PS 4: Web Scraping and Data Visualization

Sihan Lou

2024-02-07

Due 02/07 at 5:00PM Central.

"This submission is my work alone and complies with the 30538 integrity policy." Add your initials to indicate your agreement: SL

Github Classroom Assignment Setup and Submission Instructions

1. **Accepting and Setting up the PS4 Assignment Repository**
   - Each student must individually accept the repository for the problem set from Github Classroom ("ps4") – https://classroom.github.com/a/hWhtcHqH
     - You will be prompted to select your cnetid from the list in order to link your Github account to your cnetid.
     - If you can't find your cnetid in the link above, click "continue to next step" and accept the assignment, then add your name, cnetid, and Github account to this Google Sheet and we will manually link it: https://rb.gy/9u7fb6
   - If you authenticated and linked your Github account to your device, you should be able to clone your PS4 assignment repository locally.
   - Contents of PS4 assignment repository:
     - `ps4_template.qmd`: this is the Quarto file with the template for the problem set. You will write your answers to the problem set here.
2. **Submission Process**:
   - Knit your completed solution `ps4.qmd` as a pdf `ps4.pdf`.
     - Your submission does not need runnable code. Instead, you will tell us either what code you ran or what output you got.
   - To submit, push `ps4.qmd` and `ps4.pdf` to your PS4 assignment repository. Confirm on Github.com that your work was successfully pushed.

Grading

- You will be graded on what was last pushed to your PS4 assignment repository before the assignment deadline
- Problem sets will be graded for completion as: {missing (0%); - (incomplete, 50%); + (excellent, 100%)}
  - The percent values assigned to each problem denote how long we estimate the problem will take as a share of total time spent on the problem set, not the points they are associated with.
- In order for your submission to be considered complete, you need to push both your `ps4.qmd` and `ps4.pdf` to your repository. Submissions that do not include both files will automatically receive 50% credit.

```python
import pandas as pd
import altair as alt
import time

import warnings
warnings.filterwarnings('ignore')
alt.renderers.enable("png")
```

```
RendererRegistry.enable('png')
```

Step 1: Develop initial scraper and crawler

```python
import requests
from bs4 import BeautifulSoup
from urllib.parse import urljoin

headers = {"User-Agent": "Mozilla/5.0"}

url = "https://oig.hhs.gov/fraud/enforcement/"
response = requests.get(url, headers=headers)
soup = BeautifulSoup(response.text, "html.parser")

actions = soup.find_all("div", class_="usa-card__heading")

rows = []
for a in soup.select("main h2 a"):
    title = a.get_text(strip=True)
    link = urljoin(url, a.get("href", ""))

    container = a.find_parent("li")
    if container is None:
        continue

    date_tag = container.select_one("span.padding-right-105")
    date = date_tag.get_text(strip=True) if date_tag else None

    category = container.find("li",class_="usa-tag").get_text(strip=True)

    rows.append({"title": title, "date": date, "category": category, "link":
 ↪  link})
```

```
df = pd.DataFrame(rows)
print("n_rows =", len(df))
df.head()
```

```
n_rows = 20
```

	title	date	category	lin
0	Houston Transplant Doctor Indicted For Making ...	February 5, 2026	Criminal and Civil Actions	htt
1	MultiCare Health System to Pay Millions to Set...	February 4, 2026	Criminal and Civil Actions	htt
2	Brooklyn Banker Pleads Guilty to Laundering Pr...	February 3, 2026	COVID-19	htt
3	Delafield Man Sentenced to 18 Months' Imprison...	February 3, 2026	Criminal and Civil Actions	htt
4	Former NFL Player Convicted for $197M Medicare...	February 3, 2026	Criminal and Civil Actions	htt

Step 2: Making the scraper dynamic

1. Turning the scraper into a function

- a. Pseudo-Code FUNCTION scrape_actions(start_year, start_month, run_scraper):

  IF start_year < 2013 THEN PRINT "Please restrict to year >= 2013." RETURN END IF

  SET filename TO "enforcement_actions_start_year_start_month.csv"

  IF run_scraper IS FALSE THEN LOAD filename AS dataframe RETURN dataframe END IF

  SET start_date TO the first day of (start_year, start_month) SET page TO 1 SET rows TO empty list

  WHILE TRUE DO

  ```
  IF page == 1 THEN
      SET page_url TO base enforcement actions URL
  ELSE
      SET page_url TO base enforcement actions URL + "?page=" + page
  END IF

  FOR EACH link_element IN title_links DO

      SET title TO the text inside link_element
      SET link TO the full URL built from the link_element href
  ```

```
        SET container TO the parent element that contains the full action
        entry
        IF container does not exist THEN
            CONTINUE
        END IF

        SET date TO the date text found inside container (if missing, set
        to NULL)

        SET category_tags TO all category tag elements inside container
        SET categories TO empty list
        FOR EACH tag IN category_tags DO
            SET tag_text TO text inside tag
            IF tag_text is not empty AND tag_text not already in
            categories THEN
                APPEND tag_text TO categories
            END IF
        END FOR

        APPEND {title, date, category, link} TO rows

    END FOR

    WAIT 1 second
    SET page TO page + 1
```

END WHILE

CONVERT rows into a dataframe CONVERT dataframe date column into a datetime type FILTER dataframe to keep only rows with date >= start_date

SAVE dataframe to filename RETURN dataframe

END FUNCTION

- b. Create Dynamic Scraper

```python
def web_scraper(start_year, start_month):
    if start_year < 2013:
        print("Only enforcement actions after 2013 are listed")
        return None

    page = 1
    data = []
    continue_scraping = True
```

```python
while continue_scraping:
    url = f"https://oig.hhs.gov/fraud/enforcement/?page={page}"
    response = requests.get(url)
    soup = BeautifulSoup(response.text, "html.parser")

    cards = soup.find_all("li", class_="usa-card")

    if len(cards) == 0:
        break

    for card in cards:
        a = card.find("a")
        title = a.get_text(strip=True)
        link = "https://oig.hhs.gov" + a["href"]

        date_text = card.find(
            "span", class_="text-base-dark padding-right-105"
        ).get_text(strip=True)

        parts = date_text.split()
        year = int(parts[2])
        month_map = {
            "January": 1, "February": 2, "March": 3, "April": 4,
            "May": 5, "June": 6, "July": 7, "August": 8,
            "September": 9, "October": 10, "November": 11, "December":
             ↪  12}
        month = month_map[parts[0]]

        category = card.find(
            "li", class_="usa-tag"
        ).get_text(strip=True)

        if (year < start_year) or (year == start_year and month <
         ↪  start_month):
            continue_scraping = False
            break

        data.append({
            "title": title,
            "date": date_text,
            "category": category,
            "link": link
            })

        time.sleep(1)
        page += 1

    df = pd.DataFrame(data)
    df.to_csv(
        f"enforcement_actions_{start_year}_{start_month}.csv",
        index=False
    )

    return df

web_scraper(2024, 1)
```

	title	date	category
0	Houston Transplant Doctor Indicted For Making ...	February 5, 2026	Criminal and Civil Actions
1	MultiCare Health System to Pay Millions to Set...	February 4, 2026	Criminal and Civil Actions
2	Brooklyn Banker Pleads Guilty to Laundering Pr...	February 3, 2026	COVID-19
3	Delafield Man Sentenced to 18 Months' Imprison...	February 3, 2026	Criminal and Civil Actions
4	Former NFL Player Convicted for $197M Medicare...	February 3, 2026	Criminal and Civil Actions
...	...	...	...
1782	Athletico Management, PT Network, and Dynamic ...	January 4, 2024	Fraud Self-Disclosures
1783	Recover-Care Plaza West Care Center Agreed to ...	January 4, 2024	Fraud Self-Disclosures
1784	Maury County Caregiver Charged With Financial ...	January 4, 2024	State Enforcement Agencies
1785	Laredo Resident Admits To Impersonating Licens...	January 3, 2024	Criminal and Civil Actions
1786	Former Nurse Aide Indicted In Death Of Clarksv...	January 3, 2024	State Enforcement Agencies

```
df_2024 = pd.read_csv("enforcement_actions_2024_1.csv")
print("n_rows =", len(df_2024))
df_2024.head()
```

```
n_rows = 1787
```

	title	date	category	lin
0	Houston Transplant Doctor Indicted For Making ...	February 5, 2026	Criminal and Civil Actions	htt
1	MultiCare Health System to Pay Millions to Set...	February 4, 2026	Criminal and Civil Actions	htt
2	Brooklyn Banker Pleads Guilty to Laundering Pr...	February 3, 2026	COVID-19	htt
3	Delafield Man Sentenced to 18 Months' Imprison...	February 3, 2026	Criminal and Civil Actions	htt

	title	date	category	lin
4	Former NFL Player Convicted for $197M Medicare…	February 3, 2026	Criminal and Civil Actions	htt

- c. Test Your Code

```
web_scraper(2022, 1)
df_2022 = pd.read_csv("enforcement_actions_2022_1.csv")
print("n_rows =", len(df_2022))
df_2022.head()
```

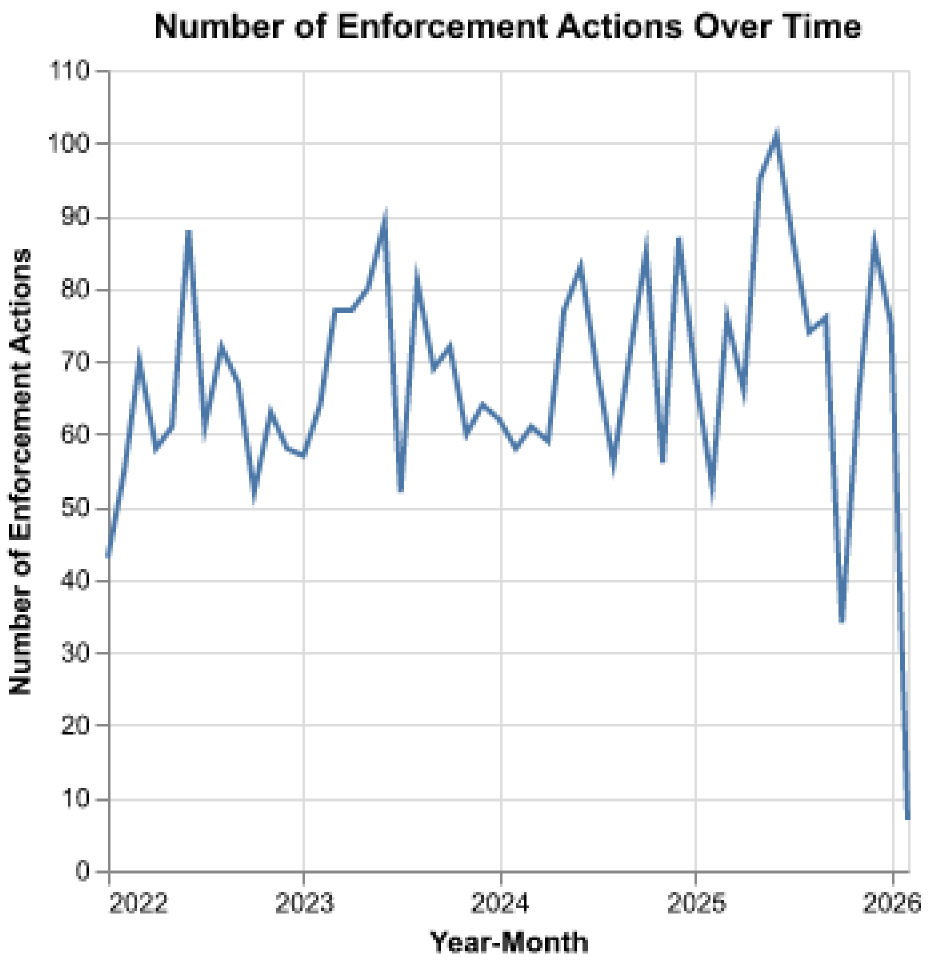
```
n_rows = 3377
```

	title	date	category	lin
0	Houston Transplant Doctor Indicted For Making …	February 5, 2026	Criminal and Civil Actions	htt
1	MultiCare Health System to Pay Millions to Set…	February 4, 2026	Criminal and Civil Actions	htt
2	Brooklyn Banker Pleads Guilty to Laundering Pr…	February 3, 2026	COVID-19	htt
3	Delafield Man Sentenced to 18 Months' Imprison…	February 3, 2026	Criminal and Civil Actions	htt
4	Former NFL Player Convicted for $197M Medicare…	February 3, 2026	Criminal and Civil Actions	htt

Step 3: Plot data based on scraped data

1. Plot the number of enforcement actions over time

```
df_2022["date"] = pd.to_datetime(df_2022["date"])
df_2022["year_month"] = df_2022["date"].dt.to_period("M").dt.to_timestamp()
plot1 = alt.Chart(df_2022).mark_line().encode(
    x=alt.X("year_month:T", title="Year-Month"),
    y=alt.Y("count()", title="Number of Enforcement Actions")
).properties(
    title="Number of Enforcement Actions Over Time"
)
plot1
```

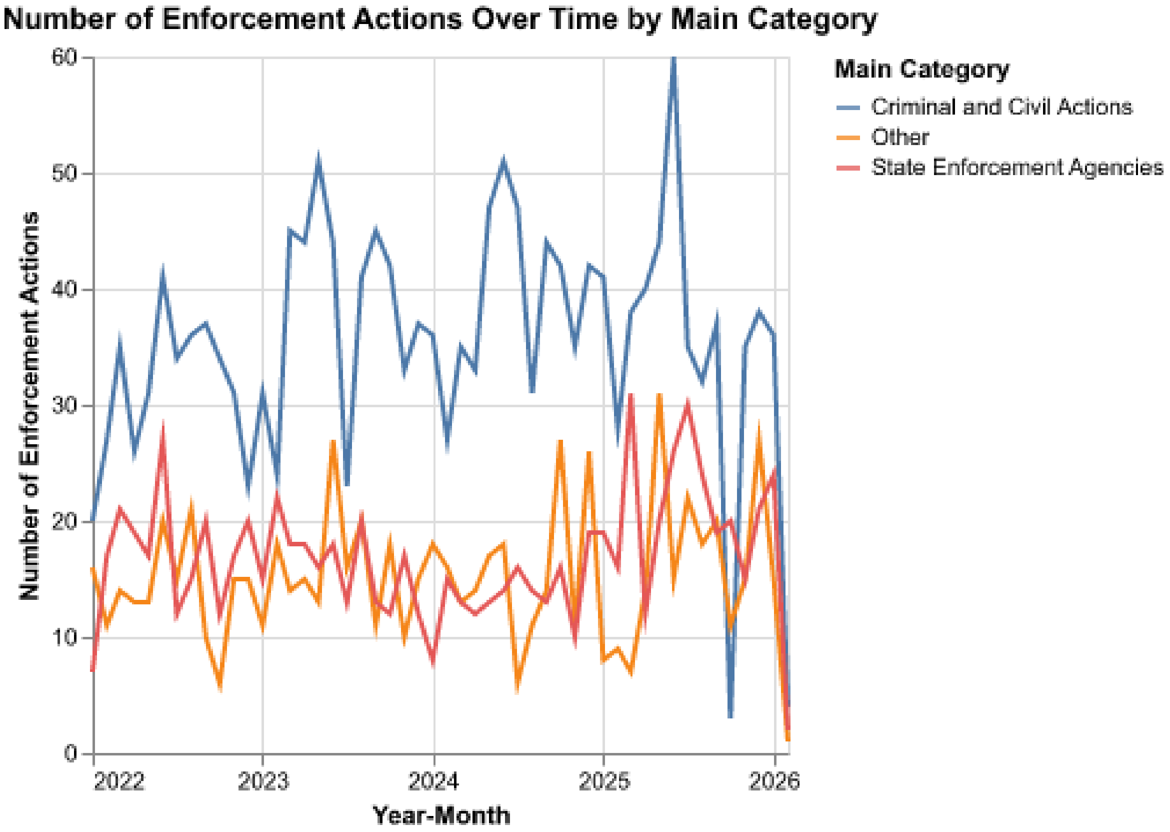


2. Plot the number of enforcement actions categorized:

- based on "Criminal and Civil Actions" vs. "State Enforcement Agencies"

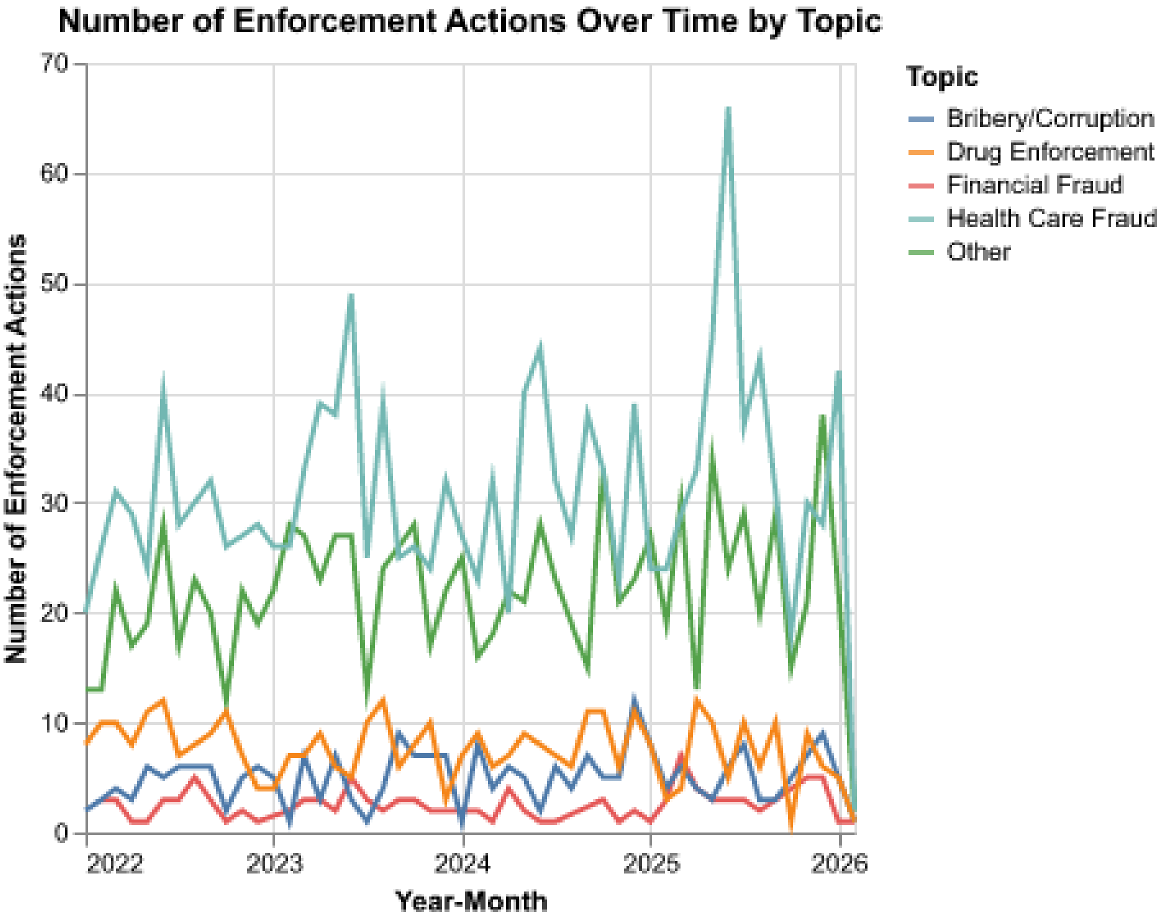
```
def get_main_category(cat):
    cat = "" if pd.isna(cat) else str(cat)
    if "Criminal and Civil Actions" in cat:
        return "Criminal and Civil Actions"
    if "State Enforcement Agencies" in cat:
        return "State Enforcement Agencies"
    return "Other"

df_2022["main_category"] = df_2022["category"].apply(get_main_category)
```

```
plot2 = alt.Chart(df_2022).mark_line().encode(
    x=alt.X("year_month:T", title="Year-Month"),
    y=alt.Y("count()", title="Number of Enforcement Actions"),
    color=alt.Color("main_category:N", title="Main Category")
).properties(
    title="Number of Enforcement Actions Over Time by Main Category"
)
plot2
```



- based on five topics

```
def assign_topic(title):
    t = "" if pd.isna(title) else str(title).lower()

    # Drug Enforcement
    if any(k in t for k in ["opioid", "fentanyl", "controlled substance",
     ↪   "drug", "pill", "pharmacy"]):
        return "Drug Enforcement"

    # Bribery/Corruption
    if any(k in t for k in ["brib", "corrupt", "kickback", "extortion",
     ↪  "gratuity"]):
        return "Bribery/Corruption"

    # Financial Fraud
    if any(k in t for k in ["bank", "financial", "loan", "mortgage", "wire
     ↪  fraud", "securities",
                            "tax", "money laundering"]):
        return "Financial Fraud"

    # Health Care Fraud
    if any(k in t for k in ["medicare", "medicaid", "health care",
     ↪  "healthcare", "hospital", "clinic",
                            "physician", "nursing", "home health", "hospice",
                             ↪  "billing", "dme",
                            "lab", "ambulance"]):
        return "Health Care Fraud"

    return "Other"

df_2022["topic"] = df_2022["title"].apply(assign_topic)
plot3 = alt.Chart(df_2022).mark_line().encode(
    x=alt.X("year_month:T", title="Year-Month"),
    y=alt.Y("count()", title="Number of Enforcement Actions"),
    color=alt.Color("topic:N", title="Topic")
).properties(
    title="Number of Enforcement Actions Over Time by Topic"
)
plot3
```


Number of Enforcement Actions Over Time by Topic
Number of Enforcement Actions
0
10
20
30
40
50
60
70
2022
2023
2024
2025
2026
Year-Month
Topic
Bribery/Corruption
Drug Enforcement
Financial Fraud
Health Care Fraud
Other