Thomas Jones CS-5565-0007 : Classification Lab

Git Repo Link: <https://github.com/wortcook/UMKC/tree/main/Fall2024/CS5565-0007/classlab>

Recording Link: <https://umsystem.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=2f7aac1e-1b77-4432-bc36-b20301039d59>

**Discussion from Question 3:**

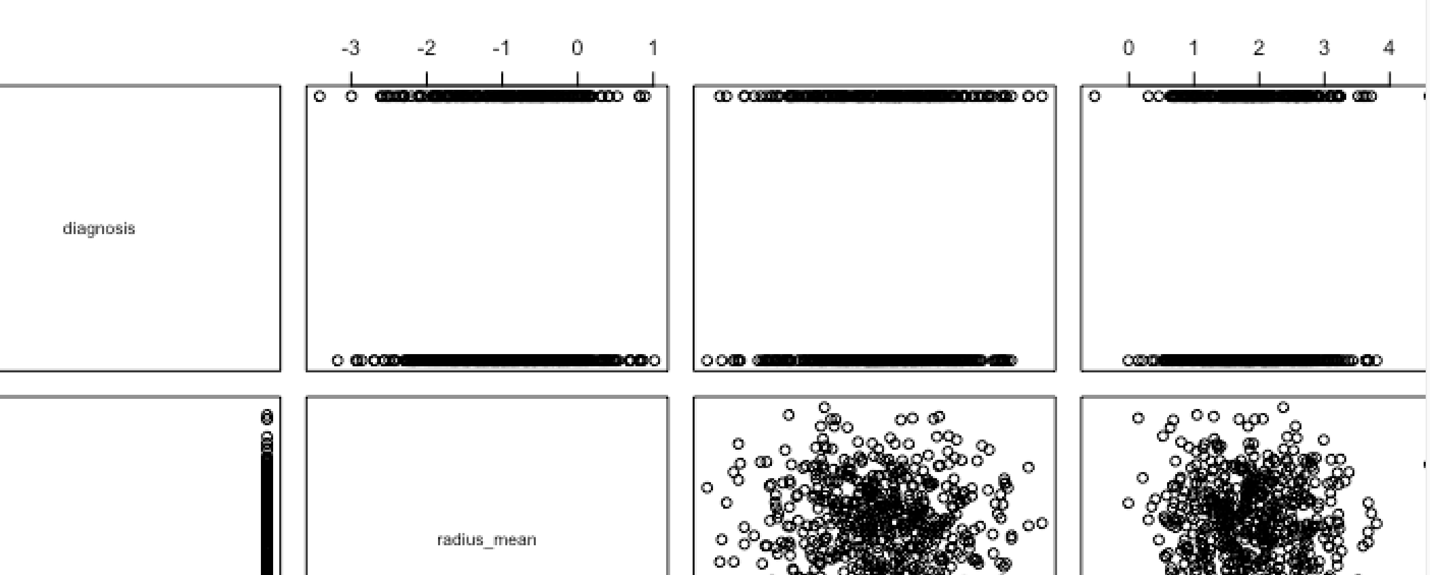
1. The Naïve Bayes model performed slightly better than either the logistic or LDA models but not in any way that is truly more informative.
2. For this dataset, there does not appear to be any better interpretive value from Naïve Bayes.
3. The dataset priors favored B and given that the data points are mostly overlapped and of limited correlation the model results err to simply call most observations the most prevalent case.

The first step for the lab was to get an overview of the data. A grid plot was produced, see below:

A grid of black dots

Description automatically generated with medium confidence

While not obvious from the overall grid plot, zooming in to a few of the plots in the upper left of the image shows that there is a great deal of overlap between the B and M diagnosis and in many cases, complete overlap.



Based on the correlation data as well as visually inspecting the single variable logistic regression plots (see next page), 7 individual data points were selected that showed at least some slope in the plots. These were:

* Perimeter mean
* Compactness mean
* Concavity worst
* Texture se
* Texture worst
* Area worst
* Symmetry worst

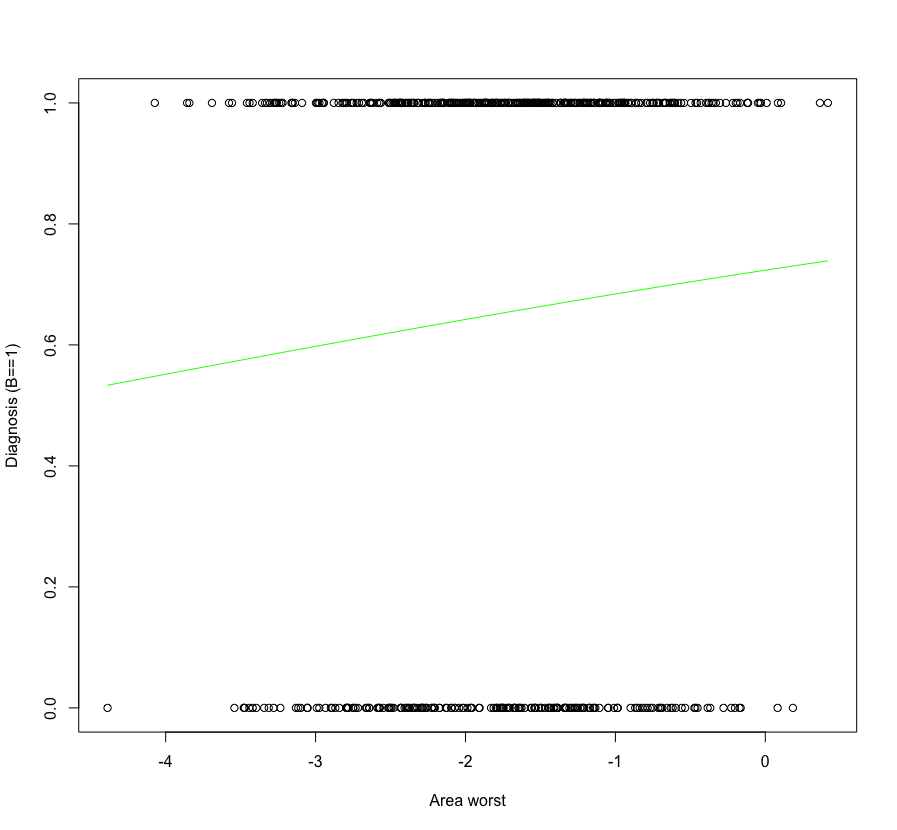
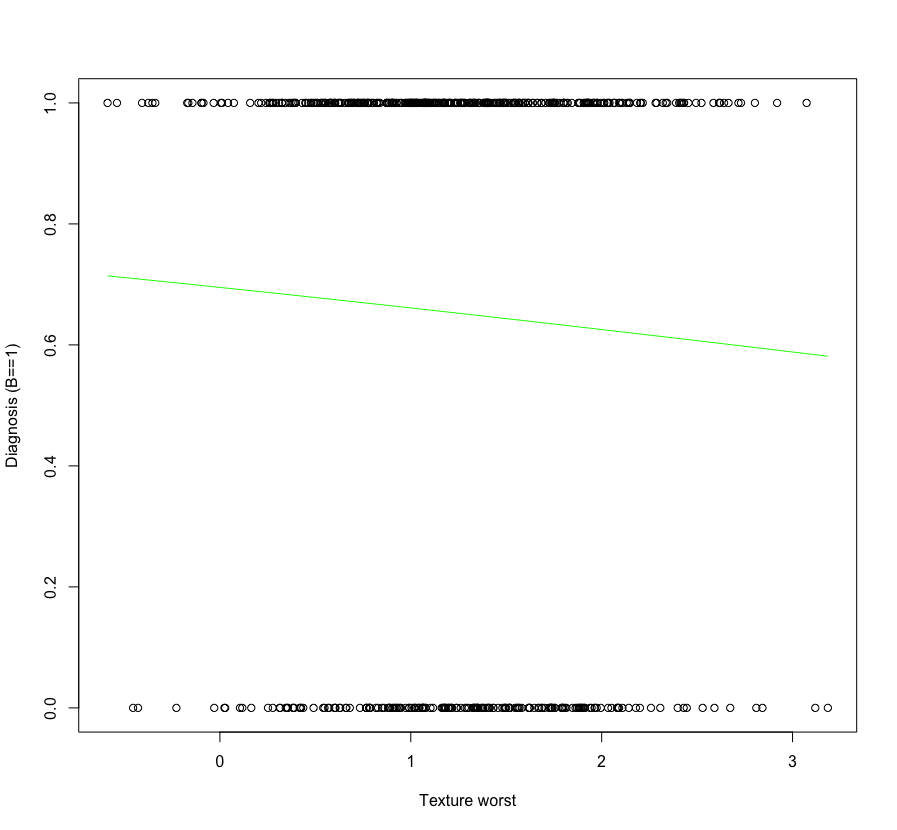
From these features, the data was then split into train/test datasets randomly with a fixed initial seed and a 80/20 train to test data split. Each algorithm (log regression, LDA, and Naïve Bayes) were then tested against the test/train data and a confusion matrix produced.

A green line on a white surface

Description automatically generatedA green line on a white surface

Description automatically generatedA green line on a white surface

Description automatically generatedA graph with a green line

Description automatically generatedA green line on a white surface

Description automatically generated

LDA Plot:

A group of blue bars

Description automatically generated with medium confidence

The confusion matrix for each algorithm is shown below. Note, for data purposes we are considering B (benign) as Positive and M as Negative.

|  |  |  |
| --- | --- | --- |
| Logistic | Test Prediction |  |
| Test Actual | **B** | **M** |
| B | 68 | 3 |
| M | 51 | 1 |

**Accuracy:** (68+1)/123 = 56.1%

**Precision:** 68/123 = 55.3%

**Specificity:** 1/123 = 0.8%

|  |  |  |
| --- | --- | --- |
| LDA | Test Prediction |  |
| Test Actual | **B** | **M** |
| B | 68 | 3 |
| M | 51 | 1 |

**Accuracy:** (68+1)/123 = 56.1%

**Precision:** 68/123 = 55.3%

**Specificity:** 1/123 = 0.8%

|  |  |  |
| --- | --- | --- |
| Naïve Bayes | Test Prediction |  |
| Test Actual | **B** | **M** |
| B | 68 | 3 |
| M | 50 | 2 |

**Accuracy:** (68+2)/123 = 56.9%

**Precision:** 68/123 = 55.3%

**Specificity:** 2/123 = 1.6%

Bases on these values the algorithms are obviously skewed to considering nearly all detections as benign which is not a good test.

**Code:**

**library(ISLR2)**

**library(MASS)**

**library(ggplot2)**

**library(reshape2)**

**library(e1071)**

**library(utils)**

**library(ggplot2)**

**data <- read.csv('./Workspace/UMKC/Fall2024/CS5565-0007/classlab/synthetic\_data\_25.csv')**

**plot(data)**

**#Create a numeric column based on M or B as 0/1**

**data$diagnosis\_binary <- ifelse(data$diagnosis=='B',1,0)**

**data\_cor <- data[,!names(data) %in% c('diagnosis')]**

**print(summary(data\_cor))**

**cors <- cor(data\_cor)**

**print(cors[,31])**

**doLogistic <- function(Y,X,data,ylabel,xlabel){**

**log\_model <- glm(Y ~ X, data=data, family=binomial)**

**Predicted\_data <- data.frame(X = seq(min(X), max(X), len=nrow(data)))**

**# Fill predicted values using regression model**

**Predicted\_data$var1 = predict(log\_model, Predicted\_data, type="response")**

**# Plot Predicted data and original data points**

**plot(y=Y, x=X, xlab=xlabel, ylab=ylabel)**

**lines(Predicted\_data$var1 ~ Predicted\_data$X, Predicted\_data, lwd=1, col="green")**

**return(log\_model)**

**}**

**doLogistic(data$diagnosis\_binary, data$perimeter\_mean, data, 'Diagnosis (B==1)', 'Perimeter mean' )**

**doLogistic(data$diagnosis\_binary, data$compactness\_mean, data, 'Diagnosis (B==1)', 'Compactnessmean' )**

**doLogistic(data$diagnosis\_binary, data$concavity\_worst, data, 'Diagnosis (B==1)', 'Concavity worst' )**

**doLogistic(data$diagnosis\_binary, data$texture\_se, data, 'Diagnosis (B==1)', 'Texture se' )**

**doLogistic(data$diagnosis\_binary, data$texture\_worst, data, 'Diagnosis (B==1)', 'Texture worst' )**

**doLogistic(data$diagnosis\_binary, data$area\_worst, data, 'Diagnosis (B==1)', 'Area worst' )**

**doLogistic(data$diagnosis\_binary, data$symmetry\_worst, data, 'Diagnosis (B==1)', 'Symmetry worst' )**

**#split data into train and test**

**#From https://www.statology.org/train-test-split-r/**

**set.seed(5)**

**sample <- sample(c(TRUE,FALSE), nrow(data), replace=TRUE, prob=c(0.8, 0.2))**

**data\_train <- data[sample,]**

**data\_test <- data[!sample,]**

**print(nrow(data\_train))**

**print(nrow(data\_test))**

**mlog\_model <- glm(**

**diagnosis\_binary ~ perimeter\_mean + compactness\_mean + concavity\_worst + texture\_se + texture\_worst + area\_worst + symmetry\_worst,**

**data = data\_train,**

**family=binomial)**

**summary(mlog\_model)**

**data\_test$probs <- predict(mlog\_model, newdata = data\_test, type="response")**

**data\_test$pred <- rep('B',nrow(data\_test))**

**data\_test$pred[data\_test$probs < 0.5] = 'M'**

**test\_actual = data\_test$diagnosis**

**test\_prediction = data\_test$pred**

**table(test\_actual, test\_prediction)**

**lda\_model = lda(**

**diagnosis ~ perimeter\_mean + compactness\_mean + concavity\_worst + texture\_se + texture\_worst + area\_worst + symmetry\_worst,**

**data = data\_train**

**)**

**plot(lda\_model)**

**lda\_pred <- predict(lda\_model, newdata = data\_test)**

**names(lda\_pred)**

**lda\_class <- lda\_pred$class**

**table(test\_actual,lda\_class)**

**bayes\_model <- naiveBayes(**

**diagnosis ~ perimeter\_mean + compactness\_mean + concavity\_worst + texture\_se + texture\_worst + area\_worst + symmetry\_worst,**

**data = data\_train**

**)**

**print(bayes\_model)**

**bayes\_class <- predict(bayes\_model, data\_test)**

**table(test\_actual,bayes\_class)**

**Console Output:**

**library(ISLR2)**

**> library(MASS)**

**> library(ggplot2)**

**> library(reshape2)**

**> library(e1071)**

**> library(utils)**

**> library(ggplot2)**

**>**

**> data <- read.csv('./Workspace/UMKC/Fall2024/CS5565-0007/classlab/synthetic\_data\_25.csv')**

**> plot(data)**

**>**

**> #Create a numeric column based on M or B as 0/1**

**> data$diagnosis\_binary <- ifelse(data$diagnosis=='B',1,0)**

**> data\_cor <- data[,!names(data) %in% c('diagnosis')]**

**> print(summary(data\_cor))**

**radius\_mean texture\_mean perimeter\_mean area\_mean smoothness\_mean compactness\_mean concavity\_mean**

**Min. :-3.4178 Min. :-4.02262 Min. :-0.526 Min. :-0.6009 Min. :-2.7145 Min. :-0.7842 Min. :-2.8907**

**1st Qu.:-1.5120 1st Qu.:-2.42148 1st Qu.: 1.384 1st Qu.: 0.6356 1st Qu.:-0.3875 1st Qu.: 1.2106 1st Qu.:-1.4416**

**Median :-0.9544 Median :-1.95537 Median : 1.850 Median : 1.0793 Median : 0.1918 Median : 1.6624 Median :-1.0098**

**Mean :-0.9668 Mean :-1.95927 Mean : 1.880 Mean : 1.0852 Mean : 0.1756 Mean : 1.7216 Mean :-1.0147**

**3rd Qu.:-0.4209 3rd Qu.:-1.49320 3rd Qu.: 2.407 3rd Qu.: 1.5140 3rd Qu.: 0.7176 3rd Qu.: 2.2776 3rd Qu.:-0.5741**

**Max. : 1.0166 Max. :-0.04259 Max. : 4.611 Max. : 2.9893 Max. : 2.5994 Max. : 4.2270 Max. : 0.9215**

**concave.points\_mean symmetry\_mean fractal\_dimension\_mean radius\_se texture\_se perimeter\_se**

**Min. :-4.4127 Min. :-0.9808 Min. :-4.2198 Min. :-5.4243 Min. :-0.370 Min. :-1.9472**

**1st Qu.:-1.9047 1st Qu.: 1.0871 1st Qu.:-2.0494 1st Qu.:-2.9063 1st Qu.: 1.431 1st Qu.:-0.2384**

**Median :-1.4168 Median : 1.5973 Median :-1.5226 Median :-2.2954 Median : 1.968 Median : 0.3362**

**Mean :-1.4399 Mean : 1.5925 Mean :-1.4917 Mean :-2.3461 Mean : 2.024 Mean : 0.3357**

**3rd Qu.:-0.9900 3rd Qu.: 2.1045 3rd Qu.:-0.9558 3rd Qu.:-1.7652 3rd Qu.: 2.554 3rd Qu.: 0.8893**

**Max. : 0.6546 Max. : 4.1408 Max. : 0.9432 Max. :-0.2113 Max. : 4.952 Max. : 2.8637**

**area\_se smoothness\_se compactness\_se concavity\_se concave.points\_se symmetry\_se fractal\_dimension\_se**

**Min. :-4.2353 Min. :-4.6829 Min. :-3.5076 Min. :-4.4236 Min. :-5.0738 Min. :-0.7363 Min. :-0.7126**

**1st Qu.:-2.1373 1st Qu.:-2.1588 1st Qu.:-1.5556 1st Qu.:-1.9862 1st Qu.:-2.4151 1st Qu.: 1.2414 1st Qu.: 0.4804**

**Median :-1.6986 Median :-1.6008 Median :-1.0945 Median :-1.4970 Median :-1.8682 Median : 1.7827 Median : 0.9258**

**Mean :-1.6786 Mean :-1.6137 Mean :-1.1002 Mean :-1.5142 Mean :-1.8644 Mean : 1.7731 Mean : 0.9106**

**3rd Qu.:-1.1736 3rd Qu.:-1.0517 3rd Qu.:-0.5987 3rd Qu.:-0.9822 3rd Qu.:-1.2351 3rd Qu.: 2.3224 3rd Qu.: 1.3190**

**Max. : 0.3742 Max. : 0.7963 Max. : 0.9093 Max. : 0.9799 Max. : 0.7484 Max. : 3.9893 Max. : 2.6479**

**radius\_worst texture\_worst perimeter\_worst area\_worst smoothness\_worst compactness\_worst concavity\_worst**

**Min. :-3.5349 Min. :-0.5888 Min. :-4.3341 Min. :-4.3878 Min. :-1.090 Min. :-0.8873 Min. :-3.3283**

**1st Qu.:-1.6436 1st Qu.: 0.7838 1st Qu.:-2.0740 1st Qu.:-2.3438 1st Qu.: 0.611 1st Qu.: 0.6965 1st Qu.:-1.7440**

**Median :-1.0519 Median : 1.2162 Median :-1.6267 Median :-1.7105 Median : 1.092 Median : 1.1983 Median :-1.2343**

**Mean :-1.1070 Mean : 1.2487 Mean :-1.6252 Mean :-1.7483 Mean : 1.102 Mean : 1.2296 Mean :-1.2580**

**3rd Qu.:-0.5829 3rd Qu.: 1.7243 3rd Qu.:-1.1924 3rd Qu.:-1.1671 3rd Qu.: 1.568 3rd Qu.: 1.7456 3rd Qu.:-0.7835**

**Max. : 1.1102 Max. : 3.1851 Max. : 0.1877 Max. : 0.4173 Max. : 3.226 Max. : 3.6788 Max. : 1.1685**

**concave.points\_worst symmetry\_worst fractal\_dimension\_worst diagnosis\_binary**

**Min. :-0.850 Min. :-3.2065 Min. :-0.2828 Min. :0.000**

**1st Qu.: 1.185 1st Qu.:-1.3677 1st Qu.: 0.8649 1st Qu.:0.000**

**Median : 1.657 Median :-0.8752 Median : 1.3742 Median :1.000**

**Mean : 1.700 Mean :-0.8935 Mean : 1.3464 Mean :0.652**

**3rd Qu.: 2.231 3rd Qu.:-0.4064 3rd Qu.: 1.7413 3rd Qu.:1.000**

**Max. : 4.961 Max. : 1.3224 Max. : 3.8110 Max. :1.000**

**>**

**> cors <- cor(data\_cor)**

**> print(cors[,31])**

**radius\_mean texture\_mean perimeter\_mean area\_mean smoothness\_mean**

**-0.0142958682 -0.0061115131 -0.0244259786 0.0181597250 -0.0156140185**

**compactness\_mean concavity\_mean concave.points\_mean symmetry\_mean fractal\_dimension\_mean**

**-0.0564290571 0.0425896740 -0.0223046357 -0.0042004457 -0.0094238678**

**radius\_se texture\_se perimeter\_se area\_se smoothness\_se**

**0.0007201138 0.0481677994 0.0221272425 0.0126192036 0.0829450928**

**compactness\_se concavity\_se concave.points\_se symmetry\_se fractal\_dimension\_se**

**0.0393272324 0.0400695099 0.0064158420 0.0013924635 0.0004565889**

**radius\_worst texture\_worst perimeter\_worst area\_worst smoothness\_worst**

**-0.0308126278 -0.0490489466 -0.0325639192 0.0758089970 -0.0136933749**

**compactness\_worst concavity\_worst concave.points\_worst symmetry\_worst fractal\_dimension\_worst**

**-0.0247695773 -0.0431186452 0.0019702958 0.0681176486 -0.0175170943**

**diagnosis\_binary**

**1.0000000000**

**>**

**> doLogistic <- function(Y,X,data,ylabel,xlabel){**

**+ log\_model <- glm(Y ~ X, data=data, family=binomial)**

**+**

**+ Predicted\_data <- data.frame(X = seq(min(X), max(X), len=nrow(data)))**

**+**

**+ # Fill predicted values using regression model**

**+ Predicted\_data$var1 = predict(log\_model, Predicted\_data, type="response")**

**+**

**+ # Plot Predicted data and original data points**

**+ plot(y=Y, x=X, xlab=xlabel, ylab=ylabel)**

**+ lines(Predicted\_data$var1 ~ Predicted\_data$X, Predicted\_data, lwd=1, col="green")**

**+**

**+ return(log\_model)**

**+ }**

**>**

**> doLogistic(data$diagnosis\_binary, data$perimeter\_mean, data, 'Diagnosis (B==1)', 'Perimeter mean' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.75979 -0.06991**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 735 AIC: 739**

**> doLogistic(data$diagnosis\_binary, data$compactness\_mean, data, 'Diagnosis (B==1)', 'Compactnessmean' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.8982 -0.1558**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 733.5 AIC: 737.5**

**> doLogistic(data$diagnosis\_binary, data$concavity\_worst, data, 'Diagnosis (B==1)', 'Concavity worst' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.4709 -0.1258**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 734.3 AIC: 738.3**

**> doLogistic(data$diagnosis\_binary, data$texture\_se, data, 'Diagnosis (B==1)', 'Texture se' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.3785 0.1240**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 734 AIC: 738**

**> doLogistic(data$diagnosis\_binary, data$texture\_worst, data, 'Diagnosis (B==1)', 'Texture worst' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.8239 -0.1557**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 734 AIC: 738**

**> doLogistic(data$diagnosis\_binary, data$area\_worst, data, 'Diagnosis (B==1)', 'Area worst' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.9617 0.1887**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 732.1 AIC: 736.1**

**> doLogistic(data$diagnosis\_binary, data$symmetry\_worst, data, 'Diagnosis (B==1)', 'Symmetry worst' )**

**Call: glm(formula = Y ~ X, family = binomial, data = data)**

**Coefficients:**

**(Intercept) X**

**0.8045 0.1941**

**Degrees of Freedom: 568 Total (i.e. Null); 567 Residual**

**Null Deviance: 735.4**

**Residual Deviance: 732.7 AIC: 736.7**

**>**

**> #split data into train and test**

**> #From https://www.statology.org/train-test-split-r/**

**> set.seed(5)**

**> sample <- sample(c(TRUE,FALSE), nrow(data), replace=TRUE, prob=c(0.8, 0.2))**

**> data\_train <- data[sample,]**

**> data\_test <- data[!sample,]**

**>**

**> print(nrow(data\_train))**

**[1] 446**

**> print(nrow(data\_test))**

**[1] 123**

**>**

**> mlog\_model <- glm(**

**+ diagnosis\_binary ~ perimeter\_mean + compactness\_mean + concavity\_worst + texture\_se + texture\_worst + area\_worst + symmetry\_worst,**

**+ data = data\_train,**

**+ family=binomial)**

**>**

**> summary(mlog\_model)**

**Call:**

**glm(formula = diagnosis\_binary ~ perimeter\_mean + compactness\_mean +**

**concavity\_worst + texture\_se + texture\_worst + area\_worst +**

**symmetry\_worst, family = binomial, data = data\_train)**

**Coefficients:**

**Estimate Std. Error z value Pr(>|z|)**

**(Intercept) 0.95358 0.56870 1.677 0.0936 .**

**perimeter\_mean 0.01929 0.14777 0.131 0.8962**

**compactness\_mean -0.25631 0.13909 -1.843 0.0654 .**

**concavity\_worst -0.20134 0.14724 -1.367 0.1715**

**texture\_se 0.17728 0.13091 1.354 0.1757**

**texture\_worst 0.02348 0.15713 0.149 0.8812**

**area\_worst 0.15624 0.12344 1.266 0.2056**

**symmetry\_worst 0.20242 0.14357 1.410 0.1586**

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**(Dispersion parameter for binomial family taken to be 1)**

**Null deviance: 564.00 on 445 degrees of freedom**

**Residual deviance: 553.45 on 438 degrees of freedom**

**AIC: 569.45**

**Number of Fisher Scoring iterations: 4**

**>**

**> data\_test$probs <- predict(mlog\_model, newdata = data\_test, type="response")**

**> data\_test$pred <- rep('B',nrow(data\_test))**

**> data\_test$pred[data\_test$probs < 0.5] = 'M'**

**> test\_actual = data\_test$diagnosis**

**> test\_prediction = data\_test$pred**

**>**

**> table(test\_actual, test\_prediction)**

**test\_prediction**

**test\_actual B M**

**B 68 3**

**M 51 1**

**>**

**> lda\_model = lda(**

**+ diagnosis ~ perimeter\_mean + compactness\_mean + concavity\_worst + texture\_se + texture\_worst + area\_worst + symmetry\_worst,**

**+ data = data\_train**

**+ )**

**> plot(lda\_model)**

**>**

**> lda\_pred <- predict(lda\_model, newdata = data\_test)**

**> names(lda\_pred)**

**[1] "class" "posterior" "x"**

**> lda\_class <- lda\_pred$class**

**> table(test\_actual,lda\_class)**

**lda\_class**

**test\_actual B M**

**B 68 3**

**M 51 1**

**>**

**> bayes\_model <- naiveBayes(**

**+ diagnosis ~ perimeter\_mean + compactness\_mean + concavity\_worst + texture\_se + texture\_worst + area\_worst + symmetry\_worst,**

**+ data = data\_train**

**+ )**

**> print(bayes\_model)**

**Naive Bayes Classifier for Discrete Predictors**

**Call:**

**naiveBayes.default(x = X, y = Y, laplace = laplace)**

**A-priori probabilities:**

**Y**

**B M**

**0.6726457 0.3273543**

**Conditional probabilities:**

**perimeter\_mean**

**Y [,1] [,2]**

**B 1.895241 0.7124754**

**M 1.931884 0.7555208**

**compactness\_mean**

**Y [,1] [,2]**

**B 1.640175 0.7523012**

**M 1.767563 0.7408640**

**concavity\_worst**

**Y [,1] [,2]**

**B -1.270854 0.7006387**

**M -1.173211 0.7600113**

**texture\_se**

**Y [,1] [,2]**

**B 2.051918 0.7927857**

**M 1.973955 0.8279037**

**texture\_worst**

**Y [,1] [,2]**

**B 1.255428 0.6632397**

**M 1.280472 0.6744733**

**area\_worst**

**Y [,1] [,2]**

**B -1.731453 0.8458489**

**M -1.850947 0.9124788**

**symmetry\_worst**

**Y [,1] [,2]**

**B -0.8589126 0.7373972**

**M -0.9834440 0.7038457**

**> bayes\_class <- predict(bayes\_model, data\_test)**

**> table(test\_actual,bayes\_class)**

**bayes\_class**

**test\_actual B M**

**B 68 3**

**M 50 2**