IN CLASS WORK

Zachary Lazerick

16 March 2023

PROBLEM 1: AR(1) Models

(a) Simulate a time series of length 100 from an AR(1) model with α equal to 0.95. Then estimate the parameter and order of the model using the `ar()` function. Comment on each estimate.

```
set.seed(2)
x <- w <- rnorm(100); alpha <- 0.95
for (i in 2:length(x)) {
  x[i] <- alpha*x[i-1] + w[i]
}

x.ar <- ar(x)

## Parameter; Order; Estimates
x.ar$ar; x.ar$order
```

```
## [1] 0.8872249
```

```
## [1] 1
```

R was able to deduce the orer of the model correctly, given what we used to construct it. However, it underestimated the α value.

(b) Simulate a time series of length 100 from an AR(1) model with α equal to .95. Then estimate the parameter and order of the model using the `ar()` function.

```
set.seed(2)
x <- w <- rnorm(100); alpha <- 1.05
for (i in 2:length(x)) {
  x[i] <- alpha*x[i-1] + w[i]
}

x2.ar <- ar(x)
```

```r
## Parameter; Order; Estimates
x2.ar$ar; x2.ar$order
```

```
## [1] 0.9466802
```

```
## [1] 1
```

Again, R was able to correctly identify the order of the model, while underestimating the value of α used.

(c) Compare the two results. What issues arise?

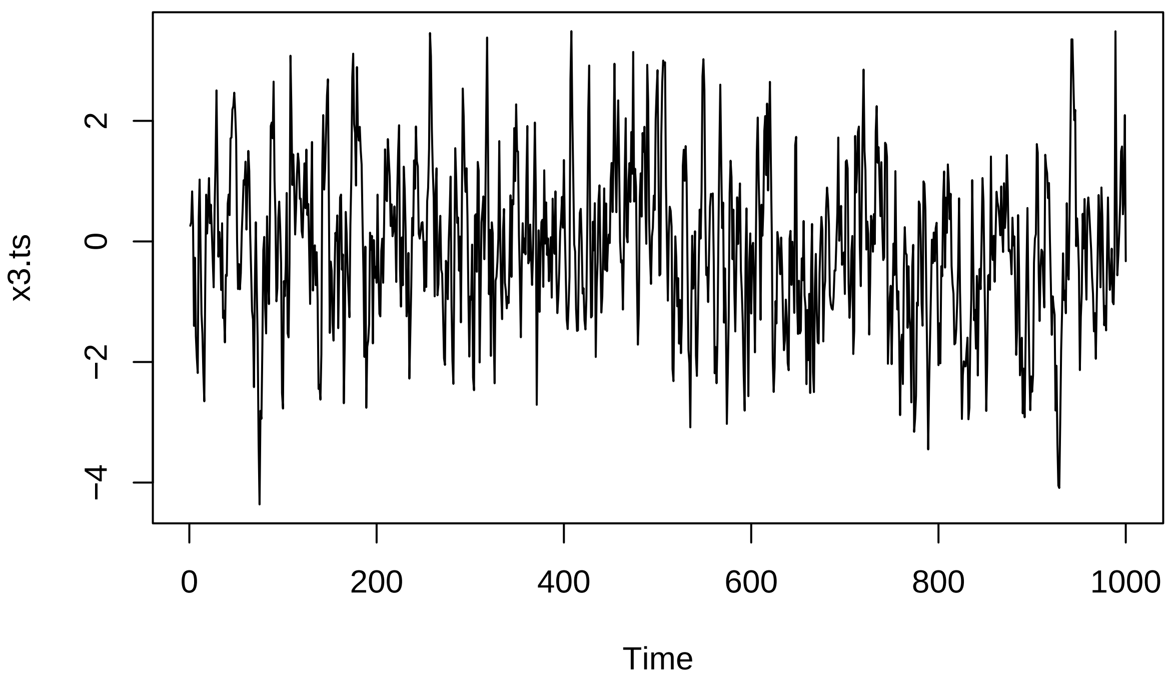
The results are similar. The one stand out issue is that R underestimates the coefficient of the model α.

PROBLEM 2: AR(2) Models

a) Simulate a time series of length 1000 for the following model, placing the simulated data in a vector `x`. Plot the time series.

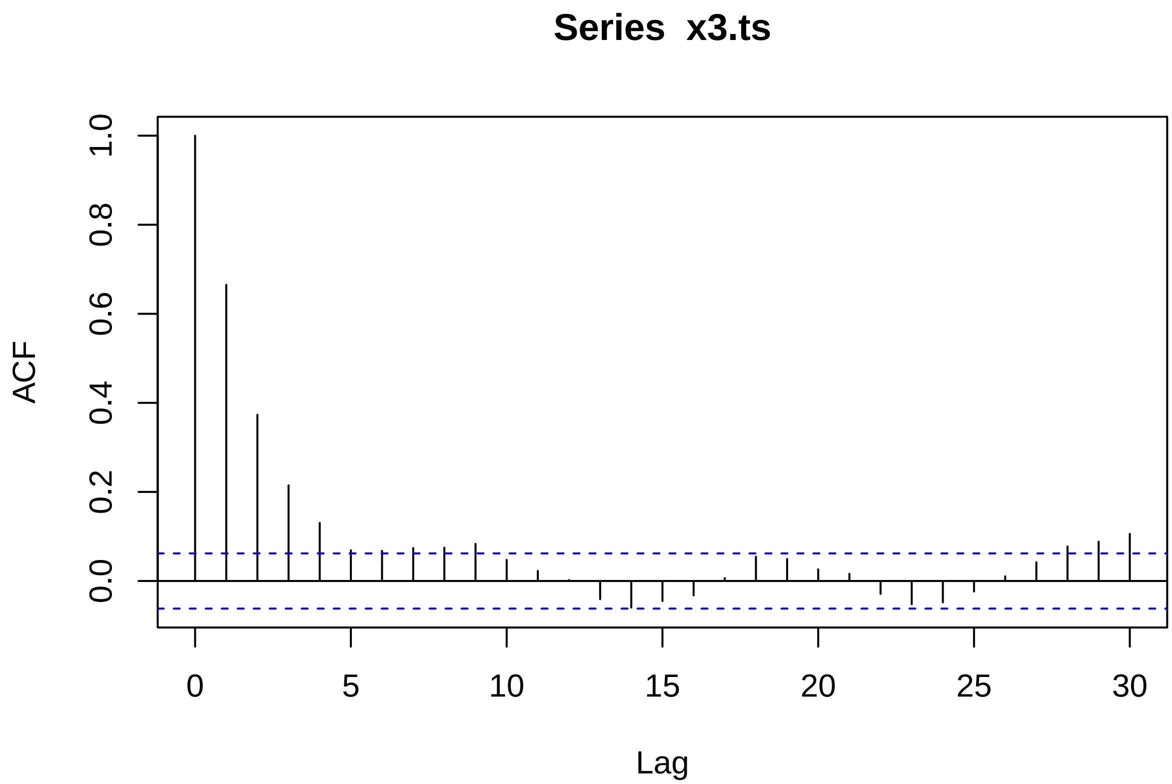
$$x_t = \frac{5}{6}x_{t-1} - \frac{1}{6}x_{t-2} + w_t$$

```
set.seed(1119)
x <- w <- rnorm(1000)
for (i in 3:length(x)) {
  x[i] <- (5/6)*x[i-1] - (1/6)*x[i-2] + w[i]
}

x3.ts <- ts(x)
plot(x3.ts)
```

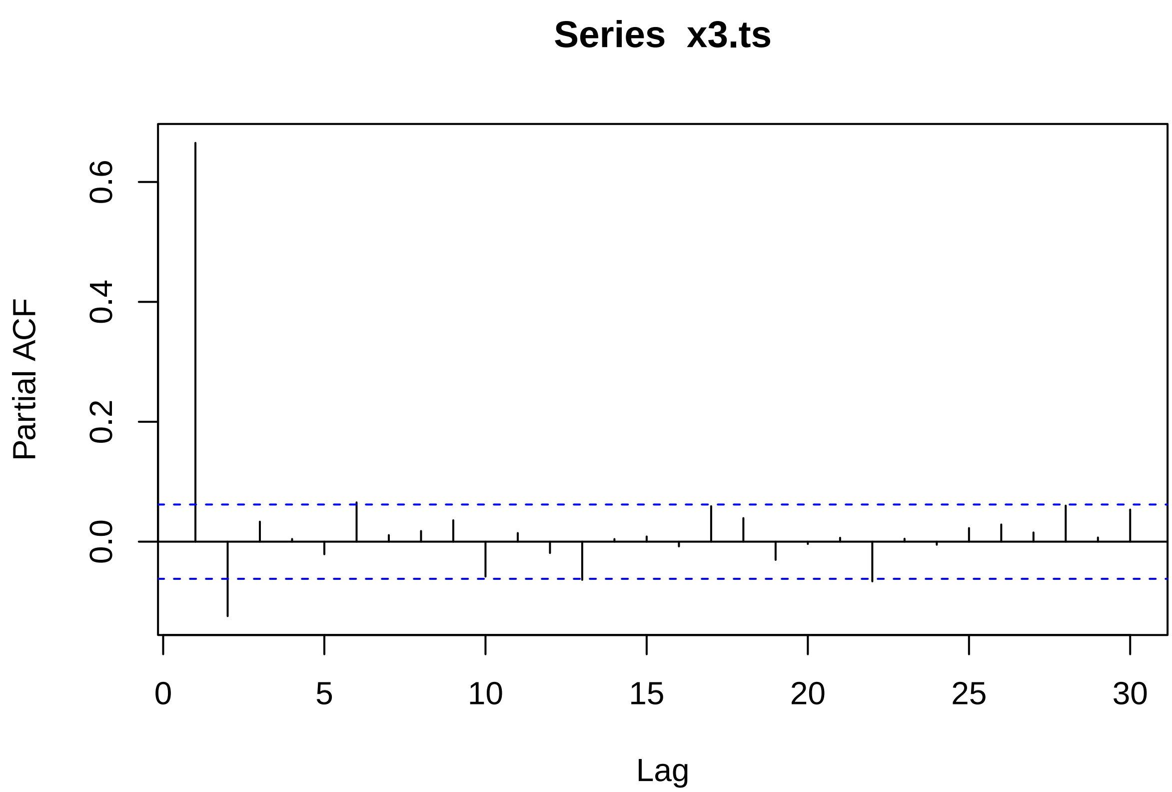


(b) Plot the correlogram and partial correlogram for the simulated data. Comment on the plots.

```
acf(x3.ts)
```




```
pacf(x3.ts)
```



The acf plot shows that we have significant auto correlation up until lag 5. Other later values that appear to just cross the threshold are likely false positive results. On average, these should be about 5% of all lags.

The pacf plot shows that we have significant auto correlation up until lag 2. Other values are all below the threshold. Thus, an AR(2) process would be suitable for this model, which fits the construction we used.

c) Fit an AR model to the data in `x` giving the parameter estimates and order of the fitted AR process.

```
x3.ar <- ar(x)

x3.ar$ar; x3.ar$order
```

```
## [1]  0.7477709 -0.1241524
```

```
## [1] 2
```

d) Construct 95% confidence intervals for the parameter estimates of the fitted model. Do the model parameters fall within the confidence intervals? Explain your results.

```
x3.ar$ar[1] + c(-1.96, 1.96) * sqrt(x3.ar$asy.var.coef[1, 1])
```

```
## [1] 0.6861773 0.8093645
```

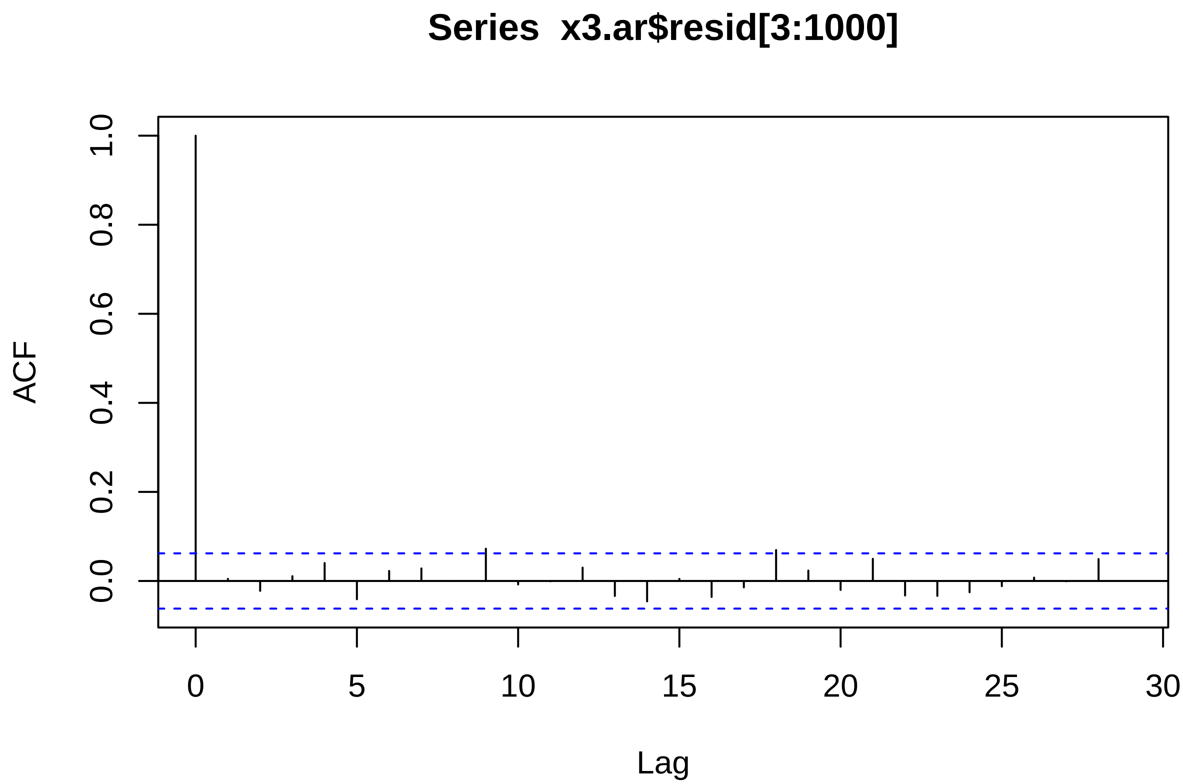
```
x3.ar$ar[2] + c(-1.96, 1.96) * sqrt(x3.ar$asy.var.coef[2, 2])
```

```
## [1] -0.18574593 -0.06255879
```

1 of the model parameters fall within the confidence intervals. The first parameter, 5/6 lies barely outside the first confidence interval. The second parameter -1/6 in contained in the second confidence interval.

e) Plot the correlogram of the residuals of the fitted model, and comment on the plot.

```
acf(x3.ar$resid[3:1000])
```



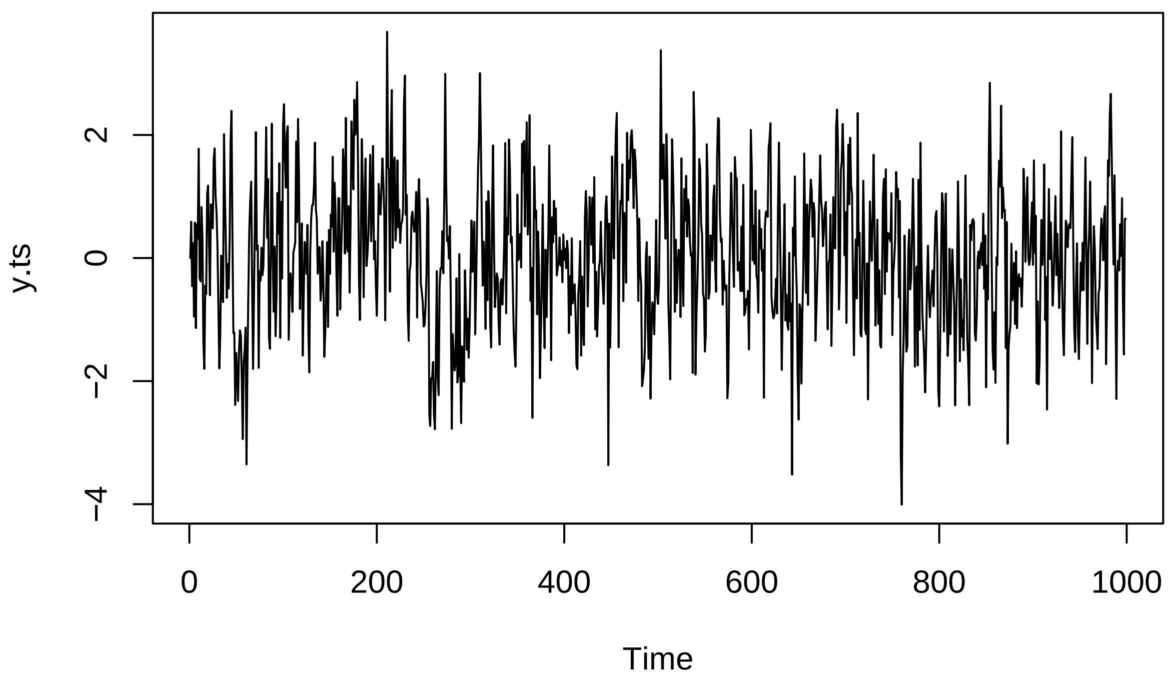
The correlogram shows that the residuals have no autocorrelation. Thus, we have uncorrelated residuals. Pog!

PROBLEM 3: AR(2) Models and Stationarity

a) Simulate a series of 1000 values for $\{x_t\}$, placing the simulated data in `x`, and use these simulated values to produce a series of 999 values for $\{y_t\} = \{\nabla x_t\}$, placing this series in the vector `y`. Plot the time series y_t.

$$x_t = \frac{3}{2}x_{t-1} - \frac{1}{2}x_{t-2} + w_t$$

```
set.seed(1137)
x <- w <- rnorm(1000)
for (i in 3:length(x)) {
  x[i] <- (3/2)*x[i-1] - (1/2)*x[i-2] + w[i]
}

y <- rep(0, 999)
for (j in 2:length(y)) {
  y[j] <- x[j] - x[j-1]
}

y.ts <- ts(y); plot(y.ts)
```



b) Fit an AR model to `y`. Give the fitted model parameter estimates and a 95% confidence interval for the underlying model parameters based on these estimates.

```
y.ar <- ar(y)
```

```r
## Parameter/Order Estimates
y.ar$ar; y.ar$order
```

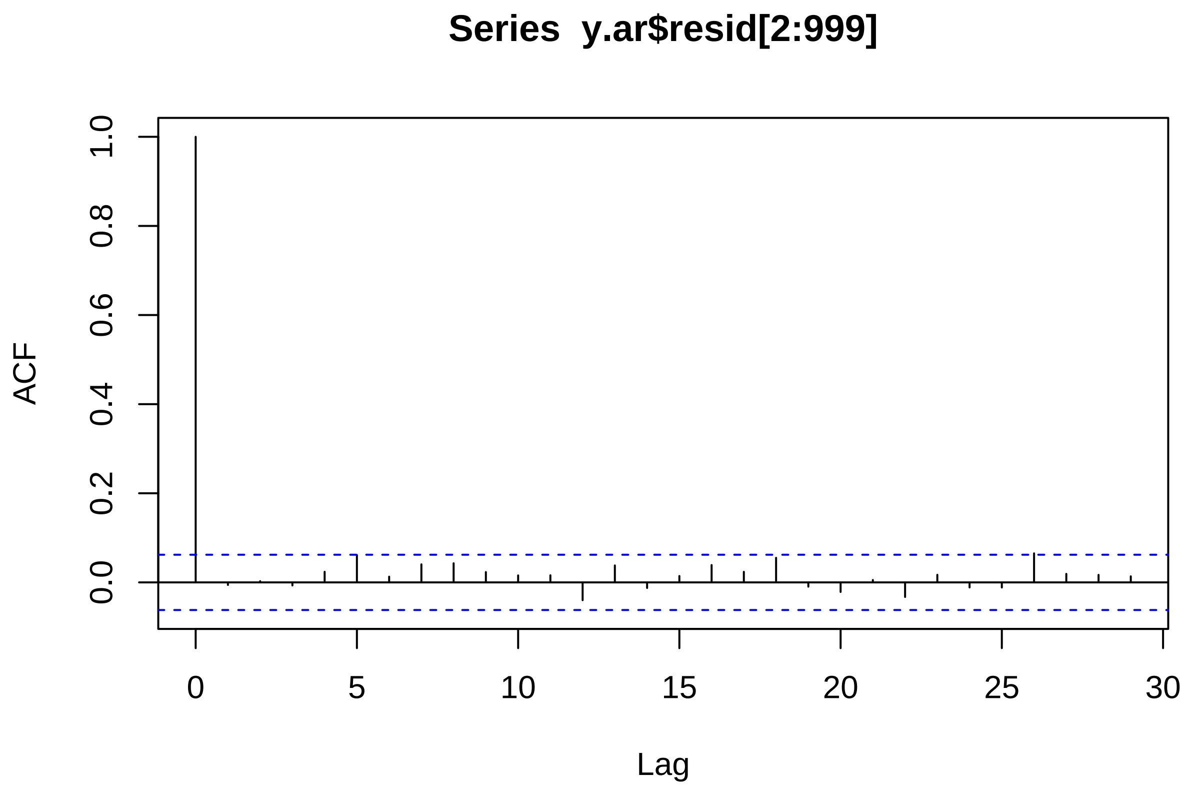
```
## [1] 0.4649324

## [1] 1
```

```r
## Confidence Intervals
y.ar$ar + c(-1.96, 1.96) * sqrt(y.ar$asy.var.coef[1, 1])
```

```
## [1] 0.4099756 0.5198892
```

c) Plot the correlogram of the residuals of the fitted model and comment.

```r
acf(y.ar$resid[2:999])
```



The correlogram shows that the residuals of ‘y’ are not auto correlated for any lag. Thus, our residuals are not correlated. POGGO!