Breast Cancer Project

Neel Kumtakar, Zenobia Dante, Shailey Rai

# **1.0 Project Objective:**

Our project focuses on a creating a well developed logistic regression model to predict and classify (based on prediction) malignant or benign tumors. The future uses of this model would be use it to classify tumors that do not have a benign or maligant label.

# **2.0 Data Description and overview:**

The dataset is called “Breast Cancer Wisconsin” and it comes from the UCI Machine Learning repository. The dataset has 569 observations and 32 variables. Each observation represents the breast of a patient.

The dataset can be viewed here: <https://www.kaggle.com/uciml/breast-cancer-wisconsin-data>

While there are 32 variables in the set, the ID variable is meaningless as it contains just the identification number of the breast examined.  There is an variable called “X” which seem to give no meaning so it is deleted from the set like ID.

Out of the 32 variables, there are 10 key features are worth examining

|  |  |  |  |
| --- | --- | --- | --- |
| Variable Number | Variable | Type | Values |
| 1 | **Diagnosis** | **Binary Categorical** | **“M” for Malignant “B” for Benign** |
| 2 | **radius** | **Continuous** | **Numeric** |
| 3 | **texture** | **Continuous** | **Numeric** |
| 4 | **perimeter** | **Continuous** | **Numeric** |
| 5 | **area** | **Continuous** | **Numeric** |
| 6 | **smoothness** | **Continuous** | **Numeric** |
| 7 | **compactness** | **Continuous** | **Numeric** |
| 8 | **concavity** | **Continuous** | **Numeric** |
| 9 | **Concave points mean** | **Continuous** | **Numeric** |
| 10 | **Symmetry** | **Continuous** | **Numeric** |
| 11 | **Fractal dimension** | **Continuous** | **Numeric** |

**Variables 2-11 has three sub variables; mean, standard error and worst. Worst is actually the largest value of the top 3 largest values in a certain category of a breast. For instance, the worst area is the largest of the top 3 largest breast areas for a certain breast.  These sub variables give in total 30 variables.**

# **3.0 Data Preprocessing**

**Missing values** – The dataset was cleansed of any observations with one or more missing values. Though the website clearly states that the dataset has no missing values, complete.cases() function was used just to make sure.

**Binary Variable** – There is a categorical variable called “Diagnosis” which identifies a breast tumor to be malignant or benign. This categorical variable was transformed into a binary discrete variable, with 0 to represent benign tumors and 1 to represent malignant tumors

**Irrelavant Variables –** Variables such as ID, and X are irrelavant variables and thus were removed from the dataset.

# **4.0 Methodology:**

For this project we will implement the following methodology.

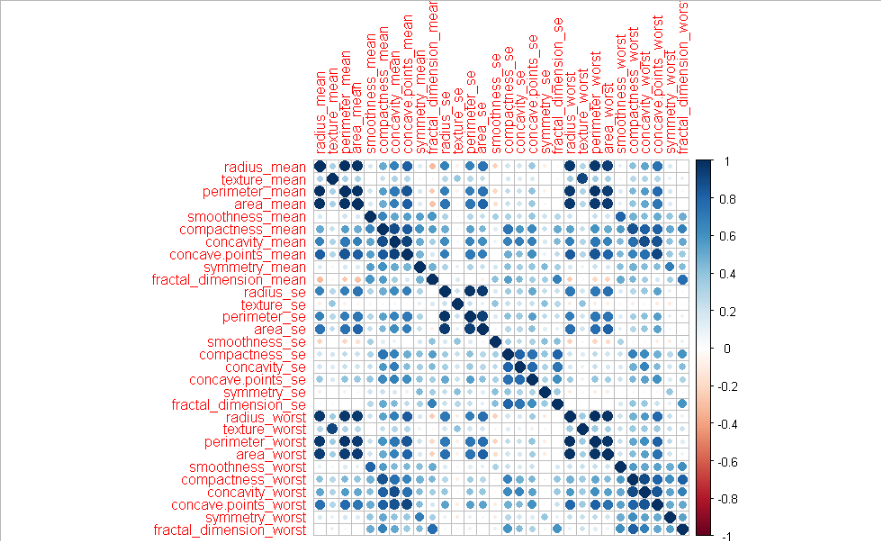
1. Variable Reduction: Analyse multicollinearity in variables other than the dependent, “Diagnosis” using a corrplot especially for a large amount of variables. Take out variables with high correlation pairs with one another.
2. Data Split: Split the data into training and testing resepectively
3. Model Fitting: Fit an orgiinal model with variables from Step 1 usign the training set in Step 2.
4. Cuttoff Determination: Use a line graph that plots the range of cuttoff values from [0,1] to determine the optimal cuttoff. Generate two graphs for accuracy and profit (mentioned later)
5. Model performance analysis: Make the model predict on the Test set using the cuttoff value in Step 4 and compare predictions against the Testing’s Diagnosis variable
6. Select the Best Stepwise, N best, and Add One Models (3 Models)
   1. For Stepwise, select the technique that generates the model with the lowest AIC and Residual Deviance
   2. For N best- For the n given amount of models, use AIC and BIC to select the best models (best 3). Then select the model with the lowest AIC and Residual Deviance
   3. For Add One – Start with an empty model and Select the variable with the greatest LRT value and coincidentally the least p value. Then see if the model has p values less than 0.05. Repeat the process until you can’t add any more variables to your model meaning at least one p value is greater than 0.05.
7. For the best Stepwise, Nbest and Add One Drop One models, repeat steps 3-5.
8. Select the final model out of the 3 models based on the F1 Measure, Profit, and Accuracy metrics. Use a scoring criteria if needed.

**Writer’s advice: When you are comparing models over k amount of criteria, and you are getting different conclusions for each of the metrics you are analysing, it’s important to use a scoring metric.**

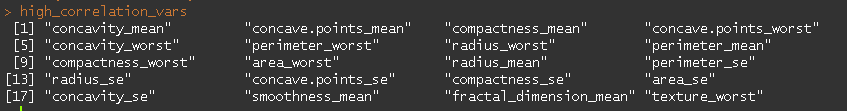
**The general formula for a score metric is a weighted average where each variable is given weight. In other words,**

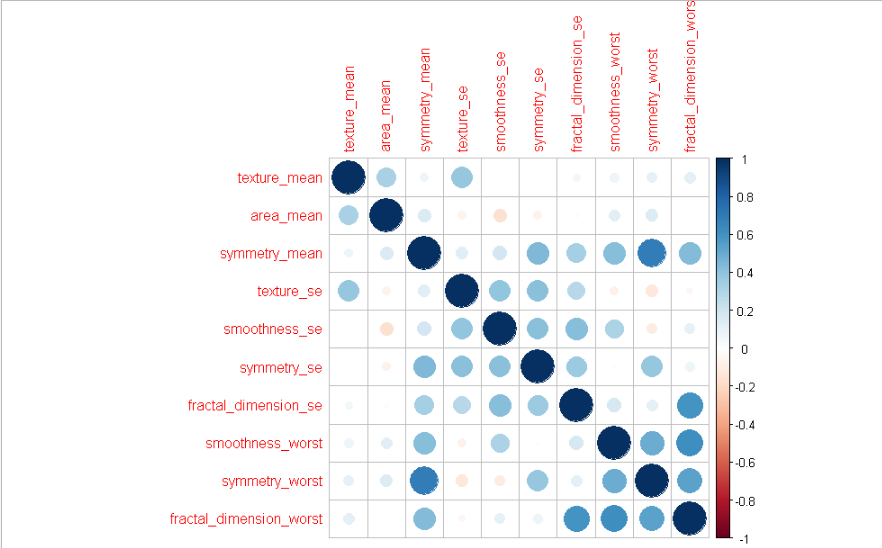
# **5.0 Variable Reduction: Multicollinearity & Correlated Variables**

We start by analysing all the variables in the set except the dependent variarble, radius\_mean. Since there is a lot of variables, we use the corrplot() to visualize the correlation matrix.



Dummies (Wiley) shows that a variables with pair wise correlation of 0.7 and higher are deemed highly correlated so the following variables below are deemed highly correlated. So we remove these variables from the set.





We are left with 10 variables, and above is the corrplot visualization of the correlation matrix of these variables. As shown below, none of these correlation pairs have correlation values above 0.7 which is good implying we don’t have high correlations in the dataset. However, it is important to note that we still have correlations between 0.5 and 0.7 for the most part, but it shouldn’t be of concern, as this would be a medium sized correlation size.

> findCorrelation(cor(Breast\_Cancer\_DT[,-1]), cutoff = 0.7, names=TRUE)

character(0)

Given the 10 variables shown in the corrplot, ths logistic regression equation (original) will be of the following form.

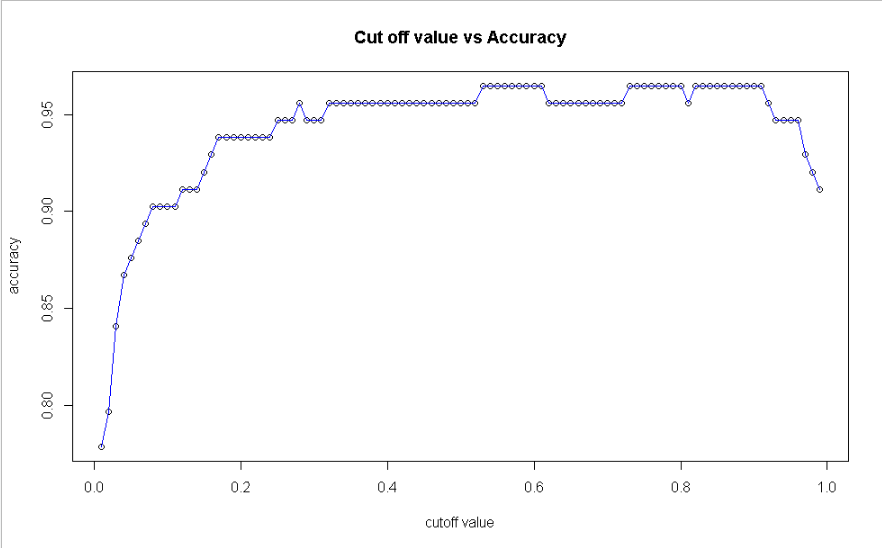
# **6.0 Model Creation and Implementation**

## **6.1 Original Logistic Model**

The logistic regression model is a probabilitic model. Unlike the regular multi regression model that has a y value that is continous, this model’s dependent variable is discerte and categorical. For this model, we need to evaluate a list of cutoff values and pick the one

**Accuracy analysis:**

Accuracy is defined by the numbers of correct classfiications the model makes on the testing set. The below graph shows the cuttoff values listering from 0 to 1 by 0.01 increments. We can see that at some cutoff value, c, when that the accuracy is the highest



> which.max(accuracy) \* 0.01

[1] 0.53

**The following line above shows that 0.53 is the ideal cuttoff value for the highest accuracy**

**Profit analysis:**

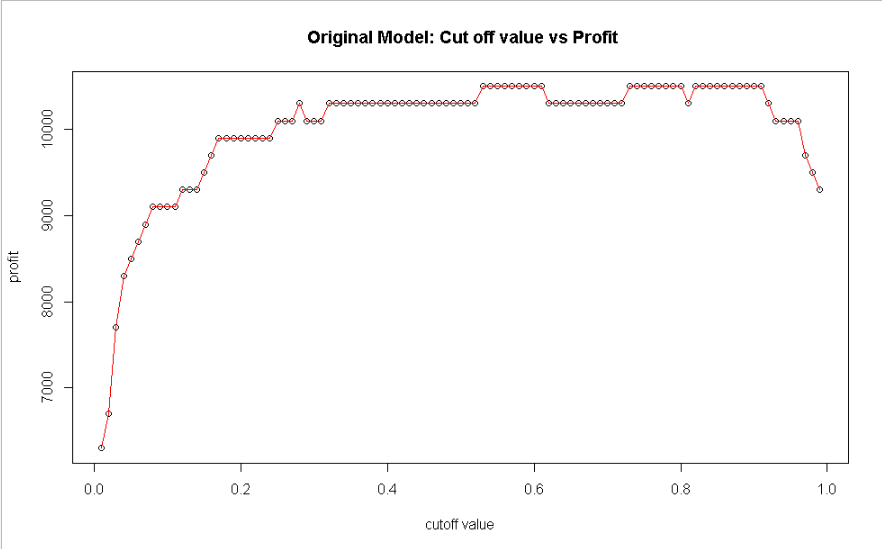
The model classification power is divided into 4 categories: True Positives, False Negatives, and True Negatives.

Information about these categories can be found here:

Suppose we want to reward the model for getting correct classification and punish the model for incorrect classifications.

Such a formula can be expressed as such:

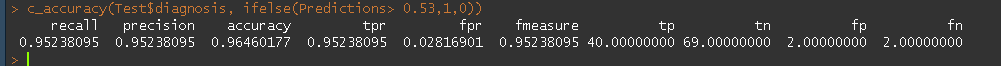
Suppose we want to reward correct classifications, i.e TN, and TP with $100, and let’s say we want to punish by taking away $100 for incorrect ones like FN and FP. The resulting formula is below.



> which.max(profit) \* 0.01

[1] 0.53

We still notice that 0.53 is the ideal cutoff point for producing the highest cutoff value. It’s great that in both the profit and the accuracy cases, the cutoff points are the same.



As indicative of above, this logistic model is pretty good. Below are the summary points that explain why.

1. **Accuracy- The model has got 96% of the observations right in the test set. This means that only 4% of the observations are incorrect**
2. **Profit- The profit computed from the TP, TN, FP and FN cases comes to about $10500. The test set contains 113 observations so a perfect model would yield $11300. This is about a 7% money loss.**
3. **Precision and Recall – Precision and Recall are at 95%. This means that model has got 95% of all positive classiications (TP + FP) correct and 95% of all negative classifications (TN+ FN) correct**

**This model is good and should be considered in the list of ideal models explored in the later sections of this paper.**

**Model selection Methods:**

**For this section we will implement the following Model Selection Methods**

1. **Add one / Drop one- Add and dropping variables one at a time.**
2. **Selection Methods – Backwards, fowards, stepwise**
3. **Regsubset – Seleecting the n best from each subset**

## **6.2 Add 1, Drop 1 selection Method**

**We start the process off with an empty model. We see that area\_mean has the greatest LRT and one of the lowest P values, so we add the variable to our model.**

> add1(empty,scope=full,test="LRT")

Single term additions

Model:

diagnosis ~ 1

Df Deviance AIC LRT Pr(>Chi)

<none> 751.44 753.44

texture\_mean 1 646.52 650.52 104.92 < 2.2e-16 \*\*\*

area\_mean 1 325.66 329.66 425.78 < 2.2e-16 \*\*\*

symmetry\_mean 1 686.80 690.80 64.64 8.974e-16 \*\*\*

texture\_se 1 751.40 755.40 0.04 0.84280

smoothness\_se 1 748.79 752.79 2.65 0.10344

symmetry\_se 1 751.42 755.42 0.02 0.87621

fractal\_dimension\_se 1 748.06 752.06 3.38 0.06603 .

smoothness\_worst 1 641.42 645.42 110.02 < 2.2e-16 \*\*\*

symmetry\_worst 1 641.42 645.42 110.02 < 2.2e-16 \*\*\*

fractal\_dimension\_worst 1 689.39 693.39 62.05 3.348e-15 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

**This variable has a p value less than 0.05, so we continue to add the next variable.**

Model:

diagnosis ~ area\_mean

Df Deviance AIC LRT Pr(>Chi)

<none> 325.66 329.66

area\_mean 1 751.44 753.44 425.78 < 2.2e-16 \*\*\*

**After we put area\_mean, we notice that smoothness\_worst has the greatest LRT so we select that variable.**

Model:

diagnosis ~ area\_mean

Df Deviance AIC LRT Pr(>Chi)

<none> 325.66 329.66

texture\_mean 1 288.98 294.98 36.681 1.391e-09 \*\*\*

symmetry\_mean 1 263.25 269.25 62.409 2.791e-15 \*\*\*

texture\_se 1 319.82 325.82 5.841 0.0156601 \*

smoothness\_se 1 313.38 319.38 12.272 0.0004599 \*\*\*

symmetry\_se 1 319.10 325.10 6.558 0.0104397 \*

fractal\_dimension\_se 1 315.61 321.61 10.048 0.0015248 \*\*

smoothness\_worst 1 177.68 183.68 147.980 < 2.2e-16 \*\*\*

symmetry\_worst 1 221.65 227.65 104.002 < 2.2e-16 \*\*\*

fractal\_dimension\_worst 1 215.63 221.63 110.025 < 2.2e-16 \*\*\*

---

**The p values are less than 0.05 for the two variables, so we continue the process.**

Model:

diagnosis ~ smoothness\_worst + area\_mean

Df Deviance AIC LRT Pr(>Chi)

<none> 177.68 183.68

smoothness\_worst 1 325.66 329.66 147.98 < 2.2e-16 \*\*\*

area\_mean 1 641.42 645.42 463.75 < 2.2e-16 \*\*\*

---

Model:

diagnosis ~ smoothness\_worst + area\_mean

Df Deviance AIC LRT Pr(>Chi)

<none> 177.68 183.68

texture\_mean 1 139.33 147.33 38.350 5.913e-10 \*\*\*

symmetry\_mean 1 172.53 180.53 5.141 0.023361 \*

texture\_se 1 170.02 178.02 7.652 0.005670 \*\*

smoothness\_se 1 175.03 183.03 2.645 0.103856

symmetry\_se 1 175.65 183.65 2.023 0.154912

fractal\_dimension\_se 1 177.68 185.68 0.000 0.996761

symmetry\_worst 1 158.48 166.48 19.196 1.179e-05 \*\*\*

fractal\_dimension\_worst 1 169.31 177.31 8.369 0.003817 \*\*

---

Model:

diagnosis ~ smoothness\_worst + area\_mean + texture\_mean

Df Deviance AIC LRT Pr(>Chi)

<none> 139.33 147.33

smoothness\_worst 1 288.98 294.98 149.65 < 2.2e-16 \*\*\*

area\_mean 1 533.60 539.60 394.28 < 2.2e-16 \*\*\*

texture\_mean 1 177.68 183.68 38.35 5.913e-10 \*\*\*

---

Model:

diagnosis ~ smoothness\_worst + area\_mean + texture\_mean

Df Deviance AIC LRT Pr(>Chi)

<none> 139.33 147.33

symmetry\_mean 1 130.29 140.29 9.0401 0.002641 \*\*

texture\_se 1 139.28 149.28 0.0505 0.822263

smoothness\_se 1 133.62 143.62 5.7079 0.016888 \*

symmetry\_se 1 136.47 146.47 2.8525 0.091233 .

fractal\_dimension\_se 1 139.21 149.21 0.1130 0.736727

symmetry\_worst 1 119.98 129.98 19.3434 1.092e-05 \*\*\*

fractal\_dimension\_worst 1 132.52 142.52 6.8069 0.009081 \*\*

Model:

diagnosis ~ smoothness\_worst + area\_mean + texture\_mean + symmetry\_worst

Df Deviance AIC LRT Pr(>Chi)

<none> 119.98 129.98

smoothness\_worst 1 182.61 190.61 62.63 2.496e-15 \*\*\*

area\_mean 1 491.95 499.95 371.97 < 2.2e-16 \*\*\*

texture\_mean 1 158.48 166.48 38.50 5.483e-10 \*\*\*

symmetry\_worst 1 139.33 147.33 19.34 1.092e-05 \*\*\*

Model:

diagnosis ~ smoothness\_worst + area\_mean + texture\_mean + symmetry\_worst

Df Deviance AIC LRT Pr(>Chi)

<none> 119.98 129.98

symmetry\_mean 1 119.88 131.88 0.10377 0.7474

texture\_se 1 118.05 130.05 1.93098 0.1647

smoothness\_se 1 118.81 130.81 1.17760 0.2778

symmetry\_se 1 119.38 131.38 0.60583 0.4364

fractal\_dimension\_se 1 119.07 131.07 0.91490 0.3388

fractal\_dimension\_worst 1 119.68 131.68 0.30342 0.5817

**Adding Texture\_Se causes the p values to rise above 0.05. We discard this variable and go back to the previous model.**

Model:

diagnosis ~ smoothness\_worst + area\_mean + texture\_mean + symmetry\_worst

Df Deviance AIC LRT Pr(>Chi)

<none> 119.98 129.98

symmetry\_mean 1 119.88 131.88 0.10377 0.7474

texture\_se 1 118.05 130.05 1.93098 0.1647

smoothness\_se 1 118.81 130.81 1.17760 0.2778

symmetry\_se 1 119.38 131.38 0.60583 0.4364

fractal\_dimension\_se 1 119.07 131.07 0.91490 0.3388

fractal\_dimension\_worst 1 119.68 131.68 0.30342 0.5817

Model:

diagnosis ~ smoothness\_worst + area\_mean + texture\_mean + symmetry\_worst

Df Deviance AIC LRT Pr(>Chi)

<none> 119.98 129.98

smoothness\_worst 1 182.61 190.61 62.63 2.496e-15 \*\*\*

area\_mean 1 491.95 499.95 371.97 < 2.2e-16 \*\*\*

texture\_mean 1 158.48 166.48 38.50 5.483e-10 \*\*\*

symmetry\_worst 1 139.33 147.33 19.34 1.092e-05 \*\*\*

Using the add1 drop1 model selection process, we end up with the following 4 variables: smoothness\_worst, area\_mean, texture\_mean, symmetry\_worst.

**ACCURACY Analysis**

Accuracy is defined by the numbers of correct classifications the model makes on the testing set. The below graph shows the cuttoff values from 0 to 1 by 0.01 increments. We can see that at some cutoff value, c, when 𝒄 ≅𝟎.7 that the accuracy is the highest .

A close up of a map

Description automatically generated

The result below shows that 0.69 is the ideal cut-off value for the highest accuracy.

A picture containing knife

Description automatically generated

**PROFIT Analysis**

**The model classification power is divided into four categories – True Positive,   
False Negative, False Positive and True Negative.**

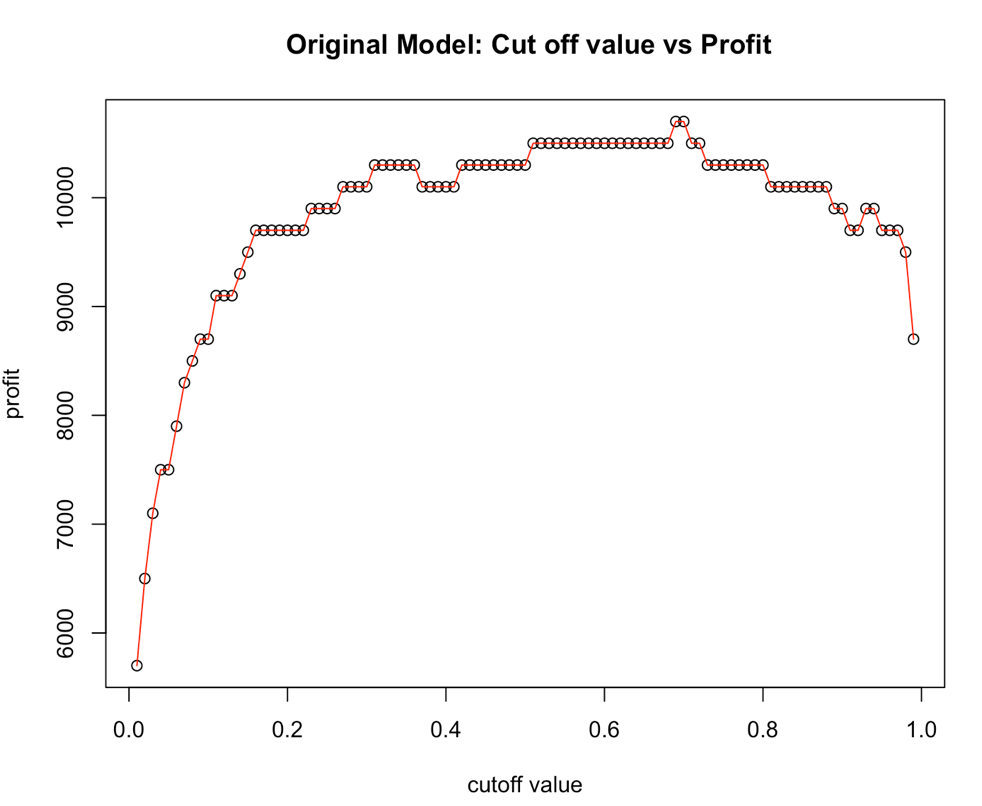
Suppose we want to reward the model for getting correct classification and punish the model for incorrect classifications.

Such a formula can be expressed as:

𝑵𝒆𝒕 𝑷𝒓𝒐𝒇𝒊𝒕=𝑷𝒓𝒐𝒇𝒊𝒕−𝑳𝒐𝒔𝒔

Suppose we want to reward correct classifications, i.e TN, and TP with $100, and let’s say we want to punish by taking away 100 for incorrect ones like FN and FP.  The resulting formula is below.

𝑵𝒆𝒕 𝑷𝒓𝒐𝒇𝒊𝒕=1𝟎𝟎 (𝑻𝑷+𝑻𝑵)−𝟏𝟎𝟎 (𝑭𝑷+𝑭𝑵)





The result above shows that 0.69 is still the ideal cut-off point for producing the highest cutoff- value. It is great that in both the profit and the accuracy cases, the cutoff points are the same.

A picture containing knife

Description automatically generated

As indicative of above, our model is pretty good. Below are the summary points that explain why.

1. **Accuracy**- The model has got 97% of the observations right in the test set.  This means that only 3% of the observations are incorrect
2. **Profit-** The profit computed from the TP, TN, FP and FN cases comes to about $22000.  The test set contains 113 observations so a perfect model would yield $22600. This is about a 3% money loss.
3. **Precision and Recall** – Precision and Recall are at 97% and 95%, respectively. This means that model has got 97% of all positive classifications (TP + FP) correct and 95% of all negative classifications (TN+ FN) correct.

**This model is good and should be considered in the list of ideal models explored in the later sections of this paper.**

## **6.3 Stepwise Selection Methods (Backwards, Forewards, and Stepwise)**

We will be using the following model selection techniques for this section: backwards, foreward and stepwise selection.

We run all the techniques, get the final models for each technique, using the full dataset as input and compare the AIC and Residual Deviance.

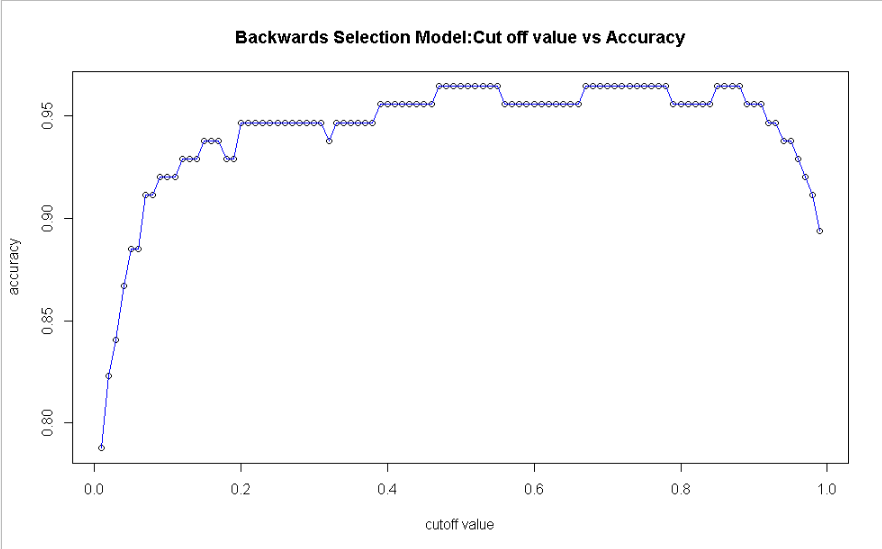
|  |  |  |
| --- | --- | --- |
| Technique | AIC | Residual Deviance |
| Forward | 130 | 120 |
| Backward | 127.8 | 111.8 |
| Stepwise | 130 | 120 |

When we compare the three models we see that the Backwards selection model is the ideal model since the AIC and the Residual Deviance are at the lowest



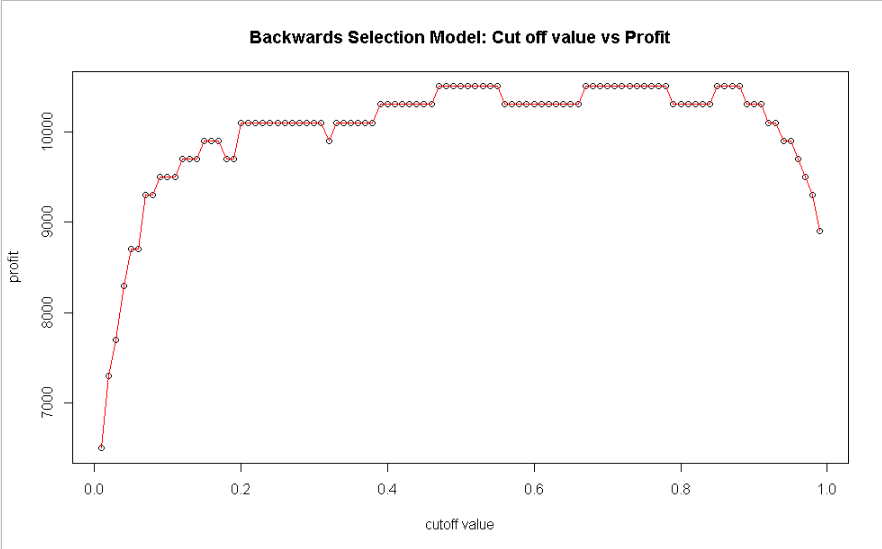
We then fit the backwards model using the training set and then generate the cuttoff graphs for both profit and accuracy with respect to the testing set.

From the graphs we can see that 0.47 is the optimal cuttoff value we should use to classify the probabilities as 0 or 1.



> which.max(accuracy) \*0.01

[1] 0.



> which.max(profit) \* 0.01

[1] 0.47

We can then classify and predict on the testing set using the optimal cuttoff value. After that we compare the predictions with the testing set’s Diagnosis values.

We get the following:

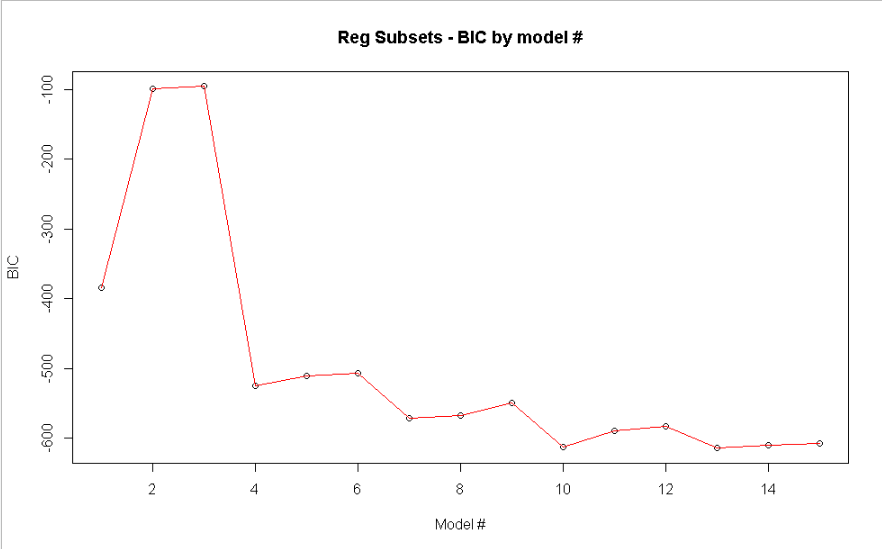


* **Accuracy**- The model has got 96% of the observations right in the test set.  This means that only 4% of the observations are incorrect
* **Profit-** The profit computed from the TP, TN, FP and FN cases comes to about $10500.  The test set contains 113 observations so a perfect model would yield $11300 This is about a 7% money loss.
* **Precision and Recall** – Precision and Recall are at 95% and 95%, respectively. This means that model has got 95% of all positive classifications (TP + FP) correct and 95% of all negative classifications (TN+ FN) correct.

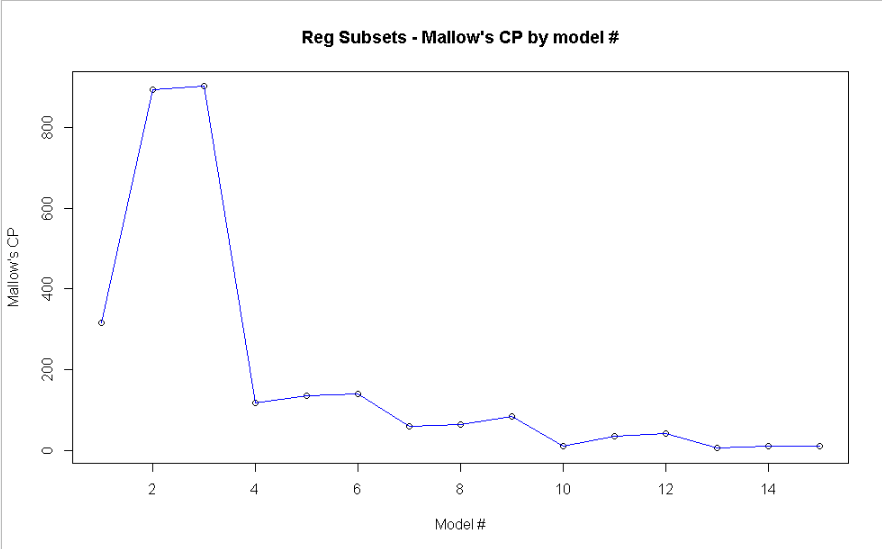
## **6.4 Regsubset / N best selection Method**

For this method, we want to choose the top 3 best models from each subset with each model having a max of 5 variables. In short, we get 15 diffferent models in total.

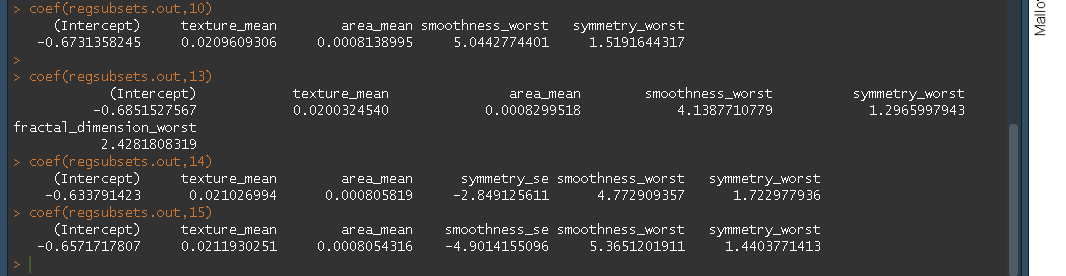
We compare the 15 models using first the BIC or Bayesian Information Criterion. A lower BIC indicates betterf fit, so we should look at models that fit this criteria. Observing the graph, we can see that graphs 10 and 13-15 are the the ones that have the lowest BIC crtierion.



Now let’s compare the model’s over Mallow’s CP



For Mallow’s CP, a lower value is better like BIC so we should look for models with the lowest CP. The models that fit this criteria are models #10, 13-15.



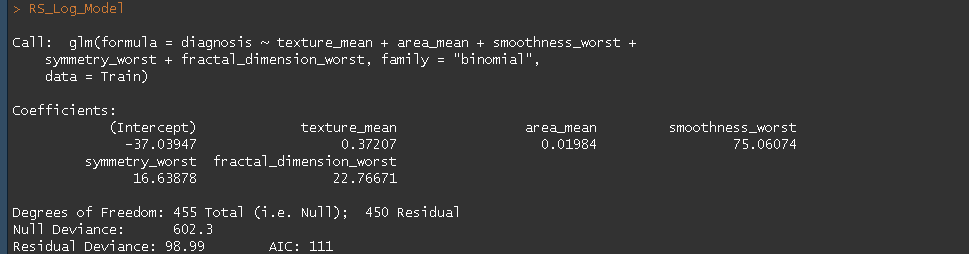
We fit each of the models using the full dataset before splitting. We get the following table as shown in the next page.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model # | Predictors | Residual Dev | AIC | Score: |
| Model # 10 | **Texture\_mean, area\_mean, smoothness\_worst,**  **Symmetry\_worst** | 120 | 130 | 125 |
| Model #13 | **Texture\_mean,**  **Area\_mean,**  **Smoothness\_worst,**  **Symmetry\_worst,**  **Fractal Dimension worst** | 119.7 | 131.7 | **125.7** |
| Model #14 | **Texture Mean**  **Area Mean**  **Symmetry SE**  **Smoothness worst**  **Symmetry Worst** | 119.4 | 131.4 | 125.4 |
| Model #15 | Texture Mean  Area Mean  Smoothness SE  Smoothness Worst  Symmetry\_worst | 118.8 | 130.8 | 124.8 |

Model #13 is marginally the winner of the 4 models as it has the highest score computed by a weighted average with weight of 0.50. In other words,

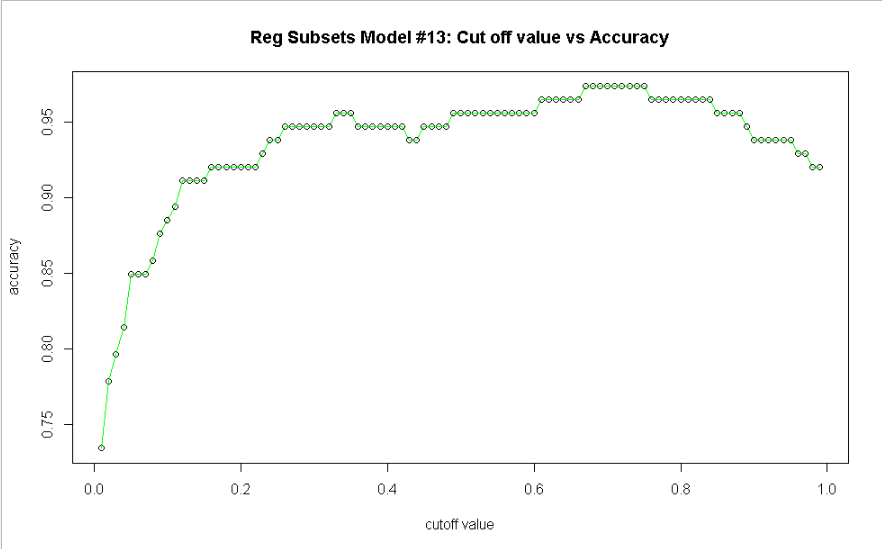
A scoring variable was needed as different conclusions were drawn from the Residual Deviance and AIC. For instance, Model #15 had the lowest Residual Deviance, yet Model #10 had the lowest AIC.

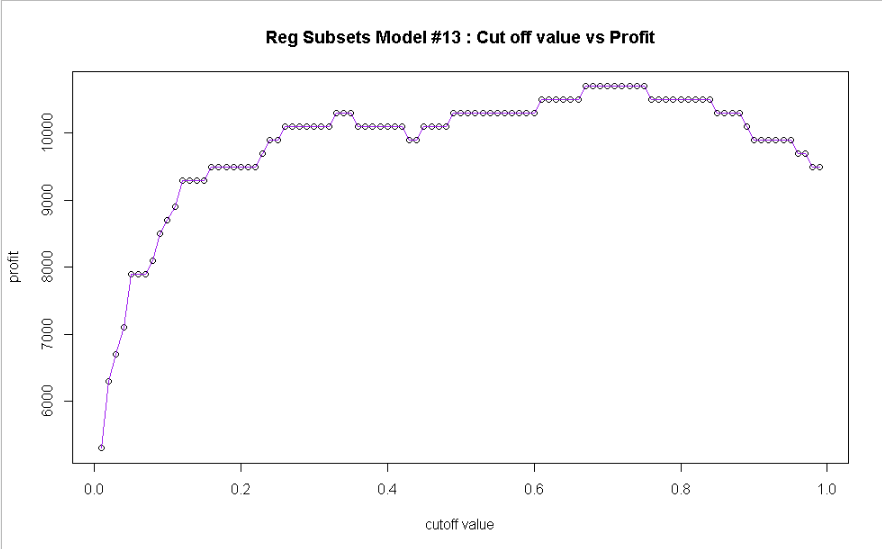
Once we have chosen Model #13, we fit it with the training set



As before our model will generate probabilities and it’s up to us to generate the appropriate cutoff value to classify the probabilities as 0 or 1 in terms of the diagnosis variable.

We run a cutoff graph for both accuracy as well as profit and we see that the optimal cutoff value is 0.67. That is, probabilities that are greater than or equal to 0.67 are categorized as 1 and those lower are categorized as 0.





> which.max(accuracy) \*0.01

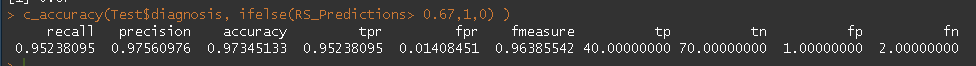
[1] 0.67

>

> which.max(profit) \* 0.01

[1] 0.67

From the cutoff value we predict and classify on the testing set using the optimal cuttoff value and compare with the testing set’s Diagnosis values.



* **Accuracy**- The model has got 97% of the observations right in the test set.  This means that only 3% of the observations are incorrect
* **Profit-** The profit computed from the TP, TN, FP and FN cases comes to about $10700.  The test set contains 113 observations so a perfect model would yield $11300. This is about a 5% money loss.
* **Precision and Recall** – Precision and Recall are at 97% and 95%, respectively. This means that model has got 97% of all positive classifications (TP + FP) correct and 95% of all negative classifications (TN+ FN) correct.

## **7.0 Conclusion:**

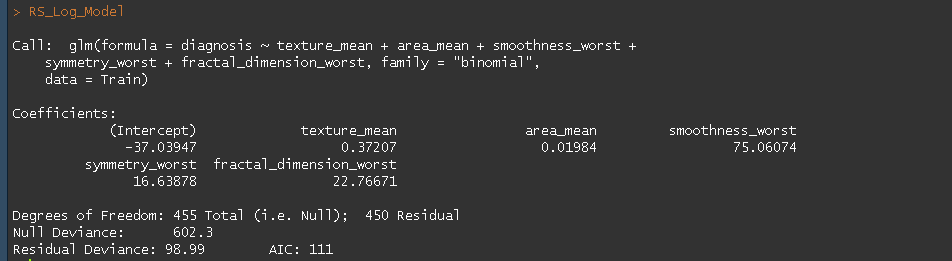
We review our 4 main models using 3 main criteria. The F1 measure which is a balance of Precision and Recall. Profit which rewards correct classification and punishes incorrect classification. Accuracy which is a proportion of correct classfications to total classfications.

Below is a quick recap of important formulas.

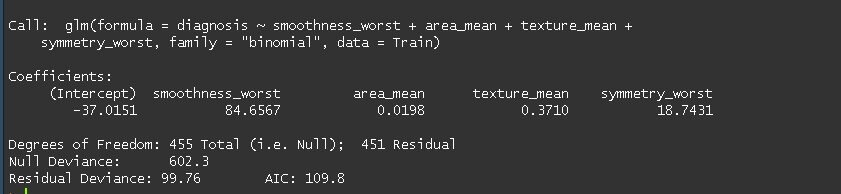
|  |  |  |  |
| --- | --- | --- | --- |
| Model | F1 Measure (Balances Precision and Recall) | Profit  (Penalty and reward for correct and incorrect classifications) | Accuracy |
| Original Model | 0.952 | $10500 | 0.965 |
| Backwards Step wise model | 0.952 | $10500 | 0.964 |
| Reg Subsets Model #13 | 0.964 | $10700 | 0.973 |
| Add One, Drop one Model | 0.964 | $10700 | 0.973 |

We see that the reg subset Model #13 and the Add one Drop Model have not only the same metric values but the greatest F1 measure, Profit and Accuracy.

Reg subsets Model #13: (5 predictors)



**Add One Drop 1 model (4 predictors)**



**When we take a closer look at these two models we see the reg subsets model has one extra variable, fractal\_dimension\_worst than the Add One Drop One Model. Additionally, the coefficients for most of the intersecting variables of both models are nearly the same. The residual deviances and AIC values do not differ by much.**

**In conclusion, if one were to go with either model, the predictions should similar.**

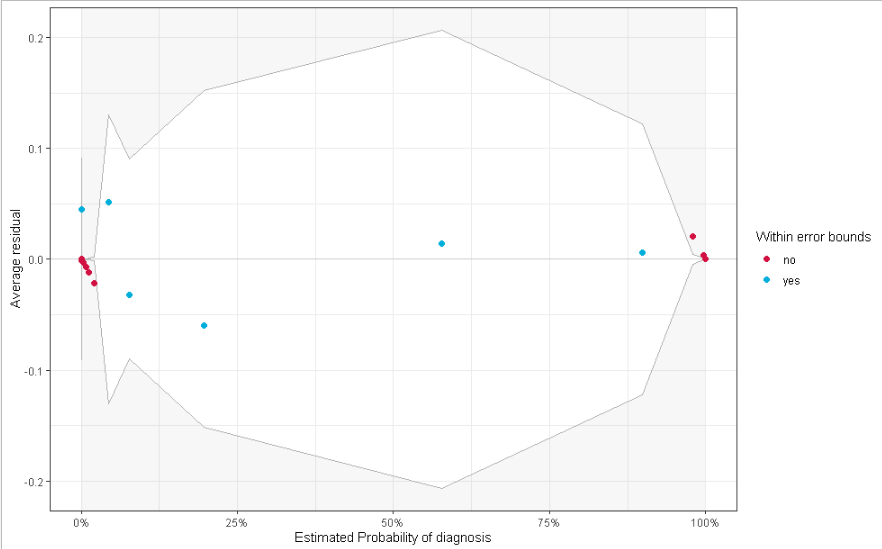
**We will want to go with the model with a slightly larger number of predictors so the reg subsets model is the ideal model.**

## **7.1 Residual Analysis:**

**For analyzing our final model, we use a binned residual plot. Binned Residual plots are defined as the following according to the R documentation website:**

“Binned residual plots are achieved by “dividing the data into categories (bins) based on their fitted values, and then plotting the average residual versus the average fitted value for each bin.” (Gelman, Hill 2007: 97). If the model were true, one would expect about 95% of the residuals to fall inside the error bounds.” (R documentation)

binned\_residuals(Final\_Model)



Warning: Probably bad model fit. Only about 29% of the residuals are inside the error bounds.

According to the warning, our model seems to a bad fit however, our metrics like Accuracy, F1, Precision etc which have been maximized thanks to our careful selection of our cutoff value tell a different story. Hence, we should ignore our residual analysis and go with the final model to predict new and unknown values.

References:

Wolberg, W. H., Street, W. N., & Mangasarian,, O. L. (2016). [Breast Cancer Wisconsin (Diagnostic) Data Set] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

Repository: Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

Schenider, Matthew J, Logistic Regression Cut off code, (Feb 2020), Drexel Lebow College of Business

##### [Daniel Lüdecke](https://www.rdocumentation.org/collaborators/name/Daniel%20L%C3%BCdecke), (R documentation) .”binned residuals: Binned Residuals For Logistic Regression”

” https://www.rdocumentation.org/packages/performance/versions/0.4.8/topics/binned\_residuals