WSP R Open Question Analysis

Lizzie Jones[1]

02/05/2021

Text analysis and visualisations for open questions

This rMarkdown explores and analyses the open questions using text, work frequency and sentiment analysis techniques which are beyond the scope of NVivo (or where R is more effective).

The key open-ended questions include:

- Q8. How did you feel when you saw WS in the wild?
- Q9a. [If respondent has heard of the white stork project and its efforts to reintroduce white storks to southern England], please briefly summarise what you have heard.
- Q11a-c. Three words used to describe white storks
- Q15. Do you support the White Stork Project?
- Q16. What are yours views on the management of White Storks?
- Q17.1-17.13. Which (if any) methods of white stork project management would you support?

In this rMarkdown script I clean and explore each question in turn in the sections below, visualise the data and comment on any interesting findings.

Q8. How did you feel when you saw white stork in the wild?

Respondents were first asked if they had every seen a White Stork in the wild (Responses = Yes/No/) Investigating the question "How did you feel when you saw a white stork in the wild?" (N = 1123) Notes from codebook - No need for NVivo thematic analysis? Run some form of sentiment analysis

Question options - "Seen in the UK", "Seen outside the UK". Respondents were then asked if they wanted to explain/elaborate on their answer using an open-ended question.

```
#Creating a non-local/local factor column for proximity to any WS release site
final_data <- mutate(final_data, SiteProximity =
                        ifelse(ReleaseSite == "No", "Not local", "Local"))
### Q8. How did you feel when you saw WS in the wild?
# Create words df to seperately clean, capitalise first letter etc
feel_df <- final_data %>%
  dplyr::select(UniqueID_all, SurveyType, Q8.WhereSeen, Q8.2_feelings)

# feel_syns <- qdap::synonyms(feel_df$Q8.2_feelings)

### Create two dataframes: one for each data collection for easier comparison
feelings_UK <- feel_df[which(feel_df$Q8.WhereSeen == "UK"),]
feelings_nonUK <- feel_df[which(feel_df$Q8.WhereSeen == "OutsideUK"),]
feelings_both <- feel_df[which(feel_df$Q8.WhereSeen == "Both"),]

# Create the custom function that will be used to clean the corpus: clean_coupus
clean_corpus_feel <- function(corpus){
  corpus <- tm_map(corpus, stripWhitespace)
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, content_transformer(tolower))
  corpus <- tm_map(corpus, removeWords, stopwords("en"))
  corpus <- tm_map(corpus, lemmatize_words)
    return(corpus)
}

# Seen in the wild within the UK (n=472)
corpus_seenUK <- Corpus(VectorSource(feelings_UK$Q8.2_feelings))
corpus_seenUK_clean <- clean_corpus_feel(corpus_seenUK)
corpus_seenUK_clean <- tm_map(corpus_seenUK_clean, removeWords,
                c("they", "the", "the ", "its", "also", "I'm", "don't"," see", "can","see ","white",
                    "think", "one", "really", "bird", "see", "stork", "feel", "was"))
# Seen in the wild outside the UK (n=671)
corpus_seen_nonUK <- Corpus(VectorSource(feelings_nonUK$Q8.2_feelings))
corpus_seen_nonUK_clean <- clean_corpus_feel(corpus_seen_nonUK)
corpus_seen_nonUK_clean <- tm_map(corpus_seen_nonUK_clean, removeWords,
                c("they", "the", "the ", "its", "also", "I'm", "don't"," see", "can","see ","white",
                    "think", "one", "really", "bird", "see", "stork", "feel", "was"))
# Seen in the wild in the UK and outside the UK (n=299)
corpus_seen_both <- Corpus(VectorSource(feelings_both$Q8.2_feelings))
corpus_seen_both_clean <- clean_corpus_feel(corpus_seen_both)
corpus_seen_both_clean <- tm_map(corpus_seen_both_clean, removeWords,
                c("they", "the", "the ", "its", "also", "I'm", "don't"," see", "can","see ","white",
                    "think", "one", "really", "bird", "see", "stork", "feel", "was"))

## Plot image and save as a PDF using the file viewer
par(mfrow=c(1,3))
wordcloud(corpus_seenUK_clean, max.words=20, random.order=FALSE, rot.per=0,
          use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
title(main="A.",col="black",font=2,line=-4)

wordcloud(corpus_seen_nonUK_clean, max.words=20, random.order=FALSE, rot.per=0,
          use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
title(main="B.",col="black",font=2,line=-4)
wordcloud(corpus_seen_both_clean, max.words=20, random.order=FALSE, rot.per=0,
          use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
title(main="C.",col="black",font=2,line=-4)
```

A.

great privileged
birds amazing
awe amazed lucky
happy saw wild estate
excited
knepp storks flying
seeing seen excite
pleased amaze

B.

fantastic
nests beautiful
saw seen
ge happy
sed nest
excited
ing birds great
e excite storks
amazed amazing
always lovely

C.

impressive
pleased
py seen impress
be excited
st birds excit
g great
s saw seeing
knepp wild time
beautiful
amazing

Q11. What descriptive words do you associate with white storks?

For this question I have taken the 3 'words to describe' columns and combined the words to create a new 'long-format' dataset to conduct preliminary word frequency analysis, word clouds etc. and also added SurveyType and ReleaseSite as possible grouping vairables. I then clean this dataframe ('words_df') by removing common English stop words, punctuation, blank spaces and convert all text to lower case. The word cloud is created using the 'wordcloud' package.

```
# Words used to describe WS - will probably combine into one column, labelled by Respondent ID and SurveyType
summary(final_data$Q11_word1)
```

```
##              Large      Rare    Elegant    Beautiful        Big
##    670        296        208       152          151          145
##   White    Majestic       Bird       Tall    Beautiful    Graceful
##    121        116         95         80          62          62
##   large    Impressive   Majestic     Babies     Elegant       Wild
##     50         46         38         26          26          26
##   Large   Magnificent    Huge   Interesting  Magnificent   beautiful
##     24         24         23         21          21          20
##  Iconic     elegant    Stunning      big      Endangered   Impressive
##     18         17         17         16          14          14
## majestic    Unusual      rare      Regal      Striking    long legs
##     13         13         12         11          11          10
##  Amazing     Nature      Nice      Pretty    Spectacular     white
##      9          9          9          9           9           9
##   White      Beauty    Exciting   Graceful        Long     impressive
##      9          8          8          8           8           7
## Large bird   Magical   Attractive  Chimneys     Exotic    Fascinating
##      7          7          6          6           6           6
##   Grace     graceful   Historic      huge       Lovely       Lucky
##      6          6          6          6           6           6
## Magestic    Migratory  Rewilding     tall        Beak         Big
##      6          6          6          6           5           5
##   Birth    Charismatic    Cool       Good        Leggy      Native
##      5          5          5          5           5           5
## Peaceful    Stunning     Unique     Awesome       Hope       iconic
##      5          5          5          4           4           4
## Interesting Long legs   Margarine   Natural    Symbolic     Ancient
##      4          4          4          4           4           3
## Attractive   Awesome     babies     Babies!      Birds       Black
##      3          3          3          3           3           3
## Breathtaking  Cute     Elegance   Endangered  Fascinating    Grand
##      3          3          3          3           3           3
##   Lanky    Long beak  Long legged   (Other)
##      3          3          3          527
```

```
summary(final_data$Q11_word2)
```

```
##           Beautiful     Rare      Large      White     Elegant
##    932       147        135        124        115         95
##  Graceful      Big    Majestic  Impressive  Beautiful  Interesting
##     63         62         61         50          44         40
##    Tall       Bird    Endangered    Wild      Babies     Elegant
##     37         33         26         22          21         20
##  Iconic    Impressive   Majestic    Unusual     large    Long legs
##     18         18         18         18          17         17
##   Huge       White      Exotic    majestic    elegant    Graceful
##     15         15         14         14          13         12
## Migratory    Native    Striking   beautiful   Exciting    Noisy
##     12         12         12         11          11         11
##   white       big    Charismatic Interesting Long beak  Magnificent
##     11         10         10         10          10         10
## Magnificent   Rare     Stunning     Beak      graceful   impressive
##     10         10         10          9           9          9
##   Leggy     Special      Large       rare       Regal       tall
##      9          9          8          8           8          8
##    wild     Magical     Nature   Prehistoric   Strong     Unusual
##      8          7          7          7           7          7
##  Amazing    Exciting   Fascinating Important     Stork      Unique
##      6          6          6          6           6          6
##  Wetland     Amazing     Beauty      Birth     exciting     Fish
##      6          5          5          5           5          5
##    Long   Long legged   Pretty      Shy       Stately     Animal
##      5          5          5          5           5          4
##  At risk    Awesome    beautiful  Big beak  Charismatic     Cute
##      4          4          4          4           4          4
## Distinctive  European    Faithful     Free      Handsome     Hope
##      4          4          4          4           4          4
##  Imposing    Magestic   Migration   Mythical    Natural     Natural
##      4          4          4          4           4          4
##   Nests        Nice   Reintroduced  (Other)
##      4          4          4          855
```

```
summary(final_data$Q11_word3)
```

```
##                      Rare   Beautiful   Beautiful      White      Large
##       1374            115          76          53         51         50
##        Bird      Majestic         Big     Elegant   Graceful Endangered
##         43            34          32          32         30         27
##  Impressive   Interesting        Wild   beautiful   Exciting Fascinating
##         26            25          23          18         18         17
##       Noisy        Babies      Iconic Interesting   Majestic  Migratory
##         17            16          16          16         13         12
##        rare     Important    Graceful        Legs    Magical    Special
##         12            11          10          10         10         10
##        Tall    impressive      Native        Beak       Hope  Impressive
##         10             9           9           8          8          8
## Magnificent    Stunning      Unique     Unusual    Awesome      Birds
##          8             8           8           8          7          7
##        Calm      exciting interesting   Long legs    Natural      Regal
##          7             7           7           7          7          7
##      Soaring       Amazing  Attractive     Awesome     Beauty    Elegant
##          7             6           6           6          6          6
##       Exotic Fascinating   Inspiring       Lucky  Rewilding    Ancient
##          6             6           6           6          6          5
##       Exotic           Fun        Huge       large      Large   Mythical
##          5             5           5           5          5          5
##        Nest         Nests      Pretty      Scarce     Serene    Special
##          5             5           5           5          5          5
## Spectacular         Stork    Striking      Unique       wild    Babies!
##          5             5           5           5          5          4
##         big          Bird       Birth  Captivating Charismatic   elegant
##          4             4           4           4          4          4
##    European          Free   Important Intelligent      Leggy  long beak
##          4             4           4           4          4          4
##   Long beak        Lovely       Loyal   Mysterious     Nature    Peaceful
##          4             4           4           4          4          4
## Prehistoric         Proud     special      (Other)
##          4             4           4         949
```

```
### Investigating relationships between 2 words
# https://uc-r.github.io/word_relationships

#### Investigating individual-word frequency
# Create a long-format words df (max 3 rows per responseID) to seperately clean, capitalise first letter etc
words_df <- final_data %>%
  dplyr::select(UniqueID_short, SurveyType, Q11_word1, Q11_word2, Q11_word3) %>%
  pivot_longer(
   cols = starts_with("Q11_"),
   names_to = "Word_num",
   values_to = "Words",
   values_drop_na = TRUE)

# Clean text
words_df$Words <- tolower(words_df$Words)#convert all text to lower case
words_df$Words <- gsub("[^[:graph:]]", " ", words_df$Words) #get rid of non graphical characters
words_df$Words <- gsub("a ", "", words_df$Words)# Remove single 'a' words
words_df$Words <- gsub("one", "", words_df$Words)# Remove one words
words_df$Words <- gsub("[[:punct:]]", "", words_df$Words)# Remove punctuation
words_df$Words <- gsub("^ ", "", words_df$Words)# Remove blank spaces at the beginning
words_df$Words <- gsub(" $", "", words_df$Words)# Remove blank spaces at the end
words_df$Words <- gsub("[ |\t]{2,}", "", words_df$Words)# Remove tabs
# words_df$Words <- gsub(" ", "", words_df$Words)# Replace blank spaces with _
head(words_df, 50)
```

```
## # A tibble: 50 x 4
##    UniqueID_short SurveyType Word_num  Words
##             <int> <fct>      <chr>     <chr>
## 1               1 Proactive  Q11_word1 "leg"
## 2               1 Proactive  Q11_word2 "babies"
## 3               1 Proactive  Q11_word3 ""
## 4               2 Proactive  Q11_word1 "white"
## 5               2 Proactive  Q11_word2 "long legs"
## 6               2 Proactive  Q11_word3 "bird"
## 7               3 Proactive  Q11_word1 ""
## 8               3 Proactive  Q11_word2 ""
## 9               3 Proactive  Q11_word3 ""
## 10              4 Proactive  Q11_word1 "hope"
## # … with 40 more rows
```

```
# Create corpus, lemmatise and remove english stopwords
corpus_words <- Corpus(VectorSource(words_df$Words))
corpus_words <- tm_map(corpus_words, lemmatize_words)
corpus_words <- tm_map(corpus_words, removeWords, stopwords("english")) #removes common english stopwords

# Create a wordcloud
wordcloud(corpus_words, scale=c(5,0.5), max.words=50, random.order=FALSE, rot.per=0.25,
          use.r.layout=FALSE, colors=brewer.pal(10,"Dark2"))
```

```r
# # Code to automatically detect, and combine synonyms (e.g., big and large) for clearer word clouds and frequency analysis
# library(dplyr)
# words_df %>%
#   group_by(syn_group) %>%
#   mutate(sum_word_count = sum(word_count)) %>%
#   filter(rank == 1)
#
# d <- data.frame(Term = row.names(m), word_count = m[,1])
# all_pos <- c("ADJECTIVE", "ADVERB", "NOUN","VERB")
# syns <- vector("list", length(all_pos))
# for(w in seq(nrow(d))){
#   # if sysns of (d$Term[w]) has been calculated skip over current w
#   emf <- getTermFilter("ExactMatchFilter", d$Term[w], TRUE)
#   for(i in seq_along(syns)){
#     terms <- getIndexTerms(all_pos[i], 1, emf)
#     if(is.null(terms)){
#       syns[i] <- NA
#     } else{
#       syns[[i]] <-  getSynonyms(terms[[1]])
#     }
#   }
#   # store the results of syns for current w
# }
```

Word frequency analysis (Words used to describe White Storks)

```r
# Frequent word analysis
#build a term-document matrix
library("tm")
TDM_words = tm::TermDocumentMatrix(corpus_words, control = list(minWordLength = 1))
m = as.matrix(TDM_words)
v = sort(rowSums(m), decreasing = TRUE)
d = data.frame(word = names(v),freq=v)

# We can find the words that appear at least 100 times by calling the findFreqTerms() function on the term.doc.matrix
HiFreq_words <- findFreqTerms(TDM_words, 100)
HiFreq_words
```

```
##  [1] "baby"       "white"      "long"       "bird"       "rare"
##  [6] "large"      "big"        "majestic"   "graceful"   "beautiful"
## [11] "elegant"    "tall"       "impressive" "interest"
```

```r
# Now you also see how associated a word is to another word or a list of words.
findAssocs(TDM_words, HiFreq_words, 0.4)
```

```
## $baby
## numeric(0)
##
## $white
## numeric(0)
##
## $long
## legs
## 0.59
##
## $bird
## numeric(0)
##
## $rare
## numeric(0)
##
## $large
## numeric(0)
##
## $big
## numeric(0)
##
## $majestic
## numeric(0)
##
## $graceful
## numeric(0)
##
## $beautiful
## numeric(0)
##
## $elegant
## numeric(0)
##
## $tall
## numeric(0)
##
## $impressive
## numeric(0)
##
## $interest
## numeric(0)
```

```
# or, just compute word strength associations
findAssocs(TDM_words, "long", 0.5) # Looks like the word "long" and "legs" are very frequently associated (51% of
the time)
```

```
## $long
## legs
## 0.59
```

```
barplot(d[1:30,]$freq, las = 2, names.arg = d[1:30,]$word,
        col ="black", main ="Most frequent words used to describe White Storks",
        ylab = "Word frequencies")
```

Most frequent words used to describe White Storks
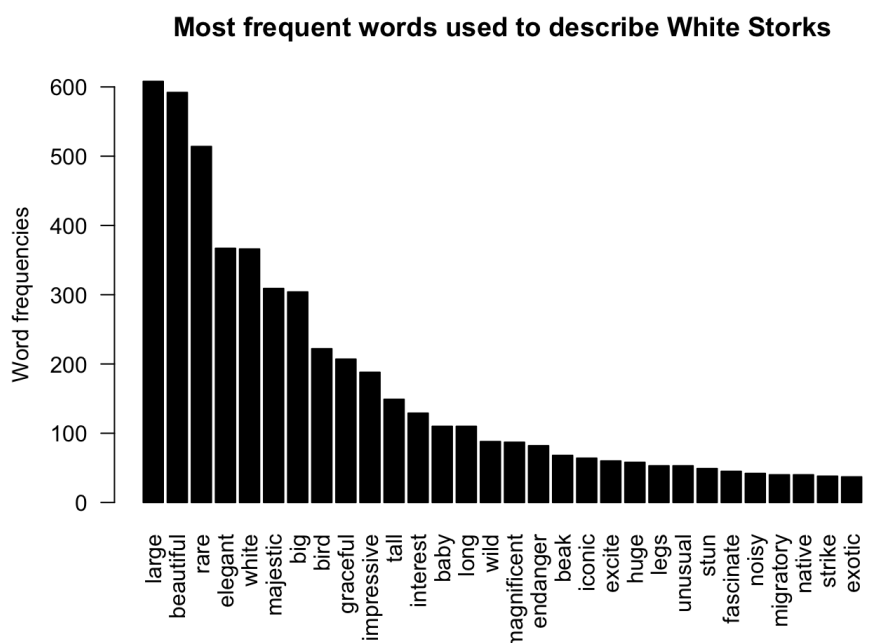
```
### Sentiment: Words to describe WS
# regular sentiment score using get_sentiment() function and method of your choice
# please note that different methods may have different scales
words_sentiment <- syuzhet::get_sentiment(corpus_words, method="syuzhet")
# see the first row of the vector
head(words_sentiment)
```

```
## [1] 119.7    0.0    0.0
```

```
# see summary statistics of the vector
summary(words_sentiment)
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.00    0.00    0.00   39.90   59.85  119.70
```

Q15a. Support for white stork reintroduction

Question = Do you support the reintroduction of white Storks to southern England?

```
### Q15. Do you support the WSP?

# Create words df to seperately clean, capitalise first letter etc
support_df <- final_data %>%
  dplyr::select(UniqueID_all, SurveyType, SiteProximity, Q15_WSP_support_open)

# Clean the data
support_df$Q15_WSP_support_text <- gsub("[^[:graph:]]", " ", support_df$Q15_WSP_support_open)
support_df$Q15_WSP_support_open <- gsub("[[:punct:]]", "", support_df$Q15_WSP_support_open)# Remove punctuation
support_df$Q15_WSP_support_text <- gsub(",", " ", support_df$Q15_WSP_support_open) # Remove commas after words
support_df$Q15_WSP_support_text <- gsub("'", "", support_df$Q15_WSP_support_open) # Remove apostrophes
support_df$Q15_WSP_support_text <- gsub("^ ", "", support_df$Q15_WSP_support_text) # Remove blank spaces at the b
eginning
support_df$Q15_WSP_support_text <- gsub(" $", "", support_df$Q15_WSP_support_text) # Remove blank spaces at the e
nd

# # Create corpus, lemmatise and remove english stopwords
# corpus_support <- Corpus(VectorSource(support_df$Q15_WSP_support_text))
# corpus_support <- tm_map(corpus_support, lemmatize_words)
# corpus_support <- tm_map(corpus_support, removeWords,stopwords("english")) #removes common english stopwords
#
# #build a term-document matrix
# TDM_support = tm::TermDocumentMatrix(corpus_support, control = list(minWordLength = 1))
# m_support = as.matrix(TDM_support)
# v_support = sort(rowSums(m_support), decreasing = TRUE)
# d_support = data.frame(word = names(v_support),freq=v_support)
#
# # Reasons for support/not support WSP
# class(support_df$Q15_WSP_support_text)
# sentiment(get_sentences(support_df$Q15_WSP_support_text))

### Create two dataframes: one for each data collection for easier comparison
proact_support <- support_df[which(support_df$SurveyType == "Proactive"),]
natrep_support <- support_df[which(support_df$SurveyType == "NatRep"),]

# Proactive survey
Corpus_pro_support <- Corpus(VectorSource(proact_support$Q15_WSP_support_text))
Corpus_pro_support <- tm_map(Corpus_pro_support, removeNumbers)
Corpus_pro_support <- tm_map(Corpus_pro_support, removeWords, stopwords("english")) #removes common english stopw
ords
Corpus_pro_support <- tm_map(Corpus_pro_support, removeWords, c("they", "the", "also"))
Corpus_pro_support <- tm_map(Corpus_pro_support, lemmatize_words)
wordcloud_pro_support <- wordcloud(Corpus_pro_support,  max.words=50, random.order=FALSE, rot.per=0,
        use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
```



```
# Nat Rep survey
Corpus_nat_support <- Corpus(VectorSource(natrep_support$Q15_WSP_support_text))
Corpus_nat_support <- tm_map(Corpus_nat_support, removeNumbers)
Corpus_nat_support <- tm_map(Corpus_nat_support, removeWords, stopwords("english")) #removes common english stopw
ords
Corpus_nat_support <- tm_map(Corpus_nat_support, removeWords, c("they", "the", "also"))
Corpus_nat_support <- tm_map(Corpus_nat_support, lemmatize_words)
wordcloud_nat_support <- wordcloud(Corpus_nat_support,  max.words=50, random.order=FALSE, rot.per=0,
        use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
```

Q16a. Expressing views on WS management

Question = Do you feel that you can express your views on the ongoing white stork reintroduction in a way that will influence management decisions?

```
# Polarity / Sentiment Analysis

### Q16. What are yours views on the management of White Storks?
head(final_data$Q16_views_management_open)
```

```
## [1]
## [2]
## [3] question seems unclear
## [4] I work in conservation and rewilding so am knowledgable and trained on the subject.
## [5] I've not looked into the project as much as I should, which is dreadful of me. To be honest I think that's
because I live in the north, and we forever seem to be far behind in the progress of bold reintroductions - even
free-living beavers are barely discussed here, so I suppose it seems a long way off to have White Storks (though
I hope not).
## [6]
## 1284 Levels:  ...
```

```
# Clean the data
final_data$Q16_views_management_text <- gsub("[^[:graph:]]", " ", final_data$Q16_views_management_open)
final_data$Q16_views_management_text <- gsub("[[:punct:]]", "", final_data$Q16_views_management_text)# Remove pun
ctuation
final_data$Q16_views_management_text <- gsub("^ ", "", final_data$Q16_views_management_text)
final_data$Q16_views_management_text <- gsub(" $", "", final_data$Q16_views_management_text)

# Reasons for support/not support WSP
class(final_data$Q16_views_management_text)
```

```
## [1] "character"
```

```
sentiment(get_sentences(final_data$Q16_views_management_text))
```

```
##       element_id sentence_id word_count  sentiment
##    1:          1           1         NA  0.0000000
##    2:          2           1         NA  0.0000000
##    3:          3           1          3 -0.2886751
##    4:          4           1         14  0.2806243
##    5:          5           1         66  0.1107823
##    ---
## 3527:       3527           1         NA  0.0000000
## 3528:       3528           1         13  0.0000000
## 3529:       3529           1         NA  0.0000000
## 3530:       3530           1          3  0.0000000
## 3531:       3531           1          4  0.0000000
```

```
# Wrd frequencies
Corpus_management <- Corpus(VectorSource(final_data$Q16_views_management_text))
Corpus_management <- tm_map(Corpus_management, removeNumbers)
Corpus_management <- tm_map(Corpus_management, removeWords, stopwords("english")) #removes common english stopwor
ds
Corpus_management <- tm_map(Corpus_management, removeWords, c("they", "the", "also"))  #You can specify words to
 remove

#build a term-document matrix
TDM_management = tm::TermDocumentMatrix(Corpus_management, control = list(minWordLength = 1))
m_management = as.matrix(TDM_management)
v_management = sort(rowSums(m_management), decreasing = TRUE)
d_management = data.frame(word = names(v_management),freq=v_management)

# Create a wordcloud
wordcloud(Corpus_management, scale=c(5,0.5), max.words=80, random.order=FALSE, rot.per=0,
          use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
```

Q17. Support for methods of WSP management?

Question = Q17.1-17.13. Which (if any) methods of white stork project management would you support?

```
# Polarity / Sentiment Analysis

### Q17. Support for methods of WSP management?
summary(final_data$Q17.13a_other_open)
```

##
##
3489
1.Could there be a reward system where people get money if they have a stork nest on their property? I feel the would work better then "compensation". I read an article (can't remember where) that said when arctic reindeer herders were paid if there were wolverine in their territory, that helped the wolverine.   2. I do not feel qualified to say if "discouraging nest building" is fair or not. I would want to know what the system is in any country where people accept storks - are storks discouraged from crucial civic buildings? Are they admired or disliked? I suggest trying to follow whichever strategy seems to work in another country, to start with.
##
1
##
Although it is often best to let nature take its course, sometimes, because the UK is a highly modified environment, management is necessary. Therefore habitat management may be required in some locations, at least in the short and medium term to provide suitable habitat for target species.
##
1
##
Any management methods involving a reintroduction would need looking at in detail and in a case by case basis ,not with generalisation. What maybe right in one area with one landowner and issue will not necessarily translate to another.
##
1
##
awareness
##
1
##
Better nature reserves. Strong legal protection for rare and common native species. Proper ban on hunting with dogs and grouse shooting. Grouse moors change of use to nature hubs, tourist sites. Wildlife crime patrols (wardens) to protect eggs etc.
##
1
##
Colour ringing all chicks to know introduced population
##
1
##
Conservation and ecology is unpredictable, so imagination is sometimes required in management
##
1
##
Culling should not be necessary if the feasibility studies have been done correctly
##
1
##
Education (of people, not storks)
##
1
##
Habitat creation that will have more widespread conservation benefits, e.g. creation of shallow wetlands or wet meadow habitats
##
1
##
Have the right habitat they will come. An artificially supported one will not work
##
1
##
I actively do not support the provision of supplementary food beyond the initial requirements of the fledgelings.
##
1
##
I am dubious of the value of reintroducing a population if supplementary feeding is required long term in order to maintain populations. Populations should not be introduced if the habitats they are being released into cannot support their populations long term
##
1
##
I am not a supported of the reintroduction of the white stork.
##
1
##
I don't know what the evidence base is so speculation is pointless
##
1
##
I mean, it's always going to be context specific, so it's hard to say which methods of management I'd support without this information. In the context is appropriate, I would consider supporting the above methods.
##
1
##
I selected "Other" to get this text box, in order to say I would only support "discouraging nest building" in a limited way to avoid dangerous interactions, but not as a general principle.
##
1
##

I would leave that decision to those how specials in these birds, my own opinion might be at odds with what the storks actually need.
##
1
##
I would need to research this further before giving an informed response.
##
1
##
I would support all options if they were appropriate. I currently don't have enough information to decide what is, and what is not, appropriate.
##
1
##
I'd support any sensible activity to manage the population in a sustainable way
##
1
##
I'm not interested in wildlife particularly
##
1
##
improving habitat for them
##
1
##
Introduce to all appropraite areas of the UK.
##
1
##
Managing habitats so they're suitable for a whole suite of species, not just storks - build it, and they will come.
##
1
##
Measures to protect and if possible expand suitable habitat, reducing the stork's impact on other areas.
##
1
##
Methods that would help natural behaviour e.g migration
##
1
##
Monitoring impact on other wildlife populations. So while I love seeing the ring-necked parakeets in my area I have to worry about their impact on other species that nest in holes in trees like jackdaws (another favourite bird that I enjoy watching).
##
1
##
Monitoring of any adverse impacts the storks might have on other wildlife eg intensive monitoring of feeding behaviour and prey selection. This should be done to inform any damage limitation measures that may be required in the future.
##
1
##
None.
##
1
##
Not sure what you mean by 'support' - I wouldn't want to be personally involved because of other commitments
##
1
Providing support / advice for residents who have a stork nesting on their building / roof.   Personally I would be delighted - but I imagine other homeowner might be less keen.   I know in Brighton there have been issues with peregrine falcons nesting on people's balconies and it can be difficult and messy for homeowners.   The more you can educate and get local support, the more successful the project will be going forward.   I also think a talk / video at local schools (maybe via Zoom) would be really beneficial.   I live in Upper Beeding and as far as I know there has been nothing about the White Stork project in our local school - and we are only 15 mins away from Knepp.
##
1
##
Removal. They are not native.
##
1
##
Research into impact would be sensible before any management
##
1
##
Since they should not be here the best thing would be to remove them entirely but it will never happen now.This project has given a bad name to serious re-introduction projects in Sussex and probably the country.
##
1
##
Stop the project to allow nature to take its course - which is exactly what rewilding claims to do
##
1

```
##
Suplimentary feeding I'm not sure about. Ideally the countryside should be able to support them, but it is no lon
ger the same as it was before they disappeared for the UK. If it was a choice between no stork or suplimentary fe
eding I would support feeding.
##
1
##
This is the problem with the project, there is no plan for any of the above for a species that we don't even know
people wanted introduced. The fact that culling if we become over populated speaks volumes that this project is a
pointless exercise.
##
1
##
Trying to educate other people that we can't have a 'sterile'\\'passive' local environment – wildlife should alwa
ys have a place and be allowed to find new homes.
##
1
##
Whatever else is needed, particularly if the existing wildlife is affected.
##
1
##
Will Storks naturally disperse away from stork dense areas to areas with no storks? If so do nothing
##
1
##
wing tag them all so any real white storks can be readily identified
##
1
```

```r
# Clean the data
final_data$Q17.13a_other_open <- gsub("[^[:graph:]]", " ", final_data$Q17.13a_other_open)
final_data$Q17.13a_other_open <- gsub("[[:punct:]]", "", final_data$Q17.13a_other_open)# Remove punctuation
final_data$Q17.13a_other_open <- gsub("^ ", "", final_data$Q17.13a_other_open)
final_data$Q17.13a_other_open <- gsub(" $", "", final_data$Q17.13a_other_open)

# Reasons for support/not support WSP
class(final_data$Q17.13a_other_open)
```

```
## [1] "character"
```

```r
sentiment(get_sentences(final_data$Q17.13a_other_open))
```

```
##       element_id sentence_id word_count sentiment
##    1:          1           1         NA         0
##    2:          2           1         NA         0
##    3:          3           1         NA         0
##    4:          4           1         NA         0
##    5:          5           1         NA         0
##   ---
## 3527:       3527           1         NA         0
## 3528:       3528           1         NA         0
## 3529:       3529           1         NA         0
## 3530:       3530           1         NA         0
## 3531:       3531           1         NA         0
```

```r
# Word frequencies
Corpus_methods <- Corpus(VectorSource(final_data$Q17.13a_other_open))
Corpus_methods <- tm_map(Corpus_methods, removeNumbers)
Corpus_methods <- tm_map(Corpus_methods, lemmatize_strings)
Corpus_methods <- tm_map(Corpus_methods, removeWords, stopwords("english")) #removes common english stopwords
Corpus_methods <- tm_map(Corpus_methods, removeWords, c("they", "the", "also"))  #You can specify words to remove

#build a term-document matrix
TDM_methods = tm::TermDocumentMatrix(Corpus_methods, control = list(minWordLength = 1))
m_methods = as.matrix(TDM_methods)
v_methods = sort(rowSums(m_methods), decreasing = TRUE)
d_methods = data.frame(word = names(v_methods),freq=v_methods)

# Create a wordcloud
wordcloud(Corpus_methods, scale=c(5,0.5), max.words=80, random.order=FALSE, rot.per=0,
          use.r.layout=FALSE, colors=brewer.pal(8, "Dark2"))
```

```
# We can find the words that appear at least 100 times by calling the findFreqTerms() function on the term.doc.ma
trix
HiFreq_methods <- findFreqTerms(TDM_methods, 100)
HiFreq_methods
```

```
## character(0)
```

```
# Now you also see how associated a word is to another word or a list of words.
findAssocs(TDM_methods, HiFreq_methods, 0.4)
```

```
## named list()
```

```
# or, just compute word strength associations
findAssocs(TDM_methods, "long", 0.5) # Looks like the word "long" and "legs" are very frequently associated (51%
 of the time)
```

```
## $long
##       dubious      maintain    reintroduce       release         value
##          0.89          0.89           0.89          0.89          0.89
##    population          term           feed         order  supplementary
##          0.82          0.80           0.67          0.63          0.63
```

```
# Increase margin size
par(mar=c(8,6,4,4))
barplot(d_methods[1:30,]$freq, las = 2, names.arg = d_methods[1:30,]$word,
        col ="black", main ="Most frequent words (preferred methods to manage WS)",
        ylab = "Word frequencies")
```

Most frequent words (preferred methods to manage WS)

1. University of Brighton, l.jones4@brighton.ac.uk (mailto:l.jones4@brighton.ac.uk)↵