Making Sense of Time: Analysing Economic Time Series

Presented by SDS x DSESC

X

Team Members

Yida

Y1 DSE

Ernest

Y2 DSE

Yan Zhe

Y2 DSE

Xiang Ting

Y3 DSE

Hanover
and Tyke

Presented by
Phyllis Schwaiger

Date
2028

Yida

Content

1. What is time series data?
2. What is time series analysis?
3. Visualising time series
4. Components of a time series
5. Stationarity (intuition + why it matters)
6. Making data stationary: differencing (with demo)
7. Forecasting techniques

What is Time Series Data?

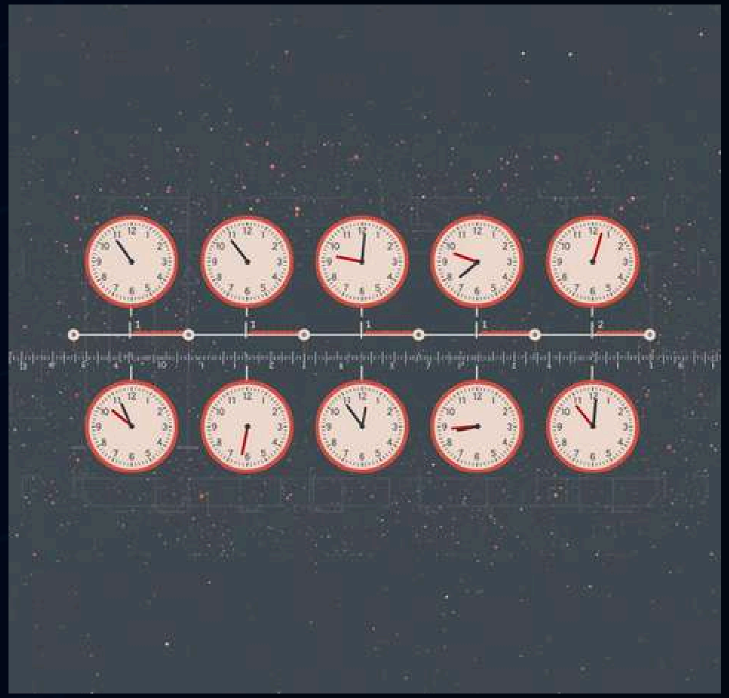
- A time-series is a sequence of measurements taken in chronological order — each value has a time attached and order matters.
- Think of each point as ‘value + time’ — not just a number.
- Example:
  - 25°C at 8am, 29°C at 2pm, 25°C at 8pm.

Why Order Matters

- Earlier values often influence later values. This means we cannot randomly shuffle time-series data.
- Example:
  - Yesterday's weather affects today's temperature.
- *Analogy: Like dominoes — what happens earlier affects what comes next.*

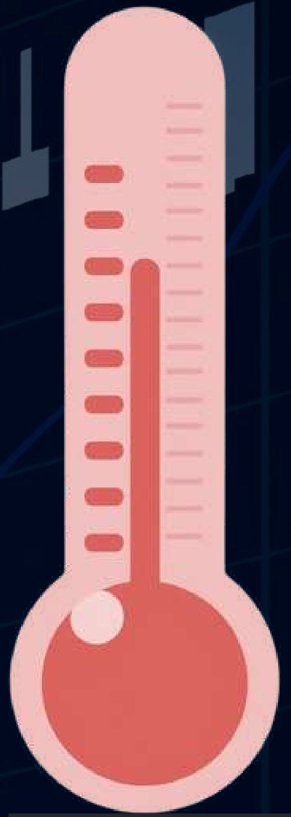
Why Fixed Intervals

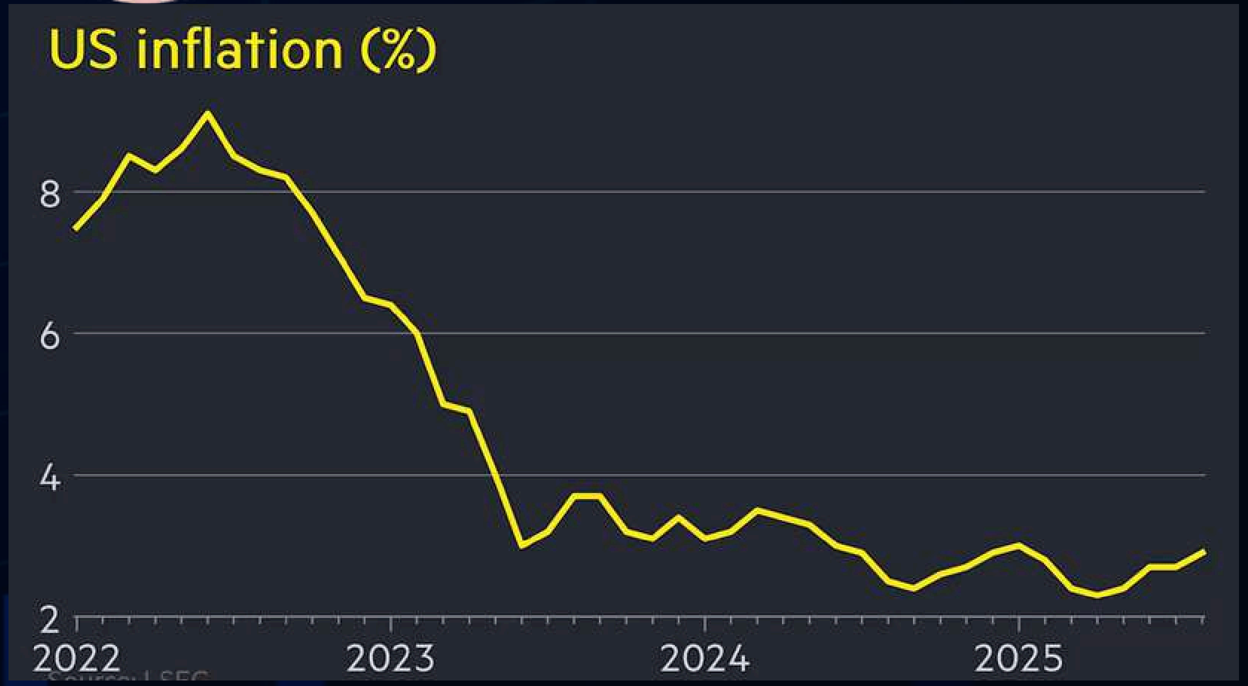
- Observations recorded at regular, evenly spaced times
  - daily, weekly, monthly, quarterly, yearly.
- Regular spacing makes patterns easier to spot and compare.
- Most forecasting methods assume fixed intervals.
- Easier to handle trends, seasonality, and missing values.
- Note: Irregular time series exist, but require different tools.



Everyday Examples

- **Hourly**: Temperature readings.
- **Daily**: Exchange rates or stock prices.
- **Monthly**: Inflation or unemployment rates.
- **Quarterly**: Company earnings reports.





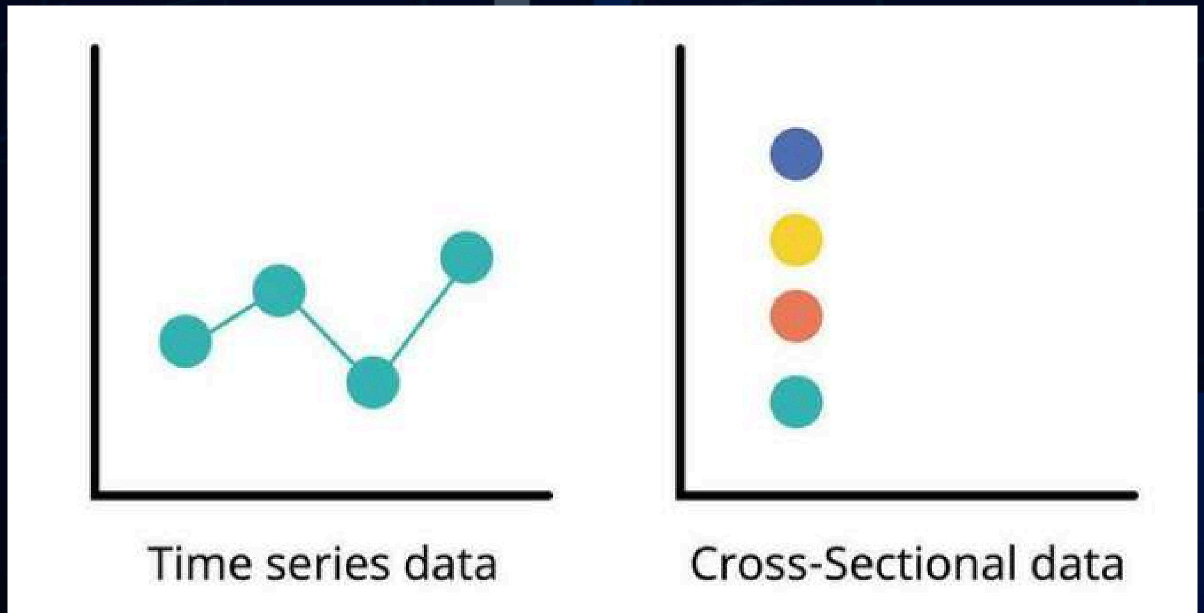
Time Series vs Cross-sectional

Time-series:

- Same unit over time (e.g., your monthly phone bill for 5 years).
- Explains how things change over time for one unit. (When/How?)

Cross-sectional:

- Many units at one time (e.g., heights of 50 students this week).
- Compares across individuals at one time. (Who/Which?)



Why is Time Series Unique

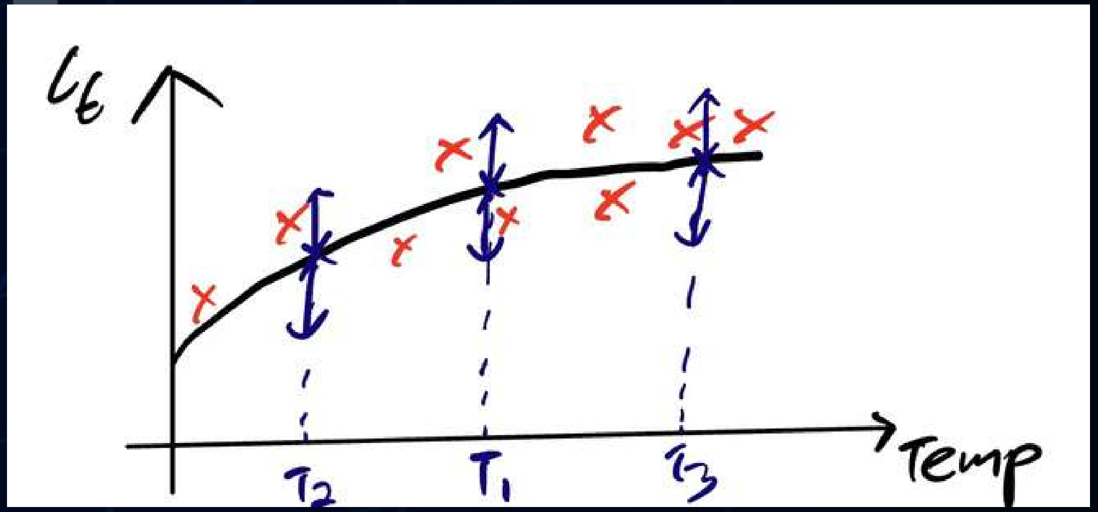
- Regression problems:
  - Predict one variable using other variables.
  - Example: ice-cream sales explained by temperature.
- Time-series problems:
  - Predict a variable using its own past values.
  - Time order matters.

Ice Cream Example: Time Series

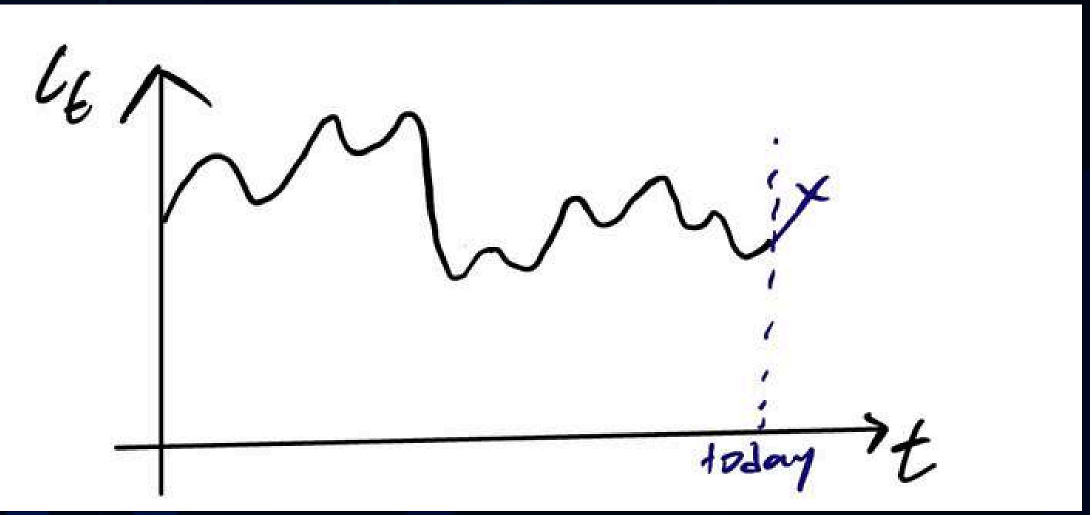
- Predict sales using past sales, not an external variable.
- Example:
  - Predict tomorrow's sales C_t using today's sales C_{t-1}.
- Key shift from regression:
  - Time becomes the x-axis.
  - Observations are ordered in time.

Difference between Regression and Time Series

- Regression problems
  - Predict C_t using another variable (e.g. temperature).
  - Same temperatures can occur again.
  - Predictions are often interpolation.



- Time-series problems
  - Predict C_t using past values (e.g. C_{t-1}).
  - Future points lie beyond observed time.
  - Predictions are extrapolation in time.



What is Time Series Analysis?

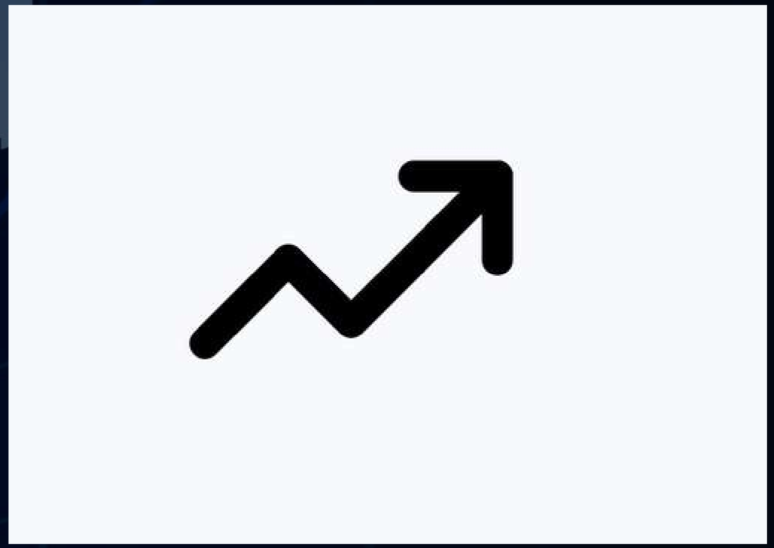
- A set of tools to analyse the timestamp data to extract meaningful insights and predictions about the future.
- Helps organisations drive better business decisions.

Forecasting

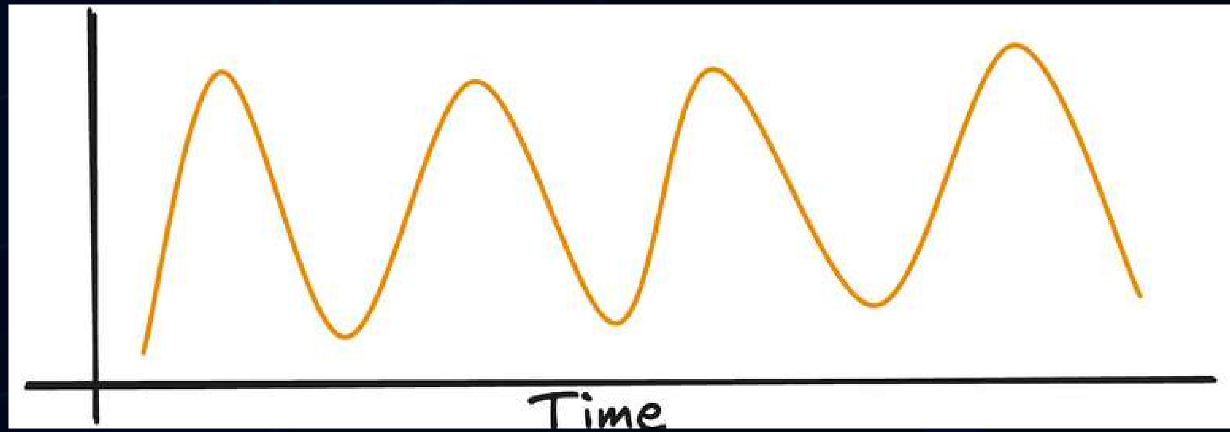
- Use past values to make short-term predictions.
- Retail:
  - Forecast future sales over time and optimise inventory, staffing and promotions.
- Agriculture:
  - Analyse and forecast weather and climate patterns over time to inform planting, irrigation, and harvesting decisions.

Understanding Patterns

- Before forecasting we usually want to know what kind of behaviour the data has.



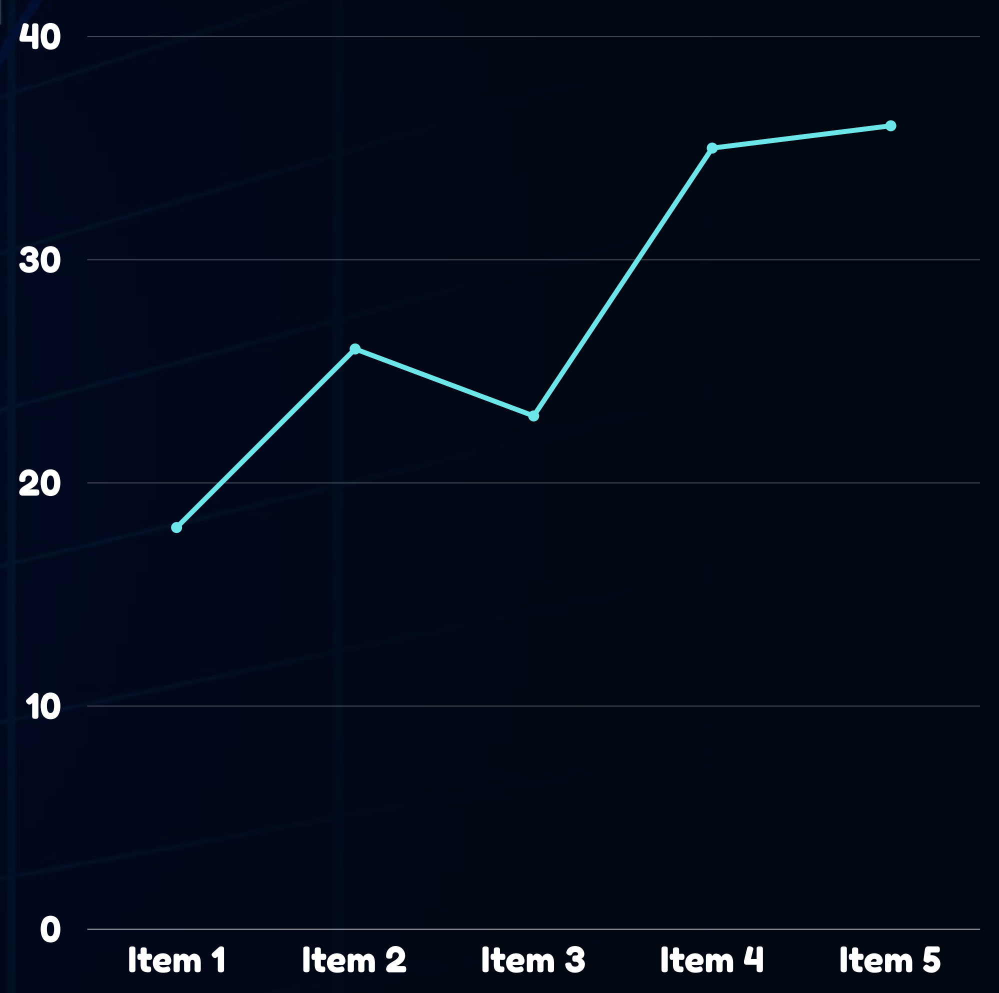
- Long-run movement (trend)
  - GDP generally growing over decades.
- Fixed periodic/calendar-based patterns (seasonality)
  - Retail sales peaking every December.



- Short-term shocks (one-off dips or spikes)
  - A sudden drop during a recession.

One quick takeaway

- Always look at the data first, visualize trends/seasonality and check dynamics before fitting models.



Visualising Time Series

U.S. Real GDP — GDPC1 (FRED)

- Source: Federal Reserve / FRED (series GDPC1).
- Frequency: Quarterly (long historical sample).
- Inflation-adjusted output, measured in chained dollars.
- Adjustment: Seasonally adjusted (SAAR) — reported as an annual rate.
- Gives us a long historical view of how the U.S. economy evolves over time — across expansions, recessions, and major shocks.

GDPC1 isn't great for seasonal demos

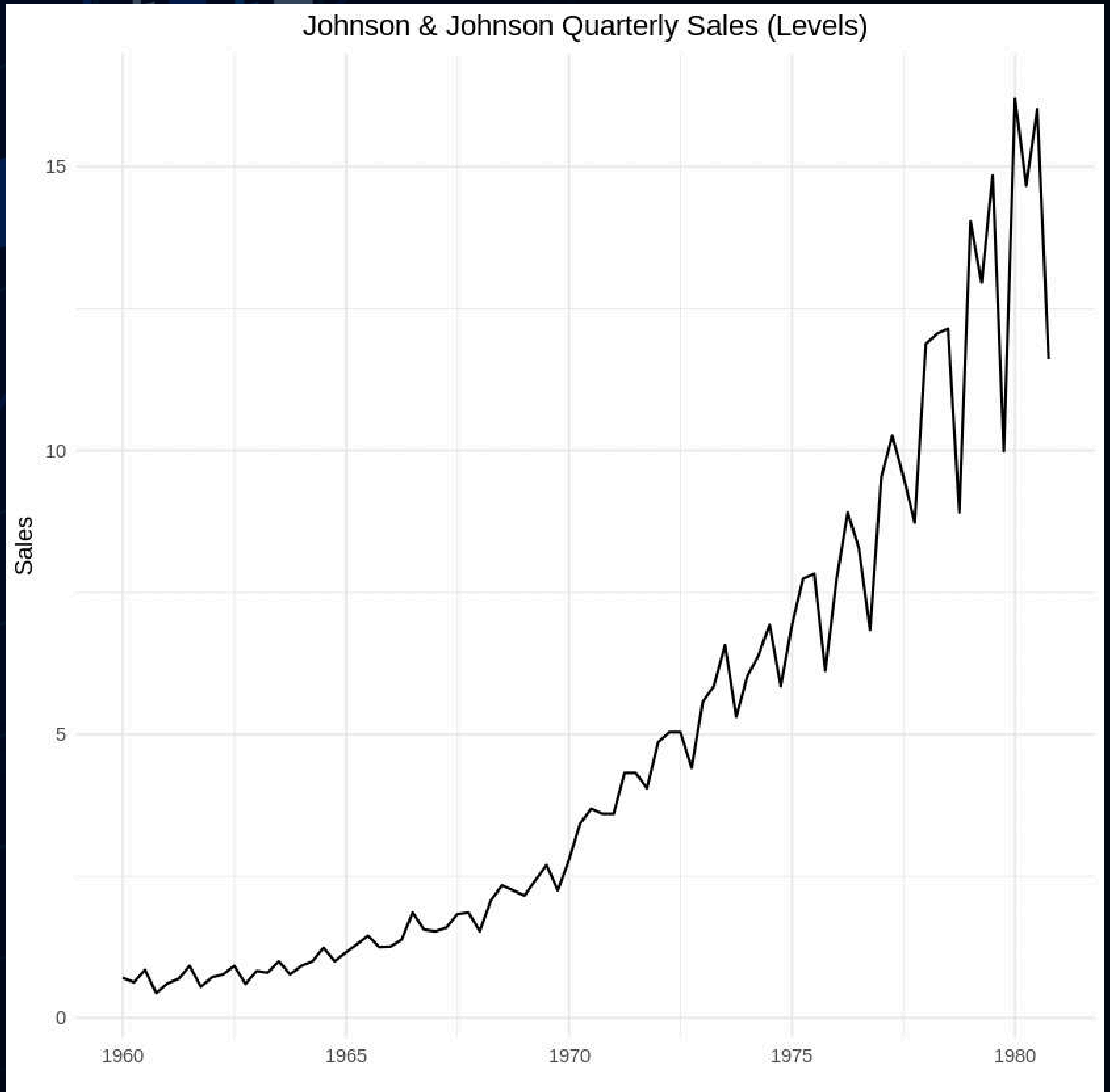
- GDPC1 is seasonally adjusted (SAAR). Seasonal effects (regular within-year patterns) have already been removed by statistical offices, so Q1 vs Q4 differences will be tiny or absent.
- So unlike data like retail sales or tourism, you shouldn't expect strong, predictable seasonal patterns in this series.
- So: keep GDPC1 for trend / cycle / growth demos. Use a different dataset to learn seasonality.

JohnsonJohnson

- Johnson & Johnson is a leading, global, research-based healthcare company that focuses on two core sectors:
    - Innovative Medicine (pharmaceuticals)
    - MedTech (medical devices)
- Quarterly earnings of Johnson & Johnson.
- Covers multiple decades (1960–1980).
- Classic benchmark time series in R.
- Shows trend + seasonal patterns.

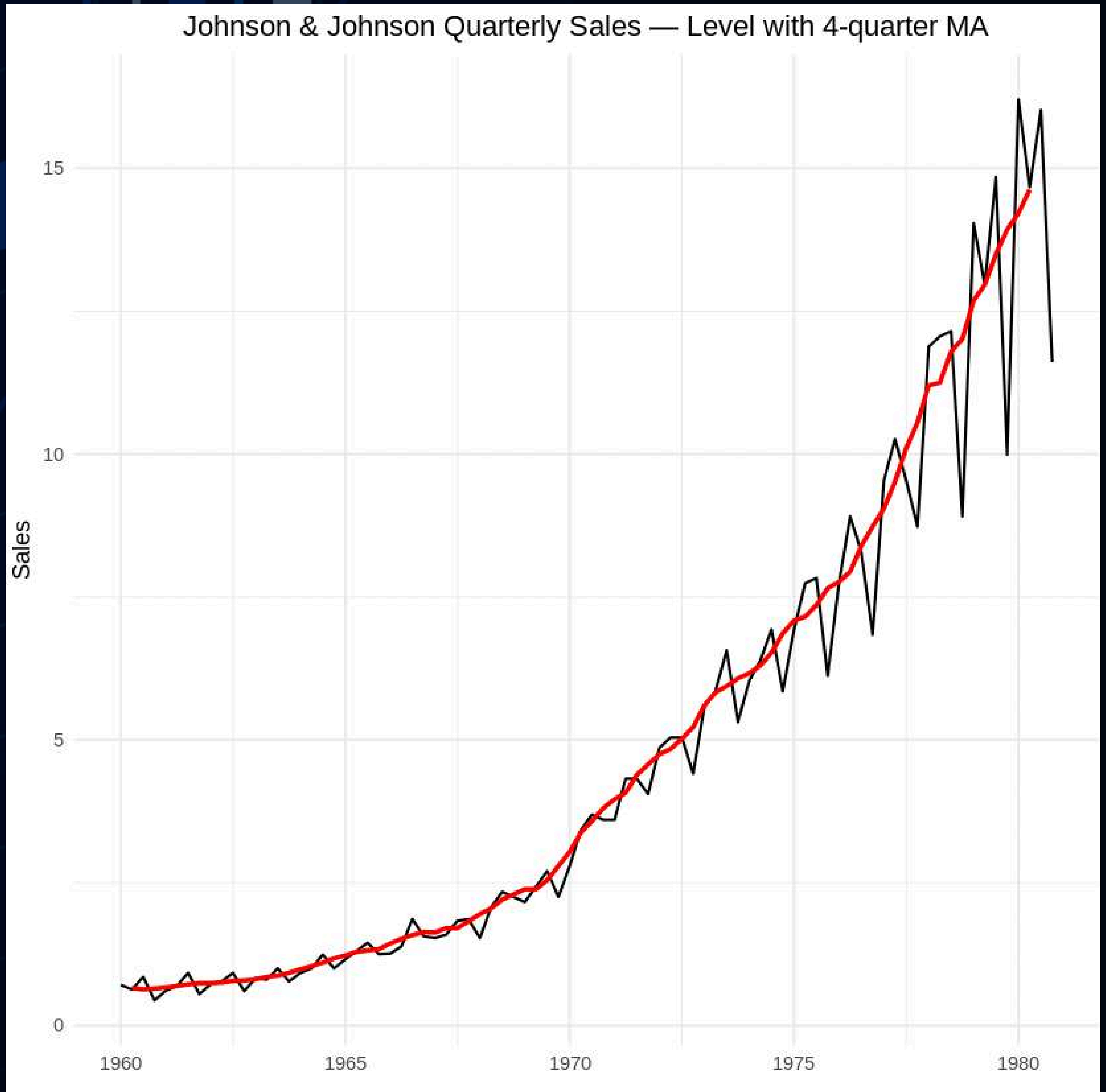
Time Plot (Main Plot)

- Plot the series against time.
- Look for:
  - Overall direction (trend).
  - Big drops or spikes.
  - Periods of instability.
- This is the most important first visual.



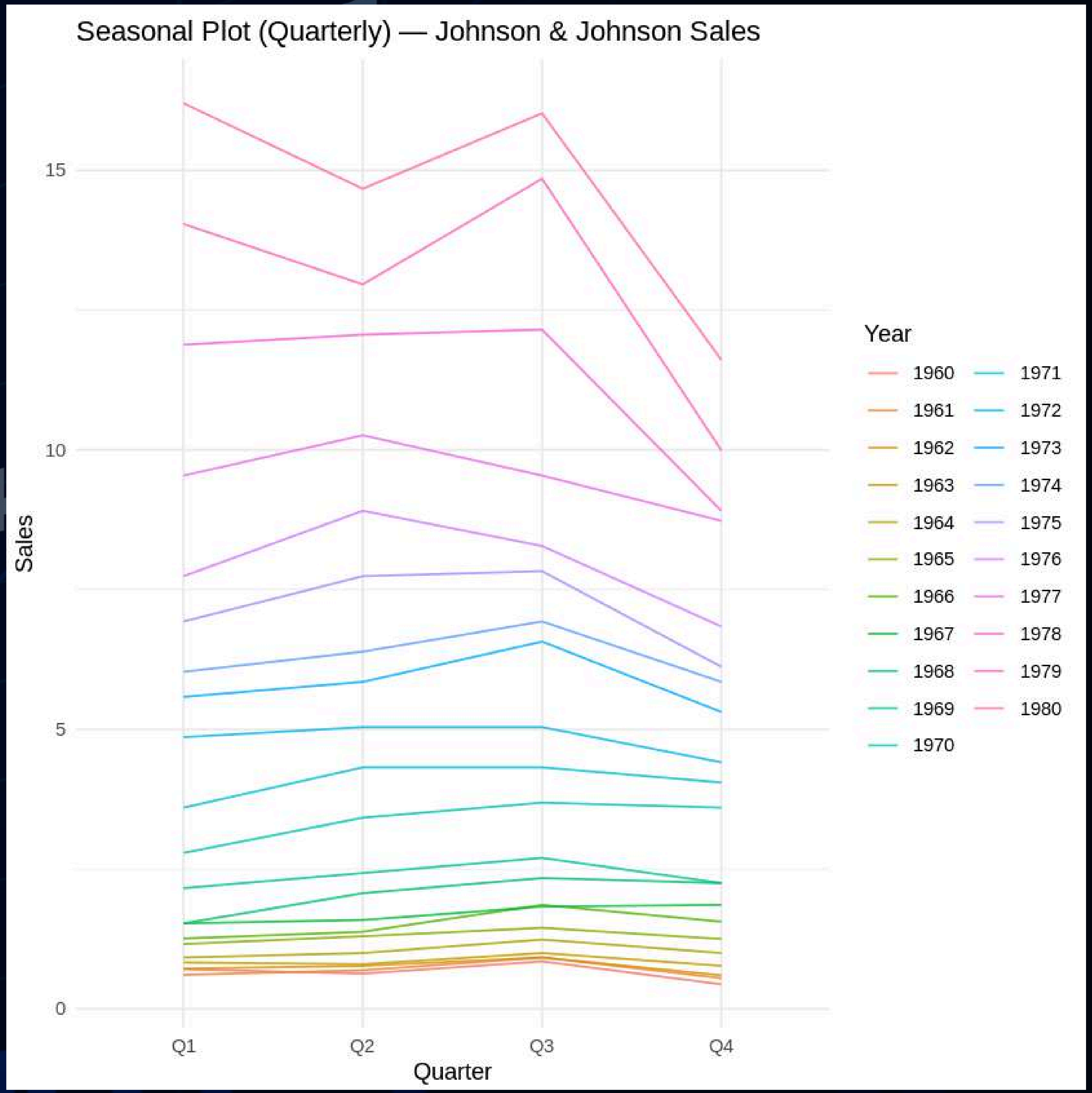
Smoothing the Series: 4-Quarter Moving Average

- Black line: raw quarterly GDP (noisy).
- Red line: 4-quarter moving average.
- Averages each year of quarterly data as it slides forward.
- Makes long-run trends and recessions easier to see.
- This is not a forecast — just a visual aid.



Seasonal Plots

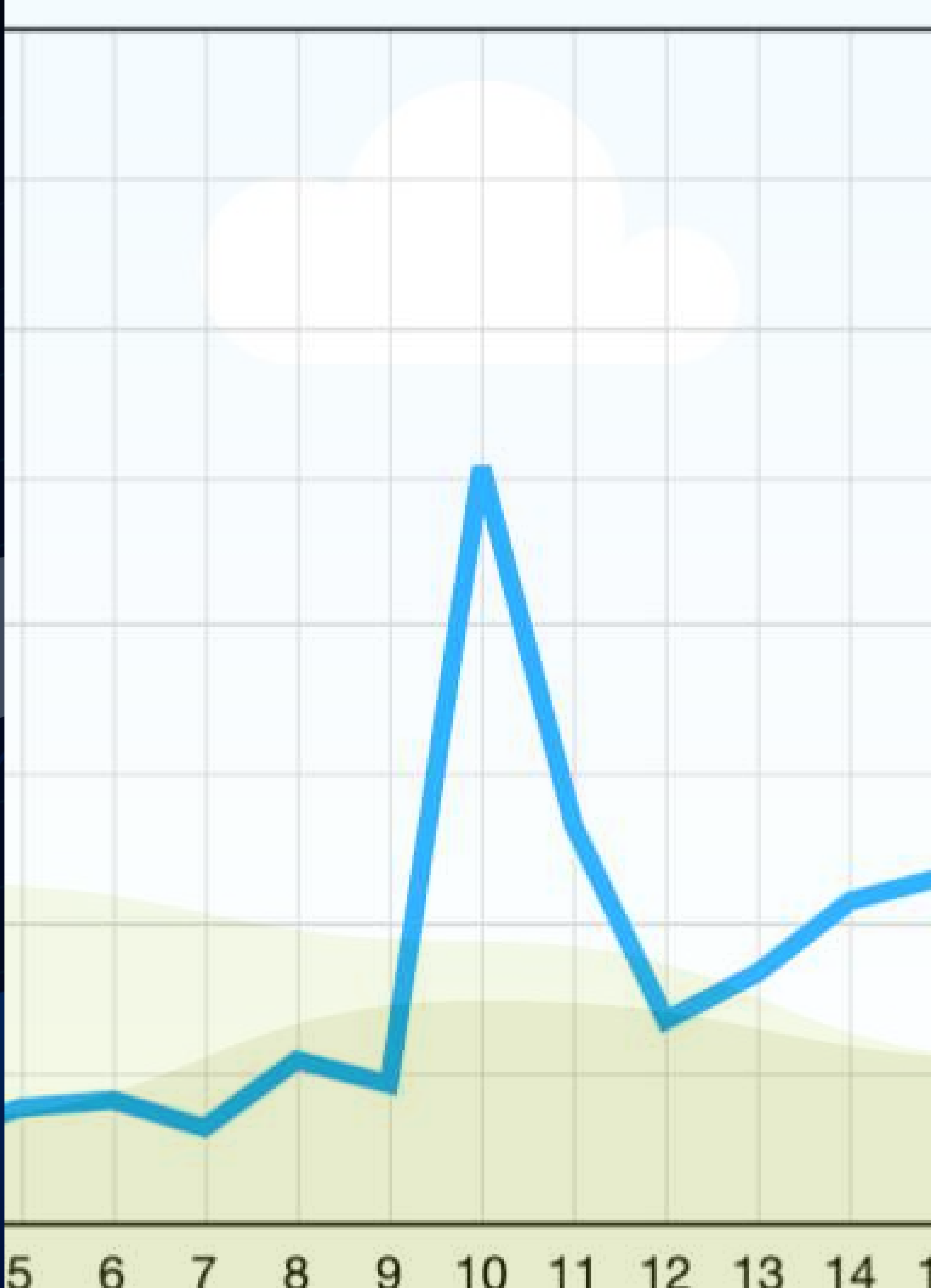
- Q1 → Q2 → Q3: rising
  - Sales generally increase from Q1 to Q2.
  - They rise further from Q2 to Q3.
- Q3 → Q4: sharp drop
  - Sales consistently fall from Q3 to Q4.
  - This drop is visible in nearly every line, regardless of the overall level of sales that year.



Ernest

What can we see in a Time Series?

→ Just a line of raw data changing over time?
→ Just one story? or multiple stories at the same time?



Decomposing a Time Series

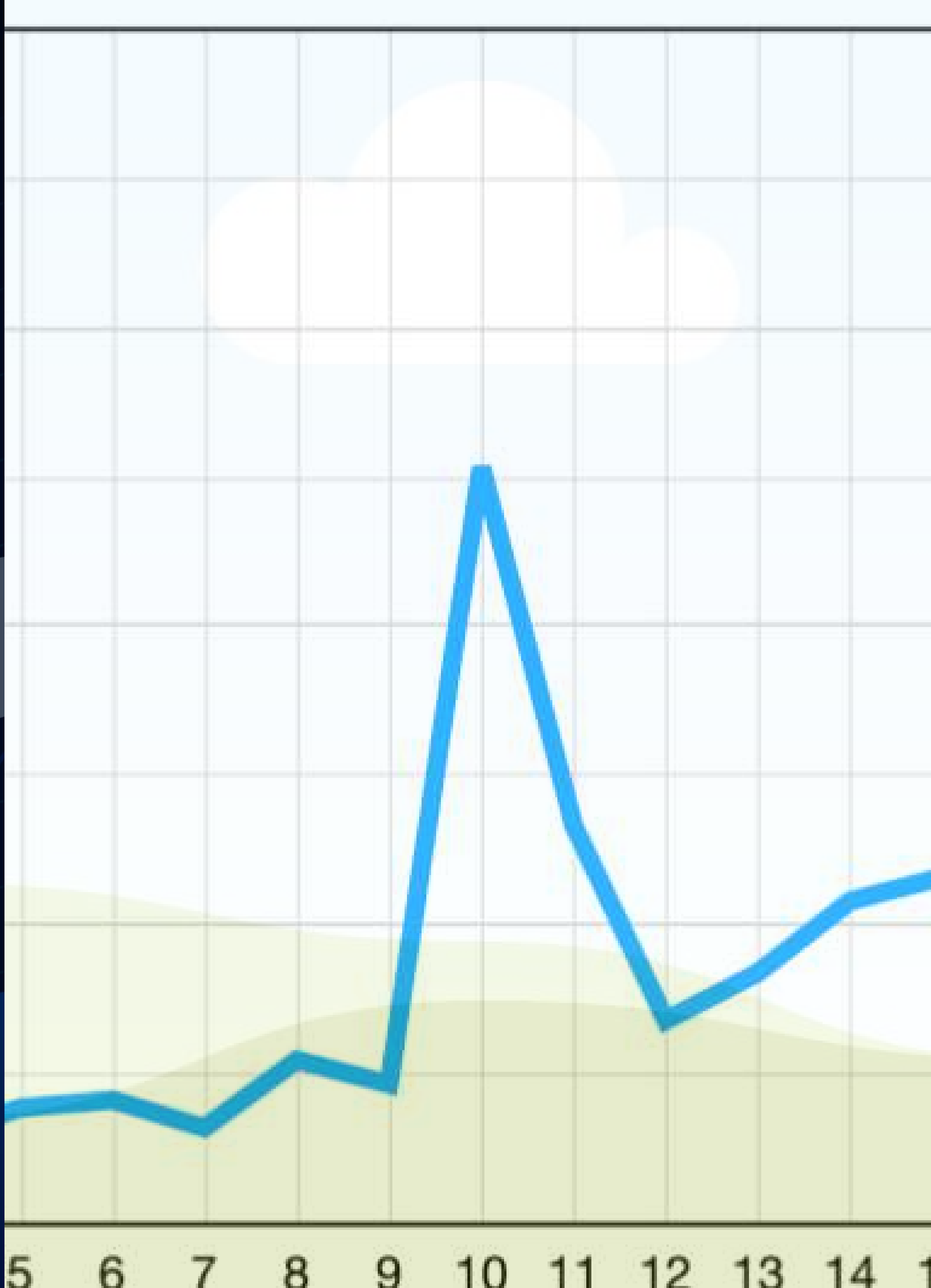
TREND

CYCLICAL

SEASONALITY

NOISE

MULTIPLE FACTORS WORKING TOGETHER



Costly Mistakes

BUSINESS

Misperception:
Higher sales every December = Business is Improving

Reality:
Seasonal spikes in December, will fade in January

POLICY-MAKING

Reality:
Short-Term Fluctuations

Misperception:
Serious Problem!

Consequence:
Unnecessary Intervention

ANALYSIS

Consequence:
Choosing over complicated models to explain simple seasonal trends

Breaking Down Time Series

LONG TERM TRENDS

REGULAR PATTERNS

RANDOM VARIATION

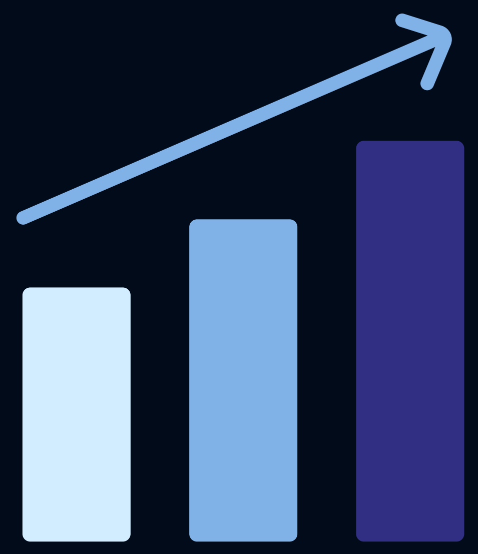
Breaking Down Time Series

Understand the structure of the data

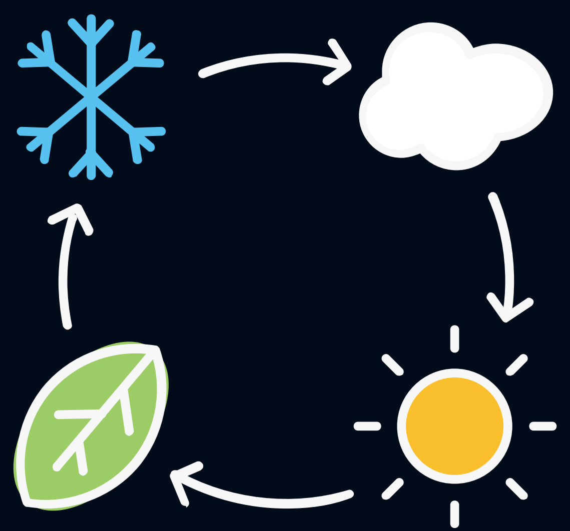
Choose Appropriate Models

Explain changes intuitively and clearly

Components of Time Series

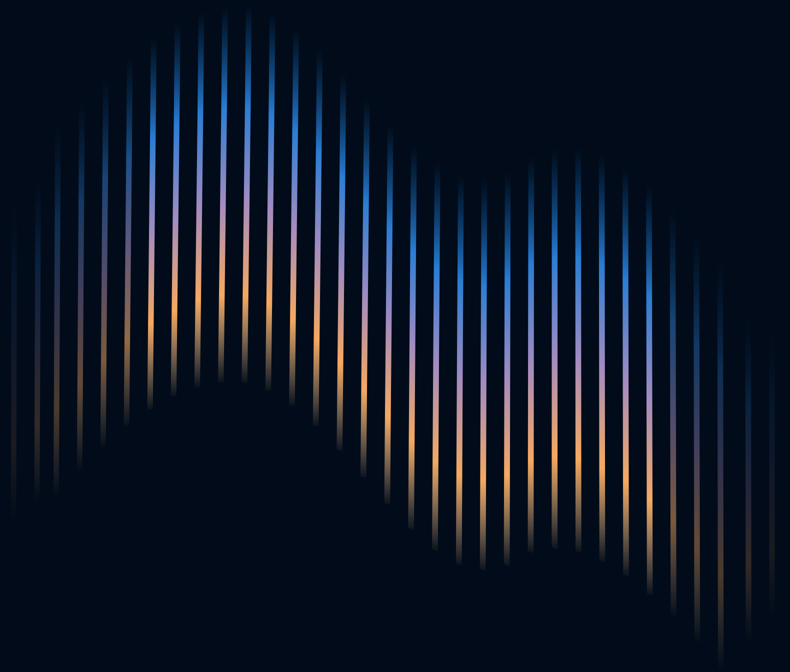


TREND



SEASONALITY

CYCLICAL



NOISE

Trend

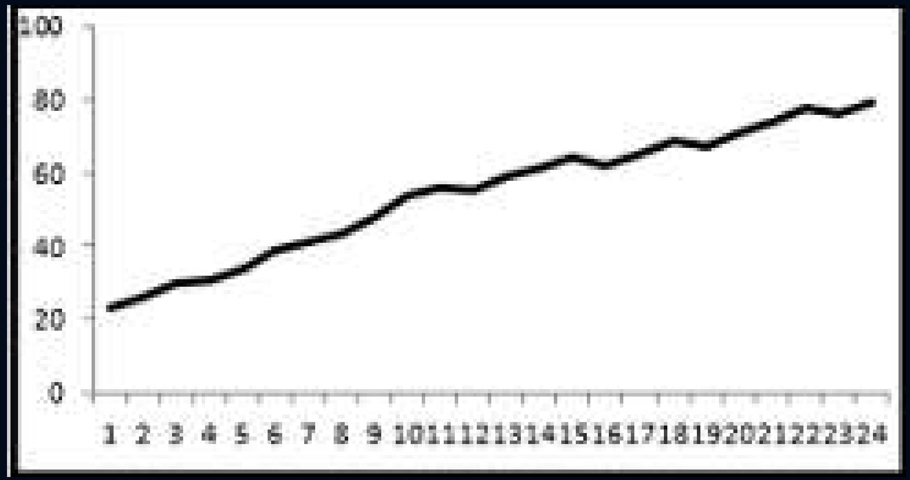
Long Run Direction of Time Series

GENERAL INCREASE

GENERAL DECREASE

STABLE

Ignore short-term fluctuations!



Components of Time Series

TREND | SEASONALITY | CYCLICAL | NOISE

Seasonality

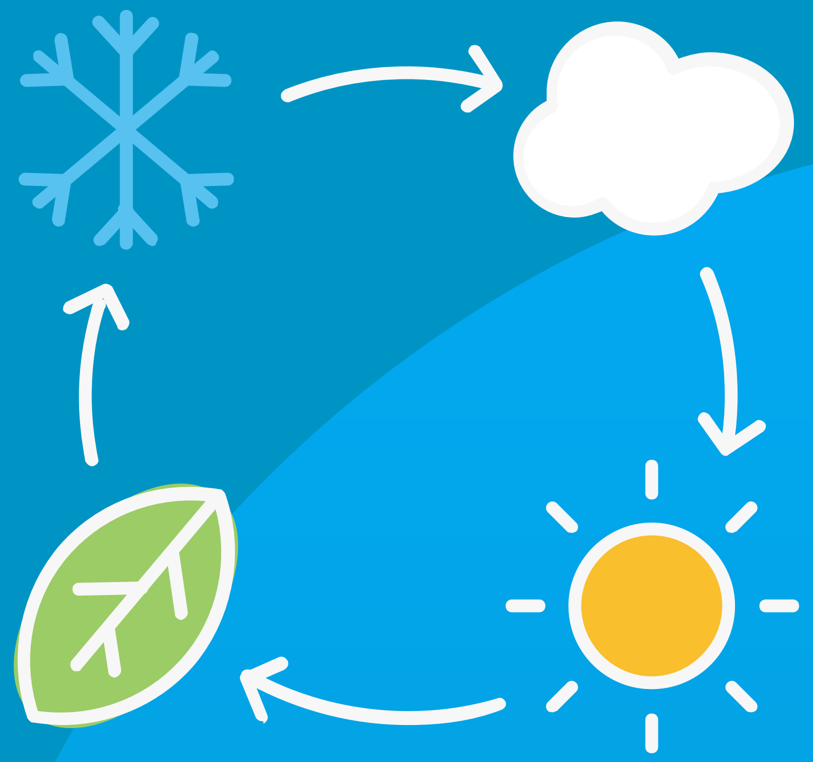
Patterns repeating at fixed and known time intervals

DAILY

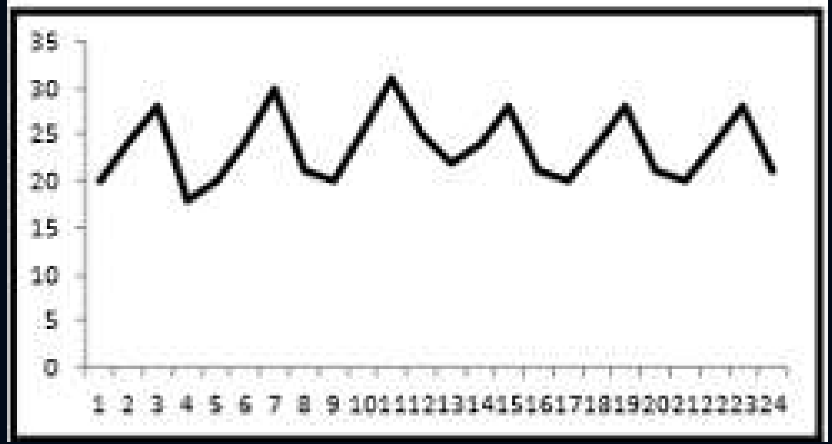
WEEKLY

MONTHLY

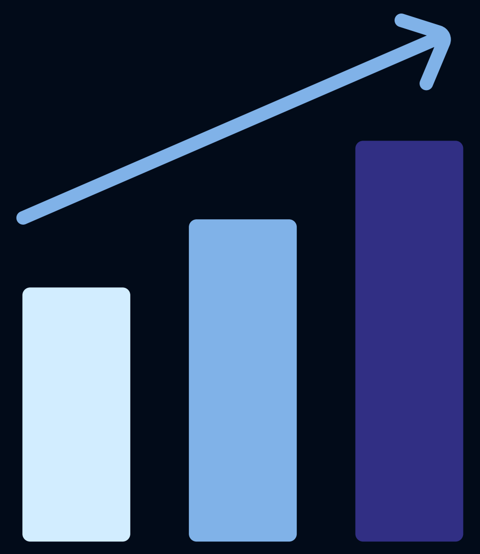
YEARLY



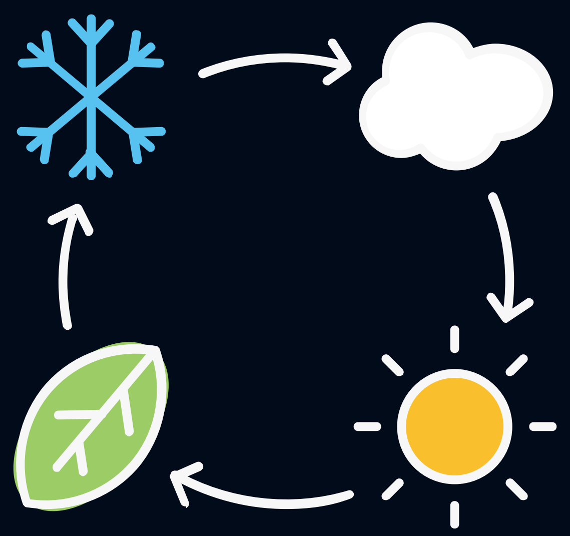
Timing of these patterns should be predictable!
(Usually within a year!)



Components of Time Series

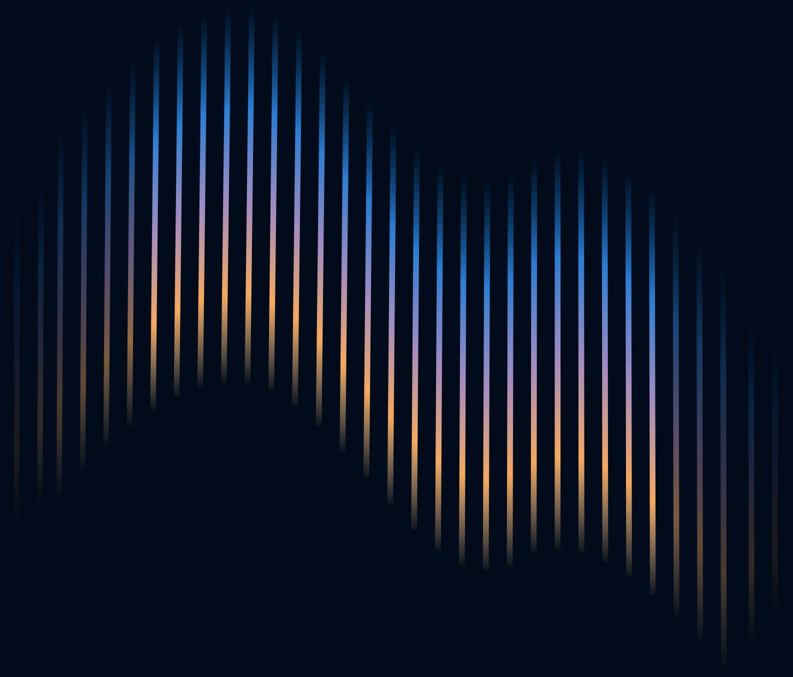


TREND



SEASONALITY

CYCLICAL



NOISE

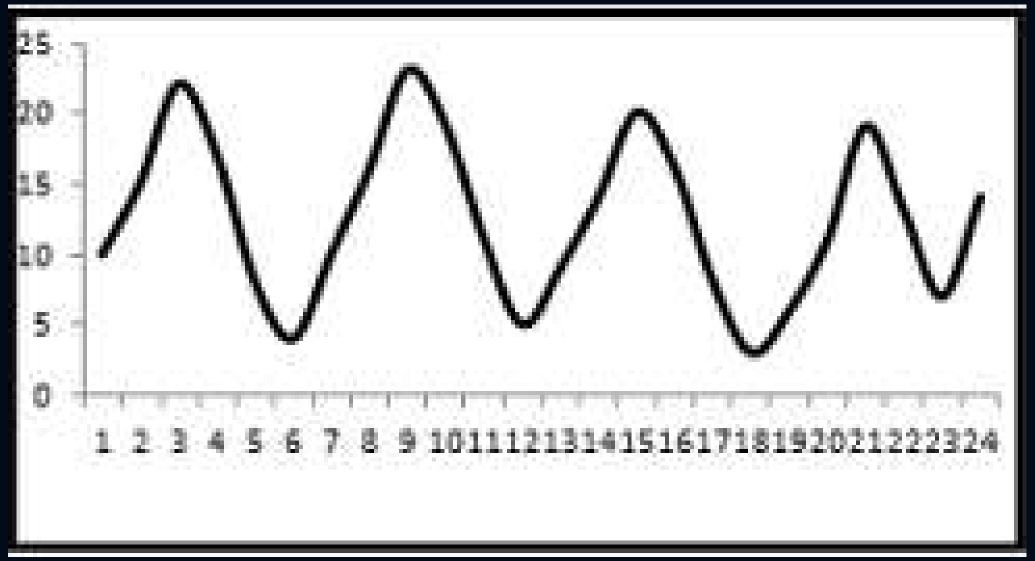
Cyclical

Medium Term Ups-And-Downs

ECONOMIC EXPANSION

ECONOMIC RECESSION

Movements do not follow fixed schedule (can last for few years or more)



Seasonality vs Cyclical

Seasonality

Fixed Timing

Within yearly calendar

Monthly
Quarterly
Weekly
Daily

Cyclical

Irregular Timing

Economic / Systemic

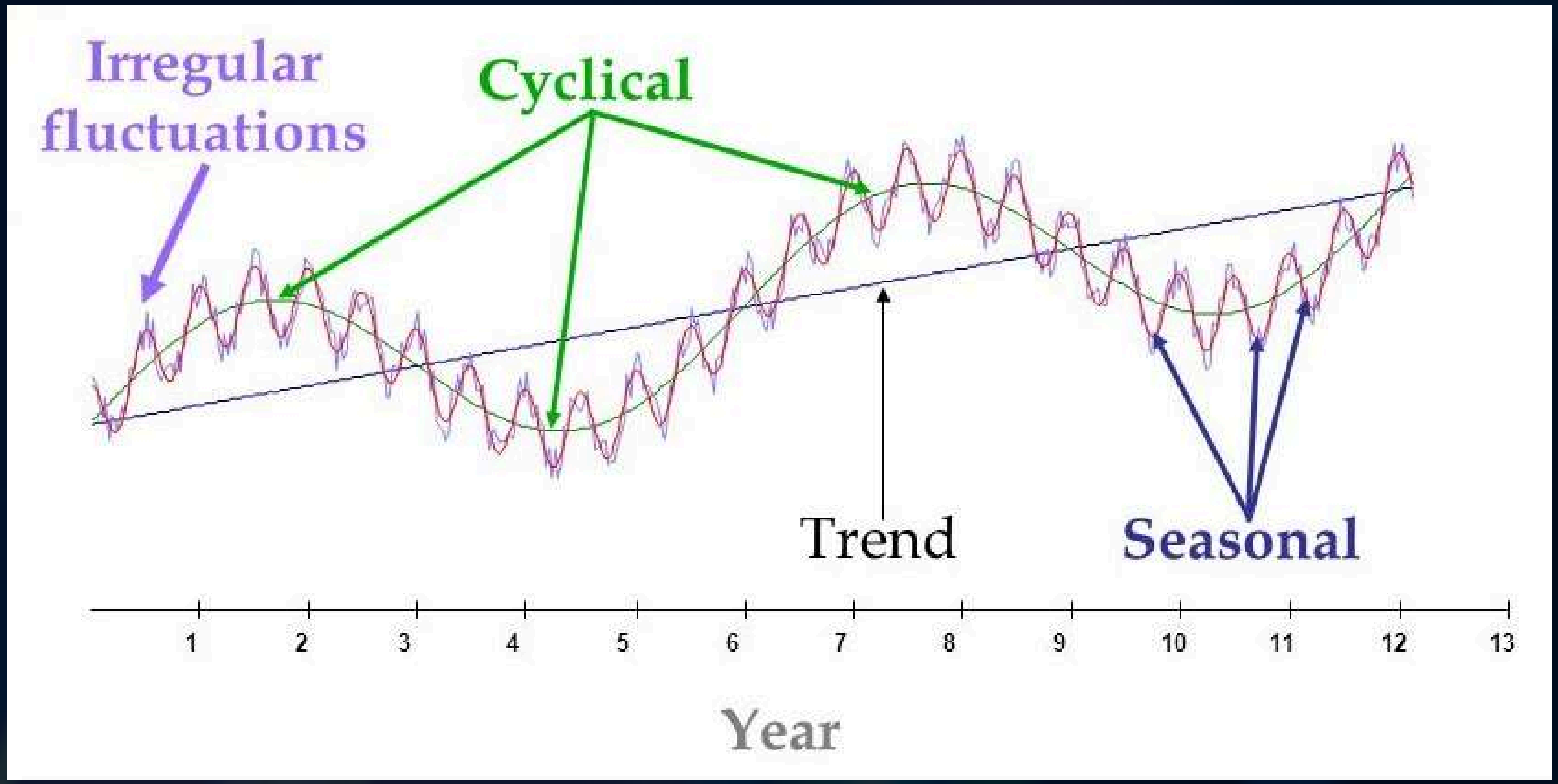
Several Years
(2 to 10 years)

Many models often have a combination of BOTH!

Seasonal + Cyclical

Changes within a year

Evenly spaced peaks and troughs



Much longer wavelengths

Irregular Spacing between peaks and troughs

Span Several Years

Components of Time Series

Noise

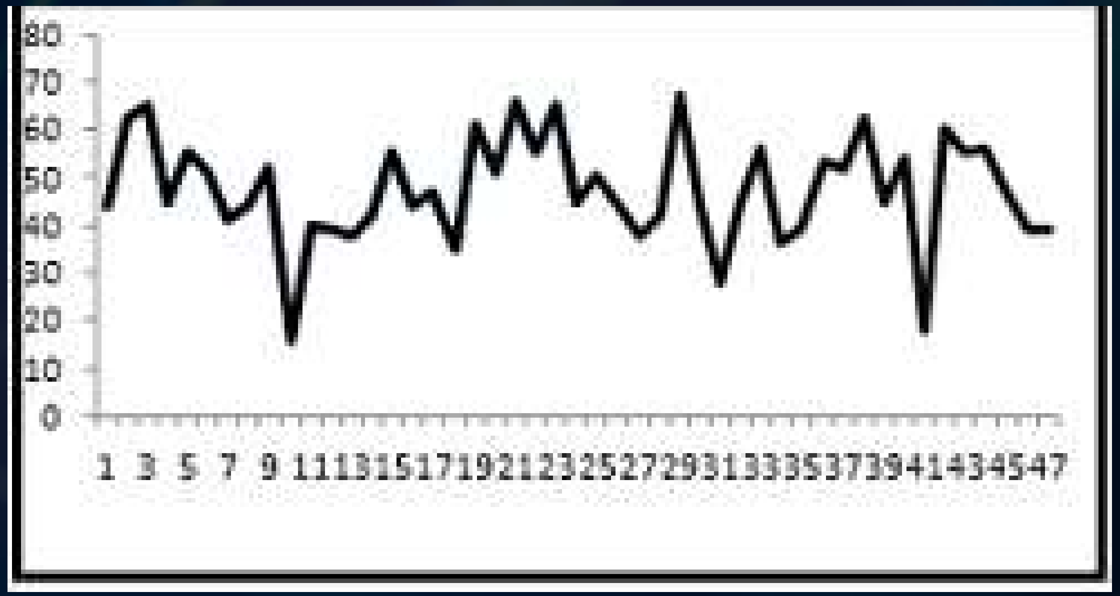
Irregular Component of Time Series

RANDOM SHOCKS

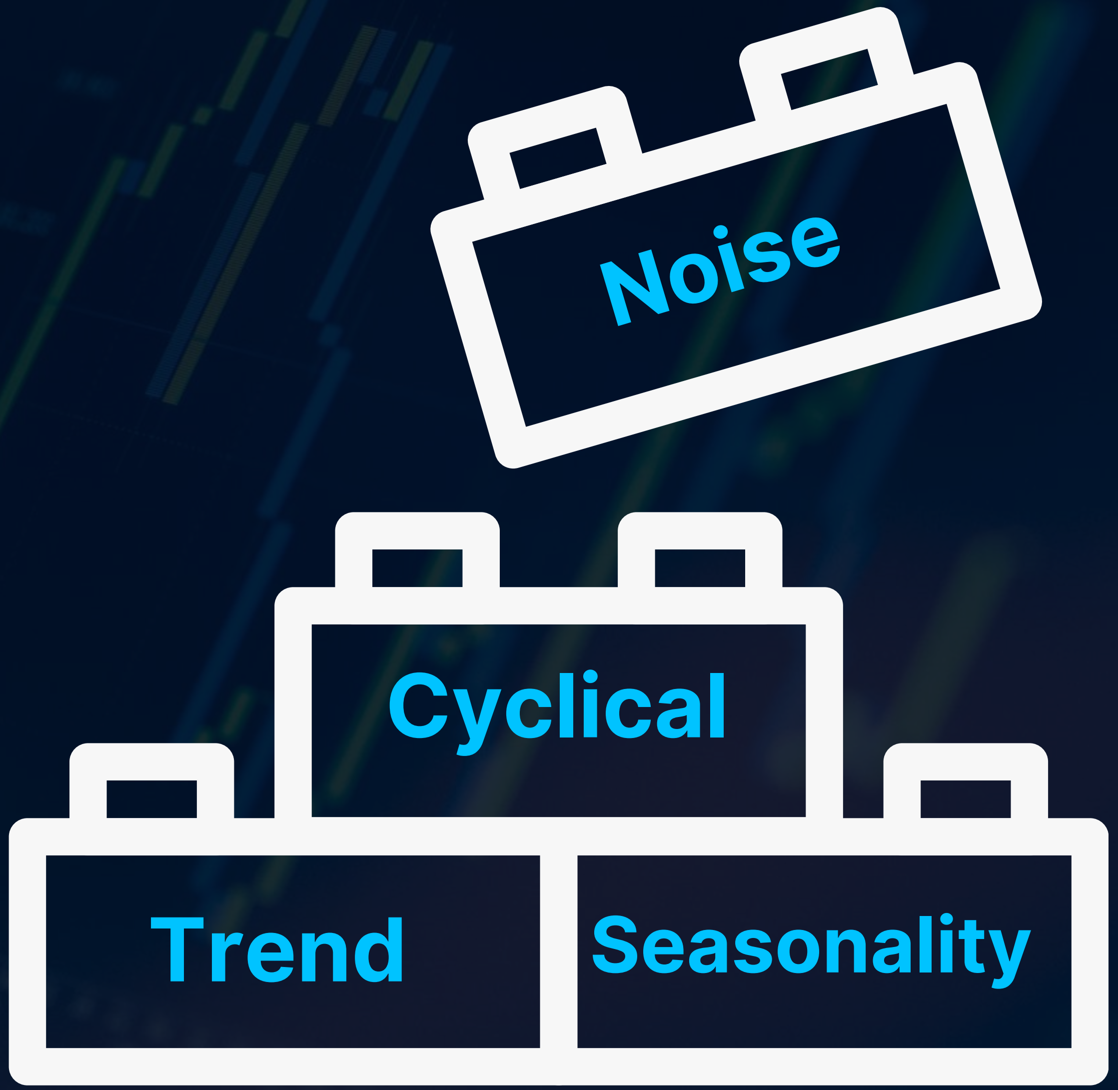
MEASUREMENT ERROR

ONE-OFF EVENTS

Movements that cannot be systematically predicted



Structure in Time Series



Rationale

Choice of Structure Types

Choices of Structure Types

Additive

Multiplicative

Additive Structure

Components add together

Additive Model

$$Y_t = T_t + S_t + C_t + I_t$$

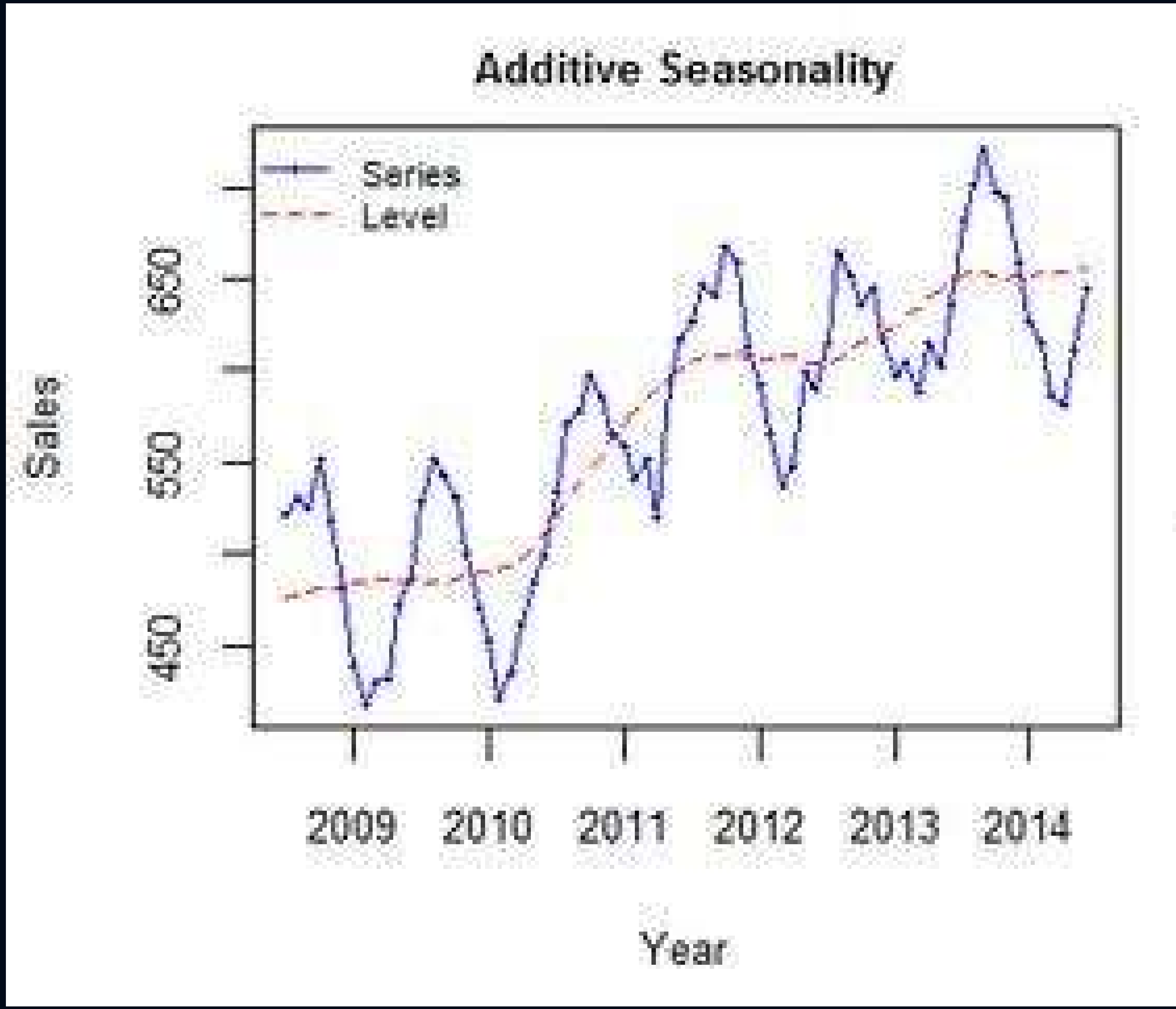
(when S_t & C_t *are independent of trend*)

TREND: OVERALL RISE / FALL

SEASONAL: CONSTANT PATTERN

CYCLICAL: CONSTANT SWING

NOISE: CONSTANT VARIATION



SALES OF A COMMERCIAL PRODUCT FROM 2009 TO 2014

Multiplicative Structure

Components scale with the level of the time series

Multiplicative Model

$$Y_t = T_t * S_t * C_t * I_t$$

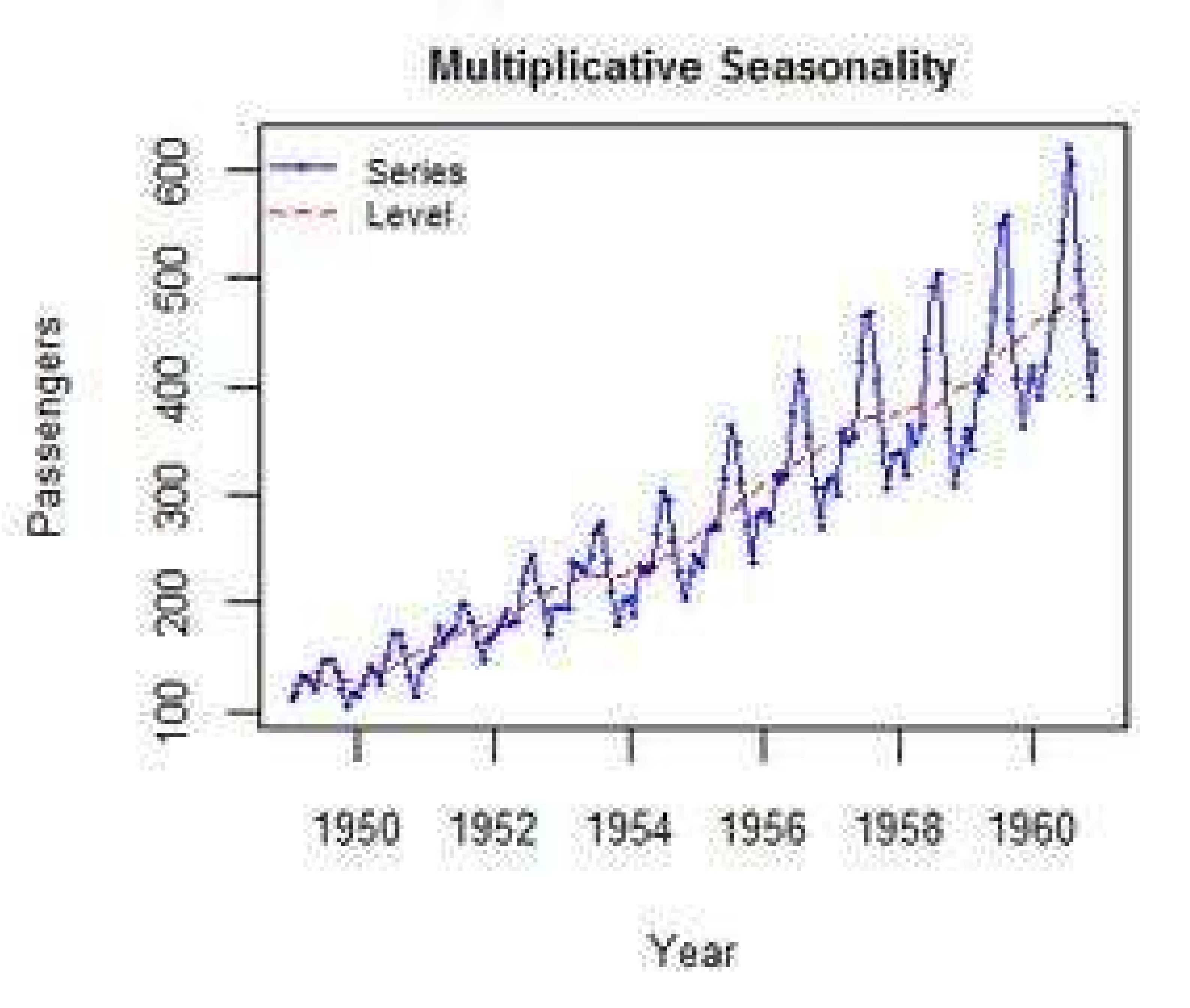
(when S_t & C_t *are dependent of trend*)

TREND: OVERALL RISE / FALL

SEASONAL: PROPORTIONAL PATTERN

CYCLICAL: VARIABLE SWINGS

NOISE: PROPORTIONAL TO LEVEL



INTERNATIONAL AIR PASSENGER ARRIVALS (1950 TO 1960)

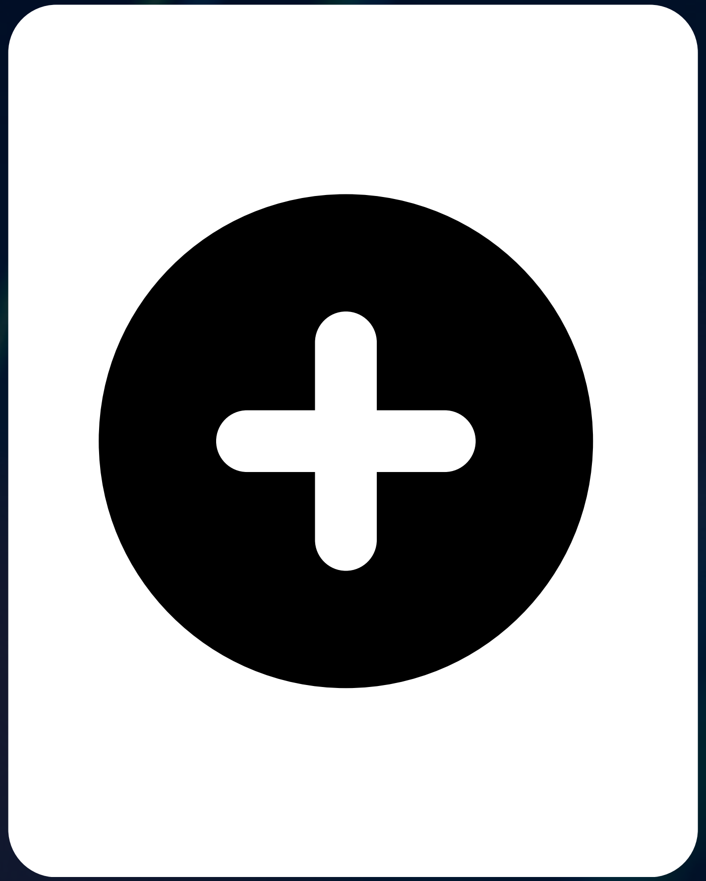
Important Detail(s)

- Quarterly U.S. Real GDP (GDPC1), inflation-adjusted
- Source: FRED, <u>seasonally adjusted</u>

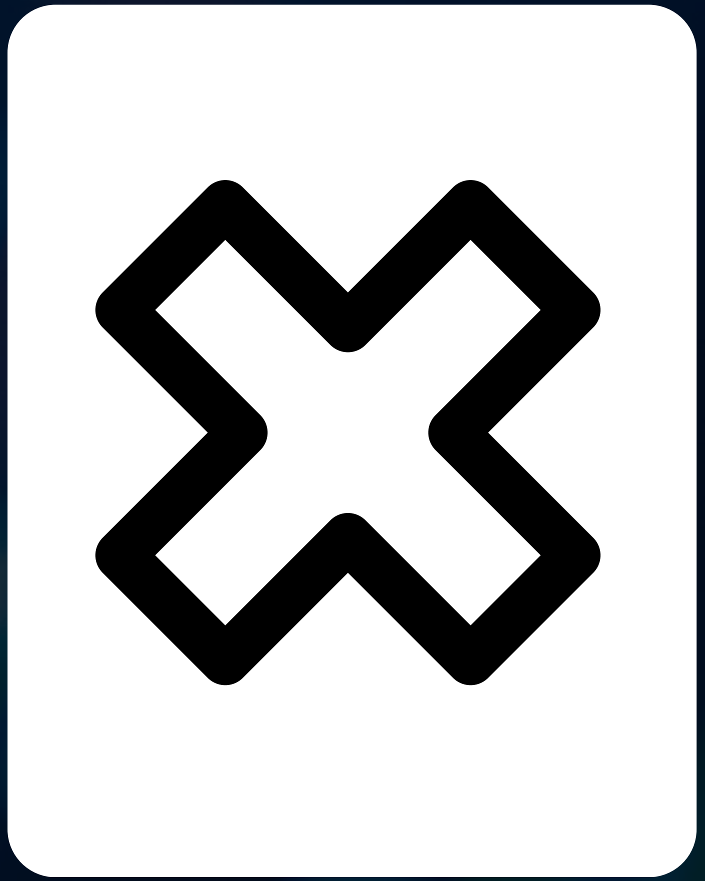
Trend | Se ty | Cyclical | Noise

Choices of Structure Types

Additive

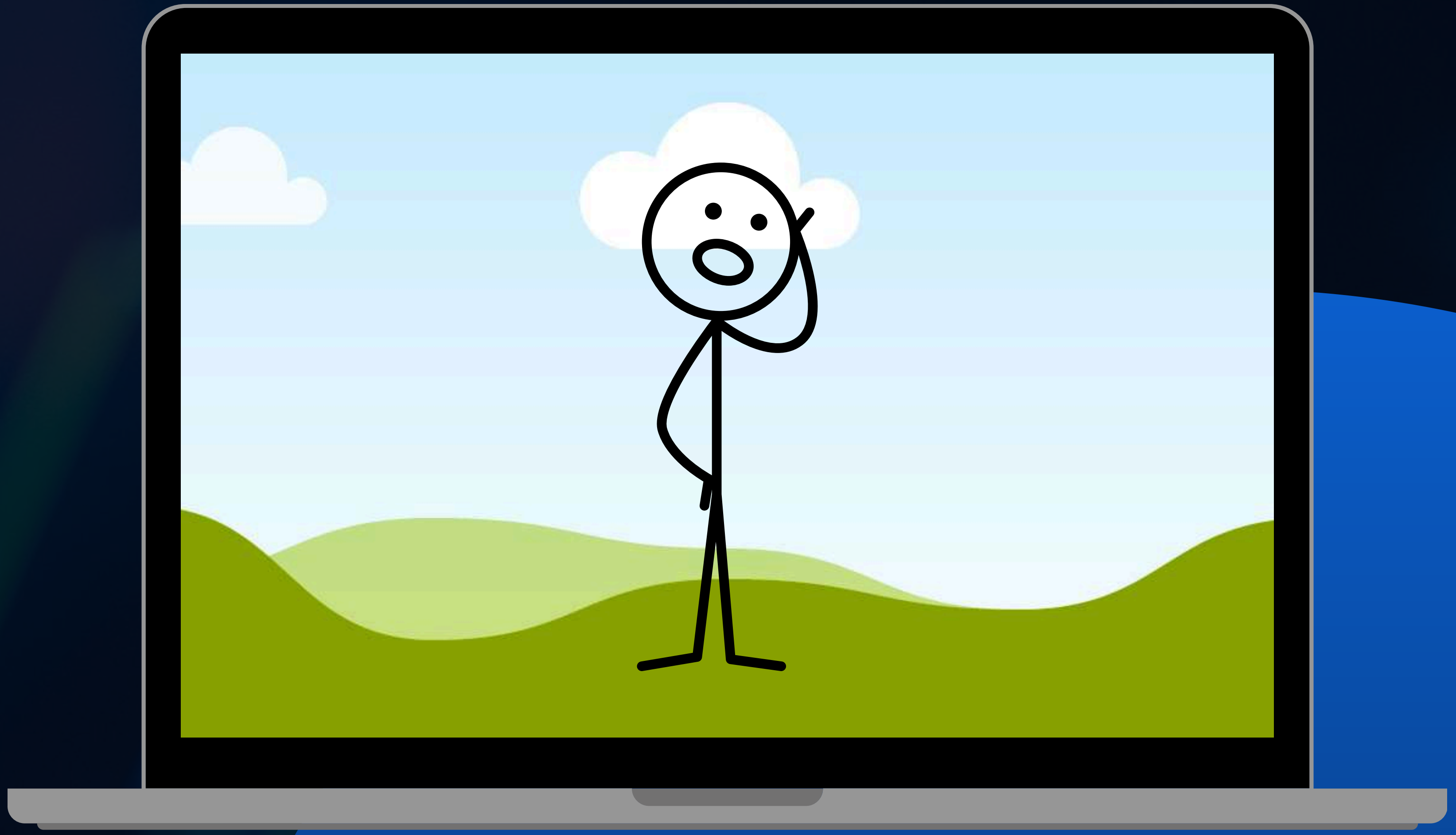


Multiplicative

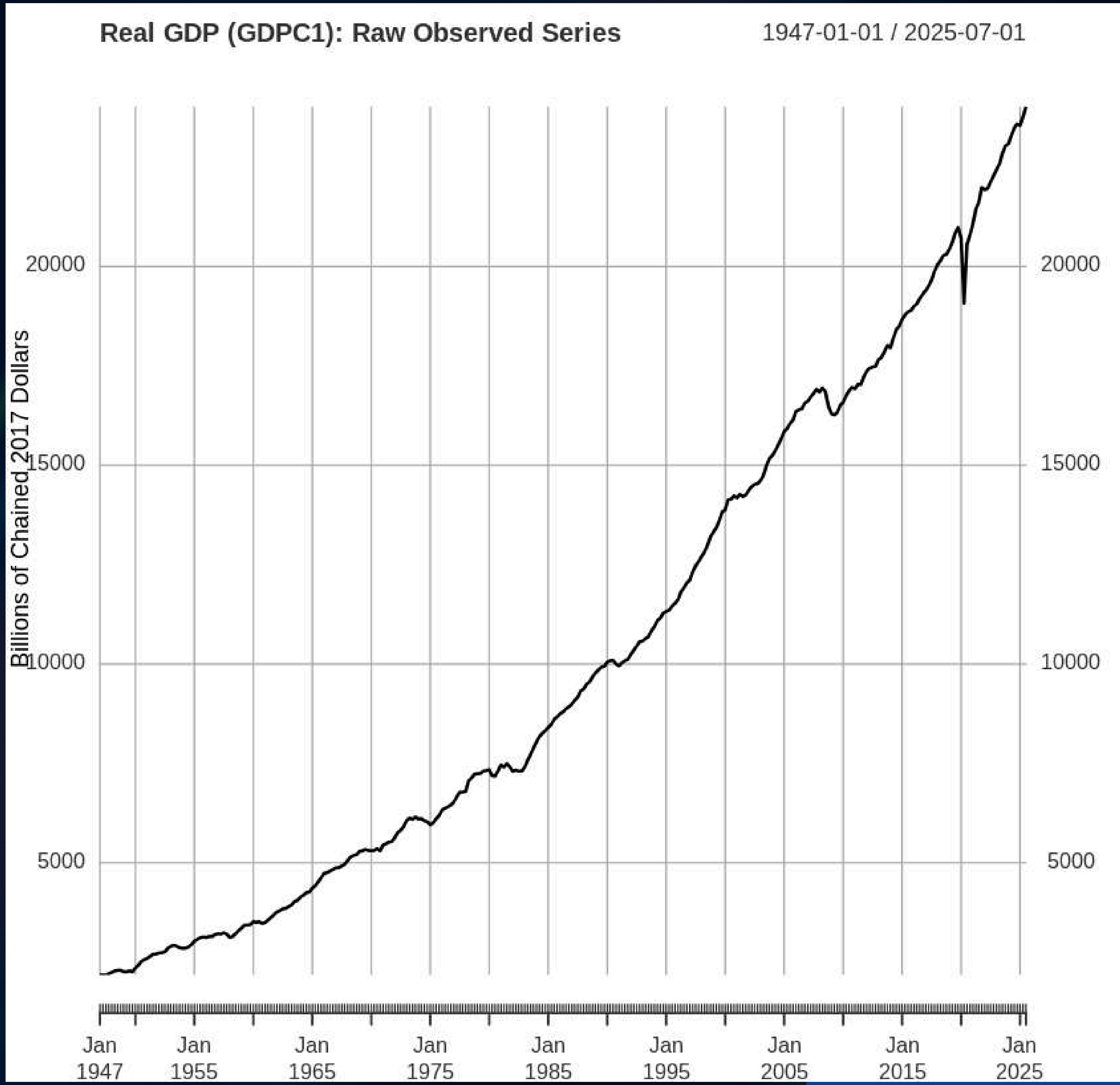


How to decide which structure?

- INSPECT THE RAW TIME SERIES
- CHECK THE SIZE OF DEVIATIONS FROM TREND
- CHOOSE MODEL THAT FITS CHANGES BETTER



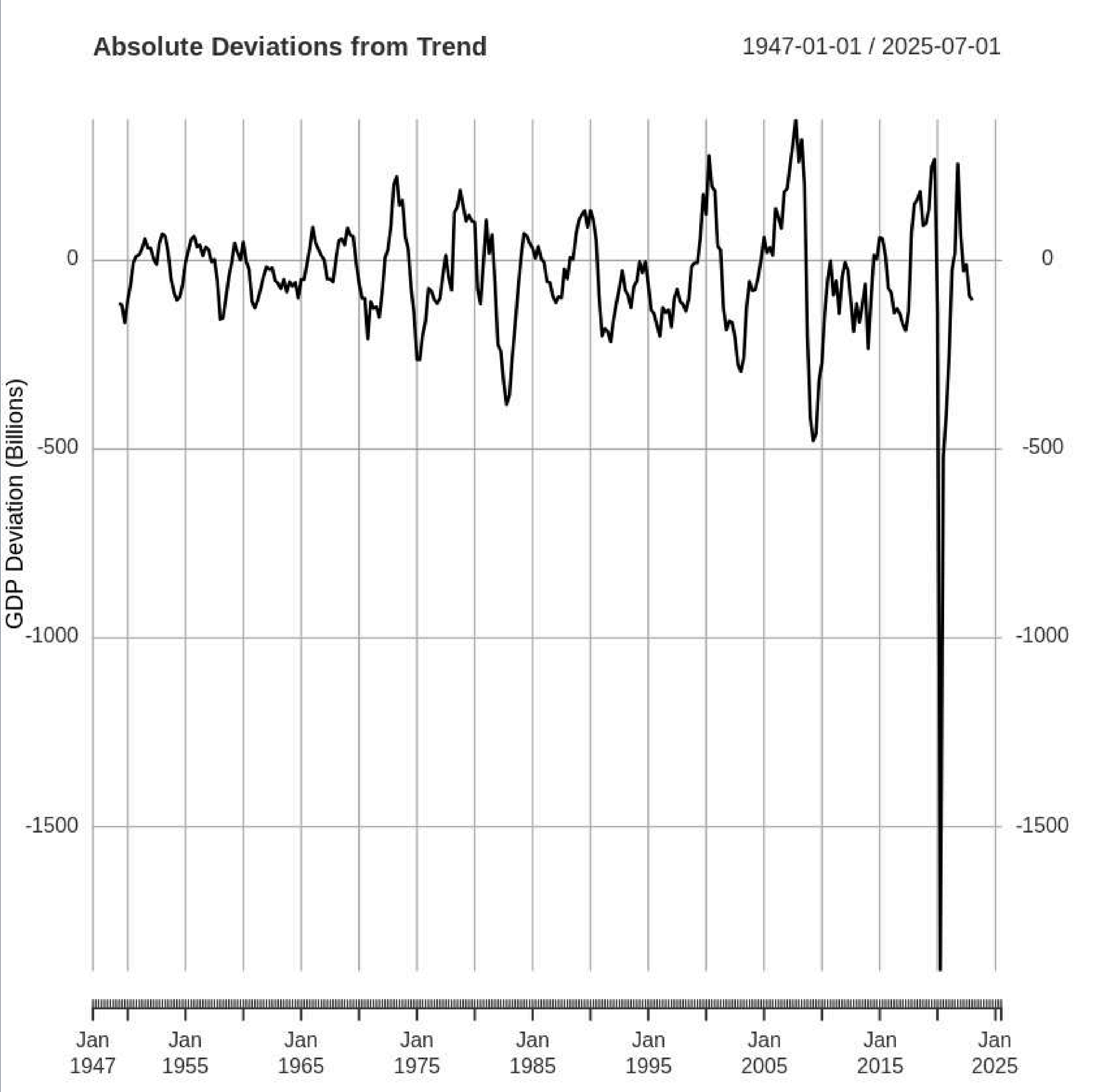
INSPECT THE RAW TIME SERIES



CHECK SIZE OF DEVIATIONS FROM TREND

Trend: General increase

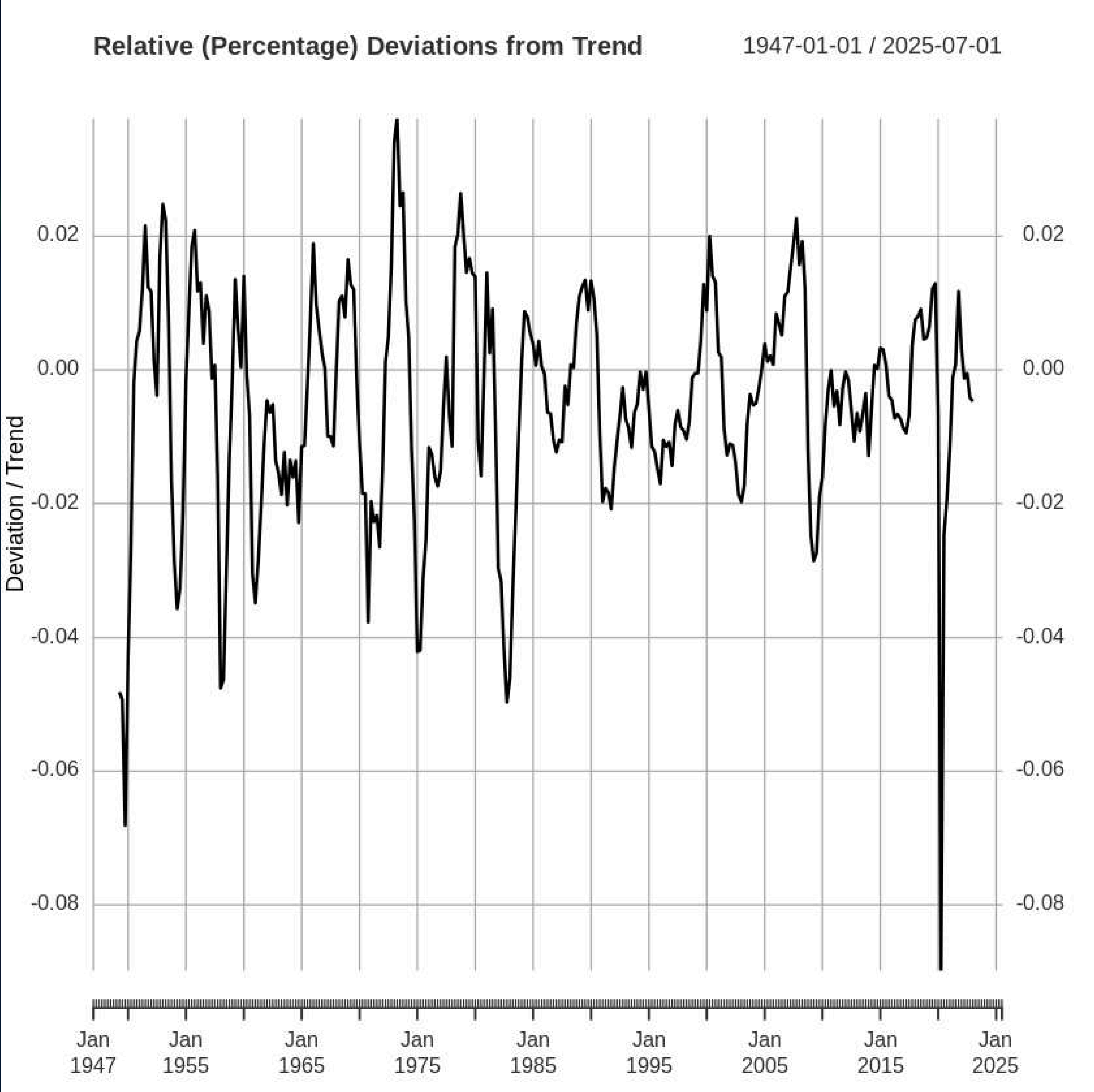
Deviations from Trend increased as level of series increases



CHECK SIZE OF DEVIATIONS FROM TREND

However, Percentage deviations from trend remain constant

because size of fluctuations is proportional to the level of series



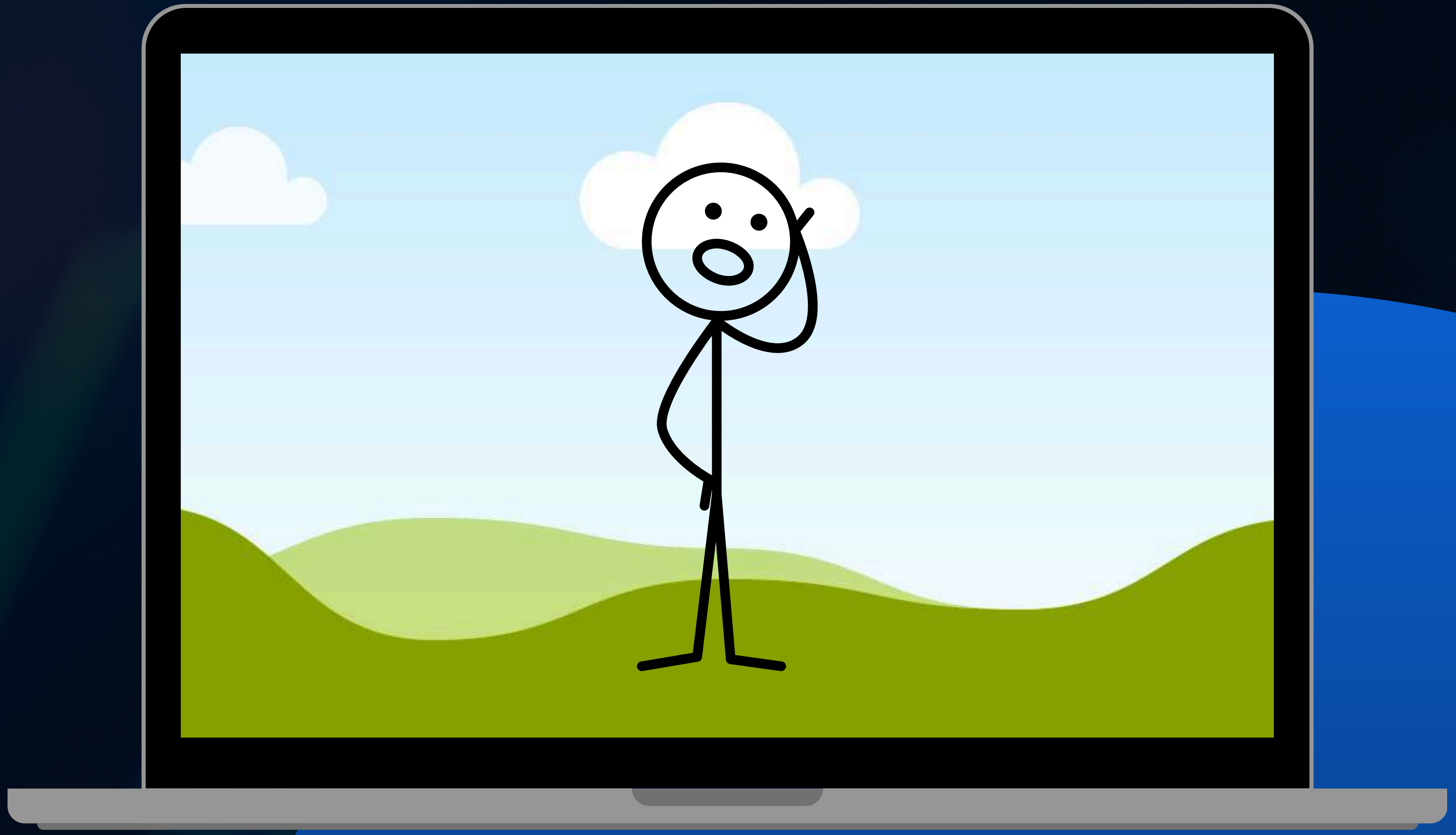
$$\frac{Y_t - T_t}{T_t}$$

How to decide which structure?

CHOOSE MODEL THAT FITS CHANGES BETTER

Additive

Multiplicative



Key Strategy

Use of log-transformation

$$Y_t = T_t \times S_t \times C_t \times I_t$$

$$\log Y_t = \log T_t + \log S_t + \log C_t + \log I_t$$

$$\log Y_t - \log T_t = \log\left(\frac{Y_t}{T_t}\right)$$

Reasons

- Convert proportional changes into additive ones
- Stabilise the size of fluctuations over time
- Make individual components easier to identify and compare

Components of Time Series

TREND

SEASONALITY

CYCLICAL

NOISE

Trend (Demo)

Long Run Direction of Time Series

```
log_trend <- rollmean(log_gdp, k = 40, align = "center", fill = NA)
```

Replace each point with a LOCAL AVERAGE of nearby observations

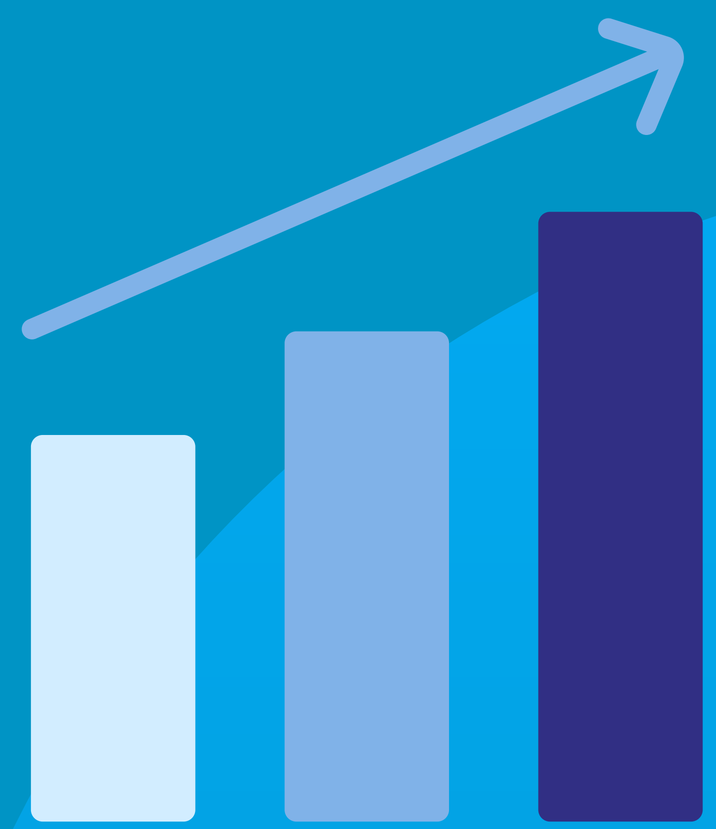
window of 40 observations

40 quarters = 10 years

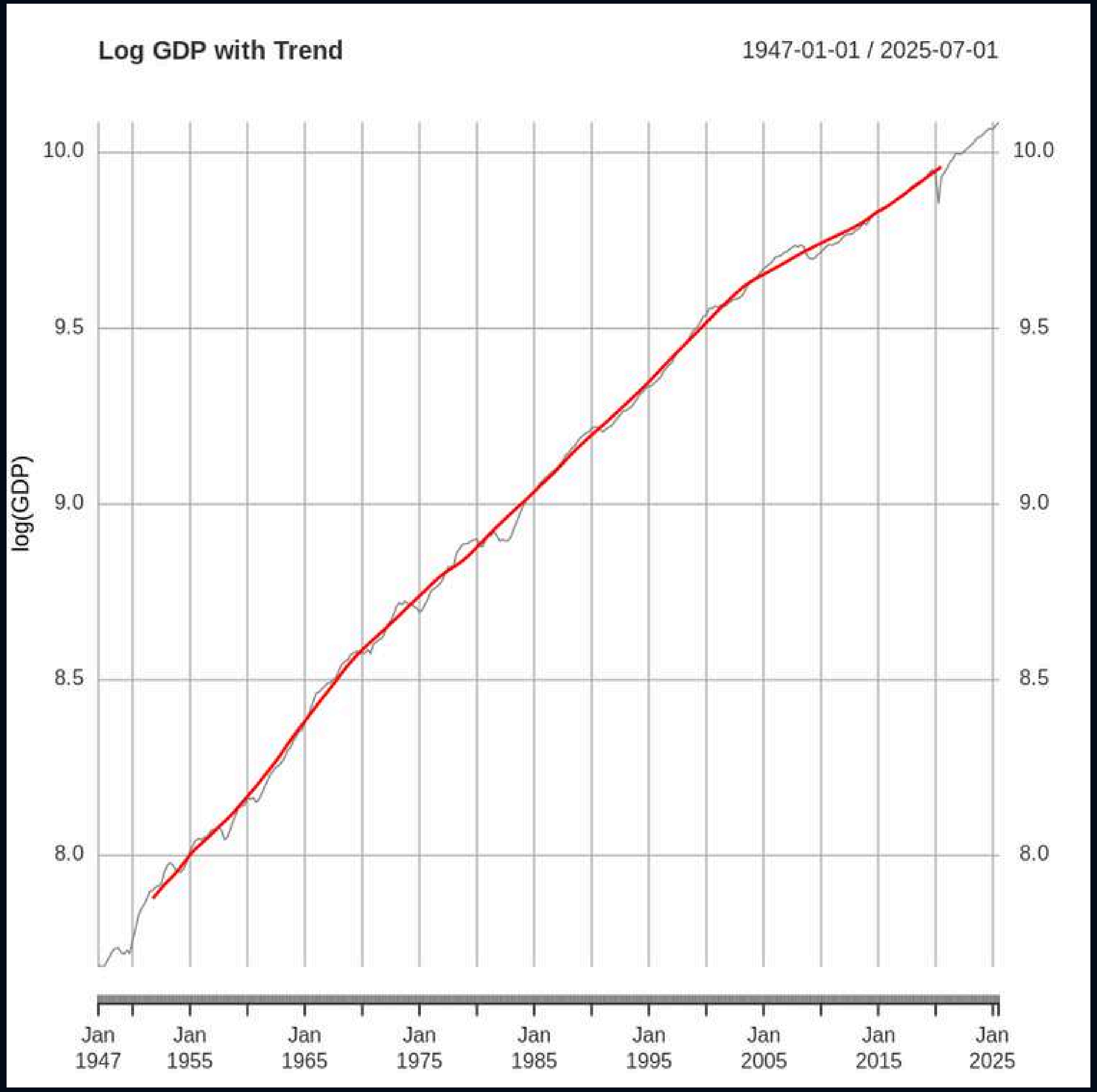
assign the average to the middle of the window

Fill values at the start and end with NA where a centered average cannot be computed

By averaging, we smooth out short-term noise and medium-term fluctuations



Trend (Demo)

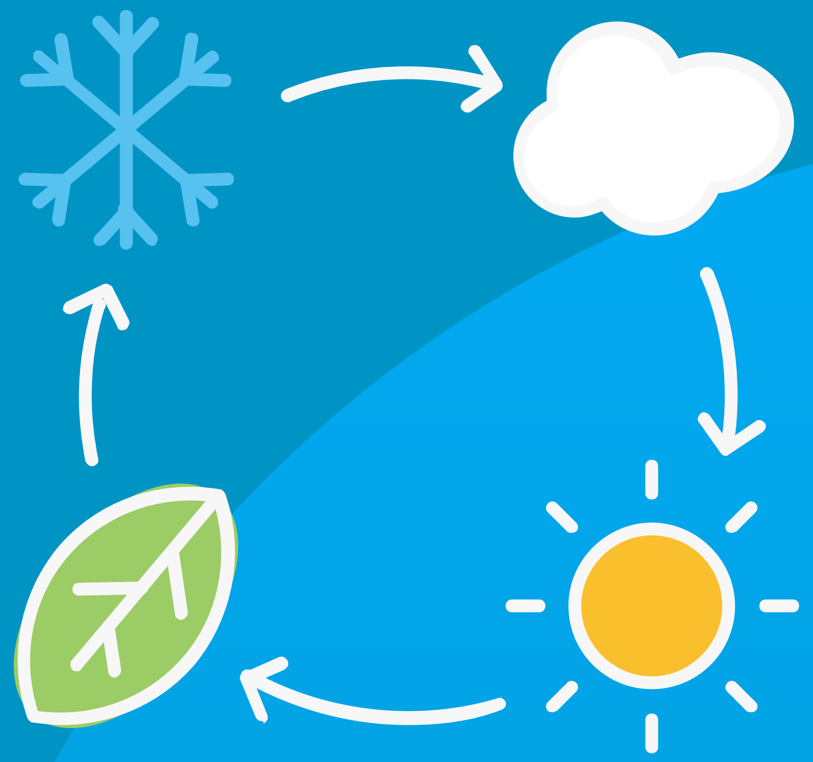


Log GDP with Trend
1947-01-01 / 2025-07-01
log(GDP)
10.0
9.5
9.0
8.5
8.0
Jan 1947
Jan 1955
Jan 1965
Jan 1975
Jan 1985
Jan 1995
Jan 2005
Jan 2015
Jan 2025

Seasonality

Patterns repeating at fixed and known time intervals

NOT APPLICABLE TO THIS DATASET!!!

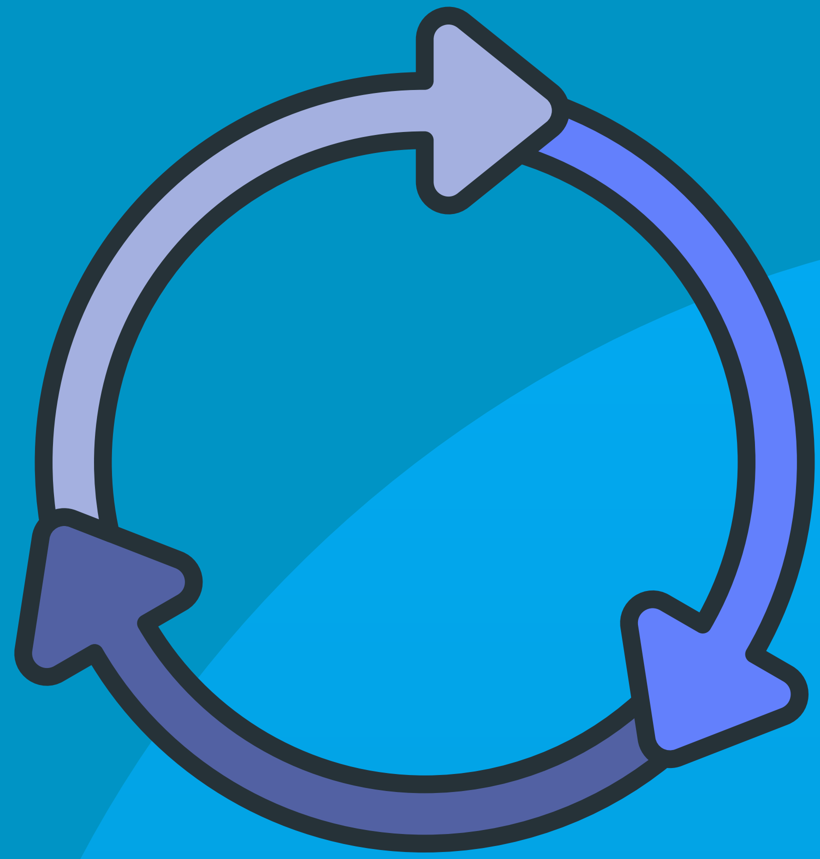


Cyclical (Demo)

Medium Term Ups-And-Downs

```r
########## Log Deviations (Cycles + Noise) ##########
log_dev <- log_gdp - log_trend
```

$\log T_t + \log C_t + \log I_t$ $\quad$ $\log T_t$



Cyclical (Demo)

Medium Term Ups-And-Downs

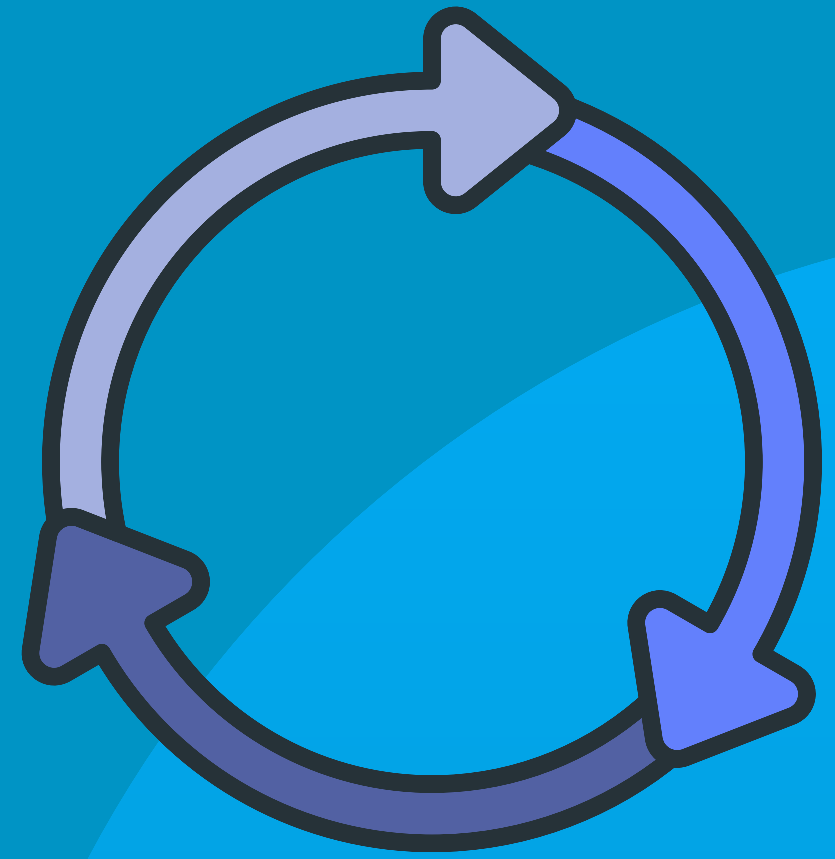
```
cycle <- rollmean(log_dev, k = 8, align = "center", fill = NA)
```

Replace each point with a LOCAL AVERAGE of nearby observations

Window of 8 Observations = 8 Quarters = 2 Years

assign the average to the middle of the window

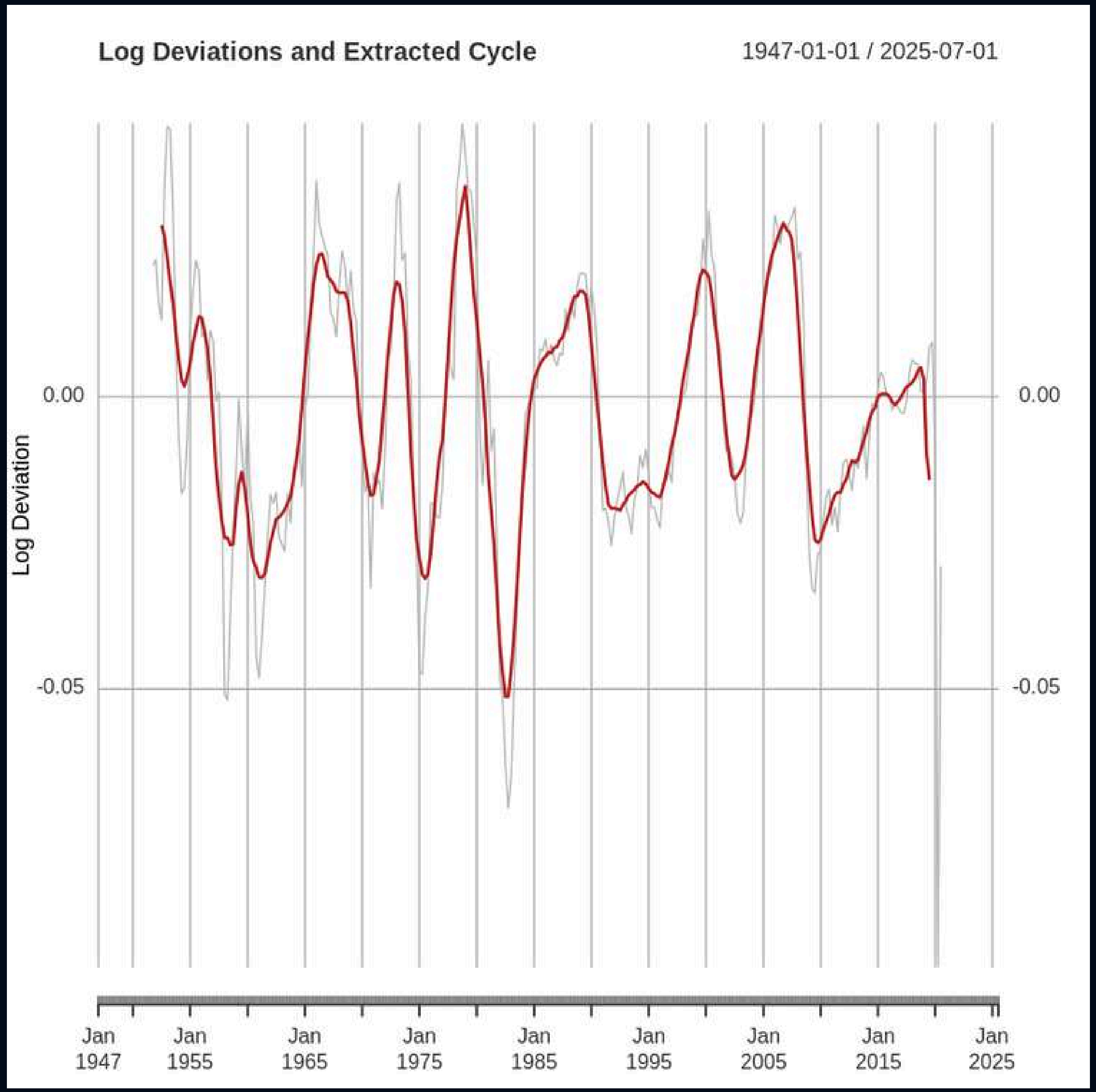
Fill values at the start and end with NA where a centered average cannot be computed

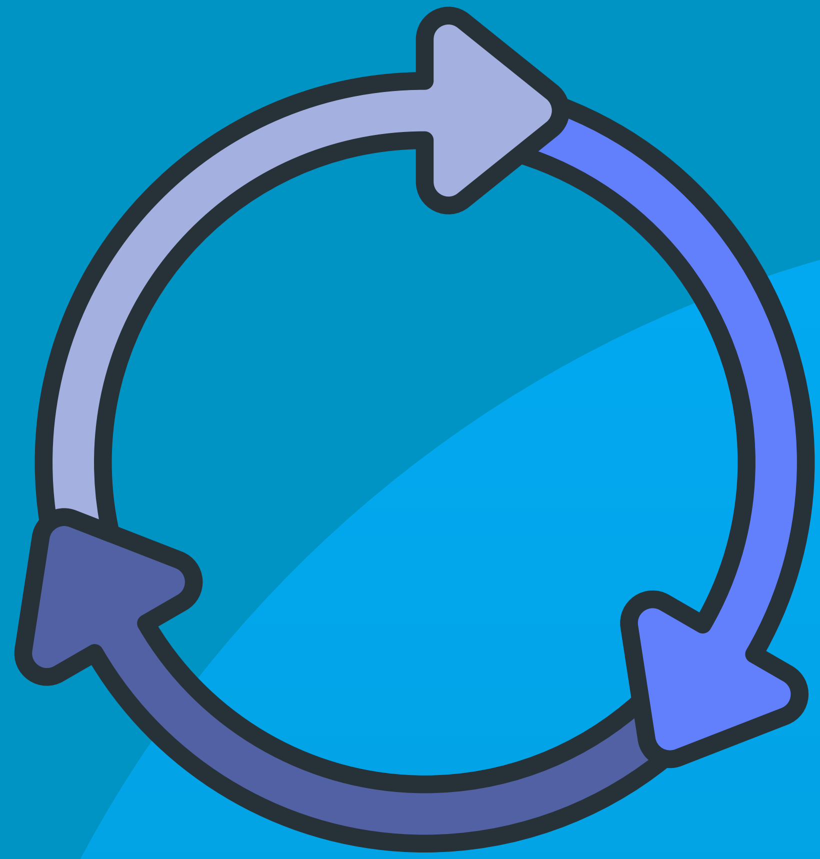


By averaging, we smooth out short-term fluctuations (noise)

Cyclical (Demo)

Medium Term Ups-And-Downs





$$\log Y_t - \log T_t = \log C_t + \log I_t$$

Noise (Demo)

Irregular Component of Time Series

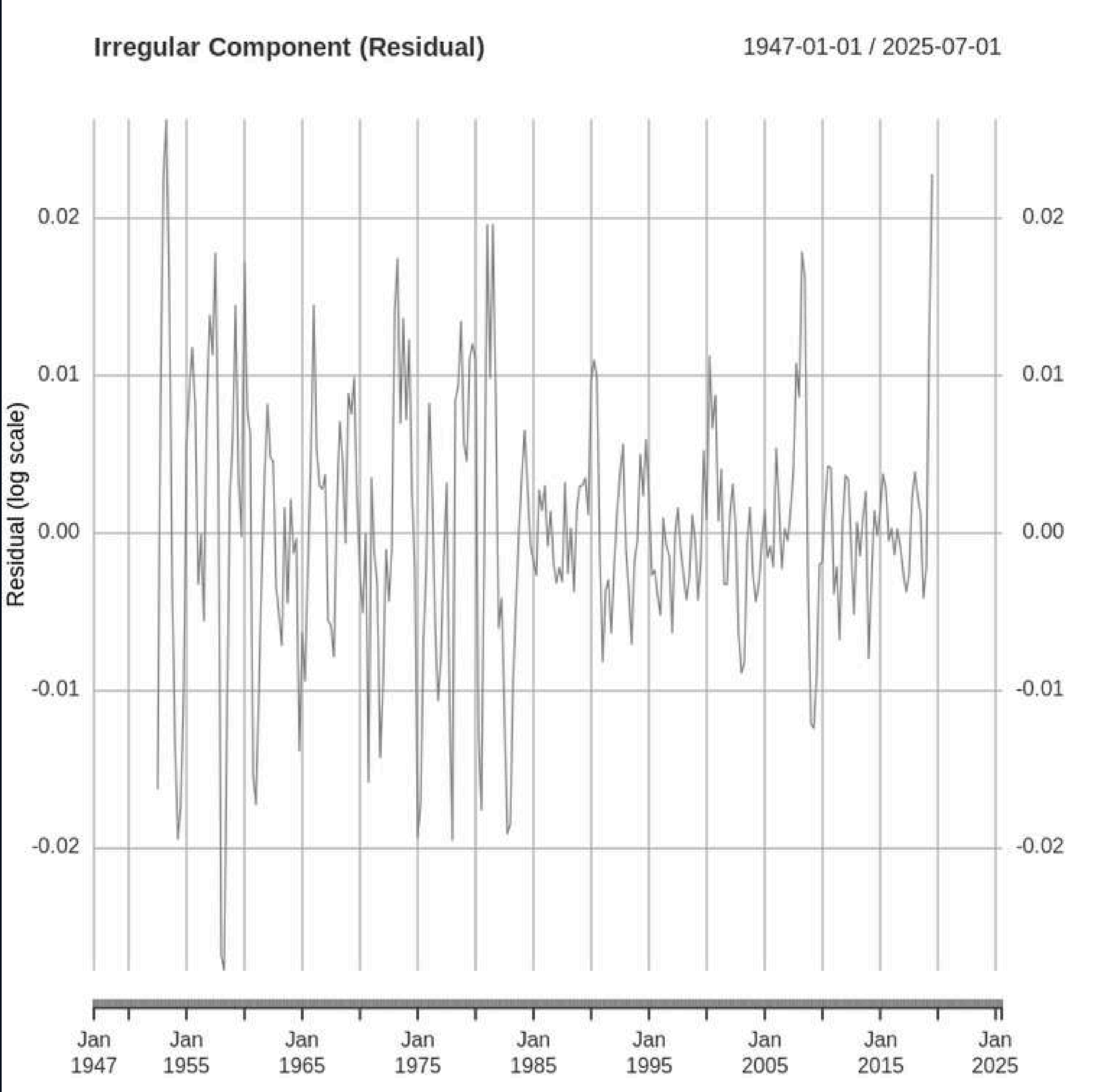
```
### Noise: residual
noise <- log_dev - cycle
```

$\log C_t + \log I_t$

$\log C_t$

Noise (Demo)

Irregular Component of Time Series



Reconstructing Multiplicative Structure

```
########## Multiplicative Components (Absolute Scale) ##########
trend_abs <- exp(log_trend)
cycle_abs <- trend_abs * (exp(cycle) - 1)
noise_abs <- trend_abs * (exp(noise) - 1)

### Reconstructed GDP
GDP_fitted <- exp(log_trend + cycle + noise)
```

Interpret components in original units

Verify use of Multiplicative Structure

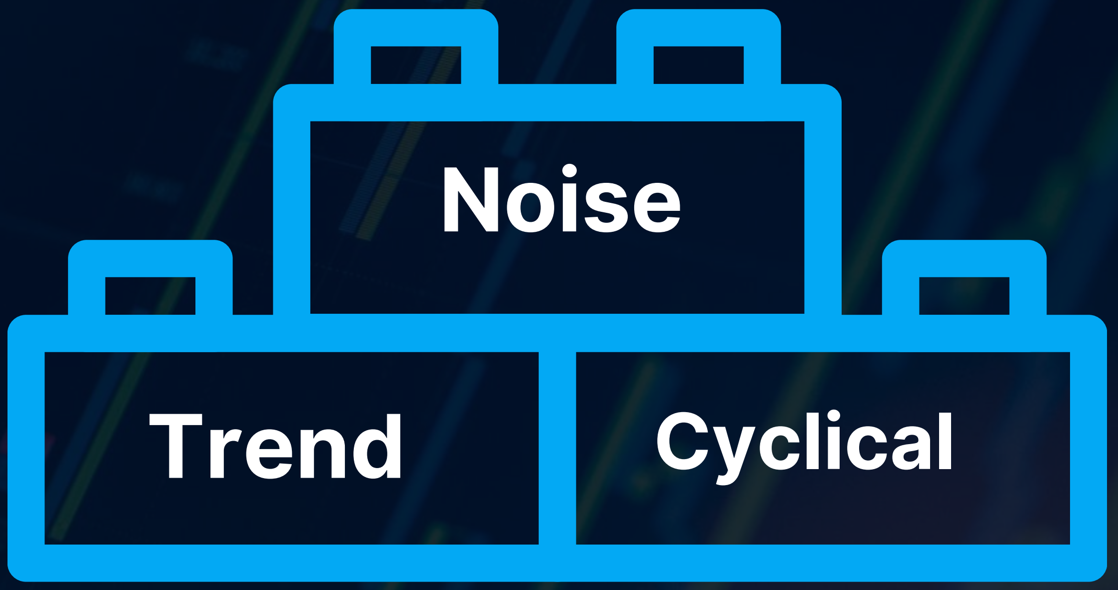
$$\hat{Y}_t = \hat{T}_t \times \hat{C}_t \times \hat{I}_t \approx Y_t$$

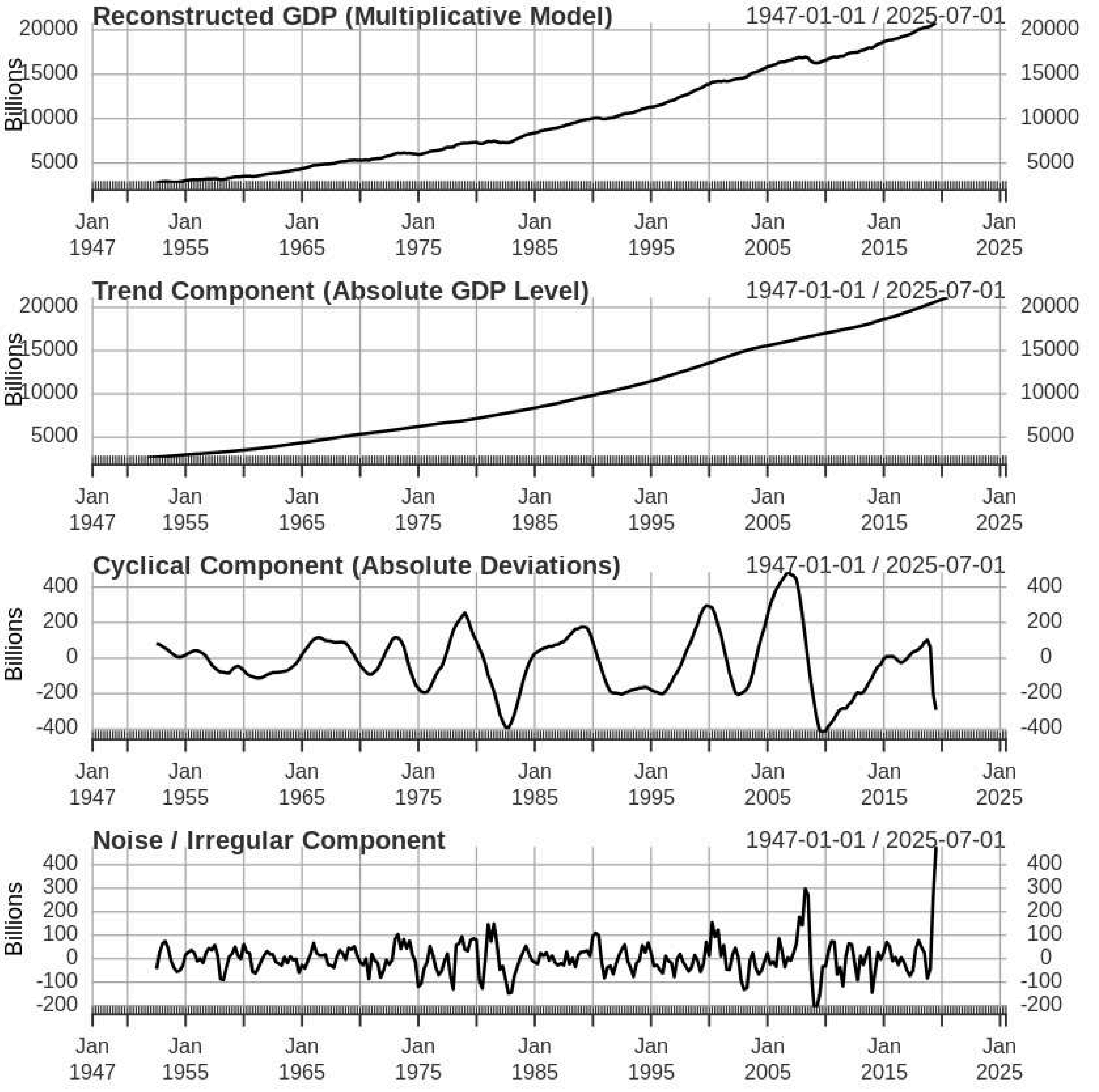
$$\log Y_t = \log T_t + \log C_t + \log I_t$$

$$\Big\downarrow \ \mathbf{exp}$$

$$Y_t = T_t \times C_t \times I_t$$

Bringing it all together





Why identify the structure?

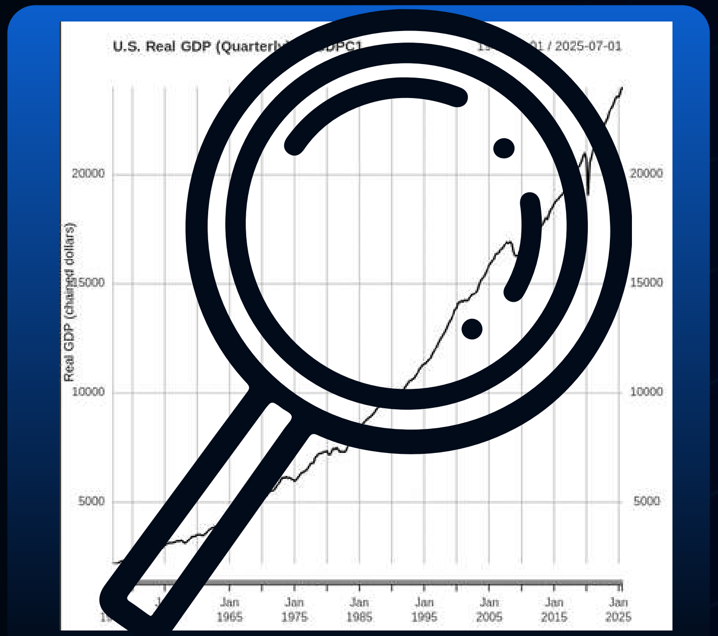
Multiplicative structures are non stationary (Mean, Variance change over time)

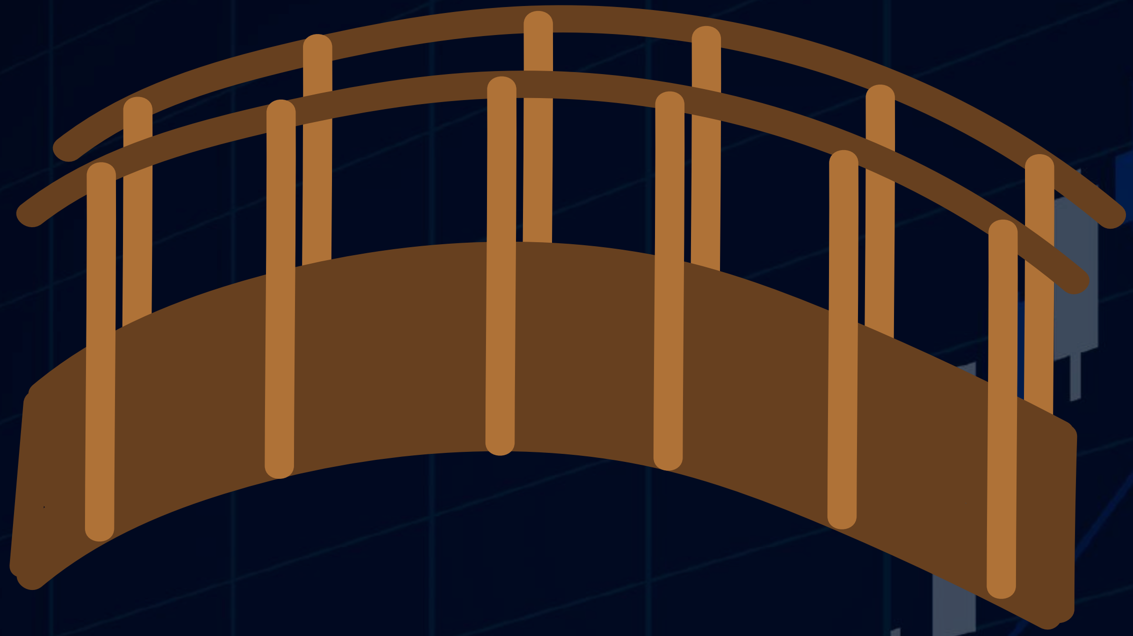
ARIMA requires stationarity (assumes mean, variance remain constant)

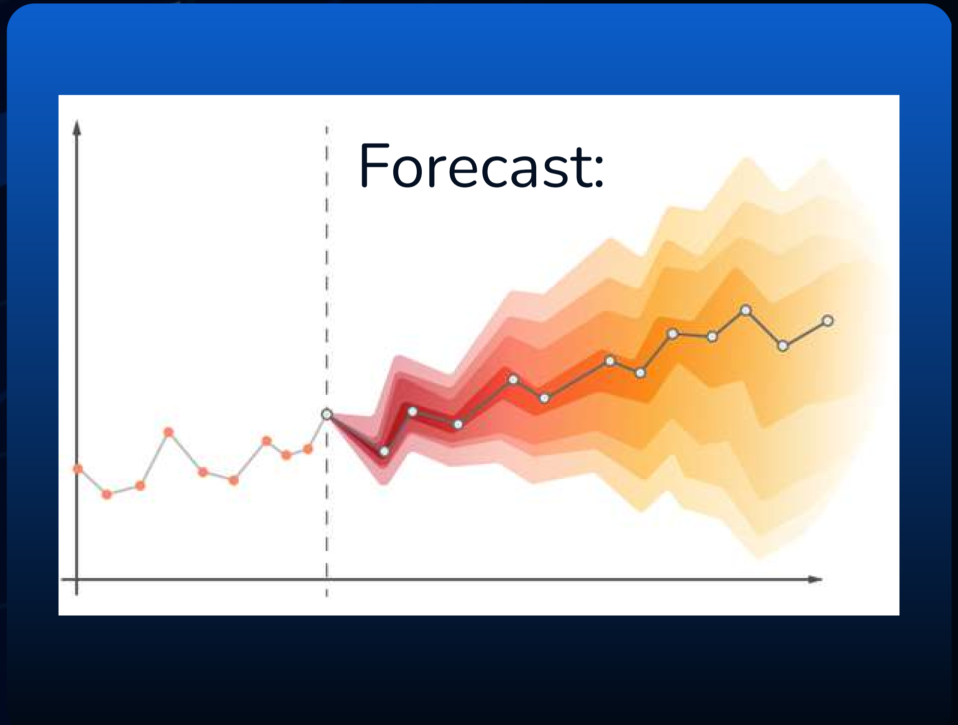
log-transform needed before ARIMA

Yan Zhe

Bridging the gap







Understanding Time Series Data

Preparation for modelling

- Stationarity
- Test for stationarity
- Time series transformation
- Tools to aid model selection

Time Series Modelling

Stationarity

What is stationarity?

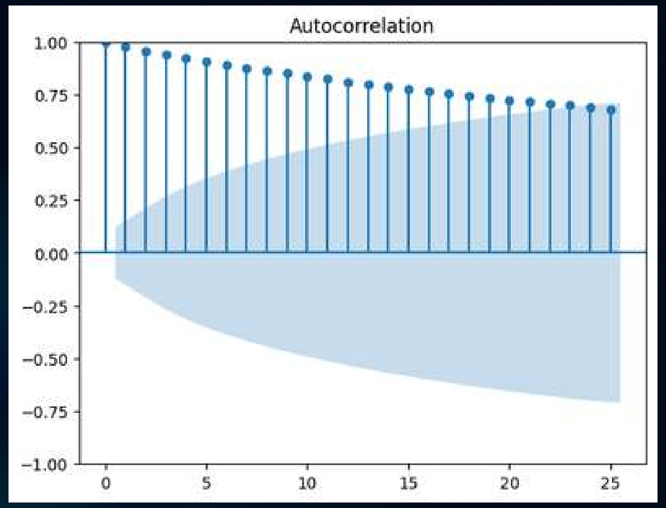
Statistical properties are constant over time

$\mathbb{E}$

Constant mean

σ^2

Constant variance

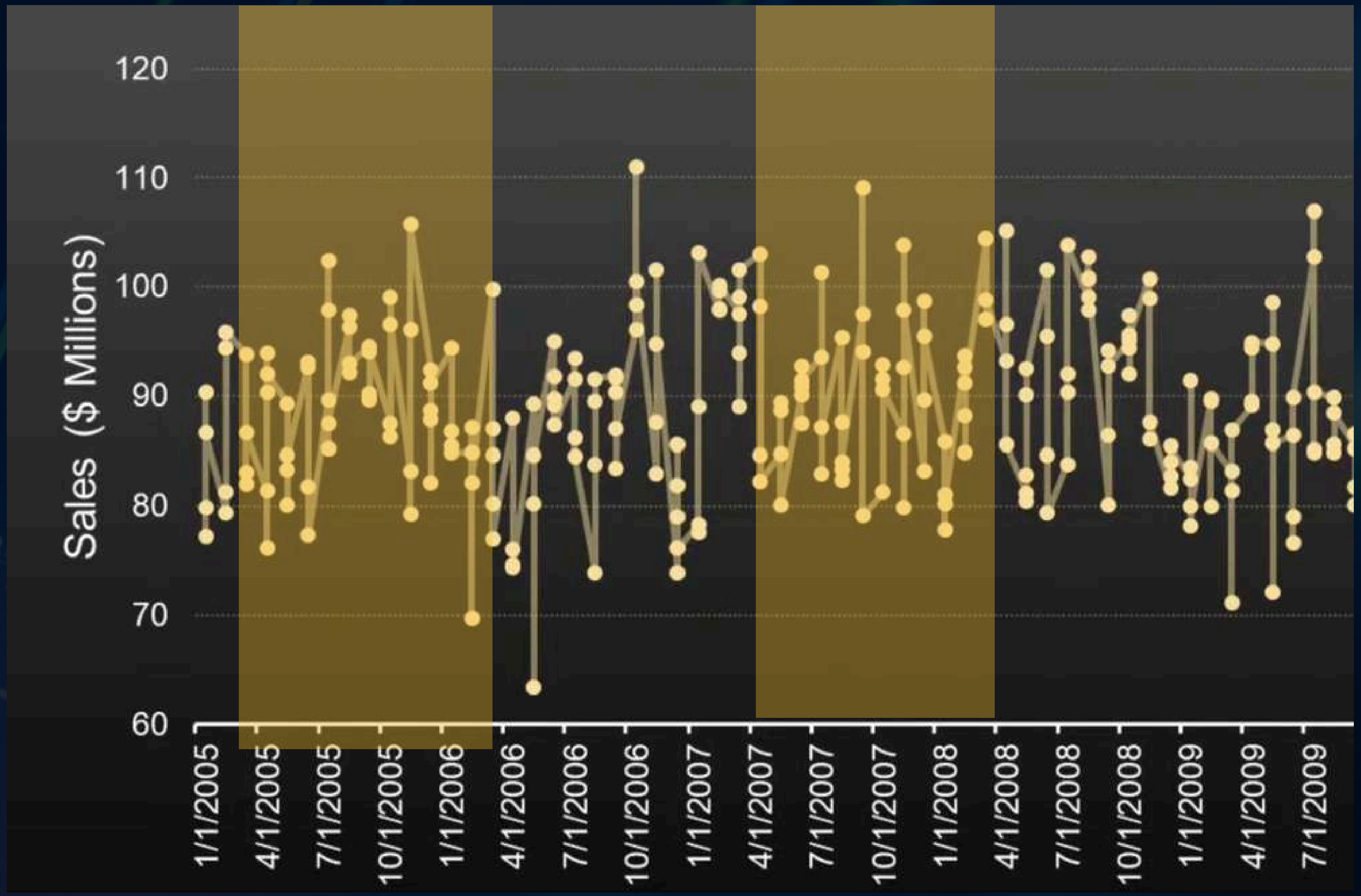


Constant autocorrelation

Purpose of stationarity

Consistency of Distribution

- Mean, Variance & Autocorrelation depend only on difference in time and **not location in time**



- Certain modelling techniques assume stationarity
- Important to ensure stationarity before attempting those approaches

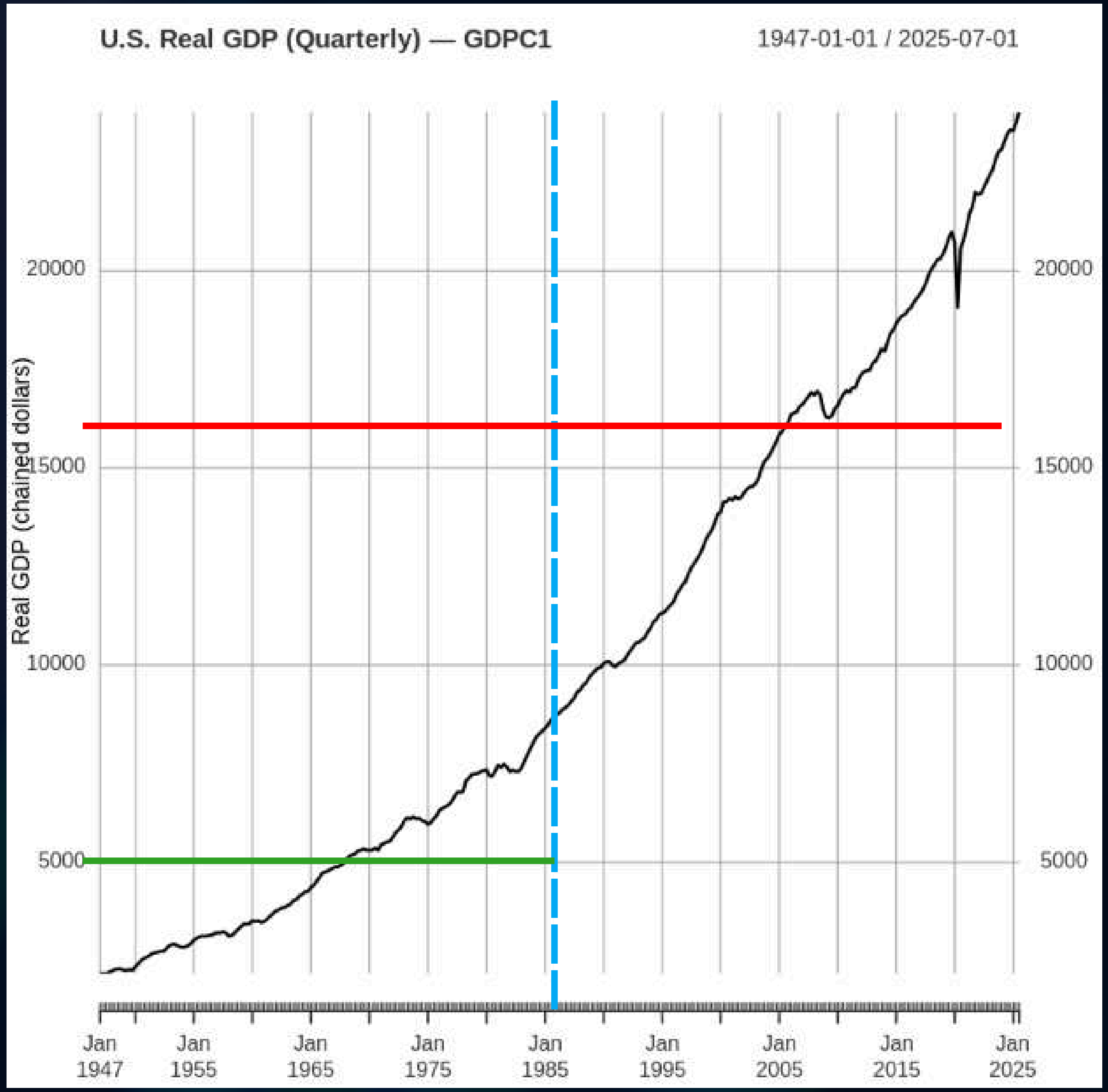
Types of non stationarity:

Deterministic Trend :

- refers to a long-term upward or downward movement with predictable changes
- violates the “constant mean” requirement for stationarity

Observing the chart plotted earlier:

- The mean of the two periods separated the dotted blue line are clearly different

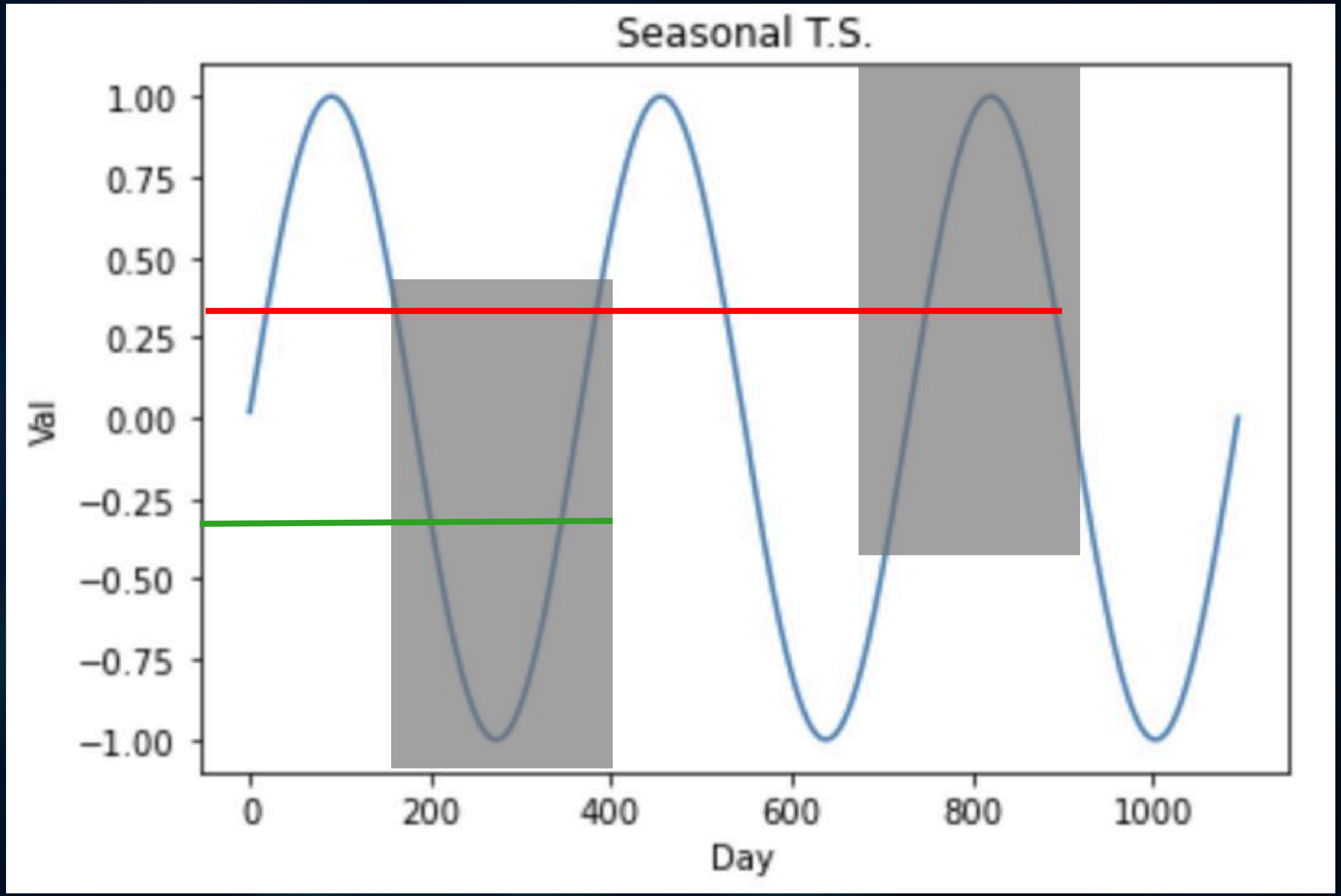


Types of non stationarity:

Presence of seasonality:

Observing this sample chart:

- Peaks and troughs at constant intervals
- Similar to sine / cosine curves

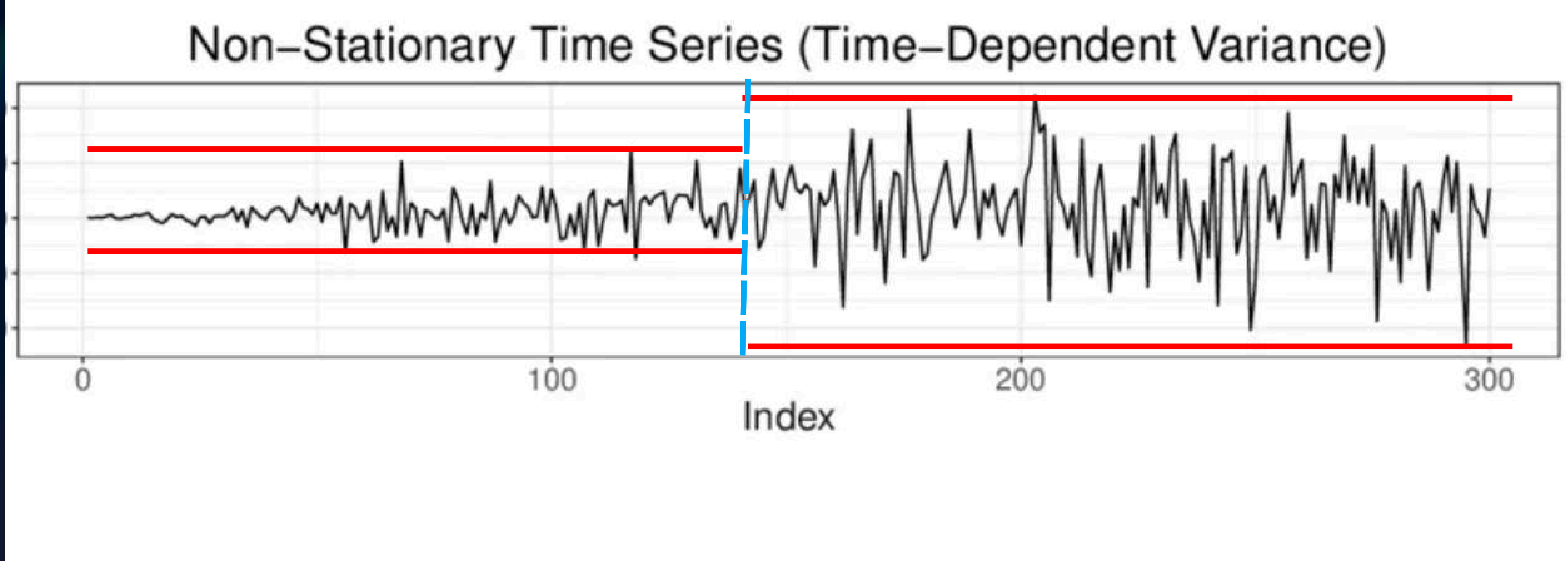


Types of non stationarity:

Non constant variance:

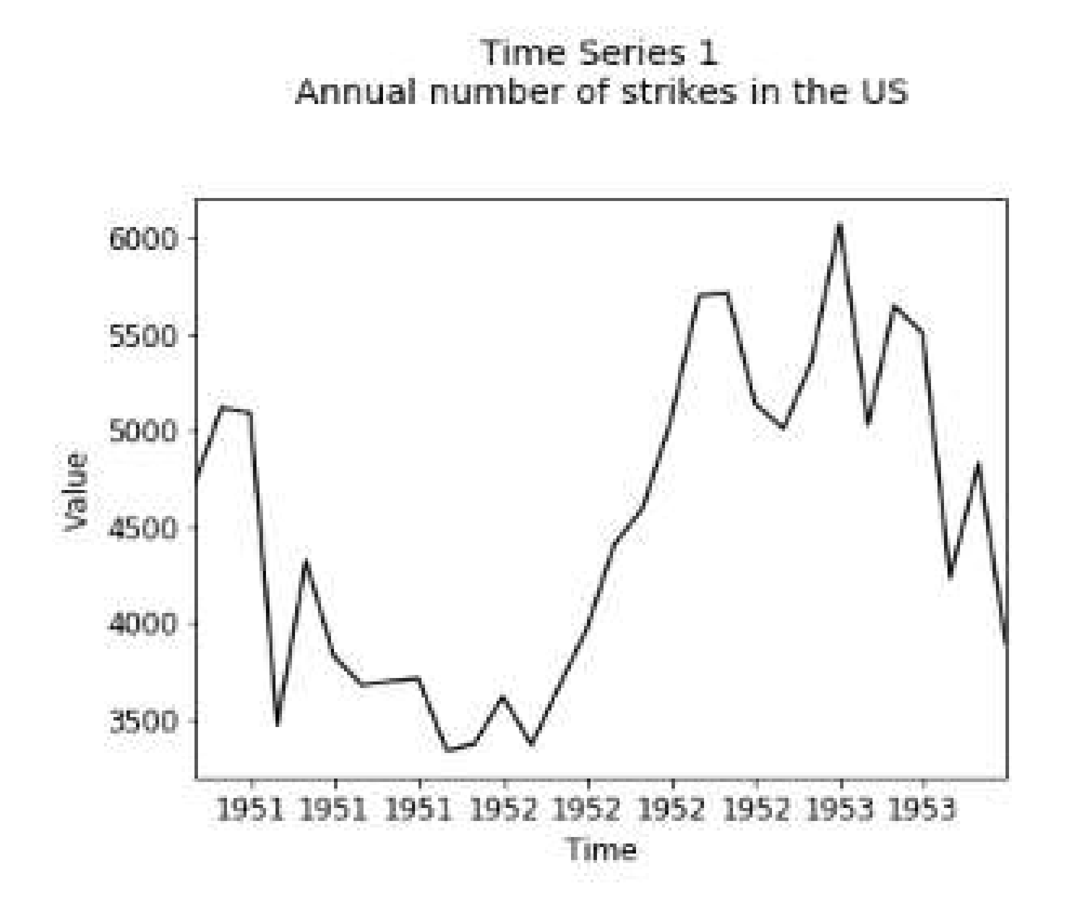
Observing the sample chart:

- The range of values in the first half of the time series is narrower than the second half
- Variance likely increasing (non-constant)



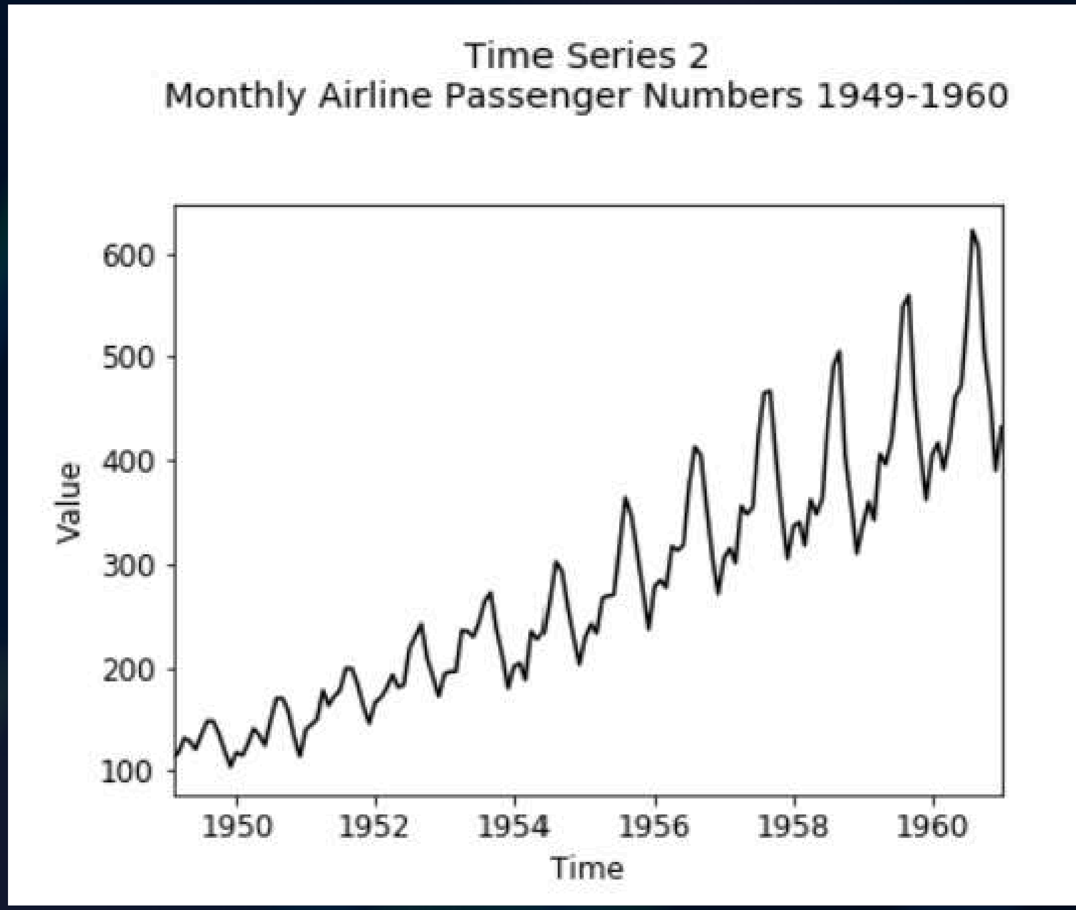
Types of non stationarity:

Test yourself:



Not Stationary!

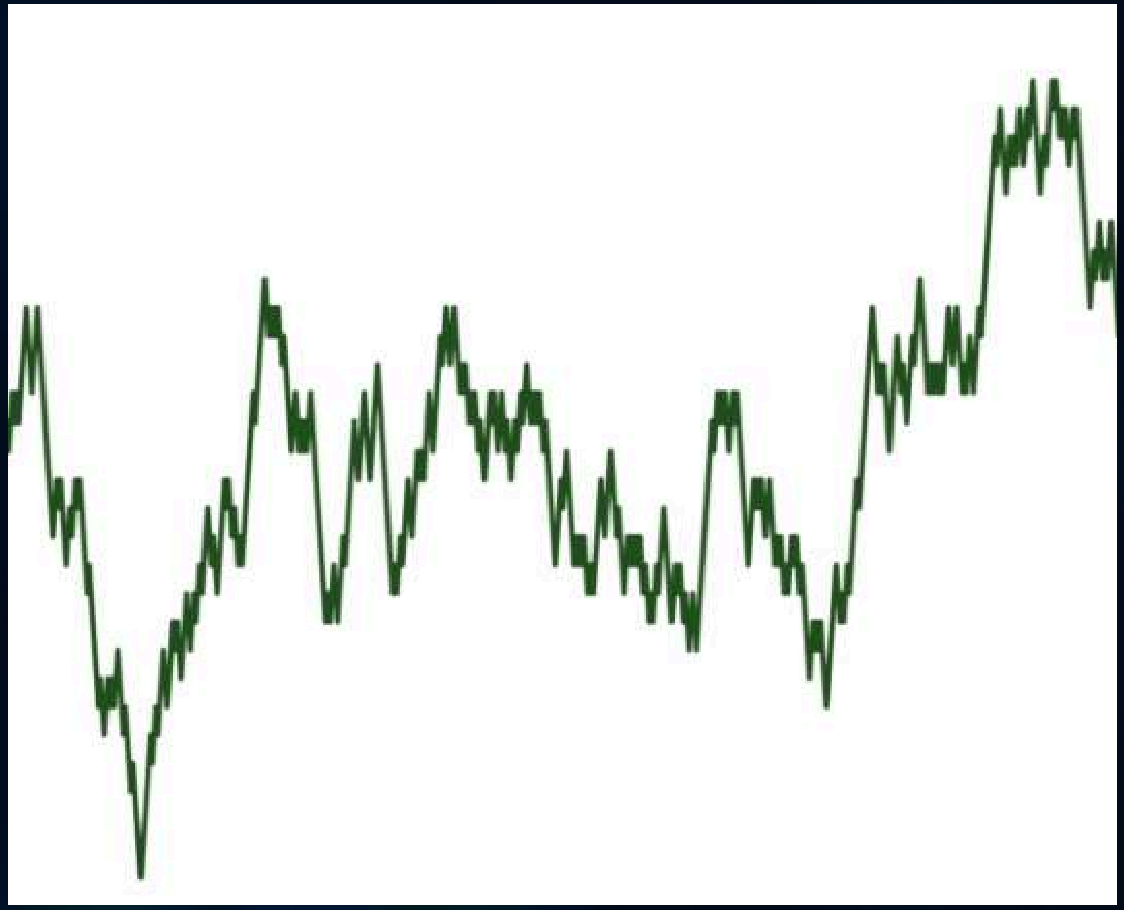
- Changing trend and level



Not Stationary!

- Upward trend (no constant mean)
- Presence of seasonality

What about a time series like this?



Types of non stationarity:

Stochastic Trend :

- trend created by accumulation of random shocks over every time period
- difficult to spot unlike deterministic trend
- Does not occur in every time series.
- If shocks do accumulate and results in a stochastic trend, time series is said to have a **unit root**

How accumulating shocks violate stationarity:

Shocks accumulating in the form

$$y_t = y_0 + \sum_{i=1}^{t} \varepsilon_i$$

$$\varepsilon_t \text{ is i.i.d. with } \mathbb{E}[\varepsilon_t] = 0, \quad \mathrm{Var}(\varepsilon_t) = \sigma^2$$

As a result

$$\mathrm{Var}(y_t) = t\sigma^2$$

Example: Accumulating shocks and GDP

- Technological innovations act as productivity shocks
- Each innovation builds on previous ones

Digitalisation raised productive capacity.

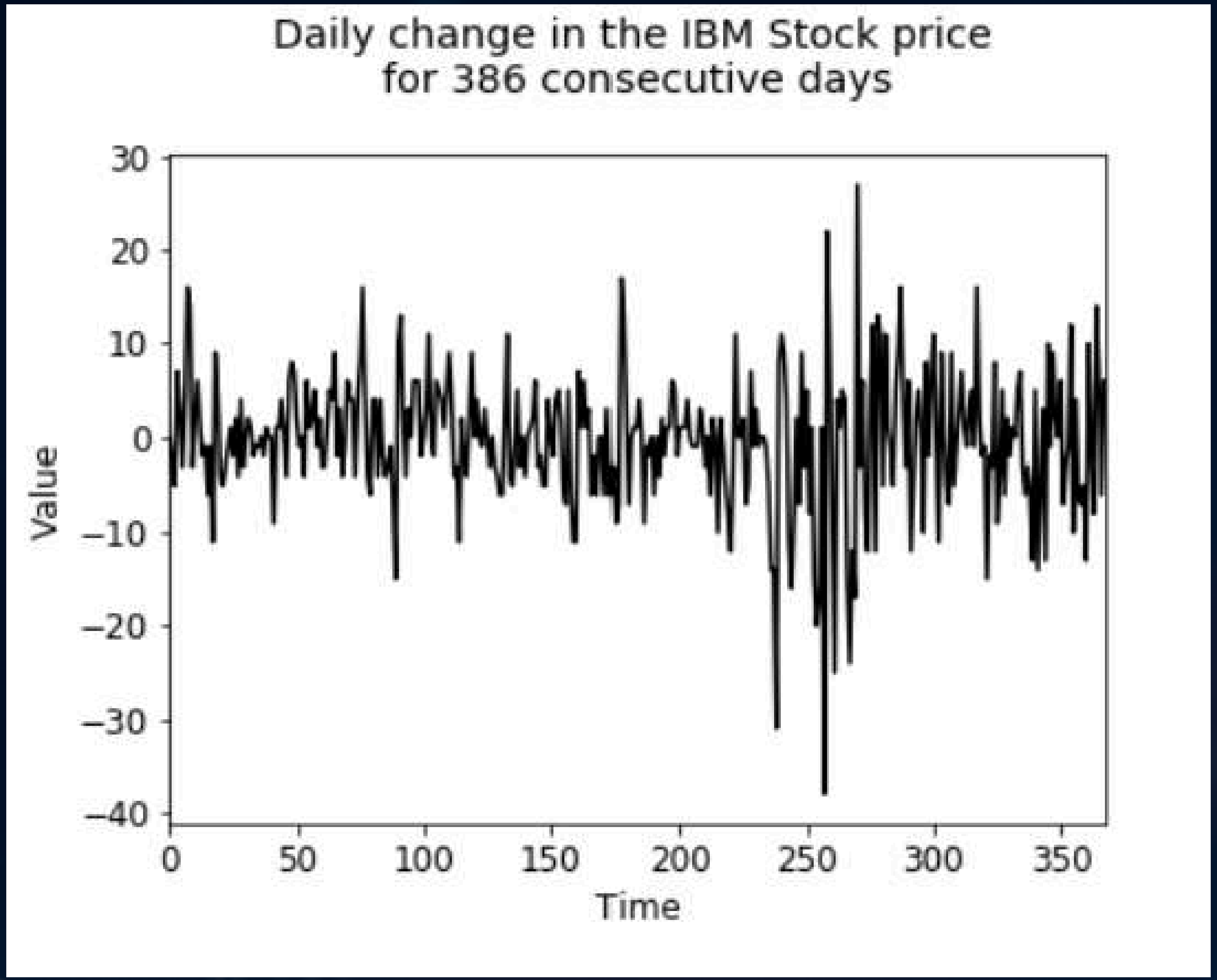
Productivity

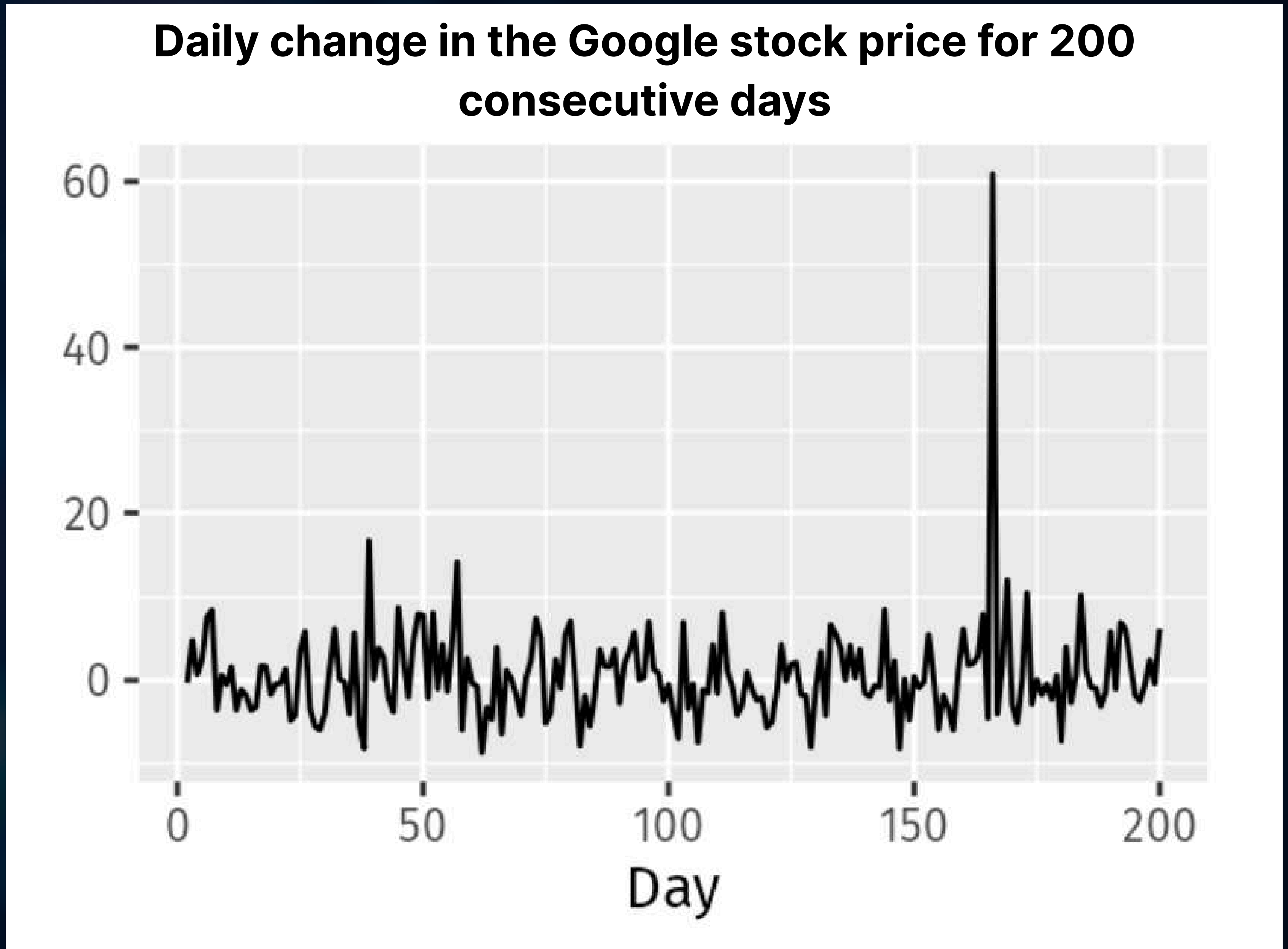
Artificial intelligence represents a further productivity shock on top of existing technology.

Productivity

Time

Stationary Time Series



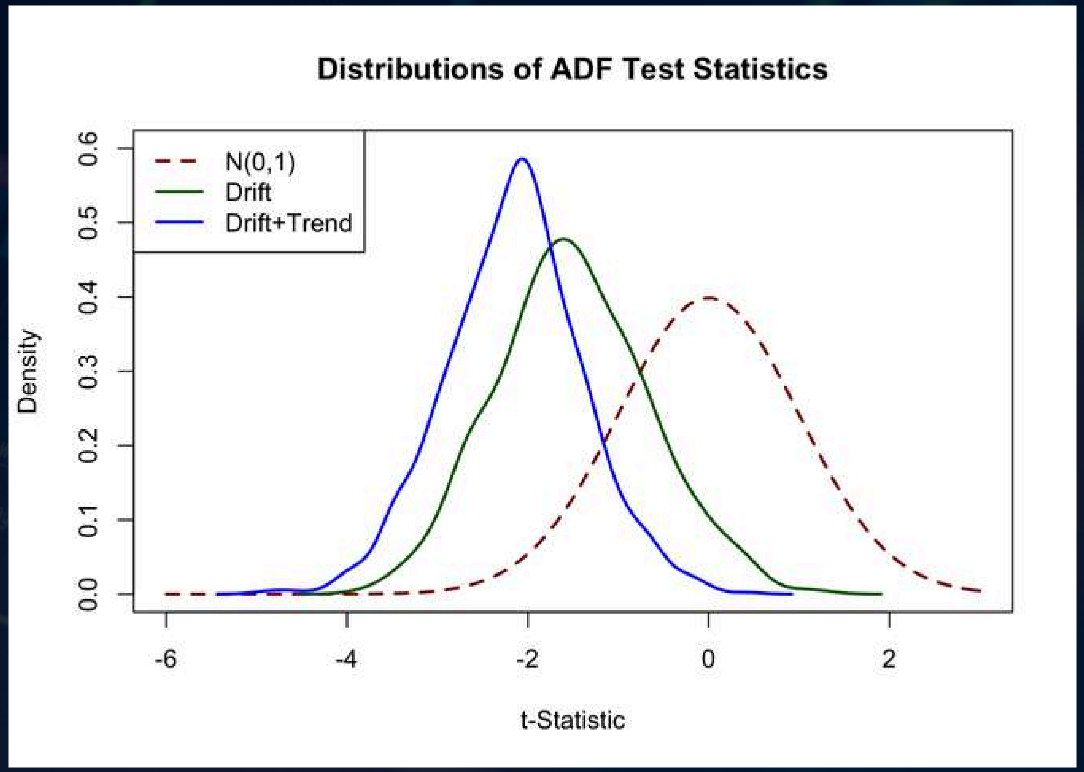


Testing for unit root

Augmented Dickey-Fuller Test :

What it is?

- It is a hypothesis test used to test for unit root
- Critical values are based on the Dickey-Fuller distribution



Dickey Fuller Distribution vs Normal Dist

How to test

- Null Hypothesis: Unit Root is present
- Alternative hypothesis: Unit root not present
- Compute test statistic
- Compare test statistic with critical values

Note: The null and alternative hypotheses are stated qualitatively to abstract from technical details.

Testing for unit root

Augmented Dickey-Fuller Test :

Function call:

```
adf_ur_dly_drift <- ur.df(dly_ts, type = "drift", lags = 4)
```

time series object

Choose appropriate argument:

- “none” if mean reverts to 0
- “drift” if mean reverts to non zero constant
- “trend” if clear trend

Values between 4-8 appropriate for quaterly data

Testing for unit root

Augmented Dickey-Fuller Test :

Results analysis

```
summary(adf_ur_dly_drift)
```

```
z.diff.lag3 -0.028486   0.058491  -0.487  0.62661
z.diff.lag4 -0.045822   0.057750  -0.793  0.42813
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 147.5 on 303 degrees of freedom
Multiple R-squared:  0.06825,   Adjusted R-squared:  0.0498
F-statistic: 3.699 on 6 and 303 DF,  p-value: 0.001455


Value of test-statistic is: -0.3334 19.4316 9.2484

Critical values for test statistics:
      1pct  5pct 10pct
tau3 -3.98 -3.42 -3.13
phi2  6.15  4.71  4.05
phi3  8.34  6.30  5.36
```

test statistic

Refer to 90% significance level

Critical values for comparison

Deriving Conclusions:

- Reject H_0 if **Test statistic < Critical value** at chosen significance level
- If H_0 rejected, time series is stationary
- Else, time series is not stationary

Making data stationary:

Log transformation

As levels of a series become larger, similar percentage changes result in exponential changes in absolute values

- 10% increase in $100 is $10
- 10% increase in $1,000,000 is $100,000

Log Transformation removes **scale based variance**

Transformed series: $x_t = \log(y_t)$

Differencing

Suppose our time series is

$$y_t = y_{t-1} + \varepsilon_t$$

ε_t is i.i.d. with $\mathbb{E}[\varepsilon_t] = 0, \quad \mathrm{Var}(\varepsilon_t) = \sigma^2$

This is not stationary!
Recall discussion of stochastic trend...

$$y_t = y_0 + \sum_{i=1}^{t} \varepsilon_i$$

which is equivalent to

$$y_t = y_{t-1} + \varepsilon_t$$

Taking difference: $\Delta y_t = y_t - y_{t-1}$

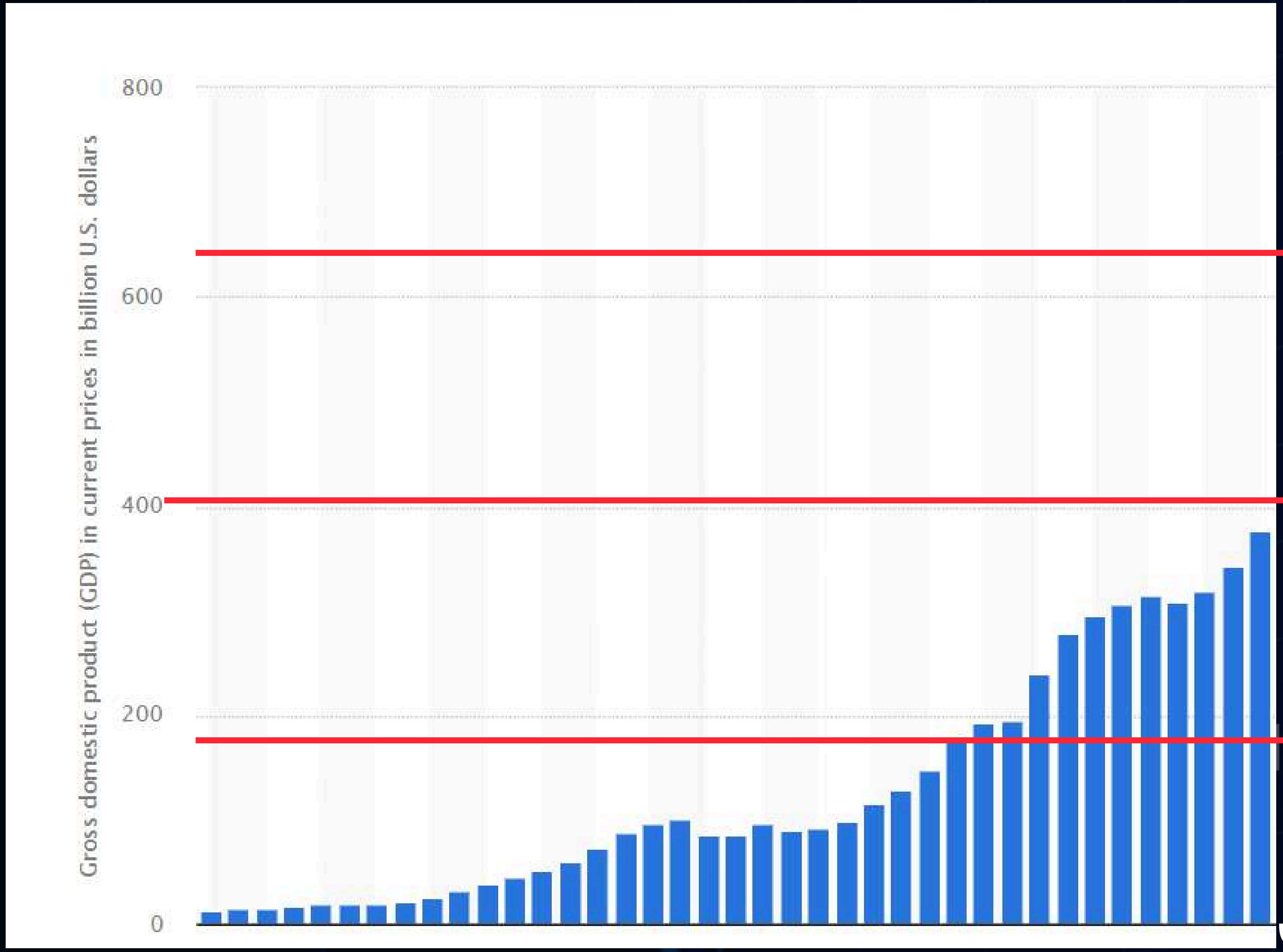
Transformed series: $\Delta y_t = (y_{t-1} + \varepsilon_t) - y_{t-1} = \varepsilon_t$

Differencing removes **stochastic trend**

For GDPC1, we will apply **log differencing**
(log transformation followed by differencing)

AutoCorrelation Function (ACF) & Partial AutoCorrelation Function (PACF)

Intuition Behind ACF



A

or

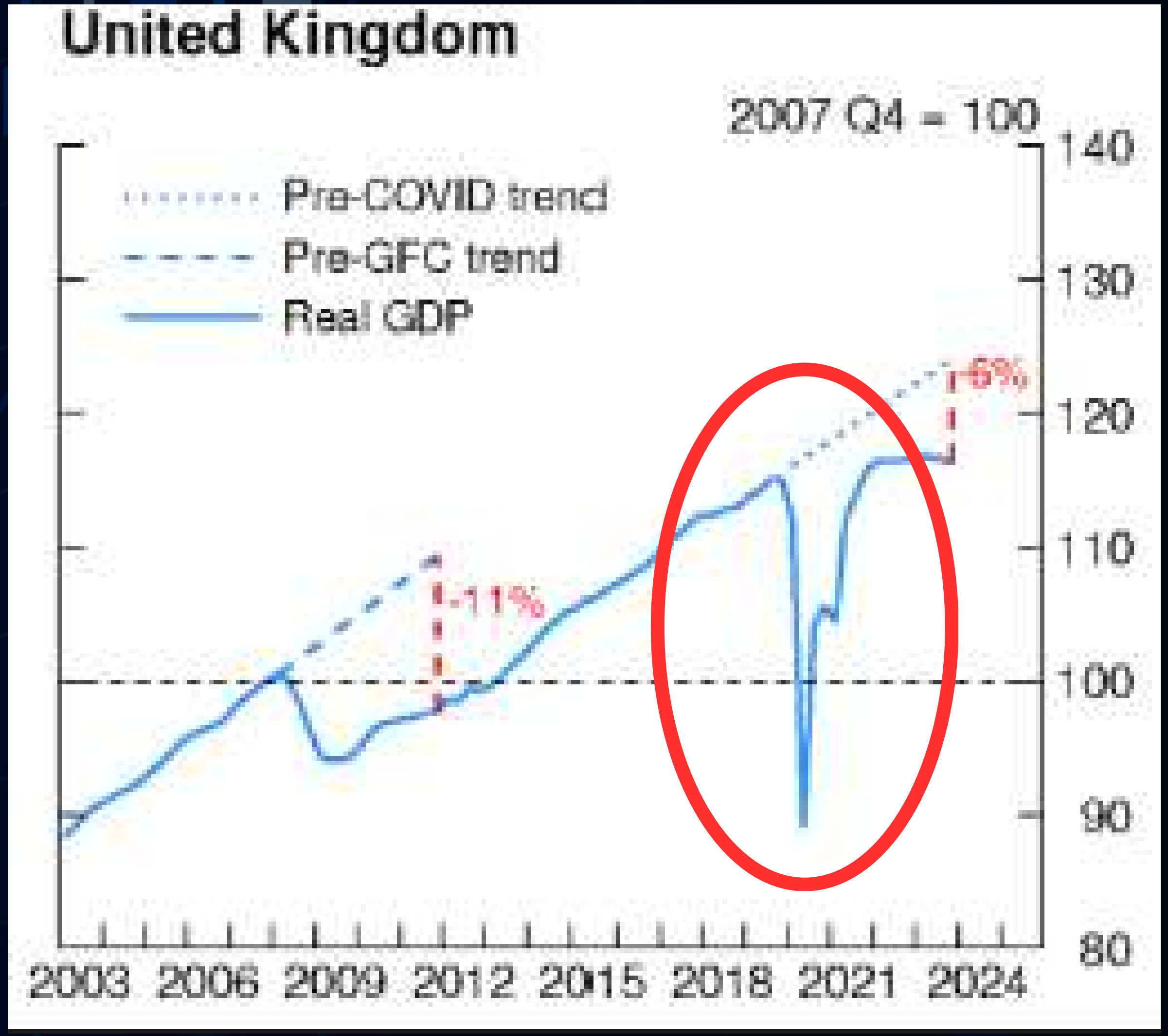
B

or

C

next period?

Black Swan Events
UK GDP level (drop caused by covid



Intuition Behind ACF

1. Past values can act as predictors for current values
2. This is due to presence of relationships between S_t and S_{t-n}

How relationships occur between time periods.
Example: Hotel Prices in Singapore every 6 months

S_{t-2} S_{t-1} S_t

Legend:

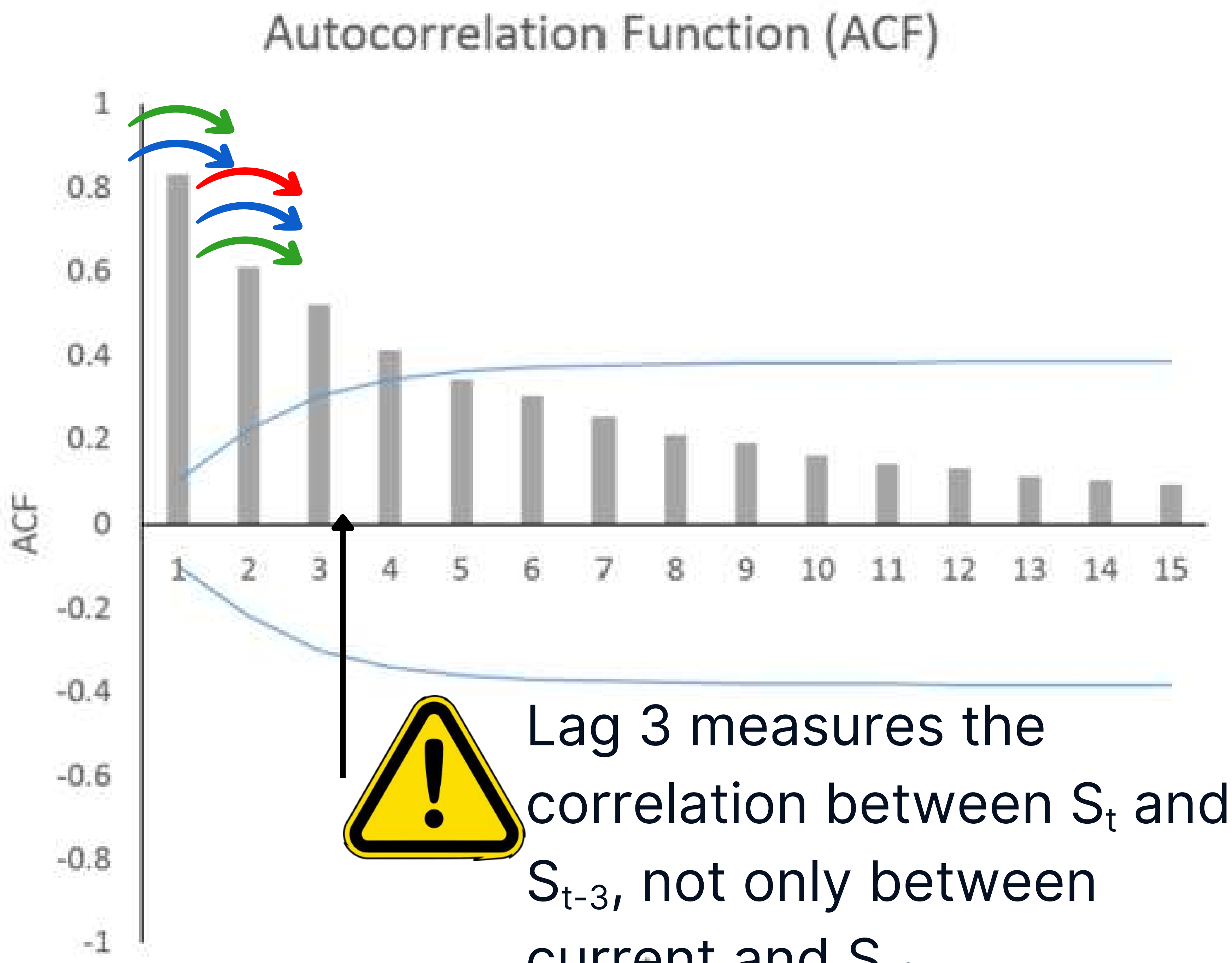
Price premium caused by F1 race

Resumption of travel post covid

AutoCorrelation (ACF)

Definition: Correlation between a time series and a lagged version of itself

- The correlation quantifies the strength and type of relationships between S_t and S_{t-n}
- Absolute value of the correlation can indicate stronger relationships
- Sign of correlation indicate type of relationship



Lag 3 measures the correlation between S_t and S_{t-3}, not only between current and S_{-3}

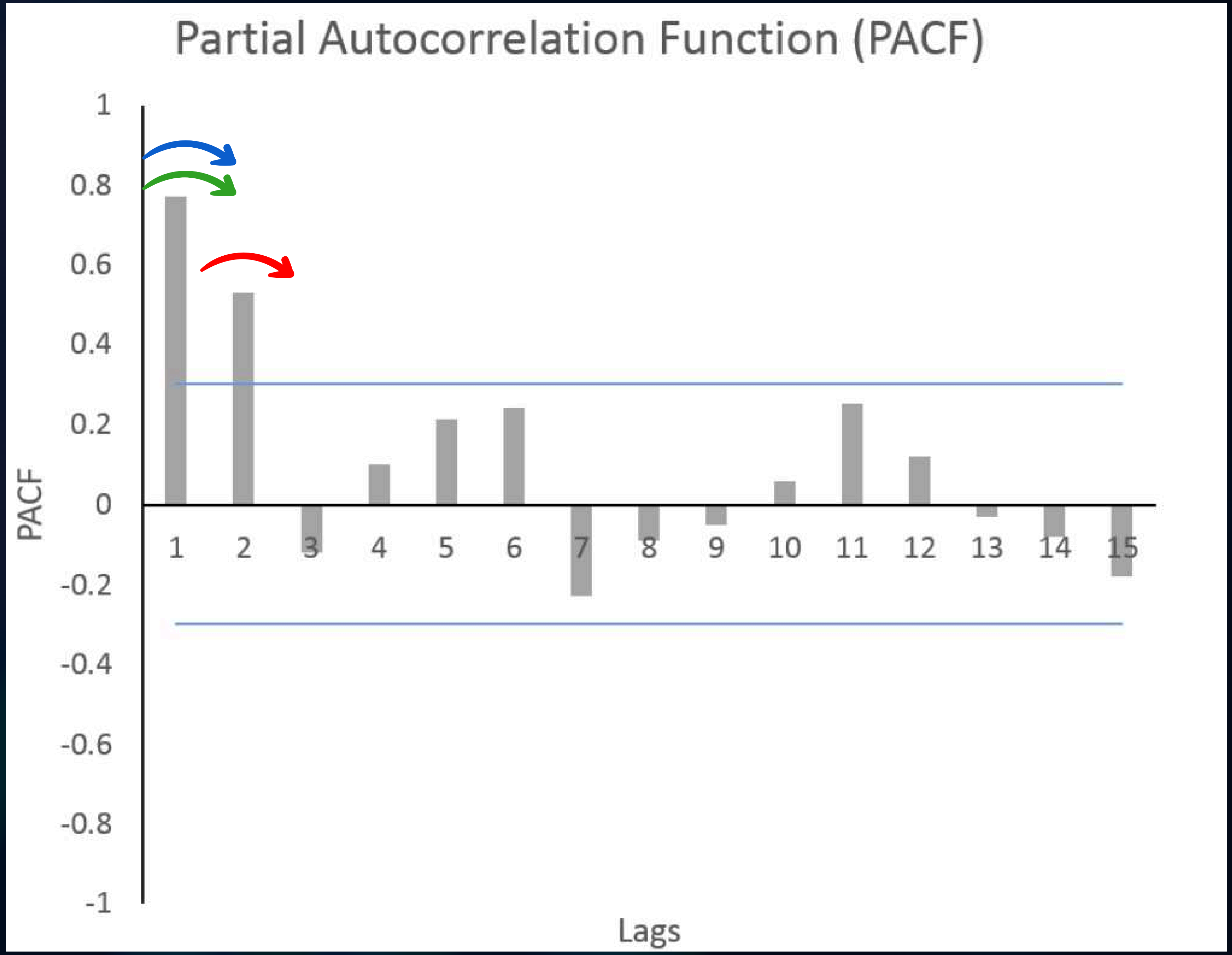
Partial AutoCorrelation (PACF)

Definition:

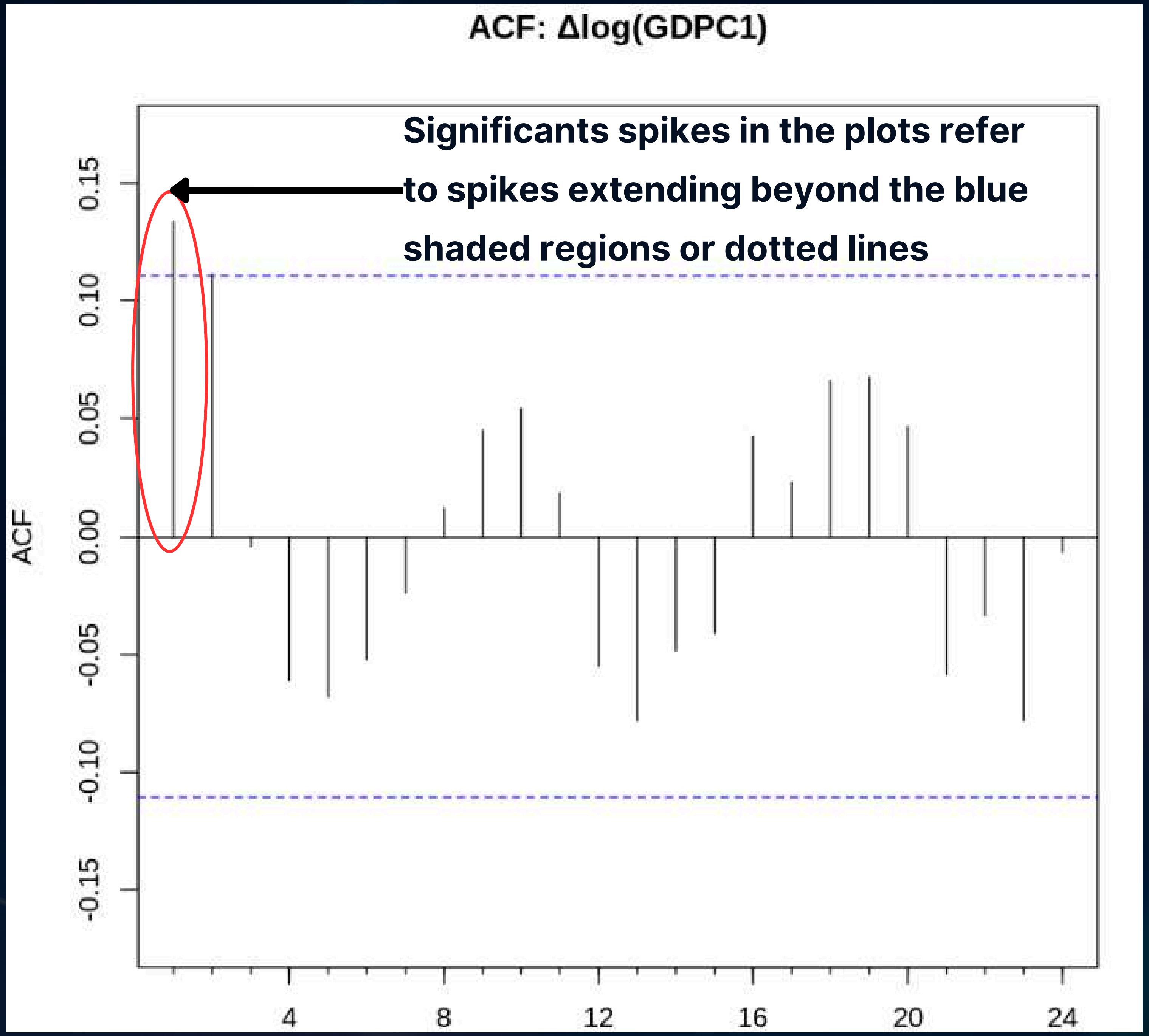
Correlation between observations at two time points, accounting for the values of the observations at all shorter lags.

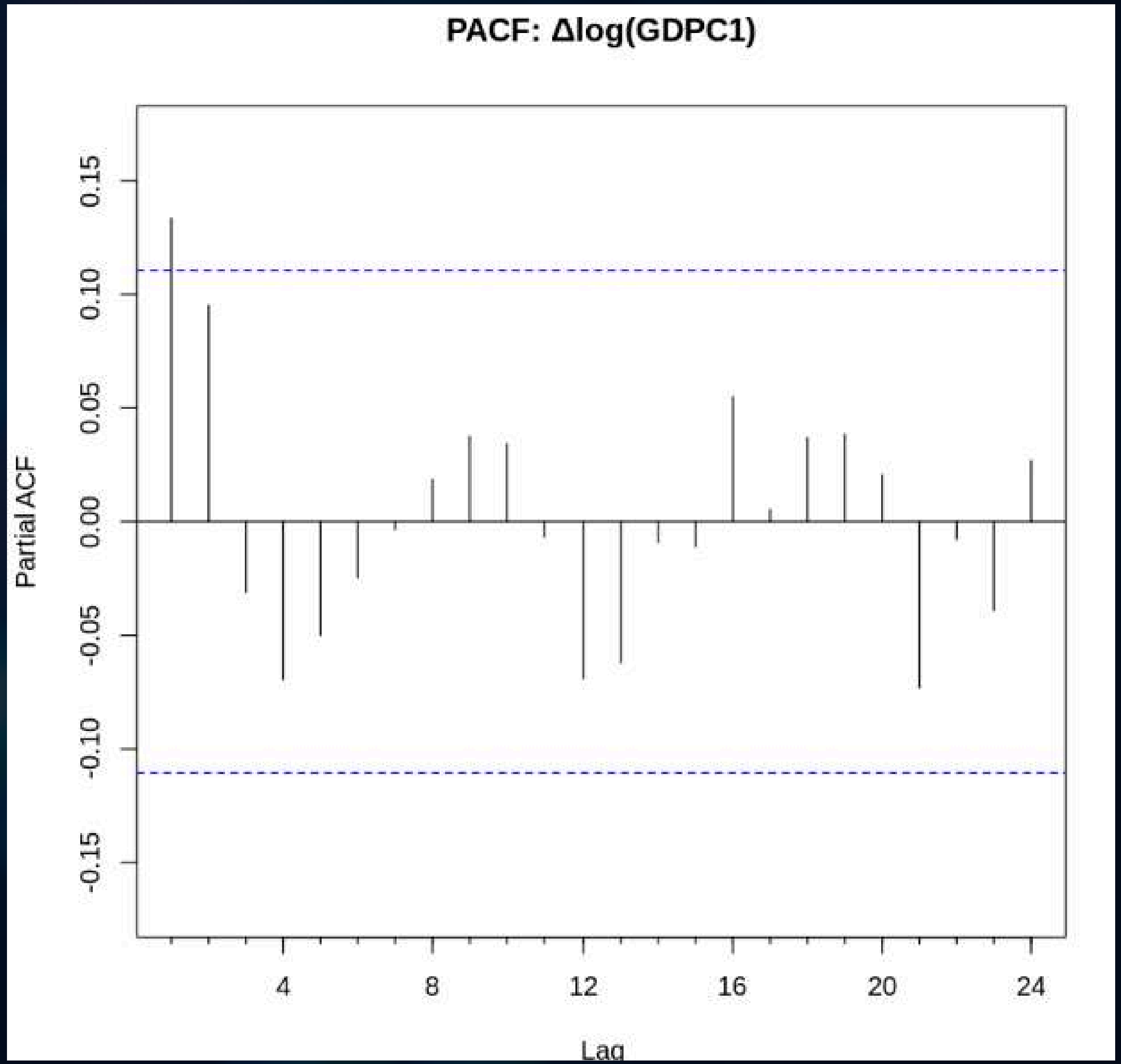
Shows direct relationship between observations at different lags

Rationale for PACF:
What if we want to build a linear model like this?
Predicted value = x_1 (Value at lag 1) + x_2 (Value at lag 2)

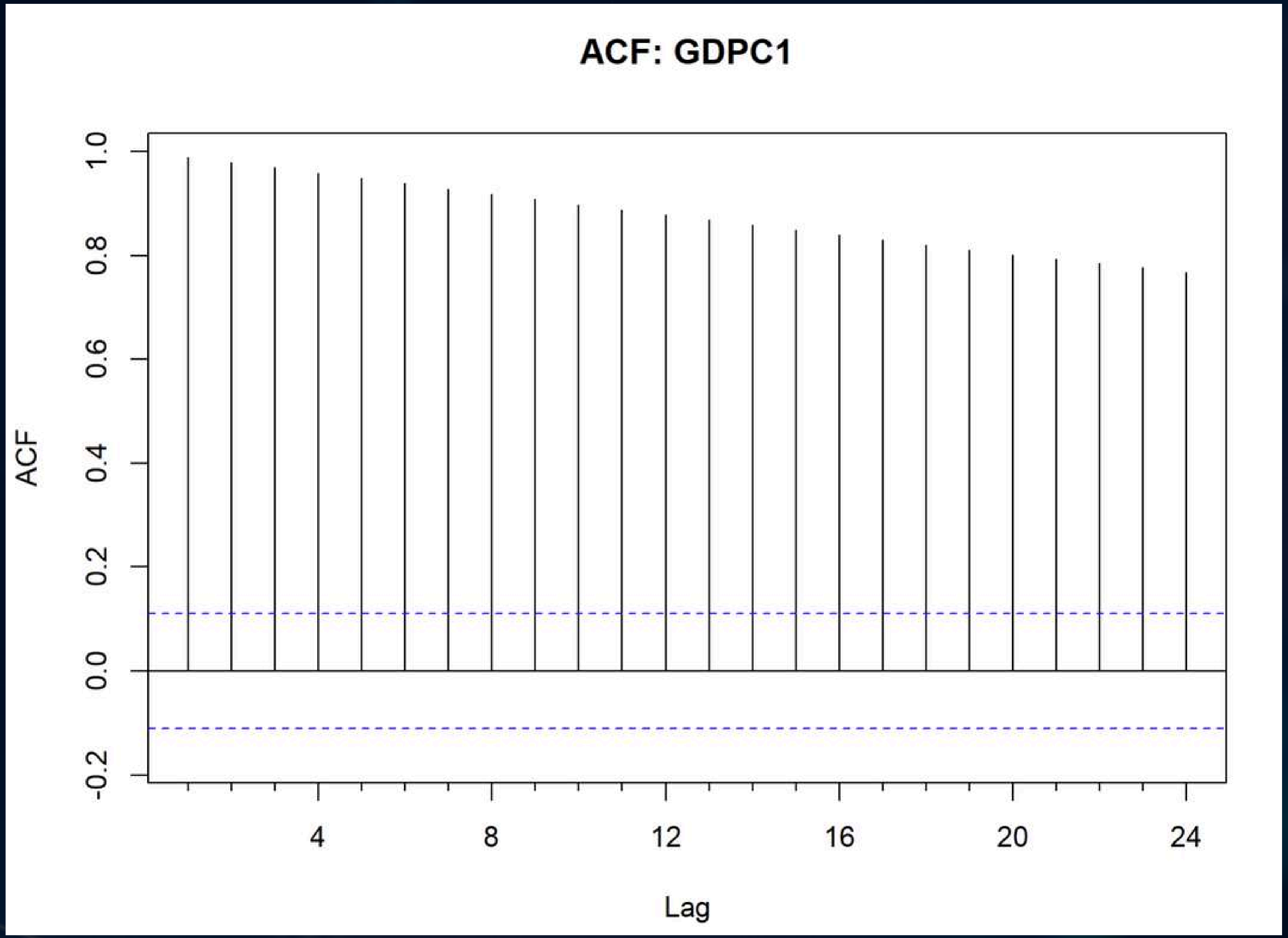


ACF & PACF plots





ACF & PACF plots



ACF & PACF assumes stationarity of time series

ACF plotted on non-stationary time series typically exhibits significant spikes over many lags

Hanover
and Tyke

Forecasting Techniques

3 Approaches to Time Series Forecasting

Forecasting models differ in **how** they use the past to answer the same question: How should past observations influence our prediction of the future?

1

SIMPLE MOVING-AVERAGE

Useful for identifying trend & trend cycles

- Natural baseline forecasting models

2

EXPONENTIAL SMOOTHING MODELS

Weighted averages that adapt to change

- Puts more weight on recent data, thus adjusting faster when the series changes
- Can be extended to handle trends & seasonality

3

ARIMA MODELS

Forecasts future values using patterns in past values

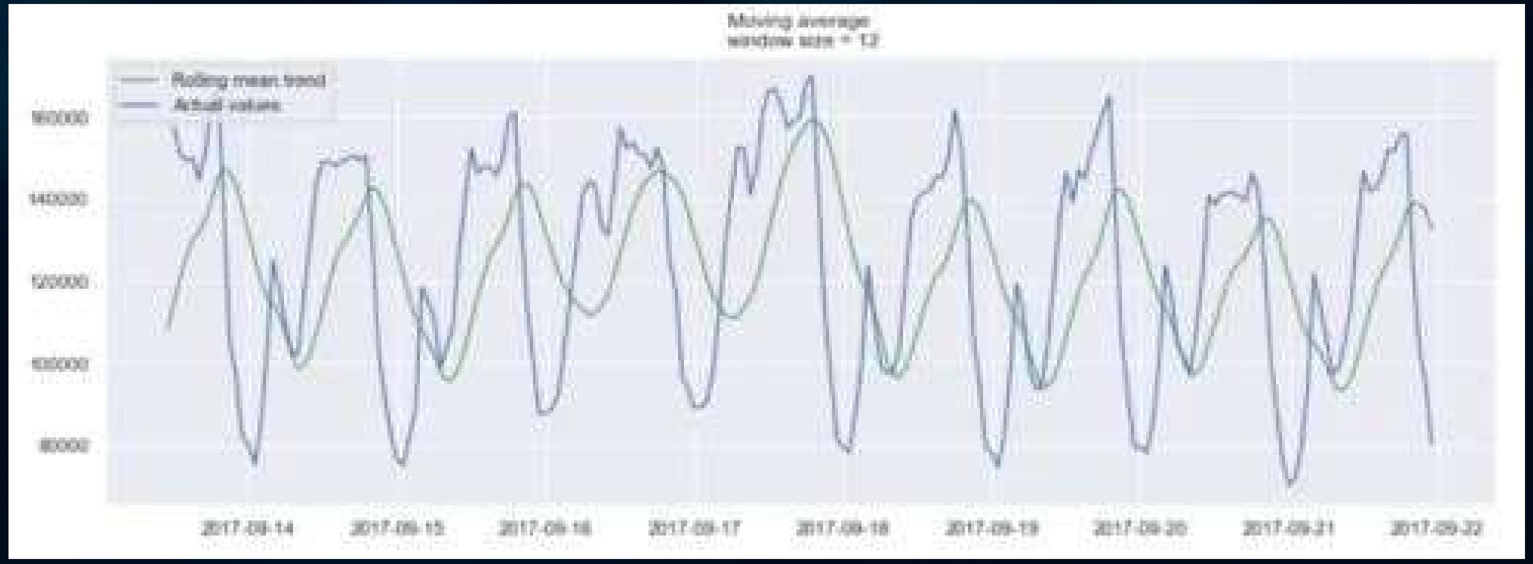
- Learn how the series moves from one period to the next
- Often used after removing trends and seasonality

Simple Moving Average

The moving average model states that the next observation is the mean of all past observations.

- **The ‘moving’ part refers to how the window shifts forward as time passes**

$$\hat{Y}_t = \frac{1}{k}\sum_{i=0}^{k-1} Y_{t-i}$$



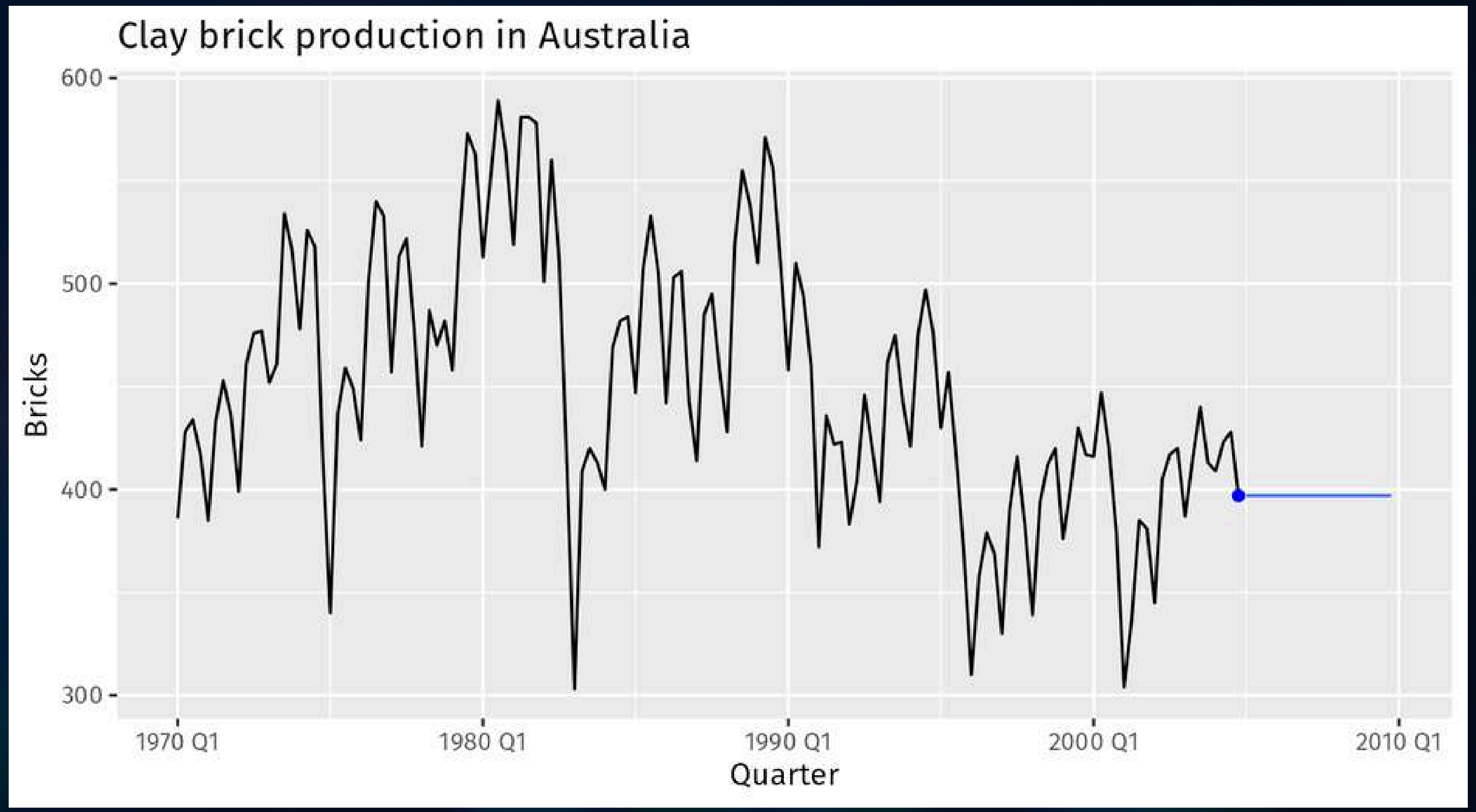
- Forecast = average of the last k observations (window)
  - Larger k → smoother, slower to react
  - Smaller k → more responsive, noisier

Simple Moving Average – Naïve Forecast

The Naïve Forecast assumes that the best guess for the future is simply most recent observation.

- **Equivalent to a moving average with window size k = 1**
- **Serves as the standard benchmark in forecasting**

$$\hat{Y}_{t+h} = Y_t$$



If a more complex model cannot outperform the naive forecast, it is not adding meanngful predictive value!

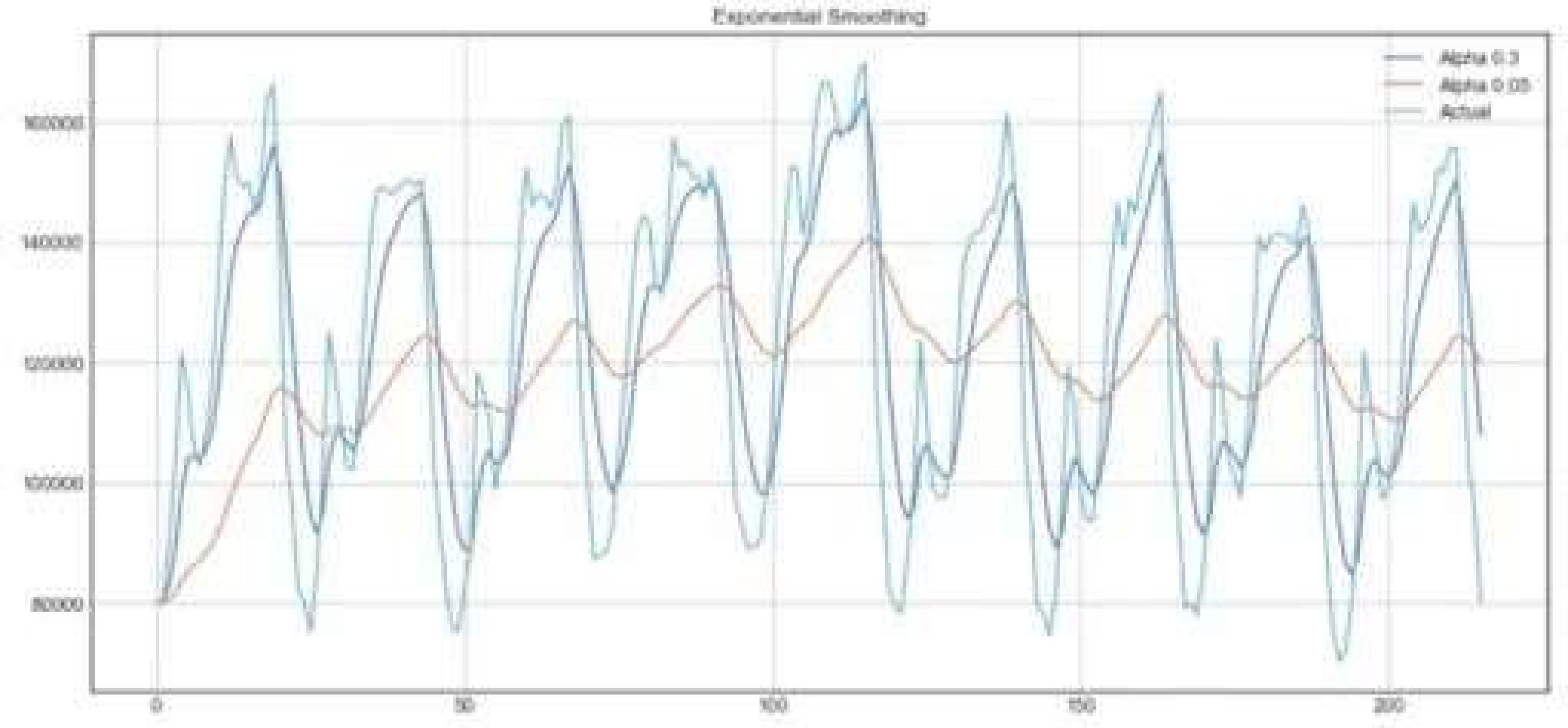
Exponential Smoothing Models

Exponential Smoothing uses similar logic to moving average, but this time, a different decreasing weight is assigned to each observation.

- **Less importance is given to observations as we move further from the present.**

$$\hat{Y}_t = \alpha Y_{t-1} + (1-\alpha)\hat{Y}_{t-1}, \quad 0 < \alpha < 1$$

$$\hat{Y}_t = \alpha Y_{t-1} + \alpha(1-\alpha)Y_{t-2} + \alpha(1-\alpha)^2 Y_{t-3} + \cdots$$

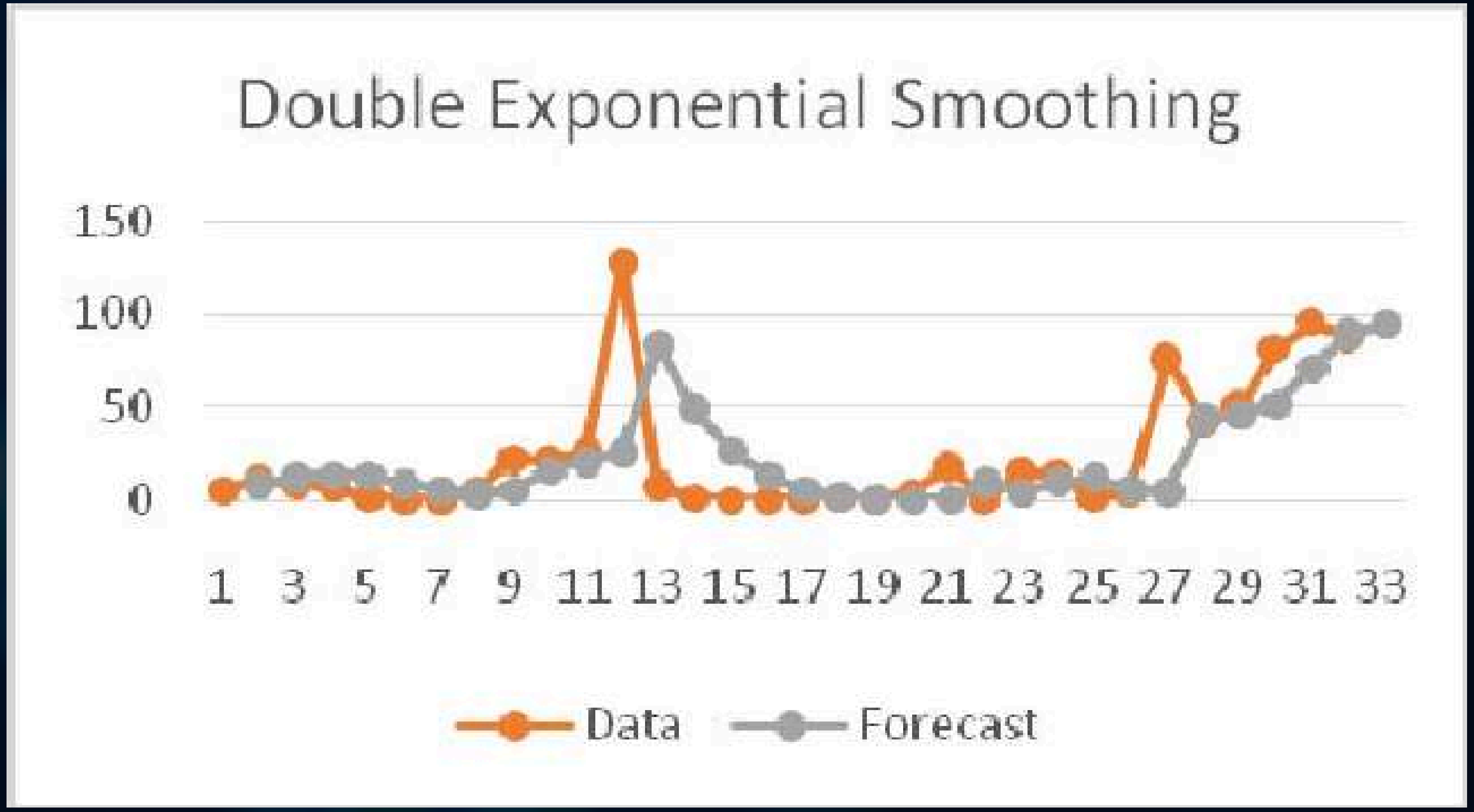


- Today's forecast = average of yesterday's actual value & yesterday's forecast
  - α controls the extent to which recent values are emphasised as it compounds over time
    - α = 1 → reduces to naive forecast
  - Large → reacts quickly
  - Smaller → smoother, slower to react

Extensions of Exponential Smoothing

1. **Double Exponential Smoothing is used when there is a trend in the time series.**
    - **Extends single exponential smoothing by tracking:**
        - **Level → current value**
        - **Trend → direction & speed of change**

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



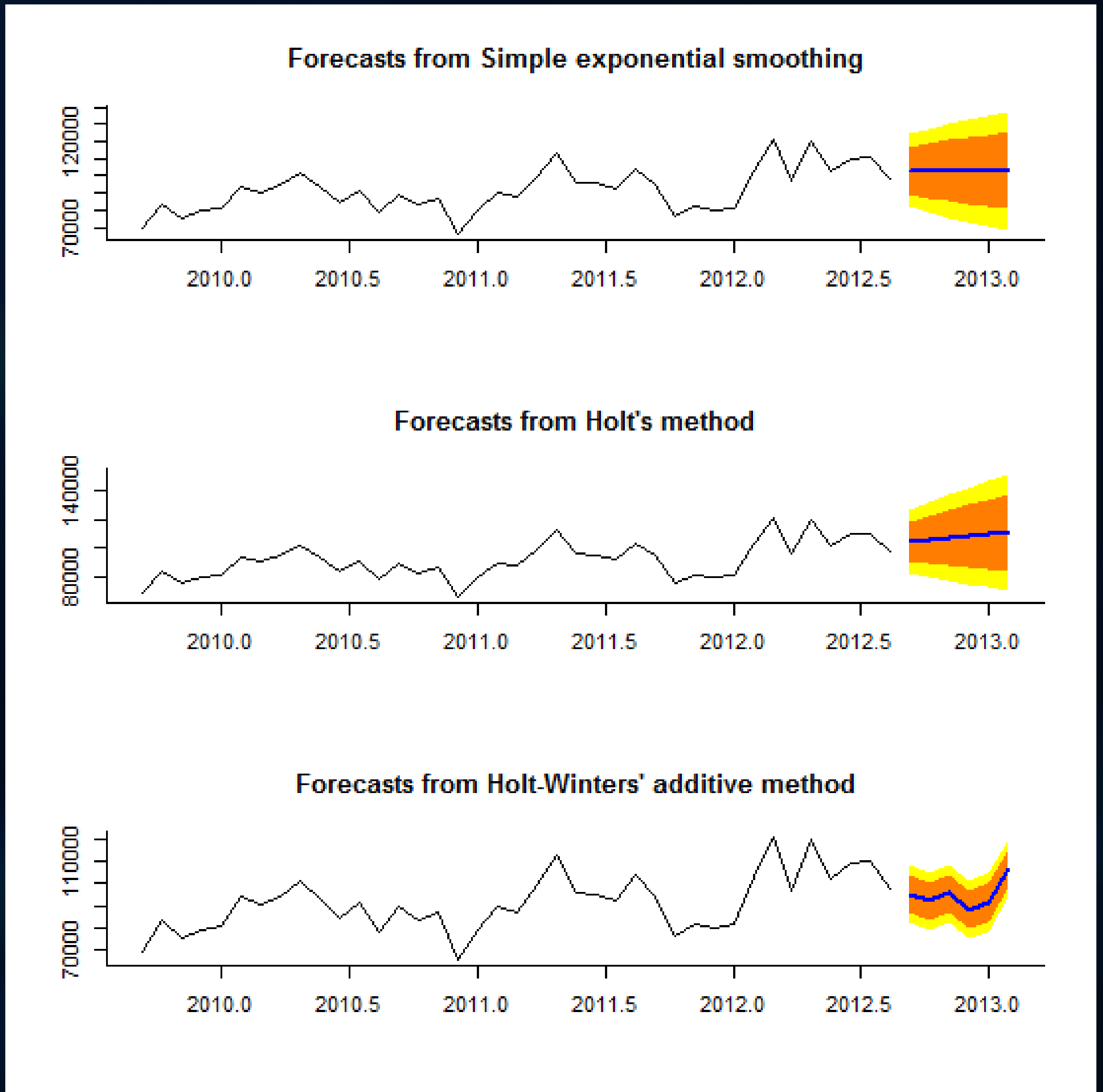
- Level equation updates our estimate of where the series currently is
    - α → controls how quickly the level adapts
- Trend equation updates how fast the series is increasing or decreasing
    - β → controls how quickly the trend adapts

Extensions of Exponential Smoothing

2. **Triple Exponential Smoothing is used when there is trend & seasonality in the time series.**
    - **Extends Double Exponential smoothing by adding a seasonal component.**

$$
\begin{aligned}
\text{Level: } \ell_t &= \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
\text{Trend: } b_t &= \beta(\ell_t - \ell_{t-1}) + (1-\beta)b_{t-1} \\
\text{Seasonal: } s_t &= \gamma(y_t - \ell_t) + (1-\gamma)s_{t-m}
\end{aligned}
$$

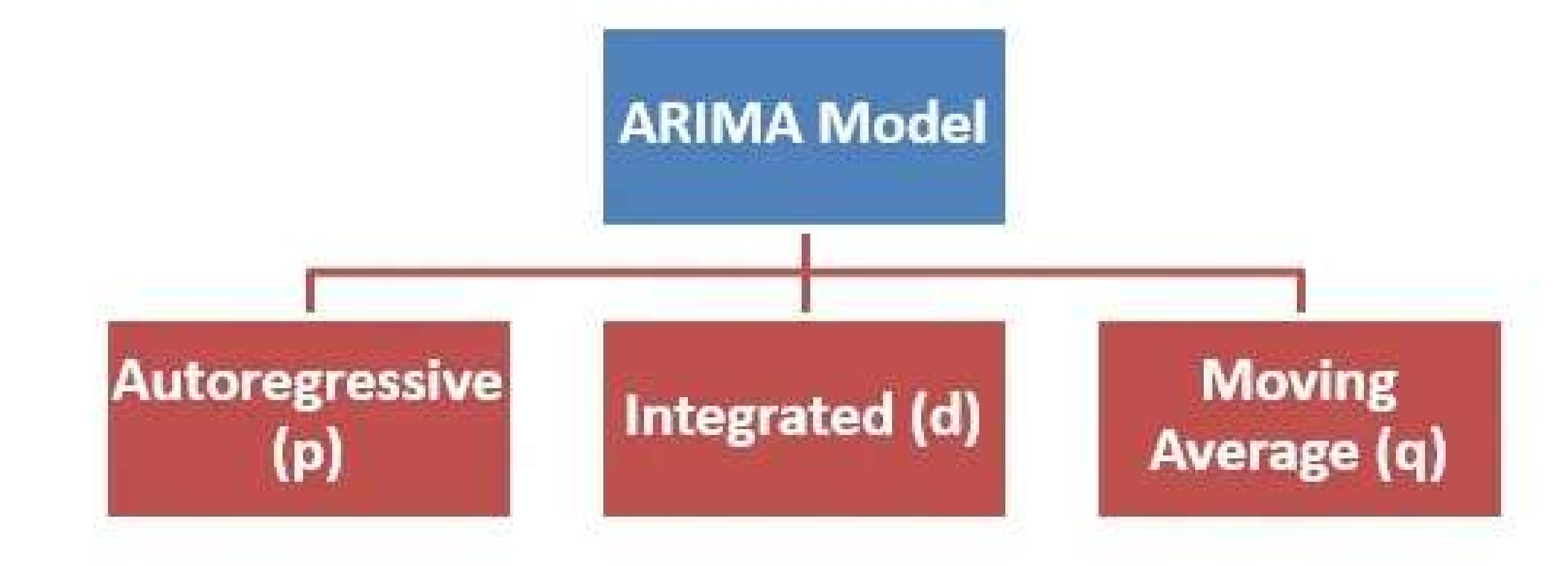
At each step, the model updates its estimate of the level, the trend, and the seasonal pattern, using a weighted average of new data and past estimates.



ARIMA Models

ARIMA (AutoRegressive Integrated Moving Average) uses patterns in past values to focus on how the series moves from one period to the next.

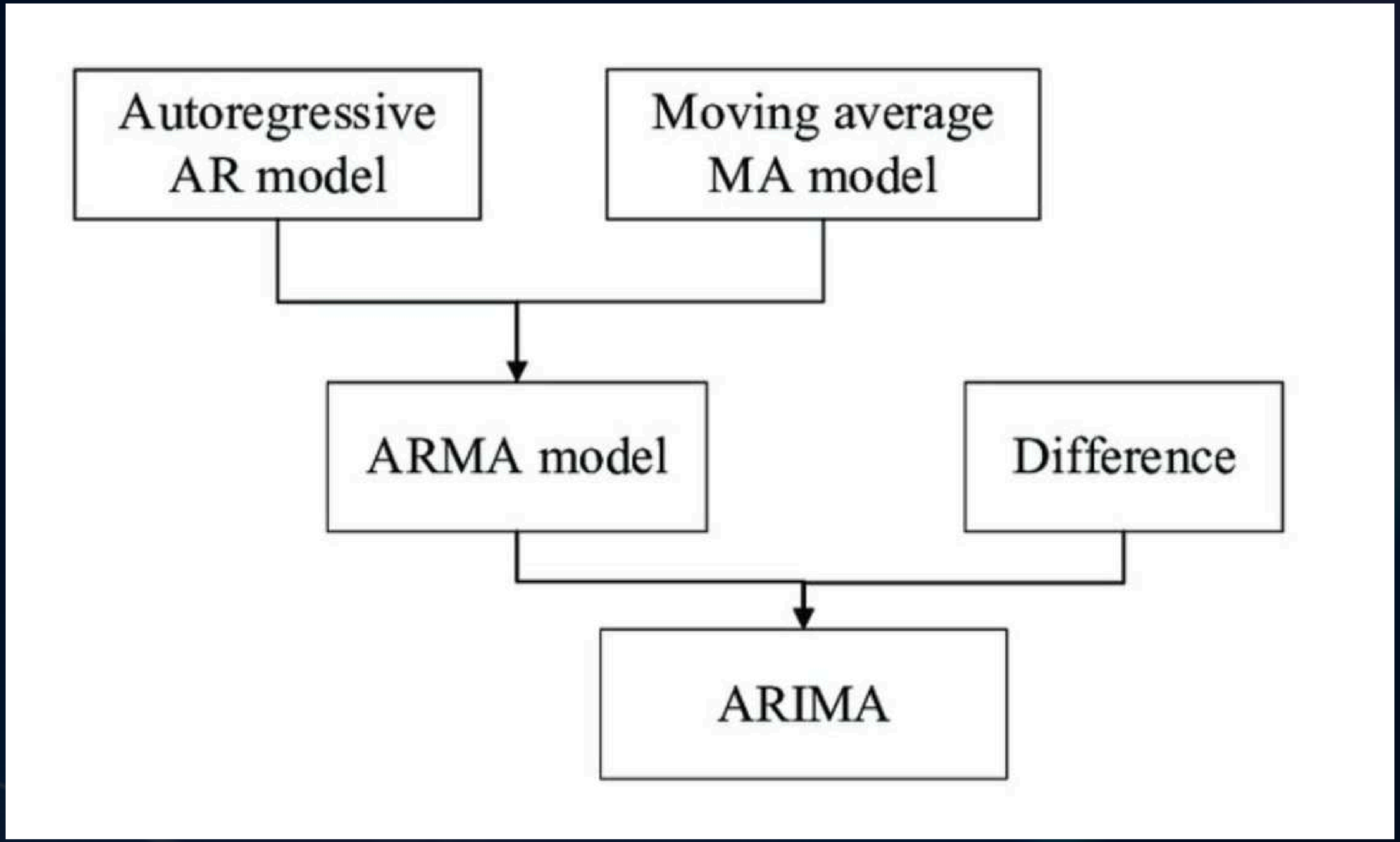
- **Works best after removing trend & seasonality → ARIMA models assume stationarity after differencing**
- **Key Idea: Models short-term dependence, not long-term structure**



ARIMA(p, d, q)

1. AR (AutoRegressive)
   - depends on past values
2. I (Integrated / Differencing)
   - differences the data to remove trend
3. MA (Moving Average)
   - depends on past forecast errors

Inside ARIMA: AR vs MA



1. AR (Autoregressive): dependence on past values
2. MA (Moving Average): dependence on past shocks (errors)
3. ARMA: AR + MA for stationary data
4. Differencing: removes trend / seasonality
5. ARIMA: ARMA applied after differencing

ARIMA Models – Choosing p, d, q

ARIMA equation

$$\underset{\text{Integrated}}{y_t^{(d)}} = c + \varepsilon_t + \underbrace{\phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p}}_{\text{Auto-Regressive}} + \underbrace{\theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}}_{\text{Moving Average}}$$

1. Choose **d** (differencing)
   - If the data has a trend / changing level, we difference until it behaves more consistently over time.
2. Choose **p** (autoregressive order)
   - The p term captures how many past values influence the current value.
   - We can use the **PACF plot** to see how many past lags have a direct effect.
3. Choose **q** (moving average order)
   - The q term captures how long the impact of shocks persists
   - We can use the **ACF plot** to see how long shocks remain correlated.

Building Models – Code Along

RECALL:

- We have a stationary series
- We have ACF / PACF diagnostics
- Now → Let's <u>fit models and forecast!</u>

What Diagnostics told us:

- Series transformed to be stationary: Δlog(GDP)
- PACF: significant spike at lag 1
- ACF: short-lived dependence

A low-order AR model is appropriate!

Building Models – Code Along

Using auto.arima() as Confirmation

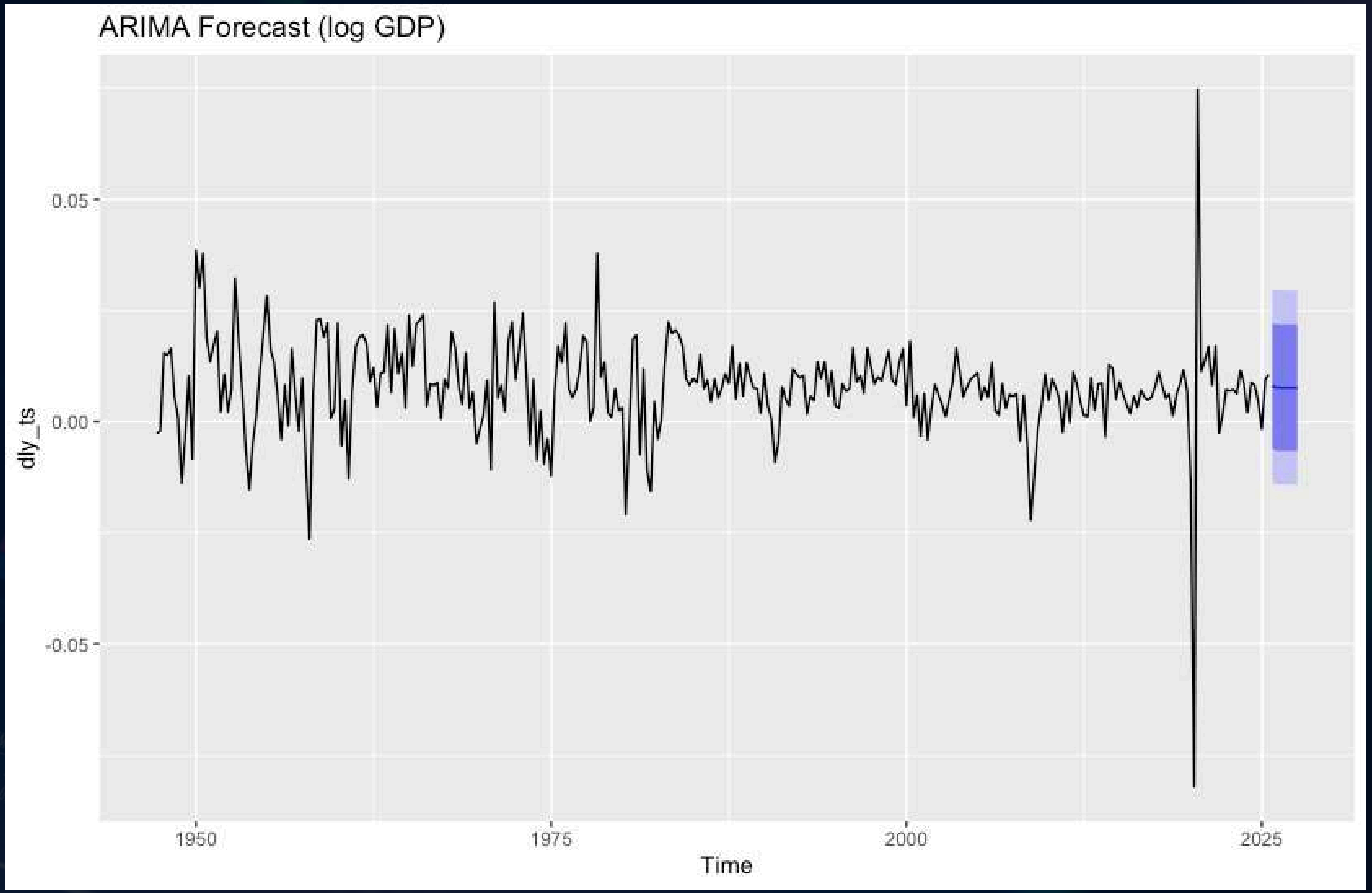
```r
# -------------------------------
# Confirmation step (software)
# -------------------------------
auto_model <- auto.arima(
  dly_ts,
  d = 0, #already differenced
  seasonal = FALSE
)
summary(auto_model)
```

```
Series: dly_ts
ARIMA(1,0,0) with non-zero mean
```

auto.arima() searches across ARIMA models, & chooses the model with lowest AICc (information criterion)

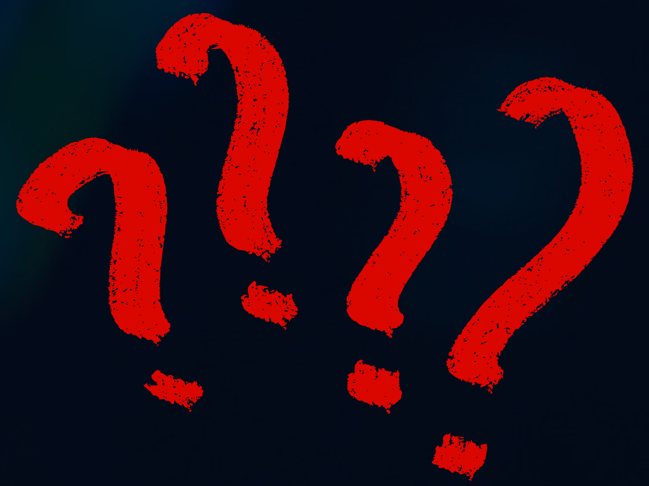
Forecasting with AR Model – Code Along

AR Model Forecast (GDP Growth)



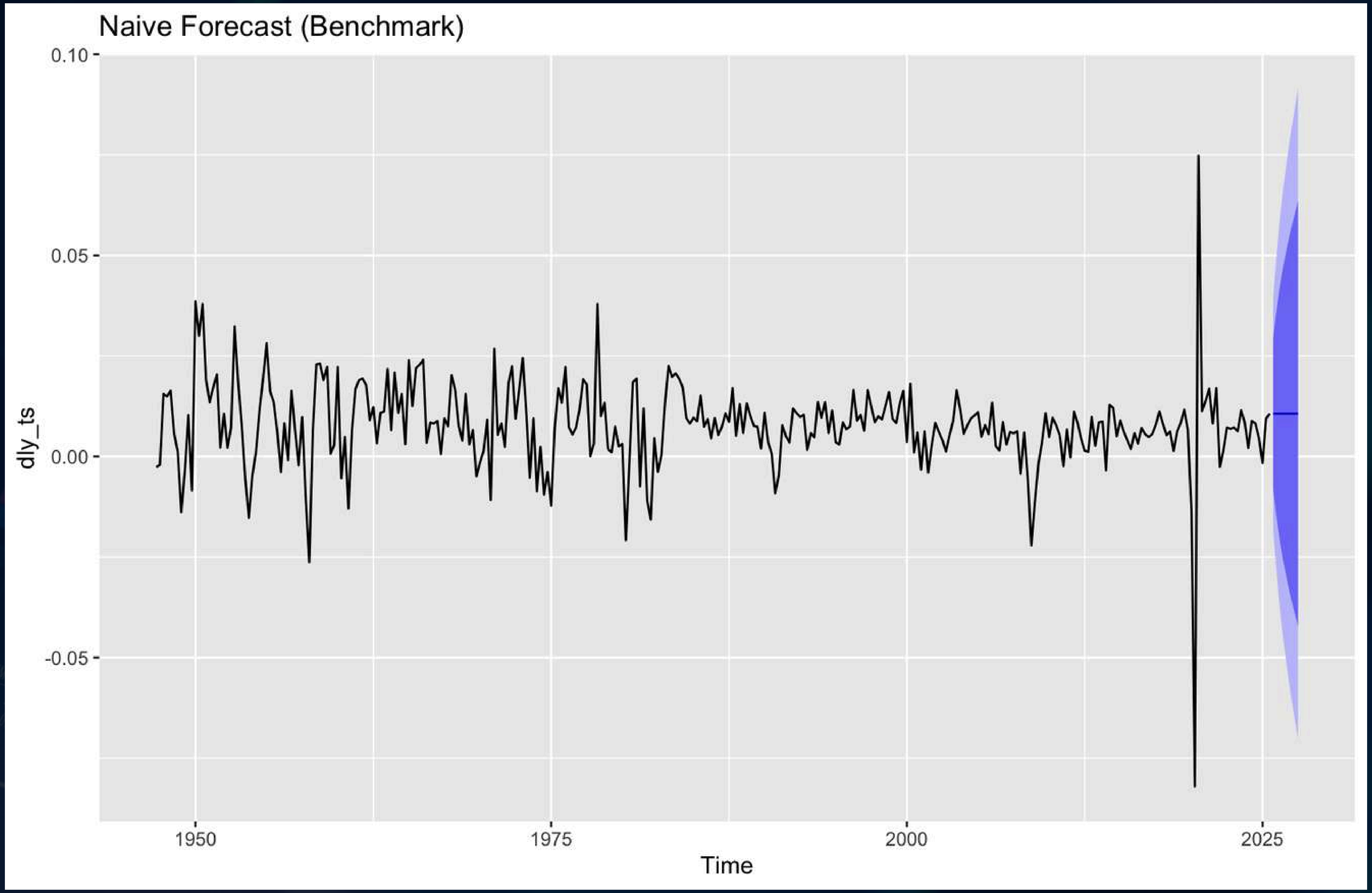
- uses recent growth to predict future growth, capturing dependence
- mean-reverting behaviour
- uncertainty increases with horizon

<u>SO, is this forecast any good?</u>



Benchmark Model – Code Along

Naive Forecast

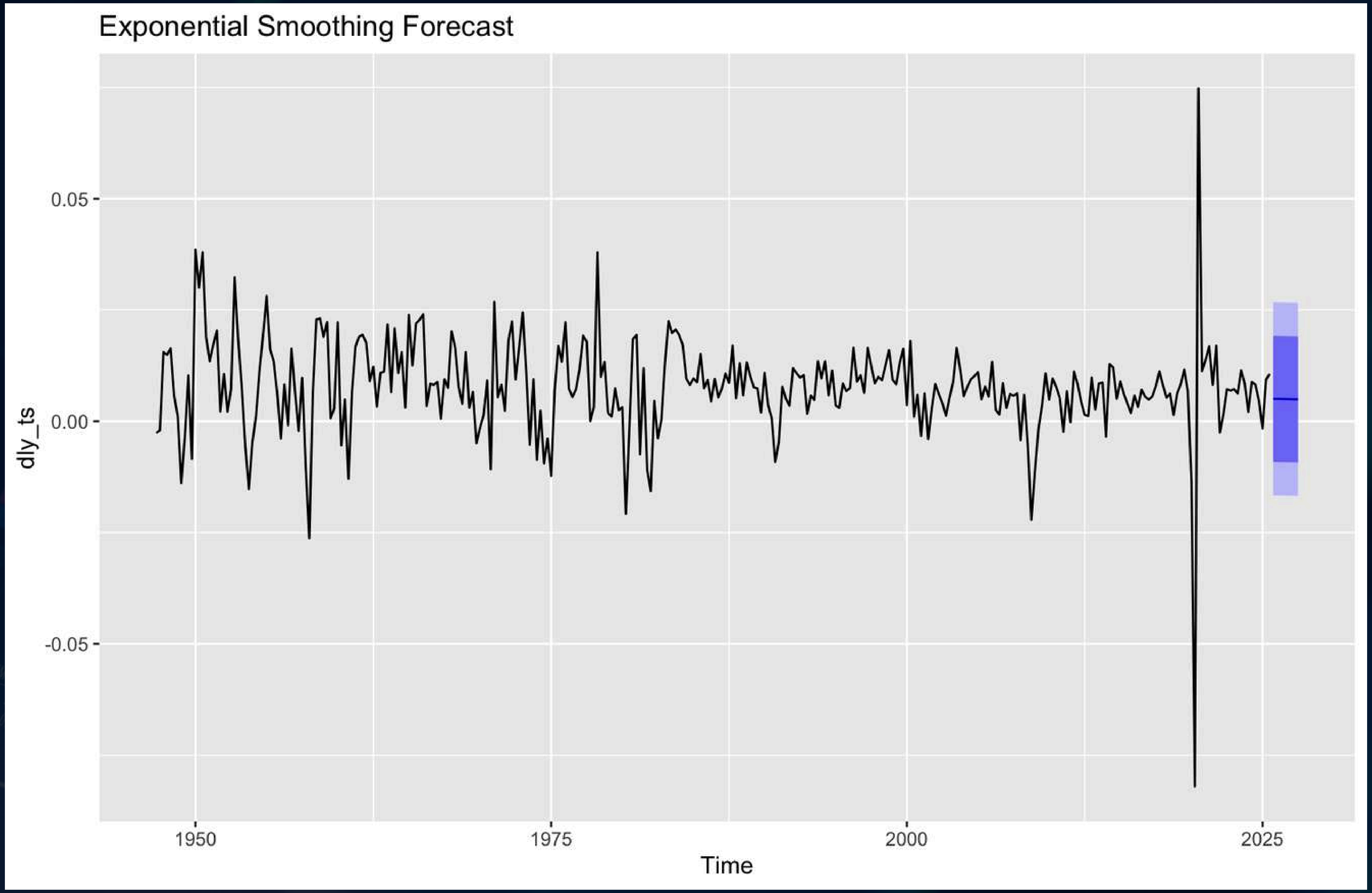


$$\hat{Y}_{t+h} = Y_t$$

- Assumes next period equals the last observation
- No dynamics nor learning
- Thus, it's a benchmark, and not a serious economic model

Exponential Smoothing Model – Code Along

Exponential Smoothing Model



$$\hat{Y}_t = \alpha Y_{t-1} + (1-\alpha)\hat{Y}_{t-1}, \quad 0 < \alpha < 1$$

- The mean forecast is pulled toward a smoothed level
- Adapts based on recent data but does not model explicit dependence
- Uncertainty is wider than AR but narrower than naive

Comparing Forecast Performance

```
> accuracy(ar_fc)
                       ME       RMSE         MAE       MPE    MAPE     MASE        ACF1
Training set 1.120362e-05 0.01092293 0.006745737 -150.4536 279.881 0.631716 0.002858389
> accuracy(naive_fc)
                       ME      RMSE         MAE       MPE     MAPE      MASE       ACF1
Training set 4.243959e-05 0.0145873 0.008584066 -288.4681 441.4001 0.8038694 -0.4878931
> accuracy(ets_fc)
                        ME       RMSE         MAE       MPE     MAPE      MASE      ACF1
Training set -2.778331e-05 0.01099098 0.006989973 -116.4864 259.8065 0.6545879 0.1192616
```

Metrics for Comparison → We choose these metrics as percentage-based metrics are not reliable when forecasting growth rates

- **RMSE (Root Mean Squared Error)**
  - Average forecast error with larger errors penalised more heavily

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}$$

- **MAE (Mean Absolute Error)**
  - Average magnitude of forecast errors

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t|$$

$$\text{MASE} = \frac{\frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t|}{\frac{1}{n-1}\sum_{t=2}^{n}|y_t - y_{t-1}|}$$

- **MASE (Mean Absolute Scaled Error)**
  - Forecast error scaled relative to native benchmark (MAE of model / MAE of naive forecast)

ATTENDANCE & FEEDBACK FORM

Please help fill in so we can improve future workshops for you!

SLIDES WILL ALSO BE ON

Github

https://github.com/NUS-SDS-Workshops/AY-25-26-Public

All SDS workshops code and slides will be on there. Do consider ⭐ starring ⭐ to stay updated!

MORE WORKSHOPS COMING THIS SEMESTER 🥳

Week 5: Collab with Math Society - MATLAB Workshop

Week 5: Collab with Product Club - Product Analytics Workshop

Week 6: Azure Cloud Platform for Data Science

Week 9: Collab with AI Society - RL/NLP Workshop

Week 10: MEGA WORKSHOP (will reveal soon)

So follow our telegram @nussds or instagram @nus.sds to stay updated!

Thank You