In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
import re
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
```

In [2]:

```
a=pd.read_csv(r"C:\Users\user\Downloads\spi_index_labelled.csv")
a
```

Out[2]:

	country	iso3c	date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.INDE
0	NaN	NaN	NaN	Pillar 1 - Data Use - Score	Pillar 2 - Data Services - Score	Pillar 3 - Data Products - Score	Pillar 4 Sources
1	Norway	NOR	2019.0	100	92.2333333333333	77.56875	80.6666666
2	Italy	ITA	2019.0	100	91.8666666666666	75.2875	
3	Austria	AUT	2019.0	100	91.3	74.55	
4	Poland	POL	2019.0	100	95.1	70.5375	79.7166666
...	...	...	...	...	...	...	
3484	Virgin Islands (U.S.)	VIR	2004.0	20	NaN	NaN	
3485	West Bank and Gaza	PSE	2004.0	20	NaN	NaN	
3486	Yemen, Rep.	YEM	2004.0	20	NaN	NaN	
3487	Zambia	ZMB	2004.0	40	NaN	NaN	
3488	Zimbabwe	ZWE	2004.0	20	NaN	NaN	

3489 rows × 79 columns

In [3]:

```
a.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3489 entries, 0 to 3488
Data columns (total 79 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   country                         3488 non-null   object
 1   iso3c                           3488 non-null   object
 2   date                            3488 non-null   float64
 3   SPI.INDEX.PIL1                  3489 non-null   object
 4   SPI.INDEX.PIL2                  693 non-null    object
 5   SPI.INDEX.PIL3                  3226 non-null   object
 6   SPI.INDEX.PIL4                  690 non-null    object
 7   SPI.INDEX.PIL5                  756 non-null    object
 8   SPI.INDEX                       690 non-null    object
 9   SPI.DIM1.5.INDEX                3489 non-null   object
 10  SPI.DIM2.1.INDEX                1283 non-null   object
 11  SPI.DIM2.2.INDEX                840 non-null    object
 12  SPI.DIM2.4.INDEX                762 non-null    object
 13  SPI.DIM3.1.INDEX                3441 non-null   object
 14  SPI.DIM3.2.INDEX                3441 non-null   object
 15  SPI.DIM3.3.INDEX                3226 non-null   object
 16  SPI.DIM3.4.INDEX                3441 non-null   object
 17  SPI.DIM4.1.CEN.INDEX            755 non-null    object
 18  SPI.DIM4.1.SVY.INDEX            3041 non-null   object
 19  SPI.DIM4.2.INDEX                1076 non-null   object
 20  SPI.DIM4.3.INDEX                840 non-null    object
 21  SPI.DIM5.1.INDEX                3489 non-null   object
 22  SPI.DIM5.2.INDEX                756 non-null    object
 23  SPI.DIM5.5.INDEX                3489 non-null   object
 24  SPI.D1.5.POV                    3489 non-null   object
 25  SPI.D1.5.CHLD.MORT              3489 non-null   object
 26  SPI.D1.5.DT.TDS.DPPF.XP.ZS      3489 non-null   object
 27  SPI.D1.5.SAFE.MAN.WATER         3489 non-null   object
 28  SPI.D1.5.LFP                    3489 non-null   object
 29  SPI.D2.1.GDDS                   1283 non-null   object
 30  SPI.D2.2.Machine.readable       840 non-null    object
 31  SPI.D2.2.Non.proprietary        840 non-null    object
 32  SPI.D2.2.Download.options       840 non-null    object
 33  SPI.D2.2.Metadata.available     840 non-null    object
 34  SPI.D2.2.Terms.of.use           840 non-null    object
 35  SPI.D2.2.Openness.subscore      840 non-null    object
 36  SPI.D2.4.NADA                   762 non-null    object
 37  SPI.D3.1.POV                    3441 non-null   object
 38  SPI.D3.2.HNGR                   3441 non-null   object
 39  SPI.D3.3.HLTH                   3441 non-null   object
 40  SPI.D3.4.EDUC                   3441 non-null   object
 41  SPI.D3.5.GEND                   3441 non-null   object
 42  SPI.D3.6.WTRS                   3441 non-null   object
 43  SPI.D3.7.ENRG                   3441 non-null   object
 44  SPI.D3.8.WORK                   3441 non-null   object
 45  SPI.D3.9.INDY                   3441 non-null   object
 46  SPI.D3.10.NEQL                  3441 non-null   object
 47  SPI.D3.11.CITY                  3441 non-null   object
 48  SPI.D3.12.CNSP                  3441 non-null   object
 49  SPI.D3.15.LAND                  3441 non-null   object
 50  SPI.D3.16.INST                  3441 non-null   object
 51  SPI.D3.17.PTNS                  3441 non-null   object
 52  SPI.D3.13.CLMT                  3226 non-null   object
 53  SPI.D4.1.1.POPU                 863 non-null    object
 54  SPI.D4.1.2.AGRI                 863 non-null    object
 55  SPI.D4.1.3.BIZZ                 3041 non-null   object
```

```
 56  SPI.D4.1.4.HOUS                3041 non-null   object
 57  SPI.D4.1.5.AGSVY               3041 non-null   object
 58  SPI.D4.1.6.LABR                3057 non-null   object
 59  SPI.D4.1.7.HLTH                3089 non-null   object
 60  SPI.D4.1.8.BZSVY               3041 non-null   object
 61  SPI.D4.2.3.CRVS                1076 non-null   object
 62  SPI.D4.3.GEO.first.admin.level  840 non-null    object
 63  SPI.D5.1.DILG                  678 non-null    object
 64  SPI.D5.2.1.SNAU                863 non-null    object
 65  SPI.D5.2.2.NABY                863 non-null    object
 66  SPI.D5.2.3.CNIN                1181 non-null   object
 67  SPI.D5.2.4.CPIBY               958 non-null    object
 68  SPI.D5.2.5.HOUS                1183 non-null   object
 69  SPI.D5.2.6.EMPL                1181 non-null   object
 70  SPI.D5.2.7.CGOV                762 non-null    object
 71  SPI.D5.2.8.FINA                1181 non-null   object
 72  SPI.D5.2.9.MONY                763 non-null    object
 73  SPI.D5.2.10.GSBP               1181 non-null   object
 74  SPI.D5.5.DIFI                  673 non-null    object
 75  income                         3488 non-null   object
 76  region                         3488 non-null   object
 77  weights                        3488 non-null   float64
 78  population                     3464 non-null   float64
dtypes: float64(3), object(76)
memory usage: 2.1+ MB
```

In [4]:

```
b=a.fillna(value=104)
b
```

Out[4]:

	country	iso3c	date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.INDE
0	104	104	104.0	Pillar 1 - Data Use - Score	Pillar 2 - Data Services - Score	Pillar 3 - Data Products - Score	Pillar 4 Sources
1	Norway	NOR	2019.0	100	92.2333333333333	77.56875	80.6666666
2	Italy	ITA	2019.0	100	91.8666666666666	75.2875	
3	Austria	AUT	2019.0	100	91.3	74.55	
4	Poland	POL	2019.0	100	95.1	70.5375	79.7166666
...	...	...	...	...	...	...	
3484	Virgin Islands (U.S.)	VIR	2004.0	20	104	104	
3485	West Bank and Gaza	PSE	2004.0	20	104	104	
3486	Yemen, Rep.	YEM	2004.0	20	104	104	
3487	Zambia	ZMB	2004.0	40	104	104	
3488	Zimbabwe	ZWE	2004.0	20	104	104	

3489 rows × 79 columns

In [5]:

```
a.mean()
```

Out[5]:

```
date          2.011500e+03
weights       1.000000e+00
population    3.244250e+07
dtype: float64
```

In [7]:

```
a.median()
```

Out[7]:

```
date             2011.5
weights             1.0
population    5868890.0
dtype: float64
```

In [8]:

```
a.mode()
```

Out[8]:

	country	iso3c	date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.INDE
0	Afghanistan	ABW	2004.0	20	59.9	4.89375	39.9166666
1	Albania	AFG	2005.0	NaN	81.0333333333333	5.41875	53.8666666
2	Algeria	AGO	2006.0	NaN	NaN	9.8	
3	American Samoa	ALB	2007.0	NaN	NaN	NaN	
4	Andorra	AND	2008.0	NaN	NaN	NaN	
...	...	...	...	...	...	...	
685	NaN	NaN	NaN	NaN	NaN	NaN	
686	NaN	NaN	NaN	NaN	NaN	NaN	
687	NaN	NaN	NaN	NaN	NaN	NaN	
688	NaN	NaN	NaN	NaN	NaN	NaN	
689	NaN	NaN	NaN	NaN	NaN	NaN	

690 rows × 79 columns

In [9]:

```
a.describe()
```

Out[9]:

	date	weights	population
count	3488.000000	3488.0	3.464000e+03
mean	2011.500000	1.0	3.244250e+07
std	4.610433	0.0	1.293668e+08
min	2004.000000	1.0	9.828000e+03
25%	2007.750000	1.0	7.152712e+05
50%	2011.500000	1.0	5.868890e+06
75%	2015.250000	1.0	2.071792e+07
max	2019.000000	1.0	1.397715e+09

In [10]:

```
a.sum()
```

Out[10]:

```
date                                                                         7016
112.0
SPI.INDEX.PIL1                Pillar 1  - Data Use - Score1001001001001001
00...
SPI.DIM1.5.INDEX              Dimension 1.5: Data use by international org
an...
SPI.DIM5.1.INDEX              Dimension 5.1: Legislation and governance111
11...
SPI.DIM5.5.INDEX              Dimension 5.5: Finance1111111111111111111111
11...
SPI.D1.5.POV                  Dimension 1.5: Data use by international org
an...
SPI.D1.5.CHLD.MORT            Dimension 1.5: Data use by international org
an...
SPI.D1.5.DT.TDS.DPPF.XP.ZS    Dimension 1.5: Data use by international org
an...
SPI.D1.5.SAFE.MAN.WATER       Dimension1.5: Data use by international orga
ni...
SPI.D1.5.LFP                  Dimension 1.5: Data use by international org
an...
weights                                                                         3
488.0
population                                                              112380821
894.5
dtype: object
```

In [12]:

```
a.count()
```

Out[12]:

```
country            3488
iso3c              3488
date               3488
SPI.INDEX.PIL1     3489
SPI.INDEX.PIL2      693
                   ...
SPI.D5.5.DIFI       673
income             3488
region             3488
weights            3488
population         3464
Length: 79, dtype: int64
```

In [13]:

```
a.max()
```

Out[13]:

```
date                                                                   2
019.0
SPI.INDEX.PIL1                                      Pillar 1  - Data Use -
Score
SPI.DIM1.5.INDEX            Dimension 1.5: Data use by international org
an...
SPI.DIM5.1.INDEX                    Dimension 5.1: Legislation and gover
nance
SPI.DIM5.5.INDEX                                      Dimension 5.5: Fi
nance
SPI.D1.5.POV                Dimension 1.5: Data use by international org
an...
SPI.D1.5.CHLD.MORT          Dimension 1.5: Data use by international org
an...
SPI.D1.5.DT.TDS.DPPF.XP.ZS  Dimension 1.5: Data use by international org
an...
SPI.D1.5.SAFE.MAN.WATER     Dimension1.5: Data use by international orga
ni...
SPI.D1.5.LFP                Dimension 1.5: Data use by international org
an...
weights
1.0
population                                                       1397715
000.0
dtype: object
```

In [14]:

```
a.dropna()
```

Out[14]:

	country	iso3c	date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.IND
1	Norway	NOR	2019.0	100	92.2333333333333	77.56875	80.666666
2	Italy	ITA	2019.0	100	91.8666666666666	75.2875	
3	Austria	AUT	2019.0	100	91.3	74.55	
4	Poland	POL	2019.0	100	95.1	70.5375	79.716666
5	Slovenia	SVN	2019.0	100	96.9333333333333	76.28125	71.441666
...	...	...	...	...	...	...	
694	Belgium	BEL	2016.0	100	53.1333333333333	60.4125	66.216666
695	Israel	ISR	2016.0	100	81.8666666666667	52.375	44.308333
708	Iceland	ISL	2016.0	80	49.5666666666667	54.75	62.891666
710	Luxembourg	LUX	2016.0	80	49.0666666666667	53.8875	60.516666
722	Singapore	SGP	2016.0	100	48.0333333333333	40.94375	50.033333

230 rows × 79 columns

In [15]:

```
a.isna()
```

Out[15]:

	country	iso3c	date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.INDEX.PIL4
0	True	True	True	False	False	False	False
1	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...
3484	False	False	False	False	True	True	True
3485	False	False	False	False	True	True	True
3486	False	False	False	False	True	True	True
3487	False	False	False	False	True	True	True
3488	False	False	False	False	True	True	True

3489 rows × 79 columns

In [16]:

```
b.columns
```

Out[16]:

```
Index(['country', 'iso3c', 'date', 'SPI.INDEX.PIL1', 'SPI.INDEX.PIL2',
       'SPI.INDEX.PIL3', 'SPI.INDEX.PIL4', 'SPI.INDEX.PIL5', 'SPI.INDEX',
       'SPI.DIM1.5.INDEX', 'SPI.DIM2.1.INDEX', 'SPI.DIM2.2.INDEX',
       'SPI.DIM2.4.INDEX', 'SPI.DIM3.1.INDEX', 'SPI.DIM3.2.INDEX',
       'SPI.DIM3.3.INDEX', 'SPI.DIM3.4.INDEX', 'SPI.DIM4.1.CEN.INDEX',
       'SPI.DIM4.1.SVY.INDEX', 'SPI.DIM4.2.INDEX', 'SPI.DIM4.3.INDEX',
       'SPI.DIM5.1.INDEX', 'SPI.DIM5.2.INDEX', 'SPI.DIM5.5.INDEX',
       'SPI.D1.5.POV', 'SPI.D1.5.CHLD.MORT', 'SPI.D1.5.DT.TDS.DPPF.XP.ZS',
       'SPI.D1.5.SAFE.MAN.WATER', 'SPI.D1.5.LFP', 'SPI.D2.1.GDDS',
       'SPI.D2.2.Machine.readable', 'SPI.D2.2.Non.proprietary',
       'SPI.D2.2.Download.options', 'SPI.D2.2.Metadata.available',
       'SPI.D2.2.Terms.of.use', 'SPI.D2.2.Openness.subscore', 'SPI.D2.4.NA
DA',
       'SPI.D3.1.POV', 'SPI.D3.2.HNGR', 'SPI.D3.3.HLTH', 'SPI.D3.4.EDUC',
       'SPI.D3.5.GEND', 'SPI.D3.6.WTRS', 'SPI.D3.7.ENRG', 'SPI.D3.8.WORK',
       'SPI.D3.9.INDY', 'SPI.D3.10.NEQL', 'SPI.D3.11.CITY', 'SPI.D3.12.CNS
P',
       'SPI.D3.15.LAND', 'SPI.D3.16.INST', 'SPI.D3.17.PTNS', 'SPI.D3.13.CL
MT',
       'SPI.D4.1.1.POPU', 'SPI.D4.1.2.AGRI', 'SPI.D4.1.3.BIZZ',
       'SPI.D4.1.4.HOUS', 'SPI.D4.1.5.AGSVY', 'SPI.D4.1.6.LABR',
       'SPI.D4.1.7.HLTH', 'SPI.D4.1.8.BZSVY', 'SPI.D4.2.3.CRVS',
       'SPI.D4.3.GEO.first.admin.level', 'SPI.D5.1.DILG', 'SPI.D5.2.1.SNA
U',
       'SPI.D5.2.2.NABY', 'SPI.D5.2.3.CNIN', 'SPI.D5.2.4.CPIBY',
       'SPI.D5.2.5.HOUS', 'SPI.D5.2.6.EMPL', 'SPI.D5.2.7.CGOV',
       'SPI.D5.2.8.FINA', 'SPI.D5.2.9.MONY', 'SPI.D5.2.10.GSBP',
       'SPI.D5.5.DIFI', 'income', 'region', 'weights', 'population'],
      dtype='object')
```

In [20]:

```
c=b.tail(10)
c
```

Out[20]:

	country	iso3c	date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.INDEX.PI
3479	Uruguay	URY	2004.0	40	104	104	1
3480	Uzbekistan	UZB	2004.0	40	104	104	1
3481	Vanuatu	VUT	2004.0	20	104	104	1
3482	Venezuela, RB	VEN	2004.0	40	104	104	1
3483	Vietnam	VNM	2004.0	40	104	104	1
3484	Virgin Islands (U.S.)	VIR	2004.0	20	104	104	1
3485	West Bank and Gaza	PSE	2004.0	20	104	104	1
3486	Yemen, Rep.	YEM	2004.0	20	104	104	1
3487	Zambia	ZMB	2004.0	40	104	104	1
3488	Zimbabwe	ZWE	2004.0	20	104	104	1

10 rows × 79 columns

In [21]:

```
d=c[[ 'SPI.DIM3.1.INDEX', 'SPI.DIM3.2.INDEX',
       'SPI.DIM3.3.INDEX', 'SPI.DIM3.4.INDEX', 'SPI.DIM4.1.CEN.INDEX',
       'SPI.DIM4.1.SVY.INDEX']]
d
```

Out[21]:

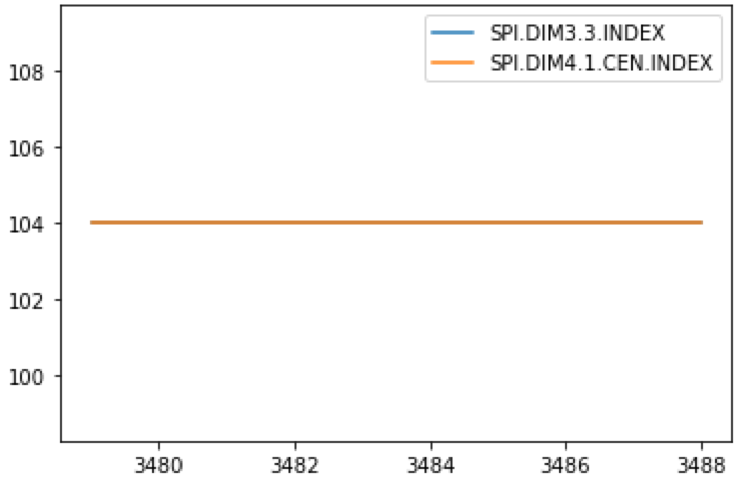
	SPI.DIM3.1.INDEX	SPI.DIM3.2.INDEX	SPI.DIM3.3.INDEX	SPI.DIM3.4.INDEX	SPI.DIM4.1
3479	0.640333333333333	0.406166666666667	104	0.2	
3480	0.375	0.388166666666667	104	0.143	
3481	0.296166666666667	0.316833333333333	104	0.2855	
3482	0.482833333333333	0.3925	104	0.5355	
3483	0.443	0.353166666666667	104	0.4	
3484	0.0166666666666667	0.170833333333333	104	0.25	
3485	0.514166666666667	0.3235	104	0.2145	
3486	0.334333333333333	0.338333333333333	104	0.25	
3487	0.390666666666667	0.3645	104	0.4	
3488	0.334333333333333	0.371666666666667	104	0.143	

In [22]:

```
d.plot.line()
```

Out[22]:

<AxesSubplot:>

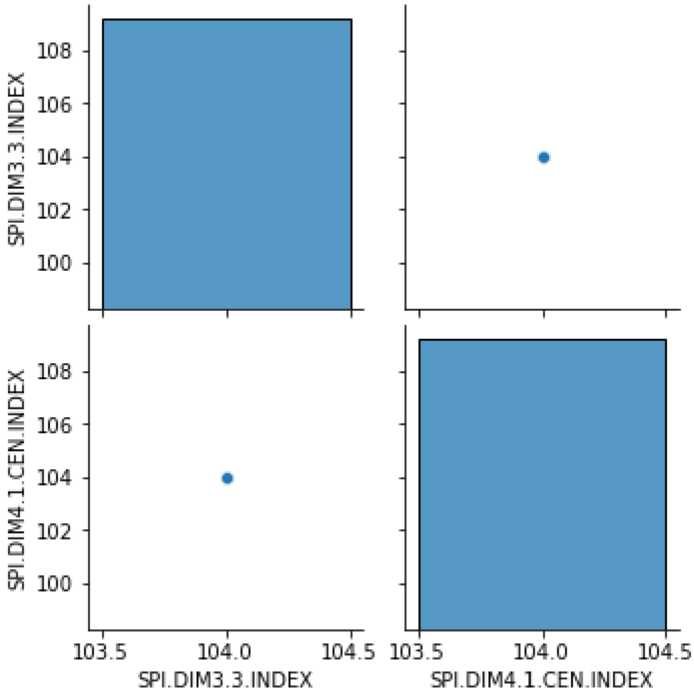


In [23]:

```
sns.pairplot(d)
```

Out[23]:

<seaborn.axisgrid.PairGrid at 0x121a25e5430>



In [24]:

```
x=d[['SPI.DIM3.1.INDEX', 'SPI.DIM3.2.INDEX', 'SPI.DIM3.4.INDEX', 'SPI.DIM4.1.CEN.INDEX',
       'SPI.DIM4.1.SVY.INDEX']]
y=d['SPI.DIM3.3.INDEX']
```

In [25]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

In [26]:

```
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[26]:

LinearRegression()

In [27]:

```
print(lr.intercept_)
```

104.0

In [28]:

```
f=pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
f
```

Out[28]:

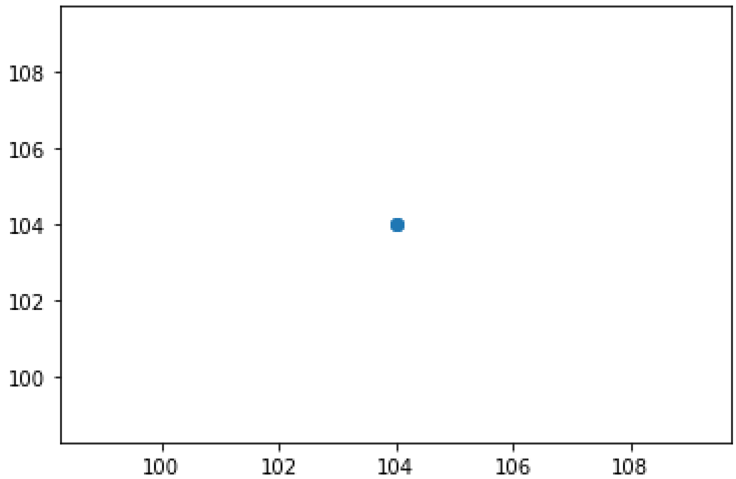
	Co-efficient
SPI.DIM3.1.INDEX	0.0
SPI.DIM3.2.INDEX	0.0
SPI.DIM3.4.INDEX	0.0
SPI.DIM4.1.CEN.INDEX	0.0
SPI.DIM4.1.SVY.INDEX	0.0

In [29]:

```
prediction=lr.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[29]:

```
<matplotlib.collections.PathCollection at 0x121a6eb7730>
```



In [30]:

```
print(lr.score(x_test,y_test))
```

```
1.0
```

In [31]:

```
from sklearn.linear_model import Ridge,Lasso
```

In [32]:

```
rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
```

Out[32]:

```
Ridge(alpha=10)
```

In [33]:

```
rr.score(x_test,y_test)
```

Out[33]:

```
1.0
```

In [34]:

```
la=Lasso(alpha=10)
la.fit(x_train,y_train)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear_model\_coordinat
e_descent.py:530: ConvergenceWarning: Objective did not converge. You migh
t want to increase the number of iterations. Duality gap: 0.0, tolerance:
0.0
  model = cd_fast.enet_coordinate_descent(
```

Out[34]:

```
Lasso(alpha=10)
```

In [35]:

```
la.score(x_test,y_test)
```

Out[35]:

```
1.0
```

In [39]:

```
a1=b.tail(500)
a1
```

Out[39]:

date	SPI.INDEX.PIL1	SPI.INDEX.PIL2	SPI.INDEX.PIL3	SPI.INDEX.PIL4	SPI.INDEX.PIL5	SPI.I
2006.0	40	104	69.525	104	104	
2006.0	40	104	65.86875	104	104	
2006.0	40	104	42.3375	104	104	
2006.0	40	104	44.53125	104	104	
2006.0	20	104	19.84375	104	104	
...	...	...	...	...	...	
2004.0	20	104	104	104	104	
2004.0	20	104	104	104	104	
2004.0	20	104	104	104	104	
2004.0	40	104	104	104	104	
2004.0	20	104	104	104	104	