Project #3

SDS348 Spring 2021

Julia Capelli (jcc5625)

In [ ]:

```
#Import Packages
import numpy as np
Let's use some `pandas` functions in Python that are equivalent
to `tidyr` functions in R for data wrangling:
![image.png](attachment:image.png)import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(color_codes=True)
```

This dataset was taken from Professor Layla Guyots Github website. This dataset is orginally from the Mass package. It is a dataset that covers breastcancer patients from the University of Wisconsin Hospitals. In this dataset Dr.William H. Wolberg assesed biopsies of breast tumors in 699 paitients and 11 varibles. There were nine variables that were scored on a scale of 1 to 10. All of the variables were compared to the outcome variable which was if the cells were benign or malignant.

In [27]:

```
# Import dataset
Biopsy = pd.read_csv("https://raw.githubusercontent.com/laylaguy
ot/datasets/main//Biopsy.csv")
```

In [101]:

```
#View Dataset
Biopsy.head
```

Out[101]:

```
<bound method NDFrame.head of     clump_thickness

uniform_cell_size  uniform_cell_shape  marg_adhesion
\
0                  5                 1
1              1
1                  5                 4
4              5
2                  3                 1
1              1
3                  6                 8
8              1
4                  4                 1
1              3
5                  8                10
10              8
6                  1                 1
1              1
7                  2                 1
2              1
8                  2                 1
1              1
9                  4                 2
1              1
10                 1                 1
1              1
11                 2                 1
1              1
12                 5                 3
3              3
13                 1                 1
1              1
14                 8                 7
5             10
15                 7                 4
6              4
16                 4                 1
1              1
17                 4                 1
1              1
18                10                 7
7              6
19                 6                 1
1              1
20                 7                 3
```

```
2             10
21                10                  5
5              3
22                 3                  1
1              1
23                 1                  1
1              1
24                 5                  2
3              4
25                 3                  2
1              1
26                 5                  1
1              1
27                 2                  1
1              1
28                 1                  1
3              1
29                 3                  1
1              1
..               ...                ...
...            ...
653                5                 10
10              8
654                3                 10
7              8
655                3                  2
1              2
656                2                  1
1              1
657                5                  3
2              1
658                1                  1
1              1
659                4                  1
4              1
660                1                  1
2              1
661                5                  1
1              1
662                1                  1
1              1
663                2                  1
1              1
664               10                 10
10             10
665                5                 10
```

```
665                5                 10

10             10
666                5                  1
1              1
667                1                  1
1              1
668                1                  1
1              1
669                1                  1
1              1
670                1                  1
1              1
671                3                  1
1              1
672                4                  1
1              1
673                1                  1
1              1
674                1                  1
1              3
675                5                 10
10              5
676                3                  1
1              1
677                3                  1
1              1
678                3                  1
1              1
679                2                  1
1              1
680                5                 10
10              3
681                4                  8
6              4
682                4                  8
8              5

     epithelial_cell_size  bare_nuclei  bland_chroma
tin  normal_nucleoli  \
0                       2            1
3                 1
1                       7           10
3                 2
2                       2            2
3                 1
```

```
3                  1
3                         3              4
3                  7
4                         2              1
3                  1
5                         7             10
9                  7
6                         2             10
3                  1
7                         2              1
3                  1
8                         2              1
1                  1
9                         2              1
2                  1
10                        1              1
3                  1
11                        2              1
2                  1
12                        2              3
4                  4
13                        2              3
3                  1
14                        7              9
5                  5
15                        6              1
4                  3
16                        2              1
2                  1
17                        2              1
3                  1
18                        4             10
4                  1
19                        2              1
3                  1
20                        5             10
5                  4
21                        6              7
7                 10
22                        2              1
2                  1
23                        2              1
3                  1
24                        2              7
3                  6
```

```
25                      1            1

2                1
26                      2            1
2                1
27                      2            1
2                1
28                      2            1
1                1
29                      1            1
2                1
..                    ...          ...
...              ...
653                     5            5
7               10
654                     5            8
7                4
655                     2            1
3                1
656                     2            1
3                1
657                     3            1
1                1
658                     2            1
2                1
659                     2            1
1                1
660                     2            1
2                1
661                     2            1
1                1
662                     2            1
1                1
663                     2            1
1                1
664                     5           10
10               10
665                     4           10
5                6
666                     2            1
3                2
667                     2            1
1                1
668                     2            1
1                1
```

```
669                     2            1
1                1
670                     2            1
1                1
671                     2            1
2                3
672                     2            1
1                1
673                     2            1
1                1
674                     2            1
1                1
675                     4            5
4                4
676                     2            1
1                1
677                     2            1
2                1
678                     3            2
1                1
679                     2            1
1                1
680                     7            3
8               10
681                     3            4
10                6
682                     4            5
10                4

     mitoses    outcome
0          1     benign
1          1     benign
2          1     benign
3          1     benign
4          1     benign
5          1  malignant
6          1     benign
7          1     benign
8          5     benign
9          1     benign
10         1     benign
11         1     benign
12         1  malignant
13         1     benign
14         4  malignant
15         1  malignant
```

```
15         1  malignant

16         1     benign
17         1     benign
18         2  malignant
19         1     benign
20         4  malignant
21         1  malignant
22         1     benign
23         1     benign
24         1  malignant
25         1     benign
26         1     benign
27         1     benign
28         1     benign
29         1     benign
..       ...        ...
653        1  malignant
654        1  malignant
655        1     benign
656        1     benign
657        1     benign
658        1     benign
659        1     benign
660        1     benign
661        1     benign
662        1     benign
663        1     benign
664        7  malignant
665        3  malignant
666        1     benign
667        1     benign
668        1     benign
669        1     benign
670        1     benign
671        1     benign
672        1     benign
673        8     benign
674        1     benign
675        1  malignant
676        1     benign
677        2     benign
678        1     benign
679        1     benign
680        2  malignant
681        1  malignant
```

```
681        1  malignant
682        1  malignant

[683 rows x 10 columns]>
```

In [102]:

```
#Show info about dataset
Biopsy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 683 entries, 0 to 682
Data columns (total 10 columns):
clump_thickness         683 non-null int64
uniform_cell_size       683 non-null int64
uniform_cell_shape      683 non-null int64
marg_adhesion           683 non-null int64
epithelial_cell_size    683 non-null int64
bare_nuclei             683 non-null int64
bland_chromatin         683 non-null int64
normal_nucleoli         683 non-null int64
mitoses                 683 non-null int64
outcome                 683 non-null object
dtypes: int64(9), object(1)
memory usage: 78.7+ KB
```

In [29]:

```
Biopsy.shape
```

Out[29]:

```
(683, 10)
```

From the output we can see the table contains 683 rows and 10 columns.

In [61]:

```
#Info about the data set
Biopsy.describe()
```

Out[61]:

	clump_thickness	uniform_cell_size	uniform_cell_shape	marg_adh
count	683.000000	683.000000	683.000000	683.00
mean	4.442167	3.150805	3.215227	2.83
std	2.820761	3.065145	2.988581	2.86
min	1.000000	1.000000	1.000000	1.00
25%	2.000000	1.000000	1.000000	1.00
50%	4.000000	1.000000	1.000000	1.00
75%	6.000000	5.000000	5.000000	4.00
max	10.000000	10.000000	10.000000	10.00

This a summary statistics for all of the numeric variables in this dataset. All of the variables were scored on a scale of 1-10 which explains why all the variables have a min and max of 10 and why they will all have the same range. Clump Thickness has the highest mean of 4.4 and mitoses has the lowest means of 1.6.

In [62]:

```
#Drop missing values
Biopsy = Biopsy.dropna()
Biopsy.count()
```

Out[62]:

```
clump_thickness           683
uniform_cell_size         683
uniform_cell_shape        683
marg_adhesion             683
epithelial_cell_size      683
bare_nuclei               683
bland_chromatin           683
normal_nucleoli           683
mitoses                   683
outcome                   683
dtype: int64
```

In [43]:

```
#Show all the variables
Biopsy.columns
```

Out[43]:

```
Index(['clump_thickness', 'uniform_cell_size', 'unif
orm_cell_shape',
       'marg_adhesion', 'epithelial_cell_size', 'bar
e_nuclei',
       'bland_chromatin', 'normal_nucleoli', 'mitose
s', 'outcome'],
      dtype='object')
```

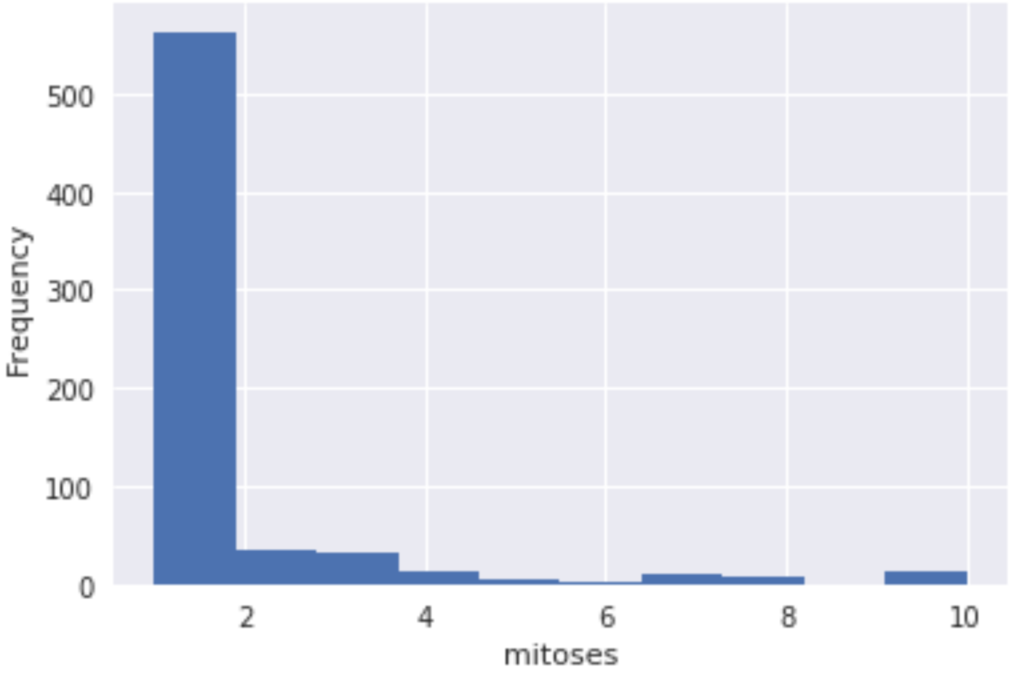
There are 10 variables tested for in this dataset

In [92]:

```
# Create a histogram
Biopsy['mitoses'].plot(kind = "hist")
plt.xlabel('mitoses') # add a label
```

Out[92]:

```
Text(0.5,0,'mitoses')
```



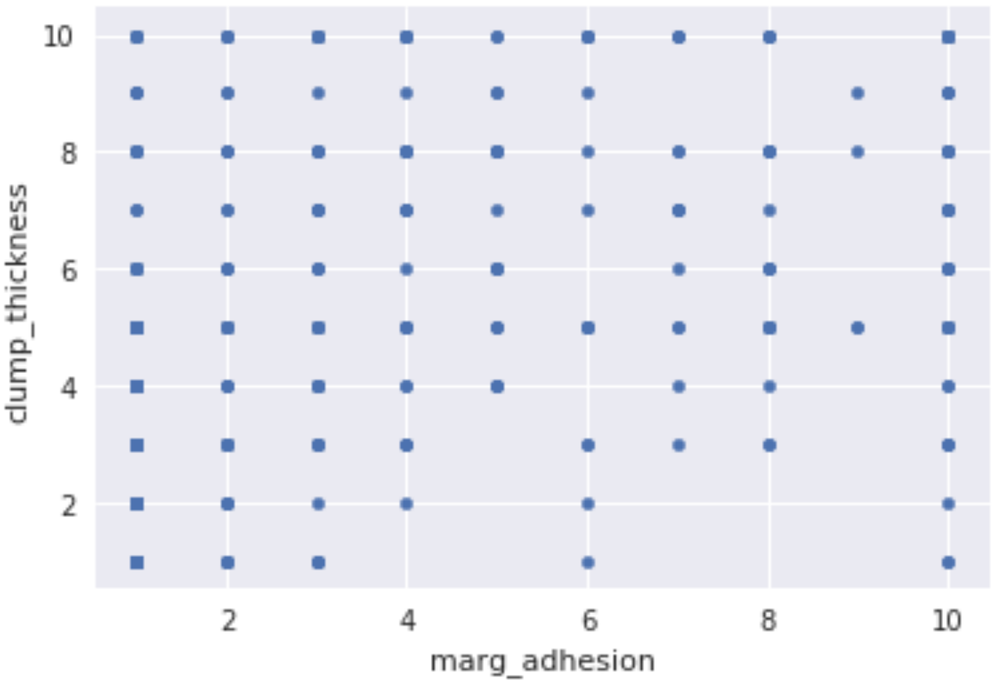
From this hisogram we can clearly see that there is a dramtic skew to the right, this is mostly likely caused by a few outlier that may have altered the data.

```
In [89]:
```

```python
# Use pandas to create a scatterplot
Biopsy.plot.scatter(x = 'marg_adhesion', y = 'clump_thickness')
```

```
Out[89]:
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe713ce
4630>
```



Marge Adhesion and clump thickness are two factors that both predicted to contribute cancer cells. Marg adhesions is the the mechanism by which two cells are able to stick to eachother. If cells loose their adhesive propertiies then this can lead to cells breaking aways and spreading cancer. Clump thickness is also a predictive propertie of cancer, since breast cancer is uncontrolled growth of breast cells, if there is clump thickness of many cells then there is a higher chance a patient will have breast cancer. However, while both of these variables can be linked to breat cancer predictions they are not related to eachother. The data in the scatterplot is not uniformed which shows no signinficant relationship

In [94]:

```
#Get a count of the number of 'M' & 'B' cells
Biopsy['outcome'].value_counts()
```

Out[94]:

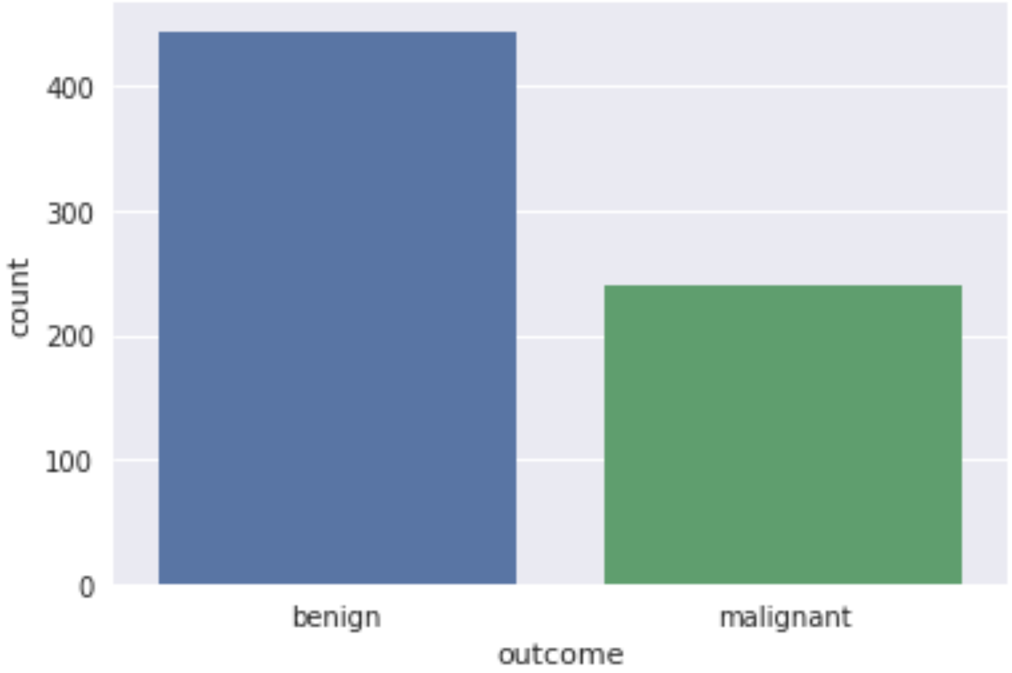
```
benign       444
malignant    239
Name: outcome, dtype: int64
```

In [93]:

```
#Visualize this count
sns.countplot(Biopsy['outcome'],label="Count")
```

Out[93]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe71342
c550>
```



There are 444 Beign and 239 Malignant tumors present in this dataset

In [82]:

```
Biopsy['outcome'].value_counts()
```

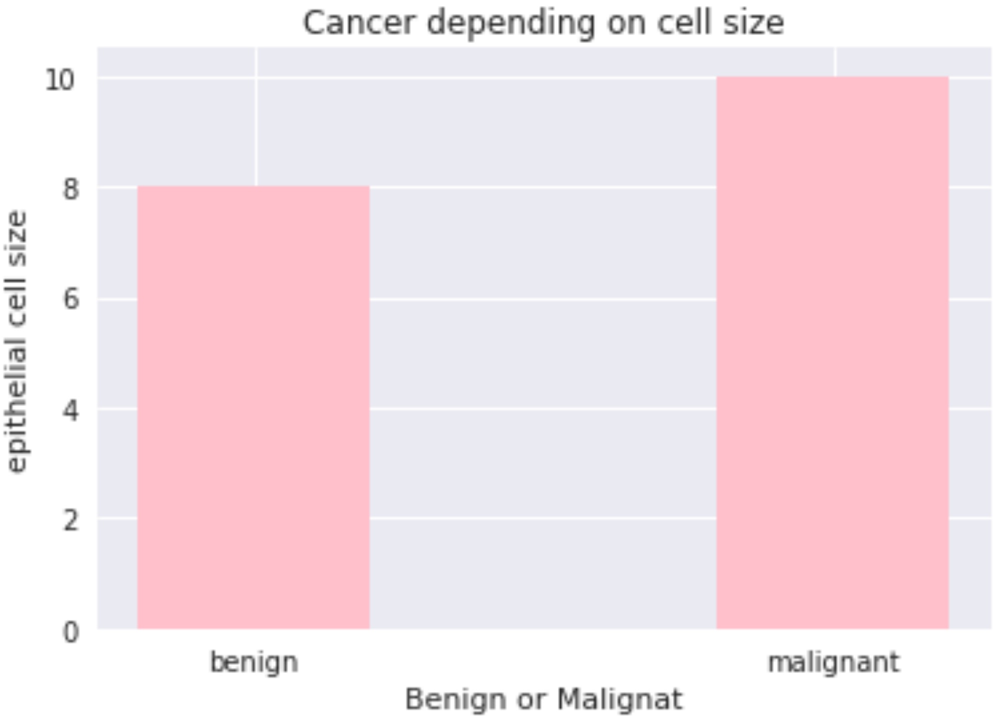
Out[82]:

```
benign       444
malignant    239
Name: outcome, dtype: int64
```

In [85]:

```
plt.bar(Biopsy.outcome, Biopsy.clump_thickness, color= 'pink', w
idth = 0.4)

plt.xlabel("Benign or Malignat")
plt.ylabel("Clump Thickness")
plt.title("Cancer depending on Clump Thickness")
plt.show()
```



After creating this graph one can determine that the greater the clump thickness is in paitents then the most likeley they will have cancer. Paitients will more clump thickness had more malignant cells.

In [ ]: