

News Sentiment vs Airline Companies

2023-07-11

Introduction

This was a project I completed in my upper-division writing class during my final quarter attending UC Irvine. Our assignment was to research the any industry of the stock market and see if we could find any variables that help explained how it fluctuates. For my project, I chose to see how news sentiment relating to the stock market influences the airline industry.

News sentiment data was taken from the paper “Measuring News Sentiment” by Adam Shapiro, Moritz Sudhof, and Daniel Wilson. “Yahoo Finance” and the “Wall Street Journal” were used to collect the stock prices of the selected companies. Each of the data was collected daily and covers from May 4, 2007, to April 5, 2021. Although most of the data sources have a date range wider than the selected dates, it was chosen because it is the longest time period where all the data sources overlap. The five selected airline companies include American Airlines, Alaska Air Group, Delta Air Lines, Southwest Airlines, and United

The regression is based off the capital asset pricing model (CAPM) and uses ordinary least squares (OLS) in order to estimate the coefficients. The first regression I did for the paper was a linear regression, but the results were far from satisfactory. After conducting a Breusch-Pagan test, I found evidence for heteroskedasticity. To take this into account, I ran another regression with robust standard errors. Finally, I added dummy variables for the 2008 recession as well as the COVID-19 pandemic to see how much they impacted the results. For a more detailed summary of my process, check the pdf file of my submitted paper.

Project Setup

Libraries

```
library(readxl)
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   1.0.1
## v tibble  3.1.8      v dplyr  1.0.10
## v tidyr   1.3.0      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(estimatr)
library(lmtest)
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(stats)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(broom)
```

Loading data

```
# News Sentiment Data
news_sentiment_data <- read_excel("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/news_sentiment_data.xlsx")

# S&P 500 Data
sp500_data <- read_csv("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/sp500_data.csv")
```

```
## Rows: 3504 Columns: 5
## -- Column specification -----
## Delimiter: ","
## chr (1): Date
## dbl (4): Open, High, Low, Close
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
# Stock Data
aal_data <- read.csv("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/aal_data.csv")
alk_data <- read.csv("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/alk_data.csv")
dal_data <- read.csv("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/dal_data.csv")
luv_data <- read.csv("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/luv_data.csv")
ual_data <- read.csv("C:/Users/onday/OneDrive/Desktop/Econ 137W Paper/ual_data.csv")
```

Data Preparation

News Sentiment

```
# Only data from May 3, 2007
news_sentiment_data <- news_sentiment_data %>%
  filter(date > '2007-05-03')

# Transform 'date' column to date data type
news_sentiment_data$date <- as.Date(news_sentiment_data$date)
```

S&P 500

```
# Transforming 'date' column to date data type
sp500_data$Date <- as.Date(sp500_data$Date, format = "%m/%d/%Y")

# Calculate daily returns
sp500_data <- sp500_data %>%
  mutate(sp500_returns = ((Close - lag(Close)) / lag(Close)) * 100)
```

Airline Companies

```
# Transform 'date' column to date data type
aal_data$Date <- as.Date(aal_data$Date, format = "%m/%d/%Y")
alk_data$Date <- as.Date(alk_data$Date, format = "%m/%d/%Y")
dal_data$Date <- as.Date(dal_data$Date)
luv_data$Date <- as.Date(luv_data$Date)
ual_data$Date <- as.Date(ual_data$Date)

# Stock data
stock_data <- list(aal_data, alk_data, dal_data, luv_data, ual_data)

# Calculating daily returns
returns <- function(df) {
  df %>%
    mutate(returns = ((Adj.Close - lag(Adj.Close)) / lag(Adj.Close)) * 100)
}
```

```

for (i in seq_along(stock_data)){
  stock_data[[i]] <- returns(stock_data[[i]])
}

# Renaming returns columns
stock_data[[1]] <- rename(stock_data[[1]], aal_returns = returns)
stock_data[[2]] <- rename(stock_data[[2]], alk_returns = returns)
stock_data[[3]] <- rename(stock_data[[3]], dal_returns = returns)
stock_data[[4]] <- rename(stock_data[[4]], luv_returns = returns)
stock_data[[5]] <- rename(stock_data[[5]], ual_returns = returns)

# Putting all stock return data into a single data frame
return_data <- stock_data %>%
  reduce(full_join, by = "Date") %>%
  select(Date, ends_with("returns"))

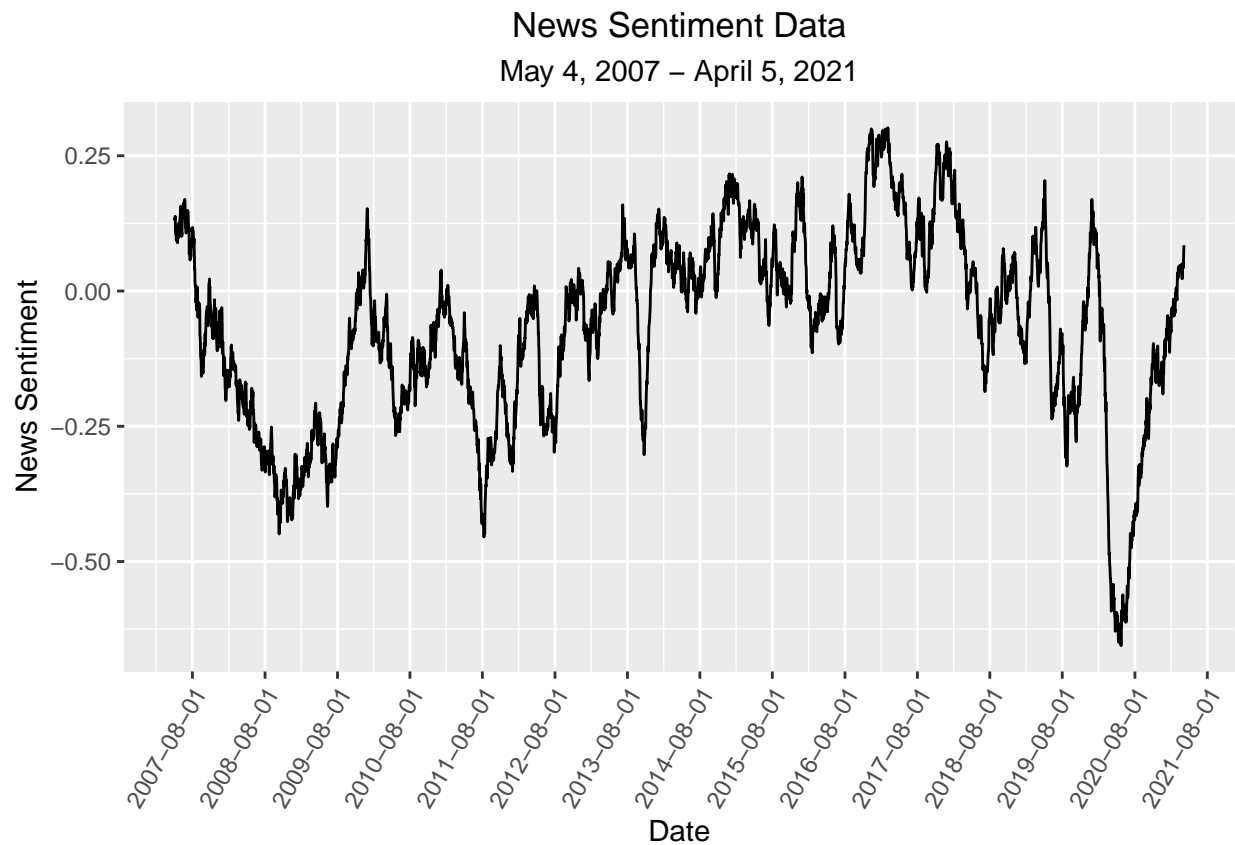
```

Data Exploration

```

# Time series chart for news sentiment
ggplot(news_sentiment_data, aes(x = date, y = news_sentiment)) +
  geom_line() +
  labs(x = 'Date',
       y = 'News Sentiment',
       title = 'News Sentiment Data',
       subtitle = 'May 4, 2007 - April 5, 2021') +
  scale_x_date(breaks = "12 month") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 60, hjust = 1))

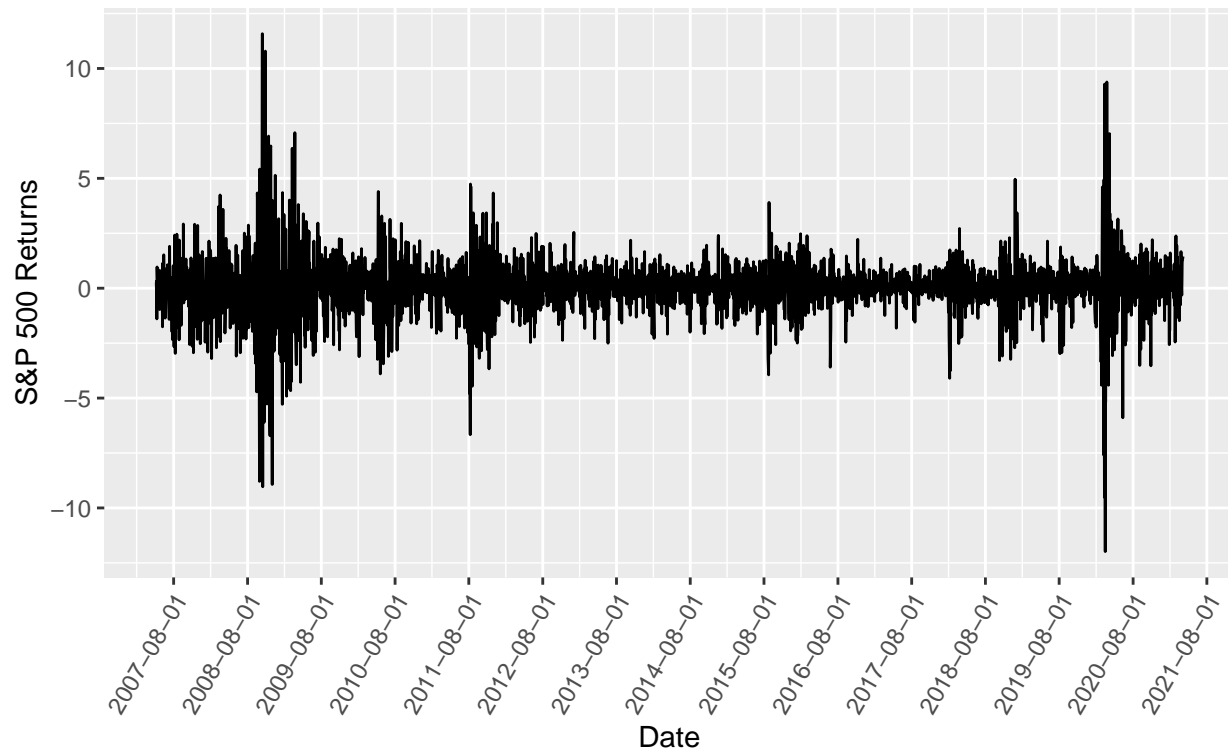
```



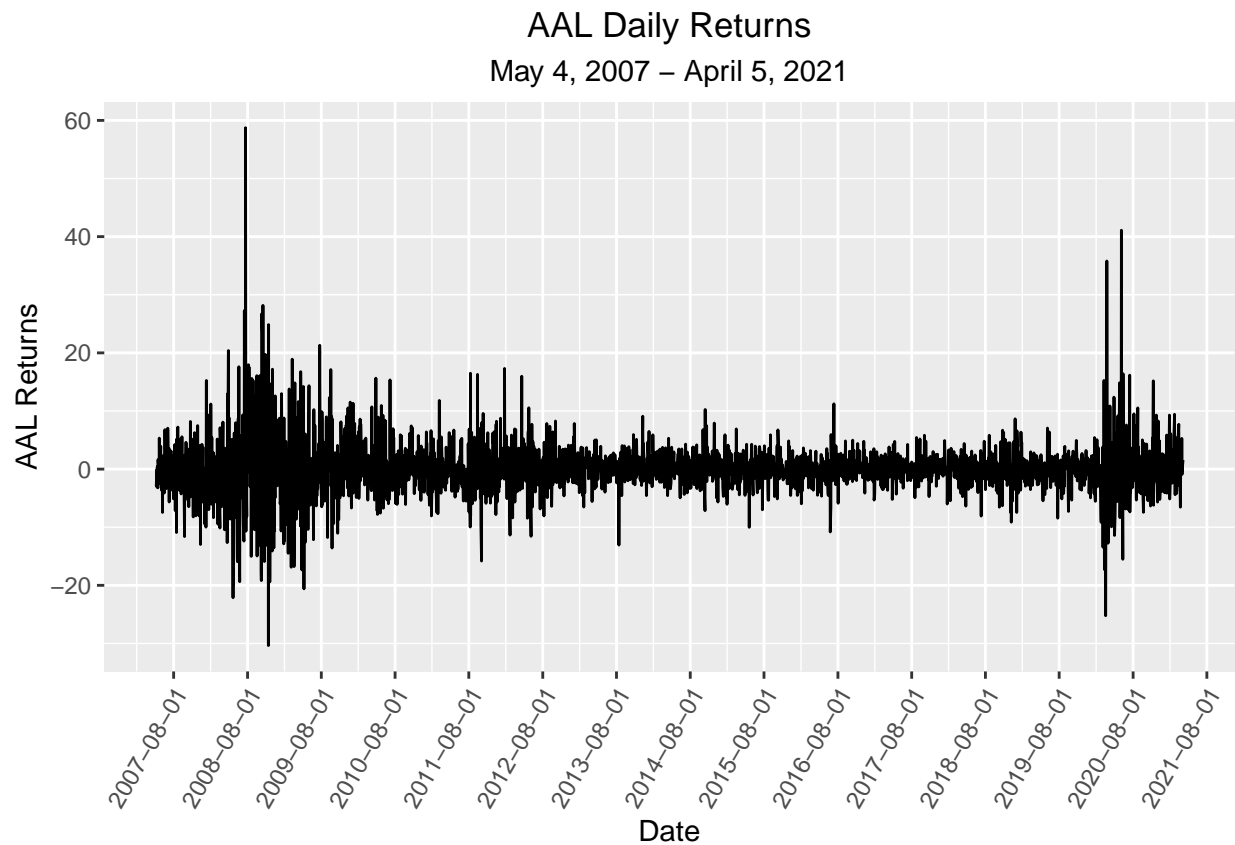
```
# Time series chart for S&P 500 returns
ggplot(sp500_data, aes(x = Date, y = sp500_returns)) +
  geom_line() +
  labs(x = 'Date',
       y = 'S&P 500 Returns',
       title = 'S&P 500 Daily Returns',
       subtitle = 'May 4, 2007 - April 5, 2021') +
  scale_x_date(breaks = "12 month") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 60, hjust = 1))
```

S&P 500 Daily Returns

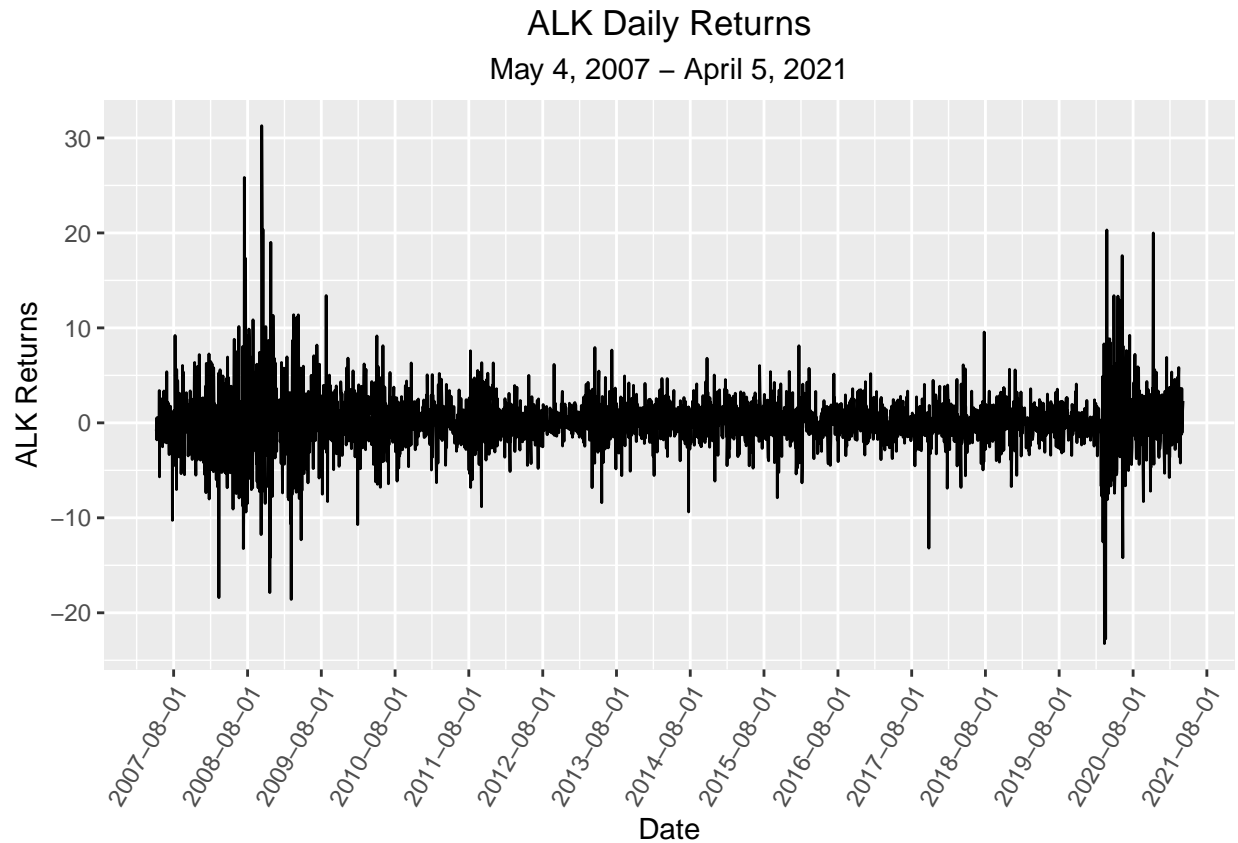
May 4, 2007 – April 5, 2021



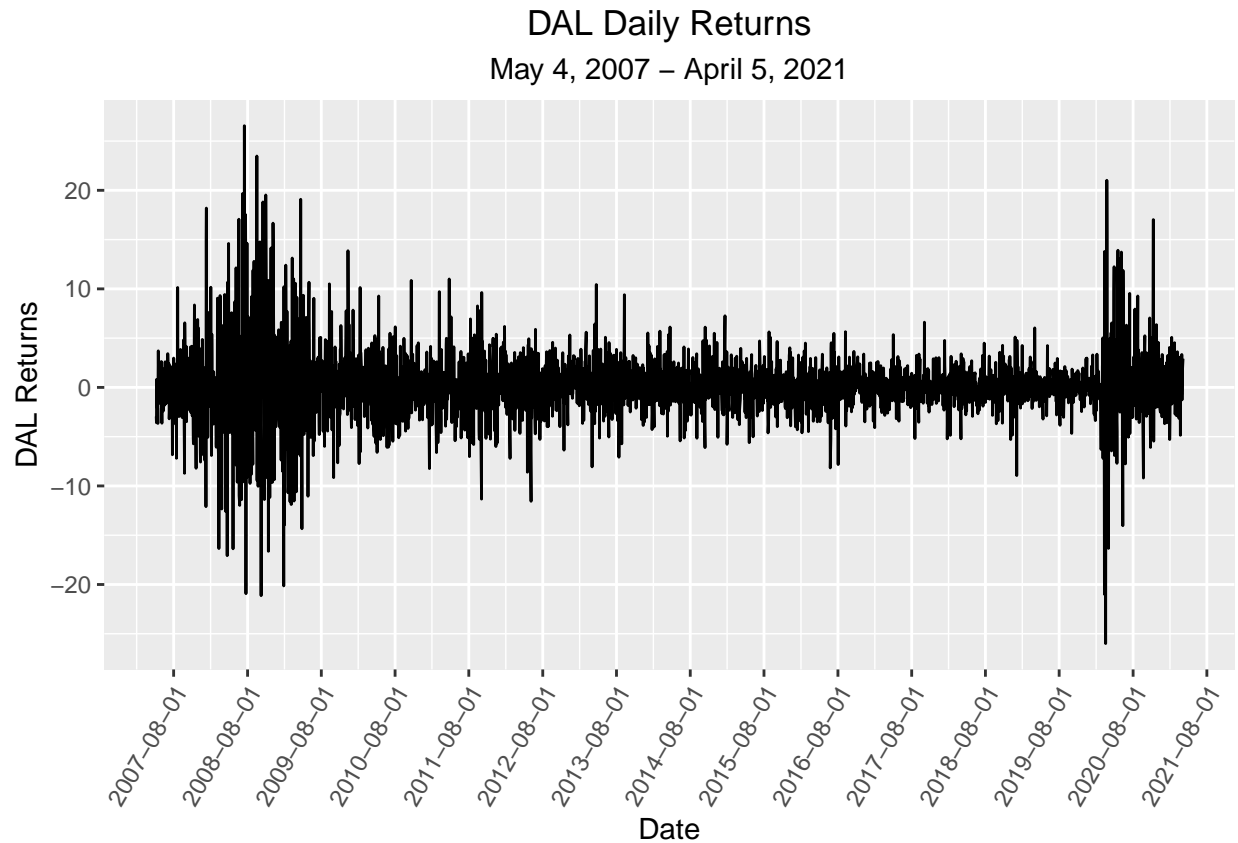
```
# Time series charts for each airline firm
ggplot(return_data, aes(x = Date, y = aal_returns)) +
  geom_line() +
  labs(x = 'Date',
       y = 'AAL Returns',
       title = 'AAL Daily Returns',
       subtitle = 'May 4, 2007 - April 5, 2021') +
  scale_x_date(breaks = "12 month") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 60, hjust = 1))
```



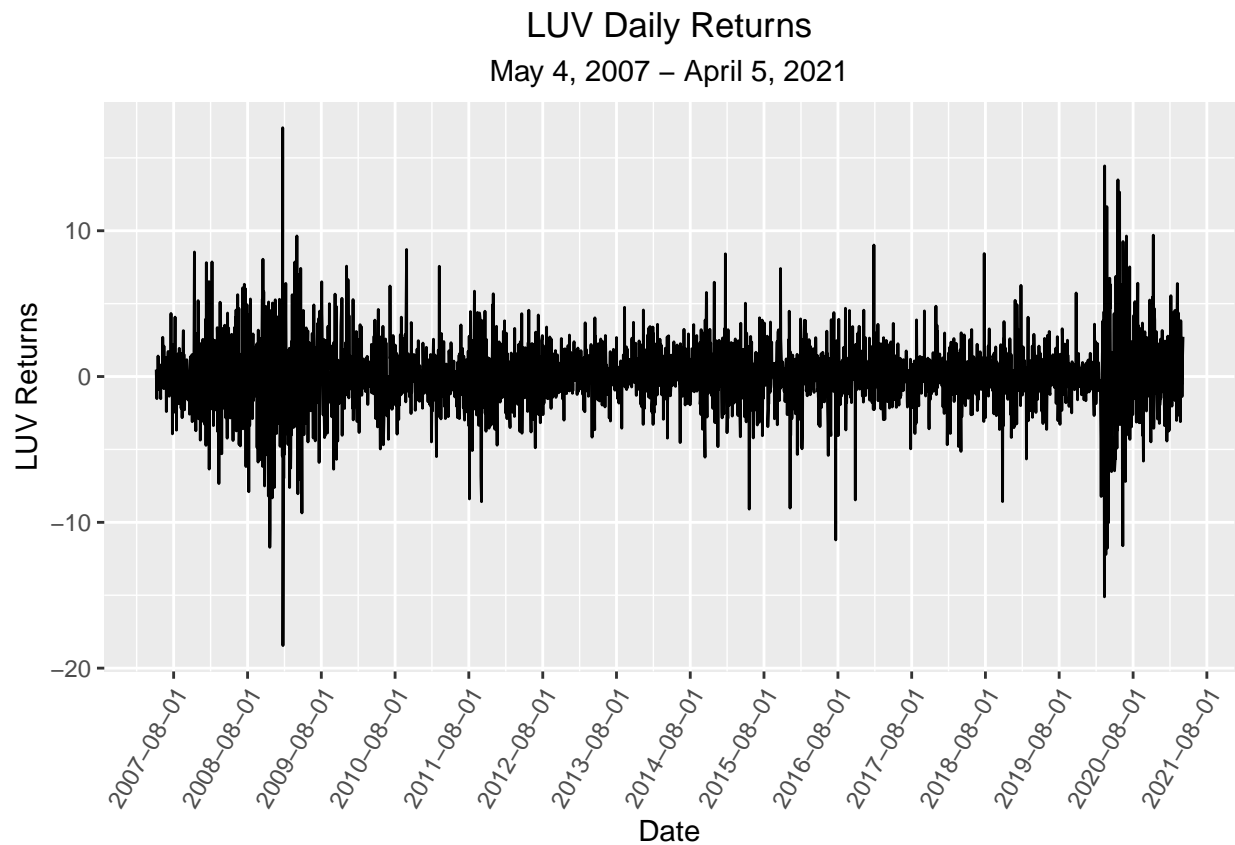
```
ggplot(return_data, aes(x = Date, y = alk_returns)) +  
  geom_line() +  
  labs(x = 'Date',  
       y = 'ALK Returns',  
       title = 'ALK Daily Returns',  
       subtitle = 'May 4, 2007 - April 5, 2021') +  
  scale_x_date(breaks = "12 month") +  
  theme(plot.title = element_text(hjust = 0.5),  
        plot.subtitle = element_text(hjust = 0.5),  
        axis.text.x = element_text(angle = 60, hjust = 1))
```



```
ggplot(return_data, aes(x = Date, y = dal_returns)) +  
  geom_line() +  
  labs(x = 'Date',  
       y = 'DAL Returns',  
       title = 'DAL Daily Returns',  
       subtitle = 'May 4, 2007 - April 5, 2021') +  
  scale_x_date(breaks = "12 month") +  
  theme(plot.title = element_text(hjust = 0.5),  
        plot.subtitle = element_text(hjust = 0.5),  
        axis.text.x = element_text(angle = 60, hjust = 1))
```

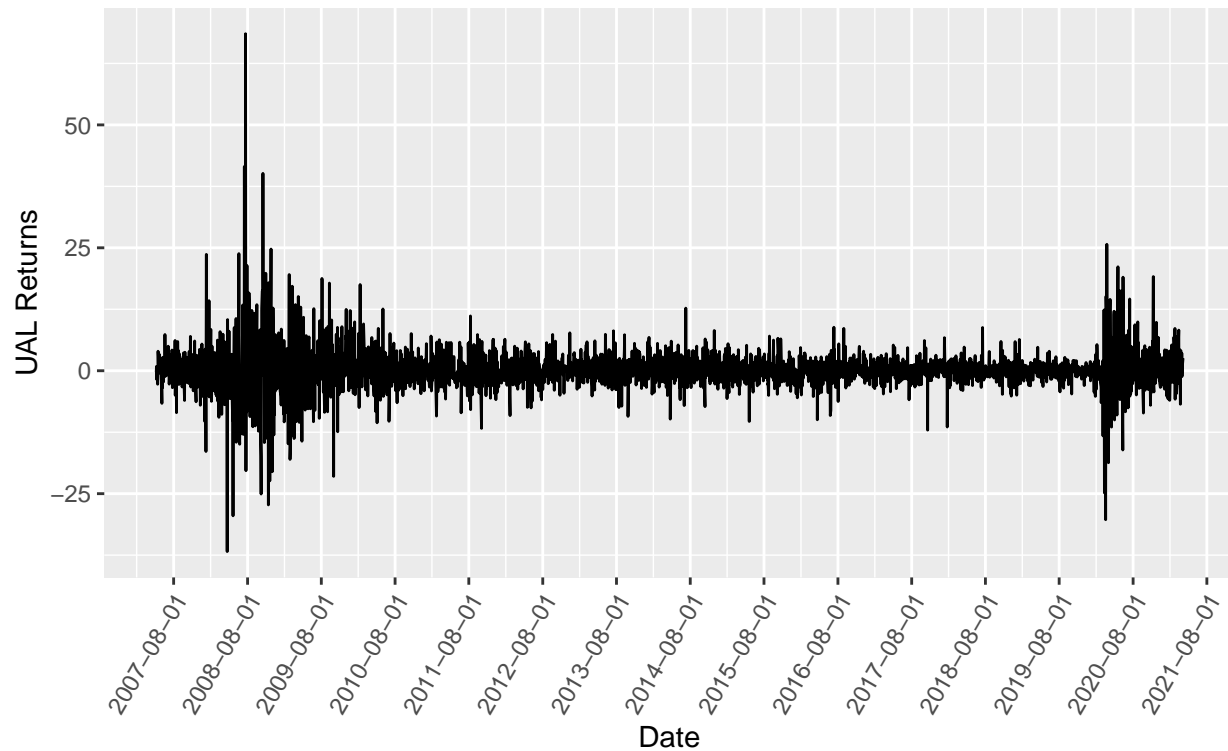
```
ggplot(return_data, aes(x = Date, y = luv_returns)) +  
  geom_line() +  
  labs(x = 'Date',  
       y = 'LUV Returns',  
       title = 'LUV Daily Returns',  
       subtitle = 'May 4, 2007 - April 5, 2021') +  
  scale_x_date(breaks = "12 month") +  
  theme(plot.title = element_text(hjust = 0.5),  
        plot.subtitle = element_text(hjust = 0.5),  
        axis.text.x = element_text(angle = 60, hjust = 1))
```



```
ggplot(return_data, aes(x = Date, y = ual_returns)) +  
  geom_line() +  
  labs(x = 'Date',  
       y = 'UAL Returns',  
       title = 'UAL Daily Returns',  
       subtitle = 'May 4, 2007 - April 5, 2021') +  
  scale_x_date(breaks = "12 month") +  
  theme(plot.title = element_text(hjust = 0.5),  
        plot.subtitle = element_text(hjust = 0.5),  
        axis.text.x = element_text(angle = 60, hjust = 1))
```

UAL Daily Returns

May 4, 2007 – April 5, 2021



```
# Summary table for stock returns
psych::describe(return_data[, -c(1)])
```

```
##          vars      n mean  sd median trimmed  mad    min    max  range  skew
## aal_returns    1 3503 0.09 4.48   0.00    0.00  2.68 -30.36  58.74  89.10  1.20
## alk_returns    2 3503 0.11 2.99   0.11    0.10  1.95 -23.24  31.28  54.52  0.50
## dal_returns    3 3503 0.09 3.43   0.08    0.07  2.11 -25.99  26.55  52.54  0.28
## luv_returns    4 3503 0.07 2.28   0.07    0.08  1.63 -18.45  17.06  35.51 -0.15
## ual_returns    5 3503 0.11 4.39   0.03    0.05  2.41 -36.77  68.54 105.31  1.34
##          kurtosis   se
## aal_returns    16.35 0.08
## alk_returns    12.71 0.05
## dal_returns     8.85 0.06
## luv_returns     6.27 0.04
## ual_returns    27.89 0.07
```

Regression Analysis

First Regression: Linear Model

```
# Dataset containing all variables needed for analysis
final_data <- full_join(return_data, news_sentiment_data, by = c('Date'='date'))
final_data <- full_join(final_data, sp500_data, by = "Date") %>%
```

```
select(Date, aal_returns, alk_returns, dal_returns, luv_returns, ual_returns, news_sentiment, sp500_r
head(final_data)
```

```
##      Date aal_returns alk_returns dal_returns luv_returns ual_returns
## 1 2007-05-04      NA      NA      NA      NA      NA
## 2 2007-05-07 -2.8769301 -0.6997635 -2.9551802  0.4768391 -0.2811318
## 3 2007-05-08 -2.0222358 -0.9395929 -3.5854890 -0.4067690  0.8739767
## 4 2007-05-09 -0.1744079  0.4403795  0.8151396 -1.0211056 -1.4812716
## 5 2007-05-10 -3.0867675 -1.7875191 -0.4547852 -1.5130852 -1.8156006
## 6 2007-05-11  0.2403772  0.5494283 -2.0304583 -0.0697821  0.3178310
## news_sentiment sp500_returns
## 1      0.1345941      NA
## 2      0.1282066      0.2563728
## 3      0.1066936     -0.1165964
## 4      0.1189226      0.3223410
## 5      0.1116518     -1.3956287
## 6      0.1008269      0.9641495
```

```
# Linear Regression
```

```
lm_models <- list(
  aal_results = lm(aal_returns ~ sp500_returns + news_sentiment, data = final_data),
  alk_results = lm(alk_returns ~ sp500_returns + news_sentiment, data = final_data),
  dal_results = lm(dal_returns ~ sp500_returns + news_sentiment, data = final_data),
  luv_results = lm(luv_returns ~ sp500_returns + news_sentiment, data = final_data),
  ual_results = lm(ual_returns ~ sp500_returns + news_sentiment, data = final_data)
)
```

```
lm_results <- lapply(lm_models, tidy)
lm_results_df <- do.call(rbind, lm_results)
lm_results_df
```

```
## # A tibble: 15 x 5
##   term          estimate std.error statistic  p.value
## * <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)    0.0119    0.0711    0.167 8.67e- 1
## 2 sp500_returns  1.62      0.0506   32.0 3.44e-197
## 3 news_sentiment -0.246     0.376   -0.654 5.13e- 1
## 4 (Intercept)    0.0438    0.0454    0.967 3.34e- 1
## 5 sp500_returns  1.23      0.0323   38.1 4.77e-266
## 6 news_sentiment -0.335     0.240   -1.40 1.63e- 1
## 7 (Intercept)    0.0387    0.0534    0.725 4.69e- 1
## 8 sp500_returns  1.32      0.0380   34.7 9.36e-227
## 9 news_sentiment 0.00883    0.282    0.0313 9.75e- 1
## 10 (Intercept)   0.0492    0.0341    1.44 1.49e- 1
## 11 sp500_returns 0.976     0.0242   40.3 4.40e-292
## 12 news_sentiment 0.232     0.180    1.29 1.98e- 1
## 13 (Intercept)   0.0470    0.0704    0.668 5.04e- 1
## 14 sp500_returns 1.54      0.0501   30.7 2.29e-183
## 15 news_sentiment -0.110     0.372   -0.295 7.68e- 1
```

After running a linear regression on the data, there appears to be no significant relationship between any of the airline companies and news sentiment. None of the p-values are close to anything statistically significant and the coefficient estimates are all close to zero. Based on the line graphs from earlier, I suspect this can be partly explained by the data being heteroskedastic.

Second Model: Linear Regression with Robust Standard Errors

```
# Breusch-Pagan Test to test for heteroskedasticity
```

```
bp_tests <- list(  
  aal_bp = bptest(lm_models[[1]]),  
  alk_bp = bptest(lm_models[[2]]),  
  dal_bp = bptest(lm_models[[3]]),  
  luv_bp = bptest(lm_models[[4]]),  
  ual_bp = bptest(lm_models[[5]])  
)  
  
bp_results <- lapply(bp_tests, tidy)  
bp_results_df <- do.call(rbind, bp_results)  
bp_results_df
```

```
## # A tibble: 5 x 4  
##   statistic p.value parameter method  
## *      <dbl>   <dbl>      <dbl> <chr>  
## 1    133.  1.05e-29         2 studentized Breusch-Pagan test  
## 2    125.  6.77e-28         2 studentized Breusch-Pagan test  
## 3    231.  6.03e-51         2 studentized Breusch-Pagan test  
## 4    116.  5.20e-26         2 studentized Breusch-Pagan test  
## 5     89.4 3.85e-20         2 studentized Breusch-Pagan test
```

As the results of the Breusch-Pagan test clearly show, there is heteroskedasticity in each model. To take this into account, I will run the same regression but use robust standard errors instead.

```
# Regression with robust standard errors
```

```
robust_models <- list(  
  ual_robust = lm_robust(ual_returns ~ sp500_returns + news_sentiment, data = final_data),  
  aal_robust = lm_robust(aal_returns ~ sp500_returns + news_sentiment, data = final_data),  
  alk_robust = lm_robust(alk_returns ~ sp500_returns + news_sentiment, data = final_data),  
  dal_robust = lm_robust(dal_returns ~ sp500_returns + news_sentiment, data = final_data),  
  luv_robust = lm_robust(luv_returns ~ sp500_returns + news_sentiment, data = final_data)  
)  
  
robust_results <- lapply(robust_models, tidy)  
robust_results_df <- do.call(rbind, robust_results)  
robust_results_df
```

```
##           term      estimate std.error  statistic      p.value  
## ual_robust.1 (Intercept)  0.046989284 0.04906831  0.95763005 3.383155e-01  
## ual_robust.2 sp500_returns  1.536000167 0.09025273 17.01887838 1.815377e-62  
## ual_robust.3 news_sentiment -0.109621899 0.52722930 -0.20792073 8.353030e-01  
## aal_robust.1 (Intercept)  0.011904527 0.05066672  0.23495751 8.142555e-01  
## aal_robust.2 sp500_returns  1.617075717 0.10229728 15.80761195 2.017701e-54  
## aal_robust.3 news_sentiment -0.245968899 0.52991126 -0.46416998 6.425548e-01  
## alk_robust.1 (Intercept)  0.043846752 0.03486563  1.25759260 2.086230e-01  
## alk_robust.2 sp500_returns  1.229270558 0.06066468 20.26336567 2.044533e-86  
## alk_robust.3 news_sentiment -0.334709758 0.34730558 -0.96373274 3.352465e-01  
## dal_robust.1 (Intercept)  0.038708481 0.03914879  0.98875300 3.228524e-01  
## dal_robust.2 sp500_returns  1.316838113 0.06977544 18.87251742 9.801967e-76
```

```
## dal_robust.3 news_sentiment 0.008832368 0.37881692 0.02331566 9.813998e-01
## luv_robust.1 (Intercept) 0.049195591 0.02883605 1.70604471 8.808847e-02
## luv_robust.2 sp500_returns 0.976202008 0.03877623 25.17526643 1.109200e-128
## luv_robust.3 news_sentiment 0.232052083 0.25689183 0.90330659 3.664254e-01
##
## conf.low conf.high df outcome
## ual_robust.1 -0.04921610 0.1431947 3500 ual_returns
## ual_robust.2 1.35904688 1.7129535 3500 ual_returns
## ual_robust.3 -1.14332981 0.9240860 3500 ual_returns
## aal_robust.1 -0.08743478 0.1112438 3500 aal_returns
## aal_robust.2 1.41650737 1.8176441 3500 aal_returns
## aal_robust.3 -1.28493518 0.7929974 3500 aal_returns
## alk_robust.1 -0.02451226 0.1122058 3500 alk_returns
## alk_robust.2 1.11032884 1.3482123 3500 alk_returns
## alk_robust.3 -1.01565167 0.3462322 3500 alk_returns
## dal_robust.1 -0.03804828 0.1154652 3500 dal_returns
## dal_robust.2 1.18003346 1.4536428 3500 dal_returns
## dal_robust.3 -0.73389199 0.7515567 3500 dal_returns
## luv_robust.1 -0.00734158 0.1057328 3500 luv_returns
## luv_robust.2 0.90017570 1.0522283 3500 luv_returns
## luv_robust.3 -0.27162083 0.7357250 3500 luv_returns
```

Even when taking into account heteroskedasticity, the results are far from definitive. This model runs into the same problems as the first one. There is not strong enough evidence to conclude that the dependent and independent variables have a linear relationship.

Model 3: Taking Into Account Recessions

During the 15-year time span of the data, there were two big economic recessions. The first was the 2008 financial crisis and the second being the 2020 COVID-19 pandemic. Each created volatile economic conditions, with the airline industry being hit in particularly hard during the pandemic. To test how much these events influenced the model, I added two dummy variables. The variable 'recession' has a value of 1 if the date is during the 2008 recession and 0 otherwise. Similarly, the variable 'covid' has a value of 1 if the date is during the pandemic and 0 otherwise.

```
# Adding dummy variables for each event
final_data$recession <- ifelse(final_data$Date > '2007-12-01' & final_data$Date < '2009-06-01', 1, 0)
final_data$covid <- ifelse(final_data$Date > '2020-02-01', 1, 0)

# New regressions
recession_models <- list(
  aal_recession = lm_robust(aal_returns ~ sp500_returns + news_sentiment + covid + recession, data = final_data),
  alk_recession = lm_robust(alk_returns ~ sp500_returns + news_sentiment + covid + recession, data = final_data),
  dal_recession = lm_robust(dal_returns ~ sp500_returns + news_sentiment + covid + recession, data = final_data),
  luv_recession = lm_robust(luv_returns ~ sp500_returns + news_sentiment + covid + recession, data = final_data),
  ual_recession = lm_robust(ual_returns ~ sp500_returns + news_sentiment + covid + recession, data = final_data)
)

recession_results <- lapply(recession_models, tidy)
recession_results_df <- do.call(rbind, recession_results)
recession_results_df
```

```
## term estimate std.error statistic p.value
```

```

## aal_recession.1      (Intercept)  0.02669453 0.05041030  0.52954523  5.964609e-01
## aal_recession.2  sp500_returns  1.61626005 0.10272109 15.73445197  5.964410e-54
## aal_recession.3  news_sentiment -0.41465579 0.47338911 -0.87593014  3.811282e-01
## aal_recession.4      covid -0.14973848 0.31814286 -0.47066426  6.379099e-01
## aal_recession.5      recession -0.12470594 0.41567418 -0.30000887  7.641882e-01
## alk_recession.1      (Intercept)  0.05157173 0.03367374  1.53151186  1.257334e-01
## alk_recession.2  sp500_returns  1.22942863 0.06094978 20.17117380  1.095786e-85
## alk_recession.3  news_sentiment -0.42503468 0.30309132 -1.40233210  1.609048e-01
## alk_recession.4      covid -0.14072476 0.21560894 -0.65268519  5.140021e-01
## alk_recession.5      recession -0.01753528 0.25216544 -0.06953881  9.445647e-01
## dal_recession.1      (Intercept)  0.04455702 0.03785204  1.17713665  2.392211e-01
## dal_recession.2  sp500_returns  1.31698739 0.07006020 18.79794010  3.537725e-75
## dal_recession.3  news_sentiment -0.05966457 0.32748949 -0.18218773  8.554460e-01
## dal_recession.4      covid -0.10971080 0.22356052 -0.49074317  6.236389e-01
## dal_recession.5      recession -0.01086305 0.32512051 -0.03341237  9.733477e-01
## luv_recession.1      (Intercept)  0.04882496 0.02891701  1.68845137  9.141375e-02
## luv_recession.2  sp500_returns  0.97591592 0.03880180 25.15130456 1.874138e-128
## luv_recession.3  news_sentiment  0.23744320 0.24550972  0.96714376  3.335390e-01
## luv_recession.4      covid  0.03655904 0.16802918  0.21757555  8.277725e-01
## luv_recession.5      recession -0.02185464 0.15875595 -0.13766184  8.905156e-01
## ual_recession.1      (Intercept)  0.06179015 0.04630319  1.33446856  1.821372e-01
## ual_recession.2  sp500_returns  1.53544162 0.09045174 16.97525746  3.620504e-62
## ual_recession.3  news_sentiment -0.27941130 0.44096580 -0.63363486  5.263606e-01
## ual_recession.4      covid -0.17743152 0.29477930 -0.60191310  5.472710e-01
## ual_recession.5      recession -0.10379525 0.44281821 -0.23439697  8.146905e-01
##
##          conf.low conf.high  df      outcome
## aal_recession.1 -0.07214203 0.1255311 3498 aal_returns
## aal_recession.2  1.41486073 1.8176594 3498 aal_returns
## aal_recession.3 -1.34280255 0.5134910 3498 aal_returns
## aal_recession.4 -0.77350286 0.4740259 3498 aal_returns
## aal_recession.5 -0.93969437 0.6902825 3498 aal_returns
## alk_recession.1 -0.01445043 0.1175939 3498 alk_returns
## alk_recession.2  1.10992790 1.3489294 3498 alk_returns
## alk_recession.3 -1.01928836 0.1692190 3498 alk_returns
## alk_recession.4 -0.56345678 0.2820073 3498 alk_returns
## alk_recession.5 -0.51194154 0.4768710 3498 alk_returns
## dal_recession.1 -0.02965729 0.1187713 3498 dal_returns
## dal_recession.2  1.17962440 1.4543504 3498 dal_returns
## dal_recession.3 -0.70175436 0.5824252 3498 dal_returns
## dal_recession.4 -0.54803304 0.3286114 3498 dal_returns
## dal_recession.5 -0.64830809 0.6265820 3498 dal_returns
## luv_recession.1 -0.00787095 0.1055209 3498 luv_returns
## luv_recession.2  0.89983947 1.0519924 3498 luv_returns
## luv_recession.3 -0.24391357 0.7188000 3498 luv_returns
## luv_recession.4 -0.29288610 0.3660042 3498 luv_returns
## luv_recession.5 -0.33311828 0.2894090 3498 luv_returns
## ual_recession.1 -0.02899385 0.1525742 3498 ual_returns
## ual_recession.2  1.35809810 1.7127851 3498 ual_returns
## ual_recession.3 -1.14398755 0.5851649 3498 ual_returns
## ual_recession.4 -0.75538831 0.4005253 3498 ual_returns
## ual_recession.5 -0.97200341 0.7644129 3498 ual_returns

```

Like the other two models, no statistically significant conclusions can be drawn from the models. Although there is slight improvement compared to the first, it is still far from being a good model.

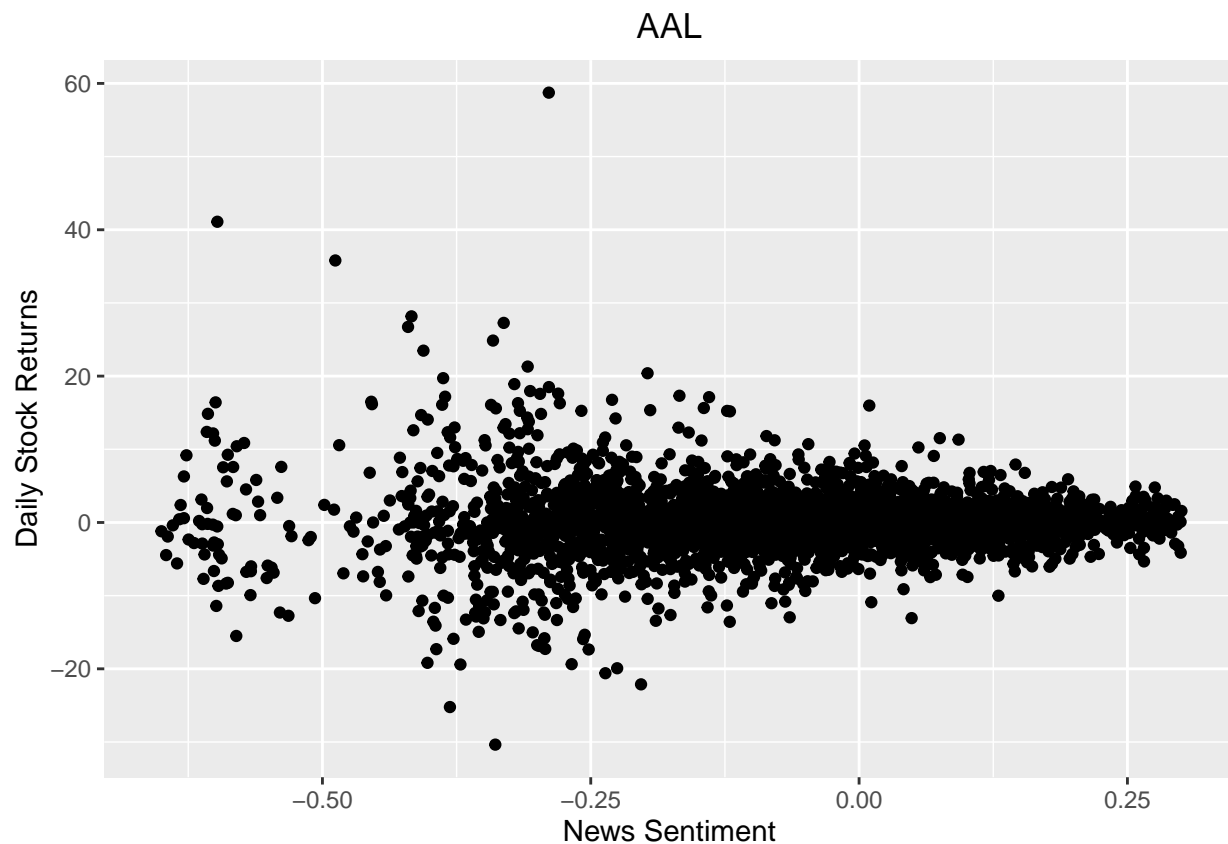
Conclusion

After running three separate regressions, there is not enough evidence to conclude a linear relationship between the daily stock returns of airline companies and news sentiment of the economy. This is likely due to a multitude of factors. The model likely suffers from omitted variable bias, where confounding variables are excluded from the model. A linear relationship may also not be the best way to explain their relationship, in the future I would like to test other regressions such as logarithmic or exponential.

Appendix

Appendix 1: News sentiment regressed against stock returns

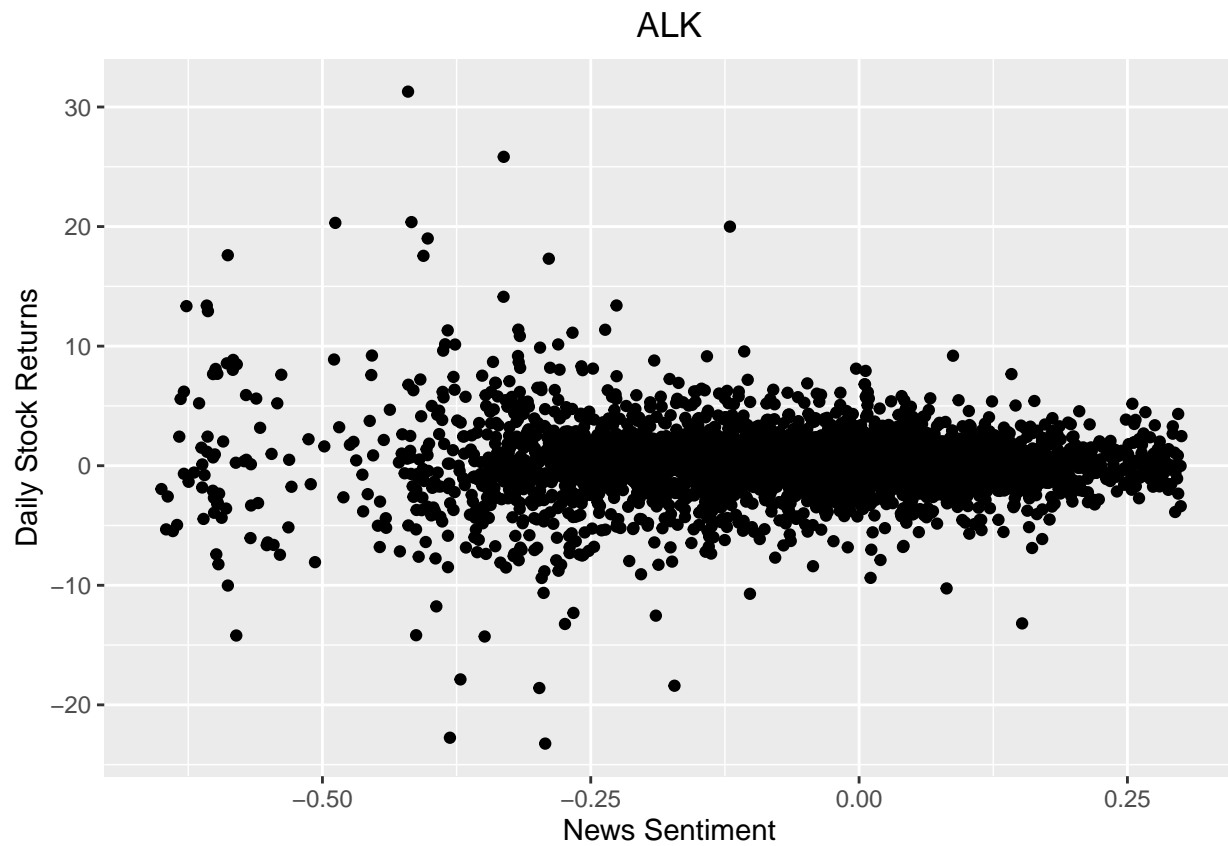
```
# AAL
ggplot(data = final_data, aes(x = news_sentiment, y = aal_returns))+
  geom_point() +
  labs(x = "News Sentiment",
       y = "Daily Stock Returns",
       title = "AAL") +
  theme(plot.title = element_text(hjust = 0.5))
```



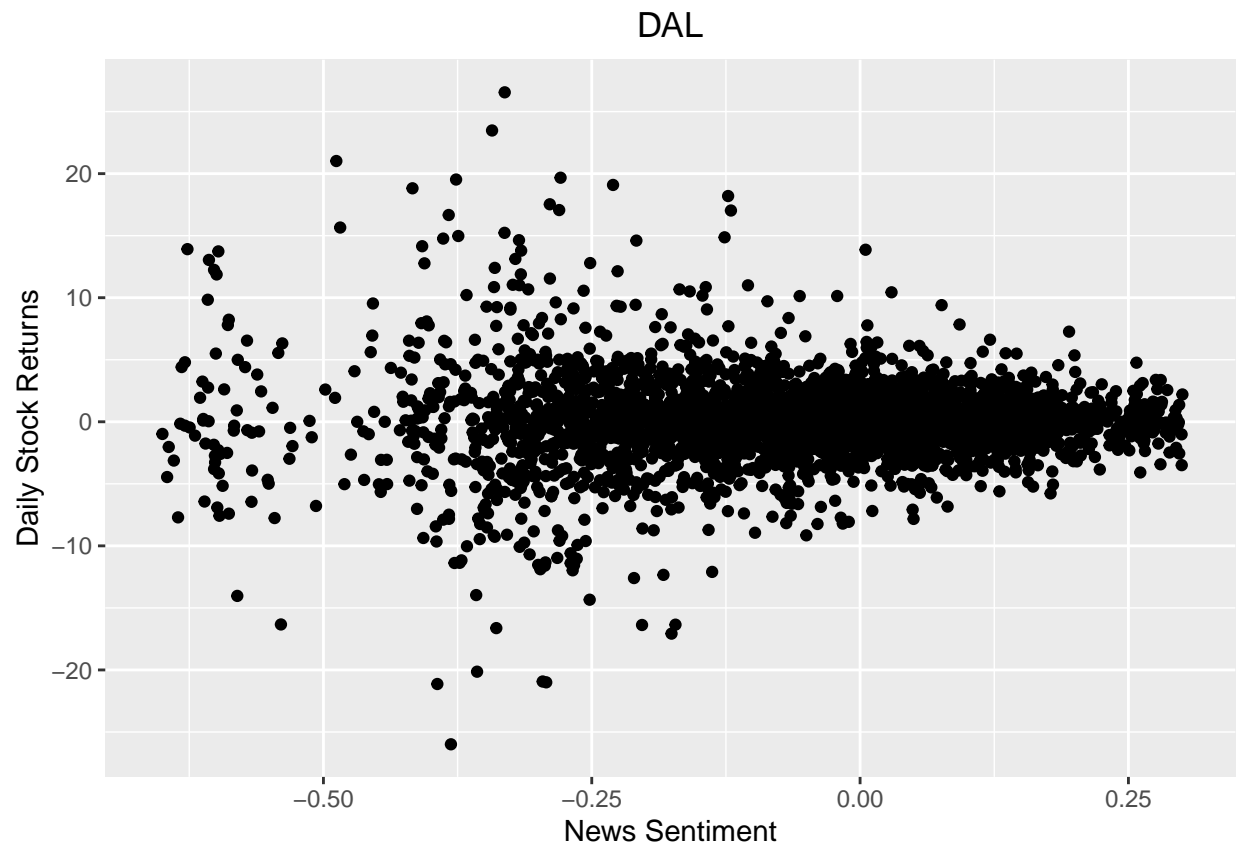
```
# ALK
ggplot(data = final_data, aes(x = news_sentiment, y = alk_returns)) +
```



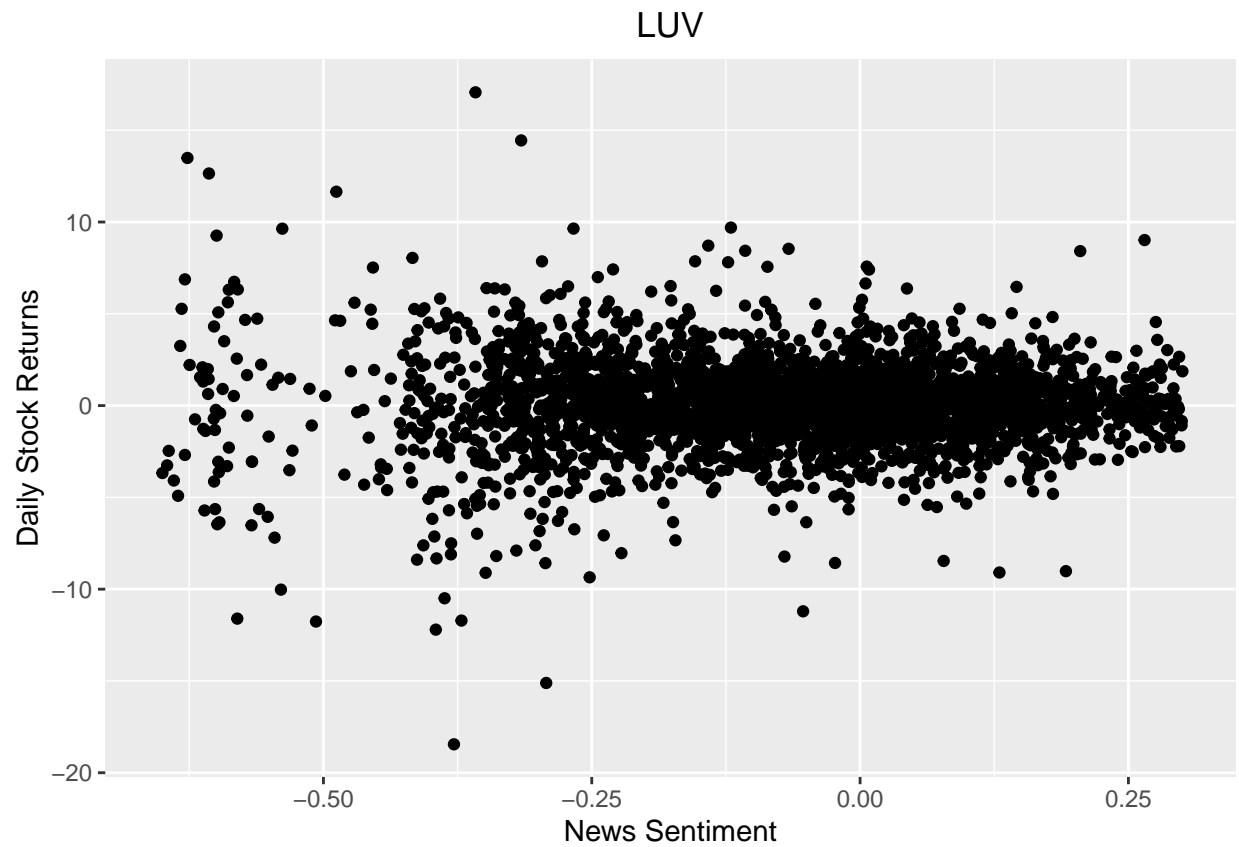
```
geom_point() +
  labs(x = "News Sentiment",
       y = "Daily Stock Returns",
       title = "ALK") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
# DAL
ggplot(data = final_data, aes(x = news_sentiment, y = dal_returns)) +
  geom_point() +
  labs(x = "News Sentiment",
       y = "Daily Stock Returns",
       title = "DAL") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
# LUV
ggplot(data = final_data, aes(x = news_sentiment, y = luv_returns)) +
  geom_point() +
  labs(x = "News Sentiment",
       y = "Daily Stock Returns",
       title = "LUV") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
# UAL
ggplot(data = final_data, aes(x = news_sentiment, y = ual_returns)) +
  geom_point() +
  labs(x = "News Sentiment",
       y = "Daily Stock Returns",
       title = "UAL") +
  theme(plot.title = element_text(hjust = 0.5))
```

