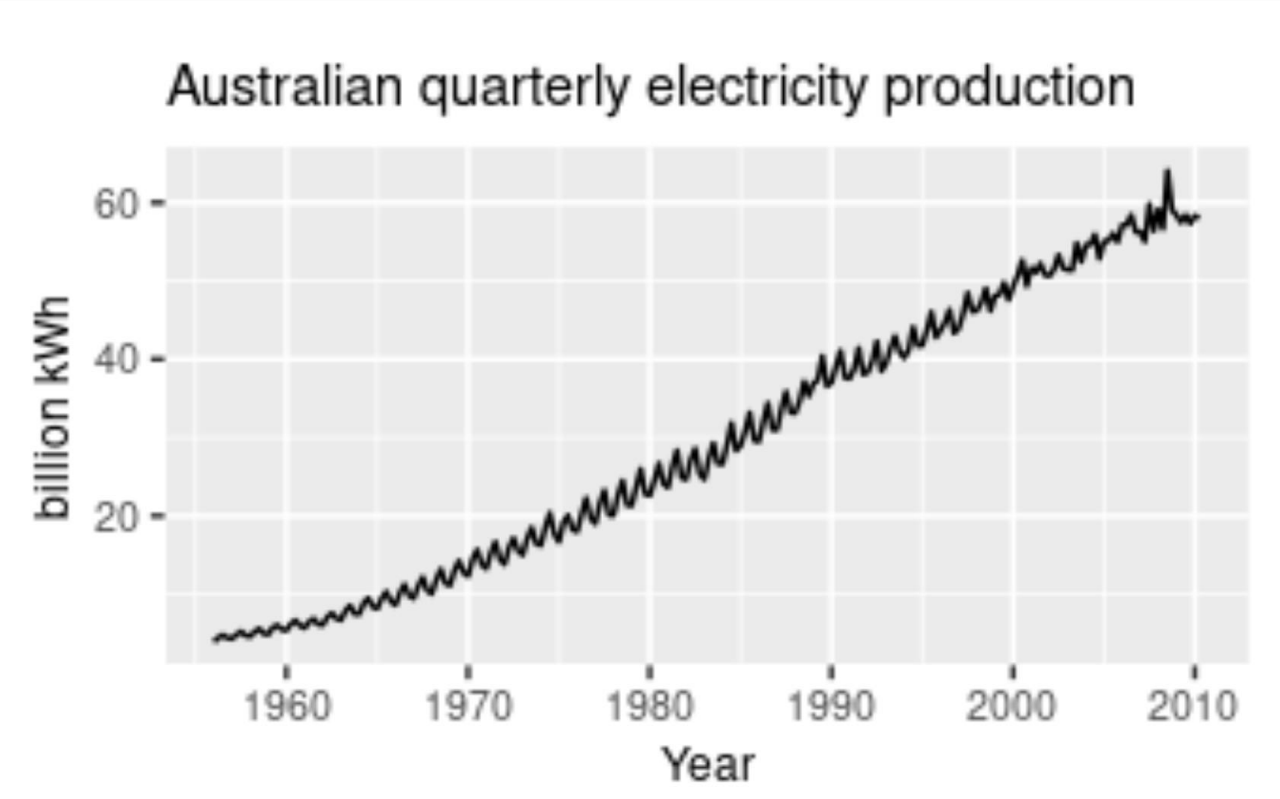


Số lượng hợp đồng tín phiếu kho bạc Hoa Kỳ trong 100 ngày giao dịch liên tiếp vào năm 1981 cho thấy:

- Không có tính thời vụ
- Xu hướng giảm rõ ràng
- Nếu chúng ta quan sát một chuỗi dài hơn thì xu hướng giảm này có thể là một phần của chu kỳ dài

(https://otexts.com/fpp2/tspatterns.html)

Các ví dụ (tt)

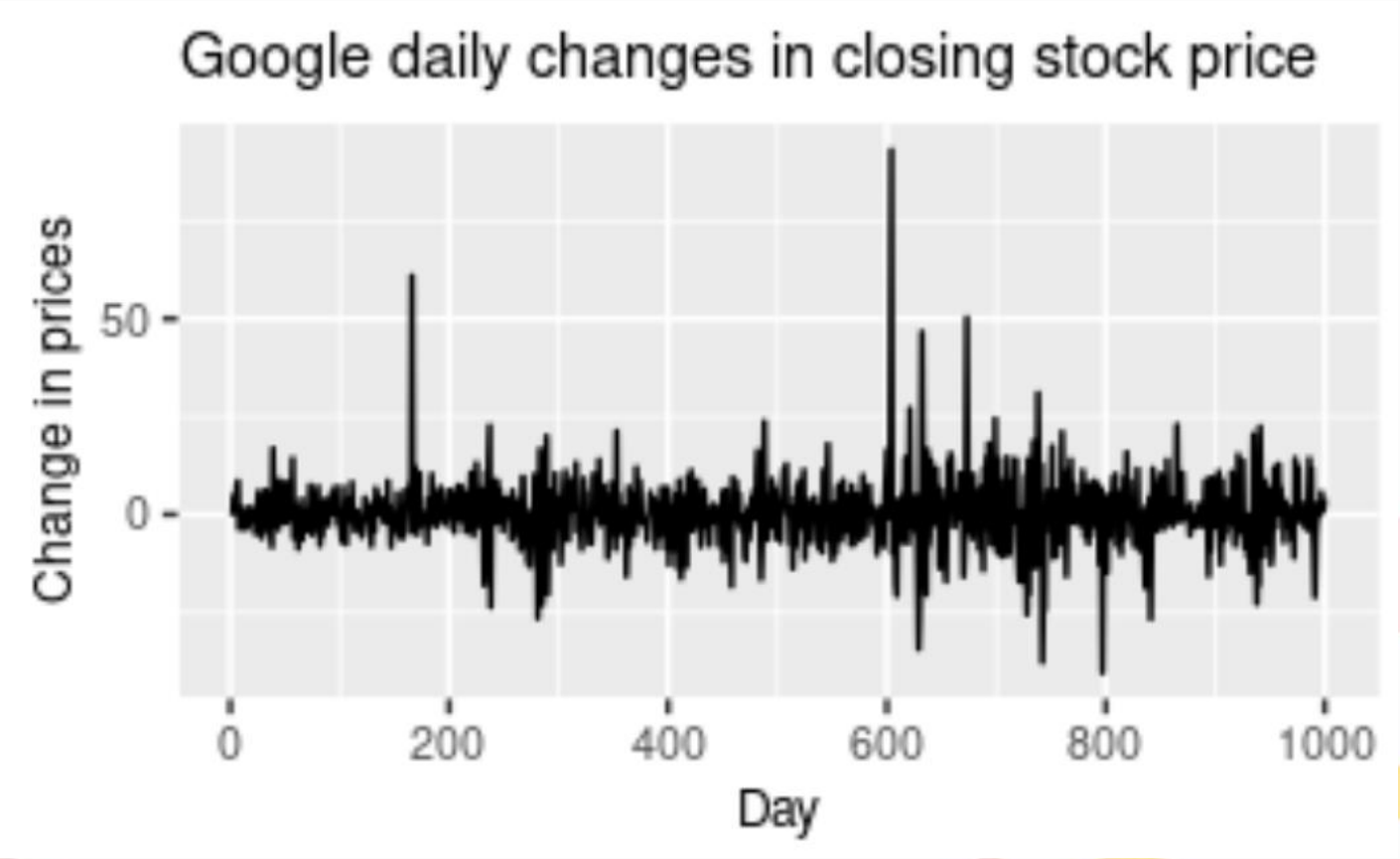


Sản lượng điện hàng quý của Úc cho thấy:

- Xu hướng tăng mạnh
- Tính thời vụ mạnh (theo từng năm)
- Không có tính chu kỳ

(https://otexts.com/fpp2/tspatterns.html)

Các ví dụ (tt)



Sự thay đổi hàng ngày trong giá cổ phiếu đóng cửa của Google:

- Không có xu hướng, tính thời vụ và tính chu kỳ
- Có những biến động ngẫu nhiên dường như không thể dự đoán được

(https://otexts.com/fpp2/tspatterns.html)

6.2. Xử lý thông tin thời gian với Pandas

Các tính năng của Pandas

- Parsing time series information from various sources and formats
- Generate sequences of fixed-frequency dates and time spans
- Manipulating and converting date times with timezone information
- Resampling or converting a time series to a particular frequency
- Performing date and time arithmetic with absolute or relative time increments

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html)

Các khái niệm và kiểu dữ liệu thời gian

Pandas có 4 khái niệm liên quan đến thời gian:

- Date times: A specific date and time with timezone support. Similar to datetime.datetime from the standard library.
- Time deltas: An absolute time duration. Similar to datetime.timedelta from the standard library.
- Time spans: A span of time defined by a point in time and its associated frequency.
- Date offsets: A relative time duration that respects calendar arithmetic. Similar to dateutil.relativedelta.relativedelta from the dateutil package.

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html)

Các khái niệm và kiểu dữ liệu thời gian (tt)

- Bảng sau thể hiện các lớp, kiểu dữ liệu và cách tạo với mỗi khái niệm

Concept	Scalar Class	Array Class	pandas Data Type	Primary Creation Method
Date times	`Timestamp`	`DatetimeIndex`	`datetime64[ns]` or `datetime64[ns, tz]`	`to_datetime` or `date_range`
Time deltas	`Timedelta`	`TimedeltaIndex`	`timedelta64[ns]`	`to_timedelta` or `timedelta_range`
Time spans	`Period`	`PeriodIndex`	`period[freq]`	`Period` or `period_range`
Date offsets	`DateOffset`	`None`	`None`	`DateOffset`

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html)

Chuyển thành nhãn thời gian (timestamps)

To convert a **Series** or list-like object of date-like objects e.g. strings, epochs, or a mixture, you can use the `to_datetime` function. When passed a `Series`, this returns a `Series` (with the same index), while a list-like is converted to a `DatetimeIndex`:

```
In [43]: pd.to_datetime(pd.Series(["Jul 31, 2009", "2010-01-10", None]))
Out[43]:
0   2009-07-31
1   2010-01-10
2          NaT
dtype: datetime64[ns]

In [44]: pd.to_datetime(["2005/11/23", "2010.12.31"])
Out[44]: DatetimeIndex(['2005-11-23', '2010-12-31'], dtype='datetime64[ns]', freq=None)
```

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html)

Lớp DatetimeIndex

One of the main uses for `DatetimeIndex` is as an index for pandas objects. The `DatetimeIndex` class contains many time series related optimizations:

- A large range of dates for various offsets are pre-computed and cached under the hood in order to make generating subsequent date ranges very fast (just have to grab a slice).
- Fast shifting using the `shift` method on pandas objects.
- Unioning of overlapping `DatetimeIndex` objects with the same frequency is very fast (important for fast data alignment).
- Quick access to date fields via properties such as `year`, `month`, etc.
- Regularization functions like `snap` and very fast `asof` logic.

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html)

Các thuộc tính time/date

There are several time/date properties that one can access from `Timestamp` or a collection of timestamps like a `DateTimeIndex`.

Property	Description
year	The year of the datetime
month	The month of the datetime
day	The days of the datetime
hour	The hour of the datetime
minute	The minutes of the datetime
second	The seconds of the datetime
microsecond	The microseconds of the datetime
nanosecond	The nanoseconds of the datetime
date	Returns datetime.date
time	Returns datetime.time
dayofyear	The ordinal day of year
weekofyear	The week ordinal of the year
week	The week ordinal of the year
dayofweek	The numer of the day of the week with Monday=0, Sunday=6
weekday	The number of the day of the week with Monday=0, Sunday=6
weekday_name	The name of the day in a week (ex: Friday)
quarter	Quarter of the date: Jan=Mar = 1, Apr-Jun = 2, etc.
days_in_month	The number of days in the month of the datetime
is_month_start	Logical indicating if first day of month (defined by frequency)
is_month_end	Logical indicating if last day of month (defined by frequency)
is_quarter_start	Logical indicating if first day of quarter (defined by frequency)
is_quarter_end	Logical indicating if last day of quarter (defined by frequency)
is_year_start	Logical indicating if first day of year (defined by frequency)
is_year_end	Logical indicating if last day of year (defined by frequency)

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html)

Offset aliases

- A number of string aliases are given to useful common time series frequencies. We will refer to these aliases as *offset aliases*.

Alias	Description
B	business day frequency
C	custom business day frequency (experimental)
D	calendar day frequency
W	weekly frequency
M	month end frequency
BM	business month end frequency
CBM	custom business month end frequency
MS	month start frequency
BMS	business month start frequency
CBMS	custom business month start frequency
Q	quarter end frequency
BQ	business quarter endfrequency
QS	quarter start frequency
BQS	business quarter start frequency
A	year end frequency
BA	business year end frequency
AS	year start frequency
BAS	business year start frequency
BH	business hour frequency
H	hourly frequency
T, min	minutely frequency
S	secondly frequency
L, ms	milliseconds
U, us	microseconds
N	nanoseconds

DateOffset objects

- Most DateOffsets have associated frequencies strings, or offset aliases, that can be passed into **freq** keyword arguments

Date Offset	Frequency String	Description
`DateOffset`	None	Generic offset class, defaults to absolute 24 hours
`BDay` or `BusinessDay`	`'B'`	business day (weekday)
`CDay` or `CustomBusinessDay`	`'C'`	custom business day
`Week`	`'W'`	one week, optionally anchored on a day of the week
`WeekOfMonth`	`'WOM'`	the x-th day of the y-th week of each month
`LastWeekOfMonth`	`'LWOM'`	the x-th day of the last week of each month
`MonthEnd`	`'M'`	calendar month end
`MonthBegin`	`'MS'`	calendar month begin
`BMonthEnd` or `BusinessMonthEnd`	`'BM'`	business month end

Thay đổi tần số

pandas.Series.asfreq

`Series.asfreq(freq, method=None, how=None, normalize=False, fill_value=None)` [source]

Convert TimeSeries to specified frequency.

Optionally provide filling method to pad/backfill missing values.

Returns the original data conformed to a new index with the specified frequency. `resample` is more appropriate if an operation, such as summarization, is necessary to represent the data at the new frequency.

Parameters: **freq** : *DateOffset or str*

Frequency DateOffset or string.

Returns: **Same type as caller**

Object converted to the specified frequency.

(https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Series.asfreq.html#pandas.Series.asfreq)

Thay đổi tần số (tt)

Ví dụ:

```
In [281]: dr = pd.date_range("1/1/2010", periods=3, freq=3 * pd.offsets.BDay())

In [282]: ts = pd.Series(np.random.randn(3), index=dr)

In [283]: ts
Out[283]:
2010-01-01    1.494522
2010-01-06   -0.778425
2010-01-11   -0.253355
Freq: 3B, dtype: float64

In [284]: ts.asfreq(pd.offsets.BDay())
Out[284]:
2010-01-01    1.494522
2010-01-04         NaN
2010-01-05         NaN
2010-01-06   -0.778425
2010-01-07         NaN
2010-01-08         NaN
2010-01-11   -0.253355
Freq: B, dtype: float64
```

(https://pandas.pydata.org/pandas-docs/stable/user_guide/timeseries.html#frequency-conversion)

6.3. Phân rã dữ liệu chuỗi thời gian (Time series decomposition)

Giới thiệu

- Dữ liệu CTG thể hiện nhiều dạng thức (pattern) khác nhau
- Việc phân rã CTG thành nhiều thành phần, mỗi thành phần biểu diễn một loại dạng thức, *giúp hiểu hơn về CTG và có thể cải thiện độ chính xác dự báo*
- Dữ liệu CTG có thể gồm các thành phần: xu hướng, chu kỳ và thời vụ
- Chúng ta thường kết hợp xu hướng và chu kỳ thành một thành phần chung xu hướng-chu kỳ (đôi khi gọi ngắn gọn là xu hướng)

→ phân rã CTG thành 3 thành phần: xu hướng-chu kỳ, thời vụ, và phần dư (chứa những gì còn lại sau khi loại đi 2 thành phần kia)

Các mô hình phân rã

- Phân rã cộng (additive decomposition): $y_t = S_t + T_t + R_t$
- Phân rã nhân (multiplicative decomposition): $y_t = S_t \times T_t \times R_t$

với y_t: dữ liệu CTG quan sát được

S_t: thành phần thời vụ (Seasonal)

T_t: thành phần xu hướng-chu kỳ (Trend-cycle)

R_t: thành phần dư (Remainder/Residual)

- **Khi nào dùng mô hình nào?**
  - PR cộng: khi biên độ của các dao động có tính thời vụ không biến thiên theo mức (level) của CTG
  - PR nhân: khi biên độ của các dao động có tính thời vụ tỉ lệ với mức của CTG
  - Một cách khác thay vì dùng PR nhân là: biến đổi dữ liệu (vd: lấy logarit) sao cho các dao động ổn định theo thời gian, sau đó dùng PR cộng vì

$$y_t = S_t \times T_t \times R_t \quad \text{is equivalent to} \quad \log y_t = \log S_t + \log T_t + \log R_t.$$

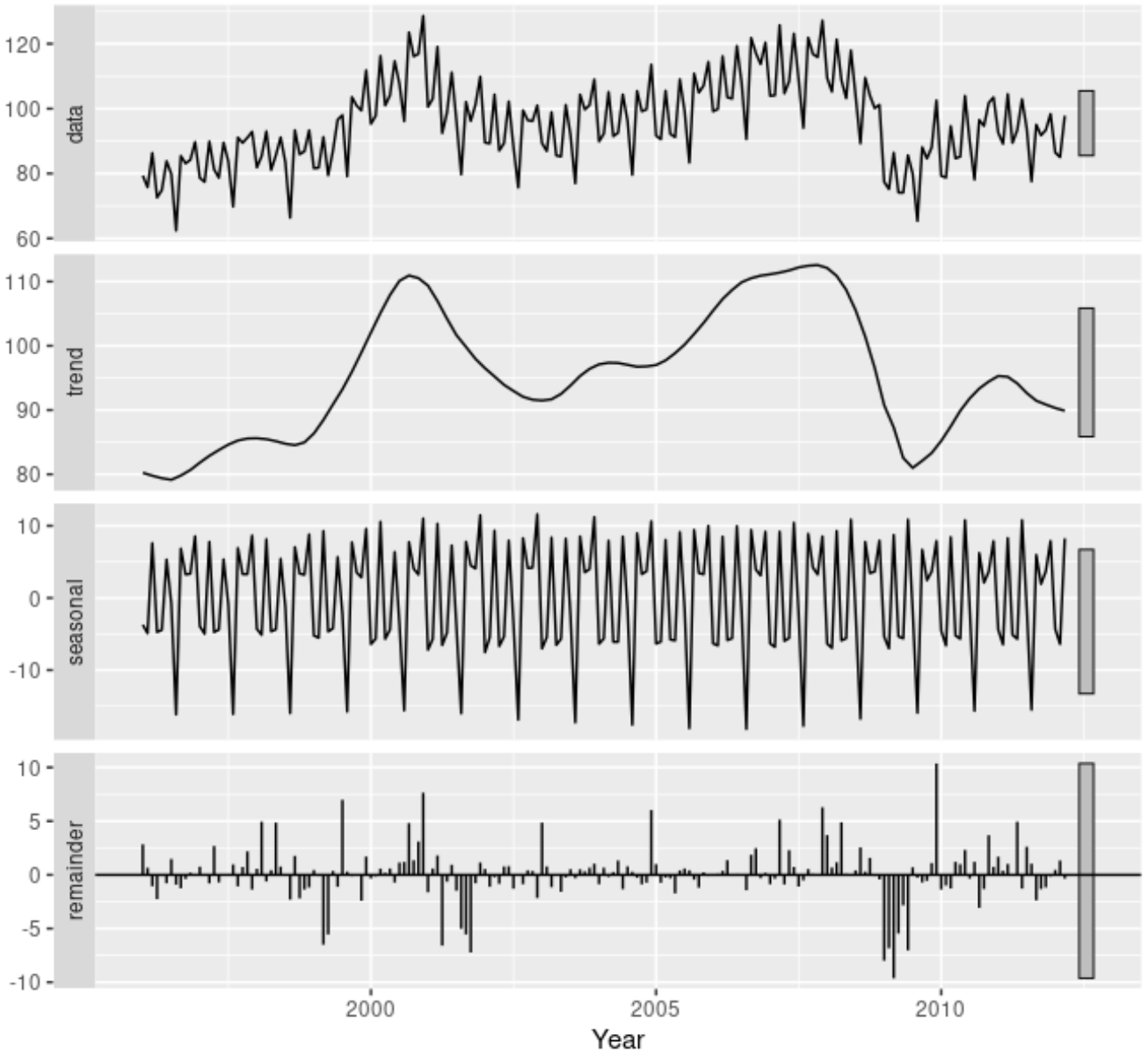
Ví dụ phân rã cộng

Dữ liệu CTG

TP xu hướng-chu kỳ

TP thời vụ

TP dư



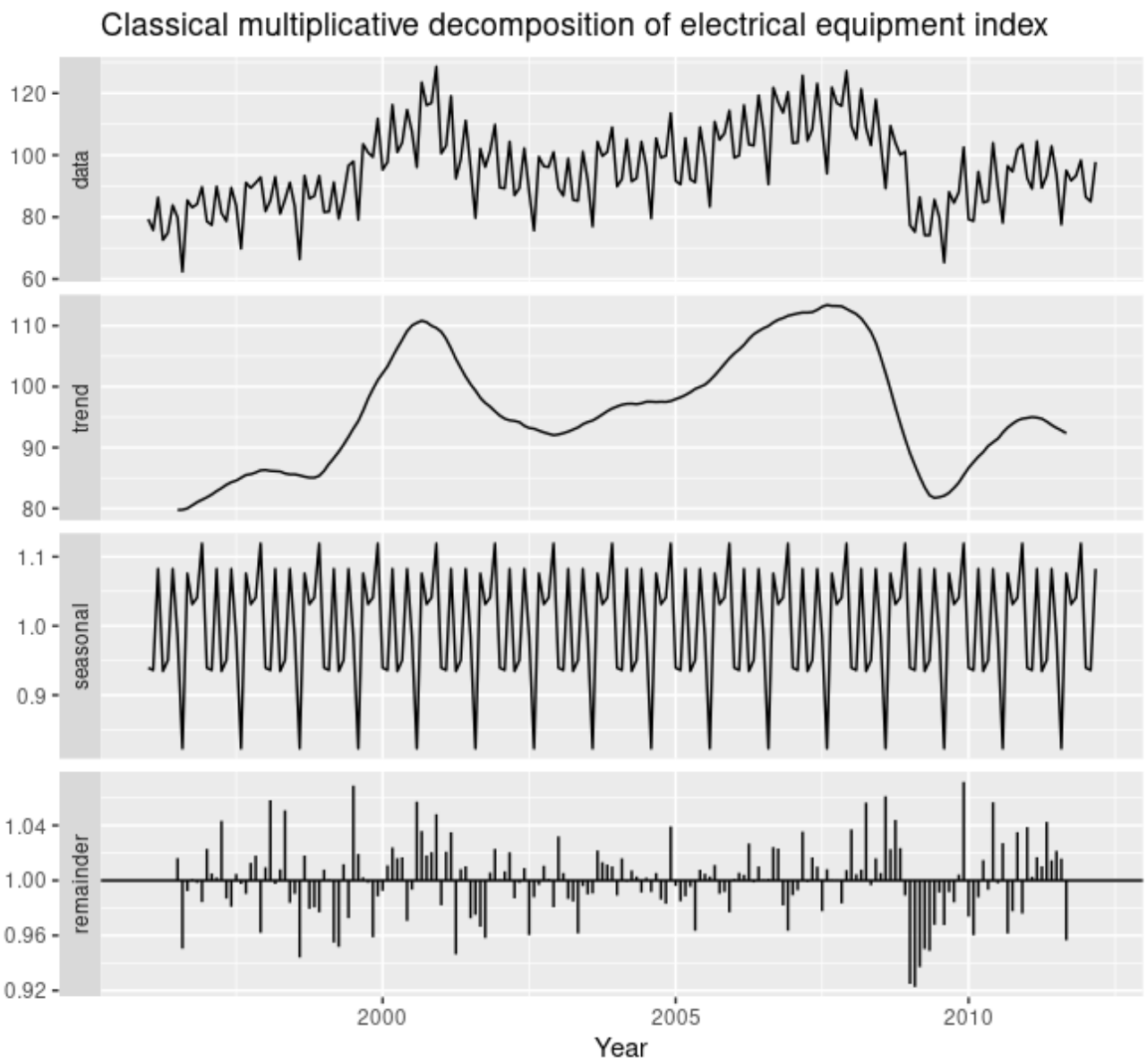
Ví dụ phân rã nhân

Dữ liệu CTG

TP xu hướng-chu kỳ

TP thời vụ

TP dư



Các phương pháp phân rã

04 phương pháp được mô tả chi tiết trong [1] (Chapter 6):

- Cổ điển
- X11
- SEATS (Seasonal Extraction in ARIMA Time Series)
- STL (Seasonal and Trend decomposition using Loess)

Các phương pháp phân rã (tt)

- **STL has several advantages** over the classical, SEATS and X11 decomposition methods:
  - Unlike SEATS and X11, STL will handle any type of seasonality, not only monthly and quarterly data.
  - The seasonal component is allowed to change over time, and the rate of change can be controlled by the user.
  - The smoothness of the trend-cycle can also be controlled by the user.
  - It can be robust to outliers (i.e., the user can specify a robust decomposition), so that occasional unusual observations will not affect the estimates of the trend-cycle and seasonal components. They will, however, affect the remainder component.
- On the other hand, STL has some disadvantages. In particular, it does not handle trading day or calendar variation automatically, and **it only provides facilities for additive decompositions.**

(https://otexts.com/fpp2/stl.html)

Các phương pháp phân rã (tt)

Các ví dụ cài đặt bằng Python dùng thư viện **statsmodels**:

- PP cổ điển:

https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/

- PP STL:

https://www.statsmodels.org/stable/examples/notebooks/generated/stl_decomposition.html

https://towardsdatascience.com/stl-decomposition-how-to-do-it-from-scratch-b686711986ec

Tiên đoán dựa trên phân rã

- Các kết quả phân rã chủ yếu hữu ích để hiểu hơn về dữ liệu CTG và khám phá các biến thiên có tính lịch sử của dữ liệu
- Tuy nhiên, ta có thể dùng kết quả phân rã để tiên đoán dữ liệu tương lai
- CTG có thể viết thành: $y_t = \hat{S}_t + \hat{A}_t$ với $\hat{A}_t = \hat{T}_t + \hat{R}_t$ (PR cộng)

  hoặc: $y_t = \hat{S}_t \hat{A}_t$ với $\hat{A}_t = \hat{T}_t \hat{R}_t$ (PR nhân)

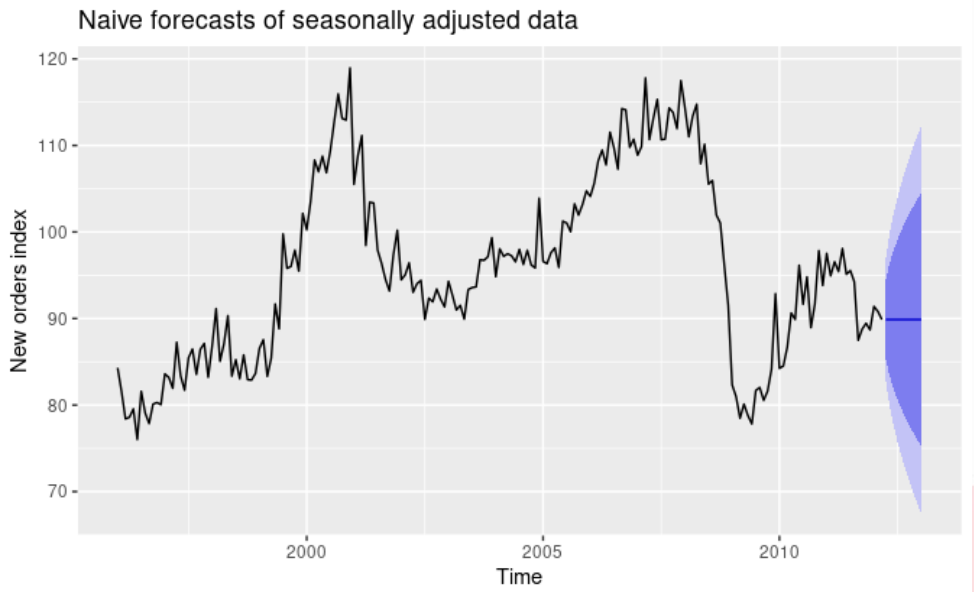
→Cần tiên đoán TP thời vụ $\hat{S}_t$ và TP đã hiệu chỉnh thời vụ $\hat{A}_t$ riêng biệt

- **TP thời vụ** được giả định là không đổi (hoặc thay đổi rất chậm) → được tiên đoán bằng cách lặp lại TP thời vụ của năm trước (seasonal naïve method)
- **TP đã hiệu chỉnh thời vụ** được tiên đoán bằng một trong các PP tiên đoán phi thời vụ (non-seasonal forecasting method) sẽ học hoặc bằng PP "ngây thơ" (naïve) (lặp lại dữ liệu cuối cùng)

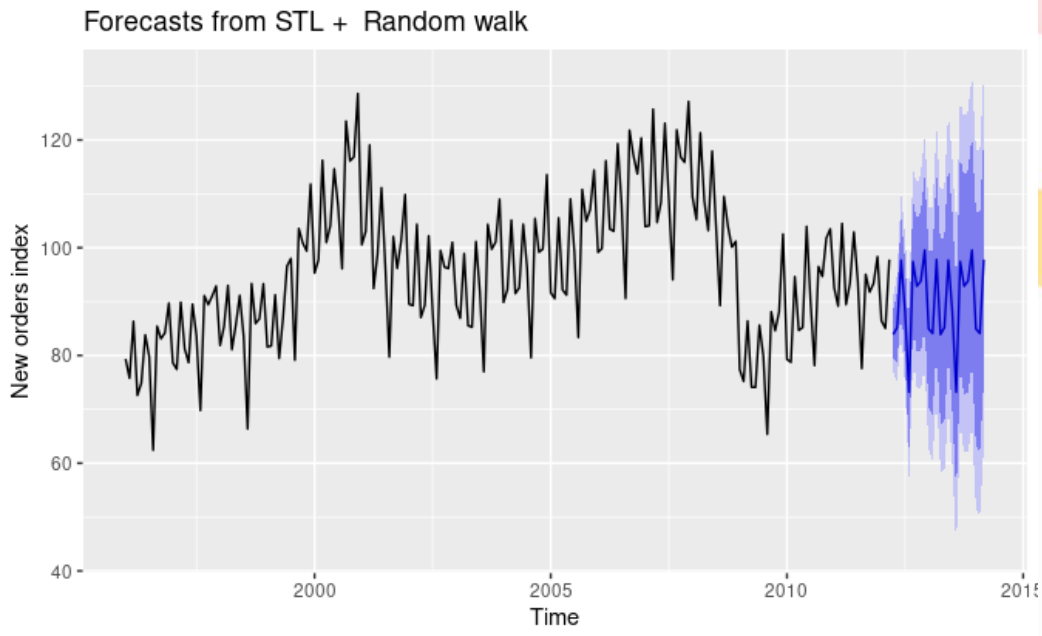
Tiên đoán dựa trên phân rã (tt)

- Ví dụ:

TP đã hiệu chỉnh thời vụ được dự báo bằng PP ”ngây thơ” (đoạn nằm ngang màu xanh)



CTG tương lai (đường biến thiên màu xanh) được dự báo bằng cách cộng thêm TP thời vụ (cũng được dự báo bằng PP “ngây thơ”)



6.4. Mô hình hoá dữ liệu chuỗi thời gian cho bài toán dự báo

Nội dung

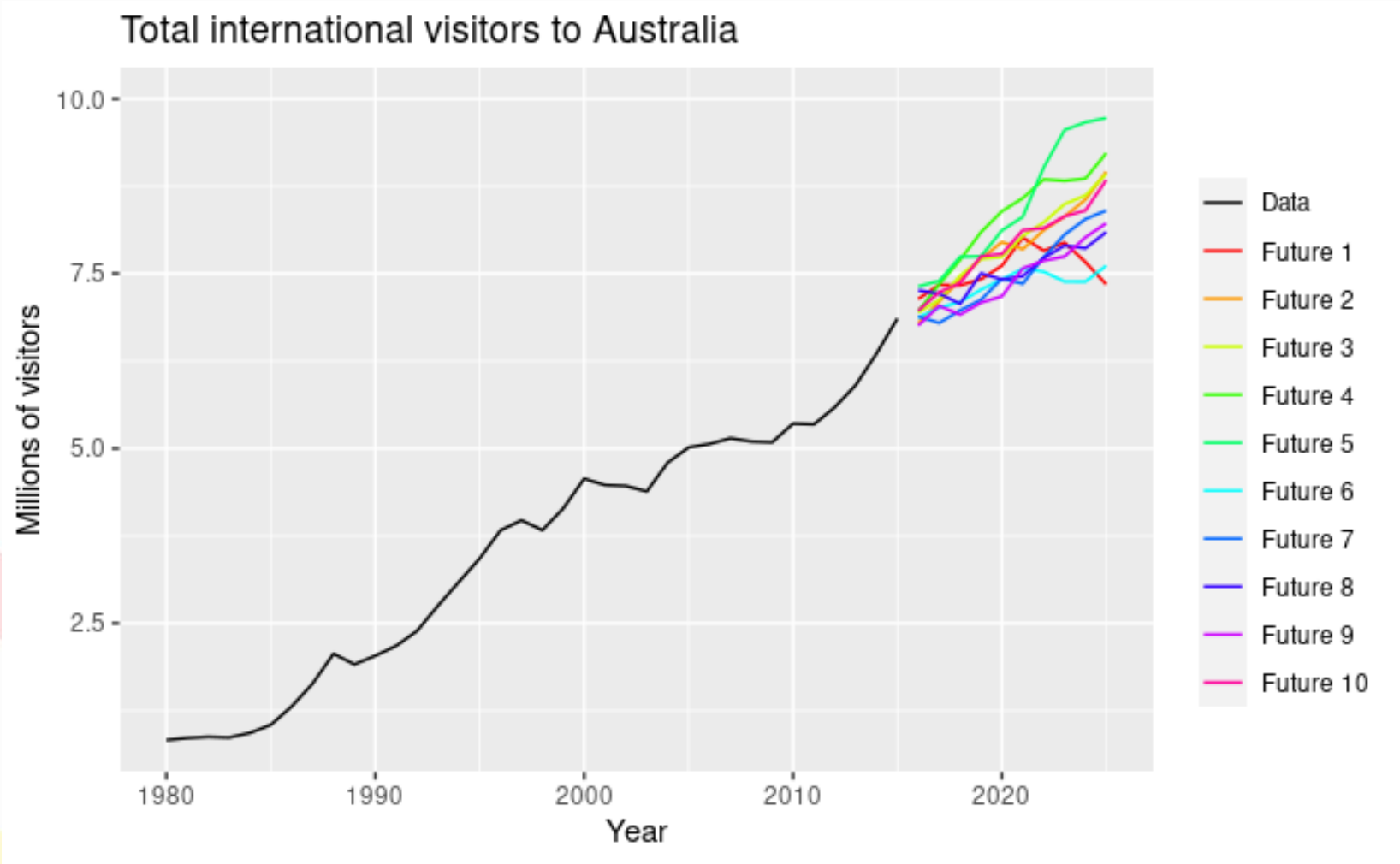
- 6.4.1. Giới thiệu chung
- 6.4.2. Các PP làm trơn mũ (Exponential smoothing)
- 6.4.3. Các mô hình ARIMA (Auto Regressive Integrated Moving Average)

6.4.1. Giới thiệu chung

- Quan điểm thống kê về dự báo
- Xây dựng và đánh giá mô hình dự báo

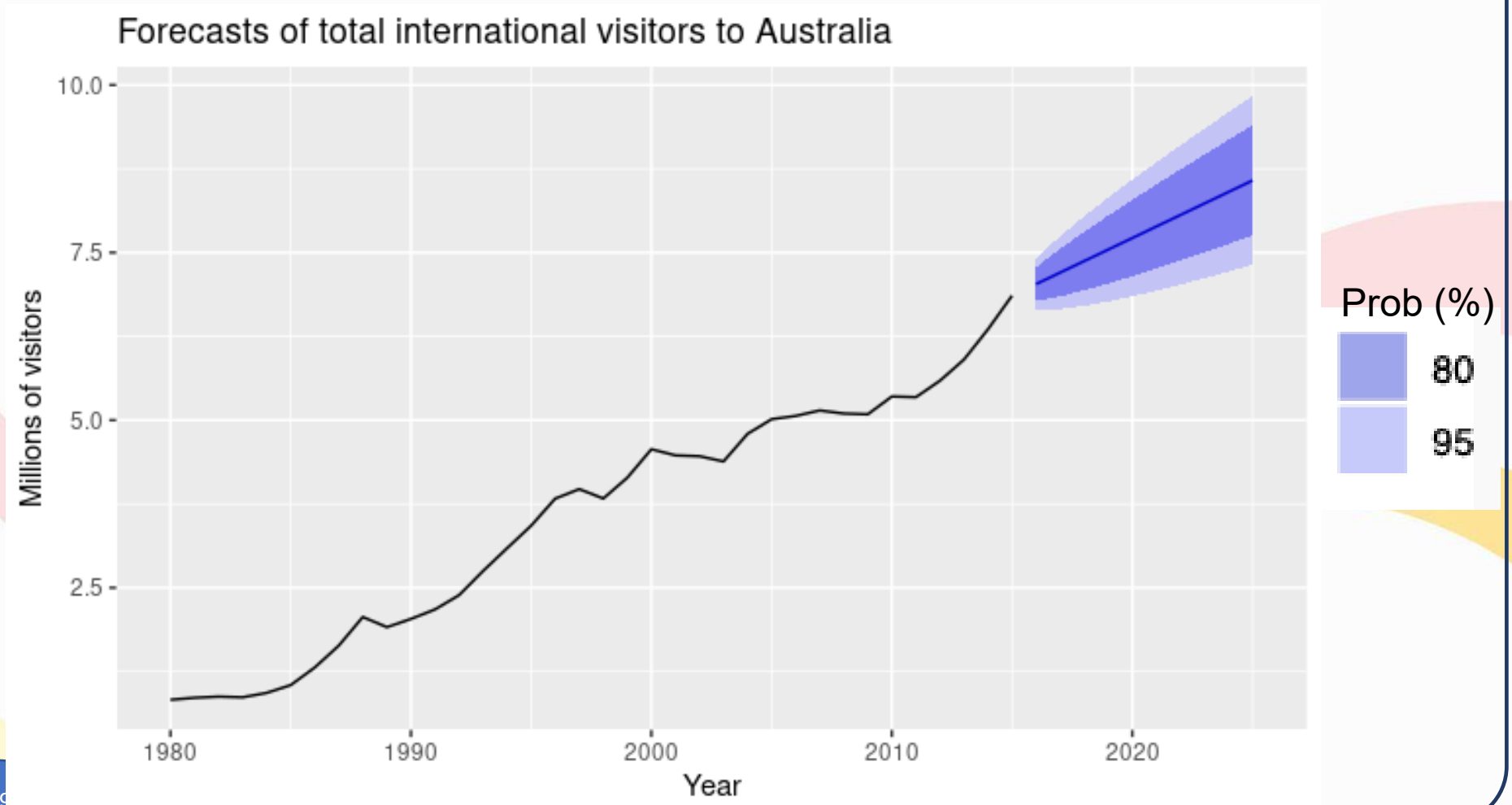
Quan điểm thống kê về dự báo

- Thời điểm dự báo càng xa hiện tại thì giá trị dự báo càng kém chắc chắn
- Ví dụ về point forecasts:



Quan điểm thống kê về dự báo (tt)

- Khi nhận được 1 giá trị dự báo, ta đang ước tính điểm chính giữa của dải các giá trị mà biến ngẫu nhiên có thể nhận
- Ví dụ về interval forecasts:



Quan điểm thống kê về dự báo (tt)

- Thing to be forecast: a random variable, y_t.
- Forecast distribution: If $\mathcal{I}$ is all observations, then $y_t|\mathcal{I}$ means "the random variable y_t given what we know in $\mathcal{I}$".
- The "point forecast" is the mean (or median) of $y_t|\mathcal{I}$
- The "forecast variance" is var$[y_t|\mathcal{I}]$
- A prediction interval or "interval forecast" is a range of values of y_t with high probability.
- With time series, $\hat{y}_{t|t-1} = \hat{y}_t|\{y_1, y_2, \ldots, y_{t-1}\}$.
- $\hat{y}_{T+h|T} = \mathrm{E}[y_{T+h}|y_1, \ldots, y_T]$ (an h-step forecast taking account of all observations up to time T).

6.4.1. Giới thiệu chung

- Quan điểm thống kê về dự báo
- Xây dựng và đánh giá mô hình dự báo

Xây dựng và đánh giá mô hình dự báo

- **Dữ liệu CTG được phân thành DL huấn luyện và DL kiểm thử**

- DL huấn luyện dùng để tính các tham số của mô hình dự báo (model fitting)
- DL kiểm thử dùng để tính độ chính xác dự báo (model forecasting)
- DL kiểm thử thường chiếm 20% tổng số mẫu, và còn phụ thuộc vào ta muốn dự báo trước bao xa (giá trị *h*)
- **Một số lưu ý:**
  - Một mô hình khớp (fit) với DL huấn luyện không nhất thiết có thể dự báo chính xác trên DL kiểm thử
  - Một mô hình có số lượng tham số đủ nhiều có thể khớp hoàn toàn vào DL
  - Mô hình quá khớp (over-fitting) vào DL sẽ không thể nắm bắt được quy luật mang tính hệ thống trong DL

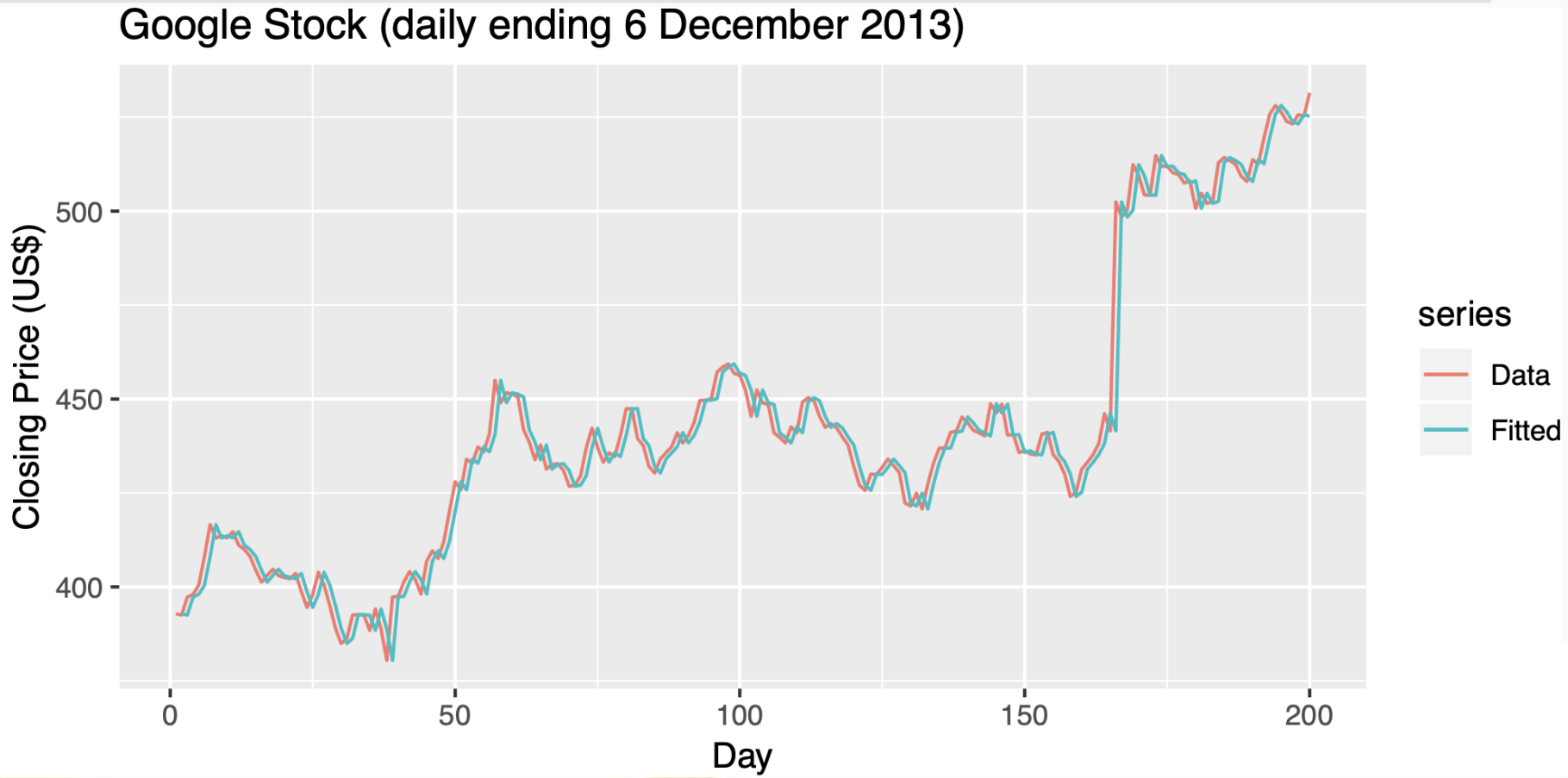
Lỗi huấn luyện

- Còn gọi là phần dư (residual)
- Là độ chênh lệch giữa giá trị quan sát trong DL huấn luyện và giá trị được khớp (fitted value) tương ứng: $e_t = y_t - \hat{y}_{t|t-1}$
- Các tính chất của phần dư mà một mô hình tốt cần thoả mãn:
  - $\{e_t\}$ không tương quan. Ngược lại thì trong phần dư vẫn còn chứa thông tin hữu ích dùng cho việc dự báo.
  - $\{e_t\}$ có giá trị trung bình bằng 0. Ngược lại thì các dự báo bị lệch (biased).
- Các tính chất này kiểm chứng việc mô hình đã "tận dụng" hết mọi thông tin trong dữ liệu chưa (nhưng không phải là cách tốt để lựa chọn mô hình)
- Phần dư lý tưởng có các tính chất thống kê giống nhiễu trắng (white noise)
- Có các test để kiểm tra phần dư như Box-Pierce và Ljung-Box

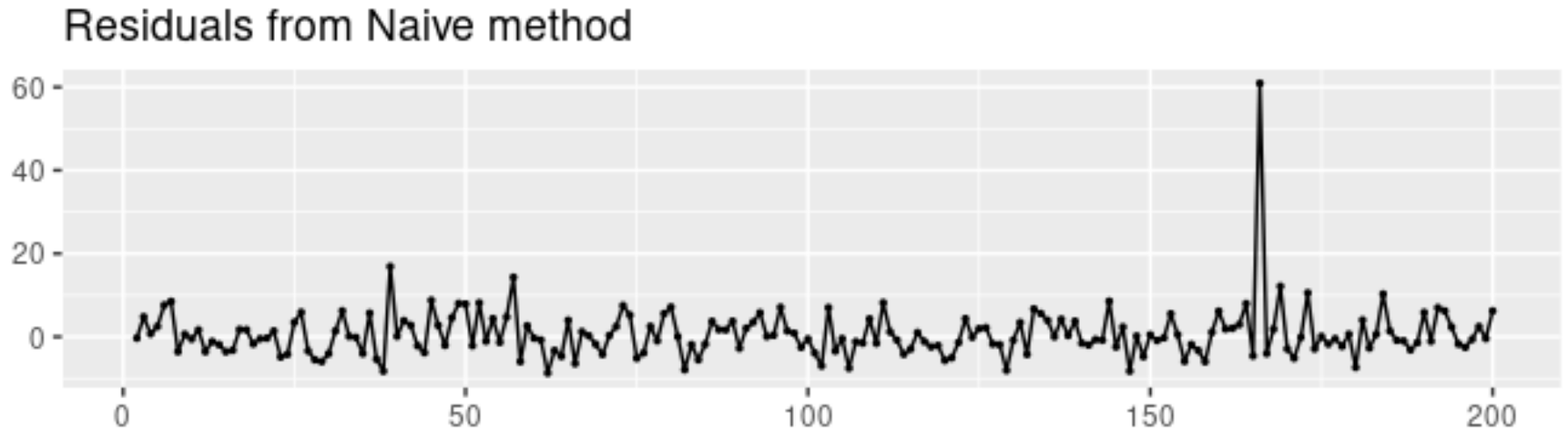
Ví dụ kiểm chứng lỗi huấn luyện

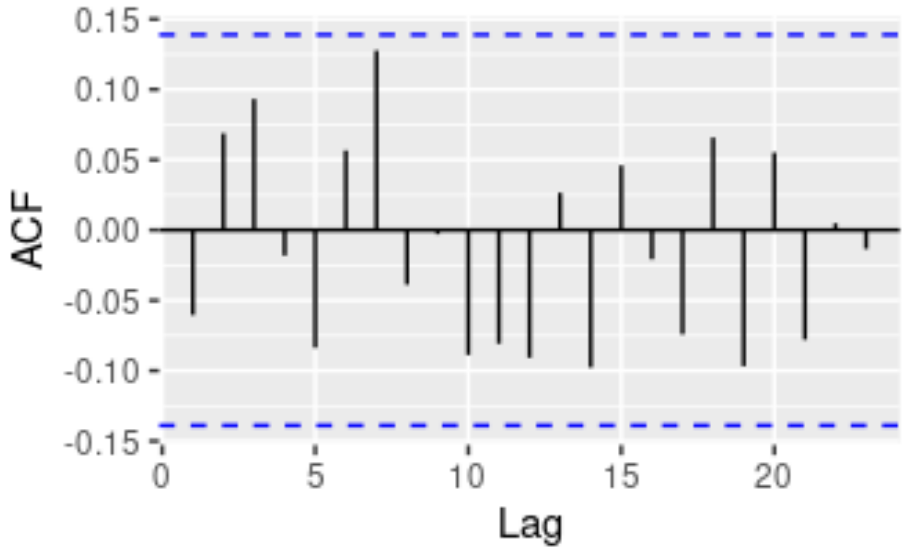
- Dự đoán giá cổ phiếu của Google dùng PP ngây thơ (Naïve):

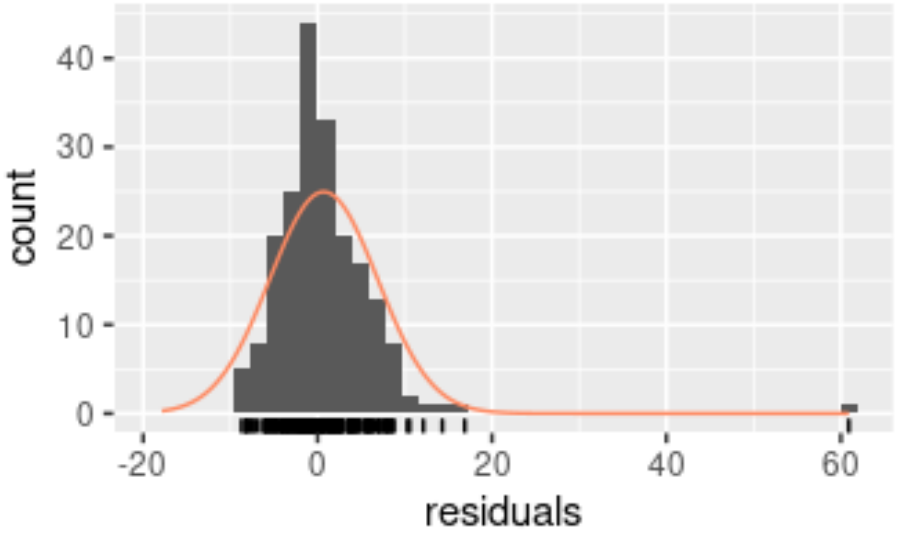
$$\hat{y}_{t|t-1} = y_{t-1}$$



Ví dụ kiểm chứng lỗi huấn luyện (tt)







(ACF: autocorrelation function)

(https://otexts.com/fpp2/residuals.html)

Lỗi dự báo

- Là độ chênh lệch giữa giá trị quan sát và giá trị được dự báo

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T},$$

với DL huấn luyện là $\{y_1,\ldots,y_T\}$ và DL kiểm thử là $\{y_{T+1},y_{T+2},\ldots\}$

  - Chỉ tính trên DL kiểm thử
  - Có thể gồm dự báo nhiều bước (multi-step forecasts)

Đo độ chính xác dự báo

y_{T+h} = $(T+h)$th observation, $h = 1, \ldots, H$

$\hat{y}_{T+h|T}$ = its forecast based on data up to time T.

e_{T+h} = $y_{T+h} - \hat{y}_{T+h|T}$

$$\text{MAE} = \text{mean}(|e_{T+h}|)$$

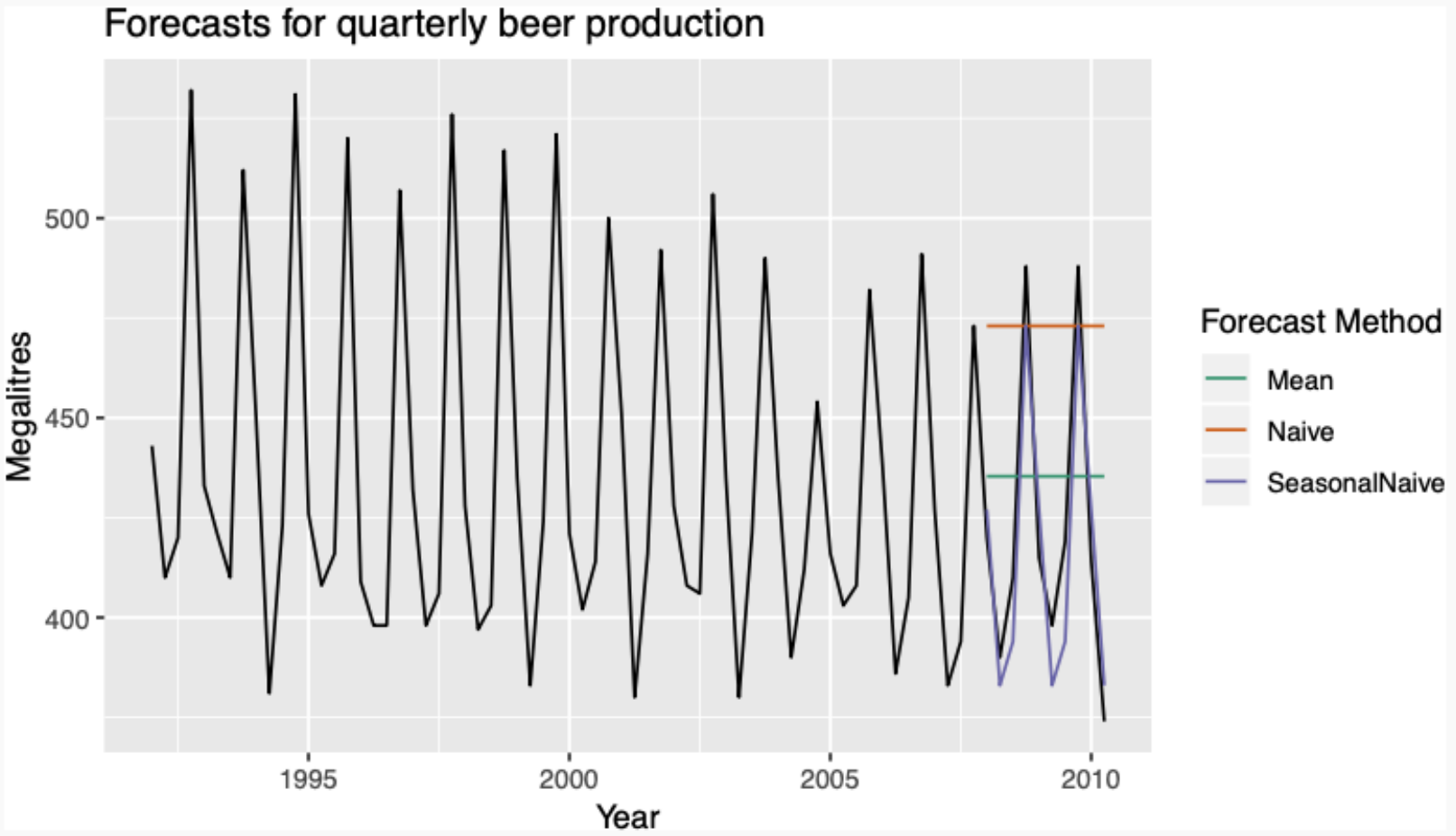
$$\text{MSE} = \text{mean}(e_{T+h}^2) \qquad \text{RMSE} = \sqrt{\text{mean}(e_{T+h}^2)}$$

$$\text{MAPE} = 100\text{mean}(|e_{T+h}|/|y_{T+h}|)$$

- MAE, MSE, RMSE are all scale dependent.
- MAPE is scale independent but is only sensible if $y_t \gg 0$ for all t, and y has a natural zero.

Đo độ chính xác dự báo (tt)

Ví dụ:



	RMSE	MAE	MAPE
Mean method	38.45	34.83	8.28
Naïve method	62.69	57.40	14.18
Seasonal naïve method	14.31	13.40	3.17

→ Seasonal naïve method
là PP tốt nhất cho tập dữ liệu này

Hai họ phương pháp mô hình hoá chính

- Các PP làm trơn mũ: mô tả xu hướng và tính thời vụ trong dữ liệu
- Các mô hình ARIMA: mô tả tự tương quan (autocorrelations) trong dữ liệu, chỉ áp dụng với CTG có tính ổn định về mặt thống kê (stationary)

6.4.2. Các PP làm trơn mũ (Exponential smoothing)

- Giới thiệu
- Làm trơn mũ đơn giản
- Các PP xu hướng
- Các PP thời vụ
- Phân loại các PP làm trơn mũ
- Các mô hình không gian trạng thái

Giới thiệu

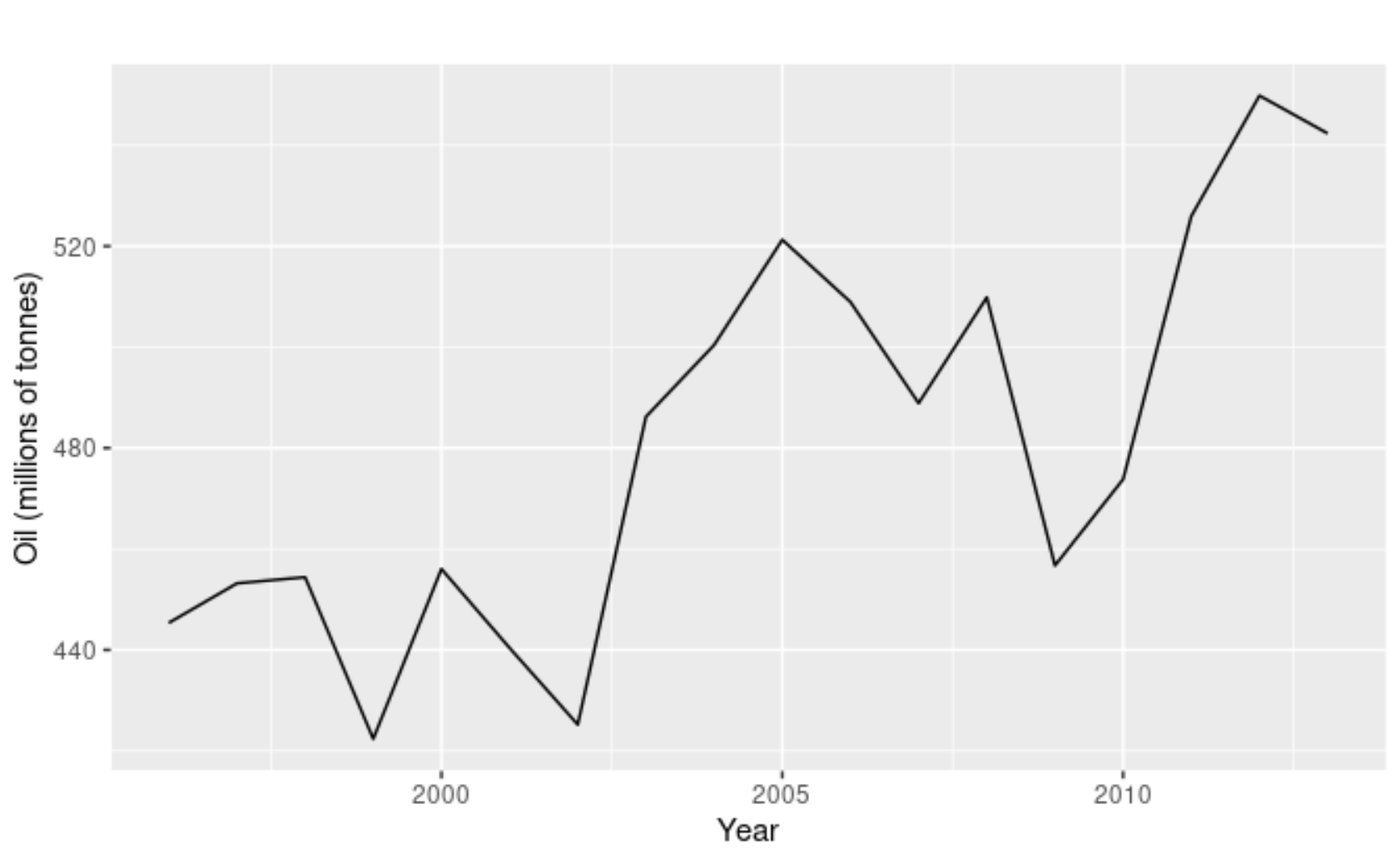
- Đề xuất vào cuối 1950s (Brown, 1959; Holt, 1957; Winters, 1960)
- Ý tưởng chính: Giá trị tương lai được dự báo bằng GTTB có trọng số của các giá trị quá khứ với trọng số giảm theo hàm mũ khi quan sát càng xa hiện tại
- Mô hình này sinh ra các dự báo đáng tin cậy một cách nhanh chóng và có thể áp dụng với nhiều dữ liệu CTG
- Gồm nhiều PP cho phép tiên đoán giá trị (point forecasts) và dải giá trị (interval forecasts)

6.4.2. Các PP làm trơn mũ (Exponential smoothing)

- Giới thiệu
- Làm trơn mũ đơn giản (Simple exponential smoothing - SES)
- Các PP xu hướng
- Các PP thời vụ
- Phân loại các PP làm trơn mũ
- Các mô hình không gian trạng thái

Simple exponential smoothing (SES)

- Phù hợp với dữ liệu không có quy luật xu hướng hoặc thời vụ rõ ràng



Oil production in Saudi Arabia from 1996 to 2013.

Các PP dự báo đơn giản

Time series $y_1, y_2, \ldots, y_T$.

Random walk forecasts (naïve method)

$$\hat{y}_{T+h|T} = y_T$$

(h=1,2,…)

Average forecasts

$$\hat{y}_{T+h|T} = \frac{1}{T}\sum_{t=1}^{T} y_t$$

→ Muốn một PP trung gian: đánh trọng số cao hơn đ/v dữ liệu gần đây hơn

→ SES dùng trung bình động có trọng số (weighted moving average) với trọng số giảm theo hàm mũ

Simple exponential smoothing (SES)

Forecast equation

$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1-\alpha)y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + \cdots$$

where $0 \leq \alpha \leq 1$ (smoothing parameter)

	Weights assigned to observations for:			
Observation	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$
y_T	0.2	0.4	0.6	0.8
y_{T-1}	0.16	0.24	0.24	0.16
y_{T-2}	0.128	0.144	0.096	0.032
y_{T-3}	0.1024	0.0864	0.0384	0.0064
y_{T-4}	$(0.2)(0.8)^4$	$(0.4)(0.6)^4$	$(0.6)(0.4)^4$	$(0.8)(0.2)^4$
y_{T-5}	$(0.2)(0.8)^5$	$(0.4)(0.6)^5$	$(0.6)(0.4)^5$	$(0.8)(0.2)^5$

Simple exponential smoothing (SES)

Component form

Forecast equation $\hat{y}_{t+h|t} = \ell_t$

Smoothing equation $\ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}$

- ℓ_t is the level (or the smoothed value) of the series at time t.
- $\hat{y}_{t+1|t} = \alpha y_t + (1-\alpha)\hat{y}_{t|t-1}$
  Iterate to get exponentially weighted moving average form.

Weighted average form

$$\hat{y}_{T+1|T} = \sum_{j=0}^{T-1} \alpha(1-\alpha)^j y_{T-j} + (1-\alpha)^T \ell_0$$

Optimisation

- Need to choose value for α and ℓ_0
- Similarly to regression — we choose α and ℓ_0 by minimising SSE(sum of squared errors)

$$\text{SSE} = \sum_{t=1}^{T} (y_t - \hat{y}_{t|t-1})^2.$$

- Unlike regression there is no closed form solution — use numerical optimization.

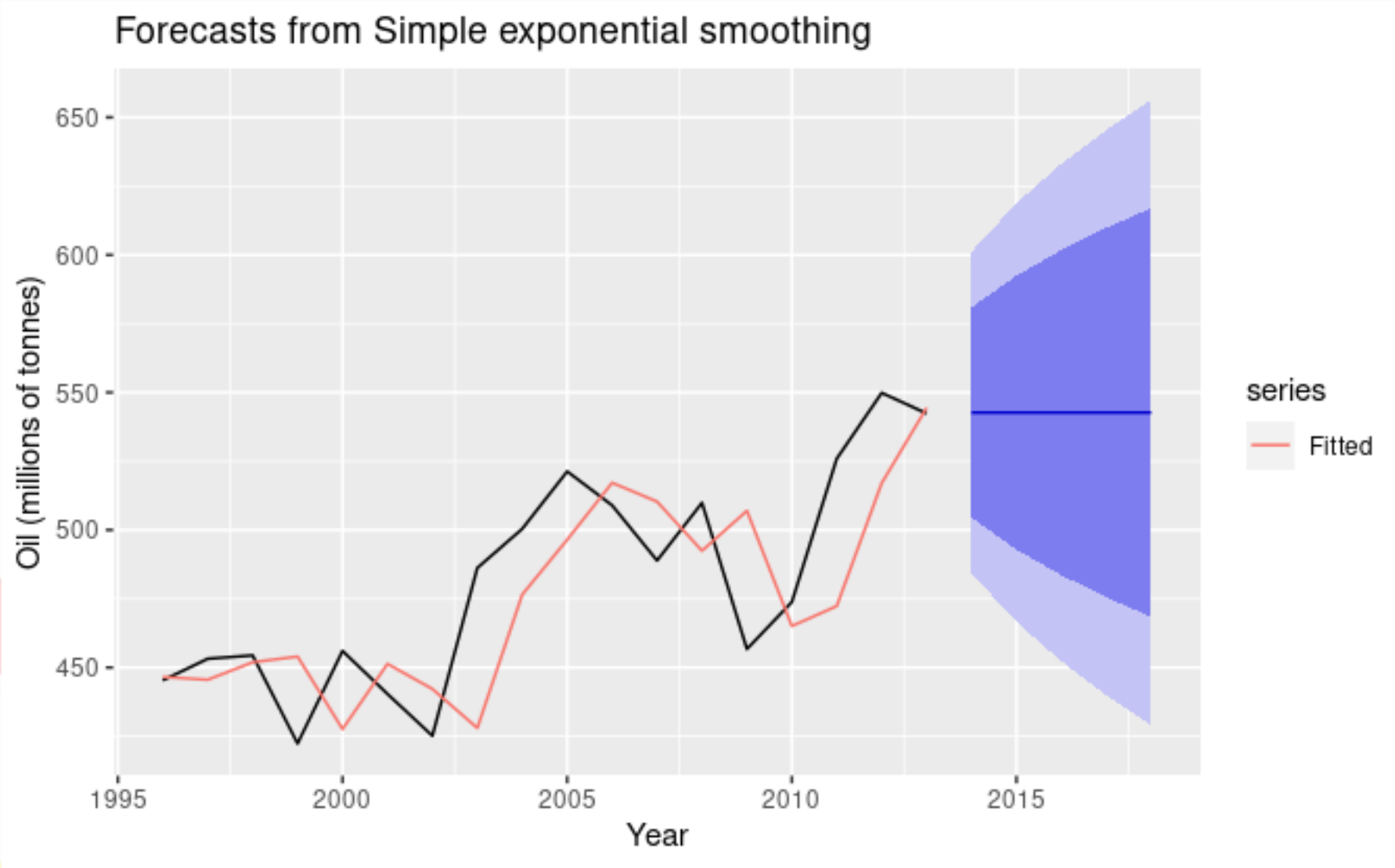
Ví dụ: Oil production in Saudi Arabia 1996–2013

Year	Time	Observation	Level	Forecast
	t	y_t	ℓ_t	$\hat{y}_{t+1\|t}$
1995	0		446.59	
1996	1	445.36	445.57	446.59
1997	2	453.20	451.93	445.57
1998	3	454.41	454.00	451.93
1999	4	422.38	427.63	454.00
2000	5	456.04	451.32	427.63
2001	6	440.39	442.20	451.32
2002	7	425.19	428.02	442.20
2003	8	486.21	476.54	428.02
2004	9	500.43	496.46	476.54
2005	10	521.28	517.15	496.46
2006	11	508.95	510.31	517.15
2007	12	488.89	492.45	510.31
2008	13	509.87	506.98	492.45
2009	14	456.72	465.07	506.98
2010	15	473.82	472.36	465.07
2011	16	525.95	517.05	472.36
2012	17	549.83	544.39	517.05
2013	18	542.34	542.68	544.39
	h			$\hat{y}_{T+h\|T}$
2014	1			542.68

$\alpha = 0.83$
$\ell_0 = 446.59$

(https://otexts.com/fpp2/ses.html)

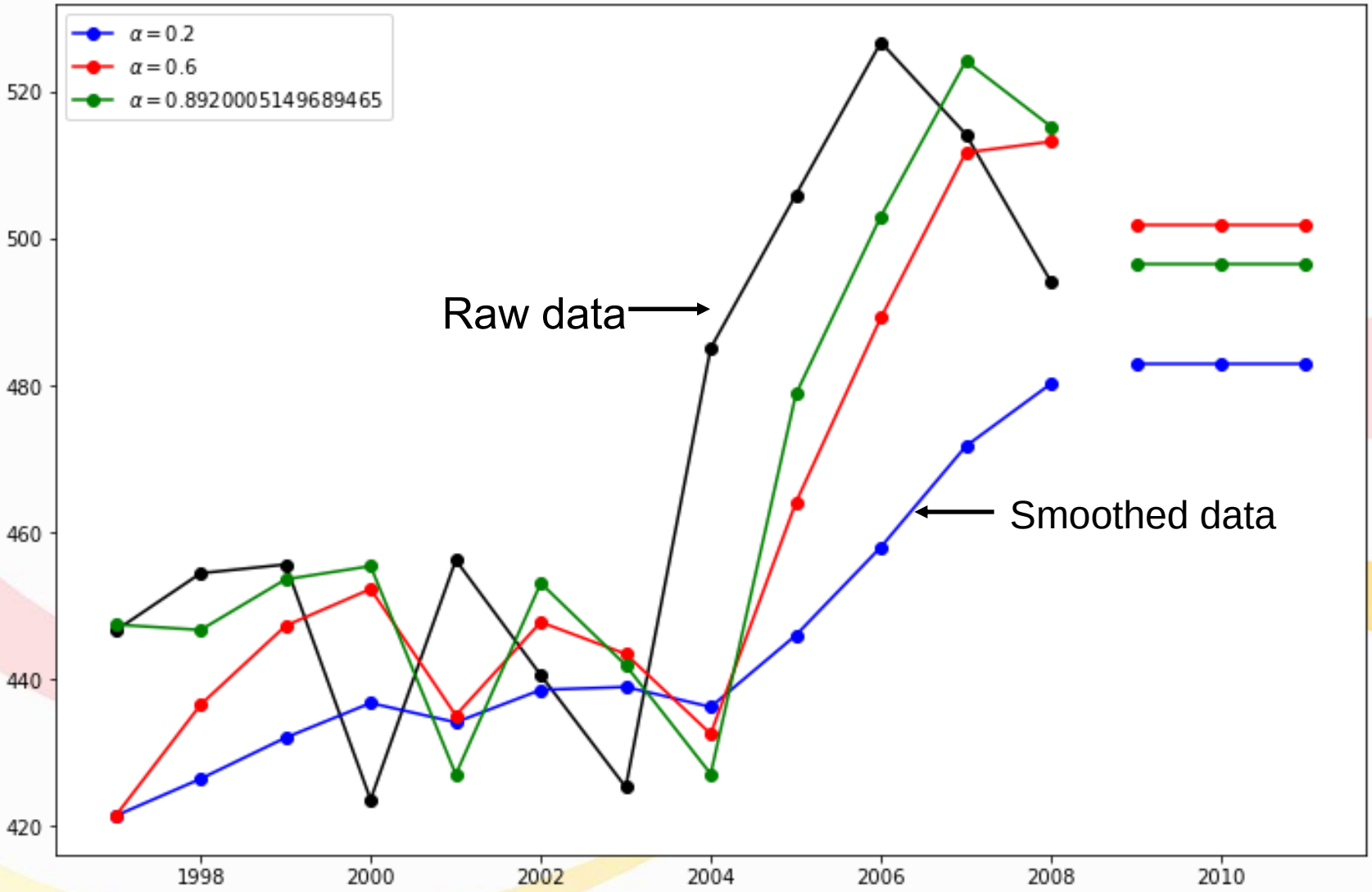
Ví dụ: Oil production in Saudi Arabia 1996–2013 (tt)



(https://otexts.com/fpp2/ses.html)

Ví dụ với Python

https://www.statsmodels.org/stable/examples/notebooks/generated/exponential_smoothing.html



6.4.2. Các PP làm trơn mũ (Exponential smoothing)

- Giới thiệu
- Làm trơn mũ đơn giản
- Các PP xu hướng (trend methods)
- Các PP thời vụ
- Phân loại các PP làm trơn mũ
- Các mô hình không gian trạng thái

Holt's linear trend method

- Holt (1957) mở rộng SES để dự báo dữ liệu có xu hướng tuyến tính

Component form

Forecast $\hat{y}_{t+h|t} = \ell_t + hb_t$

Level $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1})$

Trend $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1},$

- Two smoothing parameters α and β^* $(0 \leq \alpha, \beta^* \leq 1)$.
- ℓ_t level: weighted average between y_t & one-step ahead forecast for time t, $(\ell_{t-1} + b_{t-1} = \hat{y}_{t|t-1})$
- b_t slope: weighted average of $(\ell_t - \ell_{t-1})$ and b_{t-1}, current and previous estimate of slope.
- Choose $\alpha, \beta^*, \ell_0, b_0$ to minimise SSE.

Damped trend method

- Gardner & McKenzie (1985) mở rộng PP của Holt bằng cách thêm tham số để làm phẳng (dampen) xu hướng khi dự báo tương lai xa

Component form

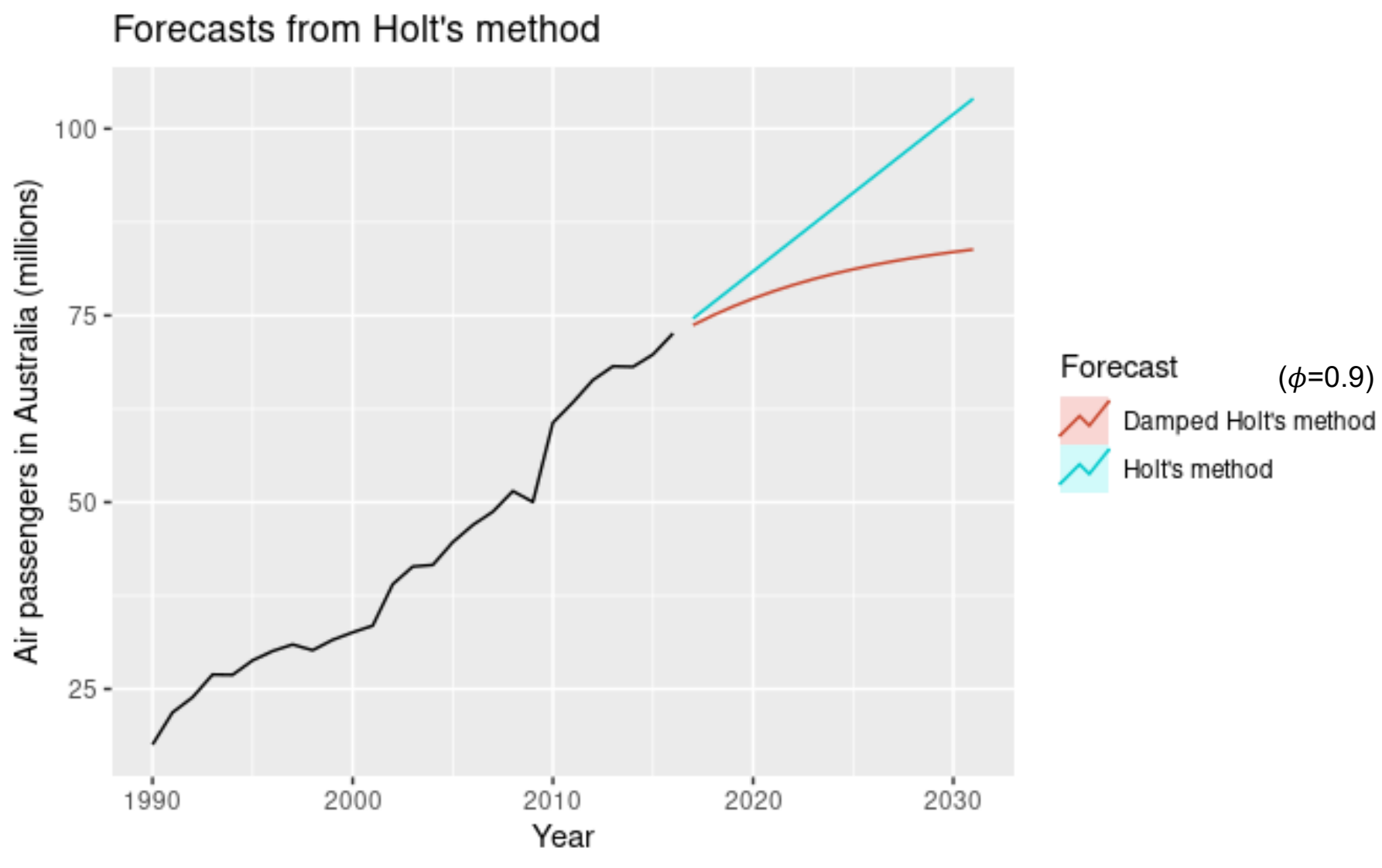
Forecast $\hat{y}_{t+h|t} = \ell_t + (\phi + \phi^2 + \cdots + \phi^h)b_t$

Level $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$

Trend $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}.$

- Damping parameter $0 < \phi < 1$.
- If $\phi = 1$, identical to Holt's linear trend.
- As $h \to \infty$, $\hat{y}_{T+h|T} \to \ell_T + \phi b_T/(1-\phi)$.
- Short-run forecasts trended, long-run forecasts constant.

Ví dụ: Air passengers in Australia 1990–2016



(https://otexts.com/fpp2/holt.html)

6.4.2. Các PP làm trơn mũ (Exponential smoothing)

- Giới thiệu
- Làm trơn mũ đơn giản
- Các PP xu hướng
- Các PP thời vụ (seasonal methods)
- Phân loại các PP làm trơn mũ
- Các mô hình không gian trạng thái

Holt-Winters' seasonal methods

- Holt (1957) và Winters (1960) đã mở rộng PP của Holt để mô hình hoá tính thời vụ
- Gồm PT dự báo và 3 PT làm trơn ứng với 3 thành phần: mức l_t, xu hướng b_t, thời vụ s_t , và 3 tham số làm trơn tương ứng α, β^*, γ
- Tần suất thời vụ (số mùa/năm): m (m=4: DL theo quý, m=12: DL theo tháng)
- Có 2 biến thể phụ thuộc vào bản chất của TP thời vụ:
  - PP cộng: khi biên độ của các dao động có tính thời vụ không biến thiên theo mức (level) của CTG
  - PP nhân: khi biên độ của các dao động có tính thời vụ tỉ lệ với mức của CTG

Holt-Winters' additive method

Component form

Forecast $\hat{y}_{t+h|t} = \ell_t + hb_t + s_{t+h-m(k+1)}$

Level $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1})$

Trend $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$

Season $s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m},$

- k = integer part of $(h-1)/m$. Ensures estimates from the final year are used for forecasting.
- Parameters: $0 \le \alpha \le 1$, $0 \le \beta^* \le 1$, $0 \le \gamma \le 1-\alpha$ and m = period of seasonality (e.g. $m = 4$ for quarterly data).

Holt-Winters' additive method (tt)

- Seasonal component is usually expressed as
  $s_t = \gamma^*(y_t - \ell_t) + (1 - \gamma^*)s_{t-m}$.
- Substitute in for ℓ_t:
  $s_t = \gamma^*(1-\alpha)(y_t - \ell_{t-1} - b_{t-1}) + [1 - \gamma^*(1-\alpha)]s_{t-m}$
- We set $\gamma = \gamma^*(1 - \alpha)$.
- The usual parameter restriction is $0 \leq \gamma^* \leq 1$, which translates to $0 \leq \gamma \leq (1 - \alpha)$.

Holt-Winters' multiplicative method

Component form

Forecast $\hat{y}_{t+h|t} = (\ell_t + hb_t)s_{t+h-m(k+1)}.$

Level $$\ell_t = \alpha\frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1})$$

Trend $$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$$

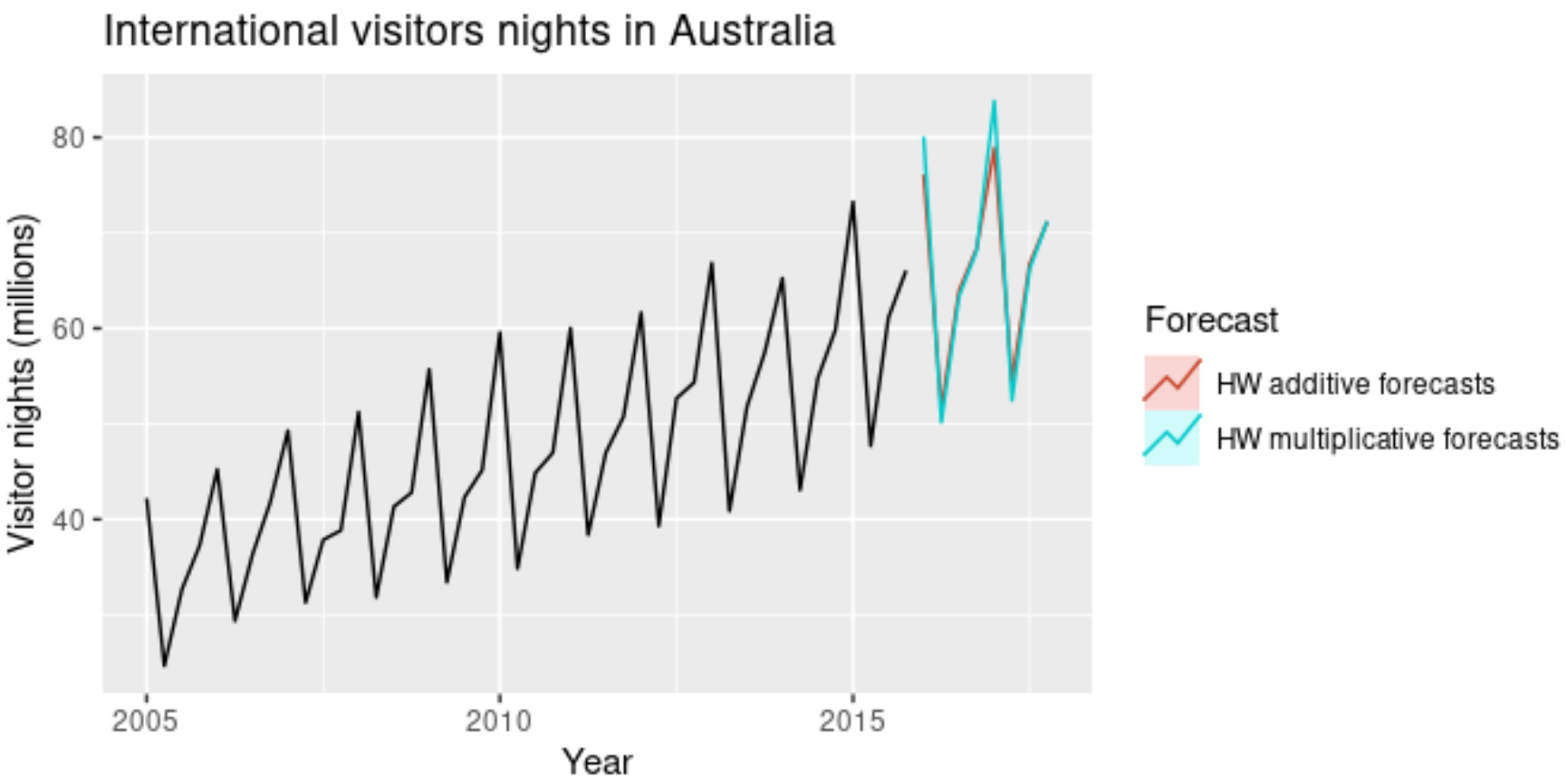
Season $$s_t = \gamma\frac{y_t}{(\ell_{t-1} + b_{t-1})} + (1-\gamma)s_{t-m}$$

- k is integer part of $(h-1)/m$.
- With additive method s_t is in absolute terms: within each year $\sum_i s_i \approx 0$.
- With multiplicative method s_t is in relative terms: within each year $\sum_i s_i \approx m$.

Ví dụ: International visitor nights in Australia

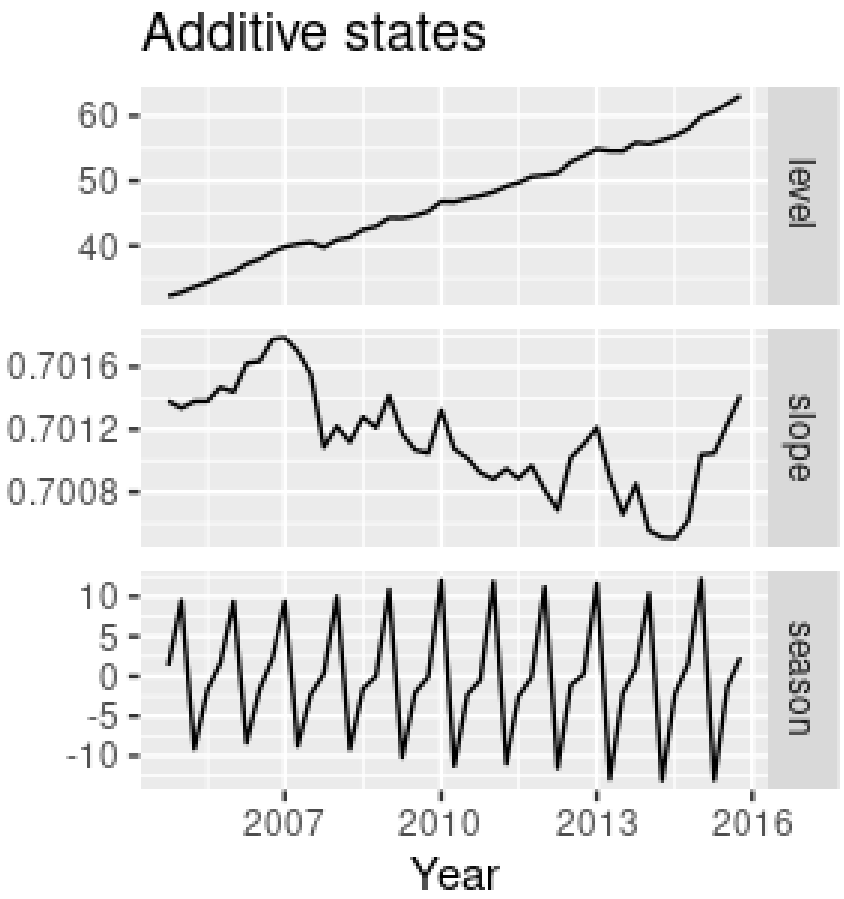
DL huấn luyện: 2005-2015 → tính thời vụ mạnh, tăng theo level

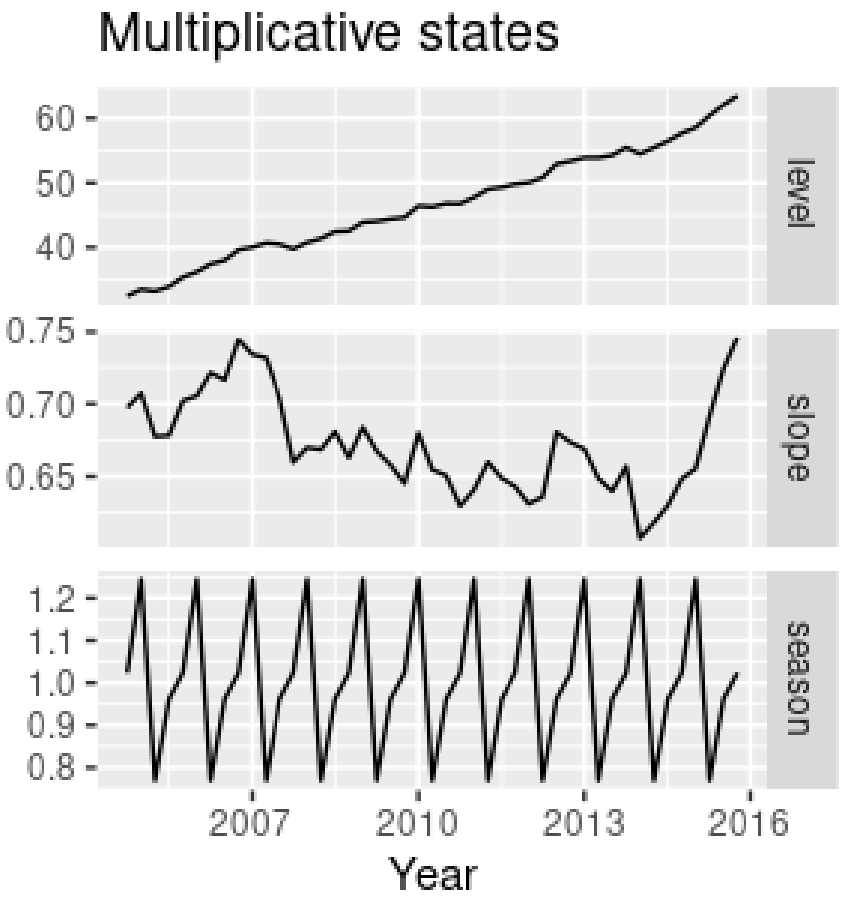
DL dự báo: 2016-2017



Ví dụ: International visitor nights in Australia (tt)

Phân tích các thành phần cho 2 mô hình Cộng và Nhân:





→ Mô hình Nhân phù hợp hơn mô hình Cộng do có thành phần thời vụ ổn định hơn

Holt-Winters' damped method

- PP Holt-Winters với xu hướng được làm phẳng (damped trend) và tính thời vụ theo mô hình nhân (multiplicative seasonality) thường cho dự báo chính xác nhất đ/v DL có tính thời vụ

$$\hat{y}_{t+h|t} = [\ell_t + (\phi + \phi^2 + \cdots + \phi^h)b_t]s_{t+h-m(k+1)}$$

$$\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$$

$$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$$

$$s_t = \gamma\frac{y_t}{(\ell_{t-1} + \phi b_{t-1})} + (1-\gamma)s_{t-m}$$

6.4.2. Các PP làm trơn mũ (Exponential smoothing)

- Giới thiệu
- Làm trơn mũ đơn giản
- Các PP xu hướng
- Các PP thời vụ
- Phân loại các PP làm trơn mũ
- Các mô hình không gian trạng thái

Các PP làm trơn mũ

Trend Component		Seasonal Component N (None)	A (Additive)	M (Multiplicative)
N	(None)	(N,N)	(N,A)	(N,M)
A	(Additive)	(A,N)	(A,A)	(A,M)
A_d	(Additive damped)	(A_d,N)	(A_d,A)	(A_d,M)

(N,N): Simple exponential smoothing

(A,N): Holt's linear method

(A_d,N): Additive damped trend method

(A,A): Additive Holt-Winters' method

(A,M): Multiplicative Holt-Winters' method

(A_d,M): Damped multiplicative Holt-Winters' method

Các công thức đệ quy

Trend	Seasonal: N	Seasonal: A	Seasonal: M
N	$\hat{y}_{t+h\|t} = \ell_t$ $\ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}$	$\hat{y}_{t+h\|t} = \ell_t + s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)\ell_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1}) + (1-\gamma)s_{t-m}$	$\hat{y}_{t+h\|t} = \ell_t s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)\ell_{t-1}$ $s_t = \gamma(y_t/\ell_{t-1}) + (1-\gamma)s_{t-m}$
A	$\hat{y}_{t+h\|t} = \ell_t + hb_t$ $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$	$\hat{y}_{t+h\|t} = \ell_t + hb_t + s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}$	$\hat{y}_{t+h\|t} = (\ell_t + hb_t)s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$ $s_t = \gamma(y_t/(\ell_{t-1} + b_{t-1})) + (1-\gamma)s_{t-m}$
A_d	$\hat{y}_{t+h\|t} = \ell_t + \phi_h b_t$ $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$	$\hat{y}_{t+h\|t} = \ell_t + \phi_h b_t + s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1} - \phi b_{t-1}) + (1-\gamma)s_{t-m}$	$\hat{y}_{t+h\|t} = (\ell_t + \phi_h b_t)s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$ $s_t = \gamma(y_t/(\ell_{t-1} + \phi b_{t-1})) + (1-\gamma)s_{t-m}$

$\phi_h = \phi + \phi^2 + \cdots + \phi^h$, and k is the integer part of $(h-1)/m$

Các hàm Python của thư viện statmodels

(https://www.statsmodels.org/stable/examples/notebooks/generated/exponential_smoothing.html)

- Simple exponential smoothing method

SimpleExpSmoothing(y).fit()

- Holt's linear trend method

Holt(y).fit()

- Damped trend method

Holt(y, damped_trend=True).fit()

- Holt-Winters seasonal methods

ExponentialSmoothing(y, trend='add', seasonal='add').fit()

ExponentialSmoothing(y, trend='add', seasonal='mul').fit()

ExponentialSmoothing(y, trend='add', seasonal='add', damped_trend=True).fit()

ExponentialSmoothing(y, trend='add', seasonal='mul', damped_trend=True).fit()

6.4.2. Các PP làm trơn mũ (Exponential smoothing)

- Giới thiệu
- Làm trơn mũ đơn giản
- Các PP xu hướng
- Các PP thời vụ
- Phân loại các PP làm trơn mũ
- Các mô hình không gian trạng thái

Các mô hình không gian trạng thái

- Các PP làm trơn mũ là các thuật toán dự báo giá trị (point forecasts)
- Các mô hình KGTT (state space models)
  - Không những dự báo giá trị mà còn dự báo dải giá trị (interval forecasts)
  - Là các mô hình xác suất có thể sinh ra phân bố dự báo (forecast distribution) dựa trên dữ liệu đã có
  - Cho phép lựa chọn mô hình phù hợp dùng các tiêu chuẩn thông tin như Akaike's Information Criterion (AIC)

Các mô hình không gian trạng thái (tt)

- Mỗi mô hình có một PT mô tả DL quan sát và các PT trạng thái mô tả các trạng thái không quan sát được (level, trend, seasonal) thay đổi theo thời gian như thế nào
- Mỗi PP làm trơn mũ có 2 mô hình: lỗi cộng (additive errors) và lỗi nhân (multiplicative errors) → tổng cộng có 18 mô hình
- Các mô hình được đặt tên là ETS (Error,Trend,Seasonal)
  - Error = {A,M}
  - Trend = {N,A,A_d}
  - Seasonal = {N,A,M}
- Mô tả chi tiết trong [1] (Chapter 7) và cài đặt bằng Python bởi hàm *tsa.statespace.ExponentialSmoothing* của thư viện statmodels (https://www.statsmodels.org/stable/statespace.html#)

Bài tập

Tham khảo 2 blog post dưới đây và thử nghiệm với dữ liệu và code kèm theo để hiểu rõ hơn về các nội dung lý thuyết đã học:

- Phân tích dữ liệu khám phá (EDA) trên dữ liệu chuỗi thời gian dùng các PP trực quan và thống kê:

https://pawarbi.github.io/blog/forecasting/r/python/rpy2/altair/2020/04/21/timeseries-part1.html

- Các bước dự báo, đánh giá PP dự báo, lựa chọn mô hình, kết hợp các mô hình dự báo:

https://pawarbi.github.io/blog/forecasting/r/python/rpy2/altair/fbprophet/ensemble_forecast/uncertainty/simulation/2020/04/21/timeseries-part2.html