Time Series

Dr. Maria Tackett

12.02.19

STA 210

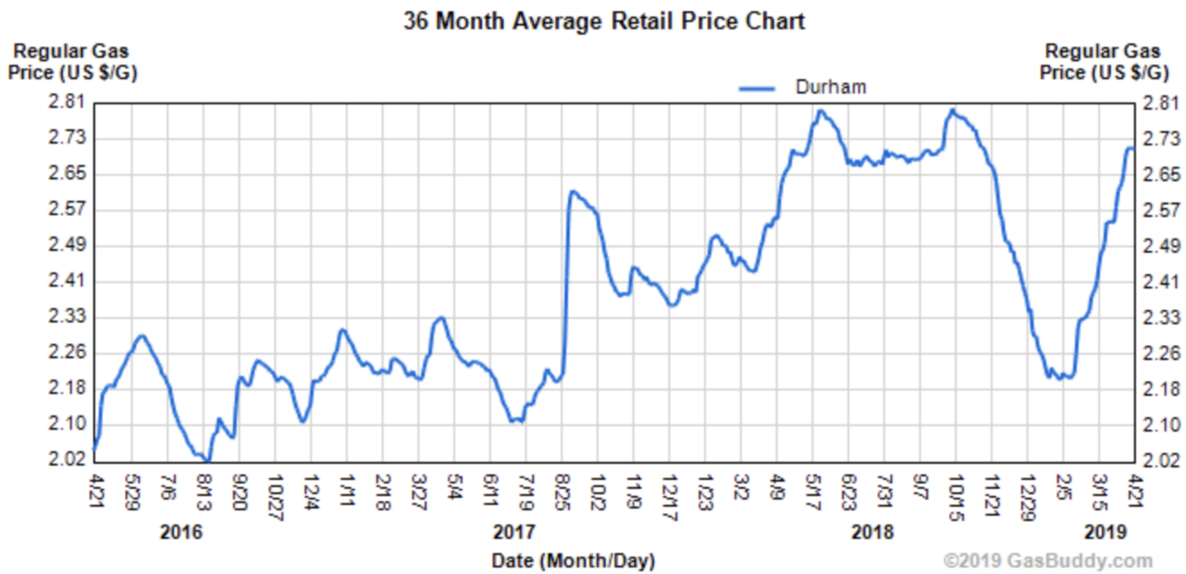
[Click for PDF of slides]()

Announcements

- Project write up due Dec 10 at 11:59p
- Project presentations on Dec 11
  - Lab 01L: 9a - 10:30a
  - Lab 02L: 10:30a - 12p
- Regression analysis feedback and grades by Wednesday
- Exam 2 grades by next Monday
- Exam 2 extra credit:
  - 90% response rate on course eval: +1 pt on Exam 02 grades

Examples of Time Series Data

Gas Prices in Durham



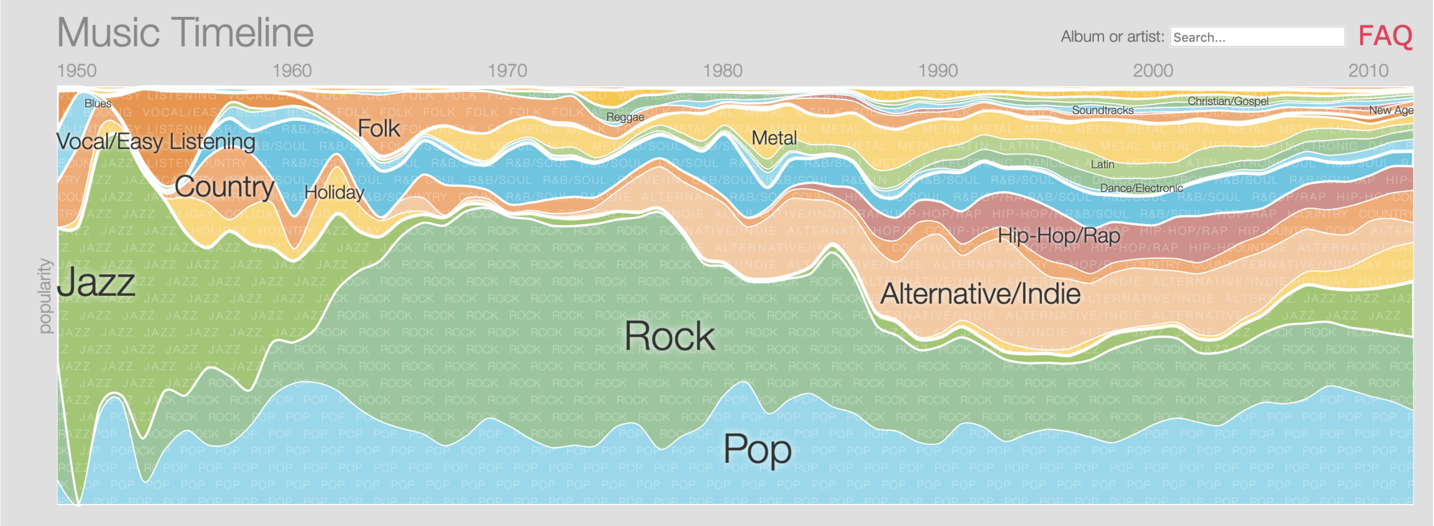
https://www.gasbuddy.com/Charts

STA 210

Apple's Stock

[Apple's Stock Price]

Google Music Timeline



http://research.google.com/bigpicture/music/

Retail Sales: 1999 - 2011

- **Goal:** Understand the change in total online sales from the fourth quarter of 1999 (Q4 1999) to the first quarter of 2011 (Q1 2011). The data may be found on the textbook website. It is originally from the U.S. Census Bureau.

- `online_sales`: Total online sales (in US dollars)
- `total_sales`: Total retail sales (in US dollars)

STA 210

Make a `ts` object

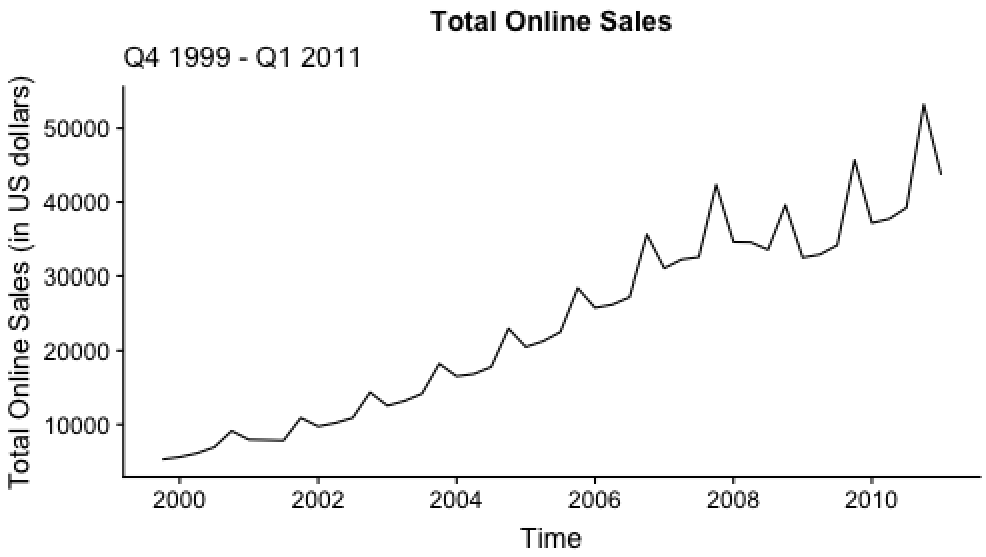
```
online_ts <- ts(ecommerce$online_sales,
                start = c(1999,4), #start in Q4 1999
                frequency = 4) #time periods are quarterly
```

```
online_ts
```

```
##       Qtr1  Qtr2  Qtr3  Qtr4
## 1999                    5286
## 2000  5592  6103  6940  9128
## 2001  7949  7899  7836 10909
## 2002  9738 10206 10908 14360
## 2003 12553 13199 14169 18236
## 2004 16533 16850 17796 22996
## 2005 20509 21284 22529 28482
## 2006 25814 26245 27246 35607
## 2007 31031 32218 32547 42349
## 2008 34595 34550 33541 39595
## 2009 32475 32902 34153 45684
## 2010 37166 37718 39230 53225
## 2011 43706
```

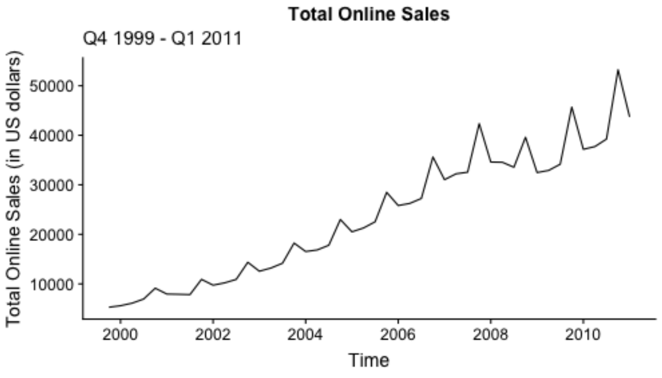
Plot online sales

```
autoplot(online_ts) +
  labs(title = "Total Online Sales",
       subtitle = "Q4 1999 - Q1 2011",
       y = "Total Online Sales (in US dollars)")
```



Trends and seasonality

- **Trending**: General movement in the data (increasing or decreasing)
  - Rescale variables so they are comparable over time (e.g. per capita variables)
  - Add **time** variables to the model
- **Seasonality**: Effects due to the season (e.g. sales during holidays)
  - Include indicator variables for the season



> Remove trends and seasonality, so you can focus on the relationships between the variables of interest

Online sales vs. total sales

```
ggplot(data = ecommerce , aes(x = total_sales, y = online_sales))
  geom_point() +
  labs(title = "Online sales vs. total sales",
       x = "Total Sales (in US Dollars)",
       y = "Online Sales (in US Dollars)")
```

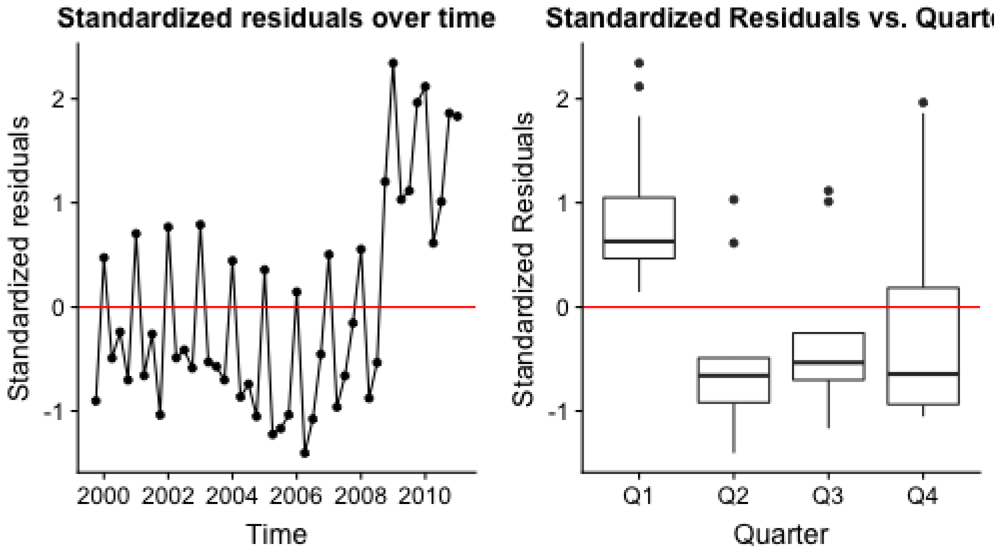
Online sales vs. total sales

term	estimate	std.error	statistic	p.value
(Intercept)	-73955.52	8119.590	-9.108	0
total_sales	0.11	0.009	12.104	0

r.squared	adj.r.squared
0.769	0.764

STA 210

Initial residuals plots



What do you learn from these residual plots?

Time Series

- One assumption for the regression methods we've used so far is that the observations are independent of one another
  - In other words, the residuals are independent
- When data is ordered over time, errors in one time period may influence error in another time period
- We'll use time series analysis to deal with this serial correlation
  - Assume the observations are measured at equally spaced time points
- Today's class is a brief introduction to time series analysis
  - *STA 444: Statistical Modeling of Spatial and Time Series Data* for more in-depth study of the subject

STA 210

Autocorrelation

- We want a measure of the correlation between the observation at time t and the observation at time $t-k$
  - k is the **lag**
- To do so, we will compute the correlation between the observations (or residuals) at time t and time $t-k$
  - This is the **autocorrelation coefficient**
- The formula for the Lag *k* autocorrelation coefficient is

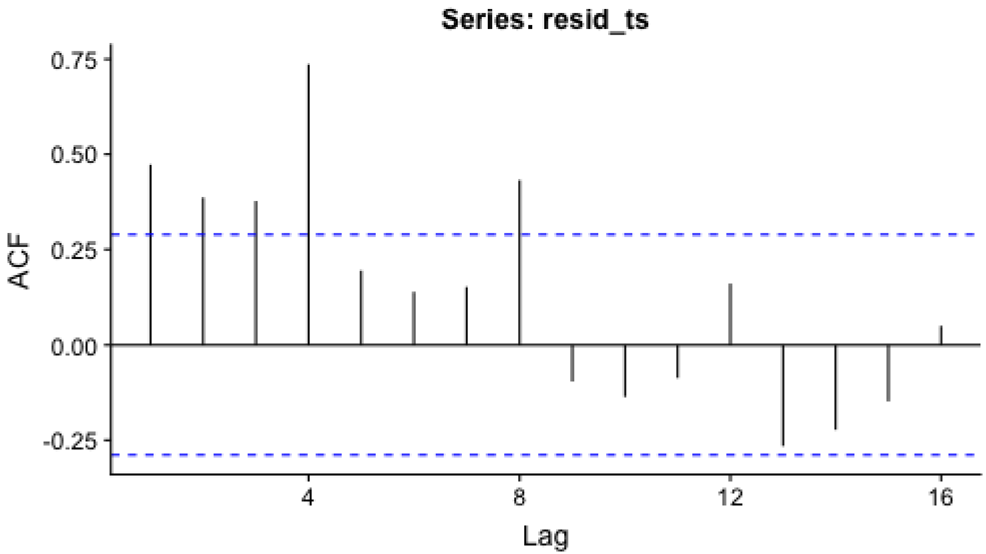
$$\hat{\rho}_k = \frac{\sum_{i=k+1}^{n} e_i e_{i-k}}{\sum_{i=1}^{n} e_i^2} = \frac{\sum_{i=k+1}^{n} (y_i - \bar{y})(y_{i-k} - \bar{y})}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

STA 210

Online Sales: Autocorrelation

- We can use the **`ggAcf()`** function in the **forecast** package to calculate the autocorrelation coefficient

```
ggAcf(resid_ts)
```



Add `quarter` to the model

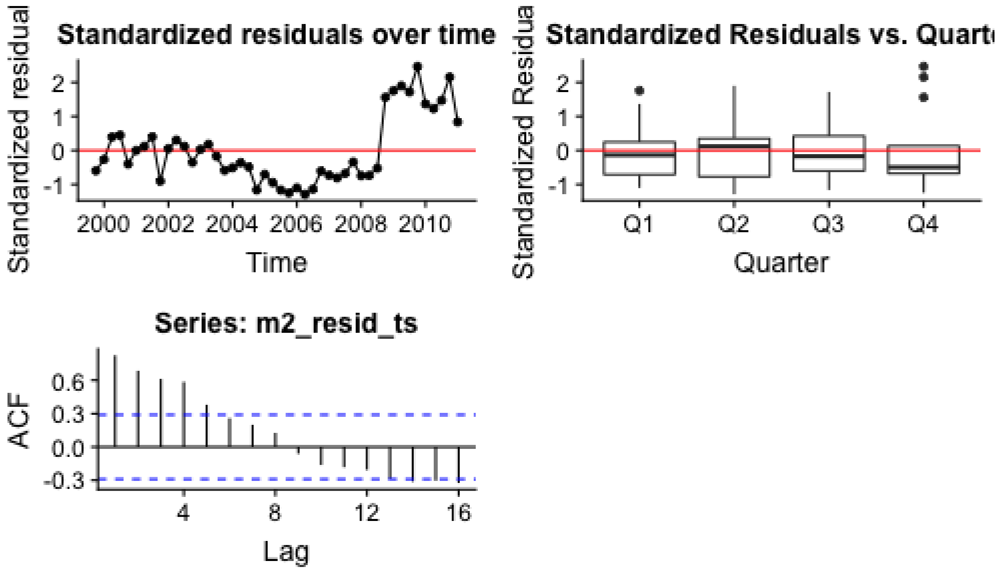
One way to deal with seasonality is to add indicator variables to the model

term	estimate	std.error	statistic	p.value
(Intercept)	-77815.038	6910.333	-11.261	0.000
total_sales	0.122	0.008	14.972	0.000
quarterQ2	-9672.053	2256.277	-4.287	0.000
quarterQ3	-8326.059	2246.124	-3.707	0.001
quarterQ4	-7564.031	2274.945	-3.325	0.002

r.squared	adj.r.squared
0.85	0.835

STA 210

Updated residual plots



- What event happened in late 2008 that can possibly explain the large jump in the residuals?
- What do the high residuals tell you about online sales during this time period?

Account for the recession

```
ecommerce %>%
  filter(Recession == 1) %>%
  select(Quarter)
```

```
## # A tibble: 10 x 1
##    Quarter
##    <chr>
##  1 4th quarter 2008
##  2 1st quarter 2009
##  3 2nd quarter 2009
##  4 3rd quarter 2009
##  5 4th quarter 2009
##  6 1st quarter 2010
##  7 2nd quarter 2010
##  8 3rd quarter 2010
##  9 4th quarter 2010
## 10 1st quarter 2011
```

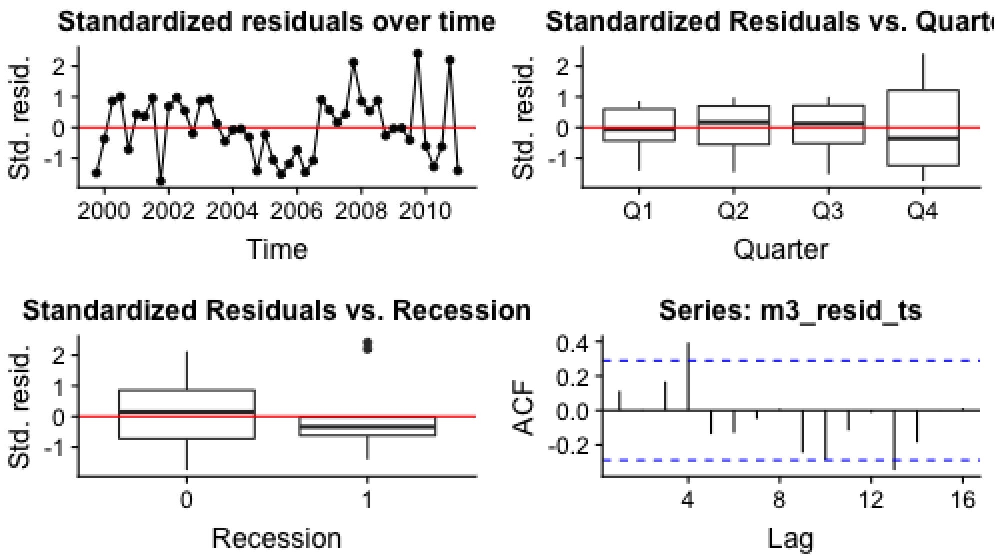
Add `Recession` to the model

term	estimate	std.error	statistic	p.value
(Intercept)	-66188.404	2641.127	-25.061	0
total_sales	0.104	0.003	32.721	0
quarterQ2	-7660.904	839.184	-9.129	0
quarterQ3	-6408.117	834.656	-7.678	0
quarterQ4	-5888.218	843.193	-6.983	0
Recession1	11930.012	735.635	16.217	0

r.squared	adj.r.squared
0.98	0.978

STA 210

Residual Plots



There is still some seasonality that isn't accounted for by the model. A more complex "deseasonalizing" method such as using yearly lags and moving averages may be required for this data. This is typically what is used by the U.S.Census Bureau when analyzing economic data.

Autoregressive Model

Autoregressive Model

- One way to deal with serial correlation is to use values of the response from previous time periods as a predictor in the model
- This is the basic structure of the **autoregressive (AR) model**
- The AR model with one lag, the **AR(1) model**, is

$$y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} + \beta_{p+1} y_{i-1} + \epsilon_i \qquad \epsilon_i \sim N(0, \sigma^2)$$

- Further lags, y_{i-2}, y_{i-3}, etc. can also be used.
    - Use the ACF plot to determine the lags to use in the model.

Online sales: Create lagged variable

- Create variable of online sales lagged by 1 time period

```
ecommerce <- ecommerce %>%
  mutate(online_sales_lag1 = lag(online_sales, n = 1))
```

```
ecommerce %>%
  select(online_sales, online_sales_lag1) %>%
  slice(1:5)
```

```
## # A tibble: 5 x 2
##   online_sales online_sales_lag1
##          <dbl>             <dbl>
## 1         5286                NA
## 2         5592              5286
## 3         6103              5592
## 4         6940              6103
## 5         9128              6940
```

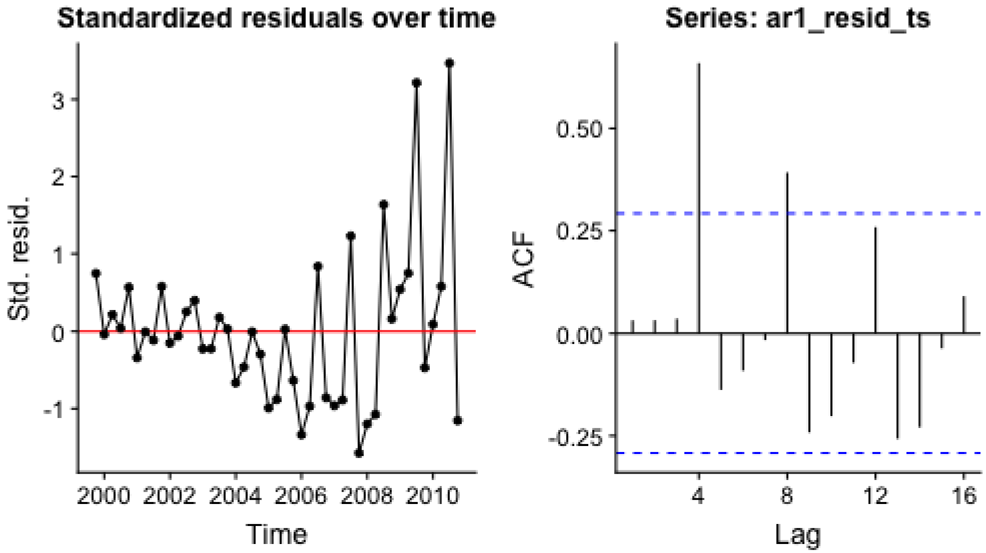
Online sales: AR(1) model

```
ar_1_model <- lm(online_sales ~ total_sales + online_sales_lag1,
                 data = ecommerce)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-33721.530	5647.694	-5.971	0
total_sales	0.048	0.007	6.485	0
online_sales_lag1	0.643	0.060	10.726	0

STA 210

Residual plots



STA 210

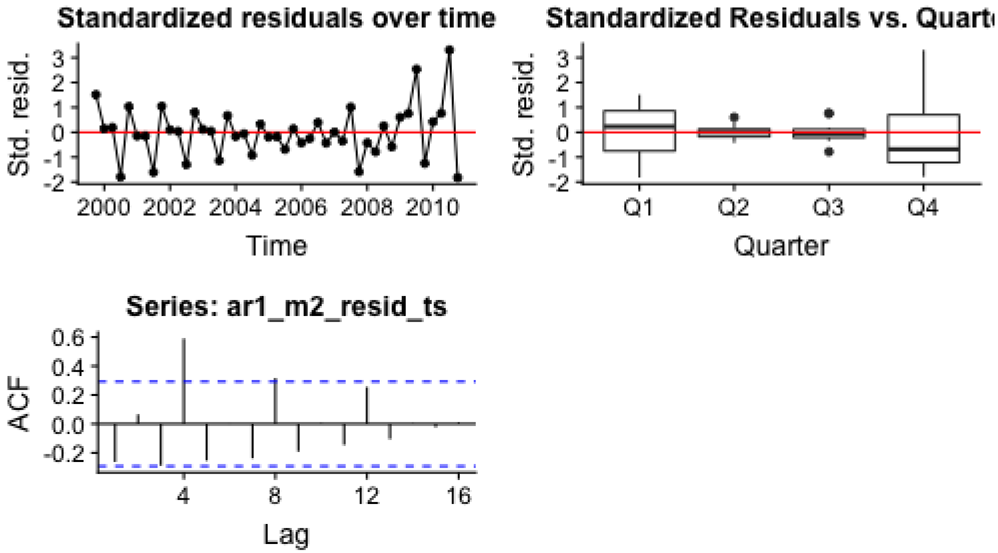
Add `quarter` to the AR(1) model

```
ar_1_m2 <- lm(online_sales ~ total_sales + quarter +
                online_sales_lag1,
                data = ecommerce)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-15119.571	6265.219	-2.413	0.021
total_sales	0.018	0.010	1.908	0.064
quarterQ2	2365.232	1486.879	1.591	0.120
quarterQ3	2760.552	1429.119	1.932	0.061
quarterQ4	8009.110	1721.425	4.653	0.000
online_sales_lag1	0.850	0.072	11.741	0.000

r.squared	adj.r.squared
0.966	0.961

Updated residual plots



We have improved upon the previous models, but there is still some seasonality that isn't accounted for by the model. As stated before, a more complex "deseasonalizing" method such as using yearly lags and moving averages is required for this data.

Interpretation

term	estimate	std.error	statistic	p.value
(Intercept)	-15119.571	6265.219	-2.413	0.021
total_sales	0.018	0.010	1.908	0.064
quarterQ2	2365.232	1486.879	1.591	0.120
quarterQ3	2760.552	1429.119	1.932	0.061
quarterQ4	8009.110	1721.425	4.653	0.000
online_sales_lag1	0.850	0.072	11.741	0.000

- Interpret `total_sales` in context of the data.
- Interpret `online_sales_lag1` in context of the data.
- Interpret the intercept in context of the data. Is the intercept meaningful?

Further Reading

- *Handbook of Regression Analysis*: Chapter 5
- *Time Series: A Data Analysis Approach* by Shumway and Stoffer
  - introductory text
- *Time Series Analysis and Its Applications* by Shumway and Stoffer
  - graduate-level text
  - freely available online

STA 210