```
X.head()
```

```
y.head()
```

Step 5: Apply Feature Scaling

Various Data Scaling Techniques:

1. Normalizer
2. MinMax Scaler
3. Binarizer
4. Standard Scaler

```
# Apply Standard Scaler
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
SSX = scaler.fit_transform(X)
```

Step 6: Train Test Split

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test =
train_test_split(SSX,y,test_size=0.2,random_state=2)
```

```
X_train.shape,y_train.shape
```

Step 7: Build CLASSIFICATION Algorithms

8.1) Logistic Regression

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression(solver='liblinear',multi_class='ovr')
lr.fit(X_train,y_train)
```

8.2) K-Nearest Neighbors Classifier(KNN)

```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(X_train,y_train)
```

8.3) Naive-Bayes Classifier

```
from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()
nb.fit(X_train, y_train)
```

8.4) Support Vector Machine (SVM)

```
from sklearn.svm import SVC
sv = SVC(kernel='linear')
sv.fit(X_train,y_train)
```