Breast Cancer Cell Data Analysis and Visualization Using R

Kyongeui Hong[1], Kangbin Yim[2#] and Keunhyuk Kim[3#]

[1]Northern Valley Regional High School at Demarest, USA
[2]Soonchunhyang University
[3]KakaoPay Corporation
[#]Advisor

ABSTRACT

Breast cancer is the most frequently occurring cancer in women. If the cancer is diagnosed and treated at an early stage, the patient has a survival rate of 99% after 5 years, but it significantly drops to 29% when it reaches a distant stage. Thus, it is very important to detect 'positive cancer cells' in the early stage, so I analyzed the 569 breast cancer cell data provided by the University of Wisconsin using R-Studio. Through this program, I visualized the relationships between 10 different cell characteritics, and researched the kind of relationship between radius and positive cancer cells by utilizing graphs. Also, I created a predictive model that can detect positive cancer cells based on logistic regression and training original 569 data. Finally, I recommended the adequate age (35-39) for the breast cancer examination by analyzing the breast cancer statistics. Through this research, I derived effective breast cancer predicting model and deeply explored big data analysis method and data-mining on R-Studio

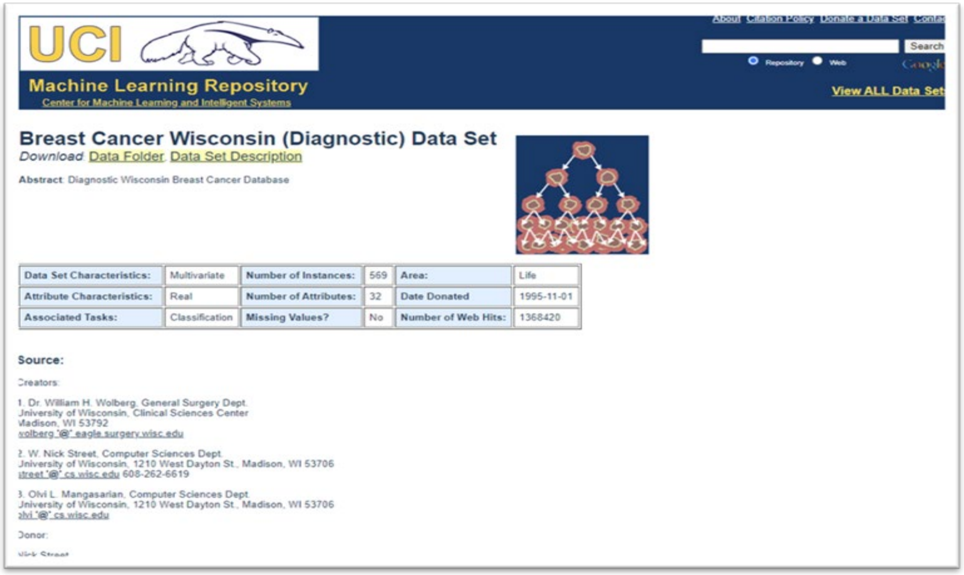
Introduction

Although the average life expectancy of modern people has dramatically increased due to the rapid development of modern medical science, 'cancer' will cause the highest death rate in Korea in 2020 (27% of all deaths), indicating it is still a complex disease for humans to overcome. Because cancer cells grow rapidly and irregularly and have the characteristics of easily metastasizing to surrounding tissues and bones, the later the cancer is detected, the higher the mortality rate is. For instance, the survival rate of breast cancer, the number one cancer in women, reaches 99% after 5 years if detected at an early stage (localized). However, the survival rate drops significantly to 29% when diagnosed at a later stage (distant). (American Cancer Society , Under the assumption that early cancer diagnosis can have a great effect on the patient's prognosis, I created a model using R-Studio that can predict breast cancer cells at an early stage by analyzing and visualizing breast cancer cell big data published by the University of Wisconsin.

Analysis Process

Collecting the Data

The University of Wisconsin is currently releasing the 1995 'Breast Cancer Wisconsin Data Set' donated by Dr. William H. Wolberg to the 'UCI Machine Learning Repository' for research purposes. [2] The reason for selecting this particular data set is because the access is given to the public. By downloading the data, various relationships between breast cancer cells were installed in R-Studio.


UCI
Machine Learning Repository
Center for Machine Learning and Intelligent Systems
About Citation Policy Donate a Data Set Contac
Search
○ Repository ● Web
Google
View ALL Data Set

Breast Cancer Wisconsin (Diagnostic) Data Set

Download: Data Folder, Data Set Description

Abstract: Diagnostic Wisconsin Breast Cancer Database

Data Set Characteristics:	Multivariate	Number of Instances:	569	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	32	Date Donated	1995-11-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	1368420

Source:

Creators:

1. Dr. William H. Wolberg, General Surgery Dept.
University of Wisconsin, Clinical Sciences Center
Madison, WI 53792
wolberg '@' eagle.surgery.wisc.edu

2. W. Nick Street, Computer Sciences Dept.
University of Wisconsin, 1210 West Dayton St., Madison, WI 53706
street '@' cs.wisc.edu 608-262-6619

3. Olvi L. Mangasarian, Computer Sciences Dept.
University of Wisconsin, 1210 West Dayton St., Madison, WI 53706
olvi '@' cs.wisc.edu

Donor:

Nick Street

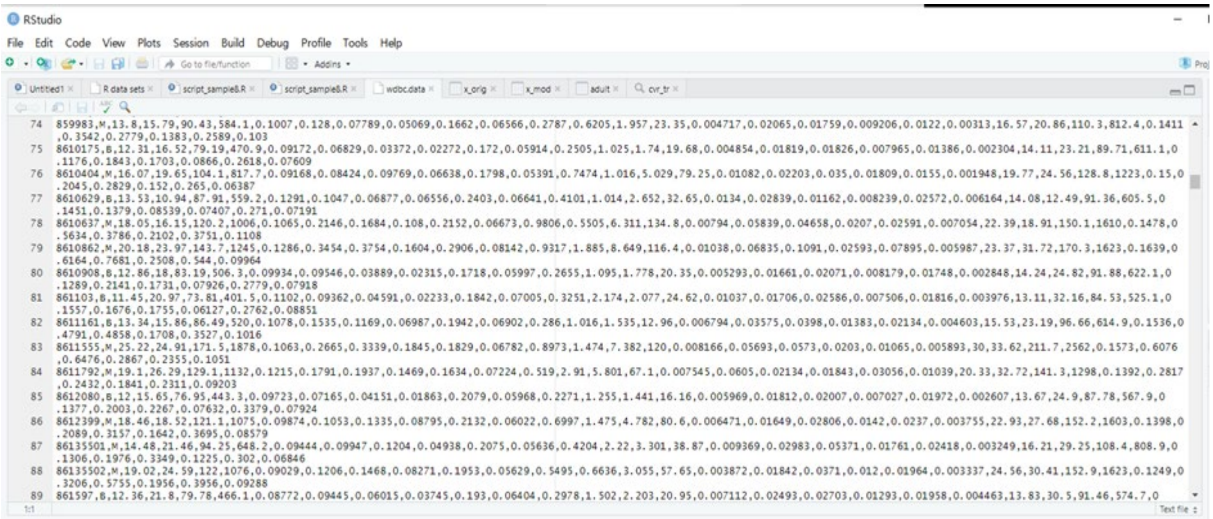
Figure 1. Website Screen Shot where you can download Wisconsin Breast Cancer data. Select wbdc.data and wdbc.names files from the data on this site.

This file contains the results of analyzing a total of 569 breast cancer cell samples, and each sample is classified into the following 10 categories.

Table 1. 10 Features of each cell nucleus in wdbc.datd

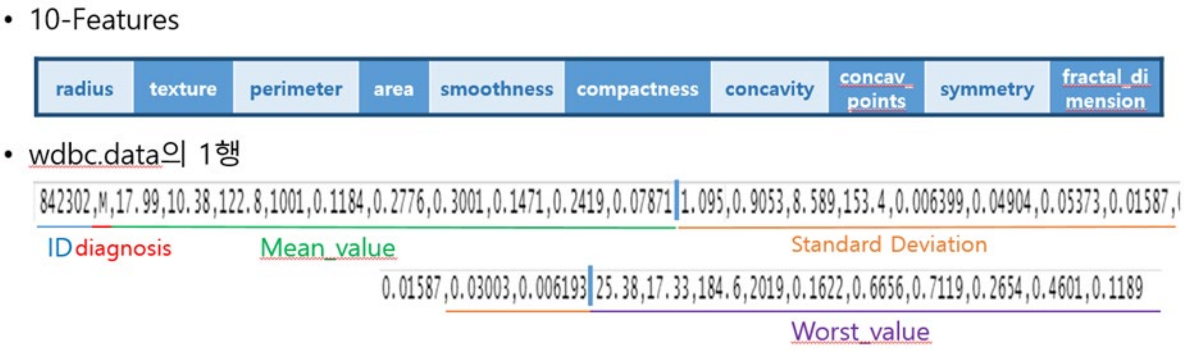
Radius	Compactness
Texture	Concavity
Perimeter	Concave Points
Area	Symmetry
Smoothness	Fractal Dimension

Since each breast cancer cell data contained in the file is raw data that does not have a separate format, information of 569 cancer cells was displayed in a long list in R-Studio.



```
859983,M,13.8,15.79,90.43,584.1,0.1007,0.128,0.07789,0.05069,0.1662,0.06566,0.2787,0.6205,1.957,23.35,0.004717,0.02065,0.01759,0.009206,0.0122,0.00313,16.57,20.86,110.3,812.4,0.1411
,0.3542,0.2779,0.1383,0.2589,0.103
8610175,B,12.31,16.52,79.19,470.9,0.09172,0.06829,0.03372,0.02272,0.172,0.05914,0.2505,1.025,1.74,19.68,0.004854,0.01819,0.01826,0.007965,0.01386,0.002304,14.11,23.21,89.71,611.1,0
.1176,0.1843,0.1703,0.0866,0.2618,0.07609
8610404,M,16.07,19.65,104.1,817.7,0.09168,0.08424,0.09769,0.06638,0.1798,0.05391,0.7474,1.016,5.029,79.25,0.01082,0.02203,0.035,0.01809,0.0155,0.001948,19.77,24.56,128.8,1223,0.15,0
.2045,0.2829,0.152,0.265,0.06387
8610629,B,13.53,10.94,87.91,559.2,0.1291,0.1047,0.06877,0.06556,0.2403,0.06641,0.4101,1.014,2.652,32.65,0.0134,0.02839,0.01162,0.008239,0.02572,0.006164,14.08,12.49,91.36,605.5,0
.1451,0.1379,0.08539,0.07407,0.271,0.07191
8610637,M,18.05,16.15,120.2,1006,0.1065,0.2146,0.1684,0.108,0.2152,0.06673,0.9806,0.5505,6.311,134.8,0.00794,0.05839,0.04658,0.0207,0.02591,0.007054,22.39,18.91,150.1,1610,0.1478,0
.5634,0.3786,0.2102,0.3751,0.1108
8610862,M,20.18,23.97,143.7,1245,0.1286,0.3454,0.3754,0.1604,0.2906,0.08142,0.9317,1.885,8.649,116.4,0.01038,0.06835,0.1091,0.02593,0.07895,0.005987,23.37,31.72,170.3,1623,0.1639,0
.6164,0.7681,0.2508,0.544,0.09964
8610908,B,12.86,18,83.19,506.3,0.09934,0.09546,0.03889,0.02315,0.1718,0.05997,0.2655,1.095,1.778,20.35,0.005293,0.01661,0.02071,0.008179,0.01748,0.002848,14.24,24.82,91.88,622.1,0
.1289,0.2141,0.1731,0.07926,0.2779,0.07918
861103,B,11.45,20.97,73.81,401.5,0.1102,0.09362,0.04591,0.02233,0.1842,0.07005,0.3251,2.174,2.077,24.62,0.01037,0.01706,0.02586,0.007506,0.01816,0.003976,13.11,32.16,84.53,525.1,0
.1557,0.1676,0.1755,0.06127,0.2762,0.08851
8611161,B,13.34,15.86,86.49,520,0.1078,0.1535,0.1169,0.06987,0.1942,0.06902,0.286,1.016,1.535,12.96,0.006794,0.03575,0.0398,0.01383,0.02134,0.004603,15.53,23.19,96.66,614.9,0.1536,0
.4791,0.4858,0.1708,0.3527,0.1016
8611555,M,25.22,24.91,171.5,1878,0.1063,0.2665,0.3339,0.1845,0.1829,0.06782,0.8973,1.474,7.382,120,0.008166,0.05693,0.0573,0.0203,0.01065,0.005893,30,33.62,211.7,2562,0.1573,0.6076
,0.6476,0.2867,0.2355,0.1051
8611792,M,19.1,26.29,129.1,1132,0.1215,0.1791,0.1937,0.1469,0.1634,0.07224,0.519,2.91,5.801,67.1,0.007545,0.0605,0.02134,0.01843,0.03056,0.01039,20.33,32.72,141.3,1298,0.1392,0.2817
,0.2432,0.1841,0.2311,0.09203
8612080,B,12,15.65,76.95,443.3,0.09723,0.07165,0.04151,0.01863,0.2079,0.05968,0.2271,1.255,1.441,16.16,0.005969,0.01812,0.02007,0.007027,0.01972,0.002607,13.67,24.9,87.78,567.9,0
.1377,0.2003,0.2267,0.07632,0.3379,0.07924
8612399,M,18.46,18.52,121.1,1075,0.09874,0.1053,0.1335,0.08795,0.2132,0.06022,0.6997,1.475,4.782,80.6,0.006471,0.01649,0.02806,0.0142,0.0237,0.003755,22.93,27.68,152.2,1603,0.1398,0
.2089,0.3157,0.1642,0.3695,0.08579
86135501,M,14.48,21.46,94.25,648.2,0.09444,0.09947,0.1204,0.04938,0.2075,0.05636,0.4204,2.22,3.301,38.87,0.009369,0.02983,0.05371,0.01761,0.02418,0.003249,16.21,29.25,108.4,808.9,0
.1306,0.1976,0.3349,0.1225,0.302,0.06846
86135502,M,19.02,24.59,122,1076,0.09029,0.1206,0.1468,0.08271,0.1953,0.05629,0.5495,0.6636,3.055,57.65,0.003872,0.01842,0.0371,0.012,0.01964,0.003337,24.56,30.41,152.9,1623,0.1249,0
.3206,0.5755,0.1956,0.3956,0.09288
861597,B,12.36,21.8,79.78,466.1,0.08772,0.09445,0.06015,0.03745,0.193,0.06404,0.2978,1.502,2.203,20.95,0.007112,0.02493,0.02703,0.01293,0.01958,0.004463,13.83,30.5,91.46,574.7,0
```

Figure2. Raw data in wdbc.data. It can be seen as a state before data processing.



- 10-Features

radius	texture	perimeter	area	smoothness	compactness	concavity	concav points	symmetry	fractal_dimension

- wdbc.data의 1행

```
842302,M,17.99,10.38,122.8,1001,0.1184,0.2776,0.3001,0.1471,0.2419,0.07871,1.095,0.9053,8.589,153.4,0.006399,0.04904,0.05373,0.01587,
0.01587,0.03003,0.006193,25.38,17.33,184.6,2019,0.1622,0.6656,0.7119,0.2654,0.4601,0.1189
```

Figure3. The meaning of raw data in wdbc.data was analyzed and expressed. It can be seen that the first row of wdbc.data represents ID, Diagnosis, Mean_value, Standard Deviation, and Worst_value.

From the data provided by the University of Wisconsin, I brought one particular row. Summarizing the numerical information listed in the photo, column 1 represents the unique ID of the cell. The letter in column 2 indicates whether the cell is positive or negative. M stands for benign tumor (positive cancer) and N stands for negative tumor (negative cancer). The next 3 to 12 columns are the average of the measured feature values based on 10 criteria (radius, texture, concavity…etc). After the average values, columns 12-22 provide standard deviation and 23-32 represent the worst value of the breast cancer cell features.

Since there are 32 pieces of information per breast cell and there are a total of 569 observations, 10,208 data were used for analysis. It is known that the more data there is, the higher the accuracy of the analysis, so I think it is important to collect data continuously and accumulate the data.

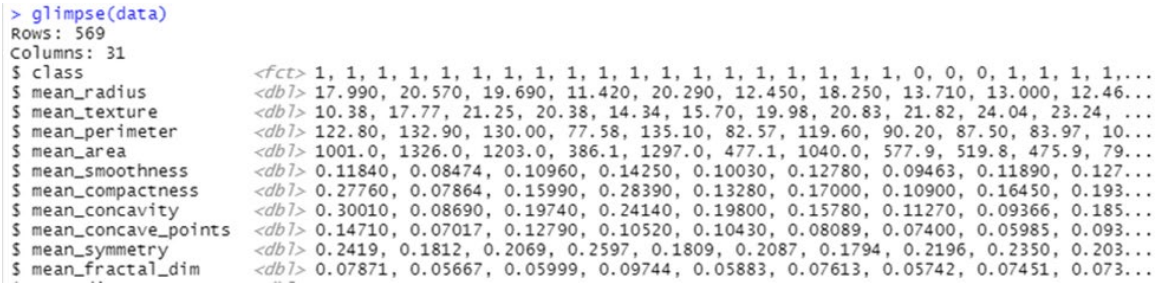
Setting the R-Studio and Refining the Data

To analyze the data provided by the University of Wisconsin by applying the R, I installed packages and libraries that are essential to imply R on PC. I utilized these libraries to analyze the relationship between the features and malignant tumors (positive cancer cells) occurring in the breast.

Table2. Description of the packages and libraries required to run R

library(dplyr)	Data Processing
library(ggplot2)	Data Visualization
library(MASS)	Package for Logistic Regression
library(boot)	Package for selecting meaningful variables
library(data.table)	Create Data Table
library(gridExtra)	Display the graph in grid format

For data analysis, it was necessary to refine the data based on the purpose of how to use the data. First, the ID number assigned to each cell was removed because it was not necessary for analysis, and M and N indicating positive and negative were converted to M=1, N=0. Then, I utilized ‘read.table ()’ to read the breast cancer cell data and refined them with a function called ‘tbl_df ()’. Finally, I named the 10 criteria on the left side of the table.



```
> glimpse(data)
Rows: 569
Columns: 31
$ class               <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1,...
$ mean_radius         <dbl> 17.990, 20.570, 19.690, 11.420, 20.290, 12.450, 18.250, 13.710, 13.000, 12.46...
$ mean_texture        <dbl> 10.38, 17.77, 21.25, 20.38, 14.34, 15.70, 19.98, 20.83, 21.82, 24.04, 23.24, ...
$ mean_perimeter      <dbl> 122.80, 132.90, 130.00, 77.58, 135.10, 82.57, 119.60, 90.20, 87.50, 83.97, 10...
$ mean_area           <dbl> 1001.0, 1326.0, 1203.0, 386.1, 1297.0, 477.1, 1040.0, 577.9, 519.8, 475.9, 79...
$ mean_smoothness     <dbl> 0.11840, 0.08474, 0.10960, 0.14250, 0.10030, 0.12780, 0.09463, 0.11890, 0.127...
$ mean_compactness    <dbl> 0.27760, 0.07864, 0.15990, 0.28390, 0.13280, 0.17000, 0.10900, 0.16450, 0.193...
$ mean_concavity      <dbl> 0.30010, 0.08690, 0.19740, 0.24140, 0.19800, 0.15780, 0.11270, 0.09366, 0.185...
$ mean_concave_points <dbl> 0.14710, 0.07017, 0.12790, 0.10520, 0.10430, 0.08089, 0.07400, 0.05985, 0.093...
$ mean_symmetry       <dbl> 0.2419, 0.1812, 0.2069, 0.2597, 0.1809, 0.2087, 0.1794, 0.2196, 0.2350, 0.203...
$ mean_fractal_dim    <dbl> 0.07871, 0.05667, 0.05999, 0.09744, 0.05883, 0.07613, 0.05742, 0.07451, 0.073...
```

Figure4. Data after processing data is shown using the glimpse() function.

By using the glimpse(data) function, the raw data has been refined and become readable. Compared to the original data set (pic2), the refined data set provided specific information about the numbers and enabled me to effectively analyze the breast cancer cells data by creating a table. The R-Studio has an advantage that could easily refine and sort a large amount of data by using only a few functions.

Data Visualization

After completing the purification of 569 breast cancer cells’ data, the next step is visualizing this refined data set. The reason for visualization is to find the correlation between breast cancer-positive cells and 10 variables (criteria). Since the association will be derived based on abundant data (10,208), it will be highly reliable. Using these relationships, the doctor will use their expertise to detect breast cancer patients in the early stage.

pairs ()

pairs () function provides scatterplots and relationships between variables by producing graphs based on data. I derived relationships between these 10 variables by using this function.

Code:

```
pairs(data%>% dplyr::select(class, starts_with('mean_')) %>%
          sample_n(min(1000, nrow(data))),
          lower.panel=function(x,y){points(x,y); abline(0, 1, col='red')},
          upper.panel=panel.cor
```

After extracting 10 average values (mean_radius, mean_texture…) of class variables using select(), sample.n() is used to ensure that the number of extracted values does not exceed 1000. In lower.panel, a linear function graph is drawn with the (x,y) point and the abline() function with the point() function, indicating the relationship between each variable. upper.panel shows the correlation coefficient between variables, which is named panel.cor. To execute pairs (), there should be the value of panel.cor, so I coded as follows.

Code:

```
panel.cor <- function(x,y,digits=2, prefix='', cex,cor…){
          usr<-par("usr"); on.exit(par(usr))
          par(usr=c(0, 1, 0, 1))
          r<-abs(cor(x,y))
          txt<-format(c(r,0.123456789), digits=digits)[1]
          txt<-paste0(prefix, txt)
          if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
          text(0.5, 0.5, txt, cex=cex.cor * r)
```

The code that derived panel.cor is excerpted from 'Learning the Data Science' p.219. The value of panel.cor represents the correlation efficiency varies from 0 to 1. If the value is close to 1, it can be interpreted that the relationship between the two variables is high, and when it is close to 0, they are not relatable.

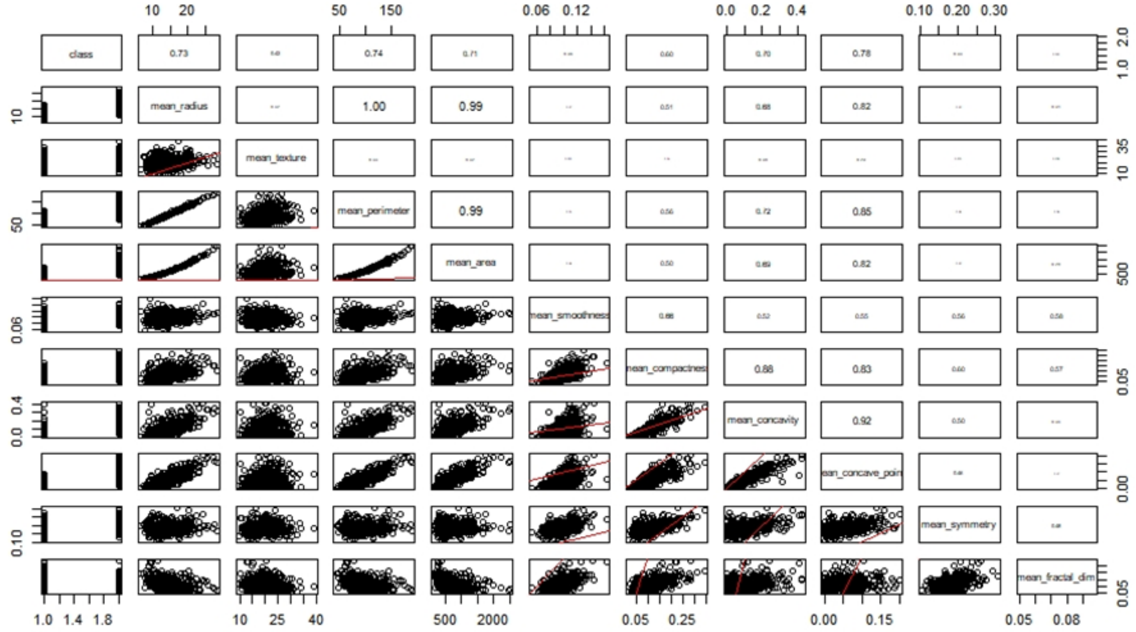


Figure 5. Data is visualized using pairs().

Below the diagonal is lower.panel, and the data was visualized as scatterplots using point () function. The upper part is upper.panel, and the correlation between variables is quantified with panel.cor. panel.cor maximized the correlation coefficient between two variables by increasing the size of the number as the value of panel.cor was greater. The above graph depicts several unique relationships between the variables that were in the data set.

i) the correlation coefficient (panel.cor value) of mean_radius & mean_perimeter, mean_radius & mean_area is 1.00, 0.99. Although the cancer cells are not perfectly circled, the relationship between the cell's area, perimeter, and radius maintains a direct relationship.

ii) mean_texture, mean_smoothness, mean_symmetry, and mean_fractal_dm are independent variables. Analyzing the correlations of each variable using pairs (), these 4 variables had very low relationships with other variables. This indicates these are independent variables, so they need to be examined individually to verify whether the cell is positive or not.

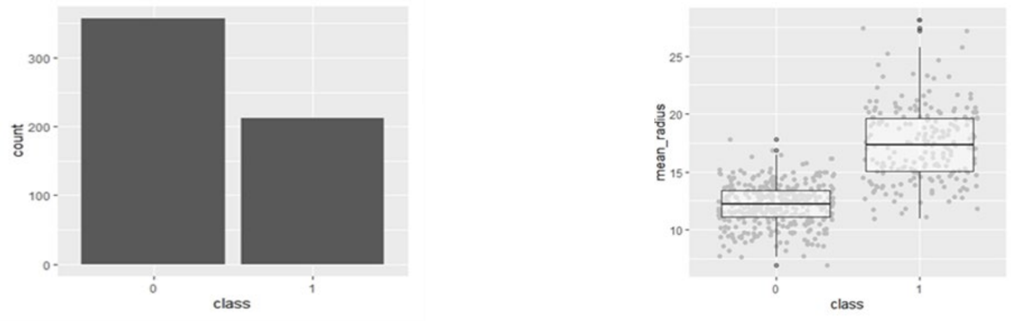
The pairs () function provided that radius has the most relationships with other variables. Thus, deriving a relationship between the radius and the positive cancer cells will reveal a lot of information. So, I visualized the relationship between the two as a graph.

geom_jitter (), geom_bar, geom_boxplot (), geom_smooth ()

Code:

```
library(ggplot2)
library(dplyr)
library(gridExtra)
p1 <- data%>% ggplot(aes(class)) + geom_bar()
p2 <- data %>% ggplot(aes(class, mean_radius)) + geom_jitter(col=’gray’)+geom_boxplot(alpha=.5)
grid.arrange(p1,p2, ncol=2)
```

geo.bar () is a function that draws a bar graph. ggplot2 decides what data I am going to use to draw graphs. After selecting particular data, choose what will be on the x, and y-axis by aesthetic function and visualize the data with geom_jitter.



Graph(1)

Graph(2)

Figure6. The correlation of count & class,mean_radius & class was visualized using function geom_bar() and geom_boxplot().

Graph (1) shows that, out of 569 data provided by the University of Wisconsin, there were more than 350 negative cells (class=0) and about 200 positive cells (class=1) that could cause breast cancer. Therefore, when patients visit a hospital with suspected breast cancer symptoms, there is a 35% possibility of being diagnosed with cancer.

Graph (2) shows the relationship between mean_radius and class (cancer positivity) as a scattergram. In the case of negative cancer cells, it was intensively distributed between 10-15, and for the positive cells, it was found that it was more widely distributed than negative between 12.5-22. Through the graph above, it was confirmed that the smaller the cell radius, the lower the probability of positive cancer cells. Also, in the pairs() function, it has been proven that mean_radius has a very close relationship with mean_perimeter and mean_area. Therefore, if cells are analyzed through radius, analysis through perimeter and area can be skipped. That means although the existing method diagnoses whether cancer is benign through 10 variables, it will be sufficient to consider only 8 variables.

Logistic Regression (Creating Cancer Predicting Model)

Logistic Regression is an analysis method that creates a prediction model based on provided data and variables and predicts outcomes by using the algorithm. Calculations are used to predict the possibility of the data falling into a certain category as a value between 0-1, and the number close to 1 indicates the prediction will more likely to occur. Recognized for its convenience, it is widely used in various fields such as medical care, communication, and data mining.

I am going to apply logistic regression to the breast cancer cells data and create a prediction model, so additionally I downloaded library(aod).

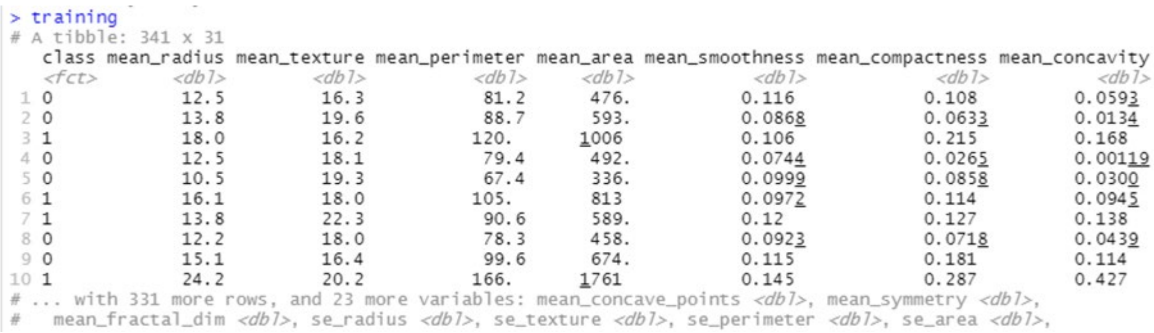
Logistic Regression Preparation

I sorted 569 data sets into 3 different groups to apply Logistic Regression. 60% of them are to train my algorithm, 20% are to examine it, and the rest 20% are for real tests.

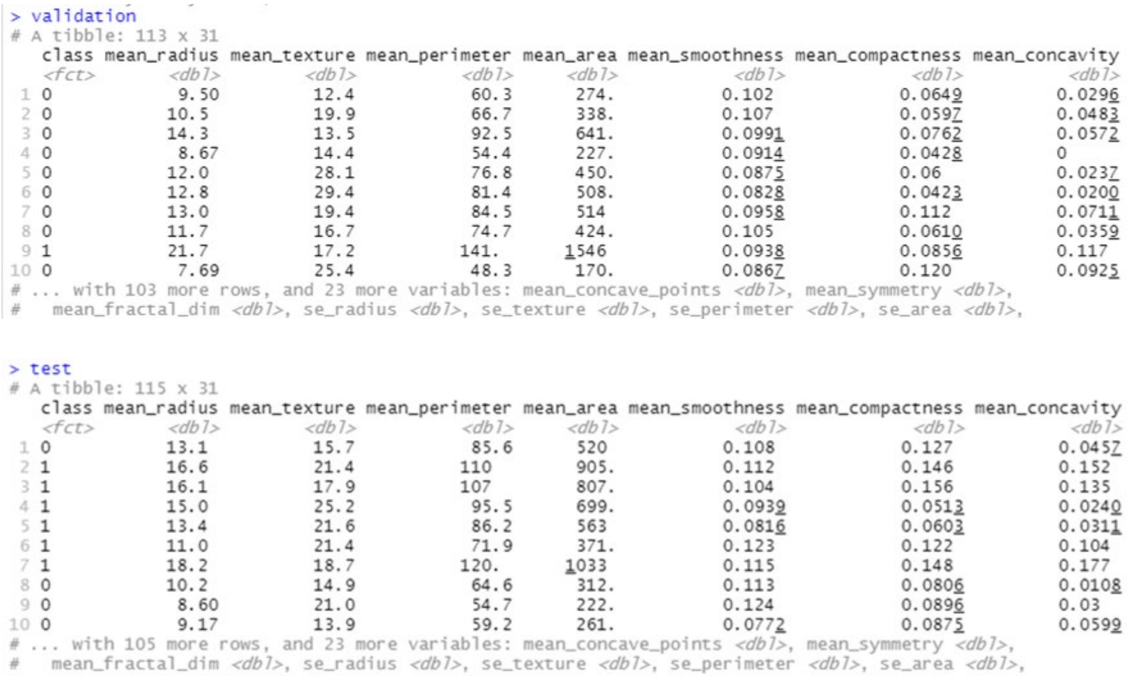
Code:

```
set.seed(1606)
n <- nrow(data)
idx <- 1:n
training_idx <- sample (idx, n*0.60)
idx <- setdiff(idx, training_idx)
validate_idx <- sample(idx, n*0.20)
test_idx <- setdiff(idx, validate_idx)
training <- data[training_idx,]
validation <- data[validate_idx.]
test <- data[test_idx,]
```

Through the code above, the existing 569 data were divided in the ratio of training 60: validation 20: test 20. Data sets have been successfully divided and finished preparation for applying the logistic regression.



```
> training
# A tibble: 341 x 31
   class mean_radius mean_texture mean_perimeter mean_area mean_smoothness mean_compactness mean_concavity
   <fct>       <dbl>        <dbl>          <dbl>     <dbl>           <dbl>            <dbl>          <dbl>
 1 0            12.5         16.3           81.2      476.          0.116            0.108          0.0593
 2 0            13.8         19.6           88.7      593.          0.0868           0.0633         0.0134
 3 1            18.0         16.2          120.      1006           0.106            0.215          0.168
 4 0            12.5         18.1           79.4      492.          0.0744           0.0265         0.00119
 5 0            10.5         19.3           67.4      336.          0.0999           0.0858         0.0300
 6 1            16.1         18.0          105.       813           0.0972           0.114          0.0945
 7 1            13.8         22.3           90.6      589.          0.12             0.127          0.138
 8 0            12.2         18.0           78.3      458.          0.0923           0.0718         0.0439
 9 0            15.1         16.4           99.6      674.          0.115            0.181          0.114
10 1            24.2         20.2          166.      1761           0.145            0.287          0.427
# ... with 331 more rows, and 23 more variables: mean_concave_points <dbl>, mean_symmetry <dbl>,
#   mean_fractal_dim <dbl>, se_radius <dbl>, se_texture <dbl>, se_perimeter <dbl>, se_area <dbl>,
```



```
> validation
# A tibble: 113 x 31
   class mean_radius mean_texture mean_perimeter mean_area mean_smoothness mean_compactness mean_concavity
   <fct>       <dbl>        <dbl>          <dbl>     <dbl>           <dbl>            <dbl>          <dbl>
 1 0            9.50         12.4           60.3      274.          0.102            0.0649         0.0296
 2 0           10.5          19.9           66.7      338.          0.107            0.0597         0.0483
 3 0           14.3          13.5           92.5      641.          0.0991           0.0762         0.0572
 4 0            8.67         14.4           54.4      227.          0.0914           0.0428         0
 5 0           12.0          28.1           76.8      450.          0.0875           0.06           0.0237
 6 0           12.8          29.4           81.4      508.          0.0828           0.0423         0.0200
 7 0           13.0          19.4           84.5      514           0.0958           0.112          0.0711
 8 0           11.7          16.7           74.7      424.          0.105            0.0610         0.0359
 9 1           21.7          17.2          141.      1546           0.0938           0.0856         0.117
10 0            7.69         25.4           48.3      170.          0.0867           0.120          0.0925
# ... with 103 more rows, and 23 more variables: mean_concave_points <dbl>, mean_symmetry <dbl>,
#   mean_fractal_dim <dbl>, se_radius <dbl>, se_texture <dbl>, se_perimeter <dbl>, se_area <dbl>,
```

```
> test
# A tibble: 115 x 31
   class mean_radius mean_texture mean_perimeter mean_area mean_smoothness mean_compactness mean_concavity
   <fct>       <dbl>        <dbl>          <dbl>     <dbl>           <dbl>            <dbl>          <dbl>
 1 0           13.1          15.7           85.6      520           0.108            0.127          0.0457
 2 1           16.6          21.4          110        905.          0.112            0.146          0.152
 3 1           16.1          17.9          107        807.          0.104            0.156          0.135
 4 1           15.0          25.2           95.5      699.          0.0939           0.0513         0.0240
 5 1           13.4          21.6           86.2      563           0.0816           0.0603         0.0311
 6 1           11.0          21.4           71.9      371.          0.123            0.122          0.104
 7 1           18.2          18.7          120.      1033           0.115            0.148          0.177
 8 0           10.2          14.9           64.6      312.          0.113            0.0806         0.0108
 9 0            8.60         21.0           54.7      222.          0.124            0.0896         0.03
10 0            9.17         13.9           59.2      261.          0.0772           0.0875         0.0599
# ... with 105 more rows, and 23 more variables: mean_concave_points <dbl>, mean_symmetry <dbl>,
#   mean_fractal_dim <dbl>, se_radius <dbl>, se_texture <dbl>, se_perimeter <dbl>, se_area <dbl>,
```

Figure 7. It shows the status of classifying each data-training data, validation data, test data.

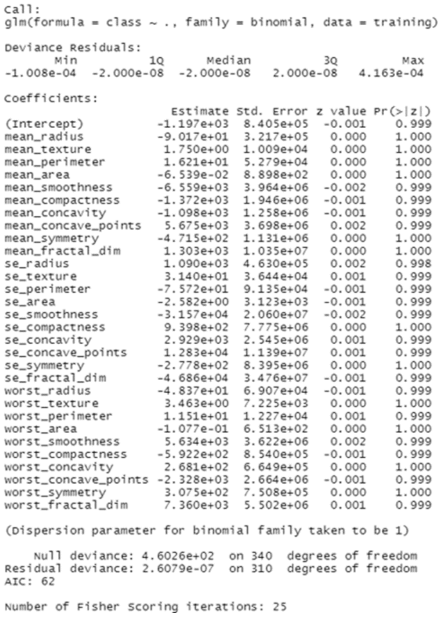
Applying Logistic Regression

After adequately dividing breast cancer cells data sets, the logistic regression model has been derived by using the glm () function.

Code:

> data_lm_full <- gli,(class ~ ., data=training, family=binomial)

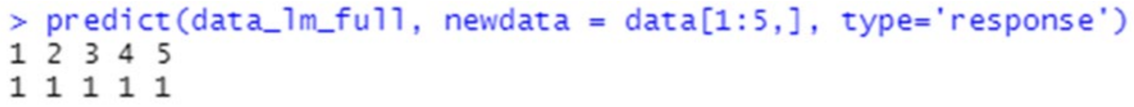
> summary(data_lm_full)



```
Call:
glm(formula = class ~ ., family = binomial, data = training)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-1.008e-04  -2.000e-08  -2.000e-08   2.000e-08   4.163e-04

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.197e+03  8.405e+05  -0.001    0.999
mean_radius          -9.017e+01  3.217e+05   0.000    1.000
mean_texture          1.750e+00  1.009e+04   0.000    1.000
mean_perimeter        1.621e+01  5.279e+04   0.000    1.000
mean_area            -6.539e-02  8.898e+02   0.000    1.000
mean_smoothness      -6.559e+03  3.964e+06  -0.002    0.999
mean_compactness     -1.372e+03  1.946e+06  -0.001    0.999
mean_concavity       -1.098e+03  1.258e+06  -0.001    0.999
mean_concave_points   5.675e+03  3.698e+06   0.002    0.999
mean_symmetry        -4.715e+02  1.131e+06   0.000    1.000
mean_fractal_dim      1.303e+03  1.035e+07   0.000    1.000
se_radius             1.090e+03  4.630e+05   0.002    0.998
se_texture            3.140e+01  3.644e+04   0.001    0.999
se_perimeter         -7.572e+01  9.135e+04  -0.001    0.999
se_area              -2.582e+00  3.123e+03  -0.001    0.999
se_smoothness        -3.157e+04  2.060e+07  -0.002    0.999
se_compactness        9.398e+02  7.775e+06   0.000    1.000
se_concavity          2.929e+03  2.545e+06   0.001    0.999
se_concave_points     1.283e+04  1.139e+07   0.001    0.999
se_symmetry          -2.778e+02  8.395e+06   0.000    1.000
se_fractal_dim       -4.686e+04  3.476e+07  -0.001    0.999
worst_radius         -4.837e+01  6.907e+04  -0.001    0.999
worst_texture         3.463e+00  7.225e+03   0.000    1.000
worst_perimeter       1.151e+01  1.227e+04   0.001    0.999
worst_area           -1.077e-01  6.513e+02   0.000    1.000
worst_smoothness      5.634e+03  3.622e+06   0.002    0.999
worst_compactness    -5.922e+02  8.540e+05  -0.001    0.999
worst_concavity       2.681e+02  6.649e+05   0.000    1.000
worst_concave_points -2.328e+03  2.664e+06  -0.001    0.999
worst_symmetry        3.075e+02  7.508e+05   0.000    1.000
worst_fractal_dim     7.360e+03  5.502e+06   0.001    0.999

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4.6026e+02  on 340  degrees of freedom
Residual deviance: 2.6079e-07  on 310  degrees of freedom
AIC: 62

Number of Fisher Scoring iterations: 25
```

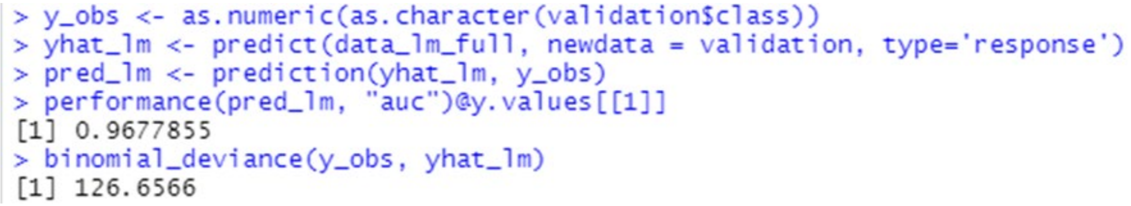
Figure 8. Check the data value using the summary() function.

In a relatively simple way, cancer predicting model applying the Logistic Regression has been created. ‘Number of Fisher Scoring Iterations: 25’ means that the calculation process was repeated 25 times until the prediction model was derived. Now the breast cancer prediction model has been completed, I used to predict () function to obtain the predicted values of 1-5 columns of 569 data.



```
> predict(data_lm_full, newdata = data[1:5,], type='response')
1 2 3 4 5
1 1 1 1 1
```

By using predict (), the predicted possibility of the examined cell being positive has been derived, and they are diagnosed all 100% positive. Compared with data from pic4, it matches 100%, so it proves that the accuracy of the prediction model is very high. However, to confirm the accuracy of this model specifically, the prediction ability was verified through coding.



```
> y_obs <- as.numeric(as.character(validation$class))
> yhat_lm <- predict(data_lm_full, newdata = validation, type='response')
> pred_lm <- prediction(yhat_lm, y_obs)
> performance(pred_lm, "auc")@y.values[[1]]
[1] 0.9677855
> binomial_deviance(y_obs, yhat_lm)
[1] 126.6566
```

As a result of the verification, AUC (Area Under the Curve) was 0.967. The closer the AUC value to 1, it indicates the better the predictive ability of this model. This breast cancer predicting model, based on data provided by the University of Wisconsin, produced 0.967, so it has been proven to be an excellent model with very high accuracy. Although it has sufficient reliability for diagnosing breast cancer at present, there are still existing limitations: using only 569 cell data, not enough variables, and the data records are from 1995. However, these limitations could be overcome because as the number of data increases, more opportunities are provided, so better models could be developed.

Recommended Age for Breast Cancer Diagnosis

Based on data related to breast cancer cells provided by the University of Wisconsin, R-Studio created a model that predicts breast cancer with high accuracy. However, to execute this predictive model, the patient must first undergo a medical examination. So, I analyzed the recommended adequate age for women to have a regular examination. According to the graph above, the number of cases drastically increased from the age of 35-39 (1544), and the number of cases reached 4700 at the age of 45-49. These figures indicate that 18.9% of breast cancer patients are concentrated in the age of mid-40s.

Breast cancer is a disease that could be cured if it is initially detected, with a 98% survival rate. Therefore, women should conduct a breast cancer examination annually from the age of 35, the period when the risk of cancer increases rapidly, and a thorough inspection is needed from 45-49.

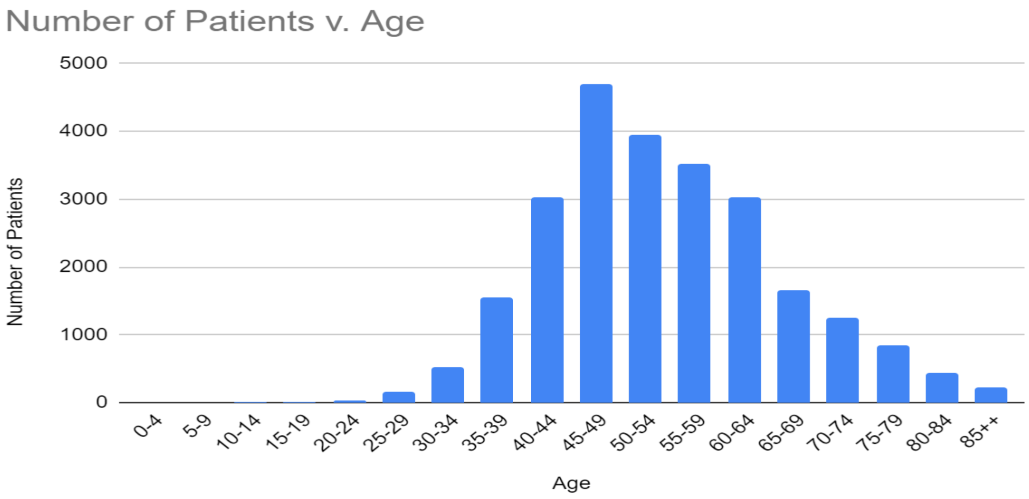


Figure 9. The correlation between Number of Patients and Age.

Conclusion

Analyzation of relationships between features of the cancer cells and creation of a predictive model using logistic regression were implemented based on the R-Studio. Through the pairs () function, texture, smoothness, fractal dimension, and symmetry were found to be independent features of the cancer cells that are not affected by other variables, and the relationship between radius and class (positive, negative) was visualized and proved cell size is directly proportional to the possibility of positive breast cancer cells. Furthermore, the breast cancer predicting model was created with high accuracy of AUC 0.967 that can effectively determine whether the cancer is positive or not. Finally, for early diagnosis of breast cancer, it is recommended to start health screening at the age of 35-39. I will continue to collect larger amounts of data and approach the breast cells with more various variables to create a more accurate breast cancer diagnosis program.

References

American Cancer Society. (2022, March 1). *Survival Rates for Breast Cancer*. American Cancer Society. https://www.cancer.org/cancer/breast-cancer/understanding-a-breast-cancer-diagnosis/breast-cancer-survival-rates.html

UCI Machine Learning Repository. (1995, November 1). *UCI Machine Learning Repository: Breast Cancer Wisconsin (Diagnostic) Data Set* [Dataset]. The University of Wisconsin Clinical Sciences Center and Computer Science Dept. https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)

UCLA Statistical Methods and Data Analytics. (n.d.). *Logit Regression | R Data Analysis Examples*. Retrieved August 22, 2022, from https://stats.oarc.ucla.edu/r/dae/logit-regression/

KOSIS. (1999–2019). *KOSIS-24 Cancer Patients Statistics* [Dataset]. Korean Statistical Office. https://kosis.kr/statHtml/statHtml.do?orgId=117&tblId=DT_117N_A00023&conn_path=I3

M. (2020, August 4). *AUC-ROC 커브*. BioinformaticsAndMe. https://bioinformaticsandme.tistory.com/328

Jonathan Gelford, Martin Goros, Brian Hernandez, and Alex Bokoro. July 2018. A System for an Accountable Data Analysis Process in R. Retrieved August,10,2022 from https://journal.r-project.org/archive/2018/RJ-2018-001/RJ-2018-001.pdf

Kwon, J. (2020). 따라하며 배우는 데이터 과학 (Learning Data Science by Following). 제이펍 주식회사.