gjow0f8gh

August 9, 2023

1 20104169 - SUMESH R

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

2 Importing Datasets

[2]:
```
df=pd.read_csv("innovation_and_development_database.csv")
df
```

[2]:

	country	code	year	eap	eca	lac	mena	sha	sa	hi	…	\
0	Aruba	ABW	1960	0	0	1	0	0	0	0.0	…	
1	Aruba	ABW	1961	0	0	1	0	0	0	0.0	…	
2	Aruba	ABW	1962	0	0	1	0	0	0	0.0	…	
3	Aruba	ABW	1963	0	0	1	0	0	0	0.0	…	
4	Aruba	ABW	1964	0	0	1	0	0	0	0.0	…	
…	…	…	…	…	…	…	…	…	..	…	…	
8290	Zimbabwe	ZWE	1998	0	0	0	0	1	0	0.0	…	
8291	Zimbabwe	ZWE	1999	0	0	0	0	1	0	0.0	…	
8292	Zimbabwe	ZWE	2000	0	0	0	0	1	0	0.0	…	
8293	Zimbabwe	ZWE	2001	0	0	0	0	1	0	0.0	…	
8294	Zimbabwe	ZWE	2002	0	0	0	0	1	0	0.0	…	

	y	stockpatEPO	poptotal	labor	rdexpgdp	\
0	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	
…	…	…	…	…	…	
8290	8.290000e+09	8.0	12153850.0	6236700.0	NaN	
8291	8.230000e+09	8.0	12388320.0	6374300.0	NaN	
8292	7.830000e+09	8.0	12627000.0	6514800.0	NaN	
8293	NaN	8.0	12820650.0	NaN	NaN	

```
8294           NaN          NaN          NaN        NaN       NaN

      patgrantedstock  plantpatstock  designpatstock  plantpat  designpat
0                 NaN            NaN             NaN       NaN        NaN
1                 NaN            NaN             NaN       NaN        NaN
2                 NaN            NaN             NaN       NaN        NaN
3                 0.0            NaN             NaN       NaN        NaN
4                 0.0            NaN             NaN       NaN        NaN
…                 …              …               …         …          …
8290             48.0            0.0             1.0       0.0        0.0
8291             48.0            0.0             1.0       0.0        0.0
8292             48.0            0.0             1.0       0.0        0.0
8293             49.0            0.0             1.0       0.0        0.0
8294              NaN            NaN             NaN       0.0        0.0

[8295 rows x 33 columns]
```

3 Data Cleaning and Data Preprocessing

[3]:
```
df.fillna(1, inplace=True)
df
```

[3]:
```
        country code  year  eap  eca  lac  mena  sha  sa   hi  …  \
0         Aruba  ABW  1960    0    0    1     0    0   0  0.0  …
1         Aruba  ABW  1961    0    0    1     0    0   0  0.0  …
2         Aruba  ABW  1962    0    0    1     0    0   0  0.0  …
3         Aruba  ABW  1963    0    0    1     0    0   0  0.0  …
4         Aruba  ABW  1964    0    0    1     0    0   0  0.0  …
…         …  …   …  …  …  …   …  …  ..  …  …
8290  Zimbabwe  ZWE  1998    0    0    0     0    1   0  0.0  …
8291  Zimbabwe  ZWE  1999    0    0    0     0    1   0  0.0  …
8292  Zimbabwe  ZWE  2000    0    0    0     0    1   0  0.0  …
8293  Zimbabwe  ZWE  2001    0    0    0     0    1   0  0.0  …
8294  Zimbabwe  ZWE  2002    0    0    0     0    1   0  0.0  …

                 y  stockpatEPO    poptotal      labor  rdexpgdp  \
0     1.000000e+00           1.0         1.0        1.0       1.0
1     1.000000e+00           1.0         1.0        1.0       1.0
2     1.000000e+00           1.0         1.0        1.0       1.0
3     1.000000e+00           1.0         1.0        1.0       1.0
4     1.000000e+00           1.0         1.0        1.0       1.0
…              …           …           …          …         …
8290  8.290000e+09           8.0  12153850.0  6236700.0       1.0
8291  8.230000e+09           8.0  12388320.0  6374300.0       1.0
8292  7.830000e+09           8.0  12627000.0  6514800.0       1.0
8293  1.000000e+00           8.0  12820650.0        1.0       1.0
```

```
8294  1.000000e+00          1.0         1.0        1.0       1.0

      patgrantedstock  plantpatstock  designpatstock  plantpat  designpat
0                 1.0            1.0             1.0       1.0        1.0
1                 1.0            1.0             1.0       1.0        1.0
2                 1.0            1.0             1.0       1.0        1.0
3                 0.0            1.0             1.0       1.0        1.0
4                 0.0            1.0             1.0       1.0        1.0
…                 …              …               …         …          …
8290             48.0            0.0             1.0       0.0        0.0
8291             48.0            0.0             1.0       0.0        0.0
8292             48.0            0.0             1.0       0.0        0.0
8293             49.0            0.0             1.0       0.0        0.0
8294              1.0            1.0             1.0       0.0        0.0

[8295 rows x 33 columns]
```

[4]:
```
df.columns
```

[4]:
```
Index(['country', 'code', 'year', 'eap', 'eca', 'lac', 'mena', 'sha', 'sa',
       'hi', 'pat', 'patepo', 'royal', 'rdexp', 'rdper', 'rdfinabro',
       'rdfinprod', 'rdperfprod', 'rdperfhe', 'rdperfpub', 'lowrdexp',
       'lowrdfinprod', 'lowrdperfprod', 'y', 'stockpatEPO', 'poptotal',
       'labor', 'rdexpgdp', 'patgrantedstock', 'plantpatstock',
       'designpatstock', 'plantpat', 'designpat'],
      dtype='object')
```

[5]:
```
df.info()
```

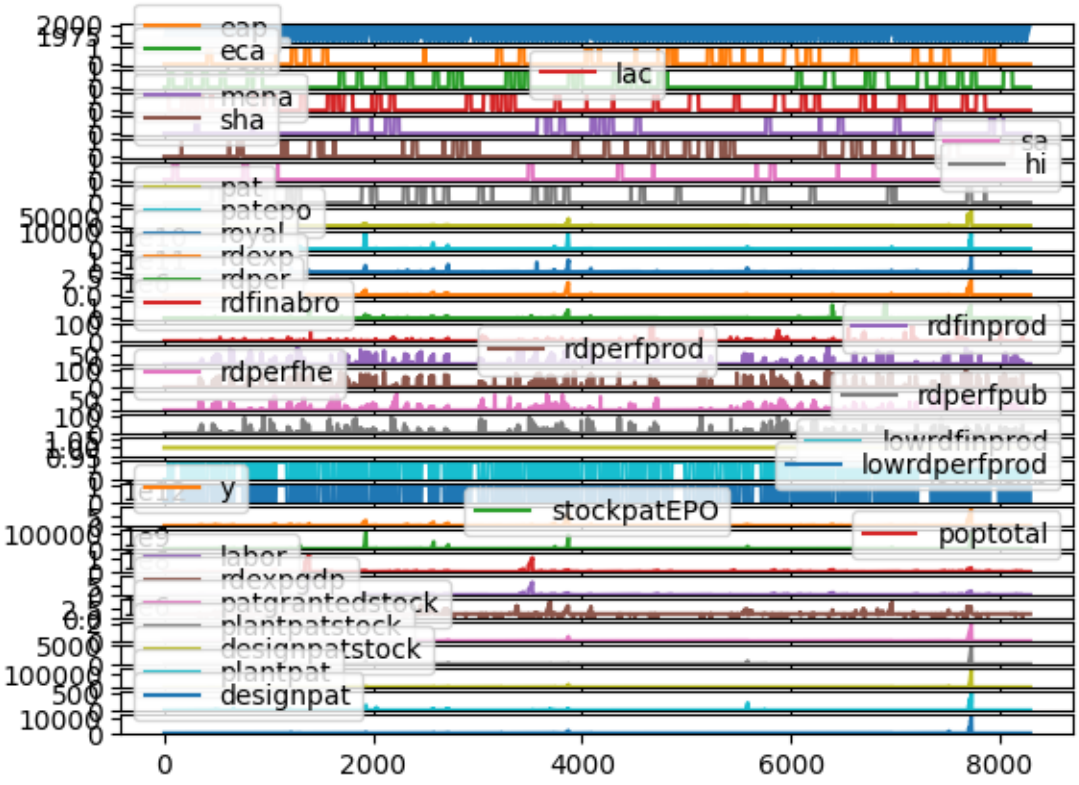
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8295 entries, 0 to 8294
Data columns (total 33 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   country          8295 non-null   object
 1   code             8295 non-null   object
 2   year             8295 non-null   int64
 3   eap              8295 non-null   int64
 4   eca              8295 non-null   int64
 5   lac              8295 non-null   int64
 6   mena             8295 non-null   int64
 7   sha              8295 non-null   int64
 8   sa               8295 non-null   int64
 9   hi               8295 non-null   float64
 10  pat              8295 non-null   float64
 11  patepo           8295 non-null   float64
 12  royal            8295 non-null   float64
```

```
 13  rdexp            8295 non-null   float64
 14  rdper            8295 non-null   float64
 15  rdfinabro        8295 non-null   float64
 16  rdfinprod        8295 non-null   float64
 17  rdperfprod       8295 non-null   float64
 18  rdperfhe         8295 non-null   float64
 19  rdperfpub        8295 non-null   float64
 20  lowrdexp         8295 non-null   float64
 21  lowrdfinprod     8295 non-null   float64
 22  lowrdperfprod    8295 non-null   float64
 23  y                8295 non-null   float64
 24  stockpatEPO      8295 non-null   float64
 25  poptotal         8295 non-null   float64
 26  labor            8295 non-null   float64
 27  rdexpgdp         8295 non-null   float64
 28  patgrantedstock  8295 non-null   float64
 29  plantpatstock    8295 non-null   float64
 30  designpatstock   8295 non-null   float64
 31  plantpat         8295 non-null   float64
 32  designpat        8295 non-null   float64
dtypes: float64(24), int64(7), object(2)
memory usage: 2.1+ MB
```

4 Line chart

[6]:
```
df.plot.line(subplots=True)
```

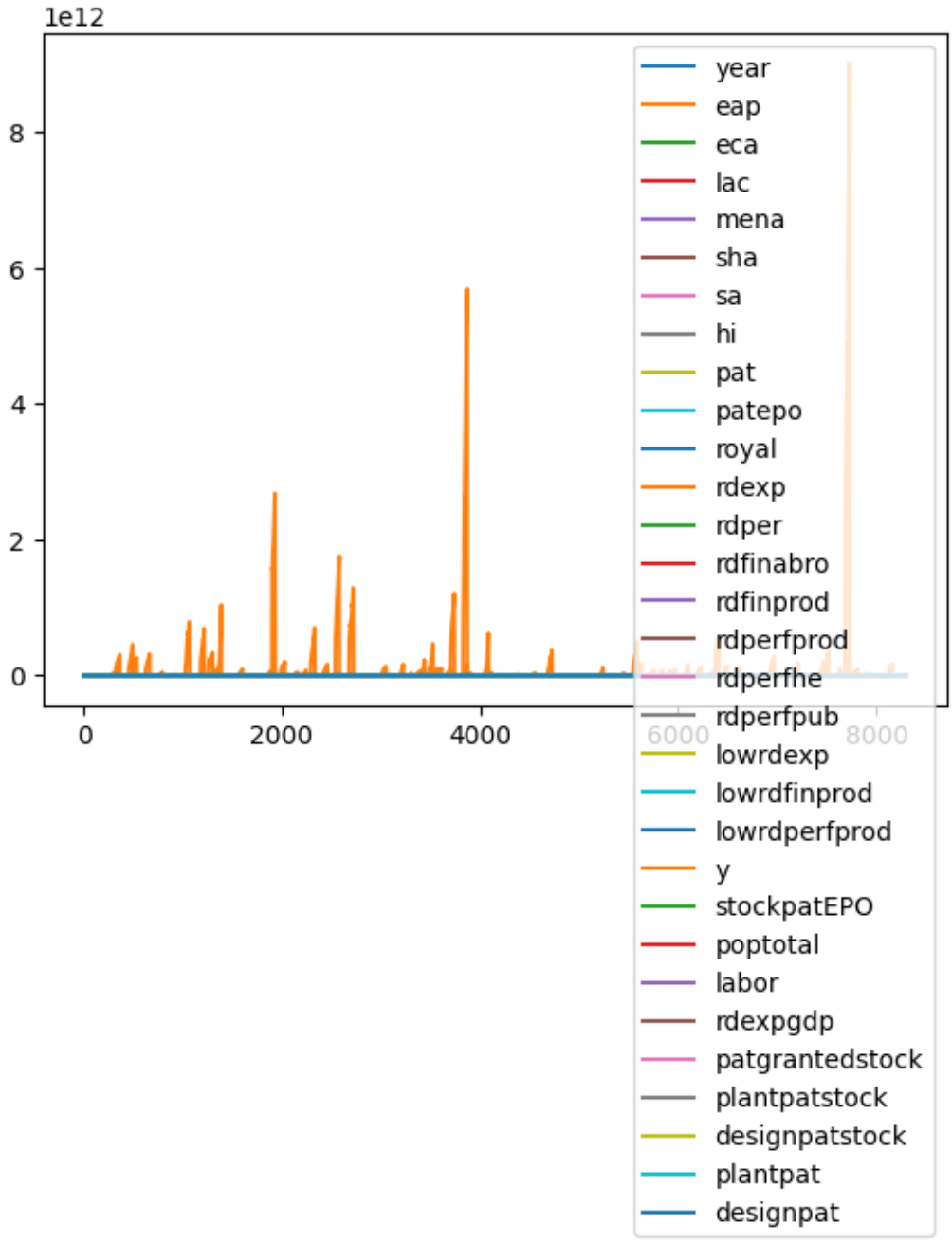
[6]:
```
array([<Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >], dtype=object)
```



5 Line chart

[7]:
```
df.plot.line()
```

[7]: <Axes: >

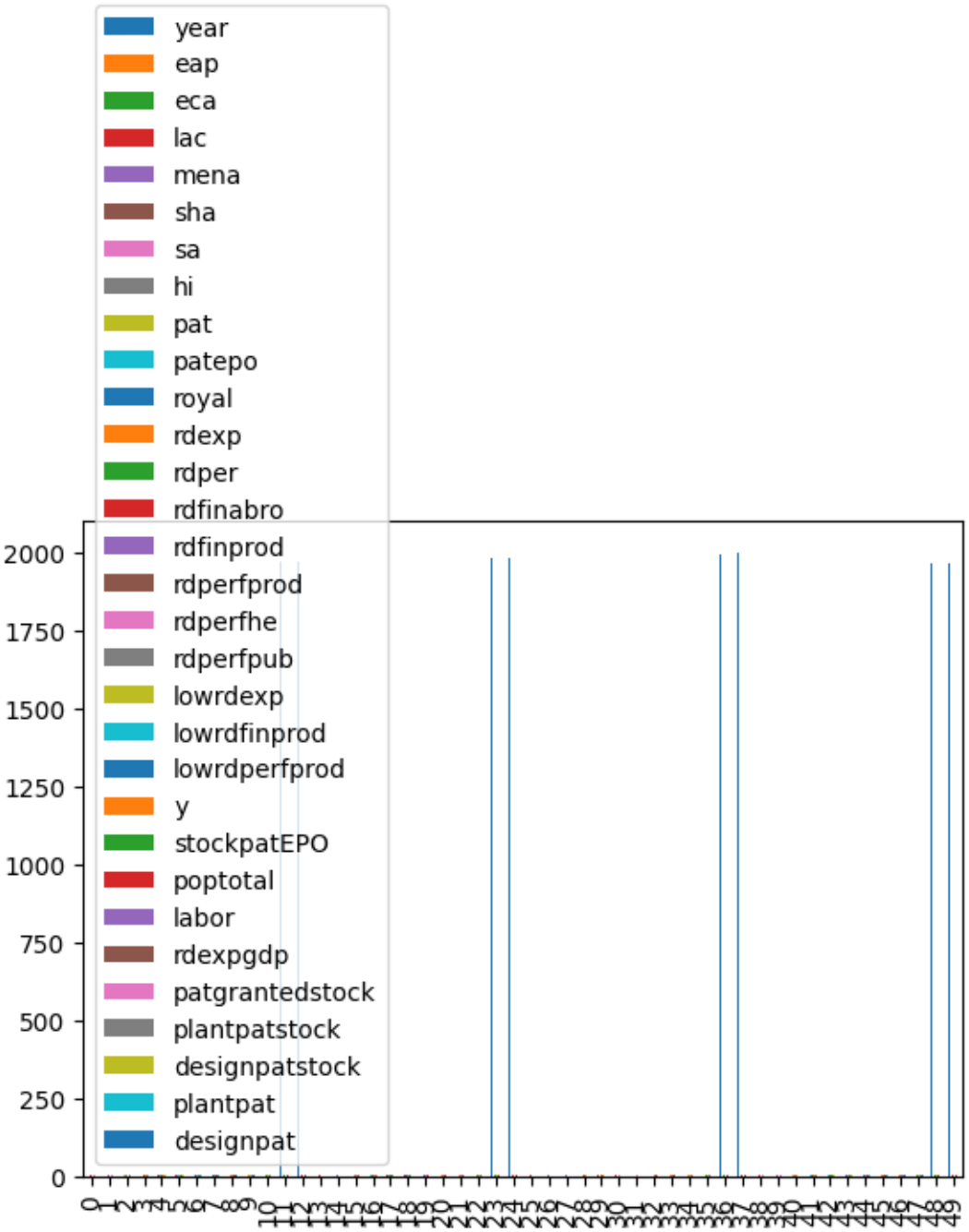


6 Bar chart

[8]:
```
b=df[0:50]
```

[9]:
```
b.plot.bar()
```

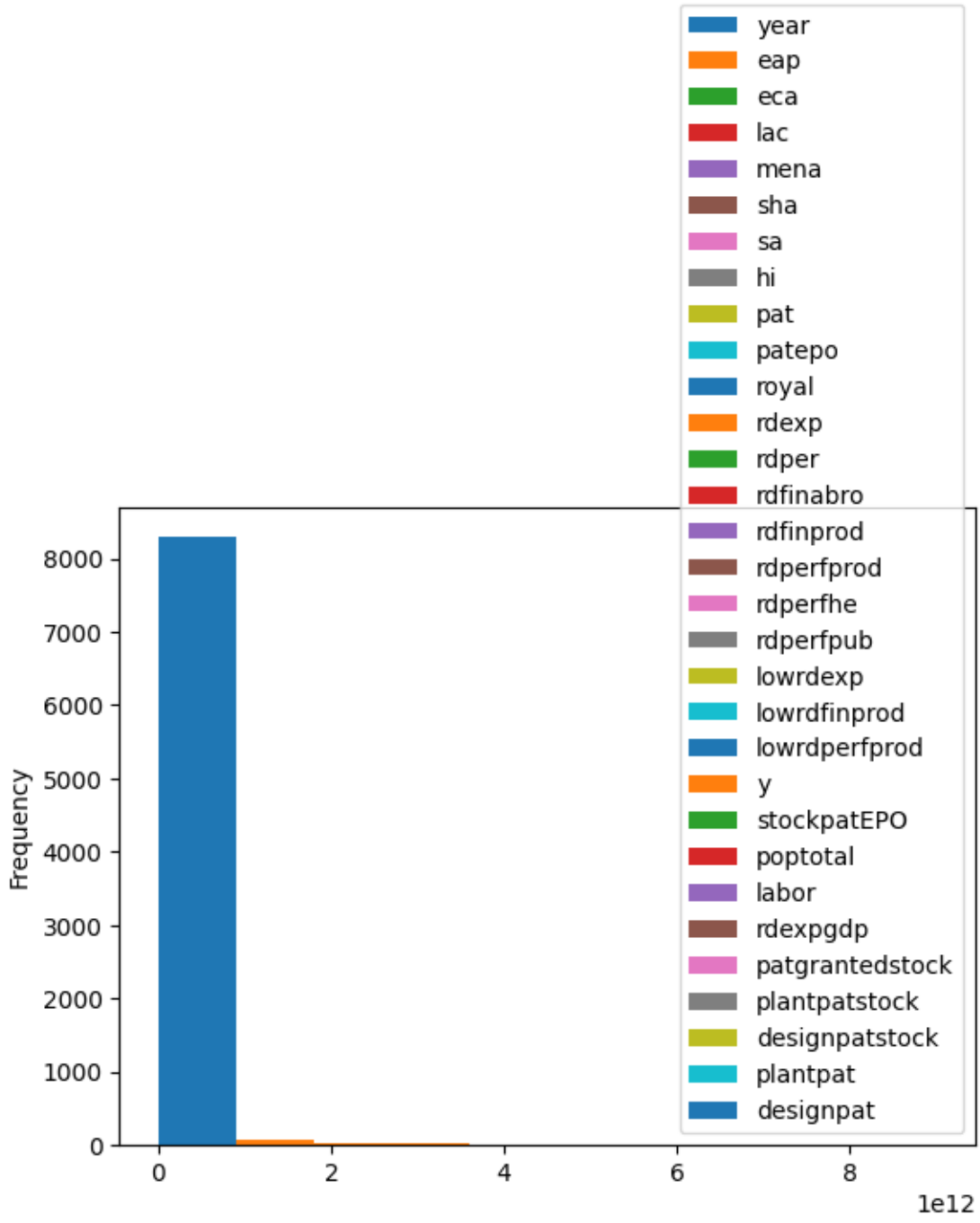
[9]: <Axes: >


year
eap
eca
lac
mena
sha
sa
hi
pat
patepo
royal
rdexp
rdper
rdfinabro
rdfinprod
rdperfprod
rdperfhe
rdperfpub
lowrdexp
lowrdfinprod
lowrdperfprod
y
stockpatEPO
poptotal
labor
rdexpgdp
patgrantedstock
plantpatstock
designpatstock
plantpat
designpat
2000
1750
1500
1250
1000
750
500
250
0

7 Histogram

[10]:
```
df.plot.hist()
```

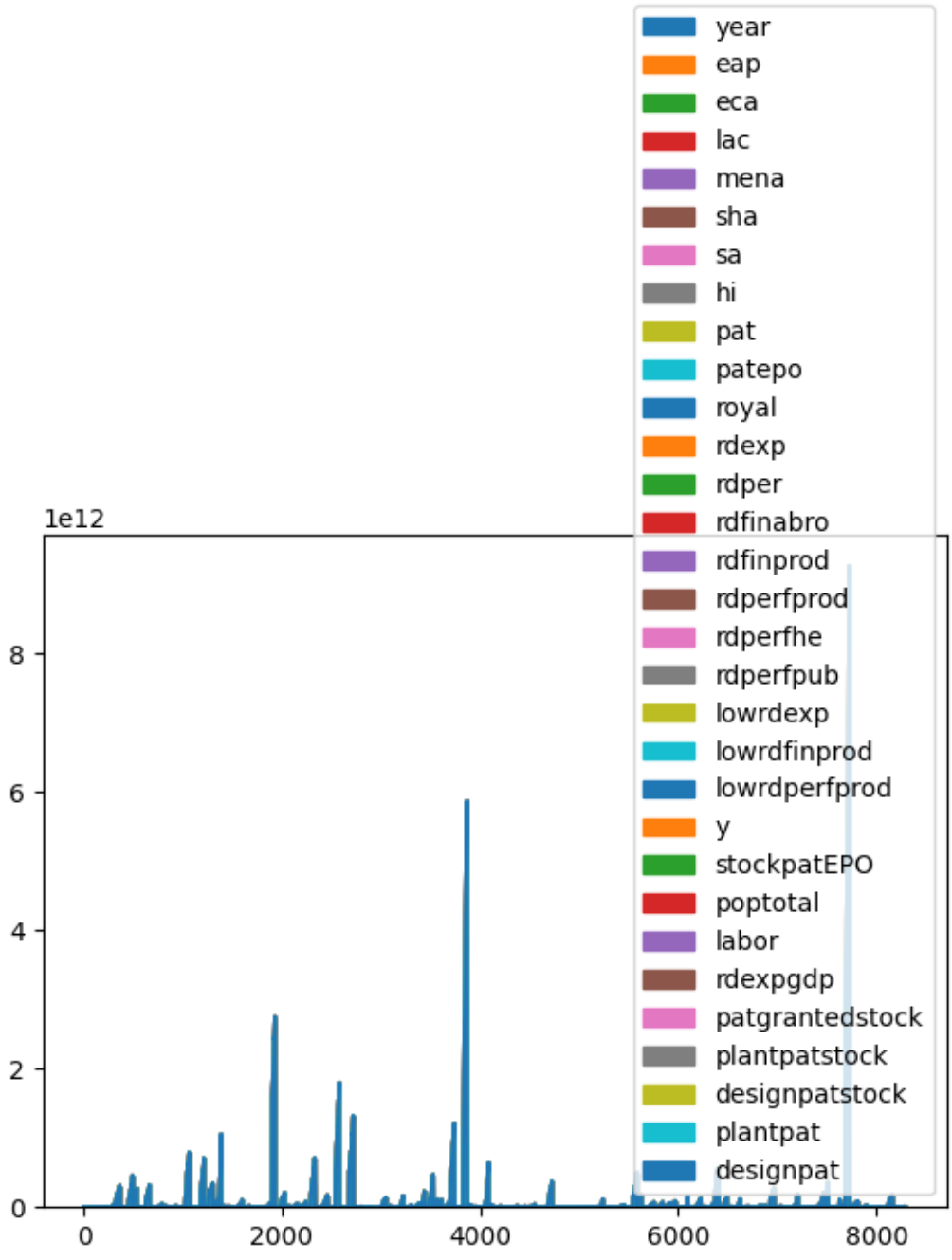
[10]:
```
<Axes: ylabel='Frequency'>
```



8 Area chart

[11]:
```
df.plot.area()
```

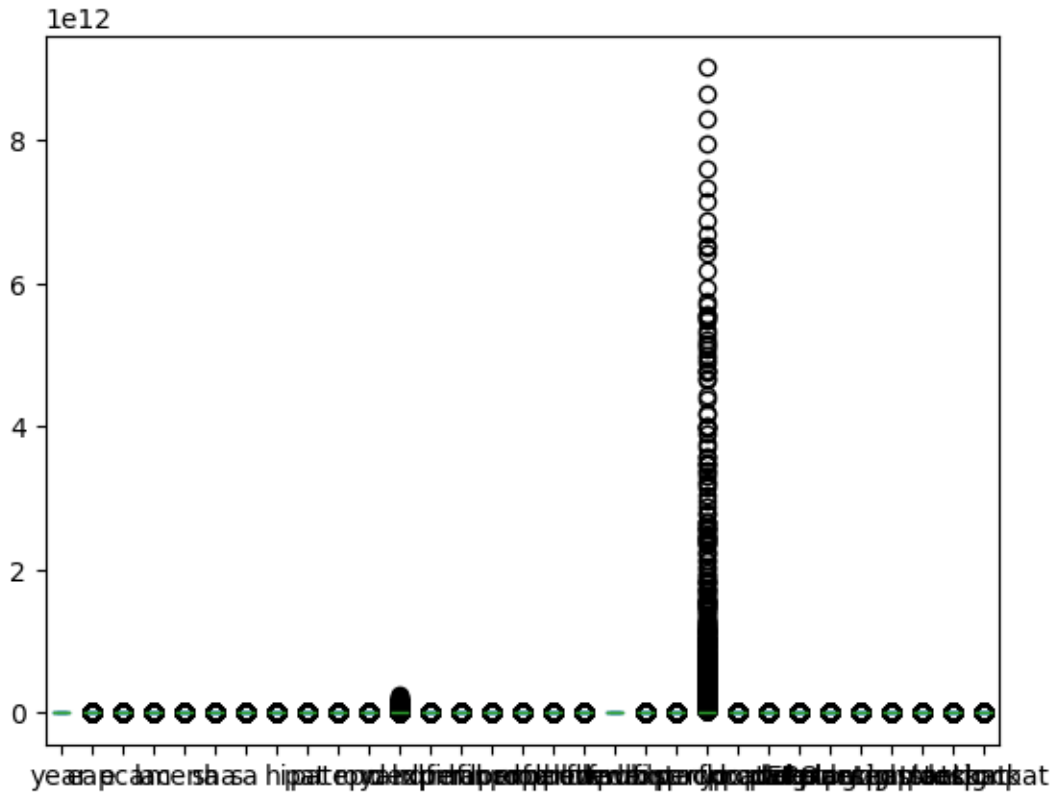
[11]: <Axes: >



9 Box chart

[12]:
```
df.plot.box()
```

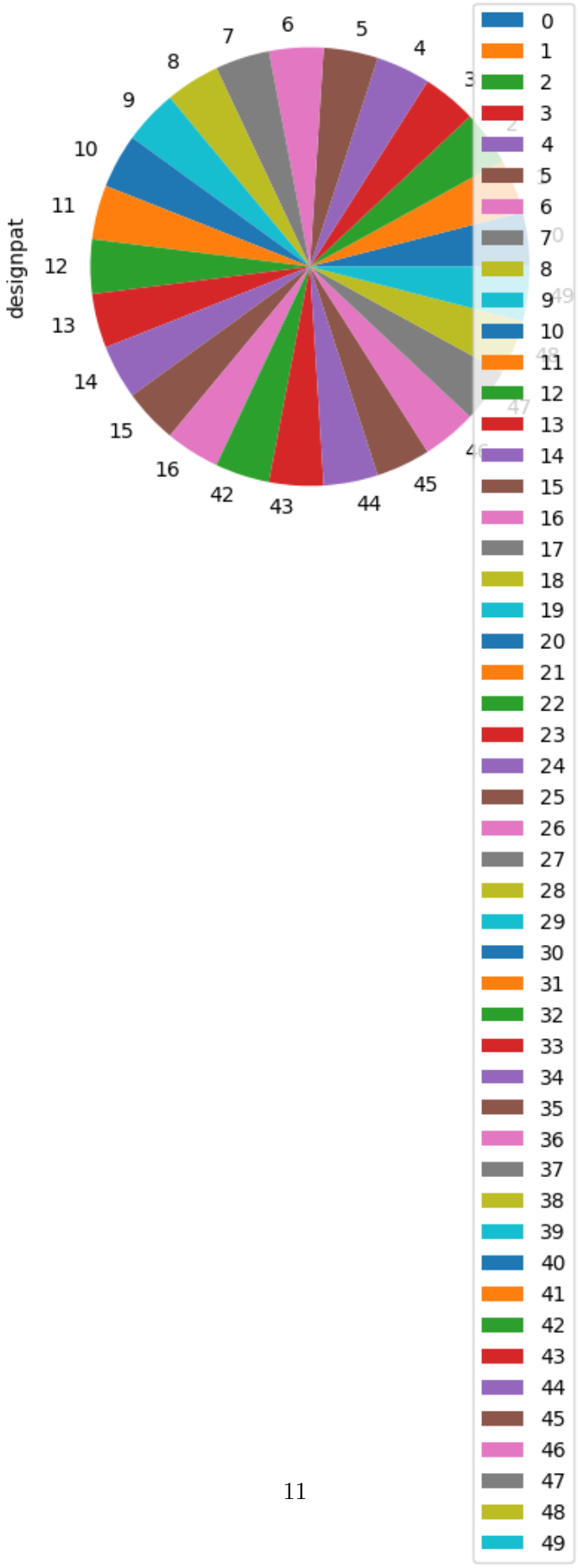
[12]: <Axes: >



10 Pie chart

[13]:
```
b.plot.pie(y='designpat' )
```

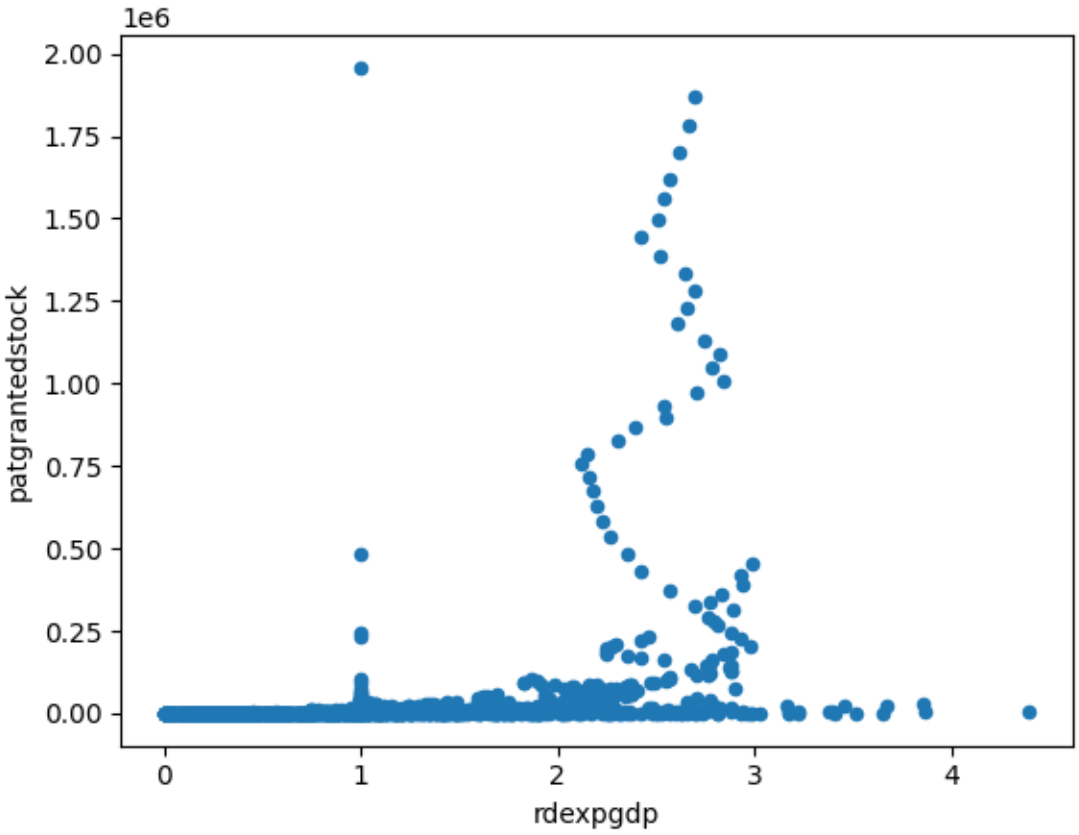
[13]: <Axes: ylabel='designpat'>


designpat
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
42
43
44
45
46
47
48
49

11 Scatter chart

[14]:
```
df.plot.scatter( x='rdexpgdp',y= 'patgrantedstock')
```

[14]:
```
<Axes: xlabel='rdexpgdp', ylabel='patgrantedstock'>
```



[15]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8295 entries, 0 to 8294
Data columns (total 33 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   country          8295 non-null   object
 1   code             8295 non-null   object
 2   year             8295 non-null   int64
 3   eap              8295 non-null   int64
```

```
 4   eca              8295 non-null   int64
 5   lac              8295 non-null   int64
 6   mena             8295 non-null   int64
 7   sha              8295 non-null   int64
 8   sa               8295 non-null   int64
 9   hi               8295 non-null   float64
 10  pat              8295 non-null   float64
 11  patepo           8295 non-null   float64
 12  royal            8295 non-null   float64
 13  rdexp            8295 non-null   float64
 14  rdper            8295 non-null   float64
 15  rdfinabro        8295 non-null   float64
 16  rdfinprod        8295 non-null   float64
 17  rdperfprod       8295 non-null   float64
 18  rdperfhe         8295 non-null   float64
 19  rdperfpub        8295 non-null   float64
 20  lowrdexp         8295 non-null   float64
 21  lowrdfinprod     8295 non-null   float64
 22  lowrdperfprod    8295 non-null   float64
 23  y                8295 non-null   float64
 24  stockpatEPO      8295 non-null   float64
 25  poptotal         8295 non-null   float64
 26  labor            8295 non-null   float64
 27  rdexpgdp         8295 non-null   float64
 28  patgrantedstock  8295 non-null   float64
 29  plantpatstock    8295 non-null   float64
 30  designpatstock   8295 non-null   float64
 31  plantpat         8295 non-null   float64
 32  designpat        8295 non-null   float64
dtypes: float64(24), int64(7), object(2)
memory usage: 2.1+ MB
```

[16]:
```
df.describe()
```

[16]:

	year	eap	eca	lac	mena	\
count	8295.000000	8295.000000	8295.000000	8295.000000	8295.000000	
mean	1981.203014	0.094515	0.195901	0.176974	0.098614	
std	12.421590	0.292561	0.396917	0.381670	0.298161	
min	1960.000000	0.000000	0.000000	0.000000	0.000000	
25%	1970.000000	0.000000	0.000000	0.000000	0.000000	
50%	1981.000000	0.000000	0.000000	0.000000	0.000000	
75%	1992.000000	0.000000	0.000000	0.000000	0.000000	
max	2002.000000	1.000000	1.000000	1.000000	1.000000	

	sha	sa	hi	pat	patepo	…	\
count	8295.000000	8295.000000	8295.000000	8295.000000	8295.000000	…	
mean	0.150090	0.026522	0.134901	411.663773	65.761061	…	

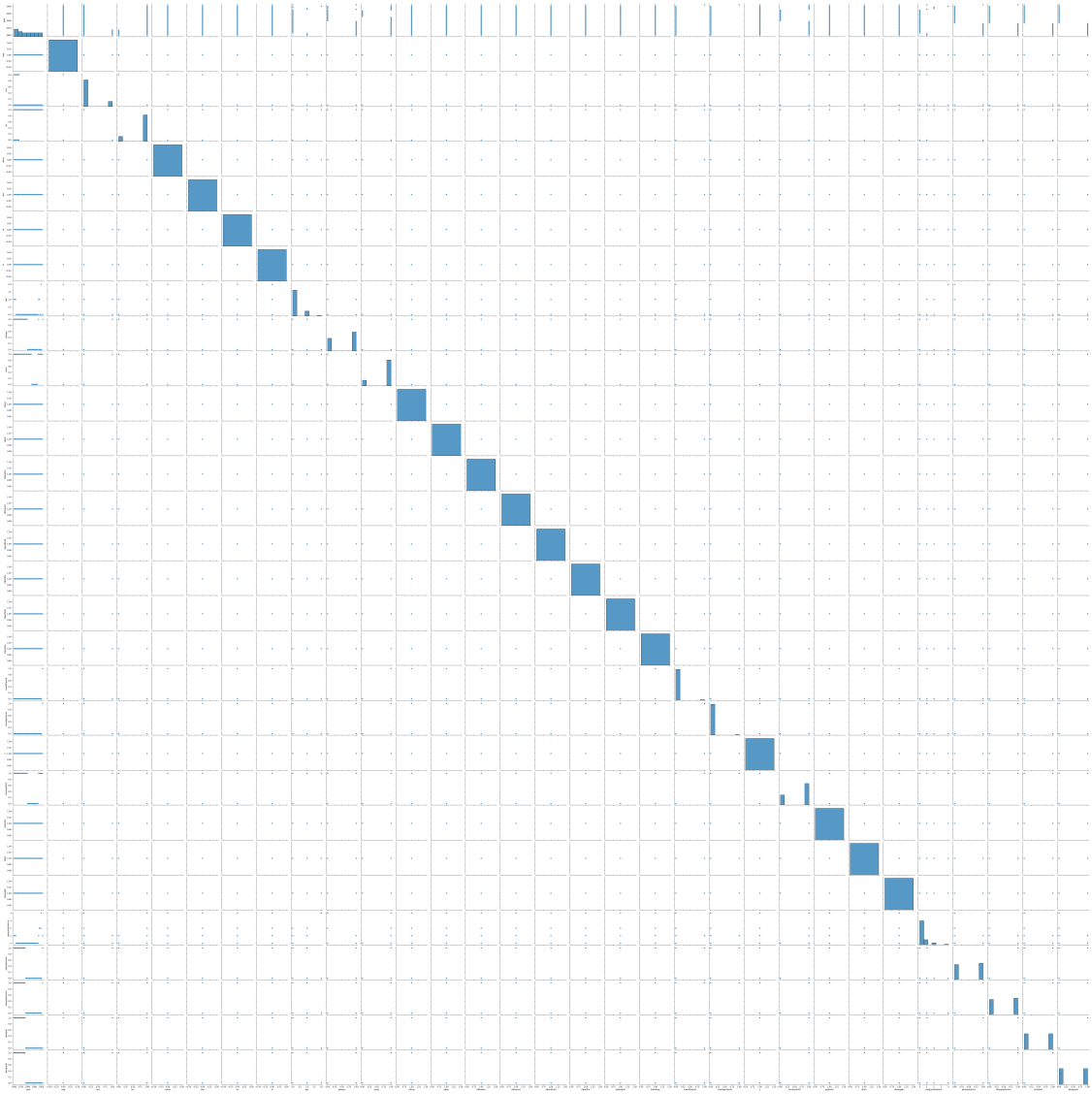
```
std       0.357182     0.160691     0.341638   3893.360772    580.347648  …
min       0.000000     0.000000     0.000000      0.000000      0.000000  …
25%       0.000000     0.000000     0.000000      0.000000      1.000000  …
50%       0.000000     0.000000     0.000000      0.000000      1.000000  …
75%       0.000000     0.000000     0.000000      2.000000      1.000000  …
max       1.000000     1.000000     1.000000  87607.000000  10300.000000  …

                 y    stockpatEPO      poptotal         labor     rdexpgdp  \
count  8.295000e+03    8295.000000  8.295000e+03  8.295000e+03  8295.000000
mean   9.342012e+10     552.738638  2.278327e+07  1.246788e+07     1.006609
std    4.884386e+11    5643.303127  9.103775e+07  5.494933e+07     0.347574
min    1.000000e+00       0.000000  1.000000e+00  1.000000e+00     0.000177
25%    1.000000e+00       1.000000  7.856450e+05  1.000000e+00     1.000000
50%    4.680000e+08       1.000000  4.375000e+06  1.773800e+06     1.000000
75%    1.800000e+10       1.000000  1.288402e+07  6.534300e+06     1.000000
max    9.010000e+12  122157.000000  1.270000e+09  8.622100e+08     4.399199

       patgrantedstock  plantpatstock  designpatstock     plantpat  \
count     8.295000e+03    8295.000000     8295.000000  8295.000000
mean      6.560155e+03      10.657263      239.712719     1.561181
std       7.441335e+04     146.687532     3807.148610    13.473203
min       0.000000e+00       0.000000        0.000000     0.000000
25%       0.000000e+00       0.000000        0.000000     0.000000
50%       1.000000e+00       0.000000        1.000000     0.000000
75%       2.900000e+01       1.000000        1.000000     1.000000
max       1.957665e+06    4843.000000   141143.000000   518.000000

          designpat
count   8295.000000
mean      27.471489
std      363.622886
min        0.000000
25%        0.000000
50%        1.000000
75%        1.000000
max    11285.000000

[8 rows x 31 columns]
```

[17]:
```
df1=df[['year', 'eap', 'eca', 'lac', 'mena', 'sha', 'sa',
       'hi', 'pat', 'patepo', 'royal', 'rdexp', 'rdper', 'rdfinabro',
       'rdfinprod', 'rdperfprod', 'rdperfhe', 'rdperfpub', 'lowrdexp',
       'lowrdfinprod', 'lowrdperfprod', 'y', 'stockpatEPO', 'poptotal',
       'labor', 'rdexpgdp', 'patgrantedstock', 'plantpatstock',
       'designpatstock', 'plantpat', 'designpat']]
```

12 EDA AND VISUALIZATION

[18]:
```
sns.pairplot(df1[0:50])
```

[18]:
```
<seaborn.axisgrid.PairGrid at 0x793d5e09dae0>
```

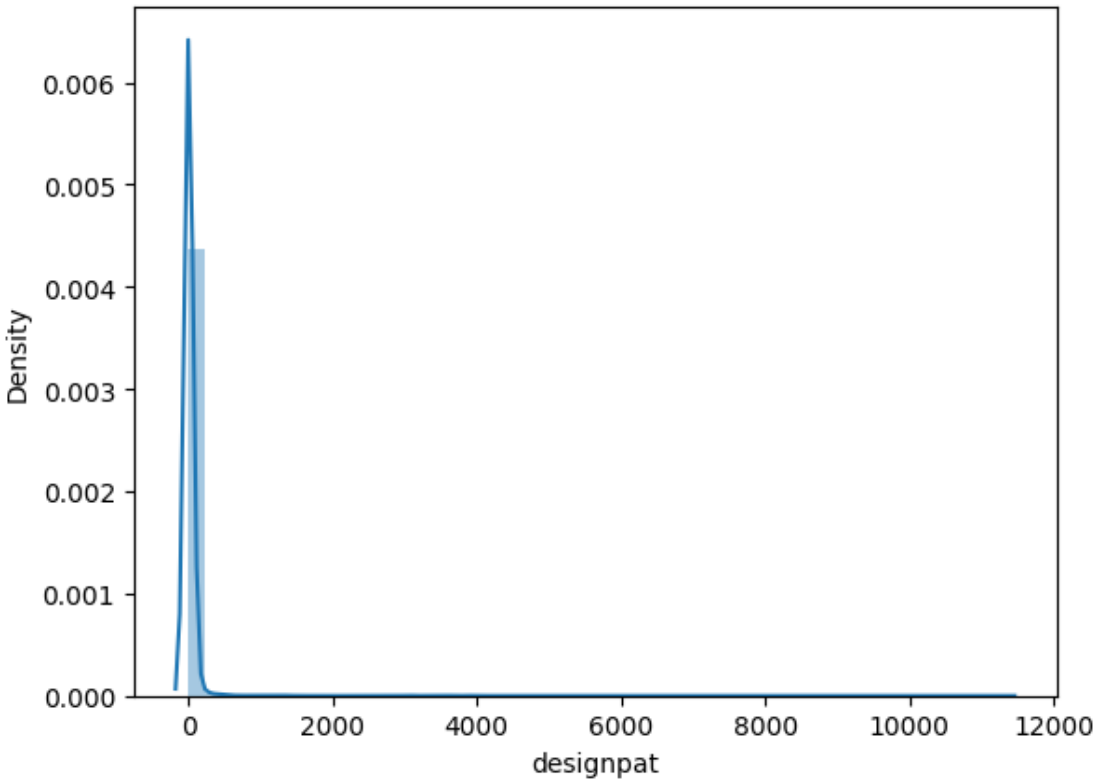


[19]:
```
sns.distplot(df1['designpat'])
```

```
<ipython-input-19-44d922bfa701>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
```

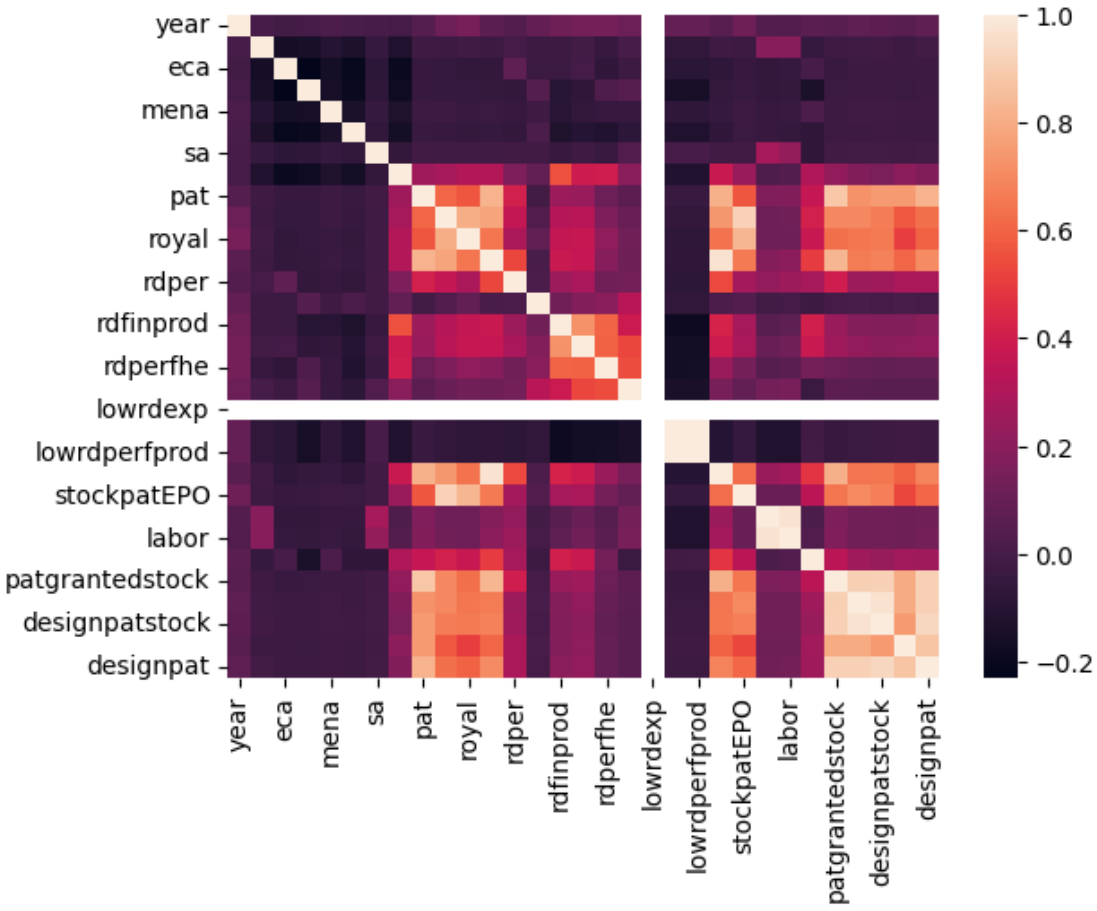
```
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df1['designpat'])
```

[19]: <Axes: xlabel='designpat', ylabel='Density'>



[20]:
```
sns.heatmap(df1.corr())
```

[20]: <Axes: >



13 TO TRAIN THE MODEL AND MODEL BULDING

[21]:
```
x=df[['year', 'eap', 'eca', 'lac', 'mena', 'sha', 'sa',
       'hi', 'pat', 'patepo', 'royal', 'rdexp', 'rdper', 'rdfinabro',
       'rdfinprod', 'rdperfprod', 'rdperfhe', 'rdperfpub', 'lowrdexp',
       'lowrdfinprod', 'lowrdperfprod', 'y', 'stockpatEPO', 'poptotal',
       'labor', 'rdexpgdp', 'patgrantedstock', 'plantpatstock',
       'designpatstock', 'plantpat']]
y=df['designpat']
```

[22]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

14 Linear Regression

[23]:
```
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(x_train,y_train)
```

[23]:
```
LinearRegression()
```

[24]:
```
lr.intercept_
```

[24]:
```
-811.7070607736804
```

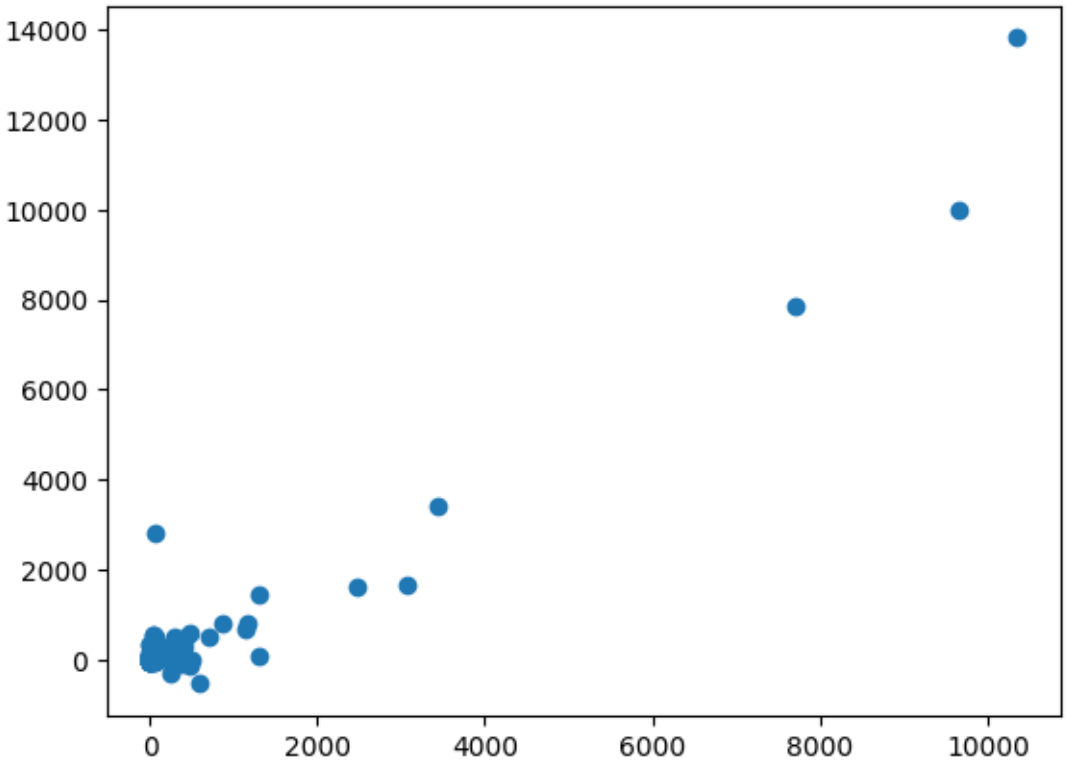
[25]:
```
coeff=pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
coeff
```

[25]:

	Co-efficient
year	4.085833e-01
eap	6.729644e+00
eca	-1.024191e+00
lac	6.151126e-01
mena	8.012824e-02
sha	1.423315e+00
sa	1.441939e+00
hi	-1.062438e+01
pat	1.308233e-02
patepo	2.213922e-02
royal	-1.263800e-08
rdexp	-1.470649e-09
rdper	-6.015484e-05
rdfinabro	-1.542915e-01
rdfinprod	3.835362e-01
rdperfprod	1.473682e-01
rdperfhe	-4.671098e-02
rdperfpub	-4.350392e-02
lowrdexp	1.219025e-12
lowrdfinprod	3.664809e+00
lowrdperfprod	-1.420297e+00
y	1.433964e-11
stockpatEPO	-4.536374e-03
poptotal	-3.153791e-08
labor	4.954050e-08
rdexpgdp	-4.312357e+00
patgrantedstock	-5.138568e-04
plantpatstock	-1.067862e+00
designpatstock	1.039823e-01
plantpat	1.281230e+01

[26]:
```
prediction =lr.predict(x_test)
plt.scatter(y_test,prediction)
```

[26]: <matplotlib.collections.PathCollection at 0x793d26362110>



15 ACCURACY

[27]:
```
lr.score(x_test,y_test)
```

[27]: 0.901608816724942

[28]:
```
lr.score(x_train,y_train)
```

[28]: 0.9638643462783607

16 Ridge and Lasso

[29]:
```
from sklearn.linear_model import Ridge,Lasso
```

[30]:
```
rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_ridge.py:216:
LinAlgWarning: Ill-conditioned matrix (rcond=6.3788e-27): result may not be
accurate.
  return linalg.solve(A, Xy, assume_a="pos", overwrite_a=True).T
```

[30]:
```
Ridge(alpha=10)
```

17 Accuracy(Ridge)

[31]:
```
rr.score(x_test,y_test)
```

[31]:
```
0.9015866978331674
```

[32]:
```
rr.score(x_train,y_train)
```

[32]:
```
0.9638642758073511
```

[33]:
```
la=Lasso(alpha=10)
la.fit(x_train,y_train)
```

[33]:
```
Lasso(alpha=10)
```

[34]:
```
la.score(x_train,y_train)
```

[34]:
```
0.9637190481323274
```

18 Accuracy(Lasso)

[35]:
```
la.score(x_test,y_test)
```

[35]:
```
0.9015386696202259
```

19 ElasticNet

[36]:
```
from sklearn.linear_model import ElasticNet
en=ElasticNet()
en.fit(x_train,y_train)
```

[36]:
```
ElasticNet()
```

[37]:
```
en.coef_
```

[37]:
```
array([ 4.32776819e-01,  2.10409937e-01, -0.00000000e+00,  0.00000000e+00,
       -0.00000000e+00,  0.00000000e+00,  0.00000000e+00, -5.27806342e-01,
        1.33108849e-02,  2.37960675e-02, -1.32565477e-08, -1.31479724e-09,
       -5.98300072e-05, -1.11129117e-01,  2.64049126e-01,  1.38548044e-01,
       -9.13728400e-02, -0.00000000e+00,  0.00000000e+00,  0.00000000e+00,
        0.00000000e+00,  5.58235221e-12, -4.73906817e-03, -2.49888662e-08,
        5.26849140e-08, -0.00000000e+00, -5.04583309e-04, -1.06544844e+00,
        1.04307693e-01,  1.26579078e+01])
```

[38]:
```
en.intercept_
```

[38]:
```
-862.8941978736544
```

[39]:
```
prediction=en.predict(x_test)
```

[40]:
```
en.score(x_test,y_test)
```

[40]:
```
0.9009443270496029
```

20 Evaluation Metrics

[41]:
```
from sklearn import metrics
print(metrics.mean_absolute_error(y_test,prediction))
print(metrics.mean_squared_error(y_test,prediction))
print(np.sqrt(metrics.mean_squared_error(y_test,prediction)))
```

```
13.878973646944198
11786.09446512705
108.56378063206462
```

21 Logistic Regression

[42]:
```
from sklearn.linear_model import LogisticRegression
```

[43]:
```
feature_matrix=df[['year', 'eap', 'eca', 'lac', 'mena', 'sha', 'sa',
       'hi', 'pat', 'patepo', 'royal', 'rdexp', 'rdper', 'rdfinabro',
       'rdfinprod', 'rdperfprod', 'rdperfhe', 'rdperfpub', 'lowrdexp',
       'lowrdfinprod', 'lowrdperfprod', 'y', 'stockpatEPO', 'poptotal',
       'labor', 'rdexpgdp', 'patgrantedstock', 'plantpatstock',
       'designpatstock', 'plantpat']]
target_vector=df[ 'designpat']
```

[44]:
```
feature_matrix.shape
```

[44]:
```
(8295, 30)
```

[45]:
```
target_vector.shape
```

[45]:
```
(8295,)
```

[46]:
```
from sklearn.preprocessing import StandardScaler
```

[47]:
```
fs=StandardScaler().fit_transform(feature_matrix)
```

[48]:
```
logr=LogisticRegression(max_iter=10000)
logr.fit(fs,target_vector)
```

[48]:
```
LogisticRegression(max_iter=10000)
```

[49]:
```
observation=[[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30
```

[50]:
```
prediction=logr.predict(observation)
print(prediction)
```

```
[212.]
```

[51]:
```
logr.classes_
```

[51]:
```
array([0.0000e+00, 1.0000e+00, 2.0000e+00, 3.0000e+00, 4.0000e+00,
       5.0000e+00, 6.0000e+00, 7.0000e+00, 8.0000e+00, 9.0000e+00,
       1.0000e+01, 1.1000e+01, 1.2000e+01, 1.3000e+01, 1.4000e+01,
       1.5000e+01, 1.6000e+01, 1.7000e+01, 1.8000e+01, 1.9000e+01,
       2.0000e+01, 2.1000e+01, 2.2000e+01, 2.3000e+01, 2.4000e+01,
       2.5000e+01, 2.6000e+01, 2.7000e+01, 2.8000e+01, 2.9000e+01,
       3.0000e+01, 3.2000e+01, 3.3000e+01, 3.4000e+01, 3.7000e+01,
       3.8000e+01, 3.9000e+01, 4.0000e+01, 4.1000e+01, 4.2000e+01,
       4.3000e+01, 4.4000e+01, 4.5000e+01, 4.6000e+01, 4.7000e+01,
       4.8000e+01, 4.9000e+01, 5.0000e+01, 5.4000e+01, 5.5000e+01,
       5.6000e+01, 5.7000e+01, 5.8000e+01, 5.9000e+01, 6.0000e+01,
       6.1000e+01, 6.2000e+01, 6.3000e+01, 6.4000e+01, 6.5000e+01,
       6.6000e+01, 6.7000e+01, 6.9000e+01, 7.0000e+01, 7.1000e+01,
       7.2000e+01, 7.3000e+01, 7.4000e+01, 7.6000e+01, 7.7000e+01,
       7.8000e+01, 8.0000e+01, 8.1000e+01, 8.2000e+01, 8.3000e+01,
       8.4000e+01, 8.5000e+01, 8.6000e+01, 8.7000e+01, 8.8000e+01,
       8.9000e+01, 9.0000e+01, 9.1000e+01, 9.3000e+01, 9.4000e+01,
       9.5000e+01, 9.6000e+01, 9.7000e+01, 9.8000e+01, 9.9000e+01,
       1.0000e+02, 1.0100e+02, 1.0200e+02, 1.0300e+02, 1.0500e+02,
       1.0600e+02, 1.0700e+02, 1.0900e+02, 1.1000e+02, 1.1200e+02,
       1.1300e+02, 1.1400e+02, 1.1500e+02, 1.1700e+02, 1.1800e+02,
       1.1900e+02, 1.2000e+02, 1.2100e+02, 1.2300e+02, 1.2400e+02,
       1.2500e+02, 1.2600e+02, 1.2700e+02, 1.2900e+02, 1.3200e+02,
       1.3300e+02, 1.3400e+02, 1.3600e+02, 1.3800e+02, 1.3900e+02,
       1.4200e+02, 1.4300e+02, 1.4400e+02, 1.4700e+02, 1.5000e+02,
```

```
1.5200e+02, 1.5300e+02, 1.5600e+02, 1.5900e+02, 1.6000e+02,
1.6200e+02, 1.6300e+02, 1.6500e+02, 1.6900e+02, 1.7200e+02,
1.7300e+02, 1.7600e+02, 1.7900e+02, 1.8000e+02, 1.8100e+02,
1.8200e+02, 1.8400e+02, 1.8500e+02, 1.8600e+02, 1.9000e+02,
1.9300e+02, 1.9500e+02, 1.9600e+02, 1.9700e+02, 2.0100e+02,
2.0200e+02, 2.0500e+02, 2.0800e+02, 2.1100e+02, 2.1200e+02,
2.1300e+02, 2.1500e+02, 2.1600e+02, 2.2100e+02, 2.2200e+02,
2.2700e+02, 2.2800e+02, 2.3000e+02, 2.3100e+02, 2.3400e+02,
2.3700e+02, 2.3900e+02, 2.4000e+02, 2.4300e+02, 2.4700e+02,
2.5000e+02, 2.5300e+02, 2.5400e+02, 2.5700e+02, 2.5800e+02,
2.6000e+02, 2.6500e+02, 2.7500e+02, 2.8200e+02, 3.0000e+02,
3.0600e+02, 3.2000e+02, 3.3000e+02, 3.3800e+02, 3.4100e+02,
3.5000e+02, 3.5600e+02, 3.6000e+02, 3.6800e+02, 3.7000e+02,
3.7200e+02, 3.8200e+02, 3.9000e+02, 3.9600e+02, 4.0100e+02,
4.1000e+02, 4.1800e+02, 4.3800e+02, 4.3900e+02, 4.6600e+02,
4.8200e+02, 4.8400e+02, 4.8500e+02, 5.0300e+02, 5.0500e+02,
5.0900e+02, 5.2200e+02, 5.3900e+02, 5.4700e+02, 5.7600e+02,
5.8800e+02, 6.2000e+02, 6.9800e+02, 7.0500e+02, 7.9500e+02,
8.3300e+02, 8.6100e+02, 9.3600e+02, 9.4800e+02, 1.0260e+03,
1.0370e+03, 1.0560e+03, 1.1350e+03, 1.1370e+03, 1.1490e+03,
1.1680e+03, 1.2070e+03, 1.2940e+03, 1.3070e+03, 1.3100e+03,
1.3640e+03, 1.4970e+03, 1.5460e+03, 2.4690e+03, 3.0520e+03,
3.0550e+03, 3.0650e+03, 3.2780e+03, 3.4280e+03, 3.4460e+03,
3.4750e+03, 3.5460e+03, 3.5700e+03, 3.6450e+03, 3.8830e+03,
3.9020e+03, 5.0690e+03, 6.0130e+03, 6.0750e+03, 7.4160e+03,
7.6970e+03, 7.7470e+03, 7.8630e+03, 8.2510e+03, 9.3250e+03,
9.6540e+03, 9.9130e+03, 1.0346e+04, 1.1285e+04])
```

[52]:
```
logr.score(fs,target_vector)
```

[52]:
```
0.8764315852923448
```

[53]:
```
logr.predict_proba(observation)[0][0]
```

[53]:
```
0.0
```

[54]:
```
logr.predict_proba(observation)
```

[54]:
```
array([[0.00000000e+00, 6.22831858e-85, 3.30126281e-86, 4.37916319e-80,
        8.08978487e-83, 2.19391650e-56, 3.45360252e-68, 2.35067676e-74,
        3.12474087e-68, 7.00530593e-60, 3.84182327e-85, 5.26617015e-55,
        8.30126619e-74, 1.00049927e-59, 9.88115347e-51, 8.19961396e-68,
        1.70214924e-50, 2.48757137e-61, 3.85665745e-50, 6.58911151e-49,
        3.28745458e-52, 7.36043550e-50, 3.89824284e-44, 7.74173071e-71,
        1.95066894e-32, 1.32214744e-51, 2.75073588e-25, 1.01593368e-42,
        3.27258276e-36, 3.55814691e-58, 2.36763445e-38, 3.88298056e-33,
        3.88973903e-49, 1.79835583e-43, 1.32011387e-54, 1.16211039e-50,
```

1.48402856e-22, 2.02686339e-29, 1.12503605e-34, 8.05018881e-30,
2.06211778e-44, 3.77050000e-15, 5.89102403e-36, 3.39091758e-12,
1.49580951e-26, 5.20083858e-31, 1.94531686e-18, 1.55476169e-35,
3.38716634e-25, 6.97311939e-09, 1.51131938e-36, 6.66741070e-19,
8.87194698e-31, 2.85516576e-26, 4.27664871e-32, 3.03165971e-26,
3.11504842e-38, 1.89465672e-46, 1.05480126e-24, 5.10023220e-29,
4.16928027e-14, 4.94838449e-35, 4.64051187e-13, 4.01650047e-43,
1.79621071e-38, 5.38060794e-41, 1.50270066e-22, 9.55644313e-29,
3.92827229e-32, 1.93117486e-21, 1.39031327e-13, 2.48784729e-17,
2.08444676e-27, 4.15490573e-37, 5.56550635e-11, 1.54408921e-04,
9.49003310e-34, 8.35342286e-25, 1.64851505e-19, 1.26156899e-26,
3.18807774e-25, 3.33357905e-25, 4.91381034e-21, 1.01772110e-32,
3.67738656e-30, 2.92804299e-14, 1.57423734e-16, 2.22846687e-31,
4.25260719e-20, 9.68853432e-32, 6.05637952e-18, 6.85980251e-15,
4.67704083e-20, 4.76743233e-12, 1.23439273e-26, 2.42556104e-28,
3.15864577e-23, 1.19392403e-20, 1.35266574e-35, 2.31011529e-19,
2.52753582e-28, 2.99437878e-25, 1.83625643e-25, 2.57831069e-47,
3.81262680e-38, 2.05515321e-26, 4.47801570e-29, 2.69716352e-43,
2.13221941e-24, 1.44493657e-22, 1.36581818e-44, 4.06002433e-24,
1.04833706e-27, 1.24010993e-12, 2.48811544e-18, 1.25722908e-36,
5.83204783e-25, 9.38344764e-17, 2.31104141e-31, 2.74893347e-28,
2.38534731e-19, 1.04788538e-20, 6.95969513e-15, 7.18874932e-24,
5.94879101e-19, 3.13523861e-23, 5.36751855e-11, 1.21265723e-29,
1.76098627e-26, 2.90766535e-35, 4.07407063e-41, 9.20503202e-43,
1.92918171e-18, 4.28752117e-11, 2.22624775e-22, 1.23538662e-19,
1.05125345e-23, 1.09332971e-17, 5.21741773e-41, 8.40172692e-13,
2.46311490e-12, 2.91424261e-06, 1.27500241e-37, 3.81946808e-07,
3.43144665e-26, 2.61401840e-11, 3.88867251e-18, 3.52980070e-11,
3.86956986e-12, 3.01234659e-10, 2.93617816e-16, 1.90031528e-29,
1.76198885e-22, 1.42481697e-23, 9.99406145e-01, 1.68638489e-40,
8.22665056e-09, 3.55184738e-33, 7.80467039e-19, 5.92058844e-17,
1.45246206e-21, 1.43656696e-13, 4.13492913e-09, 2.98485429e-20,
6.11495502e-14, 1.75771934e-16, 1.76611195e-19, 5.42847630e-10,
2.17063879e-18, 1.91318796e-23, 7.20968029e-15, 1.20584298e-10,
1.92196099e-34, 9.18547449e-11, 2.14795449e-34, 6.73999181e-16,
6.00762143e-24, 1.36495617e-10, 5.90061955e-05, 1.29802271e-13,
6.01369025e-16, 2.27236083e-26, 4.28865722e-15, 1.26930943e-13,
2.17198403e-18, 4.20734902e-17, 2.23147976e-26, 3.35498881e-11,
1.06716541e-31, 1.98362407e-24, 3.37891339e-07, 1.31396759e-40,
1.53264853e-16, 1.38384114e-25, 5.01563656e-18, 3.00805135e-04,
1.12132140e-10, 2.24379044e-12, 1.83913941e-21, 4.89982917e-23,
2.59982022e-17, 2.82120718e-10, 1.15442797e-07, 4.34121854e-20,
7.37298412e-05, 1.52036399e-19, 3.09031877e-21, 1.78697175e-20,
5.07391198e-12, 2.00946087e-22, 4.64104596e-22, 8.18535532e-20,
4.49354610e-19, 2.30232395e-18, 5.10804085e-17, 1.32081178e-15,
3.08588684e-17, 1.14193850e-18, 3.04511619e-17, 1.44678146e-14,
4.78389733e-18, 6.29368038e-21, 2.12165761e-13, 9.22257163e-15,

```
3.25794138e-15, 6.31871190e-26, 1.19641511e-11, 1.41787365e-24,
1.11373583e-18, 2.48083207e-12, 4.71046803e-26, 1.32541905e-10,
5.61521789e-20, 8.71802120e-19, 5.64128022e-21, 1.22544056e-18,
6.85909025e-13, 1.00764896e-08, 1.53370513e-13, 2.54959210e-08,
1.27111770e-11, 6.99449987e-12, 1.92054970e-13, 5.36055959e-07,
4.02202781e-09, 7.92737492e-11, 3.13523430e-13, 2.10341118e-09,
7.97160665e-07, 1.05870572e-08, 2.31446617e-08, 7.16843893e-07,
2.27223481e-11, 2.50269185e-11, 1.87794026e-21, 1.63527548e-13,
6.54465718e-09, 4.82174655e-12, 1.41529918e-09]])
```

22 Random Forest

[55]:
```
from sklearn.ensemble import RandomForestClassifier
```

[56]:
```
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

[56]:
```
RandomForestClassifier()
```

[57]:
```
parameters={'max_depth':[1,2,3,4,5],
            'min_samples_leaf':[5,10,15,20,25],
            'n_estimators':[10,20,30,40,50]
}
```

[58]:
```
from sklearn.model_selection import GridSearchCV
grid_search␣
  ↪=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="accuracy")
grid_search.fit(x_train,y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=2.
  warnings.warn(
```

[58]:
```
GridSearchCV(cv=2, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [1, 2, 3, 4, 5],
                         'min_samples_leaf': [5, 10, 15, 20, 25],
                         'n_estimators': [10, 20, 30, 40, 50]},
             scoring='accuracy')
```

[59]:
```
grid_search.best_score_
```

[59]:
```
0.8837409576300379
```

[60]:
```
rfc_best=grid_search.best_estimator_
```

[61]:
```
from sklearn.tree import plot_tree

plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[5],feature_names=x.
 ↪columns,class_names=['variable1', 'variable2', 'variable3', 'variable4',␣
 ↪'variable5', 'variable6', 'variable7', 'variable8', 'variable9',␣
 ↪'variable10',
'variable11', 'variable12', 'variable13', 'variable14', 'variable15',␣
 ↪'variable16', 'variable17', 'variable18', 'variable19', 'variable20',
'variable21', 'variable22', 'variable23', 'variable24', 'variable25',␣
 ↪'variable26', 'variable27', 'variable28', 'variable29', 'variable30',
'variable31', 'variable32', 'variable33', 'variable34', 'variable35',␣
 ↪'variable36', 'variable37', 'variable38', 'variable39', 'variable40',
'variable41', 'variable42', 'variable43', 'variable44', 'variable45',␣
 ↪'variable46', 'variable47', 'variable48', 'variable49', 'variable50',
'variable51', 'variable52', 'variable53', 'variable54', 'variable55',␣
 ↪'variable56', 'variable57', 'variable58', 'variable59', 'variable60',
'variable61', 'variable62', 'variable63', 'variable64', 'variable65',␣
 ↪'variable66', 'variable67', 'variable68', 'variable69', 'variable70',
'variable71', 'variable72', 'variable73', 'variable74', 'variable75',␣
 ↪'variable76', 'variable77', 'variable78', 'variable79', 'variable80',
'variable81', 'variable82', 'variable83', 'variable84', 'variable85',␣
 ↪'variable86', 'variable87', 'variable88', 'variable89', 'variable90',
'variable91', 'variable92', 'variable93', 'variable94', 'variable95',␣
 ↪'variable96', 'variable97', 'variable98', 'variable99',␣
 ↪'variable100','apple', 'banana', 'cherry', 'dog', 'elephant', 'frog',␣
 ↪'grape', 'honey', 'icecream', 'jelly', 'kiwi', 'lemon', 'mango', 'nut',␣
 ↪'orange', 'pear', 'quilt', 'rabbit', 'strawberry', 'tiger', 'umbrella',␣
 ↪'violet', 'watermelon', 'xylophone', 'yogurt', 'zebra', 'avocado',␣
 ↪'broccoli', 'carrot', 'dolphin', 'eggplant', 'flamingo', 'giraffe',␣
 ↪'hamburger', 'iguana', 'jalapeno', 'kangaroo', 'lizard', 'muffin', 'noodle',␣
 ↪'ostrich', 'penguin', 'quokka', 'raccoon', 'sandwich', 'taco', 'unicorn',␣
 ↪'vulture', 'waffle', 'xylophone', 'yarn', 'zeppelin', 'acorn', 'butterfly',␣
 ↪'cactus', 'dragonfly', 'elephant', 'feather', 'gazelle', 'hedgehog',␣
 ↪'iguana', 'jaguar', 'koala', 'lemur', 'meerkat', 'narwhal', 'octopus',␣
 ↪'panda', 'quokka', 'raccoon', 'sloth', 'toucan', 'urchin', 'vulture',␣
 ↪'walrus', 'xenops', 'yak', 'zebra', 'alpaca', 'bison', 'cheetah', 'dolphin',␣
 ↪'elephant', 'flamingo', 'giraffe', 'hedgehog', 'iguana', 'jellyfish',␣
 ↪'kangaroo', 'lemur', 'macaw', 'narwhal', 'opossum', 'penguin', 'quokka',␣
 ↪'rhinoceros', 'sloth', 'tiger', 'umbrellabird', 'vulture', 'whale', 'x-ray␣
 ↪tetra', 'yak', 'zebra'],filled=True)
```

[61]:
```
[Text(0.49107142857142855, 0.9166666666666666, 'year <= 1976.5\ngini =
0.595\nsamples = 3639\nvalue = [2614, 2609, 59, 30, 35, 14, 20, 15, 14, 11,
11\n9, 6, 11, 4, 16, 5, 12, 4, 6, 2, 4, 4, 6, 1\n7, 11, 2, 1, 7, 3, 2, 6, 4, 0,
2, 1, 0, 1\n1, 2, 3, 5, 1, 3, 3, 5, 1, 1, 0, 2, 2, 3\n1, 1, 6, 2, 1, 3, 2, 4, 0,
```

```
1, 3, 3, 1, 4\n3, 3, 5, 0, 2, 4, 0, 0, 3, 0, 0, 1, 2, 3\n5, 0, 1, 2, 2, 1, 1, 2,
3, 0, 1, 0, 1, 2\n0, 2, 0, 2, 3, 1, 0, 1, 1, 3, 2, 2, 3, 1\n4, 0, 2, 1, 3, 2, 1,
1, 2, 2, 1, 0, 2, 3\n2, 1, 1, 3, 2, 0, 0, 0, 1, 2, 0, 2, 1, 1\n2, 1, 1, 3, 0, 0,
3, 1, 3, 1, 4, 0, 0, 0\n1, 0, 0, 6, 2, 1, 0, 0, 3, 2, 1, 1, 1, 0\n1, 0, 2, 4, 0,
1, 0, 1, 1, 1, 0, 0, 0, 0\n0, 1, 1, 0, 1, 0, 1, 0, 0, 2, 1, 1, 2, 0\n0, 0, 1, 1,
2, 0, 1, 0, 0, 0, 1, 1]\nclass = variable1'),
 Text(0.45535714285714285, 0.75, 'gini = 0.0\nsamples = 1399\nvalue = [0, 2263,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass = variable2'),
 Text(0.5267857142857143, 0.75, 'rdperfhe <= 7.35\ngini = 0.445\nsamples =
2240\nvalue = [2614, 346, 59, 30, 35, 14, 20, 15, 14, 11, 11, 9\n6, 11, 4, 16,
5, 12, 4, 6, 2, 4, 4, 6, 1, 7\n11, 2, 1, 7, 3, 2, 6, 4, 0, 2, 1, 0, 1, 1\n2, 3,
5, 1, 3, 3, 5, 1, 1, 0, 2, 2, 3, 1\n1, 6, 2, 1, 3, 2, 4, 0, 1, 3, 3, 1, 4, 3\n3,
5, 0, 2, 4, 0, 0, 3, 0, 0, 1, 2, 3, 5\n0, 1, 2, 2, 1, 1, 2, 3, 0, 1, 0, 1, 2,
0\n2, 0, 2, 3, 1, 0, 1, 1, 3, 2, 2, 3, 1, 4\n0, 2, 1, 3, 2, 1, 1, 2, 2, 1, 0, 2,
3, 2\n1, 1, 3, 2, 0, 0, 0, 1, 2, 0, 2, 1, 1, 2\n1, 1, 3, 0, 0, 3, 1, 3, 1, 4, 0,
0, 0, 1\n0, 0, 6, 2, 1, 0, 0, 3, 2, 1, 1, 1, 0, 1\n0, 2, 4, 0, 1, 0, 1, 1, 1, 0,
0, 0, 0, 0\n1, 1, 0, 1, 0, 1, 0, 0, 2, 1, 1, 2, 0, 0\n0, 1, 1, 2, 0, 1, 0, 0, 0,
1, 1]\nclass = variable1'),
 Text(0.2857142857142857, 0.5833333333333334, 'designpatstock <= 7.5\ngini =
0.306\nsamples = 1923\nvalue = [2527, 300, 38, 22, 26, 11, 1, 10, 5, 2, 2, 3\n1,
7, 2, 5, 2, 5, 1, 3, 1, 0, 2, 2, 1, 0\n0, 0, 0, 2, 0, 0, 5, 3, 0, 0, 0, 0, 0,
0\n0, 2, 0, 1, 3, 1, 4, 0, 1, 0, 0, 0, 0, 0\n0, 4, 1, 0, 1, 0, 0, 0, 0, 2, 0, 0,
0, 0\n0, 3, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 5\n0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 1, 3, 0, 2, 3, 1, 4\n0, 0, 0, 0, 2, 0, 0, 2, 1, 0,
0, 0, 0, 0\n0, 0, 0, 2, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0\n0, 1, 0, 0, 0, 2, 0, 0, 0,
0, 0, 0, 0, 1\n0, 0, 3, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0]\nclass = variable1'),
 Text(0.14285714285714285, 0.4166666666666667, 'eca <= 0.5\ngini =
0.214\nsamples = 1773\nvalue = [2475, 281, 15, 5, 12, 6, 0, 0, 0, 0, 0, 0, 0\n0,
2, 0, 1, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0\n0, 0, 0, 0, 0, 3, 0, 1, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 1, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0]\nclass = variable1'),
 Text(0.07142857142857142, 0.25, 'designpatstock <= 0.5\ngini = 0.092\nsamples =
1387\nvalue = [2091, 60, 9, 4, 12, 4, 0, 0, 0, 0, 0, 0, 0\n0, 2, 0, 1, 0, 0, 2,
```

```
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 3,
0, 1, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 3, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 2,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.03571428571428571, 0.08333333333333333, 'gini = 0.0\nsamples =
1100\nvalue = [1735, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.10714285714285714, 0.08333333333333333, 'gini = 0.383\nsamples =
287\nvalue = [356, 60, 9, 4, 12, 4, 0, 0, 0, 0, 0, 0, 0, 0\n2, 0, 1, 0, 0, 2, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 3, 0,
1, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 3, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 2, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.21428571428571427, 0.25, 'lowrdperfprod <= 0.5\ngini = 0.479\nsamples =
386\nvalue = [384, 221, 6, 1, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.17857142857142858, 0.08333333333333333, 'gini = 0.503\nsamples =
313\nvalue = [277, 219, 3, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
```

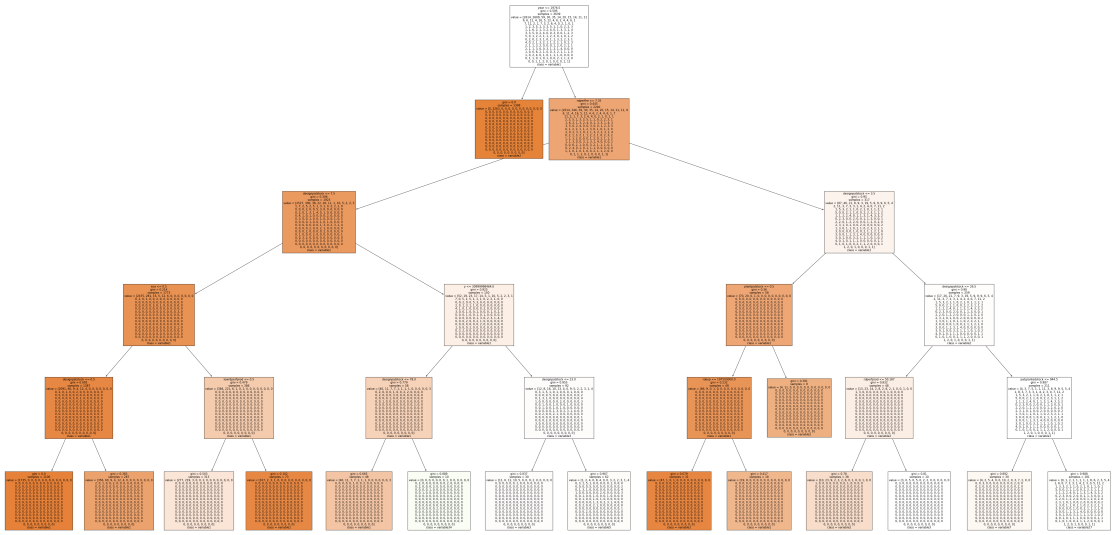
```
variable1'),
 Text(0.25, 0.08333333333333333, 'gini = 0.102\nsamples = 73\nvalue = [107, 2,
3, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass = variable1'),
 Text(0.42857142857142855, 0.4166666666666667, 'y <= 30999998464.0\ngini =
0.925\nsamples = 150\nvalue = [52, 19, 23, 17, 14, 5, 1, 10, 5, 2, 2, 3, 1\n7,
0, 5, 1, 5, 1, 1, 1, 0, 2, 2, 1, 0, 0\n0, 0, 2, 0, 0, 5, 3, 0, 0, 0, 0, 0, 0,
0\n2, 0, 1, 3, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0\n4, 1, 0, 1, 0, 0, 0, 0, 2, 0, 0, 0,
0, 0\n3, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 2, 0\n0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 1, 3, 0, 2, 3, 1, 4, 0\n0, 0, 0, 2, 0, 0, 2, 1, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0\n1, 0, 0, 0, 2, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 3, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0]\nclass = variable1'),
 Text(0.35714285714285715, 0.25, 'designpatstock <= 78.0\ngini = 0.779\nsamples
= 58\nvalue = [40, 11, 7, 7, 1, 1, 1, 1, 0, 0, 0, 0, 0, 3\n0, 2, 0, 0, 0, 1, 0,
0, 2, 2, 0, 0, 0, 0\n0, 2, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 1, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 2, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 2,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.32142857142857145, 0.08333333333333333, 'gini = 0.665\nsamples =
48\nvalue = [40, 11, 7, 7, 1, 1, 1, 0, 0, 0, 0, 0, 0, 3\n0, 0, 0, 0, 0, 1, 0, 0,
2, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.39285714285714285, 0.08333333333333333, 'gini = 0.889\nsamples =
10\nvalue = [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0\n0, 2, 0, 0, 0, 0, 0, 0,
0, 2, 0, 0, 0, 0\n0, 2, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 1, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 2, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 2, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 2, 0,
```

0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable34'),
 Text(0.5, 0.25, 'designpatstock <= 23.0\ngini = 0.955\nsamples = 92\nvalue =
[12, 8, 16, 10, 13, 4, 0, 9, 5, 2, 2, 3, 1, 4\n0, 3, 1, 5, 1, 0, 1, 0, 0, 0, 1,
0, 0, 0\n0, 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 0, 0, 2\n0, 1, 3, 1, 0, 0, 0, 0, 0, 0,
0, 0, 0, 4\n1, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 3\n0, 0, 1, 0, 0, 1, 0, 0, 0,
1, 0, 0, 0, 0\n0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 1, 3, 0,
2, 3, 1, 4, 0, 0\n0, 0, 2, 0, 0, 2, 1, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 1\n0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n3, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass = variable3'),
 Text(0.4642857142857143, 0.08333333333333333, 'gini = 0.837\nsamples =
30\nvalue = [11, 6, 12, 10, 5, 0, 0, 9, 2, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 1,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable3'),
 Text(0.5357142857142857, 0.08333333333333333, 'gini = 0.967\nsamples =
62\nvalue = [1, 2, 4, 0, 8, 4, 0, 0, 3, 2, 2, 3, 1, 4\n0, 3, 1, 5, 1, 0, 0, 0,
0, 0, 1, 0, 0, 0\n0, 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 0, 0, 2\n0, 1, 3, 1, 0, 0, 0,
0, 0, 0, 0, 0, 0, 4\n1, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 3\n0, 0, 1, 0, 0, 1,
0, 0, 0, 1, 0, 0, 0, 0\n0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
1, 3, 0, 2, 3, 1, 4, 0, 0\n0, 0, 2, 0, 0, 2, 1, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 1\n0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n3, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable5'),
 Text(0.7678571428571429, 0.5833333333333334, 'designpatstock <= 3.5\ngini =
0.95\nsamples = 317\nvalue = [87, 46, 21, 8, 9, 3, 19, 5, 9, 9, 9, 6, 5, 4\n2,
11, 3, 7, 3, 3, 1, 4, 2, 4, 0, 7, 11, 2\n1, 5, 3, 2, 1, 1, 0, 2, 1, 0, 1, 1, 2,
1\n5, 0, 0, 2, 1, 1, 0, 0, 2, 2, 3, 1, 1, 2\n1, 1, 2, 2, 4, 0, 1, 1, 3, 1, 4, 3,
3, 2\n0, 2, 3, 0, 0, 2, 0, 0, 1, 1, 3, 0, 0, 1\n2, 2, 0, 1, 2, 2, 0, 0, 0, 1, 2,
0, 2, 0\n2, 3, 1, 0, 1, 0, 0, 2, 0, 0, 0, 0, 0, 2\n1, 3, 0, 1, 1, 0, 1, 1, 0, 2,
3, 2, 1, 1\n3, 0, 0, 0, 0, 1, 0, 0, 2, 1, 1, 2, 1, 0\n3, 0, 0, 1, 1, 3, 1, 4, 0,
0, 0, 0, 0, 0\n3, 0, 1, 0, 0, 3, 2, 1, 1, 1, 0, 1, 0, 2\n4, 0, 1, 0, 1, 1, 1, 0,
0, 0, 0, 0, 1, 1\n0, 1, 0, 1, 0, 0, 2, 1, 1, 2, 0, 0, 0, 1\n1, 2, 0, 1, 0, 0, 0,
1, 1]\nclass = variable1'),
 Text(0.6785714285714286, 0.4166666666666667, 'plantpatstock <= 0.5\ngini =
0.36\nsamples = 58\nvalue = [70, 20, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

```
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0]\nclass = variable1'),
 Text(0.6428571428571429, 0.25, 'rdexp <= 197500000.0\ngini = 0.232\nsamples =
49\nvalue = [66, 9, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.6071428571428571, 0.08333333333333333, 'gini = 0.079\nsamples =
31\nvalue = [47, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.6785714285714286, 0.08333333333333333, 'gini = 0.417\nsamples =
18\nvalue = [19, 8, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable1'),
 Text(0.7142857142857143, 0.25, 'gini = 0.391\nsamples = 9\nvalue = [4, 11, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass = variable2'),
 Text(0.8571428571428571, 0.4166666666666667, 'designpatstock <= 26.5\ngini =
0.98\nsamples = 259\nvalue = [17, 26, 21, 7, 9, 3, 19, 5, 9, 9, 9, 6, 5, 4\n2,
11, 3, 7, 3, 3, 1, 4, 2, 4, 0, 7, 11, 2\n1, 5, 3, 2, 1, 1, 0, 2, 1, 0, 1, 1, 2,
```

```
1\n5, 0, 0, 2, 1, 1, 0, 0, 2, 2, 3, 1, 1, 2\n1, 1, 2, 2, 4, 0, 1, 1, 3, 1, 4, 3,
3, 2\n0, 2, 3, 0, 0, 2, 0, 0, 1, 1, 3, 0, 0, 1\n2, 2, 0, 1, 2, 2, 0, 0, 0, 1, 2,
0, 2, 0\n2, 3, 1, 0, 1, 0, 0, 2, 0, 0, 0, 0, 0, 2\n1, 3, 0, 1, 1, 0, 1, 1, 0, 2,
3, 2, 1, 1\n3, 0, 0, 0, 0, 1, 0, 0, 2, 1, 1, 2, 1, 0\n3, 0, 0, 1, 1, 3, 1, 4, 0,
0, 0, 0, 0, 0\n3, 0, 1, 0, 0, 3, 2, 1, 1, 1, 0, 1, 0, 2\n4, 0, 1, 0, 1, 1, 1, 0,
0, 0, 0, 0, 1, 1\n0, 1, 0, 1, 0, 0, 2, 1, 1, 2, 0, 0, 0, 1\n1, 2, 0, 1, 0, 0, 0,
1, 1]\nclass = variable2'),
 Text(0.7857142857142857, 0.25, 'rdperfprod <= 50.187\ngini = 0.832\nsamples =
48\nvalue = [13, 23, 14, 2, 8, 2, 8, 2, 1, 0, 0, 1, 0, 0\n1, 3, 0, 0, 0, 0, 0,
0, 0, 1, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable2'),
 Text(0.75, 0.08333333333333333, 'gini = 0.78\nsamples = 38\nvalue = [10, 23, 9,
2, 3, 2, 6, 0, 1, 0, 0, 1, 0, 0\n1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass = variable2'),
 Text(0.8214285714285714, 0.08333333333333333, 'gini = 0.81\nsamples = 10\nvalue
= [3, 0, 5, 0, 5, 0, 2, 2, 0, 0, 0, 0, 0, 0\n0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass = variable3'),
 Text(0.9285714285714286, 0.25, 'patgrantedstock <= 944.5\ngini = 0.987\nsamples
= 211\nvalue = [4, 3, 7, 5, 1, 1, 11, 3, 8, 9, 9, 5, 5, 4\n1, 8, 3, 7, 3, 3, 1,
4, 2, 3, 0, 7, 11, 2\n1, 5, 3, 2, 1, 1, 0, 2, 1, 0, 1, 1, 2, 1\n5, 0, 0, 2, 1,
1, 0, 0, 2, 2, 3, 1, 1, 2\n1, 1, 2, 2, 4, 0, 1, 1, 3, 1, 4, 3, 3, 2\n0, 2, 3, 0,
0, 2, 0, 0, 1, 1, 3, 0, 0, 1\n2, 2, 0, 1, 2, 2, 0, 0, 0, 1, 2, 0, 2, 0\n2, 3, 1,
0, 1, 0, 0, 2, 0, 0, 0, 0, 0, 2\n1, 3, 0, 1, 1, 0, 1, 1, 0, 2, 3, 2, 1, 1\n3, 0,
0, 0, 0, 1, 0, 0, 2, 1, 1, 2, 1, 0\n3, 0, 0, 1, 1, 3, 1, 4, 0, 0, 0, 0, 0, 0\n3,
0, 1, 0, 0, 3, 2, 1, 1, 1, 0, 1, 0, 2\n4, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1,
1\n0, 1, 0, 1, 0, 0, 2, 1, 1, 2, 0, 0, 0, 1\n1, 2, 0, 1, 0, 0, 0, 1, 1]\nclass =
variable7'),
 Text(0.8928571428571429, 0.08333333333333333, 'gini = 0.892\nsamples =
30\nvalue = [4, 2, 5, 4, 0, 0, 10, 2, 0, 3, 7, 2, 0, 0\n0, 0, 3, 0, 0, 0, 0, 0,
```

```
0, 0, 0, 2, 0, 0\n0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\n0, 0, 0, 0, 0, 0, 0, 0, 0]\nclass =
variable7'),
 Text(0.9642857142857143, 0.08333333333333333, 'gini = 0.988\nsamples =
181\nvalue = [0, 1, 2, 1, 1, 1, 1, 1, 8, 6, 2, 3, 5, 4\n1, 8, 0, 7, 3, 3, 1, 4,
2, 3, 0, 5, 11, 2\n1, 3, 3, 2, 1, 1, 0, 2, 1, 0, 1, 1, 0, 1\n5, 0, 0, 2, 1, 1,
0, 0, 2, 2, 3, 1, 1, 2\n1, 1, 2, 2, 4, 0, 1, 1, 3, 1, 4, 3, 3, 2\n0, 2, 3, 0, 0,
2, 0, 0, 1, 1, 3, 0, 0, 1\n2, 2, 0, 1, 2, 2, 0, 0, 0, 1, 2, 0, 2, 0\n2, 3, 1, 0,
1, 0, 0, 2, 0, 0, 0, 0, 0, 2\n1, 3, 0, 1, 1, 0, 1, 1, 0, 2, 3, 2, 1, 1\n3, 0, 0,
0, 0, 1, 0, 0, 2, 1, 1, 2, 1, 0\n3, 0, 0, 1, 1, 3, 1, 4, 0, 0, 0, 0, 0, 0\n3, 0,
1, 0, 0, 3, 2, 1, 1, 1, 0, 1, 0, 2\n4, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1\n0,
1, 0, 1, 0, 0, 2, 1, 1, 2, 0, 0, 0, 1\n1, 2, 0, 1, 0, 0, 0, 1, 1]\nclass =
variable27')]
```



23 Conclusion

24 Accuracy

[62]:
```
print("Linear Regression:",lr.score(x_test,y_test))
print("Ridge Regression:",rr.score(x_test,y_test))
print("Lasso Regression",la.score(x_test,y_test))
print("ElasticNet Regression:",en.score(x_test,y_test))
```

```
print("Logistic Regression:",logr.score(fs,target_vector))
print("Random Forest:",grid_search.best_score_)
```

```
Linear Regression: 0.901608816724942
Ridge Regression: 0.9015866978331674
Lasso Regression 0.9015386696202259
ElasticNet Regression: 0.9009443270496029
Logistic Regression: 0.8764315852923448
Random Forest: 0.8837409576300379
```

25 Linear Regression is suitable for this dataset