

# AI-Assisted Data Management for International Relations Scholars

---

Paul Poast, University of Chicago

# Summary

- Software & websites for assisting scholars in merging variables to generate analysis datasets was popular in 2000s and 2010s.

- Key Example was *EuGene*

## *Eugene: A conceptual manual*

D. Scott Bennett & Allan C. Stam



[International Interactions >](#)

Empirical and Theoretical Research in International Relations

Volume 26, 2000 - [Issue 2](#)

- Advances in LLMs have now rendered such platforms obsolete.

# The Original Need for *EuGene*

- Over the course of the 1970s, 1980s, and 1990s, an increasing amount of international relations scholarship used large-*n* data.
- Advances in computing power eventually enabled scholars to perform such analysis using spreadsheets on personal computers.
- But management of data, especially merging variables from disparate datasets in order to create an “analysis dataset”, was a chore.

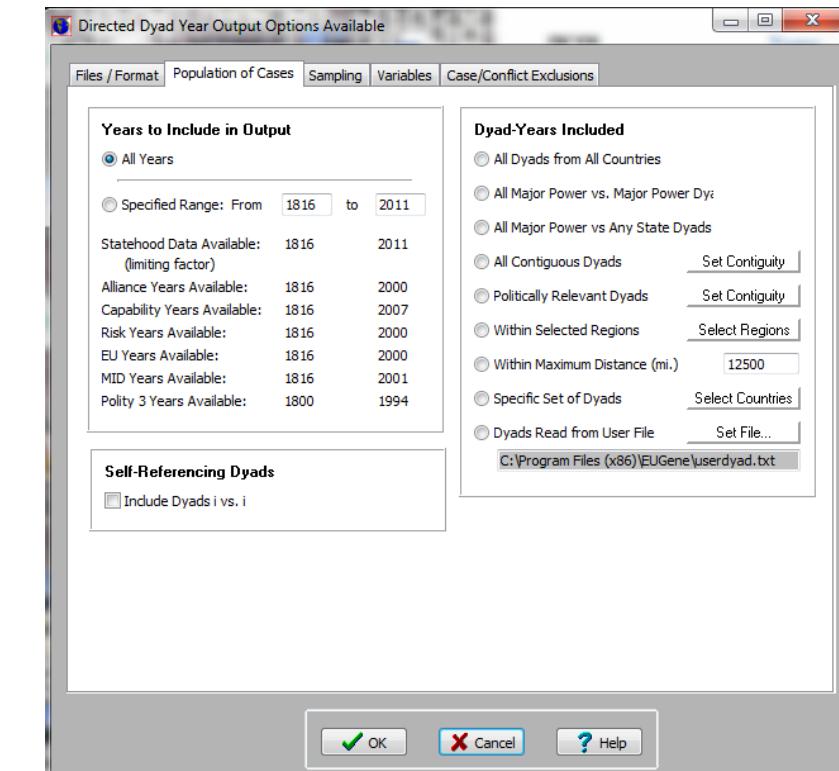
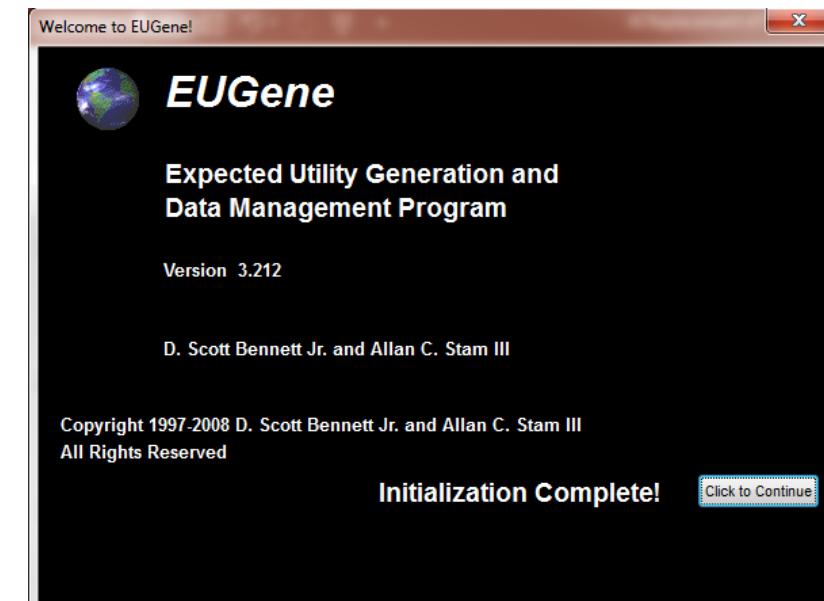
***Example: To test the “Democratic Peace”, I need to pull regime-type data from the dataset maintained by the Polity Project, and interstate conflict data from datasets maintained by the Correlates of War Project***

# The Early Solution

- Bennett and Stam introduced the *EuGene Software* in 2000.
- Through a series of menus, users could simply “point and click” to choose which variables they wanted (variables drawn from preloaded data). Once variables chosen, users could create an analysis dataset.

**Example: EuGene had the Polity Project dataset and the Correlates of War Project conflict dataset preloaded, with the available variables populated into menus. Users could just select the variables they needed using the menus, then hit “Generate Data”**

- Became very popular among peace science scholars who conducted large-*n* analysis (Bennett & Stam 2000 became highly cited paper)



# *EuGene* Inspired Extensions...

**ETH zürich** **University of Essex** **PRIOR** **COST**  
Department of Humanities, Social and Political Sciences | International Conflict Research

**GROW<sup>UP</sup> - Geographical Research On War, Unified Platform**

[View](#) [Read](#) [Download](#) [Code](#) [About](#)

Data Documentation Archive

Level of Aggregation  
 Group-Level Data  Country-Level Data

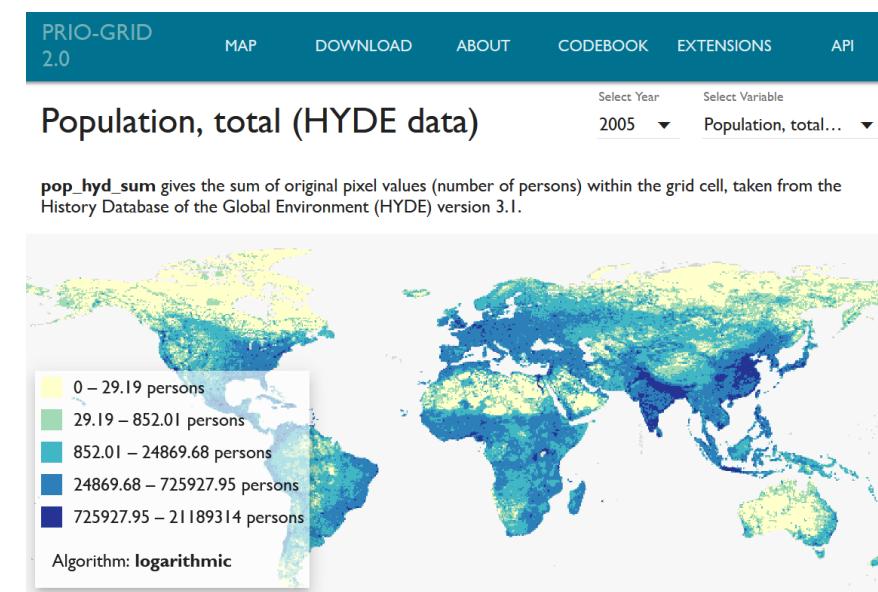
Power Access Data  
 Conflict Data (UCDP ACD, ACD2EPR)  
 Group Hierarchy Data  
 Settlement Area Data (GeoEPR variables)  
 Raster Aggregated Data (GRUMPv1 Population, DMSP Stable Nightlights, G-ECON GCP, GTOPO30 Elevation)  
 Transnational Ethnic Kin (TEK) Data  
 Ethnic Dimensions Data  
 Static PRIOR-GRID Aggregates  
 Yearly PRIOR-GRID Aggregates

GWGROUPID	YEAR	GROUPNAME	COUNTRIES_GWID	COUNTRYNAME	ISACTIVE	ISRELEVANT	GROUPSIZE
201000	1946	Whites	2	United States of America	1	1	0.69100
201000	1947	Whites	2	United States of America	1	1	0.69100
201000	1948	Whites	2	United States of America	1	1	0.69100

## World Economics and Politics Dataverse

A queryable data resource for comparative and international political economy.

[Download Data](#) [Browse FAQs](#)



**xSub** [www.cross-sub.org](http://www.cross-sub.org)

xSub Home About Data Contact Us

# xSub

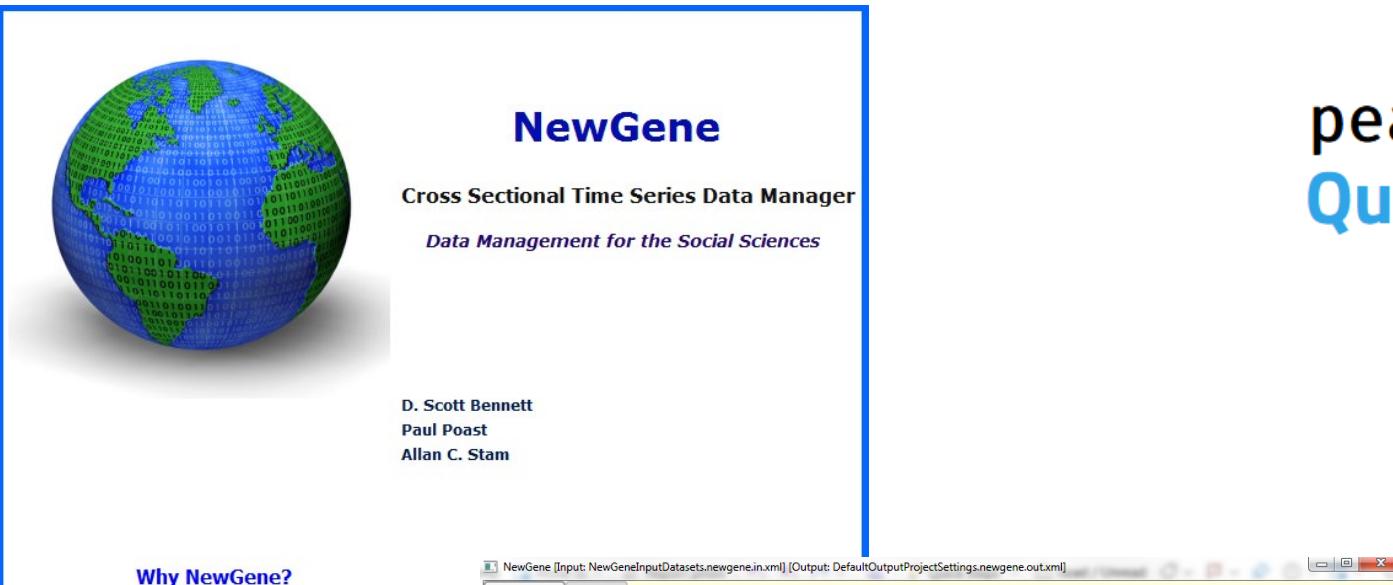
CROSS-NATIONAL DATA ON SUB-NATIONAL VIOLENCE



xSub (cross-sub.org) is a repository of micro-level, subnational event data on armed conflict and political violence around the world. Our goal is to preserve and consolidate multiple open-source data collections in one comprehensive electronic archive, creating a "database of databases" for purposes of research, education, policy-making and evaluation. The archive currently includes data on violence and protests from 139 countries (1942-2016), from 21 data sources, including both large data collections and individual scholars. To facilitate comparisons across countries and sources, xSub organizes these data into common units of analysis, by space (country, province, district, grid cell, electoral constituency) and time (year, month, week). The data are available to download, at no cost, from the data



# ...and motivated upgrades



**NewGene**  
Cross Sectional Time Series Data Manager  
*Data Management for the Social Sciences*

D. Scott Bennett  
Paul Poast  
Allan C. Stam

[Why NewGene?](#)

**Click**

[Continue to NewGene](#)

peacescience: Tools and Data for Quantitative Peace Science

CowPlus Online (Beta)



Home  
Data  
Upload  
Shared  
Guides and FAQ  
Downloads

# But EuGene, its Successors, and its Upgrades all predate LLMs



Compared to EuGene and its successors...

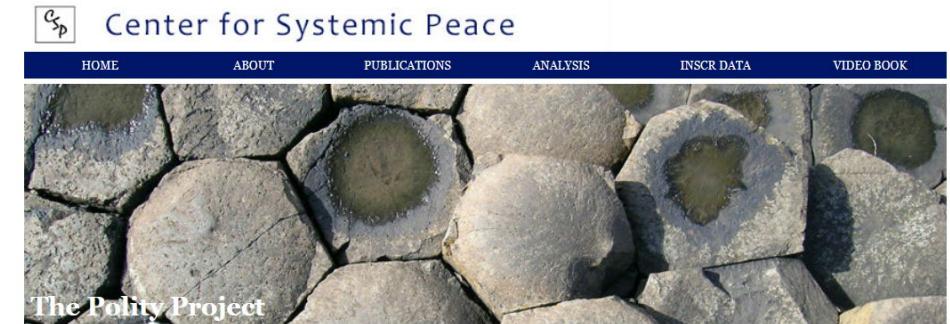
1. LLMs offer more flexibility with data manipulation and management for analysis dataset generation
2. LLMs could provide better processes for replicability

**Let's walk through an Example**

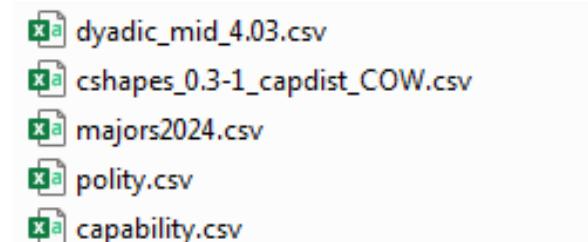
# Example of using an LLM to generate an analysis dataset

- Step 1: Go to websites to acquire the base data you want

**COW** The Correlates of War Project

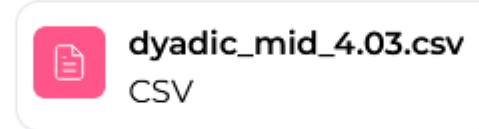
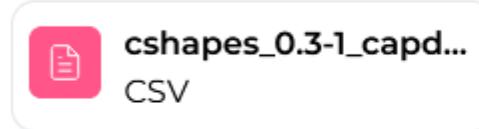
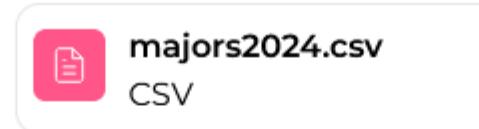


- Step 2: Obtain csv files of all the datasets



# Example Continued

- Step 3: Upload the csv files to the LLM and tell it what you want



Please use the attached datasets to create a dyad-year dataset that reports, for each pair of country-years, the polity2 score for each country, the military personnel for each country, whether a member of a dyad is a major power, the capital-to-capital distance between the two states, and whether they had a MID in that year.

# Example Continued

- Step 4: Obtain the LLM output



Your dyad-year dataset contains the following columns for each country pair and year:

- `ccode1, ccode2, year`
- `polity2_1, polity2_2`: polity2 score for country 1 and country 2
- `milper_1, milper_2`: military personnel for country 1 and 2
- `major_1, major_2`: major power status for 1 and 2
- `capdist`: capital-to-capital distance between states
- `MID`: whether the dyad had a MID in that year (1=yes, 0=no)

If you want this dataset exported, just let me know your preferred format (CSV or Excel).



**csv please**



Your file is ready!

You can download the dyad-year dataset as a CSV here:

[Download dyad\\_year\\_dataset.csv](#)

- [dyad\\_year\\_dataset.csv](#)

# Example Continued

- Step 5: Check the LLM output

ccode1	ccode2	year	polity2_1	polity2_2	milper_1	milper_2	major_1	major_2	capdist	MID
2	200	1902	9	8	111	489	1	1		1
2	200	1903	9	8	106	416	1	1		1
200	2	1902	8	9	489	111	1	1		1
200	2	1903	8	9	416	106	1	1		1
300	345	1913			358	32	1	0		1
345	300	1913			32	358	0	1		1
200	339	1946	10	-9	2053	60	1	0	1893.237	1
339	200	1946	-9	10	60	2053	0	1		1
200	651	1951	10	6	826	77	1	0	3514.525	1
200	651	1952	10	-7	872	81	1	0	3514.525	1
651	200	1951	6	10	77	826	0	1		1
651	200	1952	-7	10	81	872	0	1		1
200	630	1856	3	-10	382	18	1	0		1
200	630	1857	3	-10	272	18	1	0		1
630	200	1856	-10	3	18	382	0	1		1
630	200	1857	-10	3	18	272	0	1		1
200	235	1889	7	-5	275	30	1	0		1

- Step 6: Ask for corrections

It looks like you produced this dataset only for years where a MID took place. Please produce it for all dyad-years since, with years without a MID being coded with a zero

# Example Continued

- Step 7: Obtain New LLM Output

Thank you for clarifying!

To create a *complete* dyad-year dataset (for all possible pairs of states in each year, whether or not a MID occurred), you'll need to:

1. Create all possible country dyads for each year in the coverage of your data.
2. Merge in polity2, military personnel, major power, and capital distance for both members.
3. Merge in the MID event, with zeros for years with no MID.

Your complete dyad-year dataset (including dyads with and without MID events) is ready!

You can download it as a CSV here:

Download [complete\\_dyad\\_year\\_dataset.csv](#)

- [complete\\_dyad\\_year\\_dataset.csv](#)

# Example Continued

- Step 8: Inspect the new data. It looks correct!

ccode1	year	ccode2	polity2_1	polity2_2	milper_1	milper_2	major_1	major_2	capdist	MID
700	1800	710	-6	-6			0	0		0
700	1800	740	-6	-10			0	0		0
700	1800	730	-6	1			0	0		0
700	1800	790	-6	-6			0	0		0
700	1800	800	-6	-10			0	0		0
305	1800	700	-10	-6			0	0		0
305	1800	710	-10	-6			0	0		0
305	1800	390	-10	-10			0	0		0
305	1800	630	-10	-10			0	0		0
305	1800	740	-10	-10			0	0		0
305	1800	730	-10	1			0	0		0
305	1800	600	-10	-6			0	0		0
305	1800	790	-10	-6			0	0		0
305	1800	698	-10	-6			0	0		0
305	1800	365	-10	-10			0	0		0
305	1800	380	-10	-10			0	0		0
305	1800	800	-10	-10			0	0		0
305	1800	640	-10	-10			0	0		0
245	1800	700	-10	-6			0	0		0
245	1800	305	-10	-10			0	0		0
245	1800	710	-10	-6			0	0		0
245	1800	390	-10	-10			0	0		0
245	1800	255	-10	-10			0	0		0
245	1800	630	-10	-10			0	0		0
245	1800	740	-10	-10			0	0		0
245	1800	730	-10	1			0	0		0
245	1800	600	-10	-6			0	0		0
245	1800	790	-10	-6			0	0		0
245	1800	600	-10	-6			0	0		0

# Example Continued

- Step 9: Ask for the code used to make the dataset

Could you also provide stata code that could reproduce this dataset?

Assuming you have the following files:

- polity.csv
- capability.csv
- majors2024.csv
- cshapes\_0.3-1\_capdist\_COW.csv
- dyadic\_mid\_4.03.csv

You should adjust file paths as needed. Below is the Stata code to reproduce your dataset step by step:

Copy

```
* -----
* 1. Get list of states-years from Polity
* -----
import delimited "polity.csv", clear
keep ccode year
duplicates drop
 tempfile states_years
 save `states_years', replace

* -----
* 2. Create all possible dyad-years (no self-dyads)
* -----
use `states_years', clear
rename ccode ccode1
cross using `states_years'
rename ccode ccode2
drop if ccode1 == ccode2

* -----
* 3. Restrict to ccode1 < ccode2 to avoid duplicates
* -----
keep if ccode1 < ccode2

* -----
* 4. Bring in polity2 for both states
* -----
import delimited "polity.csv", clear
keep ccode year polity2
```

# Example Continued

- Step 10: Inspect the code and decide if you want R or Python code too, since LLM is offering it

## Key Points:

- This code generates all possible dyad-years, not just those with a MID.
- Only dyads with `ccode1 < ccode2` are kept, preventing duplicate observations (A-B and B-A in same year).
- Years and countries are based on `polity.csv` coverage. Adjust as needed for your domain.
- MID is coded as 1 if a dispute occurred, 0 otherwise.

Let me know if you want this process in R or Python, or additional features included!

**The entire 10 step process took just a few minutes (longest parts are downloading the datasets and inspecting the data)**

# The advantages & disadvantages of using LLM to conduct data management

## Advantages

- **Low learning curve:** Doesn't require you to learn how to work with a new platform or work with particular code. You can just talk to the AI.
- **Flexible:** Not constrained by data offered by platform. Just bring what you want and see if the AI can work with it.
- **Replicable:** It can give you the code that conducted all the procedures

## Possible Disadvantages

- Data not preloaded. You have to acquire the data from the websites.
  - **Counter argument:** But is that really an issue? Is it really a burden to download data from open access websites? We do this all the time (i.e. dataverse)
- It can make errors
  - **Counter argument:** But you should have always been checking data to make sure it is what you want, even back with EuGene (see Bennett 2011). Now easier to go back and say to LLM, “this isn’t quite what I want.”
- Personal data security
  - **Counter argument:** Understandable regarding open LLMs (hence, UChicago made Phoenix AI), but not really a problem if you are working with open access data (which is what EuGene/other platforms are doing anyhow).

# Thoughtful Dataset Creation

- Interfacing with an AI may force scholars to think more carefully about which data they need and why.
- This was a common concern with EuGene-type platforms

Conflict Management and Peace Science  
© The Author(s). 2011. Reprints and permissions:  
<http://www.sagepub.co.uk/journalsPermissions.nav>  
[DOI:10.1177/0738942111413055]  
Vol 28(4): 315–330

cmps

Annual Presidential Address  
Is EUGene a Collective Bad?

D. SCOTT BENNETT  
The Pennsylvania State University

“I argue that EUGene is still a valuable tool, although, like any tool, it can be misused...The use of more sophisticated methods makes tools such as EUGene particularly valuable, as they **make it quicker** to perform routine data preparation tasks, leaving more time for analysis and data interpretation”

**Maybe not good to make data preparation  
overly “quick”?**

**Chapter 18: Think locally, regress globally:  
making the most of conventional IR data**

Carlos Felipe Balcazar and Matt Malis 

Category: Chapter

Published: 05 Aug 2022

Page Range: 317–340

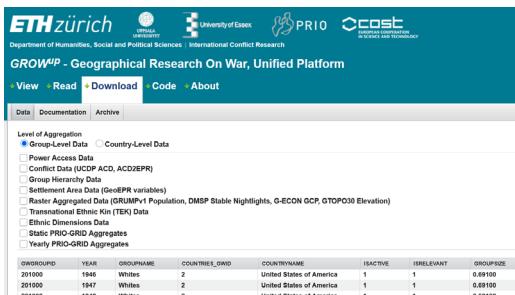
Collection: [Political Science and Public Policy 2022](#)

DOI: <https://doi.org/10.4337/9781839101014.00029>

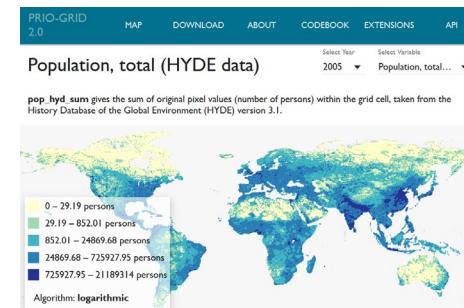
“Our overarching claim is that the increasing accessibility of conventional IR data, and tools for analyzing these data, require that researchers be ever more careful about how they approach their analyses... **treat your data as if it were costly to obtain**, with outcomes unknown ex-ante, and invest the time upfront to think through the myriad decisions which will have to be made throughout the process of analysis”

# Do LLMs render other online platforms obsolete?

- While we no longer need EuGene-like platforms, organizations still need a means of storing and providing user access to data.
- Online platforms could remain useful as data repositories



ETH zürich University of Essex PRIO COST  
Department of Humanities, Social and Political Sciences International Conflict Research  
GROWWP - Geographical Research On War, Unified Platform  
View Read Download Code About  
Data Documentation Archive  
Level of Aggregation: Group-Level Data Country-Level Data  
Power Access Data  
Conflict Data (UCDP ACED, ACED2PR)  
Group Hierarchy Data  
Settlement Area Data (GGRWP+ Population, DMSP Stable Nightlights, G-ECON GCP, GTOPO30 Elevation)  
Transnational Ethnic Kin (TEK) Data  
Ethnic Dimensions Data  
Static PRIO-GRID Aggregates  
Yearly PRIO-GRID Aggregates  
GWGROUP YEAR GROUPNAME COUNTRIES\_GWD COUNTRYNAME ISACTIVE ISRELEVANT GROUPSIZE  
201000 1946 Whites 2 United States of America 1 1 0.69100 1  
201000 1947 Whites 2 United States of America 1 1 0.69100 1  
201000 1948 Whites 2 United States of America 1 1 0.69100 1



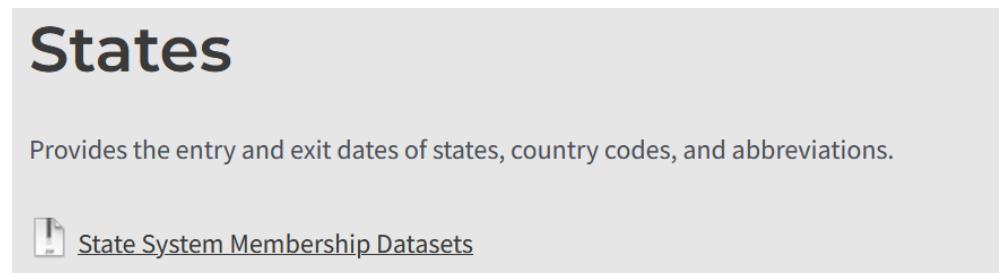
- Dataverse and ICPSR still valuable



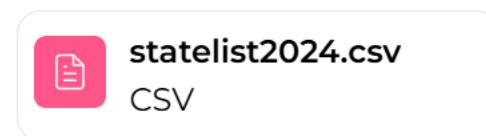
# APPENDIX

# Example A1: Create Empty Country-Year Dataset

- Step 1: Download State Membership Dataset



- Step 2: Upload to LLM & request a country-year dataset



Can you use the information in the attached dataset to create a country-year dataset?

# Example A1 continued

- Step 3: Receive LLM's response

Let's inspect the contents of `statelist2024.csv` to understand how it can be used—possibly in combination with the other dataset—to create a full country-year dataset. The `statelist2024.csv` file describes the time spans when each state (country) existed:

- `stateabb`, `ccode`, `statename`: Country identifiers
- `styear`, `stmonth`, `stday`: Start date of the country's existence
- `endyear`, `endmonth`, `endday`: End date
- `version`: Data version

To create a country-year dataset, we need to generate one row for every year a country existed, based on the `styear` and `endyear`.

Would you like to use just the main identifiers (`ccode`, `stateabb`, `statename`, `year`), or include additional columns (dates, `version`)?

I'll create the country-year dataset using the time intervals from `statelist2024.csv` with main identifiers. A new country-year dataset has been created using the information from `statelist2024.csv`. Each row now represents a country (`ccode`, `stateabb`, `statename`) in a specific year (`year`) during its existence.

If you'd like to view more rows, add variables, or export this dataset, let me know your

# Example A1 continued

- Step 4: Answer its question about variables to include. See response

please just use the main identifiers

Here is the country-year dataset using only the main identifiers from your attached file:

ccode	stateabb	statenme	year
2	USA	United States of America	1816
2	USA	United States of America	1817
2	USA	United States of America	1818
...	...	...	...

Columns included:

- ccode
- stateabb
- statenme
- year

Would you like this dataset exported as a CSV file? If so, please specify a filename.

# Example A1 continued

- Step 5: Respond “yes” and obtain your dataset

yes, please. Call it state-year

Your country-year dataset is ready!  
You can download the CSV file here: state-year.csv

- [state-year.csv](#)

- Step 6: Download and inspect the dataset. Looks correct!

A	B	C	D	
1	ccode	stateabb	statename	year
2	2	USA	United Sta	1816
3	2	USA	United Sta	1817
4	2	USA	United Sta	1818
5	2	USA	United Sta	1819
6	2	USA	United Sta	1820
7	2	USA	United Sta	1821
8	2	USA	United Sta	1822
9	2	USA	United Sta	1823
10	2	USA	United Sta	1824
11	2	USA	United Sta	1825
12	2	USA	United Sta	1826

5088	220	FRN	France	1938
5089	220	FRN	France	1939
5090	220	FRN	France	1940
5091	220	FRN	France	1941
5092	220	FRN	France	1942
5093	220	FRN	France	1944
5094	220	FRN	France	1945
5095	220	FRN	France	1946

Even  
captured gap!

# Example A2: Create an Empty Dyad-year dataset

## Step 1: Create the Dataset

Can you now make it into a dyad-year dataset?

To create a dyad-year dataset, I will pair every country with every other country for each year, so that each row represents a unique country pair (dyad) in a given year. The columns will be:

- ccode1, stateabb1, statenme1: Identifiers for the first country
- ccode2, stateabb2, statenme2: Identifiers for the second country
- year: Year of the dyad

I'll create only unique dyads (each pair is represented once per year, ignoring order) and exclude self-pairs. Let's generate this dataset. The dyad-year dataset is created!

Each row represents a unique country pair for a particular year, using only the main identifiers.

Would you like this dataset exported as a CSV as well? If so, shall I call it dyad-year.csv?



yes please

## Step 2: Download and Inspect Data

year	ccode1	stateabb1	statenme1	ccode2	stateabb2	statenme2
1816	2 USA	United Sta	United States	200 UKG	United Kingdom	
1816	2 USA	United Sta	United States	210 NTH	Netherlands	
1816	2 USA	United Sta	United States	220 FRN	France	
1816	2 USA	United Sta	United States	225 SWZ	Switzerland	
1816	2 USA	United Sta	United States	230 SPN	Spain	
1816	2 USA	United Sta	United States	235 POR	Portugal	
1816	2 USA	United Sta	United States	245 BAV	Bavaria	
1816	2 USA	United Sta	United States	255 GMY	Germany	
1816	2 USA	United Sta	United States	267 BAD	Baden	
1816	2 USA	United Sta	United States	269 SAX	Saxony	
1816	2 USA	United Sta	United States	271 WRT	Wuerttemburg	
1816	2 USA	United Sta	United States	273 HSE	Hesse Electoral	
1816	2 USA	United Sta	United States	275 HSG	Hesse Grand Ducal	
1816	2 USA	United Sta	United States	300 AUH	Austria-Hungary	
1816	2 USA	United Sta	United States	325 ITA	Italy	
1816	2 USA	United Sta	United States	327 PAP	Papal States	
1816	2 USA	United Sta	United States	329 SIC	Two Sicilies	
1816	2 USA	United Sta	United States	337 TUS	Tuscany	
1816	2 USA	United Sta	United States	365 RUS	Russia	

Your dyad-year dataset is ready!

You can download the CSV file here: [dyad-year.csv](#)