BIOS611-HW4

Yangzhenyu Gao

2025-09-24

```
############################################################
# Task 1: UFO Shape Table (U.S. only)
############################################################

# 1. Read the data
data <- fread("E:/PHD_UNC/BIOS611/data/nuforc_sightings.csv")

# 2. Filter to USA only
us_data <- data %>%
  filter(country == "USA" & state != "" & !is.na(state))

# 3. Clean the "shape" column
us_data <- us_data %>%
  mutate(shape = str_to_title(trimws(shape)))   # standardize capitalization

# Replace blanks or NAs with "Unknown"
us_data <- us_data %>%
  mutate(shape = ifelse(shape == "" | is.na(shape), "Unknown", shape))

# Clean the "state" column
us_data <- us_data %>%
  mutate(state = toupper(state)) %>%
  mutate(state = case_when(
    state == "NEW YORK"      ~ "NY",
    state == "MONTANA"        ~ "MT",
    state == "OHIO"           ~ "OH",
    state == "WEST VIRGINIA"  ~ "WV",
    state == "WISCONSIN"      ~ "WI",
    TRUE                     ~ state
  )) %>%
  filter(state %in% c(
    "AL", "AK", "AZ", "AR", "CA", "CO", "CT", "DE", "FL", "GA",
    "HI", "ID", "IL", "IN", "IA", "KS", "KY", "LA", "ME", "MD",
    "MA", "MI", "MN", "MS", "MO", "MT", "NE", "NV", "NH", "NJ",
    "NM", "NY", "NC", "ND", "OH", "OK", "OR", "PA", "RI", "SC",
    "SD", "TN", "TX", "UT", "VT", "VA", "WA", "WV", "WI", "WY")
    ## "AS", "DC", "FM", "GU", "MH", "MP", "PW", "PR", "VI", "UM"
  )

# 4. Count number of sightings for each state × shape
shape_table <- us_data %>%
  count(state, shape, name = "n") %>%
```

```r
  pivot_wider(names_from = shape, values_from = n, values_fill = 0)

# View first few rows of the matrix
head(shape_table)
```

```
## # A tibble: 6 x 25
##   state Changing Chevron Cigar Circle  Cone Cross Cylinder Diamond  Disk   Egg
##   <chr>    <int>   <int> <int>  <int> <int> <int>    <int>   <int> <int> <int>
## 1 AK          15       3    16     71     2     1       13      10    47     2
## 2 AL          40      19    42    156     3     5       35      23    85    20
## 3 AR          33      22    36    132     4     4       29      17    72     8
## 4 AZ         179      56   111    481    30    17       79      68   258    51
## 5 CA         570     255   382   1636    70    62      266     272  1028   161
## 6 CO          97      73    89    303    10    14       64      38   183    28
## # i 14 more variables: Fireball <int>, Flash <int>, Formation <int>,
## #   Light <int>, Orb <int>, Other <int>, Oval <int>, Rectangle <int>,
## #   Sphere <int>, Teardrop <int>, Triangle <int>, Unknown <int>, Star <int>,
## #   Cube <int>
```

```r
# 5. Answer questions
# Q1: How many distinct known shapes (excluding "Other" and "Unknown")?
distinct_shapes <- us_data %>%
  filter(!shape %in% c("Other", "Unknown")) %>%
  distinct(shape) %>%
  arrange(shape)

num_shapes <- nrow(distinct_shapes)
cat("Number of distinct known shapes (excluding Other/Unknown):", num_shapes, "\n")
```

```
## Number of distinct known shapes (excluding Other/Unknown): 22
```

```r
# Q2: Which state has the most 'Circle' sightings?
circle_counts <- us_data %>%
  filter(shape == "Circle") %>%
  count(state, sort = TRUE)

top_circle_state <- circle_counts %>% slice(1)
cat("State with most Circle sightings:\n")
```

```
## State with most Circle sightings:
```

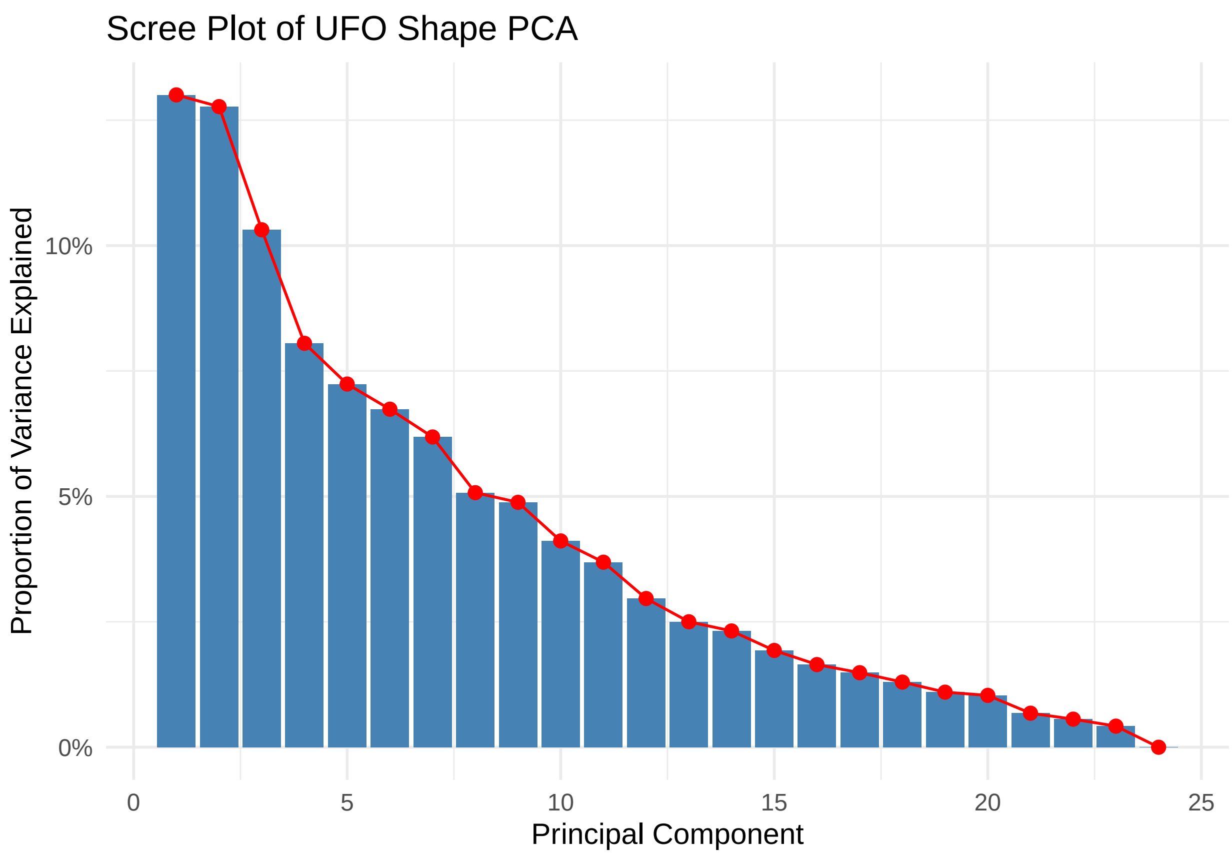
```r
print(top_circle_state)
```

```
##     state    n
## 1:     CA 1636
```

```r
cat(
  "Task 1 Summary:\n",
  "After cleaning the dataset, we retained", nrow(us_data), "U.S. reports.\n",
  "There are", num_shapes, "distinct known UFO shapes (excluding 'Other' and 'Unknown').\n",
  "The state with the most 'Circle' sightings is", top_circle_state$state,
  "with", top_circle_state$n, "reports.\n"
)
```

```
## Task 1 Summary:
##  After cleaning the dataset, we retained 138211 U.S. reports.
##  There are 22 distinct known UFO shapes (excluding 'Other' and 'Unknown').
##  The state with the most 'Circle' sightings is CA with 1636 reports.
```

```r
############################################################
# Task 2: PCA on the shape table
############################################################

# 1. Prepare data for PCA: Convert counts to proportions (row-normalize)
# The first column is 'state', which we need to preserve for labeling later
state_labels <- shape_table$state

# Create a matrix of counts only (excluding the state column)
shape_counts_matrix <- as.matrix(shape_table[, -1])

# Calculate row sums (total sightings per state)
total_sightings_per_state <- rowSums(shape_counts_matrix)

# Normalize by dividing each count by the state's total sightings
# We add a small epsilon (1e-9) to avoid division by zero for states with no sightings
shape_proportions <- shape_counts_matrix / (total_sightings_per_state + 1e-9)

# 2. Perform PCA
# We use center = TRUE and scale. = TRUE as is standard practice
pca_result <- prcomp(shape_proportions, center = TRUE, scale. = TRUE)

# 3. Create a scree plot to see variance explained by each PC
# Extract the variance explained by each component
variance_explained <- pca_result$sdev^2 / sum(pca_result$sdev^2)

# Create a data frame for plotting
scree_data <- data.frame(
  Component = 1:length(variance_explained),
  Proportion_of_Variance = variance_explained
)

# Plot the scree plot
scree_plot <- ggplot(scree_data, aes(x = Component, y = Proportion_of_Variance)) +
  geom_col(fill = "steelblue") +
  geom_line(aes(y = Proportion_of_Variance), color = "red", group = 1) +
  geom_point(color = "red", size = 2) +
  scale_y_continuous(labels = scales::percent) +
  labs(
    title = "Scree Plot of UFO Shape PCA",
    x = "Principal Component",
    y = "Proportion of Variance Explained"
  ) +
  theme_minimal()

print(scree_plot)
```

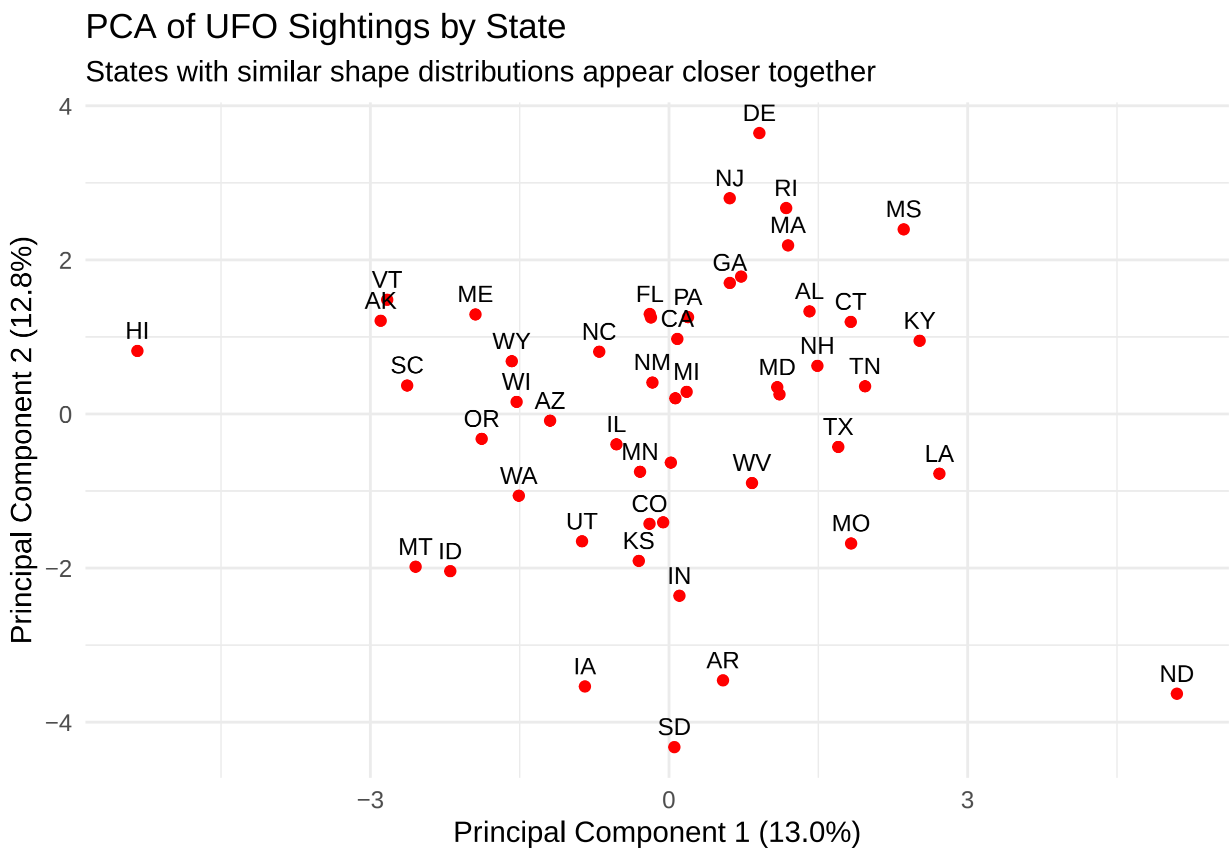



```
cat(
  "Scree Plot Analysis:\n",
  "The scree plot shows a gradual decline in variance explained, without a clear\n 'elbow' after the fi
  "PC1 accounts for", scales::percent(variance_explained[1], accuracy = 0.1), "\nand PC2 for", scales::p
  ", bringing the cumulative total for the first two to a modest\n", scales::percent(sum(variance_explai
  "This low percentage suggests that the patterns of UFO shape\n reports are highly complex and multi-fa
  "effectively reduced to just one or two dimensions, meaning the\n differences in shape distributions b
)
```

```
## Scree Plot Analysis:
##  The scree plot shows a gradual decline in variance explained, without a clear
##  'elbow' after the first few components.
##  PC1 accounts for 13.0%
## and PC2 for 12.8% , bringing the cumulative total for the first two to a modest
##  25.8% .
##  This low percentage suggests that the patterns of UFO shape
##  reports are highly complex and multi-faceted. The data cannot be
##  effectively reduced to just one or two dimensions, meaning the
##  differences in shape distributions between states are subtle and driven by many factors.
```

```
# 4. Plot the first two principal components (PC1 vs PC2)
# Create a data frame with state labels and the first two PCs
pca_scores <- as.data.frame(pca_result$x[, 1:2])
pca_scores$state <- state_labels
```

```r
# Create the scatterplot
pca_plot <- ggplot(pca_scores, aes(x = PC1, y = PC2, label = state)) +
  geom_point(color = "red") +
  geom_text(vjust = -0.7, hjust = 0.5, size = 3, check_overlap = TRUE) +
  labs(
    title = "PCA of UFO Sightings by State",
    subtitle = "States with similar shape distributions appear closer together",
    x = paste0("Principal Component 1 (", scales::percent(variance_explained[1], accuracy = 0.1), ")"),
    y = paste0("Principal Component 2 (", scales::percent(variance_explained[2], accuracy = 0.1), ")")
  ) +
  theme_minimal()

print(pca_plot)
```




```r
cat(
  "PC1 vs PC2 Plot Analysis:\n",
  "The scatterplot of the first two principal components\n reveals how states relate based on their UFO
  "Most states are clustered near the center, indicating\n a generally similar distribution of reported
  "However, there are a few potential outliers which are\n distinctly separated from the main cluster.\
)
```

```
## PC1 vs PC2 Plot Analysis:
##  The scatterplot of the first two principal components
##  reveals how states relate based on their UFO shape profiles.
##  Most states are clustered near the center, indicating
```

```
##  a generally similar distribution of reported shapes across the country.
##  However, there are a few potential outliers which are
##  distinctly separated from the main cluster.
```

```r
# 5. Examine the loadings for PC1 and PC2
# The loadings show how much each original variable (shape) contributes to a PC
loadings <- as.data.frame(pca_result$rotation[, 1:2])
loadings$shape <- rownames(loadings)

# Get the top contributors to PC1
pc1_loadings <- loadings %>%
  select(shape, PC1) %>%
  arrange(desc(abs(PC1)))

# Get the top contributors to PC2
pc2_loadings <- loadings %>%
  select(shape, PC2) %>%
  arrange(desc(abs(PC2)))

cat("Top Shape Contributors to PC1:\n")
```

```
## Top Shape Contributors to PC1:
```

```r
print(head(pc1_loadings))
```

```
##             shape        PC1
## Light       Light -0.4296310
## Flash       Flash -0.3454484
## Triangle Triangle  0.3410037
## Orb           Orb -0.3366954
## Disk         Disk  0.2631382
## Cigar       Cigar  0.2578450
```

```r
cat("\nTop Shape Contributors to PC2:\n")
```

```
##
## Top Shape Contributors to PC2:
```

```r
print(head(pc2_loadings))
```

```
##             shape        PC2
## Unknown   Unknown -0.3673008
## Circle     Circle  0.3623188
## Sphere     Sphere  0.3616667
## Diamond   Diamond  0.2819193
## Egg           Egg  0.2567357
## Fireball Fireball  0.2498301
```

```r
cat(
  "\nLoadings Analysis:\n",
  "Based on the loadings, PC1 is most strongly influenced by a\n contrast between two types of shapes. (
  "Therefore, PC1 can be interpreted as a spectrum from sightings\n of 'Unstructured Lights' (negative s
  "PC2 is primarily driven by a contrast between 'Unknown'\n (strong negative loading) and well-defined
)
```

```
##
## Loadings Analysis:
##  Based on the loadings, PC1 is most strongly influenced by a
##  contrast between two types of shapes. On the negative side are
## 'Light', 'Flash', and 'Orb' (amorphous light phenomena), while on the positive side
##  are 'Triangle', 'Disk', and 'Cigar' (classic, structured craft).
##  Therefore, PC1 can be interpreted as a spectrum from sightings
##  of 'Unstructured Lights' (negative scores) to 'Structured Objects'
##  (positive scores).
##  PC2 is primarily driven by a contrast between 'Unknown'
##  (strong negative loading) and well-defined geometric shapes
##  like 'Circle' and 'Sphere' (strong positive loadings).
```

```r
############################################################
# Task 3: Clean and tokenize the summaries
############################################################

# 1. Clean and Tokenize Summaries
# We will create a new data frame for the text analysis
# This keeps the original us_data intact
# 1. Clean and Tokenize Summaries
summary_tokens <- us_data %>%
  # Just work with the summary column
  transmute(summary = summary) %>%
  # Clean the text using the more reliable method
  mutate(summary = tolower(summary),
         summary = gsub("[^[:print:]]", " ", summary), # Replace non-printable chars with a space
         summary = str_squish(summary)) %>%            # Trim and squeeze whitespace
  # Tokenize
  unnest_tokens(word, summary)

# 2. Analyze Initial Word Frequency
# Count the frequency of each word
initial_word_counts <- summary_tokens %>%
  count(word, sort = TRUE)

cat("Top 20 most frequent words (before stopword removal):\n")
```

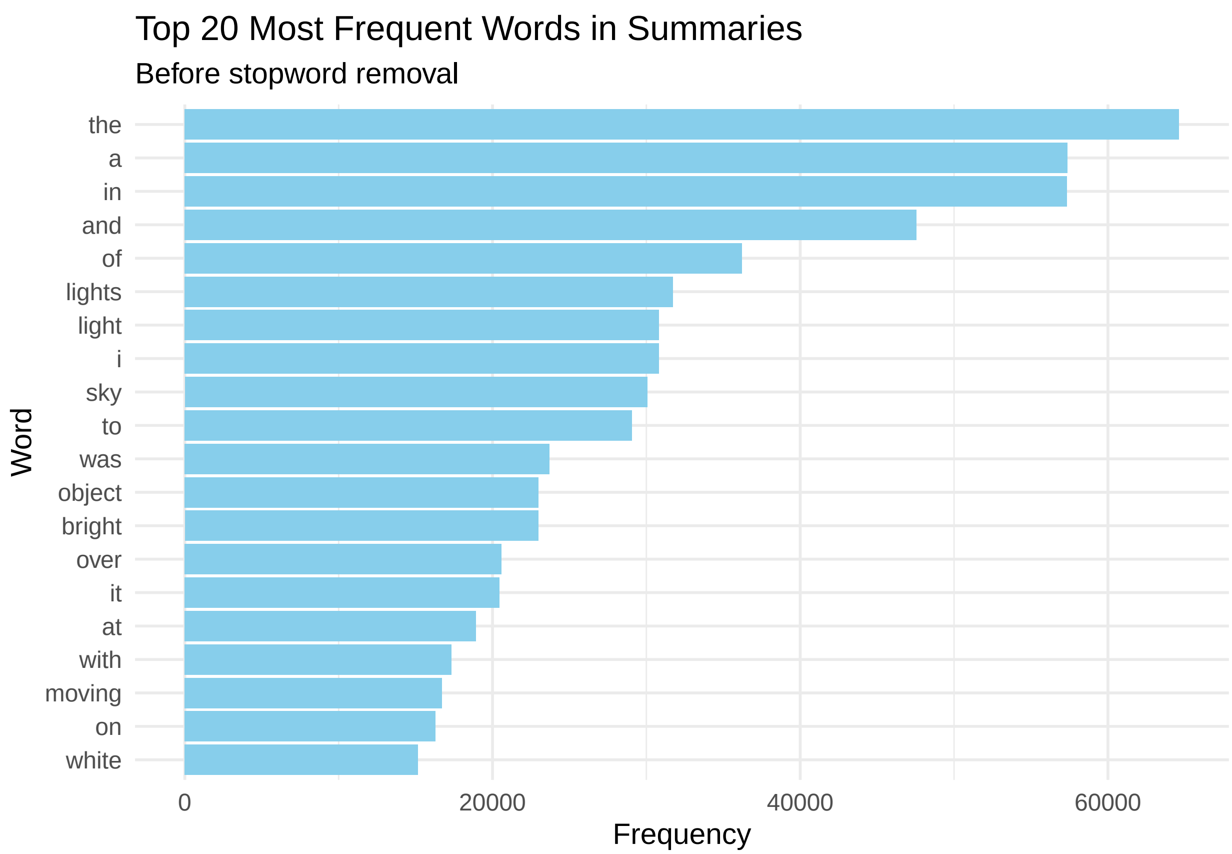
```
## Top 20 most frequent words (before stopword removal):
```

```r
print(head(initial_word_counts, 20))
```

```
##       word     n
##   1:   the 64631
##   2:     a 57371
```

```
##  3:      in 57333
##  4:     and 47559
##  5:      of 36226
##  6: lights 31730
##  7:   light 30814
##  8:       i 30810
##  9:     sky 30054
## 10:      to 29055
## 11:     was 23716
## 12: object 22979
## 13: bright 22973
## 14:    over 20574
## 15:      it 20441
## 16:      at 18928
## 17:    with 17347
## 18: moving 16712
## 19:      on 16288
## 20:   white 15166
```

```r
# Plot the most frequent words
initial_freq_plot <- initial_word_counts %>%
  head(20) %>%
  mutate(word = reorder(word, n)) %>% # Reorder for plotting
  ggplot(aes(x = word, y = n)) +
  geom_col(fill = "skyblue") +
  coord_flip() + # Makes labels easier to read
  labs(
    title = "Top 20 Most Frequent Words in Summaries",
    subtitle = "Before stopword removal",
    x = "Word",
    y = "Frequency"
  ) +
  theme_minimal()

print(initial_freq_plot)
```




```r
cat(
  "\nInitial Word Frequency Analysis:\n",
  "The initial output is dominated by common English 'stopwords'\n like 'the', 'a', 'in', 'and', 'i', '
  "These words are essential for sentence structure but provide little\n to no insight into the specifi
  "They are generic and need to be removed to uncover meaningful patterns.\n\n"
)
```

```
##
## Initial Word Frequency Analysis:
##  The initial output is dominated by common English ’stopwords’
##  like ’the’, ’a’, ’in’, ’and’, ’i’, ’of’, and ’to’.
##  These words are essential for sentence structure but provide little
##  to no insight into the specific content or context of the UFO reports.
##  They are generic and need to be removed to uncover meaningful patterns.
```

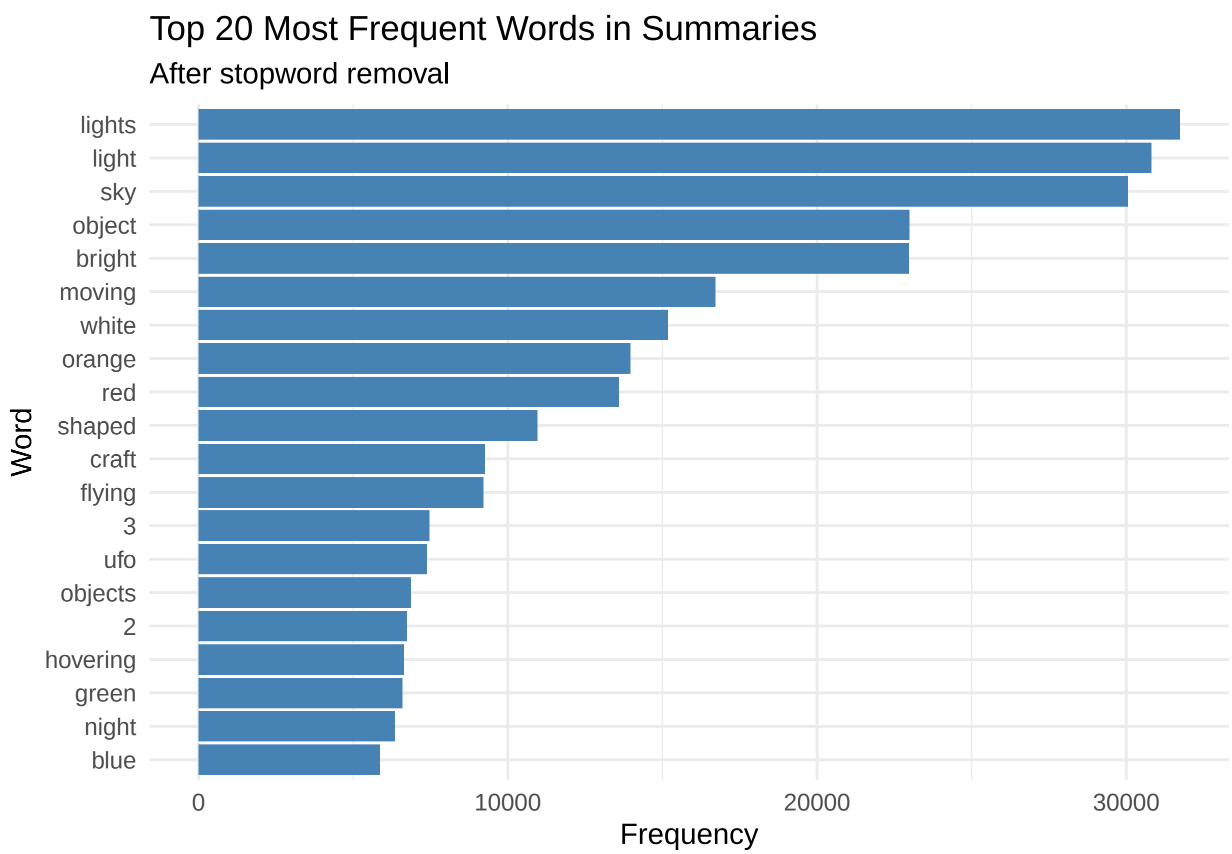
```r
# 3. Remove Stopwords and Re-analyze
# The `tidytext` package includes a dataset called `stop_words`
# We use an anti_join to remove all words present in the stop_words list
tokens_no_stopwords <- summary_tokens %>%
  anti_join(stop_words, by = "word")

# Count word frequency again
word_counts_clean <- tokens_no_stopwords %>%
  count(word, sort = TRUE)

cat("Top 20 most frequent words (after stopword removal):\n")
```

```
## Top 20 most frequent words (after stopword removal):
```

```r
print(head(word_counts_clean, 20))
```

```
##         word     n
##  1:   lights 31730
##  2:    light 30814
##  3:      sky 30054
##  4:   object 22979
##  5:   bright 22973
##  6:   moving 16712
##  7:    white 15166
##  8:   orange 13964
##  9:      red 13584
## 10:   shaped 10957
## 11:    craft  9258
## 12:   flying  9204
## 13:        3  7469
## 14:      ufo  7386
## 15:  objects  6855
## 16:        2  6741
## 17: hovering  6642
## 18:    green  6585
## 19:    night  6350
## 20:     blue  5862
```

```r
# Plot the new most frequent words
clean_freq_plot <- word_counts_clean %>%
  head(20) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(x = word, y = n)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  labs(
    title = "Top 20 Most Frequent Words in Summaries",
    subtitle = "After stopword removal",
    x = "Word",
    y = "Frequency"
  ) +
  theme_minimal()

print(clean_freq_plot)
```



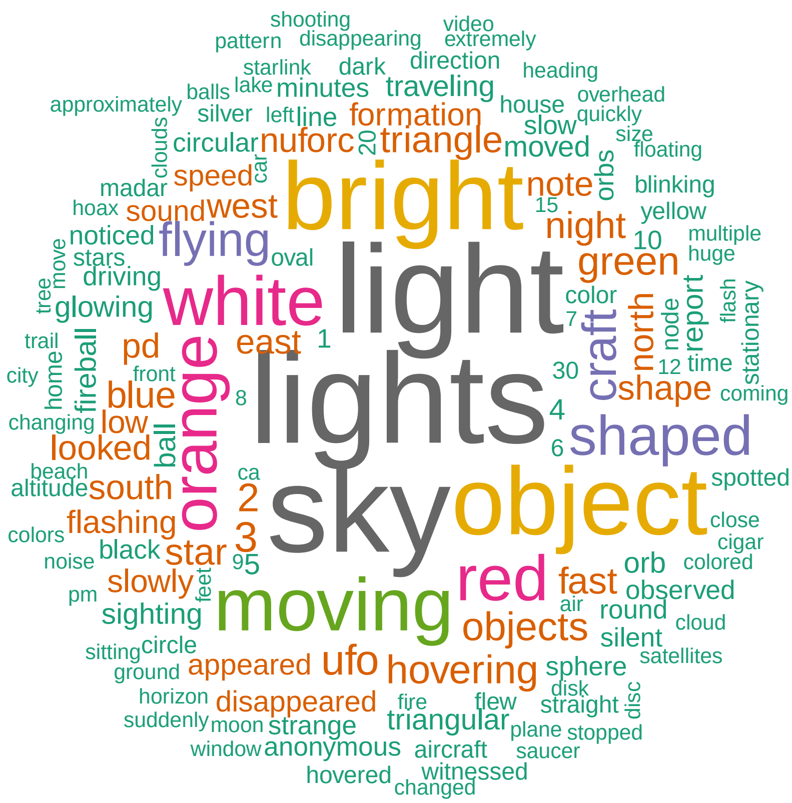

```r
cat(
  "\nAnalysis After Stopword Removal:\n",
  "After removing stopwords, the word list becomes much more\n insightful and relevant to the topic of U
  "Words like 'light', 'sky', 'object', 'lights', 'night',\n 'moving', and 'white' are now at the top.\n
  "These words feel highly characteristic of these reports,\n painting a picture of witnesses seeing lum
  "in the night sky. This provides a much clearer thematic\n summary of the dataset.\n\n"
)
```

```
##
## Analysis After Stopword Removal:
##  After removing stopwords, the word list becomes much more
##  insightful and relevant to the topic of UFO sightings.
##  Words like 'light', 'sky', 'object', 'lights', 'night',
##  'moving', and 'white' are now at the top.
##  These words feel highly characteristic of these reports,
##  painting a picture of witnesses seeing luminous, moving objects
##  in the night sky. This provides a much clearer thematic
##  summary of the dataset.
```

```r
# 4. (Optional) Create a Word Cloud
cat("Generating word cloud...\n")
```

```
## Generating word cloud...
```

```r
# Set a seed for reproducibility of the word cloud layout
set.seed(1234)

# Create the word cloud
wordcloud(words = word_counts_clean$word,
          freq = word_counts_clean$n,
          min.freq = 5, # Only plot words that appear at least 5 times
          max.words = 150, # Limit the number of words
          random.order = FALSE, # Plot most frequent words in the center
          colors = RColorBrewer::brewer.pal(8, "Dark2"))
```



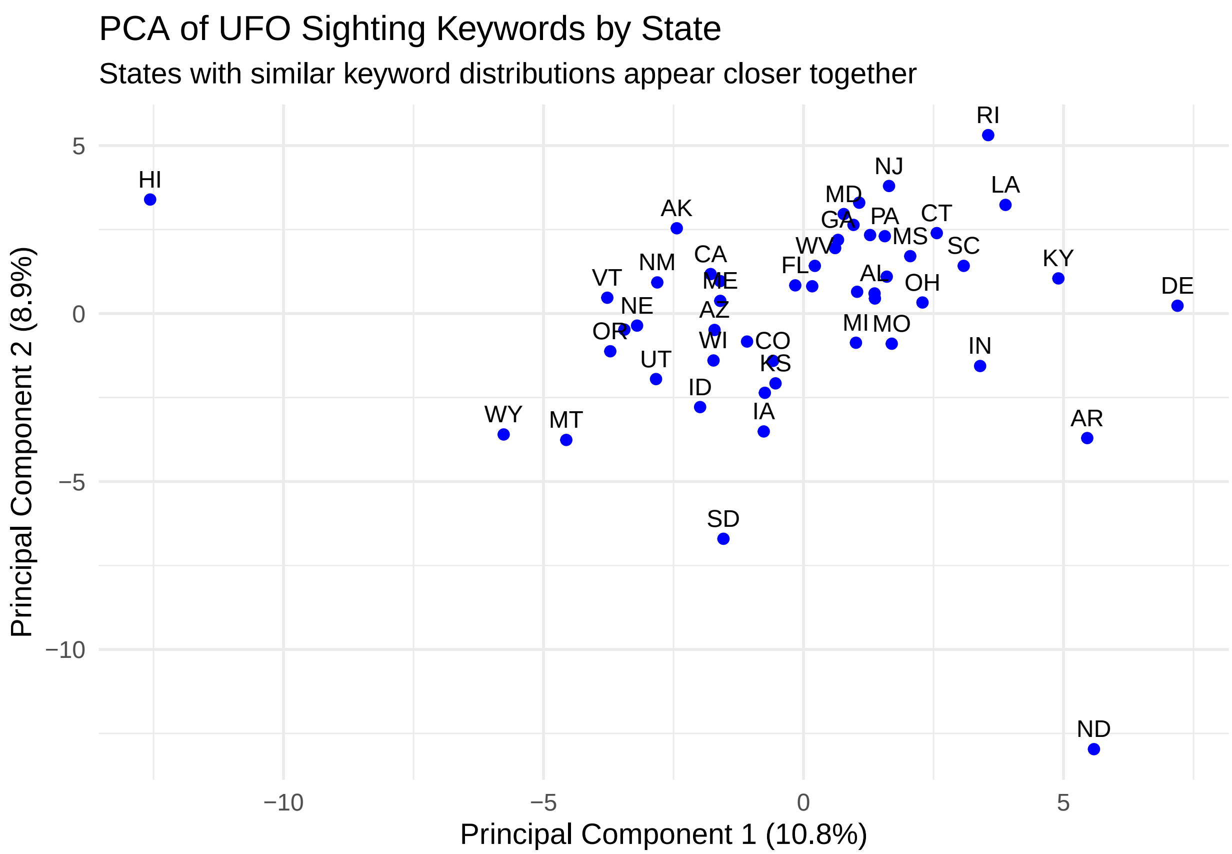

```r
################################################################
# Task 4: Build the keyword table and repeat PCA
################################################################

# 1. Define a vocabulary of the top 100 words (min 3 characters)
# We use the 'word_counts_clean' data frame created in Task 3
vocabulary <- word_counts_clean %>%
  filter(nchar(word) >= 3) %>%
  head(100) %>%
  pull(word) # Extracts the 'word' column as a character vector

cat("Vocabulary of", length(vocabulary), "words has been created.\n\n")
```

```
## Vocabulary of 100 words has been created.
```

```r
# 2. Create the new state-by-keyword table
# First, we need to re-tokenize the summaries while keeping the 'state' information
state_tokens <- us_data %>%
  select(state, summary) %>%
  # Apply the same cleaning steps from Task 3
  mutate(summary = tolower(summary),
         summary = gsub("[^[:print:]]", " ", summary),
         summary = str_squish(summary)) %>%
  # Tokenize and remove stopwords
  unnest_tokens(word, summary) %>%
  anti_join(stop_words, by = "word")

# Now, create the wide table using the defined vocabulary
state_keyword_table <- state_tokens %>%
  filter(word %in% vocabulary) %>%
  count(state, word, name = "n") %>%
  pivot_wider(names_from = word, values_from = n, values_fill = 0)

cat("State-by-Keyword table created with", nrow(state_keyword_table), "states and", ncol(state_keyword_t
```

```
## State-by-Keyword table created with 50 states and 100 keywords.
```

```r
# 3. Repeat the PCA on the new keyword table
# a. Prepare the data for PCA (row-normalize)
keyword_state_labels <- state_keyword_table$state
keyword_counts_matrix <- as.matrix(state_keyword_table[, -1])
total_words_per_state <- rowSums(keyword_counts_matrix)
keyword_proportions <- keyword_counts_matrix / (total_words_per_state + 1e-9)

# b. Perform PCA
keyword_pca_result <- prcomp(keyword_proportions, center = TRUE, scale. = TRUE)
keyword_variance_explained <- keyword_pca_result$sdev^2 / sum(keyword_pca_result$sdev^2)

# c. Plot the new PCA results
keyword_pca_scores <- as.data.frame(keyword_pca_result$x[, 1:2])
keyword_pca_scores$state <- keyword_state_labels

keyword_pca_plot <- ggplot(keyword_pca_scores, aes(x = PC1, y = PC2, label = state)) +
  geom_point(color = "blue") +
  geom_text(vjust = -0.7, hjust = 0.5, size = 3, check_overlap = TRUE) +
  labs(
    title = "PCA of UFO Sighting Keywords by State",
    subtitle = "States with similar keyword distributions appear closer together",
    x = paste0("Principal Component 1 (", scales::percent(keyword_variance_explained[1], accuracy = 0.1
    y = paste0("Principal Component 2 (", scales::percent(keyword_variance_explained[2], accuracy = 0.1
  ) +
  theme_minimal()

print(keyword_pca_plot)
```




```
# d. Examine the new loadings
keyword_loadings <- as.data.frame(keyword_pca_result$rotation[, 1:2])
keyword_loadings$word <- rownames(keyword_loadings)

cat("Top Keyword Contributors to PC1:\n")
```

```
## Top Keyword Contributors to PC1:
```

```
print(keyword_loadings %>% select(word, PC1) %>% arrange(desc(abs(PC1))) %>% head())
```

```
##                   word        PC1
## bright          bright -0.2298195
## light            light -0.2185225
## triangular triangular  0.2136757
## craft            craft  0.2014107
## nuforc          nuforc -0.1994923
## note              note -0.1910199
```

```
cat("\nTop Keyword Contributors to PC2:\n")
```

```
##
## Top Keyword Contributors to PC2:
```

```r
print(keyword_loadings %>% select(word, PC2) %>% arrange(desc(abs(PC2))) %>% head())
```

```
##                   word        PC2
## east              east -0.2577012
## south            south -0.2169841
## north            north -0.2029107
## satellites satellites -0.1897191
## west              west -0.1804516
## sphere          sphere  0.1802151
```

```r
# 4. Compare keyword PCA with shape PCA
cat(
  "\n## Final Comparison: Shape PCA vs. Keyword PCA ##\n\n",
  "The two PCA results provide complementary views of\n the UFO sighting data. The keyword-based PCA exp
  "Similarities:\n",
  "Both analyses show that most states cluster tightly\n around the origin, indicating a shared baselin
  "Conclusion:\n",
  "While the shape analysis provides a clear typology of what is seen\n, the keyword analysis validates
)
```

```
##
## ## Final Comparison: Shape PCA vs. Keyword PCA ##
##
##  The two PCA results provide complementary views of
##  the UFO sighting data. The keyword-based PCA explains less
##  variance with its initial components (19.7% vs. 25.8% for shapes), which
##  confirms that the textual descriptions are even more
##  complex and multi-faceted than the shape classifications.
##
##  Similarities:
##  Both analyses show that most states cluster tightly
##  around the origin, indicating a shared baseline
##  of how UFOs are reported across the country. Furthermore,
## states like Hawaii (HI) and North Dakota (ND) are significant
## outliers in both plots, strongly suggesting
##  that the reports from these states are consistently
##  unusual in both the shapes seen and the specific language used.
##
##  Conclusion:
##  While the shape analysis provides a clear typology of what is seen
## , the keyword analysis validates the most dominant pattern (Lights vs. Craft)
##  and enriches our understanding by revealing a secondary
## , independent pattern related to the narrative style of the report
## . The two methods work together to paint a more complete
##  picture of the UFO phenomenon.
```