Exploratory Data Analysis on Haberman Dataset

This dataset contains cases from a study that was conducted between 1958 and 1970 at the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer.

In [1]:
```
# importing libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import os
import warnings
warnings.filterwarnings('ignore')
```

Objective :- To classify/predict a patient survival who had undergone surgery for breast cancer.

In [2]:
```
# loading the dataset
data = pd.read_csv('../Applied_AI/haberman.csv')
```

In [3]:
```
#The below command prints information about no.of datapoints and features.
data.shape
```

Out[3]:
```
(306, 4)
```

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

Out[4]:
```
Index(['Age', 'year', 'positive_axillary_nodes', 'survival_status'], dtype='object')
```

Obervation:- The dataset contains 306 datapoints(rows) and 4 features including class label(columns).

In [5]:
```
#The below command prints the first five rows in the dataset
data.head()
```

Out[5]:

	Age	year	positive_axillary_nodes	survival_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 [6]:
```
#The below command prints the last five rows in the dataset

data.tail()
```

Out[6]:

	Age	year	positive_axillary_nodes	survival_status
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

Features Information

1. Age:- It represents the age of patient at the time of operation(numerical).

2. Year:- It represent the year of operation(Numerical).
3. Positive_axillary_nodes:- It represents no.of positive auxillary nodes detected(numerical).
4. Survival_status(Class Attribute):- 1 represents the patient survived 5 years or longer after the operation, 2 represents the patient died within 5 years.

In [7]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 306 entries, 0 to 305
Data columns (total 4 columns):
Age                     306 non-null int64
year                    306 non-null int64
positive_axillary_nodes    306 non-null int64
survival_status         306 non-null int64
dtypes: int64(4)
memory usage: 9.6 KB
```

In [8]:
```
data.isna().sum()
```

Out[8]:
```
Age                     0
year                    0
positive_axillary_nodes    0
survival_status         0
dtype: int64
```

Observation:- Data is clean and has no missing values

In [9]:
```
#to get the details like number of observations, min,max,25%,50%,75% ,mean,std
data.describe()
```

Out[9]:

	Age	year	positive_axillary_nodes	survival_status
count	306.000000	306.000000	306.000000	306.000000

	Age	year	positive_axillary_nodes	survival_status
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

Observation

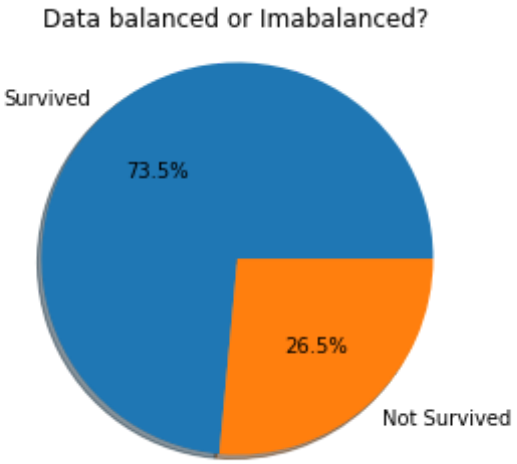
1. The age of patient varies from 30 to 83 and 75% patients are 61 year old.
2. The maximum number of nodes is 52 and 75% patients have 4 nodes.

In [10]:
```
# converting 1 as survided and 2 as not survived
data.survival_status.replace([1,2],['Survived','Not Survived'], inplace
 = True)
```

In [11]:
```
data['survival_status'].value_counts()
```

Out[11]:
```
Survived        225
Not Survived     81
Name: survival_status, dtype: int64
```

In [12]:
```
sizes = [data['survival_status'].value_counts()['Survived'],data['survi
val_status'].value_counts()['Not Survived']]
labels = ['Survived','Not Survived']
plt.pie(sizes,labels=labels,autopct='%1.1f%%', shadow=True)
plt.axis('equal')
plt.title("Data balanced or Imabalanced?")
plt.show()
```



Observation

1. The target variables is imbalanced as it does not contain euqal number of data-points for each class.It is clearly visible from the pie chart as 73.5% percent survived after 5 years of treatment.

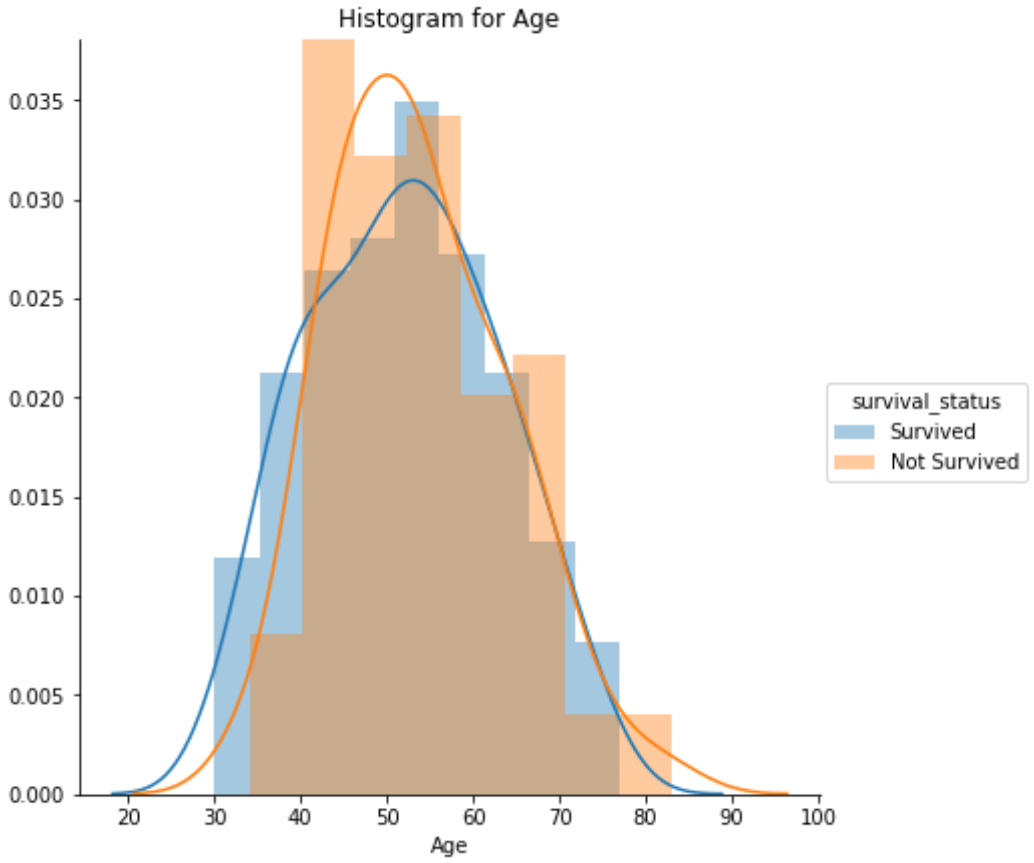
Univariate Analysis(pdf, cdf, boxplot and violin plot)

Univariate analysis is the simplest form of analyzing data. “Uni” means “one”, "variate" means "variable or numeric variable" so, in other words data has only one variable. It doesn't deal with causes or relationships and it's major purpose is to describe; it takes data, summarizes that data and finds patterns in the data.

1. Distribution plots are used to visually assess how the data points are distributed with respect to its frequency.
2. Usually the data points are grouped into bins and the height of the bars representing each group increases with
   increase in the number of data points lie within that group. (histogram)

3. Probality Density Function (PDF) is the probabilty that the variable takes a value x. (smoothed version of the histogram)
4. Kernel Density Estimate (KDE) is the way to estimate the PDF. The area under the KDE curve is 1.
5. Here the height of the bar denotes the percentage of data points under the corresponding group

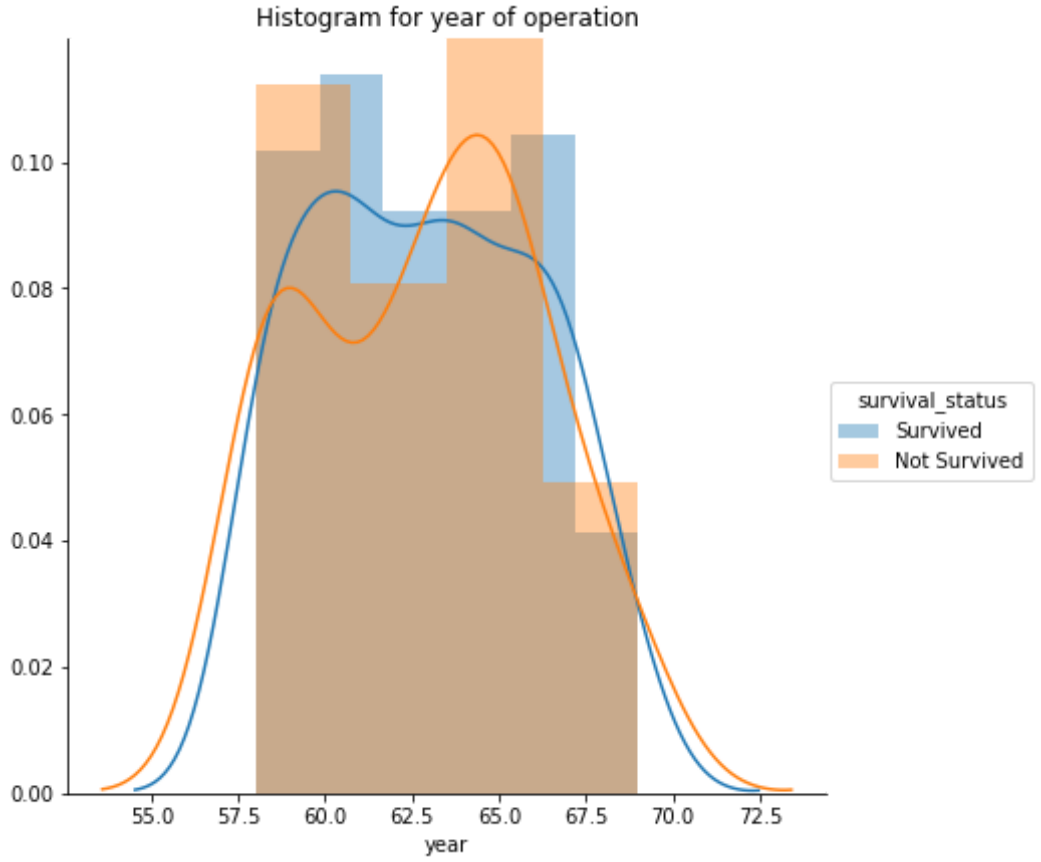
In [13]:
```
sns.FacetGrid(data, hue = 'survival_status', size = 6)\
    .map(sns.distplot, 'Age')\
    .add_legend()
plt.title('Histogram for Age')
plt.show()
```



Observation

From the above PDF, The only observation that we can draw is patients between age 30-40,70-75 survived more than 5 years.

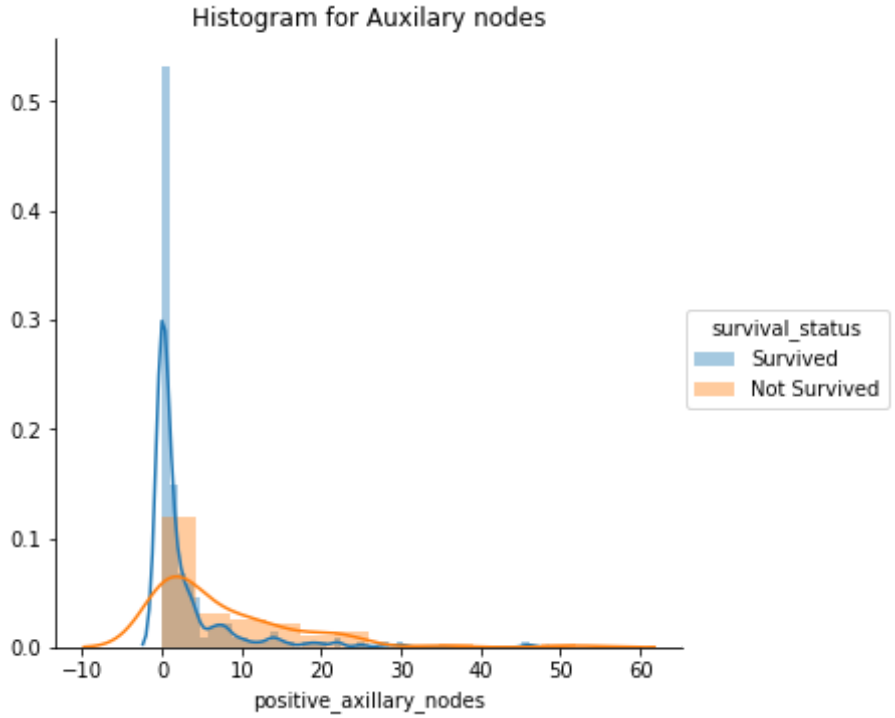
In [14]:
```
sns.FacetGrid(data, hue = 'survival_status', size = 6)\
    .map(sns.distplot, 'year')\
    .add_legend()
plt.title('Histogram for year of operation')
plt.show()
```



Observation

As we can see in the above pdf,both the classes for year features is almost overalp each other. only minor observation we can draw is patient undergone treatment in between 61-64, 66-67 survived for mor than 5 years.

In [15]:
```
sns.FacetGrid(data, hue = 'survival_status', size = 5)\
    .map(sns.distplot, 'positive_axillary_nodes')\
    .add_legend()
plt.title('Histogram for Auxilary nodes')
plt.show()
```



Observation

1. From the above PDF figure axillary nodes detection is clear for us, so we can say that probability distribution function(PDF) of survived is high when compared to not survived.
2. If positive_axillary_node <=3 then the patient is survived else not survived.

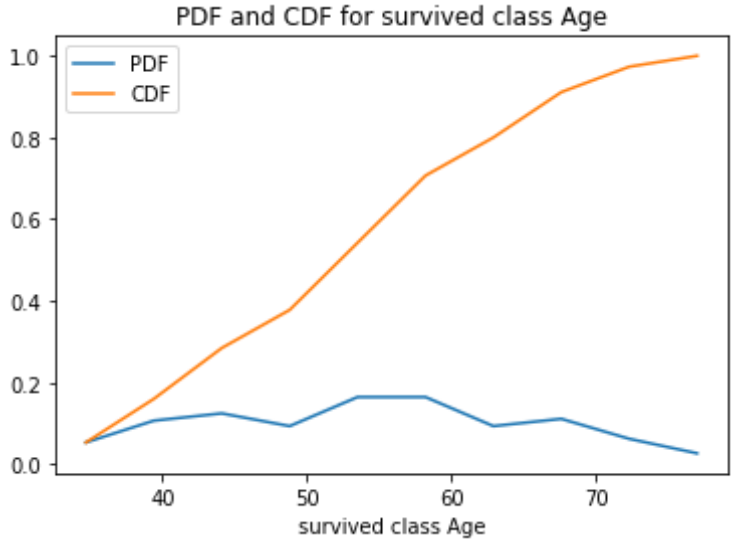
CDF

To check for a given class, how the number of individuals vary with the values of an attribute.

In [16]:
```
print('Max age: ' + str(data['Age'].max()), '\nMin age: ' + str(data['A
ge'].min()))
print('Size of data to be binnned: ' + str(data['Age'].shape))
```

```
Max age: 83
Min age: 30
Size of data to be binnned: (306,)
```

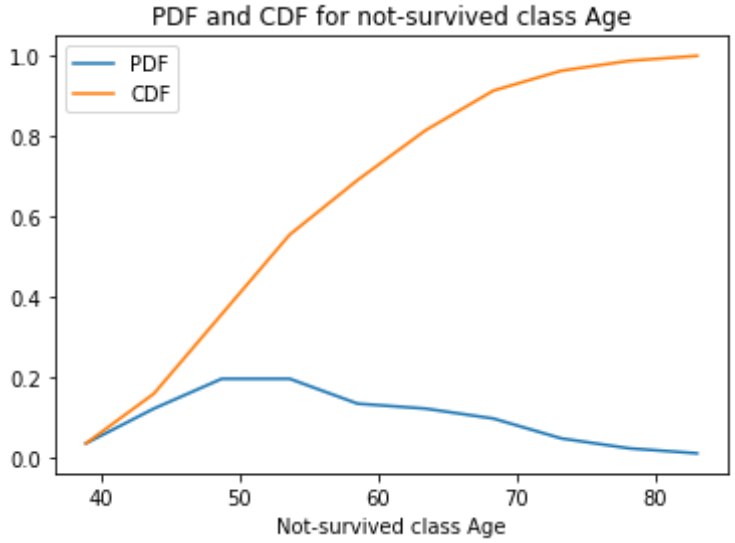
In [17]:
```
data_class_survived = data[data['survival_status'] == 'Survived']

counts, bin_edges = np.histogram(data_class_survived['Age'], bins= 10,
density = True)


pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF')
plt.plot(bin_edges[1:], cdf, label = 'CDF')
plt.xlabel('survived class Age')
plt.title('PDF and CDF for survived class Age')
plt.legend()
plt.show()
```



observation

1. From the CDF, about 80% of patients in survived class are under the age of 65 (60<age<65). On the other hand, all the patients of survived class fall under the age of 80 (or under the age of 77- 78).
2. From the PDF above, we can infer that most people who belonged to survived class(survived for more than 5 yrs) belonged to the age group 50 to 60 yrs.

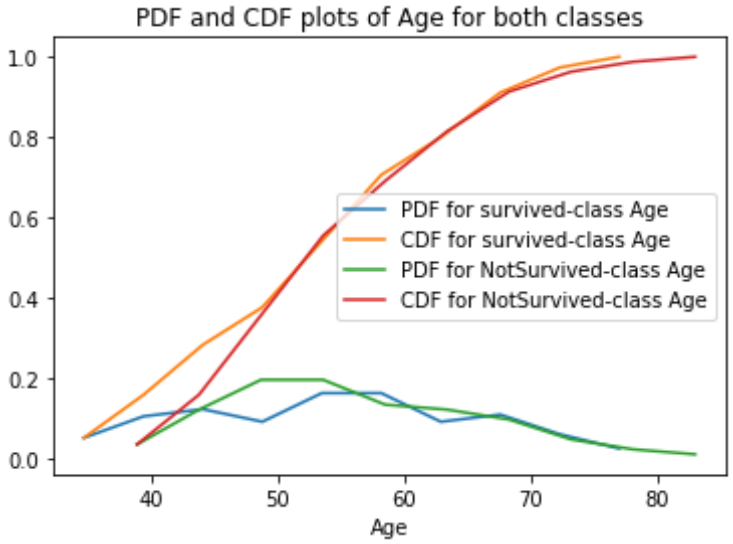
In [18]:
```
data_class_not_survived = data[data['survival_status'] == 'Not Survive
d']

counts, bin_edges = np.histogram(data_class_not_survived['Age'], bins=
10, density = True)

pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF')
plt.plot(bin_edges[1:], cdf, label = 'CDF')
plt.xlabel('Not-survived class Age')
plt.title('PDF and CDF for not-survived class Age')
plt.legend()
plt.show()
```



observations

1. The CDF also crosses 80% for age slightly less than 65 but more than 60 (similar to survived class patients). However, there patients age is more than 80 in not survived class.
2. From the above PDF plot, we see that most people who belonged to not survived class were of the age group 45 to 58. If we compare this with the plot for survived class, we find that there is not much difference as most people who survived were of the age group 50 to 60 yrs.

In [19]:
```
counts, bin_edges = np.histogram(data_class_survived['Age'], bins= 10,
density = True)
pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.title('PDF and CDF plots of Age for both classes')
plt.plot(bin_edges[1:], pdf, label = 'PDF for survived-class Age')
plt.plot(bin_edges[1:], cdf, label = 'CDF for survived-class Age')
plt.xlabel('Age')

counts, bin_edges = np.histogram(data_class_not_survived['Age'], bins =
 10, density = True)

pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF for NotSurvived-class Age')
plt.plot(bin_edges[1:], cdf, label = 'CDF for NotSurvived-class Age')
plt.legend()
plt.show()
```



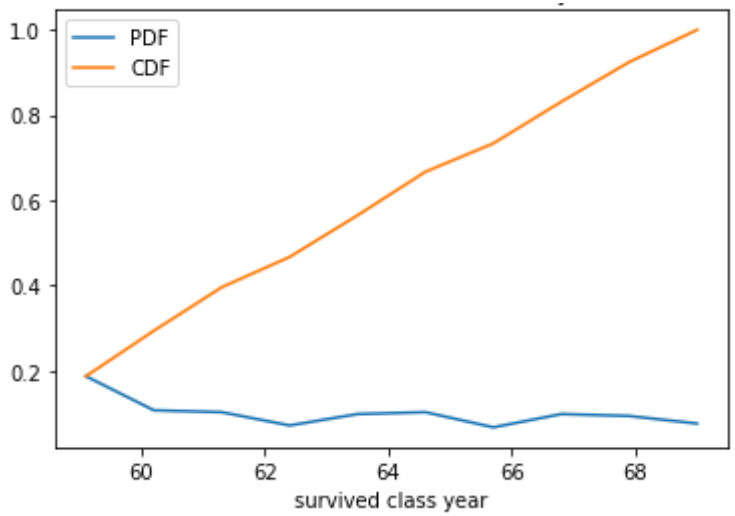
observation

It is difficult to distinguish between two classes using any rule based on Age.

After plotting PDFs and CDFs for both the classes with respect to age attribute, it becomes clear that it is very difficult to discern the two classes using Age attribute (no if-then rules possible).

In [20]:
```
counts, bin_edges = np.histogram(data_class_survived['year'], bins= 10,
 density = True)

pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF')
plt.plot(bin_edges[1:], cdf, label = 'CDF')
plt.xlabel('survived class year')
plt.title('PDF and CDF for survived class year')
plt.legend()
plt.show()
```

PDF and CDF for survived class year



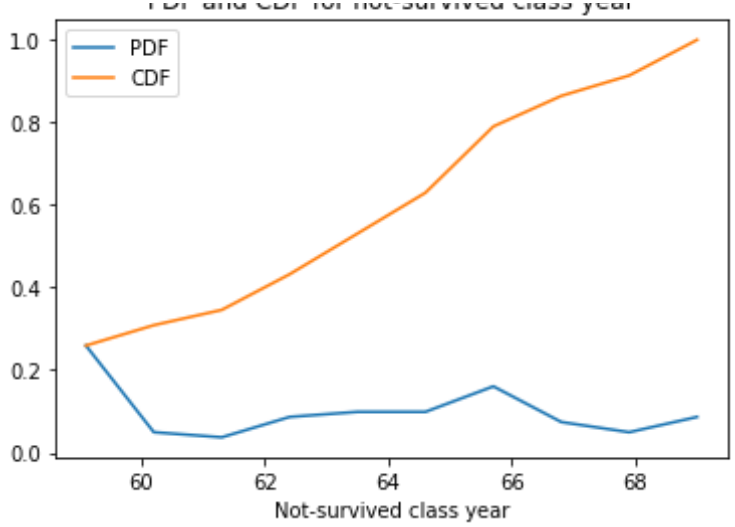
Observation

From the CDF,patients who survived more than five years after the operation has undergone the operation before 1969.

From the PDF,patients who survived more than five years after the operation has undergone the operation between 1963-65 and 1967-68.

In [21]:
```
counts, bin_edges = np.histogram(data_class_not_survived['year'], bins=
 10, density = True)

pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF')
plt.plot(bin_edges[1:], cdf, label = 'CDF')
plt.xlabel('Not-survived class year')
plt.title('PDF and CDF for not-survived class year')
plt.legend()
plt.show()
```

PDF and CDF for not-survived class year



Observation

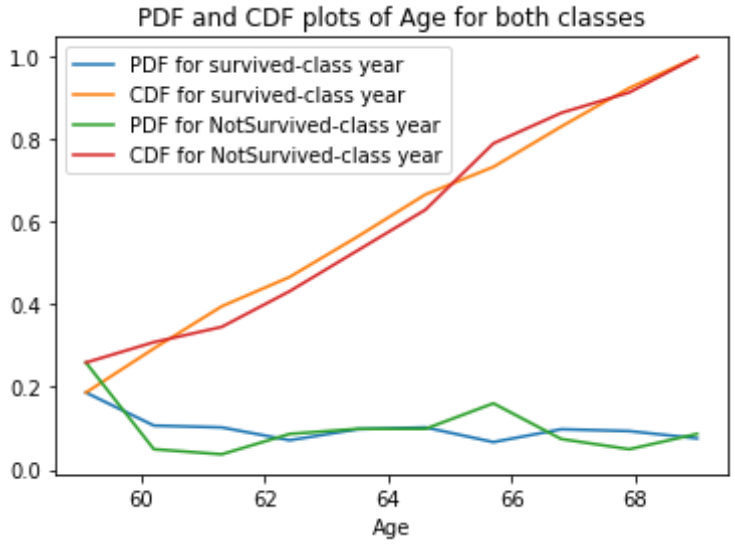
From the CDF,patients who survived less than five years after the operation has undergone the operation before 1969.

From the PDF,patients who survived less than five years after the operation has undergone the operation between 1963-64 and 1965-66.

Both pdf and cdf plot of non-survived class overlap with the survived class plot.

In [22]:
```
counts, bin_edges = np.histogram(data_class_survived['year'], bins= 10,
 density = True)
pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.title('PDF and CDF plots of Age for both classes')
plt.plot(bin_edges[1:], pdf, label = 'PDF for survived-class year')
plt.plot(bin_edges[1:], cdf, label = 'CDF for survived-class year')
plt.xlabel('Age')

counts, bin_edges = np.histogram(data_class_not_survived['year'], bins
```

```
= 10, density = True)

pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF for NotSurvived-class year')
plt.plot(bin_edges[1:], cdf, label = 'CDF for NotSurvived-class year')
plt.legend()
plt.show()
```

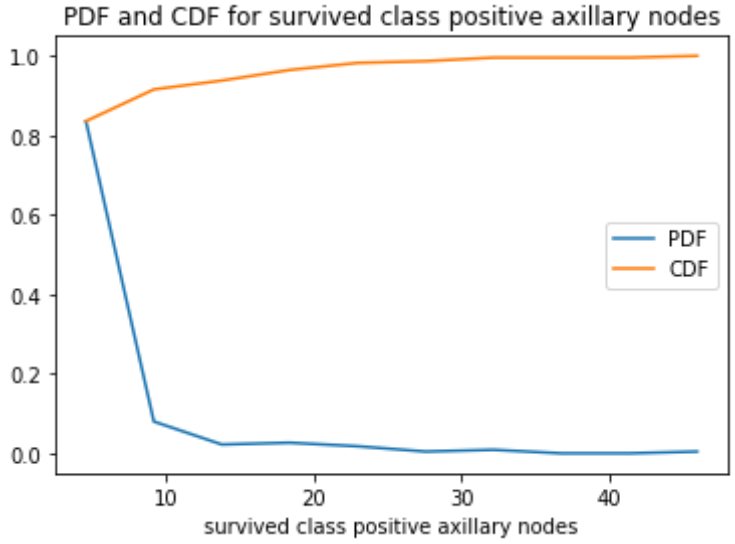


Observation

From the above plots (individual plots for both classes wrt Year of Operation as well as from plotting both together in a single plot), we find that it is not possible to derive any kind of classification rules for the two classes using Year of Operation feature alone.

In [23]:
```
counts, bin_edges = np.histogram(data_class_survived['positive_axillary
_nodes'], bins= 10, density = True)

pdf = counts/sum(counts)
```

```
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF')
plt.plot(bin_edges[1:], cdf, label = 'CDF')
plt.xlabel('survived class positive axillary nodes')
plt.title('PDF and CDF for survived class positive axillary nodes')
plt.legend()
plt.show()
```



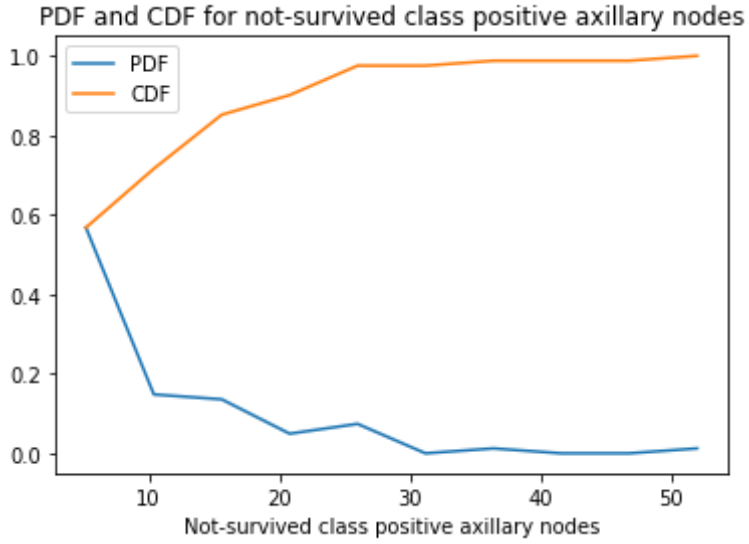
Observation

From CDF, we can say that 80% of the patients were detected with less than 2 or 3 (atleast less than 5). This could be useful information if we see some different trend in values of Auxilliary nodes detected for not-survived class too.

From PDF, We observe that in survived class 10 is the most frequent value for Axillary nodes detected.

In [24]:
```
counts, bin_edges = np.histogram(data_class_not_survived['positive_axil
lary_nodes'], bins= 10, density = True)
```

```
pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF')
plt.plot(bin_edges[1:], cdf, label = 'CDF')
plt.xlabel('Not-survived class positive axillary nodes')
plt.title('PDF and CDF for not-survived class positive axillary nodes')
plt.legend()
plt.show()
```

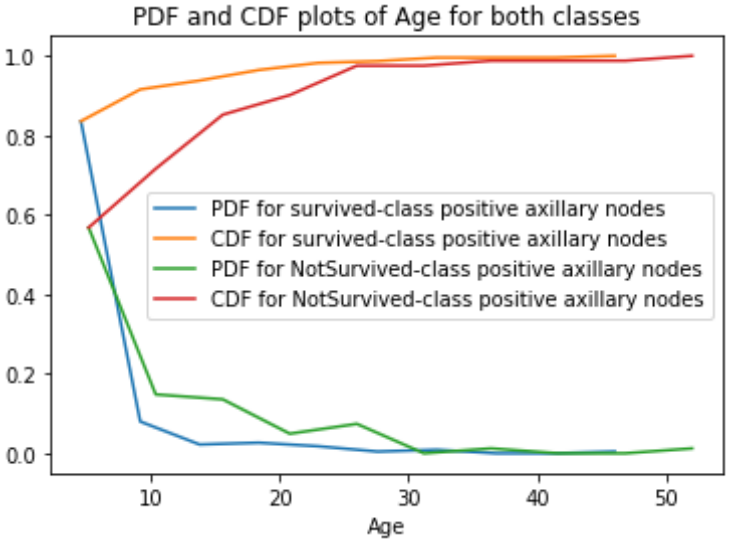


Observation

From the PDF, most of the patients had Axillary nodes detected value less than 10, we got similar observations for survived class also.

From the CDF, 80% of the patients were detected with less than (atleast) 15 axillary nodes. Although survived class had 80% of its patients with less than 5 axillary nodes, we still need to check as the difference is not too much.

In [25]:
```
counts, bin_edges = np.histogram(data_class_survived['positive_axillary
_nodes'], bins= 10, density = True)
pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.title('PDF and CDF plots of Age for both classes')
plt.plot(bin_edges[1:], pdf, label = 'PDF for survived-class positive a
xillary nodes')
plt.plot(bin_edges[1:], cdf, label = 'CDF for survived-class positive a
xillary nodes')
plt.xlabel('Age')

counts, bin_edges = np.histogram(data_class_not_survived['positive_axil
lary_nodes'], bins = 10, density = True)

pdf = counts/sum(counts)
cdf = np.cumsum(pdf)

plt.plot(bin_edges[1:], pdf, label = 'PDF for NotSurvived-class positiv
e axillary nodes')
plt.plot(bin_edges[1:], cdf, label = 'CDF for NotSurvived-class positiv
e axillary nodes')
plt.legend()
plt.show()
```

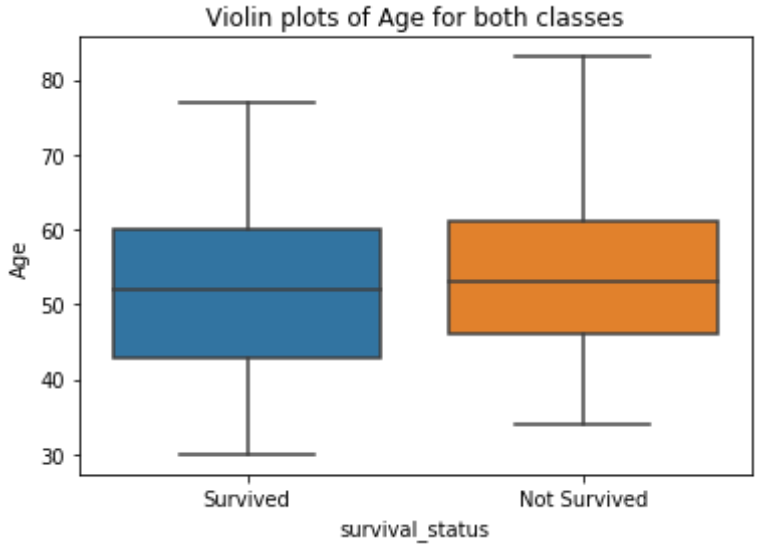


observation

From combined plot of both the classes, 80% survived class patients had number of positive axiliary nodes detected < 5, 60% of not survived class patients also had the same number of detections. PDF is all overlapping. Any rule based on this would be highly full of errors. Hence, no classification rules can be made.

Box Plot

In [26]:
```
sns.boxplot(x = 'survival_status', y = 'Age', data = data)
plt.title('Violin plots of Age for both classes')
plt.show()
```



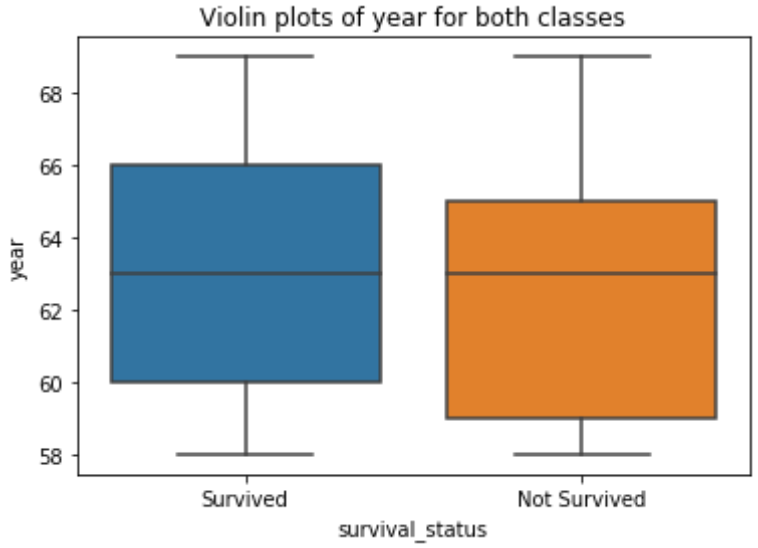
observation

For survived class: Age = 60 is the 75th percentile, so 75% of the datapoints are of age less than 60 who survived for more than 5 years. 50th percentile looks to be 52 or 53, while 25th percentile value is some value less than 45. Most of the datapoints lie within 30 and around 78.

For not survived class: 75th percentile is little more than 60 (in between 61 and 62), 50th percentile is almost the same value as the 50th percentile of survived class. 25th percentile is some value more than 45. Most of the datapoints lie within 82 and around 35.

Survived Class box plot has more Inter-quartile Range than not survived class.

In [27]:
```
sns.boxplot(x = 'survival_status', y = 'year', data = data)
plt.title('Violin plots of year for both classes')
plt.show()
```

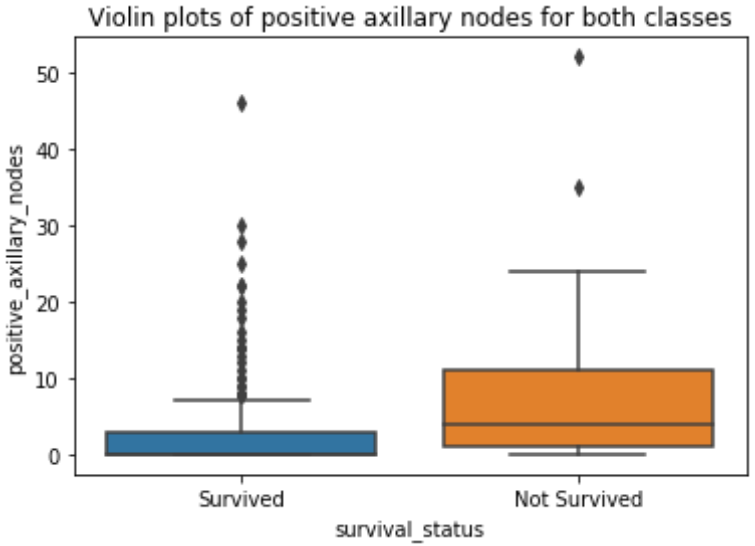


observation

For survived class: 75th percentile is the year 1966. That is, 75% of the people survived for more than 5 years underwent operation before 1966. 1963 is the 50th percentile year of percetile and 25% of the patients who survived for more than 5 years were operated before 1960.

For not survived class: Those who did not survive, 75% of them were operated before 1965. This class has the same 50th percentile as survived class- 1963, ie, 50% of the people who did not survive were operated before 1963. 25th percentile is year 1959.

In [28]:
```
sns.boxplot(x = 'survival_status', y = 'positive_axillary_nodes', data
= data)
plt.title('Violin plots of positive axillary nodes for both classes')
plt.show()
```



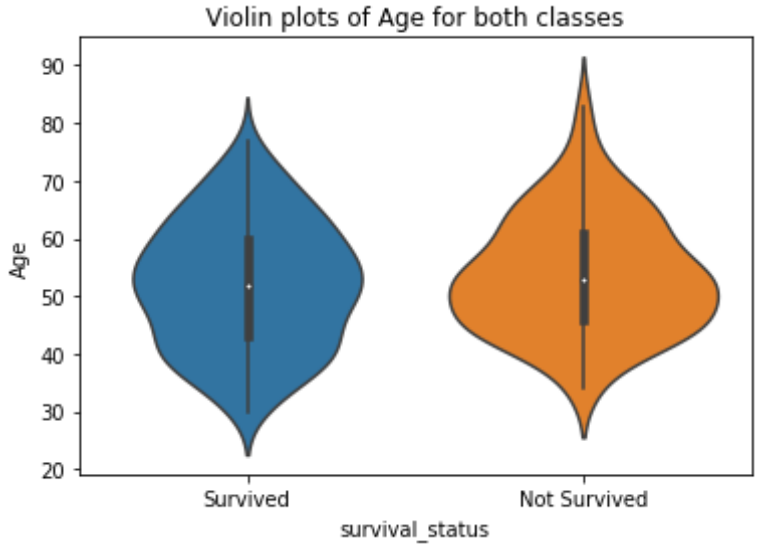
Observation

For survived class: The first observation that we could say is that there are quite a number of outliers in the values of Axillary nodes detected attribute. The 75th percentile looks to be 4, that is 75% percent of the people who underwent operation and survived for more than 5 years had less than atleat 5 axillary nodes detected.

For not survived class: 75% of the people who underwent operation and didn't survived for more than five years had less than atleast 11 positive axillary nodes detected.50% percentile people have less than 4 axillary nodes detected.

Violion Plot

In [29]:
```
sns.violinplot(x = 'survival_status', y = 'Age', data = data)
plt.title('Violin plots of Age for both classes')
plt.show()
```

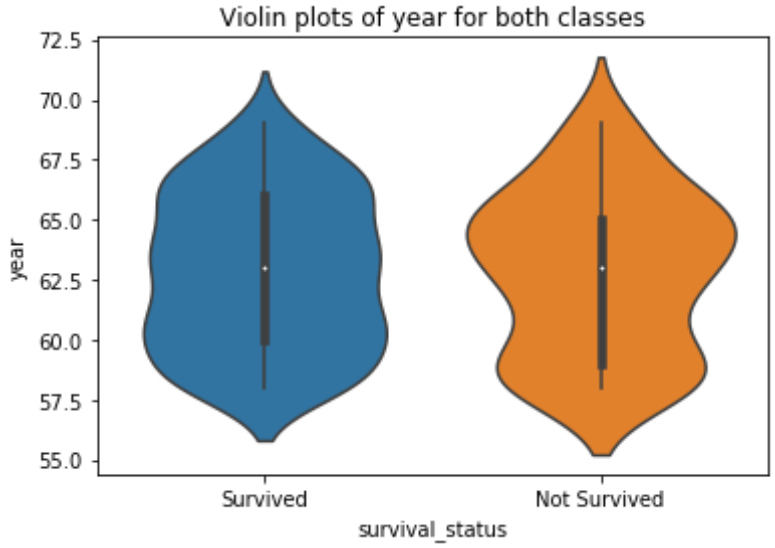


observation

For survived class: Age = 60 is the 75th percentile, so 75% of the datapoints are of age less than 60 who survived for more than 5 years. 50th percentile looks to be 52 or 53, while 25th percentile value is some value less than 45. Most of the datapoints lie within 30 and around 78.

For not survived class: 75th percentile is little more than 60 (in between 61 and 62), 50th percentile is almost the same value as the 50th percentile of survived class. 25th percentile is some value more than 45. Most of the datapoints lie within 82 and around 35.

In [30]:
```
sns.violinplot(x = 'survival_status', y = 'year', data = data)
plt.title('Violin plots of year for both classes')
plt.show()
```

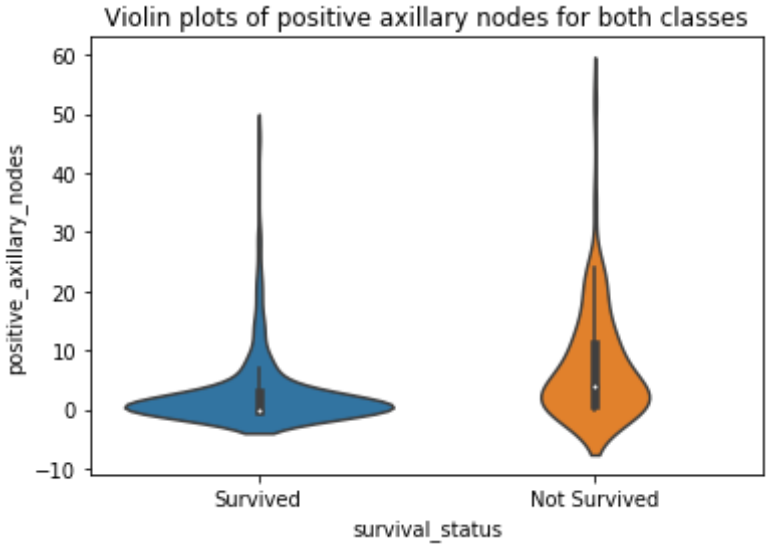


Observation

For survived class: 75th percentile is the year 1966. That is, 75% of the people survived for more than 5 years underwent operation before 1966. 1963 is the 50th percentile year of percetile and 25% of the patients who survived for more than 5 years were operated before 1960.

For not survived class: Those who did not survive, 75% of them were operated before 1965. This class has the same 50th percentile as survived class- 1963, ie, 50% of the people who did not survive were operated before 1963. 25th percentile is year 1959.

In [31]:
```
sns.violinplot(x = 'survival_status', y = 'positive_axillary_nodes', da
ta = data)
plt.title('Violin plots of positive axillary nodes for both classes')
plt.show()
```



observation

For survived class: The first observation that we could say is that there are quite a number of outliers in the values of Axillary nodes detected attribute. The 75th percentile looks to be 4, that is 75% percent of the people who underwent operation and survived for more than 5 years had less than atleat 5 axillary nodes detected. Since the top whisker is below 10, so most survivors had less than 10 detections of axillary nodes.

For not survived class: 75% of the people who underwent operation and didn't survived for more than five years had less than atleast 11 positive axillary nodes detected.50% percentile people have less than 4 axillary nodes detected.

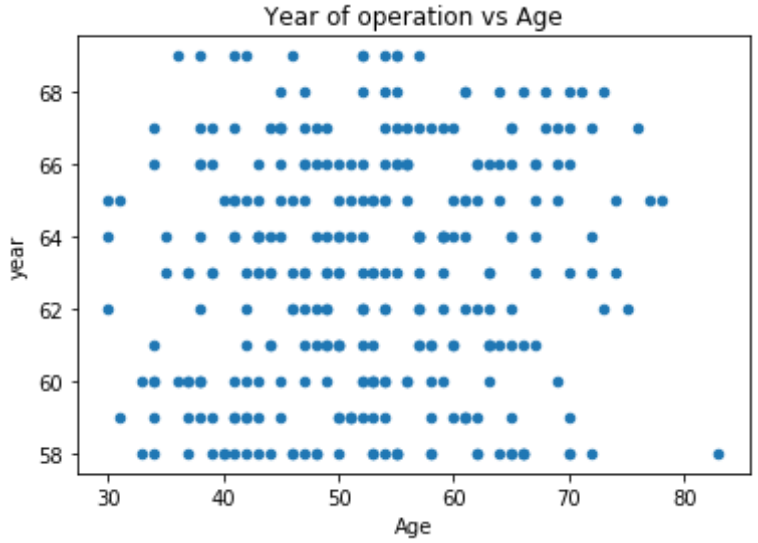
Bivariate Analysis

Bivariate analysis is one of the simplest forms of quantitative (statistical) analysis. It involves the analysis of two variables (often denoted as X, Y), for the purpose of determining the empirical relationship between them.

Scatter Plot

A scatter plot is a useful visual representation of the relationship between two numerical variables (attributes) and is usually drawn before working out a linear correlation or fitting a regression line. The resulting pattern indicates the type (linear or non-linear) and strength of the relationship between two variables.

In [32]:
```
data.plot(kind='scatter',x='Age',y='year');
plt.title('Year of operation vs Age')
plt.show()
```



observation

The idea of deriving ideas from this plot does not make any sense since a lot of better information can be derived from sns plots along with the above information as well. Since plotting a scatter plot for any pair of features woud not be as productive for this case.

Using Seaborn API to plot the data points so that each datapoints belonging to different classses as coloured differently, and datapoints belonging to the same class are colored in the same color

Seaborn Scatter plot-1. Age and Years of Operation: Age on x-axis and Years of Operation on y-axis

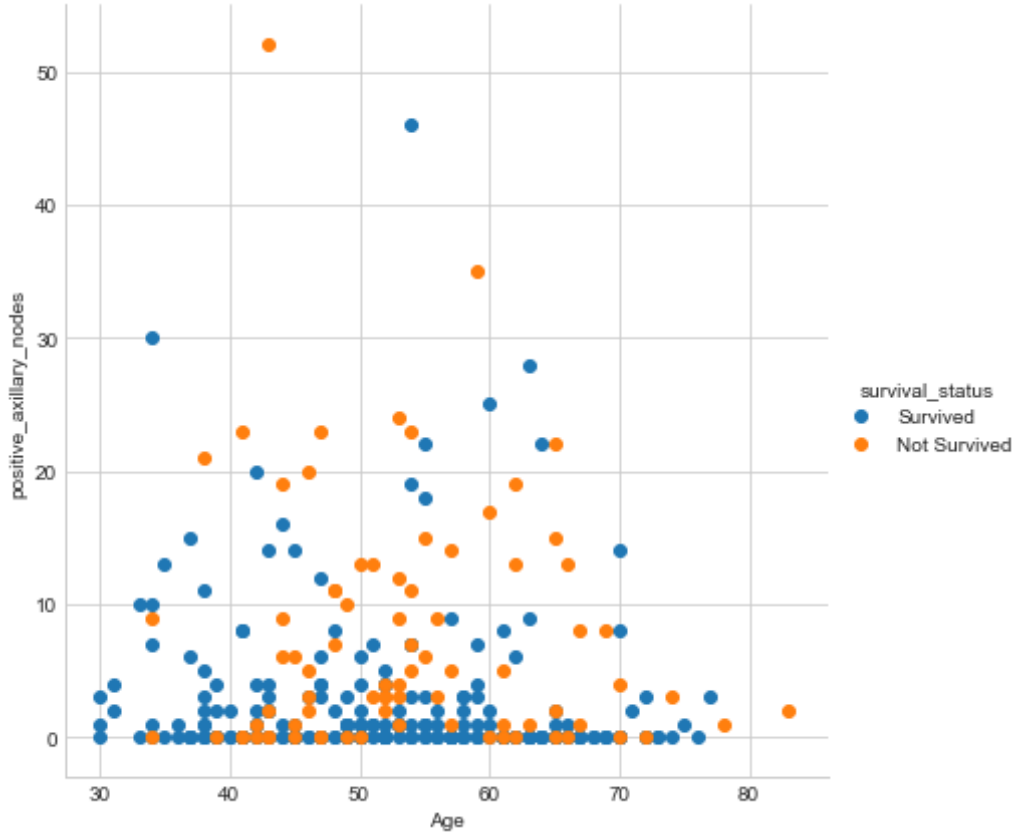
In [33]:
```
sns.set_style("whitegrid");
sns.FacetGrid(data, hue="survival_status", size=6) \
   .map(plt.scatter, "Age", "year") \
   .add_legend();
plt.show();
```

observation

Separating any of the classes does not look as simple. No linear line, in fact, even a curve seems to be incapable of making out a discerning decision boundary that can seperate the two classes.

These two features are insufficient to make any kind of conslusion or deceision.

In [34]:
```
sns.set_style("whitegrid");
sns.FacetGrid(data, hue="survival_status", size=6) \
   .map(plt.scatter, "Age", "positive_axillary_nodes") \
   .add_legend();
plt.show();
```



In [35]:
```
data['positive_axillary_nodes'].value_counts()
```

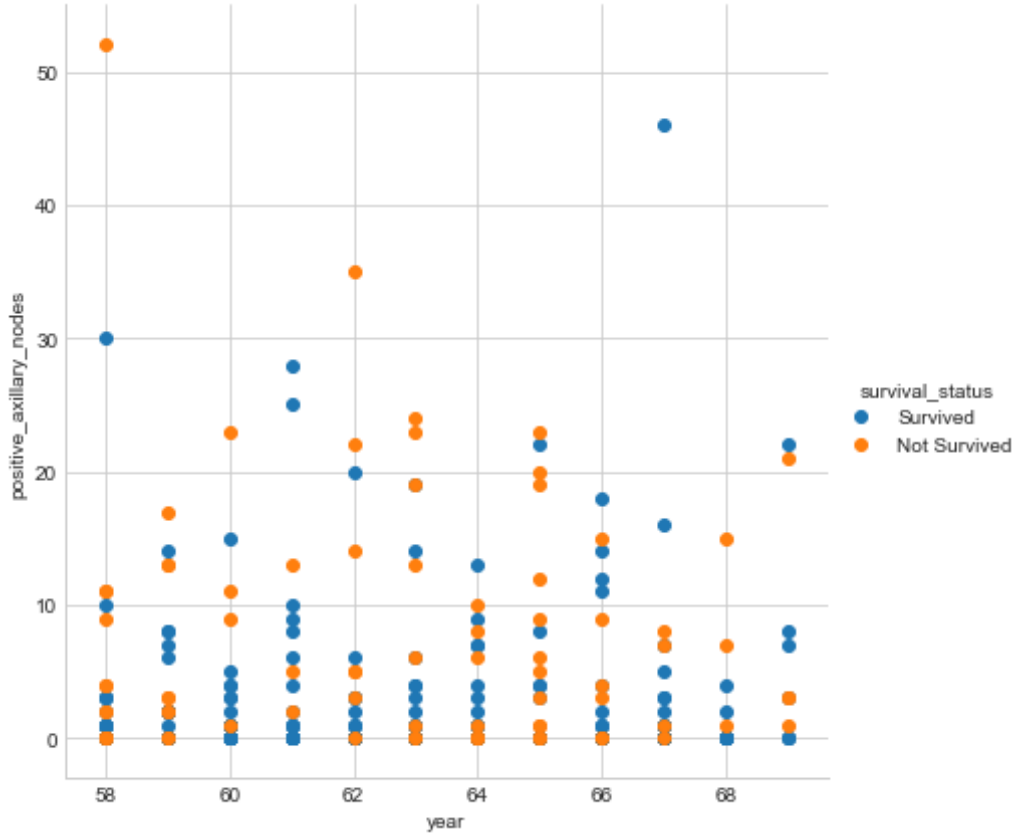
Out[35]:
```
0     136
1      41
2      20
3      20
4      13
6       7
7       7
8       7
5       6
9       6
13      5
14      4
11      4
10      3
15      3
19      3
22      3
23      3
12      2
20      2
46      1
16      1
17      1
18      1
21      1
24      1
25      1
28      1
30      1
35      1
52      1
Name: positive_axillary_nodes, dtype: int64
```

Observation

Since, most of the points on the above plot have value for 'Axillary nodes detected' as 0s and 1s and it is cross verified by printing by actual value counts. Also, from the plot we can say that most of the axillary nodes detected were less than 10 in number, this count decreases even more after 20. After age 70, the number of axillary nodes never crosses 5.

From the plot it is clear that we cannot draw any line/curve that can separate the two classes easily.

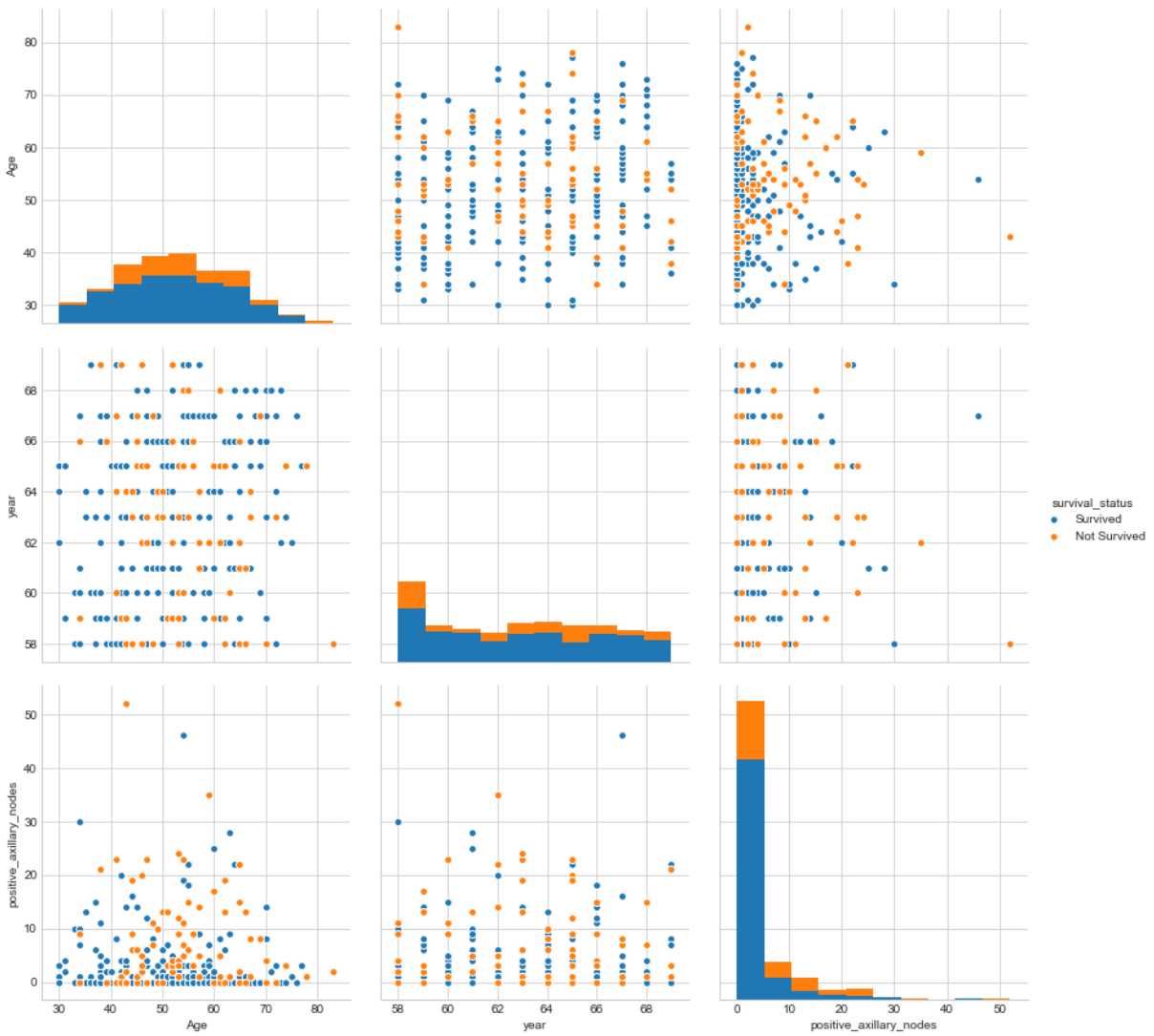
In [36]:
```
sns.set_style("whitegrid");
sns.FacetGrid(data, hue="survival_status", size=6) \
   .map(plt.scatter, "year", "positive_axillary_nodes") \
   .add_legend();
plt.show();
```



observation

Almost both the features overlapped each other, it is really difficult to draw any line/curve that can distinguish between the two classes.

In [37]:
```
plt.close()
sns.set_style('whitegrid')
sns.pairplot(data, hue = 'survival_status', size = 4)
plt.show()
```



Observation

Obtained all the 3 scatter plots that we initally had plotted one-by-one earlier individually.

Finally is difficult to classify which is the most useful feature because of too much overlapping.No two features seem to be able to discern between the two classes.

Conclusion

1. The given dataset is not linearly seprable form each class. There are too much overlapping in the data-points and hence it is very diffucult to classify
2. By plotting all pdf, cdf, box-plot, pair plots, scatter plot etc. we get only one conclusion : if the number of axillary node is less 4 or 5,than survival of patients is more.
3. We need more features and well balanaced dataset to come on very good conlusion.

------ End of Notebook ------