In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sb
```

In [10]:
```
df=pd.read_csv(r"C:\Users\Dell\OneDrive\Desktop\d_data\data1.csv")
```

In [11]:
```
df
```

Out[11]:

	States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011-LIT	2001 - POP	2011-POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP
0	Andaman and Nicobar Islands	25047	89642	81.30	86.63	356	381	846	876	34
1	Andhra Pradesh	17195	64773	60.47	67.02	76210	84581	978	993	8
2	Arunachal Pradesh	15260	71366	54.34	65.39	1098	1384	893	938	5
3	Assam	12803	36320	63.25	72.19	26656	31206	935	958	39
4	Bihar	6415	22582	47.00	61.80	82999	104099	919	918	18
5	Chandigarh	49771	136883	81.94	86.05	901	1055	777	818	7
6	Chhattisgarh	10744	48366	64.66	70.28	20834	25545	989	991	16
7	Delhi	40678	161446	57.63	76.24	13851	16788	821	868	47
8	Goa	43735	211570	78.18	87.10	1348	1459	961	973	93
9	Gujarat	18392	85979	81.67	86.21	50671	60440	920	919	4
10	Haryana	25583	106320	82.01	88.70	21145	25351	861	879	8
11	Himachal Pradesh	22795	75185	69.14	78.03	6078	6865	968	972	12
12	Jammu and Kashmir	14268	46734	67.91	75.55	10144	12541	892	889	11
13	Jharkhand	10345	36554	76.48	82.80	26946	32988	941	948	20
14	Karnataka	18344	68053	55.52	67.16	52851	61095	965	973	7
15	Kerala	20094	82753	53.56	66.41	31841	33406	1058	1084	82
16	Madhya Pradesh	11862	37180	66.60	75.37	60348	72627	919	931	5
17	Maharashtra	22777	93282	90.86	94.00	96879	112374	922	929	14
18	Manipur	12369	33695	86.66	91.85	2294	2856	978	992	19
19	Meghalaya	15657	50316	63.74	69.32	2319	2967	972	989	4
20	Mizoram	17826	53624	76.88	82.34	889	1097	935	976	9
21	Nagaland	16253	63781	70.50	76.90	1990	1979	900	931	24
22	Odisha	10453	43463	62.56	74.43	36805	41974	972	979	19
23	Puducherry	35994	103149	88.80	91.33	974	1248	1001	1037	40
24	Punjab	27881	76895	66.59	79.60	24359	27743	876	895	18
25	Rajasthan	13020	54637	63.08	72.89	56507	68548	921	928	4
26	Sikkim	16077	130127	81.24	85.85	541	611	875	890	28
27	Tamil Nadu	20972	89050	69.65	75.84	62406	72147	987	996	20
28	Tripura	15983	50859	68.81	81.42	3199	3674	948	960	12
29	Uttar Pradesh	9828	30021	73.45	80.09	166198	199812	898	912	8
30	Uttarakhand	15285	85372	73.19	87.22	8489	10086	962	963	22

	States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011-LIT	2001 - POP	2011-POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP
31	West Bengal	16583	53383	56.27	67.68	80176	91276	934	950	28

In [8]:
```
## removing first column
df=pd.read_csv(r"C:\Users\Dell\OneDrive\Desktop\d_data\RBI DATA states_wise_pop
```

In [5]:
```
df
```

Out[5]:

	States_Union Territories\t2000-01-INC\t2011-12-INC\t2001 - LIT\t2011- LIT\t2001 - POP\t2011-POP\t2001 -SEX_Ratio\t2011 -SEX_Ratio\t2001 -UNEMP\t2011 -UNEMP\t2001 -Poverty\t2011 -Poverty\tregion
0	Andaman and Nicobar Islands\t25047\t89642\t81....
1	Andhra Pradesh\t17195\t64773\t60.47\t67.02\t76...
2	Arunachal Pradesh\t15260\t71366\t54.34\t65.39\...
3	Assam\t12803\t36320\t63.25\t72.19\t26656\t3120...
4	Bihar\t6415\t22582\t47\t61.8\t82999\t104099\t9...
5	Chandigarh\t49771\t136883\t81.94\t86.05\t901\t...
6	Chhattisgarh\t10744\t48366\t64.66\t70.28\t2083...
7	Delhi\t40678\t161446\t57.63\t76.24\t13851\t167...
8	Goa\t43735\t211570\t78.18\t87.1\t1348\t1459\t9...
9	Gujarat\t18392\t85979\t81.67\t86.21\t50671\t60...
10	Haryana\t25583\t106320\t82.01\t88.7\t21145\t25...
11	Himachal Pradesh\t22795\t75185\t69.14\t78.03\t...
12	Jammu and Kashmir\t14268\t46734\t67.91\t75.55\...
13	Jharkhand\t10345\t36554\t76.48\t82.8\t26946\t3...
14	Karnataka\t18344\t68053\t55.52\t67.16\t52851\t...
15	Kerala\t20094\t82753\t53.56\t66.41\t31841\t334...
16	Madhya Pradesh\t11862\t37180\t66.6\t75.37\t603...
17	Maharashtra\t22777\t93282\t90.86\t94\t96879\t1...
18	Manipur\t12369\t33695\t86.66\t91.85\t2294\t285...
19	Meghalaya\t15657\t50316\t63.74\t69.32\t2319\t2...
20	Mizoram\t17826\t53624\t76.88\t82.34\t889\t1097...
21	Nagaland\t16253\t63781\t70.5\t76.9\t1990\t1979...
22	Odisha\t10453\t43463\t62.56\t74.43\t36805\t419...
23	Puducherry\t35994\t103149\t88.8\t91.33\t974\t1...
24	Punjab\t27881\t76895\t66.59\t79.6\t24359\t2774...
25	Rajasthan\t13020\t54637\t63.08\t72.89\t56507\t...
26	Sikkim\t16077\t130127\t81.24\t85.85\t541\t611\...
27	Tamil Nadu\t20972\t89050\t69.65\t75.84\t62406\...
28	Tripura\t15983\t50859\t68.81\t81.42\t3199\t367...
29	Uttar Pradesh\t9828\t30021\t73.45\t80.09\t1661...
30	Uttarakhand\t15285\t85372\t73.19\t87.22\t8489\...
31	West Bengal\t16583\t53383\t56.27\t67.68\t80176...

In [12]:
```
df.dtypes
```

Out[12]:
```
States_Union Territories     object
2000-01-INC                   int64
2011-12-INC                   int64
2001 - LIT                  float64
2011- LIT                   float64
2001 - POP                    int64
2011- POP                     int64
2001 -SEX_Ratio               int64
2011 -SEX_Ratio               int64
2001 -UNEMP                   int64
2011 -UNEMP                   int64
2001 -Poverty               float64
2011 -Poverty               float64
region                       object
dtype: object
```

In [13]:
```
df.size('2011-12-INC' )
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_8816\3938939694.py in <module>
----> 1 df.size('2011-12-INC' )

TypeError: 'numpy.int32' object is not callable
```

In [10]:
```
df.index  ## name of states
```

Out[10]:
```
RangeIndex(start=0, stop=32, step=1)
```

In [11]:
```
df.columns #specify the columns in data
```

Out[11]:
```
Index(['States_Union Territories', '2000-01-INC', '2011-12-INC', '2001 - LI
T',
       '2011- LIT', '2001 - POP', '2011- POP', '2001 -SEX_Ratio',
       '2011 -SEX_Ratio', '2001 -UNEMP', '2011 -UNEMP', '2001 -Poverty',
       '2011 -Poverty'],
      dtype='object')
```

In [73]:
```
df.memory_usage()  ## describe the data
```

Out[73]:
```
Index                        128
States_Union Territories     256
2000-01-INC                  256
2011-12-INC                  256
2001 - LIT                   256
2011- LIT                    256
2001 - POP                   256
2011- POP                    256
2001 -SEX_Ratio              256
2011 -SEX_Ratio              256
2001 -UNEMP                  256
2011 -UNEMP                  256
2001 -Poverty                256
2011 -Poverty                256
dtype: int64
```

In [27]:
```
df.head()
```

Out[27]:

States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011- LIT	2001 - POP	2011- POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP	2011 UNEMI
Andaman and Nicobar Islands	25047	89642	81.30	86.63	356	381	846	876	34	5
Andhra Pradesh	17195	64773	60.47	67.02	76210	84581	978	993	8	1
Arunachal Pradesh	15260	71366	54.34	65.39	1098	1384	893	938	5	1
Assam	12803	36320	63.25	72.19	26656	31206	935	958	39	4
Bihar	6415	22582	47.00	61.80	82999	104099	919	918	18	3

In [31]:
```
df.head(31)
```

Out[31]:

States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011-LIT	2001 - POP	2011-POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP	20 UNE
Andaman and Nicobar Islands	25047	89642	81.30	86.63	356	381	846	876	34	
Andhra Pradesh	17195	64773	60.47	67.02	76210	84581	978	993	8	
Arunachal Pradesh	15260	71366	54.34	65.39	1098	1384	893	938	5	
Assam	12803	36320	63.25	72.19	26656	31206	935	958	39	
Bihar	6415	22582	47.00	61.80	82999	104099	919	918	18	
Chandigarh	49771	136883	81.94	86.05	901	1055	777	818	7	
Chhattisgarh	10744	48366	64.66	70.28	20834	25545	989	991	16	
Delhi	40678	161446	57.63	76.24	13851	16788	821	868	47	
Goa	43735	211570	78.18	87.10	1348	1459	961	973	93	
Gujarat	18392	85979	81.67	86.21	50671	60440	920	919	4	
Haryana	25583	106320	82.01	88.70	21145	25351	861	879	8	
Himachal Pradesh	22795	75185	69.14	78.03	6078	6865	968	972	12	
Jammu and Kashmir	14268	46734	67.91	75.55	10144	12541	892	889	11	
Jharkhand	10345	36554	76.48	82.80	26946	32988	941	948	20	
Karnataka	18344	68053	55.52	67.16	52851	61095	965	973	7	
Kerala	20094	82753	53.56	66.41	31841	33406	1058	1084	82	
Madhya Pradesh	11862	37180	66.60	75.37	60348	72627	919	931	5	
Maharashtra	22777	93282	90.86	94.00	96879	112374	922	929	14	
Manipur	12369	33695	86.66	91.85	2294	2856	978	992	19	
Meghalaya	15657	50316	63.74	69.32	2319	2967	972	989	4	
Mizoram	17826	53624	76.88	82.34	889	1097	935	976	9	
Nagaland	16253	63781	70.50	76.90	1990	1979	900	931	24	
Odisha	10453	43463	62.56	74.43	36805	41974	972	979	19	
Puducherry	35994	103149	88.80	91.33	974	1248	1001	1037	40	
Punjab	27881	76895	66.59	79.60	24359	27743	876	895	18	
Rajasthan	13020	54637	63.08	72.89	56507	68548	921	928	4	
Sikkim	16077	130127	81.24	85.85	541	611	875	890	28	
Tamil Nadu	20972	89050	69.65	75.84	62406	72147	987	996	20	
Tripura	15983	50859	68.81	81.42	3199	3674	948	960	12	

States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011- LIT	2001 - POP	2011- POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP	20 UNE
Uttar Pradesh	9828	30021	73.45	80.09	166198	199812	898	912	8	
Uttarakhand	15285	85372	73.19	87.22	8489	10086	962	963	22	

In [32]:
```
df.tail(5)
```

Out[32]:

States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011- LIT	2001 - POP	2011- POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP	201 UNEM
Tamil Nadu	20972	89050	69.65	75.84	62406	72147	987	996	20	
Tripura	15983	50859	68.81	81.42	3199	3674	948	960	12	1
Uttar Pradesh	9828	30021	73.45	80.09	166198	199812	898	912	8	
Uttarakhand	15285	85372	73.19	87.22	8489	10086	962	963	22	
West Bengal	16583	53383	56.27	67.68	80176	91276	934	950	28	

In [35]:
```
for col in df.columns:
    print(col)
```

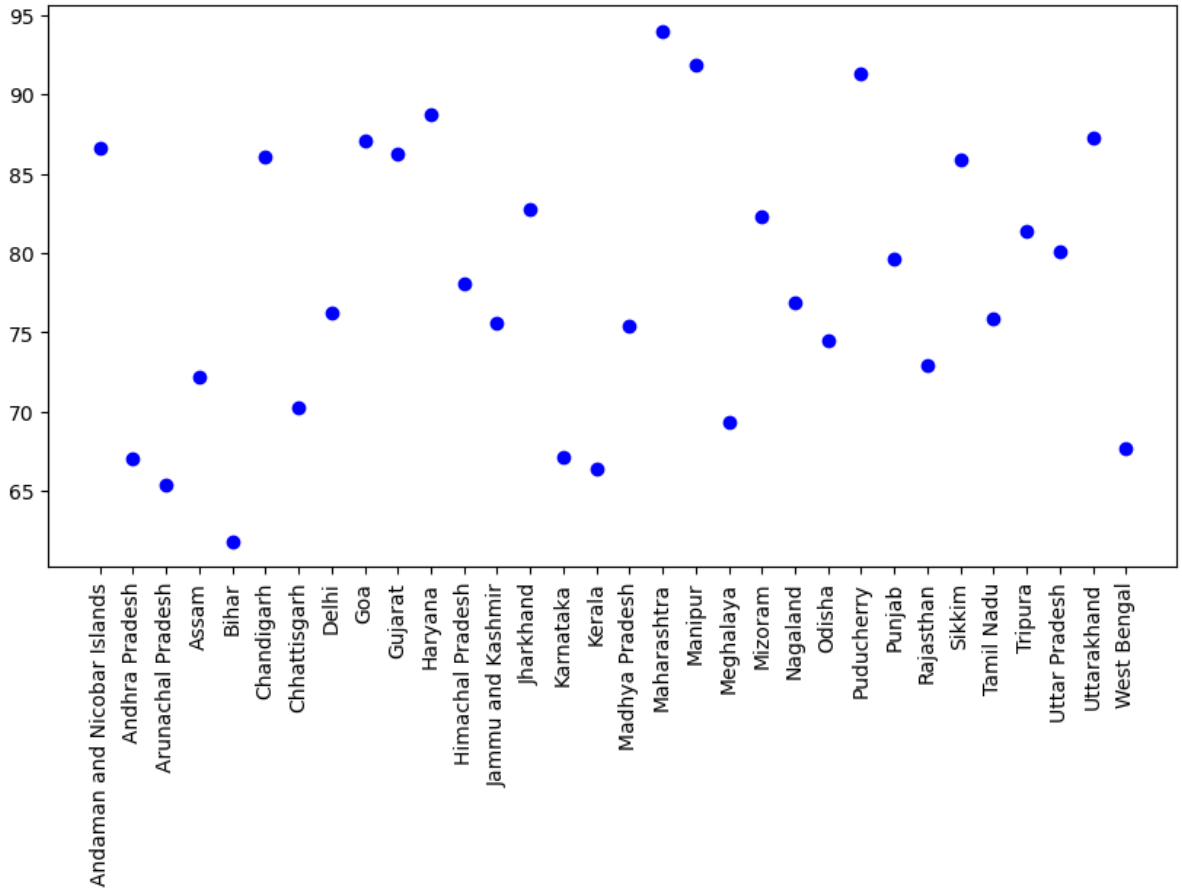
```
2000-01-INC
2011-12-INC
2001 - LIT
2011- LIT
2001 - POP
2011- POP
2001 -SEX_Ratio
2011 -SEX_Ratio
2001 -UNEMP
2011 -UNEMP
2001 -Poverty
2011 -Poverty
```

In [37]:
```
df1=df.loc[:,"2011- LIT"]
print(df1) ## accessing rows and columns in data
```

```
States_Union Territories
Andaman and Nicobar Islands    86.63
Andhra Pradesh                 67.02
Arunachal Pradesh              65.39
Assam                          72.19
Bihar                          61.80
Chandigarh                     86.05
Chhattisgarh                   70.28
Delhi                          76.24
Goa                            87.10
Gujarat                        86.21
Haryana                        88.70
Himachal Pradesh               78.03
Jammu and Kashmir              75.55
Jharkhand                      82.80
Karnataka                      67.16
Kerala                         66.41
Madhya Pradesh                 75.37
Maharashtra                    94.00
```

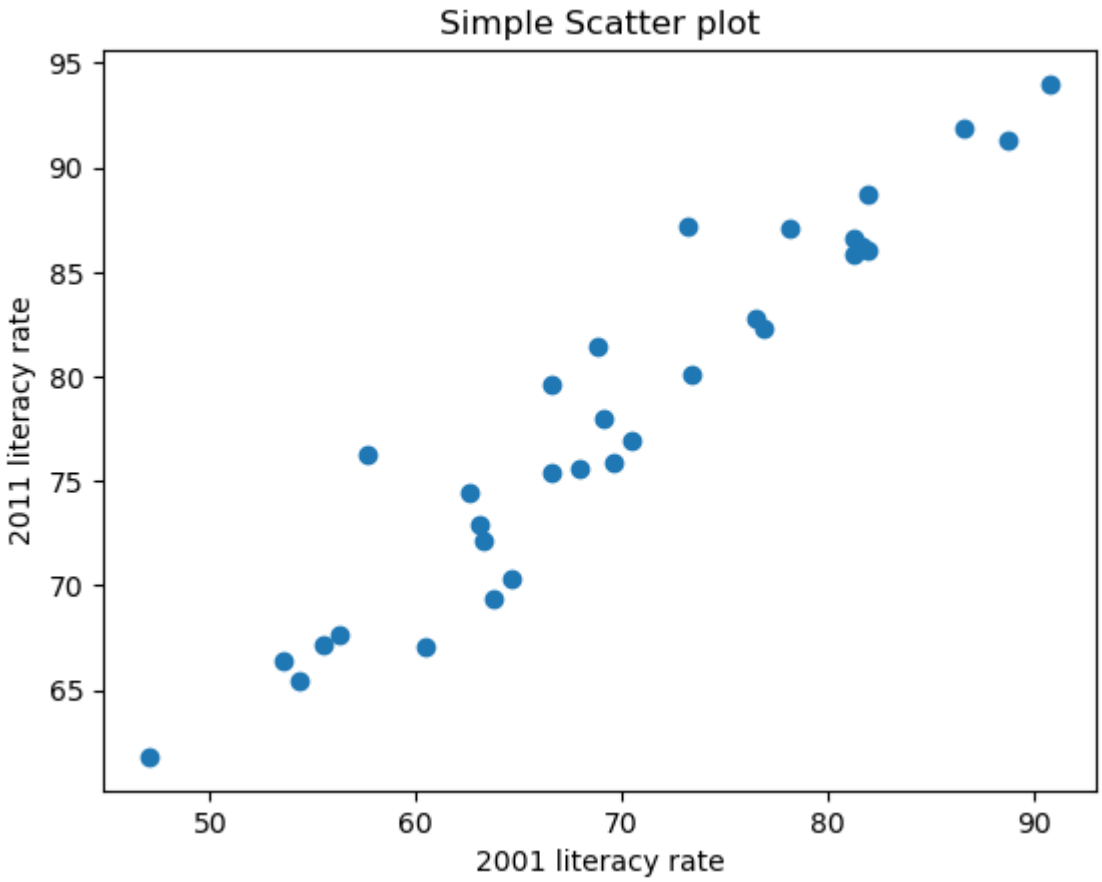
In [14]:
```
plt.figure(figsize=(10,5))
plt.xticks(rotation=90)
plt.scatter(df["States_Union Territories"],df["2011- LIT"],c='blue')
```

Out[14]: <matplotlib.collections.PathCollection at 0x23969b98d60>

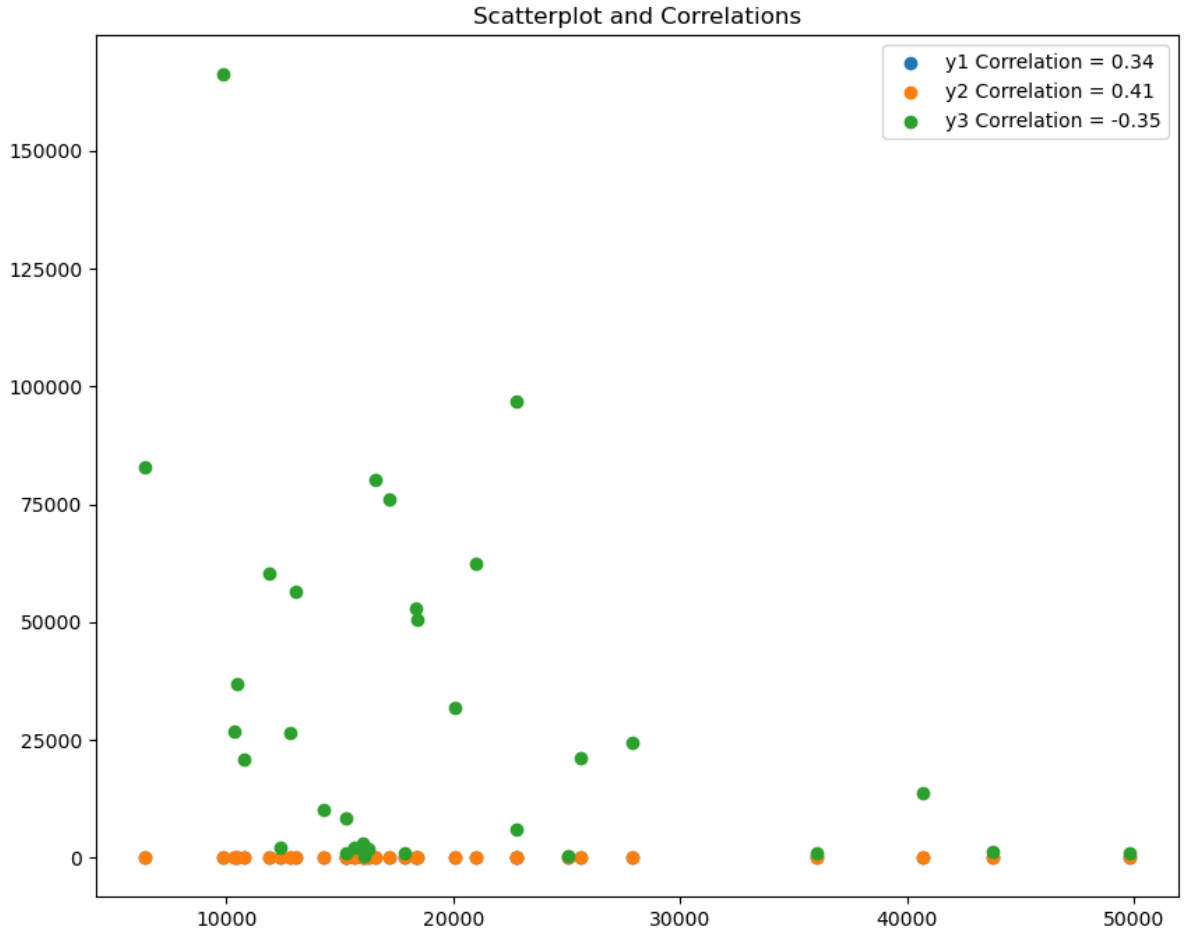


In [ ]:
```
```

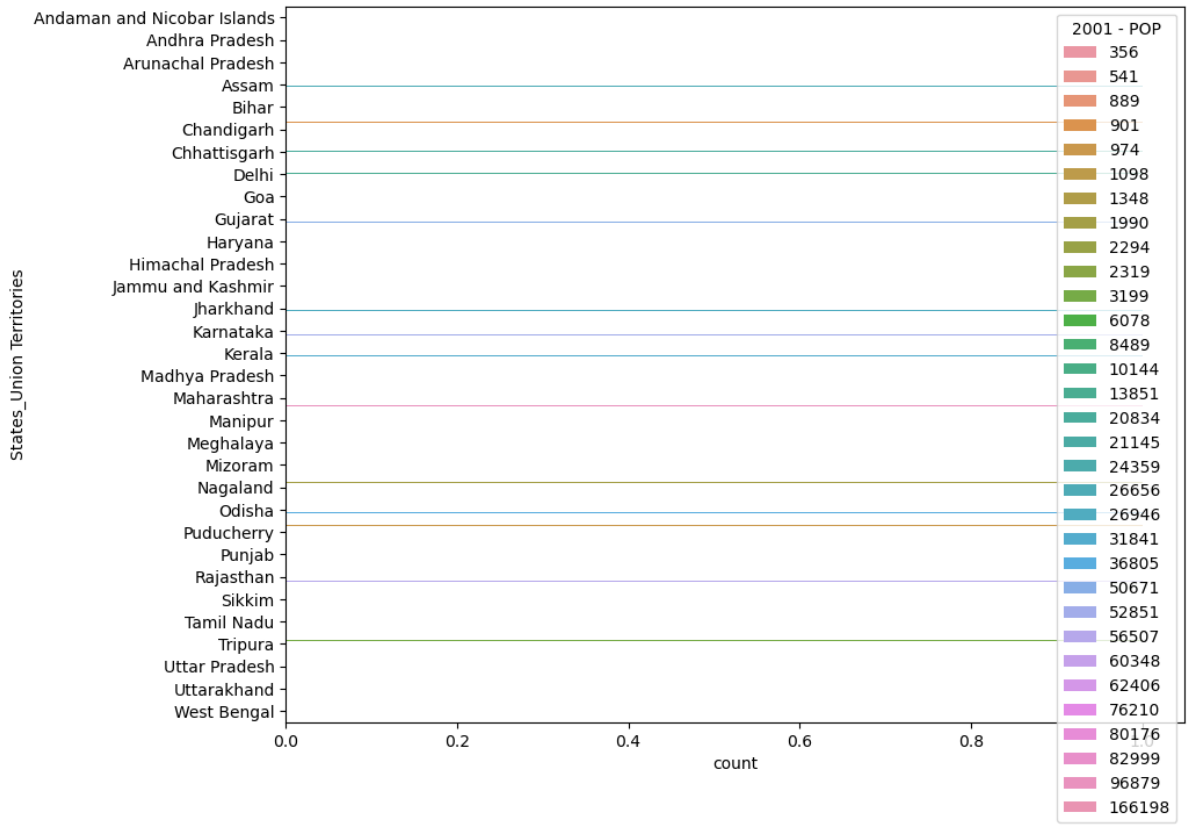
In [60]:
```
x=df["2001 - LIT"]
y=df["2011- LIT"]
plt.scatter(x, y)
plt.rcParams.update({'figure.figsize':(10,8), 'figure.dpi':100})
plt.title('Simple Scatter plot')
plt.xlabel('2001 literacy rate')
plt.ylabel('2011 literacy rate')
plt.show()
```



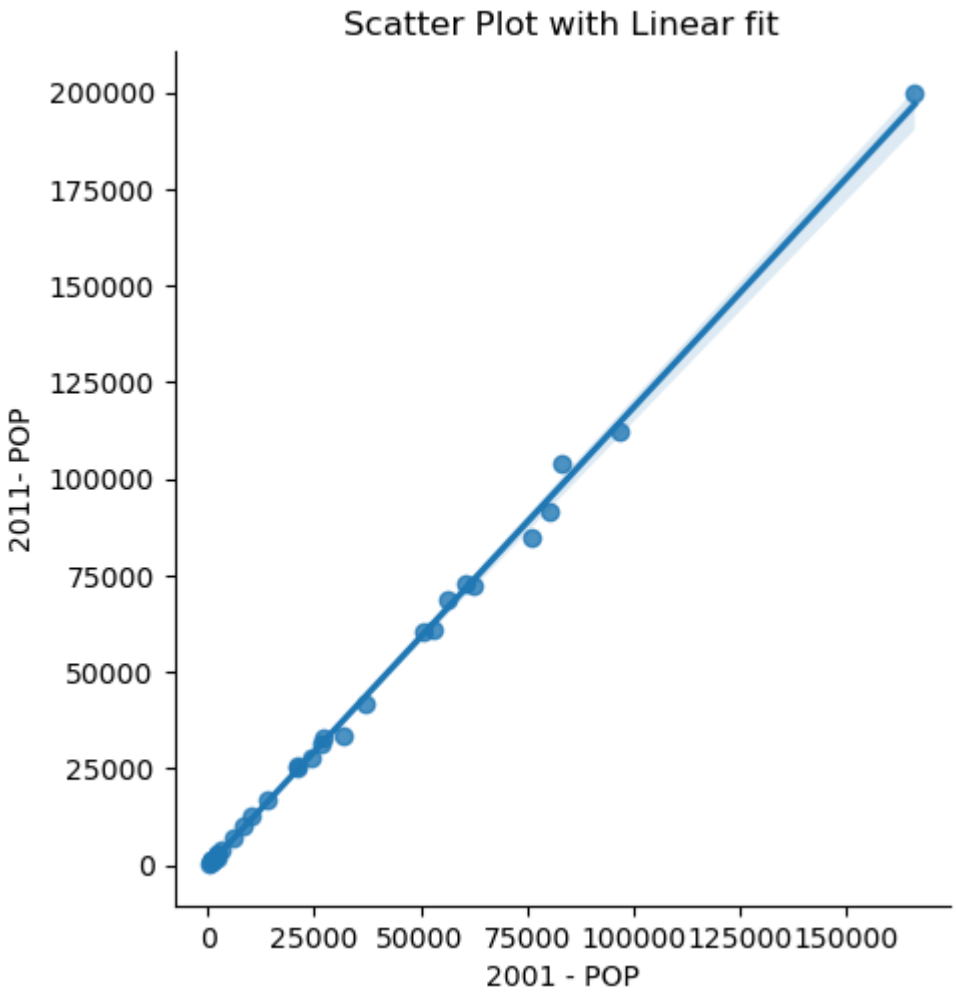
In [62]:
```
## correlation and scatter plot
# Plot
x=df["2000-01-INC"]
y1=df["2001 - LIT"]
y2=df["2011- LIT"]
y3=df["2001 - POP"]
plt.rcParams.update({'figure.figsize':(10,8), 'figure.dpi':100})
plt.scatter(x, y1, label=f'y1 Correlation = {np.round(np.corrcoef(x,y1)[0,1], 2
plt.scatter(x, y2, label=f'y2 Correlation = {np.round(np.corrcoef(x,y2)[0,1], 2
plt.scatter(x, y3, label=f'y3 Correlation = {np.round(np.corrcoef(x,y3)[0,1], 2

# Plot
plt.title('Scatterplot and Correlations')
plt.legend()
plt.show()
```



In [74]:
```
sb.countplot(y="States_Union Territories", hue="2001 - POP", data=df);
```



In [78]:
```
plt.rcParams.update({'figure.figsize':(10,8), 'figure.dpi':100})
sb.lmplot(x='2001 - POP', y='2011- POP', data=df)
plt.title("Scatter Plot with Linear fit");
```



In [13]:
```
corr_matrix = df.corr()
```

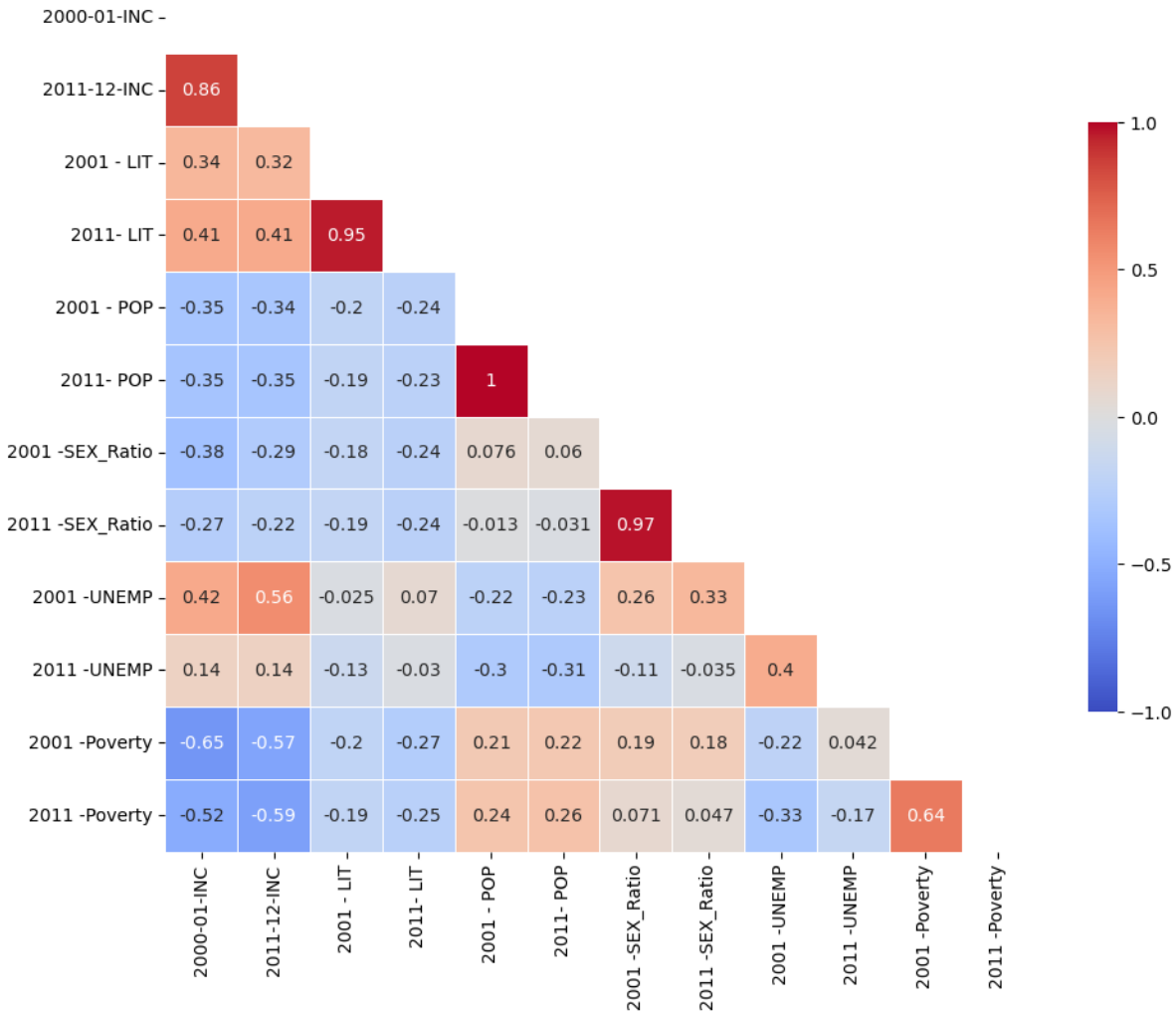
In [15]:
```
print(corr_matrix)
```

```
                 2000-01-INC  2011-12-INC  2001 - LIT  2011- LIT  2001 - POP
\
2000-01-INC         1.000000     0.856503    0.339893   0.410330   -0.347463
2011-12-INC         0.856503     1.000000    0.320797   0.409660   -0.341904
2001 - LIT          0.339893     0.320797    1.000000   0.954714   -0.197126
2011- LIT           0.410330     0.409660    0.954714   1.000000   -0.236801
2001 - POP         -0.347463    -0.341904   -0.197126  -0.236801    1.000000
2011- POP          -0.354727    -0.349059   -0.194548  -0.232625    0.998999
2001 -SEX_Ratio    -0.376493    -0.292318   -0.184731  -0.239140    0.075535
2011 -SEX_Ratio    -0.267230    -0.224162   -0.186974  -0.241932   -0.013069
2001 -UNEMP         0.418115     0.556975   -0.025393   0.069774   -0.221888
2011 -UNEMP         0.136922     0.142222   -0.130303  -0.030493   -0.303407
2001 -Poverty      -0.646359    -0.567767   -0.199488  -0.274090    0.213448
2011 -Poverty      -0.522265    -0.587595   -0.188766  -0.246837    0.242429

                 2011- POP  2001 -SEX_Ratio  2011 -SEX_Ratio  2001 -UNEMP  \
2000-01-INC      -0.354727        -0.376493        -0.267230     0.418115
2011-12-INC      -0.349059        -0.292318        -0.224162     0.556975
2001 - LIT       -0.194548        -0.184731        -0.186974    -0.025393
2011- LIT        -0.232625        -0.239140        -0.241932     0.069774
2001 - POP        0.998999         0.075535        -0.013069    -0.221888
2011- POP         1.000000         0.060463        -0.030603    -0.231557
2001 -SEX_Ratio   0.060463         1.000000         0.969892     0.257778
2011 -SEX_Ratio  -0.030603         0.969892         1.000000     0.330296
2001 -UNEMP      -0.231557         0.257778         0.330296     1.000000
2011 -UNEMP      -0.305636        -0.113035        -0.034893     0.403913
2001 -Poverty     0.223641         0.190688         0.180561    -0.217504
2011 -Poverty     0.258446         0.071169         0.046818    -0.332960

                 2011 -UNEMP  2001 -Poverty  2011 -Poverty
2000-01-INC         0.136922      -0.646359      -0.522265
2011-12-INC         0.142222      -0.567767      -0.587595
2001 - LIT         -0.130303      -0.199488      -0.188766
2011- LIT          -0.030493      -0.274090      -0.246837
2001 - POP         -0.303407       0.213448       0.242429
2011- POP          -0.305636       0.223641       0.258446
2001 -SEX_Ratio    -0.113035       0.190688       0.071169
2011 -SEX_Ratio    -0.034893       0.180561       0.046818
2001 -UNEMP         0.403913      -0.217504      -0.332960
2011 -UNEMP         1.000000       0.042132      -0.170415
2001 -Poverty       0.042132       1.000000       0.642361
2011 -Poverty      -0.170415       0.642361       1.000000
```

In [20]:
```
mask = np.zeros_like(corr_matrix, dtype=np.bool)
mask[np.triu_indices_from(mask)]= True
```

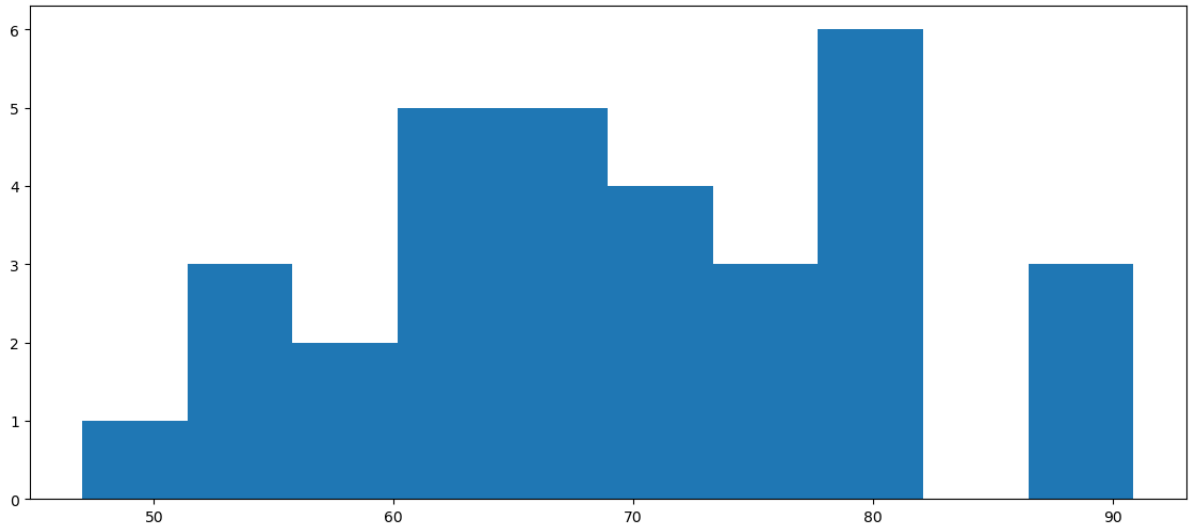
```
C:\Users\Dell\AppData\Local\Temp\ipykernel_7332\1276619264.py:1: DeprecationW
arning: `np.bool` is a deprecated alias for the builtin `bool`. To silence th
is warning, use `bool` by itself. Doing this will not modify any behavior and
is safe. If you specifically wanted the numpy scalar type, use `np.bool_` her
e.
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/de
vdocs/release/1.20.0-notes.html#deprecations (https://numpy.org/devdocs/relea
se/1.20.0-notes.html#deprecations)
  mask = np.zeros_like(corr_matrix, dtype=np.bool)
```

In [21]:
```
f, ax = plt.subplots(figsize=(11, 15))

heatmap = sb.heatmap(corr_matrix,
                     mask = mask,
                     square = True,
                     linewidths = .5,
                     cmap = "coolwarm",
                     cbar_kws = {'shrink': .4,
                                 "ticks" : [-1, -.5, 0, 0.5, 1]},
                     vmin = -1,
                     vmax = 1,
                     annot = True)

#add the column names as labels
ax.set_yticklabels(corr_matrix.columns, rotation = 0)
ax.set_xticklabels(corr_matrix.columns)

sb.set_style({'xtick.bottom': True}, {'ytick.left': True})
```



In [14]:
```
import matplotlib
from matplotlib import pyplot
pyplot.figure(figsize=(14,6))
pyplot.hist(df['2001 - LIT'])
pyplot.show
```

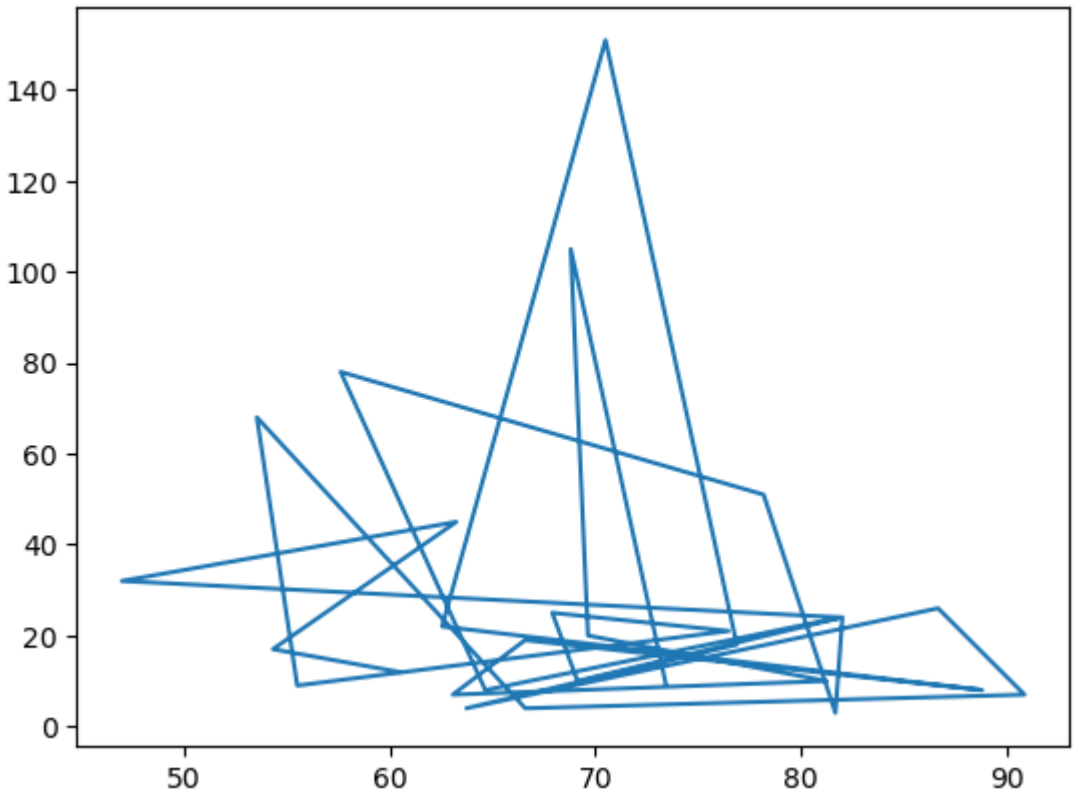
Out[14]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



In [18]:
```
from scipy.stats import shapiro
datatotest=df['2001 - LIT']
stat,p= shapiro(datatotest)
print('stat=%.2f,   p=%.30f'%(stat,p))
if p>0.05:
    print("normal distribution")
else:
    print("not a normal distribution")
```

```
stat=0.98,   p=0.789160549640655517578125000000
normal distribution
```

In [21]:
```
FirstSample=df[1:30]['2001 - LIT']
SecondSample=df[1:30]['2011 -UNEMP']
pyplot.plot(FirstSample,SecondSample)
pyplot.show()
# variables are not dependent on each other
```



In [22]:
```
## spearman rank correlation
from scipy.stats import spearmanr
stat,p=spearmanr(FirstSample,SecondSample)
print('stat=%.3f,p=%5f'%(stat,p))
if p > 0.05:
    print("independent samples")
else:
    print("dependent samples")
```

```
stat=-0.145,p=0.451619
independent samples
```

In [23]:
```
#pearson correlation
from scipy.stats import pearsonr
stat,p= pearsonr(FirstSample,SecondSample)

print('stat%.3f,p=%5f'%(stat,p))
if p>0.05:
    print("independent samples")
else:
    print("dependent samples")
```

```
stat-0.166,p=0.389130
independent samples
```

In [25]:
```
df[1:30].corr(method="pearson")
```

Out[25]:

	2000-01-INC	2011-12-INC	2001 - LIT	2011- LIT	2001 - POP	2011- POP	2001 - SEX_Ratio	2011 SEX_Ratio
2000-01-INC	1.000000	0.863177	0.331996	0.424731	-0.347981	-0.355161	-0.360909	-0.24960
2011-12-INC	0.863177	1.000000	0.300974	0.394941	-0.322241	-0.330096	-0.292710	-0.21832
2001 - LIT	0.331996	0.300974	1.000000	0.959647	-0.124461	-0.124891	-0.150699	-0.15443
2011- LIT	0.424731	0.394941	0.959647	1.000000	-0.156413	-0.155082	-0.231322	-0.22787
2001 - POP	-0.347981	-0.322241	-0.124461	-0.156413	1.000000	0.999031	0.048042	-0.04894
2011- POP	-0.355161	-0.330096	-0.124891	-0.155082	0.999031	1.000000	0.031848	-0.06722
2001 - SEX_Ratio	-0.360909	-0.292710	-0.150699	-0.231322	0.048042	0.031848	1.000000	0.96921
2011 - SEX_Ratio	-0.249609	-0.218327	-0.154439	-0.227872	-0.048941	-0.067227	0.969210	1.00000
2001 - UNEMP	0.418749	0.563748	-0.035339	0.065871	-0.229627	-0.238139	0.300535	0.36985
2011 - UNEMP	0.123810	0.135499	-0.166112	-0.056741	-0.298471	-0.299997	-0.077785	-0.00056
2001 - Poverty	-0.672840	-0.566588	-0.182454	-0.240778	0.186978	0.198303	0.213814	0.19297
2011 - Poverty	-0.536252	-0.596374	-0.137095	-0.189107	0.199560	0.217152	0.007260	-0.01691

In [8]:
```
!pip install factor_analyzer
import pandas as pd
from sklearn.datasets import load_iris
from factor_analyzer import FactorAnalyzer
import matplotlib.pyplot as plt
```

```
Collecting factor_analyzer
  Downloading factor_analyzer-0.4.1.tar.gz (41 kB)
     ---------------------------------------- 41.8/41.8 kB 509.2 kB/s eta 0:00:
00
  Installing build dependencies: started
  Installing build dependencies: finished with status 'done'
  Getting requirements to build wheel: started
  Getting requirements to build wheel: finished with status 'done'
  Preparing metadata (pyproject.toml): started
  Preparing metadata (pyproject.toml): finished with status 'done'
Requirement already satisfied: numpy in c:\users\dell\anaconda3\lib\site-pack
ages (from factor_analyzer) (1.21.5)
Requirement already satisfied: scikit-learn in c:\users\dell\anaconda3\lib\si
te-packages (from factor_analyzer) (1.0.2)
Requirement already satisfied: scipy in c:\users\dell\anaconda3\lib\site-pack
ages (from factor_analyzer) (1.9.1)
Requirement already satisfied: pandas in c:\users\dell\anaconda3\lib\site-pac
kages (from factor_analyzer) (1.4.4)
Collecting pre-commit
  Downloading pre_commit-3.2.1-py2.py3-none-any.whl (202 kB)
     ---------------------------------------- 202.7/202.7 kB 2.0 MB/s eta 0:00:
00
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\dell\anacon
da3\lib\site-packages (from pandas->factor_analyzer) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\dell\anaconda3\lib\si
te-packages (from pandas->factor_analyzer) (2022.1)
Collecting nodeenv>=0.11.1
  Downloading nodeenv-1.7.0-py2.py3-none-any.whl (21 kB)
Collecting cfgv>=2.0.0
  Downloading cfgv-3.3.1-py2.py3-none-any.whl (7.3 kB)
Requirement already satisfied: pyyaml>=5.1 in c:\users\dell\anaconda3\lib\sit
e-packages (from pre-commit->factor_analyzer) (6.0)
Collecting identify>=1.0.0
  Downloading identify-2.5.22-py2.py3-none-any.whl (98 kB)
     ---------------------------------------- 98.8/98.8 kB 5.9 MB/s eta 0:00:
00
Collecting virtualenv>=20.10.0
  Downloading virtualenv-20.21.0-py3-none-any.whl (8.7 MB)
     ---------------------------------------- 8.7/8.7 MB 9.6 MB/s eta 0:00:00
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\dell\anaconda
3\lib\site-packages (from scikit-learn->factor_analyzer) (2.2.0)
Requirement already satisfied: joblib>=0.11 in c:\users\dell\anaconda3\lib\si
te-packages (from scikit-learn->factor_analyzer) (1.1.0)
Requirement already satisfied: setuptools in c:\users\dell\anaconda3\lib\site
-packages (from nodeenv>=0.11.1->pre-commit->factor_analyzer) (63.4.1)
Requirement already satisfied: six>=1.5 in c:\users\dell\anaconda3\lib\site-p
ackages (from python-dateutil>=2.8.1->pandas->factor_analyzer) (1.16.0)
Requirement already satisfied: filelock<4,>=3.4.1 in c:\users\dell\anaconda3
\lib\site-packages (from virtualenv>=20.10.0->pre-commit->factor_analyzer)
(3.6.0)
Collecting distlib<1,>=0.3.6
  Downloading distlib-0.3.6-py2.py3-none-any.whl (468 kB)
     ---------------------------------------- 468.5/468.5 kB 7.4 MB/s eta 0:00:
00
Requirement already satisfied: platformdirs<4,>=2.4 in c:\users\dell\anaconda
3\lib\site-packages (from virtualenv>=20.10.0->pre-commit->factor_analyzer)
(2.5.2)
```

```
Building wheels for collected packages: factor_analyzer
  Building wheel for factor_analyzer (pyproject.toml): started
  Building wheel for factor_analyzer (pyproject.toml): finished with status
'done'
  Created wheel for factor_analyzer: filename=factor_analyzer-0.4.1-py2.py3-n
one-any.whl size=42071 sha256=07e0f866b1507625af88efeb3de2fd32e167e8cbd706936
f9ac045f6942a2532
  Stored in directory: c:\users\dell\appdata\local\pip\cache\wheels\6d\32\bd
\460a71becd83f7d77152f437c2fd451f5c87bc19cfcdbfcd24
Successfully built factor_analyzer
Installing collected packages: distlib, virtualenv, nodeenv, identify, cfgv,
pre-commit, factor_analyzer
Successfully installed cfgv-3.3.1 distlib-0.3.6 factor_analyzer-0.4.1 identif
y-2.5.22 nodeenv-1.7.0 pre-commit-3.2.1 virtualenv-20.21.0
```

In [14]:
```
from factor_analyzer.factor_analyzer import calculate_bartlett_sphericity
chi_square_value,p_value=calculate_bartlett_sphericity(df)
chi_square_value, p_value
```

```
C:\Users\Dell\anaconda3\lib\site-packages\numpy\core\fromnumeric.py:3438: Fut
ureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numer
ic_only=None') is deprecated; in a future version this will raise TypeError.
Select only valid columns before calling the reduction.
  return mean(axis=axis, dtype=dtype, out=out, **kwargs)
C:\Users\Dell\anaconda3\lib\site-packages\numpy\core\fromnumeric.py:3579: Fut
ureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numer
ic_only=None') is deprecated; in a future version this will raise TypeError.
Select only valid columns before calling the reduction.
  return std(axis=axis, dtype=dtype, out=out, ddof=ddof, **kwargs)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_4156\3468252620.py in <module>
      1 from factor_analyzer.factor_analyzer import calculate_bartlett_spheri
city
----> 2 chi_square_value,p_value=calculate_bartlett_sphericity(df)
      3 chi_square_value, p_value

~\anaconda3\lib\site-packages\factor_analyzer\factor_analyzer.py in calculate
_bartlett_sphericity(x)
    103     """
    104     n, p = x.shape
--> 105     x_corr = corr(x)
    106
    107     corr_det = np.linalg.det(x_corr)

~\anaconda3\lib\site-packages\factor_analyzer\utils.py in corr(x)
     89     """
     90     x = (x - np.mean(x, axis=0)) / np.std(x, axis=0, ddof=0)
---> 91     r = cov(x)
     92     return r
     93

~\anaconda3\lib\site-packages\factor_analyzer\utils.py in cov(x, ddof)
     68         The covariance matrix of the variables.
     69     """
---> 70     r = np.cov(x, rowvar=False, ddof=ddof)
     71     return r
     72

<__array_function__ internals> in cov(*args, **kwargs)

~\anaconda3\lib\site-packages\numpy\lib\function_base.py in cov(m, y, rowvar,
bias, ddof, fweights, aweights, dtype)
   2516             w *= aweights
   2517
-> 2518     avg, w_sum = average(X, axis=1, weights=w, returned=True)
   2519     w_sum = w_sum[0]
   2520

<__array_function__ internals> in average(*args, **kwargs)

~\anaconda3\lib\site-packages\numpy\lib\function_base.py in average(a, axis,
weights, returned)
    413
    414     if returned:
--> 415         if scl.shape != avg.shape:
    416             scl = np.broadcast_to(scl, avg.shape).copy()
    417         return avg, scl

AttributeError: 'float' object has no attribute 'shape'
```

In [60]:
```
df.columns
```

Out[60]:
```
Index(['States_Union Territories', '2000-01-INC', '2011-12-INC', '2001 - LI
T',
       '2011- LIT', '2001 - POP', '2011- POP', '2001 -SEX_Ratio',
       '2011 -SEX_Ratio', '2001 -UNEMP', '2011 -UNEMP', '2001 -Poverty',
       '2011 -Poverty', 'region'],
      dtype='object')
```

In [18]:
```
df1=df.drop(['States_Union Territories'],axis=1)
```

In [19]:
```
from factor_analyzer.factor_analyzer import calculate_bartlett_sphericity
chi_square_value,p_value=calculate_bartlett_sphericity(df1)
chi_square_value, p_value
```

Out[19]:
```
(451.1128801259974, 9.821087334114737e-59)
```

In [25]:
```
from factor_analyzer.factor_analyzer import calculate_kmo
kmo_all,kmo_model=calculate_kmo(df1)
kmo_model
## if kmo value is greater than 0.8,then factor analysis is used(most preferabl
```

Out[25]:
```
0.5930413385639766
```

In [37]:
```
!pip install scikit-learn
```

```
Requirement already satisfied: scikit-learn in c:\users\dell\anaconda3\lib\si
te-packages (1.0.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\dell\anaconda
3\lib\site-packages (from scikit-learn) (2.2.0)
Requirement already satisfied: scipy>=1.1.0 in c:\users\dell\anaconda3\lib\si
te-packages (from scikit-learn) (1.9.1)
Requirement already satisfied: numpy>=1.14.6 in c:\users\dell\anaconda3\lib\s
ite-packages (from scikit-learn) (1.21.5)
Requirement already satisfied: joblib>=0.11 in c:\users\dell\anaconda3\lib\si
te-packages (from scikit-learn) (1.1.0)
```

In [16]:
```
# check scikit-learn version
import sklearn
print(sklearn.__version__)
```

```
1.0.2
```

In [17]:
```
df["region"].value_counts()
```

Out[17]:
```
EAST       12
NORTH       8
SOUTH       5
WEST        3
UT          2
CENTRAL     2
Name: region, dtype: int64
```

In [18]:
```
ut1=df.groupby(["region"])
ut1
```

Out[18]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000002396A2AAD00>
```

In [19]:
```
for region ,region_df in ut1:
    print(region)
    print(region_df)
```

```
CENTRAL
   States_Union Territories  2000-01-INC  2011-12-INC  2001 - LIT  2011- LI
T  \
6               Chhattisgarh        10744        48366       64.66      70.2
8
16           Madhya Pradesh        11862        37180       66.60      75.3
7

    2001 - POP  2011- POP  2001 -SEX_Ratio  2011 -SEX_Ratio  2001 -UNEMP  \
6        20834      25545              989              991           16
16       60348      72627              919              931            5

    2011 -UNEMP  2001 -Poverty  2011 -Poverty   region
6             8          30.00          39.93  CENTRAL
16            4          37.43          31.65  CENTRAL
EAST
   States_Union Territories  2000-01-INC  2011-12-INC  2001 - LIT  2011- LI
T  \
2          Arunachal Pradesh        15260        71366       54.34      65.3
0
```

In [20]:
```
north=ut1.get_group("NORTH")
north
```

Out[20]:

	States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011- LIT	2001 - POP	2011- POP	2001 - SEX_Ratio	2011 - SEX_Ratio	2001 - UNEMP
7	Delhi	40678	161446	57.63	76.24	13851	16788	821	868	47
10	Haryana	25583	106320	82.01	88.70	21145	25351	861	879	8
11	Himachal Pradesh	22795	75185	69.14	78.03	6078	6865	968	972	12
12	Jammu and Kashmir	14268	46734	67.91	75.55	10144	12541	892	889	11
24	Punjab	27881	76895	66.59	79.60	24359	27743	876	895	18
25	Rajasthan	13020	54637	63.08	72.89	56507	68548	921	928	4
29	Uttar Pradesh	9828	30021	73.45	80.09	166198	199812	898	912	8
30	Uttarakhand	15285	85372	73.19	87.22	8489	10086	962	963	22

In [21]:
```
corr_matrix1 = north.corr()
```

In [22]:
```
print(corr_matrix1)
```

```
                 2000-01-INC  2011-12-INC  2001 - LIT  2011- LIT  2001 - POP
\
2000-01-INC         1.000000     0.911688   -0.381534   0.010872   -0.500094
2011-12-INC         0.911688     1.000000   -0.281346   0.186350   -0.538907
2001 - LIT         -0.381534    -0.281346    1.000000   0.820598    0.162061
2011- LIT           0.010872     0.186350    0.820598   1.000000   -0.093186
2001 - POP         -0.500094    -0.538907    0.162061  -0.093186    1.000000
2011- POP          -0.502605    -0.538682    0.160853  -0.095648    0.999951
2001 -SEX_Ratio    -0.627099    -0.531771    0.232080   0.086311   -0.039942
2011 -SEX_Ratio    -0.473189    -0.370267    0.133454   0.084424   -0.043826
2001 -UNEMP         0.782989     0.826497   -0.562335  -0.053486   -0.346827
2011 -UNEMP         0.794854     0.864909   -0.483277  -0.056447   -0.356759
2001 -Poverty      -0.502778    -0.452291    0.187208   0.004967    0.961902
2011 -Poverty      -0.561763    -0.524740    0.224074  -0.016145    0.977891

                 2011- POP  2001 -SEX_Ratio  2011 -SEX_Ratio  2001 -UNEMP  \
2000-01-INC      -0.502605        -0.627099        -0.473189     0.782989
2011-12-INC      -0.538682        -0.531771        -0.370267     0.826497
2001 - LIT        0.160853         0.232080         0.133454    -0.562335
2011- LIT        -0.095648         0.086311         0.084424    -0.053486
2001 - POP        0.999951        -0.039942        -0.043826    -0.346827
2011- POP         1.000000        -0.039811        -0.044147    -0.347238
2001 -SEX_Ratio  -0.039811         1.000000         0.969118    -0.478248
2011 -SEX_Ratio  -0.044147         0.969118         1.000000    -0.312164
2001 -UNEMP      -0.347238        -0.478248        -0.312164     1.000000
2011 -UNEMP      -0.355056        -0.675548        -0.551802     0.933413
2001 -Poverty     0.962684         0.091614         0.125153    -0.291250
2011 -Poverty     0.979266         0.004094        -0.009851    -0.327530

                 2011 -UNEMP  2001 -Poverty  2011 -Poverty
2000-01-INC         0.794854      -0.502778      -0.561763
2011-12-INC         0.864909      -0.452291      -0.524740
2001 - LIT         -0.483277       0.187208       0.224074
2011- LIT          -0.056447       0.004967      -0.016145
2001 - POP         -0.356759       0.961902       0.977891
2011- POP          -0.355056       0.962684       0.979266
2001 -SEX_Ratio    -0.675548       0.091614       0.004094
2011 -SEX_Ratio    -0.551802       0.125153      -0.009851
2001 -UNEMP         0.933413      -0.291250      -0.327530
2011 -UNEMP         1.000000      -0.333866      -0.315995
2001 -Poverty      -0.333866       1.000000       0.968620
2011 -Poverty      -0.315995       0.968620       1.000000
```

In [23]:
```
north.min()
```

Out[23]:
```
States_Union Territories    Delhi
2000-01-INC                  9828
2011-12-INC                 30021
2001 - LIT                  57.63
2011- LIT                   72.89
2001 - POP                   6078
2011- POP                    6865
2001 -SEX_Ratio               821
2011 -SEX_Ratio               868
2001 -UNEMP                     4
2011 -UNEMP                     7
2001 -Poverty                3.48
2011 -Poverty                8.06
region                      NORTH
dtype: object
```

In [73]:
```
north.max()
```

Out[73]:
```
States_Union Territories    Uttarakhand
2000-01-INC                       40678
2011-12-INC                      161446
2001 - LIT                        82.01
2011- LIT                          88.7
2001 - POP                       166198
2011- POP                        199812
2001 -SEX_Ratio                     968
2011 -SEX_Ratio                     972
2001 -UNEMP                          47
2011 -UNEMP                          78
2001 -Poverty                     31.15
2011 -Poverty                     29.43
region                           NORTH
dtype: object
```

In [24]:
```
ut1.max()
```

Out[24]:

region	States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011- LIT	2001 - POP	2011- POP	2001 - SEX_Ratio	2011 - SEX_Ratio	U
CENTRAL	Madhya Pradesh	11862	48366	66.60	75.37	60348	72627	989	991	
EAST	West Bengal	17826	130127	86.66	91.85	82999	104099	978	992	
NORTH	Uttarakhand	40678	161446	82.01	88.70	166198	199812	968	972	
SOUTH	Tamil Nadu	35994	103149	88.80	91.33	76210	84581	1058	1084	
UT	Chandigarh	49771	136883	81.94	86.63	901	1055	846	876	
WEST	Maharashtra	43735	211570	90.86	94.00	96879	112374	961	973	

In [25]:
```
ut1.min()
```

Out[25]:

region	States_Union Territories	2000-01-INC	2011-12-INC	2001 - LIT	2011-LIT	2001 - POP	2011-POP	2001 - SEX_Ratio	2011 - SEX_Ratio	200 UNEM
CENTRAL	Chhattisgarh	10744	37180	64.66	70.28	20834	25545	919	931	
EAST	Arunachal Pradesh	6415	22582	47.00	61.80	541	611	875	890	
NORTH	Delhi	9828	30021	57.63	72.89	6078	6865	821	868	
SOUTH	Andhra Pradesh	17195	64773	53.56	66.41	974	1248	965	973	
UT	Andaman and Nicobar Islands	25047	89642	81.30	86.05	356	381	777	818	
WEST	Goa	18392	85979	78.18	86.21	1348	1459	920	919	

In [27]:
```
ut1.describe()
```

Out[27]:

	2000-01-INC								
region	count	mean	std	min	25%	50%	75%	max	count
CENTRAL	2.0	11303.000000	790.545381	10744.0	11023.5	11303.0	11582.5	11862.0	2.0
EAST	12.0	13835.333333	3387.217800	6415.0	11890.0	15458.5	16121.0	17826.0	12.0
NORTH	8.0	21167.250000	10177.348879	9828.0	13956.0	19040.0	26157.5	40678.0	8.0
SOUTH	5.0	22519.800000	7675.041642	17195.0	18344.0	20094.0	20972.0	35994.0	5.0
UT	2.0	37409.000000	17482.508058	25047.0	31228.0	37409.0	43590.0	49771.0	2.0
WEST	3.0	28301.333333	13544.578485	18392.0	20584.5	22777.0	33256.0	43735.0	3.0

6 rows × 96 columns

In [ ]:
```
import plotly.express as px

medals_by_country = df.groupby(['region', 'Medal'])['Medal'].agg('count').unsta

medals_by_country['Total'] = medals_by_country.sum(axis=1)
medals_by_country = medals_by_country.reset_index()

medals_by_country['text'] = medals_by_country['NOC'] + '<br>' + \
    'Gold: ' + medals_by_country['Gold'].astype(str) + '<br>' + \
    'Silver: ' + medals_by_country['Silver'].astype(str) + '<br>' + \
    'Bronze: ' + medals_by_country['Bronze'].astype(str) + '<br>' + \
    'Total: ' + medals_by_country['Total'].astype(str)

fig = px.choropleth(medals_by_country, locations='NOC',
                    color='Total',
                    color_continuous_scale='Reds',
                    hover_name='text', projection="natural earth",
                    title="Total Medals by Country")
fig.show()
```