In [ ]:

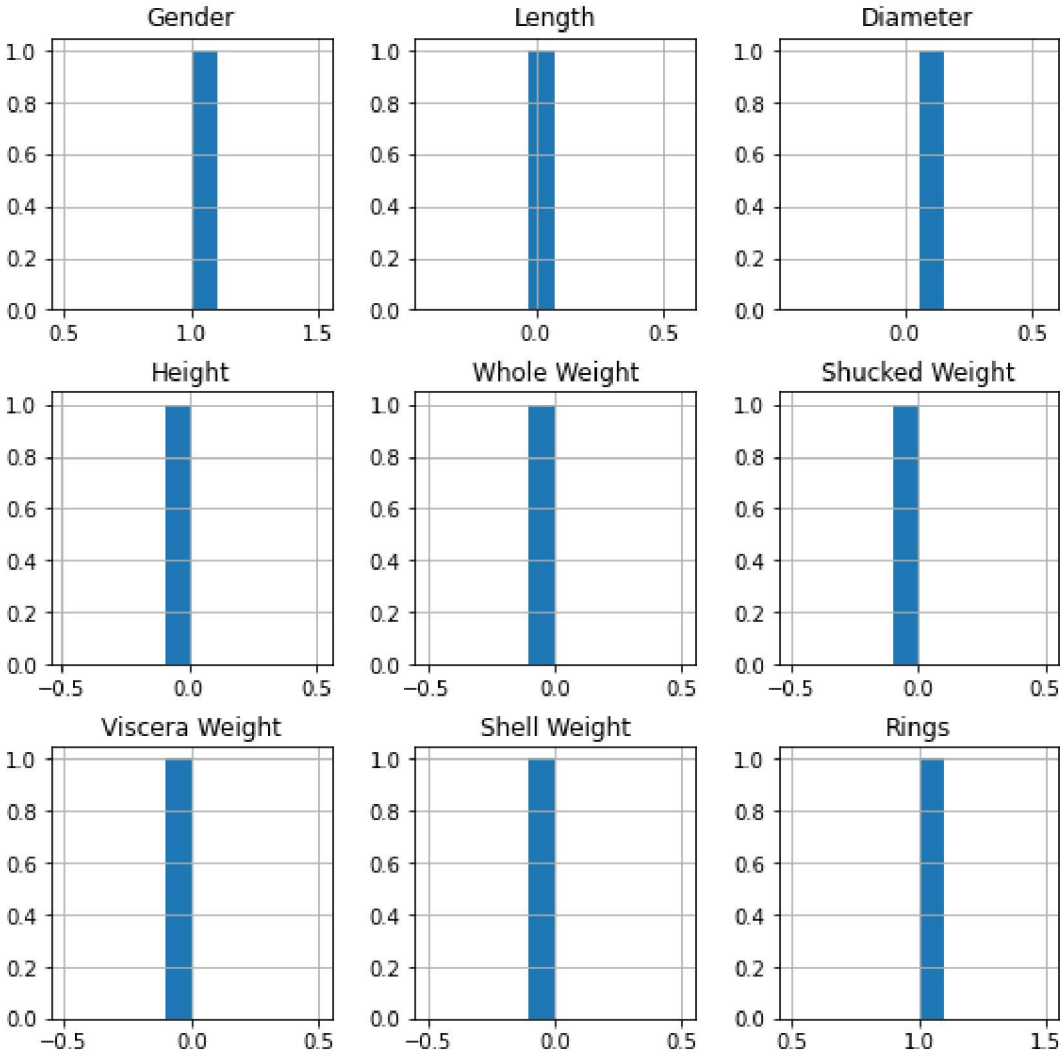
```
from numpy.lib.function_base import average
import pandas as pd
import numpy as np
from six import print_
from sklearn import preprocessing
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.naive_bayes import GaussianNB
from sklearn import svm
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.datasets import make_classification
from sklearn.model_selection import LeaveOneOut
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LogisticRegression
from matplotlib import pyplot

dataset = pd.read_csv('./abalone.data')
dataset.columns = ['Gender', 'Length', 'Diameter', 'Height', 'Whole Weight',
                   'Shucked Weight', 'Viscera Weight', 'Shell Weight', 'Rings']

_label = preprocessing.LabelEncoder()
dataset['Gender'] = _label.fit_transform(dataset['Gender'])
print("First 8 data\n", dataset.head(8))
print("Last 5 data\n", dataset.tail(5))
print("Checking data set\n" ,dataset.isnull().sum(axis = 0))
print("Describtion of the data\n" ,dataset.describe())
print("column names\n", dataset.columns)
print("The shape of the data\n", dataset.shape)
X = dataset[['Gender', 'Length', 'Diameter', 'Height', 'Whole Weight',
                    'Shucked Weight', 'Viscera Weight', 'Shell Weight']]
X.columns = ['Gender', 'Length', 'Diameter', 'Height', 'Whole Weight',
                    'Shucked Weight', 'Viscera Weight', 'Shell Weight']
Y = dataset['Rings']
dataset.groupby('Rings').hist(figsize=(9,9))
plt.show()

X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size= 0.25, random_sta

DecisionTree = DecisionTreeClassifier(criterion = "entropy", max_depth =3)
DecisionTree.fit(X_train, Y_train)
y_pred = DecisionTree.predict(X_test)


print("Decision Tree Accuracy Score is: ", metrics.accuracy_score(Y_test,y_pred))
print("Decision Tree F1 Score is: ", metrics.f1_score(Y_test,y_pred, average='macro'
print("Decision Tree Recall Score is: ", metrics.recall_score(Y_test,y_pred, average
print("Decision Tree Precision Score is: ", metrics.precision_score(Y_test,y_pred, a
print("Decision Tree Confusion Score is: ", metrics.confusion_matrix(Y_test,y_pred))

cnf_matrix = confusion_matrix(Y_test, y_pred)
p = sns.heatmap(cnf_matrix, annot = True, cmap = 'YlGnBu', fmt = 'g')
plt.title('Decision Tree Confusion matrix', y=1.1)
plt.ylabel('Acutal')
plt.xlabel('Predicted')
plt.show()

Bayes = GaussianNB()
```

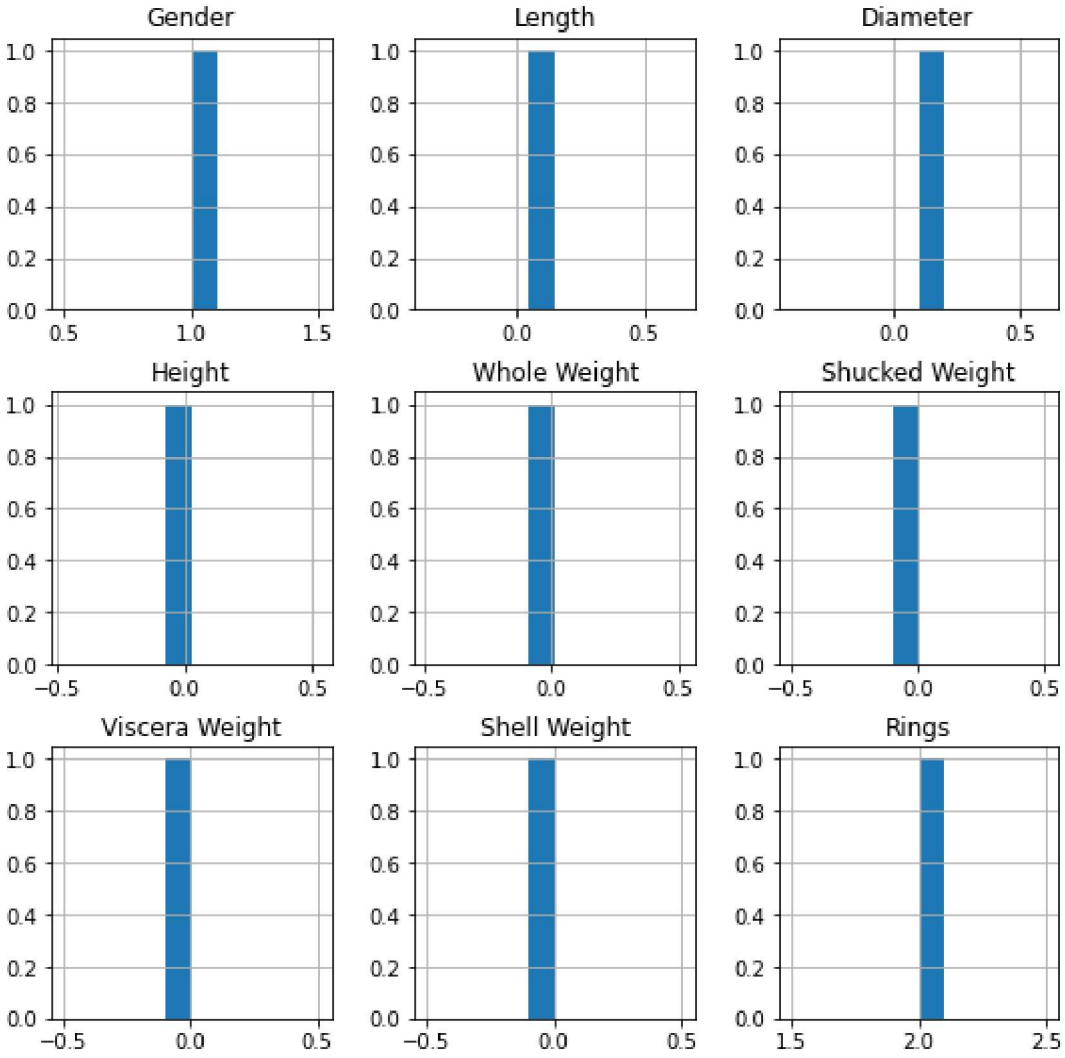
```
Bayes.fit(X_train,Y_train)
Bayes_pred = Bayes.predict(X_test)

print("\nBayes Accuracy Score is: ", metrics.accuracy_score(Y_test,Bayes_pred))
print("Bayes Accuracy Score is: ", metrics.accuracy_score(Y_test,Bayes_pred))
print("Bayes F1 Score is: ", metrics.f1_score(Y_test,Bayes_pred, average='macro'))
print("Bayes Recall Score is: ", metrics.recall_score(Y_test,Bayes_pred, average = '
print("Bayes Precision Score is: ", metrics.precision_score(Y_test,Bayes_pred, avera
print("Bayes Confusion Score is: ", metrics.confusion_matrix(Y_test,Bayes_pred))

cnf_matrix = confusion_matrix(Y_test, Bayes_pred)
p = sns.heatmap(cnf_matrix, annot = True, cmap = 'YlGnBu', fmt = 'g')
plt.title('Bayes Confusion matrix', y=1.1)
plt.ylabel('Acutal')
plt.xlabel('Predicted')
plt.show()

Knn = KNeighborsClassifier(n_neighbors = 3)
Knn.fit(X_train, Y_train)
Knn_pred = Knn.predict(X_test)

print("\nKnn Accuracy Score is: ", metrics.accuracy_score(Y_test,Knn_pred))
print("Knn Accuracy Score is: ", metrics.accuracy_score(Y_test,Knn_pred))
print("Knn F1 Score is: ", metrics.f1_score(Y_test,Knn_pred, average='macro'))
print("Knn Recall Score is: ", metrics.recall_score(Y_test,Knn_pred, average = 'macr
print("Knn Precision Score is: ", metrics.precision_score(Y_test,Knn_pred, average =
print("Knn Confusion Score is: ", metrics.confusion_matrix(Y_test,Knn_pred))

cnf_matrix = confusion_matrix(Y_test, Knn_pred)
p = sns.heatmap(cnf_matrix, annot = True, cmap = 'YlGnBu', fmt = 'g')
plt.title('Knn Confusion matrix', y=1.1)
plt.ylabel('Acutal')
plt.xlabel('Predicted')
plt.show()

Svc = svm.SVC(kernel='linear', C = 1, gamma = 'auto').fit(X_train, Y_train)
Svc_pred = Svc.predict(X_test)

print("\nSVC Accuracy Score is: ", metrics.accuracy_score(Y_test,Svc_pred))
print("SVC F1 Score is: ", metrics.f1_score(Y_test,Svc_pred, average='macro'))
print("SVC Recall Score is: ", metrics.recall_score(Y_test,Svc_pred, average = 'macr
print("SVC Precision Score is: ", metrics.precision_score(Y_test,Svc_pred, average =
print("SVC Confusion Score is: ", metrics.confusion_matrix(Y_test,Svc_pred))

cnf_matrix = confusion_matrix(Y_test, Svc_pred)
p = sns.heatmap(cnf_matrix, annot = True, cmap = 'YlGnBu', fmt = 'g')
plt.title('SVC Confusion matrix', y=1.1)
plt.ylabel('Acutal')
plt.xlabel('Predicted')
plt.show()


k = 5
kf = KFold(n_splits=k, random_state=None)
kflodDecision = DecisionTreeClassifier(criterion='entropy', max_depth=3)
accuracy_score = []
f1_score = []
recall_score = []
precision_score = []
confusion_score = []

for train_index, test_index in kf.split(X):
    X_train, X_test = X.iloc[train_index, :], X.iloc[test_index, :]
    Y_train, Y_test = Y[train_index], Y[test_index]
```

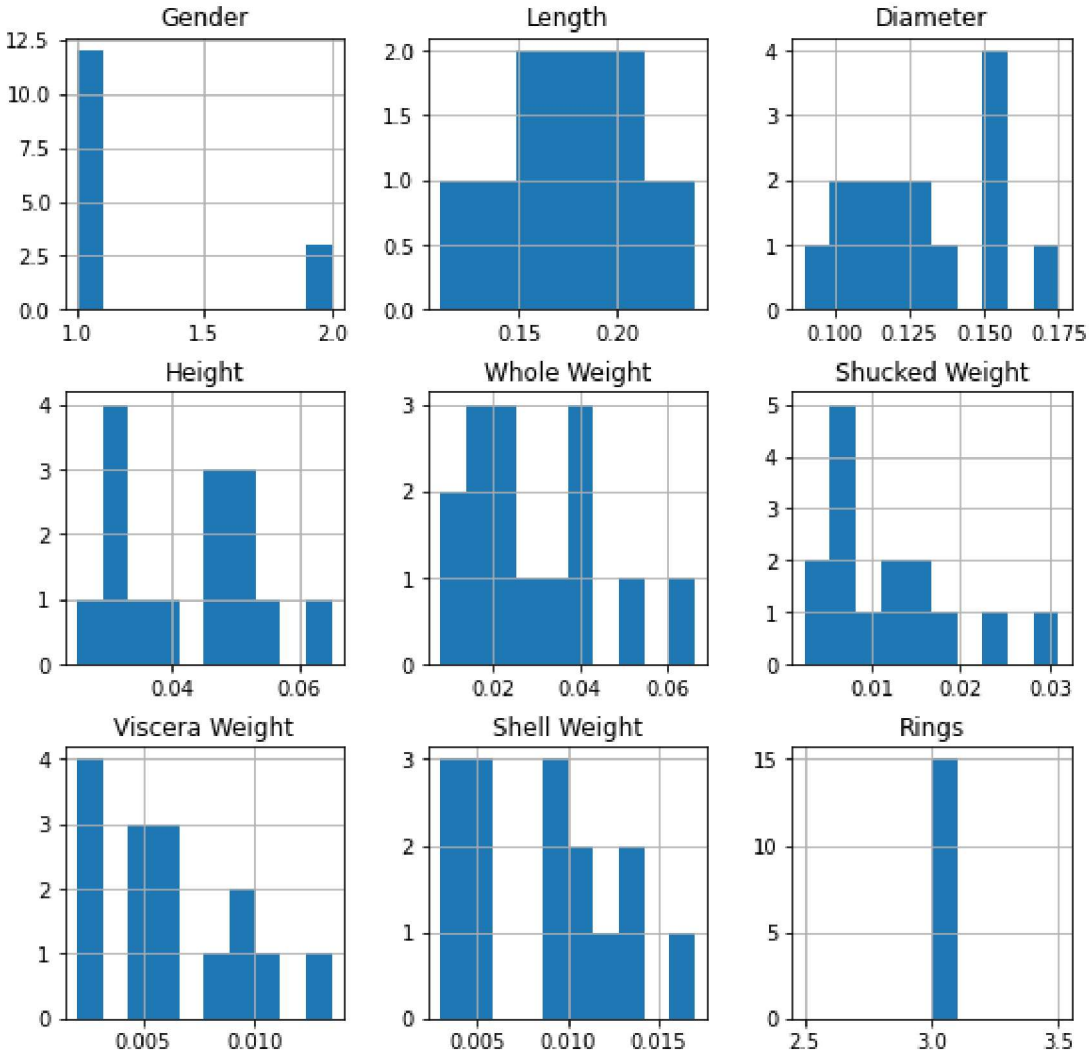
```
    kflodDecision.fit(X_train, Y_train)
    predicted_values = kflodDecision.predict(X_test)

    accuracy_score.append(metrics.accuracy_score(Y_test, predicted_values))
    f1_score.append(metrics.f1_score(Y_test, predicted_values, average = "macro"))
    recall_score.append(metrics.recall_score(Y_test, predicted_values, average = "ma
    precision_score.append(metrics.precision_score(Y_test, predicted_values, average

average_accuracy_score = sum(accuracy_score)/k
print("Accuracy of each flod is: {}".format(accuracy_score))
print("Average accuracy is: {}".format(average_accuracy_score))
average_f1_score = sum(f1_score)/k
print("F1 of each flod is {}:".format(f1_score))
print("Average F1 Score is: {}".format(average_f1_score))
average_recall_score = sum(recall_score)/k
print("Recall of each flod is: {}".format(recall_score))
print("Average recall score is: {}".format(average_recall_score))
average_precision_score = sum(precision_score)/k
print("Precision of each flod is: {}".format(precision_score))
print("Average precision score is: {}".format(average_precision_score))
```

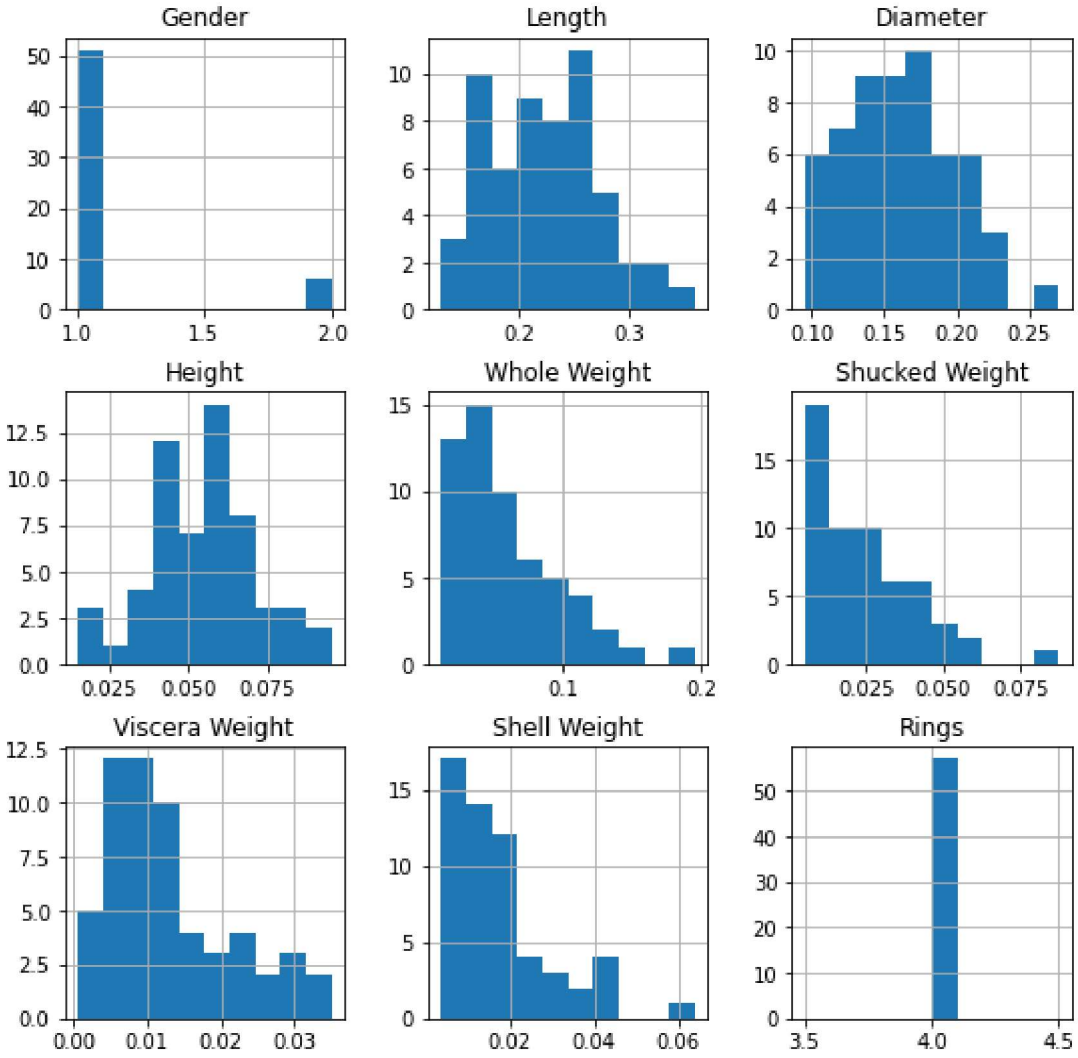
```
First 8 data
    Gender  Length  Diameter  Height  Whole Weight  Shucked Weight  \
0       2   0.350     0.265   0.090        0.2255          0.0995
1       0   0.530     0.420   0.135        0.6770          0.2565
2       2   0.440     0.365   0.125        0.5160          0.2155
3       1   0.330     0.255   0.080        0.2050          0.0895
4       1   0.425     0.300   0.095        0.3515          0.1410
5       0   0.530     0.415   0.150        0.7775          0.2370
6       0   0.545     0.425   0.125        0.7680          0.2940
7       2   0.475     0.370   0.125        0.5095          0.2165

   Viscera Weight  Shell Weight  Rings
0          0.0485         0.070      7
1          0.1415         0.210      9
2          0.1140         0.155     10
3          0.0395         0.055      7
4          0.0775         0.120      8
5          0.1415         0.330     20
6          0.1495         0.260     16
7          0.1125         0.165      9
Last 5 data
      Gender  Length  Diameter  Height  Whole Weight  Shucked Weight  \
4171       0   0.565     0.450   0.165        0.8870          0.3700
4172       2   0.590     0.440   0.135        0.9660          0.4390
4173       2   0.600     0.475   0.205        1.1760          0.5255
4174       0   0.625     0.485   0.150        1.0945          0.5310
4175       2   0.710     0.555   0.195        1.9485          0.9455

      Viscera Weight  Shell Weight  Rings
4171          0.2390        0.2490     11
4172          0.2145        0.2605     10
4173          0.2875        0.3080      9
4174          0.2610        0.2960     10
4175          0.3765        0.4950     12
Checking data set
 Gender            0
Length             0
Diameter           0
Height             0
Whole Weight       0
Shucked Weight     0
Viscera Weight     0
```

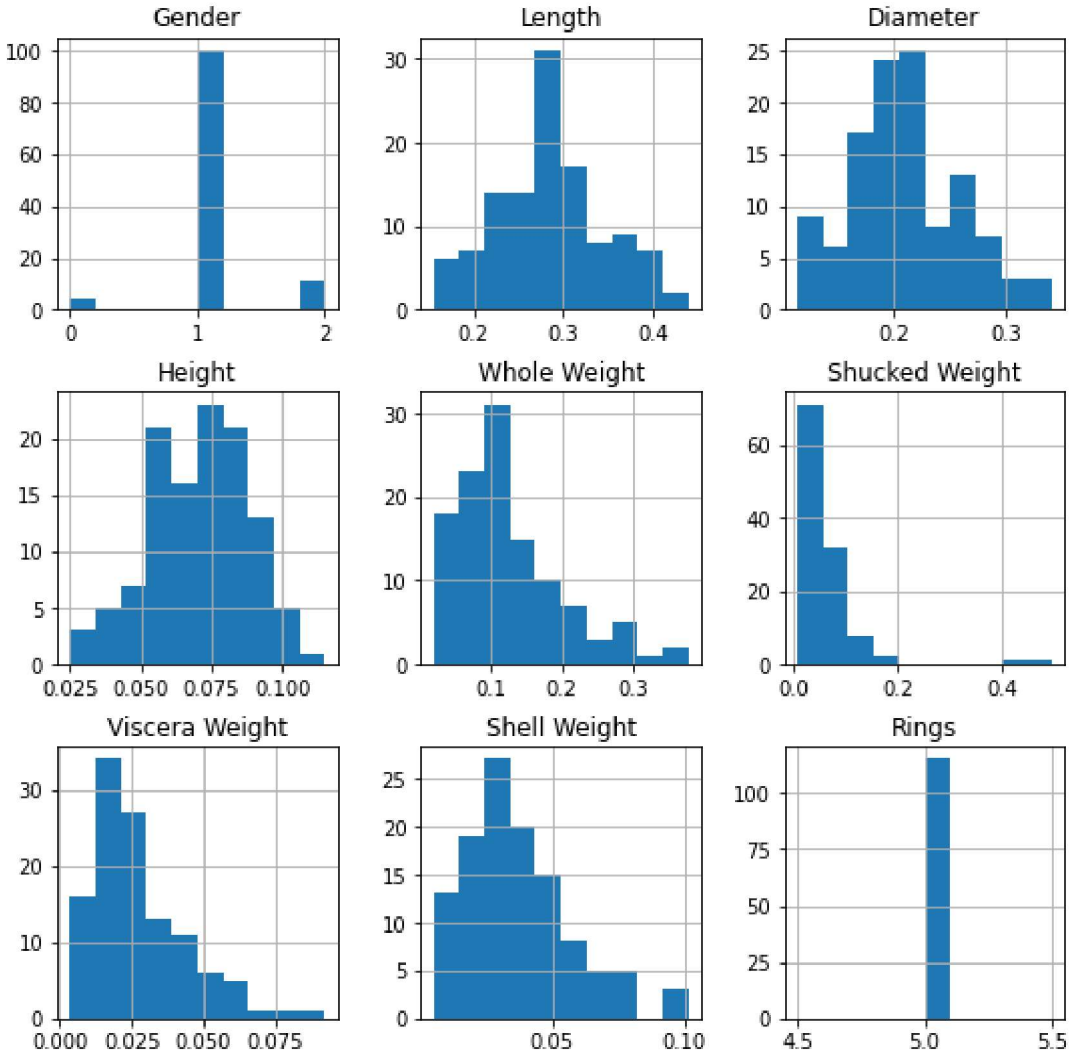
```
Shell Weight      0
Rings             0
dtype: int64
Describtion of the data
             Gender       Length     Diameter       Height  Whole Weight  \
count  4176.000000  4176.000000  4176.000000  4176.000000   4176.000000
mean      1.052682     0.524009     0.407892     0.139527      0.828818
std       0.822208     0.120103     0.099250     0.041826      0.490424
min       0.000000     0.075000     0.055000     0.000000      0.002000
25%       0.000000     0.450000     0.350000     0.115000      0.441500
50%       1.000000     0.545000     0.425000     0.140000      0.799750
75%       2.000000     0.615000     0.480000     0.165000      1.153250
max       2.000000     0.815000     0.650000     1.130000      2.825500

       Shucked Weight  Viscera Weight  Shell Weight        Rings
count      4176.00000     4176.000000   4176.000000  4176.000000
mean          0.35940        0.180613      0.238852     9.932471
std           0.22198        0.109620      0.139213     3.223601
min           0.00100        0.000500      0.001500     1.000000
25%           0.18600        0.093375      0.130000     8.000000
50%           0.33600        0.171000      0.234000     9.000000
75%           0.50200        0.253000      0.329000    11.000000
max           1.48800        0.760000      1.005000    29.000000
column names
 Index(['Gender', 'Length', 'Diameter', 'Height', 'Whole Weight',
       'Shucked Weight', 'Viscera Weight', 'Shell Weight', 'Rings'],
      dtype='object')
The shape of the data
 (4176, 9)
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\pandas\plott
ing\_matplotlib\tools.py:218: RuntimeWarning: More than 20 figures have been opened.
Figures created through the pyplot interface (`matplotlib.pyplot.figure`) are retain
ed until explicitly closed and may consume too much memory. (To control this warnin
g, see the rcParam `figure.max_open_warning`).
  fig = plt.figure(**fig_kw)
```

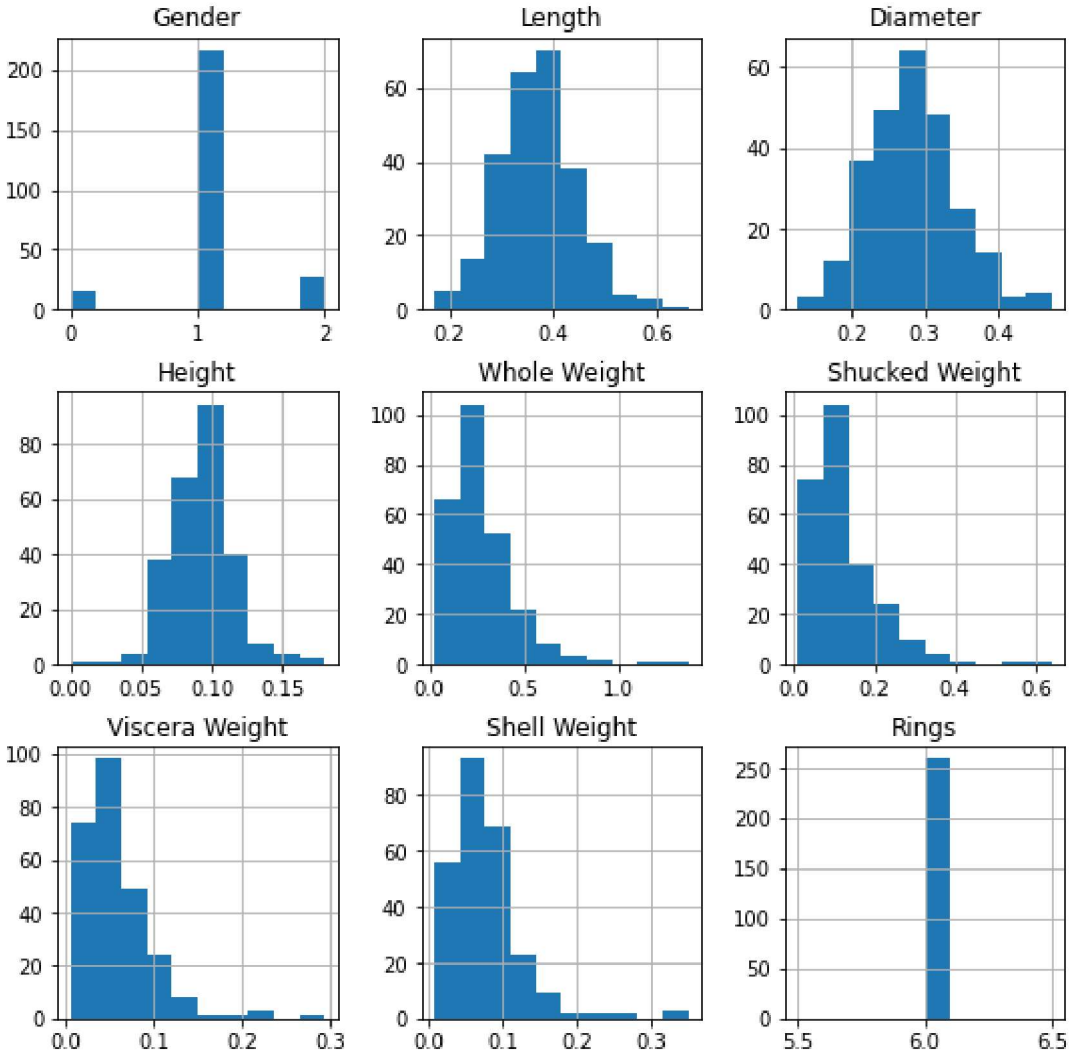

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

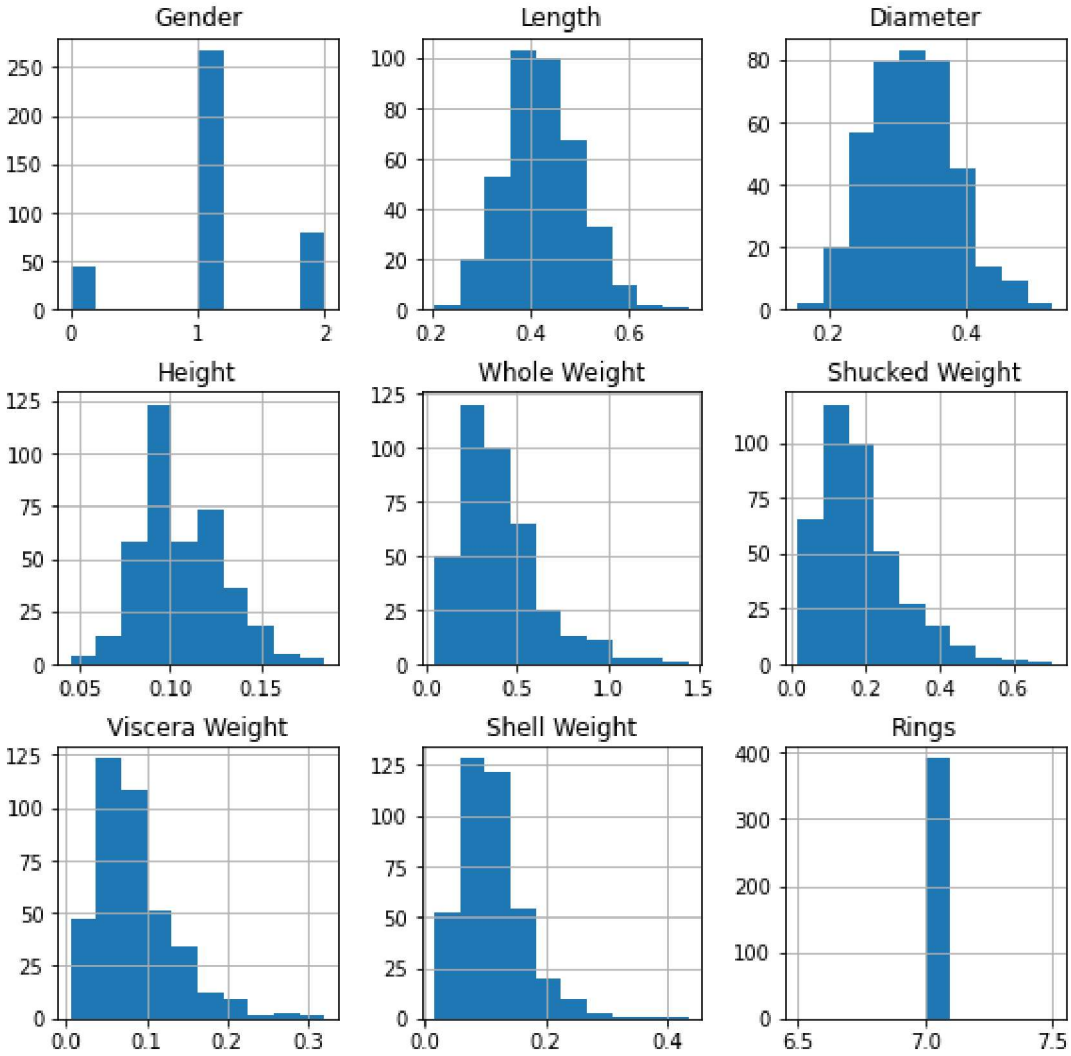

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

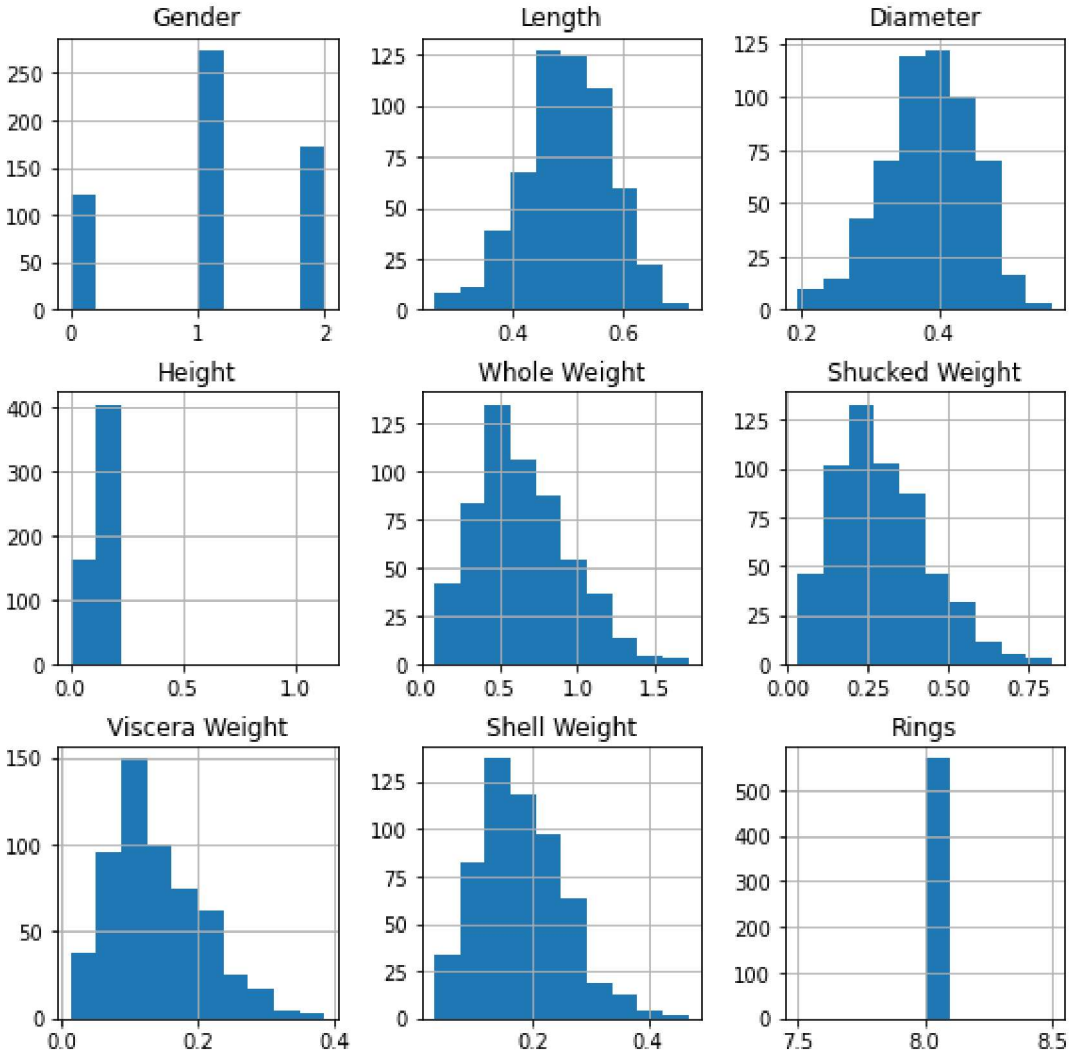

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

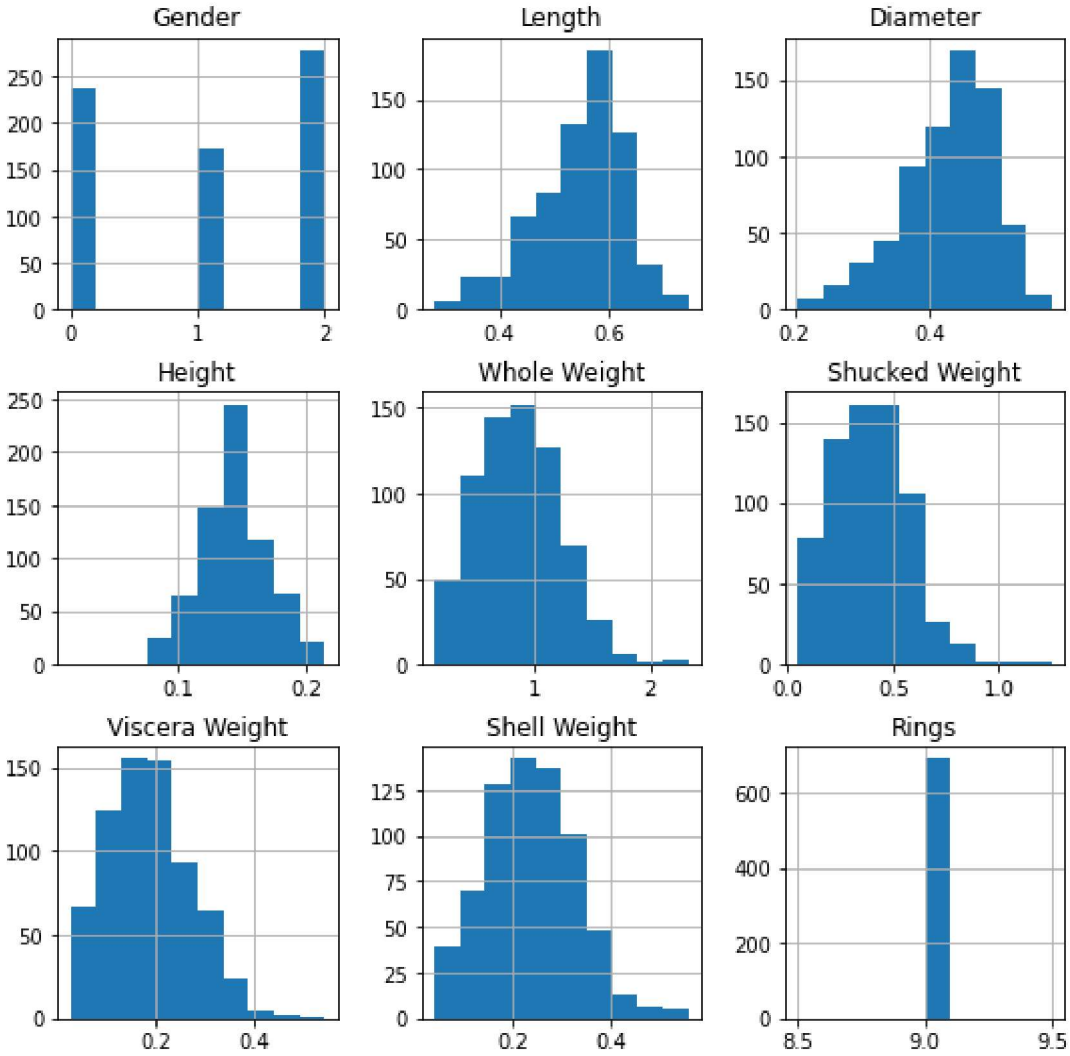

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

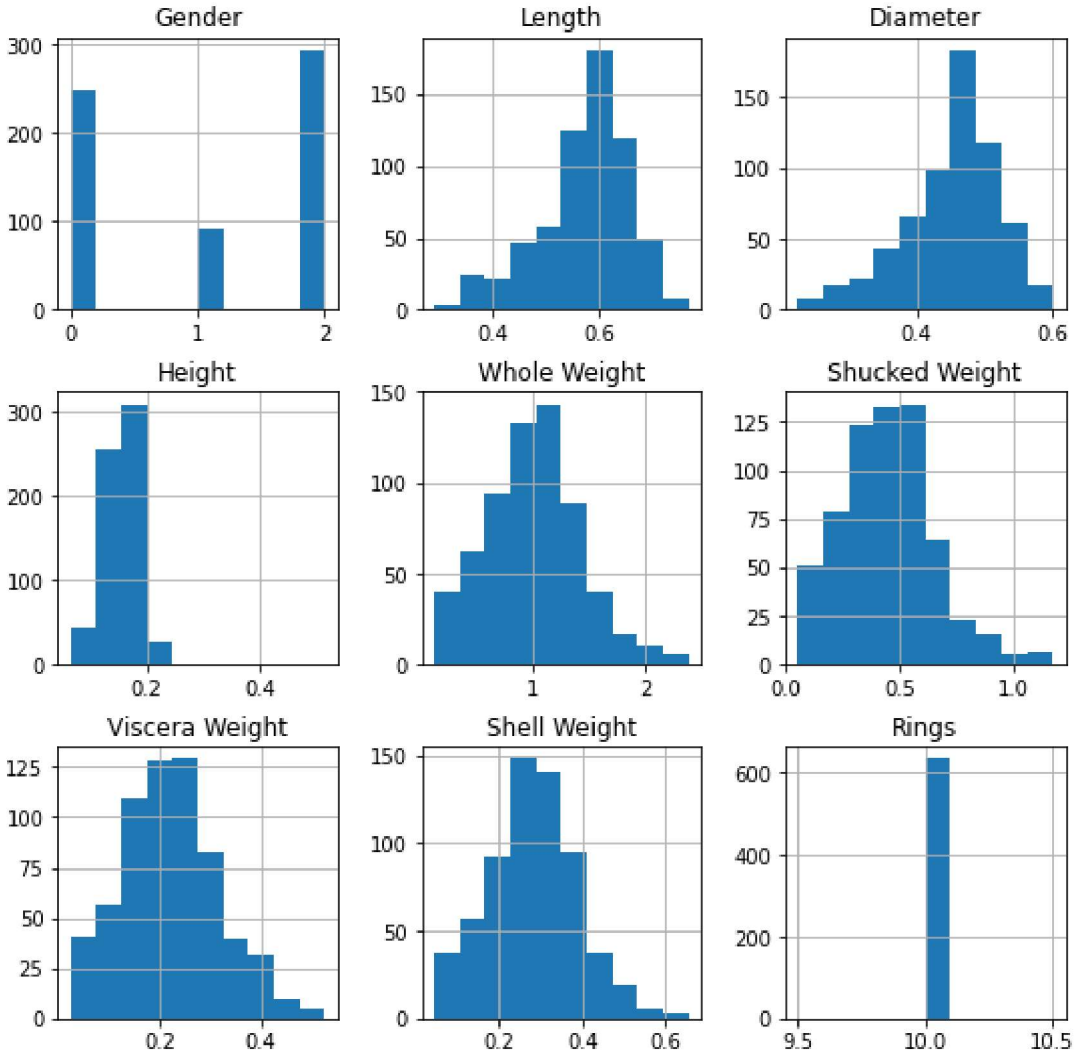

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

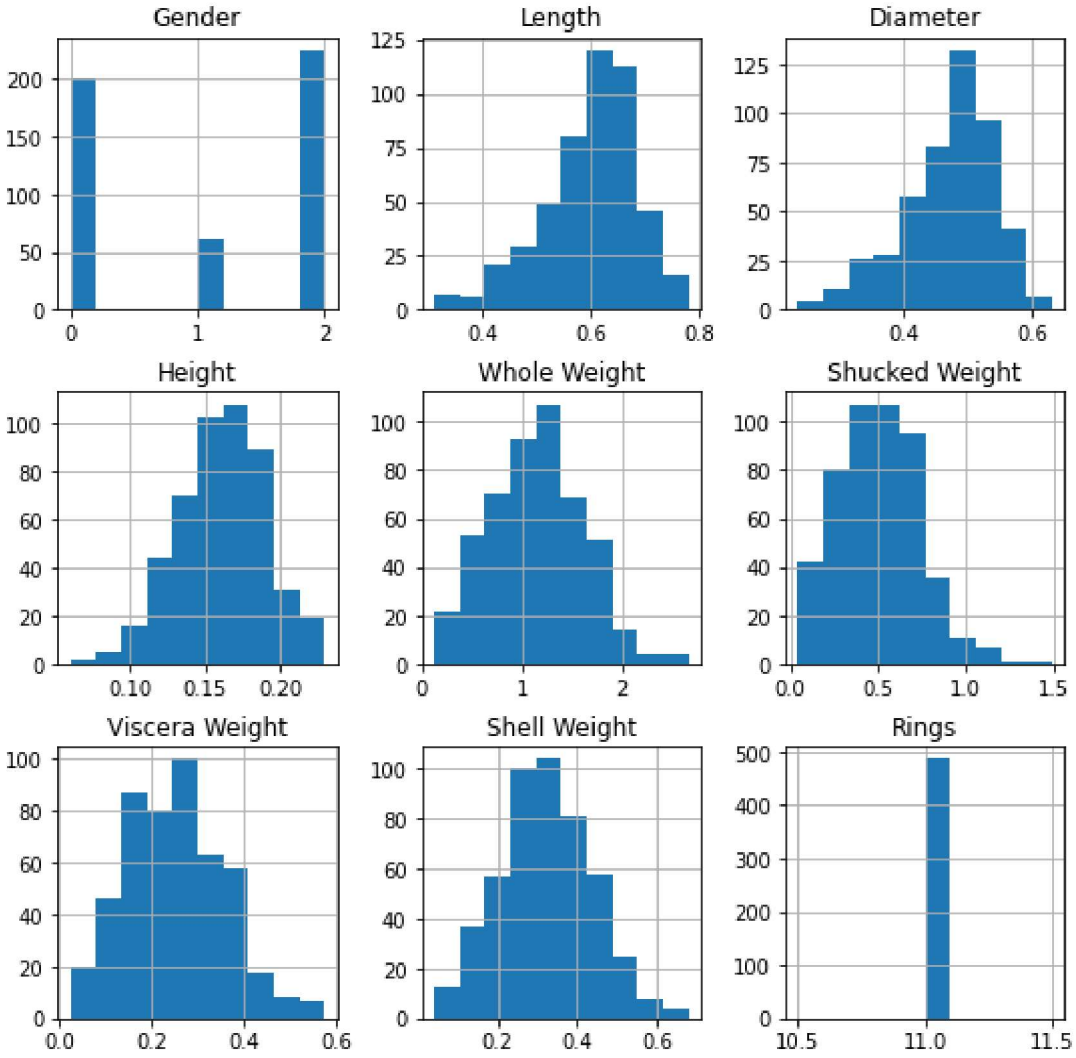

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

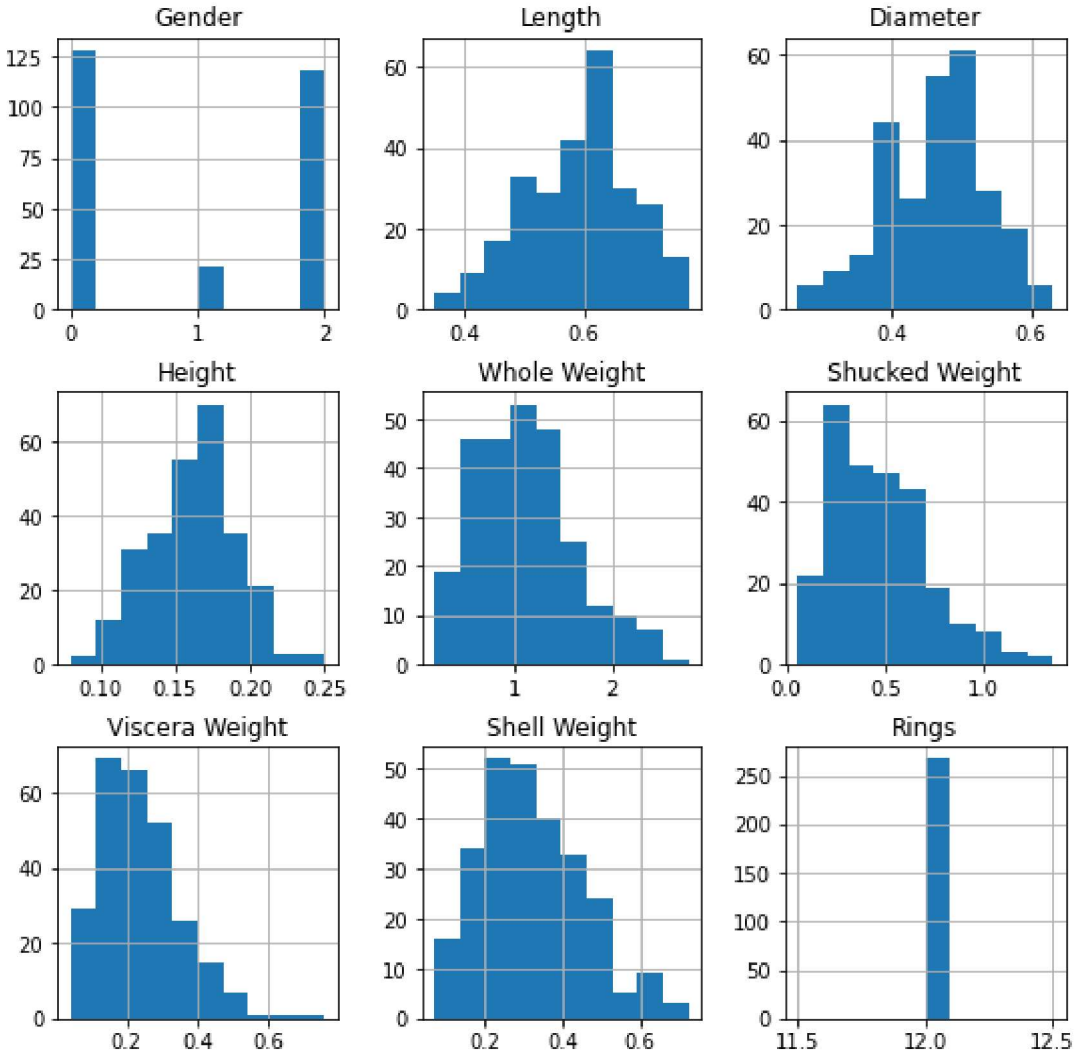

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

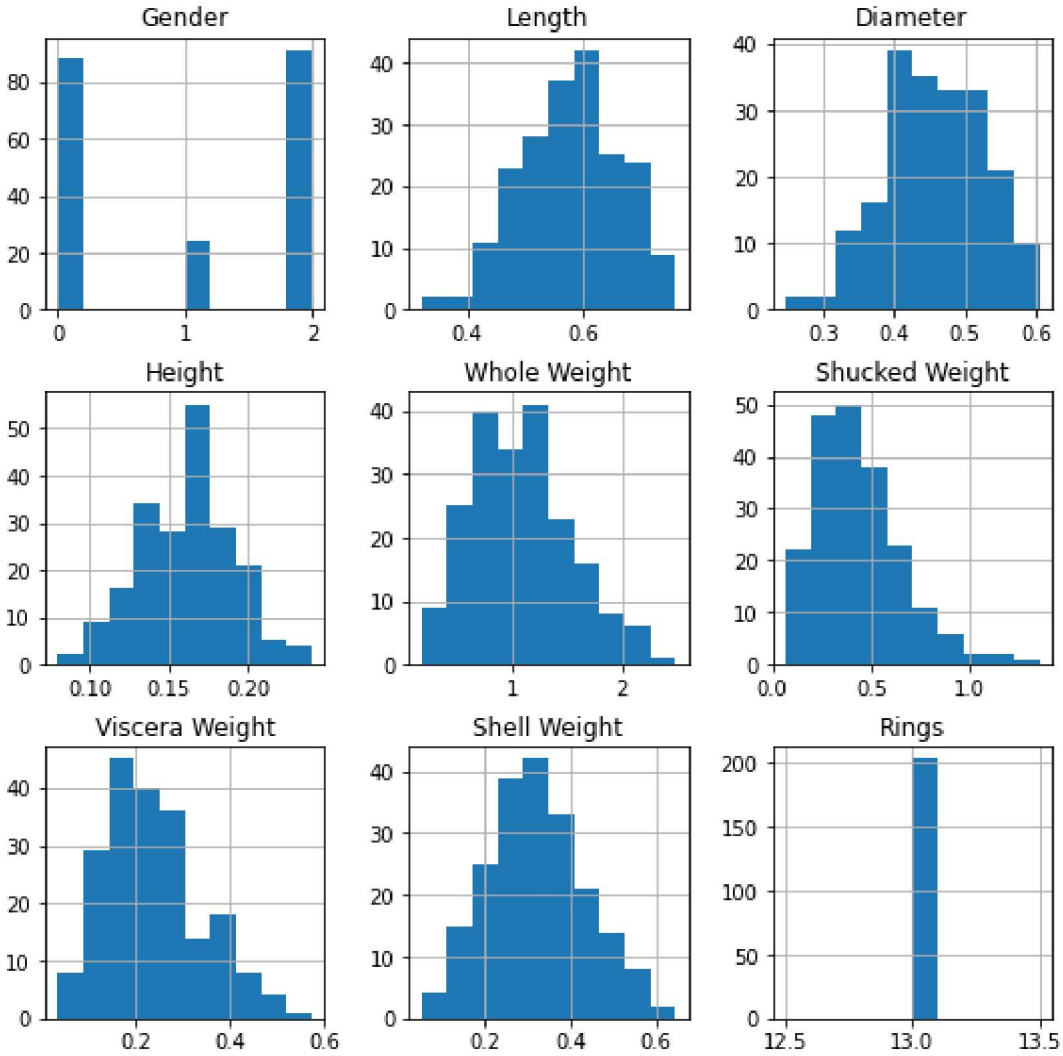

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

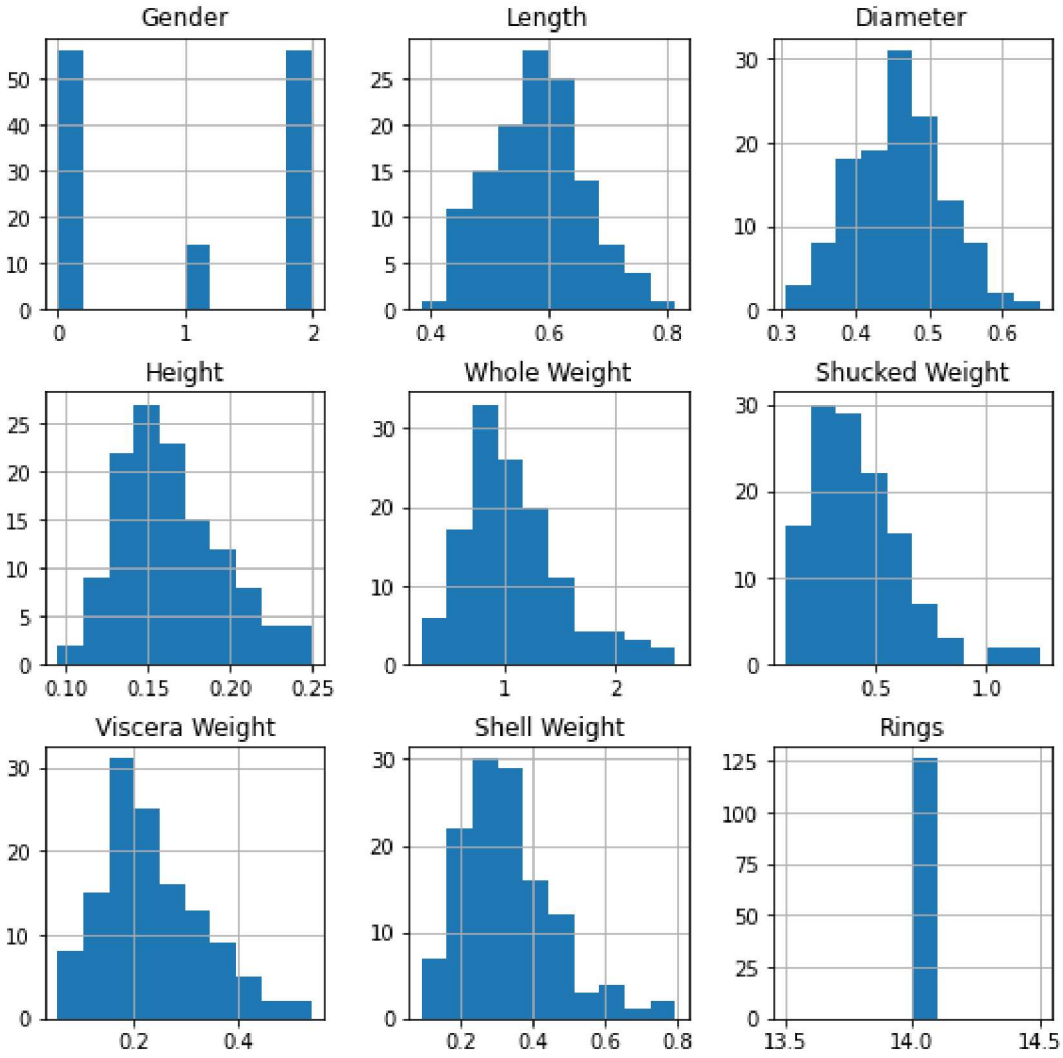

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

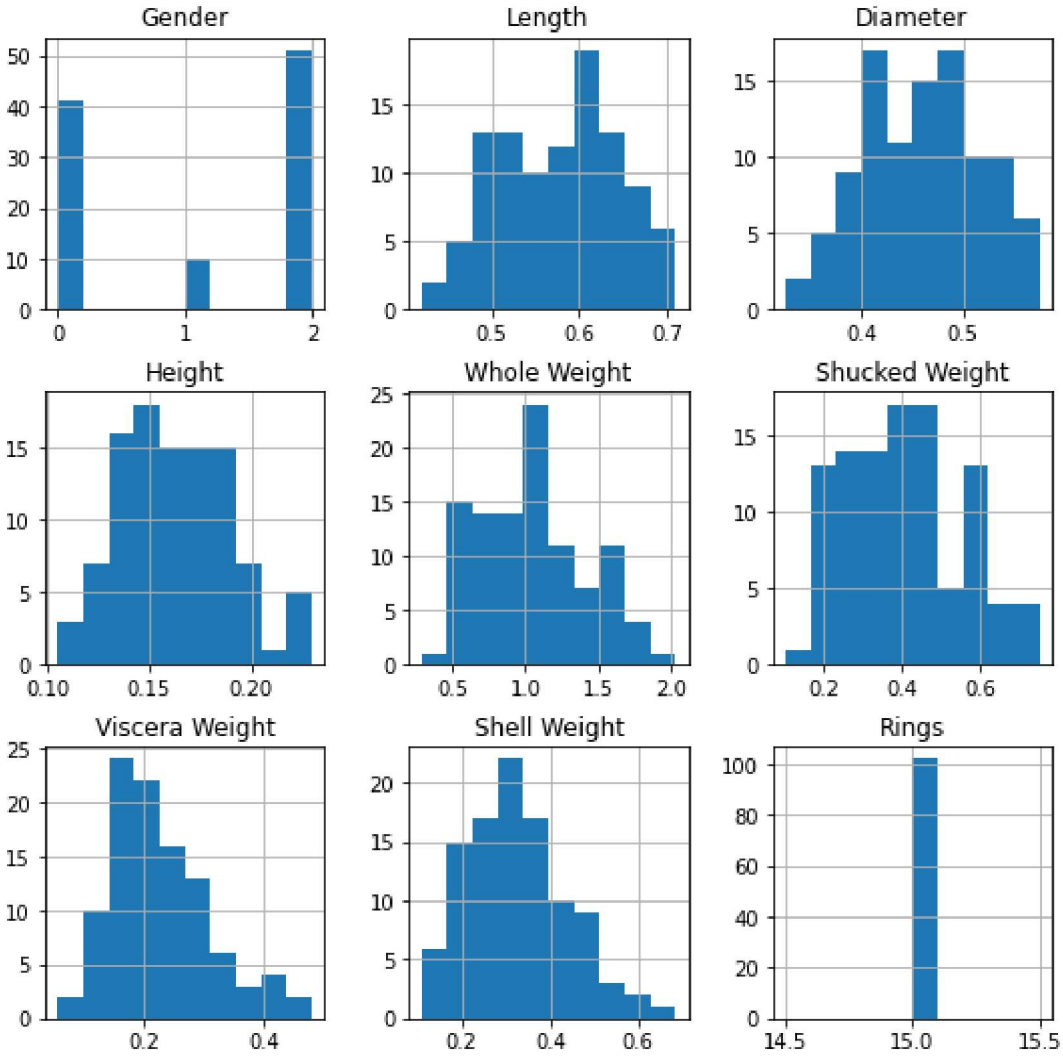

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

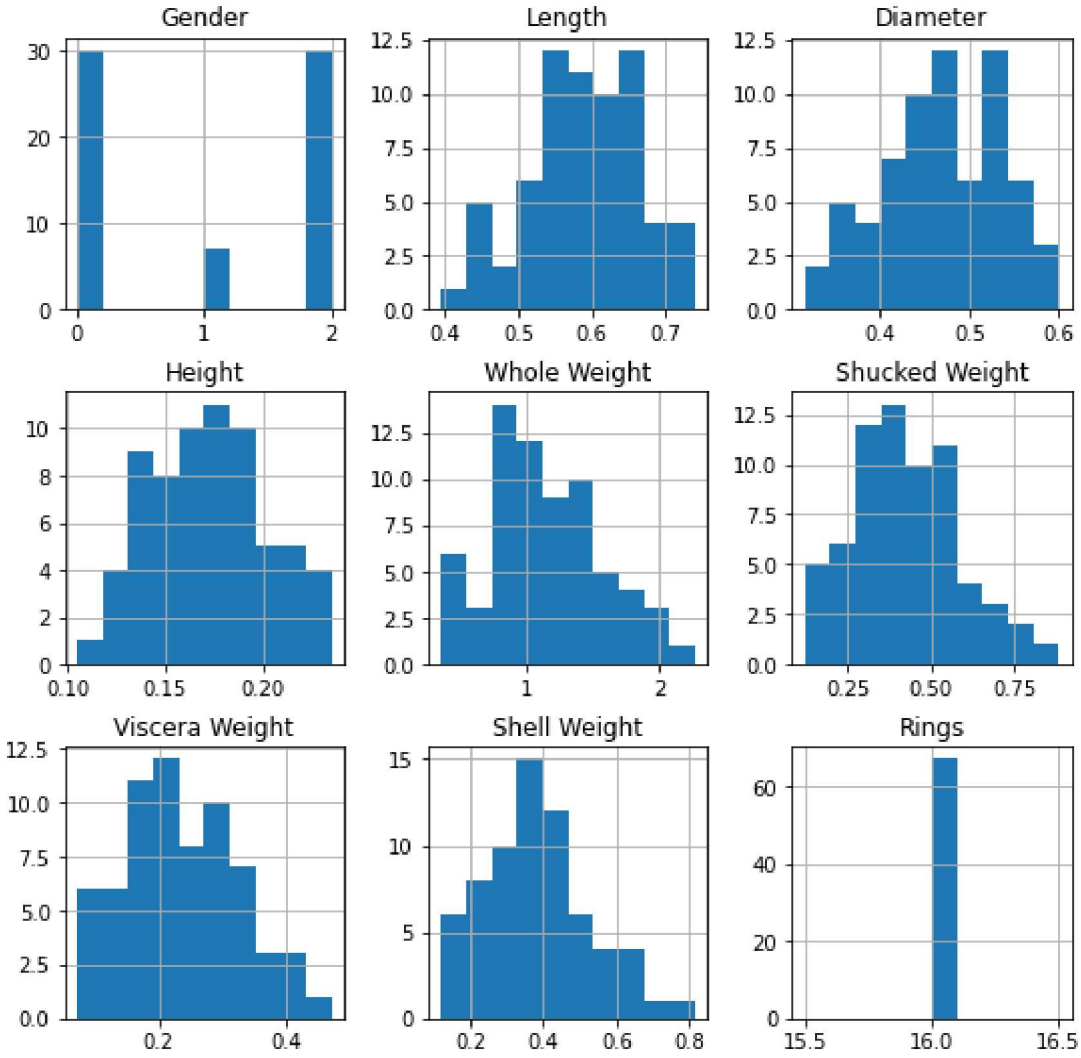

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

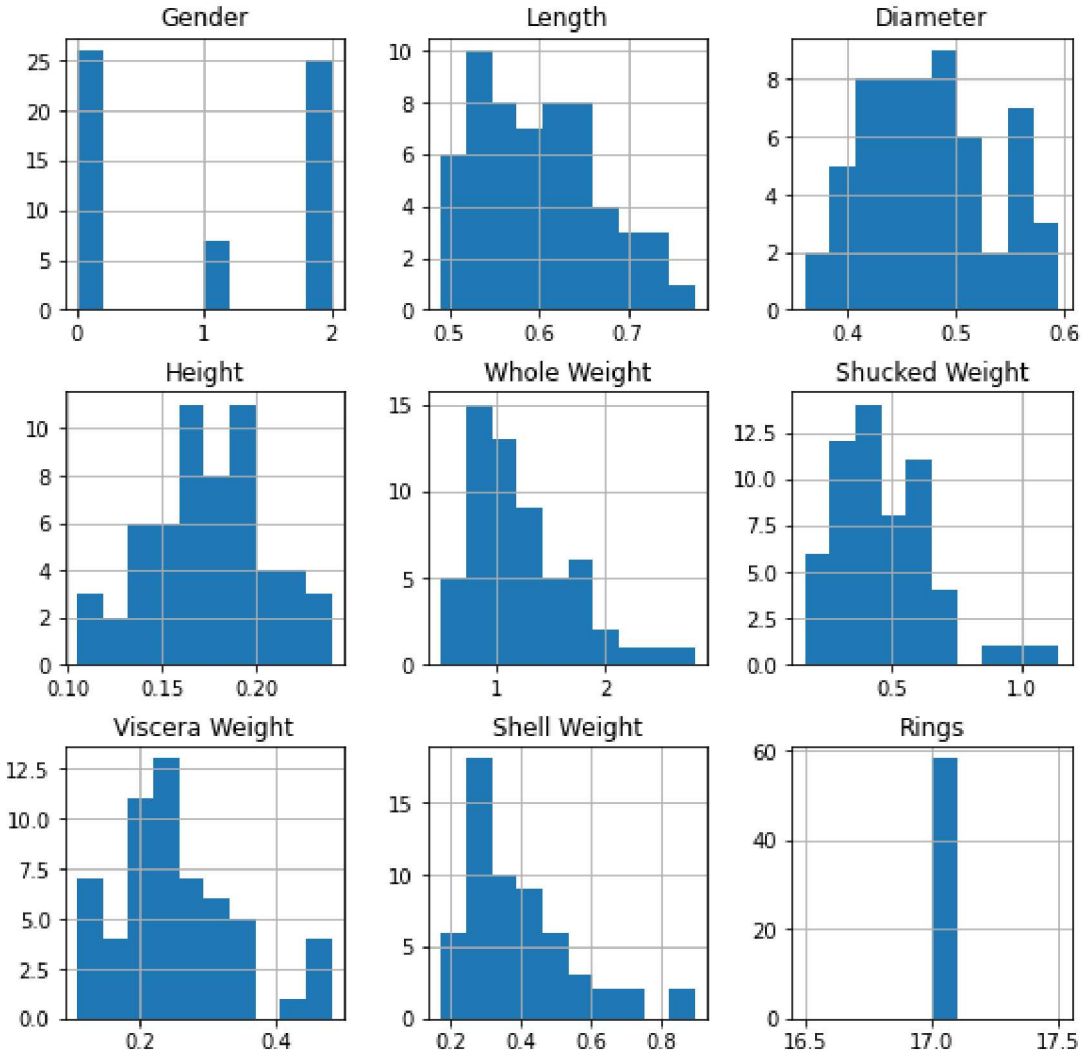

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

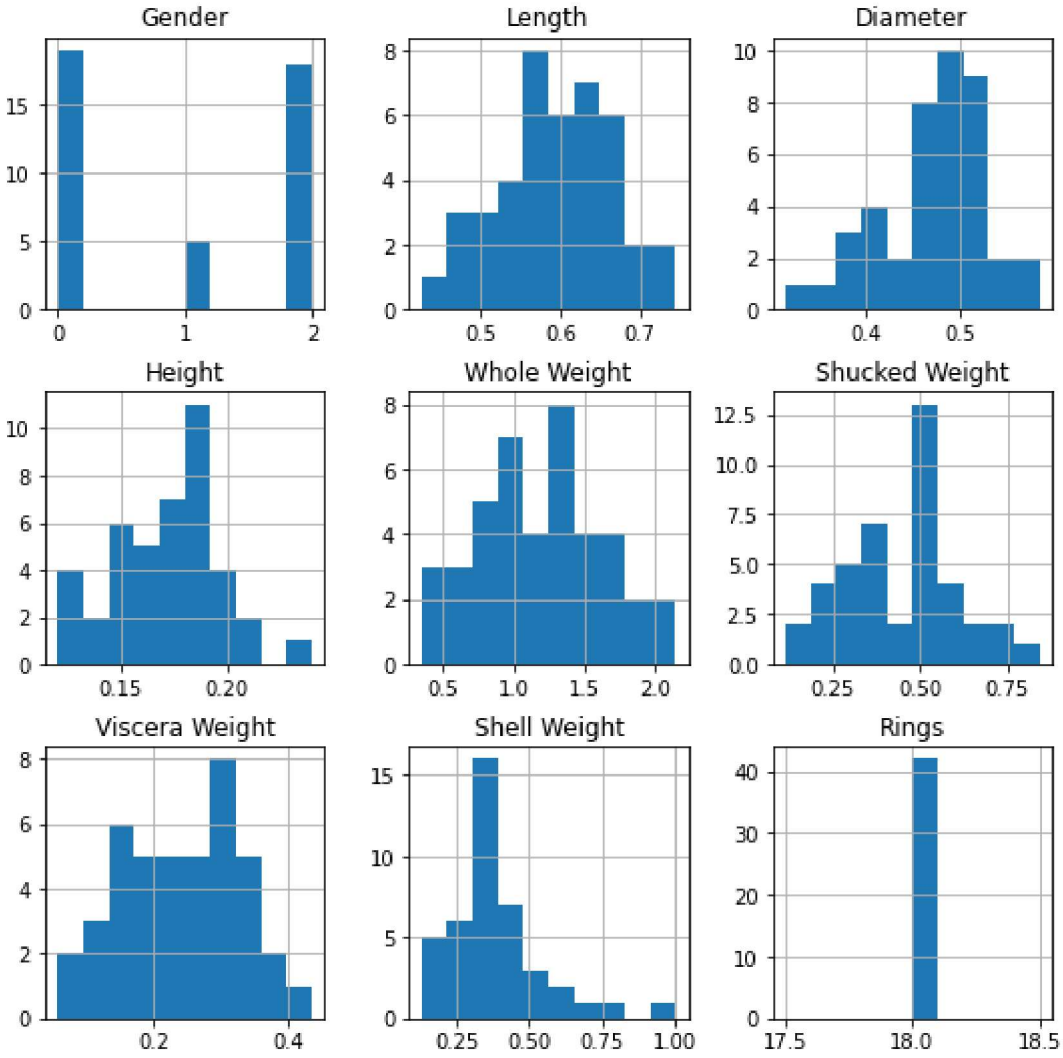

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

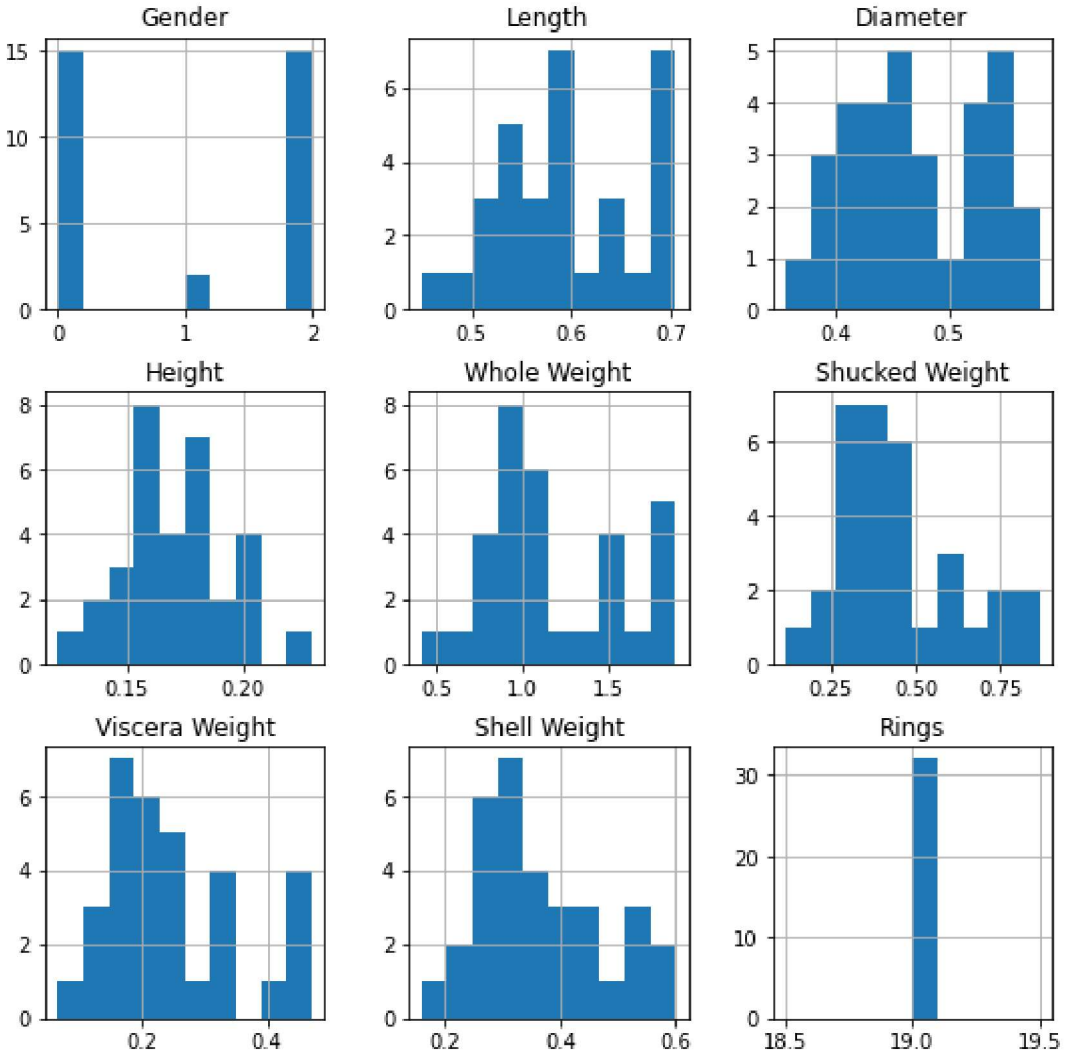

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

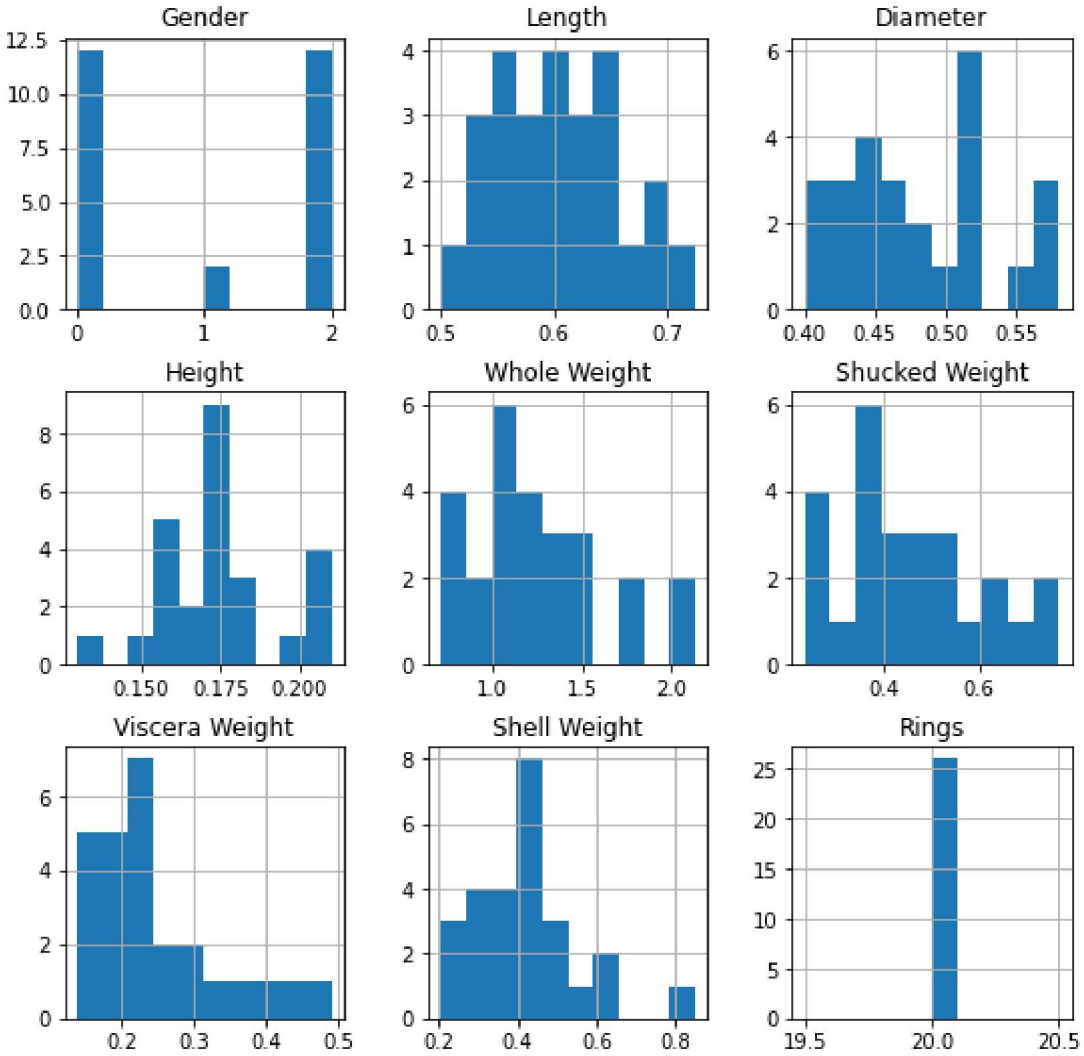

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

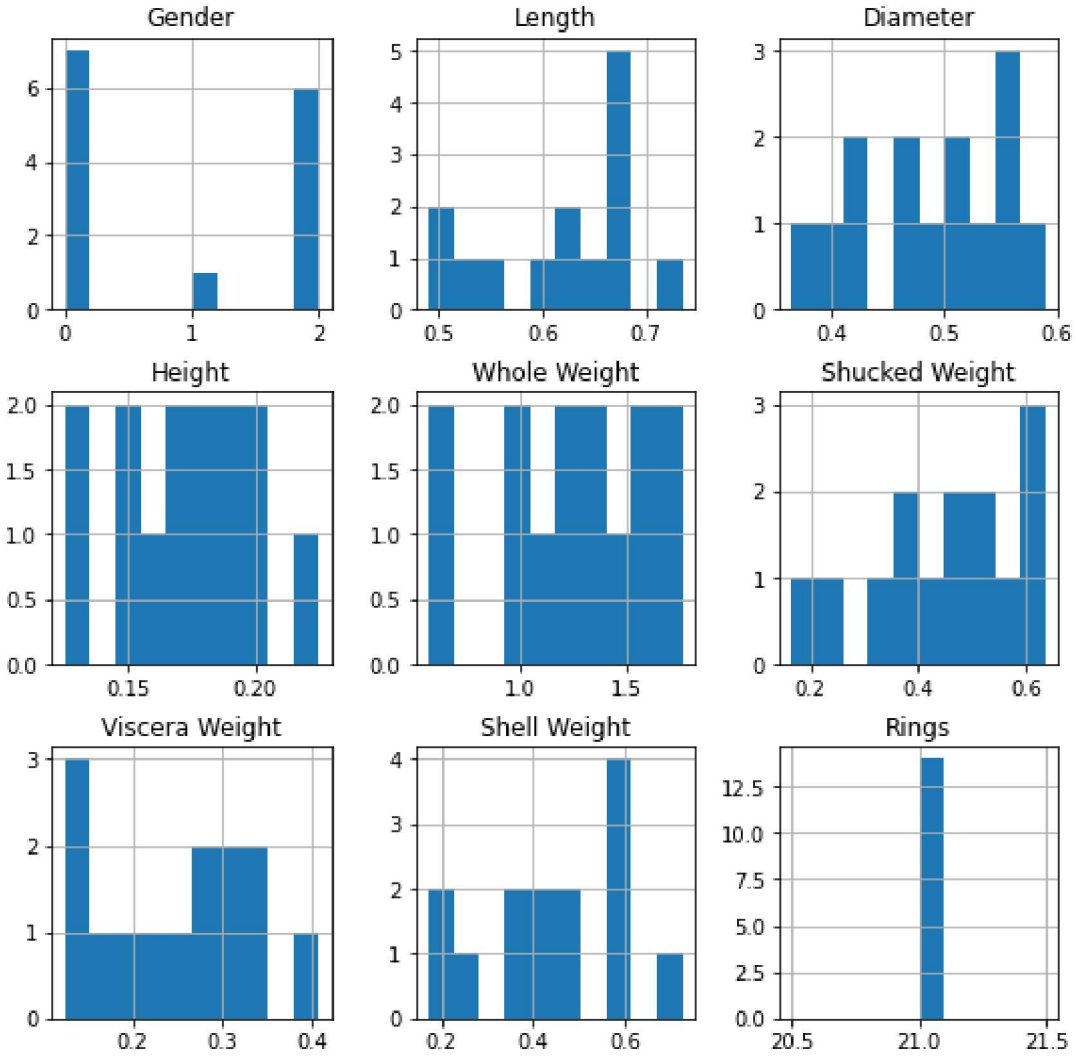

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

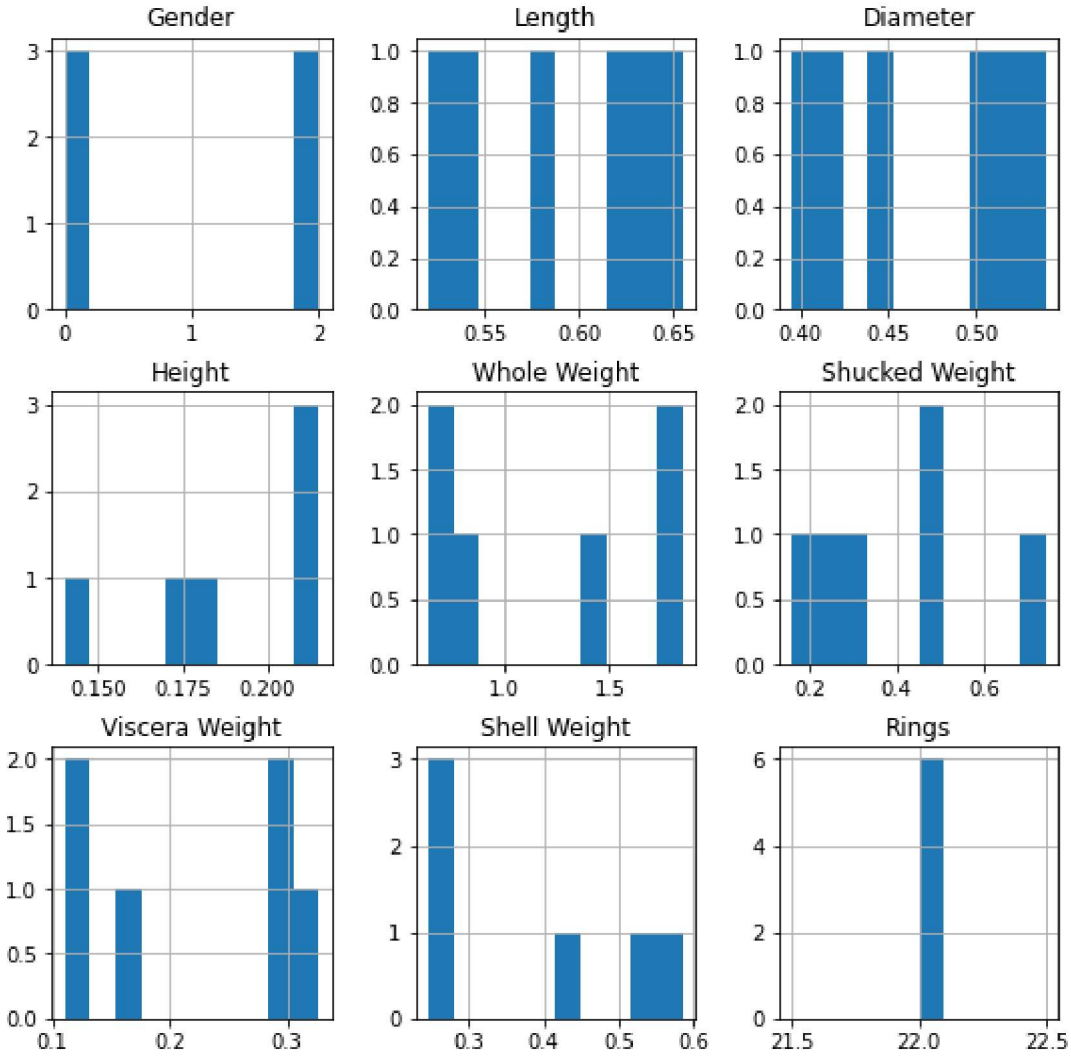

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

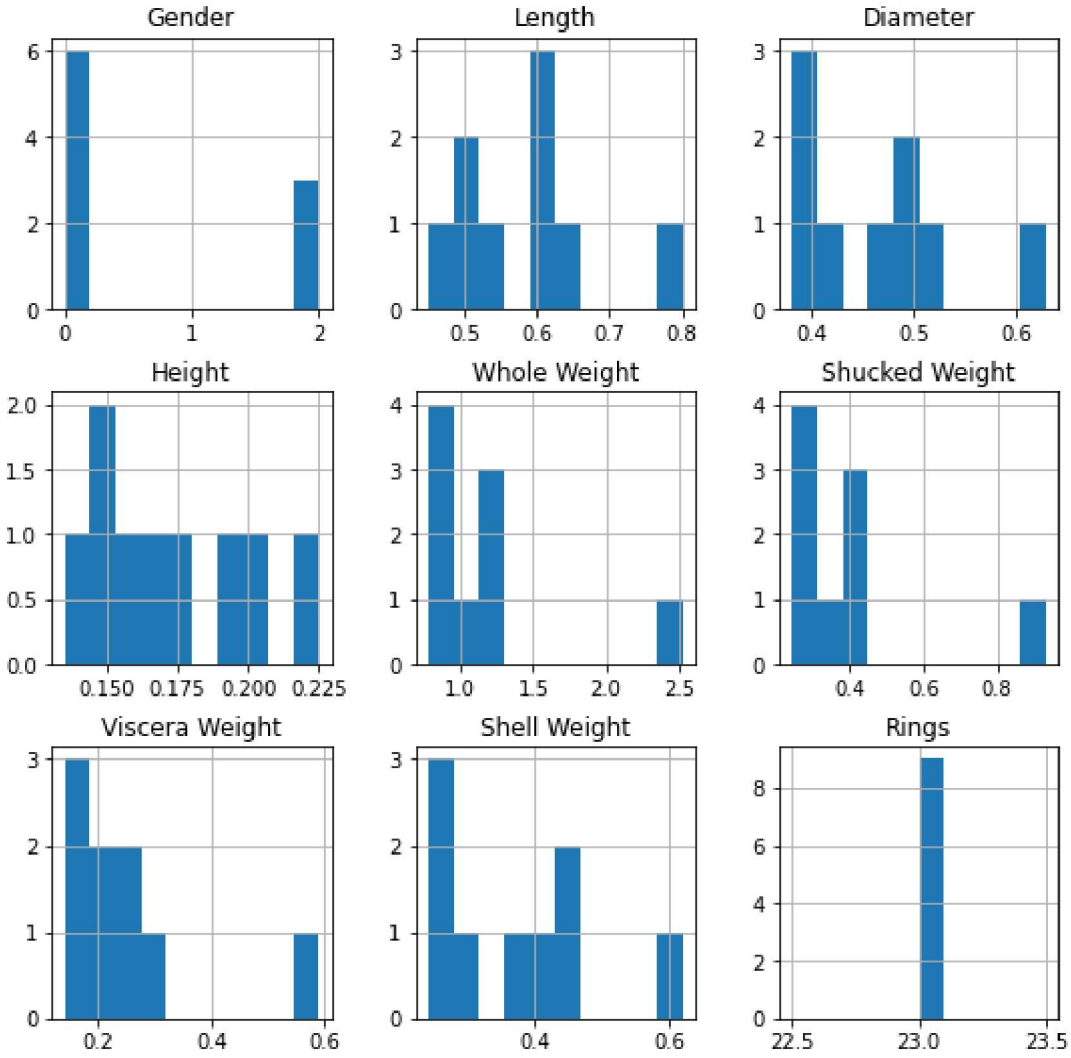

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

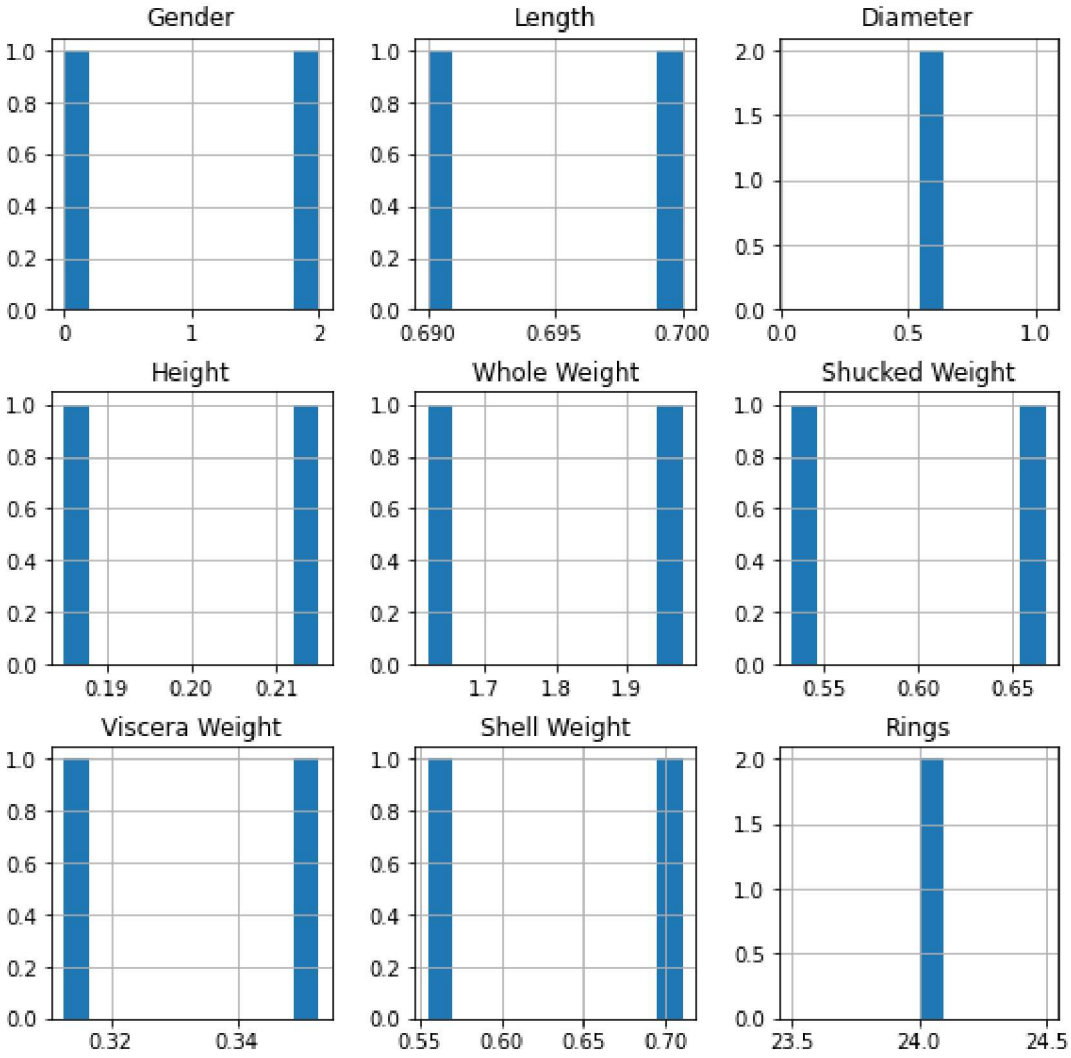

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

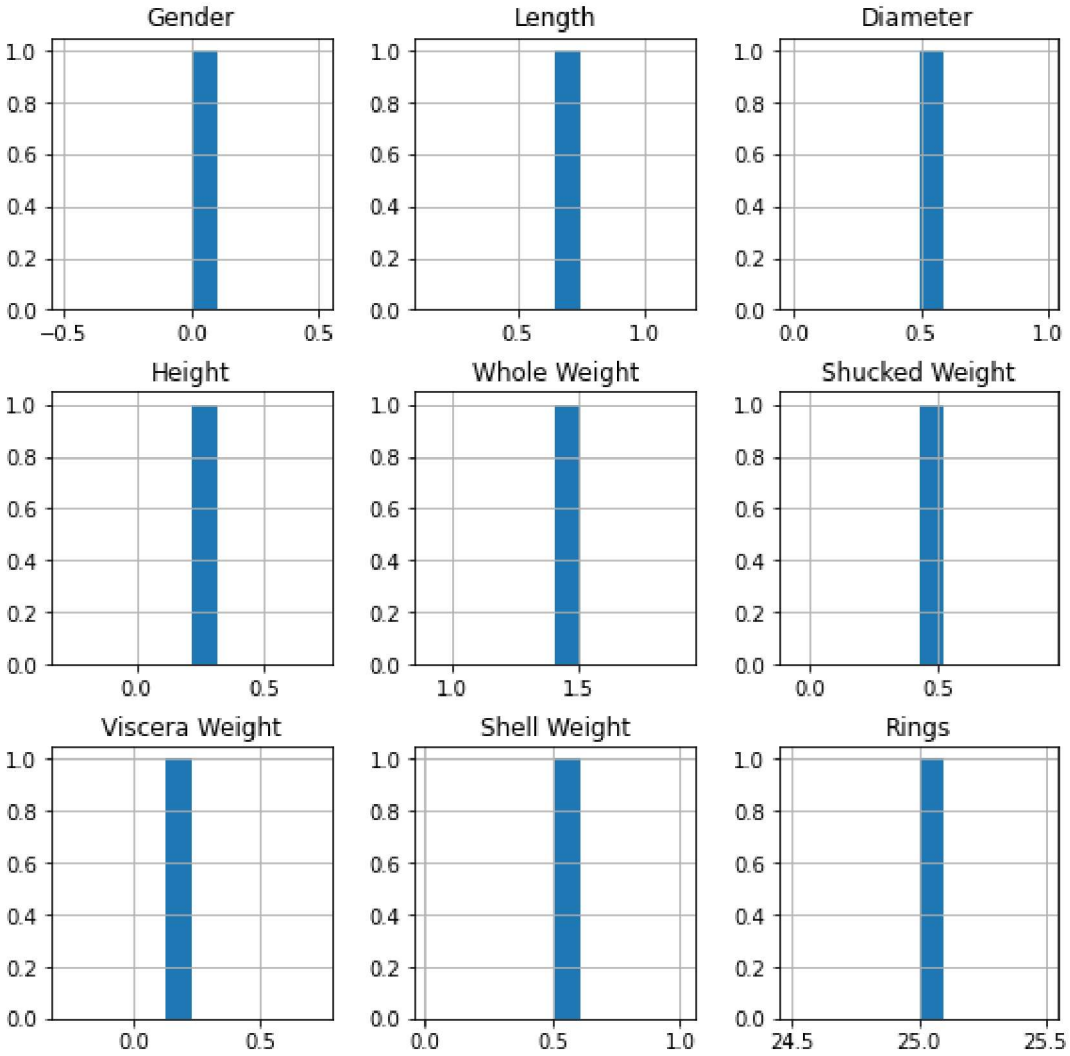

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

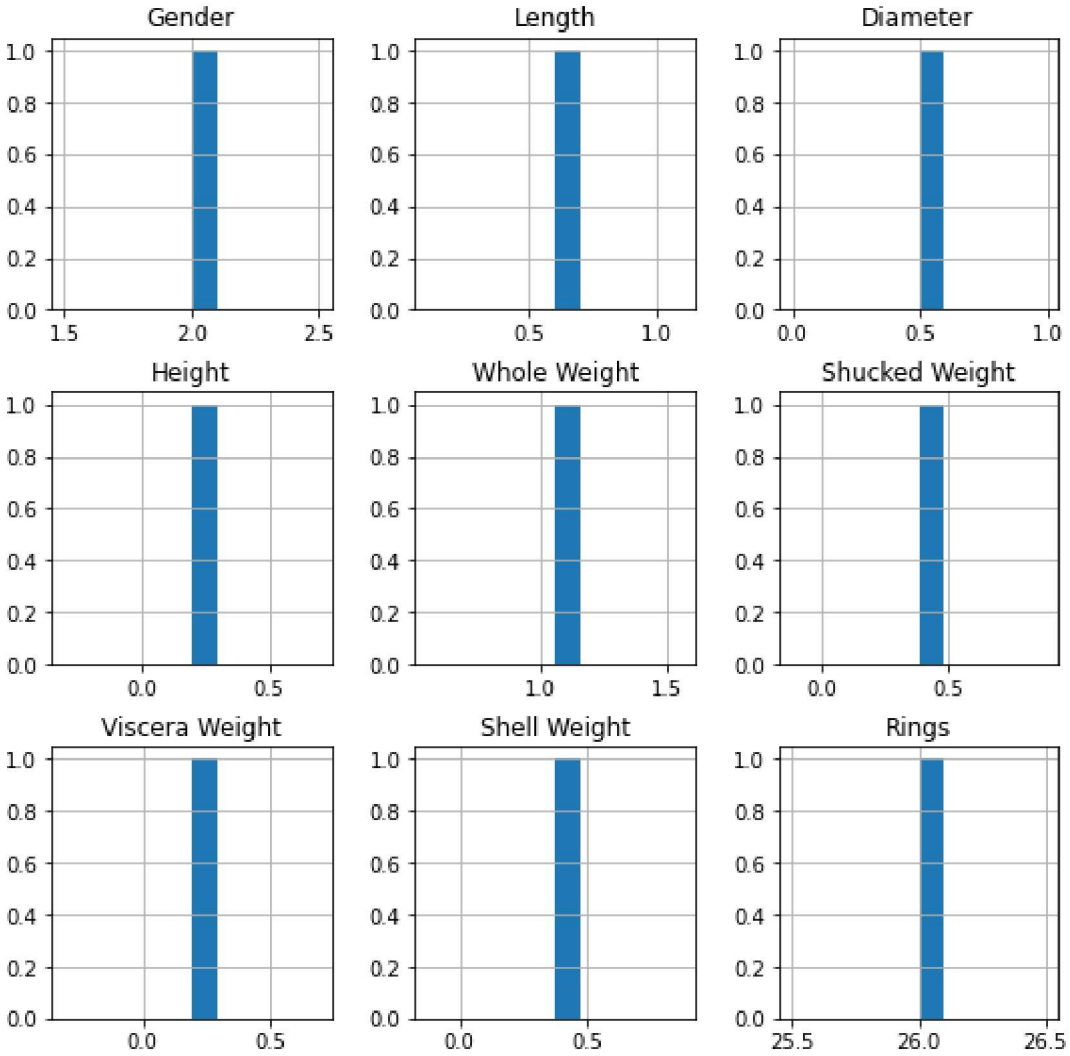

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

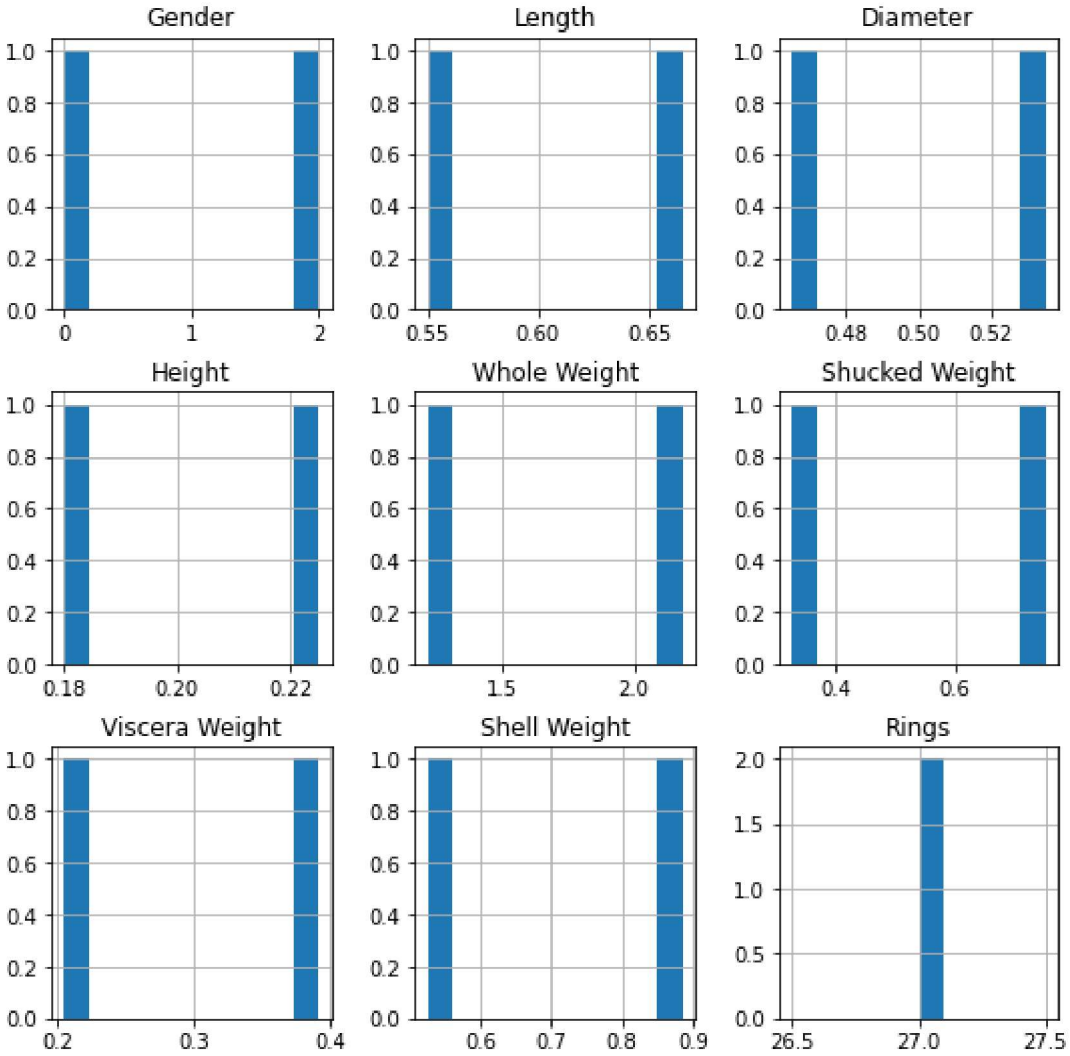

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

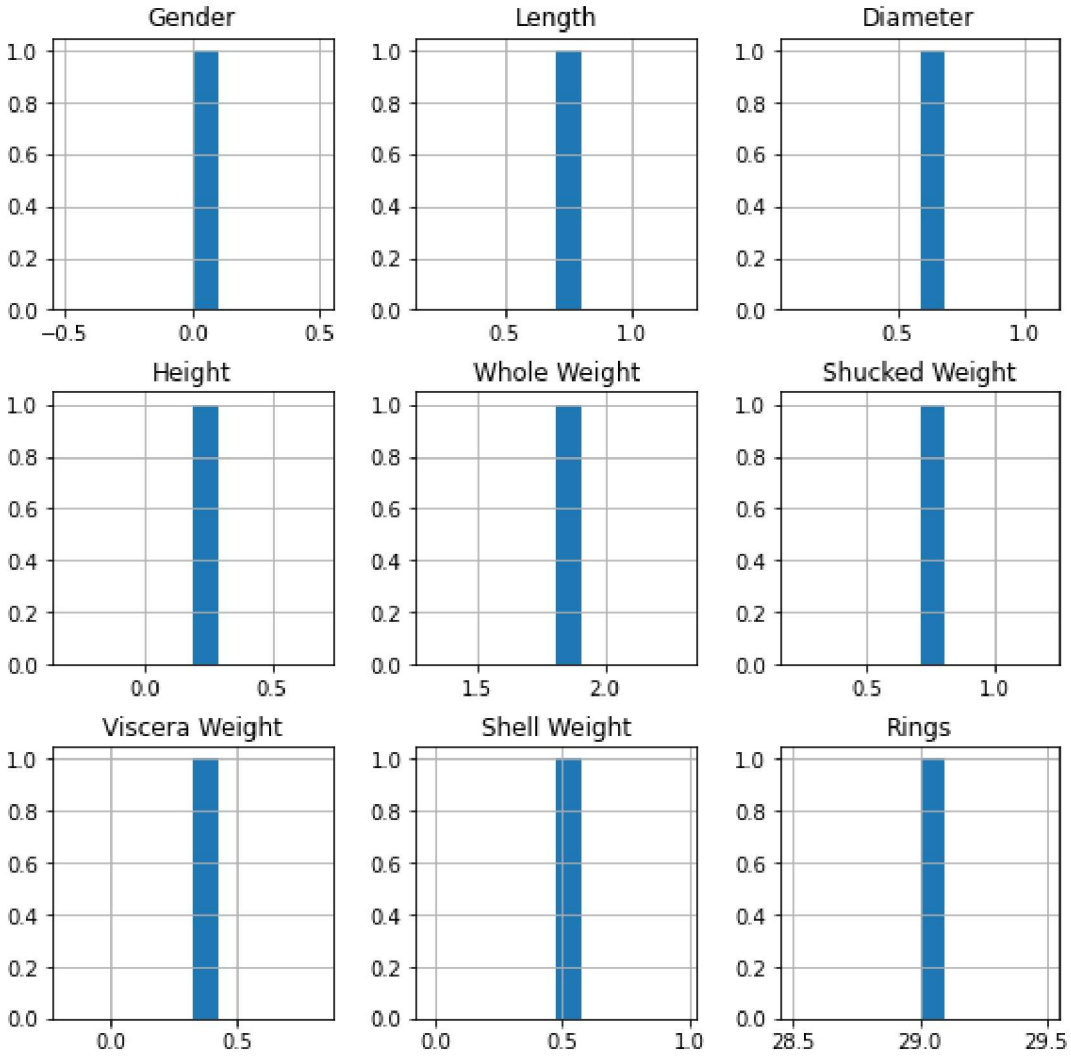

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings


Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings


Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

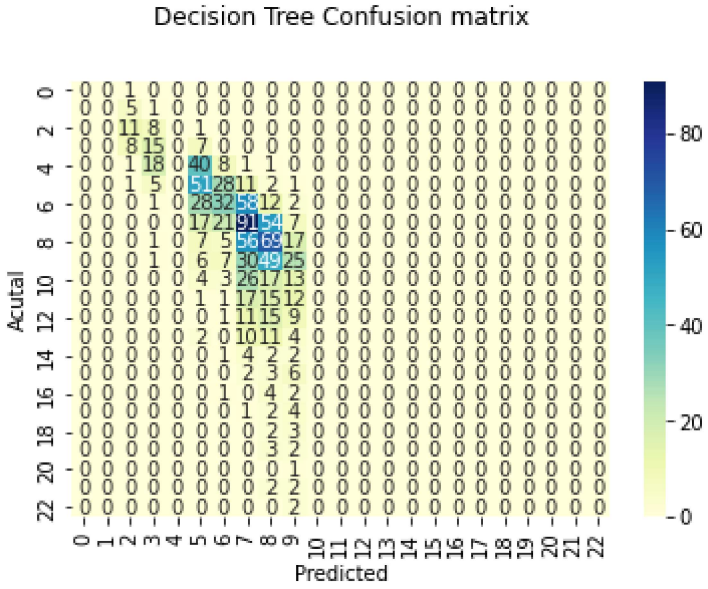

Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings


Gender
Length
Diameter
Height
Whole Weight
Shucked Weight
Viscera Weight
Shell Weight
Rings

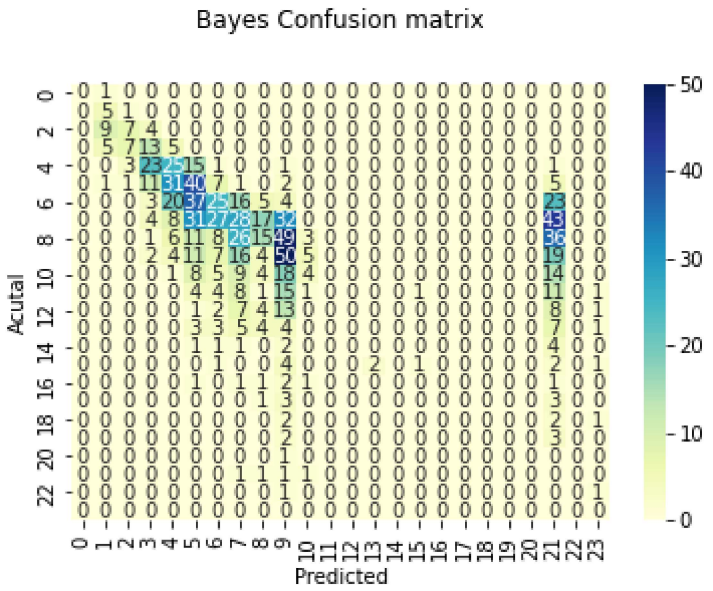



```
Decision Tree Accuracy Score is:  0.28160919540229884
Decision Tree F1 Score is:  0.1043654987201067
Decision Tree Recall Score is:  0.12790113410583515
Decision Tree Precision Score is:  0.0905434128493515
Decision Tree Confusion Score is:  [[ 0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0  0  0  0]
 [ 0  0  5  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0 11  8  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  8 15  0  7  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  1 18  0 40  8  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  1  5  0 51 28 11  2  1  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  1  0 28 32 58 12  2  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0 17 21 91 54  7  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  1  0  7  5 56 69 17  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  1  0  6  7 30 49 25  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  4  3 26 17 13  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  1  1 17 15 12  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  1 11 15  9  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  2  0 10 11  4  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  1  4  2  2  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  2  3  6  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  1  0  4  2  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  1  2  4  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  2  3  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  3  2  0  0  0  0  0  0  0  0  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]
 [ 0  0  0  0  0  0  0  0  2  2  0  0  0  0  0  0  0  0  0  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]]
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metr
ics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and be
```

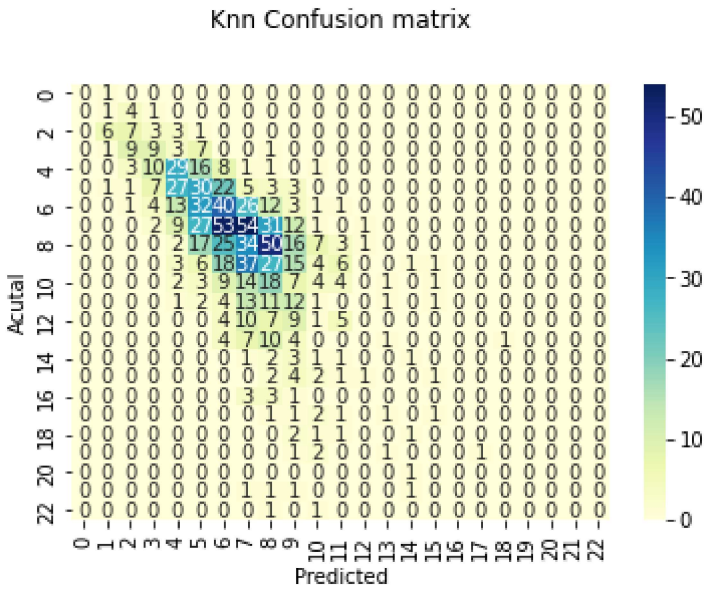
```
ing set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```



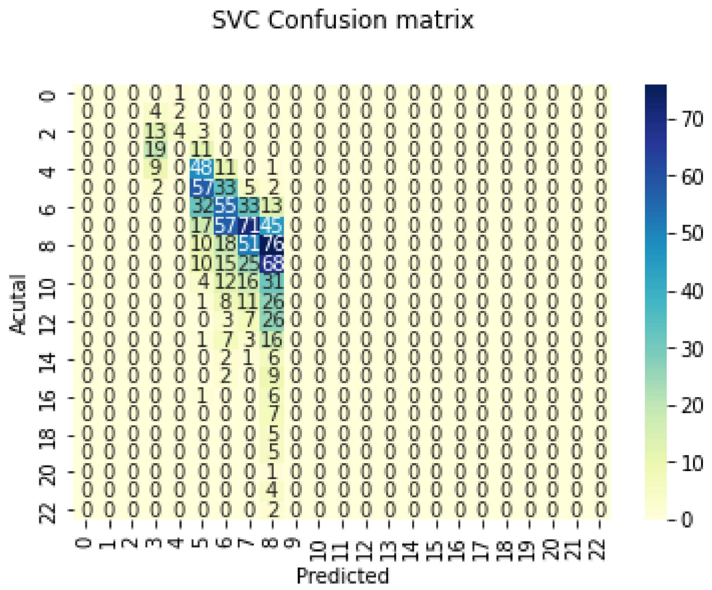

```
Bayes Accuracy Score is:  0.20402298850574713
Bayes Accuracy Score is:  0.20402298850574713
Bayes F1 Score is:  0.11363609331388785
Bayes Recall Score is:  0.14138618411084608
Bayes Precision Score is:  0.12906634242537904
Bayes Confusion Score is:  [[ 0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0  0  0]
 [ 0  5  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  9  7  4  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  5  7 13  5  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  3 23 25 15  1  0  0  1  0  0  0  0  0  0  0  0  0  0  0  1  0  0]
 [ 0  1  1 11 31 40  7  1  0  2  0  0  0  0  0  0  0  0  0  0  0  5  0  0]
 [ 0  0  0  3 20 37 25 16  5  4  0  0  0  0  0  0  0  0  0  0  0 23  0  0]
 [ 0  0  0  4  8 31 27 28 17 32  0  0  0  0  0  0  0  0  0  0  0 43  0  0]
 [ 0  0  0  1  6 11  8 26 15 49  3  0  0  0  0  0  0  0  0  0  0 36  0  0]
 [ 0  0  0  2  4 11  7 16  4 50  5  0  0  0  0  0  0  0  0  0  0 19  0  0]
 [ 0  0  0  0  1  8  5  9  4 18  4  0  0  0  0  0  0  0  0  0  0 14  0  0]
 [ 0  0  0  0  0  4  4  8  1 15  1  0  0  0  0  1  0  0  0  0  0 11  0  1]
 [ 0  0  0  0  0  1  2  7  4 13  0  0  0  0  0  0  0  0  0  0  0  8  0  1]
 [ 0  0  0  0  0  3  3  5  4  4  0  0  0  0  0  0  0  0  0  0  0  7  0  1]
 [ 0  0  0  0  0  1  1  1  0  2  0  0  0  0  0  0  0  0  0  0  0  4  0  0]
 [ 0  0  0  0  0  0  1  0  0  4  0  0  0  2  0  1  0  0  0  0  0  2  0  1]
 [ 0  0  0  0  0  1  0  1  1  2  1  0  0  0  0  0  0  0  0  0  0  1  0  0]
 [ 0  0  0  0  0  0  0  0  1  3  0  0  0  0  0  0  0  0  0  0  0  3  0  0]
 [ 0  0  0  0  0  0  0  0  0  2  0  0  0  0  0  0  0  0  0  0  0  2  0  1]
 [ 0  0  0  0  0  0  0  0  0  2  0  0  0  0  0  0  0  0  0  0  0  3  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  0]
 [ 0  0  0  0  0  0  0  1  1  1  1  0  0  0  0  0  0  0  0  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  1]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]]
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metr
ics\_classification.py:1308: UndefinedMetricWarning: Recall is ill-defined and being
set to 0.0 in labels with no true samples. Use `zero_division` parameter to control
this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metr
ics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and be
ing set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```




```
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\bas
e.py:441: UserWarning: X does not have valid feature names, but KNeighborsClassifier
was fitted with feature names
  warnings.warn(
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metr
ics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and be
ing set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
Knn Accuracy Score is:  0.23180076628352492
Knn Accuracy Score is:  0.23180076628352492
Knn F1 Score is:  0.11731548912818308
Knn Recall Score is:  0.12509555750273305
Knn Precision Score is:  0.12207702656031645
Knn Confusion Score is:  [[ 0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0]
 [ 0  1  4  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  6  7  3  3  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  1  9  9  3  7  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  3 10 29 16  8  1  1  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  1  1  7 27 30 22  5  3  3  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  1  4 13 32 40 26 12  3  1  1  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  2  9 27 53 54 31 12  1  0  1  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  2 17 25 34 50 16  7  3  1  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  3  6 18 37 27 15  4  6  0  0  1  1  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  2  3  9 14 18  7  4  4  0  1  0  1  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  1  2  4 13 11 12  1  0  0  1  0  1  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  4 10  7  9  1  5  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  4  7 10  4  0  0  0  1  0  0  0  0  1  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  1  2  3  1  1  0  0  1  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  2  4  2  1  1  0  0  1  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  3  3  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  1  1  2  1  0  1  0  1  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  2  1  1  0  0  1  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  1  2  0  0  1  0  0  0  1  0  0  0  0  0  0]
 [ 0  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  0  0  1  1  1  0  0  0  0  1  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  1  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0]]
```




```
SVC Accuracy Score is:  0.2662835249042146
SVC F1 Score is:  0.07981144900170084
SVC Recall Score is:  0.10811441455864576
SVC Precision Score is:  0.06431958591135803
SVC Confusion Score is:  [[ 0  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
0  0  0  0]
 [ 0  0  0  4  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0 13  4  3  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0 19  0 11  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  9  0 48 11  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  2  0 57 33  5  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0 32 55 33 13  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0 17 57 71 45  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0 10 18 51 76  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0 10 15 25 68  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  4 12 16 31  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  1  8 11 26  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  3  7 26  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  1  7  3 16  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  2  1  6  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  2  0  9  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  1  0  0  6  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  7  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  5  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  5  0  0  0  0  0  0  0  0  0  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  0  0]
 [ 0  0  0  0  0  0  0  0  4  0  0  0  0  0  0  0  0  0  0  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  0  0]]
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metr
ics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and be
ing set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```




```
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
Accuracy of each flod is: [0.1423444976076555, 0.34251497005988024, 0.22035928143712574, 0.2682634730538922, 0.31736526946107785]
Average accuracy is: 0.2581694983239263
F1 of each flod is [0.06286192693801045, 0.18427659401063293, 0.09871559100128074, 0.10999755118631493, 0.13752298233153204]:
Average F1 Score is: 0.11867492909355422
Recall of each flod is: [0.09795883738996175, 0.19655362295005901, 0.14829405615622318, 0.13237695925043166, 0.15543090037172674]
Average recall score is: 0.14612287522368045
Precision of each flod is: [0.046960445028547786, 0.18430492281791283, 0.07745782000404643, 0.10047738326953323, 0.13070424635437244]
Average precision score is: 0.10798096349488255
```

```
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\ridva\AppData\Local\Programs\Python\Python39\lib\site-packages\sklearn\metrics\_classification.py:1308: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```