DH@Madrid Summer School 2016

Análisis de textos poéticos y estilometría con R

S. Ros, A. Robles, A. C. Caminero

{sros, arobles, accaminero}@scc.uned.es

Madrid, 27 junio al 1 de julio de 2016

European Research Council

Digital Scholarly Editions Initial Training Network

Índice

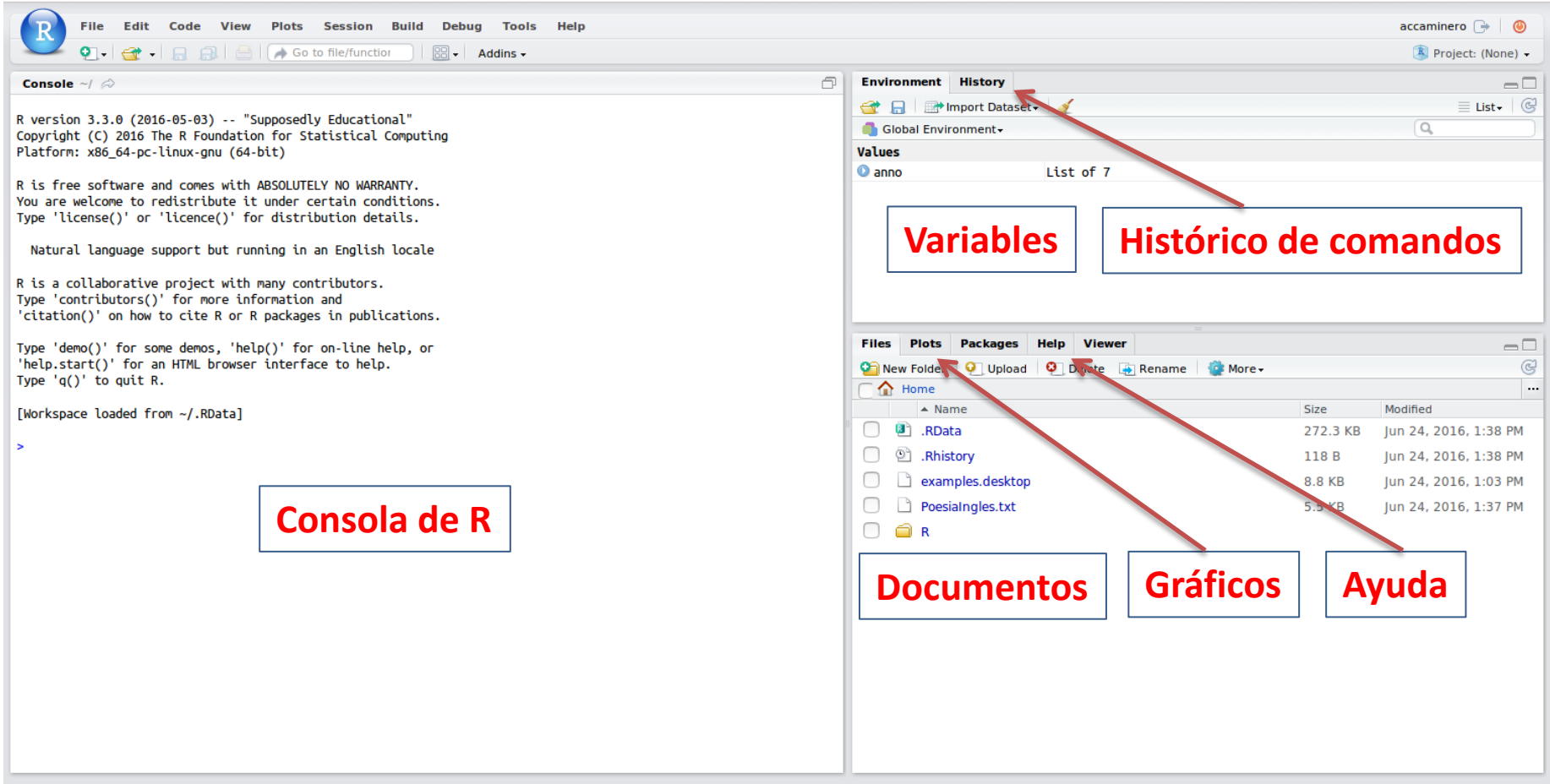
- Introducción al trabajo práctico
- Introducción al procesamiento del lenguaje natural (Natural Language Processing, NLP)
- Pasos de NLP
- Introducción al análisis de datos textuales
- Estilometría
- Referencias de interés

Introducción al trabajo práctico

- Utilizaremos un servidor RStudio Server para realizar los ejercicios de este curso.
- Los datos de acceso son:
  - http://62.204.199.197:8787/
  - Usuario y clave: el que os hemos proporcionado
- Trabajaremos sobre ficheros de textos poéticos en inglés.

Introducción al trabajo práctico

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES



erc European Research Council

Introducción al trabajo práctico

- Notación:
  - En esta presentación los segmentos de código y sus salidas se muestran en un recuadro como el siguiente:

```
➢# Comentario
➢OrdenDeR()
    Salida de la orden
```

Introducción al trabajo práctico

LiNHDi
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Utilizaremos la librería CoreNLP, que se encuentra instalada en el servidor.
- El primer paso para empezar a trabajar será cargar e inicializar la librería CoreNLP:

```
➢ library(coreNLP)
➢ initCoreNLP()
    Searching for resource: config.properties
    …
    Adding annotator sentiment
```

- Seguidamente, cargaremos en el servidor el documento PoesiaIngles.txt

Introducción a NLP

- NLP trata sobre la interacción persona-ordenador a través del lenguaje.
- Entre los retos a los que NLP se enfrenta podemos encontrar los siguientes:
  - Ambigüedad y variabilidad del lenguaje
    - Mismo concepto con varios significados y varios conceptos con el mismo significado.
  - Escalabilidad:
    - Dominios, idiomas, medio (oral, texto), frases largas...
  - …

Pasos del NLP

- Los pasos básicos de los que consiste el procesamiento del lenguaje natural son los siguientes:
  - *Tokenization* y *sentence splitting.*
  - *Lemmatization* y *POS tagging*.
  - Dependencias.
  - Reconocimiento de entidades con nombre.
  - Correferencias.

Pasos del NLP: *Tokenization* y *sentence splitting*

- Tokenizar consiste en separar los caracteres que forman el texto en palabras (tokens).
  - Ej: «Mi casa es bonita» → «Mi», «casa», «es», «bonita».
  - Es necesario tener en cuenta los espacios en blanco pero también los signos de puntuación.

Pasos del NLP: *Tokenization* y *sentence splitting*

- *Sentence splitting* es la consecuencia de la tokenización, y consiste en definir a qué frase pertenece cada uno de los tokens del texto a analizar.
- Las frases acaban con caracteres de fin de frase (., !, o ?)

Pasos del NLP: *Tokenization* y *sentence splitting*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Órdenes:

  - #Anotamos el fichero de interés
  - anno<-annotateFile("/home/**USUARIO**/PoesiaIngles.txt")
  - anno

    A CoreNLP Annotation:
    num. sentences: 32
    num. tokens: 1157

erc European Research Council

Pasos del NLP: *Tokenization* y *sentence splitting*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- # Mostramos cada token (palabra) que hemos identificado
- getToken(anno)$token

```
 [1] "The"        "curfew"      "tolls"       "the"
 [5] "knell"      "of"          "parting"     "day"
  ...
```

- # Mostramos a qué frase pertenece cada token
- getToken(anno)$sentence

```
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[31] 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
...
```

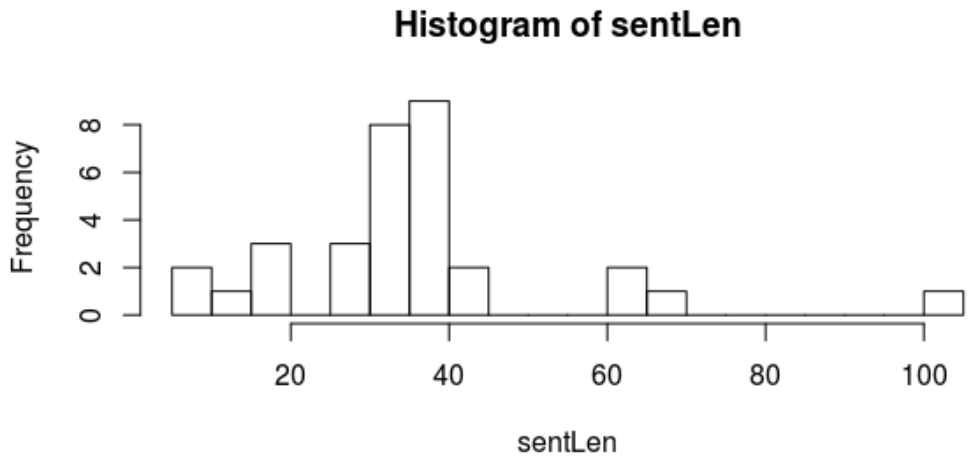
erc European Research Council

Pasos del NLP: *Tokenization* y *sentence splitting*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- sentLen <- table(getToken(anno)$sentence)
- # sentLen contiene la longitud de cada frase

- # Pintamos gráfico de longitudes de frases
- hist(sentLen, breaks=30)



Pasos del NLP: *Lematization* y *POS tagging*

- Mientras que *tokenizar* es dividir los caracteres en grupos (palabras), lematizar es producir la forma canónica de las palabras, lo que se conoce como su ***lema***.
  - Por ejemplo, «camiones» → «camión», «fue» → «ir»
- El proceso de lematizar es distinto para cada palabra, según sea nombre, verbo, …

European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

- El *POS tagging* consiste en asignar a cada token una etiqueta que identifica su categoría.
  - Ejemplo: verbo, pronombre personal, …
- Hay distintos conjuntos de categorías:
  - Penn Treebank Project: https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html
  - Universal tag set: https://github.com/slavpetrov/universal-pos-tags

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

Penn Treebank Project		Universal tag-set
VB	Verbo, forma base	VERB
VBD	Verbo, pasado	
VBG	Verbo, gerundio	
VBN	Verbo, participio	
VBP	Verbo, presente pero no en 3º persona del singular	
VBZ	Verbo, presente en 3º persona del singular	
NN	Nombre, singular	NOUN
NNS	Nombre, plural	
NNP	Nombre propio, singular	
NNPS	Nombre propio, plural	

erc European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Órdenes:

➢# Obtener la info de cada token
➢token <- getToken(anno)

➢# Presentar la info de los tokens de la primera frase
➢ token[token$sentence==1,c(1:9)]

sentence id token lemma CharacterOffsetBegin CharacterOffsetEnd POS
1 1 1 The the 0 3 DT
…
NER Speaker
1 O PER0

erc European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- # ¿Cuántas ocurrencias hay de cada etiqueta?
- # Utilizando Penn Treebank
- table(token$POS)

```
  ``   ,   :   .   ''  CC  CD  DT  EX  IN
  20  83  18  32   1  46   3 131   2 111 ...
```

- # Utilizando el universal tagset
- ut <- universalTagset(token$POS)
- table(ut)

```
ut
   .  ADJ  ADP  ADV CONJ  DET NOUN  NUM PRON  PRT
VERB    X
 160  118  111   36   46  135  279    3   76   36  156    1 ...
```

erc European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- #Calculamos la cuenta de los nombres, pronombres, … en cada frase

- nounCnt <- tapply(ut == "NOUN", token$sentence, sum)
- pronCnt <- tapply(ut == "PRON", token$sentence, sum)
- adjCnt <- tapply(ut == "ADJ", token$sentence, sum)
- verbCnt <- tapply(ut == "VERB", token$sentence, sum)

- # Agrupamos los contadores en un dataframe
- posDf <- data.frame(nounCnt,pronCnt,adjCnt,verbCnt)

erc European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- #Mostramos las primeras posiciones del dataframe, que muestra las primeras frases
- head(posDf)

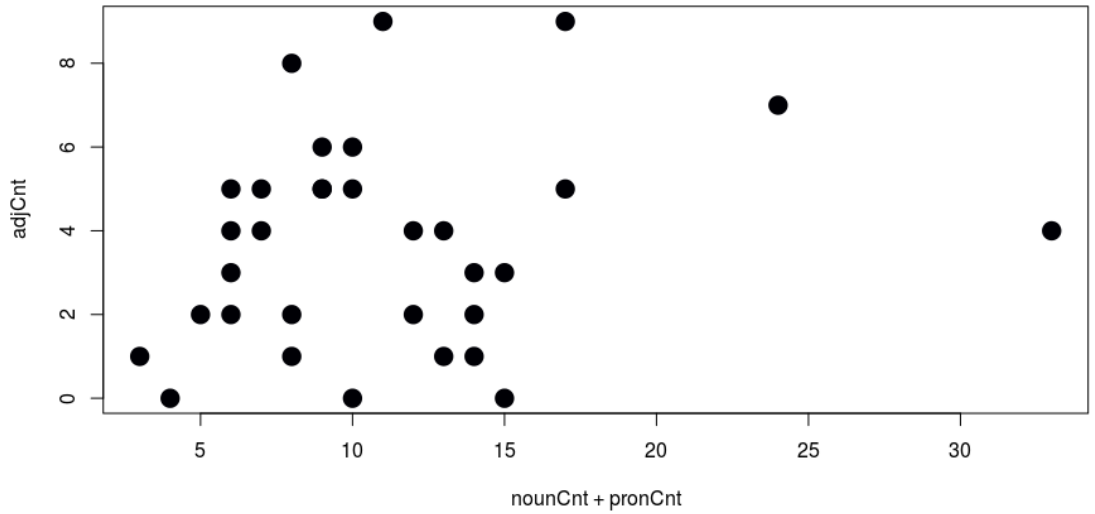
	nounCnt	pronCnt	adjCnt	verbCnt
1	12	2	1	3
2	14	3	9	12
3	8	1	5	3
4	7	2	5	5
5	10	4	3	7
6	8	4	2	6

erc European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

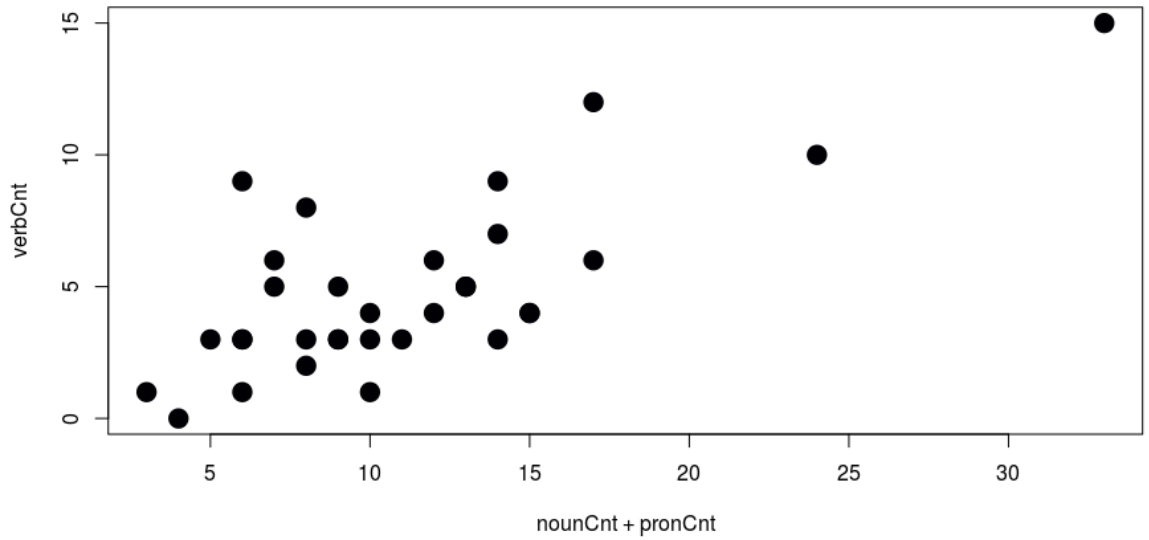
- # Pintamos un gráfico que compara cuántos nombres+pronombres hay con los adjetivos
- plot(nounCnt+pronCnt,adjCnt,pch=19,cex=2, col=rgb(0,0,0.02))



Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- # Pintamos un gráfico que compara cuántos nombres+pronombres hay con los verbos
- plot(nounCnt+pronCnt,verbCnt,pch=19,cex=2, col=rgb(0,0,0.02))



Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- #¿Cuáles son los 5 nombres que más se repiten?

- #Usando Universal Tagset
- index <- which(ut=="NOUN")
- tab <- table(token$lemma[index])
- head(sort(tab,decreasing=TRUE),5)

```
  eye  soul  wood   air breast
    3     3     3     2     2
```

erc European Research Council

Pasos del NLP: *Lematization* y *POS tagging*

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- #¿Cuáles son los 5 nombres que más se repiten?

```
# Usando Penn Treebank Project
index <- which(token$POS == "NNP")
tab <- table(token$lemma[index])
head(sort(tab,decreasing=TRUE),5)
```

```
    Ev Heaven  Chill Cromwell  Death
     2      2      1       1      1
```

Pasos del NLP: Dependencias

- Esta fase consiste en el análisis gramatical de las frases, creando una estructura que enlace las partes identificadas anteriormente.
- El resultado de esta fase es un **árbol de dependencias**.
- Las relaciones entre los lemas son binarias y sus partes son:
  - Governor.
  - Dependent.
- Además, cada relación está etiquetada con un código que identifica dicha relación.

European Research Council

Pasos del NLP: Dependencias

- Algunos de los códigos de dependencias más significativos son los siguientes:
  - nsubj: nominal subject, sujeto nominal.
    - “Clinton defeated Dole”→ nsubj (defeated, Clinton).
  - amod: adjectival modifier, modificador adjetival.
    - “Sam eats red meat” → amod (meat, red).
  - appos: appositional modifier, modificador aposicional.
    - Sam, my brother, arrived → appos(Sam, brother).
  - cc: coordinación.
    - “Bill is big and honest” → cc (big, and)
- http://nlp.stanford.edu/software/dependencies_manual.pdf

Pasos del NLP: Dependencias

- Órdenes:

  - #Generamos el árbol de dependencias, y vemos su longitud
  - parseTree <- getParse(anno)
  - length(parseTree)

    [1] 32
  - #Muestra el árbol para la primera frase
  - cat(parseTree[1])

```
(ROOT
  (S
    (S
      (NP (DT The) (NN curfew) (NNS tolls)) …
```

Pasos del NLP: Dependencias

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- # Mostramos las dependencias para la primera frase
- dep <- getDependency(anno)
- dep[dep$sentence == 1,]

```
 sentence  governor dependent     type governorIdx
dependentIdx govIndex depIndex
1        1     ROOT    knell     root         0         5      NA
5
2        1    tolls      The      det         3         1       3
1
3        1    tolls    curfew  compound         3         2
3       2
```

Pasos del NLP: Dependencias

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- #Analizar cuáles son los 3 verbos que toman como sujeto una palabra determinada en mayor número de ocasiones. Por ejemplo “toll”

- index <- which(token$lemma[dep$depIndex] == "toll")
- depSelf <- dep[index,]
- depSelf <- depSelf[depSelf$type == "nsubj",]
- sort(table(depSelf$governor),decreasing=TRUE)[1:3]

```
 knell leaves   <NA>
1     1      1
```

erc European Research Council

Pasos del NLP: Reconocimiento de entidades con nombre

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Esta fase consiste en identificar los elementos del texto y asignarles una categoría semántica.
  - Ejemplos de categorías: fecha, duración, persona, …
- Órdenes:

```
token <- getToken(anno)
#Mostramos cuántos tokens hemos encontrado para cada categoría
table(token$NER)

DATE DURATION    MISC   NUMBER      O  PERSON    TIME
  11        2       3        3   1134       3       1
```

European Research Council

Pasos del NLP: Reconocimiento de entidades con nombre

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- #Mostramos los tokens etiquetados como DURATION
- unique(token$lemma[token$NER=="DURATION"])

  [1] "day" "year“

- #Mostramos los tokens etiquetados como PERSON
- unique(token$lemma[token$NER=="PERSON"])

  [1] "rich" "Milton" "Cromwell "

erc European Research Council

Pasos del NLP: Reconocimiento de entidades con nombre

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- # Para una de las personas identificadas en el documento, ¿cuáles son las 3 primeras palabras con las que tiene dependencias de tipo gobernador, y de tipo dependiente?
- index <- which(token$lemma[dep$depIndex] == "Cromwell")
- depSelf <- dep[index,]
- sort(table(depSelf$governor),decreasing=TRUE)[1:3]

```
guiltless     <NA>     <NA>
        1       NA       NA
```

- sort(table(depSelf$dependent),decreasing=TRUE)[1:3]

```
Cromwell    <NA>    <NA>
       1      NA      NA
```

Pasos del NLP: Correferencias

- Sirven para extraer relaciones semánticas entre los tokens.
  - Detectan cuándo varias palabras se refieren a la misma persona u objeto.
  - «Elisa estaba cansada. Ella había trabajado mucho.»

Pasos del NLP: Correferencias

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Órdenes:

➢ #Calculamos y mostramos las primeras correferencias
➢ coref <- getCoreference(anno)
➢ head(coref)

```
corefId sentence start end head startIndex endIndex
1       1        1    4  14   5          4        13
2       1        2   24  25   24         60       60 …
```

➢ # Mostramos de forma agregada las palabras que forman parte de una correferencia
➢ table(token$token[coref$startIndex[coref$corefId == 10]])

```
 The their Their  them  they
   1    5    2    1    1
```

European Research Council

Introducción al análisis de datos textuales

- Hemos visto cómo anotar un texto con NLP… ¿y ahora qué?
- Partiendo de las anotaciones de NLP, se pueden aplicar una serie de técnicas para explorar y visualizar un corpus de documentos de texto.

European Research Council

Estilometría: R

- Dos características comunes en los estudios de estilometría:
  - Los textos se interpretan numéricamente
  - Los números se analizan estadísticamente

- Existe algo que sea :
  - Flexible
  - Fácil
  - Completo
  - Gratis
- R es una solución
  - Bien documentado
  - Comunidad activa

Estilometría: paquete Stylo ()

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Stylo es un paquete que aporta herramientas:
  - Stylo()
  - Classify()
  - Oppose()
  - Rolling.delta()
  - Rolling.classify()
- Muchas otras funciones secundarias

erc European Research Council

Estilometría: stylo()

- Permite **cargar y procesar** un corpus de textos.
- **Realiza** análisis estilométrico
  - *estadísticas multivariable*
- Para **visualizar y evaluar**
  - *Similaridades entre los textos de entrada*

Estilometría: stylo()

- Clave : MOST FREQUENT WORD
- Genera una tabla

PALABRAS	TEXTO1	TEXTO2	TEXTO3	TEXTO4	TEXTO
La	34	1	5	0	125
casa	3	0	89	0	123
grande	678	67	7	56	23
azul	4	89	45	8	5
bolsa	5	7	0	0	57

- Ficheros:
  - table_with_frecuencies.txt
  - wordlist.txt
  - stylo_config.txt

Estilometría: stylo()

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Se crea un directorio. Defecto corpus
- Guardamos los textos en formato
  - txt, xml (TEI) y html
  - Deben ser UTF-8
- Ejecutamos stylo()
  - Se genera gráfico
  - Se generan ficheros txt que se pueden ver y reusar (dependiendo)

erc European Research Council

Estilometría: stylo() ejemplo

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

```
#INICIANDO EL PAQUETE STYLO ----

install.packages("stylo")
library(stylo)

#CARGAR DATOS EJEMPLO ---- novels, lee, galbraith

data("novels")
data(lee)

#VER DATOS DE NOVELS es una lista en R ----

novels$ABronte_Agnes[1:3]
novels[[1]][1:2]
my.test <- stylo(gui='FALSE',parsed.corpus=novels)

# GENERA ficheros con los datos
# Cargar datos generados en una variable de R ----

dato <- read.table('table_with_frequencies.txt')
```

erc European Research Council

Estilometría: stylo() ejemplo

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

> stylo(gui = "FALSE", parsed.corpus = novels)

Available variables:

- distance.table final distances between each pair of samples
  features features (e.g. words, n-grams, ...) applied to data
- features.actually.used features (e.g. frequent words) actually analyzed
  frequencies.0.culling frequencies of words/features accross the Corpus
- list.of.edges edges of a network of stylometric similarities
- table.with.all.freqs frequencies of words/features accross the corpus
- table.with.all.zscores z-scored frequencies accross the corpus

These variables can be accessed by typing e.g.:
my.test$distance.table

erc European Research Council

Estilometría: stylo() corpus.format

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

> Stylo(corpus.format=“ XXXX”)

- Plain text
- xml (elimina etiquetas y cabeceras TEI)
- xml.drama (elimina etiquetas, cabeceras TEI, nombres entre etiquetas <speaker>)
- xml. (elimina toda etiqueta entre <head>)
- Html (intenta eliminar todas las etiquetas)

erc European Research Council

Estilometría: stylo() corpus.lang

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

> Stylo(corpus.lang=“ XXXX”)

- Intenta borrar los pronombres
- English
  - don’t : don t
  - Topsy-turvy: topsy turvy
- English.contr
- English.all
- Y más…. Ver documentacion.

erc European Research Council

Estilometría: stylo() features

> Stylo(analyzed.features="X", ngram.size=X, preserve.case=TRUE/FALSE, [TBD])

- Analyzed.features
  - "w": Word , "c": carácter
- ngram.size
  - 1 es palabras o carácter individuales
- Preserve.case
  - Mayúsculas, minúsculas
- TBD permite seleccionar ficheros del corpus.

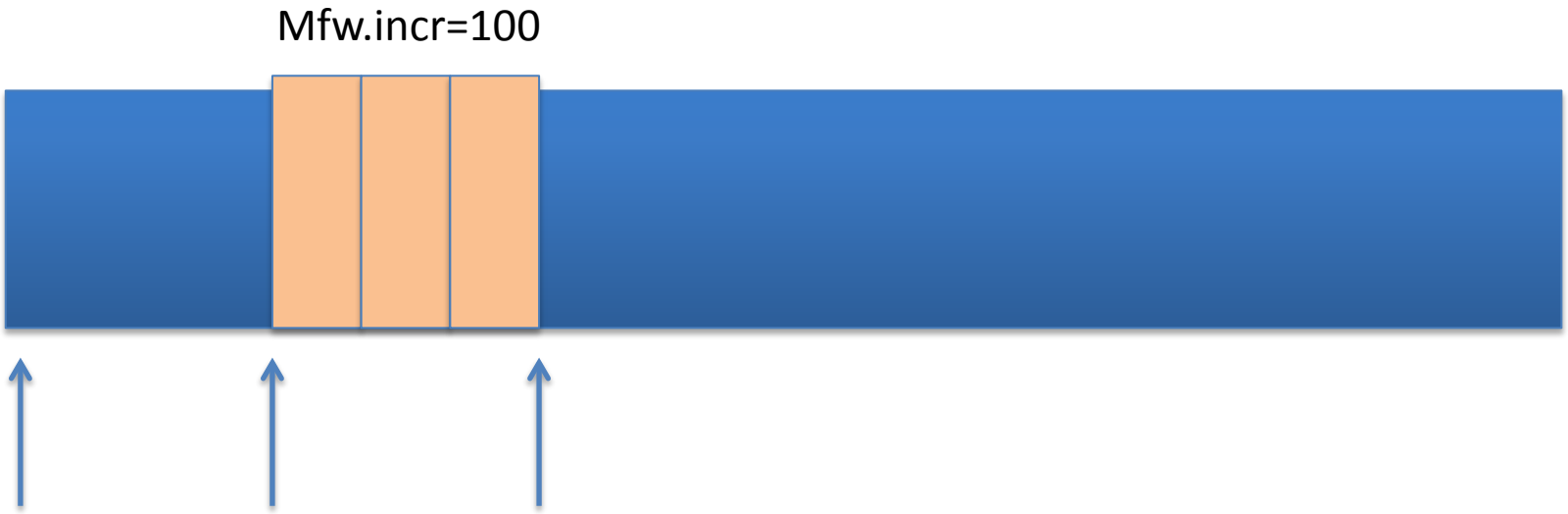
European Research Council

Estilometría: stylo()
MFW settings

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

➢ **Stylo(mfw.min=X, mfw.max=X, mfw.incr=X, start.at=X)**

Lista de palabras y sus frecuencias ordenadas de mayor a menor



Start.at=1 Mfw.min=300 Mfw.max=600

Calcula con 300, 400,500,600 MFW

Estilometría: stylo() culling settings

Se refiere al grado en el cual palabras que no aparecen en todos los textos deben ser quitados

>stylo(culling.min=X, culling.max=X, culling.incr=X)

Estilometría: stylo()
Statistics

LiNHDH
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Son técnicas no supervisadas
- Stylo (Analysis.type="XXX")
  - Cluster analysis (CA)
  - MSD: multidimensional scaling (MDS)
  - PCA, principal componet analysis (PCV,^CR)
  - Consensus Tree, (BCT)

erc European Research Council

Estilometría: stylo() distance.measure

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Distancia Euclidea: dist.euclidean

$$\delta_{(AB)} = \sqrt{\sum_{i=1}^{n} |(A_i)^2 - (B_i)^2|}$$

where:
n = the number of MFWs (most frequent words),
A, B = texts being compared,
A_i = the frequency of a given word i in the text A,
B_i = the frequency of a given word i in the text B.

erc European Research Council

Estilometría: stylo() distance.measure

- Distancia Manhattan: dist.manhattan

$$\delta_{(AB)} = \sum_{i=1}^{n} |A_i - B_i|$$

Estilometría: stylo() distance.measure

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Delta Burrows: dist.delta

$$\Delta_{(AB)} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{A_i - \mu_i}{\sigma_i} - \frac{B_i - \mu_i}{\sigma_i}\right|$$

where:
n = the number of MFWs (most frequent words or other features),
A, B = texts being compared,
A_i = the frequency of a given feature i in the text A,
B_i = the frequency of a given feature i in the text B,
μ_i = mean frequency of a given feature in the corpus,
σ_i = standard deviation of frequencies of a given feature.

Argamon (2008) showed that the above formula can be simplified algebraically:

$$\Delta_{(AB)} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{A_i - B_i}{\sigma_i}\right|$$

erc European Research Council

Estilometría: stylo() distance.measure

LiNHD
LABORATORIO DE INNOVACIÓN EN HUMANIDADES DIGITALES

- Distancia Euclidea: dist.euclidean

$$\delta_{(AB)} = \sqrt{\sum_{i=1}^{n} |(A_i)^2 - (B_i)^2|}$$

where:
n = the number of MFWs (most frequent words),
A, B = texts being compared,
A_i = the frequency of a given word i in the text A,
B_i = the frequency of a given word i in the text B.

erc European Research Council

Referencias de interés

- T. Arnold, L. Tilton. Humanities Data in R: Exploring Networks, Geospatial Data, Images, and Text (2015). Springer.
- Manning, C. D.,et al. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.
- Eder, M., Kestemont, M. and Rybicki, J. (2013). Stylometry with R: A suite of tools. In: Digital Humanities 2013: Conference Abstracts. University of Nebraska--Lincoln, NE, pp. 487-89.

European Research Council

DH@Madrid Summer School 2016

Análisis de textos poéticos y estilometría con R

S. Ros, A. Robles, A. C. Caminero

{sros, arobles, accaminero}@scc.uned.es

Madrid, 27 junio al 1 de julio de 2016

European Research Council

Digital Scholarly Editions Initial Training Network