Predicting Breast Cancer Survivability

Abstract:

In this paper, I performed an analysis of the prediction of survivability rate of breast cancer patients using R. The survivability of a patient suffers from breast cancer can be predicted by decision trees and Naïve Bayes algorithm using R. The data used is the SEER Publici Use Data with the diagnosed date after 2003. The preprocessed data set consists of 456541 records and 139 variables, which excluded the data before 2003, and only kept 'black' and
'white' in race from original dataset. Sampling multiple datasets from preprocessed data and repeat it multiple times for comparison. I have investigated two data mining techniques: the Naïve Bayes, and the C5.0 decision tree algorithms. When comparing the two results using cross table algorithm, I found out that C5.0 algorithm has a much better performance than Naïve Bayes algorithm.

Introduction:

With the fact that one in eight women in the United States will be diagnosed with breast cancer in her lifetime, breast cancer brings up more and more people's attention. Breast cancer is the most commonly diagnosed cancer in women, yet it is the second leading cause of cancer death among women. According to National Breast Cancer Foundation, "each year it is estimated that over 246,660 women in the United States will be diagnosed with breast cancer and more than 40,000 will die." The survivability of breast cancer was analyzed a lot by previous works, which gives me reference to work on and compare with.

Algorithm and Dataset:

The algorithms come with “C50” and “e1071” package in R. I will also use “gmodels” in R to do a cross table for performance testing.

Due to the limitation of computer performance, in this paper, I only analyzed the data diagnosed after 2003 in black and white population. Also, I randomly sampled the dataset twice with 50000 individuals to perform analysis. The survival status is determined by **SEER Cause-Specific Death Classification;** and the variables were trimmed down to 16 according to previous studies. These 16 variables including 15 factors and one consequence(cause of death) column. The 15 factors are: age at diagnose; marital status; race; sex; tumor extension; grade; behavior code; Regional Nodes Positive; Regional Nodes Examined; RX Summ—Surg Prim Site; radiantion; stage of cancer; and tumor marker status.

The detailed explanations to important variables are listed below:

Code	Description
0	None; diagnosed at autopsy
1	Beam radiation
2	Radioactive implants
3	Radioisotopes
4	Combination of 1 with 2 or 3
5	Radiation, NOS – method or source not specified
6	Other radiation (1973-1987 cases only)
7	Patient or patient's guardian refused radiation therapy
8	Radiation recommended, unknown if administered
9	Unknown if radiation administered

table1-RX Summ-Radiation

Code	Description
1	Single (never married)
2	Married (including common law)
3	Separated
4	Divorced
5	Widowed
6	Unmarried or domestic partner (same sex or opposite sex or unregistered)
9	Unknown

table2-marital status at dx

General Coding Structure

Code	Description
00	None; no surgical procedure of primary site; diagnosed at autopsy only
10-19	Site-specific codes. Tumor destruction; no pathologic specimen or unknown whether there is a pathologic specimen
20-80	Site-specific codes. Resection; pathologic specimen
90	Surgery, NOS. A surgical procedure to the primary site was done, but no information on the type of surgical procedure is provided.
98	Special codes for hematopoietic, reticuloendothelial, immunoproliferative, myeloproliferative diseases; ill-defined sites; and unknown primaries (See site-specific codes for the sites and histologies), except death certificate only
99	Unknown if surgery performed; death certificate only

table3-RX SUMM-surg prim site

Code	Description
1	Grade I; grade i; grade 1; well differentiated; differentiated, NOS
2	Grade II; grade ii; grade 2; moderately differentiated; moderately differentiated; intermediate differentiation
3	Grade III; grade iii; grade 3; poorly differentiated; differentiated
4	Grade IV; grade iv; grade 4; undifferentiated; anaplastic
5	T-cell; T-precursor
6	B-cell; Pre-B; B-Precursor
7	Null cell; Non T-non B;
8	N K cell (natural killer cell)
9	cell type not determined, not stated or not applicable

table4-grade

Code	Description
0	In situ — A noninvasive neoplasm; a tumor which has not penetrated the basement membrane nor extended beyond the epithelial tissue. Some synonyms are intraepithelial (confined to epithelial tissue), noninvasive and noninfiltrating.
1	Localized — An invasive neoplasm confined entirely to the organ of origin. It may include intraluminal extension where specified. For example for colon, intraluminal extension limited to immediately contiguous segments of the large bowel is localized, if no lymph nodes are involved. Localized may exclude invasion of the serosa because of the poor survival of the patient once the serosa is invaded.
2	Regional — A neoplasm that has extended 1) beyond the limits of the organ of origin directly into surrounding organs or tissues; 2) into regional lymph nodes by way of the lymphatic system; or 3) by a combination of extension and regional lymph nodes.
4	Distant — A neoplasm that has spread to parts of the body remote from the primary tumor either by direct extension or by discontinuous metastasis (e.g., implantation or seeding) to distant organs, issues, or via the lymphatic system to distant lymph nodes.
8	Localized/Regional – Only used for Prostate cases.
9	Unstaged — Information is not sufficient to assign a stage.

table5-SEER historic stage

Field Description: Records the total number of regional lymph nodes that were removed and examined by the pathologist.

Code	Description
00	No nodes were examined
01-89	Exact number of nodes examined
90	90 or more nodes were examined
95	No regional nodes were removed, but aspiration of regional nodes was performed
96	Regional lymph node removal was documented as a sampling, and the number of nodes is unknown/not stated
97	Regional lymph node removal was documented as a dissection, and the number of nodes is unknown/not stated
98	Regional lymph nodes were surgically removed, but the number of lymph nodes is unknown/not stated and not documented as a sampling or dissection; nodes were examined, but the number is unknown
99	Unknown whether nodes were examined; not applicable or negative; not stated in patient record

table6-Reginal Nodes Examined

Code	Description
00	All nodes examined are negative
01-89	Exact number of nodes positive
90	90 or more nodes are positive
95	Positive aspiration of lymph node(s) was performed
97	Positive nodes are documented, but number is unspecified
98	No nodes were examined
99	Unknown whether nodes are positive; not applicable; not stated in patient record

table7-Regional nodes positive

Code	Description
1	Positive
2	Negative
3	Borderline
4	Unknown
9	Not 1990+ Breast

table8-ER status

Result:

```
## Total Observations in Table1:  47778
##
##
##                 | predicted result
## actual result  |          1 |          2 |  Row Total |
## ---------------|------------|------------|------------|
##              1 |      41658 |       1041 |      42699 |
##                |      0.872 |      0.022 |            |
## ---------------|------------|------------|------------|
##              2 |       3231 |       1848 |       5079 |
##                |      0.068 |      0.039 |            |
## ---------------|------------|------------|------------|
##   Column Total |      44889 |       2889 |      47778 |
## ---------------|------------|------------|------------|
##
##
```

table1-decision_tree_1

```
## Total Observations in Table2:  47778
##
##
##                 | predicted result
## actual result  |          1 |          2 |  Row Total |
## ---------------|------------|------------|------------|
##              1 |      40489 |       2210 |      42699 |
##                |      0.847 |      0.046 |            |
## ---------------|------------|------------|------------|
##              2 |       2660 |       2419 |       5079 |
##                |      0.056 |      0.051 |            |
## ---------------|------------|------------|------------|
##   Column Total |      43149 |       4629 |      47778 |
## ---------------|------------|------------|------------|
##
##
```

table2- decision_tree_2

```
## Total Observations in Table1:  47778
##
##
##              | actual
##    predicted |         1 |         2 | Row Total |
## -------------|-----------|-----------|-----------|
##            1 |     38684 |      2679 |     41363 |
##              |     0.906 |     0.527 |           |
## -------------|-----------|-----------|-----------|
##            2 |      4015 |      2400 |      6415 |
##              |     0.094 |     0.473 |           |
## -------------|-----------|-----------|-----------|
## Column Total |     42699 |      5079 |     47778 |
##              |     0.894 |     0.106 |           |
## -------------|-----------|-----------|-----------|
```

table3-Naive_Bayes_1

```
## Total Observations in Table2:  47778
##
##
##              | actual
##    predicted |         1 |         2 | Row Total |
## -------------|-----------|-----------|-----------|
##            1 |     38526 |      2994 |     41520 |
##              |     0.904 |     0.580 |           |
## -------------|-----------|-----------|-----------|
##            2 |      4090 |      2168 |      6258 |
##              |     0.096 |     0.420 |           |
## -------------|-----------|-----------|-----------|
## Column Total |     42616 |      5162 |     47778 |
##              |     0.892 |     0.108 |           |
## -------------|-----------|-----------|-----------|
```

table4-Naïve_bayes_2

The results from four cross tables shows that C5.0 algorithm performed better than Naive Bayes. C5.0 has an accuracy rate around 90%, and Naïve Bayes has an accuracy rate around 85%. According to the results from decision tree, the stage of cancer is the most significant factor in survivability. However, the decision tree shows different attribute rank in other variables. Site specific surgery, regional node positive, and tumor extension are also significant according to the results from C5.0 decision tree.

Conclusion:

According to the result we got from C5.0 decision tree and Naïve Bayes, the C5.0 has a better accuracy; and the stage of cancer is the most significant factor in survivability. It is not surprising that age or sex is not the most significant factor in this case.

Discussion:

This project has a large development space. The survival status could be classified by more detailed variables not only cause of death. We could predict survival time with more algorithm too. Also, the decision tree changed when sample changed. Although the accuracy rate doesn't change a lot, however, it is still a chance that the dataset is not large enough to be convincible.

Reference:

[1] Sumbaly, Ronak, N. Vishnusri, and S. Jeyalatha. "Diagnosis of Breast Cancer Using Decision Tree Data Mining Technique." International Journal of Computer Applications 98.10 (2014): 16-24. Web.

[2] Delen, Dursun, Glenn Walker, and Amit Kadam. "Predicting Breast Cancer Survivability: A Comparison of Three Data Mining Methods." Artificial Intelligence in Medicine 34.2 (2005): 113-27. Web.
[3] Khan, Muhammad Umer, Jong Pill Choi, Hyunjung Shin, and Minkoo Kim. "Predicting Breast Cancer Survivability Using Fuzzy Decision Trees for Personalized Healthcare." 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2008): n. pag. Web.
[4] Fan, Qi, Chang-Jie Zhu, and Liu Yin. "Predicting Breast Cancer Recurrence Using Data Mining Techniques." 2010 International Conference on Bioinformatics and Biomedical Technology (2010): n. pag. Web.

Methodology and R Code:

-Data Import

The first step is to got permission for SEER data use, and downloaded the dataset. The dataset contains eight types of cancer from 1973 to 2013, it is formed in ASCII raw data format like this. I use sqlite in r to parse the dataset, and use query to rule out unneccessary type of cancers. The cleaned dataset contains 1218918 rows and 192 columns(which is huge). Doing analyze using a huge dataset is time consuming, so I randomly sampled data into several dataframes, each contains 10000 individual.

```
#transform the data into sqlite table.
df=getFields(seerHome="SEER_1973_2013_TEXTDATA")
df=pickFields(df,picks = c("race","marstat","sex","agedx","eod10ex","eod10pe"
,"eod10nd","eod10pn","eod10ne","codpubkm", "vsrtsadx","sssurg","statrec","COD
","surv"))
head(df,20)
mkSEER(df,seerHome="SEER_1973_2013_TEXTDATA",outDir="mrgd",outFile="cancDef",
```

```
       indices = list(c("sex","race"), c("histo3","seqnum"),  "ICD9"),
       writePops=TRUE,writeRData=TRUE,writeDB=TRUE)

require(dplyr)

## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

require(C50)

## Loading required package: C50

require(gmodels)

## Loading required package: gmodels

setwd("/Users/JQ/Desktop/6030/project")
#connect with sqlite
require("RSQLite")

## Loading required package: RSQLite

# connect to the sqlite file
con = dbConnect(SQLite(), dbname="cancDef.db")
summary(con)

##           Length           Class            Mode
##                1 SQLiteConnection              S4

#randomly sample two dataset
sampleset <- dbGetQuery(con, 'SELECT * FROM canc ORDER BY Random() LIMIT 6000
0')

#triming data according to reference papers.
sampleset <- sampleset %>% dplyr:: select(grep("pubcsnum", names(sampleset)),
grep("yrdx", names(sampleset)),grep("agedx", names(sampleset)),grep("marstat"
, names(sampleset)),
                                          grep("race", names(sampleset)),grep
("sex", names(sampleset)),
                                          grep("grade", names(sampleset)),gre
p("beho", names(sampleset)),grep("eod", names(sampleset)),grep("surgprif", na
mes(sampleset)),
                                          grep("radiatn", names(sampleset)),g
```

```
rep("hststga", names(sampleset)),grep("sssurg", names(sampleset)),grep("vsrts
adx",names(sampleset)),
                                                    grep("erstatus", names(sampleset)),
grep("prstatus", names(sampleset)))

# convert blanks to NA
sampleset <- as.data.frame(sapply(sampleset, function(x) gsub("^$|^ $", NA, x
)))
sapply(sampleset,function(x) sum(is.na(x)))

##     yrdx    agedx  marstat     race      sex  csexten    grade    beho2
##        0        0        0        0        0     5102        0        0
##    beho3  eod10sz  eod10ex  eod10pe  eod10nd  eod10pn  eod10ne    eod13
##        0    54898    54898    60000    54898        0        0    60000
##     eod2     eod4  eodcode surgprif  radiatn  hststga   sssurg vsrtsadx
##    60000    60000    54898        0        0        0    60000        0
## erstatus prstatus
##        0        0

#findout which variable has more than 1000 (1/10) NA, delete.
#remove yrdx,eod10pe;eod10ex;eod10sz;eod10nd;eod13;eod2;eod4;eodcode;sssurg
sampleset <- sampleset[,c(-1,-10,-11,-12,-13,-16,-17,-18,-19,-23)]


#transfer race and sex to number.
sampleset$sex <- as.numeric(sampleset$sex)#female1
sampleset$race <- as.numeric(sampleset$race)#black1

#delete NAs and change data frame to numeric
sampleset <- na.omit(sampleset)
sampleset <- as.data.frame(sapply(sampleset, as.numeric))
```

After Triming section, I start to analyze. The first method I choose is C5.0 Decision Tree. Set up test and train dataset.

```
set.seed(66666)
data_r <- sampleset[order(runif(50000)), ]
summary(sampleset$agedx)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   42.00   52.00   52.75   63.00   91.00

summary(data_r$agedx)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   42.00   52.00   52.77   63.00   91.00

head(data_r$marstat)

## [1] 2 2 2 7 2 5
```

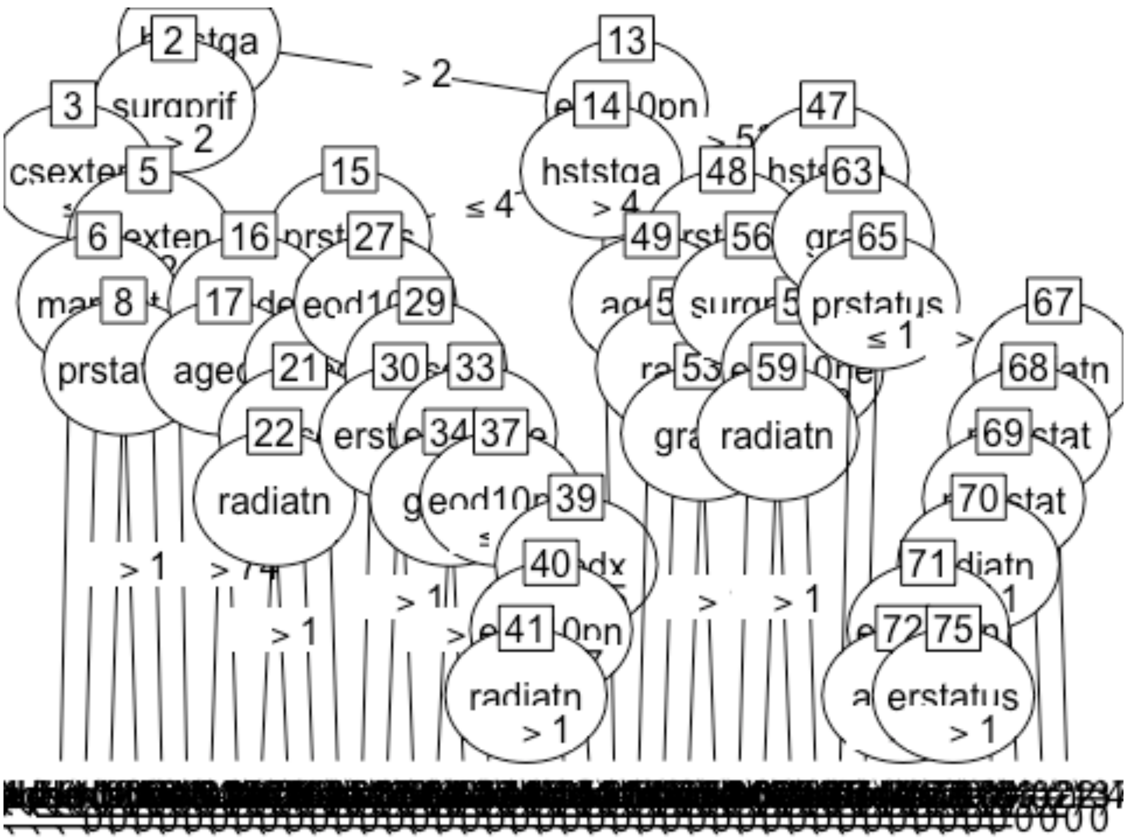
```r
train <- data_r[1:33332,]
test <- data_r[33333:50000,]

#split the data frames and check the proportion of class variable
prop.table(table(train$vsrtsadx))

##
##         1         2
## 0.8919957 0.1080043

prop.table(table(test$vsrtsadx))

##
##         1         2
## 0.8932085 0.1067915

#convert vsrtsadx to factor
train$vsrtsadx<-as.factor(train$vsrtsadx)

model1 <- C5.0(train[,-14], train$vsrtsadx)
model1

##
## Call:
## C5.0.default(x = train[, -14], y = train$vsrtsadx)
##
## Classification Tree
## Number of samples: 33332
## Number of predictors: 15
##
## Tree size: 41
##
## Non-standard options: attempt to group attributes

# display detailed information about the tree
summary(model1)

##
## Call:
## C5.0.default(x = train[, -14], y = train$vsrtsadx)
##
##
## C5.0 [Release 2.07 GPL Edition]      Sat Dec 10 19:31:10 2016
## -------------------------------
##
## Class specified by attribute `outcome'
##
## Read 33332 cases (16 attributes) from undefined.data
##
## Decision tree:
##
```

```
## hststga <= 2:
## :...surgprif > 2: 1 (29059/1705)
## :   surgprif <= 2:
## :   :...csexten <= 9: 1 (930/185)
## :       csexten > 9:
## :       :...csexten > 21: 1 (86/20)
## :           csexten <= 21:
## :           :...marstat <= 3: 1 (85/33)
## :               marstat > 3:
## :               :...prstatus <= 1: 1 (40/16)
## :                   prstatus > 1: 2 (65/19)
## hststga > 2:
## :...eod10pn <= 53:
##     :...hststga > 4: 1 (29/4)
##     :   hststga <= 4:
##     :   :...prstatus <= 1:
##     :       :...grade <= 2:
##     :       :   :...agedx <= 74: 1 (234/45)
##     :       :   :   agedx > 74: 2 (17/7)
##     :       :   grade > 2:
##     :       :   :...agedx > 60: 2 (68/29)
##     :       :       agedx <= 60:
##     :       :       :...race > 1: 1 (163/45)
##     :       :           race <= 1:
##     :       :           :...radiatn <= 1: 2 (25/9)
##     :       :               radiatn > 1: 1 (25/7)
##     :       prstatus > 1:
##     :       :...eod10pn <= 1: 1 (102/33)
##     :           eod10pn > 1:
##     :           :...race <= 1:
##     :               :...erstatus <= 1: 1 (32/13)
##     :               :   erstatus > 1: 2 (77/25)
##     :               race > 1:
##     :               :...eod10ne <= 16:
##     :                   :...grade <= 2: 1 (43/17)
##     :                   :   grade > 2: 2 (153/56)
##     :                   eod10ne > 16:
##     :                   :...eod10pn <= 12: 1 (21/2)
##     :                       eod10pn > 12:
##     :                       :...agedx > 55: 2 (43/16)
##     :                           agedx <= 55:
##     :                           :...eod10pn > 27: 1 (56/12)
##     :                               eod10pn <= 27:
##     :                               :...radiatn <= 1: 2 (12/3)
##     :                                   radiatn > 1: 1 (26/11)
##     eod10pn > 53:
##     :...hststga <= 3:
##         :...prstatus <= 1:
##         :   :...agedx <= 38: 1 (58/19)
##         :   :   agedx > 38:
```

```
##         :   :   :...race <= 1: 2 (86/31)
##         :   :       race > 1:
##         :   :       :...grade <= 1: 1 (39/15)
##         :   :           grade > 1: 2 (394/167)
##         :   prstatus > 1:
##         :   :...surgprif <= 14: 2 (649/154)
##         :       surgprif > 14:
##         :       :...eod10ne > 33: 2 (14/2)
##         :           eod10ne <= 33:
##         :           :...radiatn <= 1: 2 (17/7)
##         :               radiatn > 1: 1 (11/1)
##         hststga > 3:
##         :...grade <= 2: 1 (115/18)
##             grade > 2:
##             :...prstatus <= 1: 1 (57/14)
##                 prstatus > 1:
##                 :...radiatn > 7: 2 (186/52)
##                     radiatn <= 7:
##                     :...marstat > 6: 1 (40/11)
##                         marstat <= 6:
##                         :...marstat > 3: 2 (137/50)
##                             marstat <= 3:
##                             :...radiatn > 1: 1 (22/4)
##                                 radiatn <= 1:
##                                 :...eod10pn <= 55:
##                                     :...agedx <= 70: 1 (37/8)
##                                     :   agedx > 70: 2 (11/3)
##                                     eod10pn > 55:
##                                     :...erstatus <= 1: 1 (6/1)
##                                         erstatus > 1: 2 (62/25)
##
##
## Evaluation on training data (33332 cases):
##
##      Decision Tree
##    ----------------
##    Size      Errors
##
##      41 2894( 8.7%)   <<
##
##
##     (a)   (b)    <-classified as
##    ----  ----
##   29077   655    (a): class 1
##    2239  1361    (b): class 2
##
##
##  Attribute usage:
##
##  100.00% hststga
```

```
##   92.87% surgprif
##    9.20% eod10pn
##    9.08% prstatus
##    5.50% grade
##    3.88% agedx
##    3.62% csexten
##    3.59% race
##    1.85% radiatn
##    1.52% marstat
##    1.19% eod10ne
##    0.53% erstatus
##
##
## Time: 0.2 secs
```

```
plot(model1)
```



re-do the steps above, test another set of data

```
sampleset <- dbGetQuery(con, 'SELECT * FROM canc ORDER BY Random() LIMIT 6000
0')
```

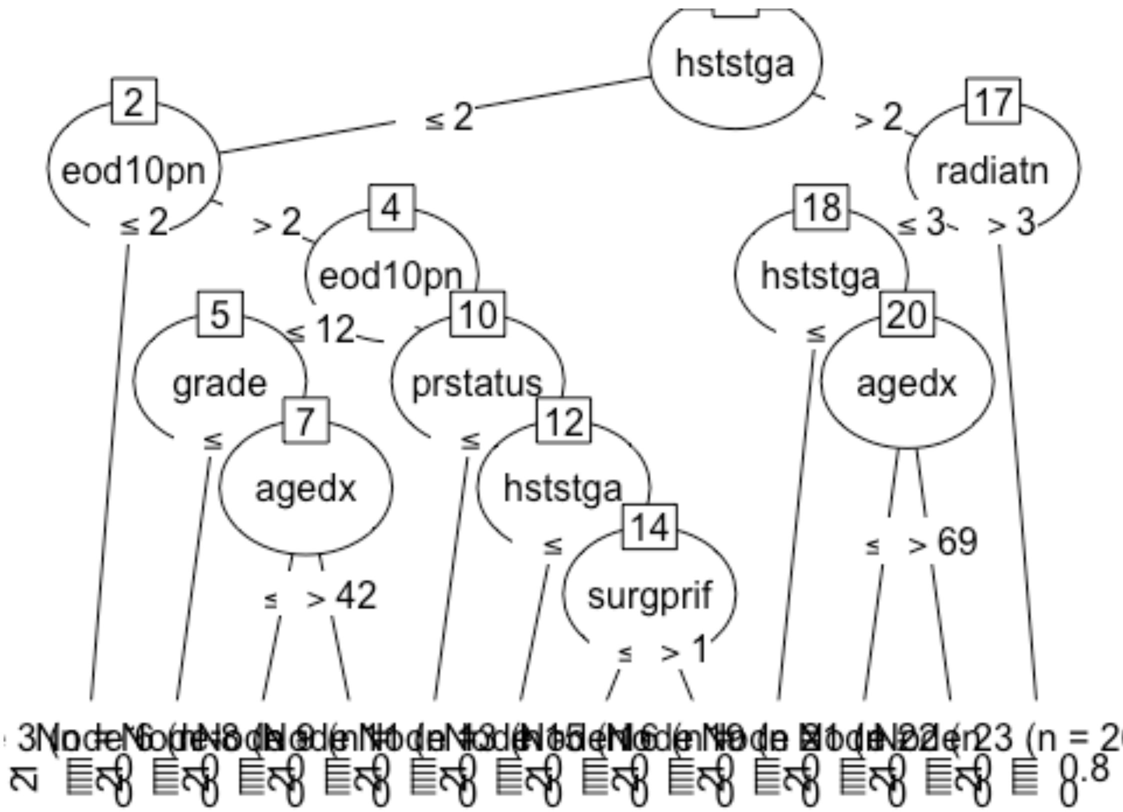
```
#triming data according to reference papers.
sampleset <- sampleset %>% dplyr:: select(grep("pubcsnum", names(sampleset)),
grep("yrdx", names(sampleset)),grep("agedx", names(sampleset)),grep("marstat"
, names(sampleset)),
                                          grep("race", names(sampleset)),grep
("sex", names(sampleset)),
                                          grep("grade", names(sampleset)),gre
p("beho", names(sampleset)),grep("eod", names(sampleset)),grep("surgprif", na
mes(sampleset)),
                                          grep("radiatn", names(sampleset)),g
rep("hststga", names(sampleset)),grep("sssurg", names(sampleset)),grep("vsrts
adx",names(sampleset)),
                                          grep("erstatus", names(sampleset)),
grep("prstatus", names(sampleset)))

# convert blanks to NA
sampleset <- as.data.frame(sapply(sampleset, function(x) gsub("^$|^ $", NA, x
)))
sapply(sampleset,function(x) sum(is.na(x)))

##     yrdx    agedx  marstat     race      sex  csexten    grade    beho2
##        0        0        0        0        0     5176        0        0
##    beho3  eod10sz  eod10ex  eod10pe  eod10nd  eod10pn  eod10ne    eod13
##        0    54824    54824    60000    54824        0        0    60000
##     eod2     eod4  eodcode surgprif  radiatn  hststga   sssurg vsrtsadx
##    60000    60000    54824        0        0        0    60000        0
## erstatus prstatus
##        0        0

#findout which variable has more than 1000 (1/10) NA, delete.
#remove yrdx,eod10pe;eod10ex;eod10sz;eod10nd;eod13;eod2;eod4;eodcode;sssurg
sampleset <- sampleset[,c(-1,-10,-11,-12,-13,-16,-17,-18,-19,-23)]


#transfer race and sex to number.
sampleset$sex <- as.numeric(sampleset$sex)#female1
sampleset$race <- as.numeric(sampleset$race)#black1

#delete NAs and change data frame to numeric
sampleset <- na.omit(sampleset)
sampleset <- as.data.frame(sapply(sampleset, as.numeric))

#randomly select sample
set.seed(66666)
data_r <- sampleset[order(runif(50000)), ]
summary(sampleset$agedx)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   42.00   52.00   52.85   63.00   91.00
```

```
summary(data_r$agedx)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   42.00   52.00   52.88   63.00   91.00

head(data_r$marstat)

## [1] 7 2 7 2 5 4

train <- data_r[1:2222,]
test <- data_r[2223:50000,]

#split the data frames and check the proportion of class variable
prop.table(table(train$vsrtsadx))

##
##         1         2
## 0.8874887 0.1125113

prop.table(table(test$vsrtsadx))

##
##         1         2
## 0.8936958 0.1063042

#convert vsrtsadx to factor
train$vsrtsadx<-as.factor(train$vsrtsadx)

model2 <- C5.0(train[,-14], train$vsrtsadx)
model2

##
## Call:
## C5.0.default(x = train[, -14], y = train$vsrtsadx)
##
## Classification Tree
## Number of samples: 2222
## Number of predictors: 15
##
## Tree size: 12
##
## Non-standard options: attempt to group attributes

# display detailed information about the tree
summary(model2)

##
## Call:
## C5.0.default(x = train[, -14], y = train$vsrtsadx)
##
##
## C5.0 [Release 2.07 GPL Edition]      Sat Dec 10 19:31:37 2016
```

```
## ------------------------------
##
## Class specified by attribute `outcome'
##
## Read 2222 cases (16 attributes) from undefined.data
##
## Decision tree:
##
## hststga > 2:
## :...radiatn > 3: 2 (26/6)
## :   radiatn <= 3:
## :   :...hststga <= 3: 2 (130/56)
## :       hststga > 3:
## :       :...agedx <= 69: 1 (20/3)
## :           agedx > 69: 2 (9/1)
## hststga <= 2:
## :...eod10pn <= 2: 1 (1520/49)
##     eod10pn > 2:
##     :...eod10pn <= 12:
##         :...grade <= 1: 1 (3)
##         :   grade > 1:
##         :   :...agedx <= 42: 2 (8)
##         :       agedx > 42: 1 (14/4)
##         eod10pn > 12:
##         :...prstatus <= 1: 1 (317/36)
##             prstatus > 1:
##             :...hststga <= 1: 1 (47/8)
##                 hststga > 1:
##                 :...surgprif <= 1: 2 (11/2)
##                     surgprif > 1: 1 (117/31)
##
##
## Evaluation on training data (2222 cases):
##
##      Decision Tree
##    ----------------
##    Size      Errors
##
##      12  196( 8.8%)   <<
##
##
##     (a)   (b)    <-classified as
##    ----  ----
##    1907    65    (a): class 1
##     131   119    (b): class 2
##
##
##  Attribute usage:
##
##  100.00% hststga
```

```
##   91.67% eod10pn
##   22.14% prstatus
##    8.33% radiatn
##    5.76% surgprif
##    2.30% agedx
##    1.13% grade
##
##
## Time: 0.0 secs
```

```
plot(model2)
```



compare these two result,

```
prediction1 <- predict.C5.0(model1,test)
CrossTable(test$vsrtsadx, prediction1,
           prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,
           dnn = c('actual result', 'predicted result'))
```

```
##
##
##    Cell Contents
```

```
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table1:  47778
##
##
##               | predicted result
## actual result |         1 |         2 | Row Total |
## --------------|-----------|-----------|-----------|
##             1 |     41658 |      1041 |     42699 |
##               |     0.872 |     0.022 |           |
## --------------|-----------|-----------|-----------|
##             2 |      3231 |      1848 |      5079 |
##               |     0.068 |     0.039 |           |
## --------------|-----------|-----------|-----------|
##  Column Total |     44889 |      2889 |     47778 |
## --------------|-----------|-----------|-----------|
##
##
```

```
prediction2 <- predict.C5.0(model2,test)
CrossTable(test$vsrtsadx, prediction2,
           prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,
           dnn = c('actual result', 'predicted result'))
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table2:  47778
##
##
##               | predicted result
## actual result |         1 |         2 | Row Total |
## --------------|-----------|-----------|-----------|
##             1 |     40489 |      2210 |     42699 |
##               |     0.847 |     0.046 |           |
## --------------|-----------|-----------|-----------|
##             2 |      2660 |      2419 |      5079 |
##               |     0.056 |     0.051 |           |
## --------------|-----------|-----------|-----------|
##  Column Total |     43149 |      4629 |     47778 |
```

```
## --------------|-----------|-----------|-----------|
##
##

# acuracy rate is
```

NaiveBayes

The second method I tried is Naive Bayes.

```
#install e1071 package for bayes analysis
require(e1071)

## Loading required package: e1071

train$vsrtsadx <- as.factor(train$vsrtsadx)
naive1 <- naiveBayes(train[,-14], train$vsrtsadx)
prediction3 <- predict(naive1, test)
CrossTable(prediction3, test$vsrtsadx,prop.chisq = FALSE, prop.t = FALSE, pro
p.r = FALSE, dnn = c('predicted', 'actual'))

##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table1:  47778
##
##
##              | actual
##    predicted |         1 |         2 | Row Total |
## -------------|-----------|-----------|-----------|
##            1 |     38684 |      2679 |     41363 |
##              |     0.906 |     0.527 |           |
## -------------|-----------|-----------|-----------|
##            2 |      4015 |      2400 |      6415 |
##              |     0.094 |     0.473 |           |
## -------------|-----------|-----------|-----------|
## Column Total |     42699 |      5079 |     47778 |
##              |     0.894 |     0.106 |           |
## -------------|-----------|-----------|-----------|
##
##
```

re-do it again

```r
sampleset <- dbGetQuery(con, 'SELECT * FROM canc ORDER BY Random() LIMIT 6000
0')

#triming data according to reference papers.
sampleset <- sampleset %>% dplyr:: select(grep("pubcsnum", names(sampleset)),
grep("yrdx", names(sampleset)),grep("agedx", names(sampleset)),grep("marstat"
, names(sampleset)),
                                          grep("race", names(sampleset)),grep
("sex", names(sampleset)),
                                          grep("grade", names(sampleset)),gre
p("beho", names(sampleset)),grep("eod", names(sampleset)),grep("surgprif", na
mes(sampleset)),
                                          grep("radiatn", names(sampleset)),g
rep("hststga", names(sampleset)),grep("sssurg", names(sampleset)),grep("vsrts
adx",names(sampleset)),
                                          grep("erstatus", names(sampleset)),
grep("prstatus", names(sampleset)))

# convert blanks to NA
sampleset <- as.data.frame(sapply(sampleset, function(x) gsub("^$|^ $", NA, x
)))
sapply(sampleset,function(x) sum(is.na(x)))

##     yrdx    agedx  marstat     race      sex  csexten    grade    beho2
##        0        0        0        0        0     5063        0        0
##    beho3  eod10sz  eod10ex  eod10pe  eod10nd  eod10pn  eod10ne    eod13
##        0    54937    54937    60000    54937        0        0    60000
##     eod2     eod4  eodcode surgprif  radiatn  hststga   sssurg vsrtsadx
##    60000    60000    54937        0        0        0    60000        0
## erstatus prstatus
##        0        0

#findout which variable has more than 1000 (1/10) NA, delete.
#remove yrdx,eod10pe;eod10ex;eod10sz;eod10nd;eod13;eod2;eod4;eodcode;sssurg
sampleset <- sampleset[,c(-1,-10,-11,-12,-13,-16,-17,-18,-19,-23)]


#transfer race and sex to number.
sampleset$sex <- as.numeric(sampleset$sex)#female1
sampleset$race <- as.numeric(sampleset$race)#black1

#delete NAs and change data frame to numeric
sampleset <- na.omit(sampleset)
sampleset <- as.data.frame(sapply(sampleset, as.numeric))

#randomly select sample
set.seed(66666)
data_r <- sampleset[order(runif(50000)), ]
summary(sampleset$agedx)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   44.00   54.00   54.73   65.00   93.00
```

```r
summary(data_r$agedx)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   44.00   54.00   54.74   65.00   93.00
```

```r
head(data_r$marstat)
```

```
## [1] 2 4 2 4 2 7
```

```r
train <- data_r[1:2222,]
test <- data_r[2223:50000,]
train$vsrtsadx <- as.factor(train$vsrtsadx)

naive2 <- naiveBayes(train[,-14], train$vsrtsadx)
prediction4 <- predict(naive2, test)
CrossTable(prediction4, test$vsrtsadx,prop.chisq = FALSE, prop.t = FALSE, pro
p.r = FALSE, dnn = c('predicted', 'actual'))
```

```
## 
## 
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Col Total |
## |-------------------------|
## 
## 
## Total Observations in Table2:  47778 
## 
## 
##              | actual 
##    predicted |         1 |         2 | Row Total | 
## -------------|-----------|-----------|-----------|
##            1 |     38526 |      2994 |     41520 | 
##              |     0.904 |     0.580 |           | 
## -------------|-----------|-----------|-----------|
##            2 |      4090 |      2168 |      6258 | 
##              |     0.096 |     0.420 |           | 
## -------------|-----------|-----------|-----------|
## Column Total |     42616 |      5162 |     47778 | 
##              |     0.892 |     0.108 |           | 
## -------------|-----------|-----------|-----------|
## 
## 
```