Haberman's Survival Exploratory Data Analysis


```
Created By: Rahul Dable
Dataset: Haberman Cancer Survival dataset from Kaggle
```


Exercise:

1. Download Haberman Cancer Survival dataset from Kaggle. You may have to create a Kaggle account to donwload data. (https://www.kaggle.com/gilsousa/habermans-survival-data-set)
2. Perform a similar alanlaysis as above on this dataset with the following sections:

- High level statistics of the dataset: number of points, numer of features, number of classes, data-points per class.
- Explain our objective.
- Perform Univaraite analysis(PDF, CDF, Boxplot, Voilin plots) to understand which features are useful towards classification.
- Perform Bi-variate analysis (scatter plots, pair-plots) to see if combinations of features are useful in classfication.
- Write your observations in english as crisply and unambigously as possible. Always quantify your results.

CODE REFERENCE: Exploratory Data Analysis.ipynb notebook provided by **APPLIED AI COURSE**.

Importing Required Libraries

In [1]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
%matplotlib inline
```

1. Loading and Analysing Dataset

haberman.csv

In [3]:
```
haberman_data = pd.read_csv('haberman.csv')
haberman_data.head()
```

Out[3]:

	age	year	nodes	status
0	30	64	1	1
1	30	62	3	1
2	30	65	0	1
3	31	59	2	1
4	31	65	4	1

In [4]:
```
haberman_data.columns
```

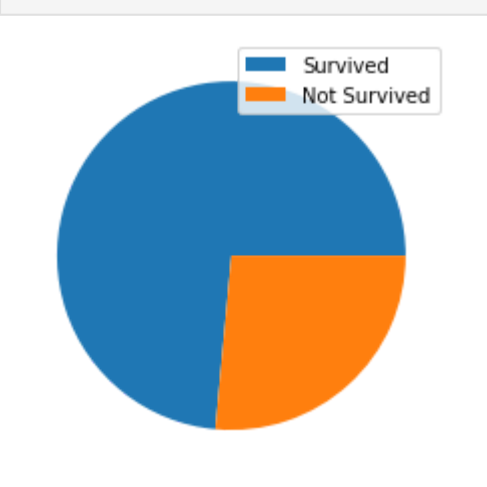
Out[4]:
```
Index(['age', 'year', 'nodes', 'status'], dtype='object')
```

Haberman's data contains total 4 coulmns.

```
1. Age if the patients
2. Year of surgery
3. Positive axillary nodes detected
4. Patient's survival status - Class Lable
        1 = Patient has survived 5 or more years
        2 = Patient has died within 5 years
```

Checking if the data is balanced

In [5]:
```
plt.pie(haberman_data.status.value_counts())
plt.legend(["Survived","Not Survived"])
plt.show()

print(haberman_data['status'].value_counts())
```




```
1    225
2     81
Name: status, dtype: int64
```

Conclusion

1. class 1 - i.e. number of patient survived = 225
2. class 2 - i.e. number of patient died = 81
3. The haberman's dataset is **imbalanced** as there is a huge difference in number of patient belong to each classlables.
4. 73% of the patients who had undergone the surgery had survived more than 5 years after surgery.

Checking for null values

In [6]:
```
haberman_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 306 entries, 0 to 305
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   age     306 non-null    int64
 1   year    306 non-null    int64
 2   nodes   306 non-null    int64
 3   status  306 non-null    int64
dtypes: int64(4)
memory usage: 9.7 KB
```

Zero Null cells found.

High level statistics

In [7]:
```
haberman_data.describe()
```

Out[7]:

	age	year	nodes	status
count	306.000000	306.000000	306.000000	306.000000
mean	52.457516	62.852941	4.026144	1.264706
std	10.803452	3.249405	7.189654	0.441899
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%	60.750000	65.750000	4.000000	2.000000
max	83.000000	69.000000	52.000000	2.000000

Conclusion

1. Min number of nodes detected in amy patient is 0 while max number of nodes is 52.
2. 50% patients had number of nodes =1 and 75% of the patients hax max 4 nodes.

In [8]:
```
haberman_data[haberman_data.nodes==0].status.value_counts(1)
```

Out[8]:
```
1    0.860294
2    0.139706
Name: status, dtype: float64
```

Conclusion:

1. 86% patients with zero nodes have survived more than 5 years.

EDA Objective

Our objective of performing exploratory data analysis is to find if there is any relationship between age, year of operation and number of detected nodes in a cancer patient to its survival status.

2.UNIVARIATE ANALYSIS

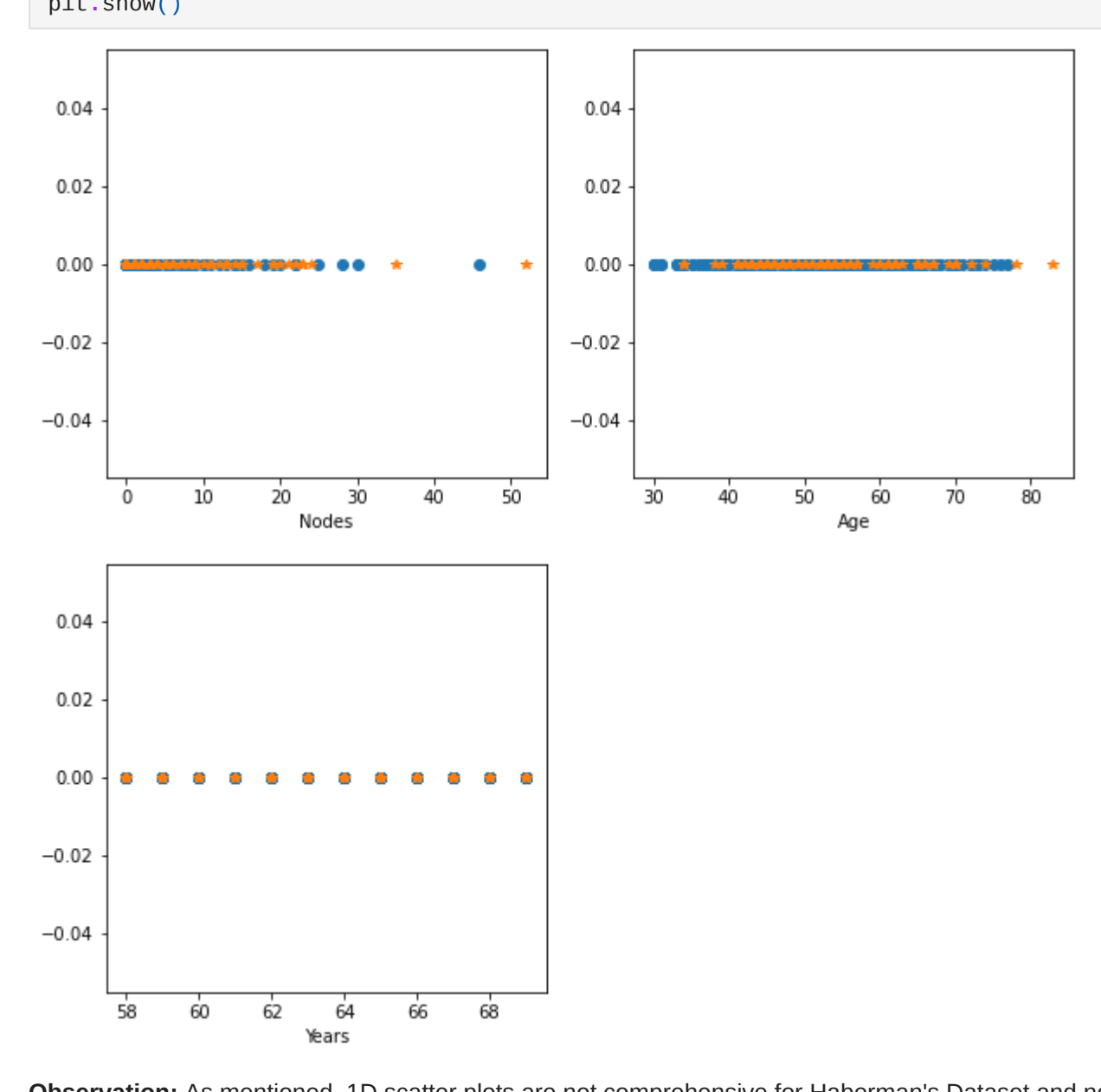
We can create 1D scatter plots for each features using single feature at a time, however they are less comprehensive and provides limited informations therefore not preferable in many cases.

2.1 1D SCATTER PLOTS

As we are plotting 1D plot, which deals with only one feature at a time, let's select one feature i.e. age, year or number of nodes from all class labels i.e. from class 1 and class2.

In [9]:
```
patient_survived = haberman_data.loc[haberman_data["status"]==1];
patient_died = haberman_data.loc[haberman_data["status"]==2]
```

In [11]:
```
plt.figure(figsize=(10,10))

plt.subplot(221)
plt.plot(patient_survived['nodes'], np.zeros_like(patient_survived['nodes']), 'o')
plt.plot(patient_died['nodes'], np.zeros_like(patient_died['nodes']), '*')
plt.xlabel('Nodes')

plt.subplot(222)
plt.plot(patient_survived['age'], np.zeros_like(patient_survived['age']), 'o')
plt.plot(patient_died['age'], np.zeros_like(patient_died['age']), '*')
plt.xlabel('Age')

plt.subplot(223)
plt.plot(patient_survived['year'], np.zeros_like(patient_survived['year']), 'o')
plt.plot(patient_died['year'], np.zeros_like(patient_died['year']), '*')
plt.xlabel('Years')
plt.show()
```

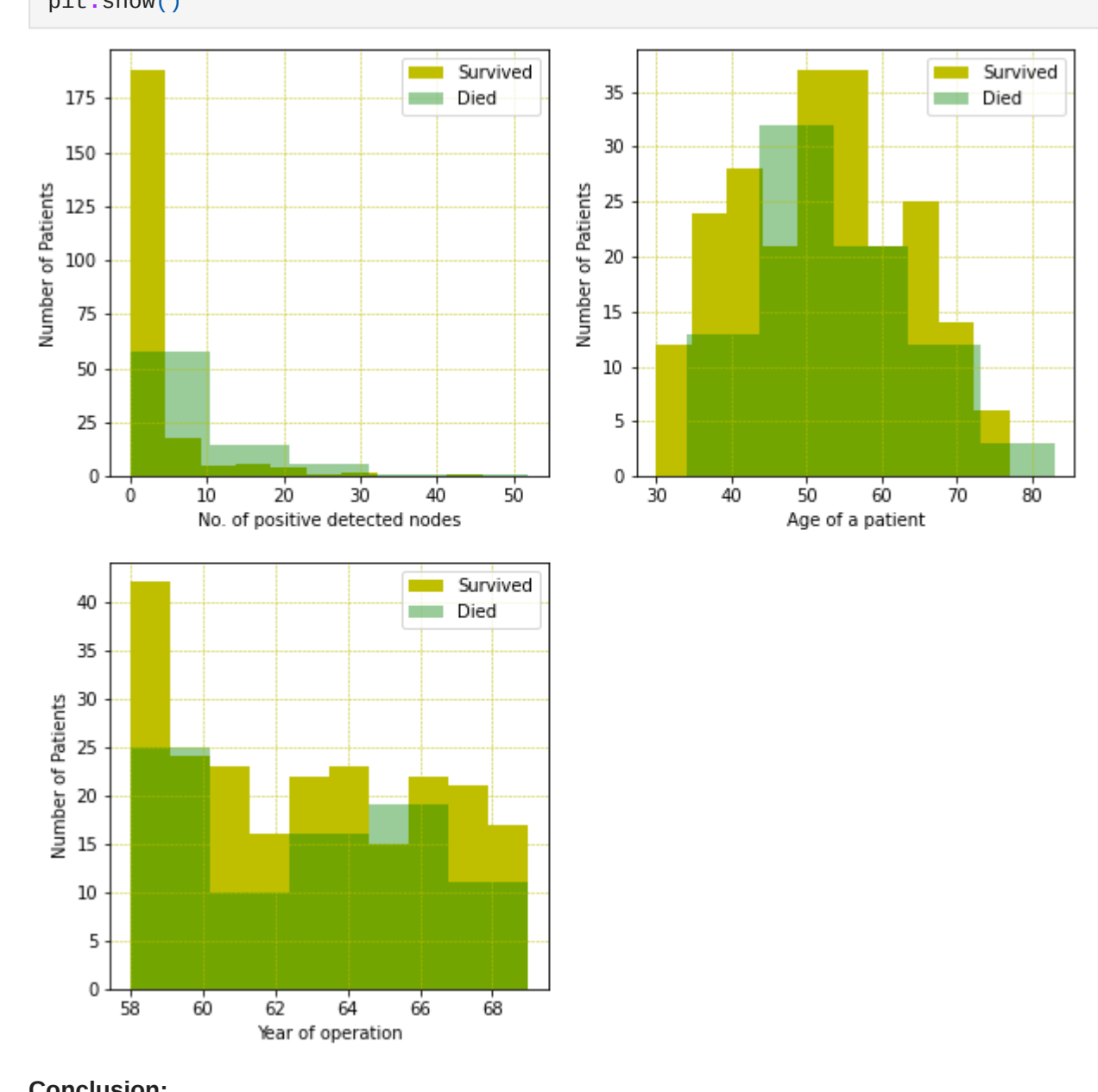


Observation: As mentioned, 1D scatter plots are not comprehensive for Haberman's Dataset and no conclusion can be drawn from them.

2.2 HISTOGRAMS

Another way of plotting are histograms which, at any given points, indicates the quantity/density of the corresponding datapoints.

In [17]:
```
plt.figure(figsize=(10,10))

plt.subplot(221)
plt.hist(patient_survived['nodes'], bins=10, color='y', alpha=1)
plt.hist(patient_died['nodes'], bins=5, color='g', alpha=0.4)
plt.xlabel('No. of positive detected nodes')
plt.ylabel('Number of Patients')
labels = ["Survived", "Died"]
plt.grid(color='y', linestyle='--', linewidth=0.5)
plt.legend(labels)

plt.subplot(222)
plt.hist(patient_survived['age'], bins=10, color='y')
plt.hist(patient_died['age'], bins=5, color='g', alpha=0.4)
plt.xlabel('Age of a patient')
plt.ylabel('Number of Patients')
labels = ["Survived", "Died"]
plt.grid(color='y', linestyle='--', linewidth=0.5)
plt.legend(labels)

plt.subplot(223)
plt.hist(patient_survived['year'], bins=10, color='y')
plt.hist(patient_died['year'], bins=5, color='g', alpha=0.4)
plt.xlabel('Year of operation')
plt.ylabel('Number of Patients')
labels = ["Survived", "Died"]
plt.grid(color='y', linestyle='--', linewidth=0.5)
plt.legend(labels)

plt.show()
```



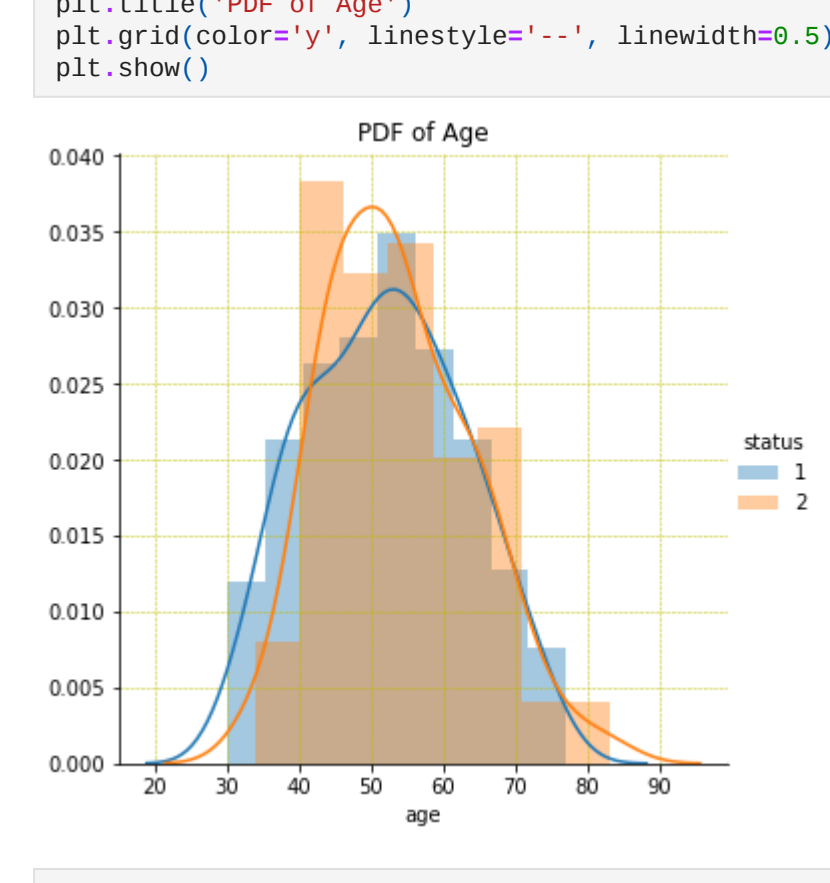
Conclusion:

```
1. From Histogram of Nodes, It can be seen than the number of patients with lesser amount of "Nodes" has better chance        of survival.
2. Nodes seems to be relatively more important feature than rest.
```

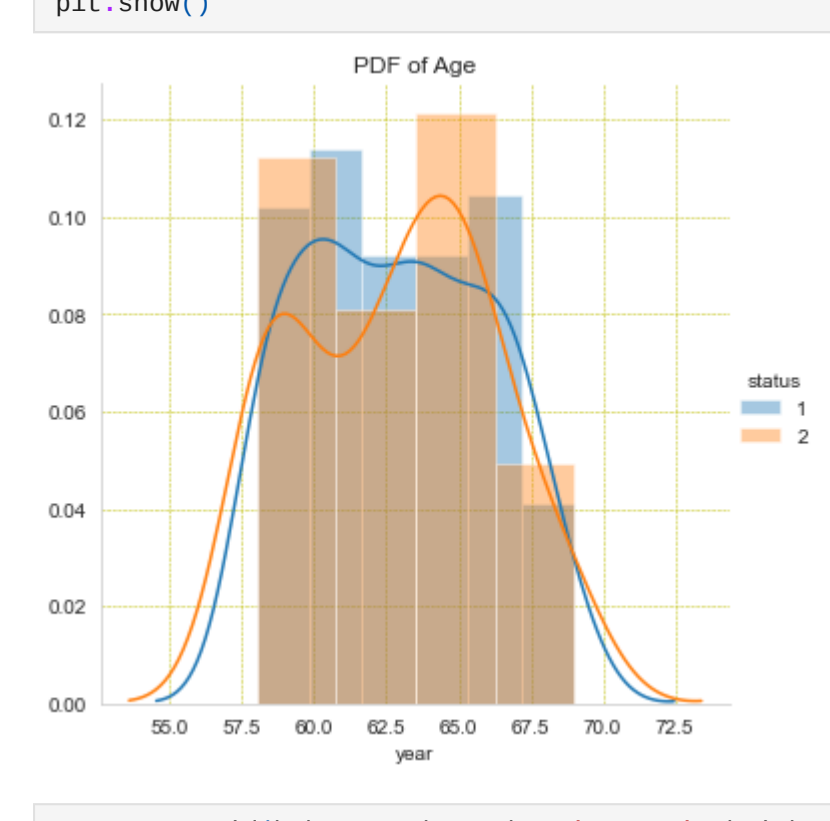
2.3 PDF

Probability Density Functions are basically the smooth approximation of histograms.

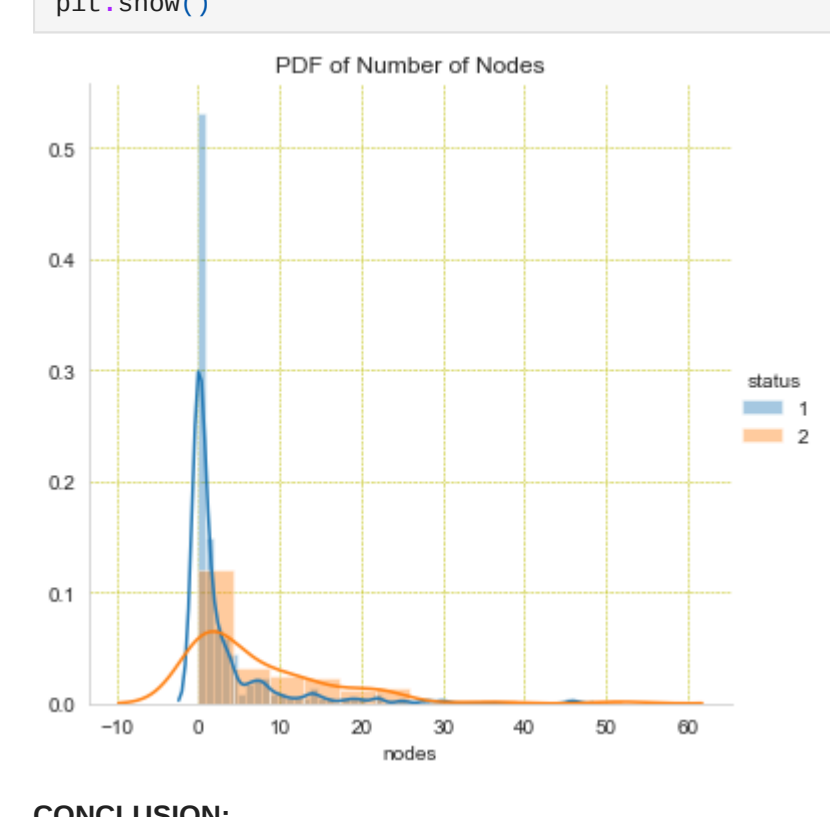
In [18]:
```
sns.FacetGrid(haberman_data, hue='status', height=5).map(sns.distplot, 'age').add_legend()
plt.title('PDF of Age')
plt.grid(color='y', linestyle='--', linewidth=0.5)
plt.show()
```



In [124…
```
sns.FacetGrid(haberman_data, hue='status', height=5).map(sns.distplot, 'year').add_legend()
plt.grid(color='y', linestyle='--', linewidth=0.5)
plt.title('PDF of Age')
plt.show()
```



In [123…
```
sns.FacetGrid(haberman_data, hue='status', height=5).map(sns.distplot, 'nodes').add_legend()
plt.grid(color='y', linestyle='--', linewidth=0.5)
plt.title('PDF of Number of Nodes')

plt.show()
```



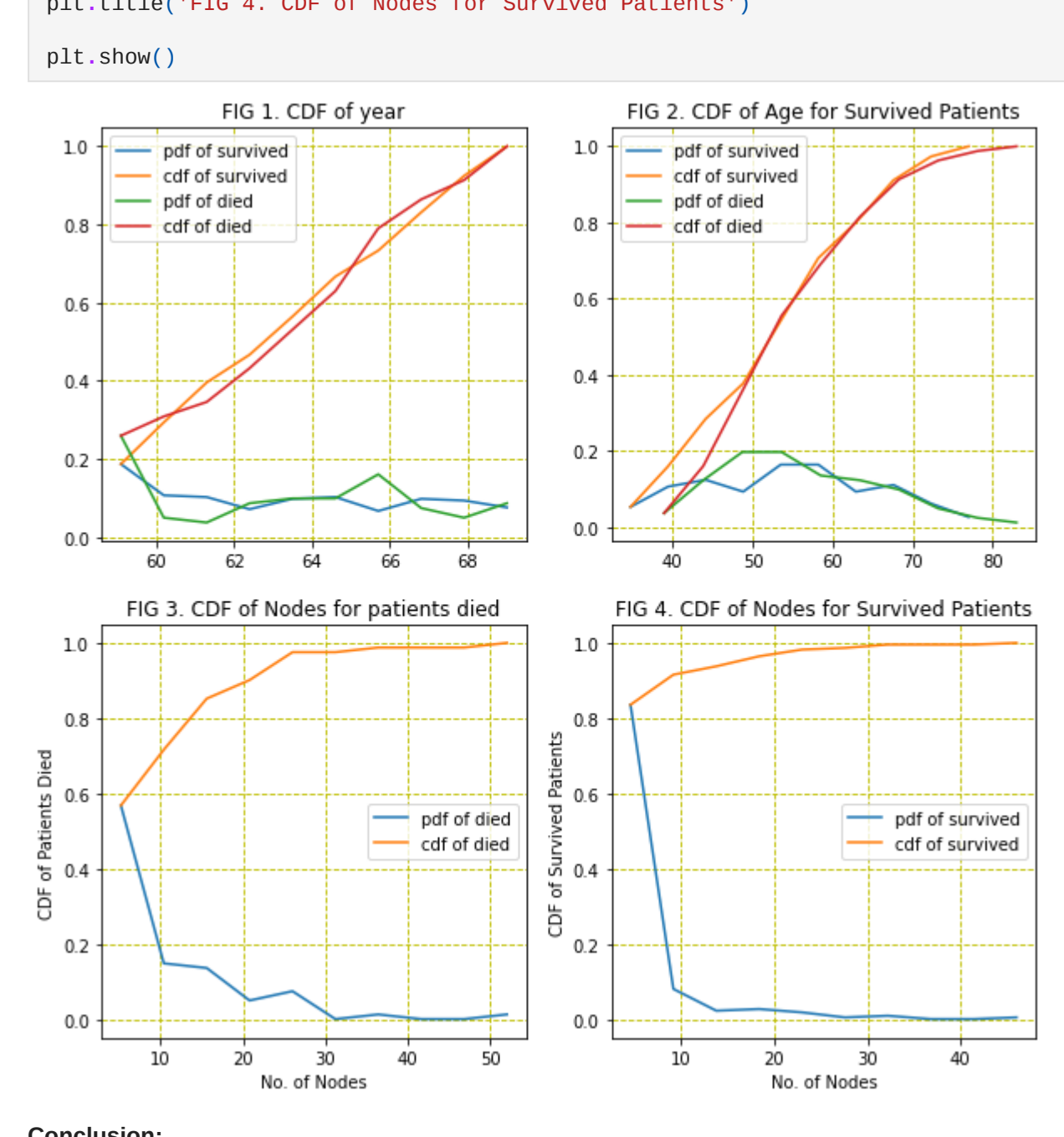
CONCLUSION:

```
1.There is massive overlap in distribution of pdf when we use age as a feature.

2.From PDF of No of detected nodes in a cancer patients, it is observed that the patients with lesser number of Number of Axillary
nodes(Lymph Nodes) has greater chance of survival.
```

2.4 CUMULATIVE DENSITY FUNCTION (CDF)

- CDF is basically a cumulative sum of all the points until that point (inclusive).
- CDF at any point tells us how much percentage of data pooints lies until that point.
- So, CDF is "Area Under The Curve (AUC) " of PDFs.

In [19]:
```
plt.figure(figsize=(10,10))

plt.subplot(221)
label = ["pdf of survived", "cdf of survived", "pdf of died", "cdf of died"]
counts, bin_edges = np.histogram(patient_survived['year'], bins=10, density= False);
pdf= counts / (np.sum(counts))
cdf= np.cumsum(pdf)
plt.plot(bin_edges[1:], pdf)
plt.plot(bin_edges[1:], cdf)
counts, bin_edges = np.histogram(patient_died['year'], bins=10, density= False);
pdf = counts / (np.sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:], pdf)
plt.plot(bin_edges[1:], cdf)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('FIG 1. CDF of year')
plt.legend(label)

plt.subplot(222)
label = ["pdf of survived", "cdf of survived", "pdf of died", "cdf of died"]
counts, bin_edges = np.histogram(patient_survived['age'], bins=10, density= False);
pdf = counts / (np.sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:], pdf)
plt.plot(bin_edges[1:], cdf)
counts, bin_edges = np.histogram(patient_died['age'], bins=10, density= False);
pdf = counts / (np.sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:], pdf)
plt.plot(bin_edges[1:], cdf)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('FIG 2. CDF of Age for Survived Patients')
plt.legend(label)

plt.subplot(223)
label = ["pdf of died", "cdf of died"]
counts, bin_edges = np.histogram(patient_died['nodes'], bins=10, density= False);
pdf = counts / (np.sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:], pdf)
plt.plot(bin_edges[1:], cdf)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.xlabel('No. of Nodes')
plt.ylabel('CDF of Patients Died')
plt.legend(label)
plt.title('FIG 3. CDF of Nodes for patients died')

plt.subplot(224)
label = ["pdf of survived", "cdf of survived"]
counts, bin_edges = np.histogram(patient_survived['nodes'], bins=10, density= False);
pdf = counts / (np.sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:], pdf)
plt.plot(bin_edges[1:], cdf)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.xlabel('No. of Nodes')
plt.ylabel('CDF of Survived Patients')
plt.legend(label)
plt.title('FIG 4. CDF of Nodes for Survived Patients')

plt.show()
```

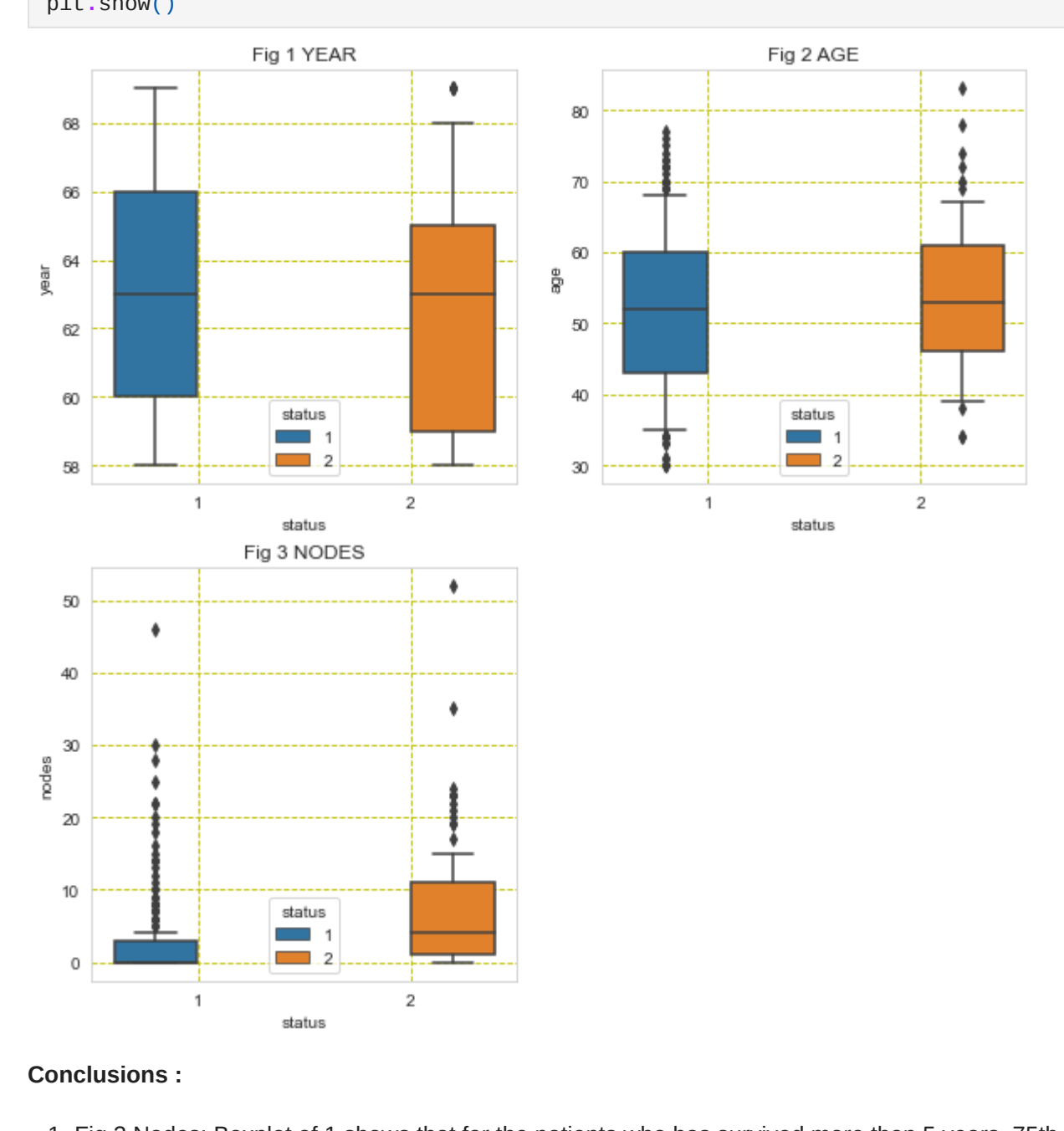


Conclusion:

1. From CDF of **Nodes** of **Survived Patients** (FIG. 4), it can be seen that the patients having less than 10 nodes has around 85% chance of more than 5 years survival.
2. PDF and CDF of features such as AGE and YEAR for both category of the patients appeares to be almost overlapping therefore doesn't provide any distinguishing information.
3. As number of nodes increases, chances of survival declined sharply.

2.5 BOX PLOT AND WHISKERS

In [118…
```
plt.figure(figsize=(10,10))

plt.subplot(221)
sns.boxplot(data = haberman_data, x='status', y='year', hue ='status', whis=0.5)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('Fig 1 YEAR')

plt.subplot(222)
sns.boxplot(data = haberman_data, x='status', y='age', hue ='status', whis=0.5)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('Fig 2 AGE')

plt.subplot(223)
sns.boxplot(data = haberman_data, x='status', y='nodes', hue ='status', whis=0.5)
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('Fig 3 NODES')

plt.show()
```

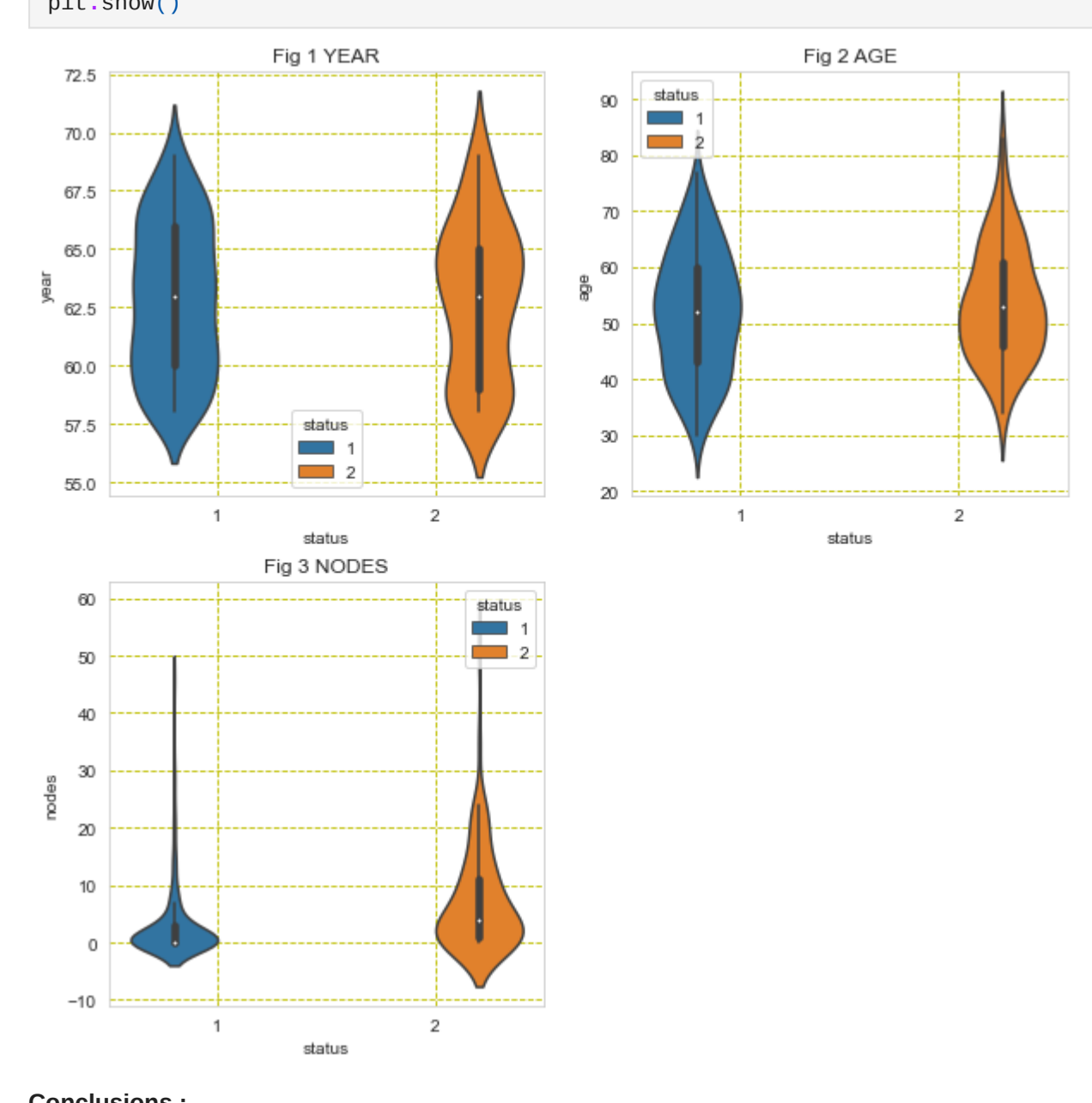


Conclusions :

1. Fig 3 Nodes: Boxplot of 1 shows that for the patients who has survived more than 5 years, 75th percentile value is at node 2 while for the patients who had died, 75th percentile value is at node 11.
2. only few patients had number of nodes more than 30, they are probably the outliers.
3. There are many points which lies outside the whiskers which might be representing outliers.

2.6 VIOLIN PLOTS:

In [116…
```
plt.figure(figsize=(10,10))

plt.subplot(221)
sns.violinplot(data=haberman_data, y='year', x='status', hue ='status')
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('Fig 1 YEAR')

plt.subplot(222)
sns.violinplot(data=haberman_data, y='age', x='status', hue ='status')
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('Fig 2 AGE')

plt.subplot(223)
sns.violinplot(data=haberman_data, y='nodes', x='status', hue ='status')
plt.grid(color='y', linestyle='--', linewidth=0.9)
plt.title('Fig 3 NODES')

plt.show()
```



Conclusions :

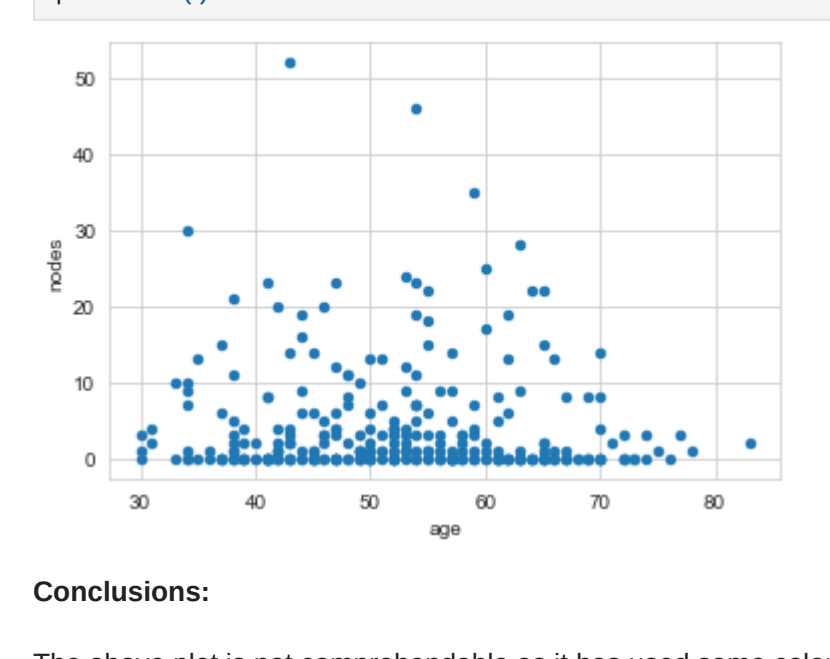
Fig 3 Nodes, It can be observed that patients having 0 nodes have greater chances of survival

---

3 BIVARIATE AND MULTIVARIATE ANALYSIS

3.1 2D SCATTER PLOTS

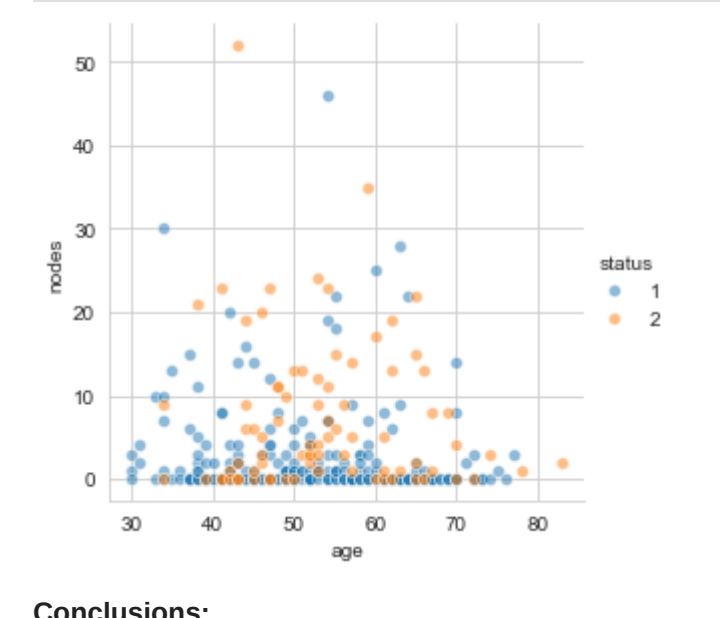
In [35]:
```
haberman_data.plot(kind= 'scatter', x ='age', y = 'nodes' )
plt.show()
```



Conclusions:

The above plot is not comprehendable as it has used same color for all three class labels. Lets try to make it more readable by setting diferent color for each class using seaborn.

In [105…
```
sns.set_style("whitegrid");
sns.FacetGrid(haberman_data, hue='status', height = 4).map(sns.scatterplot, 'age', 'nodes', alpha=0.5).add_legend();
plt.show()
```



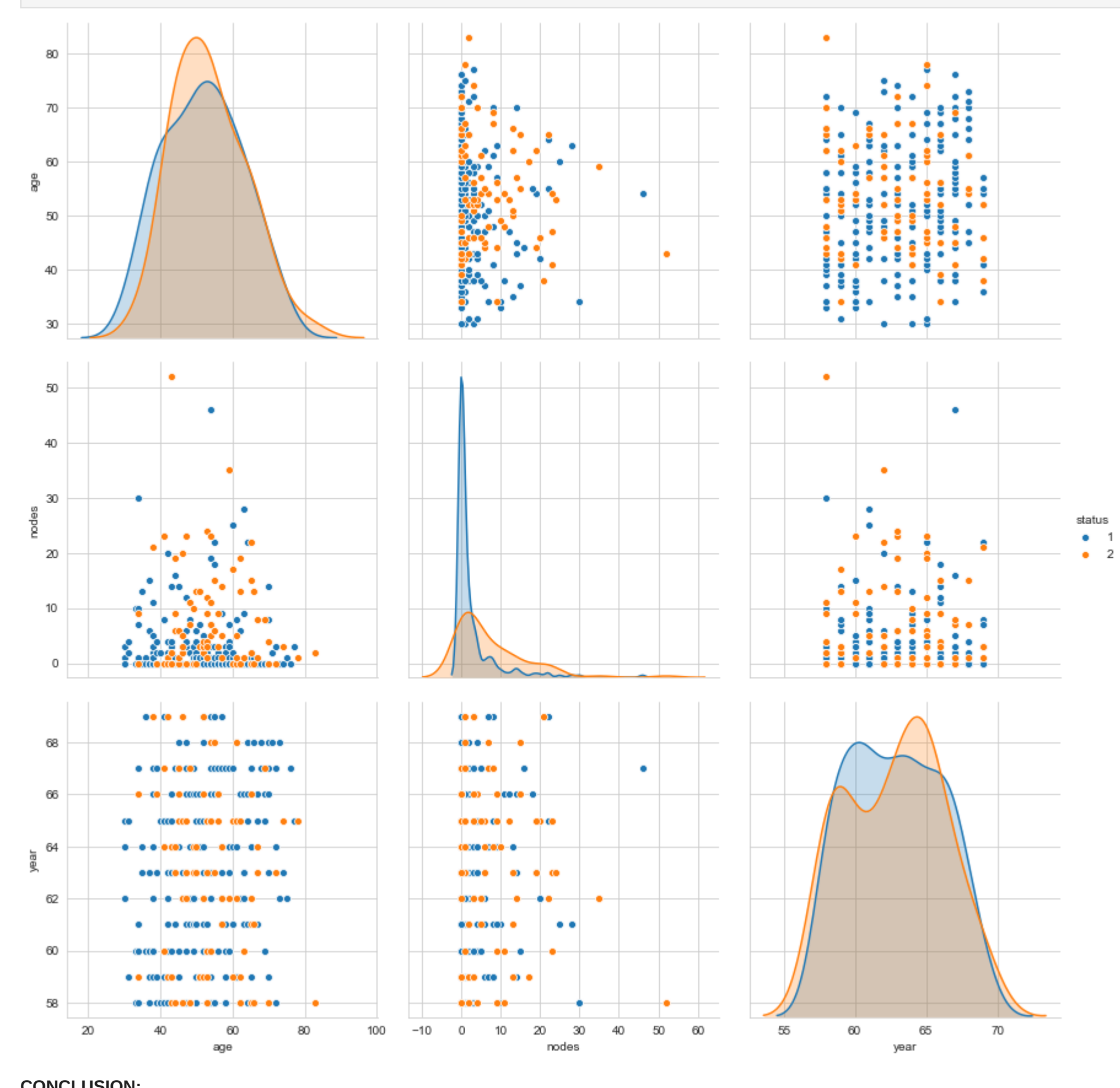
Conclusions:

As the datapoints are jumbled, it is difficulat to intrepret above figure.

3.2 PAIR PLOTS

Amongst 4 features from that dataset, lets choose three features (age, nodes and year) to plot pairplots and survival status will be indicated by hue.

In [107…
```
plt.close()
sns.set_style("whitegrid")
sns.pairplot(haberman_data, hue='status', kind='scatter',height = 4, vars=['age', 'nodes', 'year']);
plt.show()
```

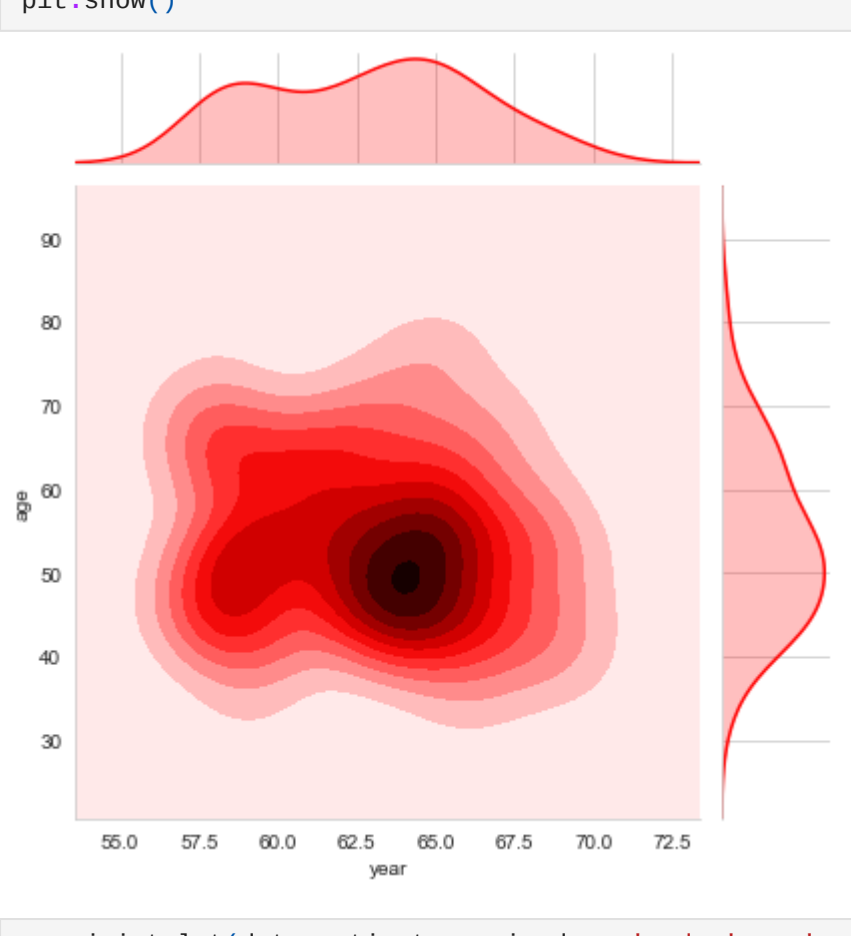


CONCLUSION:

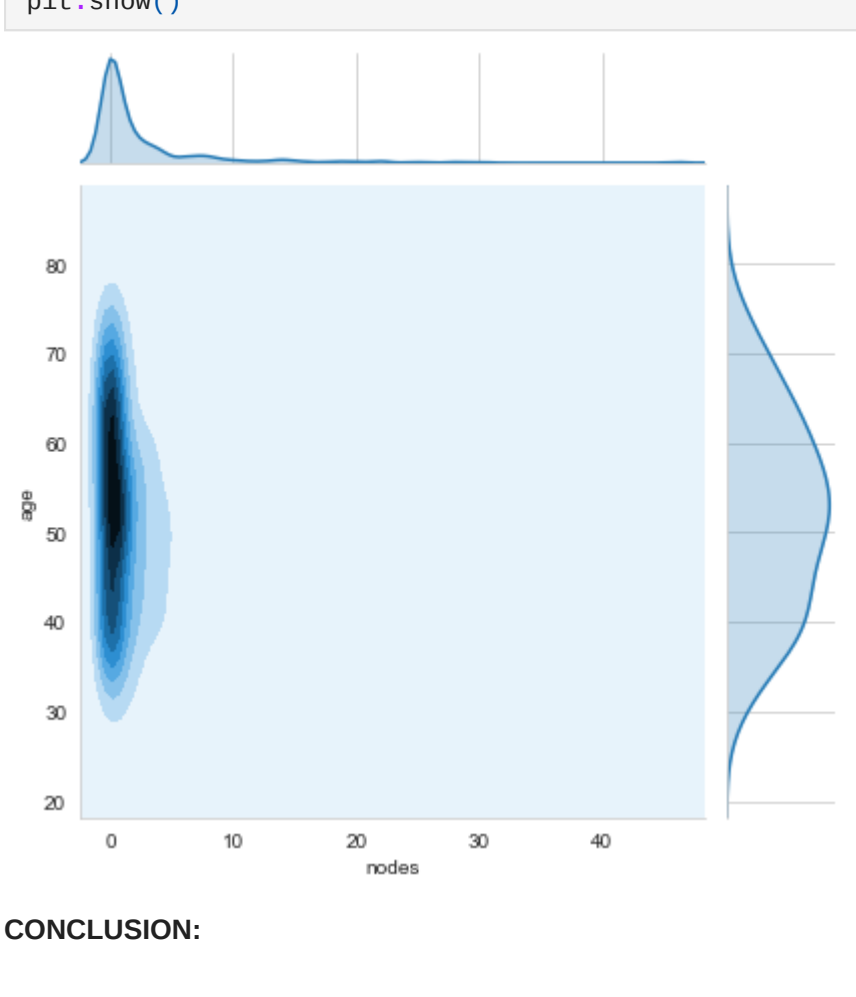
1. The data is not linearly seperable as most of the fatures are jumbled.
2. From the chances of survival of patients with zero/less nodes are comparatively higher.
3. NODES are seems to be comparatively most important feature amongst all the goven features.

3.3 Multivariate Probability Density, Contour plot

In [158…
```
sns.jointplot(data=patient_died, x='year', y='age', kind='kde', color='r')
plt.show()
```



In [242…
```
sns.jointplot(data=patient_survived, x='nodes', y='age', kind='kde')
plt.show()
```



CONCLUSION:

```
1. Chances of survival is higher for a patient with lesser number of nodes.
```