HAP 618 – Fall xxx

Final Project Paper

Breast Cancer

Background and problem description

Breast cancer is the 2nd most common type of cancer in the United States, with estimated new cases of 255,000 and estimated deaths of 41,000 for both male and female every year according to cdc.gov. These numbers can be reduced with several different methods. One of the many different methods is identifying if the tumor type is benign or malignant.

There are two types of tumor and they are benign and malignant. Benign is non-cancerous, it is a growth that is not cancer. It does not invade nearby tissue or spread to other parts of the body (cancer.gov). However, malignant is cancerous and it can invade and destroy nearby tissue and spread to other parts of the body (cancer.gov).

The problem that this project is identifying is, if there is a correlation between the tumor type and the nine attributes (independent variables) in this project. Understanding the correlation between the tumor type and the different attributes could potentially help identify the type of tumor more accurately. But also, analyze the data to understand what attributes correlate the most or least with the tumor type. According to New England Journal of Medicine, experts believe as many as 31 percent of all breast cancer cases are misdiagnosed, and many as 90,00 women are misdiagnosed with invasive breast cancer (The New York Times). Those are huge percentage and number of women's misdiagnosed.

The problem could be accomplished by using Unix system, Python, and HTML. The Unix system will mainly serve as the platform to storing all the data, Python will do the main job of aggregating the data, and HTML will display the graph and information nicely on the web. The dataset I will be using is from UCI and it is called *Breast Cancer Wisconsin (Original)* dataset.

Design of the solution:

For the project the codes that were used were; Python, Unix, and HTML. All three of these codes intertwin within each other to bring the project together. All three of these codes could have been executed on either server or on a client's computer, but for this project majority of the code is on the Unix server. There are few Python codes that are on my personal device. The project is not interactive, it consists of static webpages, except the bar charts refreshes with updated data. It was mainly designed for understanding the breast cancer data, analyzing the outcomes, correlation between tumor type and the attributes for breast cancer, and prediction model.

Python was used to aggregate the data to create bar charts and to create prediction model. The libraries that were used in the Python program are pandas, matplotlib.pyplot, and sklearn (sci-kit). The pandas library was used to import the breastcancer.csv file. The matplotlib.pyplot was used to create the bar charts and adding small detailed features such as; x-label, y-label, title, legend, assigning colors to specific class, and the x-tick marks. The graph is available for the users to see and analyze. Lastly, sklearn was used to create a prediction model by getting prediction probability on test data, and getting prediction accuracy score.

The Unix system was used to store the data and code (breast cancer data, html code, and python code). It is also responsible for running all the codes as it provided the support of displaying the result on the internet browser and creating the bar charts. The breastcancer.csv file was stored in the user's home of the Unix system. All the HTML codes were saved in /public_html and all Python codes were saved in /cgi-bin. Python code was used to read the breastcancer.csv file. Once the file was opened, it was used to aggregate the data, create bar charts, and save the bar chart as image.png in /public_html this gave the

image an URL. HTML code was used to create a web page and used the image URL to display the graph accordingly. In whole, Unix system took all the different parts of the code and displayed it as one.

HTML code, as stated before was used mainly to display the information and graph accordingly. HTML code could have been executed from the server or on a client computer, but in this project the code was executed from the server.

Implementation:

All the programs that were used in this project are important, being said that, without one it is not a complete project. However, if there is one important program that must be identified it would be the Python codes for the bar charts. The bar charts are used to analyze the data which is the main topic of the project. The following is one of the codes for bar chart.

```
110  import cgi
111  import pandas as pd
112  import matplotlib.pyplot as plt
113
114  mydata = pd.read_csv('/home/      /breastcancer.csv')
115
116  ###------------------------------------------------###
117  class_2 = mydata[mydata['Class'] == 2]
118  class_4 = mydata[mydata['Class'] == 4]
119  ###------------------------------------------------###
120  c2_uni_of_cellshape = class_2['Uniformity_of_Cell_Shape'].value_counts().sort_index()
121  cnt_uni_of_cellshape_c2 = c2_uni_of_cellshape.size
122  c4_uni_of_cellshape = class_4['Uniformity_of_Cell_Shape'].value_counts().sort_index()
123  cnt_uni_of_cellshape_c4 = c4_uni_of_cellshape.size
124  ###------------------------------------------------###
125  def uni_of_cellshape():
126      uni_of_cellshape = mydata['Uniformity_of_Cell_Shape'].value_counts().sort_index()
127      cnt_uni_of_cellshape = uni_of_cellshape.size
128      ct_labels = ['1','2','3','4','5','6','7','8','9','10']
129      plt.bar(range(cnt_uni_of_cellshape_c2), list(c2_uni_of_cellshape), color = 'orange',  label = 'Benign')
130      plt.bar(range(cnt_uni_of_cellshape_c4), list(c4_uni_of_cellshape), label = 'Malignant')
131      plt.xticks(range(cnt_uni_of_cellshape), ct_labels )
132      plt.title('Number of Instances by Uniformity of Cell Shape and Class')
133      plt.xlabel('Uniformity of Cell Shape')
134      plt.ylabel('Number of Instances')
135      plt.legend()
136      plt.savefig('/home/      /public_html/uni_of_cellshape.png')
137
138  uni_of_cellshape()
```

The following is the step-by-step on how the Python code works:

Line 110: Imports CGI
Line 111: Import pandas – used to import the breastcancer.csv
Line 112: Import matplotlib.pyplot – used to create the bar chart
Line 114: Reading the breastcancer.csv using pandas library
Line 117: Selecting the data only where Class = 2
Line 118: Selecting the data only where Class = 4
Line 120: Selecting only data from the column “Uniformity_of_Cell_Shape” and counting the values and sorting it by the index for data in class = 2.
Line 121: Counting the number of distinct values
Line 122: Selecting only data from the column “Uniformity_of_Cell_Shape” and counting the values and sorting it by the index for data in class = 4
Line 123: Counting the number of distinct values
Line 125: Defining a function “uni_of_cellshape”
Line 126 to Line 136: Is creating the bar chart including; xticks, title, x-label, y-label, legend, and saving the result as a image.png
Line 138: Printing the function “uni_of_cellshape”

Other Python codes that were implemented in this project is the prediction model codes. These Python codes were used to create a prediction model using a data mining method. This was an interesting perspective of the data, and further helps understand the significance in attribute type and result correlation to tumor type.

The HTML code was utilized to display the information and graphs and charts on the web, it was also used to make the web page look nice and organized. The HTML code was implemented towards bringing all the pieces of the project together into one web page. Also, how the images were saved as .png in html folder allowed flexibility to display the image on the web.

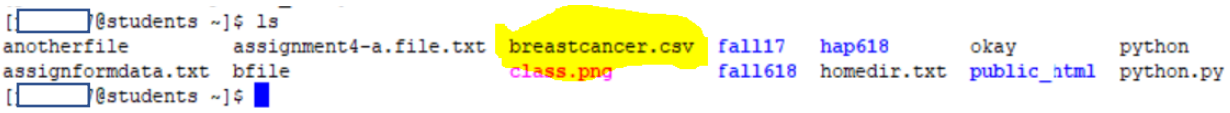
As it has been previously mentioned before, the implementation of the Unix system was to store the codes and run the codes. The Unix system also was used to store Python result for bar charts, which was in the format of .png, into HTML file. This easily made it possible to display the bar chart on the HTML web page with correct .png HTML URL.

Testing:

My project consists of four HTML pages. Majority of the codes are stored within Unix system and is running from the Unix system. Only codes that are not in the Unix system is prediction model codes.

Breast Cancer Dataset the highlighted “breastcancer.csv”:

a) Unix:



b) Content of the file breastcancer.csv:

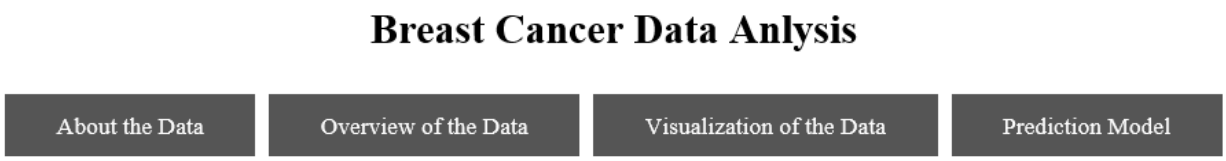
```
1369821,10,10,10,10,5,10,10,10,7,4
1371026,5,10,10,10,4,10,5,6,3,4
1371920,5,1,1,1,2,1,3,2,1,2
466906,1,1,1,1,2,1,1,1,1,2
466906,1,1,1,1,2,1,1,1,1,2
534555,1,1,1,1,2,1,1,1,1,2
536708,1,1,1,1,2,1,1,1,1,2
566346,3,1,1,1,2,1,2,3,1,2
603148,4,1,1,1,2,1,1,1,1,2
654546,1,1,1,1,2,1,1,1,8,2
654546,1,1,1,3,2,1,1,1,1,2
695091,5,10,10,5,4,5,4,4,1,4
714039,3,1,1,1,2,1,1,1,1,2
763235,3,1,1,1,2,1,2,1,2,2
776715,3,1,1,1,3,2,1,1,1,2
841769,2,1,1,1,2,1,1,1,1,2
888820,5,10,10,3,7,3,8,10,2,4
897471,4,8,6,4,3,4,10,6,1,4
897471,4,8,8,5,4,5,10,4,1,4
[redacted]@students ~]$ cat breastcancer.csv
```

Navigation:

a) Code: The following codes are identical in all three of my separate html web page.

```
<div class="navbar">
  <h2>Breast Cancer Data Anlysis</h2>
  <a href="http://students.hi.gmu.edu/[redacted]fp_about_the_data" class="button">About the Data</a>
  <a href="http://students.hi.gmu.edu/[redacted]fp_overview_of_the_data" class="button">Overview of the Data</a>
  <a href="http://students.hi.gmu.edu/[redacted]fp_visualization_of_data" class="button">Visualization of the Data</a>
  <a href="http://students.hi.gmu.edu/[redacted]fp_data_mining" class="button">Prediction Model</a>
<br>
<span style="font-size:10px">HAP 618 Final Project by [redacted]</span>

</div>
```

b) Result (output):



Bar Graph:

a) Python Code: Used the following code to create graph from the data. This python code format/layout was used to generate all the bar charts. Within the Unix system, the Python code for each graph had to be individually and separately saved onto different .cgi files. Otherwise, if more than one graph was in one .cgi file, the graph would overlap against each other. Therefore, I created total of 10 .cgi files (Number of Instances by Class + 9 different graphs for each attributes) for the bar charts.

```
import cgi
import pandas as pd
import matplotlib.pyplot as plt

mydata = pd.read_csv('/home/[redacted]breastcancer.csv')

###------------------------------------------------------###
class_2 = mydata[mydata['Class'] == 2]
class_4 = mydata[mydata['Class'] == 4]
###------------------------------------------------------###
c2_uni_of_cellshape = class_2['Uniformity_of_Cell_Shape'].value_counts().sort_index()
cnt_uni_of_cellshape_c2 = c2_uni_of_cellshape.size
c4_uni_of_cellshape = class_4['Uniformity_of_Cell_Shape'].value_counts().sort_index()
cnt_uni_of_cellshape_c4 = c4_uni_of_cellshape.size
###------------------------------------------------------###
def uni_of_cellshape():
    uni_of_cellshape = mydata['Uniformity_of_Cell_Shape'].value_counts().sort_index()
    cnt_uni_of_cellshape = uni_of_cellshape.size
    ct_labels = ['1','2','3','4','5','6','7','8','9','10']
    plt.bar(range(cnt_uni_of_cellshape_c2), list(c2_uni_of_cellshape), color = 'orange',  label = 'Benign')
    plt.bar(range(cnt_uni_of_cellshape_c4), list(c4_uni_of_cellshape), label = 'Malignant')
    plt.xticks(range(cnt_uni_of_cellshape), ct_labels )
    plt.title('Number of Instances by Uniformity of Cell Shape and Class')
    plt.xlabel('Uniformity of Cell Shape')
    plt.ylabel('Number of Instances')
    plt.legend()
    plt.savefig('/home/[redacted]/public_html/uni_of_cellshape.png')

uni_of_cellshape()
```

About the Data:

a) HTML Code: The following is part of the code, which consists of the code for the table.

```
<div class="main">
About the Dataset:
<br><table style="table-layout: fixed; border: 0.5px solid black;">
<tr>
  <td><font size="2"><b>Data Set:</b></td><td><font size="2">Breast Cancer Wisconsin (Original)</td>
</tr>
<tr>
  <td><font size="2"><b>Number of Instances:</b></td><td><font size="2">699</td>
</tr>
<tr>
  <td><font size="2"><b>Number of Attributes:</b></td><td><font size="2">10</td>
</td>
</table>
<br>
<table style="table-layout:fixed; border: 0.5px solid black;">
Attribute Information:
<tr>
  <td><font size="2"><b>Sample code number:</b></td><td><font size="2">Id number</td>
</tr>
<tr>
  <td><font size="2"><b>Clump Thickness:</b></td><td><font size="2">1-10</td>
</tr>
<tr>
  <td><font size="2"><b>Uniformity of Cell Size:</b></td><td><font size="2">1-10</td>
</tr>
<tr>
  <td><font size="2"><b>Uniformity of Cell Shape:</b></td><td><font size="2">1-10</td>
</tr>
<tr>
  <td><font size="2"><b>Marginal Adhesion:</b></td><td><font size="2">1-10</td>
</tr>
<tr>
  <td><font size="2"><b>Single Epithelial Cell Size:</b></td><td><font size="2">1-10</td>
</tr>
<tr>
  <td><font size="2"><b>Bare Nuclei:</b></td><td><font size="2">1-10</td>
</tr>
```

b) Result (output): http://students.hi.gmu.edu/~xxx/fp_about_the_data

About the Dataset:

Data Set:	Breast Cancer Wisconsin (Original)
Number of Instances:	699
Number of Attributes:	10

Attribute Information:

Sample code number:	Id number
Clump Thickness:	1-10
Uniformity of Cell Size:	1-10
Uniformity of Cell Shape:	1-10
Marginal Adhesion:	1-10
Single Epithelial Cell Size:	1-10
Bare Nuclei:	1-10
Bland Chromatin:	1-10
Normal Nucleoli:	1-10
Mitoses:	1-10
Class:	2 for benign 4 for malignant

Citation:

1. O. L. Mangasarian and W. H. Wolberg: "Cancer diagnosis via linear programming", SIAM News, Volume 23, Number 5, September 1990, pp 1 & 18.
2. William H. Wolberg and O.L. Mangasarian: "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp 9193-9196.
3. O. L. Mangasarian, R. Setiono, and W.H. Wolberg: "Pattern recognition via linear programming: Theory and application to medical diagnosis", in: "Large-scale numerical optimization", Thomas F. Coleman and Yuying Li, editors, SIAM Publications, Philadelphia 1990, pp 22-30.
4. K. P. Bennett & O. L. Mangasarian: "Robust linear programming discrimination of two linearly inseparable sets", Optimization Methods and Software 1, 1992, 23-34 (Gordon & Breach Science Publishers).

Visualization of the Data:

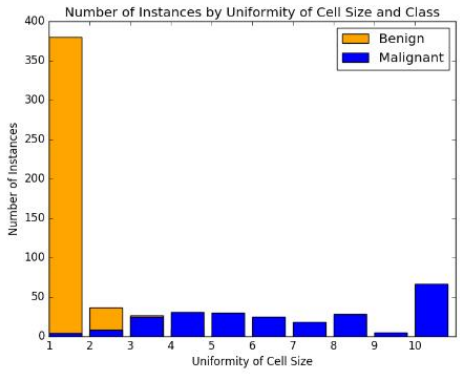
a) HTML Code:

```
<style>

table {
  border-collapse: collapse;
  width: 65%;
}
td, th {
  border: 1px solid black;
}
.textNode{
        text-align:left;
        padding:0;
        margin:0;
        vertical-align:top;
        font-size: 12px;
        word-wrap: break-word;
}
.button{
        background-color: #555555; /* Green */
        border: none;
        color: white;
        padding: 10px 30px;
        text-align: center;
        text-decoration: none;
        display: inline-block;
        font-size: 14px;
        margin: 4px 2px;
        cursor: pointer;
}
.navbar {
        overflow: hidden;
        background-color: #FFFFFF;
        position: fixed;
        top: 0;
        width: 100%;
}
.main {
        padding: 100px;
        margin-top: 30px;
}
</style>
```

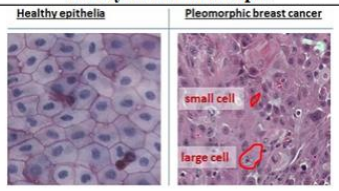
```
<body>
<center>

<div class="navbar">
  <h2>Breast Cancer Data Anlysis</h2>
  <a href="http://students.hi.gmu.edu/[redacted]fp_draft4" class="button">About the Data</a>
  <a href="http://students.hi.gmu.edu/[redacted]fp_draft3" class="button">Overview of the Data</a>
  <a href="http://students.hi.gmu.edu/[redacted]fp_draft2" class="button">Visualization of Data</a>
<br>
<span style="font-size:10px">HAP 618 Final Project by [redacted]</span>
</div>
<br><br>
<div class="main">
<table style="table-layout: fixed;">
<tr>
 <th colspan="3">  Number of Instance by Class  </th>
</tr>
<tr>
 <td><font size="2"><b>Benign</b></td>
 <td><font size="2">458</td>
 <th rowspan="2"> <center>Analysis on Graph</center> </th>
</tr>
<tr>
 <td><font size="2"><b>Malignant</b></td>
 <td><font size="2">241</td>
</tr>
<tr>
 <td colspan="2"><center><img src="http://students.hi.gmu.edu/[redacted]/class.png"  width = 400></center></td>
 <td class="textnode"> The bar graph on the left shows the number of instances by class.
 There are 458 beign instances and 241 malignant instances.</td>
</tr>
</table>
<br><!--Clump Thickness Table-->
<table style="table-layout: fixed;">
<tr>
  <th colspan="2">  Number of Instances by Clump Thickness and Class</th>
  <th><center>Analysis on Graph</center></th>
</tr>
<tr>
  <td colspan="2"><center><img src="http://students.hi.gmu.edu/[redacted]/clump_thickness.png" width = 400></center></td>
  <td class="textnode" > According to the graph, there is a correlation between tumor type and the clump thickness. Based
  on the graph, benign tumor type clump thickness is smaller than malignant which has bigger clump thickness.</td>
</tr>
</table>
</div>
</body>
</html>
```

b) Result (output): http://students.hi.gmu.edu/~xxx/fp_visualization_of_data

Number of Instances by Uniformity of Cell Size and Class

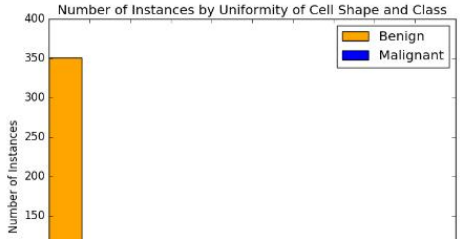


Analysis of the Graph

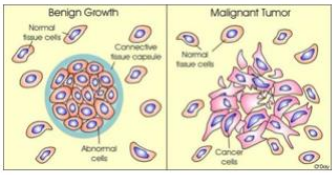


According to the graph, there is a correlation between tumor type and the uniformity of cell size. Based on the graph, benign is most likely to fall in the lower number range. As for malignant, it looks to be very consistenly spread across the range, but it definitely falls more towards the higher number of uniformity of cell size.

Number of Instances by Uniformity of Cell Shape and Class



Analysis of the Graph



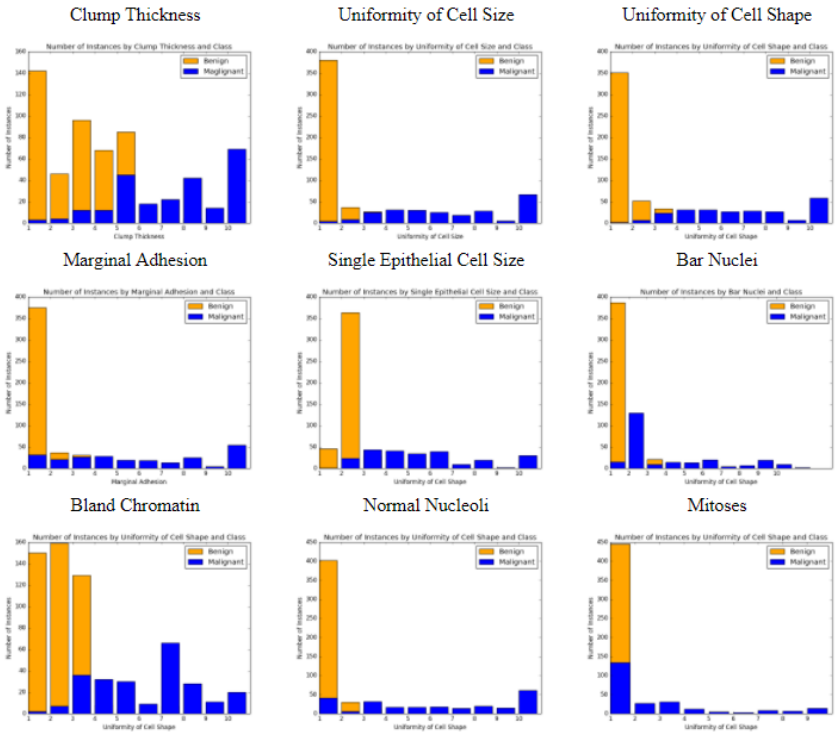
According to the graph, there is a correlation between tumor type and the uniformity of cell shape. Based on the graph, benign is most likely to fall in the lower number of

Overview of the data:

a) HTML Code: Arranged the graphs using the tables html code.

```
<div class="main">
<table style="table-layout: fixed;">
<tr>
  <td><font size="2"><center>Clump Thickness</center></td>
  <td><font size="2"><center>Uniformity of Cell Size</center></td>
  <td><font size="2"><center>Uniformity of Cell Shape</center></td>
</tr>
<tr>
  <td><img src="http://students.hi.gmu.edu/[redacted]/clump_thickness.png" width = 200></td>
  <td><img src="http://students.hi.gmu.edu/[redacted]/uni_of_cellsize.png" width = 200></td>
  <td><img src="http://students.hi.gmu.edu/[redacted]/uni_of_cellshape.png" width= 200></td>
</tr>
<tr>
  <td><font size="2"><center>Marginal Adhesion</center></td>
  <td><font size="2"><center>Single Epithelial Cell Size</center></td>
  <td><font size="2"><center>Bar Nuclei</center></td>
</tr>
<tr>
  <td><img src="http://students.hi.gmu.edu/[redacted]/marginal_adhesion.png" width = 200></td>
  <td><img src="http://students.hi.gmu.edu/[redacted]/single_epithelial.png" width = 200></td>
  <td><img src="http://students.hi.gmu.edu/[redacted]/bar_nuclei.png" width= 200></td>
</tr>
<tr>
  <td><font size="2"><center>Bland Chromatin</center></td>
  <td><font size="2"><center>Normal Nucleoli</center></td>
  <td><font size="2"><center>Mitoses</center></td>
</tr>
<tr>
  <td><img src="http://students.hi.gmu.edu/[redacted]/bland_chromatin.png" width = 200></td>
  <td><img src="http://students.hi.gmu.edu/[redacted]/normal_nucleoli.png" width = 200></td>
  <td><img src="http://students.hi.gmu.edu/[redacted]/mitoses.png" width= 200></td>
</tr>
</table>
</center>

</div>
</body>
</html>
```

b) Result (output): http://students.hi.gmu.edu/~xxx/fp_overview_of_the_data



Prediction Model

a) Python Code

```
from sklearn.metrics import accuracy_score
from sklearn.naive_bayes import GaussianNB
import matplotlib.pyplot as plt

size = y_test.size


# Initialize our classifier
gnb = GaussianNB()
# Train our classifiera
model = gnb.fit(x_train, y_train)
# Make predictions
preds = gnb.predict(x_test)
preds2 = gnb.predict_proba(x_test)
print(preds2)
print(accuracy_score(y_test, preds))

################ PLot GaussianNB

x = [i[0] for i in preds2]
y = [i[1] for i in preds2]


plt.scatter(x,y, label = 'Prediction Probability')
plt.title('GaussianNB Prediction Probability')
plt.xlabel('Probability')
plt.ylabel('Probability')

plt.scatter(x,y, label = 'Prediction Probability')
plt.scatter(range(size), y_test, alpha = 0.4, label = 'Actual Result')
plt.text(160,0.35, 'Accuracy Score: 0.9619')
plt.rcParams["figure.figsize"] = (10,5)
plt.legend()
plt.title('GaussianNB')
plt.ylabel('Prediction Probability')
plt.xlabel('Number of Cases')
plt.xticks()
plt.show()
```

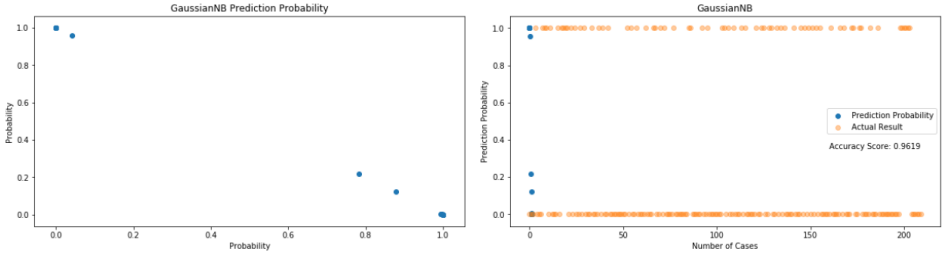
b) HTML Code:

```
<div class="main">
<center><b><span style="font-size:26px">Prediction Model</span></b></center>
<table style="table-layout: fixed;">
<tr>
  <td><img src="https://image.ibb.co/iCuoxw/Gaussian_NB_Prob.png" width = 500></td>
  <td><img src="https://image.ibb.co/kPqTxw/Gaussian_NB_Model.png" width = 500></td>
</tr>
<br>
<tr>
  <td><img src="https://image.ibb.co/dM06jb/KNeighbors_N30_Prob.png" width = 500></td>
  <td><img src="https://image.ibb.co/iy0Txw/KNeighbors_N30.png" width = 500></td>
</tr>
<br>
<tr>
  <td><img src="https://image.ibb.co/eMX8xw/Support_Vector_Machine_Prob.png" width = 500></td>
  <td><img src="https://image.ibb.co/gnBlHw/Support_Vector_Machine.png" width = 500></td>
</td>
</table>

</div>
```

- The images were saved on my personal laptop and uploaded on the web.

c) Result (output): http://students.hi.gmu.edu/~xxx/fp_data_mining

Prediction Model



Conclusions & Future Work (2-3 paragraphs or more)

Based on the bar chart results there is a significant correlation between the two tumor types and attribute results. The pattern that is seemingly repetitive in relation to tumor type, for almost every attribute, was interesting to analyze. Tumor type benign attribute results was relatively on the lower end of the number range, however for malignant attribute results were opposite of benign where malignant results were relatively on the higher end of the number range. In the project prediction model was also created. Based on the accuracy score for the prediction, it was roughly ≈ 0.96 for all three (GaussianNB, KNeighbor Classifier N=30, and Support Vector Machine) methods used in the project. This result shows that attribute result is reliable source to identifying if the tumor is benign or malignant

This project has potential growth and room for further development in the field of healthcare. As stated early on, breast cancer is becoming a nationwide problem and more or less an epidemic. It is definitely preventable and treatable with correct diagnosis and treatments. Considering the data used for this project was fairly old (between for this project is fairly old (between 1989 to 1991), it would be interesting to analyze with a recent dataset with same attributes and format.

Breast cancer is a growing problem, and there is no “right” solution to tackling this issue. However, decreasing misdiagnosis and increase in correct diagnosis can significantly help an individual go towards the correct path of treatment, as necessary.

References:

1. Dataset Citation: https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original)
    a. 1. O. L. Mangasarian and W. H. Wolberg: "Cancer diagnosis via linear programming", SIAM News, Volume 23, Number 5, September 1990, pp 1 & 18.
    b. 2. William H. Wolberg and O.L. Mangasarian: "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp 9193-9196.
    c. 3. O. L. Mangasarian, R. Setiono, and W.H. Wolberg: "Pattern recognition via linear programming: Theory and application to medical diagnosis", in: "Large-scale numerical optimization", Thomas F. Coleman and Yuying Li, editors, SIAM Publications, Philadelphia 1990, pp 22-30.
    d. 4. K. P. Bennett & O. L. Mangasarian: "Robust linear programming discrimination of two linearly inseparable sets", Optimization Methods and Software 1, 1992, 23-34 (Gordon & Breach Science Publishers).
2. HTML. w3schools.com. https://www.w3schools.com/html/. Accessed November 22, 2017.
3. What is Breast Cancer? Breastcancer.org. https://www.cancer.org/cancer/breast-cancer/about/what-is-breast-cancer.html. Published September 21, 2017. Accessed November 22, 2017.
4. Breast Cancer Statistics. cdc.gov. https://www.cdc.gov/cancer/breast/statistics/index.htm. Published June 7, 2017. Accessed November 22, 2017.
5. New England Journal of Medicine
6. www.preventcancer.org
7. www.google.com for the images
8. http://scikit-learn.org/stable/index.html
9. https://stackoverflow.com/questions/
10. https://pandas.pydata.org/pandas-docs/stable/
11. Class lecture notes on python, unix, and html.