STAT 223: Project 3

Nrepesh Joshi

Professor Kevin Hastings

Applied Analytics

5/4/20

Breast Cancer Classification Logistic Regression

==========

BRIEF DESCRIPTION OF THE DATA

==========

In this study, the goal was to identify numerical factors based on measurements taken from needle biopsies of masses from 569 women that would distinguish cancerous from non-cancerous tumors, in order to assist physicians in accurate diagnosis and appropriate treatment. The data contains the target variable “diagnosis” which is the final, presumed correct, diagnosis of whether the tumor is malignant or benign.

==========

HOW TO USE THE DATA FILE

==========

1. Title: Wisconsin Breast Cancer Data (wbcd0.csv)
2. Relevant Information:

From the original full dataset, this data has been reduced from 30 predictors to 10 predictors, which are means of several

measurements made on the 10 variables. The largest measurement and standard deviation for each variable make up the 20

predictors that have been excluded from this shortened data set. An id number in the original data has also been excluded.

3. Number of Instances: 569.
4. Number of Attributes: 10, together with the diagnosis categorical response variable.
5. Attribute Information:

1. Diagnosis: character (B for benign, M for malignant)

2. Radius: numeric

3. Texture: numeric

4. Perimeter: numeric

5. Area: numeric

6. Smoothness: numeric

7. Compactness: numeric

8. Concavity: numeric

9. (Concave) points: numeric

10. Symmetry: numeric

11, (Fractal) dimension: numeric

6. Missing Attribute Values: none. A few Concavity and Points observations are 0, but we take those as legitimate observations

==========

QUESTIONS

==========

1.The data file is in Google Classroom for download under the name “wbcd0.csv”. Bring it into R, and compute summary statistics for each variable.

= Imported data and used the summary command:

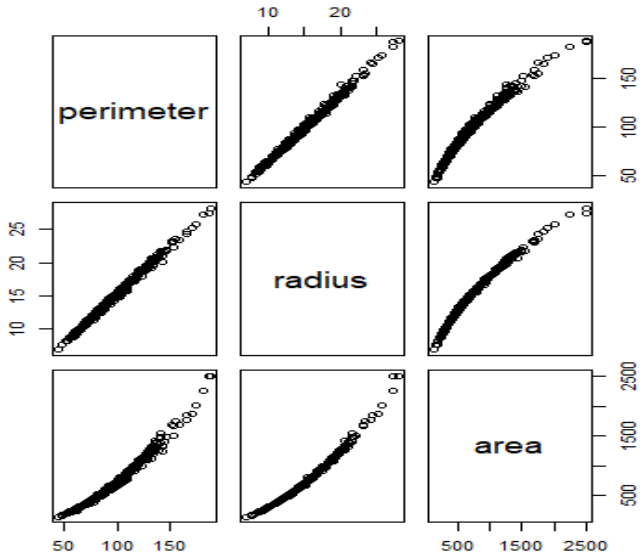
```
> #------------------------------------------------------
> # Part 1
> # Summary for each variable
> summary(dset)
 diagnosis     radius          texture         perimeter          area          smoothness
 B:357     Min.   : 6.981   Min.   : 9.71   Min.   : 43.79   Min.   : 143.5   Min.   :0.05263
 M:212     1st Qu.:11.700   1st Qu.:16.17   1st Qu.: 75.17   1st Qu.: 420.3   1st Qu.:0.08637
           Median :13.370   Median :18.84   Median : 86.24   Median : 551.1   Median :0.09587
           Mean   :14.127   Mean   :19.29   Mean   : 91.97   Mean   : 654.9   Mean   :0.09636
           3rd Qu.:15.780   3rd Qu.:21.80   3rd Qu.:104.10   3rd Qu.: 782.7   3rd Qu.:0.10530
           Max.   :28.110   Max.   :39.28   Max.   :188.50   Max.   :2501.0   Max.   :0.16340
  compactness        concavity           points           symmetry        dimension
 Min.   :0.01938   Min.   :0.00000   Min.   :0.00000   Min.   :0.1060   Min.   :0.04996
 1st Qu.:0.06492   1st Qu.:0.02956   1st Qu.:0.02031   1st Qu.:0.1619   1st Qu.:0.05770
 Median :0.09263   Median :0.06154   Median :0.03350   Median :0.1792   Median :0.06154
 Mean   :0.10434   Mean   :0.08880   Mean   :0.04892   Mean   :0.1812   Mean   :0.06280
 3rd Qu.:0.13040   3rd Qu.:0.13070   3rd Qu.:0.07400   3rd Qu.:0.1957   3rd Qu.:0.06612
 Max.   :0.34540   Max.   :0.42680   Max.   :0.20120   Max.   :0.3040   Max.   :0.09744
```

We notice that the dataset has 357 benign and 212 malignant data which is about equal and will not to cause any biases. All other features are either geometrical or a measurement of the sample and do not show anything extraordinary.

2. Three of the observations, namely radius, perimeter, and area, are clearly geometric and may show strong correlation with each other. Check graphically and analytically if this is the case. If so, what does that mean for a regression model?

= I plotted a pair plot of radius, perimeter and area features and also used the cor() command to get the numerical correlation.

```
> #-------------------------------------------------------
> # Part 2
> # Correlation between radius, area and perimeter
> pairs(dset[c("perimeter","radius","area")])
> cor(dset$perimeter, dset$radius)
[1] 0.9978553
> cor(dset$area, dset$radius)
[1] 0.9873572
> cor(dset$perimeter, dset$area)
[1] 0.9865068
```

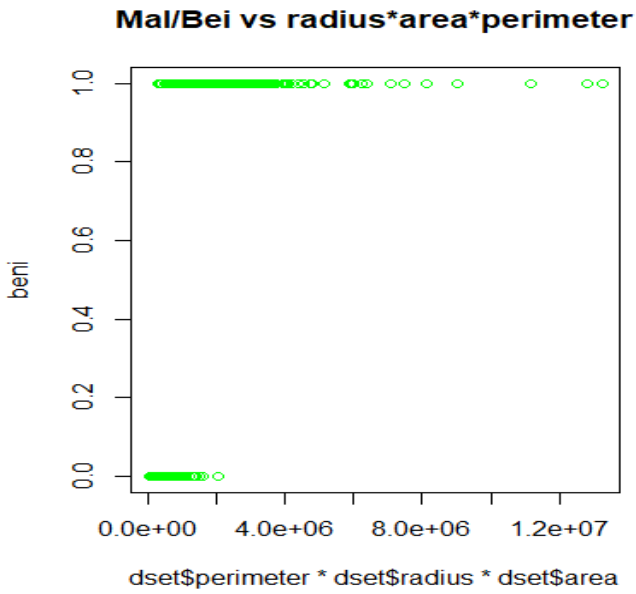


The speculation in the question about geometrical features being strongly correlated was correct. The 3 features as seen above are correlated to 98% and above correlation with each other. In addition, we see from our pair plot that we see a linearly increasing pattern between radius and area. A bit curved linear pattern between area and perimeter, also radius and area but this only supports the claim that they are highly correlated.

The features are highly correlated but that does not always mean that it is a good sign. Since all three are highly correlated (in other words multicollinearity) these result in redundancy in the model. Multicollinearity increases the standard errors of the coefficients which means that some independent variables may not be significantly different form 0. To reduce standard error, I plan to combine all three features into an interaction term which might make other coefficient significant.

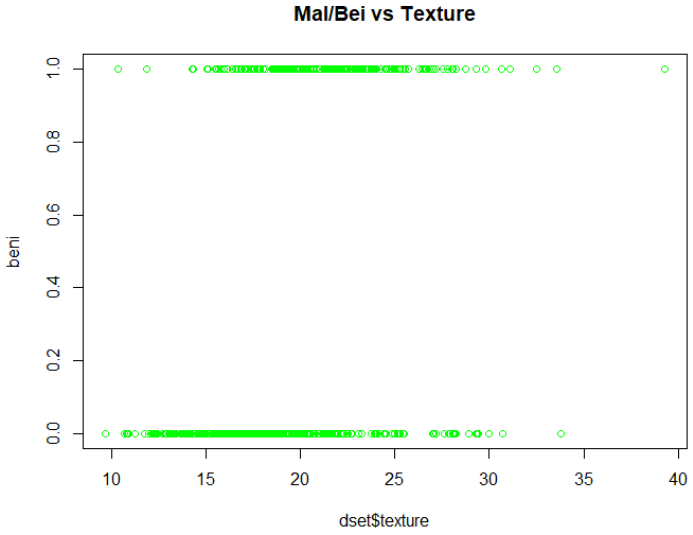
3. Encode a new variable as 1 if malignant and 0 otherwise for each observation in the data set. Bearing in mind your answer to question 2, produce relevant graphs of this malignancy indicator variable (on the y-axis) vs. each predictor that you plan to keep. What do these graphs seem to say about the accuracy we can expect if we use single predictor variables to classify the tumors?

```
> #-------------------------------------------------------
> # Part 3
> beni <- ifelse(dset$diagnosis == "B",0,1);
```



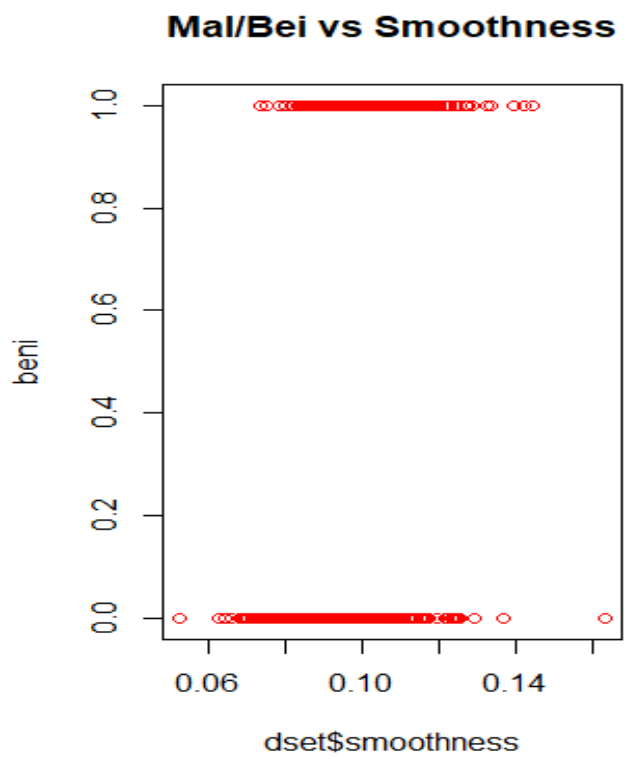

```
Call:
glm(formula = beni ~ dset$geo, family = binomial, data = dset)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-3.06239  -0.45749  -0.26342   0.02979   2.53926

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -4.506e+00  3.712e-01  -12.14   <2e-16 ***
dset$geo     4.526e-06  4.393e-07   10.30   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 323.15  on 567  degrees of freedom
AIC: 327.15

Number of Fisher Scoring iterations: 8
```

To remove multicollinearity and reduce standard error we combine all the geometric terms to one variable. This variable has less deviance and less AIC which means that adding a predictor will not affect the model but can make it better (nesting). From the graph we see that it follows a logistic 'S' curve so because of these we will keep this in our model.



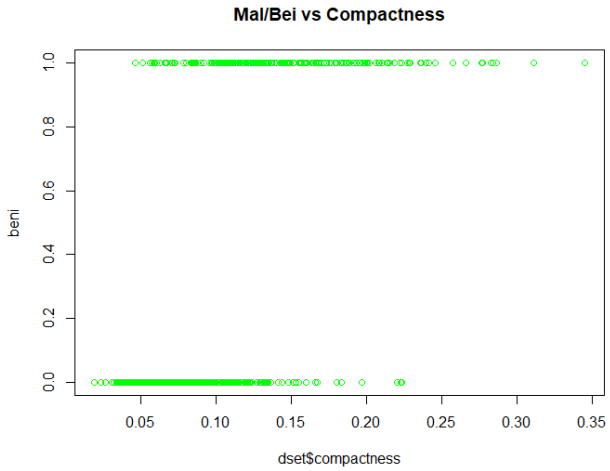

```
Call:
glm(formula = beni ~ dset$texture, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3942  -0.8451  -0.5881   1.1022   2.3477

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -5.12577    0.52638  -9.738   <2e-16 ***
dset$texture  0.23464    0.02614   8.975   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 646.52  on 567  degrees of freedom
AIC: 650.52

Number of Fisher Scoring iterations: 4
```

This feature has a high AIC and deviance which means it does not suggest adding another predictor to the model. The figure does show a strong logistic “S” curve so because of this we will keep this in our model. A reasonable cutoff can classify the data as predicted by our logistic regression.



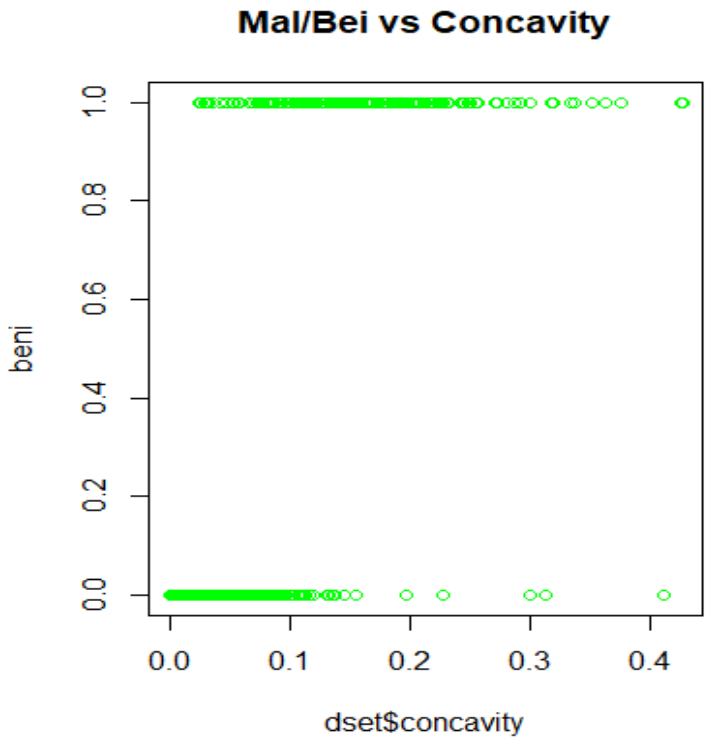

```
Call:
glm(formula = beni ~ dset$smoothness, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6352  -0.9148  -0.6357   1.1428   2.0404

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)     -6.3773     0.7474  -8.532  < 2e-16 ***
dset$smoothness 60.0857     7.5497   7.959 1.74e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 673.95  on 567  degrees of freedom
AIC: 677.95

Number of Fisher Scoring iterations: 3
```

This feature also has a high AIC and deviance which means it does not suggest adding another predictor to the model. The figure does not show a strong logistic curve but a rather lines stacking on top of each other. This would not be a good estimate for our model because there would be no reasonable cutoff value (other than 0 or 1) to successfully classify the results from our logistic regression. Hence, we will not include this feature in our model.



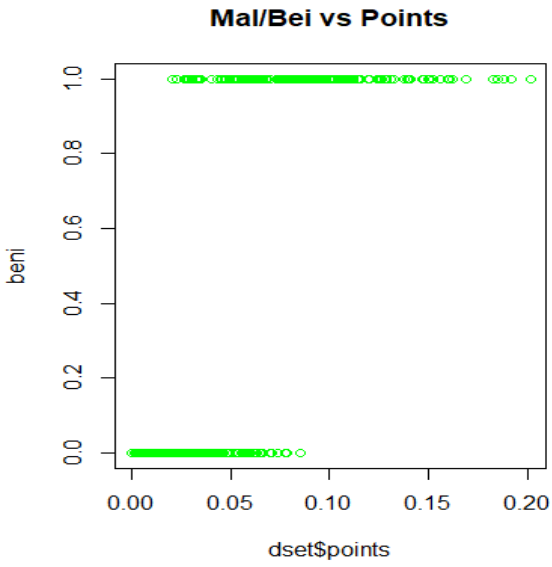

```
Call:
glm(formula = beni ~ dset$compactness, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7454  -0.6513  -0.3985   0.6546   2.3615

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)      -4.4001     0.3563  -12.35   <2e-16 ***
dset$compactness 36.3798     3.1868   11.42   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 508.79  on 567  degrees of freedom
AIC: 512.79

Number of Fisher Scoring iterations: 5
```

This feature has a high AIC and deviance which means it does not suggest adding another predictor to the model. The figure does show a strong logistic “S” curve so because of this we will keep this in our model. A reasonable cutoff can classify the data as predicted by our logistic regression.



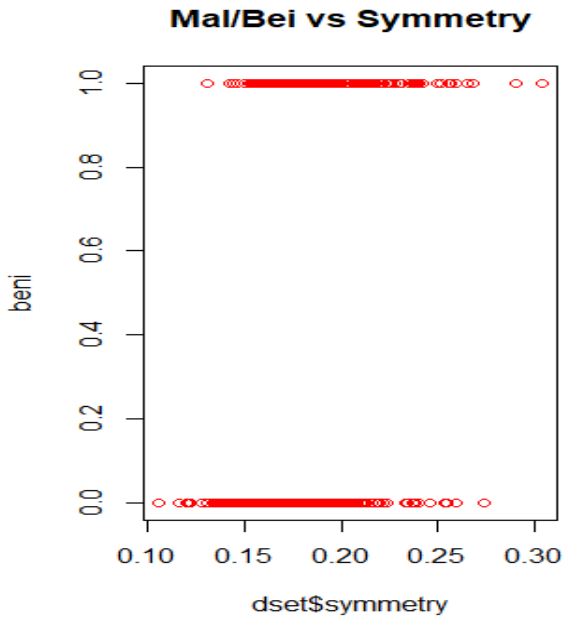

```
Call:
glm(formula = beni ~ dset$concavity, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.7647  -0.4612  -0.2912   0.3293   2.4310

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     -3.7850     0.2914  -12.99   <2e-16 ***
dset$concavity  36.8457     3.0314   12.15   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 383.23  on 567  degrees of freedom
AIC: 387.23

Number of Fisher Scoring iterations: 6
```

This variable has less deviance and less AIC which means that adding a predictor will not affect the model but can make it better (nesting). From the graph we see that it follows a logistic ‘S’ curve so because of these we will keep this in our model. A reasonable cutoff can classify the data as predicted by our logistic regression.



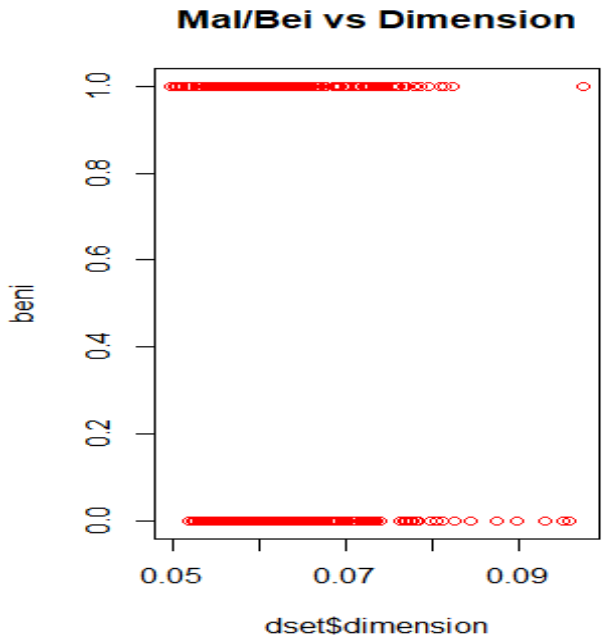

```
Call:
glm(formula = beni ~ dset$points, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5783  -0.3251  -0.1532   0.1573   2.7187

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -5.8437     0.4803  -12.17   <2e-16 ***
dset$points 106.9937     9.1190   11.73   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 258.92  on 567  degrees of freedom
AIC: 262.92

Number of Fisher Scoring iterations: 7
```

This variable has less deviance and less AIC which means that adding a predictor will not affect the model but can make it better (nesting). From the graph we see that it follows a logistic 'S' curve so because of these we will keep this in our model. A reasonable cutoff can classify the data as predicted by our logistic regression.




```
Call:
glm(formula = beni ~ dset$symmetry, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0637  -0.9187  -0.6879   1.1674   2.0446

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    -5.5689     0.6981  -7.977 1.50e-15 ***
dset$symmetry  27.6042     3.7655   7.331 2.29e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 686.80  on 567  degrees of freedom
AIC: 690.8

Number of Fisher Scoring iterations: 4
```

This feature has a high AIC and deviance which means it does not suggest adding another predictor to the model. The figure does not show a strong logistic curve but a rather lines stacking on top of each other. This would not be a good estimate for our model because there would be no reasonable cutoff value (other than 0 or 1) to successfully classify the results from our logistic regression. Hence, we will not include this feature in our model.




```
Call:
glm(formula = beni ~ dset$dimension, family = binomial, data = dset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.9816  -0.9692  -0.9598   1.3985   1.4639

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.2837     0.7799  -0.364    0.716
dset$dimension  -3.7819    12.3519  -0.306    0.759

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 751.35  on 567  degrees of freedom
AIC: 755.35

Number of Fisher Scoring iterations: 4
```

This feature has a high AIC and deviance which means it does not suggest adding another predictor to the model. The figure does not show a strong logistic curve but a rather lines stacking on top of each other. This would not be a good estimate for our model because there would be no reasonable cutoff value (other than 0 or 1) to successfully classify the results from our logistic regression. Hence, we will not include this feature in our model.

4. Bearing in mind the ideas of deviance reduction and desired small AIC, try to find the best logistic regression model that you can that predicts malignancy based on the first 469 of the observations. Then, using the last 100 observations as a test data set, determine how effective your model is at classifying tumors. (You should produce the “confusion matrix”.)
Remembering that logistic regression returns probability values, try several values of the cutoff probability for the decision of whether a tumor is malignant or not to look for one that produces the best results in your judgement.

```
Call:
glm(formula = beni ~ geo + texture + smoothness + compactness +
    concavity + points + symmetry + dimensions, family = binomial,
    data = cred_train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.09751  -0.15406  -0.04557   0.01022   2.78212

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.734e+01  5.477e+00  -3.167  0.00154 **
geo          3.098e-06  9.905e-07   3.127  0.00176 **
texture      3.668e-01  6.767e-02   5.420 5.96e-08 ***
smoothness   6.782e+01  3.394e+01   1.998  0.04567 *
compactness -1.326e+01  1.316e+01  -1.008  0.31343
concavity    1.053e+01  8.383e+00   1.256  0.20920
points       8.044e+01  2.988e+01   2.692  0.00711 **
symmetry     9.782e+00  1.150e+01   0.850  0.39510
dimensions  -7.965e+01  8.592e+01  -0.927  0.35394
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 617.53  on 468  degrees of freedom
Residual deviance: 125.60  on 460  degrees of freedom
AIC: 143.6

Number of Fisher Scoring iterations: 8

> anova(model1, test = "Chisq")
Analysis of Deviance Table

Model: binomial, link: logit

Response: beni

Terms added sequentially (first to last)


            Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                          468     617.53
geo          1   351.15       467     266.38 < 2.2e-16 ***
texture      1    31.98       466     234.40 1.559e-08 ***
smoothness   1    76.62       465     157.78 < 2.2e-16 ***
compactness  1     4.44       464     153.35 0.0351951 *
concavity    1    13.76       463     139.59 0.0002080 ***
points       1    12.32       462     127.27 0.0004483 ***
symmetry     1     0.78       461     126.49 0.3773714
dimensions   1     0.89       460     125.60 0.3461447
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

For the first model, I tried combining all three geometric terms (radius, area, perimeter) into one variable called geo. I also used texture, smoothness, compactness, concavity, points, symmetry and dimensions which are the rest of the variables. The model did not do too bad with a relatively low AIC and deviance value. One thing I did notice was that quite few variables were not statistically significant so in another model, I removed those values and trained the model again. Even the Chi Sq test is not significant for a few features.

```
Call:
glm(formula = beni ~ geo + texture + compactness + points, family = binomial,
    data = cred_train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.09343  -0.16451  -0.06024   0.01522   2.58785

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.352e+01  1.825e+00  -7.409 1.28e-13 ***
geo          2.298e-06  6.201e-07   3.706 0.000211 ***
texture      3.294e-01  6.218e-02   5.298 1.17e-07 ***
compactness -1.556e+01  9.073e+00  -1.715 0.086376 .
points       1.251e+02  2.178e+01   5.744 9.26e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 617.53  on 468  degrees of freedom
Residual deviance: 131.77  on 464  degrees of freedom
AIC: 141.77

Number of Fisher Scoring iterations: 8

> anova(model2, test = "Chisq")
Analysis of Deviance Table

Model: binomial, link: logit

Response: beni

Terms added sequentially (first to last)


            Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                          468     617.53
geo          1   351.15       467     266.38 < 2.2e-16 ***
texture      1    31.98       466     234.40 1.559e-08 ***
compactness  1    56.16       465     178.24 6.669e-14 ***
points       1    46.47       464     131.77 9.314e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

For the second model, I removed concavity, symmetry and dimensions from the model and trained it again. We see that all the features are significant in a significance level of 0.08. The AIC of 141.77 has reduced but the deviance of 131.77 remains the same. I thought that this would be a good enough model for testing, and I proceeded forward.

```
> library(gmodels)

> credpreds1 <- predict(model2, newdata = cred_test, type = "response")

> pre <- ifelse(credpreds1>.4,1,0);

> CrossTable(beni[470:569], pre[1:100])


   Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  100


               | pre[1:100]
beni[470:569] |         0 |         1 | Row Total |
--------------|-----------|-----------|-----------|
            0 |        59 |         2 |        61 |
              |    11.010 |    18.747 |           |
              |     0.967 |     0.033 |     0.610 |
              |     0.937 |     0.054 |           |
              |     0.590 |     0.020 |           |
--------------|-----------|-----------|-----------|
            1 |         4 |        35 |        39 |
              |    17.221 |    29.323 |           |
              |     0.103 |     0.897 |     0.390 |
              |     0.063 |     0.946 |           |
              |     0.040 |     0.350 |           |
--------------|-----------|-----------|-----------|
 Column Total |        63 |        37 |       100 |
              |     0.630 |     0.370 |           |
--------------|-----------|-----------|-----------|


> print("accuracy is 94%" )
[1] "accuracy is 94%"
```

For a total of 100 samples, our model has very few false positives and false negatives which is a good sign. I tried various cut offs for my linear regression model, but only 0.4 and 0.5 gave a 94% accuracy. The reason I chose 0.4 cut off as the best predictor is because it has less false positives.

False positives are more dangerous in this situation because it is worse to tell a cancer patient that the do not have cancer when they do. Our model classifies 2 false negatives which means that people were told they had cancer when they did not.

For the accuracy, I did (true positives + true negatives)/total. Which is (59+35)/100. A 94% accuracy is very good for such a simple logistic regression model. The error rate for the model is 6% which is also a good sign.

```
#-----------------------------------------------------
# Importing data

dset <- read.csv("wbcd0.csv", header = TRUE, sep="\t"); View(dset)
head(dset)

#-----------------------------------------------------
# Part 1
# Summary for each variable
summary(dset)



#-----------------------------------------------------
# Part 2
# Correlation between radius, area and perimeter
pairs(dset[c("perimeter","radius","area")])
cor(dset$perimeter, dset$radius)
cor(dset$area, dset$radius)
cor(dset$perimeter, dset$area)

#-----------------------------------------------------
# Part 3
beni <- ifelse(dset$diagnosis == "B",0,1);
plot(dset$perimeter,beni, col = "red")
modelperi <- glm(beni ~ dset$perimeter, data = dset, family=binomial); summary(modelperi)

plot(dset$area,beni, col = "red")
modelarea <- glm(beni ~ dset$area, data = dset, family=binomial); summary(modelarea)

plot(dset$radius,beni, col = "red")
modelradius <- glm(beni ~ dset$radius, data = dset, family=binomial); summary(modelradius)

plot(dset$perimeter*dset$radius*dset$area,beni, col = "green",main = "Mal/Bei vs
radius*area*perimeter")
```

```
dset$geo <- dset$perimeter*dset$radius*dset$area
modelgeo <- glm(beni ~ dset$geo, data = dset, family=binomial); summary(modelgeo)

plot(dset$texture,beni, col = "green",main = "Mal/Bei vs Texture")
modeltexture <- glm(beni ~ dset$texture, data = dset, family=binomial); summary(modeltexture)

plot(dset$smoothness,beni, col = "red",main = "Mal/Bei vs Smoothness")
modelsmoothness <- glm(beni ~ dset$smoothness, data = dset, family=binomial);
summary(modelsmoothness)

plot(dset$compactness,beni, col = "green",main = "Mal/Bei vs Compactness")
modelcompactness <- glm(beni ~ dset$compactness, data = dset, family=binomial);
summary(modelcompactness)

plot(dset$concavity,beni, col = "green",main = "Mal/Bei vs Concavity")
modelconcavity <- glm(beni ~ dset$concavity, data = dset, family=binomial);
summary(modelconcavity)

plot(dset$points,beni, col = "green",main = "Mal/Bei vs Points")
modelpoints <- glm(beni ~ dset$points, data = dset, family=binomial); summary(modelpoints)

plot(dset$symmetry,beni, col = "red",main = "Mal/Bei vs Symmetry")
modelsymmetry <- glm(beni ~ dset$symmetry, data = dset, family=binomial);
summary(modelsymmetry)

plot(dset$dimension,beni, col = "red",main = "Mal/Bei vs Dimension")
modeldimension <- glm(beni ~ dset$dimension, data = dset, family=binomial);
summary(modeldimension)



#-----------------------------------------------------
# Part 4

area <- dset$area
```

```
perimeter <- dset$perimeter
radius <- dset$perimeter

geo <- dset$geo
texture <- dset$texture
smoothness <- dset$smoothness
compactness <- dset$compactness
concavity <- dset$concavity
points <- dset$points
symmetry <- dset$symmetry
dimensions <- dset$dimension

maindset <- data.frame(beni, area, perimeter, radius, geo, texture, smoothness, compactness,
concavity, points, symmetry, dimensions)
#train_sample<-sample(1,469)
cred_train<-maindset[1:469, ]; tail(cred_train)
cred_test<- maindset[470:569, ]; head(cred_test)

model1 <- glm(beni ~ geo + texture + smoothness + compactness + concavity + points +
symmetry + dimensions, data = cred_train, family=binomial); summary(model1)
anova(model1, test = "Chisq")

model2 <- glm(beni ~ geo + texture + compactness + points , data = cred_train,
family=binomial); summary(model2)
anova(model2, test = "Chisq")


library(gmodels)
credpreds1 <- predict(model2, newdata = cred_test, type = "response")
pre <- ifelse(credpreds1>.4,1,0);
CrossTable(beni[470:569], pre[1:100])
print("accuracy is 94%" )
```