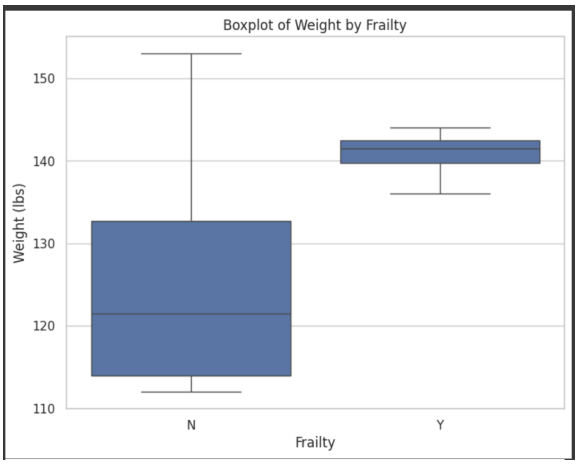
PDS Assignment (Question – 1)

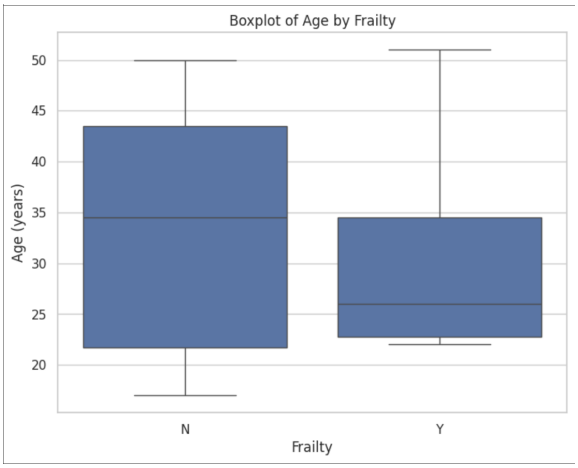
Name: Sai Pavan Pratapagiri
ID: 16343743

Output/Result for Preprocessing the given Data:

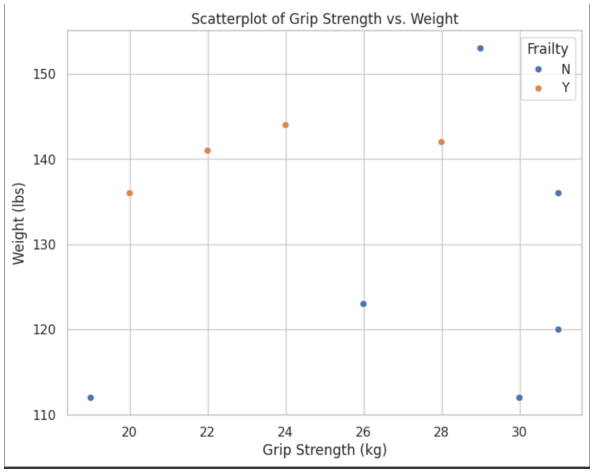
```
       Height (Inches)    Weight (Pounds)        Age  Grip strength
count         10.000000        10.000000  10.000000      10.000000
mean          68.600000       131.900000  32.500000      26.000000
std            1.670662        14.231811  12.860361       4.521553
min           65.800000       112.000000  17.000000      19.000000
25%           67.825000       120.750000  22.250000      22.500000
50%           68.450000       136.000000  29.500000      27.000000
75%           69.700000       141.750000  43.500000      29.750000
max           71.500000       153.000000  51.000000      31.000000
Index(['Height', 'Weight', 'Age', 'Grip_Strength', 'Frailty'], dtype='object')
(10, 5)
   Height  Weight  Age  Grip_Strength Frailty
0    65.8     112   30             30       N
1    71.5     136   19             31       N
2    69.4     153   45             29       N
3    68.2     142   22             28       Y
4    67.8     144   29             24       Y
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Height         10 non-null     float64
 1   Weight         10 non-null     int64
 2   Age            10 non-null     int64
 3   Grip_Strength  10 non-null     int64
 4   Frailty        10 non-null     object
dtypes: float64(1), int64(3), object(1)
memory usage: 528.0+ bytes
None
Height           False
Weight           False
Age              False
Grip_Strength    False
Frailty          False
dtype: bool
```

Visualization: Boxplot – Distribution of weight, age by frailty





Scatter Plot - Grip Strength(kgs) by weight.



- **Importing Libraries:**
  The code begins by importing the required libraries. pandas is imported as pd for data manipulation, and ttest_ind from scipy.stats is imported for performing the independent t-test.
- **Reading Data**:
  The code reads the CSV file containing the frailty data into a DataFrame named frailty_data.
- **Converting to Categorical:**
  The code converts the 'Frailty' column to a categorical variable using the pd.Categorical() function. This step is often useful for defining categorical variables explicitly, which can aid in subsequent analysis and visualization.
- **Grouping Data:**
  It separates the weight data into two groups based on the 'Frailty' column. One group comprises individuals classified as non-frail ('N'), and the other group comprises individuals classified as frail ('Y').
- **Performing t-test:**
  It performs an independent two-sample t-test (ttest_ind) to compare the mean weight between the frail and non-frail groups. The t-test is conducted assuming unequal variances between the groups (equal_var=False), which is a common approach when the assumption of equal variances is violated.
- **Printing Results:**
  The code prints out the results of the t-test, including the t-statistic and the p-value.

Output/result for Analysis:

```
Summary of the dataset:
          Height      Weight        Age  Grip_Strength
count  10.000000   10.000000  10.000000      10.000000
mean   68.600000  131.900000  32.500000      26.000000
std     1.670662   14.231811  12.860361       4.521553
min    65.800000  112.000000  17.000000      19.000000
25%    67.825000  120.750000  22.250000      22.500000
50%    68.450000  136.000000  29.500000      27.000000
75%    69.700000  141.750000  43.500000      29.750000
max    71.500000  153.000000  51.000000      31.000000

T-test Results:
T-statistic: 2.1964625958466355
P-value: 0.07309706417382401
```