Importing required Python Modules

In [1]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
```

#Setting the name of the columns to be Age, Op_Year, axil_nodes_det, Surv_status

Creating seperate dataset for each label of feature Surv_status

In [2]:

```
dataSet = pd.read_csv("haberman.csv", names = ["Age", "Op_Year", "axil_nodes_det", "Surv_status"])
dataSet1 = dataSet.loc[dataSet["Surv_status"] == 1];
dataSet2 = dataSet.loc[dataSet["Surv_status"] == 2];
```

#Fetching Metadata for the haberman dataset

In [3]:

```
rows, columns = dataSet.shape
column_name = list(dataSet)
print("Metadata for haberman dataset is:")
print("1. Number of points in dataset is: {} and Number of features {}".format(rows,columns))
print("2. Column names are {}".format(", ".join(column_name)))
print("3. Independent Variables are : {}".format(", ".join(column_name[:-1])))
print("4. Dependent Variables are : {}".format(column_name[-1]))
print("5. No of data points for each survival status is :")
print(dataSet[column_name[-1]].value_counts().to_frame())
```

```
Metadata for haberman dataset is:
1. Number of points in dataset is: 306 and Number of features 4
2. Column names are Age, Op_Year, axil_nodes_det, Surv_status
3. Independent Variables are : Age, Op_Year, axil_nodes_det
4. Dependent Variables are : Surv_status
5. No of data points for each survival status is :
   Surv_status
1          225
2           81
```

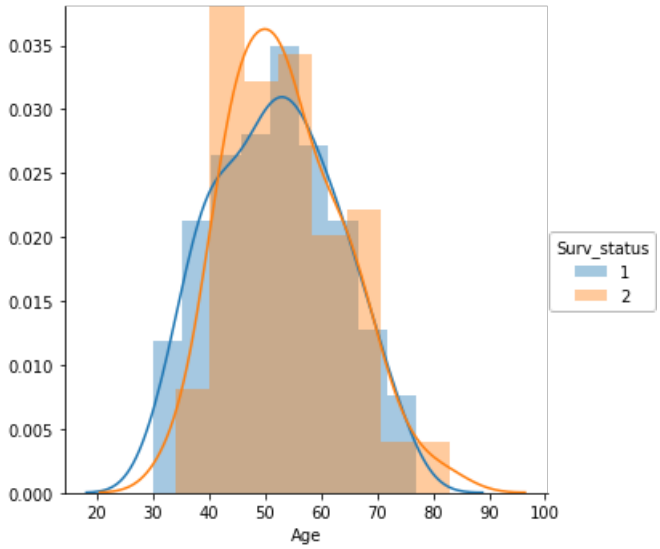
The imbalance class is present for feature Surv_status

#Objective of further analysis to identify best features for survival clasification

Performing Univariate analysis on each feature

In [4]:

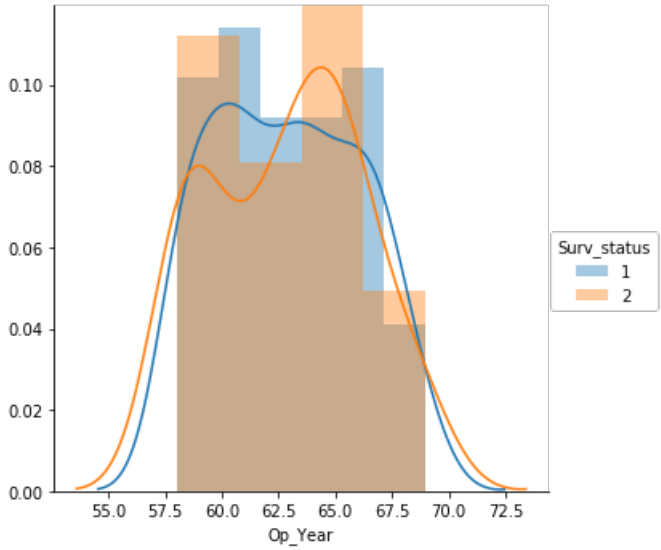
```
#For feature Age
sns.FacetGrid(dataSet, hue="Surv_status", size=5).map(sns.distplot, "Age").add_legend();
plt.show();
```



The above histogram shows that the data is almost normally distributed, which implies that the value of mean, median and MAD, Standard deviation will almost be same

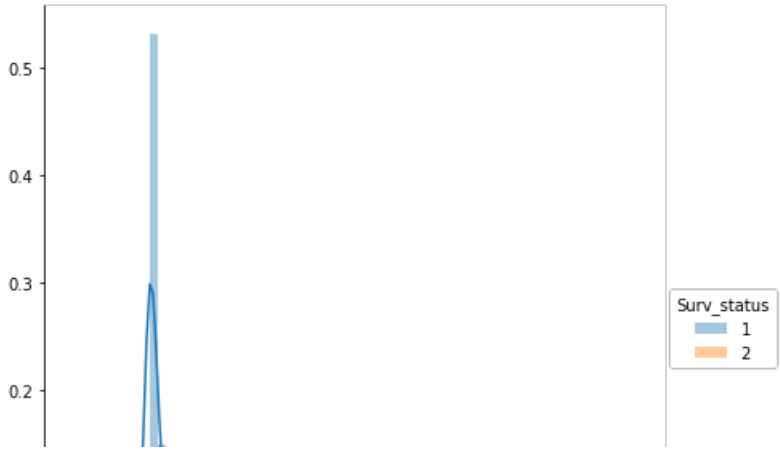
In [5]:

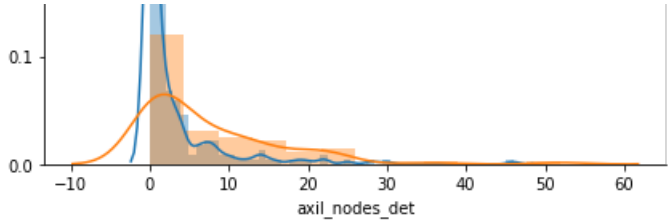
```
#For feature Op_Year
sns.FacetGrid(dataSet, hue="Surv_status", size=5).map(sns.distplot, "Op_Year").add_legend();
plt.show();
```



In [6]:

```
#For feature axil_nodes_det
sns.FacetGrid(dataSet, hue="Surv_status", size=6).map(sns.distplot, "axil_nodes_det").add_legend();
plt.show();
```





Observation:

1. None of the above histogram provides the clear boundary to distinguish output feature Surv_status
2. For feature axil_nodes_det we can say that data points value from 0 to 2 have higher chances of belonging to label 1 of Survival status.

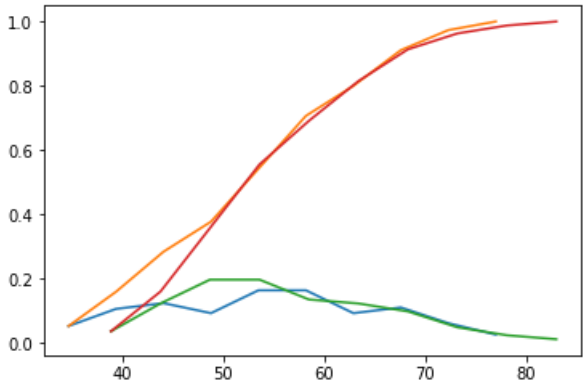
In [7]:

```
counts, bin_edges = np.histogram(dataSet1['Age'], bins=10, density = True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:], cdf)

counts, bin_edges = np.histogram(dataSet2['Age'], bins=10, density = True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:], cdf)
```

Out[7]:

```
[<matplotlib.lines.Line2D at 0xa83f068c>]
```



Observation:

1. All data points where Age <38 the label for survival status will be 1
2. All data points where Age >78 the label for survival status will be 2

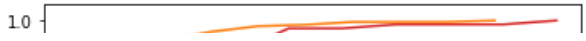
In [8]:

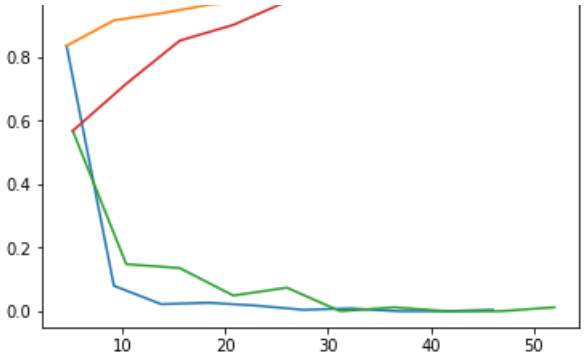
```
counts, bin_edges = np.histogram(dataSet1['axil_nodes_det'], bins=10, density = True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:], cdf)

counts, bin_edges = np.histogram(dataSet2['axil_nodes_det'], bins=10, density = True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:], cdf)
```

Out[8]:

```
[<matplotlib.lines.Line2D at 0xa81dc1ec>]
```





Observation:

1. All data points where axil_nodes_det >47 the label for survival status will be 2

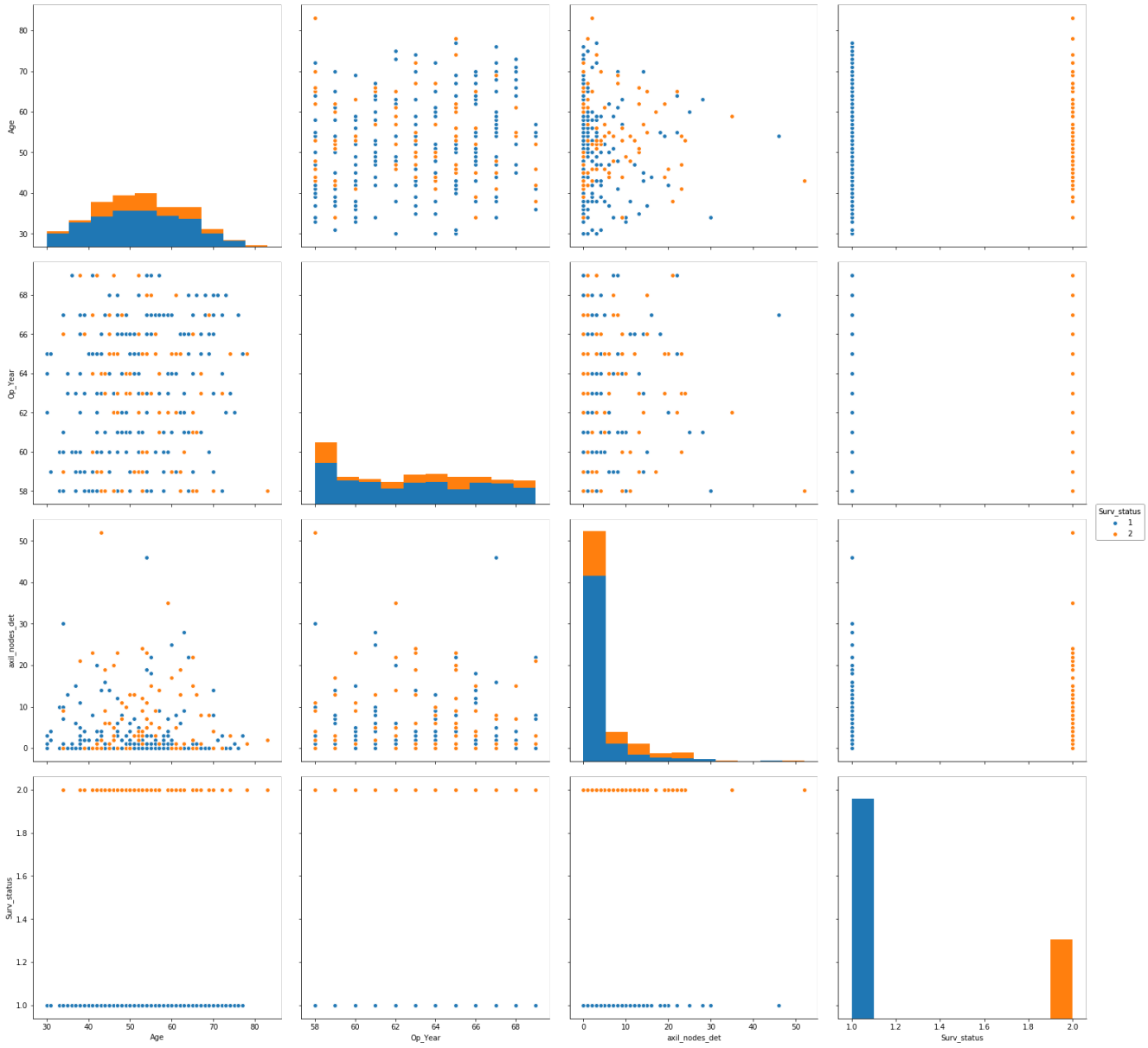
Performing Bivariate analysis on each feature

In [9]:

```
#Age, Op_Year, axil_nodes_det
sns.pairplot(dataSet, hue="Surv_status", size = 5).add_legend()
```

Out[9]:

```
<seaborn.axisgrid.PairGrid at 0xb43210ec>
```



Observation
1.This 2D scatterPlot between all features do not provide any significant information
2: Feaure axil_nodes_det is having outlier values, which may impact the outcome

Using describe function of Pandas to find mean, median, standard deviation

In [10]:

```
from statsmodels import robust
print ("\nMedian Absolute Deviation")
print(robust.mad(dataSet1["axil_nodes_det"]))
dataSet1.describe()
```

```
Median Absolute Deviation
0.0
```

Out[10]:

	Age	Op_Year	axil_nodes_det	Surv_status
count	225.000000	225.000000	225.000000	225.0
mean	52.017778	62.862222	2.791111	1.0
std	11.012154	3.222915	5.870318	0.0
min	30.000000	58.000000	0.000000	1.0
25%	43.000000	60.000000	0.000000	1.0
50%	52.000000	63.000000	0.000000	1.0
75%	60.000000	66.000000	3.000000	1.0
max	77.000000	69.000000	46.000000	1.0

Observation

1. For feature Age and Op_Year the mean and median are very close to each other representing normal distribution
2. So for features Age and Op_Year spread can be represented by standard deviations
3. For feature axil_nodes_det the means and median have significant difference hence calculating MAD(Median Absolute Deviation) to understand
4. MAD for axil_nodes_det is : 0.0

Lets verify if MAD return for feature against Surv_status Label = 1 is correct

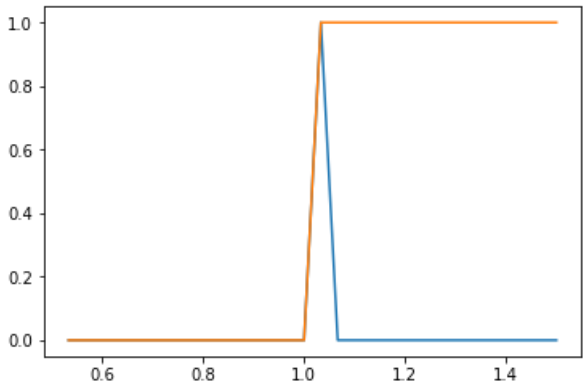
In [11]:

```
counts, bin_edges = np.histogram(dataSet1['Surv_status'], bins=30,
                                 density = True)
pdf = counts/(sum(counts))
print(pdf);
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:], cdf)
```

```
[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.]
[0.5         0.53333333 0.56666667 0.6         0.63333333 0.66666667
 0.7         0.73333333 0.76666667 0.8         0.83333333 0.86666667
 0.9         0.93333333 0.96666667 1.          1.03333333 1.06666667
 1.1         1.13333333 1.16666667 1.2         1.23333333 1.26666667
 1.3         1.33333333 1.36666667 1.4         1.43333333 1.46666667
 1.5        ]
```

Out[11]:

```
[<matplotlib.lines.Line2D at 0xa812bc6c>]
```



From the above diagram we can say that the value of MAD returned for feature Surv_status Label = 1 is correct. Since more than 50% of the data value is 0 in this case

Checking for MEan, Median and Standard Deviation for dataset of feature Surv_status with label 2

In [12]:

```
print ("\nMedian Absolute Deviation")
print(robust.mad(dataSet2["axil_nodes_det"]))
dataSet2.describe()
```

```
Median Absolute Deviation
5.930408874022407
```

Out[12]:

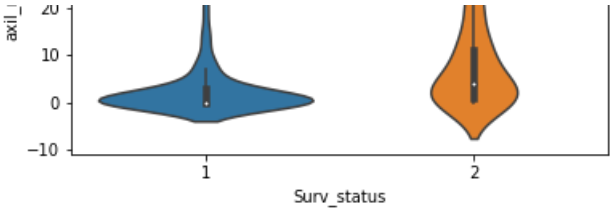
	Age	Op_Year	axil_nodes_det	Surv_status
count	81.000000	81.000000	81.000000	81.0
mean	53.679012	62.827160	7.456790	2.0
std	10.167137	3.342118	9.185654	0.0
min	34.000000	58.000000	0.000000	2.0
25%	46.000000	59.000000	1.000000	2.0
50%	53.000000	63.000000	4.000000	2.0
75%	61.000000	65.000000	11.000000	2.0
max	83.000000	69.000000	52.000000	2.0

Observation

1. For feature Age and Op_Year the mean and median are very close to each other representing normal distribution
2. So for features Age and Op_Year spread can be represented by standard deviations
3. For feature axil_nodes_det the means and median have significant difference hence calculating MAD(Median Absolute Deviation) to understand
4. MAD for axil_nodes_det is : 5.9304

In [13]:

```
sns.violinplot(x='Surv_status',y='axil_nodes_det', data=dataSet)
plt.show()
```



Conclusion

1. Dataset have imbalance class for feature survival status.
2. The data is almost normally distributed for features Age and Operations Year
3. For feature axil_nodes_det we can say that data points value from 0 to 2 have higher chances of belonging to label 1 of Survival status.
4. All data points where Age <38 the label for survival status will be 1
5. All data points where Age >78 the label for survival status will be 2
6. All data points where axil_nodes_det >47 the label for survival status will be 2
7. Outcome classification using if-else condition is not possible in this dataset, since their is high overlapping of the points.
8. Feaure axil_nodes_det is having outlier values, which may impact the outcome when complex classification technique is used for the classification