Consumer Spending & Unemployment Rates

Group 1: Jesse, Michael, Gurjap, & Anika

Main Questions

- Is there a relationship between unemployment and consumer spending?
- What is the average unemployment rates against the total average employable population?
- What did our relationship to consumer spending look like during the pandemic when unemployment rates were at their highest?

Motivation

- The role of tech/social media in our spending
- The Covid 19 Pandemic
  - Increased circulation of trends
- How has our relationship to spending changed over the years?

Datasets

- Measuring Consumer Spending
  - Vehicle Sales (1976-2023) in units
  - Restaurant Sales (1992-2023)
- Unemployment Rates
  - Unemployment by month (1976-2020)

data.world

U.S. BUREAU OF LABOR STATISTICS

Bureau of Economic Analysis
U.S. DEPARTMENT OF COMMERCE

FRED®
ECONOMIC DATA | ST. LOUIS FED

United States®
Census
Bureau

Cleanup Process

In [6]:
```
#calculation for unemployment rate based on the unemployed and employable population
unemployment_data_raw['Unemployment Rate'] = (unemployment_data_raw['Unemployed'] / unemployment_data_raw['Employabl
```

In [7]:
```
#confirming unemployment rate is in the dataframe
unemployment_data_raw.head()
```

Out[7]:

	FIPS	State	Year	Month	Employable Population	Employed	Unemployed	Unemployment Rate
0	1	Alabama	1976	1	1492409	1392154	100255	6.717663
1	2	Alaska	1976	1	159154	147809	11345	7.128316
2	4	Arizona	1976	1	972413	872738	99675	10.250274
3	5	Arkansas	1976	1	882835	817756	65079	7.371593
4	6	California	1976	1	9781720	8892663	889057	9.088964

In [8]:
```
# extract total unemployed for the year
yearly_total_unemployed = unemployment_data_raw.groupby('Year')['Unemployed'].sum()

yearly_total_unemployed
```

Out[8]:
```
Year
1976     88314307
1977     83221063
1978     73859367
1979     73105365
1980     91708995
1981     99238311
1982    128237967
```

```
#Rename Columns
sales_data_complete = sales_data_complete.rename(columns = {"DATE":"Year", "MRTS
sales_data_complete
```

	Year	Total Restaurant Sales (in millions)	Total Vehicle Sales (in millions)
0	1992-01-01	13325.0	12.591
1	1992-02-01	13474.0	12.927
2	1992-03-01	14346.0	12.824
3	1992-04-01	14065.0	12.550
4	1992-05-01	15077.0	13.098
...	...	...	...
375	2023-04-01	79298.0	16.210
376	2023-05-01	83588.0	16.079
377	2023-06-01	83171.0	16.602
378	2023-07-01	84580.0	16.442
379	2023-08-01	82945.0	15.898

380 rows × 3 columns

26]:
```
result.index = result.index.year.astype(str)
result.reset_index(inplace=True)
```

28]:
```
#Rewriting the 'Year' column to be just the year
def extract_year(date_str):
    year = date_str.split('-')[0]
    return year

# Use a for loop to apply the function to each element in the 'Year' column
for i in range(len(result['Year'])):
    result['Year'][i] = extract_year(result['Year'][i])

# Display the updated DataFrame
result
```

```
/var/folders/hs/gxynbb6j6gn5jdvw7zk_tbhh0000gn/T/ipykernel_37191/1322680015.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  result['Year'][i] = extract_year(result['Year'][i])
```

28]:

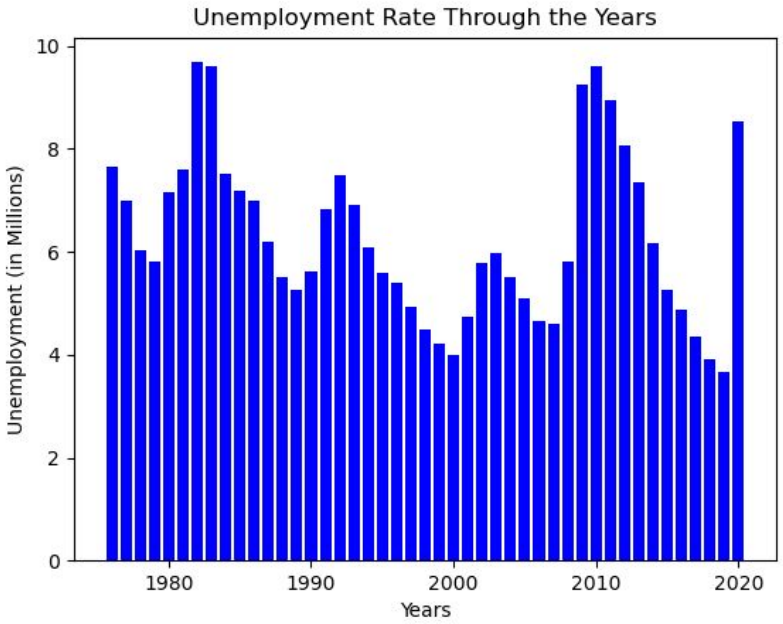
	Year	Total Restaurant Sales (in millions)	Total Vehicle Sales (in millions)
0	1992	173468.0	157.294
1	1993	185719.0	170.110

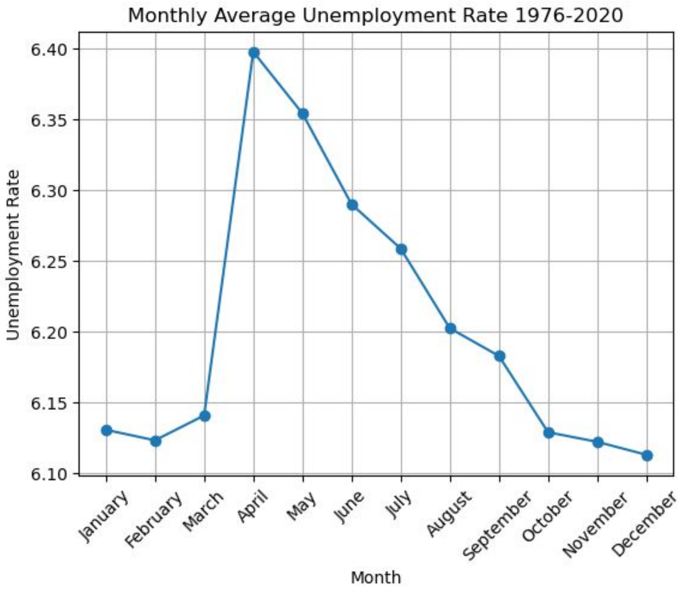
Dataframes we made

	Year	Yearly Unemployment Rate
Year		
1976	1976	7.654154
1977	1977	7.002357
1978	1978	6.018049
1979	1979	5.801846
1980	1980	7.145776
1981	1981	7.606055
1982	1982	9.685206
1983	1983	9.612601
1984	1984	7.523955
1985	1985	7.186397
1986	1986	6.999297

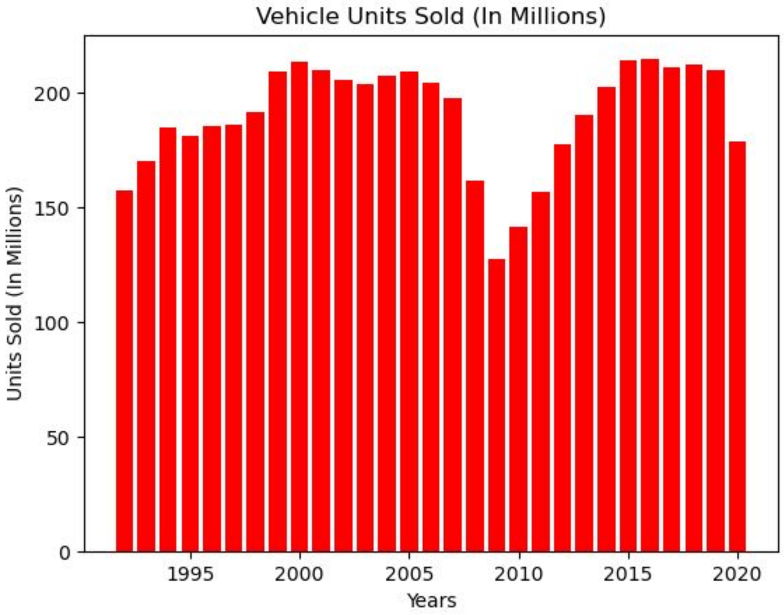
	Year	Total Restaurant Sales (in millions)	Total Vehicle Sales (in millions)
0	1992	173468.0	157.294
1	1993	185719.0	170.110
2	1994	195025.0	184.775
3	1995	202050.0	181.418
4	1996	210149.0	185.455
5	1997	223308.0	185.981
6	1998	234940.0	191.612
7	1999	244761.0	208.978
8	2000	261098.0	213.728
9	2001	272634.0	209.665
10	2002	285492.0	205.658
11	2003	302113.0	203.597
12	2004	323584.0	207.550
13	2005	343153.0	209.354
14	2006	364280.0	204.591
15	2007	381506.0	197.545
16	2008	391048.0	161.917
17	2009	388726.0	127.217

Charts 1

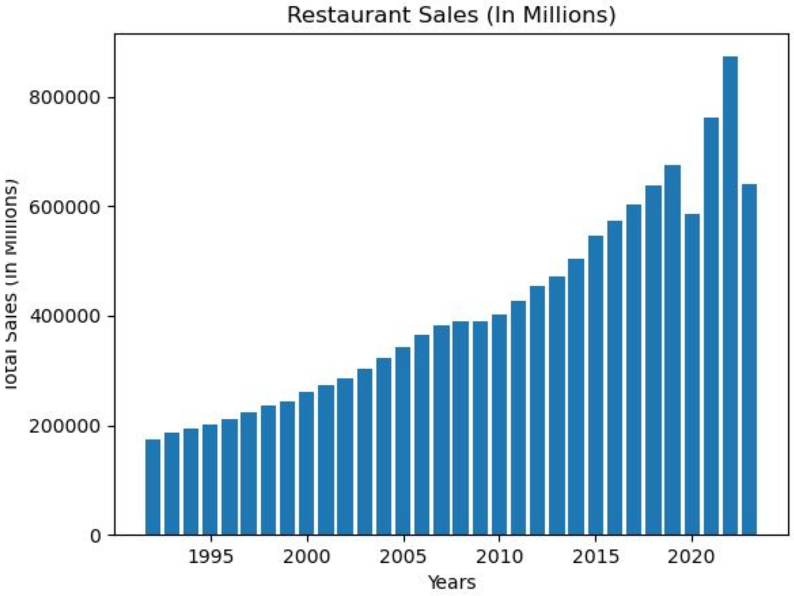

Unemployment Rate Through the Years
Unemployment (in Millions)
0
2
4
6
8
10
1980
1990
2000
2010
2020
Years


Monthly Average Unemployment Rate 1976-2020
Unemployment Rate
6.10
6.15
6.20
6.25
6.30
6.35
6.40
January
February
March
April
May
June
July
August
September
October
November
December
Month

Charts 2

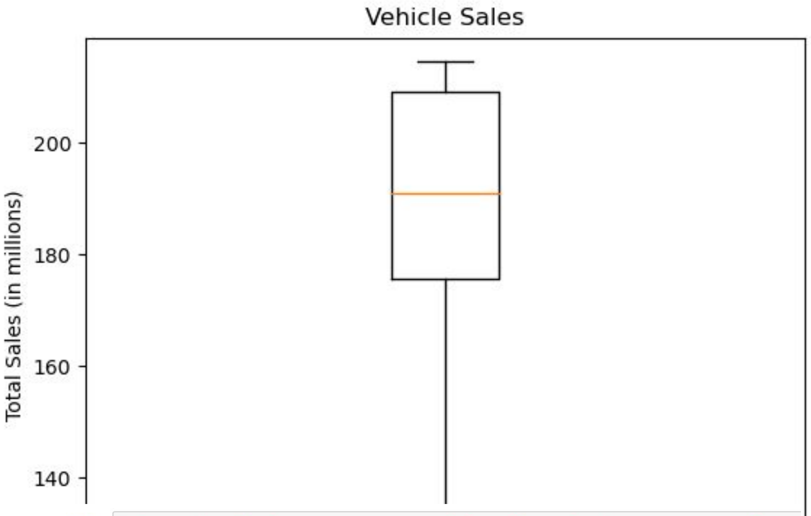


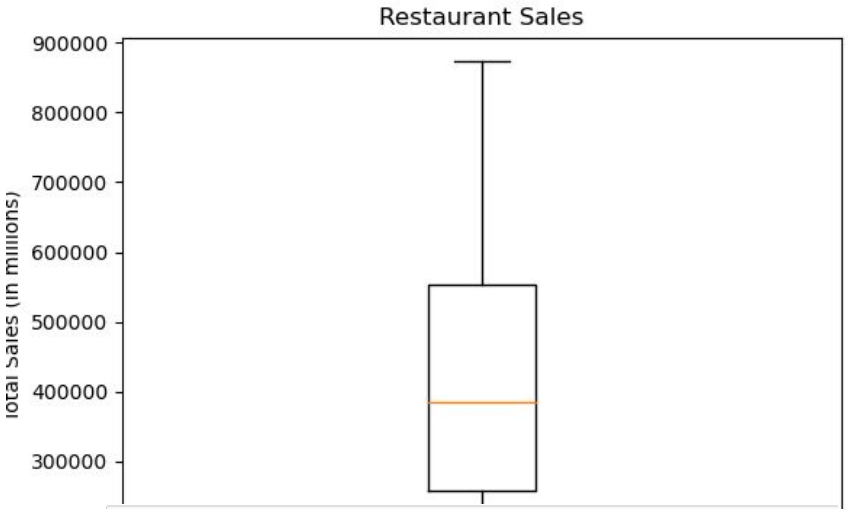
Vehicle Units Sold (In Millions)
Units Sold (In Millions)
0
50
100
150
200
1995
2000
2005
2010
2015
2020
Years



Restaurant Sales (In Millions)
0
200000
400000
600000
800000
1995
2000
2005
2010
2015
2020
Years

Potential Outliers in Sales Data






```
# Determine if there are any potential outliers in the restaurant sales u
quartiles = sales_df['Total Restaurant Sales (in millions)'].quantile([.2
lowerq = quartiles[0.25]
upperq = quartiles[0.75]
iqr = upperq-lowerq

print(f"The lower quartile of restaurant sales is: {lowerq}")
print(f"The upper quartile of restaurant sales is: {upperq}")
print(f"The interquartile range of restaurant sales is: {iqr}")
print(f"The the median of restaurant sales is: {quartiles[0.5]} ")
```

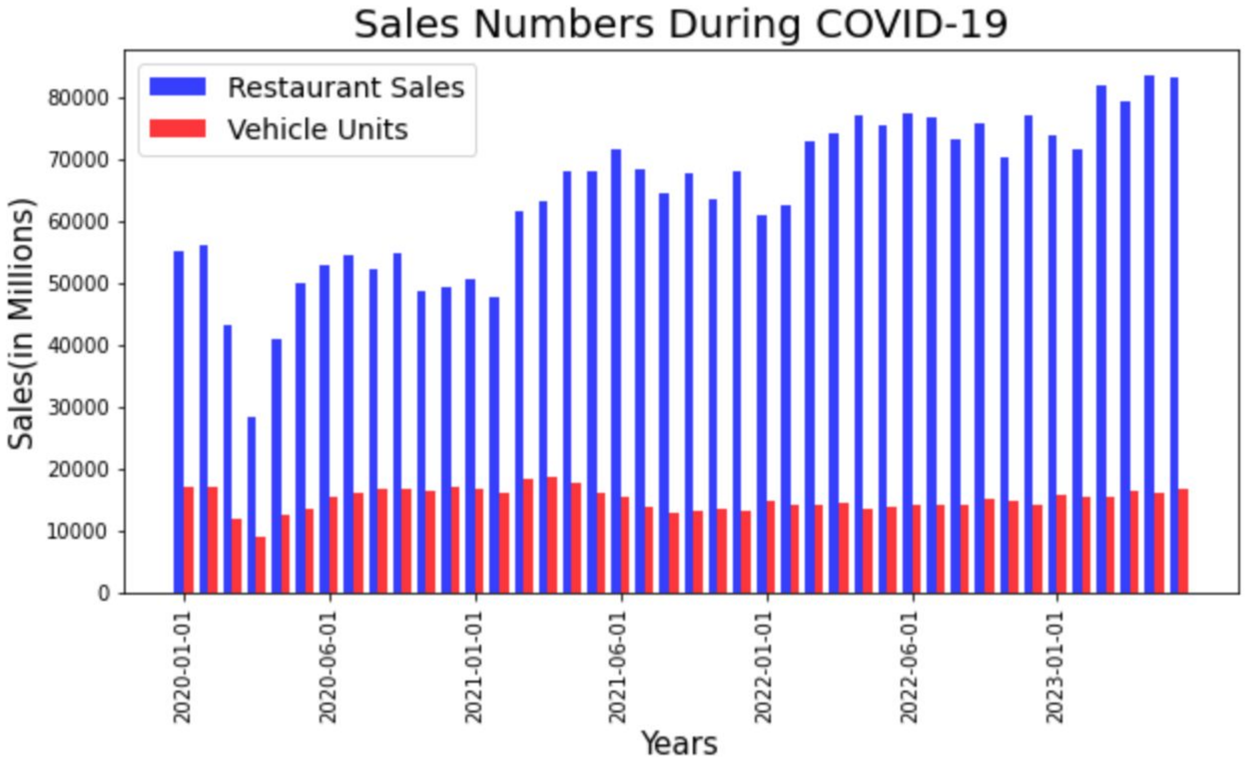
```
The lower quartile of restaurant sales is: 257013.75
The upper quartile of restaurant sales is: 552936.25
The interquartile range of restaurant sales is: 295922.5
The the median of restaurant sales is: 385116.0
```

```
# Determine if there are any potential outliers in the restaurant sales using
quartiles = sales_df['Total Vehicle Sales (in millions)'].quantile([.25,.5,.7
lowerq = quartiles[0.25]
upperq = quartiles[0.75]
iqr = upperq-lowerq

print(f"The lower quartile of vehicle sales is: {lowerq}")
print(f"The upper quartile of vehicle sales is: {upperq}")
print(f"The interquartile range of vehicle sales is: {iqr}")
print(f"The the median of vehicle sales is: {quartiles[0.5]} ")
```

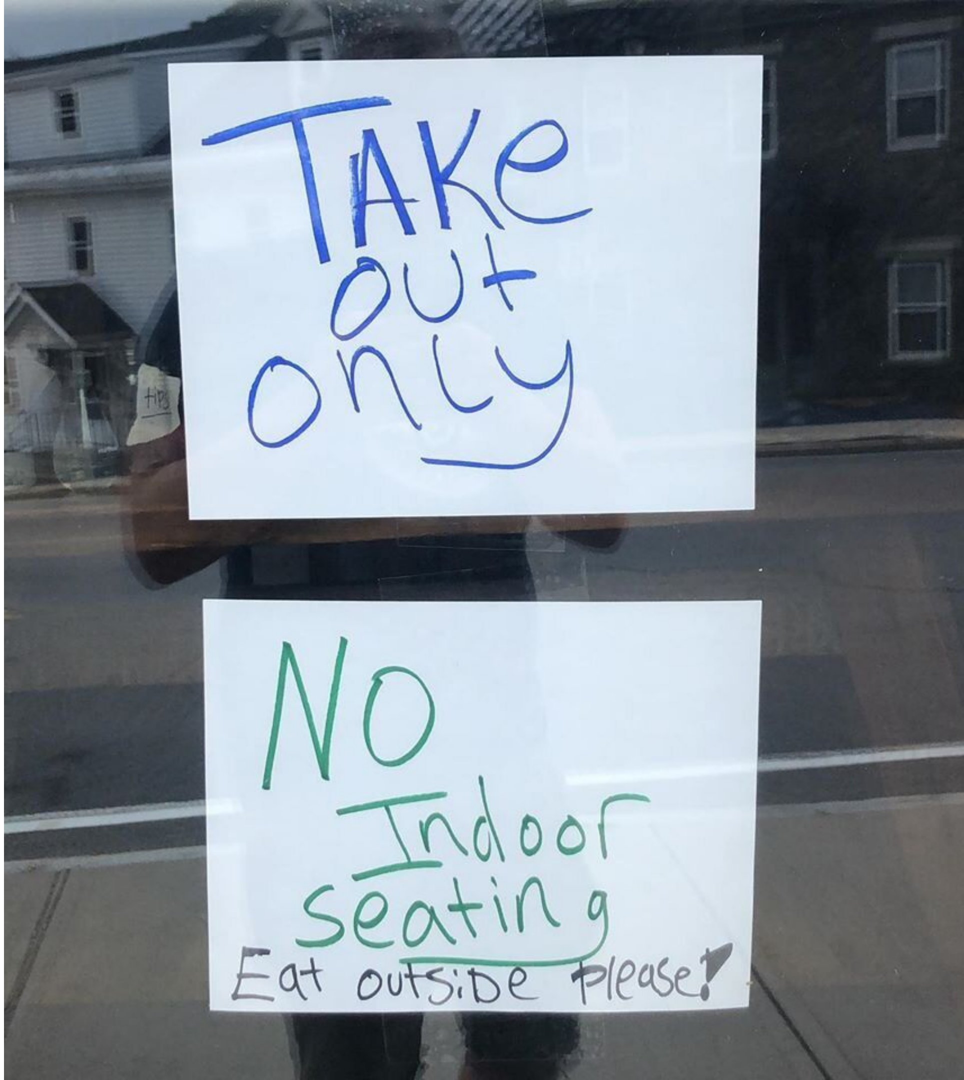
```
The lower quartile of vehicle sales is: 175.71175
The upper quartile of vehicle sales is: 209.072
The interquartile range of vehicle sales is: 33.36025000000001
The the median of vehicle sales is: 191.0985
```

COVID Years


Sales Numbers During COVID-19
Restaurant Sales
Vehicle Units
Sales(in Millions)
0
10000
20000
30000
40000
50000
60000
70000
80000
2020-01-01
2020-06-01
2021-01-01
2021-06-01
2022-01-01
2022-06-01
2023-01-01
Years

Analysis

- Inverse relationship with unemployment rates and consumer spending
    - High unemployment rates will not totally halt consumer spending
- Earlier months of the pandemic hit restaurant and vehicle sales the hardest
    - Our relationship to spending may have been strengthened after a couple months of being deprived of it
- Vehicle sales indicate people do not engage in making large purchases long after spikes in unemployment



Limitations

- May not be entirely representative
- Limited datasets available
- Different types of payments
- Lots of info cut down
- Inflation

fin