

```
import pandas as pd
import numpy as np
from sklearn import linear_model
import matplotlib.pyplot as plt

df=pd.read_csv('https://raw.githubusercontent.com/Rhevathi/DATA-SCIENCE/master/ML/1_linear_reg/Exercise/canada_per_capita_income.csv')
```

df

15	1985	11018.955850
16	1986	11482.891530
17	1987	12974.806620
18	1988	15080.283450
19	1989	16426.725480
20	1990	16838.673200
21	1991	17266.097690
22	1992	16412.083090
23	1993	15875.586730
24	1994	15755.820270
25	1995	16369.317250
26	1996	16699.826680
27	1997	17310.757750
28	1998	16622.671870
29	1999	17581.024140
30	2000	18987.382410
31	2001	18601.397240
32	2002	19232.175560
33	2003	22739.426280
34	2004	25719.147150
35	2005	29198.055690
36	2006	32738.262900
37	2007	36144.481220
38	2008	37446.486090
39	2009	32755.176820
40	2010	38420.522890
41	2011	42334.711210
42	2012	42665.255970
43	2013	42676.468370
44	2014	41039.893600
45	2015	35175.188980
46	2016	34229.193630

```
df.info()
```

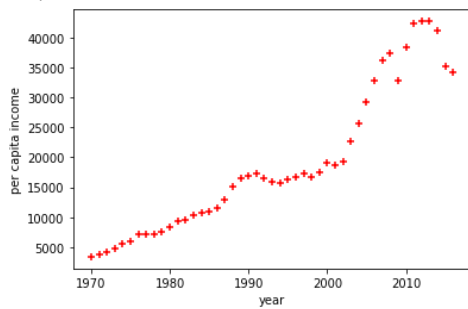
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 47 entries, 0 to 46
Data columns (total 2 columns):
#   Column                Non-Null Count  Dtype
---  --
0   year                   47 non-null    int64
1   per capita income (US$) 47 non-null    float64
dtypes: float64(1), int64(1)
memory usage: 880.0 bytes
```

```
df.isnull().sum()
```

```
year                0
per capita income (US$) 0
dtype: int64
```

```
%matplotlib inline
plt.xlabel('year')
plt.ylabel('per capita income')
plt.scatter(df['year'],df['per capita income (US$)'],color='red',marker='+')
```


```
<matplotlib.collections.PathCollection at 0x7f302c43e280>
```



```
x=df.drop('per capita income (US$)',axis=1)
```

```
x
```


year



0	1970
1	1971
2	1972
3	1973
4	1974
5	1975
6	1976
7	1977
8	1978
9	1979
10	1980
11	1981
12	1982
13	1983
14	1984
15	1985
16	1986
17	1987
18	1988

```
y=df['per capita income (US$)']
```

y



0	3399.299037
1	3768.297935
2	4251.175484
3	4804.463248
4	5576.514583
5	5998.144346
6	7062.131392
7	7100.126170
8	7247.967035
9	7602.912681
10	8355.968120
11	9434.390652
12	9619.438377
13	10416.536590
14	10790.328720
15	11018.955850
16	11482.891530
17	12974.806620
18	15080.283450
19	16426.725480
20	16838.673200
21	17266.097690
22	16412.083090
23	15875.586730
24	15755.820270
25	16369.317250
26	16699.826680
27	17310.757750
28	16622.671870
29	17581.024140
30	18987.382410
31	18601.397240
32	19232.175560
33	22739.426280
34	25719.147150
35	29198.055690
36	32738.262900
37	36144.481220
38	37446.486090
39	32755.176820
40	38420.522890
41	42334.711210
42	42665.255970
43	42676.468370
44	41039.893600
45	35175.188980
46	34229.193630

```
Name: per capita income (US$), dtype: float64
```

```
singleregmodel=linear_model.LinearRegression()  
singleregmodel.fit(x,y)
```

LinearRegression

```
singleregmodel.predict([[2020]])
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but LinearRegression was fitted with fe
warnings.warn(
array([41288.69409442])
```

```
pip install pickle
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
ERROR: Could not find a version that satisfies the requirement pickle (from versions: none)
ERROR: No matching distribution found for pickle
```

```
import pickle
```

```
with open ('reg_pickle','wb') as f:
    pickle.dump(singleregmodel,f)
```

```
with open ('reg_pickle','rb') as f:
    sreg=pickle.load(f)
```

```
sreg.predict([[2020]])
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but LinearRegression was fitted with fe
warnings.warn(
array([41288.69409442])
```

```
import joblib
```

```
joblib.dump(singleregmodel,'reg_joblib')
```

```
['reg_joblib']
```

```
sj=joblib.load('reg_joblib')
```

```
sj.predict([[2020]])
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but LinearRegression was fitted with fe
warnings.warn(
array([41288.69409442])
```

[Colab paid products](#) - [Cancel contracts here](#)

✓ 0s completed at 3:49PM

