Life Expectancy

May 9, 2024

[4]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[24]:
```python
df=pd.read_csv("C:/Users/Ravi/Downloads/lass0/Datasets_LassoRidge/
↪Life_expectencey_LR.csv")
```

[25]:
```python
df.head()
```

[25]:
```
       Country  Year      Status  Life_expectancy  Adult_Mortality  \
0  Afghanistan  2015  Developing             65.0            263.0
1  Afghanistan  2014  Developing             59.9            271.0
2  Afghanistan  2013  Developing             59.9            268.0
3  Afghanistan  2012  Developing             59.5            272.0
4  Afghanistan  2011  Developing             59.2            275.0

   infant_deaths  Alcohol  percentage_expenditure  Hepatitis_B  Measles  …  \
0             62     0.01               71.279624         65.0     1154  …
1             64     0.01               73.523582         62.0      492  …
2             66     0.01               73.219243         64.0      430  …
3             69     0.01               78.184215         67.0     2787  …
4             71     0.01                7.097109         68.0     3013  …

   Polio  Total_expenditure  Diphtheria  HIV_AIDS         GDP  Population  \
0    6.0               8.16        65.0       0.1  584.259210  33736494.0
1   58.0               8.18        62.0       0.1  612.696514    327582.0
2   62.0               8.13        64.0       0.1  631.744976  31731688.0
3   67.0               8.52        67.0       0.1  669.959000   3696958.0
4   68.0               7.87        68.0       0.1   63.537231   2978599.0

   thinness  thinness_yr  Income_composition  Schooling
0      17.2         17.3               0.479       10.1
1      17.5         17.5               0.476       10.0
2      17.7         17.7               0.470        9.9
3      17.9         18.0               0.463        9.8
4      18.2         18.2               0.454        9.5
```

```
[5 rows x 22 columns]
```

[26]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2938 entries, 0 to 2937
Data columns (total 22 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Country                  2938 non-null   object
 1   Year                     2938 non-null   int64
 2   Status                   2938 non-null   object
 3   Life_expectancy          2928 non-null   float64
 4   Adult_Mortality          2928 non-null   float64
 5   infant_deaths            2938 non-null   int64
 6   Alcohol                  2744 non-null   float64
 7   percentage_expenditure   2938 non-null   float64
 8   Hepatitis_B              2385 non-null   float64
 9   Measles                  2938 non-null   int64
 10  BMI                      2904 non-null   float64
 11  under_five_deaths        2938 non-null   int64
 12  Polio                    2919 non-null   float64
 13  Total_expenditure        2712 non-null   float64
 14  Diphtheria               2919 non-null   float64
 15  HIV_AIDS                 2938 non-null   float64
 16  GDP                      2490 non-null   float64
 17  Population               2286 non-null   float64
 18  thinness                 2904 non-null   float64
 19  thinness_yr              2904 non-null   float64
 20  Income_composition       2771 non-null   float64
 21  Schooling                2775 non-null   float64
dtypes: float64(16), int64(4), object(2)
memory usage: 505.1+ KB
```

[31]:
```
df.isna().sum()
```

[31]:
```
Country                     0
Year                        0
Status                      0
Life_expectancy             0
Adult_Mortality            10
infant_deaths               0
Alcohol                   194
percentage_expenditure      0
Hepatitis_B               553
Measles                     0
BMI                        34
under_five_deaths           0
```

```
Polio                    19
Total_expenditure       226
Diphtheria               19
HIV_AIDS                  0
GDP                     448
Population              652
thinness                 34
thinness_yr              34
Income_composition      167
Schooling               163
dtype: int64
```

[29]:
```
life_mean=df['Life_expectancy'].mean()
```

[30]:
```
df.Life_expectancy=df.Life_expectancy.fillna(life_mean)
```

[32]:
```
adult_mean=df['Adult_Mortality'].mean()
df.Adult_Mortality=df.Adult_Mortality.fillna(adult_mean)
```

[33]:
```
Alcohol_mean=df['Alcohol'].mean()
df.Alcohol=df.Alcohol.fillna(Alcohol_mean)
```

[36]:
```
hb_mean=df['Hepatitis_B'].mean()
df.Hepatitis_B=df.Hepatitis_B.fillna(hb_mean)
```

[37]:
```
BMI_mean=df['BMI'].mean()
df.BMI=df.BMI.fillna(BMI_mean)
```

[38]:
```
Polio_mean=df['Polio'].mean()
df.Polio=df.Polio.fillna(Polio_mean)
```

[39]:
```
T_exp_mean=df['Total_expenditure'].mean()
df.Total_expenditure=df.Total_expenditure.fillna(T_exp_mean)
```

[40]:
```
Diphtheria_mean=df['Diphtheria'].mean()
df.Diphtheria=df.Diphtheria.fillna(Diphtheria_mean)
```

[41]:
```
GDP_mean=df['GDP'].mean()
df.GDP=df.GDP.fillna(GDP_mean)
```

[42]:
```
Population_mean=df['Population'].mean()
df.Population=df.Population.fillna(Population_mean)
```

[43]:
```
thinness_mean=df['thinness'].mean()
df.thinness=df.thinness.fillna(thinness_mean)
```

[44]:
```
thinness_yr_mean=df['thinness_yr'].mean()
df.thinness_yr=df.thinness_yr.fillna(thinness_yr_mean)
```

[45]:
```
Income_composition_mean=df['Income_composition'].mean()
df.Income_composition=df.Income_composition.fillna(Income_composition_mean)
```

[46]:
```
Schooling_mean=df['Schooling'].mean()
df.Schooling=df.Schooling.fillna(Schooling_mean)
```

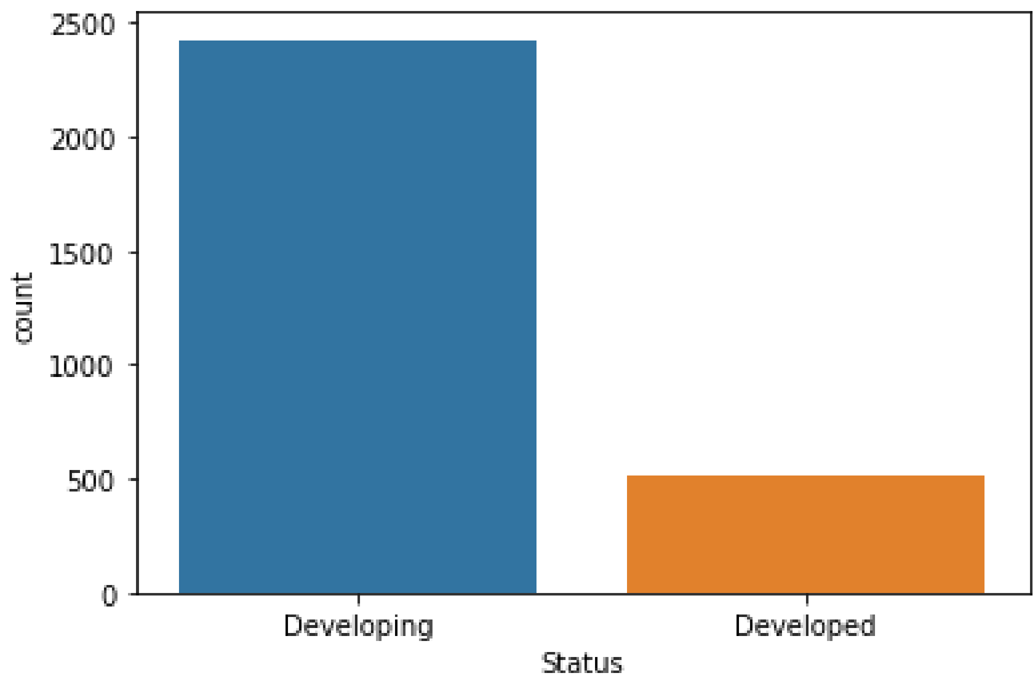
[47]:
```
df.isna().sum()
```

[47]:
```
Country                   0
Year                      0
Status                    0
Life_expectancy           0
Adult_Mortality           0
infant_deaths             0
Alcohol                   0
percentage_expenditure    0
Hepatitis_B               0
Measles                   0
BMI                       0
under_five_deaths         0
Polio                     0
Total_expenditure         0
Diphtheria                0
HIV_AIDS                  0
GDP                       0
Population                0
thinness                  0
thinness_yr               0
Income_composition        0
Schooling                 0
dtype: int64
```

[51]:
```
sns.countplot(df.Status)
```

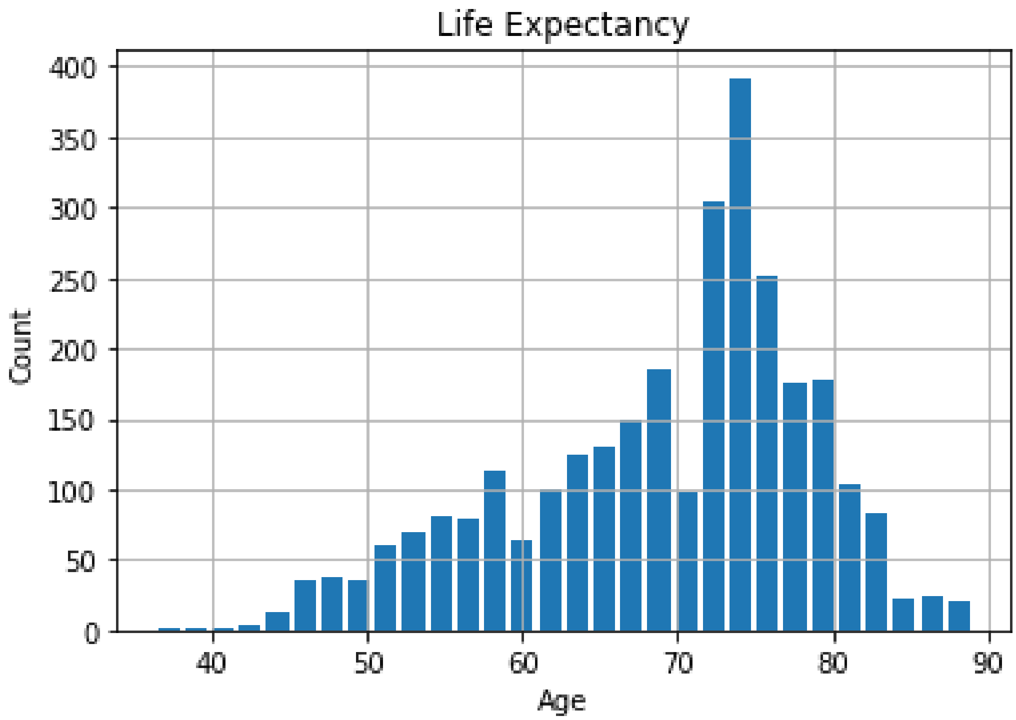
```
C:\Users\Ravi\anaconda3\lib\site-packages\seaborn\_decorators.py:36:
FutureWarning: Pass the following variable as a keyword arg: x. From version
0.12, the only valid positional argument will be `data`, and passing other
arguments without an explicit keyword will result in an error or
misinterpretation.
  warnings.warn(
```

[51]:
```
<AxesSubplot:xlabel='Status', ylabel='count'>
```




```
[53]: df["Life_expectancy"].plot.hist(grid=True, bins=30, rwidth=0.8)
      plt.title('Life Expectancy')
      plt.ylabel('Count')
      plt.xlabel('Age')
```

```
[53]: Text(0.5, 0, 'Age')
```

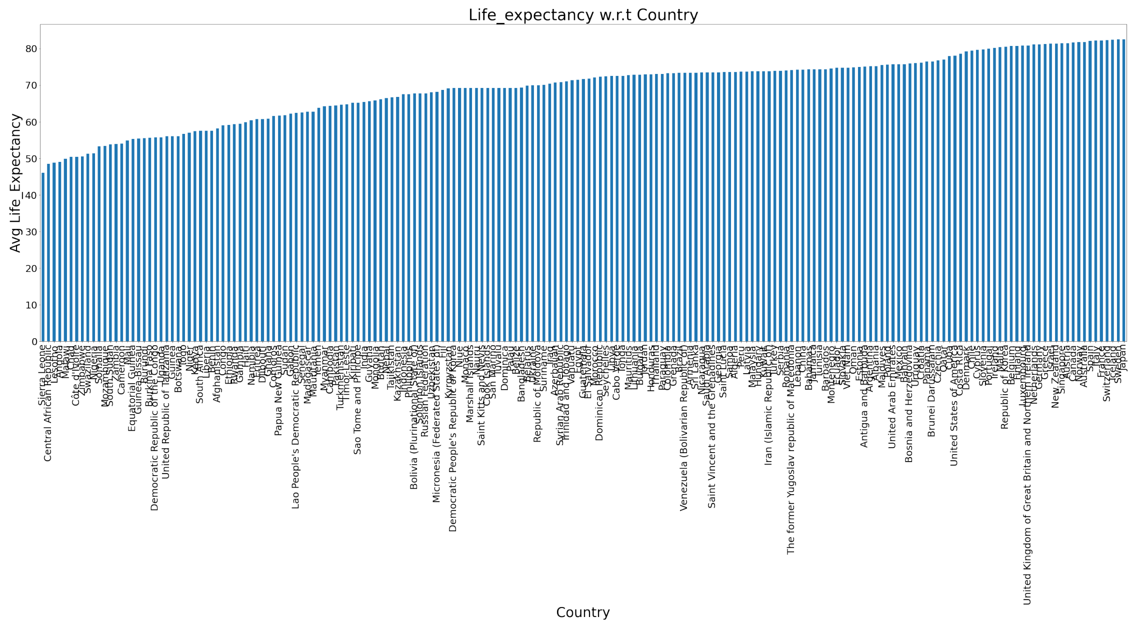


[55]:
```
df.describe(include= 'O')
```

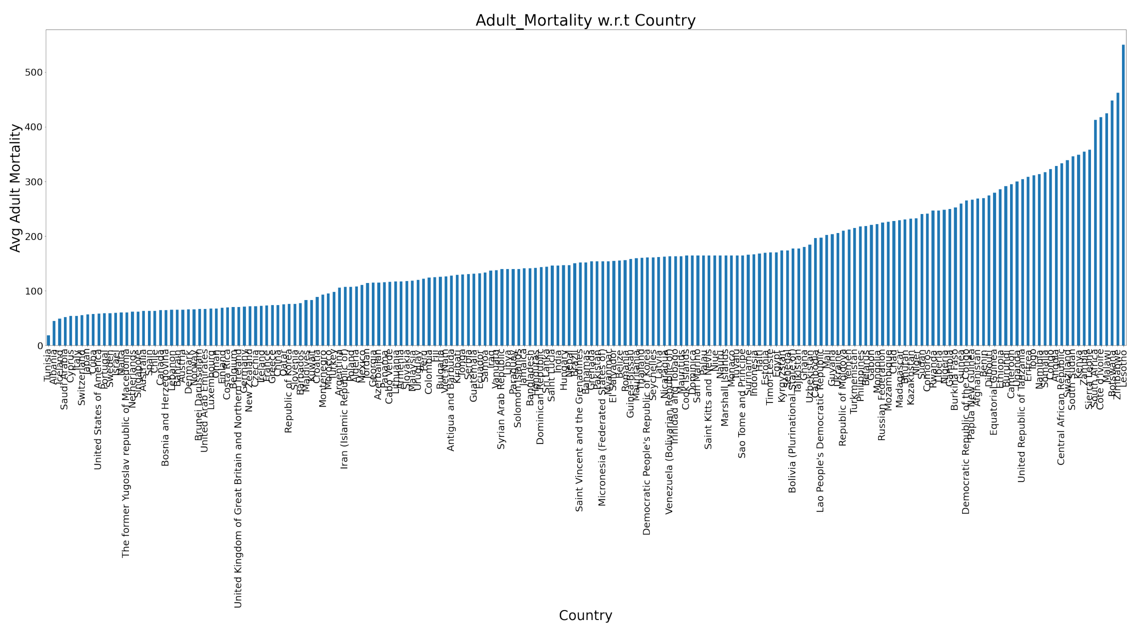
[55]:

	Country	Status
count	2938	2938
unique	193	2
top	Afghanistan	Developing
freq	16	2426

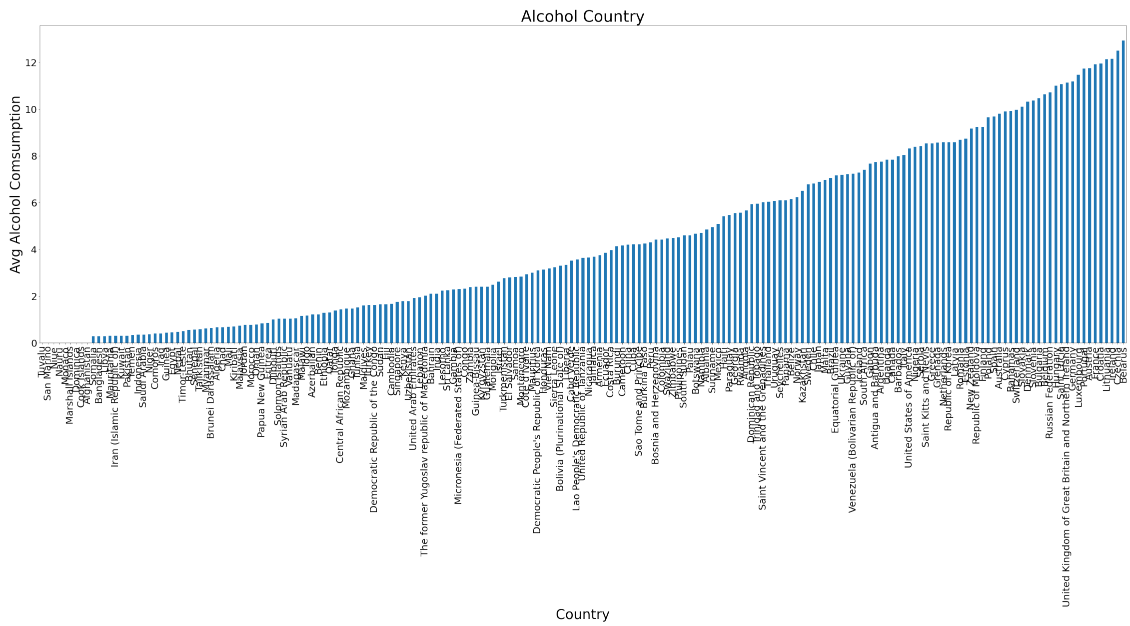
[57]:
```
le_country = df.groupby('Country')['Life_expectancy'].mean().
 ↪sort_values(ascending=True)
le_country.plot(kind='bar', figsize=(50,15), fontsize=25)
plt.title("Life_expectancy w.r.t Country",fontsize=40)
plt.xlabel("Country",fontsize=35)
plt.ylabel("Avg Life_Expectancy",fontsize=35)
plt.show()
```



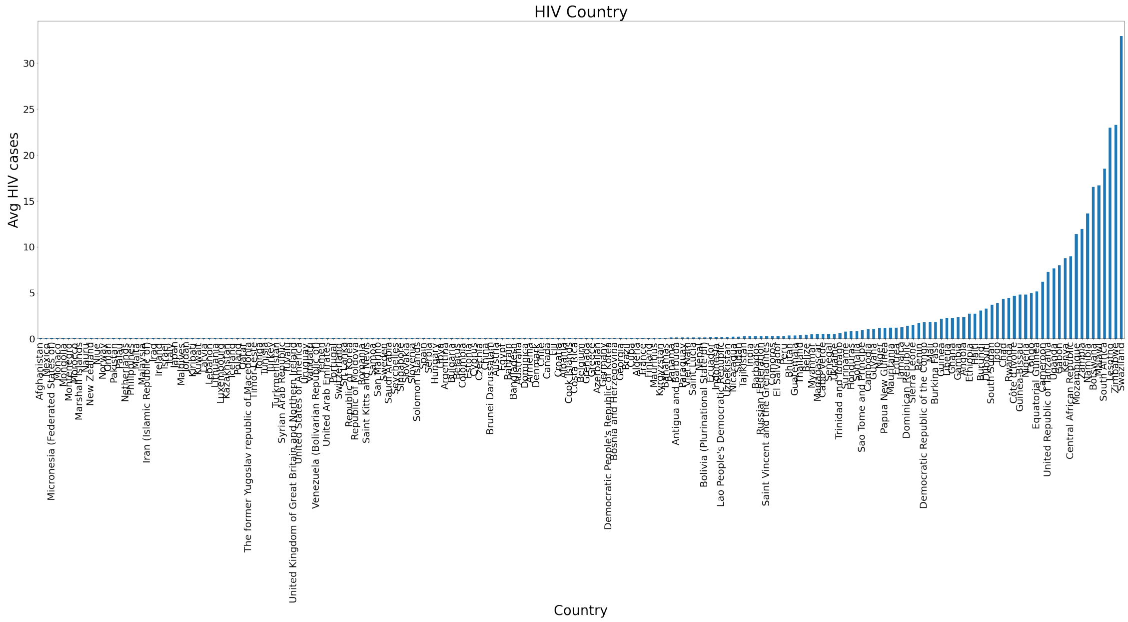

```
[59]: le_country = df.groupby('Country')['Adult_Mortality'].mean().
        ↪sort_values(ascending=True)
      le_country.plot(kind='bar', figsize=(50,15), fontsize=25)
      plt.title("Adult_Mortality Country",fontsize=40)
      plt.xlabel("Country",fontsize=35)
      plt.ylabel("Avg Adult Mortality",fontsize=35)
      plt.show()
```



[60]:
```
le_country = df.groupby('Country')['Alcohol'].mean().sort_values(ascending=True)
le_country.plot(kind='bar', figsize=(50,15), fontsize=25)
plt.title("Alcohol Country",fontsize=40)
plt.xlabel("Country",fontsize=35)
plt.ylabel("Avg Alcohol Comsumption",fontsize=35)
plt.show()
```

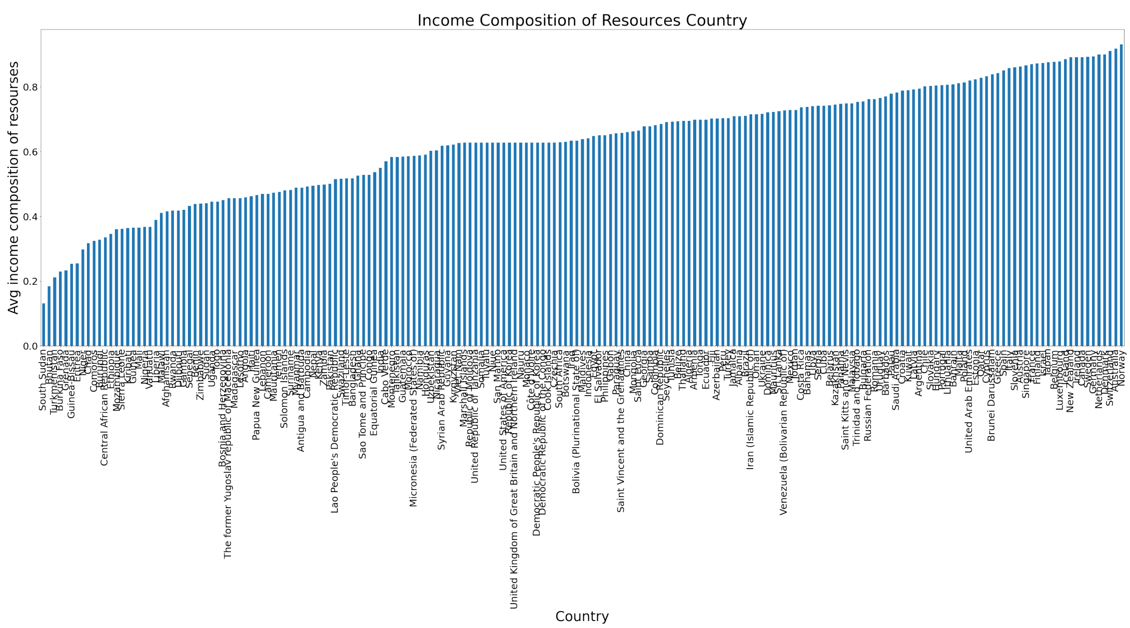


[62]:
```
le_country = df.groupby('Country')['HIV_AIDS'].mean().
↪sort_values(ascending=True)
le_country.plot(kind='bar', figsize=(50,15), fontsize=25)
plt.title("HIV Country",fontsize=40)
plt.xlabel("Country",fontsize=35)
plt.ylabel("Avg HIV cases",fontsize=35)
plt.show()
```



[64]:

```
le_country = df.groupby('Country')['Income_composition'].mean().
  ↪sort_values(ascending=True)
le_country.plot(kind='bar', figsize=(50,15), fontsize=25)
plt.title("Income Composition of Resources Country",fontsize=40)
plt.xlabel("Country",fontsize=35)
plt.ylabel("Avg income composition of resourses",fontsize=35)
plt.show()
```



[ ]:

```

```

[ ]:

```

```

[65]:

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

[66]:

```
df.Country=le.fit_transform(df.Country)
```

[68]:

```
df.Status=le.fit_transform(df.Status)
```

[9]:

```

```

[69]:

```
df.shape
```

[69]:

```
(2938, 22)
```

[70]:

```
df.isna().sum()
```

[70]:

```
Country                   0
Year                      0
Status                    0
Life_expectancy           0
Adult_Mortality           0
infant_deaths             0
Alcohol                   0
percentage_expenditure    0
Hepatitis_B               0
Measles                   0
BMI                       0
under_five_deaths         0
Polio                     0
Total_expenditure         0
Diphtheria                0
HIV_AIDS                  0
GDP                       0
Population                0
thinness                  0
thinness_yr               0
Income_composition        0
Schooling                 0
dtype: int64
```

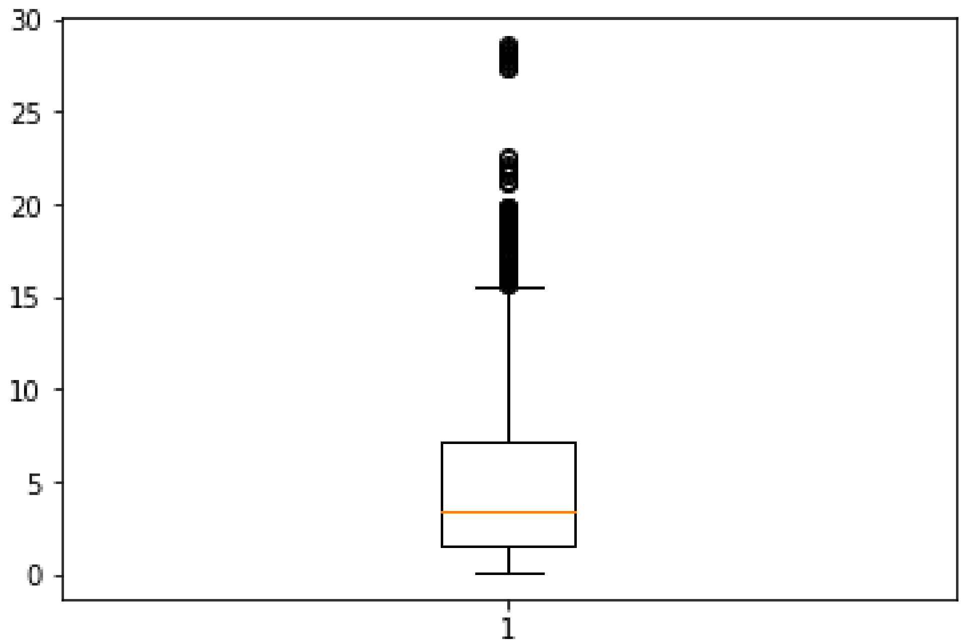
[71]:

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

[71]:
```
Country                  0
Year                     0
Status                   0
Life_expectancy          0
Adult_Mortality          0
infant_deaths            0
Alcohol                  0
percentage_expenditure   0
Hepatitis_B              0
Measles                  0
BMI                      0
under_five_deaths        0
Polio                    0
Total_expenditure        0
Diphtheria               0
HIV_AIDS                 0
GDP                      0
Population               0
thinness                 0
thinness_yr              0
Income_composition       0
Schooling                0
dtype: int64
```

[82]:
```
plt.boxplot(df.thinness)
```

[82]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x1b69f738610>,
  <matplotlib.lines.Line2D at 0x1b69f7389a0>],
 'caps': [<matplotlib.lines.Line2D at 0x1b69f738d30>,
  <matplotlib.lines.Line2D at 0x1b69f751100>],
 'boxes': [<matplotlib.lines.Line2D at 0x1b69f738280>],
 'medians': [<matplotlib.lines.Line2D at 0x1b69f751490>],
 'fliers': [<matplotlib.lines.Line2D at 0x1b69f751820>],
 'means': []}
```



[74]:
```
iqr=df.thinness.quantile(0.75)-df.thinness.quantile(0.25)
lowerlimit=df.thinness.quantile(0.25)-(iqr*1.5)
upperlimit=df.thinness.quantile(0.75)-(iqr*1.5)
```

[75]:
```
from feature_engine.outliers import Winsorizer
```

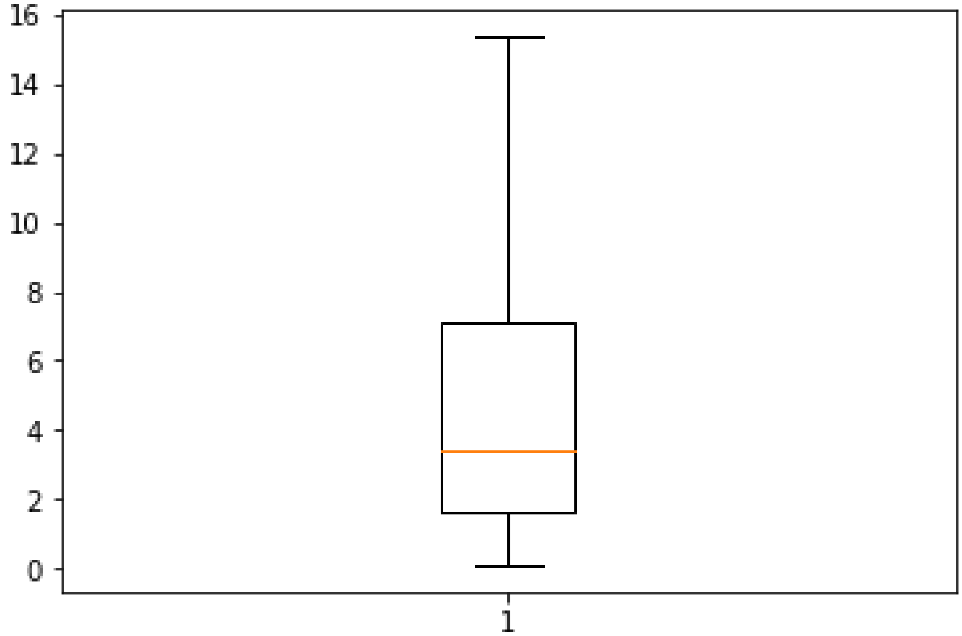
[76]:
```
winsorizer=Winsorizer(capping_method='iqr',tail='both',fold=1.
 ↪5,variables=["thinness"])
```

[77]:
```
df_t=winsorizer.fit_transform(df[['thinness']])
```

[80]:
```
df.thinness=df_t
```

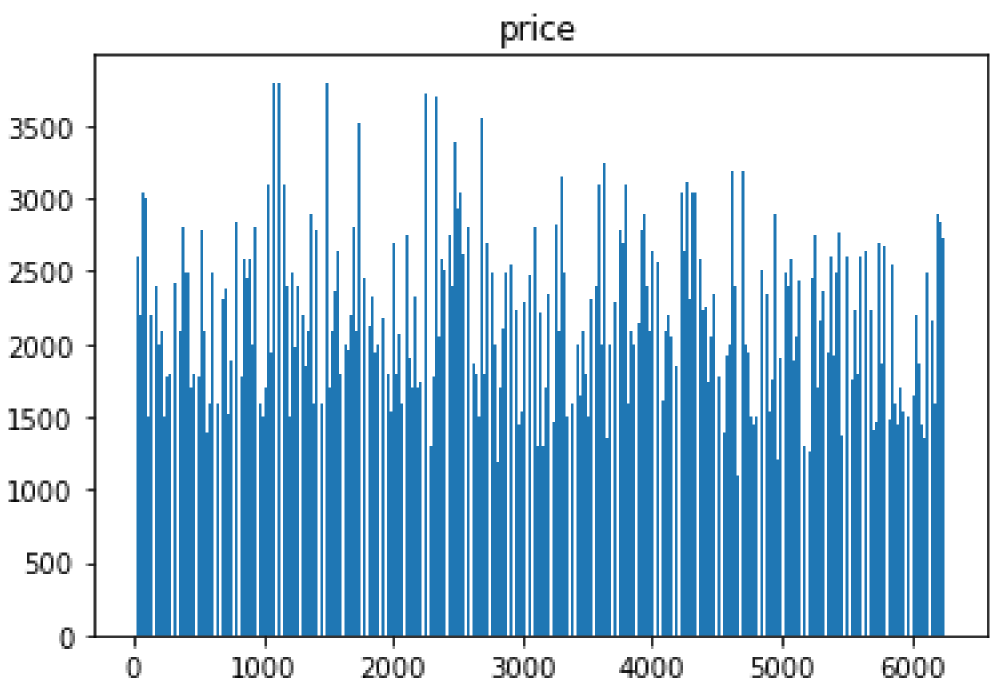
[81]:
```
plt.boxplot(df.thinness)
```

[81]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x1b69f616910>,
  <matplotlib.lines.Line2D at 0x1b69f616ca0>],
 'caps': [<matplotlib.lines.Line2D at 0x1b69f594070>,
  <matplotlib.lines.Line2D at 0x1b69f594400>],
 'boxes': [<matplotlib.lines.Line2D at 0x1b69f616550>],
 'medians': [<matplotlib.lines.Line2D at 0x1b69f594790>],
 'fliers': [<matplotlib.lines.Line2D at 0x1b69f594b20>],
 'means': []}
```



[20]:

[20]: Text(0.5, 1.0, 'price')

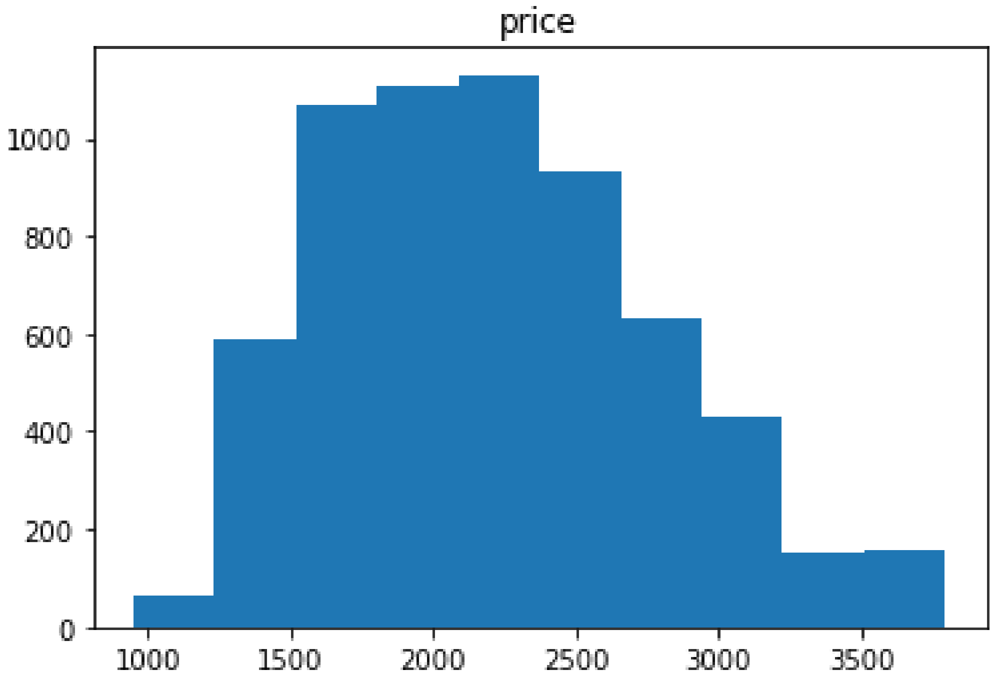


[21]:

```

```

[21]: Text(0.5, 1.0, 'price')

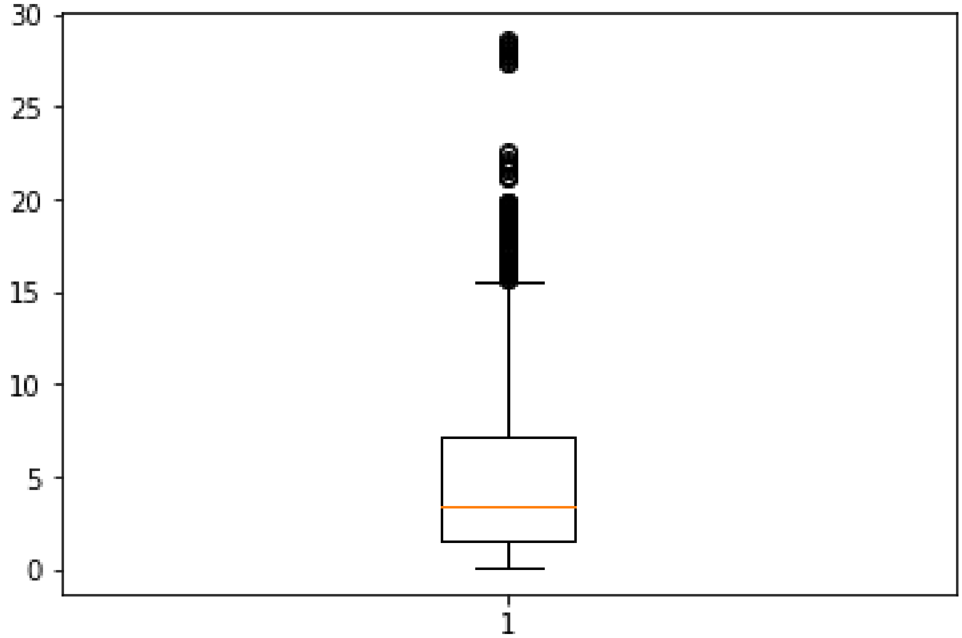


[83]:

```
plt.boxplot(df.thinness_yr)
```

[83]:

```
{'whiskers': [<matplotlib.lines.Line2D at 0x1b69f69c4f0>,
  <matplotlib.lines.Line2D at 0x1b69f69c880>],
 'caps': [<matplotlib.lines.Line2D at 0x1b69f69cc10>,
  <matplotlib.lines.Line2D at 0x1b69f69cfa0>],
 'boxes': [<matplotlib.lines.Line2D at 0x1b69f69c160>],
 'medians': [<matplotlib.lines.Line2D at 0x1b69f6a3370>],
 'fliers': [<matplotlib.lines.Line2D at 0x1b69f6a3700>],
 'means': []}
```



[ ]:

[ ]:

[ ]:

[ ]:

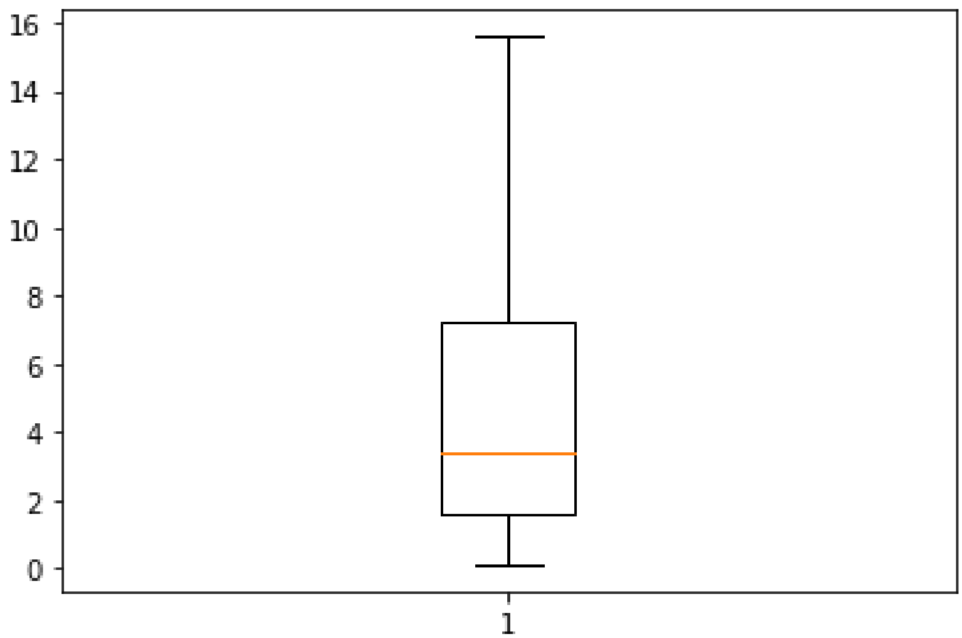
[84]:
```
IQR=df.thinness_yr.quantile(0.75)-df.thinness_yr.quantile(0.25)
lower_limit=df.thinness_yr.quantile(0.25)-(IQR*1.5)
upper_limit=df.thinness_yr.quantile(0.75)-(IQR*1.5)

from feature_engine.outliers import Winsorizer
winsor = Winsorizer(capping_method='iqr', # choose  IQR rule boundaries or␣
 ↪gaussian for mean and std
                          tail='both', # cap left, right or both tails
                          fold=1.5,
                           variables=['thinness_yr'])
```

[85]:
```
df_t=winsor.fit_transform(df[["thinness_yr"]])
df.thinness_yr=df_t
```

[86]:
```
plt.boxplot(df.thinness_yr)
```

[86]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x1b69f5c43d0>,
  <matplotlib.lines.Line2D at 0x1b69f5c4760>],
 'caps': [<matplotlib.lines.Line2D at 0x1b69f5c4af0>,
  <matplotlib.lines.Line2D at 0x1b69f5c4e80>],
 'boxes': [<matplotlib.lines.Line2D at 0x1b69f5c4040>],
 'medians': [<matplotlib.lines.Line2D at 0x1b69f43a250>],
 'fliers': [<matplotlib.lines.Line2D at 0x1b69f43a5e0>],
 'means': []}
```



[88]:
```
df.corr()
```

[88]:

	Country	Year	Status	Life_expectancy	\
Country	1.000000	0.001342	-0.031635	-0.016745	
Year	0.001342	1.000000	0.001864	0.169623	
Status	-0.031635	0.001864	1.000000	-0.481962	
Life_expectancy	-0.016745	0.169623	-0.481962	1.000000	
Adult_Mortality	0.039760	-0.078861	0.315171	-0.696359	
infant_deaths	-0.030528	-0.037415	0.112252	-0.196535	
Alcohol	-0.060052	-0.048168	-0.579371	0.391598	
percentage_expenditure	-0.032983	0.031400	-0.454261	0.381791	
Hepatitis_B	-0.018918	0.089398	-0.095642	0.203771	
Measles	-0.024593	-0.082493	0.076955	-0.157574	
BMI	0.017724	0.108327	-0.310873	0.559255	
under_five_deaths	-0.026509	-0.042937	0.115195	-0.222503	
Polio	0.017750	0.093820	-0.220098	0.461574	
Total_expenditure	0.053226	0.081860	-0.289985	0.207981	

```
Diphtheria             -0.006119  0.133853 -0.216763         0.475418
HIV_AIDS                0.090206 -0.139741  0.148590        -0.556457
GDP                    -0.015201  0.093351 -0.445911         0.430493
Population             -0.014347  0.014951  0.041091        -0.019638
thinness                0.025432 -0.049048  0.395687        -0.511941
thinness_yr             0.041255 -0.049470  0.396347        -0.509252
Income_composition     -0.023600  0.236333 -0.457302         0.692483
Schooling              -0.025217  0.203471 -0.491444         0.715066

                        Adult_Mortality  infant_deaths   Alcohol  \
Country                        0.039760      -0.030528 -0.060052
Year                          -0.078861      -0.037415 -0.048168
Status                         0.315171       0.112252 -0.579371
Life_expectancy               -0.696359      -0.196535  0.391598
Adult_Mortality                1.000000       0.078747 -0.190408
infant_deaths                  0.078747       1.000000 -0.113812
Alcohol                       -0.190408      -0.113812  1.000000
percentage_expenditure        -0.242814      -0.085612  0.339634
Hepatitis_B                   -0.138591      -0.178783  0.075447
Measles                        0.031174       0.501128 -0.051055
BMI                           -0.381449      -0.227220  0.318070
under_five_deaths              0.094135       0.996629 -0.110777
Polio                         -0.272694      -0.170674  0.213744
Total_expenditure             -0.110875      -0.126564  0.294898
Diphtheria                    -0.273014      -0.175156  0.215242
HIV_AIDS                       0.523727       0.025231 -0.048650
GDP                           -0.277053      -0.107109  0.318591
Population                    -0.012501       0.548522 -0.030765
thinness                       0.335430       0.316137 -0.436035
thinness_yr                    0.342744       0.317831 -0.426368
Income_composition            -0.440062      -0.143663  0.416099
Schooling                     -0.435108      -0.191757  0.497546

                        percentage_expenditure  Hepatitis_B   Measles  …  \
Country                              -0.032983    -0.018918 -0.024593  …
Year                                  0.031400     0.089398 -0.082493  …
Status                               -0.454261    -0.095642  0.076955  …
Life_expectancy                       0.381791     0.203771 -0.157574  …
Adult_Mortality                      -0.242814    -0.138591  0.031174  …
infant_deaths                        -0.085612    -0.178783  0.501128  …
Alcohol                               0.339634     0.075447 -0.051055  …
percentage_expenditure                1.000000     0.011679 -0.056596  …
Hepatitis_B                           0.011679     1.000000 -0.090317  …
Measles                              -0.056596    -0.090317  1.000000  …
BMI                                   0.228537     0.134929 -0.175925  …
under_five_deaths                    -0.087852    -0.184413  0.507809  …
Polio                                 0.147203     0.408519 -0.136146  …
```

Total_expenditure	0.173414	0.050084	-0.104569	…
Diphtheria	0.143570	0.499958	-0.141861	…
HIV_AIDS	-0.097857	-0.102405	0.030899	…
GDP	0.888140	0.062318	-0.068060	…
Population	-0.024648	-0.109811	0.236250	…
thinness	-0.268853	-0.087571	0.187089	…
thinness_yr	-0.272232	-0.091770	0.183881	…
Income_composition	0.380374	0.150992	-0.115764	…
Schooling	0.388105	0.171755	-0.122609	…

	Polio	Total_expenditure	Diphtheria	HIV_AIDS	\
Country	0.017750	0.053226	-0.006119	0.090206	
Year	0.093820	0.081860	0.133853	-0.139741	
Status	-0.220098	-0.289985	-0.216763	0.148590	
Life_expectancy	0.461574	0.207981	0.475418	-0.556457	
Adult_Mortality	-0.272694	-0.110875	-0.273014	0.523727	
infant_deaths	-0.170674	-0.126564	-0.175156	0.025231	
Alcohol	0.213744	0.294898	0.215242	-0.048650	
percentage_expenditure	0.147203	0.173414	0.143570	-0.097857	
Hepatitis_B	0.408519	0.050084	0.499958	-0.102405	
Measles	-0.136146	-0.104569	-0.141861	0.030899	
BMI	0.282156	0.231814	0.281059	-0.243548	
under_five_deaths	-0.188703	-0.128269	-0.195651	0.038062	
Polio	1.000000	0.130129	0.673553	-0.159489	
Total_expenditure	0.130129	1.000000	0.145597	-0.001383	
Diphtheria	0.673553	0.145597	1.000000	-0.164787	
HIV_AIDS	-0.159489	-0.001383	-0.164787	1.000000	
GDP	0.193980	0.121467	0.182795	-0.134514	
Population	-0.034882	-0.066698	-0.025458	-0.027318	
thinness	-0.229618	-0.283672	-0.238115	0.237965	
thinness_yr	-0.230798	-0.292352	-0.232253	0.241636	
Income_composition	0.355398	0.149095	0.371729	-0.247454	
Schooling	0.385832	0.218310	0.389944	-0.218620	

	GDP	Population	thinness	thinness_yr	\
Country	-0.015201	-0.014347	0.025432	0.041255	
Year	0.093351	0.014951	-0.049048	-0.049470	
Status	-0.445911	0.041091	0.395687	0.396347	
Life_expectancy	0.430493	-0.019638	-0.511941	-0.509252	
Adult_Mortality	-0.277053	-0.012501	0.335430	0.342744	
infant_deaths	-0.107109	0.548522	0.316137	0.317831	
Alcohol	0.318591	-0.030765	-0.436035	-0.426368	
percentage_expenditure	0.888140	-0.024648	-0.268853	-0.272232	
Hepatitis_B	0.062318	-0.109811	-0.087571	-0.091770	
Measles	-0.068060	0.236250	0.187089	0.183881	
BMI	0.276645	-0.063238	-0.555854	-0.564329	
under_five_deaths	-0.110640	0.535864	0.324683	0.325260	

```
Polio                    0.193980   -0.034882 -0.229618    -0.230798
Total_expenditure        0.121467   -0.066698 -0.283672    -0.292352
Diphtheria               0.182795   -0.025458 -0.238115    -0.232253
HIV_AIDS                -0.134514   -0.027318  0.237965     0.241636
GDP                      1.000000   -0.025612 -0.281809    -0.288505
Population              -0.025612    1.000000  0.133040     0.129895
thinness                -0.281809    0.133040  1.000000     0.941991
thinness_yr             -0.288505    0.129895  0.941991     1.000000
Income_composition       0.440317   -0.007951 -0.430026    -0.419782
Schooling                0.429489   -0.029465 -0.467941    -0.459156

                        Income_composition  Schooling
Country                          -0.023600  -0.025217
Year                              0.236333   0.203471
Status                           -0.457302  -0.491444
Life_expectancy                   0.692483   0.715066
Adult_Mortality                  -0.440062  -0.435108
infant_deaths                    -0.143663  -0.191757
Alcohol                           0.416099   0.497546
percentage_expenditure            0.380374   0.388105
Hepatitis_B                       0.150992   0.171755
Measles                          -0.115764  -0.122609
BMI                               0.479837   0.508105
under_five_deaths                -0.161533  -0.207111
Polio                             0.355398   0.385832
Total_expenditure                 0.149095   0.218310
Diphtheria                        0.371729   0.389944
HIV_AIDS                         -0.247454  -0.218620
GDP                               0.440317   0.429489
Population                       -0.007951  -0.029465
thinness                         -0.430026  -0.467941
thinness_yr                      -0.419782  -0.459156
Income_composition                1.000000   0.796207
Schooling                         0.796207   1.000000

[22 rows x 22 columns]
```

[89]:
```
import statsmodels.formula.api as smf
```

[90]:
```
model1=smf.ols("Life_expectancy ~␣
↪Country+Year+Status+Adult_Mortality+infant_deaths+Alcohol+percentage_expenditure+Hepatitis_
↪fit()
```

[91]:
```
model1.summary()
```

[91]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:        Life_expectancy   R-squared:                       0.821
Model:                            OLS   Adj. R-squared:                  0.819
Method:                 Least Squares   F-statistic:                     667.2
Date:                Wed, 15 Feb 2023   Prob (F-statistic):               0.00
Time:                        12:27:00   Log-Likelihood:                -8261.0
No. Observations:                2938   AIC:                         1.656e+04
Df Residuals:                    2917   BIC:                         1.669e+04
Df Model:                          20
Covariance Type:            nonrobust
================================================================================
==========
                           coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
----------
Intercept               73.9659     34.624      2.136      0.033       6.075
141.856
Country                  0.0047      0.001      3.505      0.000       0.002
0.007
Year                    -0.0089      0.017     -0.514      0.607      -0.043
0.025
Status                  -1.4795      0.271     -5.463      0.000      -2.010
-0.948
Adult_Mortality         -0.0198      0.001    -24.904      0.000      -0.021
-0.018
infant_deaths            0.0994      0.008     11.848      0.000       0.083
0.116
Alcohol                  0.0706      0.026      2.707      0.007       0.019
0.122
percentage_expenditure   0.0001   8.46e-05      1.191      0.234   -6.52e-05
0.000
Hepatitis_B             -0.0143      0.004     -3.647      0.000      -0.022
-0.007
Measles              -1.853e-05   7.63e-06     -2.429      0.015   -3.35e-05
-3.57e-06
BMI                      0.0423      0.005      8.536      0.000       0.033
0.052
under_five_deaths       -0.0747      0.006    -12.112      0.000      -0.087
-0.063
Polio                    0.0278      0.004      6.225      0.000       0.019
0.037
Total_expenditure        0.0543      0.034      1.584      0.113      -0.013
0.122
Diphtheria               0.0404      0.005      8.611      0.000       0.031
0.050
```

```
HIV_AIDS                  -0.4745      0.018    -26.835      0.000      -0.509
-0.440
GDP                     3.146e-05    1.3e-05      2.422      0.015    5.99e-06
5.69e-05
Population              5.879e-11   1.69e-09      0.035      0.972   -3.25e-09
3.37e-09
thinness_yr               -0.0914      0.026     -3.579      0.000      -0.142
-0.041
Income_composition         5.8194      0.639      9.106      0.000       4.566
7.073
Schooling                  0.6587      0.042     15.763      0.000       0.577
0.741
==============================================================================
Omnibus:                      141.337   Durbin-Watson:                   0.701
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              424.993
Skew:                          -0.179   Prob(JB):                     5.18e-93
Kurtosis:                       4.828   Cond. No.                     2.57e+10
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 2.57e+10. This might indicate that there are
strong multicollinearity or other numerical problems.
"""
```

[93]:
```
pred1 = model1.predict(pd.DataFrame(df))
```

[94]:
```
pred1
```

[94]:
```
0       60.398145
1       61.496432
2       61.581768
3       61.578167
4       61.149089
          …
2933    37.794189
2934    35.917486
2935    49.321652
2936    36.004457
2937    34.984824
Length: 2938, dtype: float64
```

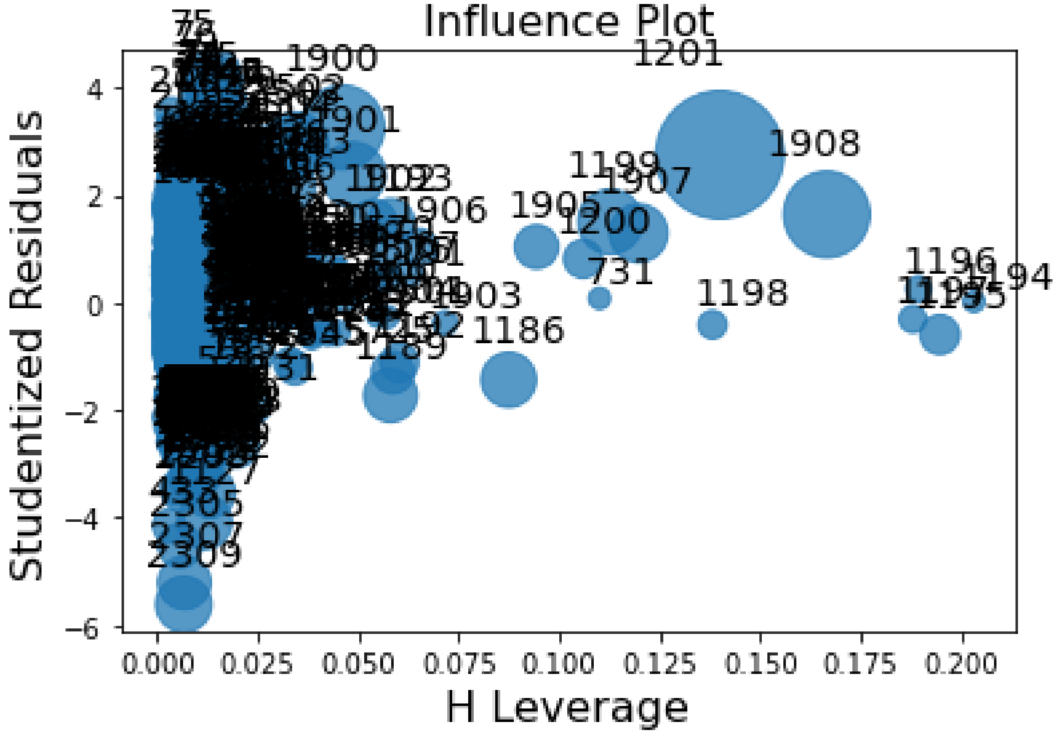
[95]:
```
import statsmodels.api as sm
```

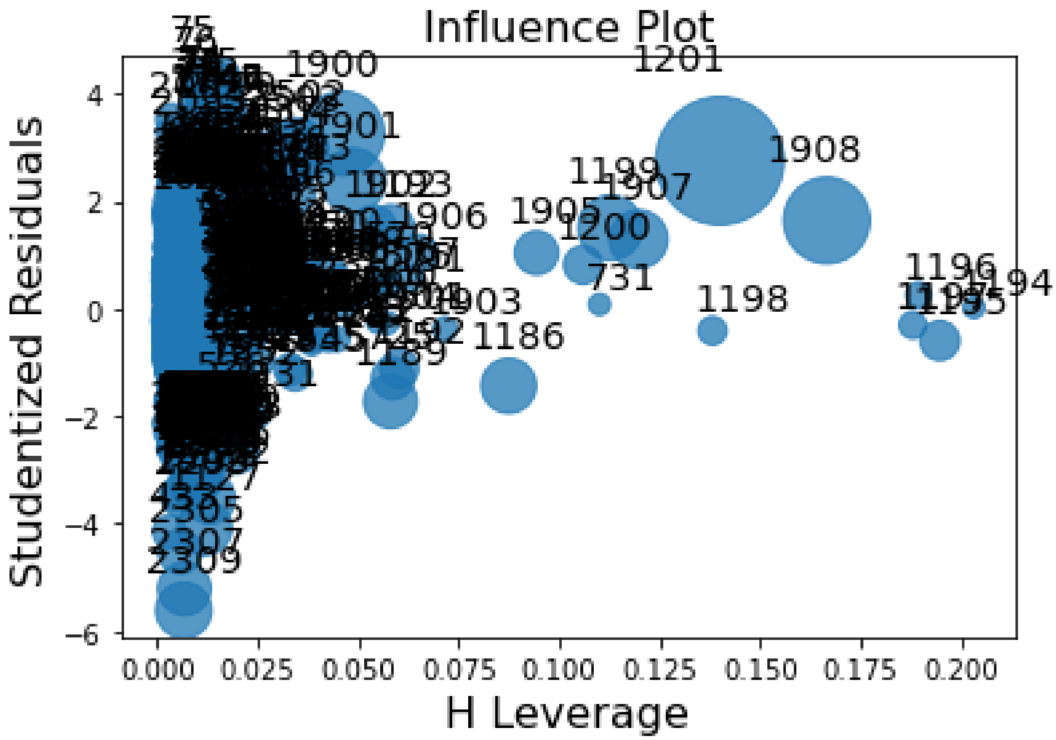
[ ]:
```

```

[100]:
```
sm.graphics.influence_plot(model1)
```

[100]:





```
[97]: df_new = df.drop(df.index[[1187]])
```

```
[111]: model2=smf.ols("Life_expectancy ~
     ↪Country+Year+Status+Adult_Mortality+infant_deaths+Alcohol+percentage_expenditure+Hepatitis_
     ↪fit()
```

```
[112]: model2.summary()
```

```
[112]: <class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:        Life_expectancy   R-squared:                       0.821
Model:                            OLS   Adj. R-squared:                  0.819
Method:                 Least Squares   F-statistic:                     667.2
Date:                Wed, 15 Feb 2023   Prob (F-statistic):               0.00
Time:                        12:48:24   Log-Likelihood:                -8258.3
No. Observations:                2937   AIC:                         1.656e+04
Df Residuals:                    2916   BIC:                         1.668e+04
Df Model:                          20
Covariance Type:            nonrobust
================================================================================
==========
                             coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
----------
Intercept                 73.1925     34.638      2.113      0.035       5.274
141.111
Country                    0.0047      0.001      3.505      0.000       0.002
0.007
Year                      -0.0085      0.017     -0.492      0.623      -0.042
0.025
Status                    -1.4827      0.271     -5.474      0.000      -2.014
-0.952
Adult_Mortality           -0.0198      0.001    -24.880      0.000      -0.021
-0.018
infant_deaths              0.1000      0.008     11.875      0.000       0.083
0.116
Alcohol                    0.0708      0.026      2.717      0.007       0.020
0.122
percentage_expenditure     0.0001   8.46e-05      1.200      0.230   -6.43e-05
0.000
Hepatitis_B               -0.0141      0.004     -3.604      0.000      -0.022
-0.006
```

```
Measles                -1.787e-05   7.67e-06     -2.329      0.020   -3.29e-05
-2.83e-06
BMI                        0.0423      0.005      8.540      0.000       0.033
0.052
under_five_deaths         -0.0752      0.006    -12.132      0.000      -0.087
-0.063
Polio                      0.0277      0.004      6.223      0.000       0.019
0.036
Total_expenditure          0.0547      0.034      1.596      0.111      -0.013
0.122
Diphtheria                 0.0403      0.005      8.580      0.000       0.031
0.050
HIV_AIDS                  -0.4745      0.018    -26.830      0.000      -0.509
-0.440
GDP                     3.137e-05     1.3e-05     2.414      0.016    5.89e-06
5.68e-05
Population              7.308e-10   1.87e-09      0.391      0.696   -2.94e-09
4.4e-09
thinness_yr               -0.0908      0.026     -3.554      0.000      -0.141
-0.041
Income_composition         5.8116      0.639      9.092      0.000       4.558
7.065
Schooling                  0.6586      0.042     15.759      0.000       0.577
0.741
==============================================================================
Omnibus:                      140.948   Durbin-Watson:                   0.701
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              424.171
Skew:                          -0.178   Prob(JB):                     7.81e-93
Kurtosis:                       4.828   Cond. No.                     2.32e+10
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 2.32e+10. This might indicate that there are
strong multicollinearity or other numerical problems.
"""
```

[103]:

```python
pred2 = model2.predict(pd.DataFrame(df_new))
```

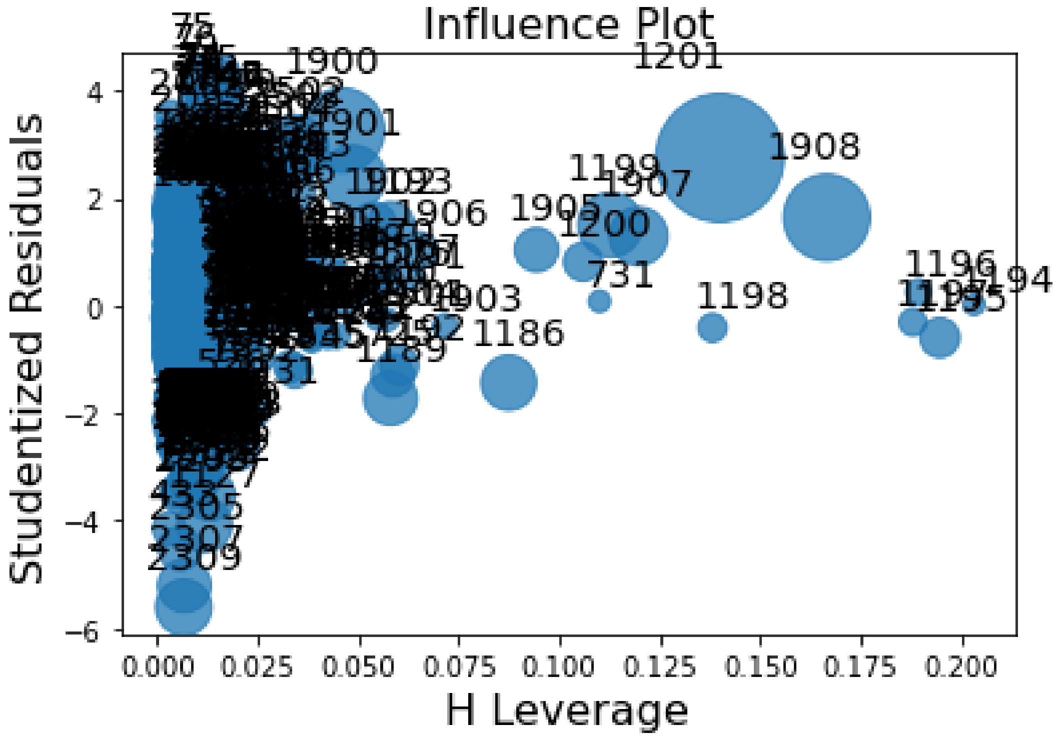
[105]:

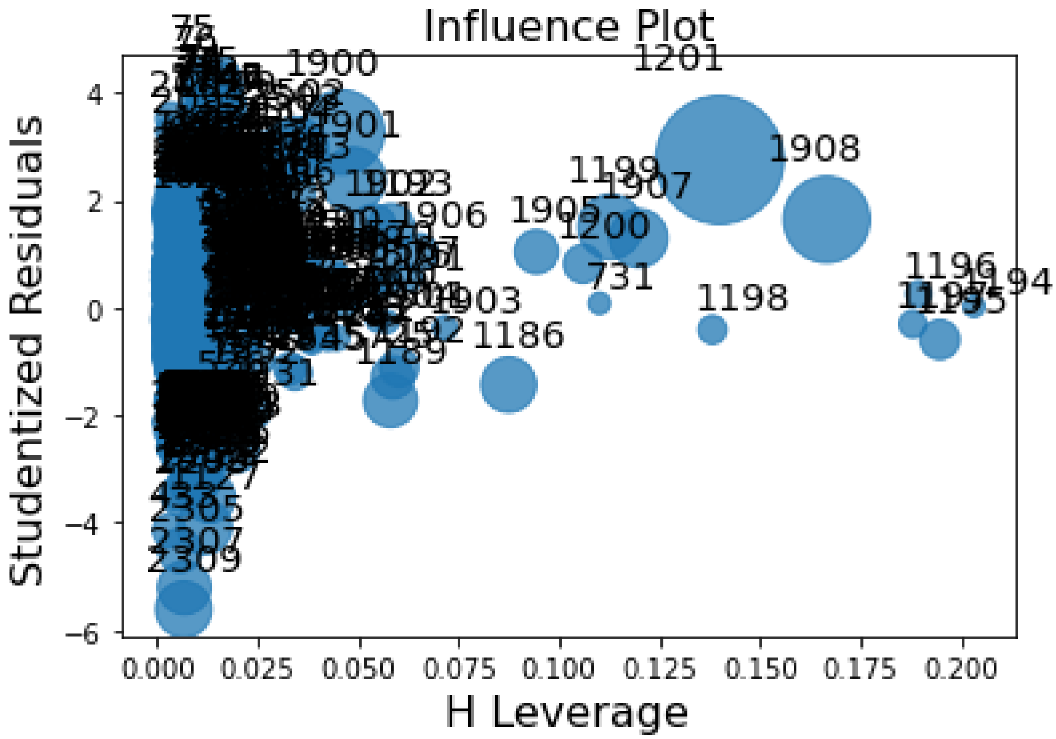
```python
# Error calculation
res1 = df.Life_expectancy - pred2
res_sqr1 = res1 * res1
mse1 = np.mean(res_sqr1)
rmse1 = np.sqrt(mse1)
rmse1
```

[105]: 4.026410527246537

[106]:
```
sm.graphics.influence_plot(model2)
```

[106]:





[115]:
```
model3=smf.ols("Life_expectancy ~
↪Country+Status+Adult_Mortality+infant_deaths+Alcohol+Hepatitis_B+Measles+BMI+under_five_dea
↪fit()
```

[116]:
```
model3.summary()
```

[116]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:        Life_expectancy   R-squared:                       0.820
Model:                            OLS   Adj. R-squared:                  0.819
Method:                 Least Squares   F-statistic:                     833.4
Date:                Wed, 15 Feb 2023   Prob (F-statistic):               0.00
Time:                        12:50:55   Log-Likelihood:                -8260.7
No. Observations:                2937   AIC:                         1.656e+04
Df Residuals:                    2920   BIC:                         1.666e+04
Df Model:                          16
Covariance Type:            nonrobust
================================================================================
======
                          coef    std err          t      P>|t|      [0.025
0.975]
```

```
--------------------------------------------------------------------------------
------
Intercept             56.5597      0.638     88.621      0.000      55.308
57.811
Country                0.0048      0.001      3.550      0.000       0.002
0.007
Status                -1.5784      0.267     -5.917      0.000      -2.101
-1.055
Adult_Mortality       -0.0198      0.001    -25.052      0.000      -0.021
-0.018
infant_deaths          0.1006      0.008     12.047      0.000       0.084
0.117
Alcohol                0.0793      0.026      3.094      0.002       0.029
0.130
Hepatitis_B           -0.0148      0.004     -3.796      0.000      -0.022
-0.007
Measles            -1.844e-05   7.63e-06     -2.416      0.016   -3.34e-05
-3.47e-06
BMI                    0.0426      0.005      8.620      0.000       0.033
0.052
under_five_deaths     -0.0756      0.006    -12.223      0.000      -0.088
-0.063
Polio                  0.0277      0.004      6.209      0.000       0.019
0.036
Diphtheria             0.0407      0.005      8.698      0.000       0.032
0.050
HIV_AIDS              -0.4707      0.018    -26.863      0.000      -0.505
-0.436
GDP                 4.423e-05   6.69e-06      6.611      0.000    3.11e-05
5.74e-05
thinness_yr           -0.0970      0.025     -3.827      0.000      -0.147
-0.047
Income_composition     5.6396      0.630      8.948      0.000       4.404
6.875
Schooling              0.6608      0.042     15.887      0.000       0.579
0.742
==============================================================================
Omnibus:                      134.445   Durbin-Watson:                   0.697
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              393.372
Skew:                          -0.171   Prob(JB):                     3.81e-86
Kurtosis:                       4.760   Cond. No.                     1.38e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 1.38e+05. This might indicate that there are
```

```
strong multicollinearity or other numerical problems.
"""
```

[127]:
```python
from sklearn.model_selection import train_test_split
```

[ ]:
```python

```

[123]:
```python
final_model=smf.ols("Life_expectancy ~
↪Country+Year+Status+Adult_Mortality+infant_deaths+Alcohol+percentage_expenditure+Hepatitis_
↪fit()
```

[124]:
```python
final_model.summary()
```

[124]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:        Life_expectancy   R-squared:                       0.821
Model:                            OLS   Adj. R-squared:                  0.819
Method:                 Least Squares   F-statistic:                     667.2
Date:                Wed, 15 Feb 2023   Prob (F-statistic):               0.00
Time:                        12:53:38   Log-Likelihood:                -8258.3
No. Observations:                2937   AIC:                         1.656e+04
Df Residuals:                    2916   BIC:                         1.668e+04
Df Model:                          20
Covariance Type:            nonrobust
================================================================================
==========
                             coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
----------
Intercept                 73.1925     34.638      2.113      0.035       5.274
141.111
Country                    0.0047      0.001      3.505      0.000       0.002
0.007
Year                      -0.0085      0.017     -0.492      0.623      -0.042
0.025
Status                    -1.4827      0.271     -5.474      0.000      -2.014
-0.952
Adult_Mortality           -0.0198      0.001    -24.880      0.000      -0.021
-0.018
infant_deaths              0.1000      0.008     11.875      0.000       0.083
0.116
Alcohol                    0.0708      0.026      2.717      0.007       0.020
0.122
percentage_expenditure     0.0001   8.46e-05      1.200      0.230   -6.43e-05
```

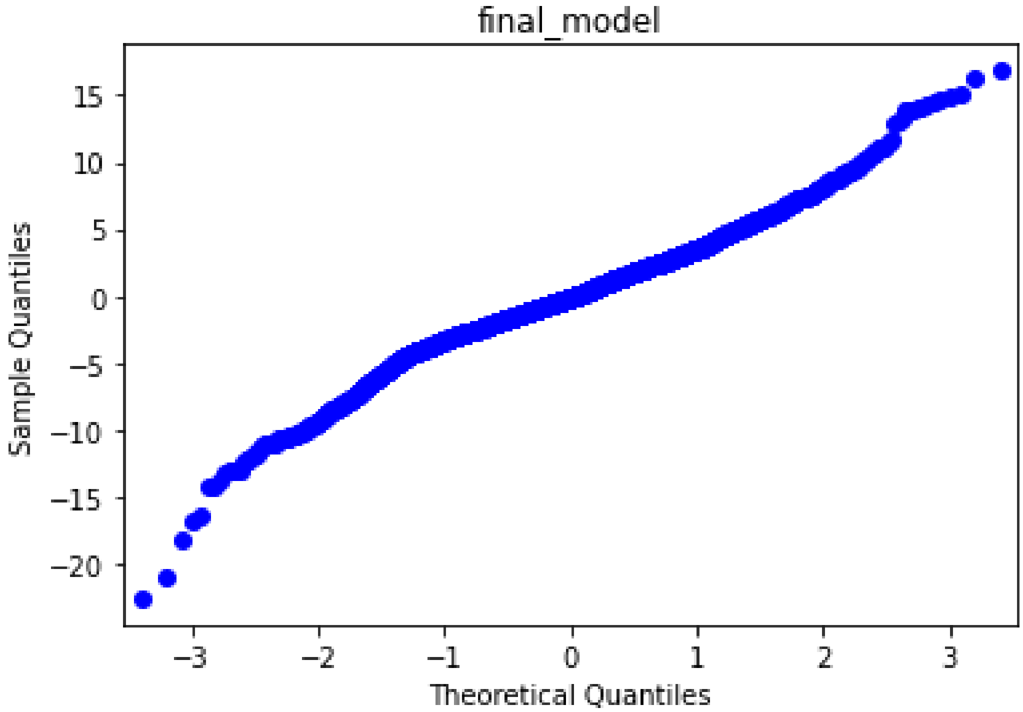
```
0.000
Hepatitis_B             -0.0141      0.004     -3.604      0.000      -0.022
-0.006
Measles              -1.787e-05   7.67e-06     -2.329      0.020   -3.29e-05
-2.83e-06
BMI                      0.0423      0.005      8.540      0.000       0.033
0.052
under_five_deaths       -0.0752      0.006    -12.132      0.000      -0.087
-0.063
Polio                    0.0277      0.004      6.223      0.000       0.019
0.036
Total_expenditure        0.0547      0.034      1.596      0.111      -0.013
0.122
Diphtheria               0.0403      0.005      8.580      0.000       0.031
0.050
HIV_AIDS                -0.4745      0.018    -26.830      0.000      -0.509
-0.440
GDP                   3.137e-05    1.3e-05      2.414      0.016    5.89e-06
5.68e-05
Population            7.308e-10   1.87e-09      0.391      0.696   -2.94e-09
4.4e-09
thinness_yr             -0.0908      0.026     -3.554      0.000      -0.141
-0.041
Income_composition       5.8116      0.639      9.092      0.000       4.558
7.065
Schooling                0.6586      0.042     15.759      0.000       0.577
0.741
==============================================================================
Omnibus:                      140.948   Durbin-Watson:                   0.701
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              424.171
Skew:                          -0.178   Prob(JB):                     7.81e-93
Kurtosis:                       4.828   Cond. No.                     2.32e+10
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 2.32e+10. This might indicate that there are
strong multicollinearity or other numerical problems.
"""
```

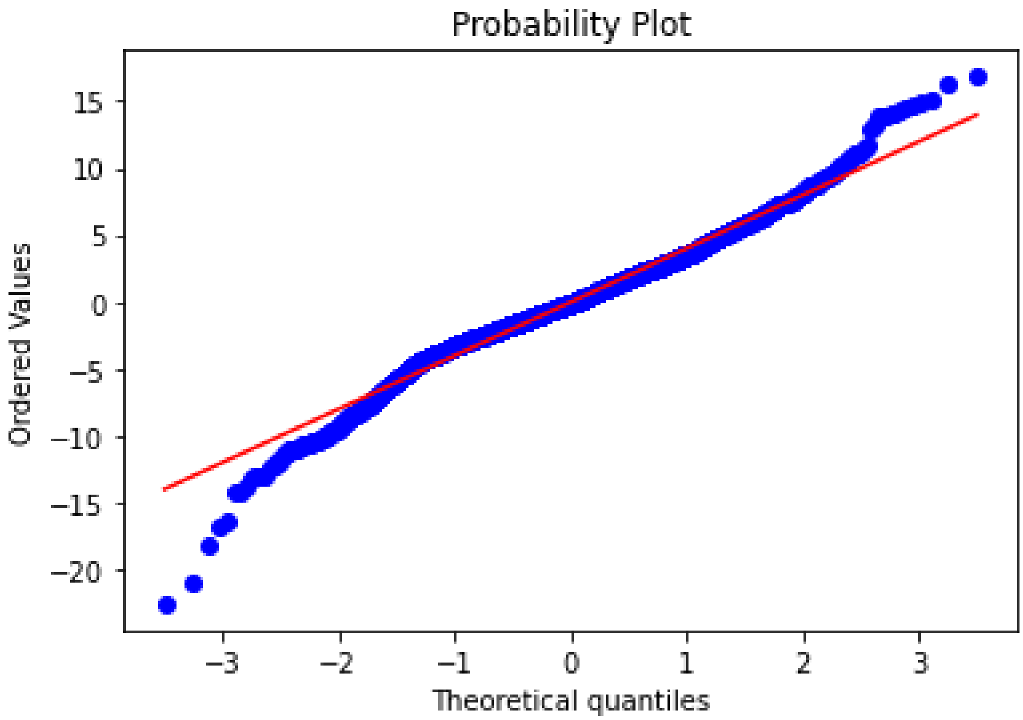
[129]:
```
res = final_model.resid
sm.qqplot(res);plt.title("final_model")
plt.show()
```

```
C:\Users\Ravi\anaconda3\lib\site-packages\statsmodels\graphics\gofplots.py:993:
UserWarning: marker is redundantly defined by the 'marker' keyword argument and
the fmt string "bo" (-> marker='o'). The keyword argument will take precedence.
```

```
ax.plot(x, y, fmt, **plot_style)
```

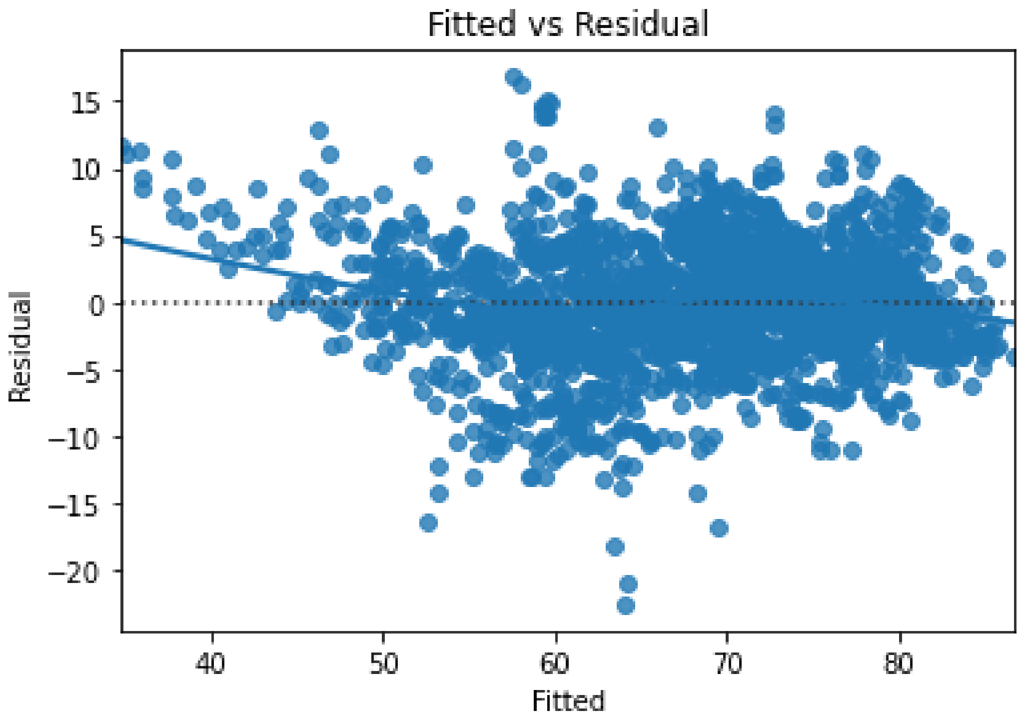


[134]:
```
from scipy import stats
import pylab
# Q-Q plot
stats.probplot(res, dist = "norm", plot = pylab)
plt.show()
```



[136]:
```
pred = final_model.predict(df_new)
```

[137]:
```
sns.residplot(x = pred, y = df_new.Life_expectancy, lowess = True)
plt.xlabel('Fitted')
plt.ylabel('Residual')
plt.title('Fitted vs Residual')
plt.show()
```



[138]:
```
from sklearn.model_selection import train_test_split
```

[139]:
```
df_train,df_test=train_test_split(df_new,test_size=0.2,random_state=10)
```

[140]:
```
model_train=smf.ols("Life_expectancy ~␣
↪Country+Year+Status+Adult_Mortality+infant_deaths+Alcohol+percentage_expenditure+Hepatitis_
↪fit()
```

[141]:
```
test_predict=final_model.predict(df_test)
```

[ ]:
```
```

[143]:
```
test_residual = test_predict - df_test.Life_expectancy
```

[144]:
```
test_rmse=np.sqrt(np.mean(test_residual*test_residual))
```

[145]:
```
test_rmse
```

[145]: 3.943450814923155

[146]:
```
train_predict=model_train.predict(df_train)
```

[148]:
```
train_residual=train_predict - df_train.Life_expectancy
```

[149]:
```
train_rmse=np.sqrt(np.mean(train_residual*train_residual))
```

[150]:
```
train_rmse
```

[150]: 4.043384141685655

[151]:
```
### RIDGE REGRESSION ###
from sklearn.linear_model import Ridge
rm = Ridge(alpha = 5, normalize = True)
```

[159]:
```
rm.fit(df_new.iloc[:,df_new.columns!='Life_expectancy'], df_new.Life_expectancy)
```

```
C:\Users\Ravi\AppData\Roaming\Python\Python39\site-
packages\sklearn\linear_model\_base.py:141: FutureWarning: 'normalize' was
deprecated in version 1.0 and will be removed in 1.2.
If you wish to scale the data, use Pipeline with a StandardScaler in a
preprocessing stage. To reproduce the previous behavior:

from sklearn.pipeline import make_pipeline

model = make_pipeline(StandardScaler(with_mean=False), Ridge())

If you wish to pass a sample_weight parameter, you need to pass it as a fit
parameter to each step of the pipeline as follows:

kwargs = {s[0] + '__sample_weight': sample_weight for s in model.steps}
model.fit(X, y, **kwargs)

Set parameter alpha to: original_alpha * n_samples.
  warnings.warn(
```

[159]: Ridge(alpha=5, normalize=True)

[161]:
```
df_new=df_new.iloc[:,[3,0,1,2,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21]]
```

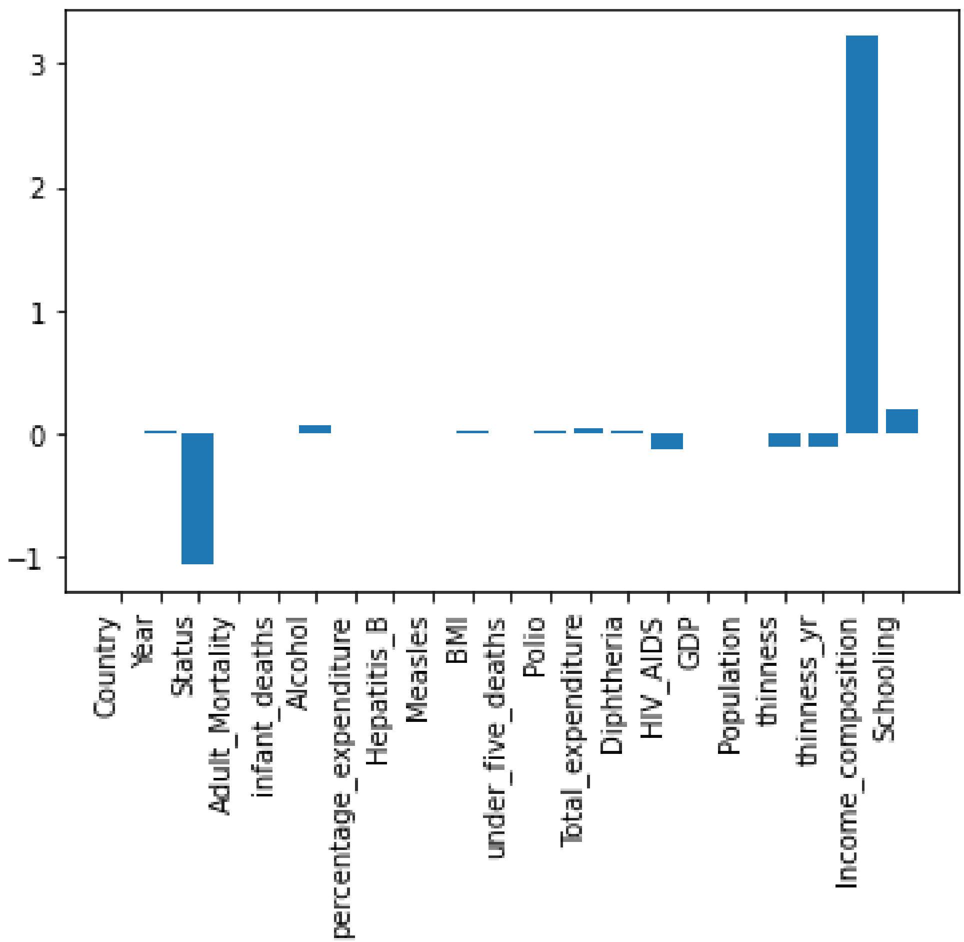
[156]:
```
# Coefficients values for all the independent vairbales
rm.coef_
rm.intercept_
```

[156]: -3.1477954992781036

[164]:
```
plt.bar(height = pd.Series(rm.coef_), x = pd.Series(df_new.columns[1:]));plt.
↪xticks(rotation=90,ha='right')
```

[164]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20],
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
```

```
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, '')])
```



[165]:
```
rm.alpha
```

[165]: 5

[166]:
```
pred_rm = rm.predict(df_new.iloc[:, 1:])
```

[168]:
```
# Adjusted r-square
rm.score(df_new.iloc[:, 1:], df_new.Life_expectancy)
```

[168]: 0.5725269210776764

[169]:
```
# RMSE
np.sqrt(np.mean((pred_rm - df_new.Life_expectancy)**2))
```

[169]: 6.216213025877878

[170]:
```
from sklearn.linear_model import Lasso

lasso = Lasso(alpha = 0.13, normalize = True)
```

[171]:
```
lasso.fit(df_new.iloc[:, 1:], df_new.Life_expectancy)
```

```
C:\Users\Ravi\AppData\Roaming\Python\Python39\site-
packages\sklearn\linear_model\_base.py:141: FutureWarning: 'normalize' was
deprecated in version 1.0 and will be removed in 1.2.
If you wish to scale the data, use Pipeline with a StandardScaler in a
preprocessing stage. To reproduce the previous behavior:

from sklearn.pipeline import make_pipeline

model = make_pipeline(StandardScaler(with_mean=False), Lasso())

If you wish to pass a sample_weight parameter, you need to pass it as a fit
parameter to each step of the pipeline as follows:

kwargs = {s[0] + '__sample_weight': sample_weight for s in model.steps}
model.fit(X, y, **kwargs)

Set parameter alpha to: original_alpha * np.sqrt(n_samples).
  warnings.warn(
```

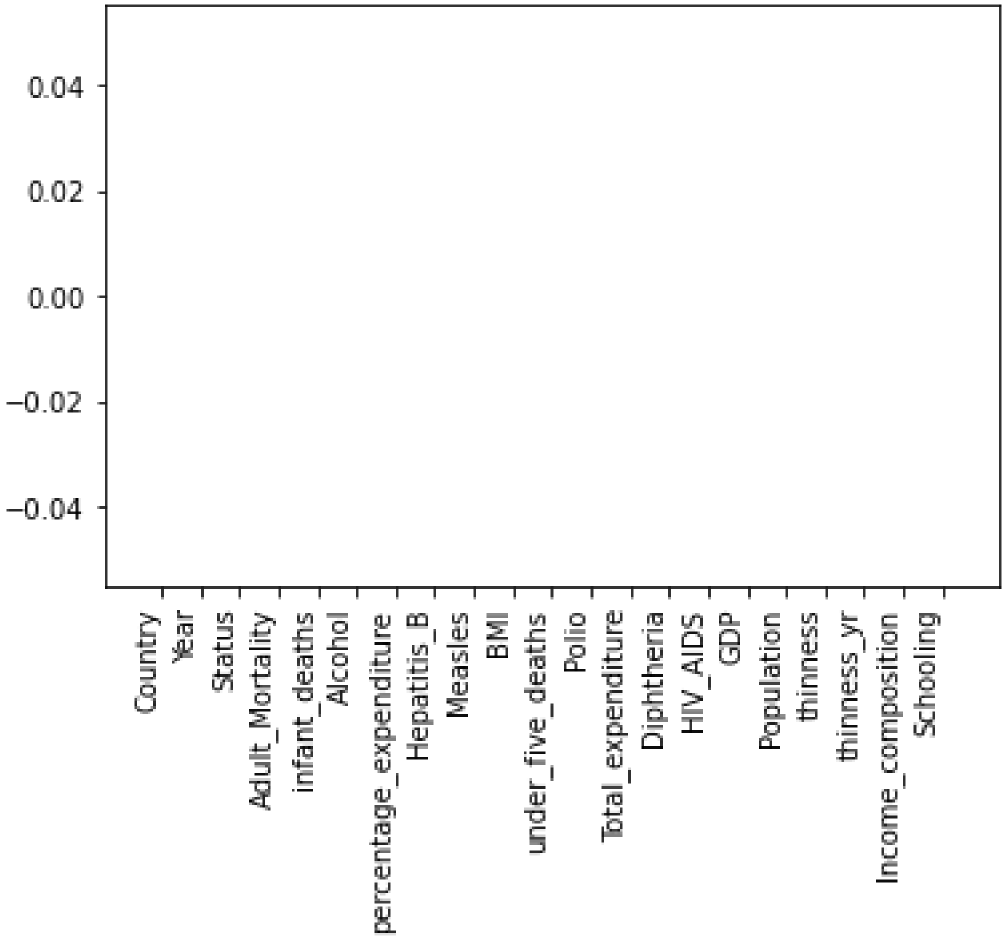
[171]: Lasso(alpha=0.13, normalize=True)

[172]:
```
# Coefficient values for all independent variables#
lasso.coef_
lasso.intercept_
```

[172]: 69.22534876300303

[175]:
```
plt.bar(height = pd.Series(lasso.coef_), x = pd.Series(df_new.columns[1:]));plt.
↪xticks(rotation=90,ha='right')
```

[175]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20],
 [Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, ''),
  Text(0, 0, '')])
```



[146]:
```
lasso.alpha
```

[146]: 0.13

[147]:
```
pred_lasso = lasso.predict(df.iloc[:, 1:])
```

[148]:
```
# Adjusted r-square
lasso.score(df.iloc[:, 1:], df.price)
```

[148]: 0.7103195992245424

[149]:
```
# RMSE
np.sqrt(np.mean((pred_lasso - df.price)**2))
```

[149]: 305.25179729893966

[ ]:
```

```