Exploratory Data analysis of Haberman Cancer Survival

Download Haberman Cancer Survival dataset from Kaggle.

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

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

In [2]:
```
#Reading Dataset
df=pd.read_csv('haberman.csv',names=["age","operation_Year","axil_nodes","survival_status"])
df.head()
```

Out[2]:

	age	operation_Year	axil_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 [3]:
```
df.shape
```

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

In our dataset

Number of Rows- 306

Number of columns- 4

Feature or Independent variable- age,operation_Year,axil_nodes

Dependent variable or output label- survival_status

In [4]:
```
# column names in our dataset
print (df.columns)
```

```
Index(['age', 'operation_Year', 'axil_nodes', 'survival_status'], dtype='object')
```

Data points per class

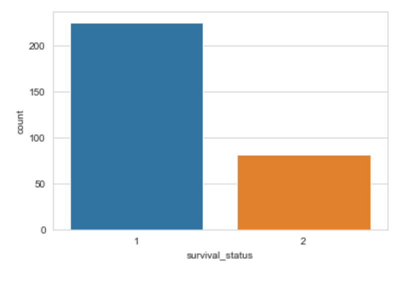
In [5]:
```
df["survival_status"].value_counts()
```

Out[5]:
```
1    225
2     81
Name: survival_status, dtype: int64
```

Data points per class Graphical representation using counter plot

In [6]:
```
sns.set_style('whitegrid')
sns.countplot(x='survival_status',data=df)
```

Out[6]:
```
<AxesSubplot:xlabel='survival_status', ylabel='count'>
```



Objective

The objective for a problem is that on the basis of Feature or Independent variable- age,operation_Year,axil_nodes we have to predict that Dependent variable or output label- survival_status

Survival status (class attribute)

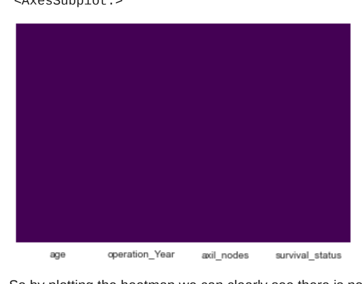
1 = the patient survived 5 years or longer

2 = the patient died within 5 year

Checking for NULL values

In [7]:
```
sns.heatmap(df.isnull(),yticklabels=False,cbar=False,cmap='viridis')
```

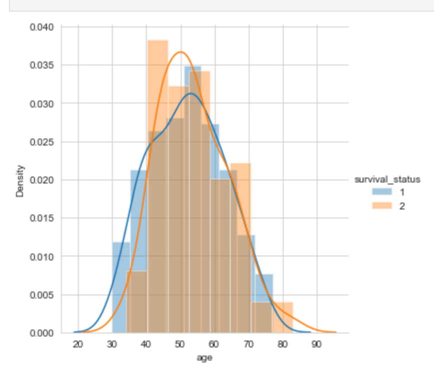
Out[7]:
```
<AxesSubplot:>
```



So by plotting the heatmap we can clearly see there is no missing value in our dataset

Performing Univariate analysis - Plot PDF, CDF, Boxplot, Voilin plots

In [8]:
```
#sns.kdeplot(data=df, x="age", hue="survival_status")
sns.FacetGrid(df, hue="survival_status", size=5) \
   .map(sns.distplot, "age") \
   .add_legend();
plt.show();
```



Observation

1.This forms a large amount of overlapping area when we plot histogram of age

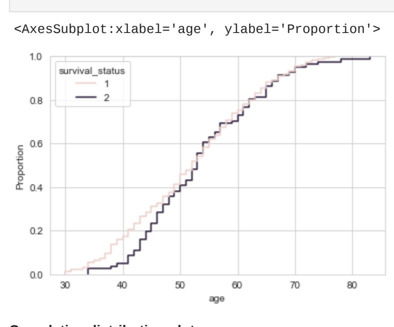
2.Commenting within the range 34 to 75 is difficult as a lot of overlapping.

3.People having age less than 34 survived 5 years or longer

4.People having age greater than 75 died within 5 year

In [9]:
```
sns.ecdfplot(data=df, x="age", hue="survival_status")
```

Out[9]:
```
<AxesSubplot:xlabel='age', ylabel='Proportion'>
```

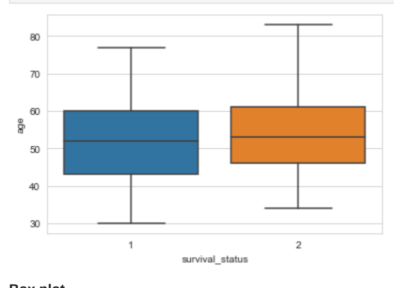


Cumulative distribution plot

1.Shows that the proportion approximately remain the same

2.95% of the people under survival_status 1 and survival_status 2 have age < 70

In [10]:
```
ax = sns.boxplot(x="survival_status", y="age", data=df)
```



Box plot

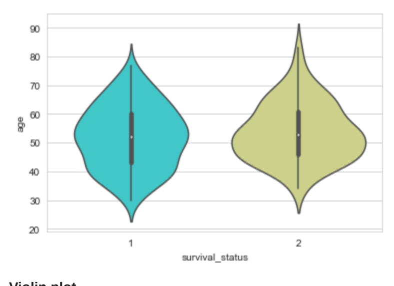
1.Shows there is a lot of overlapping

2.The box plot show the 50 th percentile of both survived states approximately near to 53 years

3.Shows the presence of no outliers

In [11]:
```
sns.violinplot(x="survival_status", y="age", data=df,palette='rainbow')
```

Out[11]:
```
<AxesSubplot:xlabel='survival_status', ylabel='age'>
```

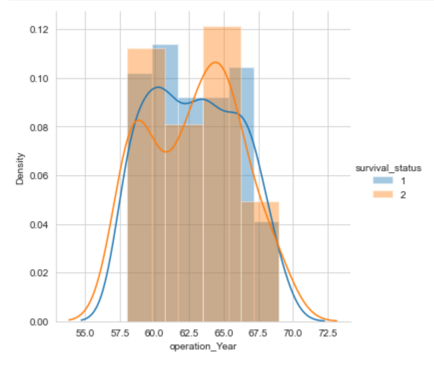


Violin plot

1.Combine the result of pdf and the box plot

2.Shows there is a lot of overlapping

In [12]:
```
#sns.kdeplot(data=df, x="operation_Year", hue="survival_status")
sns.FacetGrid(df, hue="survival_status", size=5) \
   .map(sns.distplot, "operation_Year") \
   .add_legend();
plt.show();
```



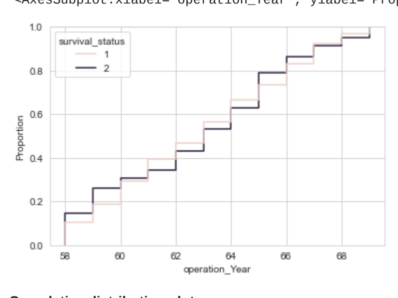
Observation

1.This forms a large amount of overlapping area when we plot histogram of operation year

2.Commenting is difficult as a lot of overlapping.

In [13]:
```
sns.ecdfplot(data=df, x="operation_Year", hue="survival_status")
```

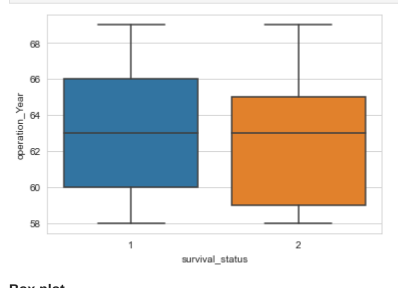
Out[13]:
```
<AxesSubplot:xlabel='operation_Year', ylabel='Proportion'>
```



Cumulative distribution plot

1.90% to 95% of the people under survival_status 1 and survival_status 2 have operation_Year < 68

In [14]:
```
ax = sns.boxplot(x="survival_status", y="operation_Year", data=df)
```



Box plot

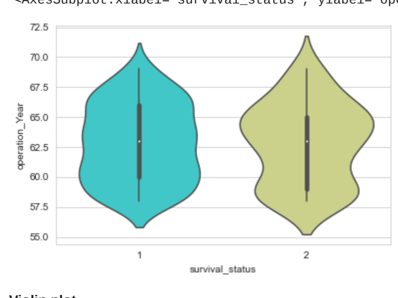
1.Shows there is a lot of overlapping

2.The box plot show the 50 th percentile of both survived states approximately near to 63 years

3.Shows the presence of no outliers

In [15]:
```
sns.violinplot(x="survival_status", y="operation_Year", data=df,palette='rainbow')
```

Out[15]:
```
<AxesSubplot:xlabel='survival_status', ylabel='operation_Year'>
```

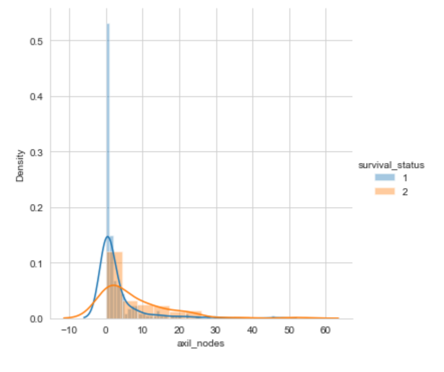


Violin plot

1.Combine the result of pdf and the box plot

2.shows there is a lot of overlapping

In [16]:
```
#sns.kdeplot(data=df, x="axil_nodes", hue="survival_status")
sns.FacetGrid(df, hue="survival_status", size=5) \
   .map(sns.distplot, "axil_nodes") \
   .add_legend();
plt.show();
```



Observation

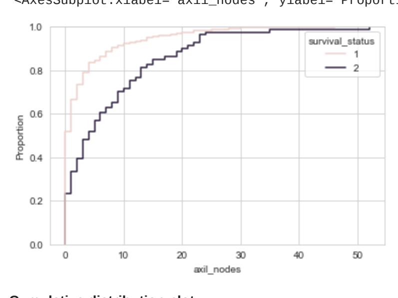
1.This forms a large amount of overlapping area when we plot histogram of axil nodes

2.We can comment that when axil nodes nearly 0 the density of people survived more than 5 years is more.

3.But from 5 axil node the pdf of survival state 2 is more means more pobability of that person died within 5 years

In [17]:
```
sns.ecdfplot(data=df, x="axil_nodes", hue="survival_status")
```

Out[17]:
```
<AxesSubplot:xlabel='axil_nodes', ylabel='Proportion'>
```

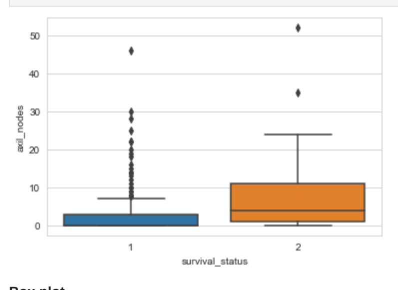


Cumulative distribution plot

1.80% of the people under survival_status 1 have axil_nodes less than or equal to 5.

2.80% of the people under survival_status 2 have axil_nodes less than 14.

In [18]:
```
ax = sns.boxplot(x="survival_status", y="axil_nodes", data=df)
```



Box plot

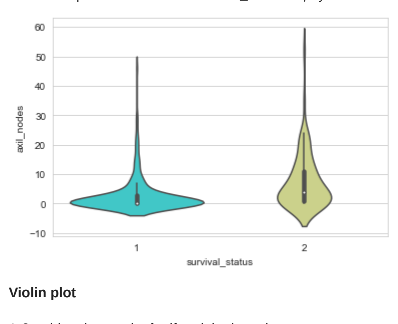
1.Shows the presence of outliers in survived state 1 and few in survived state 2.

2.75th percentile of survived people have axial_nodes less than 5

3.50th percentile of survived people not more than 5 years have axial_nodes less than 5

In [19]:
```
sns.violinplot(x="survival_status", y="axil_nodes", data=df,palette='rainbow')
```

Out[19]:
```
<AxesSubplot:xlabel='survival_status', ylabel='axil_nodes'>
```



Violin plot

1.Combine the result of pdf and the box plot

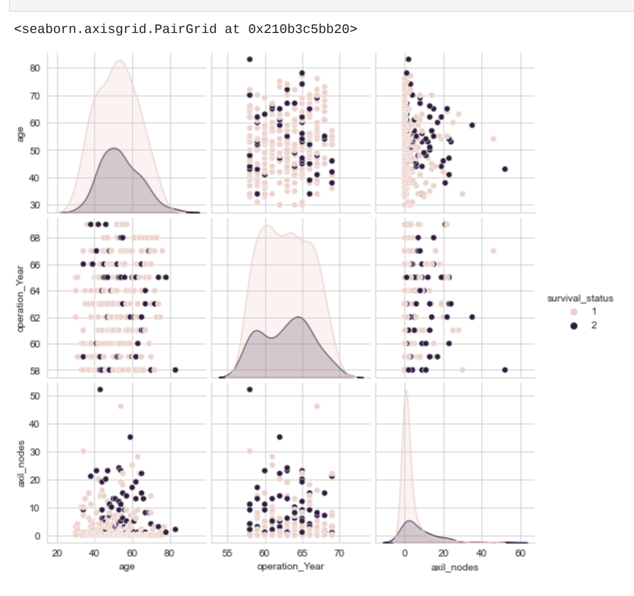
2.Most of the Patient survived have axil_node nearly one

Performing Bivariate analysis - Plot 2D Scatter plots and Pair plots

Pair plot

In [20]:
```
sns.pairplot(df,hue='survival_status')
```

Out[20]:
```
<seaborn.axisgrid.PairGrid at 0x210b3c5bb20>
```

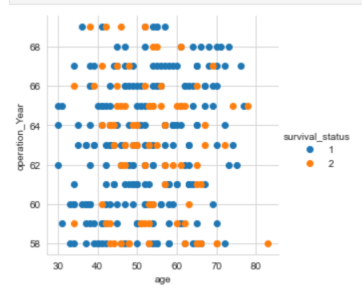


Observations

1. The survival_status is very difficult to find with the help of these three independent features age,operation_Year,axil_nodes.
2. As by plotting garphs between them in all the cases there is a lot of overlapping between them.
3. Age and axil_nodes are the most useful features to identify survival_status from others but it also contains a lot of overlapping.

2D Scatter plots

In [21]:
```
sns.set_style("whitegrid");
sns.FacetGrid(df, hue="survival_status", height=4) \
   .map(plt.scatter, "age", "operation_Year") \
   .add_legend();
plt.show();
```

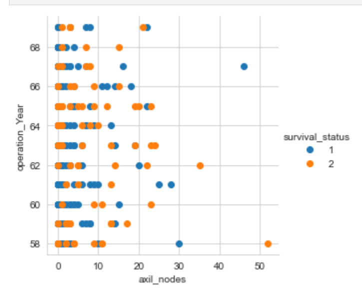


Observation

1.This forms a large amount of overlapping area when we plot scatterplot of operation_year and age

2.Commenting is difficult as a lot of overlapping.

In [22]:
```
sns.set_style("whitegrid");
sns.FacetGrid(df, hue="survival_status", height=4) \
   .map(plt.scatter, "axil_nodes", "operation_Year") \
   .add_legend();
plt.show();
```

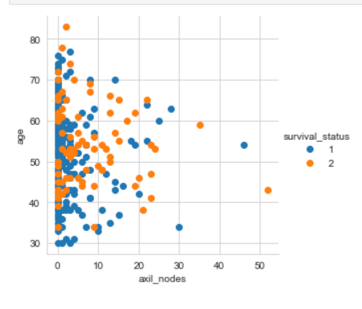


Observation

1.This forms a large amount of overlapping area when we plot scatterplot of year and axial_nodes

2.Commenting is difficult as a lot of overlapping.

In [23]:
```
sns.set_style("whitegrid");
sns.FacetGrid(df, hue="survival_status", height=4) \
   .map(plt.scatter, "axil_nodes", "age") \
   .add_legend();
plt.show();
```



Observation

1.This forms a large amount of overlapping area when we plot scatter plot of operation age and axial_nodes.

2.Commenting is difficult as a lot of overlapping.

Final Conclusions

1. The survival_status is very difficult to find with the help of these three independent features age, operation_Year, axil_nodes.
2. As by plotting garphs between them in all the cases there is a lot of overlapping between them 80% to 90%.
3. Age and axil_nodes is 25% more useful features to identify survival_status from others but it also contains a lot of overlapping.
4. Order of useful features axil_nodes > Operation_year > Age.
5. A non linear technique will be required to differentiate between the survival_status .
6. More useful features should be collected for the determination of survival_status.