Network Centrality Forecasting From Historical Diplomatic Data

Siddarth Viswanathan

1. Abstract

This project studies the relationship of network centrality measures and vector time series analysis. Network centrality measures relate for each node in a network how central it is to the other nodes. Given any dynamic network it should at each timepoint be possible to perform vector time series analysis by reducing the network structure to the network centrality measure over each node. This project studies a historical network of diplomatic representation observed every few years from the years 1817-2005. We reduce this dynamic network of diplomatic connections to appropriate network centrality measures and study vector time series models over these centrality measures for the purpose of forecasting and assessing accuracy of international power relations forecasts. We also provide cointegration analysis of the dynamic centrality-based time series and residual analysis of the fitted model.

2. Introduction

The main task of this project is to study how vector time series modeling is related to identification of central nodes in networks. There is a sparse literature on centrality aspects of networks in relation to time series models [1]. Historical diplomacy data can be represented as a network of diplomatic representation between nation-states. A nation can be said to have more influence over another nation if it has many high-level diplomat residing in the other nation. Especially before the information revolution it is especially useful to characterize nation-state relationships through the level of diplomatic representation. Since forecasting the entire network structure is not reasonable it is important to search for summary aspects of the network which can be forecasted and modeled.

In the field of international diplomacy the notion of centrality is especially important due to aspects of power in politics. Four centrality aspects are especially important: the in-degree, out-degree, eigenvector centrality, and betweenness centrality [2]. In this diplomatic example the in-degree gives the number of diplomats sent to a nation, while out-degree gives the number of diplomats sent from a nation, betweenness centrality provides a measure of how a nation serves as a bridge between other nations, eigenvector centrality is the clearest measure of influence in a network through its attribution of a higher score to those connected with higher-scores. It will be useful to explore how the intuitive notion of centrality score in diplomacy can be combined with time series forecasting to verify understanding of evolution of international power relations.

3. Data

The data from the Correlates of War Diplomatic Exchange dataset [3] contains a network of diplomatic representation sent between nations for various years between 1817 and 2005. If nation1 sends a high-level diplomat to nation2 then intuitively nation1 is imposing more power and responsibility into nation2. Therefore it is important to see the data as a weighted directed graph. There are 243 nation-states represented in the data over 38 time points. Figure 1 presents plots of the number of total diplomatic connections by year and the number of nations by year over the full dataset.

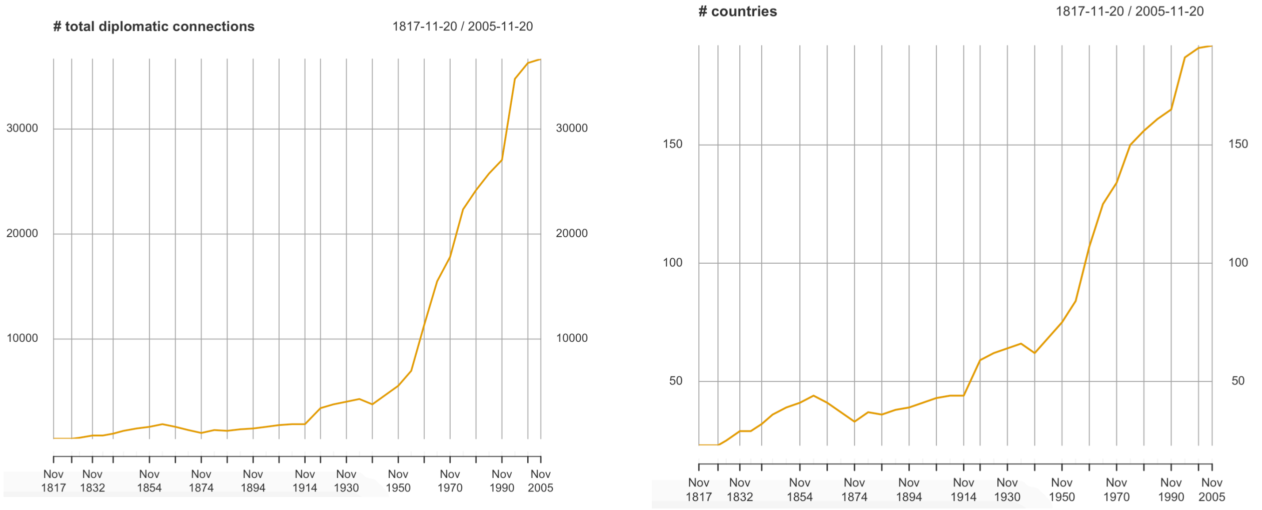


Figure 1: # total diplomatic connection by year and number of nations by year over the full dataset.

There is a large exponential growth of diplomatic connection and number of countries after WW2 and a gradual increase before then. It may be useful to try and forecast this large exponential growth with some model, so we restrict the data to the years 1832-1950. Figure 2 shows a subnetwork of the full network during the year 1832 to highlight that each time point summarized in the above plots is really a full network whose structure can usefully be summarized. Note that the graph is edges since a nation must send diplomats to another nation, also note that the graph is highly concentrated around a few central nodes such as France or UK.

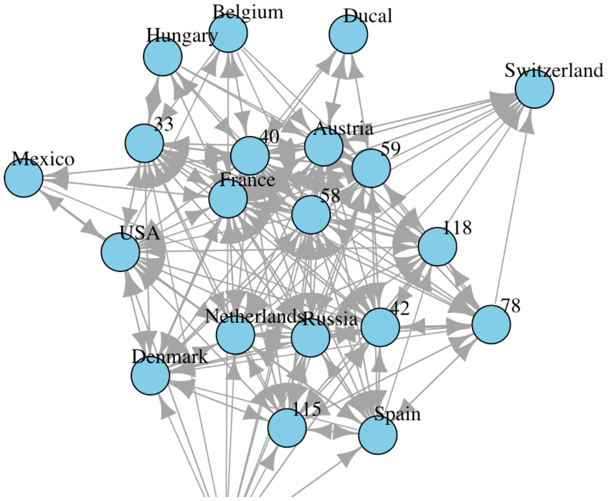
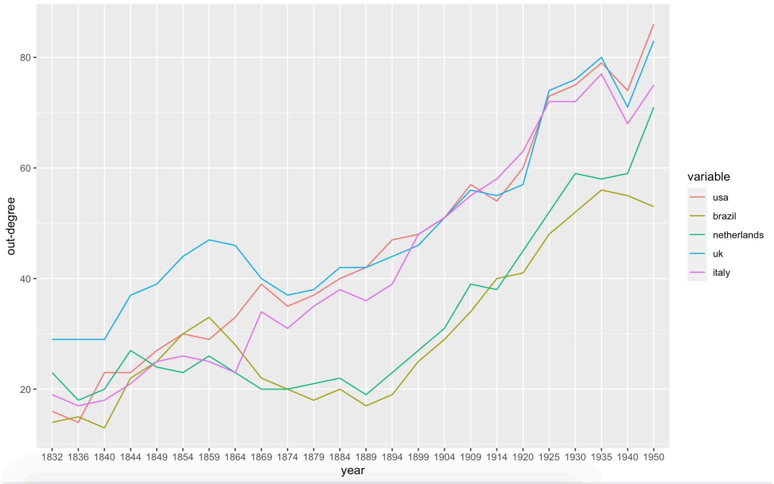


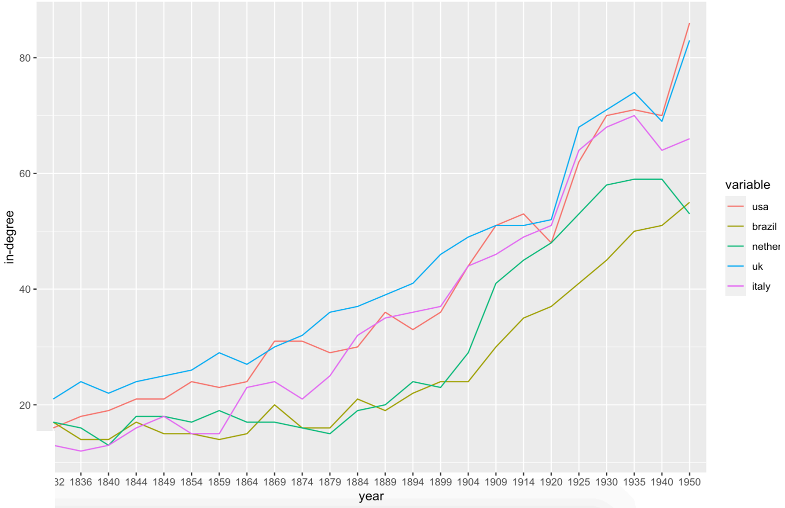
Figure 2: directed subgraph of 1832 network with some nations labeled.

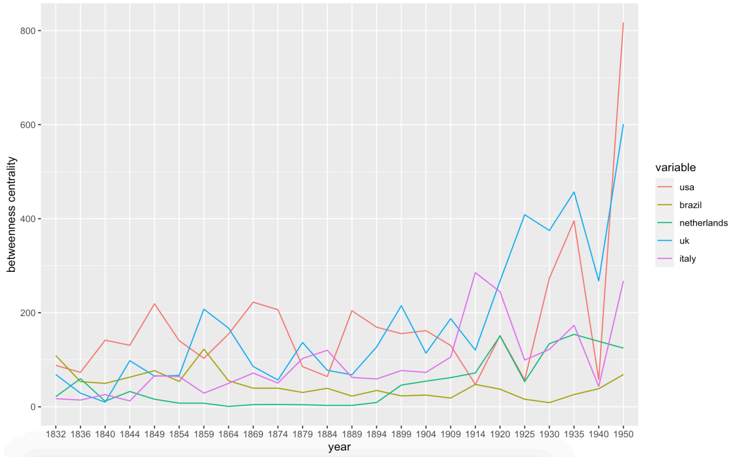
After restricting data from 1832-1950 we notice out of the 243 nation-states represented in the data only around 10%

contain consistent data entries and nonzero centrality measures throughout the entire time period. Therefore we select 5 nations—Italy, UK, USA, France, Brazil—which are present throughout the time period with nonzero centrality measures. Although each of the 243 nation-states can be turned into a time series by measuring centrality at each time point, we choose these 5 to focus centrality forecasting. A major limitation of this study is that after reduction there are only 24 data points which makes any testing or model-fitting suscpicious. However the main focus of this project is to study questions at the interface of network centrality and vector time series, so although the modeling results will not be very trustworthy the methodological process may be useful.

Figure 3 reveals the five time series for each of the derived four network centrality measures.







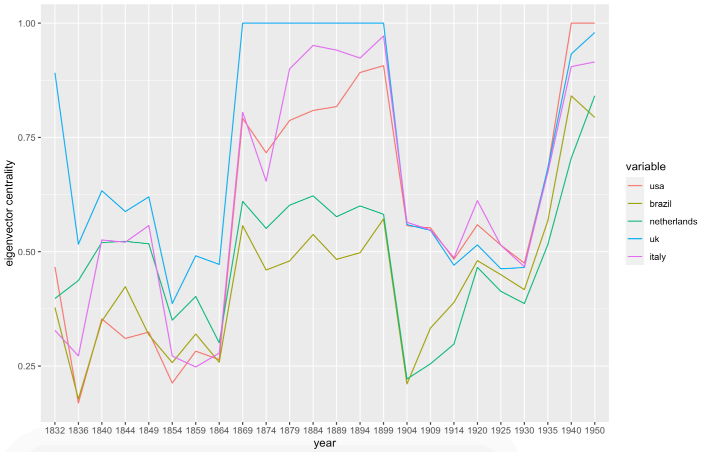


Figure 3: plot of the five time series for each of the four derived network centrality meausures from 1832-1950.

4. Model identification and fitting

First the betweenness centrality, in-degree, out-degree, eigenvector centrality are calculated for each network of diplomatic connections at each time point. These can be calculated using the igraph package in R [4]. Figure 4 illustrates that the series should be differenced by one order. All five series fail the automated Dickey-Fuller test for nonstationarity both before and after differencing, however this can be ignored due to the small data size. From visual inspection a one-lag difference creates seemingly stationary series however the small data size leads to seeming nonstationary aspects which can be assumed to inherently be stationary if the data were interpolated.

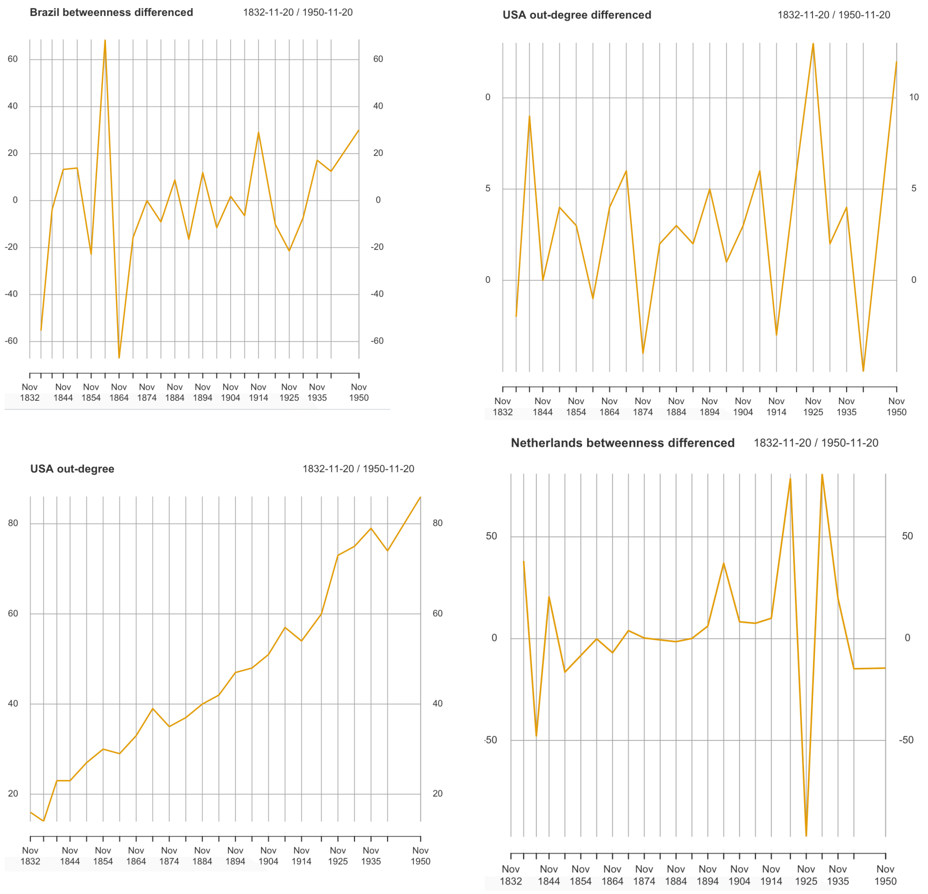


Figure 4 (from top-left clockwise) a) differenced centrality plot for Brazil b) differenced centrality plot for USA c) differenced centrality plot for Netherlands d) non-differenced centrality plot for USA.

Figure 4 reveals only a few of the series differenced or non-differenced, however from further visual inspection almost all the 20 = 5 nations * 4 centrality measures follow this this pattern of revealing stationarity after one differencing.

Since the series are differenced we run the KPSS test for a significant trend and there is not enough evidence to claim the deterministic trend parameter is nonzero. This holds across all the 20 series.

The sample CCF analysis reveals quickly falling off CCF for in-degree and out-degree but not for betweenness or eigenvector centrality. Table 2 shows CCF lag with drop below significance for the five nations' series for in-degree and out-degree. This table can be use to gather insight about potential moving average components of a model. Table 1 can be interpreted to suggest a MA(2) or MA(1) component for both in-degree and out-degree.

In-degree CCF significant lag	USA	Brazil	Netherlands	UK	Italy
USA	.	1	3	1	1
Brazil	.	.	3	2	3
Netherlands	.	.	.	0	1
UK	.	.	.	.	3

Out-degree CCF significant lag	USA	Brazil	Netherlands	UK	Italy
USA	.	2	2	1	1
Brazil	.	.	1	2	1
Netherlands	.	.	.	1	2
UK	.	.	.	.	1

Table 1 table revealing large drops in CCF plots at various lags for in-degree and out-degree time series measures suggesting a MA(2) component for both in-degree and out-degree. Betweeness andeigencentrality are not shown in the table since they revealed only oscillating patterns with no diminishing lag significance.

The sample PACF analysis did not reveal any significant drop for betweenness, eigencentrality, or in-degree. However for out-degree there was slight evidence of a quick lag drop, however there is not significant evidence. Still, it may be useful to add a AR(1) component since this will improve the future range of the forecasts.

With these CCF and PACF results it does not seem useful to consider further the eigencentrality or betweenness centralities for modeling. The models which make sense are the vector-ARIMA(1,1,1) or vector-ARIMA(1,1,2) for both in-degree and out-degree measures. The vector-ARIMA(1,1,2) leads to an average AIC score of 140.27 while the vector-ARIMA(1,1,1) due to a simpler model has a lower average AIC score of 138.39.

The vector-ARIMA(1,1,1) model is given as $(I-\Phi_1 B)$ $(1-B)$ $Z_t = (1+\Theta_1 B)a_t$ where a_t is a Gaussian error term. Table 2 summarizes the estimated model parameters.

	(se) Φ_1	(se) Θ_1	(se) intercept	σ^2 estimate
USA	(.36) -.089	(.31) -.45	(.42) 2.97	15.42
Brazil	(.35) .49	(.37) -.15	(1.2) 1.55	13.13
Netherlands	(.17) -.64	(.16) 1.0	(.98) 2.06	15.18
UK	(.16) .78	(.12) -1.0	(.46) 2.2	24.47
Italy	(.95) -.12	(.96) -.25	(.61) 2.47	16.87

Table 2 estimated model. It is difficult to interpret further these values due to the small sample size.

It is difficult to interpret further these values due to lack of interpolating between the large 5-year data intervals and the small sample size.

5. Residual and Cointegration analysis

Figure 5 shows the model residuals and CCF residual plots.

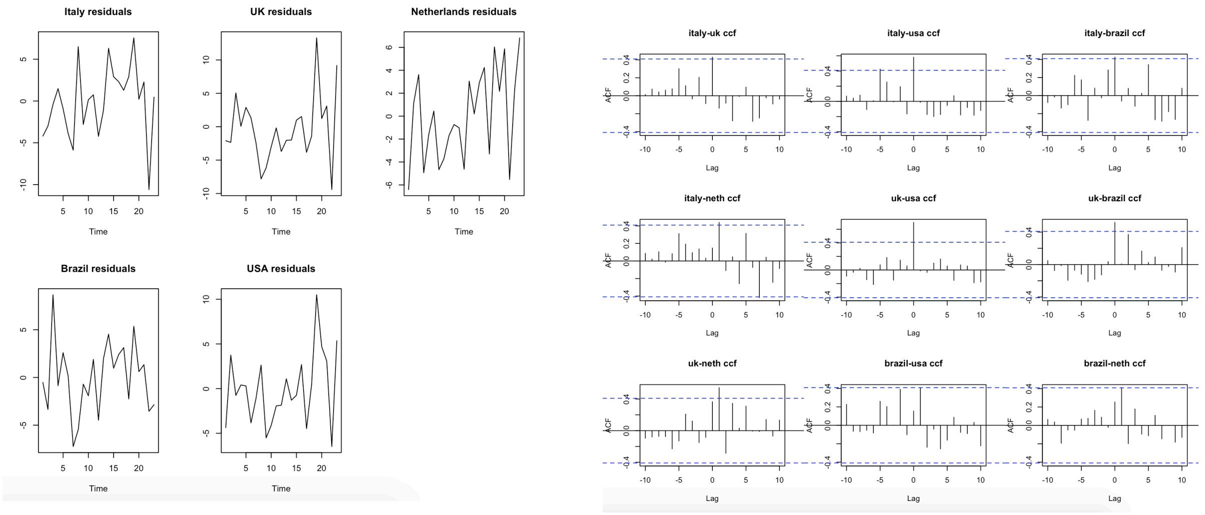


Figure 5: model residual and CCF residual plots. For the small sample size the residuals are reasonably stationary and do not rise beyond significant lags.

Each of the residual series also passes the automated Dickey-Fuller test at the .1 significance level revealing stationarity of model residuals.

The five time series of interest are not cointegrated. Table 3 shows the Johansen test for cointegration run on the in-degree, betweenness, eigen, out-degree centralities time series for the five nations.

	r (# cointegration relations)	test-stat for (r-1)	.05-critical value for (r-1)
in-degree	2	28.43	27.14
betweeness	3	25.08	21.07
eigen	2	39.39	27.14
out-degree	1	39.17	33.32

Table 3 Results from Johansen test for cointegration revealing significant cointegration among the five constructed time series.

Further evidence for cointegration in the series will come from revealed stationary of model residuals. There is significant cointegration among the data which we would expect since the centrality measurements over the diplomatic relationships are formed from a network relating the nations.

6. Forecasting

Table 4 presents the forecast with 95% confidence intervals.

	Year	Out-degree forecast	Lower 95%	Upper 95%
USA	1955	-0.27	-7.9	7.42
	1960	3.2	-5.49	12.03
	1965	2.95	-5.8	11.72
	1970	2.97	-5.79	11.75
	1975	2.98	-5.79	11.74
Brazil	1955	0.25	-6.82	7.32
	1960	0.91	-6.63	8.4
	1965	1.24	-6.35	8.87
	1970	1.46	-6.21	9.16
	1975	1.51	-6.18	9.11
Netherlands	1955	2.32	-5.41	10.11
	1960	1.87	-6.25	10.03
	1965	2.19	-6.16	10.5
	1970	1.98	-6.39	10.36
	1975	2.1	-6.44	10.51
UK	1955	0.86	-8.92	10.76

	1960	1.15	-8.81	11.15
	1965	1.38	-8.75	11.55
	1970	1.56	-8.61	11.78
	1975	1.72	-8.51	11.92
Italy	1955	1.83	-6.25	9.85
	1960	2.6	-6.02	11.15
	1965	2.46	-6.14	11.01
	1970	2.47	-6.11	11.07
	1975	2.48	-6.12	11.07

Table 4 the forecasts of the vector-ARIMA(1,1,1) with 95% confidence intervals.

The data was truncated at 1950 however the 5-year data after 1950 still exists and it would be useful to see if the model predicts the exponential rise in diplomacy post-WW2. The actual 5-period ahead forecast is consistently above 90 for the USA and above 50 for the other four countries. However the forecast provided by the model does not stray far from zero and completely does not capture the exponential growth post-WW2. Again increased sample sizes and interpolation will improve these forecasts.

7. Conclusion

This project has illustrated a relationship between network centrality measures and vector time series. A variety of centrality measures can be calculated for each node of a dynamic network which can then be used to create a vector time series over the network. The results shown can be greatly improved with more datapoints and also datapoints which are closer in time together. Model results may be improved by interpolating points in the time series since the points are around 5 years apart and any meaning to the time series lags becomes complicated. It is interesting that only in-degree and out-degree were the only centrality measures revealing any interesting CCF or PACF plot since these are the measures most directly related to diplomatic influence. The more complicated measures of betweenness and eigenvector centrality may not have been useful due to small sample sizes and network sparsity. The key idea of this project is that combining network centrality measures is a notion which fits intuitively with vector time series analysis; this is because network centrality measures allow each node of a network to form a separate but related time series from the other nodes.

8. References

[1] Huang, Qiangjuan, 2017. Centrality measures in temporal networks with time series analysis
Electronic physics letters (118)36001.

[2] Landherr, Andrea. 2010. A critical review of centrality measures in social networks, Business & Information Systems Engineering.

[3] Bayer, Reşat. 2006. “Diplomatic Exchange Data set, v2006.1.”

[4] Csardi G, Nepusz T (2006). “The igraph software package for complex network research.” *InterJournal*, Complex Systems, 1695.

9. Data

http://correlatesofwar.org Version 2006.v1 of the diplomatic exchange data set is hosted by Reşat Bayer (Koç University) under the COW Data Set Hosting Program.

10. Code

```
library(ggplot2)
library(reshape2)
library(xts)
library(igraph)
library(aTSA)
library(forecast)
library(urca)

rm(list=ls())

setwd('/Users/siddarthviswanathan/Desktop/')

##############
## 0. Load dynamic network data
##############

dat <- read.csv('Diplomatic_Exchange_2006v1.csv', stringsAsFactors = F)
codes <- read.csv('COW country codes.csv' , stringsAsFactors = F)

years <- names(table(dat$year))
all_codes <- codes$CCode

year <- "1832"
year_dat <- dat[dat$year == year, ]

##############
## 1. Basic exploratory plots
##############

## get num connections and num countries by year
numconnections_by_year <- rep(NA, length(years))
numcountries_by_year <- rep(NA, length(years))

for(i in 1:length(years)){
  year <- years[i]
  test_dat <- dat[dat$year == year, ]
  numconnections_by_year[i] <- nrow(test_dat)
  numcountries_by_year[i] <- length(unique(test_dat$ccode1))
}
par(mfrow=c(1,1))
plot(xts(numconnections_by_year, order.by=as.POSIXct(years, format="%Y")),main='# total diplomatic connections')
plot(xts(numcountries_by_year, order.by=as.POSIXct(years, format="%Y")), main='# countries')

#reduced_years <- years[4:27]
reduced_years <- years[28:32]
year <- reduced_years[10]
```

```
##############
## 2. Create and store all adjacency matrices of diplomacy
##############

list_adj_mat <- vector("list", length = length(reduced_years))

for(k in 1:length(reduced_years)){
  year <- reduced_years[k]
  print(year)

  # initialize adjacency matrix
  adj_mat <- matrix(0, nrow=length(all_codes), ncol=length(all_codes))

  # get data for the year
  year_dat <- dat[dat$year == year,]

  # add weighted entries into adjacency matrix
  for(i in 1:length(all_codes)){
    for(j in 1:length(all_codes)){
      newdat <- year_dat[which(year_dat$ccode1 == all_codes[i] & year_dat$ccode2 == all_codes[j]), ]

      if(nrow(newdat)>0){
        adj_mat[i,j] <- newdat$DR_at_2
      }
    }
  }

  # store weighted matrix into list
  list_adj_mat[[k]] <- adj_mat

}

##############
## 3. get centrality scores for all the time periods
##############

a <- graph_from_adjacency_matrix(list_adj_mat[[4]],
                                 diag=F,
                                 mode='directed',
                                 weighted=T)

degree(a)

all_codes[c] # papal states highest during this year

list_adj_mat_eigen <- vector("list", length = length(reduced_years))
list_adj_mat_betweenness <- vector("list", length = length(reduced_years))
list_adj_mat_degree_out <- vector("list", length = length(reduced_years))
list_adj_mat_degree_in <- vector("list", length = length(reduced_years))

for(i in 1:length(reduced_years)){
  # create igraph object
  graph_obj <- graph_from_adjacency_matrix(list_adj_mat[[i]],
                                           diag=F,
                                           mode='directed',
                                           weighted=T)

  # fill in information of interest
  list_adj_mat_eigen[[i]] <- eigen_centrality(graph_obj)$vector
  list_adj_mat_betweenness[[i]] <- betweenness(graph_obj)
  list_adj_mat_degree_out[[i]] <- degree(graph_obj, mode="out")
  list_adj_mat_degree_in[[i]] <- degree(graph_obj, mode="in")
}

list_adj_mat_degree_out[[5]][c(1,33,42,40,78)]
list_adj_mat_degree_in[[5]][c(1,33,42,40,78)]

# select the nodes having these names
selnodes <- V(graph_obj)[name %in% nodes_of_interest]
selegoV <- ego(graph_obj, nodes = selnodes, mode = "in", mindist = 0)
selegoG <- induced_subgraph(graph_obj,unlist(selegoV))
plot(selegoG,vertex.label=V(selegoG)$name,
     vertex.label.color="black", vertex.label.dist=1.5,
     vertex.color=c("skyblue"))

##############
## 4. create time series of network centralities and use most important nodes.
##############

# get network centralities
for(i in 1:length(list_adj_mat)){
  country_vec_eigen[i] <- list_adj_mat_eigen[[i]][country_code_position]
  country_vec_betweenness[i] <- list_adj_mat_betweenness[[i]][country_code_position]
  country_vec_degree_out[i] <- list_adj_mat_degree_out[[i]][country_code_position]
  country_vec_degree_in[i] <- list_adj_mat_degree_in[[i]][country_code_position]
}

degree_out_ts <- data.frame(usa=xts(degree_out_df[,1], order.by=new_years),
                            brazil=xts(degree_out_df[,2], order.by=new_years),
                            netherlands=xts(degree_out_df[,3], order.by=new_years),
                            uk=xts(degree_out_df[,4], order.by=new_years),
                            italy=xts(degree_out_df[,5], order.by=new_years))

##############
## 6. model identification
##############


kpss.test(diff(italy,1)) #no drift .09
kpss.test(diff(uk,1)) #no drift .12
kpss.test(diff(netherlands,1)) # .07
kpss.test(diff(usa,1)) # .1
kpss.test(diff(brazil,1)) #no drift, .1

# all series differenced by 1
# degree_in_df, degree_out df, betweenness_df, eigen_df
dataset <- eigen_df
usa <- diff(xts(dataset[,1], order.by=new_years),1)
brazil <- diff(xts(dataset[,2], order.by=new_years),1)
netherlands <- diff(xts(dataset[,3], order.by=new_years),1)
uk <- diff(xts(dataset[,4], order.by=new_years),1)
italy <- diff(xts(dataset[,5], order.by=new_years),1)

eigen_ts <- data.frame(usa=usa[-1],
                       brazil=brazil[-1],
                       netherlands=netherlands[-1],
                       uk=uk[-1],
                       italy=italy[-1])

## some cointegration analysis

summary(ca.jo(degree_out_ts, type='eigen', K=2))
summary(ca.jo(degree_in_ts, type='eigen', K=2))
summary(ca.jo(betweenness_ts, type='eigen', K=2))
summary(ca.jo(eigen_ts, type='eigen', K=2))

# lag beyond which no significant
ccf(as.numeric(usa)[-1], as.numeric(brazil)[-1]) #4


pacf.mts(degree_in_ts, 5) # some evidence of quick lag drop


#### fit model for in and out-degree ###
res <- lapply(degree_out_ts, arima, order=c(0,0,1))
res <- lapply(degree_out_ts, arima, order=c(0,0,2))
res <- lapply(degree_in_ts, arima, order=c(0,0,1))
res <- lapply(degree_in_ts, arima, order=c(0,0,2))

mod1a <- lapply(degree_out_ts, arima, order=c(1,0,1)) #138.39
mod1b <- lapply(degree_out_ts, arima, order=c(1,0,2)) #140.27
mod2a <- lapply(degree_in_ts, arima, order=c(1,0,1)) #135.3

lapply(mod1a, function(model) forecast(model,h=5))
lapply(mod1b, function(model) forecast(model,h=5))
lapply(mod2, function(model) forecast(model,h=5))

# compare forecasts with actual values
rmse(c(), c()) way off

res#############
## 7. model verification for in and out-degree
##############
mod1a$residuals
lapply(mod1a, function(model) mod1a$residuals)

par(mfrow=c(2,3))
plot(mod1a$italy$residuals, main='Italy residuals', ylab='')
plot(mod1a$uk$residuals, main='UK residuals', ylab='')
plot(mod1a$netherlands$residuals, main='Netherlands residuals', ylab='')
plot(mod1a$brazil$residuals, main='Brazil residuals', ylab='')
plot(mod1a$usa$residuals, main='USA residuals', ylab='')

par(mfrow=c(3,3))
ccf(mod1a$italy$residuals, mod1a$uk$residuals, main='italy-uk ccf')
```