Essay\_markdown\_final

River Kim

2024-04-07

# Set up

Data importing (Following Course Book “Assessment Data”)

library(ggplot2)  
library(plyr)  
library(gdata)

##   
## Attaching package: 'gdata'

## The following object is masked from 'package:stats':  
##   
## nobs

## The following object is masked from 'package:utils':  
##   
## object.size

## The following object is masked from 'package:base':  
##   
## startsWith

library(stringr)  
library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:gdata':  
##   
## first, last

## Prep Osnabrugge et al.  
data = fread("/Users/garamkim/Downloads/dataverse\_files/uk\_data.csv", encoding="UTF-8")  
data$date = as.Date(data$date)  
  
#Create time variable  
data$time= NA  
data$time[data$date>=as.Date("2001-01-01") & data$date<=as.Date("2001-06-30")] = "01/1"  
data$time[data$date>=as.Date("2001-07-01") & data$date<=as.Date("2001-12-31")] = "01/2"  
data$time[data$date>=as.Date("2002-01-01") & data$date<=as.Date("2002-06-30")] = "02/1"  
data$time[data$date>=as.Date("2002-07-01") & data$date<=as.Date("2002-12-31")] = "02/2"  
data$time[data$date>=as.Date("2003-01-01") & data$date<=as.Date("2003-06-30")] = "03/1"  
data$time[data$date>=as.Date("2003-07-01") & data$date<=as.Date("2003-12-31")] = "03/2"  
data$time[data$date>=as.Date("2004-01-01") & data$date<=as.Date("2004-06-30")] = "04/1"  
data$time[data$date>=as.Date("2004-07-01") & data$date<=as.Date("2004-12-31")] = "04/2"  
data$time[data$date>=as.Date("2005-01-01") & data$date<=as.Date("2005-06-30")] = "05/1"  
data$time[data$date>=as.Date("2005-07-01") & data$date<=as.Date("2005-12-31")] = "05/2"  
data$time[data$date>=as.Date("2006-01-01") & data$date<=as.Date("2006-06-30")] = "06/1"  
data$time[data$date>=as.Date("2006-07-01") & data$date<=as.Date("2006-12-31")] = "06/2"  
data$time[data$date>=as.Date("2007-01-01") & data$date<=as.Date("2007-06-30")] = "07/1"  
data$time[data$date>=as.Date("2007-07-01") & data$date<=as.Date("2007-12-31")] = "07/2"  
data$time[data$date>=as.Date("2008-01-01") & data$date<=as.Date("2008-06-30")] = "08/1"  
data$time[data$date>=as.Date("2008-07-01") & data$date<=as.Date("2008-12-31")] = "08/2"  
data$time[data$date>=as.Date("2009-01-01") & data$date<=as.Date("2009-06-30")] = "09/1"  
data$time[data$date>=as.Date("2009-07-01") & data$date<=as.Date("2009-12-31")] = "09/2"  
data$time[data$date>=as.Date("2010-01-01") & data$date<=as.Date("2010-06-30")] = "10/1"  
data$time[data$date>=as.Date("2010-07-01") & data$date<=as.Date("2010-12-31")] = "10/2"  
data$time[data$date>=as.Date("2011-01-01") & data$date<=as.Date("2011-06-30")] = "11/1"  
data$time[data$date>=as.Date("2011-07-01") & data$date<=as.Date("2011-12-31")] = "11/2"  
data$time[data$date>=as.Date("2012-01-01") & data$date<=as.Date("2012-06-30")] = "12/1"  
data$time[data$date>=as.Date("2012-07-01") & data$date<=as.Date("2012-12-31")] = "12/2"  
data$time[data$date>=as.Date("2013-01-01") & data$date<=as.Date("2013-06-30")] = "13/1"  
data$time[data$date>=as.Date("2013-07-01") & data$date<=as.Date("2013-12-31")] = "13/2"  
data$time[data$date>=as.Date("2014-01-01") & data$date<=as.Date("2014-06-30")] = "14/1"  
data$time[data$date>=as.Date("2014-07-01") & data$date<=as.Date("2014-12-31")] = "14/2"  
data$time[data$date>=as.Date("2015-01-01") & data$date<=as.Date("2015-06-30")] = "15/1"  
data$time[data$date>=as.Date("2015-07-01") & data$date<=as.Date("2015-12-31")] = "15/2"  
data$time[data$date>=as.Date("2016-01-01") & data$date<=as.Date("2016-06-30")] = "16/1"  
data$time[data$date>=as.Date("2016-07-01") & data$date<=as.Date("2016-12-31")] = "16/2"  
data$time[data$date>=as.Date("2017-01-01") & data$date<=as.Date("2017-06-30")] = "17/1"  
data$time[data$date>=as.Date("2017-07-01") & data$date<=as.Date("2017-12-31")] = "17/2"  
data$time[data$date>=as.Date("2018-01-01") & data$date<=as.Date("2018-06-30")] = "18/1"  
data$time[data$date>=as.Date("2018-07-01") & data$date<=as.Date("2018-12-31")] = "18/2"  
data$time[data$date>=as.Date("2019-01-01") & data$date<=as.Date("2019-06-30")] = "19/1"  
data$time[data$date>=as.Date("2019-07-01") & data$date<=as.Date("2019-12-31")] = "19/2"  
  
data$time2 = data$time  
data$time2 = str\_replace(data$time2, "/", "\_")  
  
data$stage = 0  
data$stage[data$m\_questions==1]= 1  
data$stage[data$u\_questions==1]= 2  
data$stage[data$queen\_debate\_others==1]= 3  
data$stage[data$queen\_debate\_day1==1]= 4  
data$stage[data$pm\_questions==1]= 5

Inspecting data and selecting some parts of data for the research

# Packages  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.3 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::arrange() masks plyr::arrange()  
## ✖ dplyr::between() masks data.table::between()  
## ✖ dplyr::combine() masks gdata::combine()  
## ✖ purrr::compact() masks plyr::compact()  
## ✖ dplyr::count() masks plyr::count()  
## ✖ dplyr::desc() masks plyr::desc()  
## ✖ dplyr::failwith() masks plyr::failwith()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::first() masks data.table::first(), gdata::first()  
## ✖ lubridate::hour() masks data.table::hour()  
## ✖ dplyr::id() masks plyr::id()  
## ✖ lubridate::isoweek() masks data.table::isoweek()  
## ✖ purrr::keep() masks gdata::keep()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ dplyr::last() masks data.table::last(), gdata::last()  
## ✖ lubridate::mday() masks data.table::mday()  
## ✖ lubridate::minute() masks data.table::minute()  
## ✖ lubridate::month() masks data.table::month()  
## ✖ dplyr::mutate() masks plyr::mutate()  
## ✖ lubridate::quarter() masks data.table::quarter()  
## ✖ dplyr::rename() masks plyr::rename()  
## ✖ lubridate::second() masks data.table::second()  
## ✖ dplyr::starts\_with() masks tidyr::starts\_with(), gdata::starts\_with()  
## ✖ dplyr::summarise() masks plyr::summarise()  
## ✖ dplyr::summarize() masks plyr::summarize()  
## ✖ purrr::transpose() masks data.table::transpose()  
## ✖ lubridate::wday() masks data.table::wday()  
## ✖ lubridate::week() masks data.table::week()  
## ✖ lubridate::yday() masks data.table::yday()  
## ✖ lubridate::year() masks data.table::year()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(readr)  
library(tidytext)  
library(quanteda)

## Package version: 3.3.1  
## Unicode version: 14.0  
## ICU version: 71.1  
## Parallel computing: 8 of 8 threads used.  
## See https://quanteda.io for tutorials and examples.

library(textdata)

#Filtering data columns  
colnames(data)

## [1] "id\_speech" "id\_mp"   
## [3] "period" "last\_name"   
## [5] "first\_name" "date"   
## [7] "pm\_questions" "queen\_debate\_day1"   
## [9] "queen\_debate\_others" "m\_questions"   
## [11] "u\_questions" "other\_debate"   
## [13] "leader" "prime\_minister"   
## [15] "senior\_minister" "shadow"   
## [17] "cabinet" "chair"   
## [19] "government" "female"   
## [21] "age" "electoral\_cycle"   
## [23] "party" "linear\_trend"   
## [25] "words" "text"   
## [27] "emotive\_count" "neutral\_count"   
## [29] "emotive\_rhetoric" "emotive\_rhetoric\_log"   
## [31] "emotive\_words" "top\_topic"   
## [33] "anew\_rescaled" "emotive\_rhetoric\_liwc"   
## [35] "positive\_count" "negative\_count"   
## [37] "emotive\_positive" "emotive\_negative"   
## [39] "emotive\_count\_250\_8" "neutral\_count\_250\_8"   
## [41] "emotive\_rhetoric\_250\_8" "emotive\_count\_300\_10"   
## [43] "neutral\_count\_300\_10" "emotive\_rhetoric\_300\_10"  
## [45] "emotive\_count\_a1" "neutral\_count\_a1"   
## [47] "emotive\_rhetoric\_a1" "emotive\_count\_a2"   
## [49] "neutral\_count\_a2" "emotive\_rhetoric\_a2"   
## [51] "time" "time2"   
## [53] "stage"

data <- data %>%  
 select(last\_name, first\_name, date, female, age, party, text)  
head(data)

## last\_name first\_name date female age party  
## <char> <char> <Date> <int> <int> <char>  
## 1: dalyell tam 2001-06-13 0 69 Labour  
## 2: young george 2001-06-13 0 60 Conservative  
## 3: cook robin 2001-06-13 0 55 Labour  
## 4: hague william 2001-06-13 0 40 Conservative  
## 5: kennedy charles 2001-06-13 0 42 Liberal Democrats  
## 6: trimble david 2001-06-13 0 57 Ulster Unionist Party  
## text  
## <char>  
## 1: Under the provisions of Standing Order No 1A I am now required to ascertain whether Mr Michael J Martin is willing to be chosen as Speaker  
## 2: I beg to move That Mr Michael J Martin do take the Chair of this House as Speaker Mr Dalyell I begin by congratulating you on your accession as Father of the House Given your commitment to the House and your knowledge of how to use it effectively it gives all of us pleasure to see you supervising our proceedings today I too wish you a full and speedy recovery I also hope that you will continue to be as irreverent to those on your Front Bench as your predecessor as Father of the House was to mine As we meet this afternoon there are two important vacancies in our political institutions and I am happy to make a nomination for one of them In commending the right hon Member for Glasgow Springburn Mr Martin as Speaker I congratulate him on his re election to Parliament Although his election was contested I suspect that he did not experience the aggravation that the rest of us suffered in getting here I mention in passing that we in North West Hampshire saw a new form of co operation between the other two parties the Labour party ran a low key campaign to let the Liberal Democrats have a clear shot at me and the Liberal Democrats ran a low key campaign to give Labour a shot I am happy to say that my majority increased to more than 12 000 There are two reasons why I move the motion the first of which is continuity I do not believe that our Speaker should be put into play at the beginning of every Parliament I can do no better than quote the second report of the Select Committee on Procedure published in February In our view it would be undesirable in these circumstances for a multi candidate ballot to take place automatically As we have seen since the middle of the nineteenth century there has been a strong presumption that a Speaker once elected by the House is not subsequently challenged If it were to become accepted that a change in the composition of the House following a General Election were as a matter of course to lead to a change in the occupancy of the Chair we believe there are grave dangers that the office itself would be destabilised and in danger of becoming politicised Equally however we believe it is important that the House should not be denied the right to change its Speaker however unlikely it may be that that right will be exercised I agree with that sentiment as did the House when it agreed to change our procedures Although the Speaker should be validated there should be a presumption against challenging the incumbent Continuity is not the only reason If we were starting from scratch I believe that the House would choose the right hon Member for Springburn as its Speaker today His long service in the House and deep roots in the Back Benches his work on the Chairmen s Panel and domestic Committees his experience as Deputy Speaker his genial and approachable manner underpinned by a deep affection and commitment to the House all those qualities strike a chord with the House That commitment was confirmed in his acceptance speech last October and reinforced in the statement that we have just heard which was greatly welcomed All Speakers develop their own style and we saw the right hon Member for Springburn develop his in the previous Parliament a more approachable informal style of Speaker mixing with Members in the Tea Room and elsewhere no wig or silk stockings rather like the Scandinavian approach to the monarchy but so far without the bicycle A key role lies ahead for Mr Speaker in this Parliament There is a strong view that the House should reassert the accountability of the Executive to Parliament A growing number of hon Members want the House to be more relevant to the concerns of those whom we represent more effective in what it does and a better link between Government and governed The right hon Member for Springburn is well qualified to supervise and manage that debate during this Parliament I believe that he is entitled to support from the whole House in that role The House will understand why I was unable to propose the right hon Member for Springburn last time but I have no hesitation in proposing him today Question put and agreed to Resolved That Mr Michael J Martin do take the Chair of this House as Speaker Whereupon Mr Tam Dalyell left the Chair and Mr Michael J Martin was taken out of his place and conducted to the Chair by Sir George Young and Ann Keen  
## 3: Mr Speaker Elect it is a great pleasure to congratulate you on your return to the Chair of the House I think I speak for all Members when I say it is also a great pleasure for all of us that we have completed your election for a second time in rather shorter order than the six hours that it took us the first time May I add my congratulations to my neighbour my hon Friend the Member for Linlithgow Mr Dalyell on becoming Father of the House No hon Member has shown greater affection for the traditions of the House or carried out his duties as a Member with greater diligence than my hon Friend As a fellow West Lothian Member of Parliament perhaps I can explain on his behalf to the House that his injury is a tribute to the diligence with which he pursues his constituency office as it was incurred with an excess of enthusiasm at a meeting of the Linlithgow football team It therefore was an asset not a liability in the recent election Mr Speaker Elect there have been only 155 Speakers before you in the many centuries of your office When Speaker Yelverton in 1597 was asked to describe the necessary qualities of a Speaker he replied A man big and comely his carriage majestical his nature haughty and his purse plentiful Fortunately the characteristics of the Speaker have changed over the subsequent four centuries For myself I am relieved that we no longer expect our Speaker to be haughty just as you must be relieved that the House does not expect your purse to be plentiful Over the past seven months you have shown all the necessary qualities of a modern Speaker We could have expected no less from a Speaker who has brought to the Chair of the Chamber one of the longest records of service on the Chairmen s Panel You have been fair but firm you have turned aside confrontation with humour and you have got the better of those unwise enough to challenge your authority Many an awkward moment has been defused with your trademark catch phrase It s no nice Since you came to office your voice has become familiar in households across our country You spoke for all your Scottish compatriots in the Chamber when you magnificently brushed aside an impertinent question from the BBC on your accent with the retort I don t have an accent other people have an accent Yours is of course an accent which would normally lend authority to one of the traditional roles of the Speaker the selection of a Scotch whisky for the Speaker s brand For you though as a teetotaller that selection presented some obvious difficulties Your solution delighted the Members whom you invited on to an all party committee to carry out extensive research and tasting on your behalf Not always believing that we get the press that we deserve is of course a frequent experience of Members of this House even if the rest of the country thinks that we get the press that we richly deserve If ever in this Parliament the Lobby dares hint at criticism of our Speaker it can only be because as Chairman of the Administration Committee you banned Lobby correspondents from the Terrace of the Commons unless personally supervised You thereby liberated a grateful House to relax in privacy on the Terrace in these summer months Both you and I are now of that age when we share that puzzling sensation after each election that Members of Parliament seem to be getting younger than we remember To an even younger Chamber than before your commitment that families will be remembered in the proceedings of the House is particularly welcome That commitment comes of course from the affection and importance that you attach to your own family Those who know you well know also that no words of congratulations to you would be complete unless they also expressed appreciation of Mary your wife for her support to you in your role here and in your constituency Mr Speaker Elect on first taking the Chair you said that the Speaker had a clear duty to every side of the House especially to the Back Benchers It is because you have served the whole House with impartiality that the whole House has today returned you to the Chair unanimously Your colleagues and your friends congratulate you on it  
## 4: On behalf of all my right hon and hon Friends on the Opposition Benches I offer our sincerest congratulations to you Mr Speaker Elect on your re election Many colleagues on both sides of the House will share my relief that it was accompanied by less controversy and took dramatically less time than your election in October Your unanimous re election underlines the important constitutional fact that you are now very much Speaker of the whole of the House of Commons I add the Opposition s congratulations to those expressed to the Father of the House the hon Member for Linlithgow Mr Dalyell who as a highly active Back Bencher and champion of so many causes over decades is in every sense fully qualified for the post It is also a great pleasure to be the first to congratulate the right hon Member for Livingston Mr Cook on his appointment as Leader of the House of Commons We learned from the newspapers that apparently he has been unable to express all his real views as Foreign Secretary over the past four years My hon Friends are much looking forward to asking him for all his real views as Leader of the House during a succession of Question Times and statements over coming months On these occasions Mr Speaker Elect a great deal is usually made of the distinctive parliamentary existence of the holder of your great office that results from the need to cut yourself off from previous party affiliations You have indeed cut yourself off from party affiliations and as has been mentioned served this House impartially It has always been a tradition of Speakers that they do not visit the Tea Room and the bars as part of that procedure You have created a new tradition by being available and visiting the Tea Room and bars but not having a drink there That may keep you happy although it is incomprehensible to the rest of us but we are delighted that you have amended the traditions in that respect On a more serious point you are the custodian of the rules privileges and traditions of this House As the many newly elected Members will soon learn we all look to you as the independent champion of all parties in the House and of the rights of all hon Members As you know I have always made a particular point about protecting the rights of the Opposition Front Bench come to think of it though the rights of Back Benchers need protecting too especially those who have not spoken from the Back Benches for a long time I know that you will protect their rights in this Parliament There are also those occasions of high drama Mr Speaker Elect when votes are tied and it falls to you to use your casting vote although I admit that it is not immediately obvious that that will be necessary in a large number of instances You never know we will be working on it from this side of the House I hope that the fact that the Government have such a majority will not deter hon Members on both sides of the House from doing the job that they have been sent here to do to hold the Government to account Like so many past and present Members you and I care passionately about the House of Commons and the standing in which it is held throughout the country That is why I for one deeply regret the diminution of its importance and reputation which has accelerated although it did not begin in recent years I cannot be alone in thinking that that decline contributed in some ways to the disconnection between the public and Parliament that was highlighted by the lamentably low voter turnout in the general election last week Last October I expressed the hope that you would robustly resist all attempts to downgrade marginalise or bypass the House of Commons I repeat that today Few things would give me and I hope hon Members from all parties greater satisfaction in politics than to see this House restored to the centre of our national life The great issues that will come before us during the next few years some of which may be of supreme importance to the way in which our country is governed should be fully scrutinised and debated first and foremost in this Chamber On a personal note just as you are about to resume your duties I am planning to relinquish mine as Leader of the Opposition although I shall carry them out for a few weeks more I should like to thank you for the courtesy and kindness that you have shown me behind the scenes since you took the Chair of this House last October I am sure that your advice will continue to be of enormous benefit to hon Members throughout the House Again Mr Speaker Elect I congratulate you and wish you well for what undoubtedly promises to be an important and demanding period in our parliamentary history  
## 5: Mr Speaker Elect may I entirely associate my right hon and hon Friends with the personal and unanimous congratulations that the House has recorded on your re election as Speaker this afternoon We wish you all possible success for the forthcoming Parliament You will remember something about which we have joked privately since your first election At that time I recalled our first encounter when I was a schoolboy and you were on a picket line in your then capacity as a NUPE National Union of Public Employees regional official The unanimous endorsement of the House must mean that we have witnessed the ultimate triumph of NUPE man this afternoon We congratulate you on that I should like to take this opportunity also to congratulate the new Father of the House whose courtesy and advice to all of us especially those who have entered the House over the years as new Members has been unstinting and much appreciated It is a great happiness to see him occupy that position on behalf of us all but for me that happiness is tinged with one wee regret I shall dearly miss his distinguished predecessor Sir Edward Heath at Prime Minister s questions I do not have any ambition as Lib Dem leader to write a diary and publish it The book that I want to write and publish one day is the off the record commentary of Sir Edward Heath between 3 o clock and 3 30 every Wednesday afternoon on the parliamentary Conservative party and the questions that its members put to the Prime Minister of the day I shall miss the sensation when Sir Edward was notably upset or distressed by the viewpoints given by one of his own He had a tendency to expel a great sigh of frustration which tended to move me significantly along the Bench I shall miss those noises We must all share a sense of disappointment if not foreboding about the lack of engagement of so many of our fellow citizens in the election That being so I hope that early steps will be taken to examine our procedures and practices in this place to ensure that they are as relevant and comprehensible as possible to those outside There is a danger that all of us in all parties will be engulfed if we are not careful Given the balance of the outcome of the election I hope that you Mr Speaker Elect your senior officials and the new Leader of the House will give fresh consideration to the procedures of the House to ensure that they are made more flexible in some ways the House of Lords may provide instruction for us with regard to the rights of all parties especially those in opposition We look forward to contributing constructively to such discussions at the earliest opportunity In the meantime we wish you well Sir  
## 6: Mr Speaker Elect on my behalf and that of my colleagues I offer sincere congratulations on your being re elected as Speaker of this place We can vouch for your approachability the care with which you have listened to the issues that Members have brought to you and the way in which you have responded to them We look forward with confidence to that same care and consideration being offered in the months and years to come I congratulate the Father of the House on his conduct of today s proceedings It is not necessary for me to emphasise the way in which he has built a reputation over the years as a Back Bencher who has held Governments of all colours to account That reminds us that that is the prime function that many of us have in the House I am sure Mr Speaker Elect that with your consideration for Back Benchers and for the primary role of the House you will always be ready to assist us in ensuring that the Government are held to account It is commonplace these days to regret the decline of the standing of the House but we would all do well to remember that its standing depends upon each and every one of us in the contribution that we make to the House I am sure that we shall all be conscious of that responsibility However we rely upon you Mr Speaker Elect when it comes to issues where government may perhaps wish to cut corners or to do things elsewhere to insist upon the primacy of the House I shall make a cautionary comment on the remarks of the right hon Member for Ross Skye and Inverness West Mr Kennedy the leader of the Liberal Democrats We want to see the procedures of the House become more effective and clearly and easily understood However virtually every time that the reform of our procedures is broached the result is that the life of government is made easier That is not our objective

Defining research data by filtering research words

# Filtering data containing immigration related words  
# Define the keywords to search for  
immig\_words <- c('immigration', 'immigrant', 'asylum')  
visa\_words <- "\\b(UK)?visas?\\b"  
all\_words <- paste0(c(paste0(immig\_words, collapse = "|"), visa\_words), collapse = "|")  
  
# lower case text with keywords  
tidy\_data\_notoken <- data %>%  
 mutate(desc = tolower(text)) %>%  
 filter(grepl(all\_words, desc))  
  
# arrange the data by party and count the number of speech  
tidy\_data\_notoken %>%  
 group\_by(party) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count))

## # A tibble: 9 × 2  
## party count  
## <chr> <int>  
## 1 Conservative 7230  
## 2 Labour 6251  
## 3 Scottish National Party 1104  
## 4 Liberal Democrats 954  
## 5 Others 105  
## 6 Democratic Unionist Party 67  
## 7 Plaud Cymru 63  
## 8 Green 21  
## 9 Ulster Unionist Party 14

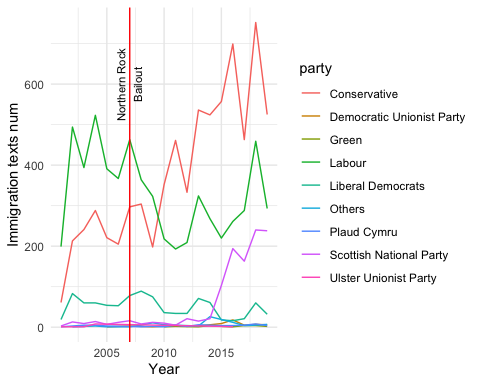
# 1. Word frequency of data

Making ‘Year’ column

# Make 'Year' using 'date'  
tidy\_data\_notoken$Year <- format(as.Date(tidy\_data\_notoken$date), "%Y")  
data$Year <- format(as.Date(data$date), "%Y")  
  
# Count the number of text by year and party  
text\_count <- tidy\_data\_notoken %>%   
 group\_by(Year, party) %>%   
 summarise(n = n(), .groups = 'drop')  
  
text\_count$Year <- as.numeric(as.character(text\_count$Year)) # Make 'Year' Column of numeric character.

A Simple descriptive plot - The number of immigration related texts by party

# Plotting counts of immigration texts by party  
ggplot(text\_count, aes(x = Year, y = n, group = party, color = party)) +  
 geom\_line() +  
 labs(y = "Immigration texts num", x = "Year") +  
 theme\_minimal() +  
 geom\_vline(xintercept = as.numeric(format(as.Date("2007-09-14"), "%Y")), col="red") +  
 annotate("text", x = as.numeric(format(as.Date("2007-09-14"), "%Y")), y = 600, label="Northern Rock\nBailout", angle=90, color = "black", size = 3)

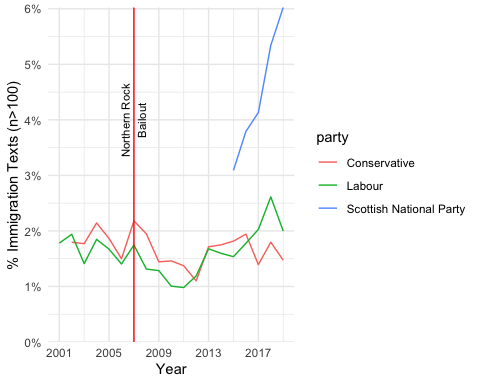


Dividing the number of immigration texts by the total number of texts

# Count the total number of texts in data by year and party  
data\_count <- data %>%  
 group\_by(Year, party) %>%  
 summarise(total\_n = n(), .groups = 'drop')  
  
# Merge the counts and calculate the ratio  
ratio\_count <- merge(text\_count, data\_count, by = c("Year", "party"))  
ratio\_count$ratio <- with(ratio\_count, n / total\_n)  
  
# Only counts more than 100 texts  
filtered\_ratio\_count <- ratio\_count %>%  
 filter(n > 100)  
  
filtered\_ratio\_count$Year <- as.numeric(as.character(filtered\_ratio\_count$Year))

A ratio plot of a number of immigration related texts by total texts

#Plotting the ratios of immigration related texts   
ggplot(filtered\_ratio\_count, aes(x = Year, y = ratio, group = party, color = party)) +  
 geom\_line() +  
 labs(y = "% Immigration Texts (n>100)", x = "Year") +  
 scale\_y\_continuous(labels = scales::percent\_format(), expand = c(0, 0), limits = c(0, NA)) +  
 scale\_x\_continuous(breaks = seq(min(filtered\_ratio\_count$Year), max(filtered\_ratio\_count$Year), by = 4)) +  
 theme\_minimal() +  
 geom\_vline(xintercept = as.numeric(format(as.Date("2007-09-14"), "%Y")), col="red") +  
 annotate("text", x = as.numeric(format(as.Date("2007-09-14"), "%Y")), y = 0.04, label="Northern Rock\nBailout", angle=90, color = "black", size = 3)



Tokenisation and processing stop words

# Tokenisation & removing stop words  
tidy\_data <- tidy\_data\_notoken %>%  
 unnest\_tokens(word, desc) %>%  
 filter(str\_detect(word, "[a-z]")) %>%  
 filter(!word %in% stop\_words$word)  
  
tidy\_data <- tidy\_data %>%  
 arrange(date)  
tidy\_data$order <- 1:nrow(tidy\_data) # Make orders of each word

# Common tokens  
word\_count <- tidy\_data %>%  
 count(word, sort = T)  
  
show(word\_count)

## word n  
## <char> <int>  
## 1: people 27644  
## 2: hon 26615  
## 3: government 26551  
## 4: immigration 19105  
## 5: minister 14229  
## ---   
## 39501: énarques 1  
## 39502: ørsted 1  
## 39503: þæt 1  
## 39504: šefcovic 1  
## 39505: štefan 1

# Common tokens by year  
word\_count\_year <- tidy\_data %>%  
 group\_by(Year) %>%  
 count(word, sort = T)  
  
show(word\_count\_year)

## # A tibble: 226,043 × 3  
## # Groups: Year [19]  
## Year word n  
## <chr> <chr> <int>  
## 1 2018 government 2122  
## 2 2018 people 2060  
## 3 2015 people 2058  
## 4 2018 immigration 2046  
## 5 2004 government 2007  
## 6 2002 people 1883  
## 7 2018 hon 1851  
## 8 2004 hon 1826  
## 9 2002 hon 1792  
## 10 2007 hon 1792  
## # ℹ 226,033 more rows

# Comon tokens by party  
word\_count\_party <- tidy\_data %>%  
 group\_by(party) %>%  
 count(word, sort = T)  
  
show(word\_count\_party)

## # A tibble: 98,986 × 3  
## # Groups: party [9]  
## party word n  
## <chr> <chr> <int>  
## 1 Labour people 12163  
## 2 Conservative government 12117  
## 3 Labour hon 12010  
## 4 Conservative hon 11848  
## 5 Conservative people 11229  
## 6 Labour government 9743  
## 7 Conservative immigration 9001  
## 8 Labour immigration 7336  
## 9 Conservative minister 6290  
## 10 Conservative country 6258  
## # ℹ 98,976 more rows

# 2. Sentiment Analysis Using ‘NRC’ Sentiment Dictionary

## 1)Total sentiment of immigration related texts

Getting NRC sentiment dictionary and tidying data by calculating sentiment ratio

get\_sentiments("nrc")

## # A tibble: 13,872 × 2  
## word sentiment  
## <chr> <chr>   
## 1 abacus trust   
## 2 abandon fear   
## 3 abandon negative   
## 4 abandon sadness   
## 5 abandoned anger   
## 6 abandoned fear   
## 7 abandoned negative   
## 8 abandoned sadness   
## 9 abandonment anger   
## 10 abandonment fear   
## # ℹ 13,862 more rows

# Make nrc sentiment tables of data  
nrc\_data <- tidy\_data %>%  
 inner\_join(get\_sentiments("nrc"), by = "word") %>%  
 count(date, sentiment) %>%  
 spread(key = sentiment, value = n, fill = 0) %>%  
 mutate(ratio = negative / (positive+1)) # Calculating negative/positive ratio. Adding 1 to avoid the 0 denominator

## Warning in inner\_join(., get\_sentiments("nrc"), by = "word"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 6 of `x` matches multiple rows in `y`.  
## ℹ Row 5657 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

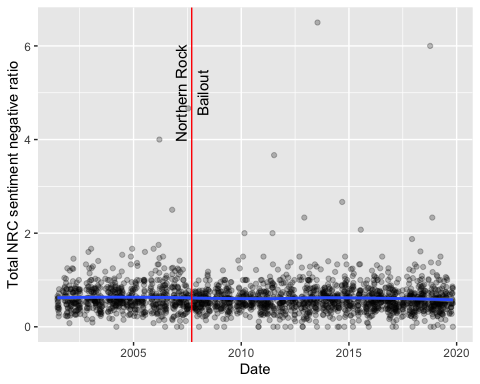
# Check sentiments  
colnames(nrc\_data)

## [1] "date" "anger" "anticipation" "disgust" "fear"   
## [6] "joy" "negative" "positive" "sadness" "surprise"   
## [11] "trust" "ratio"

Plotting the total negative ratio

# Total sentiment of text including immigration related words plot (Using negative ratio)  
nrc\_data %>%  
 ggplot(aes(date, ratio)) +  
 geom\_point(alpha=0.25) +  
 geom\_smooth(method="loess", alpha=0.5) +  
 labs(y = "Total NRC sentiment negative ratio", x = "Date") +  
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col="red") +  
 annotate("text", x = as.Date("2007-09-24"), y = 5, label="Northern Rock\nBailout", angle=90, color = "black", size = 4)

## `geom\_smooth()` using formula = 'y ~ x'

 There is no time when particularly negative emotions are revealed. Rather, negative feelings toward immigrants are less visible after the financial crisis.

## 2) Negative ratio plots by party

Calculating NRC sentiment negative ratios of texts which contain immigration related words *by party*

# Arranging data by party and calculate the ratio of negative sentiment.  
nrc\_data\_party <- tidy\_data %>%  
 inner\_join(get\_sentiments("nrc"), by = "word") %>%  
 group\_by(party, date) %>%  
 count(sentiment) %>%  
 spread(key = sentiment, value = n, fill = 0) %>%  
 mutate(ratio = negative/(positive+1)) %>%  
 ungroup()

## Warning in inner\_join(., get\_sentiments("nrc"), by = "word"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 6 of `x` matches multiple rows in `y`.  
## ℹ Row 5657 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

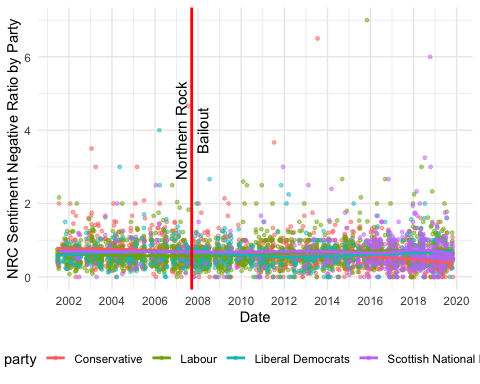
filtered\_nrc\_data\_party <-nrc\_data\_party %>%  
 group\_by(party) %>%  
 filter(n() > 100) %>% # party with more than 100 related words  
 ungroup()

Plotting a negative ratio plot of party. Points indicate the actual scores while smooth lines show the trends.

# A negative sentiment ratio plot of four parties  
filtered\_nrc\_data\_party %>%  
 ggplot(aes(date, ratio, color = party)) +  
 geom\_point(alpha=0.5, size = 1) +  
 geom\_smooth(method="loess", se = F, alpha=0.7, size = 1) +  
 scale\_x\_date(date\_breaks = "2 years", date\_labels = "%Y") +  
 labs(y = "NRC Sentiment Negative Ratio by Party", x = "Date") +  
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col="red", size = 1) +  
 annotate("text", x = as.Date("2007-09-24"), y = 4, label="Northern Rock\nBailout", angle=90, color = "black", size = 4) +  
 theme\_minimal() +  
 theme(legend.position = "bottom")

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

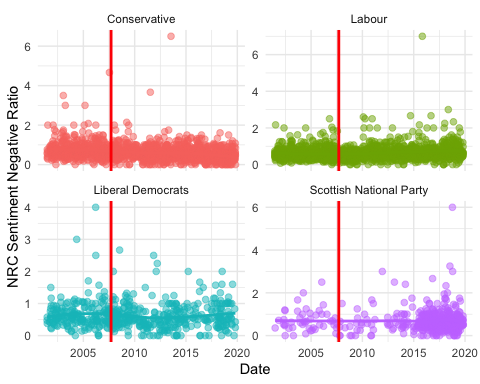
## `geom\_smooth()` using formula = 'y ~ x'

 Similar to the overall sentiment plot, there is no period when the proportion of negative words appears prominently.

Make the plot above readable, dividing plots by party

# Faceting by party  
filtered\_nrc\_data\_party %>%  
 ggplot(aes(x = date, y = ratio, color = party)) +  
 geom\_point(alpha = 0.5, size = 2) +  
 geom\_smooth(method = "loess", se = FALSE, alpha = 0.7, size = 1) +  
 labs(y = "NRC Sentiment Negative Ratio", x = "Date") +  
 facet\_wrap(~party, scales = "free\_y") +   
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col = "red", size = 1) +  
 theme\_minimal() +  
 theme(legend.position = "none")

## `geom\_smooth()` using formula = 'y ~ x'

 Similar to the overall sentiment plot, there is no period when the proportion of negative words appears prominently.

## 3) Sentiments counts by party

What is the most common sentiment? Summing up scores of sentiments by party to compare.

# Total sentiment word counts by party  
dominant\_senti\_nrc\_party <-filtered\_nrc\_data\_party %>%  
 group\_by(party) %>%  
 summarise(across(c(anger, anticipation, disgust, fear, joy, sadness, surprise, trust, negative, positive), sum, na.rm = T))

## Warning: There was 1 warning in `summarise()`.  
## ℹ In argument: `across(...)`.  
## ℹ In group 1: `party = "Conservative"`.  
## Caused by warning:  
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.  
## Supply arguments directly to `.fns` through an anonymous function instead.  
##   
## # Previously  
## across(a:b, mean, na.rm = TRUE)  
##   
## # Now  
## across(a:b, \(x) mean(x, na.rm = TRUE))

show(dominant\_senti\_nrc\_party)

## # A tibble: 4 × 11  
## party anger anticipation disgust fear joy sadness surprise trust negative  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Conser… 33439 52127 17861 56500 36474 32322 17703 99258 84545  
## 2 Labour 35105 53846 18420 57405 38456 34318 18408 97785 85101  
## 3 Libera… 5625 9303 2963 9620 5875 5778 3149 15899 14675  
## 4 Scotti… 4637 7398 2523 8180 4986 5388 2638 12649 12877  
## # ℹ 1 more variable: positive <dbl>

The large number of sentiment counts is generated by the Conservatives and Labor parties.

## 4) Sentiment changes by party

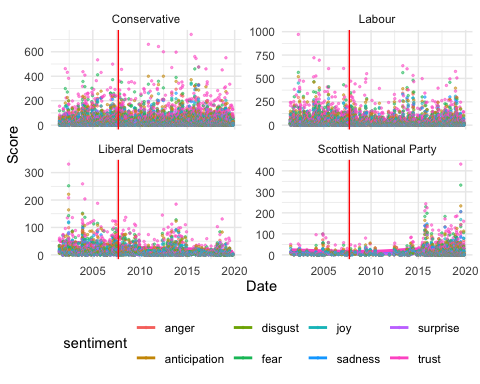
Beyond negative/positive ratio, focusing on specific sentiments trends by party.

# Combining all sentiments into one column  
long\_nrc\_data <- filtered\_nrc\_data\_party %>%  
 pivot\_longer(cols = c(anger, fear, trust, sadness, disgust, anticipation, surprise, joy), names\_to = "sentiment", values\_to = "score")

Faceting plots to compare the sentiment changes of each party

# Plotting specific sentiment changes by party  
long\_nrc\_data %>%  
 ggplot(aes(x = date, y = score, color = sentiment)) +  
 geom\_smooth(method = "loess", se = F, alpha = 1, size = 1) +  
 geom\_point(alpha = 0.5, size = 0.5) +   
 facet\_wrap(~ party, scales = "free\_y") +  
 labs(x = "Date", y = "Score") +  
 theme\_minimal() +  
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col = "red") +  
 theme\_minimal() +  
 theme(legend.position = "bottom")

## `geom\_smooth()` using formula = 'y ~ x'

 The smooth line in each plot shows a relatively steady and unchanged shape.

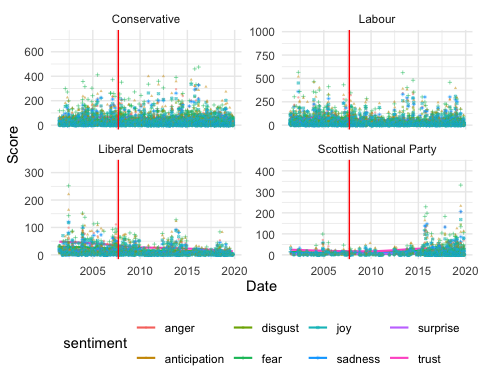
To make sentiment points recognizable, adjusting the shapes of points by sentiment and plotting

# Shaping sentiment points differently  
shape\_mapping <- c("anger" = 17, "disgust" = 18, "fear" = 19, "sadness" = 8, "trust" = 21, "anticipation" = 22, "surprise" = 23, "joy" = 24)  
  
long\_nrc\_data %>%  
 ggplot(aes(x = date, y = score, color = sentiment)) +  
 geom\_smooth(method = "loess", se = F, alpha = 0.75, size = 0.75) +   
 geom\_point(aes(shape = sentiment), alpha = 0.5, size = 0.5) +   
 facet\_wrap(~ party, scales = "free\_y", nrow = 2) +   
 labs(x = "Date", y = "Score") +  
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col = "red") +  
 theme\_minimal() +  
 theme(legend.position = "bottom")

## `geom\_smooth()` using formula = 'y ~ x'

## Warning: The shape palette can deal with a maximum of 6 discrete values because more  
## than 6 becomes difficult to discriminate  
## ℹ you have requested 8 values. Consider specifying shapes manually if you need  
## that many have them.

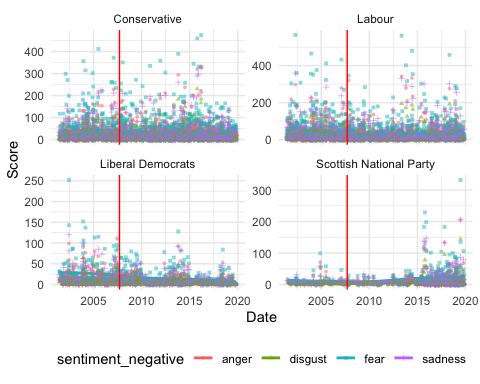
## Warning: Removed 8132 rows containing missing values or values outside the scale range  
## (`geom\_point()`).



Selecting negative sentiments to confirm sentiment changes

# Specifying negative feelings only  
long\_nrc\_data\_neg <- filtered\_nrc\_data\_party %>%  
 pivot\_longer(cols = c(anger, fear, sadness, disgust), names\_to = "sentiment\_negative", values\_to = "score")  
  
# Plots of specific negative sentiments by party   
long\_nrc\_data\_neg %>%  
 ggplot(aes(x = date, y = score, color = sentiment\_negative)) +  
 geom\_smooth(method = "loess", se = F, alpha = 0.75, size = 1) +   
 geom\_point(aes(shape = sentiment\_negative), alpha = 0.5, size = 1) +   
 facet\_wrap(~ party, scales = "free\_y", nrow = 2) +   
 labs(x = "Date", y = "Score") +  
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col = "red") +  
 theme\_minimal() +  
 theme(legend.position = "bottom")

## `geom\_smooth()` using formula = 'y ~ x'

 Regarding negative sentiments, the Conservative party presents more diverse and fluctuated points than other parties. However, the general smooth lines stay steady.

## 5) Random sample sentiment

To compare with sentiment trends of immigration related text, make a random sample of 10000 and do sentiment analysis

# Random sampling  
data\_sample <- data %>%  
 sample\_n(10000)  
  
# tidy sample  
tidy\_samp <- data\_sample %>%  
 mutate(desc = tolower(text)) %>%  
 unnest\_tokens(word, desc) %>%  
 filter(str\_detect(word, "[a-z]")) %>%  
 filter(!word %in% stop\_words$word) %>%  
 arrange(date)  
  
tidy\_samp$order <- 1:nrow(tidy\_samp)  
  
# Applying NRC dictionary and calculating total negative ratio  
samp\_nrc\_data <- tidy\_samp %>%  
 inner\_join(get\_sentiments("nrc"), by = "word") %>%  
 count(date, sentiment) %>%  
 spread(key = sentiment, value = n, fill = 0) %>%  
 mutate(ratio = negative / (positive+1))

## Warning in inner\_join(., get\_sentiments("nrc"), by = "word"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 2 of `x` matches multiple rows in `y`.  
## ℹ Row 5503 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

#Applying NRC dictionary and calculating negative ratio by party  
samp\_nrc\_party <- tidy\_samp %>%  
 inner\_join(get\_sentiments("nrc"), by = "word") %>%  
 group\_by(party, date) %>%  
 count(sentiment) %>%  
 spread(key = sentiment, value = n, fill = 0) %>%  
 mutate(ratio = negative / (positive+1)) %>%  
 ungroup()

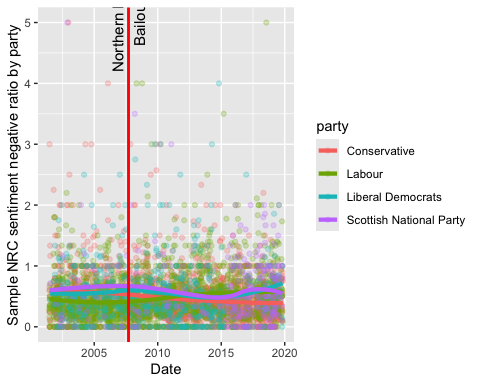
## Warning in inner\_join(., get\_sentiments("nrc"), by = "word"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 2 of `x` matches multiple rows in `y`.  
## ℹ Row 5503 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

filtered\_samp\_nrc\_party <- samp\_nrc\_party %>%  
 group\_by(party) %>%  
 filter(n() > 100) %>%  
 ungroup()

A plot of sample text negative sentiment ratio by party

filtered\_samp\_nrc\_party %>%  
 ggplot(aes(date, ratio, color = party)) +  
 geom\_point(alpha=0.25) +  
 geom\_smooth(method="loess", se = F, alpha=0.25, size = 1.5) +  
 labs(y = "Sample NRC sentiment negative ratio by party", x = "Date") +  
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col = "red", size = 1) +   
 annotate("text", x = as.Date("2007-09-24"), y = 5, label="Northern Rock\nBailout", angle=90, color = "black", size = 4)

## `geom\_smooth()` using formula = 'y ~ x'



theme\_minimal() +  
 theme(legend.position = "bottom")

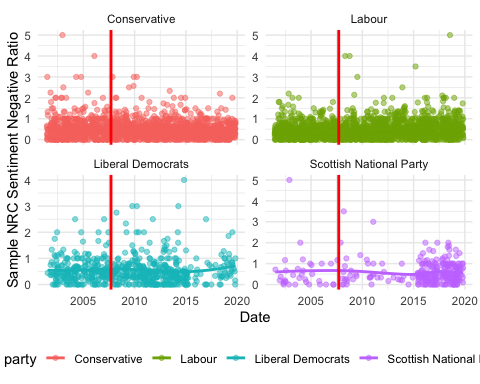
## List of 136  
## $ line :List of 6  
## ..$ colour : chr "black"  
## ..$ linewidth : num 0.5  
## ..$ linetype : num 1  
## ..$ lineend : chr "butt"  
## ..$ arrow : logi FALSE  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_line" "element"  
## $ rect :List of 5  
## ..$ fill : chr "white"  
## ..$ colour : chr "black"  
## ..$ linewidth : num 0.5  
## ..$ linetype : num 1  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_rect" "element"  
## $ text :List of 11  
## ..$ family : chr ""  
## ..$ face : chr "plain"  
## ..$ colour : chr "black"  
## ..$ size : num 11  
## ..$ hjust : num 0.5  
## ..$ vjust : num 0.5  
## ..$ angle : num 0  
## ..$ lineheight : num 0.9  
## ..$ margin : 'margin' num [1:4] 0points 0points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : logi FALSE  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ title : NULL  
## $ aspect.ratio : NULL  
## $ axis.title : NULL  
## $ axis.title.x :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 2.75points 0points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.title.x.top :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 0  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 2.75points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.title.x.bottom : NULL  
## $ axis.title.y :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : num 90  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 2.75points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.title.y.left : NULL  
## $ axis.title.y.right :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : num -90  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 0points 2.75points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : chr "grey30"  
## ..$ size : 'rel' num 0.8  
## ..$ hjust : NULL  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : NULL  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.x :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 1  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 2.2points 0points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.x.top :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : NULL  
## ..$ vjust : num 0  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 2.2points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.x.bottom : NULL  
## $ axis.text.y :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 1  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 2.2points 0points 0points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.y.left : NULL  
## $ axis.text.y.right :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 0  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 0points 0points 2.2points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.text.theta : NULL  
## $ axis.text.r :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 0.5  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : 'margin' num [1:4] 0points 2.2points 0points 2.2points  
## .. ..- attr(\*, "unit")= int 8  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ axis.ticks : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ axis.ticks.x : NULL  
## $ axis.ticks.x.top : NULL  
## $ axis.ticks.x.bottom : NULL  
## $ axis.ticks.y : NULL  
## $ axis.ticks.y.left : NULL  
## $ axis.ticks.y.right : NULL  
## $ axis.ticks.theta : NULL  
## $ axis.ticks.r : NULL  
## $ axis.minor.ticks.x.top : NULL  
## $ axis.minor.ticks.x.bottom : NULL  
## $ axis.minor.ticks.y.left : NULL  
## $ axis.minor.ticks.y.right : NULL  
## $ axis.minor.ticks.theta : NULL  
## $ axis.minor.ticks.r : NULL  
## $ axis.ticks.length : 'simpleUnit' num 2.75points  
## ..- attr(\*, "unit")= int 8  
## $ axis.ticks.length.x : NULL  
## $ axis.ticks.length.x.top : NULL  
## $ axis.ticks.length.x.bottom : NULL  
## $ axis.ticks.length.y : NULL  
## $ axis.ticks.length.y.left : NULL  
## $ axis.ticks.length.y.right : NULL  
## $ axis.ticks.length.theta : NULL  
## $ axis.ticks.length.r : NULL  
## $ axis.minor.ticks.length : 'rel' num 0.75  
## $ axis.minor.ticks.length.x : NULL  
## $ axis.minor.ticks.length.x.top : NULL  
## $ axis.minor.ticks.length.x.bottom: NULL  
## $ axis.minor.ticks.length.y : NULL  
## $ axis.minor.ticks.length.y.left : NULL  
## $ axis.minor.ticks.length.y.right : NULL  
## $ axis.minor.ticks.length.theta : NULL  
## $ axis.minor.ticks.length.r : NULL  
## $ axis.line : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ axis.line.x : NULL  
## $ axis.line.x.top : NULL  
## $ axis.line.x.bottom : NULL  
## $ axis.line.y : NULL  
## $ axis.line.y.left : NULL  
## $ axis.line.y.right : NULL  
## $ axis.line.theta : NULL  
## $ axis.line.r : NULL  
## $ legend.background : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ legend.margin : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points  
## ..- attr(\*, "unit")= int 8  
## $ legend.spacing : 'simpleUnit' num 11points  
## ..- attr(\*, "unit")= int 8  
## $ legend.spacing.x : NULL  
## $ legend.spacing.y : NULL  
## $ legend.key : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ legend.key.size : 'simpleUnit' num 1.2lines  
## ..- attr(\*, "unit")= int 3  
## $ legend.key.height : NULL  
## $ legend.key.width : NULL  
## $ legend.key.spacing : 'simpleUnit' num 5.5points  
## ..- attr(\*, "unit")= int 8  
## $ legend.key.spacing.x : NULL  
## $ legend.key.spacing.y : NULL  
## $ legend.frame : NULL  
## $ legend.ticks : NULL  
## $ legend.ticks.length : 'rel' num 0.2  
## $ legend.axis.line : NULL  
## $ legend.text :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : 'rel' num 0.8  
## ..$ hjust : NULL  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : NULL  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ legend.text.position : NULL  
## $ legend.title :List of 11  
## ..$ family : NULL  
## ..$ face : NULL  
## ..$ colour : NULL  
## ..$ size : NULL  
## ..$ hjust : num 0  
## ..$ vjust : NULL  
## ..$ angle : NULL  
## ..$ lineheight : NULL  
## ..$ margin : NULL  
## ..$ debug : NULL  
## ..$ inherit.blank: logi TRUE  
## ..- attr(\*, "class")= chr [1:2] "element\_text" "element"  
## $ legend.title.position : NULL  
## $ legend.position : chr "bottom"  
## $ legend.position.inside : NULL  
## $ legend.direction : NULL  
## $ legend.byrow : NULL  
## $ legend.justification : chr "center"  
## $ legend.justification.top : NULL  
## $ legend.justification.bottom : NULL  
## $ legend.justification.left : NULL  
## $ legend.justification.right : NULL  
## $ legend.justification.inside : NULL  
## $ legend.location : NULL  
## $ legend.box : NULL  
## $ legend.box.just : NULL  
## $ legend.box.margin : 'margin' num [1:4] 0cm 0cm 0cm 0cm  
## ..- attr(\*, "unit")= int 1  
## $ legend.box.background : list()  
## ..- attr(\*, "class")= chr [1:2] "element\_blank" "element"  
## $ legend.box.spacing : 'simpleUnit' num 11points  
## ..- attr(\*, "unit")= int 8  
## [list output truncated]  
## - attr(\*, "class")= chr [1:2] "theme" "gg"  
## - attr(\*, "complete")= logi TRUE  
## - attr(\*, "validate")= logi TRUE

Negative sentiments of sample texts also show steady lines.

Dividing the sample sentiment plot by party

# Faceting plots by party  
filtered\_samp\_nrc\_party %>%  
 ggplot(aes(date, ratio, color = party)) +  
 geom\_point(alpha = 0.5) +  
 geom\_smooth(method = "loess", se = F, alpha = 0.7, size = 1) +  
 labs(y = "Sample NRC Sentiment Negative Ratio", x = "Date") +  
 facet\_wrap(~party, scales = "free\_y") +   
 geom\_vline(xintercept = as.numeric(as.Date("2007-09-14")), col = "red", size = 1) +  
 theme\_minimal() +  
 theme(legend.position = "bottom")

## `geom\_smooth()` using formula = 'y ~ x'

 Compared to the immigration related text sentiment plots, sample sentiment plots show no significant difference.

# 3. Word Embedding GloVe

# Set up  
library(text2vec)   
library(stringr)  
library(umap)  
library(ggrepel)

## 1) Training total immigration related texts

### (1) The GloVe model for total parliament speech

Training process

# choice parameters  
WINDOW\_SIZE <- 6 # context up to 6 words  
DIM <- 300 # the length of the word vector  
ITERS <- 100 # the maximum number of iterations  
COUNT\_MIN <- 10 # minimum count of words  
  
# shuffle text  
set.seed(42L) # for reproducibility  
glove\_text <- sample(tidy\_data\_notoken$desc)  
  
# create vocab  
tokens <- space\_tokenizer(glove\_text) # tokenizing  
it <- itoken(tokens, progressbar = FALSE) # create the vocabulary object  
vocab <- create\_vocabulary(it) # create a vocabulary  
vocab\_pruned <- prune\_vocabulary(vocab, term\_count\_min = COUNT\_MIN) # keep only words that meet count threshold  
  
#vectorize and create term co-occurrence matrix  
vectorizer <- vocab\_vectorizer(vocab\_pruned)  
tcm <- create\_tcm(it, vectorizer, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric",   
 weights = rep(1, WINDOW\_SIZE))

Setting model parameters

#set model parameters  
glove <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
  
# fit model  
word\_vectors\_main <- glove$fit\_transform(tcm, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
# get output   
word\_vectors\_context <- glove$components  
glove\_embedding <- word\_vectors\_main + t(word\_vectors\_context) # word vectors. combine main and context word vectors  
  
#save   
saveRDS(glove\_embedding, file = "local\_glove.rds")

Modelling takes time. So I will use the prepared model made by same process for knitting.

url <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove.rds?raw=true"  
glove\_embedding <- readRDS(url(url, method = "libcurl"))

Visulalization of GloVe word embedding model by using umap.

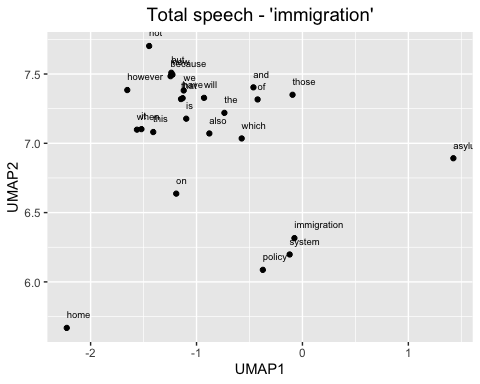
# GloVe dimension reduction (two dimension)  
glove\_umap <- umap(glove\_embedding, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread=2)  
  
# Put results in a dataframe for ggplot  
df\_glove\_umap <- as.data.frame(glove\_umap[["layout"]])  
  
# Add the labels of the words to the dataframe  
df\_glove\_umap$word <- rownames(df\_glove\_umap)  
colnames(df\_glove\_umap) <- c("UMAP1", "UMAP2", "word")

Plot the total word embedding of ‘immigration’ with the GloVe model generated above.

word <- glove\_embedding["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding, y = word, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words <- df\_glove\_umap %>%   
 inner\_join(y=select, by= "word")

The ggplot visual for GloVe regarding total immigration related texts. Total parliament speech word embedding of words related to *immigration*

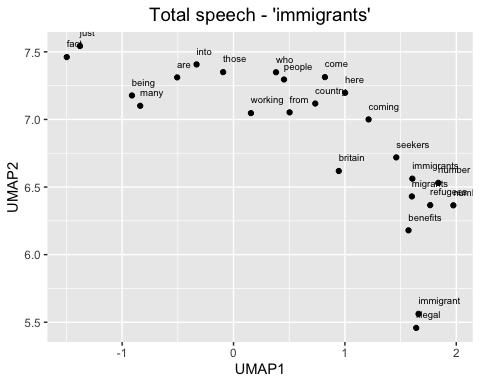
ggplot(selected\_words, aes(x = UMAP1, y = UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(UMAP1, UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Total speech - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))

 I did it for confirming the modeling process, but we might need to use this map as a comparison plot.

Total parliament speech word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (Case2: immigrants)  
word\_2 <- glove\_embedding["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding, y = word\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_2 <- df\_glove\_umap %>%   
 inner\_join(y=select, by= "word")

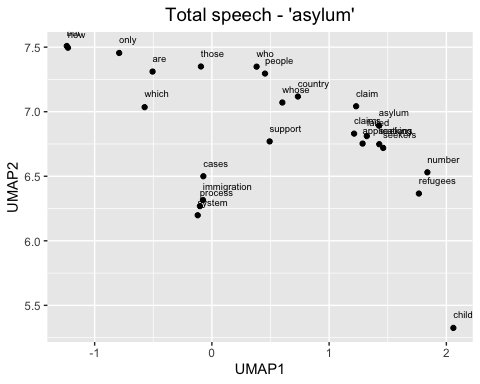
#The ggplot visual for GloVe  
ggplot(selected\_words\_2, aes(x = UMAP1, y = UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(UMAP1, UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Total speech - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))

 This can be also used for comparison. Also, word embedding maps tell us the general relationships between words, especially those we are interested in.

For the last pilot modelling map, there is a total parliament immigration related speech word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (asylum)  
word\_4 <- glove\_embedding["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding, y = word\_4, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_4 <- df\_glove\_umap %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_4, aes(x = UMAP1, y = UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(UMAP1, UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Total speech - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



### (2) The Glove model for total Parliament speech before the 2008 economic crisis

# filter data by date. Now have pre/post economic crisis text data related to immigration.  
  
data\_pre <- filter(tidy\_data\_notoken, date < as.Date("2007-09-24"))  
data\_post <- filter(tidy\_data\_notoken, date >= as.Date("2007-09-24"))

# repeat the same modeling process  
set.seed(42L)  
glove\_text\_pre <- sample(data\_pre$desc)  
  
tokens\_pre <- space\_tokenizer(glove\_text\_pre)  
it\_pre <- itoken(tokens\_pre, progressbar = FALSE)  
vocab\_pre <- create\_vocabulary(it\_pre)  
vocab\_pruned\_pre <- prune\_vocabulary(vocab\_pre, term\_count\_min = COUNT\_MIN)  
  
vectorizer\_pre <- vocab\_vectorizer(vocab\_pruned\_pre)  
tcm\_pre <- create\_tcm(it\_pre, vectorizer\_pre, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric", weights = rep(1, WINDOW\_SIZE))  
  
glove\_pre <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
word\_vectors\_main\_pre <- glove\_pre$fit\_transform(tcm\_pre, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
word\_vectors\_context\_pre <- glove\_pre$components  
glove\_embedding\_pre <- word\_vectors\_main\_pre + t(word\_vectors\_context\_pre)  
  
saveRDS(glove\_embedding\_pre, file = "local\_glove\_pre.rds")

Modelling takes time. So I will use the prepared model made by same process for knitting.

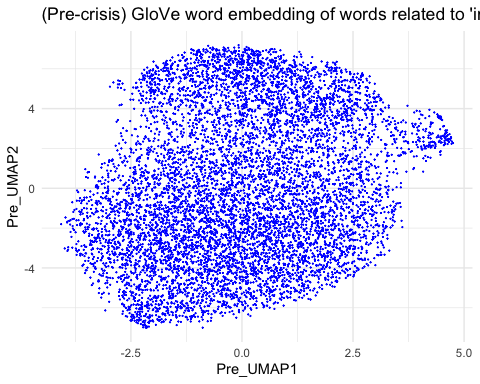
url\_pre <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove\_pre.rds?raw=true"  
glove\_embedding\_pre <- readRDS(url(url\_pre, method = "libcurl"))

Pre-crisis GloVe word embedding model of total parliament texts by two dimensional umap

# Plotting the whole word embedding of pre-crisis immigration related text  
umap\_pre <- umap(glove\_embedding\_pre, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread = 2)  
  
df\_umap\_pre <- as.data.frame(umap\_pre[["layout"]])  
df\_umap\_pre$word <- rownames(df\_umap\_pre)  
colnames(df\_umap\_pre) <- c("Pre\_UMAP1", "Pre\_UMAP2", "word")

Let’s see the whole word embedding map of two dimensions

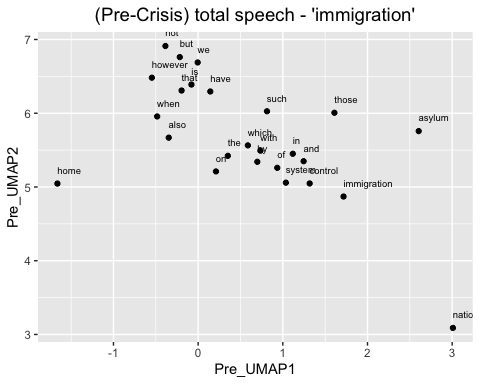
ggplot(df\_umap\_pre) +  
 geom\_point(aes(x = Pre\_UMAP1, y = Pre\_UMAP2), color = 'blue', size = 0.05) +  
 labs(title = "(Pre-crisis) GloVe word embedding of words related to 'immigration'") +  
 theme\_minimal()



Specify the map. Which words are close to the word ‘immigration’ in the total parliament texts? Pre-crisis total parliament speech word embedding of words related to *immigration*

# Plot the word embedding of words that are related for the GloVe model (Case1: immigration)  
word\_pre\_1 <- glove\_embedding\_pre["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_pre, y = word\_pre\_1, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_pre\_1 <- df\_umap\_pre %>%   
 inner\_join(y=select, by= "word")

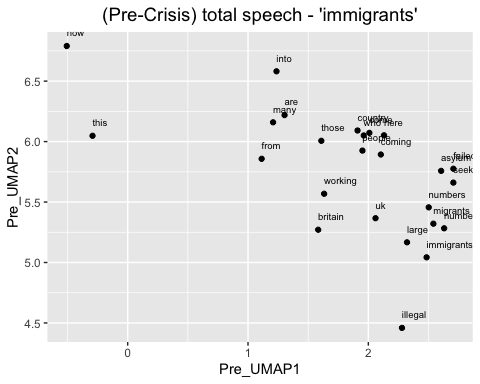
#The ggplot visual for GloVe  
ggplot(selected\_words\_pre\_1, aes(x = Pre\_UMAP1, y = Pre\_UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_UMAP1, Pre\_UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "(Pre-Crisis) total speech - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis total parliament speech word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (Case2: immigrants)  
word\_pre\_2 <- glove\_embedding\_pre["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_pre, y = word\_pre\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_pre\_2 <- df\_umap\_pre %>%   
 inner\_join(y=select, by= "word")

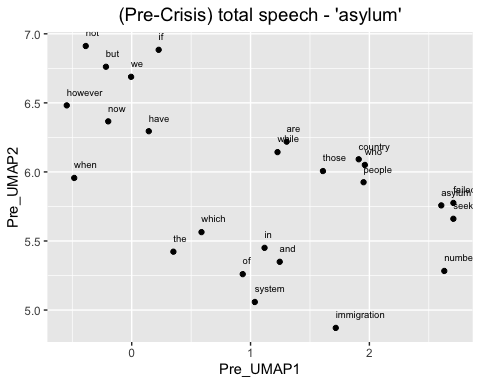
#The ggplot visual for GloVe  
ggplot(selected\_words\_pre\_2, aes(x = Pre\_UMAP1, y = Pre\_UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_UMAP1, Pre\_UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "(Pre-Crisis) total speech - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis total parliament speech word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (asylum)  
word\_pre\_3 <- glove\_embedding\_pre["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_pre, y = word\_pre\_3, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_pre\_3 <- df\_umap\_pre %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_pre\_3, aes(x = Pre\_UMAP1, y = Pre\_UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_UMAP1, Pre\_UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "(Pre-Crisis) total speech - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Now we can check the words relationships before the 2008 financail crisis.

### (3) The Glove model for total Parliament speech after the 2008 financial crisis

Again, same modeling process

set.seed(42L)  
glove\_text\_post <- sample(data\_post$desc)  
  
tokens\_post <- space\_tokenizer(glove\_text\_post)  
it\_post <- itoken(tokens\_post, progressbar = FALSE)  
vocab\_post <- create\_vocabulary(it\_post)  
vocab\_pruned\_post <- prune\_vocabulary(vocab\_post, term\_count\_min = COUNT\_MIN)  
  
vectorizer\_post <- vocab\_vectorizer(vocab\_pruned\_post)  
tcm\_post <- create\_tcm(it\_post, vectorizer\_post, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric", weights = rep(1, WINDOW\_SIZE))  
  
glove\_post <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
word\_vectors\_main\_post <- glove\_post$fit\_transform(tcm\_post, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
word\_vectors\_context\_post <- glove\_post$components  
glove\_embedding\_post <- word\_vectors\_main\_post + t(word\_vectors\_context\_post)  
  
saveRDS(glove\_embedding\_post, file = "local\_glove\_post.rds")

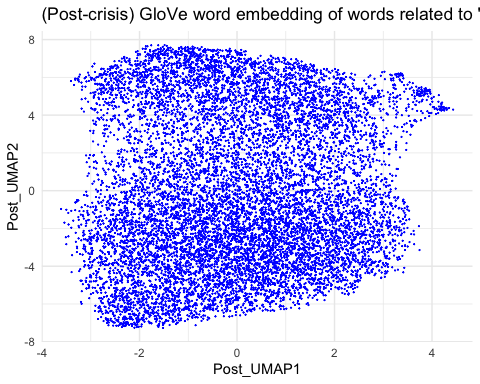
Modelling takes time. So I will use the prepared model made by same process for knitting.

url\_post <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove\_post.rds?raw=true"  
glove\_embedding\_post <- readRDS(url(url\_post, method = "libcurl"))

Pre-crisis total GloVe word embedding two dimensional umap

# Plotting the whole word embedding of pre-crisis immigration related text  
umap\_post <- umap(glove\_embedding\_post, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread = 2)  
  
df\_umap\_post <- as.data.frame(umap\_post[["layout"]])  
df\_umap\_post$word <- rownames(df\_umap\_post)  
colnames(df\_umap\_post) <- c("Post\_UMAP1", "Post\_UMAP2", "word")

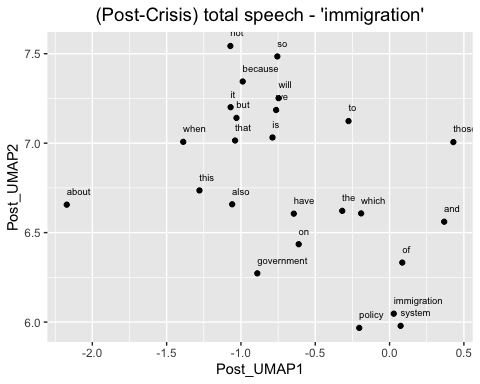
ggplot(df\_umap\_post) +  
 geom\_point(aes(x = Post\_UMAP1, y = Post\_UMAP2), color = 'blue', size = 0.05) +  
 labs(title = "(Post-crisis) GloVe word embedding of words related to 'immigration'") +  
 theme\_minimal()



Pre-crisis total parliament speech word embedding of words related to *immigration*

# Plot the word embedding of words that are related for the GloVe model (Case1: immigration)  
word\_post\_1 <- glove\_embedding\_post["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_post, y = word\_post\_1, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_post\_1 <- df\_umap\_post %>%   
 inner\_join(y=select, by= "word")

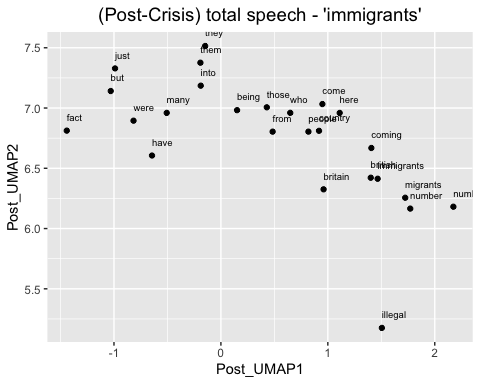
#The ggplot visual for GloVe  
ggplot(selected\_words\_post\_1, aes(x = Post\_UMAP1, y = Post\_UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_UMAP1, Post\_UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "(Post-Crisis) total speech - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis total parliament speech word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (Case2: immigrants)  
word\_post\_2 <- glove\_embedding\_post["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_post, y = word\_post\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_post\_2 <- df\_umap\_post %>%   
 inner\_join(y=select, by= "word")

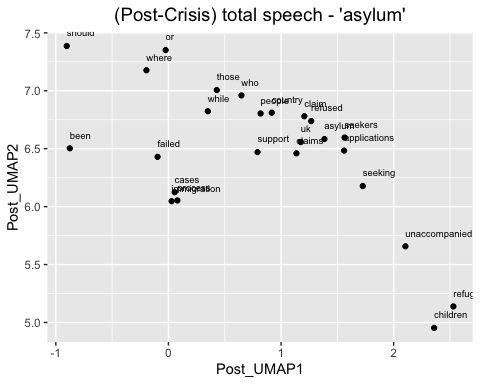
#The ggplot visual for GloVe  
ggplot(selected\_words\_post\_2, aes(x = Post\_UMAP1, y = Post\_UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_UMAP1, Post\_UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "(Post-Crisis) total speech - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Post-crisis total parliament speech word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (asylum)  
word\_post\_4 <- glove\_embedding\_post["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_post, y = word\_post\_4, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_post\_4 <- df\_umap\_post %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_post\_4, aes(x = Post\_UMAP1, y = Post\_UMAP2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_UMAP1, Post\_UMAP2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "(Post-Crisis) total speech - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))

 We can now compare the two periods with three different embedding maps.

# 2) Training by party and pre/post economic crisis

To make four different GloVe embedding models, dividing data by party and date

library(RcppParallel)  
  
# filter data by party and date. Now have pre/post economic crisis text data of Conservative and Labour party   
data\_conser\_pre <- filter(tidy\_data\_notoken, party == "Conservative" & date < as.Date("2007-09-24"))  
data\_labour\_pre <- filter(tidy\_data\_notoken, party == "Labour" & date < as.Date("2007-09-24"))  
  
data\_conser\_post <- filter(tidy\_data\_notoken, party == "Conservative" & date >= as.Date("2007-09-24"))  
data\_labour\_post <- filter(tidy\_data\_notoken, party == "Labour" & date >= as.Date("2007-09-24"))

### (1) The GloVe model for Conservative party before the 2008 economic crisis

Training process

set.seed(42L)  
glove\_text\_conser\_pre <- sample(data\_conser\_pre$desc)  
  
tokens\_conser\_pre <- space\_tokenizer(glove\_text\_conser\_pre)  
it\_conser\_pre <- itoken(tokens\_conser\_pre, progressbar = FALSE)  
vocab\_conser\_pre <- create\_vocabulary(it\_conser\_pre)  
vocab\_pruned\_conser\_pre <- prune\_vocabulary(vocab\_conser\_pre, term\_count\_min = COUNT\_MIN)  
  
vectorizer\_conser\_pre <- vocab\_vectorizer(vocab\_pruned\_conser\_pre)  
tcm\_conser\_pre <- create\_tcm(it\_conser\_pre, vectorizer\_conser\_pre, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric", weights = rep(1, WINDOW\_SIZE))  
  
glove\_conser\_pre <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
word\_vectors\_main\_conser\_pre <- glove\_conser\_pre$fit\_transform(tcm\_conser\_pre, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
word\_vectors\_context\_conser\_pre <- glove\_conser\_pre$components  
glove\_embedding\_conser\_pre <- word\_vectors\_main\_conser\_pre + t(word\_vectors\_context\_conser\_pre)  
  
saveRDS(glove\_embedding\_conser\_pre, file = "local\_glove\_conser\_pre.rds")

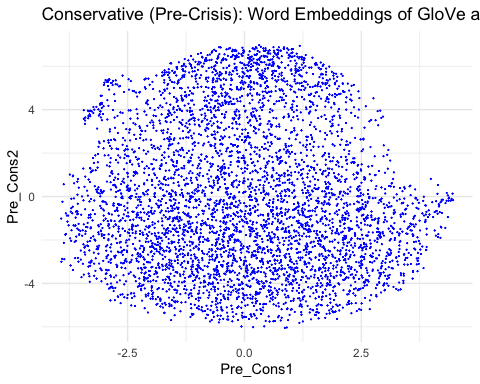
Modelling takes time. So I will use the prepared model made by same process for knitting.

url\_conser\_pre <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove\_conser\_pre.rds?raw=true"  
glove\_embedding\_conser\_pre <- readRDS(url(url\_conser\_pre, method = "libcurl"))

Total GloVe word embedding two dimensional umap

# Plotting the whole word embeddings of pre-crisis Conservative party immigration related text  
umap\_conser\_pre <- umap(glove\_embedding\_conser\_pre, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread = 2)  
  
df\_umap\_conser\_pre <- as.data.frame(umap\_conser\_pre[["layout"]])  
df\_umap\_conser\_pre$word <- rownames(df\_umap\_conser\_pre)  
colnames(df\_umap\_conser\_pre) <- c("Pre\_Cons1", "Pre\_Cons2", "word")

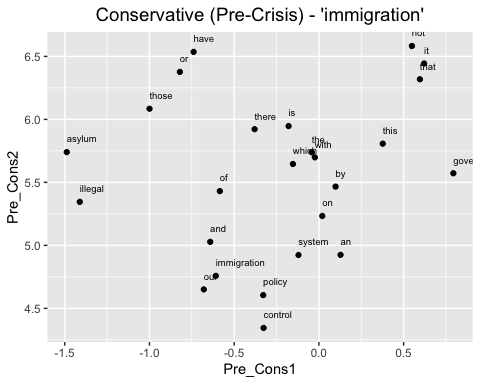
ggplot(df\_umap\_conser\_pre) +  
 geom\_point(aes(x = Pre\_Cons1, y = Pre\_Cons2), color = 'blue', size = 0.05) +  
 labs(title = "Conservative (Pre-Crisis): Word Embeddings of GloVe and UMAP") +  
 theme\_minimal()



Pre-crisis Conservative party word embedding of words related to *immigration*

# Plot the word embedding of words that are related for the GloVe model (Case1: immigration)  
word\_conser\_pre\_1 <- glove\_embedding\_conser\_pre["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_conser\_pre, y = word\_conser\_pre\_1, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_conser\_pre\_1 <- df\_umap\_conser\_pre %>%   
 inner\_join(y=select, by= "word")

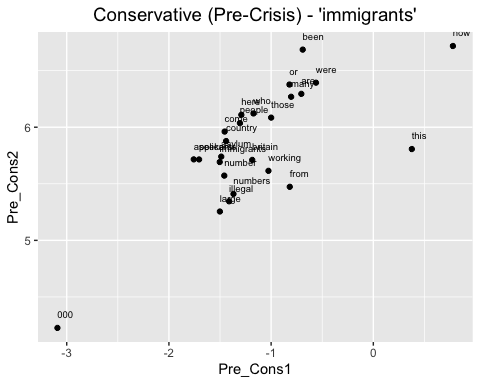
#The ggplot visual for GloVe  
ggplot(selected\_words\_conser\_pre\_1, aes(x = Pre\_Cons1, y = Pre\_Cons2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_Cons1, Pre\_Cons2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Conservative (Pre-Crisis) - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis Conservative party word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (Case2: immigrants)  
word\_conser\_pre\_2 <- glove\_embedding\_conser\_pre["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_conser\_pre, y = word\_conser\_pre\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_conser\_pre\_2 <- df\_umap\_conser\_pre %>%   
 inner\_join(y=select, by= "word")

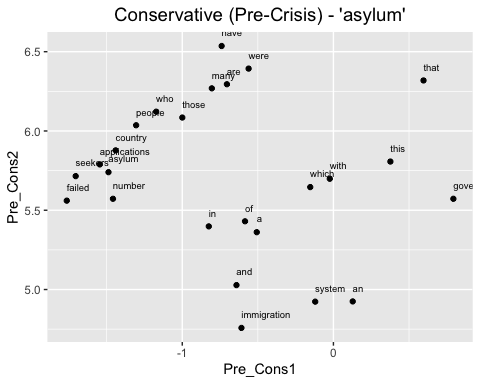
#The ggplot visual for GloVe  
ggplot(selected\_words\_conser\_pre\_2, aes(x = Pre\_Cons1, y = Pre\_Cons2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_Cons1, Pre\_Cons2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Conservative (Pre-Crisis) - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis Conservative party word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (case3: asylum)  
word\_conser\_pre\_3 <- glove\_embedding\_conser\_pre["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_conser\_pre, y = word\_conser\_pre\_3, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_conser\_pre\_3 <- df\_umap\_conser\_pre %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_conser\_pre\_3, aes(x = Pre\_Cons1, y = Pre\_Cons2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_Cons1, Pre\_Cons2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Conservative (Pre-Crisis) - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



### (2) The GloVe model for Conservative party after the 2008 economic crisis

Training process

set.seed(42L)  
glove\_text\_conser\_post <- sample(data\_conser\_post$desc)  
  
tokens\_conser\_post <- space\_tokenizer(glove\_text\_conser\_post)  
it\_conser\_post <- itoken(tokens\_conser\_post, progressbar = FALSE)  
vocab\_conser\_post <- create\_vocabulary(it\_conser\_post)  
vocab\_pruned\_conser\_post <- prune\_vocabulary(vocab\_conser\_post, term\_count\_min = COUNT\_MIN)  
  
vectorizer\_conser\_post <- vocab\_vectorizer(vocab\_pruned\_conser\_post)  
tcm\_conser\_post <- create\_tcm(it\_conser\_post, vectorizer\_conser\_post, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric", weights = rep(1, WINDOW\_SIZE))  
  
glove\_conser\_post <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
word\_vectors\_main\_conser\_post <- glove\_conser\_post$fit\_transform(tcm\_conser\_post, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
word\_vectors\_context\_conser\_post <- glove\_conser\_post$components  
glove\_embedding\_conser\_post <- word\_vectors\_main\_conser\_post + t(word\_vectors\_context\_conser\_post)  
  
saveRDS(glove\_embedding\_conser\_post, file = "local\_glove\_conser\_post.rds")

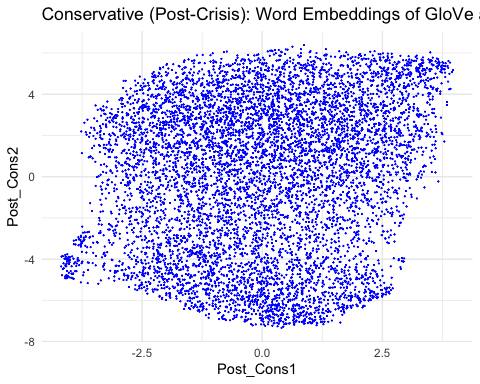
Modelling takes time. So I will use the prepared model made by same process for knitting.

url\_conser\_post <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove\_conser\_post.rds?raw=true"  
glove\_embedding\_conser\_post <- readRDS(url(url\_conser\_post, method = "libcurl"))

Total GloVe word embedding two dimensional umap

# Plotting the whole word embeddings of post-crisis Conservative party immigration related text  
umap\_conser\_post <- umap(glove\_embedding\_conser\_post, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread = 2)  
  
df\_umap\_conser\_post <- as.data.frame(umap\_conser\_post[["layout"]])  
df\_umap\_conser\_post$word <- rownames(df\_umap\_conser\_post)  
colnames(df\_umap\_conser\_post) <- c("Post\_Cons1", "Post\_Cons2", "word")

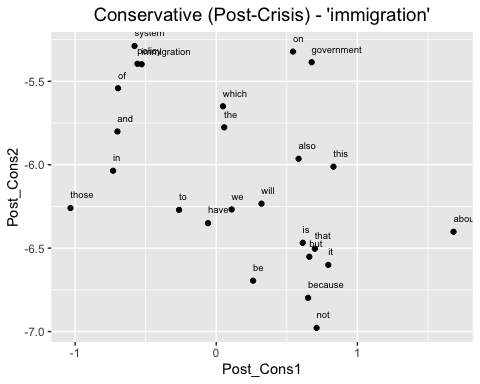
ggplot(df\_umap\_conser\_post) +  
 geom\_point(aes(x = Post\_Cons1, y = Post\_Cons2), color = 'blue', size = 0.05) +  
 labs(title = "Conservative (Post-Crisis): Word Embeddings of GloVe and UMAP") +  
 theme\_minimal()



Post-crisis Conservative party word embedding of words related to *immigration*

# Plot the word embedding of words that are related for the GloVe model (Case1: immigration)  
word\_conser\_post\_1 <- glove\_embedding\_conser\_post["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_conser\_post, y = word\_conser\_post\_1, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_conser\_post\_1 <- df\_umap\_conser\_post %>%   
 inner\_join(y=select, by= "word")

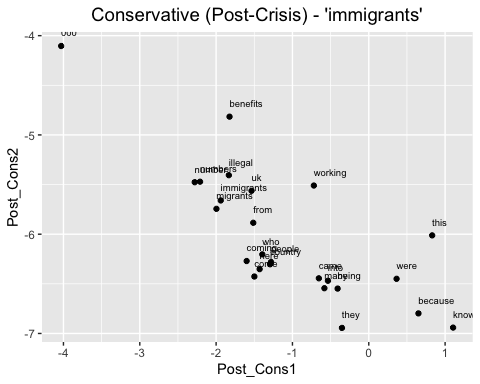
#The ggplot visual for GloVe  
ggplot(selected\_words\_conser\_post\_1, aes(x = Post\_Cons1, y = Post\_Cons2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_Cons1, Post\_Cons2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Conservative (Post-Crisis) - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Post-crisis Conservative party word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (case2: immigrants)  
word\_conser\_post\_2 <- glove\_embedding\_conser\_post["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_conser\_post, y = word\_conser\_post\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_conser\_post\_2 <- df\_umap\_conser\_post %>%   
 inner\_join(y=select, by= "word")

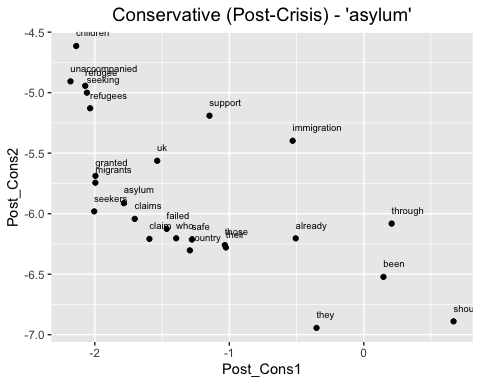
#The ggplot visual for GloVe  
ggplot(selected\_words\_conser\_post\_2, aes(x = Post\_Cons1, y = Post\_Cons2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_Cons1, Post\_Cons2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Conservative (Post-Crisis) - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Post-crisis Conservative party word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (Case3: asylum)  
word\_conser\_post\_4 <- glove\_embedding\_conser\_post["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_conser\_post, y = word\_conser\_post\_4, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_conser\_post\_4 <- df\_umap\_conser\_post %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_conser\_post\_4, aes(x = Post\_Cons1, y = Post\_Cons2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_Cons1, Post\_Cons2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Conservative (Post-Crisis) - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



### (3) The GloVe model for Labour party before the 2008 economic crisis

Training process

set.seed(42L)  
glove\_text\_labour\_pre <- sample(data\_labour\_pre$desc)  
  
tokens\_labour\_pre <- space\_tokenizer(glove\_text\_labour\_pre)  
it\_labour\_pre <- itoken(tokens\_labour\_pre, progressbar = FALSE)  
vocab\_labour\_pre <- create\_vocabulary(it\_labour\_pre)  
vocab\_pruned\_labour\_pre <- prune\_vocabulary(vocab\_labour\_pre, term\_count\_min = COUNT\_MIN)  
  
vectorizer\_labour\_pre <- vocab\_vectorizer(vocab\_pruned\_labour\_pre)  
tcm\_labour\_pre <- create\_tcm(it\_labour\_pre, vectorizer\_labour\_pre, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric", weights = rep(1, WINDOW\_SIZE))  
  
glove\_labour\_pre <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
word\_vectors\_main\_labour\_pre <- glove\_labour\_pre$fit\_transform(tcm\_labour\_pre, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
word\_vectors\_context\_labour\_pre <- glove\_labour\_pre$components  
glove\_embedding\_labour\_pre <- word\_vectors\_main\_labour\_pre + t(word\_vectors\_context\_labour\_pre)  
  
saveRDS(glove\_embedding\_labour\_pre, file = "local\_glove\_labour\_pre.rds")

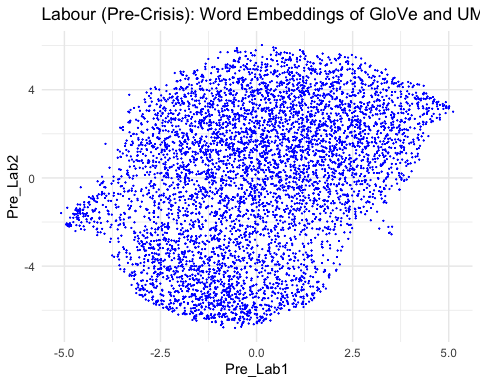
Modelling takes time. So I will use the prepared model made by same process for knitting.

url\_labour\_pre <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove\_labour\_pre.rds?raw=true"  
glove\_embedding\_labour\_pre <- readRDS(url(url\_labour\_pre, method = "libcurl"))

Total GloVe word embedding two dimensional umap

# Plotting the whole word embeddings of pre-crisis Labour party immigration related text  
umap\_labour\_pre <- umap(glove\_embedding\_labour\_pre, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread = 2)  
  
df\_umap\_labour\_pre <- as.data.frame(umap\_labour\_pre[["layout"]])  
df\_umap\_labour\_pre$word <- rownames(df\_umap\_labour\_pre)  
colnames(df\_umap\_labour\_pre) <- c("Pre\_Lab1", "Pre\_Lab2", "word")

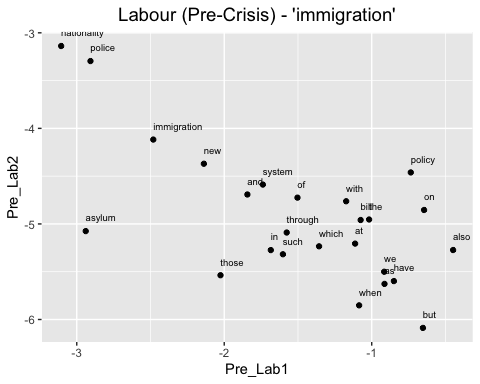
ggplot(df\_umap\_labour\_pre) +  
 geom\_point(aes(x = Pre\_Lab1, y = Pre\_Lab2), color = 'blue', size = 0.05) +  
 labs(title = "Labour (Pre-Crisis): Word Embeddings of GloVe and UMAP") +  
 theme\_minimal()



Pre-crisis Labour party word embedding of words related to *immigration*

# Plot the word embedding of words that are related for the GloVe model (case1: immigration)  
word\_labour\_pre\_1 <- glove\_embedding\_labour\_pre["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_labour\_pre, y = word\_labour\_pre\_1, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_labour\_pre\_1 <- df\_umap\_labour\_pre %>%   
 inner\_join(y=select, by= "word")

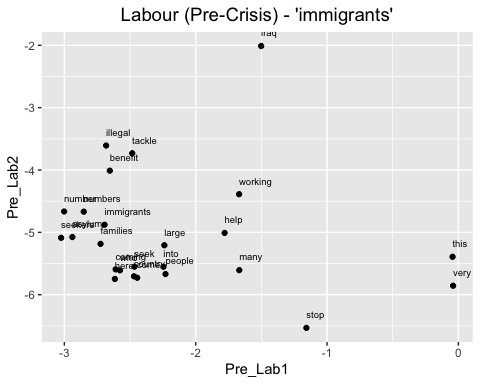
#The ggplot visual for GloVe  
ggplot(selected\_words\_labour\_pre\_1, aes(x = Pre\_Lab1, y = Pre\_Lab2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_Lab1, Pre\_Lab2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Labour (Pre-Crisis) - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis Labour party word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (Case2: immigrants)  
word\_labour\_pre\_2 <- glove\_embedding\_labour\_pre["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_labour\_pre, y = word\_labour\_pre\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_labour\_pre\_2 <- df\_umap\_labour\_pre %>%   
 inner\_join(y=select, by= "word")

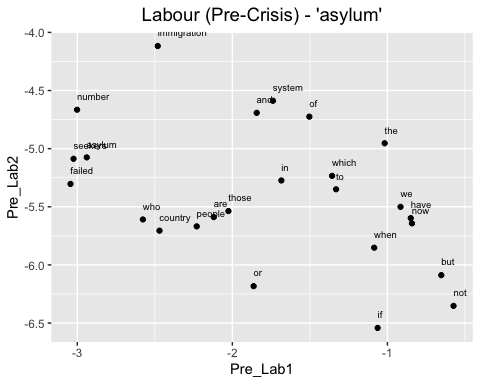
#The ggplot visual for GloVe  
ggplot(selected\_words\_labour\_pre\_2, aes(x = Pre\_Lab1, y = Pre\_Lab2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_Lab1, Pre\_Lab2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Labour (Pre-Crisis) - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis Labour party word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (case3: asylum)  
word\_labour\_pre\_3 <- glove\_embedding\_labour\_pre["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_labour\_pre, y = word\_labour\_pre\_3, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_labour\_pre\_3 <- df\_umap\_labour\_pre %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_labour\_pre\_3, aes(x = Pre\_Lab1, y = Pre\_Lab2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Pre\_Lab1, Pre\_Lab2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Labour (Pre-Crisis) - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



### (4) The GloVe model for Labour party after the 2008 economic crisis

Training process

set.seed(42L)  
glove\_text\_labour\_post <- sample(data\_labour\_post$desc)  
  
tokens\_labour\_post <- space\_tokenizer(glove\_text\_labour\_post)  
it\_labour\_post <- itoken(tokens\_labour\_post, progressbar = FALSE)  
vocab\_labour\_post <- create\_vocabulary(it\_labour\_post)  
vocab\_pruned\_labour\_post <- prune\_vocabulary(vocab\_labour\_post, term\_count\_min = COUNT\_MIN)  
  
vectorizer\_labour\_post <- vocab\_vectorizer(vocab\_pruned\_labour\_post)  
tcm\_labour\_post <- create\_tcm(it\_labour\_post, vectorizer\_labour\_post, skip\_grams\_window = WINDOW\_SIZE, skip\_grams\_window\_context = "symmetric", weights = rep(1, WINDOW\_SIZE))  
  
glove\_labour\_post <- GlobalVectors$new(rank = DIM, x\_max = 100, learning\_rate = 0.05)  
word\_vectors\_main\_labour\_post <- glove\_labour\_post$fit\_transform(tcm\_labour\_post, n\_iter = ITERS, convergence\_tol = 0.001, n\_threads = RcppParallel::defaultNumThreads())  
  
word\_vectors\_context\_labour\_post <- glove\_labour\_post$components  
glove\_embedding\_labour\_post <- word\_vectors\_main\_labour\_post + t(word\_vectors\_context\_labour\_post)  
  
saveRDS(glove\_embedding\_labour\_post, file = "local\_glove\_labour\_post.rds")

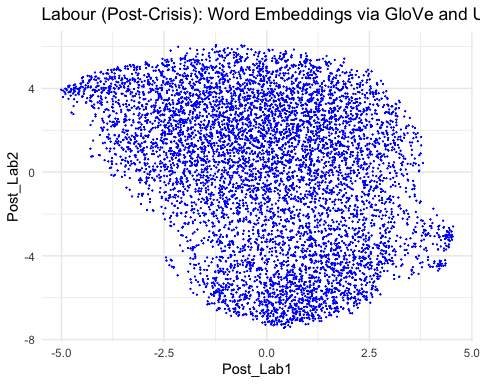
Modelling takes time. So I will use the prepared model made by same process for knitting.

url\_labour\_post <- "https://github.com/RiverKim-garam/CTA24-Final-assessment/blob/main/local\_glove\_labour\_post.rds?raw=true"  
glove\_embedding\_labour\_post <- readRDS(url(url\_labour\_post, method = "libcurl"))

Total GloVe word embedding two dimensional umap

# Plotting the whole word embeddings of post-crisis Labour party immigration related text  
umap\_labour\_post <- umap(glove\_embedding\_labour\_post, n\_components = 2, metric = "cosine", n\_neighbors = 25, min\_dist = 0.1, spread = 2)  
  
df\_umap\_labour\_post <- as.data.frame(umap\_labour\_post[["layout"]])  
df\_umap\_labour\_post$word <- rownames(df\_umap\_labour\_post)  
colnames(df\_umap\_labour\_post) <- c("Post\_Lab1", "Post\_Lab2", "word")

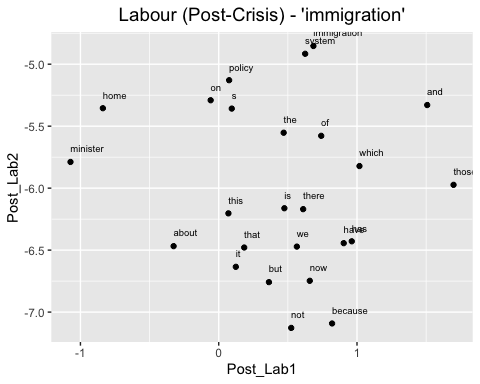
ggplot(df\_umap\_labour\_post) +  
 geom\_point(aes(x = Post\_Lab1, y = Post\_Lab2), color = 'blue', size = 0.05) +  
 labs(title = "Labour (Post-Crisis): Word Embeddings via GloVe and UMAP") +  
 theme\_minimal()



Post-crisis Labour party word embedding of words related to *immigration*

# Plot the word embedding of words that are related for the GloVe model (case1: immigration)  
word\_labour\_post\_1 <- glove\_embedding\_labour\_post["immigration",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_labour\_post, y = word\_labour\_post\_1, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_labour\_post\_1 <- df\_umap\_labour\_post %>%   
 inner\_join(y=select, by= "word")

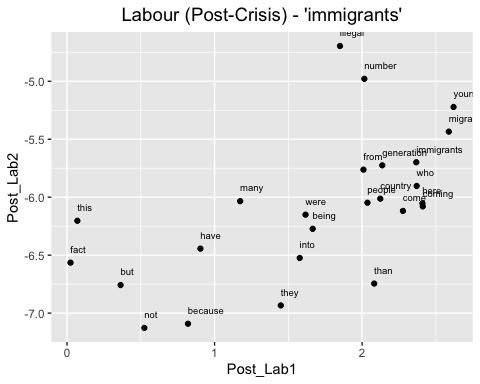
#The ggplot visual for GloVe  
ggplot(selected\_words\_labour\_post\_1, aes(x = Post\_Lab1, y = Post\_Lab2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_Lab1, Post\_Lab2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Labour (Post-Crisis) - 'immigration'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis Labour party word embedding of words related to *immigrants*

# Plot the word embedding of words that are related for the GloVe model (case2: immigrants)  
word\_labour\_post\_2 <- glove\_embedding\_labour\_post["immigrants",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_labour\_post, y = word\_labour\_post\_2, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_labour\_post\_2 <- df\_umap\_labour\_post %>%   
 inner\_join(y=select, by= "word")

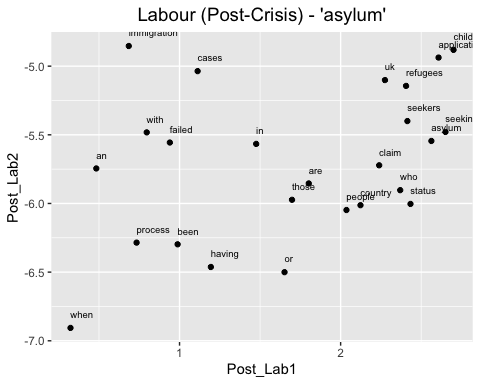
#The ggplot visual for GloVe  
ggplot(selected\_words\_labour\_post\_2, aes(x = Post\_Lab1, y = Post\_Lab2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_Lab1, Post\_Lab2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Labour (Post-Crisis) - 'immigrants'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Pre-crisis Labour party word embedding of words related to *asylum*

# Plot the word embedding of words that are related for the GloVe model (case3: asylum)  
word\_labour\_post\_3 <- glove\_embedding\_labour\_post["asylum",, drop = FALSE]  
cos\_sim = sim2(x = glove\_embedding\_labour\_post, y = word\_labour\_post\_3, method = "cosine", norm = "l2")  
select <- data.frame(rownames(as.data.frame(head(sort(cos\_sim[,1], decreasing = TRUE), 25))))  
colnames(select) <- "word"  
selected\_words\_labour\_post\_3 <- df\_umap\_labour\_post %>%   
 inner\_join(y=select, by= "word")

#The ggplot visual for GloVe  
ggplot(selected\_words\_labour\_post\_3, aes(x = Post\_Lab1, y = Post\_Lab2)) +   
 geom\_point(show.legend = FALSE) +   
 geom\_text(aes(Post\_Lab1, Post\_Lab2, label = word), show.legend = FALSE, size = 2.5, vjust=-1.5, hjust=0) +  
 labs(title = "Labour (Post-Crisis) - 'asylum'") +  
 theme(plot.title = element\_text(hjust = .5, size = 14))



Now we made three different specific words (immigration, immigrants, asylum) GloVE embedding umap by party and the period. By comparing the connections of words, we can estimate the contexts how the words were used.

# Instead of LaTex, use TinyTeX to knit the mark down result into pdf  
library(tinytex)