In [42]:

```
### Habermans-survival- Data sets
```

In [43]:

```
'''DESCRIPTION --
The dataset contains cases from a study that was conducted between 1958 and 197
at the University of Chicago's Billings Hospital on the survival of patients wh
undergone surgery for breast cancer.
Attribute Information:

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

```
"DESCRIPTION -- \nThe dataset contains cases from a study that was conducted between 1958 and 1970 \n
cago's Billings Hospital on the survival of patients who had \nundergone surgery for breast cancer.\n
\nAge of patient at time of operation (numerical)\nPatient's year of operation (year - 1900, numerica
xillary nodes detected (numerical)\nSurvival status (class attribute) 1 = the patient survived 5 year
ent \ndied within 5 year\n"
```

In [44]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

''' WE AN DWOWNLOAD DATA FROM https://www.kaggle.com/gilsousa/habermans-surviva
```

' WE AN DWOWNLOAD DATA FROM https://www.kaggle.com/gilsousa/habermans-survival-data-set' (https://www.kaggle.com/gilsousa/ha

In [45]:

```
DATA = pd.read_csv('haberman (1).csv')
print(DATA)
```

```
     age  year  nodes  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
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 x 4 columns]
```

In [46]:

```
# no of data point and features are
print(DATA.shape)
```

```
(306, 4)
```

In [47]:

```
# column names in our data set
print(DATA.columns)
```

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

In [48]:

```
# How many patients survived more than 5 years and patient died before 5 years
# denoted by 1 and 2
print(DATA['status'].value_counts())
```

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

In [49]:

```
# above result shows that this is an imbalanced dataset and people survived mor
# are more than than who died within 5 years
```

In [50]:

```
'''  OBJECTIVE   ---     Our objective is to visualize and analyse our data
and know which features are more useful or which features impact less
     by comparing them using various visualization techniques in 1D , 2D ,3D
     and classify them'''
```

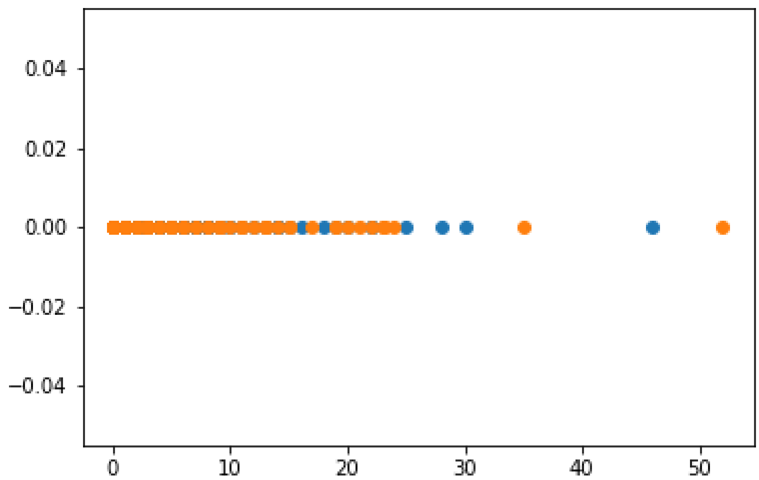
```
'  OBJECTIVE   ---     Our objective is to visualize and analyse our data \nand know which features a
features impact less\n     by comparing them using various visualization techniques in 1D , 2D ,3D \n
```

In [51]:

```
## 1D plots
```

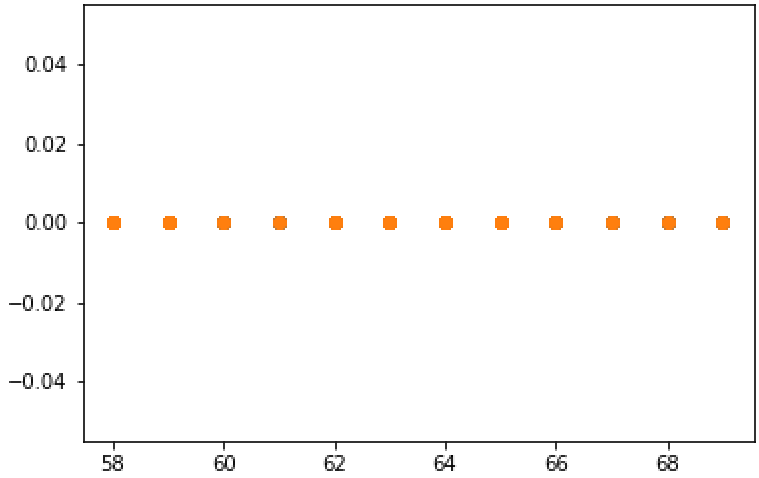
In [52]:

```
# 1D scatter plot



attribute1 = DATA[DATA['status']==1]
attribute2 = DATA[DATA['status']==2]


plt.plot(attribute1['nodes'], np.zeros_like(attribute1['nodes']), 'o')
plt.plot(attribute2['nodes'], np.zeros_like(attribute2['nodes']), 'o')
plt.show()
```



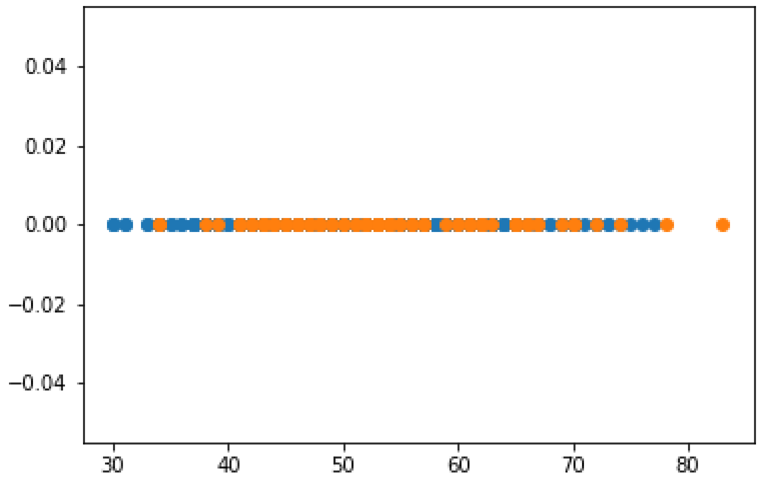
In [53]:

```
plt.plot(attribute1['year'], np.zeros_like(attribute1['year']), 'o')
plt.plot(attribute2['year'], np.zeros_like(attribute2['year']), 'o')
plt.show()
```



In [54]:

```
plt.plot(attribute1['age'], np.zeros_like(attribute1['age']), 'o')
plt.plot(attribute2['age'], np.zeros_like(attribute2['age']), 'o')
plt.show()
```



In [55]:

```
''' Observation for 1D scatter plot  :   In plot 1 sill we can classify on the
    of number of nodes by drawing a line at around 25 , In plot 2 the year attr
    not make any sense and in case of plot 3 that is attribute age does not mak
    so that we can classify it.
    Note: we cannot classify perfectly with attribute nodes too.
    Here we can also say that 1D scatter plot is not a good way of visualizing
    on this data set as there are more overlapping points. '''
```

```
' Observation for 1D scatter plot  :   In plot 1 sill we can classify on the basis \n    of number of
at around 25 , In plot 2 the year attribute does\n    not make any sense and in case of plot 3 that i
make much sense\n    so that we can classify it.\n    Note: we cannot classify perfectly with attribu
we can also say that 1D scatter plot is not a good way of visualizing for atleast\n    on this data s
rlapping points. '
```

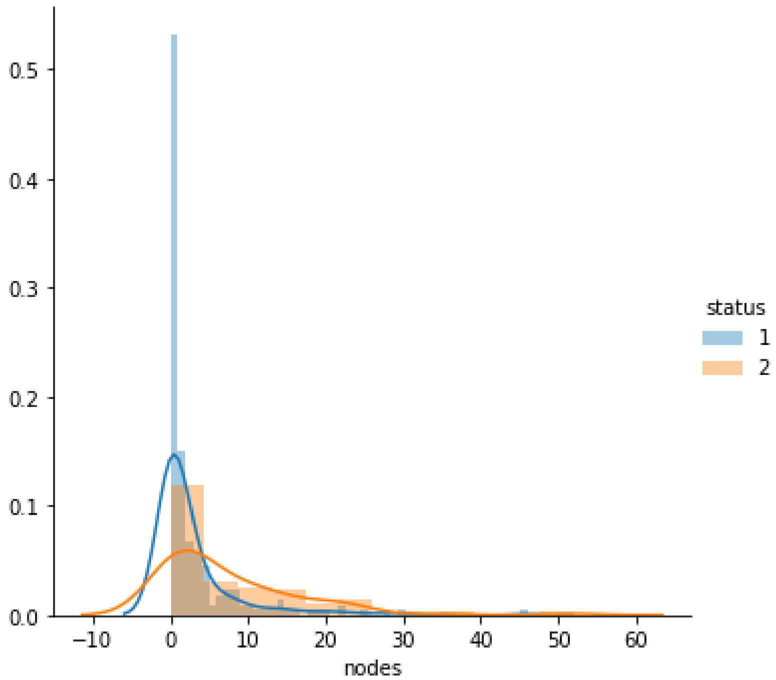
In [56]:

```
# Histograms and PDF(Probability Density Funcftion)
```

In [57]:

```
sns.FacetGrid(DATA, hue="status", size=5) \
   .map(sns.distplot, "nodes") \
   .add_legend();
plt.show();
```

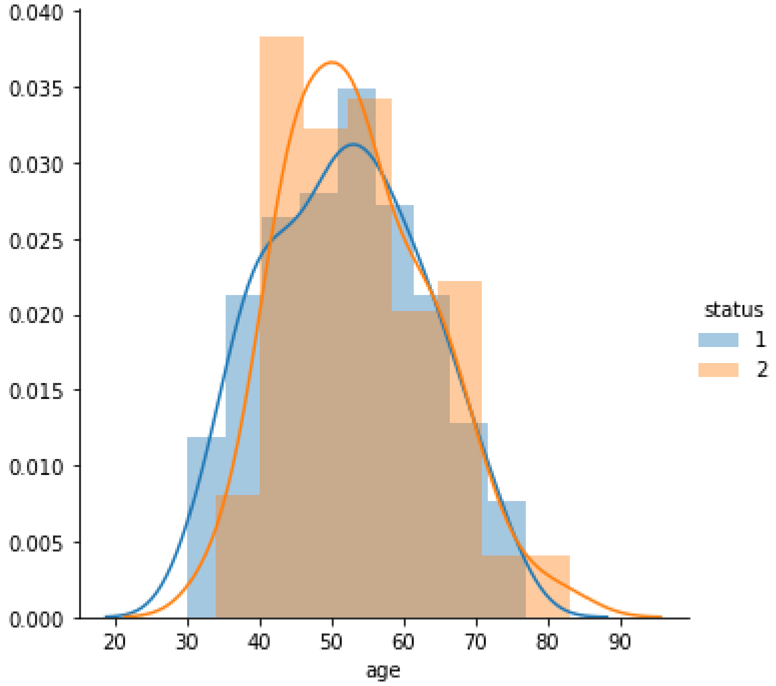
```
/usr/local/lib/python3.5/dist-packages/seaborn/axisgrid.py:230: UserWarning: The `size` paramter has
`; please update your code.
  warnings.warn(msg, UserWarning)
```



In [58]:

```
sns.FacetGrid(DATA, hue="status", size=5) \
   .map(sns.distplot, "age") \
   .add_legend();
plt.show();
```

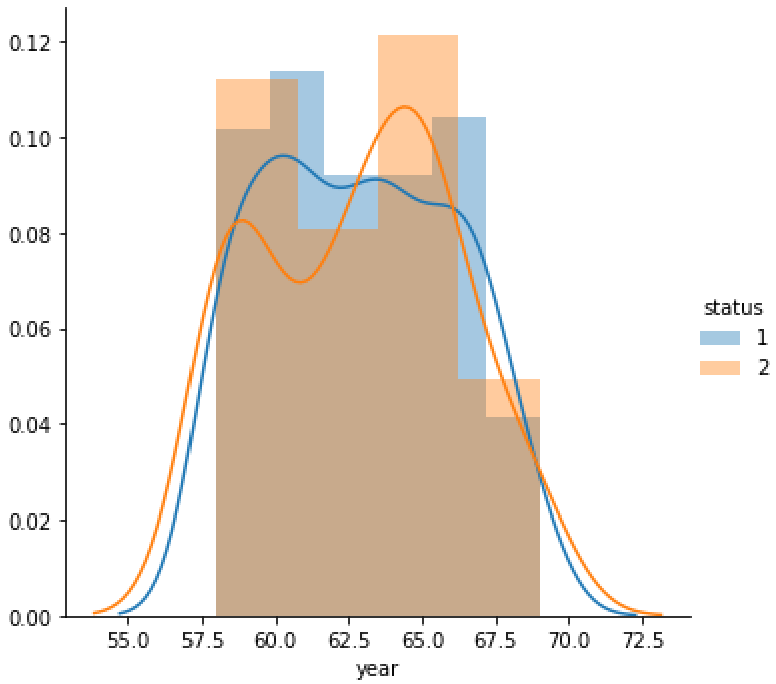
```
/usr/local/lib/python3.5/dist-packages/seaborn/axisgrid.py:230: UserWarning: The `size` paramter has
`; please update your code.
  warnings.warn(msg, UserWarning)
```



In [59]:

```
sns.FacetGrid(DATA, hue="status", size=5) \
   .map(sns.distplot, "year") \
   .add_legend();
plt.show();
```

```
/usr/local/lib/python3.5/dist-packages/seaborn/axisgrid.py:230: UserWarning: The `size` paramter has
`; please update your code.
  warnings.warn(msg, UserWarning)
```



In [60]:

```
''' Observation :  We cannot make any conclusion from these three plots as for
    all attributes the histograms and pdf are overlapped and in no way it can b
    classified or breakdown into an if statement '''
```

```
' Observation :  We cannot make any conclusion from these three plots as for \n    all attributes the
overlapped and in no way it can be\n    classified or breakdown into an if statement '
```

In [61]:

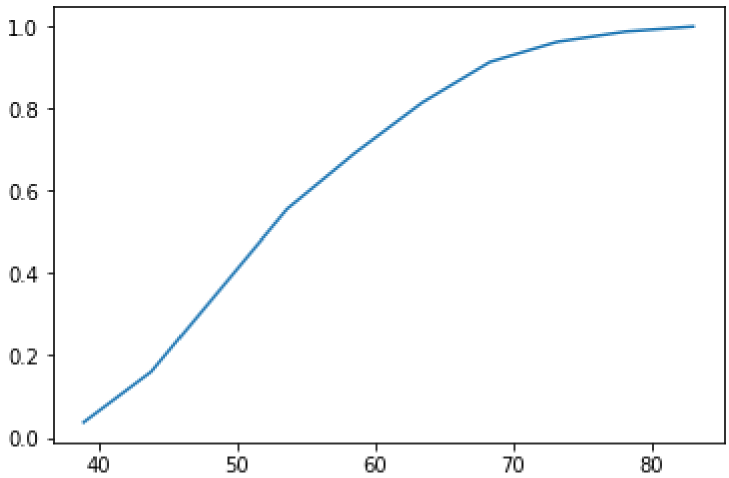
```
''' CDF (Cumulative Distribution Function)  using cdf we can see what percentag
    surviors or non survivors have age less than say 30 or Node less than say 5
```

```
' CDF (Cumulative Distribution Function)  using cdf we can see what percentage  of \n    surviors or
ess than say 30 or Node less than say 5'
```

In [62]:

```
counts,bin_edges = np.histogram(attribute2['age'],bins=10,density=True)
pdf = counts/(sum(counts))
print(pdf)
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
# WE can see that the patients who died within 5 years are mostly in the age in
# almost 50% of the people who died within 5 years are in this age interval
# we can say that the age could be the factor of more deaths
```

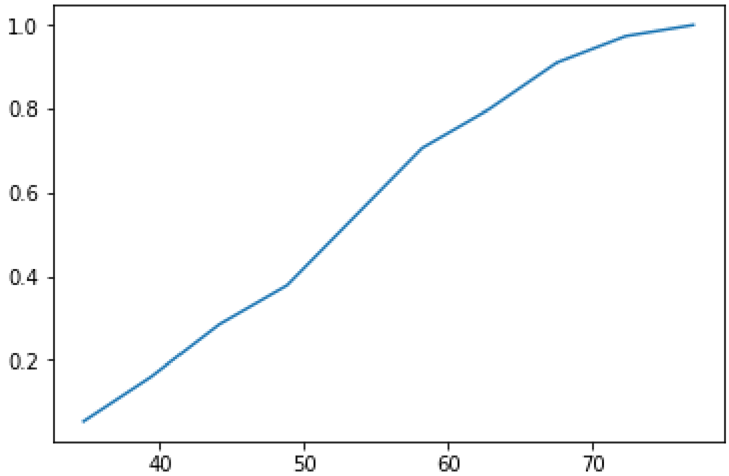
```
[0.03703704 0.12345679 0.19753086 0.19753086 0.13580247 0.12345679
 0.09876543 0.04938272 0.02469136 0.01234568]
[34.  38.9 43.8 48.7 53.6 58.5 63.4 68.3 73.2 78.1 83. ]
```



In [63]:

```
counts,bin_edges = np.histogram(attribute1['age'],bins=10,density=True)
pdf = counts/(sum(counts))
print(pdf)
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
# From this plot we can see that most of the people who lived longer than 5 yea
# are less than 65 years old so from the above graph we can conclude that the m
# people with age intervall of 65-80 didn't live longer than 5
```

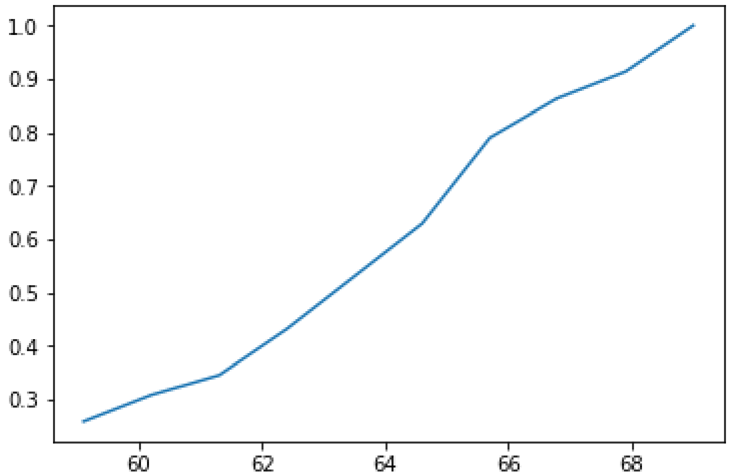
```
[0.05333333 0.10666667 0.12444444 0.09333333 0.16444444 0.16444444
 0.09333333 0.11111111 0.06222222 0.02666667]
[30.  34.7 39.4 44.1 48.8 53.5 58.2 62.9 67.6 72.3 77. ]
```



In [64]:

```
counts,bin_edges = np.histogram(attribute2['year'],bins=10,density=True)
pdf = counts/(sum(counts))
print(pdf)
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
```

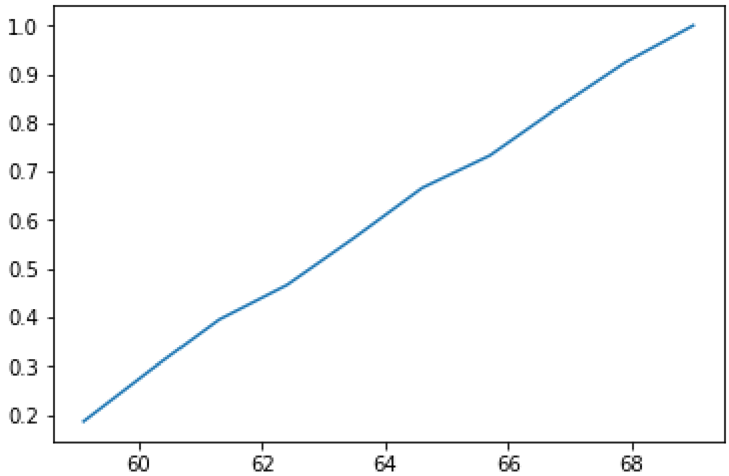
```
[0.25925926 0.04938272 0.03703704 0.08641975 0.09876543 0.09876543
 0.16049383 0.07407407 0.04938272 0.08641975]
[58.  59.1 60.2 61.3 62.4 63.5 64.6 65.7 66.8 67.9 69. ]
```



In [65]:

```
counts,bin_edges = np.histogram(attribute1['year'],bins=10,density=True)
pdf = counts/(sum(counts))
print(pdf)
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
```

```
[0.18666667 0.10666667 0.10222222 0.07111111 0.09777778 0.10222222
 0.06666667 0.09777778 0.09333333 0.07555556]
[58.  59.1 60.2 61.3 62.4 63.5 64.6 65.7 66.8 67.9 69. ]
```



In [66]:

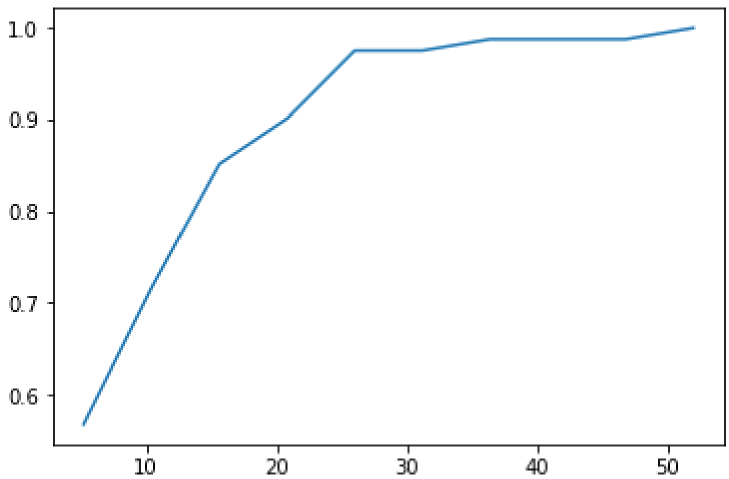
```
''' From the upper two cdf plots for year we can see a close to linear relatior
    which can make us say that year didn't effect and we can't make any conclus
    for this'''
```

```
" From the upper two cdf plots for year we can see a close to linear relation \n    which can make us
fect and we can't make any conclusion\n    for this"
```

In [67]:

```
counts,bin_edges = np.histogram(attribute2['nodes'],bins=10,density=True)
pdf = counts/(sum(counts))
print(pdf)
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
# In this plot we can say that most of the people approximately 90% died before
# years of treatment have nodes less than 25
```

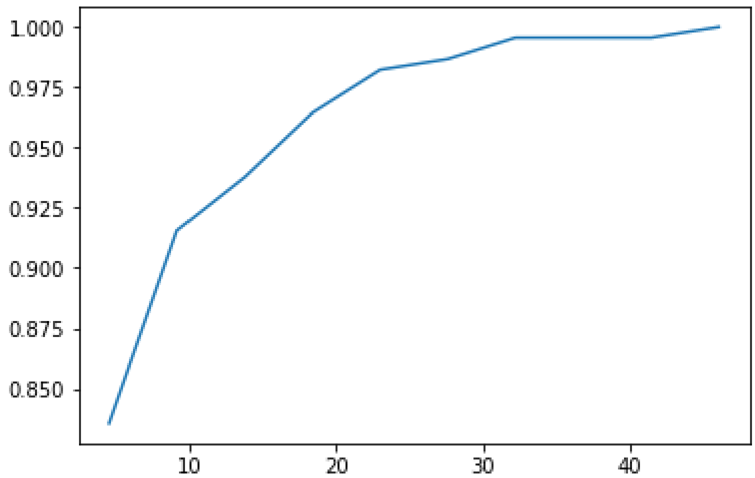
```
[0.56790123 0.14814815 0.13580247 0.04938272 0.07407407 0.
 0.01234568 0.         0.         0.01234568]
[ 0.   5.2 10.4 15.6 20.8 26.  31.2 36.4 41.6 46.8 52. ]
```



In [68]:

```
counts,bin_edges = np.histogram(attribute1['nodes'],bins=10,density=True)
pdf = counts/(sum(counts))
print(pdf)
print(bin_edges)
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
''' From this plot we can say that most of the people having nodes less than 10
    survived more than 5 years after treatment and from the above graph we
    can conclude that people having nodes between 10-25 are more likely to die
     5 years of treatment '''
```

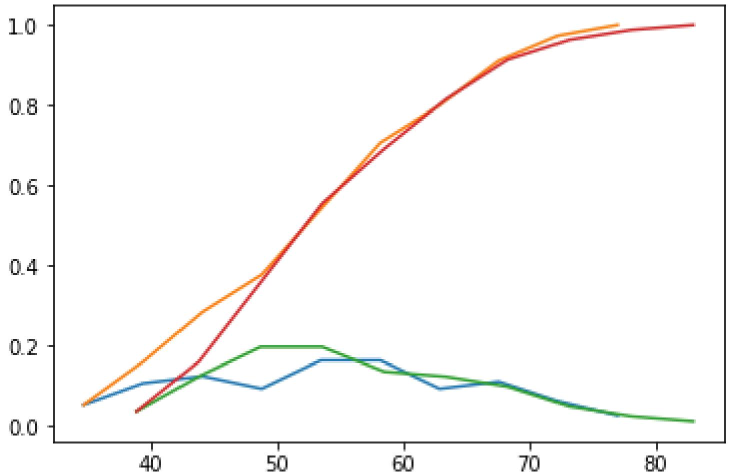
```
[0.83555556 0.08       0.02222222 0.02666667 0.01777778 0.00444444
 0.00888889 0.         0.         0.00444444]
[ 0.   4.6  9.2 13.8 18.4 23.  27.6 32.2 36.8 41.4 46. ]
```




```
' From this plot we can say that most of the people having nodes less than 10\n    survived more than
and from the above graph we \n    can conclude that people having nodes between 10-25 are more likely
years of treatment '
```

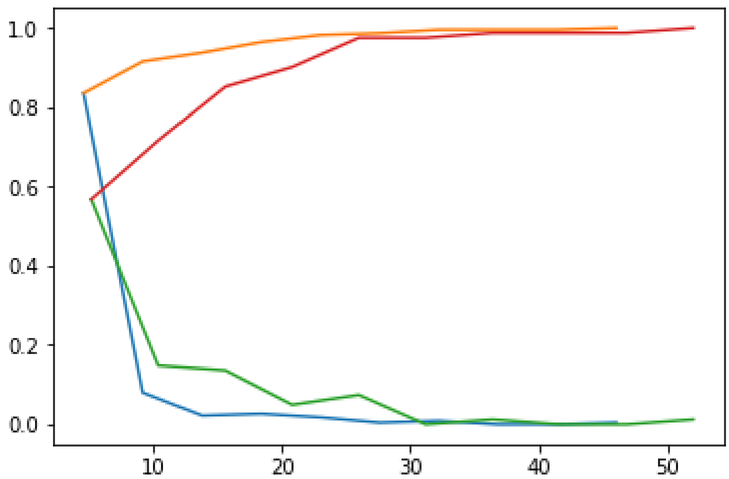
In [69]:

```
counts,bin_edges = np.histogram(attribute1['age'],bins=10,density=True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)
counts,bin_edges = np.histogram(attribute2['age'],bins=10,density=True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
```



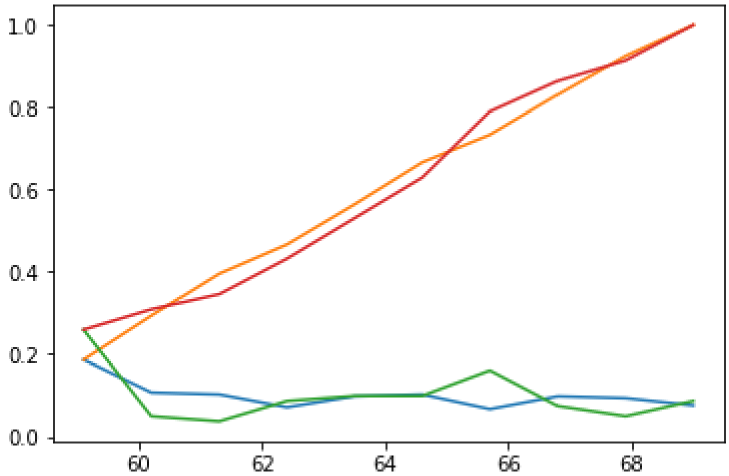
In [70]:

```
counts,bin_edges = np.histogram(attribute1['nodes'],bins=10,density=True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)
counts,bin_edges = np.histogram(attribute2['nodes'],bins=10,density=True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
```



In [71]:

```
counts,bin_edges = np.histogram(attribute1['year'],bins=10,density=True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)
counts,bin_edges = np.histogram(attribute2['year'],bins=10,density=True)
pdf = counts/(sum(counts))
cdf = np.cumsum(pdf)
plt.plot(bin_edges[1:],pdf)
plt.plot(bin_edges[1:],cdf)
plt.show()
```



In [72]:

```
''' From the above three pdf-cdf combine plots we cannot make conclusion cause
     plots are almost completely overlapping '''
```

```
' From the above three pdf-cdf combine plots we cannot make conclusion cause pdf\n     plots are almo
g '
```

In [73]:

```
# Means
print('Means for patients survived longer than 5 years')
print('1:',np.mean(attribute1['age']))
print('2:',np.mean(attribute1['year']))
print('3:',np.mean(attribute1['nodes']))
print('*'*30)
print("Means for patients survived less than 5 years")
print('1:',np.mean(attribute2['age']))
print('2:',np.mean(attribute2['year']))
print('3:',np.mean(attribute2['nodes']))
# This concludes on an average patients survived more than 5 years have around
# and the one  deid within 5 years have around 7-8 nodes
```

```
Means for patients survived longer than 5 years
1: 52.01777777777778
2: 62.86222222222222
3: 2.7911111111111113
******************************
Means for patients survived less than 5 years
1: 53.67901234567901
2: 62.82716049382716
3: 7.45679012345679
```

In [74]:

```
# Std-deviation
print("Std-deviation for patients survived longer than 5 years")
print('1',np.std(attribute1['age']))
print('2',np.std(attribute1['year']))
print('3',np.std(attribute1['nodes']))
print('*'*30)
print("Std-deviation for patients survived less than 5 years")
print('1',np.std(attribute2['age']))
print('2',np.std(attribute2['year']))
print('3',np.std(attribute2['nodes']))
```

```
Std-deviation for patients survived longer than 5 years
1 10.987655475100508
2 3.2157452144021947
3 5.857258449412138
******************************
Std-deviation for patients survived less than 5 years
1 10.104182193031312
2 3.3214236255207887
3 9.128776076761635
```

In [75]:

```
print("Median for patients survived longer than 5 years")
print('1',np.median(attribute1['age']))
print('2',np.median(attribute1['year']))
print('3',np.median(attribute1['nodes']))
print('*'*30)
print("Median for patients survived less than 5 years")
print('1',np.median(attribute2['age']))
print('2',np.median(attribute2['year']))
print('3',np.median(attribute2['nodes']))
# Here we can see that the one who survived more than five years have median=0
# which means half of the patient who survived have no nodes and this can be sa
# huge cuuse of survival of the patients
```

```
Median for patients survived longer than 5 years
1 52.0
2 63.0
3 0.0
******************************
Median for patients survived less than 5 years
1 53.0
2 63.0
3 4.0
```

In [76]:

```
print("Quantiles for patients survived longer than 5 years")
print('1',np.percentile(attribute1['age'],np.arange(0,100,25)))
print('2',np.percentile(attribute1['year'],np.arange(0,100,25)))
print('3',np.percentile(attribute1['nodes'],np.arange(0,100,25)))
print('*'*30)
print("Quantiles for patients survived less than 5 years")
print('1',np.percentile(attribute2['age'],np.arange(0,100,25)))
print('2',np.percentile(attribute2['year'],np.arange(0,100,25)))
print('3',np.percentile(attribute2['nodes'],np.arange(0,100,25)))
# Here we can see most of the patient more than 75% who survived more than 5  y
# no nodes ie nodes=0.0 which shows nodes plays a significant role in determing
# who survived more
```

```
Quantiles for patients survived longer than 5 years
1 [30. 43. 52. 60.]
2 [58. 60. 63. 66.]
3 [0. 0. 0. 3.]
******************************
Quantiles for patients survived less than 5 years
1 [34. 46. 53. 61.]
2 [58. 59. 63. 65.]
3 [ 0.  1.  4. 11.]
```

In [83]:

```
print("90th pecentile for patients survived longer than 5 years")
print('1',np.percentile(attribute1['age'],90))
print('2',np.percentile(attribute1['year'],90))
print('3',np.percentile(attribute1['nodes'],90))
print('*'*30)
print("i90th percentile for patients survived less than 5 years")
print('1',np.percentile(attribute2['age'],90)
print('2',np.percentile(attribute2['year'],90))
print('3',np.percentile(attribute2['nodes'],90))
```

```
  File "<ipython-input-83-821e2fcbbafa>", line 8
    print('2',np.percentile(attribute2['year'],90))
        ^
SyntaxError: invalid syntax
```

In [78]:

```
from statsmodels import robust
print("Median absolute deviation for patients survived longer than 5 years")
print('1',robust.mad(attribute1['age']))
print('2',robust.mad(attribute1['year']))
print('3',robust.mad(attribute1['nodes']))
print('*'*30)
print("Median absolute deviation for patients survived less than 5 years")
print('1',robust.mad(attribute2['age']))
print('2',robust.mad(attribute2['year']))
print('3',robust.mad(attribute2['nodes']))
```

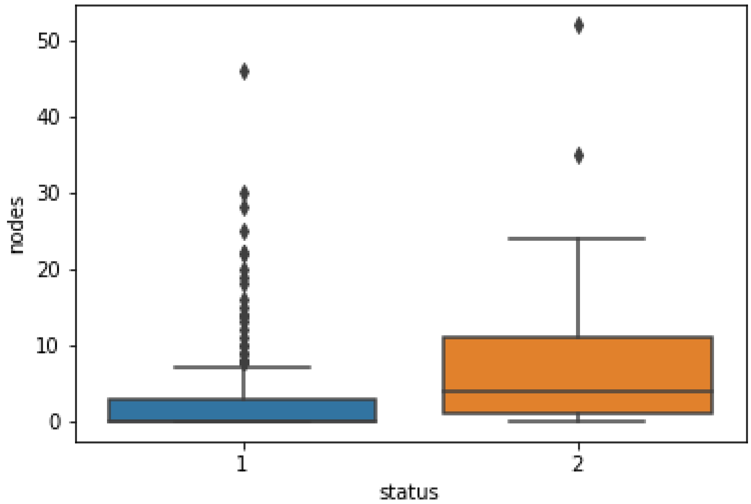
```
Median absolute deviation for patients survived longer than 5 years
1 13.343419966550417
2 4.447806655516806
3 0.0
******************************
Median absolute deviation for patients survived less than 5 years
1 11.860817748044816
2 4.447806655516806
3 5.930408874022408
```

In [79]:

```
# BOX PLOT AND WHISKERS  in this a technique called inter-quantile range is use
# the whiskers in the plot maybe corresponds to min-max value or 1.5*IQR
# 3 lines corresponding to mthe colored box are quantiles ie 75th,50th,25th pe
```

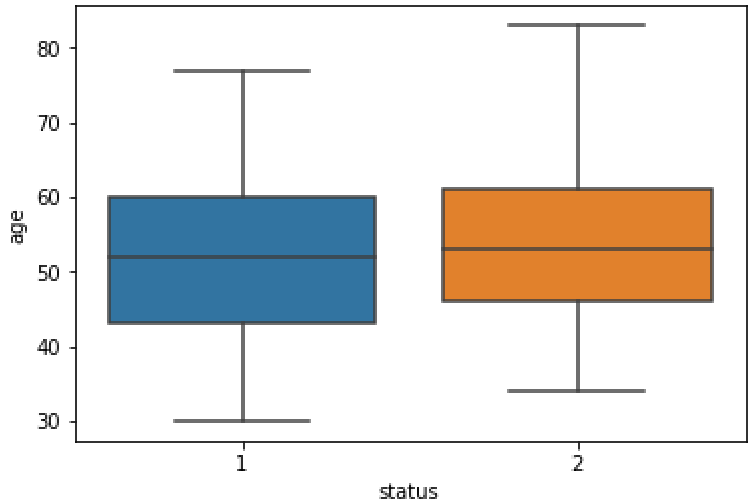
In [80]:

```
sns.boxplot(x='status',y='nodes',data=DATA)
plt.show()
```



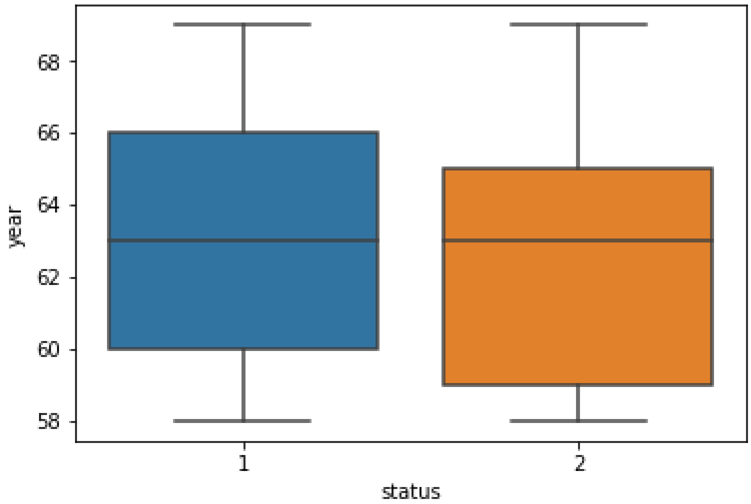
In [81]:

```
sns.boxplot(x='status',y='age',data=DATA)
plt.show()
```



In [82]:

```
sns.boxplot(x='status',y='year',data=DATA)
plt.show()
```



In [84]:

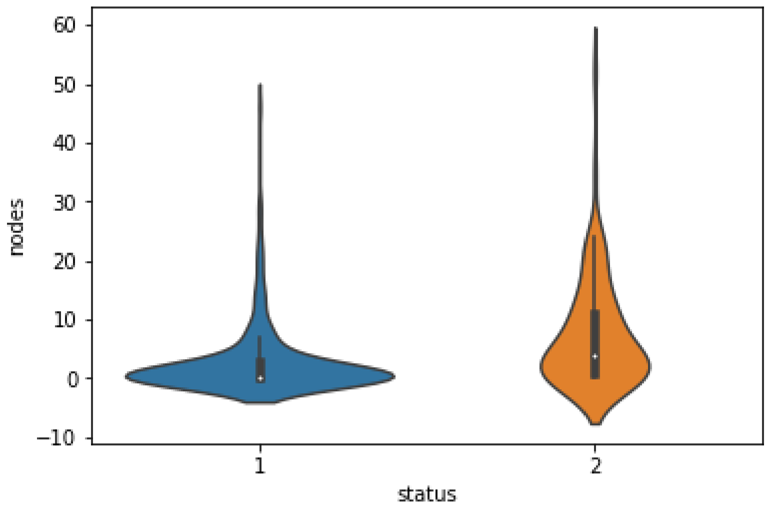
```
# From the above three box plots we cannot say anything about age and years as
# them are almost overlapping but for nodes we can say that
# if nodes>0 patient died within 5 years of operation
# else he survived more than 5
```

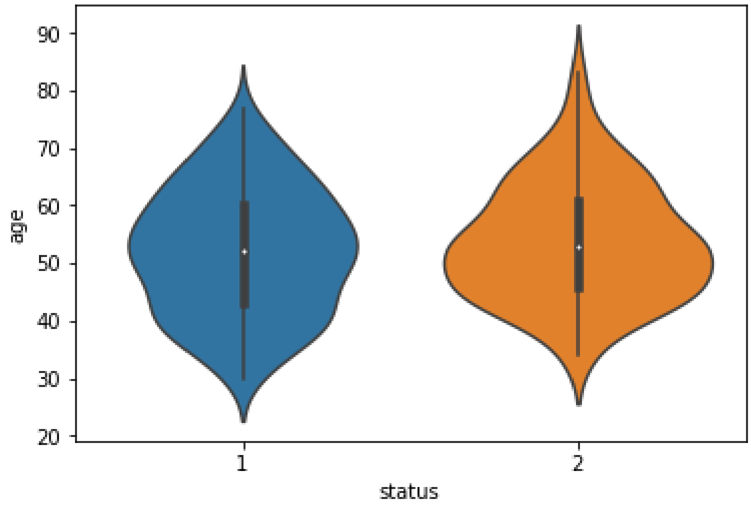
In [86]:

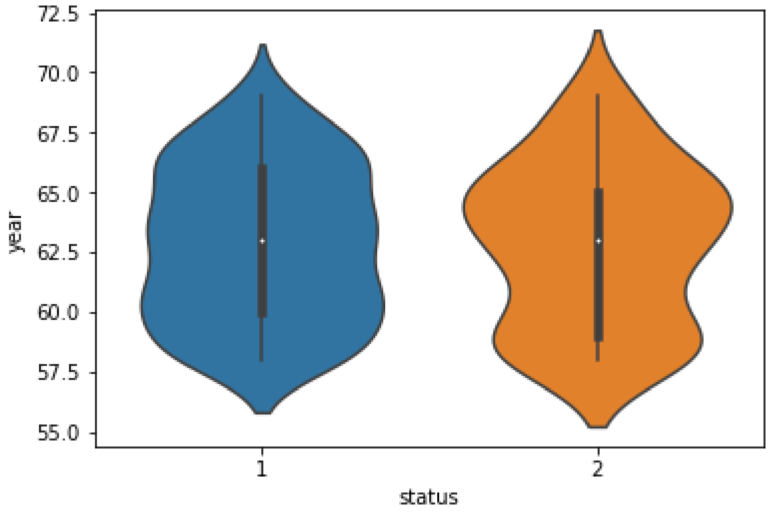
```
# VIOLIN PLOT     It combines the benefit of pds and box plots and simplifies t
# Denser region of the data are fatter and sparse ones thinner and the thick bl
# in the middle is box plots and edges are  whiskers
```

In [92]:

```
sns.violinplot(x='status',y='nodes',data=DATA)
plt.show()
sns.violinplot(x='status',y='age',data=DATA)
plt.show()
sns.violinplot(x='status',y='year',data=DATA)
plt.show()
```







In [93]:

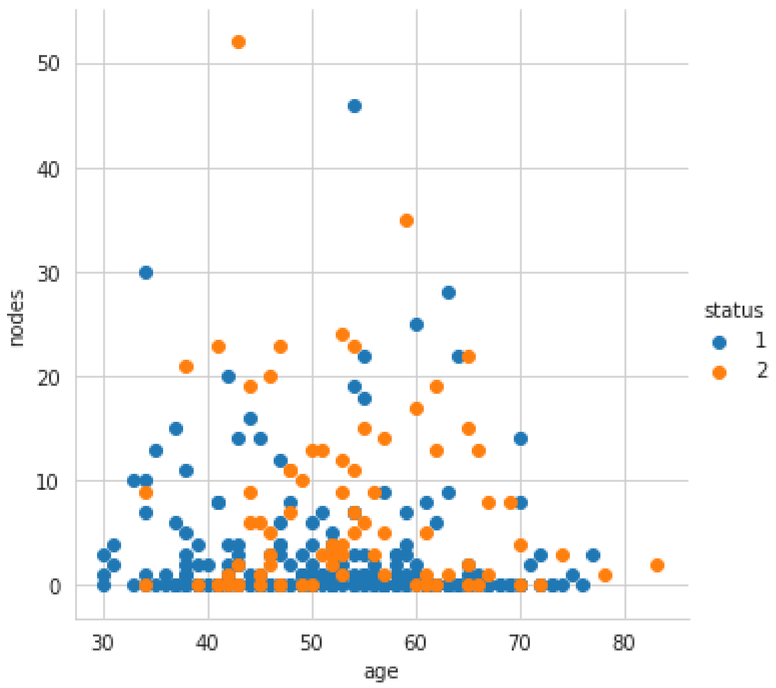
```
# 2-D plots
```

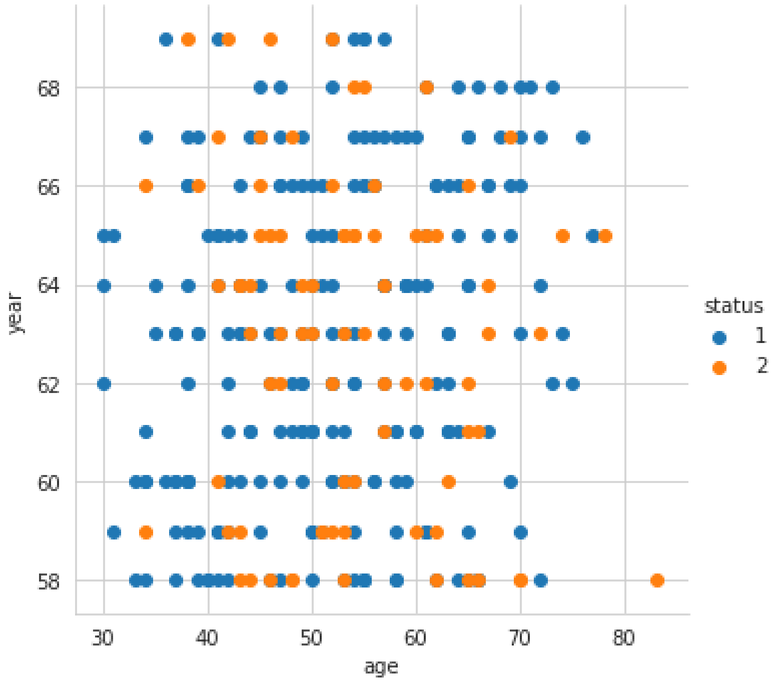
In [94]:

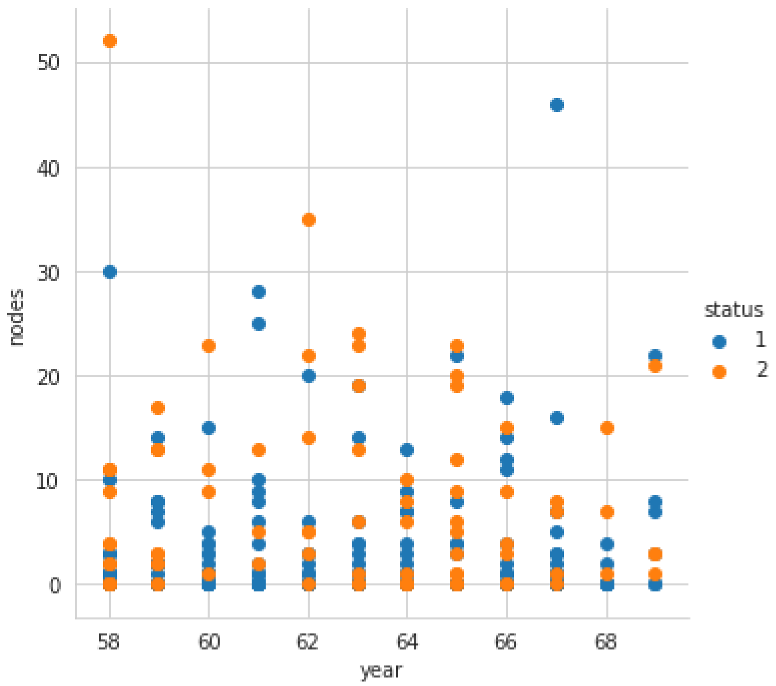
```
# 2-D colored scatter plot
```

In [110]:

```
sns.set_style("whitegrid")
sns.FacetGrid(DATA,hue="status",height=5).map(plt.scatter,"age","nodes").add_le
plt.show()
sns.set_style("whitegrid")
sns.FacetGrid(DATA,hue="status",height=5).map(plt.scatter,"age","year").add_leg
plt.show()
sns.set_style("whitegrid")
sns.FacetGrid(DATA,hue="status",height=5).map(plt.scatter,"year","nodes").add_l
plt.show()
```







In [98]:

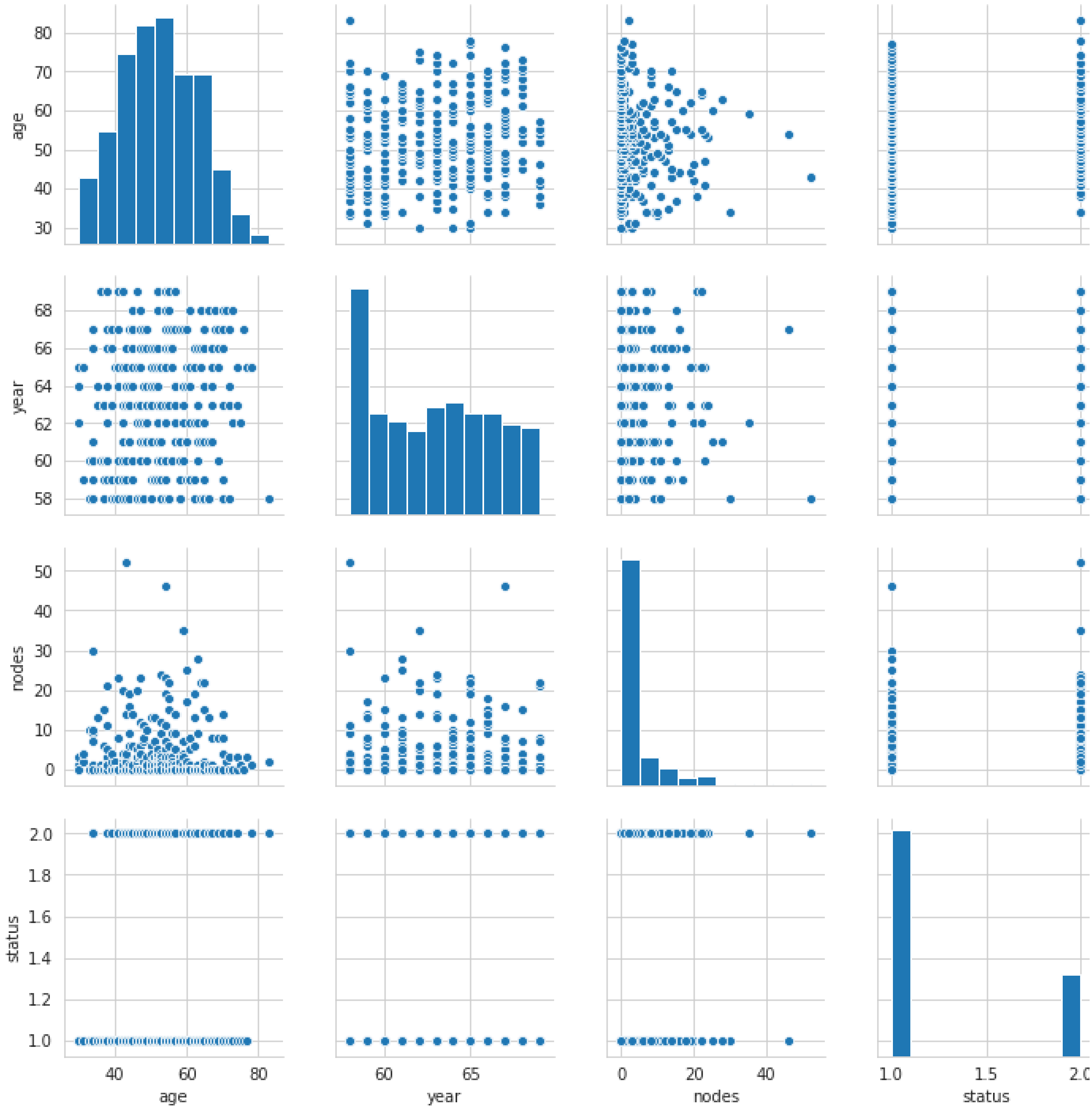
```
# observation : from above three scatter plot we cannot conclude anything
```

In [99]:

```
# PAIR-PLOT
```

In [130]:

```
plt.close()
sns.set_style("whitegrid")
sns.pairplot(DATA)#if the number of items in a category is 3, the palette is
#flattened and they come up in a greyscale.This is why i can't use hue
plt.show()
# from the pair plots below we cannot make any conclusion due to overlapping
```

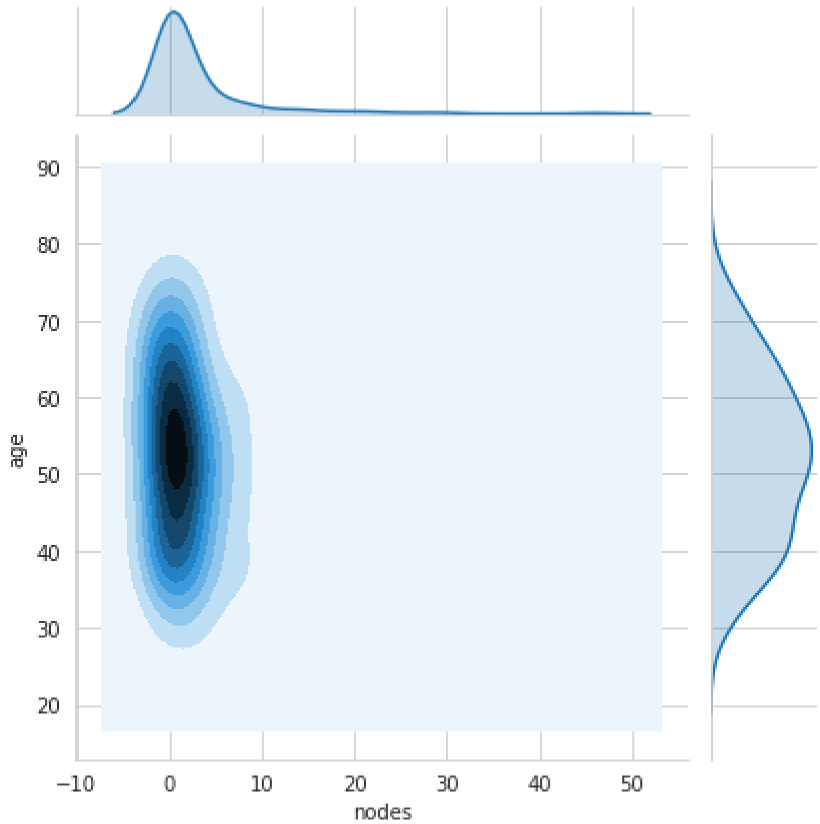


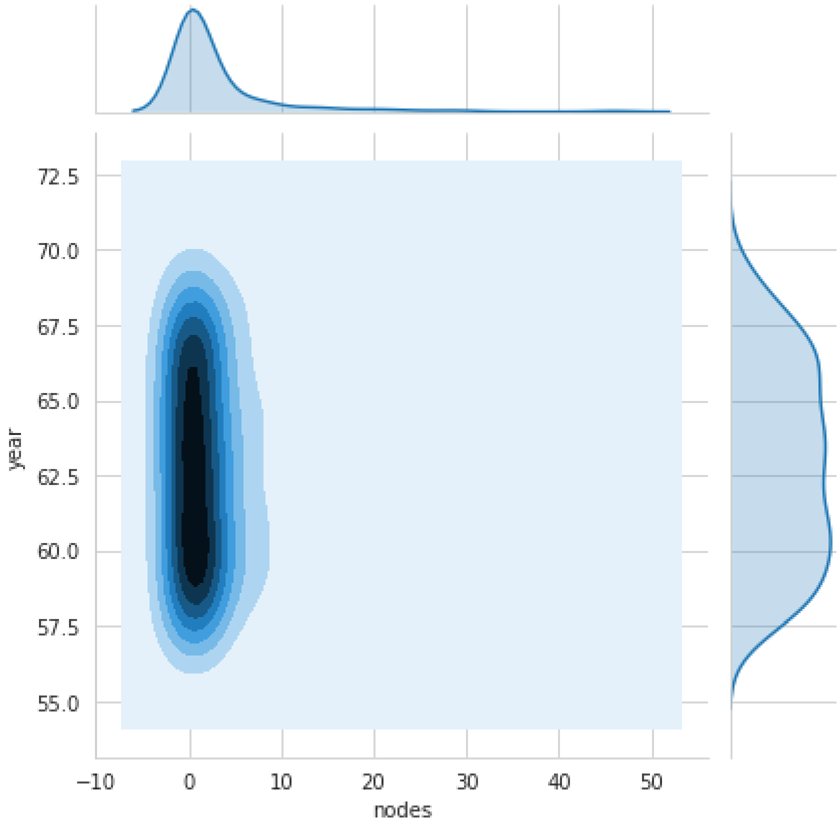
In [ ]:

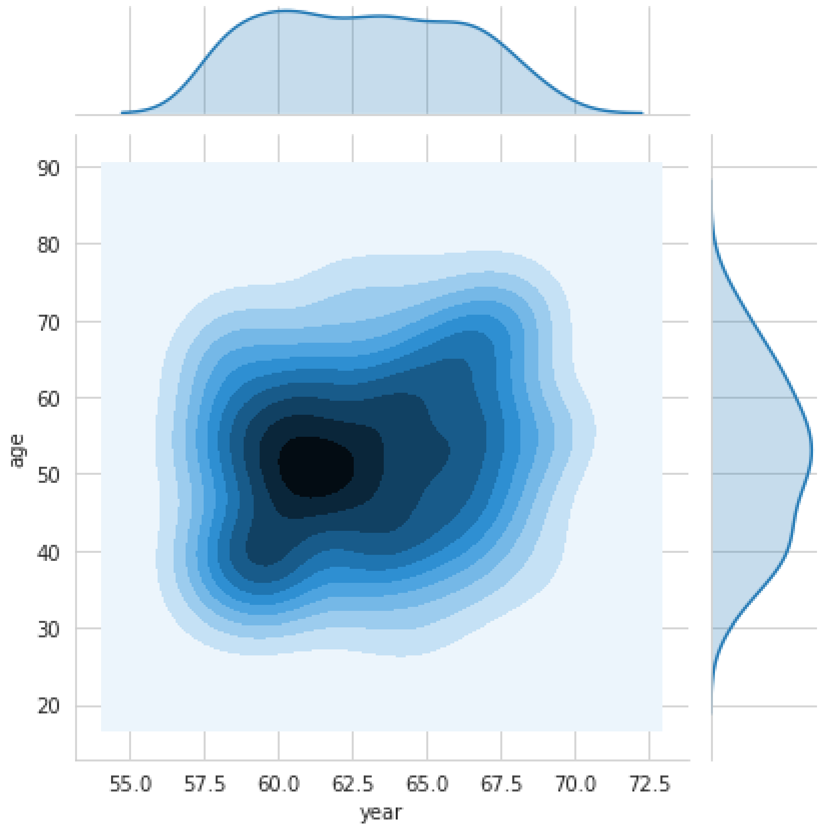
```
# CONTOUR PROBABILITY DENSITY PLOT  IT HELPS US IN HSOWING 2-D DENSITY
```

In [132]:

```
sns.jointplot(x="nodes",y='age',data=attribute1,kind='kde')
plt.show()
sns.jointplot(x="nodes",y='year',data=attribute1,kind='kde')
plt.show()
sns.jointplot(x="year",y='age',data=attribute1,kind='kde')
plt.show()
```




72.5
70.0
67.5
65.0
62.5
60.0
57.5
55.0
year
−10
0
10
20
30
40
50
nodes

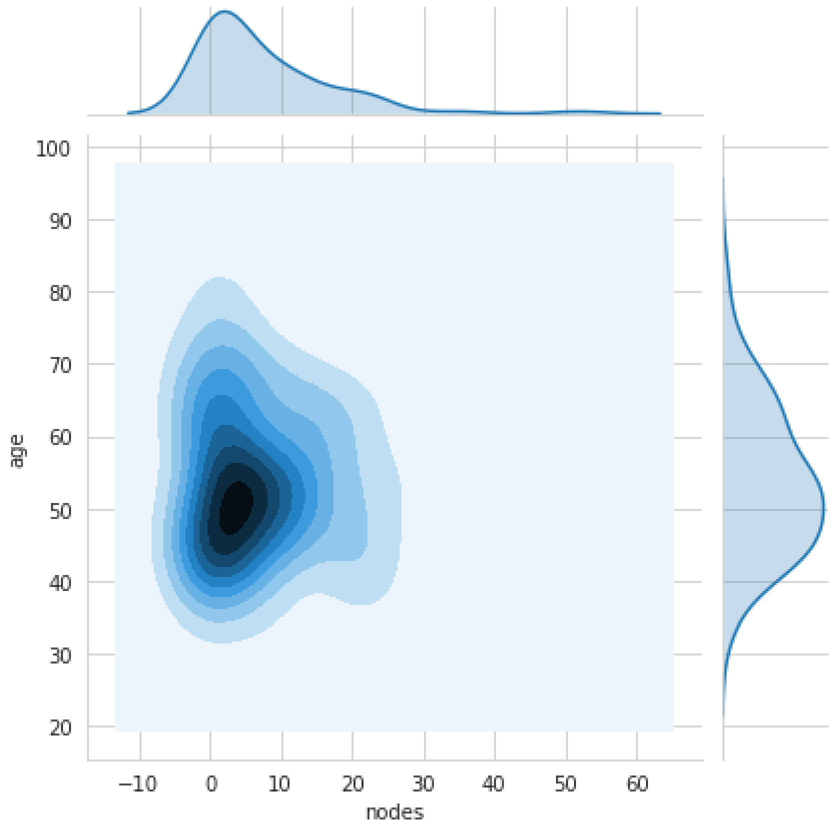

90
80
70
60
50
40
30
20
age
55.0
57.5
60.0
62.5
65.0
67.5
70.0
72.5
year

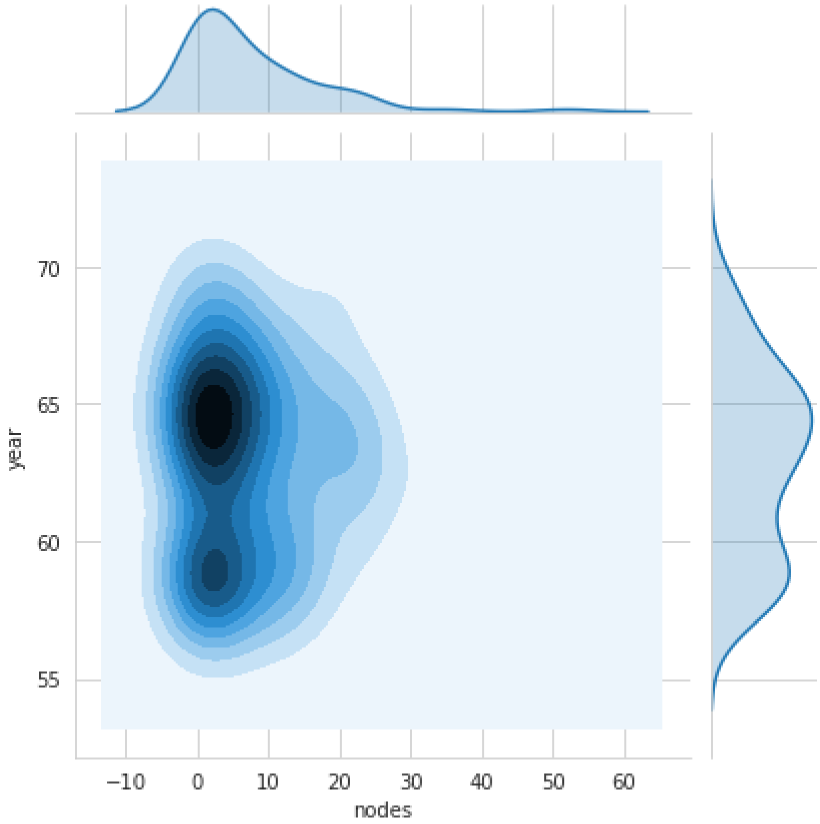
In [133]:

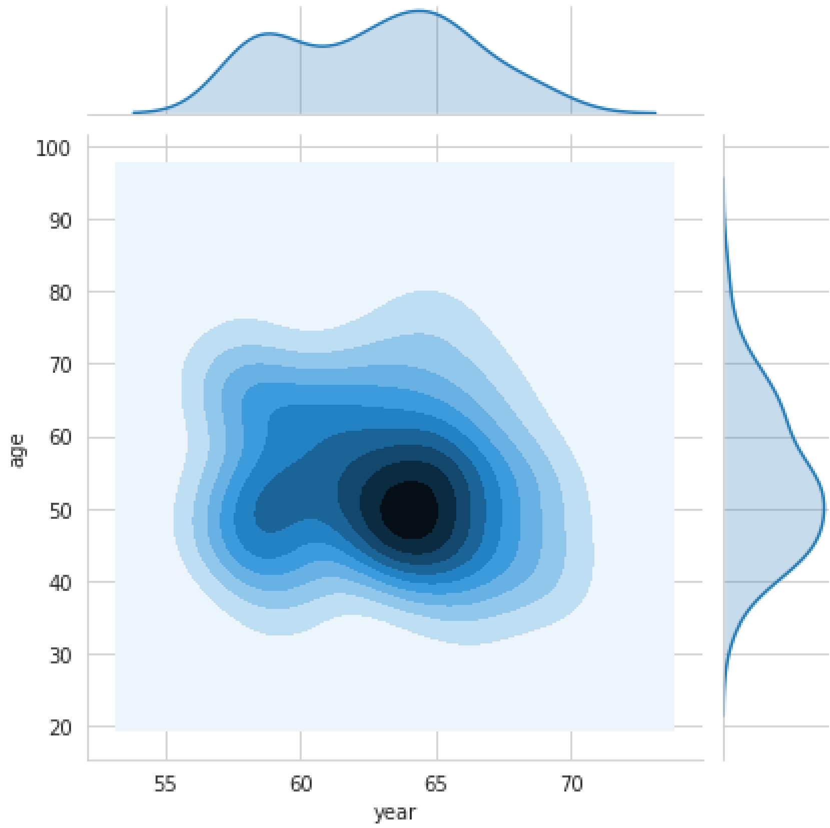
```
# observation : Here the darkest region tells us that there  are maximum number
# in that region considering last graph we can say that approximately most of t
# in interval where year is between 1961-1963 and with age between 48-55 which
# survivor in that region
```

In [134]:

```
sns.jointplot(x="nodes",y='age',data=attribute2,kind='kde')
plt.show()
sns.jointplot(x="nodes",y='year',data=attribute2,kind='kde')
plt.show()
sns.jointplot(x="year",y='age',data=attribute2,kind='kde')
plt.show()
```




70
65
60
55
year
−10
0
10
20
30
40
50
60
nodes


100
90
80
70
60
50
40
30
20
age
55
60
65
70
year

```
In [ ]:
```