# R Notebook

## Nicola Sturgeon's stronger sentiment behind Scotts' Covid19 compliance?

Twitter account sentiment analysis for the four leaders of the devolved nations shows higher levels of sentiment expression by Stergeon.

r example ref: https://www.tidytextmining.com/sentiment.html

higher levels of compliance source (not verified): https://www.thenational.scot/news/18789273.covid-scotland-compliance-new-lockdown-rules-good/

```
# install library
# install.packages("rtweet")
# install.packages("tidytext")
# install.packages("textdata")

# load library
library(rtweet)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

```
library(tidyr)
library(tidytext)
library(ggplot2)
```

Get timeline tweets since 1/8/2020. tidy to screen_name,date,text

```
leaders <- c("@BorisJohnson", "@NicolaSturgeon", "@fmwales", "@DUPleader")
tw <- get_timeline(leaders, n=1000) %>%
  select(screen_name, created_at, text) %>%
  filter(as.Date(created_at) >= as.Date("2020-08-01"))
tw$date <- as.Date(tw$created_at)
tw <- tw %>% select(screen_name, date, text)
```

Clean up the text column and arrive at words

```
tw$clean_text <- gsub("http\\S+", "", tw$text)
tw_words <- tw %>% select(screen_name, date, clean_text) %>%
  unnest_tokens(word, clean_text) %>%
  anti_join(stop_words)
```

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

Positive/negative sentiment analysis using bing wordlist

```
tw_sent <- tw_words %>%
  inner_join(get_sentiments("bing")) %>%
  count(screen_name, index = date, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(sentiment = positive - negative)
```

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

Creating a plot by screen name

```
ggplot(tw_sent, aes(index, sentiment, fill = screen_name)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~screen_name, ncol = 2, scales = "free_x")
```