```
# packages

# standard
import numpy as np
import pandas as pd

# plots
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

# PCA / Clustering
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
```

```
from google.colab import files
import pandas as pd

# Upload the file
uploaded = files.upload()

# Assuming the uploaded file is an Excel file and getting the file name
file_name = list(uploaded.keys())[0]

# Read the Excel file into a DataFrame
df = pd.read_excel(file_name)
```

Choose Files dataset.xlsx

- **dataset.xlsx**(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 10051 bytes, last modified: 8/8/2024 - 100% done

```
Saving dataset.xlsx to dataset.xlsx
```

```
df = pd.read_excel(file_name)
df
```

	Province	Life expectancy at birth	Life expectancy at 65	expenditure on eduction	Median Annual Family Income 2021 (CAD)	GDP	Drug abuse
0	Canada	81.1	20.2	3500.16	98390	201955.071429	4557.857143
1	Newfoundland and Labrador	78.9	18.2	3452.52	89530	30150.300000	852.000000
2	Prince Edward Island	80.2	19.3	2988.13	90220	6542.600000	146.000000
3	Nova Scotia	80.1	19.3	3399.24	87540	40011.500000	829.000000
4	New Brunswick	80.2	19.5	2823.77	85150	32522.700000	1006.000000
5	Quebec	81.2	20.1	3145.54	96910	391189.100000	12063.000000
6	Ontario by Local Health Integration Network	81.5	20.3	3597.92	99550	779145.000000	10980.000000
7	Ontario by Health Unit	81.5	20.3	3597.92	99550	779145.000000	10980.000000
8	Manitoba	79.5	19.6	3576.38	90880	64595.800000	1703.000000
9	Saskatchewan	79.6	19.7	4653.50	96770	81818.700000	1671.000000
10	Alberta	80.7	20.2	4082.48	106960	339308.000000	5225.000000
11	British Columbia	81.7	20.7	3073.23	99610	272008.000000	17566.000000
12	Yukon	76.7	17.0	4245.43	125790	3029.900000	233.000000
13	Northwest Territories	77.4	18.0	7696.38	134610	4396.600000	497.000000

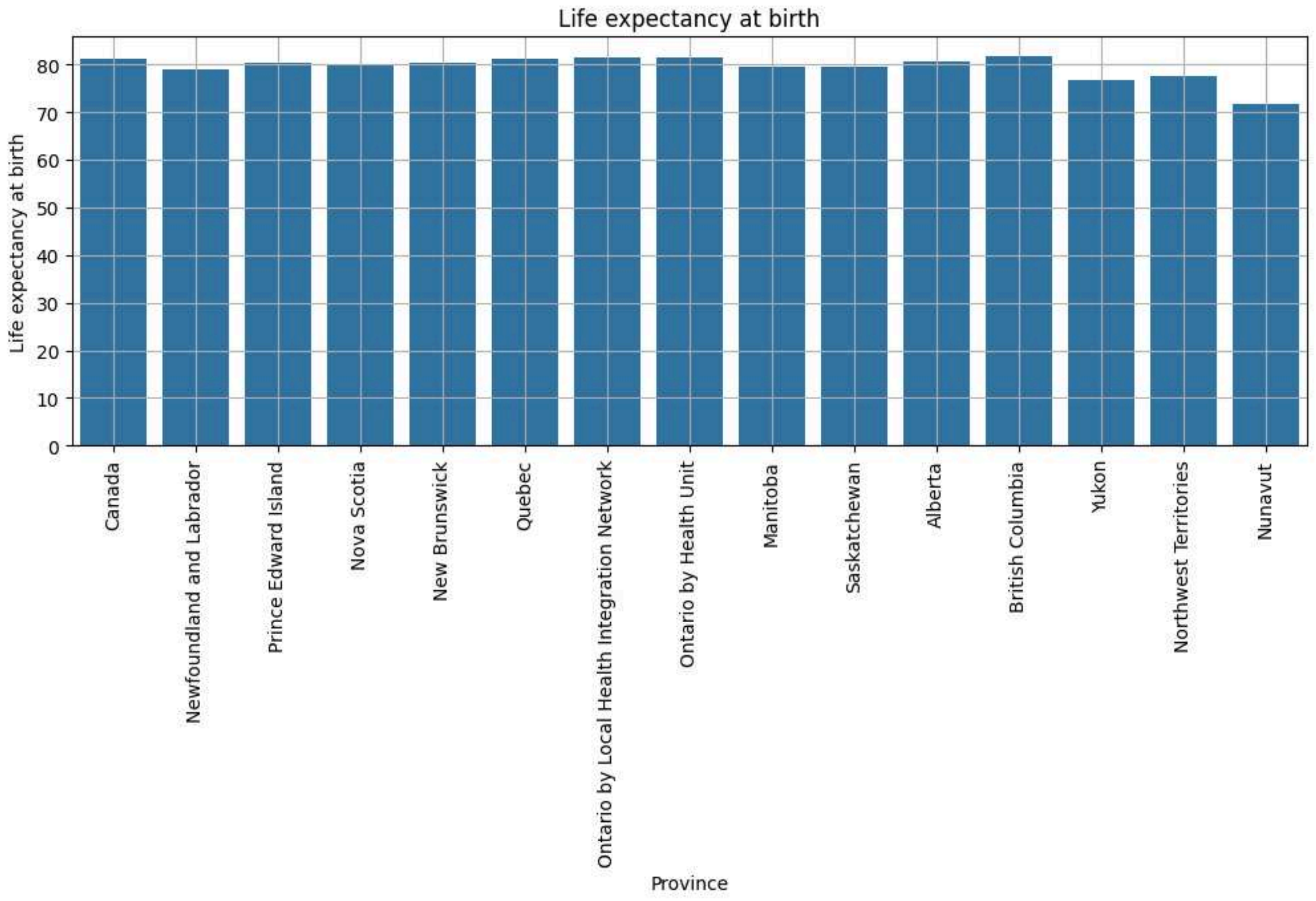
Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# list all features
features_orig = ['Life expectancy at birth', 'Life expectancy at 65', 'expenditure on eduction', 'Median Annual Family Income 2021 (CAD)','
                 ]
```

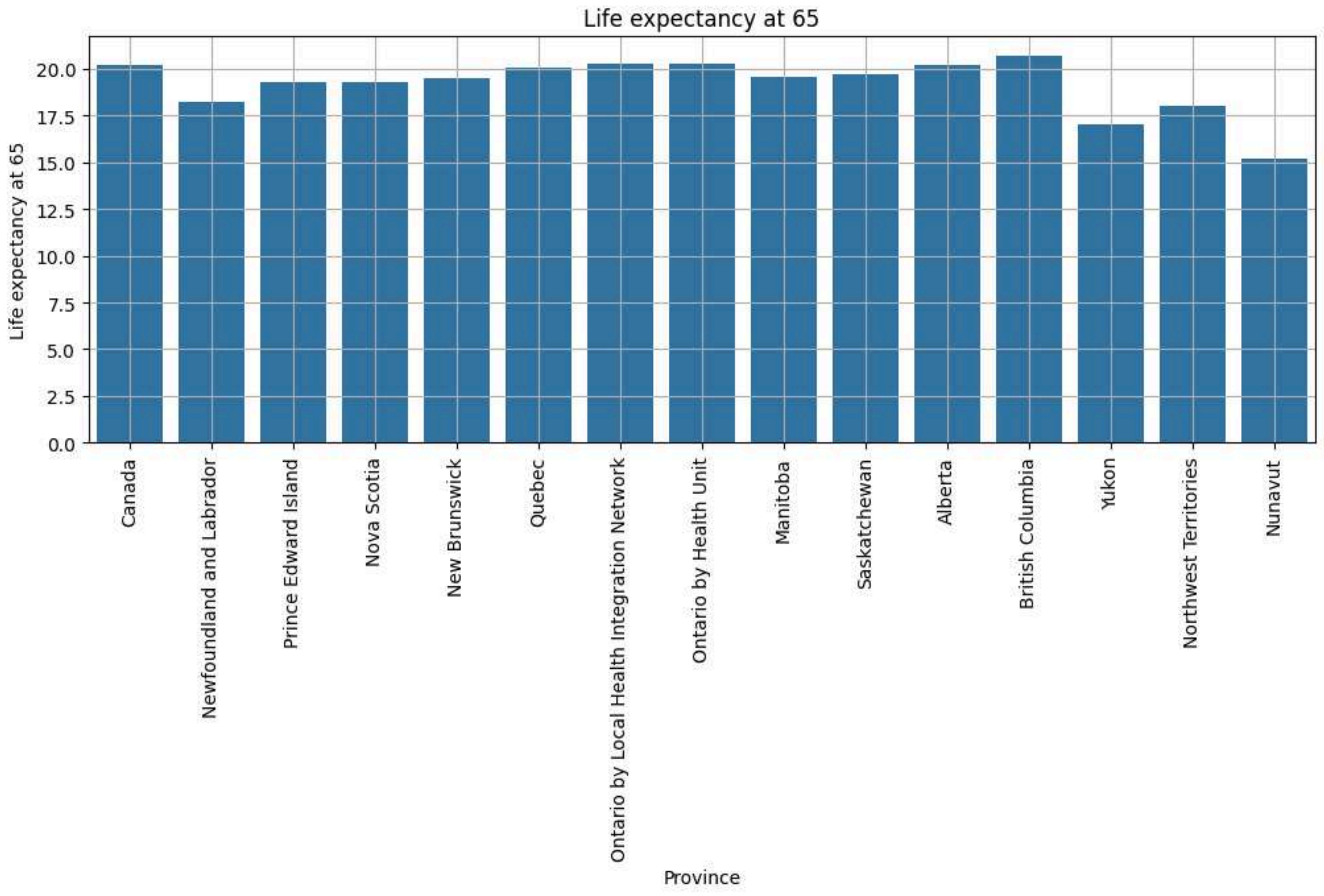
```
# summary statistics
df[features_orig].describe()
```

	Life expectancy at birth	Life expectancy at 65	expenditure on eduction	Median Annual Family Income 2021 (CAD)	GDP	Drug abuse
count	15.000000	15.000000	15.000000	15.00000	15.000000	15.000000
mean	79.460000	19.173333	3958.292667	98930.00000	201955.071429	4557.857143
std	2.619651	1.494498	1251.228825	14363.12839	267234.161247	5606.810984
min	71.600000	15.200000	2823.770000	82490.00000	3029.900000	59.000000
25%	79.200000	18.750000	3272.390000	89875.00000	18346.450000	663.000000
50%	80.200000	19.600000	3576.380000	96910.00000	64595.800000	1671.000000
75%	81.150000	20.200000	4163.955000	99580.00000	305658.000000	8102.500000
max	81.700000	20.700000	7696.380000	134610.00000	779145.000000	17566.000000

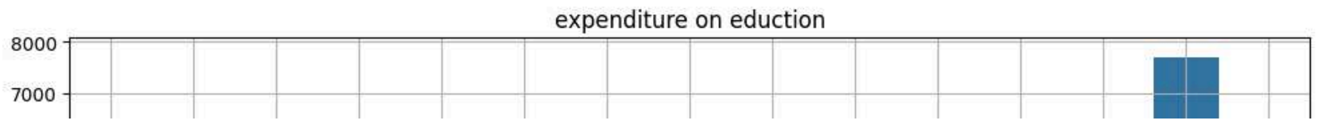
```python
# barplot of all the features
for f in features_orig:
    fig = plt.figure(figsize = (12,4))
    sns.barplot(x='Province', y=f, data=df)
    plt.xticks(rotation=90)
    plt.grid()
    plt.title(f)
    plt.show()
```



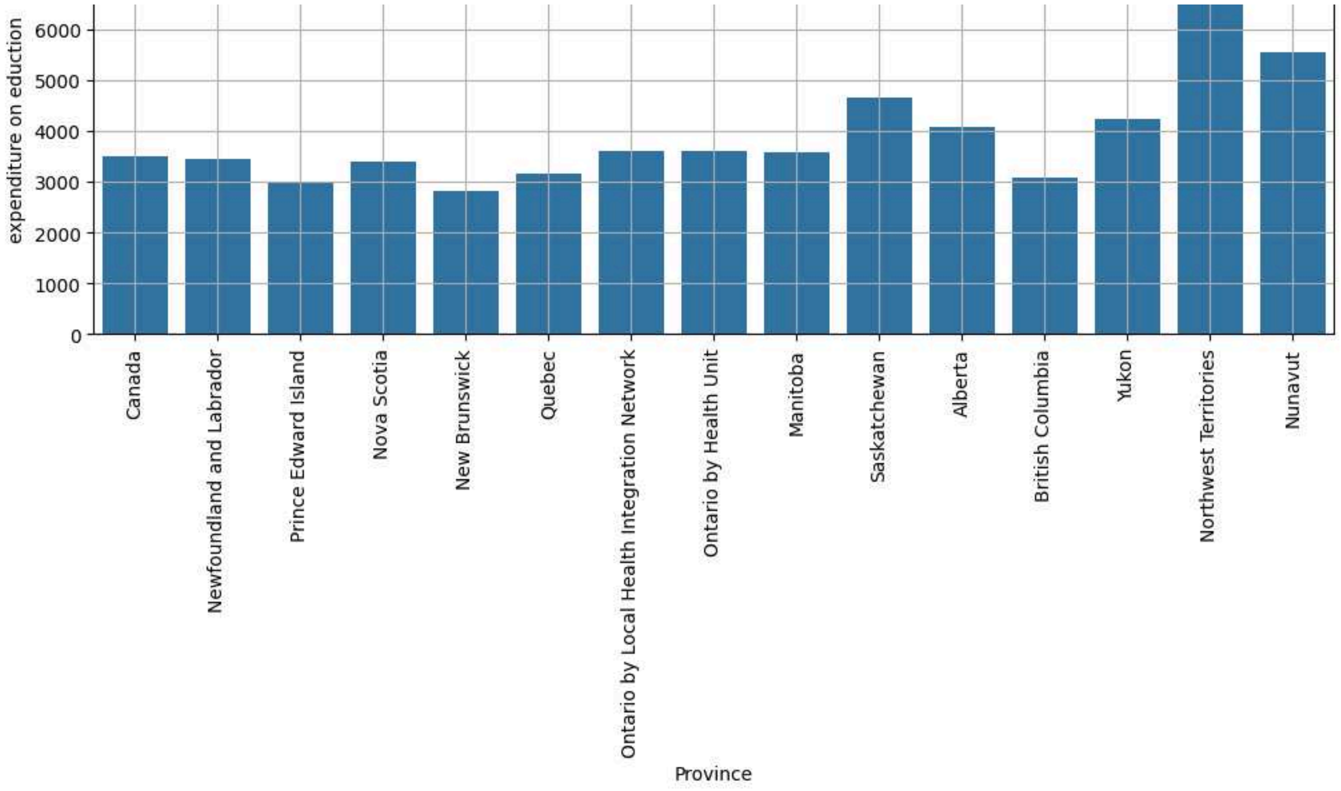
Life expectancy at birth
Life expectancy at birth
0
10
20
30
40
50
60
70
80
Canada
Newfoundland and Labrador
Prince Edward Island
Nova Scotia
New Brunswick
Quebec
Ontario by Local Health Integration Network
Ontario by Health Unit
Manitoba
Saskatchewan
Alberta
British Columbia
Yukon
Northwest Territories
Nunavut
Province

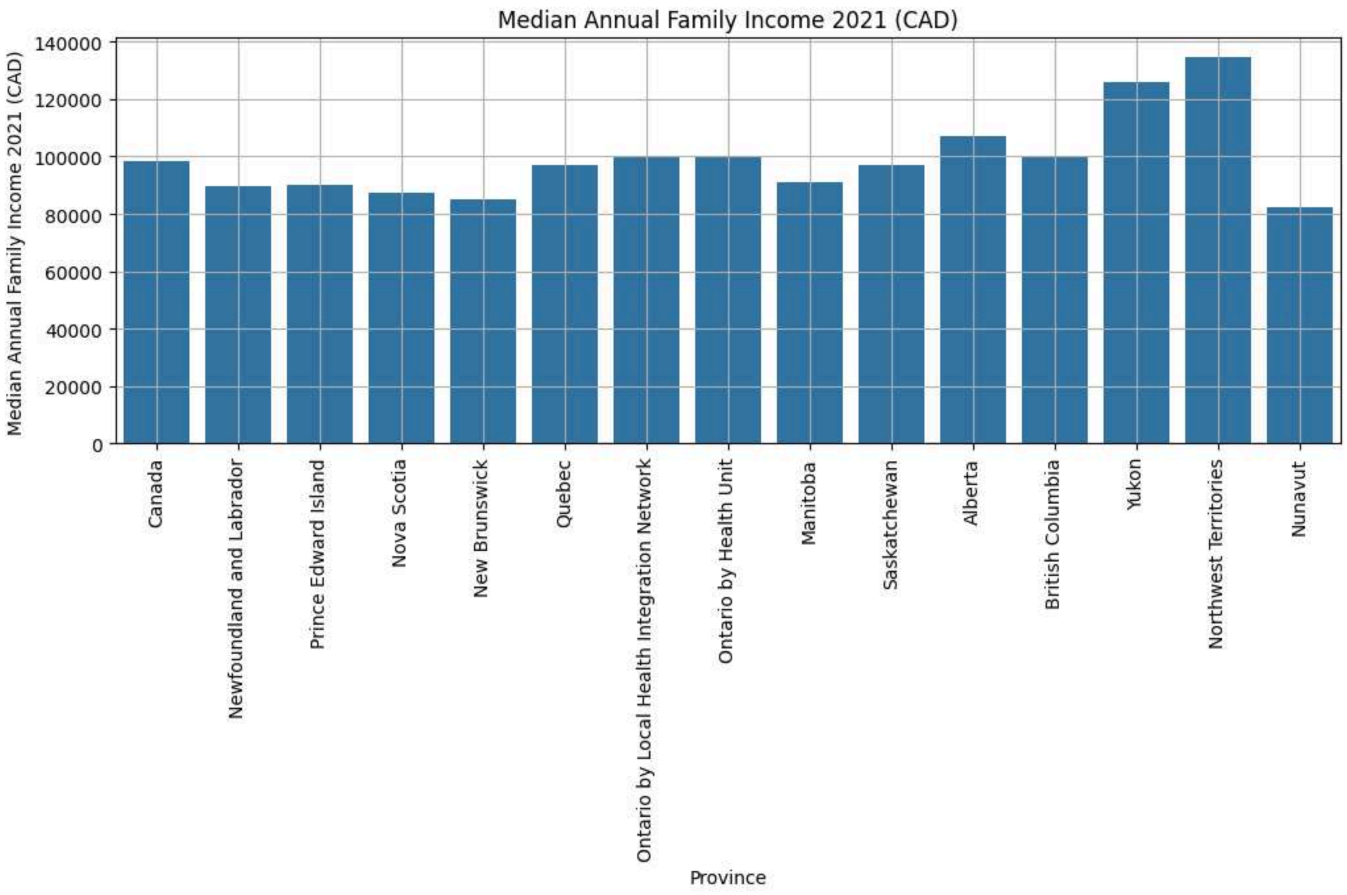


Life expectancy at 65
Life expectancy at 65
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Canada
Newfoundland and Labrador
Prince Edward Island
Nova Scotia
New Brunswick
Quebec
Ontario by Local Health Integration Network
Ontario by Health Unit
Manitoba
Saskatchewan
Alberta
British Columbia
Yukon
Northwest Territories
Nunavut
Province

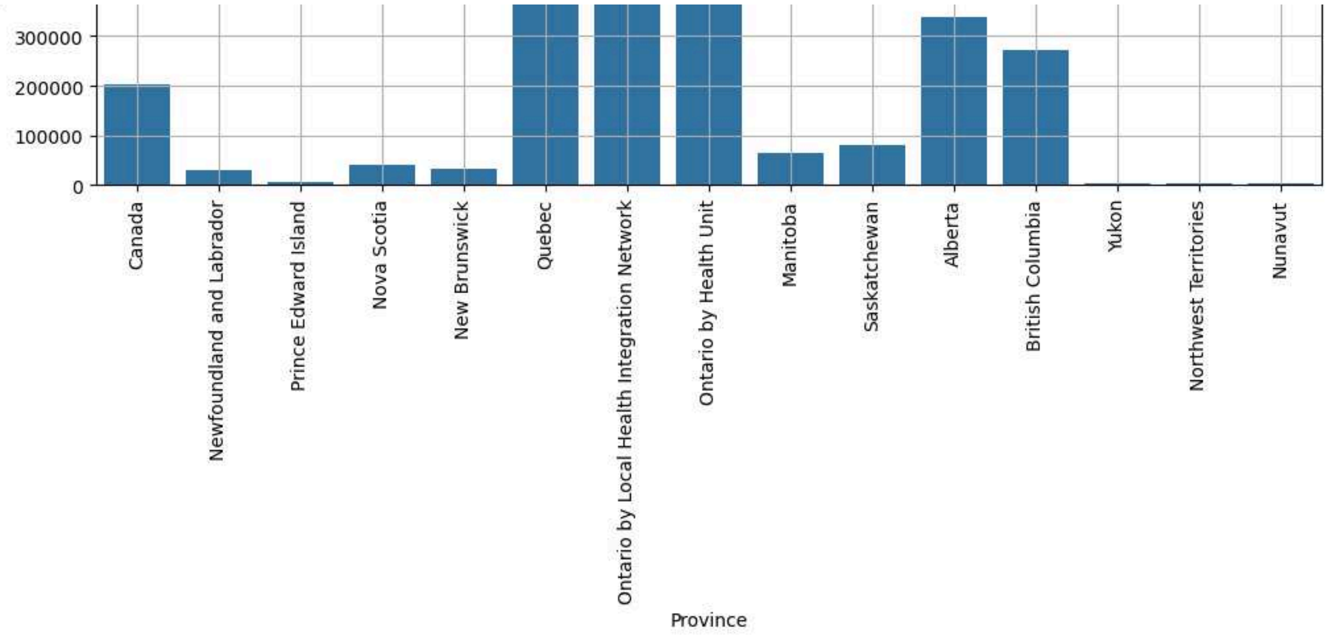


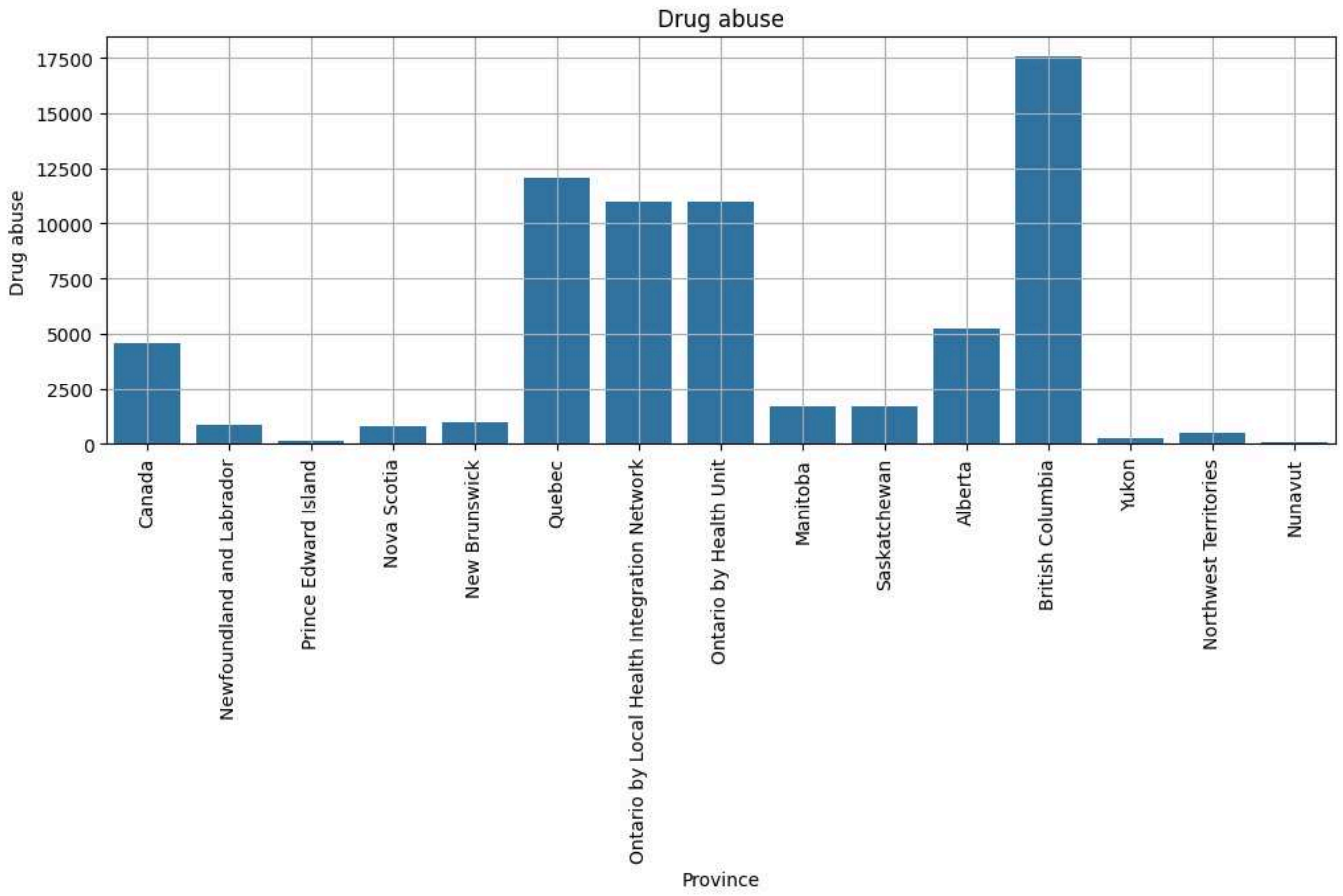
expenditure on eduction
8000
7000


expenditure on eduction
6000
5000
4000
3000
2000
1000
0
Canada
Newfoundland and Labrador
Prince Edward Island
Nova Scotia
New Brunswick
Quebec
Ontario by Local Health Integration Network
Ontario by Health Unit
Manitoba
Saskatchewan
Alberta
British Columbia
Yukon
Northwest Territories
Nunavut
Province


Median Annual Family Income 2021 (CAD)
Median Annual Family Income 2021 (CAD)
140000
120000
100000
80000
60000
40000
20000
0
Canada
Newfoundland and Labrador
Prince Edward Island
Nova Scotia
New Brunswick
Quebec
Ontario by Local Health Integration Network
Ontario by Health Unit
Manitoba
Saskatchewan
Alberta
British Columbia
Yukon
Northwest Territories
Nunavut
Province

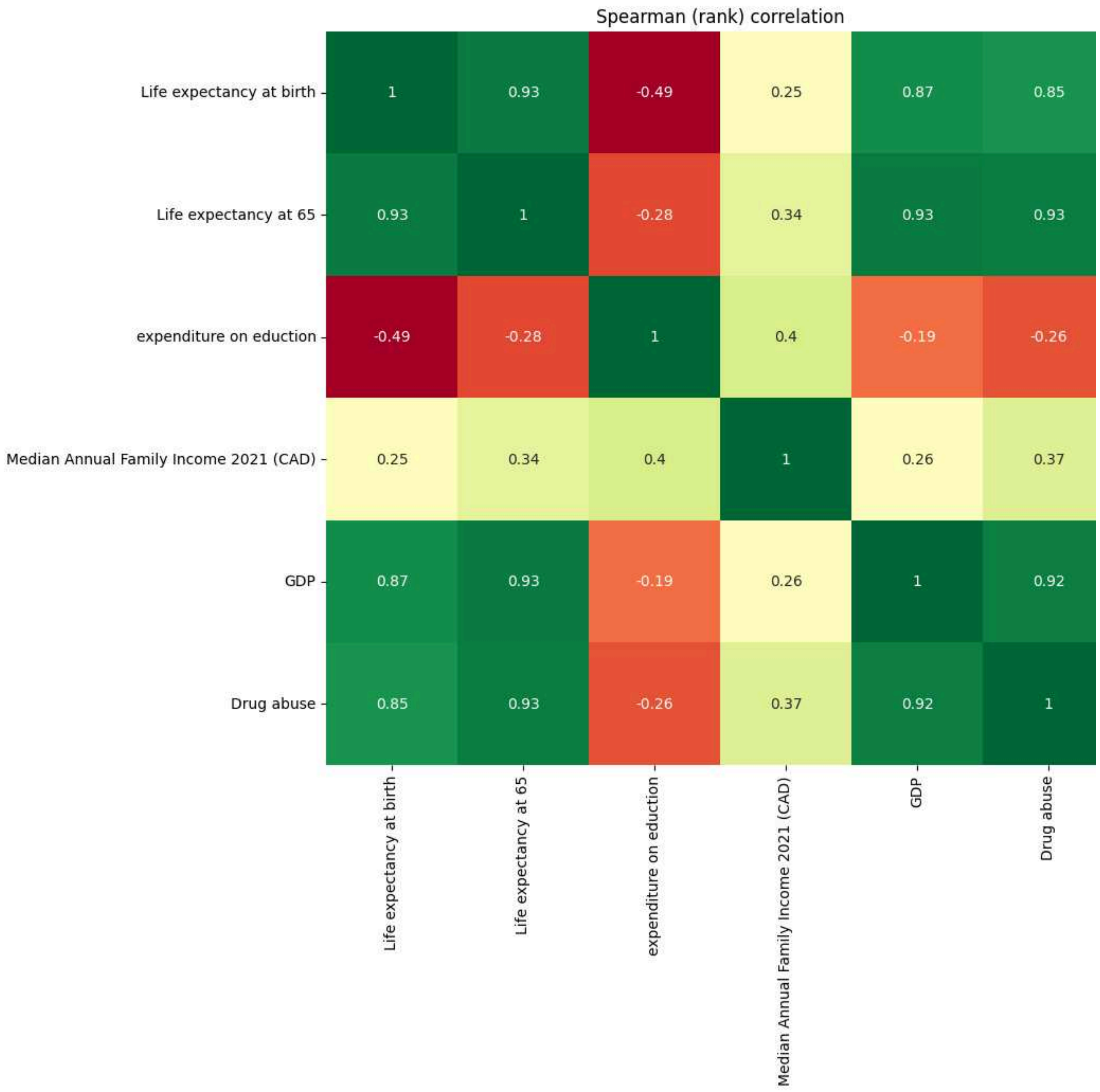
GDP
GDP
800000
700000
600000
500000
400000





CORELATION

```
# calc RANK correlation of features
corr_mat = df[features_orig].corr(method='spearman')
# plot (rank) correlation matrix
fig = plt.figure(figsize = (12,9))
sns.heatmap(corr_mat, annot=True, cmap="RdYlGn")
plt.title('Spearman (rank) correlation')
plt.show()
```



	Life expectancy at birth	Life expectancy at 65	expenditure on eduction	Median Annual Family Income 2021 (CAD)	GDP	Drug abuse
Life expectancy at birth	1	0.93	-0.49	0.25	0.87	0.85
Life expectancy at 65	0.93	1	-0.28	0.34	0.93	0.93
expenditure on eduction	-0.49	-0.28	1	0.4	-0.19	-0.26
Median Annual Family Income 2021 (CAD)	0.25	0.34	0.4	1	0.26	0.37
GDP	0.87	0.93	-0.19	0.26	1	0.92
Drug abuse	0.85	0.93	-0.26	0.37	0.92	1

Double-click (or enter) to edit

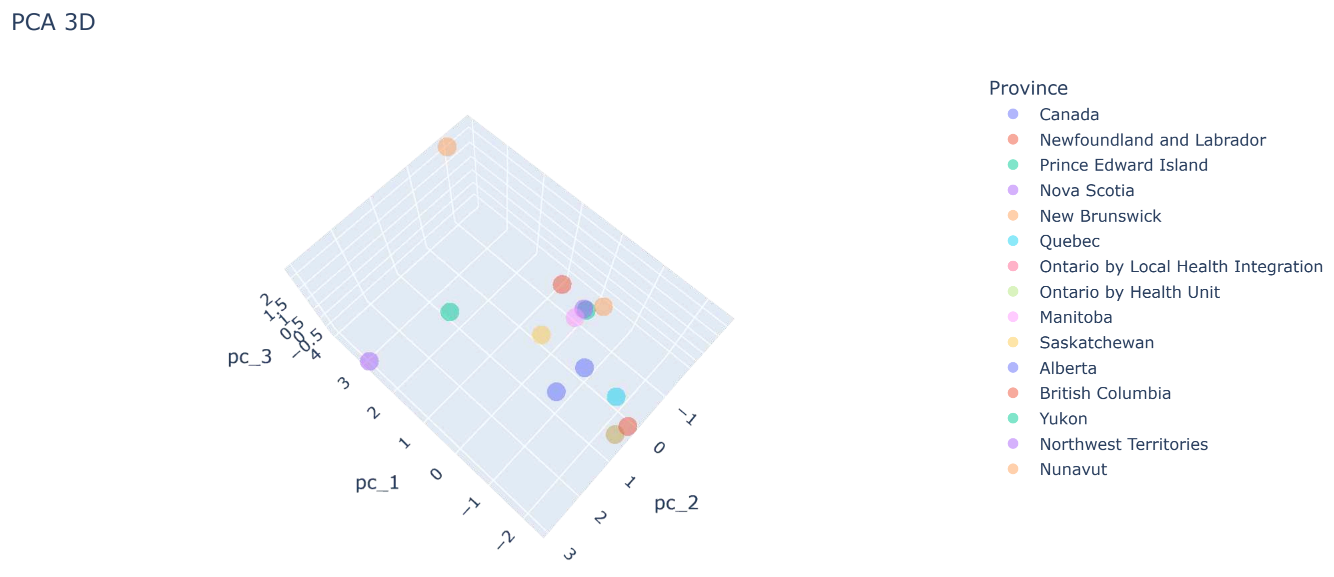
PRINCIPAL COMPONENT ANALYSIS

```
# select features
features4pca = features_orig.copy()

print('Using the following features:')
print(features4pca)
```

```
Using the following features:
['Life expectancy at birth', 'Life expectancy at 65', 'expenditure on eduction', 'Median Annual Family Income 2021 (CAD)', 'GDP', 'Drug
```

```
# use only selected features for PCA
df4pca = df[features4pca]
# standardize first
df4pca_std = StandardScaler().fit_transform(df4pca)
# define 3D PCA
pc_model = PCA(n_components=3)
# calc PCA
pc = pc_model.fit_transform(df4pca_std)
# add PCA results to original data frame
df['pc_1'] = pc[:,0]
df['pc_2'] = pc[:,1]
df['pc_3'] = pc[:,2]
```

```
import plotly.express as px

# Interactive plot of top 3 principal components
fig = px.scatter_3d(df, x='pc_1', y='pc_2', z='pc_3',
                    color='Province',
                    hover_data=['Province'],
                    opacity=0.5)
fig.update_layout(title='PCA 3D')
fig.show()
```



K - MEANS CLUSTERING

```
# define cluster algorithm and parameters
n_cl = 4 # number of clusters
kmeans = KMeans(init='random', n_clusters=n_cl, n_init=10, max_iter=300, random_state=99)

# and run it on scaled data (we will simply re-use the data from the PCA excercise)
kmeans.fit(df4pca_std);
```

```
# append cluster variable to data frame
df['cluster'] = kmeans.labels_.astype('object')
```

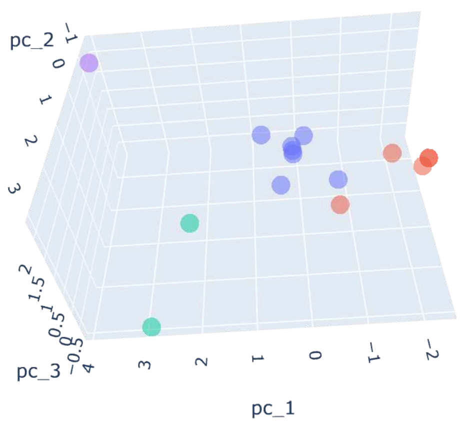
```
# show provinces of each cluster
for c in range(4):
    print('Cluster ' + str(c) + ':')
    print(df[df.cluster==c].Province.value_counts().index.tolist())
    print()
```

```
Cluster 0:
['Quebec', 'Ontario by Local Health Integration Network', 'Ontario by Health Unit', 'Alberta', 'British Columbia']

Cluster 1:
```

```
['Nunavut']

Cluster 2:
['Yukon', 'Northwest Territories']

Cluster 3:
['Canada', 'Newfoundland and Labrador', 'Prince Edward Island', 'Nova Scotia', 'New Brunswick', 'Manitoba', 'Saskatchewan']
```

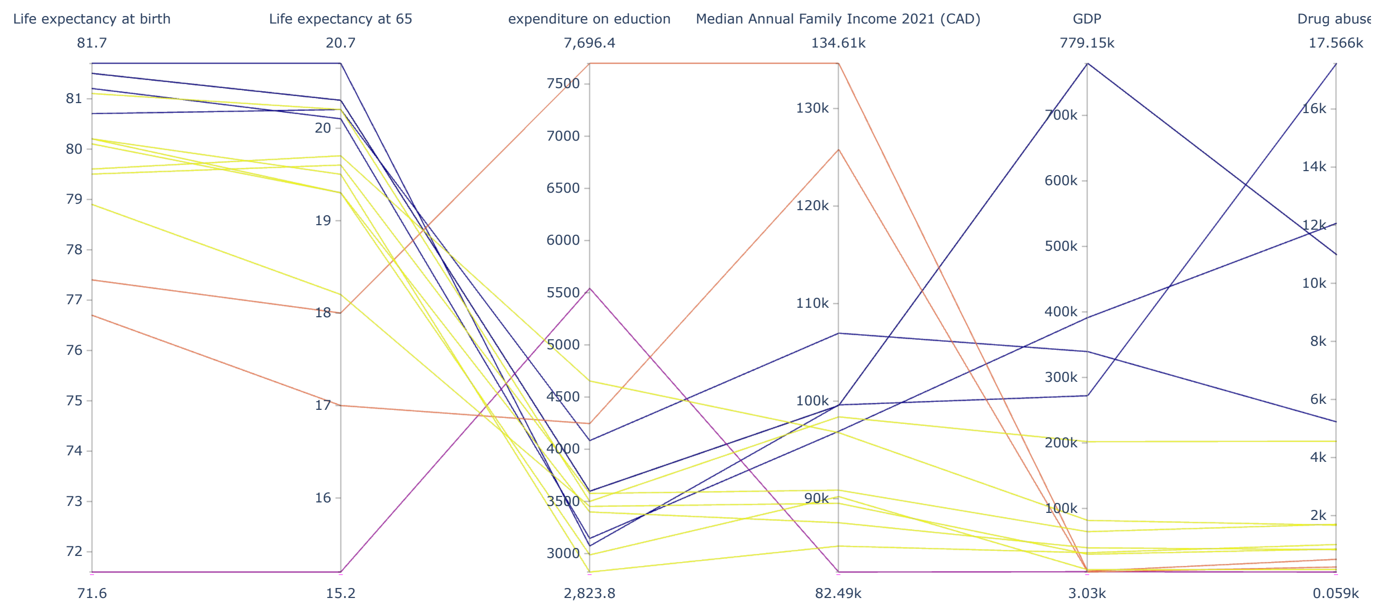
```
# visualize clusters using PCA components
fig = px.scatter_3d(df, x='pc_1', y='pc_2', z='pc_3',
                    color='cluster',
                    hover_data=['Province'],
                    opacity=0.5)
fig.update_layout(title='Clusters')
fig.show()
```

Clusters



PARALLEL PLOT FOR CLUSTERS

```
# parallel plot showing the original features by cluster
fig = px.parallel_coordinates(df[features4pca+['cluster']], color='cluster')
fig.show()
```


Life expectancy at birth
81.7
71.6
Life expectancy at 65
20.7
15.2
expenditure on eduction
7,696.4
2,823.8
Median Annual Family Income 2021 (CAD)
134.61k
82.49k
GDP
779.15k
3.03k
Drug abuse
17.566k
0.059k