UNGA and Trade Data Collection Process

Zachary Leffel

2023-11-25

#Add global variables  
min\_year = 2011  
max\_year = 2023  
ccode\_list <- c(140, 850, 770, 771, 560, 433, 750, 670, 130, 373, 481, 475, 698)

#country code list added as dataframe  
cow\_codes <- read.csv("COW-country-codes.csv")  
colnames(cow\_codes) <- c("abr", "ccode", "country")  
nam\_countries <- read.delim("nam.txt")

colnames(nam\_countries) <- c("country")  
nam\_countries <- nam\_countries %>%  
 left\_join(.,cow\_codes, by="country") %>%  
 select(ccode) %>%  
 drop\_na()  
  
ccode\_list <- nam\_countries$ccode  
class(ccode\_list)

## [1] "integer"

ccode\_list

## [1] 700 615 540 373 31 692 771 53 370 80 434 760 145 571 835 439 516 811  
## [19] 471 402 483 155 100 581 484 40 40 522 54 42 42 860 130 651 651 411  
## [37] 531 530 530 950 481 420 452 55 90 438 404 110 41 41 91 750 850 630  
## [55] 645 437 51 663 501 690 812 660 570 450 620 580 553 820 781 432 435 590  
## [73] 712 600 600 541 775 565 790 93 436 475 731 698 770 95 910 135 840 694  
## [91] 517 403 670 433 591 451 830 520 560 780 625 115 652 652 510 800 461 52  
## [109] 616 616 701 500 696 704 935 101 816 679 551 552

### Economic Data

#import/export data prep for bilateral data  
rf\_imports <- read\_excel("rf\_exports.xlsx")

## New names:  
## • `` -> `...2`  
## • `` -> `...3`  
## • `` -> `...4`  
## • `` -> `...5`  
## • `` -> `...6`  
## • `` -> `...7`  
## • `` -> `...8`  
## • `` -> `...9`  
## • `` -> `...10`  
## • `` -> `...11`  
## • `` -> `...12`  
## • `` -> `...13`

rf\_exports <- read\_excel("rf\_imports.xlsx")

## New names:  
## • `` -> `...2`  
## • `` -> `...3`  
## • `` -> `...4`  
## • `` -> `...5`  
## • `` -> `...6`  
## • `` -> `...7`  
## • `` -> `...8`  
## • `` -> `...9`  
## • `` -> `...10`  
## • `` -> `...11`  
## • `` -> `...12`  
## • `` -> `...13`  
## • `` -> `...14`

rf\_exports <- rf\_exports %>%  
 select(!...2)  
tot\_exports <- read\_excel("tot\_exports.xlsx")

## New names:  
## • `` -> `...2`  
## • `` -> `...3`  
## • `` -> `...4`  
## • `` -> `...5`  
## • `` -> `...6`  
## • `` -> `...7`  
## • `` -> `...8`  
## • `` -> `...9`  
## • `` -> `...10`  
## • `` -> `...11`  
## • `` -> `...12`  
## • `` -> `...13`

tot\_imports <- read\_excel("tot\_imports.xlsx")

## New names:  
## • `` -> `...2`  
## • `` -> `...3`  
## • `` -> `...4`  
## • `` -> `...5`  
## • `` -> `...6`  
## • `` -> `...7`  
## • `` -> `...8`  
## • `` -> `...9`  
## • `` -> `...10`  
## • `` -> `...11`  
## • `` -> `...12`  
## • `` -> `...13`

trade\_clean <- function(df, type){  
 colnames(df) <- c(type, min\_year:2022)  
 df <- reshape2::melt(df, id = c(type))  
 colnames(df) <- c("country", "year", type)  
 df <- left\_join(df, cow\_codes, by="country")  
   
 df <- df %>%  
 mutate(ccode = case\_when(  
 country=="Azerbaijan, Rep. of"~373,  
 country=="Belarus, Rep. of"~370,  
 .default = as.numeric(ccode))) %>%  
 mutate(abr = case\_when(  
 country=="Azerbaijan, Rep. of"~"AZE",  
 country=="Belarus, Rep. of"~"BLR",  
 .default = as.character(abr))) %>%  
 filter(ccode %in% ccode\_list) %>%  
 drop\_na()  
   
 if(type=="exports"){  
 df <- df %>%  
 mutate(exports=as.numeric(exports))  
 }  
 else if(type=="tot\_exports"){  
 df <- df %>%  
 mutate(tot\_exports=as.numeric(tot\_exports))  
 }  
 else if(type=="imports"){  
 df <- df %>%  
 mutate(imports=as.numeric(imports))  
 }  
 else if(type=="tot\_imports"){  
 df <- df %>%  
 mutate(tot\_imports=as.numeric(tot\_imports))  
 }  
}

tot\_exports = trade\_clean(tot\_exports, "tot\_exports")

## Warning in left\_join(df, cow\_codes, by = "country"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 10 of `x` matches multiple rows in `y`.  
## ℹ Row 230 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.

tot\_imports = trade\_clean(tot\_imports, "tot\_imports")

## Warning in left\_join(df, cow\_codes, by = "country"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 10 of `x` matches multiple rows in `y`.  
## ℹ Row 230 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.

rf\_exports = trade\_clean(rf\_exports, "exports")

## Warning in left\_join(df, cow\_codes, by = "country"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 7 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

rf\_imports = trade\_clean(rf\_imports, "imports")

## Warning in left\_join(df, cow\_codes, by = "country"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 7 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

rf\_graph <- ggplot(rf\_exports, aes(x=year, y=exports, group=country, color=country)) + geom\_line() + geom\_point()  
rf\_graph



perc <- function(t, r, y){  
 p <- r / t \* 100  
 p <- cbind(y, p)  
 return(p)  
}

trade\_gather <- function(){  
 rfi <- rf\_imports #%>%  
 #filter(ccode==code)   
 ti <- tot\_imports #%>%  
 #filter(ccode==code)   
   
 ci <- left\_join(rfi, ti, by=c("year","ccode"))  
   
 rfe <- rf\_exports #%>%  
 #filter(ccode==code)   
 te <- tot\_exports #%>%  
 #filter(ccode==code)   
   
 ce <- left\_join(rfe, te, by=c("year", "ccode"))  
   
 trade\_df <- left\_join(ci, ce, by=c("year","ccode"))  
 trade\_df\_clean <- trade\_df %>%  
 rename\_at(  
 vars(ends\_with(".x")),  
 ~str\_replace(., "\\..$", "")  
 ) %>%  
 rename\_at(  
 vars(ends\_with(".x")),  
 ~str\_replace(., "\\..$", "")  
 ) %>%  
 select\_at(  
 vars(-ends\_with(c(".y", ".x.y", ".y.y",".y.x", ".x.x")))  
 ) %>%  
 mutate(year = as.numeric(as.character(year))) %>%  
 drop\_na()  
   
 return(trade\_df\_clean)  
}  
  
trade\_country <- function(c){  
 df <- trade\_gather()  
 df <- df %>%  
 filter(ccode == c)  
   
 return(df)  
}

#get the percentage of total imports and exports that Russia made up as a partner of the target country for each year  
partner\_size <- function(country, exports\_total, exports\_rus, imports\_total, imports\_rus, years){  
 exports\_perc <- perc(exports\_total, exports\_rus, years)  
 exports\_perc  
 imports\_perc <- perc(imports\_total, imports\_rus, years)  
 imports\_perc  
 summarized\_perc <- cbind(exports\_perc, imports\_perc)  
 paste(summarized\_perc)  
 colnames(summarized\_perc) <- c("year", "exports\_perc", "year2", "imports\_perc")  
 summarized\_perc <- subset(summarized\_perc, select=c(year, exports\_perc, imports\_perc))  
 summarized\_perc <- summarized\_perc %>% as.data.frame() %>% mutate(country = country)  
 return(summarized\_perc)  
}

partner\_trade <- function(trade\_df, ccode){  
 trade\_df <- trade\_df %>%   
 mutate(perc\_exports = exports/tot\_exports\*100)%>%  
 mutate(perc\_imports = imports/tot\_imports\*100)  
   
 return(trade\_df)  
}

theme\_update(title = element\_text(family="Times New Roman"), text = element\_text(family="Times New Roman"))  
  
visualize\_imports <- function(df){  
 ggplot(df, aes(x=year, y=imports\_perc)) + geom\_line(color="firebrick") + geom\_point(shape="diamond", size=2, color="firebrick") + labs(title = "Percentage of Imports from Russia", x="Year", y="Percentage of Total Imports")  
}  
  
visualize\_exports <- function(df){  
 ggplot(df, aes(x=year, y=exports\_perc)) + geom\_line(color="cornflowerblue") + geom\_point(shape="diamond", size=2, color="cornflowerblue") + labs(title = "Percentage of Exports going to Russia", x="Year", y="Percentage of Total Exports")  
}  
  
visualize\_trade <- function(df){  
 name <- df[1,1]  
 t = paste(name,": Trade with Russia as Percentage of Total Imports/Exports")  
 df\_long <- df %>%  
 select("perc\_exports", "perc\_imports", "year") %>%  
 pivot\_longer(-year, names\_to="variable", values\_to="value")  
 ggplot(df\_long, aes(year, value, col=variable)) + geom\_line() + geom\_point(shape="diamond", size=2) + scale\_colour\_manual(values = c("cornflowerblue", "firebrick"), labels=c("Exports", "Imports")) + labs(title=t, x="Year", y="Percentage of Total Imports/Exports", color="Type")   
}  
  
visualize\_trade\_all <- function(df){  
 df\_long <- df %>%  
 select("perc\_exports", "perc\_imports", "year", "country") %>%  
 pivot\_longer(-c(year, country), names\_to="variable", values\_to="value")  
 str(df\_long)  
 #colnames(df\_long)[colnames(df\_long) == 'country'] <- #"country\_"  
 ggplot(df\_long, aes(year, value, col=variable)) + geom\_line() + geom\_point(shape="diamond", size=2) + scale\_colour\_manual(values = c("cornflowerblue", "firebrick"), labels=c("Exports", "Imports")) + labs(title="Trade with Russia as Percentage of Total Imports/Exports", x="Year", y="Percentage of Total Imports/Exports", color="Type") + facet\_wrap(~country, scales="free\_y") + scale\_x\_continuous(breaks=seq(2010,2022,5), labels = seq(2010,2022,5), limits=c(2010,2022))  
}

year\_on <- function(ccode\_list){  
 year\_on\_list <- matrix(ncol=3, nrow=0)  
 gathered <- trade\_gather()  
 year\_on\_partners <- partner\_trade(gathered)  
 year\_on\_partners <- drop\_na(year\_on\_partners)  
 for(c in ccode\_list){  
 year\_on\_change <- year\_on\_partners %>%  
 select("perc\_exports", "perc\_imports", "year", "country", "ccode") %>%  
 pivot\_longer(-c(year, country, ccode), names\_to="variable", values\_to="value") %>%  
 filter(year==2021 | year==2022) %>%  
 filter(ccode == c)  
   
 change\_exp <- (year\_on\_change[1,5] - year\_on\_change[3,5])\*100/year\_on\_change[1,5]  
 r <- c(as.numeric(c), as.numeric(change\_exp), as.numeric(year\_on\_change[1,5]))  
 year\_on\_list <- rbind(year\_on\_list, r)  
 }  
 y <- as.data.frame(year\_on\_list)  
 y <- na.omit(y)  
 colnames(y) <- c("ccode", "year\_on\_decrease", "perc\_exports\_2021")  
 y <- left\_join(y, cow\_codes, by="ccode")  
 y <- y %>%   
 select("ccode", "country", "year\_on\_decrease", "perc\_exports\_2021")  
 return(y)  
}  
  
y1 <- year\_on(ccode\_list)

## Warning in left\_join(rfi, ti, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

## Warning in left\_join(rfe, te, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 18 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

## Warning in left\_join(ci, ce, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 18 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.

## Warning in left\_join(y, cow\_codes, by = "ccode"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 15 of `x` matches multiple rows in `y`.  
## ℹ Row 4 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.

y2 <- y1 %>%  
 filter(perc\_exports\_2021 > 1)

trade\_summary <-function(ccode){  
 gathered <- trade\_country(ccode)  
 all\_trade <- partner\_trade(gathered, ccode)  
 str(all\_trade)  
 all\_trade <- drop\_na(all\_trade)  
 visualize\_trade(all\_trade)  
}

trade\_summary(140)

## Warning in left\_join(rfi, ti, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

## Warning in left\_join(rfe, te, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 18 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

## Warning in left\_join(ci, ce, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 18 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.

## 'data.frame': 0 obs. of 10 variables:  
## $ country : chr   
## $ year : num   
## $ imports : num   
## $ abr : chr   
## $ ccode : num   
## $ tot\_imports : num   
## $ exports : num   
## $ tot\_exports : num   
## $ perc\_exports: num   
## $ perc\_imports: num



trade\_complete <- function(){  
 df <- trade\_gather()  
 df <- partner\_trade(df)  
 df <- drop\_na(df)  
 return(df)  
}  
  
df <- trade\_complete()

## Warning in left\_join(rfi, ti, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

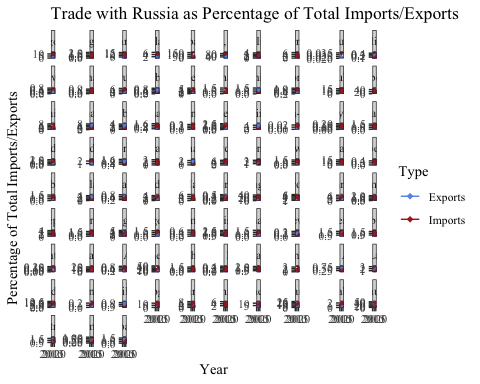
## Warning in left\_join(rfe, te, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 18 of `x` matches multiple rows in `y`.  
## ℹ Row 86 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

## Warning in left\_join(ci, ce, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 18 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.

visualize\_trade\_all(df)

## tibble [3,300 × 4] (S3: tbl\_df/tbl/data.frame)  
## $ year : num [1:3300] 2011 2011 2011 2011 2011 ...  
## $ country : chr [1:3300] "Algeria" "Algeria" "Angola" "Angola" ...  
## $ variable: chr [1:3300] "perc\_exports" "perc\_imports" "perc\_exports" "perc\_imports" ...  
## $ value : num [1:3300] 0.005893 1.087087 0.000512 0.047046 4.630534 ...

## `geom\_line()`: Each group consists of only one observation.  
## ℹ Do you need to adjust the group aesthetic?



### Agreement Scores

#Adding agreement scores as dataframe  
load("AgreementScoresAll\_Jul2023.Rdata")  
ideal\_point <- read.csv("IdealpointestimatesAll\_Jul2023.csv")

#function to calculate mean agreement with Russia over a timespan  
country\_unga\_data <- function(country\_code){  
 votes <- dfAgree[which(dfAgree$ccode1 == country\_code & dfAgree$ccode2 == 365 & dfAgree$year >= min\_year &dfAgree$year <= max\_year),]  
 print("Mean Agreement With Russia")  
 print(mean(votes$agree))  
 print("Mean Ideal Point")  
 print(mean(votes$IdealPointAll.x))  
 print("Ideal Point Distance")  
 print(mean(votes$IdealPointDistance))  
 return(votes)  
}

#function to get agreement scores by year for a set timespan  
country\_unga\_data\_vis <- function(country\_code){  
 votes <- dfAgree[which(dfAgree$ccode1 == country\_code & dfAgree$ccode2 == 365 & dfAgree$year >= min\_year &dfAgree$year <= max\_year),]  
 country\_matrix <- cbind(votes$ccode1, votes$agree, votes$year)  
 return(country\_matrix)  
}

#takes agreement data for each country and binds all of it together into a matrix  
agree\_over\_time <- function(ccode\_list){  
 agree\_vector <- lapply(ccode\_list, country\_unga\_data\_vis)  
 agree\_matrix <- do.call(rbind, as.list(agree\_vector))  
 return(agree\_matrix)  
}

country\_agreement <- agree\_over\_time(ccode\_list)  
  
#Adds column names  
colnames(country\_agreement) <- c("ccode","agree", "year")  
df\_country\_agreement <- as.data.frame(country\_agreement)  
#Dynamically adds a country name variable based on ccode  
df\_country\_agreement <- df\_country\_agreement %>%  
 mutate(country = case\_when(  
 ccode==140~"Brazil",  
 ccode==850~"Indonesia",  
 ccode==770~"Pakistan",  
 ccode==771~"Bangladesh",  
 ccode==652~"Syria",  
 ccode==560~"South Africa",  
 ccode==433~"Senegal",  
 ccode==670~"Saudi Arabia",  
 ccode==750~"India",  
 ccode==130~"Ecuador",  
 ccode==373~"Azerbaijan",  
 ccode==370~"Belarus",  
 ccode==481~"Gabon",  
 ccode==475~"Nigeria",  
 ccode==698~"Oman"  
 )) %>%  
 mutate(region = case\_when(  
 ccode==140~"South America",  
 ccode==850~"SE Asia",  
 ccode==770~"Middle East",  
 ccode==771~"SE Asia",  
 ccode==652~"Middle East",  
 ccode==560~"Africa",  
 ccode==433~"Africa",  
 ccode==670~"Middle East",  
 ccode==750~"SE Asia",  
 ccode==130~"South America",  
 ccode==373~"Eastern Europe",  
 ccode==370~"Eastern Europe",  
 ccode==481~"Africa",  
 ccode==475~"Africa",  
 ccode==698~"Middle East"  
 ))  
  
df\_trade <- trade\_complete()

## Warning in left\_join(rfi, ti, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

## Warning in left\_join(rfe, te, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 18 of `x` matches multiple rows in `y`.  
## ℹ Row 86 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.

## Warning in left\_join(ci, ce, by = c("year", "ccode")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 16 of `x` matches multiple rows in `y`.  
## ℹ Row 18 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.

merged\_df <- df\_country\_agreement %>%  
 left\_join(., df\_trade, by=join\_by("ccode","year")) %>%  
 mutate(after\_war = case\_when(  
 year==2022~1,  
 year<2022~0  
 ))

## Warning in left\_join(., df\_trade, by = join\_by("ccode", "year")): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 301 of `x` matches multiple rows in `y`.  
## ℹ Row 16 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.

lm1 <- lm(data = merged\_df, perc\_exports ~ year + agree)  
lm2 <- lm(data = merged\_df, perc\_exports ~ year + agree + region)  
summary(lm1)

##   
## Call:  
## lm(formula = perc\_exports ~ year + agree, data = merged\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.903 -0.709 -0.370 0.054 41.017   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -56.93827 35.58006 -1.600 0.110   
## year 0.02703 0.01750 1.545 0.122   
## agree 4.23292 0.92872 4.558 5.43e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.682 on 2415 degrees of freedom  
## (461 observations deleted due to missingness)  
## Multiple R-squared: 0.008596, Adjusted R-squared: 0.007775   
## F-statistic: 10.47 on 2 and 2415 DF, p-value: 2.968e-05

summary(lm2)

##   
## Call:  
## lm(formula = perc\_exports ~ year + agree + region, data = merged\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -17.5482 -1.3422 -0.3785 1.6054 19.5320   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -750.8807 320.2053 -2.345 0.020346 \*   
## year 0.3592 0.1570 2.287 0.023578 \*   
## agree 36.3277 10.3471 3.511 0.000591 \*\*\*  
## regionEastern Europe 18.7801 1.6120 11.650 < 2e-16 \*\*\*  
## regionMiddle East -0.5816 1.3331 -0.436 0.663240   
## regionSE Asia 0.2910 1.3324 0.218 0.827434   
## regionSouth America 5.5663 1.9484 2.857 0.004890 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.036 on 149 degrees of freedom  
## (2723 observations deleted due to missingness)  
## Multiple R-squared: 0.6267, Adjusted R-squared: 0.6116   
## F-statistic: 41.68 on 6 and 149 DF, p-value: < 2.2e-16

lm3 <-lm(data = merged\_df, perc\_imports ~ year + agree)  
lm4 <- lm(data = merged\_df, perc\_imports ~ year + agree + region)  
  
summary(lm3)

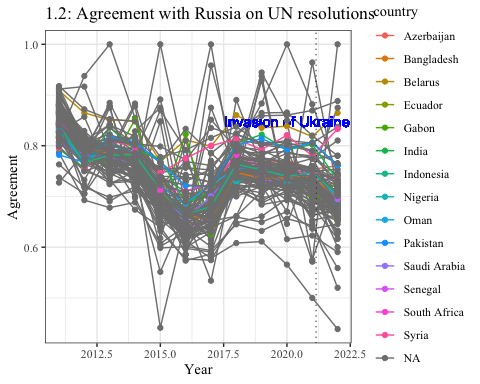
##   
## Call:  
## lm(formula = perc\_imports ~ year + agree, data = merged\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8.689 -2.912 -1.090 0.681 153.311   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -428.25254 95.73930 -4.473 8.07e-06 \*\*\*  
## year 0.20567 0.04708 4.368 1.30e-05 \*\*\*  
## agree 22.77061 2.49901 9.112 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.217 on 2415 degrees of freedom  
## (461 observations deleted due to missingness)  
## Multiple R-squared: 0.03337, Adjusted R-squared: 0.03257   
## F-statistic: 41.69 on 2 and 2415 DF, p-value: < 2.2e-16

summary(lm4)

##   
## Call:  
## lm(formula = perc\_imports ~ year + agree + region, data = merged\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -38.088 -3.735 -0.983 3.289 44.826   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1898.7109 712.9306 -2.663 0.00859 \*\*   
## year 0.9146 0.3496 2.616 0.00981 \*\*   
## agree 76.5476 23.0376 3.323 0.00112 \*\*   
## regionEastern Europe 35.4219 3.5890 9.870 < 2e-16 \*\*\*  
## regionMiddle East -2.9669 2.9681 -1.000 0.31912   
## regionSE Asia -1.7235 2.9665 -0.581 0.56212   
## regionSouth America -2.0328 4.3380 -0.469 0.64003   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 13.44 on 149 degrees of freedom  
## (2723 observations deleted due to missingness)  
## Multiple R-squared: 0.5734, Adjusted R-squared: 0.5562   
## F-statistic: 33.38 on 6 and 149 DF, p-value: < 2.2e-16

#ggplot(merged\_df, aes(x=year, y=lm2$res)) +  
#geom\_point() + geom\_line()  
#library(tseries)  
#runs.test(factor(lm2$res >0))

#Plots side-by-side line plots of each country's agreement score with Russia over time, distinguished by color.  
ggplot(df\_country\_agreement, aes(x=year, y=agree, group=ccode, color=country)) + geom\_line() + geom\_vline(xintercept=2021.15, linetype="dotted", linewidth = 0.3) + geom\_text(aes(x=2020,y=0.87,label="\nInvasion of Ukraine"), color="blue",angle=0) + geom\_point() + labs(x="Year", y="Agreement", title="1.2: Agreement with Russia on UN resolutions", group="Country")



#Plots side-by-side line plots of each country's ideal point over time, distinguished by color

#Gets data fram of Russia's military exercise data  
russia\_mil\_ex <- mil\_ex[which(mil\_ex$countryCode == 365 & mil\_ex$startYear > min\_year & mil\_ex$startYear < max\_year),]  
  
#Checks each military exercise target country has done to see if Russia was also involved, then puts any match into the country\_joint dataframe  
country\_mil\_ex\_data <- function(country\_code, country\_name){  
 country\_mil\_ex <- mil\_ex[which(mil\_ex$countryCode == country\_code & mil\_ex$startYear > min\_year & mil\_ex$startYear < max\_year),]  
 country\_joint <- merge(country\_mil\_ex, russia\_mil\_ex, by.x = "xID" , by.y = "xID")  
 print(ggplot(data=country\_joint, aes(x=startYear.x)) + geom\_bar() + labs(x="Year", y="Number of Joint Exercises", title = paste(country\_name, "/Russia Joint Exercises", sep='')))  
 by\_year <- country\_joint %>% group\_by(startYear.x) %>%tally()  
 ggplot(data=by\_year, aes(x=startYear.x, y=n)) + geom\_line() + labs(x="Year", y="Number of Joint Exercises", title = paste(country\_name, "/Russia Joint Exercises", sep='')) + geom\_line()  
 return(country\_joint)  
}

## Russia

russia\_votes <- dfAgree[which(dfAgree$ccode1 == 365 & dfAgree$year > min\_year &dfAgree$year < max\_year),]  
print("Mean Ideal Point")

## [1] "Mean Ideal Point"

print(mean(russia\_votes$IdealPointAll.x))

## [1] 0.0486335

## South Africa

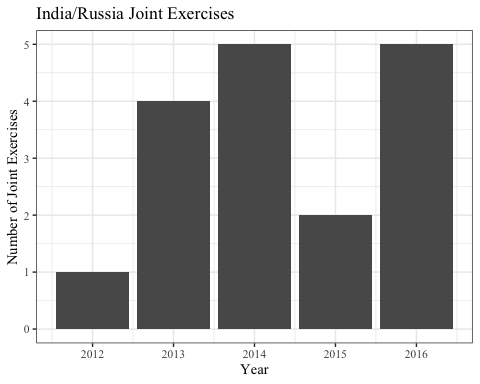
## Ecuador

## India

india\_votes <- country\_unga\_data(750)

## [1] "Mean Agreement With Russia"  
## [1] 0.7751741  
## [1] "Mean Ideal Point"  
## [1] -0.3688467  
## [1] "Ideal Point Distance"  
## [1] 0.4141368

india\_mil\_ex <- country\_mil\_ex\_data(750, "India")



#df\_india\_agree <- df\_country\_agreement[df\_country\_agreement$country == "India"]  
#ggplot(indiadf, aes(x=years, y=agree)) + geom\_line() + geom\_point()

## Saudi Arabia