Unemployment dataset

In [1]:
```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression,LogisticRegression,Lasso,Rid
```

In [2]:
```
df=pd.read_csv(r"C:\Users\USER\Desktop\Unemployment Rate, seas. adj..csv")
df.fillna(0,inplace=True)
df
```

Out[2]:

	Unnamed: 0	Advanced Economies	Argentina	Australia	Austria	Belgium	Bulgaria	Bahrain	Belai
0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0000
1	1994.0	7.818759	0.000000	9.703104	6.545480	9.753783	14.065830	0.0	0.0000
2	1995.0	7.352965	0.000000	8.467310	6.589767	9.673502	11.385830	0.0	0.0000
3	1996.0	7.321421	0.000000	8.512719	7.033851	9.544409	11.061670	0.0	0.0000
4	1997.0	7.022442	0.000000	8.358484	7.103283	9.212289	14.045830	0.0	0.0000
5	1998.0	6.646781	0.000000	7.677949	7.184796	9.340881	12.203330	0.0	0.0000
6	1999.0	6.324369	0.000000	6.867747	6.645249	8.411080	13.782500	0.0	0.0000
7	2000.0	5.844973	0.000000	6.275520	5.803798	6.875592	18.129170	0.0	0.0000
8	2001.0	5.991557	0.000000	6.757151	6.094126	6.589738	17.508330	0.0	0.0000
9	2002.0	6.550974	22.425370	6.361517	6.871619	7.526965	17.426670	0.0	0.0000
10	2003.0	6.762680	17.216580	5.927986	7.014325	8.182808	14.259170	0.0	0.0000
11	2004.0	6.526365	13.622010	5.394249	7.067989	8.390290	12.670000	0.0	0.0000
12	2005.0	6.196061	11.560380	5.034545	7.272197	8.441850	11.450830	0.0	0.0000
13	2006.0	5.748805	10.150660	4.775160	6.779983	8.259000	9.602500	16.0	0.0000
14	2007.0	5.506039	8.445448	4.373901	6.212382	7.488009	7.736667	5.6	0.0000
15	2008.0	5.913973	7.856474	4.238160	5.900164	6.964916	6.302500	3.7	0.0000
16	2009.0	8.137286	8.667093	5.565805	7.277309	7.986718	7.584167	4.0	0.0000
17	2010.0	8.446151	7.745606	5.203839	6.924717	8.371833	9.474167	3.6	0.0000
18	2011.0	8.157150	7.154081	5.081351	6.724666	7.208788	9.595000	4.0	0.0000
19	2012.0	8.281009	7.214789	5.221220	6.980560	7.632016	11.089170	3.7	0.0000
20	2013.0	8.234926	7.075874	5.665686	7.614778	8.554433	11.312500	4.3	0.0000
21	2014.0	7.600729	7.268765	6.080297	8.365268	8.662285	11.160000	3.8	0.0000
22	2015.0	6.986622	6.607571	6.053246	9.102619	8.655595	10.059170	3.5	0.0000
23	2016.0	6.510643	8.462791	5.710455	9.063808	7.850004	8.687500	4.3	0.0000
24	2017.0	5.903856	8.337106	5.589480	8.516667	7.096994	7.199167	4.1	5.6727
25	2018.0	5.352979	9.207210	5.298026	7.713245	5.958490	6.175000	4.3	4.8274
26	2019.0	5.033913	9.798736	5.167842	7.355347	5.365955	5.650833	4.7	4.2248
27	2020.0	6.885005	11.520220	6.489594	10.002220	5.528885	7.359167	5.9	4.1033
28	2021.0	5.816167	8.719054	5.089850	7.990778	6.273146	5.524167	0.0	3.9251
29	2022.0	4.636630	6.827820	3.695052	6.301216	5.579375	4.497500	0.0	3.5998
30	2023.0	0.000000	0.000000	0.000000	6.176443	0.000000	0.000000	0.0	0.0000

31 rows × 80 columns

In [3]:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31 entries, 0 to 30
Data columns (total 80 columns):
 #   Column                                             Non-Null Count  Dtype
---  ------                                             --------------  -----
 0   Unnamed: 0                                         31 non-null     float6
4
 1   Advanced Economies                                 31 non-null     float6
4
 2   Argentina                                          31 non-null     float6
4
 3   Australia                                          31 non-null     float6
4
 4   Austria                                            31 non-null     float6
4
 5   Belgium                                            31 non-null     float6
4
 6   Bulgaria                                           31 non-null     float6
4
 7   Bahrain                                            31 non-null     float6
4
 8   Belarus                                            31 non-null     float6
4
 9   Brazil                                             31 non-null     float6
4
 10  Canada                                             31 non-null     float6
4
 11  Switzerland                                        31 non-null     float6
4
 12  Chile                                              31 non-null     float6
4
 13  Colombia                                           31 non-null     float6
4
 14  Cyprus                                             31 non-null     float6
4
 15  Czech Republic                                     31 non-null     float6
4
 16  Germany                                            31 non-null     float6
4
 17  Denmark                                            31 non-null     float6
4
 18  Dominican Republic                                 31 non-null     float6
4
 19  Algeria                                            31 non-null     float6
4
 20  EMDE East Asia & Pacific                           31 non-null     float6
4
 21  EMDE Europe & Central Asia                         31 non-null     float6
4
 22  Ecuador                                            31 non-null     float6
4
 23  Egypt, Arab Rep.                                   31 non-null     float6
4
 24  Emerging Market and Developing Economies (EMDEs)   31 non-null     float6
```

```
4
 25  Spain                                       31 non-null     float6
4
 26  Estonia                                     31 non-null     float6
4
 27  Finland                                     31 non-null     float6
4
 28  France                                      31 non-null     float6
4
 29  United Kingdom                              31 non-null     float6
4
 30  Greece                                      31 non-null     float6
4
 31  High Income Countries                       31 non-null     float6
4
 32  Hong Kong SAR, China                        31 non-null     float6
4
 33  Croatia                                     31 non-null     float6
4
 34  Hungary                                     31 non-null     float6
4
 35  Ireland                                     31 non-null     float6
4
 36  Iceland                                     31 non-null     float6
4
 37  Israel                                      31 non-null     float6
4
 38  Italy                                       31 non-null     float6
4
 39  Japan                                       31 non-null     float6
4
 40  Korea, Rep.                                 31 non-null     float6
4
 41  EMDE Latin America & Caribbean              31 non-null     float6
4
 42  Low-Income Countries (LIC)                  31 non-null     float6
4
 43  Sri Lanka                                   31 non-null     float6
4
 44  Lithuania                                   31 non-null     float6
4
 45  Luxembourg                                  31 non-null     float6
4
 46  Latvia                                      31 non-null     float6
4
 47  Morocco                                     31 non-null     float6
4
 48  Mexico                                      31 non-null     float6
4
 49  Middle-Income Countries (MIC)               31 non-null     float6
4
 50  North Macedonia                             31 non-null     float6
4
 51  Malta                                       31 non-null     float6
4
 52  EMDE Middle East & N. Africa                31 non-null     float6
```

```
4
 53  Netherlands                                         31 non-null     float6
4
 54  Norway                                              31 non-null     float6
4
 55  New Zealand                                         31 non-null     float6
4
 56  Pakistan                                            31 non-null     float6
4
 57  Peru                                                31 non-null     float6
4
 58  Philippines                                         31 non-null     float6
4
 59  Poland                                              31 non-null     float6
4
 60  Portugal                                            31 non-null     float6
4
 61  Romania                                             31 non-null     float6
4
 62  Russian Federation                                  31 non-null     float6
4
 63  EMDE South Asia                                     31 non-null     float6
4
 64  Saudi Arabia                                        31 non-null     float6
4
 65  Singapore                                           31 non-null     float6
4
 66  EMDE Sub-Saharan Africa                             31 non-null     float6
4
 67  Slovakia                                            31 non-null     float6
4
 68  Slovenia                                            31 non-null     float6
4
 69  Sweden                                              31 non-null     float6
4
 70  Thailand                                            31 non-null     float6
4
 71  Tunisia                                             31 non-null     float6
4
 72  Turkey                                              31 non-null     float6
4
 73  Taiwan, China                                       31 non-null     float6
4
 74  Uruguay                                             31 non-null     float6
4
 75  United States                                       31 non-null     float6
4
 76  Venezuela, RB                                       31 non-null     float6
4
 77  Vietnam                                             31 non-null     float6
4
 78  World (WBG members)                                 31 non-null     float6
4
 79  South Africa                                        31 non-null     float6
4
dtypes: float64(80)
```

```
memory usage: 10.5 KB
```

In [4]:
```
df1=df.dropna()
```

Out[4]:

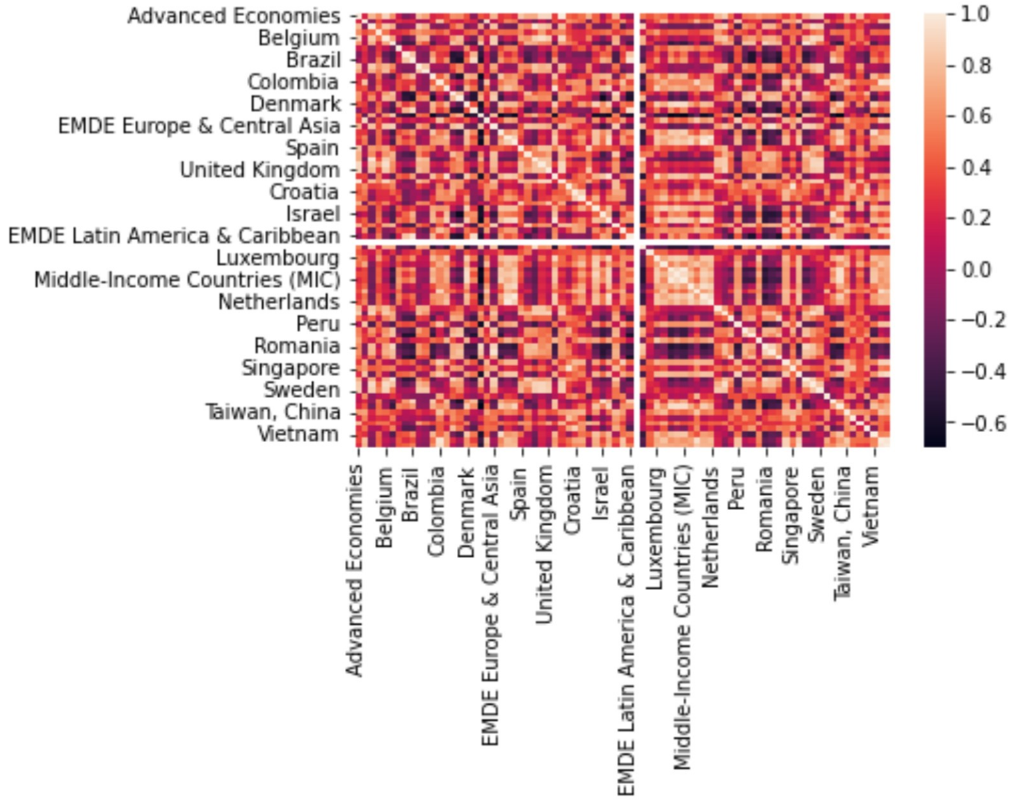
	Unnamed: 0	Advanced Economies	Argentina	Australia	Austria	Belgium	Bulgaria	Bahrain	Belaı
0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0000
1	1994.0	7.818759	0.000000	9.703104	6.545480	9.753783	14.065830	0.0	0.0000
2	1995.0	7.352965	0.000000	8.467310	6.589767	9.673502	11.385830	0.0	0.0000
3	1996.0	7.321421	0.000000	8.512719	7.033851	9.544409	11.061670	0.0	0.0000
4	1997.0	7.022442	0.000000	8.358484	7.103283	9.212289	14.045830	0.0	0.0000
5	1998.0	6.646781	0.000000	7.677949	7.184796	9.340881	12.203330	0.0	0.0000
6	1999.0	6.324369	0.000000	6.867747	6.645249	8.411080	13.782500	0.0	0.0000
7	2000.0	5.844973	0.000000	6.275520	5.803798	6.875592	18.129170	0.0	0.0000
8	2001.0	5.991557	0.000000	6.757151	6.094126	6.589738	17.508330	0.0	0.0000
9	2002.0	6.550974	22.425370	6.361517	6.871619	7.526965	17.426670	0.0	0.0000
10	2003.0	6.762680	17.216580	5.927986	7.014325	8.182808	14.259170	0.0	0.0000
11	2004.0	6.526365	13.622010	5.394249	7.067989	8.390290	12.670000	0.0	0.0000
12	2005.0	6.196061	11.560380	5.034545	7.272197	8.441850	11.450830	0.0	0.0000
13	2006.0	5.748805	10.150660	4.775160	6.779983	8.259000	9.602500	16.0	0.0000
14	2007.0	5.506039	8.445448	4.373901	6.212382	7.488009	7.736667	5.6	0.0000
15	2008.0	5.913973	7.856474	4.238160	5.900164	6.964916	6.302500	3.7	0.0000
16	2009.0	8.137286	8.667093	5.565805	7.277309	7.986718	7.584167	4.0	0.0000
17	2010.0	8.446151	7.745606	5.203839	6.924717	8.371833	9.474167	3.6	0.0000
18	2011.0	8.157150	7.154081	5.081351	6.724666	7.208788	9.595000	4.0	0.0000
19	2012.0	8.281009	7.214789	5.221220	6.980560	7.632016	11.089170	3.7	0.0000
20	2013.0	8.234926	7.075874	5.665686	7.614778	8.554433	11.312500	4.3	0.0000
21	2014.0	7.600729	7.268765	6.080297	8.365268	8.662285	11.160000	3.8	0.0000
22	2015.0	6.986622	6.607571	6.053246	9.102619	8.655595	10.059170	3.5	0.0000
23	2016.0	6.510643	8.462791	5.710455	9.063808	7.850004	8.687500	4.3	0.0000
24	2017.0	5.903856	8.337106	5.589480	8.516667	7.096994	7.199167	4.1	5.6727
25	2018.0	5.352979	9.207210	5.298026	7.713245	5.958490	6.175000	4.3	4.8274
26	2019.0	5.033913	9.798736	5.167842	7.355347	5.365955	5.650833	4.7	4.2248
27	2020.0	6.885005	11.520220	6.489594	10.002220	5.528885	7.359167	5.9	4.1033
28	2021.0	5.816167	8.719054	5.089850	7.990778	6.273146	5.524167	0.0	3.9251
29	2022.0	4.636630	6.827820	3.695052	6.301216	5.579375	4.497500	0.0	3.5998
30	2023.0	0.000000	0.000000	0.000000	6.176443	0.000000	0.000000	0.0	0.0000

31 rows × 80 columns

In [5]:

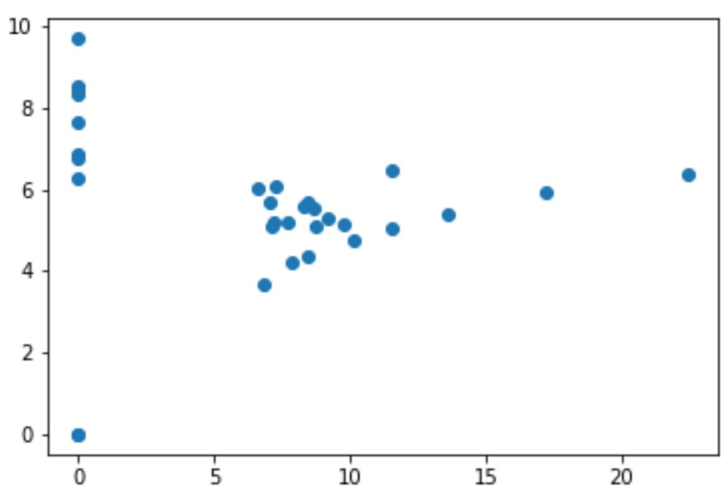
In [6]:

Out[6]: <AxesSubplot:>



In [8]:

Out[8]: [<matplotlib.lines.Line2D at 0x18129dfcb20>]



In [ ]:

In [9]:

```python
x=df1.drop(["Australia"],axis=1)
y=df1["Australia"]
```

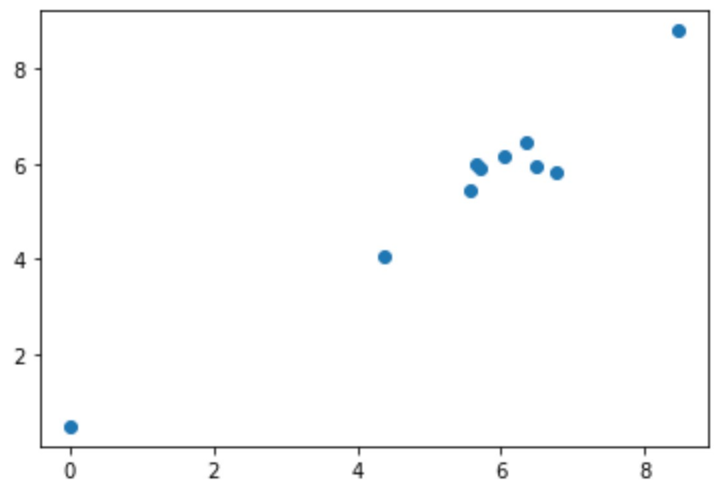
Linear

In [10]:
```
li=LinearRegression()
```

Out[10]:
```
LinearRegression()
```

In [11]:
```
prediction=li.predict(x_test)
```

Out[11]:
```
<matplotlib.collections.PathCollection at 0x1812a1c2f70>
```



In [12]:

In [13]:

Out[13]:
```
0.000000     10
22.425370     1
8.719054      1
11.520220     1
9.798736      1
9.207210      1
8.337106      1
8.462791      1
6.607571      1
7.268765      1
7.075874      1
7.214789      1
7.154081      1
7.745606      1
8.667093      1
7.856474      1
8.445448      1
10.150660     1
11.560380     1
13.622010     1
17.216580     1
6.827820      1
Name: Argentina, dtype: int64
```

In [14]:
```
df1.loc[df1["Argentina"]<1.40,"Argentina"]=1
df1.loc[df1["Argentina"]>1.40,"Argentina"]=2
```

Out[14]:
```
2.0    21
1.0    10
Name: Argentina, dtype: int64
```

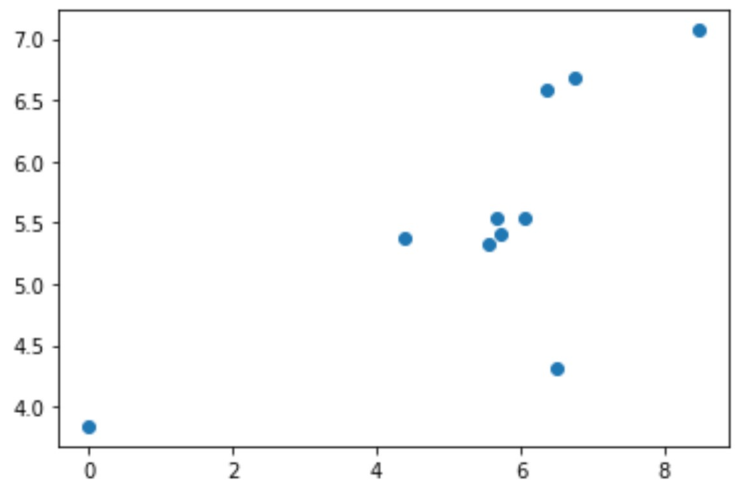
Lasso

In [15]:
```
la=Lasso(alpha=5)
```

Out[15]:
```
Lasso(alpha=5)
```

In [16]:
```
prediction1=la.predict(x_test)
```

Out[16]:
```
<matplotlib.collections.PathCollection at 0x1812a22cbe0>
```



In [17]:

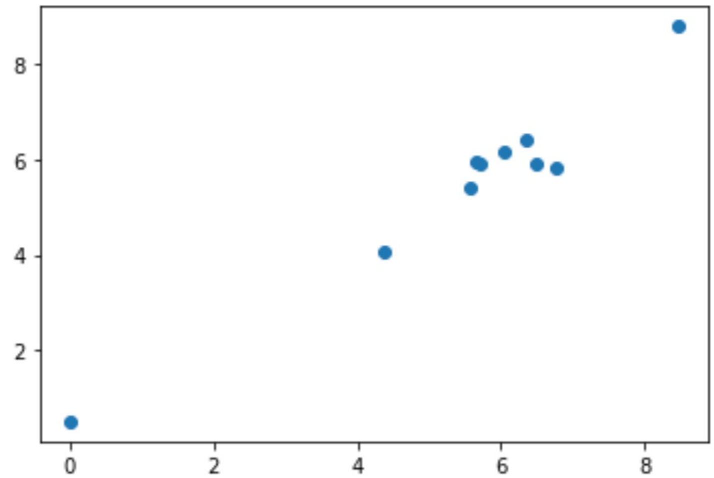
Ridge

In [18]:
```
rr=Ridge(alpha=1)
```

Out[18]:
```
Ridge(alpha=1)
```

In [19]:
```
prediction2=rr.predict(x_test)
```

Out[19]: <matplotlib.collections.PathCollection at 0x1812a28b100>



In [20]:

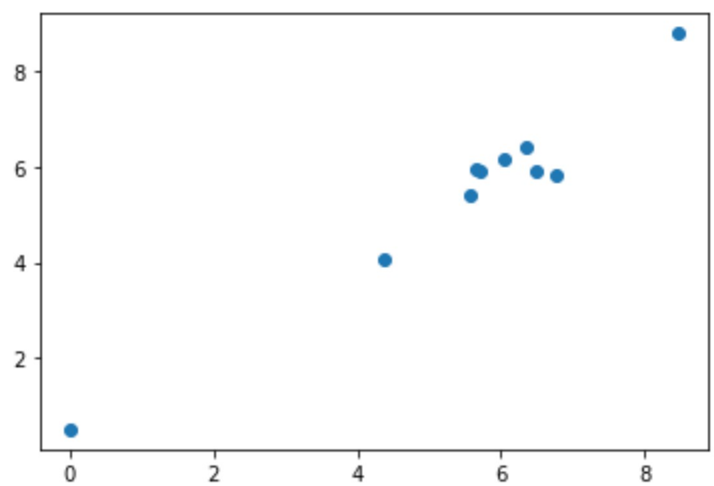
ElasticNet

In [21]:
```
en=ElasticNet()
en.fit(x_train,y_train)
```

Out[21]: ElasticNet()

In [22]:
```
prediction2=rr.predict(x_test)
```

Out[22]: <matplotlib.collections.PathCollection at 0x1812b33c820>



In [23]:

In [24]:
```
print(rr.score(x_test,y_test))
```

```
0.9589303894434331
```

Out[24]:
```
0.9999909096616653
```

Logistic

In [25]:
```
g={"Argentina":{1.0:"Low",2.0:"High"}}
df1=df1.replace(g)
```

Out[25]:
```
High     21
Low      10
Name: Argentina, dtype: int64
```

In [26]:
```
x=df1.drop(["Argentina"],axis=1)
y=df1["Argentina"]
```

In [27]:
```
lo=LogisticRegression()
```

Out[27]:
```
LogisticRegression()
```

In [28]:
```
prediction3=lo.predict(x_test)
```

Out[28]:
```
<matplotlib.collections.PathCollection at 0x1812b3bc340>
```

In [29]:

Random Forest

In [30]:
```
from sklearn.ensemble import RandomForestClassifier
```

In [31]:
```
g1={"Argentina":{"Low":1.0,"High":2.0}}
df1=df1.replace(g1)
```

In [32]:
```
x=df1.drop(["Argentina"],axis=1)
y=df1["Argentina"]
```

In [33]:
```
rfc=RandomForestClassifier()
```

Out[33]:
```
RandomForestClassifier()
```

In [34]:
```
parameter={
    'max_depth':[1,2,4,5,6],
    'min_samples_leaf':[5,10,15,20,25],
    'n_estimators':[10,20,30,40,50]
```

In [35]:
```
grid_search=GridSearchCV(estimator=rfc,param_grid=parameter,cv=2,scoring="accu
```

Out[35]:
```
GridSearchCV(cv=2, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [1, 2, 4, 5, 6],
                         'min_samples_leaf': [5, 10, 15, 20, 25],
                         'n_estimators': [10, 20, 30, 40, 50]},
             scoring='accuracy')
```

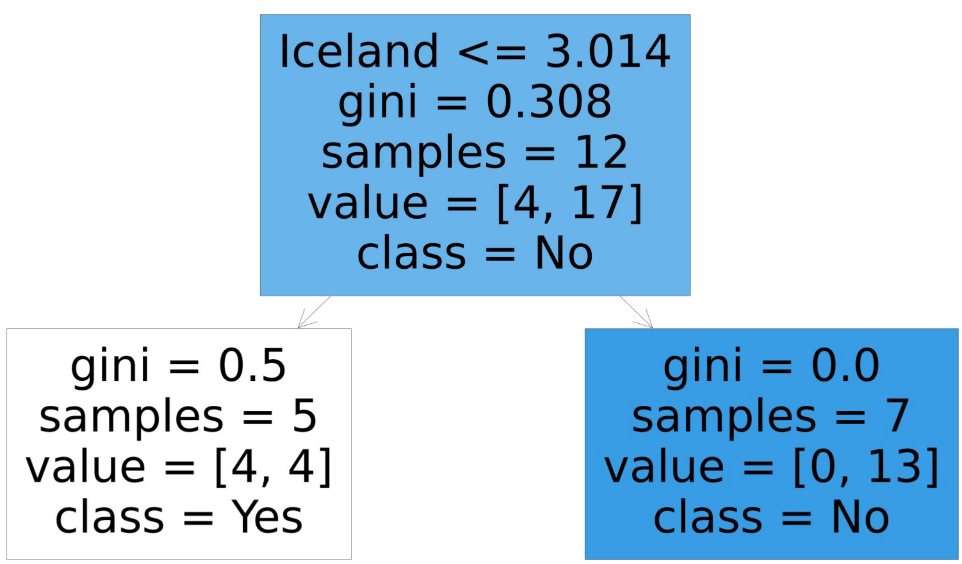
In [36]:

In [37]:

In [38]:
```
from sklearn.tree import plot_tree

plt.figure(figsize=(80,40))
```

Out[38]:
```
[Text(0.5, 0.75, 'Iceland <= 3.014\ngini = 0.308\nsamples = 12\nvalue = [4, 1
7]\nclass = No'),
 Text(0.25, 0.25, 'gini = 0.5\nsamples = 5\nvalue = [4, 4]\nclass = Yes'),
 Text(0.75, 0.25, 'gini = 0.0\nsamples = 7\nvalue = [0, 13]\nclass = No')]
```



In [39]:

```
print("Linear:",lis)
print("Lasso:",las)
print("Ridge:",rrs)
print("ElasticNet:",ens)
print("Logistic:",los)
```

```
Linear: 0.9594602355290545
Lasso: 0.4787865444920507
Ridge: 0.9589303894434331
ElasticNet: 0.8913730112352303
Logistic: 0.9
Random Forest: 0.7136363636363636
```