

Recuperación y análisis de texto con R

Clase 5 - Educación Permanente FCS

Mag. Elina Gómez (UMAD)

elina.gomez@cienciassociales.edu.uy

www.elinagomez.com

Mag. Gustavo Méndez Barbato

gustavo.mendez@cienciassociales.edu.uy



Este trabajo se distribuye con una licencia Creative Commons Attribution-ShareAlike 4.0 International License

Objetivos de hoy

- Presentación del paquete **quanteda** para el análisis de textos.
- Nubes de palabras
- Asociaciones
- Redes

Paquete quanteda

- **quanteda** es un paquete R para administrar y analizar datos textuales desarrollados por *Kenneth Benoit* y otros colaboradores. Su desarrollo inicial fue apoyado por la beca del Consejo Europeo de Investigación.
- El paquete está diseñado para usuarios de R que necesitan aplicar el procesamiento de lenguaje natural a los textos, desde los documentos originales hasta el análisis final.
- Sus capacidades coinciden o superan las que se ofrecen en muchas aplicaciones de software para usuarios finales, muchas de las cuales son caras y no de código abierto.
- Documentación quanteda
- Más información

Caso práctico: LUC en el Senado

- Análisis de los discursos vinculados a la discusión de la *Ley de Urgente Consideración (LUC)* en la Cámara de Senadores del 5 de junio de 2020.

***“Limpieza”* del texto**

- Creo un Document feature matrix (DFM), aplicando algunos argumentos que me permiten limpiar las palabras que no me interesan al efecto del análisis.
 - Homogeeinizo las palabras en minúscula
 - Elimino números
 - Elimino puntuaciones
 - Elimino stopwords (por defecto y lista propia con palabras varias (ej. “Risas”))
 - Elimino palabras con pocos caracteres (1 y 2)

“Limpieza” del texto

```
dfm_intervenciones <- quantda::dfm(quantda::tokens(intervenciones$speech,
remove_punct = TRUE, ##saco puntuación
remove_numbers = TRUE), #saco números
tolower=TRUE, #paso a minúsculas
verbose = FALSE) %>%
  quantda::dfm_remove(pattern = c(quantda::stopwords("spanish"),tolower(intervenciones$legislator)),
min_nchar=3)%>% ##saco palabras específicas
  quantda::dfm_trim(min_termfreq = 6)%>%
  quantda::dfm_group(groups = intervenciones$party) #defino grupos
```


Ponderación

- Como factor de ponderación del dfm puedo usar la métrica *tf-idf* que relativiza el peso de cada término, poniendo en relación la frecuencia de aparición por el inverso de la frecuencia en los documentos.
- Ayuda a identificar los términos más frecuentes en un documento pero que no lo son en todos.
- No es posible usarlo en funciones que impliquen agrupación.
- La función de quanteda es *dfm_tfidf()*

Nubes de palabras: general

Las nubes de palabras las hago con la función **textplot_wordcloud** del paquete **quanteda.textplot**

```
quanteda.textplots::textplot_wordcloud(dfm_tfidf(dfm_intervenciones), min.count = 2,max_words = 200,  
  random.order = FALSE,colors = RColorBrewer::brewer.pal(8,"Dark2"),comparison = F)
```



Nubes de palabras: grupos

Para hacer nubes de palabras comparando entre grupos de interés, agregamos el argumento **comparison = T**

```
quanteda.textplots::textplot_wordcloud(dfm_intervenciones, min.count = 2,max_words = 500,  
  random.order = FALSE,colors = RColorBrewer::brewer.pal(8,"Dark2"),comparison = T)
```

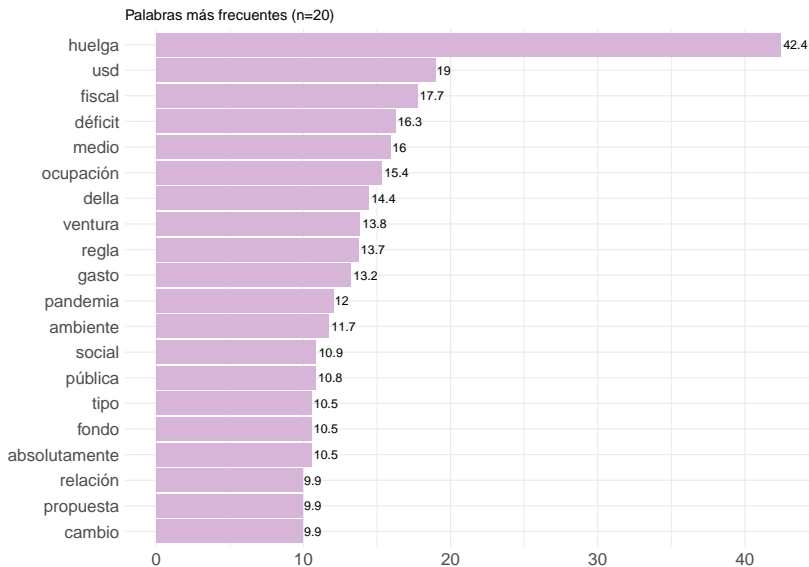


Palabras más frecuentes

Para analizar las palabras más frecuentes utilizo la función *topfeatures()*

```
topfeatures(dfm_intervenciones,20)
```

Palabras más frecuentes:



Asociación de palabras

- Buscamos la asociación de palabras en los documentos
- Analizamos la asociación con las palabras: *sindicato*, *reforma* # Asociación de palabras
- Utilizamos la función *textstat_simil* del paquete **quantda.textstats**, cuyos argumentos son el/los términos con los que quiere buscar una asociación en un *dfm* determinado.
- Defino el método de similitud (“correlation”, “cosine”, “jaccard”, “ejaccard”, “dice”, “edice”, “hamman”, “simple matching”)

```
quantda.textstats::textstat_simil(dfm_tfidf(dfm_intervenciones),selection = "sindicato",
                                method = "correlation",margin = "features")>%
  as.data.frame()%>%
  dplyr::arrange(-correlation)%>%
  dplyr::top_n(15)
```


Asociación de palabras: \$sindicato

feature1	feature2	correlation
plantea	sindicato	1
brasil	sindicato	1
pbi	sindicato	1
período	sindicato	1
inflación	sindicato	1
pone	sindicato	1
mayores	sindicato	1
recién	sindicato	1
obras	sindicato	1
ayer	sindicato	1
estrategia	sindicato	1
plantear	sindicato	1
asimismo	sindicato	1
preopinante	sindicato	1
poblaciones	sindicato	1
aquellas	sindicato	1
ordenamiento	sindicato	1

Asociación de palabras: \$reforma

feature1	feature2	correlation
emergencia	reforma	1.0000000
mercado	reforma	1.0000000
relación	reforma	0.9999932
conjunto	reforma	0.9999705
enfrentar	reforma	0.9999058
razonable	reforma	0.9999058
pregunta	reforma	0.9999058
interno	reforma	0.9999058
mujeres	reforma	0.9999058
integral	reforma	0.9999058
entiende	reforma	0.9999058
cálculo	reforma	0.9999058

Contexto de la palabra: kwic

- Con la función `quanteda::kwic()` podemos ver el contexto de aparición de una palabra, término o frase, según una ventana (cantidad de palabras previas y posteriores) determinada.
- Extraer el contexto de ciertos términos puede ser de utilidad para construir un nuevo corpus y realizar un análisis focalizado y/o comparativo.

```
quanteda::kwic(quanteda::tokens(intervenciones$speech,  
remove_punct = TRUE,  
remove_numbers = TRUE),  
pattern = quanteda::phrase(c("ley de urgente consideración")),  
window = 5)
```

Contexto de la palabra: redes de co-ocurrencia

- Con la función `quanteda.textplots::textplot_network` podemos hacer redes de co-ocurrencia entre términos.

Contexto de la palabra: redes de co-ocurrencia

