Data Science for Business Applications

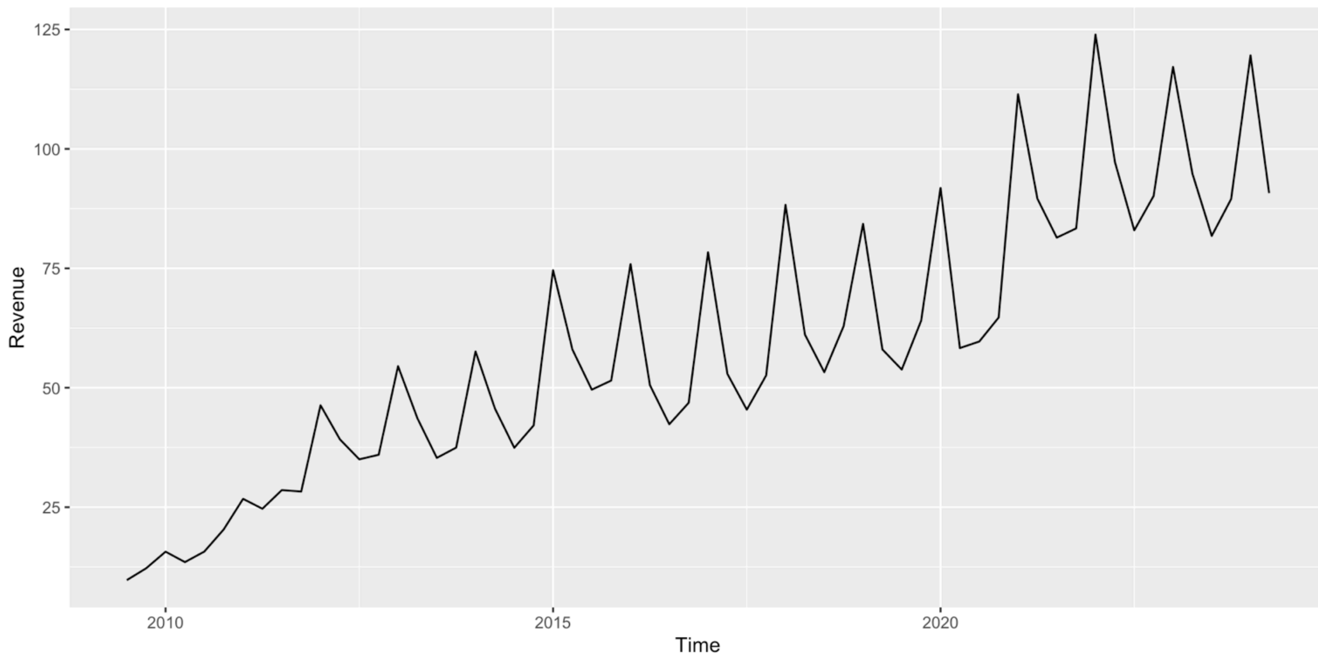
Class 05 - Time series

The University of Texas at Austin
McCombs School of Busine

Basic time series concepts

- Apple quarterly revenue (Billions of dollars)
- Goal: What is the pattern here, and how can we forecast future earnings?

```
library(tidyverse)
library(ggfortify)
ggplot(apple, aes(x=Time, y=Revenue)) +
  geom_line()
```



What are time series?

- Data where the cases represent time: data collected every day, month, year, etc.
- Time series are important for both explaining how variables change over time and forecasting the future
- Examples of time series data:
- *Google's closing daily stock price every day in 2020*
- *Inventory levels of each item at a retail store at the end of every week in 2020*
- *Number of new COVID cases in the US each day since the start of the pandemic*
- *Apple's quarterly revenue since 2009*

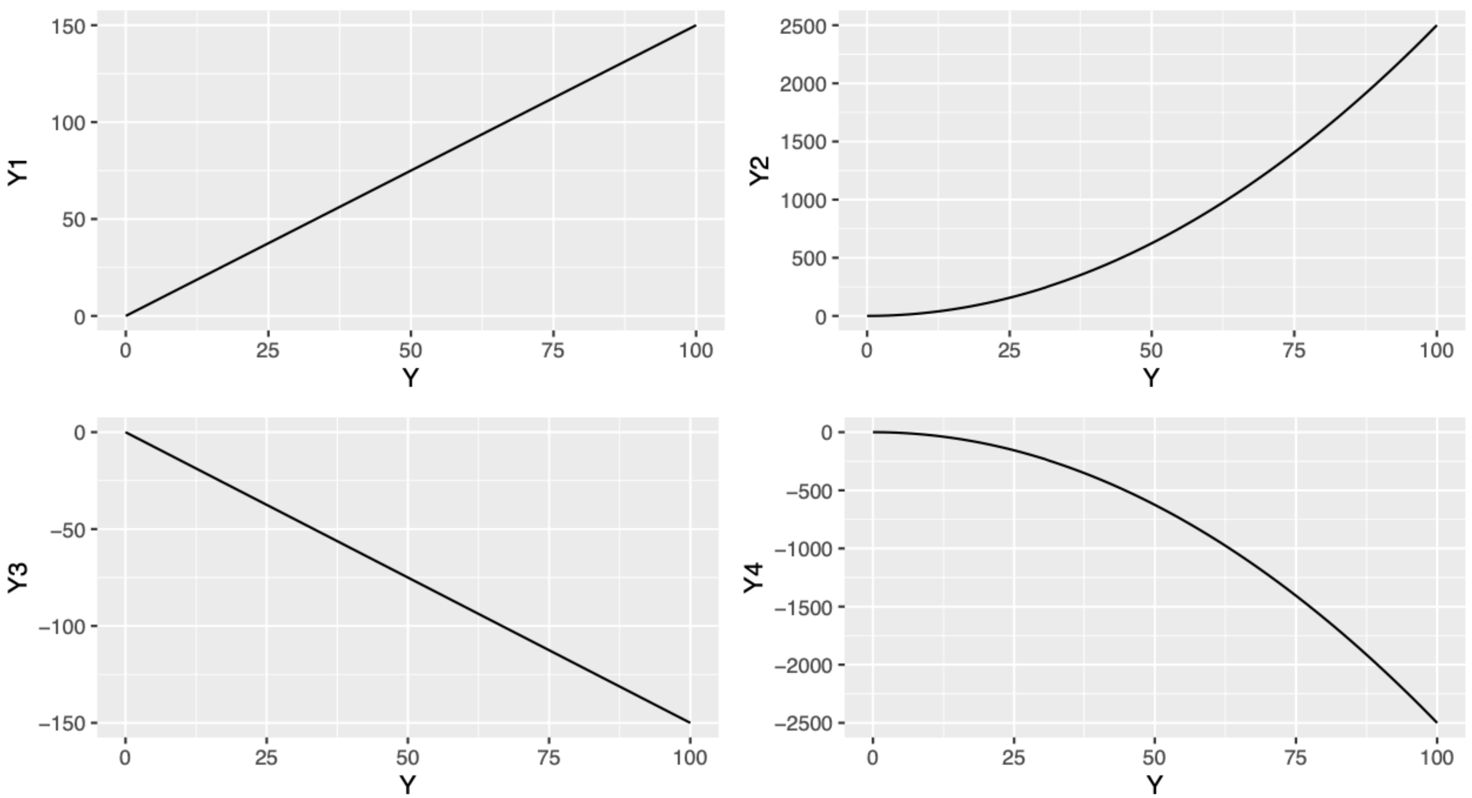
Anatomy of a time series

Some notation:

- $t = 1, 2, 3, \ldots,$ time index
- Y_t, is the value: of the variable of interest at time t
- Y_t may be composed of one or more components:
- *Trend*
- *Seasonal*
- *Cyclical*
- *Random*

Trend component

- A trend is persistent upwards or downwards movement in the data (not necessarily linear).

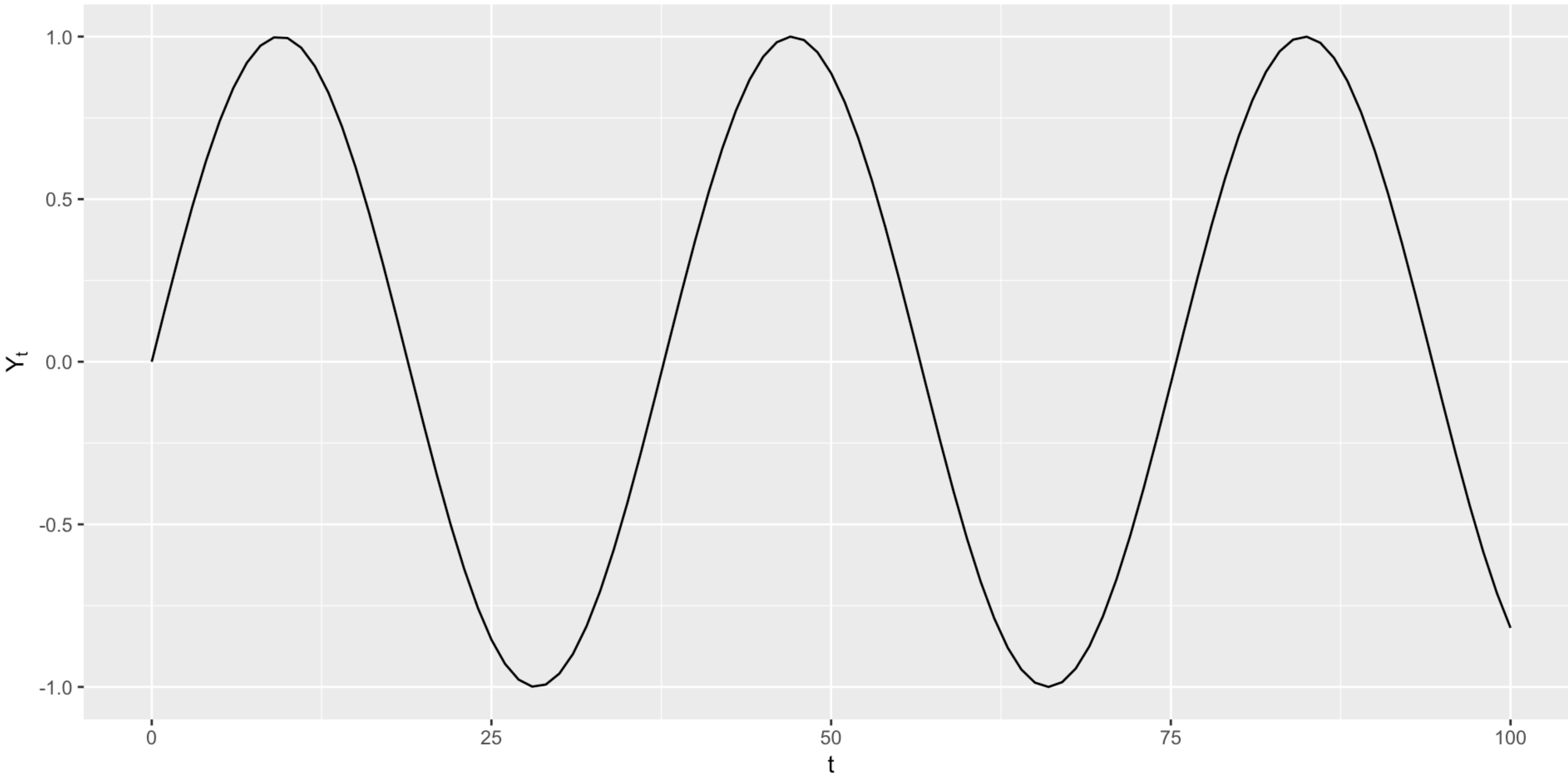


Trend component

- Example: *Moore's Law (accelerating increase of transistor count)*
- Example: *US population over time*
- A time series with no trend is called stationary.

Seasonal component

- Seasonal fluctuation occurs when predictable up or down movements occur over a regular interval.

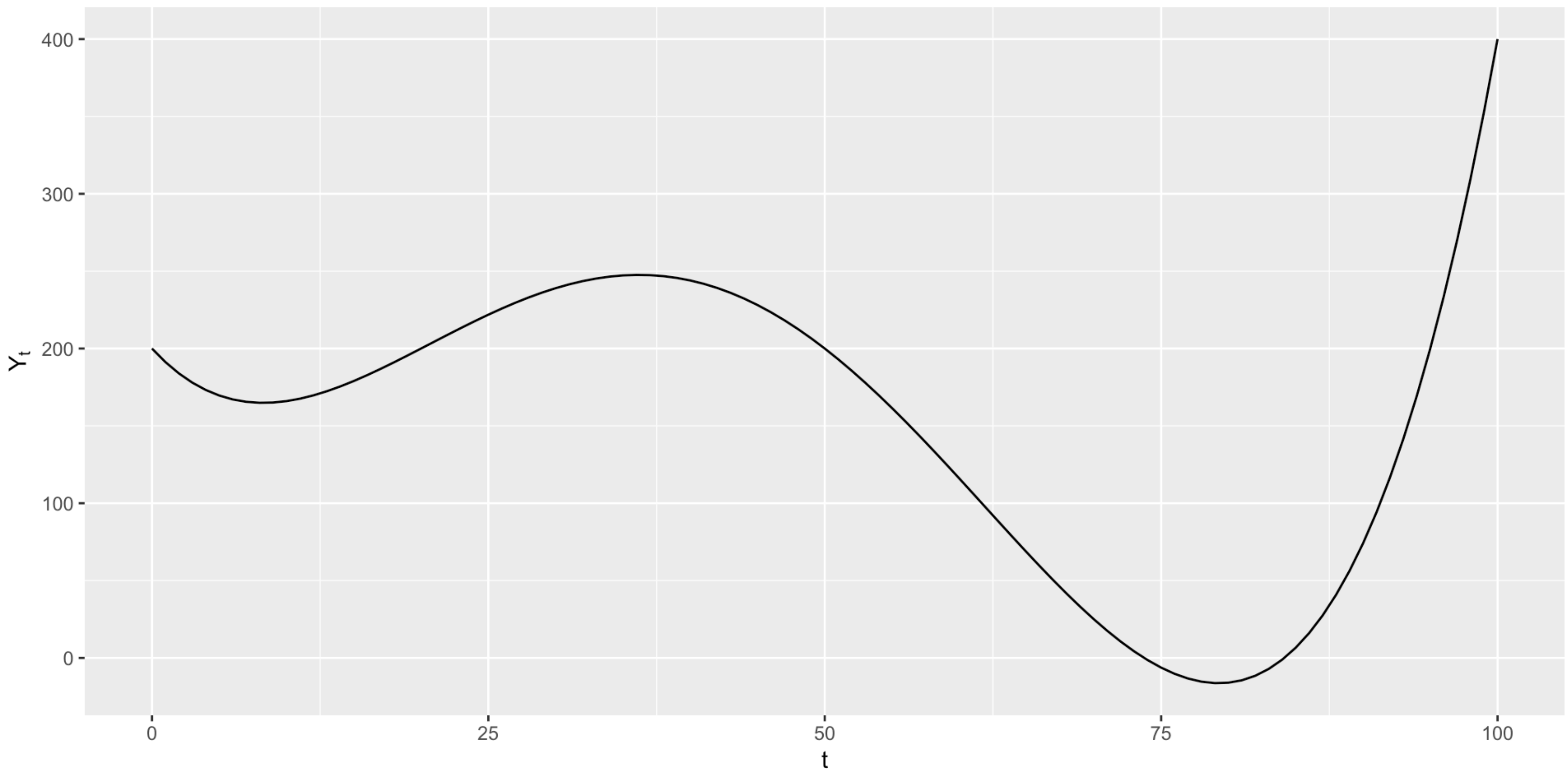


Seasonal component

- The ups and downs must occur over a regular interval (e.g., every month, or every year)
- Example: *Highway traffic volume is highest during rush hour every day*
- Example: *Supermarket sales may be highest every month right after common paydays like the 15th and 30th*

Cyclic component

- Cyclic fluctuations occur at unpredictable intervals, e.g. due to changing business or economic conditions.

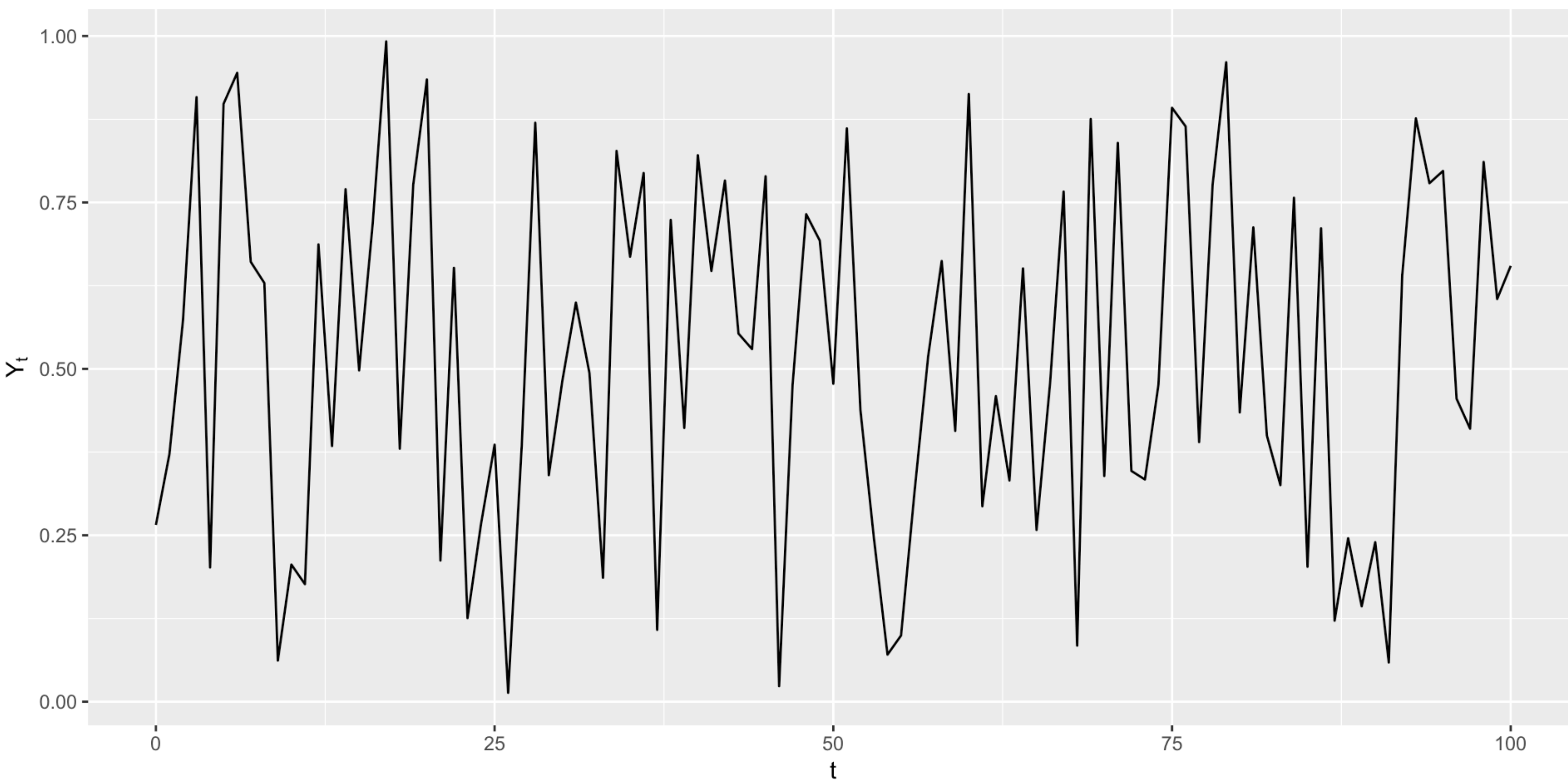


Cyclic component

- In contrast to seasonal fluctuations, cyclic fluctuations do not occur at regular, predictable intervals
- It may be possible to predict cyclic components based on some other (non-time) variable
- Example: *Restaurant sales dropped dramatically in 2020 due to COVID, as people ate out less*
- Example: *Sales of bell bottoms rose in the 60s and 70s, declined by the 80s, and then had a resurgence in the 90s*

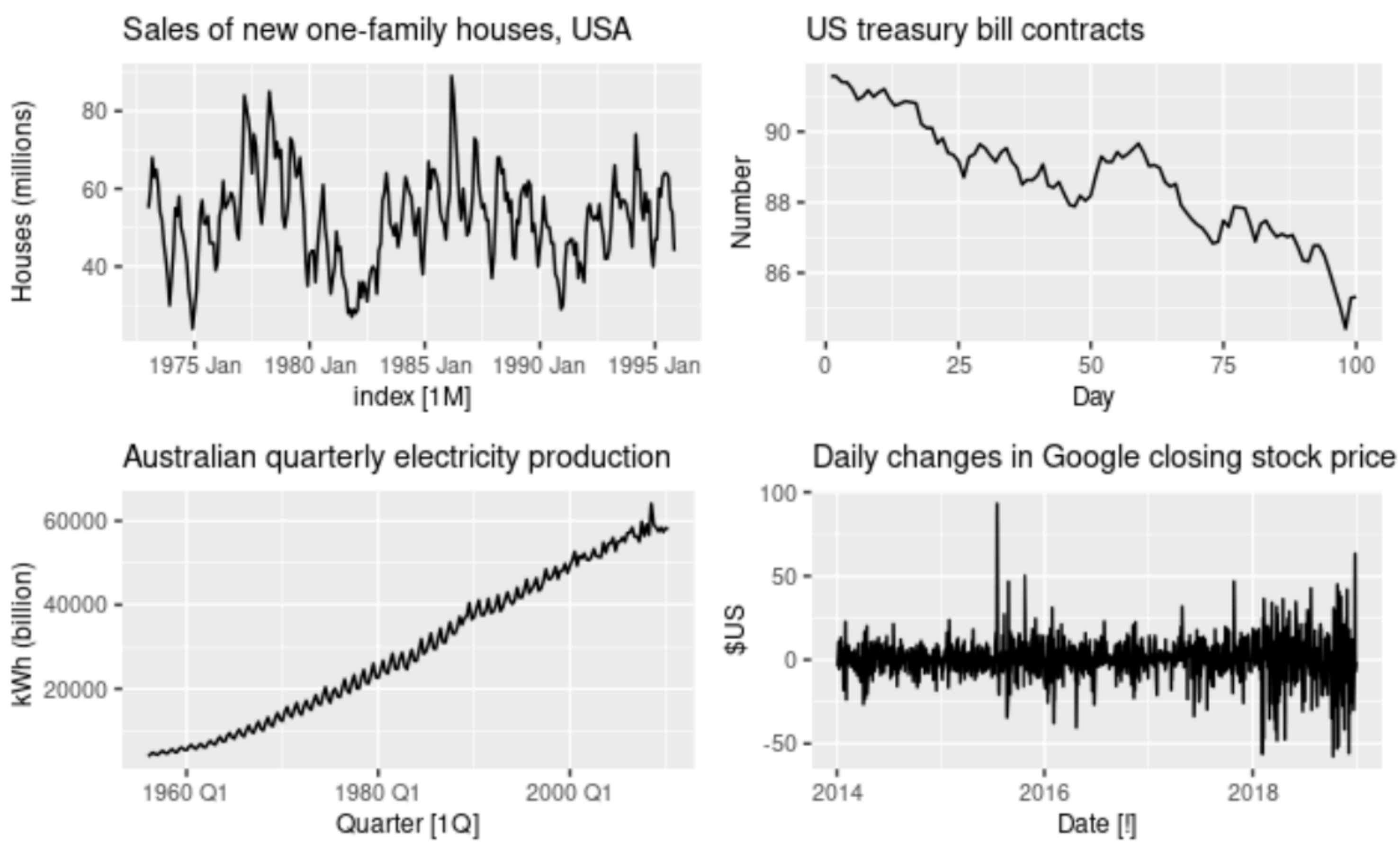
Remainder/Error component

- Any real time series will always have random noise as well, which can't be predicted or forecast.



Time Series Components

- Which component(s) you see in each of these time series?



Putting these together

Real time series will usually include a combination of these four components. We will model the time series Y_t either additively:

$$Y_t = \text{Trend} + \text{Seasonal} + \text{Random} = T_t + S_t + E_t$$

Or multiplicatively:

$$Y_t = \text{Trend} \cdot \text{Seasonal} \cdot \text{Random} = T_t \cdot S_t \cdot E_t$$

* (E_t consists of both the cyclic and error components, as both are unpredictable.) This model can be rewritten as a log model:

$$\log Y_t = \log(T_t) + \log(S_t) + \log(E_t)$$

Additive models

$$Y_t = \text{Trend} + \text{Seasonal} + \text{Random} = T_t + S_t + E_t$$

- Most appropriate when seasonal fluctuations are consistent (do not increase or decrease over time)
- The trend component T_t is a function of t (e.g., linear or quadratic)
- The seasonal component S_t is a set of dummy variable representing “seasons”
- So we can estimate additive models using regular regression

Additive decomposition

1. Run a regression predicting Y as a function of:

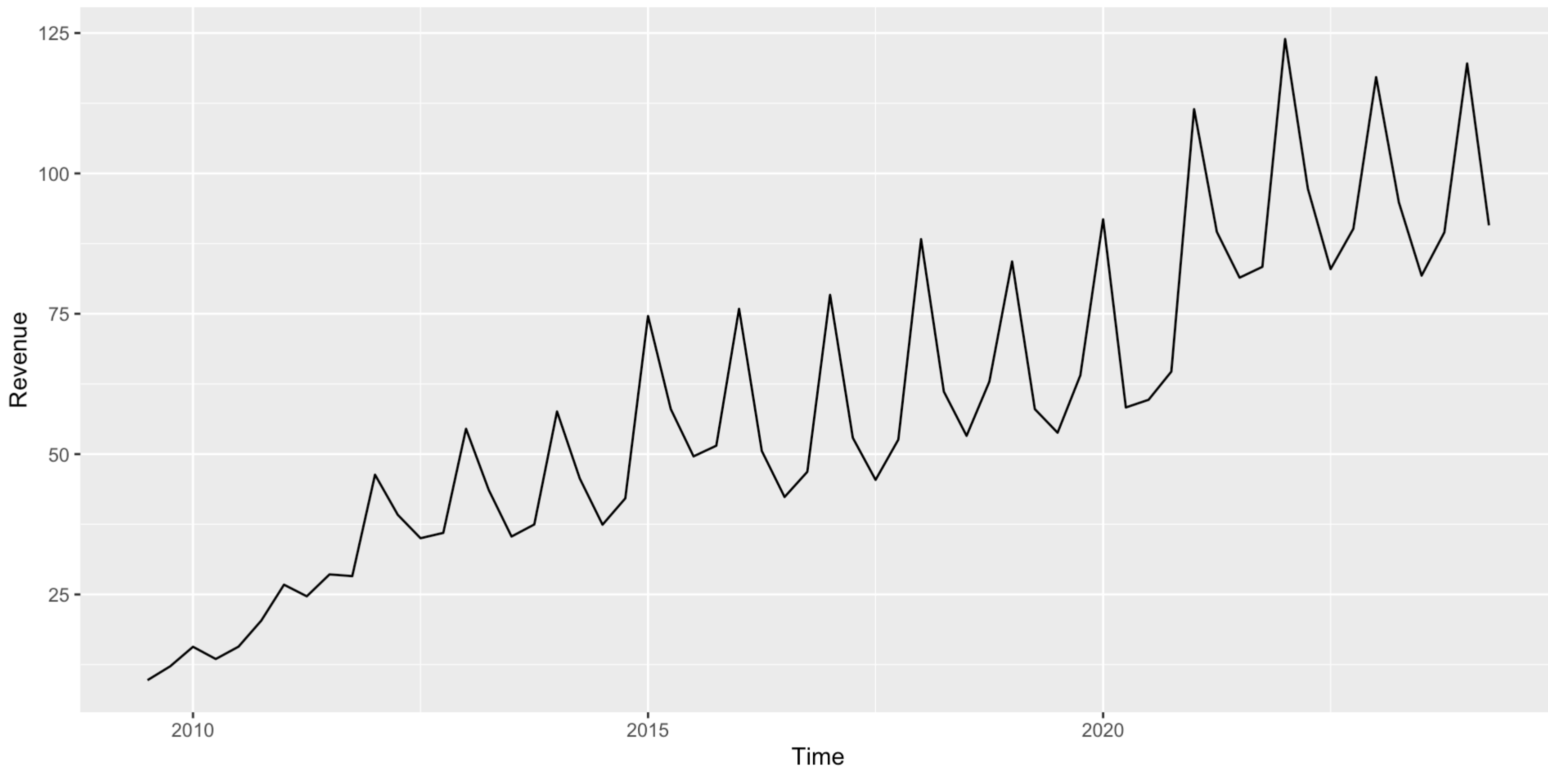
- t, t^2, $\log(t)$ etc (the trend component T_t)
- Dummy variables for the seasons (the seasonal component S_t)

2. To make a prediction for Y, plug into the model!
3. The residuals of this model correspond to the error component E_t

Apple quarterly revenue

- What components do you see here?

```
library(tidyverse)
ggplot(apple, aes(x=Time, y=Revenue)) +
  geom_line()
```



Fitting additive model

```r
lm_additive = lm(Revenue ~ Period + Quarter, data=apple)
summary(lm_additive)
```

```
Call:
lm(formula = Revenue ~ Period + Quarter, data = apple)

Residuals:
    Min      1Q  Median      3Q     Max 
-22.496  -5.135   1.280   4.923  17.928 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  33.93619    2.74731  12.353  < 2e-16 ***
Period        1.41324    0.05917  23.884  < 2e-16 ***
QuarterQ2   -20.62657    2.89298  -7.130 2.31e-09 ***
QuarterQ3   -27.44818    2.89480  -9.482 3.62e-13 ***
QuarterQ4   -24.20276    2.89298  -8.366 2.22e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.921 on 55 degrees of freedom
Multiple R-squared:  0.9269,	Adjusted R-squared:  0.9216 
F-statistic: 174.4 on 4 and 55 DF,  p-value: < 2.2e-16
```

Interpretation of the model

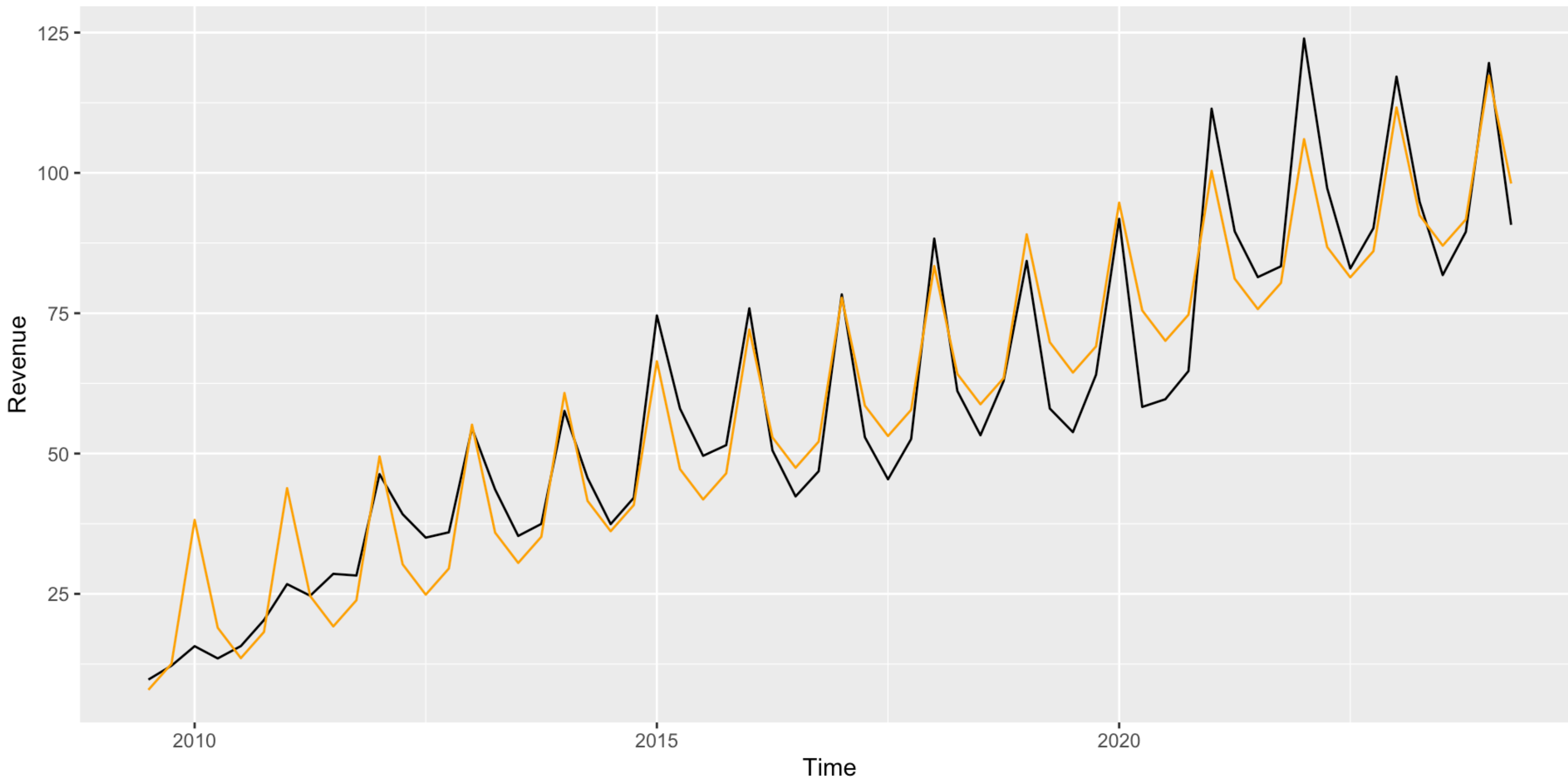
- The trend that we can infer from the variable `Period` indicates a positive growth in revenue of US$ 1.4 billion for each increase in the periods.
- The seasonal from the `Quarter` component indicates:

1. `Q2`'s are expected to be $20.7 worse than `Q1`'s
2. `Q3`'s are expected to be $27.4 worse than `Q1`'s
3. `Q4`'s are expected to be $24.2 worse than `Q1`'s
4. `Q3`'s are significantly worse than `Q1`'s

- These effects are statistically significant (`confint(lm_additive)`)
- The RSE from this model is US$ 7.921 billions of dollars.
- How can we interpret these results?

Fitting additive model

```r
ggplot(apple, aes(x = Time, y = Revenue)) +
  geom_line() +
  geom_line(aes(x = Time, y = predict(lm_additive)), col = "orange")
```



Fitting additive model

- What does the final model predict from the `Quarter` component indicates: for Apple in 2024 `Q3`?

```
predict(lm_additive, list(Period = 61, Quarter = "Q3"), interval = "prediction")
```

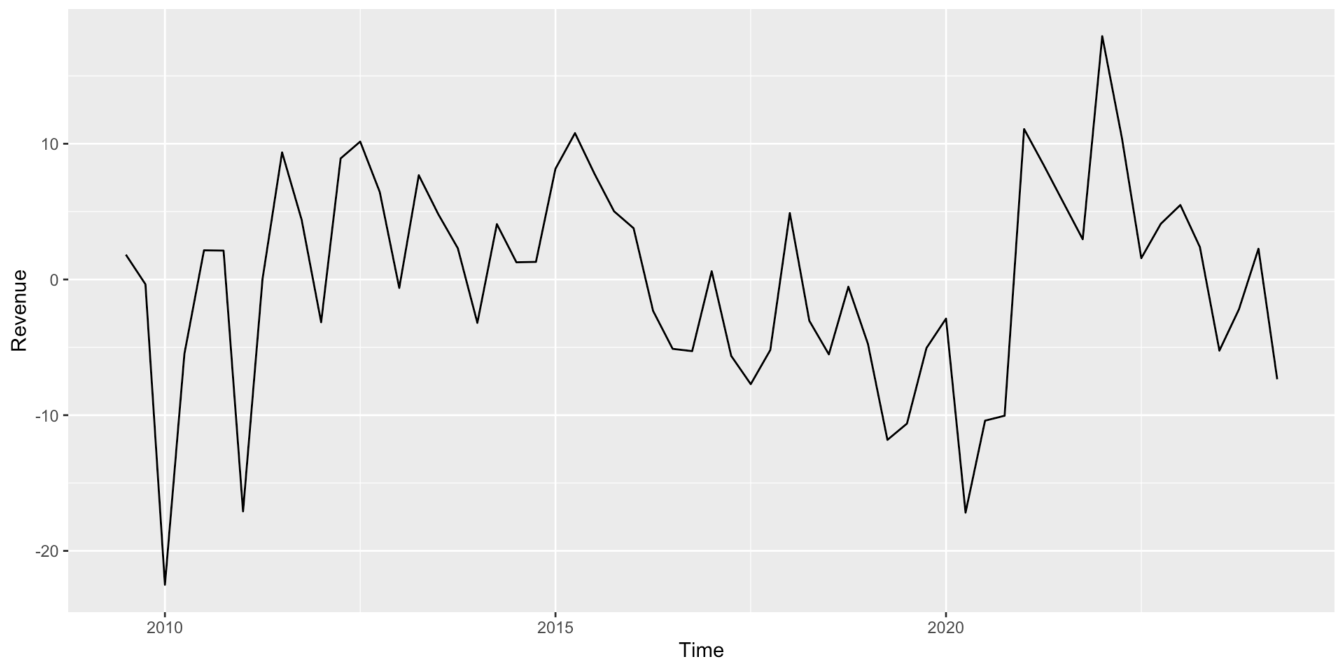
```
       fit      lwr     upr
1 92.69571 75.86745 109.524
```

- The actual revenue was US$ 85.78 billions
- What does the final model predict from the `Quarter` component indicates: for Apple in 2030 `Q1`? (Should we trust that prediction?)

Fitting additive model

- The residuals from this model show the "detrended and deasonalized" data (but there's still some trend left!):
- We hadn't yet dealt with the time dependence

```
ggplot(apple, aes(x = Time, y = Revenue)) +
  geom_line(aes(x = Time, y = residuals(lm_additive)))
```



Autorgression model

- How we deal with the time dependence ? Key idea: Instead of predicting Y_t as a function of t (or other variables), predict Y_t as a function of Y_{t-1}:

$$Y_t = \beta_0 + \beta_1 Y_{t-1} + e_t$$

- Y_{t-1} is called the “1st lag” of Y
- This is called autoregressive (AR) because it predicts the values of a time series based on previous values
- The model above is an AR(1) model
- We can have AR(p) models, with lag p

Autocorrelation

- Autocorrelation, is the correlation of Y_t with each of its lags $Y_t, Y_{t-1}, \ldots$

$$Cor(Y_t, Y_{t-1}), Cor(Y_t, Y_{t-2}), \ldots$$

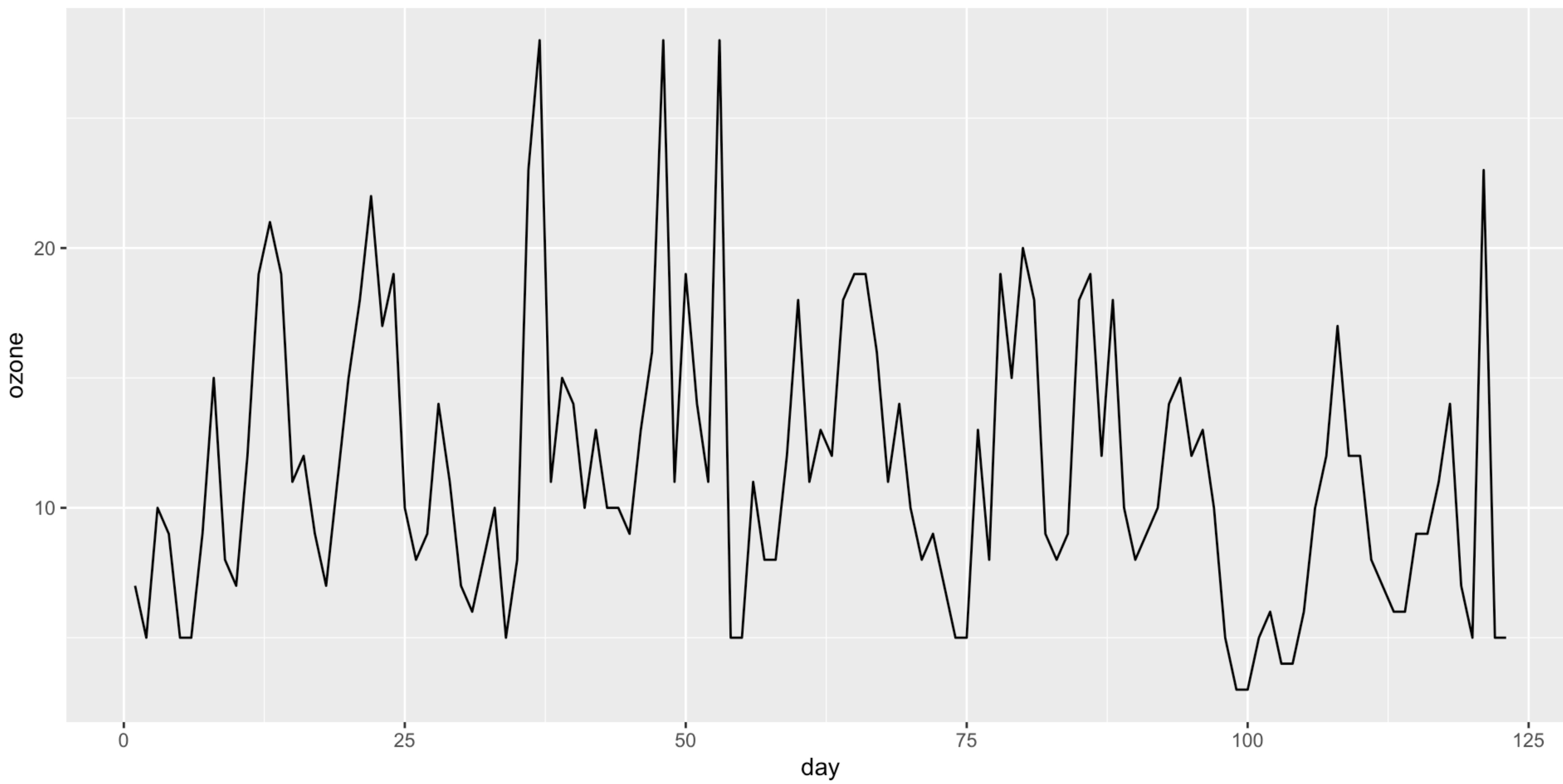
- We also have the autocorrelation of the residuals, r_t's, which indicates that there's a strong indication that the independence assumption is violated

$$Cor(r_t, r_{t-1}), Cor(r_t, r_{t-2}), \ldots$$

Ozone example

- Creating an AR(1) model: Daily ozone levels in Houston

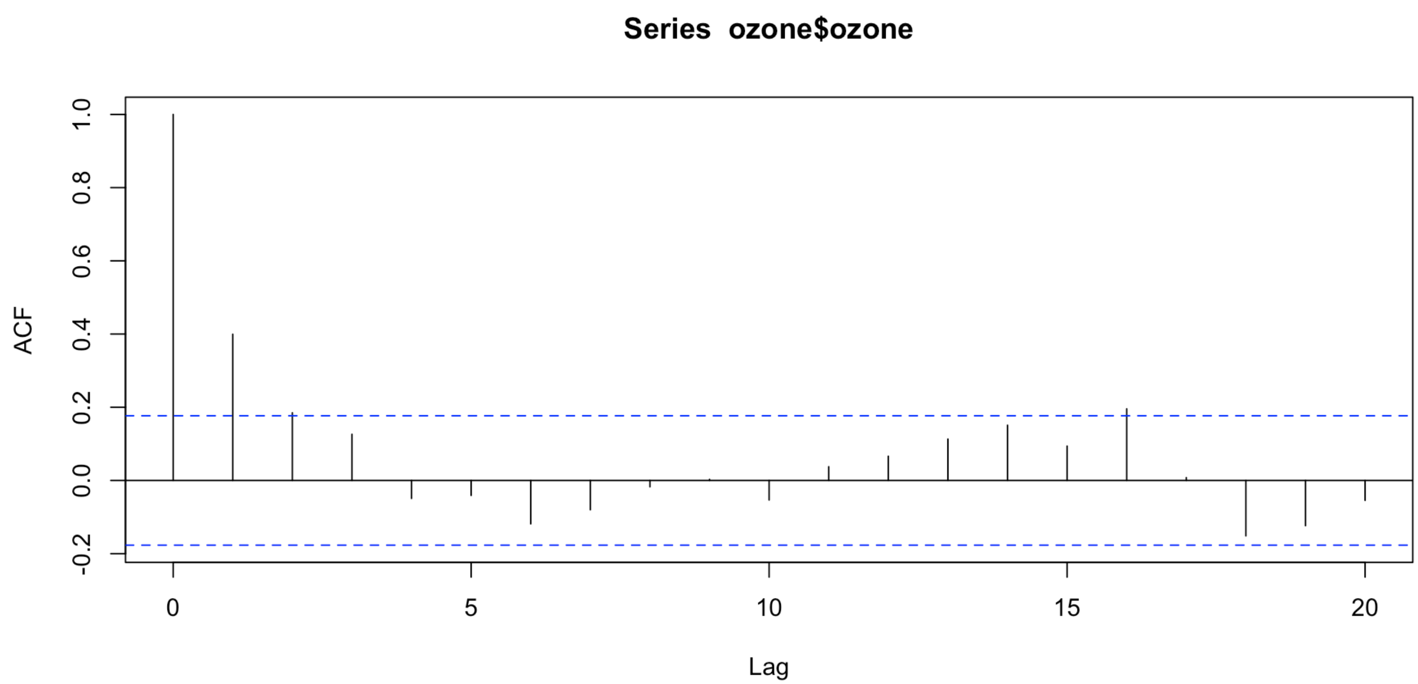
```
ggplot(ozone, aes(x = day, y = ozone)) +
  geom_line()
```



ACF plot

- Visualizing the autocorrelation function (ACF)

```
acf(ozone$ozone)
```



- Autocorrelations outside of the dashed blue lines are statistically significant.

Autorgression of the model

- We use the `lag` function to create the lagged observations

```
ozone <- ozone %>%
  mutate(lag1=lag(ozone))
ozone.model = lm(ozone ~ lag1, data=ozone)
summary(ozone.model)
```

```
Call:
lm(formula = ozone ~ lag1, data = ozone)

Residuals:
    Min      1Q  Median      3Q     Max 
-13.192  -3.464  -1.108   2.679  16.679 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.87446    1.06976   6.426 2.76e-09 ***
lag1         0.40419    0.08381   4.823 4.20e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.999 on 120 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.1624,	Adjusted R-squared:  0.1554 
F-statistic: 23.26 on 1 and 120 DF,  p-value: 4.197e-06
```

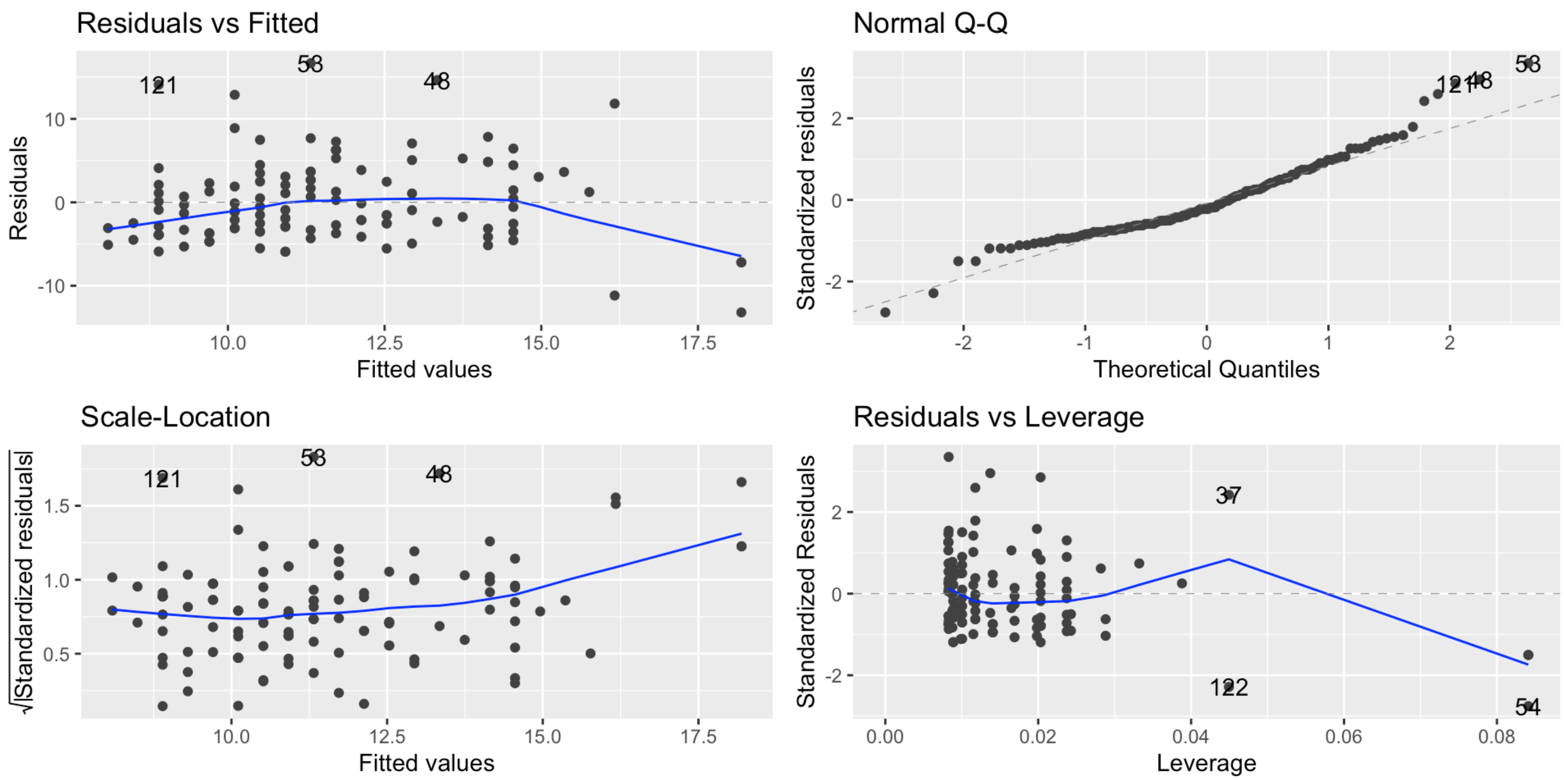
Assumptions of an AR(1) model

- Linearity, Normality, Equal Variance: Check using residual plot (linearity + homoscedasticity), Q-Q plot (normality), scale/location (homoscedasticity) like any other regression model
- Independence: Since this is a time series, we can actually check this by looking at the autocorrelation of the residuals (we want no significant autocorrelation)

Autoplot

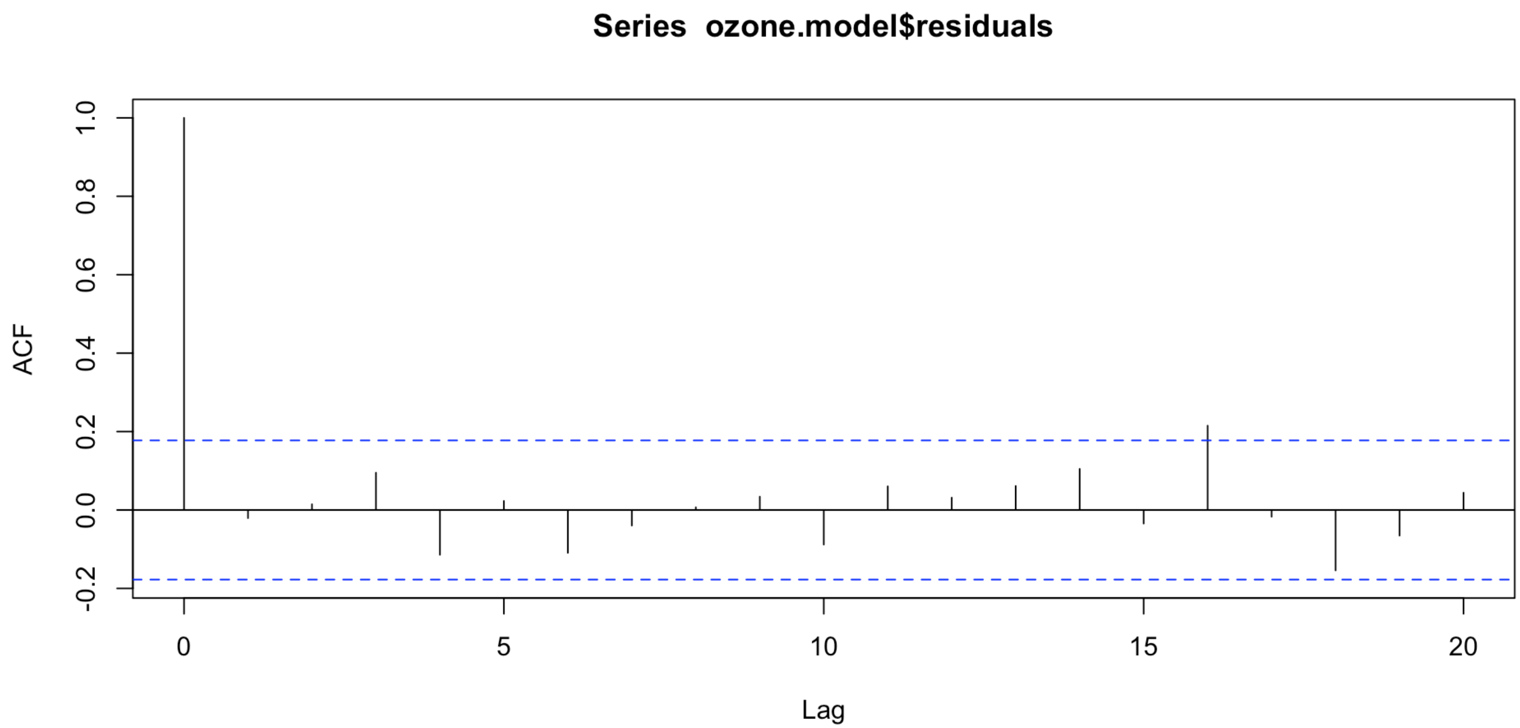
- Linearity, Normality, Equal Variance

```
autoplot(ozone.model)
```



ACF of the residuals

```r
acf(ozone.model$residuals)
```



- We expect 5% of autocorrelations to be significant just by chance, so having just 1 out of the 20 lags flagged as significant indicates we are OK on independence!

Making predictions in time series

Type	**Model**	**Predicted Y_t**
White noise	$Y_t = e_t$	0
Random sample	$Y_t = \beta_0 + e_t$	$\widehat{\beta}_0$ (or average Y)
Random walk	$Y_t = \beta_0 + Y_{t-1} + e_t$	$\widehat{\beta}_0 + Y_{t-1}$
General AR(1)	$Y_t = \beta_0 + \beta_1 Y_{t-1} + e_t$	$\widehat{\beta}_0 + \widehat{\beta}_1 Y_{t-1}$

- Unit root occurs when $\beta_1 = 1$. This means:
- The series is a **random walk**.
- There's **no mean reversion**, and any shocks will have a **permanent effect**.
- When $\beta_1 = 1$, the model is non-stationary, meaning the series tends to "drift" without stabilizing around a fixed mean.
- If $|\beta_1| < 1$, the series is **mean-reverting**, and shocks are **temporary**.

Statistical Analysis

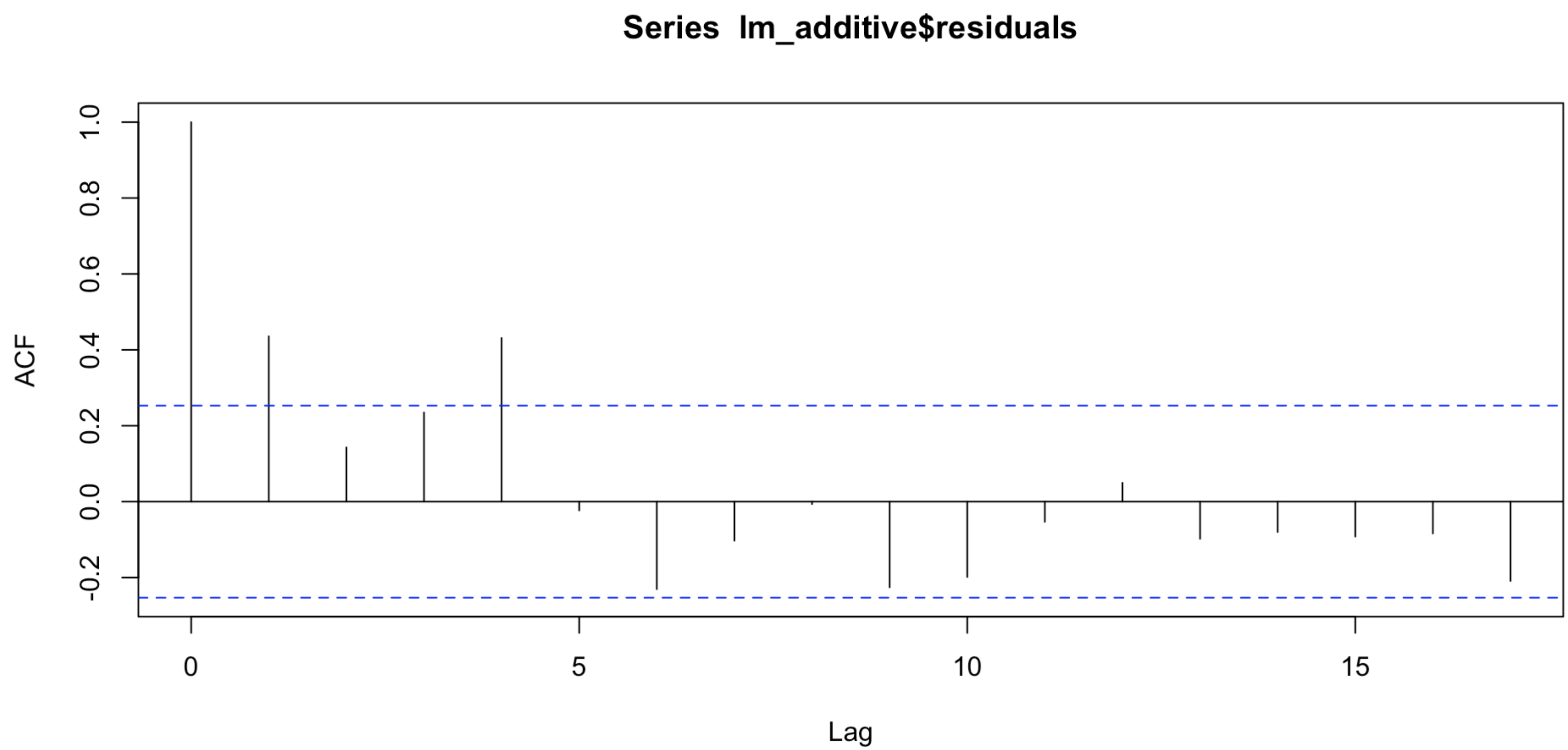
```
confint(ozone.model)
```

```
                2.5 %    97.5 %
(Intercept) 4.7564110 8.9925161
lag1        0.2382561 0.5701286
```

- The coefficient $\widehat{\beta}_1$ is associated with the variable `lag1`.
- In this case, for the larger population, with 95% confidence, $\widehat{\beta}_1$ lies between 0.24 and 0.57.
- This means that $|\beta_1| < 1$, indicating that the series is mean-reverting.

Apple Revenue ACF plot

- ACF plot of the residuals of the additive model.



Apple Revenue

- Combining decomposition and autoregression in a multiplicative model

$$\log(\texttt{Revenue}_t) = \log(\texttt{Period}_t) + \texttt{Quarter}_t + \log(\texttt{Revenue}_{t-1})$$

- We need to create the lag variable.
- It will have only one lag, and thus is an AR(1) model.

```
apple = apple %>%
  mutate(lag1 = lag(Revenue))
```

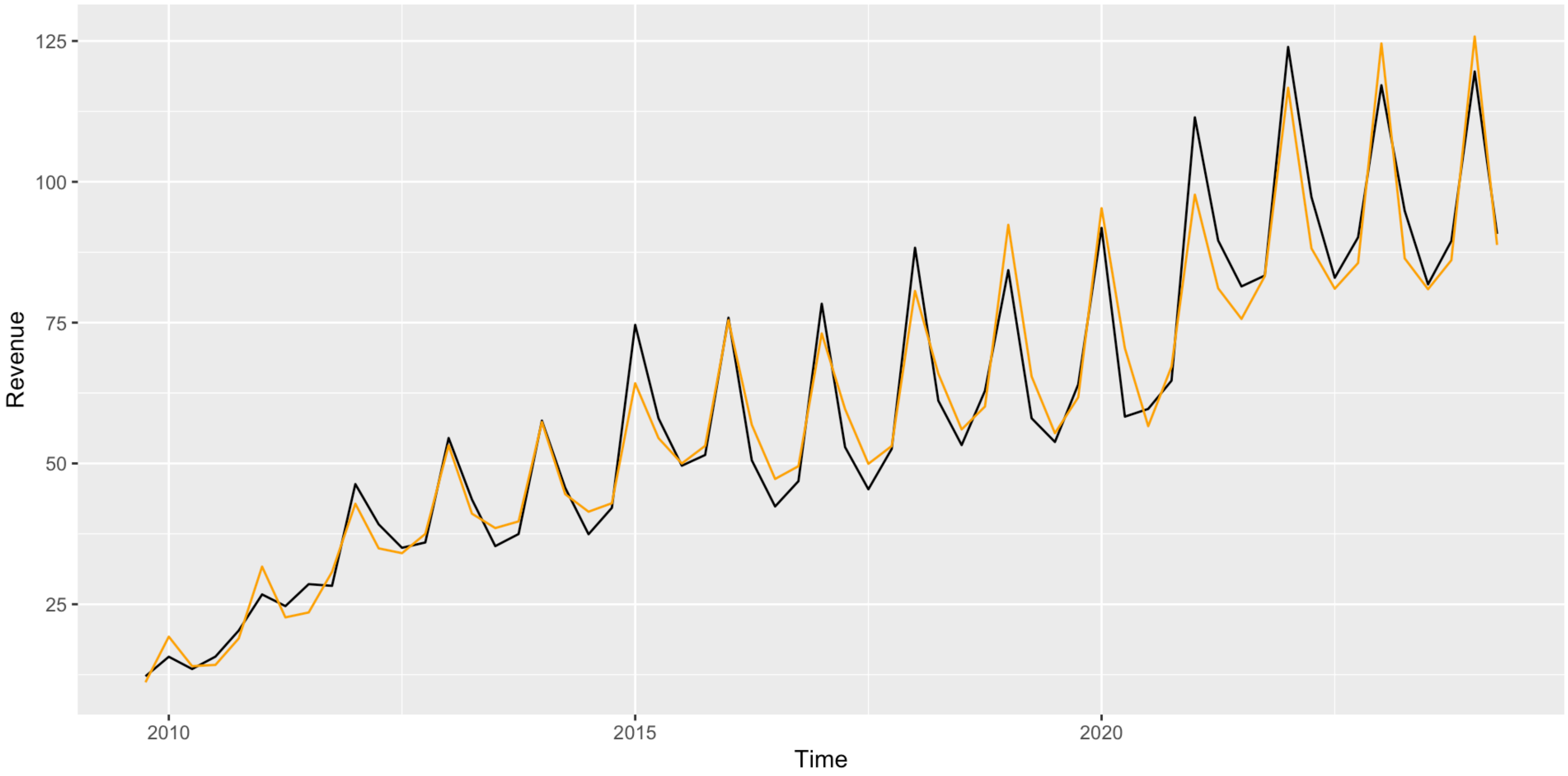
Apple Revenue

```r
log_apple = lm(log(Revenue) ~ log(Period) + Quarter + log(lag1), data = apple)
summary(log_apple)
```

```

Call:
lm(formula = log(Revenue) ~ log(Period) + Quarter + log(lag1),
    data = apple)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.204851 -0.056602  0.005991  0.066084  0.193337 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.14400    0.17945   6.375 4.56e-08 ***
log(Period)  0.20622    0.06918   2.981  0.00433 ** 
QuarterQ2   -0.53559    0.04911 -10.906 3.72e-15 ***
QuarterQ3   -0.47076    0.03397 -13.859  < 2e-16 ***
QuarterQ4   -0.31872    0.03346  -9.526 4.47e-13 ***
log(lag1)    0.63410    0.10109   6.273 6.65e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.09013 on 53 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.9751,    Adjusted R-squared:  0.9728
```

Apple Revenue Predictions

- Predictions of multiplicative model



Apple Revenue Predictions

- Confidence interval of the multiplicative model

```
confint(log_apple)
```

```
                  2.5 %     97.5 %
(Intercept)  0.78406737  1.5039420
log(Period)  0.06746219  0.3449861
QuarterQ2   -0.63409896 -0.4370871
QuarterQ3   -0.53888914 -0.4026276
QuarterQ4   -0.38583509 -0.2516142
log(lag1)    0.43133359  0.8368601
```

- The slope associated with lag is statistically significant, and its value is between minus and plus one; we have that this is a mean-reverting time series.
- We also have a better fit (here we feed `lag1` with prediction from the previous period, US$ 90.75 billions):

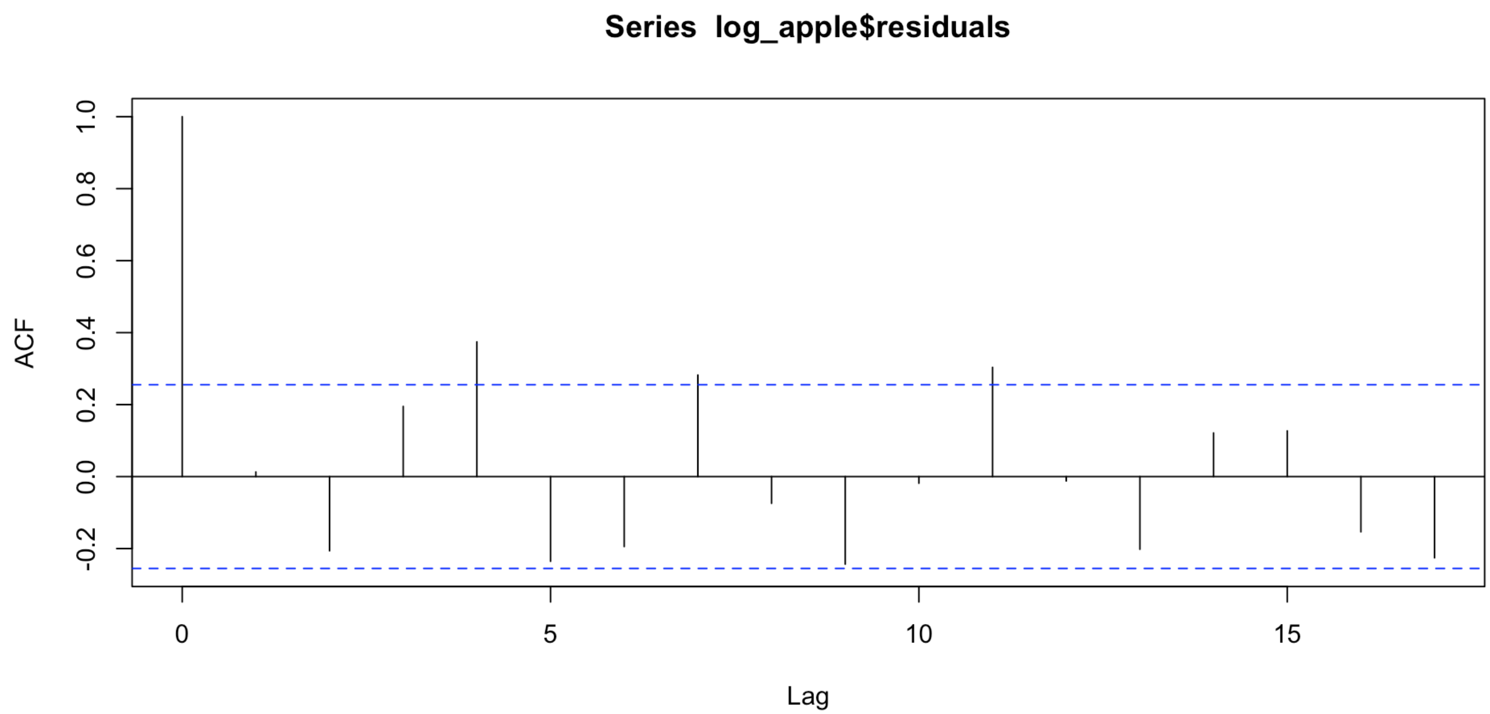
```
exp(predict(log_apple, list(Period = 61, Quarter = "Q3", lag1 = 90.75), interval = "prediction"))
```

```
       fit      lwr      upr
1 79.80492 66.06926 96.39618
```

- The confidence interval for the forecast is narrower, and the difference between what we observe and predict is smaller.

Apple Revenue ACF plot

- ACF plot of the residuals of the multiplicative model.



- The independent assumptions look better, but it might be necessary to add more lags.

Time Series Strategy

To building a time series model:

- Start with a an additive or multiplicative model with trend and seasonal components. (Plot your data! If the seasonal variation increases or decreases over time you'll want a multiplicative model.)
- Examine the usual diagnostic plots, and plot your residuals as a function of time. Do you need a (different) nonlinear time trend? A transformation of Y?
- Check your residuals for autocorrelation. If it's present, add appropriate lag terms to your model.