Classification Techniques for Breast Cancer and Heart Disease

Jaladurgam Navya
Kent State Univeristy
jnavya@kent.edu

***Abstract*— This study evaluates the performance of various machine learning models for diagnosing breast cancer using the Wisconsin Breast Cancer Dataset and predicting heart disease using a heart disease dataset. Selected five models are Logistic Regression, the K-Nearest Neighbors, Naive Bayes, the Decision Tree, and the Random Forest Techniques. Results of this research the Logistic Regression and the Random Forest classifiers had the maximum accuracy of all at 97.37%. In the heart disease analysis, the four models are same as used for breast cancer dataset except support vector classifier. This is multi class classification achieved accuracy from 69%-95%. The results demonstrate the importance of selecting suitable machine learning models in healthcare for early diagnosis and treatment, particularly in breast cancer and heart disease, showing the potential for improved clinical decision-making.**

This study evaluates the model's performance by analyzing accuracy, precision, recall, specificity, constructing confusion metrics .

Keywords—machine learning, models, scaling, confusion metrics, precision

I. Introduction

Breast cancer is the worldwide considered to be one of the most predominant forms of cancer particularly it is one of the deadliest cancer tissues that exist, and its early detection drugs are paramount in terms of aiding the patient in recovery. Recently, due to the advancements in technology, some statistics prove that the accuracy and speed by which breast cancers are being detected and classified can be improved through diagnostic Machine Learning which has a promising future in the medical field. Heart disease is similarly one of the leading causes of death across the globe and it is important for its earliest possible detection to be made for an effective treatment and control to be received.

The breast cancer dataset contains 569 instances and 30 features, this is binary classification. For the heart disease dataset has 303 instances and 13 features which is the processed Cleveland dataset, and this is multi class classification .By using these datasets evaluated and compared the five different algorithms. The steps performed include data preprocessing, which involves feature selection, applying Principal Component Analysis, followed by data visualization. The dataset was split into training and testing sets with 80% training dataset and 20 % testset for two datasets.

II. Backgroung

A. Logistic Regression

When it comes to classification problems with two classes, logistic regression model is the most that predicts the probability of a binary outcome. It is based on the logistic function mapping input features to predicted probabilities. Logistic Regression model is used in both datasets achieved a maximum accuracy of **97.37%** in the breast cancer analysis.

B. K- Nearest neighbours

K-Nearest Neighbors works by examining the data points that are nearest to a new instance and identifying the most prevalent category among them. Such a procedure is very much useful in cases of medical modeling due to its ability to tidy up most high dimensional data that might have different diseases or conditions.

C. Naive Bayes

Is a classification technique based on Bayes' Theorem with an assumption of independence amongst the predictors. This assumption is quite strong and is the reason why it is termed 'naive.' However, despite its simplicity, it tends to perform very well particularly in high-dimensional datasets.

D. Decision Tree

They are classification models which are understandable and transparent and that divide the feature space in locked manner based on the values of the features. Such a structure is tree-like where each node is assigned a feature on who, each branch is a decision rule speaking for itself and every terminal node provides an outcome.

E. Random forest

Random Forest is a form of machine-learning which uses prominent decision trees in order to push up the prediction accuracy. It is widely recognized that decision trees are prone to overfitting which makes Random Forest very useful as it combines many trees.

F. Support Vector Classifier (SVC)

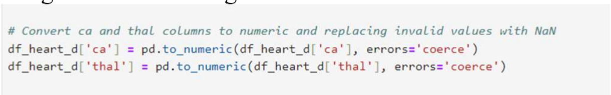
The idea is to find a separating hyperplane with the greatest possible margin of all the data regardless of its classes. Consequently, the concept is effective in mapping data not just in space but in high-dimensional space when the data becomes non-linearly separable.

Jupyter notebook: Jupyter notebook is an open-source web application that allows you to create, visualizations, data cleaning, data transformation. Utilized this notebook for the entire project.

III. Data preprocessing

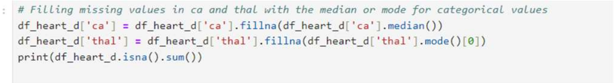
A. *Checking missing values for two datasets*

For the breast cancer dataset and heart disease dataset there are no missing values. But for the heart disease dataset has “?” symbol for ‘ca’ and ‘thal’ columns. Converted ca and thal columns to numeric and replaces in valid values with NaN shown in Fig 1. Then filling the missing values in a ca by using median and using mode for thal feature.



```
# Convert ca and thal columns to numeric and replacing invalid values with NaN
df_heart_d['ca'] = pd.to_numeric(df_heart_d['ca'], errors='coerce')
df_heart_d['thal'] = pd.to_numeric(df_heart_d['thal'], errors='coerce')
```

Fig 1: Converted ca and thal columns

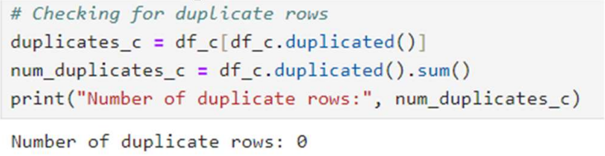


```
# Filling missing values in ca and thal with the median or mode for categorical values
df_heart_d['ca'] = df_heart_d['ca'].fillna(df_heart_d['ca'].median())
df_heart_d['thal'] = df_heart_d['thal'].fillna(df_heart_d['thal'].mode()[0])
print(df_heart_d.isna().sum())
```

Fig 2: Filling missing values with ode and median

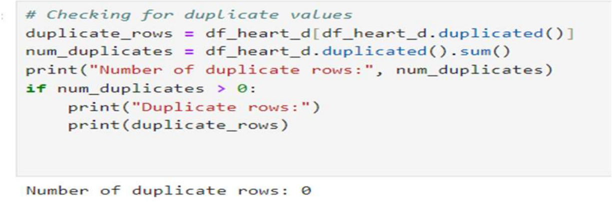
B. *Chekcing Duplicate values*

There are no duplicate values for these two datasets.



```
# Checking for duplicate rows
duplicates_c = df_c[df_c.duplicated()]
num_duplicates_c = df_c.duplicated().sum()
print("Number of duplicate rows:", num_duplicates_c)

Number of duplicate rows: 0
```

Fig 3:Breast cancer has no duplicates rows



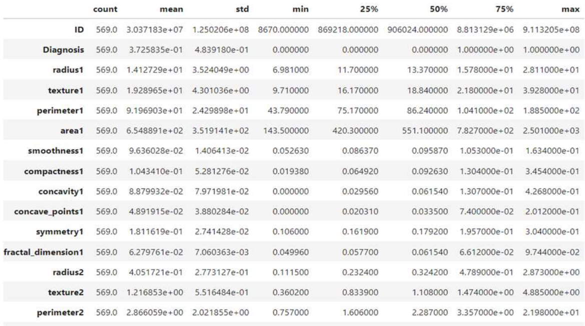
```
# Checking for duplicate values
duplicate_rows = df_heart_d[df_heart_d.duplicated()]
num_duplicates = df_heart_d.duplicated().sum()
print("Number of duplicate rows:", num_duplicates)
if num_duplicates > 0:
    print("Duplicate rows:")
    print(duplicate_rows)

Number of duplicate rows: 0
```

Fig 4:Heart disease has no duplicates rows

C. *Summary statistics for the breast cancer dataframe*

Diagnosis: This is the target variable, likely indicating whether a condition is present (1) or absent (0). The mean of 0.3725835 suggests about 37.26% of cases are positive. As a binary variable, its min is 0 and max is 1.

Radius: There are two radius measurements (radius1 and radius2). Radius1 has a mean of 14.12729, ranging from about 6.98 to 28.11. This could represent the size of cells or structures being analyzed.

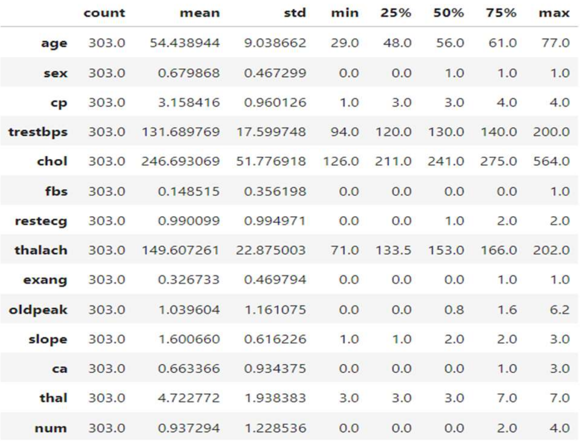


	count	mean	std	min	25%	50%	75%	max
ID	569.0	3.037183e+07	1.250206e+08	8670.000000	869218.000000	906024.000000	8.813129e+06	9.113205e+08
Diagnosis	569.0	3.725835e-01	4.839180e-01	0.000000	0.000000	0.000000	1.000000e+00	1.000000e+00
radius1	569.0	1.412729e+01	3.524049e+00	6.981000	11.700000	13.370000	1.578000e+01	2.811000e+01
texture1	569.0	1.928965e+01	4.301036e+00	9.710000	16.170000	18.840000	2.180000e+01	3.928000e+01
perimeter1	569.0	9.196903e+01	2.429898e+01	43.790000	75.170000	86.240000	1.041000e+02	1.885000e+02
area1	569.0	6.548891e+02	3.519141e+02	143.500000	420.300000	551.100000	7.827000e+02	2.501000e+03
smoothness1	569.0	9.636028e-02	1.406413e-02	0.052630	0.086370	0.095870	1.053000e-01	1.634000e-01
compactness1	569.0	1.043410e-01	5.281276e-02	0.019380	0.064920	0.092630	1.304000e-01	3.454000e-01
concavity1	569.0	8.879932e-02	7.971981e-02	0.000000	0.029560	0.061540	1.307000e-01	4.268000e-01
concave_points1	569.0	4.891915e-02	3.880284e-02	0.000000	0.020310	0.033500	7.400000e-02	2.012000e-01
symmetry1	569.0	1.811619e-01	2.741428e-02	0.106000	0.161900	0.179200	1.957000e-01	3.040000e-01
fractal_dimension1	569.0	6.279761e-02	7.060363e-03	0.049960	0.057700	0.061540	6.612000e-02	9.744000e-02
radius2	569.0	4.051721e-01	2.773127e-01	0.111500	0.232400	0.324200	4.789000e-01	2.873000e+00
texture2	569.0	1.216853e+00	5.516484e-01	0.360200	0.833900	1.108000	1.474000e+00	4.885000e+00
perimeter2	569.0	2.866059e+00	2.021855e+00	0.757000	1.606000	2.287000	3.357000e+00	2.198000e+01

Fig:5 description of breast cancer dataset.

Summary statistics for the heart disease dataframe

Mean age is about 54.4 years, ranging from 29 to 77 years. Sex feature is binary variable with mean 0.679868, likely coded as 0 for one gender and 1 for the other. The mean suggests about 68% of subjects are coded as 1. "Chest pain" type, ranging from 1 to 4 with a mean of 3.158416.’num’ is the target variable it has 0-4 multiclass values. 25th percentile: 0 ,50th percentile (median): 0 ,75th percentile: 2



	count	mean	std	min	25%	50%	75%	max
age	303.0	54.438944	9.038662	29.0	48.0	56.0	61.0	77.0
sex	303.0	0.679868	0.467299	0.0	0.0	1.0	1.0	1.0
cp	303.0	3.158416	0.960126	1.0	3.0	3.0	4.0	4.0
trestbps	303.0	131.689769	17.599748	94.0	120.0	130.0	140.0	200.0
chol	303.0	246.693069	51.776918	126.0	211.0	241.0	275.0	564.0
fbs	303.0	0.148515	0.356198	0.0	0.0	0.0	0.0	1.0
restecg	303.0	0.990099	0.994971	0.0	0.0	1.0	2.0	2.0
thalach	303.0	149.607261	22.875003	71.0	133.5	153.0	166.0	202.0
exang	303.0	0.326733	0.469794	0.0	0.0	0.0	1.0	1.0
oldpeak	303.0	1.039604	1.161075	0.0	0.0	0.8	1.6	6.2
slope	303.0	1.600660	0.616226	1.0	1.0	2.0	2.0	3.0
ca	303.0	0.663366	0.934375	0.0	0.0	0.0	1.0	3.0
thal	303.0	4.722772	1.938383	3.0	3.0	3.0	7.0	7.0
num	303.0	0.937294	1.228536	0.0	0.0	0.0	2.0	4.0

Fig 6: Description of heart disease dataset.

IV. Data visualization

Data visualization for breast cancer dataset

A. *Distribution of Diagnosis variable*

Green bar (0 - Benign): Represents the count of benign cases, around 350. This shows that there are more benign cases in the dataset.

Red bar (1 - Malignant): Represents the count of malignant cases, around 200. There are fewer malignant cases in comparison to benign ones.

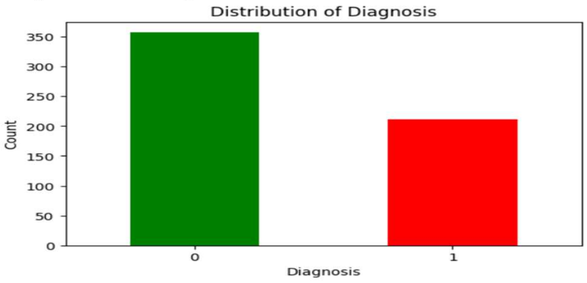


Fig 7: Distribution of diagnosis

B. *Histograms of features*

Histograms give an overview of how features are distributed and may help in understanding the nature of the data and its spread. Most of the distributions exhibit the trend of right-skewness with different amount of data variation within the features with a few variables having more evenly spread distributions. The "3"(radius3,perimeter3,…) variables typically show a broader range of values than the "2"(radius2, perimeter2,…) variables, indicating more variation or larger feature values.

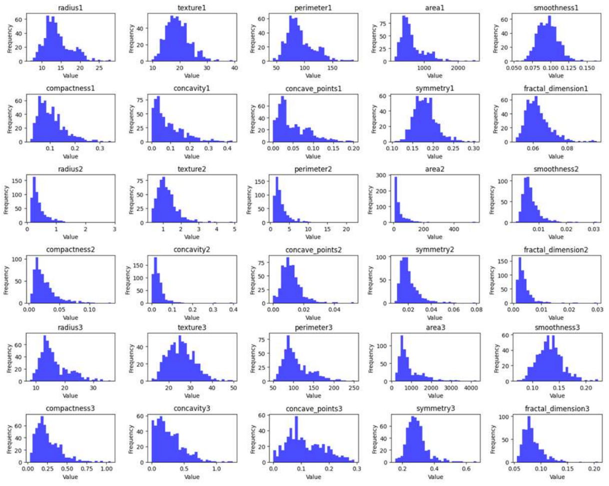


Fig :9 Histogram of features

C.Comparative Distribution of Tumor Features Based on Diagnosis

This kind of methodology facilitates the determination of the most suitable parameters, to differentiate malignant and benign cases within the problem of breast cancer classification.

For example, it is observed that radius1, perimeter1 and area1 are intricate in distinguishing benign and malignant tumors, with malignant tumors tending to have higher values for these features. Texture1 and smoothness1 have more overlap between the two groups, suggesting these features may be less distinctive for classifying tumors as benign or malignant. Radius3, perimeter3, and area3 show malignant cases (1) tend to have larger values than benign cases (0). This suggests malignant structures are generally larger.

compactness1,concave_points1,perimeter2,area2,Radius3, perimeter3, and area3 show malignant cases (1) tend to have larger values than benign cases (0). This suggests malignant structures are generally larger.

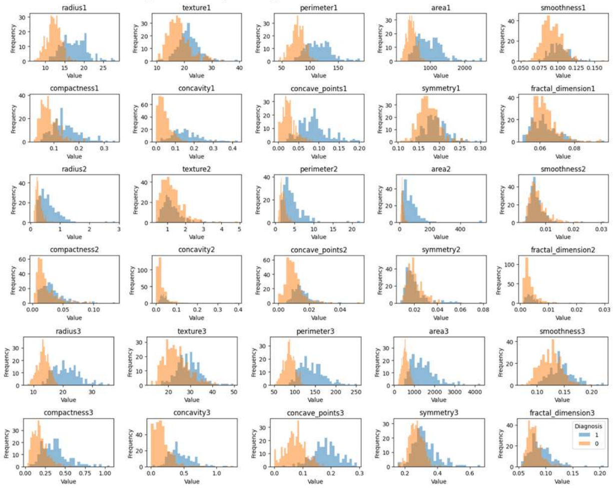


Fig : 10 Comparative Distribution of Tumor Features Based on Diagnosis (Benign vs Malignant)

Overall, features like radius, perimeter, area, and concavity show strong differences between benign and malignant tumors, making them potentially valuable for distinguishing between the two. Smoothness, symmetry, and fractal dimension show more overlap between the benign and malignant cases, indicating these features may be less predictive on their own.

C. Correlation heat map

Below figure shows that there is an extremely high correlation between radius and perimeter in both the "1", "2", and "3" feature sets, with correlation coefficients close to 1. radius1 and area1 at 0.9874 and similarly for radius3 and area3 at 0.984. The features related to the size of the objects radius, perimeter, area show very strong linear correlations with each other. It is logical to observe that the correlations suggest that these features are not mutual but are closely bounded with each other most likely due to their shape properties.

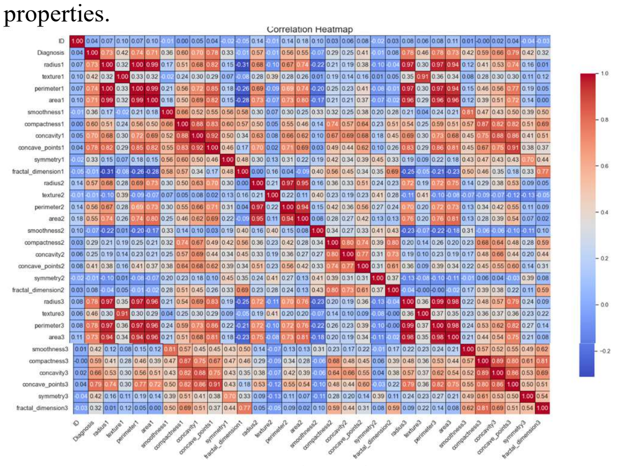
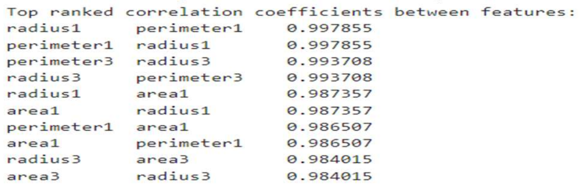


Fig 11: Correlation heat map



```
Top ranked correlation coefficients between features:
radius1     perimeter1     0.997855
perimeter1  radius1        0.997855
perimeter3  radius3        0.993708
radius3     perimeter3     0.993708
radius1     area1          0.987357
area1       radius1        0.987357
perimeter1  area1          0.986507
area1       perimeter1     0.986507
radius3     area3          0.984015
area3       radius3        0.984015
```

These are the top co-related features

Data visualization for heart disease dataset

A. Distibution of num(target) variable

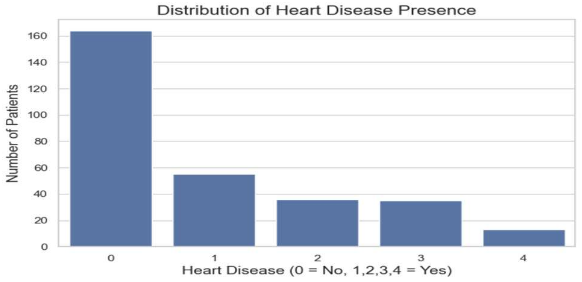


Fig 12: Distribution of heart disease

The largest group consists of patients without heart disease, with over 160 patients in this category. The heart disease 1 to 4 . The number of patients decreases progressively as the severity of heart disease increases. Category 1 has around 50 patients, followed by 2 and 3 with approximately 35–40 patients each. Category 4 is fewer than 20 patients.

B. Histograms of numeric values

The distributions suggest that most individuals in the dataset are middle-aged (40-60), with average resting blood pressure (120-140 mmHg) and cholesterol (200-250 mg/dL) values. The maximum heart rate achieved by most individuals falls between 150 and 170 bpm. All the variables show some level of right skewness, with a larger concentration of values toward the lower end of the ranges. Most values for oldpeak are concentrated around 0, with a sharp decline in frequency as oldpeak values increase. The variable ca represents the number of major vessels (ranging from 0 to 3) visible under fluoroscopy. The most frequent value is 0, with over 150 occurrences, meaning that the majority of individuals have no vessels colored by fluoroscopy. As the number of vessels increases (1, 2, and 3), the frequency decreases.

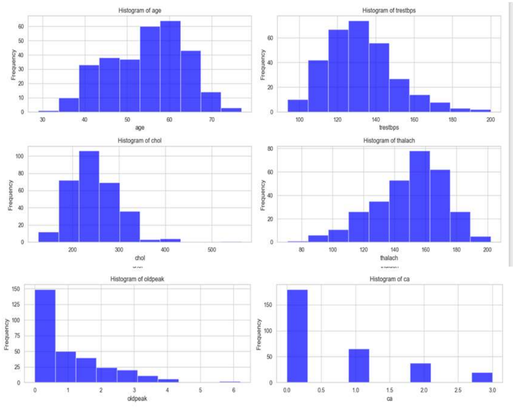


Fig 13: Histograms of numeric values

C. Distribution of categorical variables with the target

Target 0 consistently shows the highest counts across all metrics, suggesting it may represent a "normal" or low-risk group. Exang is less common overall, especially in target 0.the middle category of slope (2.0) is most prevalent across all targets. Normal thal results (3.0) are most common, particularly in target 0, while abnormal results are more evenly distributed across other targets. There's a clear data imbalance, with target 0 having significantly more cases than other groups.

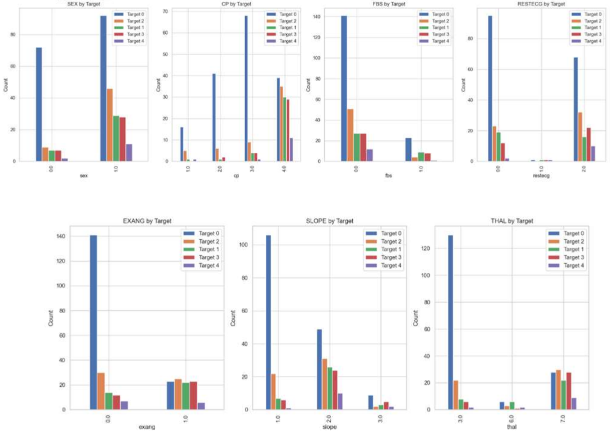
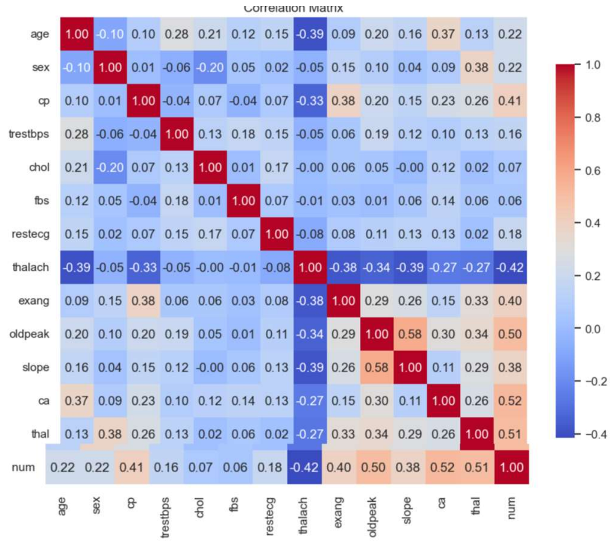


Fig: 14 Distribution for categorical with target variables

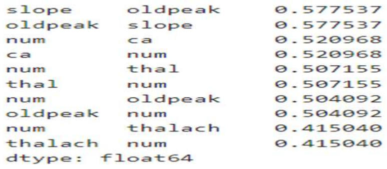
C. Heatmap

In the heart disease dataset, the strongest correlations are between "slope" and "oldpeak" (0.577537), "num" and "ca" (0.520968), and "num" and "thal" (0.507155). Additionally, "num" correlates with "oldpeak" at 0.504092 and with "thalach" at 0.415040. These values highlight important variable relationships for predicting heart disease.



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	num
age	1.00	-0.10	0.10	0.28	0.21	0.12	0.15	-0.39	0.09	0.20	0.16	0.37	0.13	0.22
sex	-0.10	1.00	0.01	-0.06	-0.20	0.05	0.02	-0.05	0.15	0.10	0.04	0.09	0.38	0.22
cp	0.10	0.01	1.00	-0.04	0.07	-0.04	0.07	-0.33	0.38	0.20	0.15	0.23	0.26	0.41
trestbps	0.28	-0.06	-0.04	1.00	0.13	0.18	0.15	-0.05	0.06	0.19	0.12	0.10	0.13	0.16
chol	0.21	-0.20	0.07	0.13	1.00	0.01	0.17	-0.00	0.06	0.05	-0.00	0.12	0.02	0.07
fbs	0.12	0.05	-0.04	0.18	0.01	1.00	0.07	-0.01	0.03	0.01	0.06	0.14	0.06	0.06
restecg	0.15	0.02	0.07	0.15	0.17	0.07	1.00	-0.08	0.08	0.11	0.13	0.13	0.02	0.18
thalach	-0.39	-0.05	-0.33	-0.05	-0.00	-0.01	-0.08	1.00	-0.38	-0.34	-0.39	-0.27	-0.27	-0.42
exang	0.09	0.15	0.38	0.06	0.06	0.03	0.08	-0.38	1.00	0.29	0.26	0.15	0.33	0.40
oldpeak	0.20	0.10	0.20	0.19	0.05	0.01	0.11	-0.34	0.29	1.00	0.58	0.30	0.34	0.50
slope	0.16	0.04	0.15	0.12	-0.00	0.06	0.13	-0.39	0.26	0.58	1.00	0.11	0.29	0.38
ca	0.37	0.09	0.23	0.10	0.12	0.14	0.13	-0.27	0.15	0.30	0.11	1.00	0.26	0.52
thal	0.13	0.38	0.26	0.13	0.02	0.06	0.02	-0.27	0.33	0.34	0.29	0.26	1.00	0.51
num	0.22	0.22	0.41	0.16	0.07	0.06	0.18	-0.42	0.40	0.50	0.38	0.52	0.51	1.00

Fig: 15 Heatmap

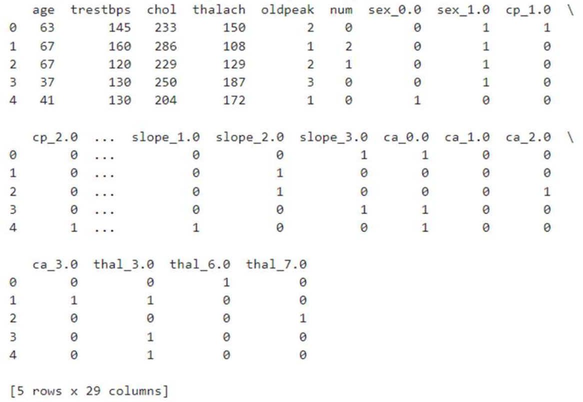


```
slope     oldpeak     0.577537
oldpeak   slope       0.577537
num       ca          0.520968
ca        num         0.520968
num       thal        0.507155
thal      num         0.507155
num       oldpeak     0.504092
oldpeak   num         0.504092
num       thalach     0.415040
thalach   num         0.415040
dtype: float64
```

These are top co-related features

D. One-Hot encoding on heart disease dataset

There are categorical features in this dataset. So used one-hot encoding method to convert these categorical variables. sex', 'cp', 'fbs', 'restecg', 'exang', 'slope', 'ca', 'thal.



```
   age  trestbps  chol  thalach  oldpeak  num  sex_0.0  sex_1.0  cp_1.0  \
0   63       145   233      150        2    0        0        1       1
1   67       160   286      108        1    2        0        1       0
2   67       120   229      129        2    1        0        1       0
3   37       130   250      187        3    0        0        1       0
4   41       130   204      172        1    0        1        0       0

   cp_2.0  ...  slope_1.0  slope_2.0  slope_3.0  ca_0.0  ca_1.0  ca_2.0  \
0       0  ...          0          0          1       1       0       0
1       0  ...          0          1          0       0       0       0
2       0  ...          0          1          0       0       0       1
3       0  ...          0          0          1       1       0       0
4       1  ...          1          0          0       1       0       0

   ca_3.0  thal_3.0  thal_6.0  thal_7.0
0       0         0         1         0
1       1         1         0         0
2       0         0         0         1
3       0         1         0         0
4       0         1         0         0

[5 rows x 29 columns]
```

Fig 16: One -hot encoding method

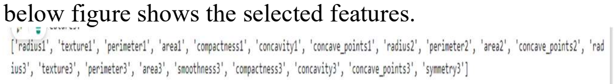
E. Splitting the datasets

For these two datasets splitting the training dataset as 80% and testing as 20%. After splitting performed feature scaling, feature selection and dimension reduction.

V. Feature selection

A. Breast Cancer

Used SelectKBest method for feature selection[1] .
f_classif: ANOVA F-value method used to select the best features. Number of features selected is k=20.Selecting the top 20 features strikes a balance between reducing dimensionality and retaining important information. In the below figure shows the selected features.

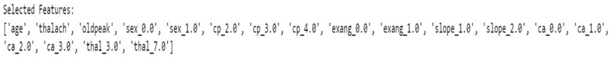


```
['radius1', 'texture1', 'perimeter1', 'area1', 'compactness1', 'concavity1', 'concave_points1', 'radius2', 'perimeter2', 'area2', 'concave_points2', 'rad
ius3', 'texture3', 'perimeter3', 'area3', 'smoothness3', 'compactness3', 'concavity3', 'concave_points3', 'symmetry3']
```

Fig :17 Features selected

B. Heart Disease

Used SelectKBest method for feature selection same as breast cancer dataset[1]. Number of features selected is k=18.Selecting the top 18 features strikes a balance between reducing dimensionality and retaining important information. In the below figure shows the selected features. If I choose 20 features the accuracy is reducing when compared to 18 features.



```
Selected Features:
['age', 'thalach', 'oldpeak', 'sex_0.0', 'sex_1.0', 'cp_2.0', 'cp_3.0', 'cp_4.0', 'exang_0.0', 'exang_1.0', 'slope_1.0', 'slope_2.0', 'ca_0.0', 'ca_1.0',
'ca_2.0', 'ca_3.0', 'thal_3.0', 'thal_7.0']
```

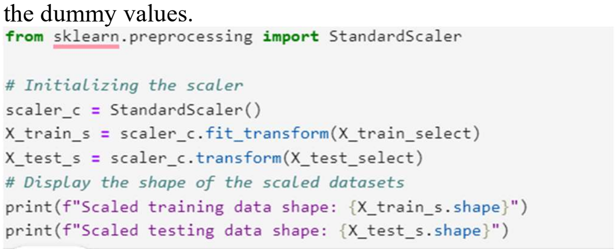
Fig : 18 features selected for heart disease

VI. Scaling

Breast cancer and heart disease

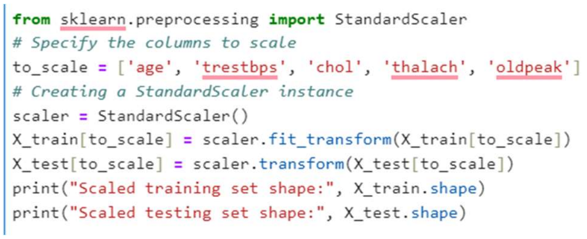
Applied StandardScaler method, the scaler was fit to the training data using the fit_transform method. The same transformation was applied to the test data using the transform method to ensure consistency in scaling, as the test data was scaled based on the training data statistics.

In the heart disease prediction task, StandardScaler was applied before feature selection. Specifically, only the following features were scaled: 'age', 'trestbps', 'chol', 'thalach', 'oldpeak' because for remaining variables created the dummy values.



```python
from sklearn.preprocessing import StandardScaler

# Initializing the scaler
scaler_c = StandardScaler()
X_train_s = scaler_c.fit_transform(X_train_select)
X_test_s = scaler_c.transform(X_test_select)
# Display the shape of the scaled datasets
print(f"Scaled training data shape: {X_train_s.shape}")
print(f"Scaled testing data shape: {X_test_s.shape}")
```

Fig : 19 Scaling for breast cancer



```python
from sklearn.preprocessing import StandardScaler
# Specify the columns to scale
to_scale = ['age', 'trestbps', 'chol', 'thalach', 'oldpeak']
# Creating a StandardScaler instance
scaler = StandardScaler()
X_train[to_scale] = scaler.fit_transform(X_train[to_scale])
X_test[to_scale] = scaler.transform(X_test[to_scale])
print("Scaled training set shape:", X_train.shape)
print("Scaled testing set shape:", X_test.shape)
```

Fig :20 Scaling for the heart disease

VII. Pricipal component analysis

PCA for breast cancer and heart disease

PCA is then applied in both the test and training sets to reduce the dimensionality for the breast cancer and heart disease datasets to six components. If PCA components is selected as 4 for the breast cancer dataset the accuracy is decreasing to 96%.The code for the scatter plot is designed within the function to picturize the first 3 principal components for visualization in the training data.

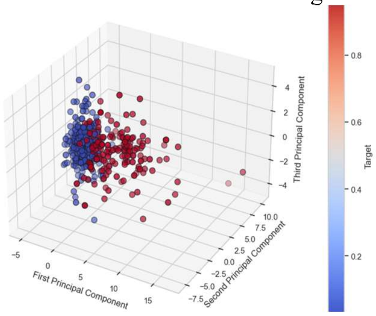


Fig 21 Breast Cancer PCA

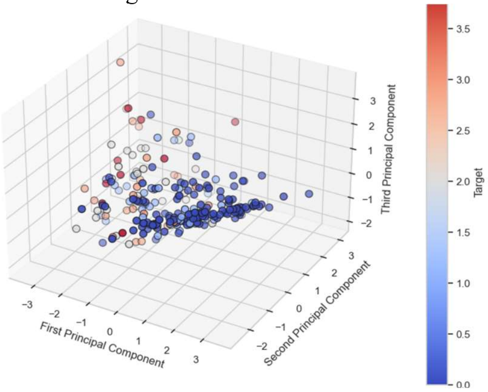
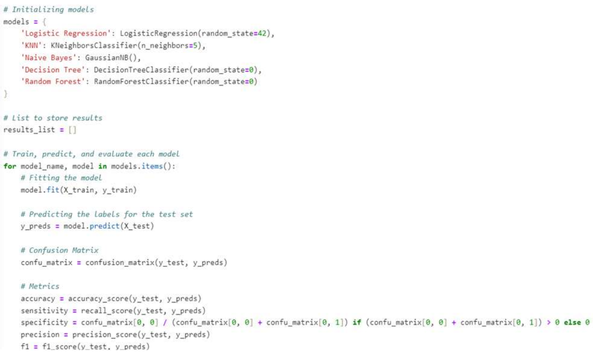


Fig 22: Heart disease PCA

VIII. Modeling

A. Breast Cancer Modeling

In the analysis of breast cancer , five classification models were utilized, including Logistic Regression (with a random state of 42), K-Nearest Neighbors (KNN) with 5 neighbors, Naive Bayes, Decision Tree (random state of 0), and Random Forest (also with a random state of 0).

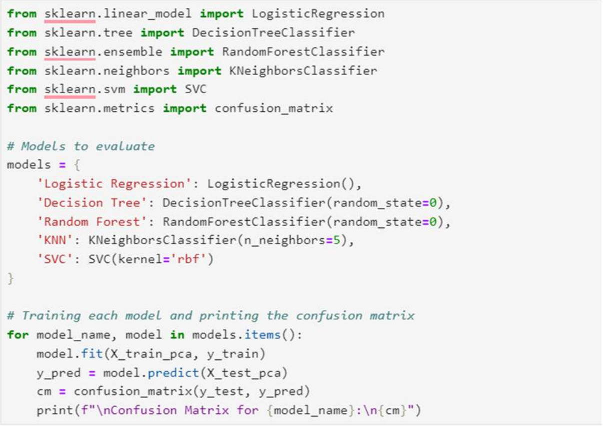


```
# Initializing models
models = {
    'Logistic Regression': LogisticRegression(random_state=42),
    'KNN': KNeighborsClassifier(n_neighbors=5),
    'Naive Bayes': GaussianNB(),
    'Decision Tree': DecisionTreeClassifier(random_state=0),
    'Random Forest': RandomForestClassifier(random_state=0)
}

# List to store results
results_list = []

# Train, predict, and evaluate each model
for model_name, model in models.items():
    # Fitting the model
    model.fit(X_train, y_train)

    # Predicting the labels for the test set
    y_preds = model.predict(X_test)

    # Confusion Matrix
    confu_matrix = confusion_matrix(y_test, y_preds)

    # Metrics
    accuracy = accuracy_score(y_test, y_preds)
    sensitivity = recall_score(y_test, y_preds)
    specificity = confu_matrix[0, 0] / (confu_matrix[0, 0] + confu_matrix[0, 1]) if (confu_matrix[0, 0] + confu_matrix[0, 1]) > 0 else 0
    precision = precision_score(y_test, y_preds)
    f1 = f1_score(y_test, y_preds)
```

Fig 23: Breast cancer modeling

B. Heart Disease Modeling

In the analysis of heart disease , five classification models were utilized, including Logistic Regression (with a random state of 42), K-Nearest Neighbors with 5 neighbors, Support vector classification(kernel='rbf'), Decision Tree (random state of 0) the default value for the criterion is indeed 'gini', and Random Forest (also with a random state of 0).



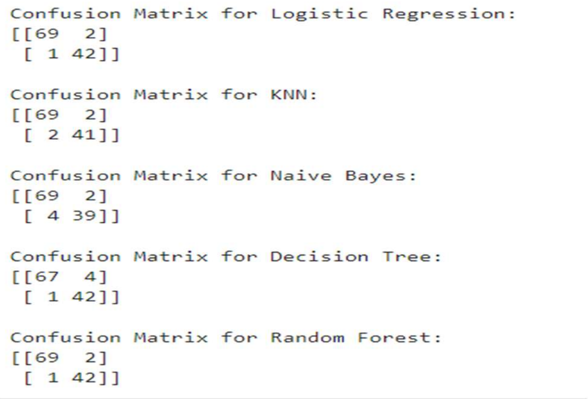
```
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix

# Models to evaluate
models = {
    'Logistic Regression': LogisticRegression(),
    'Decision Tree': DecisionTreeClassifier(random_state=0),
    'Random Forest': RandomForestClassifier(random_state=0),
    'KNN': KNeighborsClassifier(n_neighbors=5),
    'SVC': SVC(kernel='rbf')
}

# Training each model and printing the confusion matrix
for model_name, model in models.items():
    model.fit(X_train_pca, y_train)
    y_pred = model.predict(X_test_pca)
    cm = confusion_matrix(y_test, y_pred)
    print(f"\nConfusion Matrix for {model_name}:\n{cm}")
```

Fig 24: Heart disease modeling

IX. CONFUSION MATRIX

A. Breast Cancer

Logistic Regression and Random Forest behave almost similarly, with high true positives, they also have low false negatives(69, 42). KNN and Naive Bayes also exhibit a bit more false negatives and false positives. In the case of Decision Tree, though more- it has a greater number of false negatives but keeps false positives very low.

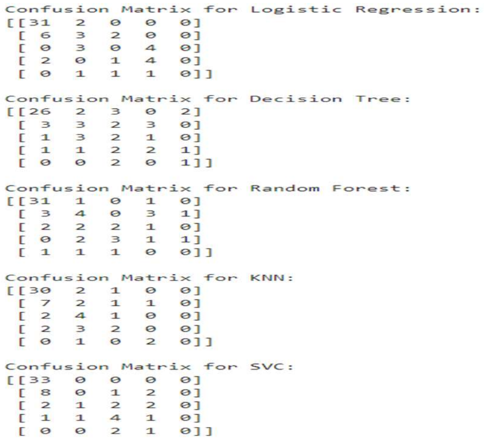


```
Confusion Matrix for Logistic Regression:
[[69  2]
 [ 1 42]]

Confusion Matrix for KNN:
[[69  2]
 [ 2 41]]

Confusion Matrix for Naive Bayes:
[[69  2]
 [ 4 39]]

Confusion Matrix for Decision Tree:
[[67  4]
 [ 1 42]]

Confusion Matrix for Random Forest:
[[69  2]
 [ 1 42]]
```

Fig 25: Confusion matrix for breast cancer

B. Heart disease

Heart disease target variable has five different classes. confusion matrix is shown below Class 1 seems to be labeled well across all models however there may be vast confusion some of the different classes, in the middle rows class 2, class 3. SVC and Random Forest perform best for class 1 however have challenges with the remaining classes .

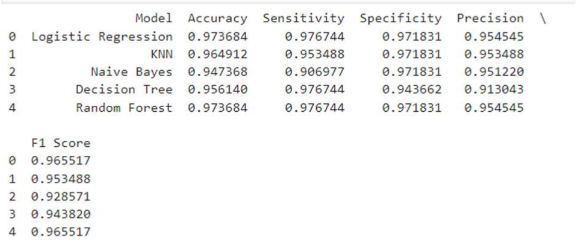


```
Confusion Matrix for Logistic Regression:
[[31  2  0  0  0]
 [ 6  3  2  0  0]
 [ 0  3  0  4  0]
 [ 2  0  1  4  0]
 [ 0  1  1  1  0]]

Confusion Matrix for Decision Tree:
[[26  2  3  0  2]
 [ 3  3  2  3  0]
 [ 1  3  2  1  0]
 [ 1  1  2  2  1]
 [ 0  0  2  0  1]]

Confusion Matrix for Random Forest:
[[31  1  0  1  0]
 [ 3  4  0  3  1]
 [ 2  2  2  1  0]
 [ 0  2  3  1  1]
 [ 1  1  1  0  0]]

Confusion Matrix for KNN:
[[30  2  1  0  0]
 [ 7  2  1  1  0]
 [ 2  4  1  0  0]
 [ 2  3  2  0  0]
 [ 0  1  0  2  0]]

Confusion Matrix for SVC:
[[33  0  0  0  0]
 [ 8  0  1  2  0]
 [ 2  1  2  2  0]
 [ 1  1  4  1  0]
 [ 0  0  2  1  0]]
```

Fig 26: Confusion Matrix for Heart disease

X. EVALUATION

A. Comparison table for breast cancer



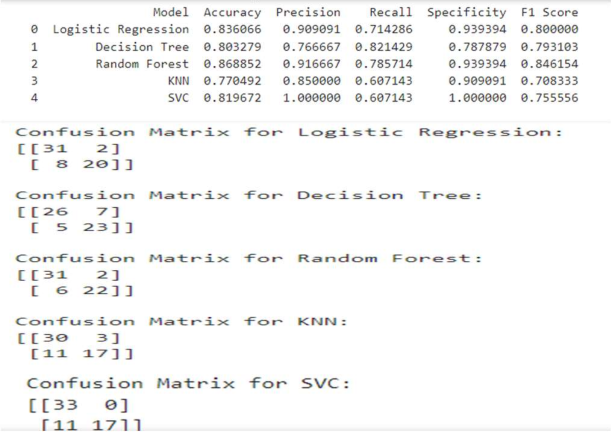
```
                 Model  Accuracy  Sensitivity  Specificity  Precision  \
0  Logistic Regression  0.973684     0.976744     0.971831   0.954545
1                  KNN  0.964912     0.953488     0.971831   0.953488
2          Naive Bayes  0.947368     0.906977     0.971831   0.951220
3        Decision Tree  0.956140     0.976744     0.943662   0.913043
4        Random Forest  0.973684     0.976744     0.971831   0.954545

   F1 Score
0  0.965517
1  0.953488
2  0.928571
3  0.943820
4  0.965517
```

Fig 27: Evaluation for breast cancer

Considering accuracy, logistic regression and random Forest remain the top performers with accuracy 97%, excelling in all measured aspects. Decision Tree with 95% and Naive Bayes with 94% have lower accuracy scores. All models show strong performance, with F1 scores ranging from 92.86% to 96.55%. Logistic Regression, Random Forest, and Decision Tree are same at 97.67%.

B. Comparison table for heart disease

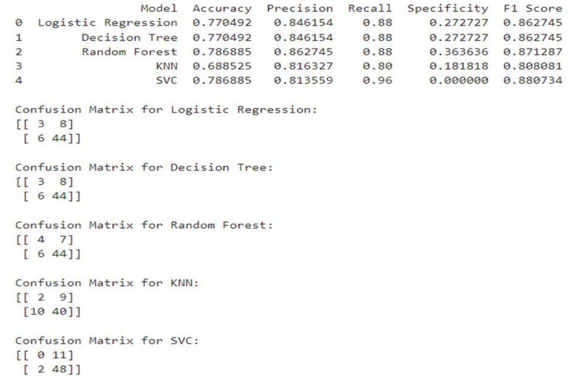
Fig 28: Treating class 0 as the negative class treat all other classes (1, 2, 3, 4) as the positive class .



```
                 Model  Accuracy  Precision    Recall  Specificity  F1 Score
0  Logistic Regression  0.836066   0.909091  0.714286     0.939394  0.800000
1        Decision Tree  0.803279   0.766667  0.821429     0.787879  0.793103
2        Random Forest  0.868852   0.916667  0.785714     0.939394  0.846154
3                  KNN  0.770492   0.850000  0.607143     0.909091  0.708333
4                  SVC  0.819672   1.000000  0.607143     1.000000  0.755556

Confusion Matrix for Logistic Regression:
[[31  2]
 [ 8 20]]

Confusion Matrix for Decision Tree:
[[26  7]
 [ 5 23]]

Confusion Matrix for Random Forest:
[[31  2]
 [ 6 22]]

Confusion Matrix for KNN:
[[30  3]
 [11 17]]

Confusion Matrix for SVC:
[[33  0]
 [11 17]]
```

Random forest with the high accuracy of 86.89% ,precision 91.67% and recall 78.57%. SVC has 100% precision and specificity but lower recall 60.71% and accuracy 81.97% . KNN and Decision Tree show lower accuracy (77.05% and 80.33%), with KNN having more false negatives, while Decision Tree balances recall (82.14%) but has lower specificity. The F1 Score, which balances precision and recall, varies across the models, with Random Forest achieving the highest F1 score of 0.846

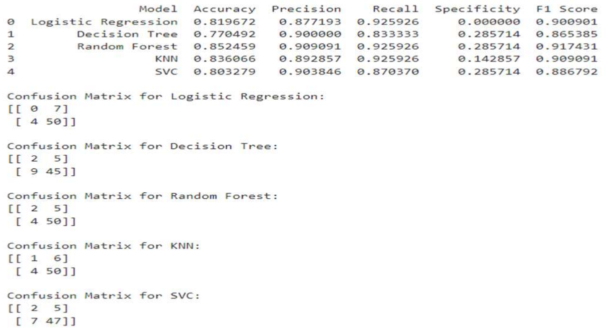
Fig :29 Treating class 1 as the negative class .Treating all other classes (0, 2, 3, 4) as the positive class



```
                 Model  Accuracy  Precision  Recall  Specificity  F1 Score
0  Logistic Regression  0.770492   0.846154    0.88     0.272727  0.862745
1        Decision Tree  0.770492   0.846154    0.88     0.272727  0.862745
2        Random Forest  0.786885   0.862745    0.88     0.363636  0.871287
3                  KNN  0.688525   0.816327    0.80     0.181818  0.808081
4                  SVC  0.786885   0.813559    0.96     0.000000  0.880734

Confusion Matrix for Logistic Regression:
[[ 3  8]
 [ 6 44]]

Confusion Matrix for Decision Tree:
[[ 3  8]
 [ 6 44]]

Confusion Matrix for Random Forest:
[[ 4  7]
 [ 6 44]]

Confusion Matrix for KNN:
[[ 2  9]
 [10 40]]

Confusion Matrix for SVC:
[[ 0 11]
 [ 2 48]]
```

Random Forest and SVC reached 78.69% any accuracy. The highest accuracy is for the two methods; specifically, Decision Tree and Logistic Regression respectively followed closely with 77.05% each. KNN algorithm had the lowest accuracy at 68.85%. On the other hand, the Random Forest seems to be the most efficient from the point of accuracy and precision, sensitivity, and specificity among all the models.

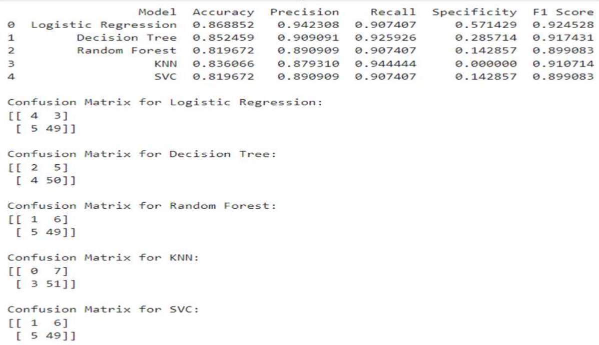
Fig : 30 Treating class 2 as the negative class .Treating all other classes (0, 1, 3, 4) as the positive class.



```
                 Model  Accuracy  Precision    Recall  Specificity  F1 Score
0  Logistic Regression  0.819672   0.877193  0.925926     0.000000  0.900901
1        Decision Tree  0.770492   0.900000  0.833333     0.285714  0.865385
2        Random Forest  0.852459   0.909091  0.925926     0.285714  0.917431
3                  KNN  0.836066   0.892857  0.925926     0.142857  0.909091
4                  SVC  0.803279   0.903846  0.870370     0.285714  0.886792

Confusion Matrix for Logistic Regression:
[[ 0  7]
 [ 4 50]]

Confusion Matrix for Decision Tree:
[[ 2  5]
 [ 9 45]]

Confusion Matrix for Random Forest:
[[ 2  5]
 [ 4 50]]

Confusion Matrix for KNN:
[[ 1  6]
 [ 4 50]]

Confusion Matrix for SVC:
[[ 2  5]
 [ 7 47]]
```

Random Forest emerges as the top performer, with the highest accuracy (85.25%) and F1-score (0.9174). Logistic Regression has 0% specificity, indicating it fails to identify any true negatives. Random Forest appears to be the most balanced and effective model overall, while Logistic Regression, despite high accuracy and recall, completely fails on negative instances.

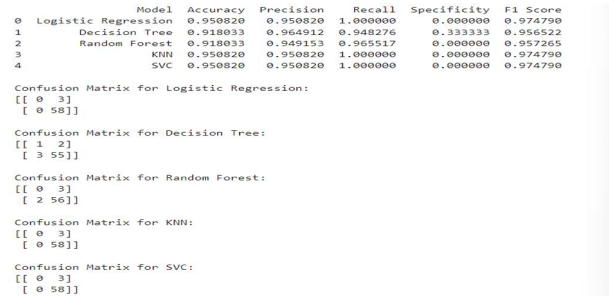
Fig : 30 Treating class 3 as the negative class .Treating all other classes (0, 1, 2, 4) as the positive class.



```
                 Model  Accuracy  Precision    Recall  Specificity  F1 Score
0  Logistic Regression  0.868852   0.942308  0.907407     0.571429  0.924528
1        Decision Tree  0.852459   0.909091  0.925926     0.285714  0.917431
2        Random Forest  0.819672   0.890909  0.907407     0.142857  0.899083
3                  KNN  0.836066   0.879310  0.944444     0.000000  0.910714
4                  SVC  0.819672   0.890909  0.907407     0.142857  0.899083

Confusion Matrix for Logistic Regression:
[[ 4  3]
 [ 5 49]]

Confusion Matrix for Decision Tree:
[[ 2  5]
 [ 4 50]]

Confusion Matrix for Random Forest:
[[ 1  6]
 [ 5 49]]

Confusion Matrix for KNN:
[[ 0  7]
 [ 3 51]]

Confusion Matrix for SVC:
[[ 1  6]
 [ 5 49]]
```

models appear to outperform others in terms of accuracy - in particular, Random Forest and Support Vector Machines both have an accuracy of 81.97% while Logistic Regression has the highest accuracy of 86.89%. Given that, all the models do correctly identify a large majority of the instances they work with. In addition, Specificity is consistently low across all models, suggesting difficulty with negative class.

Fig : 31 Treating class 4 as the negative class .Treating all other classes (0, 1, 2, 3) as the positive class.



```
                 Model  Accuracy  Precision    Recall  Specificity  F1 Score
0  Logistic Regression  0.950820   0.950820  1.000000     0.000000  0.974790
1        Decision Tree  0.918033   0.964912  0.948276     0.333333  0.956522
2        Random Forest  0.918033   0.949153  0.965517     0.000000  0.957265
3                  KNN  0.950820   0.950820  1.000000     0.000000  0.974790
4                  SVC  0.950820   0.950820  1.000000     0.000000  0.974790

Confusion Matrix for Logistic Regression:
[[ 0  3]
 [ 0 58]]

Confusion Matrix for Decision Tree:
[[ 1  2]
 [ 3 55]]

Confusion Matrix for Random Forest:
[[ 0  3]
 [ 2 56]]

Confusion Matrix for KNN:
[[ 0  3]
 [ 0 58]]

Confusion Matrix for SVC:
[[ 0  3]
 [ 0 58]]
```

SVC, KNN, and Logistic Regression attained a high accuracy rate of 95.1%. If class 4 is uncommon, you can get high accuracy by mostly predicting the majority class (not class 4). Logistic Regression, KNN, and SVC in the results, always predict the positive class (not class 4) and never predict class 4.

XI. COnclusion

Overall, Logistic Regression and Random Forest emerge as top performers with 97% accuracy for the breast cancer . the accuracy for heart disease dataset is in the range of 69% to 95%. Precision, Recall, and F1 scores are generally high, often above 80%.On the other hand, Specificity on the model is not as high as expected and sometimes equals 0% when the model has not been able to detect the '0' class correctly. That the best overall model performance is obtained when treating class 3 as the negative class, the worst model performance is achieved when class 1 is treated as the negative class.

XII. Discussion

For the heart disease dataset has class imbalance data, with class 0 having 164 samples while the other classes have much fewer samples. This imbalance is affecting the model performance, particularly for minority classes. To improve accuracy and overall model performance can use stratified k-fold cross-validation to ensure that the proportion of samples for each class is roughly the same in each fold and using a mix of oversampling and undersampling techniques.

References

[1] https://scikitlearn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html

[2] F. Tasnim and S. U. Habiba, "A Comparative Study on Heart Disease Prediction Using Data Mining Techniques and Feature Selection," 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), DHAKA, Bangladesh, 2021, pp. 338-341, doi: 10.1109/ICREST51555.2021.9331158. keywords: {Heart;Support vector machines;Radio frequency;Machine learning algorithms;Feature extraction;Random forests;Principal component analysis;heart disease prediction;data mining;feature selection},