Model Specification:

Sample Extended Autocorrelation Function

Week VIII: Video 25

STAT 485/685, Fall 2020, SFU

Sonja Isberg

Video 25 Learning Objectives

By the end of this video, we should be able to:

- Explain why the sample ACF and sample PACF cannot always help us to determine the model that best fits the dataset
- Explain why the extended autocorrelation function can help us determine the orders of an ARMA process
- Use a table of the sample extended autocorrelation function to determine the orders of an ARMA process
- Use a combination of the sample ACF, sample PACF and sample EACF together for model specification

Extended Autocorrelation Function: Motivation

What do we know so far about the usefulness of ACF and PACF in terms of model specification?[1]

	AR(p)	**MA(q)**	**ARMA(p,q)**
ACF	Tails off	Cuts off after lag q	Tails off
PACF	Cuts off after lag p	Tails off	Tails off

[1] Table from pg. 116 of Cryer & Chan (2008).

Extended Autocorrelation Function: Motivation

What do we know so far about the usefulness of ACF and PACF in terms of model specification?[1]

	AR(p)	**MA(q)**	**ARMA(p,q)**
ACF	Tails off	Cuts off after lag q	Tails off
PACF	Cuts off after lag p	Tails off	Tails off

For an ARMA model with $p > 0$ and $q > 0$, its theoretical ACF and PACF do not "cut off" after any lag k – they continue on for infinitely many lags.

So how can we use visual information to identify an ARMA model? One method involves using the *extended autocorrelation function*.

The EACF method was developed by Tsay & Tiao (1984).

[1] Table from pg. 116 of Cryer & Chan (2008).

Extended Autocorrelation Function: Idea

If the AR part of an ARMA model is known, then "filtering out" the autoregression from the observed time series results in a pure MA process.

This pure MA process will then have an ACF that cuts off after a lag equal to its order.

The autoregression can be "filtered out" by first estimating the AR coefficients using a sequence of q regressions (see pg. 116 of the textbook for more info).

However, since we know neither p nor q, we have to repeat this process iteratively, over a grid of possible AR orders and MA orders.

Extended Autocorrelation Function: Definition

For AR order $k = 1, 2, 3, \ldots$ and MA order $j = 1, 2, 3, \ldots$:

1. Let
$$W_{t,k,j} = Y_t - \tilde{\phi}_1 Y_{t-1} - \cdots - \tilde{\phi}_k Y_{t-k}$$
be the residuals from "filtering out" the AR(k) part of the model. The AR coefficients $\tilde{\phi}_1, \ldots, \tilde{\phi}_k$ are estimated using a sequence of regressions (as on pg. 116).

2. The sample autocorrelations of $\{W_{t,k,j}\}$ are the **extended sample autocorrelations**.

3. If $k = p$ (the true AR order) and $j \geq q$ (the true MA order):
   - $\{W_{t,k,j}\}$ is approximately an MA(q) model
   - So, the theoretical autocorrelations of $\{W_{t,k,j}\}$ at lags $> q$ are equal to zero

4. If $k > p$, an overfitting problem occurs, which shifts our MA order a bit. Similar rules apply to $\{W_{t,k,j}\}$, but they occur at later lags.

Extended Autocorrelation Function: Visualization

Tsay & Tiao (1984) recommend summarizing the information in a single table as follows:

- Label the rows as consecutive values k of the AR order
- Label the columns as consecutive values j of the MA order
- The $(k,j)^{\text{th}}$ element of the table corresponds to the lag-$(j+1)$ sample autocorrelation of $\{W_{t,k,j}\}$
- Rather than writing out the full value of the sample autocorrelation, simply display the symbol "$\times$" if the value is significantly different from zero, and "0" otherwise
  - Since the sample autocorrelation is asymptotically $\mathcal{N}(0, 1/\sqrt{n-k-j})$, a value is significant if it is greater than $1.96/\sqrt{n-k-j}$ in absolute value
- In this table, an ARMA(p,q) process should *theoretically* give a triangle of zeros, with the upper left-hand corner corresponding to the orders of the process

Extended Autocorrelation Function: Visualization (cont'd)

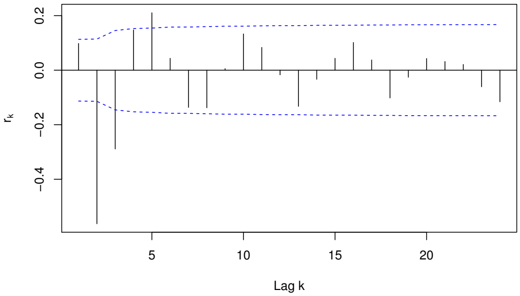
An ARMA(2,1) process should *theoretically* give the EACF table below:

AR/MA	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
0	×	×	×	×	×	×	×	×	×	×	×	×	×	×
1	×	×	×	×	×	×	×	×	×	×	×	×	×	×
2	×	0	0	0	0	0	0	0	0	0	0	0	0	0
3	×	×	0	0	0	0	0	0	0	0	0	0	0	0
4	×	×	×	0	0	0	0	0	0	0	0	0	0	0
5	×	×	×	×	0	0	0	0	0	0	0	0	0	0
6	×	×	×	×	×	0	0	0	0	0	0	0	0	0
7	×	×	×	×	×	×	0	0	0	0	0	0	0	0

The upper left-hand corner of the "triangle of zeros" corresponds to the orders $p = 2$ and $q = 1$.

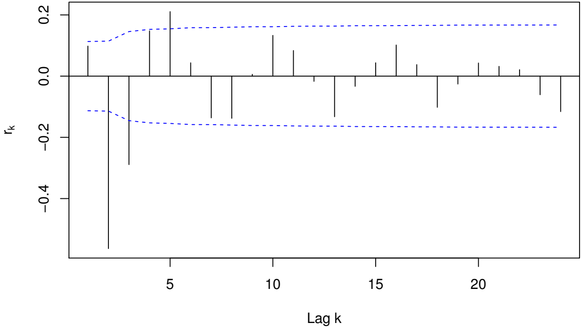
Note: The sample EACF will usually look a bit different than it does in theory.

Example 1: ARMA(2,2) Process




```
> e.vec <- rnorm(n=300, mean=0, sd=sqrt(0.2))
> y.vec <- rep(NA, times=300)
> y.vec[1] <- 0; y.vec[2] <- 0
> for (t in 3:300)
  {
     y.vec[t] <- 0.4*y.vec[t-1] - 0.5*y.vec[t-2] + e.vec[t]
                 - 0.6*e.vec[t-1] - 0.3*e.vec[t-2]
  }
> y.vec.ts <- ts(data=y.vec)
> acf(y.vec.ts, ci.type='ma', ylab=expression(r[k]))
```

Example 1: ARMA(2,2) Process (cont'd)

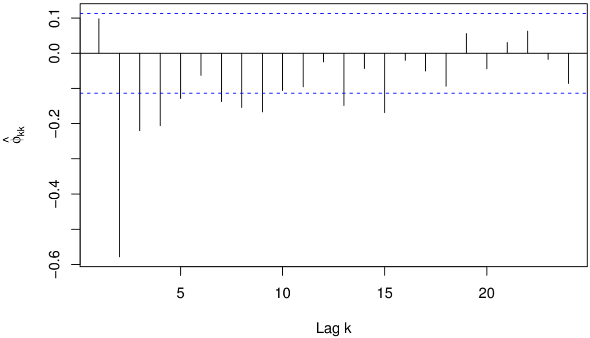


Sample ACF plot:

At each lag k, we are testing for whether or not the r_k cuts off after this lag (as it would if this were an MA($k-1$) process).

This plot might potentially suggest an MA(3) or MA(5) process. However, there is some sinusoidal behaviour in the sample ACF values, and they don't appear to really "cut off" after any lag k.

Example 1: ARMA(2,2) Process (cont'd)




```
> pacf(y.vec.ts, ylab=expression(hat(phi)[kk]))
```

The sample PACF is also not entirely clear here.

Values appear to decay exponentially, without really cutting off at any specific lag.

Example 1: ARMA(2,2) Process (cont'd)

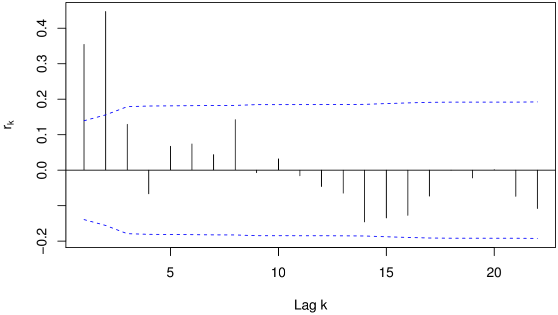
```
> eacf(y.vec.ts)

AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 o x x x x o x x o x o  o  x  o
1 x x x x x o o x o o o  o  x  o
2 x x o o x o o o o o o  o  x  o
3 x x o o x o o o o o o  o  o  o
4 x x o o x o o o o o o  o  o  o
5 x o x x x x o o o o o  o  o  o
6 x x x x o o o o o o o  o  o  o
7 x o o x o x o o o o o  o  o  o
```

The sample EACF table looks quite different than what we'd expect in theory!

There is some evidence for an ARMA(0,0) (i.e., white noise) model. However, stronger evidence points towards either ARMA(2,2) or ARMA(0,5) (i.e., MA(5)).

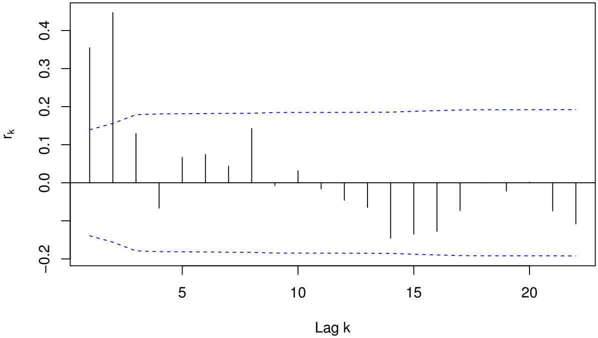
The answer is not entirely clear.

Example 2: MA(2) Process




```
> e.vec <- rnorm(n=200, mean=0, sd=1)
> y.vec <- rep(NA, times=200)
> for (t in 3:200)
  {
     y.vec[t] <- e.vec[t] + 0.2*e.vec[t-1] + 0.9*e.vec[t-2]
  }
> y.vec.ts <- ts(data=y.vec[-(1:2)])
> acf(y.vec.ts, ci.type='ma', ylab=expression(r[k]))
```

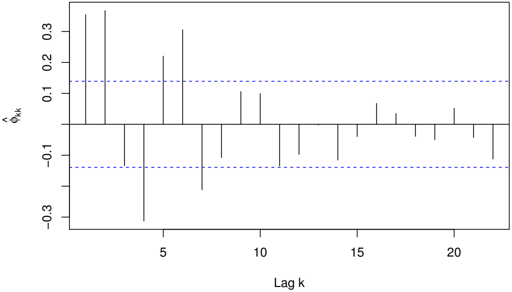
Example 2: MA(2) Process (cont'd)



The sample ACF plot appears to support the hypothesis that this data comes from an MA(2) process.

There does appear to be some sinusoidal behaviour in the plot, but we are not sure if this is due to some underlying AR behaviour, or if it is just due to sampling error.

Example 2: MA(2) Process (cont'd)




```
> pacf(y.vec.ts, ylab=expression(hat(phi)[kk]))
```

The sample PACF plot doesn't appear to give a clearer picture than the sample ACF plot, for this dataset.

One could potentially argue that this might look like an AR(7) process. However, due to the *principle of parsimony*, we would be more inclined to accept the much simpler MA(2) model.

Example 2: MA(2) Process (cont'd)

```
> eacf(y.vec.ts)
```

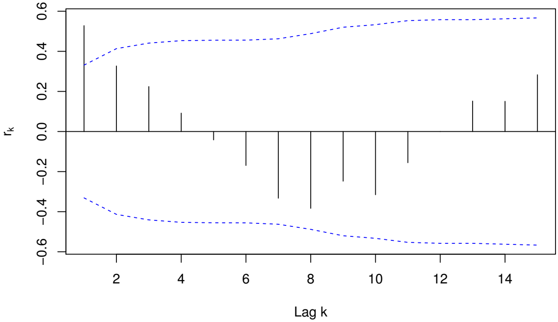
```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x x o o o o o o o o o  o  o  o
1 x x x o o o o o o o o  o  o  o
2 x x x o o o o o o o o  o  o  o
3 x x o x x o o x o o o  o  o  o
4 x x x o o x x o o o o  o  o  o
5 x x x o x x x o o o o  o  o  o
6 x x x o o x x x o o o  o  o  o
7 x x x o o o o x x o o  o  o  o
```

The "upper left-hand corner" is a lot more clear in this example – it shows up under $p = 0$ and $q = 2$, which corresponds to the MA(2) model.

However, we still have to be careful when making conclusions, since there is always a possibility of sampling error.

Example 3: Color Dataset

The `color` dataset in the `TSA` package gives the color property from 35 consecutive batches in an industrial process. Its sample ACF is:

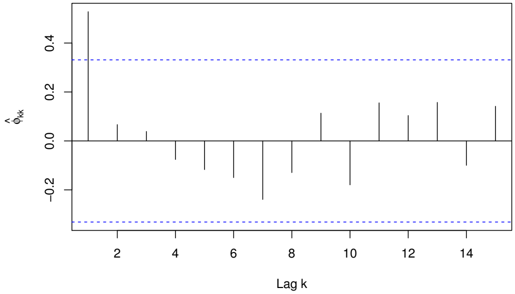



```
> data(color)
> acf(color, ci.type='ma', ylab=expression(r[k]))
```

The sample ACF plot suggests that an MA(1) model may potentially be appropriate here.

Example 3: Color Dataset (cont'd)

Let's create the sample PACF plot:




```
> pacf(color, ylab=expression(hat(phi)[kk]))
```

This plot suggests that perhaps an AR(1) model might be appropriate instead!

Example 3: Color Dataset (cont'd)

```
> eacf(color, ar.max=7, ma.max=9)
```

```
AR/MA
  0 1 2 3 4 5 6 7 8 9
0 x o o o o o o o o o
1 o o o o o o o o o o
2 o o o o o o o o o o
3 x o o o o o o o o o
4 o o o o o o o o o o
5 x o o o o o o o o o
6 x o o o o o o o o o
7 x o o o o o o o o o
```

It seems the sample EACF table doesn't help us too much in deciding between MA(1) and AR(1) for this dataset – they both form "upper left-hand corners"!

Final Comments

That's all for now!

In this video, we've learned about the extended autocorrelation function, and about some of its important properties for ARMA models of different orders.

We also learned how to read a table showing the sample extended autocorrelation function, and how it can help us identify the orders p and q of an ARMA model.

Finally, we looked at several examples of using sample ACF, sample PACF and sample EACF together to choose a model for a given time series dataset.

Next Week in STAT 485/685: Some more practice with model specification, and review of Ch. 4-6.

Thank you!

References:

[1] Cryer, J. D., & Chan, K. S. (2008). *Time series analysis: with applications in R.* Springer Science and Business Media.

[2] Chan, K. S., & Ripley, B. (2020). TSA: Time Series Analysis. R package version 1.2.1.

[3] Tsay, R. S., & Tiao, G. C. (1984). Consistent estimates of autoregressive parameters and extended sample autocorrelation function for stationary and nonstationary ARMA models. *Journal of the American Statistical Association, 79*(385), 84-96.