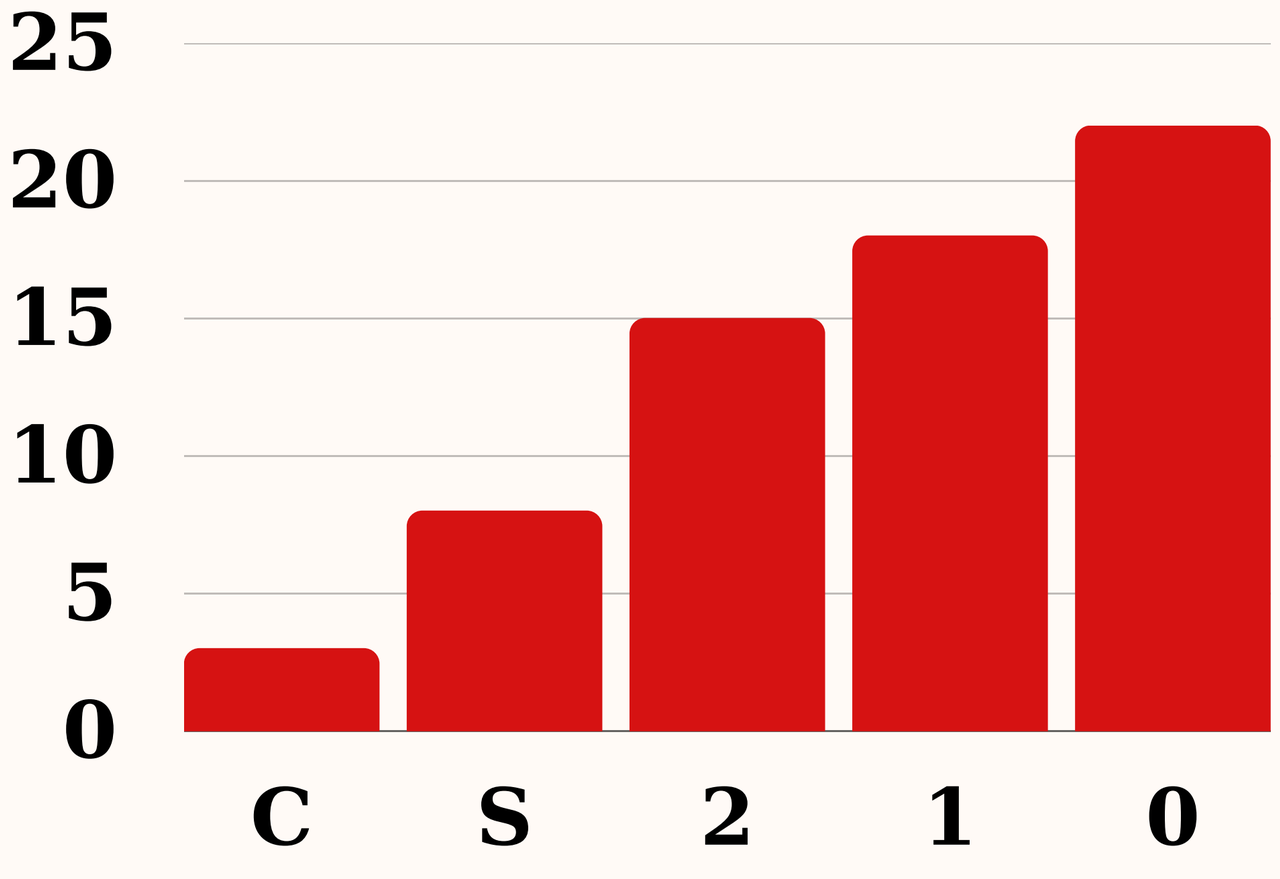
CS210 Project

B4 Swing_

- Kadir Şen
- Enes Özkan Şen
- Mehmet Berke Bakan
- Tahsin Mert Ulu
- Mükremin Berkay Can
- Mervan Yusuf Gür



OUTLİNE

- INTRODUCTION
- LITERATURE REVİEW
- HYPOTHESIS
- DATA GATHERING
- DATA CLEANING
- DATA ANALYSIS
- PREDICTION
- CONCLUSİON

INTRODUCTION

Predicting Changes in unemployment: An analysis of socio-economic factors

- **Analyzing sectorel changes, GDP, and their impact on agricultural and urban lands**
- **Examining relationships between GDP, informal employment , urban-rural population ratios.**
- **ML techniques used for data completion and prediction**

LITERATURE REVIEW

Şahin ve Yiğit, 2016

- **"Migration has a long-term impact on the labor market. A unit change in migration leads to a 0.28-unit change in unemployment**

LITERATURE REVIEW

Bahar ve Bingöl, 2010

- **"Rural to urban migration increases the urban labor force and creates employment pressure. Unemployment rates rise in both destination and source cities."**

HYPOTHESIS

"Demographic changes in Turkey have influenced unemployment rates"

Gathering Data

	A	B	C	D	E	F	G	H	I	J
1		Rural popı	Urban popı	Unemploy	GDP (currı	Unemploy	Populatio	Tarim (%)	Sanayi (%	netlı
2	1990	0,395	0, 59203		1,51E+14		54324142	47,5	20,2	32,
3	1991	0,389	0, 59976		1,50E+14	0,821	55321172	44	21,2	32,8
4	1992	0,384	0, 60518		1,58E+14	0,851	56302037	41,2	22,4	36,4
5	1993	0,379	0, 61055		1,80E+14	0,896	57296008	42,1	21,8	
6	1994	0,373	0, 6159		1,31E+14	0,858	58310245	43,6	21,7	34,3
7	1995	0,368	0, 62123		1,69E+14	0,764	59305490	44,2	20,8	33,2
8	1996	0,363	0, 62653		1,81E+14	0,663	60293786	44,3	19,9	32,1
9	1997	0,358	0, 63179		1,90E+14	0,684	61277426	39,6	22,4	34,4
10	1998	0,353	0, 63703		2,76E+14	0,689	62242204	41,9	21,1	33,4
11	1999	0,347	0, 64223		2,56E+14	0,769	63185615	36,1	23,8	36,2
12	2000	0,34	0, 64741		2,74E+14	0,65	64113547	35,8	25,8	39,4
13	2001	0,334	0, 6534		2,02E+14	0,838	65072018	37,2	25,8	38,9
14	2002	0,328	0, 65974		2,40E+14	0,1036	65988663	34,6	28,4	41,9
15	2003	0,322	0, 66602		3,15E+14	0,1054	66867327	34,2	28,8	42,2
16	2004	0,316	0, 67225	0,1222	4,09E+14	0,1084	67785075	24,4	26,2	49
17	2005	0,309	0, 6784	0,1003	5,06E+14	0,1064	68704715	21,5	27	51
18	2006	0,303	0, 6845	0,813	5,57E+14	0,872	69601333	21,2	28,3	51,5
19	2007	0,298	0, 69053	0,849	6,81E+14	0,887	70468869	19,4	29,4	51,8
20	2008	0,292	0, 69651	0,916	7,70E+14	0,971	71320726	20,7	28,7	52,6
21	2009	0,286	0, 70241	0,1082	6,49E+14	0,1255	72225639	22,3	27,4	50,7
22	2010	0,28	0, 70825	0,978	7,77E+14	0,1066	73195345	21,9	28,1	50,2
23	2011	0,275	0, 71402	0,937	8,39E+14	0,88	74173854	20,5	28,8	52,1
24	2012	0,269	0, 71974	0,893	8,81E+14	0,815	75277439	19,7	29,3	52,4
25	2013	0,264	0, 72531	0,925	9,58E+14	0,873	76576117	19,1	30,3	52,1
26	2014	0,259	0, 73077	0,1049	9,39E+14	0,988	78112073	18,8	32,9	53,7
27	2015	0,254	0, 73611	0,1078	8,64E+14	0,1024	79646178	18,3	34,4	55
28	2016	0,249	0, 74134	0,1183	8,70E+14	0,1084	81019394	18,6	34	55,1
29	2017	0,244	0, 74644	0,1241	8,59E+14	0,1082	82089826	17,7	34,5	55,2
30	2018	0,239	0, 75143	0,1219	7,78E+14	0,1089	82809304	17,1	37,3	57,6
31	2019	0,234	0, 7563	0,135	7,60E+14	0,1367	83481684	15,6	37	58,3
32	2020	0,239	0, 76105	0,1263	7,20E+14	0,1311	84135428	16	37,6	57,6
33	2021	0,234	0, 76569	0,1245	8,19E+14	0,1198	84775404	15,4	38,5	58,1
34	2022					0,1003		15,2	26,8	58

P	Q	R	S	T	U	V	W	X	Y	Z
Kayit disi	onlisans	lisans	yükseklisa	Land area	Forest are	Forest are	Agricultur	Agricultural land (% of land area)		
55,6	70801	624909	25006	769630	2,57E+14	19783480	396770	5,16E+14		
51,9	75828	673002	28742	769630	2,58E+14	19819967	400670	5,21E+14		
50,1	93686	786174	33442	769630	2,58E+14	19856454	399040	5,18E+14		
48,6	105063	967249	35797	769630	2,58E+14	19892941	399130	5,19E+14		
47,1	125878	970192	41012	769630	2,59E+14	19929428	401490	5,22E+14		
50,9	148993	1001146	49893	769630	2,59E+14	19965915	394930	5,13E+14		
54,7	168134	1065031	51320	769630	2,60E+14	20002402	393640	5,11E+14		
53,7	192326	1130019	49123	769630	2,60E+14	20038889	392420	5,10E+14		
52,4	202723	1171734	50979	769630	2,61E+14	20075376	399190	5,19E+14		
53,6	217758	1194490	53547	769630	2,61E+14	20111863	403020	5,24E+14		
53,6	239271	1260960	65068	769630	2,62E+14	20148350	404790	5,26E+14		
55,8	262649	1297389	73466	769630	2,63E+14	20241823	409680	5,32E+14		
52,1	323971	1455760	79811	769630	2,64E+14	20335296	411960	5,35E+14		
51,7	344984	1476010	90057	769630	2,65E+14	20428769	406440	5,28E+14		
48,1	384456	1558539	92566	769630	2,67E+14	20522242	412100	5,35E+14		
49,5	441014	1714090	111814	769630	2,68E+14	20615715	412230	5,36E+14		
48,5	482208	1782728	108683	769630	2,69E+14	20709188	404930	5,26E+14		
48,7	502622	1843265	104028	769630	2,70E+14	20802661	395050	5,13E+14		
43,5	548695	2183133	109281	769630	2,72E+14	20896134	391220	5,08E+14		
43,84	593955	2702418	139463	769630	2,73E+14	20989607	389110	5,06E+14		
43,25	595052	3004532	125690	769630	2,74E+14	21083080	390120	5,07E+14		
42,05	662259	3421667	168156	769630	2,75E+14	21192524	382470	4,97E+14		
39,02	755789	3890800	217588	769630	2,77E+14	21301968	384070	4,99E+14		
36,75	1802972	3405340	332470	769630	2,78E+14	21411412	384230	4,99E+14		
33,57	2013762	3628800	342101	769630	2,80E+14	21520856	385540	5,01E+14		
33,49	2285406	3900601	417084	769630	2,81E+14	21630300	385510	5,01E+14		
33,97	2555926	4071579	480215	769630	2,81E+14	21630300	383280	4,98E+14		
33,42	2768757	4241841	454673	769630	2,83E+14	21752460	379640	4,93E+14		
34,52	2829430	4420699	394174	769630	2,85E+14	21908360	378020	4,91E+14		
30,59	3002964	4538926	297001	769630	2,87E+14	22064360	377160	4,90E+14		
29	3114623	4676657	343569	769630	2,89E+14	22220360	377620	4,91E+14		
	3250101	4579047	358271							
	2647054	3754095	434485							

K	L	M	N	O
Toplam	Tarim	Sanayi	Insaat	Hizmet
76426	18181	16559	4385	37301
78868	18446	17198	4729	38495
80003	18661	17445	4624	39273
80597	17088	17095	4950	41464
81436	18133	16610	4798	41895
87139	20299	17474	4693	44673
92663	20874	19379	4835	47575
95057	21624	19665	4856	48912
98043	21161	20532	7154	49196
101097	20105	21108	6654	53230
105995	20277	21381	6662	57675
107504	20189	21181	6966	59168
108299	21605	21337	7539	57818
110764	21227	21738	7255	60544
111169	20383	22288	6265	62233
106739	18995	23382	5098	59264
115406	18655	23704	5980	67067
122803	18734	24670	5379	74020
123466	16662	25856	5418	75530

DATA CLEANING

```
## Renaming columns
df = df.rename(columns={'Unnamed: 0': 'Year'})
df = df.rename(columns={'Tarim (%)': 'Agriculture (%)'})
df = df.rename(columns={'Sanayi (%)': 'Industry (%)'})
df = df.rename(columns={'Hizmetler (%)': 'Service (%)'})
df = df.rename(columns={'Unemployment with advanced education (% of total labor force with advanced education)': 'Unemployment with advanced education (%)'})
df = df.rename(columns={'informal worker (%)': 'Off-record labour'})
df = df.rename(columns={'Kayit disi': 'Off-record worker'})

### Creating new column
df['num_students'] = df[['onlisans', 'lisans', 'yükseklisans']].sum(axis=1)
```

[6]
```
####
df["Urban population (% of total population)"]=df["Urban population (% of total population)"].str.replace(', ', ',').str.replace(",",".")
df["Urban population (% of total population)"]=df["Urban population (% of total population)"].astype(float)
```

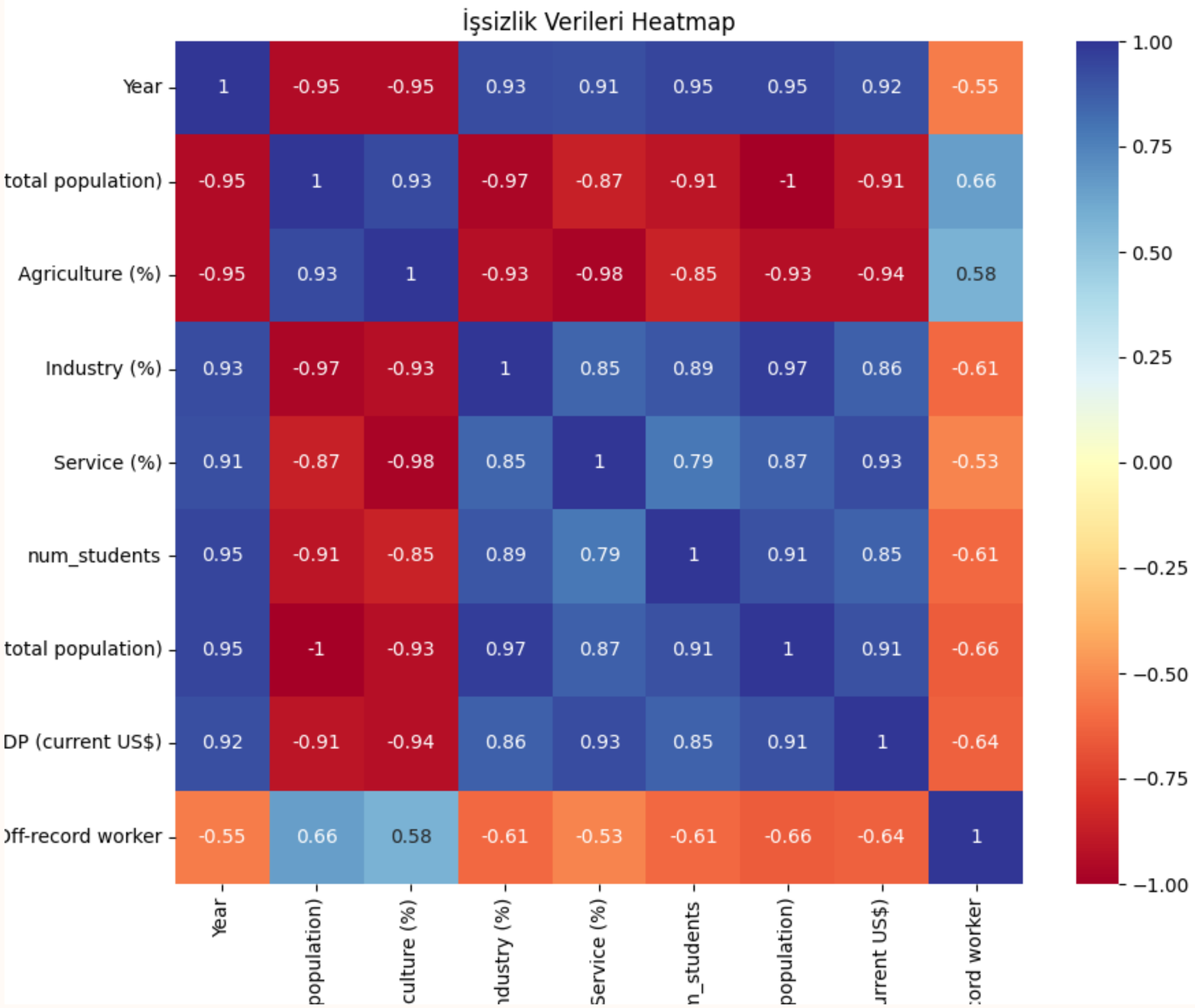
[7]
```
###### handling undesired behaviorul columns
###### sectoral distribition shoudl be out 100. Normalize each column by their weight
imputer = SimpleImputer(strategy='mean')
df[['Agriculture (%)', 'Industry (%)', 'Service (%)']] = imputer.fit_transform(df[['Agriculture (%)', 'Industry (%)', 'Service (%)']])
total = df['Agriculture (%)'] + df['Industry (%)'] + df['Service (%)']
df['Agriculture (%)'] = df['Agriculture (%)'] / total * 100
df['Industry (%)'] = df['Industry (%)'] / total * 100
df['Service (%)'] = df['Service (%)'] / total * 100


df.head()
```

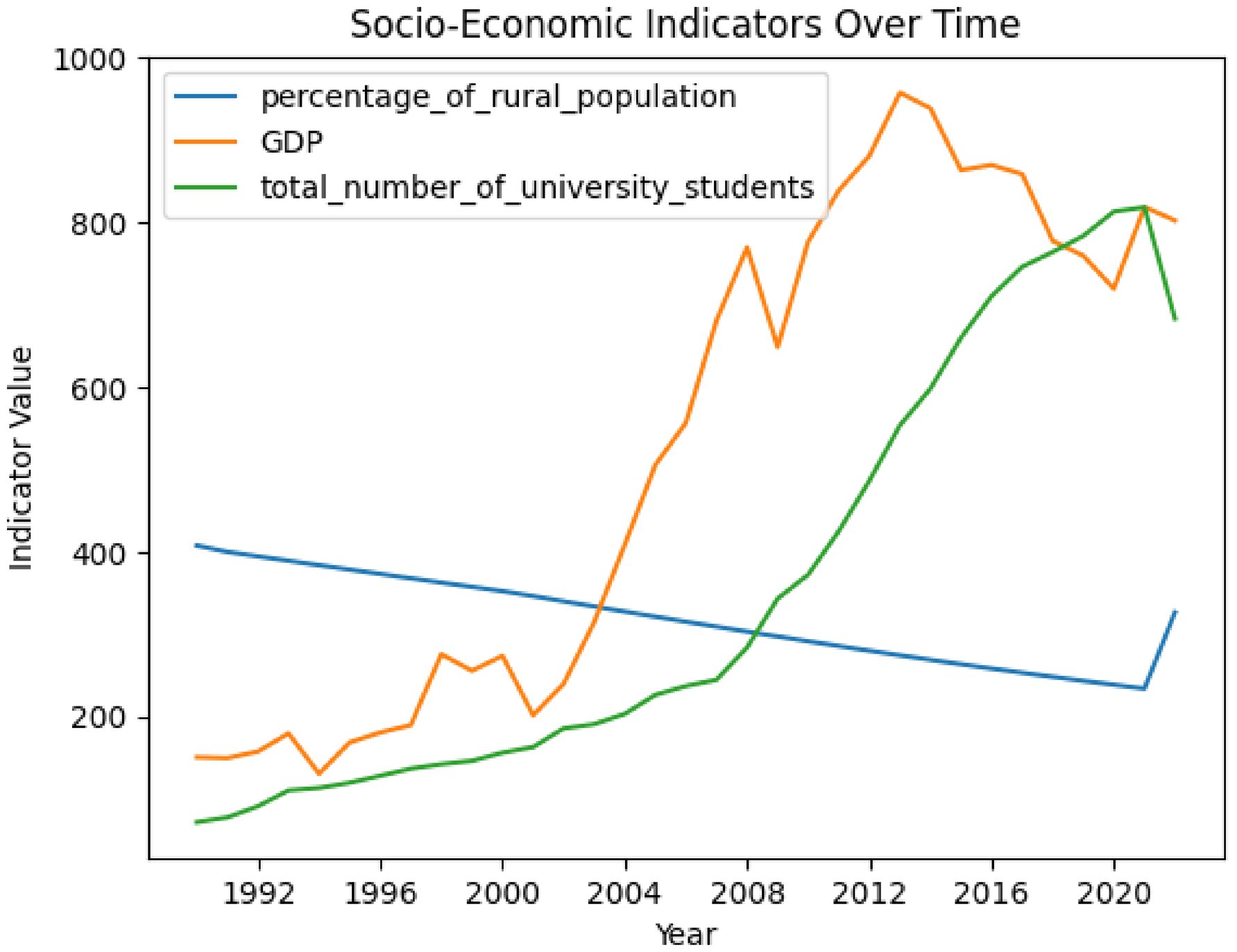
```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import MinMaxScaler
from sklearn.utils import resample
from sklearn.linear_model import LinearRegression
listoflist=[]
selected_features_urban=['Forest area (% of land area)', 'Forest area (sq. km)','Urban population (% of total population)','Rural population (% of total population)','GDP (current US$)','Service (%)', 'Industry (%)','num_students'
selected_features_gdp = ['Population, total', 'Rural population (% of total population)',  'num_students', 'Unemployment with advanced education (%)', 'GDP (current US$)', 'Off-record worker']
selected_features_pop = ['Unemployment with advanced education (%)',"Urban population (% of total population)",'Year','Population, total', 'Rural population (% of total population)', 'Agriculture (%)', 'Industry (%)', 'Service (%)
selected_features_unemp = ['Population, total', 'Rural population (% of total population)',  'Industry (%)', 'Insaat', 'Service (%)', 'num_students', 'Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)', '
selected_features_kayit=['Urban population (% of total population)','GDP (current US$)','Industry (%)', 'Insaat', 'Service (%)','Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)','Population, total','Off
selected_features_land=['Urban population (% of total population)','GDP (current US$)','Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)','Population, total','Off-record worker','Land area (sq. km)','For
selected_features_forest1=['Urban population (% of total population)','GDP (current US$)','Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)','Population, total','Off-record worker','Land area (sq. km)','
selected_features_forest2=['Urban population (% of total population)','GDP (current US$)','Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)','Population, total','Off-record worker','Land area (sq. km)','
selected_features_agricultural1=['Urban population (% of total population)','GDP (current US$)','Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)','Population, total','Off-record worker','Land area (sq.
selected_features_agricultural2=['Urban population (% of total population)','GDP (current US$)','Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)','Population, total','Off-record worker','Land area (sq.
listoflist.append(selected_features_urban)
listoflist.append(selected_features_gdp)
listoflist.append(selected_features_pop)
listoflist.append(selected_features_unemp)
listoflist.append(selected_features_kayit)
listoflist.append(selected_features_land)
listoflist.append(selected_features_forest1)
listoflist.append(selected_features_forest2)
listoflist.append(selected_features_agricultural1)
listoflist.append(selected_features_agricultural2)
```

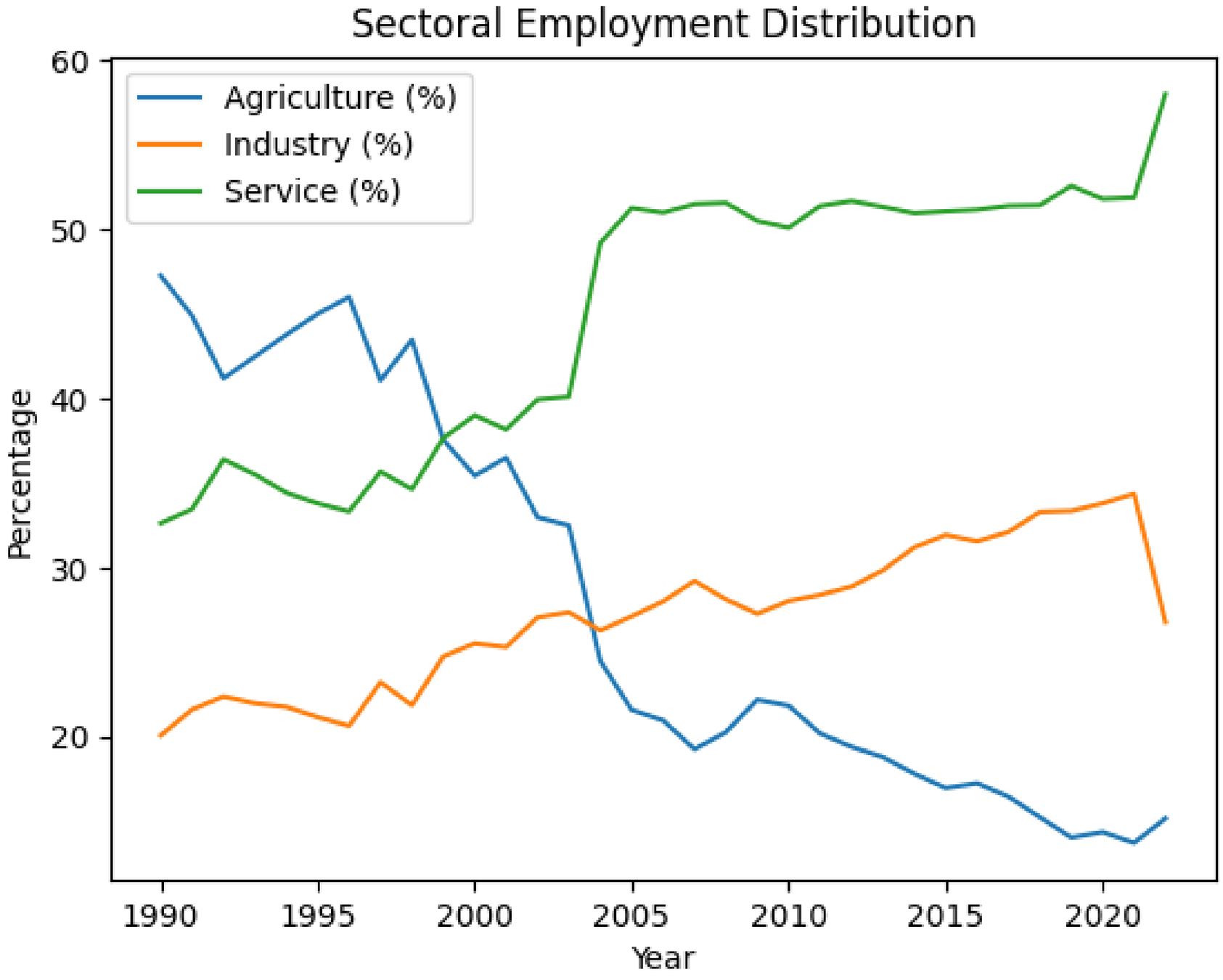
```python
target_variables = [
    'Urban population (% of total population)',
    'GDP (current US$)',
    'Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)',
    'Population, total',
    'Off-record worker',
    'Land area (sq. km)',
    'Forest area (% of land area)', 'Forest area (sq. km)',
    'Agricultural land (sq. km)', 'Agricultural land (% of land area)'
]
selected_features = ['Population, total', 'percentage_of_rural_population',  'Agriculture (%)', 'Industry (%)', 'Service (%)', 'num_students', 'Unemployment, total (out of 10 of total labor force) (modeled ILO esti

target_variable = 'Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)'
for i in range(10):
  data = df[listoflist[i]]
  print(data.columns)
  # Separate data with missing values
  missing_data_mask = data[target_variables[i]].isnull()
  missing_data = data[missing_data_mask]
  not_missing_data = data[~missing_data_mask]

  # Split data into training and test sets
  X_train, X_test, y_train, y_test = train_test_split(not_missing_data.drop(target_variables[i], axis=1), not_missing_data[target_variables[i]], test_size=0.3, random_state=42)

  # Oversample the target row
  oversampled_X_train, oversampled_y_train = resample(X_train, y_train, replace=True, n_samples=len(X_train)*10, random_state=42)

  # Use SimpleImputer to fill missing values
  imputer = SimpleImputer(strategy='mean')
  X_train_imputed = imputer.fit_transform(oversampled_X_train)
  X_test_imputed = imputer.transform(X_test)
  X_filled = imputer.transform(missing_data.drop(target_variables[i], axis=1))

  # LinearRegression model: create and train
  linear_model = LinearRegression()
  linear_model.fit(X_train_imputed, oversampled_y_train)

  # Predict missing values (Linear Regression)
  predicted_values_linear = linear_model.predict(X_filled)

  # Fill in missing values
  missing_data_index = missing_data.index
```

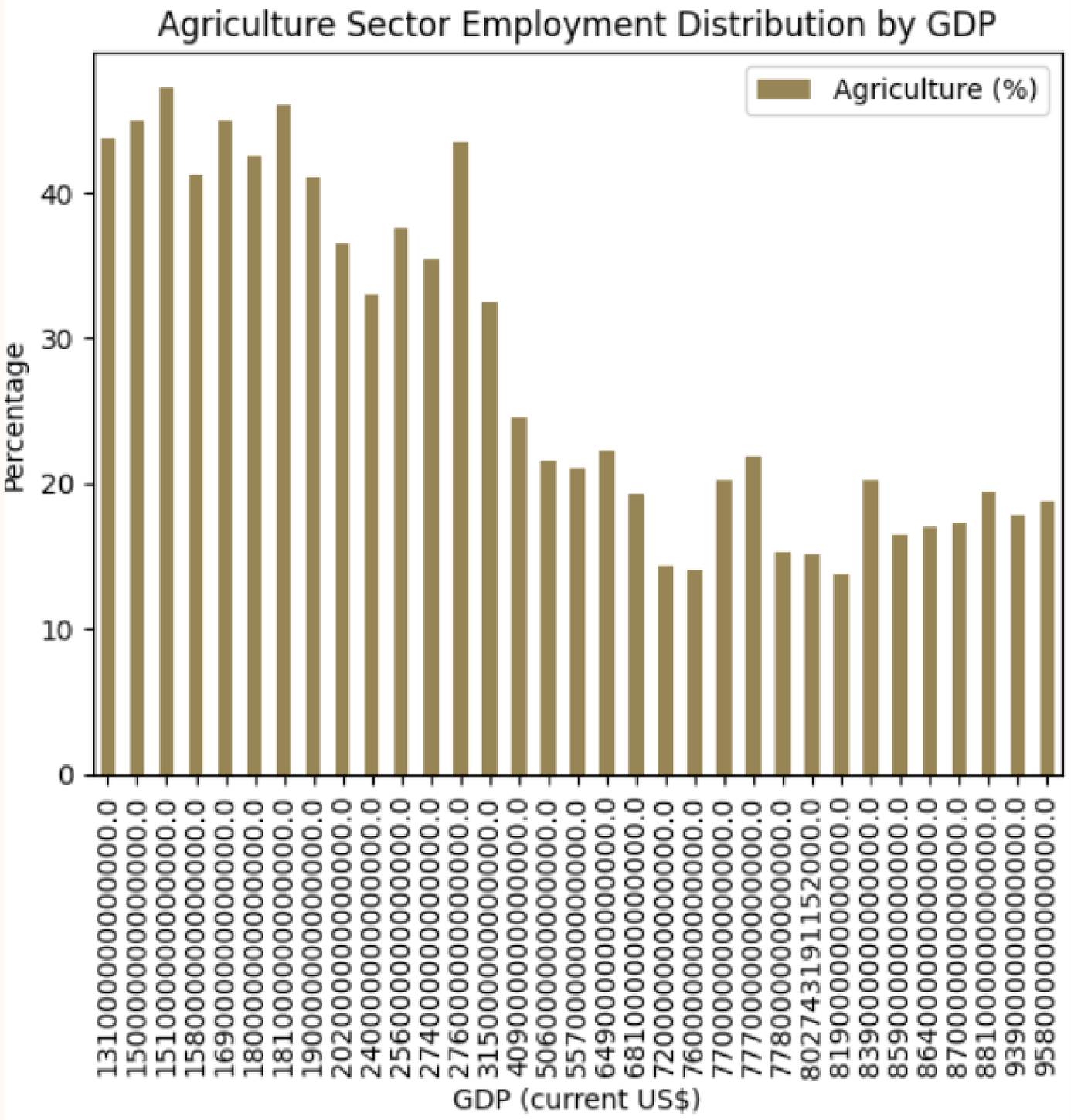
DATA ANALYSIS

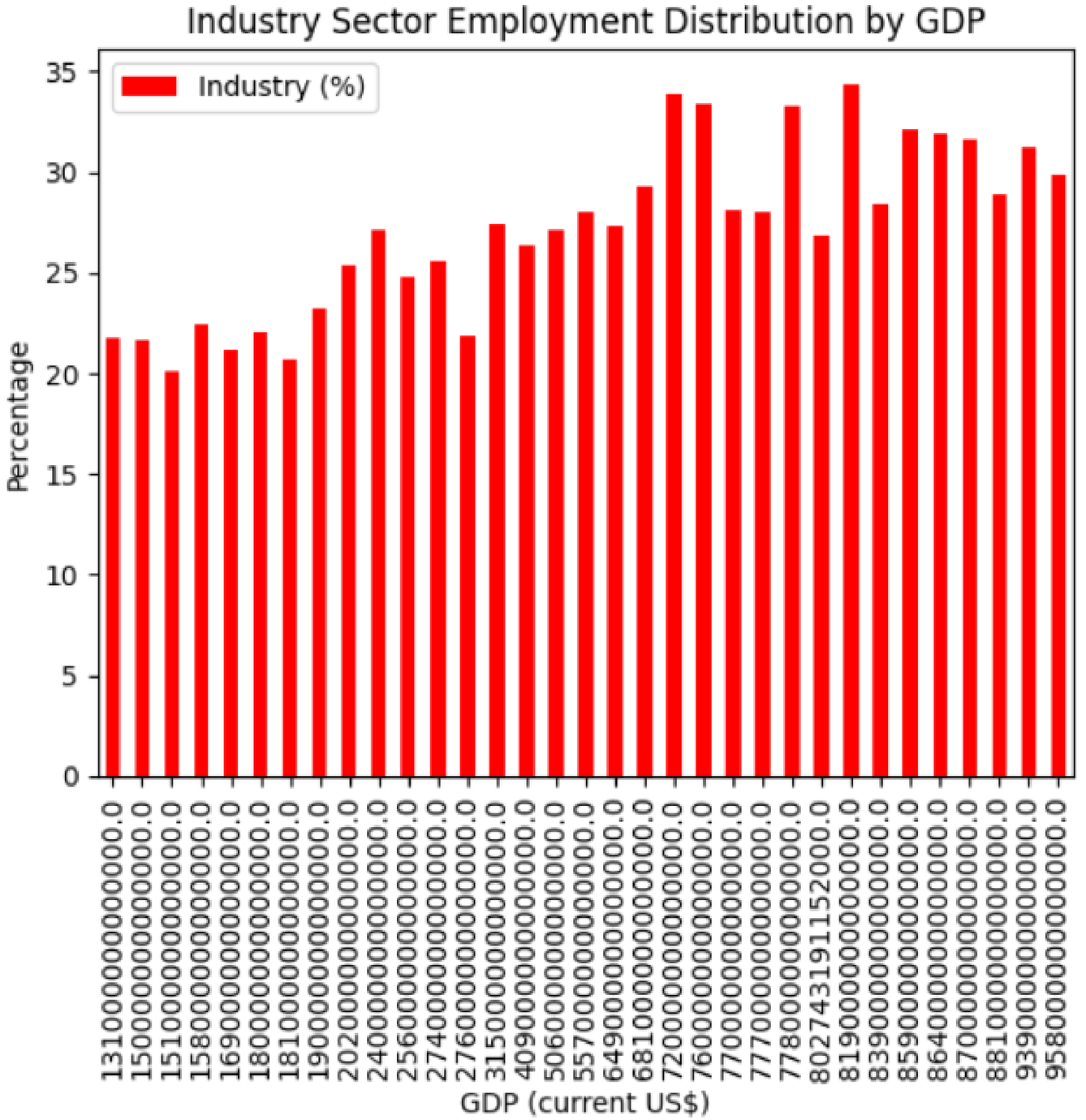
- Heatmap(correlation between columns)
- Socio-economic indicators
- Sectoral distrubitions
- Off-record workers number

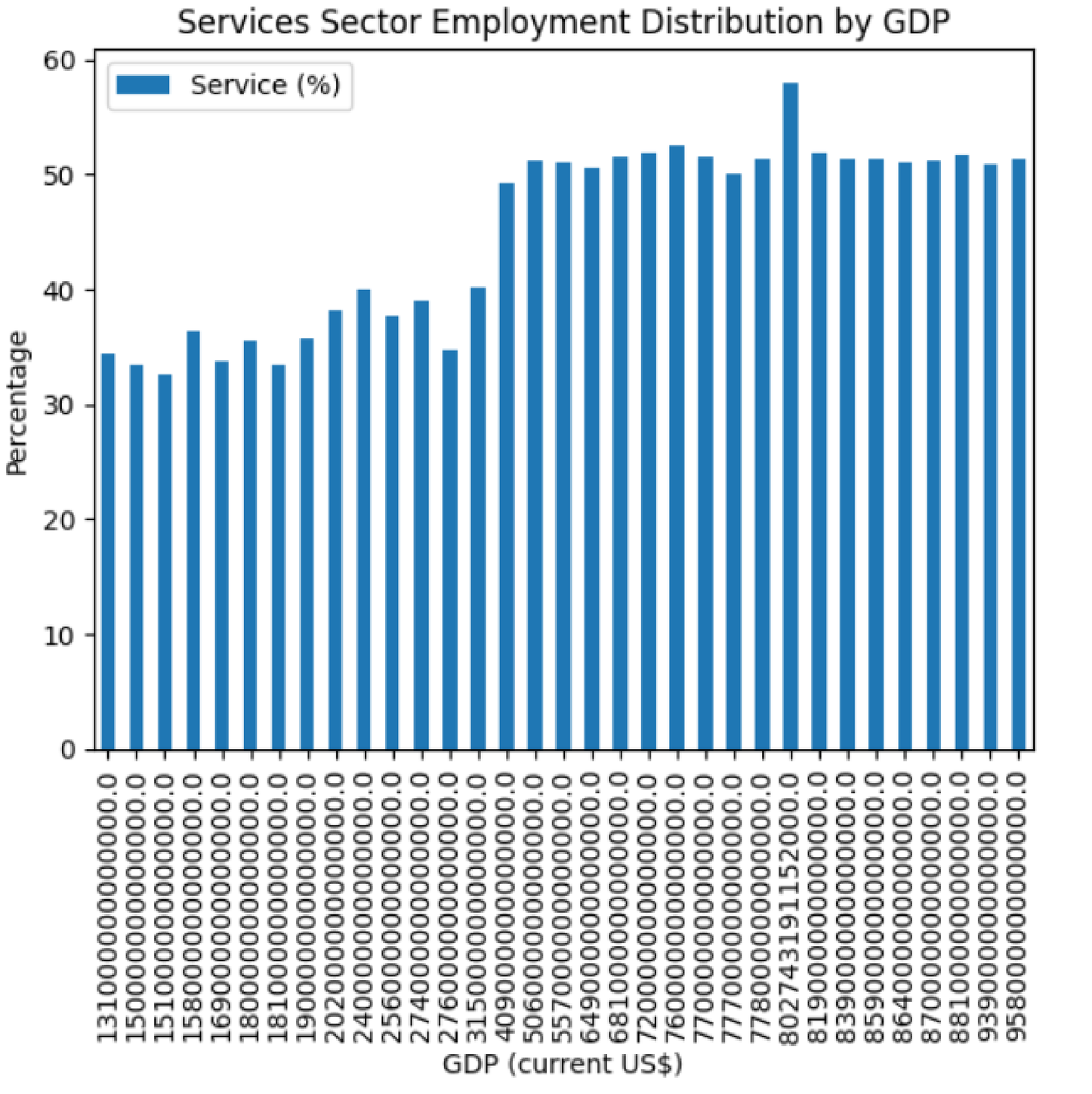

İşsizlik Verileri Heatmap
Year
total population)
Agriculture (%)
Industry (%)
Service (%)
num_students
total population)
DP (current US$)
)ff-record worker
1 -0.95 -0.95 0.93 0.91 0.95 0.95 0.92 -0.55
-0.95 1 0.93 -0.97 -0.87 -0.91 -1 -0.91 0.66
-0.95 0.93 1 -0.93 -0.98 -0.85 -0.93 -0.94 0.58
0.93 -0.97 -0.93 1 0.85 0.89 0.97 0.86 -0.61
0.91 -0.87 -0.98 0.85 1 0.79 0.87 0.93 -0.53
0.95 -0.91 -0.85 0.89 0.79 1 0.91 0.85 -0.61
0.95 -1 -0.93 0.97 0.87 0.91 1 0.91 -0.66
0.92 -0.91 -0.94 0.86 0.93 0.85 0.91 1 -0.64
-0.55 0.66 0.58 -0.61 -0.53 -0.61 -0.66 -0.64 1
Year
population)
culture (%)
ndustry (%)
Service (%)
n_students
population)
urrent US$)
cord worker
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

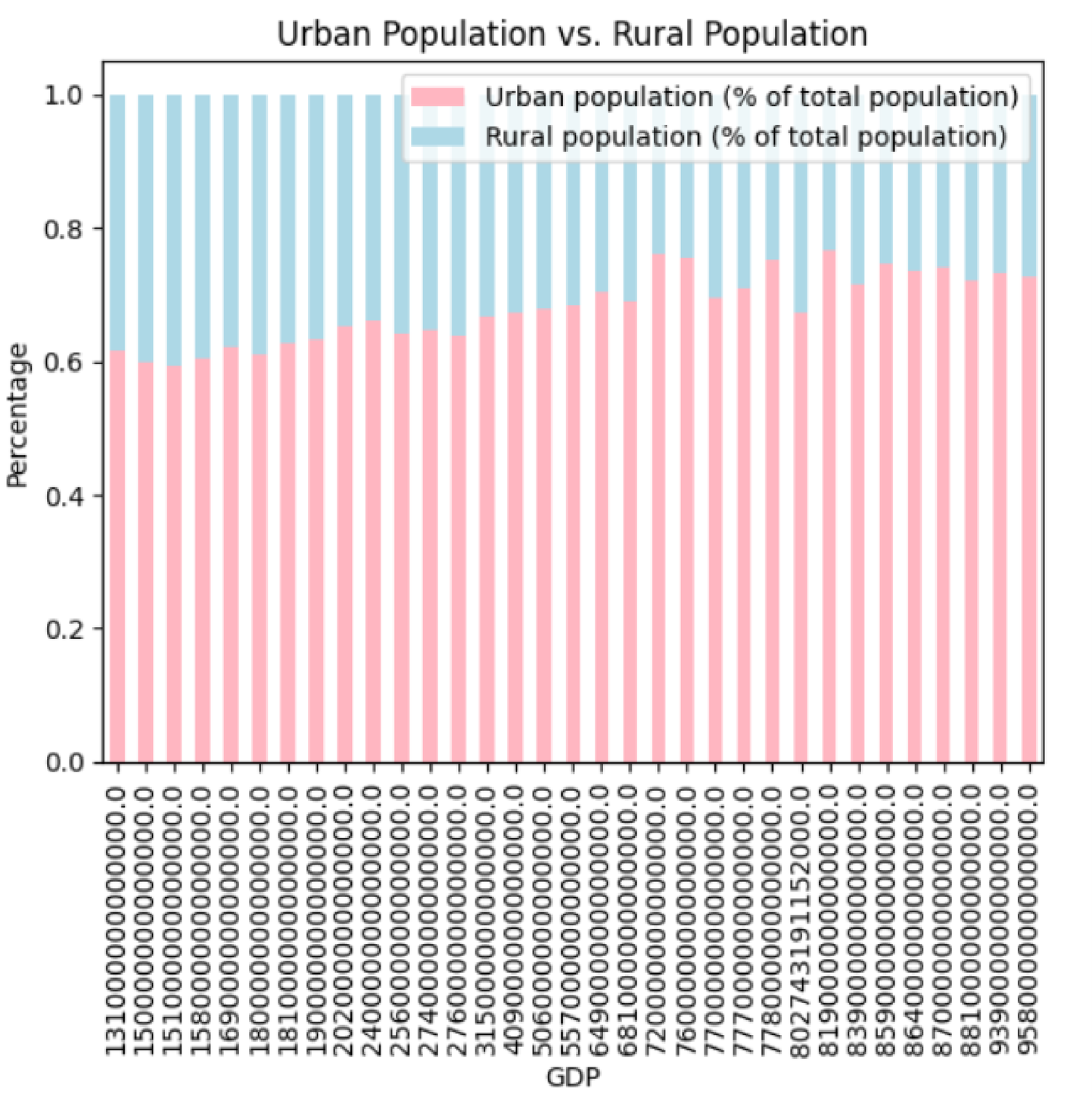

Socio-Economic Indicators Over Time
percentage_of_rural_population
GDP
total_number_of_university_students
Indicator Value
1000
800
600
400
200
1992
1996
2000
2004
2008
2012
2016
2020
Year

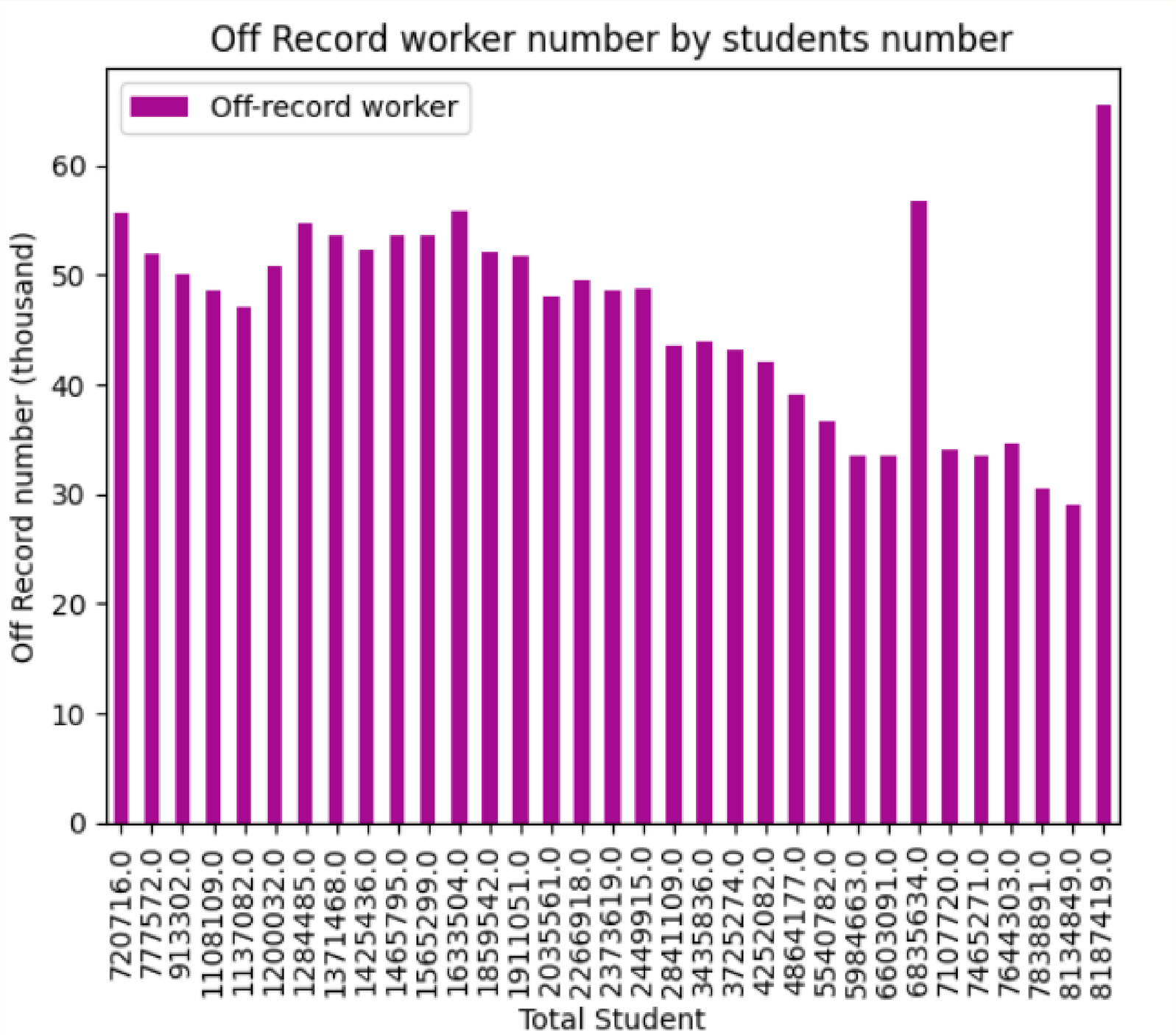

Sectoral Employment Distribution
Agriculture (%)
Industry (%)
Service (%)
Percentage
Year
60
50
40
30
20
1990
1995
2000
2005
2010
2015
2020

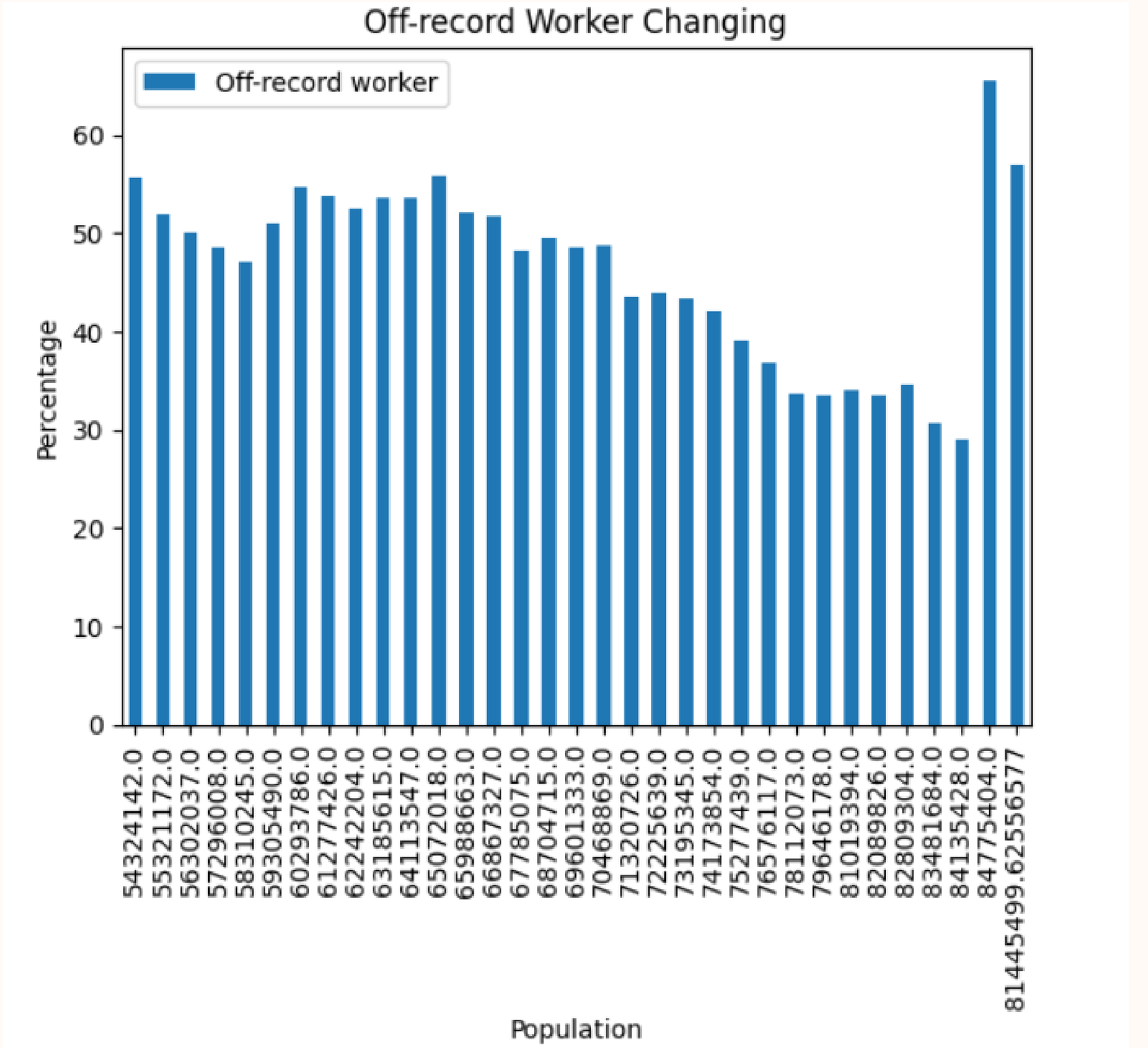

Agriculture Sector Employment Distribution by GDP
Agriculture (%)
Percentage
0
10
20
30
40
1310000000000.0
1500000000000.0
1510000000000.0
1580000000000.0
1690000000000.0
1800000000000.0
1810000000000.0
1900000000000.0
2020000000000.0
2400000000000.0
2560000000000.0
2740000000000.0
2760000000000.0
3150000000000.0
4090000000000.0
5060000000000.0
5570000000000.0
6490000000000.0
6810000000000.0
7200000000000.0
7600000000000.0
7700000000000.0
7770000000000.0
7780000000000.0
8027431911520000.0
8190000000000.0
8390000000000.0
8590000000000.0
8640000000000.0
8700000000000.0
8810000000000.0
9390000000000.0
9580000000000.0
GDP (current US$)


Industry Sector Employment Distribution by GDP
Industry (%)
Percentage
0
5
10
15
20
25
30
35
1310000000000000.0
1500000000000000.0
1510000000000000.0
1580000000000000.0
1690000000000000.0
1800000000000000.0
1810000000000000.0
1900000000000000.0
2020000000000000.0
2400000000000000.0
2560000000000000.0
2740000000000000.0
2760000000000000.0
3150000000000000.0
4090000000000000.0
5060000000000000.0
5570000000000000.0
6490000000000000.0
6810000000000000.0
7200000000000000.0
7600000000000000.0
7700000000000000.0
7770000000000000.0
7780000000000000.0
8027431911520000.0
8190000000000000.0
8390000000000000.0
8590000000000000.0
8640000000000000.0
8700000000000000.0
8810000000000000.0
9390000000000000.0
9580000000000000.0
GDP (current US$)

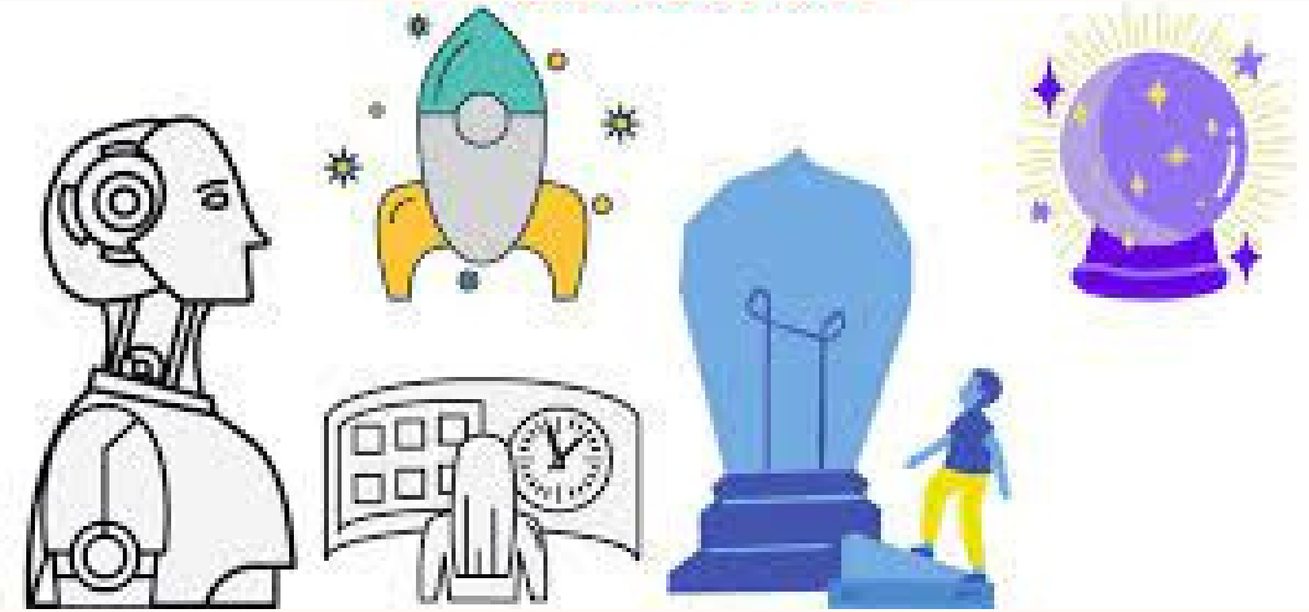

Services Sector Employment Distribution by GDP
Service (%)
Percentage
0
10
20
30
40
50
60
1310000000000.0
1500000000000.0
1510000000000.0
1580000000000.0
1690000000000.0
1800000000000.0
1810000000000.0
1900000000000.0
2020000000000.0
2400000000000.0
2560000000000.0
2740000000000.0
2760000000000.0
3150000000000.0
4090000000000.0
5060000000000.0
5570000000000.0
6490000000000.0
6810000000000.0
7200000000000.0
7600000000000.0
7700000000000.0
7770000000000.0
7780000000000.0
8027431911520000.0
8190000000000.0
8390000000000.0
8590000000000.0
8640000000000.0
8700000000000.0
8810000000000.0
9390000000000.0
9580000000000.0
GDP (current US$)


Urban Population vs. Rural Population
Urban population (% of total population)
Rural population (% of total population)
Percentage
1.0
0.8
0.6
0.4
0.2
0.0
GDP
1310000000000.0
1500000000000.0
1510000000000.0
1580000000000.0
1690000000000.0
1800000000000.0
1810000000000.0
1900000000000.0
2020000000000.0
2400000000000.0
2560000000000.0
2740000000000.0
2760000000000.0
3150000000000.0
4090000000000.0
5060000000000.0
5570000000000.0
6490000000000.0
6810000000000.0
7200000000000.0
7600000000000.0
7700000000000.0
7770000000000.0
7780000000000.0
8027431911520000.0
8190000000000.0
8390000000000.0
8590000000000.0
8640000000000.0
8700000000000.0
8810000000000.0
9390000000000.0
9580000000000.0


Off Record worker number by students number
Off-record worker
Off Record number (thousand)
0
10
20
30
40
50
60
720716.0
777572.0
913302.0
1108109.0
1137082.0
1200032.0
1284485.0
1371468.0
1425436.0
1465795.0
1565299.0
1633504.0
1859542.0
1911051.0
2035561.0
2266918.0
2373619.0
2449915.0
2841109.0
3435836.0
3725274.0
4252082.0
4864177.0
5540782.0
5984663.0
6603091.0
6835634.0
7107720.0
7465271.0
7644303.0
7838891.0
8134849.0
8187419.0
Total Student


Off-record Worker Changing
Off-record worker
Percentage
0
10
20
30
40
50
60
54324142.0
55321172.0
56302037.0
57296008.0
58310245.0
59305490.0
60293786.0
61277426.0
62242204.0
63185615.0
64113547.0
65072018.0
65988663.0
66867327.0
67785075.0
68704715.0
69601333.0
70468869.0
71320726.0
72225639.0
73195345.0
74173854.0
75277439.0
76576117.0
78112073.0
79646178.0
81019394.0
82089826.0
82809304.0
83481684.0
84135428.0
84775404.0
81445499.62556577
Population

PREDICTION



```python
###################### Prediction Unemployment Rate  ############################
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error



X = df[['GDP (current US$)','Population, total', 'Rural population (% of total population)',"Urban population (% of total population)",  'Agriculture (%)', 'Industry (%)', 'Service (%)','num_students']]
y = df['Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)']  # Bağımlı değişken

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
print('RMSE:', rmse)


```

```
RMSE: 0.13274304676343723
```

```python
import pandas as pd
from sklearn.linear_model import LinearRegression

X = df[['Rural population (% of total population)',
        'Urban population (% of total population)',
        'GDP (current US$)',
        'Agriculture (%)',
        'Industry (%)',
        'Service (%)']]
y = df['Unemployment, total (out of 10 of total labor force) (modeled ILO estimate)']

model = LinearRegression()
model.fit(X, y)

gelecek_sene = [[0.24, 0.76, 8.27432e+14, 15.2, 26.8, 58.0]]
gelecek_sene_tahmini = model.predict(gelecek_sene)

print("Prediction for the next year (2023):", gelecek_sene_tahmini)

```

```
Gelecek sene için tahmin edilen işsizlik oranı: [0.96320217]
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

```
model = sm.OLS(y, X)

result = model.fit()

aic = result.aic
print('AIC:', aic)

X = sm.add_constant(X)

model = sm.OLS(y, X)

result = model.fit()

p_values = result.pvalues

for i in range(len(X.columns)):
    print(f'{X.columns[i]}: {p_values[i]}')

```

```
RMSE: 0.13274304676343723
R2 Score: 0.5560181236782262
AIC: -45.15424234159396
const: 0.016035763284765213
GDP (current US$): 0.004431747473550619
Population, total: 0.1389213478070547
Rural population (% of total population): 0.02514834082404094
Urban population (% of total population): 0.03237787367617189
Agriculture (%): 0.3246553544430658
Industry (%): 0.04103193170475695
Service (%): 0.012231988806975253
num_students: 0.025980216969597666
```

CONCLUSION

RMSE: 0.13274306343 7

R2 SCORE: 0.5560181 2

AIC:-45.154242341 5

P-values

- **GDP : 0.00443174747**
- **Population:0.138921347**
- **Rural population: 0.025148340**
- **Urban population: 0.03237787367**
- **Agriculture:0.3246553544**
- **Industry:0.0410319343**
- **Servise:0.012231988806**