Early Detection of Breast Cancer

An analysis of malignant and non-cancerous X-ray images

STUDENT X

The University of Oklahoma
ISE 5970

Executive Summary

The project aimed at detecting patients with breast cancer in its earliest stages by analyzing X-ray images that were suspected to possibly contain tumors. The dataset and project objectives come from the 2008 challenge by Knowledge Discovery and Data Mining (KDD). Approximately 1.3 million women are diagnosed annually with breast cancer and about one out of every three women who are diagnosed lose their lives to cancer. However, early detection of breast cancer is instrumental in reducing this number. If a diagnosis can be provided early enough, there is a much higher chance of survival.

The provided data by KDD contained information about 1,712 patients, 118 of which had breast cancer. Each patient had multiple images of cancer, all of which had suspicious regions in the images. The team selected a random sample of 40% of the normal patients and 40% of the patients with cancer to create a testing data set. The team then selected all of the images in the remaining 60% to create a training data set along with a random sample of images without cancer that were equal in proportion to the number of cancerous images. This allowed the training data set to be equally distributed among cancerous and non-cancerous images.

The modeling process consisted of creating both decision trees and logistic models. From this process, the most preferred model was a logistic model found in Figure 7-1 (page 11) called "fitvi", which was created based off of the variables found to be important using a CART decision tree variable importance study. The model was chosen because it was the most successful in detecting images with cancer, with the highest true positive rate of 94.22% for images, and missing zero people with cancer. A cutoff of 0.15 is chosen and the model flags every cancerous image, missing zero images and zero people.

Based on this modeling and analysis, the team recommends using this predictive model, fitvi, to predict if each image has cancer. If a patient has one or more images that are predicted to have cancer, their file should be forwarded on to a radiologist who specializes in breast cancer detection in order to provide a more accurate diagnosis. The team is confident in their findings but due to the importance of early detection of breast cancer, the team recommends securing a secondary dataset for greater validation.

Problem Description

Approximately 1.3 million women are diagnosed annually with breast cancer, a disease where cancerous (malignant) cells form in breast tissue.[1] For women, breast cancer is the second most deadly cancer with about 465,000 deaths occurring each year from this cause.[2] The tasked problem comes from a modified version 2008 Knowledge Discovery and Data Mining (KDD) cup that aims to determine the presence of early-stage breast cancer in patients based on X-ray images. The American Cancer Society urges the importance of detecting breast cancer early: the size and spread of the malignant lesions present when detected are the most important factors when predicting the prognosis (expected outlook) of the patient.[3] With an early diagnosis, a patients' likelihood of long-term survival increases.[4]

The provided database provided contains patient information about 1,712 patients, 118 of which are malignant patients and all of which have suspicious regions in their X-ray images. Every image taken of the patient is included in their file with information about 117 feature characteristics as well as information on the X and Y locations of the lesion and nipple, the left/right breast, and the type of machine used to generate the image. In this data set, two machine types were used: Cranio-Caudal View (CC) and Mediolateral-oblique (MLO). An X-ray image from a CC machine shows the majority of glandular tissue, fatty tissue, and the edge of the chest wall as well as the nipple.[5] An MLO machine takes images at an angle showing the main area of the breast including glandular and fatty tissue.[6] The MLO images show a larger area than CC images, but both views help provide a more complete picture of the breast tissue.

The team's task was to create some predictive model based on the images' characteristics that will predict if an image shows cancer. If a patient has one or more images that are predicted to have cancer, the patient is flagged and sent to a radiologist for a more comprehensive diagnosis. The goal of the model is to not overlook a single patient without cancer. The model will most likely be very cautionary because of this. Due to the nature of cancer, a false positive for cancer is much more acceptable than a false negative. If the model produces a large number of false positives, the worst-case scenario is that the radiologist has to examine many false positive X-ray images. This model style would still filter out some of the true negatives, resulting in an overall reduced workload. Alternatively, if the model produces a large number of false negatives, the worst-case scenario is that patients go undiagnosed and will not receive treatment as early as possible that could lead to a lesser chance of survival. Based on these two scenarios, it is clear that a model producing many false positives is greatly preferred over a model producing many false negatives.

Because of the high importance of detecting breast cancer early, there have been many computer models and algorithms created that attempt to help solve this problem. Megan

[1] kdd.org, "KDD Cup 2008: Breast Cancer"
[2] kdd.org, "KDD Cup 2008: Breast Cancer"
[3] cancer.org, "The Importance of Finding Breast Cancer Early"
[4] Mangasarian, Street, and Wolberg, "Breast Cancer Diagnosis and Prognosis via Linear Programming"
[5] Stephan, breastcancer.about.com "Mammogram Views for Routine Diagnostic Screening"
[6] Stephan, breastcancer.about.com "Mammogram Views for Routine Diagnostic Screening"

Howard from St. Lawrence University used classification and regression trees (CART) as well as a logistic regression model to predict breast cancer in patients from the Breast Cancer Wisconsin (Diagnostic) Data Set which is known in R as "biopsy" from the library "MASS."[7,8] Mangasarian, Street, and Wolberg chose to attempt predicting breast cancer by linear programming methods with using the same "biopsy" data set as Howard.[9] We chose to use these techniques as well as others to perform our analysis, because the effectiveness of the techniques is highly dependent on the data ("biopsy" provides information at the cell-level instead of image-level) and the quality of the model of each technique directly relates to the model's effectiveness.

Exploratory Data Analysis

We spent a significant portion of our time on the project formatting and exploring the provided data. We first began by working to truly understand what each variable meant beyond the 117 random features. We found that each row of the dataset was information about a specific image and has a unique identifier known as the ImageID and the variable GroundTruth has a value of 1 if cancer is present in the image and a value of 0 if there are no malignant lesions shown. Each row in the data also has a StudyID that refers to each patient's identification number. Patients' have multiple images in the dataset so multiple rows have the same StudyIDs. In order to keep track of the data as well as validate the dataset did indeed have the claimed 118 malignant patients and 1,594 non-cancerous patients, we created a new column called ActualTruth. If the person had cancer, meaning they had more or one images that had a GroundTruth of 1, they received a value of true for their ActualTruth and if the person did not have cancer, they received a value of false for all their associated images.

This data was then split into two sets: a training set and a testing set. In order to split the data in a way that kept the presence of cancer representative, the training set was created by randomly selecting 60% of the patients with cancer and 60% of the patients without cancer. The remaining random 40% of each category were then combined into the test data set. The team then attempted to use these datasets to analyze the problem. However, the number of images without cancer far outweighed the images with cancer, making it challenging to create an accurate predictive model. (The model could guess "no cancer" every time and only be wrong 0.6% of the time). In order to create a train dataset that the model could use, the team selected all of the images that have cancer (398 images) and then randomly added a sample of 398 non-cancerous images.

The first step in exploring the data was a correlative analysis. A full correlative map was first created based on the scaled 117 features. While this map was too small to decipher, the team noted that there were some strong correlations located near the 1:1 line shown as clusters of dark blue dots together. Therefore smaller correlation plots were created for 20 features at a

[7] Howard, "Classification Trees and Predicting Breast Cancer," St. Lawrence University, 2010
[8] cran.r-project.org, "Package 'MASS'"
[9] Mangasarian, Street, and Wolberg, "Breast Cancer Diagnosis and Prognosis via Linear Programming"

time as shown in Figure 1-1 by the code in Figure 1-2. After greater inspection of these plots, there were small clusters of darker blue circles for features next to each other. One possible explanation for this could be that the features are related to the pixel locations. If this is a case, then it's possible that tumors often go beyond the boundaries of a single pixel; meaning, pixels next to each other often are highly correlated. A strong positive correlation shows that the pixels near each other correspond to the presence of a tumor and the strong negative correlations shows that the boundary of the tumor ends.

```
scalebesttrain<-scale(besttrain[,1:117])
par(mfrow=c(2,3))
corrplot(cor(scalebesttrain[,1:20]))
corrplot(cor(scalebesttrain[,21:40]))
scalebesttrain<-scale(besttrain[,1:117])
par(mfrow=c(2,3))
corrplot(cor(scalebesttrain[,1:20]))
corrplot(cor(scalebesttrain[,21:40]))
corrplot(cor(scalebesttrain[,41:60]))
corrplot(cor(scalebesttrain[,61:80]))
corrplot(cor(scalebesttrain[,81:100]))
corrplot(cor(scalebesttrain[,101:117]))
```

Figure 1-1. *Code used to create correlation maps.*

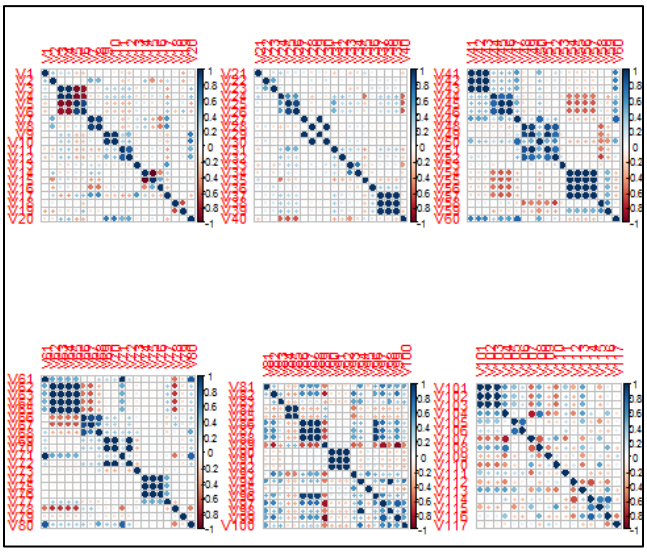

Figure 1-2. *Correlation heat maps of features.*

The next step the team took in the exploratory analysis was to look at the x-y locations of each suspicious region and its ground truth to see if certain locations were more prominent to have cancer, based on the code in figure 2-1 that utilizes the package ggplot2, a graph of each image's suspicious region location according to the x-y units is shown in figure 2-2. Color is then used to depict if the image is malignant (1=blue) or is normal (0=red).

```
qplot(xlocation,ylocation, data=besttrain,
colour=as.factor(GroundTruth), main="Location of Candidates")
```

Figure 2-1. *Code to create plot of candidates' locations.*

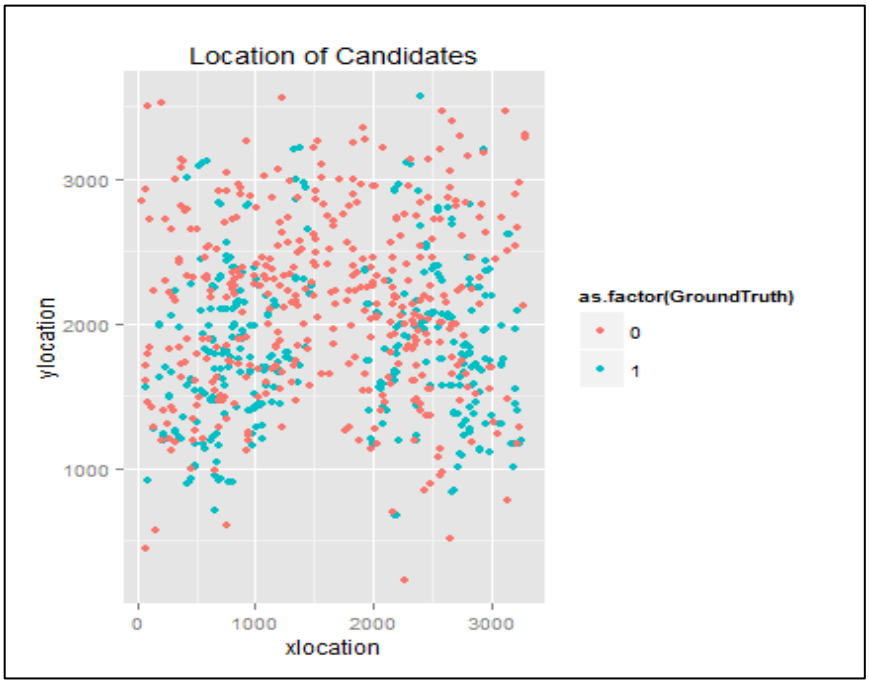

Figure 2-2. *Plot of location of Candidates.*

The graph of the location of the candidate based upon its x and y location shows that the location of the candidate caught by the machine does not predict if it is noncancerous or malignant. This may be because different images are at different scales or because the factors are truly unrelated.

To further investigate the data, a principal component analysis was also performed. Figure 3-2 shows the line plot of the first 10 principle components and their associated variances that they explain. While an elbow may not be entirely distinct, it is clear that the first five PCAs explain a large portion of the variance of the data. In order to see visually what the first two PCA rotations on the data would look like, the package ggbiplot was used to create the graph in Figure 3-3. While the axis of this graph are challenging to read since all 117 features are included, the circular groupings show that with the first two PCA rotations, a number of the images are clustered near to each other. However, many sections also have interspersed images plotted (red and blue dots very close to each other) and the circles look to be overlapping, making it difficult to still distinguish between the images' ground truth (presence of cancer) based solely on two PCAs rotations.

```
plot(prcomp(besttrain[,c(1:117)], scale. = TRUE), type = "l",
            main="Features of Images")
library(devtools)
library(ggbiplot)
```

```
pcafeatures <- prcomp(besttrain[,c(1:117)], scale. = TRUE)
summary(pcafeatures)
plot(prcomp(besttrain[,c(1:117)], scale. = TRUE), type = "l",
main="Features of Images")
class(besttrain$GroundTruth)
print(ggbiplot(pcafeatures, obs.scale = 1, var.scale = 1, groups =
as.factor(besttrain$GroundTruth), ellipse = TRUE, circle = TRUE))
```

Figure 3-1. *Code used to create Figure X82.*

Figure 3-2. *Line plot of the first 10 PCAs.*

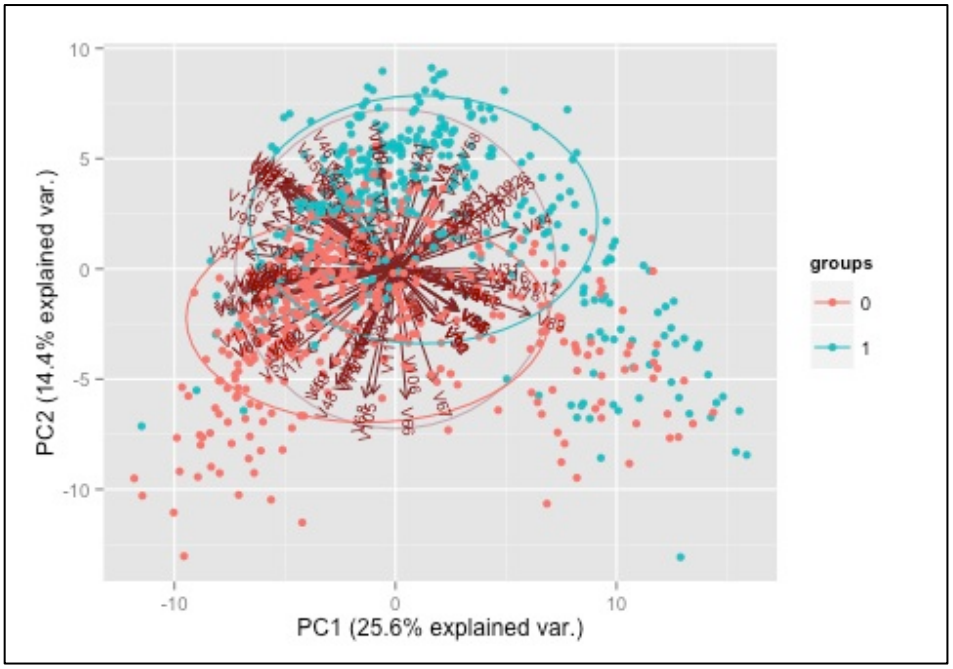

Figure 3-3. *Plot of images based on PC1 and PC2 rotations; 0=malignant image.*

Once a basic understanding of the variables was obtained as well as an understanding of the size of the data, a plan for analysis was created.

Analysis Plan

The analysis of the problem was broken down to two main modeling approaches, classification trees and logistic modeling.

A classification tree was created based on the 117 features. The tree was then pruned with the prune function in order to optimize the tree. Both trees' variable importance were then used in the creation of various logistic models. The variables' importance from each tree was plotted and a natural cutoff was selected. Then a logistic model was created with these variables and iterated on so that all of the variables included in the model had significance.

A logistic model was also created based on all 117 features. The models then went through stepwise optimization for both the Akaike information criterion (AIC) and Bayesian information criterion (BIC) criterion. From these optimal models, the team continued to iterate as well, working to remove variables with lesser significance without leading to large reductions in the overall performance of the model.

In order to determine the effectiveness of the predictive model created, the model was used to predict the presence of cancer in the images of the train data. Ideally, no person with cancer should be overlooked (predicted to not have cancer). However, it is possible that an image showing cancer could be incorrectly predicted by the model but the model overall doesn't miss flagging a person with cancer. This is because there are usually multiple images that show the cancer in a person's file and when a person has a single image that is predicted to have cancer, they are recommended to consult with a radiologist.

Other factors used to validate the model included the area of the curve (AUC) based on the ROC curve when predicting the presence of cancer in images. Additionally, the D-statistic was used to compare logistic models. The D-statistic shows the separation in means of the resulted predictions. A greater separation (a D-statistic value closer to 1) is preferred because it shows that the model can more easily decipher between the two states of the target variable. The code used to create these validation metrics is shown in Figure 4 where the name of the model (highlighted) was manipulated each time so that the output was associated with each model.

```
#Predict test values
test$predfit <- predict(fit, newdata=test, type="response")
#D-Statistic
test.1<- test[test$GroundTruth==1,]
test.0<- test[test$GroundTruth==0,]
mean(test.1$predfit)-mean(test.0$predfit)
test$predclass<-predict(fit, data=test)
#people missed---
predResults<-aggregate(predclass ~ StudyID, test, sum)
# reduce data to one obs per StudyID -- (e.g. take max of ActualTruth per
StudyID); save in dataframe
ActTruth<-aggregate(ActualTruth ~ StudyID, test, max)
#double check: total number of people in test with ActualTruth == 1
sum(ActTruth$ActualTruth)
```

```
#Then see if these unique study ID's (people) have an actual truth of "1"
#(meaning they actually have cancer and we "missed" catching them)
#merge the results above to one data frame
StudyResults<-merge(predResults, ActTruth, by="StudyID")
table(test$GroundTruth,test$predclass>=0.5)#cutoff to compare alternatives
predResults<-aggregate(predclass ~ StudyID, test, sum)
sum(predResults$predclass>0)
ActTruth<-aggregate(ActualTruth ~ StudyID, test, max)
sum(ActTruth$ActualTruth)  #total number of people in test with
ActualTruth == 1> StudyResults<-merge(predResults, ActTruth, by="StudyID")
table(StudyResults$predclass,StudyResults$ActualTruth) #The number of
people in row 1, column 1 shows the number of people we "missed"
#ROC Curve
library(Deducer)
rocplot(fit)
```

Figure 4. *Code used in validation metrics; highlighted portions were changed for each model.*

Results & Validation of Analysis

Classification Tree

To begin our analysis, the team created a decision tree based on the different variables predicting the ground truth for an image. The output of the decision tree is a factor that indicates the predicted value of whether or not an image should be flagged as cancerous (value of 1) or noncancerous (value of 0). Using the decision tree analysis, the variables predictive power was further analyzed using the CART variable importance with the code in Figure 5-1. The resulting plot is shown in Figure 5-2, where the slight elbow indicates that there is a cutoff around 10 where the most significant variables can predict the ground truth of an image. The resulting highest numbers and rank of the variables are shown in Table 1 (Appendix). These variables were later used to calculate a logistic model (fitvi). Table 1 shows the full printout of variables' importance.

```
fit<- rpart(data=besttrain, GroundTruth ~. -ActualTruth -ImageFindingID
            - StudyFindingID -ImageID -StudyID)
barplot(fit$variable.importance, main = "CART Variable Importance")
```

Figure 5-1. *Code to create the tree and plot.*

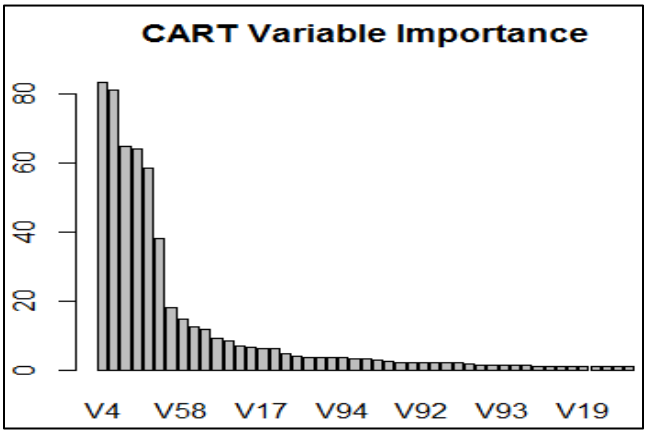

Figure 5-2. *Bar chart of decreasing variable importance based on CART.*

Pruned Classification Tree

Because the classification trees are highly volatile and the resulting tree above was difficult to read, it was necessary to create a pruned tree. The pruned tree was created with the code in Figure 6-1 and is shown in Figure 6-2.

```
library(rpart)
fitDT<-rpart(data=besttrain, as.factor(GroundTruth) ~. -ActualTruth
              - ImageFindingID -StudyFindingID -ImageID -StudyID
              -LeftBreast -MLO -xlocation -ylocation -xnipplelocation
              -ynipplelocation, control=rpart.control(minsplit=10,cp=0))
pfit<-prune(fitDT, cp=fitDT$cptable[which.min(fitDT$cptable[,"xerror"]),
            "CP"])
plot(pfit, uniform = TRUE, main = "Pruned Tree")
text(pfit, use.n=TRUE, all=TRUE, cex=.6)
```

Figure 6-1. *Code to create, prune, and plot a classification tree*.

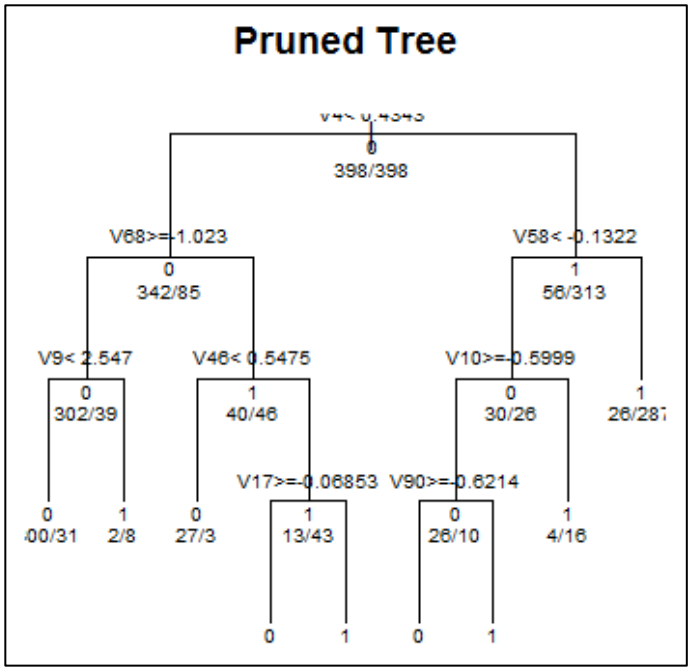

Figure 6-2. *Pruned Classification Tree for Ground Truth Values.*

From this decision tree, the team saved the predictive powers of the tree as the variable predDT to be used for further analysis later on to create the model fitD.

Logistic Model with classification tree's important variables (fitvi)

From the classification tree, the ten most important variables (the variables that were closest to the top of the tree) were used to create a logistic model. The code used to create this model is shown in Figure 7-1 and the resulting ROC curve is shown in Figure 7-2 (Appendix).

```
fitvi<-glm(data=besttrain, GroundTruth~ V4 + V3 + V6 + V5 + V14+ V21
           + V68 + V58 + V10 + V66)
```

Figure 7-1. *Code creating the model fitvi.*

Logistic Model with pruned tree variables (fitD)

There were seven variables used in the pruned decision tree. These seven variables were used to created a logistic model called fitD with the code shown in Figure 8-1. The resulting ROC curve is shown in Figure 8-2 (Appendix).

```
fitD<-glm(data=besttrain, family="binomial", GroundTruth~V4 + V68 + V58
          + V9 + V46 + V10 + V17)
```

Figure 8-1. *Code creating model fitD.*

Logistic Model with all variables (fit)

The first natural step in creating a logistic model from scratch was to create a model that incorporated all of the 117 feature variables. This model was created to be used as a stepping stone to other models. The model was known as "fit," and was created using the code shown in Figure 9-1. The resulting ROC curve is shown in Figure 9-2 (Appendix).

```
fit <- glm(data=besttrain, GroundTruth ~. -ActualTruth -ImageFindingID
        -StudyFindingID -ImageID -StudyID -LeftBreast -MLO -xlocation
        -ylocation -xnipplelocation -ynipplelocation, family="binomial")
```

Figure 9-1. *Code creating model fit.*

Logistic Model by stepwise regression of AIC (fitAIC)

A stepwise regression process was used to optimize the fit model that had all the variables so that only the variables that would create the lowest AIC value would remain. The resulting model and ROC curve are shown in Figures 10-1 and 10-2 (Appendix).

```
fitAIC<-glm(formula = GroundTruth ~ V2 + V3 + V5 + V8 + V11 + V12 + V13 +
              V14 + V15 + V17 + V18 + V20 + V21 + V23 + V24 + V25 + V27 +
              V32 + V33 + V34 + V35 + V36 + V37 + V38 + V39 + V44 + V47 +
              V50 + V51 + V54 + V55 + V57 + V58 + V61 + V62 + V67 + V73 +
              V74 + V75 + V76 + V77 + V80 + V84 + V85 + V86 + V87 + V93 +
              V98 + V100 + V104 + V105 + V107 + V108 + V112, family =
              "binomial", data = besttrain)
```

Figure 10-1. *Code creating model fitAIC.*

Logistic Model by stepwise regression of BIC (fitBIC)

A stepwise regression process was used to optimize the fit model that had all the variables so that only the variables that would create the lowest BIC value would remain. The resulting model is shown in Figure 11-1 and the resulting ROC curve is shown in Figure 11-2 (Appendix).

```
step <- stepAIC(fit, direction="both", k=log(nrow(fit$data)))
#Result: GroundTruth ~ V2 + V3 + V13 + V18 + V23 + V25 + V35 + V58 + V62
#+V73 + V74 + V77 + V86 + V87 + V107)
fitBIC <- glm(data=besttrain, GroundTruth~ V2 + V3 + V13 + V18
            + V23 + V25 + V35 + V58 + V62 +V73 + V74 + V77 + V86 + V87 +
             V107, family="binomial")
```

Figure 11-1. *Code creating model fitBIC.*

Logistic Model by highly significant variables from BIC (fitBICsigvar)

Based on the BIC optimization model, it was noted that some variables had greater significance than others. These variables with comparatively low significance were removed from the model and a new model was created called fitBICsigvar. This model's code is shown in Figure 12-1 and the ROC curve is in Figure 12-2 (Appendix).

```
fitBICsigvar <- glm(data=besttrain, GroundTruth~ V2 + V3 + V13 + V18
                + V23 + V25 + V35 + V58 + V62 + V74 + V77 + V86,
                family="binomial")
```

Figure 12-1. *Code creating model fitBICsigvar*.

Combination of Results

Table 2.

Validation metrics of created models.

Alternative	D-Statistic	Accuracy	True Positive of Images Rate	Num. Images Missed	Num. People Missed	AUC
fit	0.5697140	81.32%	81.33%	42	0	0.9715
fitvi	0.4434257	57.20%	94.22%	13	0	0.9129
fitD	0.5312627	85.65%	83.11%	38	0	0.9212
fitAIC	0.5765670	82.46%	82.22%	40	0	0.9644
fitBIC	0.5617996	83.99%	83.11%	38	0	0.9460
fitBICsigvar	0.5669441	84.52%	84.0%	36	0	0.9423
Pruned Tree	0.4434257	81.36%	78.67%	48	0	-

**Assumed cutoff of prob ≥0.5; Results based on predicting the test data.*

The six logistic models and the classification tree all have performances in a similar range. Six validation metrics were collected on the models when they were used to predict the ground truth of images in the test dataset. The best score for each of the metrics is highlighted in grey on Table 2. Different models outperformed each other on different metrics. The model fit had the largest area under the ROC curve based on a cutoff of 0.5. The model fitvi had the most true positives for images as well as missed the fewest number of malignant images. The model fitD overall had the highest accuracy when predicting the ground truth of images, and the model fitAIC had the greatest D-statistic which means that the average means of ground truth predictions had the widest separation.

Conclusion

The resulting models from the analysis all created solutions that did not misdiagnosis a person who has cancer. However, certain models would be preferred to others depending on the nature of the problem. In this case, because we are working with people's health and the detection of cancer in its earliest stages is one of the most important factors in patients'

prognoses, the team chose model fitvi. This model has the highest true positive rate, meaning most cancerous images are not missed. The tradeoff of this is that it has an accuracy of only 57.20%, meaning a large number of patients' files are being sent on to the radiologist for inspection (17,360 if the cutoff is 0.5), this is still a reduction from the overall total of files that were originally intended to be sent on (all 40,630 images in the test dataset). While the other models have less false positives, there is an ethical responsibility to use the model that has the greatest probability of detecting a cancerous image. For this reason, the team also recommends a second round of validation with further data. While the dataset provided was extremely extensive (102, 294 images), it still only provided information about 1,712 patients, and only 40% of these patients' images were used in the validation stage of the modeling process.

Additionally, the team selected a cutoff of 0.15 for the model fitvi to be used in sending patients for greater inspection to the radiologist. This 0.15 cutoff value was chosen because it resulted in zero false negatives and can be verified on the ROC curve in Figure 7-2 (Appendix). Then the team created code to return a list of patients' Study IDs that the model recommends a radiologist to inspect all corresponding images, further shown in Figure 13. The reason the Study IDs were returned and not a list of exact images was to help combat the confirmation bias, where a person only sees what they want or expect to see.

```
Radiologist <- unique(test[test$predvi>=0.15,"StudyID"])
Radiologist
```

Figure 13. *Code to return list of Study IDs to be sent to radiologist for inspection.*

While the team has selected model fitvi to be the most preferable based on this scenario, if information about what the 117 features was provided, other variables may have been selected in the model (or other intricacies of the relationship could be determined). Recommendations for further studies could also include analyzing images of patients taken at different hospitals if this tool is meant to be used at various medical institutions to ensure that it is robust.

Appendix

Table 1.

Resulting numeric value of variable importance.

V4	V3	V6	V5	V14	V21	V68	V58	V10	V66
83.419020	81.384409	65.107528	64.203256	58.551561	38.229450	18.227517	14.816350	12.569688	11.883645
V9	V46	V48	V67	V17	V45	V20	V83	V24	V35
9.434393	8.713206	7.028007	6.886790	6.403074	6.389684	5.038102	4.066163	3.939369	3.810879
V65	V94	V25	V55	V105	V109	V90	V91	V92	V1
3.775723	3.775723	3.492940	3.302108	3.247998	2.678571	2.468376	2.468376	2.468376	2.416620
V50	V61	V34	V2	V33	V93	V63	V64	V110	V26
2.232143	2.232143	2.027027	1.520270	1.520270	1.481026	1.452874	1.452874	1.389423	1.386877
V85	V84	V19	V7	V8	V52	V99			
1.386877	1.260798	1.234188	1.227222	1.227222	1.182432	1.182432			

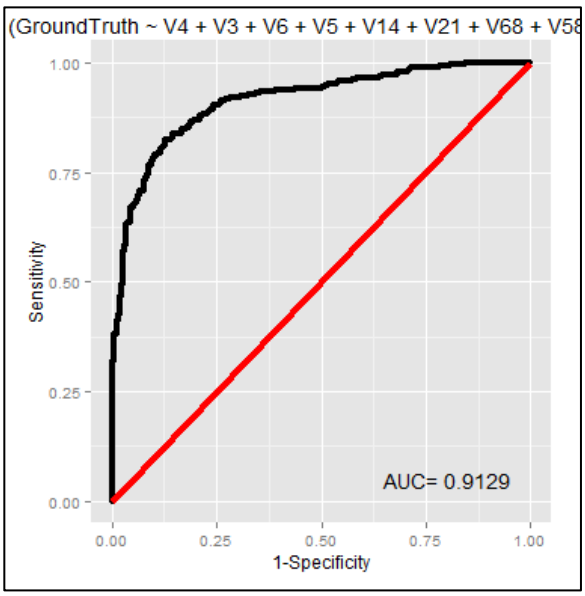

Figure 7-2. *ROC curve of model fitvi.*

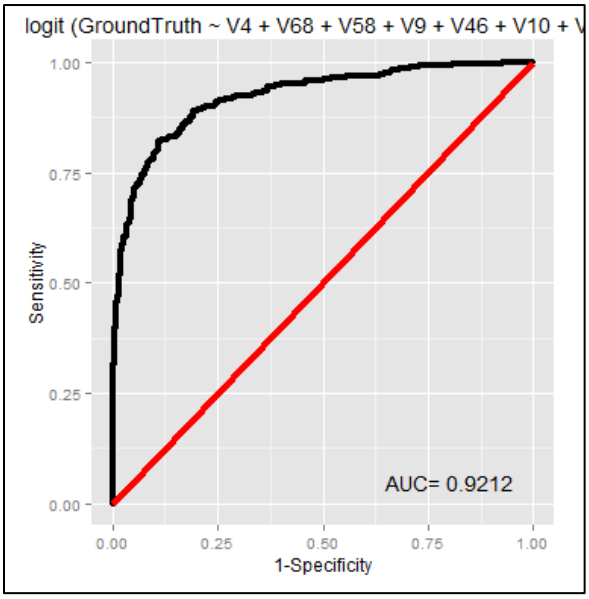

Figure 8-2. *ROC curve of model fitD.*

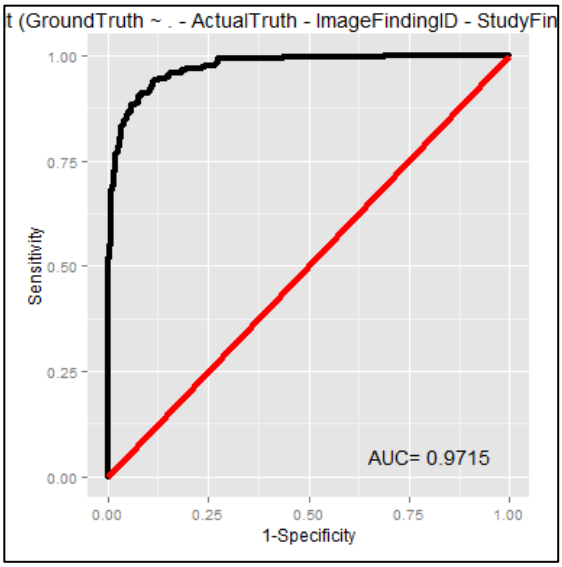

Figure 9-2. *ROC curve of model fit.*

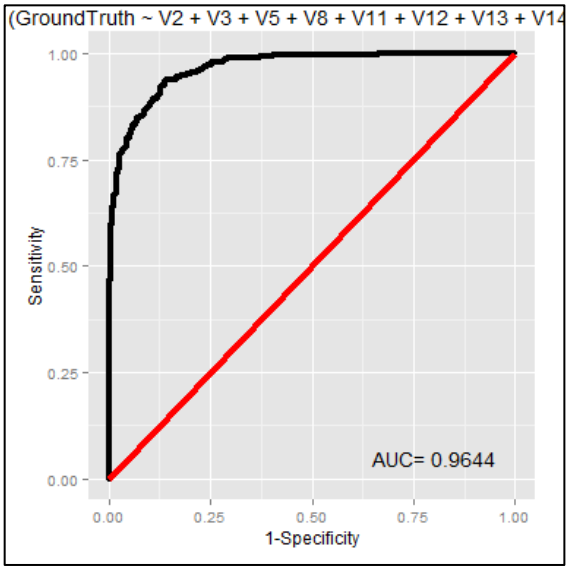

Figure 10-2. *ROC curve of model fitAIC.*

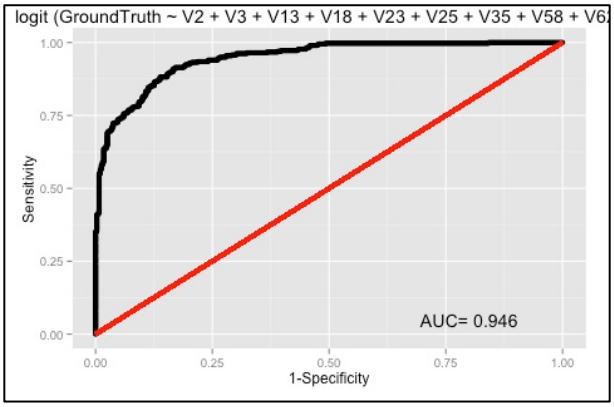

Figure 11-2. *ROC curve of model fitBIC.*

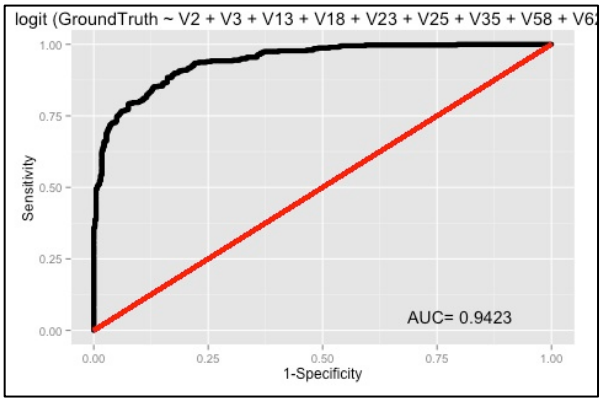

Figure 12-2. *ROC curve of model fitBICsigvar.*