Hire Ed: Job Market Dynamics for Tenure-Track Faculty Positions in Archaeology

Ben Marwick1,✉, Anne Marie Poole1, Ailin Zhang1, Setareh Shafizadeh1, and Jess Beck2

September 17, 2024

Abstract

Academic careers are frequently sought by archaeology graduate students. Job listing websites often serve as the first place for these students when seeking academic positions. We examined tenure-track job advertisements over the past decade to gain insights into the academic job market for archaeologists. Using data from the community-edited Academic Jobs Wiki for Archaeology, we examine changes in the academic job market over time. We studied the text of 449 job ads posted from 2013-2023. Our analysis focuses on shifts in archaeological topics and methods requested in job ads. We investigate whether the burden on applicants has changed over time: do institutions request more information and documents from applicants at the initial stages of application, compared to a decade ago? We also examine whether there is an increasing trend in job advertisements highlighting diversity and inclusivity, thereby encouraging a broader range of applicants. Additionally, we assess the influence of socio-political factors on the changing focus of research topics in the field. This research aims to assist current and future archaeology students and graduates in better understanding the job market and the requirements of employers, thereby aiding them in effectively preparing for their applications for positions in archaeology.

1 Department of Anthropology, University of Washington, Seattle, USA  
2 School of Archaeology, University College Dublin, Ireland

✉ Correspondence: [Ben Marwick <bmarwick@uw.edu>](mailto:bmarwick@uw.edu)

Keywords: keyword 1; keyword 2; keyword 3

Highlights: These are the highlights.

# Introduction

# Background

# Methods

Our primary data source is the Archaeology Academic Jobs Wiki. Originating in 2007, this is a set of freely accessible web pages that anyone can edit (anonymously or with a free user account) hosted by Fandom, a for-profit company. The Archaeology pages are part of the Academic Jobs Wiki, which coordinates similar collaboratively-edited resources for around 40 academic disciplines. The coordinators and contributors are nearly all anonymous or pseudonymous. Typically contributors copy and paste the text of job ads from other sources, such as the *Chronicle of Higher Education*, *Higher Ed Jobs*, and university websites, into the wiki, collecting ads originally posted in numerous different locations. Other contributors then edit the web page to add comments below an ad to share relevant information based on their experience in applying for that position, such as a tally of how many people have applied, the dates of events such as requests for more materials, interviews, offer made, rejection notices, etc. Contributors also edit the page to ask and answer questions about the positions and the application process. These comments make the Academic Jobs Wiki a unique resource for timely and specific information for job-seekers about positions they are interested in, and one of the most important internet resources for the academic job market. Because of its reputation for aggregating ads from diverse sources and rapidly-updated information that is not available elsewhere, the Academic Jobs Wiki has a large community of users that keep it updated and accurate has become an authoritative data source for studies of hiring trends in academia (e.g. Musial and Holmes, 2018) and a widely recommended resource (e.g. Lightfoot et al., 2021).

For each tenure-track job advertised on the Archaeology Academic Jobs Wiki during 2013-2023 we read the text and recorded into a Google form the name of the hiring institution, the title of the position, and exact words and phrases from the ad about the topical, geographic, and methods foci on the position. The topical focus is what we understood as the primary intellectual focus of the position. The geographic focus is the region of the world that the ideal candidate has scholarly expertise on. The methods focus is the data-generating sub-field of archaeology that is mentioned in the ad. We recorded the type and number of documents requested in each ad (e.g. cover letter, CV, statements on research, teaching, diversity, syllabi, course descriptions, writing samples, transcripts) and how many names/letters of recommenders were requested in the ad.

After completing primary data collection, we studied the topical, geographic, and methods foci of each ad and collaboratively and manually reduced the variation in the raw data into 10-15 categories to simplify analysis and visualisation. Our topic categories were: American archaeology, Ancient Europe and Mediterranean, Archaeological science, Archaeological theory, Biological anthropology, Complex societies, Digital archaeology, Environmental archaeology, Evolutionary anthropology, Indigenous and Historical archaeology, North Mesoamerican Archaeology, Pleistocene archaeology, and Public archaeology Our geographic categories were: Africa, Americas, Asia & India, Canada & Arctic, Europe, Mediterranean, Meso- & South America, Near East, Oceania, Midwest US, Northeastern US, Southeast US, Southwest US, and Western US. Our methods categories were: Archaeobotany, Archaeometry, Bioarchaeology, Ceramic analysis, Computational and Digital archaeology, Geoarchaeology, Landscape analysis, Lithic analysis, Material culture analysis, and Zooarchaeology. Ads could have multiple or none of these three foci, and some of the foci overlap. Some topics include geographic regions because this is how they are typically understood by archaeologists. For example Mesoamerican archaeology is understood to refer to a specific time period and geographic region. While these overlaps can make the data challenging to interpret, in our view it reflects the complex realities of how search committees express their needs in searching for new faculty, and is insightful in how it reveals intersections between different foci.

# Results

* basic trends
* where are the hiring universities?
* what Carnegie classifications of the hiring universities?

## Geographic trends over time

* what regions of the world are mentioned in the ads?

## Topic trends over time

## Method trends over time

## Instructions to applicants over time

library(tidyverse)  
library(here)  
library(ggbeeswarm)  
# This CSV file was downloaded from our data sheet here  
# https://docs.google.com/spreadsheets/d/1Jwe3UqJyedrV-QWlwR\_44\_\_t4xBVrCfxGyhXdi3E0sg/edit?resourcekey#gid=1686084773  
# note that you may need to download it again to get the latest updates!  
  
jobdata <- read\_csv(here::here('analysis/data/raw\_data/Tenure Track Job Advertisements in Archaeology (Responses) - Form Responses 1.csv')) %>%   
 # simplify the column names   
 janitor::clean\_names()  
  
total\_number\_of\_ads\_in\_our\_sample <- nrow(jobdata) # 550

We have 547 job advertisements in our sample

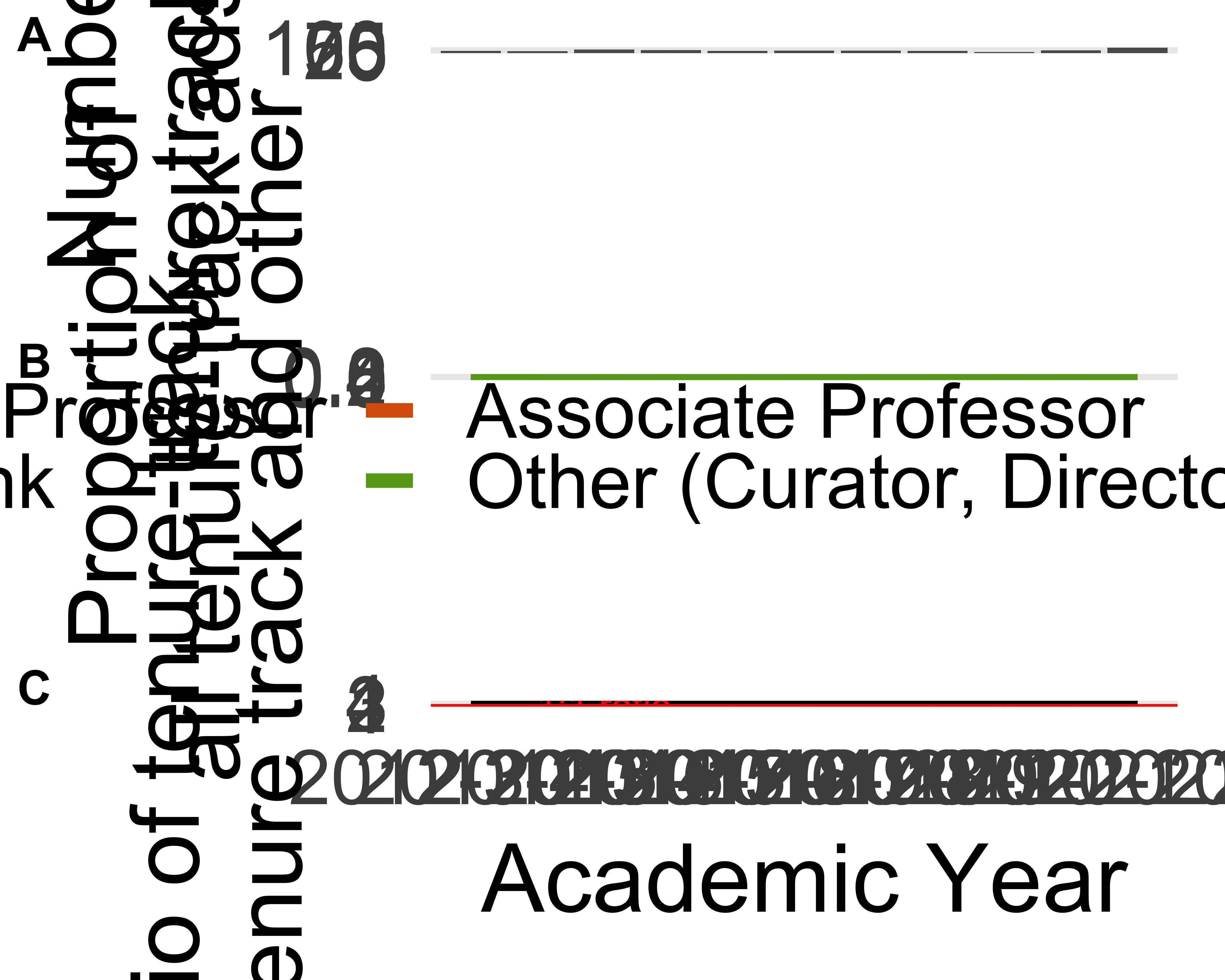
# we can get the year from the URL to the Academic Job Ads Wiki  
  
year\_ad\_posted <-   
jobdata %>%   
 pull(url\_to\_data\_source\_e\_g\_paste\_in\_url\_to\_the\_jobs\_wiki\_page) %>%   
 str\_extract(., "[[0-9]]{4}-[[0-9]]{4}|2021-22") %>%   
 str\_replace("2021-22", "2021-2022")  
 # fix for 2021-22 DONE!  
 # fix for 2023 DONE!  
  
jobdata <-   
 jobdata %>%   
 mutate(year\_ad\_posted = year\_ad\_posted) %>%   
 drop\_na(year\_ad\_posted)  
  
fig\_how\_many\_jobs\_per\_year <-   
ggplot(jobdata) +  
 aes(year\_ad\_posted) +  
 geom\_bar() +  
 scale\_x\_discrete(name = "") +  
 ylab("Number of\ntenure track job ads") +  
 theme\_minimal(base\_size = 28) +  
 guides(x = "none")

**?@fig-how-many-jobs-per-year** shows how many jobs per year in our sample

# how many jobs of each rank per year?  
  
jobdata <-  
jobdata %>%  
 # simplify rank descriptions  
 mutate(title\_of\_position\_tenure\_track\_jobs\_only = tolower(title\_of\_position\_tenure\_track\_jobs\_only)) %>%  
 mutate(job\_title\_simple = case\_when(  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,  
 "assistant prof|asst. prof|asst prof") ~ "Assistant Professor",  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,  
 "associate prof|assoc. prof") ~ "Associate Professor",  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,  
 "full prof") ~ "Full Professor",  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,  
 "assistant or associate prof|assistant/associate prof") ~ "Assistant or Associate Professor",  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,  
 "open rank|open-rank|assistant, associate, or full prof|assistant prof, associate prof, or prof") ~ "Open Rank",  
 .default = "Other (Curator, Director, etc.)"))  
  
# explore over time  
fig\_prop\_by\_job\_title\_per\_year <-   
jobdata %>%  
 group\_by(year\_ad\_posted) %>%  
 count(job\_title\_simple) %>%  
 mutate(prop = n / sum(n)) %>%  
 ggplot() +  
 aes(year\_ad\_posted,  
 prop,  
 group = job\_title\_simple,  
 colour = job\_title\_simple) +  
 geom\_line(linewidth = 2) +  
 theme\_minimal(base\_size = 28) +  
 xlab("") +  
 ylab("Proportion of\nall tenure-track ads") +  
 theme(legend.position = c(0.5, 0.5)) +  
 scale\_colour\_brewer(palette = "Dark2") +  
 guides(colour = guide\_legend(nrow=2,  
 byrow=TRUE,  
 "Job title")) +  
 guides(x = "none")

# ratio of tenure-track to untenured positions  
# base URL changes after 2018\_2019  
  
base\_url\_to\_2019 <- "https://academicjobs.fandom.com/wiki/Archaeology\_Jobs\_"  
base\_url\_after\_2020 <- "https://academicjobs.fandom.com/wiki/Archaeology\_"  
  
# starts at 2010-2011  
# fix for 2021-22  
# base UR  
  
years\_to\_2019 <- map\_chr(2012:2019, ~str\_glue('{.x}-{.x +1}'))  
years\_after\_2020 <- map\_chr(2020:2022, ~str\_glue('{.x}-{.x +1}'))  
# though it seems to start at 2007-8: https://academicjobs.fandom.com/wiki/Archaeology\_07-08  
  
# make a set of URLs for each page for each year  
urls\_for\_each\_year <- c(str\_glue('{base\_url\_to\_2019}{years\_to\_2019}'),   
 str\_glue('{base\_url\_after\_2020}{years\_after\_2020}')) %>%   
 str\_replace("2021-2022", "2021-22")  
  
library(rvest)  
  
# all years  
urls\_for\_each\_year\_headers <-   
map(urls\_for\_each\_year,  
 ~.x %>%   
 read\_html() %>%   
 html\_nodes('.mw-headline') %>%   
 html\_text())  
  
# keep only headings that are actual jobs, they include the terms:  
job\_headings <- c("college", "university")  
  
total\_number\_of\_jobs\_per\_year <-   
 map(urls\_for\_each\_year\_headers,  
 ~str\_subset(tolower(.x),  
 paste0(job\_headings, collapse = "|")))  
  
total\_number\_of\_jobs\_per\_year\_n <-   
map\_int(total\_number\_of\_jobs\_per\_year, length)  
  
total\_number\_of\_jobs\_per\_year\_tbl <-   
tibble(  
 url\_to\_data\_source\_e\_g\_paste\_in\_url\_to\_the\_jobs\_wiki\_page = urls\_for\_each\_year,  
 total\_number\_of\_jobs\_per\_year = total\_number\_of\_jobs\_per\_year\_n  
)  
  
# count of TT jobs per year from our manual data collection,  
# join with our total number of all jobs by scraping  
count\_of\_tt\_jobs\_per\_year\_from\_our\_form <-   
jobdata %>%   
 group\_by(url\_to\_data\_source\_e\_g\_paste\_in\_url\_to\_the\_jobs\_wiki\_page) %>%   
 tally() %>%   
 right\_join(total\_number\_of\_jobs\_per\_year\_tbl) %>%   
 rename(n\_tt\_jobs = n,  
 n\_total\_jobs = total\_number\_of\_jobs\_per\_year) %>%   
 mutate(n\_non\_tt\_jobs = n\_total\_jobs - n\_tt\_jobs,  
 ratio\_tt\_2\_ntt = n\_tt\_jobs / n\_non\_tt\_jobs) %>%   
 mutate(year = str\_extract(url\_to\_data\_source\_e\_g\_paste\_in\_url\_to\_the\_jobs\_wiki\_page, "[[0-9]]{4}-[[0-9]]{4}|2021-22")) %>%   
 mutate(year = ifelse(year =="2021-22", "2021-2022", year))   
  
# draw plot  
fig\_ratio\_tt\_2\_ntt\_jobs\_per\_year <-   
 ggplot(count\_of\_tt\_jobs\_per\_year\_from\_our\_form) +  
 aes(year,   
 group = 1,  
 ratio\_tt\_2\_ntt) +  
 geom\_line(linewidth = 2) +  
 geom\_hline(yintercept = 1,  
 colour = "red") +  
 annotate("text",   
 x = 3,   
 y = 1.3,   
 label = "1:1 ratio",  
 colour = "red") +  
 labs(y = "Ratio of tenure-track\nto non-tenure track and other",  
 x = "") +  
 theme\_minimal(base\_size = 28) +  
 scale\_x\_discrete(name = "Academic Year")   
 #guides(x = "none")

# save these three plots as one set  
library(cowplot)  
plot\_grid(  
 fig\_how\_many\_jobs\_per\_year,  
 fig\_prop\_by\_job\_title\_per\_year,  
 fig\_ratio\_tt\_2\_ntt\_jobs\_per\_year,  
  
 ncol = 1,  
 align = "hv",  
 axis = "lr",  
 labels = "AUTO"  
)



ggsave(here("analysis",  
 "figures",   
 "fig-panel-per-year.png"),  
 bg ="white",  
 h = 15, # experiment with h and w to get the right size and proportion   
 w = 20,  
 units = "in",  
 dpi = 900) # make the image nice and crisp

# look at only these requirements because the others are flat  
  
intresting\_requirements <-   
c("cover letter",  
 "cv",  
 "names of recommenders",  
 "diversity statement",  
 "research statement",  
 "teaching statement")  
  
jobdata\_requirements <-   
jobdata %>%   
 select(year\_ad\_posted,  
 starts\_with("documents\_requested")) %>%   
 pivot\_longer(-year\_ad\_posted) %>%   
 mutate(value = case\_when(  
 value == "Not requested in the job ad" ~ 0,  
 value == "One" ~ 1,  
 value == "Two (e.g. two syllabi)" ~ 2,  
 value == "Three" ~ 3,  
 .default = 0  
 )) %>%   
 # trim names a bit  
 mutate(name = str\_remove(name, "documents\_requested\_")) %>%   
 mutate(name = str\_replace\_all(name, "\_", " ")) %>%   
 filter(name %in% intresting\_requirements) %>%   
 mutate(name = str\_wrap(name, 10),  
 year\_ad\_posted = str\_replace(year\_ad\_posted, "-", "\n"))  
  
jobdata\_requirements\_means <-   
jobdata\_requirements %>% # average number requested per year  
 group\_by(year\_ad\_posted,   
 name) %>%   
 summarise(mean\_n = mean(value))  
  
integer\_breaks <- function(n = 5, ...) {  
 fxn <- function(x) {  
 breaks <- floor(pretty(x, n, ...))  
 names(breaks) <- attr(breaks, "labels")  
 breaks  
 }  
 return(fxn)  
}  
  
  
ggplot(jobdata\_requirements\_means) +  
 aes(year\_ad\_posted,   
 mean\_n,  
 group = name) +  
 geom\_smooth(linewidth = 2,  
 colour = "black") +  
 geom\_jitter(data = jobdata\_requirements,  
 aes(year\_ad\_posted,   
 value),  
 alpha = 0.1,  
 height = 0.2,  
 width = 0.1) +  
 facet\_wrap(~name,  
 scales = "free\_y",  
 nrow = 1) +  
 xlab("Year") +  
 ylab("Number requested in job ad") +  
 scale\_y\_continuous(breaks = integer\_breaks()) +  
 theme\_minimal(base\_size = 14) +  
 theme(axis.text.x = element\_text(size = 8),  
 strip.text = element\_text( size = 20))  
  
ggsave(here("analysis",  
 "figures",   
 "fig-requirements-per-year.png"),  
 bg ="white",  
 h = 10, # experiment with h and w to get the right size and proportion   
 w = 20,  
 units = "in",  
 dpi = 900) # make the image nice and crisp

|  |
| --- |
| Figure 1 |

# do the requirements differ for associate positions   
jobdata\_requirements\_by\_rank <-   
jobdata %>%   
 mutate(position\_title = case\_when(  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,   
 "associate") ~ "associate",  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,   
 "assistant") ~ "assistant",  
 str\_detect(title\_of\_position\_tenure\_track\_jobs\_only,   
 "full") ~ "full")) %>%   
 select(position\_title,  
 starts\_with("documents\_requested")) %>%   
 pivot\_longer(-position\_title) %>%   
 mutate(value = case\_when(  
 value == "Not requested in the job ad" ~ 0,  
 value == "One" ~ 1,  
 value == "Two (e.g. two syllabi)" ~ 2,  
 value == "Three" ~ 3,  
 .default = 0  
 )) %>%   
 filter(!is.na(position\_title))   
  
jobdata\_requirements\_by\_rank\_means <-   
 jobdata\_requirements\_by\_rank %>%   
 group\_by(position\_title,  
 name) %>%   
 summarise(mean = mean(value))  
  
ggplot() +  
 geom\_jitter(data = jobdata\_requirements\_by\_rank,  
 aes(position\_title,   
 value),  
 height = 0.05,  
 alpha = 0.1) +  
 geom\_point(data = jobdata\_requirements\_by\_rank\_means,  
 aes(position\_title,  
 mean),  
 size = 4,  
 colour = "red") +  
 facet\_wrap( ~ name,  
 scales = "free\_y") +  
 theme\_minimal()

|  |
| --- |
| Figure 2 |

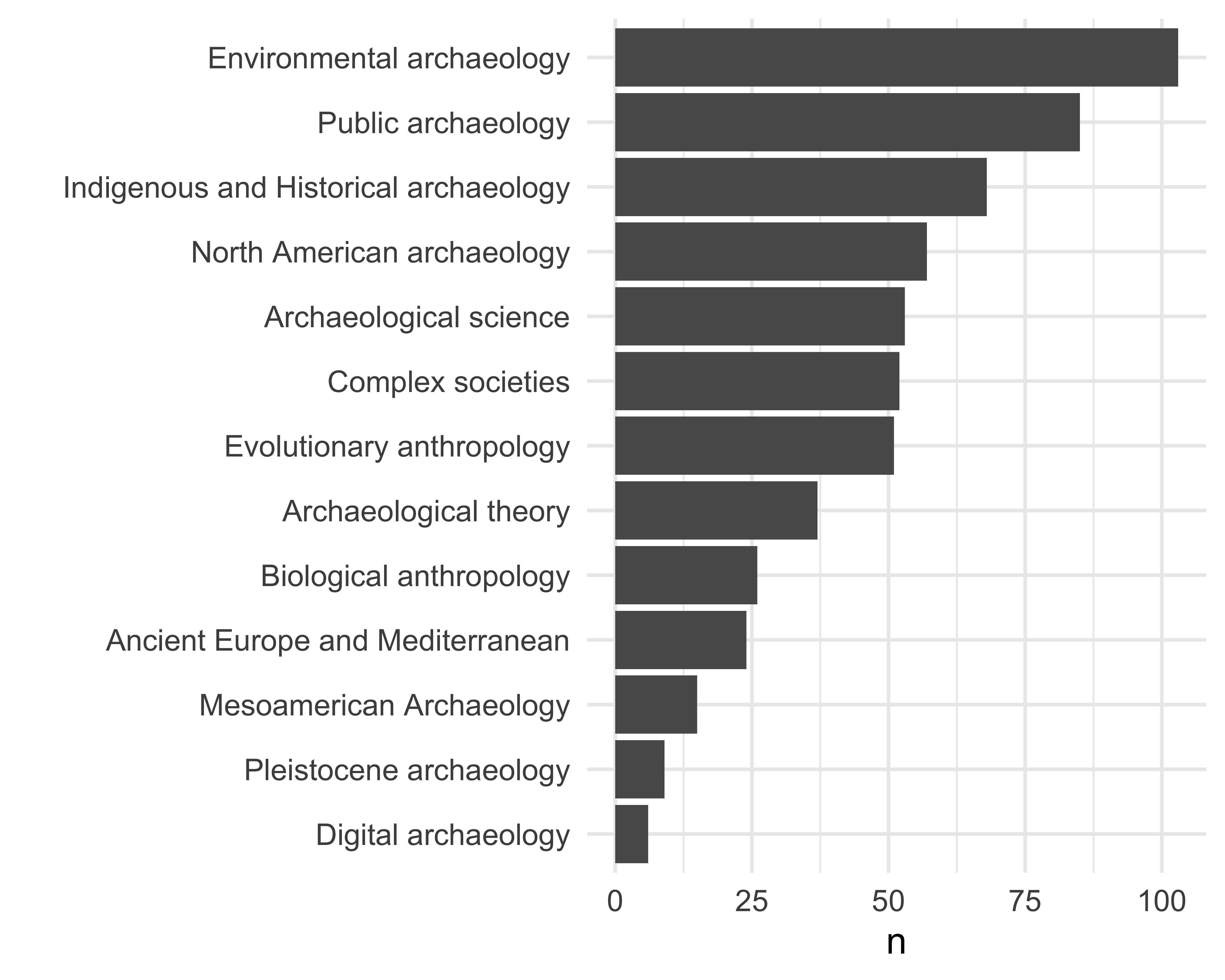
# geographic focus by year  
  
library(googlesheets4)  
library(stringi)  
  
geographic\_foci <-  
read\_sheet("https://docs.google.com/spreadsheets/d/1AHq49pIyChcgJ7rawe6KMWkdIBXydCamvg8Jslob8Ec/edit#gid=0", sheet = "geography")  
  
geographic\_foci\_clean <-  
 map(  
 str\_split(geographic\_foci$`From the data`, ";"),  
 ~.x %>%  
 str\_squish() %>%  
 stri\_remove\_empty())  
  
jobdata\_geo <-  
 jobdata %>%  
 select(geographic\_focus\_of\_position)  
  
jobdata\_geo <-  
 # add one column for each geo region in our categories  
cbind(jobdata\_geo,  
 setNames( lapply(geographic\_foci$Category2, function(x) x=NA),  
 geographic\_foci$Category2) )  
  
for(i in 1:length(geographic\_foci$Category2)){  
  
 this\_location <- geographic\_foci$Category2[i]  
  
 # create the pattern to search for  
 x <- paste0(geographic\_foci\_clean[[i]], collapse = "|")  
  
 # do the search through all the job ads for that pattern  
 y <- str\_detect(jobdata\_geo$geographic\_focus\_of\_position,  
 x)  
  
 # assign back to our data frame in the appropriate location column  
 jobdata\_geo[, this\_location] <- y  
  
}  
  
# BM TODO: check for job ads that have a location, but we're not getting it  
  
united\_states\_regions <-  
str\_subset(geographic\_foci$Category2, "US")  
  
jobdata\_geo\_year <-  
jobdata %>%  
 bind\_cols(jobdata\_geo) %>%  
 select(year\_ad\_posted,  
 geographic\_foci$Category2) %>%  
 pivot\_longer(-year\_ad\_posted) %>%  
 drop\_na()  
  
# how many times each location mentioned?  
jobdata\_geo\_year %>%  
 group\_by(name) %>%  
 summarise(n = sum(value)) %>%  
 arrange(desc(n)) %>%  
 ggplot() +  
 aes(reorder(name, n),  
 n)+  
 geom\_col() +  
 xlab("") +  
 theme\_minimal() +  
 coord\_flip()  
  
# explore trends over time. put a point on the max year  
jobdata\_geo\_year\_tally <-  
jobdata\_geo\_year %>%  
 # exclude those with <20 ads  
 filter(!name %in% c("Canada & Arctic",  
 "Oceania",  
 "Southeast US",  
 "Southwest US",  
 "Western US",  
 "Midwest US",  
 "Northeastern US"  
 )) %>%  
 group\_by(year\_ad\_posted,  
 name) %>%  
 summarise(n = sum(value)) %>%  
 mutate(prop = n / sum(n))  
  
jobdata\_geo\_year\_tally\_max <-  
 jobdata\_geo\_year\_tally %>%  
 group\_by(  
 name ) %>%  
 filter(prop == max(prop))  
  
library(ggrepel)  
  
ggplot() +  
 geom\_smooth(data = jobdata\_geo\_year\_tally,  
 aes(year\_ad\_posted,  
 prop,  
 group = name,  
 colour = name),  
 size = 3,  
 se = FALSE   
 ) +  
 xlab("Year") +  
 ylab("Proportion of all ads") +  
 scale\_colour\_brewer(palette = "Dark2") +  
 guides(colour = guide\_legend("Geographic\nfocus",  
 label.position = "bottom")) +  
 theme\_minimal( base\_size = 28) +  
 theme(legend.position="bottom")   
  
ggsave(here("analysis",  
 "figures",   
 "fig-geo-focus-by-year.png"),  
 bg ="white",  
 h = 10, # experiment with h and w to get the right size and proportion   
 w = 20,  
 units = "in",  
 dpi = 900) # make the image nice and crisp)  
  
  
# what about within the US  
# how many times each location mentioned?  
jobdata\_geo\_year %>%  
 group\_by(name) %>%  
 summarise(n = sum(value)) %>%  
 arrange(desc(n)) %>%  
 filter(name %in% united\_states\_regions) %>%   
 ggplot() +  
 aes(reorder(name, n),  
 n)+  
 geom\_col() +  
 ylab("Number of ads") +  
 xlab("") +  
 theme\_minimal(base\_size = 24) +  
 coord\_flip()  
  
ggsave(here("analysis",  
 "figures",   
 "fig-geo-us-focus-by-year.png"),  
 bg ="white",  
 h = 10, # experiment with h and w to get the right size and proportion   
 w = 20,  
 units = "in",  
 dpi = 900) # make the image nice and crisp)

|  |
| --- |
| Figure 3 |

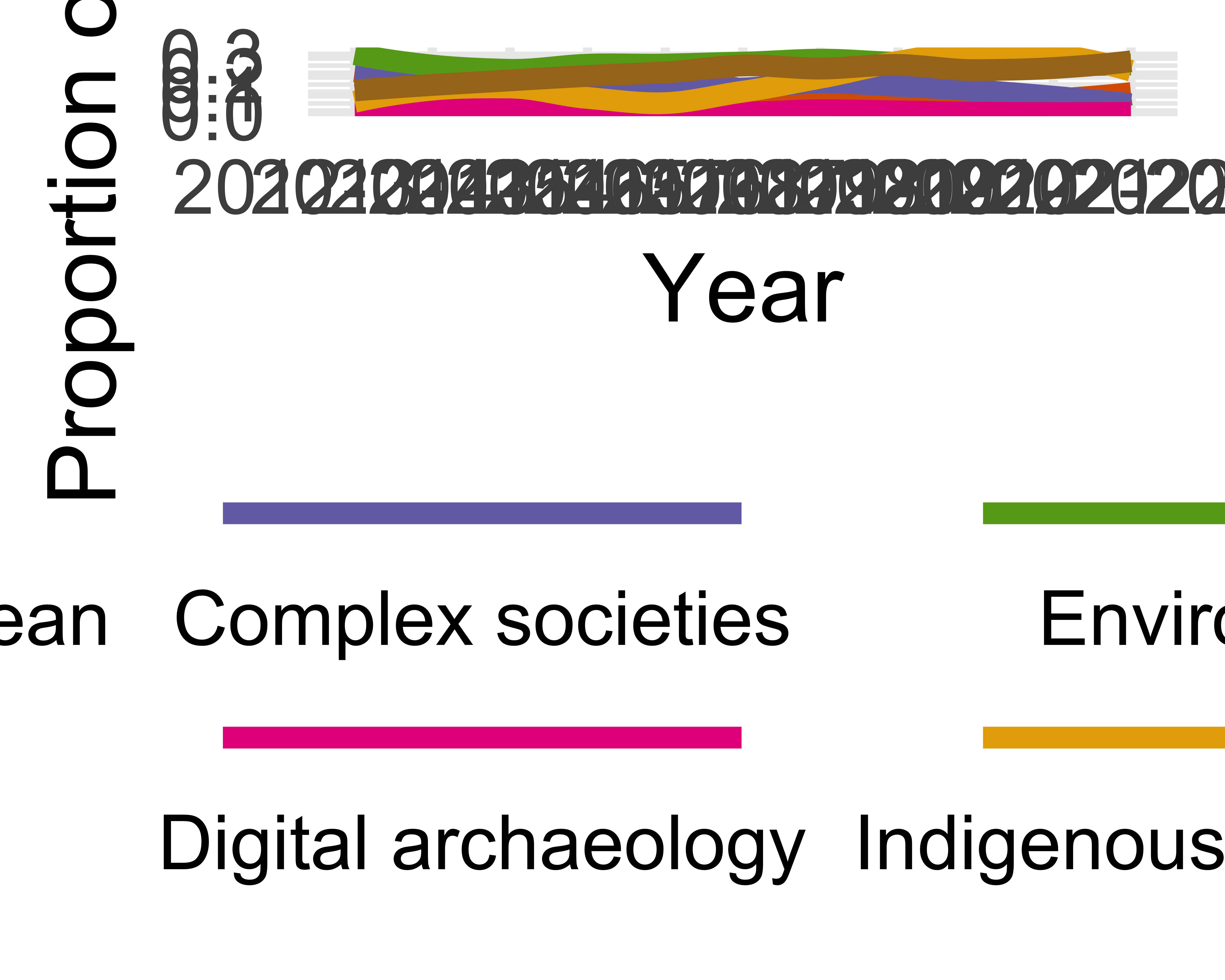
|  |
| --- |
| Figure 4 |

|  |
| --- |
| Figure 5 |

# topical focus by year  
  
library(googlesheets4)  
library(stringi)  
  
topical\_foci <-  
 read\_sheet("https://docs.google.com/spreadsheets/d/1AHq49pIyChcgJ7rawe6KMWkdIBXydCamvg8Jslob8Ec/edit#gid=0",  
 sheet = "topic")  
  
topical\_foci\_clean <-  
 map(  
 str\_split(topical\_foci$`From the data`, ";"),  
 ~.x %>%  
 str\_squish() %>%  
 stri\_remove\_empty() %>%  
 str\_to\_lower)  
  
jobdata\_topic <-  
 jobdata %>%  
 select(topical\_focus\_of\_position) %>%  
 mutate(topical\_focus\_of\_position = str\_to\_lower(topical\_focus\_of\_position))  
  
jobdata\_topic <-  
 # add one column for each topic in our categories  
 cbind(jobdata\_topic,  
 setNames( lapply(topical\_foci$Category, function(x) x=NA),  
 topical\_foci$Category) )  
  
for(i in 1:length(topical\_foci$Category)){  
  
 this\_topic <- topical\_foci$Category[i]  
  
 # create the pattern to search for  
 x <- paste0(topical\_foci\_clean[[i]], collapse = "|")  
  
 # do the search through all the job ads for that pattern  
 y <- str\_detect(jobdata\_topic$topical\_focus\_of\_position,  
 x)  
  
 # assign back to our data frame in the appropriate location column  
 jobdata\_topic[, this\_topic] <- y  
  
}  
  
jobdata\_topic\_year <-  
 jobdata %>%  
 bind\_cols(jobdata\_topic) %>%  
 select(year\_ad\_posted,  
 topical\_foci$Category) %>%  
 pivot\_longer(-year\_ad\_posted) %>%  
 drop\_na()  
  
# how many times each topic mentioned?  
jobdata\_topic\_year %>%  
 group\_by(name) %>%  
 summarise(n = sum(value)) %>%  
 arrange(desc(n)) %>%  
 ggplot() +  
 aes(reorder(name, n),  
 n)+  
 geom\_col() +  
 xlab("") +  
 theme\_minimal() +  
 coord\_flip()

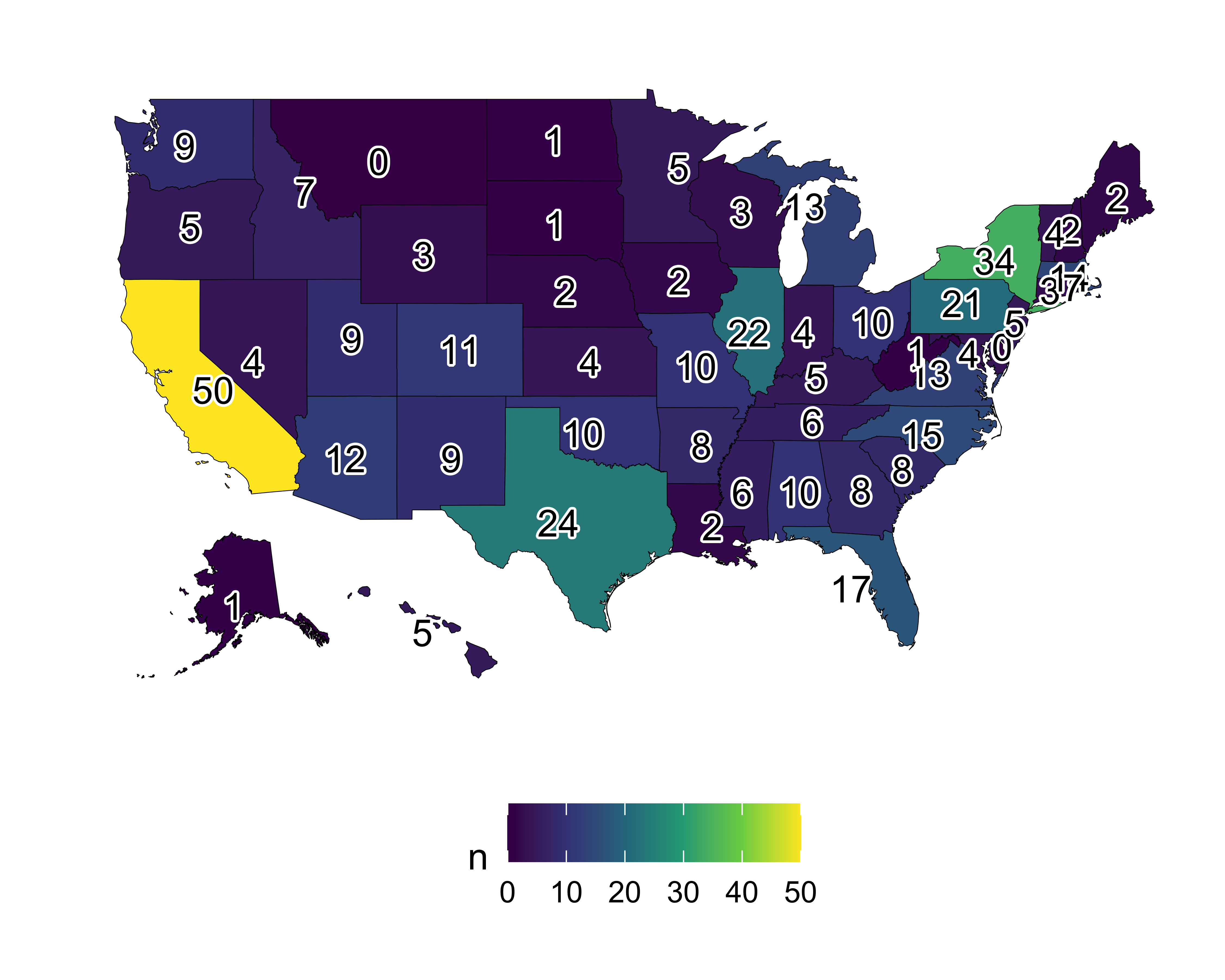


# explore trends over time. put a point on the max year  
jobdata\_topic\_year\_tally <-  
 jobdata\_topic\_year %>%  
 # exclude those with <20 ads  
 filter(!name %in% c("Digital Archaeology",  
 "Pleistocene archaeology",  
 "Mesoamerican Archaeology",  
 "Biological anthropology",  
 "Archaeological theory",  
 "Evolutionary anthropology",  
 "North American archaeology"  
 )) %>%  
 group\_by(year\_ad\_posted,  
 name) %>%  
 summarise(n = sum(value)) %>%  
 mutate(prop = n / sum(n))  
  
jobdata\_topic\_year\_tally\_max <-  
 jobdata\_topic\_year\_tally %>%  
 group\_by(  
 name ) %>%  
 filter(prop == max(prop))  
  
library(ggrepel)  
  
ggplot() +  
 geom\_smooth(data = jobdata\_topic\_year\_tally,  
 aes(year\_ad\_posted,  
 prop,  
 group = name,  
 colour = name),  
 size = 3,  
 span = 0.7,  
 se = FALSE) +  
 xlab("Year") +  
 ylab("Proportion of all ads") +  
 scale\_colour\_brewer(palette = "Dark2") +  
 guides(colour = guide\_legend("Topic\nfocus",  
 label.position = "bottom")) +  
 theme\_minimal( base\_size = 28) +  
 theme(legend.position="bottom")



ggsave(here("analysis",  
 "figures",   
 "fig-topic-focus-by-year.png"),  
 bg ="white",  
 h = 10, # experiment with h and w to get the right size and proportion  
 w = 20,  
 units = "in",  
 dpi = 900) # make the image nice and crisp))

# Draw of map to show which states have done the most hiring in our sample  
  
# get the text in parentheses after the university name that gives the  
# state or country abb  
uni\_state\_country <- # 550  
jobdata %>% # 550 rows  
 select(name\_of\_hiring\_university) %>%  
 mutate(state\_country = regmatches(name\_of\_hiring\_university,  
 gregexpr( "(?<=\\().+?(?=\\))",  
 name\_of\_hiring\_university,  
 perl = T))) %>%  
 unnest(state\_country)  
  
# tally to get counts:  
uni\_state\_country\_tally <-  
uni\_state\_country %>%  
 group\_by(state\_country) %>%  
 tally(sort = TRUE)  
  
# did we get all the job ads?  
# sum(uni\_state\_country\_tally$n) # 550 all of them  
  
state\_to\_st <- function(x){  
 c(state.abb, 'DC')[match(x, c(state.name, 'District of Columbia'))]  
}  
  
state\_name\_and\_abb <-  
enframe(state.name, value = 'state\_name') %>%  
 mutate(state\_abbr = state\_to\_st(state\_name))  
  
# filter to get US states only  
uni\_state\_country\_tally\_us <-  
uni\_state\_country\_tally %>%  
 filter(state\_country %in% state.abb) %>%  
 select(state\_abbr = state\_country, n) %>%  
 # make sure we have all states in the dataframe  
 # even those with no jobs  
 right\_join(state\_name\_and\_abb) %>%  
 select(state = state\_name, n, state\_abbr) %>%  
 mutate(state = tolower(state)) %>%  
 mutate(n = ifelse(is.na(n), 0, n))  
  
# how many jobs ads now?  
# sum(uni\_state\_country\_tally\_us$n) # 433, 78% of the total  
  
library(ggplot2)  
library(fiftystater)  
library(tidyverse)  
library(ggrepel)  
  
data("fifty\_states")  
  
ggplot(data= uni\_state\_country\_tally\_us,  
 aes(map\_id = state)) +  
 geom\_map(aes(fill = n),  
 color= "black",  
 linewidth = 0.1,  
 map = fifty\_states) +  
 expand\_limits(x = fifty\_states$long,  
 y = fifty\_states$lat) +  
 coord\_map() +  
 geom\_text\_repel(data = fifty\_states %>%  
 group\_by(id) %>%  
 summarise(lat = mean(c(max(lat), min(lat))),  
 long = mean(c(max(long), min(long)))) %>%  
 mutate(state = id) %>%  
 left\_join(uni\_state\_country\_tally\_us,  
 by = "state"),  
 aes(x = long,  
 y = lat,  
 label = n,  
 bg.color = "white",  
 bg.r = 0.1),  
 force = 0,  
 force\_pull = 100)+  
 scale\_x\_continuous(breaks = NULL) +  
 scale\_y\_continuous(breaks = NULL) +  
 labs(x = "",  
 y = "") +  
 theme(legend.position = "bottom",  
 panel.background = element\_blank()) +  
 scale\_fill\_viridis\_c()

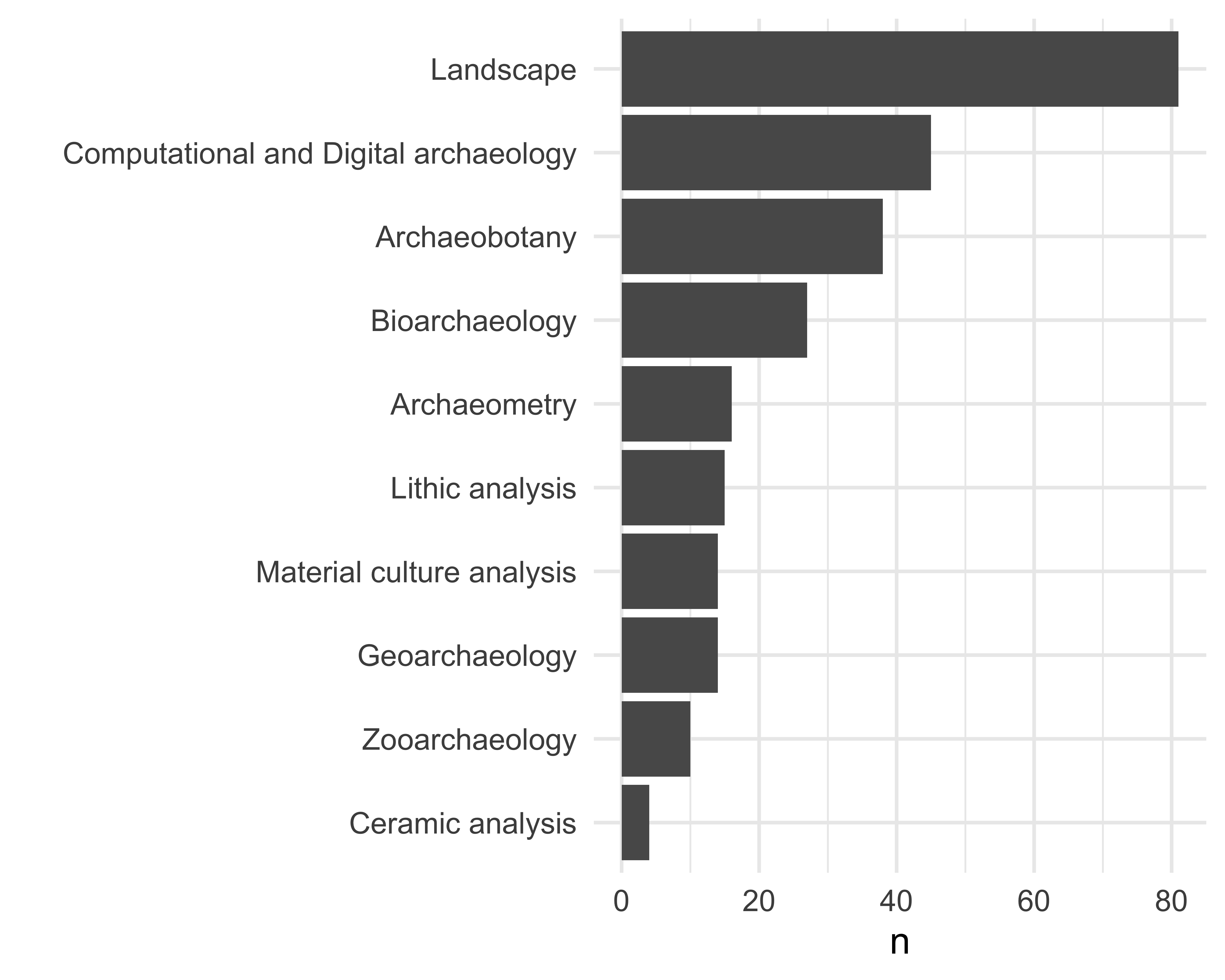


ggsave(here("analysis",  
 "figures",  
 "fig-us-state-map.png"),  
 bg ="white",  
 h = 10, # experiment with h and w to get the right size and proportion  
 w = 12,  
 units = "in",  
 dpi = 900) # make the image nice and crisp))

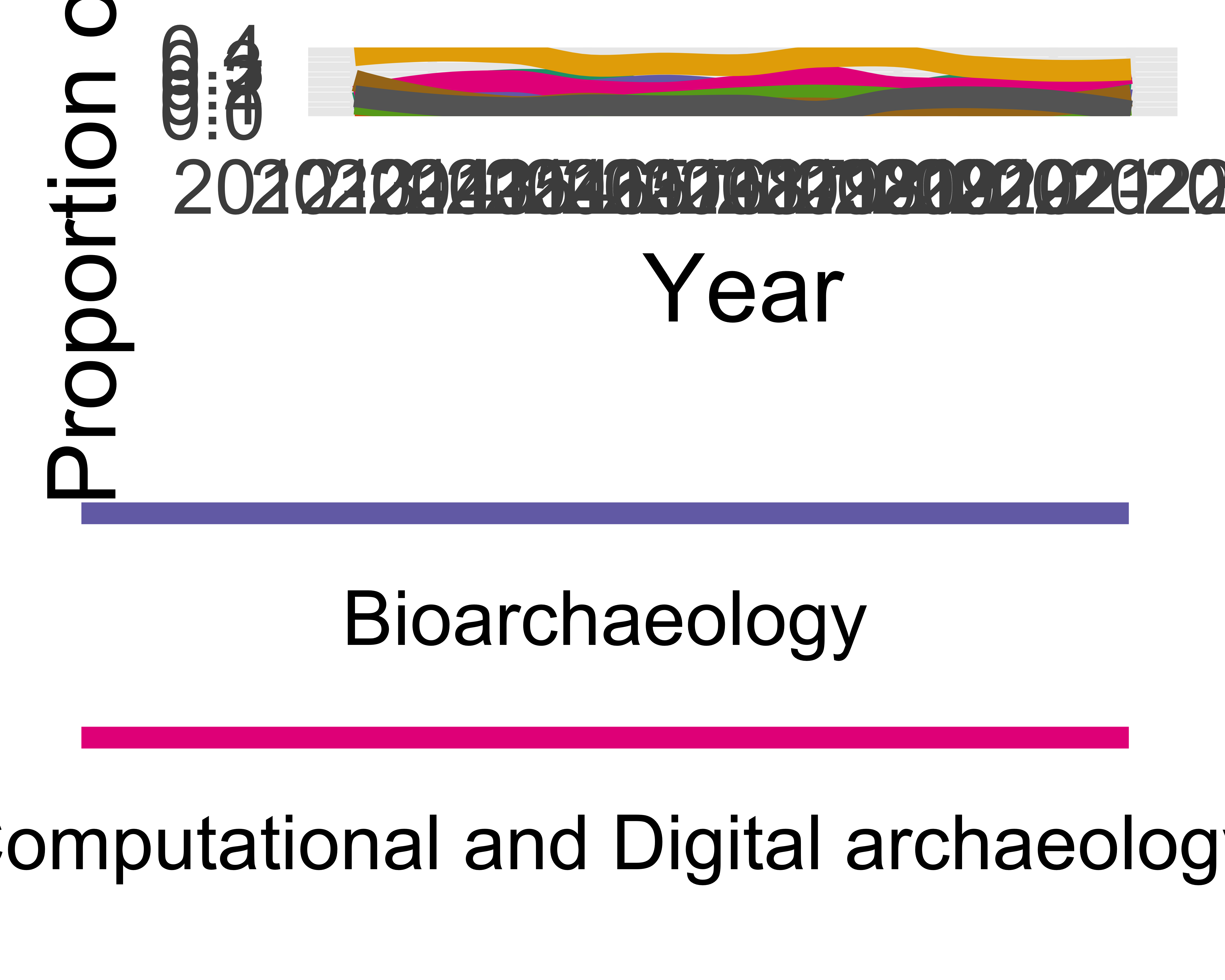
# method focus by year  
  
library(googlesheets4)  
library(stringi)  
1

[1] 1

method\_foci <-  
 read\_sheet("https://docs.google.com/spreadsheets/d/1AHq49pIyChcgJ7rawe6KMWkdIBXydCamvg8Jslob8Ec/edit#gid=0",  
 sheet = "method")  
  
method\_foci\_clean <-  
 map(  
 str\_split(method\_foci$`From the data`, ";"),  
 ~.x %>%  
 str\_squish() %>%  
 stri\_remove\_empty() %>%  
 str\_to\_lower)  
  
jobdata\_method <-  
 jobdata %>%  
 select(methods\_focus\_of\_position) %>%  
 mutate(methods\_focus\_of\_position = str\_to\_lower(methods\_focus\_of\_position))  
  
jobdata\_method <-  
 # add one column for each topic in our categories  
 cbind(jobdata\_method,  
 setNames( lapply(method\_foci$Category, function(x) x=NA),  
 method\_foci$Category) )  
  
for(i in 1:length(method\_foci$Category)){  
  
 this\_method <- method\_foci$Category[i]  
  
 # create the pattern to search for  
 x <- paste0(method\_foci\_clean[[i]], collapse = "|")  
  
 # do the search through all the job ads for that pattern  
 y <- str\_detect(jobdata\_method$methods\_focus\_of\_position,  
 x)  
  
 # assign back to our data frame in the appropriate location column  
 jobdata\_method[, this\_method] <- y  
  
}  
  
jobdata\_method\_year <-  
 jobdata %>%  
 bind\_cols(jobdata\_method) %>%  
 select(year\_ad\_posted,  
 method\_foci$Category) %>%  
 pivot\_longer(-year\_ad\_posted) %>%  
 drop\_na()  
  
# how many times each topic mentioned?  
jobdata\_method\_year %>%  
 group\_by(name) %>%  
 summarise(n = sum(value)) %>%  
 arrange(desc(n)) %>%  
 ggplot() +  
 aes(reorder(name, n),  
 n)+  
 geom\_col() +  
 xlab("") +  
 theme\_minimal() +  
 coord\_flip()



# explore trends over time. put a point on the max year  
jobdata\_method\_year\_tally <-  
 jobdata\_method\_year %>%  
 # exclude those with <20 ads  
 filter(!name %in% c("Material culture analysis",  
 "Ceramic analysis"  
 )) %>%  
 group\_by(year\_ad\_posted,  
 name) %>%  
 summarise(n = sum(value)) %>%  
 mutate(prop = n / sum(n))  
  
library(ggrepel)  
  
ggplot() +  
 geom\_smooth(data = jobdata\_method\_year\_tally,  
 aes(year\_ad\_posted,  
 prop,  
 group = name,  
 colour = name),  
 size = 3,  
 span = 0.7,  
 se = FALSE) +  
 xlab("Year") +  
 ylab("Proportion of all ads") +  
 guides(colour = guide\_legend("Method\nfocus",  
 label.position = "bottom")) +  
 theme\_minimal( base\_size = 28) +  
 scale\_colour\_brewer(palette = "Dark2") +  
 theme(legend.position="bottom")



ggsave(here("analysis",  
 "figures",   
 "fig-method-focus-by-year.png"),  
 bg ="white",  
 h = 10, # experiment with h and w to get the right size and proportion  
 w = 20,  
 units = "in",  
 dpi = 900) # make the image nice and crisp))

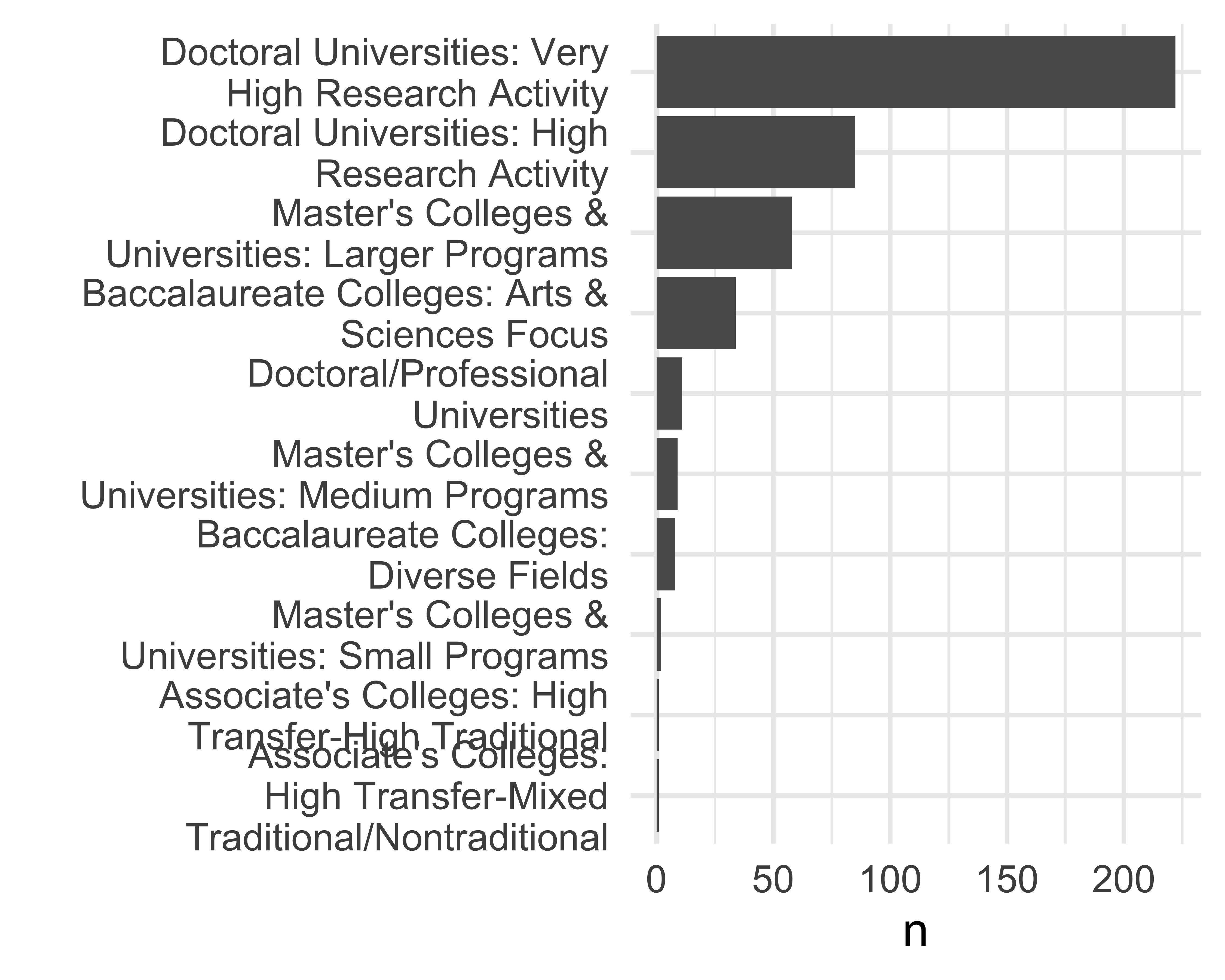
# Carnegie Classification  
library(googlesheets4)  
library(stringi)  
  
# data from https://carnegieclassifications.acenet.edu/resource/2021-update-public-file/  
CC <-  
 read\_sheet( "https://docs.google.com/spreadsheets/d/1Oe\_3X-OR7iuUThb6Tp48prBpbHrWjU4osmHKi1Hbqgc/edit#gid=1922578308",  
 sheet = "Values1")  
Basic2021 <-  
 read\_sheet( "https://docs.google.com/spreadsheets/d/1Oe\_3X-OR7iuUThb6Tp48prBpbHrWjU4osmHKi1Hbqgc/edit#gid=1922578308",  
 sheet = "Data1")   
 # remove a bunch of text from uni names to improve our joins  
  
uni\_name <- # 547  
 jobdata %>% # 547 rows  
 # remove parentheses and their contents  
 mutate(name = str\_squish(str\_replace(name\_of\_hiring\_university,  
 "\\s\*\\(.\*?\\)$",   
 ""))) %>%   
 mutate(name = str\_replace(name,  
 "California State University,[[:space:]]",  
 "California State University-")) %>%   
   
 left\_join(Basic2021,   
 keep = TRUE )   
  
# how many schools did we match with the CC data  
uni\_name %>%   
 filter(!is.na(basic2021)) %>%   
 nrow() # 245, 290, 300, 313, 404, 430 / 433 for US unis

[1] 431

# which schools don't match with the CC data?  
uni\_name %>%   
 select(name.x, name.y) %>%   
 filter(is.na(name.y)) %>%   
 distinct()

# A tibble: 81 × 2  
 name.x name.y  
 <chr> <chr>   
 1 University of British Columbia <NA>   
 2 University of Warwick <NA>   
 3 University of New Brunswick <NA>   
 4 University of Toronto <NA>   
 5 Université catholique de Louvain <NA>   
 6 Technische Universitat Darmstadt <NA>   
 7 La Trobe University <NA>   
 8 University of Toronto Scarborough <NA>   
 9 The Hong Kong Polytechnic University <NA>   
10 Cambridge <NA>   
# ℹ 71 more rows

uni\_name\_tally <-   
uni\_name %>%   
 left\_join(CC,  
 join\_by("basic2021" == "Value")) %>%   
 group\_by(Category) %>%   
 tally(sort = TRUE) %>%   
 drop\_na()  
  
  
uni\_name\_tally %>%   
 mutate(Category = str\_wrap(Category, width = 30)) %>%   
ggplot() +  
 aes(reorder(Category, n), n) +  
 geom\_col() +  
 coord\_flip() +  
 xlab("") +  
 theme\_minimal(base\_size = 14)

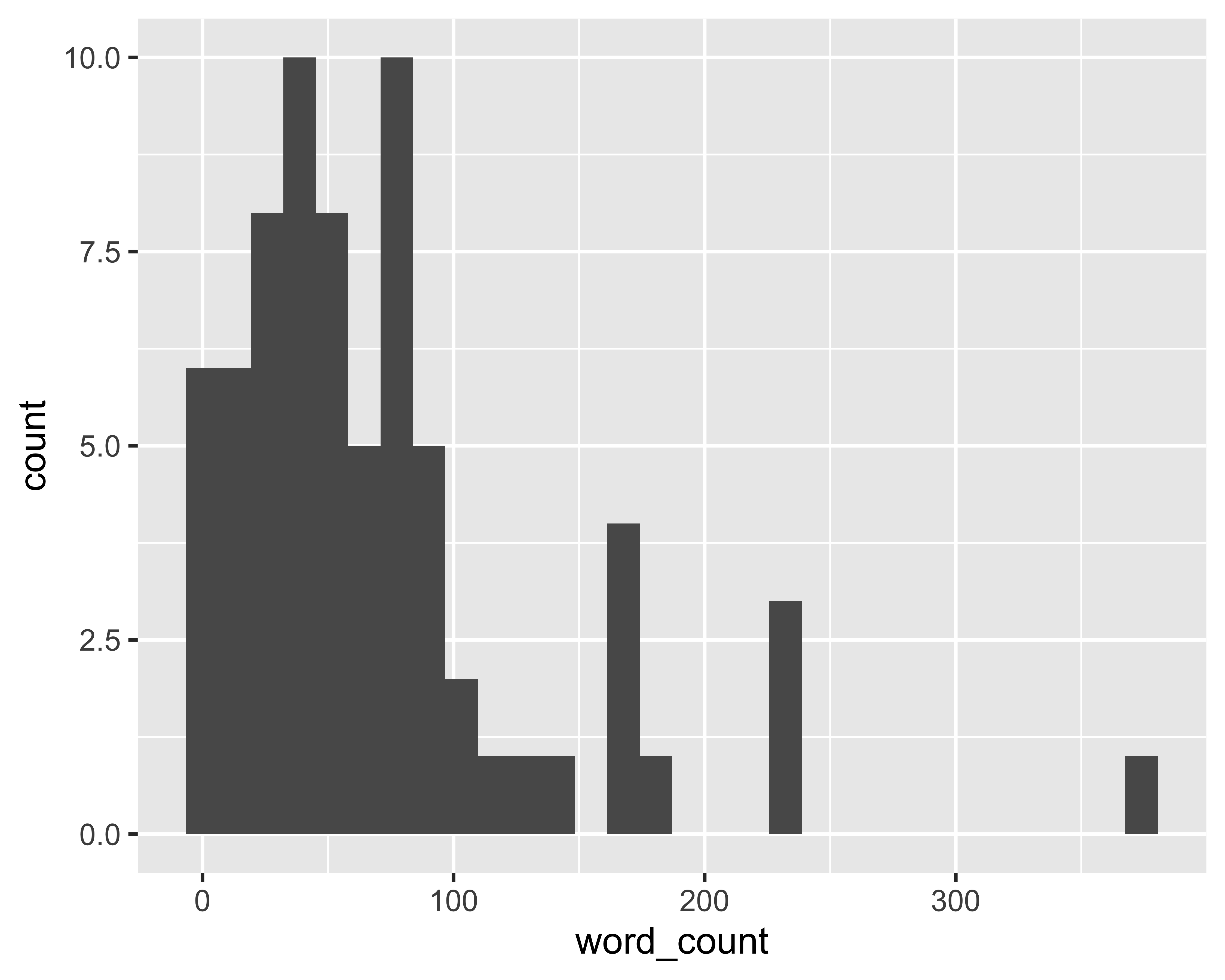


ggsave(here("analysis",  
 "figures",   
 "fig-carnegie-classification.png"),  
 bg ="white",  
 h = 7, # experiment with h and w to get the right size and proportion  
 w = 12,  
 units = "in",  
 dpi = 900) # make the image nice and crisp))  
  
  
view(jobdata)

# EEO  
library(stringr)  
library(dplyr)  
library(stopwords)  
library(tidyverse)  
library(tidytext)  
library(tm)  
  
stopwords <- stopwords("en")  
  
EEO <- jobdata %>%  
 select(equal\_employment\_opportunity\_statement) %>%   
 drop\_na  
  
remove\_stopwords <- function(text, stopwords) {  
 pattern <- paste0("\\b(", paste(stopwords, collapse = "|"), ")\\b")  
 str\_remove\_all(text, pattern)  
}  
  
EEO\_wordcount <- EEO %>%  
 mutate(word\_count = str\_count(equal\_employment\_opportunity\_statement, "\\S+")) %>%  
 print()

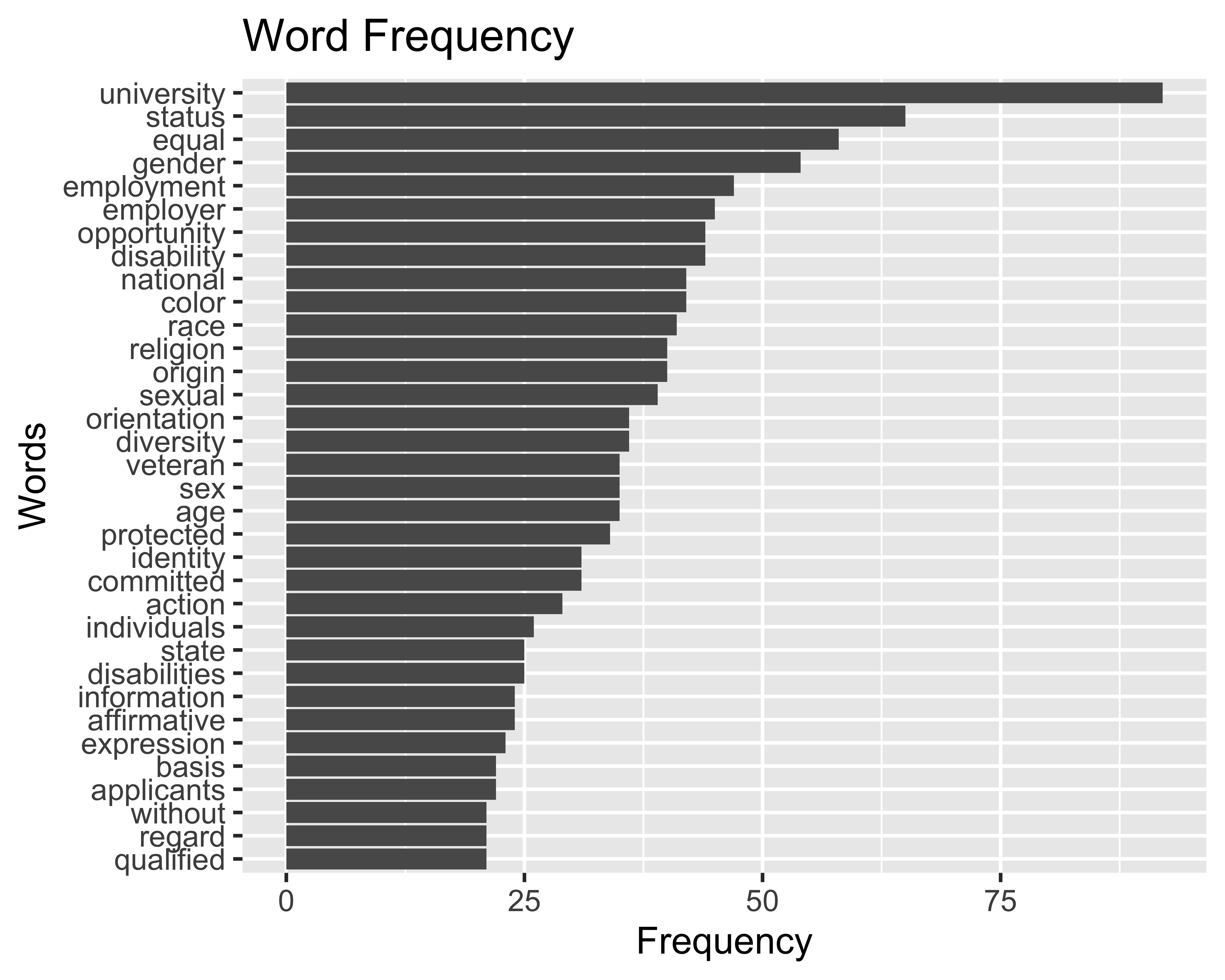
# A tibble: 72 × 2  
 equal\_employment\_opportunity\_statement word\_count  
 <chr> <int>  
 1 "The University of Hawai'i is an Equal Opportunity/Affirmative Ac… 228  
 2 "The University of Southern Maine is an EEO/AA employer, and does… 70  
 3 "Bryn Mawr College shall abide by the requirements of 41 CFR §§ 6… 84  
 4 "We are an equal opportunity employer and all qualified applicant… 46  
 5 "Centre College shall abide by the requirements of 41 CFR §§ 60-1… 83  
 6 "The University of New Brunswick is committed to employment equit… 80  
 7 "Diversity Statement\nThe University of Toronto is strongly commi… 185  
 8 "The University of South Carolina is an affirmative action, equal… 61  
 9 "The Ohio State University is an equal opportunity employer. All … 36  
10 "Marshall University is an AA/EO employer dedicated to increasing… 87  
# ℹ 62 more rows

ggplot(EEO\_wordcount) +  
 aes(word\_count) +  
 geom\_histogram()



EEO\_clean <- EEO %>%  
 mutate(equal\_employment\_opportunity\_statement = str\_to\_lower(equal\_employment\_opportunity\_statement),   
 equal\_employment\_opportunity\_statement = str\_remove\_all(equal\_employment\_opportunity\_statement, "[[:punct:]]"),  
 equal\_employment\_opportunity\_statement = remove\_stopwords(equal\_employment\_opportunity\_statement, stopwords))  
  
# this doesn't work, need help to troubleshoot  
EEO\_word <- EEO\_clean %>%  
 unnest\_tokens(word,   
 equal\_employment\_opportunity\_statement) %>%  
 anti\_join(get\_stopwords()) %>%  
 count(word, sort = TRUE)  
  
# here's the list of words of interest  
community\_word <- c("women","color","religion","race","age","veteran","disability")  
  
# prepare a dataframe to store the count of words of interest  
df <- data.frame(matrix(ncol = length(community\_word),   
 nrow = 0))  
colnames(df) <- community\_word  
  
for(i in 1: nrow(EEO\_clean)){  
 df[i, ] <- str\_detect(EEO\_clean[i, ],   
 community\_word)  
}  
  
# detect if the words of interest is present or not.  
EEO\_clean\_community\_word <-   
 bind\_cols(EEO\_clean, df) %>%   
 rowwise() %>%   
 mutate(community\_word\_tally = sum(c\_across(all\_of(community\_word)),   
 na.rm = T))

EEO\_word %>%   
 filter(n >= 20) %>%   
ggplot() +  
 aes(x = reorder(word, n), y = n) +  
 geom\_col() +  
 coord\_flip() + # Flips the axes for easier reading of text  
 labs(x = "Words", y = "Frequency", title = "Word Frequency")



# Discussion

# Conclusion

# Acknowledgements

# References

Lightfoot, E., Franklin, C., Beltran, R., 2021. Preparing for the academic job market: A guide for social work doctoral students and their mentors. Journal of Social Work Education 57, 153–164.

Musial, J., Holmes, C., 2018. Five-year study on hiring trends in gender, women’s, and feminist studies. Feminist Studies 44, 253–272.

### Colophon

This report was generated on 2024-09-17 16:48:23.65896 using the following computational environment and dependencies:

# which R packages and versions?  
if ("devtools" %in% installed.packages()) devtools::session\_info()

─ Session info ───────────────────────────────────────────────────────────────  
 setting value  
 version R version 4.4.1 (2024-06-14)  
 os macOS Sonoma 14.6.1  
 system x86\_64, darwin20  
 ui X11  
 language (EN)  
 collate en\_US.UTF-8  
 ctype en\_US.UTF-8  
 tz America/Los\_Angeles  
 date 2024-09-17  
 pandoc 3.1.11 @ /Applications/RStudio.app/Contents/Resources/app/quarto/bin/tools/x86\_64/ (via rmarkdown)  
  
─ Packages ───────────────────────────────────────────────────────────────────  
 package \* version date (UTC) lib source  
 askpass 1.2.0 2023-09-03 [1] CRAN (R 4.4.0)  
 beeswarm 0.4.0 2021-06-01 [1] CRAN (R 4.4.0)  
 bit 4.0.5 2022-11-15 [1] CRAN (R 4.4.0)  
 bit64 4.0.5 2020-08-30 [1] CRAN (R 4.4.0)  
 cachem 1.1.0 2024-05-16 [1] CRAN (R 4.4.0)  
 cellranger 1.1.0 2016-07-27 [1] CRAN (R 4.4.0)  
 cli 3.6.2 2023-12-11 [1] CRAN (R 4.4.0)  
 colorspace 2.1-0 2023-01-23 [1] CRAN (R 4.4.0)  
 cowplot \* 1.1.3 2024-01-22 [1] CRAN (R 4.4.0)  
 crayon 1.5.2 2022-09-29 [1] CRAN (R 4.4.0)  
 curl 5.2.1 2024-03-01 [1] CRAN (R 4.4.0)  
 devtools 2.4.5 2022-10-11 [1] CRAN (R 4.4.0)  
 digest 0.6.35 2024-03-11 [1] CRAN (R 4.4.0)  
 dplyr \* 1.1.4 2023-11-17 [1] CRAN (R 4.4.0)  
 ellipsis 0.3.2 2021-04-29 [1] CRAN (R 4.4.0)  
 evaluate 0.23 2023-11-01 [1] CRAN (R 4.4.0)  
 fansi 1.0.6 2023-12-08 [1] CRAN (R 4.4.0)  
 farver 2.1.2 2024-05-13 [1] CRAN (R 4.4.0)  
 fastmap 1.2.0 2024-05-15 [1] CRAN (R 4.4.0)  
 fiftystater \* 1.0.1 2024-09-17 [1] Github (wmurphyrd/fiftystater@28e7fa5)  
 forcats \* 1.0.0 2023-01-29 [1] CRAN (R 4.4.0)  
 fs 1.6.4 2024-04-25 [1] CRAN (R 4.4.0)  
 gargle 1.5.2 2023-07-20 [1] CRAN (R 4.4.0)  
 generics 0.1.3 2022-07-05 [1] CRAN (R 4.4.0)  
 ggbeeswarm \* 0.7.2 2023-04-29 [1] CRAN (R 4.4.0)  
 ggplot2 \* 3.5.1 2024-04-23 [1] CRAN (R 4.4.0)  
 ggrepel \* 0.9.5 2024-01-10 [1] CRAN (R 4.4.0)  
 glue 1.7.0 2024-01-09 [1] CRAN (R 4.4.0)  
 googledrive 2.1.1 2023-06-11 [1] CRAN (R 4.4.0)  
 googlesheets4 \* 1.1.1 2023-06-11 [1] CRAN (R 4.4.0)  
 gtable 0.3.5 2024-04-22 [1] CRAN (R 4.4.0)  
 here \* 1.0.1 2020-12-13 [1] CRAN (R 4.4.0)  
 hms 1.1.3 2023-03-21 [1] CRAN (R 4.4.0)  
 htmltools 0.5.8.1 2024-04-04 [1] CRAN (R 4.4.0)  
 htmlwidgets 1.6.4 2023-12-06 [1] CRAN (R 4.4.0)  
 httpuv 1.6.15 2024-03-26 [1] CRAN (R 4.4.0)  
 httr 1.4.7 2023-08-15 [1] CRAN (R 4.4.0)  
 janeaustenr 1.0.0 2022-08-26 [1] CRAN (R 4.4.0)  
 janitor 2.2.0 2023-02-02 [1] CRAN (R 4.4.0)  
 jsonlite 1.8.8 2023-12-04 [1] CRAN (R 4.4.0)  
 knitr 1.47 2024-05-29 [1] CRAN (R 4.4.0)  
 labeling 0.4.3 2023-08-29 [1] CRAN (R 4.4.0)  
 later 1.3.2 2023-12-06 [1] CRAN (R 4.4.0)  
 lattice 0.22-6 2024-03-20 [1] CRAN (R 4.4.1)  
 lifecycle 1.0.4 2023-11-07 [1] CRAN (R 4.4.0)  
 lubridate \* 1.9.3 2023-09-27 [1] CRAN (R 4.4.0)  
 magrittr 2.0.3 2022-03-30 [1] CRAN (R 4.4.0)  
 mapproj 1.2.11 2023-01-12 [1] CRAN (R 4.4.0)  
 maps 3.4.2 2023-12-15 [1] CRAN (R 4.4.0)  
 Matrix 1.7-0 2024-04-26 [1] CRAN (R 4.4.1)  
 memoise 2.0.1 2021-11-26 [1] CRAN (R 4.4.0)  
 mgcv 1.9-1 2023-12-21 [1] CRAN (R 4.4.1)  
 mime 0.12 2021-09-28 [1] CRAN (R 4.4.0)  
 miniUI 0.1.1.1 2018-05-18 [1] CRAN (R 4.4.0)  
 munsell 0.5.1 2024-04-01 [1] CRAN (R 4.4.0)  
 nlme 3.1-164 2023-11-27 [1] CRAN (R 4.4.1)  
 NLP \* 0.3-0 2024-08-05 [1] CRAN (R 4.4.0)  
 openssl 2.2.0 2024-05-16 [1] CRAN (R 4.4.0)  
 pillar 1.9.0 2023-03-22 [1] CRAN (R 4.4.0)  
 pkgbuild 1.4.4 2024-03-17 [1] CRAN (R 4.4.0)  
 pkgconfig 2.0.3 2019-09-22 [1] CRAN (R 4.4.0)  
 pkgload 1.3.4 2024-01-16 [1] CRAN (R 4.4.0)  
 profvis 0.3.8 2023-05-02 [1] CRAN (R 4.4.0)  
 promises 1.3.0 2024-04-05 [1] CRAN (R 4.4.0)  
 purrr \* 1.0.2 2023-08-10 [1] CRAN (R 4.4.0)  
 R6 2.5.1 2021-08-19 [1] CRAN (R 4.4.0)  
 ragg 1.3.2 2024-05-15 [1] CRAN (R 4.4.0)  
 rappdirs 0.3.3 2021-01-31 [1] CRAN (R 4.4.0)  
 RColorBrewer 1.1-3 2022-04-03 [1] CRAN (R 4.4.0)  
 Rcpp 1.0.12 2024-01-09 [1] CRAN (R 4.4.0)  
 readr \* 2.1.5 2024-01-10 [1] CRAN (R 4.4.0)  
 remotes 2.5.0 2024-03-17 [1] CRAN (R 4.4.0)  
 rlang 1.1.4 2024-06-04 [1] CRAN (R 4.4.0)  
 rmarkdown 2.27 2024-05-17 [1] CRAN (R 4.4.0)  
 rprojroot 2.0.4 2023-11-05 [1] CRAN (R 4.4.0)  
 rstudioapi 0.16.0 2024-03-24 [1] CRAN (R 4.4.0)  
 rvest \* 1.0.4 2024-02-12 [1] CRAN (R 4.4.0)  
 scales 1.3.0 2023-11-28 [1] CRAN (R 4.4.0)  
 selectr 0.4-2 2019-11-20 [1] CRAN (R 4.4.0)  
 sessioninfo 1.2.2 2021-12-06 [1] CRAN (R 4.4.0)  
 shiny 1.8.1.1 2024-04-02 [1] CRAN (R 4.4.0)  
 slam 0.1-53 2024-09-02 [1] CRAN (R 4.4.1)  
 snakecase 0.11.1 2023-08-27 [1] CRAN (R 4.4.0)  
 SnowballC 0.7.1 2023-04-25 [1] CRAN (R 4.4.0)  
 stopwords \* 2.3 2021-10-28 [1] CRAN (R 4.4.0)  
 stringi \* 1.8.4 2024-05-06 [1] CRAN (R 4.4.0)  
 stringr \* 1.5.1 2023-11-14 [1] CRAN (R 4.4.0)  
 systemfonts 1.1.0 2024-05-15 [1] CRAN (R 4.4.0)  
 textshaping 0.4.0 2024-05-24 [1] CRAN (R 4.4.0)  
 tibble \* 3.2.1 2023-03-20 [1] CRAN (R 4.4.0)  
 tidyr \* 1.3.1 2024-01-24 [1] CRAN (R 4.4.0)  
 tidyselect 1.2.1 2024-03-11 [1] CRAN (R 4.4.0)  
 tidytext \* 0.4.2 2024-04-10 [1] CRAN (R 4.4.0)  
 tidyverse \* 2.0.0 2023-02-22 [1] CRAN (R 4.4.0)  
 timechange 0.3.0 2024-01-18 [1] CRAN (R 4.4.0)  
 tm \* 0.7-14 2024-08-13 [1] CRAN (R 4.4.1)  
 tokenizers 0.3.0 2022-12-22 [1] CRAN (R 4.4.0)  
 tzdb 0.4.0 2023-05-12 [1] CRAN (R 4.4.0)  
 urlchecker 1.0.1 2021-11-30 [1] CRAN (R 4.4.0)  
 usethis 2.2.3 2024-02-19 [1] CRAN (R 4.4.0)  
 utf8 1.2.4 2023-10-22 [1] CRAN (R 4.4.0)  
 vctrs 0.6.5 2023-12-01 [1] CRAN (R 4.4.0)  
 vipor 0.4.7 2023-12-18 [1] CRAN (R 4.4.0)  
 viridisLite 0.4.2 2023-05-02 [1] CRAN (R 4.4.0)  
 vroom 1.6.5 2023-12-05 [1] CRAN (R 4.4.0)  
 withr 3.0.0 2024-01-16 [1] CRAN (R 4.4.0)  
 xfun 0.44 2024-05-15 [1] CRAN (R 4.4.0)  
 xml2 1.3.6 2023-12-04 [1] CRAN (R 4.4.0)  
 xtable 1.8-4 2019-04-21 [1] CRAN (R 4.4.0)  
 yaml 2.3.8 2023-12-11 [1] CRAN (R 4.4.0)  
  
 [1] /Library/Frameworks/R.framework/Versions/4.4-x86\_64/Resources/library  
  
──────────────────────────────────────────────────────────────────────────────

The current Git commit details are:

# what commit is this file at?   
if ("git2r" %in% installed.packages() & git2r::in\_repository(path = ".")) git2r::repository(here::here())

Local: main /Users/bmarwick/Downloads/archyjobads  
Remote: main @ origin (https://github.com/benmarwick/archyjobads)  
Head: [e6186c9] 2024-05-13: plotting EEO word freq