#DCprotests

JordanJasuta

6/2/2020

This very basic analysis was conducted on tweets using the hashtag #dcprotests in the month of May 2020. Tweets were scrapped using: https://github.com/twintproject/twint and cleaned using the jupyter notebook `Twitter.ipynb` to produce `processed_tweets.csv`. Major analysis inspiration was drawn from https://datavizm20.classes.andrewheiss.com/example/13-example/#complete-code

** *A MAJOR DISCLAIMER: this analysis is neither exhaustive nor statistically significant. Yes, twitter includes tweets from trolls and bots. Yes, the voices of more active tweeters are awarded disproportionate weight. However, this preliminary analysis can still provide important insight into the activity on the ground in Washington, DC this week, where official news and data sources have been frustratingly unhelpful.* **

Load data and drop tweets prior to 1 May 2020 (flukes unassociated with the current social movement)

```r
tweets <- read_csv('processed_tweets_jun2.csv')
```

```
## Parsed with column specification:
## cols(
##   tweets = col_character(),
##   metadata = col_character(),
##   username = col_character(),
##   text = col_character(),
##   id = col_double(),
##   date = col_date(format = ""),
##   time = col_datetime(format = ""),
##   timezone = col_character()
## )
```

```r
#drop rows from before May
tweets <- tweets %>% filter(date >= "2020-05-01")    # 651 rows lost (6.5%)
```

The timeline

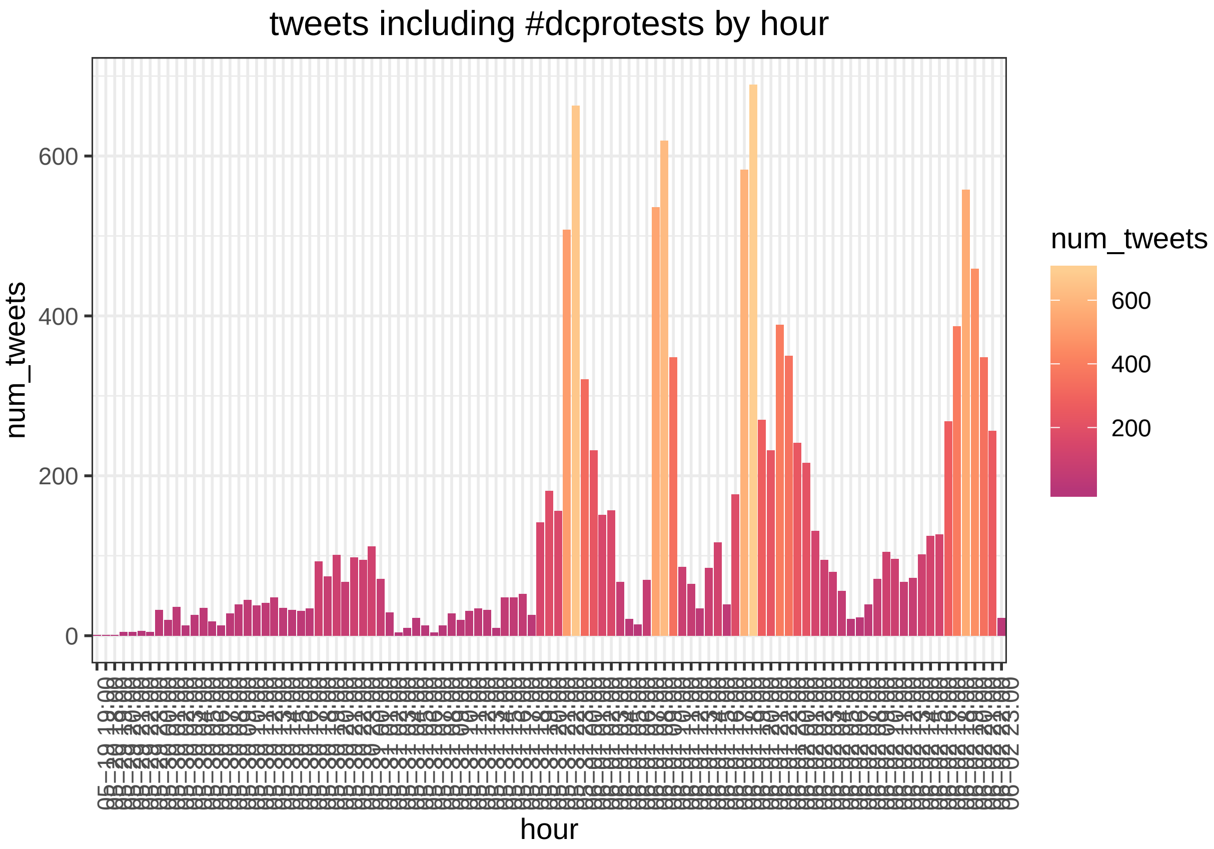
Start by plotting #dcprotests tweets by hour since 29 May 2020…

```r
# create hourly time variable
tweets$mdh <- paste(format(tweets$date, "%m-%d"),format(tweets$time, "%H:00"))

#create data table by hour and day
tweets_over_time <- as.data.frame(table(tweets$mdh))
colnames(tweets_over_time) <- c("hour", "num_tweets")

# plot
```

```
ggplot(tweets_over_time,
       aes(x = hour, y = num_tweets, fill = num_tweets)) +
  geom_col() +
  scale_fill_viridis_c(option = "magma", begin = 0.5, end = 0.9) +
  theme_bw() +
  ggtitle('tweets including #dcprotests by hour') +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



Notice spikes in tweeting activity at 22:00 (10pm) on Sunday, May 31st - this was right before the city-wide curfew started at 11pm. As the night wears on activity falls, only to rise again at 8am when people are waking up to check the news, their surroundings, social media, and messages from friends/family. On Monday, June 1st, the curfew was raised to 7pm, and tellingly, tweets spike even higher starting at 18:00 (6pm) on June 1st, an hour before the stricter curfew. The evening tweet surges grew more sustained over June 1st and June 2nd, showing no signs of dying down in the face of the curfew.

Perspective (as sorted by keywords)

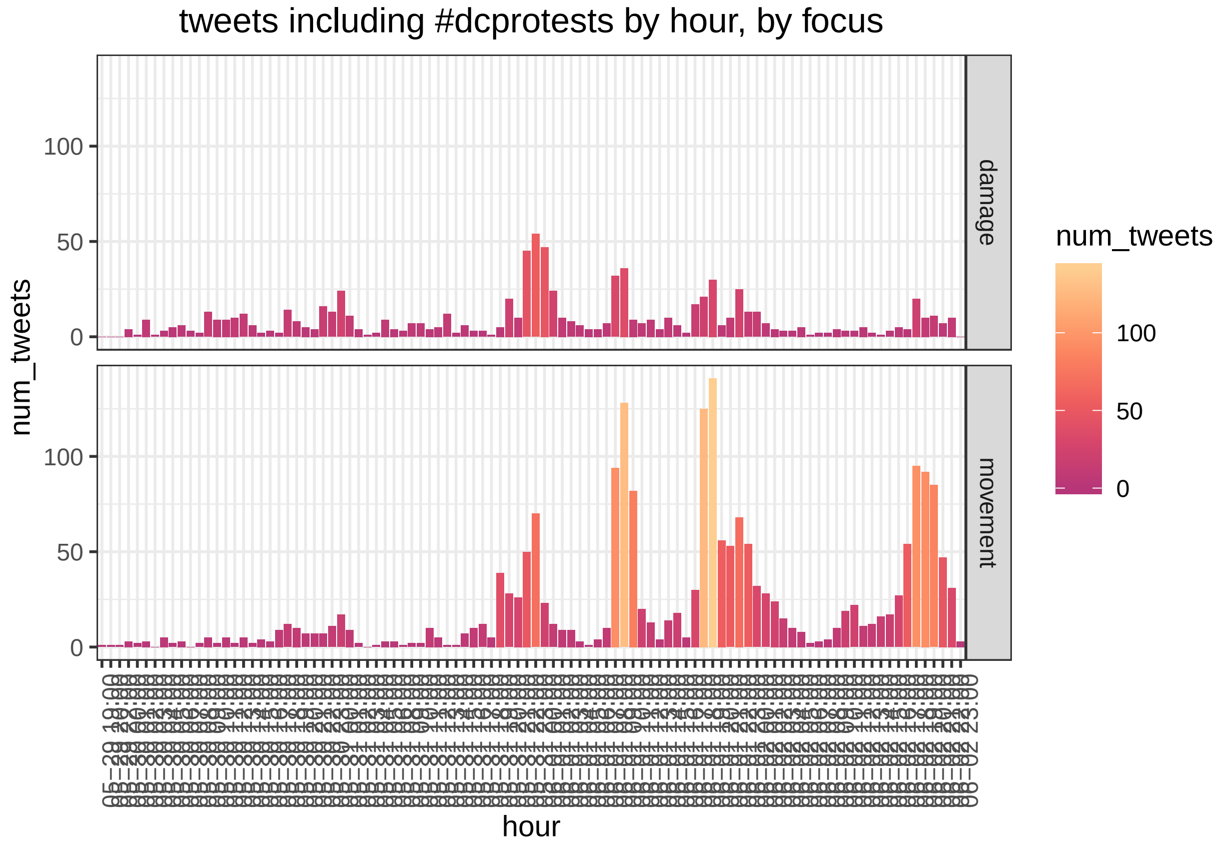
Beyond just looking at a time series of tweets, it can be useful to break down tweets by content. In this case, I wanted to look at the focus of the tweets - theoretically, they are all about the same issue, but how are twitter users percieving the activity in the capital? Are they focusing on the movement, or the collateral damage?

Ideally, I would use a more sophisticated technique to sort tweets into perspective-holding groups, such as tf-idf to extract keywords (included in the datavizm20 article) or an ML classifier. However, time is of the essense and I don't have any existing training data to feed such an analysis, so I settled for a manual skimming of tweets and logical selection of keywords. Keywords like ‘blacklivesmatter’, ‘justice’, and ‘march’ were used to tag movement-focused tweets, while keywords like ‘riots’, ‘not the way’, and ‘looting’ were used to tag damage-focused tweets. This method can obviously be improved in future iterations of this analysis.

We can now recreate the plot that showed tweets over time, now divided by perspective.

```r
tweets_by_p_over_time <- as.data.frame(table(by_perspective$mdh,by_perspective$tag))
colnames(tweets_by_p_over_time) <- c("hour", "tag", "num_tweets")

ggplot(tweets_by_p_over_time,
       aes(x = hour, y = num_tweets, fill = num_tweets)) +
  geom_col() +
  scale_fill_viridis_c(option = "magma", begin = 0.5, end = 0.9) +
  theme_bw() +
  facet_grid(tag ~ .) +
  ggtitle('tweets including #dcprotests by hour, by focus') +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```

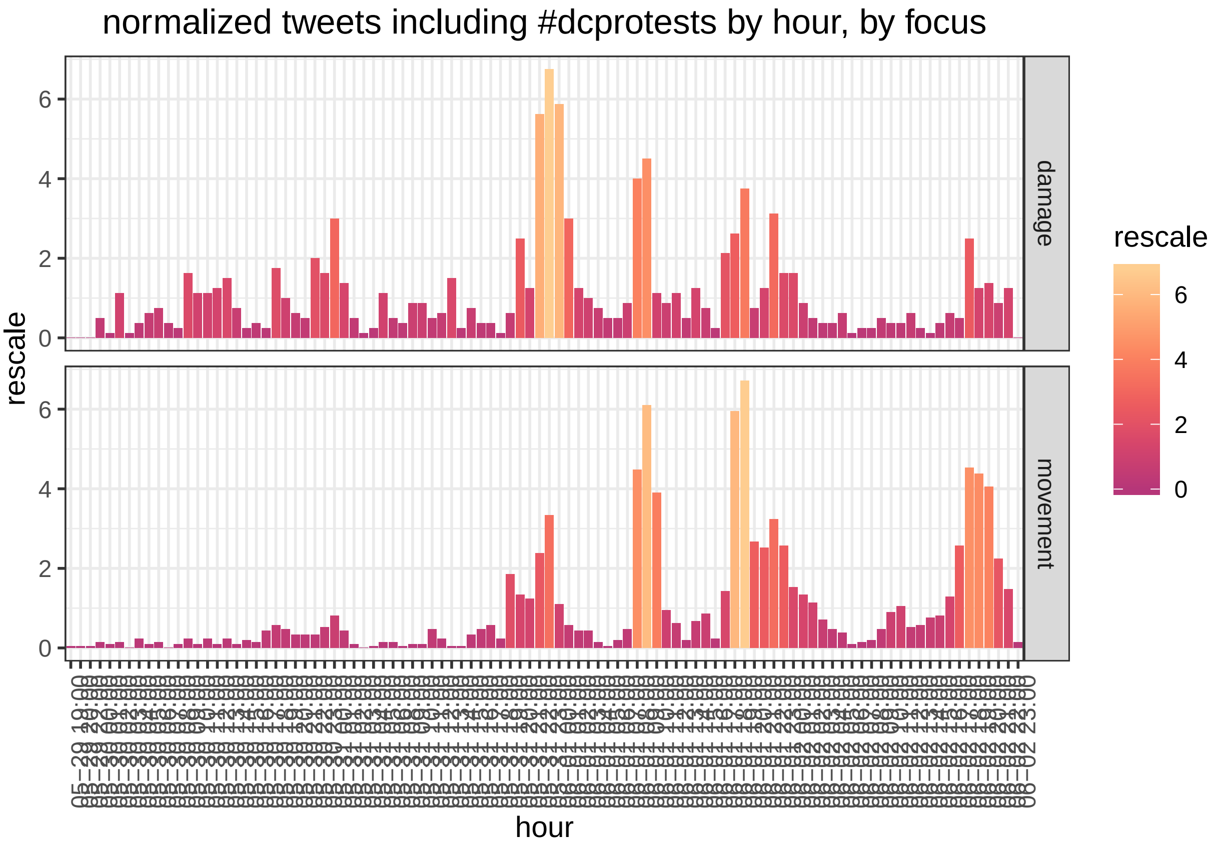


While this graphic indicates that movement-focused tweets far outnumber damage-focused tweets, this may be an artificial/misleading result of the manual keyword selection I used. To address this we can rescale the data as a proportion of the mean tweets per group.

```r
x <- tweets_by_p_over_time %>% group_by(tag) %>% summarize(sumB = mean(num_tweets))
damage <- as.integer(x[1,2])
movement <- as.integer(x[2,2])
```

```
tweets_by_p_over_time$rescale <- ifelse(tweets_by_p_over_time$tag == 'damage', tweets_by_p_over_time$n/
                                     ifelse(tweets_by_p_over_time$tag == 'movement',tweets_by_p_over_ti

ggplot(tweets_by_p_over_time,
       aes(x = hour, y = rescale, fill = rescale)) +
  geom_col() +
  scale_fill_viridis_c(option = "magma", begin = 0.5, end = 0.9) +
  theme_bw() +
  facet_grid(tag ~ .) +
  ggtitle('normalized tweets including #dcprotests by hour, by focus') +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



Damage-focused tweets showed the most activity on Sunday night (May 31st) - when most of the damage was done; followed by smaller surges at the subsequent times of peak traffic. Movement-focused tweets (talking about peaceful protest, George Floyd, etc) did show a spike on Sunday night, but soared the next morning, possibly in response to accusations of bad behavior. In the face of the 7pm curfew on Monday (June 1st), they grew even higher, insisting that peaceful protests should be allowed. Tuesday (June 2nd) showed a slightly lower but even more sustained peak, indicated that protesters and organizers may be settling in for a long fight.

Sentiment analysis

R's textdata library comes with some highly convenient pre-trained sentiment analysis packages. Let's start with a more complex look at tweet sentiment, using the 'nrc' package to classify the sentiment of words into one of 10 categories:

```r
# ALL TWEETS

by_word <- separate_rows(tweets, text, sep = " ", convert = FALSE)
perspective_by_word <- separate_rows(by_perspective, text, sep = " ", convert = FALSE)

tweet_words <- by_word %>%
  #drop_na() %>%
  unnest_tokens(word, text) %>%     # Split into word tokens
  anti_join(stop_words)          # Remove stopwords
```

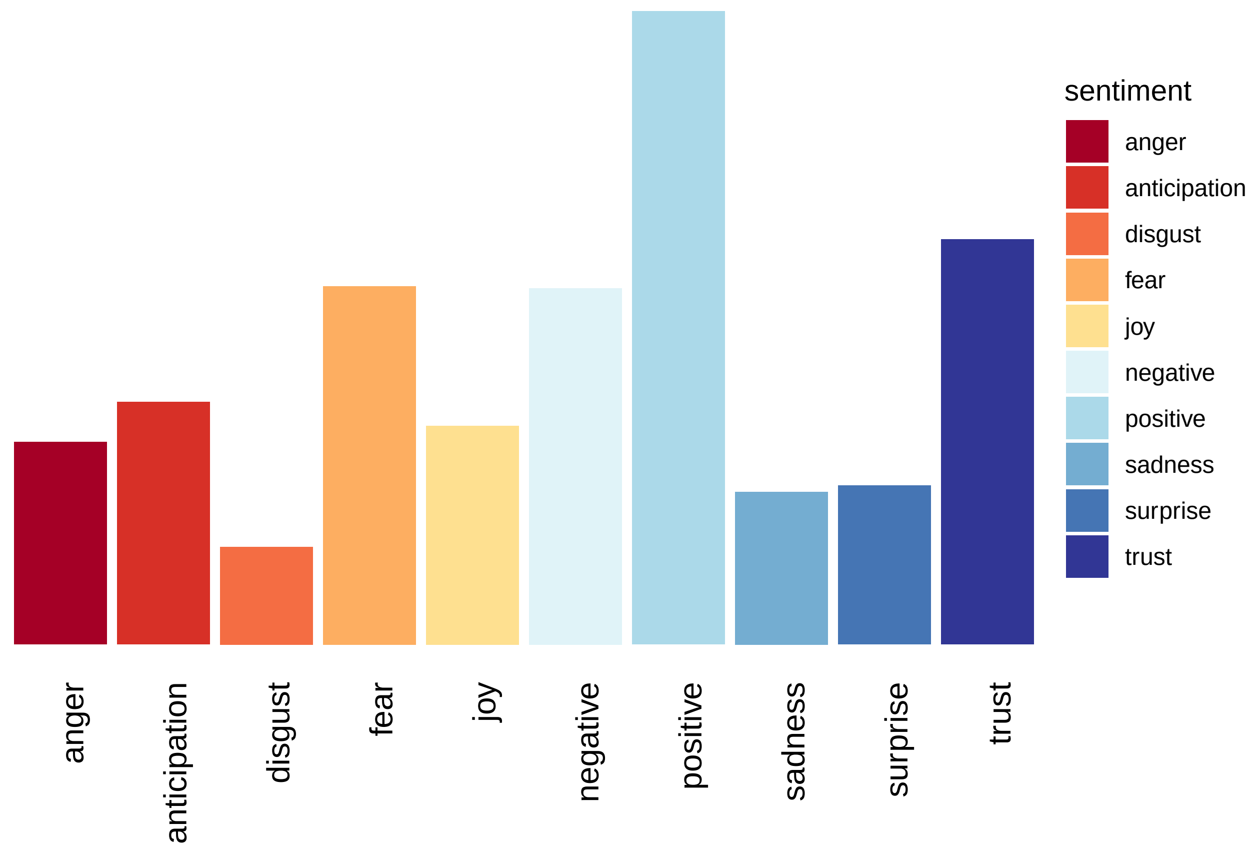
```
## Joining, by = "word"
```

```r
# join the sentiment dictionary
tweet_sentiment <- tweet_words %>%     # may need to drop trump from dataset since this word has its ow
  inner_join(get_sentiments("nrc"))
```

```
## Joining, by = "word"
```

```r
tweet_sentiment_plot <- tweet_sentiment %>%
  count(timezone, sentiment)

# plot sentiment for all tweets
library(RColorBrewer)
ggplot(tweet_sentiment_plot, aes(x = sentiment, y = n, fill = sentiment)) +
  geom_col(position = position_dodge()) +
  scale_fill_brewer(palette = 'RdYlBu') +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



While negativity, fear and anger are not insignificant, they are far outwieghed by positive sentiment, and trust is surprisingly high. This is due to a high frequency of words like 'united', 'peaceful', 'share', and 'friend', but also because the pre-trained sentiment extractor has no sense of context: 'accountability', 'justice' and even 'police' are also categorized under trust.

While this is not ideal, it's a good start. Now compare the movement-focused and damage-focused tweet sentiments:

```
# BY-PERSPECTIVE TWEETS

tweet_words_by_p <- perspective_by_word %>%
  #drop_na() %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)
```

```
## Joining, by = "word"
```

```
sent_by_persp <- tweet_words_by_p %>%
  inner_join(get_sentiments("nrc"))
```

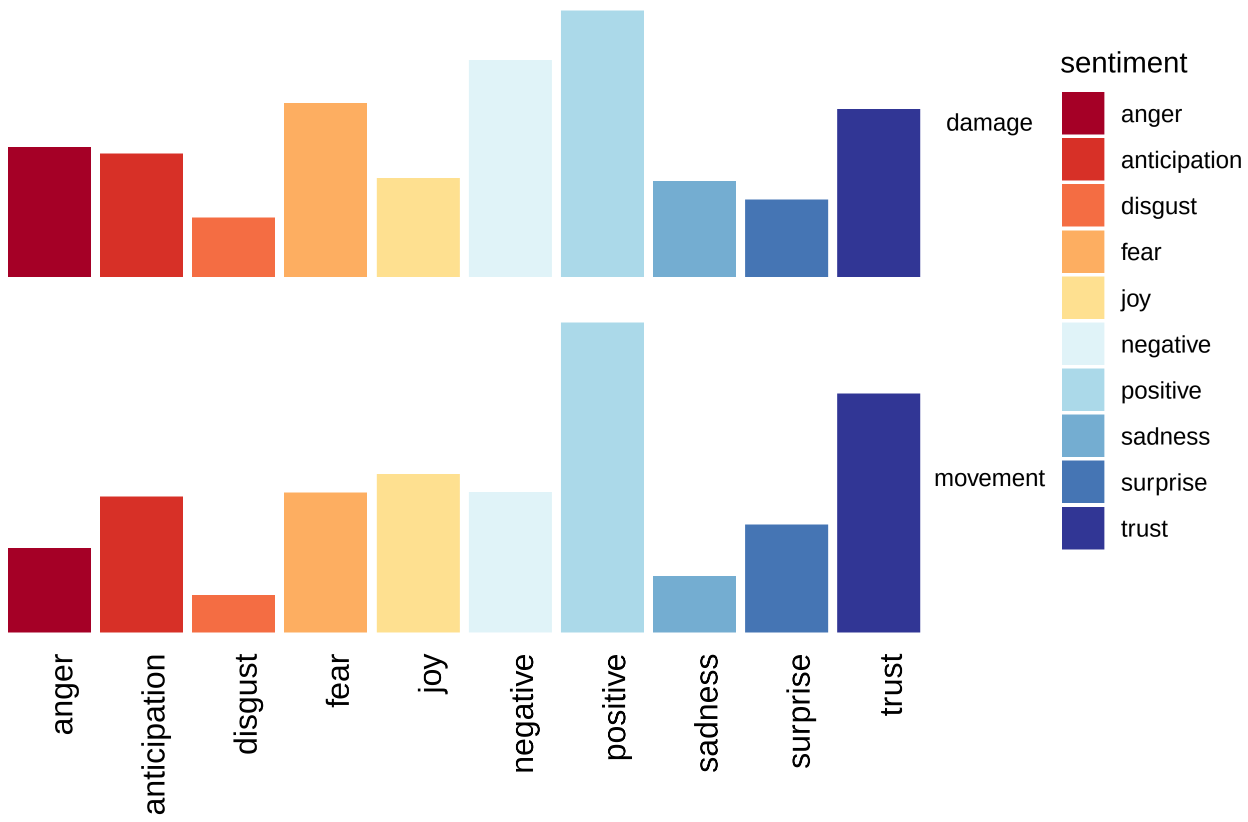
```
## Joining, by = "word"
```

```
sent_by_persp_plot <- sent_by_persp %>%
  count(tag, sentiment)
```

```
# normalize as a % for ease of visual compsrison
x <- sent_by_persp_plot %>% group_by(tag) %>% summarize(sumB = sum(n))
damage <- as.integer(x[1,2])
movement <- as.integer(x[2,2])

sent_by_persp_plot$percent <- ifelse(sent_by_persp_plot$tag == 'damage', sent_by_persp_plot$n/damage,
                                     ifelse(sent_by_persp_plot$tag == 'movement',sent_by_persp_plot$n/m

# plot sentiment of tweets by perspective groups
ggplot(sent_by_persp_plot, aes(x = sentiment, y = percent, fill = sentiment)) +
  geom_col(position = position_dodge()) +
  facet_grid(tag ~ .) +
  scale_fill_brewer(palette = 'RdYlBu') +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



We can see predictable differences like more fear and anger in the damage-focused tweets and more positivity and trust in the movement-focused tweets.

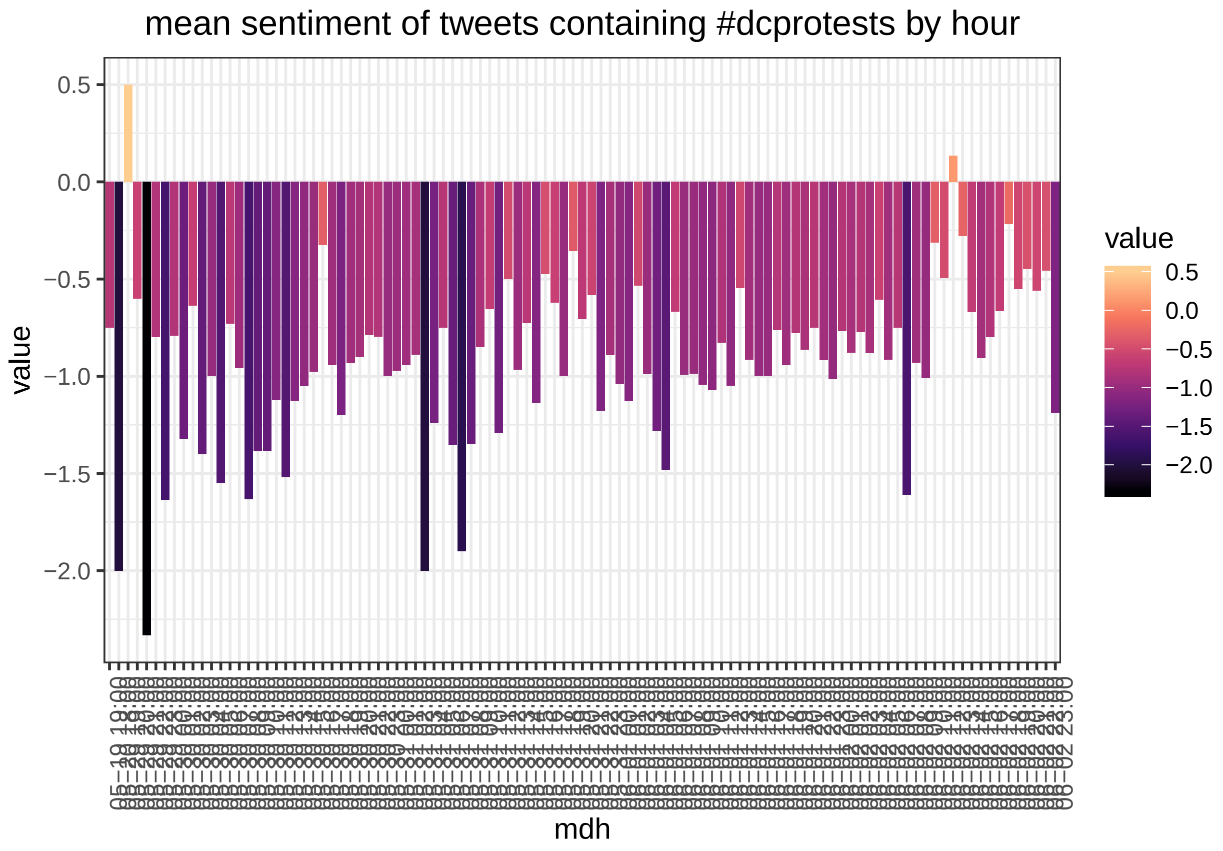
To show overall sentiment over time, we need to reduce our analysis to 2 dimensions (positive and negative).

```
# use 'afinn' package to assign sentiment values
polar_sentiment <- tweet_words %>%     # may need to drop trump from dataset since this word has its ow
  inner_join(get_sentiments("afinn"))
```

```
## Joining, by = "word"
```

```r
# group by hour
sentiment_by_hour <- polar_sentiment %>%
  group_by(mdh) %>%
  summarise(value = mean(value))

#plot mean sentiment for all tweets
ggplot(sentiment_by_hour,
       aes(x = mdh, y = value, fill = value)) +
  geom_col() +
  scale_fill_viridis_c(option = "magma", end = 0.9) +
  theme_bw() +
  ggtitle('mean sentiment of tweets containing #dcprotests by hour') +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



Of course, given the nature of the subject at hand, sentiment will be overwhelmingly negative. However, we can plot degree of negativity at different times and correlate these changes with the events on the ground.

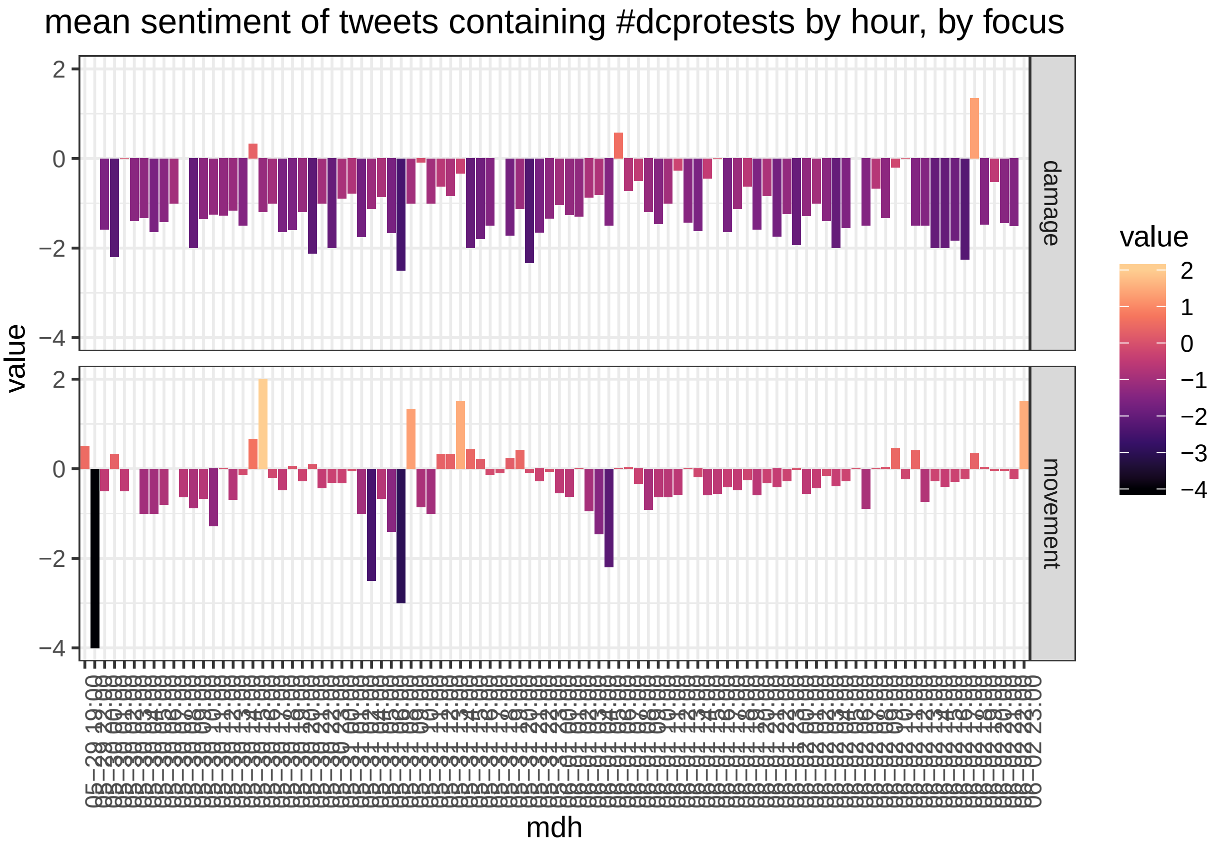
The same can again be done by perspective:

```r
polar_sent_by_p <- tweet_words_by_p %>%
  inner_join(get_sentiments("afinn"))
```

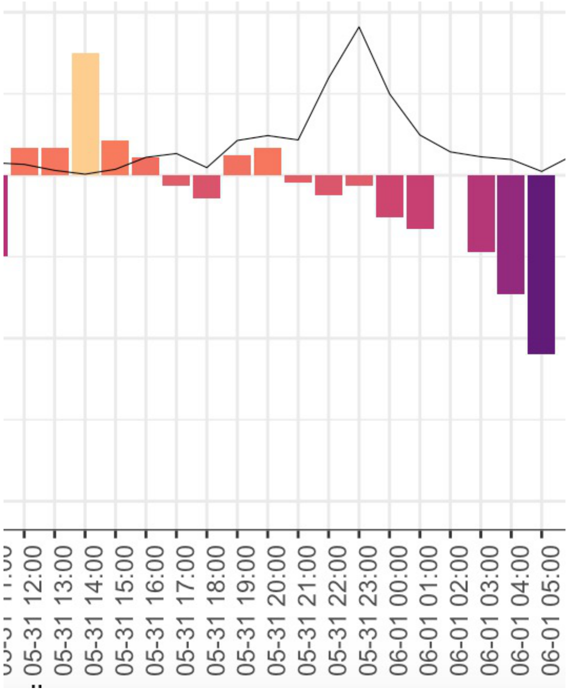
```
## Joining, by = "word"
```

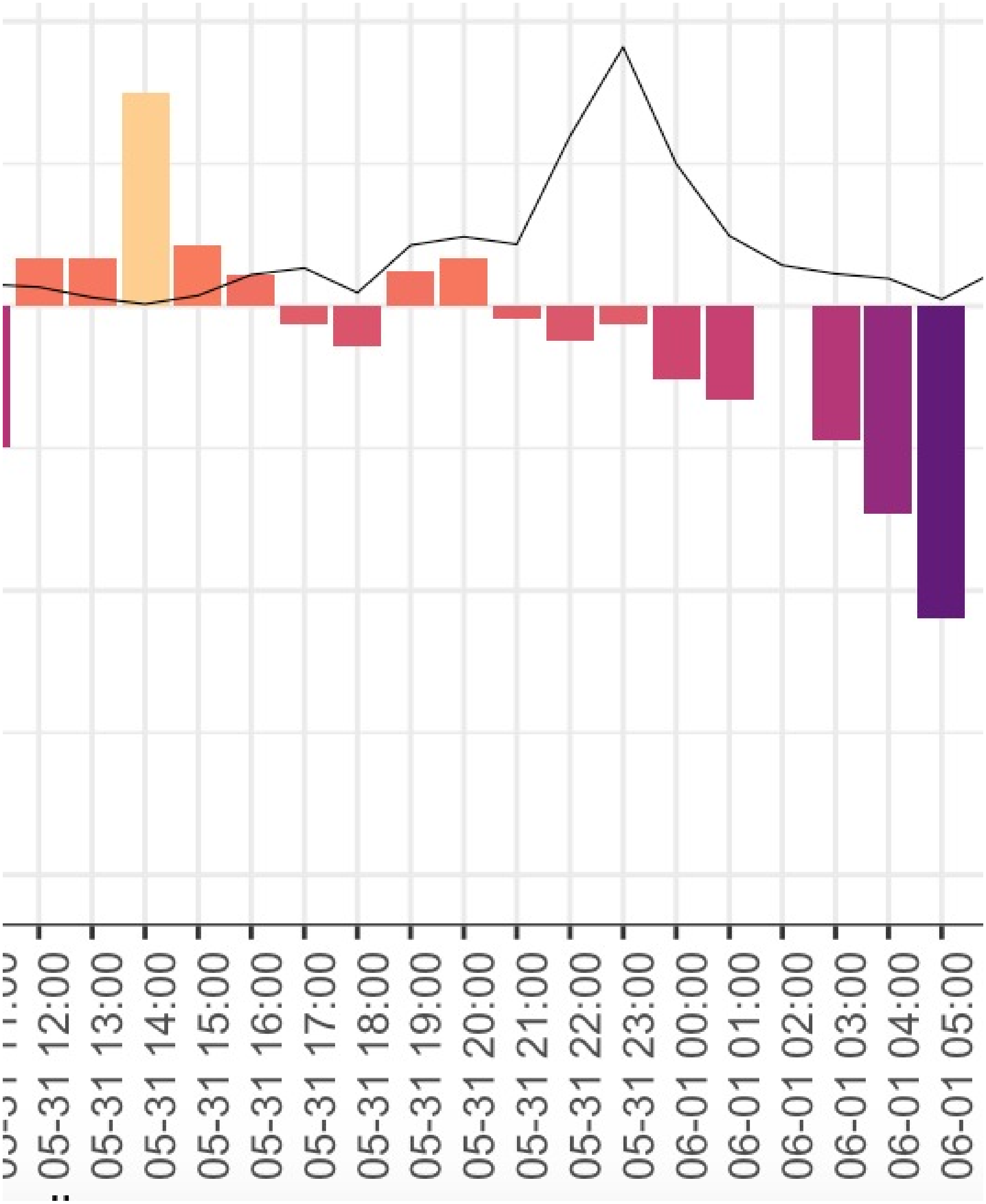
```
# group by hour
sent_by_hour_by_p <- polar_sent_by_p %>%
  group_by(mdh, tag) %>%
  summarise(value = mean(value))


ggplot(sent_by_hour_by_p,
       aes(x = mdh, y = value, fill = value)) +
  geom_col() +
  scale_fill_viridis_c(option = "magma", end = 0.9) +
  facet_grid(tag ~ .) +
  theme_bw() +
  ggtitle('mean sentiment of tweets containing #dcprotests by hour, by focus') +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



It isn't surprising to see that the overall sentiment of damage-focused tweets is more negative than the movement-focused equivalent. Fledgling positivity peaks can be seen mid-afternoon on May 30th (Saturday night) and the daytime of May 31st (Sunday). Before Sunday night, some reports of property damages had surfaced, but the marches were widely observed to be peaceful. However, the steady decline in positivity is cleary visible over the course of Sunday night, as peaceful protests gave way to clashes with police:


05-31 11:00
05-31 12:00
05-31 13:00
05-31 14:00
05-31 15:00
05-31 16:00
05-31 17:00
05-31 18:00
05-31 19:00
05-31 20:00
05-31 21:00
05-31 22:00
05-31 23:00
06-01 00:00
06-01 01:00
06-01 02:00
06-01 03:00
06-01 04:00
06-01 05:00

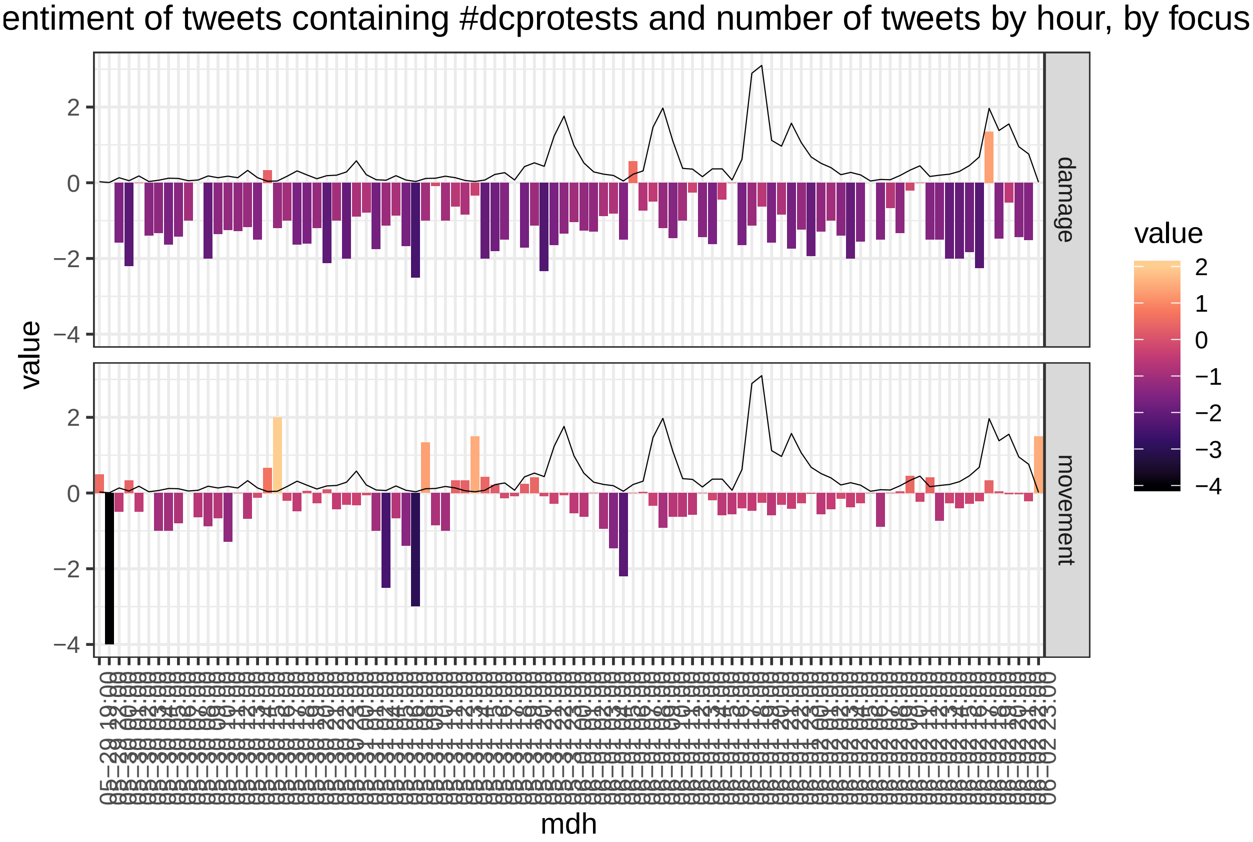


Nonetheless, the movement-focused tweets returned to a slightly negative - but still much more positive than damage-focused tweets - tone over the next two nights of protest. The 'settling' referenced earlier is also visible here. As the sentiment of movement-focused tweets remains steady over June 1st and 2nd.

For reference, we can also compare these to the number of tweets sent at each hour:

```r
num_tweets_hr_by_p <- polar_sent_by_p %>%
  count(timezone, mdh)

ggplot() +
  geom_col(data = sent_by_hour_by_p,
           aes(x = mdh, y = value, fill = value)) +
  scale_fill_viridis_c(option = "magma", end = 0.9) +
  facet_grid(tag ~ .) +
  theme_bw() +
  geom_line(data = num_tweets_hr_by_p, aes(x = mdh, y = n/150, group = 1), lwd=0.2) +
  ggtitle('mean sentiment of tweets containing #dcprotests and number of tweets by hour, by focus') +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x = element_text(angle = 90, hjust = 1))
```



Note that peaks usually occur when there aren't that many tweets - averages at play.