The Effect of Elon Musk's Tweets on Tesla's Stock Market Performance

Mira Heinemann and Esra Tonleu

Fresenius University of Applied Science

Data Science and Data Analytics (WS 2025/26)

Prof. Dr. Stephan Huber and Benjamin Groß

2026-02-02

Author Note

Correspondence concerning this article should be addressed to Mira Heinemann, Email: heinemann.mira@stud-hs.fresenius.de

The Effect of Elon Musk's Tweets on Tesla's Stock Market Performance

Word count: 9806

1 Introduction

In today's world, social media has evolved into a primary means of communicating information in financial markets allowing CEOs to communicate directly and instantly with their investors. The way in which social media facilitates communication between companies and investors is unique from more traditional methods of communication used by companies. As such, understanding how quickly and in what manner markets respond to such forms of communication is critical. This study explores how Elon Musk's tweets related to Tesla affect short-term stock price movements utilizing high-frequency intraday data and an event study design to determine how stock prices and trading volumes react to tweets based on sentiment.

1.1 Social Media and Information Flow in Financial Markets

Social media has dramatically altered the development, sharing, and utilization of information in financial markets. Social media apps like Twitter enable corporate executives, companies, and influential figures to share information directly with investors, analysts, and the public in real-time. Standardized forms of information flow (i.e. earnings calls, regulatory filings, press releases, etc.) provide formalized disclosure routes; however, social media provides a more informal, unregulated, and unpredictable route of information flow. This unpredictability creates the potential for direct impact on the stock price of companies (Garg and Tiwari (2021)).

The immediacy associated with social media has several implications regarding market efficiency. Studies have indicated that financial markets respond rapidly to new information and changes in investor attention (Cutler et al. (1989)). Additionally, media coverage and the tone of written language have been found to affect short-term stock price movements and trading activity with evidence suggesting that investor sentiment affects pricing in addition to the actual content (Tetlock (2007)). The effects of social media are amplified in terms of decreasing the costs and increasing the speed of information flow.

Additionally, social media platforms increase investor attention. Engelberg and Parsons (Engelberg and Parsons (2011)) find that as there is more exposure to media, trading activity increases despite the fact that the content of the information is limited. Additionally, previous studies have identified that Twitter messages have generated abnormal trading volume and short-term return movements for publicly traded companies (Sprenger et al. (2014)).

These findings suggest that social media can act as an information conduit and attract market attention creating rapid market reactions to the information.

1.2 Why Elon Musk and Tesla Provide a Unique Setting

Elon Musk is an excellent illustration of a corporate executive who uses social media as a direct and public tool of communication. Musk is the CEO of Tesla and regularly posts information related to Tesla and various company related topics via Twitter including but not limited to product announcements, production updates, business decisions, economic conditions, and technological developments. Musk's information messages on Twitter are not similar to typical corporate disclosures and can often be released outside of pre-planned communication times allowing information to reach investors immediately (Sprenger et al. (2014)).

Due to Musk's large and high profile presence in the media and the significant influence Tesla has in the financial markets, Musk's social media activity generates a significant amount of attention from both private and institutional investors (Yahoo Finance (2025)). Tesla is among the most heavily traded equities in the U.S. stock market with a large number of retail investors holding shares and numerous media outlets reporting on the company's current news (Yahoo Finance (2025)). This combination of Musk's high public profile, the frequent posting of Tesla-related information, and the liquidity of Tesla's stock make Tesla a very attractive candidate to use in an event study to examine the short-term, intraday market impacts of Musk's communication with investors.

1.3 The Timing Challenge in Measuring Market Reactions

One of the largest problems in analyzing the impact of social media-based information from executives on stock prices is determining the correct timing of the information being provided relative to the opening and closing of the stock exchanges. Musk's tweets may be posted prior to, during, or after-market hours and stock prices can only adjust to the information contained in the tweet when the stock exchange is open. Thus, the timing of a tweet is a critical factor in determining whether and when a market reaction can be observed.

Analyzing daily stock market data can obscure the rapid price movements that occur in response to information immediately after it is released. For instance, a tweet can trigger a rapid price movement within minutes that is reversed by the end of the trading day, leaving behind little or no evidence of the price movement in the daily returns. This is especially relevant to social media-based information dissemination, as it occurs instantaneously and can elicit an investor reaction nearly simultaneously.

To address the issue of timing, the analysis in this study focuses on the short periods of time around the date and time of each tweet and limits analysis to tweets posted during normal trading hours. This approach allows us to monitor stock price and trading activity immediately preceding and immediately following the dissemination of information contained in a tweet, resulting in a more detailed and accurate picture of the short-term market reactions than would be possible using lower frequency data.

1.4 Why an Intraday, Trading-Time Framework Is Needed

As previously described, we utilize an intraday event study framework driven by the timing of trades. Using 5-minute intraday stock price and trading volume data enables us to monitor the behavior of markets just prior to and subsequent to the posting of a tweet, enabling us to identify short-term reactions that would likely go unnoticed in the daily data.

An intraday framework is particularly relevant as tweets are published at specific points in the trading day, and the timing of the information release may be critical to the market reaction. To accurately capture market reactions, we need to calibrate the stock market data to the exact timing of every tweet, so that event windows accurately capture moments when trading is occurring and cannot be influenced by off-hours or non-trading periods.

By standardizing reactions to occur over equal periods of time, price and volume movements can be compared more consistently across events. Additionally, intraday data provide the ability to observe immediate market reactions and short-term dynamics that are lost when using daily stock market data.

1.5 Research Question and Contribution

Based on the rationale presented in the sections above, the objective of this research project is to investigate the short-term stock price responses to Elon Musk's social media communication concerning Tesla-related tweets. The central research question posed in this project is: **How does Tesla's stock price move in the short-term in response to Elon Musk's Tesla-related tweets, and do these patterns vary depending on the sentiment of the tweets?** By examining the intraday market responses to Musk's tweets, the research aims to capture the short-run price movements and trading volume that occur immediately after information is released and may not be observable in lower frequency data.

This study contributes to the existing literature in two ways. First, it employs a well established intraday event study framework that takes into account the specific timing of

Musk's tweets and the trading hours of the U.S. stock market. By aligning the timing of Musk's tweets with high-frequency stock price and trading volume data, this study addresses the timing issues associated with the use of social media to facilitate information flows. This methodology is beneficial as it allows us to more accurately identify short-term market reactions to Musk's tweets as opposed to using daily stock price data.

Secondly, the project combines illustrative examples of single tweets with aggregate results of a larger set of tweets. This dual perspective allows us to examine both the individual-level market reactions to individual tweets as well as the overall trends across different sentiment categories. During the semester project, this methodology provides a systematic and comprehensive evaluation of the structured and intuitive ways in which executive social media communication drives short-term stock price movements.

2 Background & Related Literature

This section examines how financial markets respond to news events and how investor attention impacts daily price movements. It compiles research results on the increasing effects of social media on investor sentiment and company performance; also considering how this relates to high-profile individuals like Elon Musk. Finally, it emphasizes the methodological need for the use of intraday data and event time alignment when capturing financial markets responses to digital communication by CEOs as they occur in real-time.

2.1 Market Reactions to News and Investor Attention

The rapid response of financial markets to the dissemination of publicly available data is typically triggered by news releases, which influence both the timing and magnitude of market participants' reactions to news releases. Market participants do not simply respond to the informational content of news, but to its tone and perceived visibility. Research has shown that market participants have a tendency to interpret negative news sentiment as a signal of impending downward price pressure over the short run, however, this downward price pressure can be partially reversed as the market absorbs and evaluates the newly released information (Tetlock (2007)).

The relationship between investor attention and news releases is a mediating factor regarding how news releases affect the prices of stocks. The degree of media coverage can increase visibility and trading activity regardless of the newness of the information contained in the news release (Engelberg and Parsons (2011)). Notably, stocks that are exposed to greater media coverage, or are the subject of public scrutiny due to either sensational events or celebrity status of the company's personnel, tend to exhibit greater short-term volatility and abnormal returns.

There is empirical support indicating that market participants do not respond uniformly to news releases. In general, households and other small investors are more reactive to both the volume and sentiment of news releases than large institutional investors, who generally place greater emphasis on fundamental characteristics (Bollen et al. (2011)). This disparity in market participant reaction to news releases is critical when a single influential individual, such as a CEO, drives disproportionate attention to an issue through direct communication channels such as Twitter.

2.2 Social Media and Financial Markets

The rise of social media has dramatically changed how people get information and use it to make investment decisions about companies. A number of social media platforms provide opportunities for individuals, including high profile individuals, such as CEOs of companies, to express opinions about publicly traded companies they manage. Social media platforms allow individuals to express their opinions in real-time without any interpretation by traditional media sources and directly communicate with retail investors. Sentiment expressed in a tweet regarding an individual company, especially when that sentiment is conveyed strongly by an influential person, is successful at getting the attention of investors and impacting the price of an individual company's stock in the short run (Shah et al. (2019)).

Empirical studies have shown that there is predictive value in the sentiment expressed in social media messages concerning stock price movements in the future. For instance, researchers found out that changes in Twitter sentiment often occur before movements in stock prices, suggesting that collective emotions expressed on social media may influence market behavior rather than simply reflect it. When examining hourly data over a long-term horizon (multi-months), researchers found evidence of an "information surplus" that is, social media messages reflected market sentiment prior to the price movement of the stock, therefore, there is a statistically significant lead time before the sentiment in social media moves prices. Further, the lead-time effect was more pronounced when using emotional tone in social media posts versus the number of posts, indicating it was the content of social media messages rather than their frequency that was indicative of anticipated short-term market reactions (Zheludev et al. (2014)).

2.3 Musk's Influence

This research utilizes tweet data from Elon Musk's official account on X (formerly Twitter). Musk is one of the most well-known entrepreneurs today; however, since 2022, Musk has also acquired ownership of the platform formerly known as Twitter. This acquisition has increased the number of eyes on Musk's tweets and their impact; in addition, Musk's tweets have been shown to be increasing at a much faster rate than before as seen in Figure 1 with tweets regarding Tesla, Free Speech, and US Politics trending higher during and after the time of the Twitter acquisition.

Figure 1

Screenshot from The Insider showing all of Elon Musk's tweets on Twitter (X)

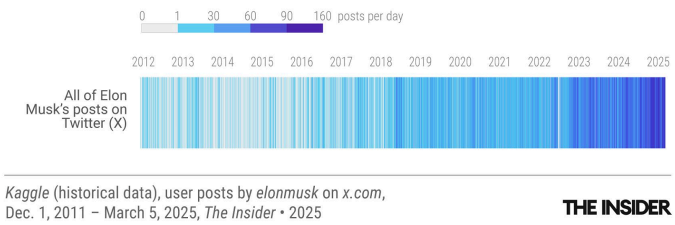


Image reproduced from The Insider for illustrative purposes. All rights belong to the original publisher.

The before mentioned lead-time effect is amplified in highly visible companies, such as Tesla, because the CEO of Tesla, Elon Musk, has a very large and frequently provocative social media presence. Musk's tweets frequently result in high engagement levels and are covered by many traditional media outlets, thereby increasing the potential reach and impact of his tweets (Kwon et al. (2022);Bushee et al. (2010)). Further, even if Musk's tweets do not present any new information regarding Tesla, the tone and timing of those tweets can result in rapid changes in the behavior of traders, particularly among retail investors, whose investment decisions are more likely to be driven by sentiment than analysis (Shah et al. (2019); Bollen et al. (2011)).

2.4 Intraday Event Studies and Timing Challenges

The timing of an information release is a major issue in empirical research of how markets react to new information. Daily data may be sufficient to capture trends and other gradual changes in stock prices over time; however, daily data may not have the resolution to capture rapid price movements resulting from high-frequency events such as social media posts.

Prior research suggests that most market adjustments occur within a minute of an announcement (Cutler et al. (1989)), therefore, using daily data would not allow researchers to capture the causal market reaction to an announcement. Therefore, prior research has been developing intraday event study approaches to utilize high-frequency stock price data and the exact timing of announcements of interest to analyse investor behavior and measure the immediate market reaction to the announcement of new information. The utilization of intraday data is especially important for analyzing communications from executives through social media platforms such as Twitter. Tweets by high profile individuals such as Elon Musk can lead to rapid changes in stock prices. Bushee et al. demonstrate that intraday event studies are best suited to measure the effect of communications by isolating the communications from the rest of the market activity by closely correlating the tweet with the price movement and reducing the risk of contamination from unrelated news (Bushee et al. (2010)).

3 Data

This section provides an overview of the two primary data sources that were analyzed within the paper. Section 3.1 explains how Elon Musk's tweets have been gathered along with metadata, timestamps and engagement, which is necessary when analyzing the content of his tweets and aligning them to specific events. Section 3.2 provides information about the intraday Tesla stock price data; by using it as a high-frequency event-study data source, to measure the short-term effects of Musk's tweets on the stock price.

3.1 Tweets from Platform Twitter (X)

Our original intention was to collect Musk's tweets through the use of the official Twitter API. However, due to recent policy changes in how data can be accessed and the cost of accessing data, using the Twitter API has become outside of the means of a student project. Therefore, we used a publically accessible dataset obtained from Kaggle entitled "Elon Musk Tweets 2010-2025." This dataset is a collection of Musk's tweets, in addition to several other tweet collections previously made available on Kaggle, and provides comprehensive coverage from Musk's first tweet until April 13, 2025.

In addition to having the text of each tweet and the exact timestamp that each tweet was posted, the dataset contains substantial amounts of metadata associated with each tweet. Specifically, it includes the retweet count, like count, quote tweet count, view count, reply count, bookmark count, as well as flags indicating if a tweet is a reply, retweet, or pinned.

3.2 Tesla Stock Market Data

In order to identify the instant effect of Elon Musk's tweets on Tesla's stock price, this paper makes use of daily stock price data, which allows us to match the time that a market reaction occurs with the exact time of when a social media event occurred. The most suitable form of data to use in an event-study are those at very high-frequency, ideally at a 1-minute interval, because these types of data show how much a stock's price and how many shares were traded at every moment after an announcement was made. Since Twitter announcements can be sent out to tens of thousands of people in a matter of seconds, the ability to track changes in a stock's price over a few seconds is especially valuable. However, the vast majority of sources that provide 1-minute interval intraday data do not make their data available for public consumption; therefore, obtaining historical 1-minute interval intraday data would require paying a subscription fee to a commercial data vendor, which limits the use of this resource in a student research environment.

As an alternative to using historical 1-minute interval intraday data, we utilize a publicly available dataset from Kaggle entitled "Tesla Intraday Stock Market Data". This dataset provides Tesla's stock price at 5-minute intervals between January 2017 and June 2020. While the dataset spans a broader time period, the empirical analysis in this study uses the period from January to March 2020. Although the data in this source is less than ideal since it does not have a 1-minute interval, the 5-minute interval is sufficient to be able to capture any short-term reactions by the market to an announcement made via social media that would occur in a typical event window used in financial microstructure research (i.e., ±15 minutes around the time of the announcement). The dataset also contains other important metrics related to trading including the opening price, highest price, lowest price, closing price, and number of shares traded for each 5-minute interval, which enables us to calculate returns, volatility, and abnormal trading activity occurring during the post-announcement period.

4 Methodology

This part details the empirical methodological design we employ in order to assess the stock market's intraday reaction to Elon Musk's Tesla-related tweets. Our methodology follows a sequential pipeline.

The study examines data from June 2019 to May 2020. This time frame is long enough to include a sufficiently large number of tweets related to Tesla that can be used for our analysis.

First, we cleanse our tweets and classify them by content, timing, and sentiment. We next process the high-frequency intraday stock price and volume data for Tesla and align these with the Twitter timelines. We then construct intraday event windows around each eligible tweet. Finally, we calculate the intraday event-study statistics to assess the price and volume reactions by the sentiment categories.

4.1 Tweet Preprocessing (Script 01)

Our empirical analysis involves creating a clean dataset of Elon Musk's tweets. We choose to retain only the original tweets and exclude retweets, replies, and quote-tweets in order to represent Musk's communications directly. This choice to limit our analyses to original tweets allows us to more directly attribute potential market reactions and classify sentiments.

```
tweets_filtered <- tweets_raw %>%
  filter(
    isRetweet != "TRUE",
    isReply   != "TRUE",
    isQuote   != "TRUE"
  )
```

We parse the tweet timestamps and convert them from UTC to U.S. Eastern Time (ET) in order to synchronize them with the US stock markets' trading hours. Since we are analyzing the period from June 2019 to May 2020 for the event-study, we use the tweets with U.S. Eastern Time (ET) in order to accurately align the tweets with the intraday periods. We identify tweets that only have links in them and flag those as tweets with only hyperlinks (which could potentially have differing informational content).

Using a list of pre-defined keywords that reflect Tesla and its related activities, we classify each tweet as being about Tesla or not.

```
tesla_keywords <- c(
  "tesla", "tsla", "model s", "model 3", "model x", "model y",
  "cybertruck", "roadster", "autopilot", "fsd", "gigafactory",
  "powerwall", "megapack", "supercharger"
```

```
)
```

For the primary analysis, we restrict ourselves to only the tweets that are classified as Tesla-related. Finally, each tweet is categorized by whether it is published during the normal US stock markets hours, on a non-trading day but still during market hours, or on a weekend. The final dataset is used as the foundation for our sentiment analysis and subsequent intraday event-study framework.

4.2 Sentiment Classification (Script 02)

During the second step of the preprocessing stage, we add sentiment classification to the Tesla-related tweets previously created in Step 01. We load musk_tesla_tweets_clean.csv and process tweet identifiers as simple character strings to remove any formatting errors. Because the sentiment models used in this analysis rely on actual linguistic input, we drop from the dataset the tweets that were flagged as only having links in the prior step because they cannot express either a positive or negative sentiment.

Sentiment scores are generated using the sentiment.ai package; this package produces a continuous sentiment score for each tweet, based upon the text of the tweet. Initially, we examined various approaches to select the most suitable approach for analyzing tweets before selecting a sentiment model. One of the initial approaches that we examined was a simple lexicon-based method (tidytext); however these do not have the ability to adequately account for the effects of context, sarcasm, or irony that quite often exist in Musk's communications. Therefore, we also investigated more complex and contextualized sentiment models, including BERT, RoBERTa, and other large language model based classifiers. These more complex models are generally accessed via Python-based APIs, and are therefore difficult to integrate into an R-based workflow. As a practical and technically consistent alternative, we adopted the sentiment.ai package, which provides deep-learning-based sentiment embeddings directly within R. Although this required configuring a Python–TensorFlow backend, the resulting setup enables a more accurate and context-sensitive sentiment determination than traditional lexicon-based methods. Each tweet receives a sentiment score on a scale from -1 (the most negative) to $+1$ (the most positive); sentiment scores that are close to zero represent neutral or ambiguous language.

To make the continuous sentiment scores easier to compare in the event-study analysis, we create discrete sentiment categories. The tweets with scores <-0.3 are labeled as negative, the tweets with scores >+0.3 are labeled as positive, and the rest of the tweets are labeled as neutral.

```
# --------------------------------------------------------
# Compute sentiment scores
# --------------------------------------------------------
tweets_with_sentiment <- tweets_for_sentiment %>%
  mutate(
    sentiment_score = sentiment_score(fullText),
    sentiment_label = case_when(
      sentiment_score < -0.3 ~ "Negative",
      sentiment_score >  0.3 ~ "Positive",
      TRUE                   ~ "Neutral"
    )
  )
```

By using symmetric thresholds close to zero, we avoid noise associated with weak or ambiguous sentiment, and prevent misclassifying tweets whose language does not clearly show a positive or negative sentiment. We save the additional dataset that includes both continuous sentiment scores and categorical sentiment labels to be used in the intraday event-study.

4.3 Intraday Stock Data Processing (Scripts 03 + 05)

We analyze five-minute intraday stock price and trading volume data for Tesla to examine short-term market reactions. The raw intraday stock price data is formatted as open, high, low, close prices and trading volume for each five-minute period. The stock price timestamps are parsed and retained in U.S. Eastern Time (ET), thereby synchronizing them with the tweet timestamps and enabling accurate intraday alignments.

As with the tweets, we only use the stock price data for the period of June 2019 to May 2020 for the event-study, and format the data by trading day and intraday time. The five-minute log returns are calculated using consecutive closing prices within each trading day to measure the short-horizon price changes excluding overnight returns.

```
mutate(
  close_lag = lag(Close),
  ret_5m_log = log(Close) - log(close_lag)
)
```

The trading volume is scaled by the same-day average volume to allow for meaningful comparisons of trading volumes over trading days with different base levels of trading activity. The resulting dataset includes prices, returns, and the normalized trading volume measures for each interval, and is saved to be merged with the tweet data for the subsequent intraday event-study analysis.

4.4 Event Window Creation (Script 06)

The intraday event-study data is created by comparing the tweets with the intraday stock data over five-minute periods. We initally planned to use an event window of -1 hour to +3 hours. However, if an event window of such size is implemented, there is a substantial reduction in the number of usable tweet observations available for analysis, as we would have to restrict the time period for tweets to 10:30 - 13:00 U.S. Eastern Time (ET).

Since the focus of the study is to examine the immediate market reactions to the tweets and the entire event window needs to occur during regular market hours, only tweets generated on trading days and during the time period from 10:00 – 15:30 (ET) are utilized. This limits the ability to generate a symmetric ±30 minute event window without going into the market opening/closing periods; however, the tweets outside of this time frame are maintained in the dataset, but are not incorporated into the intraday analysis. **Applying these restrictions results in 37 eligible Tesla-related tweets that allow for a complete ±30-minute event window.** The main reason for this is that Elon Musk tweets mostly during after-hours and on weekends.

Each tweet that is eligible for inclusion in the analysis has an event time $= 0$ defined as the first five-minute trading interval that occurs at or after the tweet timestamp on the same trading day. This provides synchronization between the trading and timing of the tweets so that the event represents the first point at which market participants are able to react to the information contained in the tweet. All eligible tweets that do not fall on the same trading day are omitted from the event window sample.

A 30-minute event window (6 five-minute intervals before and after the event time) is then created for each eligible tweet.

```
t = seq(-pre, post)
```

The event window is constructed based on trading bar index, as opposed to clock time, in order to guarantee a constant number of observations per event. The tweet level information, including sentiment scores and sentiment labels, is combined with the intraday observations. The event aligned dataset is saved and is used to estimate the intraday price and volume responses to the sentiment category.

4.5 Event-Study Estimation (Script 07)

Finally, the intraday event-study statistics are estimated utilizing the event windows established in the preceding step. The analysis is restricted to tweets that are posted during regular market hours, and utilizes ±30-minute event windows in order to ensure consistency across events. Event windows that do not contain the full complement of required five-minute intervals are eliminated; this ensures that each event produces the same number of observations as all other events throughout the dataset.

The five-minute frequencies are used to compute abnormal returns and aggregate these returns by sentiment category. For each sentiment category, the average five-minute return is calculated at each relative event time. This aggregation step is implemented as follows:

```
ar_tbl <- events_in_hours %>%
  group_by(sentiment_label, t_min) %>%
  summarise(
    mean_ret = mean(ret_5m_log, na.rm = TRUE),
    .groups = "drop"
  )
```

The sum of the average returns across each relative event time yields the cumulative abnormal returns (CAR), which captures the evolution of the price responses before and after the tweet. This approach focuses solely on the intraday price movements and avoids contaminating the results with overnight returns or post-close adjustments.

We also analyze trading behavior by calculating the mean relative trading volume for each five-minute interval within the event window and then aggregating these volumes by sentiment category. This allows us to determine if the tweets are associated with increased or decreased trading activity compared to normal intraday trading conditions.

All event-study statistics are displayed separately for negative, neutral, and positive tweets, allowing for direct comparisons of market price and volume reactions across sentiment categories. The average CAR and relative volume are graphically illustrated in the Results section, along with illustrative single-event examples.This provides the basis for assessing the short-term market impact of Elon Musk's Tesla-related tweets, dependent upon their sentiment. A simplified example of the plotting step is

```
ggplot(car_tbl, aes(x = t_min, y = CAR, color = sentiment_label)) +
  geom_line() +
  geom_vline(xintercept = 0, linetype = "dashed")
```

5 Results

In this section, we present the intraday event study results for Twitter posts made by Elon Musk regarding Tesla. The analysis combines illustrative single event examples with aggregated results across all eligible tweets. All reported results are based on tweets posted between 10:00 and 15:30 ET, ensuring that each event window contains a complete ±30-minute interval of intraday trading data.

5.1 Single-Event Examples

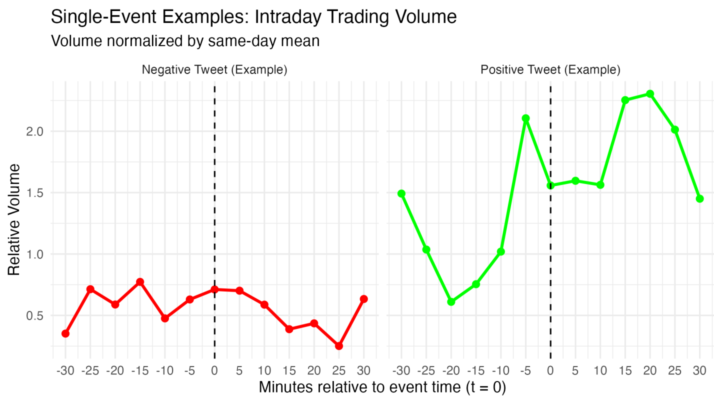
Figure 2 and Figure 3 illustrate both of the extremes (the most extreme negative sentiment tweets and the most extreme positive sentiment tweets) analyzed in terms of their intraday market reaction to tweets. Each figure illustrates how the event study framework used here provides insights into price and volume behaviors at the level of individual tweets.

Trading volume is normalized against the same day's average. The positive tweet is associated with significant trading volume at the event time, whereas the negative tweet shows lower and relatively even volume throughout most of the window. These examples illustrate that a single tweet can produce a wide variety of trading behavior.

The negative tweet shows a significant drop in CARs in the period immediately following the event timeline while the positive tweet demonstrates a positive increase in the CAR after the event timeframe. Pre-event movement and post-event movement varied significantly between the two examples, this highlights the heterogeneous nature of the individual tweet response.

Figure 2

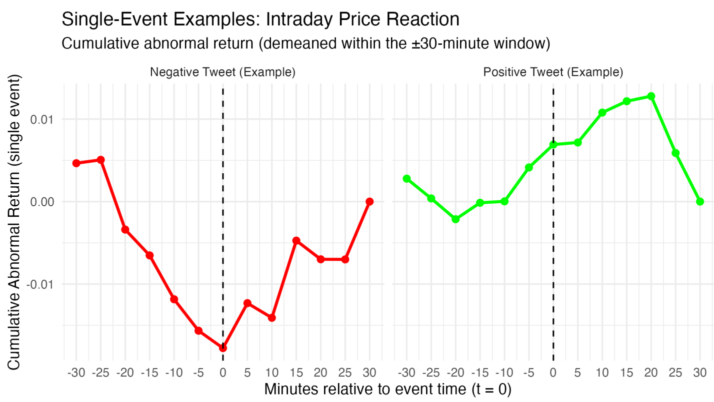
Single-event examples: intraday trading volume around a negative and a positive tweet



The figure shows intraday relative trading volume for two illustrative Tesla-related tweets: one negative and one positive example. Volume is normalized by the same-day mean trading volume. Event time t = 0 corresponds to the first 5-minute trading interval at or after the tweet. Each panel covers a ±30-minute window.

Figure 3

Single-event examples: intraday price reaction around a negative and a positive tweet



The figure shows cumulative abnormal returns (CAR) for two illustrative Tesla-related tweets: one negative and one positive example. Event time t = 0 corresponds to the first 5-minute trading interval at or after the tweet. Each panel covers a ±30-minute window.

These results collectively show a wide range of diversity in market reactions to individual tweets. Therefore, rather than relying solely on isolated examples of individual tweets, this encourages the use of aggregated event study results to identify systematic patterns in tweets.

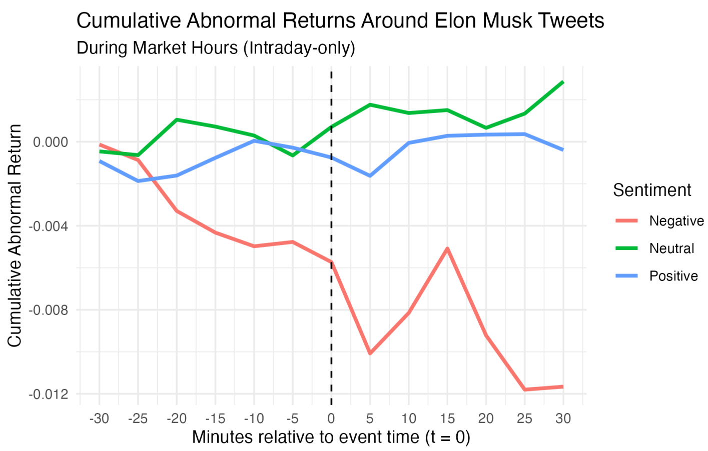
5.2 Aggregated Cumulative Abnormal Returns

Figure Figure 4 presents an overview of the average cumulative abnormal returns (CARs) for all tweets on the trading day that were sent prior to the closing bell, split into three groups based on their sentiment. Negative-sentimented tweets show an immediate drop in CAR at t=0 relative to the immediately preceding time interval; neutral- and positive-sentimented tweets do not exhibit such an effect – in fact, both have more stable or upward trending return paths than those generated by negative-sentimented tweets. These differences begin to be evident from around 10 to 15 minutes post-tweet.

The total number of tweets that were used to create the final intraday dataset is 37. The final dataset contains 22 tweets that were classified as being positive-sentimented, 8 tweets that were classified as being neutral-sentimented and 7 tweets that were classified as being negative-sentimented.

Figure 4

Average cumulative abnormal returns around Elon Musk's Tesla-related tweets by sentiment



The figure shows average cumulative abnormal returns (CAR) around Tesla-related tweets by Elon Musk, grouped by sentiment. Event time t = 0 corresponds to the first 5-minute trading interval at or after the tweet. Results are based on tweets posted on trading days between 10:00 and 15:30 ET.

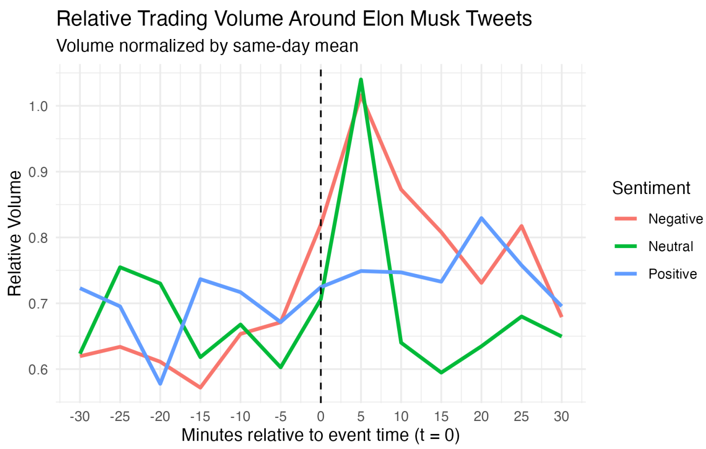
Large differences can be identified among sentiment groups after the tweet event. Negative-sentiment tweets are subject to a decrease in cumulative abnormal returns soon after event time (t = 0). This decrease is apparent in the first 10 to 15 minutes after the tweet and remains visible through the intraday window. On the other hand, tweets categorized as neutral or positive have flatter or only slightly increasing CAR paths, with no similar post-event decline. These differences indicate that short-term price reactions vary systematically with the sentiment of the tweet content.

5.3 Aggregated Trading Volume Responses

In addition to examining the direction of the sentiment in the tweet, we also examined how the overall level of investor attention (as measured by relative trading volume) responds to the tweet. As shown in figure Figure 5, we find that there is a very strong increase in the level of investor attention around the time of the tweet for all sentiment categories. We also found that neutral tweets elicit the largest increase in investor attention, which suggests that simply the fact that investors are paying more attention to the stock increases as a result of the tweet, regardless of whether the content of the tweet is clearly positive or negative.

Figure 5

Relative trading volume around Elon Musk's Tesla-related tweets by sentiment



The figure shows average relative trading volume around Tesla-related tweets by Elon Musk, grouped by sentiment. Volume is normalized by the same-day mean trading volume. Event time t = 0 corresponds to the first 5-minute trading interval at or after the tweet. Results are based on tweets posted on trading days between 10:00 and 15:30 ET.

A sharp jump in trading volume is seen for all three sentiment groups at the time of the tweet. The jumps in volume represent quite strong and rapid spikes in trading activity after a tweet. Although all sentiment categories exhibit a volume spike at t = 0, magnitude of the response differs across sentiment groups. Neutral-sentiment tweets exhibited the greatest spike in relative trading volume around the time of the event. Negative tweets also show a strong immediate increase but not as pronounced as the neutral tweets, while positive tweets exhibit a more moderate volume response in comparison. Following the peak, trading volume gradually declines but remains elevated relative to pre-event levels for several five-minute intervals. Overall, these findings indicate that Tesla-related tweets are always associated with higher than normal intraday trading activity, with the highest volume reaction being exhibited for neutral sentiment tweets.

6 Discussion

This research investigated how Tesla's stock price and trading activity changed in the minutes before and after Elon Musk's tweets by having utilized an intraday event-study approach. In combination with sentiment classified tweet data and high-frequency stock data, the analysis examined short-term changes in the market which were unobservable by using low frequency or daily datasets. The research aimed to determine if, and how, investor behavior differed based upon the communication from executives on social media, specifically, in instances where the communicator had extraordinary visibility and influence.

A couple of important trends arise from the analysis. First, the analysis reveals a consistent and sharp rise in trading volume immediately after Musk publishing a tweet regardless of the sentiment. This trend indicates that Musk's tweets function as attention grabbing events that initiate market involvement, similar to those identified by Engelberg and Parsons, who demonstrate that simply having exposure in media influences investor behavior (Engelberg and Parsons (2011)). The increase in volume for neutral tweets was particularly significant and possibly reflects interpretive uncertainty. Investors are likely to spend more time analyzing or hypothesizing regarding the significance of a neutral tweet than they would a tweet with obviously positive or negative content. This causes a burst of early trading as different segments of the market react differently to unclear information, reflecting what researchers reported: emotionally uncertain content increases trading volatility, particularly in times of unstable public mood (Bollen et al. (2011)).

Second, price movements depend on the sentiment contained within Musk's tweets. Generally speaking, tweets with negative sentiment result in a decline in CARs (cumulative abnormal returns) while tweets containing positive and neutral sentiments produce price movements that are either stable or slightly positive. Therefore, the research illustrates that sentiment-based communications from an influential figure like Musk do not only capture the attention of investors; rather, these communications are rapidly interpreted and factored into price formation, consistent with Tetlock, and Shah et al., who report that investor sentiment drives market reactions at faster time horizons (Tetlock (2007); Shah et al. (2019)). More importantly, the research illustrates that the market does not solely respond to the existence of a tweet; however, the market interprets and prices the directional tone of the content in real-time.

Despite these findings, there are a few limitations to this research. Firstly, the number of clean Musk tweets that match up with the opening of the market and meet the intraday criteria is relatively small, which may limit the generalizability of the results. Secondly, the sentiment classification was conducted by using a single model and applied uniformly to all tweets, which may either under-estimate or over-estimate the nuances of some tweets. For instance, Musk's use of sarcasm, memes or technical references in his tweets may still be imperfectly captured by the sentiment classification model we used.

Finally, although the analysis took place in narrow intraday windows, the stock market reaction could still be influenced by other concurrent events such as headline news or macroeconomic releases. Future studies may improve this by filtering out concurrent news releases using NLP techniques or conducting sensitivity analyses across broader market indexes. Furthermore, the dataset only includes Tesla stock data from 2017 to 2020. However, Musk's social media activity, and subsequently its influence, dramatically expanded beyond 2020, particularly in 2022, when he acquired Twitter. As demonstrated in Figure 1, Musk's rate of tweeting almost tripled following his acquisition of the platform, particularly on divisive or company related topics.

An extended stock data timeframe paired with these newer tweets may exhibit even more pronounced, or complex, dynamics. Although there are restrictions on the research, the results provide support for the central assertion that Elon Musk's tweets cause immediate, quantifiable impacts on Tesla's stock performance, primarily through enhanced trading activity and sentiment sensitive price movements. These findings contribute to a growing body of literature indicating that executive communications via social media may function as a real-time form of market moving disclosure, especially when the communicator is both extremely visible and perceived to be closely linked to the firm's value. Moreover, the research demonstrates the importance of utilizing intraday data and time aligned with events when researching the high-frequency effects of information in financial markets. As communications in financial markets increasingly take place in platforms similar to X, traditional models of price discovery and investor information processing may require adaptation to account for these rapid, informal, and sentiment driven modes of information transmission.

7 Conclusion

Two important takeaways can be drawn from this study's results. Firstly, there is a substantial increase in trading activity immediately after a tweet by Elon Musk, regardless of whether the content of the tweet has a positive or negative tone. The fact that Musk's communications receive so much attention from traders reflects how well known he is and the impact that his communications have on the market. Interestingly, the tweets that were classified as neutral had the largest volume reaction, which suggests that when Musk communicates and does not clearly provide positive or negative cues, investors often have different interpretations of the information being communicated, resulting in increased trading for a brief period while the collective market process seeks to understand the meaning behind the message. It appears that this ambiguity in interpreting the messages heightens the degree to which market participants become engaged with the information provided via social media.

Secondly, the sentiment expressed within Musk's tweets has a substantial effect upon the price movements in the market during the very short-term. Those tweets that were characterized by negative sentiment resulted in considerable decreases in CARs, whereas tweets characterized by neutral or positive sentiment produced more stable or slightly positive CARs. These patterns support the idea that investors do not merely respond to the fact that an executive is communicating publicly; rather they quickly analyse and value the tone of the communication. These patterns also reinforce the growing body of research that demonstrates the relationship between investor attention and real-time sentiment and the behavior of investors in the marketplace. Therefore, these results demonstrate the increasing role that social media plays as both a conduit for information about companies and as an action channel in the markets for securities.

8 Author Contributions

This project was completed collaboratively by both students. Individual contributions were as follows:

- **Mira Heinemann**: Led the tweet preprocessing pipeline, the sentiment analysis workflow and the exploratory data visualizations. She was responsible for writing the following report sections: Introduction, Methodology and Results.

- **Esra Tonleu**: Led the intraday stock data processing, event-window construction, and event-study estimation. She was responsible for writing the following report sections: Data, Background Literature, Discussion and Conclusion.

Both were equal contributors to:

- The interpretation of the results
- Writing and refinement of the report
- Final review, editing and integration of all parts to ensure consistency and coherence

9 Use of AI Tools

AI tools (ChatGPT / Copilot) were employed as coding assistant tools throughout the coding process and provided assistance to the authors with respect to identifying alternative methods of coding or implementing a solution to a coding question. The authors wrote, adapted, and understood all of their final code.

10 References

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1–8. https://doi.org/10.1016/j.jocs.2010.12.007

Bushee, B. J., Core, J. E., Guay, W. R., & Hamm, S. J. W. (2010). The role of the business press as an information intermediary. *Journal of Accounting Research*, *48*(1), 1–19. https://doi.org/10.1111/j.1475-679X.2009.00357.x

Cutler, D. M., Poterba, J. M., & Summers, L. H. (1989). What moves stock prices? *Journal of Portfolio Management*, *15*(3), 4–12. https://doi.org/10.3905/jpm.1989.409212

Engelberg, J., & Parsons, C. A. (2011). The causal impact of media in financial markets. *The Journal of Finance*, *66*(1), 67–97. https://doi.org/10.1111/j.1540-6261.2010.01626.x

Garg, N., & Tiwari, A. K. (2021). The role of social media in stock markets: A systematic review. *Journal of Business Research*, *124*, 170–178. https://doi.org/10.1016/j.jbusres.2020.11.021

Kwon, O., Lee, M., & Kim, J. (2022). CEO social media communication and stock market reactions. *Journal of Corporate Finance*, *74*, 102208. https://doi.org/10.1016/j.jcorpfin.2022.102208

Shah, D., Isah, H., & Zulkernine, F. (2019). Stock market analysis: A review and taxonomy of prediction techniques. *International Journal of Financial Studies*, *7*(2), 26. https://doi.org/10.3390/ijfs7020026

Sprenger, T. O., Tumasjan, A., Sandner, P. G., & Welpe, I. M. (2014). Tweets and trades: The information content of stock microblogs. *European Financial Management*, *20*(5), 926–957. https://doi.org/10.1111/j.1468-036X.2013.12007.x

Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *The Journal of Finance*, *62*(3), 1139–1168. https://doi.org/10.1111/j.1540-6261.2007.01232.x

Yahoo Finance. (2025). *Tesla (TSLA) stock price, quote & news*. https://finance.yahoo.com/quote/TSLA

Zheludev, I., Smith, P., & Aste, T. (2014). When can social media predict financial markets? *Scientific Reports*, *4*, 4213. https://doi.org/10.1038/srep04213

Appendix A

Tweet Preprocessing

Code of 01_import_tweets.R

```
##############################################################
# 01_import_tweets.R
# Data pipeline for Elon Musk tweets
#
# Tasks:
#   - Load raw CSV
#   - Remove retweets / replies / quotes
#   - Parse timestamps correctly (UTC -> ET)
#   - Classify Tesla-related tweets
#   - Classify timing relative to market hours (ET)
#   - Save cleaned outputs
##############################################################

library(tidyverse)
library(lubridate)

# ---------------------------------------------------------
# Define analysis period
# ---------------------------------------------------------
analysis_start <- as.Date("2019-06-01")
analysis_end   <- as.Date("2020-05-31")

# ---------------------------------------------------------
# Read raw tweet data
# ---------------------------------------------------------
tweets_raw <- read_csv(
  "data/all_musk_posts.csv",
  col_types = cols(
    id = col_character()
```

```
  )
) %>%
  mutate(
    row_original = row_number()
  )

# --------------------------------------------------------
# Keep only original Musk tweets
# --------------------------------------------------------
tweets_filtered <- tweets_raw %>%
  filter(
    isRetweet != "TRUE",
    isReply   != "TRUE",
    isQuote   != "TRUE"
  )

# --------------------------------------------------------
# Parse timestamps & basic cleaning
# --------------------------------------------------------
tweets_clean <- tweets_filtered %>%
  mutate(
    # Parse createdAt (contains +00:00 offset)
    createdAt_utc = ymd_hms(createdAt, tz = "UTC"),
    createdAt_et  = with_tz(createdAt_utc, "America/New_York"),

    # Store ET timestamp as string WITH UTC offset so timezone survives CSV
    createdAt_et_str = format(createdAt_et, "%Y-%m-%d %H:%M:%S%z"),

    date_et       = as.Date(createdAt_et),

    # Detect link-only tweets
    is_link_only = str_detect(
```

```
      str_squish(fullText),
      "^https?://t\\.co/[A-Za-z0-9]+$"
    )
  ) %>%
  filter(
    date_et >= analysis_start,
    date_et <= analysis_end
  )

# ---------------------------------------------------------
# Classify Tesla-related tweets
# ---------------------------------------------------------
tesla_keywords <- c(
  # Company
  "tesla", "tesla inc", "tesla motors", "tsla",
  # Vehicles
  "model s", "model 3", "model x", "model y",
  "cybertruck", "roadster",
  # Autonomy & software
  "fsd", "fsd beta", "full self-driving", "autopilot", "autonomy",
  "self driving", "dojo", "ai day", "neural net",
  # Manufacturing & factories
  "gigafactory", "giga berlin", "giga shanghai", "giga texas",
  # Energy products
  "megapack", "powerwall", "battery", "batteries", "cells", "4680",
  "solar", "solar roof",
  # Infrastructure
  "supercharger", "charging", "charger" )

pattern_tesla <- paste(tesla_keywords, collapse = "|")

tweets_clean <- tweets_clean %>%
```

```
  mutate(
    fullText_lower   = str_to_lower(fullText),
    is_tesla_related = str_detect(fullText_lower, pattern_tesla)
  )

# ---------------------------------------------------------------------------
# Classify timing relative to market hours (Eastern Time = Stockmarket Time)
# ---------------------------------------------------------------------------
tweets_clean <- tweets_clean %>%
  mutate(
    # Extract ET components
    date_et = as.Date(createdAt_et),
    hm_et   = format(createdAt_et, "%H:%M"),
    dow_et  = wday(createdAt_et, week_start = 1),  # 1 = Mon, 7 = Sun

    # Logical flags
    is_weekday = dow_et <= 5,
    is_market_hours =
      is_weekday &
      hm_et >= "09:30" &
      hm_et <  "16:00",

    # Final classification
    time_status = case_when(
      is_market_hours ~ "During Market Hours",
      is_weekday      ~ "Premarket/Afterhours",
      TRUE            ~ "Weekend"
    )
  )

# ---------------------------------------------------------
# Create tweet category
```

```
# ---------------------------------------------------------
tweets_clean <- tweets_clean %>%
  mutate(
    tweet_category = case_when(
      is_link_only      ~ "Link-Only",
      is_tesla_related  ~ "Tesla-Related",
      TRUE              ~ "Non-Tesla"
    )
  )

# ---------------------------------------------------------
# Final column selection
# ---------------------------------------------------------
tweets_clean <- tweets_clean %>%
  select(
    row_original,
    id,
    fullText,
    createdAt,
    createdAt_utc,
    createdAt_et_str,
    date_et,
    time_status,
    tweet_category,
    is_tesla_related,
    is_link_only,
    retweetCount,
    replyCount,
    likeCount,
    quoteCount
  )
```

```r
# -----------------------------------------------------------
# Save outputs
# -----------------------------------------------------------
write_csv(tweets_clean, "data/musk_tweets_clean.csv")

tweets_tesla <- tweets_clean %>%
  filter(is_tesla_related)

write_csv(tweets_tesla, "data/musk_tesla_tweets_clean.csv")

# -----------------------------------------------------------
# Quick diagnostics (console output only)
# -----------------------------------------------------------
cat("\n=== Tweet counts by category ===\n")
print(
  tweets_clean %>%
    count(tweet_category) %>%
    mutate(percent = round(n / sum(n) * 100, 1))
)

cat("\n=== Tesla Tweet counts by time_status ===\n")
print(
  tweets_tesla %>%
    count(time_status)
)

#############################################################
# End of 01_import_tweets.R
#############################################################
```

Appendix B

Sentiment Analysis

Code of 02_sentiment.R

```r
##############################################################
# 02_sentiment.R
# Sentiment analysis for Tesla-related Elon Musk tweets
#
# Tasks:
#   - Load cleaned Tesla-related tweets from 01
#   - Exclude link-only tweets
#   - Compute sentiment scores
#   - Save enriched dataset
##############################################################

library(tidyverse)
library(sentiment.ai)

# ---------------------------------------------------------
# Load cleaned Tesla-related tweets
# ---------------------------------------------------------
tweets_tesla <- read_csv(
  "data/musk_tesla_tweets_clean.csv",
  col_types = cols(id = col_character())
)

# ---------------------------------------------------------
# Remove link-only tweets (not meaningful for sentiment)
# ---------------------------------------------------------
tweets_for_sentiment <- tweets_tesla %>%
  filter(!is_link_only)

# ---------------------------------------------------------
```

```
# Initialize sentiment.ai engine
# ---------------------------------------------------------
# Assumes the environment was created once via
#   conda create -n sentiment-env python=3.9
#   pip install tensorflow==2.10 tensorflow-text==2.10 numpy==1.26.4
init_sentiment.ai(method = "conda", envname = "sentiment-env")

# ---------------------------------------------------------
# Compute sentiment scores
# ---------------------------------------------------------
tweets_with_sentiment <- tweets_for_sentiment %>%
  mutate(
    sentiment_score = sentiment_score(fullText),
    sentiment_label = case_when(
      sentiment_score < -0.3 ~ "Negative",
      sentiment_score >  0.3 ~ "Positive",
      TRUE                   ~ "Neutral"
    )
  )

# ---------------------------------------------------------
# Save output
# ---------------------------------------------------------
write_csv(
  tweets_with_sentiment,
  "data/musk_tesla_tweets_with_sentiment.csv"
)

# ---------------------------------------------------------
# Diagnostics (console output only)
# ---------------------------------------------------------
cat("\n=== Sentiment label distribution ===\n")
```

```
print(
  tweets_with_sentiment %>%
    count(sentiment_label)
)

cat("\n=== Tesla tweets by time_status (after sentiment) ===\n")
print(
  tweets_with_sentiment %>%
    count(time_status)
)

############################################################
# End of 02_sentiment.R
############################################################
```

Appendix C

Intraday Stock Data Processing

Code of 03_import_intraday_5min.R Code of 05_prepare_returns.R

```
##############################################################
# 03_import_intraday_5min.R
# Data pipeline for Tesla intraday stock market data
# - Load raw CSV
# - Parse Date/Time & timezone
# - Filter to analysis period (and market hours)
# - Save cleaned outputs
##############################################################

# Define analysis period
analysis_start <- as.Date("2019-06-01")
analysis_end   <- as.Date("2020-05-31")

# Load packages ---------------------------------------------------------
library(tidyverse)
library(lubridate)

# Read raw CSV -------------------------------------------------------
stock_raw <- read_csv(
  "data/all_tesla_intraday_5min.csv",
  col_types = cols(
    Open   = col_double(),
    High   = col_double(),
    Low    = col_double(),
    Close  = col_double(),
    Volume = col_double(),
    Date   = col_character(),
    Time   = col_character()
  )
```

```
)

# Parse Date/Time and set timezone -----------------------------------
stock_clean <- stock_raw %>%
  mutate(
    # Parse Date (DD-MM-YYYY)
    date = dmy(Date),

    # IMPORTANT:
    # The raw file's Date/Time are already in exchange clock time (New York time).
    # Therefore: parse as "native" local time, then *label* it as America/New_York.
    datetime_local = dmy_hms(paste(Date, Time)),
    datetime_et    = force_tz(datetime_local, tzone = "America/New_York")
  ) %>%
  mutate(
    # Store stable ET datetime string (with offset) so timezone survives CSV round-tr
    datetime_et_str = format(datetime_et, "%Y-%m-%d %H:%M:%S%z"),
    date_et = as.Date(datetime_et)
  )

# Keep only records for analysis period ------------------------------
stock_clean <- stock_clean %>%
  filter(
    date >= analysis_start,
    date <= analysis_end
  )

# Save cleaned data -------------------------------------
write_csv(stock_clean, "data/tesla_stock_clean.csv")

##############################################################
# End of pipeline
```

```
############################################################
```

```
############################################################
# 05_prepare_returns.R
# Prepare 5-minute intraday returns and normalized volume (ET-consistent)
#
# Uses:
#   - data/tesla_stock_clean.csv  (from 03_import_intraday_5min.R)
#     expected columns: Open, High, Low, Close, Volume, date_et, datetime_et_str
#
# Creates:
#   - data/tesla_intraday_returns_5min.csv
#
############################################################

library(tidyverse)
library(lubridate)

# 1) Read cleaned intraday stock data ----------------------------------
stock <- read_csv("data/tesla_stock_clean.csv") %>%
  mutate(
    # IMPORTANT:
    # Parse while respecting the embedded offset in datetime_et_str.
    # DO NOT pass tz="America/New_York" into ymd_hms() here.
    datetime_et = with_tz(ymd_hms(datetime_et_str), "America/New_York"),
    date_et = as.Date(datetime_et)
  ) %>%
  arrange(date_et, datetime_et)

# 2) Compute 5-minute log returns ---------------------------------------
stock_returns <- stock %>%
  group_by(date_et) %>%
```

```
  mutate(
    close_lag = lag(Close),
    ret_5m_log = log(Close) - log(close_lag)
  ) %>%
  ungroup()

# 3) Daily volume baselines ---------------------------------------------
stock_returns <- stock_returns %>%
  group_by(date_et) %>%
  mutate(
    vol_5m_mean_day   = mean(Volume, na.rm = TRUE),
    vol_5m_median_day = median(Volume, na.rm = TRUE),
    vol_rel_to_day_mean = if_else(vol_5m_mean_day > 0, Volume / vol_5m_mean_day, NA_r
  ) %>%
  ungroup()

# 4) Keep essential columns ---------------------------------------------
intraday_prepped <- stock_returns %>%
  mutate(
    # Save a stable ET datetime string for downstream scripts
    datetime_et_str = format(datetime_et, "%Y-%m-%d %H:%M:%S%z")
  ) %>%
  select(
    date_et,
    datetime_et_str,
    datetime_et,
    Open, High, Low, Close, Volume,
    ret_5m_log,
    vol_5m_mean_day, vol_5m_median_day, vol_rel_to_day_mean
  )

# 5) Save prepared dataset ----------------------------------------------
```

```
write_csv(intraday_prepped, "data/tesla_intraday_returns_5min.csv")


# 6) Quick summary -------------------------------------------------------
summary_stats <- intraday_prepped %>%
  summarise(
    n_rows = n(),
    n_days = n_distinct(date_et),
    min_dt_et = format(min(datetime_et, na.rm = TRUE), "%Y-%m-%d %H:%M:%S %Z"),
    max_dt_et = format(max(datetime_et, na.rm = TRUE), "%Y-%m-%d %H:%M:%S %Z"),
    na_returns = sum(is.na(ret_5m_log))
  )

print(summary_stats)

##############################################################
# End of 05_prepare_returns.R
##############################################################
```

Appendix D

Event Window Construction

Code of 06_event_window_builder.R

```
##############################################################
# 06_event_window_builder.R  (INTRADAY-ONLY, CORRECTED)
#
# - ONLY tweets during market hours AND between 10:00-15:30 ET
# - Event time (t=0) = first 5-min bar at/after tweet time, SAME DAY
# - Window: -6..+6  (±30 minutes)
# - Produces diagnostics for dropped tweets
#
# Creates file tesla_event_windows_5min.csv.
# For every tweet that is to be considered this then contains
# one row for each of the related stock market 5 minute data
# 1st row ´= -30 min
# 2nd row for same tweet = -25 min
# ...
# 12th or 13th row for same tweet = +30 min
##############################################################

library(tidyverse)
library(lubridate)

# ------------------------------
# Parameters
# ------------------------------
window_pre  <- 6
window_post <- 6

intraday_start_hm <- "10:00"
intraday_end_hm   <- "15:30"
```

```
# ------------------------------
# Load tweets (use createdAt_et; no datetime_et_str)
# ------------------------------
tweets_all <- read_csv(
  "data/musk_tesla_tweets_with_sentiment.csv",
  col_types = cols(id = col_character())
) %>%
  mutate(
    # Rebuild ET datetime from createdAt_et_str (CSV-safe)
    createdAt_et = with_tz(ymd_hms(createdAt_et_str), "America/New_York"),
    hm_et = format(createdAt_et, "%H:%M"),
    date_et = as.Date(createdAt_et)
  )

# Restrict to intraday *market-hours* tweets only
tweets <- tweets_all %>%
  filter(
    time_status == "During Market Hours",
    hm_et >= intraday_start_hm,
    hm_et <= intraday_end_hm
  )

# ------------------------------
# Load intraday returns
# ------------------------------
intraday_raw <- read_csv("data/tesla_intraday_returns_5min.csv") %>%
  mutate(
    # Rebuild ET datetime from datetime_et_str (CSV-safe)
    datetime_et = with_tz(ymd_hms(datetime_et_str), "America/New_York"),
    date_et = as.Date(datetime_et)
  ) %>%
  arrange(datetime_et)
```

```r
# ------------------------------
# Helper: align a tweet to the first bar at or after tweet time (same day)
# ------------------------------
get_bar_at_or_after_same_day <- function(dt, intraday_by_date) {
  dt_date <- as.Date(dt)

  # If the date doesn't exist in stock data, return NA
  if (!(as.character(dt_date) %in% names(intraday_by_date))) return(as.POSIXct(NA))

  day_tbl <- intraday_by_date[[as.character(dt_date)]]

  idx <- which(day_tbl$datetime_et >= dt)[1]
  if (is.na(idx)) return(as.POSIXct(NA))
  day_tbl$datetime_et[idx]
}

# Split intraday by date for fast lookups
intraday_by_date <- intraday_raw %>%
  group_by(date_et) %>%
  group_split() %>%
  setNames(as.character(unique(intraday_raw$date_et)))

# ------------------------------
# Compute event_dt for each tweet (same-day alignment)
# ------------------------------
tweets_events <- tweets %>%
  mutate(
    event_dt = map_dbl(createdAt_et, ~ as.numeric(get_bar_at_or_after_same_day(.x, in
  ) %>%
  mutate(
    event_dt = as.POSIXct(event_dt, origin = "1970-01-01", tz = "America/New_York")
```

```
  )

# Drop tweets that cannot align to a same-day bar
tweets_events <- tweets_events %>%
  filter(!is.na(event_dt))

# ------------------------------
# Build event windows (-6..+6)
# ------------------------------
# Build window by BAR INDEX around event_dt (±6 bars), not by "clock-time contains 13
build_one_window <- function(tweet_row, intraday_tbl, pre, post) {
  e_dt <- tweet_row$event_dt
  e_date <- as.Date(e_dt)

  day_tbl <- intraday_tbl %>% filter(date_et == e_date)

  # Find the bar index corresponding to event_dt
  idx <- which(day_tbl$datetime_et == e_dt)[1]
  if (is.na(idx)) return(NULL)

  start_idx <- idx - pre
  end_idx   <- idx + post

  # If we don't have enough bars before/after, drop this tweet (true boundary/gap iss
  if (start_idx < 1 || end_idx > nrow(day_tbl)) return(NULL)

  day_tbl[start_idx:end_idx, ] %>%
    mutate(
      tweet_id        = tweet_row$id,
      tweet_time_et   = tweet_row$createdAt_et,
      event_time_et   = e_dt,
      event_date      = as.Date(e_dt),
```

```
      time_status     = tweet_row$time_status,
      sentiment_score = tweet_row$sentiment_score,
      sentiment_label = tweet_row$sentiment_label,
      retweetCount    = tweet_row$retweetCount,
      replyCount      = tweet_row$replyCount,
      likeCount       = tweet_row$likeCount,
      quoteCount      = tweet_row$quoteCount,
      t = seq(-pre, post)
    )
}

event_windows_list <- vector("list", nrow(tweets_events))

for (i in seq_len(nrow(tweets_events))) {
  event_windows_list[[i]] <- build_one_window(tweets_events[i, ], intraday_raw, windo
}

event_windows <- bind_rows(event_windows_list)

# ------------------------------
# Diagnostics
# ------------------------------
cat("\n=== INTRADAY-ONLY ALIGNMENT SUMMARY (10:00-15:30 ET) ===\n")
alignment_summary <- tweets_events %>%
  summarise(
    n_events_total = n(),
    mean_shift_min = mean(as.numeric(difftime(event_dt, createdAt_et, units = "mins")
    median_shift_min = median(as.numeric(difftime(event_dt, createdAt_et, units = "mi
  )
print(alignment_summary)

cat("\n=== EVENT COUNTS BY SENTIMENT (Distinct tweets) ===\n")
```

```
print(
  event_windows %>%
    distinct(tweet_id, sentiment_label) %>%
    count(sentiment_label, name = "n_events")
)


# -------------------------------
# Save output
# -------------------------------
write_csv(event_windows, "data/tesla_event_windows_5min.csv")


#############################################################
# End of 06_event_window_builder.R
#############################################################
```

Appendix E

Event Study Estimation

Code of 07_event_study_results.R

```
##############################################################

# 07_event_study_results.R  (INTRADAY-ONLY)

#

# Intraday analysis only:

#   - Tweets During Market Hours

#   - NO after-hours / next-open analysis

#

# Uses:

#   - data/tesla_event_windows_5min.csv

#

# Produces:

#   - Event count summary

#   - CAR plot (minutes)

#   - Relative volume plot (minutes)

#   - Single-event examples (1 Positive + 1 Negative) in one 2-panel figure

#       * Positive = green

#       * Negative = red

##############################################################


library(tidyverse)

library(lubridate)


# -------------------------------

# Parameters (must match 06)

# -------------------------------

window_pre  <- 6

window_post <- 6

rows_per_event_expected <- window_pre + window_post + 1
```

```
# ------------------------------------------------------------------------------
# Single-event examples (we manually chose the most positive and most negative tweet
# from data/tesla_event_windows_5min.csv  (column sentiment_score)
# ------------------------------------------------------------------------------
# Paste your chosen tweet IDs here (must exist in data/tesla_event_windows_5min.csv)
single_positive_id <- '1214956305808527361'   #most positive tweet
single_negative_id <- '1242513824885944320'   #most negative tweet

# Where to save plots for Quarto
fig_dir <- "figs"

# ------------------------------
# Load event windows
# Each row = one 5-minute interval in the ±30-minute window around a tweet
# ------------------------------
events <- read_csv(
  "data/tesla_event_windows_5min.csv",
  col_types = cols(tweet_id = col_character())
)

# Keep only complete windows (±30 min)
event_window_counts <- events %>%
  count(tweet_id, time_status, sentiment_label, name = "n_rows_window")

events <- events %>%
  inner_join(
    event_window_counts %>%
      filter(n_rows_window == rows_per_event_expected) %>%
      select(tweet_id),
    by = "tweet_id"
  )
```

```r
# ------------------------------
# Restrict to tweets posted during market hours
# After-hours tweets cannot be evaluated intraday
# Convert event index t into actual minutes (t_min = t*5)
# ------------------------------
events_in_hours <- events %>%
  filter(time_status == "During Market Hours") %>%
  mutate(
    t_min = t * 5   # convert bar index to minutes
  )

# ------------------------------
# Count number of usable events by sentiment
# ------------------------------
cat("\n=== EVENT COUNTS (Distinct tweets) ===\n")
print(
  events_in_hours %>%
    distinct(tweet_id, sentiment_label) %>%
    count(sentiment_label, name = "n_events") %>%
    arrange(sentiment_label)
)

# ------------------------------
# Compute mean abnormal returns at each minute
# Then compute CAR by cumulatively summing mean returns
# ------------------------------
ar_tbl <- events_in_hours %>%
  group_by(sentiment_label, t_min) %>%
  summarise(
    mean_ret = mean(ret_5m_log, na.rm = TRUE),
    .groups = "drop"
  )
```

```
car_tbl <- ar_tbl %>%
  group_by(sentiment_label) %>%
  arrange(t_min) %>%
  mutate(CAR = cumsum(mean_ret)) %>%
  ungroup()

# ------------------------------
# Compute mean relative trading volume at each minute
# Volume is normalized by same-day mean (done in Script 05)
# ------------------------------
vol_tbl <- events_in_hours %>%
  group_by(sentiment_label, t_min) %>%
  summarise(
    mean_rel_vol = mean(vol_rel_to_day_mean, na.rm = TRUE),
    .groups = "drop"
  )

# ------------------------------
# Plot: CAR (minutes) - aggregated - by sentiment
# ------------------------------
p_car <- ggplot(car_tbl, aes(x = t_min, y = CAR, color = sentiment_label)) +
  geom_line(linewidth = 1.1) +
  geom_vline(xintercept = 0, linetype = "dashed") +
  scale_x_continuous(breaks = seq(-30, 30, by = 5)) +
  labs(
    title = "Cumulative Abnormal Returns Around Elon Musk Tweets",
    subtitle = "During Market Hours (Intraday-only)",
    x = "Minutes relative to event time (t = 0)",
    y = "Cumulative Abnormal Return",
    color = "Sentiment"
  ) +
```

```
  theme_minimal()

# ------------------------------
# Plot: Relative Volume (minutes) - aggregated - by sentiment
# ------------------------------
p_vol <- ggplot(vol_tbl, aes(x = t_min, y = mean_rel_vol, color = sentiment_label)) +
  geom_line(linewidth = 1.1) +
  geom_vline(xintercept = 0, linetype = "dashed") +
  scale_x_continuous(breaks = seq(-30, 30, by = 5)) +
  labs(
    title = "Relative Trading Volume Around Elon Musk Tweets",
    subtitle = "Volume normalized by same-day mean",
    x = "Minutes relative to event time (t = 0)",
    y = "Relative Volume",
    color = "Sentiment"
  ) +
  theme_minimal()

print(p_car)
print(p_vol)

# ==========================================================
# Single-event examples (combined into one 2-panel figure)
# ==========================================================

# ------------------------------
# Build single-event dataset
# ------------------------------
single_events <- events_in_hours %>%
  filter(tweet_id %in% c(single_positive_id, single_negative_id)) %>%
  mutate(
    # Label panels explicitly (so facet order/meaning is stable)
```

```
    example_type = case_when(
      tweet_id == single_positive_id ~ "Positive Tweet (Example)",
      tweet_id == single_negative_id ~ "Negative Tweet (Example)",
      TRUE ~ "Example"
    )
  ) %>%
  group_by(tweet_id) %>%
  arrange(t_min) %>%
  mutate(
    # Abnormal return for this single day/window (demeaned within the 13 bars)
    ar_5m_single = ret_5m_log - mean(ret_5m_log, na.rm = TRUE),
    CAR_single   = cumsum(ar_5m_single)
  ) %>%
  ungroup()

# Safety check: if IDs are wrong, stop with a clear message
if (!all(c(single_positive_id, single_negative_id) %in% unique(events_in_hours$tweet_
  stop("One or both single tweet IDs were not found in events_in_hours. Check the twe
}

# -------------------------------
# Plot: Single-event CAR (2 panels; green/red)
# -------------------------------
p_single_car <- ggplot(single_events, aes(x = t_min, y = CAR_single, color = example_
  geom_line(linewidth = 1.0) +
  geom_point(size = 2) +
  geom_vline(xintercept = 0, linetype = "dashed") +
  facet_wrap(~ example_type, ncol = 2, scales = "fixed") +
  scale_x_continuous(breaks = seq(-30, 30, by = 5)) +
  scale_color_manual(
    values = c(
      "Positive Tweet (Example)" = "green",
```

```
      "Negative Tweet (Example)" = "red"
    ),
    guide = "none"
  ) +
  labs(
    title = "Single-Event Examples: Intraday Price Reaction",
    subtitle = "Cumulative abnormal return (demeaned within the ±30-minute window)",
    x = "Minutes relative to event time (t = 0)",
    y = "Cumulative Abnormal Return (single event)"
  ) +
  theme_minimal()

print(p_single_car)

# =========================================================
# Single-event Relative Volume
# =========================================================

p_single_vol <- ggplot(
  single_events,
  aes(x = t_min, y = vol_rel_to_day_mean, color = example_type)
) +
  geom_line(linewidth = 1.0) +
  geom_point(size = 2) +
  geom_vline(xintercept = 0, linetype = "dashed") +
  facet_wrap(~ example_type, ncol = 2, scales = "fixed") +
  scale_x_continuous(breaks = seq(-30, 30, by = 5)) +
  scale_color_manual(
    values = c(
      "Positive Tweet (Example)" = "green",
      "Negative Tweet (Example)" = "red"
    ),
```

```
    guide = "none"
  ) +
  labs(
    title = "Single-Event Examples: Intraday Trading Volume",
    subtitle = "Volume normalized by same-day mean",
    x = "Minutes relative to event time (t = 0)",
    y = "Relative Volume"
  ) +
  theme_minimal()

print(p_single_vol)




# ==========================================================
# Save plots for Quarto
# ==========================================================
#dir.create(fig_dir, showWarnings = FALSE, recursive = TRUE)

# Aggregated plots
ggsave(file.path(fig_dir, "fig_car_intraday.png"), p_car,
       width = 16, height = 10, units = "cm", dpi = 300)

ggsave(file.path(fig_dir, "fig_volume_intraday.png"), p_vol,
       width = 16, height = 10, units = "cm", dpi = 300)

# Single-event Combine figures
ggsave(file.path(fig_dir, "fig_single_examples_car.png"), p_single_car,
       width = 18, height = 10, units = "cm", dpi = 300)

ggsave(file.path(fig_dir, "fig_single_examples_volume.png"), p_single_vol,
       width = 18, height = 10, units = "cm", dpi = 300)
```

```
#############################################################
# End of 07_event_study_results.R
#############################################################
```

Appendix F

Exploratory Data Analysis (Code + Figures)

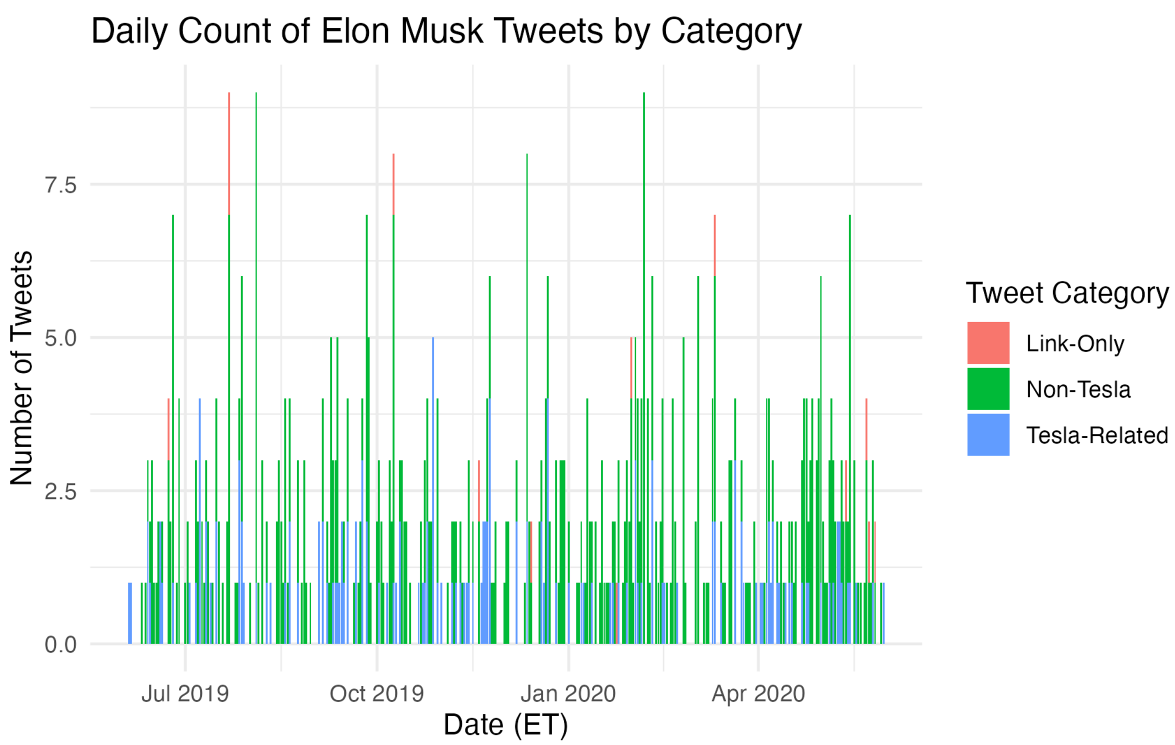
Code and Figures of A_plot_EDA.R and B_tweet_timing_histogram.R

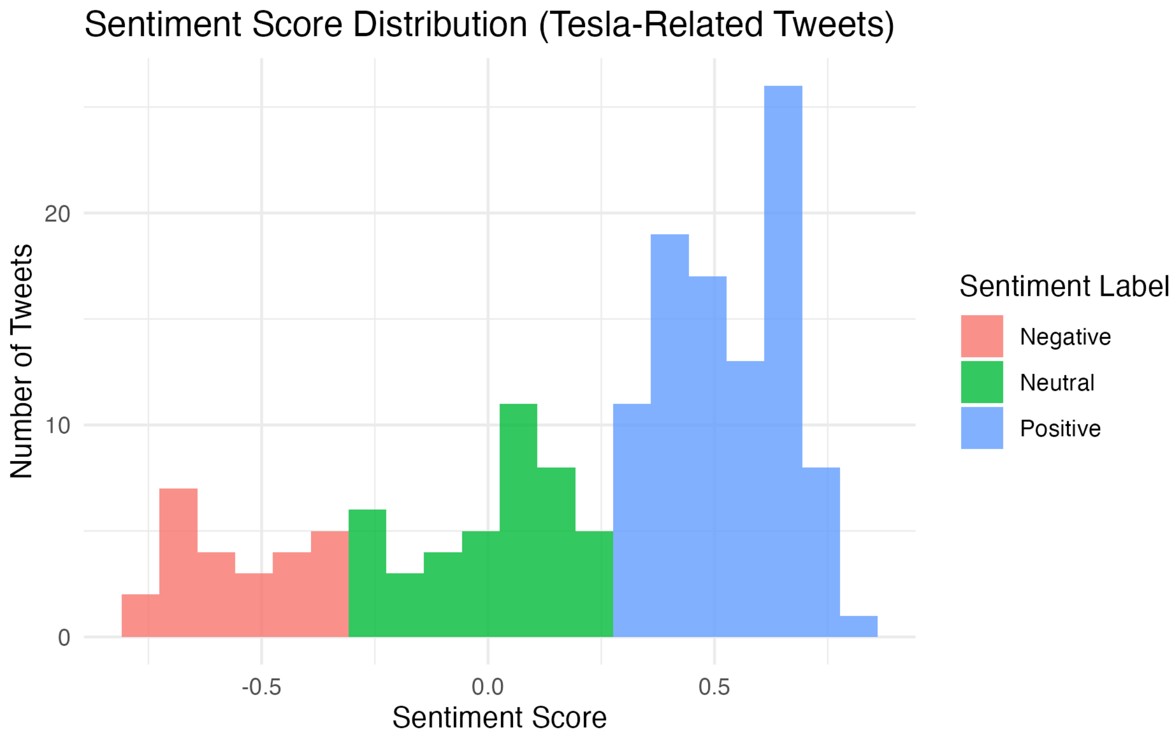
```
##############################################################
# A_plot_EDA.R
# Exploratory Data Analysis (EDA) plots for Musk tweet data
# Uses:
#   - data/musk_tweets_clean.csv
#   - data/musk_tesla_tweets_with_sentiment.csv
##############################################################

library(tidyverse)
library(lubridate)

# 1) Load cleaned tweet data -------------------------------------------
tweets_clean <- read_csv(
  "data/musk_tweets_clean.csv",
  col_types = cols(
    id = col_character()
  )
) %>%
  mutate(
    createdAt_et = with_tz(ymd_hms(createdAt_et_str), "America/New_York"),
    date_et = as.Date(createdAt_et)
  )

tweets_tesla_sent <- read_csv(
  "data/musk_tesla_tweets_with_sentiment.csv",
  col_types = cols(
    id = col_character()
  )
) %>%
```

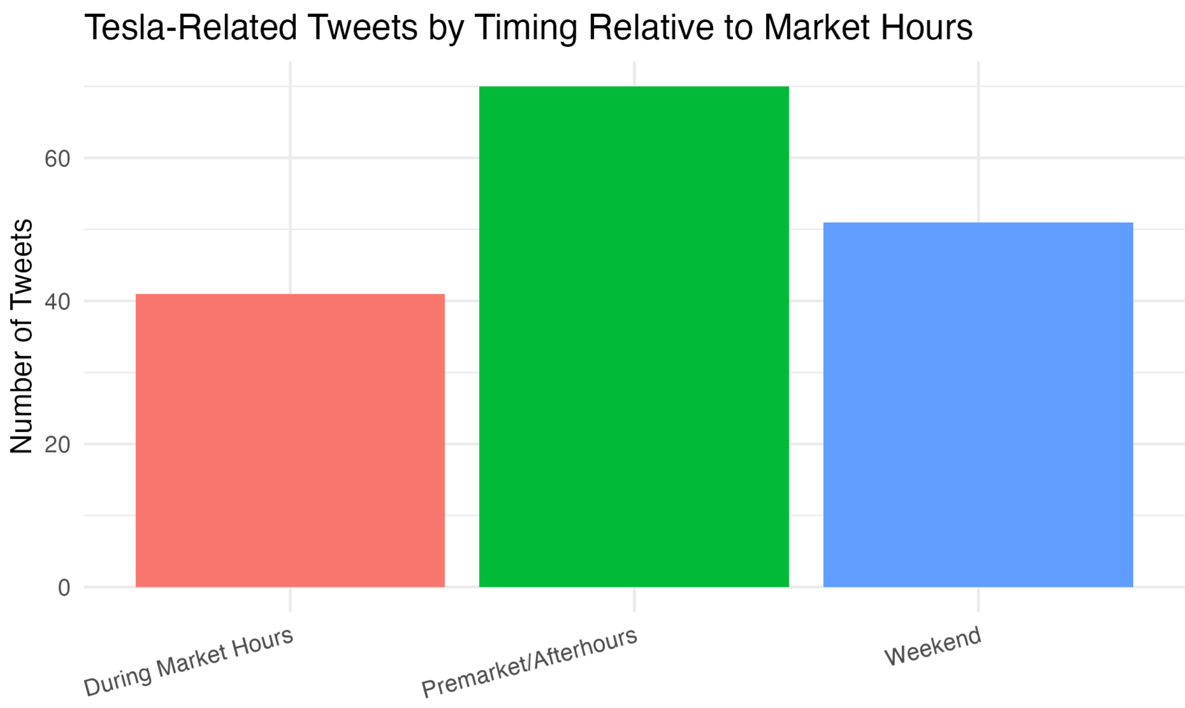
```
  mutate(
    createdAt_et = with_tz(ymd_hms(createdAt_et_str), "America/New_York"),
    date_et = as.Date(createdAt_et)
  )

# ------------------------------------------------------------------------
# PLOT 1: Daily tweet counts by category (Tesla / Non-Tesla / Link-Only)
# ------------------------------------------------------------------------
p_daily <- tweets_clean %>%
  count(date_et, tweet_category) %>%
  ggplot(aes(x = date_et, y = n, fill = tweet_category)) +
  geom_col() +
  labs(
    title = "Daily Count of Elon Musk Tweets by Category",
    x = "Date (ET)",
    y = "Number of Tweets",
    fill = "Tweet Category"
  ) +
  theme_minimal()

print(p_daily)

# ------------------------------------------------------------------------
# PLOT 2: Sentiment distribution (Tesla-related tweets)
# ------------------------------------------------------------------------
p_sentiment <- tweets_tesla_sent %>%
  ggplot(aes(x = sentiment_score, fill = sentiment_label)) +
  geom_histogram(bins = 20, alpha = 0.8, position = "identity") +
  labs(
    title = "Sentiment Score Distribution (Tesla-Related Tweets)",
    x = "Sentiment Score",
    y = "Number of Tweets",
```

```
    fill = "Sentiment Label"
  ) +
  theme_minimal()

print(p_sentiment)

# -----------------------------------------------------------------------
# PLOT 3: Tweet timing relative to market hours
# -----------------------------------------------------------------------
p_time_status <- tweets_tesla_sent %>%
  count(time_status) %>%
  ggplot(aes(x = time_status, y = n, fill = time_status)) +
  geom_col(show.legend = FALSE) +
  labs(
    title = "Tesla-Related Tweets by Timing Relative to Market Hours",
    x = "",
    y = "Number of Tweets"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 15, hjust = 1))

print(p_time_status)


# Save plots for Quarto report

# -----------------------------------------------------------------------
# SAVE FIGURES TO figs/ FOLDER
# -----------------------------------------------------------------------

# Save EDA plots
ggsave("figs/eda_daily_counts.png", p_daily,
```

```
       width = 16, height = 10, units = "cm", dpi = 300)

ggsave("figs/eda_sentiment_distribution.png", p_sentiment,
       width = 16, height = 10, units = "cm", dpi = 300)

ggsave("figs/eda_time_status.png", p_time_status,
       width = 16, height = 10, units = "cm", dpi = 300)

##############################################################
# End of 04_plot_EDA.R
##############################################################
```







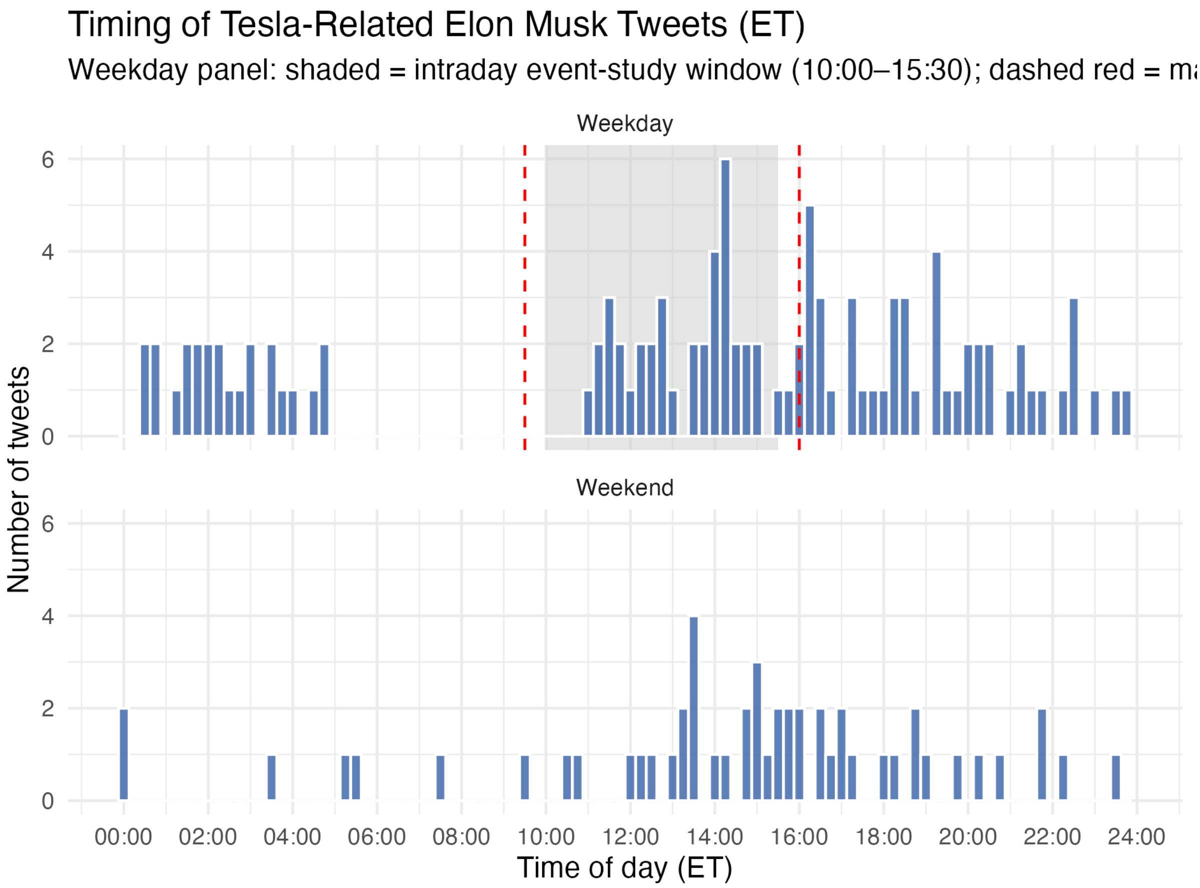

```
##############################################################
# B_tweet_timing_historgram.R
# Exploratory Data Analysis (EDA) plots for Musk tweet data
# Timing of Tesla-Related Elon Musk Tweets (ET)
#
```

```
# Uses:
#   - data/musk_tesla_tweets_clean.csv
############################################################

library(tidyverse)
library(lubridate)

# Load cleaned Tesla tweets
tweets_tesla <- read_csv(
  "data/musk_tesla_tweets_clean.csv",
  col_types = cols(id = col_character())
)

# Rebuild ET time from stable string (offset preserved in CSV)
# (Never trust createdAt_et from CSV; it may be written in UTC)
tweets_tesla <- tweets_tesla %>%
  mutate(
    createdAt_et = with_tz(ymd_hms(createdAt_et_str), "America/New_York")
  )

# Prepare time-of-day variable (decimal hours)
tweets_tesla <- tweets_tesla %>%
  mutate(
    hour_decimal = hour(createdAt_et) + minute(createdAt_et) / 60,

    # Tag weekend vs weekday so we can show them separately
    day_type = if_else(wday(createdAt_et) %in% c(1, 7), "Weekend", "Weekday")
  )

# Data used ONLY for weekday annotations so they do not appear in the Weekend panel
weekday_annot_data <- tibble(day_type = "Weekday")
```

```
# Plot ---------------------------------------------------
p_time <- ggplot(tweets_tesla, aes(x = hour_decimal)) +

  # Shaded intraday event-study window ONLY for Weekday: 10:00-15:30
  geom_rect(
    data = weekday_annot_data,
    aes(xmin = 10, xmax = 15.5, ymin = -Inf, ymax = Inf),
    inherit.aes = FALSE,
    fill = "grey80",
    alpha = 0.5
  ) +

  # Histogram of tweet times
  geom_histogram(
    binwidth = 0.25,        # 15-minute bins
    fill = "#4C72B0",
    color = "white",
    alpha = 0.9
  ) +

  # Market open / close reference lines ONLY for Weekday
  geom_vline(
    data = weekday_annot_data,
    aes(xintercept = 9.5),
    inherit.aes = FALSE,
    linetype = "dashed",
    color = "red"
  ) +
  geom_vline(
    data = weekday_annot_data,
    aes(xintercept = 16),
    inherit.aes = FALSE,
```

```
    linetype = "dashed",
    color = "red"
  ) +

  scale_x_continuous(
    breaks = seq(0, 24, by = 2),
    labels = function(x) sprintf("%02d:00", x)
  ) +

  labs(
    title = "Timing of Tesla-Related Elon Musk Tweets (ET)",
    subtitle = "Weekday panel: shaded = intraday event-study window (10:00-15:30); da
    x = "Time of day (ET)",
    y = "Number of tweets"
  ) +

  # Weekday vs weekend panels
  facet_wrap(~day_type, ncol = 1) +

  theme_minimal()

print(p_time)

# ---------------------------------------------------------------------------

# Save plots for Quarto report
# ---------------------------------------------------------------------------

# Save timing histogram plot
ggsave("figs/eda_tweet_timing_histogram.png", p_time,
       width = 16, height = 12, units = "cm", dpi = 300)
```

Timing Histogram (Weekday vs Weekend)


Timing of Tesla-Related Elon Musk Tweets (ET)
Weekday panel: shaded = intraday event-study window (10:00–15:30); dashed red = ma
Weekday
Weekend
Number of tweets
0
2
4
6
00:00
02:00
04:00
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00
22:00
24:00
Time of day (ET)