

Presentación Displex

En este documento se presenta una librería de herramientas en lo que pretende que ser una guía rápida de uso básico para el cálculo de la disponibilidad léxica, la centralidad léxica y las diferentes funciones que de esta última se derivan. Aunque la librería permite, a través de programación funcional, implementar múltiples modelos, se proporciona un conjunto de herramientas básicas prediseñadas para que se pueda empezar a usar con las funcionalidades señaladas (disponibilidad, centralidad) de forma casi inmediata. Sin embargo, consideramos necesario hacer notar que estas funciones de aplicación directa representan tan solo un subconjunto de las posibilidades que permite el sistema aquí presentado.

Actualmente está en preparación otro trabajo en el que se discuten en profundidad los modelos posibles y diferentes opciones de aplicación, así como un estudio en profundidad de diferentes aspectos de las bases teóricas del cálculo de la disponibilidad y su interpretación. Sin embargo, el desarrollo de estas herramientas posteriores requiere cierta destreza en el manejo de la sintaxis de R, así como con los conceptos de programación funcional, con lo que su presentación requiere otro ámbito de exposición.

Exposición y contextualización del sistema

El sistema se ha implementado en R¹, una implementación abierta del lenguaje de programación S, que lleva en desarrollo desde 1992 y con una primera versión estable en el año 2000. Es fácilmente extensible y permite la reutilización de código y datos, soportado por una amplia comunidad a lo largo del mundo y se usa de forma habitual en entornos de análisis de datos de ingeniería y medicina.

RStudio² es un entorno comercial de desarrollo para R que permite el uso personal gratuito y que aumenta la productividad en el uso de este lenguaje. Este sistema está disponible para la gran mayoría de plataformas y sistemas operativos. Para facilitar el trabajo, se proporcionan utilidades que se pueden ejecutar dentro de este entorno y que facilitan el proceso de análisis.

Dentro del sistema construido en RStudio nos resulta especialmente interesante la posibilidad de usar el paradigma de la *programación literaria*, según la cual, en el mismo documento, se van integrando los contenidos textuales como la exposición de hechos o la interpretación de los análisis, con bloques de código que llevan a cabo las manipulaciones y representaciones de datos. Este documento ejemplifica esta metodología. Como ejemplo de sus ventajas, al incluir en el mismo documento el análisis que se ha llevado a cabo con el código para realizarlo, se potencia la replicabilidad, característica cada vez más exigida en el mundo académico, en la que se debe permitir que otros investigadores puedan replicar, tanto para verificar como para desarrollar, el trabajo expuesto. Este objetivo se cumple al exponer en el mismo documento la disertación y el código con el que se ha realizado los análisis.

A diferencia de soluciones basados en programas autónomos, la implementación de esta solución dentro de un sistema como el formado por R junto con RStudio permite, además, la integración de todo el sistema de gestión de datos. Esto es, los resultados obtenidos a partir de las herramientas proporcionadas se incorporan al sistema, con lo que permite la integración en nuevos procesos de datos y la aplicación personal de los resultados obtenidos. En este documento se pueden encontrar algunos de estos desarrollos que pueden servir de ejemplo.

La distribución de las soluciones es otro problema que ya ha encontrado soluciones eficientes dentro de la comunidad. Repositorios de código como GitHub permite la gestión y control automatizada de versiones. Es posible llevar a cabo un control exhaustivo de las modificaciones que se van realizando, sin necesidad de

¹<https://www.r-project.org/>

²<https://rstudio.com/>

estar bajando e instalando continuamente programas. Con dos líneas de código aseguras que has instalado la última versión disponible. Además, este modelo de intercambio de código permite que los proyectos no desaparezcan, ya que se pueden crear ramas de cualquier repositorio de código para continuar su desarrollo, independientemente de los creadores originales.

Entre las múltiples herramientas construidas sobre R encontramos muy interesantes y productivas las herramientas que conforman el “Universo Tidyverse”³. Son un conjunto de librerías que facilitan y permiten una manipulación expresiva de los datos, basada en el concepto de tubería. Este tratamiento se construye mediante la secuenciación de manipulaciones, que se van encadenando, siendo el resultado de un paso la fuente de datos para la siguiente. Esta característica, junto con operadores adaptados a este tipo de trabajo proporciona una forma potente y cómoda de llevar a cabo transformaciones que, de otra forma serían muy complejas. Las manipulaciones de datos presentadas en este trabajo se realizan utilizando estas herramientas.

Instalación

Para el uso de esta herramienta es obligatorio la instalación del sistema R y se recomienda la instalación de RStudio, que permitirá la utilización de las facilidades de entorno de usuario. Se pueden encontrar en sus respectivas páginas, en las secciones de descarga, instaladores para múltiples sistemas operativos.

Una vez instalada la aplicación RStudio, para poder utilizar las herramientas que proponemos, el usuario debe llevar a cabo un paso previo de instalación que, en nuestra opinión, no es más complejo que cualquier proceso de instalación en el sistema R. Este paso ha de hacerse únicamente cuando se pretenda instalar o actualizar el paquete en el sistema. Puesto que nuestra intención es seguir trabajando en el paquete, sería recomendable realizar esta acción de forma periódica o, al menos, a medida que se vayan actualizando funcionalidades a las ahora presentadas.

Las siguientes órdenes instalan el paquete Displex del repositorio GitHub. Todo el código está implementado en R, con lo que es fácil de revisar.

```
install.packages("devtools")
library(devtools)
install_github("jmss70/displex")
```

Para instalar las herramientas del universo Tidyverse se utiliza la orden:

```
install.packages("tidyverse")
```

En este documento, además, y con el objetivo de mejorar la presentación de las tablas, se va a utilizar el paquete **kable**, que instalaremos con:

```
install.packages("kableExtra")
```

```
## Installing package into '/home/jmss/R/x86_64-pc-linux-gnu-library/4.0'
## (as 'lib' is unspecified)
```

Llevados a cabo los pasos anteriores, y si no se ha producido ningún contratiempo, el sistema está preparado para llevar a cabo el trabajo.

Carga de las librerías y los datos

Para poder usar las librerías hay que cargarlas en nuestra sesión de trabajo, poniendo a nuestra disposición las funciones que proporcionan:

```
library(displex)
library(kableExtra)
library(tidyverse)
```

³<https://www.tidyverse.org/>

Para cargar los datos, se espera que estén en un determinado formato, utilizado por ser el que habitualmente se encuentra en este tipo de estudios, creemos que por razones históricas. Sin embargo, consideramos que son redundantes y que habría que estudiar, en un futuro cercano, establecer un estandar de codificación que sea más coherente con los modelos de datos normalizados.

Se espera que los datos estén en un archivo de texto, con campos separados por espacios: - Un campo de información del hablante - Un campo de identificación de usuario - Un campo de identificación de centro de interés - Una lista de palabras separadas por comas y en orden de realización

Un ejemplo de dos líneas sería:

```
21131 001 01 mano, pie, brazo, cerebro, pulmón, nariz, extremidad, ojo, boca, diente, pelo, oreja, culo
12131 002 01 riñón, corazón, garganta, cabeza, pierna, pie, hígado, estómago, mano, brazo, antebrazo, al
```

Suponiendo que tenemos todos los datos cargados en un archivo, denominado `datos.txt`, que estará alojado en el mismo directorio que el script de procesamiento, se podrían cargar los datos como:

```
data <- read.displex("datos.txt")
data %>%
  head() %>%
  kbl() %>%
  kable_styling(full_width = F)
```

infos	users	centers	words
21131	001	01	mano, pi....
12131	002	01	riñón, c....
12213	003	01	brazo, m....
22214	004	01	brazo, o....
12214	005	01	cabeza,
22213	006	01	pie, man....

Cálculo de la disponibilidad

La función general del cálculo de la disponibilidad es `displex_availability`. Sin embargo, el uso de esta función requiere del uso de varios parámetros. Se han construido dos funciones de utilidad que encapsulan su uso y ofrecen los dos modelos que consideramos, por el momento, más interesantes. Consideramos que es preferible exponer su uso mediante ejemplos.

Modelo de López-Strassburger

Este modelo, que es el conocido y el último de una serie de iteraciones para la cuantificación de la disponibilidad, se puede usar como:

```
disponibilidad <- build.lopezstrass.availability(data)
```

Se construye entonces un nuevo marco de datos cuyos campos son el centro de interés, la palabra y la disponibilidad calculada, junto con otros datos respecto a los que se ha expresado interés, como la posición y las frecuencias relativas y acumuladas dentro del centro de interés:

```
head(disponibilidad) %>%
  kbl() %>%
  kable_styling(full_width = F)
```

centers	words	order	availability	freq.abs	freq.rel	freq.abs.cum	freq.rel.cum
01	ojo	1	0.6329875	58	0.0373952	58	0.0373952
01	brazo	2	0.5518081	50	0.0322373	108	0.0696325
01	pierna	3	0.5072154	49	0.0315925	157	0.1012250
01	mano	4	0.4894822	42	0.0270793	373	0.2404900
01	cabeza	5	0.4780536	37	0.0238556	410	0.2643456
01	pie	6	0.4501276	43	0.0277240	331	0.2134107

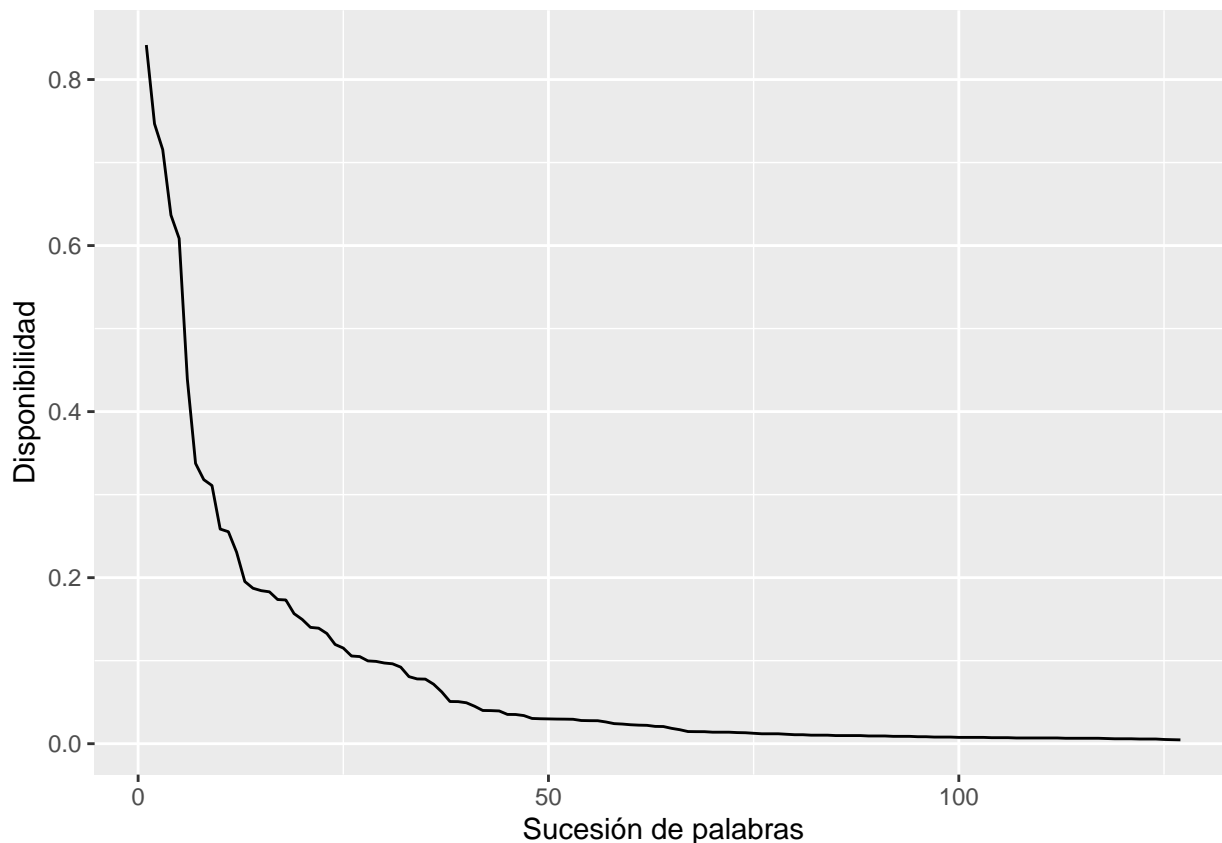
Al ser un marco de datos de R estándar, se pueden realizar sobre el todas las operaciones que permite el sistema, como seleccionar los términos del centro de interés “03”, ordenarlos en orden decreciente de disponibilidad y ver los 10 más disponibles

```
disponibilidad %>%
  filter(centers=="03") %>%
  arrange(-availability) %>%
  head(10) %>%
  kbl() %>% kable_styling(full_width = F)
```

centers	words	order	availability	freq.abs	freq.rel	freq.abs.cum	freq.rel.cum
03	cocina	1	0.8415694	69	0.0669253	69	0.0669253
03	(cuarto de) baño	2	0.7466699	66	0.0640155	135	0.1309408
03	salón	3	0.7155625	58	0.0562561	255	0.2473327
03	dormitorio	4	0.6367281	55	0.0533463	310	0.3006790
03	terrazza	5	0.6085068	62	0.0601358	197	0.1910766
03	comedor	6	0.4391517	39	0.0378274	349	0.3385063
03	salita	7	0.3376176	31	0.0300679	447	0.4335597
03	lavadero	8	0.3180560	33	0.0320078	416	0.4034918
03	pasillo	9	0.3109779	34	0.0329777	383	0.3714840
03	(cuarto de) aseo	10	0.2585570	24	0.0232784	500	0.4849661

O representar la curva de disponibilidad, esto es, la sucesión de valores de disponibilidad, una vez ordenados en valor decreciente de disponibilidad:

```
disponibilidad %>%
  filter(centers=="03") %>%
  arrange(-availability) %>%
  ggplot(aes(x=order, y=availability)) + geom_line() +
  xlab("Sucesión de palabras") + ylab("Disponibilidad")
```



Modelo de Ávila-Sánchez

Ávila y Sánchez proponen un macro-modelo para el estudio de la disponibilidad, a partir de la Teoría de los Conjuntos Difusos y mediante la modelización de los conceptos que se pretenden representar en estudios. Esta evaluación se produce en dos etapas. En la primera se cuantifica la relevancia de cada término en las pruebas obtenidas para cada hablante y centro de interés según una ley descendente según se avanza en cada listado, y en una segunda etapa se integra esa información con una ley aditiva que integra los distintos valores alcanzados para cada palabra en cada centro de interés.

Hay múltiples posibles elecciones, pero en distintas pruebas pareció más prometedoras las que utilizaban en la primera etapa una ley de Zipf-Mandelbrot y en la segunda una adición probabilística. La interpretación de los valores obtenidos corresponden a la “centralidad” de cada término en el centro de interés. Un valor de uno o muy cercano respondería a la pertenencia al núcleo del vocabulario específico del centro de interés, mientras que una valor próximo a cero indicaría que sería un término poco accesible.

Se ha implementado una función que lleva a cabo este análisis, `displex_avilasanchez_availability`, que se utiliza de la misma forma que el anterior:

```
disponibilidad <- build.avilasanchez.availability(data)
```

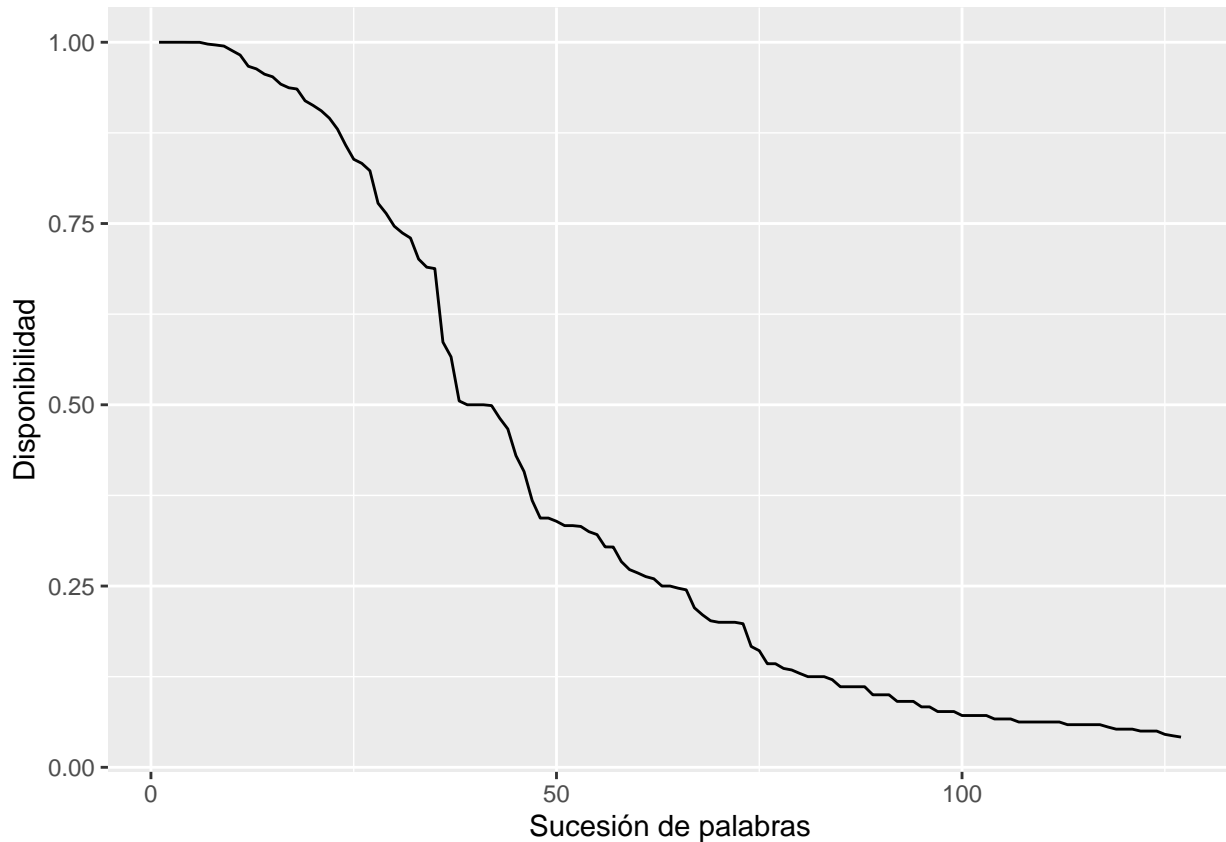
Obteniéndose, de nuevo, un nuevo marco de datos con la disponibilidad de cada término en cada centro de interés

```
disponibilidad %>%
  filter(centers=="03") %>%
  arrange(-availability) %>%
  head(10) %>%
  kable() %>%
  kable_styling(full_width = F)
```

centers	words	order	availability	freq.abs	freq.rel	freq.abs.cum	freq.rel.cum
03	cocina	1	1.0000000	69	0.0669253	69	0.0669253
03	salón	2	1.0000000	58	0.0562561	255	0.2473327
03	(cuarto de) baño	3	1.0000000	66	0.0640155	135	0.1309408
03	dormitorio	4	0.9999996	55	0.0533463	310	0.3006790
03	comedor	5	0.9999278	39	0.0378274	349	0.3385063
03	terrazza	6	0.9999003	62	0.0601358	197	0.1910766
03	salita	7	0.9974380	31	0.0300679	447	0.4335597
03	entrada	8	0.9961966	23	0.0223084	523	0.5072745
03	(cuarto de) aseo	9	0.9947151	24	0.0232784	500	0.4849661
03	pasillo	10	0.9884745	34	0.0329777	383	0.3714840

Que se puede procesar como cualquier marco de datos del sistema R:

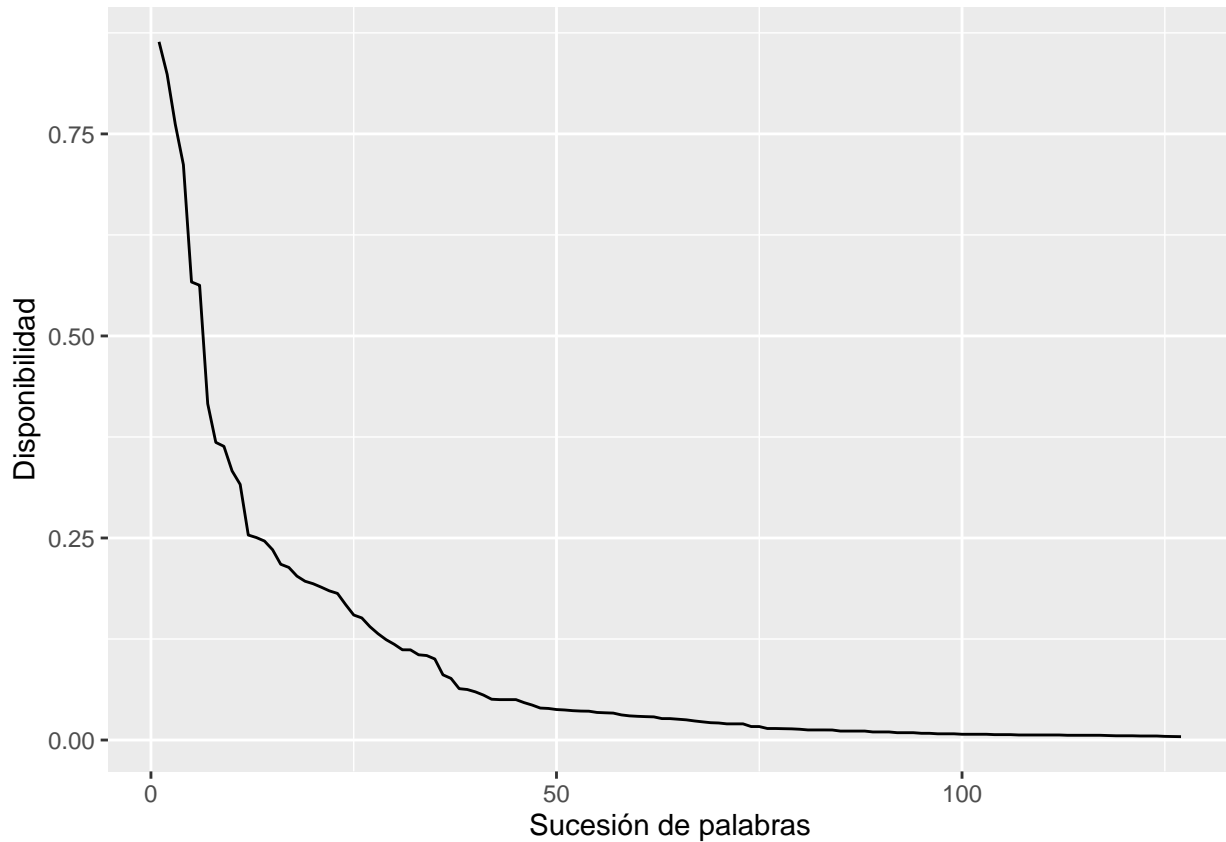
```
disponibilidad %>%
  filter(centers=="03") %>%
  arrange(-availability) %>%
  ggplot(aes(x=order, y=availability)) + geom_line() +
  xlab("Sucesión de palabras") + ylab("Disponibilidad")
```



Debido a las características de los operadores aditivos de la teoría de los conjuntos difusos, es posible que la forma de la curva no sea la esperada, no habiendo términos con valoraciones cercanas a uno (por ejemplo, si se tienen pocos datos y relativamente dispersos) o demasiados términos con valoraciones cercanas a uno (por ejemplo, si se tienen muchas muestras). Considerando esta situación, se puede regular la curva mediante un parámetro adicional, k , que va a “subir” o “bajar” la curva, pero manteniendo su forma y clasificación. El valor por defecto de k es uno. Si se da un valor entre 0 y 1, la curva “bajará” (menos términos con valores cercanos a uno), mientras que si a k se le da un valor mayor que la unidad la curva “subirá” (más términos

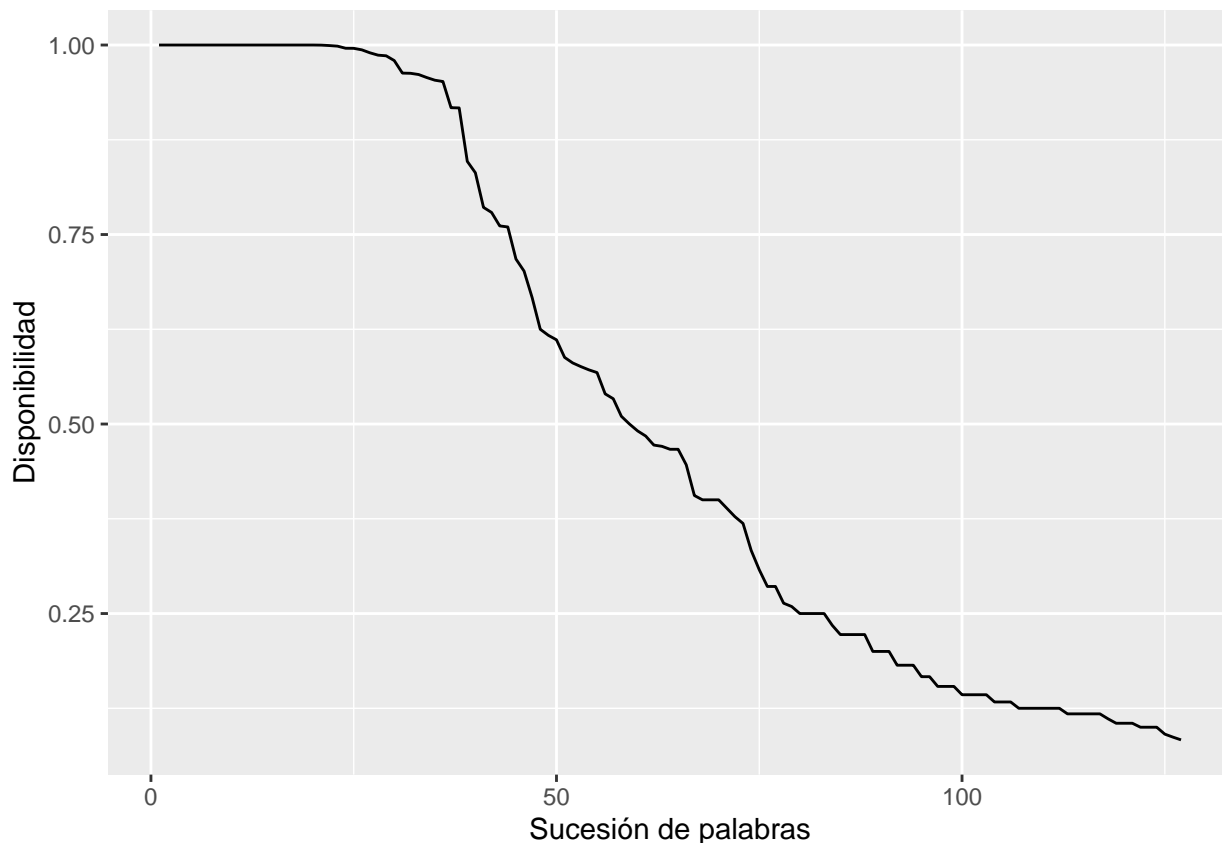
cercanos a la unidad).

```
disponibilidad <- build.avilasanchez.availability(data, k = 0.1)
disponibilidad %>%
  filter(centers=="03") %>%
  arrange(-availability) %>%
  ggplot(aes(x=order, y=availability)) + geom_line() +
  xlab("Sucesión de palabras") + ylab("Disponibilidad")
```



Obteniéndose, de nuevo, un nuevo marco de datos con la disponibilidad de cada término en cada cnetro de interés

```
disponibilidad <- build.avilasanchez.availability(data, k = 2)
disponibilidad %>%
  filter(centers=="03") %>%
  arrange(-availability) %>%
  ggplot(aes(x=seq_along(availability), y=availability)) + geom_line() +
  xlab("Sucesión de palabras") + ylab("Disponibilidad")
```



Téngase en cuenta que estos dos ejemplos son a modo de exposición y para valores muy extremos. En nuestras pruebas hemos encontrado que el valor de referencia es lo suficientemente bueno en todos los casos que hemos encontrado.

Niveles de disponibilidad

Una pregunta recurrente es considerar cual es el tamaño del conjunto de elementos que se consideran que forman parte del núcleo de un centro de interés. Para responder a esta cuestión, y a partir del marco teórico que se ha utilizado en nuestra propuesta, se proporciona una herramienta que etiqueta los términos en niveles de centralidad. El nivel 0 corresponderían a aquellos elementos que no pertenecen al núcleo, es decir, aquellos términos que no son generalmente accesibles. Los niveles 1, 2, 3, ... y sucesivos representarían un mayor grado de centralidad en el centro de interés.

```
disponibilidad <- build.avilasanchez.availability(data)
```

```
levels <- classify.availability.levels(disponibilidad)
```

```
levels %>%
  head(20) %>%
  arrange(-availability) %>%
  select(-order) %>%
  kbl() %>%
  kable_styling(full_width = F)
```


centers	words	availability	level	cutlevel	freq.abs	freq.rel	freq.abs.cum	freq.rel.cum
14	perro	1	6	0.9497065	67	0.0420590	67	0.0420590
12	coche	1	6	0.9589061	65	0.0571680	65	0.0571680
19	rojo	1	6	0.9614247	66	0.0525478	203	0.1616242
03	cocina	1	6	0.9638074	69	0.0669253	69	0.0669253
19	azul	1	6	0.9614247	68	0.0541401	137	0.1090764
14	gato	1	6	0.9497065	64	0.0401758	131	0.0822348
04	mesa	1	6	0.9562246	54	0.0518234	230	0.2207294
03	salón	1	6	0.9638074	58	0.0562561	255	0.2473327
17	euro	1	6	0.9439173	45	0.0401427	45	0.0401427
09	aire acondicionado	1	6	0.9583219	58	0.0782726	118	0.1592443
06	tenedor	1	6	0.9551142	59	0.0552952	59	0.0552952
11	árbol	1	6	0.9384632	51	0.0404762	51	0.0404762
08	pizarra	1	6	0.9505229	62	0.0476923	62	0.0476923
10	calle	1	6	0.9527790	51	0.0378338	51	0.0378338
20	ola	1	6	0.9436923	46	0.0343028	97	0.0723341
04	silla	1	6	0.9562246	63	0.0604607	63	0.0604607
09	ventilador	1	6	0.9583219	60	0.0809717	60	0.0809717
02	pantalón	1	6	0.9576618	62	0.0482866	62	0.0482866
08	mesa	1	6	0.9505229	46	0.0353846	385	0.2961538
01	cabeza	1	6	0.9576033	37	0.0238556	410	0.2643456

```

levels %>%
  filter(centers=="01") %>%
  arrange(-availability) %>%
  #head(20) %>%
  filter(level > 0) %>%
  select(-order) %>%
  kable(longtable=TRUE,booktabs=TRUE) %>%
  kable_styling(latex_options = c("hold_position", "repeat_header")) # %>%

```

centers	words	availability	level	cutlevel	freq.abs	freq.rel	freq.abs.cum	freq.rel.cum
01	cabeza	1.0000000	6	0.9576033	37	0.0238556	410	0.2643456
01	brazo	0.9999999	6	0.9576033	50	0.0322373	108	0.0696325
01	ojo	0.9999996	6	0.9576033	58	0.0373952	58	0.0373952
01	mano	0.9999963	6	0.9576033	42	0.0270793	373	0.2404900
01	pierna	0.9999664	6	0.9576033	49	0.0315925	157	0.1012250
01	pie	0.9999228	6	0.9576033	43	0.0277240	331	0.2134107
01	corazón	0.9993747	6	0.9576033	44	0.0283688	201	0.1295938
01	dedo	0.9981017	6	0.9576033	44	0.0283688	245	0.1579626
01	nariz	0.9978564	6	0.9576033	43	0.0277240	288	0.1856867
01	boca	0.9948986	6	0.9576033	33	0.0212766	479	0.3088330
01	pelo	0.9923933	6	0.9576033	27	0.0174081	595	0.3836235
01	oreja	0.9863368	6	0.9576033	29	0.0186976	568	0.3662153
01	hueso	0.9759572	6	0.9576033	21	0.0135397	686	0.4422953
01	hígado	0.9701666	6	0.9576033	29	0.0186976	539	0.3475177
01	músculo	0.9674082	6	0.9576033	19	0.0122502	705	0.4545455
01	tronco	0.9664685	6	0.9576033	10	0.0064475	1040	0.6705351
01	uña	0.9645619	6	0.9576033	36	0.0232108	446	0.2875564
01	pulmón	0.9271497	5	0.9197114	31	0.0199871	510	0.3288201
01	riñón	0.9197114	5	0.9197114	23	0.0148291	643	0.4145712

(continued)

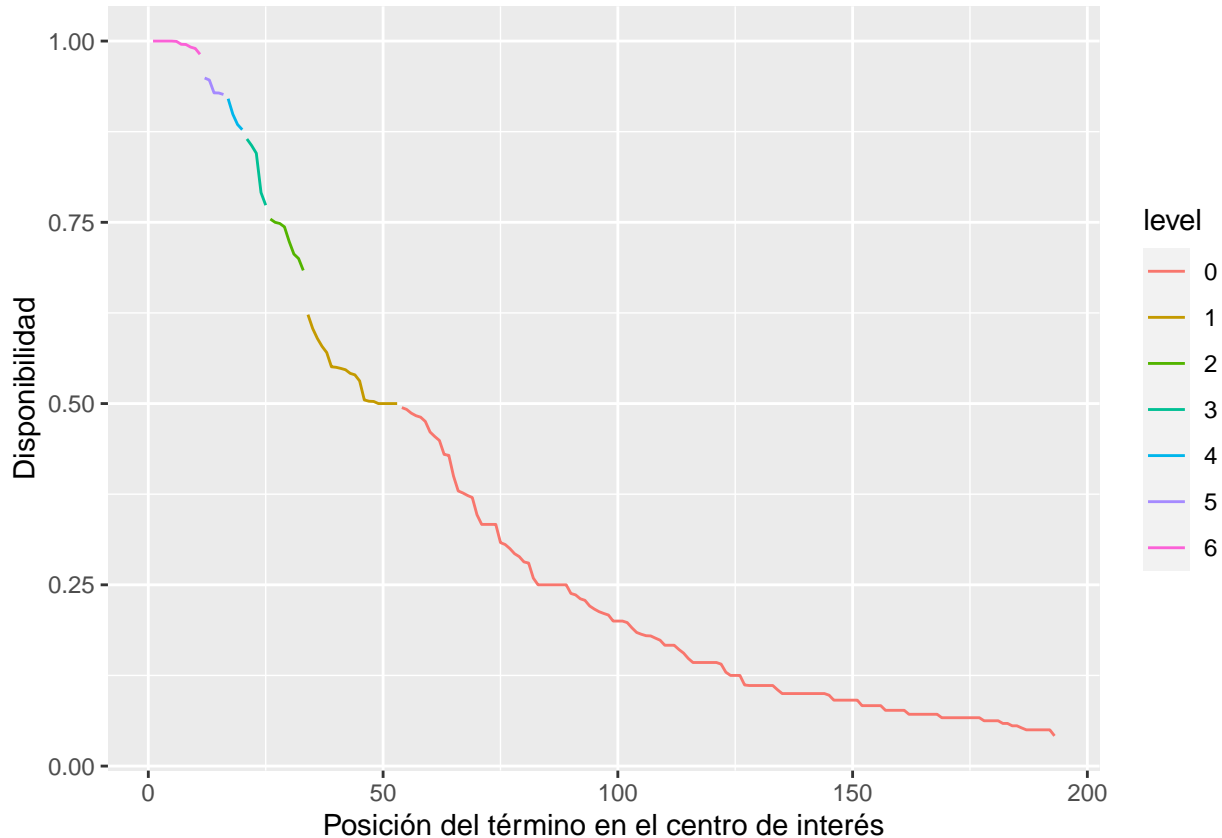
centers	words	availability	level	cutlevel	freq.abs	freq.rel	freq.abs.cum	freq.rel.cum
01	estómago	0.9119534	4	0.8569412	25	0.0161186	620	0.3997421
01	extremidad	0.8827436	4	0.8569412	8	0.0051580	1084	0.6989039
01	ceja	0.8670245	4	0.8569412	18	0.0116054	723	0.4661509
01	oído	0.8591329	4	0.8569412	13	0.0083817	851	0.5486783
01	órgano	0.8569412	4	0.8569412	7	0.0045132	1134	0.7311412
01	cara	0.8460897	3	0.7652840	10	0.0064475	1000	0.6447453
01	cerebro	0.8439199	3	0.7652840	12	0.0077369	876	0.5647969
01	codo	0.8122445	3	0.7652840	18	0.0116054	741	0.4777563
01	páncreas	0.7808868	3	0.7652840	18	0.0116054	777	0.5009671
01	rodilla	0.7777151	3	0.7652840	22	0.0141844	665	0.4287556
01	diente	0.7732121	3	0.7652840	18	0.0116054	759	0.4893617
01	sangre	0.7671473	3	0.7652840	16	0.0103159	809	0.5215990
01	hombro	0.7665272	3	0.7652840	14	0.0090264	838	0.5402966
01	cuello	0.7364473	2	0.6196081	16	0.0103159	793	0.5112830
01	esqueleto	0.7037037	2	0.6196081	4	0.0025790	1216	0.7840103
01	pestaña	0.6988165	2	0.6196081	13	0.0083817	864	0.5570600
01	cintura	0.6980489	2	0.6196081	9	0.0058027	1058	0.6821406
01	tibia	0.6939048	2	0.6196081	10	0.0064475	1030	0.6640877
01	antebrazo	0.6892100	2	0.6196081	11	0.0070922	935	0.6028369
01	lengua	0.6713687	2	0.6196081	12	0.0077369	912	0.5880077
01	tobillo	0.6612064	2	0.6196081	15	0.0096712	824	0.5312701
01	vello	0.6196081	2	0.6196081	6	0.0038685	1176	0.7582205
01	vena	0.6134745	1	0.4626262	11	0.0070922	990	0.6382979
01	enfermedad	0.6022542	1	0.4626262	5	0.0032237	1186	0.7646680
01	espalda	0.5956868	1	0.4626262	12	0.0077369	888	0.5725338
01	pecho	0.5909764	1	0.4626262	12	0.0077369	924	0.5957447
01	labio	0.5902379	1	0.4626262	10	0.0064475	1020	0.6576402
01	garganta	0.5690848	1	0.4626262	7	0.0045132	1120	0.7221148
01	peroné	0.5634803	1	0.4626262	9	0.0058027	1076	0.6937460
01	muñeca	0.5631837	1	0.4626262	11	0.0070922	968	0.6241135
01	bazo	0.5447121	1	0.4626262	11	0.0070922	946	0.6099291
01	esófago	0.5446623	1	0.4626262	11	0.0070922	957	0.6170213
01	culo	0.5344353	1	0.4626262	7	0.0045132	1113	0.7176015
01	fémur	0.5278453	1	0.4626262	12	0.0077369	900	0.5802708
01	torso	0.5112414	1	0.4626262	4	0.0025790	1260	0.8123791
01	salud	0.5040258	1	0.4626262	4	0.0025790	1248	0.8046422
01	barriga	0.5026316	1	0.4626262	4	0.0025790	1208	0.7788524
01	cadera	0.4839148	1	0.4626262	9	0.0058027	1049	0.6763378
01	falange	0.4690370	1	0.4626262	10	0.0064475	1010	0.6511928
01	piel	0.4626262	1	0.4626262	11	0.0070922	979	0.6312057

```
#column_spec(2, width = "10em")
```

Se puede construir con esta información una representación en la que se observan la distribución de las disponibilidades en el centro de interés y los diferentes conjuntos de cortes.

```
levels %>%  
  filter(centers=="04") %>%
```

```
mutate(level=factor(level)) %>%
  arrange(-availability) %>%
  ggplot(aes(x=order,y=availability,color=level)) + geom_line() +
  xlab("Posición del término en el centro de interés") +
  ylab("Disponibilidad")
```



```
clasificacion <- build.availability.levels(levels)
```

```
clasificacion %>% filter(levels > 0) %>%
  kable(longtable=TRUE,booktabs=TRUE) %>%
  kable_styling(latex_options = c("hold_position", "repeat_header")) %>%
  column_spec(4, width = "35em")
```

centers	levels	count	words
01	6	17	cabeza , brazo , ojo , mano , pierna , pie , corazón, dedo , nariz , boca , pelo , oreja , hueso , hígado , músculo, tronco , uña
01	5	2	pulmón, riñón
01	4	5	estómago , extremidad, ceja , oído , órgano
01	3	8	cara , cerebro , codo , páncreas, rodilla , diente , sangre , hombro
01	2	9	cuello , esqueleto, pestaña , cintura , tibia , antebrazo, lengua , tobillo , vello
01	1	18	vena , enfermedad, espalda , pecho , labio , garganta , peroné , muñeca , bazo , esófago , culo , fémur , torso , salud , barriga , cadera , falange , piel
02	6	16	pantalón , camisa , falda , camiseta , calcetín , jersey , vestido , zapato , abrigo , chaqueta , blusa , braga , calzoncillo, sujetador , bufanda , corbata
02	5	1	(pantalón) vaquero
02	4	5	moda , tejido , media , sombrero, bañador

(continued)

centers	levels	count	words
02	3	7	zapatilla, chándal , traje , camión , anorak , bermudas , tanga
02	2	10	cinturón , biquini , rebeca , gorro , color , pijama , bolso , guante , necesario, chaleco
02	1	18	estilo , vestir , chaquetón , pañuelo , felpa , rebajas , diseño , comodidad , gabardina , botón , lana , (pantalón) pirata, falda vaquera , vestimenta , bota , sudadera , slip , algodón
03	6	12	cocina , salón , (cuarto de) baño, dormitorio , comedor , terraza , salita , entrada , (cuarto de) aseo, pasillo , lavadero , ventana
03	5	4	hall , recibidor , (cuarto) trastero, habitación
03	4	6	tejado , jardín , techo , escalera , sala de estar, vestidor
03	3	5	patio , garaje, pared , porche, balcón
03	2	5	despensa , buhardilla, sótano , suelo , vestíbulo
03	1	5	chimenea, puerta , despacho, azotea , estudio
04	6	11	mesa , silla , cama , sofá , sillón , armario , mesita de noche, aparador , cómoda , estantería , lámpara
04	5	5	ropero , mesa de comedor, espejo , vitrina , encimera
04	4	4	cuadro , lavadora , librería , frigorífico
04	3	5	tresillo , mueble de cocina, mueble-bar , lavavajillas , zapatero
04	2	8	butaca , tele(visión) , mesilla de noche, escritorio , repisa , taquillón , tocador , cajonera
04	1	20	sinfonier , hornilla , lámpara de techo , mueble de salón , alacena , mesa camilla , cama de matrimonio , mesa de tele(visión), taburete , comodín , mecedora , televisor , mesa de salón , ordenador , cortina , armario de niños , barbacoa , cenefa , mueble de entrada , rinconero
05	6	21	pan , carne , leche , tomate , huevo , lenteja , arroz , patata , garbanzo , verdura , pescado , fruta , manzana , cebolla , lechuga , agua , plátano , pera , pimienta , mantequilla, naranja
05	5	2	aceite, pasta
05	4	7	queso , jamón , melón , sandía, azúcar, pollo , judía
05	3	7	yogur , melocotón , chocolate , legumbre , fresa , carne de cerdo, alcachofa
05	2	15	zanahoria , carne de ternera, ensalada , maíz , café , uva , coliflor , calabacín , harina , chorizo , espinaca , hamburguesa , espagueti , sopa , bocadillo
05	1	24	kiwi , refresco , salchichón , cerdo , macarrón , berenjena , galleta , carne de pollo, vino , zumo , dulce , gazpacho , hortaliza , sal , cereal , fideo , filete , limón , mandarina , pepino , col , calabaza , fibra , trigo
06	6	10	tenedor , mantel , cuchara , cuchillo , plato , servilleta , vaso , plato llano, plato hondo, ensaladera
06	5	5	copa , cucharilla , botella , cuchillo de carne , cuchillo de pescado
06	4	6	plato de postre , fuente , tenedor de pescado, jarra , cubierto , sopera
06	3	5	cucharón (de servir), salero , tenedor de carne , cuchara sopera , pan
06	2	8	salvamanteles, panera , bandeja , cuenco , taza , plato soperero , servilletero , jarra de agua
06	1	11	copa de vino , vela , pimentero , cuchara de postre , cucharilla de postre, bol , candelabro , centro floral , tenedor de ensalada , frutero , vaso de vino
07	6	20	sartén , horno , hornilla , micro(ondas) , olla , cacerola , frigorífico , lavadora , cuchillo , nevera , cucharón , espumadera , fregadero , olla exprés , (placa) vitro(cerámica), plato , lavavajillas , mesa , encimera , cuchara
07	5	3	cazo , mueble , batidora
07	4	4	silla , congelador, tenedor , cafetera

(continued)

centers	levels	count	words
07	3	3	vaso , paleta , armario
07	2	5	colador , freidora , tostadora , cacillo , cazuela (de barro)
07	1	24	placa , despensa , plato hondo , grifo , escurridor , bandeja , tortilla , cocina , cubierto , poyo , especiero , exprimidor , salero , barra americana , cocinar , cortina , cuchillo de servir , fluorescente , lavadero , mueble bajo , tabla para picar , paño , plancha , cubo de (la) basura
08	6	11	pizarra , mesa , pupitre , silla , lápiz , libro , tiza , bolí(grafo) , goma (de borrar), libreta , cuaderno
08	5	2	ordenador , sacapuntas
08	4	7	folio , mesa del profesor , borrador (de tiza), lápiz de color , aula , regla , pluma
08	3	4	rotulador, lapicero , mapa , carpeta
08	2	13	tarima , mochila , plastilina , banca , hoja (de papel) , puerta , cuadro , cartabón , escuadra , profesor , banco , lápiz de cera , libro de matemáticas
08	1	18	compás , papel , clase , estuche , armario , estantería , ventana , timbre , diccionario , proyector , tintero , calculadora , bloc , curso , libro de conocimiento del medio, librería , maleta , papelera
09	6	12	aire acondicionado , ventilador , lámpara , bombilla , estufa , radiador , (aparato) calefactor, ventana , chimenea , calefacción , caldera , vela
09	5	2	abrir ventanas, linterna
09	4	3	foco , brasero , abanico
09	3	4	flexo , fluorescente , (foco) halógeno , extractor (de humos)
09	2	5	electricidad , luz , abrir puertas, climatizador , carbón
09	1	12	puerta , tubo fluorescente , refrigeración , estufa eléctrica , lámpara de pie , leña , interruptor , calentador , aparato de calefacción , foco de jardín , bombilla de bajo consumo, cerilla
10	6	17	calle , coche , edificio , carretera , plaza , semáforo , parque , jardín , avenida , acera , farola , tienda , casa , (auto)bús , tráfico , ruido , moto(cicleta)
10	5	4	árbol, banco, museo, gente
10	4	3	callejón , bici(cleta), iglesia
10	3	12	polución , colegio , papelera , bar , fuente , bloque de pisos , ayuntamiento , paso de peatones, cine , contaminación , catedral , teatro
10	2	9	piso , asfalto , centro comercial, restaurante , policía , paso de cebra , puerto , humo , trabajo
10	1	24	flor , parada de (auto)bús, persona , peatón , atasco , comercio , barrio , chalé , población , camión , calzada , distrito , especulación , lugar para vivir , playa , rotonda , polideportivo , perro , paseo , hospital , pasaje , señal de tráfico , metro , campo de fútbol
11	6	8	árbol , flor , hierba , río , tierra , planta , animal , montaña
11	5	4	olivo , verde , vaca , gallina
11	4	9	pájaro , insecto, caballo, granja , monte , piedra , casa , cerdo , huerta
11	3	13	tractor , arado , paisaje , cabra , agua , tranquilidad , naranjo , bosque , agricultura , alegría , peral , aire puro , abeja
11	2	13	conejo , césped , camino , cultivo , sembrado , olivar , fruta , naturaleza , lago , mariposa , árbol frutal, toro , sol
11	1	35	pradera , aire limpio , oveja , vegetación , campo , terreno , manzano , avispa , hormiga , campesino , alergia , brisa , descanso , hierba seca , hierba verde, suelo , plantación , carril , limonero , paella , cortijo , roca , rama , arroyo , aire , arbusto , matorral , rosa , almendro , burro , margarita , amapola , vereda , pantano , invernadero

(continued)

centers	levels	count	words
12	6	12	coche , avión , (auto)bús , bici(cleta) , tren , moto(cicleta), barco , patín , metro , helicóptero , caballo , camión
12	5	4	avioneta, tranvía , carro , patinete
12	4	2	taxi , a pie
12	3	5	furgoneta , monopatín , globo (aerostático), coche de caballos , carreta
12	2	6	automóvil, burro , barca , triciclo , autocar , lancha
12	1	10	camioneta , andar , tractor , (tren de) cercanías, (tren) AVE , ciclomotor , gasoil , quad , moto acuática , sidecar
13	6	12	arar , sembrar , regar , podar , plantar , segar , recolectar , jardinero , abonar , labrar , cortar césped, cavar
13	5	5	recoger , agricultor, fumigar , siembra , talar
13	4	3	cortar , injertar , recolección
13	3	4	cultivar , recoger aceitunas, trasplantar , trillar
13	2	7	poda , cosechar , vendimiar, pastor , riego , arado , podador
13	1	20	varear , ganadero , hortelano , sembrar flores , limpiar , siega , recolector , roturar , arreglar flores , botánica , huerta , huerto , jardinería , recogida de algodón , segadora , recoger almendras , aventar , quitar malas hierbas, echar abono , regador
14	6	11	perro , gato , león , tigre , caballo , elefante, vaca , toro , pájaro , cerdo , conejo
14	5	6	ratón , oveja , jirafa , oso , canario , leopardo
14	4	6	burro , serpiente, gallina , rata , paloma , águila
14	3	10	mono , cebra , cabra , rinoceronte, abeja , pato , loro , ardilla , cucaracha , ballena
14	2	14	ave , avestruz , pez , gorrión , hormiga , buey , delfín , jabalí , cordero , periquito, búfalo , pollo , tortuga , camello
14	1	21	araña , lobo , ciervo , hipopótamo , pantera , guepardo , zoo(lógico), tiburón , hámster , cocodrilo , foca , pingüino , lagartija , lombriz , dogo , felino , liebre , pitón , murciélago , mosquito , gallo
15	6	14	parchís , fútbol , oca , ajedrez , cartas , dominó , baloncesto , cine , tenis , póquer , pilla-pilla , leer , tele(visión), escondite
15	5	4	pasear , mus , balonmano, comba
15	4	6	videojuego, Monopoly , damas , teatro , cinquillo , solitario
15	3	9	leer libros , juego de mesa, crucigrama , lotería , música , bailar , tute , internet , deporte
15	2	14	tres en raya , natación , puzzle , pelota , correr , dados , golf , volei(bol) , Play(-Station), naipes , Trivial , nadar , ping-pong , lectura
15	1	23	ordenador , playa , sudoku , muñeco , Tetris , canasta , escribir , rueda , baile , saltar , baraja , brisca , senderismo , gallinita ciega , guiso , balón , coleccionar sellos , hacer el tonto , hobby , parque acuático , PS2 , radio , billar
16	6	14	médico/a , profesor/ora, albañil , carpintero/a, maestro/a , abogado/a , fontanero , electricista, enfermero/a , jardinero , ingeniero , camarero/a , mecánico , bombero
16	5	3	dependiente/a, policía , pintor
16	4	6	arquitecto/a , cocinero/a , conductor/ora, panadero , juez/eza , peluquero/a
16	3	10	carnicero , zapatero , chapista , celador , ama de casa , escayolista , secretario/a , vendedor/ora , administrativo, informático
16	2	12	obrero , economista , administrador , limpiador/ora , escritor , farmacéutico/a , comercial , policía local , ingeniero de caminos, psicólogo , solador , agricultor

(continued)

centers	levels	count	words
16	1	27	torero , modisto[a] , pescadero , ebanista , dentista , taxista , tapicero , camionero , notario/a , representante , frigorista , delineante , óptico/a , cajero/a , fiscal , auxiliar de enfermería, cetrero , doctor , ingeniero de minas , maestro de música , peón , piloto , técnico , perito , sastre , maquinista , ingeniero agrónomo
17	6	10	euro , banco , moneda , bolsa , billete , peseta , dólar , hipoteca, libra , acción
17	5	2	cuenta corriente, céntimo
17	4	6	economista, comprar , cheque , ahorro , inversión , franco
17	3	10	préstamo , caja de ahorros , interés , marco , letra (de cambio), lira (italiana) , IVA , cambio , cajero , tarjeta
17	2	13	banquero , necesario, trabajo , monedero , dinero , cartilla , factura , íbex 35 , comercio , cartera , IPC , divisa , oferta
17	1	27	sueldo , hacienda , gastar , gasto , trueque , transacción , ruina , vender , impuesto , tarjeta de crédito , deuda , contable , importante , llegar a fin de mes, piso , pobreza , warrant , libreta , talón , demanda , banca , coche , yen , mercado , rublo , fianza , bancarrota
18	6	11	ratón , pantalla , teclado , ordenador , chatear , Messenger , torre , página web , correo (electrónico), impresora , disco duro
18	5	4	programa, monitor , chat , Google
18	4	8	red , PC , Web , información, navegar , software , Chat , Windows
18	3	9	altavoz , internet , web , mensaje , escáner , ADSL , virus , CD-ROM , placa base
18	2	15	computadora , bit , micro(fono) , juego , E-mail , conexión , router , Internet , webcam , hardware , archivo , Bill Gates , Software , buscar trabajo, E-Mule
18	1	26	grabadora , Microsoft , Pentium , amigo , cable , informática , automatismo , desesperación, Hispachat , interesante , Java , nueva era , punto com , procesador , tecla , CPU , invento , novio , Terra , comunicación , cámara web , ventilador , alfombrilla , foro , portátil , Basic
19	6	15	rojo , azul , blanco , amarillo , verde , negro , rosa , violeta , naranja , celeste , marrón , gris , morado , arco iris, añil
19	5	2	beis , fucsia
19	4	1	turquesa
19	3	4	azul marino, primario , cálido , magenta
19	2	6	burdeos , pastel , secundario, luz , lila , malva
19	1	11	ocre , alegría , cian , plata , azul celeste , cromatismo , tono , violeta claro, granate , dorado , verde botella
20	6	11	ola , agua , barco, pez , arena, sal , playa, alga , mar , roca , azul
20	5	4	orilla , tiburón, pesca , marea
20	4	4	espuma , oleaje , océano , pescador
20	3	14	ballena , barca , nadar , horizonte , velero , arrecife , navegar , paz , boquerón , coral , sol , calamar , tranquilidad, delfín
20	2	13	maremoto , concha , color , inmenso , mejillón , medusa , verano , almeja , caracola , gaviota , resaca , barco de vela, vida
20	1	34	acantilado , viento , sombrilla , rompeolas , red , moto de agua , fondo , lancha (motora) , brisa , puerto , sardina , bucear , piedra , pescado , bonito , mar tranquilo , Resolution , veranear , cabo , yodo , pulpo , pescar , marinero , paseo , tsunami , crucero , submarino , flotador , espigón , golfo , relajación , arrecife de coral, chiringuito , marejada

```

levels %>%
  mutate(level=factor(level)) %>%
  arrange(-availability) %>%
  ggplot(aes(x=order,y=availability,color=level)) + geom_line() + facet_wrap(~centers) +
  xlab("Secuencia de palabras (por grado descendente de compatibilidad)") +
  ylab("Disponibilidad")

```

