

WikiGenDex: Un nuevo algoritmo de identificación de género basado en fuentes abiertas

WikiGenDex: A new gender identification algorithm based on open sources

Elvira González-Salmón; Nicolás Robinson-García

Cómo citar este artículo:

González-Salmón, Elvira; Robinson-García, Nicolás (2024). "WikiGenDex: Un nuevo algoritmo de identificación de género basado en fuentes abiertas [WikiGenDex: A new gender identification algorithm based on open sources]". *Infonomy*, 2(1), e24010.

<https://doi.org/10.3145/infonomy.24.010>



Elvira González-Salmón

<https://orcid.org/0000-0003-3826-766X>

<https://www.directorioexit.info/ficha6855>

Universidad de Granada

Unit for Computational Humanities and Social Sciences (U-CHASS)

EC3 Research Group

Granada, España

elviragonzalez@go.ugr.es



Nicolás Robinson-García

<https://orcid.org/0000-0002-0585-7359>

<https://www.directorioexit.info/ficha1592>

Universidad de Granada

Unit for Computational Humanities and Social Sciences (U-CHASS)

EC3 Research Group

Granada, España

elrobinster@gmail.com

Resumen

Ante las limitaciones encontradas en otros algoritmos de asignación de género basados en nombres, presentamos un nuevo algoritmo basado en fuentes abiertas: *WikiGenDex*. Este algoritmo se basa en información de *Wikidata* y del *World Gender Name Dictionary* (pertenece a la *Organización Mundial de la Propiedad Intelectual*). A lo largo de la nota expli-

camos su mecanismo, validamos su funcionamiento, ofrecemos un ejemplo y un tutorial para usarlo con cualquier base de datos de nombres.

Palabras clave

Identificación de género; Nombres; Bibliometría; Género y ciencia; Algoritmos; Fuentes abiertas; Acceso abierto.

Abstract

Given the limitations found in other name-based gender assignment algorithms, we present a new algorithm based on open sources: *WikiGenDex*. This algorithm is based on information from *Wikidata* and the *World Gender Name Dictionary* (belonging to the *World Intellectual Property Organization*). Throughout the note we explain its mechanism, validate its operation, offer an example and a tutorial to use it with any name database.

Keywords

Gender identification; Names; Bibliometrics; Gender and science; Algorithms; Open sources; Open access.

Financiación

Este trabajo es parte del proyecto COMPARE - Contextual mapping of academic pathways analysis for research evaluation (REF: PID2020-117007RA-I00), financiado por el *Ministerio de Ciencia e Innovación* (REF: MCIN/AEI/10.13039/501100011033), y el *Fondo Social Europeo* "FSE invierte en tu futuro".

Elvira González-Salmón disfruta de un contrato FPU (REF: FPU2021/02320). Nicolás Robinson-García es investigador Ramón y Cajal (REF: RYC2019-027886-I), ambos financiados por el *Ministerio de Ciencia e Innovación*.

1. Introducción

En la última década hemos sido testigos de grandes avances en el desarrollo de algoritmos computacionales aplicados al ámbito bibliométrico. Estos avances han dado lugar a un crecimiento exponencial en el número de estudios cuantitativos sobre género y ciencia (**González-Salmón; Chinchilla-Rodríguez; Robinson-García, 2024**). Para ello, utilizan algoritmos de asignación de género basados en la premisa de que es posible inferir el género de una persona en base a la información bibliográfica que ofrecen las bases de datos científicas. Estos algoritmos estiman el género más probable de un/a autor/a bajo el supuesto de que éste viene reflejado en el nombre (a veces en combinación con otros campos como puede ser el país de afiliación o los apellidos). Por ejemplo, si Sara es un nombre que normalmente se asigna a mujer, se asume que lo más probable es que el nombre Sara en un listado de datos esté relacionado con ese género.

En esta breve nota metodológica, describimos algunas consideraciones a tener en cuenta en el uso de estos algoritmos y presentamos un nuevo algoritmo para la asignación masiva de género utilizando fuentes de información abiertas: *WikiGenDex*.

2. Supuestos previos y limitaciones en el uso de algoritmos de asignación de género

Hay diversos motivos que hacen necesario usar estos algoritmos de manera consciente y sabiendo sus limitaciones. En primer lugar, es importante tener en cuenta que estos algo-

ritmos no asignan género a un nombre, sino que estiman la probabilidad de que dicho nombre esté relacionado con ese género. No nos dicen el género exacto de una persona, introduciendo un grado de incertidumbre y limitando su uso para estudios a nivel micro.

En segundo lugar, se asume una relación entre nombre y género que no es necesariamente cierta, pues un nombre puede ser usado para designar distintos géneros. De este modo, estos algoritmos refuerzan estereotipos sobre qué género está relacionado con cada nombre. En tercer lugar, tienen una visión binaria del género: asumen que un nombre es de Mujer, Hombre, o *Unknown*. No hay espacio para otras identificaciones de género, en especial para las personas de género no-binario.

Estos algoritmos no asignan género a un nombre, sino que estiman la probabilidad de que dicho nombre esté relacionado con ese género. No nos dicen el género exacto de una persona, introduciendo un grado de incertidumbre y limitando su uso para estudios a nivel micro.

Una vez aceptadas estas limitaciones, detallamos brevemente cómo funcionan estos algoritmos. Normalmente basan sus estimaciones en listas de nombres previamente recabadas para las que sí cuentan con un género conocido. Ejemplos son las listas censales en países como Estados Unidos, listas hechas *ad hoc* a través de encuestas, o de los institutos nacionales de estadística, entre otras. Estas listas de nombres se cruzan con la lista de nombres cuyo género queremos inferir y en función de la probabilidad de veces que los nombres aparecen asignados a un género en la lista referencia, se asignará dicho género en el set de datos final. La fiabilidad del algoritmo vendrá determinada por la calidad y cobertura de la lista referencia. Si ésta proviene, por ejemplo, únicamente de un país determinado, es posible que no sea capaz de inferir el género correcto al cruzarse con conjuntos de datos provenientes de otros países.

El factor geográfico es esencial y será determinante en otros aspectos. De esta manera, el algoritmo puede tener en cuenta la geografía y al hacer *matching* considerar tanto columna de género como columna de país para asignar género. Así, el algoritmo indicará que el nombre Andrea vinculado a España normalmente se asigna a mujer, mientras que si pertenece a Italia lo más probable es que ese nombre sea asignado a hombres.

Finalmente, el factor geográfico determinará el grado de incertidumbre con el que se infieren los nombres. En países asiáticos, por ejemplo, es más común encontrar nombres asignados indistintamente a ambos géneros, mientras que en países eslavos, el apellido también puede ofrecer información útil a la hora de inferir el género de una persona. Todas estas consideraciones afectarán a la representatividad de nuestros datos al hacer análisis globales, ya que habrá nombres para los que no podremos inferir un género o lo haremos con mayor probabilidad de error, lo que puede comprometer nuestros análisis futuros.

3. WikiGenDex

La falta de transparencia de muchas de las soluciones que se ofrecen tanto de manera gratuita como de pago, añadido a las limitaciones que muchas veces muestran, hace que sea recomendable el desarrollo o implementación de metodologías abiertas y transparentes

que permitan, si no minimizar estos errores, al menos sí reportarlos de manera responsable (**González-Salmón; Robinson-García, 2024**). A continuación, presentamos una solución metodológica desarrollada con estas características y mostramos a modo de tutorial, cómo implementarla sobre otros conjuntos de datos. Llamamos a este algoritmo *WikiGenDex*.

WikiGenDex emplea dos fuentes de datos abiertas como listas de referencia. Estas son: *Wikidata* <https://www.wikidata.org/?uselang=es>

y

WGND-World Gender Name Dictionary

<https://dataverse.harvard.edu/dataverse/WGND>

En el caso de *Wikidata*, extraemos un listado de nombres y apellidos vinculados a países y género. Este listado proviene de todas las entradas a personas incluidas en la *Wikipedia*. Hay que tener en cuenta que *Wikipedia* está sesgada e incluye más registros de hombres que de mujeres (**Tripodi, 2023**). A través del *World Gender Name Dictionary*, pudimos considerar, no sólo el país de origen de una persona determinada, sino también su idioma, pudiendo enriquecer el número de variables a considerar. La figura 1 muestra una imagen de la pantalla de descarga de *Wikidata* y el código requerido para descargar todos los nombres asignados a un género y país en concreto.

La inclusión de la información geográfica tanto en el set de datos como en la lista de referencia aumenta la fiabilidad del algoritmo. Es decir, la situación más idónea es cuando además de nombres, contamos con el país al que pertenece cada nombre

El proceso de creación del algoritmo comenzó limpiando los datos que habíamos obtenido: separando nombres de segundos nombres, quitando nombres que no utilizan el alfabeto latino y poniendo los datos en minúsculas, así como quitando símbolos tales como “-” o similares. Una vez limpio el listado, es posible calcular el porcentaje de veces que cada nombre está asignado a un género por país. De este modo, asignamos un género a un nombre-país, cuando dicho género aparece asignado al menos el 70% de las veces en ese país a ese género¹. En el caso de los países eslavos², son procesados de manera diferente. En caso de que el apellido que acompaña al nombre acabe en ov, en, in, eb, yi, yj, ky, kii, kij y ob se asume que lo más probable es que ese nombre haya sido asignado a hombre, mientras que si acaba en ova, eva, ina, oba, aya, aia, ina o iha, a mujer. A estas dos listas, añadimos una tercera lista independiente de la variable país en la que incluimos nombres que aparecen vinculados a un mismo género en al menos 10 países, asumiendo que lo más probable es que ese nombre esté relacionado con ese género en los países en los que ese nombre no figura. De este modo aumentamos la exhaustividad y asignamos nombres vinculados de manera global a un género.

¹ La diferencia entre poner el umbral en 70%, 80% o 90% era mínima, debido a que generalmente hay muchos nombres que aparecen muchas veces y solo unos pocos que aparecen pocas veces.

² Como países eslavos consideramos Bielorrusia, Bosnia y Herzegovina, Bulgaria, Croacia, República Checa, Montenegro, Macedonia del Norte, Polonia, Rusia, Serbia, Eslovaquia, Eslovenia y Ucrania.

La lista de nombres e idiomas extraída del *WGND* fue facilitada por la *Organización Mundial de la Propiedad Intelectual (OMPI)*, que es la creadora del *WGND*. Esto se hizo así al no utilizar directamente su diccionario. La razón por la que no hacemos esto es porque observamos importantes limitaciones en su asignación de género, asignando género en algunos casos a iniciales que no contenían el nombre desarrollado. De esta manera, usando únicamente su diccionario de nombre-idioma-género, cuando en más del 50% de los idiomas que se habla en un país se identifica cierto nombre con cierto género, se le asigna ese género al nombre.

```

1 SELECT ?personLabel ?givenNameLabel ?countryLabel
2 WHERE {
3   ?person wdt:P31 wd:Q5 ;
4           wdt:P21 wd:Q6581072 ;
5           wdt:P27 ?country ;
6           wdt:P735 ?givenName.
7   ?person rdfs:label ?personLabel .
8   ?country rdfs:label ?countryLabel .
9   ?givenName rdfs:label ?givenNameLabel .
10  FILTER(LANG(?personLabel)='en')
11  FILTER(LANG(?countryLabel)='en')
12  FILTER(LANG(?givenNameLabel)='en')
13  FILTER(?country = wd:Q32)
14 }
15

```

Figura 1. Ejemplo de Query en *Wikidata*, donde descargamos todos los nombres asignados a mujeres (wd:Q6581072) en un país determinado, en este caso Luxemburgo (Q32).

A modo de resumen, *WikiGenDex* es un algoritmo que sigue el siguiente *modus operandi*:

1. Identificación de género de nombres pertenecientes a países no eslavos, mediante nombre.
2. Identificación de género de nombres pertenecientes a países eslavos, teniendo en cuenta apellido y nombre.
3. Identificación de género de los nombres no identificados en los pasos previos que en la mayoría de los casos son asignados a un mismo género independientemente del país de origen.
4. Identificación de género de los nombres no identificados con los pasos previos en base a los idiomas empleados en el país de origen.

En la Tabla 1 mostramos el porcentaje de nombres a los que diferentes algoritmos asignan género basado en una muestra aleatoria de 10000 nombres de investigadores/as de todo el mundo extraídos de la base de datos *Dimensions*. *WikiGenDex* cubre alrededor del 80% de los nombres analizados, siendo *ChatGPT 3.5* el que mayor cobertura muestra, sin embargo, la opacidad del software unido a fallas importantes encontradas (e.g., asignación de género a iniciales) hace que *WikiGenDex* sea más fiable.

Tabla 1. Porcentaje de nombres asignados con éxito a un género para 10000 nombres aleatorios extraídos de *Dimensions* según *WGND*, *ChatGPT 3.5*, *Gender-Guesser* y *WikiGenDex*

<i>WGND</i>	<i>ChatGPT 3.5</i>	<i>Gender-Guesser</i>	<i>WikiGenDex</i>
87,20%	99,80%	50,70%	80,45%

WikiGenDex ha sido aplicado con éxito a todos los nombres de investigadores (esto es, autores/as con un identificador de investigador/a asignado) incluidos en la base de datos *Dimensions* para el periodo 1990-2021. A modo de ejemplo, las figuras 2 y 3 muestran la cobertura global de *WikiGenDex* por país así como la distribución de asignaciones por género.

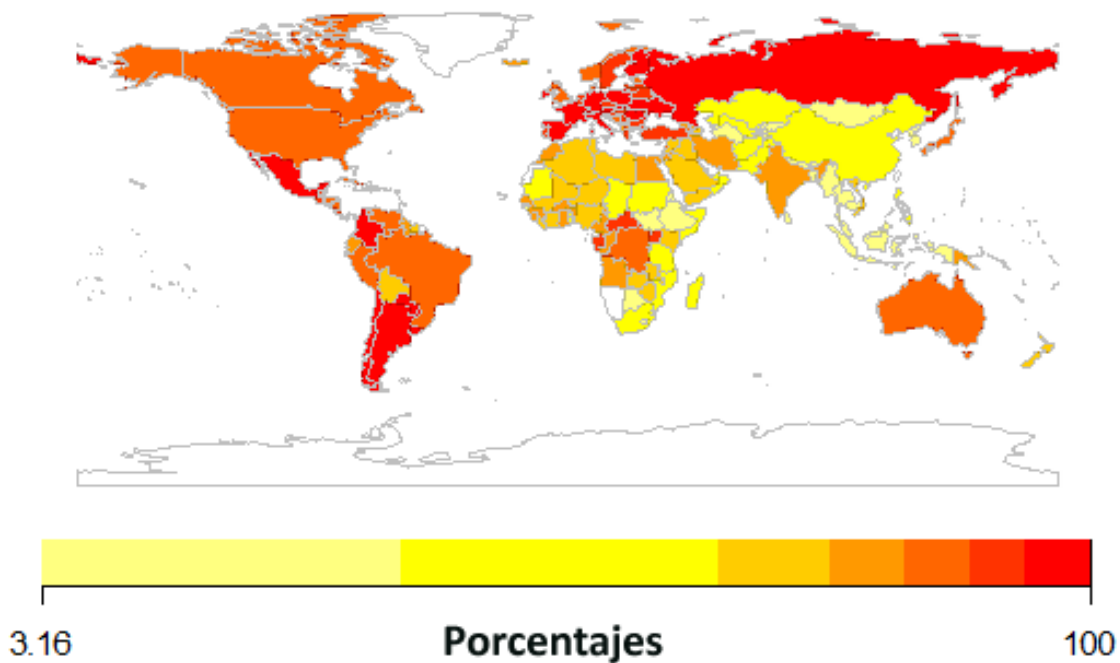


Figura 2. Porcentaje de investigadores/as con al menos 5 publicaciones identificadas en *Dimensions* en el periodo 1990-2021 con género asignado, por países.

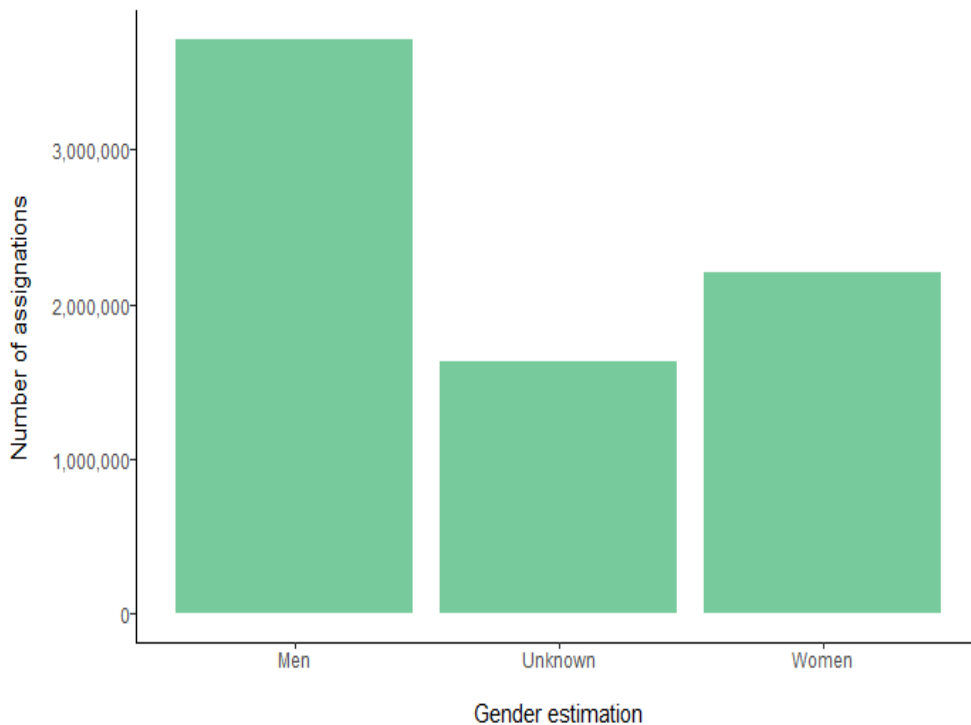


Figura 3. Número de asignaciones por género de investigadores/as con al menos 5 publicaciones identificadas en Dimensions en el periodo 1990-2021.

3. Tutorial

A continuación exponemos un tutorial de uso del algoritmo. Los *scripts* de código que se detallan están todos desarrollados en *R* y están disponibles en:

<https://github.com/egonzalezsalmon/WikidataGender>

Paso 1. Preparación del entorno y carga de datos

En un primer lugar, debemos instalar y cargar los siguientes paquetes de *R*:

```
# Instalación de paquetes
install.packages("stringi")
install.packages("dplyr")
install.packages("tidyverse")
install.packages("tidyr")
install.packages("rlang")
# Carga de paquetes en R
library(stringi)
library(dplyr)
library(tidyverse)
library(tidyr)
library(rlang)
```

Después, añadimos el set de datos al que queremos asignar género. Esta tabla deberá contar con al menos las siguientes columnas:

- Una columna donde se indique el nombre (*first_name*)
- Una columna donde se indique el apellido (*last_name*)

- Una columna donde se indique el país al que pertenece cada nombre (*country_code*). Preferiblemente en formato Alpha-2
- Una columna con un identificador único para cada fila (*unique_id*)

La mayor parte de las veces contaremos con esos datos ya. Es posible hacer funcionar el algoritmo sin la columna *country_code*, pero la precisión de este bajará considerablemente.

```
data <- read.csv("your_list.csv")
```

Ahora conviene hacer un breve ejercicio de limpieza, en el que eliminemos las filas donde *country_code* sea nulo (podemos dejar estas filas, pero la precisión del resultado será menor), eliminamos los "-" de los nombres, romanizamos los nombres, separamos los nombres de los segundos nombres y ponemos todo en minúsculas.

```
data <- subset(data, country_code != "NULL")
data$first_name <- gsub("-", "", data$first_name)
data <- data %>%
  mutate(first_name = stri_trans_general(first_name, "Latin-ASCII"))
data <- data %>%
  separate(first_name, into=c("first_name", "middle_name"), sep = " ", extra = "merge", fill =
"right")
data$first_name <- tolower(data$first_name)
data$last_name <- tolower(data$last_name)
```

Paso 2. Asignación de género a los nombres eslavos

Una vez hecho esto, asignamos el género a los nombres que pertenezcan a países eslavos. Estos países se hacen de manera separada al resto ya que podemos encontrar información de género tanto en los nombres como en los apellidos, cosa que no ocurre en el resto de países.

Empezamos abriendo las tablas donde tenemos los nombres con género asignado:

```
WDNAMESslav <- read.csv("nombreseslavos.csv")
WDNAMESrest <- read.csv("restodenombres.csv")
```

Después, seleccionamos las filas de nuestros datos que pertenecen a países eslavos. En este paso asumimos que los países (*country_code*) están escritos en Código Alpha-2. En caso de estar escrito en otro código, se puede bien cambiar los datos a Alpha-2, o bien cambiar el código que escribimos a continuación. Por ejemplo, si los datos de país están en Alpha-3, se puede cambiar de "BY" a "BLR", etc.

```
dataslav <- subset(
  data, country_code
  %in%
  c("BY", "BA", "BG", "HR", "CZ", "ME", "MK", "PL", "RU", "RS", "SK", "SI", "UA"))
```

A continuación juntamos nuestros datos de países eslavos con *WDNAMESslav*, donde tenemos información de género de países eslavos. En los casos en los que el algoritmo identifi-

que el mismo género al estudiar nombres y apellidos, asignará dicho género. En caso de que no, asignará la etiqueta *Unknown*.

```

dataslav <- left_join(x = dataslav, y = WDNAMESslav, by = c("first_name", "country_code"),
suffix = c("", "_wdnames"))
get_gender <- function(last_name) {
  if (grepl("(ov|en|in|eb|yi|yj|ky|kii|kij|ob)$", last_name, ignore.case = TRUE)) {
    return("Man")
  } else if (grepl("(ova|eva|ina|oba|aya|aia|ina|iha)$", last_name, ignore.case = TRUE)) {
    return("Woman")
  } else {
    return("Unknown")
  }
}
dataslav$gender_last_name <- sapply(dataslav$last_name, get_gender)
frequency_table <- table(dataslav$Gender, dataslav$gender_last_name)
dataslav$Combo <- paste(dataslav$Gender, dataslav$gender_last_name, sep = "_")
frequency_table2 <- table(dataslav$Combo)
dataslav <- dataslav %>%
  mutate(GenderComb = case_when(
    Combo == "Man_Man" ~ "Man",
    Combo == "Woman_Woman" ~ "Woman",
    Combo == "Woman_Man" ~ "Unknown",
    Combo == "Man_Woman" ~ "Unknown",
    Combo == "Woman_Unknown" ~ "Woman",
    Combo == "Man_Unknown" ~ "Man",
    Combo == "NA_Man" ~ "Man",
    Combo == "NA_Woman" ~ "Woman",
    Combo == "NA_Unknown" ~ "Unknown",
    TRUE ~ NA_character_
  ))

```

Seguidamente, conviene eliminar columnas extra y renombrar columnas con los nombres de columnas originales, para que después puedan juntarse con la asignación de género que hagamos a nombres de países no eslavos (para poder juntar ambas listas deben tener el mismo número de columnas y deben llamarse igual).

Paso 3. Asignación de género a los nombres no eslavos

A continuación seleccionamos los nombres que no pertenecen a países eslavos y juntamos estos nombres con WDNAMESrest, para asignar género.

```

excluded_codes <- c("BY", "BA", "BG", "HR", "CZ", "ME", "MK", "PL", "RU", "RS", "SK", "SI",
"UA")
datarest <- subset(data, !country_code %in% excluded_codes)
datarest <- left_join(x = datarest, y = WDNAMESrest, by = c("first_name", "country_code"),
multiple = "all")

```

Paso 4. Unificación y limpieza de datos

El siguiente paso consiste en juntar las dos listas que hemos creado: la que tenemos con el género asignado de países eslavos y la que contiene el género de los no eslavos.

```
data_gender <- rbind(datarest, dataslav)
```

A continuación eliminamos los registros cuyo nombre es únicamente una inicial. Ahora que hemos comprobado los nombres eslavos, no será posible asignar un género a los iniciales cuyo género ha permanecido desconocido hasta ahora.

```
initials_rows <- grepl("^\\p{L}+\\.?$", data_gender$first_name, perl = TRUE)
data_gender <- data_gender[!initials_rows, ]
data_gender$first_name <- gsub("[A-Za-z]\\..*$", "", data_gender$first_name)
data_gender$first_name <- trimws(data_gender$first_name, "right")
```

También eliminamos las filas donde el nombre (*first_name*) sea un valor perdido (NA).

```
# Eliminate first_name = NA
missing_values <- c("", "NA", "NULL")
suppressWarnings({
  data_gender$first_name[data_gender$first_name %in% missing_values] <- NA
})
suppressWarnings({
  data_gender <- data_gender[complete.cases(data_gender$first_name), ]
})
```

Como recomendación, es buen momento para guardar todos los avances en un CSV.

```
write.csv(data_gender, "data_gender.csv", row.names=FALSE)
```

Paso 5. Creación de una segunda lista de Wikidata para nombres no asignados

Ahora tratamos los nombres cuyo género asignado ha sido *Unknown*. Para ello usamos *WDNAMES2*. Esta lista de nombres ha sido creada siguiendo la lógica de que si a un nombre se le suele asignar el mismo género en 10 o más países, lo más probable es que ese nombre esté relacionado con ese género en países donde no hay registro de ese nombre. Por lo tanto, no utiliza *country_code*. Es importante tener en cuenta que este paso reduce la precisión del algoritmo, al no tener en cuenta la variable país. Si estamos satisfechos con la identificación de género obtenida hasta aquí, se puede saltar este paso.

Para usarlo, cargamos la lista en R, seleccionamos los nombres de nuestros datos que han permanecido *Unknown* hasta ahora y juntamos ambas listas.

```
WDNAMES2 <- read.csv("WDNAMES2.csv")
data_unknown <- subset(data_gender, !(Gender %in% c("Man", "Woman")))
data_unknown_gender <- left_join(x = data_unknown, y = WDNAMES2, by = c("first_name"),
  multiple = "all")
```

A continuación eliminamos columnas innecesarias de manera que volvamos a tener las mismas columnas que en nuestros datos originales, para unir la nueva lista de género con la anterior.

```
data_known_gender <- subset(data_gender, Gender!= "Unknown")
data_gender <- rbind(data_known_gender, data_unknown_gender)
```

Paso 6. Creación de un diccionario basado en *WGND* para aumentar la exhaustividad del algoritmo

Aún queda un último paso para asignar género a los nombres que siguen siendo *Unknown*. Para ello, usamos el segundo diccionario de *WGND*, que en lugar de utilizar información nombre-país-género, utiliza nombre-idioma-género. La lógica de esta lista es que cuando un nombre se asigna al mismo género en más del 50% de las lenguas habladas en ese país, se le asignará ese género.

Empezamos cargando los datos de *WGND* y seleccionando los nombres de nuestros datos que siguen teniendo como género *Unknown*. A continuación, juntamos ambas listas.

```
wgnd_2_0_code_langcode <- read.csv("wgnd_2_0_code_langcode.csv")
wgnd_2_0_name_gender_langcode <- read.csv("wgnd_2_0_name_gender_langcode.csv")
data_language <- data_gender %>%
  filter(!Gender %in% c("Man", "Woman"))
suppressWarnings({
  data_language <- left_join(data_language, wgnd_2_0_code_langcode, by =
"country_code", keep = TRUE)
})
suppressWarnings({
  data_language <- left_join(data_language, wgnd_2_0_name_gender_langcode, by =
c("first_name", "langcode"))
})
data_language$gender[is.na(data_language$gender)] <- "Unknown"
result <- table(data_language$unique_id, data_language$gender)
result_df <- as.data.frame.matrix(result)
result_df$unique_id <- row.names(result_df)
result_df$GenderLan = apply(result_df[, -ncol(result_df)], 1, function(row) {
  total = sum(row)
  if (row["Man"] > total * 0.5) {
    return("Man")
  } else if (row["Woman"] > total * 0.5) {
    return("Woman")
  } else if (row["Unknown"] > total * 0.5) {
    return("Unknown")
  } else {
    return(NA)
  }
})
data_language_result <- result_df[, -c(1:3)]
```

```
data_language$unique_id <- as.character(data_language$unique_id)
data_language <- left_join(data_language_result, data_language, by = "unique_id", multi-
  ple="any")
data_gender_known <- data_gender %>%
  filter(Gender %in% c("Man", "Woman"))
```

Eliminamos las columnas innecesarias (esto dependerá de sus datos) y los unimos con los datos anteriores, cuyo género ya se había identificado antes.

```
data_gender_final <- rbind(data_gender_known, data_language)
data_gender_final <- data_gender_final %>%
  mutate(Gender = if_else(is.na(Gender), "Unknown", Gender))
```

En el fichero *README.md* del repositorio de *GitHub* indicado previamente se encuentra un resumen de lo que nos encontramos en el repositorio. También encontramos los archivos *WDNAMESslav.csv*, *WDNAMESrest.csv* y *WDNAMES2.csv*. Por último, añadimos el enlace a la ubicación original de los archivos de *World Gender Name Dictionary*. También incluimos un séptimo y octavo paso opcionales en los que se indica cómo realizar un gráfico para mostrar los resultados y cómo crear un mapamundi donde observar la distribución global de los resultados.

4. Bibliografía

González-Salmón, Elvira; Robinson-García, Nicolás (2023). "A call for transparency in gender assignment approaches". In review in *Scientometrics*.

<https://doi.org/10.5281/zenodo.10036331>

González-Salmón, Elvira; Chinchilla-Rodríguez, Zaida; Robinson-García, Nicolás (2024). "The woman's researcher tale: A review of bibliometric methods and results for studying gender in science". *U-CHASS White papers #1*.

<https://doi.org/10.5281/zenodo.10590300>

Tripodi, Francesca (2023). "Ms. categorized: Gender, notability, and inequality on Wikipedia". *New media & society*, v. 25, n. 7, pp. 1687-1707.

<https://doi.org/10.1177/14614448211023772>