Automated Diagnostic Classification of Breast Cancer FNAs using Support Vector Machines

Michael Cassidy[1]
[1]SCEM, Western Sydney University, Parramatta, NSW 2150 NSW

Although breast cancer among the most treatable cancers when detected early; it is the second largest contributor to cancer-related mortality amongst females. This paper proposes the use of dimensionality reduction through PCA in combination with SVM to classify malignant and benign tumours examined with fine needle aspirates. An overall sensitivity of 97.9% level of sensitivity in diagnosis was achieved.

***Index Terms*—Machine learning, Medical diagnosis**

I. Introduction

BREAST cancer is malignant neoplasm originating in epithelial tissue (carcinoma). Approximately 300 genes are known to be involved in carcinogenesis [1].

Breast carcinoma is typically diagnosed through the use of core needle biopsy or an open surgical biopsy. Both of these procedures are invasive to the patient [2] Approximately 1 in 8 women in the United States will develop invasive breast cancer, with approximately 230000 of 290000 new cases being diagnosed as invasive. There are around 1 million biopsy procedures in the USA per year with only 20% of the cases yielding a diagnosis of (malignant) breast cancer. Open surgical biopsy requires the use of local or general anaesthetic along with the closure of the incision with sutures. Core needle procedures produce a smaller incision with an automated device or vacuum assisting the biopsy extraction. Precision in core needle biopsy for breast carcinoma is made more precise through imaging (such as ultrasound).

Fine needle aspirates (FNAs) diagnostic tecnique similar to core needle biopsy except the needle used is fine guage (23 guage, 0.81mm)[3].The needle is introduced into the vicinity of the tissue in question and negative pressure is applied to an attached syringe and a small amount of viscous fluid is aspirated during each pass of the area. Aspiration is a simple procedure and that doesn't necessitate the use of suturing or anaesthetic. Microscopy presents both time and accuracy challenges by requiring manual examination.

Visual diagnosis (Microscopy) of FNAs is reliably sensitive (over 90%) for the detection of breast cancer. However, there is high variability in inter-technician predictions. Meaning that the testing reliability is dependent on the individual performing the procedure. With this in mind, the author suggests the use of an automated expert system.

Researchers from MIT have developed a deep-learning model that has been used on patients in a clinical setting. The automated system has been successful in identifying dense breast tissue as reliably as *expert* radiologists. It is important to note that the system is as good as an expert, not an average (or below average) radiologist. The modelling of expert systems in medical diagnostics could see to improve the consistency and accuracy of critical diagnosis, perhaps not eliminating the need for experienced scientists; but perhaps a system that serves to verify diagnostic procedures and provide guidance to the inexperienced and expert alike [4]. Given the recent developments in the performace of expert systems (in AI), it would seem that development of more reliable, human-free, diagnosic tools would serve the medical profession and in turn the public. Furthermore, automated systems in medical diagnostics could increase not only the quality (reliability) of service but the outcome as it has been found that waiting times for delivery of diagnostic results in higher (unnecessary) patient anxiety [5].

This paper will be an exploration into the efficacy of Principal component Analysis and Support Vector machines being used to diagnose malignant breast carcinoma cells derived from FNAs through enquiry into the accuracy of using these techniques in a lower dimensional feature space where 95% of the variance in the input variables is accounted for.

Principal component analyis is one of the oldest techniques in Multivariate analysis. Introduced by Pearson in 1901 and later developed by Hotelling in 1933 it was largely left under utilised until the development of computers that could accomplish a cumbersome (or impossible task for a human) in higher dimensional space.

The central idea of PCA is to reduce the dimensionality of a dataset through finding a linear combination of an orthonormal basis that captures the majority of the variance in fewer dimensions. This idea will be being applied to creating a support vector machine using "The Kernel trick", i.e. making the computation of a separating hyperplane (the central idea of a support vector machine) less computationally expensive. [6]

The first instance of a support vector machine algorithm comes from a generalisation of the portrait algorithm. The basic idea is that we want to build (in higher dimensional space) a fence (hyperplane), that separates two distinct groups (classes) whilst being as far away from both groups as possible (a maximal margin) [6].

II. Dataset Background and Pre-processing

The "Breast Cancer Wisconsin (Diagnostic) dataset" [7]. Was obtained from the University of California's centre for machine learning database as comma separated value text file

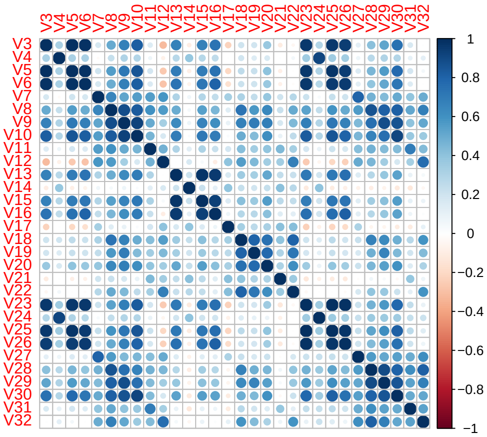


Fig. 1. Plot of correlations

(.csv) containing. The first column is the observation ID (the primary key), the second column is the diagnosis (B = benign, M = Malignant), and each subsequent row pertains to 10 measurements of 3 nuclei per patient (area, radius, perimeter, symmetry, number of concavities, size of concavities, fractal dimension, compactness, smoothness, texture). All observations in the dataset were complete (no missing values)

A. pre-processing and assessment

PCA requires the dataset to be independent (between variable samples)and identially distributed (IID), variables need to be linearly dependent, ample sampling, and to have no outliers. IID conditions were checked though conducting the Shapiro-Wilk test for normality on all 30 numeric variables under the null hypothesis that the variable in question was normal. The results of the tests (table I) indicate that there is insufficient evidence to suggest that the numeric variables in the dataset are not from a Normal distribution.

	#Reject	#not-reject
Result	0	30

TABLE I
RESULTS FROM SHAPIRO-WILK TEST FOR NORMALITY

Linear dependence was established using a plot of correlations (see 1. This plot of correlations is a matrix of correlations with other variables (including a variables correlation with itself on the main diagonal). The high proportion of blue in the plot is indicative that there is a linear relationship betweeen the input variables. Sampling adequacy was analysed through calculation of the *Kaiser-Meyer-Olkin Statistic (KOS)* for sampling adequacy for the dataset. Calculations yielded a KOS criterion of 0.83, which suggests that the sampling size is meritorious. KOS criterion is classified into 6 categories: Miserable [0,0.5), mediocre [0.5,0.6) ,middling [0.6,0.7), meritious [0.8,0.9), and marvellous [0.9,1) [8], [9].

The dataset was visualised using 30 individual histograms (see 6 in appendix). A cursory examination of the plots indicated that there were no outliers as all histograms have a clustering of parameters. Additionally, it became apparent that it would be necessary to scale the variables prior to the SVD step of PCA as the ranges of values taken by the variables varied in magnitude considerably and PCA is scale dependent [10]. Through consideration of the preceeding analysis, the dataset was deemed appropriate for dimensionality reduction via PCA.

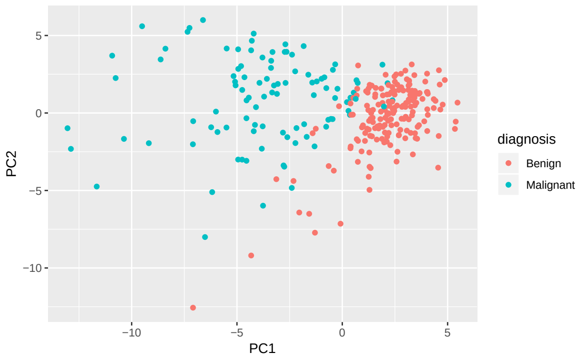


Fig. 2. Projection of training data onto PC1 and PC2

In order to be able to construct a support vector it is assumed that there is a function that maps IID random variables X to our features Y Based on some unknown conditional distribution. i.e $f : \mathbf{X} \rightarrow \mathbf{Y}$ Where X is an input matrix and Y is our decision feature. The IID nature of the variables was already assessed during an assessment of the appropriateness of PCA. The data was partitioned using 50:50-train: test split prior to model design and verification to ensure the reliability of the validation set during the testing phase of the project.

III. Data Exploration and processing

PCA was performed on the dataset to reduce the dimensionality of the data. A cut-off rule to use the first k principal components to explain more than 95% of the variance was used for modelling the support vector machine was established. The training data was plotted using the projection onto the axes the first two principal components and were separated by diagnosis (see fig. 2). The visually separable nature of fig. 2 and differing population densities for data projected onto PC1 (see fig3) indicated that using a separating hyperplane as a decision boundary may be appropriate when using more principal components as there are two distinct population densities shown for each class. Furthermore, the biplot (figure 4) shows that the first two principal components serve to contrast the malignant from the benign samples.

A. Processing

The data was processed for training a *support vector machine* in the following way:

1) Principle components were calculated using `pca.model <- prcomp(df, scale = true, subset = train)`
2) a cutoff for how many principal components to use was found by `summary(pca.model) and the components`

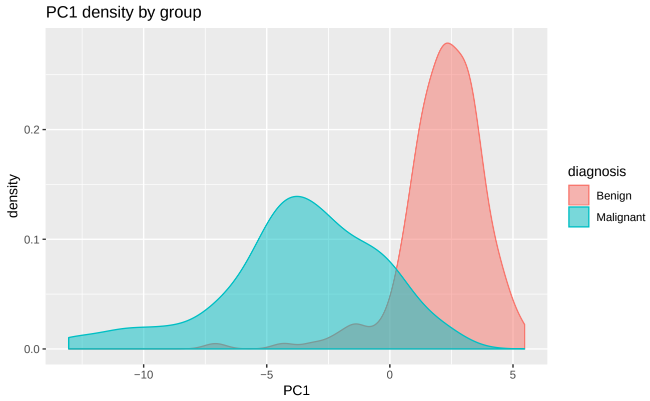


Fig. 3. Density of PC1 by Diagnosis

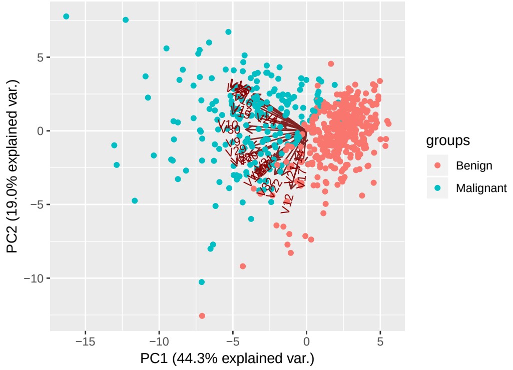


Fig. 4. Biplot of PC1 and PC2

`that had up to 95% of the cumulative variance explained`

3) A whitening matrix was constructed using : `whitener <- diag(1/pca.model$sdev[1:10],ı = 10)`
4) The *k*-principle components were used to transform the data `data.transformed <- predict(pca.model, newdata = df[train,]`
5) The transformed data was whitened to standardise the rotated data. `whitened.data <- data.tranformed %*% whitener`
6) a support linear support vector was tuned using the whitened data

The whitened, PCA transformed data was then used to train a SVM using 10-fold cross validation for a range of cost parameters.

B. *Justification of processing procedure*

PCA was used in the processing of the variables because PCA (where it is permissible to apply) is an effective method to reduce the dimensionality of a dataset. This was useful to see patterns in the data using graphs (such as 3 and 2. Without the use of PCA, seeing the separability of the diagnosis class variable would have been more difficult to discover. Additionally, PCA serves as a useful mechanism to reduce the computational expense of calculating a separating hyperplane for the data as the dimensions of the dataset are now 30x10 instead of 30x30. which will make the computation 3 times faster than the computation not using PCA due to the decreased dimensions of the matrix.

Data was whitened prior to processing to standardise the PCA transformed variables prior to training a support vector as support vector models assume that variables are within a standardised range. 10-fold cross validation was used to find a cost parameter that resulted in the model's best performance (lowest misclassification rate).

IV. Results

[ht!] PCA was conducted on variables V3-V32 with the first 10 principal components being found to explain 95% of the variance (see 5). As indicated, the first 10 of 30 principal components explained 95% of the variance. The Biplot (shown in figure 4 indicates that the contrasts beween PC1 and PC2 are linear operators that disperse malignant tumors ”away from” in their projection onto the *PC1/PC2 plane*

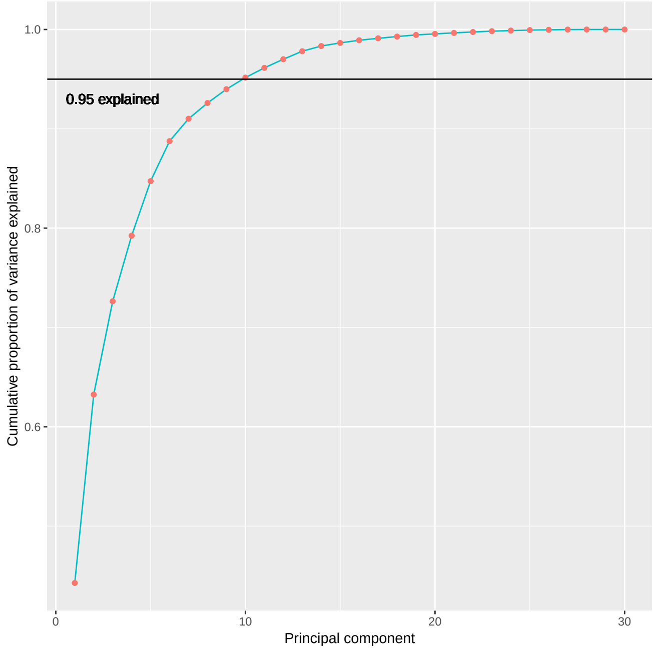


Fig. 5. Cumulative proportion of variance for Principal Components

Using the first 10 principal components the data was transformed and whitened as described in section III-A, a support vector machine with linear kernel was tuned using *10-fold cross validation* and the best model was extracted with parameters in table II.

Parameter	value
Cost	5
Gamma	0.1

TABLE II
Linear support vector parameters

The support vector with parameters listed in II was used to diagnose the testing data yielded results described by the confusion matrix in III and a misclassification rate of 0.021

	Truth	
Predict	B	M
B	170	3
M	3	109

TABLE III
CONFUSION MATRIX

V. Discussion of results

The trained support vector performed well during validation with the test set. A misclassification rate of 0.021 and an overall sensitivity of 97.9% is indicative of the promise that models of this nature have in classifying breast carcinoma in comparison to an observed manual microscopy accuracy of 90% [3]. Future enquiry could potentially explore other methods for feature engineering besides PCA as singular value decomposition is computationally expensive. Additionally, the exploration of fully automated image interpretation for this dataset may remove unaccounted for bias in the dataset as the original data was pre processed by a person prior to calculations being performed on the images. This would not have been possible when the dataset was originally created (1995)[7]. It is also important to recognise that there could potentially be much better models for classification that are outside of the scope of the task.

VI. Conclusion

The author has described and tested a system that uses machine learning techniques to diagnose breast carcinoma through the analysis of non-invasive FNAs. In comparison with manual (microscopy), it can be seen that a SVM when used with PCA is capable of outperforming experts. The results of this study is highlights the potential for expert systems to play an effective anad important role in the future of automated medical diagnostics. Future work could be driven by exploring computationally inexpensive algorithms for classification and automating cell selection during the collection procedure through modern edge detection algorithms where these systems could be implemented on handheld devices (phones with peripherals) or open source medical devices.

References

[1] R. Cammack, T. Atwood, P. Campbell, H. Parish, A. Smith, F. Vella, and J. Stirling, *Oxford Dictionary of Biochemistry and Molecular Biology*. Oxford University Press, 01 2008.

[2] "Core-needle biopsy for breast abnormalities." https://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0088567/. Accessed: 2018-10-15.

[3] W. H. Wolberg, W. N. Street, and O. Mangasarian, "Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates," *Cancer letters*, vol. 77, no. 2-3, pp. 163–171, 1994.

[4] "Automated system identifies dense tissue, a risk factor for breast cancer, in mammograms." https://www.eurekalert.org/pub_releases/2018-10/miot-asi101618.php. Accessed: 2018-10-16.

[5] K. A. Deane and L. F. Degner, "Information needs, uncertainty, and anxiety in women who had a breast biopsy with benign outcome," *Cancer Nursing*, vol. 21, no. 2, pp. 117–126, 1998.

[6] I. Steinwart, *Support vector machines*. New York: Springer, 2008.

[7] D. Dheeru and E. Karra Taniskidou, "UCI machine learning repository," 2017.

[8] H. F. Kaiser, "An index of factorial simplicity," *Psychometrika*, vol. 39, no. 1, pp. 31–36, 1974.

[9] H. F. Kaiser and J. Rice, "Little jiffy, mark iv," *Educational and psychological measurement*, vol. 34, no. 1, pp. 111–117, 1974.

[10] I. T. Jolliffe, *Principal component analysis*. New York: Springer, 2002.

Appendix
Additional Tables and figures

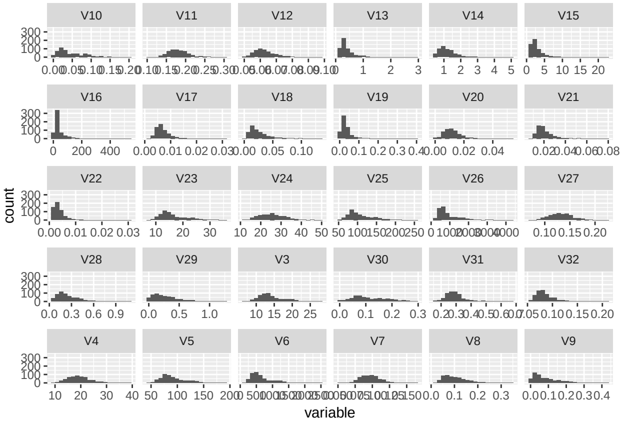


Fig. 6. Histograms of variables

Wisonsin Breast cancer dataset

Michael Cassidy

15 October 2018

Appendix

Data explortation

Normality

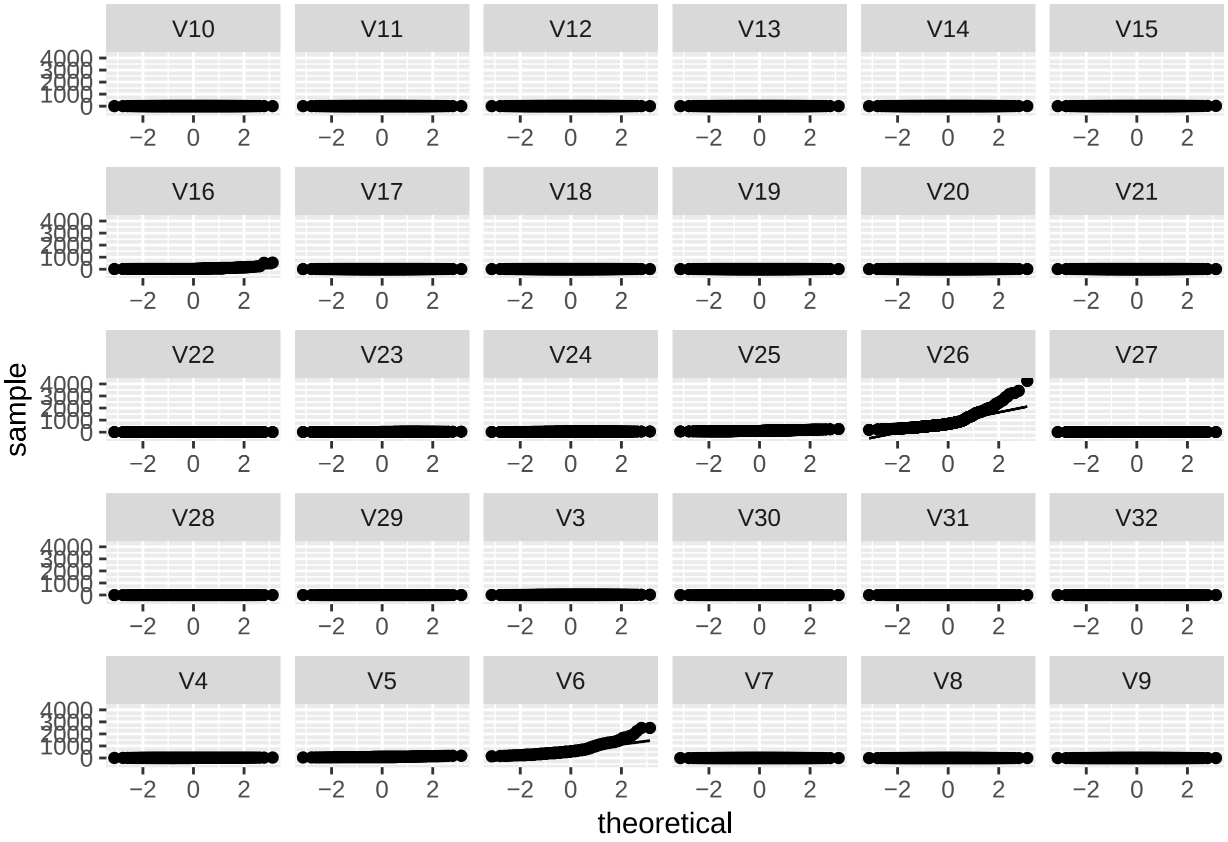
```
shapiros <- vector(length = 30)
for (i in 3:32) {
  shapiros[i-2] = shapiro.test(breast[,i])$p.value >= 0.05
}
table(shapiros)
```

```
## shapiros
## FALSE
##    30
```

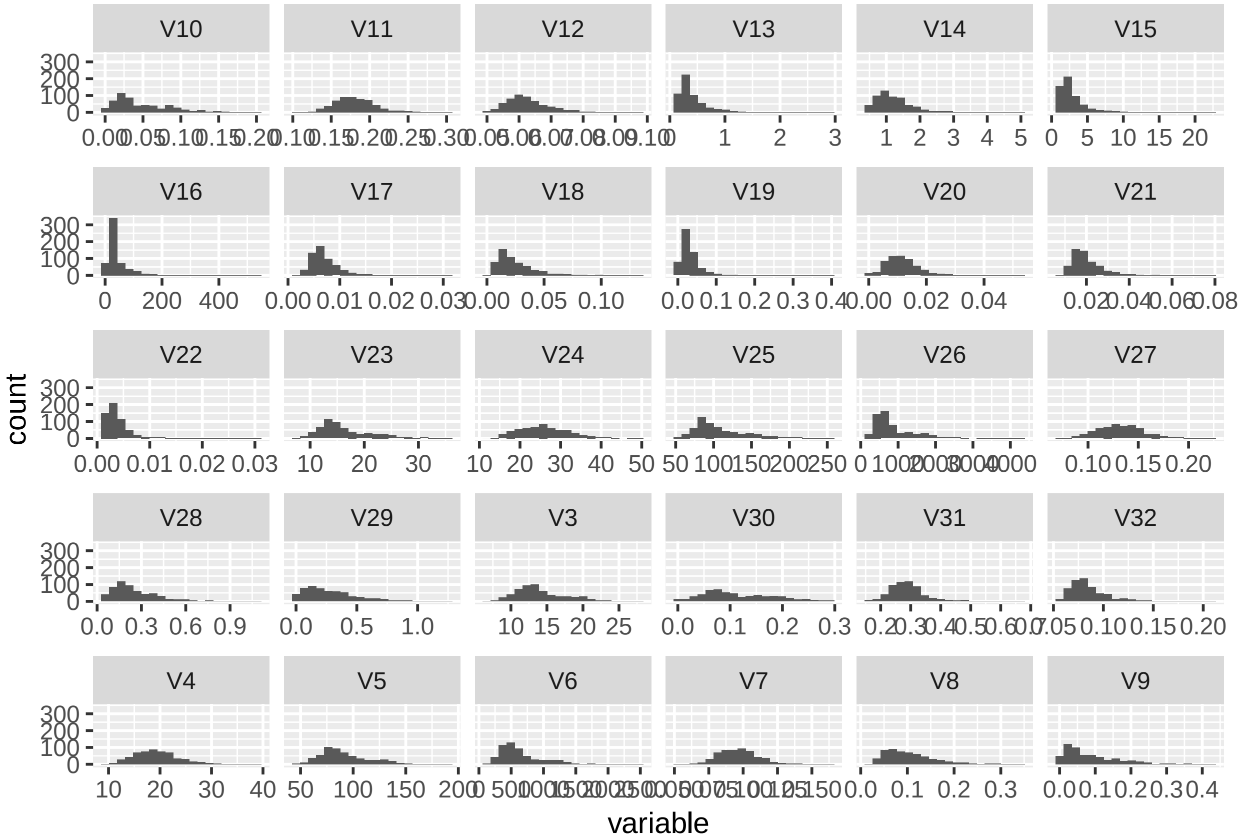
```
bartlett.test(breast[,7]~breast[,2], breast)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  breast[, 7] by breast[, 2]
## Bartlett's K-squared = 1.082, df = 1, p-value = 0.2982
```

```
breast <- breast %>% mutate(V2 = ifelse(V2 == 'B', 'Benign', 'Malignant'))
breast %>% gather(measurement, variable,3:32) %>% ggplot(aes(sample = variable)) + geom_qq() + geom_qq_
```



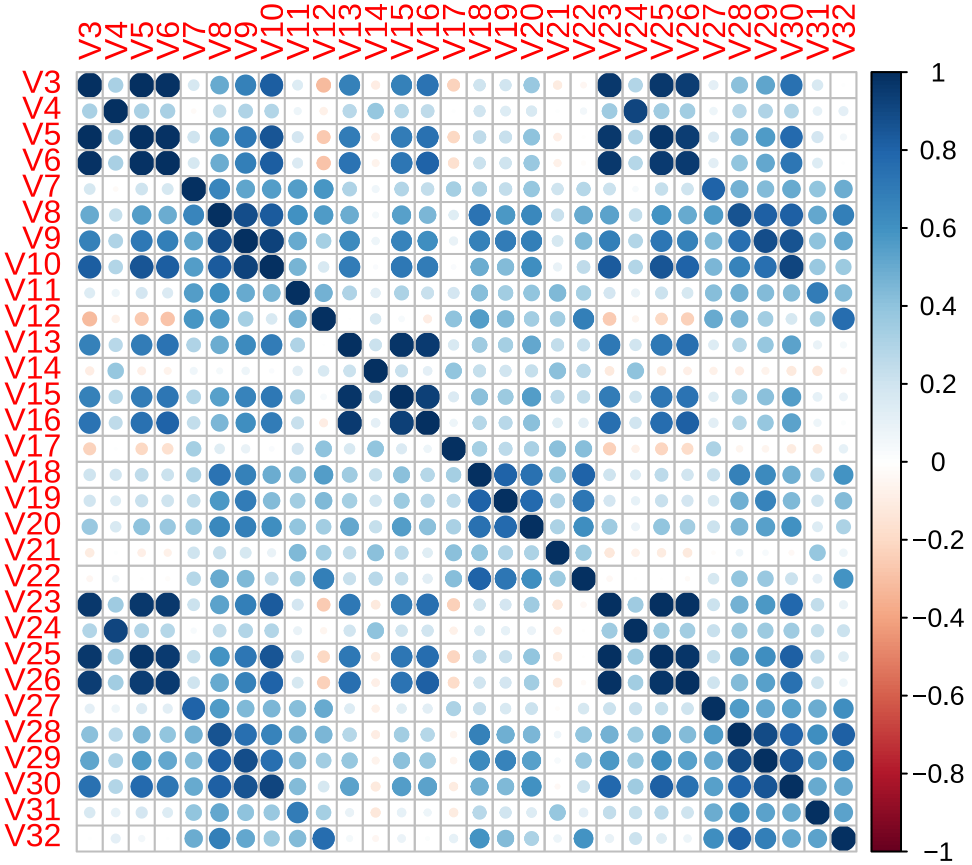
```
breast %>% gather(measurement, variable,3:32) %>% ggplot(aes(x = variable)) + geom_histogram(bins = 20)
```



```
ggsave
```

```
## function (filename, plot = last_plot(), device = NULL, path = NULL,
##     scale = 1, width = NA, height = NA, units = c("in", "cm",
##         "mm"), dpi = 300, limitsize = TRUE, ...)
## {
##     dpi <- parse_dpi(dpi)
##     dev <- plot_dev(device, filename, dpi = dpi)
##     dim <- plot_dim(c(width, height), scale = scale, units = units,
##         limitsize = limitsize)
##     if (!is.null(path)) {
##         filename <- file.path(path, filename)
##     }
##     old_dev <- grDevices::dev.cur()
##     dev(filename = filename, width = dim[1], height = dim[2],
##         ...)
##     on.exit(utils::capture.output({
##         grDevices::dev.off()
##         if (old_dev > 1) grDevices::dev.set(old_dev)
##     }))
##     grid.draw(plot)
##     invisible()
## }
## <environment: namespace:ggplot2>
```

```
breast %>% select(-V1, -V2) %>% cor() %>% corrplot()
```



There are positive correlation between the numeric variables being examined. Given that there are n=569 observations and there are 33 variables. It would be appropriate to attempt principal component analysis to reduce the number of dimensions to examine relationships further.

```
breast <- breast %>% select(id = V1, diagnosis = V2, V3:V32)
```

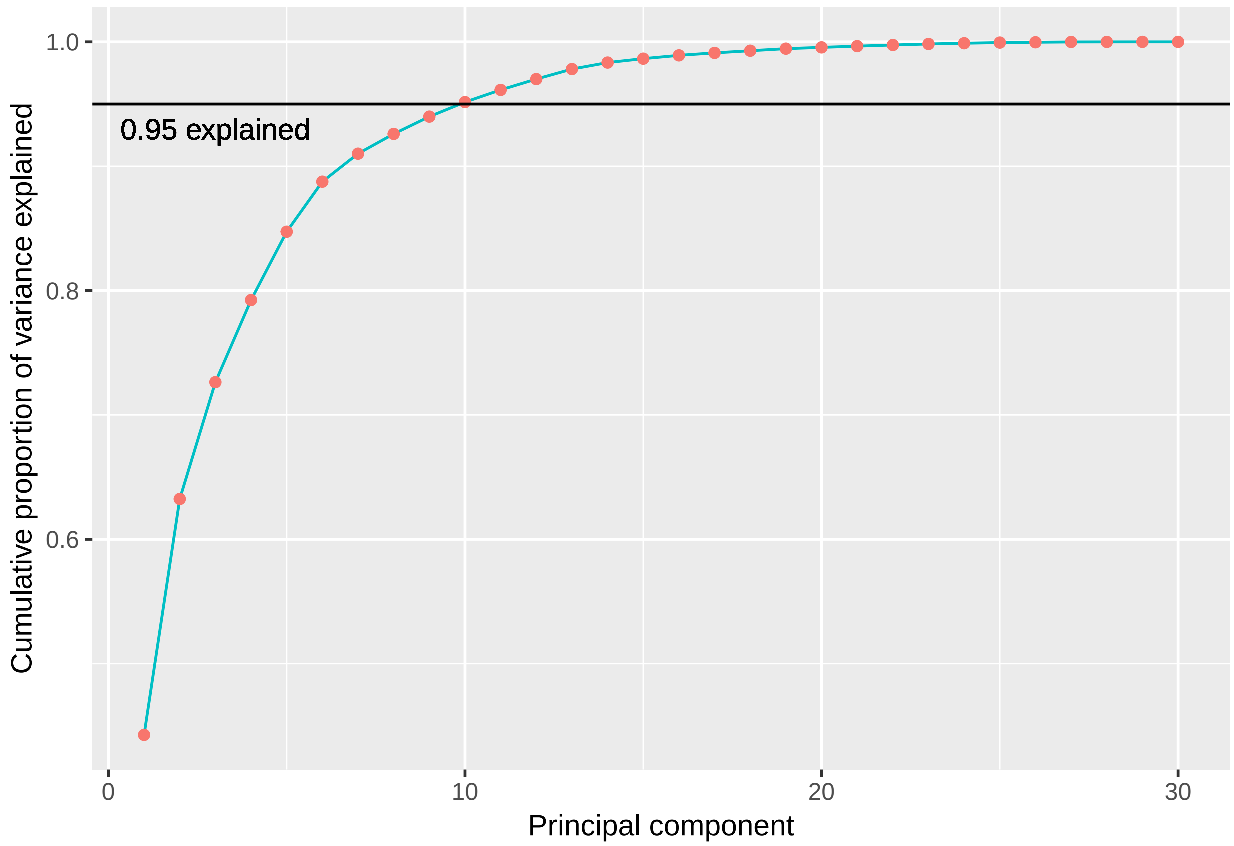
```
set.seed(123)
breast.num <- breast %>% select(-id, -diagnosis) #create a matrix of the numeric variables
num.samples <- length(breast$diagnosis)
train <- sample(num.samples, num.samples/2, replace = F)
breast.pca <- breast.num %>% prcomp(scale. = T, subset = train)
```

```
## Warning: In prcomp.default(., scale. = T, subset = train) :
##  extra argument 'subset' will be disregarded
```

```
var.prop.df <- data.frame(PC = 1:30,
                          prop.cum <- summary(breast.pca)$importance[3,1:30])
h = 0.95
var.prop.df %>%
  ggplot(aes(x = PC, y = prop.cum)) + geom_line( aes(colour = "red"), show.legend = F)+
  geom_point(aes(colour = "blue"), show.legend = F) +
  xlab("Principal component") +
  ylab("Cumulative proportion of variance explained")+
  geom_abline(intercept = 0.95, slope = 0) +
  geom_text(aes(3,0.93, label = "0.95 explained"))+

ggsave("propvar.pdf")
```

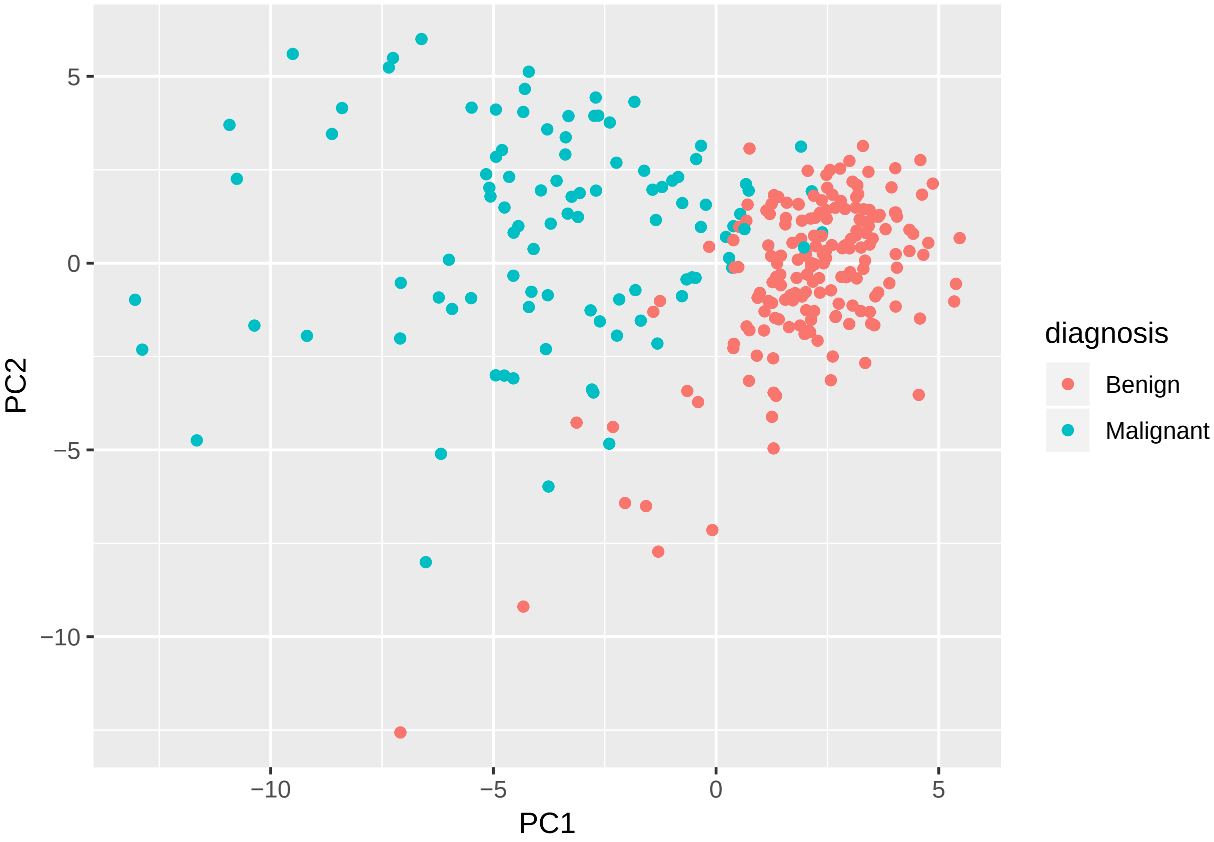
```
## Saving 6.5 x 4.5 in image
```



So we can see from the graph that we need 10 principal components to explain 95% of the variance in our dataset. Let's have a look at the first two principal components and see if we can separate the two groups.

```
# transform the data using the first two principal components and place them in a dataframe
# with their labels
pca.preds <- predict(breast.pca, newdata = breast.num[train,])[,1:2]
df.preds <- data.frame(pca.preds, diagnosis = breast$diagnosis[train])
```

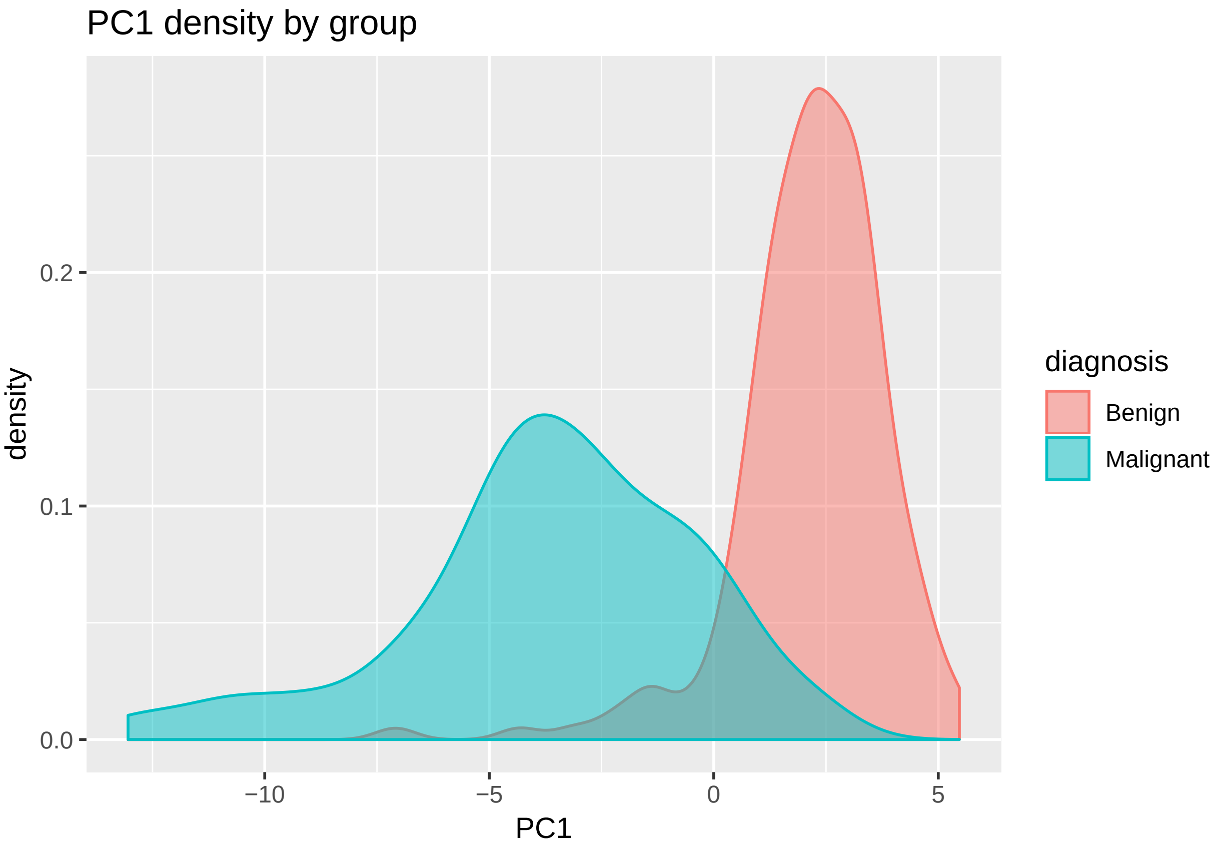
```
# Visualise
df.preds %>%
  ggplot(aes(x=PC1,y =PC2,colour = diagnosis)) + geom_point()
```



```
ggsave("PC12.pdf")
```

```
## Saving 6.5 x 4.5 in image
```

```
df.preds %>% ggplot(aes(x = PC1, fill = diagnosis, group = diagnosis))+
                      geom_density(aes(colour = diagnosis),  alpha = 0.5) + ggtitle("PC1 density by gro
```



```
ggsave("pc1density.pdf")
```

```
## Saving 6.5 x 4.5 in image
```

It looks like PC1 contrasts the diagnosis'. Let's see if we can use a linear support vector on the transformed variables to classify our diagnosis'. We will use 10-principal components however to account for more variance.

```
#Get the transformed variables
pca.preds.svm <- predict(breast.pca, newdata = breast.num[train,])[,1:10]
#Attach the diagnosis
df.preds.svm <- data.frame(pca.preds.svm, diagnosis = breast$diagnosis[train])
#Tune a suport vector with a linear kernel
tune.out=tune(svm ,diagnosis~.,data=df.preds.svm ,kernel = "linear", ranges=list(cost=c(0.001 , 0.01, 0
```

```
svm.best <- tune.out$best.model
summary(svm.best)
```

```
##
## Call:
## best.tune(method = svm, train.x = diagnosis ~ ., data = df.preds.svm,
##     ranges = list(cost = c(0.001, 0.01, 0.1, 1, 5, 10, 100)),
##     kernel = "linear")
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  5
##       gamma:  0.1
```

```
##
## Number of Support Vectors:  30
##
##  ( 16 14 )
##
##
## Number of Classes:  2
##
## Levels:
##  Benign Malignant
```

Now let’s test the model

testset <- predict(breast.pca, newdata = breast.num[-train,])[,1:10] svm.best.pred <-predict(svm.best, newdata = testset

```
testset <- predict(breast.pca, newdata = breast.num[-train,])[,1:10]
svm.best.pred <-predict(svm.best, newdata = testset)
misclassification.rate <- mean(svm.best.pred != breast[-train, "diagnosis"])
misclassification.rate
```

```
## [1] 0.02105263
```

OK so what I want to do is:

-choose k support vectors of transformed data -train a model -use the model to predict on the transformed validation set

```
misclas.svm <- vector(length = 30, mode = "numeric")
whitener = diag(1/breast.pca$sdev, nrow = 30)
for (i in 2:30) {
  #transform with pca and pick the first i columns for training and testing data
  trainset.trans <- as.matrix(predict(breast.pca, newdata = breast.num[train,])[,1:i]) %*%
    whitener[1:i,1:i]
  df.trainset.svm <- data.frame(trainset.trans, diagnosis = breast$diagnosis[train])

  testset.trans <- (predict(breast.pca, newdata = breast.num[-train,])[,1:i]) %*%
    whitener[1:i,1:i]
  #Train a svm model
  tune.svm <- tune(svm, diagnosis~.,data=df.trainset.svm ,kernel = "linear",
               ranges=list(cost=c(0.001 , 0.01, 0.1, 1,5,10,100) ))
  #precict the diagnosis on trainset.trans using the best model
  best.svm.model <- tune.svm$best.model
  svm.best.pred <- predict(best.svm.model, newdata = trainset.trans)
  misclas.svm[i] <- mean(svm.best.pred != breast[train, "diagnosis"])
}
misclas.svm = misclas.svm[-1] #remove the front uncalculated value
misclas.svm.labels = 2:30
misclas.df  = data.frame(misclas.svm.labels, misclas.svm)
```

```
trainset.trans <- as.matrix(predict(breast.pca, newdata = breast.num[train,])[,1:10]) %*%
    whitener[1:10,1:10]
  df.trainset.svm <- data.frame(trainset.trans, diagnosis = breast$diagnosis[train])

  testset.trans <- (predict(breast.pca, newdata = breast.num[-train,])[,1:10]) %*%
    whitener[1:10,1:10]
  #Train a svm model
  tune.svm <- tune(svm, diagnosis~.,data=df.trainset.svm ,kernel = "linear",
               ranges=list(cost=c(0.001 , 0.01, 0.1, 1,5,10,100) ))
  #precict the diagnosis on trainset.trans using the best model
  best.svm.model <- tune.svm$best.model
  svm.best.pred <- predict(best.svm.model, newdata = testset.trans)
  misclassification.rate <- mean(svm.best.pred != breast[-train, "diagnosis"])
  misclassification.rate
```

```
## [1] 0.02105263
```

```
table(predict = svm.best.pred , truth= breast[-train, "diagnosis"] )
```

```
##             truth
## predict     Benign Malignant
##   Benign       170         3
##   Malignant      3       109
```

```
breast.pca$rotation[1:10,1:2]
```

```
##              PC1         PC2
## V3  -0.21890244  0.23385713
## V4  -0.10372458  0.05970609
## V5  -0.22753729  0.21518136
## V6  -0.22099499  0.23107671
## V7  -0.14258969 -0.18611302
## V8  -0.23928535 -0.15189161
## V9  -0.25840048 -0.06016536
## V10 -0.26085376  0.03476750
## V11 -0.13816696 -0.19034877
## V12 -0.06436335 -0.36657547
```

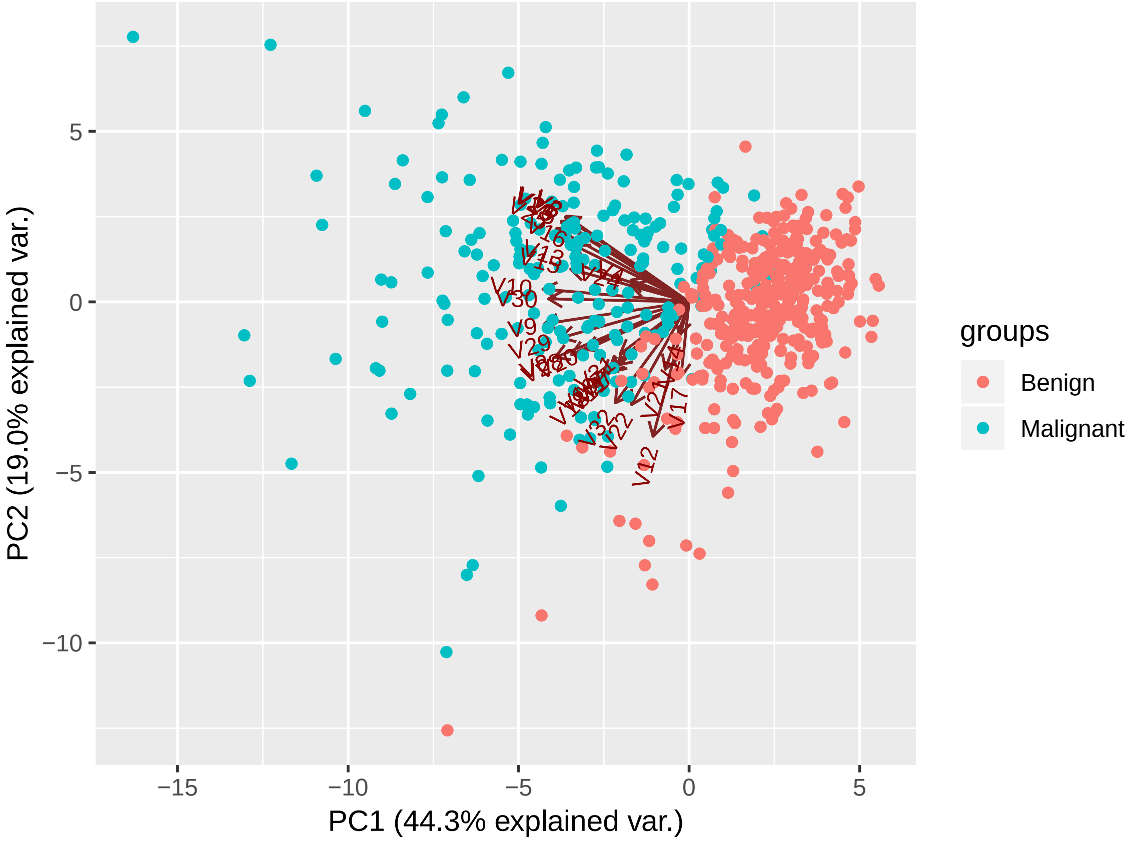
```
breast.pca$rotation[11:20,1:2]
```

```
##              PC1         PC2
## V13 -0.20597878  0.10555215
## V14 -0.01742803 -0.08997968
## V15 -0.21132592  0.08945723
## V16 -0.20286964  0.15229263
## V17 -0.01453145 -0.20443045
## V18 -0.17039345 -0.23271590
## V19 -0.15358979 -0.19720728
## V20 -0.18341740 -0.13032156
## V21 -0.04249842 -0.18384800
## V22 -0.10256832 -0.28009203
```

```
breast.pca$rotation[21:30,1:2]
```

```
##             PC1          PC2
## V23 -0.2279966  0.219866379
## V24 -0.1044693  0.045467298
## V25 -0.2366397  0.199878428
## V26 -0.2248705  0.219351858
## V27 -0.1279526 -0.172304352
## V28 -0.2100959 -0.143593173
## V29 -0.2287675 -0.097964114
## V30 -0.2508860  0.008257235
## V31 -0.1229046 -0.141883349
## V32 -0.1317839 -0.275339469
```

```
ggbiplot(breast.pca, obs.scale = 1, var.scale = 1, groups = breast[,2])
```



```
ggsave("biplot.pdf")
```

```
## Saving 6.5 x 4.5 in image
```