AB_Testing_and_Scraping

July 24, 2023

1 Task 1: Experimental design and A/B testing

Task: Design and analyze A/B tests for a hypothetical scenario.

hypothesize that version B will lead to a higher conversion rate compared to version A.

[1]:
```
import pandas as pd
import numpy as np
```

1.1 Creating Random Data for A and B Page Versions

[43]:
```
np.random.seed(42)

# Page/Version A data
A_conversion = np.random.binomial(n=1, p=0.02, size=500)
A_click_through = np.random.binomial(n=1, p=0.40, size=500)
A_bounce = np.random.binomial(n=1, p=0.80, size=500)
A_order_value = np.random.normal(loc=50, scale=10, size=500)

# Page/Version B data
B_conversion = np.random.binomial(n=1, p=0.15, size=500)
B_click_through = np.random.binomial(n=1, p=0.45, size=500)
B_bounce = np.random.binomial(n=1, p=0.75, size=500)
B_order_value = np.random.normal(loc=60, scale=12, size=500)

df = pd.DataFrame({
    'Version': ['A'] * 500 + ['B'] * 500,
    'Conversion': np.concatenate([A_conversion, B_conversion]),
    'Click_through': np.concatenate([A_click_through, B_click_through]),
    'Bounce': np.concatenate([A_bounce, B_bounce]),
    'Order_Value': np.concatenate([A_order_value, B_order_value]),
})

df
```

[43]:

	Version	Conversion	Click_through	Bounce	Order_Value
0	A	0	1	1	25.005943
1	A	0	0	1	72.909426

```
2         A           0              0       0    36.104275
3         A           0              1       1    33.546013
4         A           0              1       0    60.225704
..      …           …              …       …      …
995       B           0              0       1    48.425546
996       B           0              0       1    66.071587
997       B           0              1       0    51.260734
998       B           0              1       1    85.980673
999       B           0              0       1    74.286583

[1000 rows x 5 columns]
```

[41]:
```
df.describe()
```

[41]:

	Conversion	Click_through	Bounce	Order_Value
count	1000.000000	1000.000000	1000.000000	1000.000000
mean	0.079000	0.404000	0.765000	55.200524
std	0.269874	0.490943	0.424211	12.300895
min	0.000000	0.000000	0.000000	20.786495
25%	0.000000	0.000000	1.000000	46.768892
50%	0.000000	0.000000	1.000000	54.572486
75%	0.000000	1.000000	1.000000	63.593777
max	1.000000	1.000000	1.000000	107.114852

1.2 Hypothesis Testing

Null Hypothesis (H0): There is no significant difference between the conversion rate of Version A and Version B.

Alternative Hypothesis (Ha): There is a significant difference between the conversion rate of Version A and Version B.

1.2.1 Hypothesis between conversion rates

[51]:
```
from scipy.stats import ttest_ind

# calculate t_test between conversion rates
t_stat, p_value = ttest_ind(df[df.Version=='A']['Conversion'], df[df.
 ↪Version=='B']['Conversion'])
print(f't_stat: {t_stat}\np_value:{p_value}', end='\n\n')

# significance level
alpha = 0.05
if p_value < alpha:
    print("There is significant evidence to reject the null hypothesis.")
    print("Version B leads to a higher conversion rate compared to Version A.")
else:
    print("There is not enough evidence to reject the null hypothesis.")
```

```
    print("There is no significant difference in conversion rates between␣
 ↪Version A and Version B.")
```

```
t_stat: -7.588276816916516
p_value:7.418185923338823e-14

There is significant evidence to reject the null hypothesis.
Version B leads to a higher conversion rate compared to Version A.
```

1.2.2 Confidence Interval Function

[56]:
```
# Confidence Interval
import math

def Confidence_Interval(columns):
    columns_mean = columns.mean()
    columns_std = columns.std()

    columns_standard_error = columns_std/len(columns)
    columns_margin_error = columns_standard_error/2

    confidence_level = 0.95

    upper_bound = columns_mean + confidence_level * (columns_std / math.
 ↪sqrt(10))
    lower_bound = columns_mean - confidence_level * (columns_std / math.
 ↪sqrt(10))

    confidence_intervals = pd.DataFrame({
        'Sample Size': columns.count(),
        'Sample Mean': columns_mean,
        'Standard Error': columns_standard_error,
        'Margin of Error': columns_margin_error,
        'Lower Bound (95% CI)': lower_bound,
        'Upper Bound (95% CI)': upper_bound
    })

    return confidence_intervals
```

[87]:
```
print("Conversion Rate Confidence Interval Values\n")
Confidence_Interval(df.loc[df.Version=='A', ['Conversion']])
```

```
Conversion Rate Confidence Interval Values
```

[87]:

	Sample Size	Sample Mean	Standard Error	Margin of Error	\
Conversion	500	0.016	0.000251	0.000126	

```
             Lower Bound (95% CI)  Upper Bound (95% CI)
Conversion             -0.021733              0.053733
```

[86]:
```
Confidence_Interval(df.loc[df.Version=='B', ['Conversion']])
```

[86]:
```
             Sample Size  Sample Mean  Standard Error  Margin of Error  \
Conversion           500        0.142        0.000699         0.000349

             Lower Bound (95% CI)  Upper Bound (95% CI)
Conversion               0.037035              0.246965
```

1.2.3 Hypothesis between Click Through

[88]:
```
from scipy.stats import ttest_ind

# calculate t_test between Click_through
t_stat, p_value = ttest_ind(df[df.Version=='A']['Click_through'], df[df.
 ↪Version=='B']['Click_through'])
print(f't_stat: {t_stat}\np_value:{p_value}', end='\n\n')

# significance level
alpha = 0.05
if p_value < alpha:
    print("There is significant evidence to reject the null hypothesis.")
    print("Version B leads to a higher Click through compared to Version A.")
else:
    print("There is not enough evidence to reject the null hypothesis.")
    print("There is no significant difference in Click through between Version␣
 ↪A and Version B.")
```

```
t_stat: -2.8441804218860818
p_value:0.00454353573685893

There is significant evidence to reject the null hypothesis.
Version B leads to a higher Click through compared to Version A.
```

[89]:
```
print("Click Through Confidence Interval Values\n")
Confidence_Interval(df.loc[df.Version=='A', ['Click_through']])
```

```
Click Through Confidence Interval Values
```

[89]:
```
                Sample Size  Sample Mean  Standard Error  Margin of Error  \
Click_through           500         0.36        0.000961          0.00048

                Lower Bound (95% CI)  Upper Bound (95% CI)
Click_through               0.215656              0.504344
```

[90]:
```
Confidence_Interval(df.loc[df.Version=='B', ['Click_through']])
```

[90]:

	Sample Size	Sample Mean	Standard Error	Margin of Error	\
Click_through	500	0.448	0.000996	0.000498	

	Lower Bound (95% CI)	Upper Bound (95% CI)
Click_through	0.298457	0.597543

1.2.4 Hypothesis between Bounce

[91]:
```
from scipy.stats import ttest_ind

# calculate t_test between Bounces
t_stat, p_value = ttest_ind(df[df.Version=='A']['Bounce'], df[df.
 ↪Version=='B']['Bounce'])
print(f't_stat: {t_stat}\np_value:{p_value}', end='\n\n')

# significance level
alpha = 0.05
if p_value < alpha:
    print("There is significant evidence to reject the null hypothesis.")
    print("Version B leads to a higher Bounce compared to Version A.")
else:
    print("There is not enough evidence to reject the null hypothesis.")
    print("There is no significant difference in Bounce between Version A and␣
 ↪Version B.")
```

```
t_stat: 0.819859645047051
p_value:0.41249188079475607

There is not enough evidence to reject the null hypothesis.
There is no significant difference in Bounce between Version A and Version B.
```

[92]:
```
print("Bounce Confidence Interval Values\n")
Confidence_Interval(df.loc[df.Version=='A', ['Bounce']])
```

```
Bounce Confidence Interval Values
```

[92]:

	Sample Size	Sample Mean	Standard Error	Margin of Error	\
Bounce	500	0.776	0.000835	0.000417	

	Lower Bound (95% CI)	Upper Bound (95% CI)
Bounce	0.650624	0.901376

[93]:
```
Confidence_Interval(df.loc[df.Version=='B', ['Bounce']])
```

[93]:
```
         Sample Size  Sample Mean  Standard Error  Margin of Error  \
Bounce           500        0.754        0.000862         0.000431

         Lower Bound (95% CI)  Upper Bound (95% CI)
Bounce               0.624488              0.883512
```

1.2.5 We can conclude from above hypothesis testing that there is significance difference between both versions. Hence Version B leads to more conversion rate than Version A

2 Task 2: Real Industry project:

- Develop a script for scraping email addresses from a list of domains
- Ensure the script can locate and extract information from the "Impressum" page of each domain,
- Handle different variations of email addresses (@, [ät], at, and so on…)
- Can extract the Email Address from an Image. (Sometimes webmaster paste the Email as a JPEG or PNG, to prevent scraping)

[10]:
```
from bs4 import BeautifulSoup
import requests
import pandas as pd
import re
from PIL import Image, UnidentifiedImageError
import pytesseract
import io
```

[40]:
```
pip install pytesseract
```

```
Collecting pytesseractNote: you may need to restart the kernel to use updated
packages.

  Downloading pytesseract-0.3.10-py3-none-any.whl (14 kB)
Requirement already satisfied: packaging>=21.3 in
d:\apps\anaconda\files\lib\site-packages (from pytesseract) (23.0)
Requirement already satisfied: Pillow>=8.0.0 in d:\apps\anaconda\files\lib\site-
packages (from pytesseract) (9.4.0)
Installing collected packages: pytesseract
Successfully installed pytesseract-0.3.10

WARNING: Ignoring invalid distribution - (d:\apps\anaconda\files\lib\site-
packages)
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution - (d:\apps\anaconda\files\lib\site-
packages)
WARNING: Ignoring invalid distribution -ensorflow-intel
```

```
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
```

[11]:
```python
urls = [
"peersociallending.com",
"kreditvergleich-kostenlos.net",
"matblog.de",
"malta-tours.de",
"wiseclerk.com",
"urlaub-in-thailand.com",
"findle.top",
"niederrheinzeitung.de",
"finanziell-umdenken.blogspot.com",
"midbio.org",
"klaudija.de",
"pc-welt.wiki",
"websitevalue.co.uk",
"freizeitcafe.info",
"ladenbau.de",
"bierspot.de",
"biboxs.com",
"finance-it-blog.de",
"guenstigerkreditvergleich.com",
"cloudbiz.one",
"frag-den-heimwerker.com",
"fintech-intel.com",
"selbst-schuld.com",
"eltemkredit.com",
"binoro.de",
"siteurl.org",
"frachiseportal.at",
"finlord.cz",
"vj-coach.de",
"mountainstatescfc.org",
"crowdstreet.de"
]

def checkImageExtension(image):
    pattern = re.compile(r"\.(jpg|jpeg|png|PNG)$", re.IGNORECASE)
    if(pattern.search(image)):
        return True
    else:
        return False

output = []
for url in urls:
```

```
    try:
        response = requests.get(f'http://{url}')
        if response.status_code == 200:
            doc = BeautifulSoup(response.text, "html.parser")

            if doc.select_one("a[href*=impressum]"):
                impressumLink = doc.select_one("a[href*=impressum]").get('href')
                if (impressumLink.find("http") == -1):
                    impressumLink = f'http://{url}{impressumLink}'

                # request/open impressum link
                impressumResponse = requests.get(impressumLink).text
                impressumDoc = BeautifulSoup(impressumResponse, "html.parser")

                emails = re.findall(r"\b[A-Za-z0-9._%+-]+(?:
↪@|\[at\]|\[ \]|\[at\]|\(at\)|\[ät\]|ät|at)\b[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b",␣
↪impressumResponse, re.I)

                images = impressumDoc.find_all("img")
                imageEmails = []
                for image in images:
                    if(image.get("src").find("http") == -1):
                        image = f'http://{url}{image.get("src")}'
                    else:
                        image = image.get("src")
                    if(checkImageExtension(image)):
                        r = requests.get(image)
                        try:
                            imageText = pytesseract.image_to_string(Image.
↪open(io.BytesIO(r.content)))
                            imageEmails = re.findall(r"\b[A-Za-z0-9._%+-]+(?:
↪@|\[at\]|\[ \]|\[at\]|\(at\)|\[ät\]|ät|at)\b[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b",␣
↪imageText, re.I)
                        except UnidentifiedImageError:
                            continue
                        except Exception as e:
                            continue

                output.extend(emails + imageEmails)
                print(output, end="\n\n\n")

            else:
                print(f"No Impressum Link: {url}")

        else:
            print(f"No (Status Code {response.status_code}): {url}")
    except requests.ConnectionError as e:
```

```
        continue
```

```
No Impressum Link: peersociallending.com
['finance@kurbsn.com']


['finance@kurbsn.com', '3@o-pr.de']


['finance@kurbsn.com', '3@o-pr.de']


['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com']


['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com']


No Impressum Link: findle.top
['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com']


No Impressum Link: finanziell-umdenken.blogspot.com
No Impressum Link: midbio.org
['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de']


No Impressum Link: pc-welt.wiki
No Impressum Link: websitevalue.co.uk
['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com']


['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com',
'beratung@ladenbau.de', 'beratung@ladenbau.de', 'info@ladenbau.de',
'beratung@ladenbau.de', 'beratung@ladenbau.de',
'RckrufeinrichtenLadenbaude@blickfang-onlinemarketing.de',
'beratung@ladenbau.de']


['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
```

'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com',
'beratung@ladenbau.de', 'beratung@ladenbau.de', 'info@ladenbau.de',
'beratung@ladenbau.de', 'beratung@ladenbau.de',
'RckrufeinrichtenLadenbaude@blickfang-onlinemarketing.de',
'beratung@ladenbau.de', 'torsten[at]bierspot.de', 'torsten[at]bierspot.de']

No (Status Code 522): biboxs.com
['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com',
'beratung@ladenbau.de', 'beratung@ladenbau.de', 'info@ladenbau.de',
'beratung@ladenbau.de', 'beratung@ladenbau.de',
'RckrufeinrichtenLadenbaude@blickfang-onlinemarketing.de',
'beratung@ladenbau.de', 'torsten[at]bierspot.de', 'torsten[at]bierspot.de',
'20info@pass-consulting.com', 'info[at]pass-consulting.com', '20marketing@pass-
consulting.com', 'marketing[at]pass-consulting.com']

No Impressum Link: guenstigerkreditvergleich.com
No Impressum Link: cloudbiz.one
['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com',
'beratung@ladenbau.de', 'beratung@ladenbau.de', 'info@ladenbau.de',
'beratung@ladenbau.de', 'beratung@ladenbau.de',
'RckrufeinrichtenLadenbaude@blickfang-onlinemarketing.de',
'beratung@ladenbau.de', 'torsten[at]bierspot.de', 'torsten[at]bierspot.de',
'20info@pass-consulting.com', 'info[at]pass-consulting.com', '20marketing@pass-
consulting.com', 'marketing[at]pass-consulting.com', 'info@frag-den-
heimwerker.com', 'info@frag-den-heimwerker.com']

No Impressum Link: fintech-intel.com
['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com',
'beratung@ladenbau.de', 'beratung@ladenbau.de', 'info@ladenbau.de',
'beratung@ladenbau.de', 'beratung@ladenbau.de',
'RckrufeinrichtenLadenbaude@blickfang-onlinemarketing.de',
'beratung@ladenbau.de', 'torsten[at]bierspot.de', 'torsten[at]bierspot.de',
'20info@pass-consulting.com', 'info[at]pass-consulting.com', '20marketing@pass-
consulting.com', 'marketing[at]pass-consulting.com', 'info@frag-den-
heimwerker.com', 'info@frag-den-heimwerker.com']

['finance@kurbsn.com', '3@o-pr.de', 'info@p2p-kredite.com',

```
'info@p2p-kredite.com', 'info[at]klaudija.de', 'info[at]klaudija.de',
'christiangeradigital@gmail.com', 'christiangeradigital@gmail.com',
'beratung@ladenbau.de', 'beratung@ladenbau.de', 'info@ladenbau.de',
'beratung@ladenbau.de', 'beratung@ladenbau.de',
'RckrufeinrichtenLadenbaude@blickfang-onlinemarketing.de',
'beratung@ladenbau.de', 'torsten[at]bierspot.de', 'torsten[at]bierspot.de',
'20info@pass-consulting.com', 'info[at]pass-consulting.com', '20marketing@pass-
consulting.com', 'marketing[at]pass-consulting.com', 'info@frag-den-
heimwerker.com', 'info@frag-den-heimwerker.com', 'info@astuna.de']


No Impressum Link: siteurl.org
No Impressum Link: finlord.cz
No Impressum Link: vj-coach.de
No (Status Code 410): crowdstreet.de
```

[16]:
```
df = pd.DataFrame({'Emails': output})
df.to_csv("Emails.csv")
```

[17]:
```
emailsDF = pd.read_csv("Emails.csv")
emailsDF
```

[17]:

	Unnamed: 0	Emails
0	0	finance@kurbsn.com
1	1	3@o-pr.de
2	2	info@p2p-kredite.com
3	3	info@p2p-kredite.com
4	4	info[at]klaudija.de
5	5	info[at]klaudija.de
6	6	christiangeradigital@gmail.com
7	7	christiangeradigital@gmail.com
8	8	beratung@ladenbau.de
9	9	beratung@ladenbau.de
10	10	info@ladenbau.de
11	11	beratung@ladenbau.de
12	12	beratung@ladenbau.de
13	13	RckrufeinrichtenLadenbaude@blickfang-onlinemar…
14	14	beratung@ladenbau.de
15	15	torsten[at]bierspot.de
16	16	torsten[at]bierspot.de
17	17	20info@pass-consulting.com
18	18	info[at]pass-consulting.com
19	19	20marketing@pass-consulting.com
20	20	marketing[at]pass-consulting.com
21	21	info@frag-den-heimwerker.com
22	22	info@frag-den-heimwerker.com
23	23	info@astuna.de

3 Task 3: Retail Services csv file is attached you need to analyse and answer these questions in notebook after processing.

[3]:
```
RetailDF = pd.read_csv("retail_services.csv")
RetailDF
```

[3]:
```
     time.index  time.month time.month name time.period  time.year  \
0             1           1              Jan      Jan-92       1992
1             2           2              Feb      Feb-92       1992
2             3           3              Mar      Mar-92       1992
3             4           4              Apr      Apr-92       1992
4             5           5              May      May-92       1992
..          …           …              …        …          …
284         285           9              Sep      Sep-15       2015
285         286          10              Oct      Oct-15       2015
286         287          11              Nov      Nov-15       2015
287         288          12              Dec      Dec-15       2015
288         289           1              Jan      Jan-16       2016

     data.inventories.all department stores  \
0                                         0
1                                         0
2                                         0
3                                         0
4                                         0
..                                        …
284                                       0
285                                       0
286                                       0
287                                       0
288                                       0

     data.inventories.all other home furnishings stores  \
0                                                     0
1                                                     0
2                                                     0
3                                                     0
4                                                     0
..                                                    …
284                                                   0
285                                                   0
286                                                   0
287                                                   0
288                                                   0

     data.inventories.all other merchandise stores  \
0                                                0
```

```
1                                                    0
2                                                    0
3                                                    0
4                                                    0
..                                                   …
284                                                  0
285                                                  0
286                                                  0
287                                                  0
288                                                  0

     data.inventories.appliances and other electronics stores  \
0                                                        0
1                                                        0
2                                                        0
3                                                        0
4                                                        0
..                                                       …
284                                                      0
285                                                      0
286                                                      0
287                                                      0
288                                                      0

     data.inventories.auto and other motor vehicles  …  \
0                                                    0  …
1                                                    0  …
2                                                    0  …
3                                                    0  …
4                                                    0  …
..                                                   …  …
284                                                  0  …
285                                                  0  …
286                                                  0  …
287                                                  0  …
288                                                  0  …

     data.sales.retail trade and food services, ex auto  \
0                                                116565
1                                                115862
2                                                124200
3                                                127587
4                                                133608
..                                                  …
284                                              338500
285                                              353708
286                                              359528
```

```
287                                          423095
288                                          319532

     data.sales.retail trade, ex auto  data.sales.shoe stores  \
0                              100872                    1206
1                              100027                    1265
2                              107352                    1463
3                              111093                    1675
4                              115960                    1560
..                                ...                     ...
284                            287804                    2565
285                            299714                    2663
286                            309281                    2827
287                            368440                    3985
288                            269308                    2063

     data.sales.sporting goods stores  \
0                                 972
1                                1100
2                                1214
3                                1267
4                                1293
..                                ...
284                              3623
285                              3406
286                              3860
287                              6444
288                              3069

     data.sales.sporting goods, hobby, book, and music stores  \
0                                                    3439
1                                                    3264
2                                                    3473
3                                                    3523
4                                                    3545
..                                                    ...
284                                                  7125
285                                                  6738
286                                                  8025
287                                                 13025
288                                                  6799

     data.sales.supermarkets and other grocery (except convenience) stores  \
0                                                    0
1                                                    0
2                                                    0
3                                                    0
```

```
4                                                    0
..                                                   …
284                                              47244
285                                              48964
286                                              48505
287                                              51216
288                                              49251

     data.sales.used car dealers  data.sales.used merchandise stores  \
0                           1744                                 371
1                           1990                                 402
2                           2177                                 419
3                           2601                                 393
4                           2171                                 435
..                           …                                   …
284                         7094                                1497
285                         7283                                1633
286                         6605                                1413
287                         6507                                1436
288                         7021                                1254

     data.sales.warehouse clubs and superstores  \
0                                          2579
1                                          2615
2                                          2838
3                                          2984
4                                          3257
..                                          …
284                                       34745
285                                       37352
286                                       39731
287                                       45540
288                                       34071

     data.sales.women's clothing stores
0                                  1873
1                                  1991
2                                  2403
3                                  2665
4                                  2752
..                                  …
284                                3549
285                                3878
286                                4172
287                                5507
288                                2797
```

```
[289 rows x 197 columns]
```

[4]:
```
RetailDF.columns
```

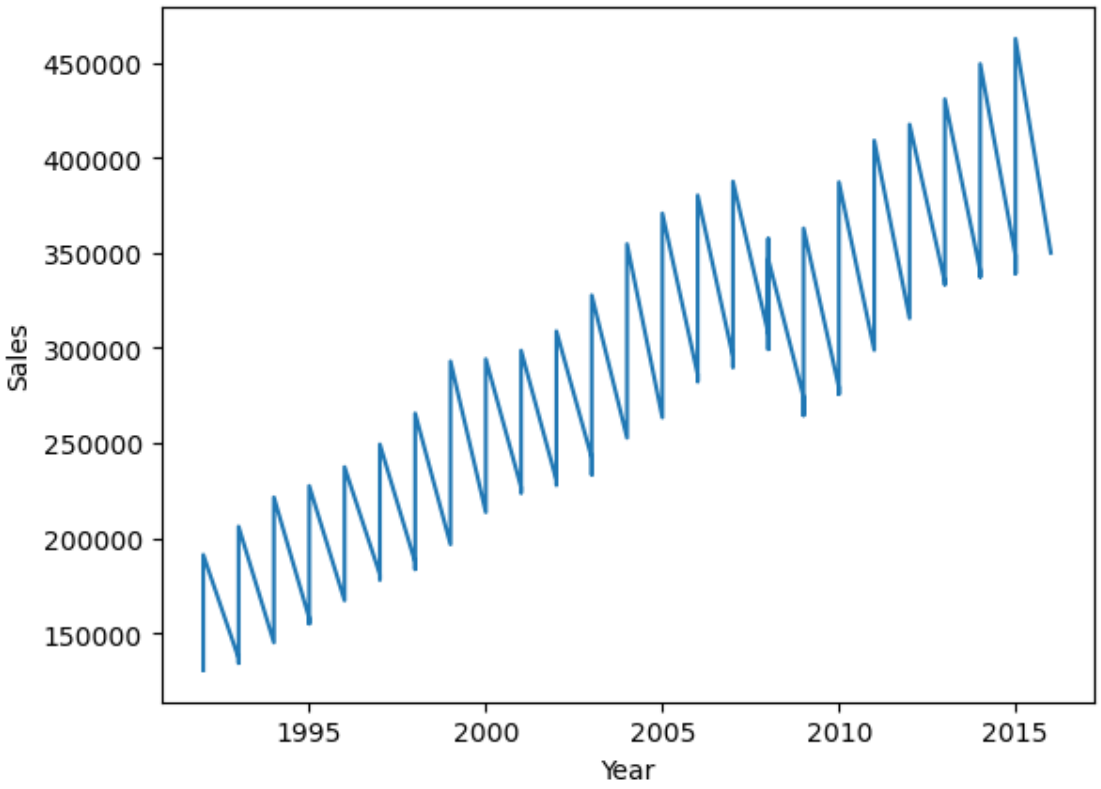
[4]:
```
Index(['time.index', 'time.month', 'time.month name', 'time.period',
       'time.year', 'data.inventories.all department stores',
       'data.inventories.all other home furnishings stores',
       'data.inventories.all other merchandise stores',
       'data.inventories.appliances and other electronics stores',
       'data.inventories.auto and other motor vehicles',
       …
       'data.sales.retail trade and food services, ex auto',
       'data.sales.retail trade, ex auto', 'data.sales.shoe stores',
       'data.sales.sporting goods stores',
       'data.sales.sporting goods, hobby, book, and music stores',
       'data.sales.supermarkets and other grocery (except convenience) stores',
       'data.sales.used car dealers', 'data.sales.used merchandise stores',
       'data.sales.warehouse clubs and superstores',
       'data.sales.women's clothing stores'],
      dtype='object', length=197)
```

[5]:
```
import seaborn as sns
import matplotlib.pyplot as plt
```

3.1 How has retail economic activity in the United States changed over the past five years?

[8]:
```
plt.plot(RetailDF['time.year'], RetailDF['data.sales.retail trade'])
plt.xlabel("Year")
plt.ylabel("Sales")
```

[8]:
```
Text(0, 0.5, 'Sales')
```



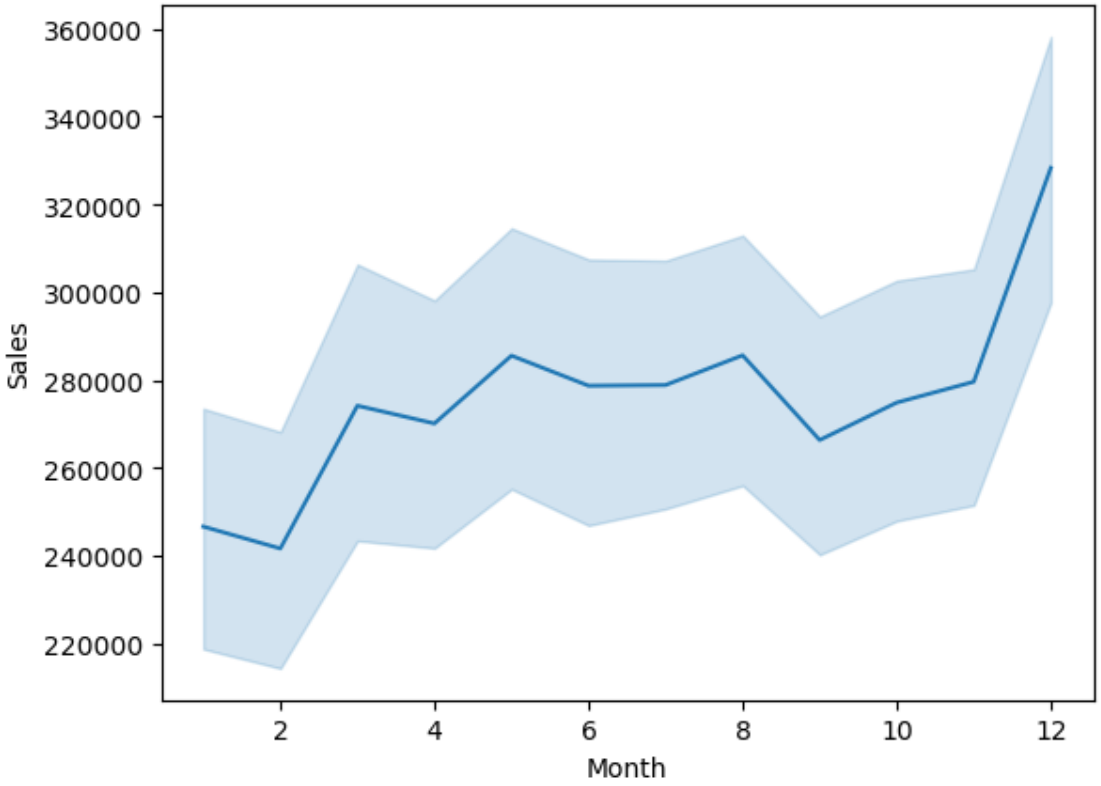
3.2 What are the key differences between the Advance Monthly Retail Trade Survey (MARTS) and the Annual Retail Trade Survey (ARTS)?

Survey Type	Frequency of Data Collection	Coverage (Number of Firms)	Data Detail and Scope	Usage and Release Timeframe
MARTS	Monthly	Approx. 5,500	Less detailed, aggregate figures	Short-term analysis, released ~2 weeks after reference month
ARTS	Annual	Approx. 12,000	More detailed, breakdowns by product lines, sectors, regions, and types of retailers	Long-term planning and trend analysis, released several months after reference year

3.3 Can we identify any seasonal patterns or trends in monthly retail sales data?

[9]:
```
sns.lineplot(RetailDF, x=RetailDF['time.month'], y=RetailDF['data.sales.retail␣
↪trade'])
plt.xlabel("Month")
plt.ylabel("Sales")
```

[9]: Text(0, 0.5, 'Sales')

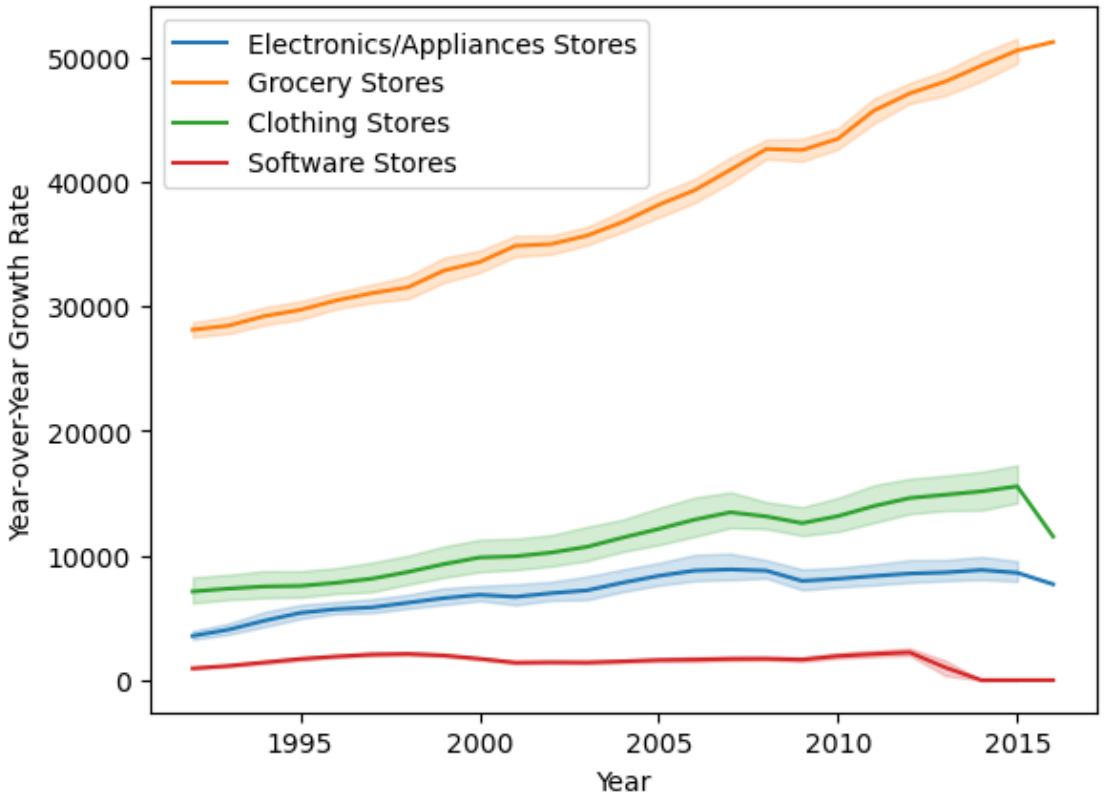


3.4 Are there any specific retail sectors that have shown significant growth or decline in recent years?

[10]:
```
sns.lineplot(data=RetailDF, x=RetailDF['time.year'], y=RetailDF['data.sales.
 ↪electronics and appliance stores'], label='Electronics/Appliances Stores')
sns.lineplot(data=RetailDF, x=RetailDF['time.year'], y=RetailDF['data.sales.
 ↪grocery stores'], label='Grocery Stores')
sns.lineplot(data=RetailDF, x=RetailDF['time.year'], y=RetailDF['data.sales.
 ↪clothing stores'], label='Clothing Stores')
sns.lineplot(data=RetailDF, x=RetailDF['time.year'], y=RetailDF['data.sales.
 ↪computer and software stores'], label='Software Stores')

plt.xlabel("Year")
plt.ylabel("Year-over-Year Growth Rate")
plt.legend()
```

[10]: <matplotlib.legend.Legend at 0x262c1f92670>



[12]:
```
RetailDF
```

[12]:

	time.index	time.month	time.month name	time.period	time.year	\
0	1	1	Jan	Jan-92	1992	
1	2	2	Feb	Feb-92	1992	
2	3	3	Mar	Mar-92	1992	
3	4	4	Apr	Apr-92	1992	
4	5	5	May	May-92	1992	
..	…	…	…	…	…	
284	285	9	Sep	Sep-15	2015	
285	286	10	Oct	Oct-15	2015	
286	287	11	Nov	Nov-15	2015	
287	288	12	Dec	Dec-15	2015	
288	289	1	Jan	Jan-16	2016	

	data.inventories.all department stores	\
0	0	
1	0	
2	0	
3	0	
4	0	

```
..                                      …
284                                     0
285                                     0
286                                     0
287                                     0
288                                     0

     data.inventories.all other home furnishings stores  \
0                                                     0
1                                                     0
2                                                     0
3                                                     0
4                                                     0
..                                                  …
284                                                   0
285                                                   0
286                                                   0
287                                                   0
288                                                   0

     data.inventories.all other merchandise stores  \
0                                                 0
1                                                 0
2                                                 0
3                                                 0
4                                                 0
..                                              …
284                                               0
285                                               0
286                                               0
287                                               0
288                                               0

     data.inventories.appliances and other electronics stores  \
0                                                     0
1                                                     0
2                                                     0
3                                                     0
4                                                     0
..                                                  …
284                                                   0
285                                                   0
286                                                   0
287                                                   0
288                                                   0

     data.inventories.auto and other motor vehicles  …  \
```

```
0                                                    0  …
1                                                    0  …
2                                                    0  …
3                                                    0  …
4                                                    0  …
..                                                   …  …
284                                                  0  …
285                                                  0  …
286                                                  0  …
287                                                  0  …
288                                                  0  …

     data.sales.retail trade and food services, ex auto  \
0                                                116565
1                                                115862
2                                                124200
3                                                127587
4                                                133608
..                                                  …
284                                              338500
285                                              353708
286                                              359528
287                                              423095
288                                              319532

     data.sales.retail trade, ex auto  data.sales.shoe stores  \
0                              100872                    1206
1                              100027                    1265
2                              107352                    1463
3                              111093                    1675
4                              115960                    1560
..                                …                       …
284                            287804                    2565
285                            299714                    2663
286                            309281                    2827
287                            368440                    3985
288                            269308                    2063

     data.sales.sporting goods stores  \
0                                 972
1                                1100
2                                1214
3                                1267
4                                1293
..                                …
284                              3623
285                              3406
```

```
286                              3860
287                              6444
288                              3069

     data.sales.sporting goods, hobby, book, and music stores  \
0                                                  3439
1                                                  3264
2                                                  3473
3                                                  3523
4                                                  3545
..                                                  …
284                                                7125
285                                                6738
286                                                8025
287                                               13025
288                                                6799

     data.sales.supermarkets and other grocery (except convenience) stores  \
0                                                     0
1                                                     0
2                                                     0
3                                                     0
4                                                     0
..                                                  …
284                                               47244
285                                               48964
286                                               48505
287                                               51216
288                                               49251

     data.sales.used car dealers  data.sales.used merchandise stores  \
0                           1744                                371
1                           1990                                402
2                           2177                                419
3                           2601                                393
4                           2171                                435
..                           …                                …
284                         7094                               1497
285                         7283                               1633
286                         6605                               1413
287                         6507                               1436
288                         7021                               1254

     data.sales.warehouse clubs and superstores  \
0                                          2579
1                                          2615
2                                          2838
```

```
3                                          2984
4                                          3257
..                                          …
284                                       34745
285                                       37352
286                                       39731
287                                       45540
288                                       34071

     data.sales.women's clothing stores
0                                  1873
1                                  1991
2                                  2403
3                                  2665
4                                  2752
..                                  …
284                                3549
285                                3878
286                                4172
287                                5507
288                                2797

[289 rows x 197 columns]
```

[17]:
```
print(RetailDF.columns.to_list())
```

```
['time.index', 'time.month', 'time.month name', 'time.period', 'time.year',
'data.inventories.all department stores', 'data.inventories.all other home
furnishings stores', 'data.inventories.all other merchandise stores',
'data.inventories.appliances and other electronics stores',
'data.inventories.auto and other motor vehicles', 'data.inventories.automobile
dealers', 'data.inventories.automotive parts and tire stores',
'data.inventories.beer, wine, and liquor stores', 'data.inventories.book
stores', 'data.inventories.building materials and garden supplies dealers',
'data.inventories.building supplies dealers', 'data.inventories.clothing
stores', 'data.inventories.computer and software stores',
'data.inventories.discount department stores', 'data.inventories.drinking
places', 'data.inventories.electronic shopping and mail-order houses',
'data.inventories.electronics and appliance stores', 'data.inventories.family
clothing stores', 'data.inventories.floor covering stores',
'data.inventories.food and beverage stores', 'data.inventories.food services and
drinking places', 'data.inventories.fuel dealers', 'data.inventories.full
service restaurants', 'data.inventories.furniture and home furnishings stores',
'data.inventories.furniture stores', 'data.inventories.furniture, home furn,
electronics, and appliance stores', 'data.inventories.gafo',
'data.inventories.gasoline stations', 'data.inventories.general merchandise
stores', 'data.inventories.gift, novelty, and souvenir stores',
'data.inventories.grocery stores', 'data.inventories.hardware stores',
```

'data.inventories.health and personal care stores', 'data.inventories.hobby, toy, and game stores', 'data.inventories.home furnishings stores', 'data.inventories.household appliance stores', 'data.inventories.jewelry stores', 'data.inventories.limited service eating places', "data.inventories.men's clothing stores", 'data.inventories.miscellaneous store retailers', 'data.inventories.motor vehicle and parts dealers', 'data.inventories.new car dealers', 'data.inventories.non-discount department stores', 'data.inventories.non-leased department stores', 'data.inventories.nonstore retailers', 'data.inventories.office supplies and stationery stores', 'data.inventories.office supplies, stationery, and gift stores', 'data.inventories.other clothing stores', 'data.inventories.other general merchandise stores', 'data.inventories.paint and wallpaper stores', 'data.inventories.pharmacies and drug stores', 'data.inventories.radio, TV, and electronics stores', 'data.inventories.retail trade', 'data.inventories.retail trade and food services', 'data.inventories.retail trade and food services, ex auto', 'data.inventories.retail trade, ex auto', 'data.inventories.shoe stores', 'data.inventories.sporting goods stores', 'data.inventories.sporting goods, hobby, book, and music stores', 'data.inventories.supermarkets and other grocery (except convenience) stores', 'data.inventories.used car dealers', 'data.inventories.used merchandise stores', 'data.inventories.warehouse clubs and superstores', "data.inventories.women's clothing stores", 'data.ratio.all department stores', 'data.ratio.all other home furnishings stores', 'data.ratio.all other merchandise stores', 'data.ratio.appliances and other electronics stores', 'data.ratio.auto and other motor vehicles', 'data.ratio.automobile dealers', 'data.ratio.automotive parts and tire stores', 'data.ratio.beer, wine, and liquor stores', 'data.ratio.book stores', 'data.ratio.building materials and garden supplies dealers', 'data.ratio.building supplies dealers', 'data.ratio.clothing stores', 'data.ratio.computer and software stores', 'data.ratio.discount department stores', 'data.ratio.drinking places', 'data.ratio.electronic shopping and mail-order houses', 'data.ratio.electronics and appliance stores', 'data.ratio.family clothing stores', 'data.ratio.floor covering stores', 'data.ratio.food and beverage stores', 'data.ratio.food services and drinking places', 'data.ratio.fuel dealers', 'data.ratio.full service restaurants', 'data.ratio.furniture and home furnishings stores', 'data.ratio.furniture stores', 'data.ratio.furniture, home furn, electronics, and appliance stores', 'data.ratio.gafo', 'data.ratio.gasoline stations', 'data.ratio.general merchandise stores', 'data.ratio.gift, novelty, and souvenir stores', 'data.ratio.grocery stores', 'data.ratio.hardware stores', 'data.ratio.health and personal care stores', 'data.ratio.hobby, toy, and game stores', 'data.ratio.home furnishings stores', 'data.ratio.household appliance stores', 'data.ratio.jewelry stores', 'data.ratio.limited service eating places', "data.ratio.men's clothing stores", 'data.ratio.miscellaneous store retailers', 'data.ratio.motor vehicle and parts dealers', 'data.ratio.new car dealers', 'data.ratio.non-discount department stores', 'data.ratio.non-leased department stores', 'data.ratio.nonstore retailers', 'data.ratio.office supplies and stationery stores', 'data.ratio.office supplies, stationery, and gift stores', 'data.ratio.other clothing stores', 'data.ratio.other general merchandise

```
stores', 'data.ratio.paint and wallpaper stores', 'data.ratio.pharmacies and
drug stores', 'data.ratio.radio, TV, and electronics stores', 'data.ratio.retail
trade', 'data.ratio.retail trade and food services', 'data.ratio.retail trade
and food services, ex auto', 'data.ratio.retail trade, ex auto',
'data.ratio.shoe stores', 'data.ratio.sporting goods stores',
'data.ratio.sporting goods, hobby, book, and music stores',
'data.ratio.supermarkets and other grocery (except convenience) stores',
'data.ratio.used car dealers', 'data.ratio.used merchandise stores',
'data.ratio.warehouse clubs and superstores', "data.ratio.women's clothing
stores", 'data.sales.all department stores', 'data.sales.all other home
furnishings stores', 'data.sales.all other merchandise stores',
'data.sales.appliances and other electronics stores', 'data.sales.auto and other
motor vehicles', 'data.sales.automobile dealers', 'data.sales.automotive parts
and tire stores', 'data.sales.beer, wine, and liquor stores', 'data.sales.book
stores', 'data.sales.building materials and garden supplies dealers',
'data.sales.building supplies dealers', 'data.sales.clothing stores',
'data.sales.computer and software stores', 'data.sales.discount department
stores', 'data.sales.drinking places', 'data.sales.electronic shopping and mail-
order houses', 'data.sales.electronics and appliance stores', 'data.sales.family
clothing stores', 'data.sales.floor covering stores', 'data.sales.food and
beverage stores', 'data.sales.food services and drinking places',
'data.sales.fuel dealers', 'data.sales.full service restaurants',
'data.sales.furniture and home furnishings stores', 'data.sales.furniture
stores', 'data.sales.furniture, home furn, electronics, and appliance stores',
'data.sales.gafo', 'data.sales.gasoline stations', 'data.sales.general
merchandise stores', 'data.sales.gift, novelty, and souvenir stores',
'data.sales.grocery stores', 'data.sales.hardware stores', 'data.sales.health
and personal care stores', 'data.sales.hobby, toy, and game stores',
'data.sales.home furnishings stores', 'data.sales.household appliance stores',
'data.sales.jewelry stores', 'data.sales.limited service eating places',
"data.sales.men's clothing stores", 'data.sales.miscellaneous store retailers',
'data.sales.motor vehicle and parts dealers', 'data.sales.new car dealers',
'data.sales.non-discount department stores', 'data.sales.non-leased department
stores', 'data.sales.nonstore retailers', 'data.sales.office supplies and
stationery stores', 'data.sales.office supplies, stationery, and gift stores',
'data.sales.other clothing stores', 'data.sales.other general merchandise
stores', 'data.sales.paint and wallpaper stores', 'data.sales.pharmacies and
drug stores', 'data.sales.radio, TV, and electronics stores', 'data.sales.retail
trade', 'data.sales.retail trade and food services', 'data.sales.retail trade
and food services, ex auto', 'data.sales.retail trade, ex auto',
'data.sales.shoe stores', 'data.sales.sporting goods stores',
'data.sales.sporting goods, hobby, book, and music stores',
'data.sales.supermarkets and other grocery (except convenience) stores',
'data.sales.used car dealers', 'data.sales.used merchandise stores',
'data.sales.warehouse clubs and superstores', "data.sales.women's clothing
stores"]
```

3.5 Is there a relationship between e-commerce sales and brick-and-mortar retail sales?

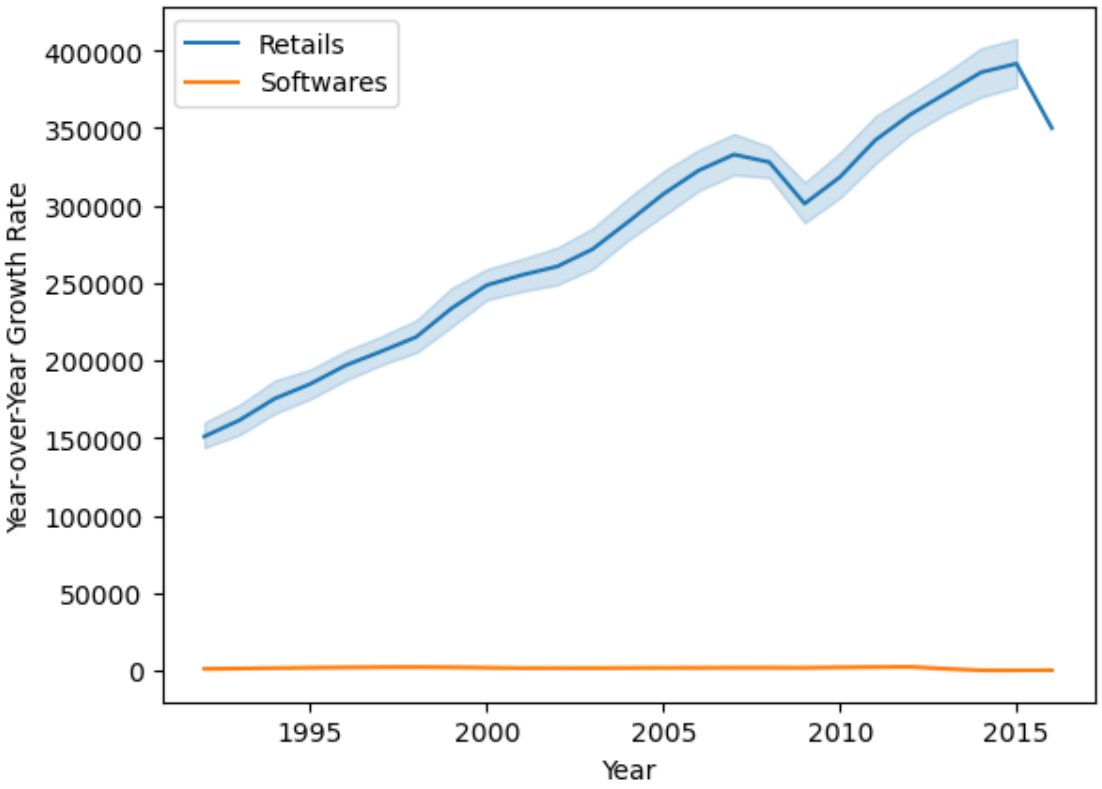
[18]:
```
RetailDF['data.sales.retail trade']
```

[18]:
```
0      130683
1      131244
2      142488
3      147175
4      152420
        …
284    379473
285    390938
286    394128
287    462497
288    350025
Name: data.sales.retail trade, Length: 289, dtype: int64
```

[22]:
```
sns.lineplot(data=RetailDF, x=RetailDF['time.year'], y=RetailDF['data.sales.
 ↪retail trade'], label='Retails')
sns.lineplot(data=RetailDF, x=RetailDF['time.year'], y=RetailDF['data.sales.
 ↪computer and software stores'], label='Softwares')

plt.xlabel("Year")
plt.ylabel("Year-over-Year Growth Rate")
plt.legend()
```

[22]:
```
<matplotlib.legend.Legend at 0x262ccb6b1f0>
```


Retails
Softwares
400000
350000
300000
250000
200000
150000
100000
50000
0
Year-over-Year Growth Rate
1995
2000
2005
2010
2015
Year

[ ]: