```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import pandas as pd
import io


data = pd.read_csv('coffee_dataset.csv')
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1339 entries, 0 to 1338
Data columns (total 44 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Unnamed: 0              1339 non-null   int64  
 1   Species                 1339 non-null   object 
 2   Owner                   1332 non-null   object 
 3   Country.of.Origin       1338 non-null   object 
 4   Farm.Name               980 non-null    object 
 5   Lot.Number              276 non-null    object 
 6   Mill                    1021 non-null   object 
 7   ICO.Number              1182 non-null   object 
 8   Company                 1130 non-null   object 
 9   Altitude                1113 non-null   object 
 10  Region                  1280 non-null   object 
 11  Producer                1107 non-null   object 
 12  Number.of.Bags          1339 non-null   int64  
 13  Bag.Weight              1339 non-null   object 
 14  In.Country.Partner      1339 non-null   object 
 15  Harvest.Year            1292 non-null   object 
 16  Grading.Date            1339 non-null   object 
 17  Owner.1                 1332 non-null   object 
 18  Variety                 1113 non-null   object 
 19  Processing.Method       1169 non-null   object 
 20  Aroma                   1339 non-null   float64
 21  Flavor                  1339 non-null   float64
 22  Aftertaste              1339 non-null   float64
 23  Acidity                 1339 non-null   float64
 24  Body                    1339 non-null   float64
 25  Balance                 1339 non-null   float64
 26  Uniformity              1339 non-null   float64
 27  Clean.Cup               1339 non-null   float64
 28  Sweetness               1339 non-null   float64
 29  Cupper.Points           1339 non-null   float64
 30  Total.Cup.Points        1339 non-null   float64
 31  Moisture                1339 non-null   float64
 32  Category.One.Defects    1339 non-null   int64  
 33  Quakers                 1338 non-null   float64
 34  Color                   1121 non-null   object 
 35  Category.Two.Defects    1339 non-null   int64  
 36  Expiration              1339 non-null   object 
 37  Certification.Body      1339 non-null   object 
 38  Certification.Address   1339 non-null   object 
 39  Certification.Contact   1339 non-null   object 
 40  unit_of_measurement     1339 non-null   object 
 41  altitude_low_meters     1109 non-null   float64
 42  altitude_high_meters    1109 non-null   float64
 43  altitude_mean_meters    1109 non-null   float64
```

```
dtypes: float64(16), int64(4), object(24)
memory usage: 460.4+ KB
```

```
data.head()
```

	Unnamed: 0	Species	Owner	Country.of.Origin	Farm.Name	Lot.Number	Mill
0	0	Arabica	metad plc	Ethiopia	metad plc	NaN	metad plc
1	1	Arabica	metad plc	Ethiopia	metad plc	NaN	metad plc
2	2	Arabica	grounds for health admin	Guatemala	san marcos barrancas "san cristobal cuch	NaN	NaN
3	3	Arabica	yidnekachew dabessa	Ethiopia	yidnekachew dabessa coffee plantation	NaN	wolensu
4	4	Arabica	metad plc	Ethiopia	metad plc	NaN	metad plc

```
from sklearn.preprocessing import LabelEncoder , OneHotEncoder
data['Species'].value_counts()
```

```
Arabica    1311
Robusta      28
Name: Species, dtype: int64
```

```
le=LabelEncoder()
data['Number.of.Bags']=le.fit_transform(data['Number.of.Bags'])
data['Number.of.Bags'].value_counts()
```

```
104    242
110    176
10     108
1       95
119     79
      ...
75       1
77       1
78       1
79       1
```

```
0        1
Name: Number.of.Bags, Length: 131, dtype: int64
```

```
le.classes_
```

```
array([   0,    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,   31,   32,
          33,   35,   36,   37,   38,   39,   40,   42,   43,   44,   45,
          48,   49,   50,   51,   53,   54,   56,   58,   60,   62,   65,
          66,   69,   70,   74,   75,   77,   80,   82,   84,   85,   90,
          93,   94,  100,  114,  120,  121,  123,  125,  127,  129,  130,
         134,  135,  138,  140,  149,  150,  160,  165,  166,  167,  170,
         175,  180,  198,  200,  202,  209,  220,  223,  226,  230,  232,
         235,  240,  243,  245,  248,  250,  252,  253,  256,  270,  274,
         275,  280,  285,  288,  300,  302,  304,  305,  310,  320,  325,
         360,  377,  380,  400,  440,  450,  500,  550,  600, 1062])
```

```
data['In.Country.Partner'].value_counts()
```

```
Specialty Coffee Association
AMECAFE
Almacafé
Asociacion Nacional Del Café
Brazil Specialty Coffee Association
Instituto Hondureño del Café
Blossom Valley International
Africa Fine Coffee Association
Specialty Coffee Association of Costa Rica
NUCOFFEE
Uganda Coffee Development Authority
Kenya Coffee Traders Association
Ethiopia Commodity Exchange
Specialty Coffee Institute of Asia
METAD Agricultural Development plc
Yunnan Coffee Exchange
Salvadoran Coffee Council
Specialty Coffee Association of Indonesia
Centro Agroecológico del Café A.C.
Asociación de Cafés Especiales de Nicaragua
Coffee Quality Institute
Asociación Mexicana De Cafés y Cafeterías De Especialidad A.C.
Tanzanian Coffee Board
Torch Coffee Lab Yunnan
Blossom Valley International\n
Central De Organizaciones Productoras De Café y Cacao Del Perú - Central Café & Cacac
Specialty Coffee Ass
Name: In.Country.Partner, dtype: int64
```

```
one_hot = OneHotEncoder()
transformed_data = one_hot.fit_transform(data['In.Country.Partner'].values.reshape(-1,1)).
one_hot.categories_
```

```
[array(['AMECAFE', 'Africa Fine Coffee Association', 'Almacafé',
        'Asociacion Nacional Del Café',
        'Asociación Mexicana De Cafés y Cafeterías De Especialidad A.C.',
        'Asociación de Cafés Especiales de Nicaragua',
```

```
            'Blossom Valley International', 'Blossom Valley International\n',
            'Brazil Specialty Coffee Association',
            'Central De Organizaciones Productoras De Café y Cacao Del Perú - Central Caf
            'Centro Agroecológico del Café A.C.', 'Coffee Quality Institute',
            'Ethiopia Commodity Exchange', 'Instituto Hondureño del Café',
            'Kenya Coffee Traders Association',
            'METAD Agricultural Development plc', 'NUCOFFEE',
            'Salvadoran Coffee Council', 'Specialty Coffee Ass',
            'Specialty Coffee Association',
            'Specialty Coffee Association of Costa Rica',
            'Specialty Coffee Association of Indonesia',
            'Specialty Coffee Institute of Asia', 'Tanzanian Coffee Board',
            'Torch Coffee Lab Yunnan', 'Uganda Coffee Development Authority',
            'Yunnan Coffee Exchange'], dtype=object)]
```

```
transformed_data = pd.DataFrame(transformed_data ,
                                columns = ['AMECAFE', 'Africa Fine Coffee Association', 'A
        'Asociacion Nacional Del Café',
        'Asociación Mexicana De Cafés y Cafeterías De Especialidad A.C.',
        'Asociación de Cafés Especiales de Nicaragua',
        'Blossom Valley International', 'Blossom Valley International\n',
        'Brazil Specialty Coffee Association',
        'Central De Organizaciones Productoras De Café y Cacao Del Perú - Central Café & C
        'Centro Agroecológico del Café A.C.', 'Coffee Quality Institute',
        'Ethiopia Commodity Exchange', 'Instituto Hondureño del Café',
        'Kenya Coffee Traders Association',
        'METAD Agricultural Development plc', 'NUCOFFEE',
        'Salvadoran Coffee Council', 'Specialty Coffee Ass',
        'Specialty Coffee Association',
        'Specialty Coffee Association of Costa Rica',
        'Specialty Coffee Association of Indonesia',
        'Specialty Coffee Institute of Asia', 'Tanzanian Coffee Board',
        'Torch Coffee Lab Yunnan', 'Uganda Coffee Development Authority',
        'Yunnan Coffee Exchange'])
transformed_data.head()
```

	AMECAFE	Africa Fine Coffee Association	Almacafé	Asociacion Nacional Del Café	Asociación Mexicana De Cafés y Cafeterías De Especialidad A.C.	Asociación de Cafés Especiales de Nicaragua	Blosso Valle Internationa
0	0.0	0.0	0.0	0.0	0.0	0.0	0
1	0.0	0.0	0.0	0.0	0.0	0.0	0
2	0.0	0.0	0.0	0.0	0.0	0.0	0
3	0.0	0.0	0.0	0.0	0.0	0.0	0
4	0.0	0.0	0.0	0.0	0.0	0.0	0

```
transformed_data.iloc[90, ]
```

```
AMECAFE
Africa Fine Coffee Association
Almacafé
Asociacion Nacional Del Café
Asociación Mexicana De Cafés y Cafeterías De Especialidad A.C.
Asociación de Cafés Especiales de Nicaragua
Blossom Valley International
Blossom Valley International\n
Brazil Specialty Coffee Association
Central De Organizaciones Productoras De Café y Cacao Del Perú - Central Café & Cacao
Centro Agroecológico del Café A.C.
Coffee Quality Institute
Ethiopia Commodity Exchange
Instituto Hondureño del Café
Kenya Coffee Traders Association
METAD Agricultural Development plc
NUCOFFEE
Salvadoran Coffee Council
Specialty Coffee Ass
Specialty Coffee Association
Specialty Coffee Association of Costa Rica
Specialty Coffee Association of Indonesia
Specialty Coffee Institute of Asia
Tanzanian Coffee Board
Torch Coffee Lab Yunnan
Uganda Coffee Development Authority
Yunnan Coffee Exchange
Name: 90, dtype: float64
```

```
data['Number.of.Bags'][90]
```

```
68
```

```
numeric_columns = [c for c in data.columns if data[c].dtype != np.dtype('O')]
numeric_columns
```

```
['Unnamed: 0',
 'Number.of.Bags',
 'Aroma',
 'Flavor',
 'Aftertaste',
 'Acidity',
 'Body',
 'Balance',
 'Uniformity',
 'Clean.Cup',
 'Sweetness',
 'Cupper.Points',
 'Total.Cup.Points',
 'Moisture',
 'Category.One.Defects',
 'Quakers',
 'Category.Two.Defects',
 'altitude_low_meters',
 'altitude_high_meters',
 'altitude_mean_meters']
```

```
len(numeric_columns) , len(data.columns)
```

```
(20, 44)
```

```
numeric_columns.remove('Aroma')
numeric_columns.remove('Flavor')
```

```
temp_data = data[numeric_columns]
temp_data
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity	Clear
0	0	114	8.67	8.75	8.50	8.42	10.00	
1	1	114	8.50	8.58	8.42	8.42	10.00	
2	2	5	8.42	8.42	8.33	8.42	10.00	
3	3	119	8.42	8.42	8.50	8.25	10.00	
4	4	114	8.25	8.50	8.42	8.33	10.00	
...	...	...	...	...	...	...	...	
1334	1334	1	7.33	7.58	5.08	7.83	10.00	
1335	1335	1	7.75	7.75	5.17	5.25	10.00	
1336	1336	1	7.17	7.42	7.50	7.17	9.33	
1337	1337	1	6.75	7.17	7.25	7.00	9.33	
1338	1338	1	6.50	6.83	6.92	6.83	9.33	

1339 rows × 18 columns

```
from sklearn.preprocessing import StandardScaler , MinMaxScaler
import warnings
warnings.filterwarnings('ignore')
normalizer = MinMaxScaler()
temp_data.dropna(axis = 1 , inplace = True)
normalized_data = normalizer.fit_transform(temp_data)
pd.DataFrame(normalized_data , columns = temp_data.columns)
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity
0	0.000000	0.876923	1.000000	1.000000	0.990676	0.962286	1.000
1	0.000747	0.876923	0.980392	0.980571	0.981352	0.962286	1.000
2	0.001495	0.038462	0.971165	0.962286	0.970862	0.962286	1.000
3	0.002242	0.915385	0.971165	0.962286	0.990676	0.942857	1.000
4	0.002990	0.876923	0.951557	0.971429	0.981352	0.952000	1.000

```
standard_scaler = StandardScaler()
standardized_data = standard_scaler.fit_transform(temp_data)
pd.DataFrame(standardized_data , columns = temp_data.columns)
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity
0	-1.730758	1.032078	3.138457	3.198164	2.655944	2.206476	0.29785
1	-1.728171	1.032078	2.717990	2.750424	2.439684	2.206476	0.29785
2	-1.725584	-1.359565	2.520123	2.329022	2.196392	2.206476	0.29785
3	-1.722997	1.141786	2.520123	2.329022	2.655944	1.790615	0.29785
4	-1.720409	1.032078	2.099656	2.539723	2.439684	1.986314	0.29785
...	...	...	...	...	...	...	...
1334	1.720409	-1.447331	-0.175812	0.116661	-6.589155	0.763194	0.29785
1335	1.722997	-1.447331	0.862989	0.564400	-6.345863	-5.548106	0.29785
1336	1.725584	-1.447331	-0.571545	-0.304742	-0.047302	-0.851325	-0.91070
1337	1.728171	-1.447331	-1.610346	-0.963182	-0.723113	-1.267185	-0.91070
1338	1.730758	-1.447331	-2.228680	-1.858662	-1.615184	-1.683046	-0.91070

1339 rows × 14 columns

```
data.isnull().sum()
```

```
Unnamed: 0                0
Species                   0
Owner                     7
Country.of.Origin         1
Farm.Name               359
Lot.Number             1063
Mill                    318
ICO.Number              157
Company                 209
Altitude                226
Region                   59
Producer                232
Number.of.Bags            0
Bag.Weight                0
```

```
In.Country.Partner          0
Harvest.Year               47
Grading.Date                0
Owner.1                     7
Variety                   226
Processing.Method         170
Aroma                       0
Flavor                      0
Aftertaste                  0
Acidity                     0
Body                        0
Balance                     0
Uniformity                  0
Clean.Cup                   0
Sweetness                   0
Cupper.Points               0
Total.Cup.Points            0
Moisture                    0
Category.One.Defects        0
Quakers                     1
Color                     218
Category.Two.Defects        0
Expiration                  0
Certification.Body          0
Certification.Address       0
Certification.Contact       0
unit_of_measurement         0
altitude_low_meters       230
altitude_high_meters      230
altitude_mean_meters      230
dtype: int64
```

```
data['altitude_low_meters'].isnull().sum()
```

```
230
```

```
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(missing_values=np.nan , strategy='mean')
agent_col = imputer.fit_transform(data['altitude_low_meters'].values.reshape(-1,1))
pd.DataFrame(agent_col).isnull().sum()
```

```
0    0
dtype: int64
```

```
data['altitude_low_meters'].isnull().sum()
```

```
230
```

```
from sklearn.preprocessing import KBinsDiscretizer
temp_data.head()
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity	Clean.Cu
0	0	114	8.67	8.75	8.50	8.42	10.0	10
1	1	114	8.50	8.58	8.42	8.42	10.0	10

```
trans = KBinsDiscretizer(n_bins =10 , encode = 'ordinal' , strategy='quantile')
new_data = trans.fit_transform(temp_data)
pd.DataFrame(new_data,columns = temp_data.columns )
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity	Clear
0	0.0	8.0	9.0	8.0	8.0	8.0	1.0	
1	0.0	8.0	9.0	8.0	8.0	8.0	1.0	
2	0.0	1.0	9.0	8.0	8.0	8.0	1.0	
3	0.0	8.0	9.0	8.0	8.0	8.0	1.0	
4	0.0	8.0	9.0	8.0	8.0	8.0	1.0	
...	...	...	...	...	...	...	...	
1334	9.0	0.0	4.0	5.0	0.0	7.0	1.0	
1335	9.0	0.0	8.0	7.0	0.0	0.0	1.0	
1336	9.0	0.0	2.0	3.0	4.0	1.0	1.0	
1337	9.0	0.0	0.0	1.0	1.0	0.0	1.0	
1338	9.0	0.0	0.0	0.0	0.0	0.0	1.0	

1339 rows × 14 columns

```
trans = KBinsDiscretizer(n_bins =10 , encode = 'ordinal' , strategy='uniform')
new_data = trans.fit_transform(temp_data)


pd.DataFrame(new_data,columns = temp_data.columns )
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity	Clear
0	0.0	8.0	9.0	9.0	9.0	9.0	9.0	
1	0.0	8.0	9.0	9.0	9.0	9.0	9.0	

```
trans = KBinsDiscretizer(n_bins =10 , encode = 'ordinal' , strategy='kmeans')
new_data = trans.fit_transform(temp_data)

pd.DataFrame(new_data,columns = temp_data.columns )
```

	Unnamed: 0	Number.of.Bags	Aftertaste	Acidity	Body	Balance	Uniformity	Clear
0	0.0	8.0	9.0	9.0	8.0	9.0	6.0	
1	0.0	8.0	8.0	8.0	8.0	9.0	6.0	
2	0.0	0.0	8.0	8.0	8.0	9.0	6.0	
3	0.0	9.0	8.0	8.0	8.0	9.0	6.0	
4	0.0	8.0	7.0	8.0	8.0	9.0	6.0	
...	...	...	...	...	...	...	...	
1334	9.0	0.0	4.0	5.0	1.0	9.0	6.0	
1335	9.0	0.0	5.0	5.0	2.0	1.0	6.0	
1336	9.0	0.0	3.0	5.0	5.0	7.0	5.0	
1337	9.0	0.0	2.0	4.0	5.0	5.0	5.0	
1338	9.0	0.0	1.0	3.0	4.0	3.0	5.0	

1339 rows × 14 columns


0s
completed at 10:36