Breast Cancer Diagnostic Prediction

Introduction:

Cancer is a leading cause of death in the United States, and breast cancer specifically is the most likely to be malignant among women. With so many at risk to cancer in the country and the world, it is important to be able to detect the disease at and to detect it at an early stage. Breast cancer is the result of out of control growth among cells. Some research shows that there may be a connection between the shape of the nucleus of a cell and cancerous cells. Statistical research and analysis can help to prove this connection and contribute to the fight against cancer, both as a way to help detect it, and provide more knowledge in the search for a cure.

Problem Statement:

Using our dataset of 32 variables measuring the size and shape of cell nuclei, we attempt to create a model that will allow us to predict whether a breast cancer cell is benign or malignant.

Dataset description:

Our dataset consists of 569 observations and 32 variables. There is an ID variable, a diagnosis variable revealing if they were benign or malignant, and 30 measurement variables detailing the size and shape of the cell nuclei. The diagnosis, a categorical variable, is our response variable and the 30 measurement variables, all of which are continuous, are our potential explanatory variables for our model.

The 30 measurement variables are actually only 10 different features of the nucleus, but with 3 different measurements of each; the mean, the standard error and the 'worst' or largest (mean of the three largest values). The 10 features included are:

a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

All of these features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. It was originally collected at the University of Wisconsin in 1995, and we obtained it from the UC Irvine Machine Learning Repository

Exploratory Data Analysis:

There are 569 observations with 32 variables. The response variable in this dataset is "diagnosis" which is a categorical variable with the values of "B" or "M". The remaining variables are different measurements of the size of the tumor, smoothness of the perimeter of the tumor, compactness, texture and symmetry of the tumor, as well as variations within the tumor of those values. A glimpse (*Figure1*) of the dataset reveals that all the measurements are continuous variables, except id which represents the patient's identification number.

```
Observations: 569
Variables: 32
$ id                      <int> 842302, 842517, 84300903, 84348301, 84...
$ diagnosis               <fctr> M, M, M, M, M, M, M, M, M, M, M, M, M...
$ radius_mean             <dbl> 17.990, 20.570, 19.690, 11.420, 20.290...
$ texture_mean            <dbl> 10.38, 17.77, 21.25, 20.38, 14.34, 15....
$ perimeter_mean          <dbl> 122.80, 132.90, 130.00, 77.58, 135.10,...
$ area_mean               <dbl> 1001.0, 1326.0, 1203.0, 386.1, 1297.0,...
$ smoothness_mean         <dbl> 0.11840, 0.08474, 0.10960, 0.14250, 0....
$ compactness_mean        <dbl> 0.27760, 0.07864, 0.15990, 0.28390, 0....
$ concavity_mean          <dbl> 0.30010, 0.08690, 0.19740, 0.24140, 0....
$ concave_points_mean     <dbl> 0.14710, 0.07017, 0.12790, 0.10520, 0....
$ symmetry_mean           <dbl> 0.2419, 0.1812, 0.2069, 0.2597, 0.1809...
$ fractal_dimension_mean  <dbl> 0.07871, 0.05667, 0.05999, 0.09744, 0....
$ radius_se               <dbl> 1.0950, 0.5435, 0.7456, 0.4956, 0.7572...
$ texture_se              <dbl> 0.9053, 0.7339, 0.7869, 1.1560, 0.7813...
$ perimeter_se            <dbl> 8.589, 3.398, 4.585, 3.445, 5.438, 2.2...
$ area_se                 <dbl> 153.40, 74.08, 94.03, 27.23, 94.44, 27...
$ smoothness_se           <dbl> 0.006399, 0.005225, 0.006150, 0.009110...
$ compactness_se          <dbl> 0.049040, 0.013080, 0.040060, 0.074580...
$ concavity_se            <dbl> 0.05373, 0.01860, 0.03832, 0.05661, 0....
$ concave_points_se       <dbl> 0.015870, 0.013400, 0.020580, 0.018670...
$ symmetry_se             <dbl> 0.03003, 0.01389, 0.02250, 0.05963, 0....
$ fractal_dimension_se    <dbl> 0.006193, 0.003532, 0.004571, 0.009208...
$ radius_worst            <dbl> 25.38, 24.99, 23.57, 14.91, 22.54, 15....
$ texture_worst           <dbl> 17.33, 23.41, 25.53, 26.50, 16.67, 23....
$ perimeter_worst         <dbl> 184.60, 158.80, 152.50, 98.87, 152.20,...
$ area_worst              <dbl> 2019.0, 1956.0, 1709.0, 567.7, 1575.0,...
$ smoothness_worst        <dbl> 0.1622, 0.1238, 0.1444, 0.2098, 0.1374...
$ compactness_worst       <dbl> 0.6656, 0.1866, 0.4245, 0.8663, 0.2050...
$ concavity_worst         <dbl> 0.71190, 0.24160, 0.45040, 0.68690, 0....
$ concave_points_worst    <dbl> 0.26540, 0.18600, 0.24300, 0.25750, 0....
$ symmetry_worst          <dbl> 0.4601, 0.2750, 0.3613, 0.6638, 0.2364...
$ fractal_dimension_worst <dbl> 0.11890, 0.08902, 0.08758, 0.17300, 0....
```

Figure 1: A glimpse of the dataset

Descriptive statistics:

Number of cases with diagnosis of "B" or "M" and the mean and standard deviations of each of the numeric variables is shown below:

```
table(wdbc$diagnosis)
```

```
##
##   B   M
## 357 212
```

radius_mean	texture_mean	perimeter_mean
14.13	19.29	91.97
area_mean	smoothness_mean	compactness_mean
654.89	0.10	0.10
concavity_mean	concave_points_mean	symmetry_mean
0.09	0.05	0.18
fractal_dimension_mean	radius_se	texture_se
0.06	0.41	1.22
perimeter_se	area_se	smoothness_se
2.87	40.34	0.01
compactness_se	concavity_se	concave_points_se
0.03	0.03	0.01
symmetry_se	fractal_dimension_se	radius_worst
0.02	0.00	16.27
texture_worst	perimeter_worst	area_worst
25.68	107.26	880.58
smoothness_worst	compactness_worst	concavity_worst
0.13	0.25	0.27
concave_points_worst	symmetry_worst	fractal_dimension_worst
0.11	0.29	0.08

Figure 2: Mean of numeric variables

radius_mean	texture_mean	perimeter_mean
3.52	4.30	24.30
area_mean	smoothness_mean	compactness_mean
351.91	0.01	0.05
concavity_mean	concave_points_mean	symmetry_mean
0.08	0.04	0.03
fractal_dimension_mean	radius_se	texture_se
0.01	0.28	0.55
perimeter_se	area_se	smoothness_se
2.02	45.49	0.00
compactness_se	concavity_se	concave_points_se
0.02	0.03	0.01
symmetry_se	fractal_dimension_se	radius_worst
0.01	0.00	4.83
texture_worst	perimeter_worst	area_worst
6.15	33.60	569.36
smoothness_worst	compactness_worst	concavity_worst
0.02	0.16	0.21
concave_points_worst	symmetry_worst	fractal_dimension_worst
0.07	0.06	0.02

Figure 3: Standard deviation of numeric variables

How are the numeric variables related to one another?

The correlation plot indicates that many of the numeric variables are highly correlated. The blue colored ones indicate positive correlation and the red ones indicate negative correlations.

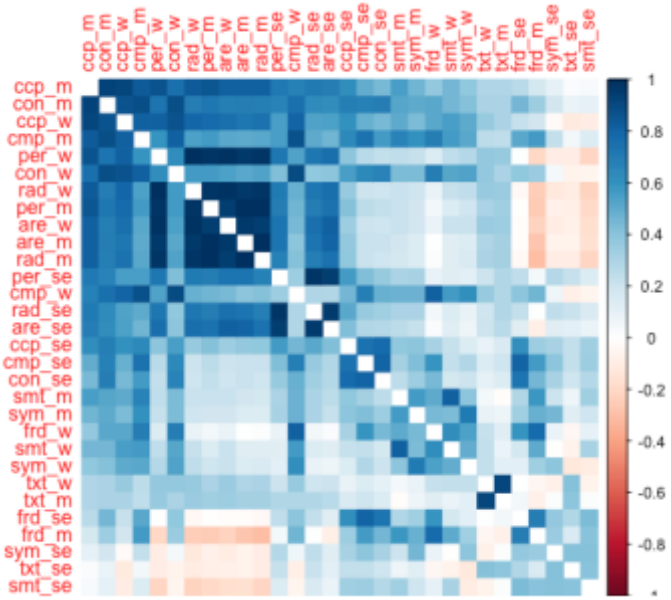

Figure 4: Correlation plot of numeric variables

Principal Components Analysis (PCA):

Why PCA?

Due to the number of variables in the model, we can try using a dimensionality reduction technique to unveil any patterns in the data. As mentioned in the Exploratory Data Analysis section, there are thirty variables that when combined can be used to model each patient's diagnosis. Using PCA we can combine our many variables into different linear combinations that each explain a part of the variance of the model. By proceeding with PCA we are assuming the linearity of the combinations of our variables within the dataset. By choosing only the linear combinations that provide a majority (>= 85%) of the co-variance, we can reduce the complexity of our model. We can then more easily see how the model works and provide meaningful graphs and representations of our complex dataset.

The first step in doing a PCA, is to ask ourselves whether the data should be scaled to unit variance. That is, to bring all the numeric variables to the same scale. Looking at the descriptive statistics (*figure 2&3*) of "area_mean" and "area_worst", we can observe that they have unusually large values for both mean and standard deviation. The units of measurements for these variables are different than the units of measurements of the other numeric variables. The effect of using variables with different scales can lead to amplified variances. This can be visually assessed by looking at the bi-plot of PC1 vs PC2, calculated from using non-scaled data (left) vs scaled data (right).

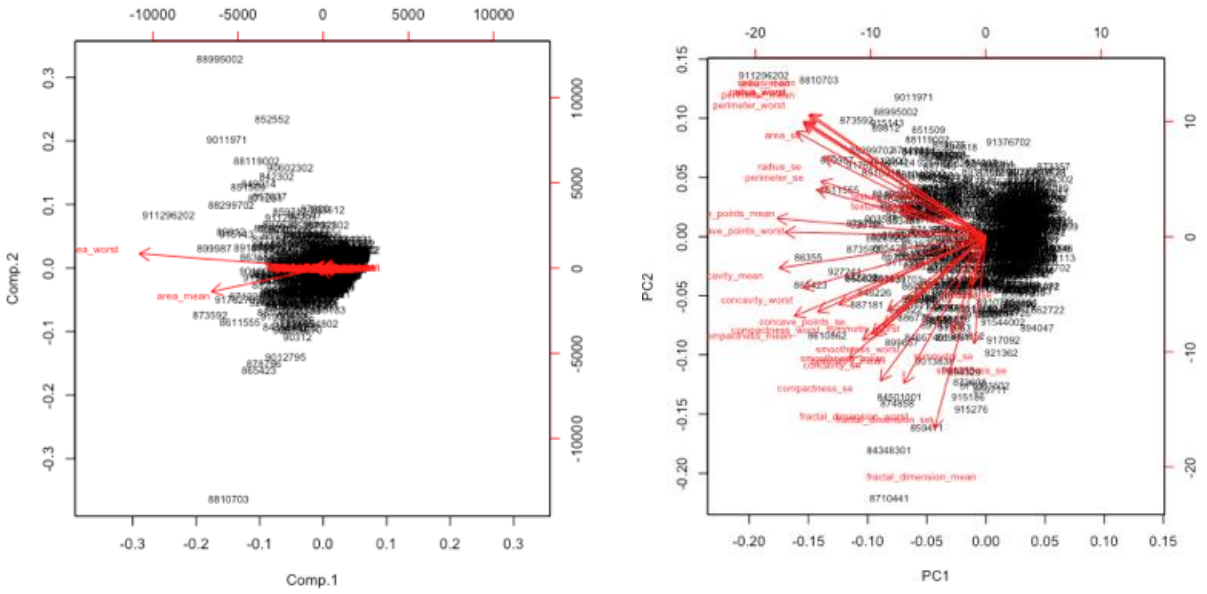

Figure 5: Bi-plot of Unscaled vs Scaled data

As we can see in figure5, 'area_worst' and 'area_mean' explain most of the variation in principal components 1 and 2. In the next section, we will illustrate how this will affect our calculations of the principal components. The decision to scale to unit variance means that we will be using a correlation matrix as opposed to using a covariance matrix.

Compare Eigen values calculated with covariance matrix vs correlation matrix:

Eigen values of PC1 and PC2 are very large compared to the rest of the components.

Table 1: Eigen values for Covariance(left) vs Correlation matrix (right)

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7
443002.67	7297.25	782.60	54.55	39.82	3.00	1.81
Comp.8	Comp.9	Comp.10	Comp.11	Comp.12	Comp.13	Comp.14
0.37	0.16	0.08	0.03	0.01	0.00	0.00
Comp.15	Comp.16	Comp.17	Comp.18	Comp.19	Comp.20	Comp.21
0.00	0.00	0.00	0.00	0.00	0.00	0.00
Comp.22	Comp.23	Comp.24	Comp.25	Comp.26	Comp.27	Comp.28
0.00	0.00	0.00	0.00	0.00	0.00	0.00
Comp.29	Comp.30					
0.00	0.00					

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9
13.28	5.69	2.82	1.98	1.65	1.21	0.68	0.48	0.42
Comp.10	Comp.11	Comp.12	Comp.13	Comp.14	Comp.15	Comp.16	Comp.17	Comp.18
0.35	0.29	0.26	0.24	0.16	0.09	0.08	0.06	0.05
Comp.19	Comp.20	Comp.21	Comp.22	Comp.23	Comp.24	Comp.25	Comp.26	Comp.27
0.05	0.03	0.03	0.03	0.02	0.02	0.02	0.01	0.01
Comp.28	Comp.29	Comp.30						
0.00	0.00	0.00						

Compare % variance explained with covariance matrix vs correlation matrix:

Majority of the variance is lumped into the first 2 PCs in the case of covariance matrix.

Table 2: % variance explained for Covariance (left) vs Correlation matrix (right)

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9
0.98	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Comp.10	Comp.11	Comp.12	Comp.13	Comp.14	Comp.15	Comp.16	Comp.17	Comp.18
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Comp.19	Comp.20	Comp.21	Comp.22	Comp.23	Comp.24	Comp.25	Comp.26	Comp.27
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Comp.28	Comp.29	Comp.30						
0.00	0.00	0.00						

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9
0.44	0.19	0.09	0.07	0.05	0.04	0.02	0.02	0.01
Comp.10	Comp.11	Comp.12	Comp.13	Comp.14	Comp.15	Comp.16	Comp.17	Comp.18
0.01	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00
Comp.19	Comp.20	Comp.21	Comp.22	Comp.23	Comp.24	Comp.25	Comp.26	Comp.27
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Comp.28	Comp.29	Comp.30						
0.00	0.00	0.00						

Compare cumulative % variance explained with covariance matrix vs correlation matrix:

Most of the variance is captured in the first 2 PC in the case of covariance matrix.

Table 3: Cumulative % variance explained for Covariance (left) vs Correlation matrix (right)

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9
0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Comp.10	Comp.11	Comp.12	Comp.13	Comp.14	Comp.15	Comp.16	Comp.17	Comp.18
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Comp.19	Comp.20	Comp.21	Comp.22	Comp.23	Comp.24	Comp.25	Comp.26	Comp.27
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Comp.28	Comp.29	Comp.30						
1.00	1.00	1.00						

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9
0.44	0.63	0.73	0.79	0.85	0.89	0.91	0.93	0.94
Comp.10	Comp.11	Comp.12	Comp.13	Comp.14	Comp.15	Comp.16	Comp.17	Comp.18
0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99
Comp.19	Comp.20	Comp.21	Comp.22	Comp.23	Comp.24	Comp.25	Comp.26	Comp.27
0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Comp.28	Comp.29	Comp.30						
1.00	1.00	1.00						

Compare the scree plots of covariance vs correlation matrix:

A steep drop in the scree-plot vs a steady drop is noted. The proportion of variance explained also shows a steady increase in the proportion of variance explained in the case of correlation matrix.

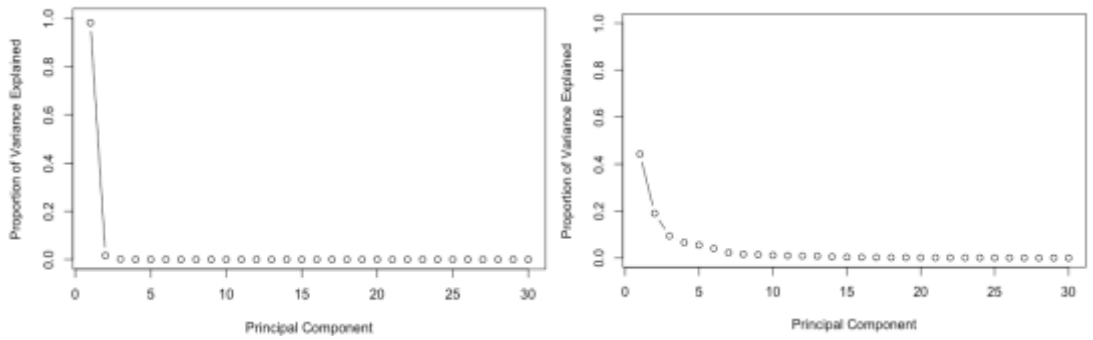

Figure 6: Scree-plot for Covariance vs Correlation matrix

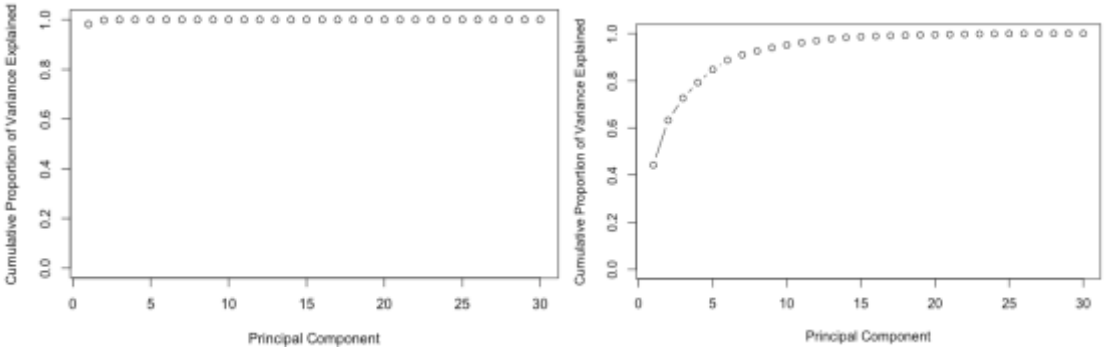

Figure 7: Scree-plot for Covariance (left) vs Correlation matrix (right)

All the above plots and tables indicate that we should scale the data to unit variance, which in turn implies that we should use a correlation matrix in our calculations of Eigen values and Eigen Vectors (aka principal components).

How many principal components to keep in the model?

As illustrated above, we will be using the correlation matrix to calculate our principal components. Our next task is to decide the number of principal components to keep. If the Eigen value of the principal component is greater than 1, then we will include that in our model. Using this criterion, we see that we can include 6 principal components (see Table 1). The first 6 PCs capture 89% of the variance. We will now take a closer look at the first 6 principal components. Plotting PC1 vs PC2, PC3, PC4, PC5, PC6, reveals a clear separation of the values.

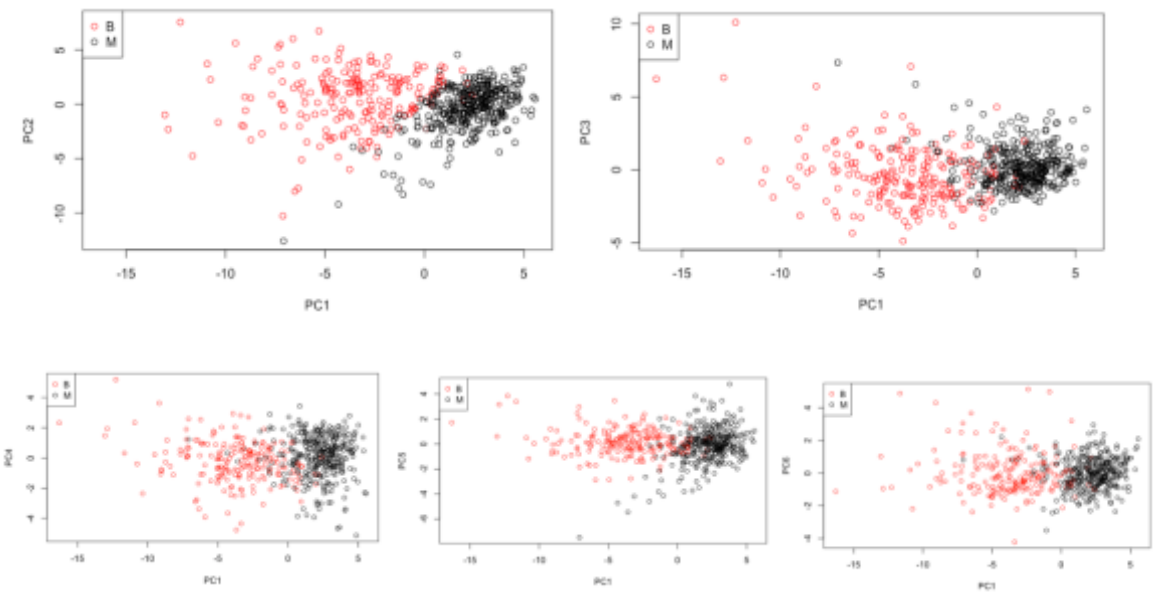

Figure 8: Scatter Plot of PC1 vs PC2-6

By using PCA we took a complex model of 30 (or more) predictors and condensed the model down to six linear combinations of the various predictors.

LDA:

From the principal component's scatter plots it is evident that there is some clustering of benign and malignant points. This suggests that we could build a linear discriminant function using these principal components. Now that we have our chosen principal components we can perform the linear discriminant analysis.

Model building and Cross-Validation:

To evaluate the effectiveness of our model in predicting the diagnosis of breast cancer, we can split the original data set into training and test data. Using the training data, we will build the model and using the test data we will make predictions. We will then compare the predictions with the original data to check the accuracy of our predictions. We will use three approaches to split and validate the data. In the first approach, we use

75% of the data as our training dataset and 25% as our test dataset. In the second approach, we use 3-fold cross validation and in the third approach we extend that to a 10-fold cross validation. For each of these approaches we will calculate the prediction accuracy by comparing the predictions with the original data.

Approach 1: 75/25 Train/Test split:

428 observations are in training dataset and 141 observations are in the test dataset. We will use the training dataset to calculate the linear discriminant function by passing it to the lda() function. A summary of the lda() function output is shown below:

```
Call:
lda(diagnosis ~ PC1 + PC2 + PC3 + PC4 + PC5 + PC6, data = wdbc.pcst.train.df)

Prior probabilities of groups:
        0         1
0.6378505 0.3621495

Group means:
        PC1        PC2        PC3        PC4         PC5        PC6
0  2.159042 -0.4137349  0.2353578  0.1706320 -0.02677891 0.01803605
1 -3.608992  0.6422433 -0.3102083 -0.2190455  0.15641364 0.04978323

Coefficients of linear discriminants:
            LD1
PC1 -0.46192354
PC2  0.17746000
PC3 -0.20135626
PC4 -0.20313361
PC5  0.13543978
PC6 -0.03737763
```

Figure 9: Summary of LDA

Using the coefficients of linear discriminants our model statement becomes

Diagnosis = – (0.46)PC1 + (0.17)PC2 – (0.20)PC3 – (0.20)PC4 + (0.13)PC5 – (0.03)PC6

Using this model, we can now predict the diagnosis for the 141 test dataset observations. The result is shown here:

```
  [1] 1 1 1 0 1 1 0 1 1 0 1 0 0 1 1 0 1 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 1 0
 [36] 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 1 1 1 0 0 1 1 0 0 1 0 0 1 1 1 1 0 0 0 1
 [71] 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1 1 1 0 0 0 0 1 1 0 1 0 0 0 0 1 0 0 0
[106] 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 1 0 1 1 0 1 0 0 0 1
[141] 1
Levels: 0 1
```

0 represents benign
1 represents malignant

We can compare the accuracy of our predictions with the original dataset and build a matrix:

```
s  0  1
0 84  5
1  0 52
```

Per this output, the model predicted 84 times that the diagnosis is 0 (benign) when the actual observation was 0 (benign) and 5 times it predicted incorrectly. Similarly, the model predicted that the diagnosis is 1 (malignant) 52 times correctly and 0 predicted incorrectly. The accuracy of this model in predicting benign tumors is 0.9438 or 94.38% accurate. The accuracy of this model in predicting malignant tumors is 1 or 100% accurate.

Benign: 94.38% accurate
Malignant: 100% accurate

Approach two: Use 3-fold cross validation:
When we split the data into training and test data set, we are essentially doing 1 out of sample test. However, this process is a little fragile. A better approach than a simple train/test split, is using multiple test sets which gives us a more precise estimate of the true out of sample predictions. Results of 3-fold cross validation:

```
     0   1
0  116   9
1    0  64

     0   1
0  112  14
1    1  63

     0   1
0  128   7
1    0  55
```

Benign: 92.16% accurate
Malignant: 99.47% accurate

Approach three: 10-fold cross validation:
Extending the idea of a 3-fold cross validation to a 10-fold cross validation to check how our model performs on out-of-sample data.

```
    0  1
0  38  0
1   0 18

    0  1
0  38  5
1   0 14

    0  1
0  37  1
1   0 19

    0  1
0  41  3
1   0 13

    0  1
0  36  2
1   0 19

    0  1
0  33  4
1   0 20

    0  1
0  34  4
1   0 19

    0  1
0  32  6
1   0 19

    0  1
0  37  1
1   1 18

    0  1
0  30  1
1   0 26
```

Benign: 93.06% accurate
Malignant: 99.47% accurate

Conclusion:

We have shown how dimensionality reduction technique like principal components analysis can be used to reduce a large number of highly correlated predictors to small set of linear combinations of those predictors. In doing so, we unveiled patterns in the data which led us to build a classification rule using linear discriminant analysis. By applying the classification rule we have constructed a diagnostic system that predicts malignant tumors at 99.47% accuracy rate and predicts benign tumors at 93.06% accuracy rate using a 10-fold cross validation plan. Although these numbers might look good, we need to ask ourselves "what is the cost of misclassification?" The cost of misclassifying someone as having cancer when they don't could cause a certain amount of emotional grief!! But the cost of misclassifying someone as not having cancer when in fact they do have cancer is obviously greater.

References:

[1] **O.L. Mangasarian, W.N. Street and W.H. Wolberg.**
Breast cancer diagnosis and prognosis via linear programming.
Operations Research, 43(4), pages 570-577, July-August 1995.

[2] **http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.names**

R code:

https://github.com/shravan-kuchkula/Classification/blob/master/BreastCancer.md

SAS code:

```
/*Read in file from website and attach variable names*/
filename webdata url "http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data";
data bcancer;
infile webdata dsd firstobs=2;
input id diagnosis $ radius_mean texture_mean perimeter_mean area_mean smoothness_mean compactness_mean concavity_mean
        concave_points_mean symmetry_mean fractal_dimension_mean radius_se texture_se perimeter_se area_se smoothness_se
        compactness_se concavity_se concave_points_se symmetry_se fractal_dimension_se radius_worst texture_worst perimeter_worst
        area_worst smoothness_worst compactness_worst concavity_worst concave_points_worst symmetry_worst fractal_dimension_worst;
put _INFILE_;
if diagnosis = "B" then diag = 0;
if diagnosis = "M" then diag = 1;
run;
```

```
/*check data*/
/*proc print data=bcancer; run;*/
/*Data summary view
proc univariate data=bcancer;
run;*/
/*Principal component analysis - decide which combinations of factors represent the data
best*/
proc princomp plots=all data=bcancer cov out=pca;
    var radius_mean texture_mean perimeter_mean area_mean smoothness_mean
compactness_mean concavity_mean
            concave_points_mean symmetry_mean fractal_dimension_mean radius_se
texture_se perimeter_se area_se smoothness_se
            compactness_se concavity_se concave_points_se symmetry_se fractal_dimension_se
radius_worst texture_worst perimeter_worst
            area_worst smoothness_worst compactness_worst concavity_worst
concave_points_worst symmetry_worst fractal_dimension_worst;
run;
/*Summary view of our new linear components*/
proc univariate data=pca;
var prin1 prin2 prin3 prin4 prin5;
histogram;
run;
/*Matrix scatterplot of the new linear components*/
proc sgscatter data=pca;
matrix prin1 prin2 prin3 prin4 prin5;
run;
/*Creating the model based on our linear components*/
proc reg data=pca;
model diag=prin1 prin2 prin3 prin4 prin5;
run;
/*
proc discrim data=pca method=normal pool=yes
            list crossvalidate;
    class diag;
    var prin1 prin2 prin3 prin4 prin5;
  run;

proc pls data=bcancer method=pcr;
          model diag = radius_mean texture_mean perimeter_mean area_mean
smoothness_mean compactness_mean concavity_mean
            concave_points_mean symmetry_mean fractal_dimension_mean radius_se
texture_se perimeter_se area_se smoothness_se
            compactness_se concavity_se concave_points_se symmetry_se fractal_dimension_se
radius_worst texture_worst perimeter_worst
```

```
        area_worst smoothness_worst compactness_worst concavity_worst
concave_points_worst symmetry_worst fractal_dimension_worst;
run;
*/
```