srcipt_rmd

Kaitlin P. Ward

12/12/2020

Pull data from CSV files

```
#read in data
subm_list_dads_2019 <- read.csv("subm_list_dads_2019.csv") #dads 2019 posts data
subm_list_dads_2020 <- read.csv("subm_list_dads_2020.csv") #dads 2020 posts data
subm_list_moms_2019 <- read.csv("subm_list_moms_2019.csv") #moms 2019 posts data
subm_list_moms_2020 <- read.csv("subm_list_moms_2020.csv") #moms 2020 posts data

#create dataframes with data and post (title+content) / convert date from UTC
dads2019 <- data.frame("date" = as.POSIXct(subm_list_dads_2019$created_utc, origin='1970-01-01'),
                       "post" = paste(subm_list_dads_2019$title, subm_list_dads_2019$selftext))
dads2020 <- data.frame("date" = as.POSIXct(subm_list_dads_2020$created_utc, origin='1970-01-01'),
                       "post" = paste(subm_list_dads_2020$title, subm_list_dads_2020$selftext))
moms2019 <- data.frame("date" = as.POSIXct(subm_list_moms_2019$created_utc, origin='1970-01-01'),
                       "post" = paste(subm_list_moms_2019$title, subm_list_moms_2019$selftext))
moms2020 <- data.frame("date" = as.POSIXct(subm_list_moms_2020$created_utc, origin='1970-01-01'),
                       "post" = paste(subm_list_moms_2020$title, subm_list_moms_2020$selftext))
```

create and clean corpus for moms/dads and 2019/2020

```
#clean corpus function
clean_corpus <- function(corpus) {
  # replace numbers with text
  corpus <- tm_map(corpus, content_transformer(removeNumbers))
  # Remove punctuation
  corpus <- tm_map(corpus, removePunctuation)
  # Transform to lower case
  corpus <- tm_map(corpus, content_transformer(tolower))
  # Add more stopwords
  corpus <- tm_map(corpus, removeWords, words = c(stopwords("en")))
  # Strip whitespace
  corpus <- tm_map(corpus,stripWhitespace)
  return(corpus)
}

#create corpus and clean corpus
dads2019_source <- VectorSource(dads2019$post)
dads2019_corpus <- VCorpus(dads2019_source)
dads2019_corpus <- clean_corpus(dads2019_corpus)
```

```
dads2020_source <- VectorSource(dads2020$post)
dads2020_corpus <- VCorpus(dads2020_source)
dads2020_corpus <- clean_corpus(dads2020_corpus)

moms2019_source <- VectorSource(moms2019$post)
moms2019_corpus <- VCorpus(moms2019_source)
moms2019_corpus <- clean_corpus(moms2019_corpus)

moms2020_source <- VectorSource(moms2020$post)
moms2020_corpus <- VCorpus(moms2020_source)
moms2020_corpus <- clean_corpus(moms2020_corpus)
```

polarity analysis

```
d19_pol <- polarity(dads2019$post)
```

```
## Warning in polarity(dads2019$post):
##   Some rows contain double punctuation.  Suggested use of `sentSplit` function.
```

```
d20_pol <- polarity(dads2020$post)
```

```
## Warning in polarity(dads2020$post):
##   Some rows contain double punctuation.  Suggested use of `sentSplit` function.
```

```
m19_pol <- polarity(moms2019$post)
```

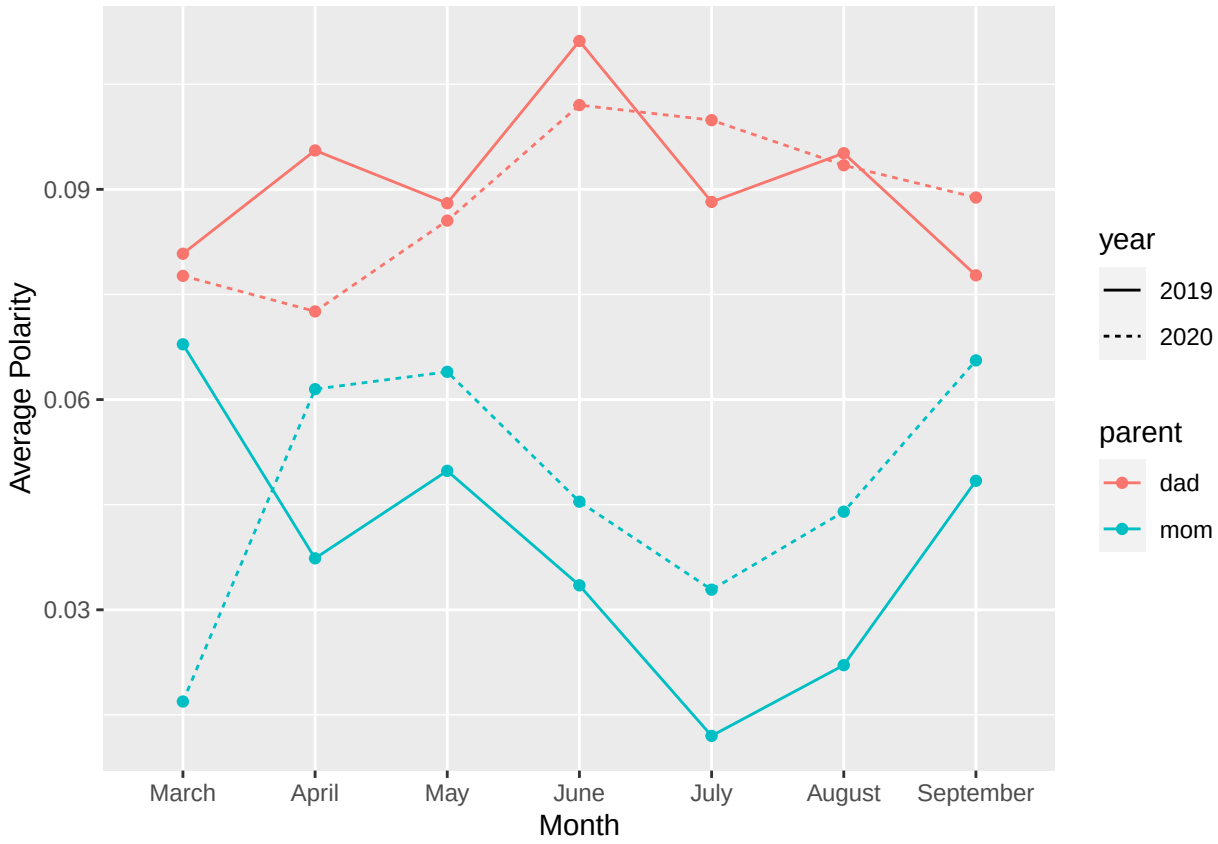
```
## Warning in polarity(moms2019$post):
##   Some rows contain double punctuation.  Suggested use of `sentSplit` function.
```

```
m20_pol <- polarity(moms2020$post)
```

```
## Warning in polarity(moms2020$post):
##   Some rows contain double punctuation.  Suggested use of `sentSplit` function.
```

```
dads2019 <- dads2019 %>%
  mutate(pol = d19_pol$all$polarity) %>%
  mutate(month = month(date)) %>%
  mutate(year = year(date)) %>%
  mutate(parent = factor("dad"))

dads2020 <- dads2020 %>%
  mutate(pol = d20_pol$all$polarity) %>%
  mutate(month = month(date))%>%
  mutate(year = year(date))%>%
  mutate(parent = factor("dad"))

moms2019 <- moms2019 %>%
  mutate(pol = m19_pol$all$polarity) %>%
  mutate(month = month(date)) %>%
  mutate(year = year(date)) %>%
  mutate(parent = factor("mom"))

moms2020 <- moms2020 %>%
  mutate(pol = m20_pol$all$polarity) %>%
  mutate(month = month(date))%>%
```

```
  mutate(year = year(date))  %>%
  mutate(parent = factor("mom"))


m_dads2019 <- dads2019 %>%
  mutate(month = factor(month)) %>%
  group_by(month, year, parent) %>%
  summarise(avg_pol = mean(pol, na.rm = TRUE))%>%
  mutate(year = factor(year))

m_dads2020 <- dads2020 %>%
  mutate(month = factor(month)) %>%
  group_by(month,year,parent) %>%
  summarise(avg_pol = mean(pol, na.rm = TRUE)) %>%
  mutate(year = factor(year))

m_moms2019 <- moms2019 %>%
  mutate(month = factor(month)) %>%
  group_by(month, year,parent) %>%
  summarise(avg_pol = mean(pol, na.rm = TRUE))%>%
  mutate(year = factor(year))

m_moms2020 <- moms2020 %>%
  mutate(month = factor(month)) %>%
  group_by(month,year,parent) %>%
  summarise(avg_pol = mean(pol, na.rm = TRUE)) %>%
  mutate(year = factor(year))


m_dads <- rbind(m_dads2019, m_dads2020)
m_moms <- rbind(m_moms2019, m_moms2020)
m_parent <- rbind(m_dads,m_moms)

ggplot(m_parent, aes(x = month, y=avg_pol, group = year:parent, color = parent))  + geom_point() + geom
  ylab("Average Polarity") +
  xlab("Month") +
  scale_x_discrete(breaks = c(3,4,5,6,7,8,9),
                   labels=c("March", "April", "May", "June","July","August","September"))
```



t test on polarity

t test moms 2019 vs 2020

```
### run moms 2020 vs 2019
t_moms2020 <- moms2020 %>%
  mutate(d_year = ifelse(year==2020,1,0))%>%
  mutate(d_year = factor(d_year))
t_moms2019 <- moms2019 %>%
  mutate(d_year = ifelse(year==2020,1,0)) %>%
  mutate(d_year = factor(d_year))
t_moms <- rbind(t_moms2019, t_moms2020)
moms_t_results <- t.test(t_moms$pol~t_moms$d_year)
moms_t_results
```

```
##
##  Welch Two Sample t-test
##
## data:  t_moms$pol by t_moms$d_year
## t = -1.8043, df = 7067.7, p-value = 0.07123
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.025504211  0.001057189
## sample estimates:
## mean in group 0 mean in group 1
##      0.03561081      0.04783432
```

t test dads 2019 vs 2020

```
## Run dads 2020 vs 2019
t_dads2020 <- dads2020 %>%
  mutate(d_year = ifelse(year==2020,1,0))%>%
  mutate(d_year = factor(d_year))
t_dads2019 <- dads2019 %>%
  mutate(d_year = ifelse(year==2020,1,0)) %>%
  mutate(d_year = factor(d_year))
t_dads <- rbind(t_dads2019, t_dads2020)
dads_t_results <- t.test(t_dads$pol~t_dads$d_year)
dads_t_results
```

```
##
##  Welch Two Sample t-test
##
## data:  t_dads$pol by t_dads$d_year
## t = 0.75601, df = 14673, p-value = 0.4497
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.005719262  0.012901029
## sample estimates:
## mean in group 0 mean in group 1
##      0.09303856      0.08944768
```

t test moms vs dad 2020

```
t_parent2020 <- rbind(t_dads,t_moms) %>%
  filter(year==2020)
parents2020_t_results <- t.test(t_parent2020$pol~t_parent2020$parent)
parents2020_t_results
```

```
##
##  Welch Two Sample t-test
##
## data:  t_parent2020$pol by t_parent2020$parent
## t = 7.6682, df = 9870.5, p-value = 1.909e-14
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.03097581 0.05225091
## sample estimates:
## mean in group dad mean in group mom
##        0.08944768        0.04783432
```

t test moms vs dad 2019

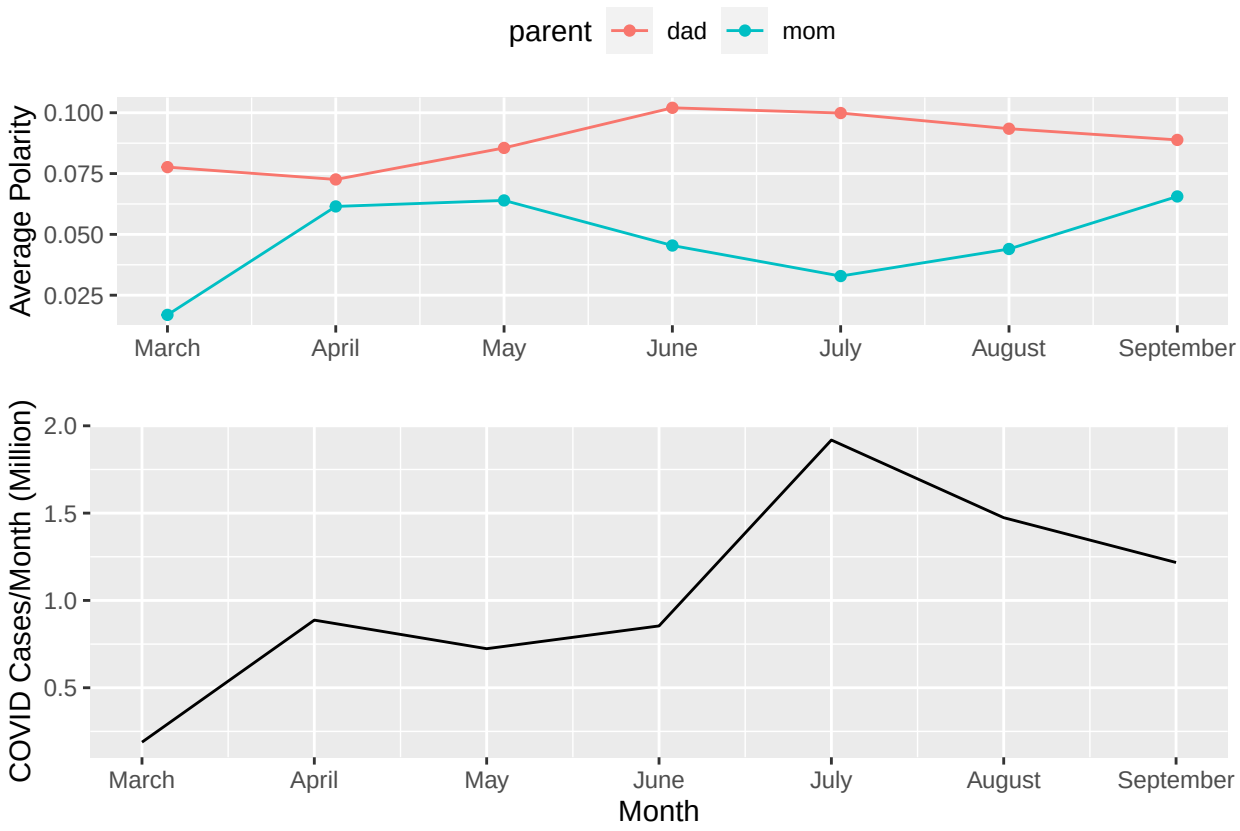
```
t_parent2019 <- rbind(t_dads,t_moms) %>%
  filter(year==2019)
parents2019_t_results <- t.test(t_parent2019$pol~t_parent2019$parent)
parents2019_t_results
```

```
##
```

```
##  Welch Two Sample t-test
##
## data:  t_parent2019$pol by t_parent2019$parent
## t = 9.1947, df = 6213.9, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.04518399 0.06967151
## sample estimates:
## mean in group dad mean in group mom
##        0.09303856        0.03561081
```

covid vs polarity analysis

```
covid <- read.csv("us.csv")
covid_month <- covid %>%
  group_by(month) %>%
  summarise(cases_month = sum(new_cases)/1000000) %>%
  filter(month>=3 & month<=9)
parent_covid <- m_parent %>%
  filter( year == 2020) %>%
  merge(covid_month) %>%
  mutate(month = as.integer(month))
g1<- ggplot(parent_covid )  + geom_line(aes(x = month, y=avg_pol,color= parent,group=parent)) + geom_po
  scale_x_continuous(breaks = c(1,2,3,4,5,6,7),
                     labels=c("March", "April", "May", "June","July","August","September")) +
  ylab("Average Polarity")+
  xlab("")+
  theme(legend.position = "top")

g2 <- ggplot(covid_month, aes(x=month, y =cases_month)) + geom_line()+
  scale_x_continuous(breaks = c(3,4,5,6,7,8,9),
                     labels=c("March", "April", "May", "June","July","August","September"))+
  ylab("COVID Cases/Month (Million)") +
  xlab("Month")#  +
#scale_y_continuous(labels = paste(ylab, "M"),breaks = ylab)

grid.arrange(g1,g2, nrow=2)
```



create create TermDocumentMatrix from each corpus - convert to tibble

```
#create tdm and transform to matrix
dads2019_tdm <- TermDocumentMatrix(dads2019_corpus)
dads2019_m <- as.matrix(dads2019_tdm)

dads2020_tdm <- TermDocumentMatrix(dads2020_corpus)
dads2020_m <- as.matrix(dads2020_tdm)

moms2019_tdm <- TermDocumentMatrix(moms2019_corpus)
moms2019_m <- as.matrix(moms2019_tdm)

moms2020_tdm <- TermDocumentMatrix(moms2020_corpus)
moms2020_m <- as.matrix(moms2020_tdm)

dads2020_tidy <- tidy(dads2020_tdm)
dads2019_tidy <- tidy(dads2019_tdm)

moms2020_tidy <- tidy(moms2020_tdm)
moms2019_tidy <- tidy(moms2019_tdm)
```

preform inner_join with nrc lexicon to complete sentiment analysis

```
nrc <-get_sentiments("nrc")
dads2020_nrc <- dads2020_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count))%>%
  # Get sentiment percentage
  mutate(average = total_count/sum(total_count)) %>%
  mutate(year = 2020)

dads2019_nrc <- dads2019_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count)) %>%
  mutate(average = total_count/sum(total_count)) %>%
  mutate(year = 2019)

moms2020_nrc <- moms2020_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count))%>%
  mutate(average = total_count/sum(total_count)) %>%
  mutate(year = 2020)

moms2019_nrc <- moms2019_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count)) %>%
  mutate(average = total_count/sum(total_count)) %>%
  mutate(year = 2019)

#join each sentiment analysis to a single data frame
dads_nrc <- rbind(dads2019_nrc,dads2020_nrc)
dads_nrc <- dads_nrc %>%
```

```
  mutate(year = factor(year)) %>%
  mutate(parent = factor("dad"))
moms_nrc <- rbind(moms2019_nrc,moms2020_nrc)
moms_nrc <- moms_nrc %>%
  mutate(year = factor(year)) %>%
  mutate(parent = factor("mom"))
parents_nrc <- rbind(dads_nrc, moms_nrc)
```

create plots

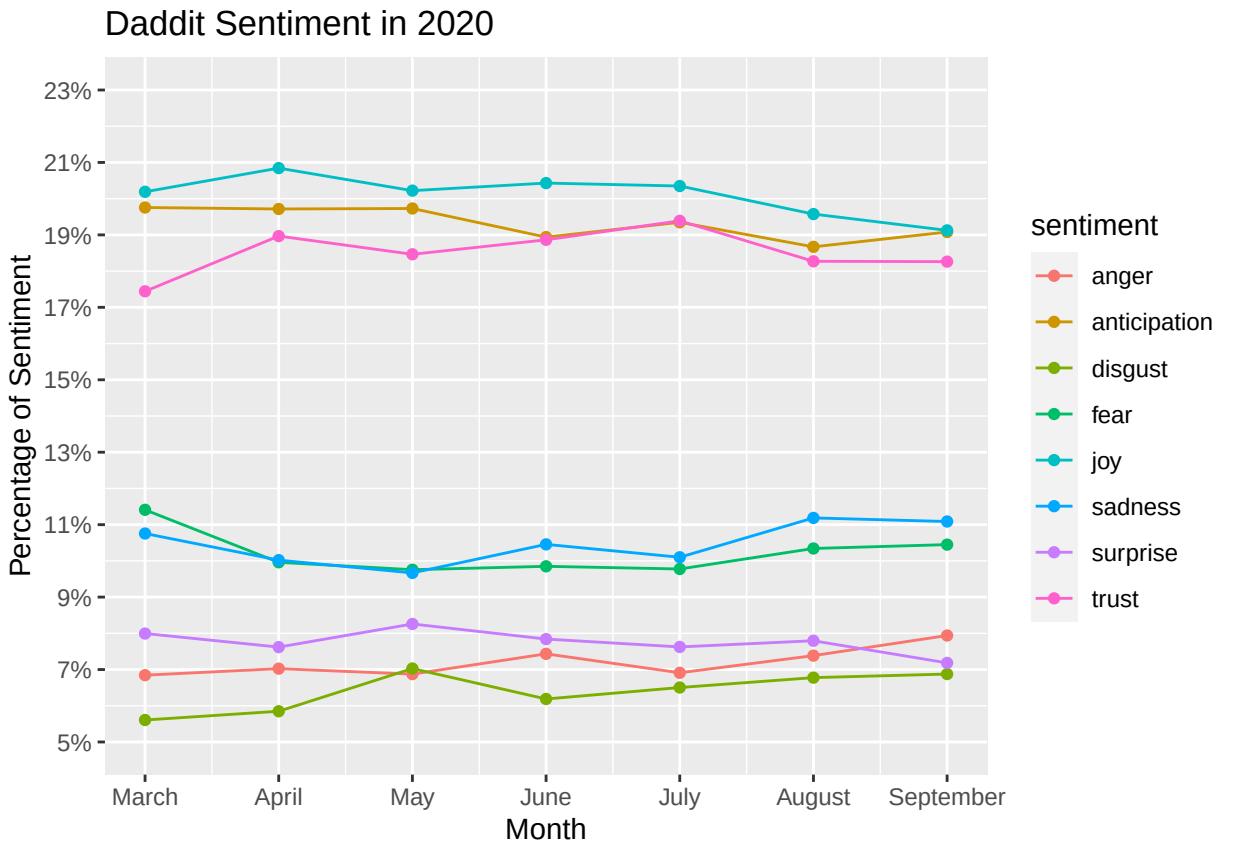
```
ggplot(dads_nrc, aes(x=sentiment, y = average, fill = year)) + geom_bar(stat="identity",position=positi
  scale_y_continuous(labels = scales::percent_format(accuracy = 1), limits = c(0.0,.25)) +
  ylab("Percent Sentiment") +
  xlab("Sentiment")+
  ggtitle("Daddit Sentiment")
```


```
ggplot(moms_nrc, aes(x=sentiment, y = average, fill = year)) + geom_bar(stat="identity",position=positi
  scale_y_continuous(labels = scales::percent_format(accuracy = 1), limits = c(0.0,.23)) +
  ylab("Percent Sentiment") +
  xlab("Sentiment")+
  ggtitle("Mommit Sentiment")
```

Emotion over time analysis

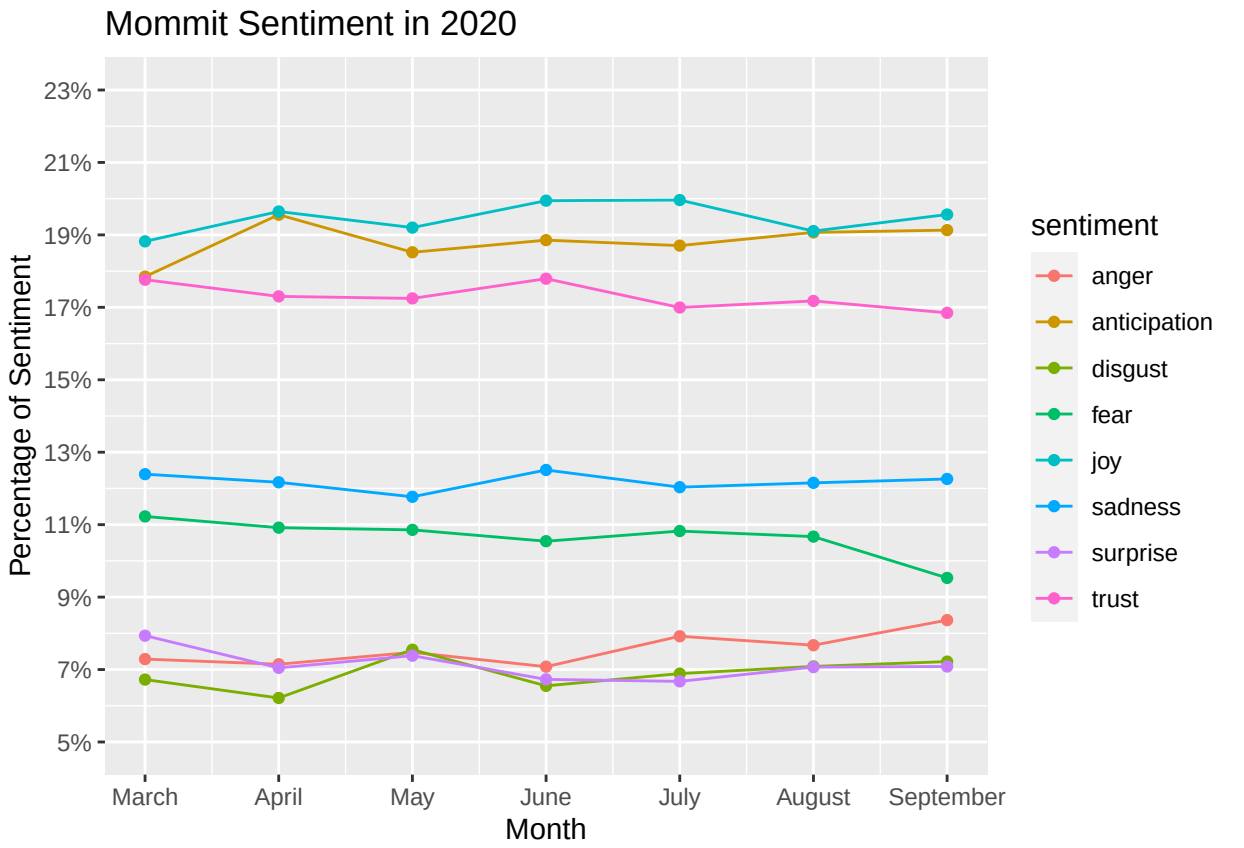
```
dads2020_d <- dads2020 %>%
  mutate(date = format(date, "%Y-%m-%d"))

date_seq <- seq(as.Date("2020-03-16"), as.Date("2020-09-15"), by="days")

dads2020_source <- VectorSource(dads2020_d[dads2020_d$date==date_seq[1],]$post)
dads2020_corpus <- VCorpus(dads2020_source)
dads2020_corpus <- clean_corpus(dads2020_corpus)

dads2020_tdm <- DocumentTermMatrix(dads2020_corpus)#control = list(tokenize = tokenizer))
dads2020_m <- as.matrix(dads2020_tdm)
dads2020_tidy <- tidy(dads2020_tdm)


dads2020_nrc_d <- dads2020_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count))%>%
```

```
  mutate(average = total_count/sum(total_count)) %>%
  mutate(date = date_seq[1]) %>%
  mutate(month = month(date))

for (d in seq_along(date_seq)) {
  if(d==1) next
  day <- date_seq[d]

  dads2020_source <- VectorSource(dads2020_d[dads2020_d$date==date_seq[d],]$post)
  dads2020_corpus <- VCorpus(dads2020_source)
  dads2020_corpus <- clean_corpus(dads2020_corpus)

  dads2020_tdm <- DocumentTermMatrix(dads2020_corpus)#control = list(tokenize = tokenizer))
  dads2020_m <- as.matrix(dads2020_tdm)
  dads2020_tidy <- tidy(dads2020_tdm)

  dads2020_nrc_dd <- dads2020_tidy %>%
    # Join to nrc lexicon by term = word
    inner_join(nrc, by = c("term" = "word")) %>%
    # Only consider Plutchik sentiments
    filter(!sentiment %in% c("positive", "negative")) %>%
    # Group by sentiment
    group_by(sentiment) %>%
    # Get total count by sentiment
    summarize(total_count = sum(count))%>%
    mutate(average = total_count/sum(total_count)) %>%
    mutate(date = date_seq[d]) %>%
    mutate(month = month(date))

  dads2020_nrc_d <- rbind(dads2020_nrc_d, dads2020_nrc_dd)

}

dads2020_nrc_m <- dads2020_nrc_d %>%
  group_by(sentiment,month) %>%
  summarise(avg = mean(average))

g4<-ggplot(dads2020_nrc_m) + geom_line(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  geom_point(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  scale_x_continuous(breaks = c(3,4,5,6,7,8,9),
                     labels=c("March", "April", "May", "June","July","August","September")) +
  ylab("Percentage of Sentiment")+
  xlab("Month")+
  ggtitle("Daddit Sentiment in 2020") +
  scale_y_continuous(breaks = c(0.05,0.07,0.09,0.11,0.13,0.15,0.17,.19,0.21,.23),
                     labels = scales::percent_format(accuracy = 1), limits = c(0.05,.23))
g4
```



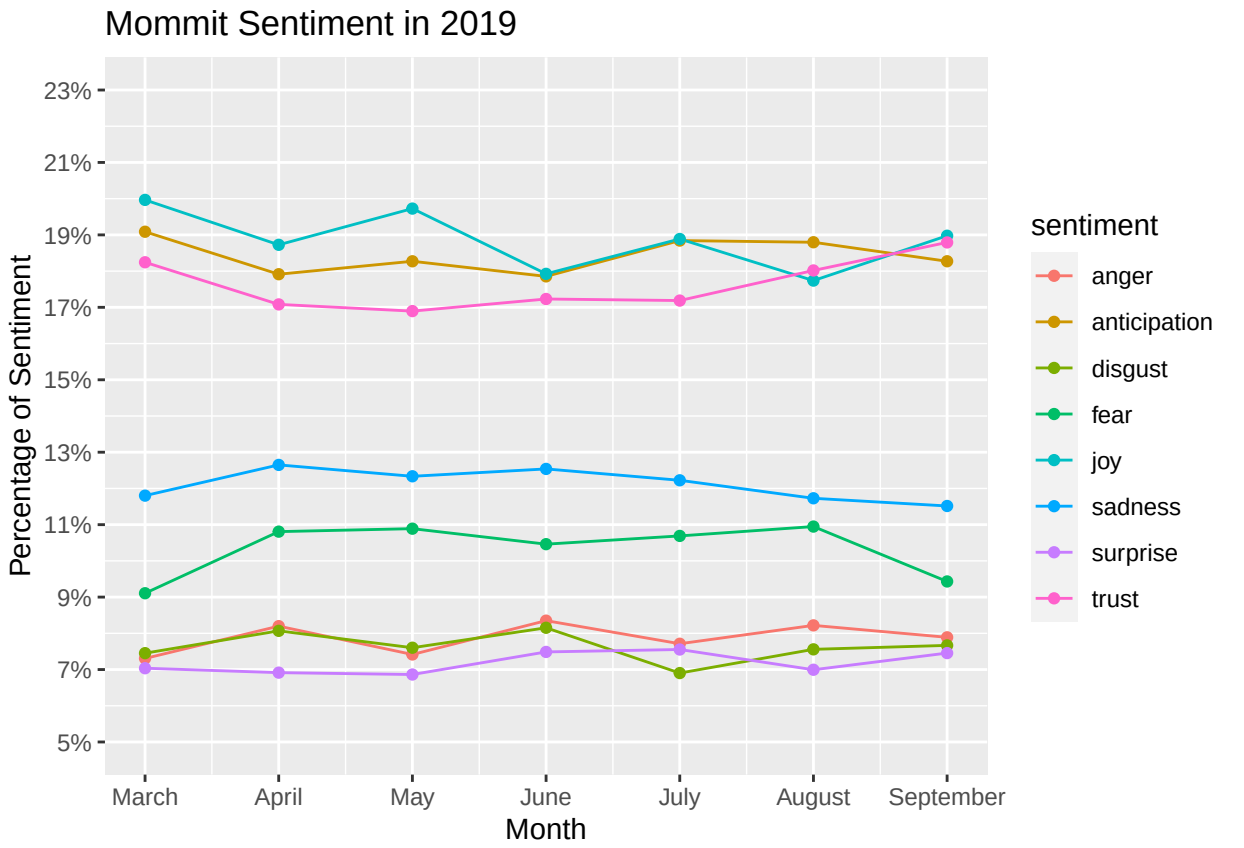

```
moms2020_d <- moms2020 %>%
  mutate(date = format(date, "%Y-%m-%d"))

date_seq <- seq(as.Date("2020-03-16"), as.Date("2020-09-15"), by="days")

moms2020_source <- VectorSource(moms2020_d[moms2020_d$date==date_seq[1],]$post)
moms2020_corpus <- VCorpus(moms2020_source)
moms2020_corpus <- clean_corpus(moms2020_corpus)

moms2020_tdm <- DocumentTermMatrix(moms2020_corpus)#control = list(tokenize = tokenizer))
moms2020_m <- as.matrix(moms2020_tdm)
moms2020_tidy <- tidy(moms2020_tdm)


moms2020_nrc_d <- moms2020_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count))%>%
  mutate(average = total_count/sum(total_count)) %>%
  mutate(date = date_seq[1]) %>%
  mutate(month = month(date))
```

```r
for (d in seq_along(date_seq)) {
  if(d==1) next

  moms2020_source <- VectorSource(moms2020_d[moms2020_d$date==date_seq[d],]$post)
  moms2020_corpus <- VCorpus(moms2020_source)
  moms2020_corpus <- clean_corpus(moms2020_corpus)

  moms2020_tdm <- DocumentTermMatrix(moms2020_corpus)#control = list(tokenize = tokenizer))
  moms2020_m <- as.matrix(moms2020_tdm)
  moms2020_tidy <- tidy(moms2020_tdm)

  moms2020_nrc_dd <- moms2020_tidy %>%
    # Join to nrc lexicon by term = word
    inner_join(nrc, by = c("term" = "word")) %>%
    # Only consider Plutchik sentiments
    filter(!sentiment %in% c("positive", "negative")) %>%
    # Group by sentiment
    group_by(sentiment) %>%
    # Get total count by sentiment
    summarize(total_count = sum(count))%>%
    mutate(average = total_count/sum(total_count)) %>%
    mutate(date = date_seq[d]) %>%
    mutate(month = month(date))
  moms2020_nrc_d <- rbind(moms2020_nrc_d, moms2020_nrc_dd)
}

moms2020_nrc_m <- moms2020_nrc_d %>%
  group_by(sentiment,month) %>%
  summarise(avg = mean(average))

g3<- ggplot(moms2020_nrc_m) + geom_line(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  geom_point(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  scale_x_continuous(breaks = c(3,4,5,6,7,8,9),
                     labels=c("March", "April", "May", "June","July","August","September")) +
  ylab("Percentage of Sentiment")+
  xlab("Month")+
  ggtitle("Mommit Sentiment in 2020") +
  scale_y_continuous(breaks = c(0.05,0.07,0.09,0.11,0.13,0.15,0.17,.19,0.21,.23),
                     labels = scales::percent_format(accuracy = 1), limits = c(0.05,.23))
g3
```



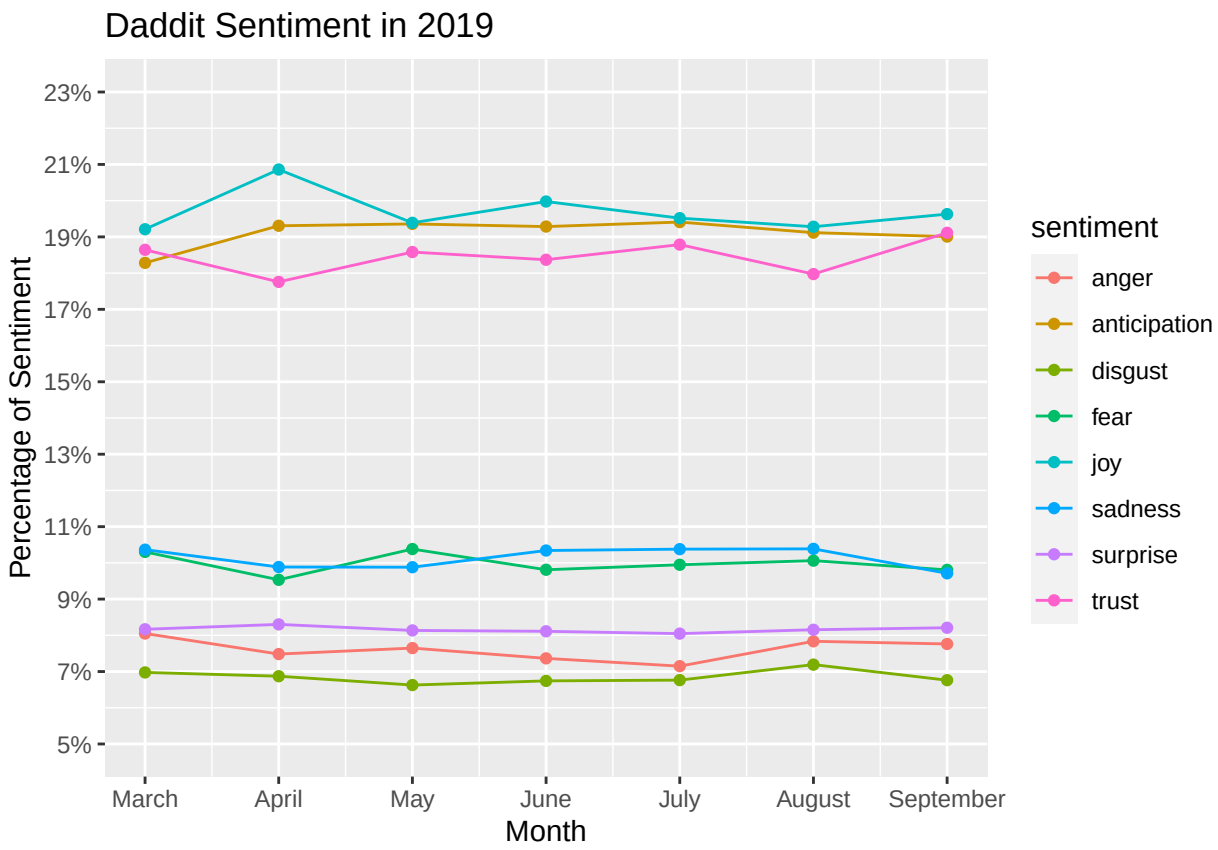

```
moms2019_d <- moms2019 %>%
  mutate(date = format(date, "%Y-%m-%d"))

date_seq <- seq(as.Date("2019-03-16"), as.Date("2019-09-15"), by="days")

moms2019_source <- VectorSource(moms2019_d[moms2019_d$date==date_seq[1],]$post)
moms2019_corpus <- VCorpus(moms2019_source)
moms2019_corpus <- clean_corpus(moms2019_corpus)

moms2019_tdm <- DocumentTermMatrix(moms2019_corpus)#control = list(tokenize = tokenizer))
moms2019_m <- as.matrix(moms2019_tdm)
moms2019_tidy <- tidy(moms2019_tdm)


moms2019_nrc_d <- moms2019_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count))%>%
  mutate(average = total_count/sum(total_count)) %>%
  mutate(date = date_seq[1]) %>%
  mutate(month = month(date))
```

```r
for (d in seq_along(date_seq)) {
  if(d==1) next
  day <- date_seq[d]

  moms2019_source <- VectorSource(moms2019_d[moms2019_d$date==date_seq[d],]$post)
  moms2019_corpus <- VCorpus(moms2019_source)
  moms2019_corpus <- clean_corpus(moms2019_corpus)

  moms2019_tdm <- DocumentTermMatrix(moms2019_corpus)#control = list(tokenize = tokenizer))
  moms2019_m <- as.matrix(moms2019_tdm)
  moms2019_tidy <- tidy(moms2019_tdm)

  moms2019_nrc_dd <- moms2019_tidy %>%
    # Join to nrc lexicon by term = word
    inner_join(nrc, by = c("term" = "word")) %>%
    # Only consider Plutchik sentiments
    filter(!sentiment %in% c("positive", "negative")) %>%
    # Group by sentiment
    group_by(sentiment) %>%
    # Get total count by sentiment
    summarize(total_count = sum(count))%>%
    mutate(average = total_count/sum(total_count)) %>%
    mutate(date = date_seq[d]) %>%
    mutate(month = month(date))

  moms2019_nrc_d <- rbind(moms2019_nrc_d, moms2019_nrc_dd)

}

moms2019_nrc_m <- moms2019_nrc_d %>%
  group_by(sentiment,month) %>%
  summarise(avg = mean(average))

g5<- ggplot(moms2019_nrc_m) + geom_line(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  geom_point(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  scale_x_continuous(breaks = c(3,4,5,6,7,8,9),
                     labels=c("March", "April", "May", "June","July","August","September")) +
  ylab("Percentage of Sentiment")+
  xlab("Month")+
  ggtitle("Mommit Sentiment in 2019") +
  scale_y_continuous(breaks = c(0.05,0.07,0.09,0.11,0.13,0.15,0.17,.19,0.21,.23),
                     labels = scales::percent_format(accuracy = 1), limits = c(0.05,.23))
g5
```




```
dads2019_d <- dads2019 %>%
  mutate(date = format(date, "%Y-%m-%d"))

date_seq <- seq(as.Date("2019-03-16"), as.Date("2019-09-15"), by="days")

dads2019_source <- VectorSource(dads2019_d[dads2019_d$date==date_seq[1],]$post)
dads2019_corpus <- VCorpus(dads2019_source)
dads2019_corpus <- clean_corpus(dads2019_corpus)

dads2019_tdm <- DocumentTermMatrix(dads2019_corpus)#control = list(tokenize = tokenizer))
dads2019_m <- as.matrix(dads2019_tdm)
dads2019_tidy <- tidy(dads2019_tdm)


dads2019_nrc_d <- dads2019_tidy %>%
  # Join to nrc lexicon by term = word
  inner_join(nrc, by = c("term" = "word")) %>%
  # Only consider Plutchik sentiments
  filter(!sentiment %in% c("positive", "negative")) %>%
  # Group by sentiment
  group_by(sentiment) %>%
  # Get total count by sentiment
  summarize(total_count = sum(count))%>%
  mutate(average = total_count/sum(total_count)) %>%
  mutate(date = date_seq[1]) %>%
  mutate(month = month(date))
```

```r
for (d in seq_along(date_seq)) {
  if(d==1) next
  day <- date_seq[d]

  dads2019_source <- VectorSource(dads2019_d[dads2019_d$date==date_seq[d],]$post)
  dads2019_corpus <- VCorpus(dads2019_source)
  dads2019_corpus <- clean_corpus(dads2019_corpus)

  dads2019_tdm <- DocumentTermMatrix(dads2019_corpus)#control = list(tokenize = tokenizer))
  dads2019_m <- as.matrix(dads2019_tdm)
  dads2019_tidy <- tidy(dads2019_tdm)

  dads2019_nrc_dd <- dads2019_tidy %>%
    # Join to nrc lexicon by term = word
    inner_join(nrc, by = c("term" = "word")) %>%
    # Only consider Plutchik sentiments
    filter(!sentiment %in% c("positive", "negative")) %>%
    # Group by sentiment
    group_by(sentiment) %>%
    # Get total count by sentiment
    summarize(total_count = sum(count))%>%
    mutate(average = total_count/sum(total_count)) %>%
    mutate(date = date_seq[d]) %>%
    mutate(month = month(date))

  dads2019_nrc_d <- rbind(dads2019_nrc_d, dads2019_nrc_dd)

}

dads2019_nrc_m <- dads2019_nrc_d %>%
  group_by(sentiment,month) %>%
  summarise(avg = mean(average))

g6<- ggplot(dads2019_nrc_m) + geom_line(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  geom_point(aes(y = avg,x=month,group = sentiment, color = sentiment)) +
  scale_x_continuous(breaks = c(3,4,5,6,7,8,9),
                     labels=c("March", "April", "May", "June","July","August","September")) +
  ylab("Percentage of Sentiment")+
  xlab("Month")+
  ggtitle("Daddit Sentiment in 2019") +
  scale_y_continuous(breaks = c(0.05,0.07,0.09,0.11,0.13,0.15,0.17,.19,0.21,.23),
                     labels = scales::percent_format(accuracy = 1), limits = c(0.05,.23))
g6
```


Daddit Sentiment in 2019
23%
21%
19%
17%
15%
13%
11%
9%
7%
5%
Percentage of Sentiment
March
April
May
June
July
August
September
Month
sentiment
anger
anticipation
disgust
fear
joy
sadness
surprise
trust