CSSS/POLS 512 - Time Series and Panel Data Methods

Lab 2: Time Series Diagnostics

Ramses Llobet

Agenda

- Box-Jenkins Method
- Time Series Diagnostics
  - Deterministic Trend and De-trend
  - Detect Seasonality
  - Discover Autocorrelation in Time Series
  - Moving average processes
  - Estimating dynamic models and residuals test

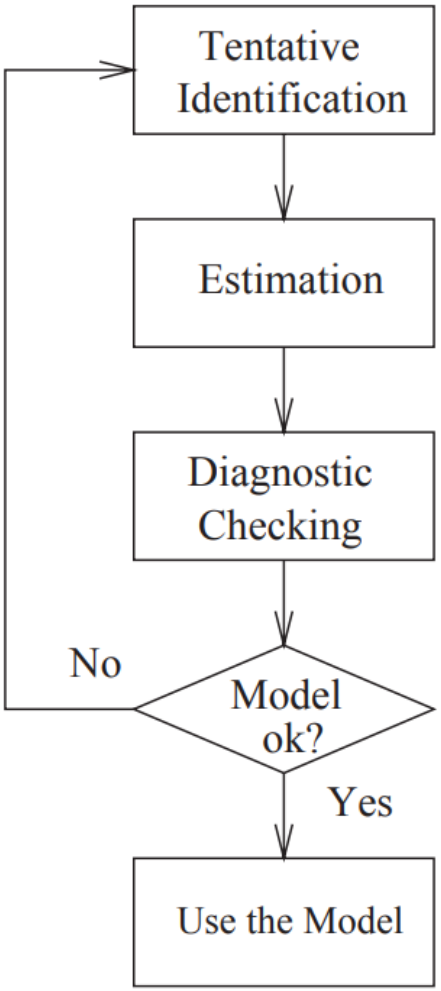
Box-Jenkins Method

- The Box-Jenkins method assumes that time series are composed by multiple temporal processes.

$$y_t = \beta_0 + \underbrace{\beta_1}_{\text{trend}} t + \overbrace{\phi_1}^{\text{AR(1)}} y_{t-1} + \underbrace{\phi_{12}}_{\text{cycle}} y_{t-12} + \overbrace{\theta_1}^{\text{MA(1)}} \varepsilon_{t-1} + \underbrace{\varepsilon_t}_{\text{white noises}}$$

- It then performs diagnostics to compare the observed series with generic forms to decide what processes occur in the data.

Box-Jenkins Method



Time Series Plot
Range-Mean Plot
ACF and PACF

Least Squares or
Maximum Likelihood

Residual Analysis
and Forecasts

Forecasting
Explanation
Control

Deterministic Trend

$$y_t = \beta_0 + \beta_1 t + \varepsilon_t, \quad \text{where} \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2)$$

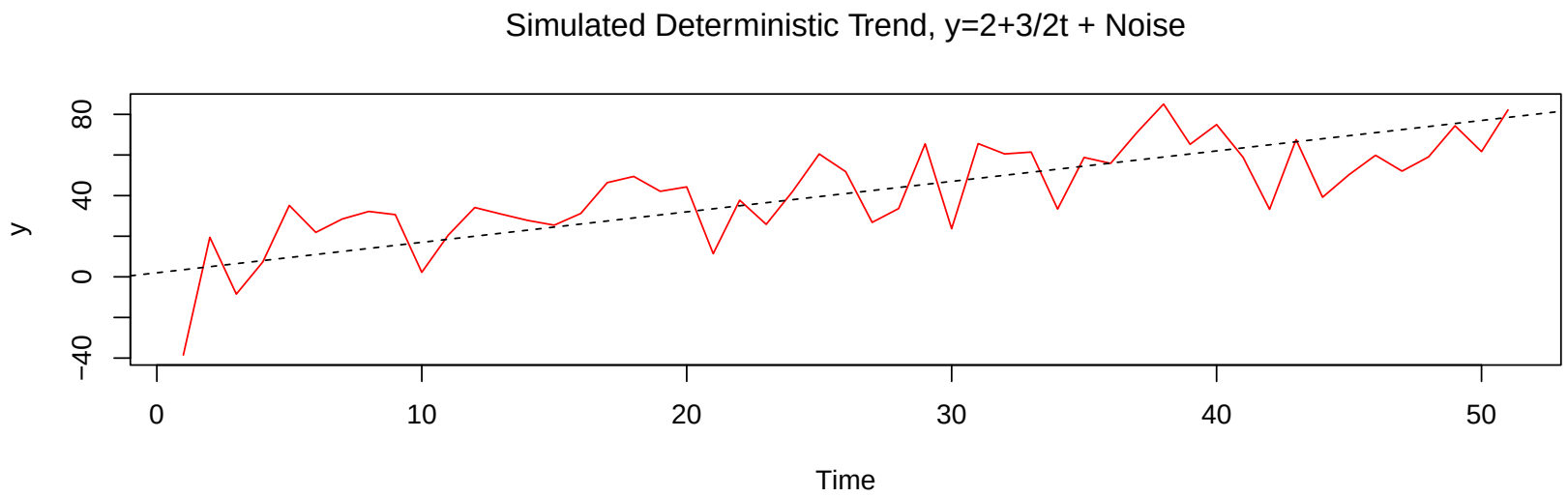
- For every one period increase in t, $\mathbb{E}(y_t)$ increases in β_1.
- In this DGP, y's dynamic process follows a linear systematic relationship.
- In the sample, once the time series is detrended, the estimated errors follow *white noise*.

$$y_t - t\hat{\beta}_1 = \hat{\beta}_0 + \hat{e}_t, \quad \text{where} \quad \hat{e}_t \sim \mathcal{N}(0, \hat{\sigma}^2)$$

Deterministic Trend

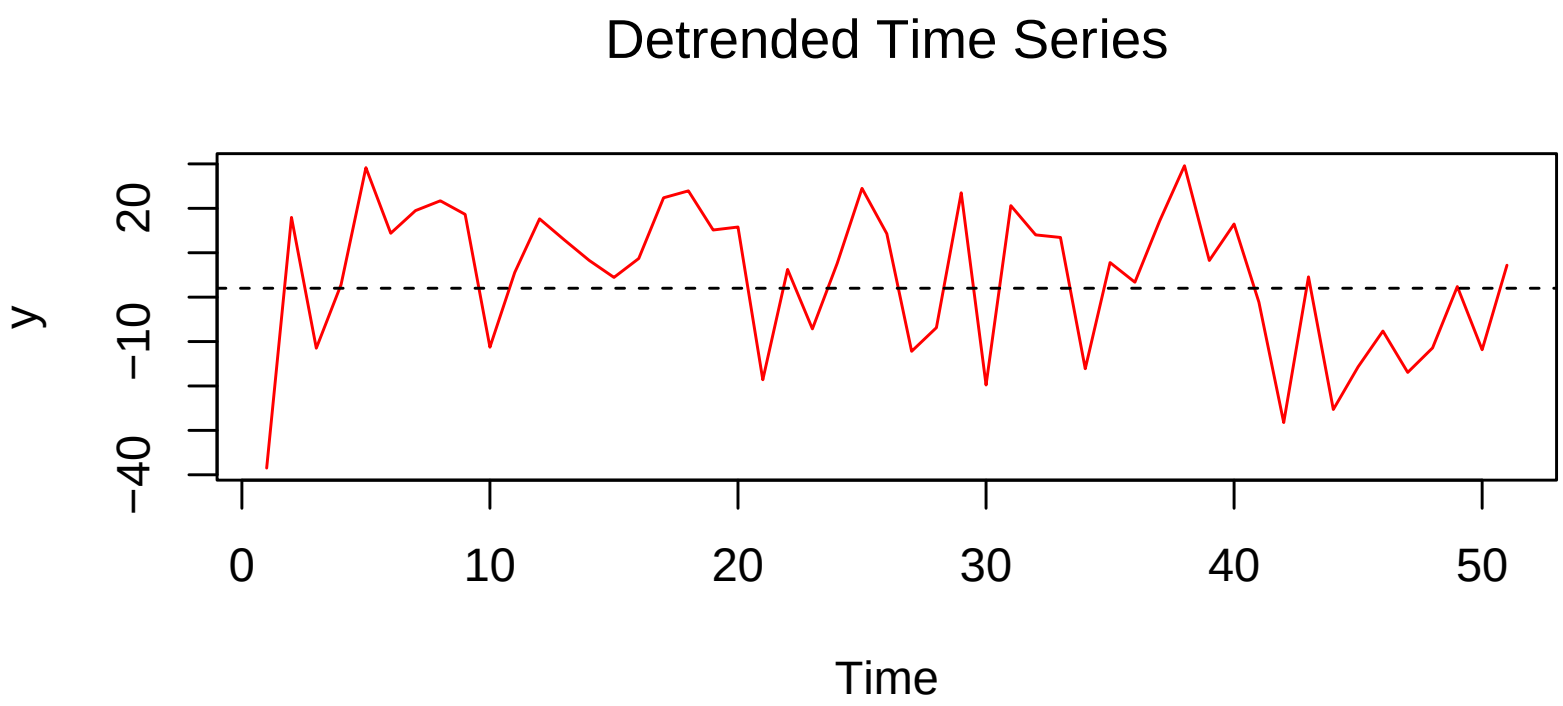
Assume the following population model:

$$y = \beta_0 + \beta_1 t + \varepsilon_t$$
$$y = 2 + 1.5t + \varepsilon_t$$



Deterministic Trend

$$y - 1.5t = 2 + \varepsilon_t$$



Deterministic Trend

```
slope1 <- lm(y~t) # Find the least squares estimate of the slope
slope1
```

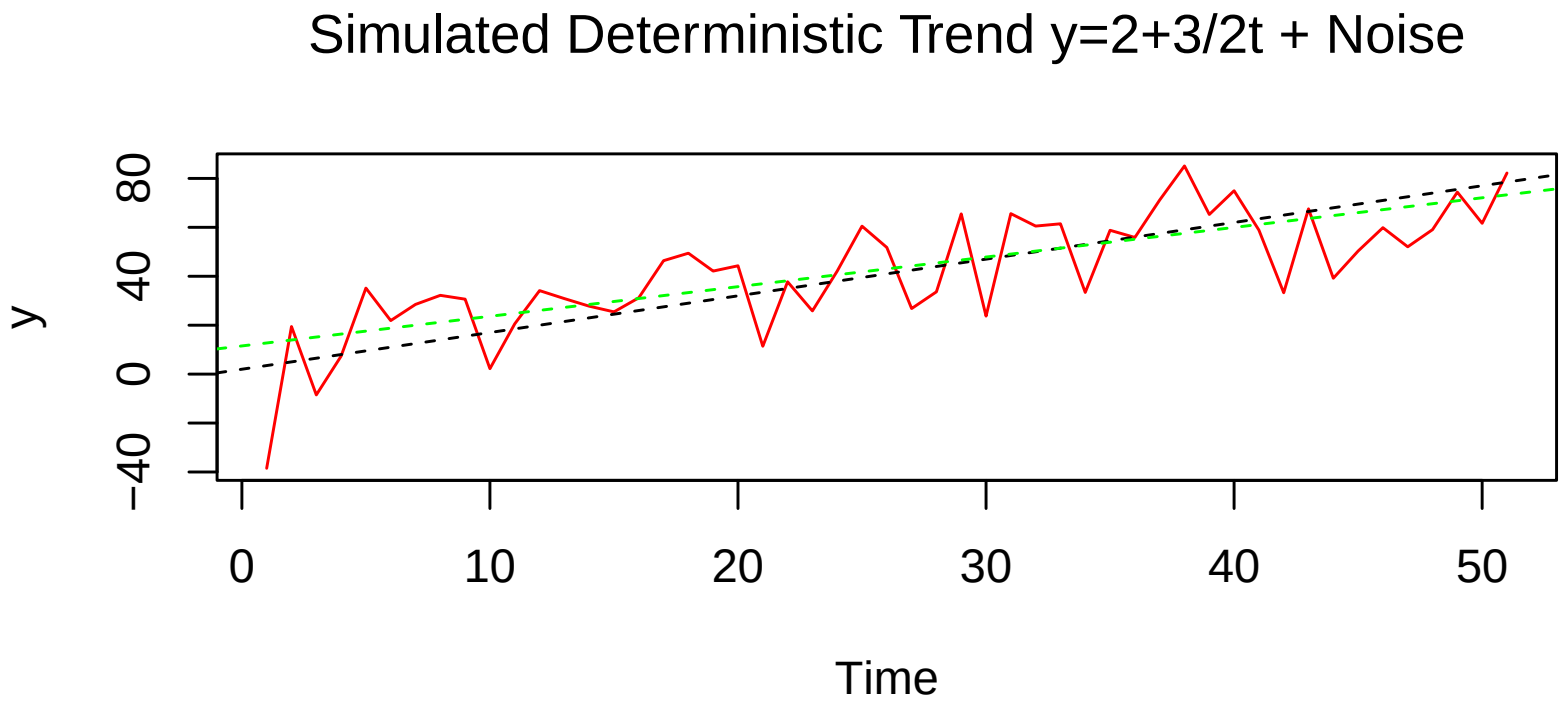
```
##
## Call:
## lm(formula = y ~ t)
##
## Coefficients:
## (Intercept)            t
##      11.529        1.211
```

After estimating the model, we get $\hat{\beta}_0 = 11.52$ and $\hat{\beta}_1 = 1.21$.
How does it compares with the population model?

$$y = 2 + 1.5t + \varepsilon_t$$

Deterministic Trend

Plot the data with the true beta and the estimated beta.

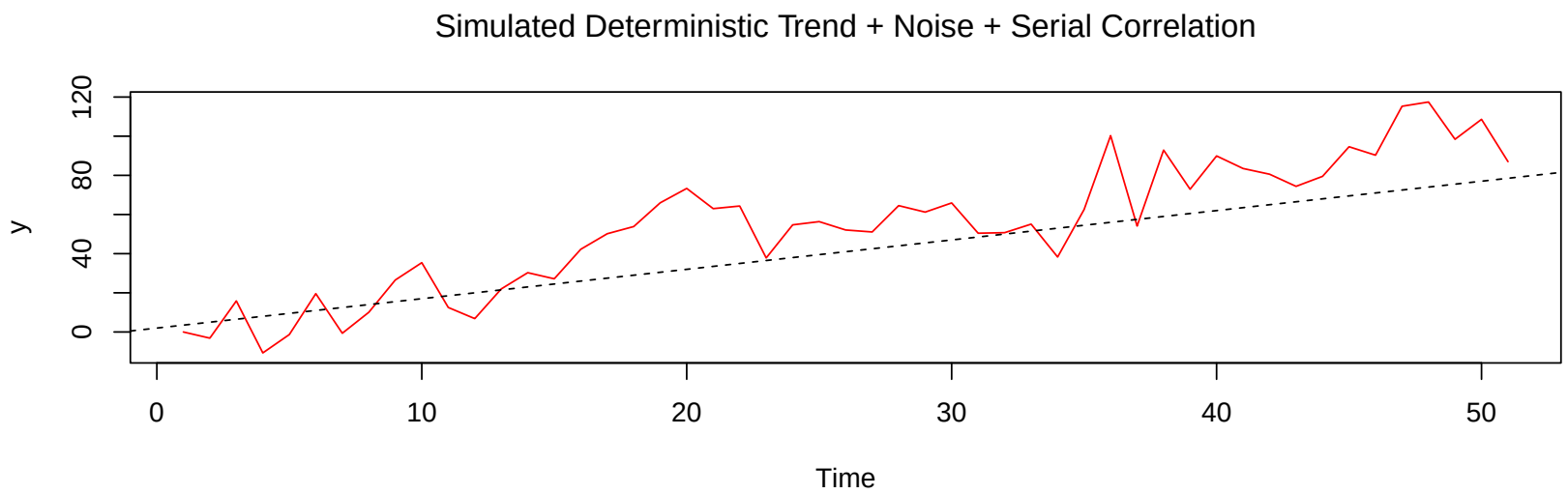


Deterministic Trend and Serial Correlation

Assume the following DGP:

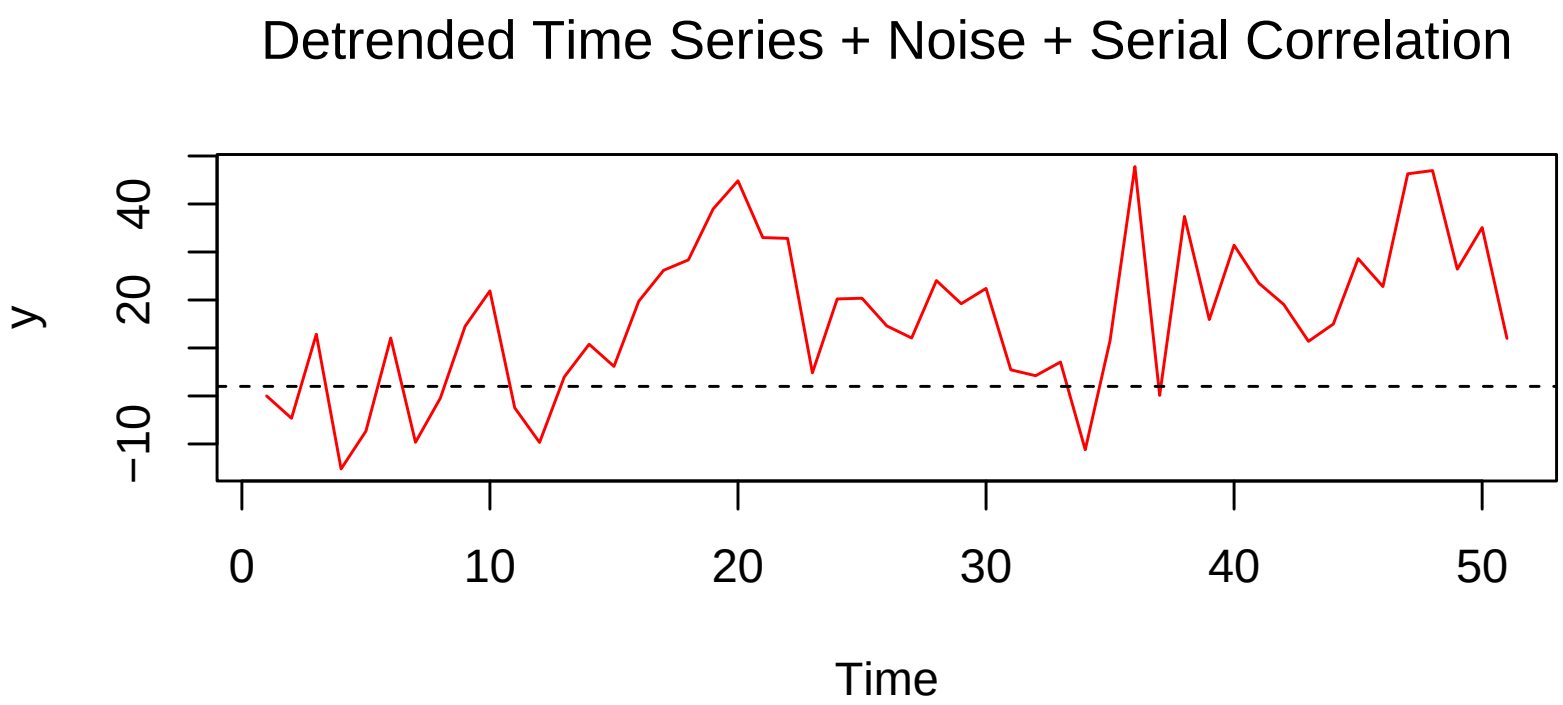
$$y = \beta_0 + \phi_1 y_{t-1} + \beta_1 t + \varepsilon_t$$

$$y = 2 + 0.33 y_{t-1} + 1.5 t + \varepsilon_t$$

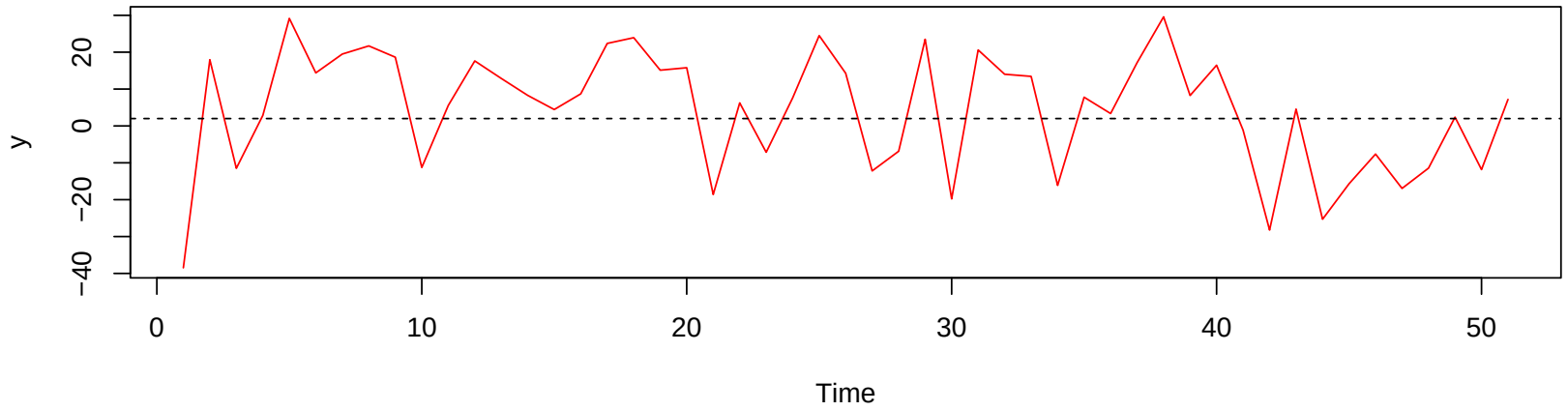


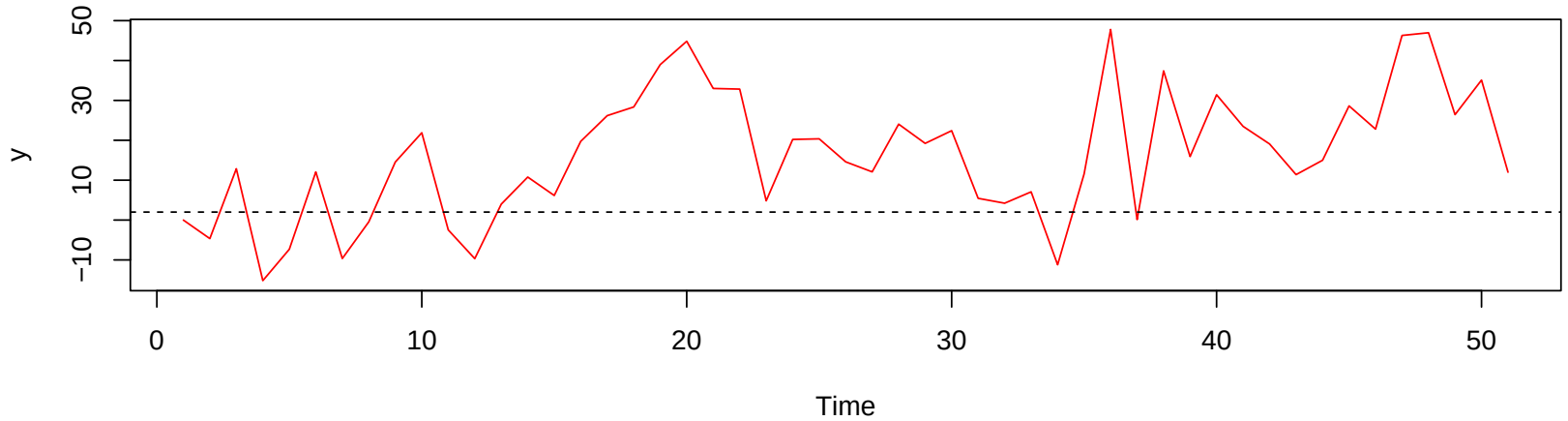
Deterministic Trends and Serial Correlation

After de-trending, serial correlation persists, indicating that the errors e are not white noise ε.



Deterministic Trends and Serial Correlation





Autoregressive Processes

$$y_t = y_{t-1}\phi_1 + \epsilon_t$$

- Past realizations, y_{t-k}, influence current levels of y.
- In the AR(1) case, each new realization of y_t incorporates the last period's realization, y_{t-1}

$$y_t = \sum_{j=0}^{\infty} \epsilon_{t-j}\phi^j$$

- If y_t is AR(1), then y_t includes the effects of every random shock back to the beginning of time.
- When $|\phi_1| < 1$, then with each passing observation, an increasing amount of the shock "leaks" out, but never completely disappears

Analyzing dynamics

When analyzing time series data:

- Begin by plotting the original series using `plot()`.
  - Time series data with trend or seasonal variation exhibits high autocorrelation, potentially biasing the sample autocorrelation function (acf).
  - To address this, **detrend**, detrend the time series data to compute the sample acf.

After detrending the original time series:

- Use the correlogram alongside the $ACF(k)$ and $PACF(k)$ statistics to analyze lag behavior.

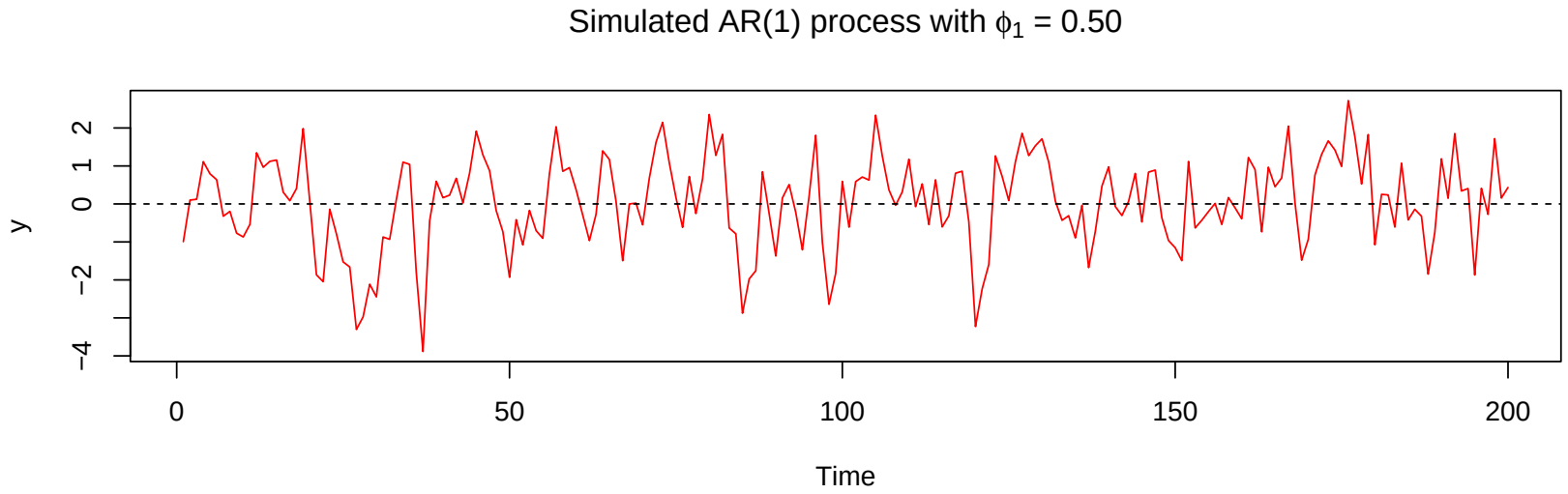
Analyzing dynamics: ACF and PACF

- The autocorrelation function `acf()` measures the correlation between past lags y_{t-k} and the present y_t.
- The partial autocorrelation function `pacf()` quantifies the correlation between y_t and y_{t-k} after accounting for intermediate values.
- Notably, `acf` considers the total variance of y, whereas `pacf` *partials out* or removes the variance between y_t and y_{t-k}.

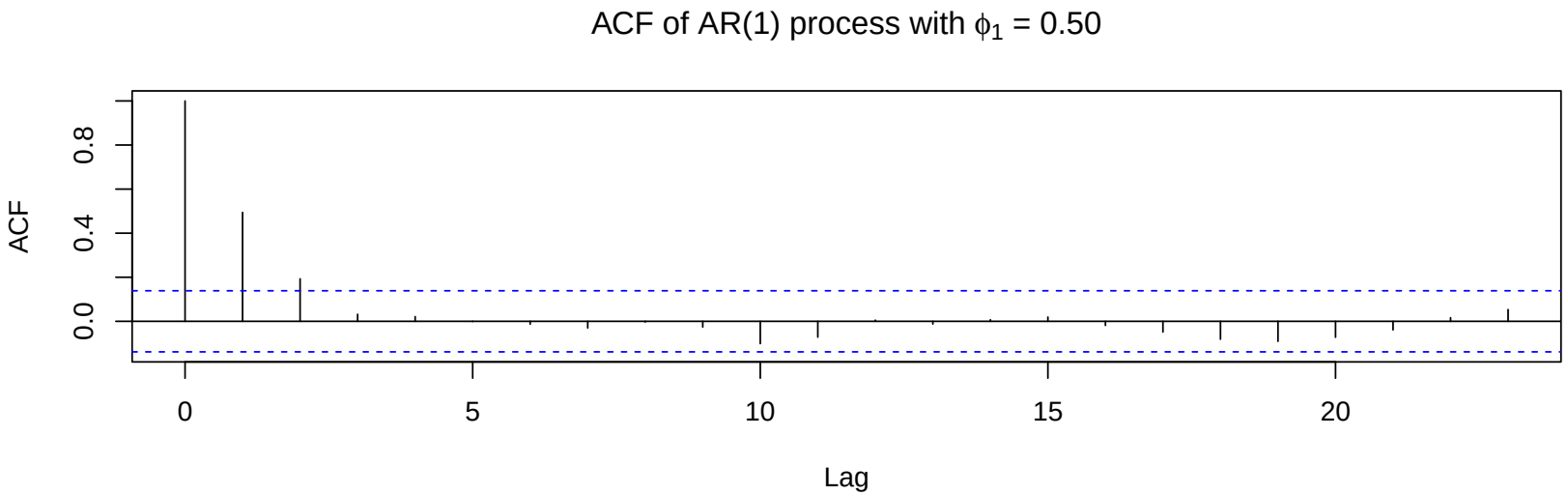
Autoregressive Processes

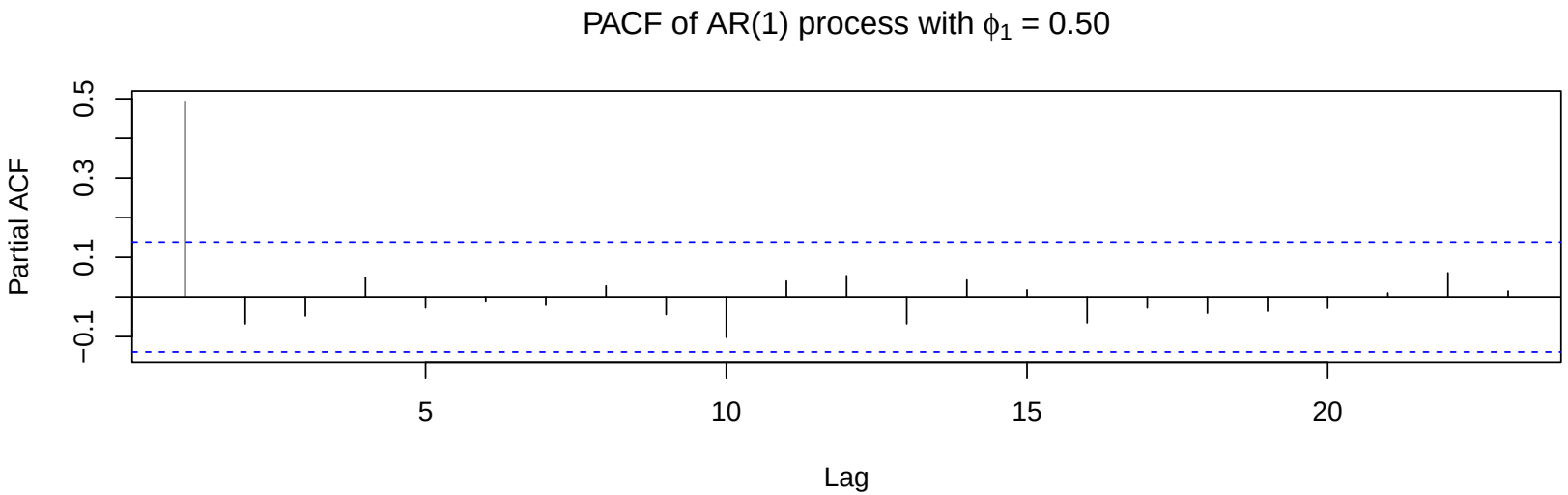
Plot of an AR(1) process with no trend:

- Notice how in the first periods that it takes longer to revert towards the mean.

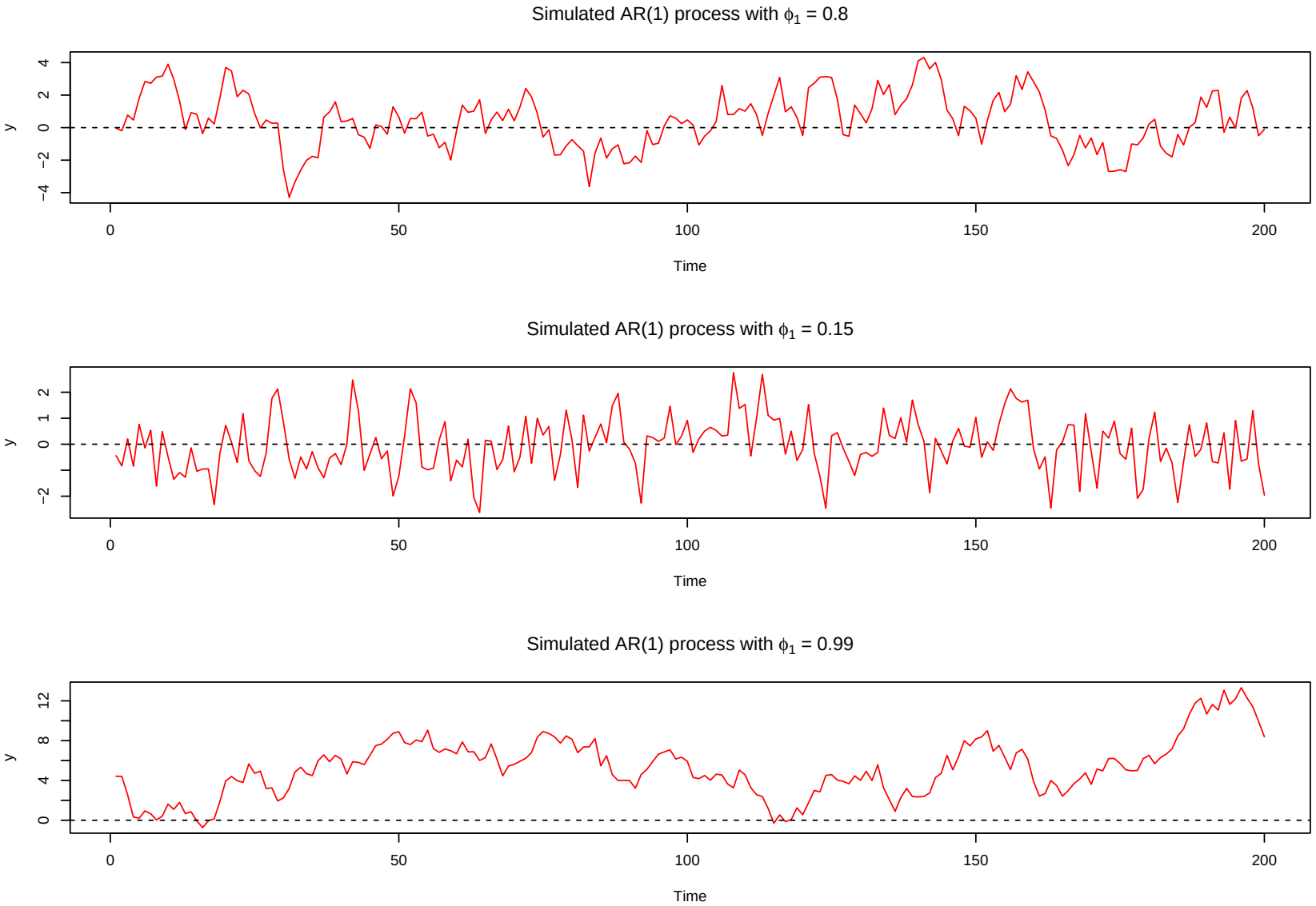


Autoregressive Processes

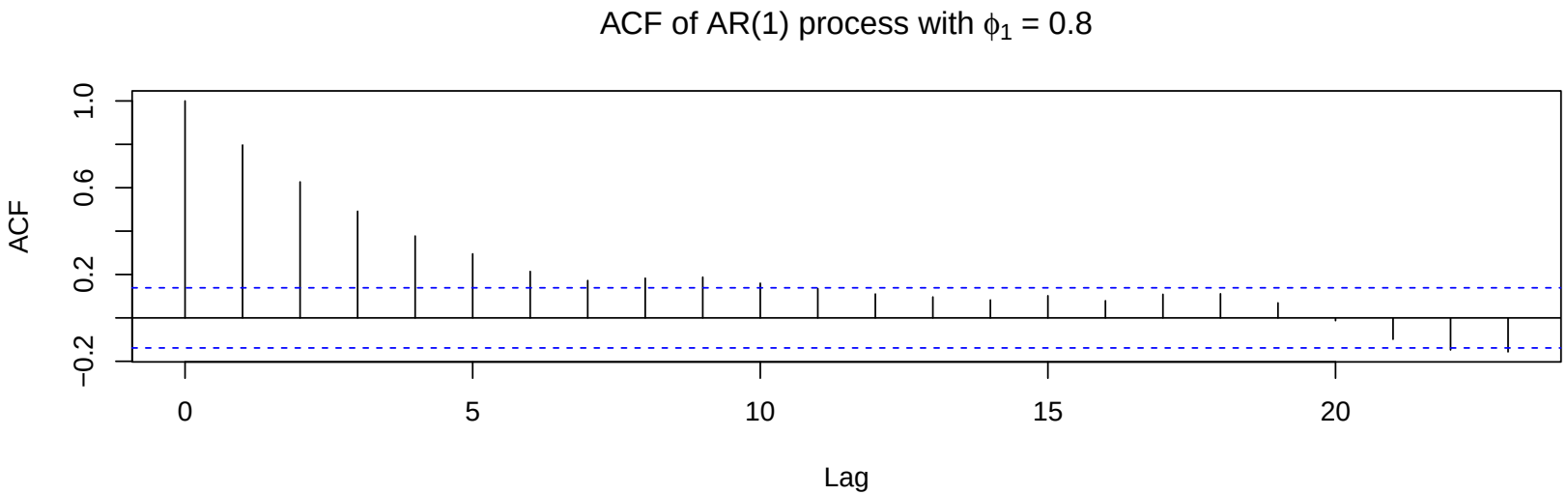


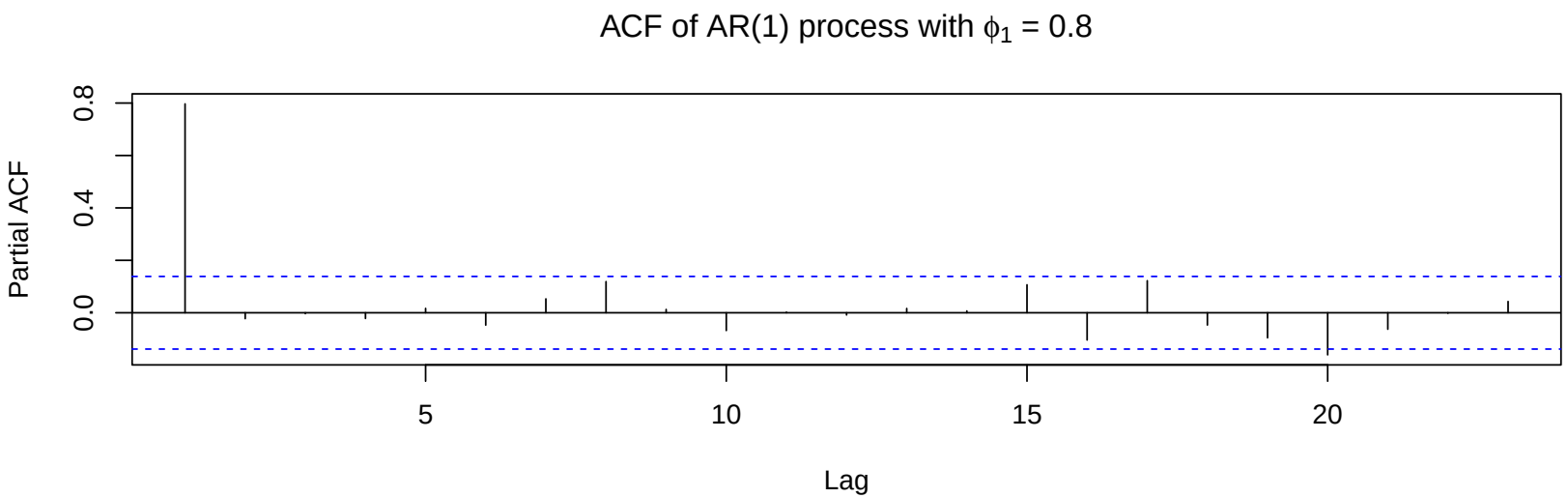


Autoregressive Processes

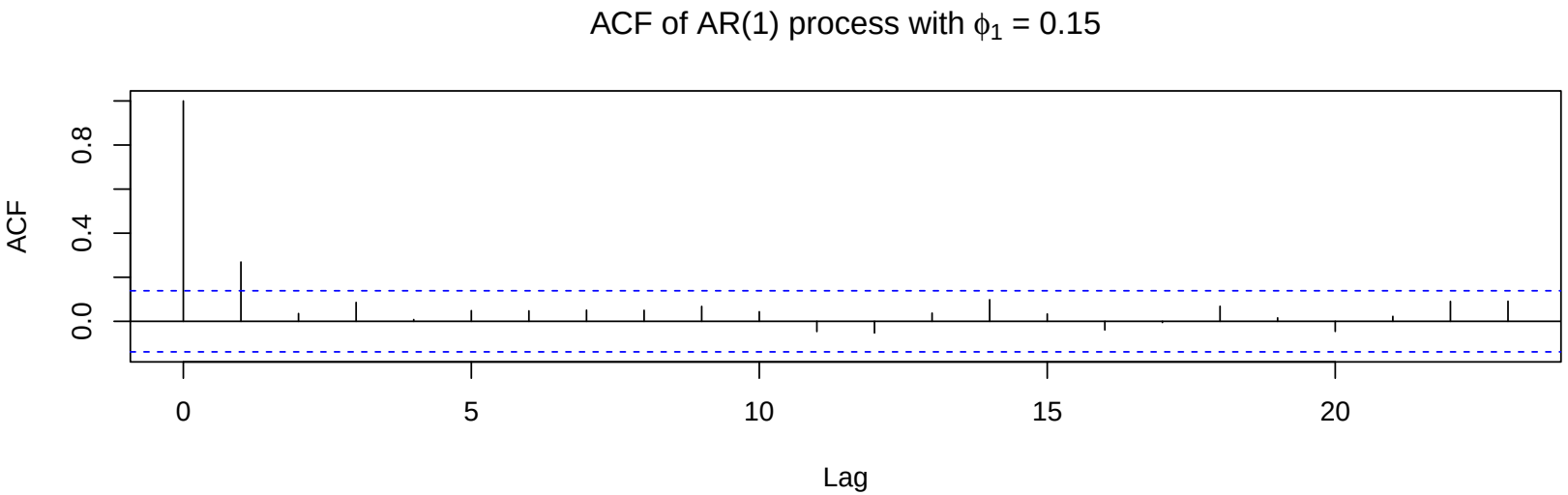


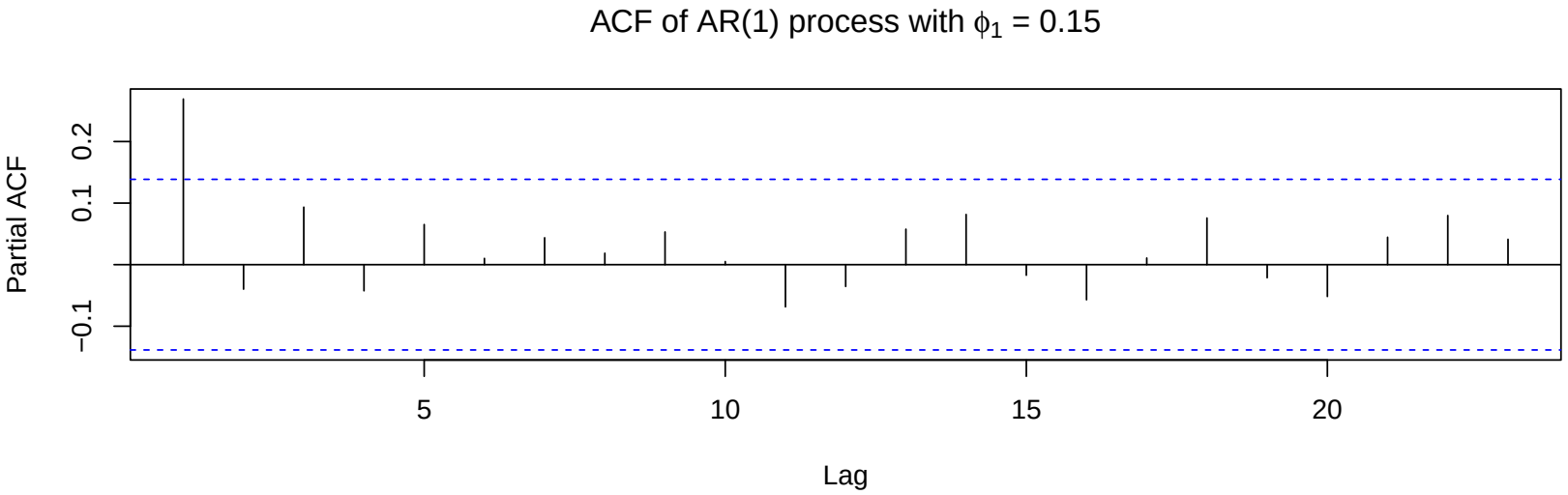
Autoregressive Processes



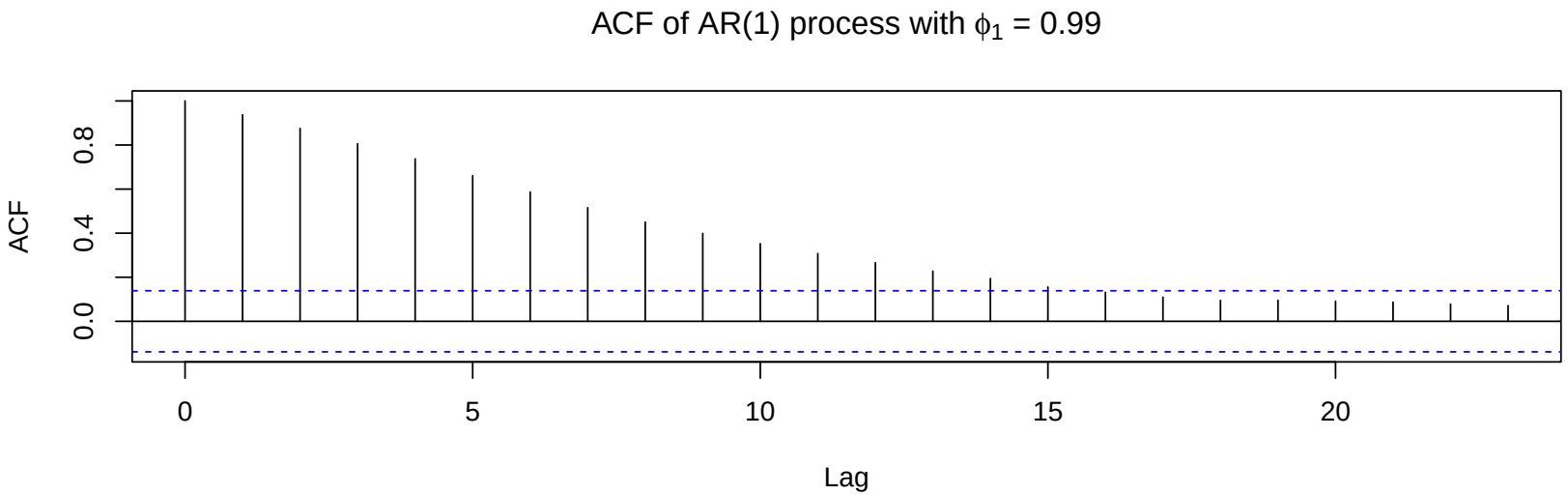


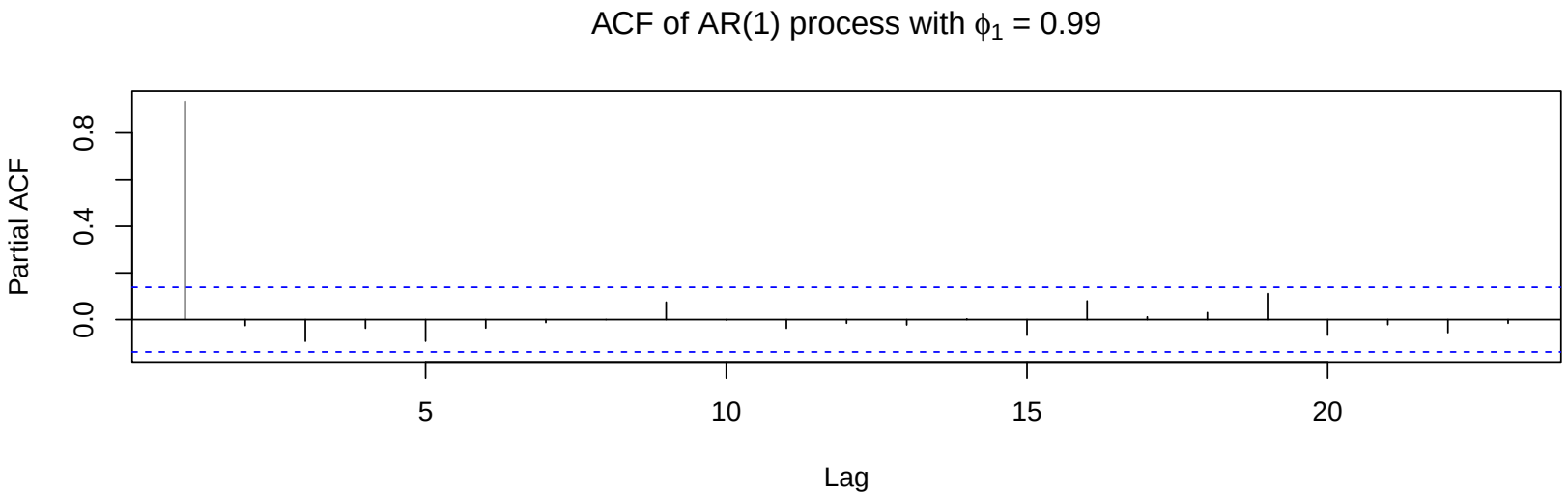
Autoregressive Processes





Autoregressive Processes





Unit Root Tests

$$y_t = \sum_{j=0}^{\infty} \epsilon_{t-j} \phi^j$$

- If y_t is AR(1), then y_t includes the effects of every random shock back to the beginning of time
- When $|\phi_1| = 1$, then we have a **random walk** or unit root, and the impact of the random shocks accumulate over time rather than dissipate
- The mean of the time series is time dependent (non-stationary)

Unit Root Tests

```r
#Check for a unit root on one of the AR(1) processes

#Perform a Phillips-Perron test or Augmented Dickey-Fuller test
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.3.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

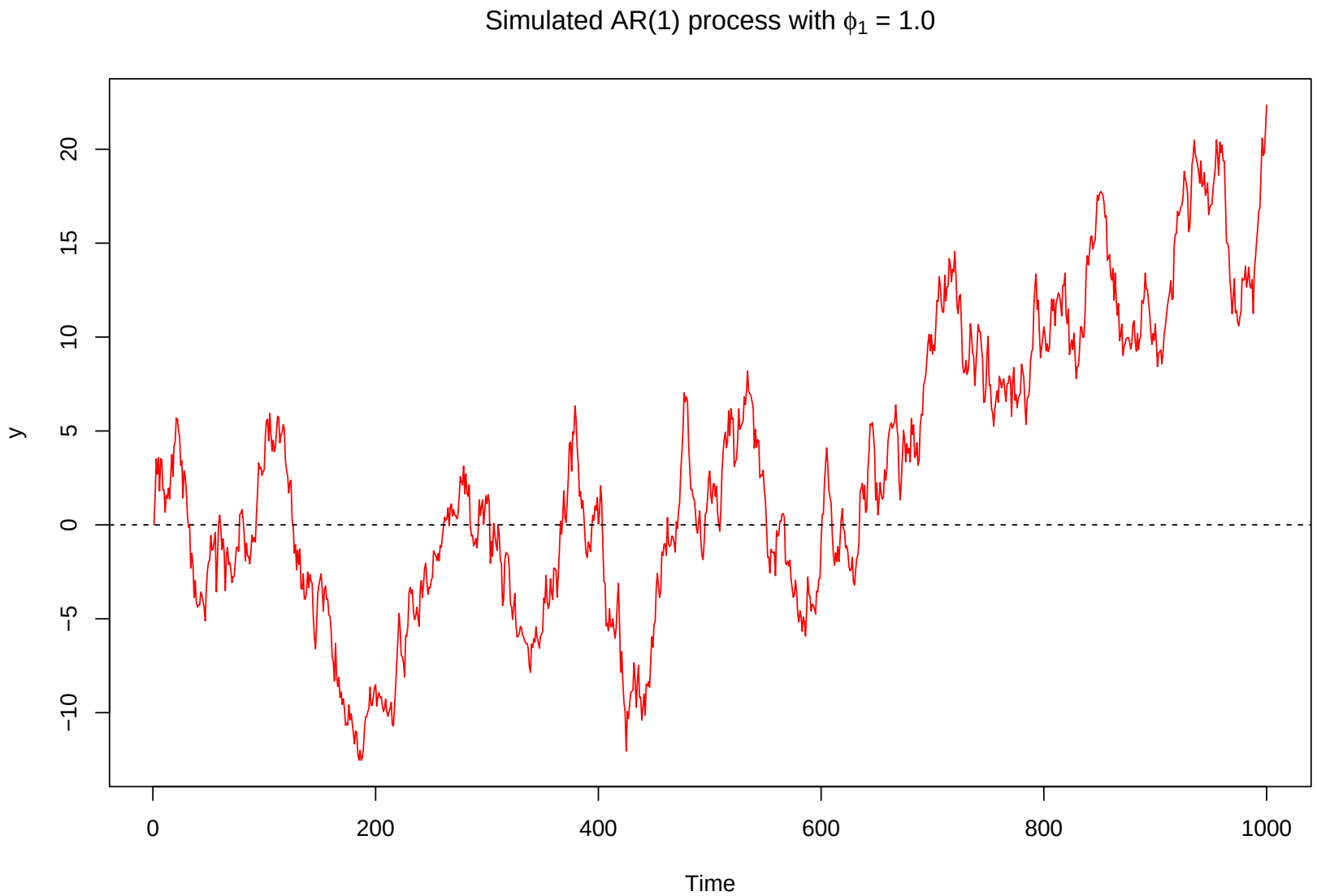
```r
PP.test(ar1.1)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  ar1.1
## Dickey-Fuller = -4.748, Truncation lag parameter = 4, p-value = 0.01
```

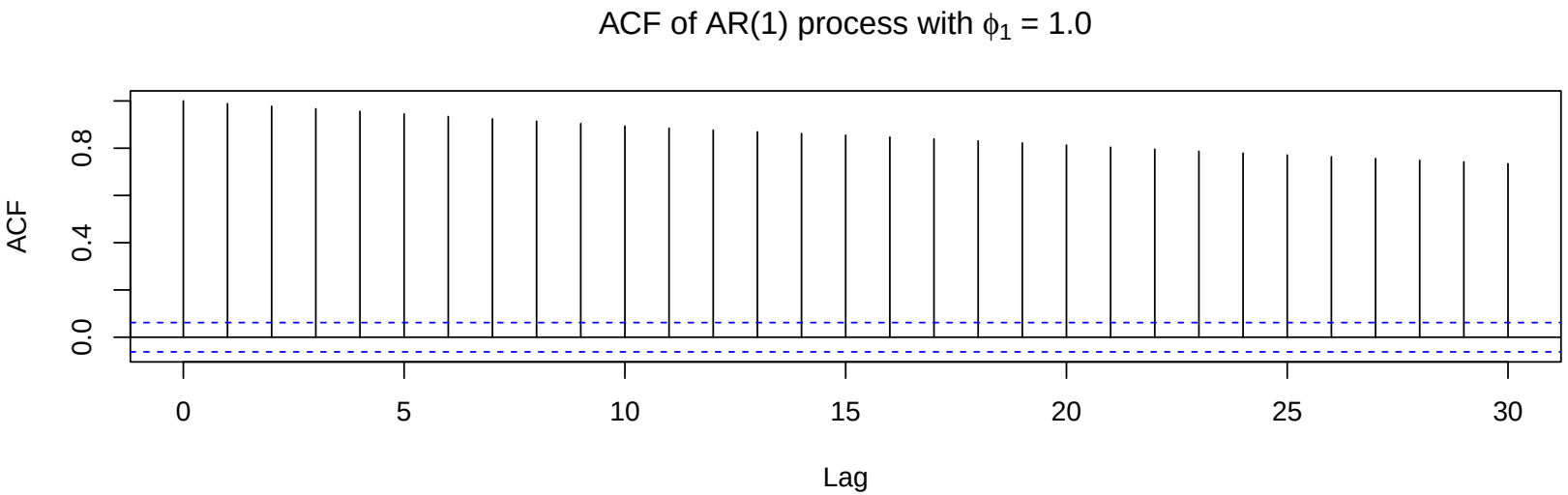
```r
adf.test(ar1.1)
```

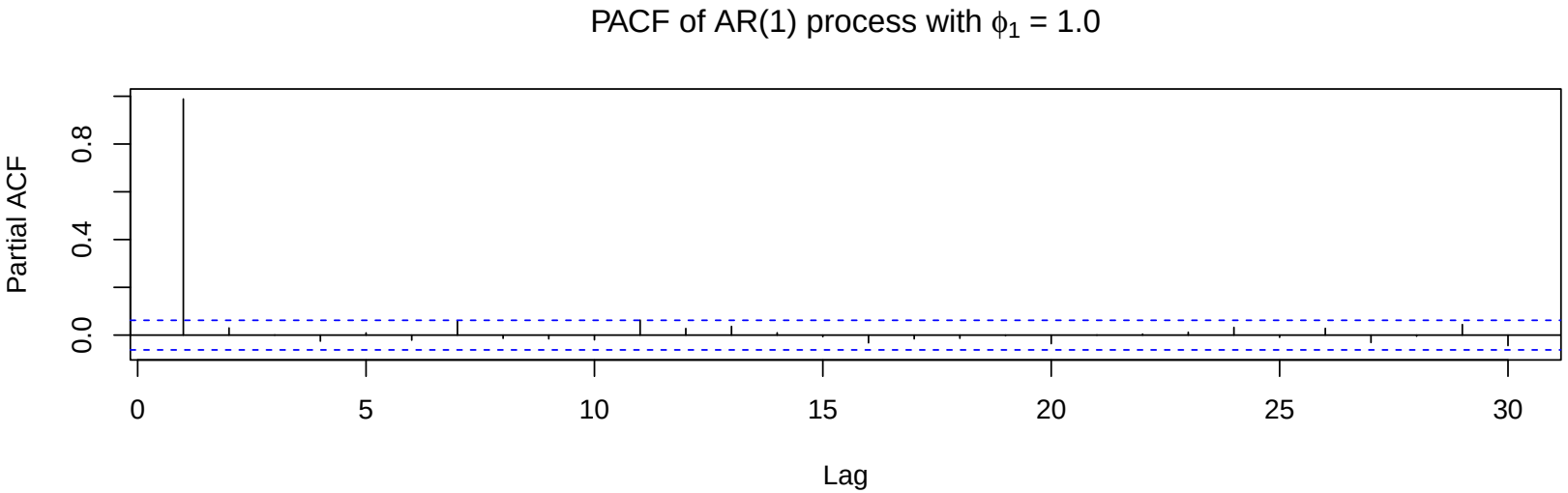
```
##
##  Augmented Dickey-Fuller Test
##
## data:  ar1.1
## Dickey-Fuller = -3.9789, Lag order = 5, p-value = 0.01139
## alternative hypothesis: stationary
```

Autoregressive Processes



Autoregressive Processes





Unit Root Tests

```
#Perform a unit root test on the data

#Perform a Phillips-Perron test or Augmented Dickey-Fuller test
PP.test(ar1.4)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  ar1.4
## Dickey-Fuller = -3.0827, Truncation lag parameter = 7, p-value = 0.12
```

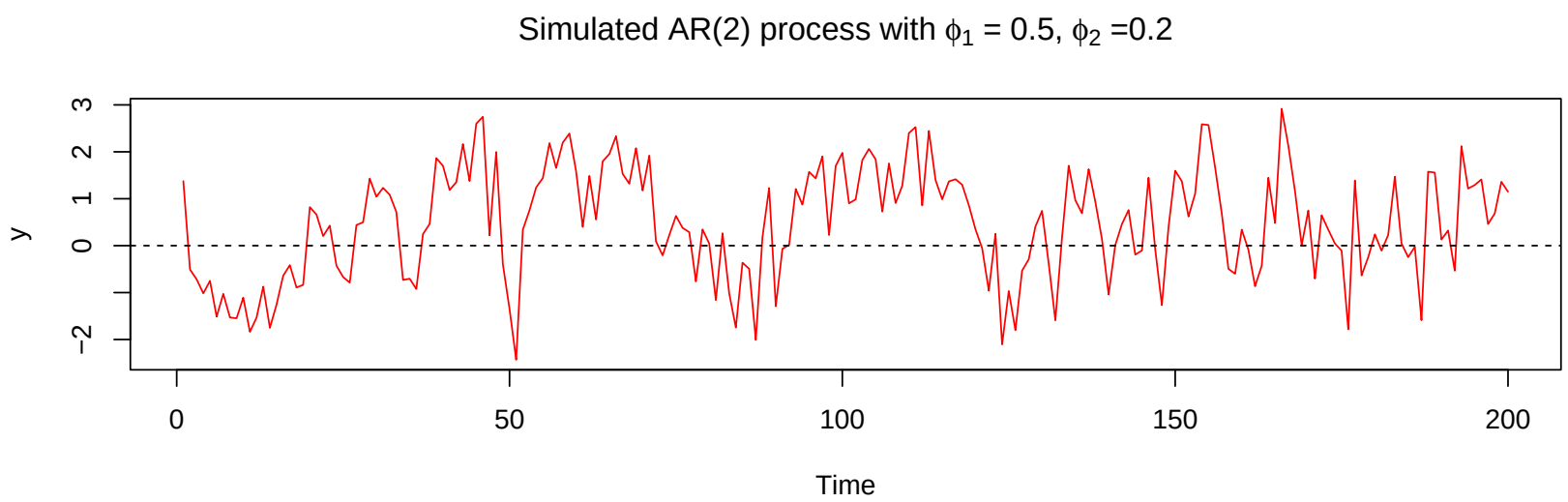
```
adf.test(ar1.4)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ar1.4
## Dickey-Fuller = -3.4858, Lag order = 9, p-value = 0.04367
## alternative hypothesis: stationary
```

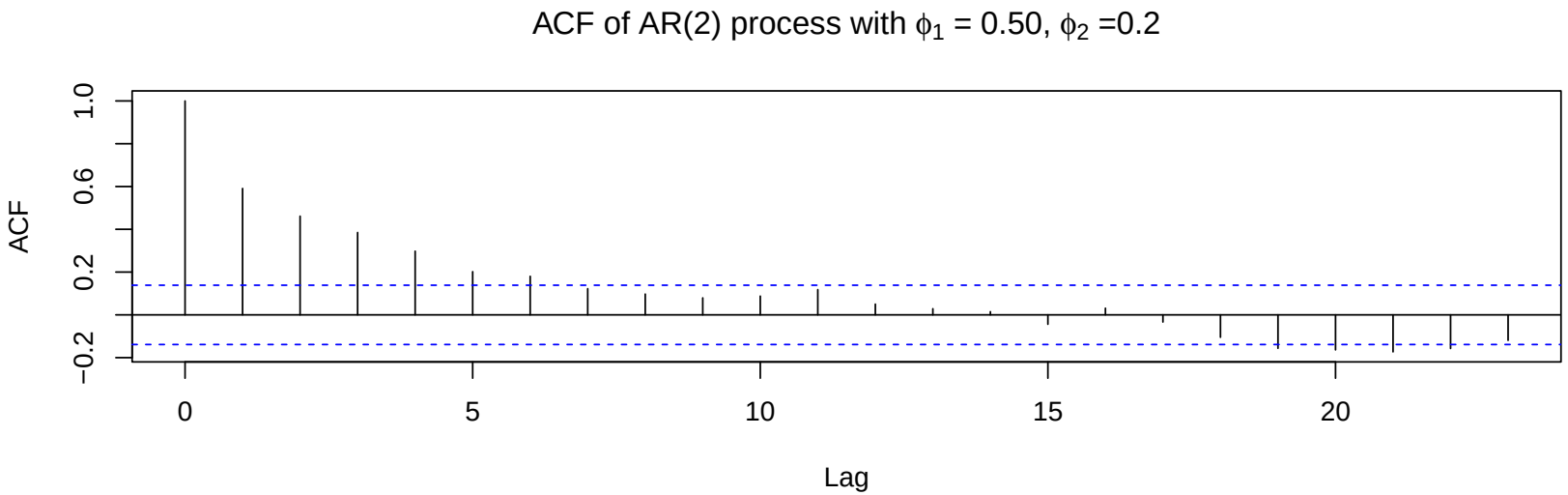
Autoregressive Processes

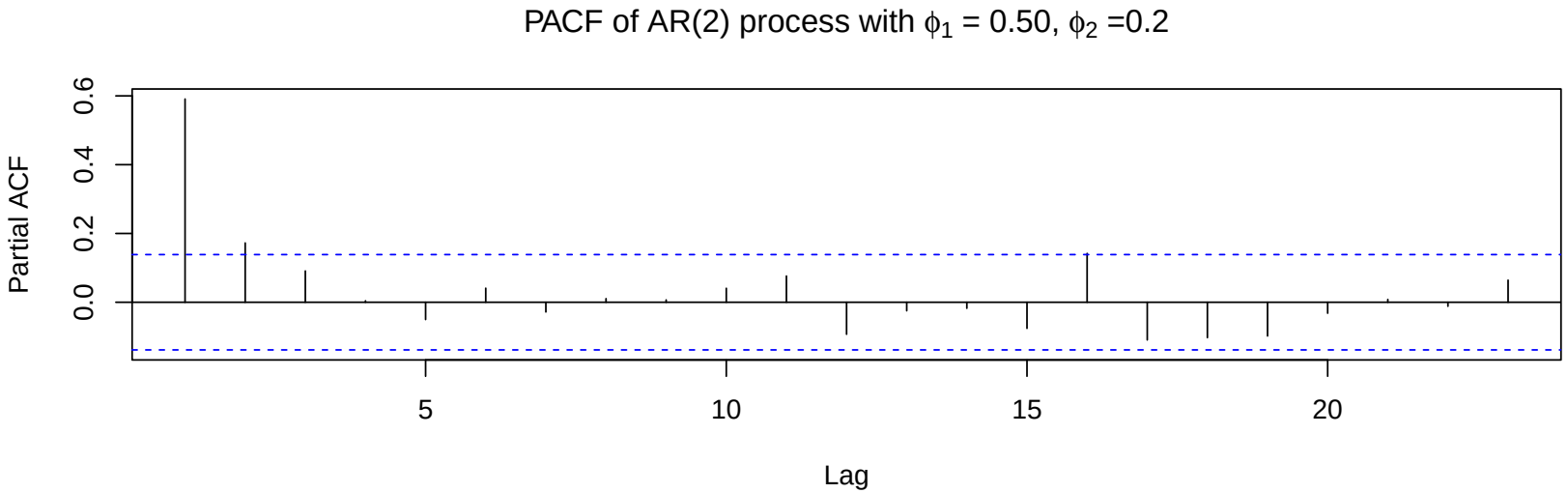
Let's simulate an AR(2) process:

$$y_t = \beta_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t$$



Autoregressive Processes





Unit Root Tests

```
#Is the time series stationary?

#Confirm results with a unit root test
PP.test(ar2.1)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  ar2.1
## Dickey-Fuller = -7.3721, Truncation lag parameter = 4, p-value = 0.01
```

```
adf.test(ar2.1)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ar2.1
## Dickey-Fuller = -3.9686, Lag order = 5, p-value = 0.0119
## alternative hypothesis: stationary
```

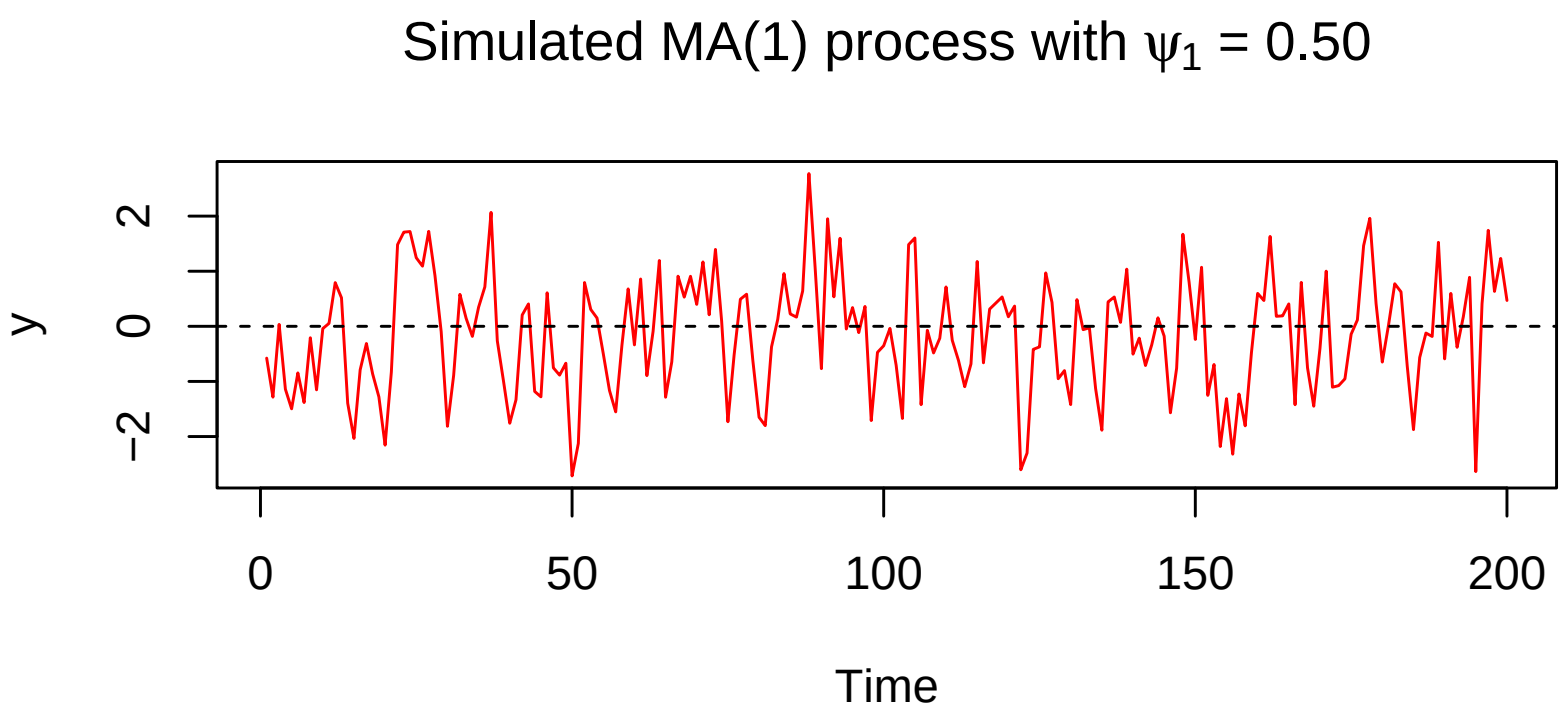
Moving Average Processes

$$y_t = \psi_1 \epsilon_{t-1} + \epsilon_t$$

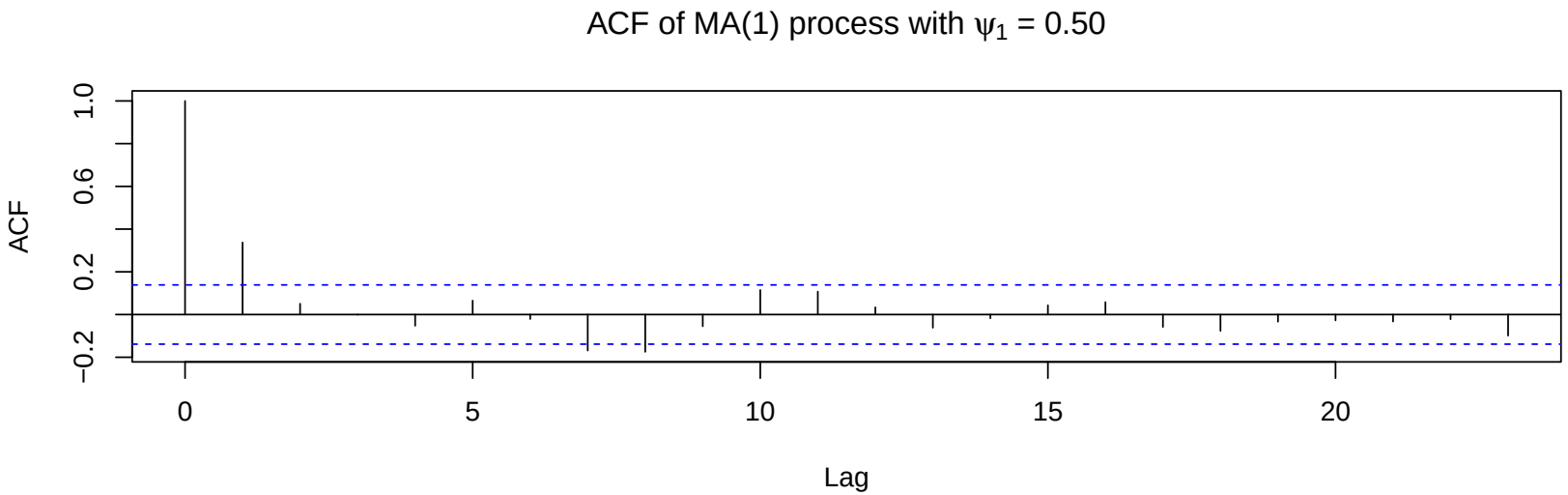
- Past random shocks, ϵ_{t-k}, influence current levels of y
- If y_t is MA(1), then the stochastic component is a weighted average of the current and previous error
- In an MA(q) process, the effects of past shocks die out after q periods
- MA(q) processes are always stationary for finite q

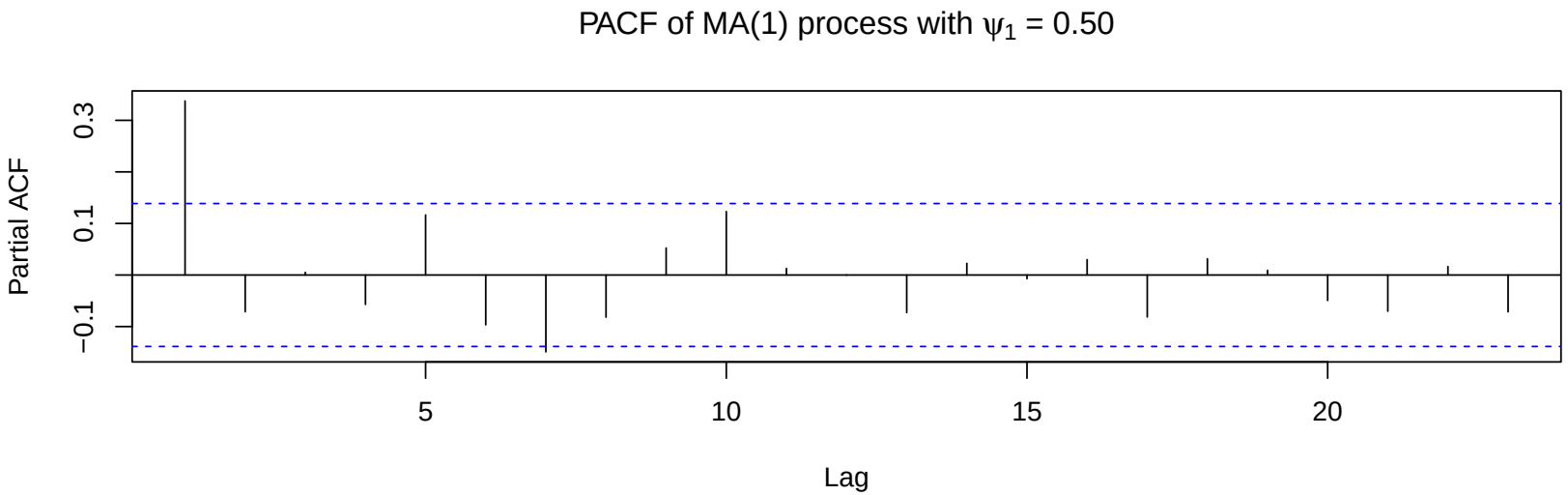
Moving Average Processes

$$y_t = 0.5\epsilon_{t-1} + \epsilon_t$$

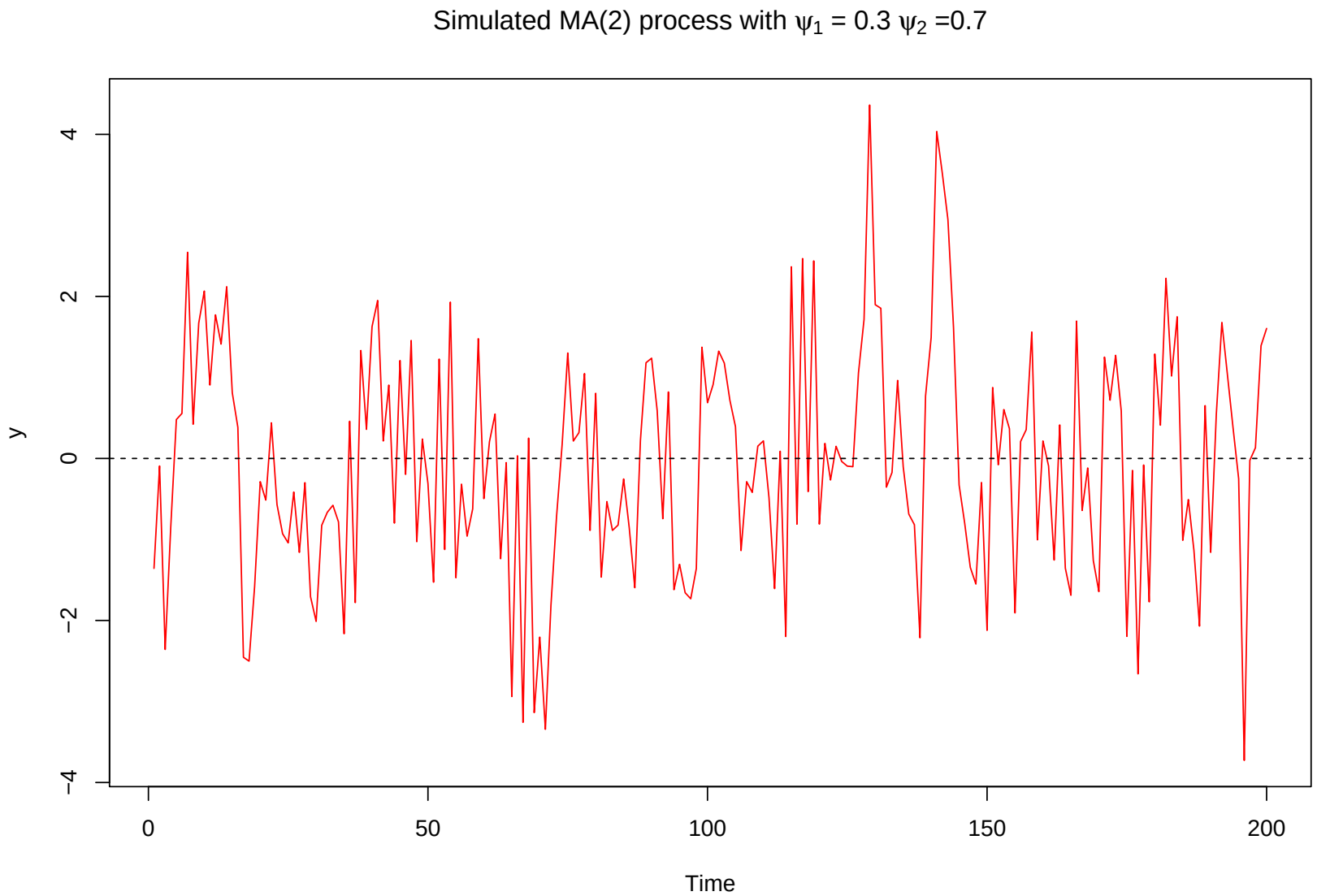


Moving Average Processes

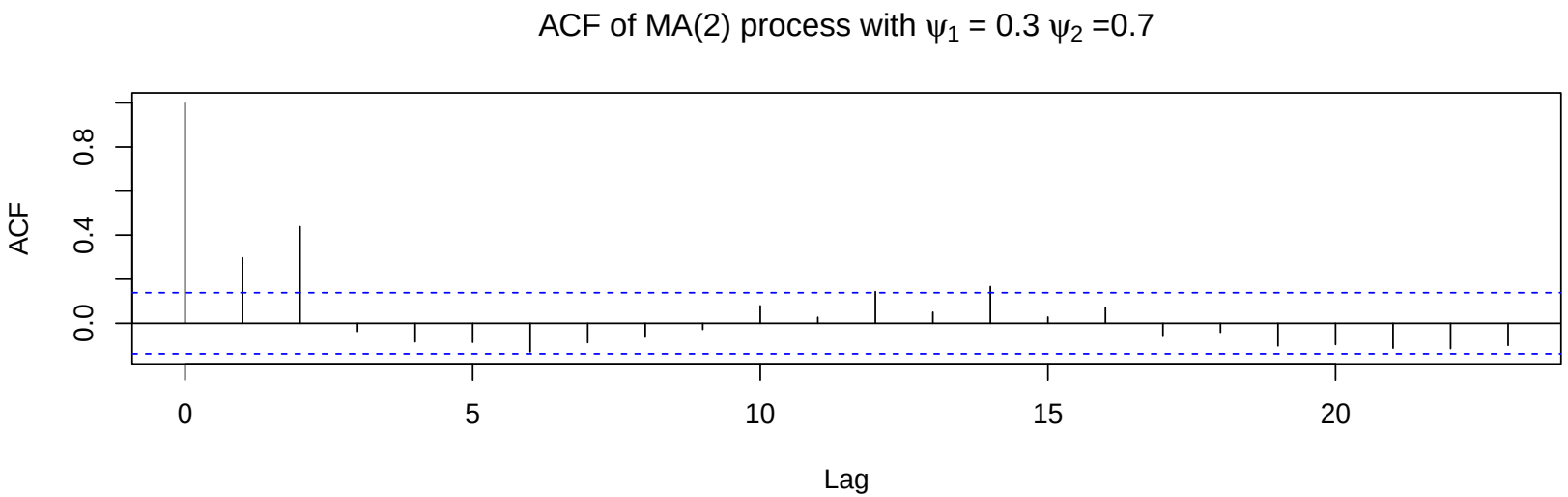


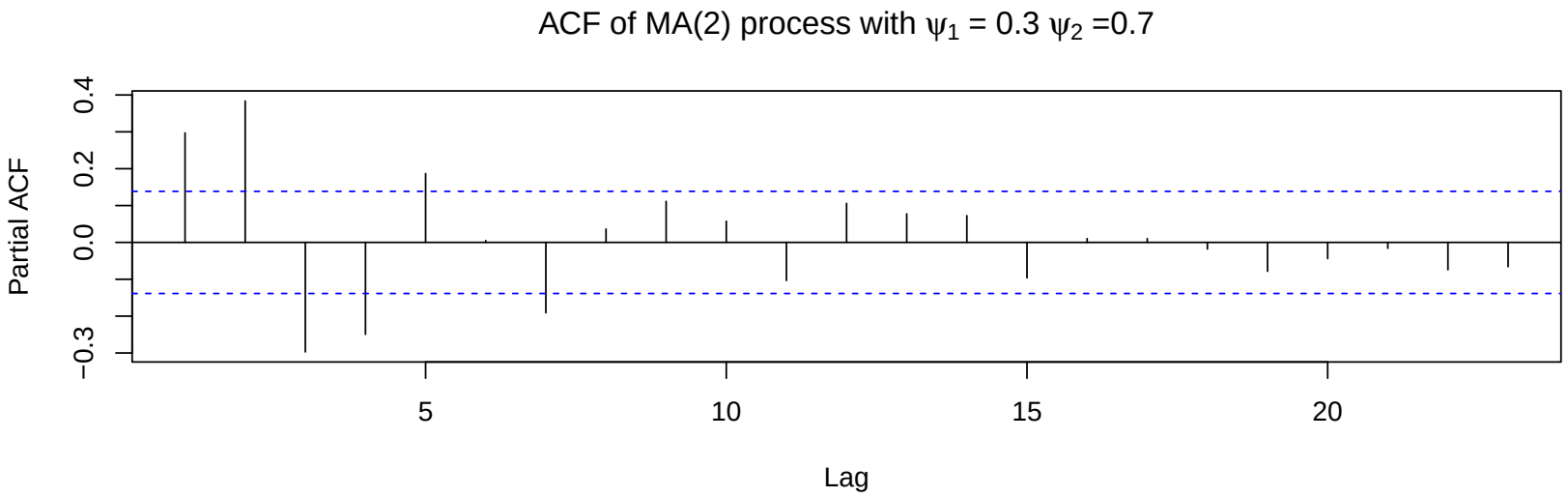


Moving Average Processes

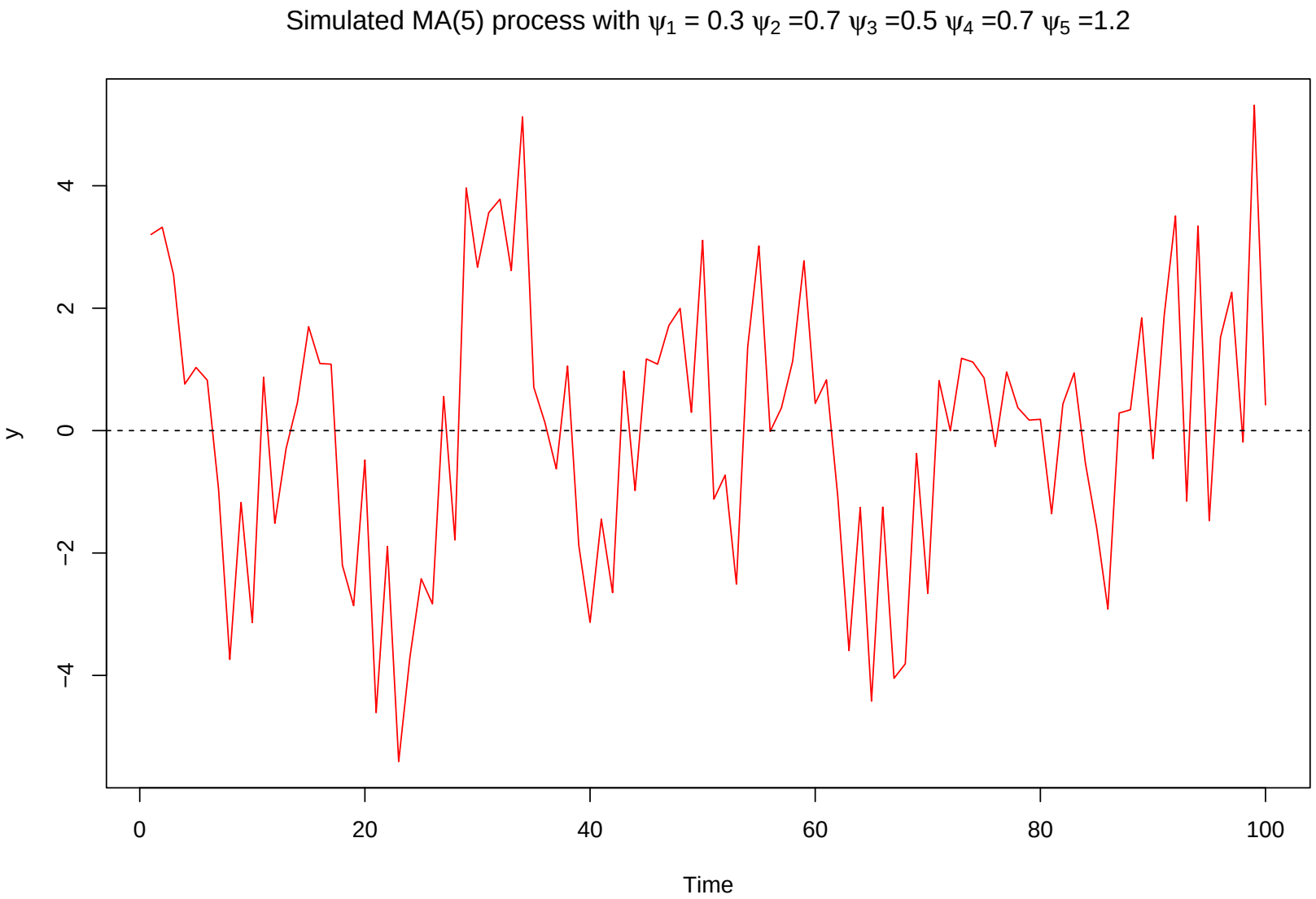


Moving Average Processes

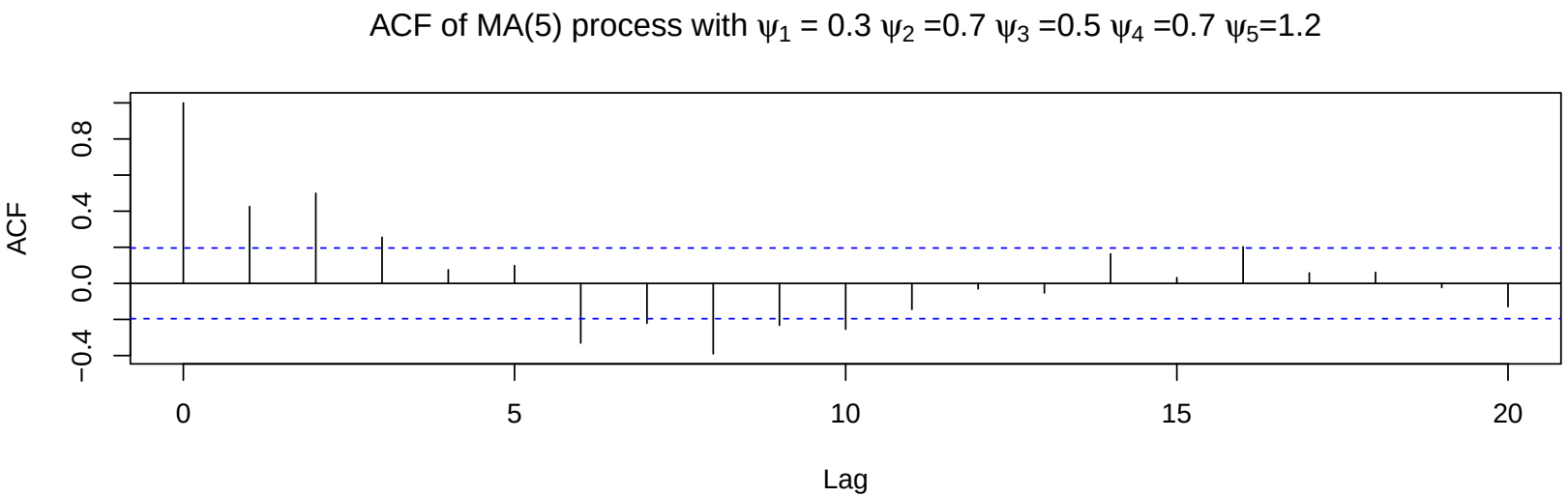


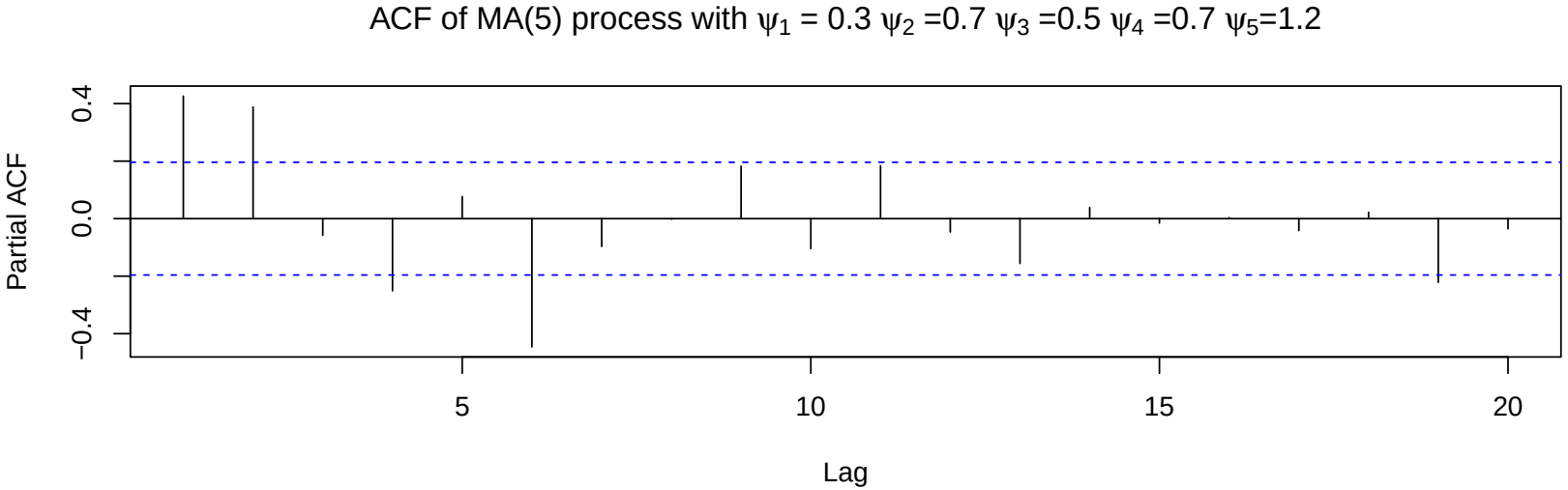


Moving Average Processes



Moving Average Processes



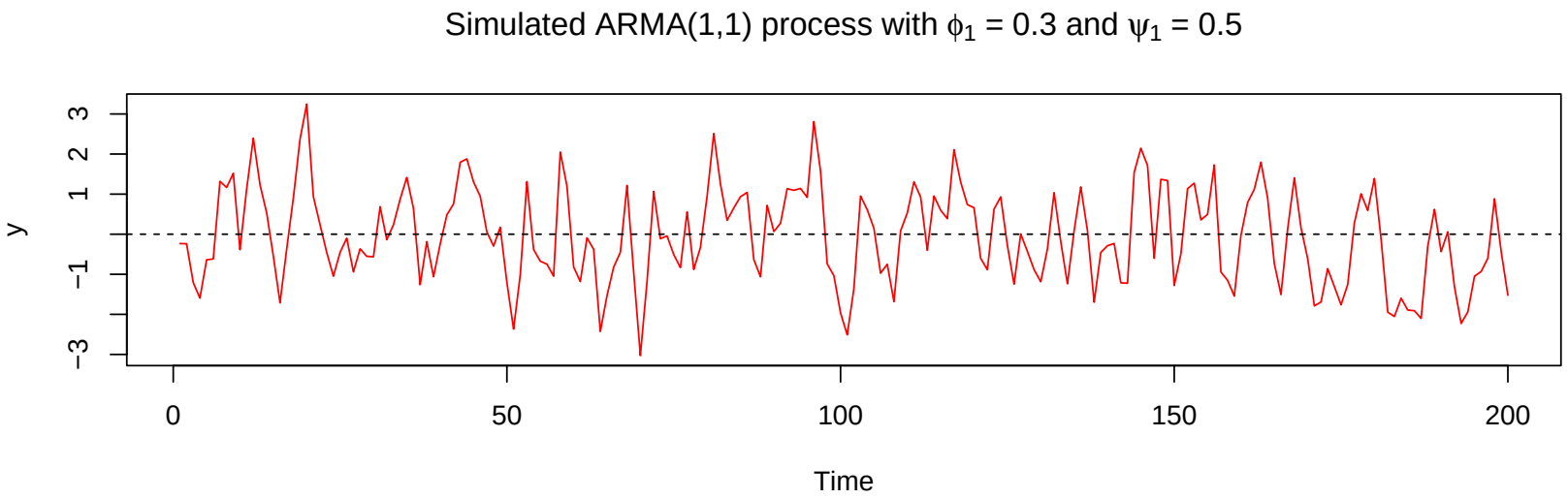


Moving Average Processes

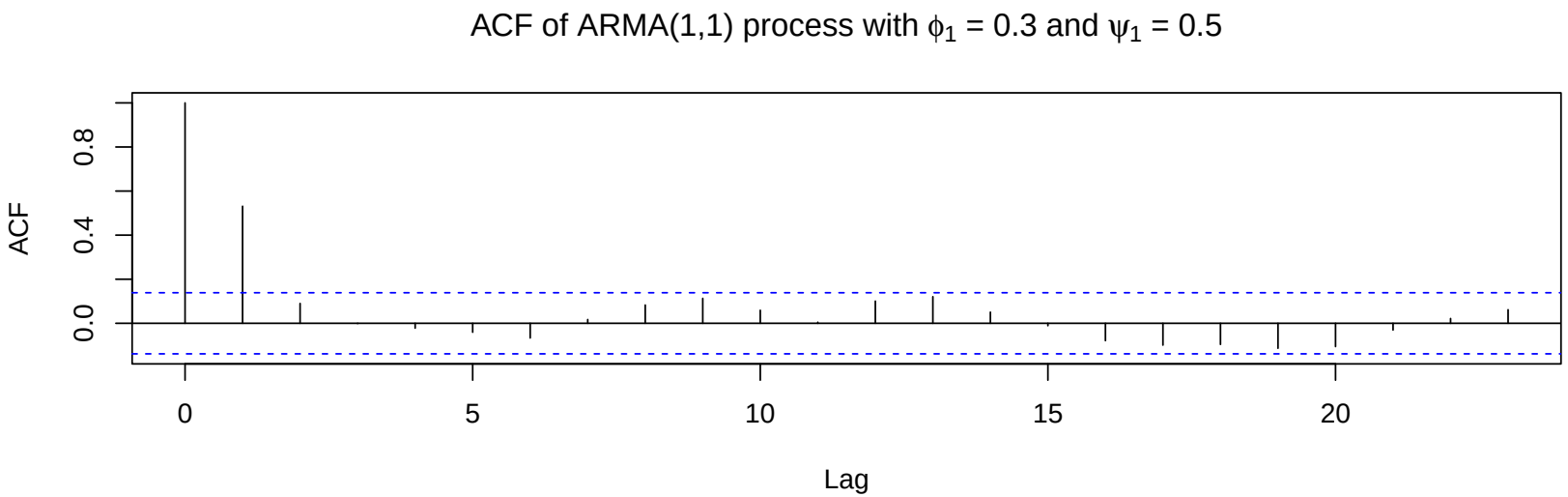
What do we learn about the effect of past shocks in an MA(q) process from the ACFs and PACFs?

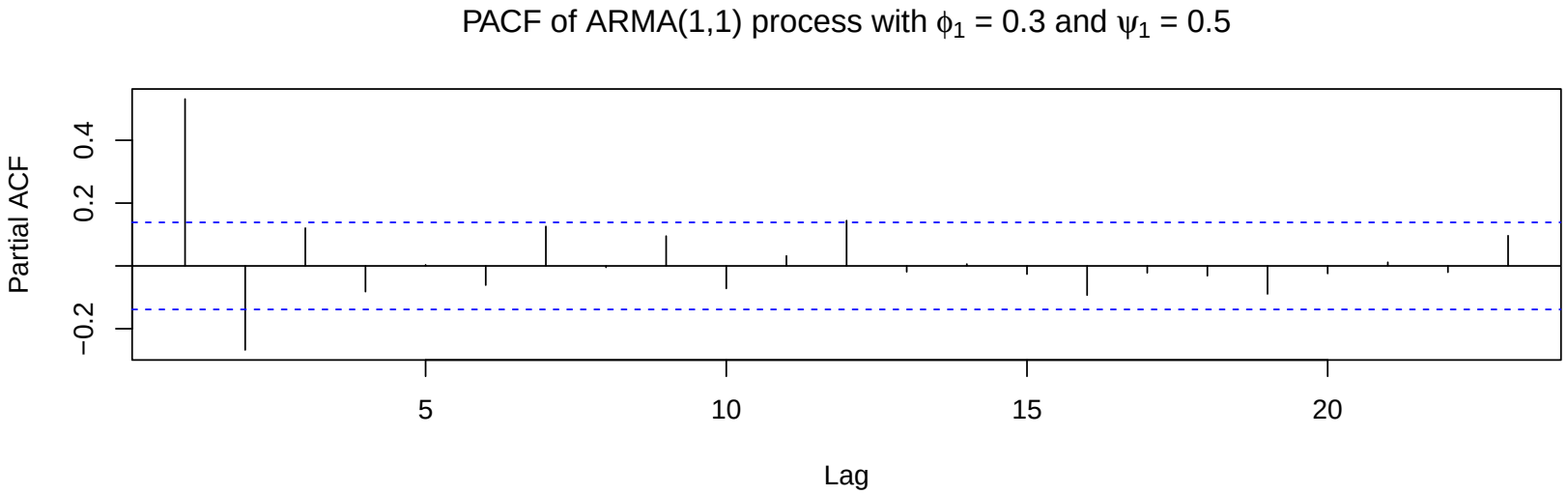
How can we identify an AR versus an MA process from the ACF and PACF.

ARMA Processes



ARMA Processes





Seasonality

- any cyclical fluctuation in a time series that recurs or repeats itself at the same phase of the cycle
- y_t is an additive or multiplicative function of y_{t-c} for some fixed cycle c (e.g. $c = 12$ for months)
- additive seasonality: corresponding months in different years share a level component
- multiplicative seasonality: corresponding months in different years related by a factor change

Practical Rules of ACF and PACF Patterns

To give you a sense, here some general recommendations to detect possible patterns of ACF and PACF for ARMA models:

- AR(p)
  - ACF: Tails off.
  - PACF: Cuts off after lag p; PACF(p) $= \phi_p$
- MA(q)
  - ACF: Cuts off after lag q.
  - PACF: Tails off
- ARMA(p, q)
  - ACF: Tails off
  - PACF: Tails off

Because it is difficult identify complex dynamic processes. We will rely on choosing a set of candidate models and test them statistically.

How to identify dynamic processes

1. `plot()` the TS, you can use `decompose()` or `stl()` after transforming the TS into a `ts()` class object.
2. De-trend and remove seasonality. I recommend to use `lm()` and extract the esitmated residuals for next steps.
3. Look at the correlogram of the de-trended data with `acf()` and `pacf()` to get an idea of potential dynamic candidates.
   3.1. If the `acf()` shows a non-decreasing behavior, use `PP.test()` and `adf.test()` to test for unit root.
4. Once you have some model candidates, estimate them using `arima()`.
5. Choose the final model that provides best fit **and** returns white noise evaluating.

Estimate time series using `arima()`

```
arima(x,
      order = c(0, 0, 0), # (AR-order, integrate-order, MA-
      seasonal = list(order = c(0, 0, 0), # seasonality
                      period = NA), # what period?
      include.mean = TRUE,
      xreg = NULL)
```

Post-estimation statistics

- After estimating candidate models, display the output and assess the fit.
  - Pay attention to AIC or BIC statistics; smaller values indicate better fit.
- Select the models that best fit the data and perform the Q-test statistic to check if residuals are white noise.
  - Use the function `Box.test()` in R.
- The most parsimonious model yielding white noise residuals is likely the correct dynamic specification.

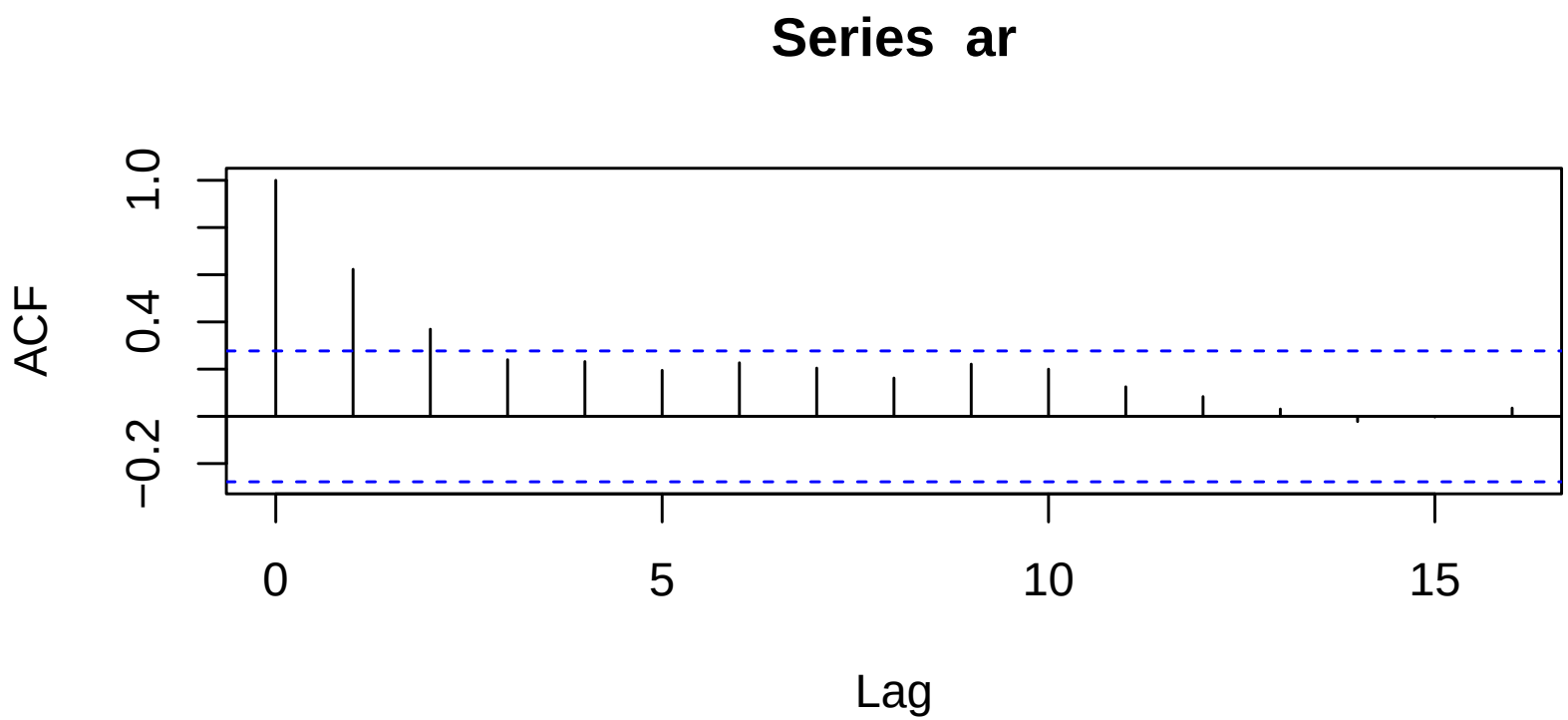
Use `arima.sim()` to Verifty Our Hypothesis

We can use `arima.sim()` to simulate time series and verify our guesses. For example, what are the differences in ACF and PACF between AR process with and without deterministic trend?

```
set.seed(98105)
ar.trend <- arima.sim(
  list(order = c(1, 0, 0), # the functional form of TS
       ar = 2/3,     # the coefficient for y_{t-1}
       ma = NULL,    # the coefficient for \epsilon_{t-1}
       beta = 4/5,   # slope
       alpha = 10), # intercept
  n = 50             # length of time series
)
ar <- arima.sim(list(order = c(1, 0, 0), ar = 2/3, ma = NUL
```

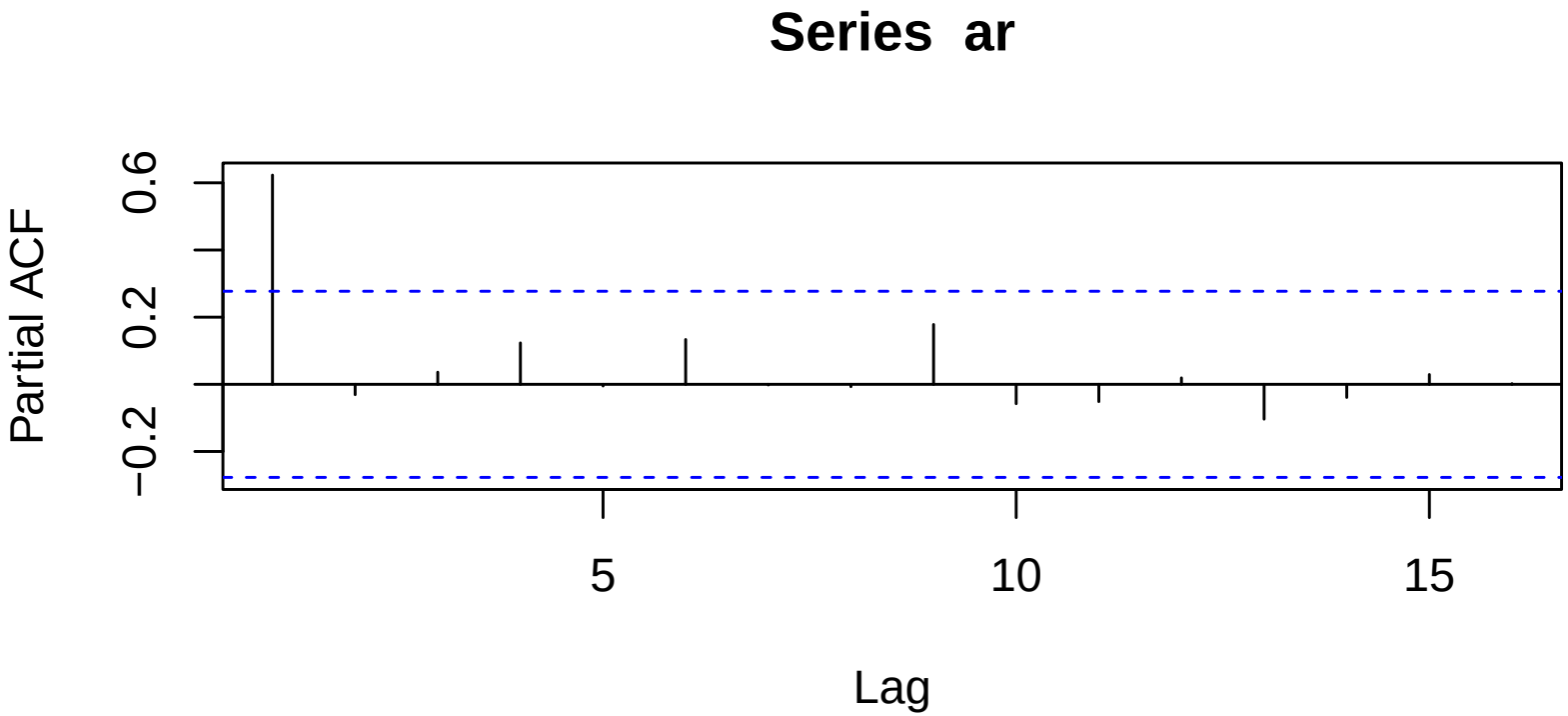
Use `arima.sim()`

As we discussed before, trend will induce an upward bias in ACF plot. That's why we want to de-trend a time series in the first place.



Use `arima.sim()`

Trend can induce slight bias for PACF at $t = 1$, but it does not affect PACF plot very much.



Use `arima.sim()`

- Note that although we know that true ϕ_1 is 0.67, the PACF correlogram of AR(1) with deterministic trend model shows that ϕ_1 is higher than 0.67, even above 0.7. This is because the trend is affecting the autocorrelation throughout the time.
- Therefore, we will need to de-trend the model and than obtain the correlogram.

Chris's custom function

- Alternatively to `arima.sim()`, you want to simulate more complex dynamic forms, use Chris's simulation code.
- In the .zip file of this lab, look at the folder source.

Coding Exercise: Identifying Unknown Time Series Processes

1. Identify the *deterministic trend* in the time series.
2. Identify the *seasonality* in the time series.
3. Remove the deterministic trend and seasonality from the time series.
4. Plot the pattern of ACF and PACF. What functional form do you think is reasonable in this time series?
5. Estimate model candidates with `arima()`.
6. Select best fitted model with white noise.
7. (optional) `arima.sim()` to verify your guess.

Summary for Time Series Diagnostics

In conclusion, to inspect any unknown time series data, we will need to utilize all available resources, including:

- Our generic knowledge about the components of our data (trend, seasonality, lag, etc)
- Applying different tools to test whether the knowledge is actually right
- Deciding whether our model with certain components that we assume are in the data fits based on statistics such as AIC (Alkarine Information Criteria)
- This whole process is called **Box-Jenkins Method.** Next week we will continue the topic of time series diagnostics.