googleTrends

January 16, 2021

Concept and development: Ryan Worm

This document is a Jupyter Notebook (.ipynb) file containing both plain-text

[1]:
```
print(f'and code (HTML and Python),')
```

```
and code (HTML and Python),
```

downloaded to pdf via *LaTeX*. This project is ongoing and projections evolve as further data is gathered.

1 Introduction

The events of the last year have highlighted large data gaps in fields of healthcare and behavioural sciences. This paper proposes a mental distress index to objectively track social stress over time, and correlate it with various economic indicators to help forecast short and intermediate behavioural responses to stressors.

By using tested infodemiological principles this paper proposes models to estimate levels of percieved stress by the general population in three general areas: sleep, employment, and anxiety. Although those attributes have some interrelation alone, significant R^2 values to multiple economic indicators strengthen the proposed Mental Distress Index (MDI).

Recent models based on Google Trends data have proven to be more accurate in predicting deaths by suicide then traditional models (Parker et al., 2017; Lee, J., 2020). Building upon these results I wanted to create a more dynamic model that represented both acute and chronic stress adaptations (Sapolsky, 2004). Lee (2020) identified accounting for search lag more accurately predicted deaths by suicide, further supporting the importance of distinguising between acute and chronic stress manifestations.

The 2019 year was used as a baseline for all Google Trends data as this was the time period of the *Mental health characteristics and suicidal thoughts (Stats Canada, 2019)* dataset on which this paper derives its baseline data. Using 2019 as a baseline we can also estimate current trends against pre-pandemic data for better contrast. This may help provide greater insight on evolving health and social trends, forecasting future behaviours.

Mental distress will be used as a broad term encompassing any possible mental health condition commonly associated with stress, most commonly a combination of major depression and generalized anxiety disorder (Sapolsky, 2004). Mental health disorders are not simple, singular disease but complex heterogenous conditions.

This paper propses modeling methodology to estimate mental distress, as identified by *Statistics Canada: Mental health characteristics and suicidal thoughts (2019).* The number of individuals consulting with a health professional or experiencing suicidal thoughts will represent the number of individuals in mental distress in one year. The monthly mean of individuals experiencing MD is then multiplied by the monthly aggregate coefficient to calculate the monthly estimation of individuals suffering MD that month.

Key Findings There were multiple economic indicators which showed significant correlation to the proposed models. Distinct patterns of acute and chronic stress may inferred from the time adjusted and non-adjusted mental distress index and represent economical manifestations of physiological acute and chronic stress. The correlation to chronic stress on alcohol ($R^2 = .22$), healthcare ($R^2 = .05$) are greater then that of acute stress ($R^2 = .10$ and .04 accordingly). Acute stress was more strongly related to cannabis ($R^2 = .32$) and tobacco sales ($R^2 = .15$). It should be noted that tobacco had no relation to chronic stress ($R^2 = 0$) while cannabis still maintained a moderate correlation ($R^2 = .22$) with chronic stress.

2 Methodology

2.1 Google Trends

Google Trends debuted in 2015 and provides two optional time periods:

- Real-time data over the last seven days
- Non-realtime data going back to 2004 up to 36 hours from query time.

Recent studies have identified keyword correlation between increased monthly Google search volume (MGSV) and death by suicide, alcohol and drugs (Parker et al., 2017; Lee, J., 2020). Of eighty-nine related search words, Lee (2020) identified four key words that he identified as having the highest correlation to increases in death by suicide:

- "Generalzied Anxiety Disorder" (GAD)
- "Anxiety Disorder"
- "Insomnia"
- "Laid off"

For the purposes of this study only Google Trends non-realtime datasets will be used. This allows for year over year comparisons from which we can gather insight.

As *General Anxiety Disorder* and *Anxiety Disorder* were very similar, Generalized Anxiety Disorder (disorder) was used alone to allow for balanced weighing between the stressor keywords. This allows the mental distress models to encompass a broader scope of stressors. The keywords for modeling coefficients were:

- GAD (disorder)
- Insomnia (disorder)
- Laid Off (keyword)

Lee, J. (2020) noted a search lag between increased MGSV keyword searches and deaths related to suicide. To explore this difference two Google Trends algorithms will be used; one with a search lag applied and one without.

This functionally creates two different models: one using preceding months and another the data of the corresponding month. These delayed and acute models may represent an economical manifestation of the acute and delayed stress response as discussed by Sapolsky (2004).

- GAD; 3 month lag
- Laid off; 2 month lag
- Insomnia; 1 month lag

Correlations between Mental Distress Index and Alcohol, Tobacco, Cannabis, Healthcare, and general GDP (excl. Cannabis) will also be evaluated to identify relationships between the impact of social stress and purchasing habits as represented by industry specific GDP or revenue.

It should be noted that Google calculates a relative interest score rather then counting absolute search totals. This is to maintain validity over time as the user base expands. Google Trends data as proven to be a valuable tool in the field of infodemiology, with over 88% of the search engine market as of October 2020. (Statista, 2020)

2.2 Data Processing

2.2.1 Data sources

- Google Trends
- Mental health characteristics and suicidal thoughts (Stats Canada, 2019)*
- Gross domestic product by industry, Sept 2020 (Stats Canada, 2020)
- Tobacco, sales and inventories, monthly production (x1000) (Stats Canada, 2020)
- Monthly retail sales of beer, wine and liquor stores in Canada from 2015 to 2020 (in billion Canadian dollars) (Statista, 2020)

*Baseline data for establishing the proposed mental distress index was established by adding 'Consultation with a health professional about emotional or mental health' and 'Suicidal thoughts (15 years and over)' together.

2.2.2 Preprocessing

- Calculating monthly mean Google Trends relative interest score of each relevant term from Oct 2017 - Jan 2021
- Adjusted and non-adjusted aggregate Google Trends relative interest calculated
- Baseline coefficients for 2019 established
- Baseline coefficients for 2018 and 2020 calculated
- Multiplied relative Time-adjusted and Non-adjusted coefficients by monthly baselines to calculate monthly estimate
- Tobacco sales, Industry (excl. Cannabis), Cannabis, Healthcare, Alcohol sales

	A	B	C	D	E	F	G	H	I	J	K
	Date	Gen Anx	Sleep	Laid Off	mGSV (Adj)	mGSV (Non-adj)	Coeff (Adj)	Coeff (Non-adj)	Vulnerabilities (Adj)	Vulnerabilities (Non-adj)	Tobacco Sales (x1000)
	Jan-18	31.51612903	49.77419355	34.32258065	39.40645161	38.53763441	0.821618048	0.845382676	587566.4533	604561.3313	1,325,425
	Feb-18	38.64285714	56.96551724	32.82142857	37.55985663	42.80993432	0.833726557	0.950809351	596225.652	679955.4607	1,383,982
	Mar-18	39.83870968	57.19354839	23.06451613	39.93474231	40.03225806	0.859050041	0.895067169	614335.3195	640092.3684	1,667,976
	Apr-18	38.06666667	51.56666667	28.56666667	40.51036866	39.4	0.892454361	0.829741997	638223.8618	593376.1598	1,568,080
	May-18	50.25806452	64.58064516	33.77419355	37.75801331	49.53763441	0.901238392	1.32130214	644505.6156	944907.2036	2,002,748
	Jun-18	45.23333333	47.43333333	23.36666667	44.32867384	38.67777778	1.050575116	0.934682778	751301.2847	668422.8106	1,785,998
	Jul-18	38	48.90322581	26.80645161	39.75806452	37.90322581	0.875977194	0.948481545	626440.4909	678290.7686	1,802,864
	Aug-18	23.61290323	47.74193548	24.93548387	40.84265233	32.09677419	1.072991402	0.802383381	767331.9182	573811.1021	1,999,349
	Sep-18	31.9	55.8	20.66666667	39.92724014	36.12222222	1.165569736	0.756983534	833537.7706	541344.158	1,502,671
	Oct-18	46.41935484	55.64516129	35.22580645	39.57849462	45.76344086	0.823635562	1.0638848	589009.2452	760819.4832	1,712,575
	Nov-18	41.87087	57.12903	29.7	33.30824373	42.89996667	0.769715051	0.975316959	550448.89	697481.6682	1,690,928
	Dec-18	46.8	67.76667	27.76471	41.41827882	47.44379333	0.826617556	0.955772402	591141.768	683504.7036	1,645,486
	Jan-19	48.74194	65.51613	22.5	47.96200828	45.58602333	1.089794464	1.039954243	779348.3475	743705.9444	1,277,868
	Feb-19	39.3	70.16129	25.6129	45.05057	45.02473	1.023640659	1.027149455	732039.5568	734548.8134	1,271,357
	Mar-19	30.19355	61.82143	42.16129	46.48709667	44.72542333	1.056281469	1.02032137	755382.0882	729665.8223	1,448,170
	Apr-19	49.87097	44.22581	48.35714	45.39209	47.48464	1.031400689	1.083267397	737589.0127	774680.6248	1,543,485
	May-19	27.89286	48.03333	36.54839	41.8957	37.49152667	0.951955591	0.855294439	680775.1747	611649.5629	1,768,670
	Jun-19	32.76667	49.74194	41.63333	42.19467333	41.38064667	0.958748873	0.944016958	685633.2776	675097.9941	1,607,691
	Jul-19	49.8	44.66667	25.41935	45.3871	39.96200667	1.031287306	0.911653515	737507.9287	651953.8169	1,699,265
	Aug-19	50.25806	44.58065	25.16667	38.06428667	40.00179333	0.86489808	0.912561168	618517.447	652602.9102	1,735,092
	Sep-19	50.83333	69.19355	23.12903	34.25555667	47.71863667	0.778355981	1.088605565	556628.3073	778498.1263	1,494,682
	Oct-19	39.19355	56.43333	33.41935	48.05340667	43.01541	1.091871221	0.981310825	780833.506	701768.0811	1,590,038
	Nov-19	33.22581	66.06452	32.66667	43.27347333	43.98566667	0.983261405	1.003445296	703163.0064	717597.1795	1,779,303
	Dec-19	53.2	60.6	35.11765	50.10573333	49.63921667	1.138504261	1.132419769	814182.3472	809831.124	1,366,580
	Jan-20	40.06452	59.77419	46.33333	44.15340667	48.72401333	0.920591282	1.068836669	658345.5118	764360.7299	1,287,983
	Feb-20	50.3	61.32258	38.23333	42.70588333	49.95197	0.94795434	1.109434082	677913.7467	793393.2933	1,380,956
	Mar-20	28.51613	64.75862	38.90323	53.61863667	44.05932667	1.153409021	0.985106979	824841.2381	704482.8376	1,769,508
	Apr-20	55.70968	53.77419	42.37931	47.68549	50.62106	1.050524221	1.066051254	751264.8881	762368.7865	1,309,294
	May-20	43.31034	47.13333	33.41935	47.65914	41.28767333	1.137566385	1.101253457	813511.6409	787543.0552	1,576,457
	Jun-20	51.12903	74.6129	41.46667	39.34292333	55.7362	0.932414455	1.346914669	666800.6571	963223.5768	1,821,939

2.2.3 Processing

Importing dependancies

[2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from numpy import nan
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
import seaborn as sns
import scipy
```

Training models

Reading in cleaned dataset

[3]:
```
db = pd.read_excel("google_trends_db.xlsx")
```

Calculating mental distress indexes

Setting date ranges

[4]:
```
startdate_18 = '2018-01-01'
startdate_19 = '2019-01-01'
startdate_20 = '2020-01-01'
startdate_21 = '2021-01-01'
enddate_18 = '2018-12-31'
enddate_19 = '2019-12-31'
enddate_20 = '2020-12-31'
enddate_21 = '2021-12-31'
```

2018 MDI Estimations and script

[5]:
```
mask_2018 = (db['Date'] >= startdate_18) & (db['Date'] <= enddate_18)
db_2018 = db.loc[mask_2018]
non_distress_2018 = db_2018['Vulnerabilities (Non-adj)'].sum()
adj_distress_2018 = db_2018['Vulnerabilities (Adj)'].sum()
print(f'The adjusted model predicts there were {f"{round(adj_distress_2018,1):
 ↪,}"} Canadians who dealt with a mental health episode in 2018.')
print(f'The non-adjusted model predicts there were␣
 ↪{f"{round(non_distress_2018,1):,}"} Canadians who dealt with a mental health␣
 ↪episode in 2018.')
```

```
The adjusted model predicts there were 7,790,068.3 Canadians who dealt with a
mental health episode in 2018.
The non-adjusted model predicts there were 8,066,567.2 Canadians who dealt with
a mental health episode in 2018.
```

2019 Baseline confirmation

[6]:
```
mask_2019 = (db['Date'] >= startdate_19) & (db['Date'] <= enddate_19)
db_2019 = db.loc[mask_2019]
non_distress_2019 = db_2019['Vulnerabilities (Non-adj)'].sum()
adj_distress_2019 = db_2019['Vulnerabilities (Adj)'].sum()
print(f'The adjusted model confirms our baseline of␣
 ↪{f"{round(adj_distress_2019,1):,}"} individuals who experienced a mental␣
 ↪health episode in 2019.')
print(f'The non-adjusted model confirms our baseline of␣
 ↪{f"{round(non_distress_2019,1):,}"} individuals who experienced a mental␣
 ↪health episode in 2019.')
```

```
The adjusted model confirms our baseline of 8,581,600.0 individuals who
experienced a mental health episode in 2019.
The non-adjusted model confirms our baseline of 8,581,600.0 individuals who
experienced a mental health episode in 2019.
```

2020 MDI Estimations and script

[7]:
```
mask_2020 = (db['Date'] >= startdate_20) & (db['Date'] <= enddate_20)
db_2020 = db.loc[mask_2020]
non_distress_2020 = db_2020['Vulnerabilities (Non-adj)'].sum()
adj_distress_2020 = db_2020['Vulnerabilities (Adj)'].sum()
print(f'The adjusted model predicts there were {f"{round(adj_distress_2020,1):
 ↪,}"} individuals who dealt with a mental health episode in 2020.')
print(f'The non-adjusted model predicts there were␣
 ↪{f"{round(non_distress_2020,1):,}"} individuals who dealt with a mental␣
 ↪health episode in 2020.')
```

```
The adjusted model predicts there were 9,789,138.0 individuals who dealt with a
mental health episode in 2020.
```

The non-adjusted model predicts there were 9,821,683.3 individuals who dealt with a mental health episode in 2020.

2021 MDI Estimations and script

[8]:
```
mask_2021 = (db['Date'] >= startdate_21) & (db['Date'] <= enddate_21)
db_2021 = db.loc[mask_2021]
non_distress_2021 = db_2021['Vulnerabilities (Non-adj)'].sum()
adj_distress_2021 = db_2021['Vulnerabilities (Adj)'].sum()
```

Isolating Jan 2018-2020 series

[9]:
```
jan_2018 = (db['Date'] == startdate_18)
db_jan2018 = db.loc[jan_2018]
jan_2019 = (db['Date'] == startdate_19)
db_jan2019 = db.loc[jan_2019]
jan_2020 = (db['Date'] == startdate_20)
db_jan2020 = db.loc[jan_2020]
jan_2021 = (db['Date'] == startdate_21)
db_jan2021 = db.loc[jan_2021]
```

Calculating naMDI for January 2018-2021

[10]:
```
non_jan18 = db_jan2018['Vulnerabilities (Non-adj)'].sum()
non_jan19 = db_jan2019['Vulnerabilities (Non-adj)'].sum()
non_jan20 = db_jan2020['Vulnerabilities (Non-adj)'].sum()
non_jan21 = db_jan2021['Vulnerabilities (Non-adj)'].sum()
```

[11]:
```
adj_jan18 = db_jan2018['Vulnerabilities (Adj)'].sum()
adj_jan19 = db_jan2019['Vulnerabilities (Adj)'].sum()
adj_jan20 = db_jan2020['Vulnerabilities (Adj)'].sum()
adj_jan21 = db_jan2021['Vulnerabilities (Adj)'].sum()
```

naMDI January 2021 vs January 2019 (Baseline)

[12]:
```
non21v19 = non_jan21 / non_jan19
rnd_non21v19 = round((non21v19*100),2)
rnd_non21v19_ = round((non21v19*100) - 100,2)
```

[13]:
```
adj21v19 = adj_jan21 / adj_jan19
rnd_adj21v19 = round((adj21v19*100),2)
rnd_adj21v19_ = round((adj21v19*100) - 100,2)
```

Soft-coding stress forecasting script

[14]:
```
print(f'The adjusted model predicts there will be␣
 ↪{f"{round(adj_distress_2021,1):,}"} individuals dealing with a mental health␣
 ↪episode in January 2021.')
```

```
print(f'The non-adjusted model predicts there will be␣
 ↪{f"{round(non_distress_2021,1):,}"} individuals dealing with a mental health␣
 ↪episode in January 2021.')
print(f'Current naMDI modeling indicate that January 2021 is {rnd_non21v19}% as␣
 ↪stressful as January 2019 for the general Canadian population, while taMDI␣
 ↪modeling suggests it will be {rnd_adj21v19}% as stressful.')
```

```
The adjusted model predicts there will be 844,543.2 individuals dealing with a
mental health episode in January 2021.
The non-adjusted model predicts there will be 726,300.3 individuals dealing with
a mental health episode in January 2021.
Current naMDI modeling indicate that January 2021 is 97.66% as stressful as
January 2019 for the general Canadian population, while taMDI modeling suggests
it will be 108.37% as stressful.
```

Calculating year-over-year estimates for taMDI and naMDI modeling

[15]:
```
adj_yoy_2020 = adj_distress_2020 - adj_distress_2019
adj_pcnt_2020 = adj_distress_2020/adj_distress_2019
non_yoy_2020 = non_distress_2020 - non_distress_2019
non_pcnt_2020 = non_distress_2020/non_distress_2019
```

Soft-coding prediction script

[16]:
```
print(f'Adjusted MDI predicts that 2020 saw an increase of␣
 ↪{f"{round(adj_yoy_2020,2):,}"} ({round(adj_pcnt_2020*100,2)}%) Canadians who␣
 ↪experienced a serious mental health episode compared to 2019.')
print(f'Non-adjusted MDI predicts that 2020 saw an increase of␣
 ↪{f"{round(non_yoy_2020,2):,}"} ({round(non_pcnt_2020*100,2)}%) Canadians who␣
 ↪experienced a serious mental health episode compared to 2019.')
```

```
Adjusted MDI predicts that 2020 saw an increase of 1,207,537.95 (114.07%)
Canadians who experienced a serious mental health episode compared to 2019.
Non-adjusted MDI predicts that 2020 saw an increase of 1,240,083.35 (114.45%)
Canadians who experienced a serious mental health episode compared to 2019.
```

Creating labels and arrays for chart

[17]:
```
dist_labels = ["2018", "2019", "2020", "Jan 2021 (forecast)"]
non_dist_df = [non_distress_2018, non_distress_2019, non_distress_2020 ,␣
 ↪non_distress_2021]
adj_dist_df = [adj_distress_2018, adj_distress_2019, adj_distress_2020 ,␣
 ↪adj_distress_2021]
```

Creating and storing model comparison dataframe

[18]:
```
dist_d = {'Model':["Time-adjusted", "Non-adjusted"],
      '2018':[(f"{round(adj_distress_2018,2):,}"),␣
 ↪(f"{round(non_distress_2018,2):,}")],
```

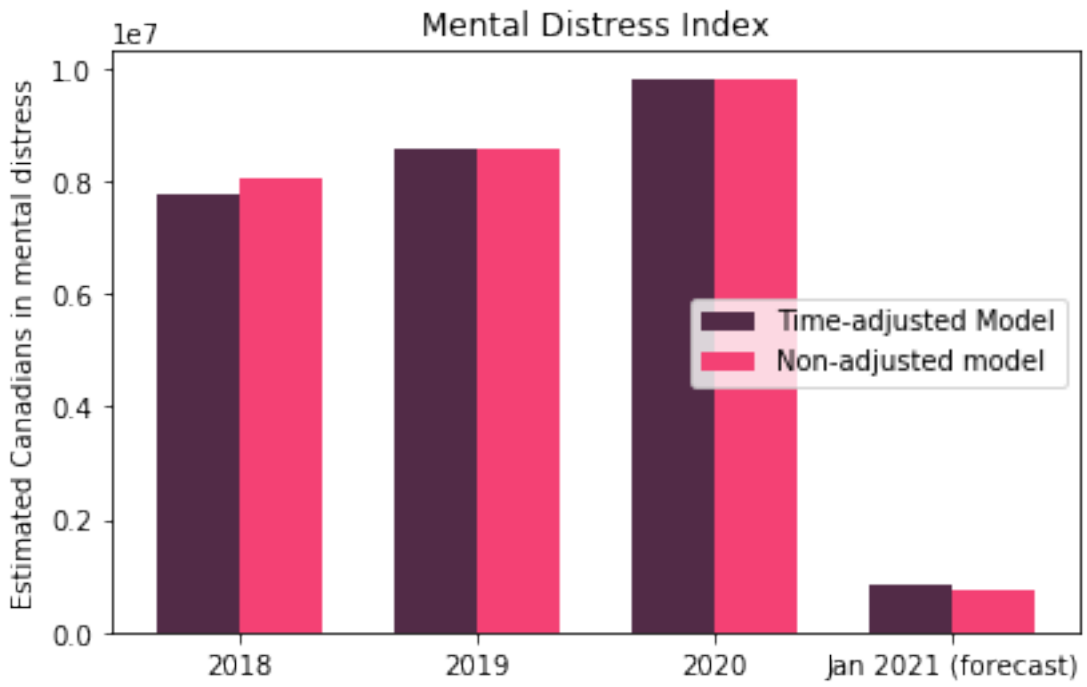
```
      '2019':[(f"{round(adj_distress_2019,2):,}"),␣
 ↪(f"{round(non_distress_2019,2):,}")],
      '2020':[(f"{round(adj_distress_2020,2):,}"),␣
 ↪(f"{round(non_distress_2020,2):,}")],
      '2021':[(f"{round(adj_distress_2021,2):,}"),␣
 ↪(f"{round(non_distress_2021,2):,}")]
     }
dist_df = pd.DataFrame(data=dist_d)
dist_df
```

[18]:

	Model	2018	2019	2020	2021
0	Time-adjusted	7,790,068.27	8,581,600.0	9,789,137.95	844,543.21
1	Non-adjusted	8,066,567.22	8,581,600.0	9,821,683.35	726,300.28

Plotting grouped bar plot for model comparison

[19]:
```
x = np.arange(len(dist_labels))
width = 0.35
fig, ax = plt.subplots(figsize=(6.5, 4.0))
rect_adj = ax.bar(x - width/2, adj_dist_df, width, color='#522B47',␣
 ↪label='Time-adjusted Model')
rect_dist = ax.bar(x + width/2, non_dist_df, width, color='#F44174', ␣
 ↪label='Non-adjusted model')
ax.set_ylabel('Estimated Canadians in mental distress')
ax.set_title('Mental Distress Index')
ax.set_xticks(x)
ax.set_xticklabels(dist_labels)
ax.legend(loc='center right')
plt.xticks(rotation=0)
plt.savefig('./dist_chart.png', bbox_inches="tight")
```



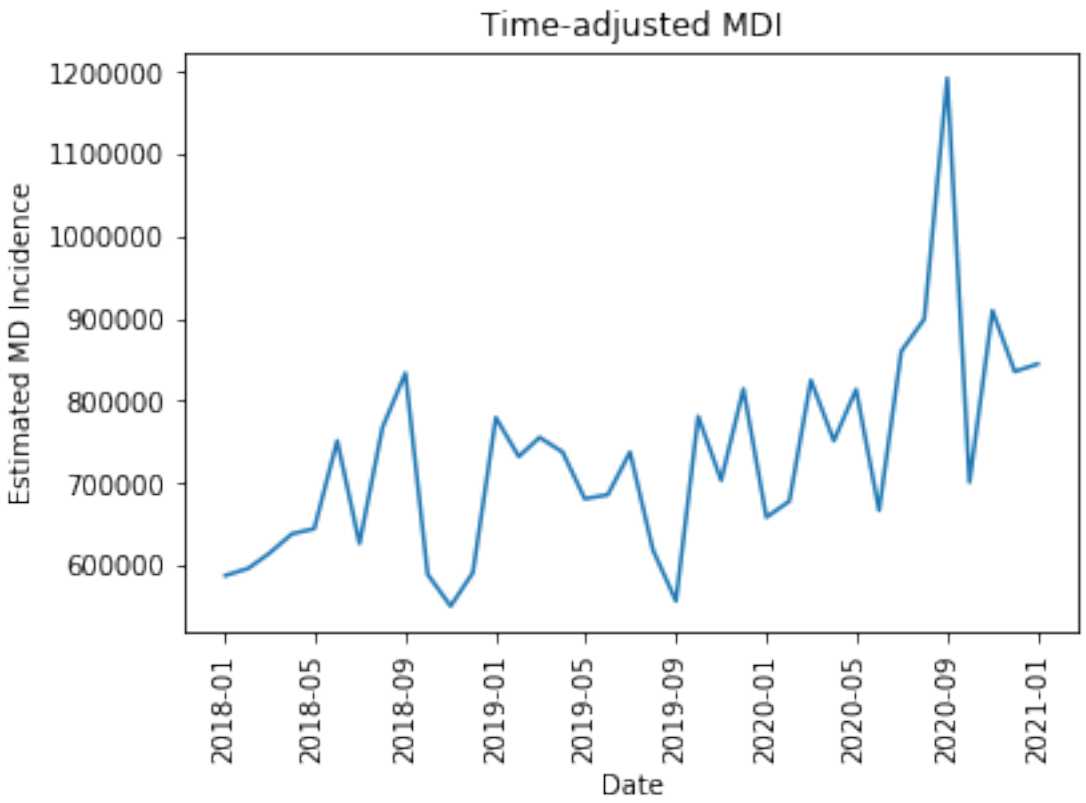
taMDI & naMDI plots

Storing date and estimations into variables

[20]:
```
date = db["Date"]
vul_adj = db["Vulnerabilities (Adj)"]
vul_non = db["Vulnerabilities (Non-adj)"]
```

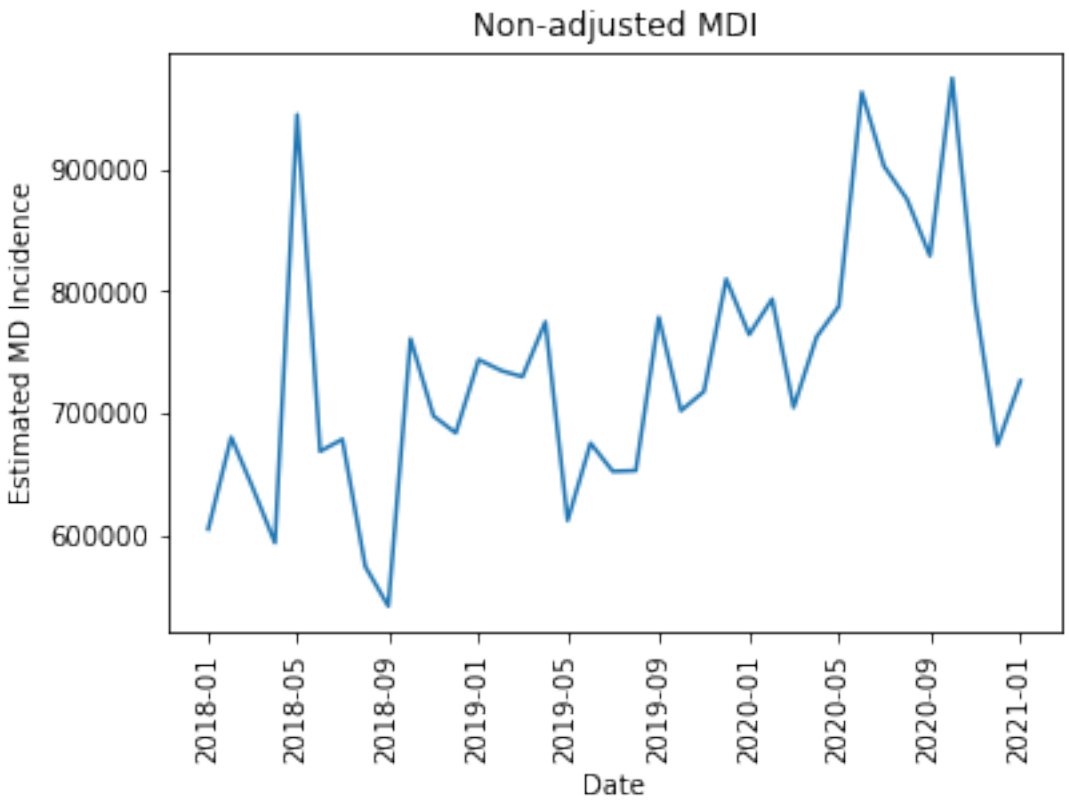
Plotting taMDI

[21]:
```
plt.plot(date,  vul_adj)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("Estimated MD Incidence")
plt.title('Time-adjusted MDI')
plt.savefig('./aMDI.png', bbox_inches="tight")
```



Plotting naMDI

[22]:
```
plt.plot(date, vul_non)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("Estimated MD Incidence")
plt.title('Non-adjusted MDI')
plt.savefig('./naMDI.png', bbox_inches="tight")
```



2.2.4 Economic indicators

Alcohol Sales

Dropping null values

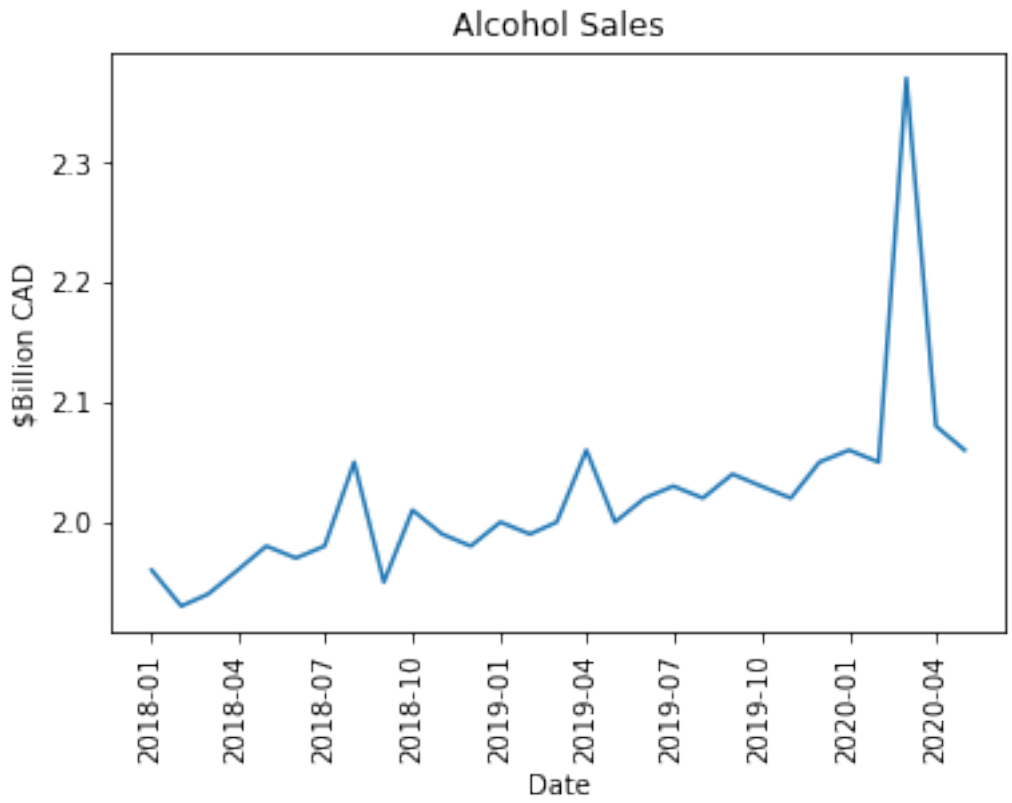
[23]:
```
alcohol_db = db[['Date', 'Alcohol Sales (Billion)']].copy()
alcohol_db2 = alcohol_db.dropna(axis=0)
```

Setting x and y values

[24]:
```
alcohol_date = alcohol_db2["Date"]
alcohol = alcohol_db2["Alcohol Sales (Billion)"]
```

Plotting and saving as .png

[25]:
```
plt.plot(alcohol_date, alcohol)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("$Billion CAD")
plt.title('Alcohol Sales')
plt.savefig('./alcohol.png', bbox_inches="tight")
```



Cigarette Sales

Dropping null values

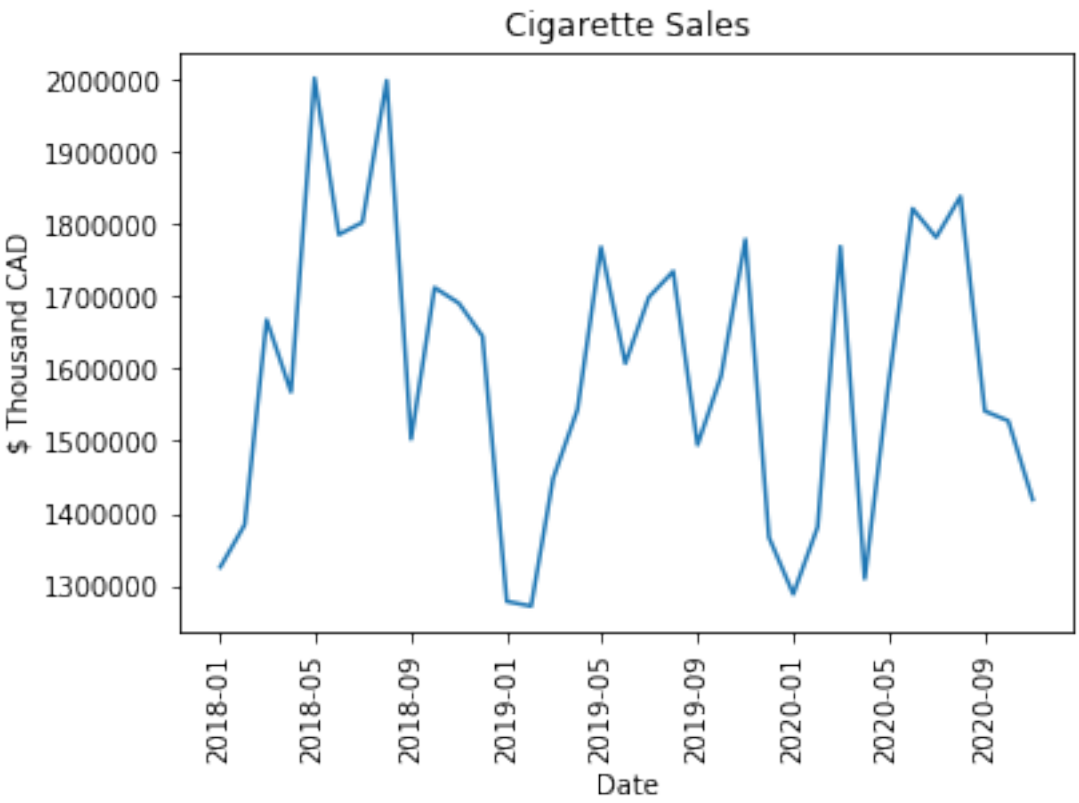
[26]:
```
cig_db = db[['Date', 'Tobacco Sales (x1000)']].copy()
cig_db2 = cig_db.dropna(axis=0)
```

Setting x and y values

[27]:
```
cig_date = cig_db2["Date"]
cig = cig_db2["Tobacco Sales (x1000)"]
```

Plotting

[28]:
```
plt.plot(cig_date, cig)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("$ Thousand CAD")
plt.title("Cigarette Sales")
plt.savefig('./cig.png', bbox_inches="tight")
```



Industry (excl. Cannabis)

Dropping null values

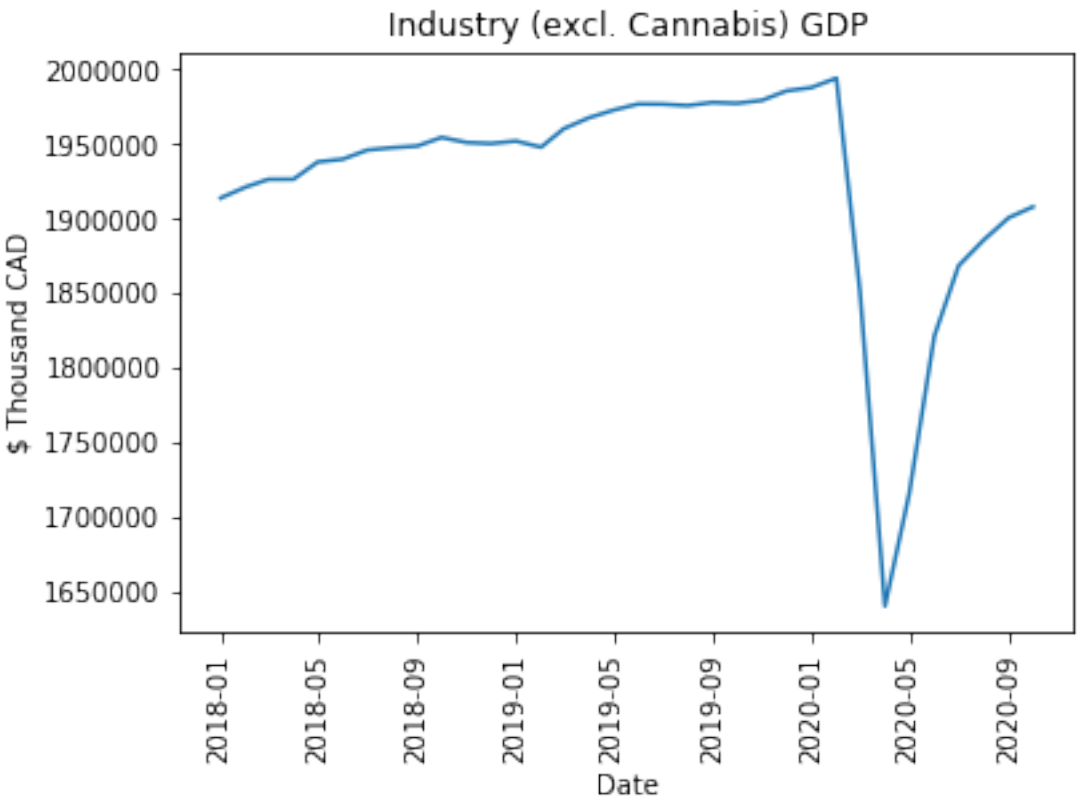
[29]:
```
ind_db = db[['Date', 'All industry (ex. Cannabis)']].copy()
ind_db2 = ind_db.dropna(axis=0)
```

Setting x and y values

[30]:
```
ind_date = ind_db2["Date"]
industry = ind_db2["All industry (ex. Cannabis)"]
```

Plotting and saving as .png

[31]:
```
plt.plot(ind_date, industry)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("$ Thousand CAD")
plt.title("Industry (excl. Cannabis) GDP")
plt.savefig('./ind.png', bbox_inches="tight")
```



Cannabis

Dropping null values

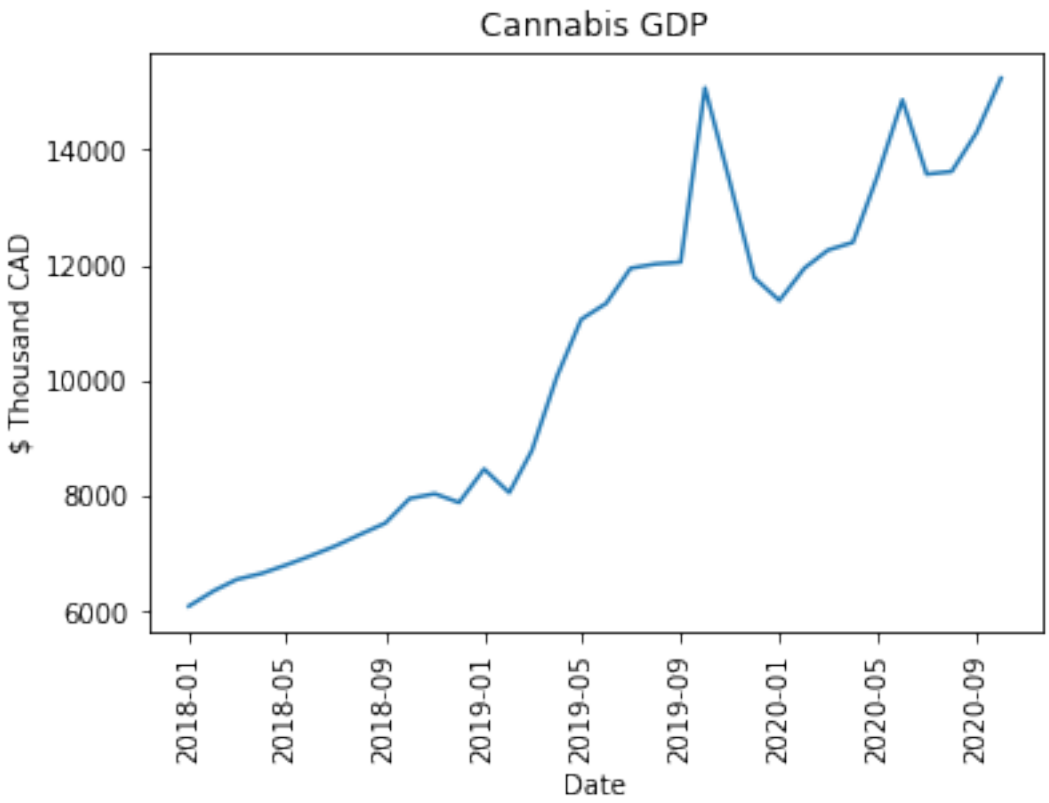
[32]:
```
cannabis_db = db[['Date', 'Cannabis Sector']].copy()
cannabis_db2 = cannabis_db.dropna(axis=0)
```

Setting x and y values

[33]:
```
cannabis_date = cannabis_db2["Date"]
cannabis = cannabis_db2["Cannabis Sector"]
```

Plotting and saving as .png

[34]:
```
plt.plot(cannabis_date, cannabis)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("$ Thousand CAD")
plt.title("Cannabis GDP")
plt.savefig('./cann.png', bbox_inches="tight")
```



Healthcare and Social Services

Dropping the null values

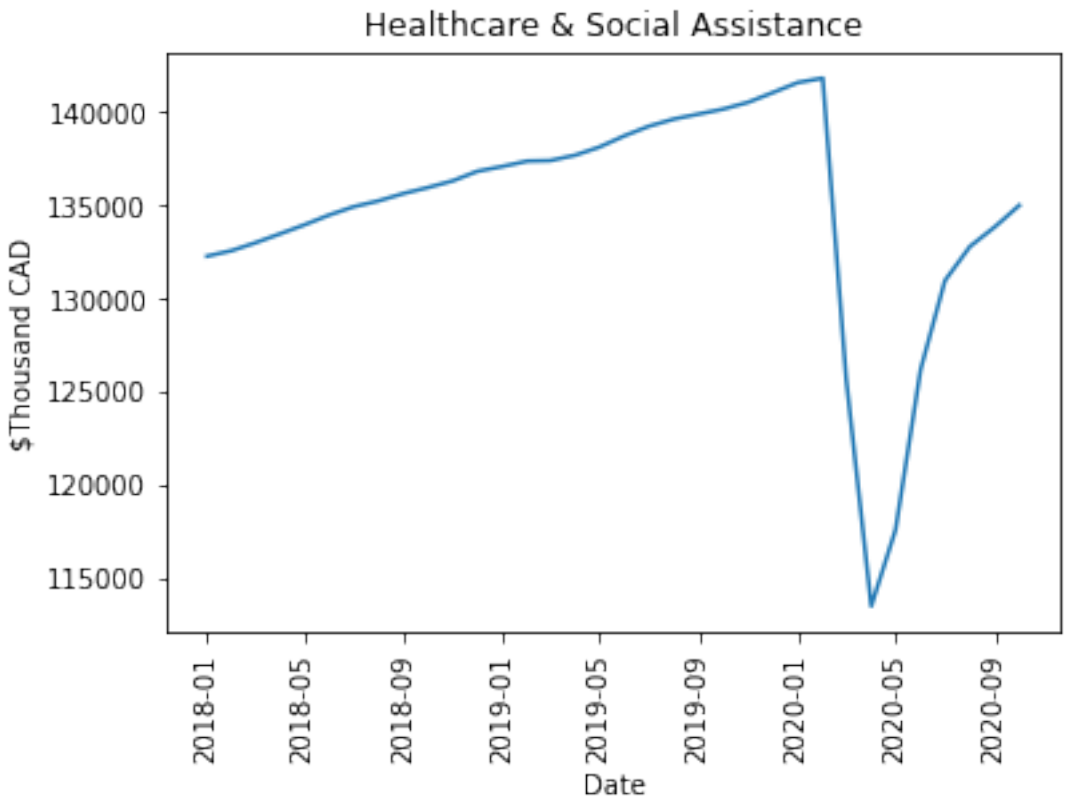
[35]:
```
healthcare_db = db[['Date', 'Healthcare & Social Assistance']].copy()
healthcare_db2 = healthcare_db.dropna(axis=0)
```

Setting the x and y values

[36]:
```
healthcare_date = healthcare_db2["Date"]
healthcare = healthcare_db2["Healthcare & Social Assistance"]
```

Plotting data

[37]:
```
plt.plot(healthcare_date, healthcare)
plt.xlabel("Date")
plt.xticks(rotation=90)
plt.ylabel("$Thousand CAD")
plt.title("Healthcare & Social Assistance")
plt.savefig('./health.png', bbox_inches="tight")
```



2.3 Calculating coefficients of determination

2.3.1 Stress and alcohol

Dropping null values

[38]:
```
alc_mdi_db = db[['Date', 'Vulnerabilities (Adj)', 'Vulnerabilities (Non-adj)',␣
↪'Alcohol Sales (Billion)']]
alc_mdi = alc_mdi_db.dropna(axis=0)
```

Transposing array for linear regression

[39]:
```
alcohol = alc_mdi["Alcohol Sales (Billion)"]
vul_adj = alc_mdi["Vulnerabilities (Adj)"]
vul_non = alc_mdi["Vulnerabilities (Non-adj)"]
x_stress_adj = np.array(vul_adj).reshape((-1, 1))
x_stress_non = np.array(vul_non).reshape((-1, 1))
```

Fitting dependant variable (alcohol) to array

[40]:
```
y_alcohol = np.array(alcohol)
```

Fitting stress and alcohol to linear regression model

[41]:
```
stress_alcohol_model_adj = LinearRegression().fit(x_stress_adj, y_alcohol)
stress_alcohol_model_non = LinearRegression().fit(x_stress_non, y_alcohol)
```

Storing rounded linear R^2 value for f-string notation

[42]:
```
linear_r_adj_alcohol = round(stress_alcohol_model_adj.score(x_stress_adj,␣
↪y_alcohol),2)
linear_r_non_alcohol = round(stress_alcohol_model_non.score(x_stress_non,␣
↪y_alcohol),2)
```

Transforming to fit independant variable

[43]:
```
transformer = PolynomialFeatures(degree=2, include_bias=False)

transformer_adj = transformer.fit(x_stress_adj)
transformer_non = transformer.fit(x_stress_non)

x_stress_adj_ = transformer_adj.transform(x_stress_adj)
x_stress_non_ = transformer_non.transform(x_stress_non)
```

Fitting polynomial regression model

[44]:
```
alcohol_poly_model_adj = LinearRegression().fit(x_stress_adj_, y_alcohol)
alcohol_poly_model_non = LinearRegression().fit(x_stress_non_, y_alcohol)
```

Scoring, rounding and storing R^2 values for alcohol

[45]:
```
poly_r_adj_alcohol = round(alcohol_poly_model_adj.score(x_stress_adj_,␣
↪y_alcohol),2)
poly_r_non_alcohol = round(alcohol_poly_model_non.score(x_stress_non_,␣
↪y_alcohol),2)
```

2.3.2 Stress and cannabis

Dropping null values

[46]:
```
cann_mdi_db = db[['Date', 'Vulnerabilities (Adj)', 'Vulnerabilities (Non-adj)',␣
↪'Cannabis Sector']]
cann_mdi = cann_mdi_db.dropna(axis=0)
```

Transposing array for linear regression

[47]:
```
cannabis = cann_mdi["Cannabis Sector"]
vul_adj = cann_mdi["Vulnerabilities (Adj)"]
vul_non = cann_mdi["Vulnerabilities (Non-adj)"]
x_stress_adj = np.array(vul_adj).reshape((-1, 1))
x_stress_non = np.array(vul_non).reshape((-1, 1))
```

Fitting variable to (cannabis)

[48]:
```
y_cannabis = np.array(cannabis)
```

Fitting cannabis to linear regression model

[49]:
```
stress_cannabis_model_adj = LinearRegression().fit(x_stress_adj, y_cannabis)
stress_cannabis_model_non = LinearRegression().fit(x_stress_non, y_cannabis)
```

Scoring, rounding and storing linear R^2 values for cannabis

[50]:
```
linear_r_adj_cann= round(stress_cannabis_model_adj.score(x_stress_adj,␣
↪y_cannabis),2)
linear_r_non_cann = round(stress_cannabis_model_non.score(x_stress_non,␣
↪y_cannabis), 2)
```

Transforming to fit independant variables

[51]:
```
transformer = PolynomialFeatures(degree=2, include_bias=False)

transformer_adj = transformer.fit(x_stress_adj)
transformer_non = transformer.fit(x_stress_non)

x_stress_adj_ = transformer_adj.transform(x_stress_adj)
x_stress_non_ = transformer_non.transform(x_stress_non)
```

Fitting cannabis to polynomial regression

[52]:
```
cannabis_poly_model_adj = LinearRegression().fit(x_stress_adj_, y_cannabis)
cannabis_poly_model_non = LinearRegression().fit(x_stress_non_, y_cannabis)
```

Scoring, rounding and storing polynomial R^2 for cannabis

[53]:
```
poly_r_adj_cann = round(cannabis_poly_model_adj.score(x_stress_adj_,␣
↪y_cannabis),2)
poly_r_non_cann = round(cannabis_poly_model_non.score(x_stress_non_,␣
↪y_cannabis),2)
```

2.3.3 Stress and Healthcare Expenditure

Dropping null values

[54]:
```
health_mdi_db = db[['Date', 'Vulnerabilities (Adj)', 'Vulnerabilities␣
↪(Non-adj)', 'Healthcare & Social Assistance']]
health_mdi = health_mdi_db.dropna(axis=0)
```

Transposing independant variables

[55]:
```
health = health_mdi['Healthcare & Social Assistance']
vul_adj = health_mdi["Vulnerabilities (Adj)"]
```

```
vul_non = health_mdi["Vulnerabilities (Non-adj)"]
x_stress_adj = np.array(vul_adj).reshape((-1, 1))
x_stress_non = np.array(vul_non).reshape((-1, 1))
```

Setting dependant variable healthcare as array

[56]:
```
y_health = np.array(health)
```

Fitting healthcare to linear regression model

[57]:
```
stress_health_model_adj = LinearRegression().fit(x_stress_adj, y_health)
stress_health_model_non = LinearRegression().fit(x_stress_non, y_health)
```

Fitting to polynomial regression model

[58]:
```
linear_r_adj_health = round(stress_health_model_adj.score(x_stress_adj,␣
↪y_health),2)
linear_r_non_health = round(stress_health_model_non.score(x_stress_non,␣
↪y_health),2)
```

Transposing independant variables array

[59]:
```
transformer = PolynomialFeatures(degree=2, include_bias=False)

transformer_adj = transformer.fit(x_stress_adj)
transformer_non = transformer.fit(x_stress_non)

x_stress_adj_ = transformer_adj.transform(x_stress_adj)
x_stress_non_ = transformer_non.transform(x_stress_non)
```

Fitting healthcare to linear regression models

[60]:
```
health_poly_model_adj = LinearRegression().fit(x_stress_adj_, y_health)
health_poly_model_non = LinearRegression().fit(x_stress_non_, y_health)
```

Fitting healthcare to polynomial regression models

[61]:
```
poly_r_adj_health = round(health_poly_model_adj.score(x_stress_adj_,␣
↪y_health),2)
poly_r_non_health = round(health_poly_model_non.score(x_stress_non_,␣
↪y_health),2)
```

2.3.4 All industries (excl. Cannabis)

[62]:
```
ind_mdi_db = db[['Date', 'Vulnerabilities (Adj)', 'Vulnerabilities (Non-adj)',␣
↪'All industry (ex. Cannabis)']]
ind_mdi = ind_mdi_db.dropna(axis=0)
```

Transposing independant variables array

[63]:
```
ind = ind_mdi['All industry (ex. Cannabis)']
vul_adj = ind_mdi["Vulnerabilities (Adj)"]
vul_non = ind_mdi["Vulnerabilities (Non-adj)"]
x_stress_adj = np.array(vul_adj).reshape((-1, 1))
x_stress_non = np.array(vul_non).reshape((-1, 1))
```

Setting dependant variable as array

[64]:
```
y_ind = np.array(ind)
```

Fitting dependant variable to linear regression model

[65]:
```
stress_ind_model_adj = LinearRegression().fit(x_stress_adj, y_ind)
stress_ind_model_non = LinearRegression().fit(x_stress_non, y_ind)
```

Scoring, rounding and storing linear regression as variable

[66]:
```
linear_r_adj_ind = round(stress_ind_model_adj.score(x_stress_adj, y_ind),2)
linear_r_non_ind  = round(stress_ind_model_non.score(x_stress_non, y_ind),2)
```

Transposing independant variables

[67]:
```
transformer = PolynomialFeatures(degree=2, include_bias=False)

transformer_adj = transformer.fit(x_stress_adj)
transformer_non = transformer.fit(x_stress_non)


x_stress_adj_ = transformer_adj.transform(x_stress_adj)
x_stress_non_ = transformer_non.transform(x_stress_non)
```

Fitting industry (excl. cannabis) to polynomial regression model

[68]:
```
ind_poly_model_adj = LinearRegression().fit(x_stress_adj_, y_ind)
ind_poly_model_non = LinearRegression().fit(x_stress_non_, y_ind)
```

Scoring, rounding and storing R^2 value for polynomial model

[69]:
```
poly_r_adj_ind = round(ind_poly_model_adj.score(x_stress_adj_, y_ind),2)
poly_r_non_ind = round(ind_poly_model_non.score(x_stress_non_, y_ind),2)
```

2.3.5 Tobacco

Dropping null values

[70]:
```
cig_mdi_db = db[['Date', 'Vulnerabilities (Adj)', 'Vulnerabilities (Non-adj)',␣
↪'Tobacco Sales (x1000)']]
cig_mdi = cig_mdi_db.dropna(axis=0)
```

Transposing dependant variables

[71]:
```
cig = cig_mdi['Tobacco Sales (x1000)']
vul_adj = cig_mdi["Vulnerabilities (Adj)"]
vul_non = cig_mdi["Vulnerabilities (Non-adj)"]
x_stress_adj = np.array(vul_adj).reshape((-1, 1))
x_stress_non = np.array(vul_non).reshape((-1, 1))
```

Setting dependant variable as array

[72]:
```
y_cig = np.array(cig)
```

Fitting dependant variable cigarettes to linear regression model

[73]:
```
cig_health_model_adj = LinearRegression().fit(x_stress_adj, y_cig)
cig_health_model_non = LinearRegression().fit(x_stress_non, y_cig)
```

Scoring, rounding and storing as variable

[74]:
```
linear_r_adj_cig = round(cig_health_model_adj.score(x_stress_adj, y_cig),2)
linear_r_non_cig = round(cig_health_model_non.score(x_stress_non, y_cig),2)
```

Transposing independant variables

[75]:
```
transformer = PolynomialFeatures(degree=2, include_bias=False)

transformer_adj = transformer.fit(x_stress_adj)
transformer_non = transformer.fit(x_stress_non)


x_stress_adj_ = transformer_adj.transform(x_stress_adj)
x_stress_non_ = transformer_non.transform(x_stress_non)
```

Fitting dependant variable cigarettes to polynomial regression model

[76]:
```
cig_poly_model_adj = LinearRegression().fit(x_stress_adj_, y_cig)
cig_poly_model_non = LinearRegression().fit(x_stress_non_, y_cig)
```

Scoring, rounding, and storing R^2 polynomial values

[77]:
```
poly_r_adj_cig = round(cig_poly_model_adj.score(x_stress_adj_, y_cig),2)
poly_r_non_cig = round(cig_poly_model_non.score(x_stress_non_, y_cig),2)
```

2.3.6 Comparing predictive models

Setting labels and values

[78]:
```
labels = ["Healthcare", "Cannabis", "Industry (excl. Cannabis)", "Tobacco",␣
↪"Alcohol"]
```

```
linear_acute_corr = [linear_r_non_health, linear_r_non_cann, linear_r_non_ind,␣
 ↪linear_r_non_cig, linear_r_non_alcohol]
linear_delay_corr = [linear_r_adj_health, linear_r_adj_cann, linear_r_adj_ind,␣
 ↪linear_r_adj_cig , linear_r_adj_alcohol]
poly_acute_corr = [poly_r_non_health, poly_r_non_cann, poly_r_non_ind,␣
 ↪poly_r_non_cig, poly_r_non_alcohol]
poly_delayed_corr = [poly_r_adj_health, poly_r_adj_cann, poly_r_adj_ind,␣
 ↪poly_r_adj_cig, poly_r_adj_alcohol]
```

Dataframe comparing linear regression R^2 scores

[79]:
```
d1 = {'${R^2}$ (Linear)':["Acute", "Delayed"],
      'Healthcare':[linear_r_non_health, linear_r_adj_health],
      'Cannabis':[linear_r_non_cann, linear_r_adj_cann],
      'Industry (excl. Cannabis)':[linear_r_non_ind, linear_r_adj_ind],
      'Cigarettes':[linear_r_non_cig, linear_r_adj_cig],
      'Alcohol':[linear_r_non_alcohol, linear_r_adj_alcohol]
     }
df1 = pd.DataFrame(data=d1)
df1
```

[79]:

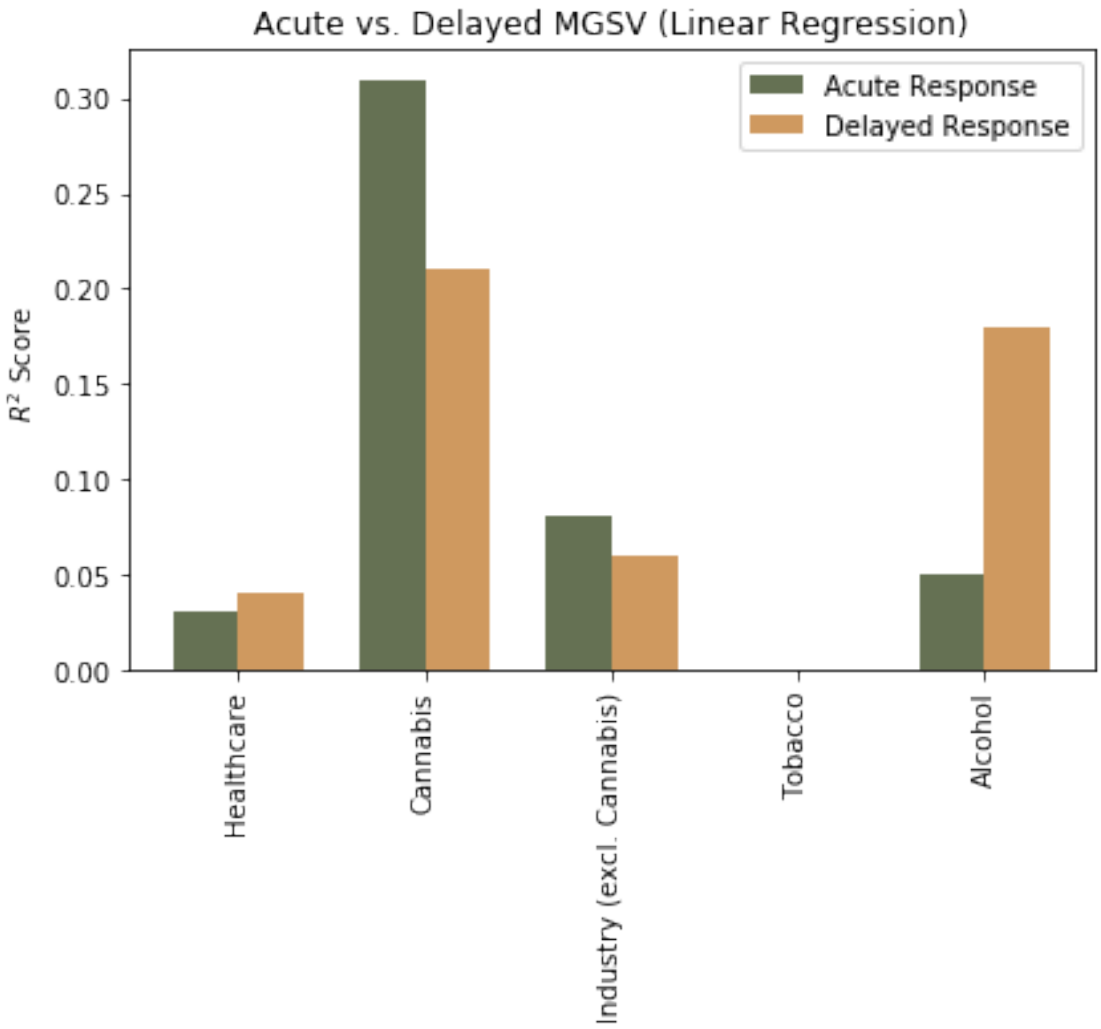
	${R^2}$ (Linear)	Healthcare	Cannabis	Industry (excl. Cannabis)	Cigarettes	Alcohol
0	Acute	0.03	0.31	0.08	0.0	0.05
1	Delayed	0.04	0.21	0.06	0.0	0.18

Plotting grouped bar chart for linear R^2 values

[80]:
```
x = np.arange(len(labels))  # the label locations
width = 0.35  # the width of the bars
fig, ax = plt.subplots()
rects1 = ax.bar(x - width/2, linear_acute_corr, width, color='#657153',␣
 ↪label='Acute Response')
rects2 = ax.bar(x + width/2, linear_delay_corr, width, color='#CF995F',␣
 ↪label='Delayed Response')

ax.set_ylabel('$R^{2}$ Score')
ax.set_title('Acute vs. Delayed MGSV (Linear Regression)')
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend()


def autolabel(rects):
```

```
    """Attach a text label above each bar in *rects*, displaying its height."""
    for rect in rects:
        height = rect.get_height()
        ax.annotate('{}'.format(height),
                    xy=(rect.get_x() + rect.get_width() / 2, height),
                    xytext=(0, 3),  # 3 points vertical offset
                    textcoords="offset points",
                    ha='center', va='bottom')


fig.tight_layout()
plt.xticks(rotation=90)
print(f'Linear regression scores of the acute response (naMDI) and delayed␣
 ↪response (taMDI) against the economic indicators.')
plt.savefig('./line_chart.png', bbox_inches="tight")
```

```
Linear regression scores of the acute response (naMDI) and delayed response
(taMDI) against the economic indicators.
```



Dataframe comparing polynomial R^2 values

[81]:
```
d3 = {'${R^2}$ (Polynomial)':["Acute", "Delayed"],
      'Healthcare':[poly_r_non_health,poly_r_adj_health],
      'Cannabis':[poly_r_non_cann, poly_r_adj_cann],
      'Industry (excl. Cannabis)':[poly_r_non_ind, poly_r_adj_ind],
      'Cigarettes':[poly_r_non_cig, poly_r_adj_cig],
      'Alcohol':[poly_r_non_alcohol, poly_r_adj_alcohol]
     }
df3 = pd.DataFrame(data=d3)
df3
```

[81]:

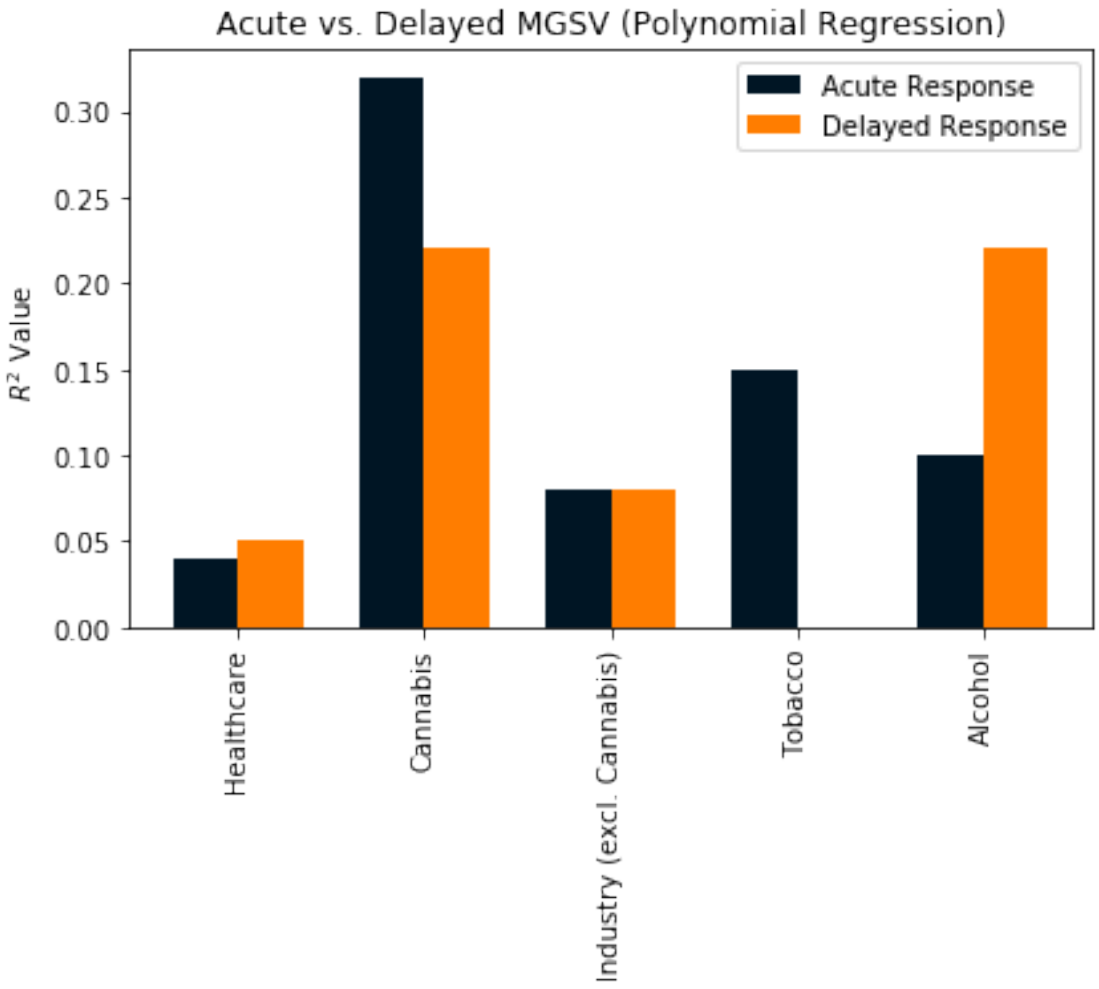
	${R^2}$ (Polynomial)	Healthcare	Cannabis	Industry (excl. Cannabis)	\
0	Acute	0.04	0.32	0.08	
1	Delayed	0.05	0.22	0.08	

```
   Cigarettes  Alcohol
0        0.15     0.10
1        0.00     0.22
```

Plotting grouped barchart for polynomial R^2 values

[82]:
```
x = np.arange(len(labels))  # the label locations
width = 0.35  # the width of the bars
fig, ax = plt.subplots(figsize=(6.5, 4.0))
rects3 = ax.bar(x - width/2, poly_acute_corr, width, color='#001524',␣
↪label='Acute Response')
rects4 = ax.bar(x + width/2, poly_delayed_corr, width, color='#FF7D00',␣
↪label='Delayed Response')
# Add some text for labels, title and custom x-axis tick labels, etc.
ax.set_ylabel('$R^{2}$ Value')
ax.set_title('Acute vs. Delayed MGSV (Polynomial Regression)')
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend()

def autolabel(rects):
    """Attach a text label above each bar in *rects*, displaying its height."""
    for rect in rects:
        height = rect.get_height()
        ax.annotate('{}'.format(height),
                    xy=(rect.get_x() + rect.get_width() / 2, height),
                    xytext=(0, 3),  # 3 points vertical offset
                    textcoords="offset points",
                    ha='center', va='bottom')


plt.xticks(rotation=90)
print(f'Polynomial regression scores of the acute response (naMDI) and delayed␣
↪response (taMDI) against the economic indicators.')
plt.savefig('./poly_chart.png', bbox_inches="tight")
```

```
Polynomial regression scores of the acute response (naMDI) and delayed response
(taMDI) against the economic indicators.
```



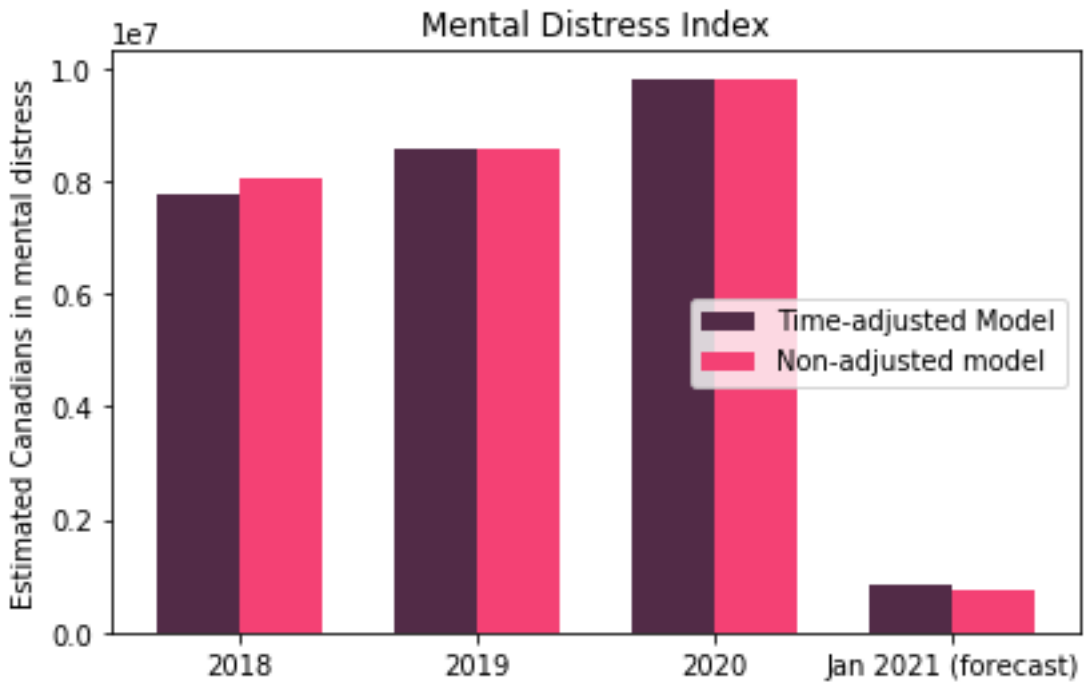
Although there were significant interrelationships shown with linear regression the polynomial regression proved better fit and was the model used to analyze interrelationships as a result.

3 Results

3.1 Modeling

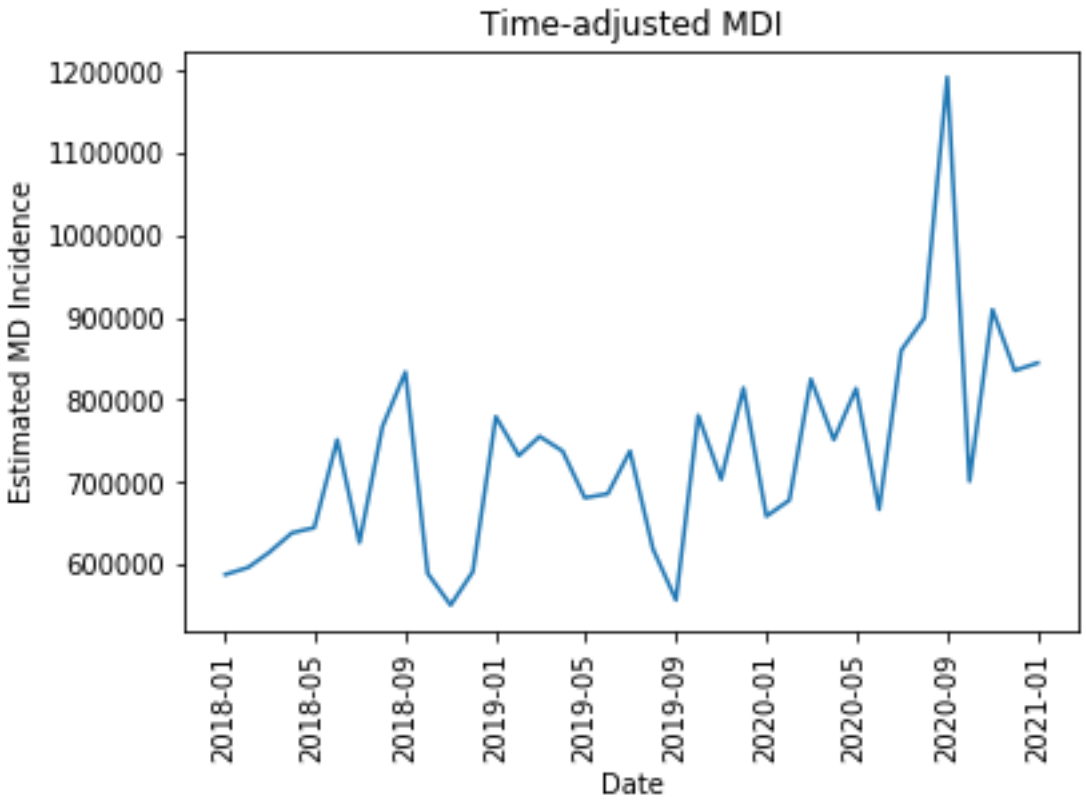
Estimated number of individuals experiencing mental distress: taMDI and naMDI modeling provided a fairly narrow range of predictions:

- Between 7,790,068.3 and 8,066,567.2 Canadians dealt with a mental health episode in 2018.
- Confirms 8,581,600.0 (our baseline) who dealt with a mental health episode in 2019.
- Between 9,789,138.0 and 9,821,683.3 Canadians dealt with a mental health episode in 2020, an increase of between 1,207,537.95 (14.07%) and 1,240,083.35 (14.45%) over 2019 estimates.
- Between 726,300.3 and 844,543.2 Canadians dealing with a mental health episode in January 2021.



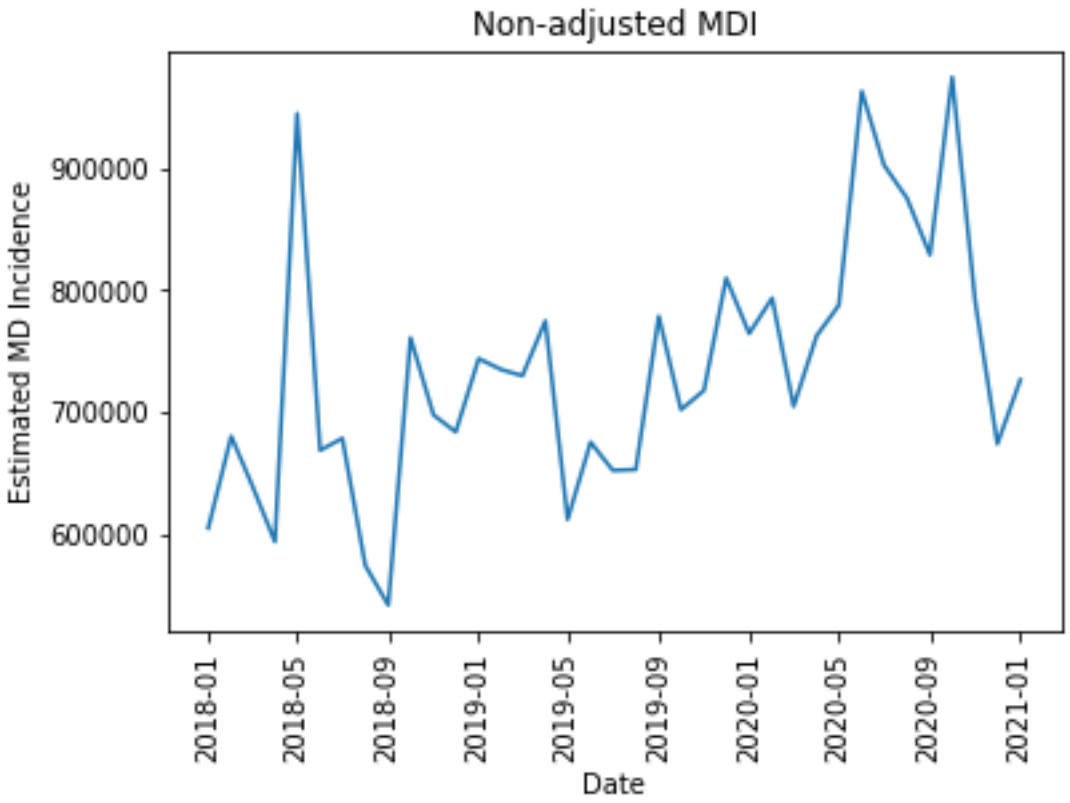
Model	2018	2019	2020	2021
Time-adjusted	7,790,068.27	8,581,600.0	9,789,137.95	844,543.21
Non-adjusted	8,066,567.22	8,581,600.0	9,821,683.35	726,300.28

3.1.1 Time-adjusted Mental Distress Index (aMDI)



Time-adjusted Mental Distress Index: Prediction model integrating the concept of time lag introducd by Lee (2020) which showed increased correlation to death by suicide.

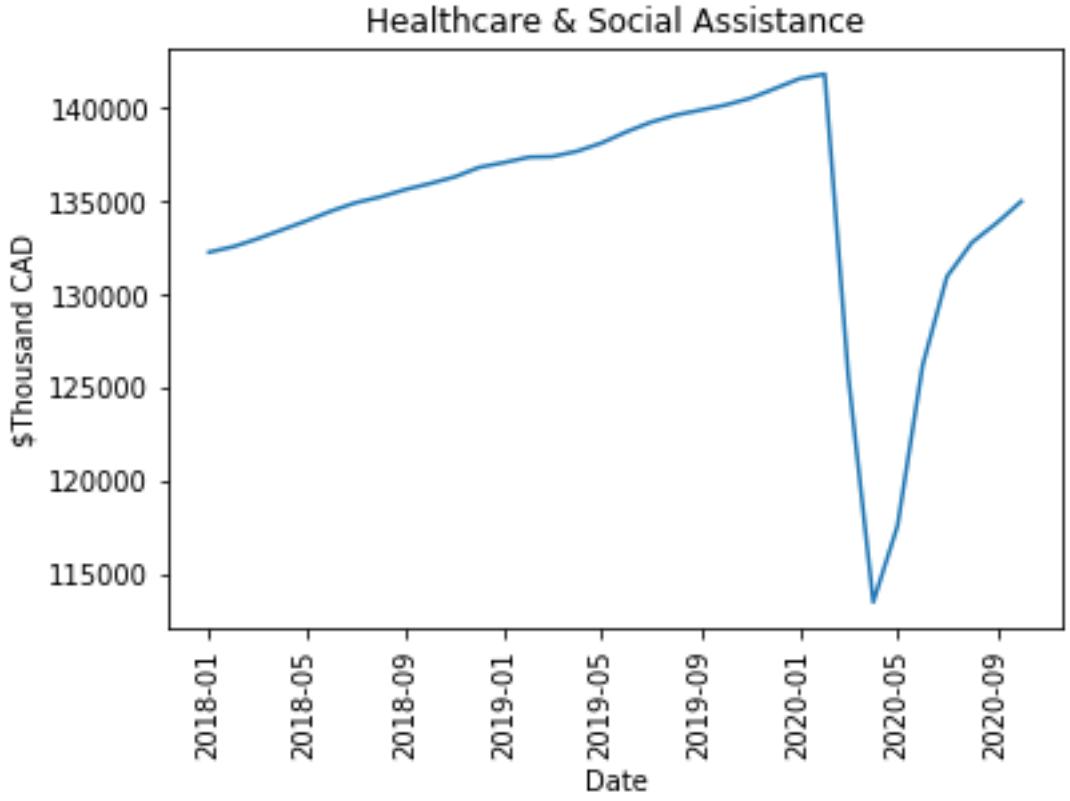
3.1.2 Non-adjusted Mental Distress Index (naMDI)



Non-adjusted Mental Distress Index: More volitile and does not account for time lag although allows for near-real-time data integration, Google Trends datasets included in this study were inclusive of Jan 11th 2021.

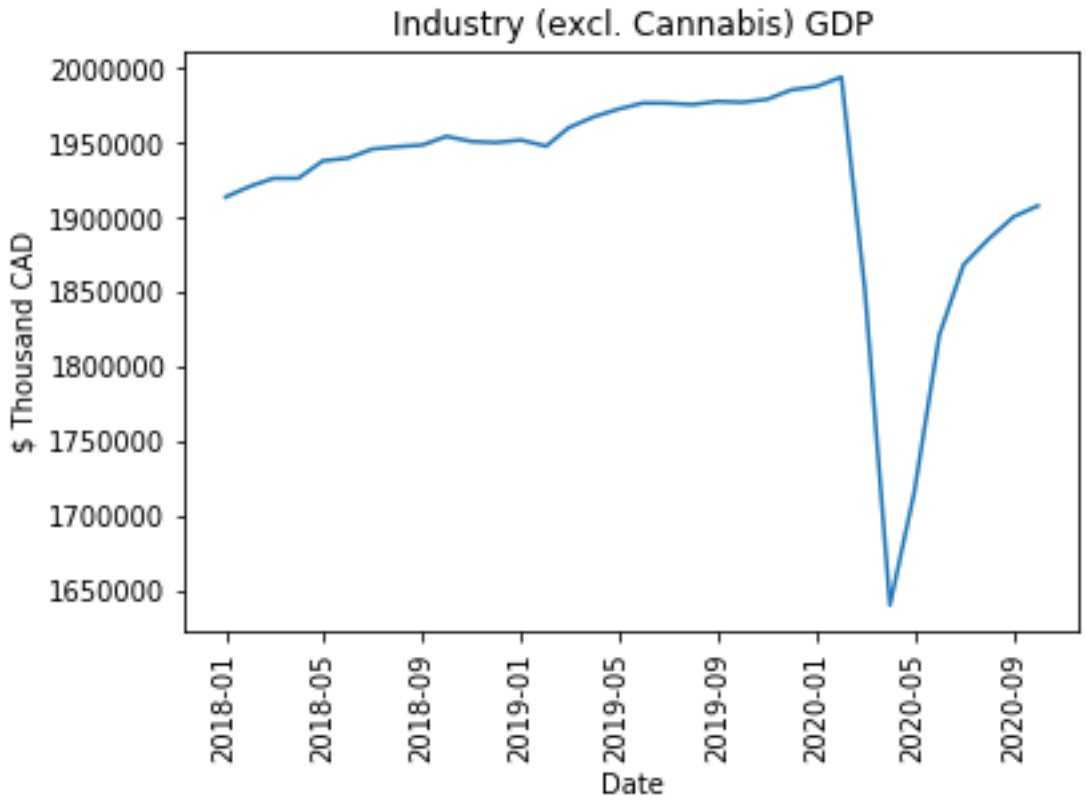
3.2 Economic Indicators

3.2.1 Healthcare and Social Assistance



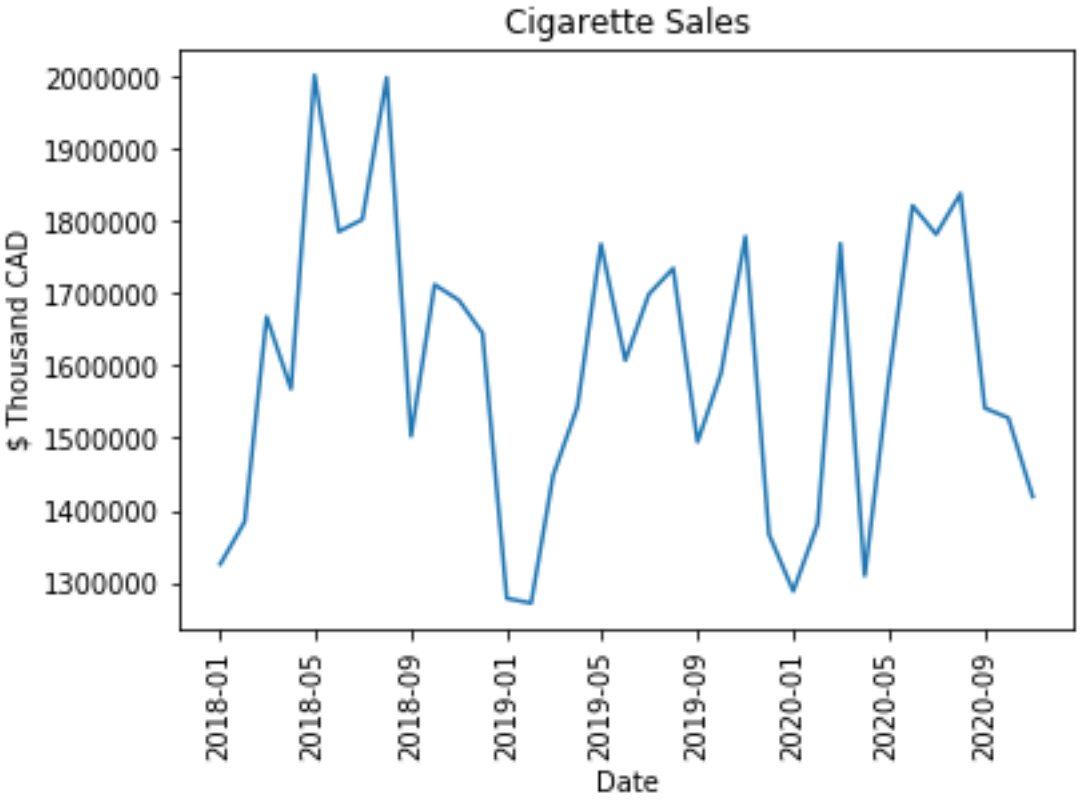
Healthcare and Social Assistance: sector encompasses services involved in diagnosis and treatment, providing residential care for medical and social assistance including counselling, welfare, child protection, community housing and food services, vocational rehabilitation and child care, to those requiring assistance.

3.2.2 Industry (excl. Cannabis)



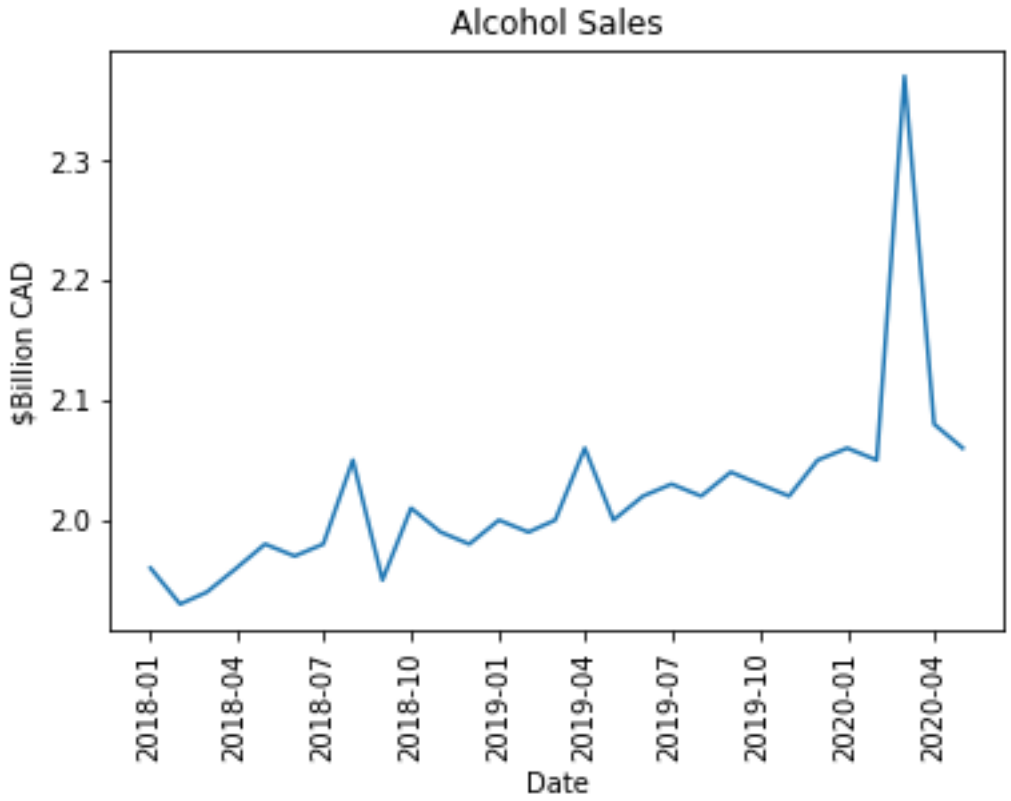
Industry Specific GDP; Industry (excl. Cannabis): The economy was massively affected by spring lockdowns and varying restrictions through the year due to the ongoing pandemic. Employment stressors would be one factor that explain the identified correlation but further study is needed before any conclusions can be made.

3.2.3 Cigarettes and Tobacco



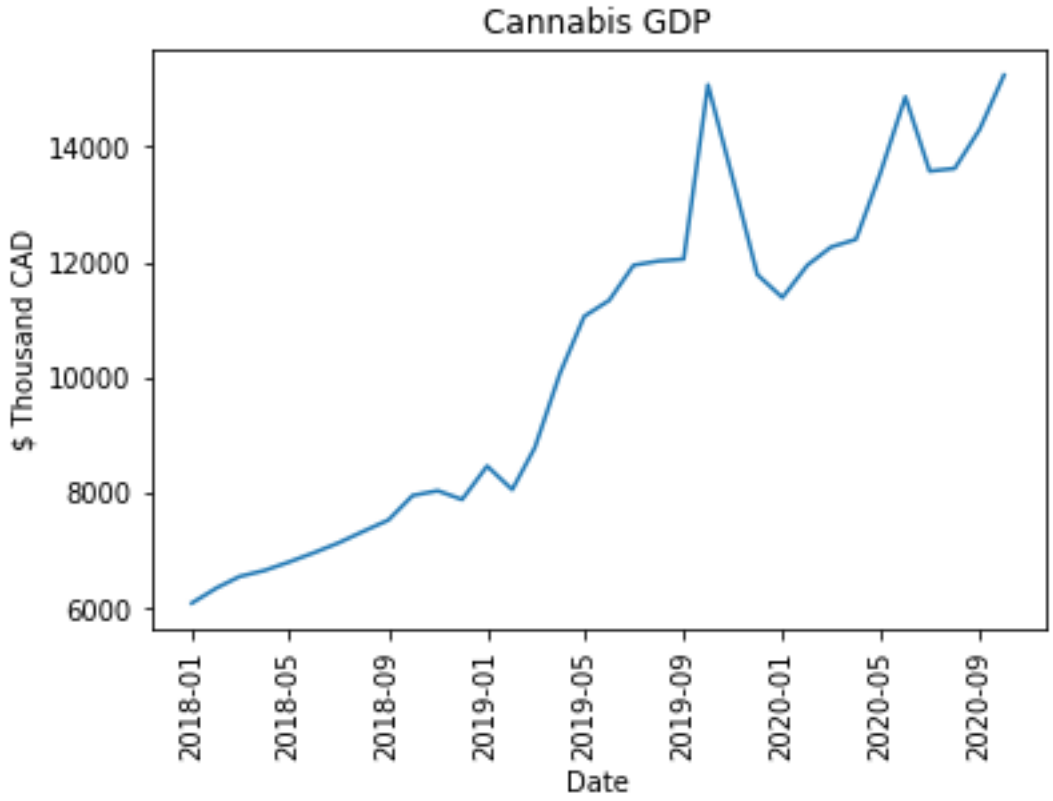
Tobacco: The carcinogenic properties of cigarettes and tobacco are well known. Other health implications include cardiorespiratory seasonal changes in cigarettes seem to indicate that weather may play a factor, with notable dips in the winter months. Highs in May and Sept suggest that long weekens or holidays may significantly contribute to tobacco sales.

3.2.4 Alcohol



Alcohol: The depressive properties of alcohol are of especial interest in the mental health field. Inhibition and impulse control are correlated to increased risk of high risk behaviour including gambling, alcohol, drug use and suicide.

3.2.5 Industry Specific GDP; Cannabis



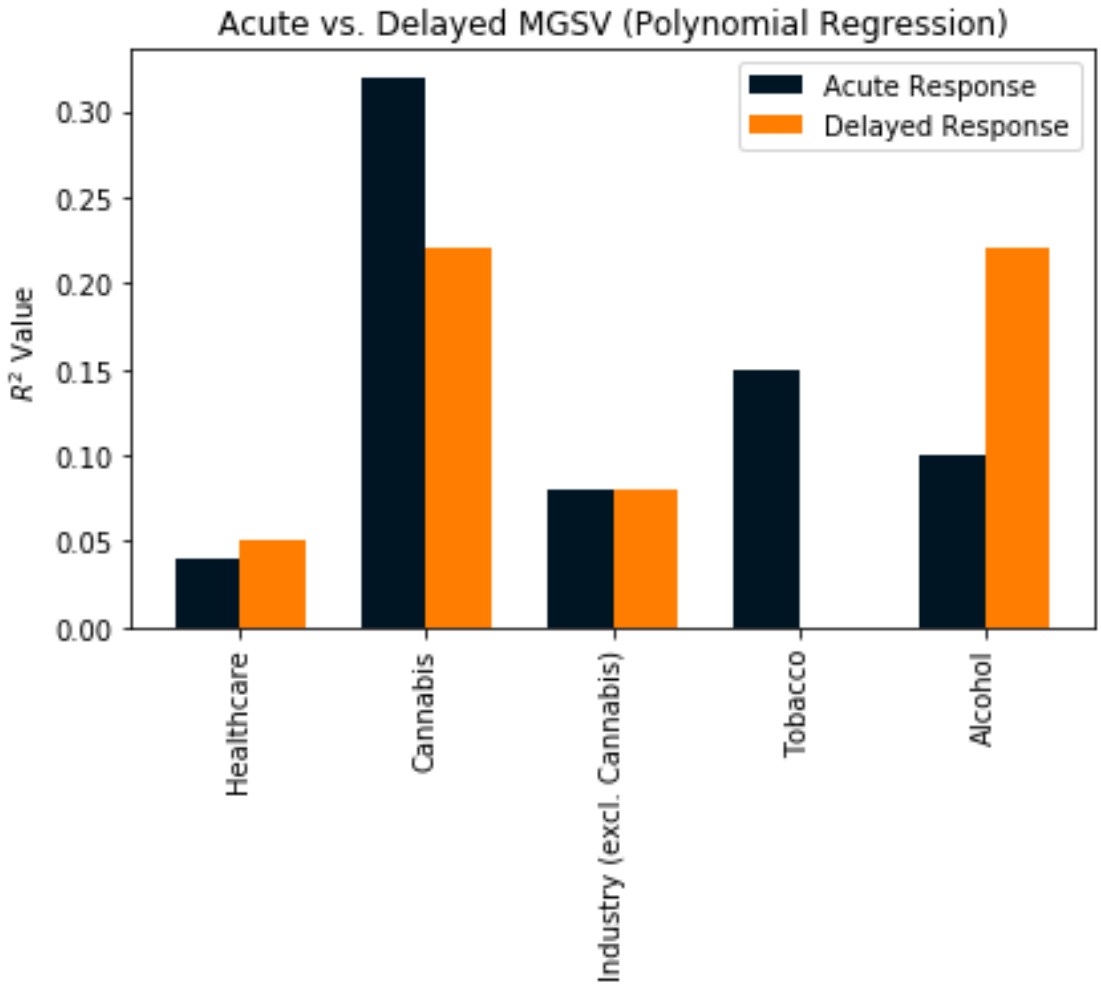
Cannabis: Commonly used to self-medicate for anxiety and depression, research is still in its early phases but the psycho-active properties increase the risk of psychosis and other mental disorders with increased risk to those under 25 years of age.

*It should be noted when looking at Cannabis GDP that the industry was only legalized recreationally in October 2018.

4 Analysis

4.1 Implications of acute and chronic stress

The patterns and differences between acute and delayed (chronic) stress are also interesting to note. The spending patterns can be viewed as "stress profiles", representing a modifier to an individuals likelihood of partaking in a particular activity.



There does appear to be a low to moderate correlation between the proposed mental distress indicies and different economic indicators, although more research is required to explore the possible interrelationships. The statistically significant polynomial R^2 values across multiple attributes suggests that social stress as measured by MGSV does have a positive correlation with spending in industries related to self-medication, with weaker interrelationships with healthcare and general industry.

The two proposed models carry different estimates, with naMD estimating January 2021 -2.34% more stressful than January 2019 while taMDI modeling projects it is +8.37% more stressful based on prior months. To predict a *true* current estimate of stress being experienced by the public, a more detailed model encompasing both the acute and delayed models together should be established.

It is interesting to note different the emerging trends when comparing the acute and delayed inter-relationships:

- The correlation to chronic stress on alcohol ($R^2 = .22$), healthcare ($R^2 = .05$) are greater then that of acute stress ($R^2 = .10$ and $.04$ accordingly).
- Acute stress is strongly related to cannabis ($R^2 = .32$) and tobacco sales ($R^2 = .15$). It should be noted delayed stress showed $R^2 = 0$ with tobacco while cannabis still had a significant relationship months after MGSV data ($R^2 = .22$).
- Industry (excluding Cannabis) functions as a baseline, allowing us to compare the attributes

against an aggregated industrial average sales index. The polynomial regression model weakly correlates both acute and delayed stress as having an equal ($R^2 = .08$) interrelationship with industry (excl. cannabis).

These models suggest Canadians are generally experiencing mildly less acute stress than our 2019 baseline but greater chronic stress from previous months. Extrapolating that information against the plotted polynomial R^2 interrelationships and we may be able to gather some insight into the behavioural trends that may be expected going forward.

4.2 Limitations

It is unknown what percentage of respondents to the *Mental health characteristics and suicidal thoughts* (Stats Canada, 2019) were:

- Seeking mental healthcare assistance but not reporting suicidal ideation
- Reported suicidal ideation but were not seeking mental healthcare assistance
- Both confirmed they were experiencing suicidal ideation and currently under care of mental healthcare professional or social worker

Clarification regarding the *Mental health characteristics and suicidal thoughts (2019)* study may lower the absolute numbers by up to 43%, but the year over year modeling leaves the aggregate coefficients unchanged. It should also be noted that Google Trends is a proprietary product of Alphabet Inc. and as a result it is impossible to see the underlying raw data or methods. Due to the anonymous sampling of Google Trends data there may be natural variation when repeated which should be evaluated further to improve forecasting predictions. Google Trends gave varying MGSV results when search terms were repeated despite consistent parameters. It is assumed that these results are normally distributed but machine learning could be used to create a more accurate model to account for distributions.

5 Conclusion

A strategic multi-faceted plan is urgently needed to prevent collapse of the healthcare system and support a stable economic recovery. In order for this to be done effectively we need up-to-date and relevant socioeconomic data in order to make informed decisions. Identifying models to assess social stress and predict behavioural trends could provide insights and data for economics, healthcare, and public policy. Preliminary research supports the validity of Google Trends as a tool in measuring socioeconomic and healthcare patterns. Further research is needed to clarify and improve these forecasting models as well as exploring the impact demographics may impact the data.

References:

- https://blog.google/products/search/a-new-window-into-our-world-with-real/
- https://support.google.com/trends/answer/4365533?hl=en
- Mental health characteristics and suicidal thoughts (Stats Canada, 2019)
- Gross domestic product by industry, Sept 2020 (Stats Canada, 2020)
- Tobacco, sales and inventories, monthly production (x1000) (Stats Canada, 2020)
- Monthly retail sales of beer, wine and liquor stores in Canada from 2015 to 2020 (in billion Canadian dollars)(Statista, 2020)
- Worldwide desktop market share of leading search engines from January 2010 to October 2020 (Statista, 2020)

- Lee, J., (2020) Search trends preceding increases in suicide: A cross-correlation study of monthly Google search volume and suicide rate using transfer function models
- Parker, et al. (2017). Forecasting state-level premature deaths from alcohol, drugs, and suicides using Google Trends data
- Sapolsky, Robert M. (2004). Why zebras don't get ulcers. New York: Owl Book/Henry Holt and Co.