Hw 1

September 7, 2022

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

1 Problem 1

Origin is qualitative. The rest are qualitative. (Note: I do not consider name to be a predictor)

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

1.1 Range and Mean and Standard Deviation for All data points

[3]:
```
df.iloc[:,:7].describe()
```

[3]:
```
              mpg   cylinders  displacement       weight  acceleration  \
count  397.000000  397.000000    397.000000   397.000000    397.000000
mean    23.515869    5.458438    193.532746  2970.261965     15.555668
std      7.825804    1.701577    104.379583   847.904119      2.749995
min      9.000000    3.000000     68.000000  1613.000000      8.000000
25%     17.500000    4.000000    104.000000  2223.000000     13.800000
50%     23.000000    4.000000    146.000000  2800.000000     15.500000
75%     29.000000    8.000000    262.000000  3609.000000     17.100000
max     46.600000    8.000000    455.000000  5140.000000     24.800000

             year
count  397.000000
mean    75.994962
std      3.690005
min     70.000000
25%     73.000000
50%     76.000000
75%     79.000000
max     82.000000
```

1.2 Range Mean and Standard Dev for Datset excluding the 10th through 85th entries

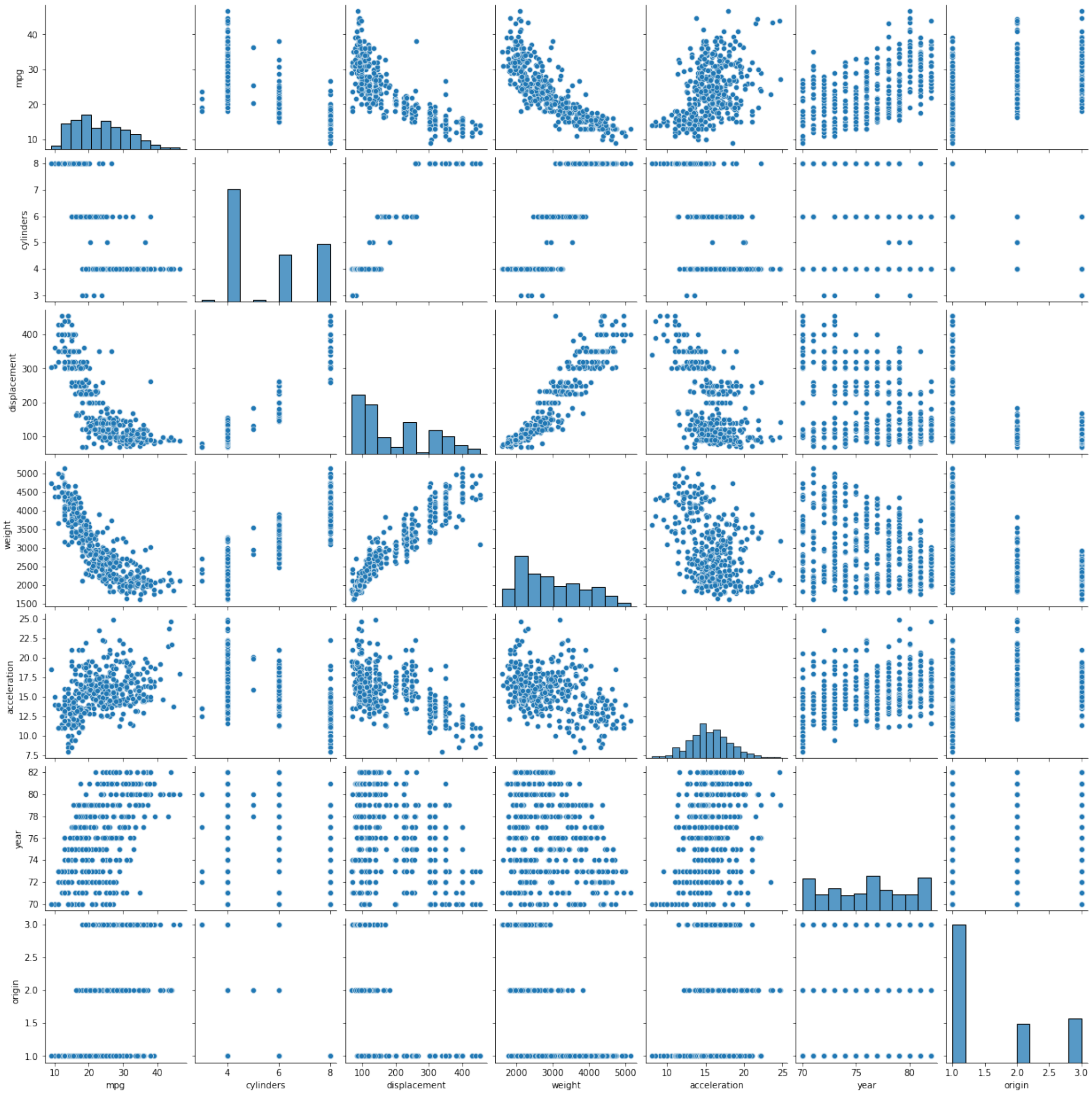
```
[4]: df.drop(df.index[10:85]).iloc[:,:7].describe()
```

```
[4]:               mpg   cylinders  displacement       weight  acceleration  \
     count  322.000000  322.000000    322.000000   322.000000    322.000000
     mean    24.409317    5.378882    187.680124  2936.807453     15.700621
     std      7.913357    1.657398    100.120925   810.987533      2.706436
     min     11.000000    3.000000     68.000000  1649.000000      8.500000
     25%     18.000000    4.000000    100.250000  2216.000000     14.000000
     50%     23.900000    4.000000    145.500000  2797.500000     15.500000
     75%     30.650000    6.000000    250.000000  3516.000000     17.275000
     max     46.600000    8.000000    455.000000  4997.000000     24.800000

                  year
     count  322.000000
     mean    77.130435
     std      3.131849
     min     70.000000
     25%     75.000000
     50%     77.000000
     75%     80.000000
     max     82.000000
```

1.3 Graphical Investigation of Predictors

```
[5]: sns.pairplot(df)
```

```
[5]: <seaborn.axisgrid.PairGrid at 0x1ba0bc8ea00>
```

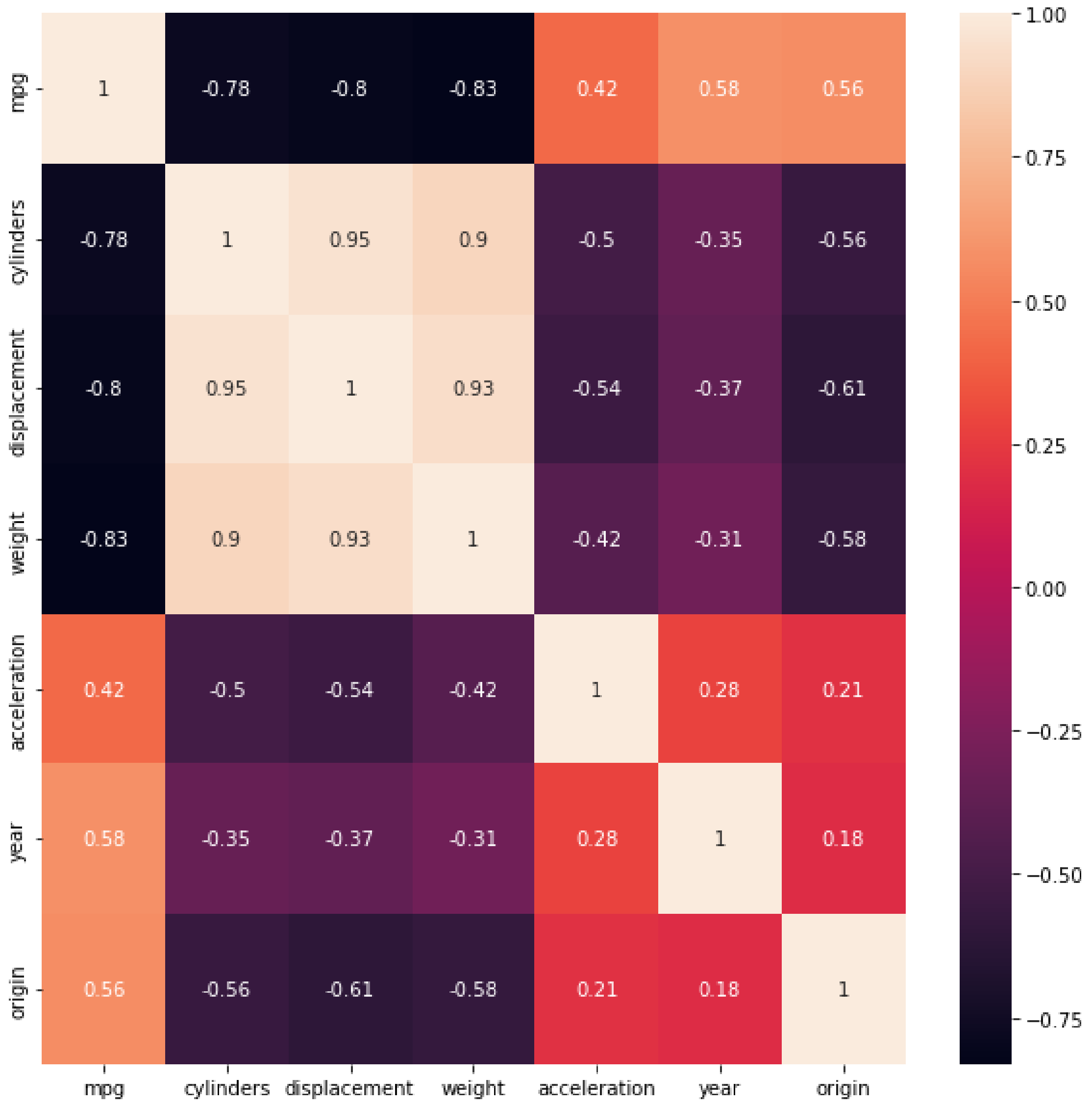


Some of the predictors are correlated with each other like displacement and weight. This could introduce confounding variable problem.

1.4 Correlation Matrix of Dataset

[6]:
```
plt.figure(figsize = (10,10))
sns.heatmap(df.corr(),annot = True)
```

[6]:
```
<AxesSubplot:>
```



All Predictors seem to have good Correlation with MPG, though origin needs to be one hot encoded to be used.

2 Problem 2

[7]:
```
df = pd.read_csv('CodParasite.txt', sep="\t", index_col = 0)
df["log_intensity"] = np.log(1+ df["Intensity"])
```

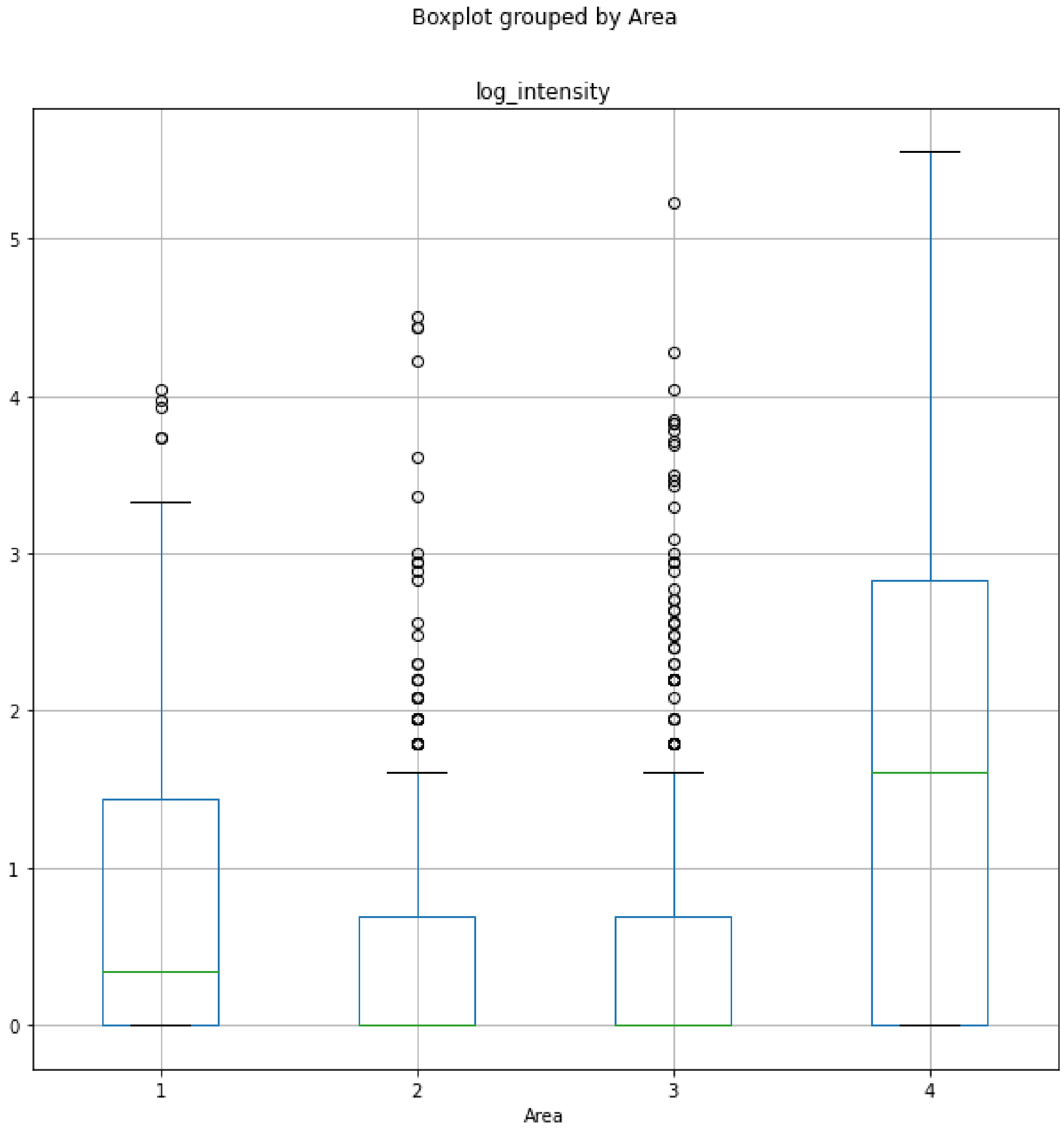
[8]:
```
df
```

[8]:
```
        Intensity  Prevalence  Year  Depth  Weight  Length  Sex  Stage  Age  \
Sample
```

```
1             0.0           0  1999    220   148.0    26.0    0      0    0
2             0.0           0  1999    220   144.0    26.0    0      0    0
3             0.0           0  1999    220   146.0    27.0    0      0    0
4             0.0           0  1999    220   138.0    26.0    0      0    0
5             0.0           0  1999    220    40.0    17.0    0      0    0
…           …         …   …    …     …     …  …    …  …
1250         90.0           1  2001    228   224.0    31.0    1      1    2
1251        104.0           1  2001    140   690.0    43.0    2      1    3
1252        125.0           1  2001    140   754.0    44.0    2      1    3
1253        128.0           1  2001    140  1270.0    55.0    2      4    7
1254        257.0           1  2001    228   370.0    35.0    2      1    3

        Area  log_intensity
Sample
1          2       0.000000
2          2       0.000000
3          2       0.000000
4          2       0.000000
5          2       0.000000
…      …            …
1250       4       4.510860
1251       4       4.653960
1252       4       4.836282
1253       4       4.859812
1254       4       5.552960

[1254 rows x 11 columns]
```

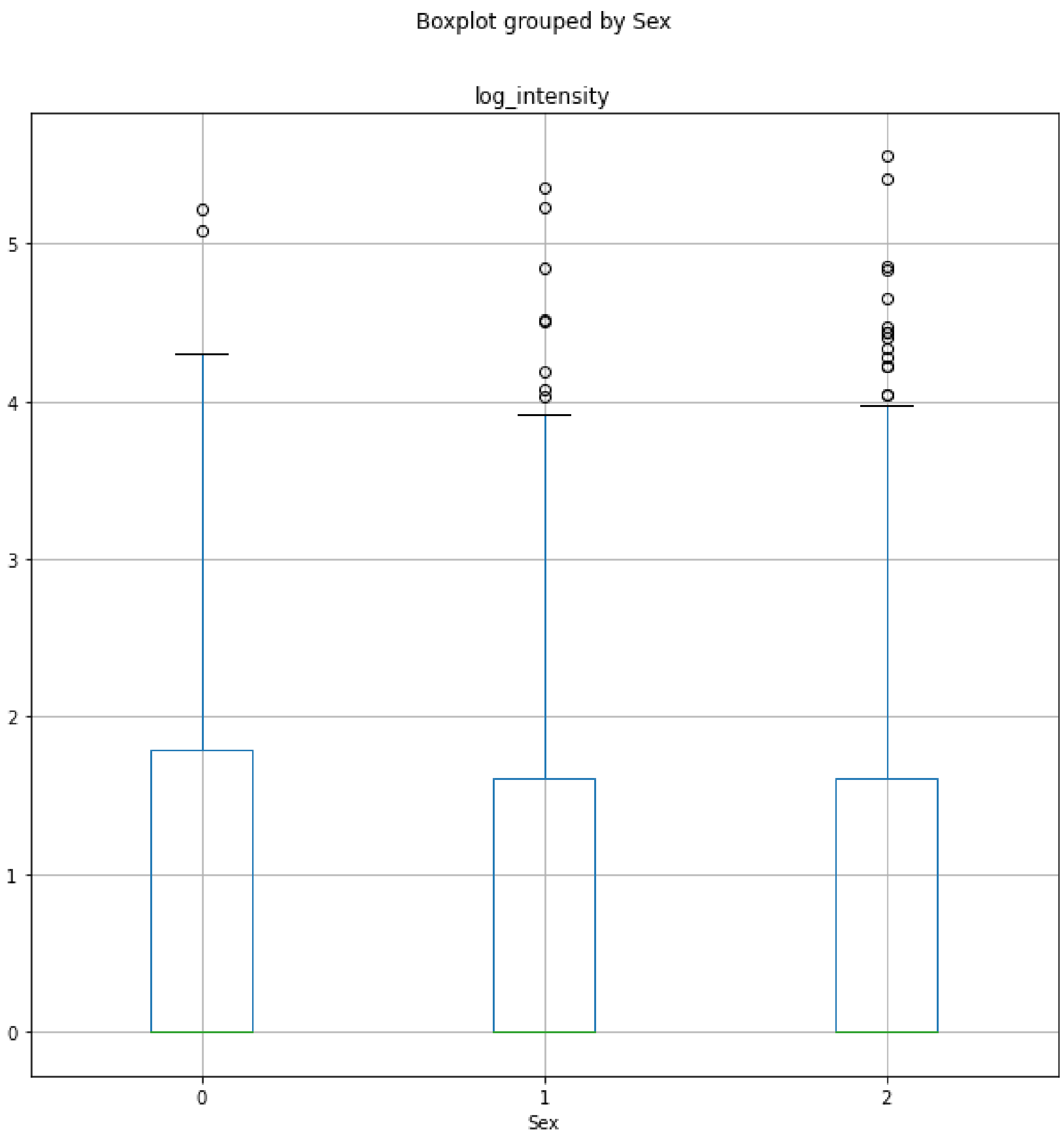
```
[9]: df.boxplot(column = ["log_intensity"], by = "Area", figsize = (10,10))
```

```
[9]: <AxesSubplot:title={'center':'log_intensity'}, xlabel='Area'>
```



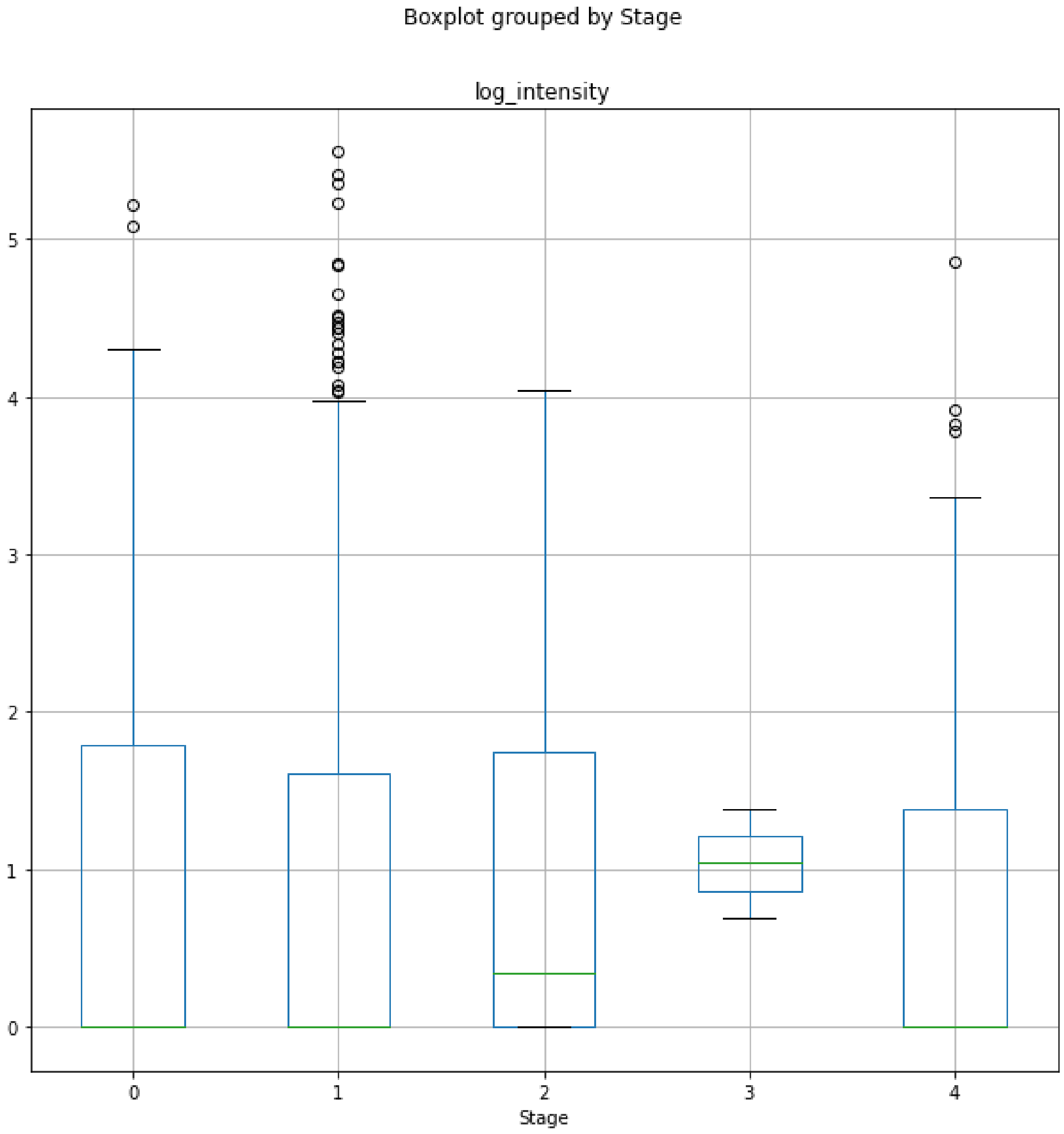
```
[10]: df.boxplot(column = ["log_intensity"], by = "Sex", figsize = (10,10))
```

```
[10]: <AxesSubplot:title={'center':'log_intensity'}, xlabel='Sex'>
```



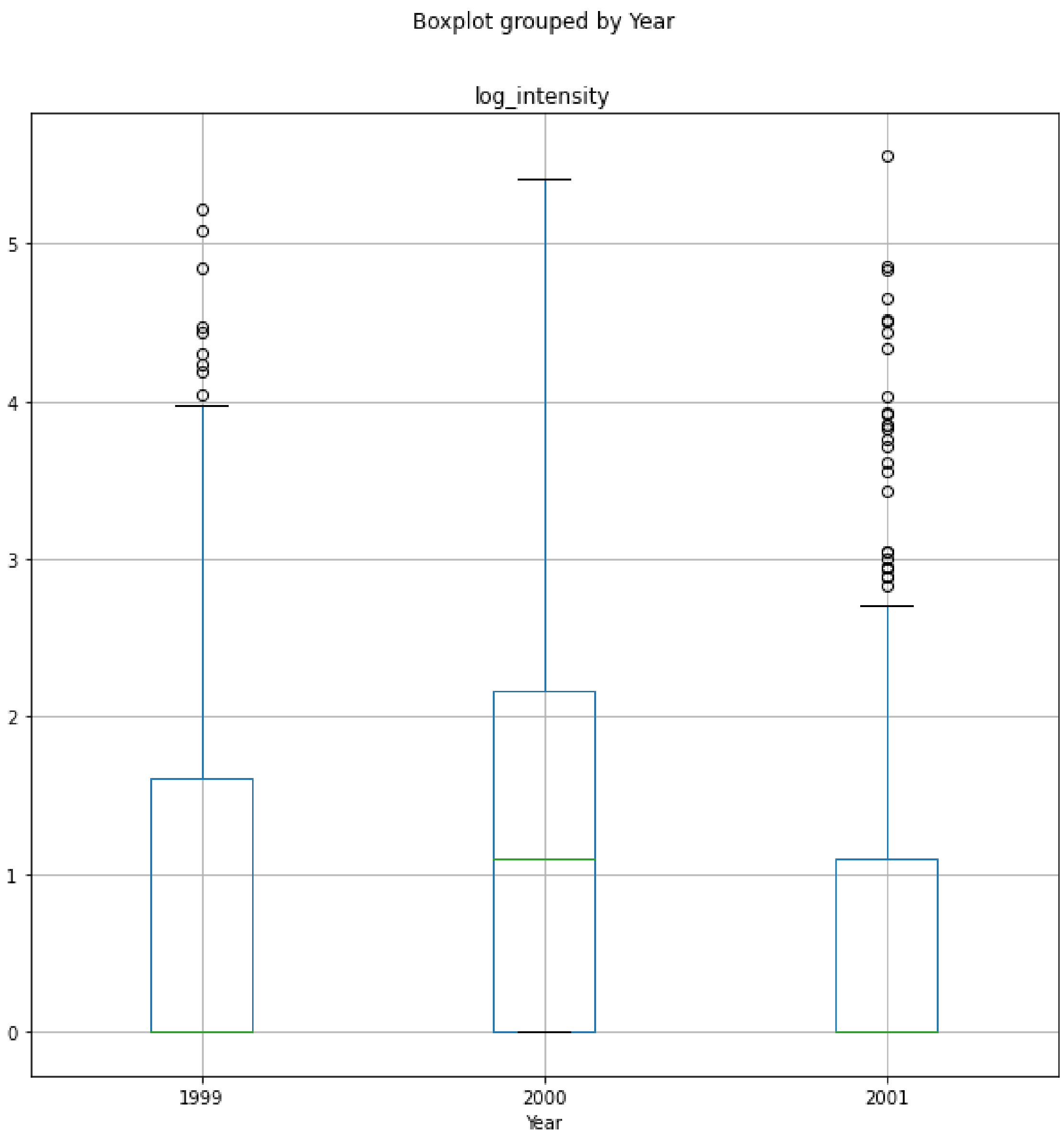
```
[11]: df.boxplot(column = ["log_intensity"], by = "Stage", figsize = (10,10))
```

```
[11]: <AxesSubplot:title={'center':'log_intensity'}, xlabel='Stage'>
```



```
[12]: df.boxplot(column = ["log_intensity"], by = "Year", figsize = (10,10))
```

```
[12]: <AxesSubplot:title={'center':'log_intensity'}, xlabel='Year'>
```



Area seems the most likely to be a good predictor of parasites since the distribution of each area is different from each other. All other box plots look similar to each other within the class.

3 Problem 3

```
[13]: df = pd.read_csv('Owls.txt', sep="\t")
      df
```

```
[13]:          Nest FoodTreatment SexParent  ArrivalTime  SiblingNegotiation  \
      0  AutavauxTV      Deprived      Male        22.25                   4
      1  AutavauxTV      Satiated      Male        22.38                   0
```

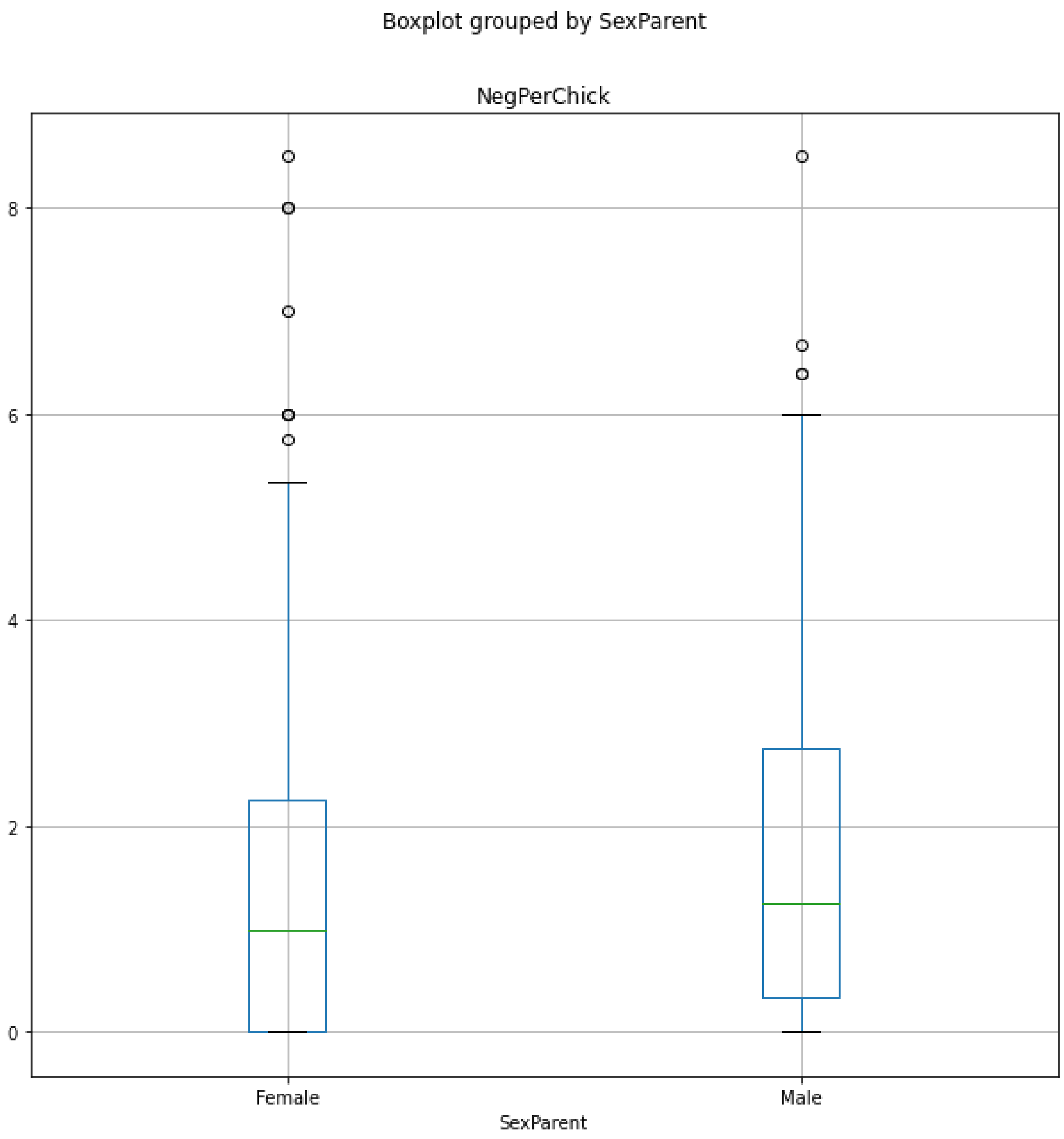
```
2    AutavauxTV      Deprived      Male        22.53                     2
3    AutavauxTV      Deprived      Male        22.56                     2
4    AutavauxTV      Deprived      Male        22.61                     2
..            …            …       …          …                 …
594     Yvonnand     Deprived    Female        27.25                     7
595     Yvonnand     Deprived      Male        28.45                     5
596     Yvonnand     Deprived    Female        28.86                    15
597     Yvonnand     Deprived      Male        29.21                    10
598     Yvonnand     Satiated    Female        29.23                     0

     BroodSize  NegPerChick
0            5     0.800000
1            5     0.000000
2            5     0.400000
3            5     0.400000
4            5     0.400000
..         …          …
594          7     1.000000
595          7     0.714286
596          7     2.142857
597          7     1.428571
598          7     0.000000

[599 rows x 7 columns]
```

[14]:

```python
df.boxplot(column = ["NegPerChick"], by = "SexParent", figsize = (10,10))
```

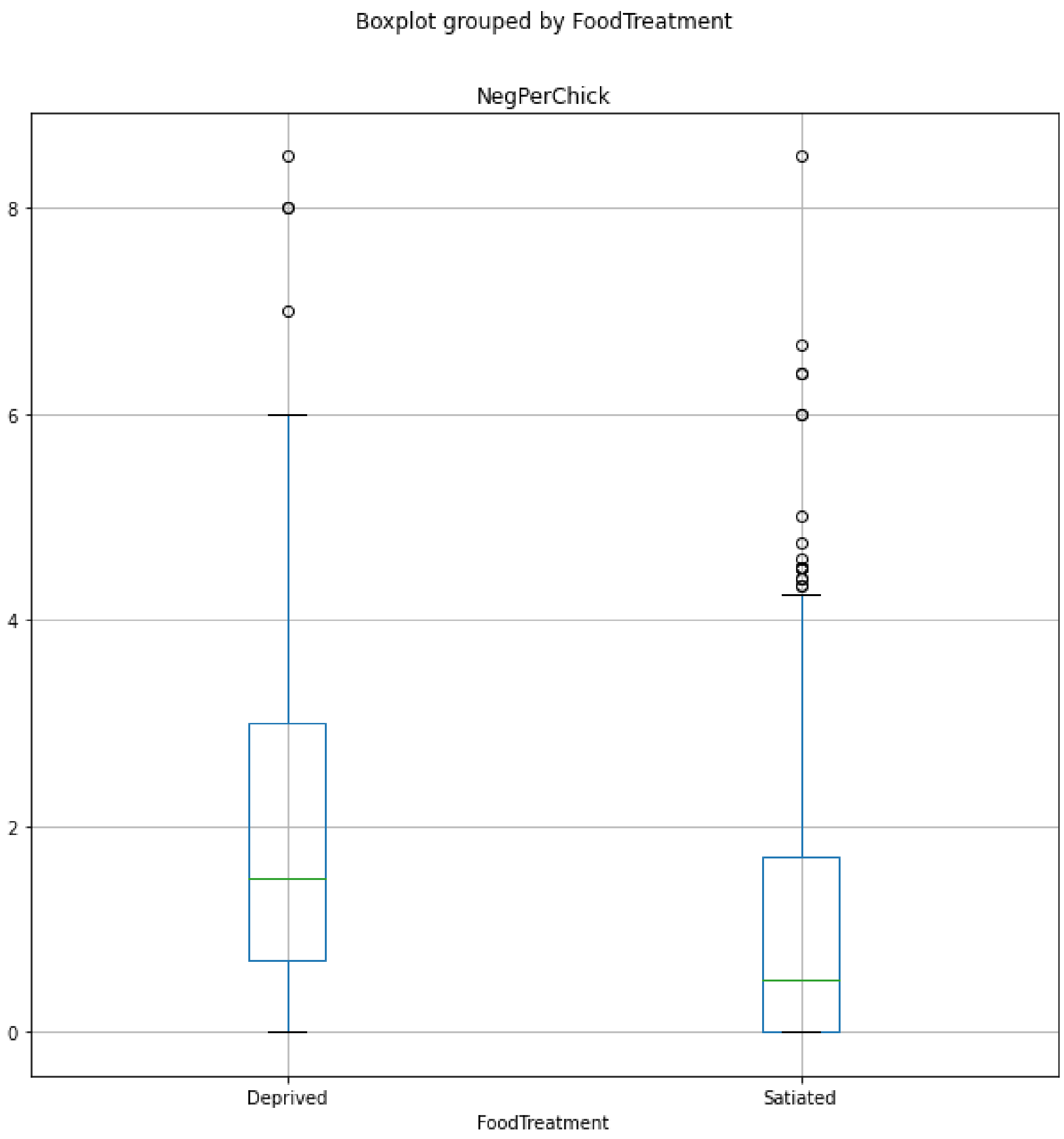
[14]:

```
<AxesSubplot:title={'center':'NegPerChick'}, xlabel='SexParent'>
```



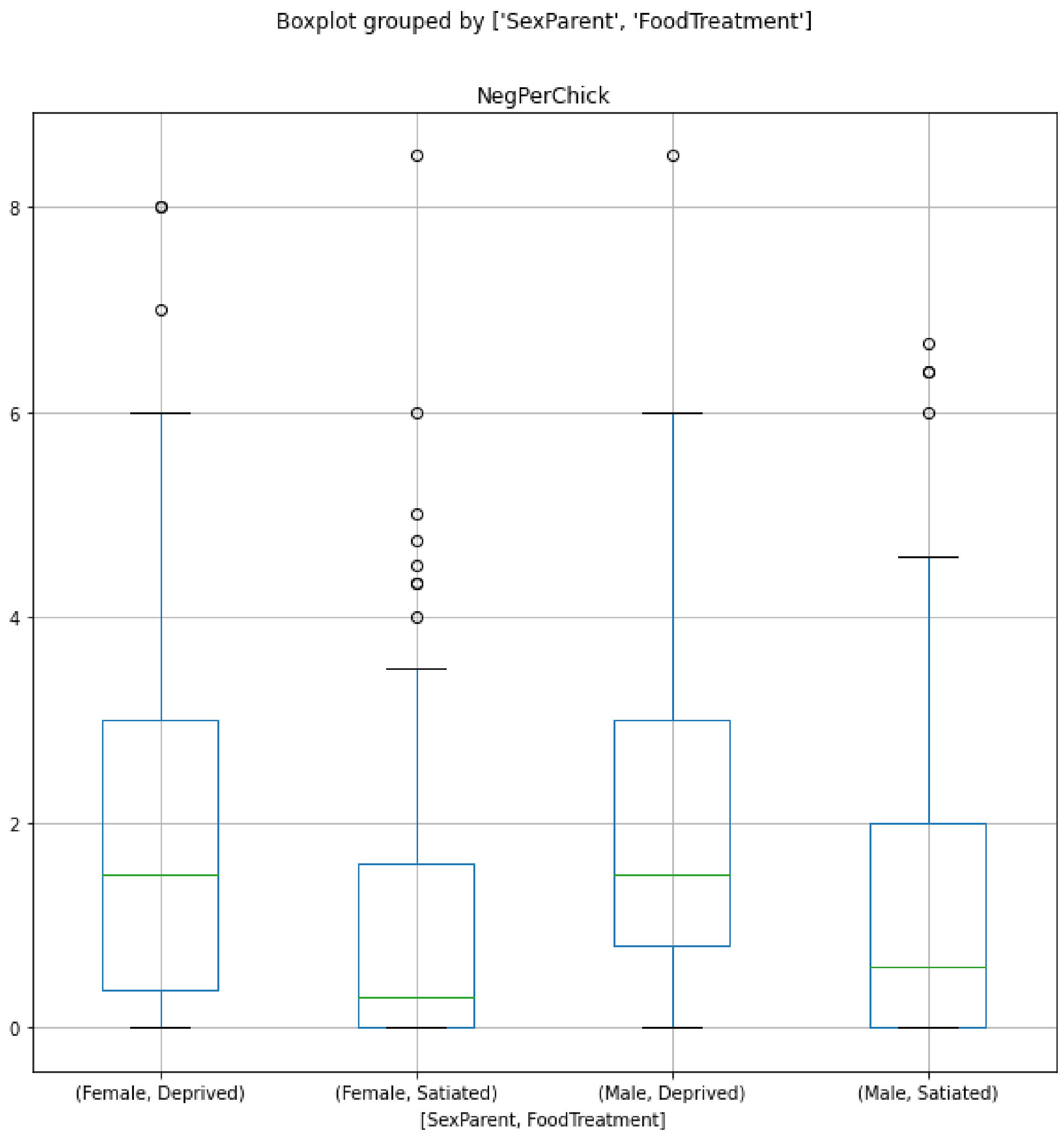
```
[15]: df.boxplot(column = ["NegPerChick"], by = "FoodTreatment", figsize = (10,10))
```

```
[15]: <AxesSubplot:title={'center':'NegPerChick'}, xlabel='FoodTreatment'>
```



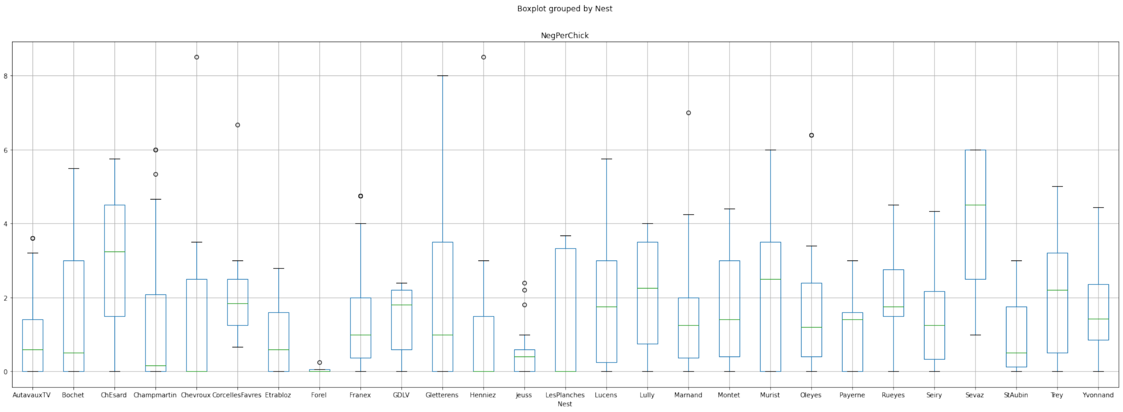
```
[16]: df.boxplot(column = ["NegPerChick"], by = ["SexParent","FoodTreatment"],␣
       ↪figsize = (10,10))
```

```
[16]: <AxesSubplot:title={'center':'NegPerChick'}, xlabel='[SexParent,
      FoodTreatment]'>
```



```
[17]: df.boxplot(column = ["NegPerChick"], by = "Nest", figsize = (30,10))
```

```
[17]: <AxesSubplot:title={'center':'NegPerChick'}, xlabel='Nest'>
```



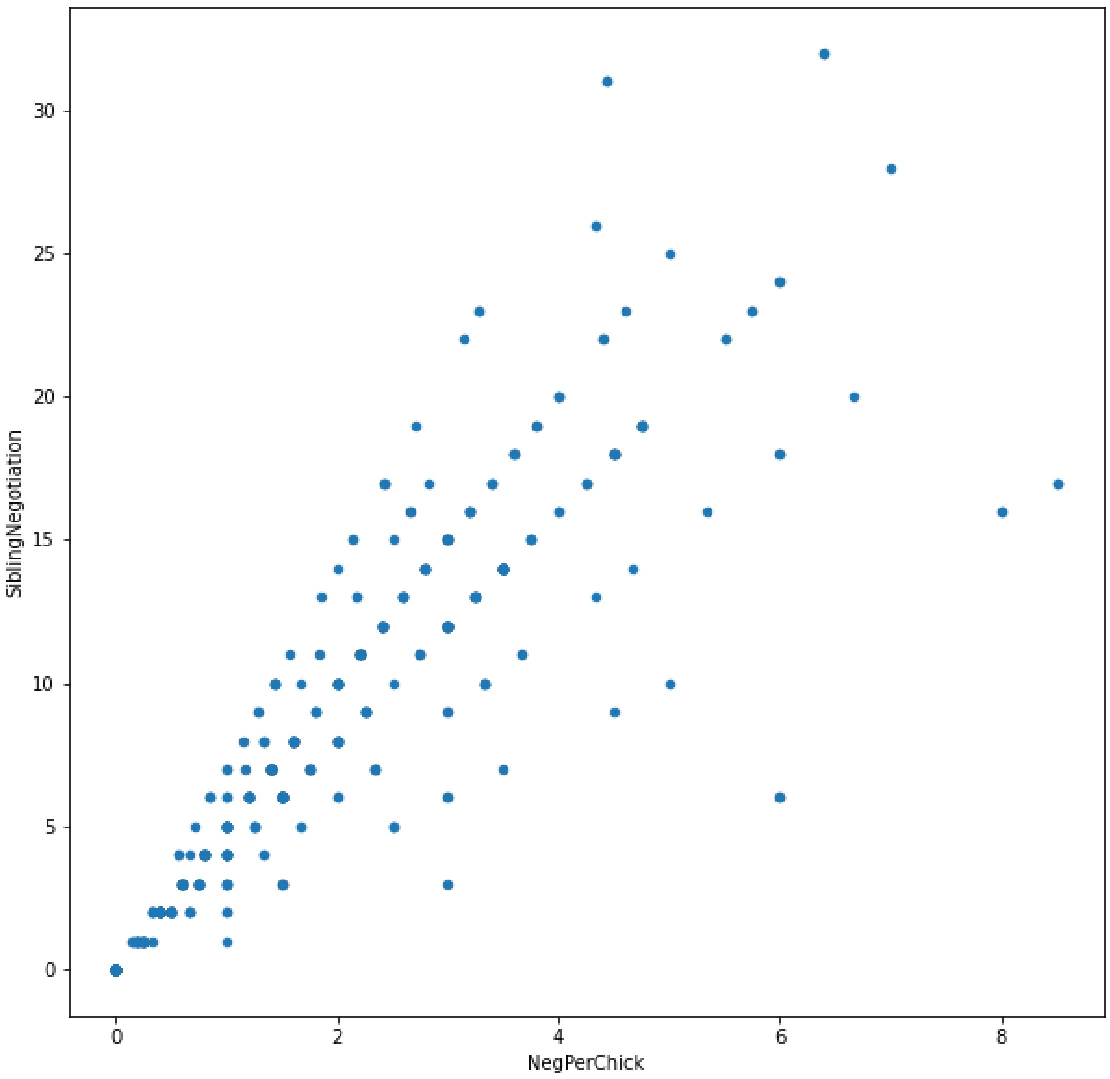
There do seem to be differences across different nesting sites as the distributions differ. This could also be due to the data having less points per nesting site thus increasing the variance.

[18]:

```
df.plot.scatter(x="NegPerChick", y = "SiblingNegotiation", figsize = (10,10))
```

[18]:

```
<AxesSubplot:xlabel='NegPerChick', ylabel='SiblingNegotiation'>
```

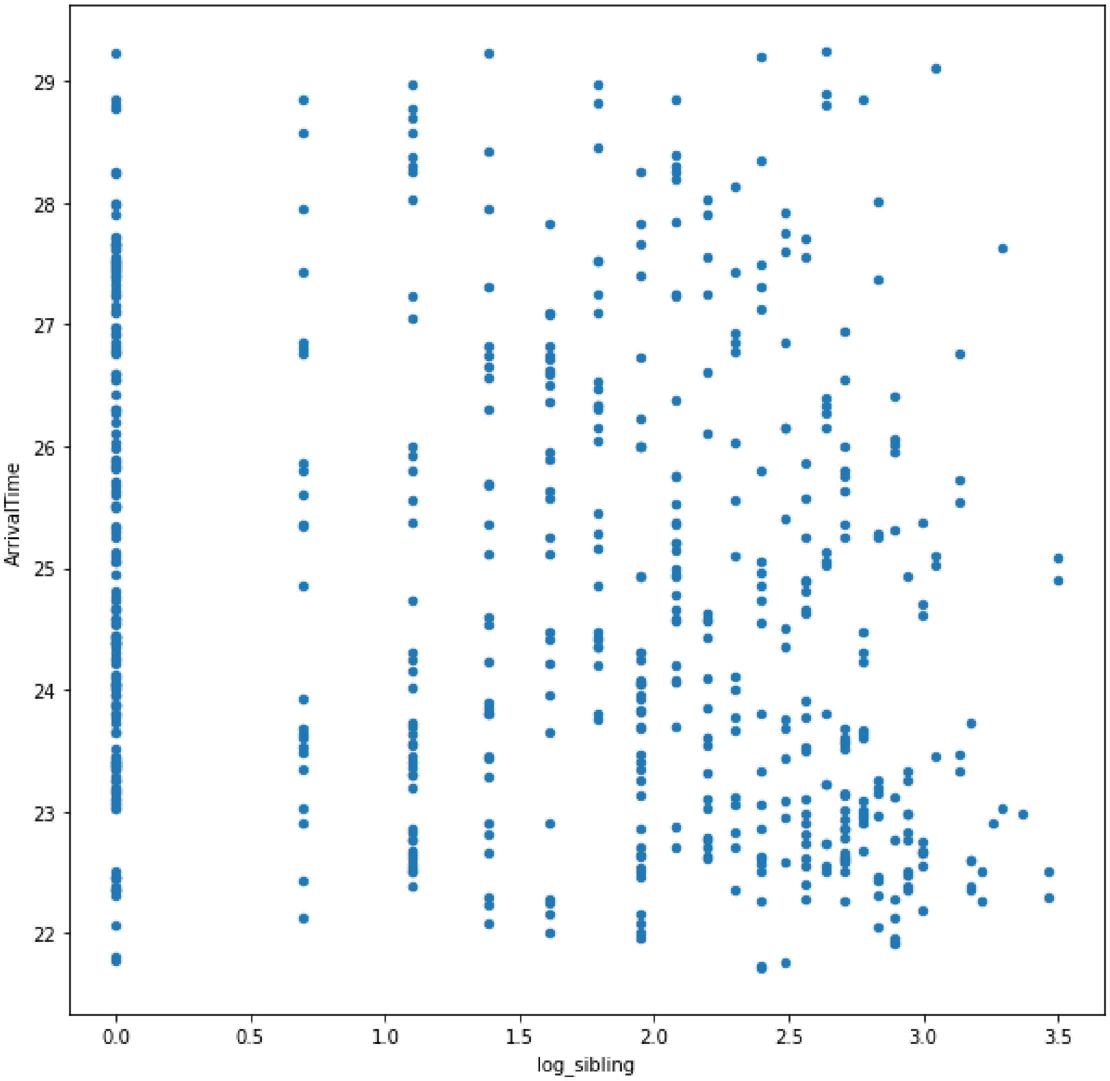


There seems to be some correlation with increasing variance as NegPerChick increases

```
[19]: df["log_sibling"] = np.log(1+df["SiblingNegotiation"])
```

```
[20]: df.plot.scatter(x="log_sibling", y = "ArrivalTime", figsize = (10,10))
```

```
[20]: <AxesSubplot:xlabel='log_sibling', ylabel='ArrivalTime'>
```


ArrivalTime
22
23
24
25
26
27
28
29
log_sibling
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5