Plotting for Exploratory data analysis (EDA)

Haberman Dataset

Toy Dataset: Haberman Dataset :https://www.kaggle.com/gilsousa/habermans-survival-data-set/version/1 (https://www.kaggle.com/gilsousa/habermans-survival-data-set/version/1)

- Objective: To find the survival status of a patient given three varibles(i.e age,year(year of operation),nodes)

In [112]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import warnings
warnings.filterwarnings("ignore")

hd = pd.read_csv("haberman.csv")
```

In [113]:
```
# (Q) how many data-points and features?
print (hd.shape)
```

```
(306, 4)
```

In [114]:
```
#(Q) What are the column names in our dataset?
print (hd.columns)
```

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

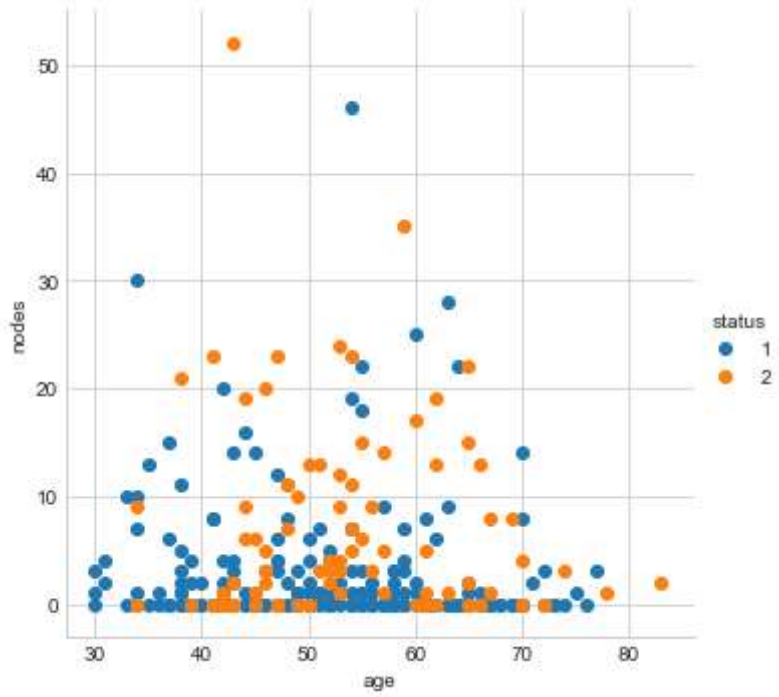
In [115]:
```
#(Q) How many data points for each class are present?
#There are two classes 1 and 2(in status column) .
#1 means patient survived more than 5 years.
#2 means patient survived less than 5 years.

hd["status"].value_counts()
# balanced-dataset vs imbalanced datasets
```

Out[115]:
```
1    225
2     81
Name: status, dtype: int64
```

2-D Scatter Plot

In [104]:
```
sns.set_style("whitegrid");
sns.FacetGrid(hd, hue="status", size=5) \
   .map(plt.scatter, "age", "nodes") \
   .add_legend();
plt.show();
```



Observation(s):

1. Using age and nodes variables it is not possible to distinguish the survival status.

Pair-plot

In [107]:
```
sns.pairplot(hd,hue="status",vars=["age","year","nodes"],size=5);
plt.show()
```

Observation(s)

1.It is not possible to identify the status using pairplots also.

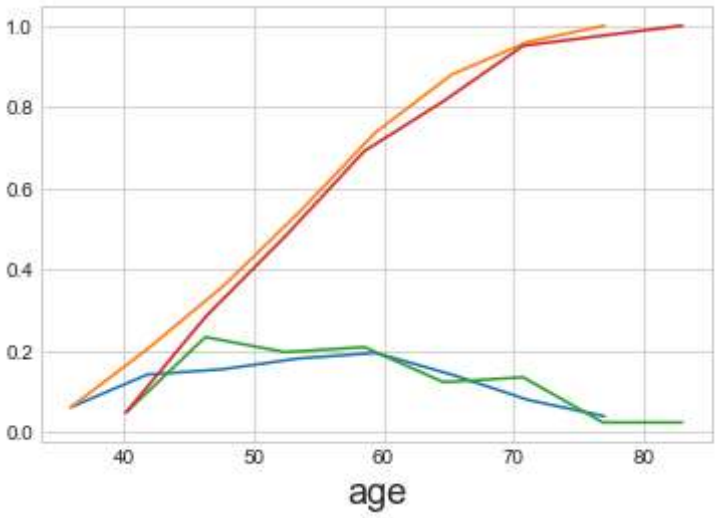
CDF

In [111]:
```
# age
x, y = np.histogram(hd_1['age'], bins=8,
                                    density = True)
pdf = x/(sum(x))
print(pdf);
print(y)
cdf = np.cumsum(pdf)
plt.plot(y[1:],pdf)
plt.xlabel('age', fontsize=18)
plt.plot(y[1:], cdf)


# 2
x, y = np.histogram(hd_2['age'], bins=8,
                                    density = True)
pdf = x/(sum(x))
print(pdf);
print(y)
cdf = np.cumsum(pdf)
plt.plot(y[1:],pdf)
plt.plot(y[1:], cdf)
```

```
[0.06222222 0.14222222 0.15555556 0.18222222 0.19555556 0.14222222
 0.08       0.04      ]
[30.    35.875 41.75  47.625 53.5   59.375 65.25  71.125 77.   ]
[0.04938272 0.2345679  0.19753086 0.20987654 0.12345679 0.13580247
 0.02469136 0.02469136]
[34.    40.125 46.25  52.375 58.5   64.625 70.75  76.875 83.   ]
```

Out[111]: [<matplotlib.lines.Line2D at 0x21a08075748>]

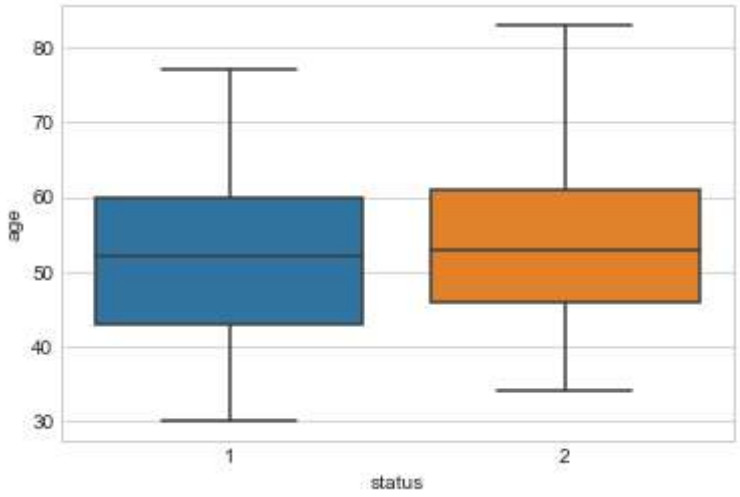


Observation(s):

1.The survival status is 1 for all the patients below 40 years of age.

Box plot and Whiskers

In [95]:
```
#age
sns.boxplot(x='status',y='age', data=hd)
plt.show()
```



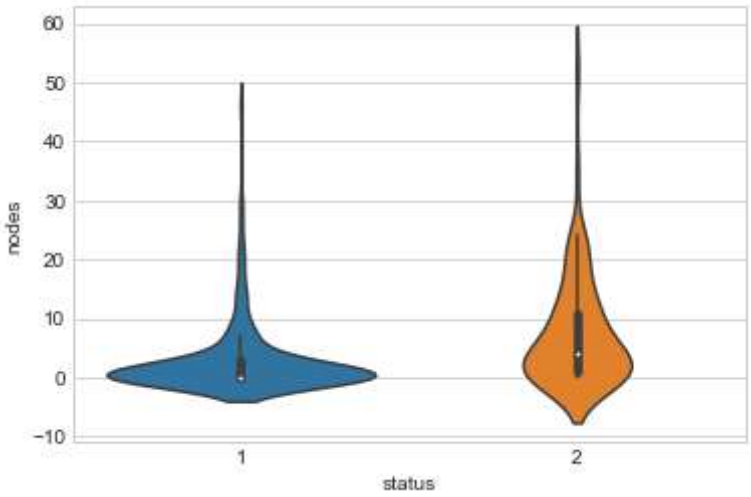
Observations

1.Survival status is 1 if the age of the patient is less than 34.

2.Survival status is 2 if the age of the patient is greater than 77.

Violin plots

In [116]:
```
sns.violinplot(x="status", y="nodes", data=hd, size=10)
plt.show()
```

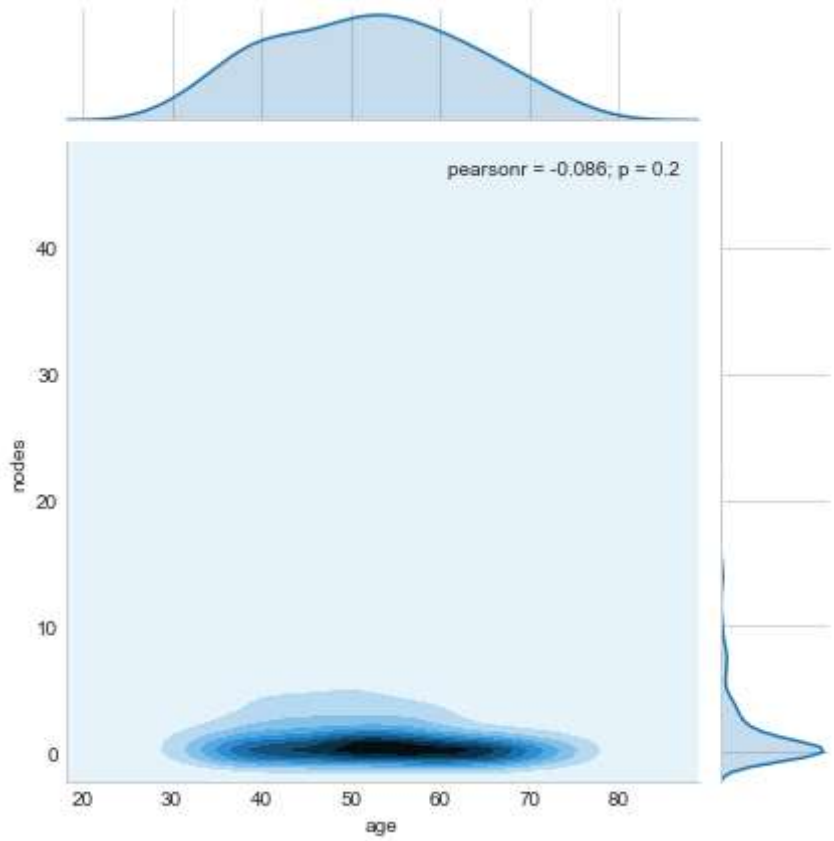


Observations:

1.If the number of nodes are above 50 then the status is 2.

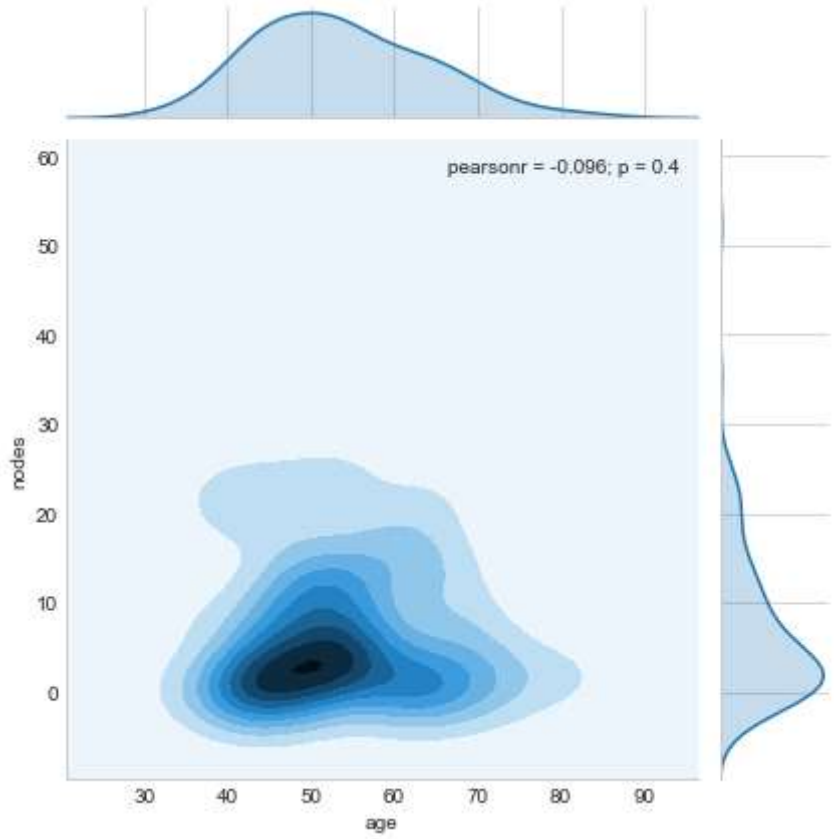
Multivariate probability density, contour plot

In [83]:
```
sns.jointplot(x="age", y="nodes", data=hd_1, kind="kde");
plt.show();
```



Observation(s) 1.The survival status is 1 if age is between 50 and 60 and no of nodes is 0.

In [84]:
```
sns.jointplot(x="age", y="nodes", data=hd_2, kind="kde");
plt.show();
```



Observations: 1.If the age is between 40 and 50 and the number of nodes are between '0' and '10' status tends to be 2.

Summary

- 1.Survival status is 1 if the age of the patient is less than 34.
- 2.Survival status is 2 if the age of the patient is greater than 77.
- 3.If the number of nodes are above 50 then the status is 2.
- 4.The survival status is 1 if age is between 50 and 60 and no of nodes is 0.
- 5.If the age is between 40 and 50 and the number of nodes are between '0' and '10' status tends to be 2.