# Class 3: Integrated models and ARIMA

Andrew Parnell
andrew.parnell@mu.ie

PRESS RECORD https://andrewcparnell.github.io/TSDA/

# Learning outcomes

- Understand how differencing works to help make data stationary
- Know the basics of the ARIMA(p, d, q) framework
- Understand how to fit an ARIMA(p, d, q) model in a realistic setting

# Reminder: stationarity

- A time series is said to be weakly stationary if:
  - The mean is stable
  - The variance is stable
  - The autocorrelation doesn't depend on where you are in the series

# Reminder: ARMA models

- Combine the autoregressive and the moving average framework into one
- The general equation for an ARMA(p, q) model is:

$$y_t = \alpha + \sum_{i=1}^{p} \beta_i y_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} + \epsilon_t$$

# Combining ARMA with the random walk to produce ARIMA

- There is one other time series model we have already met, that of the random walk:

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

  where $\epsilon_t \sim N(0, \sigma^2)$
- We could re-write this as:

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

  i.e. the *differences* are random normally-distributed noise

# Differencing

- Differencing is a great way of getting rid of a trend
- If $y_t \approx y_{t-1} + b$ then there will be an increasing linear slope in the time series
- Creating $y_t - y_{t-1}$ will remove it and all values will hover around the value $b$
- Even when the trend is non-linear differencing might help
- Differencing twice will remove a quadratic trend for the same reasons
- You can do even higher levels of differencing but this starts to cause problems
- The twice differenced series is:

$$(y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) = y_t - 2y_{t-1} + y_{t-2}$$

# Idea: combine differencing into the ARMA framework

- We can combine these ideas into the ARMA framework to produce an ARIMA model (the I stands for integrated, i.e. differenced)
- An ARIMA model isn't really stationary as the differences are actually removing part of the trend
- The ARIMA model is written as ARIMA(p,d,q) where $p$ and $q$ are as before and $d$ is the number of differences

# Example: the ARIMA(1,1,1) model

- If we want to fit an ARIMA(1,1,1) model we first let $z_t = y_t - y_{t-1}$ then fit the model:
$$z_t \sim N(\alpha + \beta z_{t-1} + \theta \epsilon_{t-1}, \sigma^2)$$
- This is equivalent to an ARMA model on the first differences
- Note that by default `forecast` does not include the term $\alpha$ in the model. You need to add `include.drift = TRUE`

# Fitting an ARIMA(1, 1, 0) model to the wheat data

- Recall that the ARMA(2,1) fit wasn't very good to the wheat data
- Instead try an ARIMA(1, 1, 0) model (i.e. AR(1) on the first differences)

```
wheat = read.csv('../../data/wheat.csv')
Arima(wheat$wheat, order = c(1, 1, 0),
      include.drift = TRUE)
```

```
## Series: wheat$wheat
## ARIMA(1,1,0) with drift
##
## Coefficients:
##           ar1     drift
##       -0.0728  529.4904
## s.e.   0.1503  401.5641
##
## sigma^2 = 9945763:  log likelihood = -491.7
## AIC=989.39   AICc=989.89   BIC=995.25
```

# General format: the ARIMA(p,d,q) model

- First take the $d$th difference of the series $y_t$, and call this $z_t$
- If you want to do this by hand in R you can use the `diff` function, e.g. `diff(y, differences = 2)`
- Then fit the model:

$$z_t \sim N\left(\alpha + \sum_{i=1}^{p} \beta_i z_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j}, \sigma^2\right)$$

# Choosing $p$, $d$ and $q$

- It's much harder to have an initial guess at all of $p$, $d$ and $q$ in one go
- We can usually guess at the number of differences $d$ from the time series and ACF plots. If there is a very high degree of autocorrelation it's usually a good idea to try a model with $d=1$ or 2
- I've never met a model where you needed to difference more than twice. Beware of over-differencing

# Revisiting the real-world example

# Steps in an ARIMA time series analysis

1. Plot the data and the ACF/PACF
2. Decide if the data look stationary or not. If not, perform a suitable transformation and return to 1. **If the data has a strong trend or there is a high degree of autocorrelation try 1 or 2 differences**
3. Guess at values of $p$, $d$, and $q$ for an ARIMA(p, d, q) model
4. Fit the model
5. Try a few models around it by increasing/decreasing $p$, $d$ and $q$ and checking the AIC (or others)
6. Check the residuals. If the residuals look strange (skewed or heavy tailed) perform a Box-Cox transformation and return to step 1
7. Forecast into the future

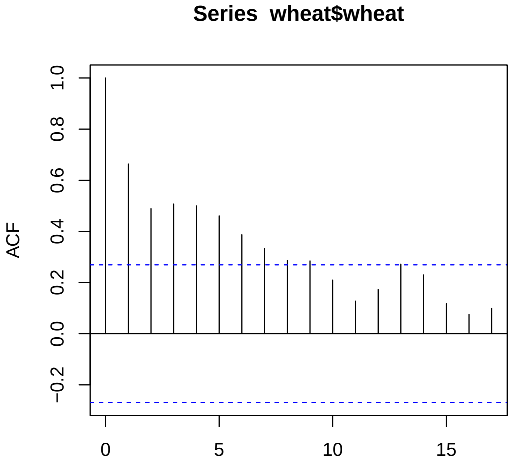
# A real example: wheat data

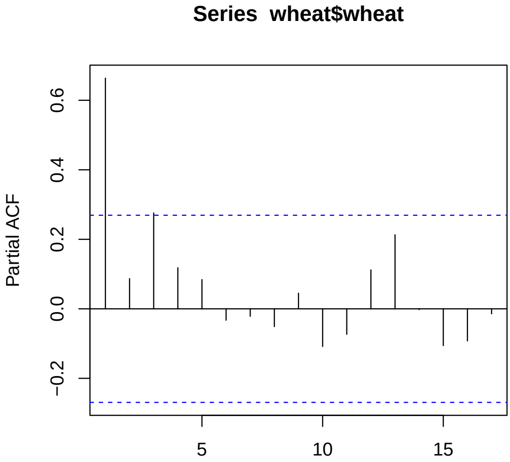
- Plot reminder

```r
wheat = read.csv('../../data/wheat.csv')
plot(wheat$year, wheat$wheat, type = 'l')
```

# ACF and PACF

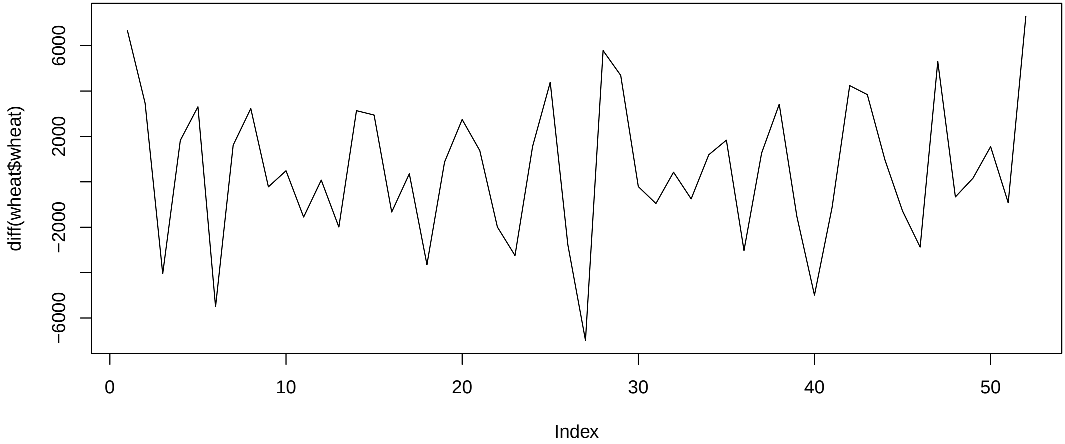
```
par(mfrow = c(1, 2))
acf(wheat$wheat)
pacf(wheat$wheat)
```



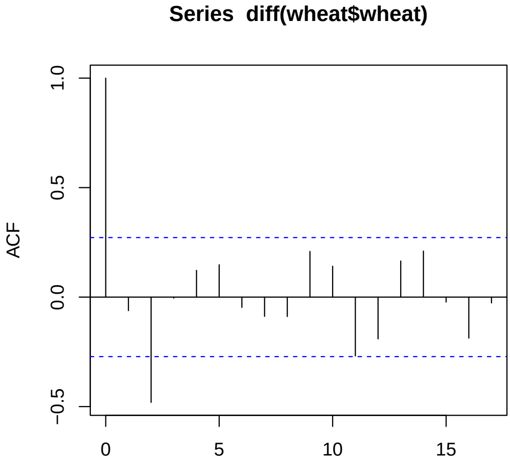
# Plot of first differences

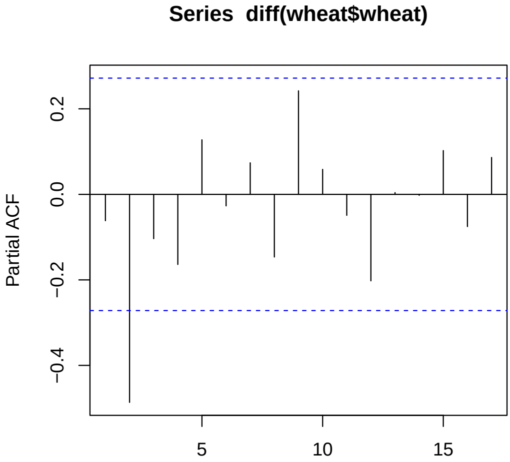
```
plot(diff(wheat$wheat), type = 'l')
```



# ACF/PACF of first differences

```r
par(mfrow = c(1, 2))
acf(diff(wheat$wheat))
pacf(diff(wheat$wheat))
```



# First model

```
Arima(wheat$wheat, order = c(0, 1, 0),
      include.drift = TRUE)
```

```
## Series: wheat$wheat
## ARIMA(0,1,0) with drift
##
## Coefficients:
##          drift
##       546.4265
## s.e.  429.8333
##
## sigma^2 = 9795708:  log likelihood = -491.81
## AIC=987.63   AICc=987.87   BIC=991.53
```

- This is just a random walk model. Can also get these from `forecast` with the function `naive`

# Next models

- Try ARIMA(1, 1, 1), ARIMA(1, 1, 0), ARIMA(0, 1, 1)

```
Arima(wheat$wheat, order = c(1, 1, 1),
      include.drift = TRUE)$aic
```

```
## [1] 979.1519
```

```
Arima(wheat$wheat, order = c(1, 1, 0),
      include.drift = TRUE)$aic
```

```
## [1] 989.3936
```

```
Arima(wheat$wheat, order = c(0, 1, 1),
      include.drift = TRUE)$aic
```

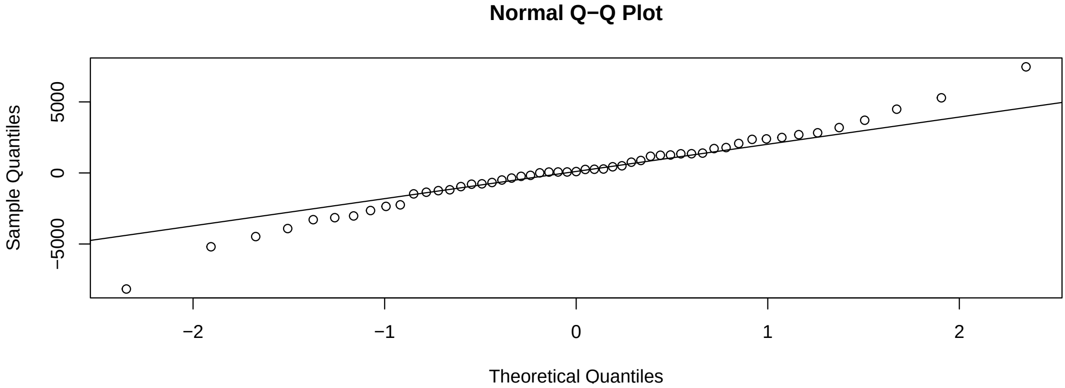
```
## [1] 981.2407
```

- Best one seems to be ARIMA(1, 1, 1). (though BIC suggests others)

# Check residuals

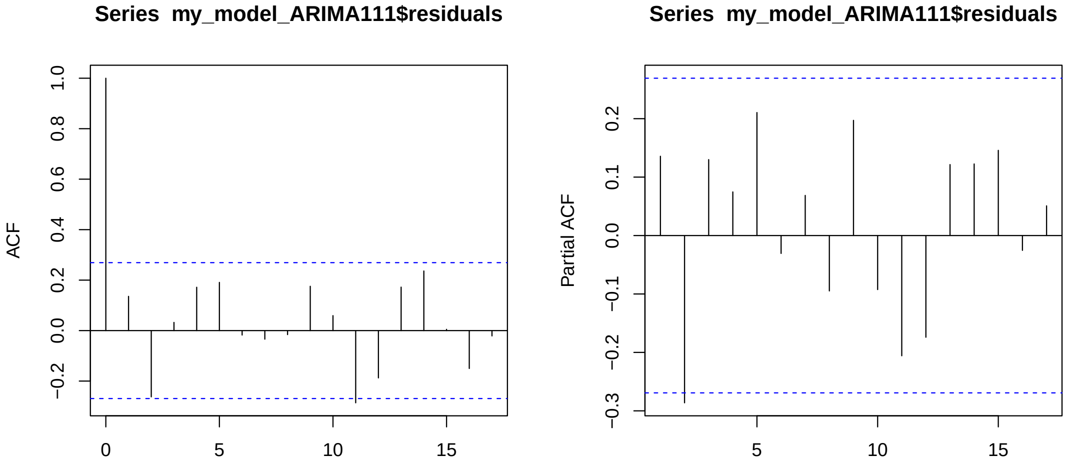
- Check the residuals of this model

```
my_model_ARIMA111 = Arima(wheat$wheat, order = c(1, 1, 1),
                          include.drift = TRUE)
qqnorm(my_model_ARIMA111$residuals)
qqline(my_model_ARIMA111$residuals)
```



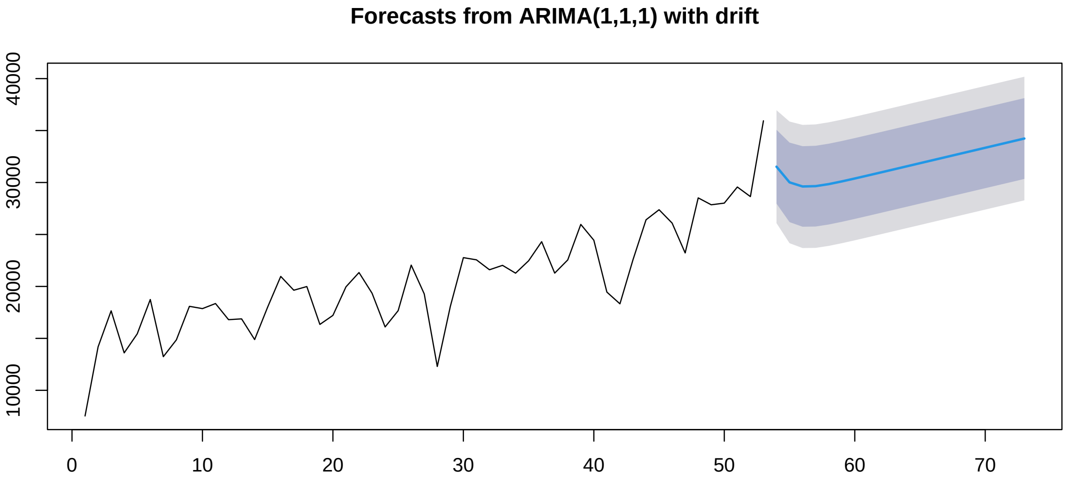
# Check residual ACF and PACF

```r
par(mfrow=c(1,2))
acf(my_model_ARIMA111$residuals)
pacf(my_model_ARIMA111$residuals)
```



# Forecast into the future

```
plot(forecast(my_model_ARIMA111, h = 20))
```



# Why do we need to the drift term?

- Without the drift term the forecast will stabilise at or near the first few values of the series
- The MA part of the model is obviously flat (as previously discussed) because there are no further errors to correct
- The AR part of the model reverts back to the estimated mean of the last data point because the $\beta$ parameter is less than 1 - it dampens out the future predictions and stops them from going crazy
- The drift keeps the values going up into the future
- `forecast` doesn't seem to like including the drift/mean when there are multiple differences and AR terms too (not sure why)

# Summary

- ARIMA models extend the ARMA framework to further add in differencing
- ARIMA models are no longer stationary as soon as $d > 0$
- A single difference will remove a linear trend, a second difference will remove a quadratic trend (but can also capture non-linear trends too)
- Can spot the need for differencing from the time series plot and the ACF
- Do not over-difference your data!