In [ ]:
```
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
import matplotlib.pyplot as plt
from sklearn.metrics import accuracy_score
```

In [ ]:
```
data = pd.read_csv('C:\\Users\\huzai\\OneDrive\\Desktop\\fruit.txt', delimiter='\t')
```

In [ ]:
```
data.head()
```

Out[ ]:

	fruit_label	fruit_name	fruit_subtype	mass	width	height	color_score
0	1	apple	granny_smith	192	8.4	7.3	0.55
1	1	apple	granny_smith	180	8.0	6.8	0.59
2	1	apple	granny_smith	176	7.4	7.2	0.60
3	2	mandarin	mandarin	86	6.2	4.7	0.80
4	2	mandarin	mandarin	84	6.0	4.6	0.79

In [ ]:
```
# Select features and target variable
X = data[['mass', 'width', 'height', 'color_score']]
y = data['fruit_label']
```

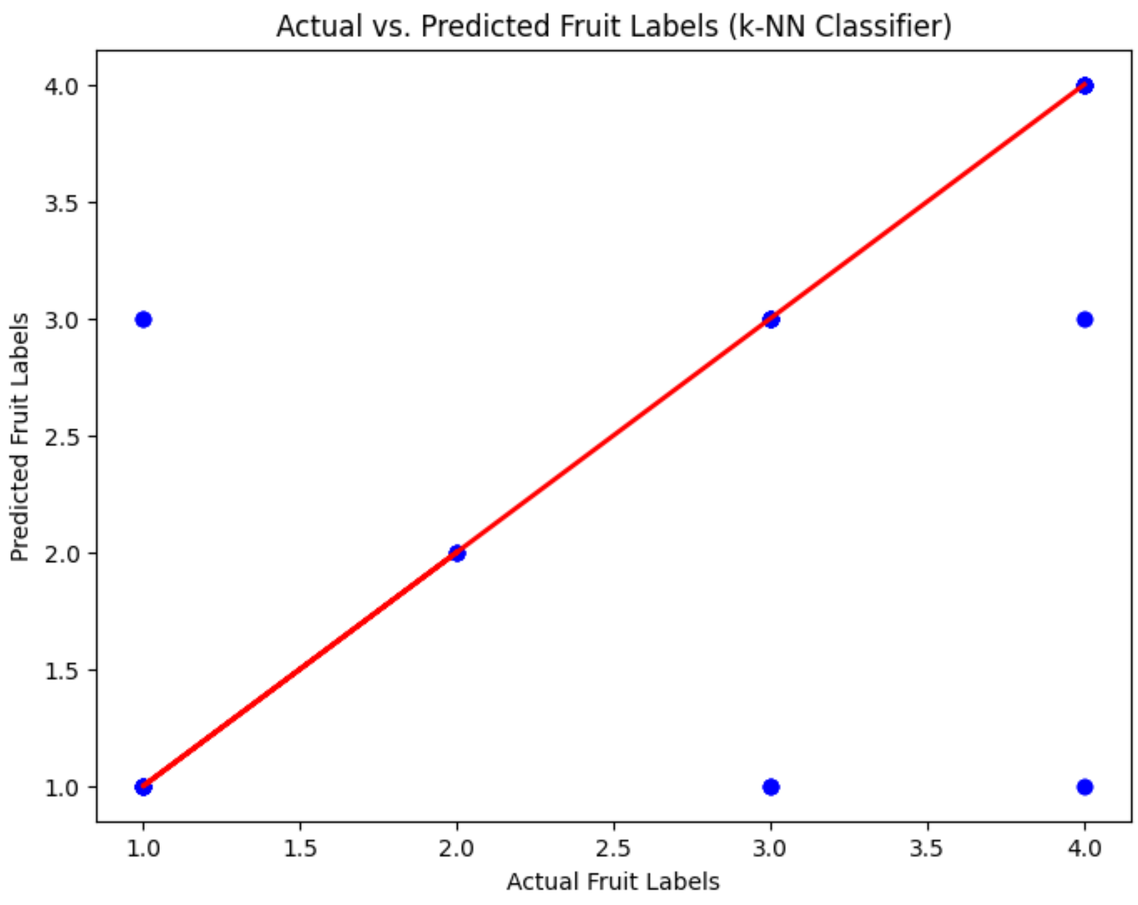
In [ ]:
```
k = 3
knn_classifier = KNeighborsClassifier(n_neighbors=k)

# Train the k-NN classifier using the training data
knn_classifier.fit(X, y)
```

Out[ ]:
```
KNeighborsClassifier
KNeighborsClassifier(n_neighbors=3)
```

In [ ]:
```
predictions = knn_classifier.predict(X)
```

In [ ]:
```
plt.figure(figsize=(8, 6))
plt.scatter(y, predictions, color='blue', label='Actual vs. Predicted')
plt.plot(y, y, color='red', linewidth=2, label='Regression Line')
plt.xlabel('Actual Fruit Labels')
plt.ylabel('Predicted Fruit Labels')
plt.title('Actual vs. Predicted Fruit Labels (k-NN Classifier)')
plt.show()
```



In [ ]:
```
accuracy = accuracy_score(y, predictions)
print(f'Accuracy: {accuracy * 100:.2f}%')
```

```
Accuracy: 81.36%
```

In [ ]:
```
result_df = pd.DataFrame({'Actual Price': y, 'Predicted Price': predictions})
print(result_df.head(20))
```

```
    Actual Price  Predicted Price
0              1                1
1              1                1
2              1                1
3              2                2
4              2                2
5              2                2
6              2                2
7              2                2
8              1                1
9              1                1
10             1                1
11             1                1
12             1                3
13             1                1
14             1                1
15             1                1
16             1                1
17             1                1
18             1                1
19             1                1
```