A Comprehensive Evaluation of Breast Cancer through Machine Learning Models

1st Akshat Taparia
Roll Number:210110014
Dept of Electrical Engineering, IITB
Mumbai, India.
210110014@iitb.ac.in
2nd Kishan Prajapati
Roll Number:21d070048
Dept of Electrical Engineering, IITB
Mumbai, India.
21d070048@iitb.ac.in
3rd Tanishka Pradhan
Roll Number:210040159
Dept of Electrical Engineering, IITB
Mumbai, India.
210040159@iitb.ac.in

Abstract: This study delves into the comparative analysis of several machine learning algorithms—Nearest Neighbor (NN) search, Support Vector Machine (SVM), and Random Forest—applied to the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. The primary focus is on evaluating their performance in terms of classification test accuracy, alongside sensitivity and specificity metrics. Through rigorous experimentation and analysis, this research aims to provide insights into the effectiveness of these algorithms in diagnosing breast cancer cases. By considering a diverse range of classifiers, this study contributes to a deeper understanding of their strengths and weaknesses, offering valuable guidance for the selection of optimal models in real-world medical applications.

Index Terms: Machine learning algorithms, Nearest Neighbor (NN) search, Support Vector Machine (SVM), Random Forest, Wisconsin Diagnostic Breast Cancer (WDBC) dataset, Classification, Test accuracy, Sensitivity, Specificity, Medical diagnosis.

1. Data Preprocessing

- It’s essential to highlight that the data utilized for analysis originates from data.csv.
- First the data is read from the csv file then the feature ”Unnamed: 32” has no use so we can drop it.
- The data has two type of breast cancer to diagnose one is M which is called Malignant and the other one is Benign. Benign tumors can cause serious problems if they grow near vital organs, press on a nerve, or restrict blood flow. Some common types of benign tumors include adenomas and fibroids. Malignant cells grow in an uncontrolled way and can invade nearby tissues and spread to other parts of the body through the blood and lymph system. Some symptoms of malignant tumors include: Breast pain and nipple discharge, Abdominal pain or changes in stool, Lesions or sores on the skin, and Shortness of breath. Moreover , Malignant tumors are cancerous and can spread to other tissues and organs. Benign tumors are non-cancerous and generally less harmful than malignant tumors.

- The column diagnosis is the target vector, so it can be either M or B.
- Then we divide the data obtained based on whether the target is M or B and describe them. It can be seen the tables describing the M and B type cancer characteristics are very different.
- Consider the radius mean feature on an average the mean radius for M type cancer diagnosis is greater than B type means people with large mean radius may have high chance of having the M type breast cancer which is cancerous in nature. Also the standard deviation is also high for M type, so it is not easy to say that a low radius mean size woman doesn't have M-type cancer. Also with higher value of radius mean in M type cells the perimeter mean is seen high in M type cells this is obvious , further we could see high correlation between this two features. Similar goes to area mean.
- Smoothness mean corresponds to variation in local radius length, as the standard deviation of radius mean in M type is high this obvious that smoothness mean of M type cell is greater than B type cell. Compactness mean in this data is calculated by the formula mean($(perimeter^2/area - 1.0)$). Compactness measures the similarity between the shape of a breast tumor and its fitting circle. The closer the compactness value is to 1, the less likely the tumor is to be malignant, expressed as in above formula. From the given reason as we can see compactness mean of B type is much lesser than M type cells.
- concavity - severity of concave portions of the contour
- concave points - number of concave portions of the contour.
- Symmetry are not very closely related to occurrence of breast cancer, there is usually seen asymmetric between breast or (breast cells) but that can be due to hormonal changes or there may be some symptoms of initial breast cancer.
- The fractal dimension of breast tissue samples is a numeric description of tumor growth patterns.
- Each of the characteristics are written after three measurements which includes mean , standard error and worst (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance radius mean means mean of radius of cell , radius se means standard error of radius and radius worst means the mean of the three largest radius.

2. Data Visualization

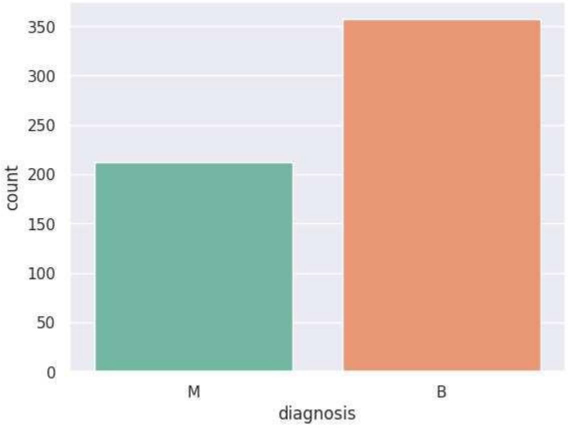

Fig. 1. Count of M and B cell types

- As seen the data has less people who are at risk of getting cancer that is M type breast cells. Also the data is not very unbalanced to classify.

- We have also shown the pair plots for all the features with itself, some feature which we discuss earlier like radius-mean and perimeter-mean or radius-se and perimeter-se etc has high correlation.
- There have been seen a high correlation between radius-mean and perimeter-mean and area-mean.Also between radius-worst , perimeter-worst and area-worst
- But radius-se , area-se and perimeter-se are area , hence are not properly correlated. The above observations are made using scatter plot.

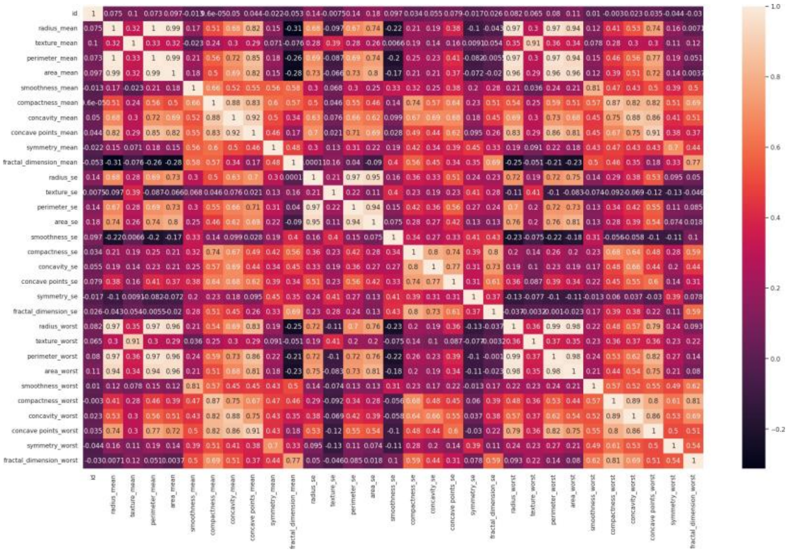

Fig. 2. Heat Map of correlation matrix

- We consider features with a correlation above 0.95 to be highly correlated, leading us to drop one of them.
  Correlated features: radius-mean and perimeter-mean with correlation 0.9978552814938109
  Correlated features: radius-mean and area-mean with correlation 0.9873571700566127
  Correlated features: radius-mean and radius-worst with correlation 0.9695389726112055
  Correlated features: radius-mean and perimeter-worst with correlation 0.9651365139559871
  Correlated features: perimeter-mean and area-mean with correlation 0.9865068039913907
  Correlated features: perimeter-mean and radius-worst with correlation 0.9694763634663146
  Correlated features: perimeter-mean and perimeter-worst with correlation 0.9703868870426394
  Correlated features: area-mean and radius-worst with correlation 0.9627460860470841
  Correlated features: area-mean and perimeter-worst with correlation 0.9591195743552645
  Correlated features: area-mean and area-worst with correlation 0.9592133256498998
  Correlated features: radius-se and perimeter-se with correlation 0.9727936770160764
  Correlated features: radius-se and area-se with correlation 0.951830112110991
  Correlated features: radius-worst and perimeter-worst with correlation 0.993707916102949
  Correlated features: radius-worst and area-worst with correlation 0.9840145644590742
  Correlated features: perimeter-worst and area-worst with correlation 0.9775780914063871
- So removing the perimeter-mean , area-mean , perimeter-worst , area-worst and perimeter-se. Here We are removing very highly correlated data only.
- Also it can be seen that features like radius-mean and radius-se are highly correlated but have different representation , for example radius-mean is the mean radius of cell from all direction and radius-se is the standard error in those directions.

3. ML model Training

- Now we will split the train , test data for our model training and normalised the train and test with unit variance and zero mean.

Now , let’s begin with simple ML model for classifications

3.1. Logistic Regression

- For this binary classification the decision boundary for our model will be a hyperplane in Higher dimension so we are checking if our data can be linearly classified or not.
- Hyperparameter grid:-
  ’penalty’: [’l1’, ’l2’ , ’elasticnet’], —– Regularization penalty
  ’C’: [0.001, 0.01, 0.1, 1, 10, 100], —– Inverse of regularization strength
  ’solver’: [’saga’], —– Solver algorithm for optimization
  ’l1-ratio’ : [0.01, 0.1, 0.4 , 0.6 , 0.8 , 1]
- best hyperparameters obtained after **max-iter=10000** were :-
  Best hyperparameters:- ’C’: 1, ’l1-ratio’: 0.4, ’penalty’: ’elasticnet’, ’solver’: ’saga’

3.1.1. Results and Classification report

- Classification Report:-

Class	Metrics			
	Precision	Recall	F1-Score	Support
B	0.99	0.99	0.99	72
M	0.98	0.98	0.98	42
Average	**0.98**	**0.98**	**0.98**	**114**
	Accuracy: 0.98			

TABLE I
CLASSIFICATION REPORT(LOGISTIC REGULARIZATION)

3.2. Support Vector Machine

- So support vector machine classifies the mode base on their location and not on basis of distribution of the data , we will implement the SVC support vector classifier using grid search CV and will see what results we are getting.
- Hyperparameter grid:-
  ’C’: [0.1, 1, 10, 100],—–Regularization parameter
  ’gamma’: [1, 0.1, 0.01, 0.001],—–Kernel coefficient for ’rbf’
  ’kernel’: [’rbf’, ’linear’] —–Kernel type
- best hyperparameters obtained were :-
  Best hyperparameters:- ’C’: 100, ’gamma’: 0.001, ’kernel’: ’rbf’

3.2.1. Results and Classification report

- Classification Report:-

Class	Metrics			
	Precision	Recall	F1-Score	Support
B	0.99	1.00	0.99	72
M	1.00	0.98	0.99	42
Average	**0.99**	**0.99**	**0.99**	**114**
	Accuracy: 0.99			

TABLE II
CLASSIFICATION REPORT (SVM)

3.3. *Desicion Tree*

- A decision tree ML model uses a tree-like structure to make predictions. It splits data into branches based on features, enabling straightforward interpretation and handling of both categorical and numerical data.
- Hyperparameter grid:-
  ’criterion’: [’gini’, ’entropy’],
  ’max-depth’: [None, 5, 10, 15],
  ’min-samples-split’: [2, 5, 10],
  ’min-samples-leaf’: [1, 2, 4]
- best hyperparameters obtained were :-
  Best hyperparameters:- ’criterion’: ’entropy’, ’max-depth’: 5, ’min-samples-leaf’: 2, ’min-samples-split’: 5

3.3.1. Results and Classification report

- Classification Report:-

Class	Metrics			
	Precision	Recall	F1-Score	Support
B	0.93	0.97	0.95	72
M	0.95	0.88	0.91	42
Macroavg	0.94	0.93	0.93	114
Weighted avg	0.94	0.94	0.94	114
	Accuracy: 0.94			

TABLE III
CLASSIFICATION REPORT (DESICION TREE)

3.3.2. Desicion Tree Visualization

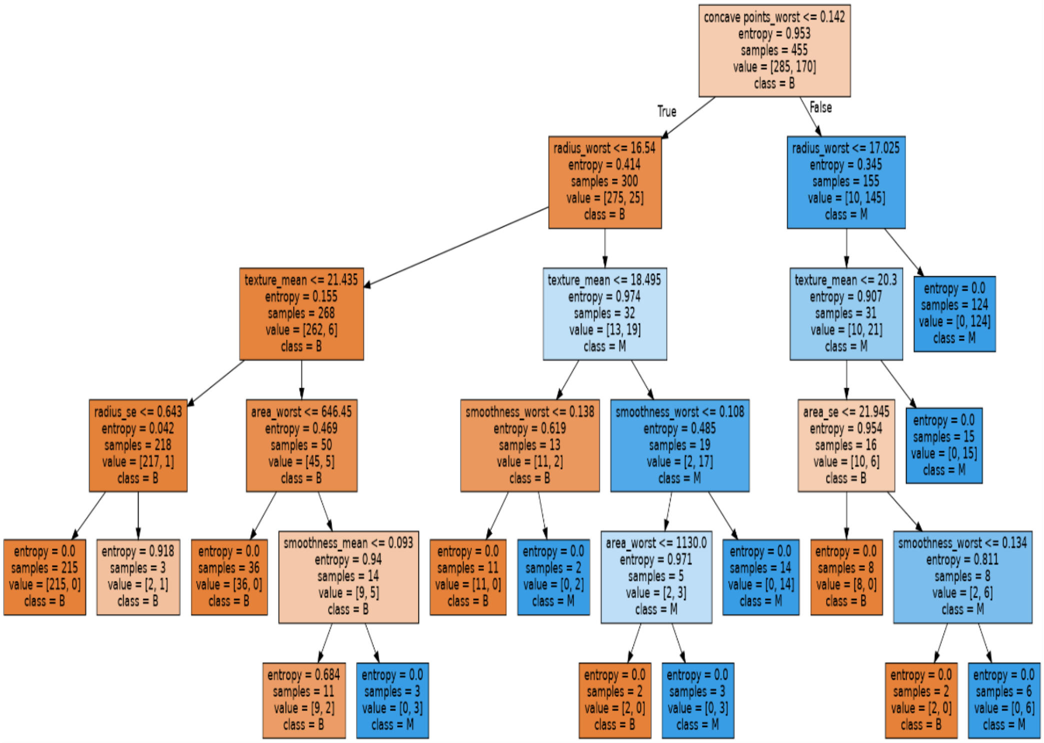
concave points_worst <= 0.142 entropy = 0.953 samples = 455 value = [285, 170] class = B
True
False
radius_worst <= 16.54 entropy = 0.414 samples = 300 value = [275, 25] class = B
radius_worst <= 17.025 entropy = 0.345 samples = 155 value = [10, 145] class = M
texture_mean <= 21.435 entropy = 0.155 samples = 268 value = [262, 6] class = B
texture_mean <= 18.495 entropy = 0.974 samples = 32 value = [13, 19] class = M
texture_mean <= 20.3 entropy = 0.907 samples = 31 value = [10, 21] class = M
entropy = 0.0 samples = 124 value = [0, 124] class = M
radius_se <= 0.643 entropy = 0.042 samples = 218 value = [217, 1] class = B
area_worst <= 646.45 entropy = 0.469 samples = 50 value = [45, 5] class = B
smoothness_worst <= 0.138 entropy = 0.619 samples = 13 value = [11, 2] class = B
smoothness_worst <= 0.108 entropy = 0.485 samples = 19 value = [2, 17] class = M
area_se <= 21.945 entropy = 0.954 samples = 16 value = [10, 6] class = B
entropy = 0.0 samples = 15 value = [0, 15] class = M
entropy = 0.0 samples = 215 value = [215, 0] class = B
entropy = 0.918 samples = 3 value = [2, 1] class = B
entropy = 0.0 samples = 36 value = [36, 0] class = B
smoothness_mean <= 0.093 entropy = 0.94 samples = 14 value = [9, 5] class = B
entropy = 0.0 samples = 11 value = [11, 0] class = B
entropy = 0.0 samples = 2 value = [0, 2] class = M
area_worst <= 1130.0 entropy = 0.971 samples = 5 value = [2, 3] class = M
entropy = 0.0 samples = 14 value = [0, 14] class = M
entropy = 0.0 samples = 8 value = [8, 0] class = B
smoothness_worst <= 0.134 entropy = 0.811 samples = 8 value = [2, 6] class = M
entropy = 0.684 samples = 11 value = [9, 2] class = B
entropy = 0.0 samples = 3 value = [0, 3] class = M
entropy = 0.0 samples = 2 value = [2, 0] class = B
entropy = 0.0 samples = 3 value = [0, 3] class = M
entropy = 0.0 samples = 2 value = [2, 0] class = B
entropy = 0.0 samples = 6 value = [0, 6] class = M

Fig. 3. Visualization of Desicion Tree

3.4. Random Forest

- Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes or mean prediction of the individual trees for regression tasks.
- Hyperparameter grid:-
  ’n-estimators’: [100, 200, 300],
  ’max-depth’: [None, 10, 20],
  ’min-samples-split’: [2, 5, 10],
  ’min-samples-leaf’: [1, 2, 4],
  ’max-features’: [’auto’, ’sqrt’]
- best hyperparameters obtained were :-
  Best hyperparameters:- ’max-depth’: 10, ’max-features’: ’sqrt’, ’min-samples-leaf’: 1, ’min-samples-split’: 5, ’n-estimators’: 200

3.4.1. Results and Classification report

- Classification Report:-

Class	Metrics			
	Precision	Recall	F1-Score	Support
B	0.97	0.99	0.98	72
M	0.98	0.95	0.96	42
Macroavg	0.97	0.97	0.97	114
Weighted avg	0.97	0.97	0.97	114
	Accuracy: 0.97			

TABLE IV
CLASSIFICATION REPORT (RANDOM FOREST)

3.5. Adaboost

- We iteratively combine weak classifiers to build a strong classifier.
- Hyperparameter grid:-
  ’n-estimators’: [50, 100, 200],
  ’learning-rate’: [0.1, 0.5, 1.0]
- best hyperparameters obtained were :-
  Best hyperparameters:- ’learning-rate’: 0.5, ’n-estimators’: 100

3.5.1. Results and Classification report

- Classification Report:-

Class	Metrics			
	Precision	Recall	F1-Score	Support
B	0.99	0.99	0.99	72
M	0.98	0.98	0.98	42
Macroavg	0.98	0.98	0.98	114
Weighted avg	0.98	0.98	0.98	114
	Accuracy: 0.98			

TABLE V
CLASSIFICATION REPORT (ADABOOST)

3.6. Ensemble

- Now we will combine above model and use majority voting for our prediction.

3.6.1. Majority Vote Predictions

```
[’B’ ’B’ ’B’ ’M’ ’B’ ’B’ ’B’ ’M’ ’B’ ’B’ ’M’ ’B’ ’B’ ’B’ ’M’ ’B’ ’B’ ’B’ ’M’ ’M’ ’B’ ’B’ ’B’ ’B’ ’M’ ’B’ ’M’ ’B’ ’M’
’M’ ’B’ ’M’ ’B’ ’M’ ’B’ ’B’ ’M’ ’M’ ’M’ ’M’ ’M’ ’B’ ’B’ ’B’ ’B’ ’B’ ’M’ ’B’ ’M’ ’B’ ’M’ ’B’ ’B’ ’M’ ’B’ ’B’ ’M’
’M’ ’B’ ’B’ ’M’ ’M’ ’B’ ’B’ ’M’ ’B’ ’B’ ’M’ ’M’ ’B’ ’M’ ’B’ ’B’ ’B’ ’M’ ’M’ ’B’ ’M’ ’M’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’
’M’ ’B’ ’M’ ’M’ ’B’ ’M’ ’M’ ’B’ ’B’ ’B’ ’B’ ’B’ ’M’ ’M’ ’M’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’B’ ’M’]
```

3.6.2. Results and Classification report

- Classification Report:-

Class	Metrics			
	Precision	Recall	F1-Score	Support
B	1.00	1.00	1.00	72
M	1.00	1.00	1.00	42
Macroavg	1.00	1.00	1.00	114
Weighted avg	1.00	1.00	1.00	114
	Accuracy: 1.00			

TABLE VI
CLASSIFICATION REPORT (ENSEMBLE)

We did great in individual model , further with ensemble different models are performing amazingly when combined.

4. Feature Importance for Different Models

4.1. Logistic Regression Feature Importance

4.1.1. Sorted Coefficients

Feature	**Coefficient**
radius_se	1.8666
concave points_mean	1.2787
radius_worst	1.2773
texture_worst	1.1625
concave points_worst	1.1441
area_se	1.1311
area_worst	1.0948
concavity_worst	0.7966
compactness_se	-0.7399
concavity_mean	0.6771
smoothness_worst	0.6761
fractal_dimension_se	-0.5947
fractal_dimension_worst	0.5353
symmetry_worst	0.5256
fractal_dimension_mean	-0.4680
radius_mean	0.4163
texture_mean	0.3190
compactness_mean	-0.3184
texture_se	-0.1337
concavity_se	-0.1293
symmetry_se	-0.0182
smoothness_se	-0.0178
concave points_se	0.0091
smoothness_mean	0.0000
symmetry_mean	0.0000
compactness_worst	0.0000

TABLE VII
SORTED COEFFICIENTS FOR LOGISTIC REGRESSION MODEL.

4.1.2. Interpretation

The coefficient of ’radius_se’ being the highest and the lowest for ’smoothness_mean: 0.0’, ’symmetry_mean: 0.0’, ’compactness_worst: 0.0’ suggests that the feature associated with the higher coefficient has a stronger influence on the predicted outcome compared to the feature associated with the lower coefficient.

4.2. Support Vector Classifier Feature Importance

4.2.1. Sorted Coefficients

Feature	Importance
symmetry_worst	0.9825
radius_mean	0.9912
texture_mean	0.9912
smoothness_mean	0.9912
compactness_mean	0.9912
concavity_mean	0.9912
concave points_mean	0.9912
symmetry_mean	0.9912
fractal_dimension_mean	0.9912
radius_se	0.9912
texture_se	0.9912
area_se	0.9912
smoothness_se	0.9912
compactness_se	0.9912
concavity_se	0.9912
concave points_se	0.9912
symmetry_se	0.9912
fractal_dimension_se	0.9912
radius_worst	0.9912
texture_worst	0.9912
area_worst	0.9912
smoothness_worst	0.9912
compactness_worst	0.9912
concavity_worst	0.9912
concave points_worst	0.9912
fractal_dimension_worst	0.9912

TABLE VIII
FEATURE IMPORTANCE SORTED BY IMPACT ON ACCURACY.

4.2.2. Interpretation

- Weights asigned to the features (coefficients in the primal problem). This is only available in the case of linear kernel. but also it doesn’t make sense. In linear SVM the resulting separating plane is in the same space as your input features. Therefore its coefficients can be viewed as weights of the input’s ”dimensions”.
- In other kernels, the separating plane exists in another space - a result of kernel transformation of the original space. Its coefficients are not directly related to the input space. In fact, for the rbf kernel the transformed space is infinite-dimensional.
- So , we can try removing one feature one by one and and can see in absence of which feature the model is performing best
- Obviously this is kernelized SVC so removing one feature has no huge change cause in accuracy , but we can say by looking the chage in accuracy that removing the symmetryworst cause the highest change in accuracy and removing the fractaldimensionworst cause the least change , which suggest us that from top to bottom the feature importance decreases for SVC.

4.3. Decision Tree Feature Importance

4.3.1. Sorted Coefficients

Feature	Importance
concave points_worst	0.6048
radius_worst	0.1815
texture_mean	0.0839
smoothness_worst	0.0447
area_worst	0.0357
area_se	0.0207
radius_se	0.0152
smoothness_mean	0.0133
radius_mean	0.0
compactness_mean	0.0
concavity_mean	0.0
concave points_mean	0.0
symmetry_mean	0.0
fractal_dimension_mean	0.0
texture_se	0.0
smoothness_se	0.0
compactness_se	0.0
concavity_se	0.0
concave points_se	0.0
symmetry_se	0.0
fractal_dimension_se	0.0
texture_worst	0.0
compactness_worst	0.0
concavity_worst	0.0
symmetry_worst	0.0
fractal_dimension_worst	0.0

TABLE IX
FEATURE IMPORTANCE IN DECISION TREE MODEL

4.3.2. Interpretation

- The major impact of the feature we got to see is 'concave pointsworst' and the importance decreases from top to bottom , Also some feature are zero coefficients in the above list which suggest us that those feature are pruned during training of Desicion tree to reduce overfitting.

4.4. Random Forest Feature Importance

4.4.1. Sorted Coefficients

Feature	Importance
concave points_worst	0.1693
concave points_mean	0.1405
area_worst	0.1305
radius_worst	0.1123
radius_mean	0.0790
concavity_mean	0.0647
concavity_worst	0.0597
area_se	0.0567
compactness_worst	0.0232
radius_se	0.0232
texture_worst	0.0194
texture_mean	0.0191
compactness_mean	0.0151
symmetry_worst	0.0123
smoothness_worst	0.0114
concavity_se	0.0114
fractal_dimension_se	0.0078
concave points_se	0.0068
fractal_dimension_mean	0.0054
symmetry_se	0.0054
smoothness_mean	0.0053
texture_se	0.0051
fractal_dimension_worst	0.0049
compactness_se	0.0044
smoothness_se	0.0041
symmetry_mean	0.0030

TABLE X
FEATURE IMPORTANCE FOR RANDOM FOREST

4.4.2. Interpretation

- In the provided list of feature importances, ”concave points_worst” has the highest importance value of 0.1823, while ”symmetry_mean” has the lowest importance value of 0.0026.

4.5. *Adaptive Boosting Feature Importance*

4.5.1. Sorted Coefficients

Feature	Importance
area_worst	0.11
area_se	0.1
compactness_se	0.08
texture_worst	0.07
smoothness_worst	0.07
texture_mean	0.06
radius_se	0.06
concavity_worst	0.06
compactness_mean	0.05
concave points_mean	0.05
smoothness_mean	0.04
fractal_dimension_worst	0.04
concavity_mean	0.03
symmetry_mean	0.03
concave points_worst	0.03
radius_mean	0.02
texture_se	0.02
symmetry_se	0.02
smoothness_se	0.01
concave points_se	0.01
fractal_dimension_se	0.01
radius_worst	0.01
compactness_worst	0.01
symmetry_worst	0.01
fractal_dimension_mean	0.0
concavity_se	0.0

TABLE XI
ADAPTIVE BOOSTING FEATURE IMPORTANCE

4.5.2. Interpretation

- This is the feature importance for adaptive boosting in which areaworst being the most importance and concavityse being the least.

4.6. *Summary of Feature Importance*

So, different models assign varying degrees of importance to features, which can be advantageous when combining them through ensemble methods like majority voting. which I have done , Here's how it works:

- Diverse Feature Importance: Various machine learning algorithms, such as decision trees, support vector machines and Random forests assess feature importance differently. Decision trees might give high importance to features with high predictive power at certain splits, while SVMs might emphasize features that best separate classes. Consequently, each model captures distinct aspects of the data's complexity.
- Ensemble Diversity: When combining models using ensemble methods using majority voting, these divergent views on feature importance can go hand in hand with each other. Models that excel at recognizing different patterns and capturing various aspects of the data can compensate for each other's weaknesses.
- Enhanced Stability: Ensemble methods also enhance the stability of predictions by reducing variance. Since individual models may perform inconsistently due to fluctuations in the training data or randomly initializing something, combining their predictions smoothens out these fluctuations, resulting in more stable and reliable predictions.