

using `r` I would like to narrow rows to only 50 states and make columns the healthcare expense for each state for the years 2013-2023 only for this csv
<https://raw.githubusercontent.com/KieraG2026/U.S.-Spending-Categories-by-State-1997-2023/main/Expense%20Categories%20by%20State%201997-2023.csv>

Here is a reproducible **R (tidyverse)** solution that:

1. **Loads the CSV** from your GitHub URL
2. **Keeps only the 50 U.S. states** (excludes DC, territories, etc.)
3. **Selects only the healthcare expenditure columns for years 2013–2023**
4. **Produces a clean data frame with one row per state**

⚠ I haven't seen the exact column names yet (the header didn't show up in the preview), so I'll show you how to *inspect* and then *select* the right columns.

✅ Step-by-Step R Code

```
r
```

[Copy code](#)

```
# load libraries
library(tidyverse)
```

```
# read the data
df <- read_csv("https://raw.githubusercontent.com/KieraG2026/U.S.-Spending

# inspect column names
colnames(df)
```

Run that once to see how the years are labeled (e.g., "Healthcare_2013" , "2014 Healthcare" , "Health 2015" , etc.).

Once you know the exact pattern for healthcare columns, you can select them.

✅ Filtering to 50 States Only

Most spreadsheets have a *location* column like "State" or "Location" . Here we define the vector of 50 US states:

```
r
```

 Copy code

```
# define the 50 US states
us_states <- c("Alabama", "Alaska", "Arizona", "Arkansas", "California", "Colorado",
               "Connecticut", "Delaware", "Florida", "Georgia", "Hawaii", "Idaho",
               "Illinois", "Indiana", "Iowa", "Kansas", "Kentucky", "Louisiana",
               "Maine", "Maryland", "Massachusetts", "Michigan", "Minnesota",
               "Mississippi", "Missouri", "Montana", "Nebraska", "Nevada",
               "New Hampshire", "New Jersey", "New Mexico", "New York",
               "North Carolina", "North Dakota", "Ohio", "Oklahoma", "Oregon",
               "Pennsylvania", "Rhode Island", "South Carolina", "South Dakota",
               "Tennessee", "Texas", "Utah", "Vermont", "Virginia", "Washington",
               "West Virginia", "Wisconsin", "Wyoming")
```

✓ Select Healthcare for 2013–2023

Once you know the column name pattern, here's an example **dplyr** pipeline.

If the healthcare columns are literally named like Healthcare_2013 ,
Healthcare_2014 , ..., Healthcare_2023 :

r

 Copy code

```
df_healthcare_50 <- df %>%  
  # keep only rows where the state is one of the 50 states  
  filter(Location %in% us_states) %>%  
  
  # select only the state and healthcare columns for 2013–2023  
  select(Location,  
         Healthcare_2013:Healthcare_2023)
```

If the healthcare columns are named like 2013 Healthcare , 2014
Healthcare , etc.:

r

 Copy code

```
df_healthcare_50 <- df %>%  
  filter(Location %in% us_states) %>%  
  select(Location, starts_with("2013 Healthcare"), starts_with("2014 Healthc  
    starts_with("2015 Healthcare"), starts_with("2016 Healthcare"),  
    starts_with("2017 Healthcare"), starts_with("2018 Healthcare"),  
    starts_with("2019 Healthcare"), starts_with("2020 Healthcare"),
```

```
starts_with("2021 Healthcare"), starts_with("2022 Healthcare"),
starts_with("2023 Healthcare"))
```

Result

- `df_healthcare_50` will have **50 rows**, one per state.
- It will contain only healthcare expense columns for **2013–2023**.

You can now:

```
r
```

 Copy code

```
# view a preview
```

```
print(df_healthcare_50)
```

```
# save to disk
```

```
write_csv(df_healthcare_50, "healthcare_2013_2023_states.csv")
```

Notes

- Make sure the *state name column* in your dataset matches what you use (Location above). Use `colnames(df)` to verify.
 - If your healthcare columns have a different pattern, let me know the column names and I can tailor the `select()` call.
-

If you want, I can help you with **plotting trends** or **reshaping** the data (e.g., to long format) after this step.



prefer to name columns just by years 2013-2023

Perfect! We can rename the healthcare columns to just the years 2013–2023 while keeping one row per state. Here's a clean **R solution** assuming the column names contain "Healthcare" somewhere in them.

```
r
```

 Copy code

```
library(tidyverse)
```

```
# 1. Read the CSV
```

```
df <- read_csv("https://raw.githubusercontent.com/KieraG2026/U.S.-Spending
```

```
# 2. Define the 50 US states
```

```
us_states <- c("Alabama", "Alaska", "Arizona", "Arkansas", "California", "Colorado",  
              "Connecticut", "Delaware", "Florida", "Georgia", "Hawaii", "Idaho",  
              "Illinois", "Indiana", "Iowa", "Kansas", "Kentucky", "Louisiana",  
              "Maine", "Maryland", "Massachusetts", "Michigan", "Minnesota",  
              "Mississippi", "Missouri", "Montana", "Nebraska", "Nevada",  
              "New Hampshire", "New Jersey", "New Mexico", "New York",  
              "North Carolina", "North Dakota", "Ohio", "Oklahoma", "Oregon",
```

```
"Pennsylvania", "Rhode Island", "South Carolina", "South Dakota",  
"Tennessee", "Texas", "Utah", "Vermont", "Virginia", "Washington",  
"West Virginia", "Wisconsin", "Wyoming")
```

```
# 3. Filter to 50 states and select healthcare columns 2013-2023
```

```
df_healthcare_50 <- df %>%  
  filter(Location %in% us_states) %>%  
  select(Location, contains("Healthcare")) %>%  
  # 4. Keep only columns for 2013-2023  
  select(Location, matches("201[3-9]|202[0-3]")) %>%  
  # 5. Rename healthcare columns to just years  
  rename_with(~ str_extract(., "\\d{4}"), -Location)
```

```
# 6. Preview
```

```
print(df_healthcare_50)
```

✓ What this does

1. `filter(Location %in% us_states)` keeps only the 50 U.S. states.
2. `select(Location, contains("Healthcare"))` grabs all healthcare-related columns.
3. `matches("201[3-9]|202[0-3]")` ensures only 2013–2023 columns remain.
4. `rename_with(~ str_extract(., "\\d{4}"), -Location)` renames columns to just the year.

The final `df_healthcare_50` will have:

Location	2013	2014	...	2023
Alabama
Alaska
...

If you want, I can also give a **version that automatically pivots this to long format** so each row is `State-Year-HealthcareExpense` —which is perfect for plotting trends.

Do you want me to do that?

