Machine Learning and Pattern Recognition
Assignment 3
COMP 8740

Pledge: “As a student of the University of Windsor, I pledge to pursue all endeavors with honor and integrity, and will not tolerate or engage in academic or personal dishonesty. I confirm that I have not received any unauthorized assistance in preparing for or writing this assignment. I acknowledge that a mark of 0 may be assigned for copied work.”

Ankur Mangroliya (110127190)

Group	Name	Student ID
1	Gagandeep Singh	110123330
2	Deon Victor Lobo	110127749
3	Akshar Patel	110126131
4	Ankur Mangroliya	110127190

Introduction :

In this report, we will explore supervised classification techniques using Scikit-learn. After downloading various datasets, including circles, moons, and spirals, we will apply Decision Tree, Random Forest, and Neural Network classifiers, specifying the NN architecture and evaluating their performance through 10-fold cross-validation. Subsequently, we will also focus on the breast cancer dataset, where two feature selection methods like Chi-squared and mrmr will be applied. These selected features will be combined with Random Forest, their performance will be assessed using multi-class metrics such as PPV, NPV, specificity, sensitivity, and accuracy. This project aims to provide a comprehensive understanding of classifier performance, feature selection methods, and their application in real-world datasets.

Implementation:

The provided code defines three classifiers (Decision Tree, Random Forest, Neural Network) with specific configurations and hyperparameters. These classifiers are stored in a dictionary called models. The code then iterates through different datasets, reads the data, and separates features and labels. For each dataset, the code further iterates through each model, performs cross-validation, and extracts key performance metrics (PPV, NPV, Specificity, Sensitivity, and Accuracy). The decision boundary for the first estimator of each model is visualized. Results, including model names, dataset information, and average metrics, are stored and printed as tables. The code aims to assess and compare the performance of different classifiers on various datasets, providing insights into their effectiveness.

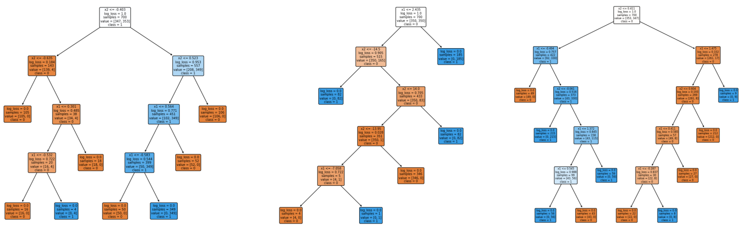
Question No. 1-3: DecisionTreeClassifier:

First we have used decisionTreeClassifier for binary classification. It works by recursively partitioning the input space into regions and assigning a label or value to each region. The algorithm makes decisions based on features of the input data, attempting to create a tree-like structure where each internal node represents a decision based on a particular feature, each branch represents the outcome of the decision, and each leaf node represents the final classification.

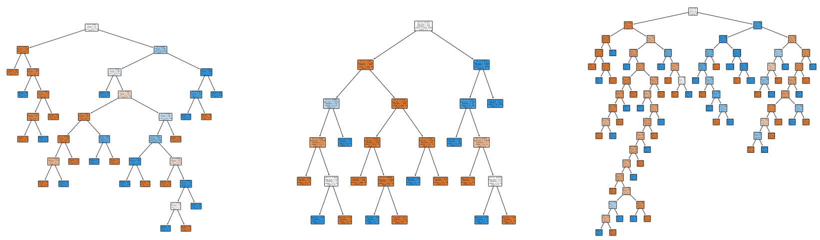
Parameters used :

- Criterion :**“log_loss”** as the criterion function, measuring the logarithm of the likelihood of true labels given predicted probabilities. This function needs to be minimized to get better model accuracy. The Decision Tree classifier is configured with the 'log_loss' criterion, aiming to optimize for log loss, which is suitable for classification tasks.
- Splitter :**“best”** as the splitter, used to take the optimal values while splitting to internal nodes.
- it was giving the best possible accuracy for different datasets.
- Max Depth: None (i.e., the tree grows until all leaves are pure or contain fewer samples than min_samples_split).
- Min Samples Split: 5 (minimum number of samples required to split an internal node).
- Min Samples Leaf: 1 (minimum number of samples required to be at a leaf node).
- Max Features: 2 (number of features to consider for the best split at each node).
- Random State: 25 (for reproducibility).

Below are shown the decisionTrees made by the classifier model.

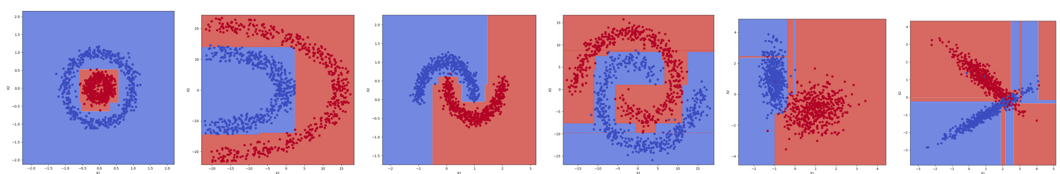


Circles dataset DT HalfKernal dataset DT Moon dataset DT



Spiral dataset DT twoGaussian33 dataset DT twoGaussian42 dataset DT

Below is the scatter plot showing the converged functions for different dataset (highlighted in the red and blue)



As per the above plots we can see that the decision tree classifier was able to converge to the given data pattern but it was not perfect as compared to other Classifiers.

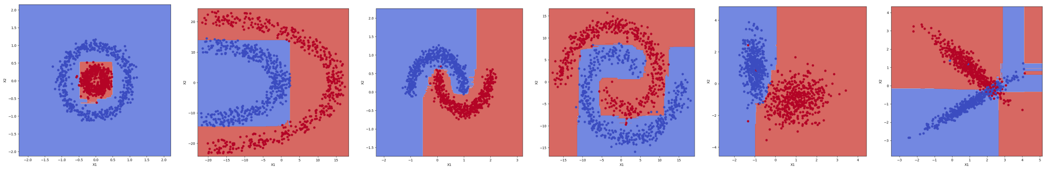
RandomForestClassifier:

Next we have used RandomForestClassifier. Basically it extends the idea of decisionTree and is an ensemble machine learning model that builds multiple decision trees during training and outputs the class that is the mode of the classes (classification) of the individual trees. It improves accuracy and generalization by combining the predictions of diverse trees.

Parameters used:

- Criterion: "**log_loss**" as the criterion function as it was giving the best accuracy as compared to "gini" and "entropy".
- Number of Estimators: 300 (number of trees in the forest).
- Max Depth: None (similar to Decision Tree, the trees in the forest are grown until min_samples_split conditions are met).
- Min Samples Split: 50 (minimum number of samples required to split an internal node).
- Min Samples Leaf: 10 (minimum number of samples required to be at a leaf node).
- Max Features: 'sqrt' (square root of the number of features to consider for the best split at each node).
- Bootstrap: True (whether bootstrap samples are used when building trees).
- Random State: 42 (for reproducibility).
- Number of Jobs: -1 (utilize all available processors for parallel processing).

Below is shown the scatter plot for the converged randomForest function for different dataset.



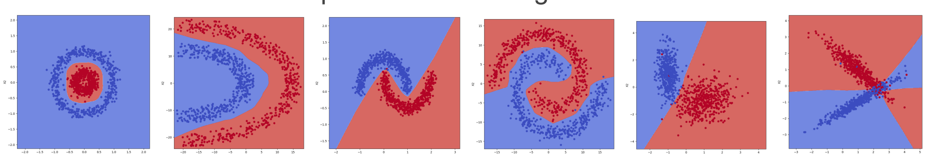
As per the plot we can see the functions are very similar to decisionTrees as it takes multiple decisionTrees into consideration while calculating the accuracy and creating the function.

MLPClassifier:
Further we are using a multi layer perceptron to classify different dataset. In this we have used 2 hidden layers, one with 40 neurons and one with 30 neurons.As the patterns are not very complex 2 hidden layers would be enough. We tried multiple combinations of the number of hidden layers and number of neurons but this looks like a good value as the model is giving 100% accuracy for most of the datasets. In this connection weights are updated based on back-propagation algorithms during training.

In this we have experimented with below parameters:

- Hidden Layer Sizes: (40, 30) (two hidden layers with 40 and 30 neurons, respectively).
- “max_iter”: 1500 has been taken as the number of iterations during the learning phase. This will help the model to converge properly and understand the features of the dataset.

Below is shown the scatter plot for the converged MLP function for different dataset.



As per the above plot we can see that the multi layer perceptron is effectively able to understand the pattern of the different dataset it gives 100% accuracy for most of the dataset with only two hidden layers and 70 hidden neurons in total.

Below are different parameters for three classifiers on all dataset

**********************************circles0.3.csv**

	Classifier	Dataset	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Desition Tree	circles0.3.csv	0.993151	0.987013	0.993464	0.986395	0.99
1	Random Forest	circles0.3.csv	1	0.974522	1	0.972789	0.986667
2	Neural Network	circles0.3.csv	1	1	1	1	1

**********************************halfkernel.csv**

	Classifier	Dataset	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Desition Tree	halfkernel.csv	0.993377	1	0.993333	1	0.996667
1	Random Forest	halfkernel.csv	0.993377	1	0.993333	1	0.996667
2	Neural Network	halfkernel.csv	1	1	1	1	1

moons1.csv

	Classifier	Dataset	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Desition Tree	moons1.csv	0.987013	0.993151	0.986395	0.993464	0.99
1	Random Forest	moons1.csv	0.941558	0.945205	0.938776	0.947712	0.943333
2	Neural Network	moons1.csv	0.993464	0.993197	0.993197	0.993464	0.993333

spiral1.csv

	Classifier	Dataset	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Desition Tree	spiral1.csv	0.986486	0.986842	0.986842	0.986486	0.986667
1	Random Forest	spiral1.csv	0.919463	0.927152	0.921053	0.925676	0.923333
2	Neural Network	spiral1.csv	1	0.993464	1	0.993243	0.996667

twogaussians33.csv

	Classifier	Dataset	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Desition Tree	twogaussians33.csv	0.973684	0.972973	0.972973	0.973684	0.973333
1	Random Forest	twogaussians33.csv	0.980132	0.973154	0.97973	0.973684	0.976667
2	Neural Network	twogaussians33.csv	0.993377	0.986577	0.993243	0.986842	0.99

twogaussians42.csv

	Classifier	Dataset	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Desition Tree	twogaussians42.csv	0.892617	0.86755	0.891156	0.869281	0.88
1	Random Forest	twogaussians42.csv	0.92029	0.839506	0.92517	0.830065	0.876667
2	Neural Network	twogaussians42.csv	0.965753	0.922078	0.965986	0.921569	0.943333

By examining the accuracy metric among the classifiers with the specified parameters, it becomes evident that the Neural Network outperforms the Decision Tree and Random Forest across all datasets. The Neural Network, with its capacity to capture intricate patterns in data, demonstrates superior classification performance. However, it's worth noting that the dataset characteristics play a crucial role in model effectiveness.

In this scenario, the Decision Tree and Random Forest classifiers exhibit similar accuracies. The Random Forest, designed to enhance predictive performance by aggregating multiple decision trees, doesn't manifest a substantial advantage over the Decision Tree. This outcome is likely due to the dataset's simplicity; the complexity that Random Forests excel in handling might not be fully utilized in this context. Consequently, both Decision Tree and Random Forest achieve comparable accuracies, showcasing that, in situations with less intricate data, the additional sophistication of a Random Forest may not necessarily yield significant improvements.

.

Question No. 4-6 : Report and comment on the results obtained for Breast cancer dataset

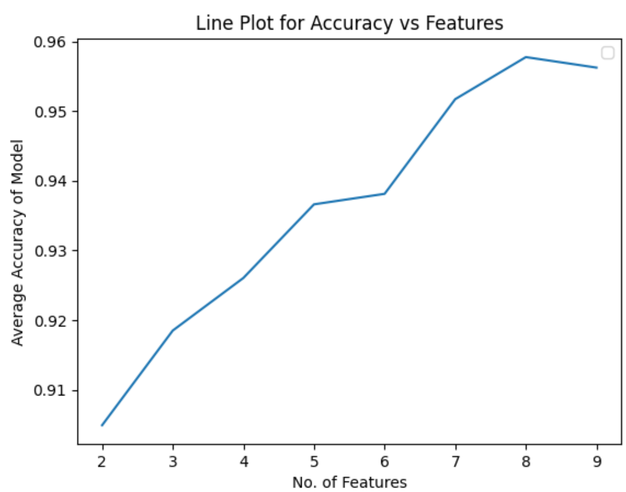
Why did we choose One-vs-One (OvO) ?

- **Binary Nature of Random Forest:** Random Forest is inherently capable of handling multi-class classification problems. It internally builds multiple decision trees, each distinguishing between different classes. Hence, utilizing OvO with Random Forest aligns well with its natural capabilities.
- **Computational Overhead:** If the number of classes is very high, OvO might result in a larger number of classifiers, which could lead to increased computational overhead. In our case we have only 5 classes which is why there will be n(n-1)/2 i.e. 10 combinations of these classes which is not much. Hence we can easily implement this for small to moderate-sized datasets.
- **Simpler Interpretation:** OvO might provide a clearer interpretation of how well the Random Forest model distinguishes between each pair of classes. This could be beneficial for understanding the nuances of class relationships.
- **Potentially Improved Accuracy:** OvO may lead to better accuracy in scenarios where the dataset is not excessively large like in our case there are only 159 rows, and the Random Forest algorithm is not computationally prohibitive.
- **Binary Classifier Suitability:** OvO trains multiple binary classifiers, each distinguishing between two classes. If the base binary classifier is particularly well-suited for binary tasks (e.g., good at handling imbalanced data), OvO might perform better.
- **Class Imbalance:** OvO might be more suitable when dealing with imbalanced classes because each binary classifier focuses on a specific pair of classes. This allows for more balanced training sets.

- **Complex Decision Boundaries:** OvO might be beneficial when the decision boundaries between classes are complex and overlapping. Each binary classifier only needs to focus on distinguishing between two classes, potentially capturing finer details.
- **Algorithm Behavior:**Some machine learning algorithms might behave differently with OvO compared to OvA. For example, certain algorithms might benefit from having more focused training sets in the OvO strategy.
- **Error Correlation:** If errors in classification are correlated (i.e., a model tends to confuse certain pairs of classes), OvO might be more resilient as classifiers focusing on other class pairs are unaffected.
- **Scalability:** In scenarios with a moderate number of classes, OvO might be more scalable. However, for a very large number of classes, the computational cost of training and testingN×(N−1)/2 classifiers in OvO can be significant.

Handling class imbalance :

The Synthetic Minority Over-sampling Technique (SMOTE) from the imbalanced-learn library is utilized to address class imbalance in the dataset. By creating an instance of the SMOTE algorithm with a designated random state (random_state=42), the imbalanced class distribution is rebalanced. The application of SMOTE is crucial in mitigating issues associated with imbalanced class distributions, enhancing the model's ability to generalize across different classes during classification tasks.



Implementation of Chi2 With RandomForest and OneVsOne scheme:

In this first we calculated the chi2 value for each feature and arranged it in descending order and adding one feature at a time we applied 10 cross validation using OneVsOne method. We found that the average accuracy of the model is best at 8 features as seen in the graph below:

As we can see the average accuracy increases with the increase in features and after 8 features the accuracy can be seen to decrease. Max average accuracy is around 96%.

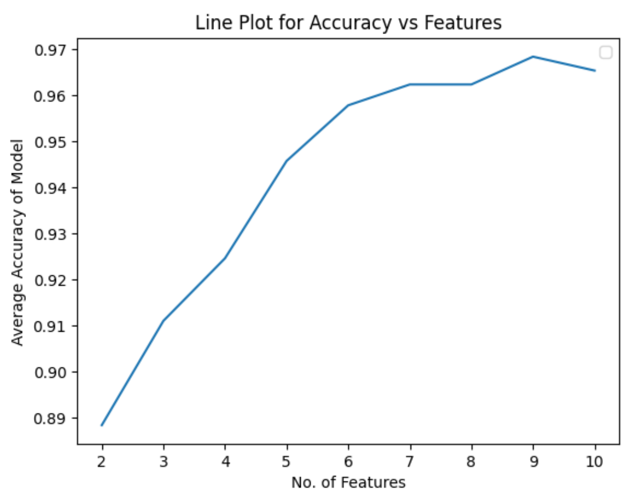
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	8	0.981481	1	0.995283	1	0.996226
1	Her2	RF	BreastCancer	8	0.962264	0.990566	0.990566	0.962264	0.984906
2	LumA	RF	BreastCancer	8	0.784314	0.939252	0.948113	0.754717	0.909434
3	LumB	RF	BreastCancer	8	0.777778	0.947867	0.943396	0.792453	0.913208
4	Normal	RF	BreastCancer	8	0.962264	0.990566	0.990566	0.962264	0.984906

Below are the 8 features with where selected for classification:

Features	**Chi2 value**
AGR3	22.489446
FOXA1	20.483725
MIA	19.793050
ATL2	19.187792
PPP1R14C	17.323979
ROPN1	16.369230
SOX11	16.284443
WNT6	16.229333

Implementation of mrmr With RandomForest and OneVsOne scheme :

In this script, the mRMR (minimum Redundancy Maximum Relevance) feature selection method is applied to a resampled dataset (X_resampled and y_resampled). Subsequently, a loop iterates over different numbers of features (n), where for each iteration, a Random Forest classifier is initialized and integrated into a One-vs-One (OvO) multiclass strategy. Cross-validation predictions are made, and a confusion matrix is generated. After calculating the accuracies for each feature selected we observe that the best accuracy is with 9 with an accuracy of 97%. As we can observe from the graph of accuracies vs the number of features the accuracy keeps increasing till 9 features and for the 10th feature selected it starts decreasing.



	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	9	1	0.995305	1	0.981132	0.996226
1	Her2	RF	BreastCancer	9	0.962264	0.990566	0.990566	0.962264	0.984906
2	LumA	RF	BreastCancer	9	0.86	0.953488	0.966981	0.811321	0.935849
3	LumB	RF	BreastCancer	9	0.836364	0.966667	0.957547	0.867925	0.939623
4	Normal	RF	BreastCancer	9	0.945455	0.995238	0.985849	0.981132	0.984906

Below are the 8 features with where selected for classification:

FOXA1, DST, NDP, PMAIP1, AGR3, CENPF, MIA, TBC1D9, MUCL1, ATL2, NDC80, AR, GATA3, ZNF238, TNXB

Appendix

Chi2 With RandomForest and OneVsOne scheme Outputs

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	2	0.944444	0.990521	0.985849	0.962264	0.981132
1	Her2	RF	BreastCancer	2	0.803922	0.943925	0.95283	0.773585	0.916981
2	LumA	RF	BreastCancer	2	0.666667	0.895455	0.929245	0.566038	0.856604
3	LumB	RF	BreastCancer	2	0.583333	0.912195	0.882075	0.660377	0.837736
4	Normal	RF	BreastCancer	2	0.818182	0.961905	0.95283	0.849057	0.932075

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	3	0.944444	0.990521	0.985849	0.962264	0.981132
1	Her2	RF	BreastCancer	3	0.785714	0.956938	0.943396	0.830189	0.920755
2	LumA	RF	BreastCancer	3	0.673469	0.907407	0.924528	0.622642	0.864151
3	LumB	RF	BreastCancer	3	0.692308	0.920188	0.924528	0.679245	0.875472
4	Normal	RF	BreastCancer	3	0.87037	0.971564	0.966981	0.886792	0.950943

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	4	0.962963	0.995261	0.990566	0.981132	0.988679
1	Her2	RF	BreastCancer	4	0.821429	0.966507	0.95283	0.867925	0.935849
2	LumA	RF	BreastCancer	4	0.733333	0.909091	0.943396	0.622642	0.879245
3	LumB	RF	BreastCancer	4	0.649123	0.923077	0.90566	0.698113	0.864151
4	Normal	RF	BreastCancer	4	0.90566	0.976415	0.976415	0.90566	0.962264

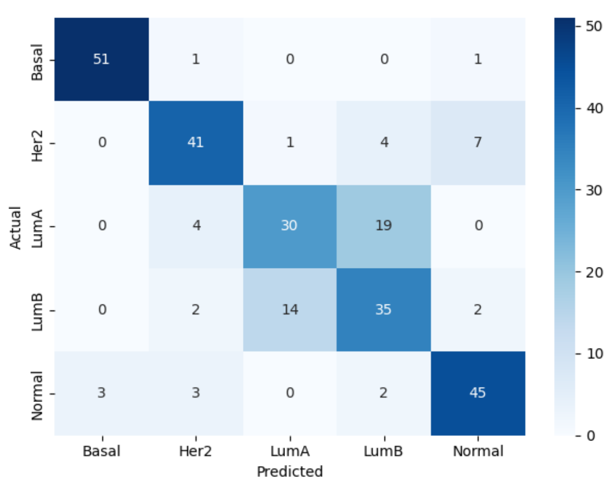
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	5	0.962963	0.995261	0.990566	0.981132	0.988679
1	Her2	RF	BreastCancer	5	0.888889	0.976303	0.971698	0.90566	0.958491
2	LumA	RF	BreastCancer	5	0.723404	0.912844	0.938679	0.641509	0.879245
3	LumB	RF	BreastCancer	5	0.719298	0.942308	0.924528	0.773585	0.89434
4	Normal	RF	BreastCancer	5	0.90566	0.976415	0.976415	0.90566	0.962264

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	6	0.962963	0.995261	0.990566	0.981132	0.988679
1	Her2	RF	BreastCancer	6	0.924528	0.981132	0.981132	0.924528	0.969811
2	LumA	RF	BreastCancer	6	0.679245	0.919811	0.919811	0.679245	0.871698
3	LumB	RF	BreastCancer	6	0.72549	0.925234	0.933962	0.698113	0.886792
4	Normal	RF	BreastCancer	6	0.925926	0.985782	0.981132	0.943396	0.973585

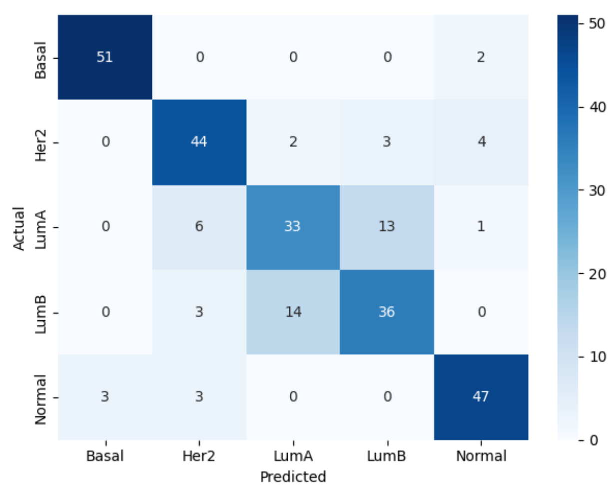
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	7	0.963636	1	0.990566	1	0.992453
1	Her2	RF	BreastCancer	7	0.961538	0.985915	0.990566	0.943396	0.981132
2	LumA	RF	BreastCancer	7	0.754717	0.938679	0.938679	0.754717	0.901887
3	LumB	RF	BreastCancer	7	0.769231	0.938967	0.943396	0.754717	0.90566
4	Normal	RF	BreastCancer	7	0.943396	0.985849	0.985849	0.943396	0.977358

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	8	0.981481	1	0.995283	1	0.996226
1	Her2	RF	BreastCancer	8	0.962264	0.990566	0.990566	0.962264	0.984906
2	LumA	RF	BreastCancer	8	0.784314	0.939252	0.948113	0.754717	0.909434
3	LumB	RF	BreastCancer	8	0.777778	0.947867	0.943396	0.792453	0.913208
4	Normal	RF	BreastCancer	8	0.962264	0.990566	0.990566	0.962264	0.984906

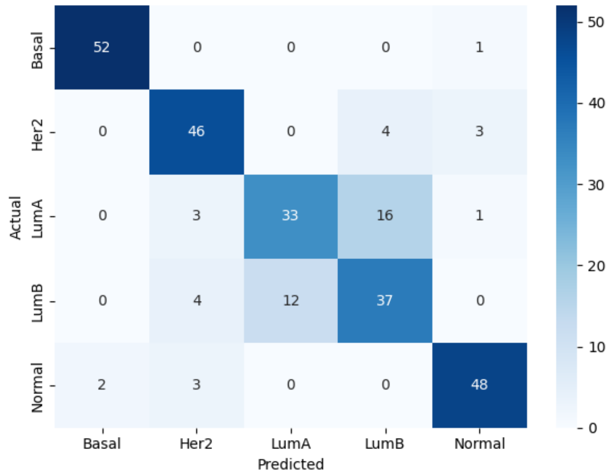
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	9	1	1	1	1	1
1	Her2	RF	BreastCancer	9	0.944444	0.990521	0.985849	0.962264	0.981132
2	LumA	RF	BreastCancer	9	0.78	0.934884	0.948113	0.735849	0.90566
3	LumB	RF	BreastCancer	9	0.759259	0.943128	0.938679	0.773585	0.90566
4	Normal	RF	BreastCancer	9	0.962963	0.995261	0.990566	0.981132	0.988679



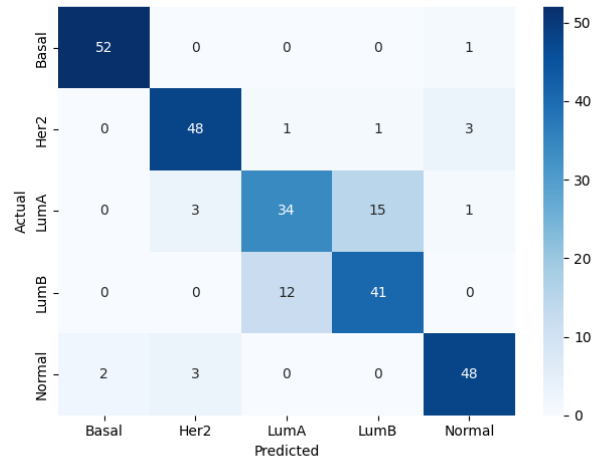
Number Of Features 2



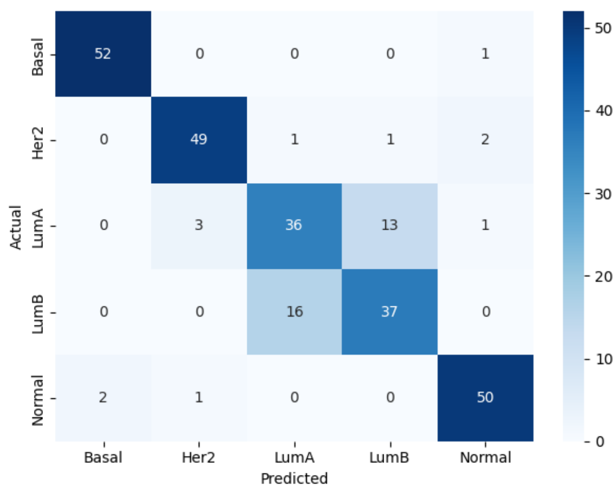
Number Of Features 3



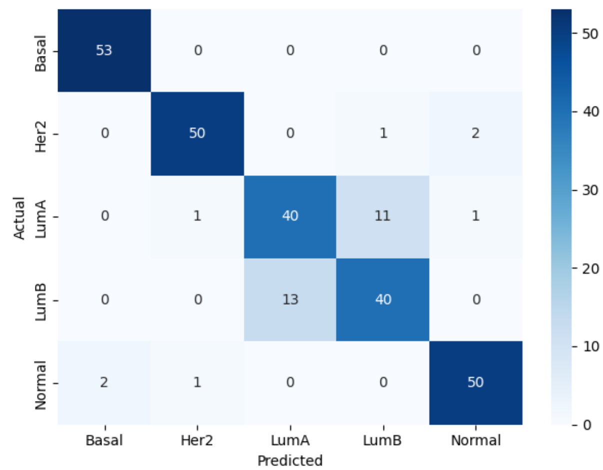
Number Of Features 4



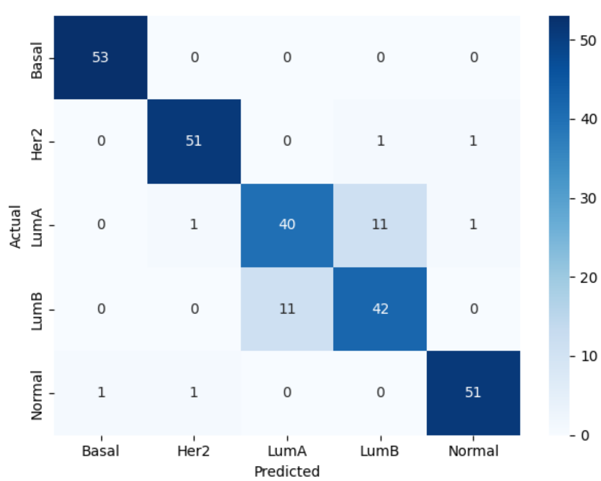
Number Of Features 5



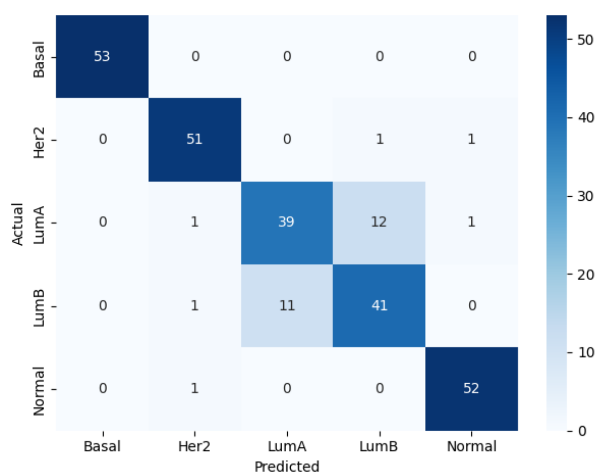
Number Of Features 6



Number Of Features 7



Number Of Features 8



Number Of Features 9

mrmr With RandomForest and OneVsOne scheme

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	2	0.961538	0.985915	0.990566	0.943396	0.981132
1	Her2	RF	BreastCancer	2	0.5625	0.915423	0.867925	0.679245	0.830189
2	LumA	RF	BreastCancer	2	0.659091	0.891403	0.929245	0.54717	0.85283
3	LumB	RF	BreastCancer	2	0.54	0.87907	0.891509	0.509434	0.815094
4	Normal	RF	BreastCancer	2	0.890909	0.980952	0.971698	0.924528	0.962264

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	3	0.962264	0.990566	0.990566	0.962264	0.984906
1	Her2	RF	BreastCancer	3	0.846154	0.957746	0.962264	0.830189	0.935849
2	LumA	RF	BreastCancer	3	0.622222	0.886364	0.919811	0.528302	0.841509
3	LumB	RF	BreastCancer	3	0.57377	0.911765	0.877358	0.660377	0.833962
4	Normal	RF	BreastCancer	3	0.888889	0.976303	0.971698	0.90566	0.958491

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	4	0.962264	0.990566	0.990566	0.962264	0.984906
1	Her2	RF	BreastCancer	4	0.872727	0.97619	0.966981	0.90566	0.954717
2	LumA	RF	BreastCancer	4	0.615385	0.901408	0.90566	0.603774	0.845283
3	LumB	RF	BreastCancer	4	0.686275	0.915888	0.924528	0.660377	0.871698
4	Normal	RF	BreastCancer	4	0.907407	0.981043	0.976415	0.924528	0.966038

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	5	1	1	1	1	1
1	Her2	RF	BreastCancer	5	0.907407	0.981043	0.976415	0.924528	0.966038
2	LumA	RF	BreastCancer	5	0.730769	0.929577	0.933962	0.716981	0.890566
3	LumB	RF	BreastCancer	5	0.716981	0.929245	0.929245	0.716981	0.886792
4	Normal	RF	BreastCancer	5	0.962264	0.990566	0.990566	0.962264	0.984906

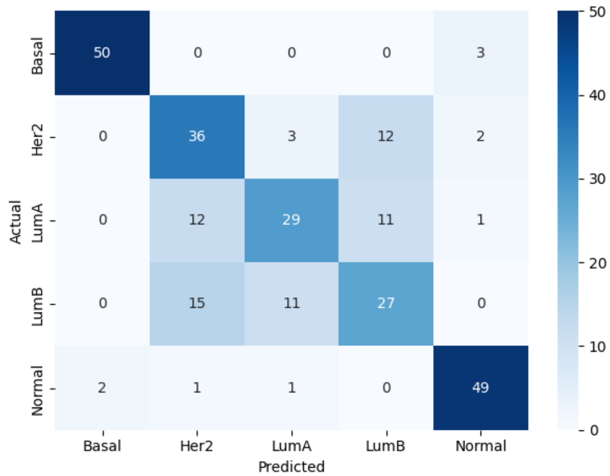
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	6	1	1	1	1	1
1	Her2	RF	BreastCancer	6	0.944444	0.990521	0.985849	0.962264	0.981132
2	LumA	RF	BreastCancer	6	0.759259	0.943128	0.938679	0.773585	0.90566
3	LumB	RF	BreastCancer	6	0.784314	0.939252	0.948113	0.754717	0.909434
4	Normal	RF	BreastCancer	6	0.981132	0.995283	0.995283	0.981132	0.992453

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	7	1	1	1	1	1
1	Her2	RF	BreastCancer	7	0.945455	0.995238	0.985849	0.981132	0.984906
2	LumA	RF	BreastCancer	7	0.792453	0.948113	0.948113	0.792453	0.916981
3	LumB	RF	BreastCancer	7	0.82	0.944186	0.957547	0.773585	0.920755
4	Normal	RF	BreastCancer	7	0.962963	0.995261	0.990566	0.981132	0.988679

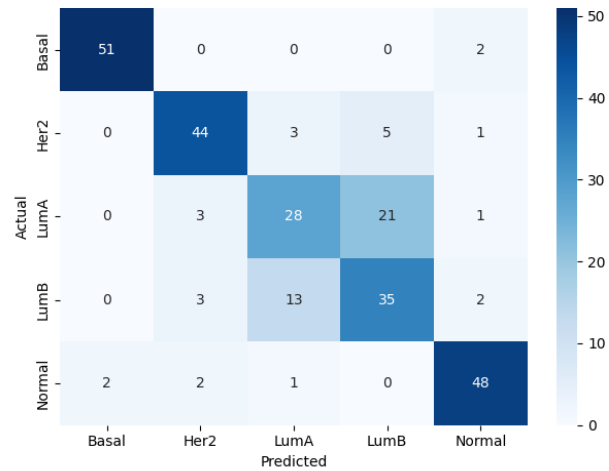
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	8	1	0.995305	1	0.981132	0.996226
1	Her2	RF	BreastCancer	8	0.944444	0.990521	0.985849	0.962264	0.981132
2	LumA	RF	BreastCancer	8	0.807692	0.948357	0.95283	0.792453	0.920755
3	LumB	RF	BreastCancer	8	0.826923	0.953052	0.957547	0.811321	0.928302
4	Normal	RF	BreastCancer	8	0.945455	0.995238	0.985849	0.981132	0.984906

	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	9	1	0.995305	1	0.981132	0.996226
1	Her2	RF	BreastCancer	9	0.962264	0.990566	0.990566	0.962264	0.984906
2	LumA	RF	BreastCancer	9	0.86	0.953488	0.966981	0.811321	0.935849
3	LumB	RF	BreastCancer	9	0.836364	0.966667	0.957547	0.867925	0.939623
4	Normal	RF	BreastCancer	9	0.945455	0.995238	0.985849	0.981132	0.984906

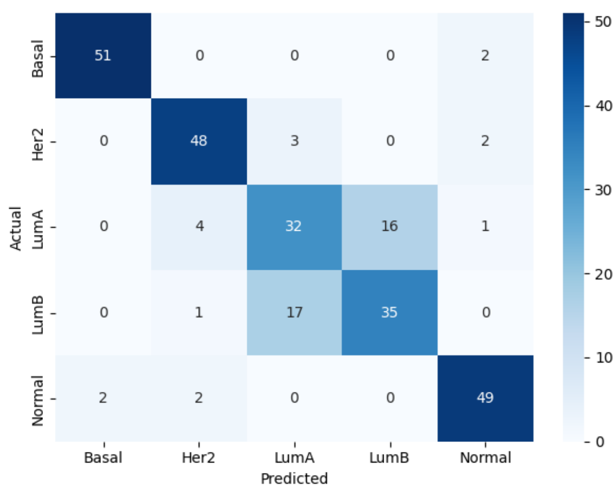
	Class	Classifier	Dataset	No. of Features	PPV	NPV	Specificity	Sensitivity	Accuracy
0	Basal	RF	BreastCancer	10	1	0.995305	1	0.981132	0.996226
1	Her2	RF	BreastCancer	10	0.980769	0.99061	0.995283	0.962264	0.988679
2	LumA	RF	BreastCancer	10	0.854167	0.9447	0.966981	0.773585	0.928302
3	LumB	RF	BreastCancer	10	0.807018	0.966346	0.948113	0.867925	0.932075
4	Normal	RF	BreastCancer	10	0.928571	0.995215	0.981132	0.981132	0.981132



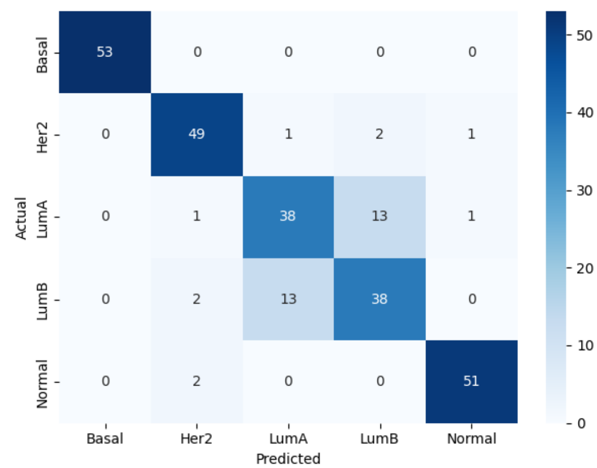
Number Of Features 2



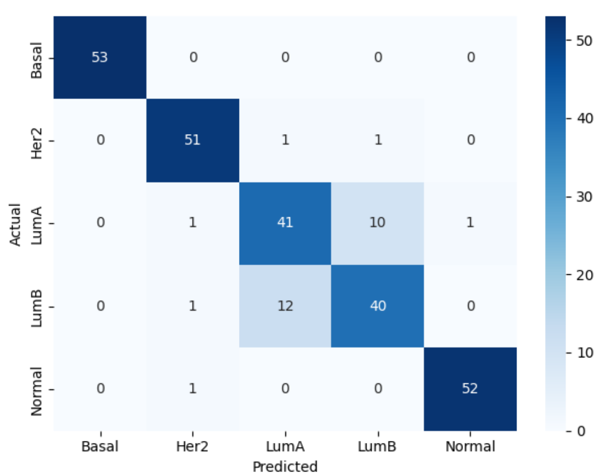
Number Of Features 3



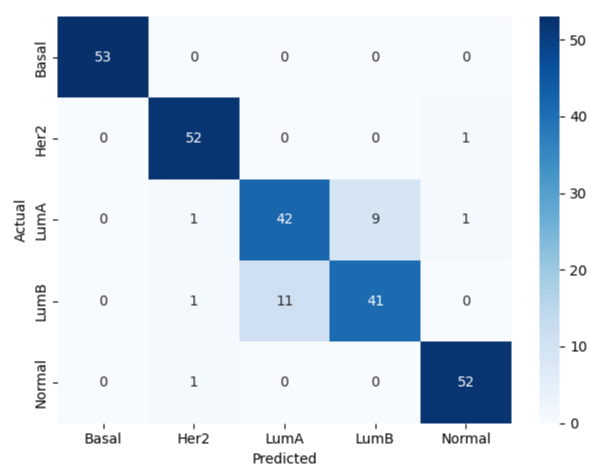
Number Of Features 4



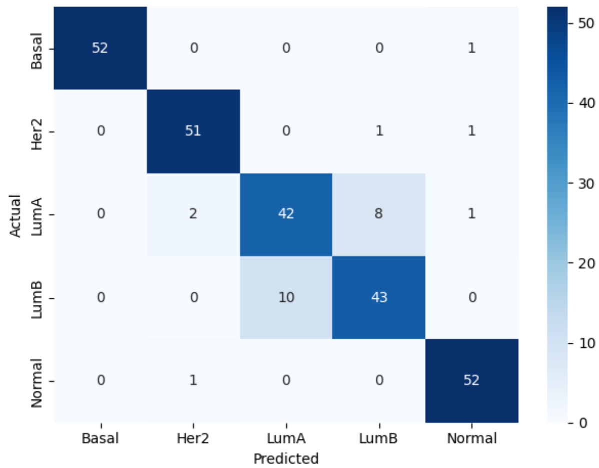
Number Of Features 5



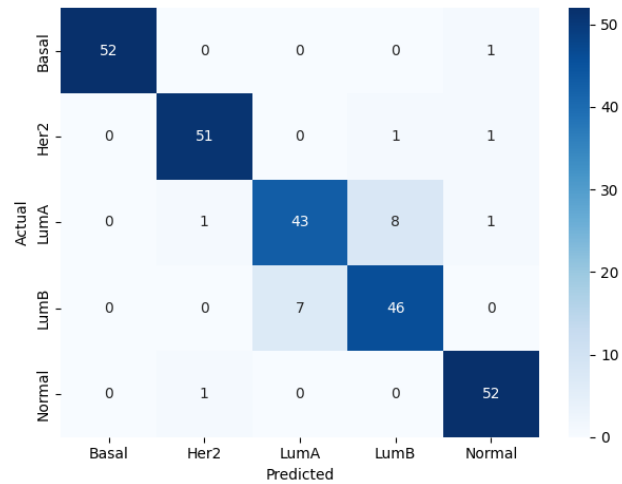
Number Of Features 6



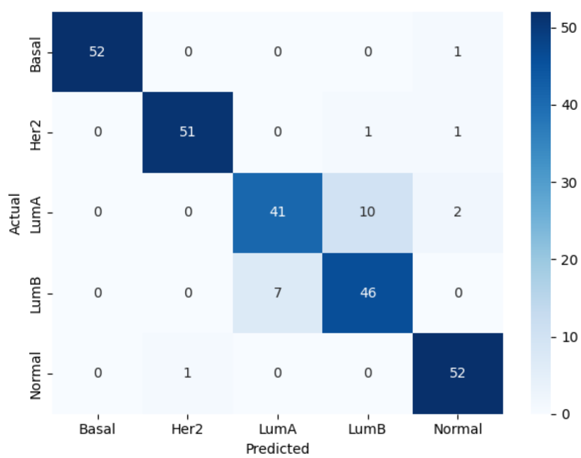
Number Of Features 7



Number Of Features 8



Number Of Features 9



Number Of Features 10