Objective - Classify the people who survived after breast cancer surgery.

```
In [118]: import numpy as np
          import pandas as pd
          import matplotlib.pyplot as plt
          import seaborn as sns
```

```
In [119]: #import the dataset
          dt = pd.read_csv("habermans-survival-data-set\haberman.csv")
          dt.columns = ["AGE","OP_YEAR","AXIL_NODES","SURVIVAL_STATUS"]
          dt.head()
```

Out[119]:

	AGE	OP_YEAR	AXIL_NODES	SURVIVAL_STATUS
0	30	62	3	1
1	30	65	0	1
2	31	59	2	1
3	31	65	4	1
4	33	58	10	1

```
In [120]: dt.shape
```

```
Out[120]: (305, 4)
```

```
In [121]: dt["SURVIVAL_STATUS"].value_counts()
```

```
Out[121]: 1    224
          2     81
          Name: SURVIVAL_STATUS, dtype: int64
```

Observation :

1. The dataset contains 305 records and 4 columns.
2. Out of 4 columns, 3 columns(AGE,OP_YEAR and AXIL_NODES) are input variables and 1 (SURVIVAL_STATUS) class variable.
3. The dataset contains 224 data points which belongs to Class 1 and 81 data points belongs to Class 2.
4. The dataset is an imbalance dataset.

```
In [122]: dt.describe()
```

Out[122]:

	AGE	OP_YEAR	AXIL_NODES	SURVIVAL_STATUS
count	305.000000	305.000000	305.000000	305.000000
mean	52.531148	62.849180	4.036066	1.265574
std	10.744024	3.254078	7.199370	0.442364
min	30.000000	58.000000	0.000000	1.000000
25%	44.000000	60.000000	0.000000	1.000000
50%	52.000000	63.000000	1.000000	1.000000
75%	61.000000	66.000000	4.000000	2.000000
max	83.000000	69.000000	52.000000	2.000000

Observation

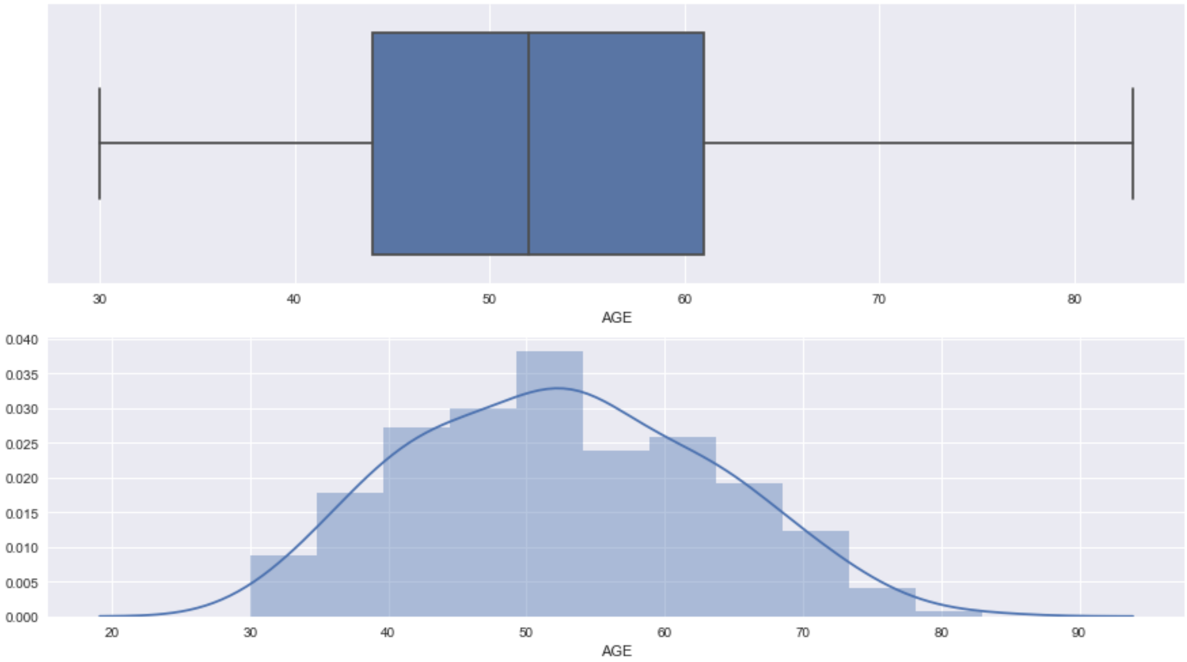
1. The dataset doesn't contains any NULL value.
2. We are getting the mean,standard deviation and quantiles for all the input variables.
3. Age of patient ranges from 30 to 83.
4. Positive axillary nodes ranges from 0 to 52.
5. 25% of patients are having 0 positive axillary nodes. Even though maximum no. of positive axillary is 52 but 75% of patients are having less than equal to 4 axillary nodes.

Univariate Analysis

1. AGE

In [124]:
```
f, (ax_box,ax_hist) = plt.subplots(2)
sns.set_style("whitegrid")
sns.set(rc={'figure.figsize':(15,8.27)})
sns.boxplot(dt["AGE"],ax = ax_box)
sns.distplot(dt["AGE"],ax = ax_hist)
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: Use
rWarning: The 'normed' kwarg is deprecated, and has been replaced by the 'den
sity' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```



Observation:

1.The AGE column data is normally distributed.

2.There are no outliers in the AGE column.

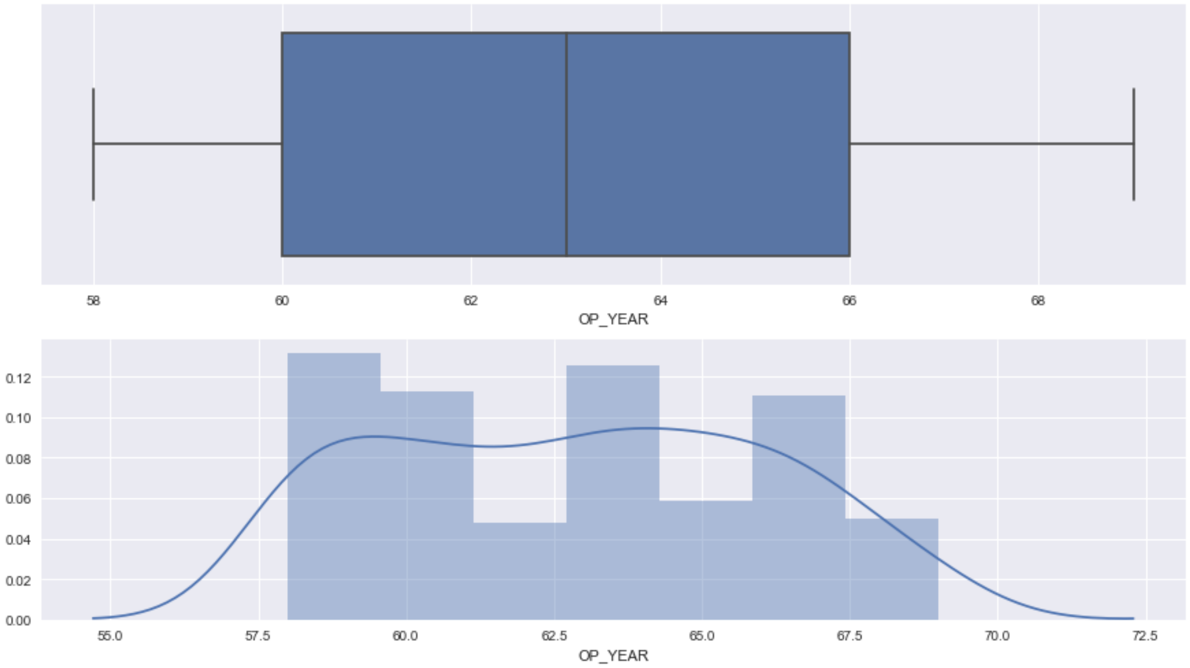
3.Mean is equal to Median for AGE column.

2. OP_YEAR

In [125]:

```
f, (ax_box,ax_hist) = plt.subplots(2)
sns.set_style("whitegrid")
sns.set(rc={'figure.figsize':(15,8.27)})
sns.boxplot(dt["OP_YEAR"],ax = ax_box)
sns.distplot(dt["OP_YEAR"],ax = ax_hist)
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: Use
rWarning: The 'normed' kwarg is deprecated, and has been replaced by the 'den
sity' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```



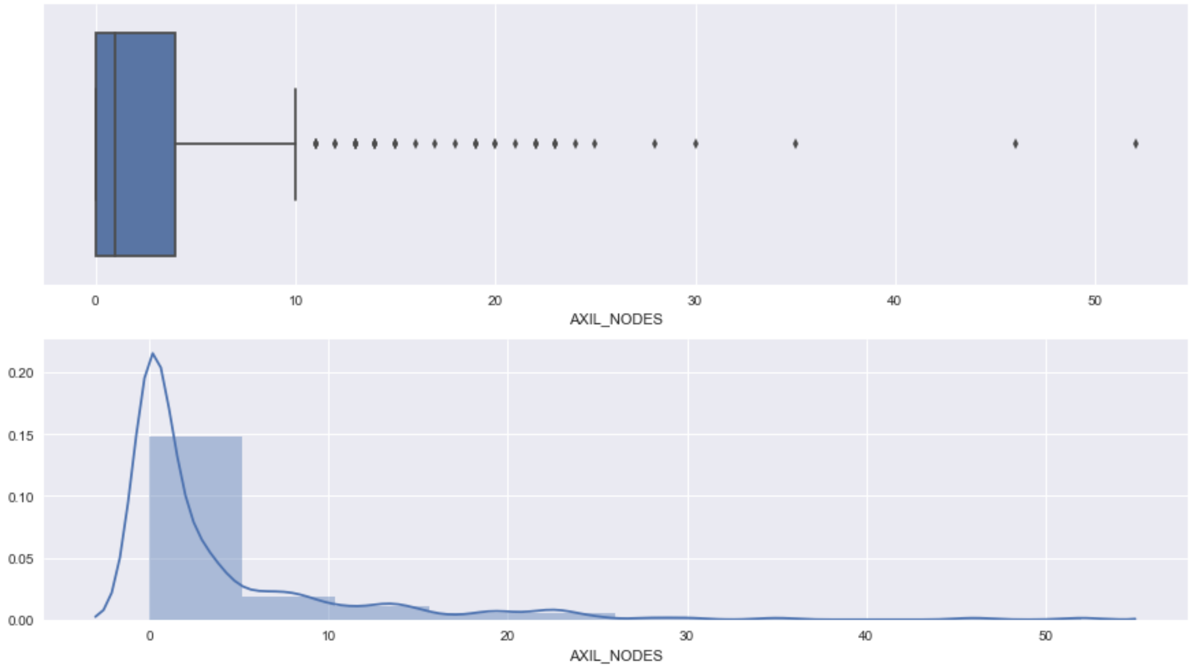
Observations:

1. Mean is almost equal to median.
2. No outliers in OP_YEAR column.
3. Data is normally distributed.

3. AXIL_NODES

```
In [126]: f, (ax_box,ax_hist) = plt.subplots(2)
          sns.set_style("whitegrid")
          sns.set(rc={'figure.figsize':(15,8.27)})
          sns.boxplot(dt["AXIL_NODES"],ax = ax_box)
          sns.distplot(dt["AXIL_NODES"],ax = ax_hist,bins = 10)
          plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462: Use
rWarning: The 'normed' kwarg is deprecated, and has been replaced by the 'den
sity' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```



Observations:

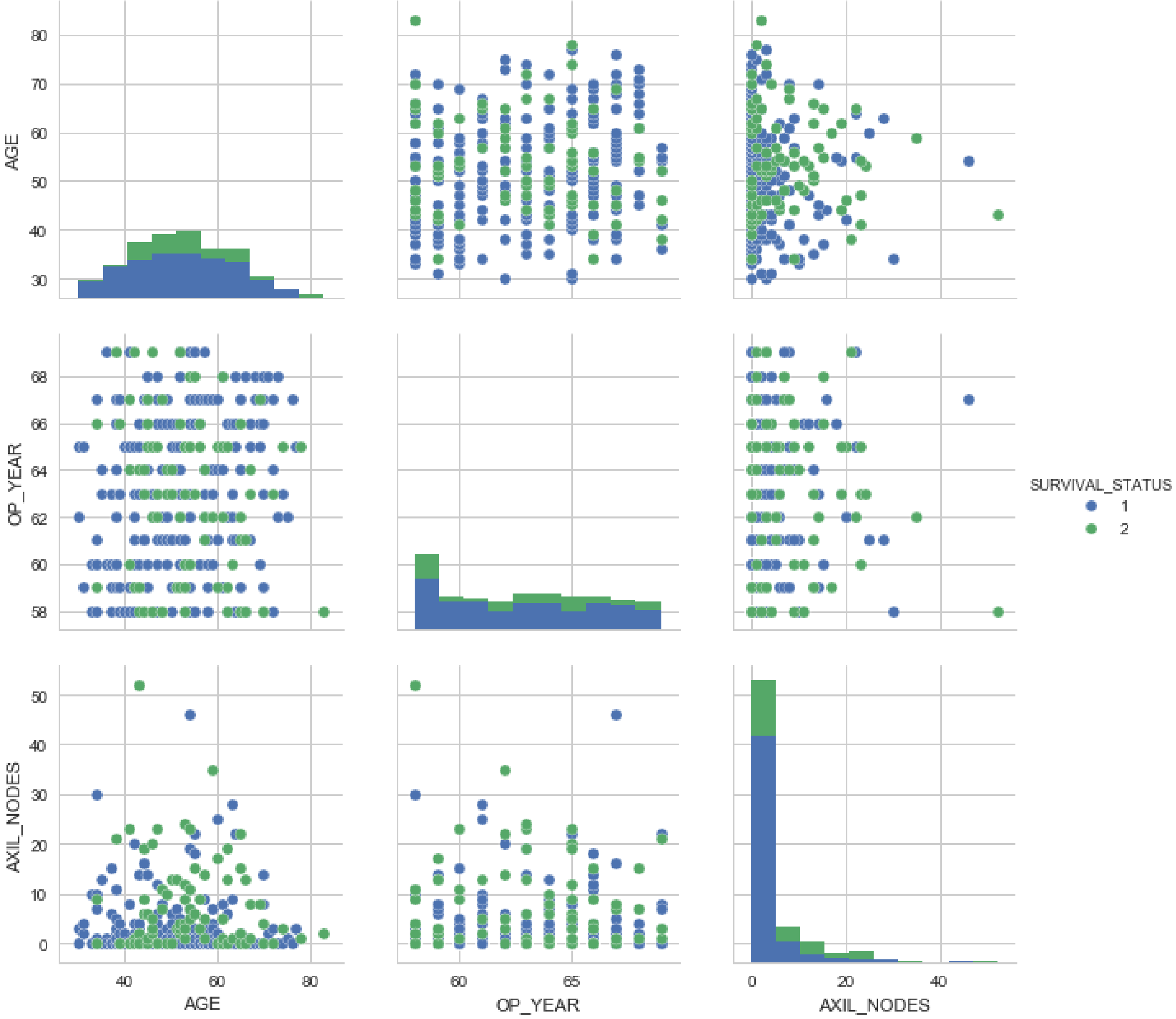
1. The column AXIL_NODES data is right skewed.
2. Outliers are present in AXIL_NODES columns

Bivariate Analysis

Plotting pair plot.

```
In [127]: #Plotting scatter plot
          sns.set_style("whitegrid")

          sns.pairplot(dt,hue='SURVIVAL_STATUS',vars=[dt.columns[0],dt.columns[1],dt.columns[2]],size=3)
```

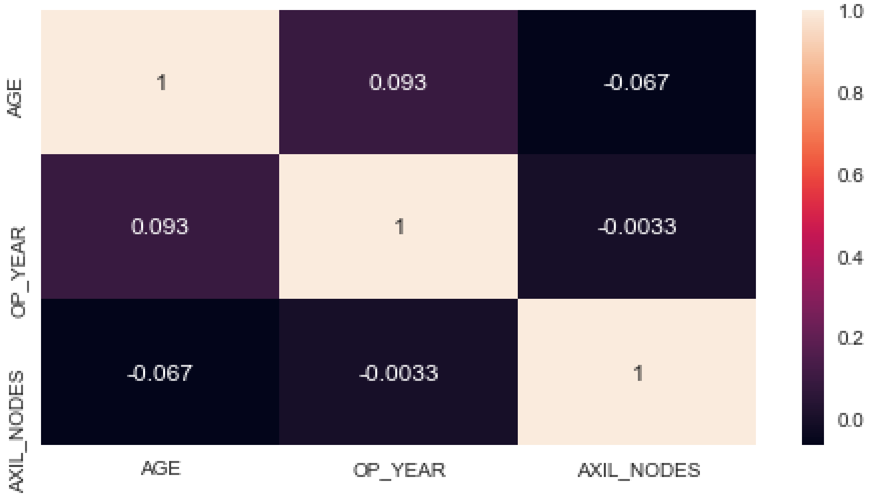
```
Out[127]: <seaborn.axisgrid.PairGrid at 0x14c2d5077f0>
```



Observation:

Here we can clearly see that two classes are linearly inseperable.

```
In [128]: #Checking corelation between all the features.
          plt.figure(figsize=(8,4))

          corr_values = dt[['AGE','OP_YEAR','AXIL_NODES']].corr()
          #print(corr_values)
          sns.heatmap(corr_values,
                      xticklabels = corr_values.columns.values,
                      yticklabels = corr_values.columns.values,
                      annot = True
                     )
          plt.show()
```



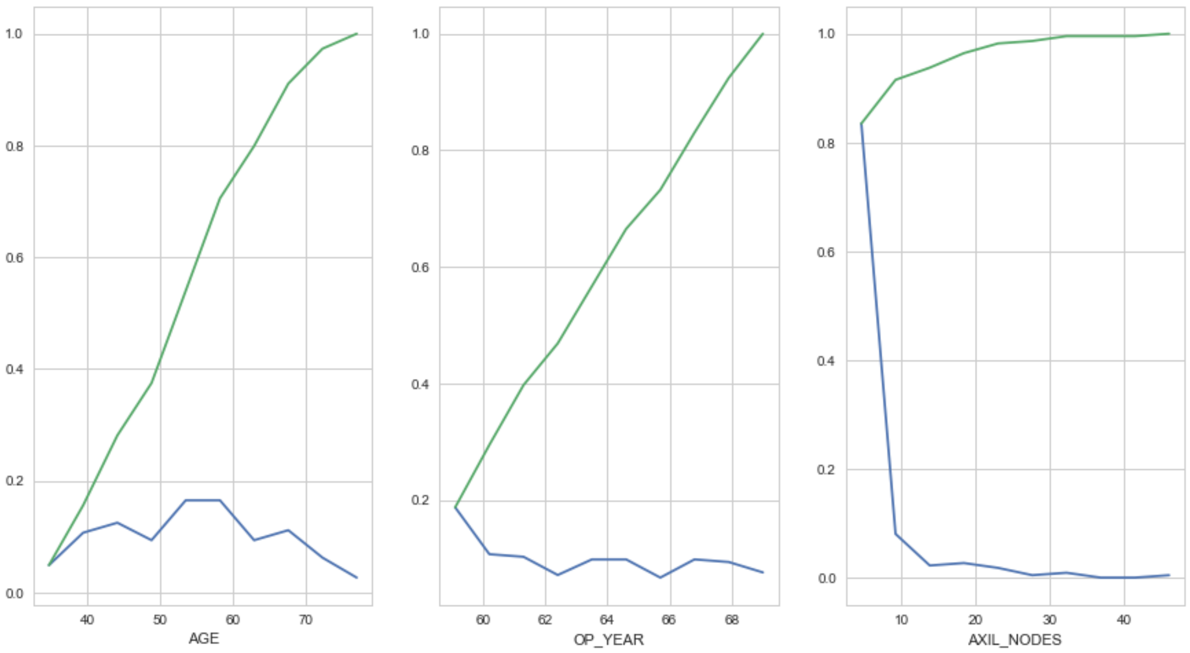
Observations:

The above plot shows that all the features are indpendent. There are no corelation between them.

In [129]:

```
#Plotting CDF and PDF of the survivors

Survival = dt[dt['SURVIVAL_STATUS']==1]
plt.figure(figsize=(15,8))
sns.set_style("whitegrid")
for i,attr in enumerate(list(dt.columns)[:-1]):
    plt.subplot(1,3,i+1)
    print("----------------------------------",attr,"-------------------------
--------\n")
    counts,bin_edges = np.histogram(Survival[attr],bins=10,density=True)
    print("Bin_Edges:- ",bin_edges)
    print('\n')
    pdf = counts/sum(counts)
    print("PDF:- ",pdf)
    print('\n')
    cdf = np.cumsum(pdf)
    print("CDF:- ",cdf)
    print('\n')
    plt.plot(bin_edges[1:],pdf)
    plt.plot(bin_edges[1:],cdf)
    plt.xlabel(attr)
```

```
--------------------------------- AGE ---------------------------------

Bin_Edges:-  [30.  34.7 39.4 44.1 48.8 53.5 58.2 62.9 67.6 72.3 77. ]


PDF:-  [0.04910714 0.10714286 0.125      0.09375    0.16517857 0.16517857
 0.09375    0.11160714 0.0625     0.02678571]


CDF:-  [0.04910714 0.15625    0.28125    0.375      0.54017857 0.70535714
 0.79910714 0.91071429 0.97321429 1.        ]


--------------------------------- OP_YEAR ---------------------------------

Bin_Edges:-  [58.  59.1 60.2 61.3 62.4 63.5 64.6 65.7 66.8 67.9 69. ]


PDF:-  [0.1875     0.10714286 0.10267857 0.07142857 0.09821429 0.09821429
 0.06696429 0.09821429 0.09375    0.07589286]


CDF:-  [0.1875     0.29464286 0.39732143 0.46875    0.56696429 0.66517857
 0.73214286 0.83035714 0.92410714 1.        ]


--------------------------------- AXIL_NODES -------------------------------
--

Bin_Edges:-  [ 0.   4.6  9.2 13.8 18.4 23.  27.6 32.2 36.8 41.4 46. ]


PDF:-  [0.83482143 0.08035714 0.02232143 0.02678571 0.01785714 0.00446429
 0.00892857 0.         0.         0.00446429]


CDF:-  [0.83482143 0.91517857 0.9375     0.96428571 0.98214286 0.98660714
 0.99553571 0.99553571 0.99553571 1.        ]
```

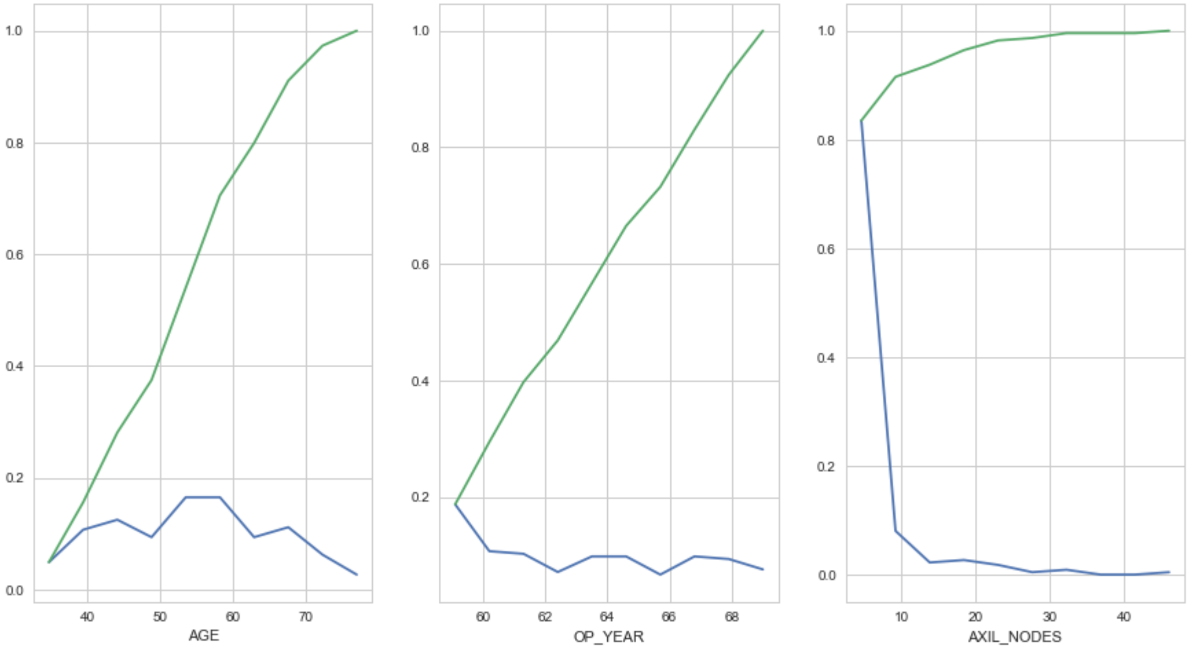


Observations:

```
Around 83% of the survivors have positive AXIL_NODES less than 5.
```

```
In [130]: #Plotting CDF and PDF of the survivors

          died = dt[dt['SURVIVAL_STATUS']==2]
          plt.figure(figsize=(15,8))
          sns.set_style("whitegrid")
          for i,attr in enumerate(list(dt.columns)[:-1]):
              plt.subplot(1,3,i+1)
              print("----------------------------------",attr,"-------------------------
          --------\n")
              counts,bin_edges = np.histogram(Survival[attr],bins=10,density=True)
              print("Bin_Edges:- ",bin_edges)
              print('\n')
              pdf = counts/sum(counts)
              print("PDF:- ",pdf)
              print('\n')
              cdf = np.cumsum(pdf)
              print("CDF:- ",cdf)
              print('\n')
              plt.plot(bin_edges[1:],pdf)
              plt.plot(bin_edges[1:],cdf)
              plt.xlabel(attr)
```

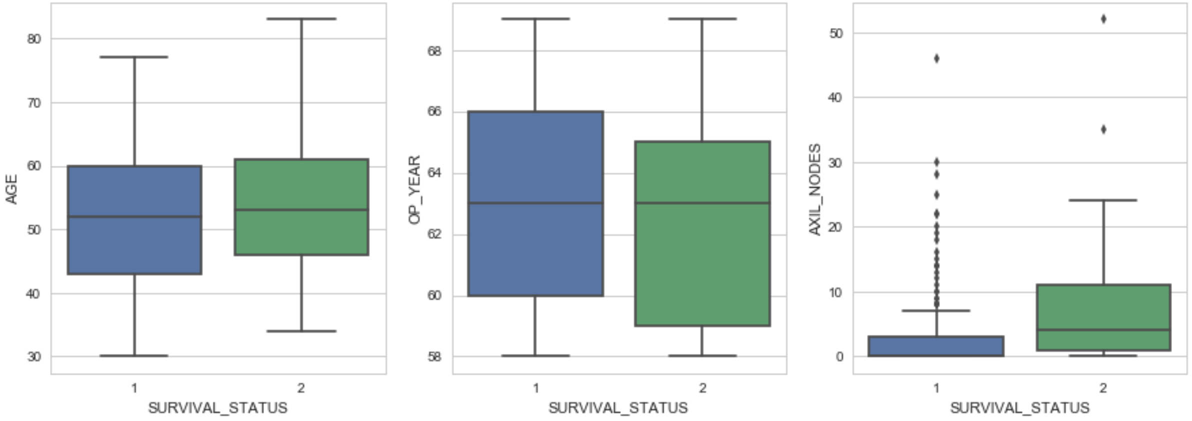
```
---------------------------------- AGE ---------------------------------

Bin_Edges:-  [30.  34.7 39.4 44.1 48.8 53.5 58.2 62.9 67.6 72.3 77. ]


PDF:-  [0.04910714 0.10714286 0.125      0.09375    0.16517857 0.16517857
 0.09375    0.11160714 0.0625     0.02678571]


CDF:-  [0.04910714 0.15625    0.28125    0.375      0.54017857 0.70535714
 0.79910714 0.91071429 0.97321429 1.        ]


---------------------------------- OP_YEAR ---------------------------------

Bin_Edges:-  [58.  59.1 60.2 61.3 62.4 63.5 64.6 65.7 66.8 67.9 69. ]


PDF:-  [0.1875     0.10714286 0.10267857 0.07142857 0.09821429 0.09821429
 0.06696429 0.09821429 0.09375    0.07589286]


CDF:-  [0.1875     0.29464286 0.39732143 0.46875    0.56696429 0.66517857
 0.73214286 0.83035714 0.92410714 1.        ]


---------------------------------- AXIL_NODES -------------------------------
--

Bin_Edges:-  [ 0.   4.6  9.2 13.8 18.4 23.  27.6 32.2 36.8 41.4 46. ]


PDF:-  [0.83482143 0.08035714 0.02232143 0.02678571 0.01785714 0.00446429
 0.00892857 0.         0.         0.00446429]


CDF:-  [0.83482143 0.91517857 0.9375     0.96428571 0.98214286 0.98660714
 0.99553571 0.99553571 0.99553571 1.        ]
```



Observations:-

Around 97% of those who died were having AXIL_NODES less than equal to 26.

In [131]:
```
#Box plot
fig, axes = plt.subplots(1, 3, figsize=(15, 5))
for i,attr in enumerate(list(dt.columns)[:-1]):
    sns.boxplot(x='SURVIVAL_STATUS',y=attr,data=dt, ax=axes[i])
plt.show()
```



In [132]:
```
#Calculating 25th,50th and 75th percentile of the survivors and those who died
 w.r.t to axillary nodes
print(np.percentile(Survival['AXIL_NODES'],(25,50,75)))
print(np.percentile(died['AXIL_NODES'],(25,50,75)))
```

```
[0. 0. 3.]
[ 1.  4. 11.]
```

Observations:

1. Person who survived having AGE between 40 to 43 && OP_YEAR between 65 to 66 and AXIL_NODES between 0 to 4.