Machine Learning for Breast Cancer Research

Lydia Jeong, Robyn Forman, Nicholas Lindsay-Lewis

INF2179: Machine Learning with Applications in Python

Table of Contents

1 Introduction

In spite of significant research and public awareness, breast cancer remains as the most common cancer and the largest cause of cancer deaths among women worldwide (World Health Organization, 2020). An estimated 1 in 8 Canadian women will develop breast cancer, and 1 in 33 will die of it (Government of Canada, 2019). Therefore, early diagnosis and treatment is vital for improving the patient's chances of survival. Fortunately, with advancement in medical research, breast cancer can be detected with a high level of certainty. Using the Wisconsin Diagnostic Breast Cancer (WDBC) dataset, machine learning algorithms can be developed to predict cancer diagnosis.

The WDBS dataset contains quantitative cytological data collected from Fine Needle Aspiration (FNA) samples of breast tumors. Data collection involved extracting a small tissue sample from the tumor with a thin needle, capturing the histological images of these samples, and collecting cell nucleus characteristics based on quantitative measurements obtained from the images (Figure 1).

Figure 1. Histological image of a FNA sample of breast tumor.

Following the collection of this diagnostic dataset, Wolberg et al. followed-up with the patients classified in the malignant category. The Wisconsin Prognostic Breast Cancer (WPBC) dataset was purposed to track recurrence or non-recurrence of malignancy over time. With the WPBC, Wolberg and team set out to produce a prognosis prediction model; to predict the time or likelihood of recurrence of cancer in a patient. The WDBC dataset is available in CSV format on Kaggle: Breast Cancer Wisconsin (Diagnostic) Data Set. The WPBC dataset is publicly accessible through UCI Machine Learning Repository.

2 Data Description

Part 1: Wisconsin Diagnostic Breast Cancer (WDBC) dataset

The WDBS dataset provides information about key cell nuclear characteristics from examined breast tumor, including nuclear size, shape, and texture. It contains 569 rows x 32 columns. Each instance contains the following attributes:

Attribute name	**Type**	**Description**
1) ID number	Integer	
2) Diagnosis	Categorical	M = malignant, B = benign

3) - 33)
In the remaining 30 columns, there are three measures for each of the following 10 features:
the mean, the standard error, and the worst value (mean of the three largest or extreme values).

Attribute name	**Type**	**Description**
Radius (μm)	Float	
Texture	Float	Standard deviation of image gray-scale values
Perimeter (μm)	Float	
Area (μm^2)	Float	
Smoothness	Float	Local variation in radius lengths
Compactness	Float	Perimeter ^{2} / area - 1.0
Concavity	Float	Severity of concave portions of the contour
Concave Points	Float	Number of concave portions of the cell contour
Symmetry	Float	
Fractal Dimension	Float	"coastline approximation" - 1

There are 357 Benign and 212 Malignant samples.

Part 2: Wisconsin Prognostic Breast Cancer (WPBC) dataset

The WPBC dataset contains 198 rows x 35 columns. Features in this dataset include:

Attribute name	**Type**	**Description**
1) ID number	Integer	
2) Outcome	Categorical	R = recur, N = nonrecur
3) Time	Float	Recurrence time if R, disease-free time if N
3) - 33) 30 cell nuclei features collected in the diagnostic dataset	Float	See WDBC dataset description in Part 1
34) Tumor size	Float	Diameter of the excised tumor in centimeters
35) Lymph node status	Float	Number of positive axillary lymph nodes observed at time of surgery

There are 151 patients in the non-recur and 47 patients in the recur category.

3 Exploratory Data Analysis

Conducting basic data analysis, the following phenomena were of interest:

1. From the generated boxplots, the malignant cell nucleus typically had higher values compared to benign cell nucleus for most of the features (Figure 2 shown below, and Figure 34 and 35 in the Appendix). The features that did not display this trend were the mean fractal dimension, standard error of texture, smoothness, and symmetry, suggesting that they may not be relevant or useful features for classifying between the Malignant (M) and Benign (B) classes. As expected, they were dropped later in the analysis during the feature selection process.

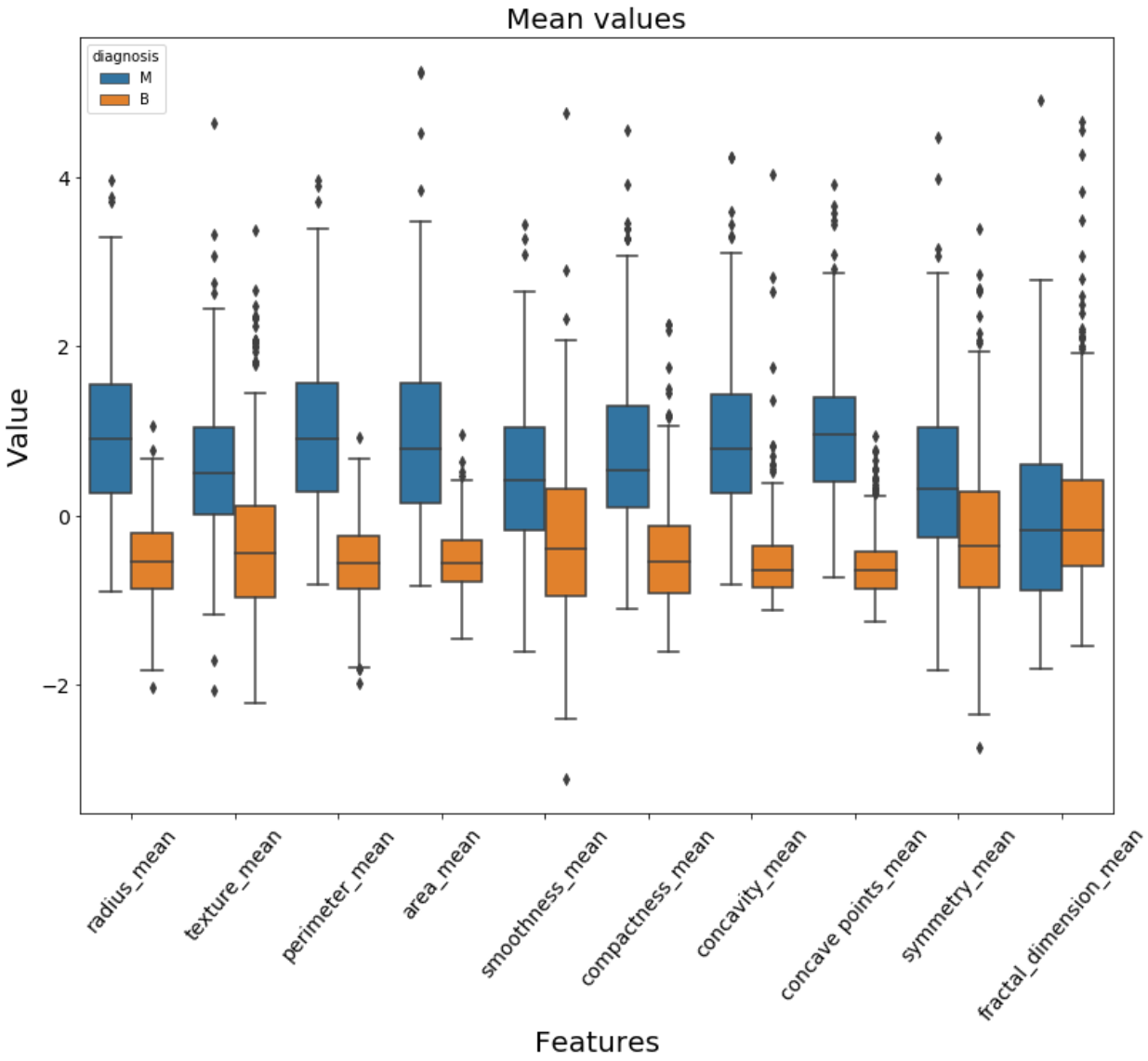


Figure 2. Boxplot comparing the Malignant (M) and Benign (B) classes for the first ten features.

2. The frequency distribution between the M and B illustrated some overlap between the two classes for all features (Figure 3).

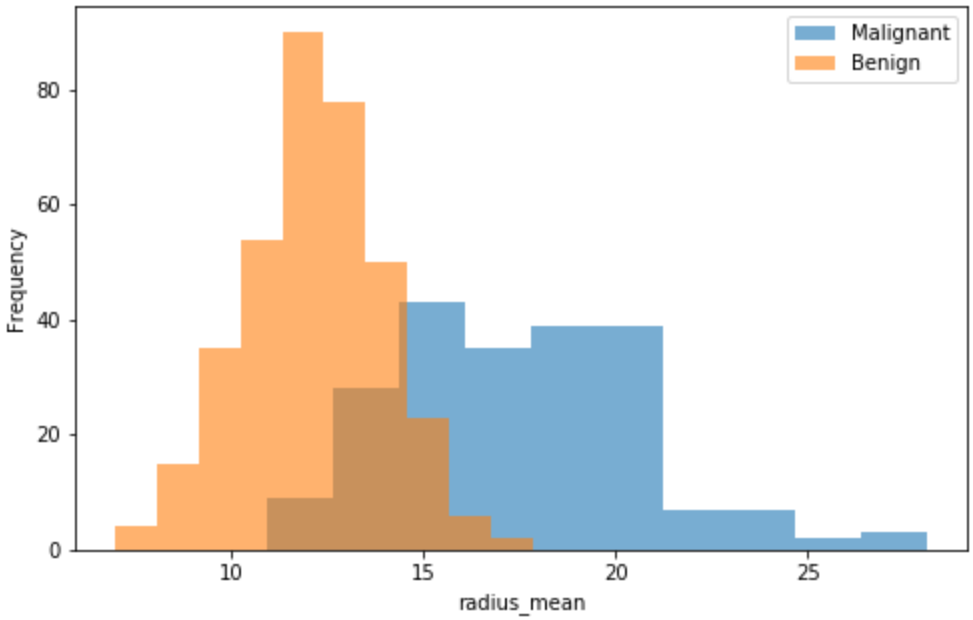


Figure 3. Histogram of mean radius, one of the stronger predictors.

3. A heatmap illustrated high correlation between some variables. For example, the radius mean showed high correlation with the perimeter mean and the area mean (Figure 4). It is necessary to drop some of these attributes for our machine learning predictive analysis to eliminate any feature dependencies.

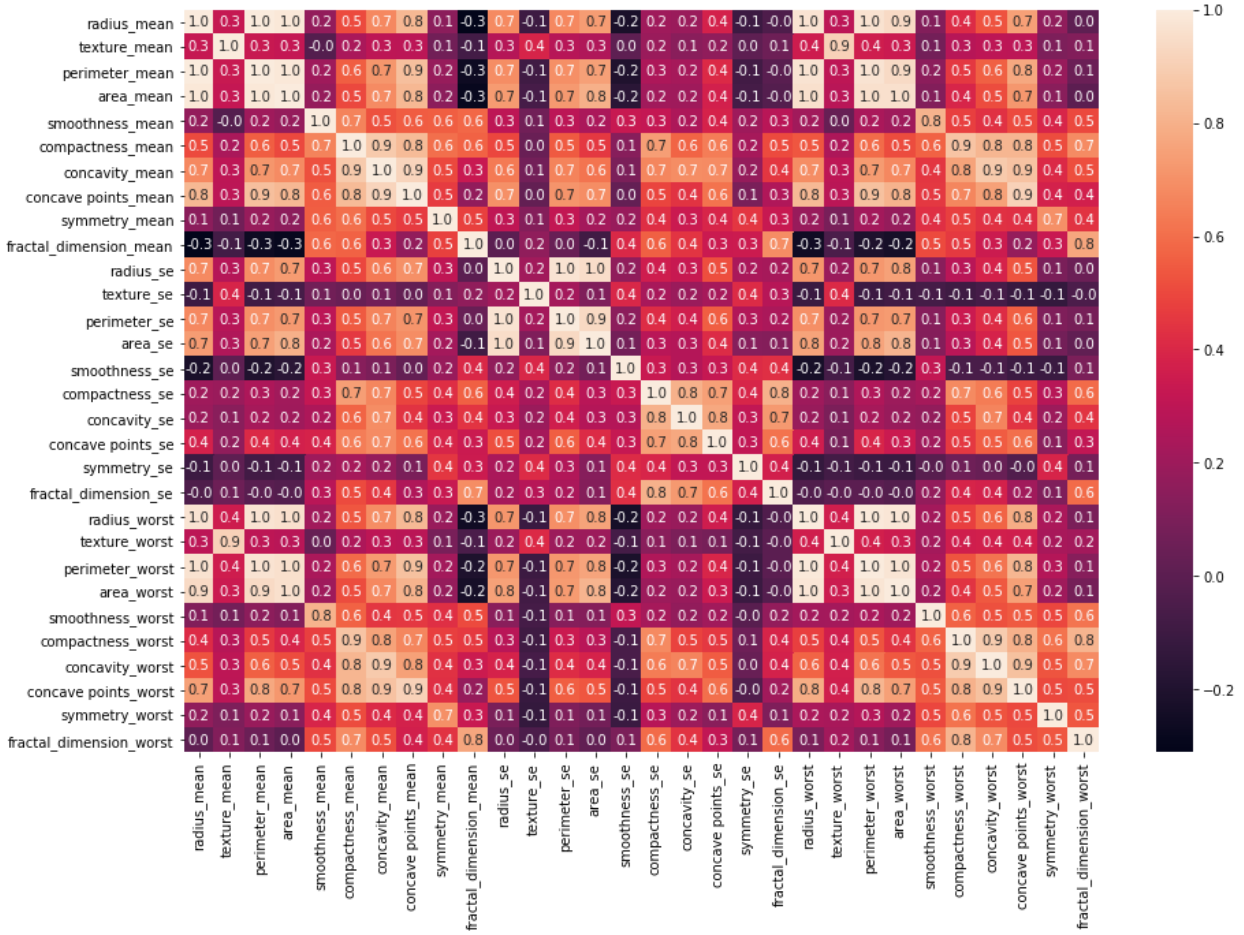


Figure 4. Heatmap displaying correlation values between each feature.

4 Research Questions

These interesting phenomena observed from basic exploratory analyses gave rise to three research questions, which are proposed below:

Question 1: What attributes are the best predictors of breast tumor diagnosis?

- What methods of classification work best to infer dependencies and interpret models?

Question 2: How can we predict breast tumor diagnosis based on the features of the cell nucleus?

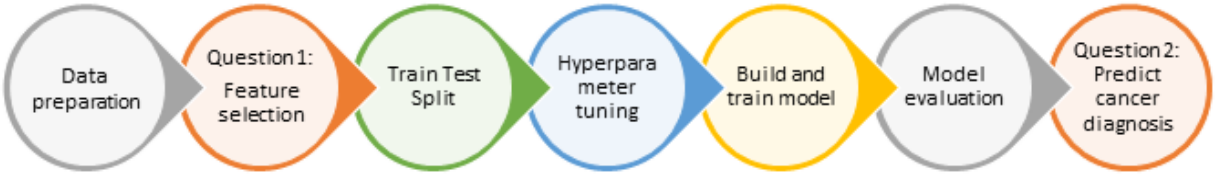
- How can we measure and ensure optimal accuracy of breast tumor diagnosis?

Question 3: Can we predict whether cancer will reoccur? If yes, can we predict the time when the cancer will likely recur after patients have been treated?

For Question 1, the rationale was to determine the attributes that lead to highest predictive accuracy and drop those that do not have a significant impact on diagnosis so that we can make a more accurate prediction based on select features. For Question 2, we instantiated our data and conducted analysis through kNN, Gradient Boosting, XGBoost, and logistic regression classifier to determine the model with the best predictive accuracy. Finally, we instantiated time-series data from the prognostic dataset to test their accuracy in tumor classification and attempt to predict recurrence of cancer.

5 Techniques

Part 1: Breast cancer diagnosis



Data was prepared by checking for any missing values, errors, or duplicates, and removing them, as necessary. Feature selection was determined by selecting the top nodes in the decision tree. The most relevant or important features were verified from random forest's feature importance property, which computed the Gini importance values for each feature. From the selected features, data normalization was performed, and the datasets were split into 75% training and 25% testing sets (Figure 5). kNN, Gradient Boosting, XGBoost, and logistic regression were selected for machine learning analysis. Hyperparameter tuning was performed for each model to determine the best parameters that can improve the model's performance. The models were trained and tested. Accuracy was determined by the 5-fold cross-validation score (Stone, 1974), and these values were compared for four models. The model with the highest accuracy was selected for predicting cancer diagnosis.

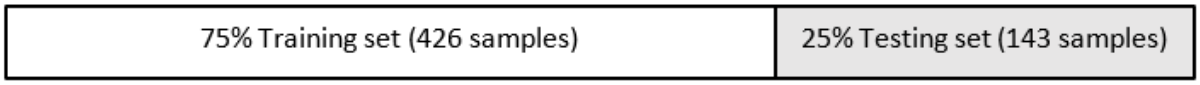


Figure 5. Splitting of the 569-patient data into training and testing sets.

Part 2: Breast cancer prognosis

For predicting cancer recurrence, similar techniques were used as the cancer diagnosis study. In the prognosis dataset, there were four missing values in the lymph node status. These missing

values were replaced with the mean value of the lymph node status column. Feature selection was performed to keep only relevant features and remove all irrelevant or redundant features. Univariate selection method was used to select the features with the strongest relationship with the target variable. Chi-squared test was used to select the top five best features. Again, the dataset was split into 75% training and 25% testing set, resulting in 148 and 50 samples respectively (Figure 6). We chose to use kNN and XGBoost for our analysis. Following data normalization, hyperparameter tuning was performed on both models, then model training, and model evaluation was achieved. Accuracy was determined by the 5-fold cross-validation score (Stone, 1974), and these values were compared for two models. The model with the highest accuracy was selected for predicting cancer prognosis.

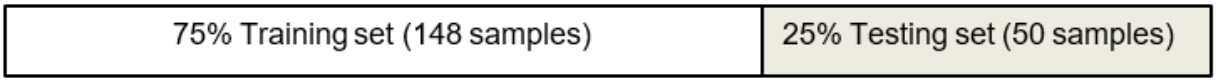


Figure 6. Splitting of the 198-patient data into training and testing sets.

6 Analysis

Question 1: What attributes are the best predictors of cancer diagnosis?

1) Decision Tree

Tree based methods were used to seek predictors that provided the best information gain. First and foremost, we pruned the tree and found the best effective alpha value (Scikit-learn, 2020). After training the decision tree with the effective alphas, the best alpha value of 0.015 was obtained, which maximized the testing accuracy as shown in Figure 7. Based on this alpha value, the tree's number of nodes and depth was reduced to an appropriate number (Figure 8).

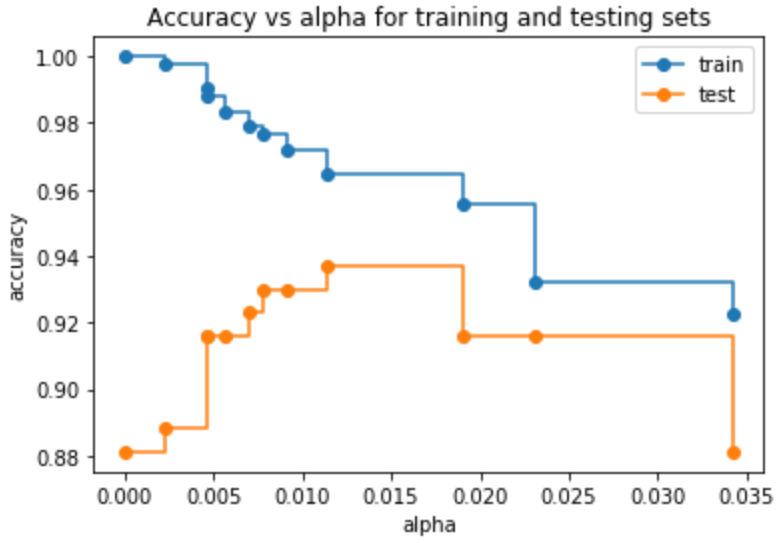


Figure 7. Training and testing accuracy vs. alpha values.

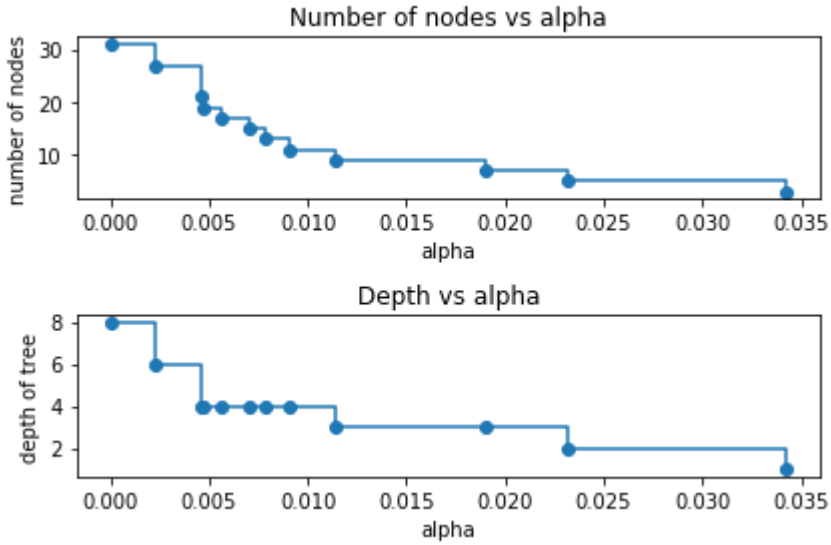


Figure 8. Number of nodes and depth of tree vs. alpha values.

Using this best alpha value, we generated the decision tree as illustrated in Figure 9. As shown, the best predictors of diagnosis were the mean concave points, worst radius, worst perimeter, standard error area, mean texture, worst texture, and worst symmetry. Only these features were kept, thereby reducing the number of features from 30 to 7.

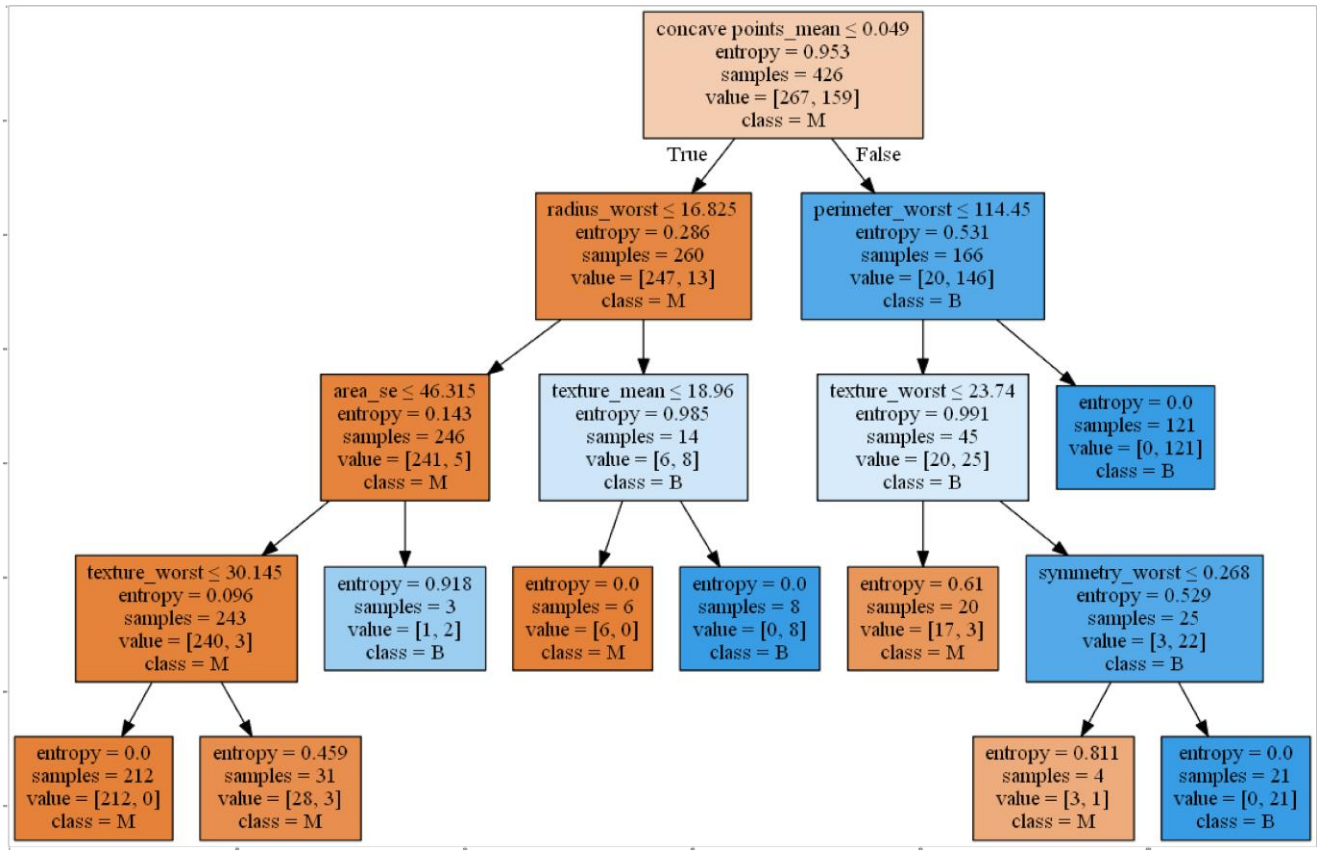


Figure 9. Pruned decision tree generated from 0.015 effective alpha value.

2) Random Forest Classifier's Feature Importance property

From these seven predictor variables, we determined the most important feature by using Random Forest Classifier's feature_importances_. This property returned feature importance values, also known as Gini importance, in which higher values represent greater importance.

Best parameters were calculated for a random forest classifier, which had maximum depth of seven and number of estimators of 250 (Figure 10). Following model training and fitting with these selected parameters, feature_importance_ property was used to calculate the Gini importance of each predictor variable.

```
rf = RandomForestClassifier()
parameters = {
    'n_estimators': [5, 50, 250, 500],
    'max_depth': [1, 3, 5, 7, 9]
}

cv = GridSearchCV(rf, parameters, cv=5)
cv.fit(tr_features, tr_labels.values.ravel())

print_results(cv)
```

```
BEST PARAMS: {'max_depth': 7, 'n_estimators': 250}
```

Figure 10. Hyperparameter tuning for random forest classifier.

As verified in Figure 11, the most important feature (with the highest feature importance value) was concave points_mean, followed by perimeter_worst and radius_worst. This result confirmed our decision tree, which had concave points_mean as the root node and perimeter_worst and radius_worst as the top nodes. Therefore, the best predictors were the mean concave points, followed by worst perimeter, worst radius, standard error area, worst symmetry, worst texture, and mean texture.

```
concave points_mean    0.324796
perimeter_worst        0.233568
radius_worst           0.190753
area_se                0.108920
symmetry_worst         0.053020
texture_worst          0.050727
texture_mean           0.038217
```

Figure 11. List of best features with corresponding Gini importance values.

Question 2: How can we predict breast cancer diagnosis?

From the seven best predictor variables we have selected, we wanted to determine how to predict cancer diagnosis with the best accuracy. To achieve this task, we built four different models as aforementioned, calculated their accuracy, and selected the model with the best predictive accuracy.

1) k-Nearest Neighbour Classification

The number of neighbours for kNN that maximized the testing accuracy was three, resulting in a testing accuracy of 0.986 and training accuracy of 0.974.

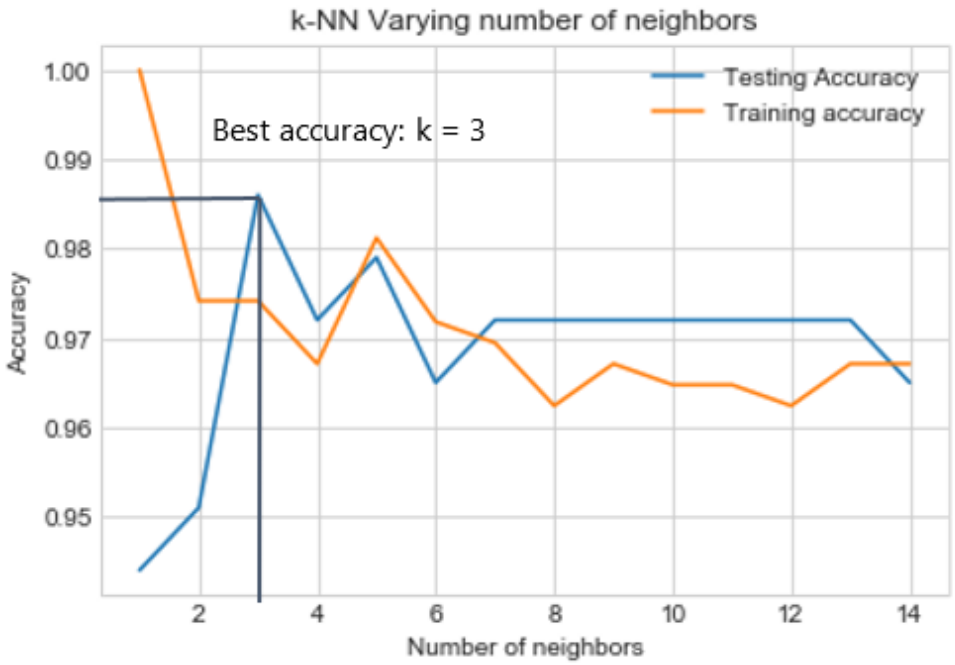


Figure 12. Best value for number of neighbors for testing and training accuracy.

F1 scores for benign and malignant were 0.99 and 0.98 respectively, and overall accuracy was 0.99. Five-fold cross-validation score was 0.968 (Figure 13).

```
KNN Testing accuracy:  0.986013986013986
KNN Training accuracy:  0.9741784037558685


              precision    recall  f1-score   support

           0       0.98      1.00      0.99        98
           1       1.00      0.96      0.98        45

    accuracy                           0.99       143
   macro avg       0.99      0.98      0.98       143
weighted avg       0.99      0.99      0.99       143



Cross-validation scores: [0.97368421 0.93859649 0.97368421 0.97368421 0.98230088]
Mean Cross-validation score: 0.9683900015525542
```

Figure 13. kNN testing, training, precision, recall, F1, and cross-validation score for benign (0) and malignant (1) cells.

2) Gradient Boosting Classifier

Hyperparameter tuning using grid search (cv = 5) resulted in a learning rate of 0.01, maximum depth of three, and number of estimators of 250 (Figure 14).

```
1  gb = GradientBoostingClassifier()
2  parameters = {
3      'n_estimators': [5, 50, 250, 500],
4      'max_depth': [1, 3, 5, 7, 9],
5      'learning_rate': [0.01, 0.1, 1, 10, 100]}
6
7  cv = GridSearchCV(gb, parameters, cv=5)
8  cv.fit(tr_features, tr_labels.values.ravel())
9
10 print_results(cv)
```

```
BEST PARAMS: {'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 250}
```

Figure 14. Hyperparameter tuning for gradient boosting classifier.

F1 accuracy score was 0.94, and the testing and training accuracies were 0.944 and 0.993 respectively (Figure 15). Training accuracy being higher than the testing accuracy may signify overfitting. This issue will be discussed in the limitations. Cross-validation score was computed, which was 0.961.

```
GB Testing accuracy:  0.9440559440559441
GB Training accuracy:  0.9929577464788732


              precision    recall  f1-score   support

           0       0.95      0.97      0.96        98
           1       0.93      0.89      0.91        45

    accuracy                           0.94       143
   macro avg       0.94      0.93      0.93       143
weighted avg       0.94      0.94      0.94       143

Cross-validation scores: [0.96491228 0.92105263 0.97368421 0.98245614 0.96460177]
Mean Cross-validation score:  0.9613414066138798
```

Figure 15. Gradient Boosting testing, training, precision, recall, F1, and cross-validation score for benign (0) and malignant (1) cells.

3) XGBoost Classifier

Hyperparameter tuning for XGBoost using randomized search (cv = 5) led to the following best parameters (Figure 16):

```
{'min_child_weight': 5,
 'max_depth': 8,
 'learning_rate': 0.3,
 'gamma': 0.4,
 'colsample_bytree': 0.4}
```

Figure 16. Hyperparameter tuning for XGBoost.

Testing and training accuracy were 0.958 and 0.983 respectively, f1-score was 0.96, and cross-validation score was 0.96 (Figure 17).

```
XGB Testing accuracy:  0.958041958041958
XGB Training accuracy:  0.9835680751173709


              precision    recall  f1-score   support

           0       0.96      0.98      0.97        98
           1       0.95      0.91      0.93        45

    accuracy                           0.96       143
   macro avg       0.96      0.95      0.95       143
weighted avg       0.96      0.96      0.96       143

Cross-validation scores: [0.93859649 0.94736842 0.98245614 0.96491228 0.96460177]
Mean Cross-validation score:  0.9595870206489675
```

Figure 17. XGBoost testing, training, precision, recall, F1, and cross-validation score for benign (0) and malignant (1) cells.

4) Logistic Regression Classifier

For logistic regression, a heatmap was displayed to check for feature dependencies because highly correlated features are not effective and reliable for the model. As displayed in Figure 18, feature dependencies were evident (correlation value higher than 0.5). Concave points_mean, radius_worst, perimeter_worst, and area_se were highly correlated with each other, while texture_mean was highly correlated with texture_worst. For that reason, one of these features was dropped. A function was created that removed a correlation threshold value higher than 0.5 (Figure 19). Running this function reduced the number of variables from seven to three, resulting in all correlation values below the threshold (Figure 20). The remaining features were concave points_mean, texture_mean, and symmetry_worst.

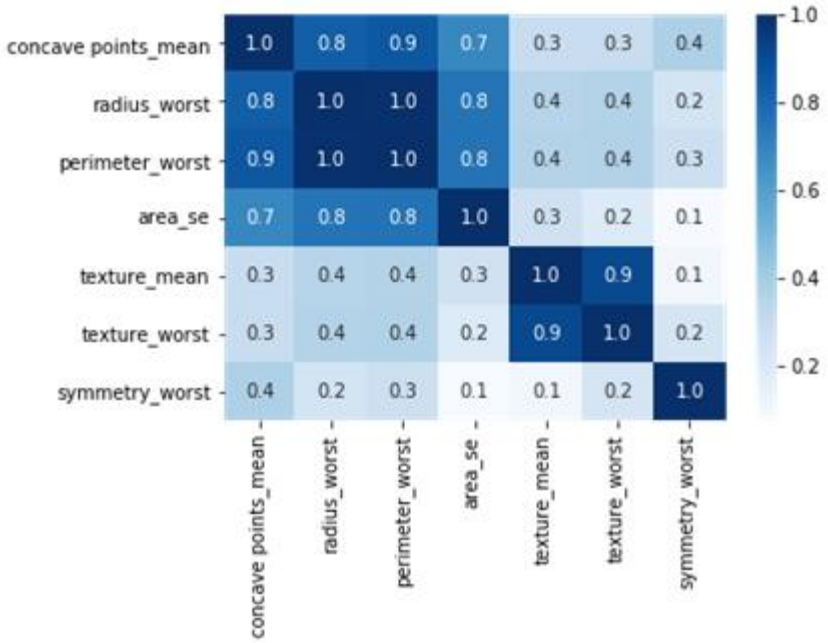


Figure 18. Heatmap showing correlation between each feature.

```python
def correlation(dataset, threshold):
    col_corr = set() # Set of all the names of deleted columns
    corr_matrix = dataset.corr()
    for i in range(len(corr_matrix.columns)):
        for j in range(i):
            if (corr_matrix.iloc[i, j] >= threshold) and (corr_matrix.columns[j] not in col_corr):
                colname = corr_matrix.columns[i] # getting the name of column
                col_corr.add(colname)
                if colname in dataset.columns:
                    del dataset[colname] # deleting the column from the dataset

    print(dataset)

correlation(df_X, 0.5)
```

Figure 19. Function that removes features with correlation value above the threshold.

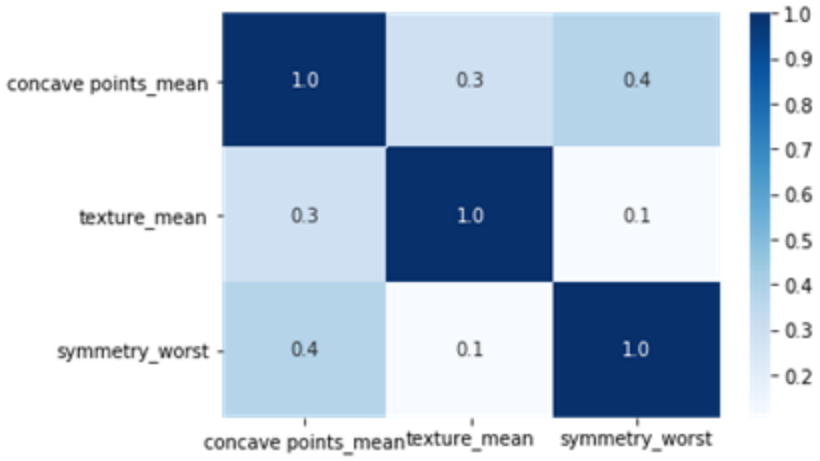


Figure 20. Heatmap showing correlation between the three features (concave points_mean, texture_mean, and symmetry_worst). All three features are independent of each other.

Hyperparameter tuning using grid search method (cv = 5) gave the following best parameters (Figure 21):

```
1  best_clf.best_estimator_
LogisticRegression(C=545.5594781168514)
```

Figure 21. Hyperparameter tuning for logistic regression.

Evaluating the model's predictions, the logistic regression's testing and training accuracies were 0.944 and 0.941 respectively, f1-score was 0.94, and cross-validation score was 0.935 (Figure 22).

```
LR Testing accuracy:  0.9440559440559441
LR Training accuracy:  0.9413145539906104


              precision    recall  f1-score   support

           0       0.93      0.99      0.96        98
           1       0.97      0.84      0.90        45

    accuracy                           0.94       143
   macro avg       0.95      0.92      0.93       143
weighted avg       0.95      0.94      0.94       143

Cross-validation scores: [0.92105263 0.92982456 0.94736842 0.93859649 0.9380531 ]
Mean Cross-validation score:  0.934979040521658
```

Figure 22. Logistic regression testing, training, precision, recall, F1, and cross-validation score for benign (0) and malignant (1) cells.

5) Summary table

A summary table below depicts cross-validation, accuracy, recall, precision, and F1-scores for all four models (Table 1). This table indicates that our prediction with KNN had the best predictive accuracy. Therefore, we chose KNN to predict diagnosis.

Table 1. Accuracy values of each four models

	Cross-validation	Accuracy	Recall	Precision	F1-score
KNN	0.968390	0.986014	0.955556	1.000000	0.977273
Gradient Boosting	0.961341	0.944056	0.888889	0.930233	0.909091
XGBoost	0.959587	0.958042	0.911111	0.953488	0.931818
Logistic Regression	0.933225	0.944056	0.866667	0.951220	0.906977

As an extra for our analysis, Pycaret package was run to examine whether it confirmed our result. Pycaret determines the best model based on comparison between the accuracy of 15 different models. The table below displays the accuracy of the 15 different models (Table 2).

Table 2. Table from Pycaret displaying accuracies for 15 different models.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
0	Logistic Regression	0.9700	0.9917	0.9400	0.9812	0.9577	0.9346	0.9378	0.0134
1	K Neighbors Classifier	0.9624	0.9804	0.9062	0.9929	0.9459	0.9175	0.9213	0.0065
2	Extra Trees Classifier	0.9599	0.9914	0.9267	0.9666	0.9437	0.9128	0.9159	0.2059
3	CatBoost Classifier	0.9599	0.9888	0.9400	0.9538	0.9449	0.9135	0.9157	3.4275
4	Ridge Classifier	0.9549	0.0000	0.8790	1.0000	0.9338	0.9001	0.9062	0.0082
5	SVM - Linear Kernel	0.9548	0.0000	0.9462	0.9382	0.9387	0.9031	0.9072	0.0060
6	Light Gradient Boosting Machine	0.9524	0.9871	0.9333	0.9408	0.9347	0.8974	0.9000	0.0653
7	Extreme Gradient Boosting	0.9523	0.9836	0.9262	0.9462	0.9341	0.8968	0.8992	0.0449
8	Linear Discriminant Analysis	0.9499	0.9875	0.8724	0.9929	0.9271	0.8894	0.8952	0.0063
9	Naive Bayes	0.9447	0.9930	0.8919	0.9599	0.9222	0.8795	0.8835	0.0049
10	Gradient Boosting Classifier	0.9447	0.9809	0.9190	0.9314	0.9237	0.8805	0.8822	0.1539
11	Random Forest Classifier	0.9397	0.9737	0.8924	0.9430	0.9154	0.8688	0.8713	0.1300
12	Ada Boost Classifier	0.9373	0.9789	0.9262	0.9107	0.9160	0.8661	0.8690	0.1236
13	Quadratic Discriminant Analysis	0.9297	0.9879	0.8990	0.9162	0.9050	0.8494	0.8524	0.0068
14	Decision Tree Classifier	0.9071	0.9029	0.8857	0.8743	0.8769	0.8026	0.8062	0.0056

Contrary to our result, Logistic regression had the best accuracy as indicated in the table. However, according to our result, logistic regression had the lowest accuracy with score of 0.93, while KNN classifier had the best accuracy with score of 0.97.

6) Make predictions

As aforementioned, kNN was selected for predicting cancer diagnosis. Table 3A below shows the predicted and actual labels of the first ten patients, where 0 is benign and 1 is malignant. Table 3B indicates the number of incorrectly diagnosed patients. There were two patients (patient 91594602 and 855167) who were misclassified as benign when they were in fact, malignant. This outcome was further reinforced by the confusion matrix (Figure 23) that describes the performance of the kNN classification model. All 98 benign patients have been correctly predicted, whereas from the 45 malignant patients, two of them have been erroneously classified as benign.

Table 3. A) Sample of ten patients with predicted and actual labels and whether the prediction was correct. B) Two patients in the 'Malignant' category that were classified as benign.

A

	id	Predicted	Actual	Correct
109	864018	0	0	True
514	91594602	0	1	False
13	846381	1	1	True
3	84348301	1	1	True
240	88350402	0	0	True
548	923169	0	0	True
536	91979701	1	1	True
319	894335	0	0	True
551	923780	0	0	True
318	894329	0	0	True

B

Number of incorrectly diagnosed patients: 2

	id	Predicted	Actual	Correct
514	91594602	0	1	False
40	855167	0	1	False

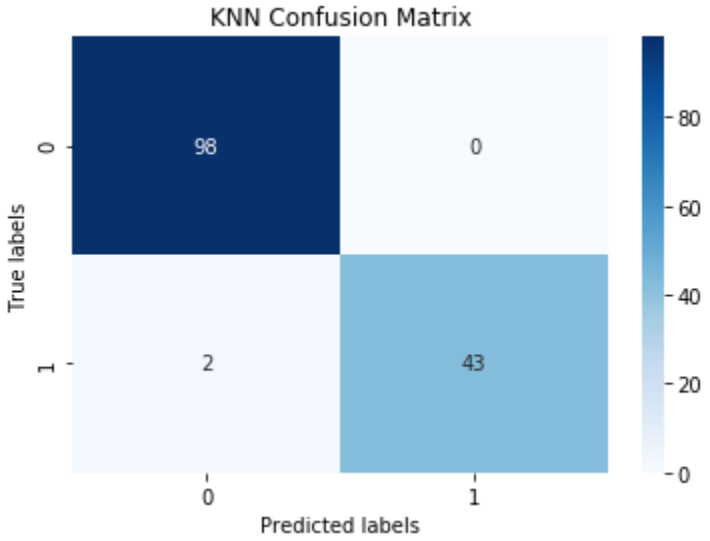


Figure 23. kNN Confusion matrix.

7) Visualize in 3-dimensional graph

Finally, to visualize the graph in three-dimensional space, top three features with the greatest feature importance were selected, which were:

- mean concave points
- worst radius
- worst perimeter

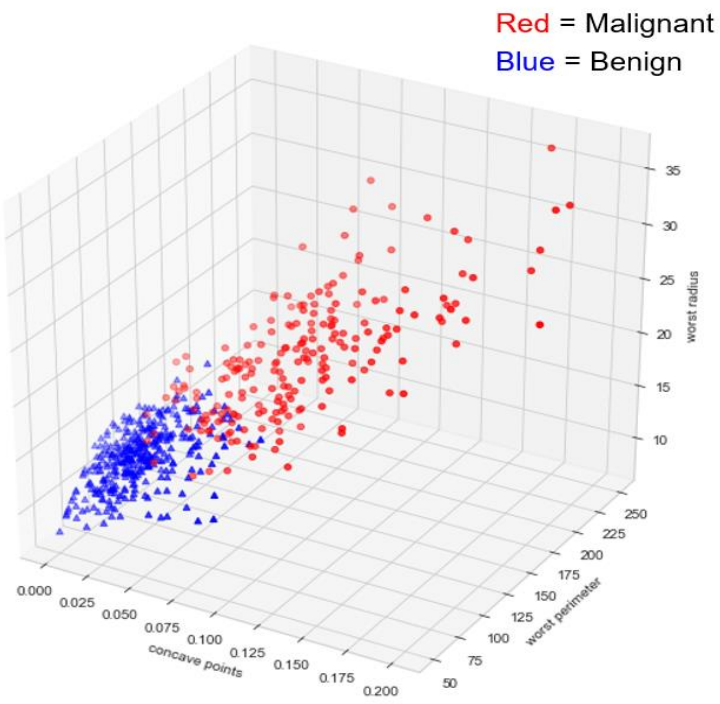


Figure 24. 3D graph visualizing the mean concave points, worst perimeter, and worst radius for malignant and benign classes.

Figure 24 illustrates a clear distinction between the two classes, Malignant and Benign. The Malignant group represented in red generally had higher values compared to the Benign group represented in blue.

Question 3: Can we predict if cancer will recur? Can we predict recurrence time?

The 'Malignant' patients from the cancer diagnosis study were followed-up after surgical removal of tumors to collect data about whether the disease had recurred, and the number of months it took for recurrence (Wolberg et al., 1995a). The prognostic datasets were obtained from this follow-up study. The purpose of the prognostic analysis was to predict cancer recurrence as well as time of recurrence.

1) Remove missing values

After visual inspection of the dataset, it was apparent that the variable lymph node status had four missing values (Figure 25). As a result, these four values were set to the mean of the lymph node status.

```
1 df[['lymph_node_status']][df['lymph_node_status'] == "?"]
```

	lymph_node_status
6	?
28	?
85	?
196	?

Figure 25. Missing values in lymp_node_status column.

2) Feature selection

No additional missing values, errors, or duplicates were present in the data. Next step involved selecting the most important or relevant features out of the 35 features. This task was performed with a chi-squared (chi^2) statistical test to select the five most significant features from the dataset. Table 4 below shows the list of five most significant features with its corresponding score. As listed, the most important features were the worst area, mean area, standard error area, worst perimeter, and lymph node status.

Table 4. Five best features.

	Specs	Score
23	area_worst	2666.167018
3	area_mean	908.125356
13	area_se	148.869075
22	perimeter_worst	63.065828
31	lymph_node_status	50.590415

3) k-Nearest Neighbour Classification

Following data normalization, the best value for the number of neighbors was calculated. The k value for kNN that maximized the testing accuracy was two, resulting in a testing accuracy of 0.82 and training accuracy of 0.80 (Figure 25).

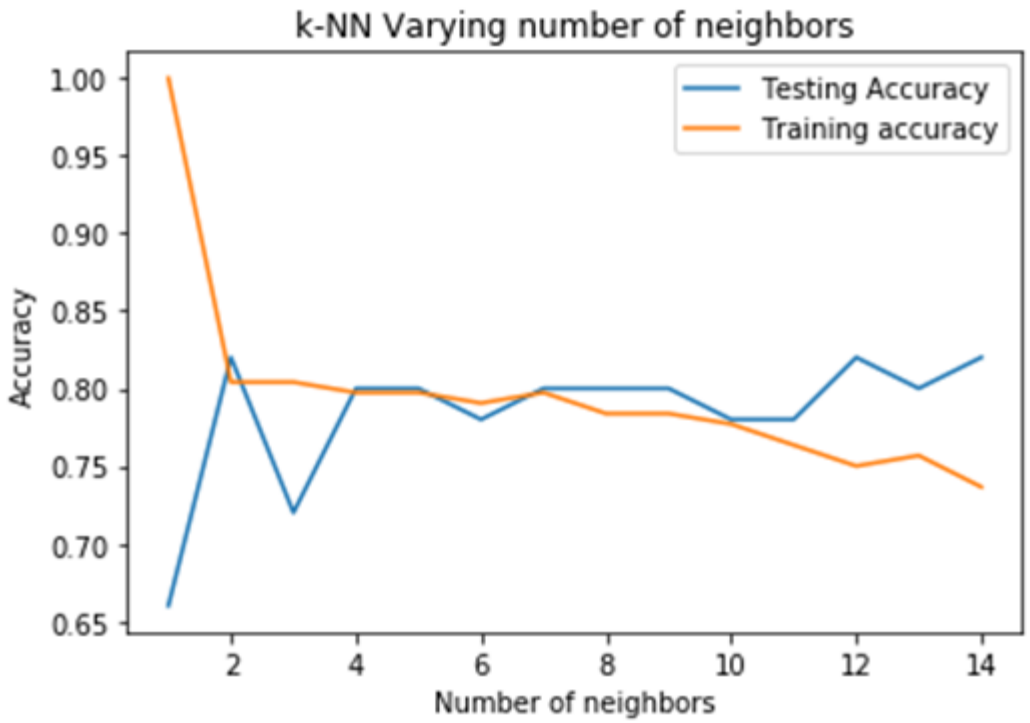


Figure 25. Best value for number of neighbours for testing and training accuracy.

F1 score for non-recur and recur was 0.90 and 0.31 respectively, and overall accuracy was 0.82. Five-fold cross-validation score was 0.737 (Figure 26).

```
KNN Testing accuracy:  0.82
KNN Training accuracy:  0.8040540540540541

              precision    recall  f1-score   support

           0       0.85      0.95      0.90        41
           1       0.50      0.22      0.31         9

    accuracy                           0.82        50
   macro avg       0.67      0.59      0.60        50
weighted avg       0.79      0.82      0.79        50


Cross-validation scores: [0.775      0.7        0.75       0.71794872 0.74358974]
Mean Cross-validation score: 0.7373076923076923
```

Figure 26. kNN testing, training, precision, recall, F1, and cross-validation score for non-recur (0) and recur (1) patients.

4) XGBoost Classifier

Hyperparameter tuning for XGBoost using randomized search (cv = 5) resulted in the following best parameters depicted in Figure 27:

```
{'min_child_weight': 7,
 'max_depth': 6,
 'learning_rate': 0.05,
 'gamma': 0.0,
 'colsample_bytree': 0.4}
```

Figure 27. Best hyperparameters for XGBoost using randomized search (cv = 5).

Testing and training accuracy were 0.86 and 0.80 respectively, f1-score for non-recur and recur was 0.92 and 0.36 respectively, overall f1-score was 0.86, and cross-validation score was 0.763 (Figure 28).

```
XGB Testing accuracy:  0.86
XGB Training accuracy:  0.8040540540540541

              precision    recall  f1-score   support

           0       0.85      1.00      0.92        41
           1       1.00      0.22      0.36         9

    accuracy                           0.86        50
   macro avg       0.93      0.61      0.64        50
weighted avg       0.88      0.86      0.82        50

Cross-validation scores: [0.75       0.775      0.7        0.79487179 0.79487179]
Mean Cross-validation score:  0.7629487179487178
```

Figure 28. XGBoost testing, training, precision, recall, F1, and cross-validation score for non-recur (0) and recur (1) patients.

5) Summary table

A summary table below displays cross-validation, accuracy, recall, precision, and F1-scores for KNN and XGBoost (Table 5). As XGBoost performed better than KNN based on the cross-validation scores, we chose XGBoost to predict cancer recurrence.

Table 5. Accuracy of kNN and XGBoost.

	Cross-validation	Accuracy	Recall	Precision	F1-score
KNN	0.737308	0.82	0.222222	0.5	0.307692
XGBoost	0.762949	0.86	0.222222	1.0	0.363636

6) Make predictions

A confusion matrix was presented to describe cancer prognosis prediction (Figure 29). The entire 41 patients who did not get cancer recurrence following treatment were correctly predicted. However, from the nine patients who did get cancer recurrence following treatment, only two of them had been correctly classified while seven of them had been erroneously diagnosed. This explains the low recall value for the recur group.

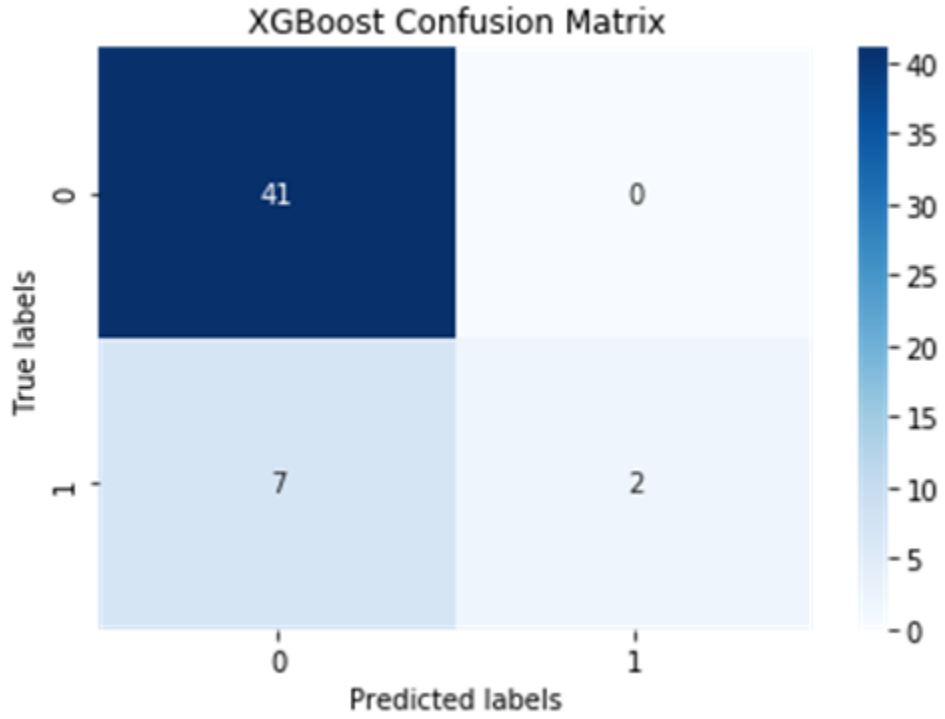


Figure 29. Confusion matrix of XGBoost.

7) Predicting when cancer will recur

We have been able to predict if cancer will recur with somewhat adequate accuracy. But can we predict the *time* cancer will recur? This question had to be tackled differently from our earlier

questions because the target variable was numerical as opposed to categorical. Since patients who had cancer recurrence were only relevant in our analysis, the rows where the outcome was in the 'recur' category were filtered. This selection reduced the dataset to 47 rows. Again, univariate feature selection was performed to select a subset of features that had the strongest relationship with the recurrence time variable. These features were worst area, mean area, standard error area, lymph node status, and worst perimeter (Figure 30).

	Specs	Score
23	area_worst	9598.486491
3	area_mean	4386.888397
13	area_se	1124.893190
31	lymph_node_status	225.103007
22	perimeter_worst	218.977097

Figure 30. Five best features for the target variable, recurrence time.

Each five features were plotted against the recurrence time variable (Figure 31) to visually decide what type of regression model was appropriate for our analysis.

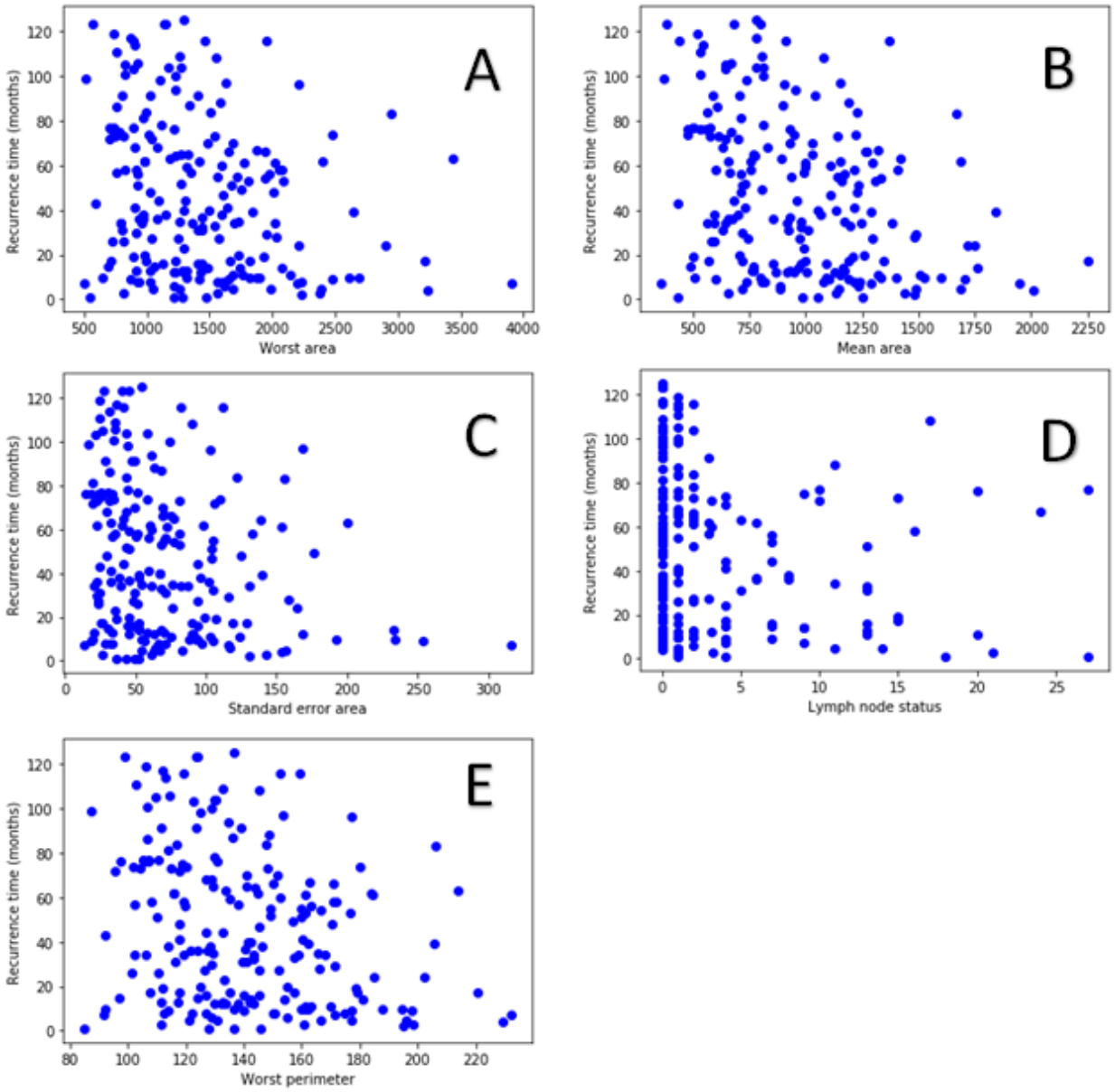


Figure 31. Scatter plots of five best features: A) worst area, B) mean area, C) standard error area, D) lymph node status, and E) worst perimeter against recurrence time.

It is unclear whether the relationship is linear or polynomial from visually examining the scatter plots. We tried using polynomial regression of degree two. The result displayed in Table 6 reports the accuracy of this model. Mean Absolute Error was 41.08, Mean Squared Error was 2338.74, and R^2 score was -2.10.

Table X. Table showing degree of polynomial, MAE, MSE, and R2.

	Degree	MAE	MSE	R2
Polynominal regression	2	41.084732	2338.740191	-2.103094

The Mean Absolute Error and Mean Squared Error values appeared fairly large, and the R^2 score was a negative value. From these results, it is inferred that the polynomial regression model did not closely fit my data as the data values did not closely represent the fitted polynomial regression line. Consequently, I do not think we can accurately predict the time of recurrence from this model due to limitations, which will be further addressed in the discussion.

7 Discussion

Part 1: Cancer diagnosis

Contrary to our assumption, logistic regression had the lowest score, while kNN classifier had the best performance. The explanation for logistic regression's low performance may be due to the reduction in the number of features to three to avoid feature dependencies. When we ran the model without taking multicollinearity into account, the accuracy score for logistic regression was comparatively higher. Some researchers that have used logistic regression with the same datasets reached accuracy as high as 0.98, however, they did not take into consideration feature dependencies in the dataset, which insinuated that the model was less reliable and potentially numerically unstable. The rationale for kNN's high performance may be due to feature selection in combination with kNN that resulted in improvement of its predictive accuracy. Furthermore, all numeric features, no missing values and minimal noisy data, may have also contributed to kNN's high performance.

In terms of limitations, there were several that were noted. One important limitation to accentuate was the occurrence of overfitting, particularly in the gradient boosting and Xgboost models. Overfitting was the result of our high training accuracy, as the model was excessively trained to the data, capturing noise, and making the model less generalized. This overfitting may have been due to extreme outliers present in the data. Because it was anticipated that removal of outliers would reduce overfitting, we ran all models with the removed outliers. Nevertheless, this modification reduced the sample size from 569 to 277 and more overlap between the malignant and benign classes were evident (Figure 32). This significant reduction in sample size and more overlap between the two classes resulted in lower performance, and hence, we decided to keep the outliers in our analysis. Moreover, we have also tried increasing the size of training dataset to reduce overfitting, however, increasing training set did not have any effect, therefore, we decided to keep the size as default 75%.

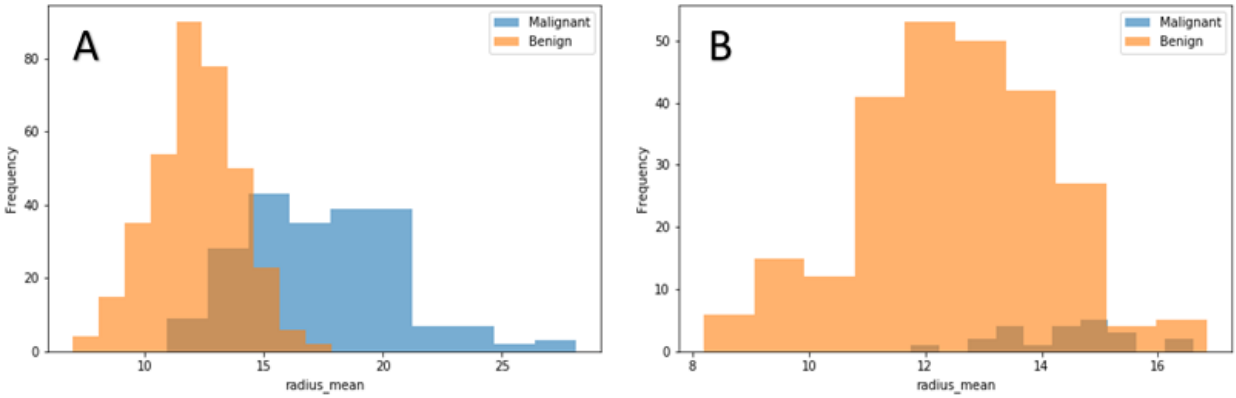


Figure 32. Histogram of the mean radius feature showing significant overlap between the Malignant and Benign classes after removal of outliers (B). (A) shows less overlap when outliers were present.

Part 2: Cancer prognosis

As a continuation of the diagnosis study, cancer prognosis was also investigated. We had seen that XGBoost had slightly better performance (76.3% cross-validation accuracy) compared to kNN (73.7%). However, in both models, recall score was substantially low (22% for both) for the 'Recur' patients group while relatively high recall score was observed for 'Non-recur' patients group (95% and 100% for kNN and XGBoost respectively). Explanation for this low recall score in 'Recur' group was almost certainly due to the small sample size. There were only nine patients in the 'Recur' class as opposed to 41 patients in the 'Non-recur' class.

Although we could have used logistic regression in this study since logistic regression seemed suitable for this research, we did not end up choosing this model because there were many features that were significantly correlated with each other. For example, from the five selected features in our dataset, four features (area_worst, area_mean, area_se, and perimeter_worst), were highly dependent with one another. Removing these features would result in training a model with only one feature, which would not be ideal as it may lead to overfitting.

The second part of this question was concerned with predicting time for cancer recurrence. As addressed in the analysis section, no distinct trend was noticeable when the features were plotted against the time variable. We estimated perhaps using a polynomial regression model of degree two may potentially work, however, as previously denoted, the results produced a considerably high error values and low R^2 accuracy score. For limitations for our prognosis study, it is worth noting that although patients were labeled as nonrecurrent (N), it is never certain if the disease was in fact, nonrecurrent for these patients. It is possible that cancer recurred later in time after the study was complete, but this data would not be documented in the study. In addition, there is

no specific cutoff value at which point the patient is considered a nonrecurrent case as it is simply the time of their last check-up. One final point to denote is the fact that cancer recurrence occurred at a moment in time before it has actually been detected. The values in “time to recur" field is not the actual time the cancer recurred, but the time that the recurrence was detected, although it is assumed that this difference between the actual and detected time is small.

For future research, it would be worthwhile and interesting to study the relationship between gene expression and breast cancer. Since cancer diagnosis correlates with gene expression, our impetus for this further study may prove more effective than looking at cells from tissues. (Mazzanti et al., 2004).

8 Self Assessment

Introduction
Writing - Robyn, Lydia
Background research - Robyn, Lydia

Data description
Writing – Lydia
Review – Nick

Exploratory data analysis
Analysis – Lydia
Writing – Lydia
Review - Nick

Research questions
Lydia, Nick, Robyn

Techniques
Writing for both part 1 and 2 – Lydia

Analysis
Writing for question 1 - Lydia
Writing for question 2 - Lydia
Writing for question 3 - Lydia
Analysis for all 3 questions – Lydia

Discussion
Lydia

.

9 References

Government of Canada (2019). Breast cancer. Retrieved from https://www.canada.ca/en/public-health/services/chronic-diseases/cancer/breast-cancer.html

Mazzanti, C., Zeiger, A. M., Costourous, N., …Libutti, S. K. (2004). Using Gene Expression Profiling to Differentiate Benign versus Malignant Thyroid Tumors. *Cancer Research*, 64(8), 2898-2903.

Scikit-learn (2020). Post pruning decision trees with cost complexity pruning. Retrieved from https://scikit-learn.org/stable/auto_examples/tree/plot_cost_complexity_pruning.html

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society*, 36(2), 111-147.

Street W. N., Wolberg W. H., & Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. *IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology*, 1905, 861-870.

UCI Machine Learning Repository (1995). Breast Cancer Wisconsin (Diagnostic) Data Set. *Archive.Ics.Uci.Edu*, 2020, Retrieved from https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic).

Wolberg W. H., Street W. N., Heisey, D., & Mangasarian, O. L. (1994). Machine learning techniques to diagnose breast cancer from fine-needle aspirates. *Cancer Letters*, 77, 163-171.

Wolberg W. H., Street W. N., Heisey, D., & Mangasarian, O. L. (1995). Image analysis and machine learning applied to breast cancer diagnosis and prognosis. *Analytical and Quantitative Cytology and Histology*, 17(2), 77-87.

Wolberg W. H., Street W. N., Heisey, D., & Mangasarian, O. L. (1995). Computerized breast cancer diagnosis and prognosis from fine needle aspirates. *Archives of Surgery*, 130(5), 511-516.

Wolberg W. H., Street W. N., Heisey, D., & Mangasarian, O. L. (1995). Computer-derived nuclear features distinguish malignant from benign breast cytology. *Human Pathology*, 26(7), 792-796.

Wolberg W. H., Street W. N., Heisey, D., & Mangasarian, O. L. (1995). Breast Cancer Diagnosis and Prognosis via Linear Programming. *Operations Research*, 43(4), 570–577.

World Health Organization (2020). Breast cancer. Retrieved from https://www.who.int/cancer/prevention/diagnosis-screening/breast-cancer/en/

10 Appendix

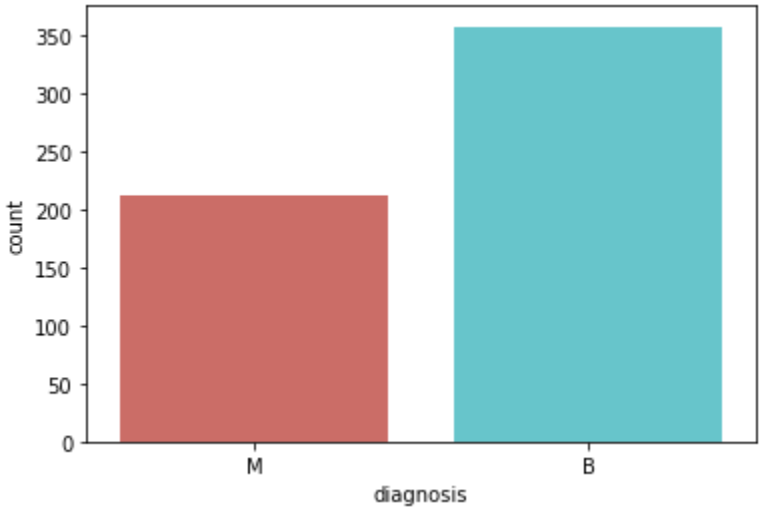


Figure 33. Count of Malignant (M) and Benign (B) samples.

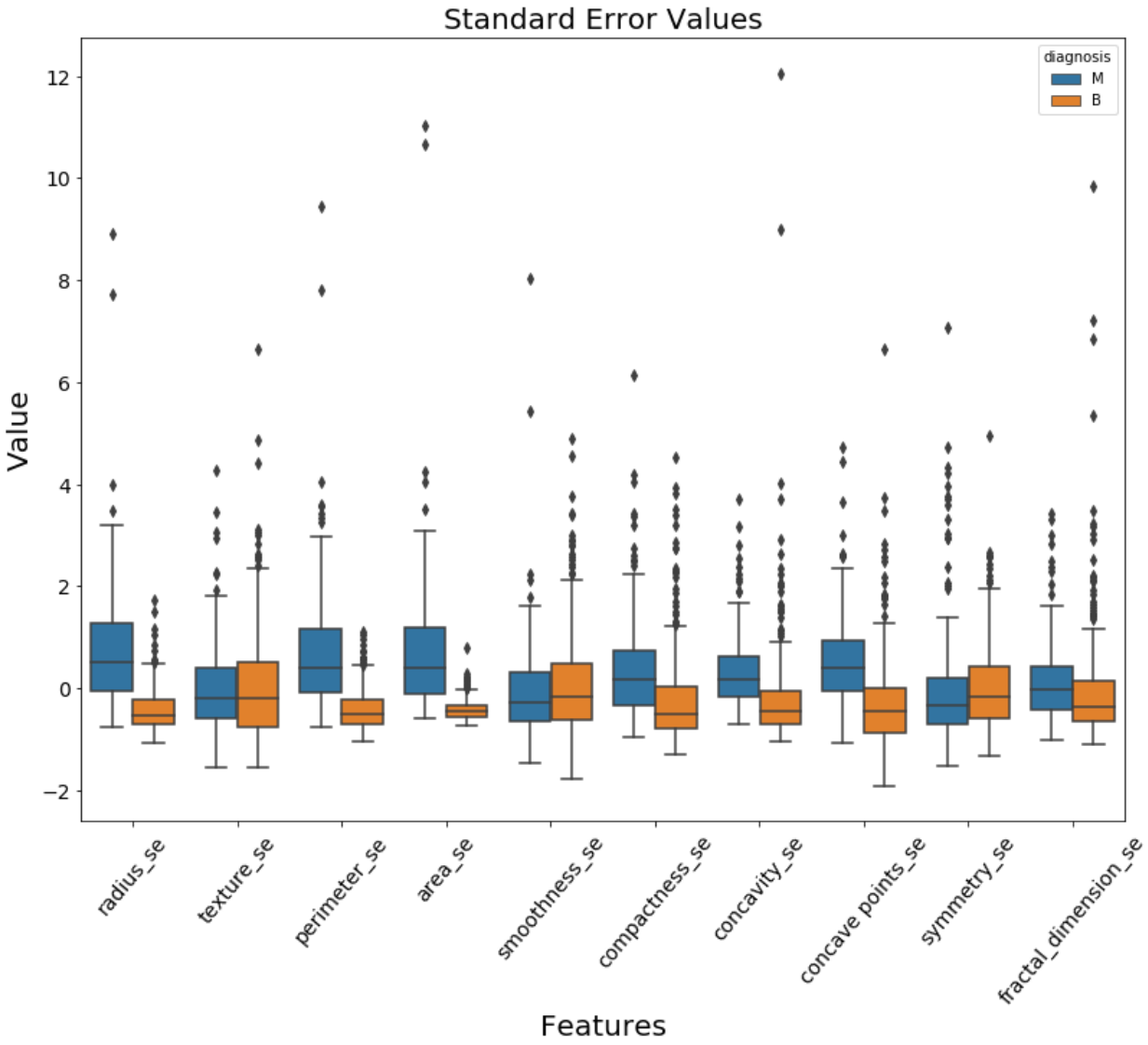


Figure 34. Box plot illustrating the difference between Malignant (M) and Benign (B) cell nucleus based on standard error attributes.

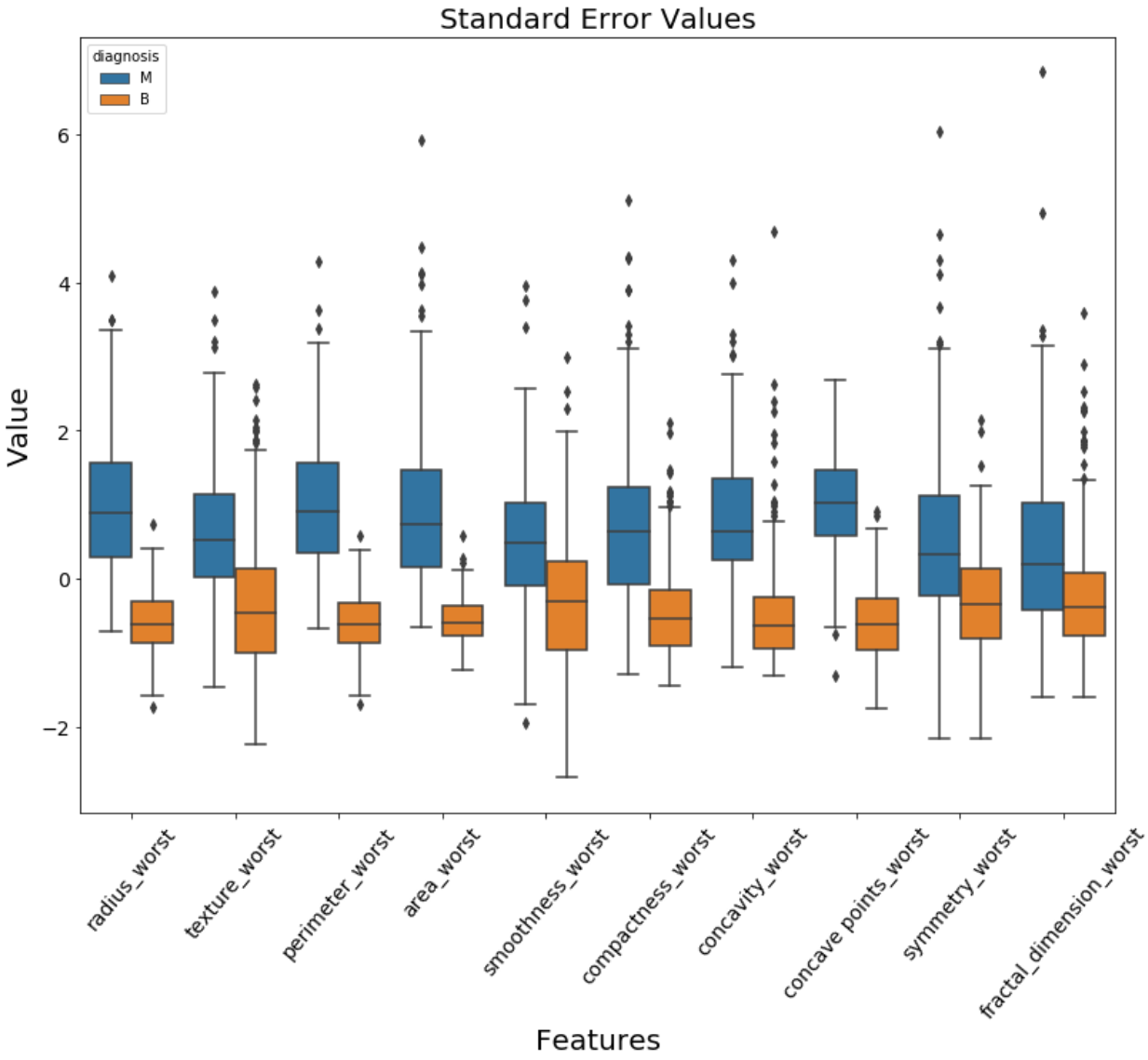
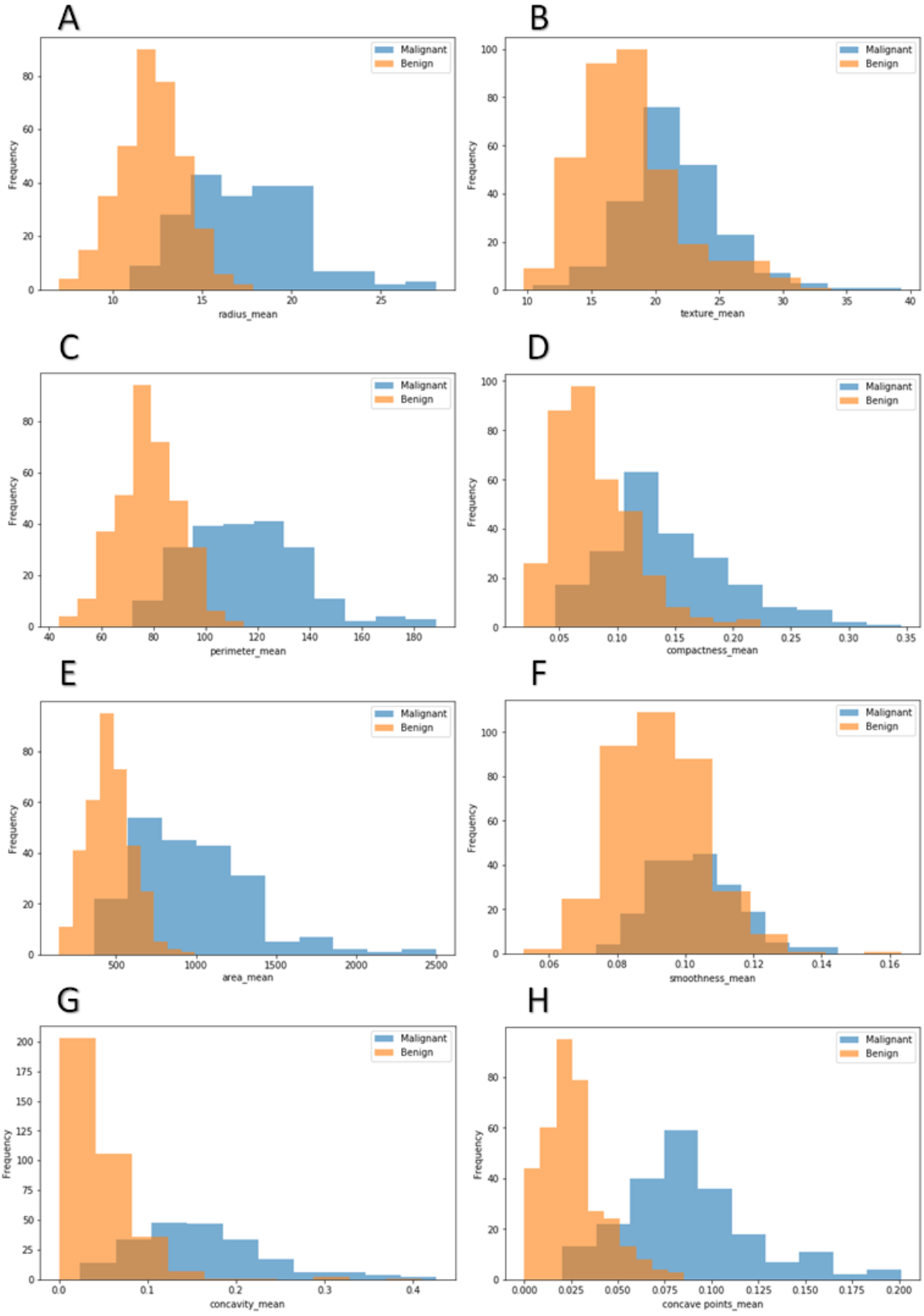


Figure 35. Box plot illustrating the difference between Malignant (M) and Benign (B) cell nucleus based on worst attributes.


A
Malignant
Benign
Frequency
radius_mean
B
texture_mean
C
perimeter_mean
D
compactness_mean
E
area_mean
F
smoothness_mean
G
concavity_mean
H
concave points_mean

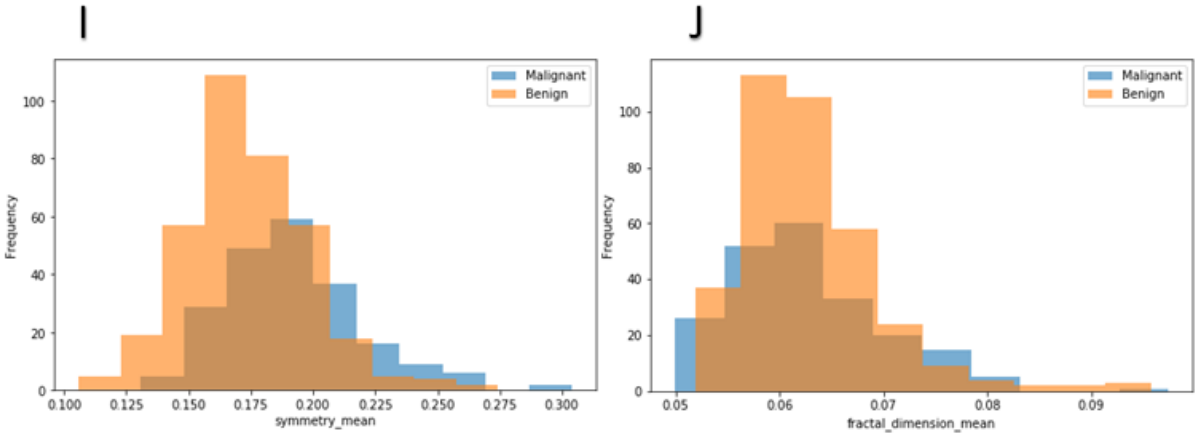


Figure 36A-J. Histogram showing frequency distribution of Malignant and Benign cell nucleus mean of radius (A), texture (B), perimeter (C), compactness (D), area (E), smoothness (F), concavity (G), concave points (H), symmetry (I), fractal dimension (J).

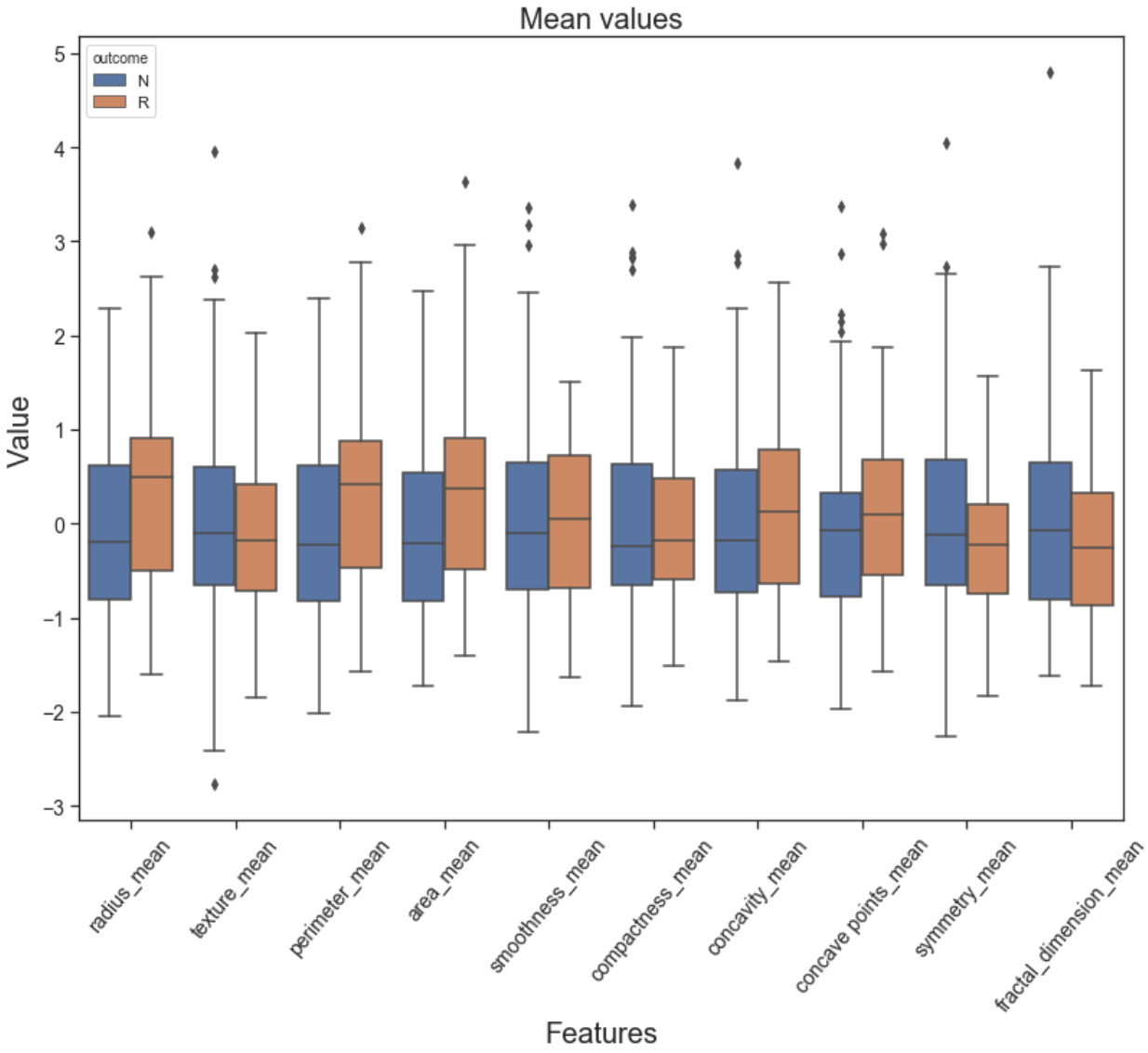


Figure 37. Box plot illustrating the difference between Non-recur (N) and Recur (N) cell nucleus based on mean attributes.

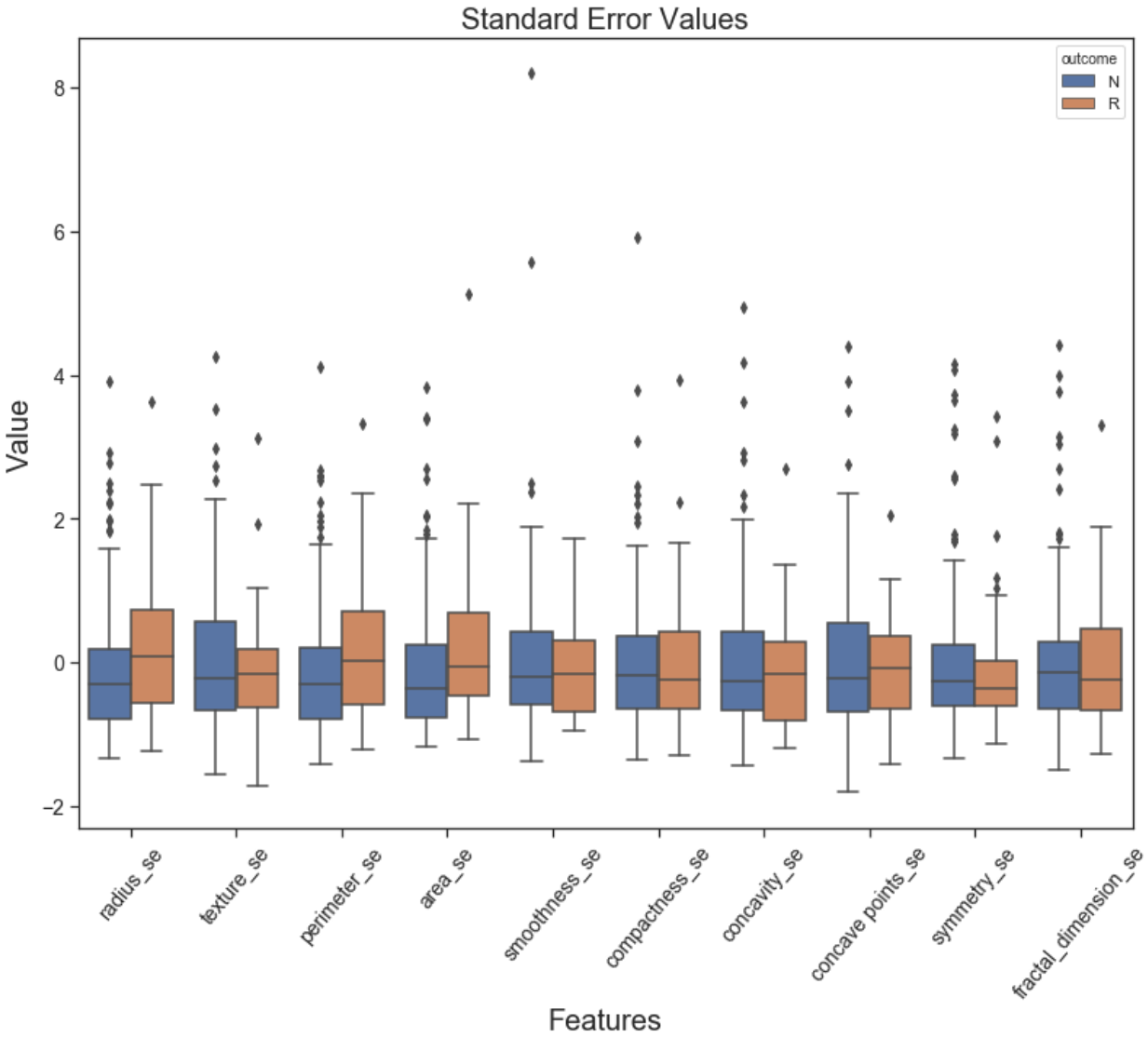


Figure 38. Box plot illustrating the difference between Non-recur (N) and Recur (N) cell nucleus based on standard error attributes.

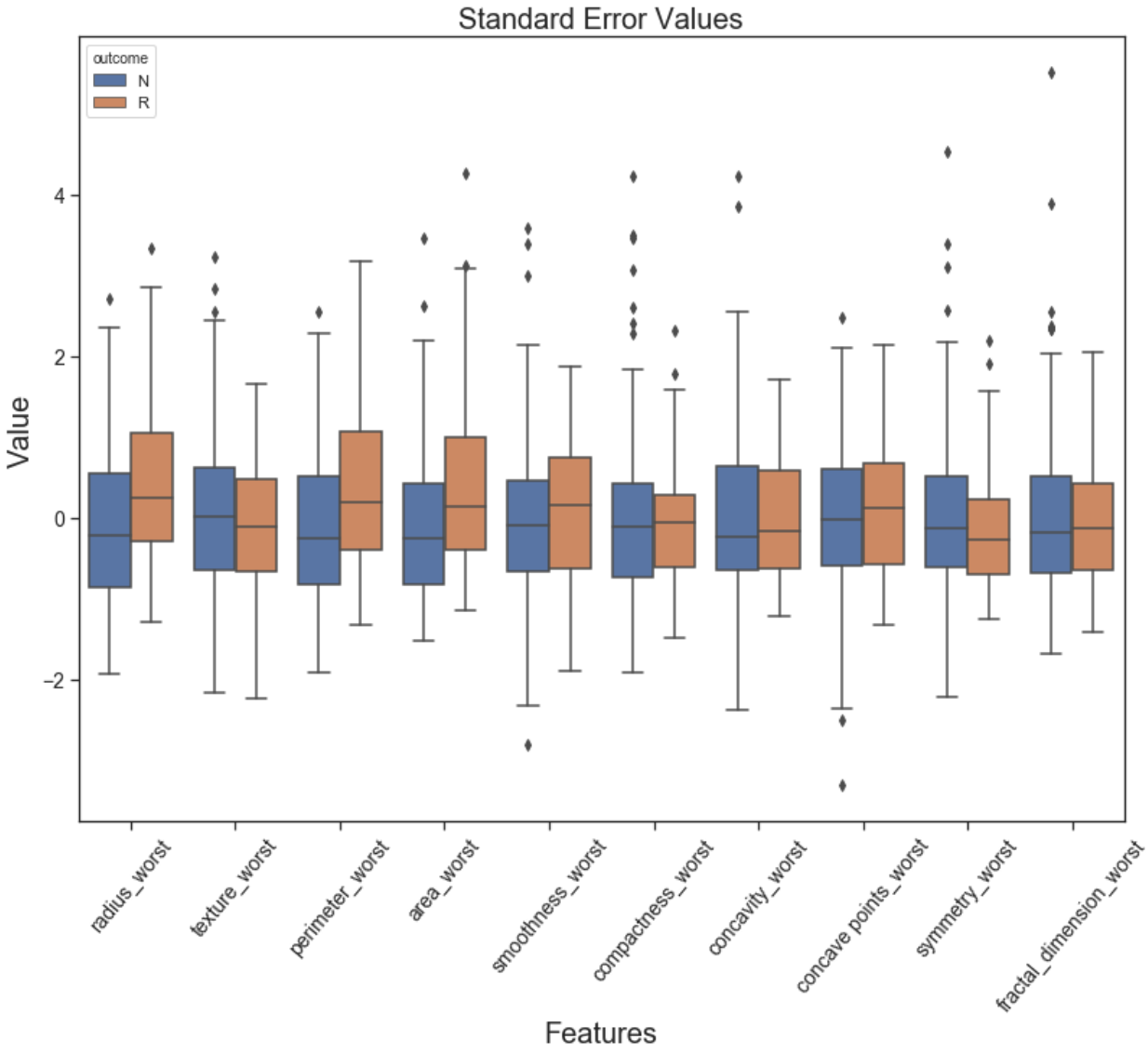


Figure 39. Box plot illustrating the difference between Non-recur (N) and Recur (N) cell nucleus based on worst attributes.

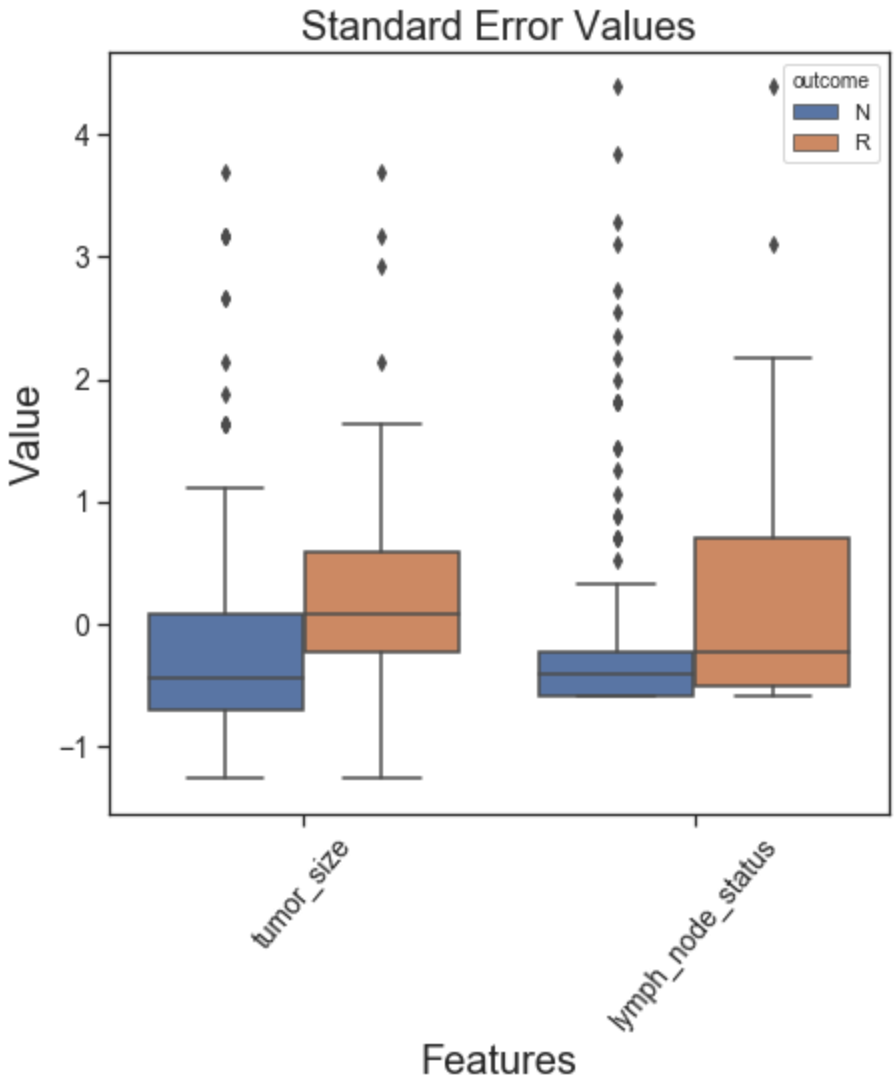


Figure 40. Box plot illustrating the difference between Non-recur (N) and Recur (N) cell nucleus based on tumor size and lymph node status.

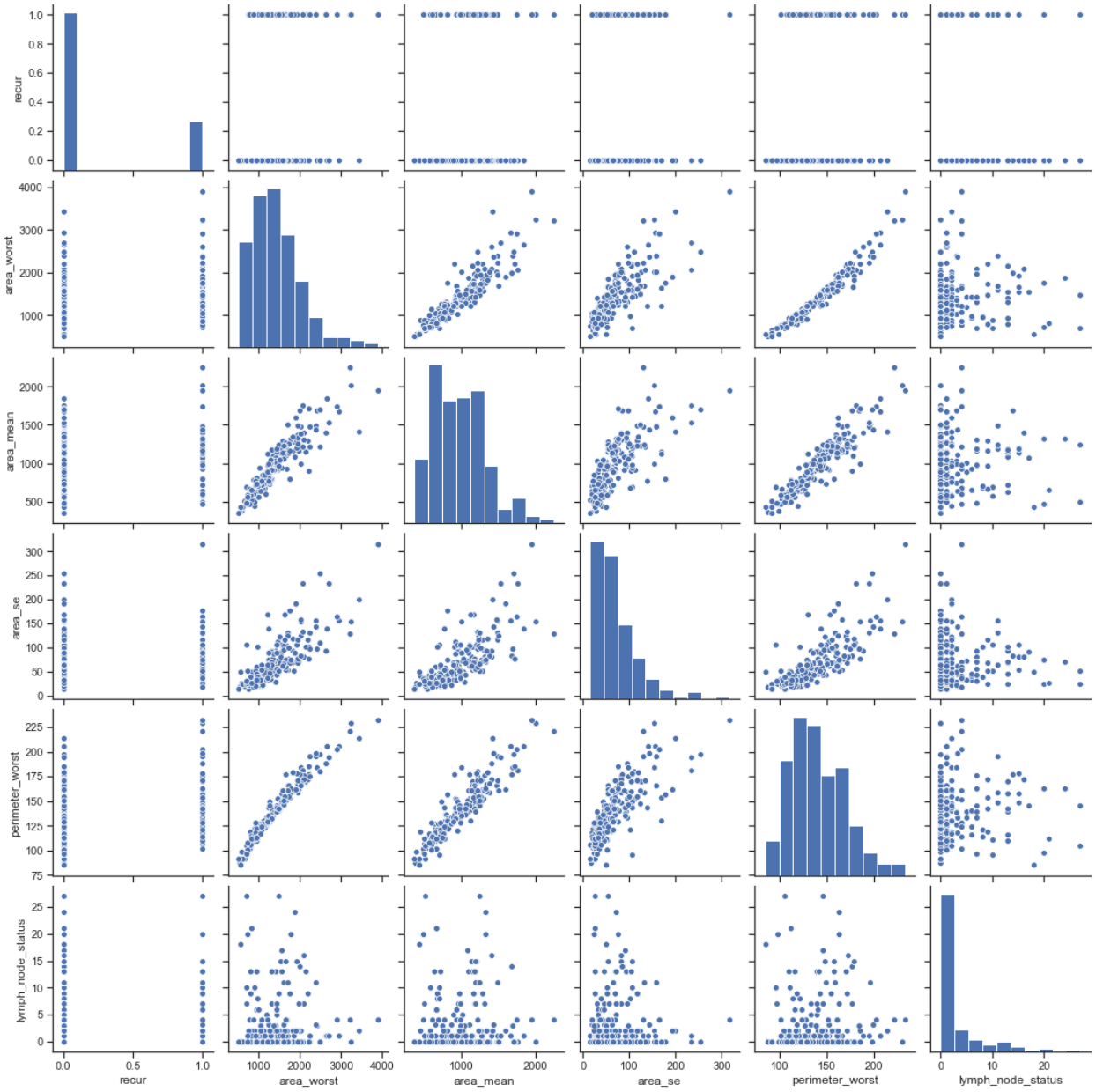


Figure 41. Scatterplot matrix showing the correlation between features.