Final Project: Breast Cancer Detection

Abhijit Mandal

2020-11-20

## Assignment

**Post the last step of your Final Project. This should be an attached file that contains each step in the final project. Include the following:.**

## Question A:

**Overall, write a coherent narrative that tells a story with the data as you complete this section.**

## Answer for A

As much as data science is playing a pivotal role everywhere, healthcare also finds it prominent application. Breast Cancer is the top rated type of cancer amongst women; which took away 627,000 lives alone. This high mortality rate due to breast cancer does need attention, for early detection so that prevention can be done in time. As a potential contributor to state-of-art technology development, data mining and machine learning finds a multi-fold application in predicting Brest cancer. The objective of this project is to classify each of the tumor to be malignant or benign.

I used the dataset from Kaggle <https://www.kaggle.com/uciml/breast-cancer-wisconsin-data> for my research.

## Question B.

**Summarize the problem statement you addressed.**

## Answer for B:

I focused on the below problem statement:

* How do we define a tumor as malignant or benign ?
* Can any benign tumor turn to malignant at later time ?
* What are the characteristics of a malignant and benign tumor (size, mass, texture, smoothness etc)?
* Does the chances of a breast cancer varies from individual to individual?

## Code

## Set the working directory to the root of your DSC 520 directory  
  
setwd("~/Documents/GitHub/dsc520")  
  
## Loading Library  
library(readr)  
library(class)  
library(gmodels)  
library(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

#load data  
breastcancer\_DF <- read.csv("data/BreastCancerData.csv")  
  
str(breastcancer\_DF)

## 'data.frame': 569 obs. of 32 variables:  
## $ id : int 842302 842517 84300903 84348301 84358402 843786 844359 84458202 844981 84501001 ...  
## $ diagnosis : chr "M" "M" "M" "M" ...  
## $ radius\_mean : num 18 20.6 19.7 11.4 20.3 ...  
## $ texture\_mean : num 10.4 17.8 21.2 20.4 14.3 ...  
## $ perimeter\_mean : num 122.8 132.9 130 77.6 135.1 ...  
## $ area\_mean : num 1001 1326 1203 386 1297 ...  
## $ smoothness\_mean : num 0.1184 0.0847 0.1096 0.1425 0.1003 ...  
## $ compactness\_mean : num 0.2776 0.0786 0.1599 0.2839 0.1328 ...  
## $ concavity\_mean : num 0.3001 0.0869 0.1974 0.2414 0.198 ...  
## $ concave.points\_mean : num 0.1471 0.0702 0.1279 0.1052 0.1043 ...  
## $ symmetry\_mean : num 0.242 0.181 0.207 0.26 0.181 ...  
## $ fractal\_dimension\_mean : num 0.0787 0.0567 0.06 0.0974 0.0588 ...  
## $ radius\_se : num 1.095 0.543 0.746 0.496 0.757 ...  
## $ texture\_se : num 0.905 0.734 0.787 1.156 0.781 ...  
## $ perimeter\_se : num 8.59 3.4 4.58 3.44 5.44 ...  
## $ area\_se : num 153.4 74.1 94 27.2 94.4 ...  
## $ smoothness\_se : num 0.0064 0.00522 0.00615 0.00911 0.01149 ...  
## $ compactness\_se : num 0.049 0.0131 0.0401 0.0746 0.0246 ...  
## $ concavity\_se : num 0.0537 0.0186 0.0383 0.0566 0.0569 ...  
## $ concave.points\_se : num 0.0159 0.0134 0.0206 0.0187 0.0188 ...  
## $ symmetry\_se : num 0.03 0.0139 0.0225 0.0596 0.0176 ...  
## $ fractal\_dimension\_se : num 0.00619 0.00353 0.00457 0.00921 0.00511 ...  
## $ radius\_worst : num 25.4 25 23.6 14.9 22.5 ...  
## $ texture\_worst : num 17.3 23.4 25.5 26.5 16.7 ...  
## $ perimeter\_worst : num 184.6 158.8 152.5 98.9 152.2 ...  
## $ area\_worst : num 2019 1956 1709 568 1575 ...  
## $ smoothness\_worst : num 0.162 0.124 0.144 0.21 0.137 ...  
## $ compactness\_worst : num 0.666 0.187 0.424 0.866 0.205 ...  
## $ concavity\_worst : num 0.712 0.242 0.45 0.687 0.4 ...  
## $ concave.points\_worst : num 0.265 0.186 0.243 0.258 0.163 ...  
## $ symmetry\_worst : num 0.46 0.275 0.361 0.664 0.236 ...  
## $ fractal\_dimension\_worst: num 0.1189 0.089 0.0876 0.173 0.0768 ...

The dataset has 569 observation with 33 variables. Out of 33 variables or features of this dataset, One is identification Number, another is a cancer diagnosis, 30 are numerically valued laboratory measurements and the last variable is X which has all NA value. The diagnosis is coded as “M” to indicate malignant and “B” to indicate benign. By looking at the output of str command I can see that the 30 measurement numeric features include the mean, standard error and worst value for the 10 different characteristics of the cell. Radius Texture Perimeter Area Smoothness Compactness Concavity Concave points Symmetry Fractal dimension

## Question C:

**Summarize how you addressed this problem statement (the data used and the methodology employed).**

## Answer For C

In this project, I first analyzed the data and looked for any cleanups needed, then I derived correlation between the variables, after visualizing and analyzing the data I used machine learning algorithm KNN to derive at a conclusion.I considered variables such as tumor size, mass, texture, smoothness, thickness etc that can help in predicting the chances of a tumor being malignant or benign, I used K-nearest neighbor algorithm to classify the tumor, the result of this algorithm provided an accurate response.

## Code

#The first variable is id which doesn't provide any useful information, will exclude these from the model.  
  
breastcancer\_DF <- select(breastcancer\_DF,-id)  
  
#The diagnosis variable is the outcome I want to predict. This feature indicates whether the cell is from the malignant or benign group.  
  
table(breastcancer\_DF$diagnosis)

##   
## B M   
## 357 212

round(prop.table(table(breastcancer\_DF$diagnosis)) \* 100, digits = 1)

##   
## B M   
## 62.7 37.3

#The table() shows that this dataset has 357 benign cells and 212 malignant cells. The prop.table() shows that 62.7 percent is Benign and 37.3 percent of the mass is malignant.  
  
## Missing values  
  
sum(is.na(breastcancer\_DF))

## [1] 0

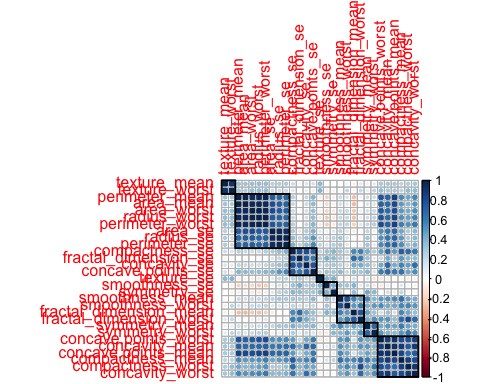
# There are no missing values in the dataset so we can consider it for modeling  
  
head(breastcancer\_DF)

## diagnosis radius\_mean texture\_mean perimeter\_mean area\_mean smoothness\_mean  
## 1 M 17.99 10.38 122.80 1001.0 0.11840  
## 2 M 20.57 17.77 132.90 1326.0 0.08474  
## 3 M 19.69 21.25 130.00 1203.0 0.10960  
## 4 M 11.42 20.38 77.58 386.1 0.14250  
## 5 M 20.29 14.34 135.10 1297.0 0.10030  
## 6 M 12.45 15.70 82.57 477.1 0.12780  
## compactness\_mean concavity\_mean concave.points\_mean symmetry\_mean  
## 1 0.27760 0.3001 0.14710 0.2419  
## 2 0.07864 0.0869 0.07017 0.1812  
## 3 0.15990 0.1974 0.12790 0.2069  
## 4 0.28390 0.2414 0.10520 0.2597  
## 5 0.13280 0.1980 0.10430 0.1809  
## 6 0.17000 0.1578 0.08089 0.2087  
## fractal\_dimension\_mean radius\_se texture\_se perimeter\_se area\_se  
## 1 0.07871 1.0950 0.9053 8.589 153.40  
## 2 0.05667 0.5435 0.7339 3.398 74.08  
## 3 0.05999 0.7456 0.7869 4.585 94.03  
## 4 0.09744 0.4956 1.1560 3.445 27.23  
## 5 0.05883 0.7572 0.7813 5.438 94.44  
## 6 0.07613 0.3345 0.8902 2.217 27.19  
## smoothness\_se compactness\_se concavity\_se concave.points\_se symmetry\_se  
## 1 0.006399 0.04904 0.05373 0.01587 0.03003  
## 2 0.005225 0.01308 0.01860 0.01340 0.01389  
## 3 0.006150 0.04006 0.03832 0.02058 0.02250  
## 4 0.009110 0.07458 0.05661 0.01867 0.05963  
## 5 0.011490 0.02461 0.05688 0.01885 0.01756  
## 6 0.007510 0.03345 0.03672 0.01137 0.02165  
## fractal\_dimension\_se radius\_worst texture\_worst perimeter\_worst area\_worst  
## 1 0.006193 25.38 17.33 184.60 2019.0  
## 2 0.003532 24.99 23.41 158.80 1956.0  
## 3 0.004571 23.57 25.53 152.50 1709.0  
## 4 0.009208 14.91 26.50 98.87 567.7  
## 5 0.005115 22.54 16.67 152.20 1575.0  
## 6 0.005082 15.47 23.75 103.40 741.6  
## smoothness\_worst compactness\_worst concavity\_worst concave.points\_worst  
## 1 0.1622 0.6656 0.7119 0.2654  
## 2 0.1238 0.1866 0.2416 0.1860  
## 3 0.1444 0.4245 0.4504 0.2430  
## 4 0.2098 0.8663 0.6869 0.2575  
## 5 0.1374 0.2050 0.4000 0.1625  
## 6 0.1791 0.5249 0.5355 0.1741  
## symmetry\_worst fractal\_dimension\_worst  
## 1 0.4601 0.11890  
## 2 0.2750 0.08902  
## 3 0.3613 0.08758  
## 4 0.6638 0.17300  
## 5 0.2364 0.07678  
## 6 0.3985 0.12440

## Corelation  
library(corrplot)

## corrplot 0.84 loaded

corr\_mat <- cor(breastcancer\_DF[,3:ncol(breastcancer\_DF)])  
corrplot(corr\_mat, order = "hclust", tl.cex = 1, addrect = 8)



#cor(select(breastcancer\_DF,-diagnosis))  
  
summary(breastcancer\_DF)

## diagnosis radius\_mean texture\_mean perimeter\_mean   
## Length:569 Min. : 6.981 Min. : 9.71 Min. : 43.79   
## Class :character 1st Qu.:11.700 1st Qu.:16.17 1st Qu.: 75.17   
## Mode :character Median :13.370 Median :18.84 Median : 86.24   
## Mean :14.127 Mean :19.29 Mean : 91.97   
## 3rd Qu.:15.780 3rd Qu.:21.80 3rd Qu.:104.10   
## Max. :28.110 Max. :39.28 Max. :188.50   
## area\_mean smoothness\_mean compactness\_mean concavity\_mean   
## Min. : 143.5 Min. :0.05263 Min. :0.01938 Min. :0.00000   
## 1st Qu.: 420.3 1st Qu.:0.08637 1st Qu.:0.06492 1st Qu.:0.02956   
## Median : 551.1 Median :0.09587 Median :0.09263 Median :0.06154   
## Mean : 654.9 Mean :0.09636 Mean :0.10434 Mean :0.08880   
## 3rd Qu.: 782.7 3rd Qu.:0.10530 3rd Qu.:0.13040 3rd Qu.:0.13070   
## Max. :2501.0 Max. :0.16340 Max. :0.34540 Max. :0.42680   
## concave.points\_mean symmetry\_mean fractal\_dimension\_mean radius\_se   
## Min. :0.00000 Min. :0.1060 Min. :0.04996 Min. :0.1115   
## 1st Qu.:0.02031 1st Qu.:0.1619 1st Qu.:0.05770 1st Qu.:0.2324   
## Median :0.03350 Median :0.1792 Median :0.06154 Median :0.3242   
## Mean :0.04892 Mean :0.1812 Mean :0.06280 Mean :0.4052   
## 3rd Qu.:0.07400 3rd Qu.:0.1957 3rd Qu.:0.06612 3rd Qu.:0.4789   
## Max. :0.20120 Max. :0.3040 Max. :0.09744 Max. :2.8730   
## texture\_se perimeter\_se area\_se smoothness\_se   
## Min. :0.3602 Min. : 0.757 Min. : 6.802 Min. :0.001713   
## 1st Qu.:0.8339 1st Qu.: 1.606 1st Qu.: 17.850 1st Qu.:0.005169   
## Median :1.1080 Median : 2.287 Median : 24.530 Median :0.006380   
## Mean :1.2169 Mean : 2.866 Mean : 40.337 Mean :0.007041   
## 3rd Qu.:1.4740 3rd Qu.: 3.357 3rd Qu.: 45.190 3rd Qu.:0.008146   
## Max. :4.8850 Max. :21.980 Max. :542.200 Max. :0.031130   
## compactness\_se concavity\_se concave.points\_se symmetry\_se   
## Min. :0.002252 Min. :0.00000 Min. :0.000000 Min. :0.007882   
## 1st Qu.:0.013080 1st Qu.:0.01509 1st Qu.:0.007638 1st Qu.:0.015160   
## Median :0.020450 Median :0.02589 Median :0.010930 Median :0.018730   
## Mean :0.025478 Mean :0.03189 Mean :0.011796 Mean :0.020542   
## 3rd Qu.:0.032450 3rd Qu.:0.04205 3rd Qu.:0.014710 3rd Qu.:0.023480   
## Max. :0.135400 Max. :0.39600 Max. :0.052790 Max. :0.078950   
## fractal\_dimension\_se radius\_worst texture\_worst perimeter\_worst   
## Min. :0.0008948 Min. : 7.93 Min. :12.02 Min. : 50.41   
## 1st Qu.:0.0022480 1st Qu.:13.01 1st Qu.:21.08 1st Qu.: 84.11   
## Median :0.0031870 Median :14.97 Median :25.41 Median : 97.66   
## Mean :0.0037949 Mean :16.27 Mean :25.68 Mean :107.26   
## 3rd Qu.:0.0045580 3rd Qu.:18.79 3rd Qu.:29.72 3rd Qu.:125.40   
## Max. :0.0298400 Max. :36.04 Max. :49.54 Max. :251.20   
## area\_worst smoothness\_worst compactness\_worst concavity\_worst   
## Min. : 185.2 Min. :0.07117 Min. :0.02729 Min. :0.0000   
## 1st Qu.: 515.3 1st Qu.:0.11660 1st Qu.:0.14720 1st Qu.:0.1145   
## Median : 686.5 Median :0.13130 Median :0.21190 Median :0.2267   
## Mean : 880.6 Mean :0.13237 Mean :0.25427 Mean :0.2722   
## 3rd Qu.:1084.0 3rd Qu.:0.14600 3rd Qu.:0.33910 3rd Qu.:0.3829   
## Max. :4254.0 Max. :0.22260 Max. :1.05800 Max. :1.2520   
## concave.points\_worst symmetry\_worst fractal\_dimension\_worst  
## Min. :0.00000 Min. :0.1565 Min. :0.05504   
## 1st Qu.:0.06493 1st Qu.:0.2504 1st Qu.:0.07146   
## Median :0.09993 Median :0.2822 Median :0.08004   
## Mean :0.11461 Mean :0.2901 Mean :0.08395   
## 3rd Qu.:0.16140 3rd Qu.:0.3179 3rd Qu.:0.09208   
## Max. :0.29100 Max. :0.6638 Max. :0.20750

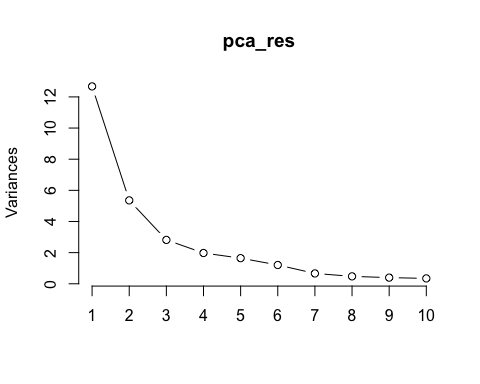
#I will be using KNN algorithm, after looking at the output of the summary() , the range for radius\_mean is 6.981 to 28.110 and the range of smoothness\_mean is 0.05263 to 0.16340, the impact of the radius will be larger than the smoothness in the distance calculation which can cause for modeling.  
  
normalize <- function(x) {  
 return ((x - min(x)) / (max(x) - min(x)))  
}  
  
  
updated\_breastcancer\_DF <- as.data.frame(lapply(select(breastcancer\_DF,-diagnosis), normalize))  
  
summary(select(updated\_breastcancer\_DF,radius\_mean,smoothness\_mean))

## radius\_mean smoothness\_mean   
## Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.2233 1st Qu.:0.3046   
## Median :0.3024 Median :0.3904   
## Mean :0.3382 Mean :0.3948   
## 3rd Qu.:0.4164 3rd Qu.:0.4755   
## Max. :1.0000 Max. :1.0000

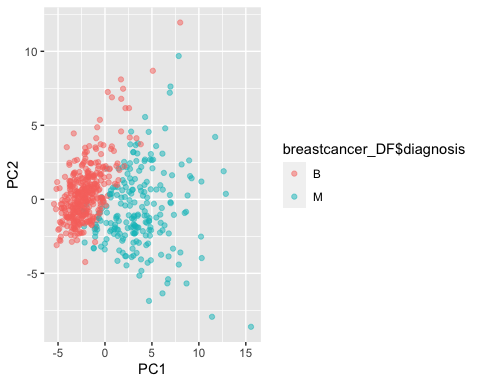
#Creating a training set for building the KNN model and testing set for checking the accuracy of the model.   
#I will use the 80% for training and 20% to simulate the new patients.  
  
binaryClassifier\_split <- sample(1:nrow(updated\_breastcancer\_DF), 0.8 \* nrow(updated\_breastcancer\_DF))  
trainds <- updated\_breastcancer\_DF[binaryClassifier\_split,]  
testds <- updated\_breastcancer\_DF[-binaryClassifier\_split,]  
  
trained\_dataset <- breastcancer\_DF[binaryClassifier\_split,1]  
test\_dataset <- breastcancer\_DF[-binaryClassifier\_split,1]  
  
#Data preprocessing  
  
#Because there is so much correlation, some machine learning models can fail. We are going to create a PCA version of the data  
pca\_res <- prcomp(breastcancer\_DF[,3:ncol(breastcancer\_DF)], center = TRUE, scale = TRUE)  
plot(pca\_res, type="l")  
  
summary(pca\_res)

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6 PC7  
## Standard deviation 3.5602 2.3145 1.67860 1.40601 1.28301 1.09859 0.81534  
## Proportion of Variance 0.4371 0.1847 0.09716 0.06817 0.05676 0.04162 0.02292  
## Cumulative Proportion 0.4371 0.6218 0.71895 0.78712 0.84388 0.88550 0.90842  
## PC8 PC9 PC10 PC11 PC12 PC13 PC14  
## Standard deviation 0.69036 0.62876 0.58783 0.54148 0.51013 0.49123 0.39543  
## Proportion of Variance 0.01643 0.01363 0.01192 0.01011 0.00897 0.00832 0.00539  
## Cumulative Proportion 0.92485 0.93849 0.95040 0.96051 0.96948 0.97781 0.98320  
## PC15 PC16 PC17 PC18 PC19 PC20 PC21  
## Standard deviation 0.30645 0.2796 0.23982 0.22774 0.21104 0.17623 0.17248  
## Proportion of Variance 0.00324 0.0027 0.00198 0.00179 0.00154 0.00107 0.00103  
## Cumulative Proportion 0.98644 0.9891 0.99111 0.99290 0.99444 0.99551 0.99654  
## PC22 PC23 PC24 PC25 PC26 PC27 PC28  
## Standard deviation 0.16495 0.15477 0.13050 0.12436 0.08933 0.08164 0.03850  
## Proportion of Variance 0.00094 0.00083 0.00059 0.00053 0.00028 0.00023 0.00005  
## Cumulative Proportion 0.99747 0.99830 0.99889 0.99942 0.99970 0.99992 0.99998  
## PC29  
## Standard deviation 0.02635  
## Proportion of Variance 0.00002  
## Cumulative Proportion 1.00000

#The two first components explains the 0.6324 of the variance. We need 10 principal components to explain more than 0.95 of the variance and 17 to explain more than 0.99  
  
library(ggplot2)



pca\_df <- as.data.frame(pca\_res$x)  
ggplot(pca\_df, aes(x=PC1, y=PC2, col=breastcancer\_DF$diagnosis)) + geom\_point(alpha=0.5)

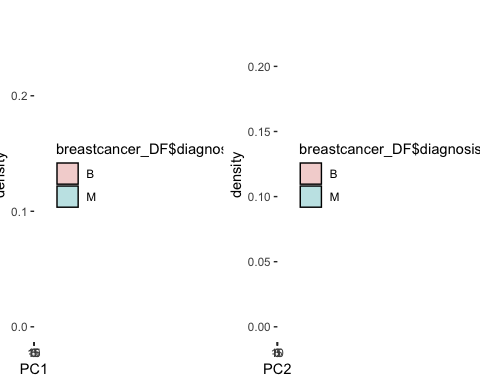


#The data can be easly separated.  
library(gridExtra)

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

g\_pc1 <- ggplot(pca\_df, aes(x=PC1, fill=breastcancer\_DF$diagnosis)) + geom\_density(alpha=0.25)   
g\_pc2 <- ggplot(pca\_df, aes(x=PC2, fill=breastcancer\_DF$diagnosis)) + geom\_density(alpha=0.25)   
grid.arrange(g\_pc1, g\_pc2, ncol=2)



# Applying k nearest neighbour algorithm, using K as 23  
knnTestprediction <- knn(trainds,testds,cl=trained\_dataset,k=23)  
  
#Model Performance   
#After the modeling of the data in knn algorithm, checking the performance of the model using confusion matrix  
confusionMatrix <- table(test\_dataset,knnTestprediction)  
confusionMatrix

## knnTestprediction  
## test\_dataset B M  
## B 71 0  
## M 4 39

modelaccuracy <- (confusionMatrix[[1,1]] + confusionMatrix[[2,2]]) / sum(confusionMatrix)  
modelaccuracy

## [1] 0.9649123

#The classification the model is divided into four categories  
# Top Left - True negative : predicted value was benign and identified as benign  
# Bottom Right - True positive : predicted value was malignant and identified as malignant  
# Top Right - False Positive: predicted value was malignant but cancer was actually benign  
# Bottom Left - False negative: predicted value was benign but the cancer was actually malignant  
  
#False negative should be as much less as possible for our model as it is misleading to the patient and the disease may continue to spread.   
#False positive is less dangerous than the false negative but it can add an extra financial burden on the patient/ health system and additional stress on the patient.

## Question D:

**Summarize the interesting insights that your analysis provided.** As out of interest I looked at using various other models for comparison with KNN, the areas where KNN is certified as best model were Specificity, Positive Prediction Value and Precision.

## Answer For D

## Question E:

**Summarize the implications to the consumer (target audience) of your analysis**

## Answer For E

The intent of this project is to assist doctors in diagnosing breast cancer for patients, allowing physicians to spend more time on treating the disease.Using machine learning methods for diagnostic can significantly increase processing speed and on a big scale can make the diagnostic significantly cheaper.

## Question F:

**Discuss the limitations of your analysis and how you, or someone else, could improve or build on it.**

## Answer For F

We have features of a tumor but I was not sure what does they mean or actually how much do we need to know about these features I believe that we do not need to know meaning of these features however in order to imagine in our mind we should know something like variance, standard deviation, number of sample (count) or max min values. These type of information helps to understand about what is going on data. For example , the question is appeared in my mind the area\_mean feature’s max value is 2500 and smoothness\_mean features’ max 0.16340. Also, it would have been great if i could compare the result of my data model vs other machine learning algorithms like Random Forest, SVM etc. In future we can look into the implementation of artificial neural net and deep learning for predictive model development with a larger and un- structured data set. This will use unsupervised learning algorithms such SVM etc. to first label the data and distributing them over training set, cross-validation set and test set.