SMU Data Science

Statistics 6372

Sec. 402 (2019 Spr)

Final Project

Jeremy Otsap [jotsap@smu.edu](mailto:jotsap@smu.edu)

Tej Tenmattam [ttenmattam@smu.edu](mailto:ttenmattam@smu.edu)

Arnold Zhang [arnoldz@smu.edu](mailto:arnoldz@smu.edu)

Contents

[Introduction 3](#_Toc6775172)

[Data Description 3](#_Toc6775173)

[Overview of the Data Set 3](#_Toc6775174)

[Exploratory Data Analysis 4](#_Toc6775175)

[Data Assumption Checks 4](#_Toc6775176)

[Analysis of Data Correlation 4](#_Toc6775177)

[Overall Sample Behavior 6](#_Toc6775178)

[Principal Components Analysis 8](#_Toc6775179)

[Objective 1 – Explanatory Analysis: Logistic Regression 10](#_Toc6775180)

[Comparison of outputs for different logistic models: 12](#_Toc6775181)

[Interpretation for Objective 1 Logistic Regression 15](#_Toc6775182)

[Conclusion: 15](#_Toc6775183)

[Objective 2 – Model Creation & Comparison 16](#_Toc6775184)

[Logistic Regression: 16](#_Toc6775185)

[k-Nearest Neighbours: 16](#_Toc6775186)

[Random Forest: 17](#_Toc6775187)

[Results: 17](#_Toc6775188)

[Conclusion: 18](#_Toc6775189)

[Future Work: 18](#_Toc6775190)

[APPENDIX 19](#_Toc6775191)

# Introduction

Data science plays an increasing role for organizations to manage their info and contribute to their processes. Statistical science forms a crucial backbone—it spans real world considerations and provides a framework to any object of study, providing characterization, modeling, and ultimately prediction.

In this statistics project we have two primary objectives. The first objective is to demonstrate exploratory data analysis and building a logistic regression model.

With the logistic regression model as the base, the second objective is to create competing models to contrast and enhance the prediction performance metrics.

We considered data from multiple data sources and selected a cancer diagnostic set from an academic setting. The project was an exercise in working and refining models for explanatory and predictive analysis.

# Data Description

## Overview of the Data Set

The data is a collection of the Breast Cancer Diagnostic Data Set collected by the University of Wisconsin-Madison. 569 observations were taken using digitized imaging of breast mass biopsy samples taken between 1989 and 1991. Biopsies were acquired using FNA, Fine Needle Aspiration.

The data set was taken from UC Irvine’s Machine Learning repository.

<https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>

Below is a description of the data set:

DEPENDENT VARIABLE

Diagnosis: Binomial – B [benign] or M [malignant]

INDEPENDENT VARIABLES

Ten real-valued features are computed for each cell nucleus:

a) radius (distance 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)

For each of these metrics the data set included

* Mean
* Standard Error
* Worst [for asymmetrical masses, this is the “longest” or “largest” value]

# Exploratory Data Analysis

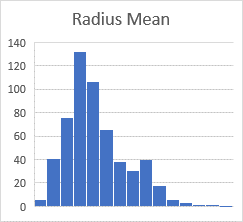
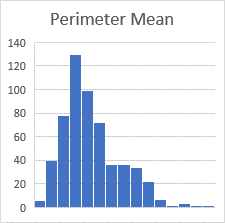
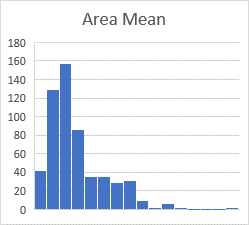
First, we validate assumptions of the data, then we apply visual analysis aided by a PCA to identify the influential parameters.

## Data Assumption Checks

**Sample Size**: The sample size is moderately sized with over 500 observations. Approximately 37.3% were Malignant and 62.7% were Benign.

**Independence**: Specific details as to patient selection were not available, so we will assume that independence is maintained. Points of possible concern would be that it is a sample without proper diversity to represent the true population.

**Normality**: Normality assumptions were for the most part met. In cases where skew exists, the Central Limit Theorem applied.

Figures A-1: Assumption Checks

## Analysis of Data Correlation

**Radius, Perimeter, Area**

These are obviously highly correlated since they are all a function of the radius and Pi.

This is confirmed with a visual examination via scatter plot that shows a strong positive correlation present between size-based measurements of the tumorous mass and the “malignant” vs “benign” diagnosis.

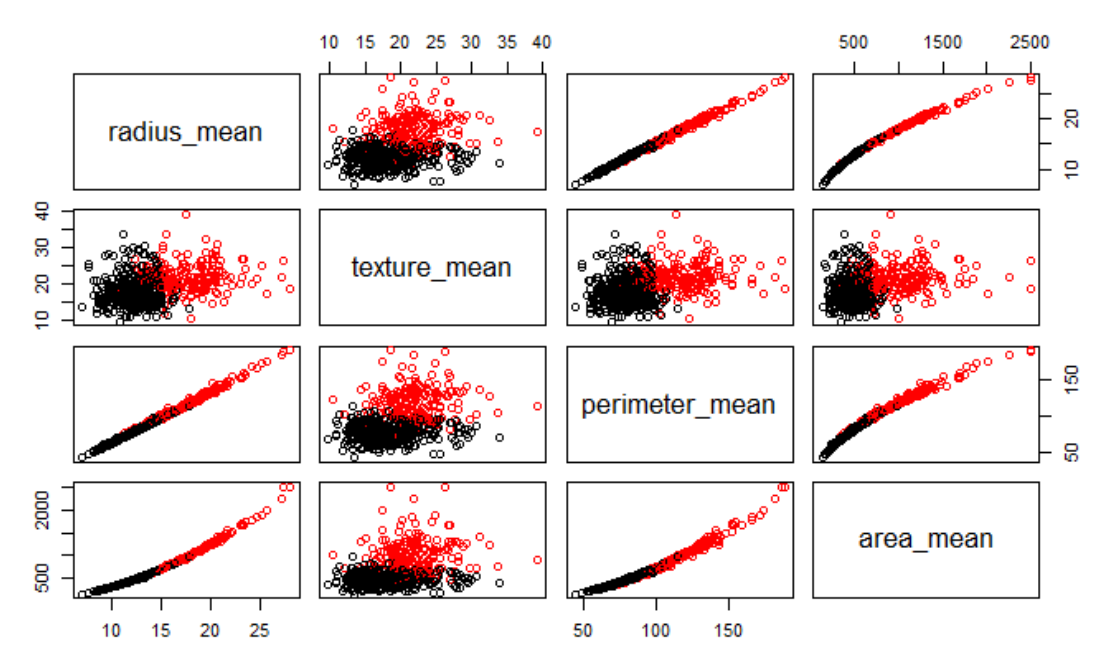
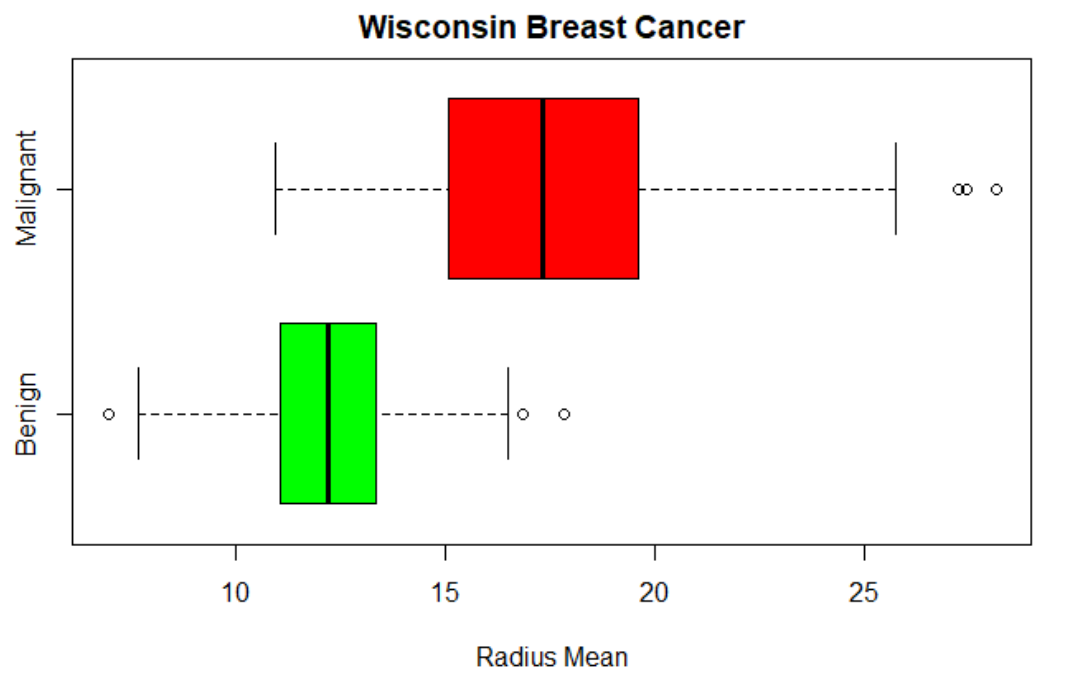
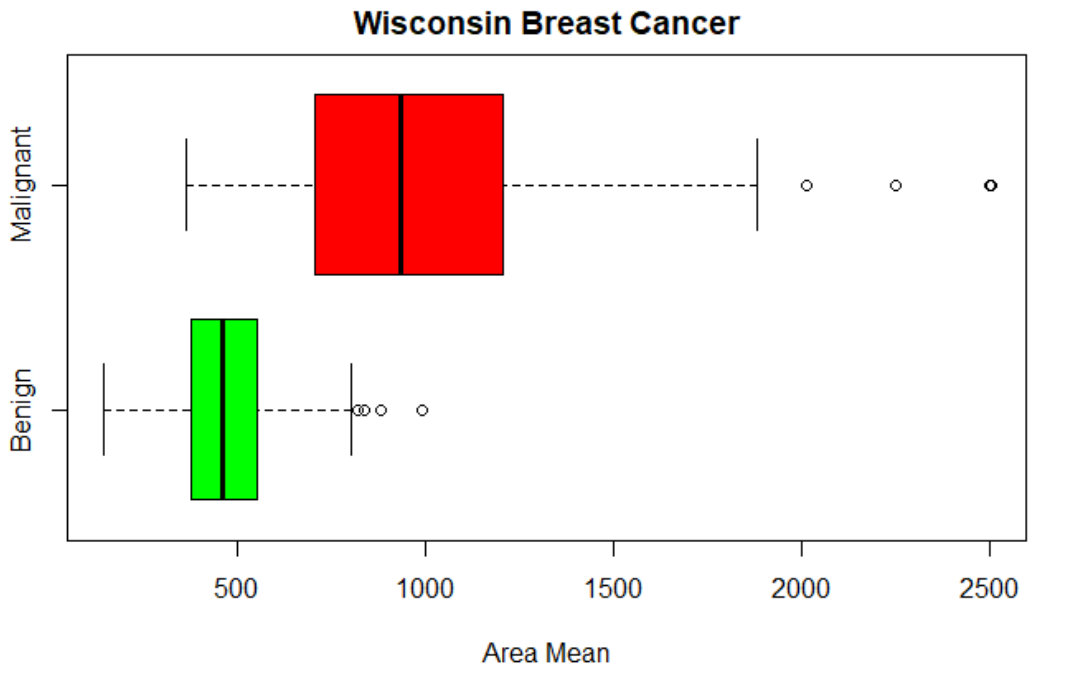


Figure A-2: Correlations

The line chart below shows the progression of measurements, from smallest to largest, for both radius and area, separated by diagnosis. Overall the malignant biopsies appear to be larger.

Figures A-3: Comparison of Benign & Malignant

A box plot comparison also validates this initial observation.

Figures A-4: Boxplots of Benign vs. Malignant

## Overall Sample Behavior

Looking at the dashboard below, it’s apparent that despite the majority of samples taken being benign, the malignant samples overall make up more than half the area, and the variance of the malignant samples is almost 7 times higher.



Figure A-5: Variance

When we select just the Benign subset, we can see across all measurements of variance, standard deviation, and standard error for measurements of the samples size [i.e. radius, perimeter, and area], that these are lower across the board.



Figure A-6: Variance, Benign

When selecting the Malignant sample, again, the mean, standard error, and variance of size-related measurements go higher than the overall average of those categories.

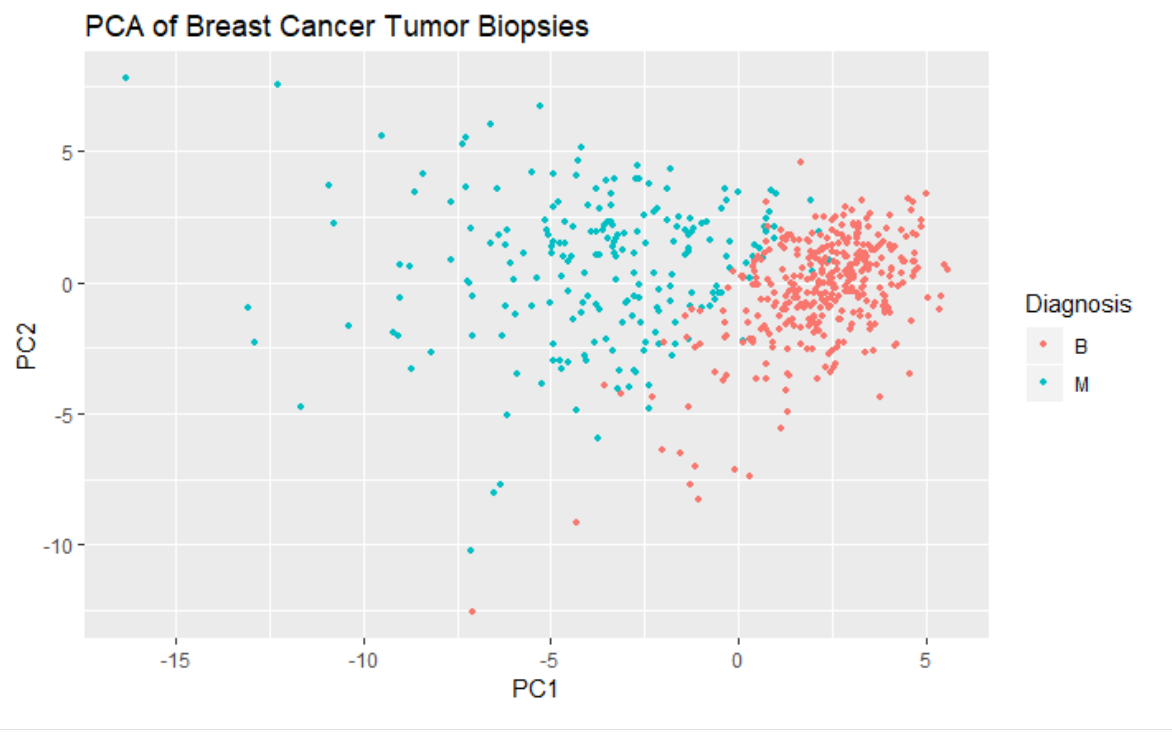
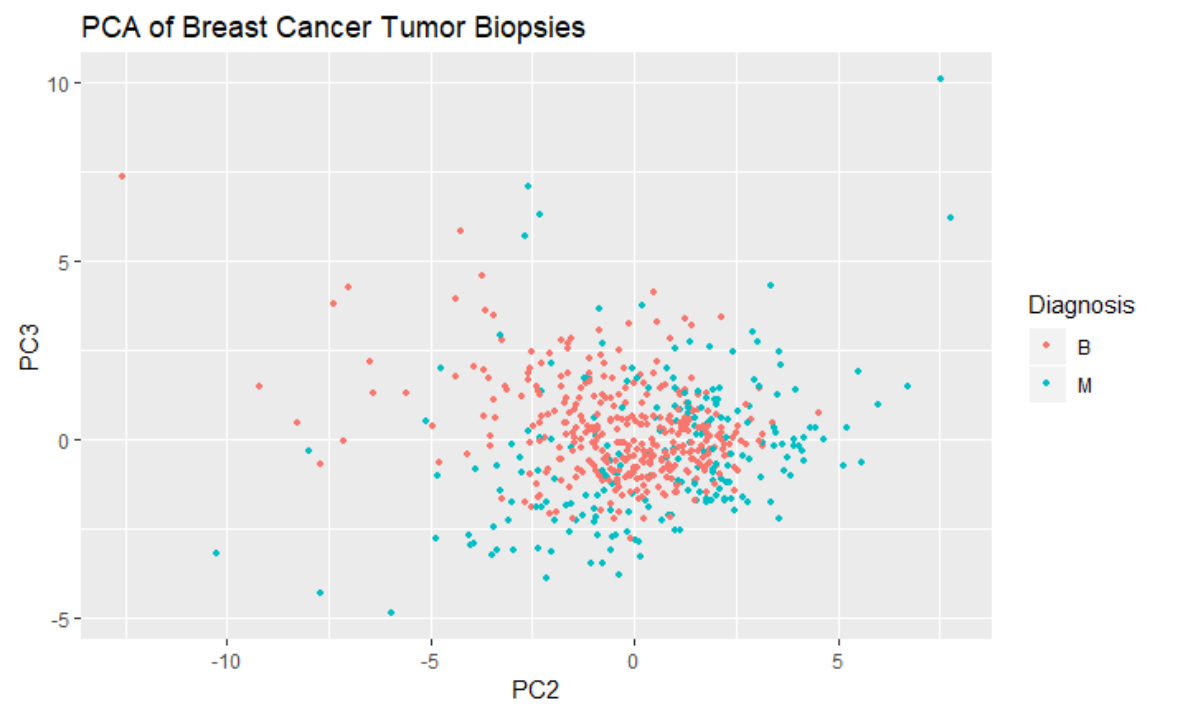
Other qualities like smoothness, symmetry, and fractal do not appear to favor either Malignant or Benign diagnoses. (see below highlighted Figure A-7).



Figure A-7: Variance, Malignant

## Principal Components Analysis

Below we will conduct PCA on the predictors and plot the first few Principal Components [PC's] against each other and examine what level of separation they provide. The number of PCs to explore can be dictated by the scree plot. As part of the initial exploratory data analysis, this will allow us to see if there are clear patterns that emerge in the data.

Figures A-8: PCA Scree plots

We can see in the first graphic that a clear separation exists between malignant and benign, which complements our initial visual examinations above. Because this clear separation exists in the PC's, this implies that a predictive model will probably do well.

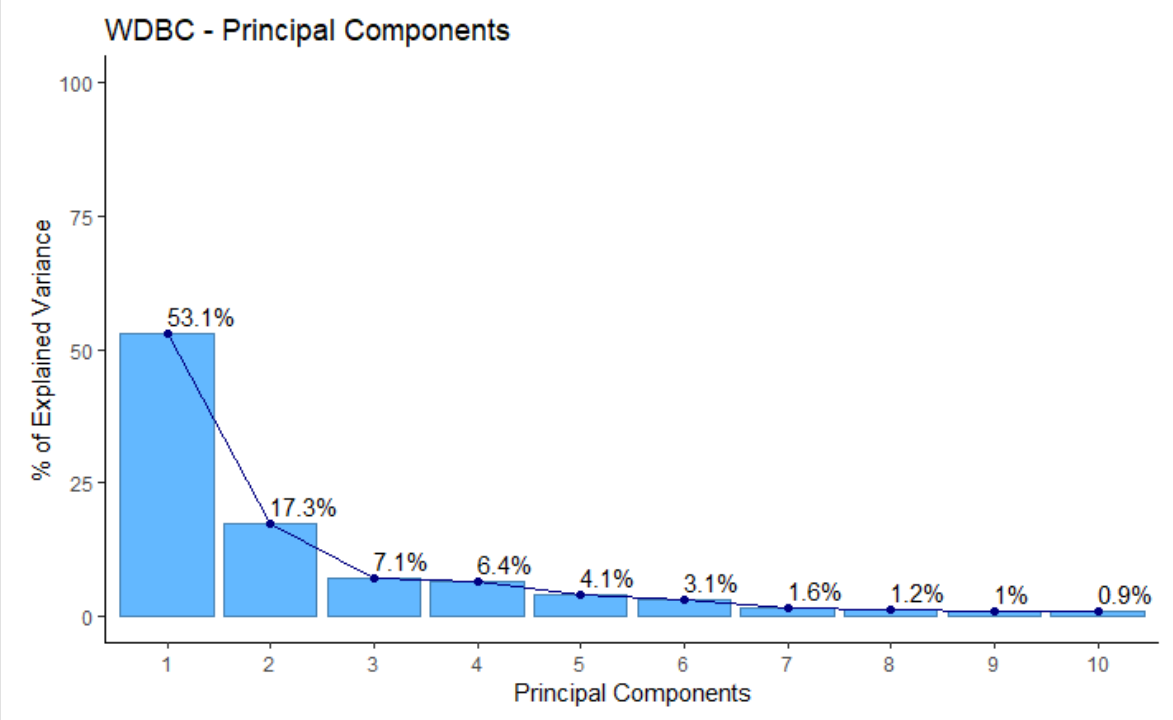


Figure A-9: % of Explained Variance vs. Principal Components

**Importance of components:**

**Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7**

**Standard deviation** 0.5751101 0.3281171 0.21051526 0.19984110 0.15953874 0.13832097 0.09923315

**Proportion of Variance** 0.5309769 0.1728349 0.07114442 0.06411259 0.04086072 0.03071494 0.01580837

**Cumulative Proportion** 0.5309769 0.7038118 0.77495621 0.83906880 0.87992952 0.91064446 0.92645284

Looking at the performance of the PC’s, we can see that 70% of the variance is explained by the first 2 components, and improvement significantly tapers off following that. This is further emphasized by the two scatterplots above showing significantly poorer separation using PC2 x PC2 compared to PC1 x PC2. We can see from this pairs plot of just the first few variables, that the malignant and benign groups are pretty well separated.

Given the above, we suspect an LDA analysis is appropriate to determine if the quantitative measurement parameters can be used to determine binomial categorical diagnosis response. However, we will do multiple analysis models and compare the results.

# Objective 1 – Explanatory Analysis: Logistic Regression

Our initial model will be a logistic regression without modeling in the variable interactions. However, the immediate concern is the covariance between multiple parameters. Running the full model through a VIF analysis shows significant covariance, with most of the scores having VIF values well above 10.

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

3806.115296 11.884048 3786.400419 347.878657 8.194282

compactness\_mean concavity\_mean concave\_points\_mean symmetry\_mean fractal\_dimension\_mean

50.505168 70.767720 60.041733 4.220656 15.756977

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

75.462027 4.205423 70.359695 41.163091 4.027923

compactness\_se concavity\_se concave\_points\_se symmetry\_se fractal\_dimension\_se

15.366324 15.694833 11.520796 5.175426 9.717987

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

799.105946 18.569966 405.023336 337.221924 10.923061

compactness\_worst concavity\_worst concave\_points\_worst symmetry\_worst fractal\_dimension\_worst

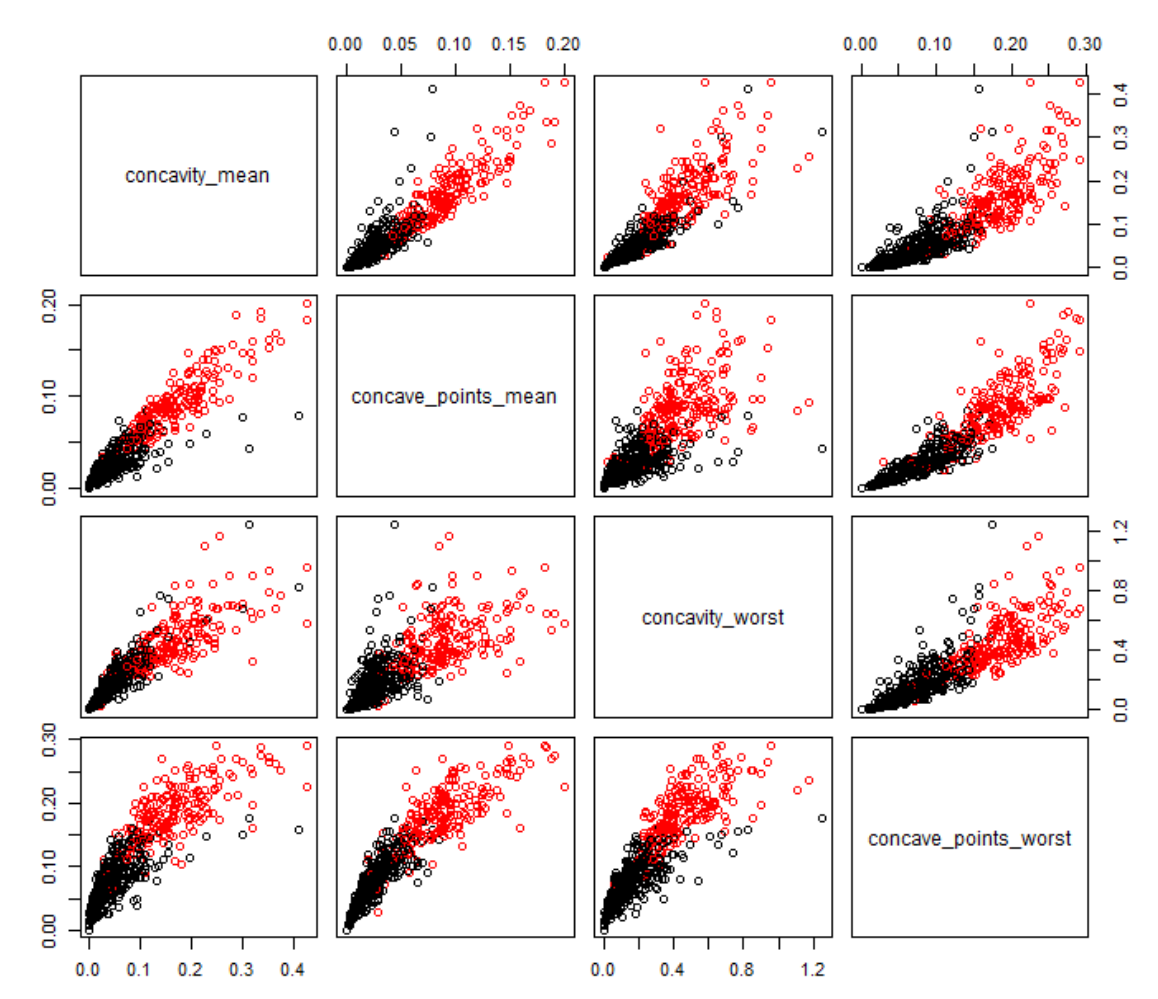
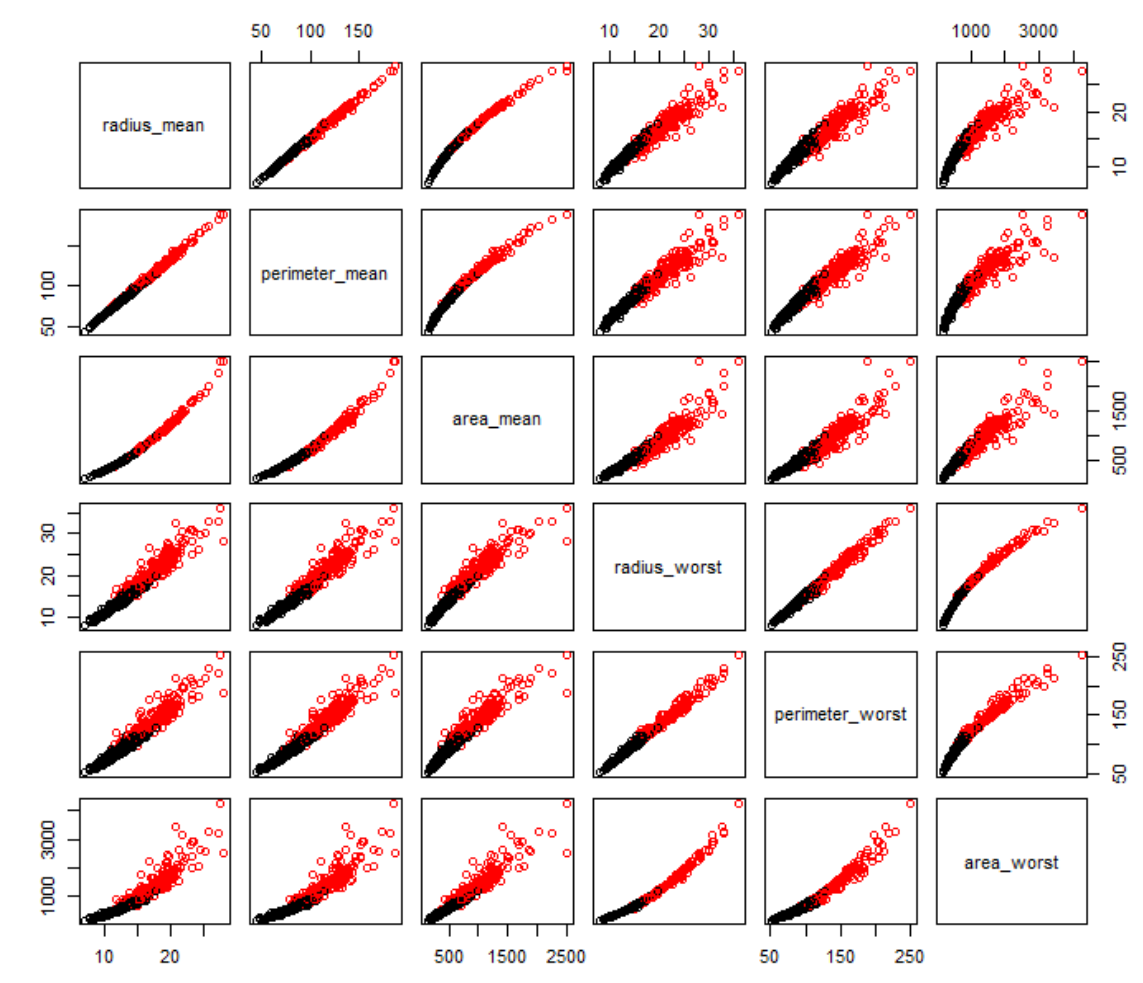
36.982755 31.970723 36.763714 9.520570 18.861533

This is not entirely surprising because:

1. Perimeter, Radius, and Area are all of a function of each other with respect to Pi.
2. The other category of measurements is simply the standard error [i.e. variance] or the worst single measurement used to derive this.

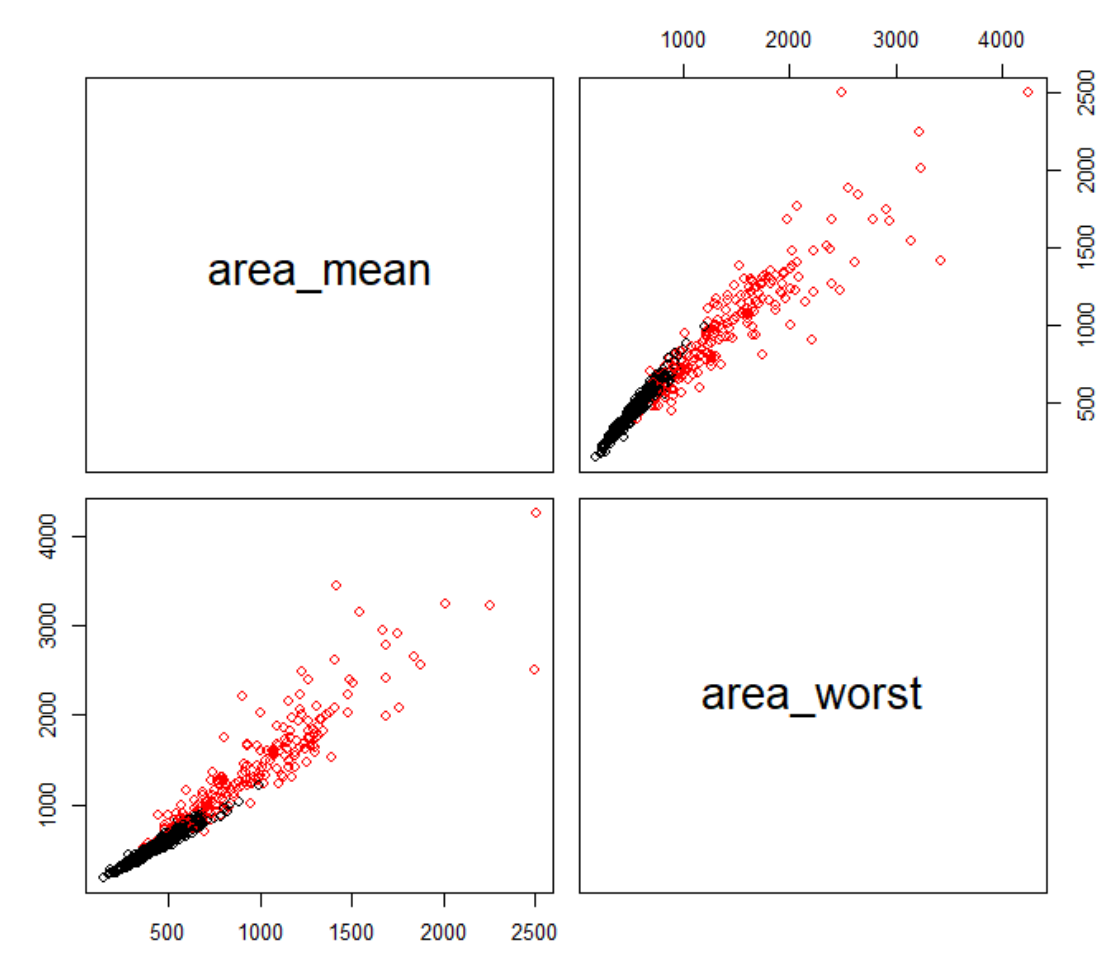
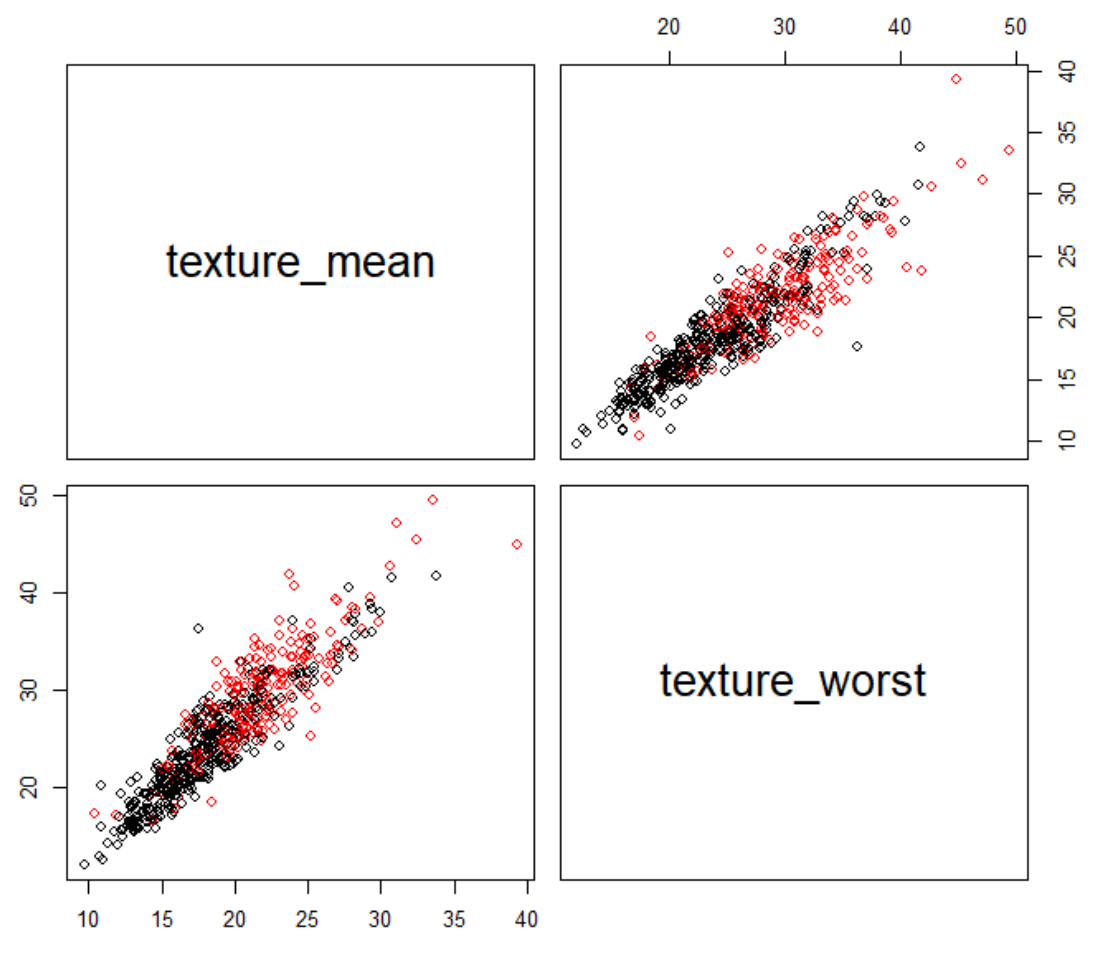
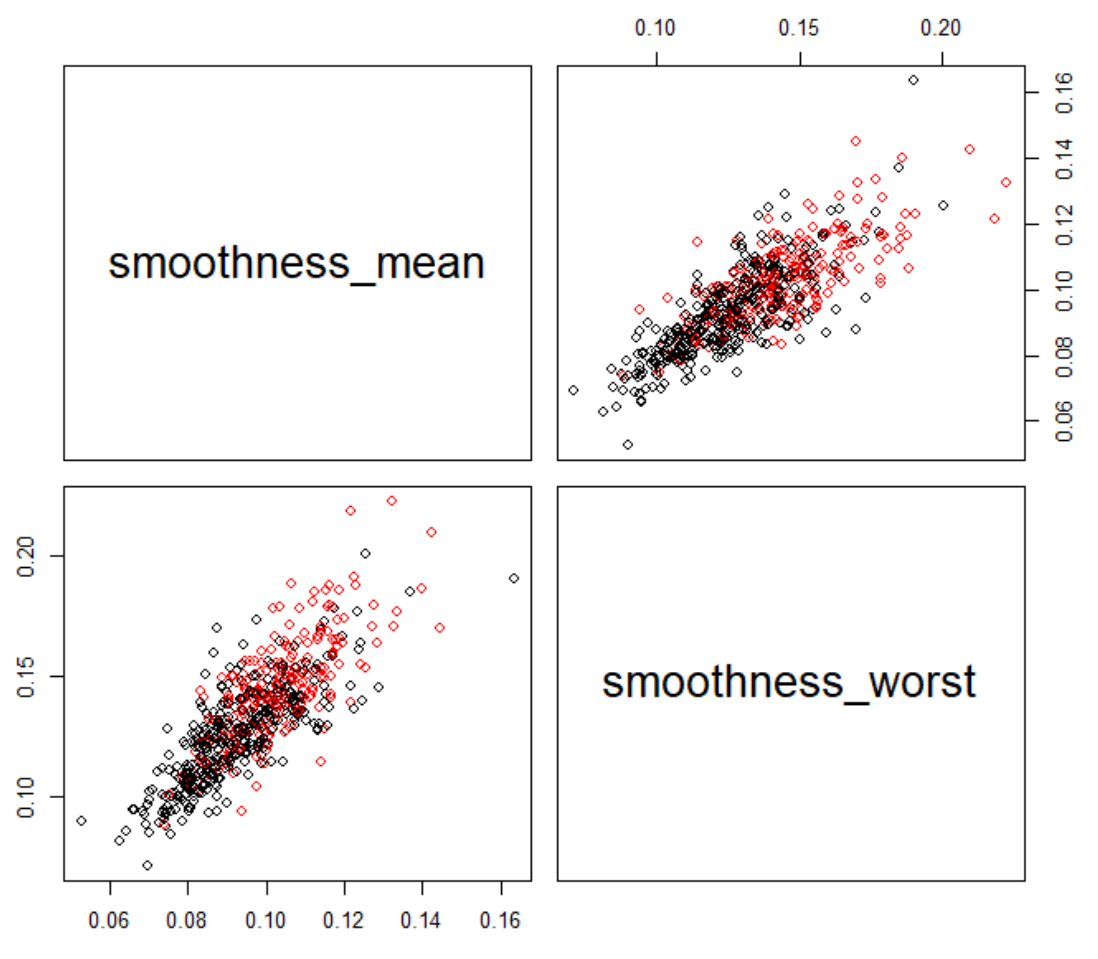
Visually inspecting this correlation also verifies this assumption as well.

**Radius, Perimeter, Area - Mean vs Worst** **Concavity & Concavity Point: Mean vs Worst**



Figures A-10: Correlations, Mean vs. Worst

**Area: Mean vs Worst Texture: Mean vs Worst Smoothness: Mean vs Worst**

Figures A-11: Correlations, Mean vs. Worst

By simplifying the model to only use the Area measurements to represent the tumor size [which overall had the lower comparative VIF values than respective Radius and Perimeter measurements], and only keep the “mean” category of measurements, we were able to reduce the parameter VIF’s to values less than 10 across the board.

texture\_mean area\_mean smoothness\_mean compactness\_mean concavity\_mean

1.764829 3.250272 4.288467 5.763065 4.987014

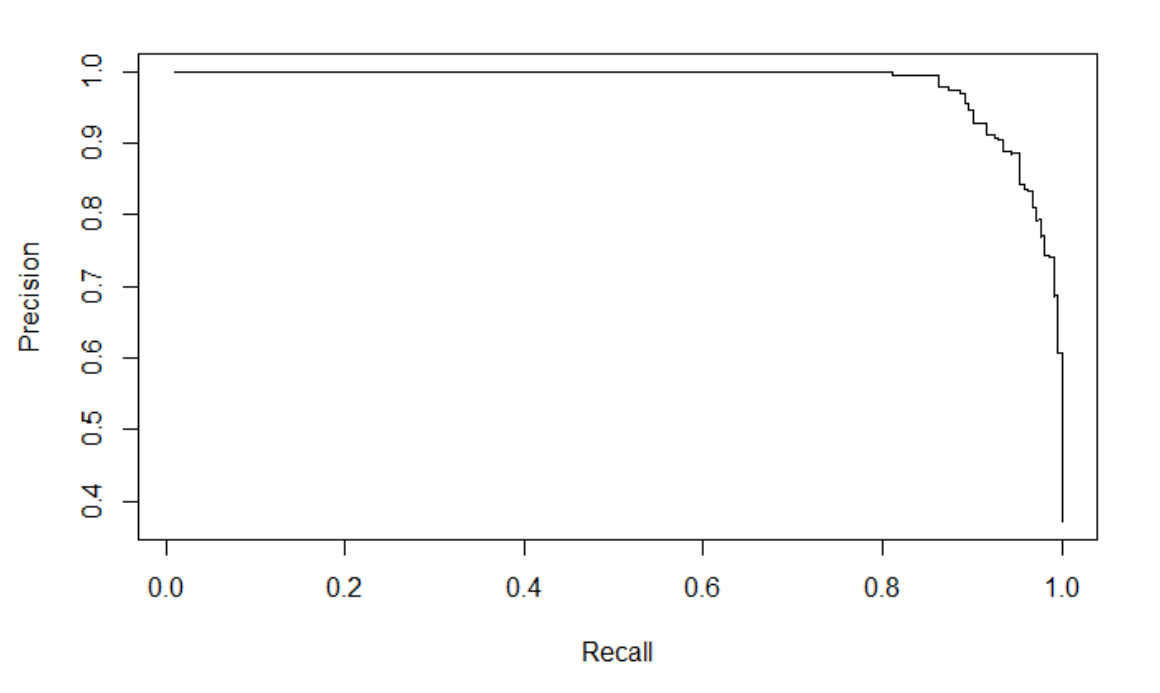
concave\_points\_mean symmetry\_mean fractal\_dimension\_mean

5.744502 1.823926 8.311224

When comparing the two logistic regression models, the “full” model had a significantly worse AIC score than the simplified one.

**ROC Curve for Reduced Logistic Model**

We can see significant lift on the reduced model that accounts for Area mean, texture mean, smoothness mean, and concave points mean.



Figures A-12: ROC Curve for Reduced Logistic Model

## Comparison of outputs for different logistic models:

In the following section, we are comparing different logistic models.

OUTPUT FOR FULL LOGISTIC MODEL:

glm(formula = diagnosis ~ ., family = binomial(link = "logit"),

data = bc.boolean, control = list(maxit = 50))

Deviance Residuals:

Min 1Q Median 3Q Max

-8.49 0.00 0.00 0.00 8.49

Coefficients:

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

(Intercept) -1.704e+16 1.216e+08 -140095174 <2e-16 \*\*\*

radius\_mean -4.508e+15 4.930e+07 -91446497 <2e-16 \*\*\*

texture\_mean -2.681e+13 2.257e+06 -11879545 <2e-16 \*\*\*

perimeter\_mean 5.592e+14 7.131e+06 78428443 <2e-16 \*\*\*

area\_mean 6.374e+12 1.492e+05 42711010 <2e-16 \*\*\*

smoothness\_mean 4.804e+16 5.731e+08 83827859 <2e-16 \*\*\*

compactness\_mean -2.931e+16 3.789e+08 -77357485 <2e-16 \*\*\*

concavity\_mean 3.110e+14 2.971e+08 1046677 <2e-16 \*\*\*

concave\_points\_mean 8.171e+15 5.623e+08 14531119 <2e-16 \*\*\*

symmetry\_mean -8.564e+15 2.110e+08 -40582198 <2e-16 \*\*\*

fractal\_dimension\_mean -1.030e+16 1.583e+09 -6507058 <2e-16 \*\*\*

radius\_se 8.161e+15 8.821e+07 92524229 <2e-16 \*\*\*

texture\_se -4.617e+14 1.047e+07 -44105313 <2e-16 \*\*\*

perimeter\_se -8.989e+14 1.168e+07 -76951480 <2e-16 \*\*\*

area\_se 6.591e+12 3.971e+05 16595576 <2e-16 \*\*\*

smoothness\_se -1.037e+16 1.882e+09 -5509676 <2e-16 \*\*\*

compactness\_se 4.729e+16 6.164e+08 76728075 <2e-16 \*\*\*

concavity\_se -2.689e+16 3.696e+08 -72764771 <2e-16 \*\*\*

concave\_points\_se 1.993e+17 1.549e+09 128680606 <2e-16 \*\*\*

symmetry\_se -4.806e+16 7.749e+08 -62015895 <2e-16 \*\*\*

fractal\_dimension\_se -4.364e+17 3.317e+09 -131546226 <2e-16 \*\*\*

radius\_worst 1.102e+15 1.647e+07 66911477 <2e-16 \*\*\*

texture\_worst 1.390e+14 1.974e+06 70396825 <2e-16 \*\*\*

perimeter\_worst 6.421e+13 1.686e+06 38076691 <2e-16 \*\*\*

area\_worst -9.950e+12 9.082e+04 -109559685 <2e-16 \*\*\*

smoothness\_worst -6.379e+15 4.076e+08 -15649766 <2e-16 \*\*\*

compactness\_worst -7.493e+15 1.088e+08 -68848424 <2e-16 \*\*\*

concavity\_worst 6.282e+15 7.632e+07 82312806 <2e-16 \*\*\*

concave\_points\_worst -7.308e+15 2.597e+08 -28137342 <2e-16 \*\*\*

symmetry\_worst 1.098e+16 1.404e+08 78221084 <2e-16 \*\*\*

fractal\_dimension\_worst 4.755e+16 6.771e+08 70226947 <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: 792.96 on 538 degrees of freedom

**AIC: 854.96**

The AIC for Full Logistic is 854.96.

OUTPUT FOR LASSO MODEL

|  |
| --- |
| > summary(main.lasso.glm)  Call:  glm(formula = diagnosis ~ concavity\_mean + concave\_points\_mean +  fractal\_dimension\_mean + radius\_se + smoothness\_se + compactness\_se +  symmetry\_se + fractal\_dimension\_se + radius\_worst + texture\_worst +  smoothness\_worst + concavity\_worst + concave\_points\_worst +  symmetry\_worst, family = binomial(link = "logit"), data = bc.clean)  Deviance Residuals:  Min 1Q Median 3Q Max  -1.6432 -0.0231 -0.0011 0.0001 3.4988  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) -49.94186 14.89788 -3.352 0.000801 \*\*\*  concavity\_mean 12.70977 38.43786 0.331 0.740903  concave\_points\_mean 26.32819 70.41186 0.374 0.708466  fractal\_dimension\_mean -44.16790 144.43201 -0.306 0.759754  radius\_se 15.63875 4.93316 3.170 0.001524 \*\*  smoothness\_se 273.77291 203.10733 1.348 0.177683  compactness\_se -95.33134 60.50492 -1.576 0.115119  symmetry\_se -73.18646 122.55678 -0.597 0.550398  fractal\_dimension\_se -270.98565 545.40470 -0.497 0.619293  radius\_worst 1.25629 0.42384 2.964 0.003036 \*\*  texture\_worst 0.37798 0.08829 4.281 1.86e-05 \*\*\*  smoothness\_worst 31.84886 36.50939 0.872 0.383019  concavity\_worst 8.12142 10.47793 0.775 0.438282  concave\_points\_worst 30.35111 27.64683 1.098 0.272285  symmetry\_worst 23.54037 15.20640 1.548 0.121609  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 751.440 on 568 degrees of freedom  Residual deviance: 55.675 on 554 degrees of freedom  **AIC: 85.675**  Number of Fisher Scoring iterations: 10  The AIC for Lasso is 85.675.  OUTPUT FOR STEPWISE SELECTION MODEL  > summary(main.glm.step)  Call:  glm(formula = diagnosis ~ radius\_mean + texture\_mean + area\_mean +  smoothness\_mean + compactness\_mean + concavity\_mean + concave\_points\_mean +  symmetry\_mean + fractal\_dimension\_mean + 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 + concavity\_worst + symmetry\_worst +  fractal\_dimension\_worst, family = binomial(link = "logit"),  data = bc.clean)  Deviance Residuals:  Min 1Q Median 3Q Max  -0.003832 0.000000 0.000000 0.000000 0.004291  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) -5.914e+03 2.619e+05 -0.023 0.982  radius\_mean -6.630e+03 1.150e+05 -0.058 0.954  texture\_mean 1.913e+02 1.345e+03 0.142 0.887  area\_mean 6.077e+01 1.079e+03 0.056 0.955  smoothness\_mean 3.914e+04 2.517e+05 0.155 0.876  compactness\_mean -8.621e+04 9.326e+05 -0.092 0.926  concavity\_mean 2.852e+04 2.402e+05 0.119 0.905  concave\_points\_mean 5.886e+04 1.544e+06 0.038 0.970  symmetry\_mean -1.964e+04 1.347e+05 -0.146 0.884  fractal\_dimension\_mean 1.626e+05 1.120e+06 0.145 0.885  perimeter\_se -1.253e+03 1.822e+04 -0.069 0.945  area\_se 1.562e+02 2.259e+03 0.069 0.945  smoothness\_se -9.793e+04 1.472e+06 -0.067 0.947  compactness\_se 9.217e+04 7.142e+05 0.129 0.897  concavity\_se -8.131e+04 1.097e+06 -0.074 0.941  concave\_points\_se 4.398e+05 6.736e+06 0.065 0.948  symmetry\_se -1.038e+05 2.160e+06 -0.048 0.962  fractal\_dimension\_se -1.092e+06 1.065e+07 -0.103 0.918  radius\_worst 2.226e+03 2.134e+04 0.104 0.917  texture\_worst 7.269e+01 3.150e+03 0.023 0.982  perimeter\_worst 1.267e+02 1.355e+03 0.093 0.926  area\_worst -1.626e+01 1.165e+02 -0.140 0.889  concavity\_worst 6.737e+03 1.051e+05 0.064 0.949  symmetry\_worst 2.201e+04 3.283e+05 0.067 0.947  fractal\_dimension\_worst 5.899e+04 1.032e+06 0.057 0.954  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 7.5144e+02 on 568 degrees of freedom  Residual deviance: 1.6713e-04 on 544 degrees of freedom  **AIC: 50**  Number of Fisher Scoring iterations: 25  The AIC for the Stepwise Selection Model is 50. |

OUTPUT FOR SIMPLIFIED LOGISTIC MODEL

glm(formula = diagnosis ~ texture\_mean + area\_mean + smoothness\_mean +

compactness\_mean + concavity\_mean + concave\_points\_mean +

symmetry\_mean + fractal\_dimension\_mean, family = binomial(link = "logit"),

data = bc.boolean, control = list(maxit = 50))

Deviance Residuals:

Min 1Q Median 3Q Max

-2.02338 -0.14079 -0.03572 0.01120 3.00158

Coefficients:

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

(**Intercept) -27.253090 6.400270 -4.258 2.06e-05 \*\*\***

texture\_mean 0.384407 0.063485 6.055 1.40e-09 \*\*\*

area\_mean 0.011787 0.002728 4.320 1.56e-05 \*\*\*

smoothness\_mean 79.598004 32.979533 2.414 0.0158 \*

compactness\_mean -13.731887 12.700676 -1.081 0.2796

concavity\_mean 13.249936 8.090709 1.638 0.1015

concave\_points\_mean 57.230227 28.042055 2.041 0.0413 \*

symmetry\_mean 17.779963 10.854712 1.638 0.1014

fractal\_dimension\_mean -26.454327 82.469431 -0.321 0.7484

---

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: 149.01 on 560 degrees of freedom

**AIC: 167.01**

The AIC for Simplified Logistic is 167.01.

CONFIDENCE INTERVALS FOR PARAMETERS OF REDUCED MODEL

2.5 % 97.5 %

(Intercept) -4.053919e+01 -15.31563155

texture\_mean 2.674975e-01 0.51837370

area\_mean 6.757782e-03 0.01749949

smoothness\_mean 1.771586e+01 147.27093557

compactness\_mean -3.951821e+01 11.10875994

concavity\_mean -2.529860e+00 29.24616163

concave\_points\_mean 3.911450e+00 114.21970795

symmetry\_mean -3.644306e+00 39.25139134

fractal\_dimension\_mean -1.925487e+02 132.60062490

## Interpretation for Objective 1 Logistic Regression

The significant parameters to the model were:

area\_mean [p-value < 0.001], texture\_mean [p-value < 0.001],

smoothness\_mean [p-value 0.0158], and concave\_points\_mean [p-value 0.0413].

area\_mean e0.011787 =~ 1.012

Texture\_mean e0.384407 =~ 1.469

smoothness\_mean e79.598004 =~ 3.706 \* 10^34

concave\_points\_mean e57.230227 = 7.158 \* 10^24

For Area, holding all other parameters constant, the 1 unit is [e0.011787 \* 1 unit ] = 1.012 times more likely to be Malignant than Benign, and

the 5 units is [ e0.011787 \* 5 units ] = 1.061 times more likely to be Malignant than Benign.

For Texture, holding all other parameters constant, 1 unit is [e0.384407 \* 1 unit ] = 1.469 times more likely to be Malignant than Benign, and

the 5 units is [e0.384407 \* 5 units] = 6.835 times more likely to be Malignant than Benign.

For Smoothness, holding all other parameters constant, 1 unit is [e79.598004 \* 1 unit ] =

3.706 \* 10^34 times more likely to be Malignant than Benign, and the

5 units is [e79.598004 \* 5 units ] = 6.996 \* 10^172 times more likely to be Malignant than Benign.

For Concave points, holding all other parameters constant, the 1 unit is [e57.230227 \* 1 unit ] = 7.158 \*10^24 times more likely to be Malignant than Benign, and

the 5 units [e57.230227 \* 5 units ] = 1.879 \* 10^124 times more likely to be Malignant than Benign.

## Conclusion:

Model-building and effectiveness depends highly upon the particular area, scope of study, and the nature of data available. Different models will vary in applicability and desirability of metrics, especially in the overall interpretability and predictive ability. In this case, the AIC metric is improved, going from a full to a reduced logistic model. The AIC is further improved with a LASSO, and is best when applying stepwise selection. Despite the Stepwise Feature Selection having a lower AIC score of 50, when looking at the practical significance in light of the parameter estimates, it likely overfits this data set, and will perform poorly for additional observations in the same or similar studies.

# Objective 2 – Model Creation & Comparison

We have selected 5-fold cross-validation for the models, given the relatively small sample size of the dataset. Also, from page 184 of book “An Introduction to Statistical Learning, 2013”.

|  |
| --- |
| *To summarize, there is a bias-variance trade-off associated with the choice of k in k-fold cross-validation. Typically, given these considerations, one performs k-fold cross-validation using k = 5 or k = 10, as these values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor from very high variance.* |

## Logistic Regression:

From the confusion matrix and model metrics, it can be observed that the logistic regression model has a prediction accuracy of 92.6%, recall of 90.5%, and F-Score of 90.14%.

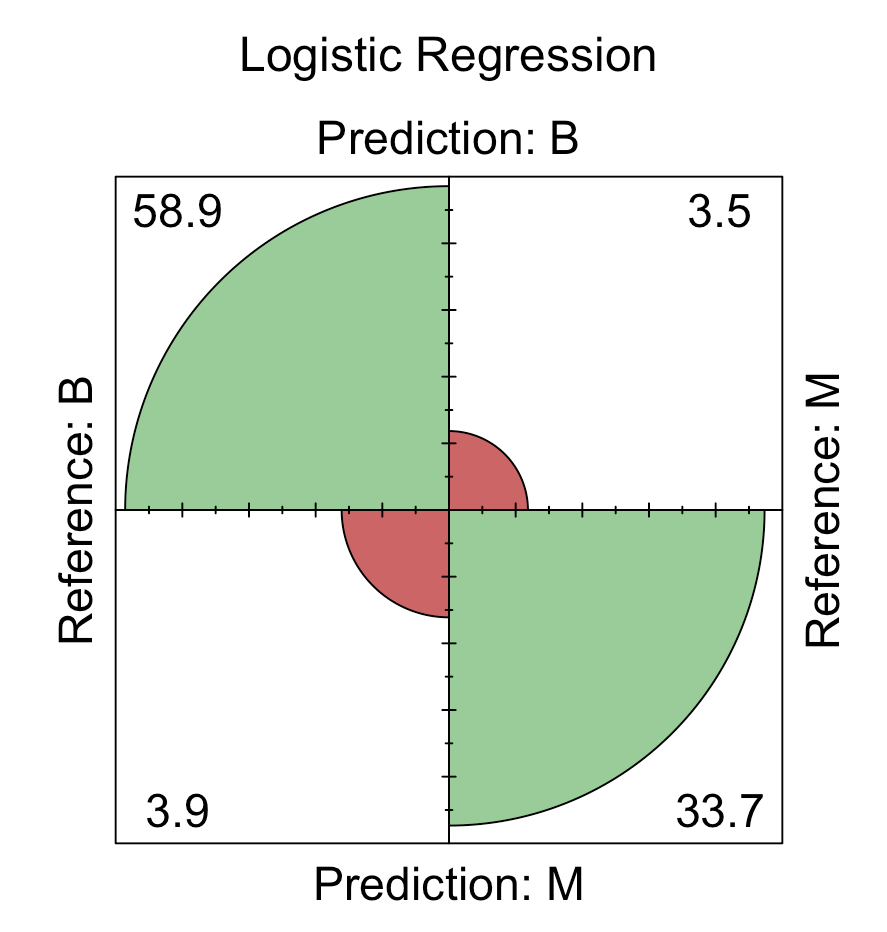


Figure B-1: Logistic Regression

## k-Nearest Neighbours:

From the confusion matrix and model metrics, it can be observed that the kNN model has a prediction accuracy of 97.3%, recall of 93.86%, and F-score of 96.36%.

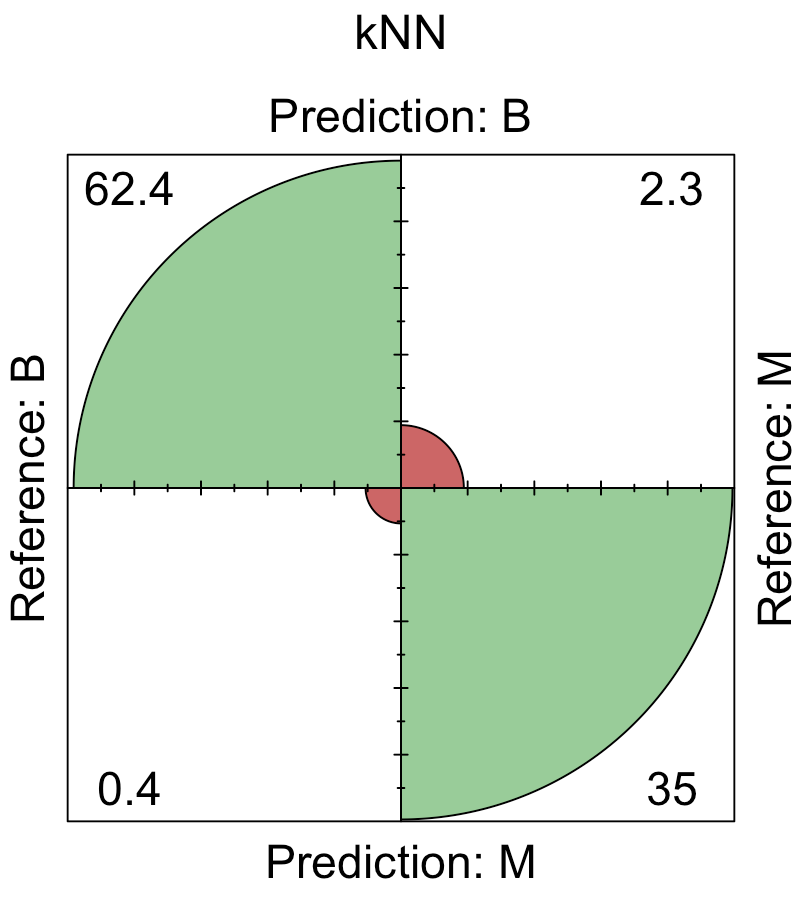
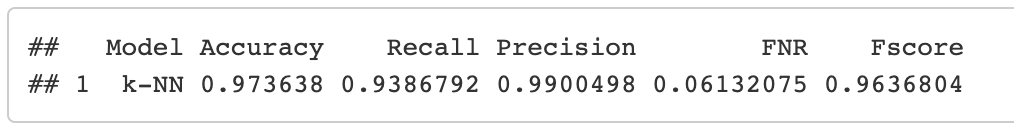
 

Figure B-2: kNN

## Random Forest:

From the confusion matrix and model metrics, it can be observed that the Random Forest model has a prediction accuracy of 94.9%, recall of 91.03%, and F-score of 93.01%.

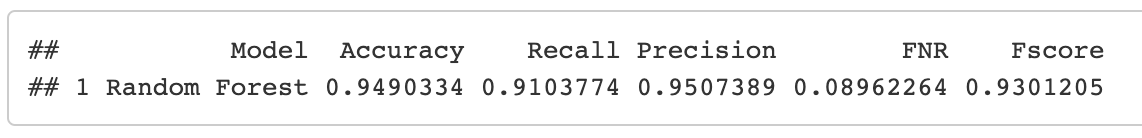
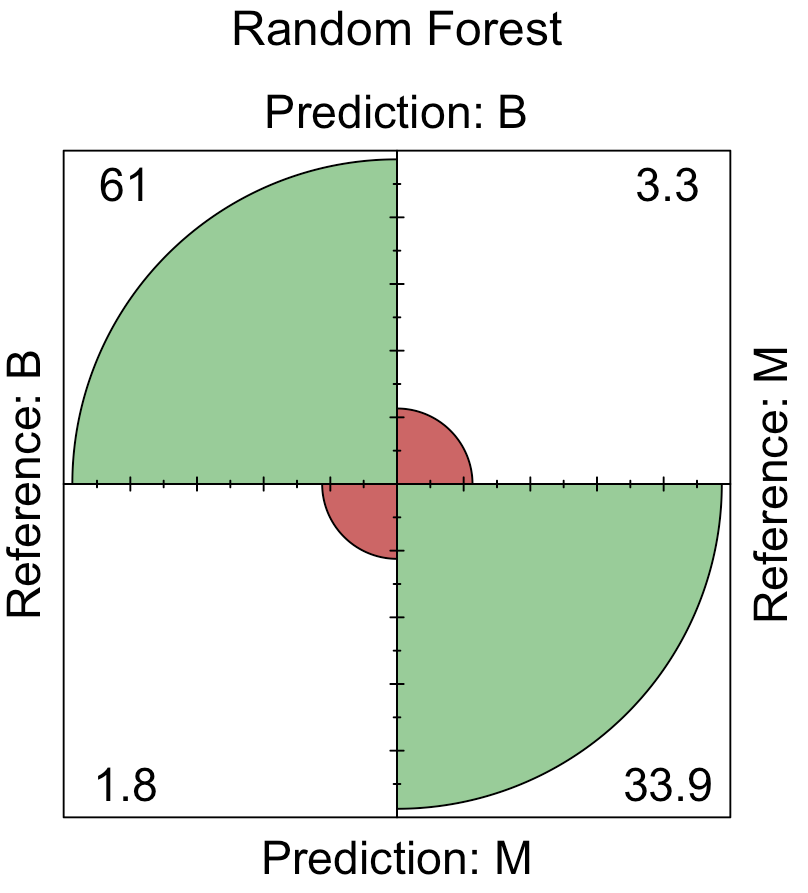
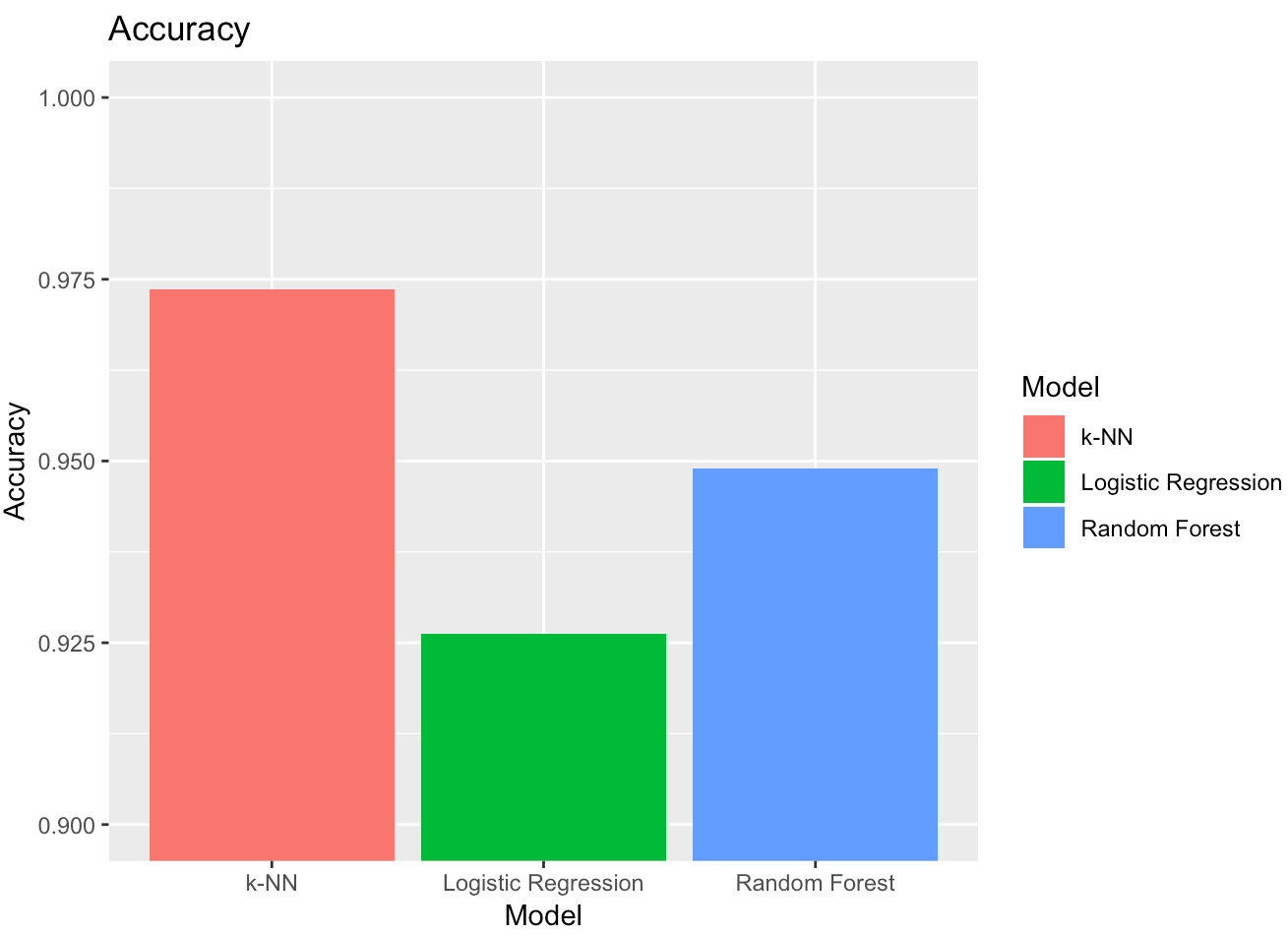
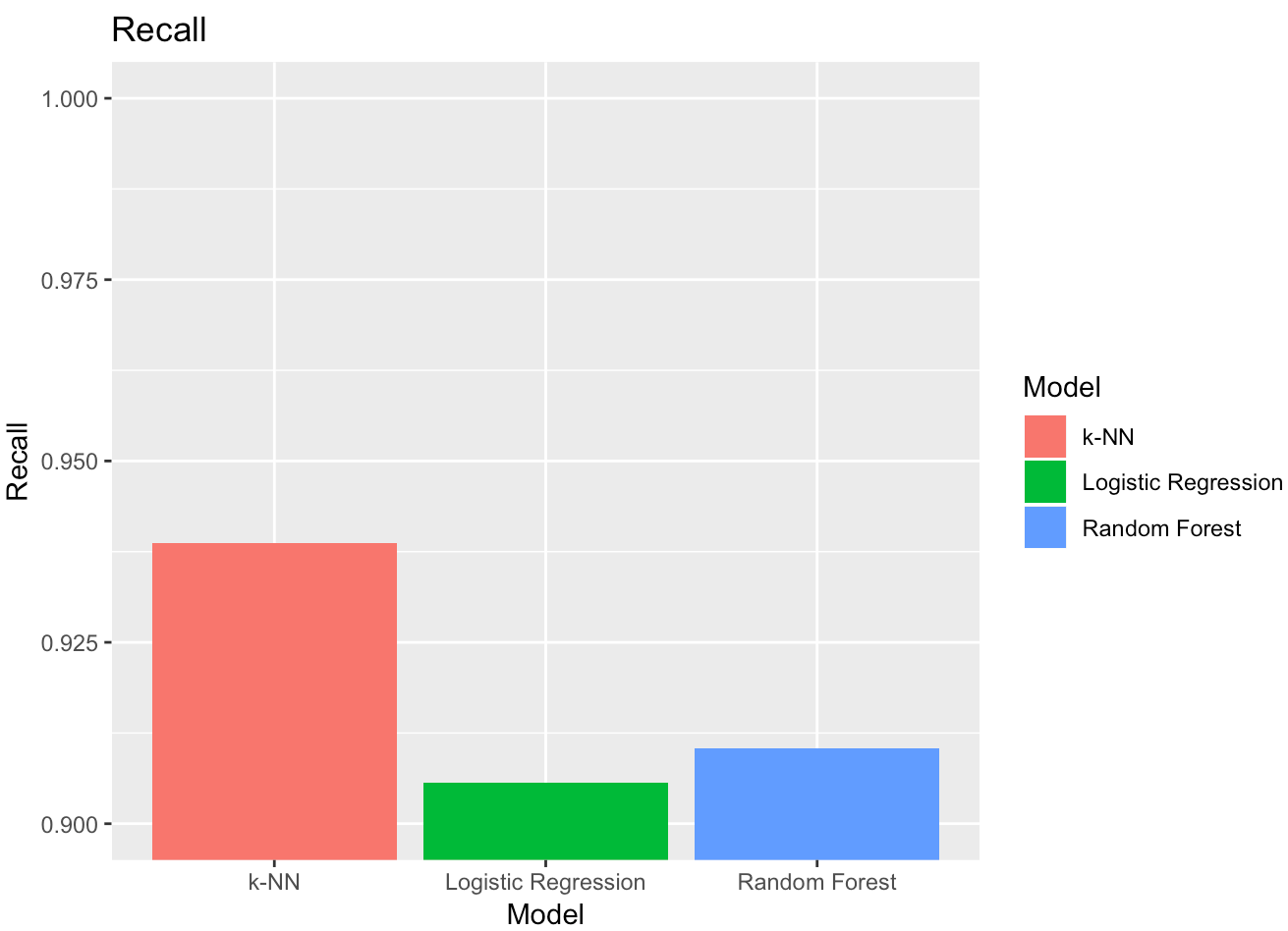
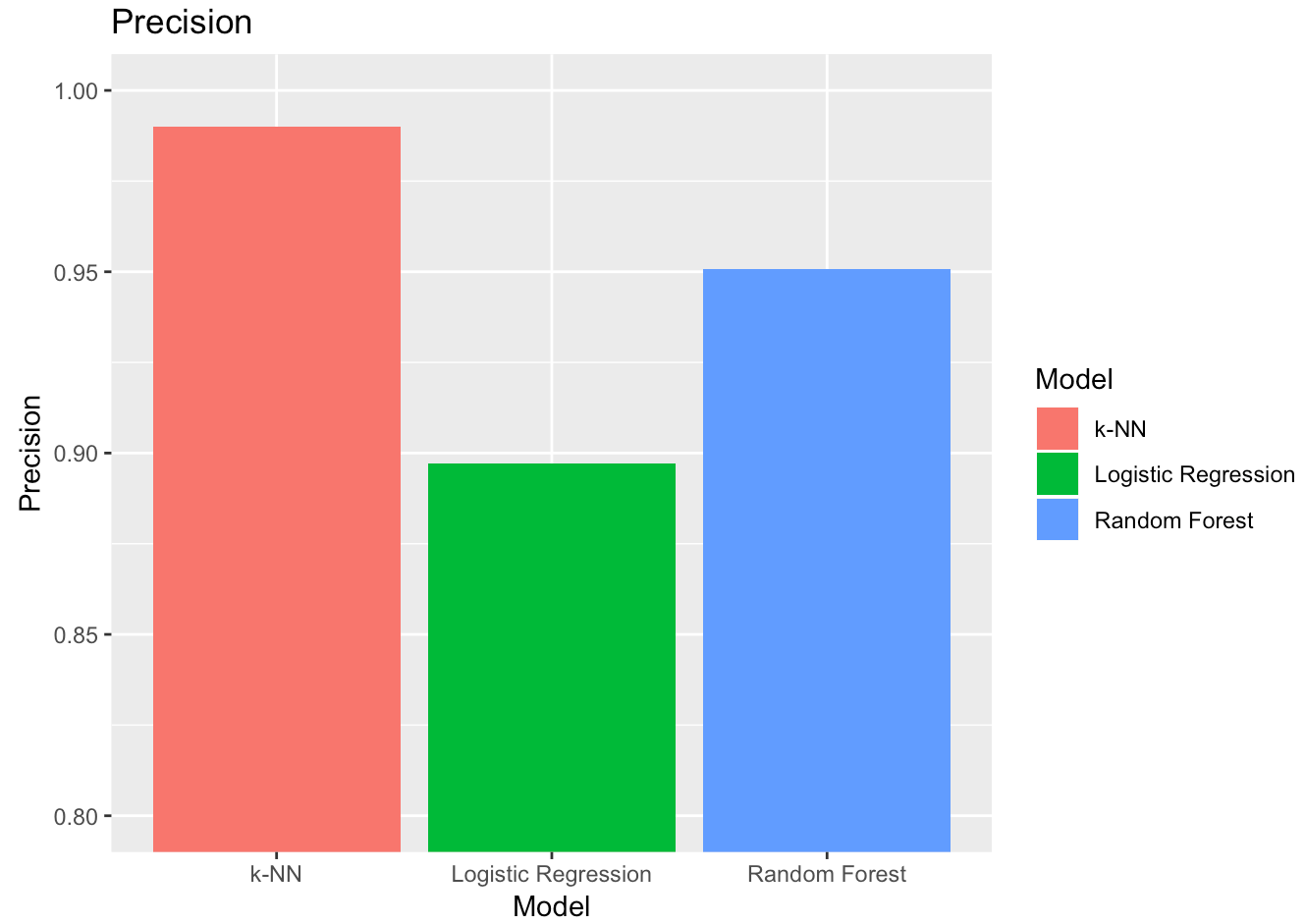
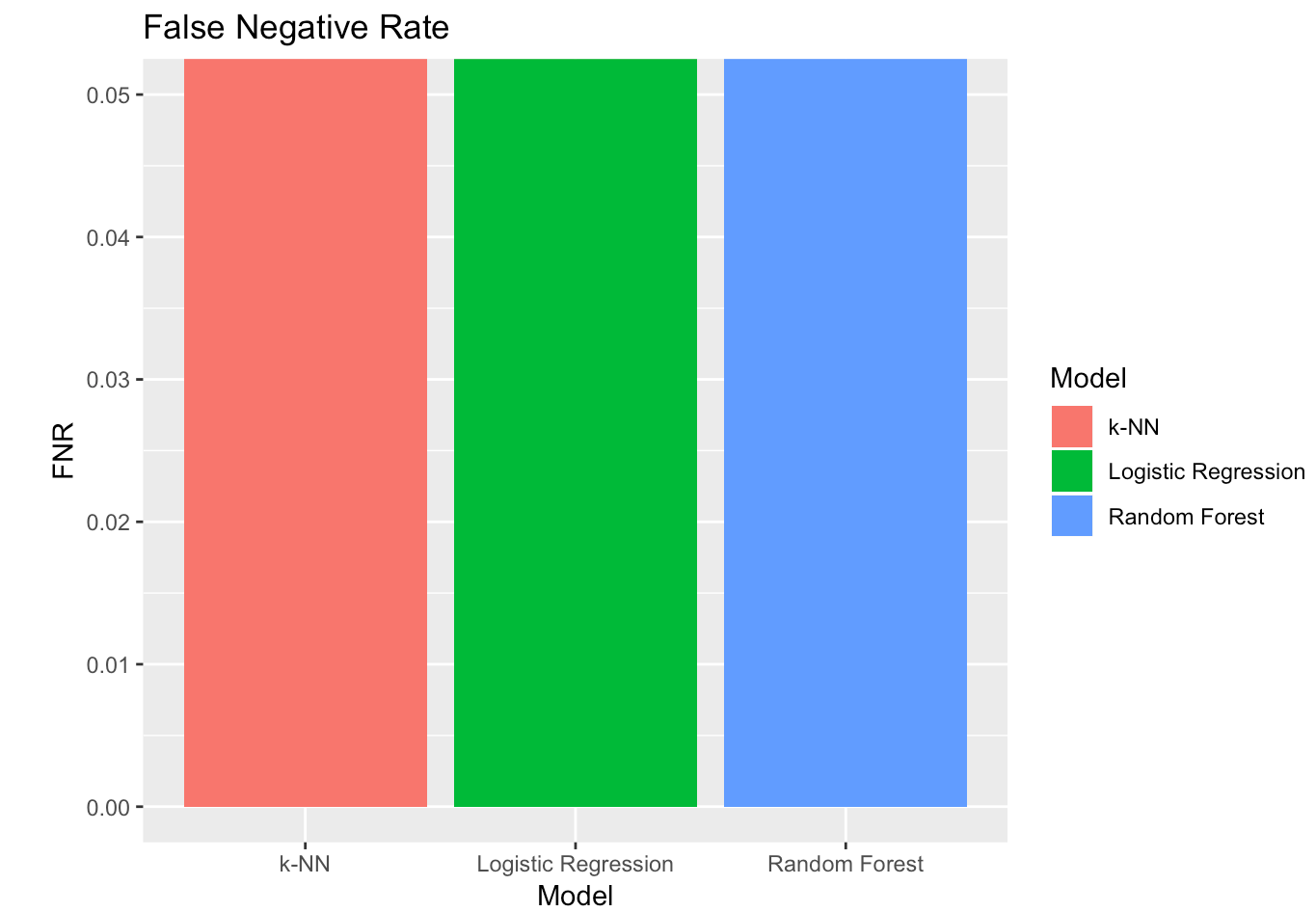
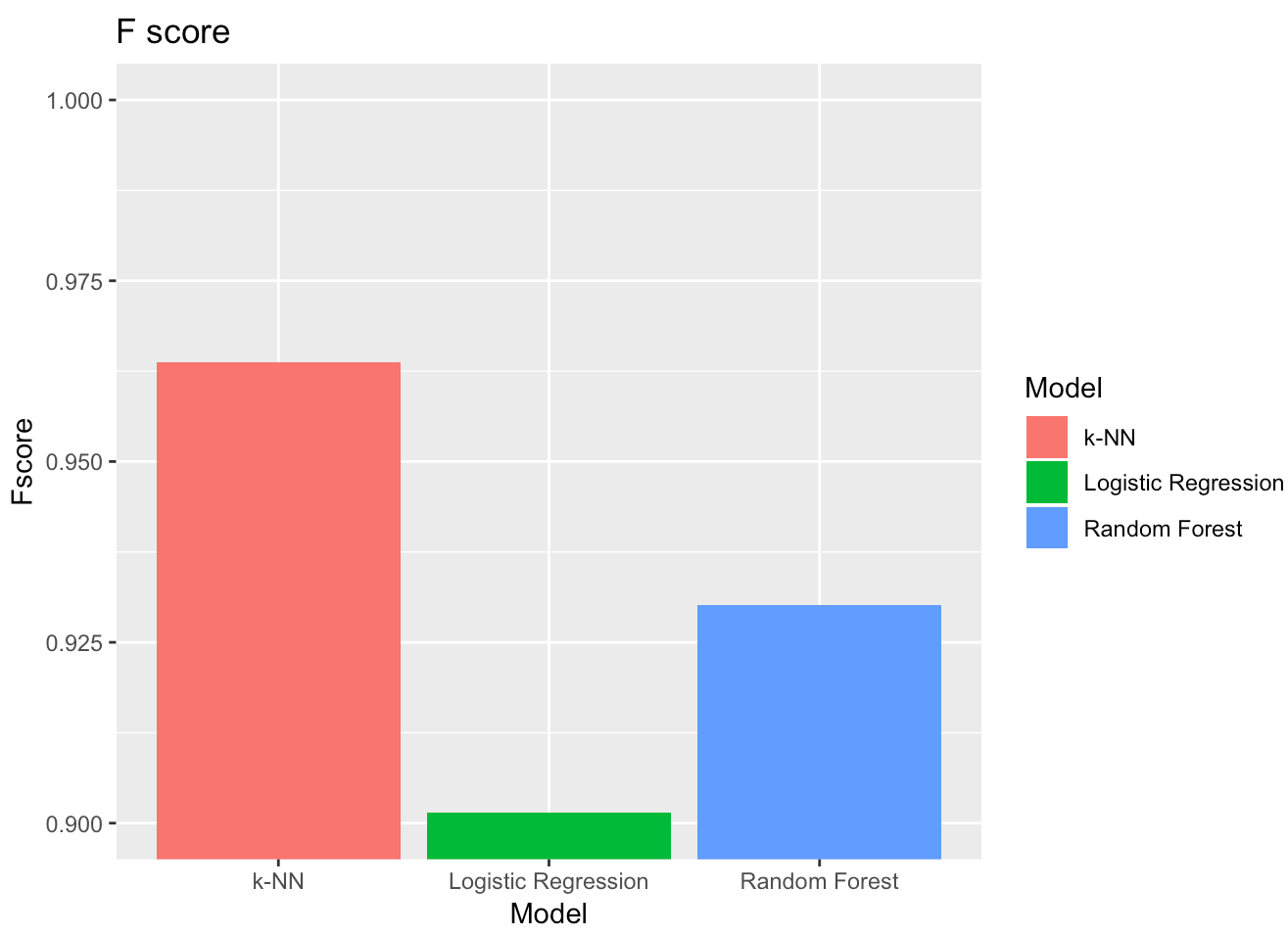


Figure B-3: Random Forest

Figures B-4: Model performance metrics

## Results:

As discussed in Objective 1, the explained variance percentage plot (Figure A-9) for PCA analysis showed that the first 10 components account for approximately 95.8% of the variance in the data. We chose to proceed with these components for tuning our models. The tuned models with PCA produced an average CV accuracy of approximately 92.6% for Logistic Regression, 97.3% for kNN, 94.9% for Random Forest.

## Conclusion:

The performance of an algorithm depends greatly upon the data set. A more powerful algorithm might not always outperform a weaker one. All 3 models performed well for classification of breast cancer. However, k-NN slightly outperformed Random Forest and Logistic Regression (in accuracy/recall/f-score). In the health domain, recall and f-score are more informative than just relying on the accuracy metric.

## Future Work:

* Exploring the difference in just using the 10 original features rather than also including the additional correlated attributes as separate features. This would significantly reduce the dimensionality of the dataset.

# APPENDIX

**Code: Loading, Cleaning, & Normalizing Data**

bc<-read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data",header=F,sep=",")

names(bc)<- c('id\_number', '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')

# Data Summary

summary(bc)

# Normalize Data

bc.clean <- bc[,-c(1)]

normalize <- function(x){

return (( x - min(x))/(max(x) -min(x)))

}

bc.clean.normalized <- as.data.frame(

lapply(bc.clean[,2:31],normalize)

)

bc.clean.normalized <- cbind(

bc.clean[,1],

bc.clean.normalized

)

names(bc.clean.normalized)[1] <- "diagnosis"

summary(bc.clean.normalized)

**Code: Box Plot Analysis**

#Box Plot: Area Mean

boxplot(area\_mean ~ diagnosis,data=wdbc,

horizontal=TRUE,

names=c("Benign","Malignant"),

col=c("green","red"),

xlab="Area Mean", main="Wisconsin Breast Cancer")

#Box Plot: Radius Mean

boxplot(radius\_mean ~ diagnosis,data=wdbc,

horizontal=TRUE,

names=c("Benign","Malignant"),

col=c("green","red"),

xlab="Radius Mean",

main="Wisconsin Breast Cancer")

**Code: Principal Component Analysis**

pc.bc<-prcomp(bc[,-c(1,2)],scale.=TRUE)

pc.bc.scores<-pc.bc$x

#Adding the response column to the PC's data frame

pc.bc.scores<-data.frame(pc.bc.scores)

pc.bc.scores$Diagnosis<-bc$diagnosis

#Use ggplot2 to plot the first few pc's

library(ggplot2)

ggplot(data = pc.bc.scores, aes(x = PC1, y = PC2)) +

geom\_point(aes(col=Diagnosis), size=1)+

ggtitle("PCA of Breast Cancer Tumor Biopsies")

ggplot(data = pc.bc.scores, aes(x = PC2, y = PC3)) +

geom\_point(aes(col=Diagnosis), size=1)+

ggtitle("PCA of Breast Cancer Tumor Biopsies")

**Code: Logistic Regression Full & VIF**

main.glm <- glm(diagnosis ~ . , data=bc.clean, family = binomial(link = "logit") , control = list(maxit = 50))

summary(main.glm)

# VIF for covariance between Radius, Perimeter, Area

vif(main.glm) -> main.glm.vif

main.glm.vif

**Code: Logistic Regression Simplified & VIF**

# REDUCED model removing all "SE"" measurements, all "Worst"", and only using "Area" in place of 'perimeter' and 'radius'

redux.glm <- glm(diagnosis ~ texture\_mean + area\_mean + smoothness\_mean + compactness\_mean + concavity\_mean + concave\_points\_mean + symmetry\_mean + fractal\_dimension\_mean , data=bc.clean, family = binomial(link = "logit") , control = list(maxit = 60))

summary(redux.glm)

# VIF for covariance between Radius, Perimeter, Area

vif(redux.glm) -> redux.glm.vif

redux.glm.vif

#95% CONFIDENCE INTERVALS

confint(redux.glm, level = 0.95)

**Code: ROC Curve for Reduced Logistic Model**

#ROC CURVE TO ASSESS

library(ROCR)

bc\_lasso\_pred <- predict(redux.glm, newx = bc.clean, type = "response")

bc.lasso.pred <- prediction(bc\_lasso\_pred, bc.clean$diagnosis)

bc.lasso.perf <- performance(bc.lasso.pred, measure = "prec", x.measure = "rec")

plot(bc.lasso.perf)

**Code: LASSO Model Selection**

library(glmnet)

#NOTE: GLMNET requires dataframe to be converted to matrix

bc\_lasso\_mat <- model.matrix(diagnosis ~ ., bc.clean)[,-1]

bc.lasso.glm <- glmnet(bc\_lasso\_mat, bc.clean$diagnosis, family = "binomial" )

bc.lasso.cv <- cv.glmnet(bc\_lasso\_mat, bc.clean$diagnosis, family = "binomial")

bc\_lambda\_lasso <- bc.lasso.cv$lambda.min

bc\_lambda\_lasso

# Output the final coefficients from GLMNET LASSO

predict(bc.lasso.cv, type = "coefficients", s = bc\_lambda\_lasso )

**Code: AIC Stepwise Feature Selection**

library(MASS)

main.glm.step <- stepAIC(

main.glm, trace = 0, family = binomial(link = "logit"), direction = "both", test="Chisq"

)

plot(main.glm.step)

summary(main.glm.step)

**Code: Plotting Logistic Regression Simplified Model – Area Mean, Texture Mean, Smoothness Mean**

# Area Mean vs Diagnosis

library(popbio)

logi.hist.plot(bc.boolean$radius\_mean, bc.boolean$diag\_bool, boxp = F, type = "hist")

# Texture Mean vs Diagnosis

library(popbio)

logi.hist.plot(bc.boolean$texture\_mean, bc.boolean$diag\_bool, boxp = F, type = "hist")

# Smoothness Mean vs Diagnosis

library(popbio)

logi.hist.plot(bc.boolean$smoothness\_mean, bc.boolean$diag\_bool, boxp = F, type = "hist")

**Code: Setting up 5-fold cross-validation**

ctrl <- trainControl(method = "cv",

number = 5)

**Code: Function for plotting confusion matrices**

cm\_plot <- function(ml, title) {

confusionMatrix(ml)$table %>%

round(1) %>%

fourfoldplot(

color = c("#CC6666", "#99CC99"),

main=title,

conf.level=0,

margin=1

)

}

**Code: Logistic Regression [Predictive w/ PCA Components]**

library(e1071)

logit.ml <- train(pc\_wdbc\_c~., full\_wdbc, method = "glm", family = "binomial", trControl =ctrl)

logit.cm <- confusionMatrix(logit.ml)

cm\_plot(logit.ml, "Logistic Regression")

logit.metrics <- data.frame (

"Model" = "Logistic Regression",

"Accuracy" = (logit.cm$table[1,1] + logit.cm$table[2,2])/100,

"Recall" = logit.cm$table[2,2] / (logit.cm$table[2,2] + logit.cm$table[1,2]),

"Precision" = logit.cm$table[2,2] / (logit.cm$table[2,1] + logit.cm$table[2,2]),

"FNR" = (logit.cm$table[1,2] / (logit.cm$table[2,2] + logit.cm$table[1,2])),

"Fscore" = (2 \* logit.cm$table[2,2]) / (2 \* logit.cm$table[2,2] + logit.cm$table[1,2] + logit.cm$table[2,1])

)

logit.metrics

**Code: k-Nearest Neighbors**

knn.ml <- train(pc\_wdbc\_c~., full\_wdbc, method = "knn", trControl =ctrl)

knn.cm <- confusionMatrix(knn.ml)

cm\_plot(knn.ml, "kNN")

knn.metrics <- data.frame (

"Model" = "k-NN",

"Accuracy" = (knn.cm$table[1,1] + knn.cm$table[2,2])/100,

"Recall" = knn.cm$table[2,2] / (knn.cm$table[2,2] + knn.cm$table[1,2]),

"Precision" = knn.cm$table[2,2] / (knn.cm$table[2,1] + knn.cm$table[2,2]),

"FNR" = (knn.cm$table[1,2] / (knn.cm$table[2,2] + knn.cm$table[1,2])),

"Fscore" = (2 \* knn.cm$table[2,2]) / (2 \* knn.cm$table[2,2] + knn.cm$table[1,2] + knn.cm$table[2,1])

)

knn.metrics

**Code: Random Forest**

rf.ml <- train(pc\_wdbc\_c~., full\_wdbc, method = "rf", trControl =ctrl)

rf.cm <- confusionMatrix(rf.ml)

cm\_plot(rf.ml, "Random Forest")

rf.metrics <- data.frame (

"Model" = "Random Forest",

"Accuracy" = (rf.cm$table[1,1] + rf.cm$table[2,2])/100,

"Recall" = rf.cm$table[2,2] / (rf.cm$table[2,2] + rf.cm$table[1,2]),

"Precision" = rf.cm$table[2,2] / (rf.cm$table[2,1] + rf.cm$table[2,2]),

"FNR" = (rf.cm$table[1,2] / (rf.cm$table[2,2] + rf.cm$table[1,2])),

"Fscore" = (2 \* rf.cm$table[2,2]) / (2 \* rf.cm$table[2,2] + rf.cm$table[1,2] + rf.cm$table[2,1])

)

rf.metrics

**Code: Model Performance - Confusion Matrices**

#Take a look at all confusion matrices:

par(mfrow=c(1,3))

cm\_plot(knn.ml, "k-NN")

cm\_plot(logit.ml, "Logistic Regression")

cm\_plot(rf.ml, "Random Forest")

**Code: Model Performance - Metrics:**

metrics1 <- rbind(knn.metrics,logit.metrics, rf.metrics)

metrics1 # Taking a look at everything together

ggplot(metrics1, aes(Model, Accuracy)) + geom\_bar(stat="identity", aes(fill=Model)) + coord\_cartesian(ylim=c(0.9,1)) + ggtitle("Accuracy")

ggplot(metrics1, aes(Model, Recall)) + geom\_bar(stat="identity", aes(fill=Model)) + coord\_cartesian(ylim=c(0.9,1)) + ggtitle("Recall")

ggplot(metrics1, aes(Model, Precision)) + geom\_bar(stat="identity", aes(fill=Model)) + coord\_cartesian(ylim=c(0.8,1)) + ggtitle("Precision")

ggplot(metrics1, aes(Model, FNR)) + geom\_bar(stat="identity", aes(fill=Model)) + coord\_cartesian(ylim=c(0,0.05)) + ggtitle("False Negative Rate")

ggplot(metrics1, aes(Model, Fscore)) + geom\_bar(stat="identity", aes(fill=Model)) + coord\_cartesian(ylim=c(0.9,1)) + ggtitle("F score")