Breast Cancer Diagnosis Classification Analysis

AUSTIN'S TEMPLATE

December 9th, 2022

Contents

1 Introduction (Should we do an Abstract?)

1.1 Background

In our analysis, we will be using the Breast Cancer Wisconsin (Diagnostic) Data Set from Kaggle. This data set contains `569 observations` of breast cancer cells with `32 variables` describing each cell. Some of the variables include characteristics like `radius`, `texture`, `area`, `perimeter` of the cell.

The goal of our analysis is to predict whether a cell is benign or malignant based on the `32 variables`. A cell is considered benign if it is not cancerous and malignant if it is cancerous. Normally in most machine learning models, we do our best to train the model to reduce the overall error rate. While this is an important goal, our group was more concerned with the `Type-II error rate`. By reducing the `Type-II error rate`, we can ensure that we are not making the mistake of classifying a malignant cell as benign. This is important in the world of Oncology because if a malignant cell is classified as benign, it could lead to a patient not

receiving the proper treatment. Whereas if a benign cell is classified as malignant, the patient may be alarmed, but a false alarm is better than a missed diagnosis.

In order to achieve our goal, we conducted the following steps:

1. Data Cleaning
2. Explatory Data Analysis
3. Classification Analysis
4. Regression Analysis
5. Overall Analysis Summary

1.2 Data

In order to properly train and test our models, we first had to split the data into a training and test set. We decided to use a `60/40 split` for our training and test data. This allows us to allocate more data to the training set, which will allow us to train our models more effectively. We also decided to remove the `ID` column from the data set because it was not relevant to our analysis.

1.3 Variable Descriptions

*Insert text here*

2 Explatory Data Analysis

To begin our analysis, we first wanted to get a better understanding of the data so we could properly prepare it for our models. As you can see in Figure A below, we have more information regarding the benign cells than the malignant cells. To be more specific, we have 357 benign cells and 212 malignant cells. Since we are worried about the `Type-II error rate`, we would want more information regarding the malignant cells in an ideal world. However since this is not the case, is important to note because it could lead to a bias in our model if we do not take this into account.

Figure A: Malignant vs Benign Cell Counts
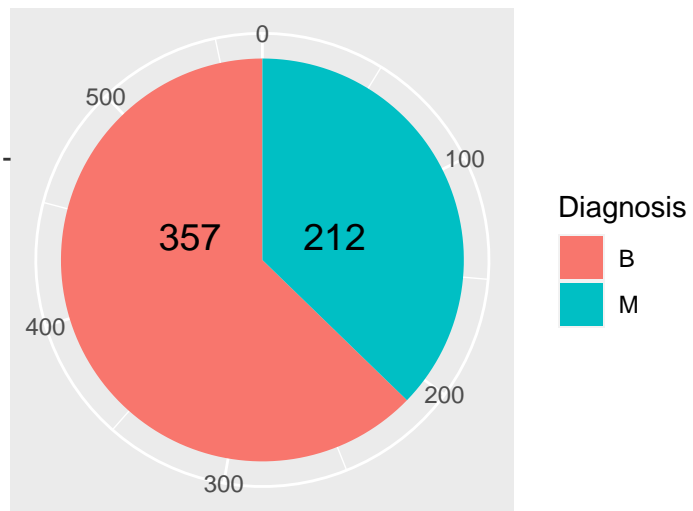
Insert text here regarding the GGPairs Plot

Figure B: Pairwise Scatterplots of Features
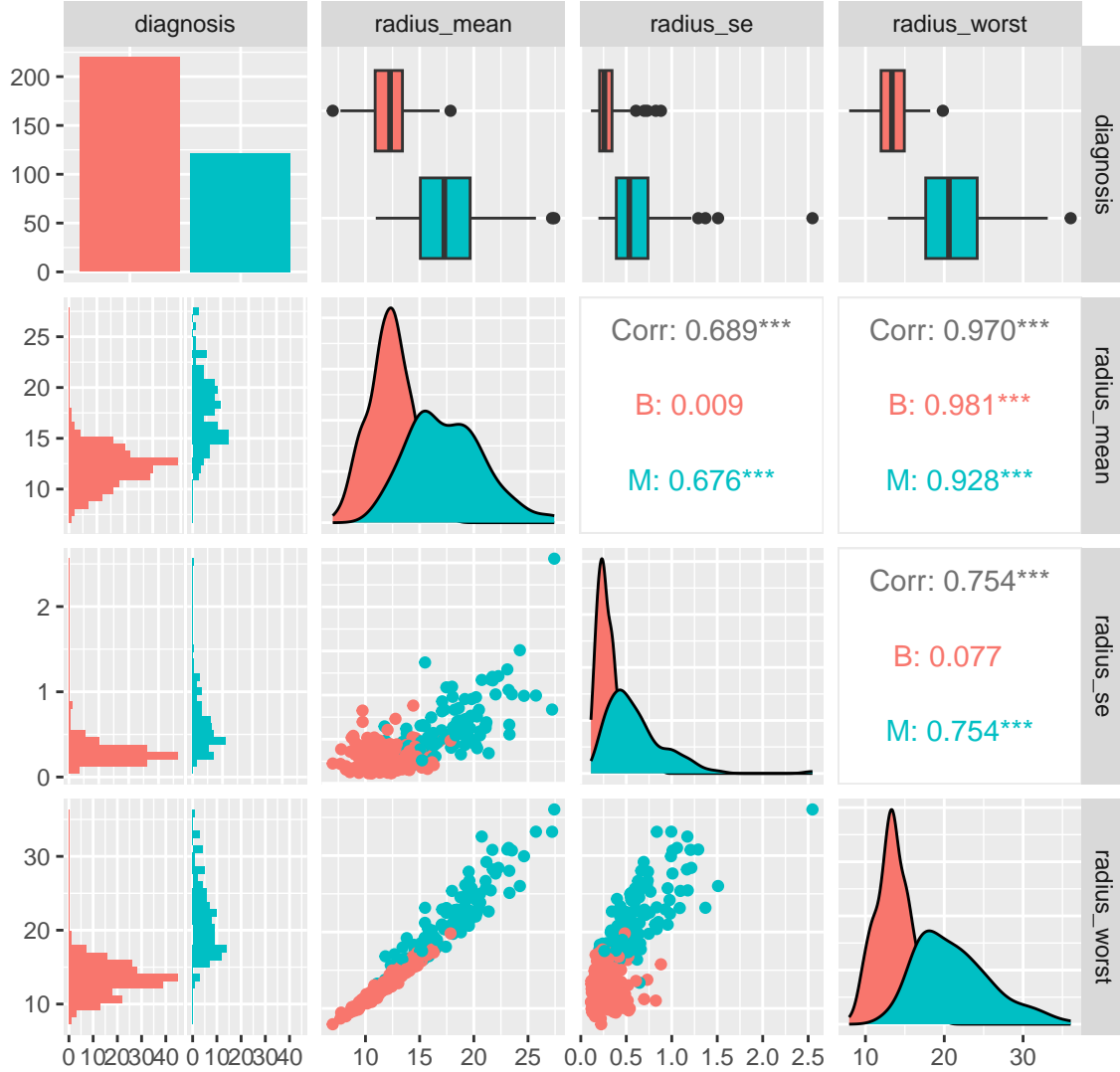
3   Classification Analysis

During our classification analysis, we attacked `Type-II error` in two different ways. Austin used different paramters for each model and to figure out which parameters yield the lowest `Type-II error rate`. However, testing many different parameters can be time consuming and computationally expensive. Danny used a different approach by tuning his models to be more sensitive to Malignant cells. This allows him to reduce the `Type-II error rate` by classifying more cells as Malignant. However, on the downside, this also increases the `Type-I error rate` by classifying more cells as Malignant. By harnessing both approaches, we were able to reduce the `Type-II error rate` to its fullest while also maintaining a good `Type-I error rate`.

3.1   Austin's Classification Proccess and Analysis

For all of models that Austin used were tuned using a `10-fold cross validation` method that was repeated 10 times. Cross validation is method that is used to train a model on a subset of the data and then test the

model on the remaining data. This process is repeated 10 times with each subset of data being used as the test set once. By using this method we are able to effectively train our model while also testing it on data that it has not seen before.

3.1.1 KNN Model

The first model that we used was a K Nearest Neighbor's model. In order to tune this model, we used the tuneGrid parameter to test different values of k. We tested 100 values of k ranging from 1 to 100 and then plotted the Type-II error rate (in purple) as well as the overall error rate (in black) for each value of k as shown in Figure C below. The model with the lowest Type-II error rate was the model with k = 10. Any value of k greater than 10 resulted in a higher Type-II error rate as well as a higher overall error rate. This can be attributed to the fact that the model is overfitting the data.

Figure C: KNN Type II and Overall Error
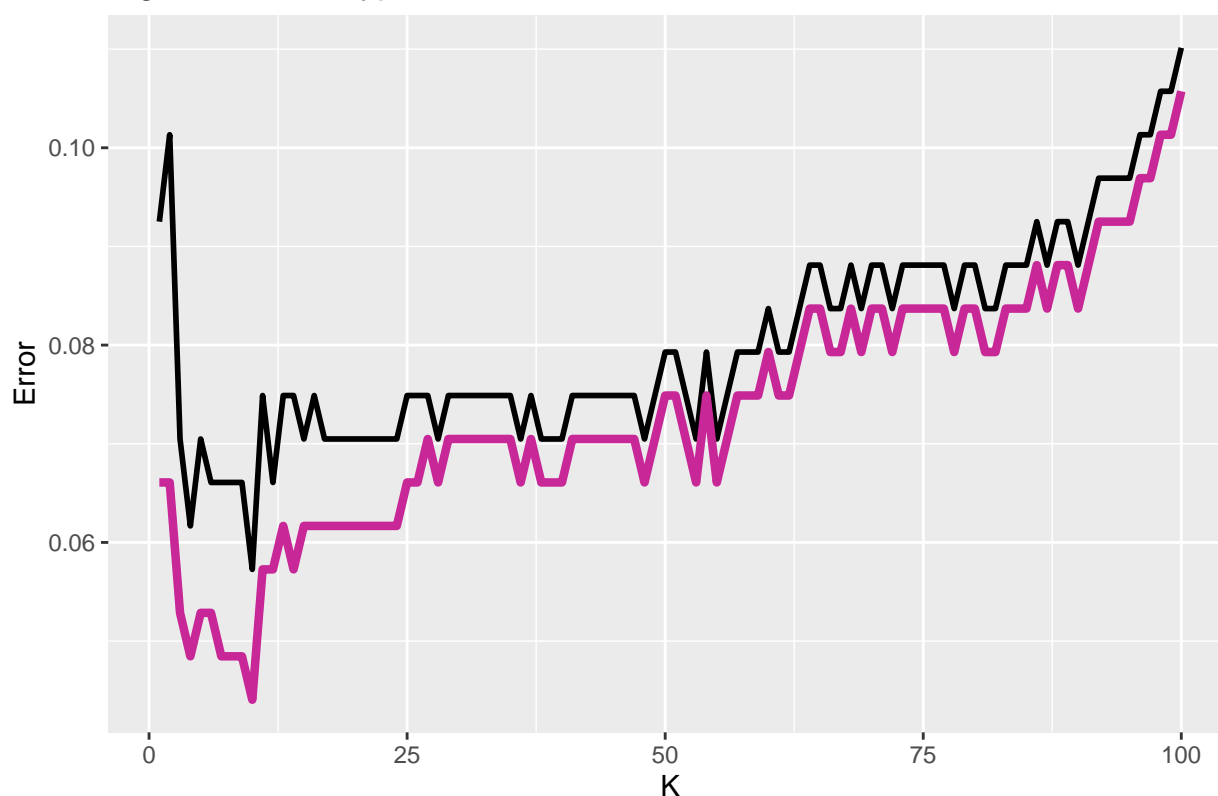
3.1.2 Tuned KNN Model

Using the information from the previous plot, we were able to tune our model to have a k value of 10. This tuned model resulted in a Type-II error rate of 0.0440529 and an overall accuracy of 0.9339207. The confusion matrix for this model is shown in Figure D below. As you can see out of the total 342 training samples, 12 were misclassified as benign when they were actually malignant.

Figure D: KNN Confusion Matrix
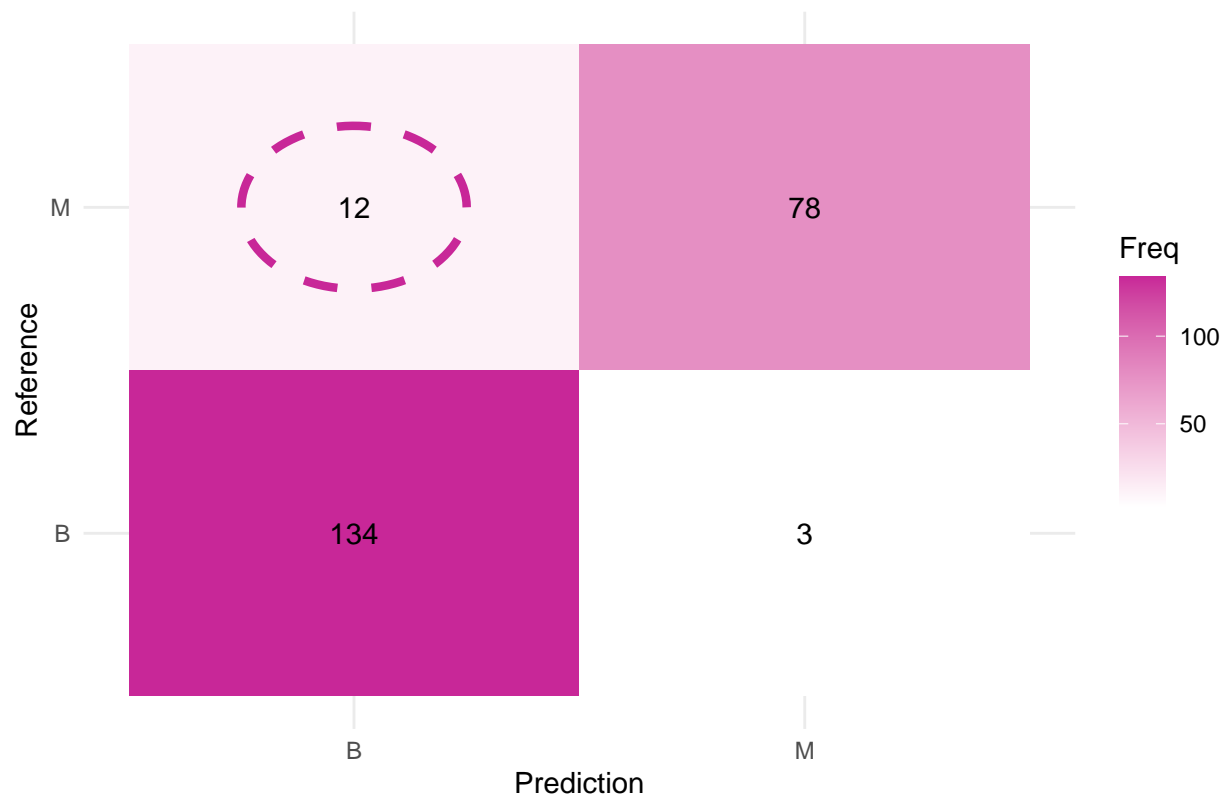
3.1.3   Random Forrest Model

The second model that we used was a `Random Forrest model`. In order to tune this model, we used the `tuneGrid` parameter to test different numbers of `trees`. We tested `10` values of `trees` ranging from `1` to `100` and then plotted the `Type-II error rate` (in purple) as well as the overall error rate (in black) for each value of `trees` as shown in Figure E below. The model with the lowest `Type-II error rate` was the model with `trees = 21`. Any value of `trees` greater than 21 resulted in a higher `Type-II error rate` or a higher computation time for the same `Type-II error rate`.

Figure E: Random Forrest Type II and Overall Error
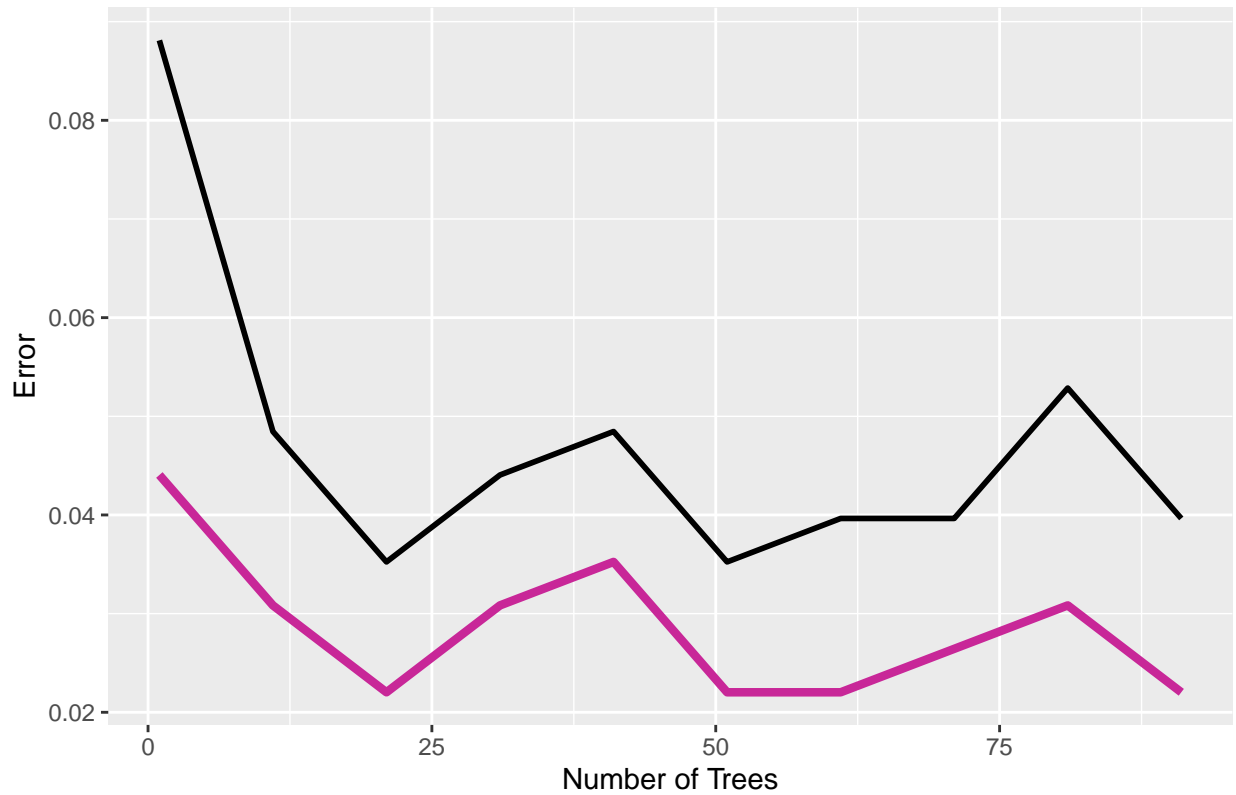
3.1.4 Tuned Random Forrest

Using the information from the previous plot, we were able to tune our model to have a `trees` value of 21. This tuned model resulted in a `Type-II error rate` of 0.0220264 and an overall accuracy of 0.9603524. The confusion matrix for this model is shown in Figure F below. As you can see out of the total 342 training samples, 5 were misclassified as benign when they were actually malignant. And we were able to further reduce our `Type-II error rate` from 0.0440529 to 0.0220264 when compared to the KNN model.

Figure F: Random Forrest Confusion Matrix
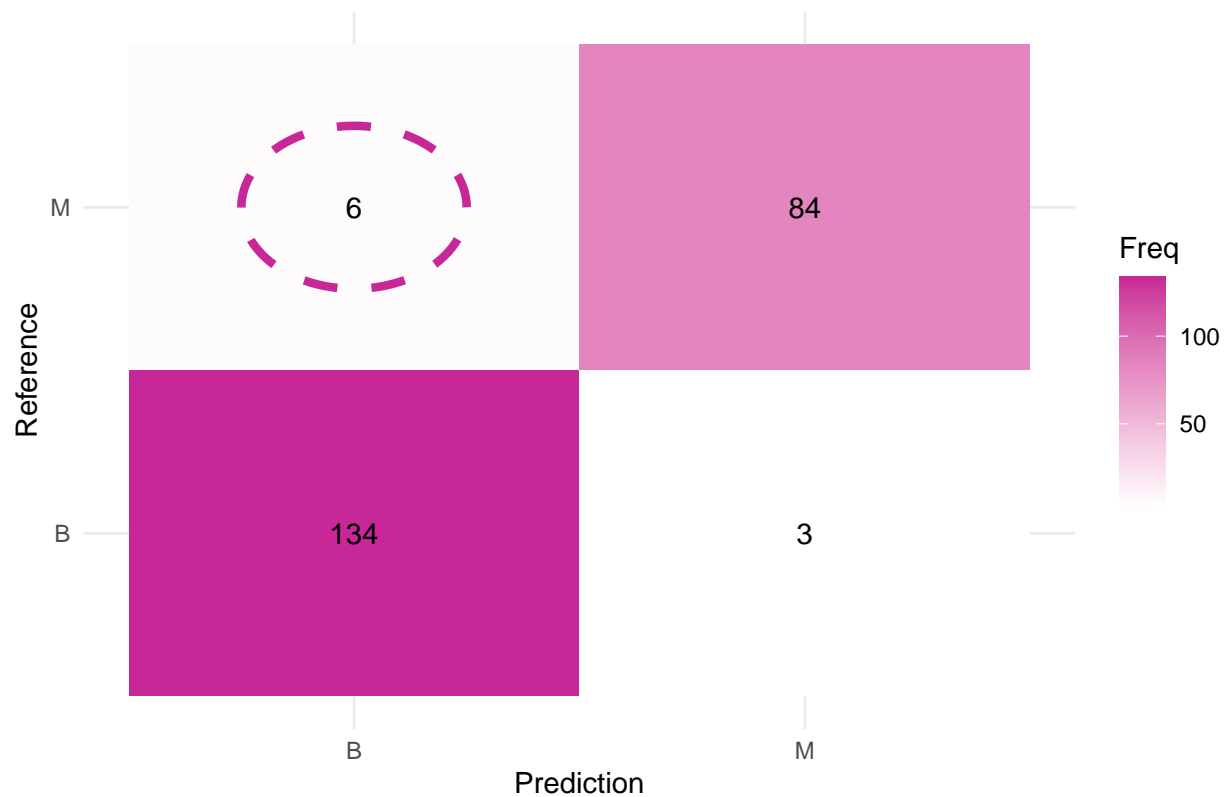
3.1.5 Radial Support Vector Machine

The third model that we used was a `Radial Support Vector Machine`. To begin we used a basic model with the default parameters and then tuned the model using the `tuneGrid` parameter. The basic SVM model resulted in a `Type-II error rate` of 0.0044053 and an overall accuracy of 0.969163. The confusion matrix for this model is shown in Figure G below. As you can see out of the total 342 training samples, 1 was misclassified as benign when they it was actually malignant. This is a great improvement in our `Type-II error rate` when compared to the other models.

Figure G: Radial SVM Confusion Matrix
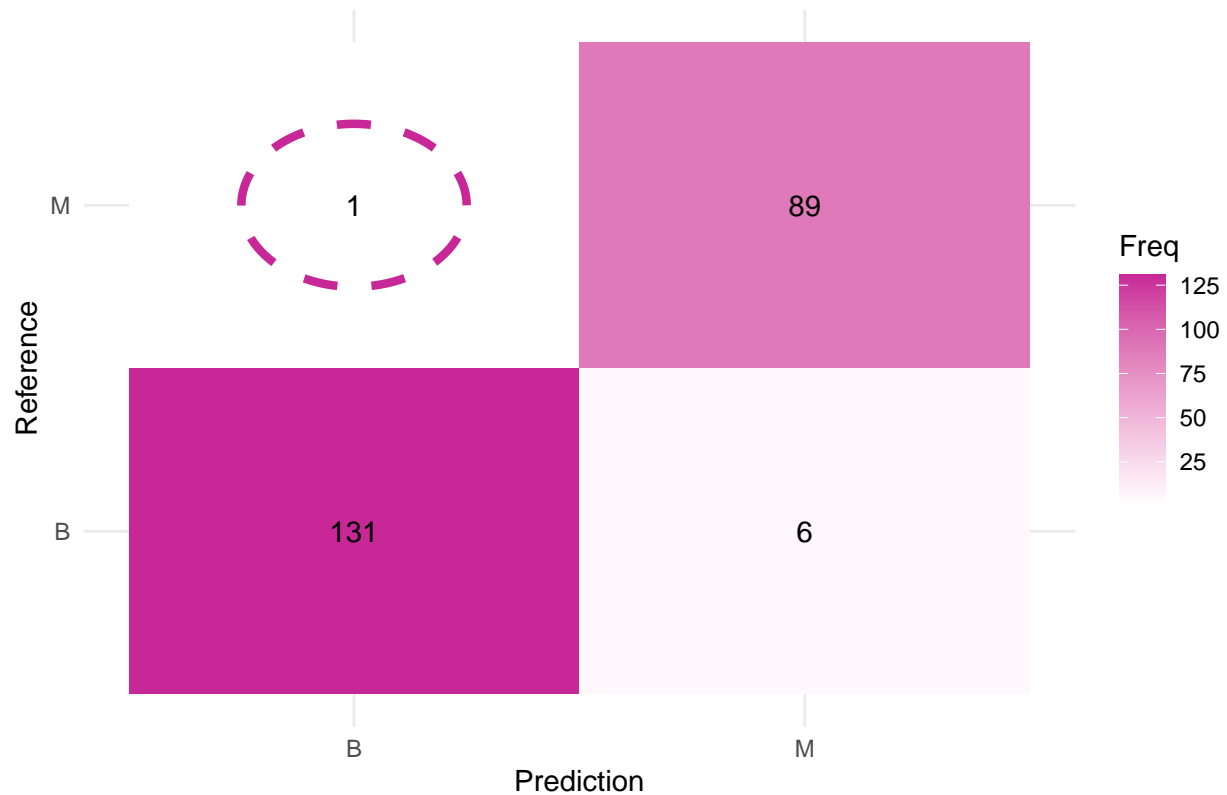
3.1.6 Tuned Radial Support Vector Machine

The final model that we used was a tuned `Radial Support Vector Machine`. The tuned model resulted in a `Type-II error rate` of 0.0088106 and an overall accuracy of 0.9735683. The confusion matrix for this model is shown in Figure H below. Out of the total 342 training samples, 2 were misclassified as benign when they were actually malignant. As you can see, our `Type-II error rate` was actually increased to 0.0088106 when compared to the basic SVM model. Since this svm model was tuned using carets `tune.svm` function, the overall error was reduced, but this resulted in a slightly higher `Type-II error rate`. For this reason, our basic model is actually better than the tuned model when it comes to meeting our goal of reducing the `Type-II error rate`.

Figure H: Tuned Radial SVM Confusion Matrix
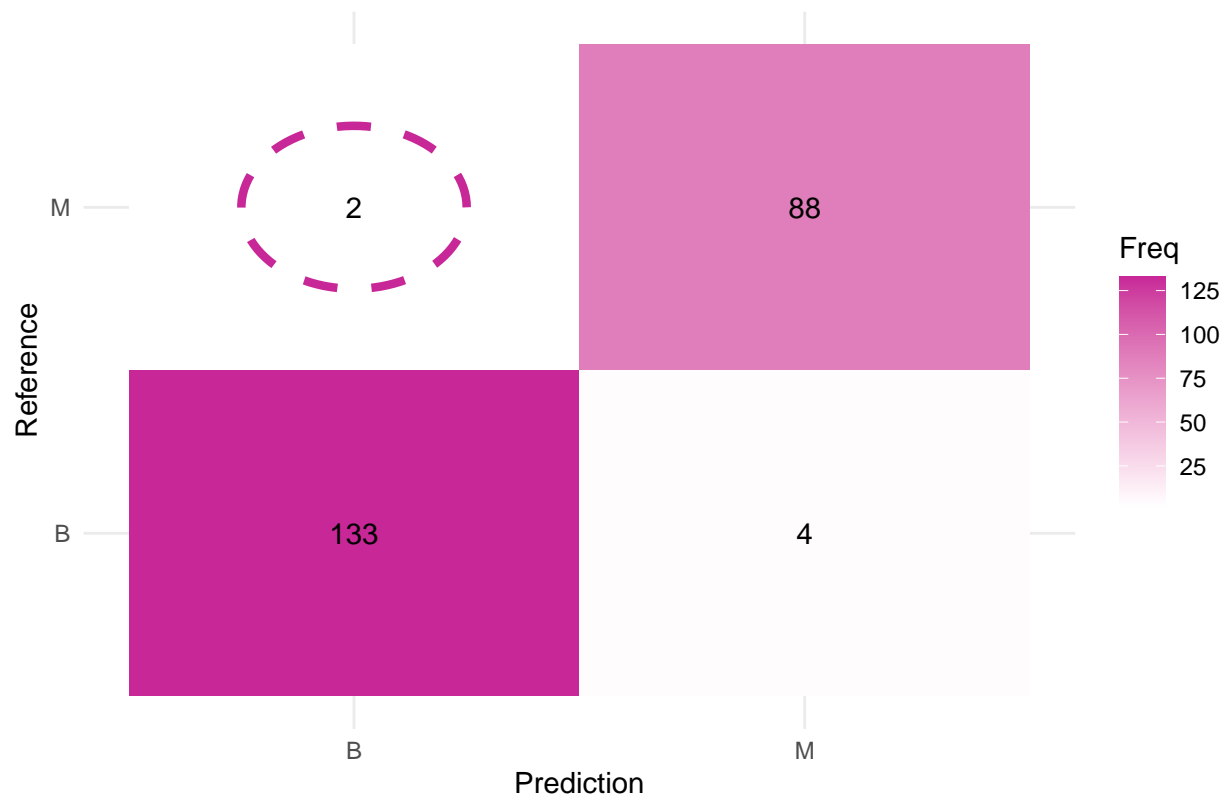
Insert text here

3.2 Dannys's Classification Proccess and Analysis

Insert text here

4 Ethans's Regression Proccess and Analysis

Insert text here

5 Analysis Summary

Insert text here

6 Potential Improvements

Following our analysis, we have identified a few potential improvements that could be made to our analysis to further improve our models. The obvious and first improvement that could be made is to **collect more data**. With more data, we would allow our models to see more examples of malignant cells and be able to better classify them. As you saw in the Exploratory Data Analysis section, we have more more information regarding the benign cells than the malignant cells. If we had more data, we could balance out the data

set and have more information regarding the malignant cells. Another possible improvement is to **collect different or more features** regarding each cell. Most of the features included the mean, worst and standard error of geometry measurements of the cell. This leads to many variables being highly correlated with each other and could lead to multicollinearity. If we had more features, we could reduce the multicollinearity and diversify the information we have regarding each cell. Finally, if we had **access to better technology** and more time or money, we could use a more advanced machine learning model. Because this project was done for a class, we were limited to the models we could use as there was a time contraint, especially in a team enviroment.

7   Works Cited

- Breast Cancer Wisconsin (Diagnostic) Data Set:
  - https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data?datasetId=180&sortBy=voteCount
- Definition of Features (Variables):
  - https://www.causeweb.org/usproc/sites/default/files/usclap/2017-2/Evaluating_Benign_and_Malignant_Breast_Cancer_Cells_from_Fine-Needle_Aspirates.pdf