Document created Date : 30-01-2019

Exercise Haberman Cancer Survival dataset Assignment1

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 (https://www.kaggle.com/gilsousa/habermans-survival-data-set))
2. Perform a similar alanlaysis as above on this dataset with the following sections:
3. High level statistics of the dataset: number of points, numer of features, number of classes, data-points per class.
4. Explain our objective.
5. Perform Univaraite analysis(PDF, CDF, Boxplot, Voilin plots) to understand which features are useful towards classification.
6. Perform Bi-variate analysis (scatter plots, pair-plots) to see if combinations of features are useful in classfication.
7. Write your observations in english as crisply and unambigously as possible. Always quantify your results.

About Habernman Cancer Survival DataSet

Information : The dataset contains cases from a study that was conducted beween 1958 and 1970 @ the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer.

- Number of Instances: 306
- Number of Attributes: 4 (including the class attribute)
- Attribute Information:
- Age of patient at time of operation
- Patient's year of operation (year)
- Number of positive axillary nodes detected
- Survival status (class attribute) 1 = the patient survived 5 years or longer, 2 = the patient died within 5 year
- Missing Attribute Values: None

Columns Names:

- #30 = age (Age of the patient at time of operation)
- #64 = yearof_op (Patient's Year of operation)
- #1 = positive_axil_nodes(Number of positive axillary nodes detected)
- #1.1 = surv_status(Survival status(class attribute) 1 = the patient survived 5 years or longer, 2 = the patient died within 5 year)

Source from kaggle (https://www.kaggle.com/gilsousa/habermans-survival-data-set (https://www.kaggle.com/gilsousa/habermans-survival-data-set))

C:\Users\Ramesh Battu> import required libraries

In [70]:

```
# importing requred Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

"""Loading Haberman DataSet and printing (Original dataset)"""


df = pd.read_csv("haberman.csv")
df.head()
```

Out[70]:

	30	64	1	1.1
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 [12]:

```
# AS the dataset does not have the headers, so defining header names and reloading
haberman = pd.read_csv("haberman.csv" , names=['age','yearof_op','positive_axil_nodes',
haberman
```

Out[12]:

	age	yearof_op	positive_axil_nodes	surv_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
5	33	58	10	1
6	33	60	0	1
7	34	59	0	2
8	34	66	9	2
9	34	58	30	1
10	34	60	1	1
11	34	61	10	1
12	34	67	7	1
13	34	60	0	1
14	35	64	13	1
15	35	63	0	1
16	36	60	1	1
17	36	69	0	1
18	37	60	0	1
19	37	63	0	1
20	37	58	0	1
21	37	59	6	1
22	37	60	15	1
23	37	63	0	1
24	38	69	21	2
25	38	59	2	1
26	38	60	0	1
27	38	60	0	1
28	38	62	3	1
29	38	64	1	1
...	...	...	...	...
276	67	66	0	1

	age	yearof_op	positive_axil_nodes	surv_status
277	67	61	0	1
278	67	65	0	1
279	68	67	0	1
280	68	68	0	1
281	69	67	8	2
282	69	60	0	1
283	69	65	0	1
284	69	66	0	1
285	70	58	0	2
286	70	58	4	2
287	70	66	14	1
288	70	67	0	1
289	70	68	0	1
290	70	59	8	1
291	70	63	0	1
292	71	68	2	1
293	72	63	0	2
294	72	58	0	1
295	72	64	0	1
296	72	67	3	1
297	73	62	0	1
298	73	68	0	1
299	74	65	3	2
300	74	63	0	1
301	75	62	1	1
302	76	67	0	1
303	77	65	3	1
304	78	65	1	2
305	83	58	2	2

306 rows × 4 columns

High level statistics of the dataset :

Number of points, numer of features, number of classes, data-points per class.

In [68]:

```
# Number of Points  and features
print(haberman.shape)
# 306 row's and 4 columns.
```

```
(306, 4)
```

In [69]:

```
print(haberman.columns)
```

```
Index(['age', 'yearof_op', 'positive_axil_nodes', 'surv_status'], dtype='o
bject')
```

In [18]:

```
# data Points per class
haberman.surv_status.value_counts()
```

Out[18]:

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

Observation : imbalnced dataset and more than 70% % of dataset was survived data set.

In [19]:

```
# summary of Habenman's Cancer Survival DataSet
haberman.describe()
```

Out[19]:

	age	yearof_op	positive_axil_nodes	surv_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

objective

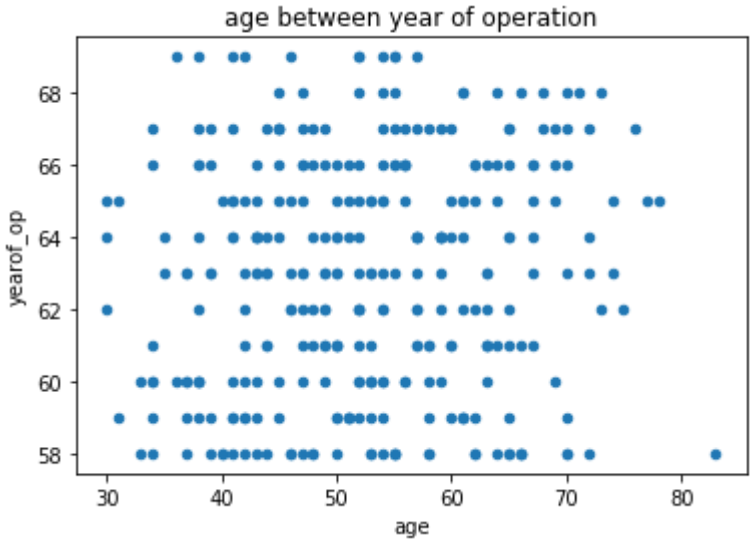
- Based on patients age, Year of Operation , positive Axial Nodes detection and survival status.
- why the patients are survived till 5 years or longer ?
- why the patients are died within 5 years ?
- whats are the chances to survive long ?
- whats are the Causes to died within 5 years ?

Performing Bi-variate analysis ("scatter plots", "pair-plots")

- **Scatter Plots**

In [21]:

```
# 2-D Scatter Plot

haberman.plot(kind="scatter", x = "age", y = "yearof_op", title = "age between year of
plt.show()
```



Observation(S)

- Can't able to classify by using age and yearof_op features .

In [24]:

```
# 2-D Scatter plot with colour coding for each class

sns.set_style("whitegrid")
sns.FacetGrid(haberman,hue='surv_status', size=6) \
    .map(plt.scatter,'age', 'yearof_op') \
    .add_legend()
plt.show()
```

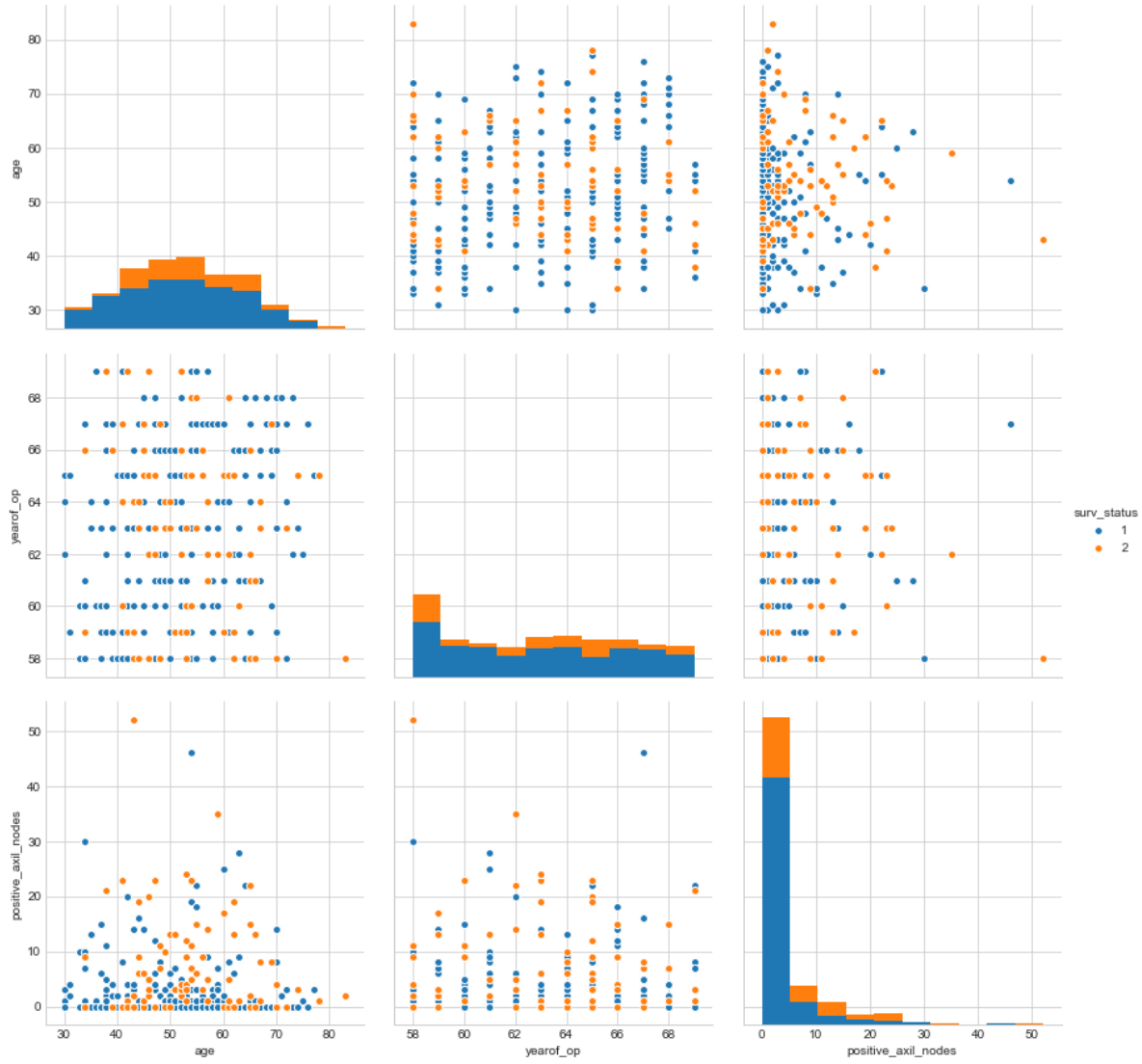
Observation

- By using 'age' and 'yearof_op' features in colour coding of 2-D scatterplot ,can not be classify the surv_status because of both are overlapped.

Pair-Plots

In [27]:

```
# pair wise plots:
plt.close();
sns.pairplot(haberman, hue='surv_status', vars=['age','yearof_op','positive_axil_nodes'
plt.show()
```



Observation(S)

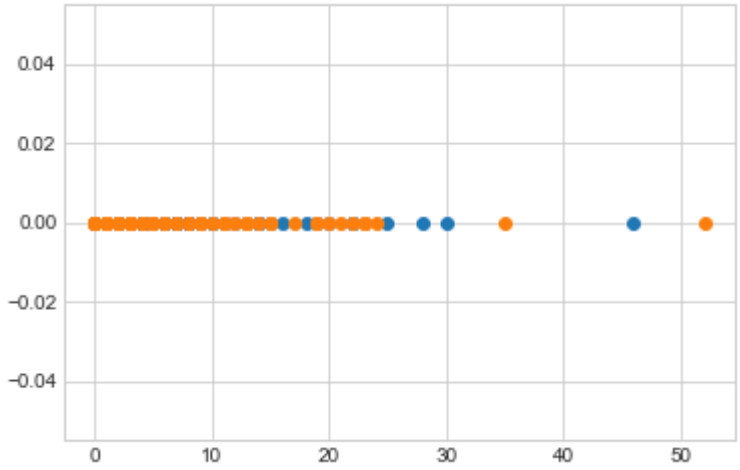
- Every pair plot was overlapped so can not be classify survival status between 1(survive 5years or longer) and 2(died within 5 years)
- but using yearof_op and positive_axil_nodes features can be moderately useful to classify the survival status,

Performing Univaraite analysis("PDF", "CDF", "Boxplot", "Voilin plots")

- **PDF**

In [35]:

```
# Histogram & PDF

patients_survive = haberman.loc[haberman["surv_status"] == 1] # patient_survive = patie
patients_died = haberman.loc[haberman["surv_status"] == 2]    # patients_died = patient

plt.plot(patients_survive['positive_axil_nodes'], np.zeros_like(patients_survive['posit
plt.plot(patients_died['positive_axil_nodes'], np.zeros_like(patients_died['positive_ax
plt.show()
```



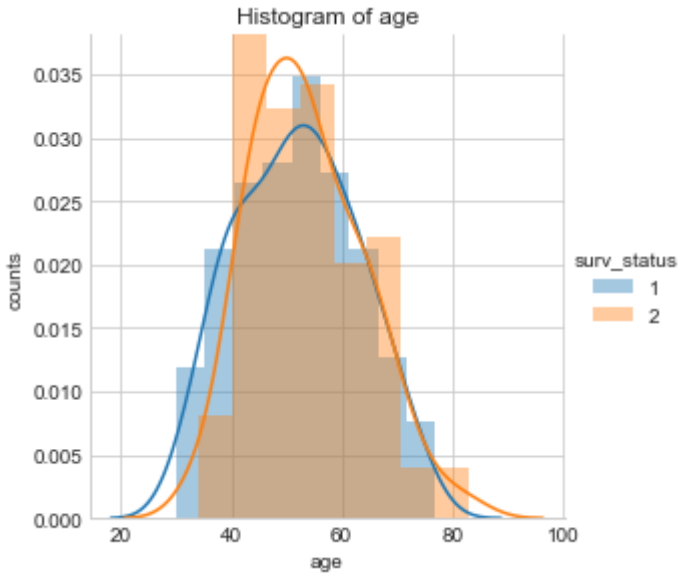
Observatios:

- very hard to classify , overlapping alot.

In [38]:

```
# Histogram of age feature
sns.FacetGrid(haberman, hue='surv_status',size = 4 ) \
    .map(sns.distplot, 'age') \
    .add_legend()
plt.ylabel('counts')
plt.title('Histogram of age')
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462:
UserWarning: The 'normed' kwarg is deprecated, and has been replaced by th
e 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462:
UserWarning: The 'normed' kwarg is deprecated, and has been replaced by th
e 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

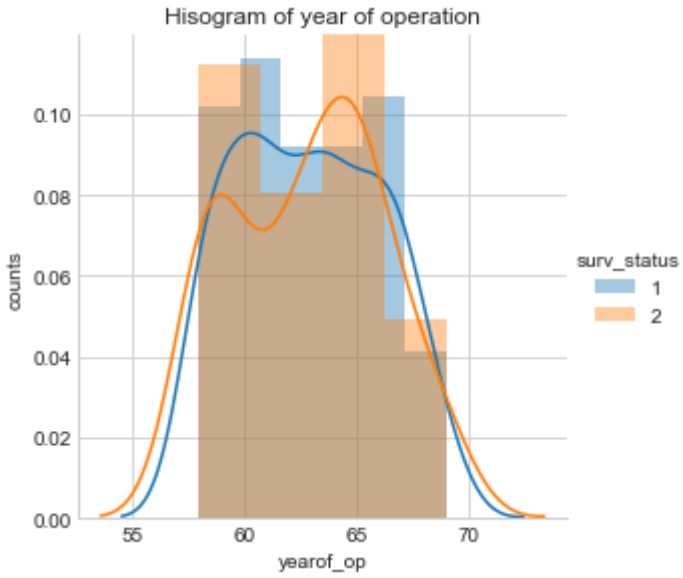


Observation : Very hard to classify by using 'age' featuture in histgram plot , because of overlapping alot

In [40]:

```
# Hisogram of yearof_op feature
sns.FacetGrid(haberman, hue = 'surv_status', size = 4) \
    .map(sns.distplot, 'yearof_op') \
    .add_legend()
plt.ylabel('counts')
plt.title('Hisogram of year of operation')
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462:
UserWarning: The 'normed' kwarg is deprecated, and has been replaced by th
e 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462:
UserWarning: The 'normed' kwarg is deprecated, and has been replaced by th
e 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

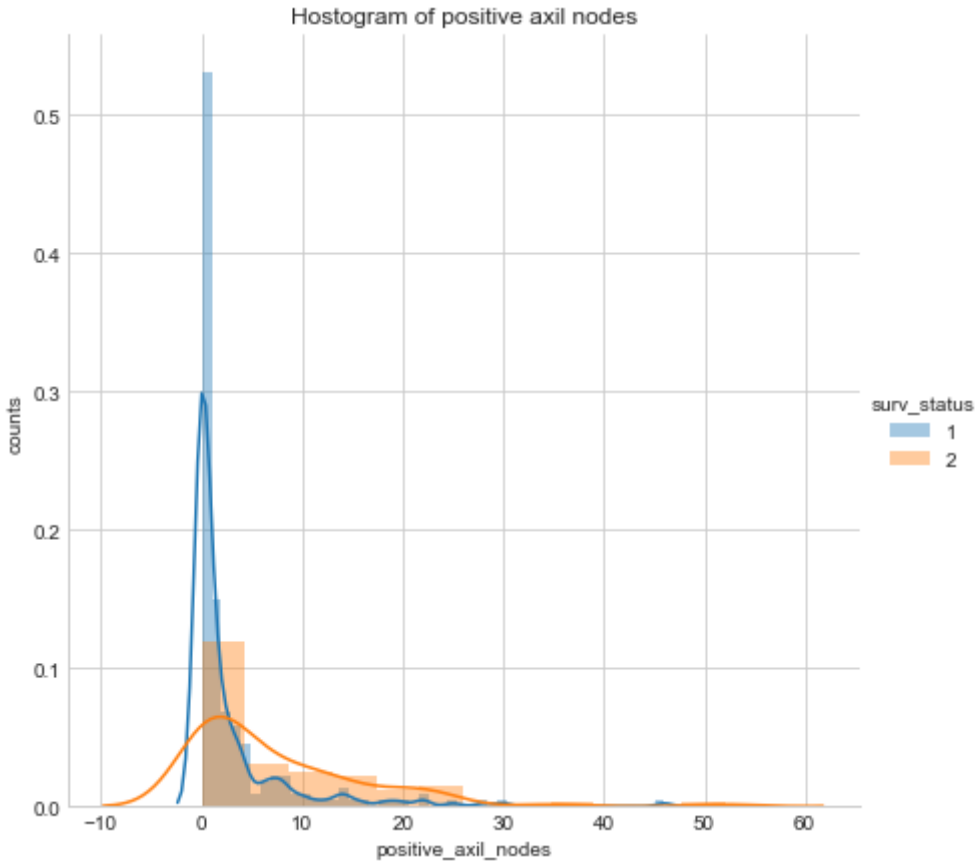


Observation : very hard to classify by using 'yearof_op' featuture in histgram plot , because of overlapping alot

In [41]:

```
# Histogram of positive_axil_nodes feature
sns.FacetGrid(haberman, hue = 'surv_status', size = 6) \
    .map(sns.distplot, 'positive_axil_nodes') \
    .add_legend()
plt.ylabel('counts')
plt.title('Hostogram of positive axil nodes')
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462:
UserWarning: The 'normed' kwarg is deprecated, and has been replaced by th
e 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6462:
UserWarning: The 'normed' kwarg is deprecated, and has been replaced by th
e 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```



Observations:

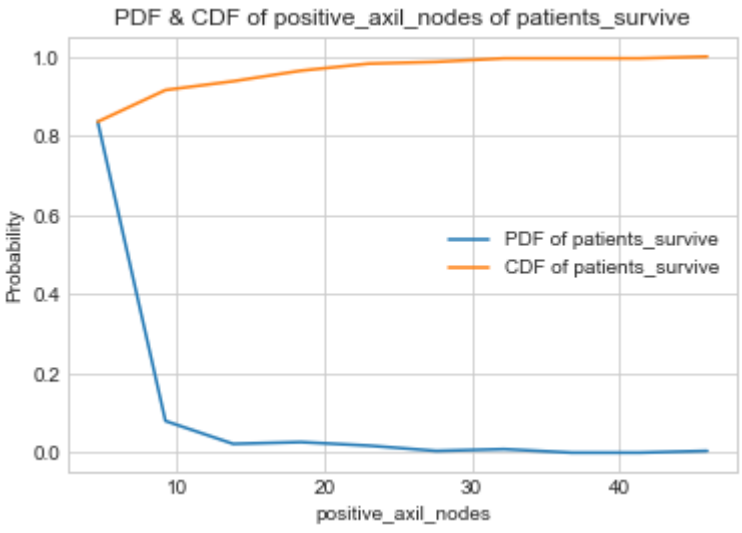
- Using 'positive_axil_nodes' feature , can classify the survival status ,
- And the maximum patients who had positive Axillary nodes lessthan 5 are survived .
- so we can use positive_axil_nodes feature to classify survival status and its easy for further analysis.

PDF & CDF of positive_axil_nodes of patients_survive

In [50]:

```
# PDF & CDF of positive_axil_nodes of patients_survive

counts , bin_edge = np.histogram(patients_survive['positive_axil_nodes'], bins = 10,
                                 density=True)
pdf = counts/sum((counts))
print('pdf:', pdf)
print('bin_edge:',bin_edge)

# compute CDF
cdf = np.cumsum(pdf)
plt.plot(bin_edge[1:],pdf, label='PDF of patients_survive')
plt.plot(bin_edge[1:],cdf, label= 'CDF of patients_survive')
plt.xlabel('positive_axil_nodes')
plt.ylabel('Probability')
plt.title('PDF & CDF of positive_axil_nodes of patients_survive')
plt.legend()
plt.show()
```

```
pdf: [0.83555556 0.08       0.02222222 0.02666667 0.01777778 0.00444444
 0.00888889 0.         0.         0.00444444]
bin_edge: [ 0.   4.6  9.2 13.8 18.4 23.  27.6 32.2 36.8 41.4 46. ]
```

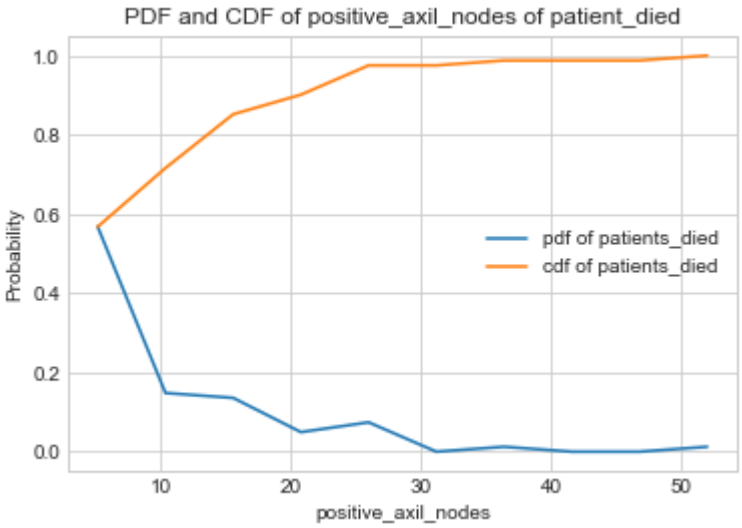


PDF and CDF of positive_axil_nodes of patients_died

In [55]:

```
# PDF AND CDF of positive_axil_nodes of patients_died

counts, bin_edge= np.histogram(patients_died['positive_axil_nodes'], bins= 10,
                               density=True)
pdf = counts/sum((counts))
print(pdf)
print(bin_edge)

cdf = np.cumsum(pdf)
plt.plot(bin_edge[1:],pdf,label='pdf of patients_died')
plt.plot(bin_edge[1:],cdf , label= 'cdf of patients_died')
plt.xlabel('positive_axil_nodes')
plt.ylabel('Probability')
plt.title('PDF and CDF of positive_axil_nodes of patient_died')
plt.legend()
plt.show()
```

```
[0.56790123 0.14814815 0.13580247 0.04938272 0.07407407 0.
 0.01234568 0.         0.         0.01234568]
[ 0.   5.2 10.4 15.6 20.8 26.  31.2 36.4 41.6 46.8 52. ]
```

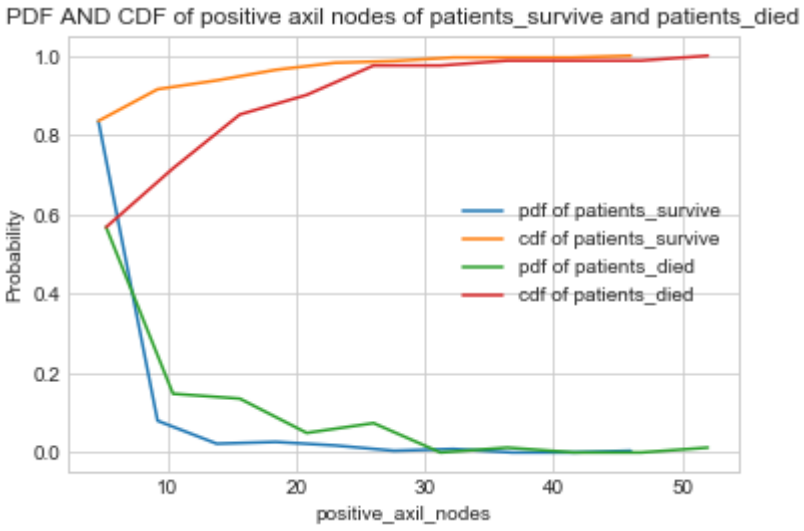


PDF AND CDF of positive axil nodes of patients_survive and patients_died

In [60]:

```
counts, bin_edge = np.histogram(patients_survive['positive_axil_nodes'], bins=10 ,
                                density=True)
pdf = counts/sum((counts))
print('pdf of patients_survive:', pdf)
print ('bin_edge of patients_survive:', bin_edge)

cdf = np.cumsum(pdf)
plt.plot(bin_edge[1:],pdf, label = 'pdf of patients_survive')
plt.plot(bin_edge[1:], cdf , label = 'cdf of patients_survive')
plt.xlabel('positive_axil_nodes')
plt.ylabel('Probabily')
plt.title('PDF AND CDF of positive axil nodes of patients_survive and patients_died')
plt.legend()


counts, bin_edge = np.histogram(patients_died['positive_axil_nodes'], bins = 10 ,
                                density=True)

pdf = counts/sum((counts))
print('pdf of patients_died:', pdf)
print('bin_edge of patients_died:', bin_edge)

cdf = np.cumsum(pdf)
plt.plot(bin_edge[1:], pdf , label = 'pdf of patients_died')
plt.plot(bin_edge[1:], cdf , label = 'cdf of patients_died')
plt.xlabel('positive_axil_nodes')
plt.ylabel('Probability')
plt.title('PDF AND CDF of positive axil nodes of patients_survive and patients_died')
plt.legend()
plt.show()
```

```
pdf of patients_survive: [0.83555556 0.08       0.02222222 0.02666667 0.01
777778 0.00444444
 0.00888889 0.         0.         0.00444444]
bin_edge of patients_survive: [ 0.   4.6  9.2 13.8 18.4 23.  27.6 32.2 36.
8 41.4 46. ]
pdf of patients_died: [0.56790123 0.14814815 0.13580247 0.04938272 0.07407
407 0.
 0.01234568 0.         0.         0.01234568]
bin_edge of patients_died: [ 0.   5.2 10.4 15.6 20.8 26.  31.2 36.4 41.6 4
6.8 52. ]
```



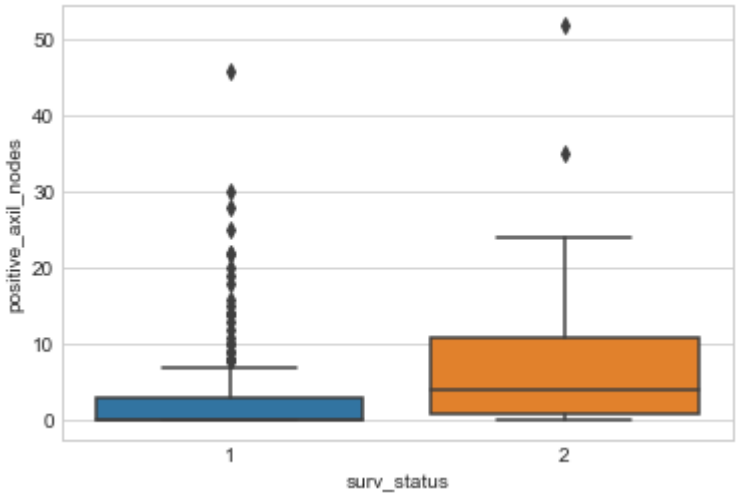
Observatios:

- 1. Probability of 83 % patients surivived who had lessthan 3 positive axilary nodes .
- 2. Probability of 57 % patients died who had lessthan 5 positive axilary nodes.

Box Plot

In [62]:

```
#Box Plot

sns.boxplot(data=haberman, x = 'surv_status', y = 'positive_axil_nodes')
plt.show()
```



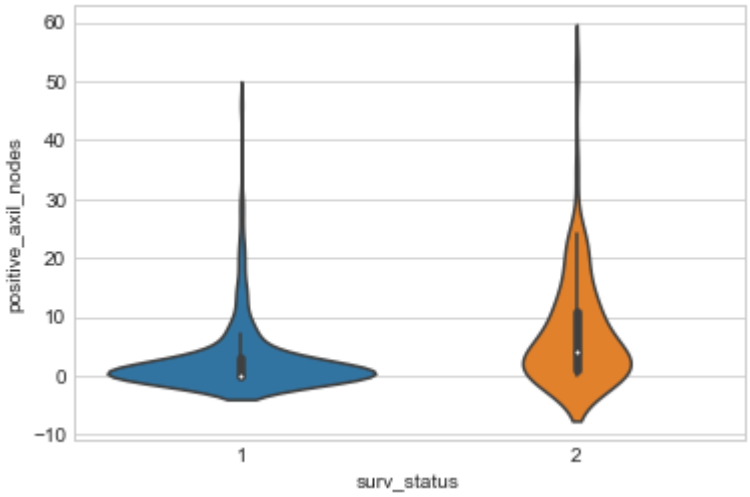
Box Plot Observatins:

1. 75 % of patients are survived at 3 positive Axil Nodes and its hard to classify 50% and 25 % because of its overlapped .
2. 75 % of patients are died at 11 positive axil nodes, 50 % of patients are died at 4 positive axil nodes and 25 % of patients are died at 1 positive axil node.

Violin Plot:

In [67]:

```
# Violin plot:
sns.violinplot(data = haberman, x = 'surv_status', y = 'positive_axil_nodes', size=(6))
plt.show()
```



Violin Plot Observations :

1. 50 % of patients are survived at 0 positive axil nodes , 75% of patients are survived at 3 positive nodes and 25 % is overlapped.
2. 75 % of patients are died at 11 positive axil nodes , 50 % of patients are died at 4 positive axil nodes and of patients are died at 0 positive axil node.

Summarizing plots

Observations And Conclusions :

- 1. unable to calssify properly by any of EDA , due to very less amount of dataset and imbalnced dataset.
- 2. As per the given dataset , positive axillir nodes feature was very useful to moderatively classify survival status
- 3. who had less amount of positive axillir nodes detection (zero or less than 3), always there is a huge Probabilities to survive 5 years or longer .
- 4. who had more amount of positive axillir nodes detection (5 or grater than or equal to 11) , those patinets have a chance to died .
- 5. more than 70 % patients were survived and 27 % patients were died .

Thank You.

sign off Ramesh Battu