

# Text analysis

*YZ Analytics*

Calculating Term Frequency and Inverse Document Frequency by points:

Code based from <https://www.tidytextmining.com/tfidf.html>.

```
Wine <- read_csv("../data/winemag-data-130k-v2.csv")
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
```

```
## cols(
##   X1 = col_double(),
##   country = col_character(),
##   description = col_character(),
##   designation = col_character(),
##   points = col_double(),
##   price = col_double(),
##   province = col_character(),
##   region_1 = col_character(),
##   region_2 = col_character(),
##   taster_name = col_character(),
##   taster_twitter_handle = col_character(),
##   title = col_character(),
##   variety = col_character(),
##   winery = col_character()
## )
```

```
Wine_points_tfidf <- Wine %>%
  unnest_tokens(word, description) %>%
  count(points, word, sort = TRUE) %>%
  bind_tf_idf(word, points, n)
```

```
Wine_points_tfidf %>%
  filter(points == 100) %>%
  arrange(desc(tf_idf)) %>%
  head()
```

```
## # A tibble: 6 x 6
##   points word          n      tf    idf  tf_idf
##   <dbl> <chr>      <int>  <dbl> <dbl>  <dbl>
## 1    100 masseto         2 0.00150  1.95 0.00292
## 2    100 frog           2 0.00150  1.66 0.00248
## 3    100 cerretalto     1 0.000749 3.04 0.00228
## 4    100 fragility     1 0.000749 3.04 0.00228
## 5    100 master's       1 0.000749 3.04 0.00228
## 6    100 proclaim       1 0.000749 3.04 0.00228
```

We see that the words with the highest TF-IDF values are the unique words in the 100-point wine descriptions that occur only 1-2 in the vocabulary of all the descriptions.

Let's look specifically at the words with the highest TF-IDF values for 80-point wines:

```
Wine_points_tfidf %>%
  filter(points == 80) %>%
```

```
arrange(desc(tf_idf)) %>%
head()
```

```
## # A tibble: 6 x 6
##   points word      n      tf   idf   tf_idf
##   <dbl> <chr>   <int>  <dbl> <dbl>   <dbl>
## 1     80 strange    19 0.00180 0.560 0.00101
## 2     80 weedy     19 0.00180 0.560 0.00101
## 3     80 acceptable 16 0.00152 0.647 0.000982
## 4     80 weird     12 0.00114 0.847 0.000965
## 5     80 pickled    18 0.00171 0.560 0.000956
## 6     80 tastes     64 0.00607 0.154 0.000936
```

These words occur more frequently than the words in the 100-point descriptions. However, the frequencies are pretty low. It might be useful to separate points into different levels (perhaps 80-86 is low rating, 97-93 is medium, and 94-100 is high).

```
Wine$rating <- cut(Wine$points,
                   breaks=c(-Inf, 86, 93, Inf),
                   labels=c("low", "medium", "high"))
```

```
Wine_rating_tfidf <- Wine %>%
  unnest_tokens(word, description) %>%
  count(rating, word, sort = TRUE) %>%
  bind_tf_idf(word, rating, n)
```

```
Wine_rating_tfidf %>%
  filter(rating == "high") %>%
  arrange(desc(tf_idf)) %>%
  head()
```

```
## # A tibble: 6 x 6
##   rating word      n      tf   idf   tf_idf
##   <fct> <chr> <int>  <dbl> <dbl>   <dbl>
## 1 high  2025   226 0.000673 0.405 0.000273
## 2 high  2030   219 0.000653 0.405 0.000265
## 3 high  2023   152 0.000453 0.405 0.000184
## 4 high  2026    84 0.000250 0.405 0.000101
## 5 high  2035    81 0.000241 0.405 0.0000979
## 6 high  2027    77 0.000229 0.405 0.0000930
```

We can also look at TF-IDF based on variables other than points. For example, we can look at TF-IDF values based on variety of wine.

```
wine_words <- Wine %>%
  unnest_tokens(word, description) %>%
  count(variety, word, sort = TRUE)
```

```
plot_wine <- wine_words %>%
  bind_tf_idf(word, variety, n) %>%
  mutate(word = fct_reorder(word, tf_idf)) %>%
  mutate(variety = factor(variety, levels = c("Pinot Noir",
                                              "Cabernet Sauvignon",
                                              "Chardonnay",
                                              "Red Blend",
                                              "Riesling")))
```

```
plot_wine %>%
  group_by(variety) %>%
  top_n(15, tf_idf) %>%
  ungroup() %>%
  mutate(word = reorder(word, tf_idf)) %>%
  ggplot(aes(word, tf_idf, fill = variety)) +
  geom_col(show.legend = FALSE) +
  labs(x = NULL, y = "tf-idf") +
  facet_wrap(~variety, ncol = 2, scales = "free") +
  coord_flip()
```



```

wine_explore <- Wine %>%
  select(description, points) %>%
  mutate(description = gsub('[:punct:] ]+', ' ', tolower(description)))

words <- str_split(wine_explore$description, ' ')
all_words <- data.frame(points = rep(wine_explore$points, sapply(words, length)), words = unlist(words))

words_grouped <- all_words %>%
  group_by(words) %>%
  summarize(
    points = mean(points),
    count = n()
  ) %>%
  filter(count > 10) %>%
  arrange(desc(points))

top <- words_grouped[1:10,] %>% cbind(top_bottom = 'top')
bottom <- words_grouped[(nrow(words_grouped) - 9):nrow(words_grouped),] %>% cbind(top_bottom = 'bottom')
top_bottom <- rbind(top, bottom)

ggplot(top_bottom, aes(x = reorder(words, points), y = points, fill = top_bottom)) +
  geom_bar(stat = 'identity') +
  coord_flip() +
  scale_fill_manual(values = c('#00b4fb', '#fa6560')) +
  ggtitle('Wine Review Words with the Highest and Lowest Mean Points', subtitle = NULL) +
  xlab('Average Points') +
  ylab('Word') +
  labs(fill = 'Top or Bottom')

```

