hw6_Sayan_Biswas

Sayan Biswas

4 April 2019

Part A

Problem 1

```
cross_validation <- function(formula,data,n){
  data_cv <- crossv_kfold(data,n)
  data_cv <- data_cv %>%
    mutate(fit=map(train,~lm(formula,data = .)),
           prediction_test = map2(test,fit, ~ add_predictions(as_tibble(.x), .y)),
           rmse_test = map2_dbl(fit,test, ~ rmse(.x,.y)))
  return(mean(data_cv$rmse_test))}
```

Problem 2

```
data("BostonHousing")
```

The below model was used by me to predict crime rate in the previous assignment and the RMSE for the same is shown below:

```
formula <- as.formula(log2(crim) ~ log2(dis)+rad+log2(nox))
set.seed(1)
cross_validation(formula,BostonHousing,5)
```

```
## [1] 1.187778
```

The models which I tried to get a lower RMSE are as follows:

1. Adding "lstat" as my predictor variable.

```
formula <- as.formula(log2(crim) ~ log2(dis)+rad+log2(nox)+log2(lstat))
set.seed(1)
cross_validation(formula,BostonHousing,5)
```

```
## [1] 1.169106
```

2. Another model by adding additional predictor variable "zn":

```
formula <- as.formula(log2(crim) ~ log2(dis)+rad+log2(nox)+log2(lstat)+zn)
set.seed(1)
cross_validation(formula,BostonHousing,5)
```

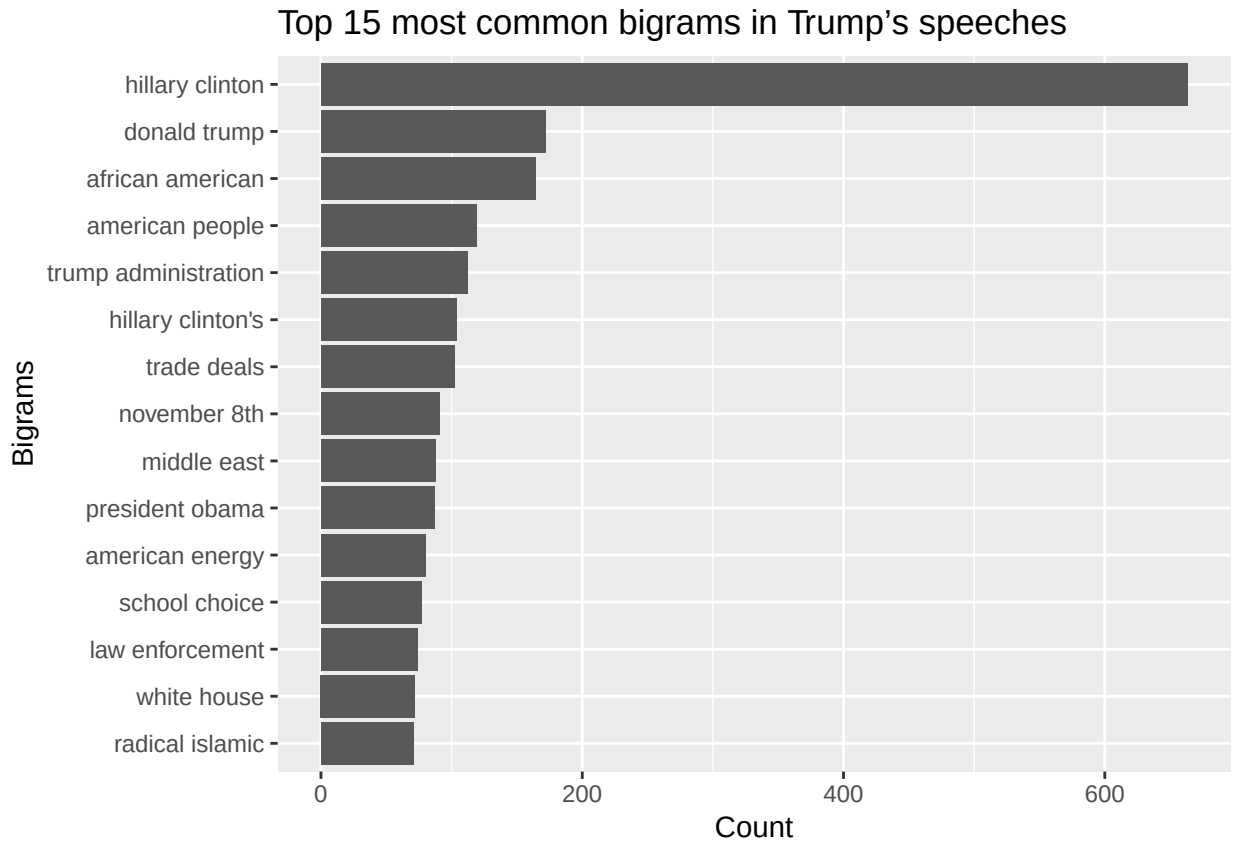
```
## [1] 1.149327
```

Part B

Problem 3

```
text <- read_lines("full_speech.txt")
trump_speech <- tibble(line=1:length(text),text=text)
```

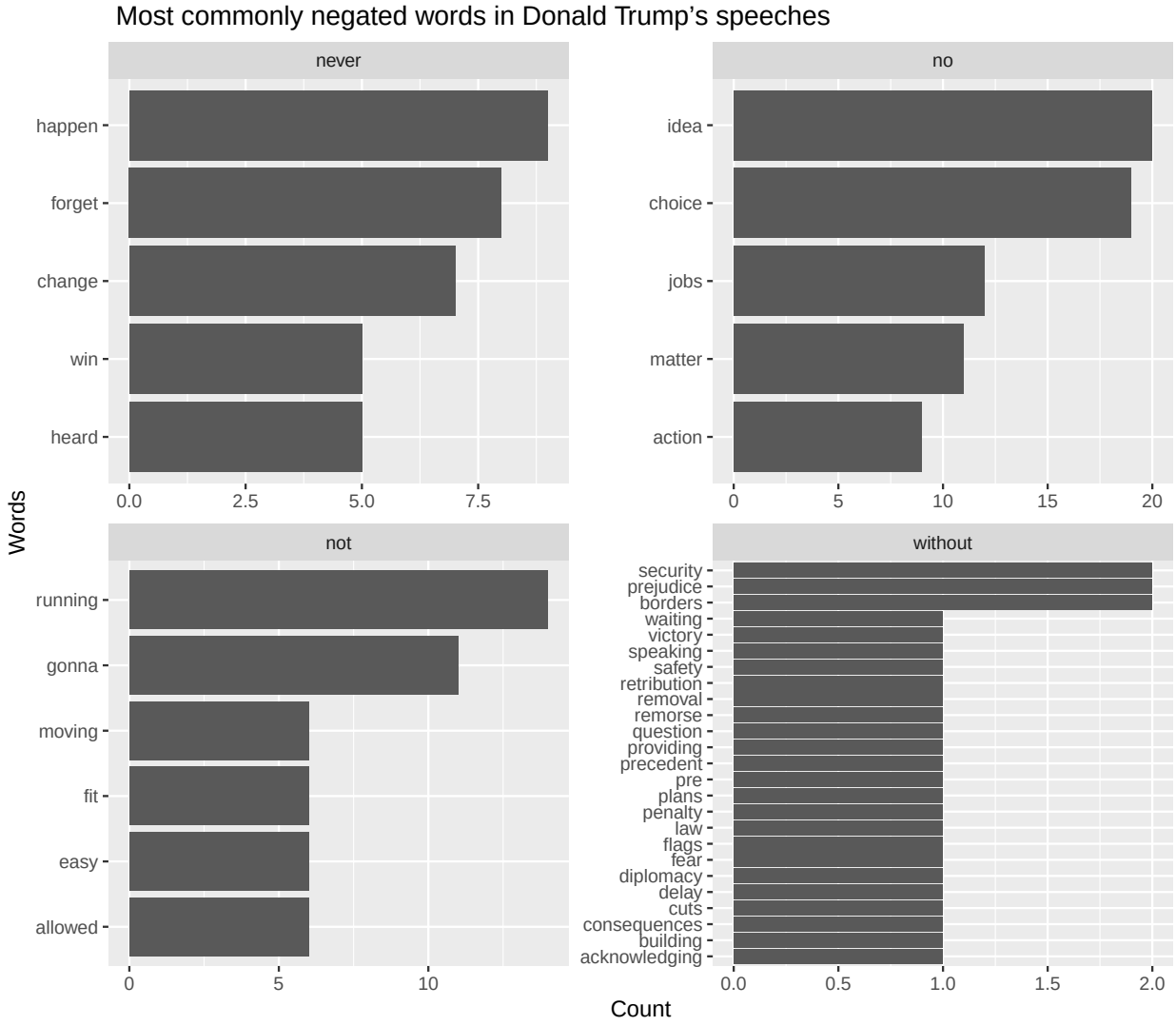
```
trump_speech %>%
  unnest_tokens(word,text,token="ngrams",n=2)%>%
  separate(word, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% stop_words$word & !word1=="applause") %>%
  filter(!word2 %in% stop_words$word & !word2=="applause") %>%
  unite(bigram, word1, word2, sep = " ")%>%
  count(bigram,sort=T)%>%
  top_n(15)%>%
  ggplot(aes(x=reorder(bigram,n),y=n))+
  geom_col()+
  coord_flip()+
  labs(title = "Top 15 most common bigrams in Trump's speeches",
       x="Bigrams",
       y="Count")
```



Problem 4

```
negation_words <- c("not", "no", "never", "without")

trump_speech %>%
  unnest_tokens(word,text,token="ngrams",n=2)%>%
  separate(word, c("word1", "word2"), sep = " ") %>%
  filter(word1 %in% negation_words) %>%
  filter(!word2 %in% stop_words$word & !word2=="applause") %>%
  group_by(word1)%>%
  count(word2,sort=T)%>%
  top_n(5)%>%
  ggplot(aes(x=reorder(word2,n),y=n))+
  geom_col()+
  coord_flip()+
  facet_wrap(~word1,scales = "free")+
  labs(title = " Most commonly negated words in Donald Trump's speeches",
       x="Words",
       y="Count")
```


Most commonly negated words in Donald Trump's speeches
never
happen
forget
change
win
heard
0.0
2.5
5.0
7.5
no
idea
choice
jobs
matter
action
0
5
10
15
20
not
running
gonna
moving
fit
easy
allowed
0
5
10
without
security
prejudice
borders
waiting
victory
speaking
safety
retribution
removal
remorse
question
providing
precedent
pre
plans
penalty
law
flags
fear
diplomacy
delay
cuts
consequences
building
acknowledging
0.0
0.5
1.0
1.5
2.0
Words
Count

Problem 5

```
trump_speech %>%
  unnest_tokens(word,text,token="ngrams",n=2)%>%
  separate(word, c("word1", "word2"), sep = " ") %>%
  filter(!word1 %in% negation_words) %>%
  filter(!word2 %in% stop_words$word & !word2=="applause") %>%
  inner_join(get_sentiments("loughran"),by=c("word2"="word"))%>%
  group_by(sentiment)%>%
  count(word2,sort=T)%>%
  top_n(5)%>%
  ggplot(aes(x=reorder(word2,n),y=n))+
  geom_col()+
  coord_flip()+
  facet_wrap(~sentiment,scales = "free")+
  labs(title =
      "Sentiment analysis of Donald Trump's speeches",
       x="Words",
       y="Count")
```