Exploratory_Data_Analysis

March 19, 2019

0.1 Import all required packages

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        sns.set()
```

1 Configs

```
In [2]: df_path = './Data/haberman.csv'
```

2 Dataset Descriptions

```
In [3]: df = pd.read_csv(df_path, index_col=False, names=['Age', 'Op_Year', 'Num_Nodes', 'Surv_S
        df.head()
```

Out[3]:

	Age	Op_Year	Num_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

The 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.

Age
Age of patient at time of operation (numerical)
Op_Year
Patient's year of operation (year - 1900, numerical)
Num_Nodes
Number of positive axillary nodes detected (numerical)
Surv_Status
Survival status (class attribute)
-- 1 = the patient survived 5 years or longer
-- 2 = the patient died within 5 year

2.1 Objective

Identify the useful features which helps us in classifying the patient survied or not

3 High level statistics of the dataset

```
In [4]: print('Number of data points :', df.shape[0])
        print('Number of columns/ features :', df.shape[1])
        print('Number of clasess :', len(df['Surv_Status'].unique()))
        value_counts = df['Surv_Status'].value_counts()
        print('number of data points per class 1 & 2')
        print('Class 1 size: ',value_counts[1], '\tClass 2 size: ', value_counts[2])
```

```
Number of data points : 306
Number of columns/ features : 4
Number of clasess : 2
number of data points per class 1 & 2
Class 1 size:  225         Class 2 size:  81
```

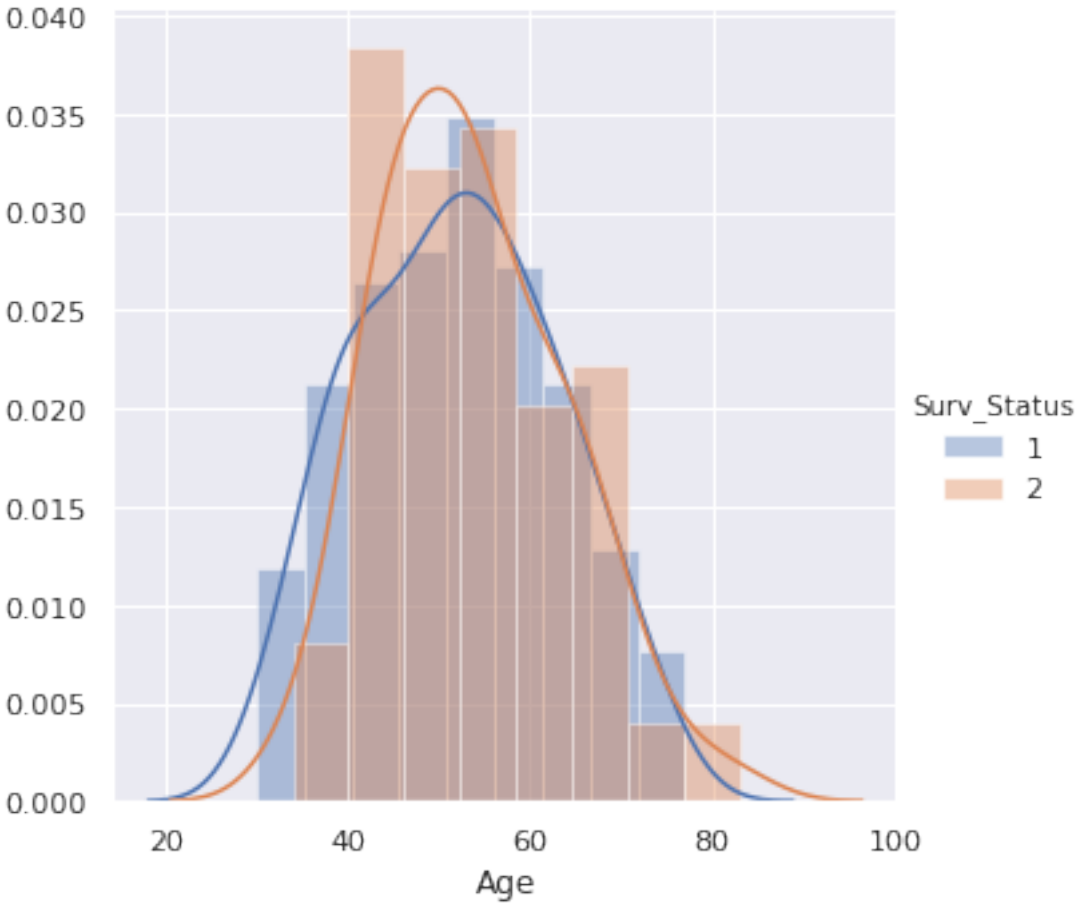
Observation
Dataset is highly imbalanced survived : not survived = 25:9

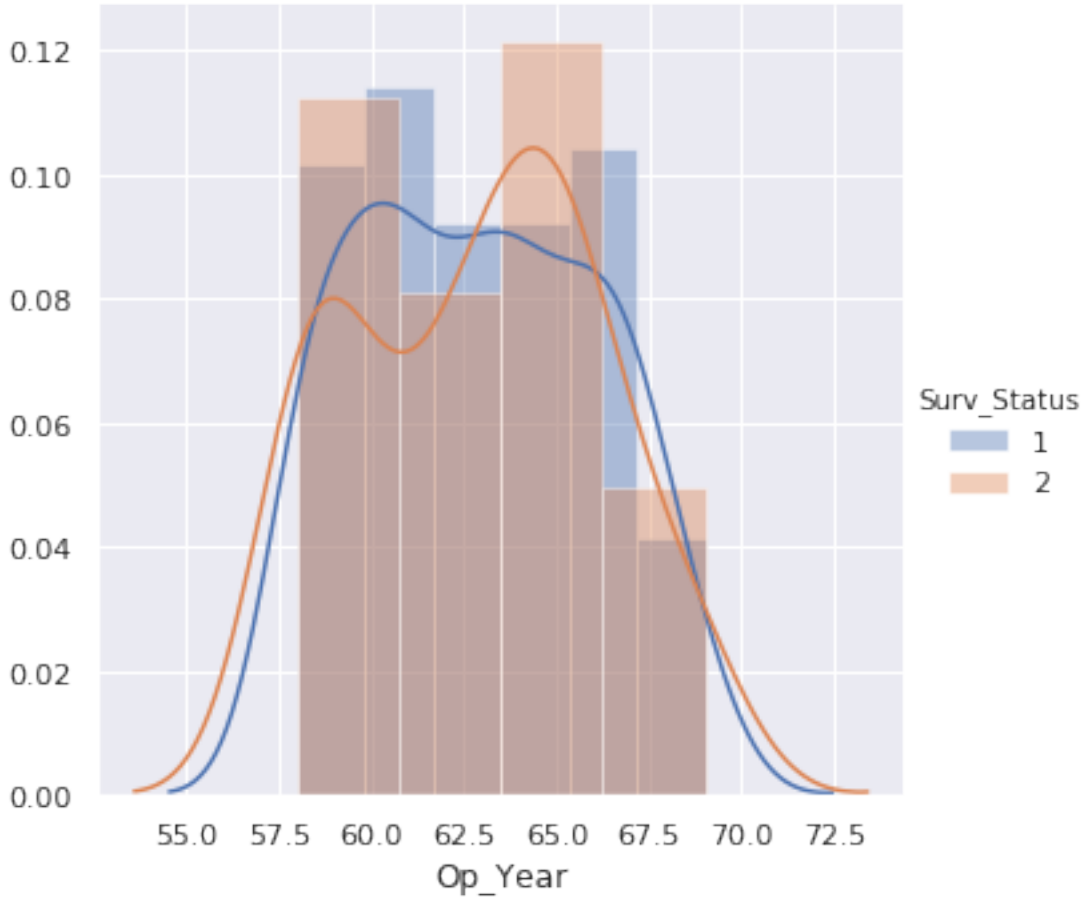
4 Univaraite analysis

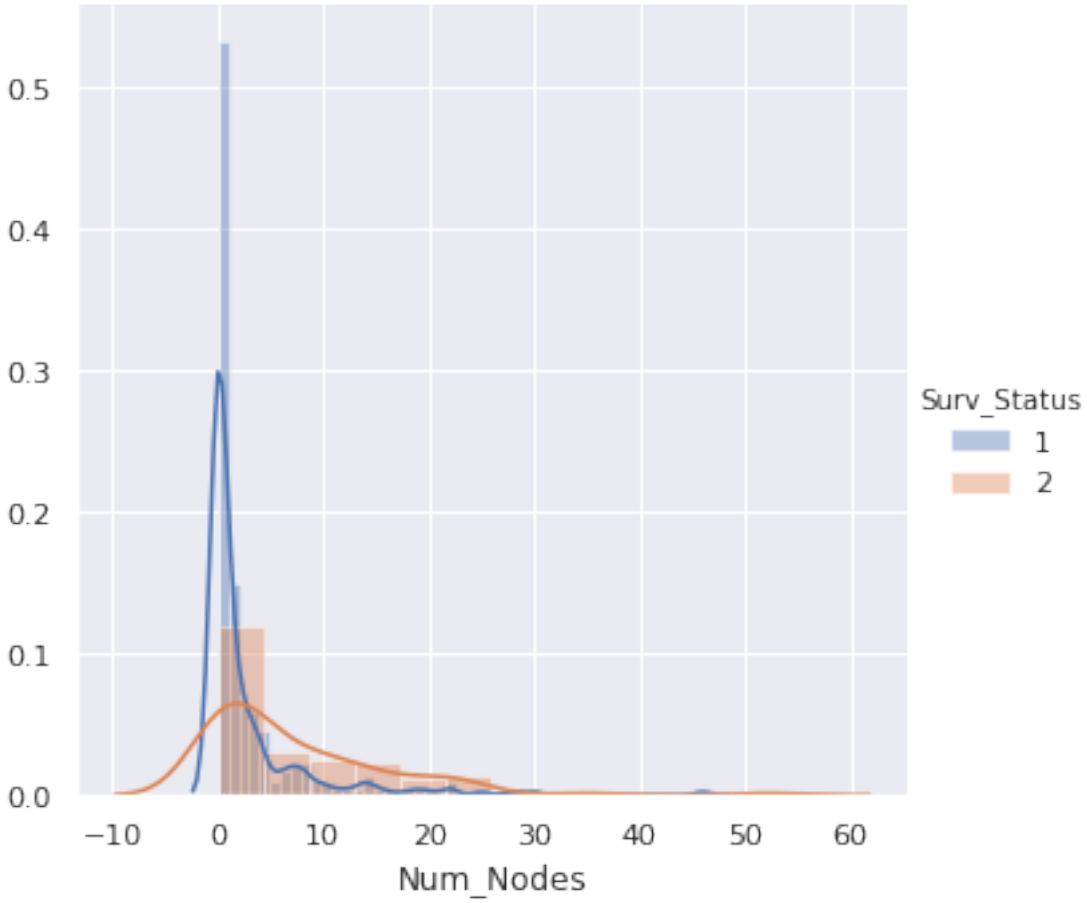
4.1 a) PDF plot

```
In [5]: sns.FacetGrid(df, hue='Surv_Status', height=5).map(sns.distplot, 'Age').add_legend();
        plt.show();

        sns.FacetGrid(df, hue='Surv_Status', height=5).map(sns.distplot, 'Op_Year').add_legend()
        plt.show();

        sns.FacetGrid(df, hue='Surv_Status', height=5).map(sns.distplot, 'Num_Nodes').add_legend
        plt.show();
```

```
/home/nisheels/anaconda3/lib/python3.6/site-packages/scipy/stats/stats.py:1713: FutureWarning: U
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```


0.040
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
20
40
60
80
100
Age
Surv_Status
1
2


0.12
0.10
0.08
0.06
0.04
0.02
0.00
55.0
57.5
60.0
62.5
65.0
67.5
70.0
72.5
Op_Year
Surv_Status
1
2



Observations

There is considerable overlap between the distributions of survived and not survived for all features

For not survived class the Num_Nodes feature has slighly higher value when compared with survived one as there is a slight shift towardds the right for its peak position

4.2 b) PDF & CDF plot

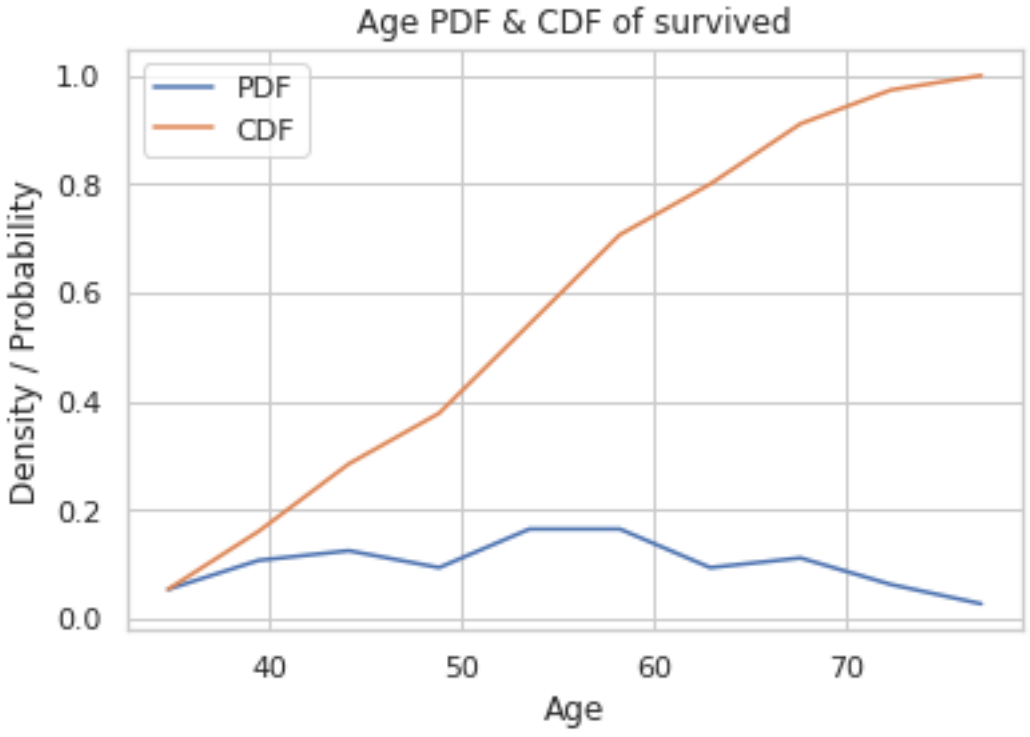
```
In [6]: df_survived = df[df['Surv_Status'] == 1 ]
        df_not_survived = df[df['Surv_Status'] == 2]
        features_list = ['Age', 'Op_Year', 'Num_Nodes']

In [19]: def plot_distribution(df_temp, feat_name, class_name):
             # get histogram from the data frame
             counts, bin_edges = np.histogram(df_temp[feat_name], bins=10, density=True)

             # plot PDF and CDF of the distribution in single image
             pdf = counts / sum(counts)
             cdf = np.cumsum(pdf)
```

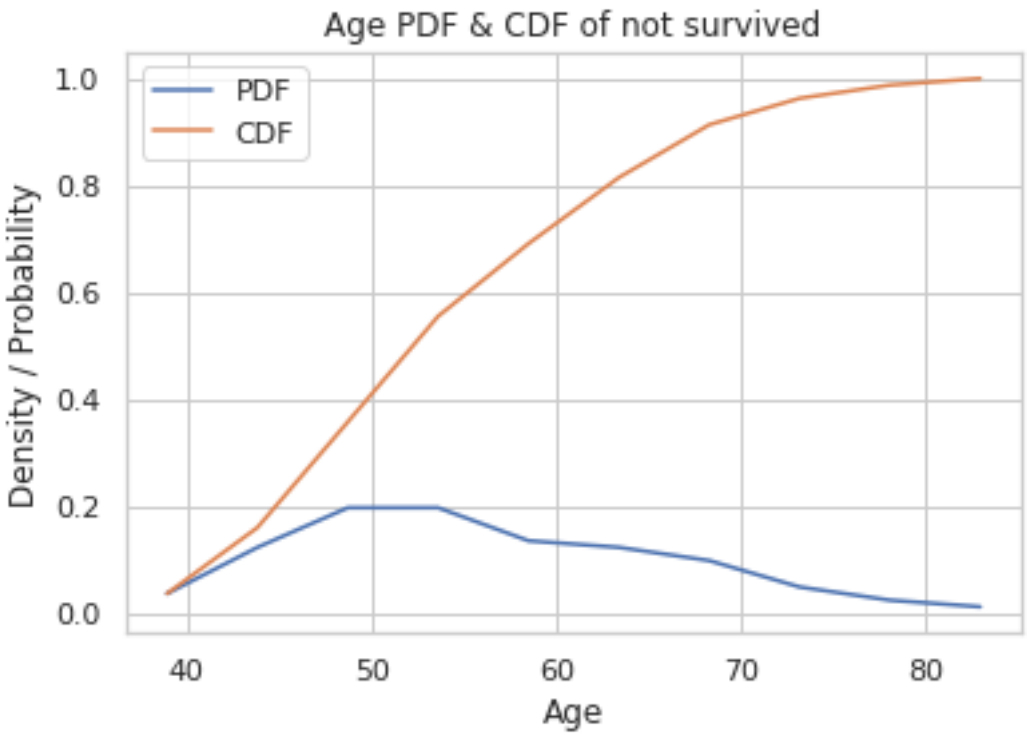
```
    plt.plot(bin_edges[1:], pdf, label='PDF')
    plt.plot(bin_edges[1:], cdf, label='CDF')

    # set title, labels to axes
    plt.xlabel(feat_name)
    plt.ylabel('Density / Probability')
    plt.title(feat_name + ' PDF & CDF of '+ class_name)

    # add legend
    plt.legend()

    # display the image
    plt.show()
```

4.2.1 Plot for survied data

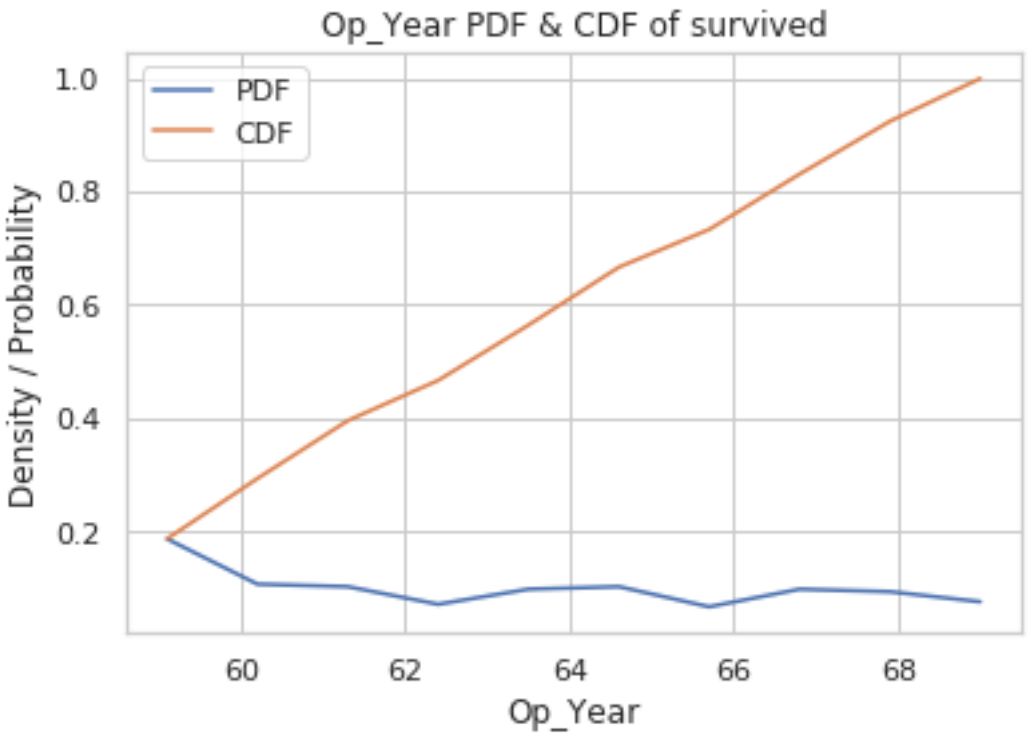
```
In [31]: for feat_name in features_list:
             print('=' * 30 + "\t " + feat_name + '\tPDF & CDF\t' + '=' * 30)
             plot_distribution(df_survived, feat_name, 'survived')
             plot_distribution(df_not_survived, feat_name, 'not survived')
             print('=' * 100)
```

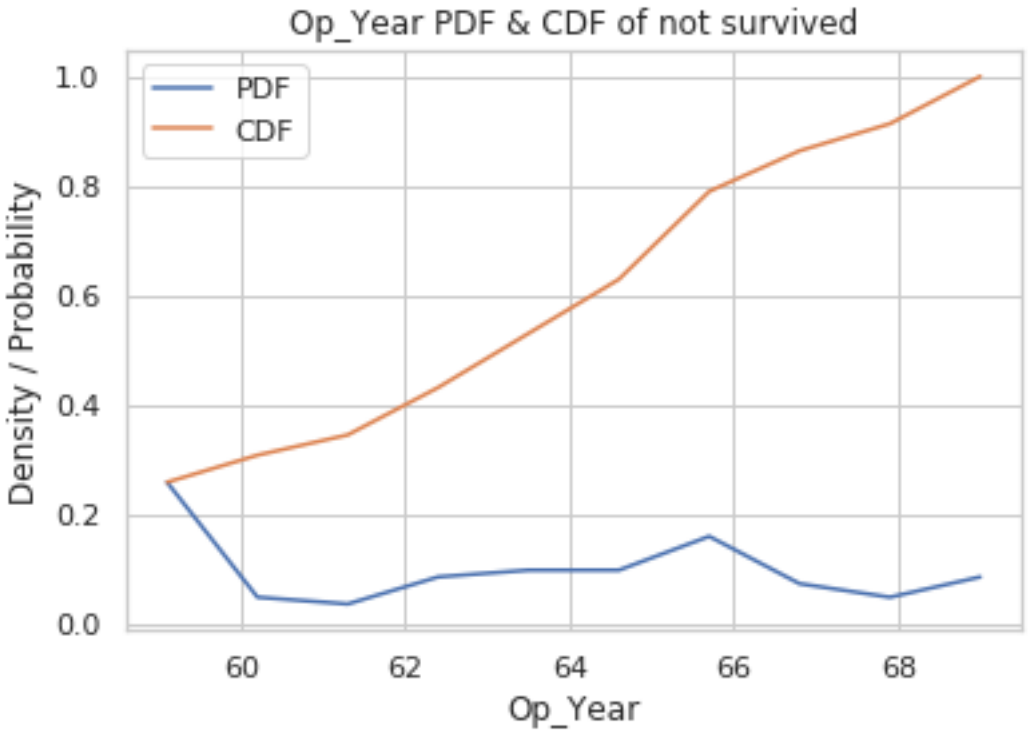
```
==============================          Age         PDF & CDF         ==============================
```



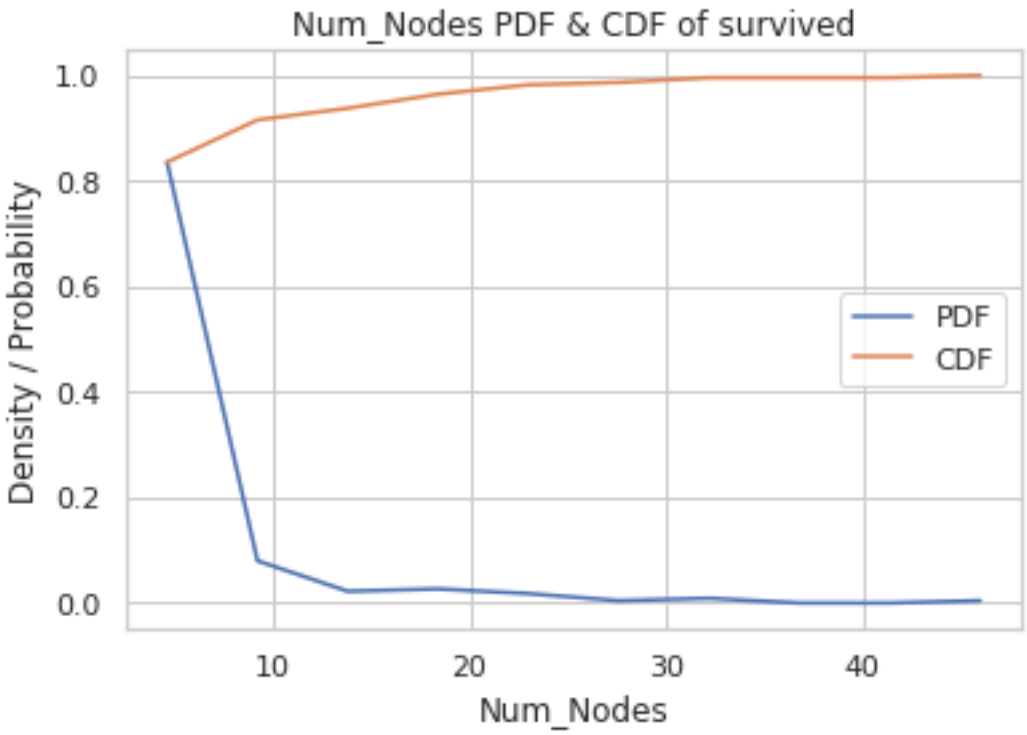


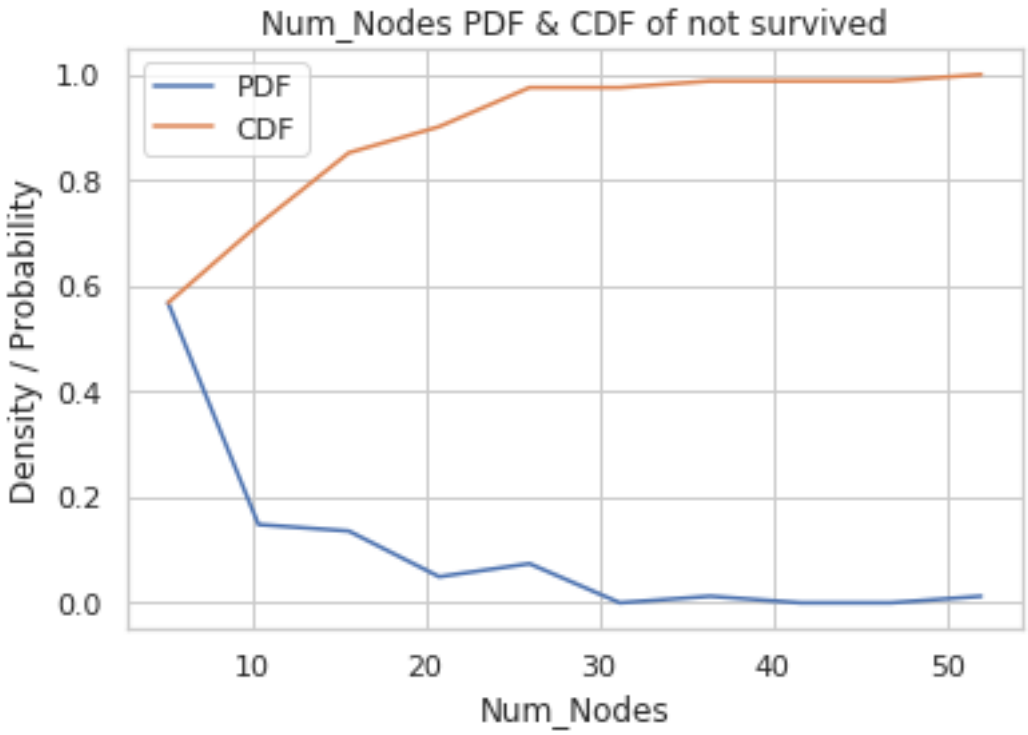

```
================================================================================================
==============================          Op_Year         PDF & CDF         ========================
```


Op_Year PDF & CDF of survived
PDF
CDF
Density / Probability
1.0
0.8
0.6
0.4
0.2
60
62
64
66
68
Op_Year


Op_Year PDF & CDF of not survived
PDF
CDF
Density / Probability
1.0
0.8
0.6
0.4
0.2
0.0
60
62
64
66
68
Op_Year

```
================================================================================================
================================  Num_Nodes  PDF & CDF  ========================
```



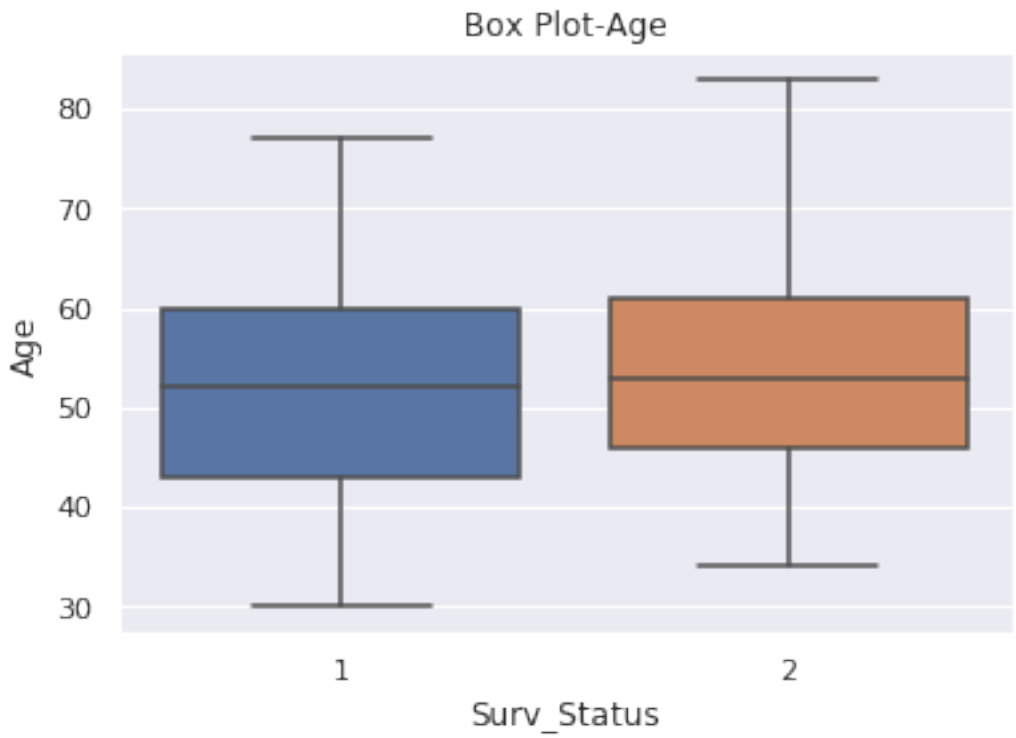


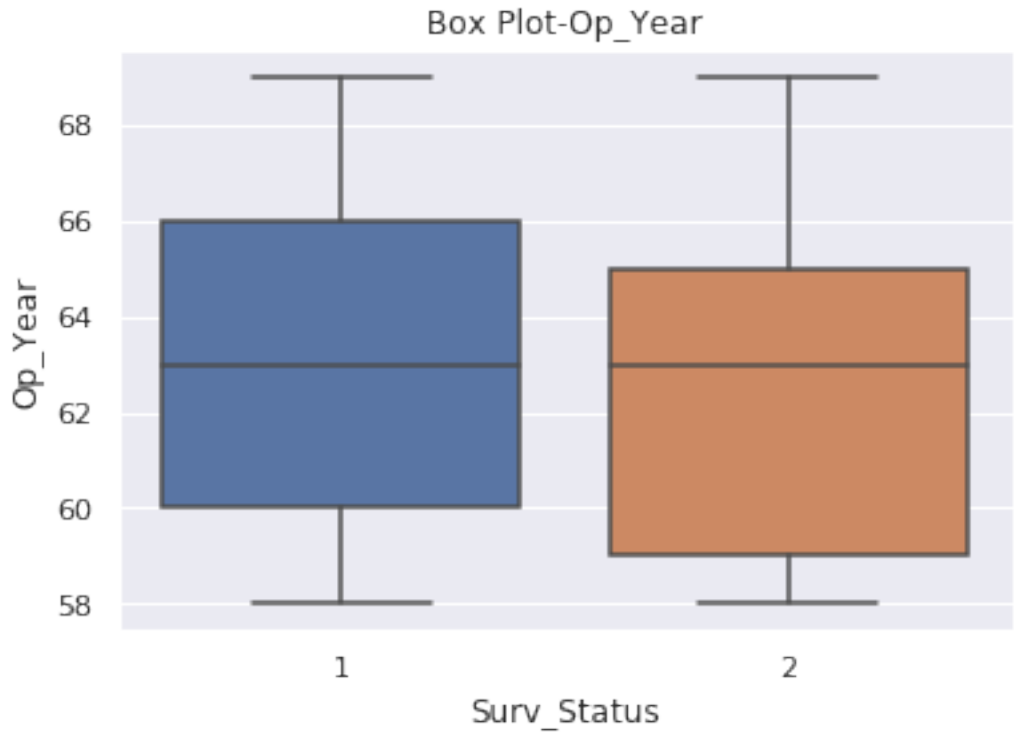
==========

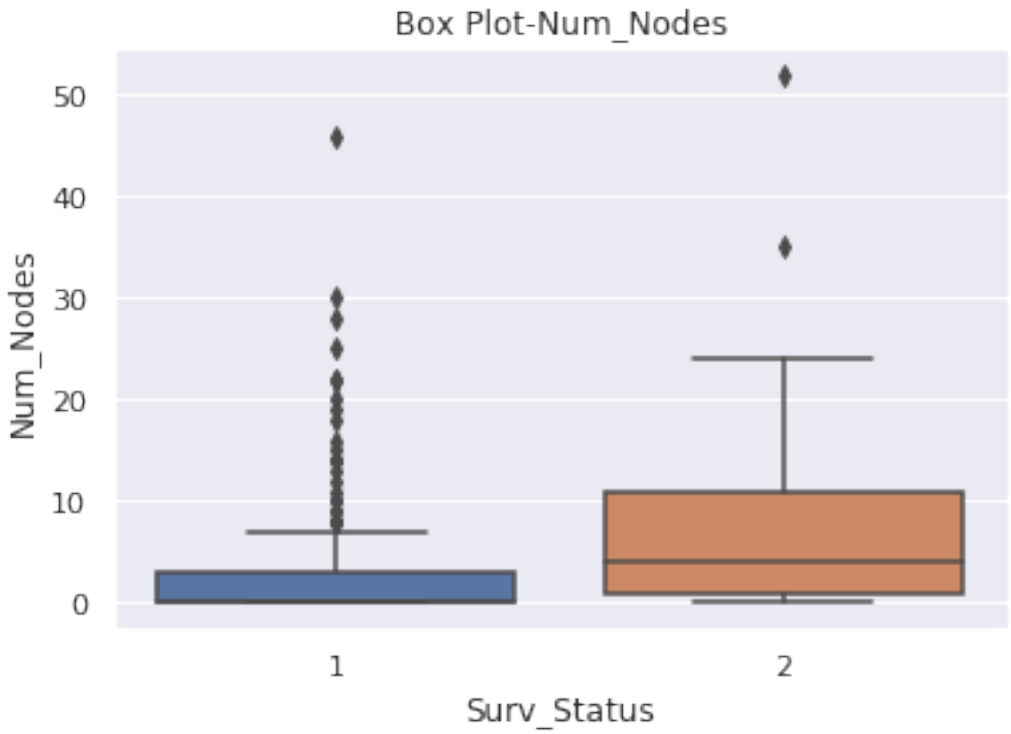
Observation
There are no significant differences in PDF & CDF between the clasess for any feature

4.3 C) Box plots

```
In [10]: for feat_name in features_list:
             sns.boxplot(x='Surv_Status', y=feat_name, data=df)
             plt.title('Box Plot-' + feat_name)
             plt.show()
```


Box Plot-Age
Age
80
70
60
50
40
30
1
2
Surv_Status


Box Plot-Op_Year
Op_Year
68
66
64
62
60
58
1
2
Surv_Status

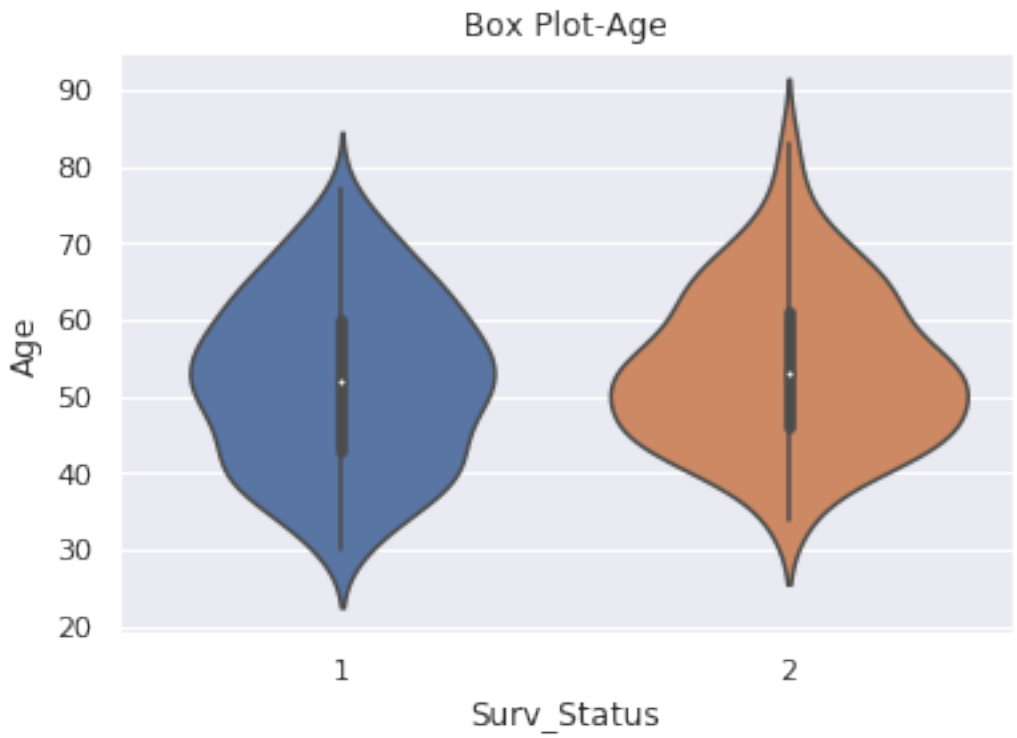


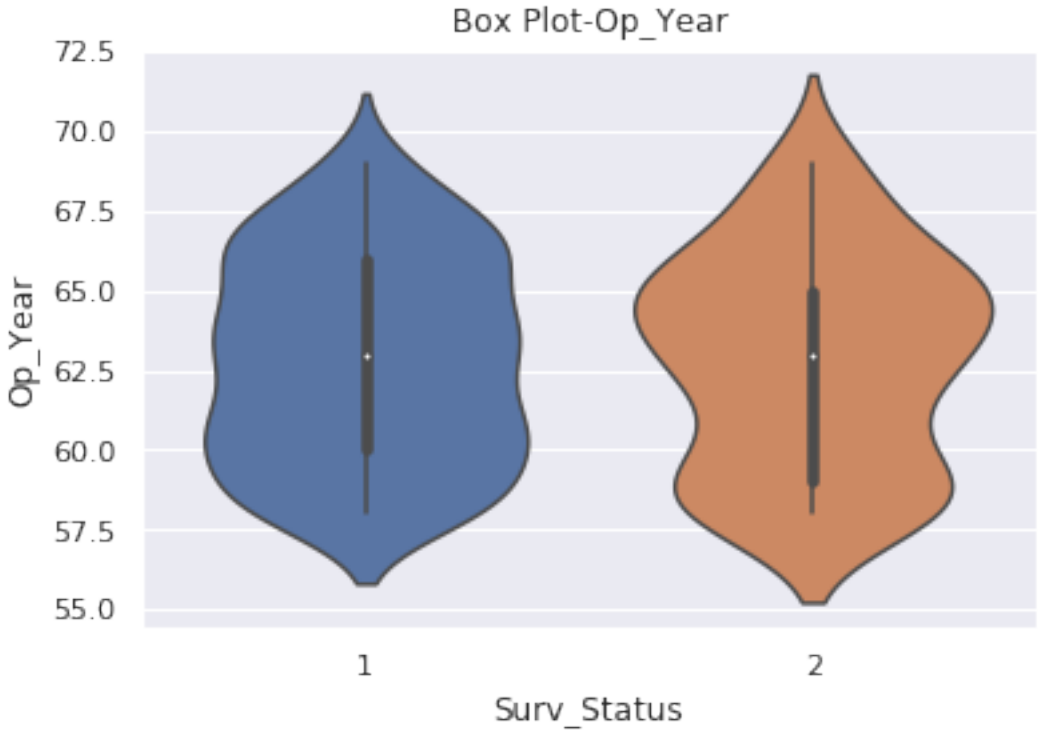
Observations
There is considerable difference between medians of the feature Num_nodes
IQR of not survived is much broader compared to survived
The Num nodes feature may be really useful in improving prediction
The number of outliers for survived seems bit high

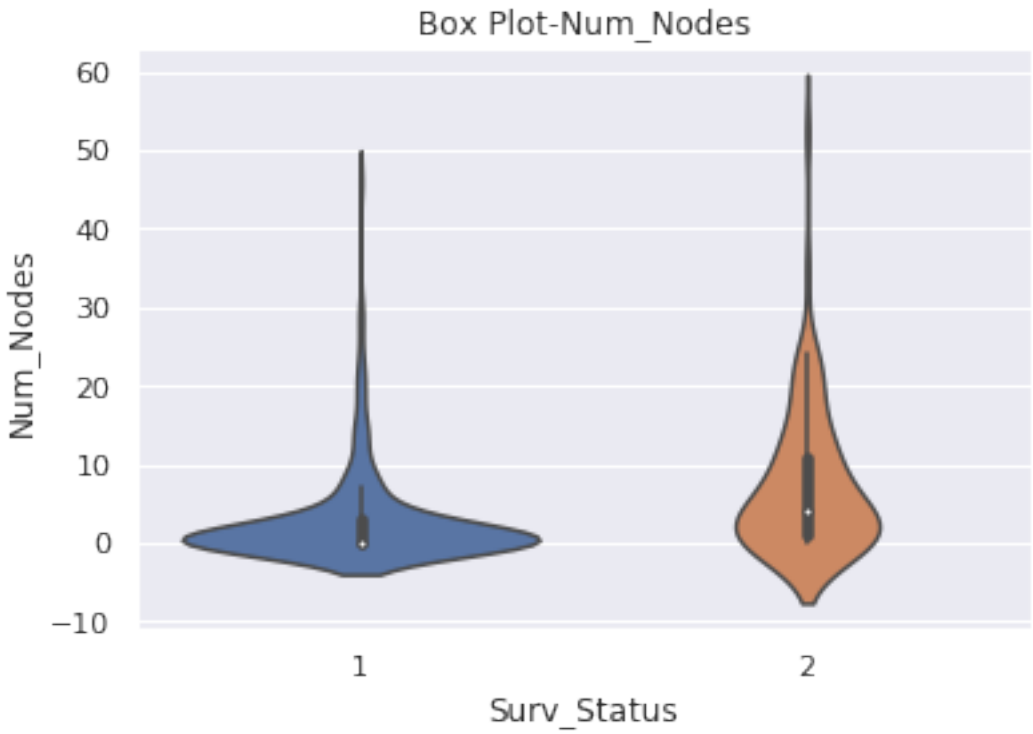
4.3.1 D) Violin Plots

```
In [11]: for feat_name in features_list:
             sns.violinplot(x='Surv_Status', y=feat_name, data=df)
             plt.title('Box Plot-' + feat_name)
             plt.show()
```

```
/home/nisheels/anaconda3/lib/python3.6/site-packages/scipy/stats/stats.py:1713: FutureWarning: U
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```


Box Plot-Age
90
80
70
60
50
40
30
20
Age
1
2
Surv_Status


Box Plot-Op_Year
72.5
70.0
67.5
65.0
62.5
60.0
57.5
55.0
Op_Year
1
2
Surv_Status



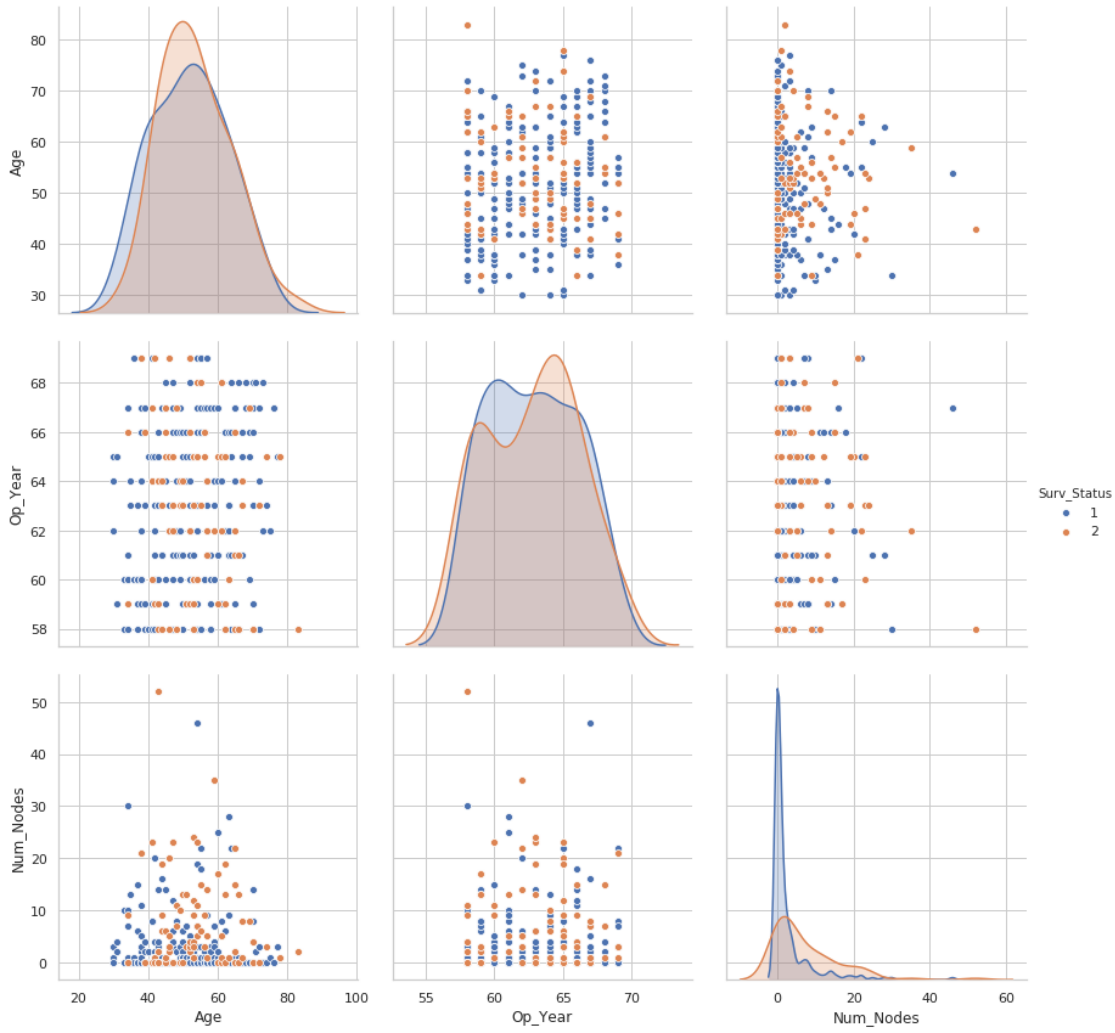
Observations
The num_nodes feature is highly right skewed
The density of Num nodes feature is much concentrated around zero for survived class

5 Multivariate Analysis

5.1 A) Pairwise Scatter Plots

```
In [12]: plt.close();
         sns.set_style('whitegrid');
         sns.pairplot(df, hue='Surv_Status', vars=features_list, height=4);
         plt.show()
```

```
/home/nisheels/anaconda3/lib/python3.6/site-packages/scipy/stats/stats.py:1713: FutureWarning: U
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```



Observations
Scatter plots shows there is no correlation between any two features

6 Contour Plots

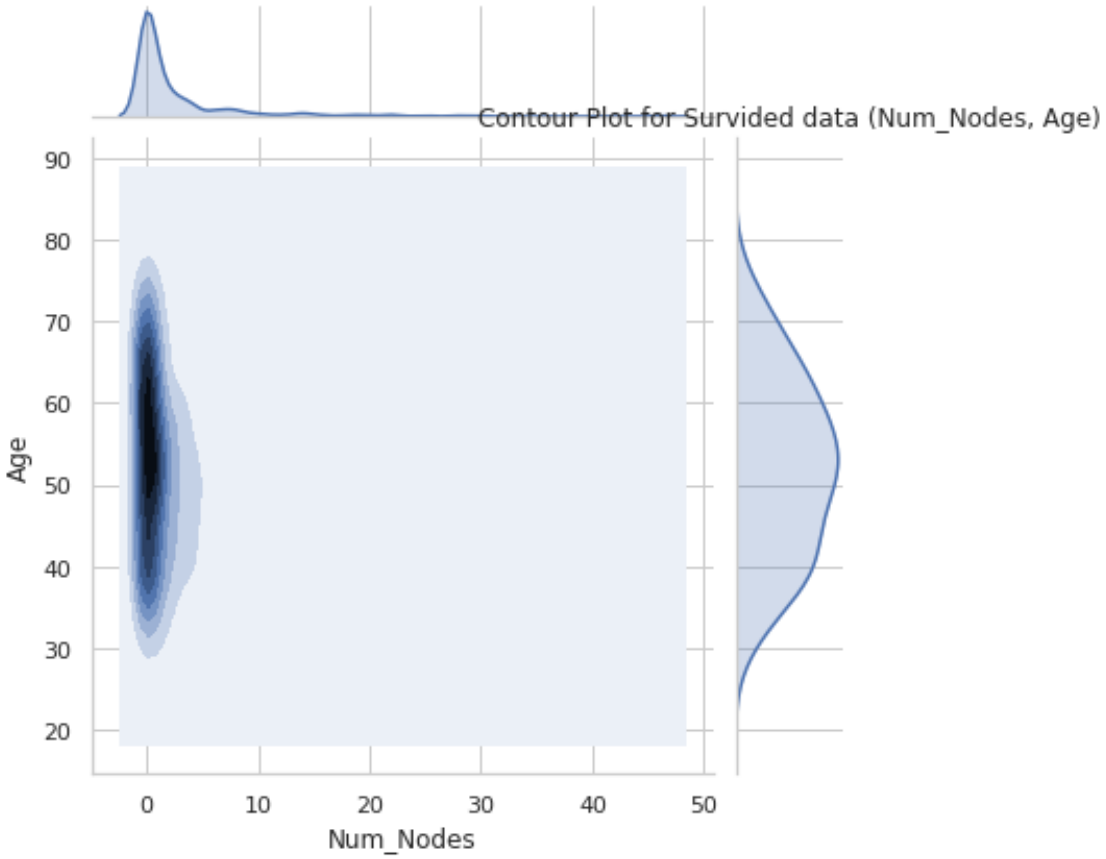
```
In [13]: # set a list of feature combinations
         feat_combinations = [('Num_Nodes', 'Age',), ('Op_Year', 'Age',), ('Num_Nodes', 'Op_Year
```

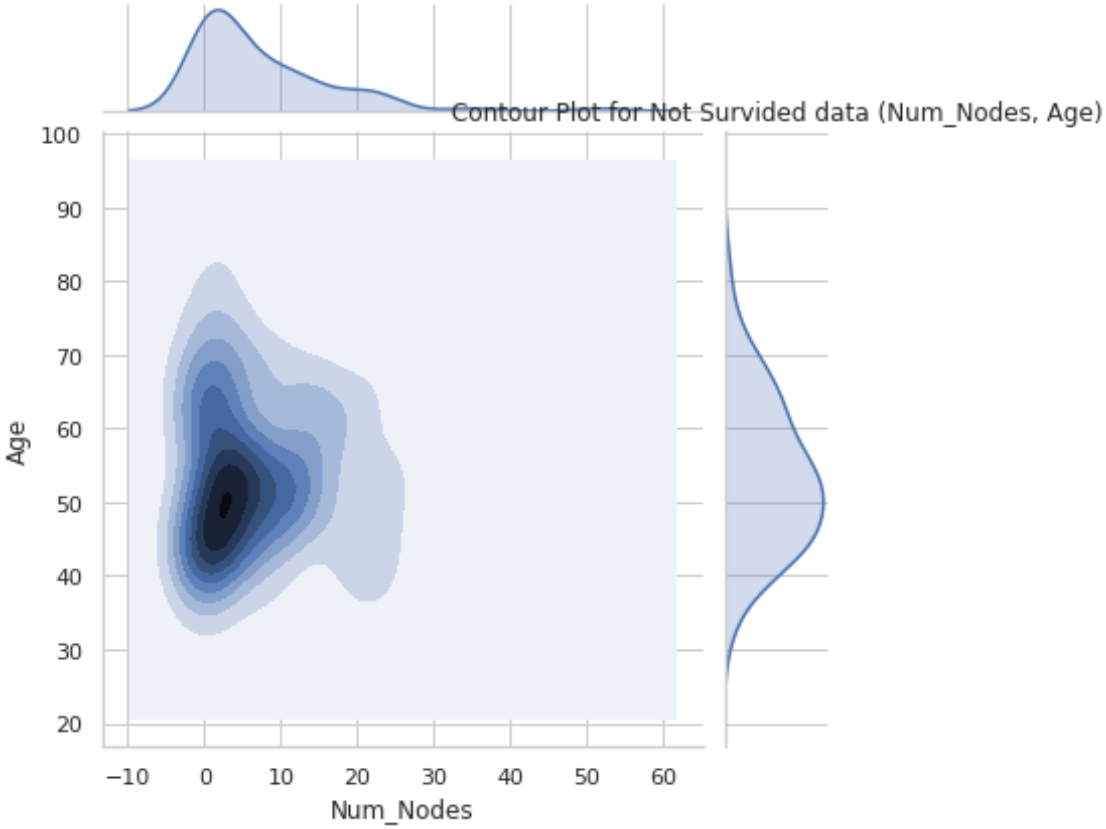
```
In [33]: #2D Density plot, contors-plot
         for feature_combs in feat_combinations:
             print('=' * 30 + '\t' + feature_combs[0] + '\t' + feature_combs[1] + '\t Contour Pl
             sns.jointplot(x=feature_combs[0], y=feature_combs[1], data=df_survived, kind="kde")
             plt.title('Contour Plot for Survived data (%s, %s)'%feature_combs)
             plt.show();
             sns.jointplot(x=feature_combs[0], y=feature_combs[1], data=df_not_survived, kind="k
             plt.title('Contour Plot for Not Survived data (%s, %s)'%feature_combs)
```

```
            plt.show();
            print('='*100)
```

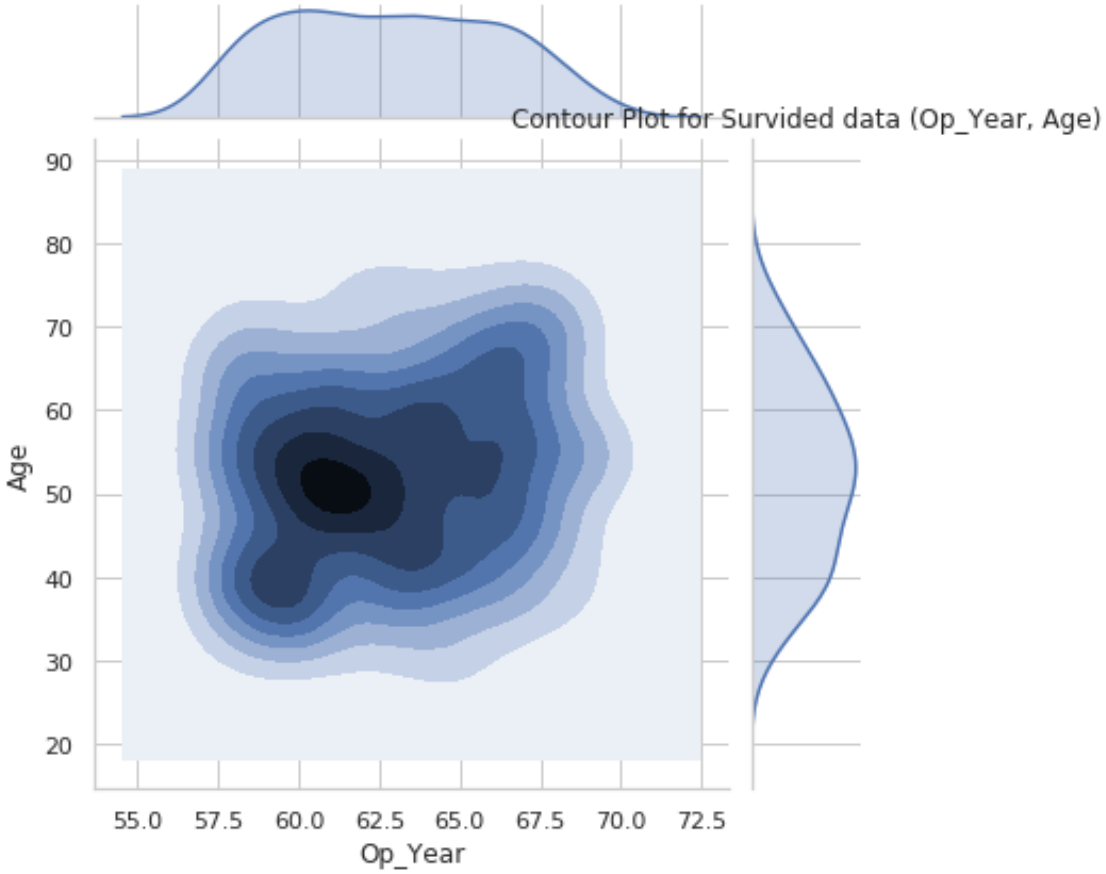
```
==============================        Num_Nodes        Age        Contour Plot        ==========
```

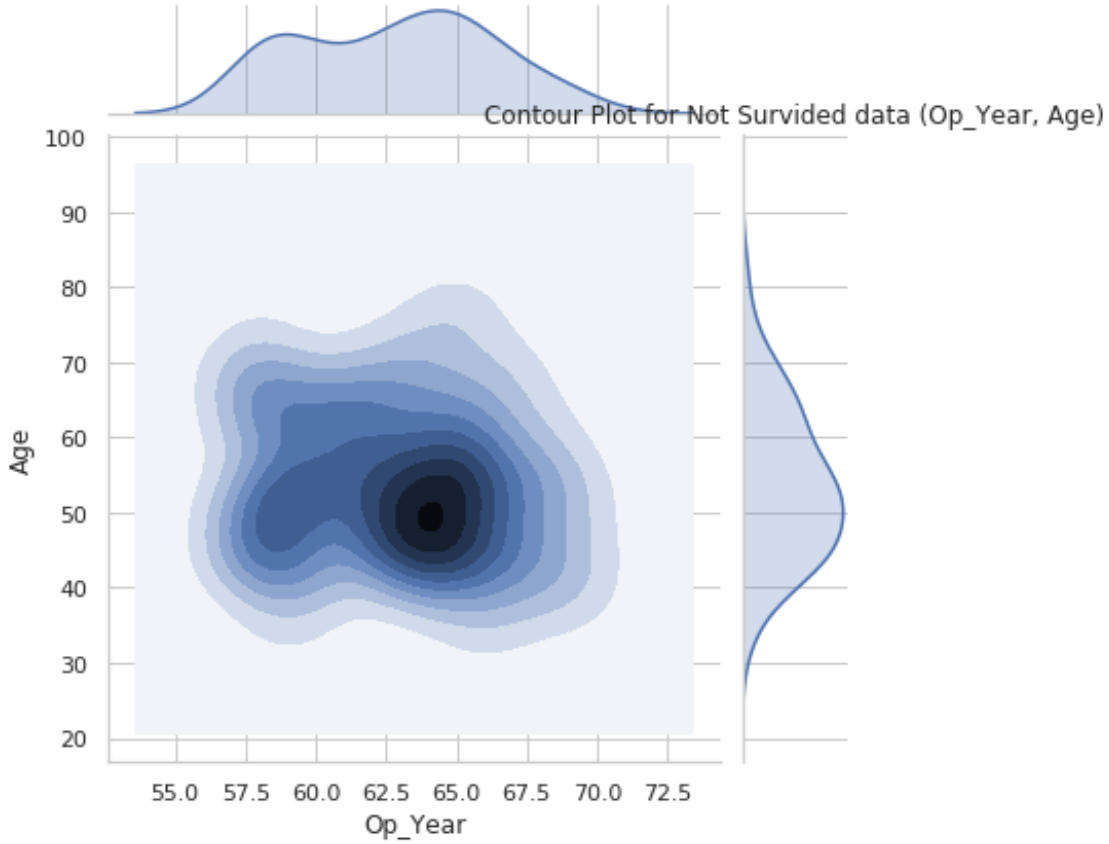
```
/home/nisheels/anaconda3/lib/python3.6/site-packages/scipy/stats/stats.py:1713: FutureWarning: U
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```



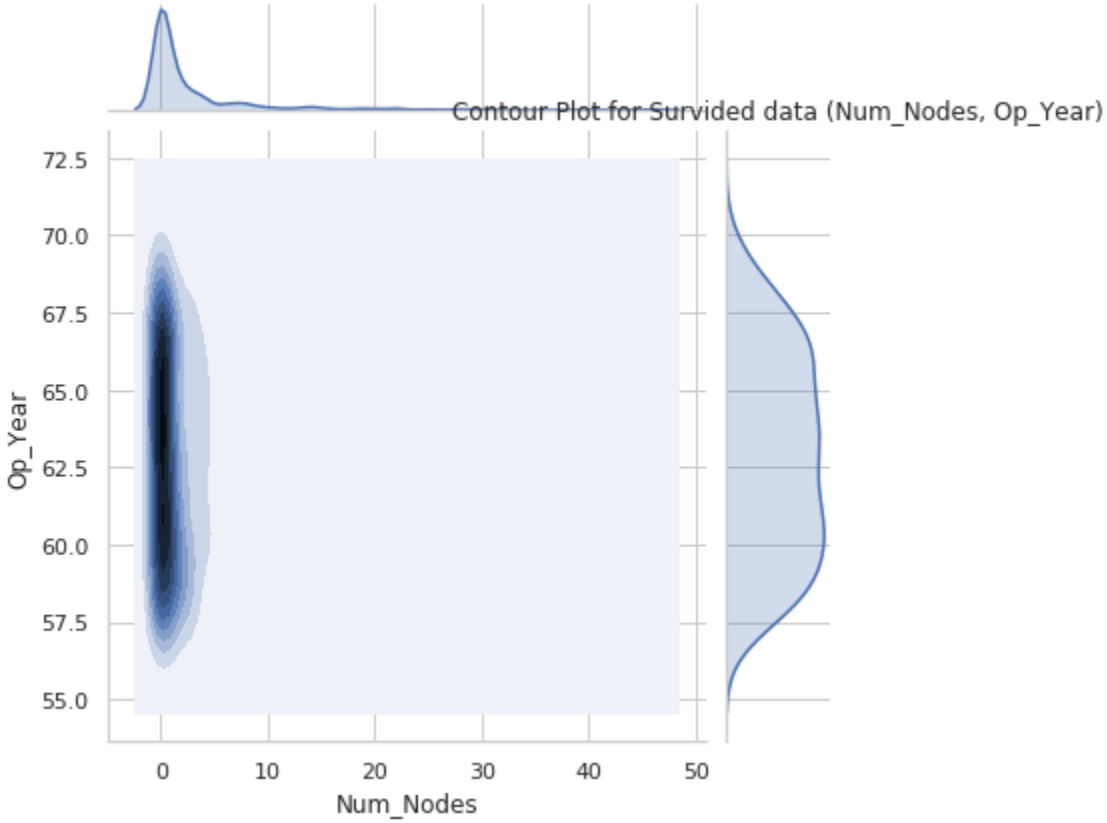


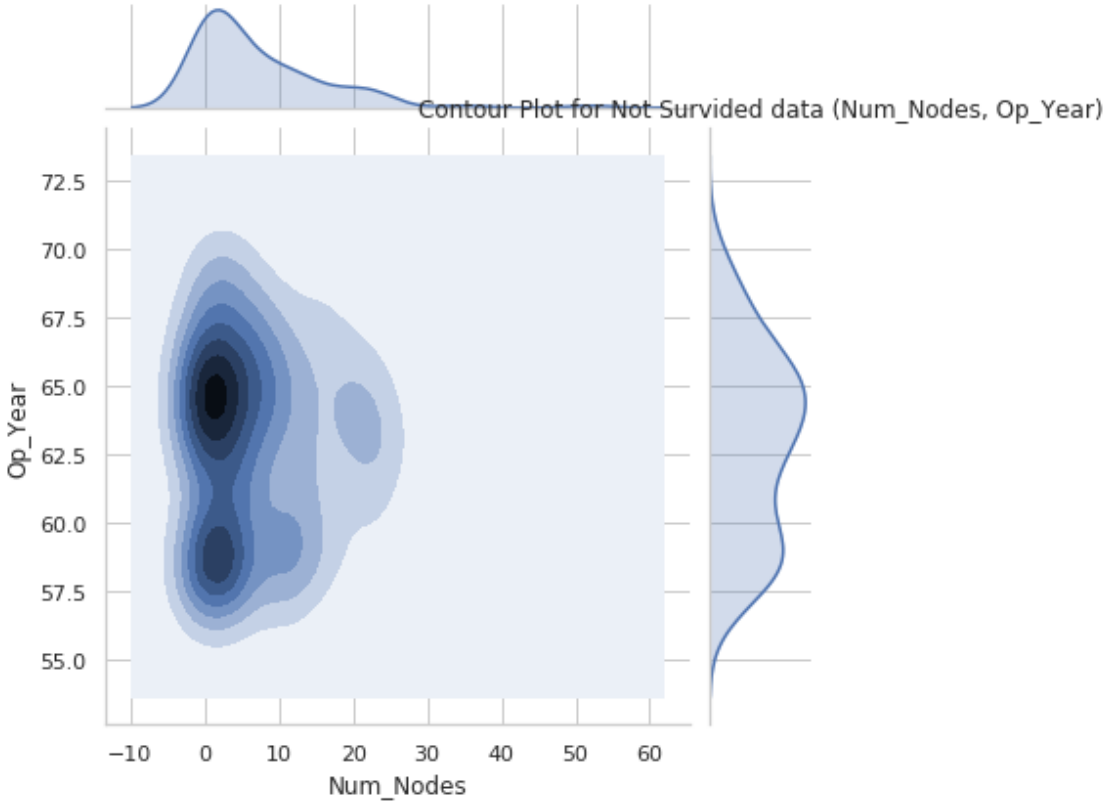

```
================================================================================================
==============================        Op_Year        Age        Contour Plot        ============
```


Contour Plot for Survided data (Op_Year, Age)
Age
90
80
70
60
50
40
30
20
55.0
57.5
60.0
62.5
65.0
67.5
70.0
72.5
Op_Year




```
================================================================================================
==============================        Num_Nodes        Op_Year        Contour Plot        ======
```


Contour Plot for Survided data (Num_Nodes, Op_Year)
72.5
70.0
67.5
65.0
62.5
60.0
57.5
55.0
Op_Year
0
10
20
30
40
50
Num_Nodes



==========

7 Conclusion

From the EDA results, the Num_nodes feature is the most important for classification